

Ćwiczenie nr 5 - Klasyfikacji zbiorów wielowymiarowych danych z wykorzystaniem sieci neuronowych (badanie własności klasyfikatora typu bagging).

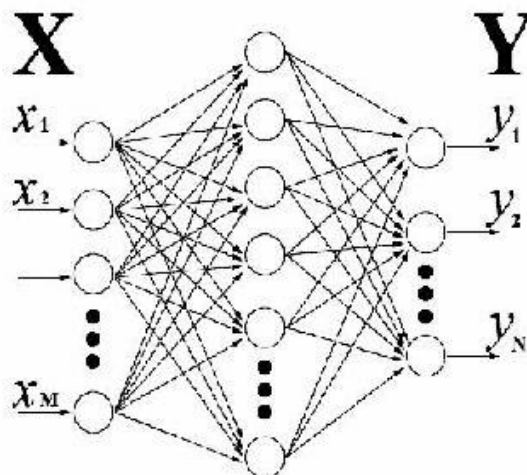
1. Cele ćwiczenia

- o Wybrać trzy zbiory danych z repozytorium UCI. Podzielić każdy zbiór (pełnowymiarowy) na zbiór do uczenia i testowania. Dokonać uczenia na zbiorze testowym przy pomocy dwóch sieci neuronowych. Jedna ze stosunkowo niewielką liczbą neuronów, druga z dużo większą.
- o Znaleźć jakość klasyfikacji dla każdego z danych i każdej sieci jako średnią z wyników klasyfikacji na zbiorach testowych (+ wariancja) dla co najmniej 5 różnych wyborów zbioru treningowego i testowego.
- o Na jak najprostszej dwuwarstwowej sieci neuronowej zbudować klasyfikator typu bagging (trenowanie sieci na różnych podzbiorach zbioru treningowego) i sprawdzić dla najtrudniejszego do klasyfikacji zbioru UCI czy otrzymany wynik klasyfikacji jest lepszy niż otrzymany przy pomocy skomplikowanej sieci neuronowej.

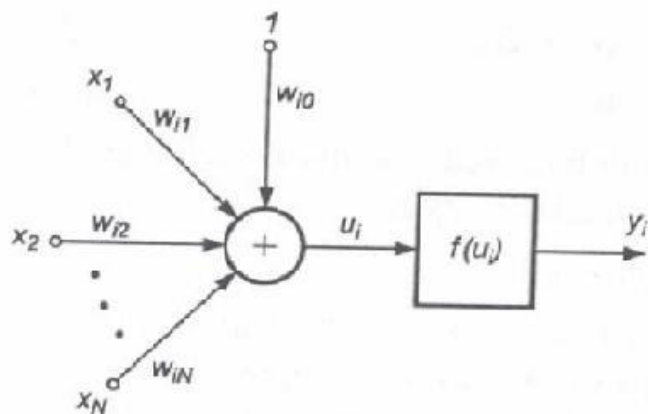
2. Wprowadzenie teoretyczne

Sztuczna sieć neuronowa (SSN) to ogólna nazwa struktur matematycznych i ich programowych lub sprzętowych modeli, realizujących obliczenia lub przetwarzanie sygnałów poprzez rzędy elementów, zwanych neuronami, wykonujących pewną podstawową operację na swoim wejściu. Oryginalną inspiracją takiej struktury była budowa naturalnych układów nerwowych, w szczególności mózgu. Czasem mianem sztuczne sieci neuronowe określa się interdyscyplinarną dziedzinę wiedzy zajmującą się konstrukcją, trenowaniem i badaniem możliwości tego rodzaju sieci.

Cechą wspólną wszystkich sieci neuronowych jest to, że na ich strukturę składają się neurony połączone ze sobą synapsami. Z synapsami związane są wagi (patrz Rys. 2, np. w_{il}), czyli wartości liczbowe, których interpretacja zależy od modelu oraz fakt, iż przetwarzają wektor wejściowy X na wektor wyjściowy Y (Rys. 1).



Rys. 1 Przykładowy schemat Sieci Neuronowej (Multi Layer Perceptron)



Rys. 2 Model Neuronu (w - wagi synaptyczne, u_i - suma, $f(u_i)$ - funkcja aktywacji)

Sieci jednokierunkowe (np. Rys. 1)

Sieci jednokierunkowe to sieci neuronowe, w których nie występuje sprzężenie zwrotne, czyli pojedynczy wzorzec lub sygnał przechodzi przez każdy neuron dokładnie raz w swoim cyklu. Najprostszą siecią neuronową jest pojedynczy perceptron progowy, opracowany przez McCullocha i Pittsa w roku 1943. W bardziej zaawansowanych rozwiązaniach stosuje się funkcje przejścia. Najpopularniejszą klasę funkcji stosowanych w sieciach neuronowych stanowią funkcje sigmoidalne, np. tangens hiperboliczny. Sieć zbudowana z neuronów wyposażonych w nieliniową funkcję przejścia ma zdolność nieliniowej separacji wzorców wejściowych. Jest więc uniwersalnym klasyfikatorem. Do uczenia perceptronów wielowarstwowych stosuje się algorytmy spadku gradientowego, między innymi algorytm propagacji wstecznej.

Sieci rekurencyjne

Mianem sieci rekurencyjnej określa się sieć, w której połączenia między neuronami stanowią graf z cyklami. Wśród różnorodności modeli rekurencyjnych sztucznych sieci neuronowych wyróżnić można:

- sieć Hopfielda - układ gęsto połączonych ze sobą neuronów (każdy z każdym, ale bez połączeń zwrotnych) realizującą dynamikę gwarantującą zbieżność do preferowanych wzorców
- maszyna Boltzmanna - opracowana przez Hintona i T. Sejnowskiego stochastyczna modyfikacja sieci Hopfielda; modyfikacja ta pozwoliła na uczenie neuronów ukrytych i likwidację wzorców pasożytniczych, kosztem zwiększenia czasu symulacji.

Sieci Hopfielda i maszyny Boltzmanna stosuje się jako pamięci adresowane kontekstowo, a także do rozwiązywania problemów minimalizacji (np. problemu komiwojażera).

Samoorganizujące się mapy

Samoorganizujące się mapy (Self Organizing Maps, SOM), zwane też sieciami Kohonena to sieci neuronów, z którymi są stowarzyszone współrzędne na prostej, płaszczyźnie lub w dowolnej n-wymiarowej przestrzeni. Uczenie tego rodzaju sieci polega na zmianach współrzędnych neuronów, tak aby dążyły one do wzorca zgodnego ze strukturą analizowanych danych. Sieci zatem "rozpinają się" wokół zbiorów danych, dopasowując do nich swoją strukturę. Sieci te stosowane są do klasyfikacji wzorców, np. głosek mowy ciągłej, tekstu, muzyki. Do najciekawszych zastosowań należy rozpinanie siatki wokół komputerowego modelu skanowanego obiektu.

Sztuczne sieci neuronowe znajdują zastosowanie w rozpoznawaniu i klasyfikacji wzorców (przydzielaniu wzorcom kategorii), predykcji szeregów czasowych, analizie danych statystycznych, odsumianiu i kompresji obrazu i dźwięku oraz w zagadnieniach sterowania i automatyzacji.

3. Rezultaty

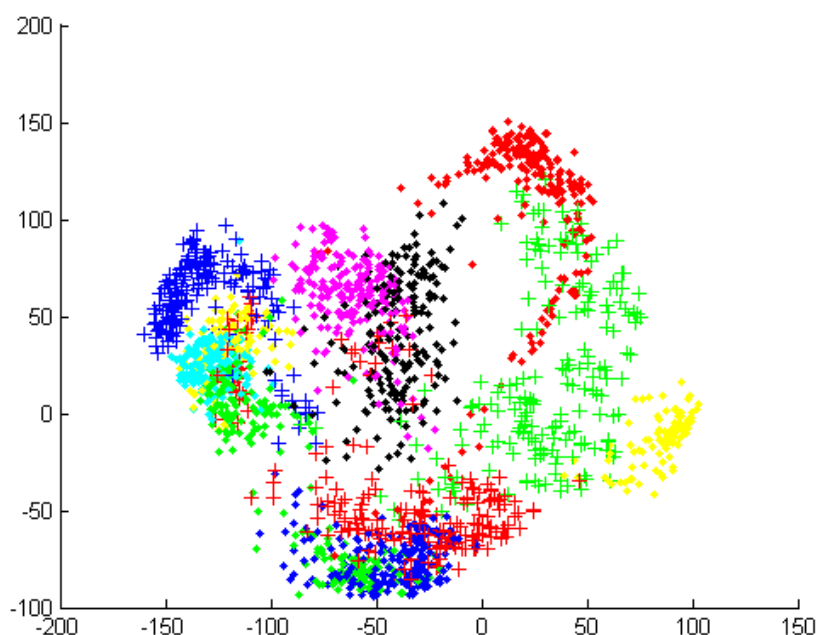
- Wybór danych do analizy:

1. Zbiór - pendigits (<http://www.ics.uci.edu/~mllearn/databases/pendigits/>). Zawiera on dane

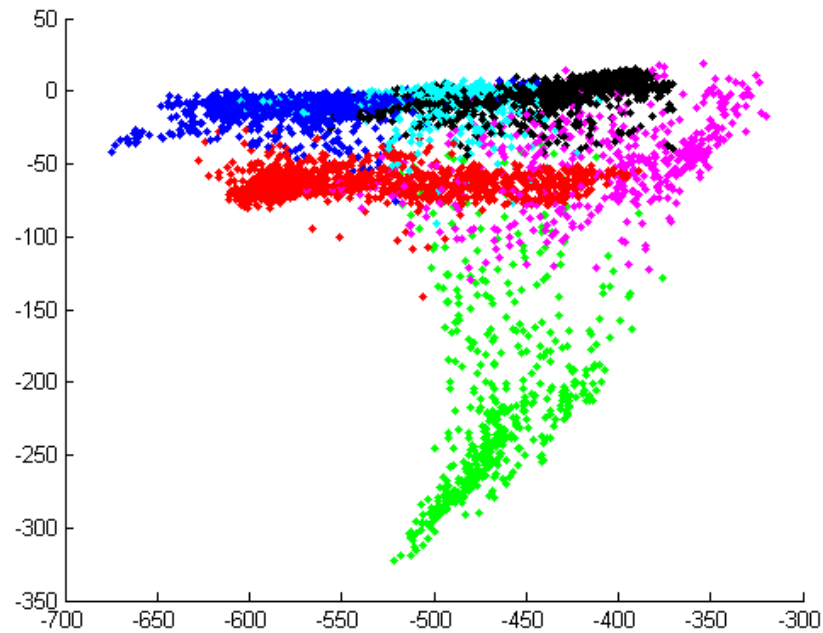
reprezentujące cyfry zapisywane na tablicie graficznym. Każda cyfra zapisana jest jako 8 elementowy wektor współrzędnych (x,y) pomierzonych przez urządzenie w równych odstępach czasu - próbkowanie (właściwie współrzędne te są interpolowane, ze względu na zmienną długość wektorów współrzędnych uzyskiwanych z tabletu, zależną od pisańca i rodzaju cyfry). Baza zawiera zbiór danych uczących (pendigits.tes) i zbiór danych testowych (pendigits.tra). Problemem do rozwiązania jest rozpoznanie cyfr ze zbioru testowego na podstawie danych uczących. Zbiór testowy został utworzony przez inne osoby niż zbiór uczący.

2. Zbiór - satimage (<http://mlearn.ics.uci.edu/databases/statlog/satimage/>). Zbiór zawiera 36 atrybutów będących wartościami pikseli obrazu satelitarnego. Celem klasyfikacji w tym przypadku jest określenie rodzaju fotografowanego podłoża (red soil, cotton crop, grey soil, damp grey soil, soil with vegetation stubble, mixture class (all types present), very damp grey soil).
3. Zbiór - vehicle (<http://mlearn.ics.uci.edu/databases/statlog/vehicle>). Zbiór ten zawiera rekordy złożone z 18 atrybutów opisujących sylwetkę samochodu uzyskanych ze zdjęcia (np: współczynniki zwartości, kolistości). Celem klasyfikacji jest określenie marki samochodu (OPEL Manta, SAAB 9000, double decker bus, Cheverolet Van) na podstawie podanych cech.

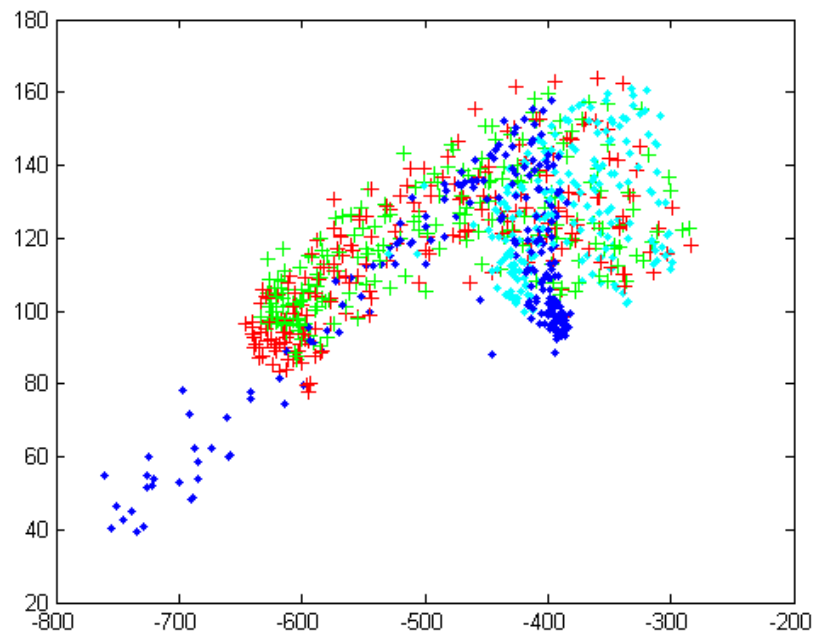
o Wizualizacja (PCA)



Rys. 3 Pendigits wizualizacja 2D



Rys. 4 Satimage wizualizacja 2D



Rys. 5 Vehicles wizualizacja 2D

o Symulacje

W poniższej tabeli zaprezentowano wyniki symulacji wykonywanych dla dwóch sieci neuronowych - małej (16 w warstwie wejściowej, 10 w ukrytej i 1 na wyjściu) oraz dużej (64 we, 40 hid, 1 wy). Dla każdego zbioru wykonano po 6 symulacji. W tabeli pokazano uzyskane rezultaty (wartości określają stosunek trafionych prób do wielkości zbioru testowego). Tabela prezentuje wartość maksymalną (MAX), średnią (AVE), i odchylenie standardowe (STD). Jak widzimy najslabiej wypadło klasyfikowanie samochodów, dlatego ten zbiór posłuży do następnego etapu ćwiczenia, czyli zastosowania klasyfikatora typu bagging.

Mała sieć [16 10 1]				Duża sieć [64 40 1]		
	MAX	AVE	STD	MAX	AVE	STD
pendigits	0.8394	0.7305	0.1157	0.7199	0.6695	0.517
satimage	0.7710	0.7267	0.0338	0.6955	0.6747	0.0329
vehicles	0.7060	0.6613	0.0396	0.6520	0.6340	0.0144

Zasada działania metody "Bagging":

- metoda bootstrap generujemy m różnych zbiorów (próbek) wybierając k pozycji z k-elementowego zbioru danych treningowych z zastępowaniem część pozycji może pojawić się więcej niż raz, innych może nie być w ogóle,
- trenujemy każdy model na innej takiej próbce zbioru treningowego (imitacja uczenia na różnych zbiorach treningowych)
- wyjście z systemu otrzymujemy przez uśrednienie lub głosowanie.

VEHICLES			
	MAX	AVE	STD
Mała sieć	0.7060	0.6613	0.0396
Duża sieć	0.6520	0.6340	0.0144
bagging	0.7500	0.7133	0.0197

4. Wnioski

- Zbyt mała ilość neuronów powoduje zmniejsza skuteczność klasyfikacji (sieć niedouczona),
- Sieć ze zbyt dużą ilością neuronów może charakteryzować się słabymi możliwościami generalizacji (sieć przeuczona), powodującymi zwiększenie błędu klasyfikacji,
- Dla danego zbioru danych powinna się znaleźć optymalna struktura Sieci, dająca najlepsze możliwości generalizacji,
- Zastosowanie metody "bagging" powoduje zmniejszenie błędu klasyfikacji, Metoda "bagging" wykorzystuje wiele sieci z małą ilością neuronów (klasyfikatorów prostych).
- Metoda "bagging" dzięki losowaniu prób uczących, poprawia ich "reprezentatywność", zwiększając w ten sposób skuteczność uczenia.

Kody: [Cwiczenie](#), [Bagging](#),