

Sprawozdanie

Wykorzystanie logiki rozmytej do predykcji przebiegu czasowego

Laboratorium komputerowe WFIlS AGH

Amadeusz Filipek

1. Wstęp

Celem ćwiczenia jest wykonanie rozmytego systemu wnioskującego do rozwiązania problemu predykcyjnego.

Logika rozmyta jest logiką wielo-wartościową, która operuje na całym zakresie wartości logicznych $[0, 1]$ w odróżnieniu od klasycznej logiki binarnej, która operuje na dwóch wartościach 0 oraz 1. Logika rozmyta bazuje na teorii zbiorów rozmytych. Każdemu parametrowi przypisuje się funkcje przynależności, które określają w jakim stopniu dane stwierdzenie jest prawdziwe np. dla parametru temperatury funkcje przynależności mogą opisywać w jakim stopniu prawdziwe jest określenie „zimno” lub „ciepło”. Logika rozmyta znajduje zastosowanie między innymi w elektronicznych systemach sterowania a także w układach sztucznej inteligencji.

Rozmyty system wnioskujący stanowi układ logiczny wykorzystujący logikę rozmytą. Składa się z określonej ilości danych wejściowych oraz jednego wyjścia. Zarówno dla wejść jak i wyjścia zdefiniowane są funkcje przynależności. Ilość wprowadzonych funkcji przynależności określana jest mianem ziarnistości. Dodatkowo, każda z funkcji przynależności ma określony kształt zdefiniowany przez osobę projektującą układ. Najczęściej wykorzystywane są funkcje o postaci trójkątnej, trapezowej lub dzwonowej. System przetwarza dane wejściowe na wyjście za pomocą reguł. Reguły to warunki logiczne wprowadzone w postaci „if-then”. Nie ma ustalonych wytycznych co do wprowadzania reguł do systemu po za faktem, że reguły nie mogą być sprzeczne. Reguły wprowadza się na podstawie wiedzy ekspertów w danej dziedzinie. Wydajność układu w dużej mierze zależy od wprowadzonych reguł.

Wśród systemów wnioskujących można wyróżnić dwie metody implementacji układu: Mamdani oraz Takagi-Sugeno-Kang. Metody te różnią się sposobem projektowania bazy reguł. Układ Mamdani bazuje na wiedzy eksperta, zatem baza reguł konstruowana jest przez ludzi o dużej wiedzy na dany temat. Inaczej jest w przypadku układu Sugeno, gdzie baza reguł jest generowana samoczynnie przy wykorzystaniu technik adaptacyjnych. Wtedy przygotowuje się zestaw uczący dla układu i system w sposób samoczynny generuje bazę reguł.

2. Realizacja ćwiczenia

W ramach ćwiczenia wykonano rozmyty system wnioskujący metodą Sugeno przy wykorzystaniu środowiska obliczeniowego *Matlab*. System miał za zadanie aproksymować danego przebiegu wyjściowego na podstawie trzech sygnałów wejściowych. Funkcje przynależności sygnałów wejściowych przyjęto postaci zgeneralizowanego dzwonu *gbell* podczas gdy funkcje wyjściowe są postaci liniowej. Aproksymowany zestaw danych podzielono na zestaw uczący i testujący. Zbadano działanie układu w zależności od stosunku podziału danych, ziarnistości układu oraz od liczby epok procesu nauczania. Podział zestawu danych przebiegał w sposób losowy. System rozmyty został wygenerowany przy wykorzystaniu funkcji *genfis1*. Baza reguł została wygenerowana przy

wykorzystaniu funkcji *anfis*. Uzyskane wyniki błędu średniokwadratowego *mse* przedstawione są w poniższych tabelach.

Tabela 1. Błąd średniokwadratowy aproksymacji przy stosunku danych trenujących do uczących 50:50

epoch	Ziarnistość	3	5	7
10	mse_train	2.70E-04	6.30E-06	2.10E-07
	mse_test	3.00E-03	0.13	0.4
20	mse_train	2.70E-04	1.30E-05	1.15E-08
	mse_test	1.00E-02	0.94	0.12
40	mse_train	2.10E-04	4.90E-06	3.48E-08
	mse_test	3.20E-03	3.50E-02	0.3

Tabela 2. Błąd średniokwadratowy aproksymacji przy stosunku danych trenujących do uczących 70:30

epoch	Ziarnistość	3	5	7
10	mse_train	3.90E-04	4.65E-05	1.90E-05
	mse_test	1.80E-03	4.20E-02	7.20E-01
20	mse_train	4.40E-04	2.70E-05	7.60E-07
	mse_test	1.20E-03	2.60E-02	9.20E-01
40	mse_train	3.40E-04	1.39E-05	2.50E-07
	mse_test	1.30E-03	3.50E-02	7.40E-02

Tabela 3. Błąd średniokwadratowy aproksymacji przy stosunku danych trenujących do uczących 90:10

epoch	Ziarnistość	3	5	7
10	mse_train	4.60E-04	1.10E-04	4.60E-05
	mse_test	1.20E-03	4.50E-03	3.20E-02
20	mse_train	4.60E-04	7.40E-05	3.10E-06
	mse_test	7.10E-04	4.40E-02	2.70E-02
40	mse_train	3.50E-04	5.40E-05	3.90E-06
	mse_test	1.60E-03	2.10E-03	4.80E-02

Rezultaty przedstawione w poniższych tabelach wskazują na lepsze wyniki aproksymacji systemu dla mniejszej ilości wykorzystanych funkcji przynależności (ziarnistości). Wynik ten może być związany z faktem trzech sygnałów wejściowych, dla których generowana jest baza reguł. Przy ziarnistościach 5 oraz 7 baza ta staje się ogromna i układ staje się skomplikowany. Rezultaty nie wskazują na jasną zależność liczby epok procesu uczenia od jakości pracy układu. Stosunek danych trenujących do danych uczących wykazuje za to, że większa ilość wykorzystanych do trenowania danych prowadzi do lepszych wyników systemu. Kombinacja parametrów, która prowadzi do najlepszego rezultatu została zaznaczona w tabeli 3. na szaro i została osiągnięta przy 27 regułach. Dla tej kombinacji przetestowano inną funkcję generowania systemu rozmytego *genfis2*. Jednakże, funkcja

ta generuje układ rozmyty bez specyfikacji ziarnistości, ponieważ wykorzystuje procedurę substraktywnej klasteryzacji. Uzyskane wyniki wynoszą $mse_{train} = 2.4 \cdot 10^{-4}$ i $mse_{test} = 1 \cdot 10^{-3}$ przy 25 regułach układu i zadanym promieniu wpływu klastra 0.1 (*radii*) dla każdego z wejść i wyjść. Sprawdzono też kolejną funkcję generacji systemu rozmytego *genfis3*. Funkcja ta generuje system poprzez procedurę klasteryzacji „c-średnich”. Uzyskane wyniki wynoszą $mse_{train} = 2.3 \cdot 10^{-4}$ i $mse_{test} = 4 \cdot 10^{-3}$ przy 25 regułach układu i zadanej ilości 25 klastrów (liczba klastrów determinuje liczbę reguł układu). Uzyskane wyniki wszystkimi metodami są sobie bliskie. Funkcje *genfis2* oraz *genfis3* mają jednak zaletę, ponieważ nie trzeba z góry podawać ziarnistości oraz liczby reguł układu. Traci się jednak kontrolę nad tymi parametrami.

3. Podsumowanie

W ramach ćwiczenia wykonano rozmyty system wnioskujący i zastosowano go do ekstrapolacji przebiegu sygnału. Zbadano jakość pracy systemu w zależności od wykorzystanej liczby epok procesu uczenia, stosunku podziału zestawu danych na zbiory uczące i testujące oraz od liczby funkcji przynależności układu. Zaobserwowano, że wyniki pogarszały się wraz ze wzrostem ziarnistości układu co ma związek z komplikacją systemu ze względu na silnie rosnącą liczbę reguł układu. Zauważono także, że nie ma jasnej reguły co do liczby epok procesu nauczania, jednak wartości pośrednie 20 epok dawały najlepsze rezultaty. Z mniejszą liczbą epok może być związany fakt niedouczenia systemu podczas gdy zbyt duża liczba epok prowadzi do przeuczenia układu co jest widoczne gdy błąd dla danych uczących jest bardzo bliski zera natomiast błąd dla danych testujących rośnie. Dodatkowo zauważono, że większa liczba danych uczących przekazuje więcej informacji do systemu i pozwala go lepiej nauczyć co przekłada się na lepsze wyniki. W drugiej części ćwiczenia zbadano też alternatywne metody generowania systemu rozmytego. Uzyskano porównywalne wyniki, jednak funkcje oparte o metody klasteryzacji mają zaletę w postaci automatycznej generacji struktury systemu w tym doboru ziarnistości.