Inżynieria Mechatroniczna

Podstawy Sztucznej Inteligencji i Uczenia Głębokiego:

5: Dane i modele – od strony praktycznej

Ziemowit Dworakowski

AGH w Krakowie

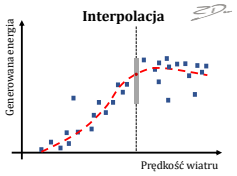
1

Ogólna pętla przetwarzania danych i testowania modeli

2

Różnica pomiędzy klasycznym przetwarzaniem danych i tym opartym na AI

Klasycznie:



3

Jak kiedyś rozwiązywano te problemy (i jak często robi się to nadal…)

Drzewo decyzyjne

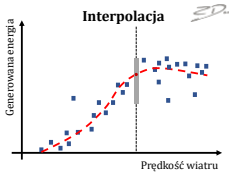
Nie ma limitu dot. złożoności wykonywanych analiz…

Jeśli chcemy automatyzować decyzje – najprościej zacząć od zamiany wybranych kroków na tym drzewie rezultatami modeli uczenia maszynowego lub optymalizować samo drzewo (np. zmieniając progi)

4

Różnica pomiędzy klasycznym przetwarzaniem danych i tym opartym na AI

Klasycznie:

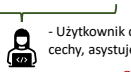


Zalety:
- Kontrolujemy wszystkie zmienne i możemy wyjaśnić tok rozumowania
- Używamy drzew decyzyjnych (prostych koncepcyjnie i intuicyjnych)

Wady:
- Potrzebujemy dużo czasu i dziedzinowego eksperta
- Jeśli wymiarów jest więcej niż 3 – musimy akceptować uproszczenia

5

Uczenie maszynowe („płytkie”, shallow learning)



- Użytkownik definiuje zadanie, przygotowuje zbiory i cechy, asystuje w treningu modelu

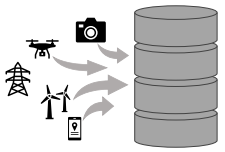
Skupmy się na razie na tym obszarze

Uczenie głębokie (ang. deep learning)

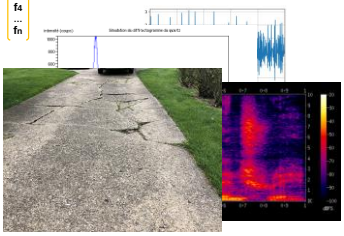
- Użytkownik definiuje zadanie, asystuje w treningu modelu

6

Źródła danych



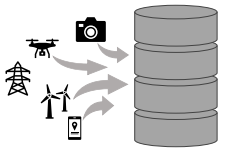
f1 Timestamp („Znacznik czasu")
f2 Temperatura zewnętrzna
f3 Amplituda drgań
f4
…
fn



Zbieramy:
- Wektory zmiennych (cech)
- Sygnały 1D (np. czasowe)
- Sygnały 2D (np. obrazy)
- Metadane i zmienne opisowe

7

Źródła danych



Nie ma znaczenia, czy pracujemy z sygnałami, danymi SCADA, widmami, obrazami, sygnaturami drgań… Ostatecznie i tak musimy je zamienić na wektory cech.

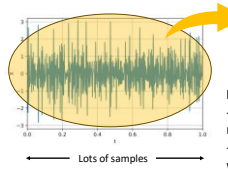
Zbieramy:
- Wektory zmiennych (cech)
- Sygnały 1D (np. czasowe)
- Sygnały 2D (np. obrazy)
- Metadane i zmienne opisowe

Ekstrakcja cech

Cechy to jedyny typ danych który może być używany przez płytkie modele AI

8

Ekstrakcja cech



f1
f2
f3
f4
…
fn

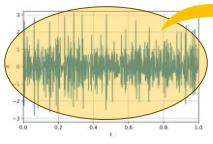
Które cechy zawierają znaczącą informację?

Idealna cecha powinna być:
- Wrażliwa wyłącznie na wartości docelowe
- Niewrażliwa na inne zmienne i warunki eksperymentalne

Zwykle trudno takie znaleźć

9

Ekstrakcja cech



f_1
f_2
f_3
f_4
...
f_n

Które cechy zawierają znaczącą informację?

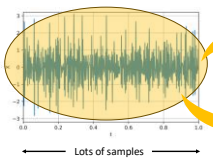
Akceptowalna cecha powinna:
- Być wrażliwa na niektóre wartości docelowe
- Być wrażliwa na warunki eksperymentalne – pozwalając na ich filtrację
- Nie być zbyt zaszumiona
- Zawierać jakąś unikatową informację

To nie wygląda jak jasny przepis na poszukiwanie cech…

10

Ekstrakcja cech

Zamiast tego użyjmy jednego z poniższych sposobów:



Wyznaczmy więcej cech niż potrzebujemy a potem dokonajmy **selekcji** podzbioru który wydaje się działać najlepiej

To wygląda jak problem optymalizacyjny…

Pomyślmy, jak można rozpoznać klasy „ręcznie” a potem zastosujmy cechy, które na to pozwalają

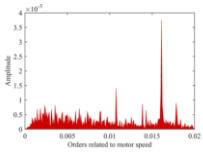
a to wymaga wiedzy dziedzinowej…

Tak czy owak – będziemy wykonywali dużo zaawansowanego przetwarzania sygnałów, by uzyskać cechy które faktycznie pozwalają na klasyfikację

11

Ekstrakcja cech

Wykonajmy kilka przykładów:
Zapis drgań w czasie – do rozpoznania stanu technicznego maszyny



RMS zawiera informację o energii zawartej w sygnale

Kurtoza informuje jak bardzo „poszarpany” jest sygnał

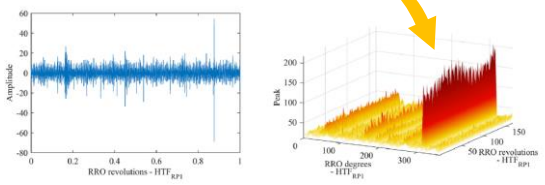
Wyznaczenie widma częstotliwościowego albo rzędów pozwala na wyznaczenie energii w pasmach, która może korelować z konkretnymi typami uszkodzeń

Jeśli poszukujemy konkretnych uszkodzeń, możemy zaprojektować cechy dostosowane do ich wykrywania.

12

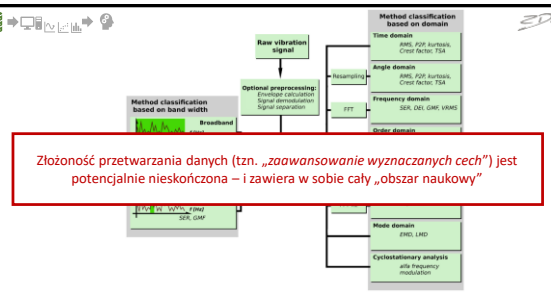
Ekstrakcja cech

Ten grzbiet odnosi się do uszkodzenia konkretnego zęba konkretnego elementu przekładni obiegowej



Źródło: K. Dziedziech, A. Jablonski, and Z. Dworakowski, "Epicyclic gearbox fault detection by Instantaneous Circular Pitch Cycle Map," *Mech. Syst. Signal Process.*, vol. 121, pp. 600–614, 2019

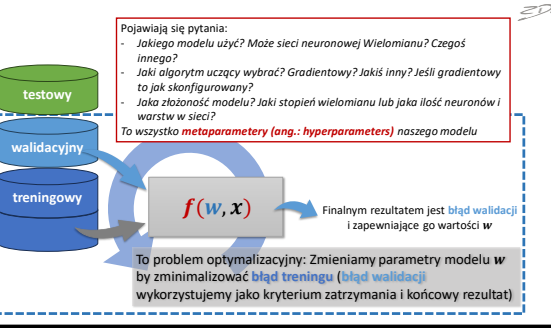
13



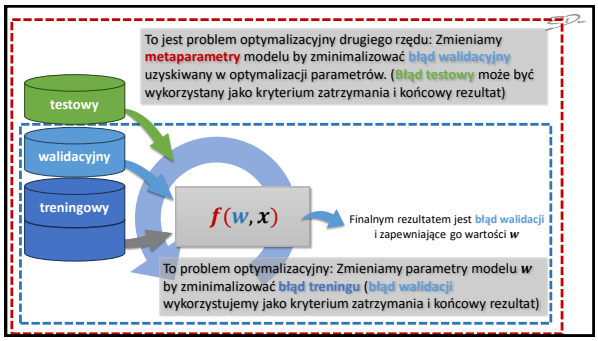
Złożoność przetwarzania danych (tzn. *„zaawansowanie wyznaczanych cech"*) jest potencjalnie nieskończona – i zawiera w sobie cały „obszar naukowy"

Źródło: A. Jablonski, Z. Dworakowski, K. Dziedziech, and F. Chaari, "Vibration-based diagnostics of epicyclic gearboxes – From classical to soft-computing methods," *Measurement*, vol. 147, p. 106811, 2019

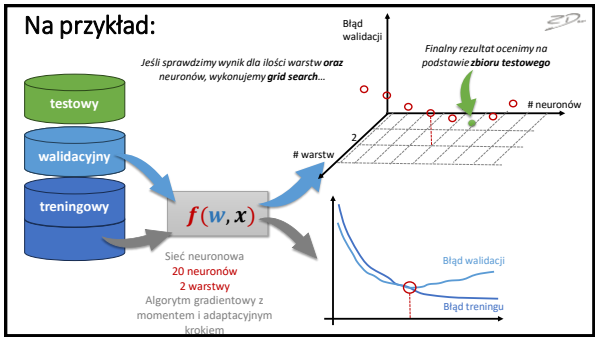
14



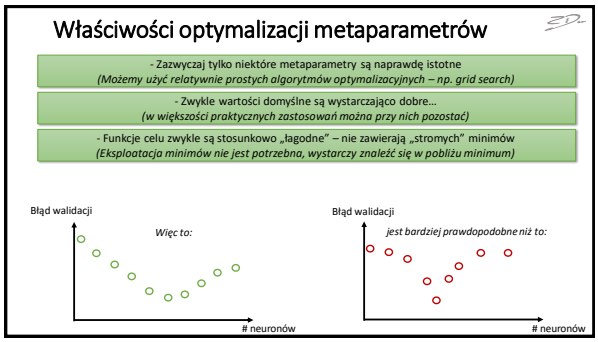
15


To jest problem optymalizacyjny drugiego rzędu: Zmieniamy metaparametry modelu by zminimalizować błąd walidacyjny uzyskiwany w optymalizacji parametrów. (Błąd testowy może być wykorzystany jako kryterium zatrzymania i końcowy rezultat)
testowy
walidacyjny
treningowy
f(w, x)
Finalnym rezultatem jest błąd walidacji i zapewniające go wartości w
To problem optymalizacyjny: Zmieniamy parametry modelu w by zminimalizować błąd treningu (błąd walidacji wykorzystujemy jako kryterium zatrzymania i końcowy rezultat)

16


Na przykład:
Błąd walidacji
Jeśli sprawdzimy wynik dla ilości warstw oraz neuronów, wykonujemy grid search...
Finalny rezultat ocenimy na podstawie zbioru testowego
testowy
walidacyjny
treningowy
neuronów
warstw
2
f(w, x)
Sieć neuronowa
20 neuronów
2 warstwy
Algorytm gradientowy z momentem i adaptacyjnym krokiem
Błąd walidacji
Błąd treningu

17


Właściwości optymalizacji metaparametrów
- Zazwyczaj tylko niektóre metaparametry są naprawdę istotne
(Możemy użyć relatywnie prostych algorytmów optymalizacyjnych – np. grid search)
- Zwykle wartości domyślne są wystarczająco dobre...
(w większości praktycznych zastosowań można przy nich pozostać)
- Funkcje celu zwykle są stosunkowo „łagodne" – nie zawierają „stromych" minimów
(Eksploatacja minimów nie jest potrzebna, wystarczy znaleźć się w pobliżu minimum)
Błąd walidacji
Więc to:
neuronów
Błąd walidacji
jest bardziej prawdopodobne niż to:
neuronów

18

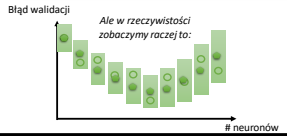
Właściwości optymalizacji metaparametrów

- Zazwyczaj tylko niektóre metaparametry są naprawdę istotne
(Możemy użyć relatywnie prostych algorytmów optymalizacyjnych – np. grid search)

- Zwykle wartości domyślne są wystarczająco dobre…
(w większości praktycznych zastosowań można przy nich pozostać)

- Funkcje celu zwykle są stosunkowo „łagodne" – nie zawierają „stromych" minimów
(Eksploatacja minimów nie jest potrzebna, wystarczy znaleźć się w pobliżu minimum)

- Funkcje celu są niedeterministyczne
(kilkukrotne uruchomienie klasyfikacji daje różne efekty dla tych samych metaparametrów)



19

Właściwości optymalizacji metaparametrów

- Zazwyczaj tylko niektóre metaparametry są naprawdę istotne
(Możemy użyć relatywnie prostych algorytmów optymalizacyjnych – np. grid search)

- Zwykle wartości domyślne są wystarczająco dobre…
(w większości praktycznych zastosowań można przy nich pozostać)

- Funkcje celu zwykle są stosunkowo „łagodne" – nie zawierają „stromych" minimów
(Eksploatacja minimów nie jest potrzebna, wystarczy znaleźć się w pobliżu minimum)

- Funkcje celu są niedeterministyczne
(kilkukrotne uruchomienie klasyfikacji daje różne efekty dla tych samych metaparametrów)

- Każdy poziom optymalizacji wymaga wykładniczo dłuższego czasu (i kolejnego podzbioru danych)
(Jeśli próbujemy znaleźć „idealne metaparametry", poświęcamy czas i zużywamy dane które można było wykorzystać do lepszego nauczenia modelu…)

20

Więc jak się to w praktyce robi?

Sposób 1: Upewnij się, że model jest wystarczająco złożony by rozpoznać wzorce w danych – i zastosuj wartości domyślne – lub skopiuj wartości metaparametrów z działającego rozwiązania problemu podobnej klasy
(Oszczędza czas, nie wymaga dużo danych)

Sposób 2: Wybierz rozsądny punkt startowy dla optymalizacji metaparametrów (wartości domyślne?) a następnie zmieniaj metaparametry po jednym próbując uzyskać statystycznie istotną poprawę
(Przeznaczamy czas w miarę potrzeb, ryzykujemy zakończenie optymalizacji w minimum lokalnym lub stagnację. Efektywnie ten sposób to algorytm 1+1)

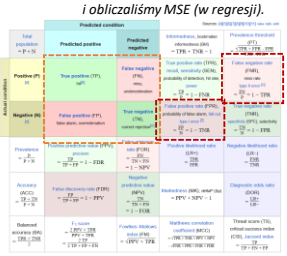
Sposób 3: Zdecyduj, które metaparametry mają największy wpływ na wyniki – i wykonaj ich pełną optymalizację (np. z zastosowaniem metody grid search)
(Ryzykujemy poświęcenie długiego czasu i otrzymanie wniosku, iż te wartości nie są bardzo istotne)

Sposób 4: Wykonaj jednowymiarową optymalizację dla każdego metaparametru niezależnie
(Łatwa i szybka optymalizacja, ale wysokie ryzyko zakończenia w minimum lokalnym. Poszczególne metaparametry zwykle znacznie na siebie wpływają)

21

I wreszcie – ocena modeli!

Jak do tej pory, sumowaliśmy jedynie błędy (w klasyfikacji) i obliczaliśmy MSE (w regresji).



W klasyfikacji, poza **skutecznością (accuracy)** czyli uśrednioną wartością błędów, dwie (chyba) najważniejsze miary to procentowa szansa na pojawienie się błędów typu „pozytywnego" i „negatywnego" w każdej klasie:

False positive rate (FPR) i **False negative rate (FNR)**

Jak często obiekt z innej klasy ma przypisaną tą etykietę?

Jak często obiekt z tej klasy nie ma przypisanej tej etykiety?

22

Tablica pomyłek (confusion matrix)

	Prawdziwe A (100)	Prawdziwe B (200)	Prawdziwe C (100)	Prawdziwe D (20)
Zwrócone A	100	10	30	0
Zwrócone B	0	150	10	0
Zwrócone C	0	0	60	0
Zwrócone D	0	40	0	20

Tablicy pomyłek używamy do:

- Wyznaczenia prawdopodobieństwa poprawnej klasyfikacji
- Szacowania prawdpodobieństwa obecności klasy
- Planowania eksperymentów (które klasy wymagają więcej danych)
- Dostrajania modelu (które klasy wymagają wyższej skuteczności)

23

A co z regresją?

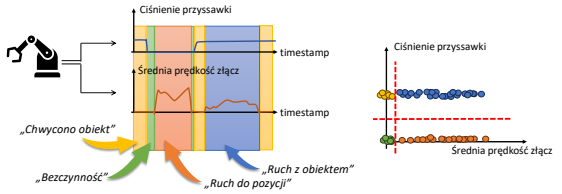
MAE (Mean absolute error) ← *Łatwa interpretacja (jednostka!)*

MSE (Mean squared error) ← *Skupiony na „dużych błędach"*

RMSE (Root mean squared error) ← *Skupiony na „dużych błędach" ale nieco łatwiejszy do interpretacji*

24

Zastosowanie w rozpoznawaniu stanu pracy



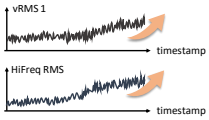
Możemy oczywiście dysponować większą ilością zmiennych operacyjnych (tworząc wielowymiarową przestrzeń cech). Tak czy owak,

To zadanie klasyfikacyjne – które umiemy rozwiązać!

25

Zastosowanie w predykcji (np. RUL)

Powiedzmy, że mamy maszynę wirnikową, dla której mierzymy takie cechy:



Wzrost wartości vRMS oraz drgań w wysokich częstotliwościach sugeruje zużywanie się łożysk…

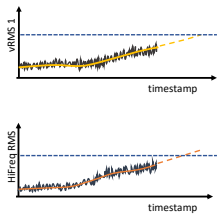
Więc pewnie ulegną uszkodzeniu…

Kiedyś…

Ale kiedy?

26

Zastosowanie w predykcji (np. RUL)



Możemy sprawdzić na jakim poziomie wartości cech zwykle następuje uszkodzenie a następnie wpisać w punkty jakiś model przewidujący trend – i sprawdzić kiedy linie się przetną…

To zadanie regresji, umiemy takie rozwiązywać!

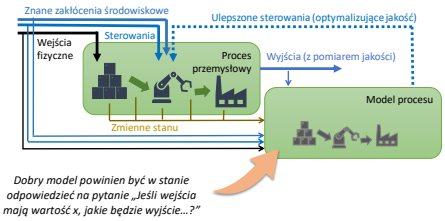
Lub nawet lepiej:

Możemy zebrać dane łączące wartości mierzonych cech z pozostałym czasem życia (Remaining Useful Life,RUL) łożyska – i potem wykonać zadanie regresji przewidując ten czas

Co również umiemy zrobić!

27

Zastosowanie w modelowaniu procesów

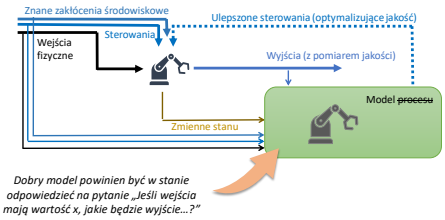


Dobry model powinien być w stanie odpowiedzieć na pytanie „Jeśli wejścia mają wartość x, jakie będzie wyjście…?"

A to też zadanie regresji, które umiemy rozwiązywać!

28

Zastosowanie w identyfikacji



Dobry model powinien być w stanie odpowiedzieć na pytanie „Jeśli wejścia mają wartość x, jakie będzie wyjście…?"

A to też zadanie regresji, które umiemy rozwiązywać!

29

Jednoklasowa klasyfikacja (detekcja anomalii)

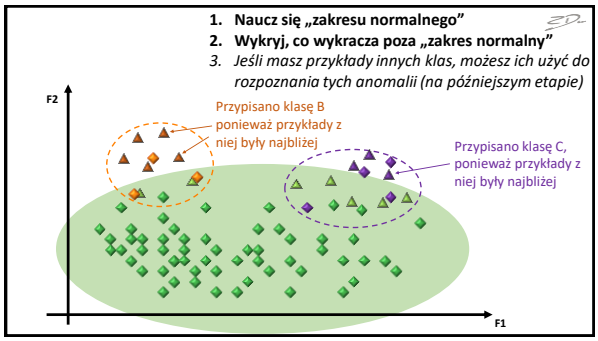
W niektórych problemach mamy nadmiar obiektów należących do jednej klasy i znaczny niedobór tych należących do innej:

- „**Zdrowi** ludzie"
- „**Normalne** zachowanie w miejscu publicznym"
- „**Normalny** stan operacyjny linii montażowej
- „**Typowa** pogoda we wrześniu"

Wówczas naszym celem może być modelowanie zakresu **normalnego** i wykrywanie wszystkich odstępstw od normy **(anomalii, outlierów)**

W tym zadaniu zwykle „nie wiemy czego oczekiwać" – tzn. istnieje potencjalnie nieskończony zbiór „odstępstw od normy"

30



31

Jednoklasowa klasyfikacja (detekcja anomalii)

Jak zdefiniować „zakres normalny”?

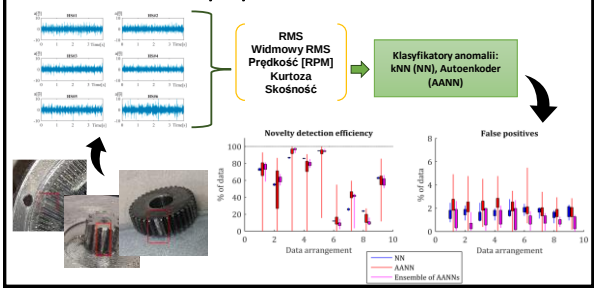
1. Możemy określić maksymalny dystans pomiędzy próbkami w zbiorze normalnym i wykrywać punkty, które go znacząco przekraczają > np. **kNN**

2. Możemy estymować gęstość prawopodobieństwa obecności próbek w przestrzeni cech – a potem wykrywać próbki, które znajdują się w obszarach zerowego prawdopodobieństwa -> np. **GMM**

3. Możemy wytrenować system regresyjny by przewidywał jedną cechę na podstawie pozostałych. Dla znanych danych (w zakresie normalnym) powinien działać skutecznie, dla nietypowych danych (anomalii) będzie generował duże błędy -> np. **sieć MLP** lub **autoenkoder**

32

Zastosowanie w wykrywaniu anomalii



33

(Ponieważ działają na podstawie zbioru danych pozyskanych z jakiegoś rzeczywistego rozkładu. Przy ponownym pozyskaniu danych rezultaty będą inne!)

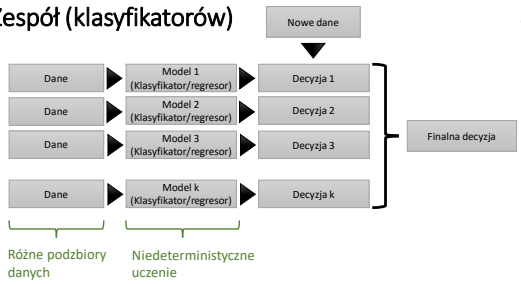
Klasyfikatory i regresory powinny być **zawsze** traktowane jako niedeterministyczne

Jak zachować wysoką powtarzalność?

(Jak się upewnić, że wyniki które uzyskujemy nie wynikają z przypadkowego szczęścia lub pecha w rozkładzie danych?)

34

Zespół (klasyfikatorów)



Im bardziej **różnorodne** i **dokładne** modele, tym lepiej

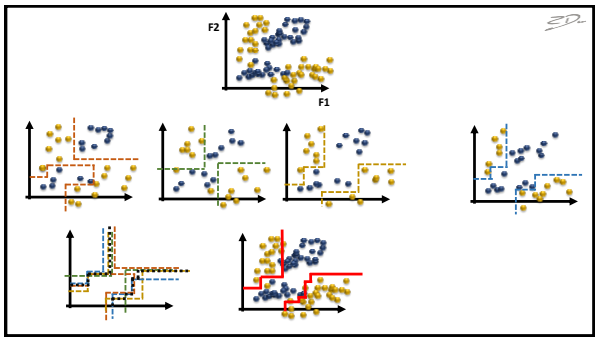
35

Las losowy

1. Wykonaj agregację bootstrapową (**Bagging, Bootstrap Aggregating)** poprzez losowanie (ze zwracaniem) wielu małych podzbiorów danych uczących
2. Jeśli zbiór danych zawiera dużo cech, wykonaj losowanie również w ich zakresie (tzn. niech różne podzbiory korzystają z przypadkowego podzbioru cech)
3. Wytrenuj proste drzewa decyzyjne na tych podzbiorach (każde drzewo używa innego podzbioru)
4. Uśrednij odpowiedzi z wszystkich drzew

Im bardziej **różnorodne** i **dokładne** modele, tym lepiej

36



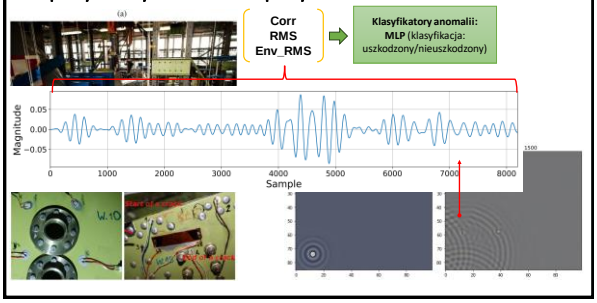
37

Zespoły klasyfikatorów

Klasyfikacja zespołowa zapewnia
zwiększoną stabilność (powtarzalność) kosztem:
- braku bezpośredniej kontroli nad procesem klasyfikacji,
- braku możliwości optymalizacji struktury klasyfikatorów, oraz
- dużego obciążenia pamięci i dużej mocy obliczeniowej

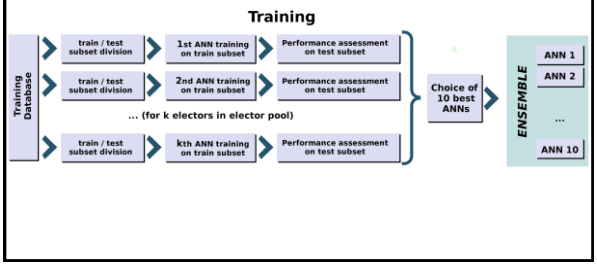
38

Zespoły klasyfikatorów: przykład



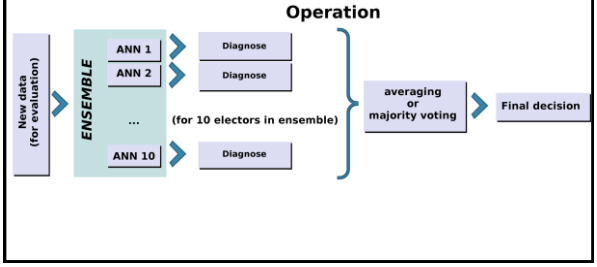
39

Zespoły klasyfikatorów: przykład



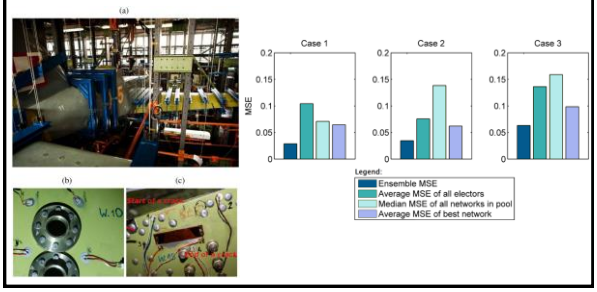
40

Zespoły klasyfikatorów: przykład



41

Zespoły klasyfikatorów: przykład



42

Materiał do powtórki:

1. Jak działa drzewo decyzyjne i jakie ma cechy?
2. Jakie są źródła danych w inżynierii i jak można je przetwarzać w celu wydobycia cech (na przykładach)?
3. Zaprezentuj schemat treningu klasyfikatora ze wskazaniem celów dwóch poziomów optymalizacji
4. Jakie są charakterystyczne właściwości problemu optymalizacji metaparametrów i jakie są sposoby jego rozwiązywania?
5. Jakie miary jakości są wykorzystywane w klasyfikacji i regresji, jak działa i do czego służy tablica pomyłek?
6. Przedstaw przykłady praktycznego stosowania metod uczenia maszynowego w identyfikacji, predykcji, rozpoznawaniu stanów operacyjnych, modelowaniu procesów i detekcji anomalii.
7. Jak działa klasyfikacja zespołowa (Na przykładzie lasu losowego), jakie ma wady i zalety?

43