AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

Porównanie jakości uogólnienia i efektywności działania różnych metod klasyfikacji z sieciami neuronowymi SONN

Autor: Łukasz Dziedzia

Promotor: dr Adrian Horzyk

Motywacja

Lawinowy przyrost informacji wymaga rozwijania skutecznych i możliwie automatycznych algorytmów analizujących dane.

Sieci SONN – ciekawa koncepcja dotycząca klasyfikacji danych rozwijana na AGH.

Własne zainteresowania oraz doświadczenie.

Cel pracy


Kompleksowe porównanie sieci SONN z wybranymi metodami klasyfikacji.
Utworzenie aplikacji pozwalające na dokonywanie porównań metod klasyfikacji w oparciu o zaproponowaną metodykę.
Implementacja nowoczesnej biblioteki zawierającej wybrane metody klasyfikacji.

Eksploracja danych

Eksploracja danych(*ang. Data Mining*) – dziedzina dostarczająca metody pozwalające na znajdowanie ukrytych prawidłowości w dużych zbiorach danych (niemożliwych do analizy przez człowieka).

Słowa kluczowe: OLAP(Online Analytical Processing), Machine Learning, KDD (Knowledge Discovery in Databases), data warehouses

Metody eksploracji danych

- Regresja – to proces wyszukiwania funkcji modelującej dane z najmniejszym błędem
- Reguły asocjacyjne – to proces wyszukiwania zależności między danymi (np. analiza koszykowa)
- Klasteryzacja – to proces grupowania danych i szukania tych grup na podstawie różnych relacji - zwykle podobieństwa
- **Klasyfikacja –** to proces przyporządkowywania danych do określonych klas na podstawie pewnych wspólnych cech definiujących poszczególne klasy

Metody klasyfikacji

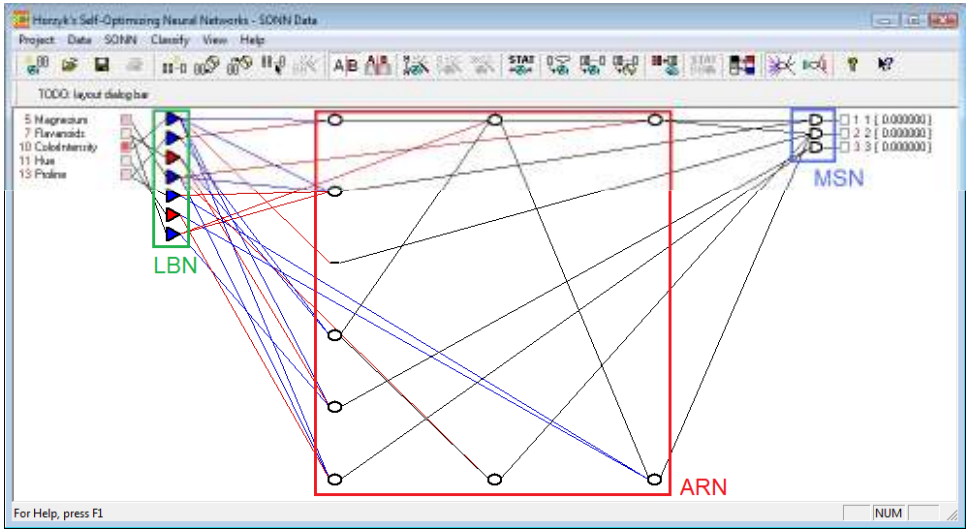
- **K najbliższych sąsiadów**
- **RBFN**
- **SVM**
- **PNN**
- **SONN**
- Naiwny klasyfikator Bayes'a
- Drzewa decyzyjne
- MLP
- Fuzzy logic
- Rough sets
- …

Sieci neuronowe

Sieci neuronowe są fascynujące, ale…

Problem:

Konstruując sieć neuronową konieczne jest określenie struktury sieci i jej parametrów.



Rozwiązanie:

Ontogeniczne sieci neuronowe – potrafią dopasować swoją strukturę do stawianego problemu

Ontogeniczne sieci neuronowe SONN

SONN (*Self-Optimizing Neural Networks*) – koncepcja ontogenicznych sieci neuronowych opracowywana od 1999 roku przez dr Adriana Horzyka.

SONN jest zestawem różnego rodzaju algorytmów optymalizujących strukturę i parametry sieci neuronowej dla podanego zbioru uczącego pod kątem klasyfikacji.

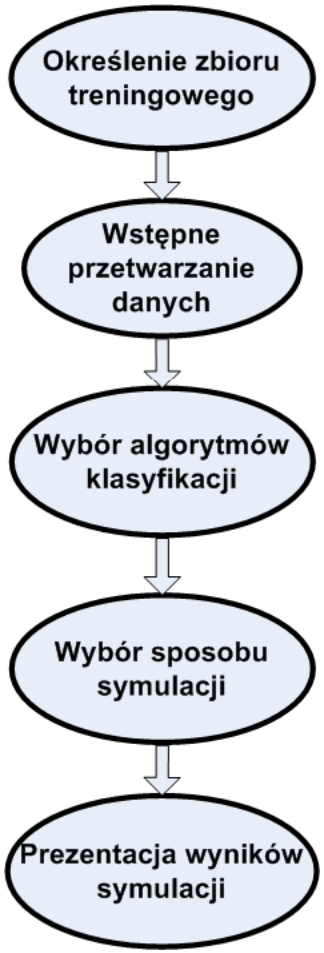
Wybrane zagadnienia:

- Kwantyzacja połączona z binaryzacją danych warstwy wejściowej
- Wybór znaczących cech
- Redukcja wymiaru przestrzeni cech
- **Automatyczny dobór parametrów i automatyczna konstrukcja topologii sieci**

Kryteria porównania

- **Jakość uogólnienia** (predictive accuracy) – umiejętność klasyfikacji
- **Efektywność działania** – czas treningu (klasyfikacja jest z reguły pomijalnie mała) oraz wymagania pamięciowe
- Interpretowalność – zrozumiałość mechanizmu klasyfikacji na podstawie modelu klasyfikatora
- Odporność – możliwość radzenia sobie z danymi zaszumionymi lub niepełnymi
- Skalowalność – możliwość stosowania klasyfikatora do danych różnej wielkości (np. niektóre klasyfikatory są binarne)
- Inne kryteria – często związane z konkretną dziedziną (np. analiza giełdowa, medyczna, etc.)

Proces porównywania metod klasyfikacji



Wybranie zbioru danych (problemu klasyfikacji). Import z różnych źródeł

Wstępne przetwarzanie danych. Wybór najbardziej znaczących cech, normalizacja, itp. Dzięki takim operacjom algorytmy klasyfikacji potrafią osiągać lepszą skuteczność lub szybciej tworzyć klasyfikator.

Wybór porównywanych algorytmów klasyfikacji. Możliwa dowolna konfiguracja pozwalająca na testowanie różnie sparametryzowanych algorytmów tego samego typu.

Określenie metod walidacji zastosowanych do porównania algorytmów. Ustawienia związane z poziomem szczegółowości zbieranych statystyk.

Przedstawienie obszernych statystyk. Wskaźniki jakości oraz efektywności działania, macierz niepewności, szczegóły utworzonego klasyfikatora, analiza danych użytych w procesie porównania.

Zaimplementowana biblioteka algorytmów

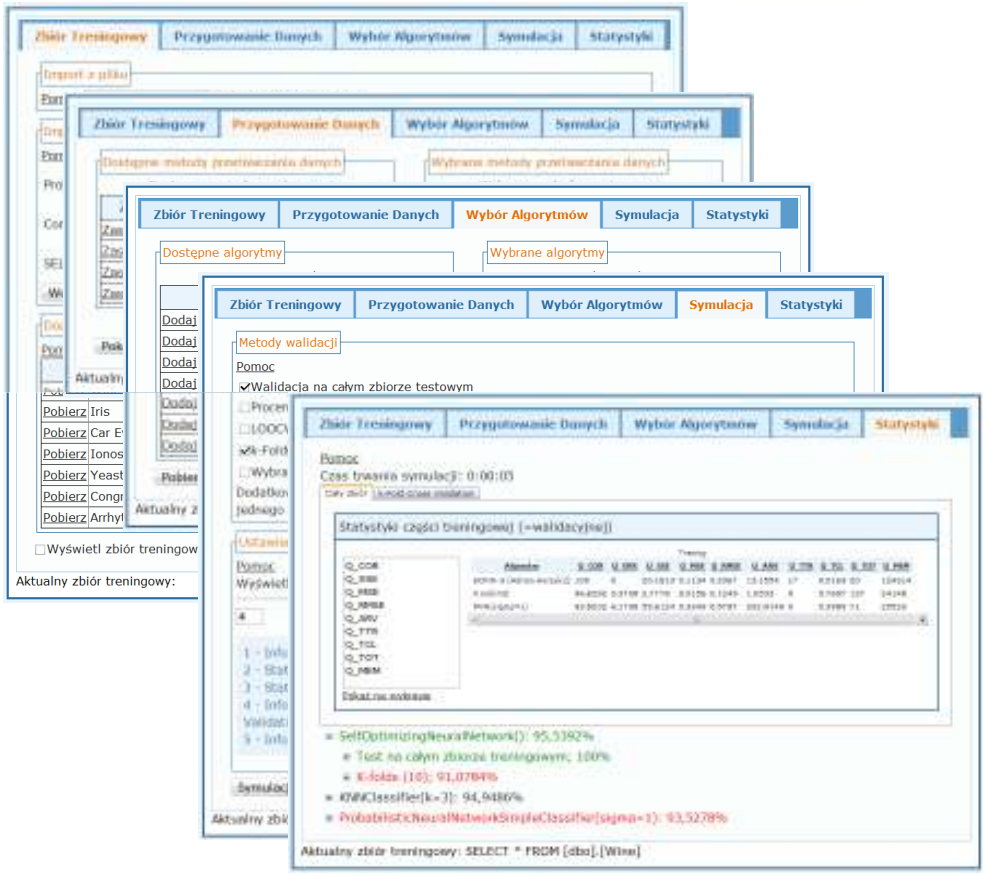
```
public interface IAlgorithm
{
    void Train(ITrainingData trainingData);
    IClassificationResults Classify(double[] testData);
}


public interface IAlgorithmInfoProvider
{
    Dictionary<string, string> SpecificInformation { get; }
}


public interface IClassificationResults
{
    Dictionary<string, double> ClassResults { get; }
    KeyValuePair<string, double> ClassifiedAs { get; }

    double GetResultForClass(string className);
}
```

Rozszerzalność (zestaw interfejsów)

Wysoka jakość (inżynieria oprogramowania)

Kompatybilność (możliwość importu danych z rozmaitych źródeł)

Aplikacja wspierająca proces porównywania zrealizowana w ramach pracy



- **Dostępność** (aplikacja internetowa)
- **Funkcjonalność** (zaprojektowana pod kątem porównywania metod klasyfikacji)
- **Intuicyjność** (czytelne kroki, system pomocy)

Jakość uogólnienia dla analizowanych zbiorów – dane treningowe

Algorytm	Iris	Wine	Congressional Voting	Ionosphere	Car Evaluation	Yeast	Arrhythmia
SONN-3	99,85	100,00	100	97,28	100,00	100,00	100,00
KNN(k=1)	100	100,00	100	100	100,00	100,00	100,00
KNN(k=3)	95,04	96,01	92,49	85,82	88,68	49,19	50,49
KNN(k=5)	93,11	92,82	91,16	82,4	86,22	48,29	49,78
KNN(k=10)	93,41	91,13	91,24	78,32	82,83	44,41	47,20
KNN(k=15)	89,48	87,76	89,22	75,25	79,79	44,53	44,86
KNN(k=30)	78,44	83,08	88,48	69,39	72,40	42,06	44,08
SVML(c=1;e=0,1)	85,7	99,06	97,32	91,61	80,57	50,44	86,16
SVML(c=10;e=0,1)	93,93	99,88	97,42	94,14	81,83	51,13	95,65
SVML(c=20;e=0,01)	94,22	99,81	97,42	94,37	82,29	49,65	97,22
SVMR(c=10;s=1;e=0,1)	97,78	100,00	99,82	99,34	97,99	60,57	100,00
SVMR(c=10;s=5;e=0,1)	78,22	98,81	96,7	93,99	81,08	54,28	83,46
SVMR(c=10;s=10;e=0,1)	66,67	98,56	95,81	90,15	75,00	50,85	73,85
RBFN(k=5;f=2;c=1)	95,04	97,75	91,39	78,6	76,95	50,13	59,00
RBFN(k=10;f=2;c=1)	98,3	99,19	93,92	89,43	81,98	57,77	62,44
RBFN(k=5;f=1;c=2)	94,81	98,56	93,44	89,55	78,11	46,80	59,41
RBFN(k=10;f=1;c=2)	97,48	99,44	95,25	91,55	81,51	57,70	61,92
RBFN(k=15;f=1;c=2)	98,52	99,69	96,12	92,28	83,40	59,31	64,13
RBFN(k=20;f=1;c=2)	98,52	99,81	96,48	92,5	85,87	60,16	65,71
RBFN(k=5;f=2;c=2)	96,07	96,94	91,85	79,52	76,43	51,34	59,76
PNN(sigma=0,1)	97,85	100,00	100	100	100,00	63,61	100,00
PNN(sigma=1)	91,33	94,13	90,88	78,92	47,66	49,56	74,85
PNN(sigma=10)	91,04	92,82	89,37	65,37	41,60	48,07	51,50
PNN(sigma=100)	91,04	92,82	89,35	65,31	41,53	48,05	51,28

Wartości wskaźnik Q_COR(avg) dla wszystkich analizowanych zbiorów.

Metoda walidacji: 10-fold cross-validation, dane treningowe.

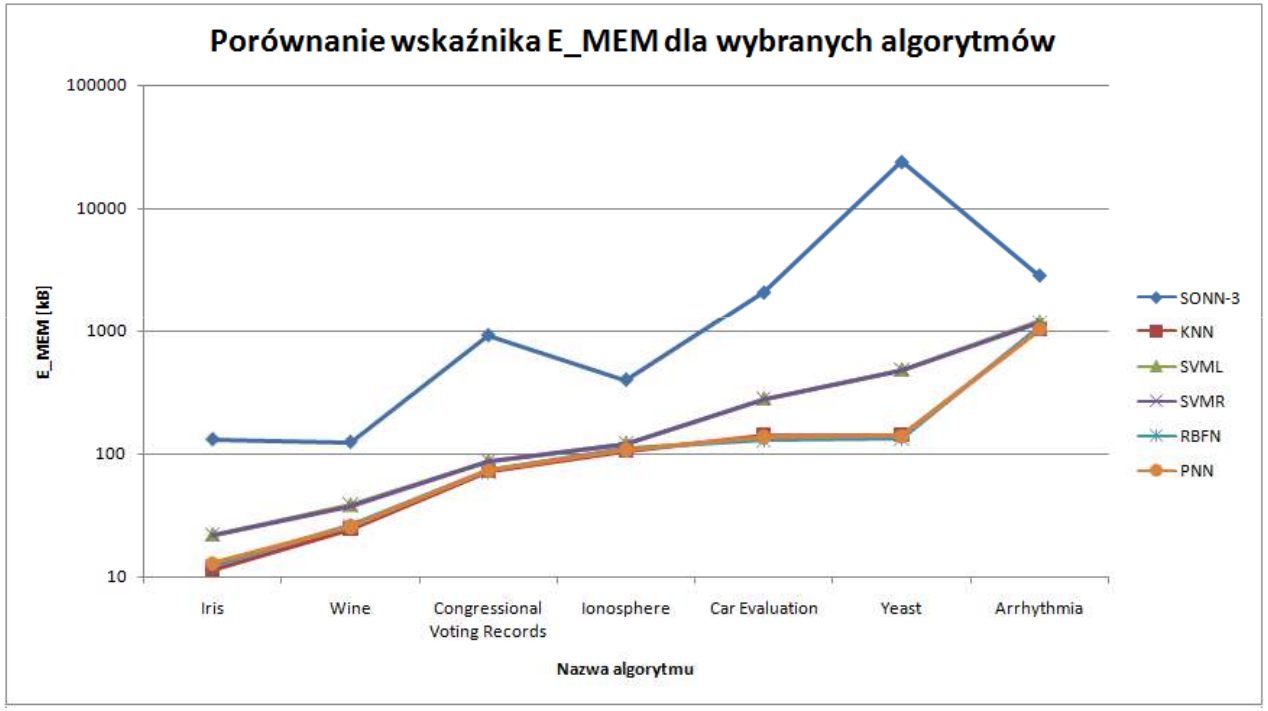
Jakość uogólnienia dla analizowanych zbiorów – dane nieznane

Algorytm	Iris	Wine	Congressional Voting	Ionosphere	Car Evaluation	Yeast	Arrhythmia
SONN-3	91,33	91,08	94,71	86,62	94,39	35,92	43,55
KNN(k=1)	95,33	95,52	92,88	86,90	91,09	52,30	53,55
KNN(k=3)	94,67	93,27	92,39	83,48	87,96	47,24	48,90
KNN(k=5)	94,67	88,73	91,93	80,35	85,18	47,30	50,86
KNN(k=10)	93,33	90,39	90,32	76,92	82,41	44,68	44,45
KNN(k=15)	88,67	88,79	89,20	74,08	76,91	44,81	46,23
KNN(k=30)	82,67	84,87	88,94	67,26	70,78	44,88	44,23
SVML(c=1;e=0,1)	85,33	97,71	96,31	87,17	80,21	48,79	71,02
SVML(c=10;e=0,1)	91,33	98,30	95,85	89,44	81,42	49,59	64,84
SVML(c=20;e=0,01)	92,00	98,30	95,85	88,87	82,00	48,99	63,52
SVMR(c=10;s=1;e=0,1)	96,67	97,75	95,86	95,15	96,82	58,89	62,62
SVMR(c=10;s=5;e=0,1)	76,67	97,16	95,85	90,87	81,31	52,97	70,79
SVMR(c=10;s=10;e=0,1)	66,67	98,30	95,39	87,75	74,94	50,74	67,46
RBFN(k=5;f=2;c=1)	96,00	97,78	89,87	74,63	77,14	50,67	58,38
RBFN(k=10;f=2;c=1)	96,00	98,86	93,77	86,63	81,71	57,34	60,61
RBFN(k=5;f=1;c=2)	94,67	98,86	91,48	88,88	78,36	45,36	57,94
RBFN(k=10;f=1;c=2)	94,67	98,89	94,69	90,59	81,77	56,67	60,39
RBFN(k=15;f=1;c=2)	97,33	98,89	96,53	92,02	83,45	58,15	61,28
RBFN(k=20;f=1;c=2)	96,67	98,30	95,17	92,29	85,30	58,29	62,15
RBFN(k=5;f=2;c=2)	96,00	96,11	91,95	77,76	76,56	50,60	56,83
PNN(sigma=0,1)	96,00	95,52	93,10	88,04	92,88	56,54	51,78
PNN(sigma=1)	90,67	93,24	90,32	73,50	47,86	48,78	47,58
PNN(sigma=10)	90,67	92,12	88,95	64,98	41,49	47,10	49,76
PNN(sigma=100)	90,67	92,12	88,95	64,98	41,38	47,10	49,76

Wartości wskaźnik Q_COR(avg) dla wszystkich analizowanych zbiorów.

Metoda walidacji: 10-fold cross-validation, dane nieznane.

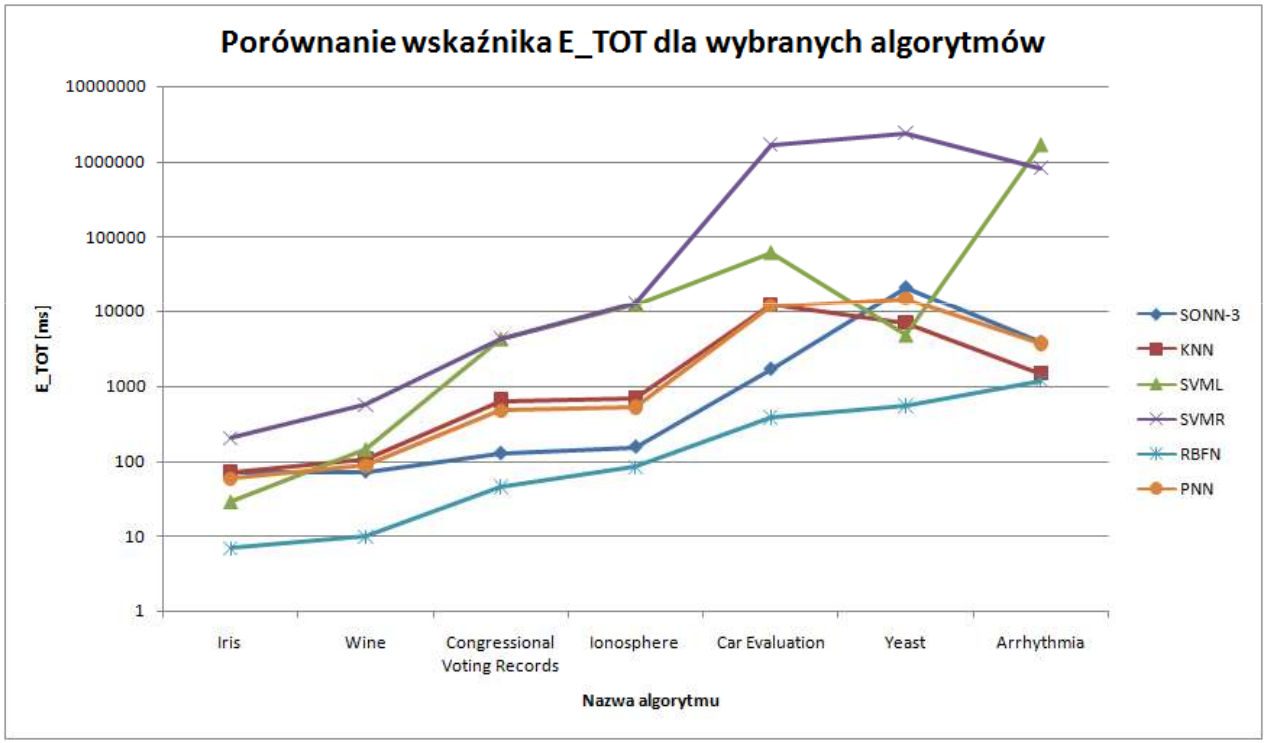
Analiza wskaźnika E_MEM



Wymagania pamięciowe sieci SONN są znacznie większe od pozostałych metod.

Sieci SONN potrafią automatycznie zredukować wymiar przestrzeni cech (np. Arrhythmia)

Analiza wskaźnika E_TOT



Algorytm RBFN okazał się najszybszym dla każdego zbioru.

Algorytm SONN uzyskuje bardzo konkurencyjne wyniki w zakresie czasu tworzenia sieci.

Czas tworzenia klasyfikatorów SVM związany jest z ilością klas problemu klasyfikacji.

Wnioski

- Jakoś uogólnienia sieci SONN okazała się porównywalna z większością analizowanych algorytmów.
- Wymagania pamięciowe metody SONN są zdecydowanie większe od porównywanych metod (ok. 10-100 krotnie większe zużycie pamięci).
- Sieci SONN okazały się niewrażliwe na dysproporcje w wielkościach wartości cech (zasługa algorytmu kwantyzacji i binaryzacji ADLBCA).
- Sieci SONN osiągają najgorsze rezultaty dla zbiorów danych charakteryzujących się nierównomiernym rozkładem wzorców według klas (Yeast, Arrhytyhmia).
- Sieci SONN osiągają bardzo dobre wyniki dla zbioru CarEvaluation. Ponieważ zbiór ten składa się z cech o wartościach symbolicznych można przypuszczać, że dla takich zbiorów kolejne wersje sieci SONN będą osiągały równie zadowalające rezultaty.
- Niektóre algorytmy nie potrafiły utworzyć klasyfikatora (zbyt długi czas lub brak możliwości ukończenia obliczeń) dla konkretnych zestawów parametrów (problem ten nie występuje dla SONN).
- Analiza czasu klasyfikacji obok czasu treningu okazała się uzasadniona (algorytmy *lazy*).
- Ilość klas problemu klasyfikacji dla niektórych algorytmów (np. SVM) znacząco wpływa na czas obliczeń.

Koniec

Dziękuję za uwagę