

Rozmyte Mapy Kognitywne i ich zastosowanie w klasyfikacji

Piotr Szwed

Katedra Informatyki Stosowanej AGH

06.10.2021

Plan prezentacji

- Historia i statystyki
- Definicja
- Uczenie
- Zastosowanie w klasyfikacji
- Propozycja klasyfikatora opartego na FCM

Historia

- 1976 – Mapy Kognitywne zostały zaproponowane przez politologa Roberta Axelroda
- 1986 – Rozmyte Mapy Kognitywne (Fuzzy Cognitive Maps - FCM) – Bart Kosko
- Ponad 2000 publikacji w WoS lub Scopus
- Kilka wariantów:
 - Fuzzy Cognitive Maps
 - Fuzzy Cognitive Networks
 - Rough Fuzzy Cognitive Networks
 - Intuitionistic Fuzzy Cognitive Maps

FCM - statystyki WOS

Publications

2,218

Total

From 1900 to 2021

Citing Articles

13,226 Analyze

Total

11,343 Analyze

Without self-citations

Times Cited

29,851

Total

13,361

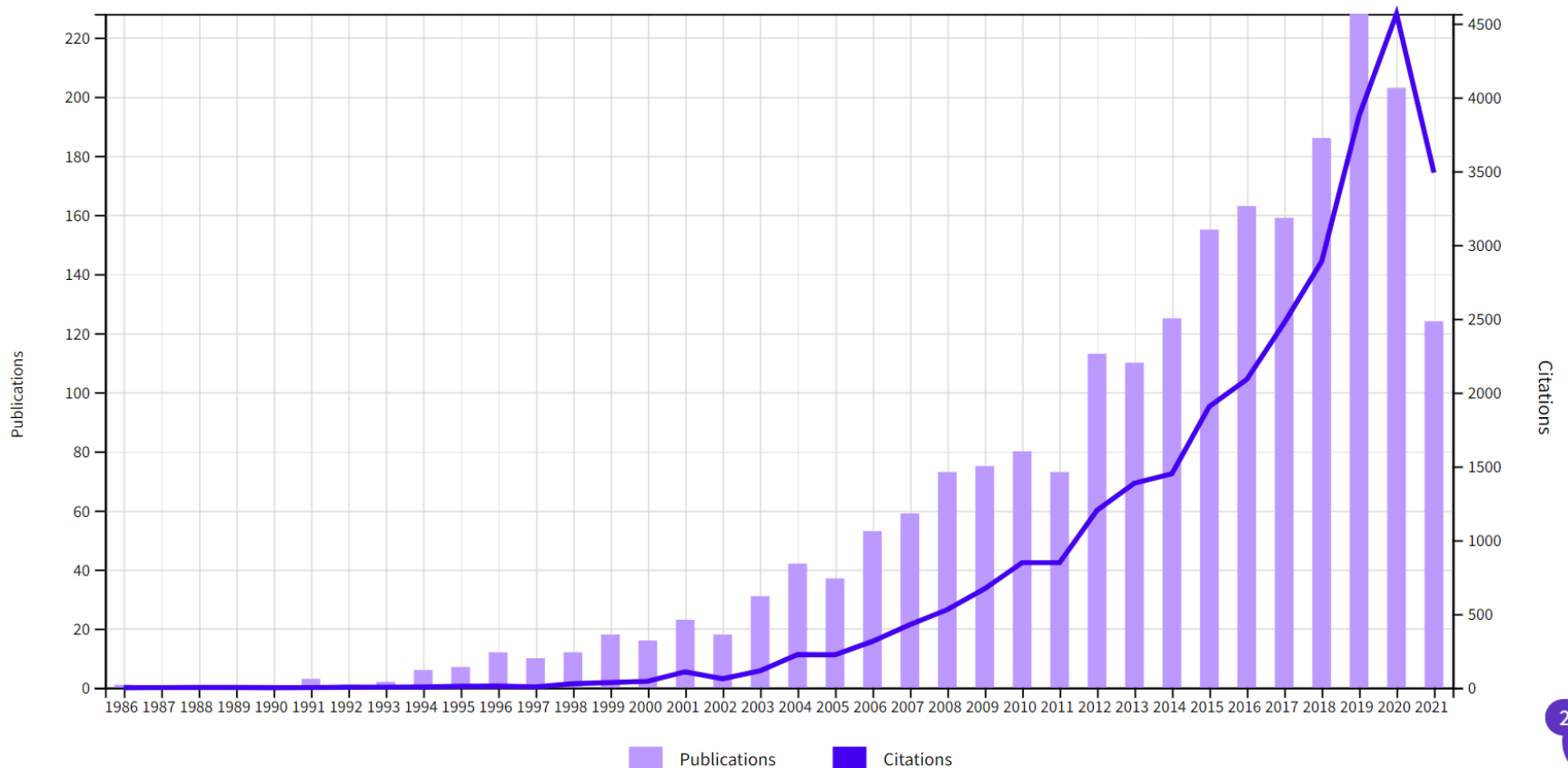
Without self-citations

13.46

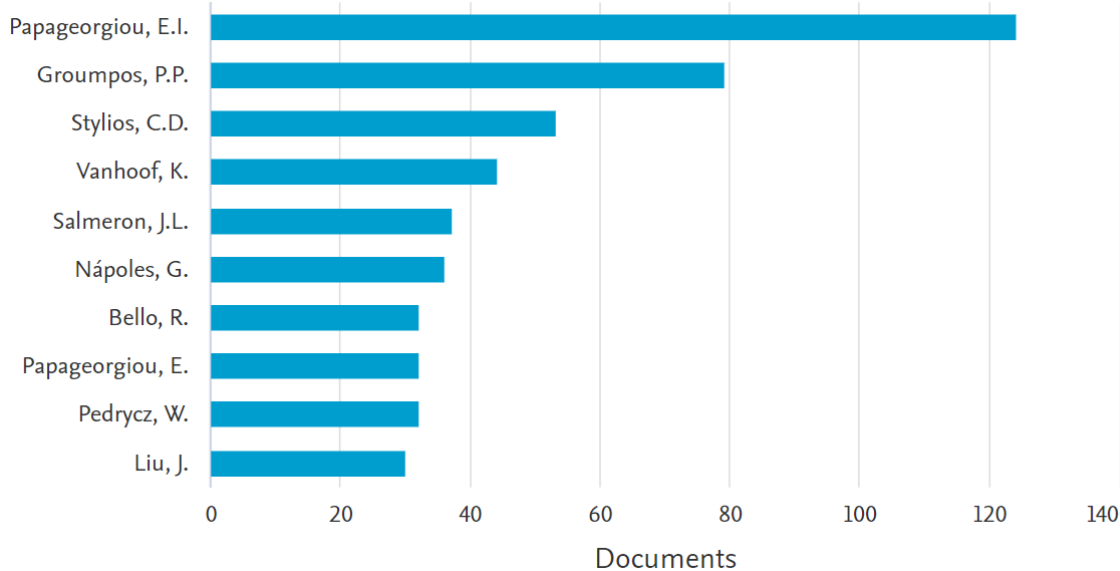
Average per item

71

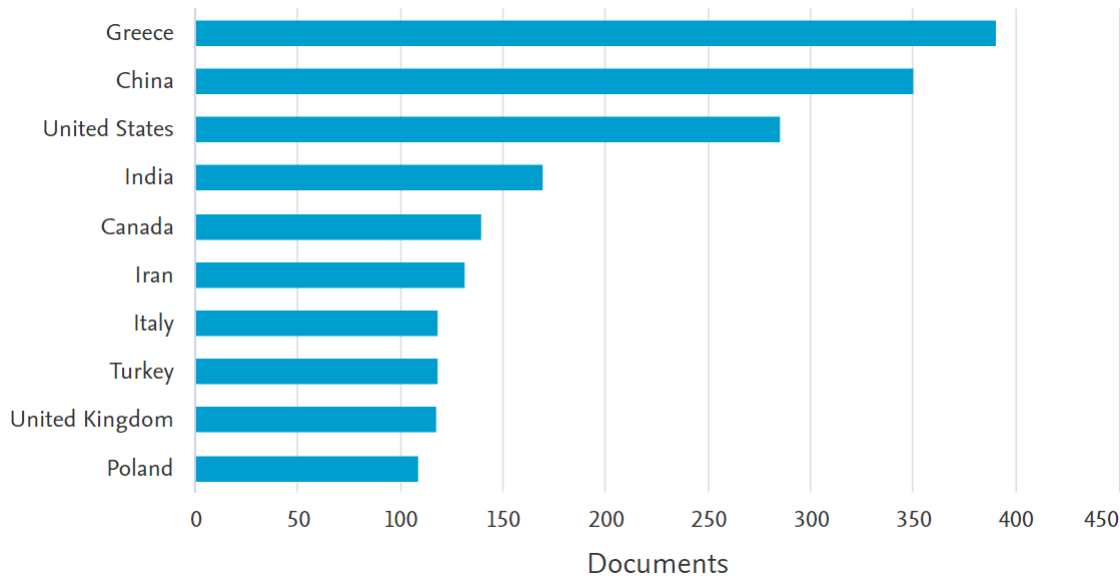
H-Index



FCM - statystyki Scopus

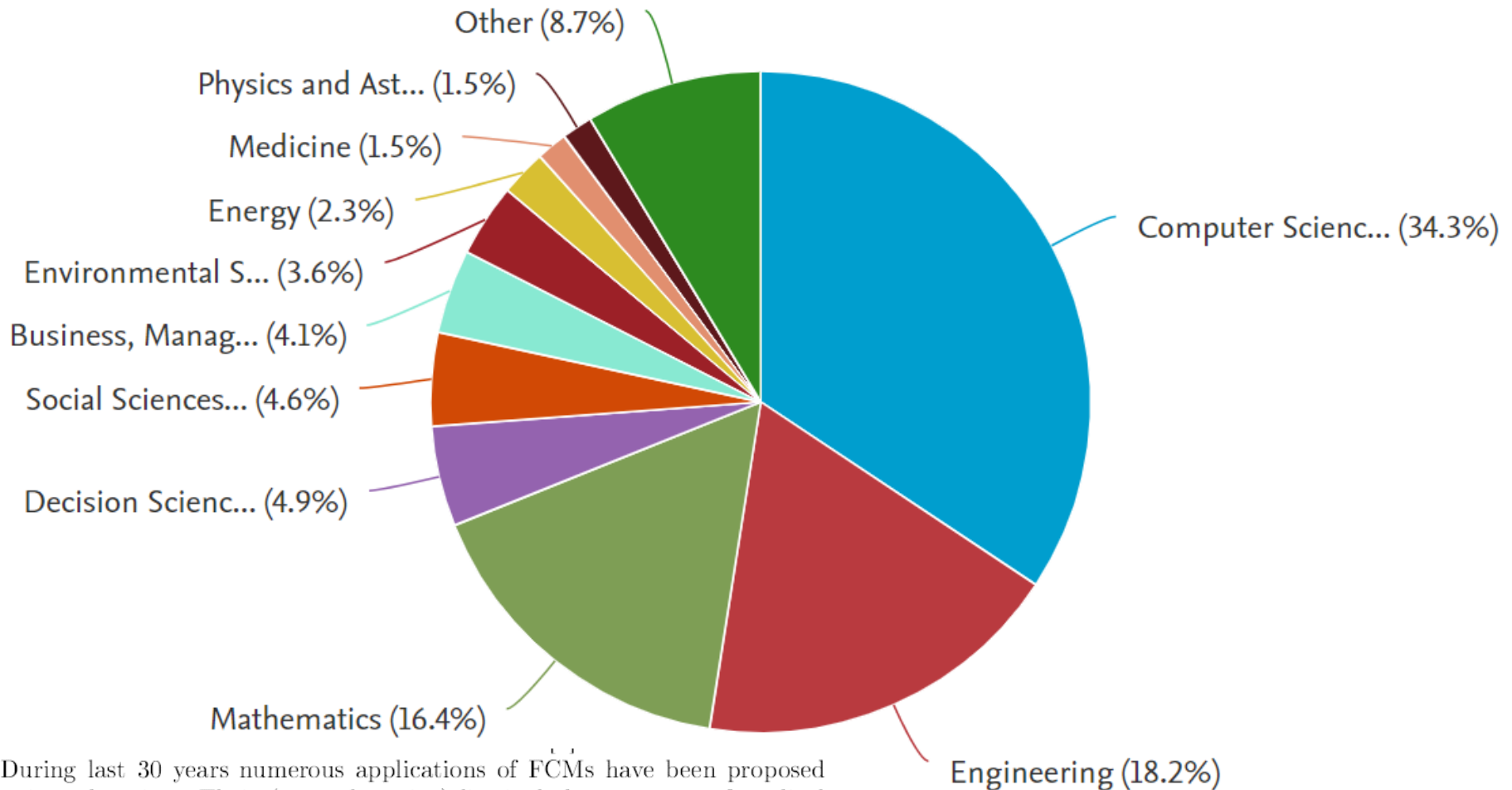


Autorzy



Kraje

FCM - statystyki Scopus 2

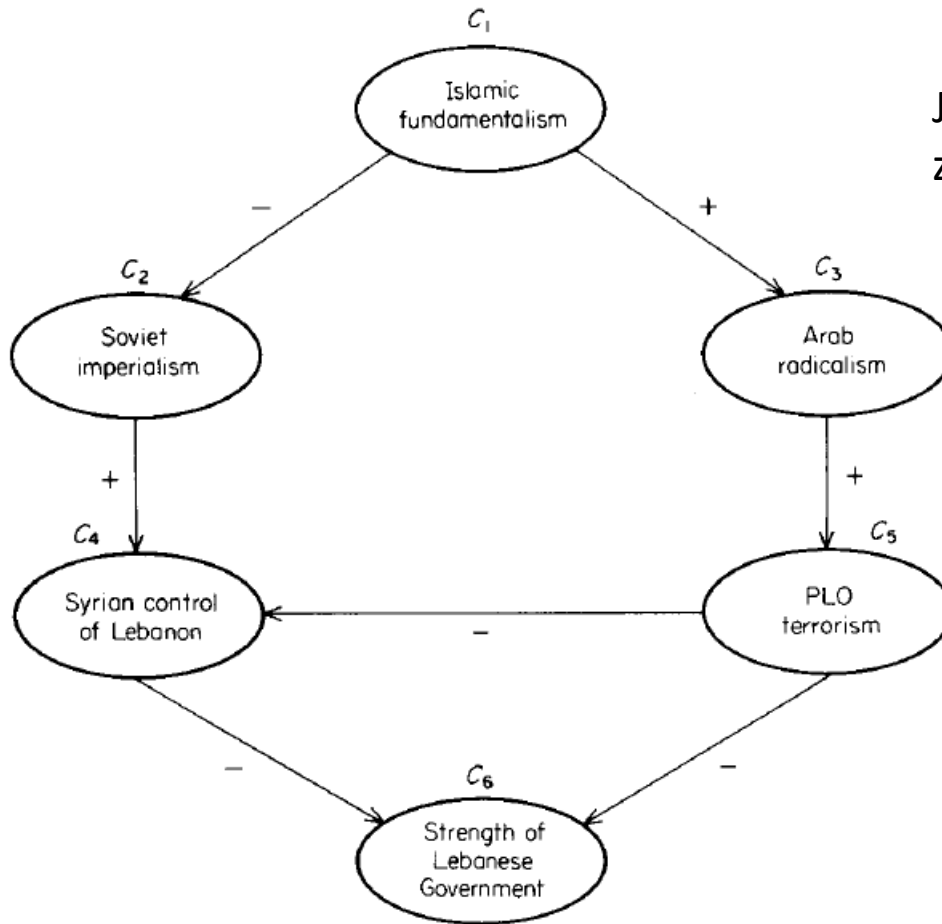


During last 30 years numerous applications of FCMs have been proposed in various domains. Their (not exhaustive) list includes: support of medical decisions [6], identifying gene regulatory network from gene expression data [7], recognition of emotions in speech [8], process control [9], ecosystem modeling [10], analysis of development of economic systems and introduction of new technologies [11], academic units development [12], prediction of time series [4], traffic prediction [13] project risk modeling [14], reliability engineering [15] and security risk assessment for IT systems [16, 17].

Rozmyte Mapy Kognitywne (FCMs)

- Stanowią reprezentację wiedzy w postaci zbioru pojęć/czynników (ang. concept) i relacji przyczynowych pomiędzy nimi
- FCM jest grafem skierowanym:
 - **Wierzchołki** symbolizują **pojęcia** $C = \{c_1, \dots, c_n\}$
 - **Krawędzie** odpowiadają **relacjom** (wpływowom)
 - **Wagi** (wartości liczbowe przypisane do krawędzi) opisują siły wpływów

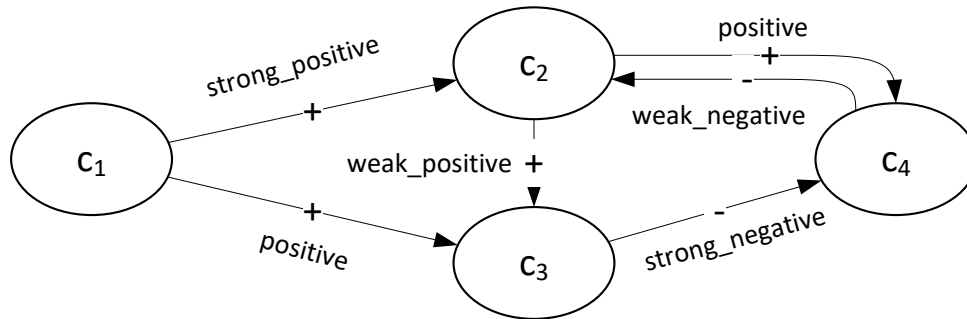
Przykład



Jako wagi często używane są wartości ze zbioru $\{-1, 1\}$ (oznaczone jako -/+)

Mapa kognitywna oparta na analizie Kissingera
„Starting out in the direction of Middle East peace” z roku 1982

FCM – macierz wpływu



Wartości lingwistyczne wag:

strong_negative,
negative,
medium_negative,
neutral,
medium_positive,
positive,
strong_positive

$$W = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -0.33 \\ 0.66 & 0.33 & 0 & 0 \\ 0 & 0.66 & -1 & 0 \end{bmatrix}$$

Macierz wpływu $W = [w_{ij}]$

Element w_{ij} jest wagą łuku łączącego pojęcia c_j oraz c_i (lub ma wartość 0, jeżeli brak jest połączenia).

FCM - zachowanie

- **Stan systemu** A jest n -wymiarowym wektorem ($n = |C|$) poziomów aktywacji pojęć.
- **Poziomy aktywacji** pojęć mapy kognitywnej są rozmyte: przybierają wartości z przedziału $[0,1]$ lub $[-1,1]$.
- **Przejawem zachowania** FCM jest ciąg stanów:

$$\alpha = A(0), A(1), \dots, A(t), \dots$$

gdzie

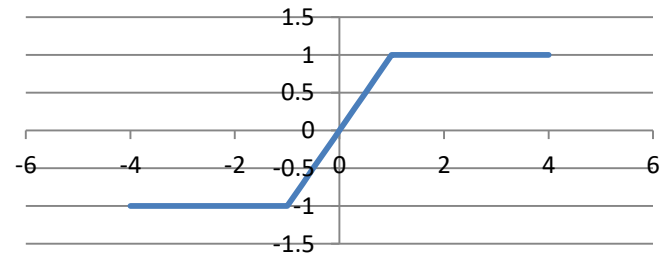
$$A_i(t + 1) = f_i \left(\sum_{j=1}^n w_{ij} A_j(t) \right)$$

f_i jest nazywaną funkcją aktywacji (ang. *activation function*, *squashing function*)

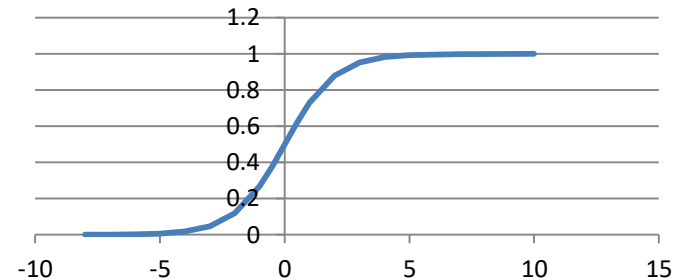
Funkcja aktywacji

- W wyniku pomnożenia wiersza W przez wektor $A(i)$ otrzymywane są wartości z zakresu $[-n, n]$.
- Funkcja aktywacji ma na celu sprowadzenie tych wartości do założonego przedziału $[-1, 1]$ lub $[0, 1]$.

$$f_{cut}(x) = \begin{cases} 1 & \text{dla } x > 1 \\ x & \text{dla } x \in [-1, 1] \\ -1 & \text{dla } x < -1 \end{cases}$$



$$f_{sigmoid}(x) = \frac{1}{1 + \exp(-\lambda x)}$$



Typy atraktorów

Wagi macierzy FCM i typ funkcji aktywacji określają zachowanie FCM. Trzy typy atraktorów :

- Stały (ang. fixed point)
- Cykliczny
- Chaotyczny

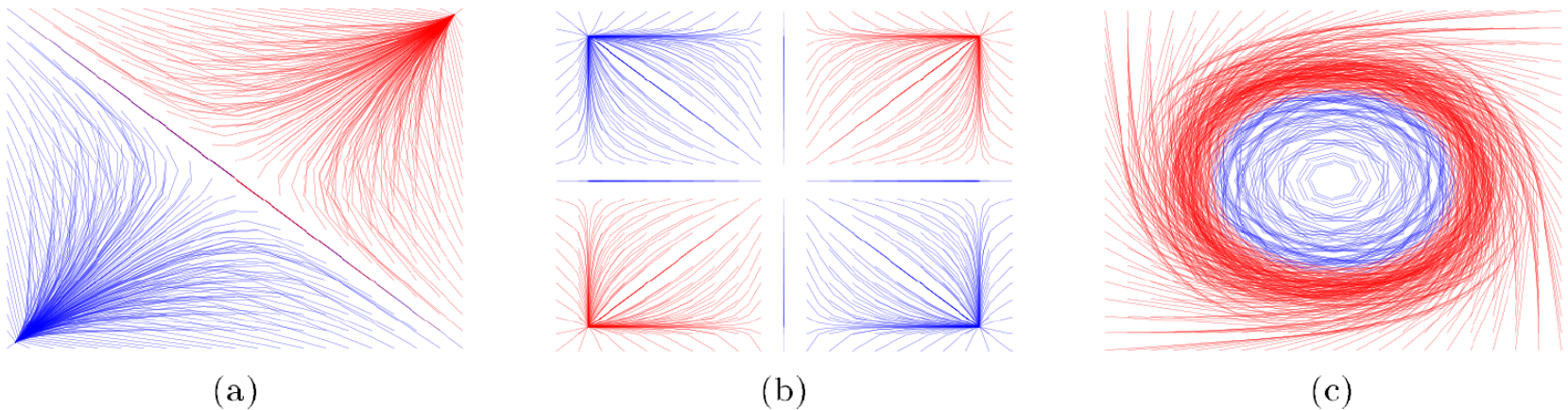
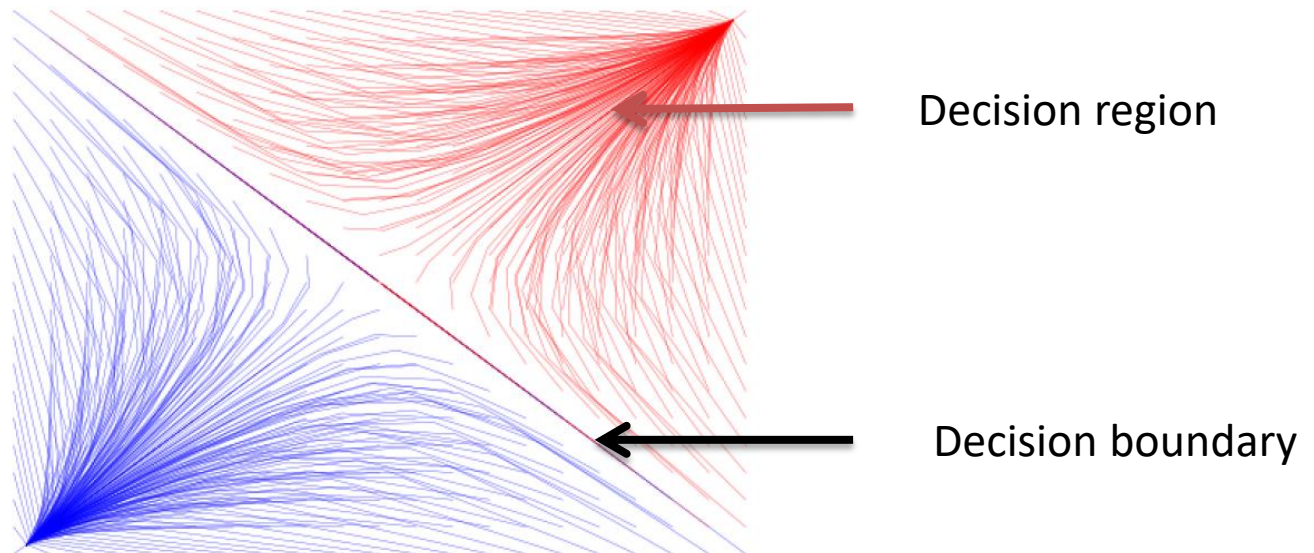


Figure 1: Examples of FCM attractors: (a) two fixed point (b) four fixed point (c) cyclic. Trajectories of 2D FCMs starting at evenly spaced initial points tend to concentrate approaching attractors. Separate attraction basins are marked with different colors.

Interpretacja trajektorii

- **Reprezentacja zachowania modelowanego systemu.**
 - Trajektoria FCM = ewolucja stanów w systemu w czasie.
 - **Zastosowania:** scenariusze rozwoju lub predykcja szeregów czasowych
- **Proces wnioskowania**
 - Poziomy aktywacji wybranych pojęć w punkcie stałym = rezultaty wnioskowania
 - **Zastosowania:** podejmowanie decyzji, sterowanie, klasyfikacja

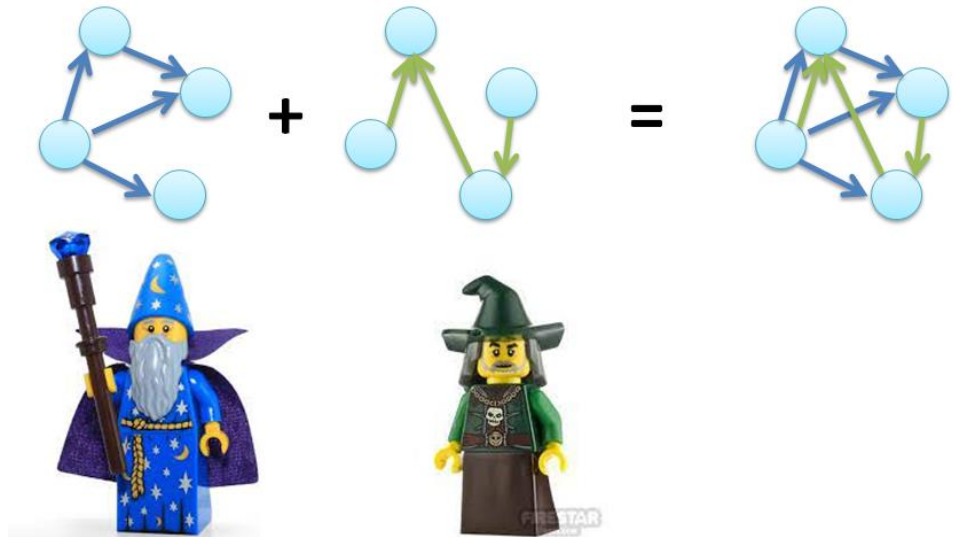


FCM i sieci neuronowe

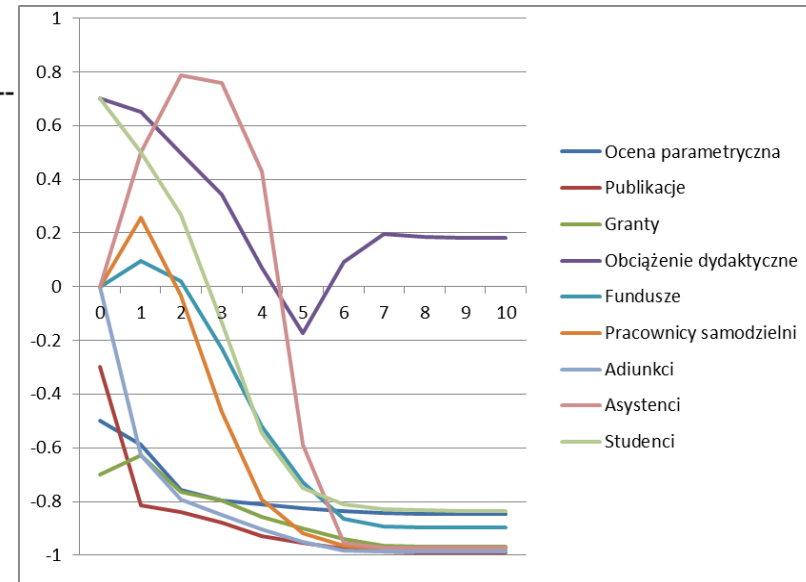
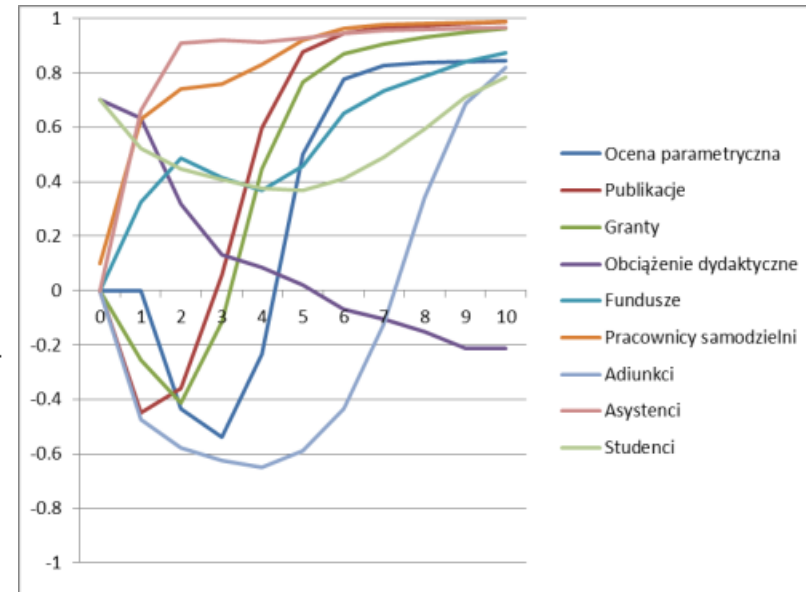
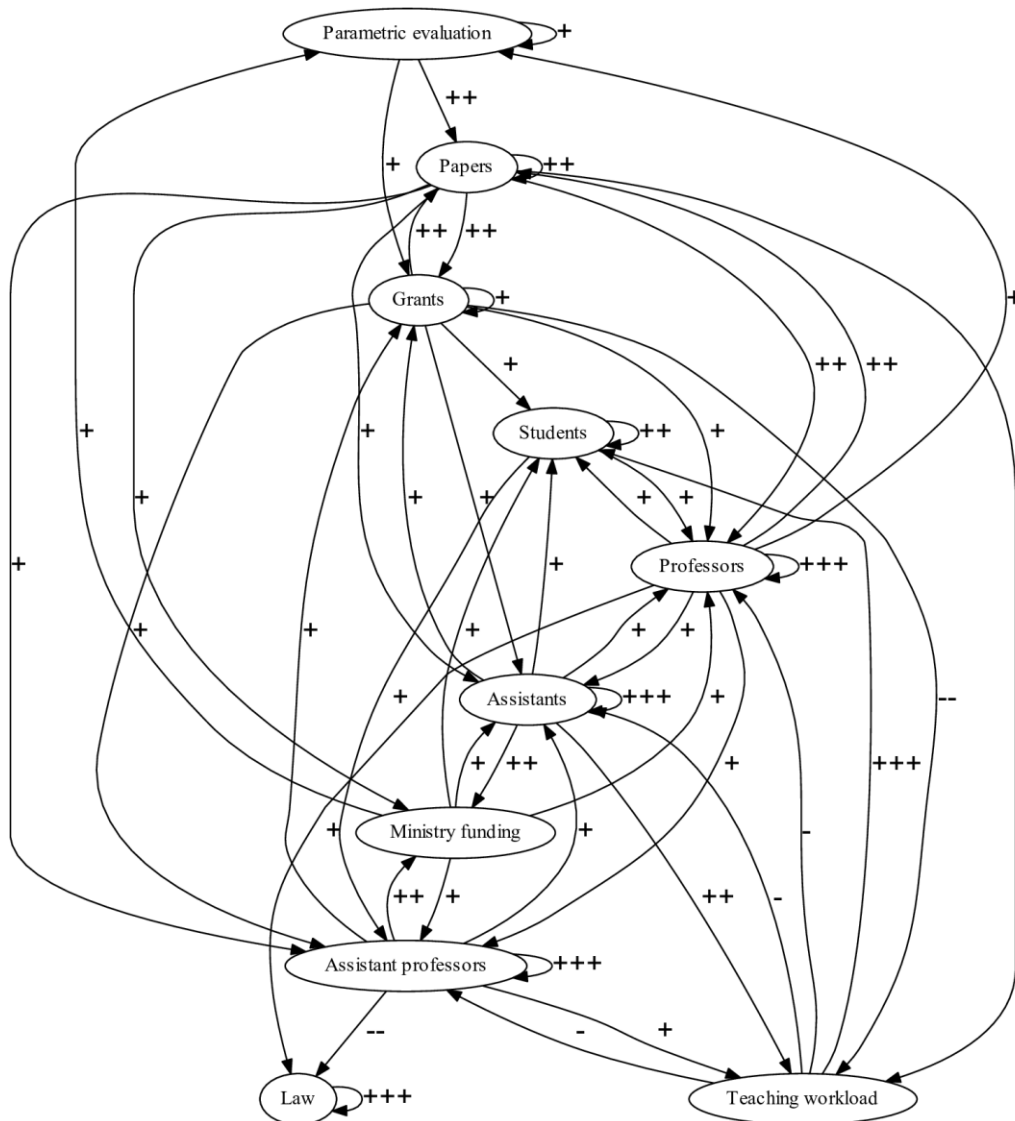
- Podobieństwa
 - Pojęcia odpowiadają neuronom
 - Liniowa operacja mnożenia przez wagi i nieliniowa funkcja aktywacji
 - Często wyróżniane są pojęcia wejściowe, wyjściowe
 - Mogą mieć architekturę feed-forward (jak MLP) lub rekurencyjną

- Różnice

- Są postrzegane jako modele semantyczne
- Mogą być tworzone przez ekspertów (głosowanie i agregacja)
- Są interpretowalne (ang. interpretable)



Przykład - FCM



Uczenie FCM

- Zazwyczaj ustalona jest architektura
- Celem jest znalezienie wag zapewniających oczekiwane zachowanie podczas predykcji lub wnioskowania
- Dwa typy metod uczenia:
 - **Hebbowskie** (ang. Hebbian)
„Synapsa łącząca dwa neurony ulega wzmocnieniu jeśli neuron pre- i post synaptyczny są jednocześnie aktywne. [...] Siłą napędową uczenia Hebbowskiego są korelacje aktywności neuronów pre- i post synaptycznych.”
[https://brain.fuw.edu.pl/edu/index.php/Uczenie_maszynowe_i_sztuczne_sieci_neuronowe]
 - **Sterowane błędami** (ang. error driven)
Używana jest funkcja kosztu wyrażająca różnice (błędy) pomiędzy oczekiwanymi i wyznaczonymi wyjściami FCM. Funkcja kosztu jest minimalizowana za pomocą wybranego algorytmu optymalizacji.

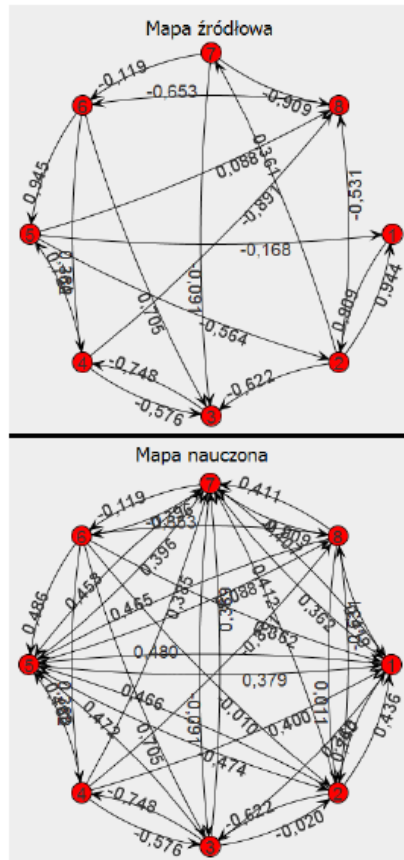
Hebbowskie metody uczenia

- Wejścia:
 - Struktura FCM
 - Początkowy zestaw wag określony przez ekspertów
 - Oczekiwane wartości pojęć wyjściowych
- Wagi są iteracyjnie uaktualniane na podstawie równoczesnych zmian poziomów aktywacji pojęć
- Struktura mapy i znaki wag pozostają niezmiennione – mapa jest interpretowalna i wyraża opinie ekspertów dotyczące ich wpływów
- Metody:
 - Differential Hebbian Learning
 - Active Hebbian Learning [E. Papageorgiou, C. D. Stylios, P. P. Groumpos, Active hebbian learning algorithm to train fuzzy cognitive maps, 2004]
 - Nonlinear Hebbian Learning [E. Papageorgiou, C. Stylios, P. Groumpos, Fuzzy cognitive map learning based on nonlinear Hebbian rule 2003]
 - Data Driven Nonlinear Hebbian learning [W. Stach, L. Kurgan, W. Pedrycz, Data-driven nonlinear hebbian learning method for fuzzy cognitive maps, 2008]

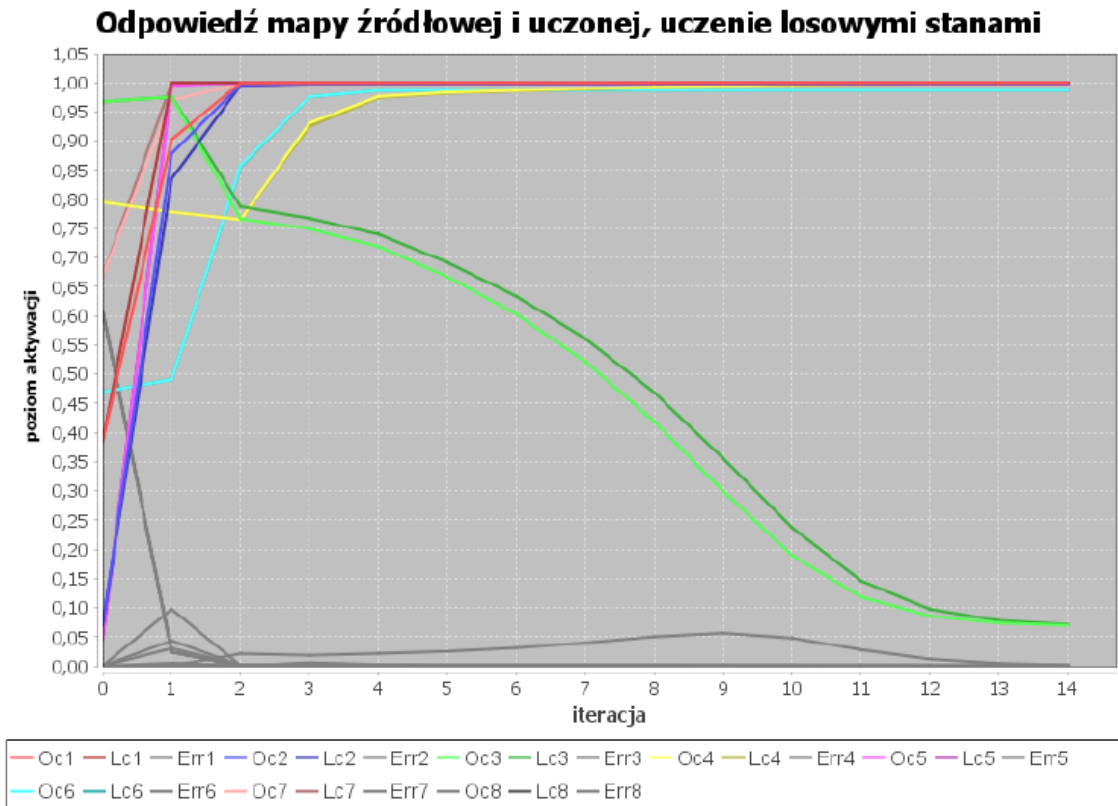
Metody uczenia sterowane błędami

- Zastosowanie: predykcja szeregów czasowych i klasyfikacja
- Typowe funkcje kosztów:
 - wartości bezwzględne błędów lub ich kwadraty
 - mogą obejmować kary za przekroczenie wartości
 - Składniki regularyzacyjne, np. Lasso L1
- Algorytmy:
 - Genetyczny GA
 - Real Coded Genetic Algorithm RCGA
[W. Stach, L. Kurgan, W. Pedrycz, Parallel learning of large fuzzy cognitive maps, 2007]
 - Particle Swarm Optimization
 - Differential Evolution
 - Ant Colony Optimization
 - Simulated Annealing
- Problem wielkości populacji. RCGA do 300 pojęć [Y. Chen, L. J. Mazlack, A. A. Minai, L. J. Lu, Inferring causal networks using fuzzy cognitive maps and evolutionary algorithms with application to gene regulatory network reconstruction, 2015]
- Ostatnie lata: **algorytmy gradientowe** – zwłaszcza dla szeregów czasowych

Problem – mapy stają się gęste ...i przestają być interpretowalne



(a) Mapy w formie grafu



(b) Odpowiedź mapy źródłowej i nauczonej.

Rys. 5.7. Porównanie mapy źródłowej i nauczonej metodą gradientu prostego.

[Elpidiusz Wszółek, Implementacja algorytmów uczenia rozmytych map kognitywnych na platformie OpenCL, praca mag. 2016]

Zbieżność funkcji sigmoidalnej

W zależności od zastosowań jest pożądane, aby mapa miała jeden lub kilka punktów stałych (ang fixed point).

- **Pojedynczy punkt**

- może być docelowym stanem w projektowanym **systemie sterowania** [T. L. Kottas, Y. S. Boutalis, M. A. Christodoulou, Fuzzy cognitive networks: Adaptive network estimation and control paradigms, 2010]
- Oczekuje się, że dla danego zestawu wag ustalonego przez ekspertów (reprezentacja wiedzy) wszystkie trajektorie będą zmierzały do pojedynczego punktu stałego wyrażającego **znaczenie czynników**.

- **Większa liczba punktów stałych**

- W zagadnieniach **klasyfikacji** punkty stałe mogą odpowiadać etykietom klas
- Dla danego rozmiaru mapy n liczba punktów stałych zależy od parametru λ sterującego nachyleniem $f_s(x) = \frac{1}{1+\exp(-\lambda x)}$
- Małe wartości λ gwarantują istnienie pojedynczego punktu stałego [C. J. Knight, D. J. Lloyd, A. S. Penn, Linear and sigmoidal fuzzy cognitive maps: an analysis of fixed points, 2014]

Zastosowanie w klasyfikacji

Przepis na klasyfikator FCM

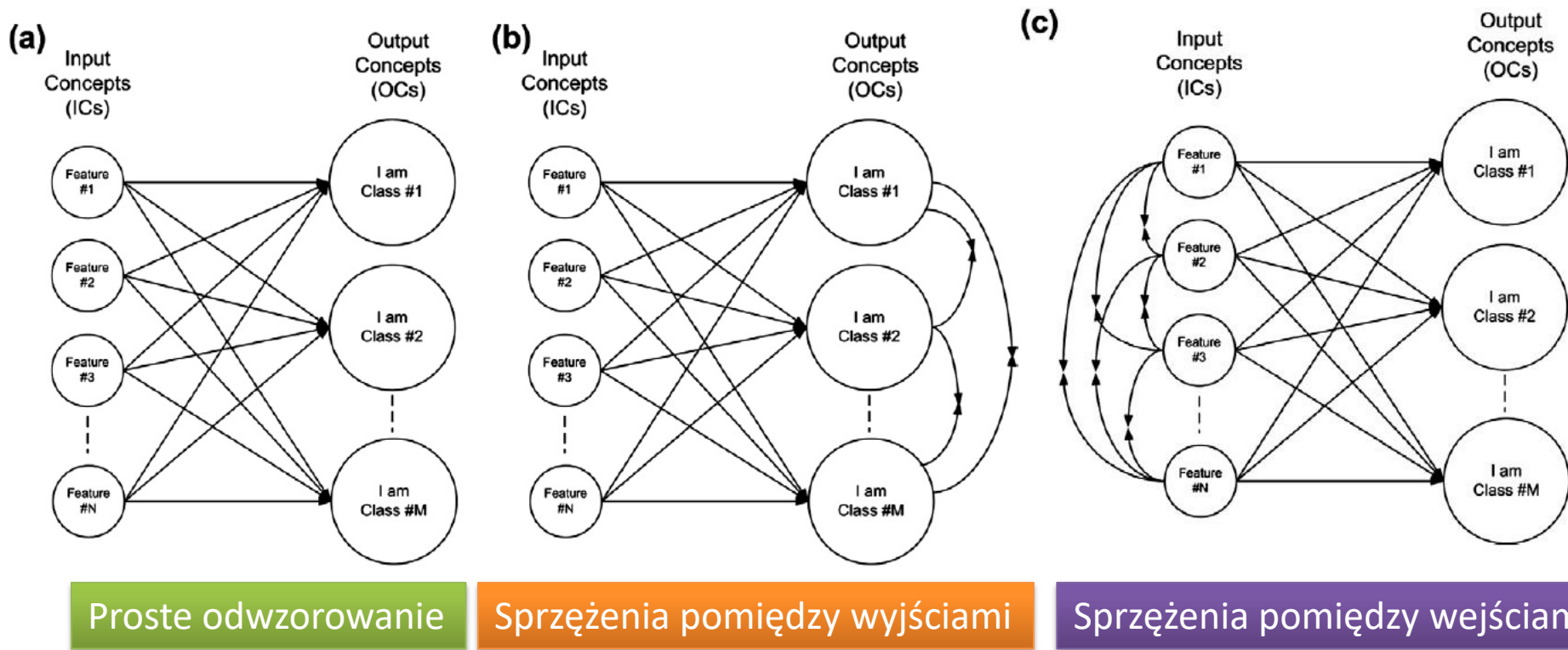
1. Ustal strukturę mapy.
Feedforward lub różne sprzężenia zwrotne.
2. W jaki sposób do modelu będą dostarczane dane?
Bezpośrednio lub po przetworzeniu.
3. W jaki sposób będzie odczytywany wynik klasyfikacji?
Jeden koncept czy kilka?
4. Wybór funkcji kosztu i algorytmu optymalizacji

Dwa typy:

- Niskopoziomowe (low-level) – pojęcia odpowiadają atrybutom obiektów.
- Wysokopoziomowe (high-level) – wprowadzane są abstrakcyjne pojęcia, reprezentujące „granule informacji”

Niskopoziomowe klasyfikatory FCM 1

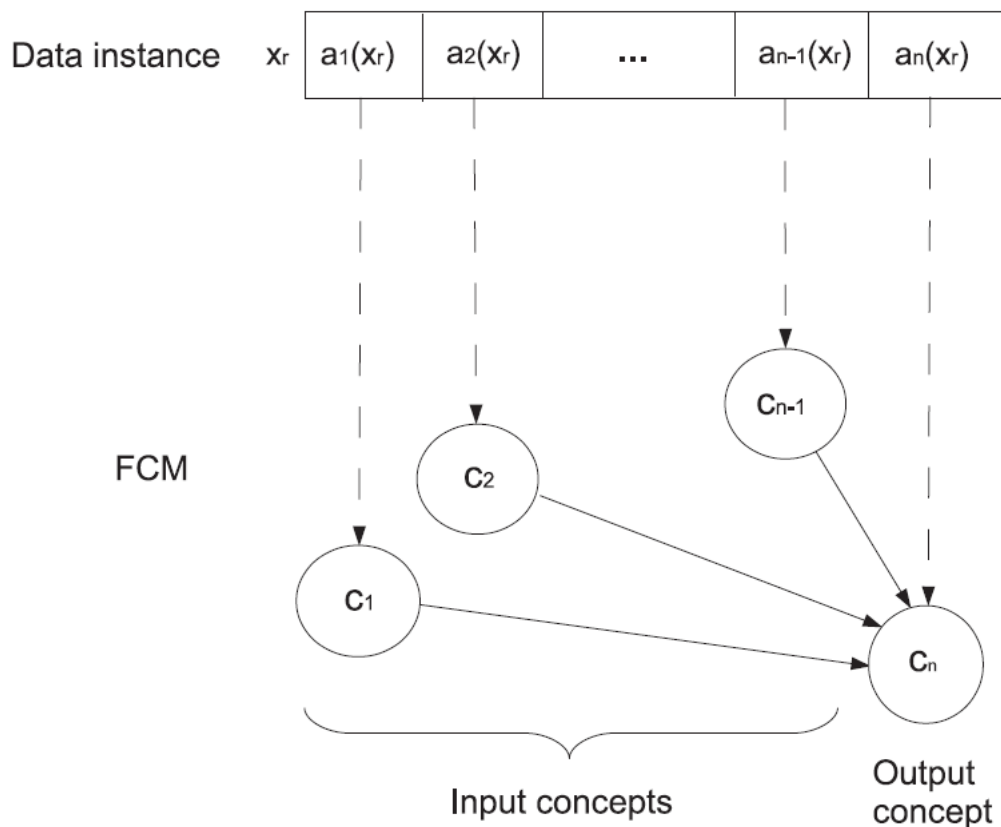
Liczba pojęć wyjściowych odpowiadająca liczbie klas



[G. A. Papakostas, D. E. Koulouriotis, A. S. Polydoros, V. D. Tourassis, Towards Hebbian learning of fuzzy cognitive maps in pattern classification problems, 2012]

Niskopoziomowe klasyfikatory FCM 2

Pojedyncze wyjście

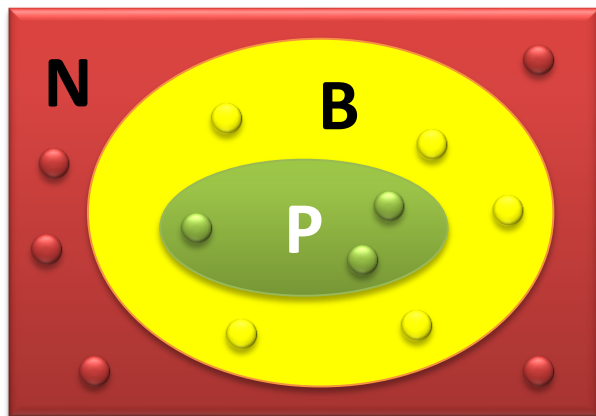


[W. Froelich, Towards improving the efficiency of the fuzzy cognitive map classifier, 2017]

Grupowanie wyjść
odpowiadających etykiетom
klas lub wyznaczanie
marginesów



Wysokopoziomowy: Rough Cognitive Networks



Zbiór przybliżony (ang. Rough Set RS).

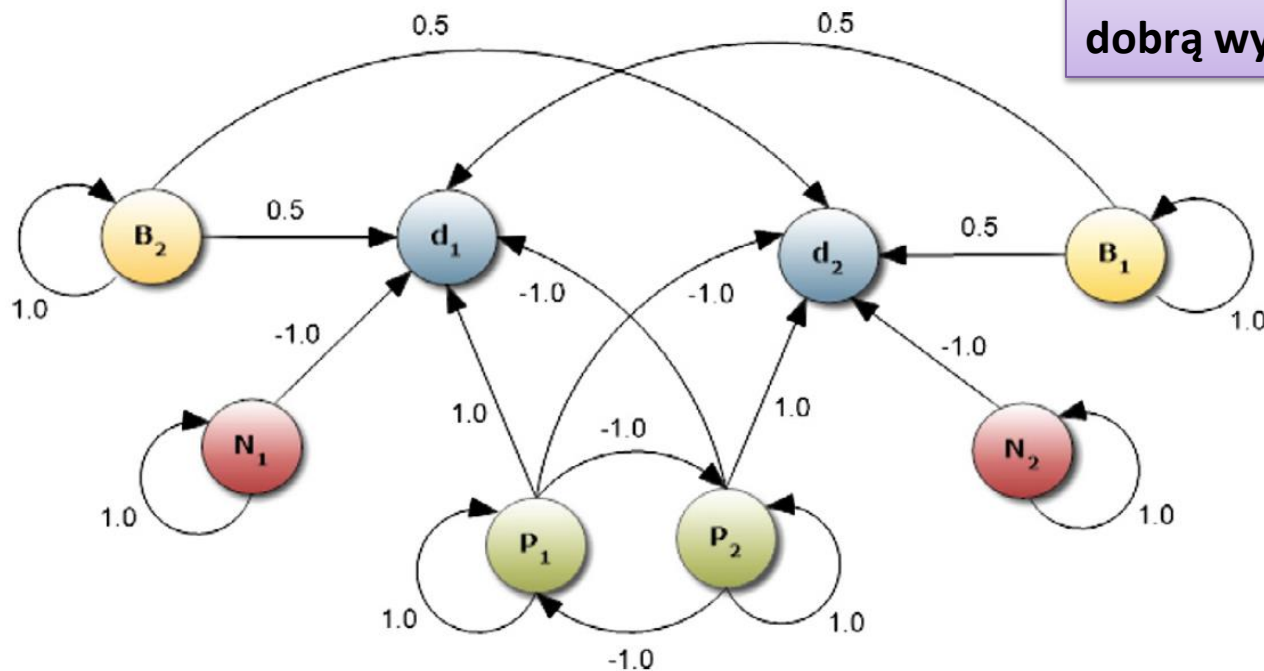
Dla danej relacji nierozróżnialności:

P (positive) – te obiekty na pewno należą do RS

B (boundary) – te obiekty być może należą do RS

N (negative) – te obiekty nie należą do RS

Jak dobrać relację
nierozróżnialności, aby uzyskać
dobrą wydajność?



[G. Nápoles, I. Grau, E. Papageorgiou, R. Bello, K. Vanhoof, Rough cognitive networks, 2016]

Zastosowanie FCM w klasyfikacji - podsumowanie

- Kilkanaście publikacji
- Dąży się do osiągnięcia punktu stałego (wiele iteracji FCM)
- Niskopoziomowe
 - Uczenie za pomocą algorytmu genetycznego lub metod Hebbowskich
 - Najbardziej wydajne dla prostych architektur (w praktyce równoważnych Regresji Logistycznej)
 - Liczba atrybutów jest ograniczeniem dla wydajności
 - Przetestowane na kilkunastu zbiorach danych z repozytorium UCI
- Wysokopoziomowe
 - RCN – zastosowano metaheurystykę Harmony Search
 - Architektura mapy zależy wyłącznie od liczby klas ($4 \times$ liczba klas)
 - RCN przetestowano na kilkudziesięciu zbiorach danych z UCI

Propozycja klasyfikatora

Opublikowana w

[P.Szwed, Classification and Feature Transformation with Fuzzy Cognitive Maps, ASOC, 2021]

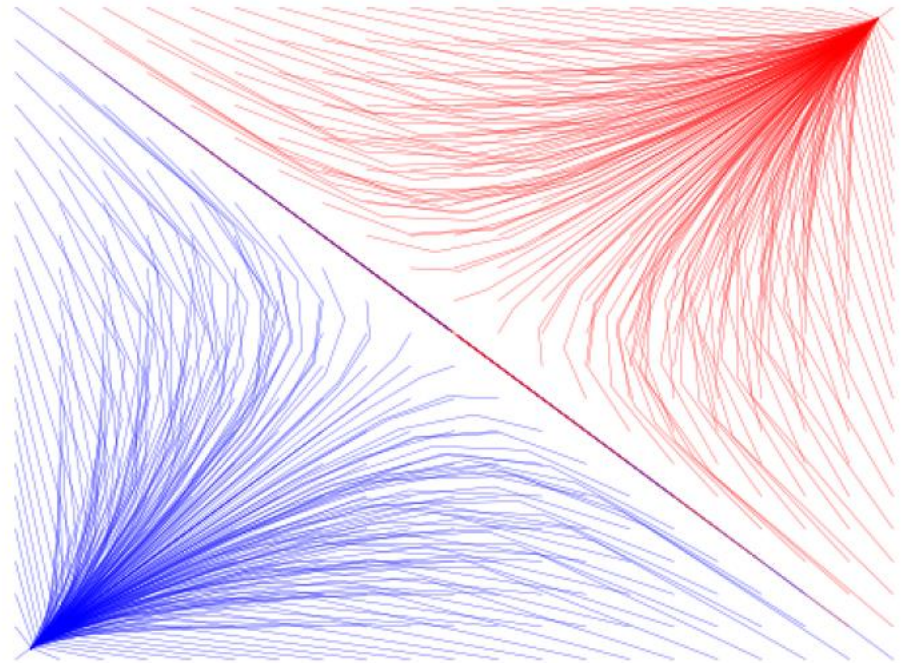
Projekt klasyfikatora

Klasyfikator niskopoziomowy

liczba pojęć wejściowych = liczba cech

Głębokość – parametr d

- Klasyfikator nie opiera się na zbieżności do punktu stałego.
- Konstruowane są trajektorie zawierające $d + 1$ elementów:
 $A(0), \dots, A(d - 1), A(d)$



Projekt klasyfikatora

Zakres cech i funkcja aktywacji

- Cechy są skalowane do przedziału $[0,1]$
- Nieco zmodyfikowana sigmoidalna funkcja aktywacji:

$$f(x) = \frac{1}{1 + e^{-\lambda(x-0.5)}}$$

Równanie stanu i wyraz wolny (**bias**)

$$A(t + 1) = f(\mathbf{W} \cdot A(t) + \mathbf{b})$$

Uczenie

- Zastosowano metodę gradientową
- Tensorflow + gradient tape (wyznaczanie gradientów metodą symboliczną)

Projekt klasyfikatora

Architektura z pętlami wstecznymi, także od pojęć wyjściowych

- FCMB – pojedyncze pojęcie wyjściowe z funkcją kosztu logloss
- FCMMC – liczba wyjść = liczbie klas, skalowanie softmax + crossentropy

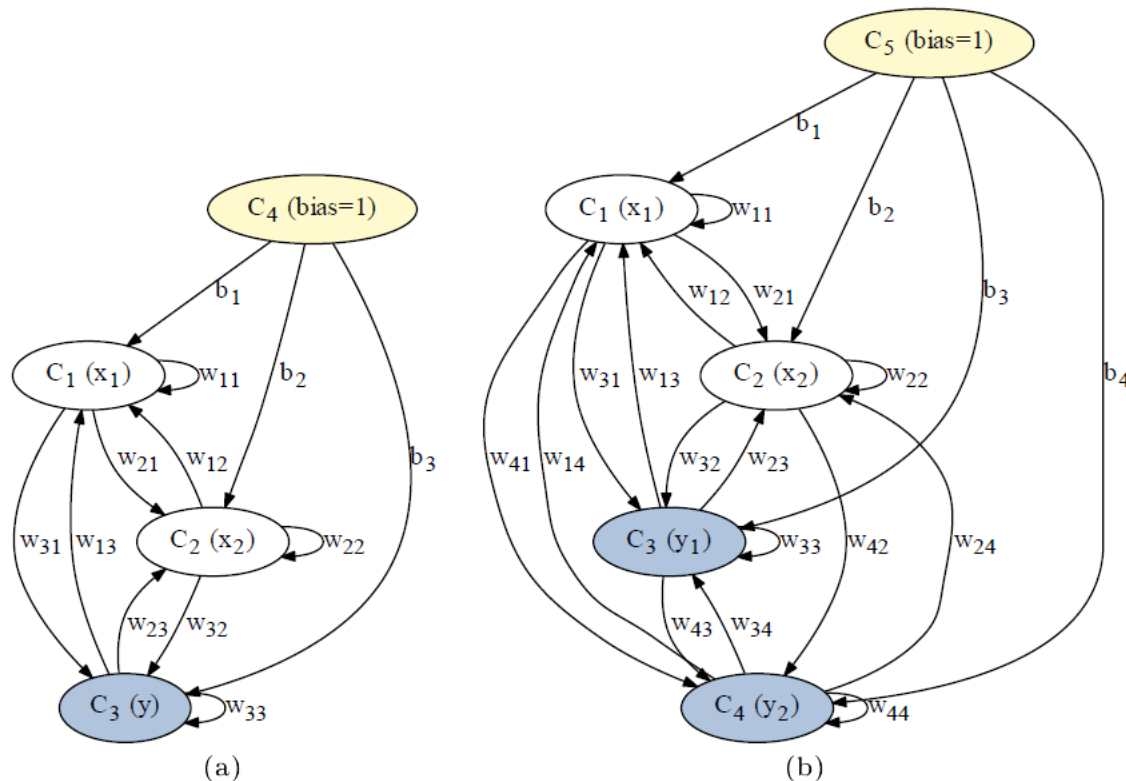


Figure 3: Two FCM architectures for 2D binary classification problems (a) FCMB – single output concept (b) FCMMC – two output concepts, each for class labels

Przykład 2D

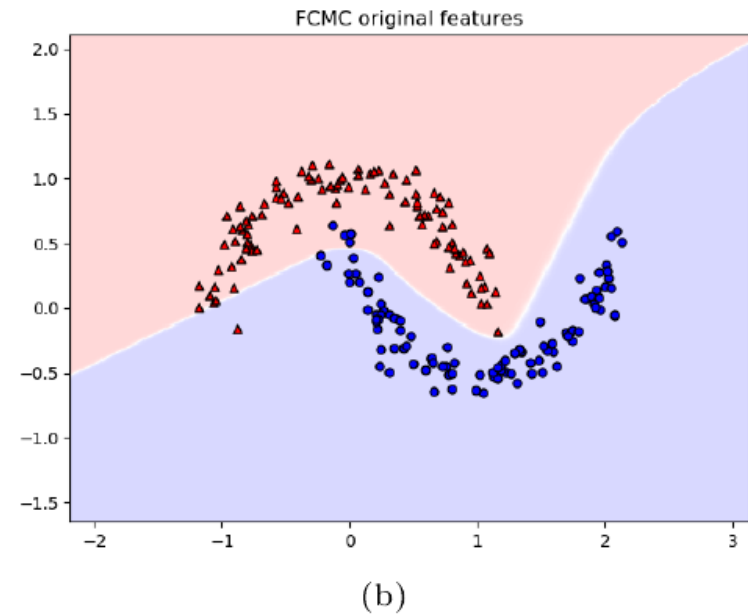
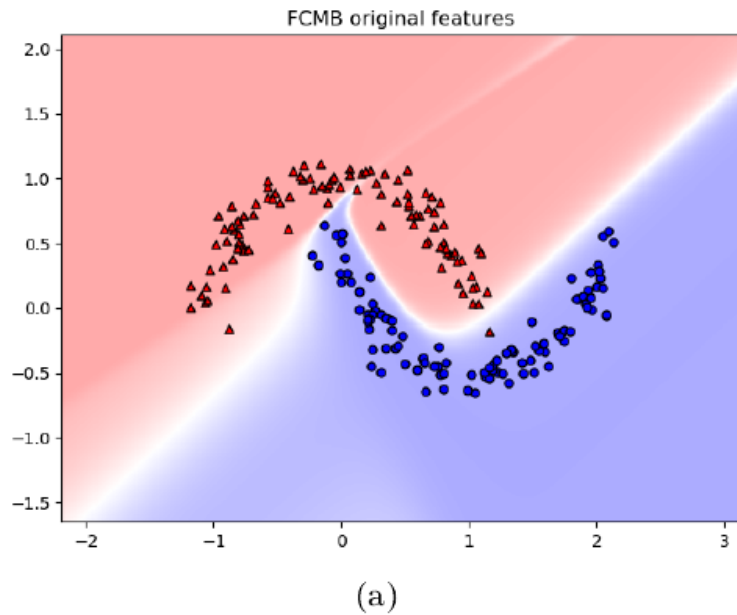
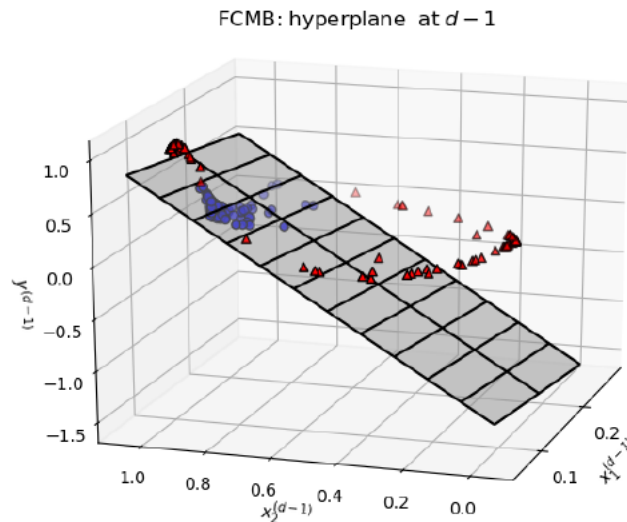


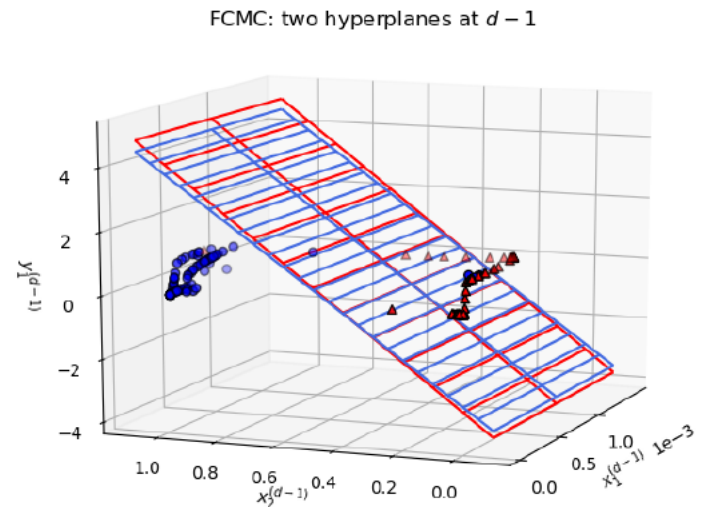
Figure 4: Example data set and class boundaries (a) FCMB classifier (b) FCMC classifier

Przykład 2D

**Ostatnia iteracja jest równoważna Regresji Logistycznej.
Jak więc wyglądają obserwacje po $d-1$ iteracjach FCM?**



(a)



(b)

Figure 4: 3D projection of transformed feature space at $d-1$ (a) FCMB classifier - one hyperplane separating data; blue points are placed below the hyperplane and red points above (b) FCMC classifier - two hyperplanes, each for a class label. Coordinates of hyperplanes were calculated for $y_*^{(d-1)} = 0.5$

Przykład 2D

Jak wygląda przetransformowana przestrzeń cech po $d-1$ iteracjach

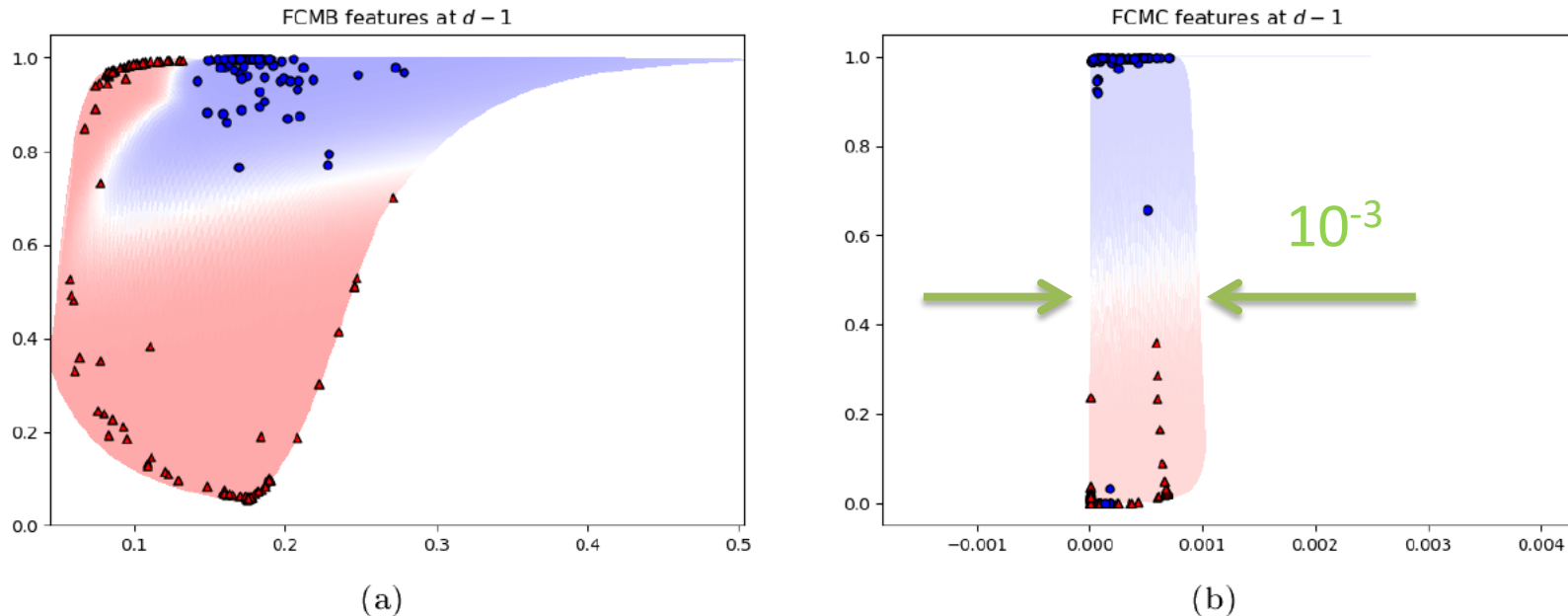


Figure 5: Transformed feature space at $d-1$ and class boundaries (a) FCMB classifier (b) FCMC classifier

Hipoteza: klasyfikator przekształca przestrzeń cech w taki sposób, że ostatni krok (równoważny Regresji Logistycznej) jest łatwiejszy...

Eksperymenty

- Przetestowano klasyfikator na 21 zbiorach danych (głównie z UCI)
- Zastosowano 5-krotną walidację krzyżową
- Porównano z 11 konfiguracjami klasyfikatorów z biblioteki scikit-learn
- Zbierano dane po **d-1** iteracjach (przetransformowana przestrzeń cech) aby:
 - Sprawdzić, czy są lepiej zgrupowane (metryki Davies-Bouldina, Silhouette i Calińskiego-Harabasa)
 - Określić wydajność ciągów przetwarzania (ang. pipeline) transformacja FCM + klasyfikator

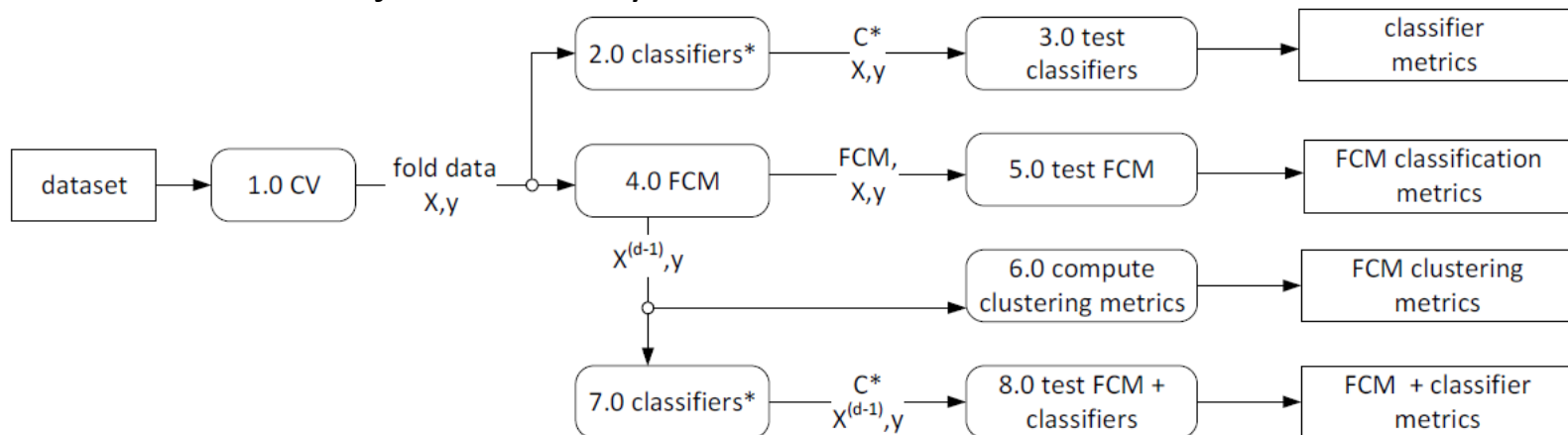
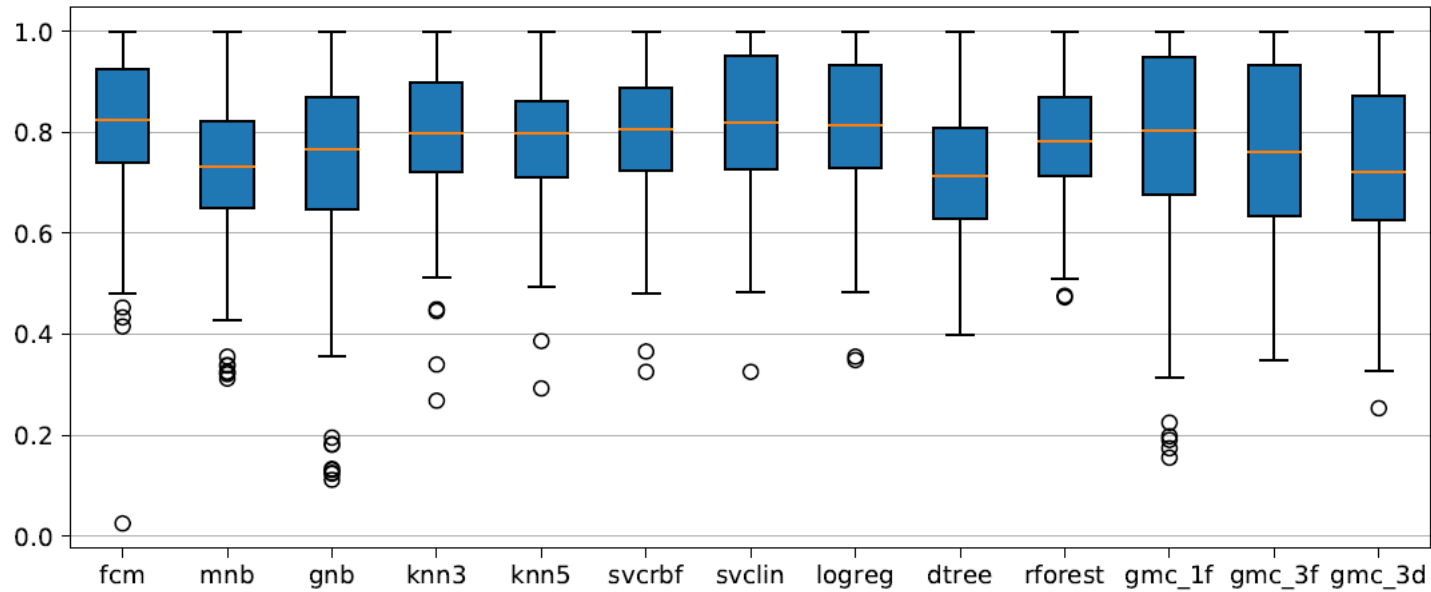


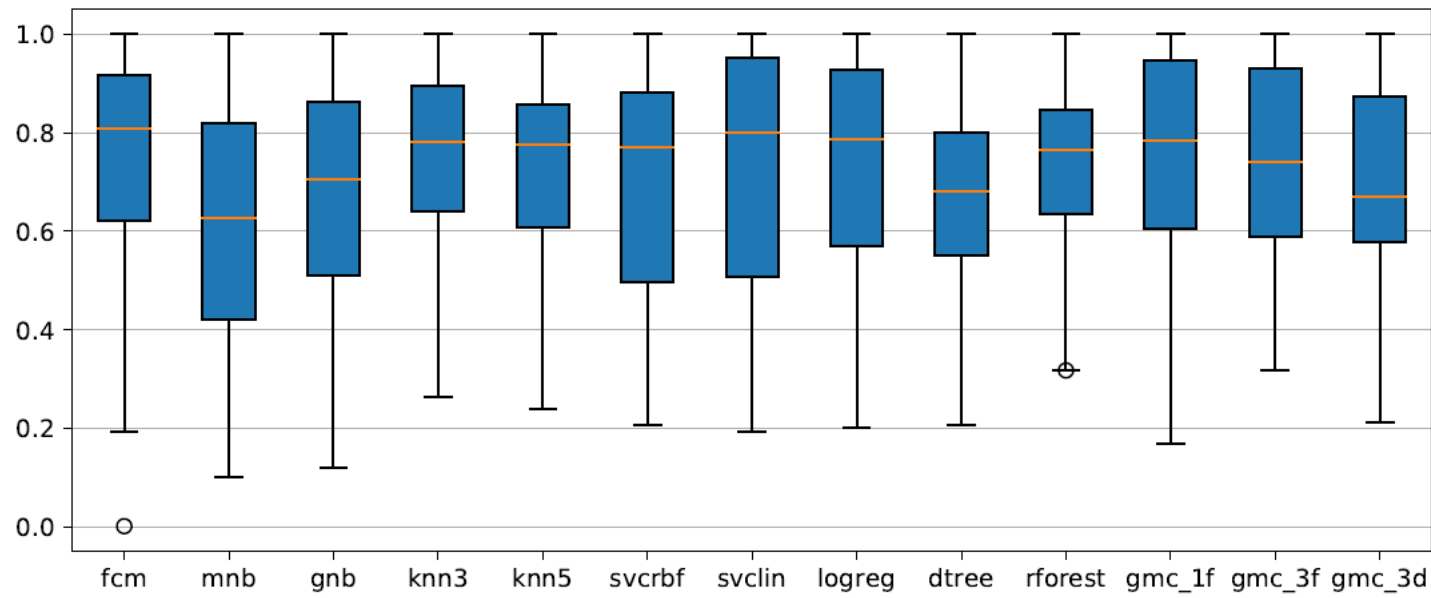
Figure 7: Design of experiments

Wyniki

Accuracy



F1 measure



Wyniki: testy Friedmana + Nemenyi

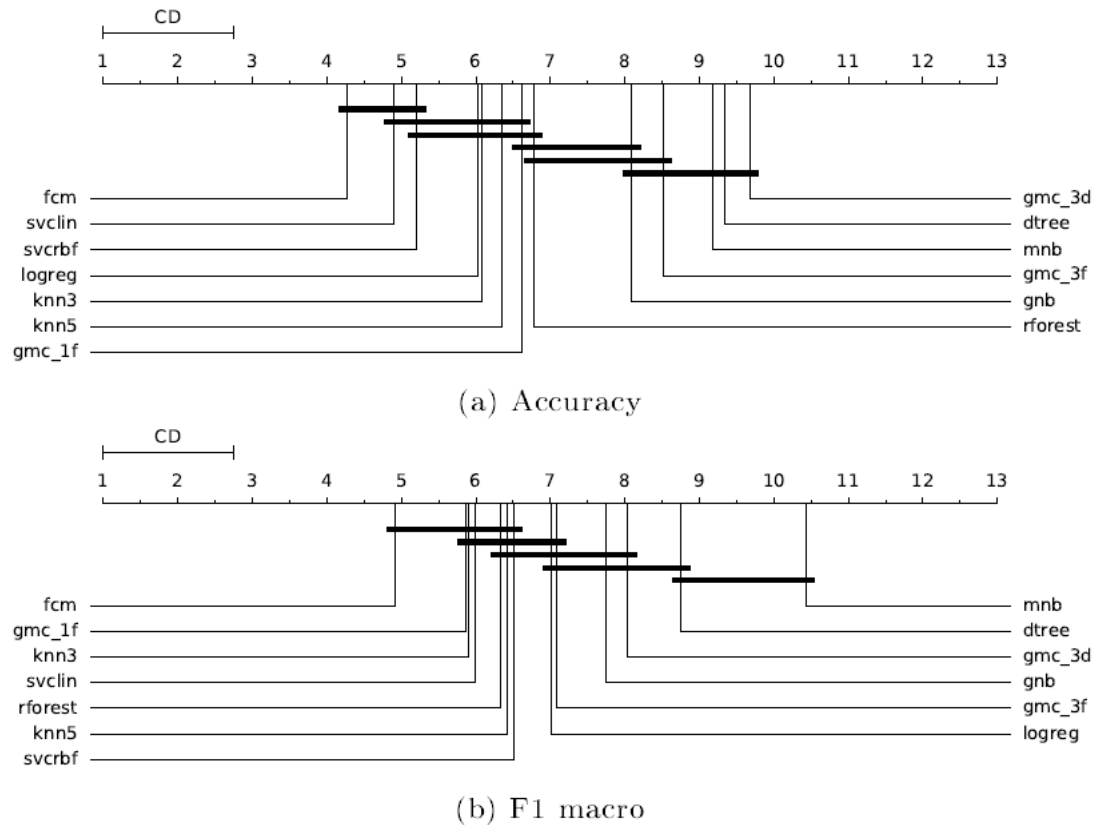


Figure 9: Results of Nemenyi post hoc tests for (a) accuracy (b) F1 macro score computed on folds. Classifiers are ordered according to their average ranks (*smaller=better*). Bold lines link classifiers that could not be discerned according to critical difference (CD) computed during the test.

Wyniki – transformacja cech

Tam gdzie FCM klasyfikuje
dobrze również
transformuje dobrze
(według metryk
grupowania)

Dataset	DB orig	DB transf	SLH orig	SLH transf	CH orig	CH transf
iris	0.87 0.84	0.36 0.37	0.46 0.46	0.77 0.76	252.51 68.64	669.07 194.82
wine	1.31 1.20	0.24 0.34	0.29 0.32	0.84 0.76	64.76 19.66	1246.62 150.84
breast cancer	1.22 1.20	0.28 0.34	0.34 0.34	0.80 0.75	251.74 66.05	3820.31 513.49
glass	4.09 2.16	3.93 2.46 •	-0.05 0.00	0.04 -0.00 •	20.81 11.38	28.17 7.91 •
seeds	0.93 0.88	0.73 0.74	0.38 0.42	0.54 0.54	219.51 67.24	251.18 74.09
ionosphere	3.85 3.38	2.10 2.49	0.16 0.17	0.29 0.24	16.15 5.53	56.94 11.10
sonar	4.80 2.68	2.26 2.52	0.04 0.12	0.16 0.14	7.03 5.90	31.96 7.68
blood transfusion	4.62 3.55	2.97 4.10 •	0.05 0.07	0.12 0.12	14.55 6.26	59.57 12.58
vehicle	10.11 6.69	5.13 5.46	-0.02 -0.03	0.12 0.10	34.39 9.20	123.04 28.21
ecoli	2.68 1.68	2.37 1.74 •	0.14 0.15	0.20 0.20	73.05 25.67	97.75 29.69
yeast	2.91 3.01	3.62 4.02 •	-0.01 -0.04	-0.04 • -0.06 •	52.55 14.66	70.61 17.91
tic tac toe	5.67 4.28	1.90 2.79	0.04 0.07	0.23 0.12	21.59 10.31	200.93 21.95
heart	2.48 2.38	0.87 1.23	0.13 0.13	0.45 0.31	32.94 8.87	218.97 28.94
haberman	6.37 4.81	5.29 4.50	0.06 0.06	0.07 0.07	3.96 2.13	7.71 2.30
german credit	10.21 8.13	5.52 6.82	0.01 0.01	0.01 0.0 •	6.30 2.50	20.32 3.46
diabets	3.54 3.45	1.56 1.62	0.09 0.09	0.22 0.21	39.65 10.47	189.88 46.55
olivetti faces 8	1.93 1.47	1.42 1.12	0.10 0.06	0.25 0.20	13.62 4.03	18.14 4.78
olivetti faces 16	1.82 1.28	1.34 1.06	0.13 0.12	0.27 0.22	12.03 4.14	15.47 4.24
olivetti faces 28	1.86 1.32	1.64 1.17	0.12 0.09	0.21 0.14	10.66 3.34	11.21 3.12
digits	2.13 1.86	0.73 0.92	0.16 0.20	0.55 0.47	116.14 35.88	759.63 138.53
fashion2000	3.36 3.21	2.57 2.74	0.05 0.05	0.13 0.11	120.67 30.81	140.68 33.64
fashion10000	3.49 3.49	2.34 2.56	0.05 0.05	0.15 0.13	585.49 147.10	689.86 162.06

Table 8: Clustering scores: DB – Davies Bouldin (*lower=better*), SLH – Silhouette Coefficient (*-1: bad, 1: perfect*), CH – Calinski Harabasz (*higher=better*). Scores were computed as averages during 5-fold CV for training folds (normal font) and test folds (bold font). Worse values obtained after data transformation are marked with bullets.

Wyniki

Porównanie efektywności klasyfikatorów działających na oryginalnych i przetransformowanych cechach

Alg	FCM+Alg	Accuracy								F1 macro							
		datasets				folds				datasets				folds			
		t-test		Wilcoxon		t-test		Wilcoxon		t-test		Wilcoxon		t-test		Wilcoxon	
mnb	fcm mnb	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+
gnb	fcm gnb	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+
knn3	fcm knn3	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+
knn5	fcm knn5	-	+	=	=	-	+	-	+	-	+	-	+	-	+	-	+
sverbf	fcm sverbf	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+
svclin	fcm svclin	=	=	=	=	-	+	-	+	-	+	-	+	-	+	-	+
logreg	fcm logreg	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+
dtree	fcm dtree	=	=	=	=	-	+	-	+	-	+	-	+	-	+	-	+
rforest	fcm rforest	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
gmc 1f	fcm gmc 1f	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
gmc 3f	fcm gmc 3f	-	+	-	+	-	+	-	+	=	=	=	=	-	+	-	+
gmc 3d	fcm gmc 3d	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+

Table 11: Statistical comparison of classifier performance tested on original data and data preprocessed with FCM. The table shows results of paired t-test and Wilcoxon for mean score values for the whole datasets, as well as particular folds during 5-fold CV.

Na ogół przetwarzanie wstępne poprawiało rezultaty...

Podsumowanie

- Klasyfikator niskopoziomowy (wydajność spada wraz z liczbą cech)
- W tym momencie gradient obliczany metodą różniczkowania symbolicznego (słaba wydajność dla dużej liczby wsadów)
- Największy zbiór danych (fashion10000) 784 atrybuty i 10000 obserwacji
- Brak metody doboru hiperparametrów
- Mapy nie są interpretowalne

- Rezultaty są obiecujące zarówno dla klasyfikacji, jak i transformacji cech
- Algorytm wyznaczania gradientu bez obliczeń symbolicznych został zaimplementowany (ale poza Tensorflow)
- Kod można znaleźć na https://github.com/pszwed-ai/fcm_classifier_transformer

Dziękuję za uwagę