INSTYTUT BADAŃ SYSTEMOWYCH

POLSKIEJ AKADEMII NAUK

Mgr inż. Jakub Piekoszewski

Rozprawa doktorska p.t.

# Uogólnione samoorganizujące się sieci neuronowe o drzewopodobnych strukturach w grupowaniu danych ze szczególnym uwzględnieniem danych medycznych opisujących ekspresję genów

Promotor:

Prof. dr hab. inż. Marian B. Gorzałczany

Promotor pomocniczy:

Dr inż. Filip Rudziński

WARSZAWA, 2016

## Spis treści

## Wykaz najważniejszych oznaczeń

$L$ – liczba wektorów danych uczących (danych wejściowych),

$l$ – numer wektora danych uczących, $l = 1,2,...,L$,

$\boldsymbol{x}_l$ – wektor danych uczących o numerze $l$; $\boldsymbol{x}_l \in \Re^n$, przy czym $\Re$ oznacza zbiór liczb rzeczywistych, $\boldsymbol{x}_l = [x_{l1}, x_{l2}, ..., x_{ln}]^T$,

$n$ – liczba wejść sieci neuronowej, jak również wymiar przestrzeni wektorów danych uczących oraz wektorów wagowych neuronów sieci,

$i$ – numer wejścia sieci neuronowej oraz numer składowej wektora danych uczących, $i = 1,2,...,n$,

$m$ – liczba neuronów (liczba wyjść) sieci neuronowej,

$j$ – numer neuronu (numer wyjścia) sieci neuronowej, $j = 1,2,...,m$,

$\boldsymbol{w}_j$ – wektor wagowy neuronu o numerze $j$; $\boldsymbol{w}_j \in \Re^n$, $\boldsymbol{w}_j = [w_{j1}, w_{j2}, ..., w_{jn}]^T$,

$k_{\max}$ – maksymalna liczba iteracji procesu uczenia sieci neuronowej (iteracja oznacza jednokrotne przeliczenie jednej próbki danych uczących),

$k$ – numer kolejnej iteracji procesu uczenia sieci neuronowej, $k = 1,2,...,k_{\max}$,

$e_{\max}$ – maksymalna liczba epok procesu uczenia sieci neuronowej (epoka oznacza jednokrotne przeliczenie wszystkich próbek danych uczących); $k_{\max} = L \cdot e_{\max}$,

$e$ – numer kolejnej epoki procesu uczenia sieci neuronowej, $e = 1,2,...,e_{\max}$,

$\eta$ – współczynnik uczenia sieci neuronowej,

WTM – klasa algorytmów nienadzorowanego uczenia konkurencyjnego (ang. *Winner Takes Most*),

$j_{\boldsymbol{x}}$ – numer neuronu zwyciężającego we współzawodnictwie neuronów, gdy na wejścia sieci podawany jest wektor danych uczących $\boldsymbol{x}$, $j_{\boldsymbol{x}} \in \{1,2,...,m\}$ ($\boldsymbol{w}_{j_{\boldsymbol{x}}}$ jest wektorem wagowym neuronu o numerze $j_{\boldsymbol{x}}$),

$S(j, j_{\boldsymbol{x}}, k)$ – funkcja sąsiedztwa topologicznego neuronu zwyciężającego $j_{\boldsymbol{x}}$ w $k$-tej iteracji procesu uczenia, dla $j$-tego neuronu sieci ($j = 1,2,...,m$) w algorytmach WTM,

$\lambda$ – promień sąsiedztwa topologicznego neuronu zwyciężającego o numerze $j_{\boldsymbol{x}}$; parametr funkcji sąsiedztwa $S(j, j_{\boldsymbol{x}}, k)$,

$\boldsymbol{r}_j$ – wektor położenia neuronu o numerze $j$ w topologicznej strukturze samoorganizującej się sieci neuronowej; $\boldsymbol{r}_j \in \Re^q$, $\boldsymbol{r}_j = [r_{j1}, r_{j2}, ..., r_{jq}]^T$, gdzie $q$ jest wymiarem topologicznej struktury sieci (np. $q$ = 1 dla struktury sieci w postaci łańcucha neuronów, $q$ = 2 dla struktury sieci w postaci mapy neuronów),

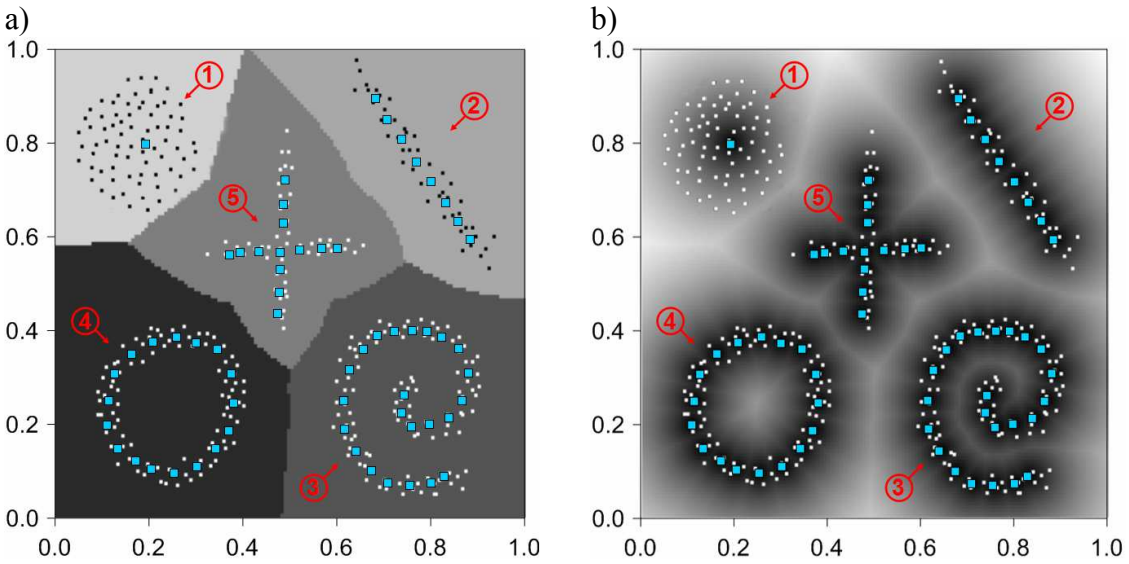
$p$ – liczba neuronów w podstrukturze samoorganizującej się sieci neuronowej ($p \le m$).

# 1. Wprowadzenie

Wraz z postępem technologicznym nastąpił ogromny wzrost możliwości współczesnych systemów informatycznych w zakresie rejestrowania i gromadzenia bardzo dużych ilości informacji. W związku z tym, zaistniała również potrzeba zastąpienia dotychczasowych metod analizy danych, których wydajność staje się powoli niewystarczająca, nowymi – umożliwiającymi automatyczne przetwarzanie dużych ilości danych zawartych w złożonych i wielowymiarowych zbiorach danych. Jednym z ważniejszych nurtów badań naukowych w tym zakresie są prace nad metodami automatycznego grupowania danych (ang. *clustering techniques*) [36, 56, 60, 64, 89, 100], należącymi do szerszej grupy metod tzw. odkrywania wiedzy z danych (ang. *knowledge discovery in data*), czy też drążenia danych (ang. *data mining*) [9, 25, 30, 74, 98, 106, 113]. Metody grupowania danych znajdują szerokie praktyczne zastosowanie, m.in. do budowy inteligentnych systemów wspomagania decyzji, automatycznej klasyfikacji danych, kompresji danych, czy też wyszukiwania korelacji pomiędzy danymi. W niniejszej pracy metody te zostaną zastosowane m.in. do wyszukiwania podobieństw pomiędzy danymi opisującymi ekspresję genów w przypadku różnych chorób nowotworowych.

Grupowanie danych (ang. *clustering*), zwane również analizą skupień (ang. *cluster analysis*), jest procesem polegającym na podziale zbioru danych na podzbiory (grupy) odpowiadające występującym w nim skupiskom danych. Proces odbywa się w taki sposób, aby poszczególne próbki danych przyporządkowane do tego samego podzbioru (grupy) były jak najbardziej do siebie podobne (w sensie określonej miary podobieństwa), natomiast próbki danych przyporządkowane do różnych podzbiorów (grup) były jak najbardziej do siebie niepodobne. Podział na grupy może być „ostry” (ang. *crisp clustering*) lub rozmyty (ang. *fuzzy clustering*) [60, 106]. W pierwszym przypadku, pojedyncza próbka danych należy wyłącznie do jednej grupy, natomiast w drugim należy do każdej z grup w pewnym stopniu, przy czym stopień jej przynależności do grupy jest liczbą rzeczywistą z zakresu od 0 (brak przynależności) do 1 (pełna przynależność). Idea grupowania ostrego i rozmytego została zilustrowana na rys. 1.1 z wykorzystaniem przykładowego zbioru danych (różnymi kolorami oznaczono obszary danych należące do poszczególnych skupisk (w przypadku rys. 1.1a), natomiast różnymi odcieniami szarości – stopnie przynależności próbek danych do grup (w przypadku rys. 1.1b) – ciem-

niejszy odcień oznacza wyższy stopień przynależności). Istnieją warianty grupowania danych dopuszczające brak przynależności próbek danych do jakiejkolwiek grupy (np. próbek reprezentujących szum).




Rys. 1.1. Przykładowy zbiór danych zawierający pięć skupisk danych (punkty koloru czarnego lub białego), w tym skupisko o charakterze objętościowym (1) i cztery skupiska o charakterze szkieletowym (2 – 5) oraz ich wielopunktowe prototypy (punkty koloru niebieskiego), jak również wyniki „ostrego” (a) i rozmytego (b) grupowania danych

Grupowanie danych, w ogólnym przypadku, odbywa się przy braku jakiejkolwiek wstępnej wiedzy o właściwościach zbioru danych, a w szczególności – o liczbie potencjalnych skupisk danych i ich złożoności. I tak, można wyróżnić szereg odrębnych kategorii problemów grupowania danych, w których rozważa się różne warianty zbiorów danych ze skupiskami: a) o charakterze objętościowym lub szkieletowym (patrz rys. 1.1), b) wzajemnie odseparowanymi, stykającymi się lub częściowo nakładającymi się, c) o zbalansowanej lub znacznie zróżnicowanej gęstości danych i/lub wielkości, d) o różnych poziomach zaszumienia lub jego braku, itd. Ponadto, algorytmom grupowania danych często stawia się dodatkowe wymagania, takie jak: automatyczne wyznaczanie liczby skupisk (gdy liczba ta jest nieznana) oraz ich wielopunktowych prototypów (na rys. 1.1 oznaczono kolorem niebieskim przykłady prototypów jedno- (1) i wielopunktowych (2 – 5 )). Wspomniane wymagania mogą dotyczyć również wygenerowania informacji o gęstości, wielkości i kształcie poszczególnych skupisk danych, np. poprzez analizę ich prototypów (analizę liczby punktów prototypu, odległości między nimi, itp.).

Odrębnym problemem jest grupowanie danych w zbiorach zawierających skupiska o znacznie zróżnicowanych kształtach, gęstościach danych lub objętościach. Szczególną kategorią tego typu danych, o wyjątkowo dużej złożoności, są numeryczne wyniki sekwencjonowania ludzkiego genomu [20, 85]. Dane te zostały pozyskane w ramach tzw. Human Genome Project, rozpoczętego w 1990 roku przez Narodowe Centrum Badań Genomu Człowieka (ang. *National Center for Human Genome Research*) przy Narodowym Instytucie Zdrowia USA. W 2001 roku opublikowano na łamach tygodników „Science" i „Nature" około 90% ludzkiego genomu [62, 119]. W niniejszej pracy rozważany jest problem grupowania danych w czterech zbiorach należących do tej kategorii i opisujących tzw. ekspresję genów (ang. *gene expression*), czyli procesy transkodowania informacji genetycznej zawartej w genach na tzw. produkty funkcyjne (RNA lub białka) [54]. Dane te opisują poziomy aktywności poszczególnych genów w trakcie przebiegu takiego procesu. Celem grupowania danych jest wyodrębnienie grup genów (liczba grup nie jest znana z góry) o zbliżonym poziomie aktywności (ang. *coexpressed genes*). Zakłada się, że geny te pełnią podobne funkcje w organizmie człowieka i ewentualnie mogą być odpowiedzialne za powstawanie określonych chorób (w rozważanych przypadkach, chorób nowotworowych) [8, 33, 110]. Rozważane dane charakteryzują się: a) dużą wymiarowością, b) znacznym zaszumieniem wynikającym z niedoskonałości procesów powstawania i przetwarzania tzw. mikromacierzy służących do badania ekspresji genów [54], c) skupiskami o skomplikowanych kształtach (częściej o charakterze szkieletowym, a rzadziej objętościowym) oraz d) zróżnicowanym rozkładem gęstości i objętości (zwykle zawierają duże skupisko danych reprezentujące nieistotny szum oraz wiele małych skupisk reprezentujących poszukiwane grupy genów).

Dotychczas nie opracowano uniwersalnej metody spełniającej wyżej wymienione wymagania i zdolnej do efektywnego rozwiązywania problemów grupowania danych w zbiorach o tak szerokim zakresie złożoności. Tradycyjne podejścia (ich syntetyczny przegląd przedstawia podrozdział 1.1) w zdecydowanej większości przypadków wymagają podania z góry liczby skupisk w zbiorze danych, są ukierunkowane na rozwiązywanie jedynie wybranych kategorii problemów grupowania danych (np. wykrywają skupiska wyłącznie o określonych kształtach; przykładem jest metoda *k-means* [71], która skutecznie wykrywa skupiska głównie o sferycznych kształtach objętościowych) i nie generują wielopunktowych prototypów skupisk danych (ograniczają się zwykle do prototypów jednopunktowych, np. wspomniana metoda *k-means*). Mankamenty te

znacznie ograniczają przydatność metod tradycyjnych w problemach grupowania danych w złożonych, wielowymiarowych zbiorach danych, w tym danych opisujących ekspresję genów. Na tym tle, konieczne jest opracowanie nowych technik grupowania danych, które będą w stanie uwzględnić wymienione wymagania.

Obiecujące wyniki uzyskiwane są przy pomocy metod grupowania danych wywodzących się z obszaru tzw. inteligencji obliczeniowej (ang. *computational intelligence*) [34], bazujących na uczeniu konkurencyjnym (szerszy przegląd wybranych podejść tej kategorii przedstawia rozdział 2). Niniejsza praca proponuje tego typu oryginalne narzędzie, tzw. samoorganizującą się sieć neuronową o ewoluującej, drzewopodobnej strukturze topologicznej, wyposażonej w mechanizmy jej rozłączania i ponownego łączenia wybranych podstruktur. Proponowana sieć neuronowa jest w stanie automatycznie określać liczbę skupisk w rozważanym zbiorze danych (równą liczbie rozłączonych podstruktur sieci) oraz wyznaczać ich wielopunktowe prototypy. Dotyczy to skupisk o różnorodnych kształtach, wielkościach i gęstościach danych w nich zawartych. Proponowana sieć neuronowa (szczegółowo zaprezentowana w rozdziale 3) zostanie zastosowana do efektywnego grupowania danych w różnorodnych, złożonych i wielowymiarowych zbiorach danych (w rozdziale 4), w tym zbiorach danych opisujących ekspresję genów (w rozdziałach 5, 6 i 7). W dalszej części Wprowadzenia, dla porządku, przedstawiono syntetyczny przegląd tradycyjnych technik i strategii grupowania danych (w podrozdziale 1.1), ze szczególnym uwzględnieniem grupowania danych opisujących ekspresję genów (w podrozdziale 1.2). Ponadto, sformułowano tezę ogólną pracy wraz z jej tezami szczegółowymi (w podrozdziale 1.3) i przedstawiono zakres pracy (podrozdział 1.4).

## 1.1. Tradycyjne metody i strategie grupowania danych

Tradycyjne techniki grupowania danych można podzielić na następujące kategorie [25, 56, 60, 64, 89, 100, 113]: metody hierarchiczne, metody bazujące na optymalizacji funkcji celu, metody bazujące na analizie gęstości danych oraz metody wykorzystujące uczenie konkurencyjne.

Metody hierarchiczne (ang. *hierarchical clustering* lub *connectivity-based clustering*) [25, 30, 60, 89, 113] tworzą hierarchiczną strukturę rozkładu skupisk danych w zbiorze w postaci tzw. dendrogramu, wykorzystując w tym celu jedną z dwóch strategii:

dzielącą (ang. *divisive algorithms*) lub łączącą (ang. *agglomerative algorithms*). W pierwszym przypadku, zbiór danych dzielony jest na podzbiory, a następnie powstałe podzbiory dzielone są ponownie na mniejsze, itd. Z kolei, w drugim przypadku, pojedyncze próbki danych łączone są w pary, a następnie powstałe pary próbek łączone są w większe grupy, itd. W obu przypadkach węzły drzewa (dendrogramu) reprezentują podzbiory (skupiska) powstałe w wyniku kolejnych podziałów/łączeń. Rezultatem grupowania są skupiska danych reprezentowane przez liście drzewa, „przyciętego" na określonym (w zależności od potrzeb) poziomie po zakończeniu procesu dzielenia/łączenia. Głównymi mankamentami tych metod są: a) brak zdolności do automatycznego wyznaczania liczby skupisk w zbiorze danych (liczba ta zależy od przyjętego poziomu „przycięcia" drzewa), b) duża wrażliwość na zaszumienie danych, c) brak zdolności do generowania prototypów (nawet jednopunktowych) skupisk danych.

Kolejna kategoria obejmuje metody grupujące dane w ramach optymalizacji pewnej funkcji celu (ang. *objective function-based algorithms*) [25, 30, 60, 89, 113], w szczególności: metody grupowania „ostrego", rozmytego, probabilistycznego i przybliżonego oraz algorytmy ewolucyjne (genetyczne). Metody grupowania „ostrego" (ang. *hard or crisp clustering algorithms*) przyporządkowują poszczególne próbki danych wyłącznie do jednej z grup (liczba grup jest z góry określona). Najbardziej znanym reprezentantem tej kategorii metod jest algorytm *k-means* [71], który w ramach tzw. kwantyzacji wektorowej minimalizuje średnią wariancję odległości (podobieństwa) pomiędzy wszystkimi możliwymi parami wektorów reprezentujących odpowiednio: próbkę danych i najbliższy jej jednopunktowy prototyp skupiska danych. Metody grupowania rozmytego (ang. *fuzzy clustering*) przyporządkowują próbki danych do wszystkich grup jednocześnie w stopniu określonym tzw. funkcją przynależności (przyjmującą wartości od 0 – brak przynależności do 1 – pełna przynależność), znaną z teorii logiki rozmytej [98, 123]. Typowym przedstawicielem tych metod jest algorytm FCM (ang. *Fuzzy C-Means*) [9, 32], będący w pewnym sensie uogólnieniem algorytmu *k-means*. Algorytmy grupowania probabilistycznego (ang. *probabilistic clustering algorithms*), podobnie jak algorytmy grupowania „ostrego", przyporządkowują próbki danych tylko do jednej grupy. Wyznaczają dla każdej z próbek danych prawdopodobieństwa przynależności do poszczególnych grup, a następnie przyporządkowują próbki danych do tej grupy, dla której prawdopodobieństwo jest największe. Analogicznie funkcjonują metody grupowania przybliżonego (ang. *possibilistic clustering algorithms*),

przy czym zamiast prawdopodobieństwa wyznaczają "możliwość" przynależności próbki danych do grupy na podstawie tzw. stopnia jej zgodności (ang. *degree of compatibility*) z jednopunktowym prototypem danej grupy [113]. W odróżnieniu od poprzedniej klasy metod, stopień zgodności zależy wyłącznie od próbki danych i prototypu grupy do której należy, natomiast nie zależy od stopnia jej zgodności z prototypami pozostałych grup. Podsumowując, algorytmy te umożliwiają generowanie jednopunktowych prototypów skupisk danych, natomiast nie wykrywają w sposób automatyczny liczby skupisk danych oraz są wrażliwe na ich kształty (najczęściej dobrze wykrywają skupiska o sferycznych kształtach objętościowych, natomiast słabo radzą sobie z wykrywaniem skupisk o kształtach szkieletowych).

Wyjątkową podkategorię metod grupowania danych, bazujących na optymalizacji funkcji celu, stanowią algorytmy ewolucyjne (genetyczne) (ang. *genetic clustering algorithms*) [41, 43, 45, 80, 98]. Metody te przetwarzają wiele wariantów podziału danych na grupy jednocześnie w ramach pewnej populacji rozwiązań. Poszczególne rozwiązania zakodowane w postaci tzw. chromosomów ewoluują w trakcie trwania procesu optymalizacji, wzajemnie rekombinując ze sobą wzorem organizmów żywych. Celem każdego z nich jest osiągnięcie wyższego „poziomu rozwoju", rozumianego w sensie jak najlepszej jakości podziału danych na grupy. Kluczową rolę w tym procesie odgrywa tzw. funkcja przystosowania (ang. *fitness function*), pełniąca tutaj role funkcji celu i jednocześnie miary jakości grupowania danych (ang. *cluster validity measure*), służącej do oceny poziomu rozwoju (jakości) danego rozwiązania. Przykłady można znaleźć w pracach [7, 11, 79, 84]. Zaletą algorytmów genetycznych jest ich uniwersalny charakter. Mogą być wykorzystane do dowolnej kategorii problemów grupowania danych, pod warunkiem dobrania odpowiedniej miary jakości. Podstawową wadą tych metod jest stosunkowo duża złożoność obliczeniowa i konieczność przetworzenia dużej liczby iteracji (pokoleń w ramach procesu ewolucji) w przypadkach problemów z bardzo złożonymi skupiskami danych, szczególnie o kształtach szkieletowych.

Trzecia kategoria metod grupowania danych obejmuje algorytmy bazujące na analizie gęstości danych (ang. *density-based clustering algorithms*) [100, 36]. Wykorzystują one różnorodne, heurystyczne strategie do wykrywania lokalnych zmian gęstości danych w przestrzeni danych, w celu lokalizacji granic (krawędzi) skupisk danych. Zaletą tych metod jest odporność na zaszumienie danych, zdolność do automatycznego wykrywania liczby skupisk danych oraz do grupowania danych tworzących nawet bardzo

złożone skupiska. Natomiast ich wadą jest duża wrażliwość na zróżnicowany rozkład gęstości danych w skupiskach oraz konieczność dobrania z góry wartości specjalnych parametrów sterujących procesem grupowania, specyficznych dla użytej strategii heurystycznej. Najbardziej znanym przedstawicielem tej kategorii metod jest algorytm DBSCAN (ang. *Density-Based Spatial Clustering of Applications with Noise*) [35], w którym do prawidłowego funkcjonowania konieczne jest dobranie maksymalnego promienia sąsiedztwa dla każdej próbki danych oraz minimalnej liczby próbek koniecznych do utworzenia grupy.

Ostatnią kategorię metod grupowania danych stanowią algorytmy bazujące na uczeniu konkurencyjnym (ang. *competitive learning algorithms*), w szczególności wykorzystujące samoorganizujące się sieci neuronowe (ang. *self-organizing neural networks*) [25, 43, 45, 57, 59, 66, 67, 68, 78, 86, 98, 105, 113, 125] (należy zaznaczyć, że idea uczenia konkurencyjnego jest podstawą funkcjonowania wielu innych sieci neuronowych, np. samouczących się sieci neuronowych [25, 57, 59, 68, 78, 86, 98, 105, 125] czy też sieci ART (ang. *Adaptive Resonance Theory*) [18, 19, 77, 111], jednakże ich zastosowania w problemach grupowania danych są marginalne i nie będą w pracy rozważane). Grupowanie danych odbywa się w ramach procesu uczenia sieci neuronowej w trybie nienadzorowanym (ang. *unsupervised learning*), z wykorzystaniem algorytmu WTM (ang. *Winner Takes Most*) lub jego wariantów. W większości przypadków, metody automatycznie wykrywają liczbę skupisk w zbiorach danych oraz generują wielopunktowe prototypy skupisk w postaci struktur neuronów połączonych ze sobą topologicznymi wiązaniami. Przykładowe podejścia przedstawiono w [44, 46, 47, 48, 49]. Do tej kategorii metod należy również proponowane w pracy narzędzie generujące wielopunktowe prototypy w postaci drzewopodobnej struktury neuronów [50, 51, 52, 53]. Szerszy przegląd metod grupowania danych, wykorzystujących samoorganizujące się sieci neuronowe, zostanie przedstawiony w rozdziale 2.

Podsumowując należy zaznaczyć, że zdecydowana większość tradycyjnych metod funkcjonuje bardzo dobrze jedynie dla wybranych kategorii problemów grupowania danych, a słabo sprawdza się lub w ogóle się nie sprawdza dla pozostałych. Niewiele metod jest zdolnych do realizacji dodatkowych zadań wspomnianych w pierwszej części Wprowadzenia (w szczególności do automatycznego wyznaczania liczby skupisk i ich wielopunktowych prototypów), a te istniejące nie sprawdzają się w problemach wykrywania skupisk o zróżnicowanych gęstościach danych, zróżnicowanych objętościach

oraz złożonych kształtach, w szczególności szkieletowych. W odpowiedzi na powyższe mankamenty, wykorzystuje się różne strategie usprawniające proces grupowania danych. Przykładem jest łączenie klasycznych metod grupowania danych, specjalizujących się w odmiennych kategoriach problemów, w tzw. zespoły (ang. *clustering ensembles*) [118]. Rezultaty grupowania danych – zwane inaczej „podziałami" (ang. *partitions*) – uzyskane z wykorzystaniem poszczególnych metod wchodzących w skład zespołu wiązane są ze sobą w jeden, ostateczny podział o potencjalnie wyższej jakości. Efektywność tej strategii nie jest obecnie zbyt wysoka z powodu stosowanych niedoskonałych miar podobieństwa pomiędzy podziałami (np. *Rand index* [93]). Kolejna strategia, która ma na celu wykrywanie nieznanej liczby skupisk w zbiorze danych, jest stosowana z wykorzystaniem metod grupowania wymagających określenia z góry tej liczby. W ramach tej strategii, procedura grupowania danych jest wielokrotnie powtarzana dla różnych liczb skupisk (najczęściej z zakresu [2, $k_{\max}$], gdzie $k_{max}$ jest zakładaną, maksymalną liczbą skupisk). Spośród uzyskanych podziałów wybierany jest jeden ostateczny, którego jakość – wyznaczona za pomocą zewnętrznych miar jakości (ang. *cluster validity measures*) – jest najlepsza. Wadami tej strategii są: a) wielokrotne zwiększenie złożoności obliczeniowej procesu (proporcjonalnie do liczby powtórzeń), b) brak jasnych kryteriów doboru maksymalnej liczby skupisk, c) zależność rezultatów od rodzaju użytej miary jakości. Szczególnie skomplikowanym zadaniem wymagającym stosowania specjalnych strategii jest grupowanie danych opisujących ekspresję genów. Strategie te zostaną przedstawione w kolejnym podrozdziale, jako kontynuacja dotychczasowych rozważań.

## 1.2. Strategie grupowania danych opisujących ekspresję genów

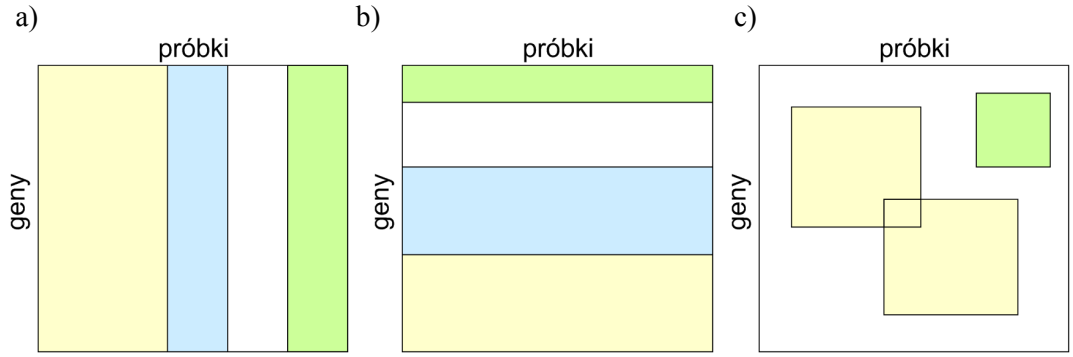
Dane opisujące ekspresję genów są wynikiem serii eksperymentów badawczych z obszaru mikrobiologii molekularnej, w których kluczową rolę pełnią tzw. mikromacierze DNA (ang. *DNA microarrays, DNA chips, gene chips*). Mikromacierz jest szklaną płytką z naniesionymi tysiącami pól o mikroskopowej wielkości w formie uporządkowanej, prostokątnej siatki (macierzy). Każde pole to sonda (ang. *spot*) zawierająca krótki fragment wzorcowej sekwencji DNA (unikatowej względem pozostałych sond), reprezentująca jeden gen o znanej nazwie kodowej. W ujęciu uproszczonym, pojedynczy eksperyment polega na naniesieniu na płytkę próbki materiału genetycznego (np. tkanki zdrowej lub nowotworowej pacjenta) oznakowanej specjalnym markerem (barwnikiem)

fluorescencyjnym i oświetlaniu jej promieniem lasera. Na powierzchni każdej sondy dochodzi do tzw. procesu hybrydyzacji, w którym wzbudzone laserem wzorcowe sekwencje DNA łączą się w pary z komplementarnymi sekwencjami DNA badanej próbki, a fluorescencyjny barwnik emituje w tym czasie światło o natężeniu proporcjonalnym do długości połączonej sekwencji DNA. Intensywność emisji światła fluorescencyjnego wyraża poziom aktywności genu w trakcie przebiegu procesu hybrydyzacji. Eksperyment jest powtarzany wielokrotnie dla różnych próbek materiału genetycznego, np. komórek tego samego typu będących w stanie patologicznym i normalnym lub należących do różnych pacjentów. Wyniki takiej serii eksperymentów zapisywane są do jednego zbioru danych w formie $u \times v$ wymiarowej macierzy liczb rzeczywistych (zwanej również macierzą ekspresji genów) $\boldsymbol{A}_{u\times v} = [a_{ij}]_{u\times v}$, $i = 1,2,...,u$, $j = 1,2,...,v$, gdzie liczba wierszy $u$ jest równa liczbie genów uwzględnionych w eksperymentach, natomiast liczba kolumn $v$ jest równa liczbie próbek materiału genetycznego (lub też równa liczbie eksperymentów). Wartość $a_{ij}$ wyraża poziom aktywności $i$-tego genu w $j$-tej próbce (w $j$-tym eksperymencie).

Analiza podobieństwa poziomów aktywności genów w macierzy $\boldsymbol{A}_{u\times v}$ może dostarczyć wiele cennych informacji, m.in. o mechanizmach powstawania chorób nowotworowych, o odpowiedzialności genów za określone funkcje w organizmie, o sposobach identyfikacji (klasyfikacji) przypadków chorób, rodzajów tkanek, itd. W szczególności, metody analizy skupisk znajdują tu kluczowe zastosowania. W zależności od przyjętej reprezentacji tzw. przestrzeni cech danych wyrażających ekspresję genów, wyróżnia się trzy podejścia (patrz również rys. 1.2):

- grupowanie bazujące na próbkach (ang. *sample-based clustering*) lub inaczej grupowanie próbek (rys. 1.2a), w którym przestrzeń cech jest reprezentowana przez $n$ elementowy zbiór identyfikatorów genów (nazw kodowych genów); celem grupowania jest wykrycie skupisk próbek materiału genetycznego o jak największym stopniu podobieństwa (dla których geny o tej samej nazwie kodowej posiadają zbliżony poziom aktywności); rozważa się grupowanie danych w zbiorze $\boldsymbol{X}_{L\times n} = (\boldsymbol{A}_{u\times v})^T = \boldsymbol{A}^T_{v\times u}$, $\boldsymbol{X}_{L\times n} = \{\boldsymbol{x}_l^T\}_{l=1}^L$, $\boldsymbol{x}_l = [x_{l1}, x_{l2}, \ldots, x_{ln}]^T$, $x_{li} = a_{il}$, $l = 1, 2,..., L$, $i = 1, 2,..., n$, gdzie $L = v$ jest liczbą próbek, natomiast $n = u$ jest liczbą genów,

- grupowanie bazujące na genach (ang. *gene-based clustering*) lub inaczej grupowanie genów (rys. 1.2b), w którym przestrzeń cech jest reprezentowana przez *n* elementowy zbiór identyfikatorów próbek materiału genetycznego (etykiet przypadków klinicznych choroby lub pacjentów); celem grupowania jest wykrycie skupisk genów o jak największym stopniu podobieństwa (skupisk genów posiadających zbliżony poziom aktywności dla tych samych próbek materiału genetycznego); rozważa się grupowanie danych w zbiorze $\boldsymbol{X}_{L\times n} = \boldsymbol{A}_{u\times v}$, $\boldsymbol{X}_{L\times n} = \{\boldsymbol{x}_l^T\}_{l=1}^{L}$, $\boldsymbol{x}_l = [x_{l1}, x_{l2}, \ldots, x_{ln}]^T$, $x_{li} = a_{li}$, $l = 1, 2, ..., L$, $i = 1, 2, ..., n$, gdzie $L = u$ i $n = v$ oznaczają, odpowiednio, liczbę genów i liczbę próbek,
- grupowanie w podprzestrzeniach (ang. *subspace clustering*) lub inaczej równoczesne grupowanie próbek i genów (rys. 1.2c), w którym przestrzeń cech nie jest jednoznaczna, lecz identyfikowana automatycznie w trakcie trwania procesu grupowania i niezależnie dla każdego skupiska danych; w ogólnym przypadku, przestrzeń cech może być reprezentowana zarówno przez pewien podzbiór identyfikatorów genów, jak i podzbiór identyfikatorów próbek materiału genetycznego; celem grupowania jest wykrycie genów o zbliżonym poziomie aktywności w próbkach materiału genetycznego o dużym stopniu podobieństwa.




Rys. 1.2. Ilustracja idei grupowania bazującego na próbkach (a), bazującego na genach (b) oraz grupowania w podprzestrzeniach (c)

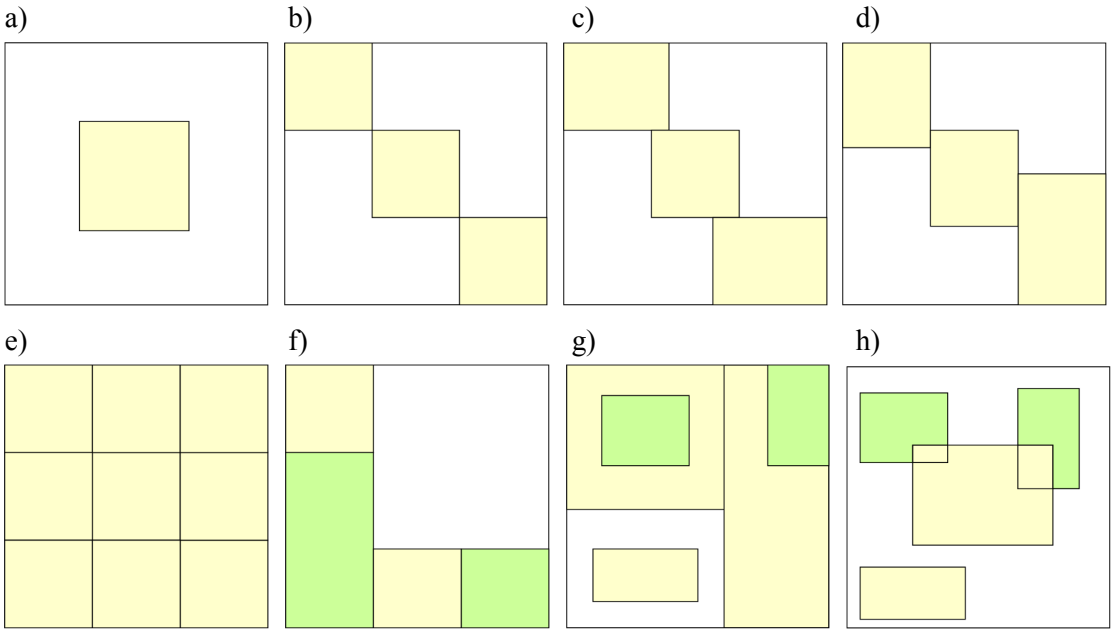
Grupowanie bazujące na próbkach umożliwia wyodrębnianie próbek materiału genetycznego komórek o podobnym stanie klinicznym (np. grupy komórek w stanie patologicznym). Analiza ma zastosowanie m.in. do wykrywania pacjentów z podobnymi objawami chorobowymi, wykrywania nieznanych rodzajów chorób, ich mutacji i podtypów [42], itp. Z kolei, grupowanie bazujące na genach wykrywa grupy genów pełniące podobne funkcje w organizmie człowieka. Dalsza identyfikacja danej funkcji genów

jest możliwa np. metodami medycznymi (nie będą one rozważane) lub poprzez porównanie genów ze znanymi wzorcami przechowywanymi w bazach danych genów. W niniejszej pracy zostanie w tym celu wykorzystana baza genów DAVID (ang. *Database for Annotation, Visualization and Integrated Discovery*) [28], dostępna na serwerze Laboratory of Immunopathogenesis and Bioinformatics, National Cancer Institute at Frederick w USA (http://david.abcc.ncifcrf.gov) oraz współpracujące z nią oprogramowanie do analizy funkcyjnej genów tzw. DAVID Functional Annotation Tool. Ostatnie podejście – grupowanie w podprzestrzeniach (w literaturze znane również jako *biclustering*, *bidimensional clustering*, *simultaneous clustering*, *block clustering* oraz *coclustering* [81]) – jak już wcześniej stwierdzono, wyodrębnia podzbiory genów o zbliżonej aktywności w określonych podzbiorach podobnych do siebie próbek materiału genetycznego.

Do grupowania bazującego na próbkach oraz bazującego na genach wykorzystuje się klasyczne techniki grupowania danych, których syntetyczny przegląd został przedstawiony powyżej, w rozdziale 1.1. Natomiast grupowanie w podprzestrzeniach wymaga zastosowania specyficznych dla tego podejścia strategii. Przyjmuje się założenie, że z pierwotnej macierzy ekspresji genów $\boldsymbol{A}_{u\times v}$ (zawierającej poziomy aktywności wszystkich $u$ typów genów w $v$ zbadanych próbkach materiału genetycznego) można wyodrębnić pewną, nieznaną z góry, liczbę mniejszych macierzy $\boldsymbol{A}'_{p\times q}$ – tzw. biklastrów (ang. *biclusters*) – zawierających poziomy aktywności podzbioru genów ($p < u$), wybranych z podzbioru próbek materiału genetycznego ($q < v$). W pracy [73] dokonano kategoryzacji biklastrów ze względu na ich wzajemną lokalizację w macierzy ekspresji genów. Wyróżniono dziewięć kategorii biklastrów (dla wygody osiem z nich zostało zilustrowanych na rys. 1.3):

- pojedynczy biklaster (przypadek szczególny) (rys. 1.3a); przykłady metod wykrywających tego typu biklastry można znaleźć w [8] i [83],
- rozłączne biklastry wierszy i kolumn (ang. *exclusive row and column biclusters*) (rys. 1.3b); przypadek rozważany w pracach [17] i [121],
- rozłączne biklastry wierszy (ang. *exclusive-rows biclusters*) (rys. 1.3c) lub kolumn (ang. *exclusive-columns biclusters*) (rys. 1.3d); metody zaproponowane w [103] i [108] mają zastosowanie do kategorii rozłącznych biklastrów wierszy oraz – po dokonaniu transpozycji macierzy ekspresji genów – do kategorii rozłącznych biklastrów kolumn,

- nienakładające się biklastry o strukturze szachownicy (ang. *non-overlapping biclusters with checkerboard structure*) (rys. 1.3e); przykłady można znaleźć w [65],
- nienakładające się biklastry o strukturze drzewa (ang. *non-overlapping biclusters with tree structure*) (rys. 1.3f); przykład metody ukierunkowanej na wykrywanie tej kategorii biklastrów zaproponowano w [112],
- nienakładające się biklastry nierozłączne (stykające się bokami) (ang. *non-overlapping non-exclusive biclusters*); przypadek, który można uznać za podkategorię nienakładających się biklastrów o strukturze szachownicy lub o strukturze drzewa,
- nakładające się biklastry o strukturze hierarchicznej (ang. *overlapping biclusters with hierarchical structure*) (rys. 1.3g); przypadek rozważany w [63],
- nakładające się biklastry rozmieszczone w dowolny sposób (ang. *arbitrarily positioned overlapping biclusters*) (rys. 1.3h); metody do pewnego stopnia uniwersalne, umożliwiające wykrywanie biklastrów zlokalizowanych w różny sposób w macierzy ekspresji genów można znaleźć w pracach [5, 22, 40, 70, 95, 102, 122].




Rys. 1.3. Podział biklastrów na kategorie ze względu na ich wzajemną lokalizację w macierzy ekspresji genów [73]: pojedynczy biklaster (a), rozłączne biklastry wierszy i kolumn (b), rozłączne biklastry wierszy (c), rozłączne biklastry kolumn (d), nie nakładające się biklastry o strukturze szachownicy (e), nie nakładające się biklastry o strukturze drzewa (f), nakładające się biklastry o strukturze hierarchicznej (g) oraz nakładające się biklastry rozmieszczone w różnorodny sposób (h)

Wykrywanie biklastrów należących do różnych kategorii wymaga stosowania odmiennych strategii algorytmicznych. Przegląd wybranych strategii można znaleźć w pracach [73, 99]. W ogólnym przypadku, wyróżnia się podejścia: a) iteracyjnie łączące wiersze i kolumny, b) bazujące na strategii algorytmicznej "dziel i zwyciężaj", c) bazujące na strategii algorytmicznej zachłannej, d) bazujące na identyfikacji rozkładu parametrów oraz e) wyliczeniowe.

Strategia iteracyjnie łącząca wiersze i kolumny (ang. *iterative row and column clustering combination*) polega wykrywaniu biklastrów poprzez kombinację rozwiązań grupowania próbek materiału genetycznego i genów, uzyskanych niezależnie z wykorzystaniem klasycznych technik i strategii grupowania danych (przedstawionych w rozdziale 1.1). I tak, po zakończeniu procesów grupowania genów i próbek, w macierzy ekspresji genów można wyróżnić niewielkie biklastry, tzn. podmacierze genów o zbliżonym poziomie aktywności, które stanowią część wspólną pewnych dwóch skupisk: skupiska próbek (wykrytego w rezultacie grupowania bazującego na próbkach) i skupiska genów (wykrytego w rezultacie grupowania bazującego na genach). Następnie, zmienia się położenie biklastrów w macierzy (poprzez zmianę położenia wybranych grup kolumn lub wierszy), w taki sposób, aby porównywalne biklastry były zlokalizowane bezpośrednio obok siebie i tworzyły jeden większy biklaster (dwa biklastry są porównywalne, jeżeli poziomy aktywności genów jednego są zbliżone do poziomów aktywności genów drugiego). Proces powtarza się wielokrotnie, aż do uzyskania możliwie jak największych biklastrów. Powyższa strategia została wykorzystana m.in. w metodach: ITWC (ang. *Interrelated Two-Way Clustering*) [108] (do grupowania próbek i genów użyto algorytmu *k-means*), CTWC (ang. *Coupled Two-Way Clustering)* [40] (grupowanie danych przeprowadzono z wykorzystaniem hierarchicznej metody SPC (ang. *SuperParamagnetic Clustering*) [15]) oraz DCC (ang. *Double Conjugated Clustering*) [17] (grupowanie próbek i genów z wykorzystaniem samoorganizujących się sieci neuronowych).

Podejście bazujące na strategii algorytmicznej "dziel i zwyciężaj" (ang. *divide and conquer*) polega na budowaniu podziału macierzy ekspresji genów na biklastry w dwóch etapach. W pierwszym etapie macierz dzielona jest na dwie części w taki sposób, aby uzyskać najmniejszą wariancję poziomów aktywności genów wewnątrz obu części. Następnie, wzorem strategii „dziel i zwyciężaj”, obie powyższe części są ponownie dzielone na mniejsze w ten sam sposób. Proces jest powtarzany, aż do osiągnię-

cia zakładanej z góry liczby części. W każdym przypadku podziału, granica może przebiegać pomiędzy wierszami lub pomiędzy kolumnami macierzy (wybierany jest korzystniejszy wariant). Przed podziałem, wiersze i kolumny są sortowane względem swego podobieństwa (tzn., podobieństwa poziomów aktywności zawartych w nich genów). Ostateczny podział w ramach pierwszego etapu wyznacza początkowe biklastry, które w kolejnym etapie łączone są w większe, podobnie jak to ma miejsce w strategii iteracyjnej łączącej wiersze i kolumny. Grupowanie blokowe (ang. *block clustering*) [55] jest typowym przykładem algorytmu wykorzystującego tę strategię. Inne przykłady można znaleźć w pracach [95, 121].

W ramach strategii zachłannej (ang. *greedy algorithm*) macierz ekspresji genów jest kopiowana na potrzeby dalszego przetwarzania w celu wyodrębnienia z niej pojedynczego biklastra. Następnie macierz jest stopniowo modyfikowana poprzez usuwanie/wstawianie z/do niej wierszy lub kolumn, skopiowanych z oryginalnej macierzy ekspresji genów w taki sposób, aby jej tzw. spójność (ang. *coherence*) oraz rozmiar były jak największe. Miarą spójności macierzy jest średnia arytmetyczna kwadratów tzw. pozostałości jej elementów (ang. *mean squared residue*) (szczegóły można znaleźć w pracy [73]). W kolejnych iteracjach wybierany jest taki rodzaj operacji modyfikującej macierz, który w danej chwili wydaje się najkorzystniejszy (chociaż w ujęciu ostatecznym może okazać się błędny). Podstawową wadą tego podejścia jest możliwość łatwego „przeoczenia" dobrego rozwiązania w trakcie trwania procesu przetwarzania macierzy, co zwykle prowadzi do uzyskania słabego rezultatu (np. relatywnie małego biklastra). W celu wykrycia większej liczby biklastrów, procedura powinna być powtarzana dla różnych, losowych permutacji lokalizacji wierszy i kolumn w początkowej macierzy. Przykładowe algorytmy bazujące na strategii zachłannej przedstawiają prace [5, 8, 22, 70, 122].

W przypadku strategii bazującej na identyfikacji rozkładu parametrów (ang. *distribution parameter identification*) zakłada się pewien model statystyczny danych w macierzy ekspresji genów i podejmuje się próbę identyfikacji jego parametrów poprzez optymalizację specjalnej funkcji celu. Podobnie jak w przypadku strategii zachłannej, tym razem również strategia pozwala na wyodrębnienie z macierzy tylko jednego biklastra. Pozostałe biklastry mogą być znalezione po wielokrotnym powtórzeniu procesu optymalizacji dla różnych, losowych parametrów inicjujących. Przykładowe metody z tej kategorii opisane są w pracach [65, 100, 102, 103].

Strategia wyliczeniowa (ang. *exhaustive bicluster enumeration*) polega na wyznaczeniu wszystkich możliwych biklastrów istniejących w macierzy ekspresji genów. Spośród nich wybierana jest pewna liczba najlepszych, według z góry określonych preferencji. Wadą podejścia jest bardzo duża złożoność obliczeniowa. Pewną redukcję złożoności obliczeniowej można osiągnąć ograniczając maksymalną liczbę wierszy i kolumn w pojedynczym biklastrze (np. algorytm SAMBA (ang. *Statistical-Algorithmic Method for Bicluster Analysis*) [107]). Inne przykładowe podejścia prezentują prace [4, 70, 117].

Reasumując, przedstawiony powyżej przegląd różnych typów rozważanych biklastrów oraz różnorodnych metod ich wykrywania wskazuje pośrednio na generalnie niezbyt dużą uniwersalność poszczególnych metod oraz znaczny udział elementów heurystycznych w ich funkcjonowaniu. Co więcej, zastosowanie metod ukierunkowanych na wykrywanie określonych typów biklastrów wymaga zwykle pewnej wstępnej wiedzy o rozważanym zbiorze danych, która to wiedza nie musi być zawsze dostępna. Ponadto, zastosowanie powyższych metod w sytuacji, gdy określone typy biklastrów nie występują w rozważanym zbiorze danych musi dać zafałszowany obraz skupisk w tym zbiorze. Biorąc powyższe pod uwagę, można stwierdzić, że efektywne metody grupowania próbek i grupowania genów (pierwsze dwa podejścia zilustrowane na rys. 1.2a, b) wciąż odgrywają bardzo istotną rolę w analizie danych pochodzących z mikromacierzy DNA. Proponowana w ramach niniejszej rozprawy metoda grupowania danych opisujących ekspresję genów ma zastosowanie przede wszystkim do wspomnianego grupowania bazującego na próbkach oraz bazującego na genach. Niemniej jednak, metoda ta może być również wykorzystana do grupowania danych w podprzestrzeniach, w ramach scharakteryzowanej wcześniej strategii iteracyjnie łączącej wiersze i kolumny.

## 1.3. Teza pracy

Podsumowując rozważania przedstawione dotychczas we Wprowadzeniu, można zauważyć, że efektywne rozwiązywanie problemów grupowania danych w różnorodnych, złożonych i wielowymiarowych zbiorach danych, w tym danych opisujących ekspresję genów, wciąż pozostaje otwartym, aktywnym i ważnym obszarem badawczym. Niniejsza praca, lokująca się w tym obszarze, proponuje oryginalne, uniwersalne narzędzie do grupowania danych – uogólnioną samoorganizującą się sieć neuronową o ewoluującej, drzewopodobnej strukturze topologicznej wyposażonej w mechanizmy jej roz-

łączania i ponownego łączenia wybranych podstruktur w celu jak najlepszego odwzorowania skupisk występujących w rozważanym zbiorze danych. Proponowane podejście adresuje poruszane wcześniej wymagania, a w szczególności:

- jest w stanie automatycznie wykrywać liczbę skupisk w rozważanym zbiorze danych,
- jest w stanie automatycznie tworzyć wielopunktowe prototypy wykrytych skupisk, przy czym liczba, rozmiary, kształty i lokalizacje w przestrzeni danych tych prototypów są również automatycznie „dostrajane" w taki sposób, aby jak najlepiej odwzorować rzeczywisty rozkład skupisk w zbiorze danych,
- jest w stanie wykrywać skupiska danych praktycznie o dowolnych kształtach, w tym zarówno skupiska o charakterze objętościowym, jak i szkieletowym oraz skupiska o zróżnicowanych rozmiarach i zróżnicowanej gęstości zawartych w nich danych,
- jest w stanie automatycznie przyporządkowywać poszczególne dane do określonych grup,
- posiada zdolność do budowy rozmytego modelu grupowania danych poprzez określenie stopnia przynależności poszczególnych danych do grup,
- przetwarza dane w trybie uczenia nienadzorowanego,
- posiada zdolność do uogólniania nabytej wiedzy na przypadki nie występujące w fazie uczenia (praca systemu jako klasyfikatora),
- efektywnie przetwarza duże zbiory danych pochodzące z baz danych lub zasobów sieci Internet, w tym zbiory będące rezultatem procesów przetwarzania łańcuchów DNA z wykorzystaniem tzw. mikromacierzy (np. zbiory danych opisujących ekspresję genów).

Zatem można postawić następującą tezę, której prawdziwość zostanie wykazana w niniejszej pracy:

**Uogólniona samoorganizująca się sieć neuronowa o ewoluującej, drzewopodobnej strukturze topologicznej może być wykorzystana jako efektywne i uniwersalne narzędzie teoretyczne w problemach grupowania danych w złożonych, wielowymiarowych zbiorach danych, w tym danych medycznych opisujących ekspresję genów.**

W celu wykazania prawdziwości powyższej tezy ogólnej sformułowano następujące zadania (tezy) szczegółowe:

- Syntetyczne zaprezentowanie dotychczasowych technik i strategii grupowania danych, adresujących do pewnego stopnia rozważane we Wprowadzeniu problemy oraz omówienie ich mankamentów; zadanie to jest punktem wyjścia do dalszych rozważań.

- Przedstawienie konstrukcji oryginalnego narzędzia grupowania danych – uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze – jako odpowiedzi na mankamenty dotychczasowych technik grupowania danych. Warto zauważyć, że:

  - narzędzie to generuje – w fazie uczenia w trybie nienadzorowanym – wielopunktowe prototypy skupisk wykrytych w rozważanym zbiorze danych, w tym:
    - automatycznie generuje po jednym wielopunktowym prototypie dla każdego skupiska danych,
    - automatycznie „dostraja" położenie tych prototypów w przestrzeni danych stosownie do lokalizacji reprezentowanych przez nie skupisk,
    - automatycznie dostosowuje liczbę punktów każdego z prototypów (rozmiar prototypu) i jego kształt, stosownie do wielkości i kształtu reprezentowanego przez niego skupiska danych oraz gęstości danych w nim zawartych,
  - generowanie wielopunktowych prototypów odbywa się poprzez stopniową ewolucję drzewopodobnej struktury topologicznej sieci neuronowej z wykorzystaniem następujących mechanizmów, automatycznie uaktywnianych w trakcie trwania procesu uczenia:
    - mechanizmu usuwania pojedynczych, małoaktywnych neuronów ze struktury sieci (niska aktywność neuronu wiąże się z jego lokalizacją w obszarze o małej gęstości lub braku danych),
    - mechanizmu usuwania wybranych, pojedynczych połączeń topologicznych pomiędzy neuronami w celu podziału struktury sieci na dwie podstruktury, które w trakcie dalszego uczenia mogą się znów dzielić na części (pojedyncza podstruktura reprezentuje jeden wielopunktowy prototyp określonego skupiska danych),
    - mechanizmu usuwania podstruktur sieci, które zawierają zbyt małą liczbę neuronów,

  - mechanizmu wstawiania do struktury (podstruktury) sieci dodatkowych neuronów w otoczeniu neuronów bardzo aktywnych w celu przejęcia części ich aktywności (w efekcie następuje „przesuwanie" neuronów do obszarów o wyższej gęstości danych),
  - mechanizmu wstawiania dodatkowych połączeń topologicznych pomiędzy określonymi, niepołączonymi neuronami w celu ponownego łączenia wybranych podstruktur sieci (łączenie dwóch prototypów w jeden większy).

- Opracowanie algorytmu uczenia uogólnionej samoorganizującej się sieci neuronowej o drzewopodobnej strukturze poprzez rozbudowanie znanych metod uczenia konkurencyjnego o wymienione wyżej dodatkowe mechanizmy.
- Test praktycznej użyteczności proponowanej sieci neuronowej w problemach grupowania danych w różnorodnych, złożonych i wielowymiarowych zbiorach danych, takich jak:
  - syntetyczne dwu- i trójwymiarowe zbiory danych, które uznawane są za wzorcowe testy (ang. *benchmarks*) reprezentujące różne kategorie problemów grupowania danych, w tym:
    - pięć zbiorów danych dwuwymiarowych zawierających skupiska o różnorodnych kształtach i rozkładzie gęstości danych,
    - pięć zbiorów danych należących do tzw. pakietu FCPS (ang. *Fundamental Clustering Problem Suite*) dostępnego na serwerze Uniwersytetu Philipps'a w Marburgu w Niemczech (https://www.uni-marburg.de/fb12/datenbionik/data?language_sync=1),
  - rzeczywiste, wielowymiarowe zbiory danych, dostępne na serwerze Uniwersytetu Kalifornijskiego w Irvine (http://archive.ics.uci.edu/ml), które są powszechnie wykorzystywane jako testy odniesienia w analizach porównawczych m.in. technik grupowania danych; do testów wykorzystano trzy reprezentatywne zbiory danych:
    - zbiór *Congressional Voting Records* zawierający 16 atrybutów symbolicznych oraz 435 rekordów podzielonych na 2 skupiska (klasy) danych,
    - zbiór *Breast Cancer Wisconsin (Diagnostic)* zawierający 30 atrybutów numerycznych oraz 569 rekordów podzielonych na 2 skupiska (klasy) danych,

  - zbiór *Wine* zawierający 13 atrybutów numerycznych oraz 178 rekordów podzielonych na 3 skupiska (klasy) danych,

- rzeczywiste, złożone i wielowymiarowe zbiory danych medycznych, dostępne na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/datasets.html), opisujące poziomy ekspresji genów; rozważono następujące zbiory danych:
  - zbiór *Leukemia* reprezentujący różne typy nowotworów układu krwionośnego; zbiór zawiera 3571 atrybutów numerycznych oraz 72 rekordy,
  - zbiór *Mixed Lineage Leukemia* (*MLL*) prezentujący rozszerzony – w stosunku do zbioru *Leukemia* – zbiór typów nowotworów układu krwionośnego; zbiór zawiera 2474 atrybutów numerycznych oraz 72 rekordy,
  - zbiór *Colon* reprezentujący przypadki choroby nowotworowej jelita grubego; zbiór zawiera 1096 atrybutów numerycznych oraz 62 rekordy,
  - zbiór *Lymphoma* reprezentujący typy nowotworów układu limfatycznego; zbiór zawiera 4026 atrybutów numerycznych oraz 62 rekordy.

- Przeprowadzenie analizy porównawczej proponowanego podejścia z szeregiem innych, do pewnego stopnia alternatywnych, metodologii.

## 1.4. Zakres pracy

Niniejsza praca składa się z ośmiu rozdziałów, wykazu literatury oraz dwóch dodatków. Praca zawiera część teoretyczną (podrozdziały 1.1 i 1.2 Wprowadzenia oraz rozdziały 2 i 3) oraz część aplikacyjną (rozdziały 4, 5, 6 i 7). W części teoretycznej przedstawiono przegląd współczesnych technik grupowania danych, adresujących do pewnego stopnia problemy poruszane we Wprowadzeniu, m.in. automatyczne wyznaczanie liczby skupisk w rozważanym zbiorze danych oraz generowanie wielopunktowych prototypów dla skupisk o różnorodnych kształtach, wielkościach i gęstościach zawartych w nich danych. Przegląd ten stanowi punkt wyjścia do dalszych rozważań nad proponowaną w następnej kolejności oryginalną techniką grupowania danych – uogólnioną samoorganizującą się siecią neuronową o ewoluującej, drzewopodobnej strukturze. W części aplikacyjnej pracy przedstawiono funkcjonowanie proponowanego narzędzia w różnorodnych problemach grupowania danych, w tym – w syntetycznych zbiorach danych, uznawanych za wzorcowe testy różnych technik grupowania oraz w rzeczywistych zbiorach danych udostępnianych na serwerze Uniwersytetu Kalifornijskiego w Irvine (będących uniwersalnymi testami odniesienia w analizach porównawczych) jak również w zbiorach danych medycznych – zawartych w repozytorium Uniwersytetu Shenzhen w Chinach – będących wynikiem przetwarzania informacji genetycznej i opisujących ekspresję genów.

W rozdziale 2, w ramach wspomnianego wyżej przeglądu współczesnych metod grupowania danych, rozważono podejścia wykorzystujące samoorganizujące się sieci neuronowe, w tym, a) konwencjonalną, samoorganizującą się sieć neuronową (SOM), jako bazę do konstrukcji pozostałych metod rozważanych w pracy oraz b) wybrane warianty SOM o zmieniającej się strukturze topologicznej w trakcie trwania procesu uczenia. Warianty te obejmują: a) podejścia GSOM (ang. *Growing SOM*) [94] i GGN (ang. *Growing Grid Network*) [39], które umożliwiają sukcesywne rozbudowywanie struktury sieci poprzez dodawanie do niej nowych neuronów, b) sieci MIGSOM (ang. *Multilevel Interior Growing Self-organizing Maps*) [6] i IGG (ang. *Incremental Grid Growing*) [14], które ponadto są zdolne do podziału struktury na części, c) pięć najbardziej zaawansowanych wariantów SOM: GNG (ang. *Growing Neural Gas*) [38], IGNG (ang. *Incremental Growing Neural Gas*) [92], GCS (ang. *Growing Cell Structures*) [37], GWR (ang. *Grow When Required*) [75] oraz dynamiczne, samoorganizujące się sieci neuro-

nowe z jednowymiarowym sąsiedztwem topologicznym DSOM (ang. *Dynamic SOM*) [46] o strukturze łańcucha neuronów.

Rozdział 3 przestawia proponowane w ramach niniejszej pracy oryginalne narzędzie teoretyczne – uogólnioną samoorganizującą się sieć neuronową o ewoluującej, drzewopodobnej strukturze topologicznej. Narzędzie to jest istotnym rozwinięciem wspomnianej w rozdziale 2 dynamicznej, samoorganizującej się sieci z jednowymiarowym sąsiedztwem, w ramach którego udoskonalono mechanizmy usuwania oraz wstawiania nowych połączeń topologicznych pomiędzy neuronami (w celu poprawienia zdolności sieci do dzielenia swojej struktury na części i ponownego ich łączenia ze sobą) oraz mechanizmów automatycznego modyfikowania liczby neuronów sieci. Wprowadzone istotne uogólnienia przyczyniają się do znacznej poprawy efektywności proponowanego narzędzia w problemach grupowania danych zlokalizowanych w skupiskach o skomplikowanych kształtach, o różnych wielkościach i ze znacznie zróżnicowaną gęstością zawartych w nich danych, w tym danych zaszumionych.

W rozdziale 4 przeprowadzono test funkcjonowania proponowanej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze z wykorzystaniem syntetycznych i rzeczywistych zbiorów danych. W pierwszym przypadku, rozważono 10 zbiorów danych reprezentujących różne kategorie złożonych problemów grupowania danych, natomiast w drugim przypadku, wykorzystano 3 zbiory danych z repozytorium Uniwersytetu Kalifornijskiego w Irvine.

Rozdział 5 jest pierwszym z serii trzech rozdziałów, które przedstawiają test praktycznej użyteczności proponowanego narzędzia teoretycznego w problemach grupowania danych opisujących ekspresję genów. W rozdziale tym rozważane są dwa zbiory danych będących wynikiem analizy materiału genetycznego pobranego od pacjentów ze zdiagnozowaną chorobą nowotworową układu krwionośnego. Celem eksperymentów jest zbadanie skuteczności proponowanego podejścia, po pierwsze, w wykrywaniu różnych typów choroby (podział przypadków choroby na dwie lub trzy kategorie) i po drugie, w wykrywaniu grup genów pełniących podobne funkcje w przypadku rozważanej choroby.

W rozdziałach 6 i 7 rozważane są dwa analogiczne problemy grupowania danych będących rezultatem analiz materiału genetycznego pacjentów ze zdiagnozowaną chorobą nowotworową jelita grubego (rozdział 6) oraz układu chłonnego (rozdział 7). Po-

dobnie jak w rozdziale 5, celem eksperymentów jest badanie skuteczności samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze w problemach diagnozowania pacjentów zdrowych i chorych (rozdział 6) oraz wyodrębniania różnych podkategorii chorób nowotworowych (rozdział 7).

Rozdział 8 syntetycznie podsumowuje wyniki uzyskane w pracy, wyszczególniając najważniejsze oryginalne osiągnięcia autora oraz przedstawiając wnioski. Ponadto, praca zawiera wykaz literatury oraz dwa dodatki opisujące rzeczywiste zbiory danych wykorzystane w części eksperymentalnej rozprawy, odpowiednio, w rozdziale 4 (dodatek A) oraz w rozdziałach 5, 6 i 7 (dodatek B).

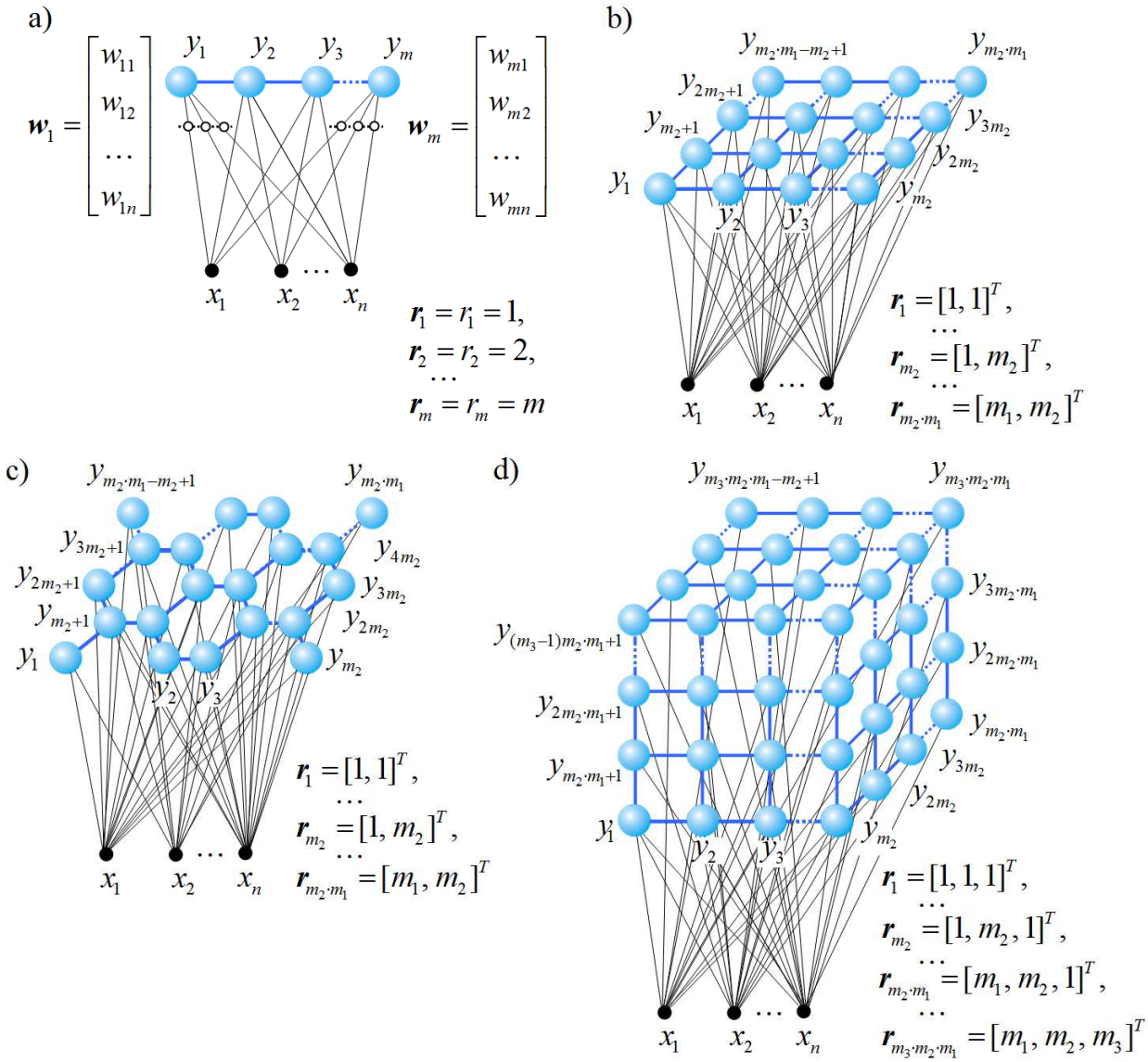
# 2. Samoorganizujące się sieci neuronowe i ich warianty w grupowaniu danych

Przedstawiony w tym rozdziale przegląd różnych wariantów samoorganizujących się sieci neuronowych (ang. *self-organizing neural networks*), mających zastosowanie w problemach grupowania danych, stanowi punkt wyjścia do rozważań nad proponowaną w kolejnym rozdziale uogólnioną samoorganizującą się siecią neuronową o ewoluującej, drzewopodobnej strukturze. W podrozdziale 2.1 syntetycznie omówiono konwencjonalne, samoorganizujące się mapy neuronów SOM (ang. *Self-Organizing Map*) [25, 57, 59, 66, 67, 68, 78, 86, 98, 105, 113, 125]. Są one podstawą do konstrukcji wszystkich narzędzi grupowania danych rozważanych w dalszej części pracy. Kolejne trzy podrozdziały obejmują różne, wybrane warianty sieci SOM, które wyposażono w dodatkowe narzędzia do dynamicznej modyfikacji struktury sieci w trakcie trwania procesu jej uczenia. Zdolność sieci do zmiany swojej struktury ma kluczowe znaczenie z punktu widzenia grupowania danych (szersza dyskusja na ten temat zostanie przedstawiona w dalszej części rozdziału). Podrozdział 2.2 obejmuje dwa podejścia: GSOM (ang. *Growing SOM*) [94] i GGN (ang. *Growing Grid Network*) [39], które umożliwiają sukcesywne rozbudowywanie struktury sieci poprzez dodawanie do niej nowych neuronów. Podrozdział 2.3 obejmuje sieci MIGSOM (ang. *Multilevel Interior Growing Self-organizing Maps*) [6] i IGG (ang. *Incremental Grid Growing*) [14], które oprócz własności wyżej wymienionych sieci, posiadają zdolność do podziału struktury na części. Z kolei, podrozdział 2.4 przedstawia pięć najbardziej zaawansowanych wariantów SOM: GNG (ang. *Growing Neural Gas*) [38], IGNG (ang. *Incremental Growing Neural Gas*) [92], GCS (ang. *Growing Cell Structures*) [37], GWR (ang. *Grow When Required*) [75] oraz DSOM (ang. *Dynamic SOM*) [46], zdolnych do zmiany rozmiaru swojej struktury poprzez dodawanie lub usuwanie neuronów oraz do jej podziału na części.

## 2.1. Konwencjonalne, samoorganizujące się mapy neuronów (SOM)

Samoorganizująca się mapa neuronów, zwana również mapą Kohonena (ang. *Kohonen's map*), jest jednowarstwową siecią neuronową o $n$ wejściach $x_i$ ($i = 1, 2, \ldots, n$) i $m$ wyjściach $y_j$ ($j = 1, 2, \ldots, m$), w której wyróżnia się dwa rodzaje połączeń neuro-

nowych (tak jak to pokazano na rys. 2.1): tzw. połączenia wagowe (oznaczone kolorem czarnym) oraz nie-fizyczne tzw. połączenia topologiczne (oznaczone kolorem niebieskim).




Rys. 2.1. Wybrane rodzaje struktur samoorganizujących się map neuronów: jednowymiarowa (łańcuch neuronów) (a), dwuwymiarowa (mapa neuronów) (b), dwuwymiarowa (plaster miodu) (c) oraz trójwymiarowa (sześcian) (d)

Połączenia wagowe przekazują sygnały podawane na wejścia sieci $x_i$ do $j$-tego neuronu za pomocą współczynników wagowych $w_{ji}$ (dla przykładu, wybrane współczynniki wagowe pierwszego i ostatniego ($m$-tego) neuronu oznaczono czarnymi kółeczkami na rys. 2.1a); sygnał wyjściowy $j$-tego neuronu jest określony następująco:

$$y_j = \sum_{i=1}^{n} w_{ji} x_i = \boldsymbol{w}_j^T \boldsymbol{x}, \qquad j = 1,2,...,m, \tag{2.1}$$

gdzie $\boldsymbol{x} \in \Re^n$, $\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$ oraz $\boldsymbol{w}_j \in \Re^n$, $\boldsymbol{w}_j = [w_{j1}, w_{j2}, ..., w_{jn}]^T$ są wektorami, odpowiednio, danych wejściowych i współczynników wagowych *j*-tego neuronu.

Z kolei, nie-fizyczne połączenia topologiczne (realizowane przy pomocy odpowiednich funkcji sąsiedztwa – patrz dalszy ciąg rozdziału) wiążą między sobą określone neurony w jedną strukturę topologiczną, która może przyjmować różnorodne, uporządkowane, przestrzenne formy geometryczne, np. łańcucha neuronów (rys. 2.1a), siatki neuronów (rys. 2.1b; najczęściej rozważany przypadek), plastra miodu (rys. 2.1c), formy sześciennej (rys. 2.1d), itp. Położenie *j*-tego neuronu w strukturze topologicznej określa tzw. wektor położenia $\boldsymbol{r}_j \in \Re^q$, $\boldsymbol{r}_j = [r_{j1}, r_{j2}, ..., r_{jq}]^T$, $r_{jh} \in \{1,2,...,m_h\}$, $h = 1,2,...,q$, gdzie $q$ oznacza wymiarowość struktury topologicznej, natomiast $m_h$ oznacza rozmiar struktury w *h*-tym wymiarze (dla najczęściej rozważanego przypadku mapy neuronów $q = 2$; $m_1$ i $m_2$ oznaczają, odpowiednio, jej szerokość i wysokość – patrz rys. 2.1b).

W sensie geometrycznym, wektory $\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$ i $\boldsymbol{w}_j = [w_{j1}, w_{j2}, ..., w_{jn}]^T$ określają współrzędne położenia, odpowiednio, próbki danych wejściowych i *j*-tego neuronu w *n*-wymiarowej przestrzeni danych, zwanej również przestrzenią wagową. Natomiast, wektor $\boldsymbol{r}_j = [r_{j1}, r_{j2}, ..., r_{jq}]^T$ określa współrzędne położenia *j*-tego neuronu w *q*-wymiarowej tzw. przestrzeni topologicznej.

***Proces uczenia samoorganizującej się sieci neuronowej***

Proces uczenia sieci neuronowej odbywa się iteracyjnie ($k$ oznacza numer iteracji, $k = 1,2,...,k_{\max}$, $k_{\max}$ jest maksymalną liczbą iteracji), przy czym w każdej pojedynczej iteracji podawany jest na wejścia sieci wektor danych uczących $\boldsymbol{x}_l = [x_{l1}, x_{l2}, ..., x_{ln}]^T$ ($l = 1,2,...,L$, gdzie $L$ jest liczbą wektorów uczących w zbiorze danych uczących), w celu przeliczenia jej odpowiedzi i adaptacji wektorów wagowych. Cykl $L$ iteracji, w których wszystkie wektory danych uczących zostaną podane na wejścia sieci nazwany jest epoką uczenia ($e$ oznacza numer epoki, $e = 1,2,...,e_{\max}$; $k_{\max} = L \cdot e_{\max}$). W ramach pojedynczej epoki uczenia, wektory danych uczących mogą być podawane na wejścia sieci w sposób porządkowy (według kolejności ich występowania w zbiorze danych uczących) lub losowy (każdy jest wybierany raz w losowej kolejności). Proces uczenia zo-

staje zakończony po przeliczeniu z góry założonej liczby iteracji $k_{\max}$ (lub liczby epok $e_{\max}$).

W ramach algorytmu uczenia WTM (ang. *Winner Takes Most*) [68], w *k*-tej iteracji procesu uczenia, po podaniu na wejścia sieci wektora danych uczących $\boldsymbol{x}_l(k)$ ($l \in \{1,2,...,L\}$, $k = 1,2,...,k_{\max}$), wagi sieci neuronowej podlegają adaptacji zgodnie z regułą Kohonena [66] (patrz również ilustracja tego procesu na rys. 2.2):

$$\boldsymbol{w}_j(k+1) = \boldsymbol{w}_j(k) + \eta(k) \cdot S(j, j_{\boldsymbol{x}}, k) \cdot [\boldsymbol{x}_l(k) - \boldsymbol{w}_j(k)], \tag{2.2}$$

gdzie $\eta(k)$ jest współczynnikiem uczenia (stałym dla danej epoki i malejącym po zakończeniu każdej kolejnej epoki według założonego z góry scenariusza, np. liniowo), $S(j, j_{\boldsymbol{x}}, k)$ jest tzw. funkcją sąsiedztwa topologicznego (szerzej omówioną w dalszej części rozdziału), a $j_{\boldsymbol{x}}$ jest numerem neuronu zwyciężającego we współzawodnictwie neuronów, tzn. zlokalizowanego najbliżej wektora danych uczących $\boldsymbol{x}_l(k)$ w *n*-wymiarowej przestrzeni danych:

$$j_{\boldsymbol{x}} = \arg \min_{j=1,2,...,m} d(\boldsymbol{x}_l(k), \boldsymbol{w}_j(k)), \tag{2.3}$$

przy czym $d(\boldsymbol{x}, \boldsymbol{w})$ jest miarą odległości pomiędzy wektorem wagowym $\boldsymbol{w}$, a wektorem danych wejściowych $\boldsymbol{x}$ i może przyjmować różne postacie, a w szczególności:

- postać euklidesowej miary odległości (ang. *Euclidean distance*):

$$d(\boldsymbol{x}, \boldsymbol{w}) = \sqrt{\sum_{i=1}^{n} (x_i - w_i)^2}, \tag{2.4}$$

- postać tzw. miary odległości Manhattan (ang. *Manhattan distance*):

$$d(\boldsymbol{x}, \boldsymbol{w}) = \sum_{i=1}^{n} |x_i - w_i|, \tag{2.5}$$

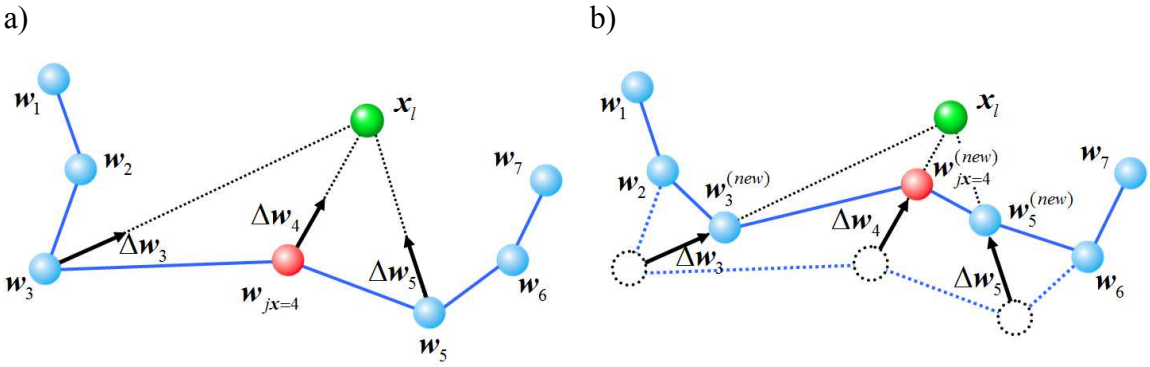
- postać miary odległości maksimum (ang. *maximum distance*) zwanej również miarą Czebyszewa:

$$d(\boldsymbol{x}, \boldsymbol{w}) = \max_{1 \le i \le n} |x_i - w_i|, \tag{2.6}$$

- postać uogólnionej miary odległości Minkowskiego (ang. *Minkowski distance*):

$$d(\boldsymbol{x}, \boldsymbol{w}) = \left[ \sum_{i=1}^{n} (x_i - w_i)^r \right]^{\frac{1}{r}}, \quad r > 0; \tag{2.7}$$

w szczególnych przypadkach wartości *r*, miara Minkowskiego sprowadza się do wyżej wymienionych miar, tzn. dla $r = 1$ – do miary Manhattan, dla $r = 2$ – do miary euklidesowej oraz dla $r = \infty$ – do miary maksimum (Czebyszewa).




Rys. 2.2. Adaptacja wag neuronu zwyciężającego (oznaczonego kolorem czerwonym) oraz jego bezpośrednich sąsiadów, w odpowiedzi na podanie na wejście sieci wektora danych uczących (oznaczonego kolorem zielonym); stan sieci przed aktualizacją wektorów wagowych (a) oraz po ich aktualizacji (b)

Funkcja sąsiedztwa topologicznego $S(j, j_x, k)$, występująca w regule (2.2), pełni kluczową rolę w procesie uczenia. Funkcja ta decyduje, które z neuronów sieci i w jakim stopniu adaptują swoje wektory wagowe po podaniu na wejścia sieci wektora danych uczących $\boldsymbol{x}_l(k)$, w $k$-tej iteracji tego procesu. W zależności od przyjętego modelu sąsiedztwa topologicznego neuronów, funkcja sąsiedztwa przyjmuje różne postacie, a w szczególności:

- dla tzw. prostokątnego sąsiedztwa topologicznego:

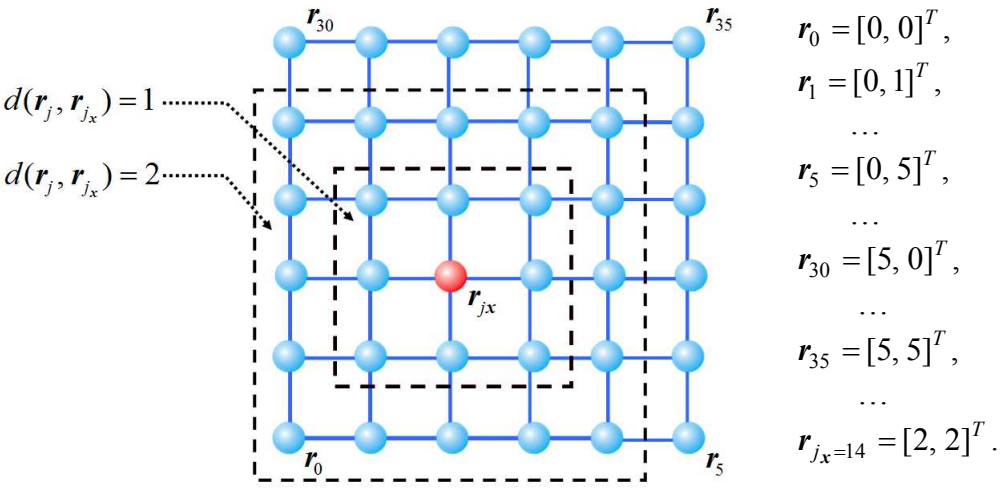
$$S(j, j_x, k) = \begin{cases} 1, & dla \quad d(\boldsymbol{r}_j(k), \boldsymbol{r}_{j_x}(k)) \le \lambda(k), \\ 0, & dla \quad d(\boldsymbol{r}_j(k), \boldsymbol{r}_{j_x}(k)) > \lambda(k), \end{cases} \tag{2.8}$$

- dla tzw. gaussowskiego sąsiedztwa topologicznego:

$$S(j, j_x, k) = e^{\frac{d^2(\boldsymbol{r}_j, \boldsymbol{r}_{j_x})}{2\lambda^2(k)}}, \tag{2.9}$$

gdzie $\lambda(k)$ jest tzw. promieniem sąsiedztwa topologicznego (stałym lub malejącym w trakcie trwania procesu uczenia w analogiczny sposób jak współczynnik uczenia $\eta(k)$), natomiast $d(\boldsymbol{r}_j, \boldsymbol{r}_{j_x})$ jest miarą odległości maksimum (miarą Czebyszewa) (2.6).

Na rys. 2.3 przedstawiono przykładową strukturę topologiczną sieci neuronowej – mapę neuronów, z wyróżnionym (kolorem czerwonym) neuronem zwyciężającym we współzawodnictwie neuronów w ramach algorytmu WTM oraz neurony znajdujące się zasięgu jego prostokątnego sąsiedztwa topologicznego (określonego formułą (2.8)) o promieniu $\lambda = 1$ i $\lambda = 2$ (zasięg sąsiedztwa został oznaczony przerywaną linią).



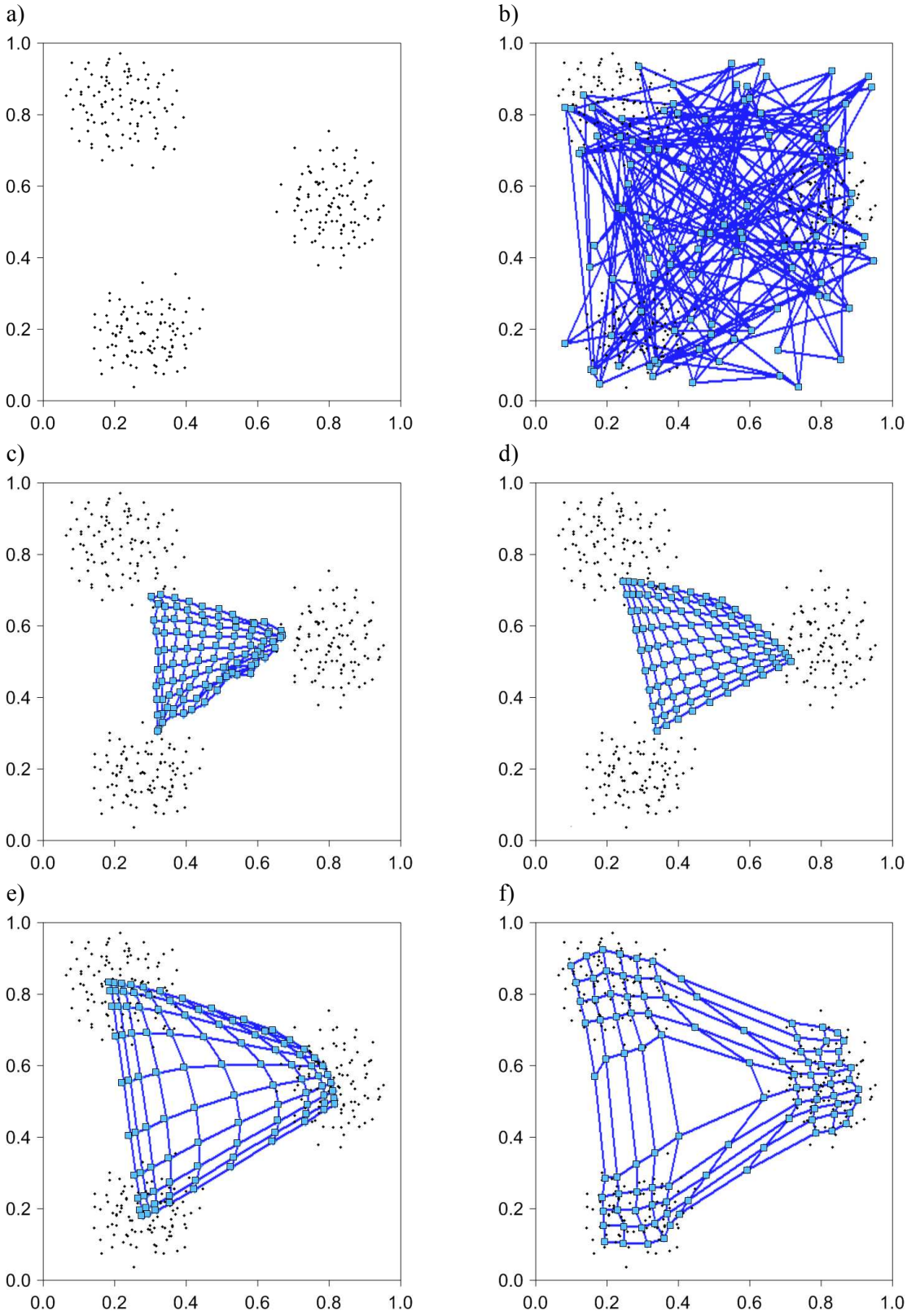

Rys. 2.3. Ilustracja zasięgu prostokątnego sąsiedztwa topologicznego wokół neuronu zwyciężającego we współzawodnictwie neuronów (oznaczonego kolorem czerwonym) o promieniu $\lambda = 1$ (mniejszy zasięg) i $\lambda = 2$ (większy zasięg)

W celu ilustracji procesu uczenia sieci neuronowej rozważono przykładowy zbiór danych zawierający trzy wyraźnie odseparowane od siebie skupiska danych (rys. 2.4a) oraz sieć neuronową zawierającą $m$ = 100 neuronów o strukturze topologicznej z dwuwymiarowym sąsiedztwem (mapa neuronów). Szczegółowe parametry sieci neuronowej oraz parametry procesu jej uczenia przedstawia tabela 2.1. Na rys. 2.4 przedstawiono rozłożenie struktury sieci neuronowej w przestrzeni danych uczących w wybranych etapach (epokach) procesu uczenia.

Tabela 2.1. Parametry samoorganizującej się sieci neuronowej wykorzystanej do analizy zbioru danych z rys. 2.4a oraz procesu jej uczenia (wybrane etapy procesu przedstawiają rys. 2.4b–f)

| Nazwa parametru | Wartość (rodzaj) parametru |
|---|---|
| liczba neuronów sieci | $m = 100$ (siatka 10 × 10) |
| maksymalna liczba epok uczenia | $e_{\max} = 10\,000$ |
| sposób wyznaczania neuronu zwyciężającego (o numerze $j_x$) | zależność (2.3) z euklidesową miarą odległości (2.4) |
| współczynnik uczenia sieci[1)] w (2.2) | $\eta(e) = 0.5 \div 0.001$ |
| funkcja sąsiedztwa topologicznego $S(j, j_x, k)$ w (2.2) | funkcja sąsiedztwa gaussowskiego (2.9) z miarą odległości Czebyszewa (2.6) |
| promień sąsiedztwa topologicznego[1)] w (2.9) | $\lambda(e) = 3 \div 1$ |

[1)] parametr malejący liniowo po zakończeniu kolejnych epok uczenia $e$; stały dla danej epoki.




Rys. 2.4. Ilustracja procesu uczenia sieci SOM: zbiór danych wejściowych (a) i rozmieszczenie wektorów wagowych sieci neuronowej w przestrzeni danych uczących na początku procesu uczenia ($e$ = 0) (b), w epoce $e$ = 10 (c), $e$ = 500 (d), $e$ = 5 000 (e) oraz po zakończeniu procesu uczenia ($e$ = 10 000) (f)

W początkowej fazie uczenia (rys. 2.4b), topologia powiązań miedzy neuronami jest nieregularna i chaotyczna, co wynika z inicjowania wektorów wagowych sieci $\boldsymbol{w}_j(k=0)$ ($j=1,2,...,m=100$) wartościami losowanymi z przedziału [0, 1]. W trakcie uczenia (rys. 2.4c–f) sieć dąży do takiego rozkładu neuronów w przestrzeni danych uczących, aby topologia sieci była jak najbardziej uporządkowana (do postaci siatki neuronów) oraz dopasowana do rozmieszczenia skupisk danych w rozważanym zbiorze najlepiej jak to jest możliwe (końcowy rezultat uczenia widoczny jest rys. 2.4f).

***Tryby pracy samoorganizującej się sieci neuronowej***

Po zakończeniu procesu uczenia, SOM funkcjonuje w dwóch trybach pracy:

- praca w trybie tzw. książki kodowej (ang. *code book*), w którym sieć odwzorowuje wektor danych wejściowych $\boldsymbol{x}=[x_1,x_2,...,x_n]^T$ na wektor wagowy $\boldsymbol{w}_{j_{\boldsymbol{x}}}=[w_{j_{\boldsymbol{x}}1},w_{j_{\boldsymbol{x}}2},...,w_{j_{\boldsymbol{x}},n}]^T$ neuronu o numerze $j_{\boldsymbol{x}}$ określonym formułą (2.3),
- praca w trybie tzw. mapy cech (ang. *feature map*), w którym sieć odwzorowuje wektor danych wejściowych $\boldsymbol{x}=[x_1,x_2,...,x_n]^T$ na wektor położenia $\boldsymbol{r}_{j_{\boldsymbol{x}}}=[r_{j_{\boldsymbol{x}}1},r_{j_{\boldsymbol{x}}2},...,r_{j_{\boldsymbol{x}},q}]^T$ neuronu o numerze $j_{\boldsymbol{x}}$ określonym formułą (2.3).

Podstawowym zadaniem sieci SOM funkcjonującej w trybie książki kodowej jest tzw. kwantyzacja wektorowa (ang. *vector quantization*) [68], czyli nieliniowe i nieodwracalne odwzorowanie statyczne zbioru $L$ wektorów danych wejściowych $\boldsymbol{X}_{L\times n}=\{\boldsymbol{x}_l^T\}_{l=1}^L=\{[x_{l1},x_{l2},...,x_{ln}]\}_{l=1}^L$ do zbioru $m$ wektorów wagowych $\boldsymbol{W}_{m\times n}=\{\boldsymbol{w}_j^T\}_{j=1}^m=\{[w_{j1},w_{j2},...,w_{jn}]\}_{j=1}^m$ ($m<L$) (zbiór taki nazywany jest książką kodową). W sensie geometrycznym, odwzorowanie dzieli przestrzeń danych wejściowych na $m$ obszarów oddziaływań poszczególnych neuronów, ograniczonych tzw. wielobokami Voronoia.

Z kolei, sieć SOM funkcjonująca w trybie mapy cech umożliwia wizualizację relacji zachodzących pomiędzy danymi w formie tzw. U-macierzy (ang. *Unified distance matrix*) [68, 114], utworzonej z zestawu $m$ liczb $U_j$ ($j=1,2,...,m$):

$$U_j=\sum_{h\in S_j} d(\boldsymbol{w}_h,\boldsymbol{w}_j), \tag{2.10}$$

gdzie $S_j$ jest zbiorem numerów neuronów bezpośrednio sąsiadujących z neuronem o numerze $j$ w topologicznej strukturze sieci, natomiast $d(\boldsymbol{w}_h, \boldsymbol{w}_j)$ oznacza miarę odległości pomiędzy wektorami wagowymi $\boldsymbol{w}_h$ i $\boldsymbol{w}_j$ (np. miarę euklidesową (2.4)). Wartości liczb $U_j$ mogą być prezentowane w postaci np. wykresu słupkowego – histogramu (w przypadku sieci z jednowymiarowym sąsiedztwem topologicznym), mapy kolorowych punktów rozmieszczonych na płaszczyźnie (w przypadku sieci z dwuwymiarowym sąsiedztwem topologicznym), itp., przy czym położenie słupków w histogramie lub punktów na mapie określone jest przez wektory położenia $\boldsymbol{r}_j$ neuronów.

Na rys. 2.5 przedstawiono ilustrację funkcjonowania sieci neuronowej z rys. 2.4f w obu trybach pracy. Rys. 2.5a przedstawia zbiór danych z zaznaczonym kolorem zielonym wektorem danych $\boldsymbol{x}$, podanym na wejścia sieci, a rys. 2.5b – topologiczną strukturę SOM po zakończeniu procesu jej uczenia z zaznaczonym kolorem czerwonym wektorem wagowym $\boldsymbol{w}_{j_x}$, który odwzorowuje wektor $\boldsymbol{x}$ na podstawie reguły (2.3). Z kolei, rys. 2.5c przedstawia podział przestrzeni danych wejściowych na obszary aktywności poszczególnych neuronów (granice między tymi obszarami oznaczono kolorem zielonym) ograniczone wielobokami Voronoia oraz rys. 2.5d – U-macierz sieci neuronowej z zaznaczonym na niej kolorem czerwonym położeniem neuronu o numerze $j_x$ i widocznymi granicami (w postaci punktów o ciemniejszym odcieniu koloru niebieskiego) pomiędzy trzema skupiskami danych.

Konwencjonalna SOM ma ograniczoną zdolność wykrywania skupisk w zbiorach danych, co podkreśla Bezdek w pracy [10]:

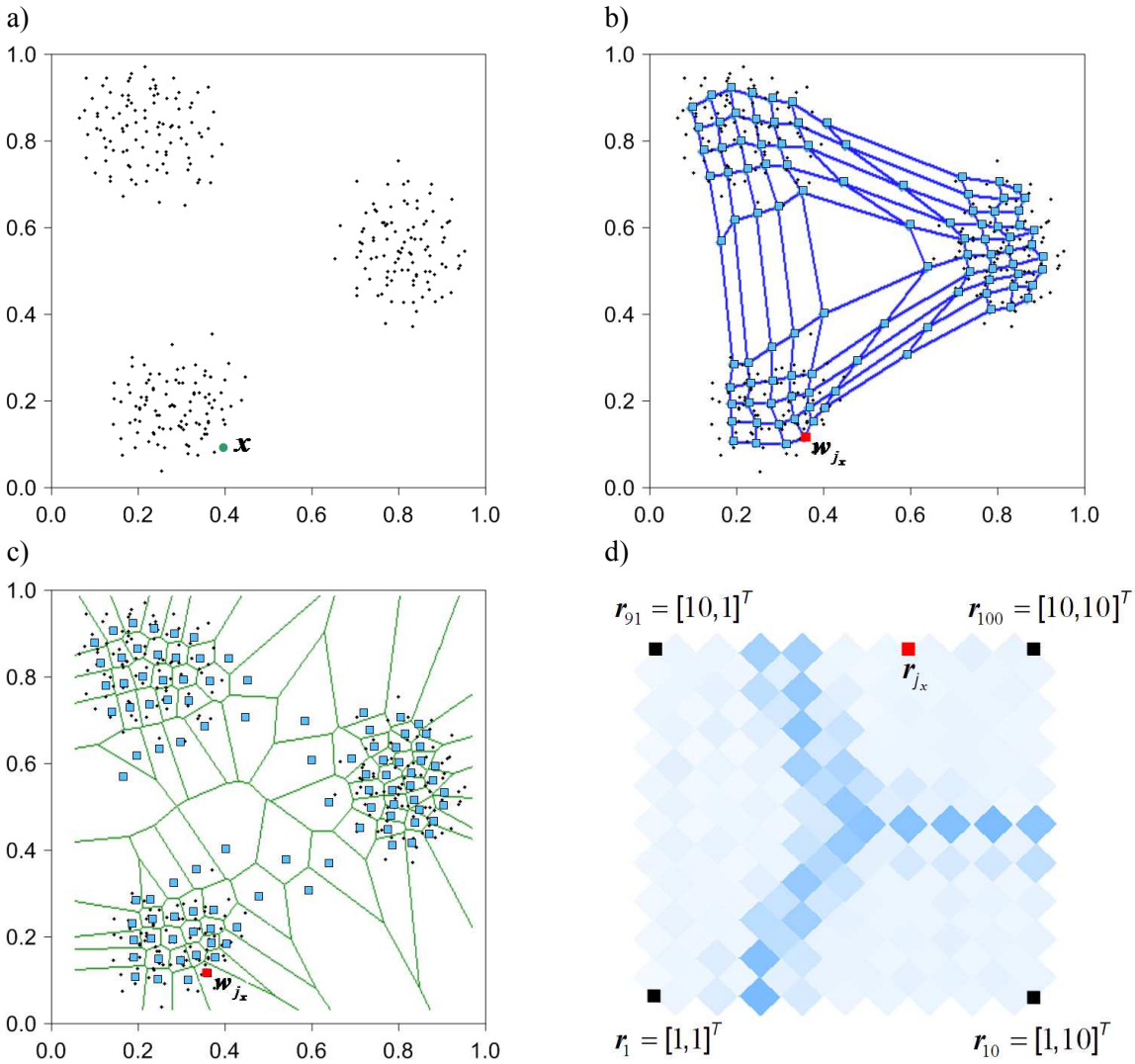
> *"... Kohonen's self-organizing feature mapping (SOFM), which is not a clustering method, but which often lends ideas to clustering algorithms."*

oraz

> *"SOFM, on the other hand, attempts to find topological structure hidden in the data and display it in one or two dimensions."*

Sieć nie jest zdolna do automatycznego wyznaczania liczby skupisk danych oraz granic pomiędzy nimi. W przypadku funkcjonowania SOM w trybie mapy cech, możliwe jest jedynie subiektywne oszacowanie tej liczby oraz przybliżone oznaczenie granic pomiędzy skupiskami na podstawie wizualnej analizy U-macierzy. Jednak analiza taka jest możliwa tylko dla zbiorów danych o stosunkowo małej złożoności (głównie zawierają-

cych skupiska dobrze odseparowane od siebie o charakterze objętościowym). Konwencjonalna SOM pracująca w trybie książki kodowej jest nieprzydatna z punktu widzenia grupowania danych (ma natomiast inne zastosowania, np. do kompresji danych).




Rys. 2.5. Ilustracja trybów pracy sieci SOM: przykładowy zbiór danych wejściowych (a), rozmieszczenie wektorów wagowych w przestrzeni danych (b), praca w trybie książki kodowej, czyli podział przestrzeni danych na obszary oddziaływań wektorów wagowych, ograniczonych wielobokami Voronoia (c) oraz praca w trybie mapy cech, na której widoczny jest podział na trzy strefy reprezentujące odpowiednio trzy skupiska danych (d)

Powyższe mankamenty konwencjonalnej sieci SOM mogą być wyeliminowane poprzez wprowadzenie do procesu uczenia sieci narzędzi dynamicznie modyfikujących jej strukturę topologiczną, tzn.: mechanizmów dodawania lub usuwania neuronów oraz wstawiania lub usuwania połączeń topologicznych pomiędzy neuronami. Mechanizmy

te umożliwią automatyczne dostosowanie kształtu struktury sieci do rozkładu gęstości danych w lokalnych obszarach przestrzeni danych wejściowych, celem wykrycia występujących w nich skupisk danych. Powyższa ogólna idea sprowadza się do wielokrotnej aktywacji – w trakcie trwania procesu uczenia sieci – następujących operacji:

- redukcji rozmiaru struktury sieci poprzez usuwanie z niej neuronów zlokalizowanych w obszarach o małej gęstości danych (za pomocą mechanizmu usuwania neuronów),
- powiększania struktury sieci poprzez dodawanie do niej nowych neuronów i lokowanie ich w obszarach o dużej gęstości danych (z wykorzystaniem mechanizmu wstawiania nowych neuronów),
- dzielenia struktury na mniejsze podstruktury poprzez usuwanie połączeń topologicznych pomiędzy neuronami zlokalizowanymi w obszarach o mniejszej gęstości danych (z wykorzystaniem mechanizmu usuwania połączeń topologicznych),
- ponownego łączenia wybranych podstruktur sieci poprzez wstawianie nowych połączeń topologicznych pomiędzy neuronami zlokalizowanymi w obszarach o większej gęstości danych (za pomocą mechanizmu wstawiania nowych połączeń topologicznych).

W rezultacie, sieć neuronowa nabiera zdolności do automatycznego wykrywania liczby skupisk danych oraz generowania ich wielopunktowych prototypów. Liczba skupisk danych jest równa liczbie podstruktur sieci neuronowej, powstałych w wyniku podziału pierwotnej struktury. Pojedyncza podstruktura sieci reprezentuje jeden wielopunktowy prototyp pojedynczego skupiska danych, przy czym wektory wagowe neuronów należących do tej podstruktury reprezentują punkty prototypu, a liczba tych punktów jest „dostrajana" w sposób automatyczny.

Po zakończeniu procesu uczenia, praca sieci neuronowej w trybie mapy cech traci na znaczeniu, gdyż w wyniku powyższych modyfikacji jej topologiczna struktura jest sprowadzana do nieregularnej formy posiadającej ograniczone możliwości wizualnej prezentacji relacji pomiędzy danymi. Natomiast kluczowego znaczenia nabiera praca sieci w trybie książki kodowej, w którym określa się przynależność próbki danych wejściowych $\boldsymbol{x}$ (po podaniu jej na wejścia sieci) do takiej grupy danych, która jest reprezentowana przez tzw. najbliższy jej wielopunktowy prototyp (ang. *nearest multi-point prototype* [12, 13]), tzn. prototyp zawierający neuron o numerze $j_{\boldsymbol{x}} \in \{1,2,...,m\}$ określonym formułą (2.3). W przypadku wariantu ostrego grupowania danych (wspomnianym

we Wprowadzeniu; patrz również rys. 1.1), granice decyzyjne dla tej grupy danych są równoważne zewnętrznym granicom obszaru przestrzeni danych, który powstał po scaleniu obszarów Voronoia wszystkich neuronów należących do powyższego prototypu. W przypadku wariantu rozmytego grupowania danych, stopień przynależności próbki danych wejściowych $\boldsymbol{x}$ do określonej grupy może być wyznaczany następująco [124]:

$$\mu(\boldsymbol{x}) = \frac{1}{1 + d(\boldsymbol{x}, \boldsymbol{w}_{j_x})}, \tag{2.11}$$

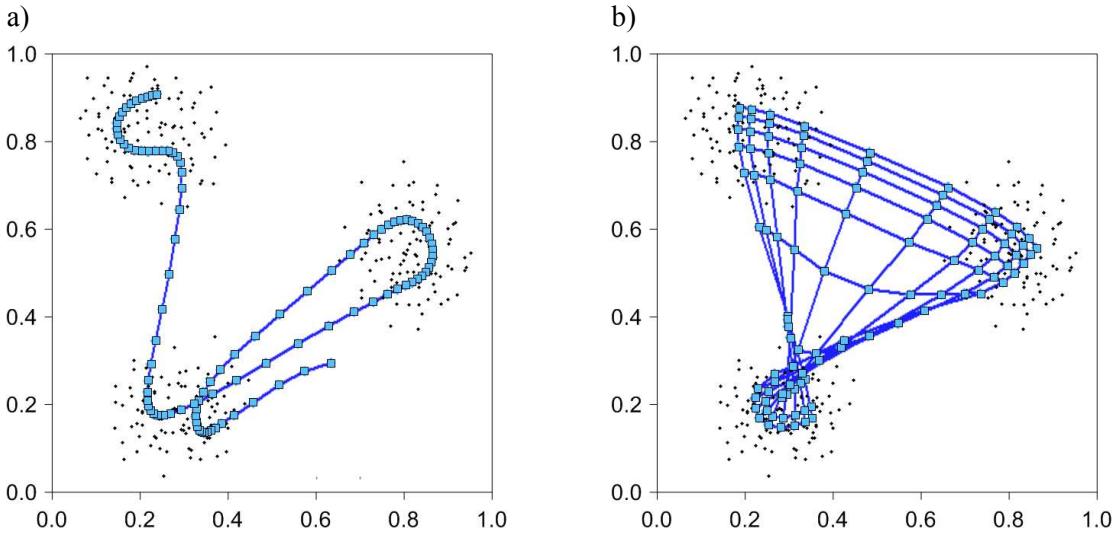
gdzie $d(\boldsymbol{x}, \boldsymbol{w}_{j_x})$ jest miarą odległości (np. miarą euklidesową (2.4)) pomiędzy wektorem $\boldsymbol{x}$, a wektorem wagowym $\boldsymbol{w}_{j_x}$ neuronu o numerze $j_x$. W następnej części tego rozdziału zostaną przedstawione podejścia, które, w mocno ograniczonym stopniu, podejmują próbę implementacji powyższych idei.

## 2.2. Samoorganizujące się sieci neuronowe z mechanizmami powiększania rozmiaru struktury topologicznej

W pierwszej kolejności zostaną przedstawione dwie modyfikacje konwencjonalnej sieci SOM, które umożliwiają powiększanie rozmiaru topologicznej struktury sieci neuronowej w trakcie trwania procesu jej uczenia. Modyfikacje polegają na wprowadzeniu do klasycznego algorytmu uczenia WTM mechanizmów dodawania nowych neuronów do struktury sieci. Aktywacja tych mechanizmów odbywa się według scenariusza zakładającego stopniowe powiększanie rozmiaru struktury topologicznej SOM, rozpoczynając od struktury z niewielką liczbą neuronów (np. 2 lub 4 neurony w przypadku sieci, odpowiednio, z jednowymiarowym lub dwuwymiarowym sąsiedztwem topologicznym), a kończąc, gdy struktura sieci osiągnie założoną z góry liczbę neuronów.

Motywacją do wprowadzenia takich modyfikacji była potrzeba ograniczenia złożoności obliczeniowej algorytmu uczenia w pierwszych epokach procesu uczenia [94] oraz potrzeba ograniczenia niekorzystnego rozłożenia struktury SOM w przestrzeni danych wejściowych [46], np. zjawiska wielokrotnego przechodzenia łańcucha neuronów przez to samo skupisko danych (w przypadku SOM z jednowymiarowym sąsiedztwem topologicznym; patrz rys. 2.6a) lub zjawiska „skręcania się” topologicznej struktury sieci (w przypadku SOM z dwuwymiarowym sąsiedztwem topologicznym; patrz rys. 2.6b). Złożoność obliczeniowa algorytmu uczenia maleje, jeżeli m.in. w procesie adaptacji wektorów wagowych uczestniczy mniejsza liczba neuronów. Liczbę tę można

ograniczyć poprzez zmniejszenie promienia sąsiedztwa topologicznego $\lambda$ w (2.8) i (2.9) lub poprzez zmniejszenie całkowitej liczby neuronów sieci. Z kolei, opisane wyżej przypadki niekorzystnego rozmieszczenia struktury sieci w przestrzeni danych wejściowych mogą być ograniczone poprzez zwiększenie promienia sąsiedztwa $\lambda$ (w taki sposób, aby obejmowało większość lub nawet wszystkie neurony sieci, niezależnie od położenia neuronu zwyciężającego w topologicznej strukturze sieci) lub, analogicznie jak wyżej, poprzez zmniejszenie liczby neuronów. Zatem, aby osiągnąć oba założone cele, należy ograniczyć liczbę neuronów sieci w początkowej fazie jej uczenia, a następnie umożliwić sieci stopniowe powiększanie jej struktury poprzez dodawanie nowych neuronów. Taką strategię uczenia SOM wykorzystują podejścia przedstawione w tym podrozdziale.

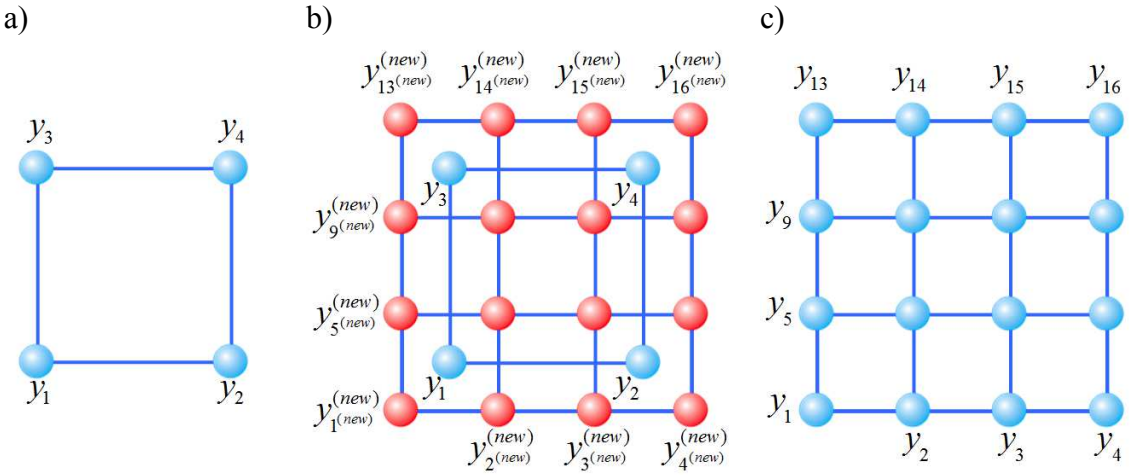



Rys. 2.6. Ilustracja niekorzystnego zjawiska „skręcania się" topologicznej struktury SOM w przestrzeni danych z rys. 2.4a: rozmieszczenie wektorów wagowych SOM z jednowymiarowym (a) i dwuwymiarowym (b) sąsiedztwem topologicznym

### 2.2.1. Sieci typu GSOM (ang. *Growing SOM*) [94]

Jedna z pierwszych propozycji mechanizmu wstawiania nowych neuronów, umożliwiającego powiększanie struktury SOM z dwuwymiarowym sąsiedztwem topologicznym, została przedstawiona w pracy [94]. Na początku procesu uczenia, inicjowana jest sieć neuronowa zawierająca 4 lub 9 neuronów (mapa $m = m_1 \times m_2$ neuronów). Po przeliczeniu pewnej, zakładanej z góry, liczby iteracji procesu uczenia i ustabilizowaniu roz-

łożenia struktury sieci w przestrzeni danych uczących, generowana jest nowa struktura sieci z czterokrotnie większą liczbą neuronów, tzn. dwukrotnie większą liczbą wierszy i kolumn mapy (mapa $m^{(new)} = 4m = 2m_1 \times 2m_2$ neuronów). Ilustracja tej operacji została przedstawiona na rys. 2.7.




Rys. 2.7. Ilustracja operacji powiększania topologicznej struktury GSOM [94]: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czerwonym oznaczono nowe neurony)

Każdy $j$-ty neuron ( $j = 1,2,...,m$ ), o wektorze wagowym $\boldsymbol{w}_j$, zlokalizowany w topologicznej strukturze pierwotnej sieci neuronowej we współrzędnych $\boldsymbol{r}_j = [r_{j1}, r_{j2}]^T$ ( $r_{j1} \in \{1,2,...,m_1\}$ ; $r_{j2} \in \{1,2,...,m_2\}$ ), jest zastępowany czterema nowymi neuronami o numerach, odpowiednio, $j_1^{(new)}$, $j_2^{(new)}$, $j_3^{(new)}$ i $j_4^{(new)}$, zlokalizowanymi w strukturze topologicznej nowej sieci neuronowej we współrzędnych, odpowiednio, $\boldsymbol{r}_{j_1^{(new)}} = [2r_{j1} - 1, 2r_{j2} - 1]^T$, $\boldsymbol{r}_{j_2^{(new)}} = [2r_{j1} - 1, 2r_{j2}]^T$, $\boldsymbol{r}_{j_3^{(new)}} = [2r_{j1}, 2r_{j2} - 1]^T$ i $\boldsymbol{r}_{j_4^{(new)}} = [2r_{j1}, 2r_{j2}]^T$. Ich wektory wagowe są określone następująco:

$$
\begin{aligned}
\boldsymbol{w}_{j_1^{(new)}}^{(new)} &= 1/16 (9\boldsymbol{w}_j + 3\boldsymbol{w}_{j_2} + 3\boldsymbol{w}_{j3} + \boldsymbol{w}_{j_4}), \\
\boldsymbol{w}_{j_2^{(new)}}^{(new)} &= 1/16 (3\boldsymbol{w}_j + 9\boldsymbol{w}_{j_2} + \boldsymbol{w}_{j3} + 3\boldsymbol{w}_{j_4}), \\
\boldsymbol{w}_{j_3^{(new)}}^{(new)} &= 1/16 (3\boldsymbol{w}_j + \boldsymbol{w}_{j_2} + 9\boldsymbol{w}_{j3} + 3\boldsymbol{w}_{j_4}), \\
\boldsymbol{w}_{j_4^{(new)}}^{(new)} &= 1/16 (\boldsymbol{w}_j + 3\boldsymbol{w}_{j_2} + 3\boldsymbol{w}_{j3} + 9\boldsymbol{w}_{j_4}),
\end{aligned}
\tag{2.12}
$$

gdzie $\boldsymbol{w}_{j_2}$, $\boldsymbol{w}_{j_3}$ i $\boldsymbol{w}_{j_4}$ są wektorami wagowymi neuronów bezpośrednio sąsiadujących z *j*-tym neuronem w topologicznej strukturze pierwotnej sieci neuronowej, tzn. położonych we współrzędnych, odpowiednio, $\boldsymbol{r}_{j_2} = [r_{j1} + 1, r_{j2}]^T$, $\boldsymbol{r}_{j_3} = [r_{j1}, r_{j2} + 1]^T$ i $\boldsymbol{r}_{j_4} = [r_{j1} + 1, r_{j2} + 1]^T$ (w przypadku skrajnych neuronów nie posiadających neuronów sąsiednich w lokalizacjach określonych wektorami $\boldsymbol{r}_{j2}$, $\boldsymbol{r}_{j3}$, lub $\boldsymbol{r}_{j4}$, w formule (2.12) pomija się wektory wagowe, odpowiednio, $\boldsymbol{w}_{j_2}$, $\boldsymbol{w}_{j_3}$ lub $\boldsymbol{w}_{j_4}$). Proces powiększania topologicznej struktury sieci może być wielokrotnie powtarzany aż do osiągnięcia zakładanej liczby neuronów.

### 2.2.2. Sieci typu GGN (ang. *Growing Grid Network*) [39]

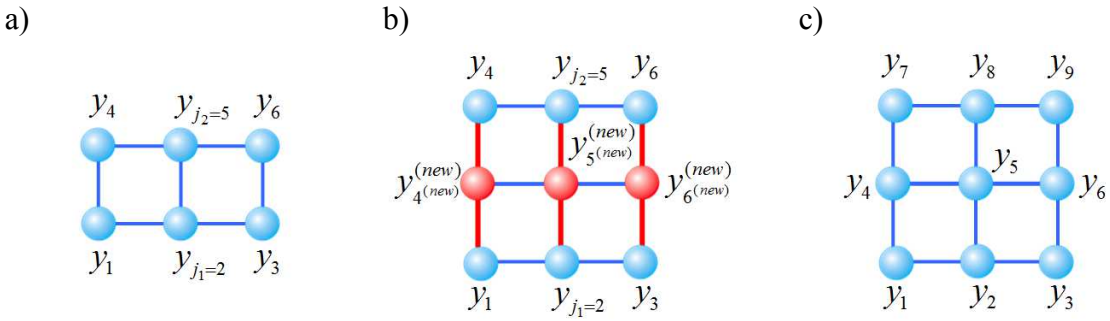
W pracy [39] zaproponowano sieć GGN (ang. *Growing Grid Network*) z bardzo podobnym do powyższego mechanizmem powiększania jej topologicznej struktury. Tym razem, po przeliczeniu pewnej, zakładanej z góry, liczby iteracji procesu uczenia i ustabilizowaniu rozłożenia struktury sieci w przestrzeni danych uczących, wyznaczane są dwa neurony: najbardziej aktywny neuron w całej sieci (tzn. najczęściej zwyciężający w ramach algorytmu WTM) oznaczony numerem $j_1$ oraz jeden z jego bezpośrednich sąsiadów w topologicznej strukturze sieci (który jest najbardziej oddalony od $j_1$-tego neuronu w przestrzeni danych wejściowych) oznaczony numerem $j_2$. Następnie, w przypadku, gdy oba wyznaczone neurony są umieszczone w jednej kolumnie topologicznej struktury sieci, wstawiany jest nowy wiersz neuronów, tak jak to jest przedstawione na rys. 2.8. Analogicznie, wstawiana jest nowa kolumna neuronów, w przypadku gdy neurony są umieszczone w jednym wierszu topologicznej struktury sieci, tak jak to pokazuje rys. 2.9.

Wektor wagowy $\boldsymbol{w}_{j^{(new)}}^{(new)}$ nowego neuronu ($j^{(new)}$ oznacza jego nowy numer), wstawionego pomiędzy dwa neurony o numerach $j_1$ i $j_2$, wyznaczany jest następująco:

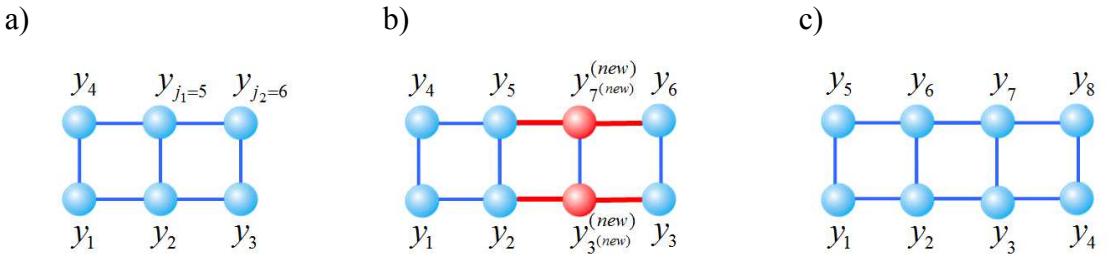
$$\boldsymbol{w}_{j^{(new)}}^{(new)} = \frac{\boldsymbol{w}_{j_1} + \boldsymbol{w}_{j_2}}{2}. \tag{2.13}$$

Wektory wagowe pozostałych neuronów w nowym wierszu (kolumnie) wyznaczane są analogicznie. Proces dodawania nowych neuronów zostaje zakończony, gdy liczba neu-

ronów osiągnie zakładaną z góry wartość lub, gdy liczba zwycięstw każdego z neuronów spadnie poniżej zakładanej z góry wartości.




Rys. 2.8. Ilustracja operacji wstawiania nowego wiersza neuronów do topologicznej struktury sieci GGN [39]: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czerwonym oznaczono nowe neurony i nowe połączenia topologiczne)




Rys. 2.9. Ilustracja operacji wstawiania nowej kolumny neuronów do topologicznej struktury sieci GGN [39]: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czerwonym oznaczono nowe neurony i nowe połączenia topologiczne)

## 2.3. Samoorganizujące się sieci neuronowe z mechanizmami powiększania rozmiaru struktury topologicznej i jej podziału na niezależne części

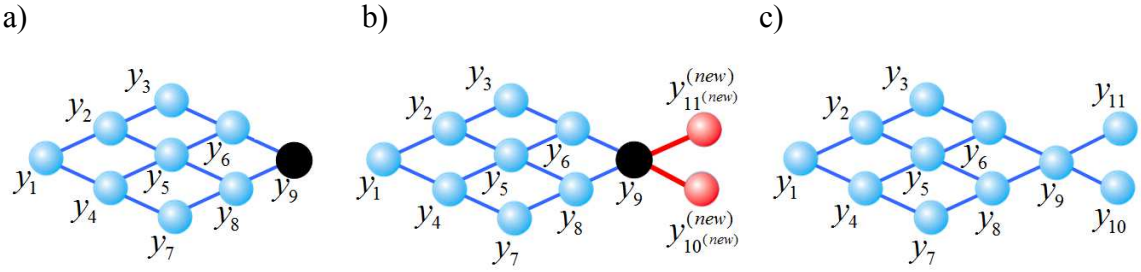
Przedstawione w poprzednim podrozdziale koncepcje sieci SOM o dynamicznie powiększającej się strukturze topologicznej w trakcie trwania procesu uczenia, należy uznać za zalążek szerszej idei samoorganizujących się sieci neuronowych (niekoniecznie map neuronowych) o zmiennej strukturze topologicznej. Tym razem zostaną przedstawione kolejne dwie modyfikacje konwencjonalnej SOM, zdolne nie tylko do powiększania swojej struktury topologicznej, ale również do jej podziału na mniejsze części (podstruktury), które w dalszych etapach procesu uczenia mogą ponownie dzielić się

na jeszcze mniejsze części. Jak wspomniano na wstępie tego rozdziału, zdolność sieci do podziału swojej struktury ma kluczowe znaczenie z punktu widzenia grupowania danych.

### 2.3.1. Sieci typu MIGSOM (ang. *Multilevel Interior Growing Self-organizing Maps*) [6]

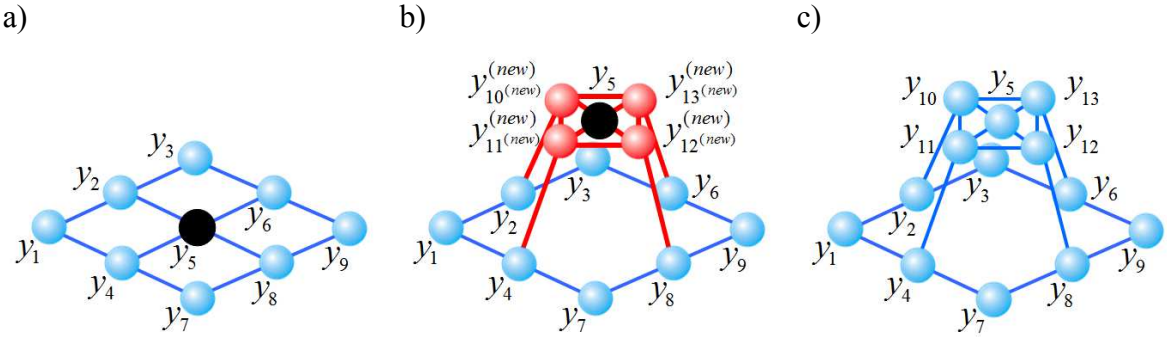
W pracy [6] zaproponowano sieć MIGSOM (ang. *Multilevel Interior Growing Self-organizing Map*) zdolną do zmiany wymiarowości struktury topologicznej, z dwuwymiarowej (mapa neuronów) na trójwymiarową (wielopoziomowa mapa neuronów). Proces uczenia sieci odbywa się podobnie jak w poprzednich dwóch wariantach SOM, przy czym po zakończeniu każdej kolejnej epoki uczenia wyznaczany jest jeden neuron, którego skumulowany błąd kwantyzacji [6] obliczony po zakończeniu danej epoki jest największy i zarazem większy od zakładanego z góry progu (ang. *growth threshold*). W zależności od położenia tego neuronu w topologicznej strukturze sieci aktywowane są trzy rodzaje operacji.

W pierwszym przypadku (zilustrowanym na rys. 2.10), gdy wyznaczony neuron (oznaczony kolorem czarnym na rys. 2.10a i b) jest umieszczony na brzegu pierwszego poziomu mapy neuronów, nowe neurony (oznaczone kolorem czerwonym na rys. 2.10b) wstawiane są w jego bezpośrednim sąsiedztwie topologicznym oraz łączone z nim nowymi połączeniami topologicznymi. W zależności od położenia tego neuronu na krawędzi mapy neuronów możliwe jest wstawienie jednego, dwóch (patrz rys. 2.10b) lub trzech nowych neuronów.




Rys. 2.10. Ilustracja operacji wstawiania nowych neuronów na krawędzi topologicznej struktury sieci MIGSOM [6]: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czarnym oznaczono neuron brzegowy o największym skumulowanym błędzie kwantyzacji; kolorem czerwonym oznaczono nowe neurony)

W drugim przypadku (przedstawionym na rys. 2.11), gdy wyznaczony neuron jest zlokalizowany wewnątrz mapy neuronów (patrz rys. 2.11a), zostaje on przeniesiony do kolejnego, wyższego poziomu, a w jego bezpośrednim sąsiedztwie wstawiane są cztery nowe neurony oraz nowe połączenia topologiczne, tak jak to pokazano na rys. 2.11b i c.




Rys. 2.11. Ilustracja operacji wstawiania kolejnej warstwy nowych neuronów do topologicznej struktury sieci MIGSOM [6]: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czarnym oznaczono neuron wewnątrz mapy o największym skumulowanym błędzie kwantyzacji; kolorem czerwonym oznaczono nowe neurony)

W ostatnim przypadku (przedstawionym na rys. 2.12), gdy wyznaczony neuron jest zlokalizowany na krawędzi wyższej warstwy (poziomu) neuronów, wstawiany jest jeden nowy neuron na tym samym poziomie (i nowe połączenie topologiczne, łączące go z neuronem wyznaczonym) oraz usuwane jest połączenie topologiczne pomiędzy wyznaczonym neuronem, a jego bezpośrednim sąsiadem na niższym poziomie (tak jak to pokazano na rys. 2.12b i c). Wielokrotne powtarzanie powyżej operacji może doprowadzić do usunięcia wszystkich połączeń topologicznych pomiędzy neuronami należącymi do dwóch różnych poziomów, co z kolei spowoduje podzielenie struktury sieci na dwie części.

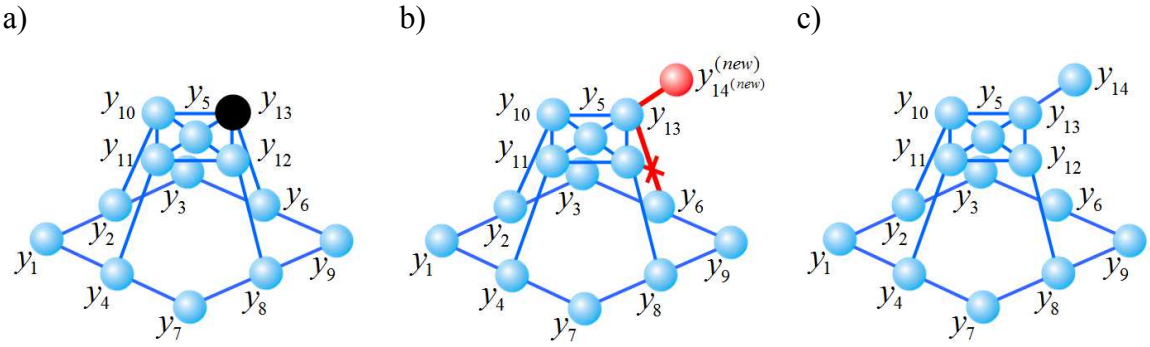
W zależności od lokalizacji nowego neuronu względem neuronów sąsiednich (patrz rys. 2.13), jego wektor wagowy $\boldsymbol{w}_{j^{(new)}}^{(new)}$ wyznaczany jest następująco:

- jeżeli nowy neuron jest zlokalizowany na krawędzi struktury (rys. 2.13a):

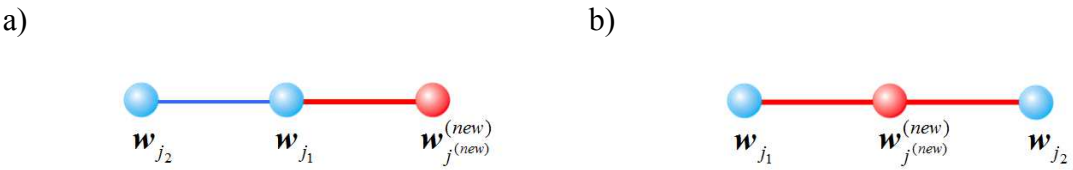
$$\boldsymbol{w}_{j^{(new)}}^{(new)} = 2\boldsymbol{w}_{j_1} - \boldsymbol{w}_{j_2}, \tag{2.14}$$

- jeżeli nowy neuron jest zlokalizowany wewnątrz struktury (rys. 2.13b):

$$\boldsymbol{w}_{j^{(new)}}^{(new)} = \frac{\boldsymbol{w}_{j_1} - \boldsymbol{w}_{j_2}}{2}. \tag{2.15}$$




Rys. 2.12. Ilustracja operacji wstawiania nowego neuronu w skrajnym obszarze kolejnej warstwy neuronów w topologicznej strukturze sieci MIGSOM [6]: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czarnym oznaczono neuron brzegowy na wyższym poziomie o największym skumulowanym błędzie kwantyzacji; kolorem czerwonym oznaczono nowy neuron)



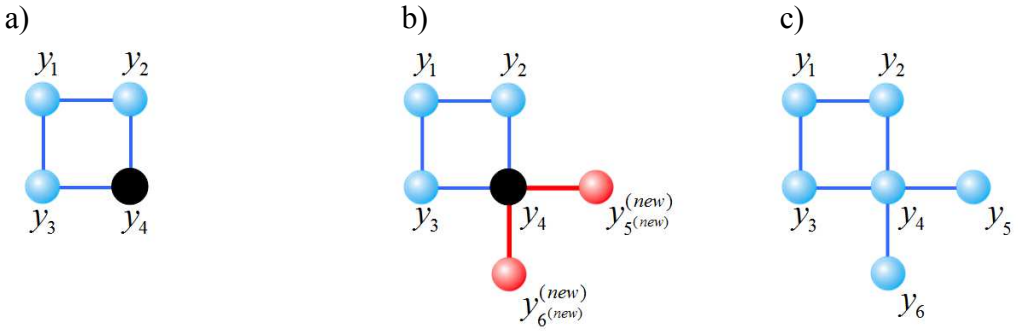

Rys. 2.13. Ilustracja wariantów lokalizacji nowego neuronu względem jego neuronów sąsiednich w topologicznej strukturze sieci MIGSOM

### 2.3.2. Sieci typu IGG (ang. *Incremental Grid Growing*) [14]

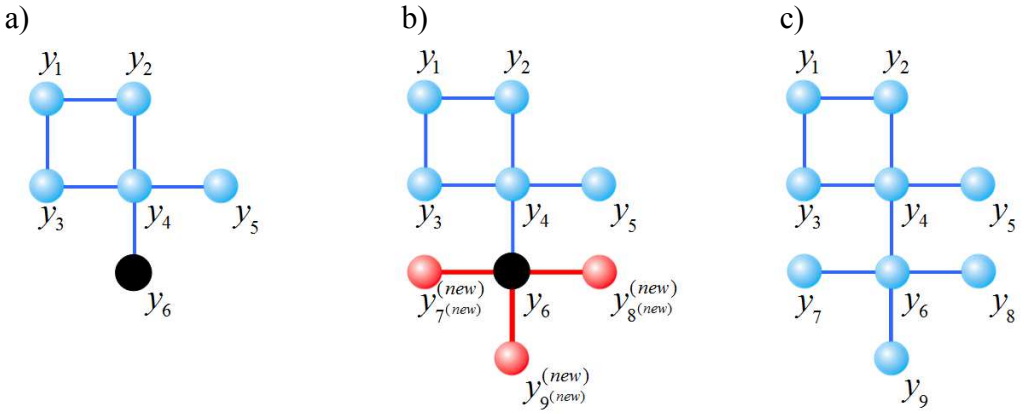
W pracy [14] zaproponowano sieć IGG (ang. *Incremental Grid Growing*), która po zakończeniu każdej kolejnej epoki uczenia aktywuje dwa mechanizmy odpowiedzialne za dodawanie nowych neuronów do struktury sieci oraz za dodawanie lub usuwanie połączeń topologicznych. Podobnie jak w poprzednich podejściach, proces uczenia rozpoczyna się od inicjowania sieci z małą liczba neuronów (np. 4 neurony). W kolejnych epokach uczenia struktura sieci powiększa się, aż do zakładanej z góry maksymalnej liczby neuronów.

Funkcjonowanie mechanizmu wstawiania nowych neuronów do sieci ilustrują rys. 2.14 (przypadek wstawiania dwóch nowych neuronów) i rys. 2.15 (przypadek wstawiania trzech nowych neuronów). Kolorem czarnym oznaczono neuron o największym skumulowanym błędzie kwantyzacji [14]. W jego sąsiedztwie wstawiane są nowe neu-

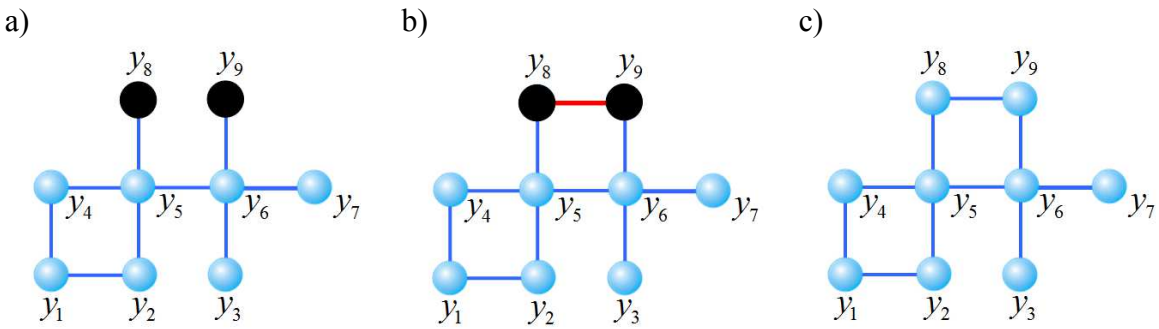
rony (oznaczone kolorem czerwonym). Wektory wagowe nowych neuronów obliczane są analogicznie jak w przypadku sieci MIGSOM [6] (patrz podrozdział 2.3.1).




Rys. 2.14. Ilustracja operacji wstawiania nowych neuronów do sieci IGG (przypadek wstawiania dwóch neuronów): struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji

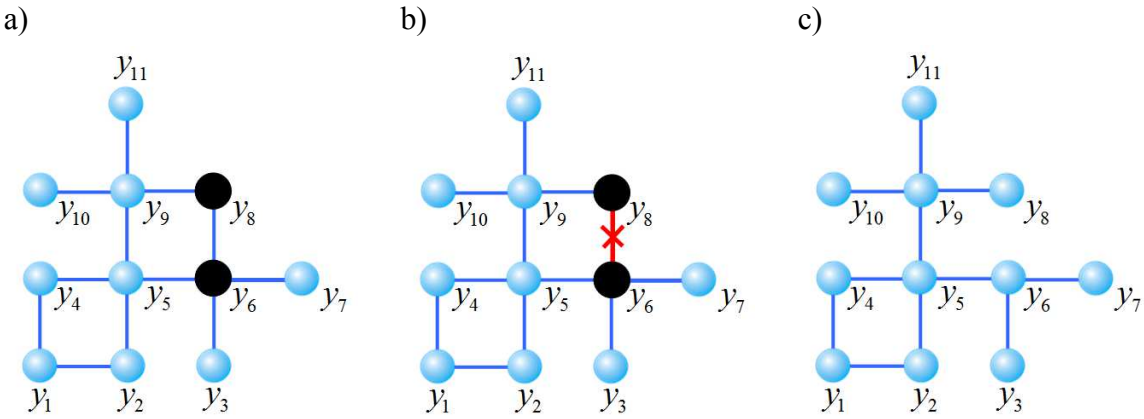



Rys. 2.15. Ilustracja operacji wstawiania nowych neuronów do sieci IGG (przypadek wstawiania trzech neuronów): struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji




Rys. 2.16. Ilustracja operacji wstawiania nowego połączenia topologicznego w sieci IGG: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czarnym oznaczono neurony wytypowane do połączenia)

Mechanizm dodawania nowego połączenia topologicznego jest aktywowany dla takich par neuronów, które są zlokalizowane w przestrzeni danych w odległości mniejszej od z góry zakładanego progu (odległość pomiędzy neuronami wyznaczana jest według miary euklidesowej; na rys. 2.16 przedstawiono funkcjonowanie tego mechanizmu). Analogicznie, mechanizm usuwania połączenia topologicznego jest aktywowany dla takich par neuronów, które są oddalone od siebie w przestrzeni danych na odległość większą od zakładanego z góry progu (patrz rys. 2.17).




Rys. 2.17. Ilustracja operacji usuwania połączenia topologicznego w sieci IGG: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czarnym oznaczono rozłączane neurony)

## 2.4. Samoorganizujące się sieci neuronowe z mechanizmami powiększania i redukcji rozmiaru struktury topologicznej oraz jej podziału na niezależne części

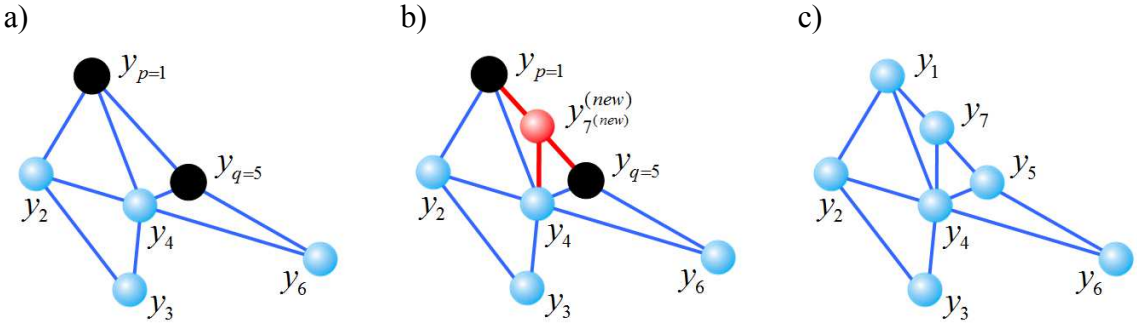
W tym podrozdziale przedstawiono pięć najbardziej zaawansowanych wariantów sieci SOM, które oprócz właściwości wyżej rozważanych podejść (czyli zdolności do powiększania swojej struktury topologicznej oraz do rozłączania jej na mniejsze części) posiadają dodatkowo zdolność do redukcji struktury poprzez usuwanie z niej neuronów.

### 2.4.1. Sieci typu GCS (ang. *Growing Cell Structures*) [37]

Topologiczna struktura kolejnej modyfikacji konwencjonalnej sieci SOM – sieci GCS (ang. *Growing Cell Structures*) [37] – ma postać komórkową. W sensie geometrycznym jest nieregularną formą trójkątów (tzw. komórek), przylegających wzajemnie do siebie bokami, tzn. każde połączenie topologiczne jest bokiem trójkąta, a jego wierz-

chołkami są neurony. Modyfikacja struktury sieci w trakcie trwania procesu uczenia odbywa się cyklicznie, każdorazowo po zakończeniu określonej z góry liczby iteracji. Aktywowane są dwa mechanizmy odpowiedzialne, odpowiednio, za dodawanie nowych neuronów do sieci lub usuwanie istniejących.

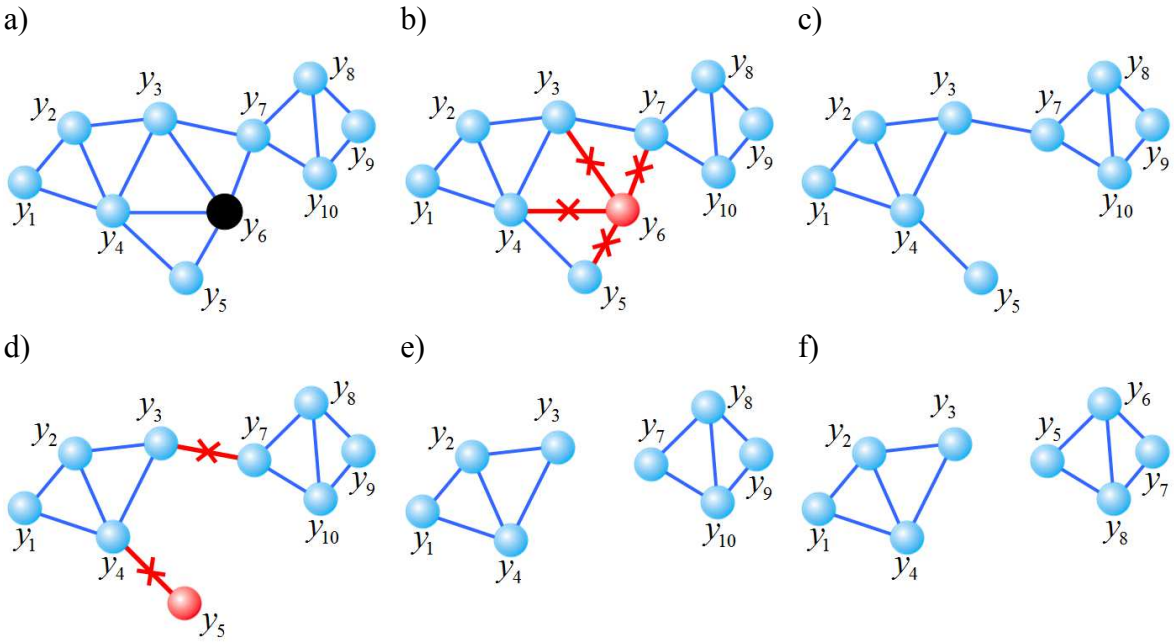
W celu dodania nowego neuronu do struktury sieci (patrz również na rys. 2.18), wyznaczany jest neuron o największej liczbie zwycięstw (uzyskanych w ostatnim cyklu uczenia, w ramach algorytmu WTM) oraz jeden z jego bezpośrednich sąsiadów, najbardziej oddalony od niego w przestrzeni danych (neurony o numerach, odpowiednio, $p$ i $q$, oznaczone na rys. 2.18a kolorem czarnym). Nowy neuron wstawiany jest pomiędzy dwa wyznaczone neurony, przy czym połączenia topologiczne pomiędzy neuronami uzupełniane są w taki sposób, aby nowy neuron był wierzchołkiem jednego (lub więcej) trójkątów w topologicznej strukturze sieci (tak jak to pokazano na rys. 2.18b-c). Wektory wagowe nowych neuronów obliczane są analogicznie jak w metodzie MIGSOM (patrz podrozdział 2.3.1).




Rys. 2.18. Ilustracja operacji dodawania nowego neuronu do topologicznej struktury sieci GCS: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b) oraz po zakończeniu (c) operacji (kolorem czerwonym oznaczono nowy neuron i nowe połączenia topologiczne)

Po zakończeniu operacji dodawania nowego neuronu, aktywowany jest kilkuetapowy mechanizm usuwania neuronów z sieci, zilustrowany na rys. 2.19. W pierwszym etapie, wyznaczane są neurony o liczbie zwycięstw mniejszej od z góry przyjętego progu (oznaczone kolorem czarnym na rys. 2.19a). W kolejnym etapie, neurony te są usuwane wraz z ich połączeniami topologicznymi (rys. 2.19b). W powstałej strukturze sieci (rys. 2.19c) wyszukiwane są neurony nie będące wierzchołkami trójkątów oraz połączenia topologiczne nie będące bokami trójkątów (rys. 2.19d) i one również są usuwane. Proces ten jest powtarzany, aż zostaną usunięte wszystkie takie neurony i połączenia

(rys. 2.19e). W końcowym etapie, pozostałe neurony sieci są przenumerowane w porządku naturalnym (rys. 2.19f).




Rys. 2.19. Ilustracja operacji usuwania neuronu z topologicznej struktury sieci GCS: struktura sieci przed rozpoczęciem (a), w trakcie trwania (b-e) oraz po zakończeniu (f) operacji (kolorem czerwonym oznaczono usuwane neurony i usuwane połączenia topologiczne)

### 2.4.2. Sieci typu GNG (ang. *Growing Neural Gas*) [38]

Sieć neuronowa GNG (ang. *Growing Neural Gas*) [38] została opracowana na podstawie sieci neuronowej samouczącej się, zwanej gazem neuronowym (ang. *Neural Gas*) [76]. Modyfikacja struktury sieci GNG zachodzi zarówno w każdej iteracji procesu uczenia (po aktualizacji jej wektorów wagowych), jak i po zakończeniu pewnego cyklu uczenia, trwającego z góry założoną liczbę iteracji (w szczególnym przypadku, po epoce uczenia). I tak, w pojedynczej iteracji procesu uczenia, po zaprezentowaniu wektora danych uczących na wejścia sieci aktywowane są kolejno następujące operacje:

1) utworzenie nowego połączenia topologicznego (o ile nie istniało) pomiędzy dwoma neuronami, których wektory wagowe są zlokalizowane najbliżej wektora danych uczących w przestrzeni danych; każde połączenie topologiczne charakteryzuje się tzw. wiekiem, przy czym dla nowego połączenia przyjmuje się wiek zerowy,

2) aktualizacja wektorów wagowych powyższych neuronów oraz ich sąsiadów zgodnie z regułą uczenia (2.2), a następnie zwiększenie wieku połączeń topologicznych tych neuronów, które nie biorą udziału w aktualizacji wektorów wagowych,

3) usunięcie połączeń topologicznych, których wiek przekracza z góry ustaloną wartość progową,

4) usunięcie z sieci neuronów nie posiadających połączeń topologicznych.

Z kolei, po zakończeniu cyklu uczenia aktywowana jest operacja wstawiania nowego neuronu do struktury sieci i łączenia go wiązaniami topologicznymi z dwoma neuronami o największych skumulowanych błędach kwantyzacji [38]. Uśrednienie wektorów wagowych tych neuronów daje wektor wagowy nowego neuronu. Proces uczenia zostaje zakończony po przeliczeniu ustalonej z góry liczby iteracji lub po osiągnięciu ustalonej z góry liczby neuronów.

### 2.4.3. Sieci typu GWR (ang. *Grow When Required*) [75]

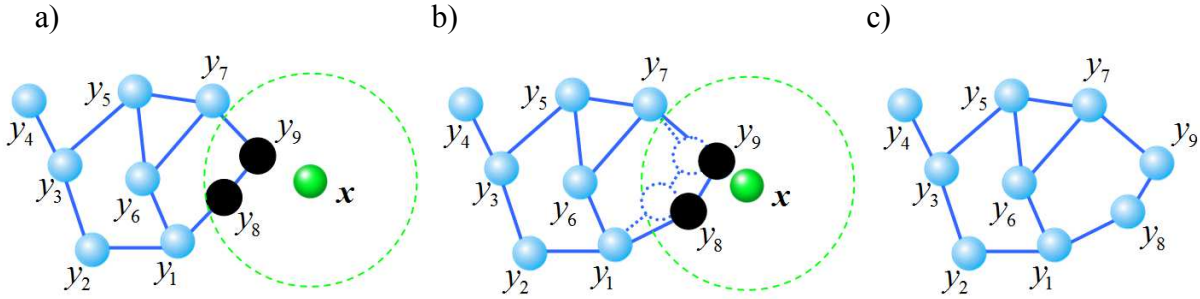
Wariantem sieci GNG jest sieć GWR (ang. *Grow When Required*) przedstawiona w pracy [75], w której operacja dodawania nowych neuronów aktywowana jest w każdej iteracji, tuż po zakończeniu operacji usuwania neuronów z sieci (patrz pkt 4 procedury modyfikacji struktury sieci GNG). Pozostałe operacje aktywowane są tak jak w przypadku sieci GNG. Nowy neuron wstawiany jest pomiędzy dwa neurony, których wektory wagowe są zlokalizowane najbliżej wektora danych uczących w przestrzeni danych, jeżeli spełnione są dwa warunki:

- liczba zwycięstw neuronu, którego wektor wagowy jest zlokalizowany najbliżej wektora danych uczących, jest mniejsza od przyjętego z góry progu,
- poziom aktywności tego neuronu jest mniejszy od przyjętego z góry progu; poziom aktywności jest mierzony pewną arbitralnie dobraną funkcją, przyjmującą wartości z przedziału (0, 1], odwrotnie proporcjonalne do odległości pomiędzy wektorem wagowym neuronu, a wektorem danych uczących (tzn. wartość równą 1, jeżeli odległość ta jest zerowa oraz bliską zeru, jeżeli odległość ta jest relatywnie duża – szczegóły zawiera praca [75]).

### 2.4.4. Sieci typu IGNG (ang. *Incremental Growing Neural Gas*) [92]

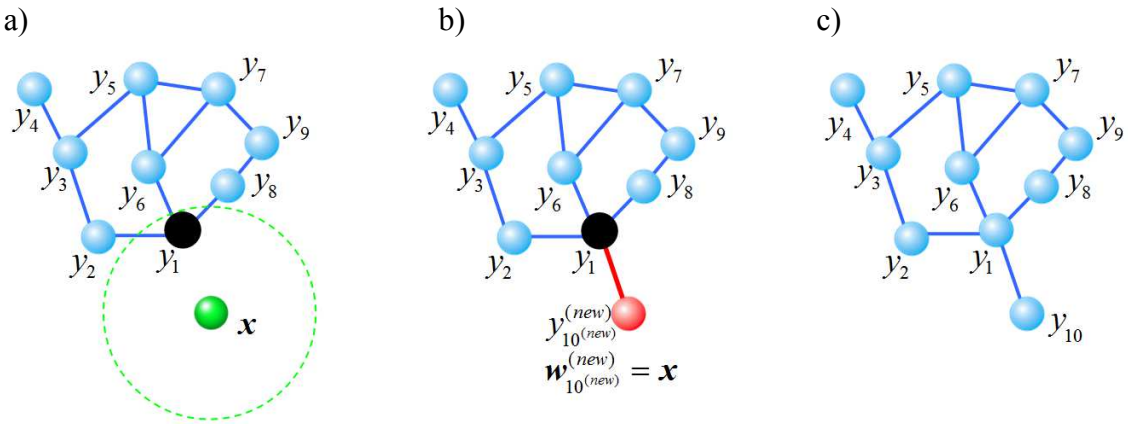
W pracy [92] przedstawiono sieć IGNG (ang. *Incremental Growing Neural Gas*), która jest kolejnym wariantem sieci GNG, bardzo podobnym koncepcyjnie do sieci GWR. Wprowadzono możliwość dodawania nowych neuronów, które w początkowej fazie swojego funkcjonowania w strukturze sieci nie muszą być połączone wiązaniami topologicznymi z innymi neuronami (łączone są w późniejszym czasie lub są usuwane ze struktury, jeżeli do połączenia nie dojdzie po określonej z góry liczbie iteracji). Algorytm uczenia sieci IGNG jest analogiczny do algorytmu uczenia sieci GWR, przy czym warunki aktywacji mechanizmów wstawiania nowych neuronów są inne. Po zakończeniu operacji usuwania neuronów z sieci (patrz podrozdział 2.4.2, pkt 4 procedury modyfikacji struktury sieci GNG), aktywacja mechanizmu wstawiania nowych neuronów do sieci uzależniona jest od rozmieszczenia pozostałych neuronów w przestrzeni danych.

Jeżeli odległości pomiędzy próbką danych uczących, a dwoma najbliższymi jej neuronami w przestrzeni danych są mniejsze od zakładanego z góry progu, wówczas operacja wstawienia nowego neuronu nie zostaje aktywowana (patrz na rys. 2.20). Wagi tych neuronów podlegają jedynie adaptacji, zgodnie z regułą uczenia (2.2), w ramach pkt 2 procedury modyfikacji struktury sieci GNG (patrz podrozdział 2.4.2).



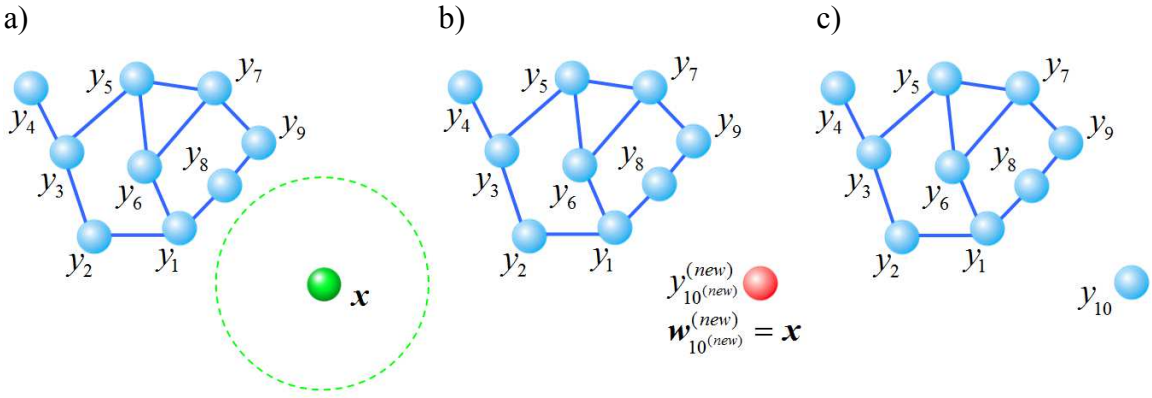

Rys. 2.20. Przypadek rozmieszczenia neuronów sieci w przestrzeni danych względem próbki danych uczących podanej na wejścia sieci, w którym operacja wstawienia nowego neuronu nie jest aktywowana: struktura sieci IGNG przed (a), w trakcie (b) oraz po (c) adaptacji wektorów wagowych

Jeżeli odległość pomiędzy próbką danych uczących, a najbliższym jej neuronem w przestrzeni danych jest mniejsza od zakładanego z góry progu oraz odległość pomiędzy tą próbką, a kolejnym najbliższym neuronem jest większa od tego progu, wówczas operacja wstawienia nowego neuronu zostaje aktywowana, tak jak to pokazano na rys. 2.21. Nowy neuron zostaje połączony wiązaniem topologicznym z neuronem najbliższym próbce danych, a jego wektor wagowy jest równy wektorowi próbki danych.




Rys. 2.21. Przypadek rozmieszczenia neuronów sieci w przestrzeni danych względem próbki danych uczących podanej na wejścia sieci, w którym operacja wstawienia nowego neuronu jest aktywowana: struktura sieci IGNG przed (a), w trakcie (b) oraz po (c) dodaniu nowego neuronu

W ostatnim przypadku, jeżeli odległość pomiędzy próbką danych uczących, a najbliższym jej neuronem w przestrzeni danych jest większa od zakładanego z góry progu, wówczas operacja wstawienia nowego neuronu również zostaje aktywowana, tak jak to pokazano na rys. 2.22, lecz nowy neuron nie zostaje połączony wiązaniem topologicznym z innym neuronem. Podobnie jak w poprzednim przypadku, jego wektor wagowy jest równy wektorowi próbki danych. Jeżeli po upływie z góry założonej liczby iteracji taki neuron nie zostanie połączony z innymi, wówczas zostanie usunięty z sieci.




Rys. 2.22. Przypadek rozmieszczenia neuronów sieci w przestrzeni danych względem próbki danych uczących podanej na wejścia sieci, w którym operacja wstawienia nowego neuronu jest aktywowana, lecz nowy neuron nie jest połączony z innymi neuronami: struktura sieci IGNG przed (a), w trakcie (b) oraz po (c) dodaniu nowego neuronu

### 2.4.5. Sieci typu DSOM (ang. *Dynamic SOM*) [46]

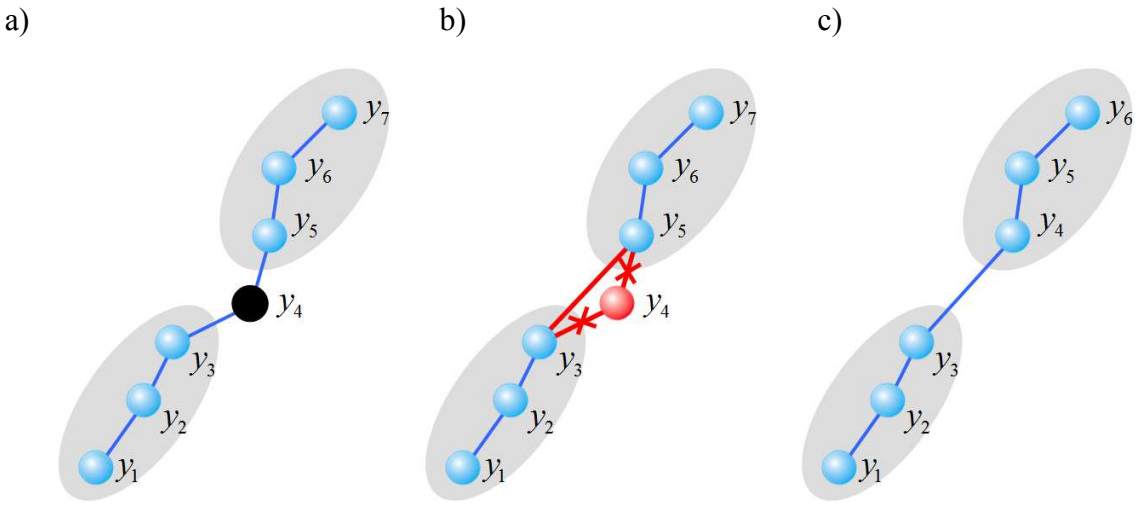
W literaturze dotyczącej wyżej przedstawionych wariantów sieci SOM, zdolnych do podziału topologicznej struktury na części, zagadnienie ponownego łączenia dwóch podstruktur sieci w jedną większą strukturę nie jest w żaden sposób podejmowane. Autorzy nie adresują tego problemu bezpośrednio, pomimo że rozważane podejścia posiadają mechanizmy umożliwiające łączenie podstruktur sieci neuronowej, tzn. mechanizmy wstawiania nowych połączeń topologicznych do struktury sieci. Operacja taka jest jedną z kluczowych w procesie uczenia sieci, gdyż umożliwia ewentualną korektę podziału struktury sieci, dokonanego we wczesnej fazie uczenia, jeżeli takowy podział okazałby się w dalszej fazie uczenia niekorzystny.

Dynamiczna, samoorganizująca się sieć neuronowa z jednowymiarowym sąsiedztwem topologicznym DSOM (ang. *Dynamic SOM*) przedstawiona w pracach [46, 47, 48, 49], w odróżnieniu od poprzednich podejść, adresuje powyższy problem bezpośrednio. Modyfikacja struktury sieci, mającej tym razem postać łańcucha neuronów, odbywa się z wykorzystaniem pięciu mechanizmów, aktywowanych kolejno po zakończeniu każdej epoki uczenia sieci: a) mechanizmu usuwania pojedynczych neuronów z łańcucha sieci, b) mechanizmu rozłączania łańcucha sieci na dwa podłańcuchy, które w trakcie dalszego uczenia mogą się znów dzielić na części, c) mechanizmu usuwania zbyt krótkich podłańcuchów sieci, d) mechanizmu wstawiania dodatkowych neuronów do łańcucha sieci oraz e) mechanizmu łączenia wybranych podłańcuchów sieci.

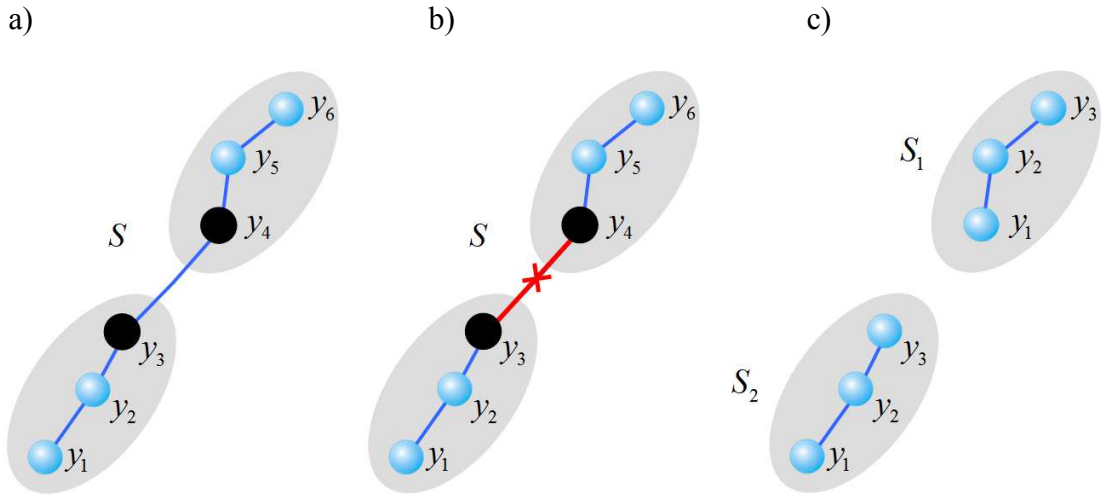
Mechanizm usuwania pojedynczych neuronów (przedstawiony na rys. 2.23) jest aktywowany dla tych neuronów, których liczba zwycięstw po zakończonej epoce uczenia jest mniejsza od z góry przyjętego progu (tzw. małoaktywnych neuronów).

Mechanizm rozłączania łańcucha neuronów na dwa podłańcuchy (patrz rys. 2.24) polega na usunięciu połączenia topologicznego pomiędzy dwoma neuronami, które oddalone są od siebie w przestrzeni danych na odległość większą od z góry przyjętego progu. Mechanizm aktywowany jest wielokrotnie dla każdej takiej pary neuronów, przy czym w pierwszej kolejności rozłączane są neurony najbardziej oddalone od siebie.

Trzeci mechanizm usuwa zbyt krótkie podłańcuchy sieci, tzn. zawierające liczbę neuronów mniejszą od zakładanego z góry progu (zwykle 2, 3 neurony). Najczęściej takie podłańcuchy nie odwzorowują danych zlokalizowanych w większych skupiskach, lecz pojedyncze próbki danych uznawane często za tzw. szum.




Rys. 2.23. Operacja usuwania małoaktywnego neuronu z sieci DSOM: struktura sieci po zakończeniu epoki uczenia (a) (kolorem czarnym oznaczono neuron wytypowany do usunięcia), w trakcie (b) oraz po zakończeniu operacji usuwania neuronu i po przenumerowaniu neuronów (c) (kolorem czerwonym oznaczono usuwany neuron i modyfikowane połączenia topologiczne)




Rys. 2.24. Operacja rozłączania łańcucha neuronów sieci DSOM na dwa podłańcuchy: struktura sieci przed rozłączeniem (a) w trakcie (b) i po rozłączeniu na dwa podłańcuchy oraz po przenumerowaniu neuronów (kolorem czarnym oznaczono parę neuronów znacznie oddalonych od siebie w przestrzeni danych, a kolorem czerwonym usuwane połączenie topologiczne)

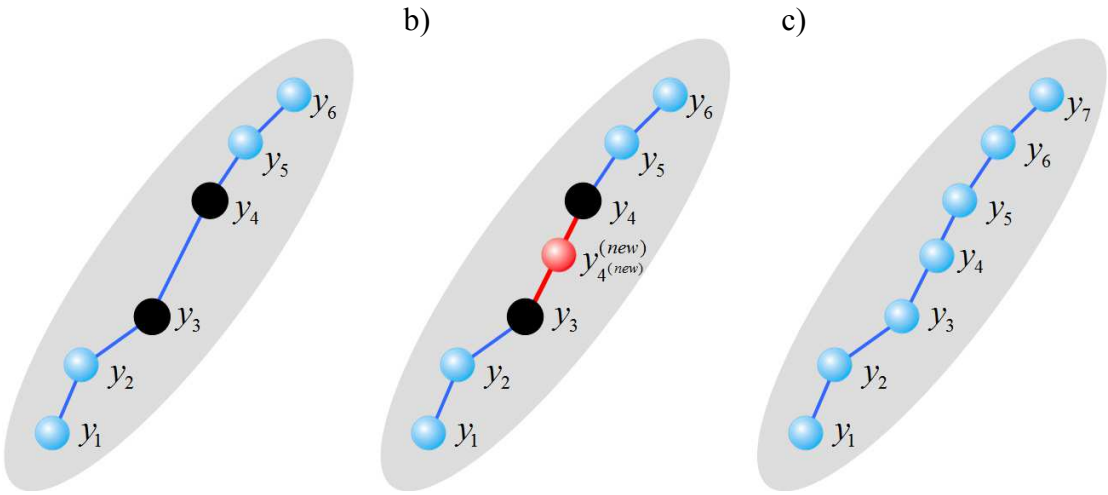
Kolejny mechanizm wstawia dodatkowe neurony do łańcucha sieci, w otoczeniu bardzo aktywnych neuronów (tzn. takich, których liczba zwycięstw w zakończonej epoce uczenia jest większa od z góry przyjętego progu), w celu przejęcia części ich aktyw-

ności. Może być rozważany dla trzech różnych przypadków rozmieszczenia bardzo aktywnych neuronów względem danych w przestrzeni danych.

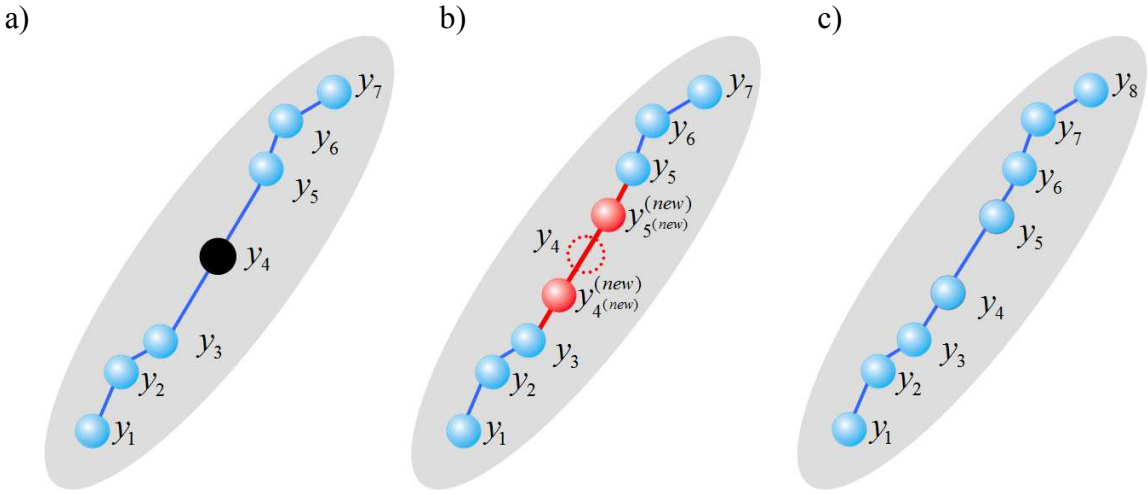
W pierwszym przypadku (przedstawionym na rys. 2.25), nowy neuron zostaje wstawiony pomiędzy dwa bardzo aktywne neurony. Nowy neuron przejmuje część aktywności od swoich sąsiadów w kolejnej epoce uczenia. Operacja zachodzi dla neuronów zlokalizowanych w obszarach o dużej gęstości danych.

W drugim przypadku (patrz rys. 2.26), mechanizm jest aktywowany dla jednego bardzo aktywnego neuronu, którego bezpośredni sąsiedzi w topologicznej strukturze sieci wykazują stosunkowo małą aktywność. Neuron ten zostaje zastąpiony dwoma nowymi neuronami. Przewiduje się, że po zakończeniu kolejnej epoki uczenia poziom aktywności nowych neuronów będzie mniejszy i zbliży się do poziomu aktywności neuronów sąsiednich.

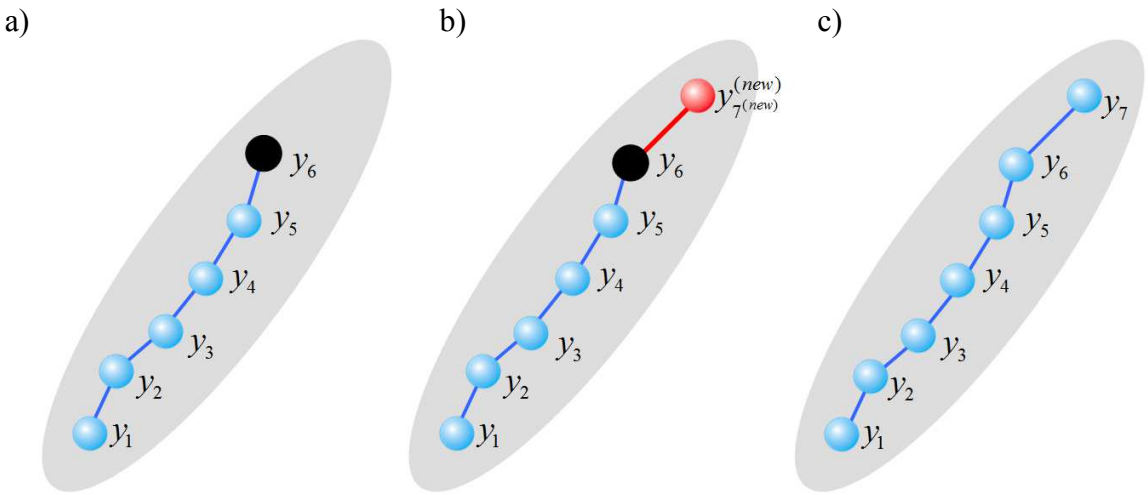
W ostatnim przypadku, nowy neuron dodawany jest na końcu (patrz rys. 2.27) lub analogicznie na początku podłańcucha sieci w bezpośrednim sąsiedztwie topologicznym bardzo aktywnego neuronu. Mechanizm ten odgrywa bardzo ważną rolę w procesie uczenia sieci, gdyż umożliwia „rozrastanie" jej struktury w obszarach skupisk danych o „cienkich" kształtach szkieletowych.




Rys. 2.25. Operacja wstawiania nowego neuronu pomiędzy dwa bardzo aktywne neurony sieci DSOM: struktura sieci po zakończeniu epoki uczenia (a), w trakcie (b) oraz po zakończeniu (c) operacji dodawania neuronu (kolorem czarnym oznaczono bardzo aktywne neurony, natomiast kolorem czerwonym oznaczono nowy neuron i modyfikowane połączenia topologiczne)

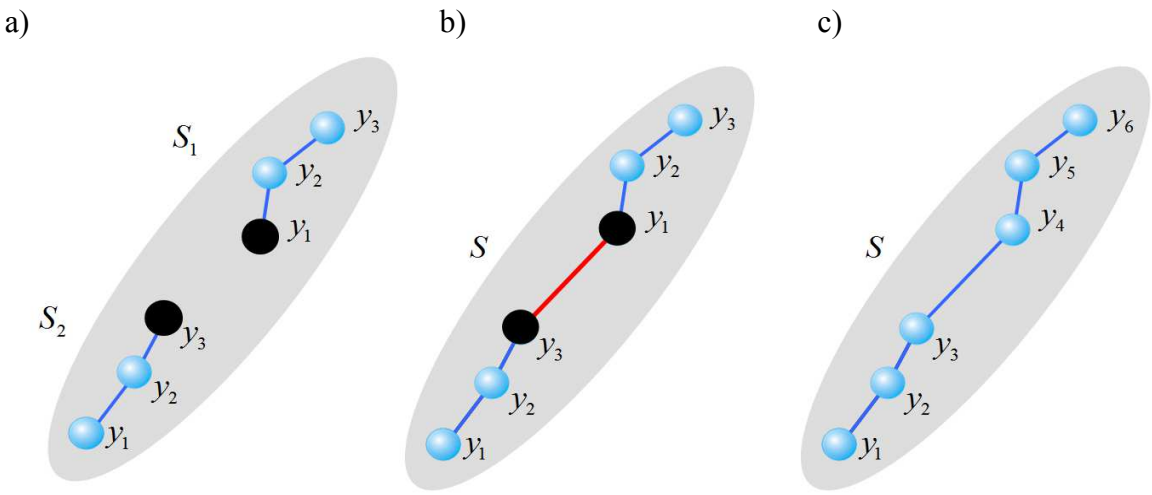



Rys. 2.26. Operacja zastępowania bardzo aktywnego neuronu sieci DSOM dwoma nowymi neuronami: struktura sieci po zakończeniu epoki uczenia (a), w trakcie (b) oraz po zakończeniu (c) operacji dodawania neuronów (kolorem czarnym oznaczono bardzo aktywny neuron, natomiast kolorem czerwonym oznaczono nowe neurony i modyfikowane połączenia topologiczne)




Rys. 2.27. Operacja dodania nowego neuronu na końcu łańcucha neuronów sieci DSOM: struktura sieci po zakończeniu epoki uczenia (a), w trakcie (b) oraz po zakończeniu (c) operacji dodawania neuronu (kolorem czarnym oznaczono bardzo aktywny neuron, natomiast kolorem czerwonym oznaczono nowy neuron i nowe połączenie topologiczne)

Mechanizm łączenia wybranych podłańcuchów sieci polega na wstawieniu połączenia topologicznego pomiędzy dwa neurony należące od odrębnych podłańcuchów, zlokalizowane na początku lub na końcu tych podłańcuchów. Mechanizm jest aktywowany, jeżeli odległość pomiędzy tymi dwoma neuronami w przestrzeni danych jest mniejsza od z góry przyjętego progu.




Rys. 2.28. Operacja połączenia dwóch podłańcuchów neuronów sieci DSOM w jeden łańcuch: struktura sieci przed połączeniem (a) i po połączeniu (b) podłańcuchów oraz po przenumerowaniu neuronów (c) (kolorem czarnym oznaczono parę neuronów zlokalizowanych blisko siebie w przestrzeni danych, a kolorem czerwonym nowe połączenie topologiczne)

## 2.5. Podsumowanie

W tym rozdziale, w pierwszej kolejności, przedstawiono klasyczną ideę grupowania danych bazującą na konstrukcji i wizualnej analizie struktury topologicznej (tzw. mapy neuronów lub U-macierzy) konwencjonalnych, samoorganizujących się sieci neuronowych (SOM). Omówiono syntetycznie podstawowe zagadnienia teoretyczne dotyczące konstrukcji i procesu uczenia konwencjonalnych SOM. Następnie, przedstawiono ewolucję powyższej idei grupowania danych, polegającą na stopniowym wprowadzeniu do sieci SOM – w ramach różnych jej wariantów – mechanizmów dynamicznie modyfikujących jej strukturę w trakcie procesu uczenia, w celu umożliwienia sieci lepszego dopasowania się do rozkładu skupisk w zbiorze danych. Mechanizmy te uruchamiają następujące operacje: a) dzielenie struktury na mniejsze podstruktury (mechanizm usuwania połączeń topologicznych pomiędzy neuronami) oraz b) ponowne łączenie podstruktur ze sobą (mechanizm wstawiania nowych połączeń topologicznych), jak również c) powiększanie i d) redukcję rozmiaru struktury sieci (odpowiednio, mechanizmy wstawiania i usuwania neuronów). Pierwsze dwa mechanizmy są odpowiedzialne za zdolność sieci do automatycznego wykrywania liczby skupisk w zbiorze danych, natomiast pozostałe dwa za zdolność sieci do automatycznego doboru liczby neuronów pełniących rolę punktów w wielopunktowych prototypach skupisk danych.

Tabela 2.1 przedstawia zestawienie rozważanych wariantów sieci SOM, wraz z wyszczególnieniem powyższych własności. I tak, warianty GSOM [94] i GGN [39] (podrozdział 2.2) ukierunkowane są jedynie na powiększanie swojej struktury i nie są w stanie dzielić jej na części. Sieci MIGSOM [6] i IGG [14] (podrozdział 2.3) są zdolne do usuwania pojedynczych połączeń topologicznych pomiędzy parami neuronów i dzięki temu mogą dzielić się na części. Pozostałe podejścia (podrozdział 2.4) posiadają nie tylko zdolność do dodawania nowych neuronów i usuwania swoich połączeń topologicznych, ale również do usuwania neuronów i do wstawiania nowych połączeń topologicznych.

Tabela 2.1. Zestawienie rozważanych w pracy wariantów sieci SOM

| L.p. | Nazwa metody | Mechanizmy modyfikujące topologiczną strukturę sieci neuronowej | | | | Zakres modyfikacji topologicznej struktury sieci neuronowej | | | | Podrozdział |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Wstawianie neuronów | Usuwanie neuronów | Wstawianie połączeń topologicznych | Usuwanie połączeń topologicznych | Powiększanie rozmiaru struktury | Redukcja rozmiaru struktury | Dzielenie na podstruktury | Ponowne łączenie podstruktur | |
| 1 | SOM[1)] | - | - | - | - | - | - | - | - | 2.1 |
| 2 | GSOM[2)] | + | - | - | - | + | - | - | - | 2.2 |
| 3 | GGN[3)] | + | - | - | - | + | - | - | - | |
| 4 | MIGSOM[4)] | + | - | - | + | + | - | + | - | 2.3 |
| 5 | IGG[5)] | + | - | + | + | + | - | + | b/d | |
| 6 | GCS[6)] | + | + | + | + | + | + | + | - | 2.4 |
| 7 | GNG[7)] | + | + | + | + | + | + | + | b/d | |
| 8 | GWR[8)] | + | + | + | + | + | + | + | b/d | |
| 9 | IGNG[9)] | + | + | + | + | + | + | + | b/d | |
| 10 | DSOM[10)] | + | + | + | + | + | + | + | + | |

b/d – brak jednoznacznej informacji w literaturze na temat możliwości ponownego łączenia podstruktur sieci neuronowej, [1)] SOM (ang. *Self-Organizig Map*) [66, 67, 68], [2)] GSOM (ang. *Growing SOM*) [94], [3)] GGN (ang. *Growing Grid Network*) [39], [4)] MIGSOM (ang. *Multilevel Interior Growing Self-organizing Maps*) [6], [5)] IGG (ang. *Incremental Grid Growing*) [14], [6)] GCS (ang. *Growing Cell Structures*) [37], [7)] GNG (ang. *Growing Neural Gas*) [38], [8)] GWR (ang. *Grow When Required*) [75], [9)] IGNG (ang. *Incremental Growing Neural Gas*) [92], [10)] DSOM (ang. Dynamic SOM) [46, 47, 48, 49].

Podsumowując, należy wyraźnie podkreślić, że w przypadku zdecydowanej większości omawianych wariantów SOM literatura źródłowa przedstawia niewielką liczbę, relatywnie nieskomplikowanych eksperymentów praktycznych, głównie z wykorzystaniem syntetycznych, dwuwymiarowych zbiorów danych. Problemy grupowania danych w złożonych, wielowymiarowych zbiorach danych nie są z reguły podejmowane. Ponadto, wątpliwa jest skuteczność czterech wariantów SOM (IGG, GNG, GWR, IGNG; patrz tabela 2.1, kolumna „Ponowne łączenie podstruktur") w automatycznym wykrywaniu liczby skupisk danych. Są one wprawdzie zdolne do podziału swojej struktury na podstruktury, ale autorzy tych podejść nie podejmują zagadnienia ponownego łączenia podstruktur sieci (pomimo wyposażenia tych sieci w mechanizmy bezpośrednio odpowiedzialne za łączenie podstruktur, tzn. w mechanizmy wstawiania nowych połączeń topologicznych), która to operacja odgrywa kluczową rolę w wykrywaniu skupisk w zbiorach danych. Na tym tle na uwagę zasługuje podejście DSOM, które adresuje bezpośrednio to zagadnienie, a jego skuteczność potwierdzają rezultaty licznych eksperymentów praktycznych z wykorzystaniem syntetycznych i dwuwymiarowych oraz rzeczywistych i wielowymiarowych zbiorów danych, przedstawione w pracach [46, 47, 48, 49].

Wspomniane podejście, bazujące na łańcuchach neuronów, jest jednak ukierunkowane przede wszystkim na wykrywanie „cienkich" szkieletowych skupisk o różnorodnych kształtach. Podejście to jest mniej efektywne w przypadku ważnej klasy skupisk objętościowych, przede wszystkim – w generowaniu wielopunktowych prototypów tych skupisk. Prototypy te – w postaci „wijących się" w obrębie skupisk podłańcuchów neuronów – nie gwarantują w każdym przypadku poprawnego rozkładu poszczególnych punktów danego prototypu w obrębie odpowiadającego mu skupiska danych. Znacznie lepiej sprawdzają się w tej roli struktury drzewopodobne, które również efektywnie mogą być wykorzystywane w wykrywaniu „cienkich" skupisk szkieletowych. Stąd też, podejście DSOM stanowi punkt wyjścia w konstrukcji proponowanego w tej pracy (i przedstawionego w następnym rozdziale) oryginalnego narzędzia do grupowania danych – uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej wyposażonej w mechanizmy jej rozłączania i ponownego łączenia.

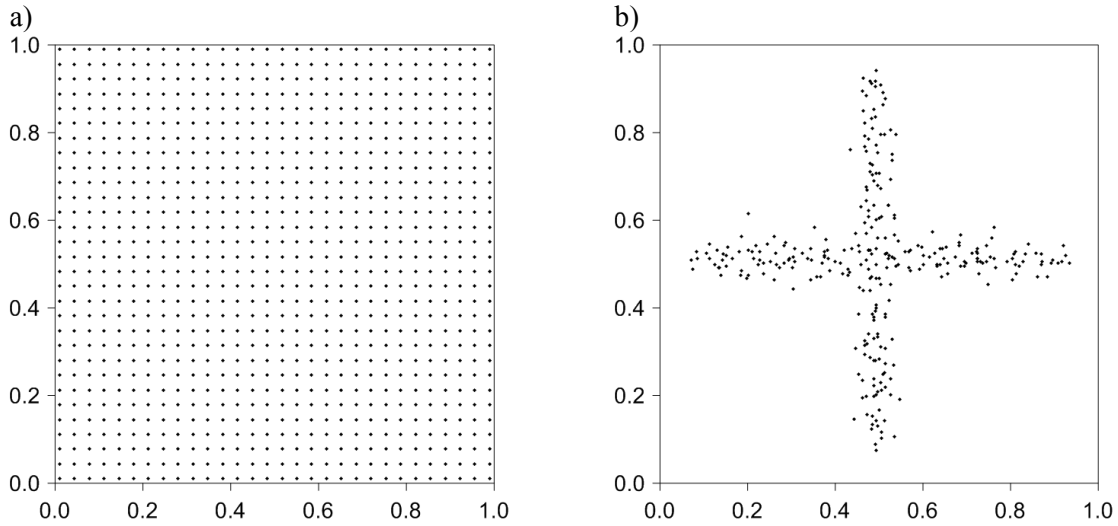
# 3. Uogólnione samoorganizujące się sieci neuronowe o drzewopodobnych strukturach z mechanizmami ich modyfikacji

Rozdział ten jest pierwszym z dwóch rozdziałów prezentujących oryginalne narzędzie do grupowania danych – uogólnioną samoorganizującą się sieci neuronową o ewoluującej, drzewopodobnej strukturze topologicznej – oraz praktyczny test jej funkcjonowania z wykorzystaniem znanych testowych zbiorów danych, reprezentujących różne kategorie problemów grupowania danych. Narzędzie to stanowi odpowiedź na mankamenty alternatywnych podejść, przede wszystkim wariantów konwencjonalnych SOM, przedstawionych w poprzednim rozdziale oraz tradycyjnych technik grupowania danych, o których była mowa we Wprowadzeniu.

Jak już wspomniano w końcowej części poprzedniego rozdziału, punktem wyjścia w konstrukcji proponowanej sieci neuronowej jest – omówiona krótko w podrozdziale 2.4.5 oraz przedstawiona w pracach [46, 47, 48, 49] – dynamiczna, samoorganizująca się sieć neuronowa z jednowymiarowym sąsiedztwem topologicznym (DSOM). W proponowanym rozwiązaniu znacząco udoskonalono mechanizmy usuwania wybranych istniejących i wstawiania nowych połączeń topologicznych pomiędzy neuronami oraz mechanizmy usuwania i wstawiania neuronów. Wprowadzone zmiany maja na celu umożliwienie topologicznej strukturze sieci neuronowej przyjmowanie – w trakcie procesu uczenia – nieporównanie bardziej złożonych form geometrycznych w postaci drzewopodobnych struktur neuronowych (zamiast łańcucha neuronów, jak to ma miejsce w DSOM). W celu zilustrowania przykładowych mankamentów sieci DSOM, poniżej przedstawiono dwa przykłady grupowania danych w syntetycznych, dwuwymiarowych zbiorach danych z jej wykorzystaniem (szczegóły dotyczące doboru parametrów sterujących procesem uczenia sieci DSOM znaleźć można w wymienionych wyżej pracach). Następnie przedstawiono konstrukcję proponowanej, samoorganizującej się sieci neuronowej z ewoluującą, drzewopodobną strukturą topologiczną oraz mechanizmy umożliwiające jej modyfikację w trakcie trwania procesu uczenia. W końcowej części rozdziału omówiono algorytm uczenia sieci, jako adaptację klasycznego algorytmu WTM z podrozdziału 2.1.

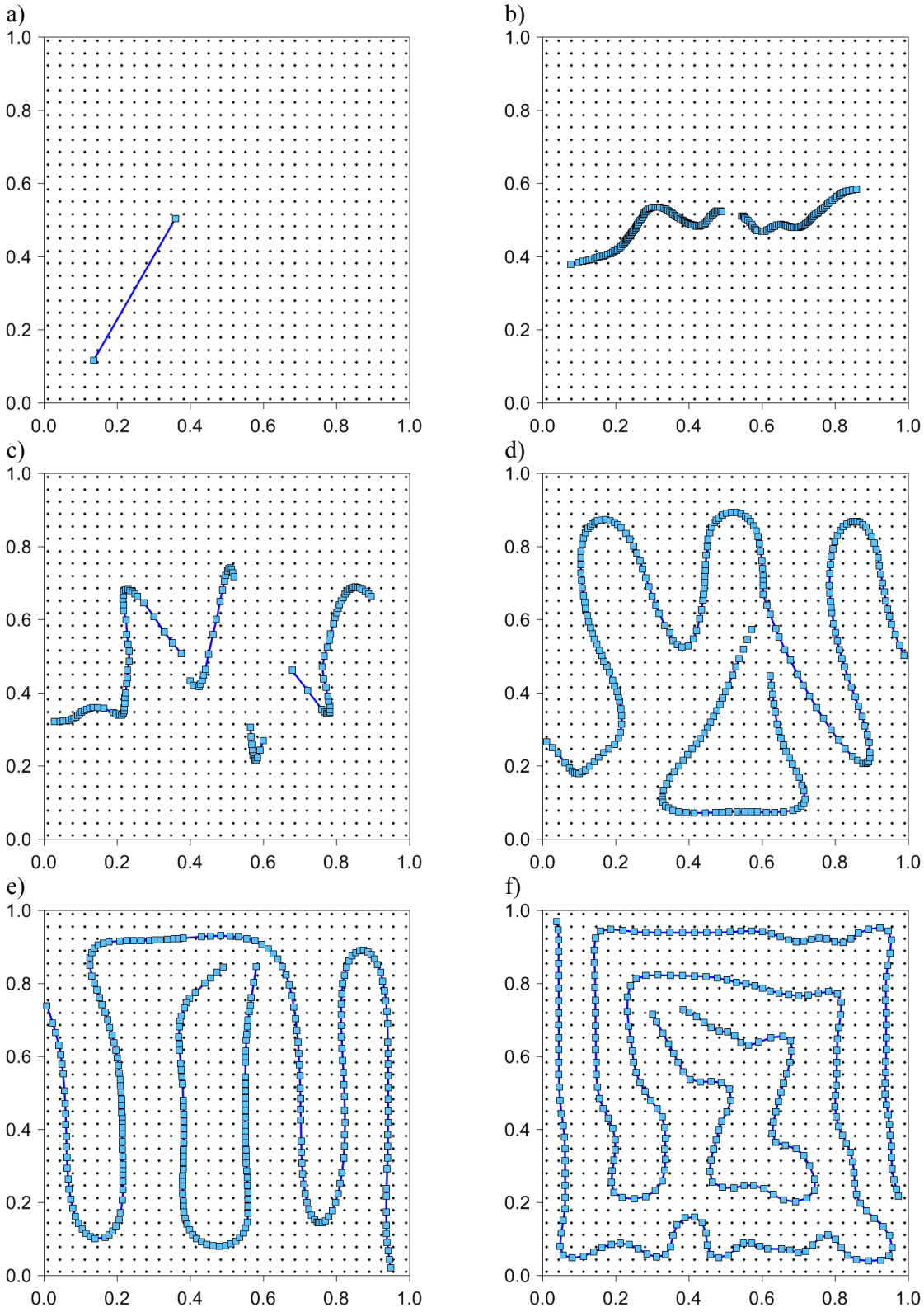
## 3.1. Wprowadzenie

Prawidłowe odwzorowanie kształtu skupisk danych przez ich wielopunktowe prototypy jest kluczowym problemem w zadaniach grupowania danych. Prototypy przyjmujące formę łańcucha neuronów (generowane przez sieć DSOM) z przyczyn naturalnych najlepiej odwzorowują większość skupisk o „cienkich" kształtach szkieletowych, natomiast gorzej sprawdzają się w przypadku skupisk o bardziej złożonych kształtach, w tym skupisk o charakterze objętościowym. W celu przykładowej ilustracji tego problemu, poniżej zostaną rozważone dwa eksperymenty z wykorzystaniem syntetycznych, dwuwymiarowych zbiorów danych przedstawionych na rys. 3.1.

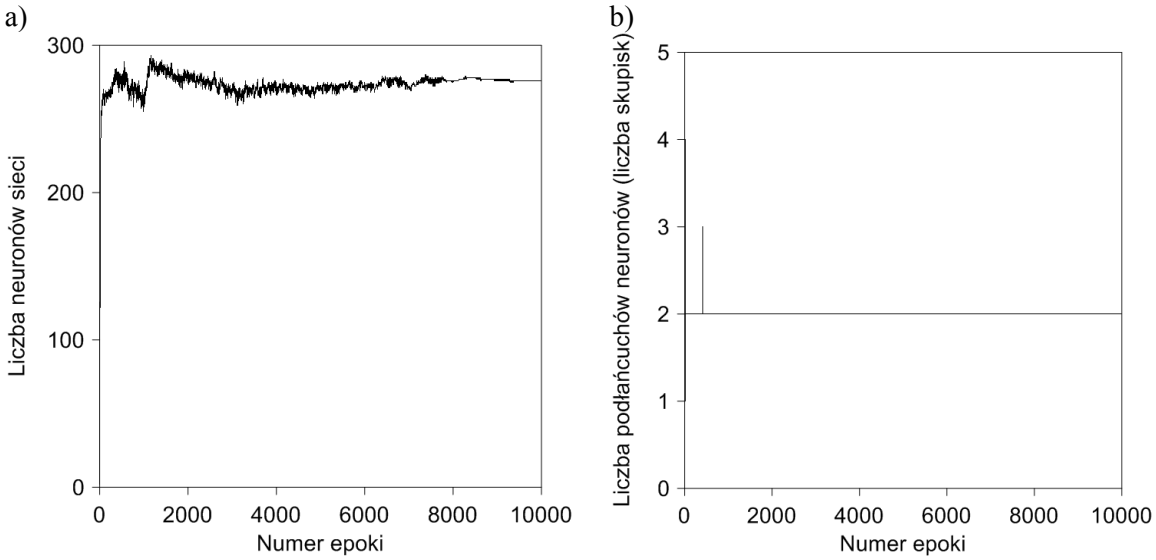



Rys. 3.1. Syntetyczne zbiory danych zawierające po jednym skupisku danych: rozmieszczonych równomiernie na płaszczyźnie (a) oraz o kształcie krzyża (b)

Zbiory reprezentują dwie kategorie problemów grupowania danych. Zawierają po jednym skupisku danych, przy czym w pierwszym przypadku dane rozmieszczone są równomiernie na względnie dużym obszarze (rys. 3.1a), natomiast w drugim, dane rozmieszczone są w obszarze o kształcie krzyża (rys. 3.1b). Rozmieszczenie struktur sieci DSOM w wybranych epokach procesów uczenia, jak również przebiegi zmian liczby neuronów i liczby podłańcuchów sieci w tych eksperymentach przedstawiają, odpowiednio, rys. 3.2 i rys. 3.3 (dla zbioru danych z rys. 3.1a) oraz rys. 3.4 i rys. 3.5 (dla zbioru danych z rys. 3.1b).



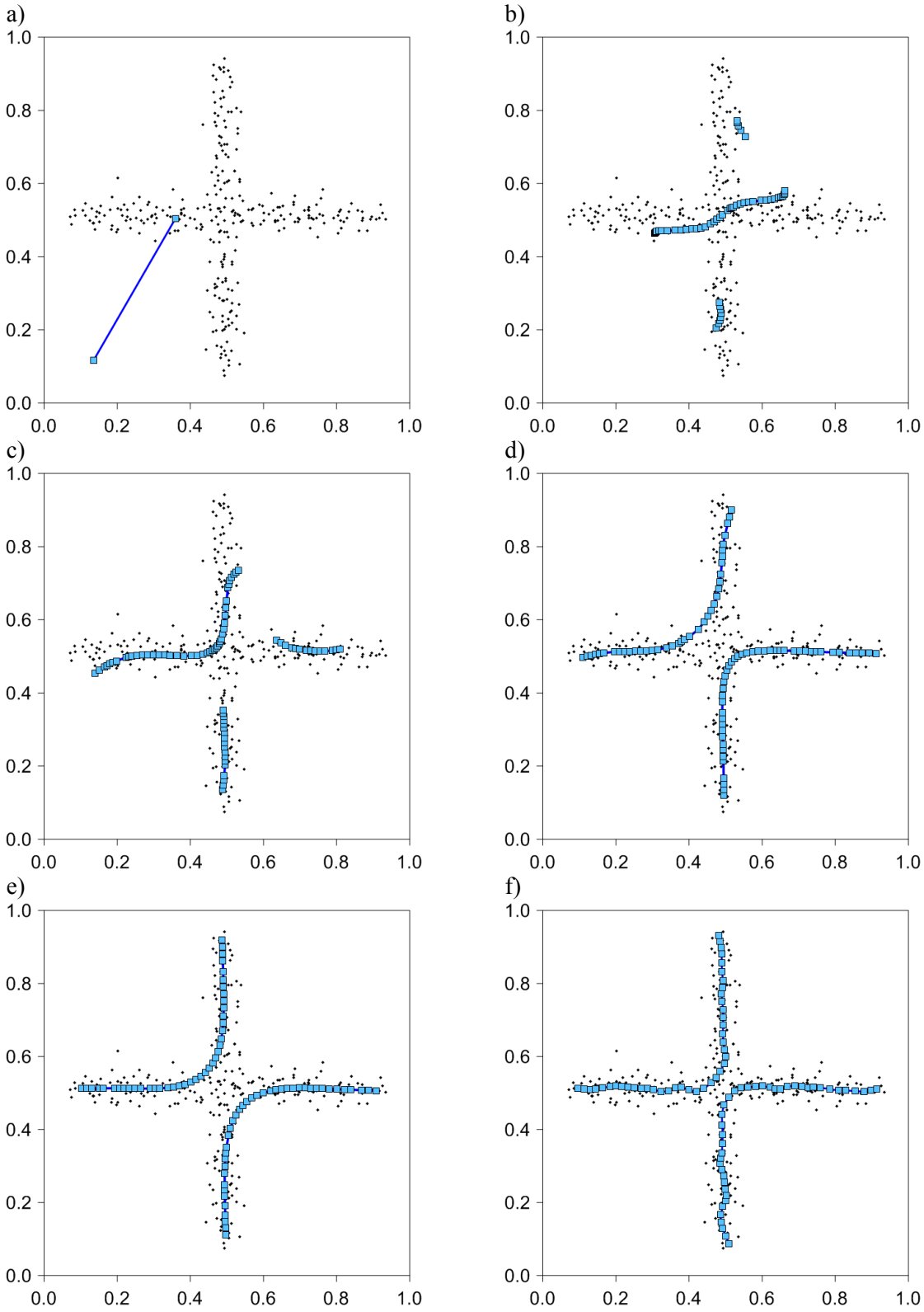

Rys. 3.2. Rozmieszczenie struktury sieci DSOM w obszarze danych z rys. 3.1a na początku procesu uczenia (a) oraz w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)



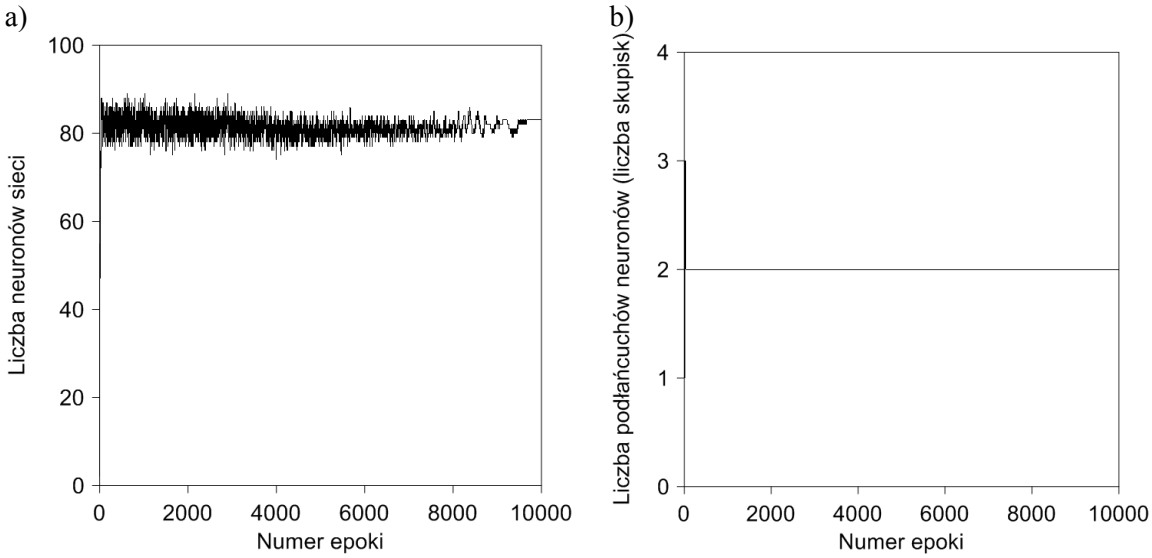

Rys. 3.3. Przebieg zmian liczby neuronów (a) oraz podłańcuchów (b) sieci DSOM w trakcie trwania procesu uczenia dla problemu z rys. 3.2

W przypadku pierwszego eksperymentu (dla zbioru danych z rys. 3.1a), sieć neuronowa już na wczesnym etapie procesu uczenia podzieliła się na dwa podłańcuchy (co jest widoczne na rys. 3.2b), które w trakcie dalszego uczenia tak niekorzystnie rozłożyły się w obszarze danych, że ich ponowne połączenie nie było już możliwe. Połączenie dwóch podłańcuchów jest możliwe, jeżeli ich skrajne neurony są zlokalizowane w przestrzeni danych blisko siebie (co jak widać na rys. 3.2d-f nie ma miejsca). W rezultacie sieć neuronowa błędnie wykryła dwa skupiska w zbiorze danych, zamiast jednego. W przypadku drugiego eksperymentu (dla zbioru danych z rys. 3.1b), sieć neuronowa również podzieliła się na dwa podłańcuchy, które w rezultacie rozłożyły się w obszarze skupiska danych, tak jak to jest przedstawione na rys. 3.4f. Tym razem również, sieć neuronowa błędnie wykryła dwa skupiska danych, zamiast jednego.

W przypadku obu eksperymentów, liczba istniejących skupisk danych nie została prawidłowo zidentyfikowana przez sieć neuronową. Można jednak zauważyć, że wprowadzenie efektywnych mechanizmów umożliwiających łączenie określonych podstruktur sieci dałoby potencjalne możliwości udoskonalenia jej funkcjonowania w problemach grupowania danych. Jest to jedna z idei, która będzie rozwijana w ramach proponowanego podejścia.




Rys. 3.4. Rozmieszczenie struktury sieci DSOM w obszarze danych z rys. 3.1b na początku procesu uczenia (a) oraz w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

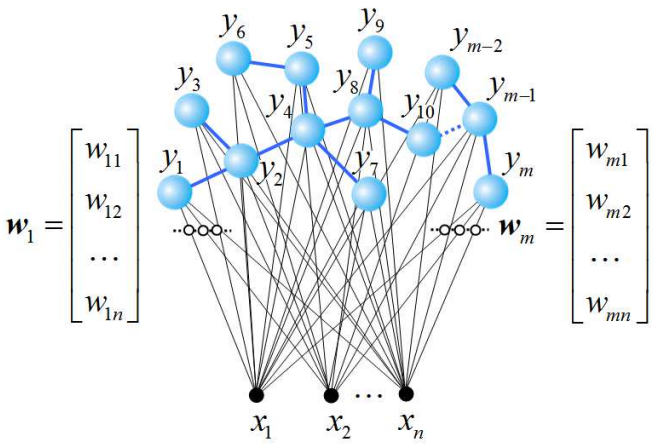



Rys. 3.5. Przebieg zmian liczby neuronów (a) oraz podłańcuchów (b) sieci DSOM w trakcie trwania procesu uczenia dla problemu z rys. 3.4

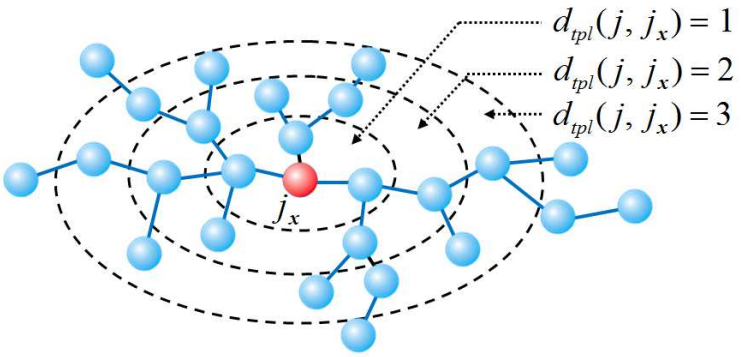
## 3.2. Konstrukcja uogólnionej samoorganizującej się sieci neuronowej o drzewopodobnej strukturze

Jednym z istotnych elementów konstrukcji proponowanej oryginalnej, uogólnionej samoorganizującej się sieci neuronowej jest wprowadzenie mechanizmów umożliwiających jej topologicznej strukturze przyjmowanie złożonych, drzewopodobnych form geometrycznych umożliwiających znacznie lepsze – niż w przypadku łańcucha neuronów – „wpasowanie się" w różnorodne skupiska danych. W proponowanym podejściu dopuszcza się zatem występowanie dodatkowych połączeń topologicznych, tzn. każdy neuron może być powiązany z więcej niż dwoma (jak to miało miejsce w przypadku łańcucha neuronów) neuronami sąsiednimi. W rezultacie, topologiczne powiązania pomiędzy neuronami tworzą geometryczną, drzewopodobną formę, tak jak to pokazano na rys. 3.6. Określenie współrzędnych lokalizacji *j*-tego neuronu ( $j = 1, 2, \ldots, m$ ) w takiej strukturze nie jest możliwe za pomocą wektorów położenia $\boldsymbol{r}_j$, gdyż neurony nie tworzą uporządkowanej formy geometrycznej (tak jak to jest w przypadku sieci SOM przedstawionych na rys. 2.1). Zatem zmianie ulega pierwotna definicja miary odległości $d(\boldsymbol{r}_j, \boldsymbol{r}_{j_x})$ występująca w funkcjach $S(j, j_x, k)$ sąsiedztwa topologicznego prostokątnego (2.8) i gaussowskiego (2.9), wykorzystywanych w regule (2.2) algorytmu uczenia WTM. Miara odległości $d(\boldsymbol{r}_j, \boldsymbol{r}_{j_x})$ zostaje zastąpiona miarą odległości $d_{tpl}(j, j_x)$ ro-

zumianą w sensie liczby połączeń topologicznych pomiędzy parami neuronów, które występują na najkrótszej ścieżce poprowadzonej w topologicznej strukturze sieci od $j$-tego neuronu do neuronu o numerze $j_x$ (określonego formułą (2.3)), tak jak to pokazano na rys. 3.7.




Rys. 3.6. Ilustracja drzewopodobnej struktury topologicznej proponowanej uogólnionej samoorganizującej się sieci neuronowej




Rys. 3.7. Ilustracja zasięgu sąsiedztwa topologicznego wokół neuronu zwyciężającego we współzawodnictwie neuronów (oznaczonego kolorem czerwonym) o promieniu $\lambda = 1$ (najmniejszy zasięg), $\lambda = 2$ i $\lambda = 3$ (największy zasięg)

## 3.3. Mechanizmy umożliwiające modyfikację drzewopodobnej struktury sieci neuronowej w trakcie trwania procesu uczenia

Wprowadzono następujące mechanizmy umożliwiające modyfikację drzewopodobnej struktury sieci neuronowej, automatycznie uaktywniane w trakcie trwania procesu uczenia:

- mechanizm usuwania pojedynczych, małoaktywnych neuronów ze struktury sieci,

- mechanizm usuwania pojedynczych połączeń topologicznych pomiędzy neuronami w celu podziału struktury sieci na dwie podstruktury, które w trakcie dalszego uczenia mogą się znów dzielić na części,
- mechanizm usuwania podstruktur sieci, które zawierają małą liczbę neuronów,
- mechanizm wstawiania do struktury (podstruktury) sieci dodatkowych neuronów w otoczeniu neuronów nadaktywnych,
- mechanizm wstawiania dodatkowych połączeń topologicznych pomiędzy niepołączonymi neuronami w celu ponownego łączenia wybranych podstruktur sieci.

Mechanizmy te są bezpośrednio odpowiedzialne za zdolność sieci do automatycznego generowania wielopunktowych prototypów skupisk danych – w formie drzewopodobnych podstruktur neuronów wzajemnie powiązanych połączeniami topologicznymi – w tym do:

- automatycznego generowania po jednym prototypie dla każdego skupiska danych, co w rezultacie przekłada się na automatyczne wykrywanie liczby tych skupisk,
- automatycznego dostosowywania rozmiaru (liczby punktów) i kształtu każdego z prototypów, odpowiednio do kształtu i wielkości reprezentowanego przez niego skupiska oraz gęstości zawartych w nim danych,
- automatycznego dostosowywania położenia tych prototypów w przestrzeni danych, odpowiednio do lokalizacji skupisk.

### 3.3.1. Mechanizm usuwania pojedynczych, małoaktywnych neuronów ze struktury sieci neuronowej

Rozważana jest operacja usuwania pojedynczego $j$-tego neuronu ( $j \in \{1,2,...,m\}$ ) z topologicznej struktury sieci, którego aktywność mierzona liczbą zwycięstw $lzw_j$ po zakończeniu danej epoki uczenia jest niższa od zakładanego poziomu $lzw_{\min}$:

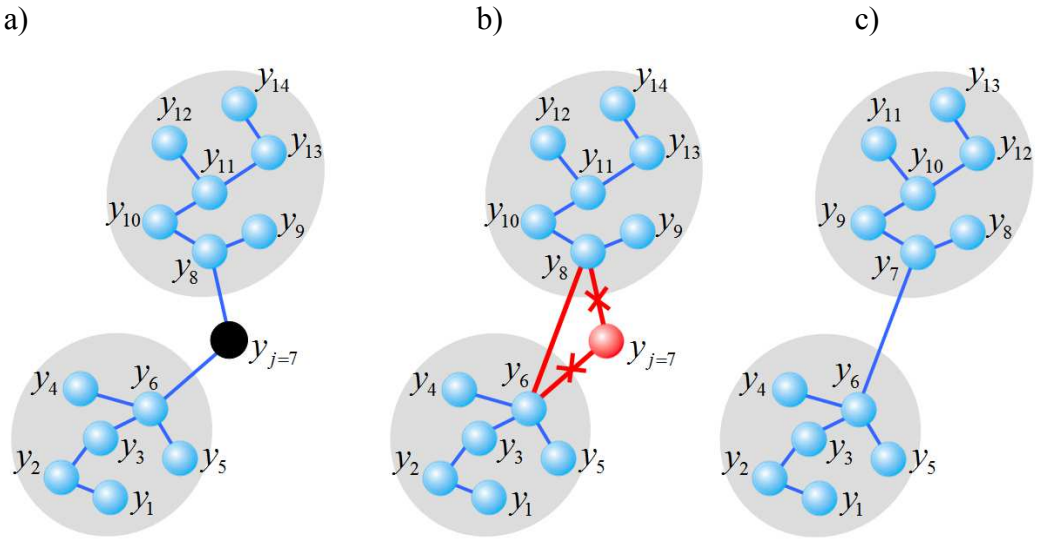
$$lzw_j < lzw_{\min}, \tag{3.1}$$

gdzie $lzw_{\min}$ jest dobierany eksperymentalnie (zwykle $lzw_{\min} \in \{2, 3, 4\}$ ). Po usunięciu $j$-tego neuronu, modyfikowana jest konfiguracja połączeń topologicznych wszystkich neuronów bezpośrednio z nim sąsiadujących w następujący sposób:

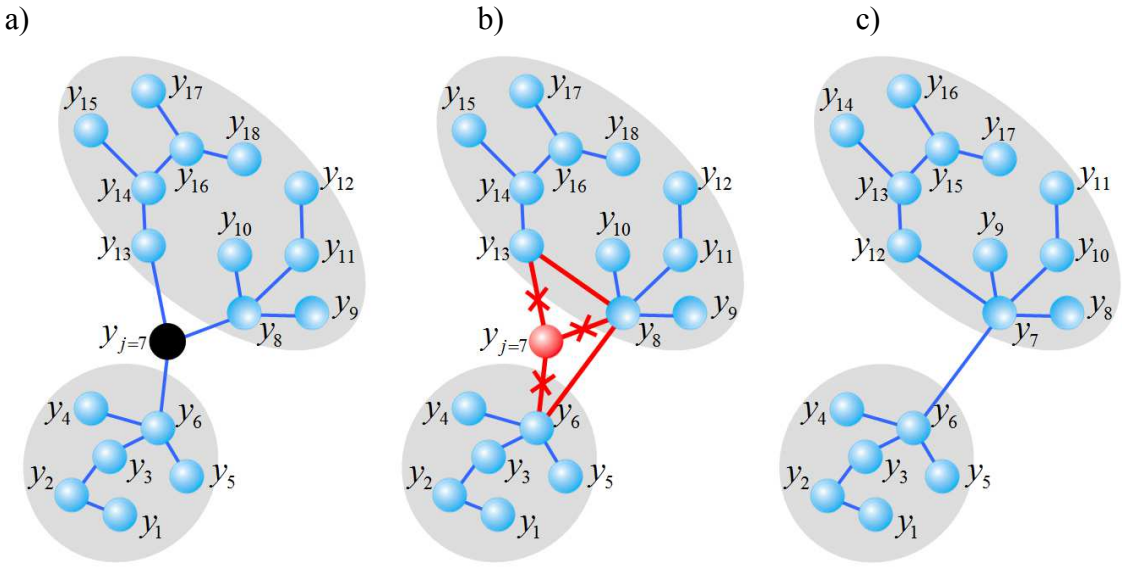
- jeżeli $j$-ty neuron miał jednego bezpośredniego sąsiada, wówczas połączenie topologiczne łączące oba neurony jest usuwane,

- jeżeli *j*-ty neuron miał dwóch bezpośrednich sąsiadów, wówczas połączenia topologiczne łączące go z sąsiadami są usuwane, a w ich miejsce wstawiane jest jedno nowe połączenie topologiczne pomiędzy jego sąsiadami (patrz rys. 3.8),
- jeżeli *j*-ty neuron miał trzech lub więcej sąsiadów, wówczas połączenia topologiczne łączące go z sąsiadami są usuwane, a w ich miejsce wstawiane są nowe połączenia pomiędzy jednym z sąsiadów, który był zlokalizowany najbliżej *j*-tego neuronu w przestrzeni danych, a pozostałymi sąsiadami (patrz rys. 3.9 ilustrujący przypadek, gdy *j*-ty neuron ma trzech sąsiadów).

Operacja zostaje zakończona po przenumerowaniu neuronów sieci. Mechanizm usuwania neuronów może być uaktywniony dla struktur (podstruktur) zawierających więcej niż 2 neurony (w przypadku mniejszych struktur neuronów – patrz podrozdział 3.3.3).




Rys. 3.8. Ilustracja operacji usuwania pojedynczego, mało aktywnego neuronu połączonego bezpośrednio z dwoma sąsiadami: struktura sieci z wytypowanym do usunięcia *j*-tym neuronem (oznaczonym kolorem czarnym; $j = 7$) (a), podczas usuwania neuronu i rekonfiguracji połączeń topologicznych (zmiany oznaczono na czerwono) (b) oraz po przenumerowaniu neuronów (c).



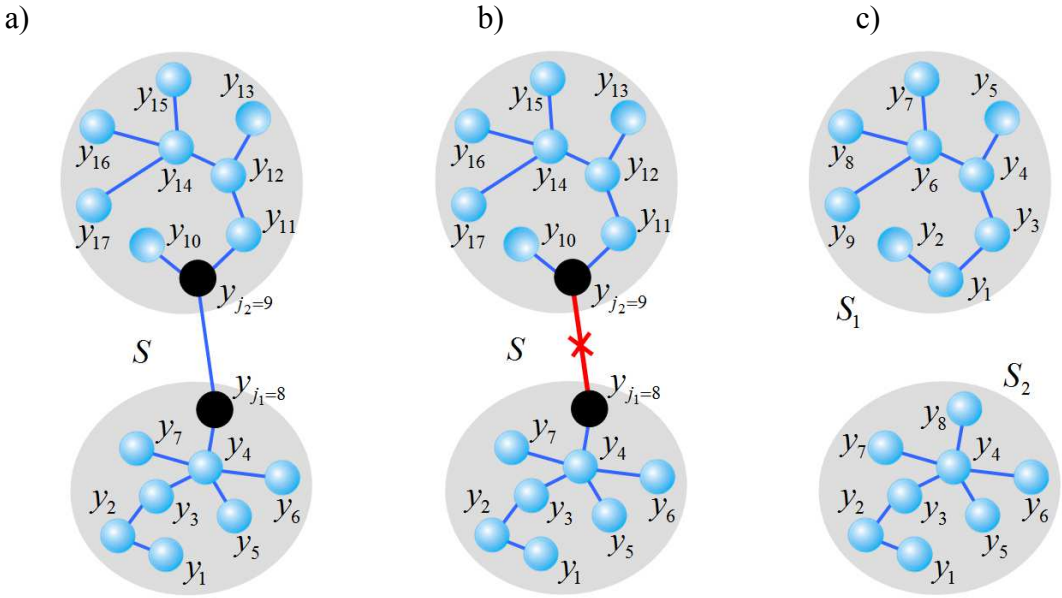

Rys. 3.9. Ilustracja operacji usuwania pojedynczego, mało aktywnego neuronu połączonego bezpośrednio z trzema sąsiadami: struktura sieci z wytypowanym do usunięcia $j$-tym neuronem (oznaczonym kolorem czarnym; $j = 7$) (a), podczas usuwania neuronu i rekonfiguracji połączeń topologicznych (zmiany oznaczono na czerwono) (b) oraz po przenumerowaniu neuronów (c).

### 3.3.2. Mechanizm usuwania pojedynczych połączeń topologicznych w celu podziału struktury sieci neuronowej na podstruktury

Rozważana jest operacja usuwania połączenia topologicznego łączącego dwa neurony o numerach $j_1$ oraz $j_2$ (patrz rys. 3.10), jeżeli spełniony jest warunek:

$$d(\boldsymbol{w}_{j_1}, \boldsymbol{w}_{j_2}) > \alpha_{rozł} d_{śr}, \tag{3.2}$$

gdzie $d_{śr} = \frac{1}{P}\sum_{p=1}^{P} d_p$ jest średnią arytmetyczną odległości $d_p$ pomiędzy dwoma bezpośrednio sąsiadującymi neuronami, liczoną dla wszystkich $P$ par takich neuronów, występujących w strukturze sieci, natomiast $\alpha_{rozł}$ jest eksperymentalnie dobieranym współczynnikiem (zwykle $\alpha_{rozł} \in [3, 4]$). W rezultacie operacji usuwania połączenia topologicznego, struktura sieci zostaje podzielona (rozłączona) na dwie podstruktury. Operacja zostaje zakończona po przenumerowaniu neuronów w obu podstrukturach. Mechanizm usuwania połączeń topologicznych może być uaktywniony dla struktur (podstruktur) zawierających więcej niż trzy neurony.



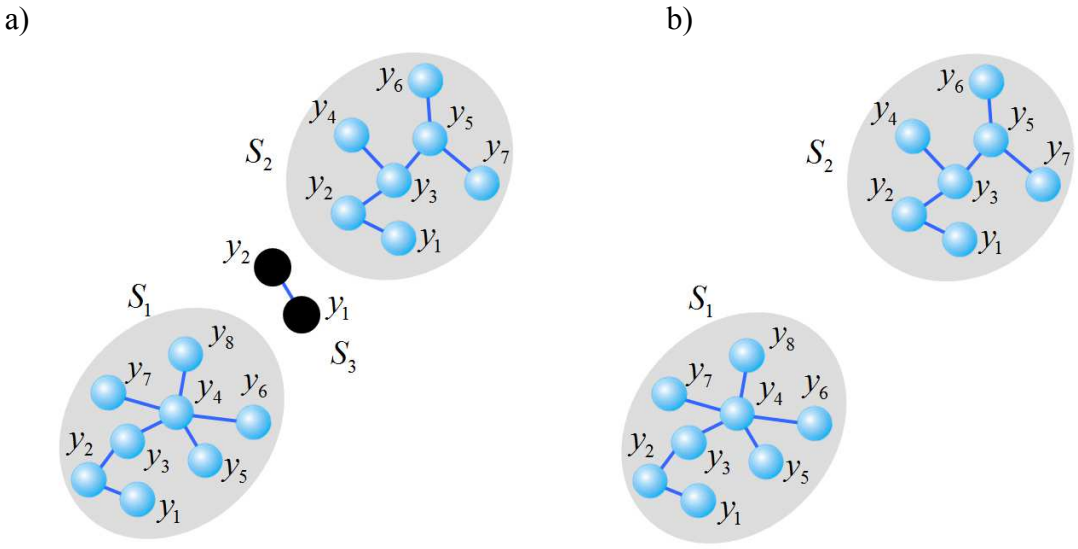

Rys. 3.10. Ilustracja operacji usuwania pojedynczego połączenia topologicznego: struktura sieci $S$ z neuronami o numerach $j_1$ oraz $j_2$ (oznaczonymi kolorem czarnym; $j_1 = 8$ i $j_2 = 9$), których połączenie zostało wytypowane do usunięcia (a), podczas usuwania połączenia (zmiany oznaczono na czerwono) (b) oraz struktura sieci po podziale na podstruktury $S_1$ i $S_2$ oraz po przenumerowaniu neuronów (c)

### 3.3.3. Mechanizm usuwania podstruktur sieci neuronowej, zawierających małą liczbę neuronów

Rozważana jest operacja usuwania podstruktury sieci neuronowej, zawierającej liczbę neuronów $m_s$ mniejszą od zakładanego poziomu $m_{\min}$ (patrz rys. 3.11):

$$m_s < m_{\min}, \tag{3.3}$$

gdzie $m_{\min}$ jest dobierany eksperymentalnie (zwykle $m_{\min} \in \{3, 4\}$). Podstruktury sieci zawierające niewielką liczbę neuronów są zbędne, gdyż zazwyczaj nie odwzorowują skupiska danych.




Rys. 3.11. Ilustracja operacji usuwania podstruktury sieci zawierającej małą liczbę neuronów: struktura sieci z wytypowaną do usunięcia podstrukturą (oznaczoną kolorem czarnym) (a) oraz struktura sieci po zakończeniu tej operacji (b)

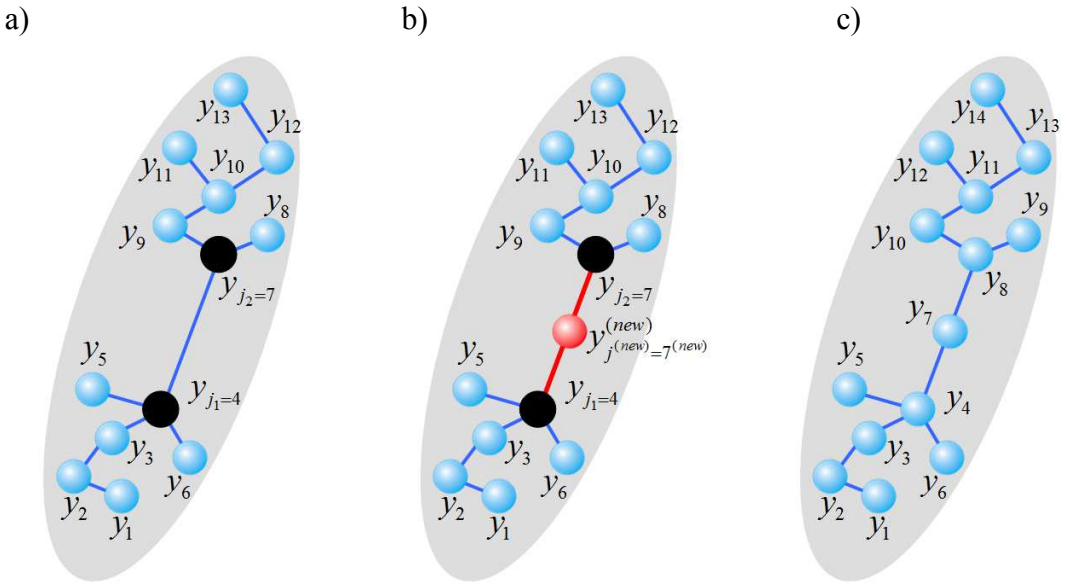
### 3.3.4. Mechanizm wstawiania nowych neuronów do struktury (podstruktury) sieci neuronowej w otoczeniu neuronów nadaktywnych

Mechanizm wstawiania nowych neuronów do struktury (podstruktury) sieci obejmuje dwie operacje. W przypadku pierwszej z nich, rozważane są dwa bezpośrednio sąsiadujące ze sobą neurony o numerach $j_1$ oraz $j_2$ ($j_1, j_2 \in \{1,2,...,m\}$, $j_1 \neq j_2$), których aktywność mierzona liczbą zwycięstw, odpowiednio, $lzw_{j_1}$ i $lzw_{j_2}$ po zakończeniu danej epoki uczenia jest wyższa od zakładanego poziomu $lzw_{\max}$:

$$lzw_{j_1} > lzw_{\max} \quad \text{i} \quad lzw_{j_2} > lzw_{\max}, \tag{3.4}$$

gdzie $lzw_{\max}$ jest dobierany eksperymentalnie (zwykle $lzw_{\max} \in \{2, 3, 4\}$). Nowy neuron o numerze $j^{(new)}$ jest wstawiany pomiędzy neurony $j_1$ oraz $j_2$, tak jak to pokazano na rys. 3.12, a jego wektor wagowy jest wyznaczany następująco:

$$\boldsymbol{w}_{j^{(new)}} = \frac{\boldsymbol{w}_{j_1} + \boldsymbol{w}_{j_2}}{2}. \tag{3.5}$$




Rys. 3.12. Ilustracja operacji wstawiania nowego neuronu do struktury (podstruktury) sieci pomiędzy dwa bardzo aktywne neurony: struktura sieci z wytypowanymi bardzo aktywnymi neuronami (oznaczonymi kolorem czarnym) (a), podczas wstawiania neuronu (zmiany oznaczono na czerwono) (b) oraz po przenumerowaniu neuronów (c)

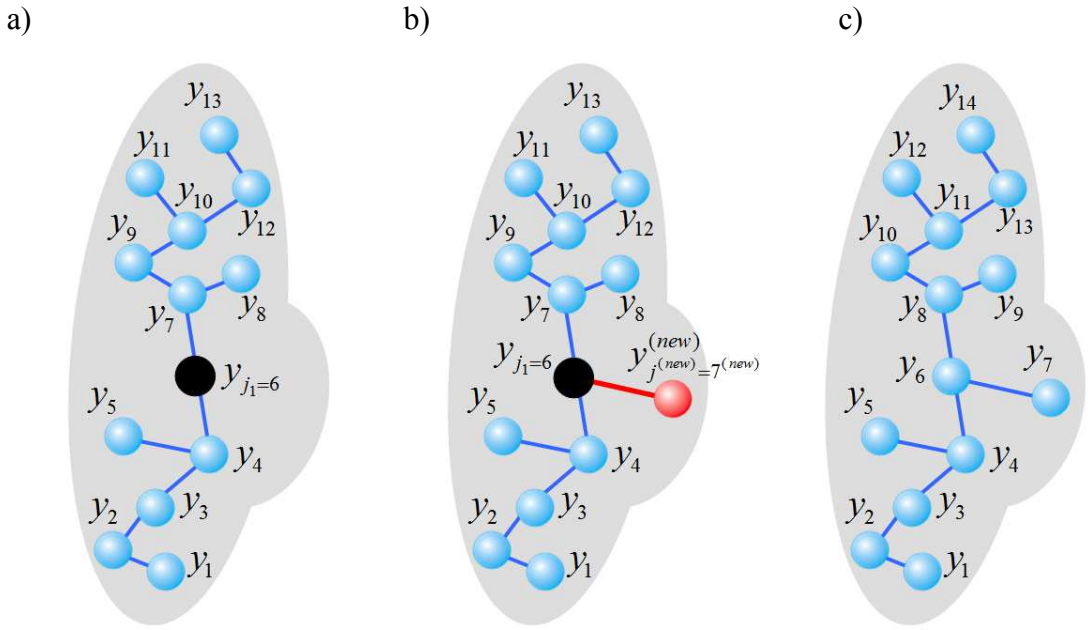
W przypadku drugiej operacji, rozważany jest jeden neuron o numerze $j_1$ ($j_1 \in \{1,2,...,m\}$) oraz wszystkie neurony o numerach $j \in \{1,2,...,m\}$ takie, że $d_{tpl}(j, j_1) = 1$, których aktywności mierzone liczbą zwycięstw, odpowiednio, $lzw_{j_1}$ oraz $lzw_j$ po zakończeniu danej epoki uczenia spełniają następujące warunki:

$$lzw_{j_1} > lzw_{\max} \quad \text{i} \quad lzw_j < lzw_{\max}, \tag{3.6}$$

gdzie $lzw_{\max}$ jest taki jak w warunku (3.4). Nowy neuron o numerze $j^{(new)}$ jest wstawiany w sąsiedztwie neuronu o numerze $j_1$, tak jak to pokazano na rys. 3.13, a jego wektor wagowy jest wyznaczany następująco:

$$\boldsymbol{w}_{j^{(new)}} = [w_{j^{(new)}1}, w_{j^{(new)}2}, ..., w_{j^{(new)}n}]^T, \; w_{j^{(new)}i} = w_{j_1 i}(1 + \varepsilon_i), \tag{3.7}$$

gdzie $\varepsilon_i$ jest liczbą losowaną z przedziału [-0.01, 0.01]. Obie operacje zostają zakończone po przenumerowaniu neuronów. W kolejnych epokach procesu uczenia sieci, nowy neuron przejmuje część aktywności swojego nadaktywnego sąsiada, co prowadzi do wyrównania ich aktywności (w efekcie, reprezentują one porównywalne liczby próbek danych).



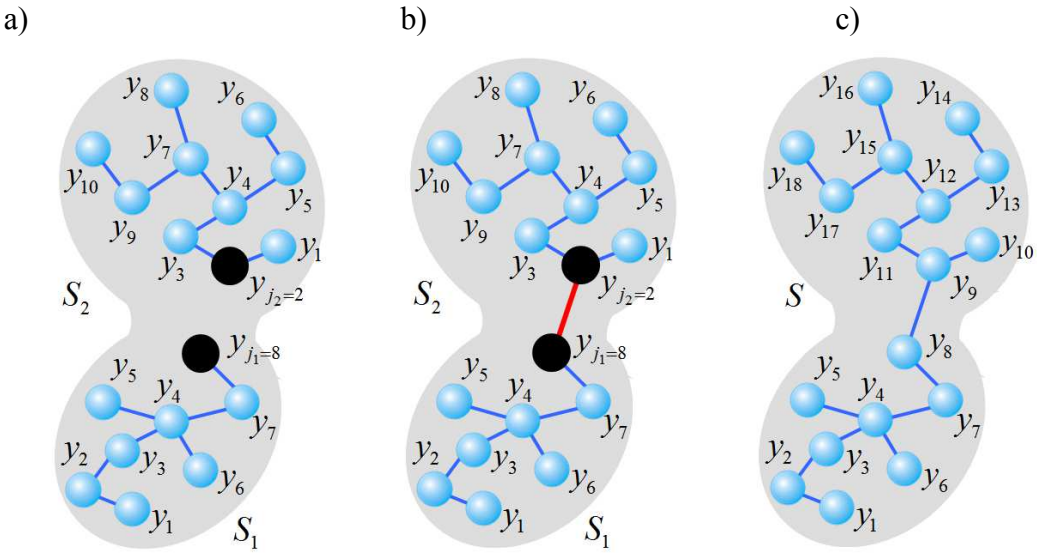

Rys. 3.13. Ilustracja operacji wstawiania nowego neuronu do struktury (podstruktury) sieci w sąsiedztwie bardzo aktywnego neuronu: struktura sieci z wytypowanym bardzo aktywnym neuronem (oznaczonym kolorem czarnym) (a), podczas wstawiania neuronu (zmiany oznaczono na czerwono) (b) oraz po przenumerowaniu neuronów (c)

### 3.3.5. Mechanizm wstawiania nowych połączeń topologicznych w celu ponownego łączenia podstruktur sieci neuronowej

Rozważana jest operacja wstawienia nowego połączenia topologicznego pomiędzy dwoma neuronami $j_1$ oraz $j_2$ należącymi do dwóch różnych podstruktur sieci, odpowiednio, $S_1$ i $S_2$ (patrz rys. 3.14), jeżeli spełniony jest następujący warunek:

$$d(\boldsymbol{w}_{j_1}, \boldsymbol{w}_{j_2}) < \alpha_{łącz} \frac{d_{śrS_1} + d_{śrS_2}}{2}, \tag{3.8}$$

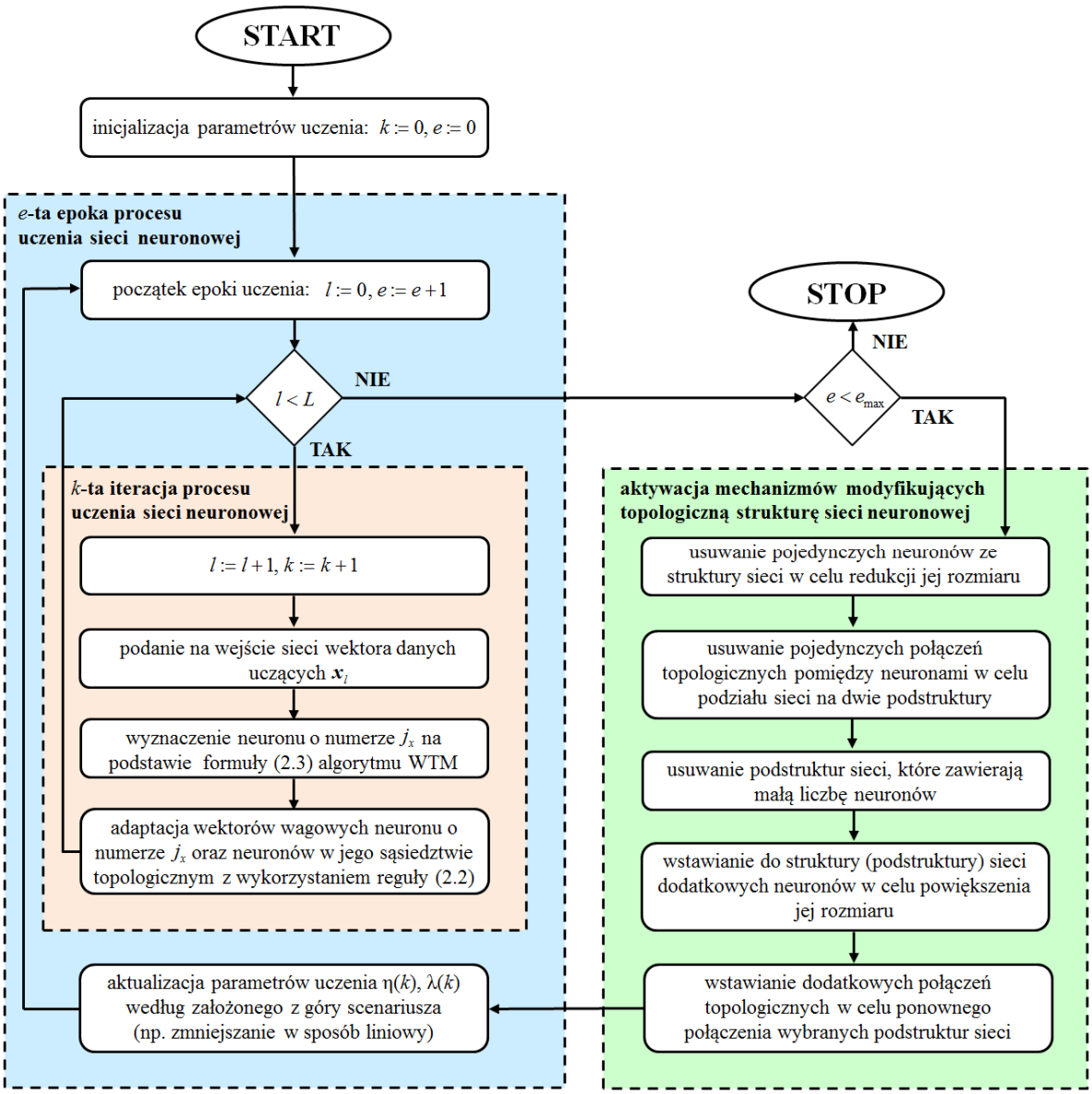
gdzie $d_{śrS_1}$ oraz $d_{śrS_2}$ są obliczane dla podstruktur, odpowiednio, $S_1$ i $S_2$, w taki sam sposób jak $d_{śr}$ w warunku (3.2), natomiast $\alpha_{łącz}$ jest eksperymentalnie dobieranym współczynnikiem (zwykle $\alpha_{łącz} \in [3, 4]$).




Rys. 3.14. Ilustracja operacji wstawiania nowego połączenia topologicznego: podstruktury sieci $S_1$ i $S_2$ z neuronami o numerach $j_1$ oraz $j_2$ (oznaczonymi kolorem czarnym; $j_1 = 8$ i $j_2 = 2$), które zostały wytypowane do połączenia (a), podczas wstawiania połączenia (zmiany oznaczono na czerwono) (b) oraz struktura sieci $S$ po połączeniu i przenumerowaniu neuronów (c).

## 3.4. Algorytm uczenia uogólnionej samoorganizującej się sieci neuronowej o drzewopodobnej strukturze

Algorytm uczenia proponowanej sieci neuronowej został przedstawiony na rys. 3.15 w postaci schematu blokowego. Proces uczenia sieci poprzedza faza inicjalizacji parametrów uczenia, takich jak: początkowa liczba neuronów struktury sieci ($m_{\text{start}}$), liczba epok uczenia ($e_{\max}$), rodzaj funkcji sąsiedztwa topologicznego neuronów sieci $S(j, j_x, k)$, początkowe wartości współczynnika uczenia sieci ($\eta(e=1) = \eta_{\text{start}}$) oraz promienia sąsiedztwa topologicznego ($\lambda(e=1) = \lambda_{\text{start}}$), jak również wartości parametrów $lzw_{\min}$, $lzw_{\max}$, $m_{\min}$, $\alpha_{rozl}$ i $\alpha_{łącz}$, odpowiedzialnych za funkcjonowanie mechanizmów usuwania i wstawiania neuronów, usuwania podstruktur o niewielkich rozmiarach oraz usuwania i wstawiania połączeń topologicznych. Kolejne fazy uczenia obejmują proces adaptacji wektorów wagowych sieci z wykorzystaniem klasycznego algorytmu WTM (ang. *Winner Takes Most*) oraz proces modyfikacji topologicznej struktury sieci z wykorzystaniem przedstawionych wyżej mechanizmów, uruchamiany po zakończeniu każdej kolejnej epoki uczenia.




Rys. 3.15. Algorytm uczenia samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej

Szczególną uwagę warto zwrócić na sposób aktualizacji współczynnika uczenia sieci ($\eta(e)$) oraz promienia sąsiedztwa topologicznego ($\lambda(e)$). Zalecane jest [68], aby ich wartości były stałe dla danej epoki procesu uczenia, natomiast przed rozpoczęciem każdej kolejnej epoki (z wyjątkiem pierwszej) były zmniejszane np. w sposób liniowy, aż do osiągnięcia w ostatniej epoce wartości minimalnych (odpowiednio $\eta(e = e_{\max}) = \eta_{\text{stop}}$ i $\lambda(e = e_{\max}) = \lambda_{\text{stop}}$). Taka strategia ma za zadanie zmniejszenie stopnia adaptacji wektorów wagowych i stabilizowanie struktury sieci w końcowym etapie procesu uczenia. We wszystkich eksperymentach przedstawionych w pracy, w

przypadku promienia sąsiedztwa $\lambda$, przyjęto odmienną strategię, zakładającą stały promień sąsiedztwa $\lambda = 2$ przez cały czas trwania procesu uczenia (co było wystarczające do osiągnięcia stabilizacji sieci neuronowej pod koniec jej uczenia).

## 3.5. Podsumowanie

W niniejszym rozdziale przedstawiono konstrukcję oryginalnej, uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej wyposażonej w mechanizmy jej rozłączania na podstruktury, ponownego łączenia wybranych podstruktur oraz automatycznej regulacji liczby neuronów sieci. Mechanizmy te umożliwiają topologicznej strukturze sieci przyjmowanie złożonych, drzewopodobnych form geometrycznych „wpasowujących się", w procesie uczenia, w różnorodne skupiska danych w rozważanym zbiorze. Dzięki temu, proponowana sieć neuronowa jest w stanie: a) automatycznie generować po jednym wielopunktowym prototypie (w postaci określonej podstruktury sieci) dla każdego skupiska danych, co oznacza również wykrywanie liczby tych skupisk, b) automatycznie dostosowywać położenie tych prototypów w przestrzeni danych, odpowiednio do lokalizacji skupisk oraz c) automatycznie „dostrajać" liczbę punktów każdego z prototypów, stosownie do kształtu, wielkości oraz gęstości danych reprezentowanego przez niego skupiska.

W następnym rozdziale zostanie przedstawiony test funkcjonowania proponowanej sieci neuronowej w różnorodnych kategoriach problemów grupowania danych, z wykorzystaniem szeregu standardowych zbiorów danych. W kolejnych rozdziałach proponowana sieć neuronowa zostanie wykorzystana w złożonych problemach grupowania danych opisujących ekspresję genów w przypadku różnych chorób nowotworowych.

# 4. Grupowanie danych w standardowych zbiorach danych z wykorzystaniem proponowanych sieci neuronowych

Rozdział ten przedstawia praktyczny test funkcjonowania proponowanej, uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej. Do testów wykorzystano 7 syntetycznych zbiorów danych dwuwymiarowych i 3 zbiory danych trójwymiarowych, reprezentujące wybrane kategorie problemów grupowania danych (w podrozdziale 4.1) oraz 3 zbiory danych wielowymiarowych, pochodzące z serwera Uniwersytetu Kalifornijskiego w Irvine (http://archive.ics.uci.edu/ml) i reprezentujące rzeczywiste problemy grupowania danych (podrozdział 4.2).

## 4.1. Grupowanie danych w syntetycznych dwu- i trójwymiarowych zbiorach danych

Tabela 4.1 zawiera zestawienie informacji o syntetycznych zbiorach danych, wykorzystywanych w tym podrozdziale do testów proponowanej sieci neuronowej. Zbiory danych nr 1, 2, 3 i 7 zostały opracowane we własnym zakresie i reprezentują: a) rozważane w podrozdziale 3.1, szczególne przypadki problemów wykrywania pojedynczych skupisk danych równomiernie rozmieszczonych na płaszczyźnie (zbiór danych nr 1) oraz rozmieszczonych na obszarze w kształcie krzyża (zbiór danych nr 2), b) problemy wykrywania większej liczby skupisk danych w postaci wzajemnie skręconych spiral (zbiór danych nr 3) oraz o różnorodnych, złożonych kształtach o charakterze objętościowym i szkieletowym (zbiór danych nr 7). Pozostałe zbiory danych zostały zaczerpnięte z literatury. Zbiór nr 6, rozważany w [104], reprezentuje problem grupowania danych w zbiorze zawierającym 5 skupisk danych o różnych kształtach i wielkościach (dwa objętościowe, dwa o kształcie pierścieniowym oraz jedno w kształcie krzywej sinusoidalnej). Pięć kolejnych zbiorów danych, rozważanych w pracy [115], pochodzi z tzw. pakietu podstawowych problemów grupowania danych FCPS (ang. *Fundamental Clustering Problem Suite*). Parametry sterujące procesem uczenia proponowanych sieci neuronowych dla wszystkich rozważanych w tym podrozdziale eksperymentów numerycznych przedstawia tabela 4.2 (minimalnie skorygowano jedynie jeden z tych parametrów w przypadku eksperymentów przedstawionych w rozdziale 4.2).

Tabela 4.1. Zestawienie informacji o rozważanych syntetycznych zbiorach danych

| Lp. | Charakterystyka skupisk w zbiorze danych (nazwa zbioru danych i źródło) | Wymiarowość danych | Liczba próbek danych | Liczba skupisk danych |
|---|---|---|---|---|
| 1 | Zbiór danych równomiernie rozmieszczonych na płaszczyźnie | 2 | 900 | 1 |
| 2 | Skupisko danych w kształcie krzyża | 2 | 300 | 1 |
| 3 | Skupiska danych w kształcie wzajemnie skręconych spiral | 2 | 1000 | 2 |
| 4 | Romboidalne skupiska danych, stykające się ze sobą (zbiór danych *TwoDiamonds* [115]) | 2 | 800 | 2 |
| 5 | Objętościowe skupiska danych o różnych gęstościach (zbiór danych *Lsun* [115]) | 2 | 400 | 3 |
| 6 | Skupiska danych o różnorodnych rozmiarach i kształtach [104] | 2 | 1690 | 5 |
| 7 | Skupiska danych o złożonych kształtach szkieletowych i objętościowych | 2 | 1900 | 7 |
| 8 | Skupisko danych równomiernie rozmieszczonych na sferze (zbiór danych *GolfBall* [115]) | 3 | 4002 | 1 |
| 9 | Skupiska danych o kształtach sferycznych, jedno wewnątrz drugiego (zbiór danych *Atom* [115]) | 3 | 800 | 2 |
| 10 | Skupiska danych w postaci połączonych pierścieni (zbiór danych *Chainlink* [115]) | 3 | 1000 | 2 |

### 4.1.1. Zbiór danych rozmieszczonych równomiernie na płaszczyźnie

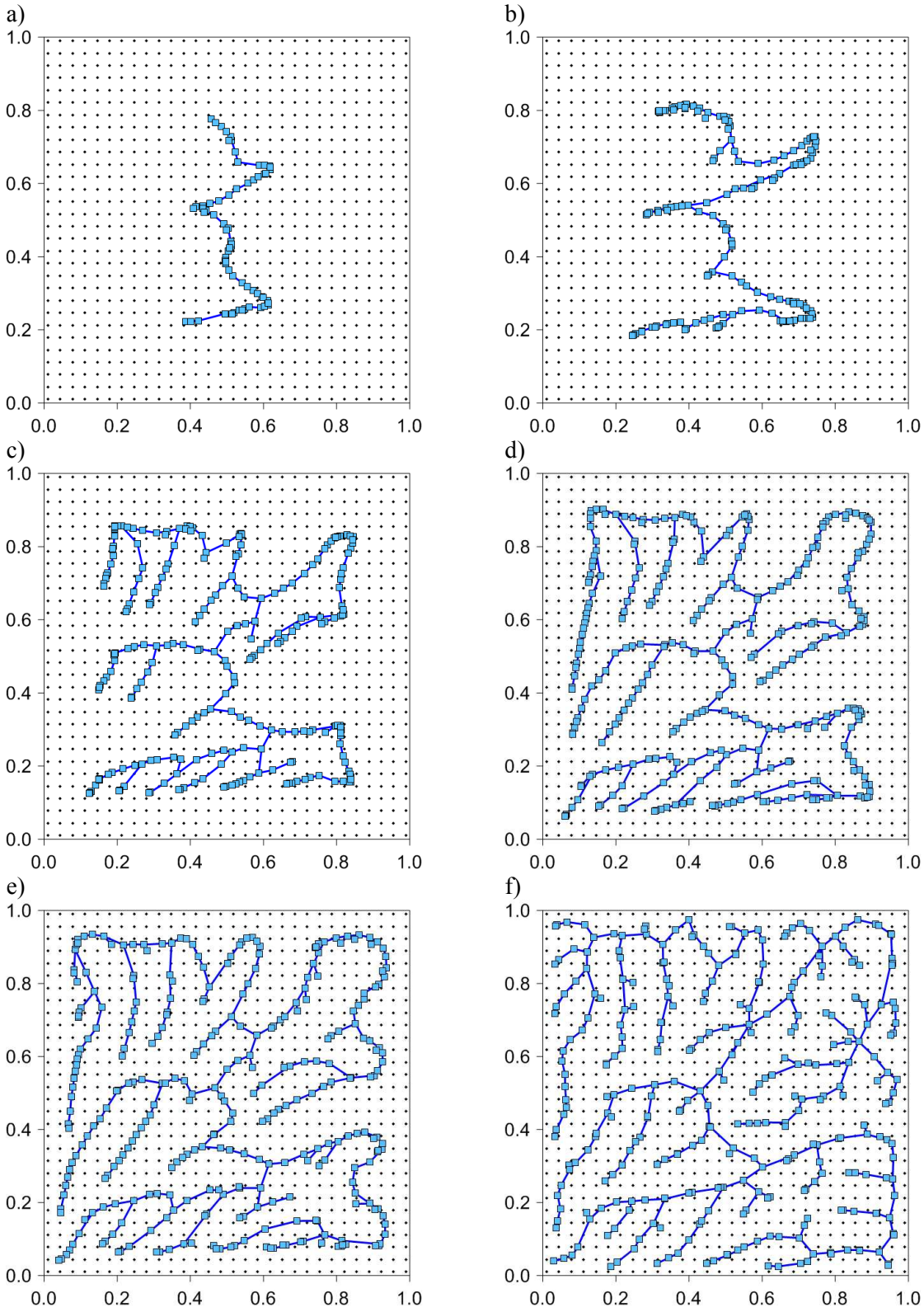
Pierwszy zbiór danych reprezentuje problem rozważany w podrozdziale 3.1, dla którego sieć DSOM (będąca punktem wyjścia w konstrukcji proponowanej w tej pracy sieci neuronowej o drzewopodobnej strukturze) dała błędne wyniki, tzn. wykryła dwa skupiska danych zamiast jednego. Tym razem, sieć neuronowa prawidłowo wykryła jedno skupisko danych obejmujące cały zbiór, co zilustrowano na rys. 4.1, przedstawiającym rozłożenie topologicznej struktury sieci w przestrzeni danych, w wybranych epokach procesu uczenia. Wykrycie jednego skupiska obejmującego cały zbiór może być interpretowane jako stwierdzenie braku lokalnych skupisk w tym zbiorze. Sposób inicjowania początkowej struktury sieci neuronowej jest analogiczny jak na rys. 3.2a i nie będzie tu (oraz w kolejnych eksperymentach) przedstawiany. Na rys. 4.2 widoczne są przebiegi zmian liczby neuronów sieci (ustabilizowanej na poziomie 307 neuronów po

zakończeniu uczenia – rys. 4.2a) oraz liczby jej podstruktur (ustabilizowanej na poziomie równym 1 – rys. 4.2b) podczas uczenia.

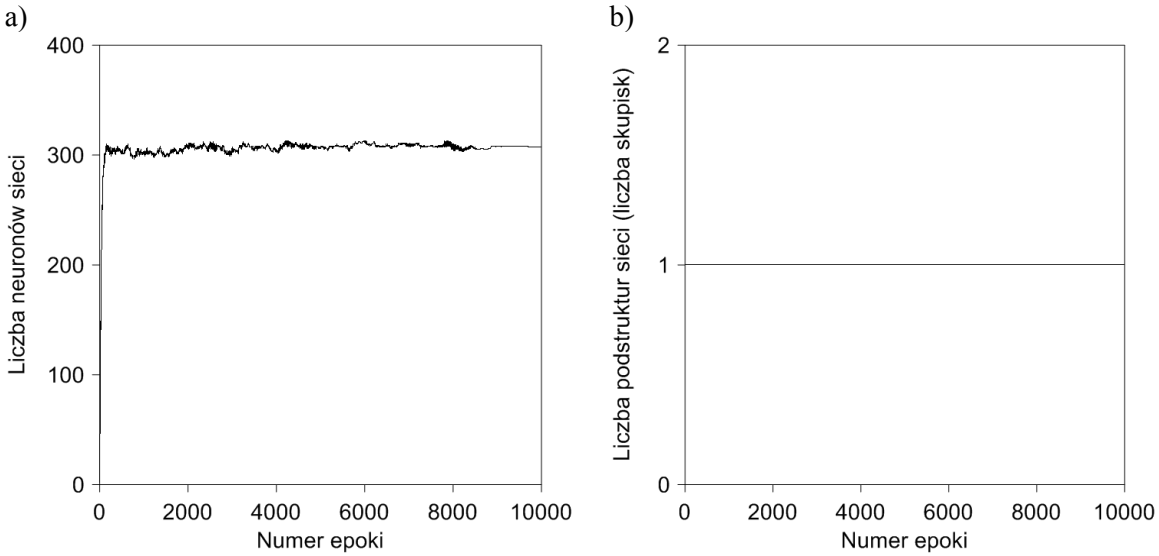
Tabela 4.2. Parametry proponowanych sieci neuronowych wykorzystywanych w eksperymentach numerycznych przedstawionych w rozdziale 4 (z minimalną korektą parametru $\alpha_{rozł}$ w rozdziale 4.2)

| Nazwa parametru/funkcji | Wartość (rodzaj) parametru |
|---|---|
| początkowa liczba neuronów sieci | $m_{start} = 2$ |
| maksymalna liczba epok uczenia | $e_{\max} = 10\,000$ |
| sposób wyznaczania neuronu zwyciężającego (o numerze $j_x$) | zależność (2.3) z euklidesową miarą odległości (2.4) |
| współczynnik uczenia sieci[1)] w (2.2) | $\eta(e) = 7 \cdot 10^{-4} \div 10^{-6}$ |
| funkcja sąsiedztwa topologicznego $S(j, j_x, k)$ w (2.2) | funkcja sąsiedztwa gaussowskiego (2.9) z miarą odległości Czebyszewa (2.6) |
| promień sąsiedztwa topologicznego w (2.9) | $\lambda(e) = 2$ |
| parametr odpowiedzialny za usuwanie pojedynczych mało aktywnych neuronów | $lzw_{\min} = 2$ |
| parametr odpowiedzialny za wstawianie dodatkowych neuronów do struktury | $lzw_{\max} = 4$ |
| parametr regulujący mechanizm usuwania małych podstruktur sieci | $m_{\min} = 3$ |
| parametr regulujący mechanizm usuwania połączeń topologicznych | $\alpha_{rozł} = 4$ |
| parametr regulujący mechanizm wstawiania połączeń topologicznych | $\alpha_{łącz} = 4$ |

[1)] parametr malejący liniowo po zakończeniu kolejnych epok uczenia *e*; stały dla danej epoki.



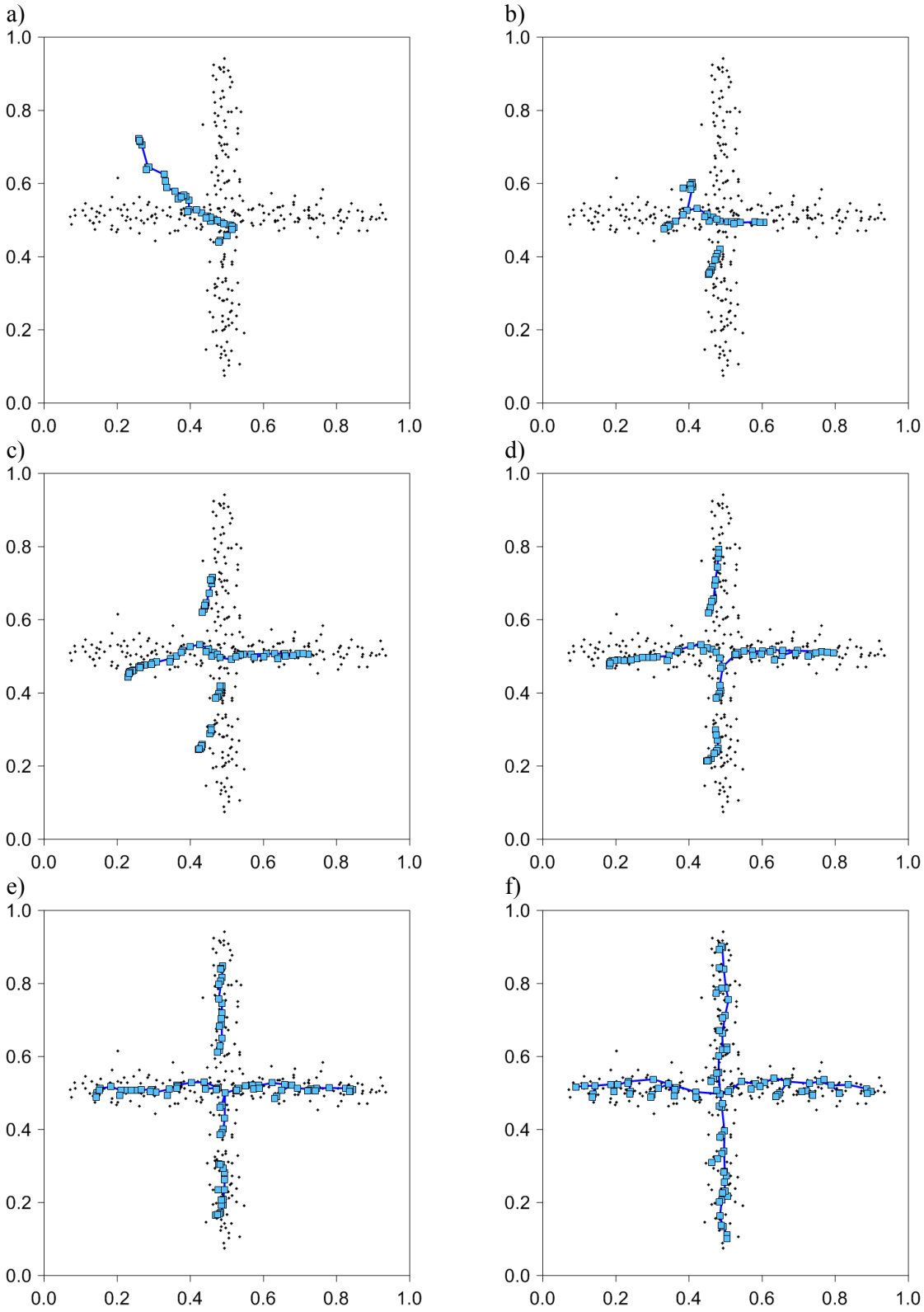

Rys. 4.1. Syntetyczny zbiór danych nr 1 (patrz tabela 4.1 i rys. 3.1a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (a), $e = 20$ (b), $e = 50$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)



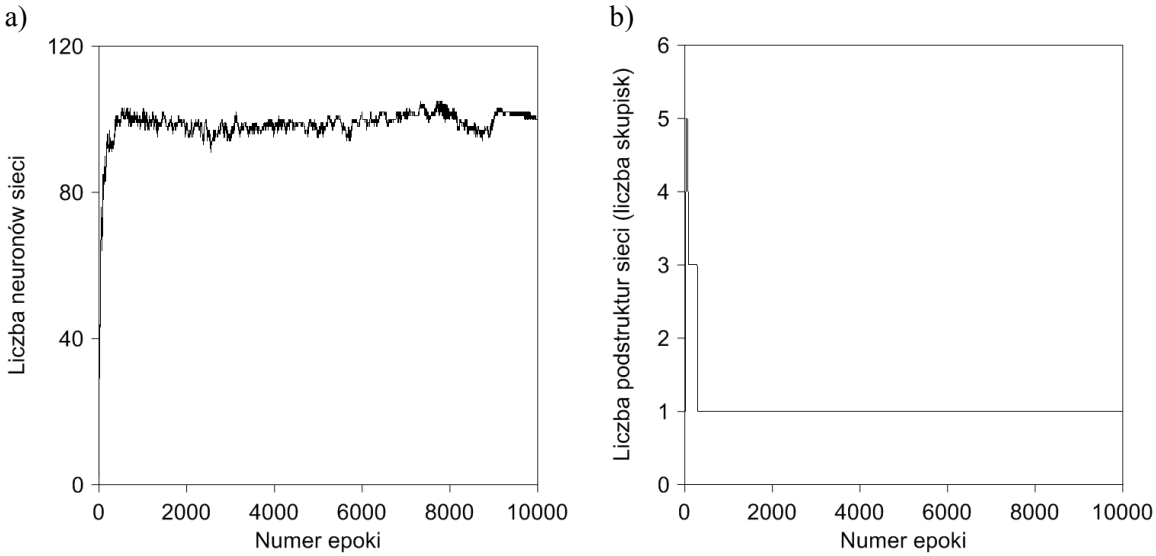

Rys. 4.2. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.1

### 4.1.2. Zbiór danych rozmieszczonych na obszarze w kształcie krzyża

Kolejny zbiór danych reprezentuje drugi z problemów rozważanych w podrozdziale 3.1, dla którego DSOM nie była w stanie poprawnie wykryć faktu, że zbiór ten zawiera jedno skupisko danych. Również i tym razem, proponowana sieć neuronowa o drzewopodobnej strukturze prawidłowo wykryła jedno skupisko danych. Na rys. 4.3 widoczne są wybrane epoki procesu uczenia, natomiast na rys. 4.4 przedstawiono przebiegi zmian liczby neuronów sieci oraz liczby jej podstruktur w trakcie trwania procesu uczenia. Po rozpoczęciu uczenia, początkowo sieć neuronowa podzieliła się na pięć podstruktur, które stosunkowo szybko (już po 200-nej epoce uczenia) ponownie połączyły się ze sobą w jedna strukturę (rys. 4.4b), odpowiadającą prawidłowej liczbie skupisk danych w rozważanym zbiorze danych. Po zakończeniu uczenia, struktura sieci zawierająca automatycznie dobrana liczbę 100 neuronów bardzo dobrze odwzorowuje szkieletowy kształt skupiska danych (patrz rys. 4.3f). Eksperyment ten jest przykładową ilustracją procesu formowania wielopunktowego prototypu pojedynczego skupiska danych o kształcie szkieletowym w trakcie trwania procesu grupowania danych (uczenia sieci). Położenie i kształt tego prototypu w przestrzeni danych są dostosowywane automatycznie do położenia i kształtu obszaru zajmowanego przez skupisko danych. Liczba punktów prototypu jest dostosowana automatycznie do rozmiaru skupiska danych. Problem odwzorowania skupisk o kształtach szkieletowych będzie jeszcze szerzej poruszony w dalszej części rozdziału.




Rys. 4.3. Syntetyczny zbiór danych nr 2 (patrz tabela 4.1 i rys. 3.1b) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (a), $e = 20$ (b), $e = 50$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

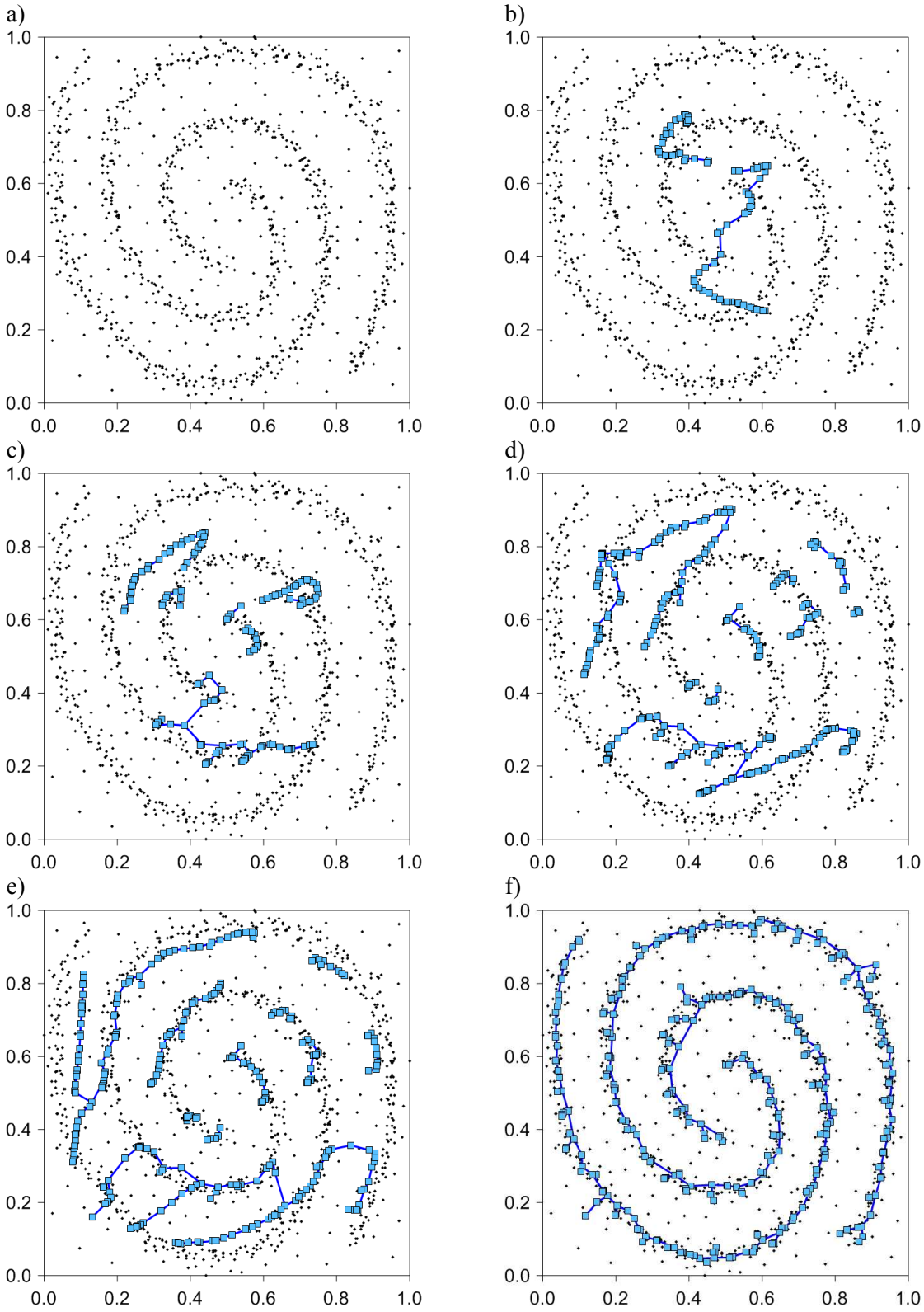



Rys. 4.4. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.3

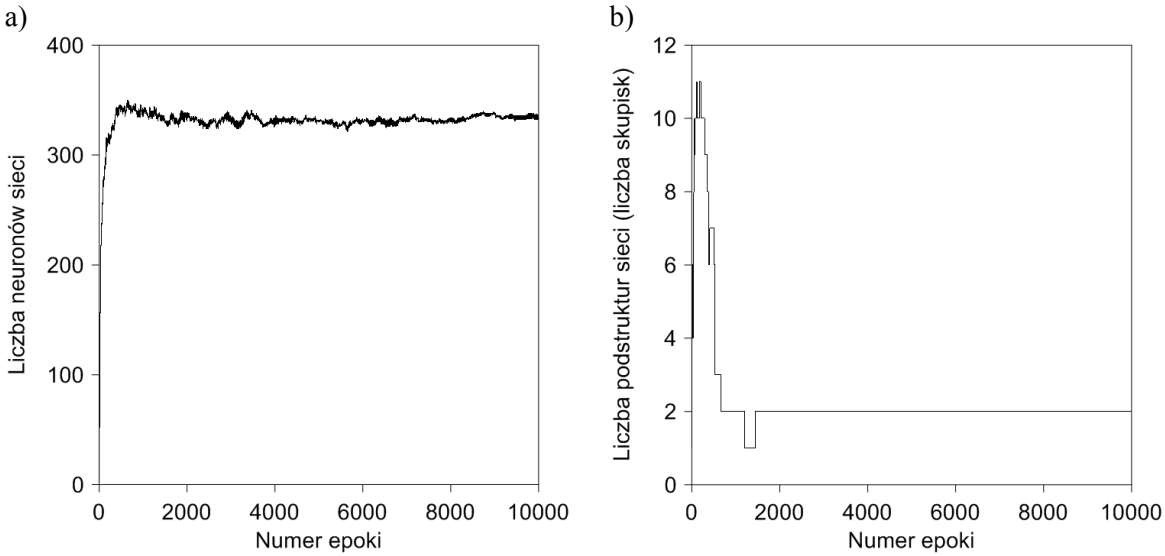
### 4.1.3. Zbiór danych ze skupiskami w kształcie wzajemnie skręconych spiral

W ramach trzeciego testu, rozważany jest zbiór danych zawierający dwa skupiska w kształcie wzajemnie skręconych spiral (rys. 4.5a). Test ten reprezentuje kategorie problemów wykrywania skupisk danych liniowo nieseparowalnych i jest jednym z trudniejszych zadań dla technik grupowania danych. Konwencjonalne podejścia zdolne do generowania jednopunktowych prototypów skupisk danych (np. algorytm *k-means*) z oczywistych względów nie są w stanie wykryć poprawnie skupisk danych w tym zbiorze, nawet jeżeli ich liczba zostanie podana z góry. Jedynie podejścia generujące wielopunktowe prototypy skupisk danych, takie jak sieć neuronowa DSOM, czy proponowana w tej pracy sieć o drzewopodobnej strukturze topologicznej mogą skutecznie rozwiązać to zadanie (zastosowanie DSOM do rozwiązania rozważanego problemu można znaleźć w [46, 47]).

Rys. 4.5 przedstawia rozłożenie struktury proponowanej sieci neuronowej w obszarze danych, w wybranych epokach procesu uczenia. Zmiany liczby neuronów oraz liczby podstruktur w trakcie uczenia widoczne są na rys. 4.6. W rezultacie uczenia, struktura sieci podzieliła się na dwie podstruktury (rys. 4.5f), z których każda reprezentuje jedno skupisko danych. Liczba tych podstruktur została ustabilizowana na prawidłowym poziomie równym 2, już po upływie ok.15% czasu uczenia. Ostateczna liczba neuronów sieci (366 neuronów) została automatycznie „dostrojona” do rozmiaru skupisk.




Rys. 4.5. Syntetyczny zbiór danych nr 3 (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

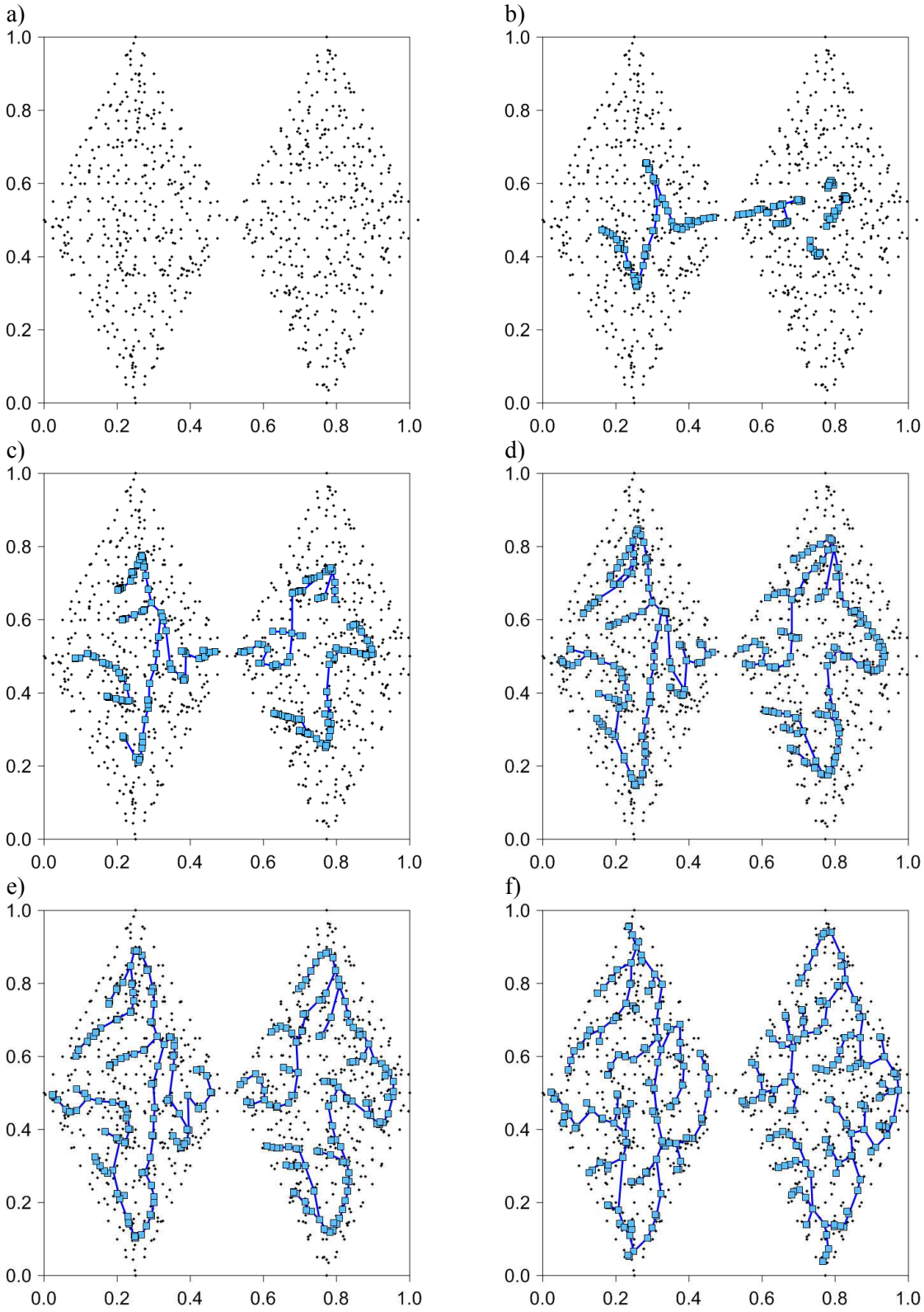



Rys. 4.6. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.5

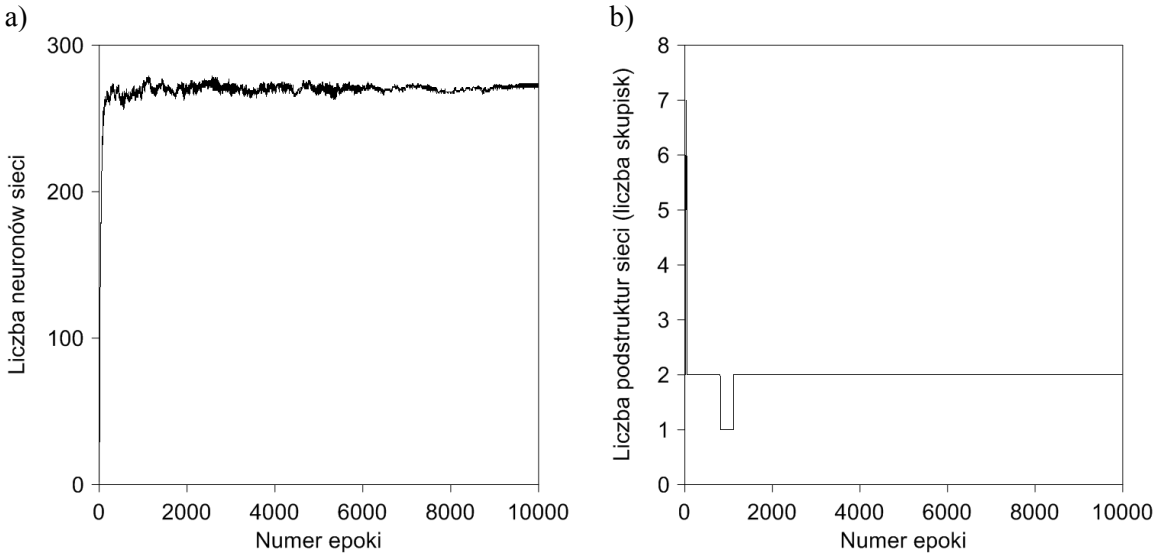
### 4.1.4. Zbiór danych *TwoDiamonds* [115] zawierający stykające się romboidalne skupiska danych

Zbiór danych *TwoDiamods* jest pierwszym z pięciu rozważanych w niniejszej pracy zbiorów danych, pochodzących z pakietu podstawowych problemów grupowania danych FCPS (ang. *Fundamental Clustering Problem Suite*) [115]. Zawiera 800 próbek danych, zlokalizowanych w przestrzeni danych w dwóch obszarach o kształtach rombów (po 400 próbek w każdym obszarze). Obszary danych stykają się ze sobą jednym z narożników, tak jak to jest pokazane na rys. 4.7a. Dla metod generujących wielopunktowe prototypy skupisk danych, rozważany problem jest testem odporności na tendencję do łączenia – w okolicy styku skupisk danych – dwóch prototypów reprezentujących te skupiska, w jeden większy prototyp.

Rozłożenie struktury sieci w obszarze danych w wybranych epokach procesu uczenia zilustrowano na rys. 4.7. Prototypy obu skupisk danych zostały wyznaczone prawidłowo i nie są połączone ze sobą w okolicy styku tych skupisk (rys. 4.7f). Zmiany liczby podstruktur sieci oraz liczby neuronów (ich ostateczna liczba wynosi 274) podczas uczenia przedstawiono na rys. 4.8. Można zauważyć, że podstruktury sieci tylko raz połączyły się ze sobą (ok. 1000-nej epoki), lecz połączenie to trwało stosunkowo krótko (ok. 100 epok). Test potwierdził odporność proponowanej metody na tendencję do łączenia prototypów reprezentujących rozważane skupiska danych.




Rys. 4.7. Syntetyczny zbiór danych nr 4 – *TwoDiamonds* [115] (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

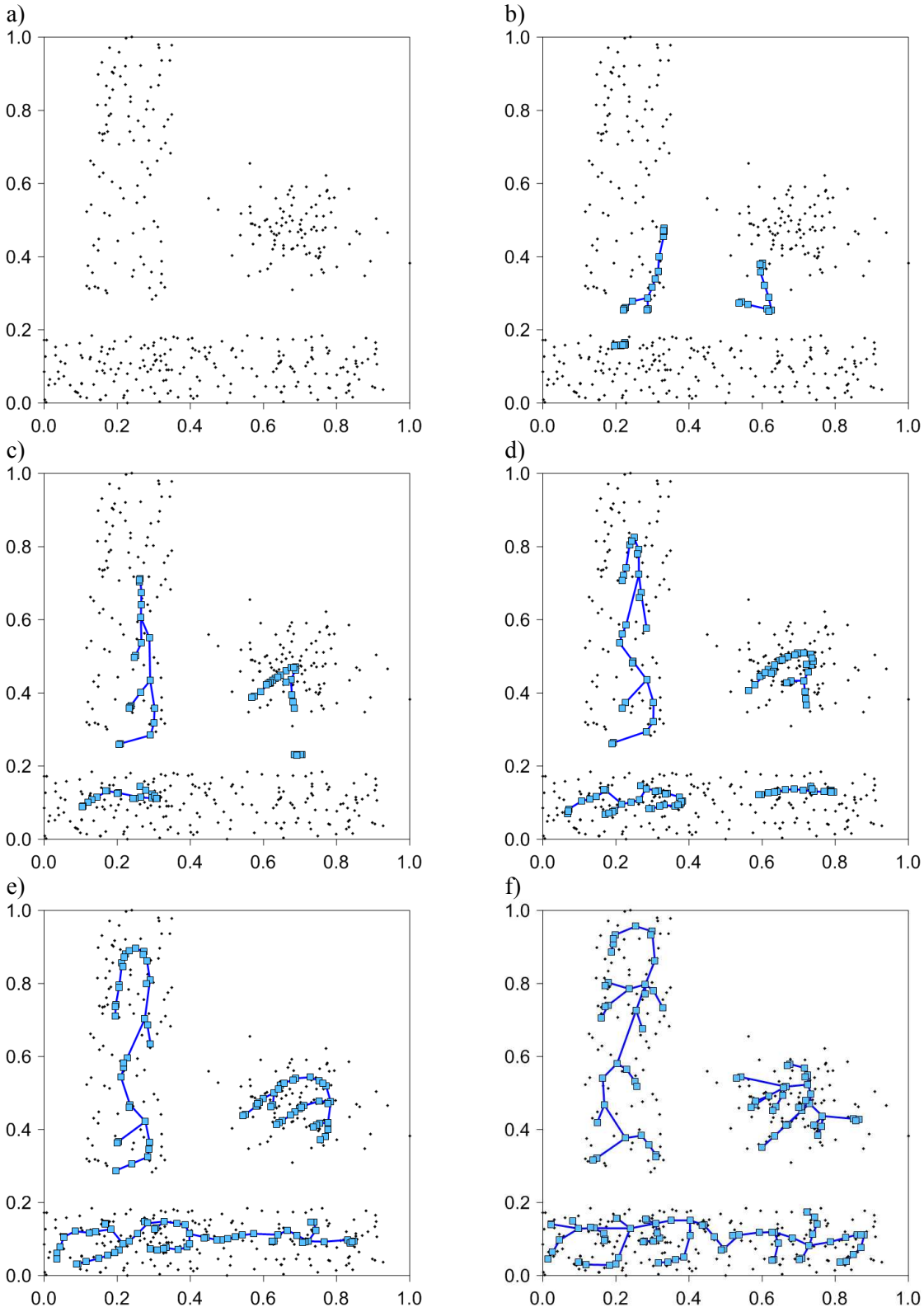



Rys. 4.8. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.7

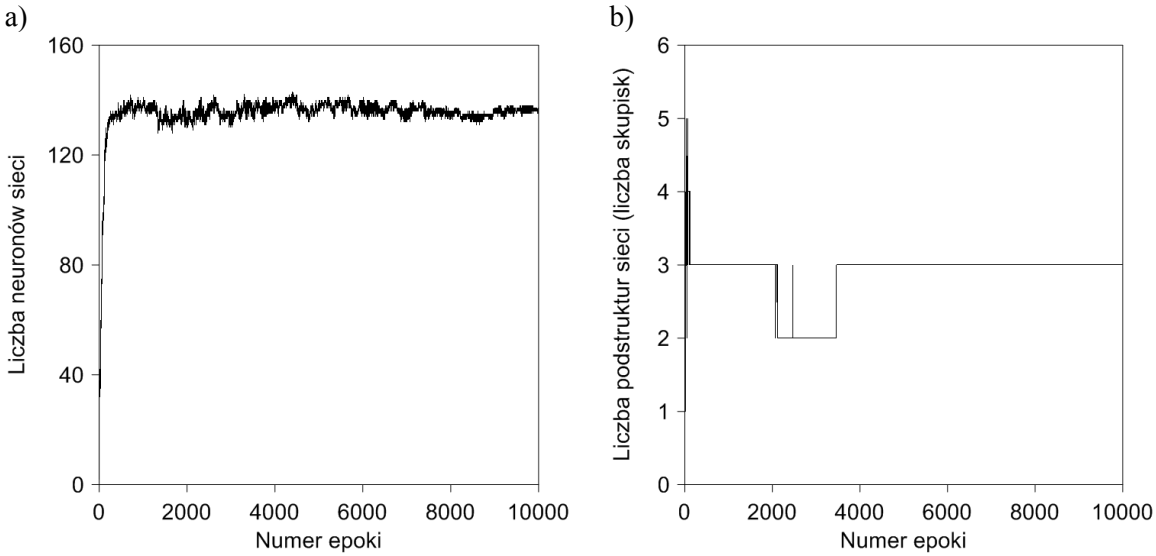
### 4.1.5. Zbiór danych *Lsun* [115] zawierający objętościowe skupiska danych o różnych gęstościach

Drugi ze zbiorów danych – o nazwie *Lsun* – należących do pakietu FCPS (ang. *Fundamental Clustering Problem Suite*), reprezentuje problem grupowania danych o zróżnicowanej gęstości. Składa się z 400 próbek danych tworzących trzy, dobrze odseparowane od siebie skupiska, które różnią się między sobą gęstością danych (zróżnicowanie gęstości danych występuje również w obrębie każdego z nich z osobna).

Podobnie jak w poprzednich eksperymentach, w celu ilustracji przebiegu eksperymentu, na rys. 4.9 przedstawiono wybrane epoki procesu uczenia sieci neuronowej. Zmiany liczby neuronów sieci oraz liczby jej podstruktur można zaobserwować na rys. 4.10. Po zakończeniu uczenia, sieć neuronowa podzieliła się na trzy podstruktury o łącznej liczbie 136 neuronów. Każda z podstruktur jest wielopunktowym prototypem danych, prawidłowo reprezentującym odpowiednie skupiska danych występujące w tym zbiorze. Liczba punktów prototypów danych została automatycznie dostosowana do gęstości danych w lokalnych obszarach skupisk danych (więcej punktów w obszarach o większej gęstości i odwrotnie). Test pokazuje, że proponowana sieć o drzewopodobnej strukturze jest niewrażliwa na zróżnicowanie gęstości danych w skupiskach danych. Dzięki mechanizmom dodawania i usuwania neuronów sieć ta jest w stanie skutecznie dostosować rozmiar prototypu do reprezentowanego przez niego skupiska danych.




Rys. 4.9. Syntetyczny zbiór danych nr 5 – *Lsun* [115] (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

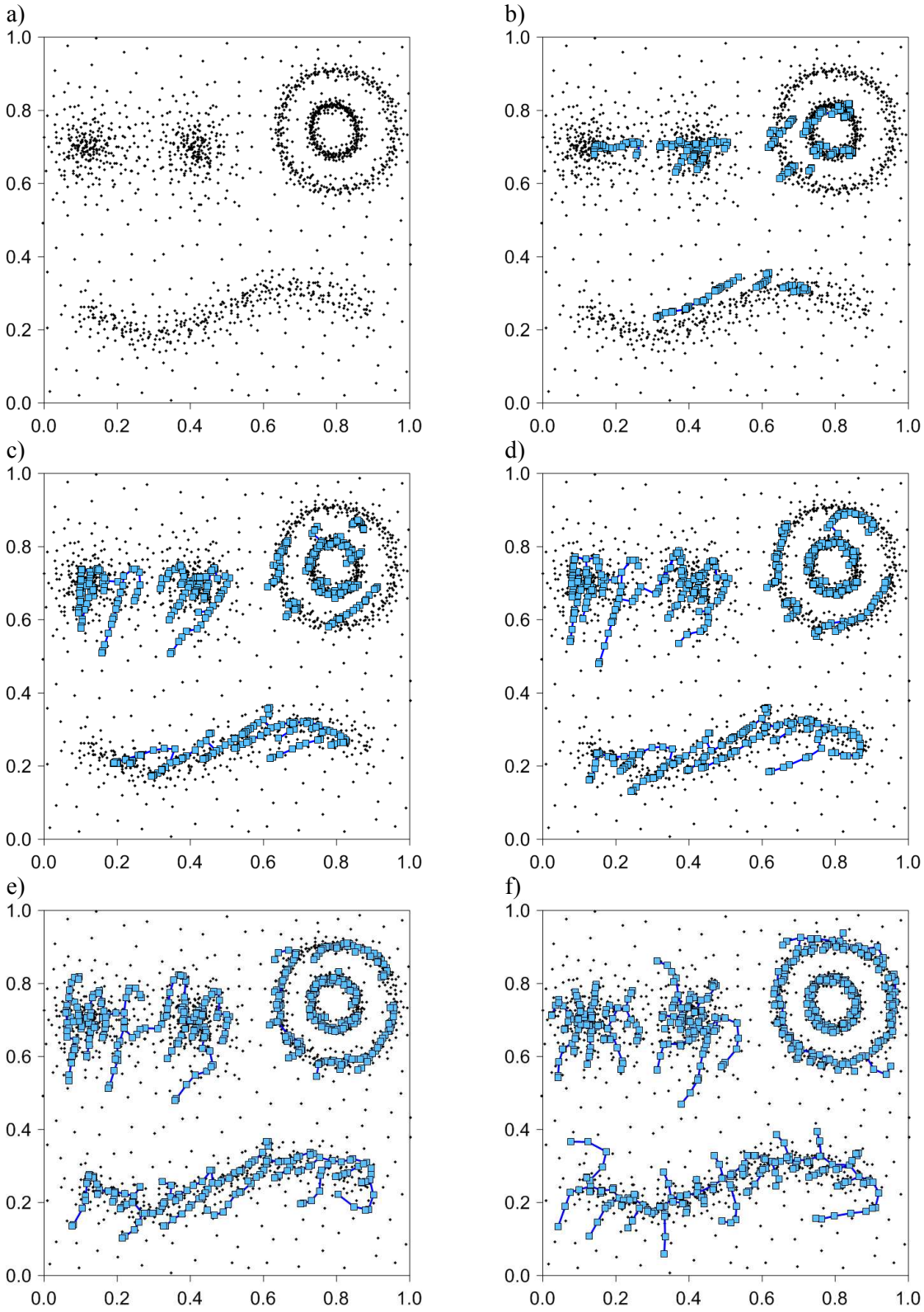



Rys. 4.10. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.9

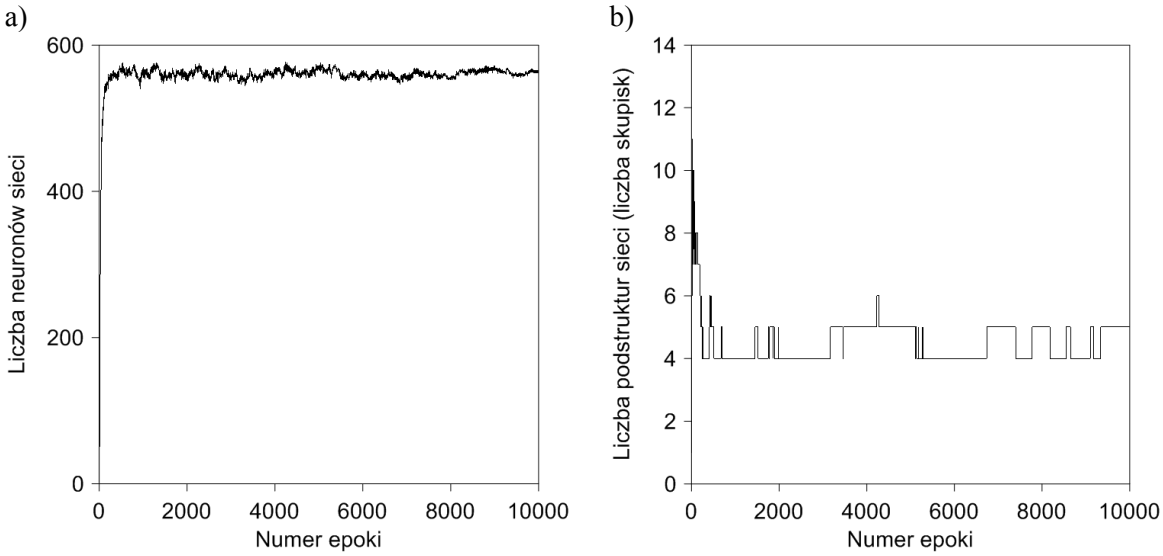
### 4.1.6. Zbiór danych z pracy [104] ze skupiskami o różnorodnych rozmiarach i kształtach

Zbiór danych nr 6 (patrz tabela 4.1), pochodzący z pracy [104], jest jednym z dwóch rozważanych w tym rozdziale syntetycznych zbiorów danych, które zawierają stosunkowo dużą liczbę skupisk (w tym przypadku pięć) o różnych kształtach i rozmiarach. Drugi ze zbiorów będzie rozważany w kolejnym podrozdziale. Próbki danych zbioru (1690 próbek) w zdecydowanej większości są rozmieszczone w pięciu skupiskach (dwa o kształcie pierścieni, jedno w kształcie krzywej sinusoidalnej oraz dwa o kształtach objętościowych, częściowo zachodzące na siebie). Dane zawierają również szum (rys. 4.11a).

Poniżej przedstawiono rozłożenie struktury sieci w obszarze danych w wybranych epokach uczenia (na rys. 4.11) oraz zmiany liczby neuronów i podstruktur sieci neuronowej (na rys. 4.12) w trakcie uczenia. W rezultacie, sieć neuronowa prawidłowo wygenerowała pięć wielopunktowych prototypów o łącznej liczbie 562 neuronów, odpowiednio po jednym dla każdego skupiska danych. Warto zwrócić szczególną uwagę na skupiska w kształcie pierścieni i ich wielopunktowe prototypy. Wykrycie takich skupisk danych metodami konwencjonalnymi, podobnie jak to miało miejsce w przypadku zbioru ze skupiskami w kształcie spiral (patrz podrozdział 4.1.3) jest bardzo trudne lub wręcz niemożliwe (np. w przypadku bardzo popularnego algorytmu *k-means*).




Rys. 4.11. Syntetyczny zbiór danych nr 6 z pracy [104] (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

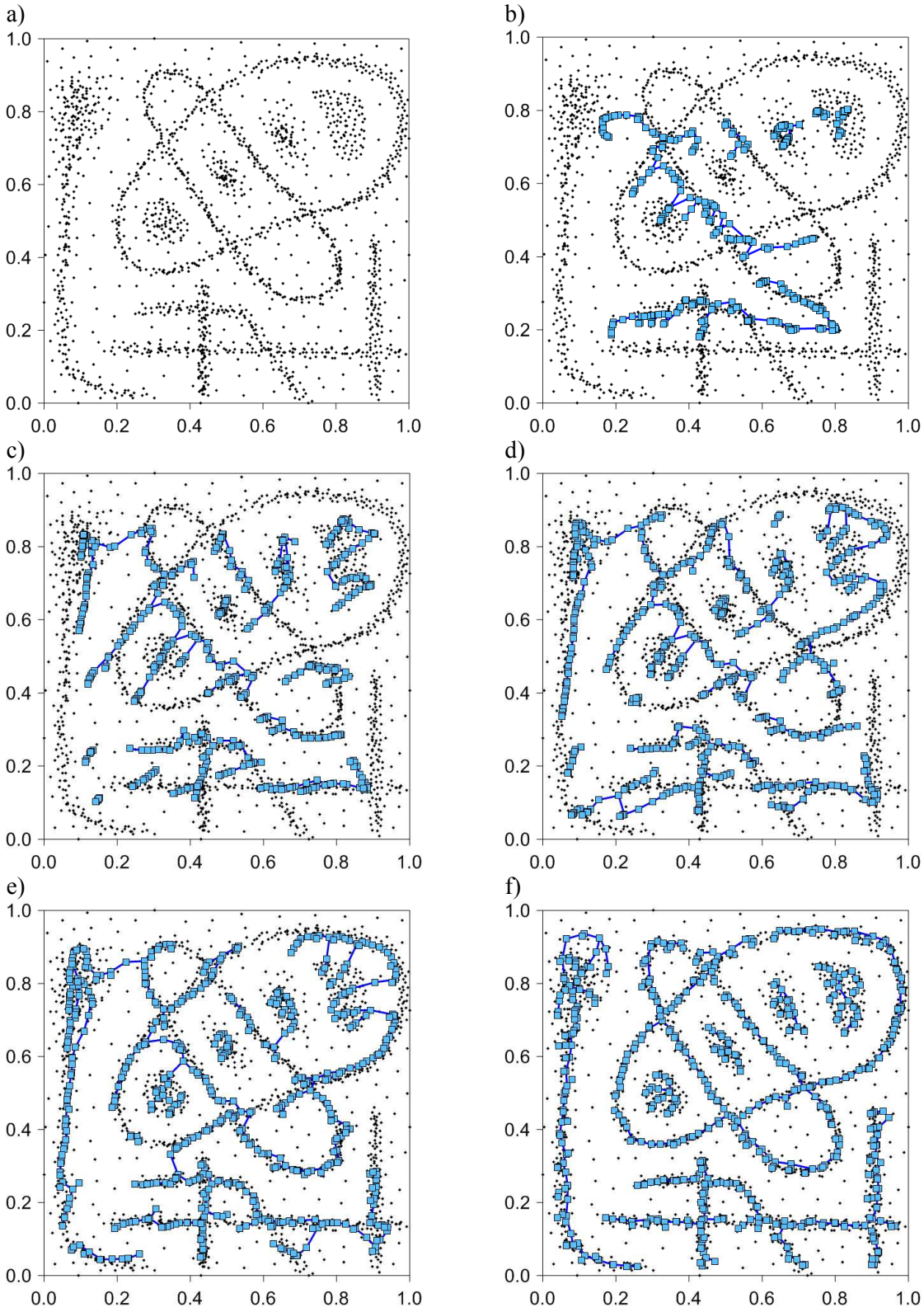



Rys. 4.12. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.11

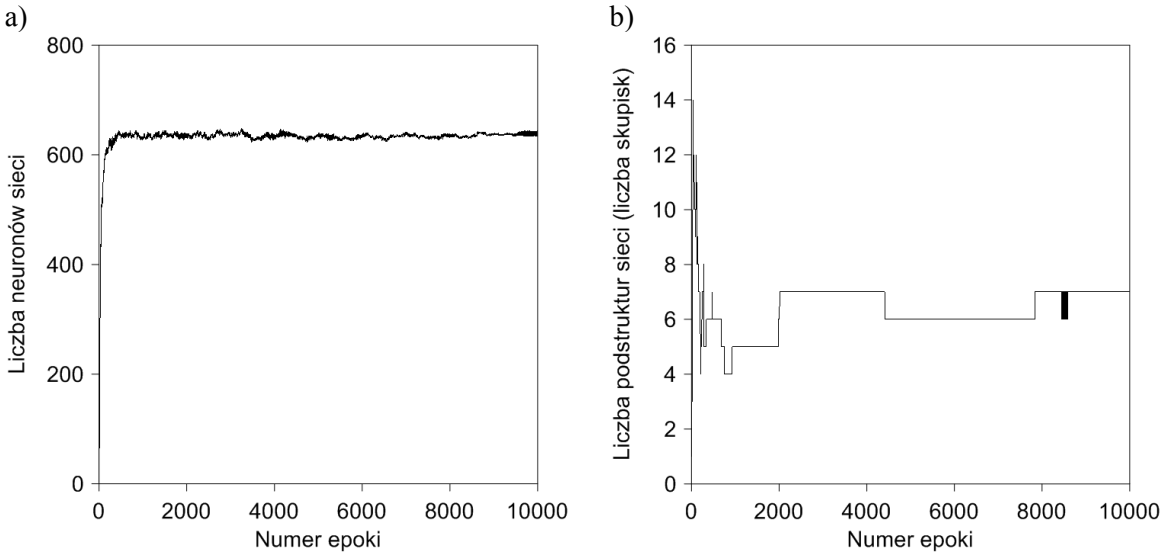
### 4.1.7. Zbiór danych ze skupiskami o złożonych kształtach szkieletowych i objętościowych

Ostatni ze zbiorów danych dwuwymiarowych (nr 7, patrz tabela 4.1) zawiera siedem skupisk danych o zdecydowanie najbardziej złożonych kształtach i rozmiarach, spośród wszystkich wyżej rozważanych zbiorów danych (patrz rys. 4.13a). Zbiór ten zawiera m.in. dwa skupiska danych o kształtach szkieletowych: jedno w postaci dwóch pętli krzyżujących się nawzajem oraz drugie w postaci czterech krzyżujących się obszarów o kształcie liniowym. Są one wyjątkowo trudne do odwzorowania nawet z wykorzystaniem wielopunktowych prototypów. Dla przykładu, wielopunktowe prototypy generowane przez sieć DSOM w postaci łańcuchów neuronów z oczywistych względów nie nadają się do odwzorowania tego typu skupisk danych.

Po zakończeniu procesu uczenia (którego przebieg przedstawiono, analogicznie jak wyżej, na rys. 4.13 i rys. 4.14), sieć neuronowa prawidłowo wygenerowała siedem wielopunktowych prototypów skupisk danych (o łącznej liczbie 636 neuronów), po jednym prototypie dla każdego skupiska danych.




Rys. 4.13. Syntetyczny zbiór danych nr 7 (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

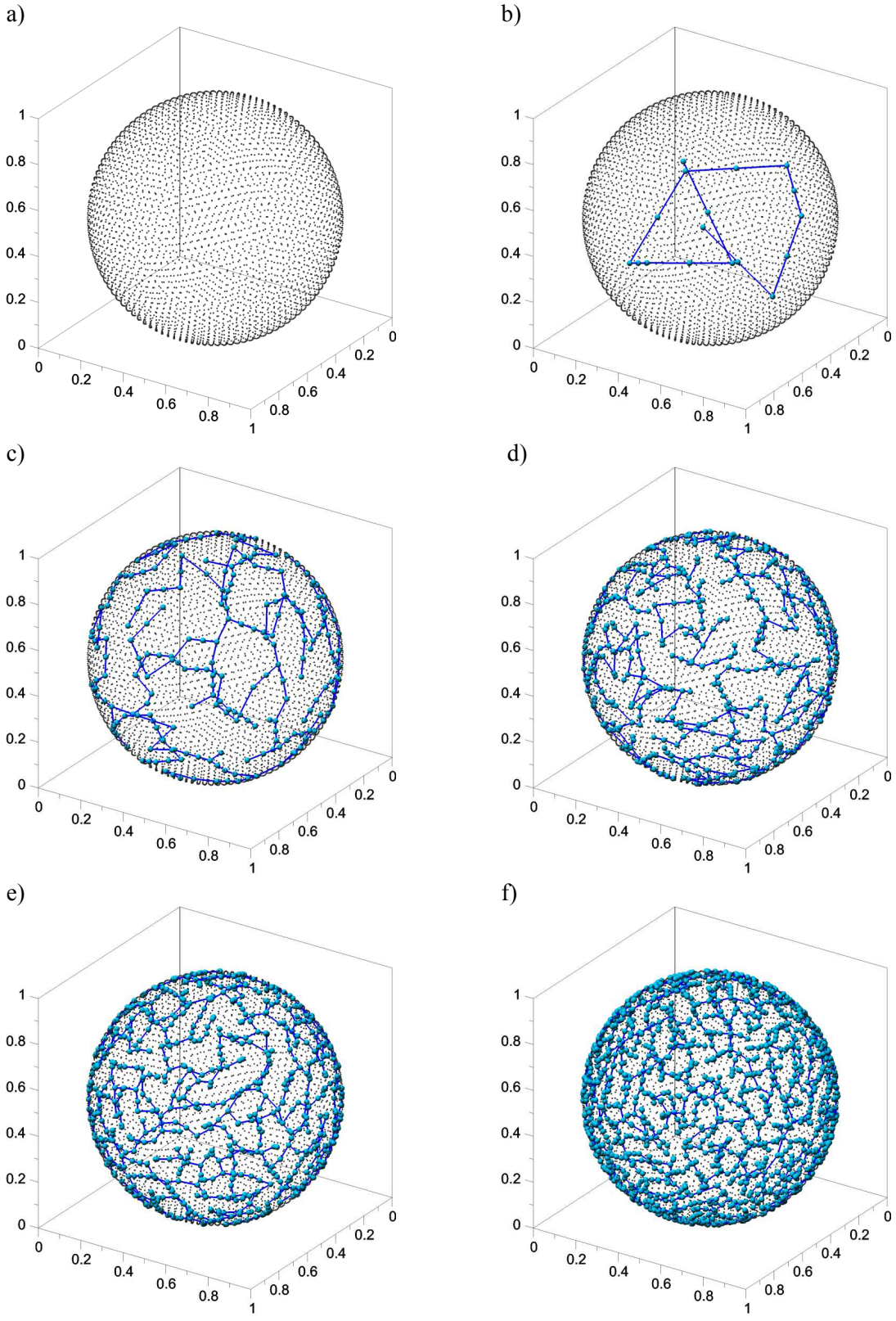



Rys. 4.14. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.13

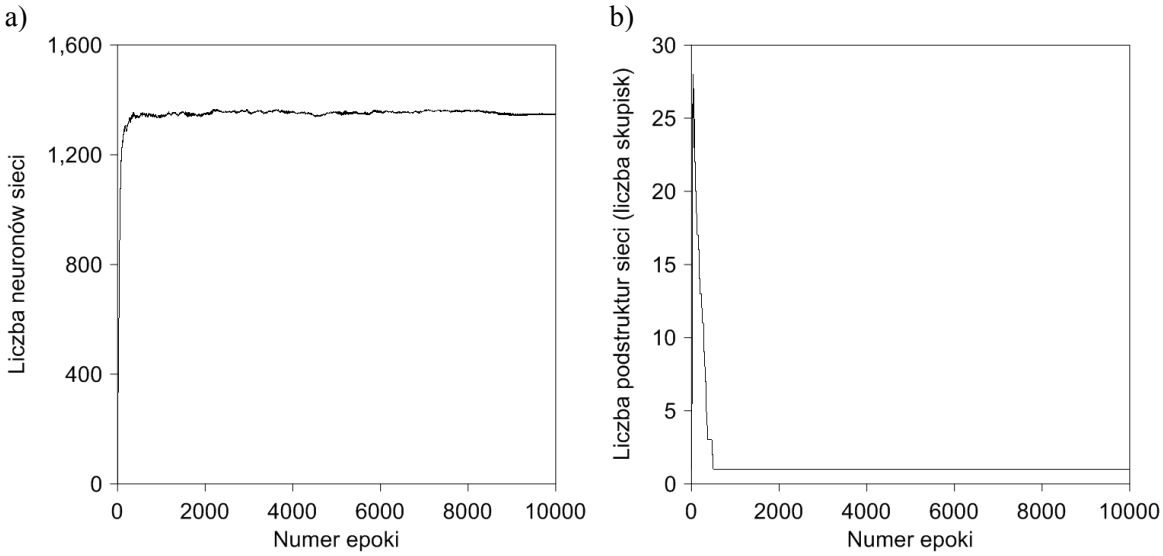
### 4.1.8. Zbiór danych *GolfBall* [115] zawierający dane równomiernie rozmieszczone w obszarze o kształcie sfery

Zbiór danych *GolfBall* jest trzecim z rozważanych zbiorów danych, należących do pakietu podstawowych problemów grupowania danych FCPS [115] i zarazem pierwszym z trzech zbiorów danych trójwymiarowych. Analogicznie jak w przypadku zbioru danych nr 1 (patrz podrozdział 4.1.1), zbiór *GolfBall* reprezentuje kategorię problemów grupowania danych równomiernie rozmieszczonych w obszarze, tym razem, o kształcie sfery. Test jest bardziej zaawansowaną próbą odporności metod grupowania danych na błędne wykrywanie wielu skupisk danych w zbiorze, w którym istnieje tylko jedno duże skupisko danych (analogiczne testy zostały przeprowadzone w podrozdziałach 4.1.1 i 4.1.2 z wykorzystaniem danych dwuwymiarowych).

Przebieg procesu uczenia sieci przedstawiono na rys. 4.15 oraz rys. 4.16. Po zainicjowaniu procesu uczenia, w ciągu zaledwie kilkudziesięciu epok uczenia, struktura sieci została podzielona na 28 podstruktur (patrz rys. 4.16b). Po upływie 497 epoki uczenia (czyli ok. 5% całkowitego okresu uczenia) podstruktury sieci ponownie połączyły się w jedną strukturę, prawidłowo odwzorowującą istniejące skupisko danych. Liczba neuronów sieci została ustabilizowana na poziomie 1349 neuronów (patrz rys. 4.16a). Test ponownie wykazał zdolność sieci do wykrywania również braku lokalnych skupisk danych w rozważanym zbiorze, co sieć sygnalizuje wykryciem jednego dużego skupiska obejmującego cały zbiór danych.




Rys. 4.15. Syntetyczny zbiór danych nr 8 – *GolfBall* [115] (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

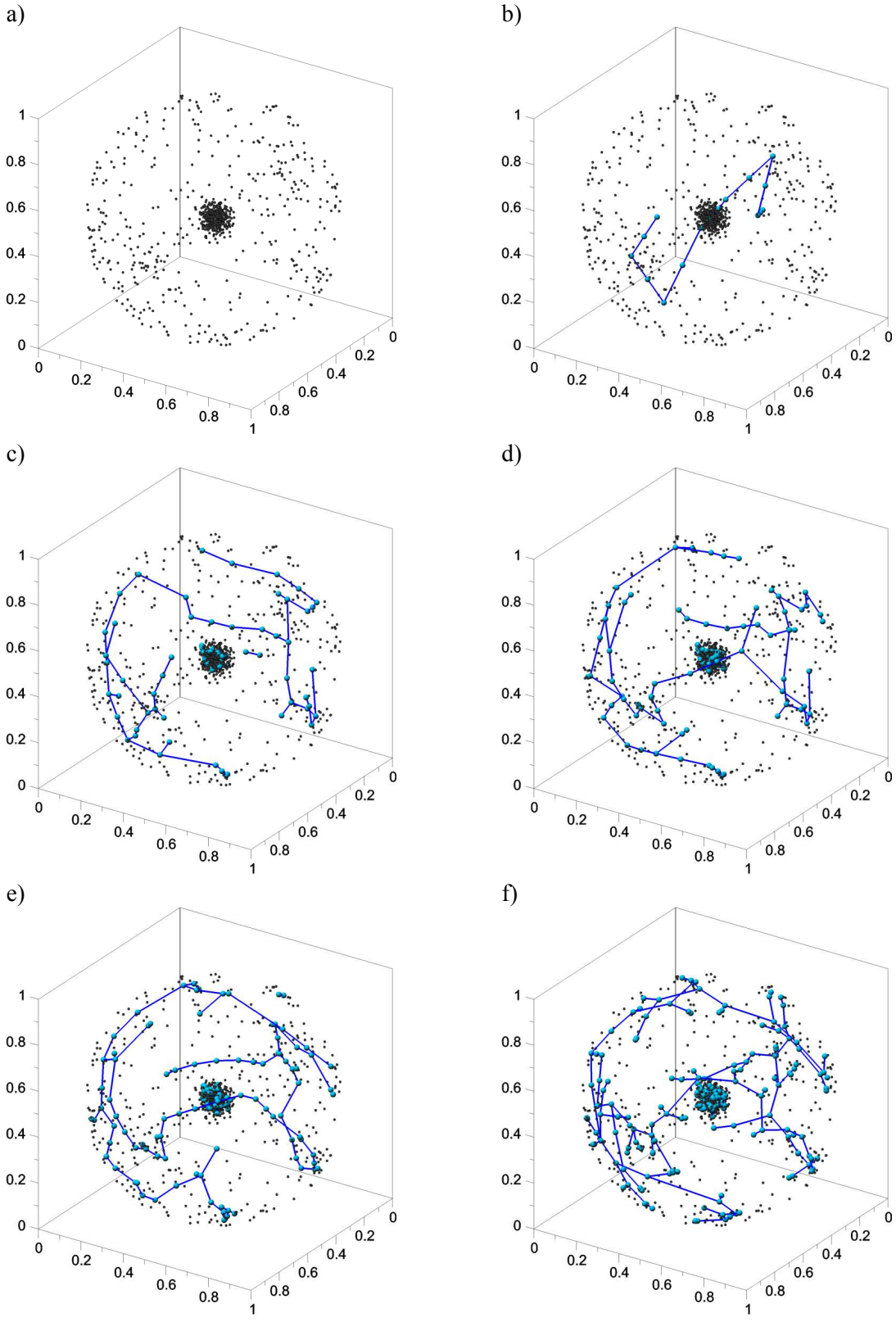



Rys. 4.16. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.15

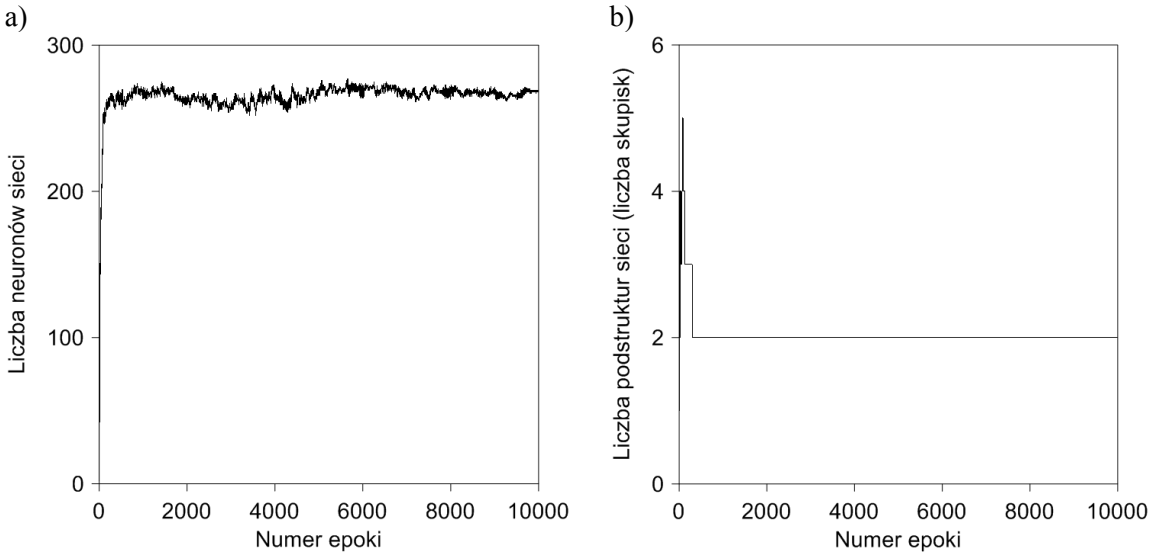
### 4.1.9. Zbiór danych *Atom* [115] z dwoma sferycznymi skupiskami danych

Zbiór danych *Atom* jest kolejnym zbiorem danych trójwymiarowych, pochodzącym z pakietu podstawowych problemów grupowania danych FCPS [115]. Zawiera dwa skupiska danych o sferycznych kształtach: jedno mniejsze zlokalizowane jest wewnątrz drugiego, większego (patrz rys. 4.17a). Zbiór danych *Atom* łączy w sobie dwie kategorie problemów grupowania danych: problem wykrywania liniowo nieseparowalnych skupisk danych (problem był rozważany w ramach testu z wykorzystaniem zbioru danych nr 3 – patrz tabela 4.1 i podrozdział 4.1.3) oraz problem wykrywania skupisk o różnych gęstościach danych (problem był rozważany w ramach testu z wykorzystaniem zbioru danych nr 5 *Lsun* [115] – patrz tabela 4.1 i podrozdział 4.1.5).

I tym razem, proponowana sieć neuronowa prawidłowo wykryła dwa skupiska w zbiorze danych, co można zaobserwować na rys. 4.17, przedstawiającym wybrane etapy procesu jej uczenia. Po bardzo niewielkiej liczbie epok, sieć neuronowa podzieliła się na dwie podstruktury (patrz rys. 4.18b), natomiast liczba neuronów niemal przez cały okres uczenia sieci oscylowała wokół liczby końcowej, równiej 268 (patrz rys. 4.18a).




Rys. 4.17. Syntetyczny zbiór danych nr 9 – *Atom* [115] (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)

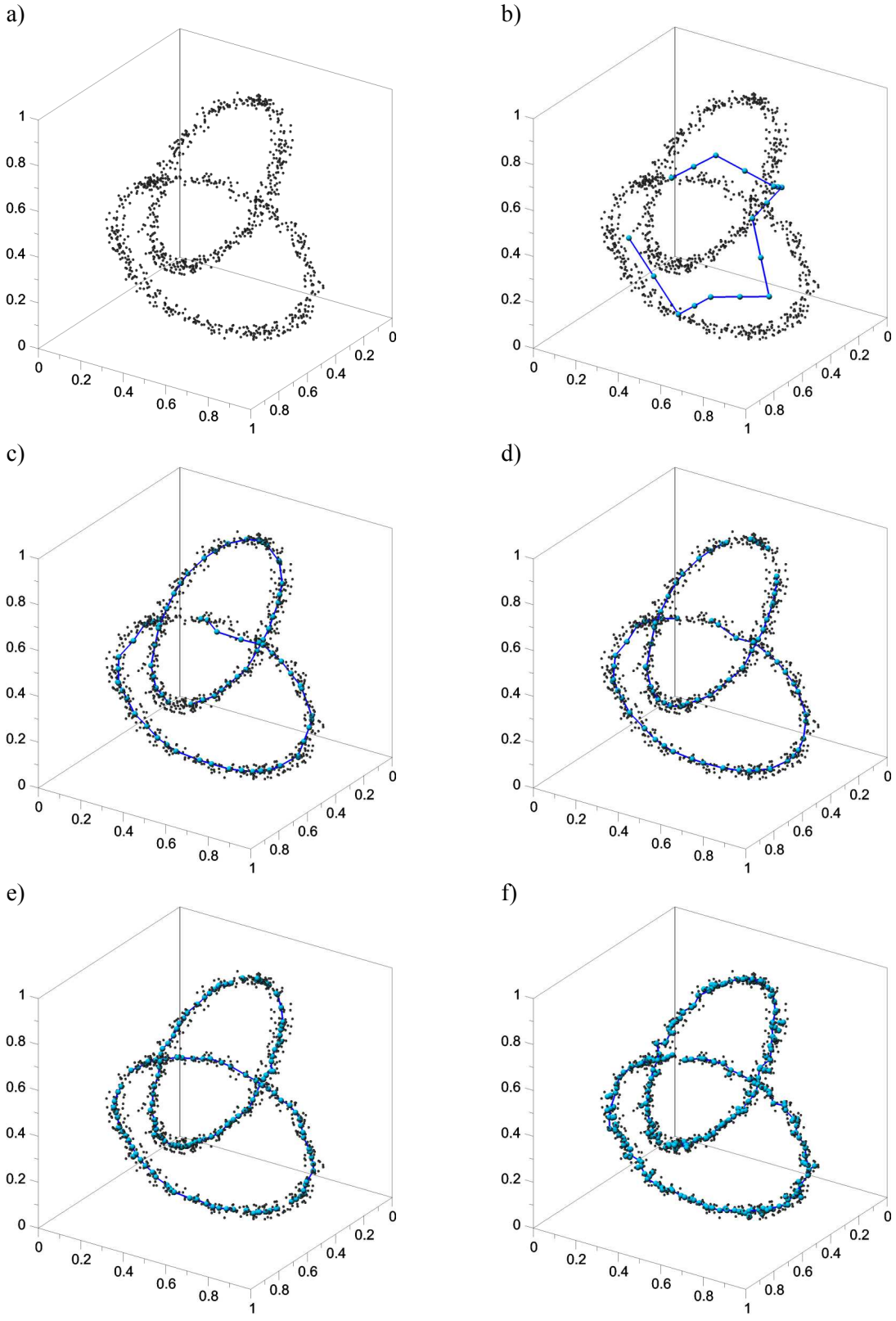



Rys. 4.18. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.17

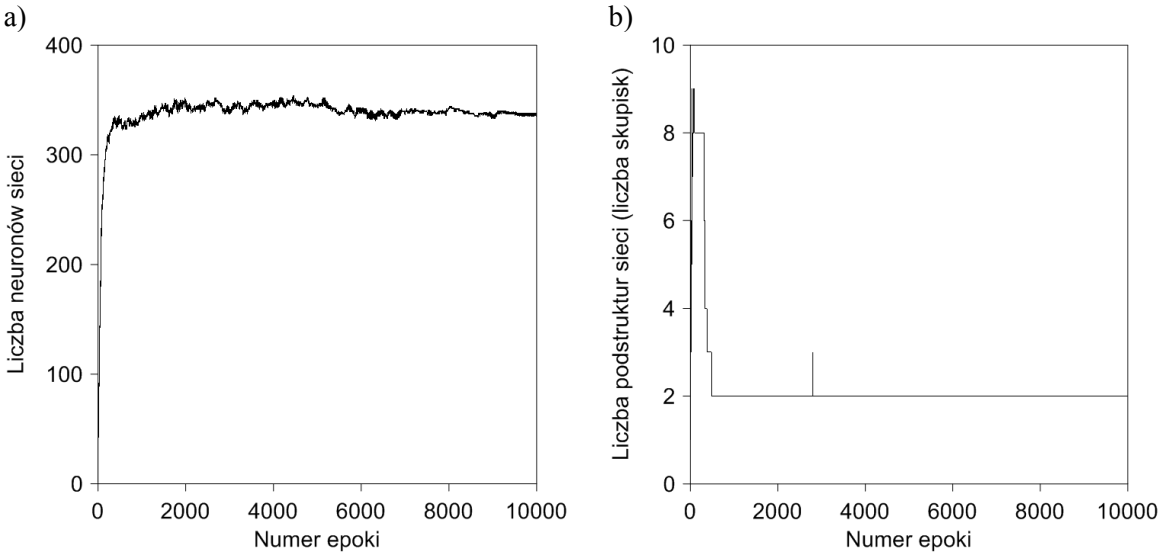
### 4.1.10. Zbiór danych *Chainlink* [115] ze skupiskami w postaci połączonych pierścieni

Ostatni z rozważanych w tym rozdziale syntetycznych zbiorów danych – zbiór danych *Chainlink* – również pochodzi z pakietu podstawowych problemów grupowania danych FCPS [115]. Podobnie jak w przypadku zbioru danych nr 3 (patrz tabela 4.1), zawierającego dwa skupiska w kształcie spiral oraz zbioru danych nr 9 *Atom,* zawierającego dwa skupiska o kształtach sferycznych, rozważany obecnie zbiór *Chainlink* reprezentuje kategorię problemów wykrywania liniowo nieseparowalnych skupisk danych. Zawiera dwa skupiska danych w kształcie wzajemnie połączonych pierścieni (dwa „oczka" łańcucha).

Analogicznie jak we wszystkich poprzednich eksperymentach, na rys. 4.19 zilustrowano wybrane epoki procesu uczenia sieci neuronowej, natomiast na rys. 4.20 pokazano przebiegi zmian liczby neuronów sieci oraz liczby podstruktur sieci podczas uczenia. Po zakończeniu uczenia, dwie podstruktury sieci neuronowej (o łącznej liczbie 338 neuronów) bardzo dobrze odwzorowują pierścieniowe kształty obu istniejących w zbiorze skupisk danych. Tak samo jak w przypadku testów z wykorzystaniem wspomnianych wyżej zbiorów danych nr 3 (dwa skupiska w kształcie spiral) i nr 9 (*Atom*), również i tym razem rozważany test potwierdził wysoką skuteczność proponowanej sieci neuronowej w wykrywaniu liniowo nieseparowalnych skupisk danych.




Rys. 4.19. Syntetyczny zbiór danych nr 10 – *Chainlink* [115] (patrz tabela 4.1) (a) oraz rozmieszczenie struktury proponowanej sieci neuronowej w obszarze danych w wybranych epokach uczenia: $e = 10$ (b), $e = 20$ (c), $e = 100$ (d), $e = 200$ (e) i $e = 10000$ – koniec uczenia (f)




Rys. 4.20. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie jej uczenia dla problemu z rys. 4.19

## 4.2. Grupowanie danych w rzeczywistych, wielowymiarowych zbiorach danych

W celu potwierdzenia wysokiej efektywności proponowanej, uogólnionej sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej w grupowaniu danych, zostaną teraz przedstawione trzy przykładowe eksperymenty z wykorzystaniem powszechnie znanych, rzeczywistych i wielowymiarowych zbiorów danych, pochodzących z serwera Uniwersytetu Kalifornijskiego w Irvine (http://archive.ics.uci.edu/ml). Tabela 4.3 zawiera zestawienie informacji o tych zbiorach danych (liczbę próbek, atrybutów oraz klas); szczegółowe informacje o rozważanych zbiorach zawiera dodatek A niniejszej pracy.

Tabela 4.3. Zestawienie informacji o rozważanych rzeczywistych zbiorach danych

| Lp. | Nazwa zbioru danych | Liczba atrybutów | | Liczba próbek danych | Liczba skupisk (klas) danych |
|---|---|---|---|---|---|
| | | numerycznych | symbolicznych | | |
| 1 | *Congressional Voting Records* | 0 | 16 | 435 | 2 |
| 2 | *Breast Cancer Wisconsin (Diagnostic)* | 30 | 0 | 569 | 2 |
| 3 | *Wine* | 13 | 0 | 178 | 3 |

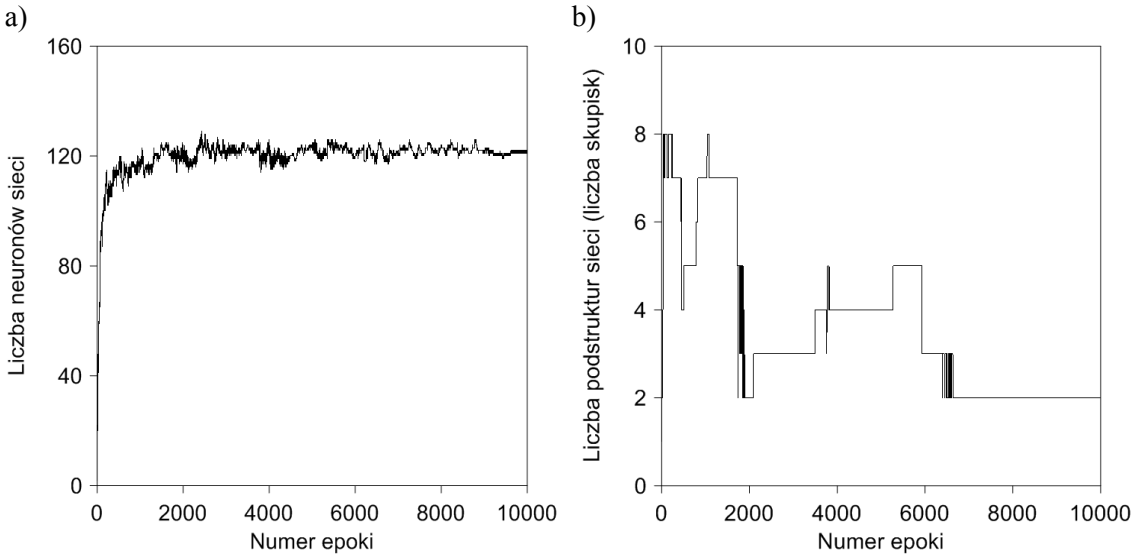
Należy podkreślić, że zbiory danych zawierają informacje o przynależności poszczególnych próbek danych do odpowiednich klas (grup), jednakże nie jest ona w żaden sposób wykorzystywana podczas procesu uczenia sieci neuronowej (odbywa się on w pełni nienadzorowanym trybie). W wyniku szeregu eksperymentów numerycznych stwierdzono, że w przypadku danych o znacznie wyższych, niż 2 lub 3, wymiarowościach, nieznaczne zmniejszenie parametru $\alpha_{rozł}$ sterującego rozłączaniem struktury sieci poprawia wyniki grupowania. Stąd, w niniejszym podrozdziale przyjęto $\alpha_{rozł} = 3$ (wartości pozostałych parametrów, zawarte w tabeli 4.2, pozostają bez zmian).

W związku z tym, że wymiarowość rozważanych zbiorów danych jest większa od 3, nie jest możliwa bezpośrednia, graficzna ilustracja rozmieszczenia próbek danych oraz struktur sieci neuronowych w przestrzeni danych, jak to miało miejsce w przypadku poprzednich eksperymentów z wykorzystaniem danych syntetycznych. Przedstawiono zatem jedynie przebiegi zmian liczby neuronów oraz liczby podstruktur sieci neuronowych w trakcie trwania procesów uczenia. W każdym z eksperymentów, po zakończeniu uczenia dokonano tzw. etykietowania otrzymanych podstruktur sieci, czyli nadano im nazwy klas odpowiednio do reprezentowanych przez te podstruktury skupisk (klas) danych. Operacja ta pozwoliła na analizę dokładności grupowania danych, poprzez wyznaczenie liczby poprawnych decyzji i liczby błędnych decyzji sieci neuronowych, które po zakończeniu uczenia funkcjonowały w trybie książki kodowej (patrz podrozdział 2.1), jako klasyfikatory danych (odpowiedzią klasyfikatora na zadany wektor danych wejściowych $\boldsymbol{x}$ jest etykieta klasy nadana tej podstrukturze sieci neuronowej, która zawiera neuron o numerze $j_{\boldsymbol{x}}$ określony formułą (2.3)). Wyniki analizy zostały przedstawione w tabelach w postaci tzw. tablic pomyłek (ang. *confusion matrices*).

### 4.2.1. Zbiór danych *Congressional Voting Records*

Zbiór danych *Congressional Voting Records* zawiera wyniki głosowania członków Izby Reprezentantów Kongresu USA, należących do Partii Konserwatywnej lub Partii Demokratycznej, w 16 rozpatrywanych kwestiach. Przebieg zmian liczby neuronów oraz liczby podstruktur sieci w trakcie uczenia przedstawia rys. 4.21. Po zakończeniu uczenia, topologiczna struktura sieci została podzielona na dwie podstruktury (o sumarycznej liczbie 122 neuronów), z których każda reprezentuje jedno skupisko danych. Tabela 4.4 zawiera wyniki analizy odpowiedzi sieci neuronowej – funkcjonującej jako klasyfikator danych – dla wszystkich próbek danych rozważanego zbioru. Liczba po-

prawnych decyzji klasyfikatora wynosi 94,71%, co należy uznać za bardzo dobry wynik (biorąc pod uwagę, że został on uzyskany w drodze uczenia z danych w pełni nienadzorowanym trybie).



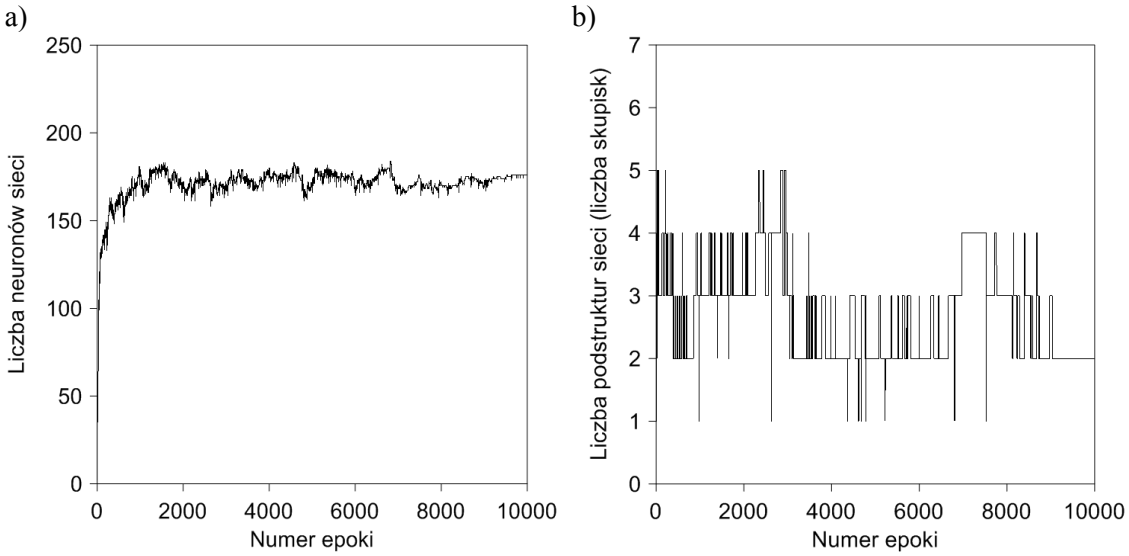

Rys. 4.21. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych w zbiorze *Congressional Voting Records*)

Tabela 4.4. Analiza wyników grupowania danych w zbiorze *Congressional Voting Records* z wykorzystaniem proponowanej sieci neuronowej

| Etykieta klasy | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie: | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|
| | | *republican* | *democrat* | | | |
| *republican* | 168 | 158 | 10 | 158 | 10 | 94.05% |
| *democrat* | 267 | 13 | 254 | 254 | 13 | 95.13% |
| **RAZEM** | **435** | **171** | **264** | **412** | **23** | **94.71%** |

### 4.2.2. Zbiór danych *Breast Cancer Wisconsin (Diagnostic)*

Zbiór danych *Breast Cancer Wisconsin (Diagnostic)* zawiera wyniki badań histopatologicznych biopsji cienkoigłowej, wykonywanych przy diagnostyce raka piersi. Próbki danych charakteryzują wygląd pobranych do badań komórek, zdiagnozowanych do jednego z dwóch przypadków: zmiana złośliwa (nowotworowa) lub zmiana łagodna (patrz dodatek A). Na rys. 4.22 przedstawiono przebiegi zmian liczby neuronów i liczby podstruktur sieci neuronowej w trakcie uczenia.




Rys. 4.22. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych w zbiorze *Breast Cancer Wisconsin (Diagnostic)*)

Po zakończeniu uczenia, liczba neuronów sieci ustabilizowała się na poziomie 176 neuronów, natomiast liczba podstruktur sieci odpowiada liczbie klas zbioru danych. Tabela 4.5 zawiera rezultaty analizy odpowiedzi sieci neuronowej dla wszystkich próbek danych rozważanego zbioru. Liczba poprawnych decyzji sieci neuronowej, funkcjonującej jako klasyfikator danych, wynosi 90.51%. Podobnie jak w przypadku eksperymentu z rozdziału 4.2.1 – i z tych samych względów – wynik ten należy uznać za bardzo dobry.

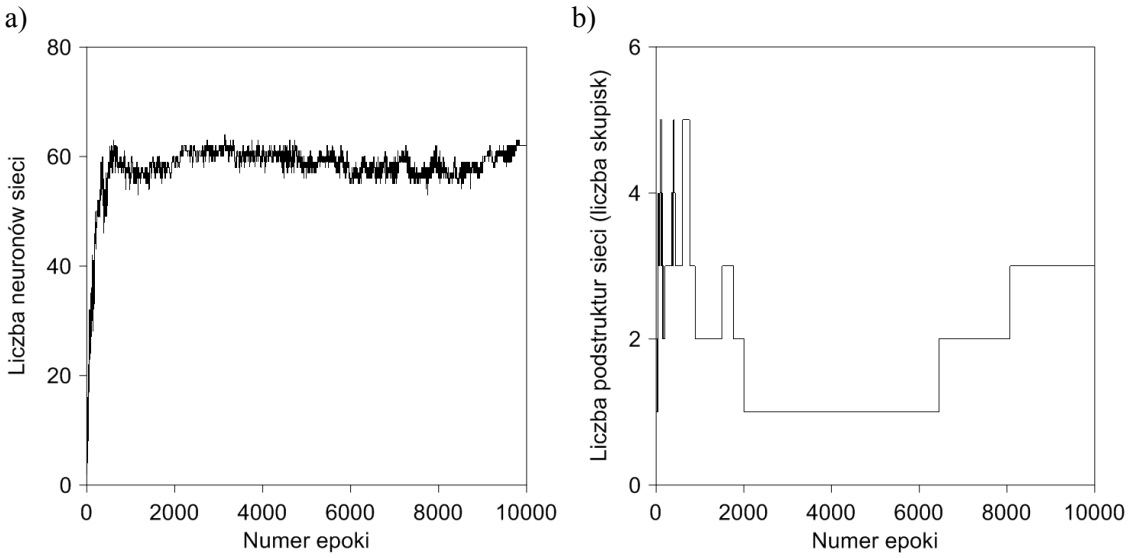
Tabela 4.5. Analiza wyników grupowania danych w zbiorze *Breast Cancer Wisconsin (Diagnostic)* z wykorzystaniem proponowanej sieci neuronowej

| Etykieta klasy | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie: | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|
| | | *malignant* | *benign* | | | |
| *malignant* | 212 | 166 | 46 | 166 | 46 | 78.30% |
| *benign* | 357 | 8 | 349 | 349 | 8 | 97.76% |
| **RAZEM** | **569** | **174** | **395** | **515** | **54** | **90.51%** |

### 4.2.3. Zbiór danych *Wine*

Ostatni z rozważanych rzeczywistych zbiorów danych, tzn. zbiór o nazwie *Wine*, zawiera wyniki analizy chemicznej win z trzech różnych odmian, uprawianych w tym

samym regionie Włoch. Rys. 4.23 przedstawia zmiany liczby neuronów oraz liczby podstruktur sieci podczas jej uczenia.




Rys. 4.23. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych w zbiorze *Wine*)

Po zakończeniu uczenia, utworzone trzy podstruktury sieci (o łącznej liczbie 62 neuronów) reprezentują, odpowiednio, trzy kategorie win. Tabela 4.6 zawiera rezultaty analizy odpowiedzi sieci neuronowej dla próbek danych ze zbioru *Wine*. Eksperyment ponownie potwierdza wysoką skuteczność (na poziomie 94.38% poprawnych decyzji) proponowanej sieci neuronowej w wykrywaniu skupisk w rzeczywistych, wielowymiarowych zbiorach danych, w przypadku pracy w pełni nienadzorowanym trybie.

Tabela 4.6. Analiza wyników grupowania danych w zbiorze *Wine* z wykorzystaniem proponowanej sieci neuronowej

| Etykieta klasy | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie: | | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | | | |
| 1 | 59 | 57 | 2 | 0 | 57 | 2 | 96.61% |
| 2 | 71 | 4 | 65 | 2 | 65 | 6 | 91.55% |
| 3 | 48 | 0 | 2 | 46 | 46 | 2 | 95.83% |
| **RAZEM** | **178** | **61** | **69** | **48** | **168** | **10** | **94.38%** |

## 4.3. Podsumowanie

W niniejszym rozdziale przedstawiono praktyczne testy funkcjonowania proponowanej w ramach niniejszej pracy, uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej. Do testów wykorzystano 10 syntetycznych dwu i trójwymiarowych zbiorów danych oraz 3 rzeczywiste, wielowymiarowe zbiory danych, z których każdy reprezentuje odrębną kategorię problemów grupowania danych. We wszystkich eksperymentach, sieć neuronowa prawidłowo wykryła liczbę skupisk danych istniejących w rozważanych zbiorach. Ponadto, sieć ta wygenerowała dla nich wielopunktowe prototypy o kształtach i wielkości (liczbie punktów – neuronów) odpowiednich do kształtów i rozmiarów tych skupisk. Lokalizacja prototypów w przestrzeni danych została przeprowadzona w sposób automatyczny i adekwatny do lokalizacji, charakteru i gęstości danych w poszczególnych skupiskach.

W następnych rozdziałach zostaną przeprowadzone kolejne, bardziej rozbudowane testy praktycznej użyteczności proponowanej sieci neuronowej w rzeczywistych problemach grupowania danych, z wykorzystaniem zbiorów opisujących ekspresję genów.

# 5. Zastosowanie proponowanych sieci neuronowych do grupowania danych opisujących ekspresję genów w przypadku nowotworowych chorób układu krwionośnego

Niniejszy rozdział jest pierwszym z trzech rozdziałów, które przedstawiają zastosowania proponowanych sieci neuronowych w problemach grupowania danych w rzeczywistych, złożonych i wielowymiarowych zbiorach danych opisujących ekspresję genów. W ramach tych eksperymentów zostanie wykazane, że proponowana sieć neuronowa jest efektywnym narzędziem do grupowania tego typu danych, w tym zarówno do grupowania bazującego na próbkach, jak i bazującego na genach (oba podejścia zostały przedstawione w podrozdziale 1.2).

W pierwszej kolejności zostanie rozważony problem grupowania danych w dwóch zbiorach, pochodzących z badań patomorfologicznych chorób nowotworowych układu krwionośnego (badań nad klasyfikacją różnych typów białaczek). W pierwszej części rozdziału (podrozdział 5.1), syntetycznie przedstawiono problem oraz postawione w nim zadania. Następnie (w podrozdziale 5.2), zaprezentowano procesy uczenia sieci neuronowych (procesy grupowania danych) oraz omówiono – w podrozdziale 5.3 – otrzymane rezultaty. Na zakończenie (w podrozdziale 5.4), porównano je z wynikami uzyskanymi przy wykorzystaniu innych, do pewnego stopnia alternatywnych metod.

## 5.1. Przedstawienie problemu

Rozważane zbiory danych *Leukemia* i *Mixed Lineage Leukemia* (ten ostatni nazywany będzie dalej zbiorem *MLL*) zawierają poziomy aktywności genów (poziomy ekspresji genów), zaobserwowane w trakcie przebiegu procesów transkodowania informacji genetycznej DNA na produkty funkcyjne (RNA lub białka) z wykorzystaniem mikromacierzy DNA (syntetyczna charakterystyka takiego procesu została opisana w podrozdziale 1.2). Zbiory te są dostępne na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/datasets.html).

Zbiór danych *Leukemia* zawiera poziomy aktywności 3571 genów, wyodrębnionych z próbek materiału genetycznego komórek nowotworowych pobranych od 72 pacjentów. Każda próbka jest przyporządkowana do jednej z dwóch klas, odpowiadają-

cych dwóm typom choroby zdiagnozowanej u pacjenta: ostrej białaczki szpikowej AML (ang. *Acute Myelogenous Leukemia*) oraz ostrej białaczki limfoblastycznej ALL (ang. *Acute Lymphoblastic Leukemia*) [58] (odpowiednio, 25 i 47 próbek klas AML i ALL). Z kolei, zbiór danych *MLL* zawiera poziomy aktywności 2474 genów wyodrębnionych z próbek materiału genetycznego pochodzących również od 72 pacjentów. Tym razem jednak, każda próbka jest przyporządkowana do jednej z trzech klas, odpowiadających trzem typom białaczek: ALL i AML (jak powyżej) oraz białaczki typu MLL (ang. *Mixed Lineage Leukemia*), którą uznaje się za szczególny przypadek białaczek ALL lub AML [3] (statystycznie 22% białaczek ALL i 5% AML uznaje się za białaczki typu MLL [16]; zbiór zawiera, odpowiednio, 24, 28 i 20 próbek klas ALL, AML i MLL). Należy wspomnieć, że rozważane zbiory danych są wariantami większych zbiorów o tych samych nazwach (zawierających odpowiednio, 7129 genów w zbiorze *Leukemia* oraz 12582 genów w *MLL*; szczegóły można znaleźć w pracy [42]) i zostały udostępnione publicznie po usunięciu kopii genów oraz genów, dla których informacja o poziomie aktywności była zaszumiona (procedura ta została opisana szczegółowo w pracach [23] i [31]).

Dla każdego ze zbiorów danych zostaną przeprowadzone procesy uczenia dwóch proponowanych sieci neuronowych. W przypadku zbioru danych *Leukemia*, zadaniem pierwszej sieci neuronowej jest podział próbek materiału genetycznego, wyodrębnionego z komórek nowotworowych pobranych od pacjentów (dalej nazywanych krótko próbkami danych), na dwie grupy reprezentujące, odpowiednio, białaczki typu ALL i AML. Ponadto, zadaniem sieci jest wyznaczenie dwóch wielopunktowych prototypów tych grup, które mogą być wykorzystane w przyszłości do przyporządkowywania próbek danych nieuczestniczących w procesie uczenia sieci neuronowej do jednej lub drugiej grupy (praca sieci w trybie klasyfikatora). Z kolei, zadaniem drugiej sieci neuronowej jest podział genów na grupy genów o podobnych poziomach ekspresji i wyznaczenie wielopunktowych prototypów tych grup. W przypadku zbioru danych *MLL*, zadania obu sieci neuronowych są analogiczne, przy czym pierwsza sieć ma dokonać podziału próbek danych na trzy grupy reprezentujące, odpowiednio, białaczki typu ALL, AML i MLL. Jak już wspomniano na wstępie tego rozdziału, we wszystkich przypadkach procesów uczenia sieci neuronowych przyjęto założenie, że liczba grup nie jest znana i zostanie wyznaczona w sposób automatyczny podczas uczenia, które – co należy podkreślić – odbywa się w pełni nienadzorowanym trybie. Celem przeprowadzonych ekspe-

rymentów jest zbadanie przydatności proponowanego narzędzia do budowy systemów wspomagania badań patomorfologicznych, a w szczególności badań nad klasyfikacją typów chorób nowotworowych układu krwionośnego.

## 5.2. Procesy uczenia uogólnionych samoorganizujących się sieci neuronowych o drzewopodobnych strukturach

Procesy uczenia proponowanych sieci neuronowych przeprowadzono w analogiczny sposób jak w przypadku eksperymentów rozważanych w rozdziale 4. Dla porządku, tabela 5.1 przedstawia zestawienie parametrów sterujących procesem uczenia rozważanych sieci neuronowych we wszystkich eksperymentach numerycznych prezentowanych w tym rozdziale oraz rozdziałach 6 i 7. Warto zauważyć, że zdecydowana większość tych parametrów jest identyczna jak w tabeli 4.2 dla eksperymentów przedstawionych w rozdz. 4 (uwzględniając minimalną korektę parametru $\alpha_{rozł}$ dla wielowymiarowych danych z rozdz. 4.2). Pośrednio wskazuje to na ich względnie uniwersalny charakter w odniesieniu zarówno do szerokiej klasy danych, jak i szerokiego spektrum problemów grupowania danych. W rozważanych problemach grupowania danych opisujących ekspresję genów, ze względu na bardzo małą liczbę próbek w zbiorach, parametry: $lzw_{\max}$, $\alpha_{łącz}$ i $m_{\min}$ kontrolujące, odpowiednio, liczbę neuronów w sieci, łączenie jej podstruktur oraz usuwanie podstruktur z małą liczbą neuronów, zostały wyjątkowo zredukowane do swych minimalnych wartości, tzn. $lzw_{\min} = lzw_{\max} = 2$, $\alpha_{łącz} = \alpha_{rozł} = 3$, $m_{\min} = 2$.

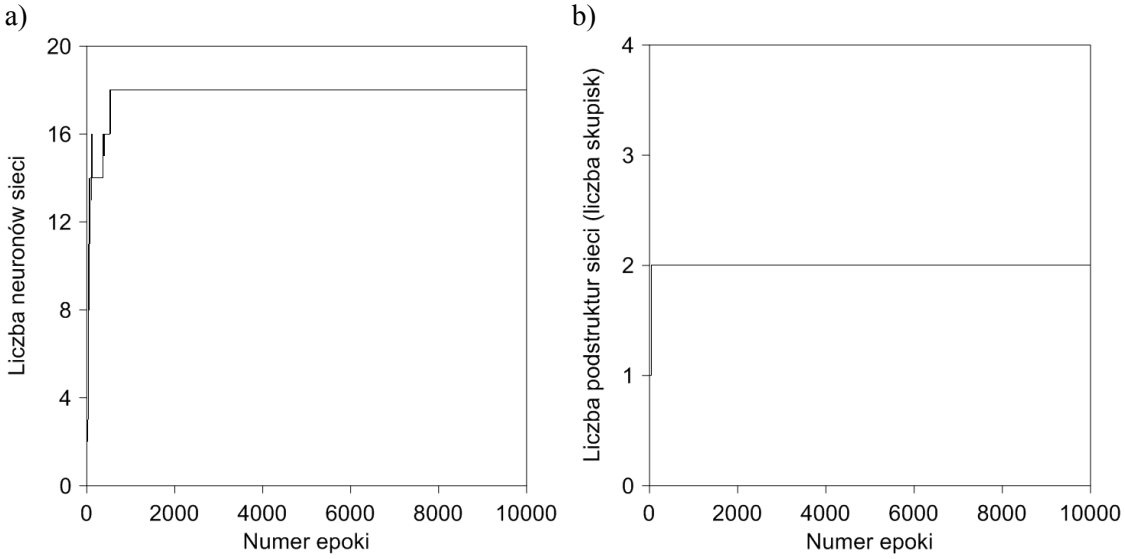
### 5.2.1. Procesy uczenia sieci neuronowych dla zbioru danych *Leukemia*

Przebieg procesu uczenia proponowanej sieci neuronowej, której zadaniem jest przyporządkowanie próbek danych zbioru *Leukemia* do grup odpowiadających dwóm typom białaczki, tzn. ALL i AML (grupowanie bazujące na próbkach) ilustruje rys. 5.1. Na rys. 5.1a widoczne są zmiany liczby neuronów sieci w trakcie uczenia. Już po upływie ok. 1000 epok, liczba neuronów ustabilizowała się na poziomie $m = 18$. Z kolei, na rys. 5.1b można zaobserwować niemal błyskawiczne ustabilizowanie liczby podstruktur sieci neuronowej na poziomie równym liczbie klas w rozważanym zbiorze danych (dwie podstruktury, z których każda odpowiada jednej klasie danych).

Tabela 5.1. Parametry proponowanych sieci neuronowych wykorzystywanych do grupowania danych bazującego na genach oraz bazującego na próbkach w zbiorach danych z rozdziałów 5, 6 i 7

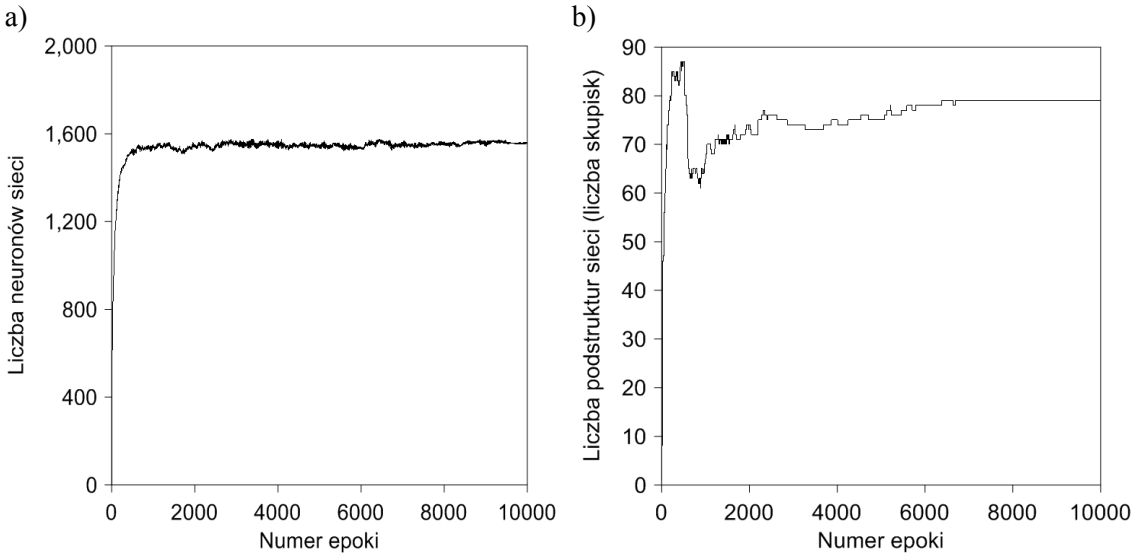
| Nazwa parametru/funkcji | Wartość (rodzaj) parametru |
|---|---|
| początkowa liczba neuronów sieci | $m_{start} = 2$ |
| maksymalna liczba epok uczenia | $e_{\max} = 10\,000$ |
| sposób wyznaczania neuronu zwyciężającego (o numerze $j_x$) | zależność (2.3) z euklidesową miarą odległości (2.4) |
| współczynnik uczenia sieci[1)] w (2.2) | $\eta(e) = 7 \cdot 10^{-4} \div 10^{-6}$ |
| funkcja sąsiedztwa topologicznego $S(j, j_x, k)$ w (2.2) | funkcja sąsiedztwa gaussowskiego (2.9) z miarą odległości Czebyszewa (2.6) |
| promień sąsiedztwa topologicznego w (2.9) | $\lambda(e) = 2$ |
| parametr odpowiedzialny za usuwanie pojedynczych mało aktywnych neuronów | $lzw_{\min} = 2$ |
| parametr odpowiedzialny za wstawianie dodatkowych neuronów do struktury | $lzw_{\max} = 2$ |
| parametr regulujący mechanizm usuwania małych podstruktur sieci | $m_{\min} = 2$ |
| parametr regulujący mechanizm usuwania połączeń topologicznych | $\alpha_{rozl} = 3$ |
| parametr regulujący mechanizm wstawiania połączeń topologicznych | $\alpha_{łącz} = 3$ |

[1)] parametr malejący liniowo po zakończeniu kolejnych epok uczenia *e*; stały dla danej epoki.




Rys. 5.1. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *Leukemia* bazujące na próbkach)

Analogicznie jak wyżej, przebieg procesu uczenia kolejnej sieci neuronowej, której zadaniem jest tym razem przyporządkowanie genów do odpowiednich grup (grupowanie bazujące na genach) ilustruje rys. 5.2. Na rys. 5.2a widoczne są zmiany liczby neuronów w trakcie uczenia (ostatecznie ustabilizowanej na poziomie $m = 1560$), natomiast na rys. 5.2b zmiany liczby podstruktur sieci neuronowej (ostatecznie ustabilizowanej na poziomie 79 podstruktur).



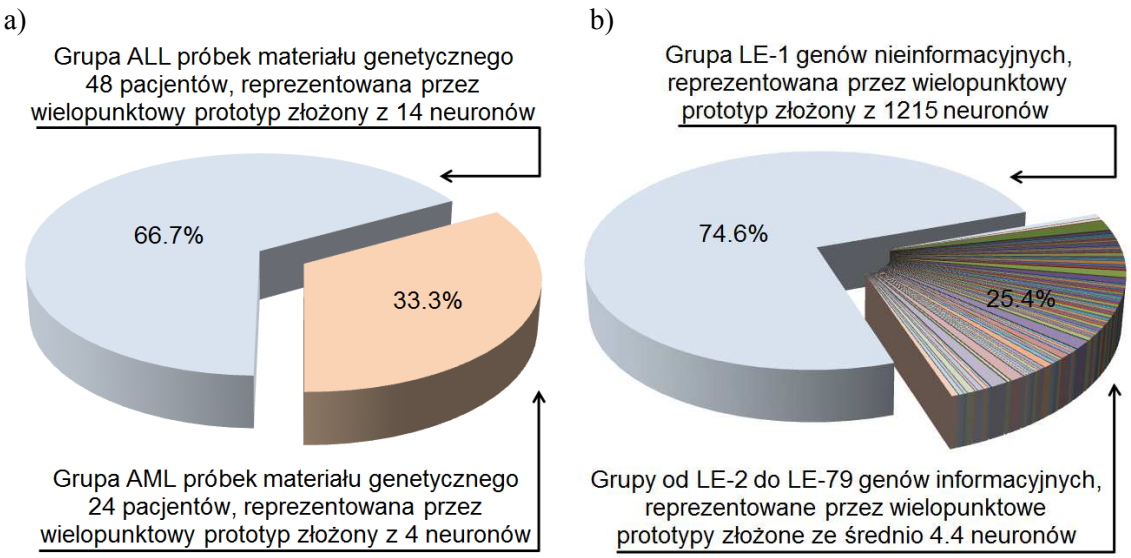

Rys. 5.2. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *Leukemia* bazujące na genach)

Po zakończeniu procesów uczenia sieci neuronowych została przeprowadzona kalibracja sieci, tzn. nadanie otrzymanym wielopunktowym prototypom grup (podstrukturom sieci neuronowych) odpowiednich etykiet klas. Kalibracja umożliwia weryfikację uzyskanych rezultatów i ocenę efektywności sieci neuronowej w rozważanym problemie.

W przypadku sieci neuronowej grupującej próbki danych, nadano następujące etykiety: ALL dla wielopunktowego prototypu składającego się z 14 neuronów (o numerach porządkowych od 1 do 14) oraz AML dla wielopunktowego prototypu złożonego z 4 neuronów (o numerach porządkowych od 15 do 18). Oba prototypy reprezentują grupy, odpowiednio, 48 i 24 próbek danych pacjentów. Proporcje podziału próbek danych na grupy ilustruje rys. 5.3a. W przypadku sieci neuronowej grupującej geny, każdemu z 79 wielopunktowych prototypów grup genów nadano etykietę według wzorca LE-*i*, gdzie *i* jest numerem porządkowym grupy od 1 do 79. Prototyp największej grupy (2663 genów) o etykiecie LE-1 składa się z 1215 neuronów o numerach porządkowych od 267 do 1481. Pozostałe 78 prototypów posiada od 2 do 24 neuronów (średnio 4.4

neuronów na prototyp) i reprezentuje grupy od 5 do 51 genów (średnio 11.6 genów na grupę).

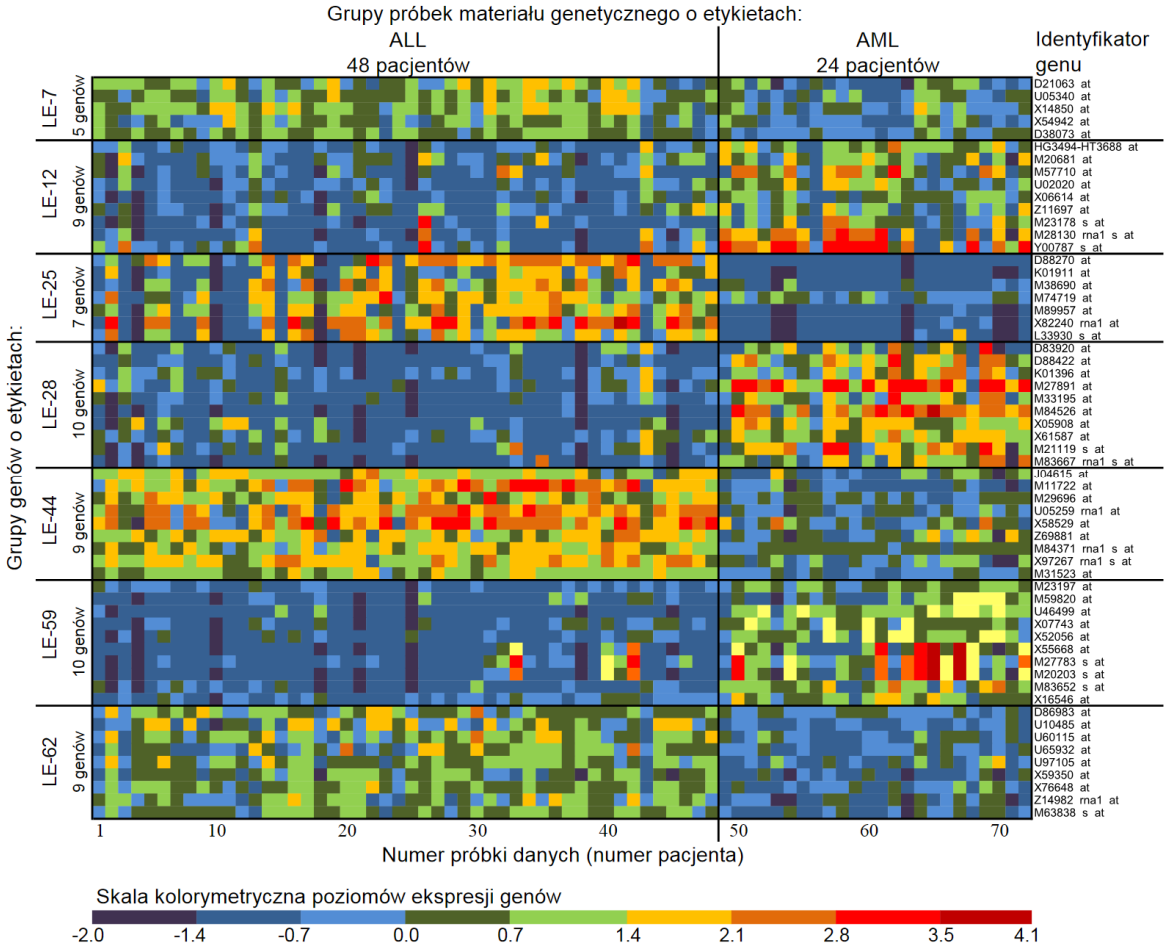
W literaturze (np. [69, 82]) panuje pogląd, że jedynie grupy o względnie niewielkiej liczbie genów mogą nieść istotną informację z punktu widzenia diagnostyki chorób nowotworowych. Geny takich grup nazywane są informacyjnymi (ang. *informative genes*), a ich całkowita liczba jest stosunkowo mała względem liczby pozostałych genów, uznawanych za nieinformacyjne (zwykle nie przekracza ona 10% [109] i [120]). Zakwalifikowanie danej grupy genów do kategorii informacyjnych lub nieinformacyjnych jest względnie uznaniowe, a granica pomiędzy tymi kategoriami ma charakter nieostry (zależy zwykle od liczby genów w grupie). W rozważanym problemie przyjęto, że największa grupa LE-1, której geny stanowią 74.6% całkowitej liczby genów w zbiorze *Leukemia* (patrz rys. 5.3b), jest grupą genów nieinformacyjnych i nie będzie brana pod uwagę podczas dalszej analizy otrzymanych rezultatów (w kolejnym podrozdziale). Pozostałe geny uznano za informacyjne (stanowią 25.4% całkowitej liczby genów w rozważanym zbiorze, a po uwzględnieniu pierwotnej liczby genów (część genów będących kopiami i zaszumionych zostało usuniętych z pierwotnego zbioru *Leukemia*, patrz komentarz w podrozdziale 5.1), stanowią 12.7% ogólnej liczby genów zbioru *Leukemia*).



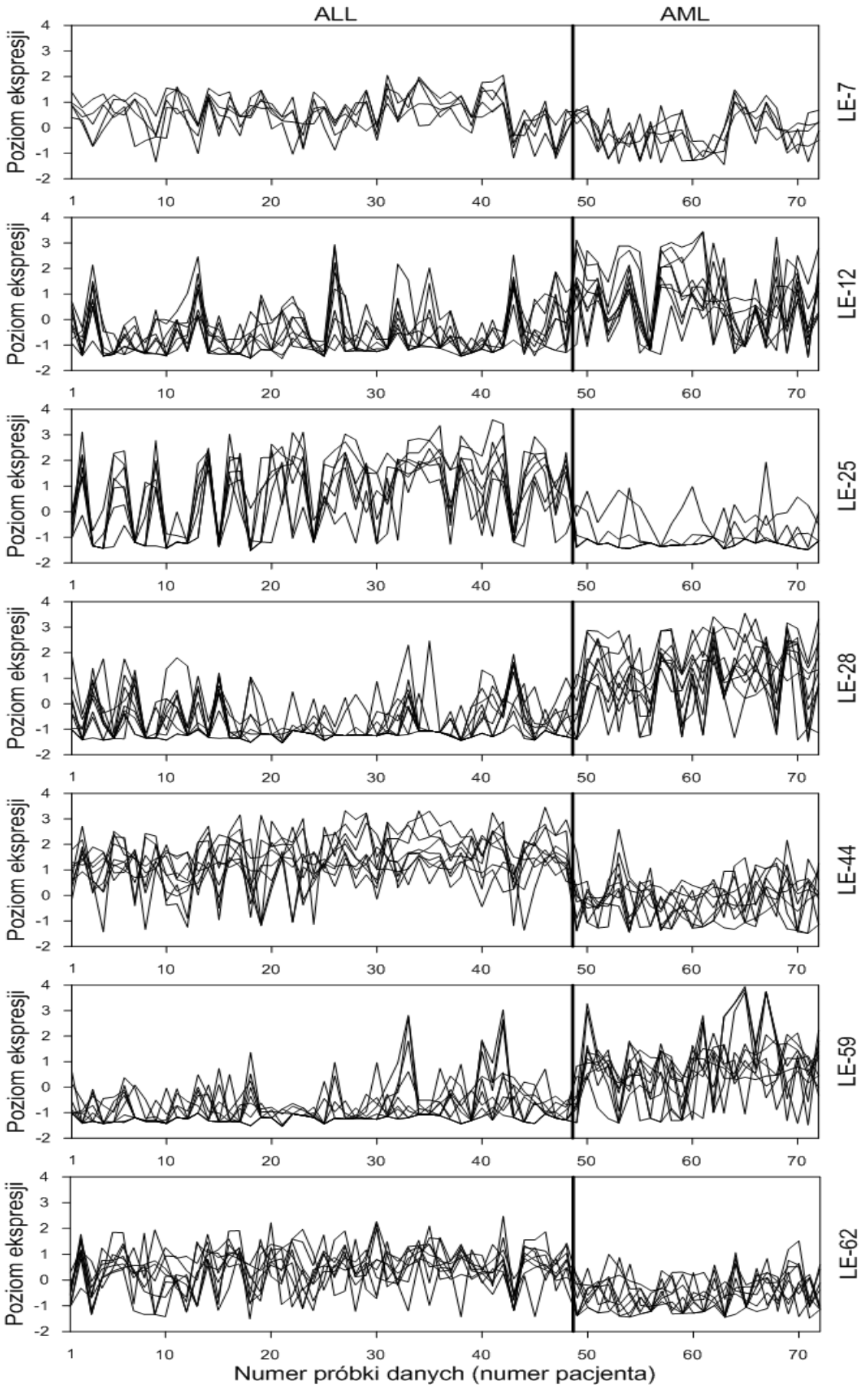

Rys. 5.3. Proporcje podziału danych zbioru *Leukemia* na grupy otrzymane w rezultacie grupowania bazującego na próbkach (a) oraz bazującego na genach (b)

Z powodu dużej wymiarowości zbioru danych *Leukemia*, nie jest możliwa jego graficzna prezentacja oraz prezentacja wszystkich otrzymanych grup genów. Na rys. 5.4 przedstawiono jedynie kilka wybranych grup o następujących etykietach: LE-7, LE-12, LE-25, LE-28, LE-44, LE-59 i LE-62. Kolorowymi punktami oznaczono poziomy eks-

presji genów (w skali kolorymetrycznej). Czarną linią pionową oznaczono granicę pomiędzy dwoma grupami próbek danych o etykietach ALL i AML (będącą rezultatem grupowania bazującego na próbkach). Z kolei, czarnymi liniami poziomymi oznaczono granice pomiędzy grupami genów (rezultat grupowania bazującego na genach). Po prawej stronie rysunku przedstawiono – dla wybranych grup genów – ich identyfikatory, które zostaną w kolejnym podrozdziale wykorzystane do analizy funkcji pełnionych przez nie w organizmie człowieka, co może mieć istotne znaczenie dla badań patomorfologicznych chorób nowotworowych. Szersza analiza znaczenia genów nie będzie tu rozważana, gdyż może ona być przeprowadzana jedynie z wykorzystaniem metod medycznych, co zdecydowanie wykracza poza ramy niniejszej rozprawy. Dodatkowo, jako dopełnienie wizualizacji poziomów ekspresji genów z rys. 5.4, na rys. 5.5 przedstawiono poziomy ekspresji genów z poszczególnych grup w formie nakładających się wzajemnie krzywych. Rysunek ten pozwala na wizualną ocenę podobieństwa wartości poziomów ekspresji genów należących do tych samych grup.



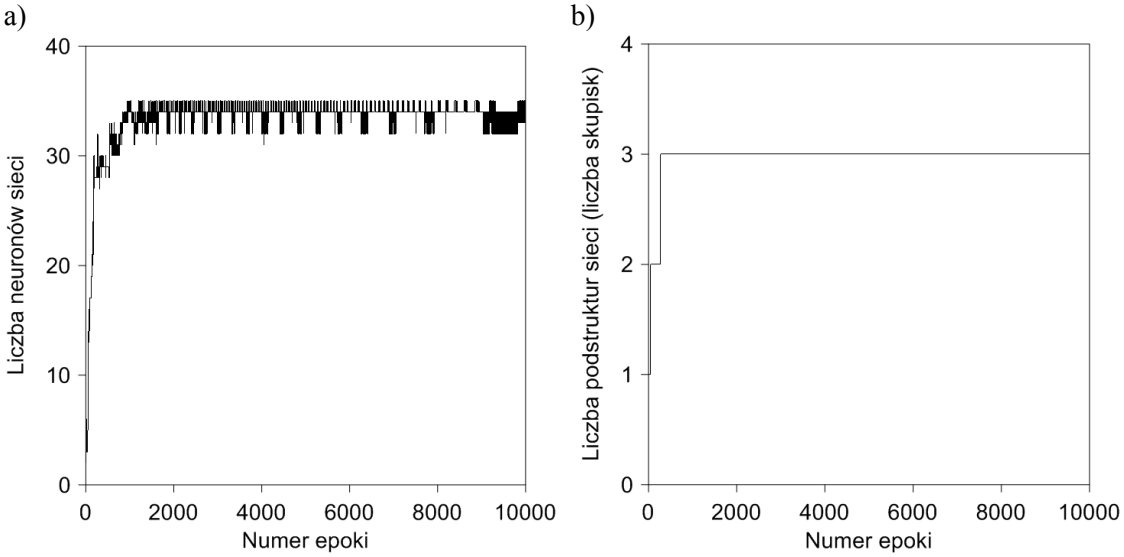

Rys. 5.4. Graficzna ilustracja poziomów ekspresji genów ze zbioru *Leukemia* oraz granic podziału pomiędzy grupami próbek danych i grupami genów




Rys. 5.5. Graficzna ilustracja poziomów ekspresji genów z rys. 5.4

### 5.2.2. Procesy uczenia sieci neuronowych dla zbioru danych *MLL*

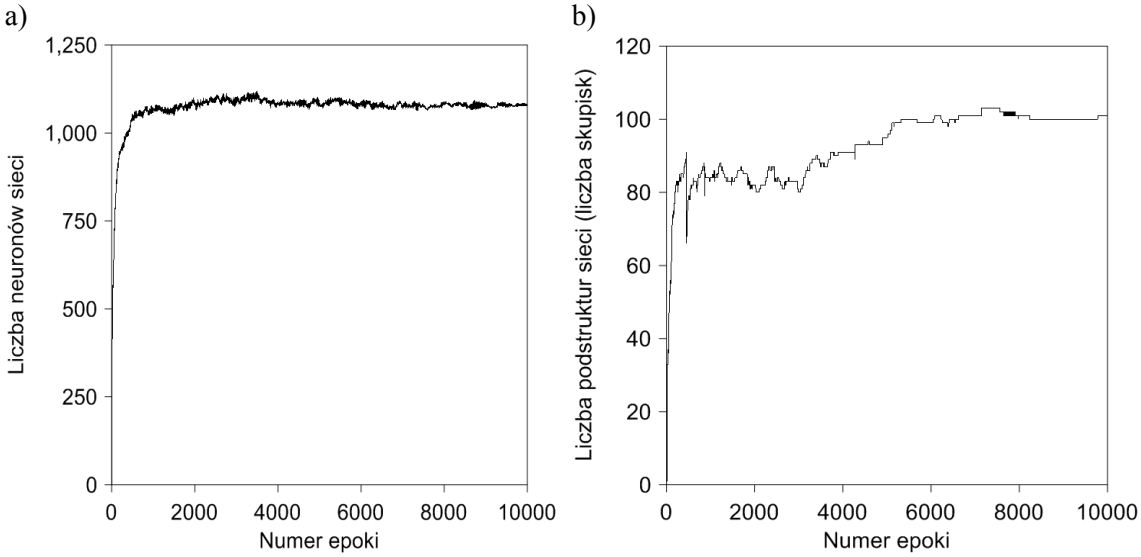
Procesy uczenia proponowanych sieci neuronowych dla danych ze zbioru *MLL* zostały przeprowadzone w sposób analogiczny, jak w przypadku zbioru *Leukemia*. Na rys. 5.6a i rys. 5.6b przedstawiono przebiegi zmian liczby neuronów oraz liczby podstruktur podczas uczenia sieci neuronowej grupującej próbki danych. Ostateczna liczba neuronów wynosi $m = 33$, natomiast liczba podstruktur sieci (równa 3) jest równa liczbie klas próbek danych. Z kolei, na rys. 5.7a i rys. 5.7b przedstawiono analogiczne przebiegi dla sieci neuronowej grupującej geny. Końcowa liczba neuronów sieci wynosi $m = 1082$, a liczba jej podstruktur jest równa 101.



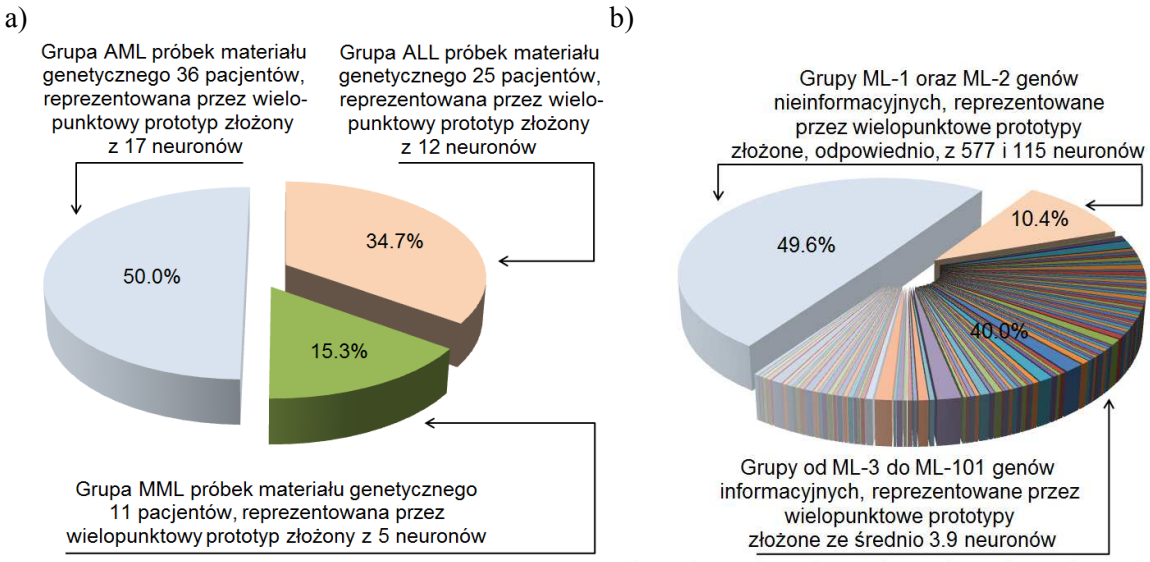

Rys. 5.6. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *MLL* bazujące na próbkach)

Po zakończeniu procesów uczenia sieci neuronowych przeprowadzono ich kalibrację. Trzem wielopunktowym prototypom reprezentującym grupy próbek danych (liczące, odpowiednio, 25, 36 i 11 próbek) nadano kolejno etykiety: ALL, AML i MLL. Prototypy składają się, odpowiednio, z 12 neuronów o numerach od 17 do 28, 17 neuronów o numerach od 1 do 16 oraz z 5 neuronów o numerach od 29 do 33. Proporcje podziału próbek danych na grupy ilustruje rys. 5.8a. Z kolei, każdemu ze 101 prototypów grup genów nadano etykietę według wzorca ML-*i* (gdzie *i* jest numerem grupy od 1 do 101). Prototypy o etykietach ML-1 i ML-2 dwóch największych grup (odpowiednio, 1226 i 258 genów uznanych za nieinformacyjne, podobnie jak to miało miejsce w przypadku grupy LE-1) składają się z 577 neuronów o numerach od 399 do 975 (prototyp ML-1) i

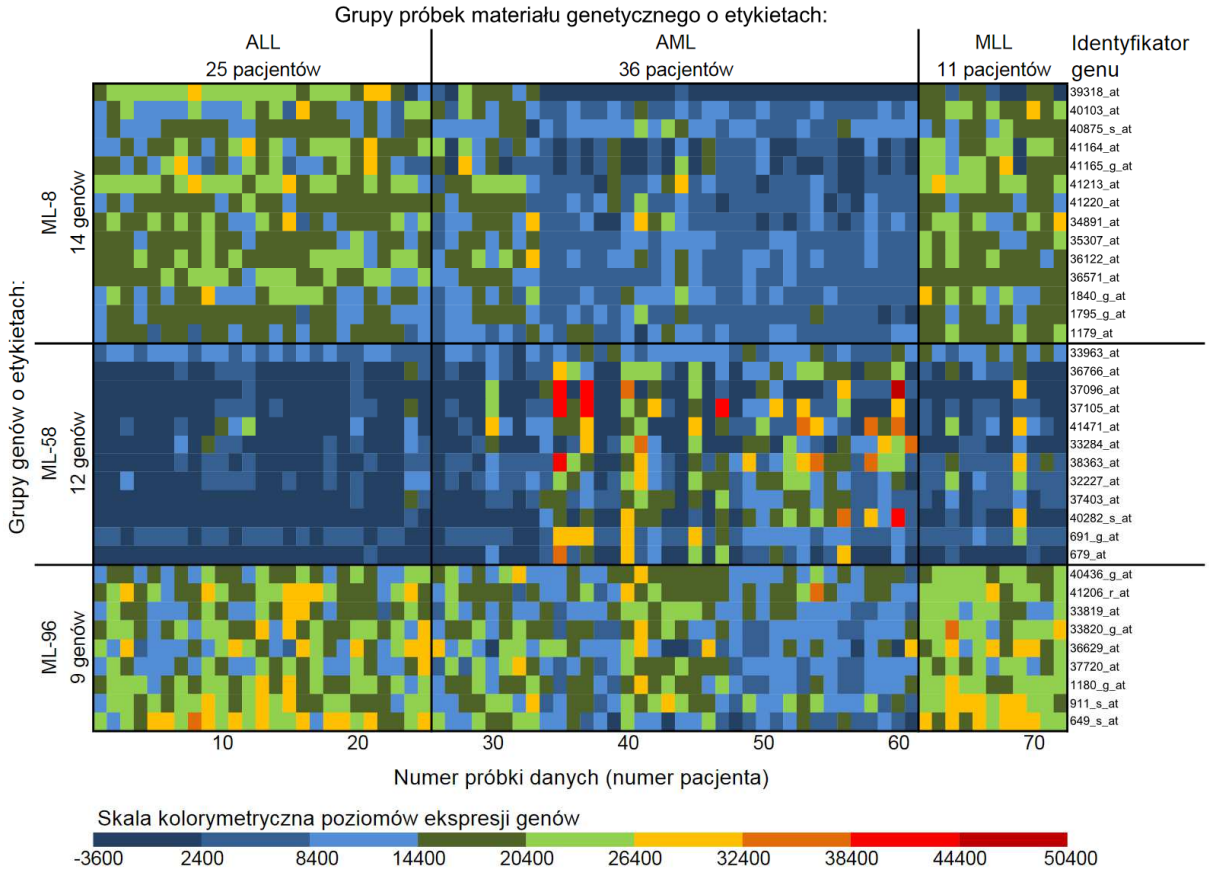
115 neuronów o numerach od 214 do 328 (prototyp ML-2). Pozostałe 99 prototypów posiada od 2 do 18 neuronów (średnio 3.9 neuronów na prototyp) i reprezentują grupy o liczbie od 4 do 41 genów informacyjnych (średnio 10 genów na grupę) (patrz również na rys. 5.8b; geny informacyjne stanowią 40% całkowitej liczby genów rozważanego zbioru danych oraz 7.9% liczby genów pierwotnego zbioru *MLL*). Podobnie jak wyżej, z powodu dużej wymiarowości zbioru danych *MLL*, graficzna prezentacja zbioru i wszystkich otrzymanych grup genów nie jest możliwa. Na rysunkach 5.9 i 5.10 przedstawiono jedynie wybrane grupy o etykietach: ML-8, ML-58 i ML-96.



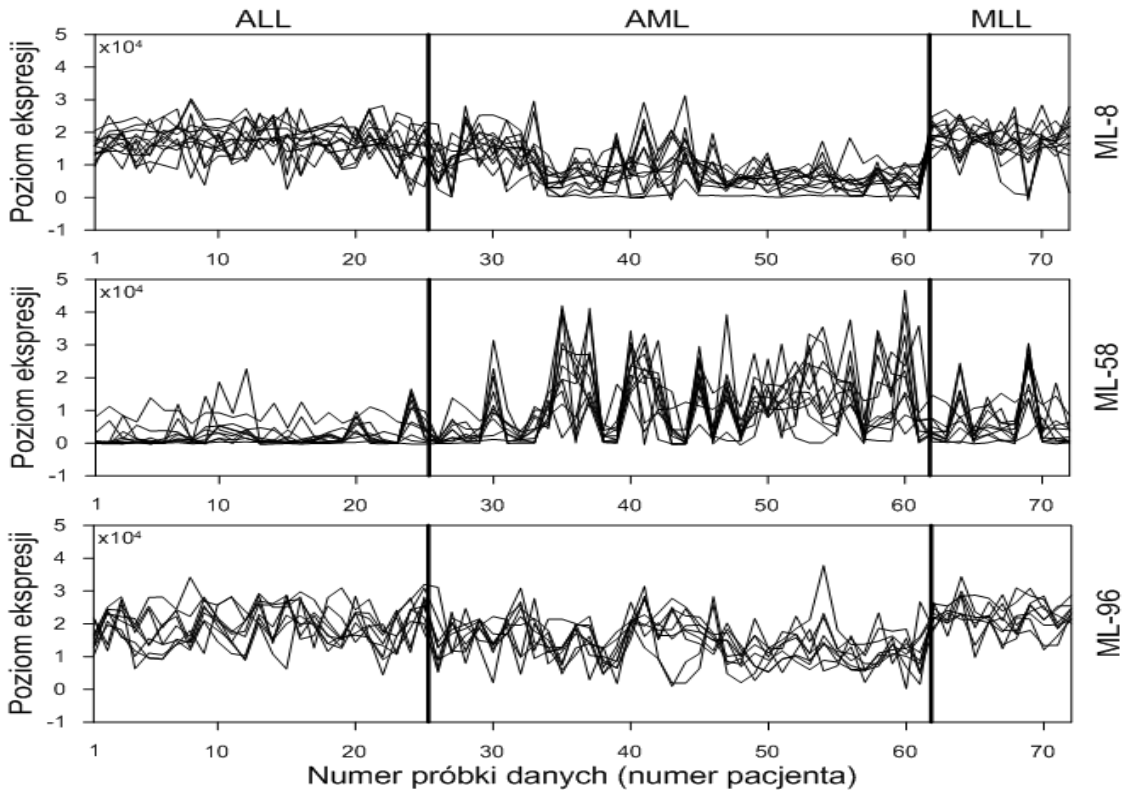

Rys. 5.7. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *MLL* bazujące na genach)




Rys. 5.8. Proporcje podziału danych zbioru *MLL* na grupy otrzymane w rezultacie grupowania bazującego na próbkach (a) oraz bazującego na genach (b)




Rys. 5.9. Graficzna ilustracja poziomów ekspresji genów ze zbioru *MLL* oraz granic podziału pomiędzy grupami próbek danych i grupami genów




Rys. 5.10. Graficzna ilustracja poziomów ekspresji genów z rys. 5.9.

## 5.3. Ocena efektywności grupowania danych

W tym podrozdziale zostanie przeprowadzona analiza wyników grupowania bazującego na próbkach i bazującego na genach, otrzymanych dla rozważanych zbiorów danych. Analiza ma na celu weryfikację efektywności proponowanej metody w grupowaniu danych opisujących ekspresję genów w problemie diagnozowania nowotworowych chorób układu krwionośnego.

***Ocena efektywności grupowania danych bazującego na próbkach***

Oba rozważane zbiory danych tzn. *Leukemia* i *MLL* zawierają informacje o rzeczywistej przynależności próbek danych do odpowiednich klas. Należy podkreślić, że informacja ta nie była wykorzystywana podczas uczenia sieci neuronowych (jest to proces uczenia nienadzorowanego), jednakże teraz zostanie użyta do bezpośredniej weryfikacji rezultatów grupowania danych bazującego na próbkach. Statystyczne zestawienie osiągniętych wyników zawiera tabela 5.2 (dla przypadku zbioru danych *Leukemia*) i tabela 5.3 (dla zbioru danych *MLL*). Liczba poprawnych decyzji dotyczących przyporządkowania próbek danych zbioru *Leukemia* do odpowiednich klas wynosi 98.6%, przy czym warto zwrócić uwagę na bezbłędne przyporządkowanie próbek do klasy ALL (100% poprawnych decyzji). Podobnie, liczba poprawnych decyzji o przyporządkowaniu próbek danych zbioru *MLL* jest równie wysoka i wynosi 86.1%. Tym razem, wszystkie próbki danych klasy AML zostały prawidłowo przyporządkowane (100% poprawnych decyzji). Słabszy wynik ogólny (niemniej i tak bardzo dobry) wynika ze stosunkowo niskiej liczby poprawnych decyzji dla próbek klasy MLL (równa 55%). Błędne przyporządkowanie części próbek grupy MLL (9 próbek danych) do grup ALL i AML może wynikać z faktu, że białaczki typu MLL są szczególnymi przypadkami białaczek ALL i AML. Można przypuszczać, że podobieństwo genów pomiędzy próbkami ALL i ich szczególnymi przypadkami MLL (oraz analogicznie pomiędzy próbkami AML i ich szczególnymi przypadkami MLL) jest stosunkowo duże, co utrudnia ich rozróżnianie podczas grupowania (dla przykładu, na rys. 5.9 wyraźnie widoczne jest podobieństwo poziomów ekspresji genów z grup ALL i MLL).

Tabela 5.2. Rezultaty grupowania danych bazującego na próbkach (zbiór danych *Leukemia*)

| Etykieta klasy | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie: | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|
| | | ALL | AML | | | |
| ALL | 47 | 47 | 0 | 47 | 0 | 100% |
| AML | 25 | 1 | 24 | 24 | 1 | 96% |
| **RAZEM** | **72** | **48** | **24** | **71** | **1** | **98.6%** |

Tabela 5.3. Rezultaty grupowania danych bazującego na próbkach (zbiór danych *MLL*)

| Etykieta klasy | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie: | | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|---|
| | | ALL | AML | MLL | | | |
| ALL | 24 | 23 | 1 | 0 | 23 | 1 | 95.8% |
| AML | 28 | 0 | 28 | 0 | 28 | 0 | 100% |
| MLL | 20 | 2 | 7 | 11 | 11 | 9 | 55% |
| **RAZEM** | **72** | **25** | **36** | **11** | **62** | **10** | **86.1%** |

***Ocena efektywności grupowania danych bazującego na genach***

Bezpośrednia ocena rezultatów grupowania bazującego na genach nie jest możliwa, gdyż zbiory *Leukemia* i *MLL* nie zawierają żadnych informacji o przynależności genów do klas. Zawierają natomiast unikatowe identyfikatory genów, powszechnie znane i stosowane w badaniach molekularnych komórek nowotworowych. Mogą one być wykorzystane do dalszej analizy otrzymanych wyników grupowania genów, mającej na celu określenie pojedynczych genów lub grup genów mających bezpośredni związek z chorobą nowotworową (np. odpowiedzialnych za jej powstawanie, czas i dynamikę rozwoju, itp.) lub też wykrywanie nieznanych rodzajów chorób, mutacji, podtypów [42]. Oczywiście tak głęboka analiza powinna być przeprowadzona metodami medycznymi przez eksperta z dziedziny inżynierii genetycznej lub molekularnej i zdecydowanie wykracza poza ramy niniejszej rozprawy. Natomiast teraz, na potrzeby oceny efektywności proponowanego narzędzia, zostaną przedstawione rezultaty podstawowej analizy statystycznej funkcji genów informacyjnych (tzw. analizy funkcyjnej lub funkcjonalnej genów) występujących w poszczególnych grupach genów w zbiorach danych *Leukemia* i *MLL* (jak już wcześniej wspomniano, grupa genów nieinformacyjnych nie była uwzględniana podczas tej analizy). Należy podkreślić, że prezentacja rezultatów otrzy-

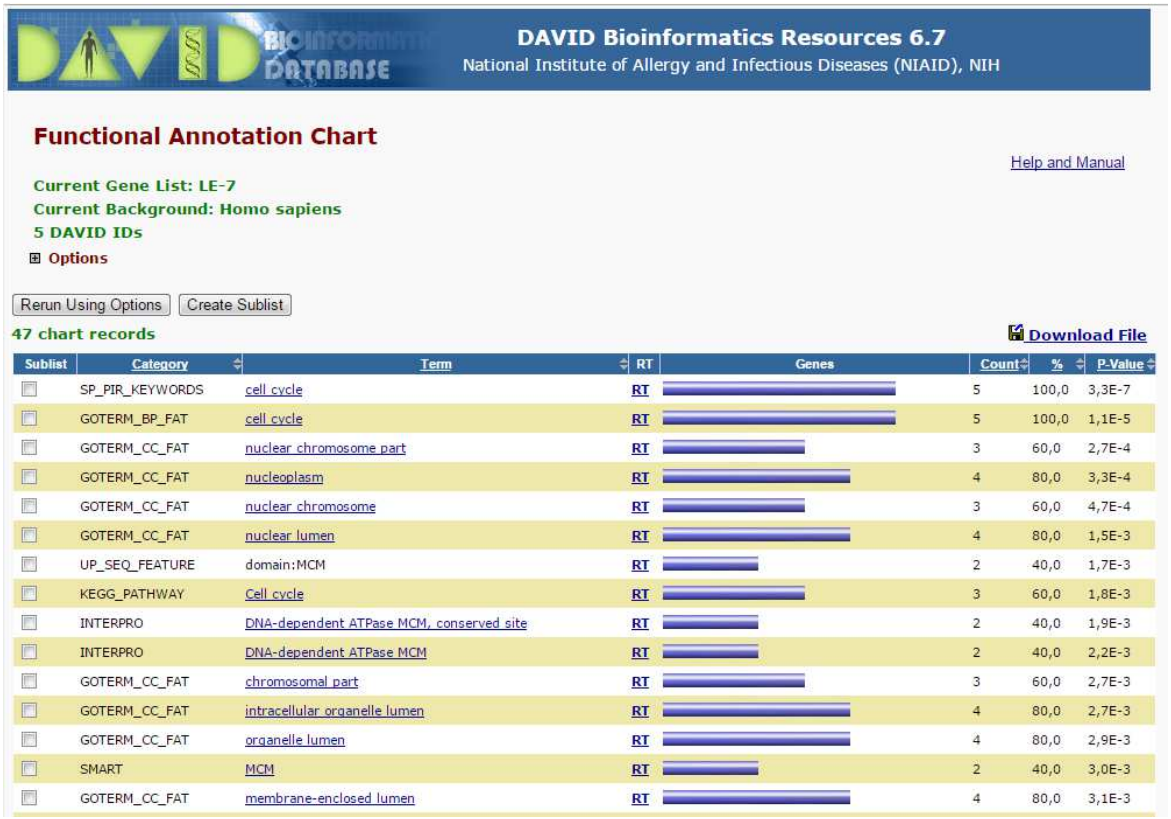
manych dla wszystkich grup genów byłaby nieczytelna (ze względu na dużą liczbę tych grup) i w zasadzie nie jest potrzebna. Uznano, że do oceny efektywności proponowanej sieci neuronowej jako techniki grupowania wystarczająca będzie prezentacja rezultatów otrzymanych dla dwóch grup genów, po jednej wybranej z każdego zbioru danych.

Do przeprowadzenia analizy funkcji genów wykorzystano bazę genów wzorcowych DAVID (ang. *Database for Annotation, Visualization and Integrated Discovery*) [28], dostępną na serwerze Laboratory of Immunopathogenesis and Bioinformatics, National Cancer Institute at Frederick w USA (http://david.abcc.ncifcrf.gov) oraz współpracujące z nią oprogramowanie do analizy funkcyjnej genów DAVID Functional Annotation Tool. Wykonano tzw. dokładne testy Fishera (ang. *Fisher Exact Test*) dla badanych grup genów, których celem było wyznaczenie wszystkich statystycznie znaczących funkcji genów należących do danej grupy. Przyjmuje się, że funkcja genu jest statystycznie znacząca, jeżeli jej poziom istotności (ang. *p-value*) wyznaczony testem Fishera jest mniejszy od 0.05 [23].

Na rys. 5.11 przedstawiono w postaci tabelarycznej rezultaty testu Fishera dla grupy LE-7 (5 genów) wybranej ze zbioru danych *Leukemia*. Każdy wiersz tabeli zawiera informacje dotyczące danej funkcji genów (wymienionej w kolumnie „*Term”*), m.in.: listę identyfikatorów tzw. genów funkcyjnych, tzn. pełniących daną funkcję w grupie (kolumna „*Genes”*; niebieski pasek jest łączem do kolejnej podstrony WWW z listą genów), liczbę genów funkcyjnych (kolumna „*Count”*), procentowy udział genów funkcyjnych w grupie (kolumna *%*) oraz poziom istotności danej funkcji (kolumna „*P-Value”*).

W ogólnym przypadku, funkcje genów danej grupy o najmniejszym poziomie istotności *p-value* mają statystycznie największe znaczenie. Grupa genów jest tym lepsza, im procentowy udział genów pełniących takie funkcje w grupie jest większy. W grupie LE-7 wszystkie geny pełnią funkcję związaną z cyklem komórkowym (udział 100%; funkcja *cell cycle*, dla której *p-value* jest najmniejsze i wynosi $3.3 \cdot 10^{-7}$). Jest to najważniejsza funkcja genów w tej grupie. Pozostałe, statystycznie najbardziej znaczące funkcje genów, dla których *p-value* jest mniejsze od 0.001, są związane z konstrukcją jądra komórkowego (funkcje *nuclear chromosome part*, *nucleoplasm*, *nuclear lumen* oraz *nuclear chromosome*). Tabela 5.4 przedstawia zestawienie tych funkcji oraz genów funkcyjnych. Ze względu na stosunkowo duży procentowy udział genów pełniących w grupie LE-7 tę samą funkcję należy uznać ją za bardzo dobrą (potencjalnie wartościową

dla dalszych badań o charakterze medycznym). Podobne rezultaty otrzymano dla zdecydowanej większości grup genów informacyjnych zbioru *Leukemia* (jak już wyżej wspomniano, ze względu na dużą liczbę tych grup, nie będą one dalej prezentowane).



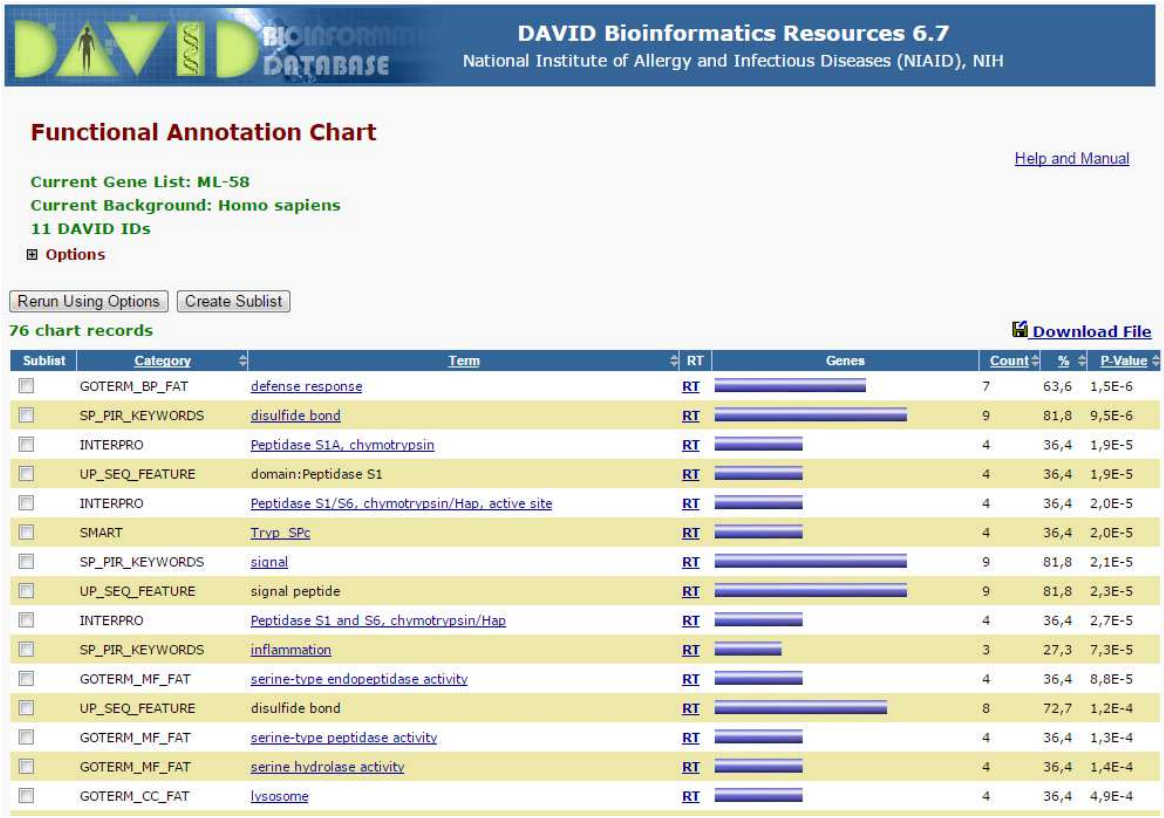

| Sublist | Category | Term | RT | Genes | Count | % | P-Value |
|---|---|---|---|---|---|---|---|
| | SP_PIR_KEYWORDS | cell cycle | RT | | 5 | 100,0 | 3,3E-7 |
| | GOTERM_BP_FAT | cell cycle | RT | | 5 | 100,0 | 1,1E-5 |
| | GOTERM_CC_FAT | nuclear chromosome part | RT | | 3 | 60,0 | 2,7E-4 |
| | GOTERM_CC_FAT | nucleoplasm | RT | | 4 | 80,0 | 3,3E-4 |
| | GOTERM_CC_FAT | nuclear chromosome | RT | | 3 | 60,0 | 4,7E-4 |
| | GOTERM_CC_FAT | nuclear lumen | RT | | 4 | 80,0 | 1,5E-3 |
| | UP_SEQ_FEATURE | domain:MCM | RT | | 2 | 40,0 | 1,7E-3 |
| | KEGG_PATHWAY | Cell cycle | RT | | 3 | 60,0 | 1,8E-3 |
| | INTERPRO | DNA-dependent ATPase MCM, conserved site | RT | | 2 | 40,0 | 1,9E-3 |
| | INTERPRO | DNA-dependent ATPase MCM | RT | | 2 | 40,0 | 2,2E-3 |
| | GOTERM_CC_FAT | chromosomal part | RT | | 3 | 60,0 | 2,7E-3 |
| | GOTERM_CC_FAT | intracellular organelle lumen | RT | | 4 | 80,0 | 2,7E-3 |
| | GOTERM_CC_FAT | organelle lumen | RT | | 4 | 80,0 | 2,9E-3 |
| | SMART | MCM | RT | | 2 | 40,0 | 3,0E-3 |
| | GOTERM_CC_FAT | membrane-enclosed lumen | RT | | 4 | 80,0 | 3,1E-3 |

Rys. 5.11. Rezultaty testu Fishera dla grupy genów LE-7 wybranej ze zbioru danych *Leukemia*, generowane przez program DAVID Functional Annotation Tool

Tabela 5.4. Statystycznie najbardziej znaczące funkcje genów grupy LE-7, wybrane spośród rezultatów testu Fishera przedstawionych na rys. 5.11

| Lp. | Nazwa funkcji genów | Geny funkcyjne w grupie genów LE-7 | | | Poziom istotności *p-value* |
|---|---|---|---|---|---|
| | | Ilość | Identyfikatory genów | Udział | |
| 1 | *cell cycle* (kategoria SP_PIR_KEYWORDS) | 5 | D21063_at, U05340_at, X14850_at, X54942_at, D38073_at | 100% | $3.3 \cdot 10^{-7}$ |
| 2 | *cell cycle* (kategoria GOTERM_BP_FAT) | 5 | D21063_at, U05340_at, X14850_at, X54942_at, D38073_at | 100% | $1.1 \cdot 10^{-5}$ |
| 3 | *nuclear chromosome part* | 3 | X14850_at, D21063_at, D38073_at | 60% | $2.7 \cdot 10^{-4}$ |
| 4 | *nucleoplasm* | 4 | D21063_at, U05340_at, X14850_at, D38073_at | 80% | $3.3 \cdot 10^{-4}$ |
| 5 | *nuclear chromosome* | 3 | X14850_at, D21063_at, D38073_at | 60% | $4.7 \cdot 10^{-4}$ |

Na rys. 5.12 widoczne są rezultaty testu Fishera dla kolejnej, przykładowej grupy ML-58 (12 genów, przy czym gen „679_at" został uznany przez program DAVID za równoważny genowi „37105_at" i pominięty w teście; ostatecznie badana grupa liczyła 11 genów), tym razem wybranej ze zbioru danych *MLL*. Tabela 5.5 przedstawia szczegółowe zestawienie najważniejszych funkcji oraz genów funkcyjnych tej grupy.




| Sublist | Category | Term | RT | Genes | Count | % | P-Value |
|---|---|---|---|---|---|---|---|
| | GOTERM_BP_FAT | defense response | RT | | 7 | 63,6 | 1,5E-6 |
| | SP_PIR_KEYWORDS | disulfide bond | RT | | 9 | 81,8 | 9,5E-6 |
| | INTERPRO | Peptidase S1A, chymotrypsin | RT | | 4 | 36,4 | 1,9E-5 |
| | UP_SEQ_FEATURE | domain:Peptidase S1 | RT | | 4 | 36,4 | 1,9E-5 |
| | INTERPRO | Peptidase S1/S6, chymotrypsin/Hap, active site | RT | | 4 | 36,4 | 2,0E-5 |
| | SMART | Tryp_SPc | RT | | 4 | 36,4 | 2,0E-5 |
| | SP_PIR_KEYWORDS | signal | RT | | 9 | 81,8 | 2,1E-5 |
| | UP_SEQ_FEATURE | signal peptide | RT | | 9 | 81,8 | 2,3E-5 |
| | INTERPRO | Peptidase S1 and S6, chymotrypsin/Hap | RT | | 4 | 36,4 | 2,7E-5 |
| | SP_PIR_KEYWORDS | inflammation | RT | | 3 | 27,3 | 7,3E-5 |
| | GOTERM_MF_FAT | serine-type endopeptidase activity | RT | | 4 | 36,4 | 8,8E-5 |
| | UP_SEQ_FEATURE | disulfide bond | RT | | 8 | 72,7 | 1,2E-4 |
| | GOTERM_MF_FAT | serine-type peptidase activity | RT | | 4 | 36,4 | 1,3E-4 |
| | GOTERM_MF_FAT | serine hydrolase activity | RT | | 4 | 36,4 | 1,4E-4 |
| | GOTERM_CC_FAT | lysosome | RT | | 4 | 36,4 | 4,9E-4 |

Rys. 5.12. Rezultaty testu Fishera dla grupy genów ML-58 wybranej ze zbioru danych *MLL*, generowane przez program DAVID Functional Annotation Tool

Tabela 5.5. Statystycznie najbardziej znaczące funkcje genów grupy ML-58, wybrane spośród rezultatów testu Fishera przedstawionych na rys. 5.12

| Lp. | Nazwa funkcji genów | Geny funkcyjne w grupie genów ML-58 | | | Poziom istotności *p-value* |
|---|---|---|---|---|---|
| | | Ilość | Identyfikatory genów | Udział | |
| 1 | *defense response* | 7 | 41471_at, 38363_at, 37403_at, 33963_at, 37105_at, 40282_s_at, 33284_at | 63.6% | $1.5 \cdot 10^{-6}$ |
| 2 | *disulfide bond* | 9 | 38363_at, 33963_at, 37105_at, 40282_s_at, 37096_at, 33284_at, 691_g_at, 36766_at, 32227_at | 81.8% | $9.5 \cdot 10^{-6}$ |

Statystycznie najbardziej znaczące funkcje genów w grupie ML-58 są związane z odpowiedzią immunologiczną komórek i białkami rodziny Dsb (ang. *Disulfide bond*) (dla przyjętego *p-value* nie większego niż $10^{-5}$ są to dwie funkcje *defense response* oraz *disulfide bond*). W przypadku tej grupy, procentowy udział genów pełniących wyżej wymienione funkcje w grupie jest relatywnie wysoki (odpowiednio, 63.6% i 81.8%). Zatem, podobnie jak w przypadku grupy LE-7, tym razem grupę ML-58 należy również uznać za potencjalnie wartościową dla dalszych badań medycznych. Wyniki dla zdecydowanej większości pozostałych grup genów informacyjnych ze zbioru *MLL* były równie dobre, tak jak to miało miejsce w przypadku zbioru danych *Leukemia*.

Podsumowując, otrzymane rezultaty grupowania danych bazującego na próbkach oraz bezpośrednia analiza decyzji sieci neuronowej (odnośnie przyporządkowania poszczególnych próbek danych z obu rozważanych zbiorów danych *Leukemia* i *MLL* do odpowiednich klas) jednoznacznie wskazują na bardzo wysoką efektywność proponowanego narzędzia (średnio ponad 92% poprawnych decyzji). Z kolei, analiza rezultatów grupowania bazującego na genach równie wyraźnie wskazuje, że proponowane narzędzie jest zdolne do wyszukiwania grup genów niosących potencjalnie istotne informacje z medycznego punktu widzenia (tzn. grup o niewielkiej liczbie genów, w których udział genów pełniących te same funkcje jest relatywnie wysoki). A zatem można rozważać w przyszłości efektywne wykorzystanie metody m.in. do wspomagania badań patomorfologicznych chorób nowotworowych. Aby potwierdzić wykazaną wysoką efektywność proponowanego narzędzia, w dalszej części rozdziału zostanie ono porównane z innymi, do pewnego stopnia alternatywnymi technikami.

## 5.4. Analiza porównawcza

Analiza porównawcza proponowanej, uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze (oznaczonej dalej krótko USOM) z innymi podejściami została przedstawiona w dwóch częściach, oddzielnie dla grupowania bazującego na próbkach i dla grupowania bazującego na genach.

### *Analiza porównawcza rezultatów grupowania danych bazującego na próbkach*

W pierwszej kolejności dokonano bezpośredniego porównania rezultatów grupowania danych bazującego na próbkach, otrzymanych dla zbiorów danych *Leukemia* i *MLL* z wykorzystaniem proponowanej sieci neuronowej USOM oraz pięciu alternatyw-

nych metod grupowania danych, tzn. algorytmów *k-means* [71, 86], EM (ang. *Expectation Maximization*) [27, 72], FFTA (ang. *Farthest First Traversal Algorithm*) [87], MSSRCC (ang. *Minimum Sum-Squared Residue Coclustering*) [23] oraz dynamicznej, samoorganizującej się sieci neuronowej z jednowymiarowym sąsiedztwem topologicznym DSOM [46]. Rezultaty grupowania danych zarówno bazujących na próbkach, jak i na genach dla metody MSSRCC zostały odczytane z pracy [23] (z rys. 5b i d; liczby błędnych i poprawnych decyzji podano w przybliżeniu, gdyż autorzy [23] nie podają ich bezpośrednio). Natomiast dla pozostałych metod, rezultaty grupowania danych otrzymano na podstawie własnych obliczeń z wykorzystaniem aplikacji WEKA (ang. *Waikato Environment for Knowledge Analysis*) dostępnej na stronach Uniwersytetu Waikato w Nowej Zelandii (http://www.cs.waikato.ac.nz/ml/weka) (dla metod *k-means*, EM i FFTA) oraz własnych implementacji (dla sieci DSOM i USOM). Zestawienie rezultatów grupowania danych bazującego na próbkach dla zbioru danych *Leukemia* i *MLL* przedstawiają, odpowiednio, tabela 5.6 i tabela 5.7.

W przypadku grupowania danych bazującego na próbkach dla zbioru *Leukemia*, proponowana sieć neuronowa (USOM) dała najlepsze rezultaty (98.6% poprawnych decyzji o przynależności poszczególnych próbek danych do odpowiednich klas), spośród wszystkich rozważanych alternatywnych technik. Warto zwrócić uwagę, że rezultaty metod MSSRCC i DSOM są również bardzo dobre (odpowiednio 95.8% i 93.1% poprawnych decyzji), natomiast pozostałych metod są poniżej oczekiwań (tzn. mniejsze od 80% poprawnych decyzji). Z kolei, w przypadku grupowania danych bazującego na próbkach dla zbioru *MLL*, podejście MSSRCC dało najlepszy rezultat (93% poprawnych decyzji), a wynik dla proponowanej sieci neuronowej USOM jest drugi w kolejności (86.1% poprawnych decyzji). Spośród pozostałych metod jedynie sieć DSOM uzyskała wynik powyżej oczekiwań (83.3% poprawnych decyzji). Należy tu jednak podkreślić, że z wyjątkiem sieci DSOM, pozostałe metody grupowania danych wymagają do poprawnego funkcjonowania podania z góry liczby grup próbek danych, co w bardzo istotny sposób faworyzuje je w porównaniu z sieciami USOM i DSOM (rezultaty USOM oraz DSOM zostały osiągnięte bez określania z góry liczby grup; sieci neuronowe automatycznie wykryły prawidłową liczbę grup w obu rozważanych zbiorach danych).

Tabela 5.6. Rezultaty grupowania bazującego na próbkach dla zbioru *Leukemia* z wykorzystaniem różnych metod grupowania danych

| Lp. | Metoda grupowania danych*) | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|
| 1 | *k-means* | 39 | 33 | 54.2% |
| 2 | EM | 53 | 19 | 73.6% |
| 3 | FFTA | 40 | 32 | 55.5% |
| 4 | MSSRCC | ~ 67 | ~ 5 | 93.1% |
| 5 | DSOM | 69 | 3 | 95.8% |
| 6 | **USOM** | **71** | **1** | **98.6%** |

*) EM – *Expectation Maximization* [27, 72], FFTA – *Farthest First Traversal Algorithm* [87], MSSRCC - *Minimum Sum-Squared Residue Coclustering* [23], DSOM – dynamiczna, samoorganizująca się sieć neuronowa z jednowymiarowym sąsiedztwem topologicznym [46], **USOM – proponowana w tej pracy uogólniona samoorganizująca się sieć neuronowa o ewoluującej, drzewopodobnej strukturze topologicznej.**

Tabela 5.7. Rezultaty grupowania bazującego na próbkach dla zbioru *MLL* z wykorzystaniem różnych metod grupowania danych

| Lp. | Metody grupowania danych*) | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|
| 1 | *k-means* | 53 | 19 | 73.6% |
| 2 | EM | 47 | 25 | 65.3% |
| 3 | FFTA | 46 | 26 | 63.9% |
| 4 | MSSRCC | ~ 67 | ~ 5 | 93.4% |
| 5 | DSOM | 60 | 12 | 83.3% |
| 6 | **USOM** | **62** | **10** | **86.1%** |

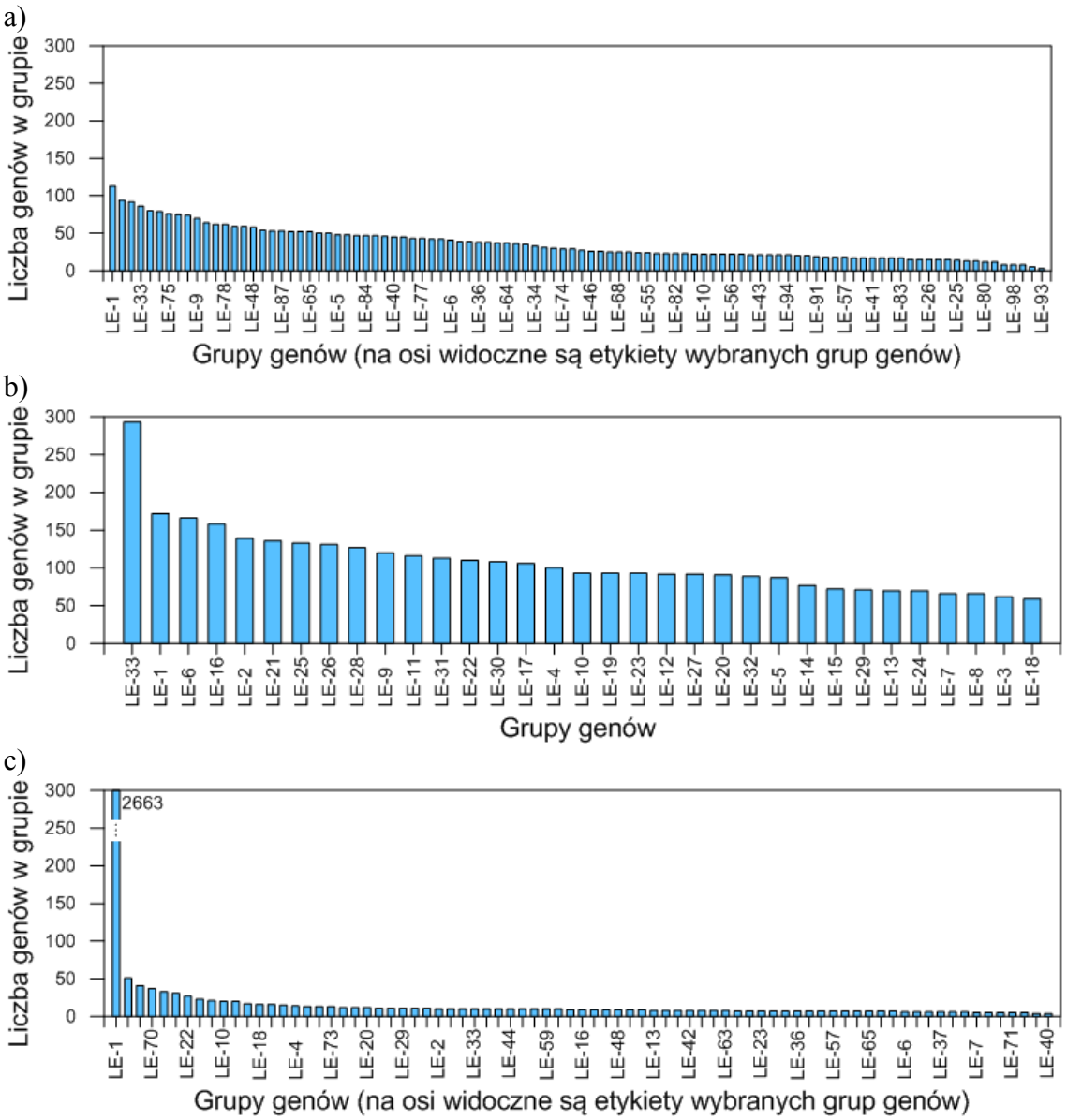
*) patrz tabela 5.6.

### *Analiza porównawcza rezultatów grupowania danych bazującego na genach*

W drugiej kolejności, porównano rezultaty grupowania danych bazującego na genach, otrzymanych dla zbiorów danych *Leukemia* i *MLL*. Tym razem jednak, do porównania z USOM wybrane zostały metody MSSRCC i DSOM, ponieważ dały one najlepsze wyniki grupowania bazującego na próbkach. Bezpośrednia analiza rezultatów grupowania bazującego na genach nie jest możliwa, gdyż informacja o prawidłowym przyporządkowaniu poszczególnych genów do określonych grup jest niedostępna. Przeprowadzono natomiast porównanie wyników pod względem dwóch subiektywnych kryteriów: a) liczby grup o względnie małej liczbie genów (jak już wspomniano wyżej, są to najcenniejsze grupy niosące potencjalnie istotne informacje z punktu widzenia diagno-

styki chorób nowotworowych [69, 82]) oraz b) procentowego udziału genów funkcyjnych w grupach (potencjalnie wartościowych dla dalszych badań o charakterze medycznym).

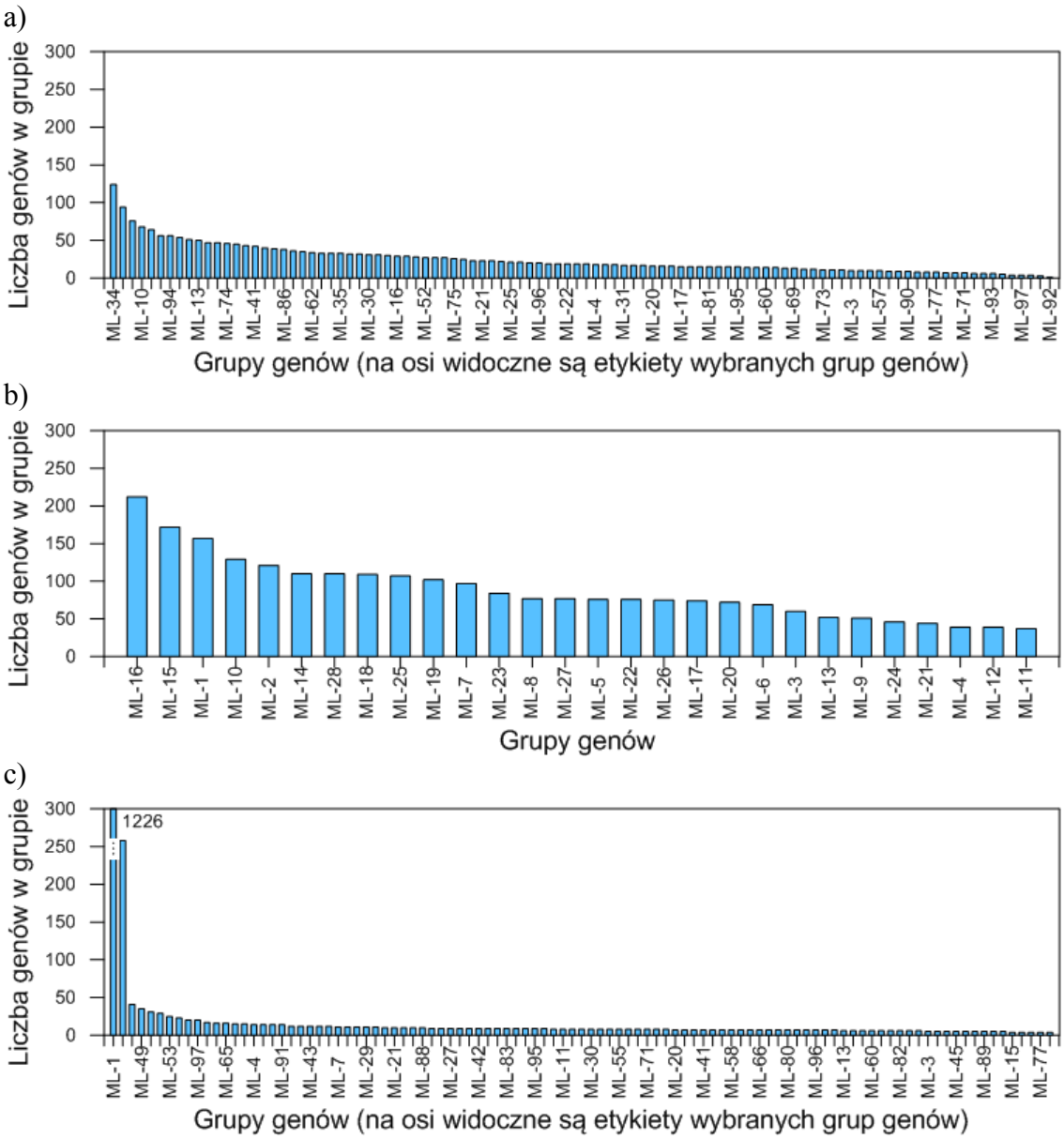
Na rys. 5.13 przedstawiono histogramy liczby genów w poszczególnych grupach genów w zbiorze danych *Leukemia* (wysokość pojedynczego słupka histogramu reprezentującego daną grupę genów jest proporcjonalna do liczby genów w tej grupie; słupki histogramu uporządkowano w kolejności od największej grupy genów do najmniejszej).




Rys. 5.13. Histogramy liczby genów w poszczególnych grupach genów w zbiorze danych *Leukemia*, otrzymanych z wykorzystaniem metod MSSRCC (a), DSOM (b) oraz USOM (c)

W przypadku metody MSSRCC liczba grup genów była założona arbitralnie z góry (100 grup). Otrzymane grupy genów zawierały od 3 do 113 genów (średnio 35.7 genów na grupę). Jak widać na rys. 5.13a, grupy o małej liczbie genów (przyjęto próg do 20 genów włącznie) stanowią 27% (27 grup) całkowitej liczby grup. W przypadku metod DSOM i USOM liczby grup genów zostały wykryte w sposób automatyczny (odpowiednio, 33 i 79 grup). Sieć neuronowa DSOM generowała względnie duże grupy od 59 do 293 genów (średnio 108 genów na grupę). Istnieją wątpliwości, czy tak duże grupy wniosą jakąkolwiek istotną informację do diagnostyki chorób nowotworowych. Z kolei, proponowana w pracy sieć neuronowa USOM dała największą liczbę względnie małych grup genów informacyjnych (od 5 do 51 genów w 78 grupach; średnio 11.6 genów na grupę) i jedną grupę genów nieinformacyjnych (2663 geny). Liczba grup zawierających co najwyżej 20 genów stanowiła 88.6% (70 grup) całkowitej liczby grup (patrz również rys. 5.13c). Nie ulega wątpliwości, że proponowana sieć neuronowa dała najlepsze rezultaty pod względem liczby grup zawierających małą liczbę genów. Jako jedyna spośród rozważanych metod wykryła jedno, duże skupisko genów nieinformacyjnych (istnienie takich skupisk genów w zbiorach wyrażających ekspresję genów zostało potwierdzone w literaturze, np. [73, 109, 120]).

Histogram liczby genów w grupach ze zbioru *MLL* przedstawia rys. 5.14. Wnioski z otrzymanych wyników są analogiczne do powyższych. W przypadku metody MSSRCC, grupy genów (ich liczba równa 100 była arbitralnie założona z góry) zawierały od 1 do 124 genów (średnio 24.7 genów na grupę). Grupy o bardzo małej liczbie genów (do 20 genów włącznie) stanowią 56% (56 grup) całkowitej liczby grup (patrz rys. 5.14a). Warto zwrócić uwagę, że metoda MSSRCC utworzyła stosunkowo kontrowersyjną grupę zawierającą 1 gen, co należy uznać za mankament tej metody (pojedynczy gen nie niesie żadnych informacji o relacjach zachodzących pomiędzy genami). W przypadku sieci DSOM, znowu otrzymano względnie duże grupy od 37 do 212 genów (28 grup; średnio 88.4 genów na grupę). Z kolei, proponowana sieć USOM dała największą liczbę względnie małych grup genów informacyjnych (od 4 do 41 genów w 99 grupach; średnio 10 genów na grupę) i dwie grupy genów nieinformacyjnych (w sumie 1484 geny). Liczba grup o wielkości nie większej niż 20 genów stanowiła 93.9% (93 grupy) całkowitej liczby grup (patrz również rys. 5.14c). Niniejsza analiza dla zbioru *MLL* potwierdza wyżej sformułowany wniosek, że proponowana sieć neuronowa dała najlepsze rezultaty pod względem liczby grup zawierających małą liczbę genów.




Rys. 5.14. Histogramy liczby genów w poszczególnych grupach genów w zbiorze danych *MLL*, otrzymanych z wykorzystaniem metod MSSRCC (a), DSOM (b) oraz USOM (c)

W dalszej części analizy porównawczej, rozważane metody grupowania genów zostały porównane pod względem zdolności do generowania grup zawierających jak największą liczbę genów funkcyjnych. Dla każdego przypadku metody grupowania genów, przeprowadzono analizę funkcyjną wszystkich grup genów ze zbiorów danych *Leukemia* i *MLL*, wykorzystując w tym celu program DAVID Functional Annotation Tool, w sposób analogiczny jak to zostało opisane w podrozdziale 5.3. Następnie, do porównania, z każdego zbioru wybrano po trzy reprezentatywne grupy o największym średnim udziale genów funkcyjnych. Do obliczenia średniego udziału genów funkcyjnych w da-

nej grupie za każdym razem wybierano udziały pięciu funkcji o najmniejszym poziomie istotności *p-value*. Tabela 5.8 i tabela 5.9 przedstawiają szczegółowe zestawienia tych grup (odpowiednio, dla zbiorów *Leukemia* i *MLL*). W przypadku grup genów zarówno ze zbioru *Leukemia*, jak i *MLL*, największy średni udział genów funkcyjnych można zaobserwować w grupach otrzymanych z wykorzystaniem proponowanej metody USOM, odpowiednio, od 37.8% (grupa LE–44) do 80% (grupa LE–7) dla zbioru *Leukemia* i od 44.4% (grupa ML–96) do 51.4% (grupa ML–8) dla zbioru *MLL*. Pozostałe metody generowały grupy genów o zdecydowanie mniejszym udziale genów funkcyjnych.

Tabela 5.8. Udział genów funkcyjnych w wybranych grupach genów ze zbioru *Leukemia,* otrzymanych metodami MSSRCC, DSOM i USOM

| L.p. | Metoda grupowania danych*) | Nazwa grupy genów | Liczba genów w grupie | Średnia liczba genów funkcyjnych | Średni udział genów funkcyjnych | Średni poziom istotności *p-value* |
|---|---|---|---|---|---|---|
| 1 | MSSRCC | LE-38 | 38 | 15 | 39.5% | $1.0 \cdot 10^{-3}$ |
| | | LE-61 | 25 | 4.2 | 16.8% | $4.4 \cdot 10^{-4}$ |
| | | LE-82 | 22 | 6 | 27.3% | $3.1 \cdot 10^{-3}$ |
| 2 | DSOM | LE-18 | 59 | 9.2 | 15.6% | $1.1 \cdot 10^{-4}$ |
| | | LE-29 | 71 | 35 | 49.3% | $2.0 \cdot 10^{-9}$ |
| | | LE-32 | 89 | 15.6 | 17.5% | $3.4 \cdot 10^{-4}$ |
| 3 | **USOM** | LE-7 | 5 | 4 | 80% | $2.2 \cdot 10^{-4}$ |
| | | LE-28 | 10 | 4.2 | 42% | $5.4 \cdot 10^{-4}$ |
| | | LE-44 | 9 | 3.4 | 37.8% | $6.1 \cdot 10^{-4}$ |

*) patrz tabela 5.6.

Tabela 5.9. Udział genów funkcyjnych w wybranych grupach genów ze zbioru *MLL,* otrzymanych metodami MSSRCC, DSOM i USOM

| L.p. | Metoda grupowania danych*) | Nazwa grupy genów | Liczba genów w grupie | Średnia liczba genów funkcyjnych | Średni udział genów funkcyjnych | Średni poziom istotności *p-value* |
|---|---|---|---|---|---|---|
| 1 | MSSRCC | ML-47 | 18 | 4.6 | 25.5% | $8.2 \cdot 10^{-3}$ |
| | | ML-89 | 31 | 11.6 | 37.4% | $8.4 \cdot 10^{-5}$ |
| | | ML-91 | 14 | 6 | 42.8% | $1.9 \cdot 10^{-2}$ |
| 2 | DSOM | ML-4 | 39 | 7 | 17.9% | $5.4 \cdot 10^{-6}$ |
| | | ML-12 | 39 | 6.2 | 15.9% | $3.5 \cdot 10^{-4}$ |
| | | ML-21 | 44 | 15.8 | 35.9% | $7.3 \cdot 10^{-5}$ |
| 3 | **USOM** | ML-8 | 14 | 7.2 | 51.4% | $3.2 \cdot 10^{-3}$ |
| | | ML-58 | 12 | 5.6 | 46.6% | $1.4 \cdot 10^{-5}$ |
| | | ML-96 | 9 | 4 | 44.4% | $2.5 \cdot 10^{-3}$ |

*) patrz tabela 5.6.

## 5.5. Podsumowanie

W niniejszym rozdziale przedstawiono rezultaty zastosowania uogólnionych samoorganizujących się sieci neuronowych o ewoluujących, drzewopodobnych strukturach topologicznych do grupowania danych reprezentujących różne typy nowotworowych chorób układu krwionośnego (białaczek). Rozważono dwa zbiory danych *Leukemia* i *Mixed Lineage Leukemia* (*MLL*), dostępne na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/datasets.html). Dla każdego z nich przeprowadzono po dwa eksperymenty grupowania danych, tzn. grupowanie danych bazujące na próbkach oraz bazujące na genach.

Kombinacja otrzymanych wyników posłużyła do utworzenia dwóch macierzy ekspresji genów (po jednej dla każdego ze zbiorów danych) z oznaczonym podziałem na poszczególne grupy próbek danych i grupy genów (wybrane fragmenty tych macierzy zostały przedstawione w pracy). Następnie, dokonano oceny efektywności proponowanego narzędzia, odrębnie względem grupowania próbek danych (bezpośrednio wykorzystując w tym celu informację o rzeczywistej przynależności próbek danych do odpowiednich klas chorób nowotworowych) oraz pod względem grupowania genów (poprzez analizę genów funkcyjnych w wybranych grupach genów z wykorzystaniem programu DAVID Functional Annotation Tool - http://david.abcc.ncifcrf.gov). Wysoka liczba poprawnych decyzji o przynależności poszczególnych próbek danych do klas wyraźnie wskazuje na bardzo wysoką efektywność proponowanej sieci neuronowej (średnio ponad 92% poprawnych decyzji) w grupowaniu próbek danych. Z kolei, wizualna ocena macierzy ekspresji genów oraz analiza genów funkcyjnych w poszczególnych grupach genów również jednoznacznie wykazują wysoką efektywność sieci, tym razem w grupowaniu genów.

Na zakończenie rozdziału, przedstawiono rezultaty analizy porównawczej proponowanego narzędzia z innymi, do pewnego stopnia alternatywnymi metodami grupowania danych. Rozważono trzy kryteria porównawcze: liczbę poprawnych decyzji o przynależności próbek danych do klas, liczbę grup genów o względnie małej ilości genów (niosą one potencjalnie istotne informacje z punktu widzenia diagnostyki chorób nowotworowych) oraz procentowy udział genów funkcyjnych w grupach. Analiza ponownie potwierdziła wysoką efektywność proponowanego narzędzia i jego znaczącą przewagę nad innymi podejściami w rozważanym problemie.

# 6. Grupowanie danych opisujących ekspresję genów z wykorzystaniem proponowanych sieci neuronowych w zagadnieniach diagnozowania nowotworu jelita grubego

Rozdział ten jest drugim z kolei, który przedstawia zastosowanie – proponowanych w niniejszej pracy – uogólnionych samoorganizujących się sieci neuronowych o ewoluujących, drzewopodobnych strukturach topologicznych do grupowania danych opisujących ekspresję genów. Tym razem, problem dotyczy grupowania danych w zbiorze reprezentującym przypadki chorób nowotworowych jelita grubego. W podrozdziale 6.1 przedstawiono rozważany problem. Następnie, w podrozdziałach 6.2 i 6.3 zaprezentowano, odpowiednio, procesy grupowania danych bazujące na próbkach i genach oraz dokonano analizy otrzymanych wyników. Na zakończenie rozdziału, w podrozdziale 6.4, porównano proponowane narzędzie z alternatywnymi (do pewnego stopnia) metodami.

## 6.1. Przedstawienie problemu

W poprzednim eksperymencie rozważano zbiory danych będące rezultatem badań patomorfologicznych chorób układu krwionośnego, tzn. badań nad klasyfikacją typów tych chorób. Tym razem, rozważany zbiór danych o nazwie *Colon* [2] pochodzi z badań diagnostycznych chorób nowotworowych jelita grubego. Zbiór dostępny jest w archiwum Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/datasets.html).

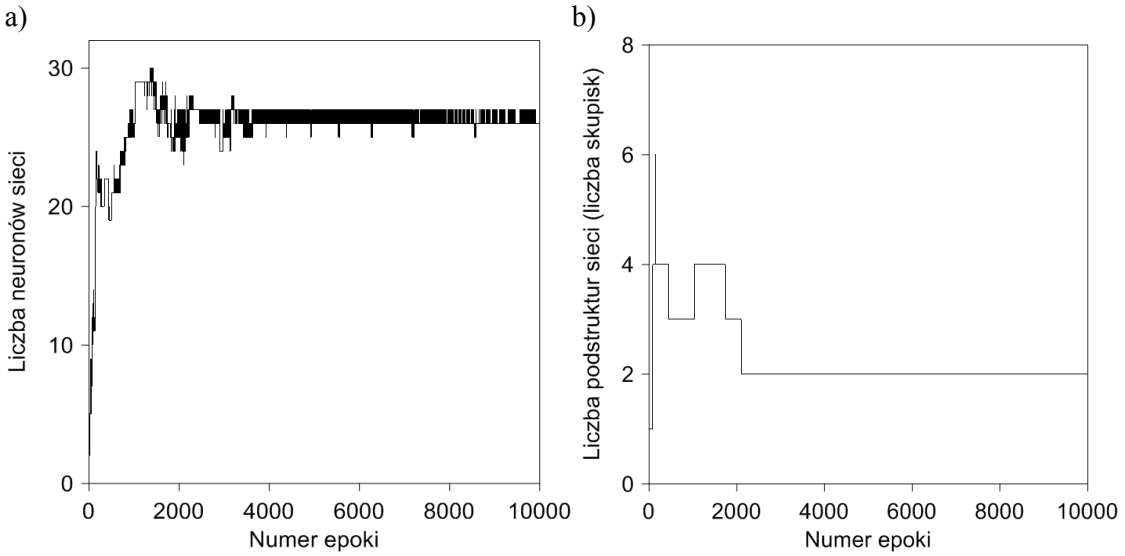
Zbiór danych *Colon* zawiera 62 próbki materiału genetycznego pacjentów, pobrane za pomocą biopsji z jelita grubego, w tym 40 próbek pacjentów ze zdiagnozowaną chorobą nowotworową (64.5% całkowitej liczby próbek). W pierwotnej wersji zbioru *Colon*, każda próbka zawiera poziomy ekspresji 2000 genów. Natomiast na potrzeby rozważanego eksperymentu wykorzystano zredukowany wariant pierwotnego zbioru *Colon*, w którym próbki zawierają po 1096 genów. Zbiór ten powstał – podobnie jak w przypadkach zbiorów danych *Leukemia* i *MLL* – poprzez usunięcie genów nadmiarowych, tzn. zaszumionych i wielokrotnie skopiowanych (szczegółowa procedura redukcji wymiarowości zbioru *Colon* została omówiona w [23, 29 i 61]).

Eksperymenty numeryczne oraz analiza wyników zostaną przeprowadzone podobnie jak w poprzednim rozdziale. W pierwszej kolejności zostanie przeprowadzony proces grupowania danych bazujący na próbkach, a w drugiej kolejności proces grupowania danych bazujący na genach. Ocena rezultatów obu procesów grupowania danych jest próbą odpowiedzi, czy proponowana sieć neuronowa może być wykorzystana, jako potencjalne narzędzie do, odpowiednio, wspomagania diagnostyki chorób nowotworowych (ocena rezultatów grupowania bazującego na próbkach) i patogenezy chorób nowotworowych, tzn. badania przyczyn o podłożu genetycznym powstawania chorób nowotworowych (ocena rezultatów grupowania bazującego na genach). Głębsza analiza otrzymanych wyników może być przeprowadzona z wykorzystaniem metod medycznych, co – z oczywistych względów – wykracza poza zakres niniejszej rozprawy.

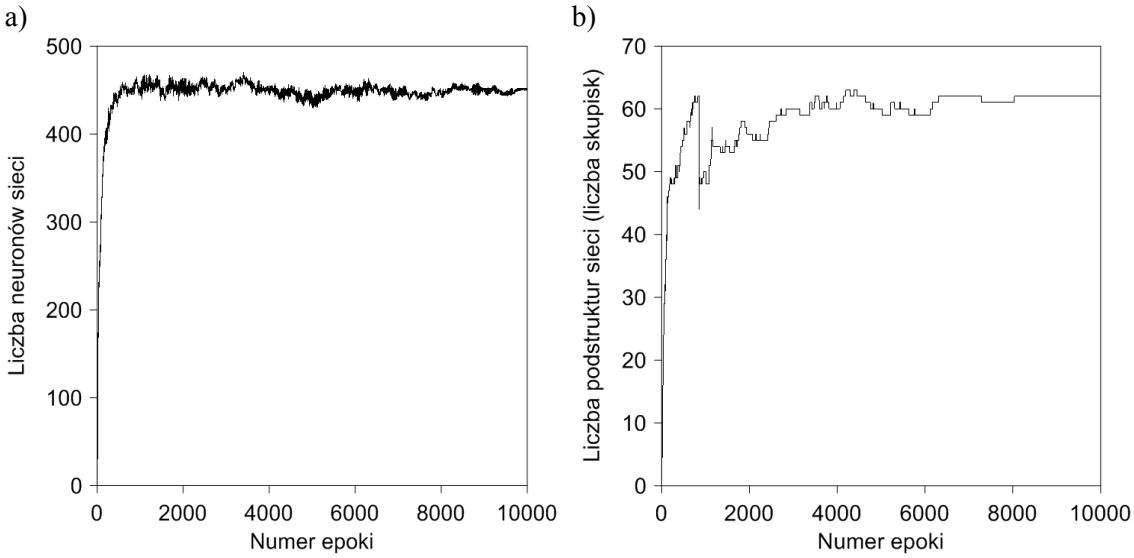
## 6.2. Procesy uczenia uogólnionych samoorganizujących się sieci neuronowych o drzewopodobnych strukturach

Procesy uczenia proponowanych sieci neuronowych, wykorzystanych w eksperymentach przedstawionych w tym rozdziale, przeprowadzono w analogiczny sposób jak w przypadku eksperymentów z rozdziału 5. Na rys. 6.1 przedstawiono przebiegi zmian liczby neuronów sieci (rys. 6.1a) oraz liczby podstruktur sieci (rys. 6.1b) w procesie grupowania danych bazującego na próbkach. Po zakończeniu procesu uczenia, liczba podstruktur sieci neuronowej ustabilizowała się (już po upływie ok. 2000 epok uczenia) się na poziomie równym liczbie grup próbek danych w rozważanym zbiorze (dwie podstruktury, z których każda odpowiada jednej grupie próbek danych). Liczba neuronów sieci neuronowej oscylowała przez większą część procesu uczenia (od ok. 4000 epoki) pomiędzy wartościami 25 i 27, aby ostatecznie osiągnąć wartość $m = 26$.

Na rys. 6.2 przedstawiono analogiczne przebiegi zmian liczby neuronów sieci (rys. 6.2a) oraz liczby podstruktur sieci (rys. 6.2b) dla przypadku grupowania danych bazującego na genach. Tym razem, po zakończeniu procesu uczenia sieć neuronowa osiągnęła rozmiar $m = 451$ neuronów i składała się z 62 podstruktur topologicznych. Otrzymane podstruktury topologiczne obu sieci neuronowych stanowią wielopunktowe prototypy, odpowiednio, 2 grup próbek materiału genetycznego pacjentów (zdrowych i chorych) oraz 62 grup genów wykrytych w rozważanym zbiorze danych.




Rys. 6.1. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *Colon* bazujące na próbkach)

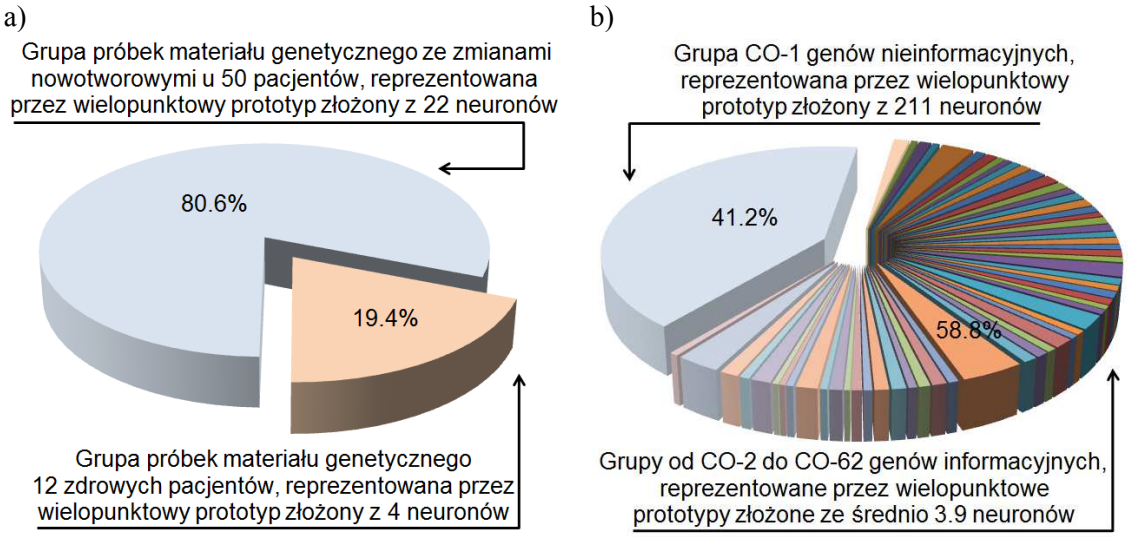



Rys. 6.2. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *Colon* bazujące na genach)

Po dokonaniu tzw. kalibracji sieci, czyli nadaniu otrzymanym wielopunktowym prototypom skupisk danych unikatowych etykiet, przyporządkowano poszczególne próbki danych oraz poszczególne geny do odpowiednich grup. I tak, w przypadku sieci neuronowej, której zadaniem było grupowanie próbek danych, pierwszemu prototypowi składającemu się z 22 neuronów (o numerach porządkowych od 1 do 22) nadano etykie-

tę „C”, oznaczającą tkankę ze zmianami nowotworowymi. Natomiast drugiemu prototypowi składającemu się z 4 neuronów (o numerach porządkowych od 23 do 26) przypisano etykietę „N”, oznaczającą tkankę zdrową. Oba prototypy reprezentują grupy złożone, odpowiednio, z 50 oraz 12 próbek materiału genetycznego pacjentów.

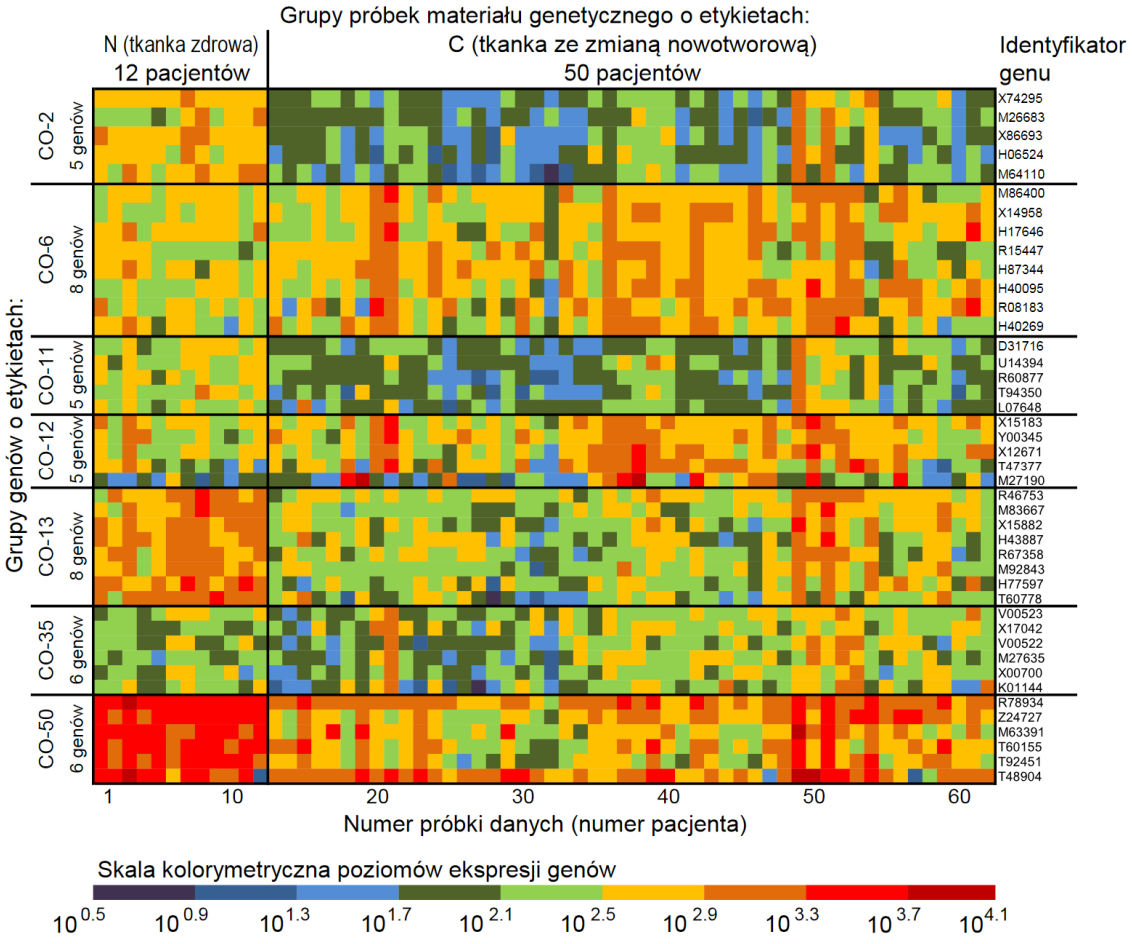
Z kolei, w przypadku sieci neuronowej odpowiedzialnej za grupowanie danych bazujące na genach, otrzymanym wielopunktowym prototypom grup genów przypisano etykiety według wzorca CO-*i*, gdzie *i* jest numerem porządkowym grupy (numery od 1 do 62). Największą grupę złożoną z 452 genów (41.2% wszystkich genów zbioru *Colon*) reprezentuje prototyp o etykiecie CO-1 składający się z 211 neuronów (o numerach porządkowych od 164 do 274). Pozostałe 61 grup genów reprezentowane są przez prototypy posiadające od 2 do 18 neuronów (średnio 3.9 neuronów na prototyp) i zawierają od 4 do 44 genów (średnio 10.6 genów na grupę). Geny informacyjne stanowią 58.8% całkowitej liczby genów rozważanego zbioru danych oraz 32.2% liczby genów pierwotnego zbioru *Colon*. Proporcje podziału próbek danych i genów na grupy są zilustrowane na rys. 6.3. Na podstawie analogicznych kryteriów jak w przypadku prototypów LE-1, ML-1 i ML-2 z rozdziału 5, przyjęto, że prototyp CO-1 reprezentuje grupę genów nieinformacyjnych. Nie będzie ona uwzględniana w późniejszej analizie w dalszej części tego rozdziału.



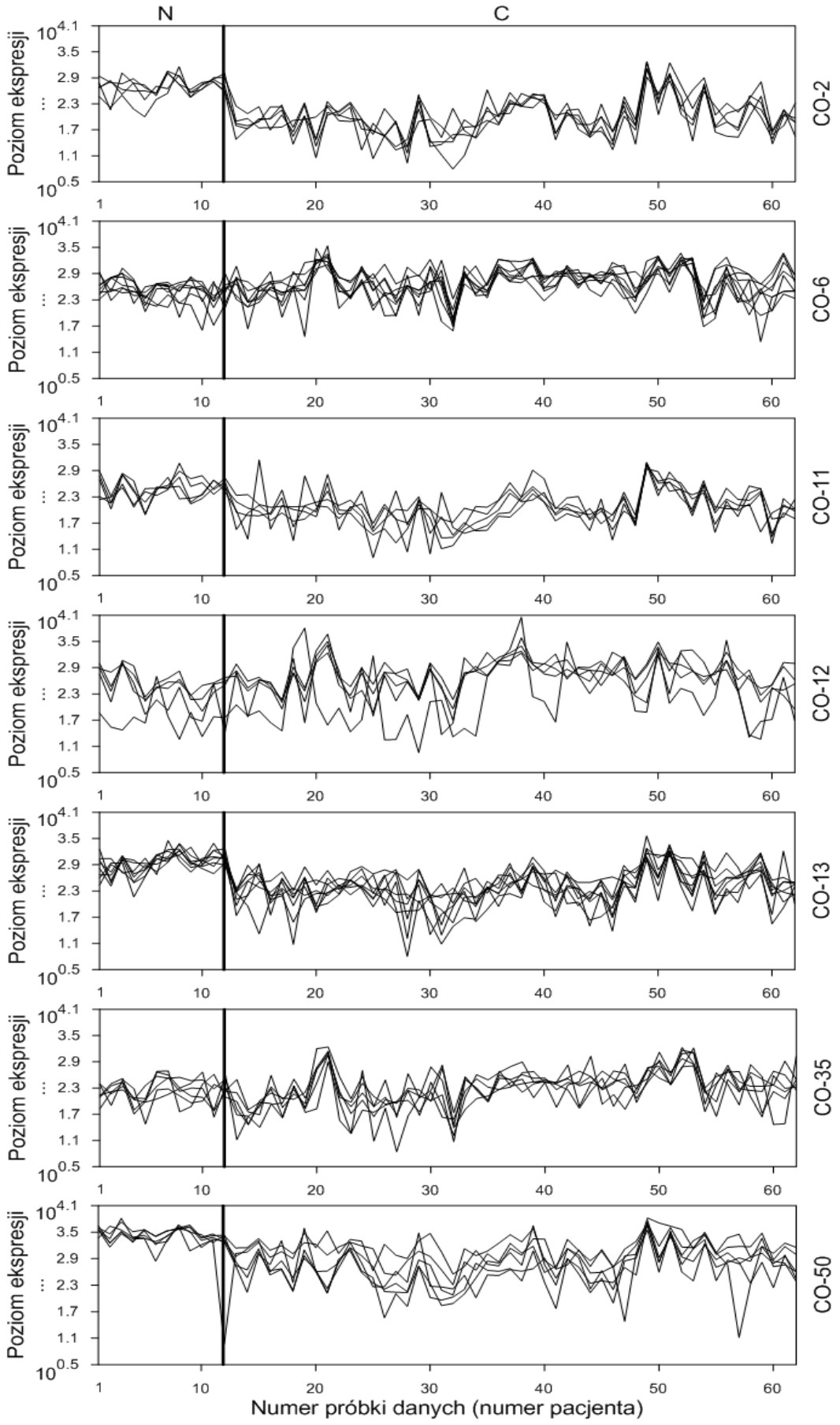

Rys. 6.3. Proporcje podziału danych zbioru *Colon* na grupy otrzymane w rezultacie grupowania bazującego na próbkach (a) oraz bazującego na genach (b)

Ze względu na dużą wymiarowość zbioru danych *Colon*, nie jest możliwe zaprezentowanie wszystkich otrzymanych grup genów w postaci graficznej (podobnie jak

miało to miejsce w przypadku zbiorów *Leukemia* i *MLL* z poprzedniego rozdziału). Na rys. 6.4 przedstawiono 7 wybranych, reprezentatywnych grup genów o następujących etykietach: CO-2, CO-6, CO-11, CO-12, CO-13, CO-35 oraz CO-50. Kolorowe punkty określają poziomy ekspresji poszczególnych genów, przy czym w celu poprawienia wizualnego odbioru rysunku zwiększono jego kontrast przeliczając pierwotne wartości poziomów ekspresji genów według skali logarytmicznej o podstawie 10, analogicznie jak to zostało zrobione w pracy [23]. Czarną pionową linią oznaczono granicę pomiędzy grupą próbek materiału genetycznego tkanek zdrowych, a grupą próbek materiału genetycznego tkanek ze zmianami nowotworowymi. Analogicznie, liniami poziomymi oznaczono granice pomiędzy grupami genów. Wizualną ocenę podobieństwa poziomów ekspresji genów należących do tych samych grup umożliwia rys. 6.5 (poziomy ekspresji genów z rys. 6.4 widoczne są w formie nakładających się wzajemnie krzywych).




Rys. 6.4. Graficzna ilustracja poziomów ekspresji genów ze zbioru *Colon* oraz granicami podziału pomiędzy grupami próbek danych i grupami genów




Rys. 6.5. Graficzna ilustracja poziomów ekspresji genów z rys. 6.4

## 6.3. Ocena efektywności grupowania danych

Podobnie jak w poprzednim rozdziale, zostanie teraz przeprowadzona ocena rezultatów grupowania danych w zbiorze *Colon*, odrębnie dla grupowania bazującego na próbkach i bazującego na genach.

### *Ocena efektywności grupowania danych bazującego na próbkach*

Do oceny rezultatów grupowania bazującego na próbkach wykorzystano informację o rzeczywistej przynależności poszczególnych próbek danych do odpowiednich klas, zawartej w zbiorze *Colon*. Należy podkreślić, że informacja ta nie była wykorzystywana podczas uczenia sieci neuronowej (uczenie odbywa się trybie nienadzorowanym). Zestawienie otrzymanych rezultatów przedstawia tabela 6.1.

Tabela 6.1. Rezultaty grupowania danych bazującego na próbkach (zbiór danych *Colon*)

| Etykieta klasy*) | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie*): | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|
| | | C | N | | | |
| C | 40 | 39 | 1 | 39 | 1 | 97.5% |
| N | 22 | 11 | 11 | 11 | 11 | 50% |
| **RAZEM** | **62** | **50** | **12** | **50** | **12** | **80.6%** |

*) C – tkanka ze zmianami nowotworowymi, N – tkanka zdrowa.

Liczba poprawnych decyzji o przyporządkowaniu próbek danych do odpowiednich klas wynosi 80.6%. Wynik ten należy uznać za bardzo dobry, zważywszy na fakt, że został otrzymany w ramach nienadzorowanego trybu uczenia sieci neuronowej. Szczególnie bardzo dobre rezultaty osiągnięto w rozpoznawaniu przypadków próbek materiału genetycznego ze zmianami nowotworowymi (97.5% poprawnych decyzji; tylko jedna próbka ze zmianami nowotworowymi została błędnie zakwalifikowana jako zdrowa), co świadczy o bardzo dobrej tzw. czułości sieci neuronowej (ang. *sensitivity*), czyli zdolności do poprawnego wykrywania stanów patologicznych (w ogólnym przypadku, czułość narzędzia jest jednym z najważniejszych parametrów decydujących o jego przydatności do wspomagania diagnostyki chorób nowotworowych). Natomiast słabszy wynik (50.0% poprawnych decyzji) uzyskano dla drugiej, mniej licznej grupy próbek tkanek nie zaatakowanych przez chorobę nowotworową. Świadczy on o słabszej zdolności sieci neuronowej do poprawnej kwalifikacji stanów normalnych (tzw. trafność de-

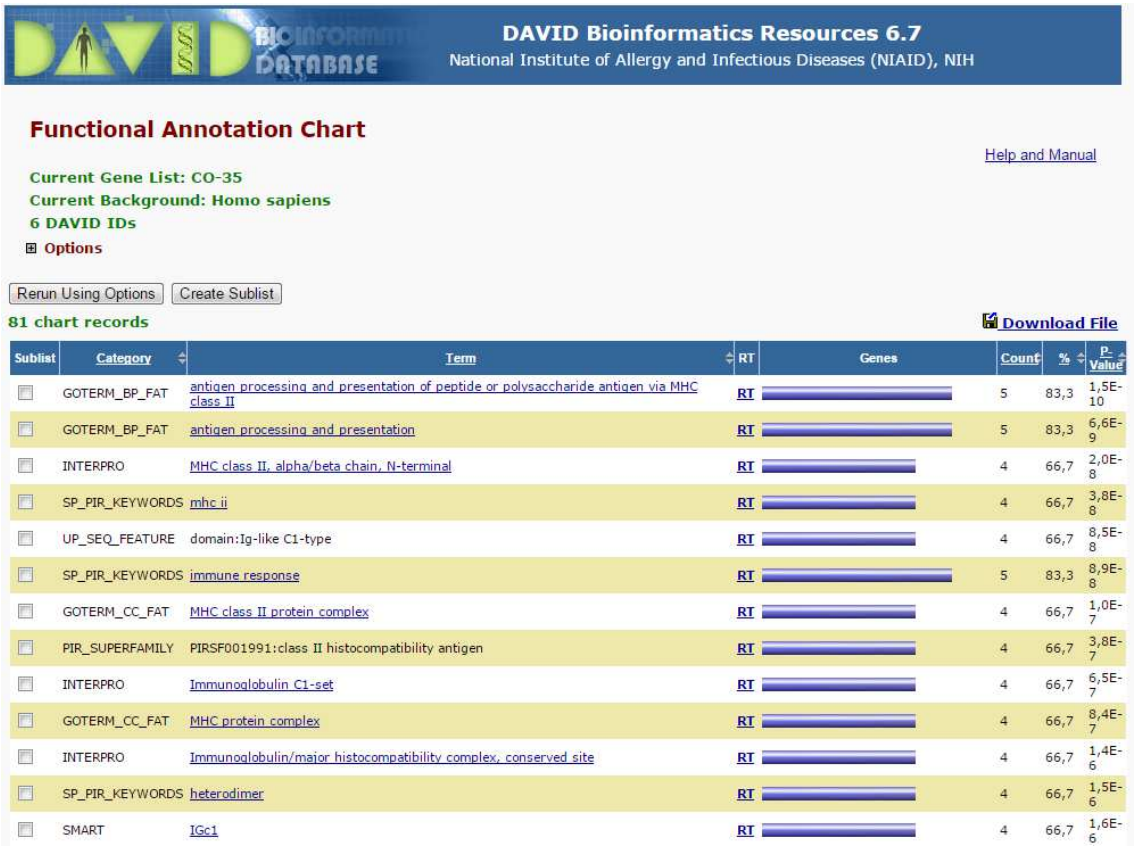
cyzji lub specyficzność – ang. *specificity*) i tendencji do generowania fałszywych alarmów. Należy jednak zauważyć, że trafność decyzji jest parametrem o mniejszym znaczeniu dla diagnostyki chorób nowotworowych. Nawet jeżeli sieć neuronowa wstępnie zakwalifikuje pewne zdrowe próbki do kategorii tkanek ze zmianami nowotworowymi, to i tak ostateczna decyzja w tym zakresie zostanie podjęta przez lekarza specjalistę, który powiadomiony przez system dokona dalszych badań.

***Ocena efektywności grupowania danych bazującego na genach***

Podobnie jak w przypadku zbiorów danych *Leukemia* i *MLL*, nie jest możliwa bezpośrednia ocena rezultatów grupowania bazującego na genach w zbiorze danych *Colon*, gdyż zbiór ten nie zawiera ani informacji o przynależności genów do grup, ani o liczbie tych grup. Dokonano natomiast subiektywnej oceny rezultatów grupowania poprzez analizę statystyczną funkcji genów informacyjnych występujących w grupach od CO-2 do CO-62 (grupę CO-1 genów nieinformacyjnych pominięto), za pomocą bazy genów wzorcowych DAVID oraz współpracującego z nią oprogramowania do analizy funkcyjnej genów DAVID Functional Annotation Tool (analogicznie jak w podrozdziale 5.3). Z powodu dużej liczby grup genów, prezentacja wyników powyższych analiz dla każdej grupy z osobna nie byłaby czytelna i z tego powodu nie będzie przedstawiana.

Natomiast, dla przykładu, na rys. 6.6 przedstawiono rezultaty analizy statystycznej grupy CO-35 (6 genów). Przyjęto, że dla statystycznie najbardziej istotnych funkcji genów poziom istotności *p-value* jest nie większy niż $10^{-7}$ (ostrzejsze kryterium od powszechnie przyjmowanego poziomu *p-value* < 0.05 [23]) Tabela 6.2 zawiera zestawienie tych funkcji wraz z wyszczególnieniem genów funkcyjnych. Udział genów funkcyjnych w grupie CO-35 dla trzech funkcji *antigen processing and presentation of peptide or polysaccharide antigen via MHC class II*, *antigen processing and presentation* oraz *immune response* wynosi 83.3%, a dla pozostałych trzech funkcji jest również wysoki (66.7%). Świadczy to o dużej wartości grupy CO-35 dla dalszych badań o charakterze medycznym. Zbliżone rezultaty uzyskano dla zdecydowanej większości pozostałych grup genów zbioru *Colon.*

Na podstawie bardzo dobrej zdolności sieci neuronowej do generowania informacyjnie istotnych grup genów oraz do skutecznego rozpoznawania stanów patologicznych choroby można wnioskować, że sieć będzie efektywnym narzędziem do wspomagania diagnostyki i badań nad patogenezą nowotworów jelita grubego.




| Sublist | Category | Term | RT | Genes | Count | % | P-Value |
|---|---|---|---|---|---|---|---|
| | GOTERM_BP_FAT | antigen processing and presentation of peptide or polysaccharide antigen via MHC class II | RT | | 5 | 83,3 | 1,5E-10 |
| | GOTERM_BP_FAT | antigen processing and presentation | RT | | 5 | 83,3 | 6,6E-9 |
| | INTERPRO | MHC class II, alpha/beta chain, N-terminal | RT | | 4 | 66,7 | 2,0E-8 |
| | SP_PIR_KEYWORDS | mhc ii | RT | | 4 | 66,7 | 3,8E-8 |
| | UP_SEQ_FEATURE | domain:Ig-like C1-type | RT | | 4 | 66,7 | 8,5E-8 |
| | SP_PIR_KEYWORDS | immune response | RT | | 5 | 83,3 | 8,9E-8 |
| | GOTERM_CC_FAT | MHC class II protein complex | RT | | 4 | 66,7 | 1,0E-7 |
| | PIR_SUPERFAMILY | PIRSF001991:class II histocompatibility antigen | RT | | 4 | 66,7 | 3,8E-7 |
| | INTERPRO | Immunoglobulin C1-set | RT | | 4 | 66,7 | 6,5E-7 |
| | GOTERM_CC_FAT | MHC protein complex | RT | | 4 | 66,7 | 8,4E-7 |
| | INTERPRO | Immunoglobulin/major histocompatibility complex, conserved site | RT | | 4 | 66,7 | 1,4E-6 |
| | SP_PIR_KEYWORDS | heterodimer | RT | | 4 | 66,7 | 1,5E-6 |
| | SMART | IGc1 | RT | | 4 | 66,7 | 1,6E-6 |

Rys. 6.6. Rezultaty testu Fishera dla grupy genów CO-35 wybranej ze zbioru danych *Colon*, generowane przez program DAVID Functional Annotation Tool

Tabela 6.2. Statystycznie najbardziej znaczące funkcje genów grupy CO-35, wybrane spośród rezultatów testu Fishera przedstawionych na rys. 6.6

| Lp. | Nazwa funkcji genów | Geny funkcyjne w grupie genów CO-35 | | | Poziom istotności *p-value* |
|---|---|---|---|---|---|
| | | Ilość | Identyfikatory genów | Udział | |
| 1 | *antigen processing and presentation of peptide or polysaccharide antigen via MHC class II* | 5 | K01144, V00523, V00522, M27635, X00700 | 83.3% | $1.5 \cdot 10^{-10}$ |
| 2 | *antigen processing and presentation* | 5 | K01144, V00523, V00522, M27635, X00700 | 83.3% | $6.6 \cdot 10^{-9}$ |
| 3 | *MHC class II, alpha beta chain, N-terminal* | 4 | V00523, V00522, M27635, X00700 | 66.7% | $2.0 \cdot 10^{-8}$ |
| 4 | *mhc ii* | 4 | V00523, V00522, M27635, X00700 | 66.7% | $3.8 \cdot 10^{-8}$ |
| 5 | *domain:Ig-like C1-type* | 4 | V00523, V00522, M27635, X00700 | 66.7% | $8.5 \cdot 10^{-8}$ |
| 6 | *immune response* | 5 | K01144, V00523, V00522, M27635, V00522 | 83.3% | $8.9 \cdot 10^{-8}$ |

## 6.4. Analiza porównawcza

Analiza porównawcza proponowanego narzędzia z alternatywnymi technikami grupowania danych została przeprowadzona analogicznie jak w podrozdziale 5.4, tzn. oddzielnie dla grupowania bazującego na próbkach i oddzielnie dla grupowania bazującego na genach.

### *Analiza porównawcza rezultatów grupowania danych bazującego na próbkach*

Rezultaty analizy porównawczej funkcjonowania uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej (USOM) i alternatywnych metod, w problemie grupowania danych bazującego na próbkach dla zbioru *Colon*, przedstawia tabela 6.3. Rozważono pięć metod wykorzystanych wcześniej w analizie porównawczej w podrozdziale 5.4, tzn.: algorytmy: *k-means* [71, 86], EM (ang. *Expectation Maximization*) [27, 72], FFTA (ang. *Farthest First Traversal Algorithm*) [87] i MSSRCC (ang. *Minimum Sum-Squared Residue Coclustering*) [23] oraz dynamiczną, samoorganizującą się sieć neuronową z jednowymiarowym sąsiedztwem topologicznym DSOM [46]. Rezultaty dla metody MSSRCC pochodzą z pracy [23], natomiast pozostałe wyniki otrzymano na podstawie własnych obliczeń, analogicznie jak w przypadku eksperymentów z rozdziału 5.

Tabela 6.3. Rezultaty grupowania bazującego na próbkach dla zbioru *Colon* z wykorzystaniem różnych metod grupowania danych

| Lp. | Metoda grupowania danych*) | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|
| 1 | *k-means* | 34 | 28 | 54.8% |
| 2 | EM | 32 | 30 | 51.6% |
| 3 | FFTA | 34 | 28 | 54.% |
| 4 | MSSRCC | 53 | 9 | 85.7% |
| 5 | DSOM | 42 | 20 | 67.7% |
| 6 | **USOM** | **50** | **12** | **80.6%** |

*) patrz tabela 5.6.

Metoda MSSRCC i proponowana sieć USOM dały znacząco lepsze rezultaty (odpowiednio, 85.7% i 80.6% poprawnych decyzji o przynależności próbek danych do odpowiednich grup) od pozostałych metod grupowania danych (od 51.6% do 67.6% po-

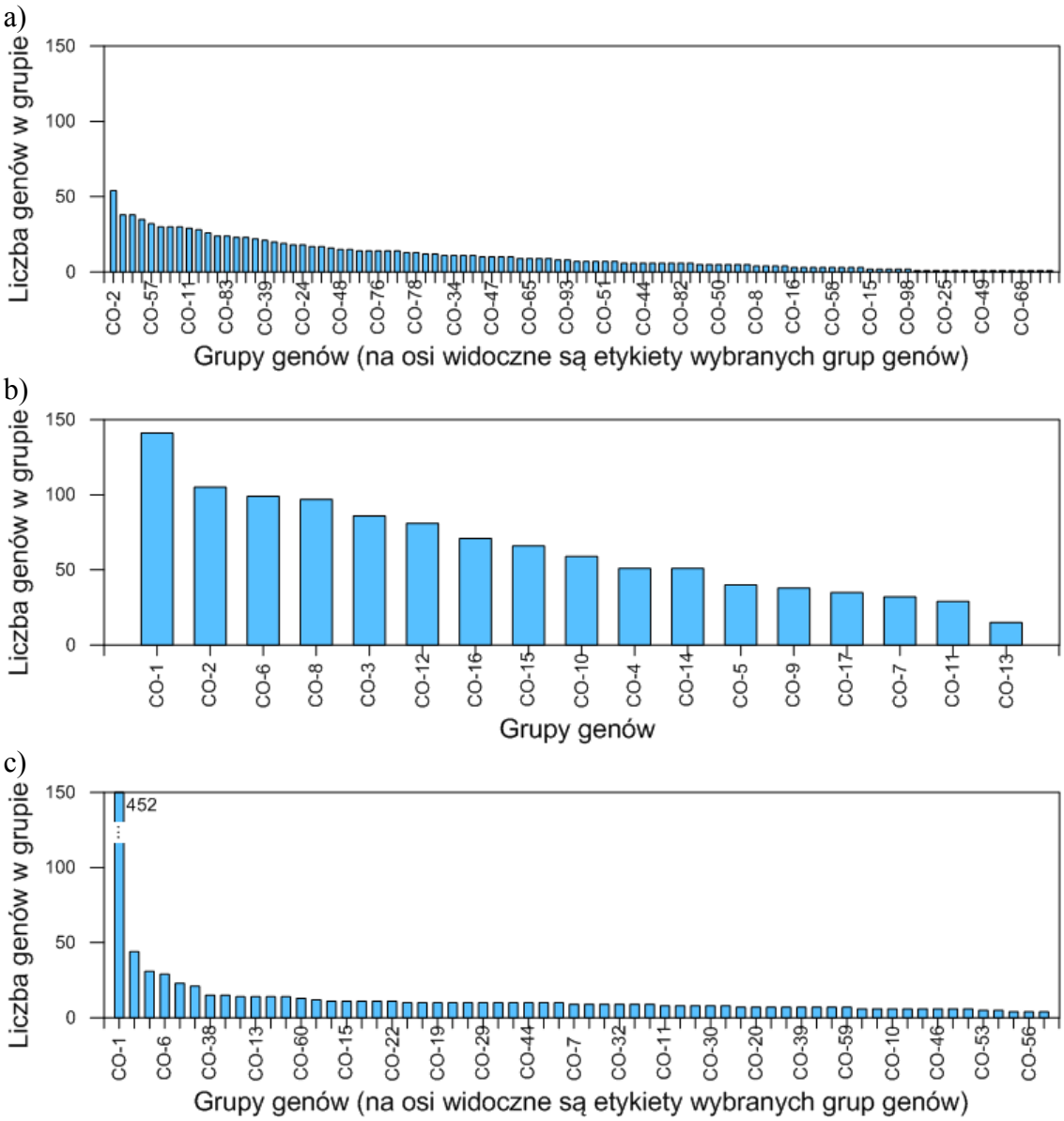
prawnych decyzji). Należy podkreślić, że spośród tych dwóch najlepszych podejść, metoda MSSRCC wymaga określenia z góry liczby grup w zbiorze danych, co znacząco faworyzuje ją w porównaniu z USOM, która wyznacza liczbę grup w sposób automatyczny.

***Analiza porównawcza rezultatów grupowania danych bazującego na genach***

W celu porównania funkcjonowania proponowanej metody w problemie grupowania danych bazującym na genach dla zbioru *Colon,* wybrano dwie metody: MSSRCC, która daje najlepszy rezultat grupowania bazującego na próbkach oraz DSOM, która – podobnie jak proponowana USOM – nie wymaga podania z góry liczby grup istniejących w zbiorze danych. Pozostałe metody pominięto, gdyż wymagają one podania z góry liczby grup genów (liczba tych grup w zbiorze *Colon* jest nieznana; w przypadku metody MSSRCC, w pracy [23] przyjęto arbitralnie liczbę grup równą 100). Podobnie jak dla zbiorów danych *Leukemia* i *MLL* dokonano subiektywnego porównania wyników grupowania bazującego na genach pod względem dwóch kryteriów: a) liczby grup o względnie małej liczbie genów (najcenniejszych w diagnostyce chorób nowotworowych [69, 82]) oraz b) procentowego udziału genów funkcyjnych w grupach. Rys. 6.7 przedstawia histogramy liczb genów w poszczególnych grupach dla trzech rozważanych metod grupowania danych.

W przypadku metody MSSRCC (rys. 6.7a), grupy liczyły od 1 do 54 genów (średnio 11 genów na grupę), przy czym aż 15 z nich to grupy z pojedynczymi genami (co bardzo niekorzystnie świadczy o tej metodzie, gdyż takie grupy nie niosą żadnych informacjo o relacjach zachodzących pomiędzy genami i ich funkcjami). Liczba grup o względnie niewielkiej liczbie genów (przyjęto od 2 do 20 genów włącznie) stanowi 68% (68 grup) całkowitej liczby grup. W przypadku metod DSOM i USOM liczba grup (wykryta w sposób automatyczny) wynosi, odpowiednio, 17 i 61. Sieć DSOM utworzyła względnie duże grupy zawierające od 15 do 141 genów (średnio 64.5 genów na grupę), przy czym tylko jedna grupa zawiera względnie małą liczbę genów (mniej niż 20 genów), co stanowi 5.9 % wszystkich grup (patrz rys. 6.7b). Z kolei, proponowana sieć USOM utworzyła jedną dużą grupę genów nieinformacyjnych (452 geny) oraz 61 mniejszych grup, które zawierały od 4 do 44 genów (średnio 10.6 genów na grupę). Liczba względnie małych grup (zawierających co najwyżej 20 genów) wynosi 56

(91.8% wszystkich grup; patrz rys. 6.7c) i jest to najlepszy rezultat spośród wszystkich rozważanych metod.




Rys. 6.7. Histogramy liczby genów w poszczególnych grupach genów w zbiorze danych *Colon*, otrzymanych z wykorzystaniem metod MSSRCC (a), DSOM (b) oraz USOM (c)

Analizę liczby genów funkcyjnych w poszczególnych grupach zbioru *Colon*, utworzonych przez metody MSSRCC, DSOM i USOM przeprowadzono z wykorzystaniem programu DAVID Functional Annotation Tool (analogicznie jak w rozdziale 5.3). Dla każdej z metod wybrano po trzy reprezentatywne grupy, tzn. grupy o największym średnim udziale genów funkcyjnych. W obliczeniach uwzględniono pierwszych pięć funkcji o najmniejszym poziomie istotności *p-value*. Szczegółowe zestawienia tych

grup zawiera tabela 6.4. Proponowana metoda grupowania USOM utworzyła grupy o największym średnim udziale genów funkcyjnych w grupie (od 57.1% dla grupy CO-41 do 93.3% dla grupy CO-34). Pozostałe metody generowały grupy genów o zdecydowanie niższym udziale procentowym genów funkcyjnych.

Tabela 6.4. Udział liczby genów funkcyjnych w wybranych grupach genów ze zbioru *Colon,* otrzymanych metodami MSSRCC, DSOM i USOM

| L.p. | Metoda grupowania danych[*1)] | Nazwa grupy genów | Liczba genów w grupie[*2)] | Średnia liczba genów funkcyjnych | Średni udział genów funkcyjnych | Średni poziom istotności *p-value* |
|---|---|---|---|---|---|---|
| 1 | MSSRCC | CO-13 | 14 (18) | 7.4 | 52.8% | $1.3 \cdot 10^{-2}$ |
| | | CO-20 | 4 (7) | 2 | 50.0% | $3.9 \cdot 10^{-2}$ |
| | | CO-84 | 17 | 12 | 70.6% | $2.8 \cdot 10^{-6}$ |
| 2 | DSOM | CO-9 | 24 (38) | 6.8 | 28.3% | $1.2 \cdot 10^{-3}$ |
| | | CO-10 | 25 (59) | 9.6 | 38.4% | $1.1 \cdot 10^{-3}$ |
| | | CO-16 | 32 (71) | 5.6 | 17.5% | $1.1 \cdot 10^{-5}$ |
| 3 | **USOM** | CO-34 | 6 (11) | 5.6 | 93.3% | $1.5 \cdot 10^{-8}$ |
| | | CO-35 | 6 | 4.4 | 73.3% | $3.0 \cdot 10^{-8}$ |
| | | CO-41 | 7 (12) | 4 | 57.1% | $2.4 \cdot 10^{-3}$ |

[*1)] patrz tabela 5.6, [*2)] liczba genów rozpoznanych przez aplikację DAVID Functional Annotation Tool oraz liczba wszystkich genów (w nawiasach).

## 6.5. Podsumowanie

W niniejszym rozdziale przedstawiono wyniki zastosowania uogólnionych samoorganizujących się sieci neuronowych o ewoluujących, drzewopodobnych strukturach topologicznych do grupowania próbek danych oraz genów w zbiorze danych *Colon*. Zbiór ten jest dostępny na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/data-sets.html) i reprezentuje przypadki osób zdrowych oraz cierpiących na chorobę nowotworową jelita grubego.

Przeprowadzono dwa eksperymenty grupowania danych bazującego na próbkach oraz bazującego na genach. Celem eksperymentów był test przydatności proponowanego narzędzia do budowy systemów wspomagania diagnostyki oraz patogenezy chorób nowotworowych. Na podstawie kombinacji otrzymanych wyników utworzono macierz poziomów ekspresji genów z zaznaczonymi granicami pomiędzy grupami próbek oraz granicami pomiędzy grupami genów (ze względu na duży rozmiar tej macierzy, został

przedstawiony jedynie wybrany jej fragment). Następnie oceniono efektywność proponowanego narzędzia.

W przypadku grupowania bazującego na próbkach otrzymano bardzo dobre rezultaty (80.6% poprawnych decyzji o przynależności próbek danych do odpowiednich grup) i to przy założeniu, że liczba grup nie jest znana lub określana z góry, ale musi być wyznaczona w sposób automatyczny. Sieć neuronowa wykazała się szczególnie wysoką czułością (97.5% poprawnych decyzji), czyli zdolnością do poprawnej klasyfikacji stanów patologicznych, co ma niezwykle ważne znaczenie w diagnostyce chorób nowotworowych. W przypadku grupowania bazującego na genach, analiza z wykorzystaniem programu DAVID Functional Annotation Tool wykazała bardzo dobrą skuteczność proponowanego narzędzia w generowaniu dużej liczby grup genów o najwyższym udziale genów funkcyjnych, co ma istotne znaczenie dla wspomagania badań nad patogenezą chorób nowotworowych. Analiza porównawcza potwierdziła przewagę proponowanego narzędzia nad innymi, do pewnego stopnia alternatywnymi metodami.

# 7. Zastosowanie proponowanych sieci neuronowych do grupowania danych opisujących ekspresję genów w przypadku nowotworowych chorób układu limfatycznego

W niniejszym rozdziale – ostatnim w części aplikacyjnej pracy – zostaną przedstawione wyniki zastosowania uogólnionych samoorganizujących się sieci neuronowych o ewoluujących, drzewopodobnych strukturach topologicznych do grupowania danych reprezentujących przypadki różnych typów nowotworowych chorób układu limfatycznego. W pierwszej kolejności przedstawiono szczegóły eksperymentu (podrozdział 7.1) i jego wyniki (podrozdział 7.2), natomiast w drugiej kolejności dokonano analizy efektywności proponowanego narzędzia (podrozdział 7.3) oraz analizy porównawczej z innymi, do pewnego stopnia, alternatywnymi metodami (podrozdział 7.4).

## 7.1. Przedstawienie problemu

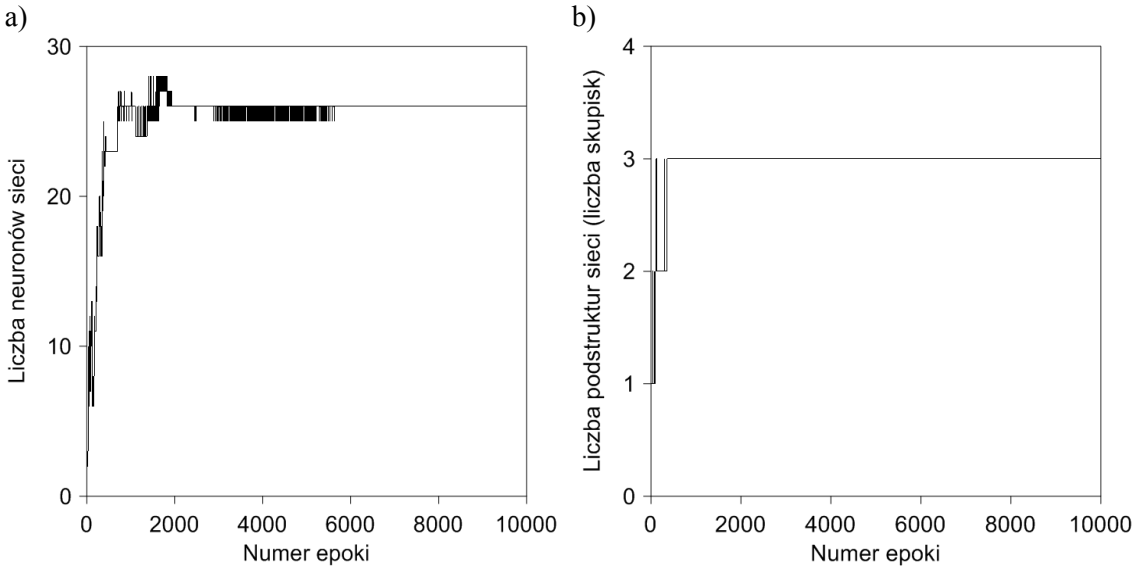
W rozważanym problemie zostanie wykorzystany zbiór danych *Lymphoma*, dostępny na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering, http://csse.szu.edu.cn/staff/zhuzx/datasets.html). Zbiór zawiera poziomy aktywności 4026 genów wyodrębnionych z 62 próbek materiału genetycznego. Każda próbka jest przyporządkowana do jednego z 3 typów chorób nowotworowych układu limfatycznego takich jak: chłoniak rozlany z dużych limfocytów B – DLBCL (ang. *Diffuse Large B-Cell Lymphoma*), przewlekła białaczka limfocytowa – CLL (ang. *Chronic Lymphocytic Leukemia*) oraz chłoniak grudkowy – FL (ang. *Follicular Lymphoma*) (informacje dotyczące nazw typów tych chorób zostały zaczerpnięte z pracy [116]).

Podobnie jak w poprzednich eksperymentach, dwie sieci neuronowe zostaną wykorzystane, odpowiednio, do grupowania bazującego na próbkach oraz bazującego na genach. Zadaniem pierwszej sieci jest podział zbioru danych na właściwą liczbę grup próbek materiału genetycznego, odpowiadającą wymienionym wyżej typom choroby oraz wyznaczenie wielopunktowych prototypów tych grup. Zadaniem drugiej sieci jest podział zbioru danych na grupy genów o podobnych poziomach ekspresji. W odróżnieniu od rozważanych we wcześniejszych rozdziałach zbiorach danych, zbiór *Lymphoma* nie zawiera unikatowych identyfikatorów wszystkich genów, lecz jedynie identyfikatory

2382 genów (59.2% całkowitej liczby genów). Z tego względu, analiza efektywności sieci neuronowej oraz analiza porównawcza w zakresie grupowania bazującego na genach zostały ograniczone do podzbioru genów o znanych identyfikatorach (przy czym, w procesie grupowania genów brały udział wszystkie geny). Celem eksperymentu jest ponowne zbadanie i zarazem potwierdzenie wykazanej w rozdziale 5 i 6 przydatności proponowanego narzędzia do budowy systemów wspomagania badań nad chorobami nowotworowymi, tym razem nad chorobami układu limfatycznego.

## 7.2. Procesy uczenia uogólnionych samoorganizujących się sieci neuronowych o drzewopodobnych strukturach

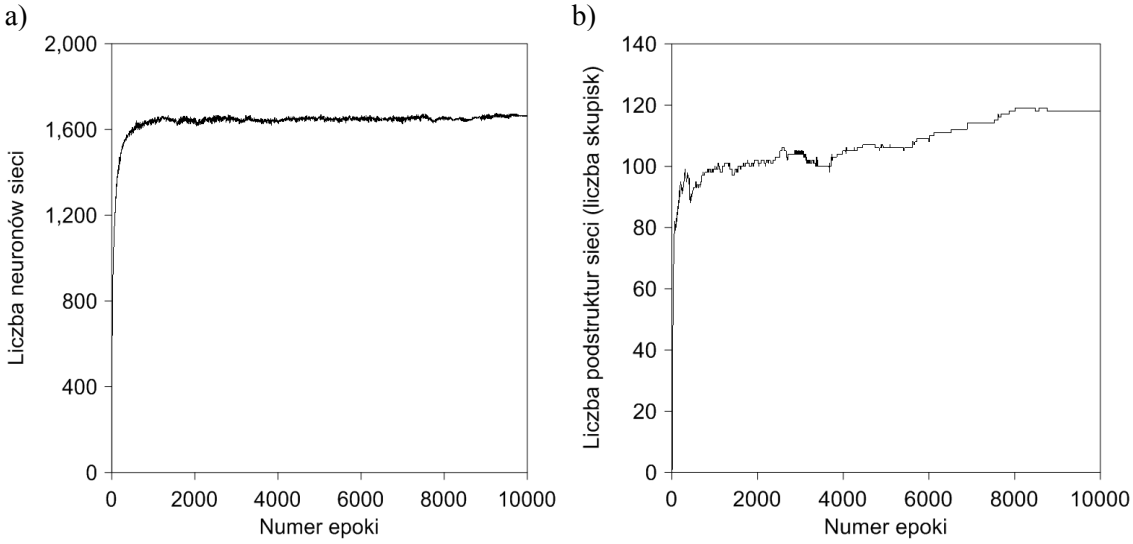
Przebieg zmian najważniejszych parametrów sieci neuronowej, podczas procesu grupowania danych bazującego na próbkach w zbiorze *Lymphoma*, przedstawia rys. 7.1. Po upływie 6000 epok uczenia (rys. 7.1a), liczba neuronów ustabilizowała się na poziomie $m = 26$. Sieć neuronowa bardzo szybko (już po 200 epokach; patrz rys. 7.1b) wykryła prawidłową liczbę 3 grup próbek danych i utworzyła dla nich 3 wielopunktowe prototypy, które w trakcie dalszego uczenia ewoluowały, aby jak najlepiej dopasować swoje kształty, rozmiary oraz położenie do rozkładu danych w zbiorze.




Rys. 7.1. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *Lymphoma* bazujące na próbkach)

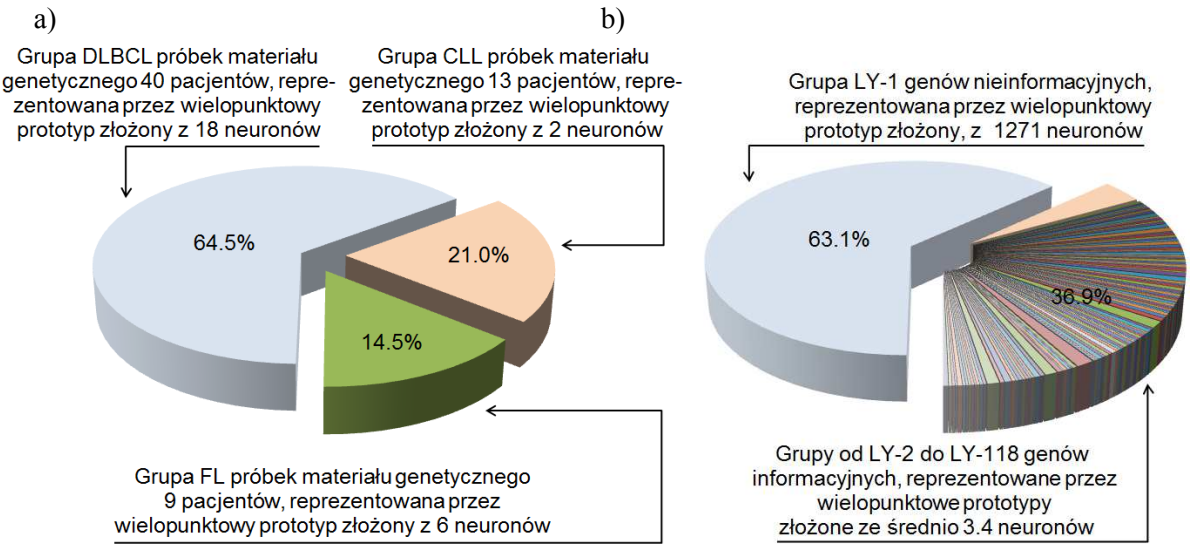
Analogicznie, na rys. 7.2 przedstawiono przebiegi zmian liczby neuronów i liczby topologicznych podstruktur sieci neuronowej (wielopunktowych prototypów grup ge-

nów) podczas grupowania bazującego na genach. Ostatecznie, sieć neuronowa składa się z $m = 1665$ neuronów (rys. 7.2a), rozmieszczonych w 118 niezależnych podstrukturach (rys. 7.2b).




Rys. 7.2. Przebieg zmian liczby neuronów (a) oraz liczby podstruktur (b) proponowanej sieci neuronowej w trakcie procesu uczenia (grupowanie danych zbioru *Lymphoma* bazujące na genach)

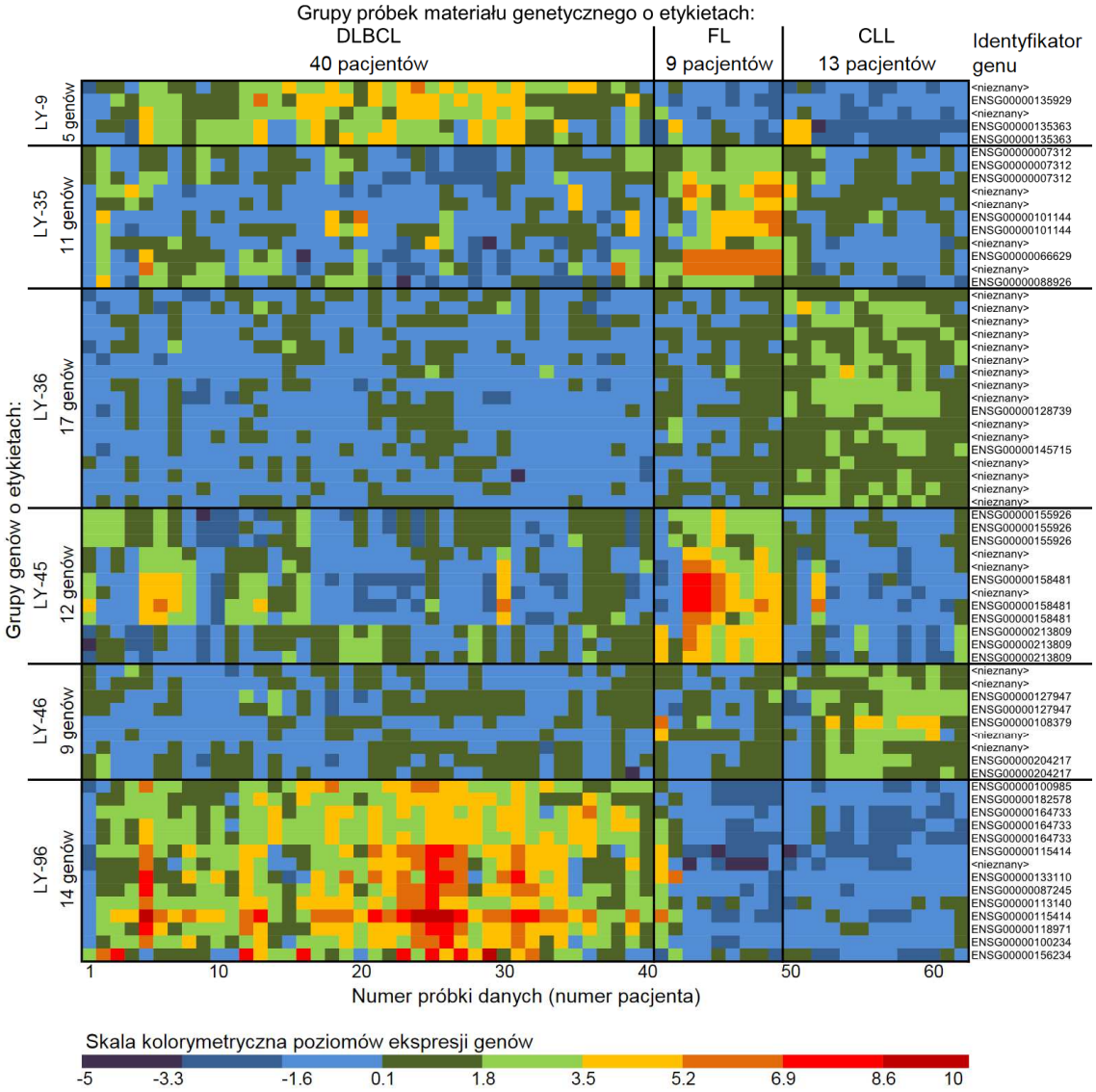
Po zakończeniu procesu uczenia sieci neuronowych, przeprowadzono ich kalibrację, nadając otrzymanym wielopunktowym prototypom grup unikatowe etykiety klas. Trzem prototypom grup próbek danych nadano następujące etykiety: DLBCL dla prototypu złożonego z 18 neuronów (o numerach porządkowych od 1 do 18), CLL dla prototypu składającego się z 2 neuronów (o numerach porządkowych 19 i 20) oraz FL dla prototypu złożonego z 6 neuronów (o numerach porządkowych od 21 do 26). Prototypy te reprezentują grupy, odpowiednio, 40, 13 oraz 9 próbek danych. Z kolei, prototypom grup genów (118 prototypów) przypisano unikatową etykietę według wzorca LY-*i*, gdzie *i* jest kolejnym numerem grupy od 1 do 118. Prototyp LY-1 składający się z 1271 neuronów o numerach porządkowych od 395 do 1665, reprezentuje największą grupę genów (2542 genów; 63.1% całkowitej liczby genów w zbiorze), które zostały uznane za nieinformacyjne (na podstawie analogicznych kryteriów, jak w przypadku prototypów LE-1, ML-1 i ML-2 w rozdziale 5 oraz CO-1 w rozdziale 6). Pozostałe 117 prototypów posiada od 2 do 70 neuronów (średnio 3.4 neuronów na prototyp) i reprezentują grupy od 5 do 150 genów (średnio 12.7 genów na grupę). Proporcje podziału zbioru *Lymphoma* na grupy próbek danych i grupy genów ilustruje rys. 7.3.




Rys. 7.3. Proporcje podziału danych zbioru *Lymphoma* na grupy otrzymane w rezultacie grupowania bazującego na próbkach (a) oraz bazującego na genach (b)

Ze względu na dużą wymiarowość zbioru *Lymphoma* nie jest możliwa prezentacja poziomów ekspresji wszystkich jego genów. Dlatego, na rys. 7.4 i rys. 7.5 przedstawiono poziomy ekspresji genów w wybranych grupach genów (o następujących etykietach: LY-9, LY-35, LY-36, LY-45, LY-46 oraz LY-96), odpowiednio, w formie macierzy ekspresji genów (poziomy ekspresji oznaczono kolorami wg skali kolorymetrycznej) oraz w formie nakładających się wzajemnie krzywych (ilustrujących podobieństwa poziomów ekspresji genów należących do tych samych grup). Na rys. 7.4 oznaczono brakujące identyfikatory genów słowem „<nieznany>”. Linie koloru czarnego (dwie pionowe i pięć poziomych) oznaczają granice pomiędzy otrzymanymi grupami (odpowiednio, pomiędzy grupami próbek danych i pomiędzy grupami genów).

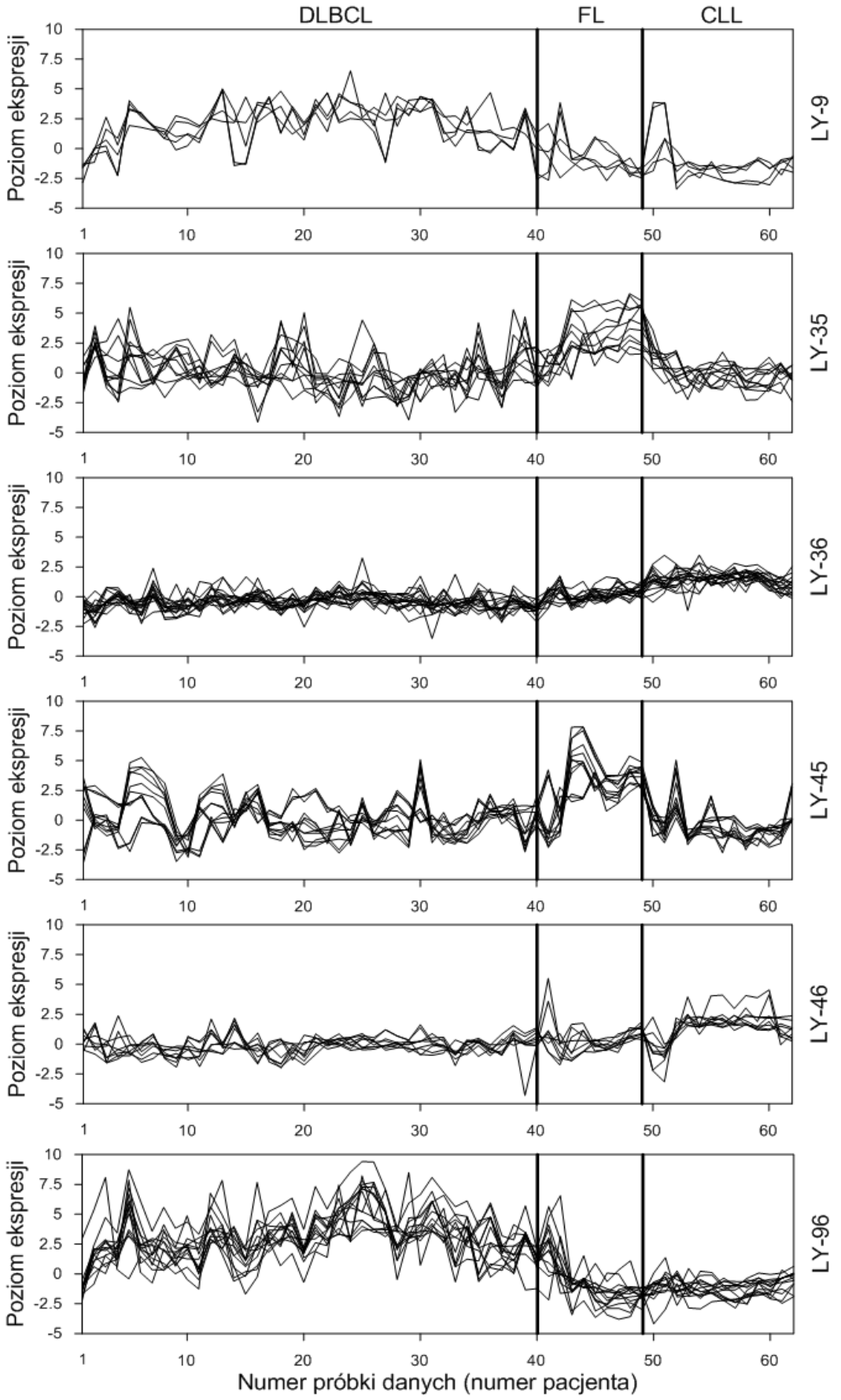
Na potrzeby analizy funkcyjnej genów oraz analizy porównawczej konieczna była konwersja identyfikatorów genów do tzw. formatu *Ensembl* rozpoznawalnego przez aplikację DAVID Functional Annotation Tool (pierwotne identyfikatory udostępnione w ramach zbioru *Lymphoma* nie były rozpoznawalne przez ten program). Do konwersji wykorzystano bazę genów GeneCards: Human Gene Database Instytutu Nauki Weizmanna w Rehovot w Izraelu (http://www.genecards.org). Dla wygody, tabela 7.1 przedstawia pierwotne identyfikatory i ich odpowiedniki w formacie *Ensembl* tych genów, które wykorzystano do analizy efektywności proponowanego narzędzia (wykaz identyfikatorów pozostałych genów, m.in. z rys. 7.4, zawiera załącznik B rozprawy).




Rys. 7.4. Graficzna ilustracja poziomów ekspresji genów ze zbioru *Lymphoma*, wraz z granicami podziału pomiędzy grupami próbek danych i grupami genów

Tabela 7.1. Pierwotne identyfikatory i ich odpowiedniki w formacie *Ensembl* wybranych genów zbioru *Lymphoma* (na podstawie bazy GeneCards: Human Gene Database)

| Lp. | Identyfikator pierwotny | Identyfikator w formacie *Ensembl* | Lp. | Identyfikator pierwotny | Identyfikator w formacie *Ensembl* |
|---|---|---|---|---|---|
| 1 | MMP9 | ENSG00000100985 | 6 | MMP2 | ENSG00000087245 |
| 2 | CSF1 | ENSG00000182578 | 7 | SPARC | ENSG00000113140 |
| 3 | CTSB | ENSG00000164733 | 8 | CCND2 | ENSG00000118971 |
| 4 | FN1 | ENSG00000115414 | 9 | TIMP3 | ENSG00000100234 |
| 5 | OSF2 | ENSG00000133110 | 10 | BCA1 | ENSG00000156234 |




Rys. 7.5. Graficzna ilustracja poziomów ekspresji genów z rys. 7.4

## 7.3. Ocena efektywności grupowania danych

Weryfikacja rezultatów grupowania danych bazującego na próbkach oraz bazującego na genach dla zbioru danych *Lymphoma* zostanie przeprowadzona analogicznie jak w podrozdziałach 5.3 i 6.3. Ze względu na brakujące identyfikatory genów, ocena grupowania bazującego na genach zostanie ograniczona do tych grup, których przynajmniej połowa genów jest znana.

### *Ocena efektywności grupowania danych bazującego na próbkach*

Tabela 7.2 przedstawia rezultaty grupowania danych zbioru *Lymphoma* bazującego na próbkach. Liczby poprawnych i błędnych decyzji zostały wyznaczone na bazie informacji o rzeczywistej przynależności poszczególnych próbek danych do odpowiednich klas (informacja ta nie była wykorzystywana na etapie uczenia sieci neuronowej, które odbywało się w trybie nienadzorowanym).

Otrzymano bardzo dobry wynik, zarówno pod względem całkowitej liczby poprawnych decyzji (93.6%), jak i liczby poprawnych decyzji dla próbek z poszczególnych klas (95.2%, 100% i 77.8% dla próbek klas, odpowiednio, DLBCL, CLL i FL). Tylko 4 próbki danych z 62 próbek zbioru *Lymphoma* zostały błędnie przyporządkowane (2 próbki klasy DLBCL zostały błędnie uznane za próbki klasy FL oraz 2 próbki klasy FL zostały błędnie uznane za próbki klasy CLL).

Tabela 7.2. Rezultaty grupowania danych bazującego na próbkach (zbiór danych *Lymphoma*)

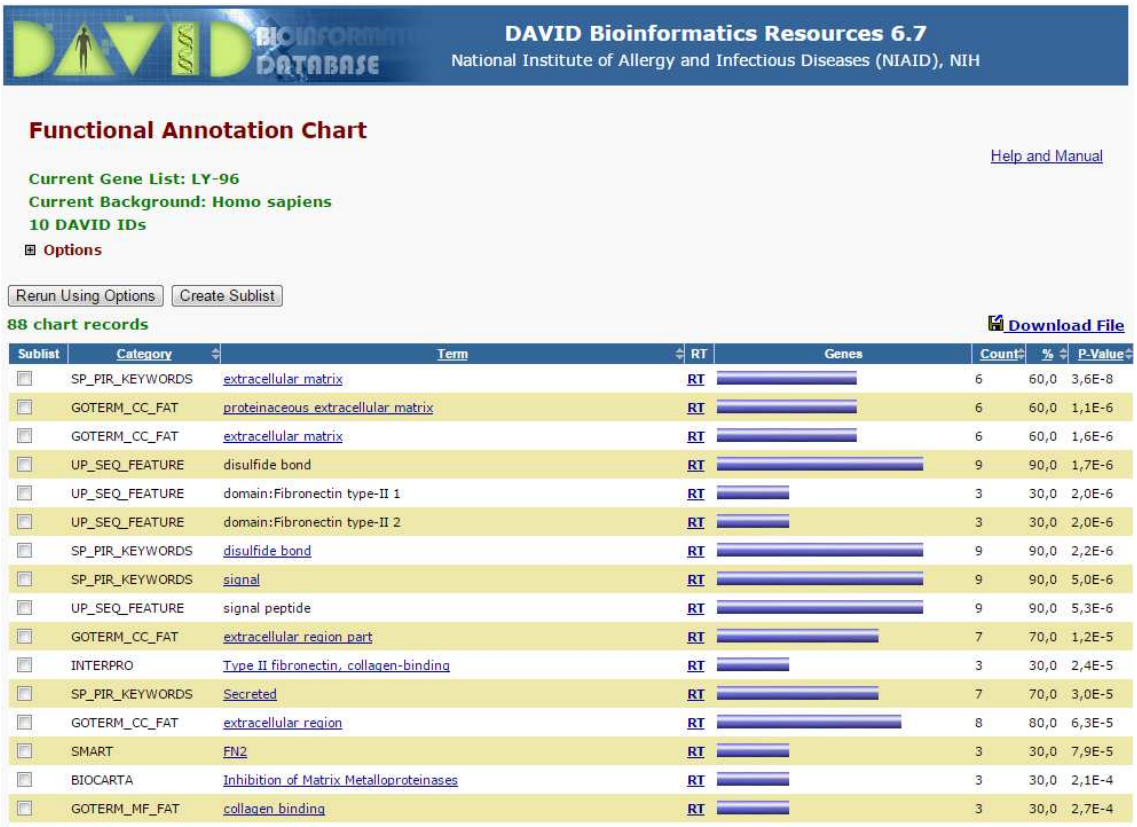
| Etykieta klasy | Liczba próbek danych | Liczba próbek danych odwzorowanych przez podstrukturę o etykiecie: | | | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|---|---|---|
| | | DLBCL | CLL | FL | | | |
| DLBCL | 42 | 40 | 0 | 2 | 40 | 2 | 95.2% |
| CLL | 11 | 0 | 11 | 0 | 11 | 0 | 100% |
| FL | 9 | 0 | 2 | 7 | 7 | 2 | 77.8% |
| **RAZEM** | **62** | **40** | **13** | **9** | **58** | **4** | **93.6%** |

### *Ocena efektywności grupowania danych bazującego na genach*

Ocena efektywności grupowania bazującego na genach zostanie dokonana analogicznie jak w podrozdziałach 5.3 (dla zbiorów danych *Leukemia* i *MLL*) i 6.3 (dla zbio-

ru danych *Colon*). Tym razem jednak, ze względu na brakujące identyfikatory genów w obecnie rozważanym zbiorze danych *Lymphoma*, zakres oceny efektywności grupowania bazującego na genach będzie ograniczony do tych grup genów, które zawierają przynajmniej połowę genów o znanych identyfikatorach.

Na rys. 7.6 przedstawiono rezultaty analizy funkcyjnej przeprowadzonej z wykorzystaniem narzędzia DAVID Functional Annotation Tool i testu Fishera dla przykładowej grupy LY-96 zawierającej 14 genów, w tym jeden o nieznanym identyfikatorze oraz trzy geny, których identyfikatory zostały uznane przez bazę DAVID za równoważne.




| Sublist | Category | Term | RT | Genes | Count | % | P-Value |
|---|---|---|---|---|---|---|---|
| | SP_PIR_KEYWORDS | extracellular matrix | RT | | 6 | 60,0 | 3,6E-8 |
| | GOTERM_CC_FAT | proteinaceous extracellular matrix | RT | | 6 | 60,0 | 1,1E-6 |
| | GOTERM_CC_FAT | extracellular matrix | RT | | 6 | 60,0 | 1,6E-6 |
| | UP_SEQ_FEATURE | disulfide bond | RT | | 9 | 90,0 | 1,7E-6 |
| | UP_SEQ_FEATURE | domain:Fibronectin type-II 1 | RT | | 3 | 30,0 | 2,0E-6 |
| | UP_SEQ_FEATURE | domain:Fibronectin type-II 2 | RT | | 3 | 30,0 | 2,0E-6 |
| | SP_PIR_KEYWORDS | disulfide bond | RT | | 9 | 90,0 | 2,2E-6 |
| | SP_PIR_KEYWORDS | signal | RT | | 9 | 90,0 | 5,0E-6 |
| | UP_SEQ_FEATURE | signal peptide | RT | | 9 | 90,0 | 5,3E-6 |
| | GOTERM_CC_FAT | extracellular region part | RT | | 7 | 70,0 | 1,2E-5 |
| | INTERPRO | Type II fibronectin, collagen-binding | RT | | 3 | 30,0 | 2,4E-5 |
| | SP_PIR_KEYWORDS | Secreted | RT | | 7 | 70,0 | 3,0E-5 |
| | GOTERM_CC_FAT | extracellular region | RT | | 8 | 80,0 | 6,3E-5 |
| | SMART | FN2 | RT | | 3 | 30,0 | 7,9E-5 |
| | BIOCARTA | Inhibition of Matrix Metalloproteinases | RT | | 3 | 30,0 | 2,1E-4 |
| | GOTERM_MF_FAT | collagen binding | RT | | 3 | 30,0 | 2,7E-4 |

Rys. 7.6. Rezultaty testu Fishera dla grupy genów LY-96 wybranej ze zbioru danych *Lymphoma*, generowane przez program DAVID Functional Annotation Tool

Statystycznie najbardziej znaczące funkcje genów grupy LY-96, przy przyjętym poziomie istotności *p-value* nie większym niż $10^{-5}$, prezentuje tabela 7.3. Grupę można uznać za bardzo dobrą, gdyż procentowy udział genów pełniących te same funkcje (tutaj *disulfide bond*, *signal* i *signal peptide*) jest bardzo duży (90% całkowitej liczby ge-

nów). Analogiczne wnioski dotyczą pozostałych grup genów, uwzględnionych w analizie funkcyjnej genów.

Tabela 7.3. Statystycznie najbardziej znaczące funkcje genów grupy LY-96, wybrane spośród rezultatów testu Fishera przedstawionych na rys. 7.6

| Lp. | Nazwa funkcji genów | Geny funkcyjne w grupie genów LY-96 | | | Poziom istotności *p-value* |
|---|---|---|---|---|---|
| | | Ilość | Identyfikatory genów*) | Udział | |
| 1 | *extracellular matrix* (kategoria SP_PIR_KEYWORDS) | 6 | 100234, 115414, 087245, 100985, 133110, 113140 | 60% | $3.6 \cdot 10^{-8}$ |
| 2 | *proteinaceous extracellelar matrix* | 6 | 100234, 115414, 087245, 100985, 133110, 113140 | 60% | $1.1 \cdot 10^{-6}$ |
| 3 | *extracellelar matrix* (kategoria GOTERM_CC_FAT) | 6 | 100234, 115414, 087245, 100985, 133110, 113140 | 60% | $1.6 \cdot 10^{-6}$ |
| 4 | *disulfide bond* (kategoria UP_SEQ_FEATURE) | 9 | 100234, 164733, 156234, 182578, 115414, 087245, 100985, 133110, 113140 | 90% | $1.7 \cdot 10^{-6}$ |
| 5 | *domain:Fibronectin type-II 1* | 3 | 115414, 087245, 100985 | 30% | $2.0 \cdot 10^{-6}$ |
| 6 | *domain:Fibronectin type-II 2* | 3 | 115414, 087245, 100985 | 30% | $2.0 \cdot 10^{-6}$ |
| 7 | *disulfide bond* (kategoria SP_PIR_KEYWORDS) | 9 | 100234, 164733, 156234, 182578, 115414, 087245, 100985, 133110, 113140 | 90% | $2.2 \cdot 10^{-6}$ |
| 8 | *signal* | 9 | 100234, 164733, 156234, 182578, 115414, 087245, 100985, 133110, 113140 | 90% | $5.0 \cdot 10^{-6}$ |
| 9 | *signal peptide* | 9 | 100234, 164733, 156234, 182578, 115414, 087245, 100985, 133110, 113140 | 90% | $5.3 \cdot 10^{-6}$ |

*) 6 ostatnich cyfr identyfikatorów genów formatu *Ensembl* ENSG00000XXXXXX (gdzie X oznacza cyfrę).

## 7.4. Analiza porównawcza

Funkcjonowanie proponowanej, uogólnionej samoorganizującej się sieci neuronowej w grupowaniu danych bazującym na próbkach zostało porównane z wynikami czterech metod: *k-means*, FFTA, EM oraz DSOM, analogicznie jak w poprzednich eksperymentach przedstawionych w podrozdziałach 5.4 i 6.4. Z kolei, funkcjonowanie sieci w grupowaniu bazującym na genach zostało porównane tylko z wynikami metody DSOM. Przeprowadzono węższy zakres analizy porównawczej (w porównaniu z analogiczną analizą z wcześniejszych rozdziałów), gdyż nie znaleziono żadnych pozycji lite-

raturowych, w których rozważane byłyby alternatywne metody grupowania danych w zbiorze *Lymphoma*. Prawdopodobnie brak wszystkich identyfikatorów genów jest przyczyną mniejszego zainteresowania tym zbiorem w literaturze dotyczącej problematyki grupowania danych opisujących ekspresję genów.

### *Analiza porównawcza rezultatów grupowania danych bazującego na próbkach*

Rezultaty grupowania danych bazującego na próbkach, w zbiorze danych *Lymphoma*, z wykorzystaniem proponowanej sieci USOM i metod alternatywnych: *k-means*, FFTA, EM oraz DSOM przedstawia tabela 7.4.

Tabela 7.4. Rezultaty grupowania bazującego na próbkach dla zbioru *Lymphoma* z wykorzystaniem różnych metod grupowania danych

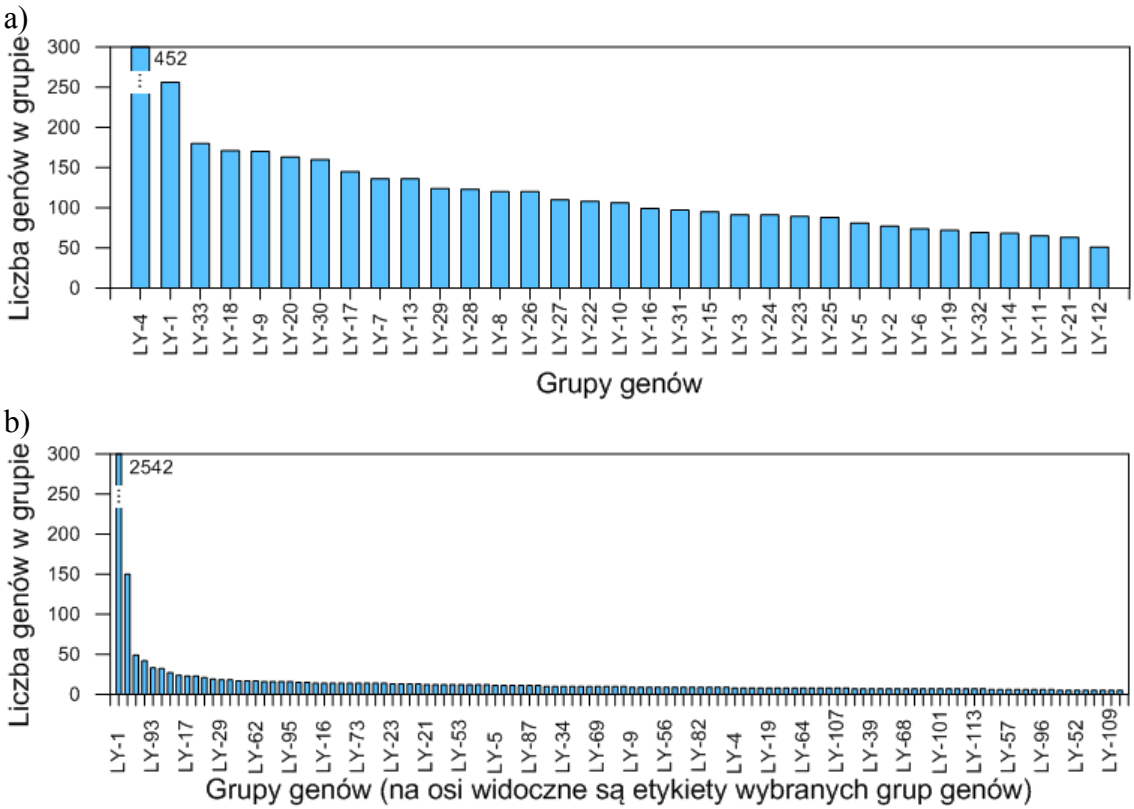
| Lp. | Metoda grupowania danych*) | Liczba poprawnych decyzji | Liczba błędnych decyzji | Procent poprawnych decyzji |
|---|---|---|---|---|
| 1 | *k-means* | 42 | 20 | 67.7% |
| 2 | EM | 38 | 24 | 61.3% |
| 3 | FFTA | 47 | 15 | 75.8% |
| 5 | DSOM | 57 | 5 | 91.9% |
| 6 | **USOM** | **58** | **4** | **93.6%** |

*) patrz tabela 5.6.

Najlepszy rezultat otrzymano dla proponowanej sieci USOM (93.6% poprawnych decyzji o przynależności próbek danych do poszczególnych grup), natomiast drugi w kolejności otrzymano dla sieci DSOM (91.9% poprawnych decyzji). Pozostałe metody dały wyniki znacząco gorsze. Na podkreślenie zasługuje fakt, że USOM i DSOM dodatkowo automatycznie wykryły prawidłową liczbę grup z zbiorze *Lymphoma* (w przypadku pozostałych metod liczba ta była zakładana z góry).

### *Analiza porównawcza rezultatów grupowania danych bazującego na genach*

Analiza porównawcza funkcjonowania metod USOM i DSOM w grupowaniu genów została przeprowadzona analogicznie jak w poprzednich eksperymentach z rozdziału 5 i 6. Histogram liczby genów w poszczególnych grupach otrzymanych z wykorzystaniem metod USOM oraz DSOM przedstawia rys. 7.7.




Rys. 7.7. Histogramy liczby genów w poszczególnych grupach genów w zbiorze danych *Lymphoma*, otrzymanych z wykorzystaniem metod DSOM (a) oraz USOM (b)

Sieć neuronowa DSOM wykryła automatycznie 33 grupy genów, zawierających od 59 do 293 genów (średnio 122 geny na grupę). Z kolei, proponowana sieć neuronowa USOM wykryła automatycznie 118 grup genów, przy czym jedną z nich o wielkości 2542 genów (63.1% całkowitej liczby genów) uznano za grupę genów nieinformacyjnych. Spośród pozostałych 117 grup genów informacyjnych o wielkości od 5 do 150 genów (średnio 12.7 genów na grupę), można wyróżnić 107 grup (90.7% całkowitej liczby grup) o wielkości nie większej niż 20 genów włącznie. Analiza potwierdza wyraźną przewagę sieci USOM nad DSOM pod względem zdolności do generowania grup o względnie małej liczbie genów, które niosą potencjalnie wartościowe informacje z punktu widzenia badań medycznych.

Porównanie metod DSOM i USOM pod względem procentowego udziału genów pełniących te same funkcje w grupie zostało ograniczone do grup genów zawierających co najmniej połowę genów o znanych identyfikatorach. Tabela 7.5 przedstawia po trzy grupy genów wybrane dla każdej z rozważanych metod oraz rezultaty testów Fishera, przeprowadzonych dla każdej z nich, z wykorzystaniem programu DAVID Functional

Annotation Tool. Do porównania wybrano grupy, dla których średni procentowy udział genów pełniących te same funkcje jest największy (parametr ten został wyznaczony analogicznie jak w podrozdziałach 5.4 i 6.4). Zdecydowanie największy średni udział genów funkcyjnych w grupach zbioru danych *Lymphoma* można zaobserwować dla proponowanej sieci neuronowej USOM (od 48.6% dla grupy LY–49 do 60.0% dla grupy LY–96), co potwierdza wykazaną już w poprzednich eksperymentach przewagę sieci USOM nad metodami alternatywnymi (tutaj nad metodą DSOM) pod względem generowania grup genów o największym procentowym udziale genów funkcyjnych.

Tabela 7.5. Udział liczby genów funkcyjnych w wybranych grupach genów ze zbioru *Lymphoma,* otrzymanych metodami DSOM oraz USOM

| L.p. | Metoda grupowania danych[*1)] | Nazwa grupy genów | Liczba genów w grupie[*2)] | Średnia liczba genów funkcyjnych | Średni udział genów funkcyjnych | Średni poziom istotności *p-value* |
|---|---|---|---|---|---|---|
| 1 | DSOM | LY-3 | 50 (91) | 9.4 | 18.8% | $3.1 \cdot 10^{-5}$ |
| | | LY-8 | 61 (120) | 11 | 18.0% | $3.1 \cdot 10^{-6}$ |
| | | LY-18 | 60 (171) | 16.6 | 27.7% | $4.1 \cdot 10^{-8}$ |
| 2 | **USOM** | LY-49 | 7 (10) | 3.4 | 48.6% | $3.3 \cdot 10^{-4}$ |
| | | LY-96 | 10 (14) | 6 | 60.0% | $3.3 \cdot 10^{-7}$ |
| | | LY-114 | 7 (12) | 3.8 | 54.3% | $1.4 \cdot 10^{-2}$ |

[*1)] patrz tabela 5.6, [*2)] liczba genów ze znanymi etykietami, które były brane pod uwagę w testach Fishera oraz liczba wszystkich genów w grupie (w nawiasach).

## 7.5. Podsumowanie

Niniejszy rozdział był trzecim z kolei i zarazem ostatnim prezentującym zastosowania uogólnionych samoorganizujących się sieci neuronowych o ewoluujących, drzewopodobnych strukturach topologicznych w problemach grupowania danych opisujących ekspresję genów. Tym razem rozważono zbiór danych *Lymphoma*, który zawiera poziomy ekspresji genów pochodzących z próbek materiału genetycznego pacjentów chorych na trzy typy nowotworu układu limfatycznego.

Przeprowadzono procesy uczenia sieci neuronowej funkcjonującej jako narzędzie zarówno do grupowania bazującego na próbkach, jak i do grupowania bazującego na genach, analizę efektywności proponowanego narzędzia oraz analizę porównawczą z innymi, do pewnego stopnia alternatywnymi podejściami. Eksperyment potwierdził

wnioski płynące z poprzednich dwóch rozdziałów. Efektywność proponowanego narzędzia jest bardzo wysoka i najlepsza (w porównaniu z efektywnością pozostałych rozważanych metod), zarówno pod względem grupowania danych bazującego na próbkach (93.6% poprawnych decyzji), jak i grupowania danych bazującego na genach (najwięcej grup o względnie małej liczbie genów i największym udziale genów funkcyjnych w grupach).

# 8. Podsumowanie

Niniejsza praca dotyczy tematyki budowy możliwie uniwersalnych narzędzi teoretycznych do efektywnego rozwiązywania problemów grupowania danych w różnorodnych, złożonych i wielowymiarowych zbiorach danych, w tym danych opisujących ekspresję genów. Praca proponuje w tym zakresie oryginalne narzędzie – uogólnioną samoorganizującą się sieć neuronową o ewoluującej, drzewopodobnej strukturze topologicznej oraz ideę grupowania danych, bazującą na konstrukcji i analizie drzewopodobnych, wielopunktowych prototypów skupisk danych, których liczba, kształt, rozmiar i lokalizacja w przestrzeni danych są automatycznie wyznaczane w ramach procesu uczenia nienadzorowanego proponowanej sieci neuronowej.

Praca składa się z dwóch części: teoretycznej oraz aplikacyjnej. Część teoretyczna, obejmująca podrozdziały 1.1 i 1.2 Wprowadzenia oraz rozdziały 2 i 3, poświęcona jest w pierwszej kolejności przeglądowi najważniejszych zagadnień dotyczących grupowania danych (Wprowadzenie), a następnie (w rozdziale 2) konwencjonalnym, samoorganizującym się sieciom neuronowym oraz ich różnorodnym wariantom możliwym do wykorzystania w grupowaniu danych jak również analizie istotnych mankamentów i ograniczeń tych sieci w problemach grupowania. Rozważania te stanowią koncepcyjny punkt wyjścia do przedstawionej (w rozdziale 3) konstrukcji proponowanej, uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej, jako teoretycznego narzędzia grupowania danych. Część aplikacyjna, obejmująca rozdziały 4, 5, 6 i 7, poświęcona jest obszernemu testowi praktycznej użyteczności proponowanego narzędzia w różnorodnych problemach grupowania danych, ze szczególnym uwzględnieniem grupowania danych w zbiorach opisujących ekspresję genów.

Rozdział 2 prezentuje podstawowe zagadnienia dotyczące konstrukcji i procesu uczenia konwencjonalnych, samoorganizujących się sieci neuronowych (SOM) oraz klasyczną ideę grupowania danych bazującą na konstrukcji i wizualnej analizie struktury topologicznej neuronów (tzw. mapy neuronów lub U-macierzy). Następnie przedstawia pewną ewolucję tej idei poprzez stopniowe wprowadzenie do konwencjonalnej sieci SOM – w ramach różnych jej wariantów – mechanizmów umożliwiających modyfikację struktury topologicznej neuronów w trakcie procesu uczenia. Rozważono 9 wariantów SOM wybranych z literatury: dwie sieci zdolne tylko do powiększania swojej

struktury topologicznej (poprzez aktywację mechanizmów umożliwiających dodawanie do niej nowych neuronów), kolejne dwie, zdolne nie tylko do powiększania struktury topologicznej, ale również do jej podziału na podstruktury (poprzez aktywację mechanizmów usuwających połączenia topologiczne pomiędzy neuronami) oraz pięć wariantów SOM, które posiadają zdolność do zarówno powiększania, jak i redukcji rozmiaru struktury topologicznej (poprzez dodawanie i usuwanie neuronów) oraz do dzielenia jej na części. Analiza ograniczeń i mankamentów tych sieci w problemach grupowania danych stanowi koncepcyjny punkt wyjścia do proponowanego w pracy oryginalnego rozwiązania.

W rozdziale 3 przedstawiono konstrukcję proponowanej, uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej (USOM) oraz propozycje idei grupowania danych, bazującej na automatycznym budowaniu (z wykorzystaniem USOM) i analizie drzewopodobnych, wielopunktowych prototypów skupisk danych. Szczegółowo omówiono wszystkie mechanizmy modyfikujące strukturę sieci USOM, umożliwiające redukcję i powiększanie rozmiaru struktury sieci oraz dzielenie jej na części i ponowne ich łączenie, jak również warunki aktywacji tych mechanizmów w trakcie procesu uczenia z wykorzystaniem algorytmu uczenia nienadzorowanego WTM.

W rozdziale 4, pierwszym z czterech rozdziałów części aplikacyjnej pracy, przedstawiono dwuetapowy, praktyczny test użyteczności proponowanego podejścia w różnych kategoriach problemów grupowania danych. W pierwszym etapie, rozważono 10 syntetycznych dwu- i trójwymiarowych zbiorów danych, pochodzących m.in. z pakietu Fundamental Clustering Problem Suite (FCPS), dostępnego na serwerze Uniwersytetu Philipps'a w Marburgu w Niemczech (https://www.uni-marburg.de/fb12/datenbionik/data?language_sync=1) oraz w pracy [104], które są uznawane za standardowe testy różnorodnych technik grupowania danych. W drugim etapie, rozważono 3 rzeczywiste, wielowymiarowe zbiory danych, dostępne na serwerze Uniwersytetu Kalifornijskiego w Irvine (http://archive.ics.uci.edu/ml), które są powszechnie wykorzystywane jako testy odniesienia w analizach porównawczych różnych algorytmów z obszaru inteligencji obliczeniowej, m.in. technik grupowania danych.

W rozdziałach 5, 6 i 7, proponowane narzędzie zostało wykorzystane do grupowania danych w rzeczywistych, złożonych i wielowymiarowych zbiorach danych opisujących ekspresję genów. Rozważono cztery zbiory danych reprezentujące problemy dia-

gnozowania chorób nowotworowych: układu krwionośnego (zbiory danych *Leukemia* i *Mixed Lineage Leukemia* (*MLL*)), jelita grubego (zbiór danych *Colon*) oraz układu limfatycznego (zbiór danych *Lymphoma*). Wszystkie zbiory danych są dostępne na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/datasets.html). Celem testów było wykazanie wysokiej skuteczności proponowanego narzędzia w wykrywaniu różnych typów chorób (podział przypadków choroby na dwie lub trzy kategorie) oraz w wykrywaniu grup genów pełniących podobne funkcje w organizmie człowieka, które to grupy genów mogą być odpowiedzialne za dany typ choroby nowotworowej.

Na zakończenie każdego z powyższych trzech rozdziałów przeprowadzono analizę porównawczą wyników proponowanej sieci neuronowej oraz innych, do pewnego stopnia alternatywnych, metod. Wyniki tych analiz wyraźnie wskazują, że spośród wszystkich rozważanych technik grupowania danych, proponowana sieć neuronowa jest najbardziej efektywnym narzędziem – zarówno w sensie wysokiego procentu poprawnych decyzji dla grupowania danych bazującego na próbkach, jak i bardzo dobrej oceny rezultatów grupowania danych bazującego na genach. Ponadto, proponowane podejście wykazało się najlepszą zdolnością do automatycznego wyznaczania wielopunktowych prototypów skupisk danych, w tym ich prawidłowej (tzn. równej liczbie skupisk danych) liczby, kształtów, rozmiarów i lokalizacji, odpowiednio, do postaci, wielkości i lokalizacji wykrytych skupisk w przestrzeni danych.

Podsumowując, zarówno rozważania teoretyczne jak i przeprowadzone badania eksperymentalne wraz z analizą porównawczą z alternatywnymi metodami pozwalają stwierdzić, że wykazana została prawdziwość tezy głównej przedstawionej we Wprowadzeniu mówiącej, że: uogólniona samoorganizująca się sieć neuronowa o ewoluującej drzewopodobnej strukturze może być wykorzystana jako efektywne i uniwersalne narzędzie teoretyczne w problemach grupowania danych w różnorodnych, złożonych i wielowymiarowych zbiorach danych, w tym danych opisujących ekspresję genów. Rozwinięciem powyższego wniosku są następujące wnioski szczegółowe:

1. Proponowane podejście posiada zdolność do automatycznego wykrywania liczby skupisk w zbiorach danych.
2. Proponowane podejście posiada zdolność do automatycznego generowania wielopunktowych prototypów dla poszczególnych skupisk danych, przy czym liczba,

rozmiary, kształty i lokalizacja tych prototypów w przestrzeni danych są również automatycznie „dostrajane" w taki sposób, aby jak najlepiej odwzorować faktyczny rozkład skupisk w zbiorze danych.

3. Proponowane podejście posiada zdolność wykrywania skupisk danych o nieregularnych, zróżnicowanych kształtach i wielkościach (w tym, zarówno skupisk „cienkich", szkieletowych jak i objętościowych), jak również skupisk o zróżnicowanej gęstości zawartych w nich danych.
4. Proponowane podejście – w przypadku grupowania „ostrego" – umożliwia automatyczne przyporządkowanie poszczególnych próbek danych do określonych grup.
5. Proponowane podejście umożliwia również budowę rozmytego modelu grupowania danych poprzez określenie stopnia przynależności poszczególnych danych do określonych grup.
6. Proponowane podejście przetwarza dane w trybie uczenia nienadzorowanego.
7. Proponowane podejście posiada zdolność do uogólniania nabytej wiedzy na przypadki nie występujące w fazie uczenia (praca systemu jako klasyfikatora).
8. Proponowane podejście jest w stanie efektywnie przetwarzać duże zbiory danych pochodzące z baz danych lub zasobów sieci Internet, w tym dane opisujące ekspresję genów i będące rezultatem procesów przetwarzania łańcuchów DNA z wykorzystaniem tzw. mikromacierzy.

Do najważniejszych, oryginalnych osiągnięć autora, zawartych w niniejszej pracy, należy zaliczyć:

1. Opracowanie koncepcji grupowania danych bazującej na konstrukcji i analizie wielopunktowych prototypów (o drzewopodobnych kształtach) dla poszczególnych skupisk danych.
2. Konstrukcja oryginalnego narzędzia grupowania danych – uogólnionej samoorganizującej się sieci neuronowej o ewoluującej, drzewopodobnej strukturze topologicznej – umożliwiającego generowanie wielopunktowych prototypów skupisk wykrytych w zbiorach danych. Generowanie prototypów odbywa się w trakcie nienadzorowanego uczenia sieci. Wspomniane narzędzie posiada zdolność do:
   – automatycznego generowania po jednym prototypie dla każdego wykrytego skupiska danych,

- automatycznego „dostrajania” położenia tych prototypów w przestrzeni danych odpowiednio do lokalizacji skupisk,
- automatycznego „dostrajania” liczby punktów każdego z prototypów i jego kształtu stosownie do wielkości i kształtu reprezentowanego przez niego skupiska danych oraz gęstości danych w nim zawartych.

3. Opracowanie następujących mechanizmów umożliwiających modyfikację topologicznej struktury sieci w trakcie procesu uczenia:
   - mechanizmu usuwania pojedynczych małoaktywnych neuronów ze struktury sieci, ulokowanych zwykle w obszarach o małej gęstości danych lub ich braku,
   - mechanizmu usuwania pojedynczych połączeń topologicznych pomiędzy neuronami, w celu podziału struktury sieci na dwie podstruktury, które w trakcie dalszego uczenia mogą się znów dzielić na części, w celu lepszego odwzorowania skupisk danych,
   - mechanizmu usuwania podstruktur sieci, które zawierają zbyt małą liczbę neuronów i, w związku z tym, zwykle nie odwzorowują skupiska danych,
   - mechanizmu wstawiania do struktury (podstruktury) sieci dodatkowych neuronów w otoczeniu neuronów nadaktywnych w celu przejęcia części ich aktywności, co przekłada się na „przesuwanie” neuronów do obszarów o wyższej gęstości danych,
   - mechanizmu wstawiania dodatkowych połączeń topologicznych pomiędzy określonymi niepołączonymi neuronami, w celu ponownego łączenia wybranych podstruktur sieci (utworzenia większego prototypu na bazie dwóch mniejszych),
4. Opracowanie algorytmu uczenia proponowanej sieci neuronowej poprzez rozbudowanie algorytmu uczenia konkurencyjnego WTM o wymienione wyżej dodatkowe mechanizmy, w tym sformułowanie warunków ich automatycznej i wielokrotnej aktywacji celem stopniowej modyfikacji struktury sieci.
5. Opracowanie implementacji komputerowej proponowanego narzędzia teoretycznego.
6. Rozbudowany test praktycznej użyteczności proponowanej sieci neuronowej w problemach grupowania danych w różnorodnych, złożonych i wielowymiarowych zbiorach danych, w tym:
   - grupowanie danych w syntetycznych dwu- i trójwymiarowych zbiorach danych, uznawanych za standardowe testy (ang. *benchmarks*) reprezentujące różne kate-

gorie problemów grupowania, w tym danych, pochodzących z Fundamental Clustering Problem Suite (FCPS), dostępnych na serwerze Uniwersytetu Philipps'a w Marburgu w Niemczech (https://www.uni-marburg.de/fb12/daten-bionik/data ?language_sync=1),

- grupowanie danych w rzeczywistych, wielowymiarowych zbiorach danych, dostępnych na serwerze Uniwersytetu Kalifornijskiego w Irvine (http://archive.ics.uci.edu/ml), które są powszechnie wykorzystywane w różnorodnych analizach porównawczych dotyczących, m.in. technik grupowania danych,
- grupowanie danych w rzeczywistych, złożonych i wielowymiarowych zbiorach danych medycznych opisujących ekspresję genów, dostępnych na serwerze Uniwersytetu Shenzhen w Chinach (College of Computer Science & Software Engineering; http://csse.szu.edu.cn/staff/zhuzx/datasets.html); rozważono cztery zbiory danych reprezentujące problemy diagnozowania chorób nowotworowych układu krwionośnego, jelita grubego oraz układu limfatycznego.

7. Przeprowadzenie analizy porównawczej proponowanego podejścia z szeregiem innych, do pewnego stopnia alternatywnych, metodologii.

Spośród dalszych kierunków badań dotyczących zarówno rozwoju proponowanych w pracy uogólnionych samoorganizujących się sieci neuronowych o ewoluujących, drzewopodobnych strukturach jak i ich zastosowań, wymienić można między innymi:

- udoskonalanie mechanizmów modyfikujących topologiczną strukturę sieci neuronowej, m.in. poszukiwanie bardziej zaawansowanych procedur aktywujących te mechanizmy, bazujących na analizie rozkładu gęstości danych w zbiorach danych,
- uogólnienie drzewopodobnej struktury sieci neuronowej na przypadek sieci o strukturze grafu, w której połączenia topologiczne między neuronami mogą tworzyć zamknięte „ścieżki",
- zastosowanie proponowanej sieci neuronowej w innych niż grupowanie danych obszarach, np. do wspomagania procesów optymalizacji numerycznej, wspomagania automatycznego (nienadzorowanego) budowania klasyfikatorów (w tym, rozmytych klasyfikatorów regułowych) na podstawie danych; planowane jest podjęcie badań w tym zakresie, jako kontynuacja wcześniejszych prac autora [90, 91, 96, 97], nie wymienionych dotychczas w niniejszej rozprawie.

## Literatura


1. Alizadeh A. A., Eisen M. B., Davis R. E., et al.: *Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling*. Nature, vol. 403, 2000, pp. 503 – 511.
2. Alon U.: *Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays*. In: Proceedings of the National Academy of Sciences of the United States of America, vol. 96, no.12, 1999, pp. 6745 – 6750.
3. Armstrong S. A.: *MLL translocations specify a distinct gene expression profile that distinguishes a unique Leukemia*. Nature Genetics, vol. 30, 2002, pp. 41 – 47.
4. Ayadi W., Elloumi M., Hao J. K.: *A biclustering algorithm based on a bicluster enumeration tree: Application to DNA microarray data*. BioData Mining, vol. 2, 2009.
5. Ayadi W., Elloumi M., Hao J. K.: *BicFinder: a biclustering algorithm for microarray data analysis*. Knowledge and Information Systems: An International Journal, vol. 30, no. 2, 2012, pp. 341 – 358.
6. Ayadi T., Hamdani T. M., Alimi A. M.: *Enhanced data topology preservation with multilevel interior growing self-organizing maps*. Lecture Notes in Engineering and Computer Science, vol. 2183, no. 1, 2010, pp. 154 – 159.
7. Bandyopadhyay S., Maulik U.: *An evolutionary technique based on K-Means algorithm for optimal clustering in $R^N$*. Information Sciences, vol. 146, no. 1 – 4, 2002, pp. 221 – 237.
8. Ben-Dor A., Chor B., Karp R., Yakhini Z.: *Discovering local structure in gene expression data: the order–preserving submatrix problem*. Journal of Computational Biology, vol. 10, no. 3 – 4, 2003, pp. 373 – 384.
9. Bezdek J. C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, 1981.
10. Bezdek J. C., Pal N. R., Tsao E. K.: *Generalized clustering networks and Kohonen's self-organizing scheme*. IEEE Transactions on Neural Networks, vol. 4, no. 4, 1993, pp. 549 – 557.
11. Bezdek J. C., Boggavaparu S., Hall L. O., Bensaid A.: *Genetic algorithm guided clustering*. In proc. of the First IEEE Conference on Evolutionary Computation, vol. 1, 1994, pp. 34 – 39.
12. Bezdek J. C., Keller J., Krisnapuram R., Pal N. R.: *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Springer Science & Business Media, New York, 2005.
13. Bezdek J. C., Reichherzer T. R., Lim G. S., Attikiouzel Y.: *Multiple-prototype classifier design*. IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews, vol. 28, no. 1, 1998, pp. 67 – 79.
14. Blackmore J.: *Incremental grid growing: encoding high-dimensional structure into a two-dimensional feature map*. In proc. of the 1993 IEEE International Conference on Neural Networks (ICNN), vol. 1, 1993, pp. 450 – 455.
15. Blatt M., Wiseman S., Domany E.: *Superparamagnetic clustering of data.* Physical Review Letters, vol. 76, 1996, pp. 3251 – 3254.

16. Braekeleer M. D., Morel F., Bris M-J. L., et al.: *The MLL gene and translocations involving chromosomal band 11q23 in acute leukemia*. Anticancer Research, vol. 25, 2005, pp. 1931 – 1944.
17. Busygin S., Jacobsen G., Kramer E.: *Double conjugated clustering applied o leukemia microarray data*. In proc. of the 2nd SIAM International Conference on Data Mining (ICDM), Workshop on Clustering High Dimensional Data, 2002.
18. Carpenter G. A., Grossberg S.: *Adaptive resonance theory*. In: Arbib M.A. (ed.), The Handbook of Brain Theory and Neural Networks, Second Edition, MIT Press, Cambridge, MA, USA, 2002.
19. Carpenter G. A., Grossberg S.: *Adaptive resonance theory*. In: Sammut C., Webb G. (Eds.), Encyclopedia of Machine Learning and Data Mining, Springer-Verlag, Berlin, 2015.
20. Chial H.: *DNA sequencing technologies key to the Human Genome Project*. Nature Education, vol. 1, no. 1, 2008.
21. Cheng G., Zell A.: *Externally growing cell structures for pattern classification*. 2000.
22. Cheng Y., Church G. M.: *Biclustering of expression data*. In proc. of the 8th International Conference on Intelligent Systems for Molecular Biology (ISMB'00), 2000, pp. 93 – 103.
23. Cho H., Dhillon I. S.: *Coclustering of human cancer microarrays using minimum sum-squared residue coclustering*. IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 5, no. 3, 2008, pp. 385 – 400.
24. Chung D., Keles S.: *Sparse partial least squares classification for high dimensional data*. Statistical Applications in Genetics and Molecular Biology, vol. 9, no. 1, 2010.
25. Cios K., Pedrycz W., Swinarski R.: *Data Mining Methods for Knowledge Discovery*. Kluwer Academic Publishers, Boston, Dordrecht, London, 2000.
26. Dettling M.: *BagBoosting for tumor classification with gene expression data*. Bioinformatics, vol. 20, no. 18, 2004, pp. 3583 – 3593.
27. Dempster A., Laird N., Rubin D.: *Maximum likelihood from incomplete data via the EM algorithm*. Journal of the Royal Statistical Society, Series B, vol. 30, no.1, 1977, pp. 1 – 38.
28. Dennis G. Jr., Sherman B. T., Hosack D. A., et al.: *DAVID: Database for annotation, visualization, and integrated discovery*. Genome Biology, vol. 4, no. 9, 2003.
29. Deodhar M., Cho H., Gupta G., et al.: *Bregman bubble coclustering: A robust framework for mining dense cluster.* ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 2, no. 2, 2008.
30. Duda R., Hart P., Stork D. G.: *Pattern Classification*. John Wiley & Sons, New York, 2000.
31. Dudoit S., Fridlyand J.: *A prediction-based resampling method for estimating the number of clusters in a dataset*. Genome Biology, vol. 3, no. 7, 2002, pp. 1 – 21.
32. Dunn J. C.: *A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters*. Journal of Cybernetics, vol. 3, no. 3 1973, pp. 32 – 57.
33. Eisen M. B., Spellman P. T., Brown P. O., Botstein D.: *Cluster analysis and display of genome-wide expression patterns*. In: Proceedings of the National Academy of Science of the United States of America, vol. 95, no. 25, 1998, pp. 14863 – 14868.

34. Engelbrecht A. P.: *Computational Intelligence: An Introduction*. John Wiley & Sons, 2nd edition, 2007.
35. Ester M., Kriegel H. P., Sander J., Xu X.: *A density-based algorithm for discovering clusters in large spatial databases with noise*. In proc. of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, 1996, pp. 226 – 231.
36. Everitt B. S., Landau S., Leese M., Stahl D.: *Cluster Analysis*. 5th Edition, Wiley Series in Probability and Statistics. John Wiley & Sons, 2011.
37. Fritzke B.: *Growing cell structures - A self-organizing network for unsupervised and supervised learning*. Neural Networks, vol. 7, no.9, 1994, pp. 1441 – 1460.
38. Fritzke B.: *A growing neural gas network learns topologies*. Advances in Neural Information Processing Systems, vol. 7, 1995, pp. 626 – 632.
39. Fritzke B.: *Growing Grid - a self-organizing network with constant neighborhood range and adaptation strength*. Neural Processing Letters, vol. 2, no. 5, 1995, pp. 9 – 13.
40. Getz G., Levine E., Domany E.: *Coupled two-way clustering analysis of gene microarray data*. In: Proceedings of the Natural Academy of Sciences of the United States of America, vol. 97, no. 22, 2000, pp. 12079 – 12084.
41. Goldberg D. E.: *Algorytmy Genetyczne i Ich Zastosowania*. WNT, Warszawa, 1998.
42. Golub T. R., Slonim D. K., Tamayo P., et al.: *Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring*. Science, vol. 286, no.15, 1999, pp. 531 – 537.
43. Gorzałczany M. B., Rudziński F.: *Application of genetic algorithms and Kohonen networks to cluster analysis*. In: L. Rutkowski, J. H. Siekmann, R. Tadeusiewicz, L. A. Zadeh (Eds.), *Artificial Intelligence and Soft Computing – ICAISC 2004* (*Lecture Notes in Computer Science*, vol. 3070), Springer, Berlin, Heidelberg, New York, 2004, pp. 556 – 561.
44. Gorzałczany M. B., Rudziński F.: *Modified Kohonen networks for complex cluster-analysis problems*. In: L. Rutkowski, J. H. Siekmann, R. Tadeusiewicz, L. A. Zadeh (Eds.), *Artificial Intelligence and Soft Computing – ICAISC 2004* (*Lecture Notes in Computer Science*, vol. 3070), Springer, Berlin, Heidelberg, New York, 2004, pp. 552 – 567.
45. Gorzałczany M. B., Rudziński F.: *Algorytmy genetyczne oraz sieci Kohonena w zagadnieniach wyznaczania skupisk (klastrów) w złożonych zbiorach danych*. Zeszyty Naukowe Politechniki Świętokrzyskiej, Elektryka 41, Kielce, 2004, str. 55 – 69.
46. Gorzałczany M. B., Rudziński F.: *Cluster analysis via dynamic self-organizing neural networks*. In: L. Rutkowski, R. Tadeusiewicz, L. A. Zadeh, J. Żurada (Eds.), *Artificial Intelligence and Soft Computing – ICAISC 2006* (*Lecture Notes in Computer Science*, vol. 4029), Springer, Berlin, Heidelberg, New York, 2006, pp. 593 – 602.
47. Gorzałczany M. B., Rudziński F.: *WWW-newsgroup-document clustering by means of dynamic self-organizing neural networks*. In: L. Rutkowski, R. Tadeusiewicz, L. A. Zadeh, J. M. Żurada (Eds.), *Artificial Intelligence and Soft Computing – ICAISC 2008* (*Lecture Notes in Computer Science*, vol. 5097), Springer, Berlin, Heidelberg, New York, 2008, pp. 40 – 51.

48. Gorzałczany M. B., Rudziński F.: *Clustering and filtering of measurement data based on dynamic self-organizing neural networks*. Pomiary Automatyka Kontrola, vol. 56, no. 12, 2010, pp. 1416 – 1419.
49. Gorzałczany M. B., Rudziński F.: *Generalized self-organizing maps for automatic determination of the number of clusters and their multi-prototypes in cluster analysis*. IEEE Transactions on Neural Networks and Learning Systems, *submitted*.
50. Gorzałczany M. B., Piekoszewski J., Rudziński F.: *Generalized tree-like self-organizing neural networks with dynamically defined neighborhood for cluster analysis*. In: L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, J. M. Żurada (Eds.), *Artificial Intelligence and Soft Computing* (*Lecture Notes in Computer Science*, vol. 8468), Springer, Cham, Heidelberg, New York, Dordrecht, London, 2014, pp. 713 – 725.
51. Gorzałczany M. B., Piekoszewski J., Rudziński F.: *Microarray Leukemia gene data clustering by means of generalized self-organizing neural networks with evolving tree-like structures*. In: L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, J. M. Żurada (Eds.), *Artificial Intelligence and Soft Computing* (*Lecture Notes in Computer Science*, vol. 9119), Springer, Cham, Heidelberg, New York, Dordrecht, London, 2015, pp. 15 – 25.
52. Gorzałczany M. B., Rudziński F., Piekoszewski J.: *Generalized SOMs with splitting-merging tree-like structures for WWW-document clustering*. In: J. M. Alonso, H. Bustince, M. Reformat (Eds.), *2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (IFSA-EUSFLAT-15), Gijon, Spain, June 30, 2015* (Advances in Intelligent Systems Research, vol. 89), Atlantis Press, 2015, pp. 186 – 193.
53. Gorzałczany M. B., Rudziński F., Piekoszewski J.: *Gene expression data clustering using tree-like SOMs with evolving splitting-merging structures*. In Proc. of IEEE World Congress on Computational Intelligence, 25-29 July 2016, Vancouver, Canada, *submitted*.
54. Harrington C. A., Rosenow C., Retief J.: *Monitoring gene expression using DNA microarrays*. Current Opinion in Microbiology, vol. 3, no. 3, 2000, pp 285 – 291.
55. Hartigan J. A.: *Direct clustering of a data matrix*. Journal of the American Statistical Association, vol. 67, no. 337, 1972, pp. 123 – 129.
56. Hartigan J. A.: *Clustering Algorithms*. John Wiley & Sons, New York, 1975.
57. Hertz J., Krogh A., Palmer R. G.: *Wstęp do Teorii Obliczeń Neuronowych*. WNT, Warszawa, 1993.
58. Hołowiecki J. (Red.): *Hematologia Kliniczna*. Wydawnictwo Lekarskie PZWL, Warszawa, 2007.
59. Hopfield J. J.: *Neural networks and physical systems with emergent collective computational abilities*. In: Proceedings of the National Academy of Science of the united States, vol. 79, no. 8, 1982, pp. 2554 – 2558.
60. Höppner F., Klawonn F., Kruse R., Runkler T.: *Fuzzy Clusters Analysis*. John Wiley & Sons, Chichester, 1999.
61. Hussain S. F.: *Bi-clustering gene expression data using cosimilarity*. In: Proceedings of the 7th International Conference on Advanced Data Mining and Applications, vol. 1, 2011, pp. 190 – 200.
62. International Human Genome Sequencing Consortium: *Initial sequencing and analysis of the human genome*. Nature, vol. 409, 2001, pp. 860 – 921.

63. Ji L., Mock K., Tan K.: *Quick Hierarchical Biclustering on Microarray Gene Expression Data*. 6th IEEE Symposium on BioInformatics and BioEngineering (BIBE2006), 2006, pp. 110 – 120.
64. Kaufman L., Rousseeuw P. J.: *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 1990.
65. Klugar Y., Basri R., Chang J. T., Gerstein M.: *Spectral biclustering of microarray data: coclustering genes and conditions*. Genome Research, vol. 13, no. 4, 2003, pp. 703 – 716.
66. Kohonen T.: *Self-organized formation of topologically correct feature maps*. Biological Cybernetics, vol. 43, 1982, pp. 59 – 69.
67. Kohonen T.: *The Self-Organizing Maps*. Proceedings of the IEEE, vol. 78, no. 9, pp. 1464 – 1480.
68. Kohonen T.: *Self-Organizing Maps*. Springer-Verlag, Berlin, 2001.
69. Li H-D., Liang Y-Z., Xu Q-S., et al.: *Recipe for uncovering predictive genes using support vector machines based on model population analysis*. IEEE Transactions on Computational Biology and Bioinformatics, vol. 8, no. 6, 2011, pp. 1633 – 1641.
70. Liu X., Wang L.: *Computing the maximum similarity bi-clusters of gene expression data*. Bioinformatics, vol. 23, no. 1, 2007, pp. 50 – 56.
71. MacQueen J. B.: *Some methods for classification and analysis of multivariate observations*. In proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probabilit, Berkeley, University of California, 1967, vol. 1, pp. 281 – 297.
72. McGregor A., Hall M., Lorier P., Brunskill J.: *Flow clustering using machine learning techniques*. In: C. Barakat, I. Pratt (Eds.), *Passive and Active Network Measurement* (*Lecture Notes in Computer Science*, vol. 3015), Springer, Berlin, Heidelberg, New York, 2004, pp. 205 – 214.
73. Madeira S. C., Oliveira A. L.: *Biclustering algorithms for biological data analysis: A survey*. IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 1, no. 1, 2004, pp. 24 – 45.
74. Marques de Sa J.P.: *Pattern Recognition*. Concepts, Methods and Applications. Springer-Verlag, Berlin, 2001.
75. Marsland S., Shapiro J., Nehmzow U.: *A self-organising network that grows when required*. Neural Networks, vol. 15, no. 8 – 9, 2002, pp. 1041 – 1058.
76. Martinetez T.M., Schulten K.J.: *A "neural-gas" network learns topologies*. In: T. Kohonen, K. Makisara, O. Simula, J. Kangas (Eds.), *Artificial Neural Networks*, North-Holland, Amsterdam, 1991, pp. 397 – 402.
77. Massey L.: *Real-world text clustering with adaptive resonance theory neural networks*. In proc. of IEEE International Joint Conference on Neural Networks (IJCNN 2005), vol. 5, 2005, pp. 2748 – 2753.
78. Masters T.: *Sieci Neuronowe w Praktyce*. WNT, Warszawa, 1996
79. Maulik U., Bandyopadhyay S.: *Genetic algorithm based clustering technique*. Pattern Recognition, vol. 33, no. 9, 2000, pp. 1455 – 1465.
80. Michalewicz Z.: *Algorytmy Genetyczne + Struktury Danych = Programy Ewolucyjne*. WNT, Warszawa, 1996.
81. Mitra S., Data S., Perkins T., Michailidis G.: *Introduction to Machine Learning and Bioinformatics*. Chapman & Hall/CRC, 2007.

82. Mohamad M. S., Omatu S., Deris S., Yoshioka M.: *A modified binary particle swarm optimization for selecting the small subset of informative genes from gene expression data*. IEEE Transactions on Information Technology in Biomedicine, vol. 15, no. 6, 2011, pp. 813 – 822.
83. Murali T. M., Kasif S.: *Extracting conserved gene expression Motifs from gene expression data*. In proc. of the Pacific Symposium Biocomputing, vol. 8, 2003, pp. 77 – 88.
84. Murthy C. A., Chowdhury N.: *In search of optimal clusters using genetic algorithms*. Pattern Recognition Letters, vol. 17, no. 8, 1996, pp. 825 – 832.
85. National Human Genome Researche Institute: All About The Human Genome Project (HGP). (http://www.genome.gov/10001772).
86. Osowski S.: *Sieci Neuronowe w Ujęciu Algorytmicznym*. WNT, Warszawa, 1994.
87. Panda M., Patra M. R.: *Detecting network intrusions: A clustering approach*. IJSDIA International Journal of Secure Digital Information Age, vol. 1, no. 2, 2009.
88. Pedrycz W.: *Fuzzy Control and Fuzzy Systems*. Research Studies Press, John Wiley & Sons, New York. 1989.
89. Pedrycz W.: *Knowledge-Based Clustering*. John Wiley & Sons, Hoboken, New Jersey, 2005.
90. Piekoszewski J.: *An application of selected theoretical tools from the artificial intelligence area to the breast cancer recognition problem*, In proc. of the 10th European Conference of Young Researchers and Scientists, Žilina, 2013, pp. 91 – 94.
91. Piekoszewski J.: *A comparison of machine learning methods for cancer classification using microarray expression data*. In proc. of the 11th European Conference of Young Researchers and Scientists, Žilina, 2015, pp. 50 – 54.
92. Prudent Y., Ennaji A.: *An incremental growing neural gas learns topologies*. In proc. of the International Joint Conference on Neural Networks, Montreal (IJCNN2005), vol. 5, 2005, pp. 1211 – 1216.
93. Rand W.M.: *Objective criteria for the evaluation of clustering methods*. Journal of the American Statistical Association, vol. 66, 1971, pp. 846 – 850.
94. Rodrigues J. S., Almeida L. B.: *Improving the learning speed in topological maps of patterns*. In: The International Neural Network Society (INNS), the IEEE Neural Network Council Cooperating Societies, International Neural Network Conference (INNC), Paris, 1990, pp. 813 – 816.
95. Rodriguez-Baena D. S., Perez-Pulido A. J., Aguilara-Ruiz J. S.: *A biclustering algorithm for extracting bit-patterns from binary datasets*. Bioinformatics, vol. 27, no. 19, 2011, pp. 2738 – 2745.
96. Rudziński F., Głuszek A., Piekoszewski J.: *Optymalizacja lokalizacji centrów dystrybucyjnych z wykorzystaniem wielokryterialnego algorytmu genetycznego e-NSGA-II*. Logistyka, tom 6, 2014, str. 9223 – 9231.
97. Rudziński F., Piekoszewski J.: *The maintenance costs estimation of electrical lines with the use of interpretability-oriented genetic fuzzy rule-based systems*. Przegląd Elektrotechniczny, vol. 8, 2013, pp. 43 – 47.
98. Rutkowska D., Piliński M., Rutkowski L.: *Sieci Neuronowe, Algorytmy Genetyczne i Systemy Rozmyte*. PWN, Warszawa - Łódź, 1997.

99. Saber H. B., Elloumi M.: *DNA Microarray Data Analysis: A new survey on biclustering*. International Journal for Computational Biology, vol. 4, no. 1, 2015, pp. 21 – 37.
100. Sander J.: *Generalized Density Based Clustering for Spatial Data Mining*. Herbert Utz Verlag, 1999.
101. Segal E., Taskar B., Gasch A., Friedman N., Koller D.: *Rich probabilistic models for gene expression*. Bioinformatics, vol. 17, no.1, 2001, pp. 243 – 252.
102. Segal E., Taskar B., Gasch A., Friedman N., Koller D.: *Decomposing gene expression into cellular processes*. In Proc. of the Pacific Symposium on Biocomputing, vol. 8, 2003, pp. 89 – 100.
103. Sheng Q., Moreau Y., De Moor B.: *Biclustering micrarray data by gibbs sampling*. Bioinformatics, vol. 19, no. 2, 2003, pp. 196 – 205.
104. Shen F., Hasegawa O.: *An incremental network for on-line unsupervised classification and topology learning*. Neural Networks, vol. 19, no. 1, 2006, pp. 90 – 106.
105. Tadeusiewicz R.: *Sieci Neuronowe*. Akademicka Oficyna Wydawnicza PLJ, Warszawa, 1993.
106. Tan P. N., Steinbach M., Kumar V.: *Introduction to Data Mining*. Addison-Wesley, 2005.
107. Tanay A., Sharan R., Shamir R.: *Discovering statistically significant biclusters in gene expression data*. Bioinformatics, vol. 18, no. 1, 2002, pp. 136 – 144.
108. Tang C., Zhang L., Zhang A., Ramanathan M.: *Interrelated two-way clustering: An unsupervised approach for gene expression data analysis*. In proc. of the 2nd IEEE International Symposium on Bioinformatics and Bioengineering (BIBE 2001), 2001, pp. 41 – 48.
109. Tang C., Zhang A., Pei J.: *Mining phenotypes and informative genes from gene expression data*. In proc. of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD '03), Washington DC, USA, 2003, pp. 655 – 600.
110. Tavazoie S., Hughes, D., Campbell M. J., et al.: *Systematic determination of genetic network architecture*. Nature Genetics, vol. 22, no. 3, 1999, pp. 281 – 285.
111. Tian D. X., Liu Y. H., Shi J. R.: *Dynamic clustering algorithm based on adaptive resonance theory*. In: G. R. Liu, V. B. C. Tan, X. Han (Eds.), *Computational Methods*, Springer, 2006, pp. 1229 – 1248.
112. Tibshirani R., Hastie T., Eisen M., et al.: *Clustering methods for the analysis of DNA microarray data*. Technical report, Department of Health Research and Policy, Department of Genetics and Department of Biochemistry, Stanford University, 1999.
113. Theodoridis S., Koutroumbas K.: *Pattern Recognition*. Academic Press, London, 1999.
114. Ultsch A.: *U*-Matrix: A tool to visualize clusters in a high dimensional data*. Technical Report, vol. 36, Dept. of Mathematics and Computer Science, University of Marburg, Germany, 2003, pp. 1 – 12.
115. Ultsch A.: *Clustering with SOM: U*C*. In proc. of the Workshop on Self-Organizing Maps, Paris, France, 2005, pp. 75 – 82.
116. Walewski J., Borawska A.: *Nowotwory Układu Chłonnego*. Centrum Medycznego Kształcenia Podyplomowego, Warszawa, 2011.

117. Wang H., Wang W., Yang J., Yu S. J.: *Clustering by pattern similarity in large data sets*. In proc. of the 2002 ACM SIGMOD International Conference on Management of Data, 2002, pp. 394 – 405.
118. Vega-Pons S., Ruiz-Shulcloper J.: *A survey of clustering ensemble algorithms*. International Journal of Pattern Recognition and Artificial Intelligence, vol. 25, no. 3, 2011, pp. 337 – 372.
119. Venter J.C., Adams M.D., Myers E.W.: *The sequence of the human genome*. Science, vol. 291, no. 5507, 2001, pp. 1304 – 1351.
120. Xing E. P., Karp R. M.: *Cliff: Clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts*. Bioinformatics, vol. 17, no. 1, 2001, pp. 306 – 315.
121. Xu R., Wunsch D. C.: *Bartmap: A viable structure for biclustering*. Neural Network, vol. 24, no. 7, 2011, pp. 709 – 716.
122. Yang J., Wang W., Wang H., Yu P.: *Enhanced biclustering on expression data*. In proc. of the 3rd IEEE Conference on Bioinformatics and Bioengineering, 2003, pp. 321 – 327.
123. Zadeh L. A.: *Fuzzy sets*, Information and Control, vol. 8, 1965, pp. 338 – 353.
124. Zimmermann H.-J., Zysno P.: *Quantifying vagueness in decision models*. European Journal of Operational Research, vol. 22, 1985, pp. 148–158.
125. Żurada J., Barski M., Jędruch W.: *Sztuczne Sieci Neuronowe*. Wydawnictwo naukowe PWN, Warszawa, 1996.

# Dodatek A. Charakterystyka wybranych zbiorów danych z repozytorium Uniwersytetu Kalifornijskiego w Irvine

## A.1. Zbiór *Congressional Voting Records*

Autor zbioru: Jeff Schlimmer (Jeffrey.Schlimmer@a.gp.cs.cmu.edu)

Źródło: Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL: Congressional Quarterly Inc. Washington, D.C., 1985.

Data utworzenia: 27.04.1987

Nazwa archiwum: ftp://ftp.ics.uci.edu/pub/machine-learning-databases/voting-records/

Liczba rekordów: 435 rekordów podzielonych na 2 klasy (etykiety klas: *democrat* oraz *republican*) – patrz tabela A.1.1

Liczba wykorzystywanych atrybutów wejściowych: 16 (wszystkie atrybuty są atrybutami symbolicznymi o nazwach, odpowiednio, *handicapped-infants*, *water-project-cost-sharing*, *adoption-of-the-budget-resolution*, *physician-fee-freeze*, *el-salvador-aid*, *religious-groups-in-schools*, *anti-satellite-test-ban*, *aid-to-nicaraguan-contras*, *mx-missile*, *immigration*, *synfuels-corporation-cutback*, *education-spending*, *superfund-right-to-sue*, *crime*, *duty-free-exports* oraz *export-administration-act-south-africa*)

Tabela A.1.1. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Congressional Voting Records Database*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *democrat* | 267 |
| *republican* | 168 |
| RAZEM: | 435 |

## A.2. Zbiór *Breast Cancer Wisconsin* (*Diagnostic*)

Autor zbioru: Dr. William H. Wolberg, General Surgery Dept., University of Wisconsin, Clinical Sciences Center, Madison, WI 53792 (wolberg@eagle.surgery.wisc.edu),

W. Nick Street, Computer Sciences Dept., University of Wisconsin, 1210 West Dayton St., Madison, WI 53706 (street@cs.wisc.edu 608-262-6619)

Olvi L. Mangasarian, Computer Sciences Dept., University of Wisconsin, 1210 West Dayton St., Madison, WI 53706 (olvi@cs.wisc.edu)

Źródło zbioru: W.N. Street, W.H. Wolberg and O.L. Mangasarian Nuclear feature extraction for breast tumor diagnosis. IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology, volume 1905, pages 861-870, San Jose, CA, 1993.

Data utworzenia: 11.1995

Nazwa archiwum: ftp://ftp.ics.uci.edu/pub/machine-learning-databases/breast-cancer-wisconsin/

Liczba rekordów: 569 rekordy podzielone na 2 klasy (nazwy klas: *malignant* oraz *benign*) – patrz tabela A.1.2

Liczba wykorzystywanych atrybutów wejściowych: 30 (wszystkie atrybuty są atrybutami numerycznymi o nazwach, odpowiednio, *radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symetry* oraz *fractal dimension*)

Tabela A.1.2. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Breast Cancer Wisconsin Diagnostic*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *malignant* | 212 |
| *benign* | 357 |
| RAZEM: | 569 |

## A.3. Zbiór *Wine*

Autor zbioru: Stefan Aeberhard, email: stefan@coral.cs.jcu.edu.au

Źródło zbioru: Forina, M. et al, PARVUS - An Extendible Package for Data Exploration, Classification and Correlation. Institute of Pharmaceutical Food Analysis and Technologies, Via Brigata Salerno, 16147 Genoa, Italy.

Data utworzenia: 21.09.1998

Nazwa archiwum: ftp://ftp.ics.uci.edu/pub/machine-learning-databases/wine/

Liczba rekordów: 178 rekordów podzielonych na 3 klasy (nazwy klas: *class1, class2* oraz *class3*) – patrz tabela A.1.3

Liczba wykorzystywanych atrybutów wejściowych: 13 (wszystkie atrybuty są atrybutami numerycznymi o nazwach, odpowiednio, *alcohol, malic acid, ash, alcalinity of ash, magnesium, total phenols, flavanoids, nonflavanoid phenols, proanthocyanis, color intensity, hue, OD280/OD315 of diluted wines* oraz *proline*)

Tabela A.1.3. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Wine*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *Class1* | 59 |
| *Class2* | 71 |
| *Class3* | 48 |
| RAZEM: | 178 |

# Dodatek B. Charakterystyka wybranych zbiorów danych opisujących ekspresję genów z repozytorium Uniwersytetu Shenzhen w Chinach

## B.1. Zbiór danych *Leukemia*

Autorzy zbioru: Todd R. Golub (golub@broad.mit.edu), i inni (patrz źródło poniżej).

Źródło: *Molecular classification of cancer: Class discovery and class prediction by gene expression*. Science, vol. 286, 1999, pp. 531 – 537.

Data utworzenia: 14.10.1999

Nazwa archiwum: Broad Institute Cancer Program Legacy Publications Resources (http://www.broadinstitute.org/cgi-bin/cancer/)

Liczba rekordów: 72 rekordy podzielone na 2 klasy (etykiety klas: *ALL* (ang. *Acute Lymphoblastic Leukemia*) oraz *AML* (ang. *Acute Myelogenous Leukemia*)) – patrz tabela B.1.1

Liczba atrybutów: 7129 (liczby rzeczywiste), przy czym w eksperymentach w rozdziale 5 wykorzystano 3571 wybranych atrybutów, reprezentujących niezaszumione poziomy ekspresji unikatowych (niepowtarzających się) genów.

Tabela B.1.1. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Leukemia*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *ALL* | 47 |
| *AML* | 25 |
| RAZEM: | 72 |

Tabela B.1.2. Zestawienie informacji o wybranych genach ze zbioru danych *Leukemia* (poziomy ekspresji tych genów prezentowane są na rys. 5.4)

| Lp. | Identyfikator genu | Nazwa genu |
|---|---|---|
| 1 | D21063_at | Minichromosome maintenance complex component 2 |
| 2 | U05340_at | Cell division cycle 20 |
| 3 | X14850_at | H2A histone family, member X |
| 4 | X54942_at | CDC28 protein kinase regulatory subunit 2 |
| 5 | D38073_at | Minichromosome maintenance complex component 3 |
| 6 | HG3494-HT3688_at | CCAAT/enhancer binding protein (C/EBP), beta |
| 7 | M20681_at | Solute carrier family 2 (Facilitated glucose transporter), member 3 |
| 8 | M57710_at | Lectin, galactoside-binding, soluble, 3 |
| 9 | U02020_at | Nicotinamide phosphoribosyltransferase |
| 10 | X06614_at | Retinoic acid receptor, alpha |
| 11 | Z11697_at | CD83 antigen (Activated B lymphocytes, immuno-globulin superfamily) |
| 12 | M23178_s_at | Chemokine (C-C motif) ligand 3-like 3 |
| 13 | M28130_rna1_s_at | Interleukin 8 |
| 14 | Y00787_s_at | Chemokine (C-X-C motif) ligand 8 |
| 15 | D88270_at | Pre-B lymphocyte 1 |
| 16 | K01911_at | Neuropeptide Y |
| 17 | M38690_at | CD9 molecule |
| 18 | M74719_at | Transcriptor factor 4 |
| 19 | M89957_at | CD79b molecule, immunoglobulin-associated beta |
| 20 | X82240_rna1_at | T-cell leukemia/lymphoma 1A |
| 21 | L33930_s_at | CD24 molecule |
| 22 | D83920_at | Ficolin (Collagen/fibrinogen domain containing) 1 |
| 23 | D88422_at | Cystatin A (stefin A) |
| 24 | K01396_at | Serpin peptidase inhibitor, clade A (alpha-1 antipro-teinase, antitrypsin), member 1 |
| 25 | M27891_at | Cystatin C |
| 26 | M33195_at | Fc fragment of IgE, high affinity I, receptor for: gamma polypeptide |
| 27 | M84526_at | Complement factor D (Adipsin) |

| | | |
|---|---|---|
| 28 | X05908_at | Annexin A1 |
| 29 | X61587_at | Ras homolog family member G |
| 30 | M21119_s_at | Lysozyme |
| 31 | M83667_rna1_a_at | CCAAT/Enhancer binding protein (C/EBP), delta |
| 32 | J04615_at | Small nuclear ribonucleoprotein polypeptide N; SNRPN upstream reading frame |
| 33 | M11722_at | DNA nucleotidylexotransferase |
| 34 | M29696_at | Interleukin 7 receptor |
| 35 | U05259_rna1_at | CD79a molecule, immunoglobulin-associated alpha |
| 36 | X58529_at | Immunoglobulin heavy constant gamma 1 (G1m marker) |
| 37 | Z69881_at | ATPase, Ca++ transporting, ubiquitous |
| 38 | M84371_rna1_s_at | CD19 molecule |
| 39 | X97267_rna1_s_at | Protein tyrosine phosphatase, receptor type, C-associated protein |
| 40 | M31523_at | Transcription factor 3 |
| 41 | M23197_at | CD33 molecule |
| 42 | M59820_at | Colony stimulating factor 3 receptor (Granulocyte) |
| 43 | U46499_at | Microsomal glutathione S-transferase 1 |
| 44 | X07743_at | Pleckstrin |
| 45 | X52056_at | Spi-1 proto-oncogene |
| 46 | X55668_at | Proteinase 3 |
| 47 | M27783_s_at | Elastase, neutrophil expressed |
| 48 | M20203_s_at | Elastase, beutrophil expressed |
| 49 | M83652_s_at | Complement factor properdin |
| 50 | X16546_at | Nonsecretory ribonuclease precursor (Ribonuclease US) |
| 51 | D86983_at | Peroxidasin |
| 52 | U10485_at | Lymphoid-restricted membrane protein |
| 53 | U60115_at | Four and a half LIM domains 1 |
| 54 | U65932_at | Extracellular matrix protein 1 |
| 55 | U97105_at | Dihydropyrimidinase-like 2 |
| 56 | X59350_at | CD22 molecule |
| 57 | X76648_at | Glutareodoxin (Thioltransferase) |
| 58 | Z14982_rna1_at | Protesome (prosome, macropain) subunit, beta type |
| 59 | M63838_s_at | Interferon, gamma-inducible protein 16 |

## B.2. Zbiór danych *Mixed Lineage Leukemia* (*MLL*)

Autor zbioru: Scott Armstrong (Scott.Armstrong@childrens.harvard.edu), i inni (patrz źródło poniżej).

Źródło zbioru: *MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia*. Nature Genetics, vol. 30, 2002, pp. 41 – 47.

Data utworzenia: 02.12.2001

Nazwa archiwum: Broad Institute Cancer Program Legacy Publications Resources

(http://www.broadinstitute.org/cgi-bin/cancer/)

Liczba rekordów: 72 rekordy podzielone na 3 klasy (etykiety klas: *ALL* (ang. *Acute Lymphoblastic Leukemia*), *AML* (ang. *Acute Myelogenous Leukemia*) oraz *MLL* (ang. *Mixed Lineage Leukemia*)) – patrz tabela B.2.1

Liczba atrybutów: 12533 (liczby rzeczywiste), przy czym w eksperymentach w rozdziale 5 wykorzystano 2474 wybranych atrybutów, reprezentujących niezaszumione poziomy ekspresji unikatowych (niepowtarzających się) genów.

Tabela B.2.1. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Mixed Lineage Leukemia*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *ALL* | 24 |
| *AML* | 28 |
| *MLL* | 20 |
| RAZEM: | 72 |

Tabela B.2.2. Zestawienie informacji o wybranych genach ze zbioru danych *Mixed Lineage Leukemia* (poziomy ekspresji tych genów prezentowane są na rys. 5.9)

| Lp. | Identyfikator genu | Nazwa genu |
|---|---|---|
| 1 | 39318_at | T-cell leukemia/lymphoma 1A |
| 2 | 40103_at | Hypothetical protein LOC1000129652 |

| | | |
|---|---|---|
| 3 | 40875_s_at | Small nuclear ribonucleoprotein 70kDa (U1) |
| 4 | 41164_at | Immunoglobulin heavy constant 1 (G1m marker) |
| 5 | 41165_g_at | Immunoglobulin heavy constant 1 (G1m marker) |
| 6 | 41213_at | Peroxiredoxin 1 |
| 7 | 41220_at | Septin 9 |
| 8 | 34891_at | Dynein, light chain, LC8-type 1 |
| 9 | 35307_at | GDP dissociation inhibitor 2 |
| 10 | 36122_at | Mitochondrial RNase P subunit 3 |
| 11 | 36571_at | Topoisomerase (DNA) II beta 180kDa |
| 12 | 1840_g_at | RAN, member RAS oncogene family |
| 13 | 1795_g_at | Cyclin D3 |
| 14 | 1179_at | Heat shock 70kDa protein 8 |
| 15 | 33963_at | Azurocidin 1 |
| 16 | 36766_at | Ribonuclease, RNase A family, 2 |
| 17 | 37096_at | Elastase, neutrophil expressed |
| 18 | 37105_at | Cathepsin G |
| 19 | 41471_at | S100 calcium binding protein A9 |
| 20 | 33284_at | Myeloperoxidase |
| 21 | 38363_at | TYRO protein tyrosine kinase binding protein |
| 22 | 32227_at | Serglycin |
| 23 | 37403_at | Annexin A1 |
| 24 | 40282_s_at | Complement factor D (adipsin) |
| 25 | 691_g_at | Prolyn 4-hydroxylase, beta polypeptide |
| 26 | 679_at | Cathepsin G |
| 27 | 40436_g_at | Solute carrier famil 25 (adenine nucleotide translocator) |
| 28 | 41206_r_at | Cytochrome c oxidase subunit Via polypeptide 1 |
| 29 | 33819_at | Lactate dehydrogenase B |
| 30 | 33820_g_at | Heat shock 70kDa protein 8 |
| 31 | 36629_at | TSC22 domain family, member 3 |
| 32 | 37720_at | Heat shock 60kDa protein 1 (chaperonin) pseudogene 5 |
| 33 | 1180_g_at | Heat shock 70kDa protein 8 |
| 34 | 911_s_at | Calmodulin 3 (phosphorylase kinase, delta) |
| 35 | 649_s_at | Chemokine (C-X-C motif) receptor 4 |

## B.3. Zbiór danych *Colon*

Autor zbioru: Uri Alon (urialon@weizmann.ac.il), i inni (patrz źródło poniżej).

Źródło: *Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays*. In: Proceedings of the National Academy of Science of the United States, vol. 96, no. 12, 1999, pp. 6745 – 6750.

Data utworzenia: 1999

Nazwa archiwum: Princeton University gene expression project

(http://microarray.princeton.edu/oncology)

Liczba rekordów: 62 rekordy podzielone na 2 klasy (etykiety klas: *C* (*Colon*) oraz *N* (*Normal*)) – patrz tabela B.3.1

Liczba atrybutów: 2000 (liczby rzeczywiste), przy czym w eksperymentach w rozdziale 6 wykorzystano 1096 wybranych atrybutów, reprezentujących niezaszumione poziomy ekspresji unikatowych (niepowtarzających się) genów.

Tabela B.3.1. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Colon*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *C* | 40 |
| *N* | 22 |
| RAZEM: | 62 |

Tabela B.3.2. Zestawienie informacji o wybranych genach ze zbioru danych *Colon* (poziomy ekspresji tych genów prezentowane są na rys. 6.4)

| Lp. | Identyfikator genu | Nazwa genu |
|---|---|---|
| 1 | X74295 | Integrin, alpha 7 |
| 2 | M26683 | Chemokine (C-C motif) ligand 2 |
| 3 | X86693 | SPARC-like 1 (hevin) |
| 4 | M64110 | Caldesmon 1 |
| 5 | M86400 | Tyrosine 3-monooxygenase |
| 6 | X14954 | High mobility group AT-hook 1 |

| | | |
|---|---|---|
| 7 | R15447 | Calnexin |
| 8 | H40095 | Solute carrier family 2 |
| 9 | R08183 | Heat shock 10kDa protein 1 |
| 10 | H40269 | Phosphoglycerate kinase 1, pseudogene 1 |
| 11 | D31716 | Kruppel-like factor 9 |
| 12 | U14394 | TIMP matallopaptidase inhibitor 3 |
| 13 | R60877 | Zinc finger E-box binding momeobox 1 |
| 14 | T94350 | Peripheral myelin protein 22 |
| 15 | L07648 | MAX interactor 1, dimerization protein |
| 16 | X15183 | Heat shock protein 90kDa alpha (Cytosolic), class A member 1 |
| 17 | Y00345 | Poly(A) binding protein, cytoplasmic 1 |
| 18 | T47377 | S100 calcium binding protein P |
| 19 | M27190 | Regenerating islet-derived 1 alpha |
| 20 | R46753 | Cyclin-dependent kinase inhibitor 1A (P21, cip1) |
| 21 | M83667 | CCAAT/Enhancer binding protein (C/EBP), delta |
| 22 | H43887 | Complement factor D (Adipsin) |
| 23 | R67358 | Dual specificity phosphatase 1 |
| 24 | M92843 | ZFP36 ring factor protein |
| 25 | H77597 | Matallothionein 1G |
| 26 | T60778 | Matrix Gla protein |
| 27 | V00523 | Major histocompatibility complex |
| 28 | X17042 | Serglycin |
| 29 | K01144 | CD74 molecule, major histocompatibility complex |
| 30 | Z24727 | Tropomyosin 1 (alpha) |
| 31 | T60155 | Actin, alpha 2, smooth muscle, aorta |
| 32 | T92451 | Tropomyosin 2 (beta) |
| 33 | T48904 | Heat shock 27kDa protein 1 |

## B.4. Zbiór danych *Lymphoma*

Autor zbioru: Ash A. Alizadeh (arasha@Stanford.edu), i inni (patrz źródło poniżej).

Źródło zbioru: *Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling*. Nature, vol. 403, no. 6769, 2000, pp. 503 – 511.

Data utworzenia: 03.02.2000

Nazwa archiwum: Lymphoma/Leukemia Molecular Profiling Project

(http://llmpp.nih.gov/lymphoma/)

Liczba rekordów: 62 rekordy podzielone na 3 klasy (etykiety klas: *DLBCL* (ang. *Diffuse Large B-cell Lymphoma*), *CLL* (ang. *Chronic Lymphocytic Leukemia*) oraz *FL* (*Follicular Lymphoma*)) – patrz tabela B.4.1

Liczba atrybutów: 4026 (liczby rzeczywiste) reprezentujących niezaszumione poziomy ekspresji unikatowych (niepowtarzających się) genów.

Tabela B.4.1. Liczba rekordów przyporządkowanych do poszczególnych klas w zbiorze *Lymphoma*

| Etykieta klasy | Liczba rekordów należących do klasy |
|---|---|
| *DLBCL* | 42 |
| *CLL* | 11 |
| *FL* | 9 |
| RAZEM: | 62 |

Tabela B.4.2. Zestawienie informacji o wybranych genach ze zbioru danych *Lymphoma* (poziomy ekspresji tych genów prezentowane są na rys. 7.4)

| Lp. | Pierwotny identyfikator genu | Identyfikator genu w formacie *Ensembl* | Nazwa genu |
|---|---|---|---|
| 1 | MMP9 | ENSG00000100985 | Matrix metallopeptidase 9 |
| 2 | CSF1 | ENSG00000182578 | Colony stimulating factor 1 (Macrophage) |
| 3 | CTSB | ENSG00000164733 | Cathepsin B |
| 4 | FN1 | ENSG00000115414 | Fibronectin 1 |

| | | | |
|---|---|---|---|
| 5 | OSF2 | ENSG00000133110 | Periostin, osteoblast specific factor |
| 6 | MMP2 | ENSG00000087245 | Matrix metallopeptidase 2 |
| 7 | SPARC | ENSG00000113140 | Secreted protein, acidic, cysteine-rich (Osteonectin) |
| 8 | CCND2 | ENSG00000118971 | Cyclin D2 |
| 9 | TIMP3 | ENSG00000100234 | TIMP metallopeptidase inhibitor 3 |
| 10 | BCA1 | ENSG00000156234 | Chemokine (C-X-C motif) ligand 13 |
| 11 | CYP27A1 | ENSG00000135929 | Cytochrome P450, family 27, subfamily A, polypeptide 1 |
| 12 | LMO2 | ENSG00000135363 | LIM domain only 2 (Rhombotin-like 1) |
| 13 | CD79B | ENSG00000007312 | CD79b molecule, immunoglobulin-associated beta |
| 14 | BMP7 | ENSG00000101144 | Bone morphogenetic protein 7 |
| 15 | EML1 | ENSG00000066629 | Echinoderm microtubule associated protein like 1 |
| 16 | F11 | ENSG00000088926 | Coagulation factor XI |
| 17 | SNRPN | ENSG00000128739 | Small nuclear ribonucleprotein polypeptide N |
| 18 | RASA1 | ENSG00000145715 | RAS P21 protein activator (GTPase activating protein) 1 |
| 19 | SLA | ENSG00000155926 | Src-like-adaptor |
| 20 | CD1C | ENSG00000158481 | CD1c molecule |
| 21 | KLRK1 | ENSG00000213809 | Killer cell lectin-like receptor subfamily K, member 1 |
| 22 | PTPN12 | ENSG00000127947 | Protein tyrosine phosphatase, nonreceptor type 12 |
| 23 | WNT3 | ENSG00000108379 | Wingless-type MMTV integration site family, member 3 |
| 24 | BMPR2 | ENSG00000204217 | Bone morphogenetiv protein kinase receptor, type II (serine/threonine kinase) |