

Zadaniem jest wykorzystanie grafu AANG do określenia możliwie niewielkiej ilości takich reprezentatywnych kombinacji o możliwie wysokim stopniu dyskryminatywności dla dowolnego zbioru uczącego. Ze względu na to, iż stopień dyskryminatywności każdej kombinacji nie może być większy niż minimalny stopień dyskryminacji każdego z reprezentowanych przez kombinację wzorców, warto rozpocząć proces budowy kombinacji od wzorców najbardziej skorelowanych z wzorcami innych klas. To właśnie te wzorce będą stanowić „wąskie gardło” i będzie je najtrudniej dołączyć do jakiegokolwiek kombinacji i uzyskać jeszcze możliwie duży jej stopień dyskryminatywności i dużą jej reprezentatywność. Pierwszym krokiem budowy klasyfikatora ASONN jest więc konwersja zbioru uczącego do postaci struktury AGDS, a następnie jej transformacja do postaci grafu AANG. Graf AANG składać się będzie z neuronów ASN reprezentujących zdefiniowane w zbiorze uczącym klasy i wzorce uczące definiujące te klasy oraz neurony receptoryczne ASRN reprezentujące wartości parametrów definiujące te wzorce uczące. Wszystkie te neurony są odpowiednio połączone (ADEF), a połączenia ważone w zależności od tego, co definiują (wzorzec czy klasę), oraz dodatkowo połączone są neurony receptoryczne reprezentujące porządkowalne wartości podobne (ASIM) (rys. 7.4, 7.5, 7.6, 7.9). Sposób wyznaczenia wag dla tych połączeń zostanie opisany w kolejnych podrozdziałach. Taki graf neuronowy bez uczenia staje się od razu prostym klasyfikatorem, który jak pokazują wyniki klasyfikacji zaprezentowane w tabelach 7.2-5 są stosunkowo dobre, lecz odbywa się to kosztem dużej struktury takiego grafu, którą dla zadań klasyfikacji można zoptymalizować. Graf AANG zostanie jednak wykorzystany do szybkiego wyznaczania korelacji pomiędzy wzorcami oraz do znalezienia dyskryminatywnych i zarazem reprezentatywnych kombinacji przedziałów i podzbiorów wartości parametrów wzorców uczących reprezentujących hiperprostopadłością (semassele) zawierające wzorce poszczególnych klas. Te kombinacje zostaną następnie przekształcone na nowe neurony ASN i będą stanowiły trzon decyzyjny klasyfikatora ASONN.

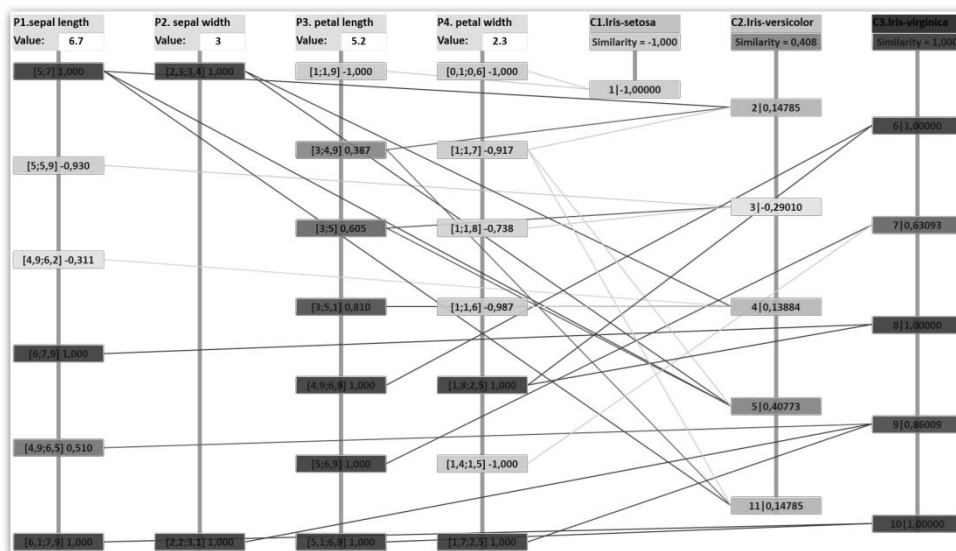
Zastosowano tutaj następujące nazewnictwo:

- Asocjacyjne neurony receptoryczne reprezentujące wartości parametrów będą nazwane krótko **neuronami wartości (VN – value neuron)**.
- Asocjacyjne neurony reprezentujące wzorce zdefiniowane przy pomocy odpowiednich połączeń z neuronami VN będą nazywane krótko **neuronami wzorców (SN – sample neuron)**.
- Asocjacyjne neurony reprezentujące klasy określone w zbiorze uczącym zdefiniowane przy pomocy połączeń z wszystkimi neuronami wzorców SN, które do tej klasy należą, będą nazywane krótko **neuronami klas (CN – class neuron)**.

Warto zaznaczyć, iż w grafach AANG nie występują ani duplikaty wartości dla poszczególnych parametrów, ani duplikaty wzorców, gdyż są one wyeliminowane już na etapie budowy tych grafów, co decyduje również o oszczędności takiej reprezentacji oraz umożliwia bardzo szybkie wyznaczenie wszystkich korelacji przy stałej złożoności obliczeniowej. Proces tworzenia kombinacji dokonujemy osobno dla każdej klasy poprzez aktywację neuronu reprezentującego daną klasę w grafie AANG. Wzorce uczące są zwykle wcześniej zebrane i jakoś uporządkowane, a więc można tylko się domyślać, czy ich uporządkowanie ma jakieś znaczenie. Zwykle wzorce uczące nie posiadają też żadnej flagi ich istotności czy też powtarzalności, a więc reprezentujące je neurony SN mają zwykle takie same wagi połączeń z neuronami klas CN. Trzeba więc przyjąć pewne dodatkowe założenia i dokonać uproszczeń wynikających z ograniczonej ilości dostępnych informacji na temat danych uczących. Dokonanie aktywacji kolejnych neuronów SN, reprezentujących wzorce danej klasy (połączone z CN), dostarcza informacji o stopniu ich skorelowania z wzorcami innych klas, jak pokazano to na rysunkach 7.6-8. Aktywacja neuronu wzorca SN_k , powoduje aktywację połączonych (ADEF) z nim neuronów VN_i reprezentujących wartości parametrów, definiujących wzorzec reprezentowany przez neuron SN_k . Neurony wartości parametrów VN_i są połączone również z wszystkimi innymi neuronami wzorców SN_n , które takie wartości parametrów posiadają. Te inne wzorce mogą należeć do tej samej klasy, co aktywowany wcześniej neuron SN_k , lub też mogą należeć do innej klasy. Jeśli te wzorce należą do tej samej klasy, wtedy występuje **pozytywna korelacja** z punktu widzenia reprezentatywności potencjalnej kombinacji. Jeśli zaś te wzorce należą do innej klasy, wtedy występuje **negatywna korelacja** z punktu widzenia dyskryminatywności danej kombinacji. Zwykle takie wzorce należą częściowo do tej samej klasy, a częściowo do innych klas. Pozytywne korelacje będą też nazywane **korelacjami własnymi**, a negatywne **korelacjami obcymi**. Żeby znaleźć wzorce najbardziej negatywnie skorelowane z innymi klasami, wtedy każdy aktywowany neuron wartości VN_i przez neuron SN_k , pobudza wszystkie połączone z nim neurony wzorców SN_n , za wyjątkiem wzorca SN_k , który jest przez chwilę w stanie refrakcji. W wyniku tego każdy połączony neuron SN_n zostanie pobudzony w stopniu odpowiadającym jego stopniu korelacji z neuronem SN_k . Ponadto neurony VN_i mogą również pobudzać sąsiednie (bliskie co do wartości) neurony VN_m dzięki łączącym je połączeniom ASIM, a te następnie mogą z odpowiednio mniejszą siłą pobudzać inne lub te same neurony SN_n . Im więcej identycznych lub bliskich wartości parametrów posiadają neurony SN_n z neuronem SN_k , tym bardziej zostaną pobudzone (rys. 7.6). Aktywowany neuron klasy CN połączony z neuronem SN_k blokuje przez aktywację i refrakcję wszystkie neurony z nim powiązane. Wobec tego inne neurony SN_n , niepołączone z aktywowanym neuronem klasy CN, przesyłają wstecz do wszystkich skojarzonych z nimi neuronów wartości VN_i informację o swoim stopniu pobudzenia (tzn. stopniu korelacji obcej z wzorcem SN_k), a neurony VN_i , definiujące wzorzec źródłowy SN_k , zbierają te informacje od wszystkich neuronów SN_n należących do innych klas i przesyłają dalej

stosowną informację o tym wstecz do neuronu wzorca SN_k , który dzięki temu uzyskuje informację o swoim stopniu skorelowania ze wszystkimi wzorcami wszystkich innych klas. Ze względu na zbieranie i przesyłanie informacji przez neurony VN_i o stopniu pobudzenia neuronów SN_n model obliczeniowy ASONNów jest hybrydowy z punktu widzenia AASów i asocjacyjnych obliczeń. Obliczenie tego dla pojedynczego wzorca danej klasy odbywa się ze stałą złożonością obliczeniową w przypadku realizacji równoległej. Taki proces obliczania korelacji z wzorcami innych klas trzeba powtórzyć kolejno dla każdego neuronu wzorca reprezentującego daną klasę po to, by wyznaczyć maksymalnie negatywnie skorelowany neuron SN (rys. 7.8). W jednowątkowym modelu symulacyjnym wyznaczenie najbardziej negatywnie skorelowanych wzorców wszystkich klas odbywa się przy złożoności liniowej.

Następnie dla najbardziej negatywnie skorelowanego wzorca dyskryminowanej klasy, który na razie nie jest połączony z żadnym neuronem kombinacji, uruchamiany jest proces tworzenia kombinacji. W tym celu tworzony jest nowy neuron kombinacji KN w grafie AANG. Neuron najbardziej negatywnie skorelowany stanie się neuronem zarodkowym $germSN$ dla neuronu KN , z którym zostanie asocjacyjnie powiązany (ADEF). **Neuron zarodkowy $germSN$** to neuron wzorca, który inicjuje powstanie neuronu kombinacji KN i determinuje dalsze jego połączenia z neuronami tej samej klasy CN , którą reprezentuje, biorąc pod uwagę pożądany wynikowy stopień dyskryminatywności całej kombinacji. Neuron kombinacji KN jest ponadto łączony z receptorycznymi neuronami przedziałów wartości RN_i lub receptorycznymi neuronami podzbiorów wartości UN_i , które są również tworzone dla poszczególnych parametrów definiujących wzorec $germSN$ w trakcie tworzenia neuronu KN (rys. 7.12, 7.13, 7.15, 7.16). Powstanie neuronu RN_i lub UN_i uzależnione jest od tego, czy określony parametr reprezentuje wartości porządkowalne. Receptoryczne neurony przedziałów wartości RN_i będą tworzone dla wartości poszczególnych parametrów porządkowalnych i będą określały pewien rozmyty przedział wartości, reprezentowany przez VN_y , na które ten neuron będzie reagował pobudzeniem lub aktywacją zgodnie z funkcją ściętego kapelusza Gaussa (7.17) (rys. 7.15). Natomiast receptoryczne neurony podzbiorów wartości UN_i będą tworzone dla pozostałych wartości poszczególnych parametrów (np. symbolicznych) i będą określały pewien podzbiór wartości, reprezentowany przez VN_y , na które będzie reagował aktywacją. W trakcie genezy neuronów RN_i lub UN_i są one łączone z neuronami wartości VN_i , które reprezentują. Następnie, aktywowany jest neuron kombinacji KN , aktywujący wszystkie powiązane z nim neurony przedziałów RN i neurony podzbiorów UN , które aktywują powiązane z nimi neurony wartości VN , pobudzające między innymi najmocniej pozytywnie skorelowane neurony wzorców SN oraz w zależności od swojego rodzaju mogą również aktywować sąsiednie neurony wartości VN o najbliższych wartościach. Zostają więc pobudzone lub aktywowane wszystkie neurony i neurony receptoryczne, które powinny mieć największy wpływ na plastyczność neuronów KN , RN_i i UN_i związanych ze zwiększaniem reprezentatywności tworzonej kombinacji. Neurony RN_i i UN_i są łączone w pierwszym rzędzie z neuronami wartości, które w największym stopniu zwiększają reprezentatywność kombinacji, która wzrasta z każdym nowo reprezentowanym wzorcem klasy definiowanej przez tą kombinację. Dołączanie kolejnych wzorców do kombinacji odbywa się poprzez nawiązywanie połączeń pomiędzy reprezentującymi je neuronami wzorców SN oraz neuronem kombinacji KN , jak również poprzez połączenie definiujących je neuronami wartości VN odpowiednio z neuronami RN_i lub UN_i . Określenie VN ów, które w największym stopniu zwiększają reprezentatywność neuronu kombinacji KN , dokonujemy poprzez pobudzenie stosownych SN ów $VNami$, które obecnie są powiązane już $RN-nami$ lub $UN-nami$ tego KNa . W wyniku tego stosowne SNy są pobudzone w stopniu odpowiadającym ich aktualnej korelacji z KN . Te najbardziej pobudzone i pozytywnie skorelowane SNy warto byłoby móc połączyć z KN poprzez dodanie pozostałych definiujących je VN ów. Na drodze stoją jednak dwie trudności. Jedną z nich jest konieczność dodawania połączeń VN ów z $RNami$ w kolejności wynikającej z ich porządku reprezentowanych przez nie wartości, gdyż inaczej by nie powstały stosowne przedziały. Drugą trudnością są pobudzone obce SNy , które są również w pewnym stopniu powiązane z $VNami$ tego KNa , lecz dla uzyskania odpowiedniego stopnia dyskryminacji nie może być za dużo definiujących je VN ów połączonych z $RNami$ i $UNami$ neuronu KN . Sposób rozwiązania tych trudności zostanie przedstawiony w kolejnych podrozdziałach. Trzeba więc kontrolować stopień pobudzenia wszystkich neuronów innych klas, żeby nie spaść poniżej pożądanego stopnia dyskryminacji. Dopiero wtedy dany neuron wartości VN może być powiązany z neuronami RN_i lub UN_i . W taki sposób przebiega w pewnym uproszczeniu tworzenie połączeń z neuronami RN_i i UN_i oraz tworzona jest **dyskryminatywno-reprezentatywna kombinacja** przedziałów i podzbiorów reprezentowana przez KN dla danego neuronu zarodkowego $germSN$. Taki hybrydowy proces tworzenia każdej kombinacji ma złożoność maksymalnie liniową przy realizacji równoległej. W związku z tym, iż kolejne kombinacje tworzone są głównie na bazie wzorców jeszcze niereprezentowanych w innych kombinacjach, łączna złożoność obliczeniowa tworzenia wszystkich kombinacji dla wszystkich klas jest również co najwyżej liniowa. Tworzenie dyskryminatywno-reprezentatywnych kombinacji ASONN ma charakter **pozytywnie redundantny**, więc często zdarza się, że część neuronów wzorców SN jest równocześnie połączona z kilkoma różnymi neuronami kombinacji KN reprezentującymi wzorce poszczególnych klas. Wpływa to pozytywnie na pewną, lecz ograniczoną niewrażliwość tego systemu klasyfikacji na możliwe uszkodzenia grafu oraz możliwości uogólniające tego klasyfikatora.



Rys. 7.2 Ewaluacja wzorca uczącego nr 146 ze zbioru Iris, definiującego klasę Virginica, w sieci ASONN, zbudowanej dla zbioru uczącego Iris poprzez wydzielenie z grafu AANG wyspecjalizowanego i zoptymalizowanego podgrafu ASONN przeznaczanego do klasyfikacji.

Informacje ukryte w grafie AANG można wydobyc poprzez wywoływanie skojarzeń, tzn. poprzez celowe pobudzenie i aktywacje różnych kombinacji neuronów, które pobudzają, hamują lub aktywują inne neurony lub neurony receptoryczne, które poprzez swoją zmianę stanu lub aktywację, mogą dostarczać pewnych informacji albo rozpocząć procesy plastyczności. W przypadku tworzenia klasyfikatora ASONN wykorzystano tylko część tych możliwości, np. tylko asocjacje definiujące ADEF oraz asocjacje podobieństwa ASIM. Nie wykorzystano natomiast asocjacji następstwa ASEQ i kontekstu ACON, gdyż zwykle kolejność wzorców uczących w zbiorach danych jest przypadkowa lub nie ma istotnego znaczenia z punktu widzenia klasyfikacji. Ponadto, dla przyspieszenia transformacji wzorców uczących z klasycznych tabel do postaci AGDS i AANG oraz zbierania części informacji dokonano pewnych uproszczeń i hybrydyzacji.

W następujących podrozdziałach zostaną szczegółowo przedstawione poszczególne etapy budowy klasyfikatora ASONN, rozpoczynając od konwersji zbioru uczącego na AGDS, następnie jego transformacji do grafu AANG, jego dalszej rozbudowy o wyspecjalizowane neurony dla potrzeb klasyfikacji, aż do wyodrębnienia elementów asocjacyjnego podgrafu klasyfikatora ASONN (rys. 7.2). Cały proces budowy klasyfikatora zostanie zilustrowany na przykładzie zbioru uczącego Iris z ML Repository, żeby zobrazować poszczególne etapy jego tworzenia, co ułatwi wyobrażenie sobie asocjacyjnych procesów, zachodzących w trakcie jego budowy. Zbiór Iris został wybrany z tego powodu, iż występują w nim różne korelacje pomiędzy wzorcami klas, więc jest ciekawy z punktu widzenia dyskryminacji, a zarazem na tyle mały, że można na nim zilustrować ten proces. Zostaną też przedstawione i porównane wyniki działania klasyfikatora ASONN z innymi wiodącymi na świecie klasyfikatorami na przykładzie różnych znanych zbiorów uczących z ML Repository, różnych ze względu na: wielkość, stopień korelacji wzorców, liczebność wzorców reprezentujących poszczególne klasy, ich duplikaty występujące w niektórych zbiorach oraz rodzaj i ilość parametrów wejściowych itd.

Tab. 7.1 Zbiór danych uczących Iris z ML Repository z zaznaczeniem duplikatów wzorców automatycznie ujawniających się podczas jego konwersji na strukturę AGDS.

Lp.	klasa	długość liścia	szerokość liścia	długość płatk	szerokość płatk	Lp.	klasa	długość liścia	szerokość liścia	długość płatk	szerokość płatk	Lp.	klasa	długość liścia	szerokość liścia	długość płatk	szerokość płatk
1	Iris Setosa	5,1	3,5	1,4	0,2	51	Iris Versicolor	7,0	3,2	4,7	1,4	101	Iris Virginica	6,3	3,3	6,0	2,5
2	Iris Setosa	4,9	3,0	1,4	0,2	52	Iris Versicolor	6,4	3,2	4,5	1,5	102	Iris Virginica	5,8	2,7	5,1	1,9
3	Iris Setosa	4,7	3,2	1,3	0,2	53	Iris Versicolor	6,9	3,1	4,9	1,5	103	Iris Virginica	7,1	3,0	5,9	2,1
4	Iris Setosa	4,6	3,1	1,5	0,2	54	Iris Versicolor	5,5	2,3	4,0	1,3	104	Iris Virginica	6,3	2,9	5,6	1,8
5	Iris Setosa	5,0	3,6	1,4	0,2	55	Iris Versicolor	6,5	2,8	4,6	1,5	105	Iris Virginica	6,5	3,0	5,8	2,2
6	Iris Setosa	5,4	3,9	1,7	0,4	56	Iris Versicolor	5,7	2,8	4,5	1,3	106	Iris Virginica	7,6	3,0	6,6	2,1
7	Iris Setosa	4,6	3,4	1,4	0,3	57	Iris Versicolor	6,3	3,3	4,7	1,6	107	Iris Virginica	4,9	2,5	4,5	1,7
8	Iris Setosa	5,0	3,4	1,5	0,2	58	Iris Versicolor	4,9	2,4	3,3	1,0	108	Iris Virginica	7,3	2,9	6,3	1,8
9	Iris Setosa	4,4	2,9	1,4	0,2	59	Iris Versicolor	6,6	2,9	4,6	1,3	109	Iris Virginica	6,7	2,5	5,8	1,8
10	Iris Setosa	4,9	3,1	1,5	0,1	60	Iris Versicolor	5,2	2,7	3,9	1,4	110	Iris Virginica	7,2	3,6	6,1	2,5
11	Iris Setosa	5,4	3,7	1,5	0,2	61	Iris Versicolor	5,0	2,0	3,5	1,0	111	Iris Virginica	6,5	3,2	5,1	2,0
12	Iris Setosa	4,8	3,4	1,6	0,2	62	Iris Versicolor	5,9	3,0	4,2	1,5	112	Iris Virginica	6,4	2,7	5,3	1,9
13	Iris Setosa	4,8	3,0	1,4	0,1	63	Iris Versicolor	6,0	2,2	4,0	1,0	113	Iris Virginica	6,8	3,0	5,5	2,1
14	Iris Setosa	4,3	3,0	1,1	0,1	64	Iris Versicolor	6,1	2,9	4,7	1,4	114	Iris Virginica	5,7	2,5	5,0	2,0
15	Iris Setosa	5,8	4,0	1,2	0,2	65	Iris Versicolor	5,6	2,9	3,6	1,3	115	Iris Virginica	5,8	2,8	5,1	2,4
16	Iris Setosa	5,7	4,4	1,5	0,4	66	Iris Versicolor	6,7	3,1	4,4	1,4	116	Iris Virginica	6,4	3,2	5,3	2,3
17	Iris Setosa	5,4	3,9	1,3	0,4	67	Iris Versicolor	5,6	3,0	4,5	1,5	117	Iris Virginica	6,5	3,0	5,5	1,8
18	Iris Setosa	5,1	3,5	1,4	0,3	68	Iris Versicolor	5,8	2,7	4,1	1,0	118	Iris Virginica	7,7	3,8	6,7	2,2
19	Iris Setosa	5,7	3,8	1,7	0,3	69	Iris Versicolor	6,2	2,2	4,5	1,5	119	Iris Virginica	7,7	2,6	6,9	2,3
20	Iris Setosa	5,1	3,8	1,5	0,3	70	Iris Versicolor	5,6	2,5	3,9	1,1	120	Iris Virginica	6,0	2,2	5,0	1,5
21	Iris Setosa	5,4	3,4	1,7	0,2	71	Iris Versicolor	5,9	3,2	4,8	1,8	121	Iris Virginica	6,9	3,2	5,7	2,3
22	Iris Setosa	5,1	3,7	1,5	0,4	72	Iris Versicolor	6,1	2,8	4,0	1,3	122	Iris Virginica	5,6	2,8	4,9	2,0
23	Iris Setosa	4,6	3,6	1,0	0,2	73	Iris Versicolor	6,3	2,5	4,9	1,5	123	Iris Virginica	7,7	2,8	6,7	2,0
24	Iris Setosa	5,1	3,3	1,7	0,5	74	Iris Versicolor	6,1	2,8	4,7	1,2	124	Iris Virginica	6,3	2,7	4,9	1,8
25	Iris Setosa	4,8	3,4	1,9	0,2	75	Iris Versicolor	6,4	2,9	4,3	1,3	125	Iris Virginica	6,7	3,3	5,7	2,1
26	Iris Setosa	5,0	3,0	1,6	0,2	76	Iris Versicolor	6,6	3,0	4,4	1,4	126	Iris Virginica	7,2	3,2	6,0	1,8
27	Iris Setosa	5,0	3,4	1,6	0,4	77	Iris Versicolor	6,8	2,8	4,8	1,4	127	Iris Virginica	6,2	2,8	4,8	1,8
28	Iris Setosa	5,2	3,5	1,5	0,2	78	Iris Versicolor	6,7	3,0	5,0	1,7	128	Iris Virginica	6,1	3,0	4,9	1,8
29	Iris Setosa	5,2	3,4	1,4	0,2	79	Iris Versicolor	6,0	2,9	4,5	1,5	129	Iris Virginica	6,4	2,8	5,6	2,1
30	Iris Setosa	4,7	3,2	1,6	0,2	80	Iris Versicolor	5,7	2,6	3,5	1,0	130	Iris Virginica	7,2	3,0	5,8	1,6
31	Iris Setosa	4,8	3,1	1,6	0,2	81	Iris Versicolor	5,5	2,4	3,8	1,1	131	Iris Virginica	7,4	2,8	6,1	1,9
32	Iris Setosa	5,4	3,4	1,5	0,4	82	Iris Versicolor	5,5	2,4	3,7	1,0	132	Iris Virginica	7,9	3,8	6,4	2,0
33	Iris Setosa	5,2	4,1	1,5	0,1	83	Iris Versicolor	5,8	2,7	3,9	1,2	133	Iris Virginica	6,4	2,8	5,6	2,2
34	Iris Setosa	5,5	4,2	1,4	0,2	84	Iris Versicolor	6,0	2,7	5,1	1,6	134	Iris Virginica	6,3	2,8	5,1	1,5
35	Iris Setosa	4,9	3,1	1,5	0,1	85	Iris Versicolor	5,4	3,0	4,5	1,5	135	Iris Virginica	6,1	2,6	5,6	1,4
36	Iris Setosa	5,0	3,2	1,2	0,2	86	Iris Versicolor	6,0	3,4	4,5	1,6	136	Iris Virginica	7,7	3,0	6,1	2,3
37	Iris Setosa	5,5	3,5	1,3	0,2	87	Iris Versicolor	6,7	3,1	4,7	1,5	137	Iris Virginica	6,3	3,4	5,6	2,4
38	Iris Setosa	4,9	3,1	1,5	0,1	88	Iris Versicolor	6,3	2,3	4,4	1,3	138	Iris Virginica	6,4	3,1	5,5	1,8
39	Iris Setosa	4,4	3,0	1,3	0,2	89	Iris Versicolor	5,6	3,0	4,1	1,3	139	Iris Virginica	6,0	3,0	4,8	1,8
40	Iris Setosa	5,1	3,4	1,5	0,2	90	Iris Versicolor	5,5	2,5	4,0	1,3	140	Iris Virginica	6,9	3,1	5,4	2,1
41	Iris Setosa	5,0	3,5	1,3	0,3	91	Iris Versicolor	5,5	2,6	4,4	1,2	141	Iris Virginica	6,7	3,1	5,6	2,4
42	Iris Setosa	4,5	2,3	1,3	0,3	92	Iris Versicolor	6,1	3,0	4,6	1,4	142	Iris Virginica	6,9	3,1	5,1	2,3
43	Iris Setosa	4,4	3,2	1,3	0,2	93	Iris Versicolor	5,8	2,6	4,0	1,2	143	Iris Virginica	5,8	2,7	5,1	1,9
44	Iris Setosa	5,0	3,5	1,6	0,6	94	Iris Versicolor	5,0	2,3	3,3	1,0	144	Iris Virginica	6,8	3,2	5,9	2,3
45	Iris Setosa	5,1	3,8	1,9	0,4	95	Iris Versicolor	5,6	2,7	4,2	1,3	145	Iris Virginica	6,7	3,3	5,7	2,5
46	Iris Setosa	4,8	3,0	1,4	0,3	96	Iris Versicolor	5,7	3,0	4,2	1,2	146	Iris Virginica	6,7	3,0	5,2	2,3
47	Iris Setosa	5,1	3,8	1,6	0,2	97	Iris Versicolor	5,7	2,9	4,2	1,3	147	Iris Virginica	6,3	2,5	5,0	1,9
48	Iris Setosa	4,6	3,2	1,4	0,2	98	Iris Versicolor	6,2	2,9	4,3	1,3	148	Iris Virginica	6,5	3,0	5,2	2,0
49	Iris Setosa	5,3	3,7	1,5	0,2	99	Iris Versicolor	5,1	2,5	3,0	1,1	149	Iris Virginica	6,2	3,4	5,4	2,3
50	Iris Setosa	5,0	3,3	1,4	0,2	100	Iris Versicolor	5,7	2,8	4,1	1,3	150	Iris Virginica	5,9	3,0	5,1	1,8

Konwersja danych do postaci struktury AGDS

Klasyfikatory zwykle budujemy dla zbiorów danych uczących zawierających wzorce zebrane w przeszłości i zapisane w postaci tabel o określonej liczbie parametrów (cech, atrybutów). **Wzorce uczące** określają wartości tych parametrów oraz klasę, którą reprezentują. W związku z tym, iż takie dane nie są zwykle zbierane na żywo, a kolejność ich występowania w tabelach zazwyczaj nie ma żadnego znaczenia dla klasyfikacji, asocjacje następstwa ASEQ ani kontekstu ACON nie będą w tym przypadku rozważane. Asocjacje ASEQ i ACON mają ogromne znaczenie w procesach poznawczych (kognitywnych), ale można je wprowadzać do modelu tylko wtedy, gdy są one rzeczywiste. W odwrotnym przypadku skonstruowany model oparty o asocjacyjne powiązania ASEQ wynikające z kolejności wzorców w zbiorze uczącym spowodowałyby błędne skojarzenia na skutek pobudzenia i łatwiejszej aktywacji neuronów reprezentujących kolejne wzorce uczące. Z tego powodu niezbędna może okazać się pomoc eksperta określającego, czy następstwo wzorców w określonym zbiorze danych ma znaczenie. Asocjacje następstwa ASEQ i kontekstu ACON zostaną wykorzystane podczas formowania się wiedzy na przykładach z zakresu lingwistyki.

Prezentowany zbiór Iris z ML Repository składa się ze 150 wzorców, przy czym każdy z nich opisany jest przez 4 parametry, określające kolejno: długość i szerokość liścia, długość i szerokość płatk kwiatu oraz nazwy gatunku (tutaj klasy). W tym zbiorze występują 3 gatunki irysów: Setosa, Versicolor i Virginica, przy czym każda klasa reprezentowana jest przez 50 wzorców uczących. Niektóre bardziej skomplikowane etapy budowy tego klasyfikatora będą dla czytelności prezentowane na 13 skorelowanych wzorcach z tego zbioru.

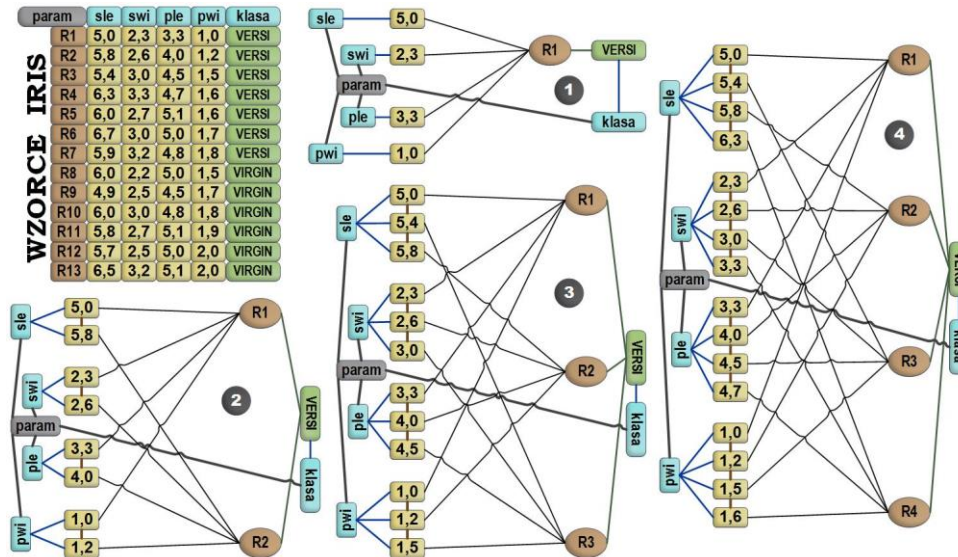
Porównując tabelę 7.1 przedstawiającą zbiór danych Iris reprezentowany przy pomocy klasycznej tabeli danych oraz grafową strukturę AGDS z rysunku 7.5 może się pozornie wydawać, że struktury AGDS są bardzo nieoszczędnym sposobem przechowywania danych, gdyż oprócz wartości (węzłów grafu) zawierają całą masę różnego rodzaju powiązań (krawędzi grafu) pomiędzy nimi. Jeśliby patrzeć na struktury AGDS tylko jako na struktury przechowujące dane, w istocie byłyby nieoszczędne, szczególnie dla małych zbiorów uczących. Jednak one nie tylko przechowują dane jak w przypadku klasycznych tabel, lecz przechowują również ważne relacje pomiędzy tymi danymi w postaci powiązań, które ujawniają swoją użyteczność w przypadku potrzeby odnalezienia pewnych danych. Wtedy te powiązania eliminują konieczność stosowania bardziej kosztownych

obliczeniowo algorytmów zawierających wiele iteracyjnych pętli obliczeniowych, dostarczając wyniki przy niższej złożoności obliczeniowej niż potrzebnej do wykonania tych klasycznych algorytmów przeszukujących pasywne tabele. Kosztem nieco bardziej skomplikowanej struktury uzyskujemy bogatą funkcjonalność, która z powodzeniem działa w naturze od milionów lat, tworząc skomplikowane, lecz bardzo efektywnie przetwarzające dane struktury nerwowe. Oszczędność takich struktur ujawnia się dopiero podczas reprezentacji bardzo dużych ilości danych ze względu na brak duplikacji reprezentowanych danych i ich kombinacji, możliwość ich reprezentacji z ograniczoną dokładnością oraz agregację wspólnych powiązań. Wobec tego poza innymi zaletami struktury AGDS mogą również pod kątem efektywności pamięciowej przewyższyć klasyczne formy przechowywania danych dla dużych zbiorów danych.

Konwersję rekordów danych (wzorców) przechowywanych w klasycznych tabelach do struktury AGDS przeprowadzamy w taki sposób, że zamieniamy nazwy atrybutów (parametrów, cech), nazwy rekordów (ew. ich klucze) oraz poszczególne ich wartości na węzły grafu AGDS (rys. 7.3 i 7.4). Węzły reprezentujące wartości łączymy krawędziami EDEF z węzłami reprezentującymi nazwy (klucze) rekordów. Węzły reprezentujące atrybuty tabel będą później transformowane na wejścia sensoryczne (tzw. SENSINy) grafu AANG. Jeśli w tabeli występują duplikaty rekordów, wtedy inkrementowana jest tylko wartość węzła informująca o ilości takich duplikatów. Duplikaty rekordów odnajdywane są z łatwością dzięki temu, że również reprezentacja wartości poszczególnych atrybutów nie jest duplikowana. Dzięki temu odnajdywanie podobieństw, różnic i korelacji pomiędzy rekordami jest łatwe i bardzo szybkie. Jeśli klucze jednej tabeli powiązane są pewnymi relacjami z danymi w innej tabeli, wtedy te powiązania zapisujemy przy pomocy krawędzi EDEF. Natomiast wartości danych, zapisanych w poszczególnych kolumnach takich tabel, łączymy również krawędziami EDEF z węzłem reprezentującym dany atrybut, gdyż wartości te definiują zakres zmienności i możliwe wartości danego atrybutu. Węzły wartości atrybutów są następnie łączone ze sobą krawędziami ESIM w zależności od ich bliskości (podobieństwa), jeśli tylko istnieje pewna relacja porządkująca dla nich, uzyskując zarazem posortowane listy niezduplikowanych wartości dla poszczególnych atrybutów, które są częścią grafu AGDS. W bazach danych stosuje się indeksowanie, które tutaj jest zbędne, gdyż wszystkie porządkowalne wartości atrybutów są automatycznie sortowane w trakcie ich dodawania do tego grafu. Jeśli kolejność rekordów w tabeli ma znaczenie, wtedy węzły reprezentujące rekordy są dodatkowo łączone krawędziami skierowanymi ESEQ odwzorowując ich kolejność. Ponadto, graf odwzorowuje wszystkie znane powiązania pomiędzy danymi i rekordami. Jeśli do dyspozycji jest ekspert, który może wskazać inne rzeczywiste lub dodatkowe zależności pomiędzy atrybutami danych, wtedy dodanie takich połączeń jest również wartościowe.

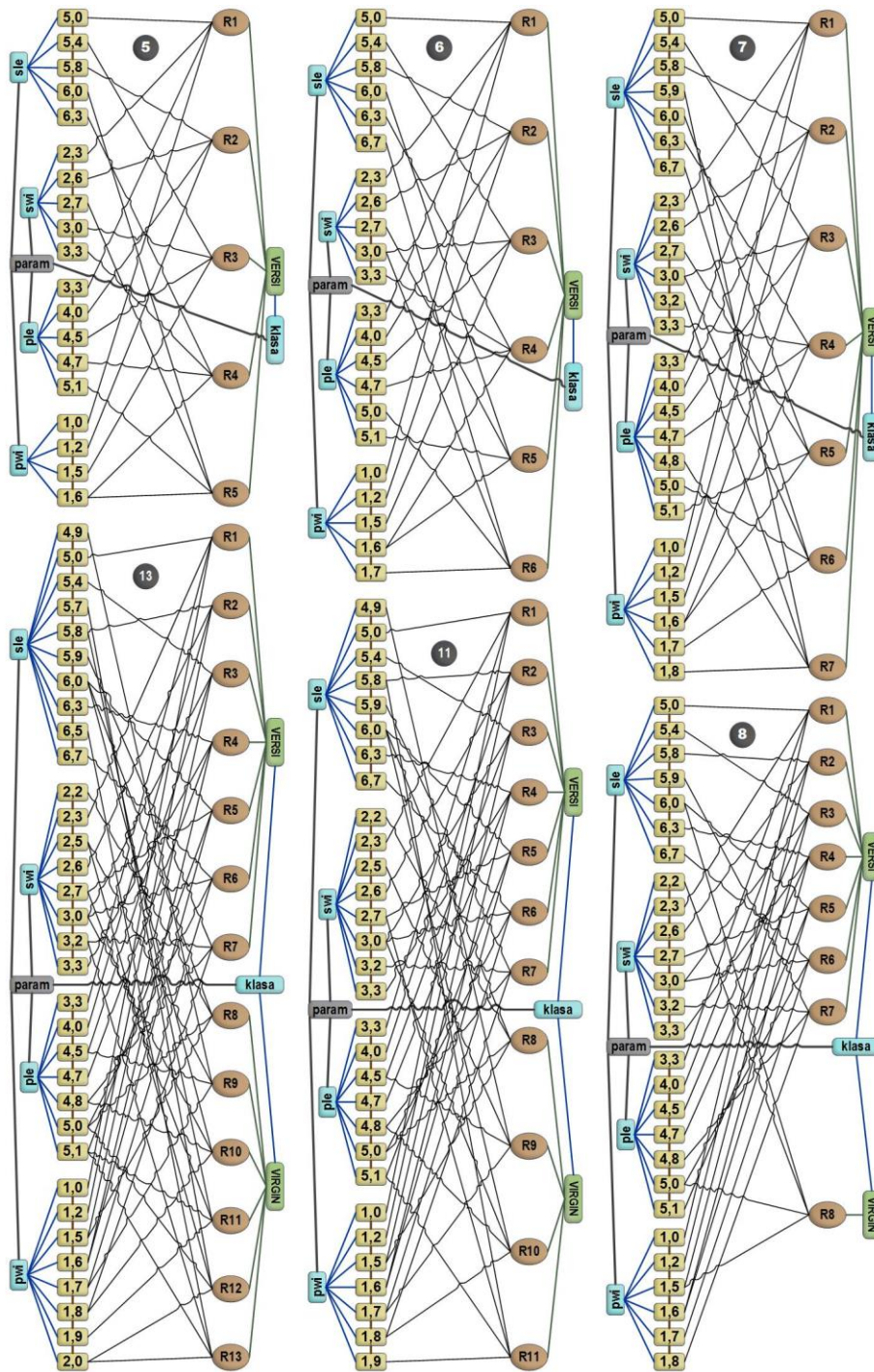
Kilka etapów stopniowej konwersji kilkunastu wybranych, skorelowanych wzorców z tabeli 7.1 na strukturę AGDS zilustrowano na rysunkach 7.3 i 7.4, zaś etap końcowy wygenerowany przez aplikację komputerową pokazano na rysunku 7.5. W bardziej zaawansowanych etapach budowy struktury AGDS (rys. 7.4) węzły reprezentujące wspólne wartości atrybutów połączone są z węzłami reprezentującymi różne wzorce. Dzięki tym połączeniom można łatwo określić wzorce zawierające mniejsze lub większe wartości dla wybranych porządkowalnych atrybutów oraz wyznaczyć ich podobieństwo, korelacje, różnice, stopień dyskryminacji, o których źródłem informacji staje się ten graf. Na rysunku 7.4 można również dostrzec, iż same wartości poszczególnych parametrów nie dublują się, lecz są reprezentowane przez te same węzły, a więc automatycznie ze wzrostem ilości reprezentowanych wzorców zawierających te same wartości struktura AGDS staje się coraz bardziej oszczędna. Atrybuty (cechy) wzorców oraz nazwy klas mogą być traktowane równorzędnie jako pewne parametry rekordów z konwertowanej tabeli z rysunku 7.3, z których klasy można wyłonić później tak, jak to pokazano na rysunku 3.8. Struktura AGDS ma postać pasywną tak samo jak inne struktury danych stosowane w informatyce, a więc chcąc wydobyć pewne informacje z nich trzeba wykorzystać algorytmy przeszukujące i porównujące wartości reprezentowane przez poszczególne połączone ze sobą węzły. Do pewnej automatyzacji dochodzi dopiero po ich transformacji na postać aktywnego asocjacyjnego grafu neuronowego AANG, w którym można pobudzać różne kombinacje neuronów i otrzymać różne informacje na podstawie tak wywołanych skojarzeń. Teoretycznie dane można grupować względem wartości dowolnego parametru, dla którego wszystkie lub prawie wszystkie wartości się powtarzają, gdyż każda klasa powinna być zawsze określona przez pewną ilość reprezentantów w zbiorze (uczącym). Zazwyczaj poszczególne wartości parametru reprezentujące jakieś ważne klasy powtarzają się najliczniej, a więc można próbować automatycznie określić nazwy potencjalnie najważniejszych klas w danym zbiorze. Jednak nawet mniej licznie powtarzające się wartości dla poszczególnych parametrów określają pewne klasy pod warunkiem, że każda lub prawie każda wartość tego parametru powtórzy się kilkakrotnie w tym zbiorze (uczącym). Tak więc każdy parametr, którego wszystkie lub prawie wszystkie wartości powtarzają się w zbiorze danych, automatycznie określa **klasę** poprzez te wartości. Ma to fundamentalne znaczenie w systemach skojarzeniowych, w których taką powtarzalność można automatycznie wykrywać i wykorzystywać **do automatycznego definiowania nowych semasseli**, gdyż każdy semassel powinien być określony przez minimalnie kilka kombinacji lub układów podobnie jak klasa przez kilku reprezentantów. Teoretycznie im większy jest zbiór wzorców, tym większe jest prawdopodobieństwo wystąpienia większej ilości parametrów, dla których wszystkie lub większość wartości się powtarza i umożliwia określenie nowych klas. Te najczęściej powtarzające się mogą w grafie AANG utworzyć semassele, które są fundamentem dla dalszych

skojarzeń i **automatycznego formowania się wiedzy skojarzeniowej**. Niekoniecznie jednak tak określone klasy są klasami w zamyśle twórcy danego zbioru danych. Ważne jest jednak to, iż struktury AGDS można wykorzystać do efektywnej **eksploracji klas** (*class mining*). Jeśli jednak z góry wiadomo, który parametr rekordu jest nazwą klasy w zamyśle twórcy zbioru danych, wtedy można go od razu wyróżnić, a węzły grafu reprezentujące powiązane z nim wartości połączyć krawędziami EDEF z węzłami reprezentującymi nazwy rekordów, które będą definiować te klasy.



Rys. 7.3 Wybrane skorelowane wzorce ze zbioru danych Iris z tabeli 7.1 oraz pierwsze etapy ich konwersji na postać struktury AGDS.

Na rysunku 7.5 wygenerowanym przez aplikację komputerową po lewej stronie u góry znajdują się węzły reprezentujące atrybuty, pod nimi węzły reprezentujące wartości tych atrybutów, następnie z prawej strony u dołu są węzły reprezentujące nazwy rekordów, a powyżej nich węzły reprezentujące klasy. Ze względu na czytelność pogrupowano węzły grafu reprezentujące wartości poszczególnych atrybutów oraz nazwy rekordów (wzorców) reprezentujących poszczególne klasy przedstawiając je w kolumnach pod sobą. W rzeczywistości jednak każdy z nich jest oddzielnym węzłem grafu połączonym z węzłem u góry reprezentującym atrybut albo klasę, którą definiuje wraz z innymi węzłami z tej samej kolumny. Klasyczna tabela danych uczących przedstawiona w tabeli 7.1 posiada 966 komórek (danych) zawierających wartości parametrów wzorców łącznie z etykietami atrybutów, etykietami



Rys. 7.4. Kolejne etapy konwersji kilku wybranych, skorelowanych wzorców ze zbioru danych Iris z tabeli 7.1 na postać struktury AGDS nawiązujące do rysunku 7.3.

wzorców (rekordów) i nazwami klas. Struktura AGDS pokazana na rysunku 7.5 odzwierciedlająca tę tabelę zawiera tylko 277 węzłów przechowujących dane oraz 977 krawędzi EDEF i ESIM odwzorowujących asocjacyjne relacje ADEF i ASIM występujące pomiędzy nimi. Te krawędzie jednak upraszczają część klasycznych algorytmów, a więc pełnią też rolę funkcjonalną. Ponadto graf AGDS zawiera w sobie listy posortowanych wartości poszczególnych atrybutów, przyspieszając wyszukiwanie podobnych, kolejnych lub poprzednich rekordów bez konieczności stosowania dodatkowych indeksów lub konieczności przesortowania danych, co jest niezbędne w przypadku korzystania z klasycznych tabel danych.

Już na etapie konwersji zbioru uczącego Irys do postaci AGDS automatycznie ujawnia się, iż 3 wzorce z klasy Setosa (10, 35 i 38) oraz 2 wzorce z klasy Virginica (102 i 143) są identyczne (tab. 7.1). Ze względu na to, iż zasady obliczeń asocjacyjnych nie pozwalają na duplikowanie reprezentacji identycznych wartości danych ani ich kombinacji (semasseli) w strukturze AGDS, zduplikowane wartości atrybutów oraz zduplikowane wzorce

uczące są reprezentowane przez te same węzły w strukturze AGDS, a później przez te same neurony receptoryczne VN i neurony SN w grafie AANG. Duplikaty wartości atrybutów odwzorowane są w postaci odpowiednio większej ilości krawędzi EDEF wychodzących z węzłów je reprezentujących. Zaś duplikaty wzorców (rekordów) z danej klasy pozostawiają po sobie ślad jedynie w postaci odpowiednio większej wartości wagi krawędzi EDEF łączącej taki węzeł reprezentujący ten wzorec z węzłem reprezentującym klasę. Wagi EDEF zliczają ilość wystąpień duplikatów danych lub ich kombinacji (tutaj: wzorec), które definiują inny węzeł (tutaj: klasę). W przypadku, gdy tylko same wartości atrybutów są identyczne, a nazwy klas są inne, dochodzi do wieloznaczności, co jest odzwierciedlone w postaci wspólnego węzła reprezentującego powtarzającą się kombinację wartości atrybutów, który połączony jest krawędziami EDEF z kilkoma węzłami reprezentującymi różne klasy. Takie węzły od razu na etapie konwersji danych do struktury AGDS ujawniają takie niejednoznaczności, które mogą być poprawne (multiklasyfikacja, wieloznaczność) lub niepoprawne (sprzeczność, błędne dane). Trudniejszym zadaniem od wyszukiwania duplikatów jest określenie stopnia podobieństwa lub korelacji pomiędzy wzorcami uczącymi, należącymi do tej samej lub różnych klas, co jest szczególnie istotne z punktu widzenia klasyfikacji rozważanej w tym rozdziale. Dzięki braku duplikacji reprezentacji i wielokrotnym połączeniom definicyjnym EDEF to zadanie jest również proste, a wyniki można bardzo szybko wyznaczyć. Klasyfikacyjne algorytmy operujące na tabelach musiałyby wykonać kilka zagnieżdżonych pętli, aby wyznaczyć duplikaty, korelacje lub podobieństwa.

Podsumowując, jednokrotna konwersja zbiorów danych ze struktur tabelarycznych do postaci struktury AGDS umożliwia:

- ✓ posortowanie wszystkich rekordów równocześnie według wszystkich ich atrybutów bez konieczności indeksowania;
- ✓ automatyczne wykrywanie duplikatów i ich zagregowaną reprezentację;
- ✓ natychmiastowe określenie kolejności, podobieństw, różnic i korelacji dla dowolnych rekordów biorąc pod uwagę wybrane lub wszystkie ich parametry;
- ✓ łatwe wykrywanie i eksplorację klas (*class mining*) bez konieczności wielokrotnego przeszukiwania i eksploracji danych (*data mining*);
- ✓ jego łatwą transformację do postaci aktywnego asocjacyjnego grafu neuronowego AANG realizującego asocjacyjne neuroobliczenia grafowe, które umożliwiają wywoływanie skojarzeń oraz korzystanie z asocjacyjnych mechanizmów plastyczności i genetyzy.

Budowa aktywnego asocjacyjnego grafu neuronowego AANG

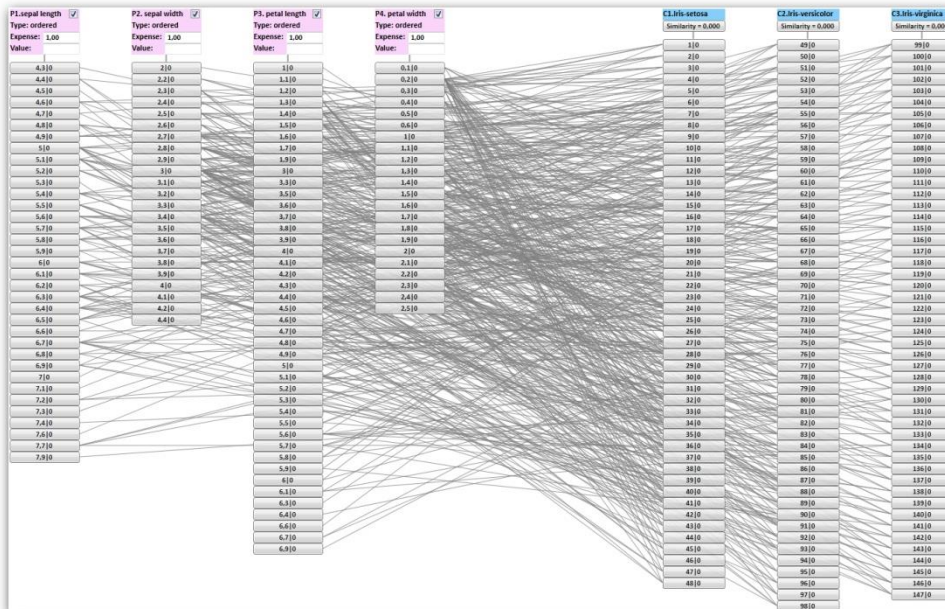
Budowę aktywnego asocjacyjnego grafu neuronowego AANG (*active associative neural graph*) dla potrzeb budowy asocjacyjnego klasyfikatora ASONN można przeprowadzić zasadniczo na dwa sposoby, które zostały już częściowo zilustrowane w poprzednim rozdziale na przykładzie sortowania asocjacyjnego ASSORT:

1. Można najpierw dokonać konwersji danych do postaci struktury AGDS, a następnie przekształcić jej węzły na neurony, a jej etykietowane krawędzie na ważone połączenia. Wtedy wagi połączeń wyznaczane są na podstawie podobieństw wartości reprezentowanych przez VNy, wartości wag krawędzi ESIM i EDEF oraz ilości i stopnia pozytywnych i negatywnych korelacji wartości poszczególnych atrybutów definiujących wzorce.
2. Można zbudować inicjalną strukturę grafu AANG składającą się z SENSINów, a następnie kolejnymi wzorcami pobudzać graf AANG, stopniowo go rozbudowując, korzystając z automatycznych mechanizmów i zasad obliczeń asocjacyjnych. Potrzebne jest wtedy określenie programów plastyczności poszczególnych elementów, które automatycznie aktywują się w pewnych okolicznościach i kontekście stanu poszczególnych elementów tego grafu.

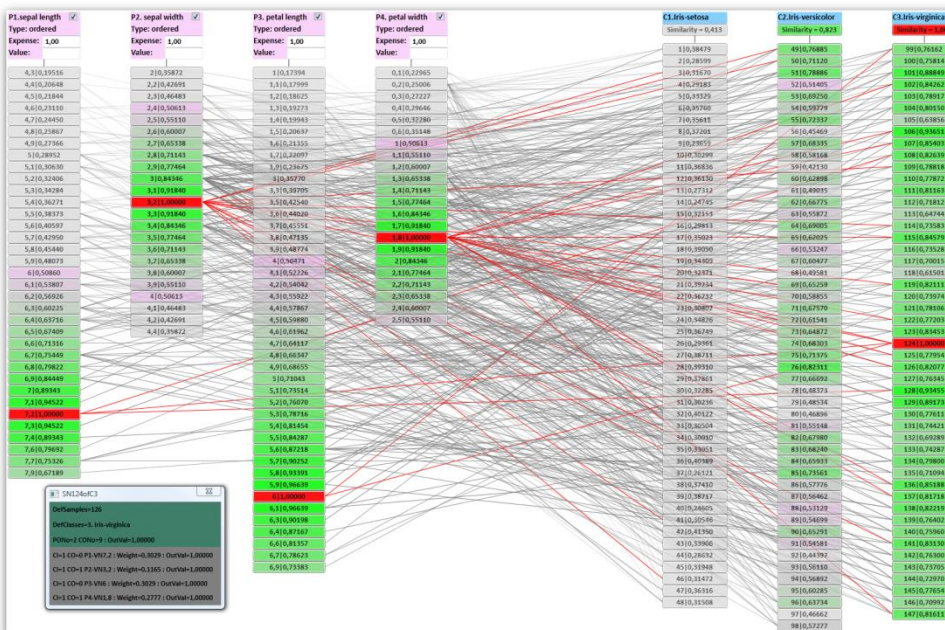
Do budowy klasyfikatora ASONN zastosowano hybrydowe podejście, wykorzystując zarówno możliwość pobudzenia i aktywacji neuronów grafu AANG, jak również część obliczeń wykonano w neuronach tak jak na pasywnych węzłach struktury AGDS, które zbierają część informacji i przesyłają je do powiązanych z nimi węzłów (neuronów). Dzięki takiemu podejściu i brakowi połączeń ASEQ i ACON nie trzeba było korzystać ze skomplikowanej symulacji równoległości i czasu. Struktura AGDS w procesie budowy klasyfikatora ASONN umożliwiła szybkie wyznaczenie podobieństw, różnic, korelacji i ich stopnia pomiędzy wzorcami, w celu określenia ich reprezentatywno-dyskryminatywnych kombinacji definiujących poszczególne klasy. Maksymalizacja poziomu reprezentatywności odbywa się jednak często kosztem stopniowego zmniejszania się stopnia dyskryminacji, a więc rozwiązanie jest zawsze pewnym kompromisem pomiędzy nimi. Budowane kombinacje reprezentowane przez neurony kombinacji KN grafu AANG są w asocjacyjny sposób (ADEF) związane z innymi neuronami reprezentującymi przedziały, podzbiory i wzorce uczące. Ze względu na to, iż ASONN jest klasyfikatorem, a nie multiklasyfikatorem, wszystkie wzorce uczące o identycznych wartościach atrybutów

definiujące równocześnie kilka klas traktowane są jako sprzeczne i nie są brane pod uwagę przy budowie tego klasyfikatora. Reprezentujące je neurony wzorców *SN* są w strukturze AANG hibernowane na czas jego budowy.

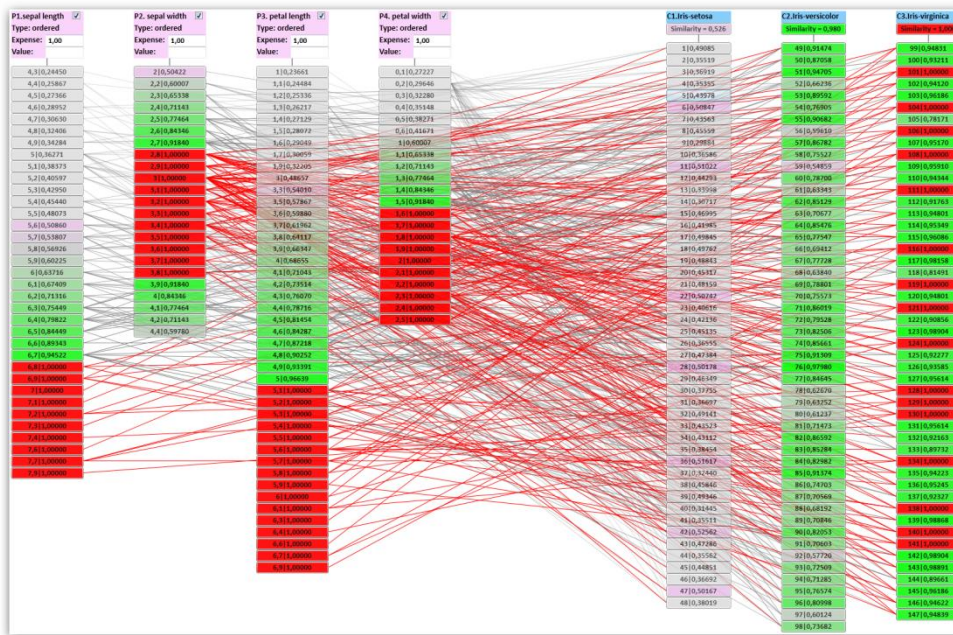
Budowę kombinacji warto rozpocząć od najbardziej negatywnie skorelowanych neuronów wzorców, stosując zasadę, iż rozwiązanie problemu należy rozpocząć od najtrudniejszych jego części, gdyż one stanowią wąskie gardło rozwiązania. Takie podejście jest uzasadnione, gdyż to właśnie najbardziej negatywnie skorelowane neurony będą wyznaczały maksymalny osiągalny stopień dyskryminatywności kombinacji oraz najtrudniej będzie je dołączyć do jakiegokolwiek neuronu kombinacji, żeby nie utracić jego stopnia dyskryminatywności. Najprościej dołączyć można do kombinacji neurony wzorców najmniej skorelowanych z wzorcami innych klas, nie redukując stopnia dyskryminatywności całej kombinacji.



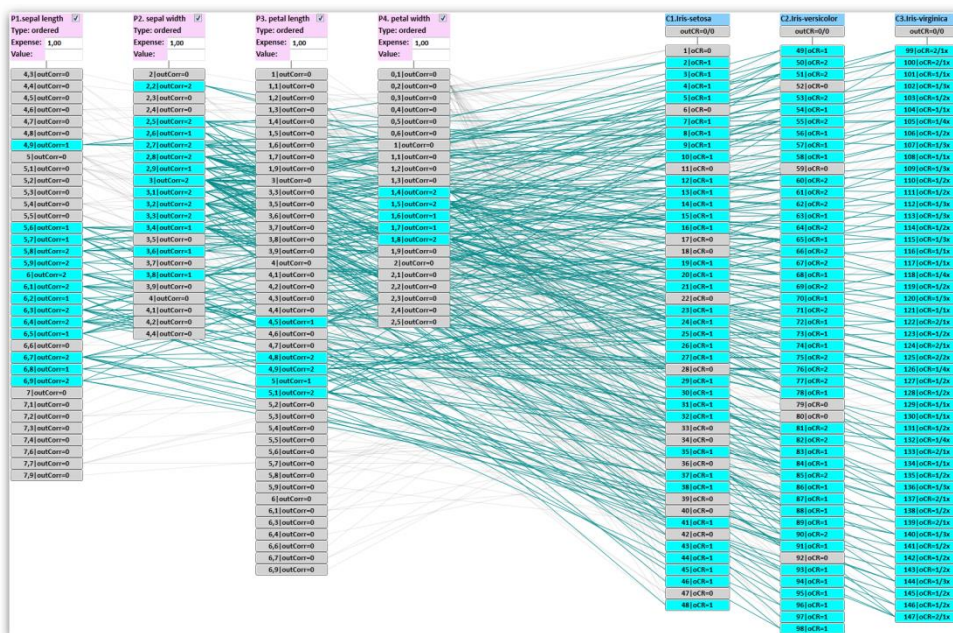
Rys. 7.5 Grafowa asocjacyjna struktura AGDS wygenerowana przez aplikację komputerową dla 150 wzorców uczących ze zbioru Iris pochodzącego z bazy ML Repository.



Rys. 7.6 Aktywacja neuronu wzorca SN₁₂₄, reprezentującego wzorec S₁₂₆ w grafie AANG, od razu ujawnia wszystkie korelacje i podobieństwa ze wzorcami tej samej jak również innych klas dzięki asocjacyjnym połączeniom ADEF i ASIM.



Rys. 7.7 Aktywacja wybranych zakresów wartości dla poszczególnych parametrów wejściowych, powodujących aktywację 16 wzorców uczących klasy Virginica w grafie AANG.



Rys 7.8 Korelacja wszystkich neuronów wzorców z klasy Virginica z wszystkimi pozostałymi neuronami wzorców innych klas. Liczby określają ilość skorelowanych atrybutów.

Negatywne korelacje można bardzo łatwo i szybko wyznaczyć w grafie AANG poprzez aktywowanie neuronów *SN* reprezentujących badane wzorce uczące, które pobudzają neurony wartości *VN* reprezentujące definiujące je wartości atrybutów, a te w zależności od ilości połączeń z innymi neuronami wzorców, które słabiej lub mocniej pobudzają, określając w ten sposób stopień skorelowania z nimi. Jeśli tak pobudzone neurony wzorców *SN* są równocześnie połączone z neuronem klasy *CN* aktywowanym przez neuron zalążkowy *germSN* rozbudowywanej kombinacji, wtedy mamy do czynienia z korelacjami własnymi, a więc pożądanymi potencjalnymi kandydatami do rozbudowy kombinacji. Jeśli zaś nie są równocześnie połączone z aktywowanym neuronem klasy *CN*, wtedy te neurony definiują inne klasy i mamy do czynienia z korelacjami obcymi, a neurony tych wzorców muszą zostać w pożądanym stopniu zdyskryminowane od wszystkich pozostałych neuronów wzorców powiązanych z rozważanym neuronem konstruowanej kombinacji.

Przed rozpoczęciem procesu tworzenia neuronów kombinacji *KN*, trzeba najpierw określić stopień korelacji obcej wszystkich neuronów wzorców *SN*. Następnie dla wszystkich niepowiązanych jeszcze z żadnym *KNem*

neuronów wzorców *SN* o najwyższym stopniu korelacji obcej trzeba stopniowo dodawać kolejne neurony *KN* i łączyć je z najbardziej pozytywnie skorelowanymi neuronami *SN* tak długo, aż aktualnie tworzona kombinacja osiągnie możliwie wysoki poziom reprezentatywności przy pożądanym lub akceptowalnym stopniu dyskryminatywności. Kolejne neurony *KN* budowane są dla kolejnych najbardziej negatywnie skorelowanych neuronów *SN*, które jeszcze nie są połączone z istniejącymi neuronami *KN*. Proces ten trwa tak długo, dopóki wszystkie niezahibernowane neurony *SN* (reprezentujące niesprzeczne wzorce uczące) każdej klasy nie zostaną połączone z przynajmniej jednym neuronem *KN*. W trakcie tworzenia kolejnych *KN*ów często dochodzi do kilkukrotnego połączenia najmniej negatywnie skorelowanych *SN*ów z różnymi *KN*ami, otrzymując pewną pożądaną redundancję.

O stopniu korelacji wzorców uczących decydują definiujące je wartości atrybutów, które mogą być bliskie lub identyczne. Najistotniejsze są oczywiście wartości identyczne, gdyż ich ilość decyduje o możliwości dyskryminacji wzorców różnych klas. W przypadku wzorców tej samej klasy, wartości identyczne pozytywnie wpływają na możliwość bardziej oszczędnej ich reprezentacji w grafie AANG oraz łatwość budowy bardziej reprezentatywnych kombinacji dla nich. Bliskie wartości definiujące wzorce różnych klas decydują o uwarunkowaniu dyskryminacji, a pośrednio o jakości uogólniania, gdyż ich nieuwzględnienie w procesie budowy kombinacji może spowodować utratę dyskryminacji przez niewielką zmianę wartości atrybutu. Rysunek 7.6 prezentuje strukturę węzłów AGDS i równocześnie strukturę neuronów grafu AANG dla zbioru uczącego Iris z ML Repository oraz pewien jego stan pobudzenia poprzez aktywację neuronu wzorca *SN*₁₂₄, reprezentującego wzorec uczący *S*₁₂₆. Zilustrowano wszystkie bezpośrednio i pośrednio pobudzone neurony wartości, wzorców i klas. Kolorystycznie i liczbowo pokazano stopień pobudzenia każdego z neuronów w grafie AANG. Czerwony kolor oznacza aktywację, a zielony pobudzenie i jego nasilenie. Graf AANG od razu, tzn. ze stałą złożonością obliczeniową, ujawnia wszystkie istniejące podobieństwa i korelacje. Aktywacja neuronu wzorca *SN*₁₂₄ (zaznaczonego na czerwono) spowodowała aktywację definiujących go receptorycznych neuronów wartości *VN*_{7,2}, *VN*_{3,2}, *VN*_{6,0} i *VN*_{1,8} oraz pewne pobudzenia sąsiednich (bliskich) receptorycznych neuronów wartości (zaznaczonych na zielono), które łącznie pobudziły różne neurony wzorców i pośrednio ujawniły podobieństwa wzorca *S*₁₂₆ do innych wzorców tego zbioru uczącego, reprezentowanych przez odpowiednie neurony wzorców *SN*.

Na tym przykładzie również widać, iż aktywowany wzorec jest przede wszystkim podobny do wielu wzorców z jego klasy. To z punktu widzenia klasyfikacji jest zjawiskiem pozytywnym. Stopień korelacji neuronu wzorca *SN*₁₂₄ z neuronami wzorców innych klas jest tutaj dużo mniejszy. Najbardziej negatywnie skorelowanym neuronem wzorca innej klasy jest *SN*₇₆ z klasy Versicolor. W opisie działania grafu AANG posłużono się terminem „neuron wzorca”, a nie „wzorec”, gdyż „neuron wzorca” reprezentuje w grafie AANG pewien wzorec lub kilka zduplikowanych wzorców, dlatego też numeracja neuronów wzorców, których jest 147, różni się od numeracji samych wzorców, których jest 150, gdyż zbiór Irys zawiera takie duplikaty. Ze względu na zasady obliczeń asocjacyjnych duplikaty są reprezentowane przez te same neurony wzorców, np. *SN*₁₀ reprezentuje *S*₁₀, *S*₃₅ i *S*₃₈, które mają identyczne wartości (tab. 7.1). W tym przypadku **neuron wzorca** (*sample neuron*) *SN*₁₂₄ reprezentuje **wzorec uczący** (*sample*) o numerze 126 oznaczonym jako *S*₁₂₆.

Rysunek 7.7 ujawnia 16 aktywowanych neuronów wzorców *SN* powiązanych z neuronem klasy Virginica pod wpływem aktywacji neuronów wartości reprezentujących pewne zwarte przedziały wartości dla poszczególnych atrybutów. Aktywację osiągnął też neuron klasy Virginica, co sprawia, że taka właśnie kombinacja wartości parametrów może być wykorzystana do dyskryminacji i budowy klasyfikatora, gdyż żaden neuron wzorca powiązany z innym neuronem klasy nie został aktywowany, lecz co najwyżej pobudzony. Doszło więc do skojarzenia wybranych sąsiednich i bliskich co do reprezentowanych wartości neuronów wartości *VN* z 16 neuronami wzorców reprezentujących aż 32% wzorców tej samej klasy. Można uznać, że ta kombinacja reprezentuje pokaźną ich część, a więc jest reprezentatywna. Jeśliby znaleźć takie kombinacje zawierające również pozostałe wzorce tej klasy, wtedy wspólnie te kombinacje reprezentowałyby wszystkie wzorce tej klasy i mogłyby być razem wykorzystane w roli klasyfikatora. Na takich kombinacjach bazuje klasyfikator ASONN. W jaki sposób jednak znaleźć takie kombinacje? Na gruncie klasycznej informatyki problem ten jest NP-trudny, gdyż próba znalezienia takich kombinacji wiąże się ze sprawdzeniem $O(n!)$ możliwości [129]. Na gruncie asocjacyjnej sztucznej inteligencji AAI wykorzystującej wiedzę uformowaną w strukturze i parametrach grafów AANG i równoległej realizacji obliczeń asocjacyjnych mamy możliwość wyznaczenia wszystkich dyskryminatywnych kombinacji przy liniowej złożoności obliczeniowej. W przypadku hybrydowego podejścia zastosowanego tutaj wszystkie kombinacje wyznaczane są przy złożoności kwadratowej. Trzeba jednak zaznaczyć, iż wyznaczony zbiór kombinacji nie musi być jedynym możliwym, gdyż może istnieć kilka takich zbiorów, które dyskryminują wszystkie wzorce uczące. W przypadku klasyfikacji wystarcza nam jednak jeden taki zbiór pod warunkiem, że zawarte w nim kombinacje są możliwie jak najbardziej reprezentatywne przy możliwie wysokim stopniu dyskryminacji definiujących je wzorców od wzorców innych klas. Wyznaczenie kombinacji nie zachodzi jednak ani losowo, ani przypadkowo, ale związane jest z wykorzystaniem wiedzy o zbiorze danych „zakodowanej” w strukturze i parametrach tego grafu, co sprawia, że otrzymujemy takie optymalne lub suboptymalne rozwiązanie

szybko w porównaniu z klasycznymi algorytmami operującymi na tabelach. Zresztą podobną sytuację mamy w trakcie ludzkich procesów myślowych. Ludzie, korzystając ze swojej wiedzy, niejednokrotnie podejmują mądre i efektywne decyzje, aczkolwiek może się okazać, iż nie są one jedynymi możliwymi ani najbardziej efektywnymi w określonej sytuacji lub kontekście. Podobnie jest w AASach.

Celem budowy klasyfikatora ASONN jest znalezienie możliwie niewielkiej (najlepiej minimalnej) ilości reprezentatywnych kombinacji dla każdej klasy, które by zarazem dyskryminowały reprezentowane przez nie wzorce od wszystkich pozostałych wzorców innych klas w stopniu możliwie dużym i niemniejszym od jedności dla uzyskania poprawnej i jednoznacznej klasyfikacji wszystkich wzorców uczących. Oczywiście im wyższy stopień dyskryminatywności zostanie osiągnięty dla poszczególnych kombinacji danej klasy, tym większa pewność, że wzorce testowe lub walidacyjne nie zostaną błędnie przyporządkowane do niewłaściwej klasy i równocześnie wynik klasyfikacji będzie jednoznaczny. Natomiast wymaganie zbyt wysokiego stopnia dyskryminacji prowadzi zazwyczaj do tworzenia się dużej ilości mniej reprezentatywnych kombinacji, a to wpływa na obniżenie się możliwości uogólniających klasyfikatora. Może to spowodować, iż pewne wzorce testowe lub walidacyjne nie znajdują się w obrębie żadnej z nich, a wynik działania klasyfikatora będzie oparty tylko o maksymalne wartości pobudzenia rozmytych granic hiperprostopadłościaków reprezentowanych przez te kombinacje. Dążąc więc do maksymalizacji poprawności uogólniania klasyfikatora ASONN, należy znaleźć sensowny kompromis pomiędzy reprezentatywnością i dyskryminatywnością poszczególnych kombinacji.

Dążąc do możliwie minimalnej ilości maksymalnie reprezentatywnych kombinacji dla każdej klasy, trzeba rozpocząć proces ich tworzenia od najbardziej skorelowanych neuronów wzorców *SN*, które będą wyznaczały granice możliwości osiągnięcia określonego przez nie stopnia dyskryminacji dla kombinacji je zawierających. Ze względu na to, iż kombinacje w metodzie ASONN tworzone są poprzez ich rozbudowę (poszerzanie zakresów wartości atrybutów porządkowalnych lub powiększanie podzbiorów wartości pozostałych atrybutów) w celu osiągnięcia możliwie dużej ich reprezentatywności, dołączanie trudnych dyskryminacyjnie wzorców, powinno następować jak najwcześniej, gdyż w trakcie rozbudowy kombinacji, stopień dyskryminatywności kombinacji może tylko spadać. Rozbudowa połączeń neuronów kombinacji zachodzi tak długo, jak długo rośnie poziom reprezentatywności kombinacji (tzn. ilość aktywowanych neuronów tej samej klasy), nie obniżając jej dyskryminatywności poniżej określonego minimum. Takie minimum można kontrolować poprzez określenie maksymalnego dopuszczalnego pobudzenia neuronów wzorców powiązanych z innymi neuronami klas. Maksymalne pobudzenie neuronów wzorców jest w tym przypadku przekazywane do powiązanych neuronów klas, których zbyt duże pobudzenie lub aktywacja może hamować procesy plastyczne neuronów kombinacji. Żeby nie przerwać procesów plastyczności neuronu kombinacji za późno, trzeba dodatkowo pobudzać inne neurony klas, niepowiązane z neuronem budowanej kombinacji, w stopniu równym ilorazowi pożądanego stopnia dyskryminacji i ilości neuronów wartości (ilości atrybutów) definiujących neurony wzorców tej kombinacji. Wtedy neuron innej (obcej) klasy osiągnie aktywację w momencie przekroczenia wartości pożądanego stopnia dyskryminacji i np. poprzez połączenie tłumiące (ASUP), wyhamuje plastyczność połączeniową, blokując połączenie takiego neuronu wartości *VN* z neuronami zakresów *RN* lub podzbiorów *SN* neuronu kombinacji *KN*. Można to zrealizować przy pomocy dodatkowych neuronów hamujących sumując i przekazując stosowne pobudzenia i hamowania pomiędzy neuronami oraz określając aktywację lub hamowanie pewnych procesów plastyczności neuronalnej w ich kontekście. Taki sposób działania ujawnia również zasadnicze różnice pomiędzy klasyczną metodologią programowania, a metodologią asocjacyjnych obliczeń. Klasyczna informatyka dostarcza programiście szeroki wachlarz możliwości w zakresie budowy wyrażeń matematycznych, pętli obliczeniowych, rozbudowanych warunków, enkapsulacji itd. Obliczenia asocjacyjne wydają się być bardzo ograniczone w tym zakresie, umożliwiając operowanie na pobudzeniach, hamowaniach i plastyczności, natomiast dzięki temu są uniwersalne i można je zautomatyzować, co potwierdza sposób działania ludzkiego umysłu, w którym wiedza formuje się dzięki programom i funkcjom biologicznych neuronów.

W pierwszym etapie budowy kombinacji dla danej klasy trzeba więc wyznaczyć stopień skorelowania wszystkich wzorców tej klasy z wszystkimi pozostałymi wzorcami innych klas, jak pokazano na rysunku 7.8 dla wzorców klasy *Virginica*. Na tym rysunku każdy z neuronów wzorców, korzystając z mechanizmów asocjacyjnych, wyznacza swoją korelację z wzorcami obcymi. Wartość parametru *outCorr* neuronów wartości oznacza maksymalny stopień korelacji obcej neuronu wzorca klasy *Virginica* z jakimś neuronem wzorca innej klasy. Neurony wartości *VN* bowiem pośredniczą w zbieraniu informacji dla neuronów wzorców *Virginica*, przekazując im korelacje od neuronów innych klas. Parametr *oCR* (*out-correlation*) składa się z jednej albo dwu wartości, z których pierwsza określa maksymalny stopień korelacji obcej tego neuronu wzorca z każdym innym neuronem wzorca definiującym inną klasę. Zaś druga liczba określa ilość w takim stopniu skorelowanych neuronów wzorców definiujących inne klasy. Np. $oCR=2/1x$ dla wzorców definiujących klasę *Virginica* oznacza, że dany neuron wzorca *Virginica* posiada aż 2 identyczne wartości atrybutów z 1 neuronem reprezentującym wzorec innej klasy. Zaś $oCR=1/3x$ oznacza, że dany neuron wzorca posiada tylko 1 identyczną wartość atrybutu, ale aż z 3 neuronami reprezentującymi wzorce innych klas, przy czym dla każdego z tych 3 skorelowanych wzorców ta identyczna wartość może dotyczyć innego atrybutu (parametru). W przypadku neuronów wzorców innych klas (*Setosa* i *Versicolor*) występuje tylko jedna liczba np. $oCR=2$, wskazująca maksymalny stopień

korelacji obcej tego neuronu wzorca z neuronami wzorców rozważanej klasy (*Virginica*). W prosty obliczeniowo sposób można wyznaczyć najbardziej skorelowane neurony wzorców każdej klasy, rozpoczynając od nich następnie rozbudowę sieci połączeń neuronów kombinacji klasyfikatora ASONN.

Z wszystkich neuronów wzorców $SN_{99} - SN_{147}$, reprezentujących wzorce uczące nr $S_{101} - S_{150}$ klasy *Virginica*, najbardziej skorelowanym z wzorcami innych klas okazuje się być neuron SN_{125} , reprezentujący wzorec uczący S_{127} , który posiada $oCR=2/2x$, czyli istnieją 2 neurony wzorców **outSN** definiujące inne klasy, które posiadają równocześnie 2 połączenia z tymi samymi neuronami wartości, z którymi posiada połączenia neuron SN_{125} . Wobec tego neuron SN_{125} stanie się załączkiem pierwszej kombinacji (*combination germ*) dla klasy wzorców *Virginica*. W przypadku czystych rozwiązań asocjacyjnych można stopień korelacji danego neuronu przekształcić na czas aktywacji neurogenezy neuronu kombinacji, żeby neurony najbardziej skorelowane automatycznie inicjowały ten proces jako pierwsze. Tutaj ze względu na brak symulacji równoległości i czasu oraz hybrydowość tego rozwiązania zostanie w liniowym procesie przeszukiwania wybrany ten o maksymalnym stopniu i liczbie korelacji obcych. Ze względu na to, iż wszystkie atrybuty wzorców Iris posiadają porządek liczbowy (reprezentujące pewne wielkości fizyczne), do budowy tej kombinacji zostaną wykorzystane **neurony przedziałów RN** oraz połączenia ASIM i ADEF. Dla atrybutów nieposiadających porządku do budowy kombinacji byłyby wykorzystane **neurony podzbiorów UN**, a neurony wartości VN reprezentujące ich wartości nie byłyby łączone pomiędzy sobą przy pomocy połączeń ASIM, lecz tylko połączeniami ADEF z neuronami wzorców SN . Dodanie do grafu AANG neuronów RN, UN i KN oraz połączeń pomiędzy nimi i neuronami VN, SN i CN jest przykładem rozbudowy grafu AANG pod wpływem danych i postawionego celu, który w ogólności reprezentuje tutaj zbiór potrzeb systemu AAS.

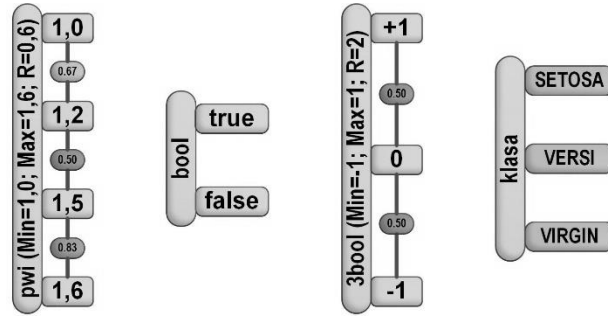
Najpierw zostaną wyznaczone wagi dla połączeń typu ASIM i ADEF grafu AANG, a potem będzie już można rozpocząć budowę klasyfikatora ASONN. Połączenia asocjacyjne typu ASIM łączą wartości atrybutów odwzorowane w VN ach poszczególnych parametrów P reprezentowanych przez sensoryczne elementy wejściowe $SENSIN$ (rys. 7.9). Dla uproszczenia i przyspieszenia obliczeń każdy porządkowalny $SENSIN$ przechowuje wartość minimalną Min^P , maksymalną Max^P oraz wartość zakresu zmienności $Range^P$ (7.2), które są na bieżąco wyznaczane i aktualizowane w trakcie dołączania kolejnych receptorycznych neuronów wartości VN do elementu wejściowego $SENSIN$. Dzięki temu te wartości są zawsze aktualne i dostępne z każdego neuronu wartości VN (*value neuron*). Waga łącząca dwa sąsiednie receptorowe neurony wartości VN , które reprezentują w grafie AANG najbliższe i najbardziej podobne wartości dla określonego parametru P , wyznaczana jest według następującej zależności (7.1):

$$w_{ASIM}^P = \left(\frac{Range^P - |Val - Val_{ASIM}|}{Range^P} \right)^n \quad (1.1)$$

$$Range^P = Max^P - Min^P \quad (1.2)$$

gdzie za n – podstawiamy wartość 1 lub ewentualnie 2 dla nieliniowego wyostżenia stopnia podobieństwa podobnych wartości.

Waga w_{ASIM}^P łącząca neurony wartości VN reprezentujące bliskie (podobne) wartości dla określonego parametru P jest tak skonstruowana, żeby dla wartości skrajnych Min^P i Max^P $w_{ASIM}^P = 0$, odwzorowując w taki sposób zerowe podobieństwo pomiędzy wartościami skrajnymi. To jednak wymusza konieczność aktualizacji wszystkich wag dla $SENSIN$ a, dla którego zmienia się któraś wartość skrajna, co dzieje się w pesymistycznym przypadku tyle razy w trakcie budowy grafu AANG, ile występuje wzorców w zbiorze uczącym. Taka pesymistyczna sytuacja jest jednak bardzo mało prawdopodobna, lecz żeby uniknąć konieczności wielokrotnej aktualizacji wszystkich wag (szczególnie w przypadku klasycznej realizacji niekorzystającej z równoległości obliczeń), można najpierw zbudować cały graf AGDS, a dopiero na końcu w trakcie jego przekształcenia do postaci AANG jednokrotnie wyznaczyć wszystkie wagi na podstawie zależności (7.1). Przykłady różnych połączeń ASIM oraz wartości wag zilustrowano rysunku 7.9. Każdą wagę można więc szybko i łatwo wyznaczyć przy stałej złożoności obliczeniowej, a w przypadku realizacji równoległej wszystkie równocześnie przy stałej złożoności obliczeniowej, zaś w przypadku klasycznych sekwencyjnych obliczeń niekorzystającej z GPU przy liniowej złożoności obliczeniowej. Neurony wartości VN nie są łączone ASIM, gdy nie istnieje relacja porządkująca reprezentowane przez nie wartości (rys. 7.9).



Rys. 7.9 Wagi połączeń ASIM dla $n=1$ w zależności (7.1) dla wartości porządkowalnych, logicznych i symbolicznych.

Następnie wyznaczone są wagi typu ADEF łączące neurony wzorców SN z definiującymi je neuronami wartości VN (rys. 7.10). Neurony SN reprezentują kombinacje (proste semassele) wartości reprezentowanych przez neurony VN . Mogą automatycznie powstawać na skutek procesów neurogenezy wywołanych pod wpływem równoczesnej aktywacji pewnej kombinacji neuronów wartości VN opisanych w rozdziale 6. Neurony wartości VN są różnie skorelowane z wzorcami poszczególnych klas, a więc są w różnym stopniu dyskryminatywne dla definiowanych przez nie wzorców SN . Z tego też powodu powinny mieć różne wartości wag $w_{VN,SN}$, odwzorowując ich reprezentatywność dla wzorców klasy neuronu wzorca SN , dla którego waga jest wyznaczana. Dzięki takiemu podejściu na wynik klasyfikacji większy wpływ będą miały neurony wartości VN , które mają mniej połączeń z SN ami reprezentującymi wzorce innych klas. Wartości wag $w_{ADEF}^{VN,SN}$ i $w_{ADEF}^{SN,VN}$ wzajemnych połączeń heterogenicznych ADEF dla takich połączeń wyznaczamy jako:

$$w_{VN,SN}^{ADEF} = \frac{\frac{\|\{\overline{SN}:VN \leftrightarrow \overline{SN} \leftrightarrow CN\}\|}{\|\{\overline{SN}:VN \leftrightarrow \overline{SN}\}\|}}{\sum_{VN_i \leftrightarrow SN} \frac{\|\{\overline{SN}:VN_i \leftrightarrow \overline{SN} \leftrightarrow CN\}\|}{\|\{\overline{SN}:VN_i \leftrightarrow \overline{SN}\}\|}} \quad (1.3.)$$

$$w_{SN,VN}^{ADEF} = 1 \quad (1.4.)$$

Na rysunku 7.10 uwidoczniło wagi ADEF wyznaczone dla 13-elementowego zbioru wzorców Iris z rysunku 7.3. Kolorem czerwonym wyróżniono VN y i ich połączenia typu ADEF pomiędzy najbardziej negatywnie skorelowanymi SN ami różnych klas poprzez łączące je VN y reprezentujące ich wspólne, identyczne wartości atrybutów.

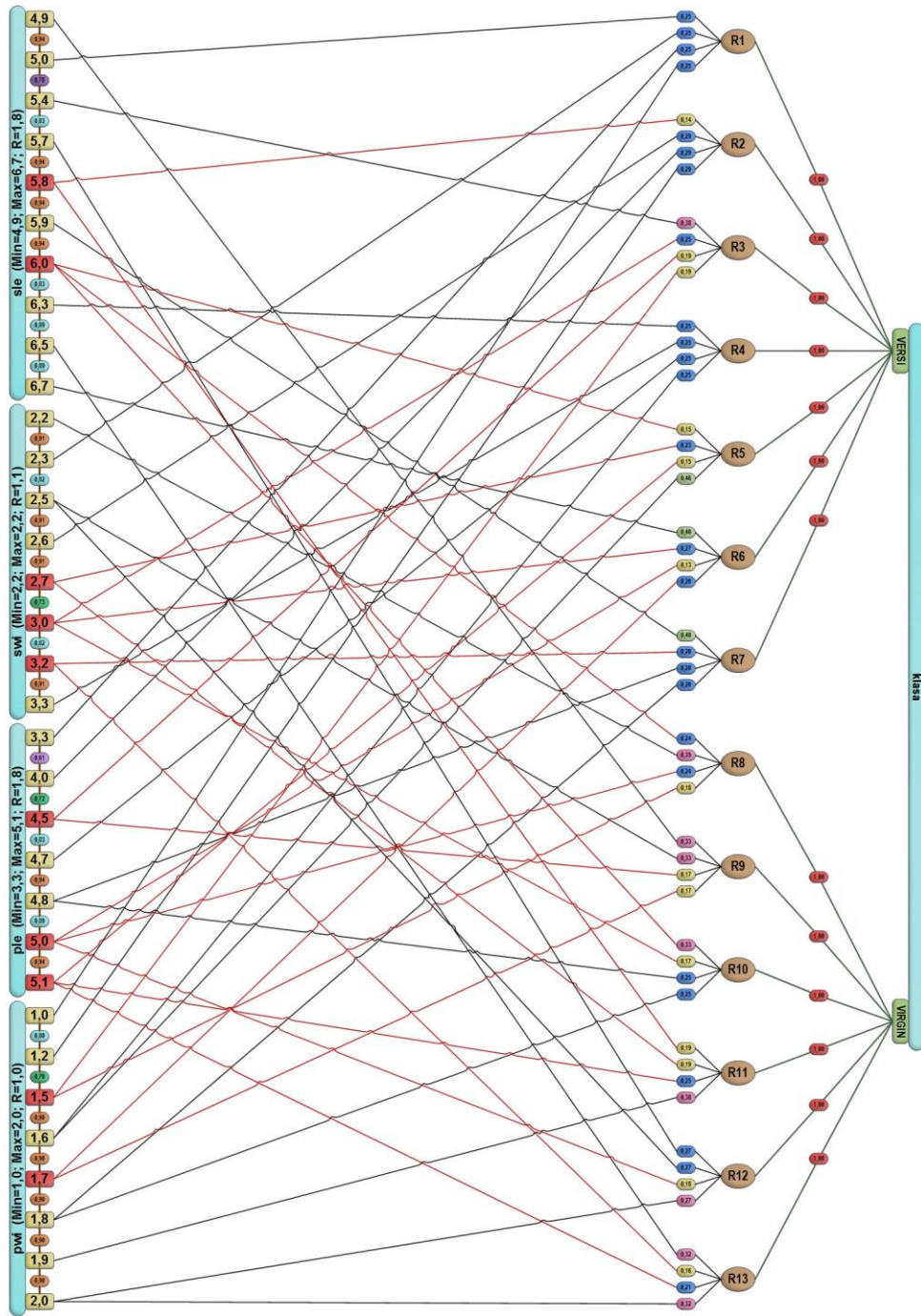
Klasy C reprezentowane są przez neurony klas CN , które połączone są ADEF z wszystkimi neuronami wzorców SN , które reprezentują wzorce S należące do tych klas C . Wagi tych połączeń określone są tak, iż aktywacja dowolnego neuronu wzorca SN powoduje aktywację neuronu klasy CN (7.5), a aktywacja neuronu klasy CN powoduje aktywację wszystkich neuronów wzorców SN (7.6). A więc klasa wywoływana jest przez każdego jej reprezentanta, zaś klasa wywołuje wszystkich swoich reprezentantów. W ogólności te wagi można byłoby zróżnicować na podstawie np. częstości występowania (powtarzalności) prezentacji poszczególnych wzorców uczących, lecz ze względu na specyficzną postać zbiorów uczących stosowanych w inteligencji obliczeniowej taką dywersyfikację nie ma na podstawie czego sensownie przeprowadzić.

$$w_{SN,CN}^{ADEF} = 1 \quad (1.5.)$$

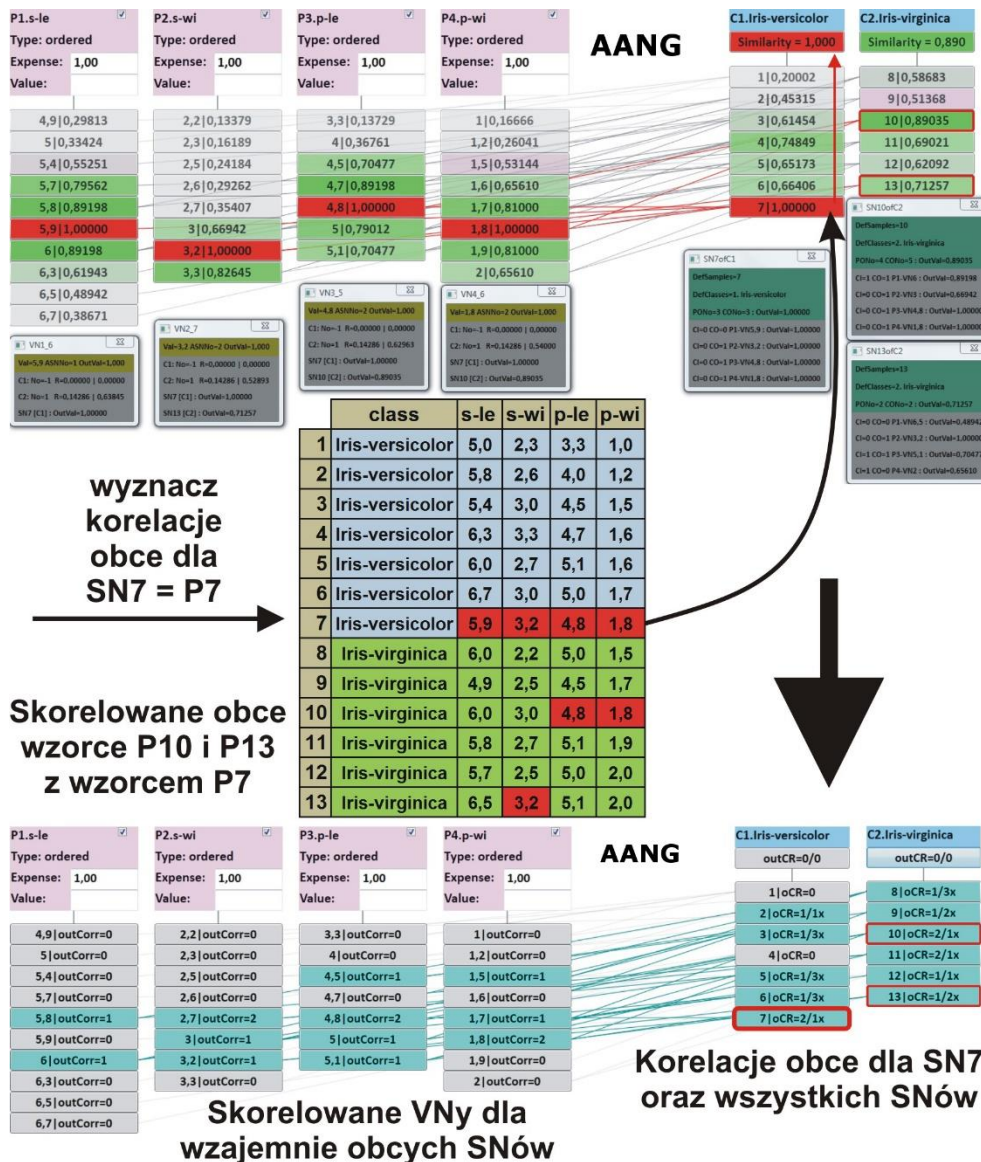
$$w_{CN,SN}^{ADEF} = 1 \quad (1.6.)$$

Progi aktywacji wszystkich neuronów wykorzystywanych w konstrukcji ASONN są niezmiennie i zawsze równe jedności:

$$\theta = 1 \quad (1.7.)$$



Rys. 7.10 Struktura grafu AANG dla 13 wzorców uczących Iris z rysunku 7.3 wraz z wyznaczonymi wagami ADEF (7.3-7.6) oraz wagami ASIM (7.1). Kolorem czerwonym zaznaczono najbardziej negatywnie skorelowane VN_y oraz ich połączenia, które mają istotny wpływ na wynik dyskryminacji.



Rys. 7.11 Etapy wyznaczania maksymalnej dyskryminacji obcej dla wybranego neuronu wzorca na przykładzie 13 elementowego zbioru danych uczących Iris z rysunku 7.3.

Wykorzystanie grafu AANG do odnalezienia dyskryminatywno-reprezentatywnych kombinacji przedziałów i podzbiorów wartości atrybutów reprezentowanych przez neurony wartości VN powoduje stopniową rozbudowę tego grafu o nowe elementy KN, RN, UN oraz nowe połączenia pomiędzy nimi i wcześniej utworzonymi neuronami VN, SN i CN w trakcie transformacji struktury AGDS do grafu AANG. Rozbudowa grafu AANG odbywa się następująco: Najpierw dla każdego neuronu wzorca SN w grafie AANG obliczany jest stopień jego korelacji obcej, tzn. z wszystkimi pozostałymi neuronami wzorców innych klas. W tym celu np. neuron SN_7 (rys. 7.11) poprzez połączenia ADEF aktywuje neuron klasy CN_1 , którą definiuje, oraz wszystkie definiujące go neurony wartości $VN_{5,9}$, $VN_{3,2}$, $VN_{4,8}$, $VN_{1,8}$, z którymi jest również powiązany połączeniami ADEF. Neuron klasy CN_1 aktywuje i blokuje (refrakcją) wszystkie neurony wzorców tej samej klasy. Natomiast każdy aktywowany neuron wartości VN pobudza wszystkie powiązane z nim neurony **obcych wzorców** $outSN$, a więc np. $outSN_{10}$ i $outSN_{13}$ z rysunku 7.11, w stopniu wynikającym z jego negatywnej korelacji z nimi, zależnej od ilości powiązań z tymi neuronami, np. dwa dla $outSN_{10}$ i jedno dla $outSN_{13}$. W zastosowanym tutaj hybrydowym modelu zamiast blokowania neuronów tej samej klasy refrakcją zastosowano sprawdzenie połączenia rozważanego neuronu wzorca z klasą CN_1 . Dla uproszczenia można też przyjąć, iż jeżeli neuron innej klasy $outSN$ jest powiązany z aktywowanym neuronem wartości VN, wtedy jego licznik korelacji obcych jest inkrementowany, a następnie przesyłany z powrotem do wszystkich powiązanych z nim neuronów wartości VN. Neurony wartości VN wyznaczają na tej podstawie maksimum ze stopni obcych korelacji (tutaj: $outCorr = 2$ na rysunku 7.11) wszystkich powiązanych z nimi obcych neuronów wzorców $outSN$ oraz licznosc takich obcych korelacji o maksymalnym stopniu. W dalszej kolejności aktywowane na początku neurony wartości VN przekazują wstecz informacje o

stopniu i liczności korelacji obcych do neuronu wzorca SN ($oCR = 2/1$ na rysunku 7.11), który zainicjował ten proces. Neuron ten wyznacza swój stopień i licznosc korelacji obcych na podstawie maksymalnej wartosci i licznosci otrzymanych przez definiujace go neurony wartosci VN . W modelu bardziej zaawansowanym mozna wyznaczać stopień korelacji obcej SN ów na podstawie wszystkich obcych korelacji, a nie tylko tych o najwyższym stopniu. Ponadto mozna uwzględnic również pobudzenia pochodzace od sąsiednich VN ów pobudzonych przez aktywowane VNy . Bardziej zaawansowane modele najlepiej realizowac jednak w sposob równolegly.

Ze względu na zastosowany tutaj model hybrydowy mozna przekazywanie pewnych informacji uprościć, nie modelujac wszystkiego w kategoriach pobudzeń, hamowan, aktywacji i refrakcji biologicznych neuronów. Korzystajac ze specyficznej struktury polaczen asocjacyjnych AGDS mozliwe staje się określenie maksymalnej korelacji obcej kazdego neuronu wzorca SN przy niewielkiej zlozonosci obliczeniowej. Jesli dysponujemy mozliwoscia wykorzystania GPU, wtedy caly ten proces zachodzi zawsze przy stalej zlozonosci obliczeniowej dla pojedynczego SNa oraz przy liniowej zlozonosci dla wszystkich SN ów. Stopień korelacji obcej kazdego neuronu wzorca SN jest wyznaczany jednokrotnie dla calogo grafu AANG na poczatku procesu budowy klasyfikatora $ASONN$.

Następnie dla kazdego neuronu klasy CN w grafie AANG nalezy kolejno uruchomic proces tworzenia kolejnych neuronów kombinacji, który bedzie trwal tak dlugo, dopoki wszystkie niezahibernowane neurony wzorców nie zostana reprezentowane przez ktoraś z tworzonych kombinacji. To zapewni w 100% poprawna klasyfikacje wszystkich niesprzecznych wzorców uczacych. Kazda nowa kombinacja reprezentowana przez neuron KN tworzona jest od pewnego zalazkowego neuronu wzorca ***germSN*** (*combination germ*) wyznaczonego spozród pozostalych, najbardziej negatywnie skorelowanych SN ów, które jeszcze nie są reprezentowane w zadnej kombinacji. Neuron *germSN* zostanie polaczony z neuronem kombinacji KN polaczeniem ADEF, a definiujace go neurony wartosci VN zostana polaczone z nowo utworzonymi neuronami przedzialów RN i podzbiorów UN , które zostana również polaczone ADEF z neuronem KN . W celu określenia *germSN* neuron kazdej klasy CN wysyla (mozliwie równolegle) zapytanie do wszystkich powiazanych z nim neuronów SN o podanie jego stopnia i licznosci korelacji obcej. W neuronowym modelu równoleglym neurony SN moga odpowiadac aktywacja po czasie (7.8), określonym przez ich stopień i licznosc korelacji obcej, tzn. szybciej reaguja te bardziej skorelowane, a wtedy CN uruchomi dla nich proces budowy kombinacji, a pozostale aktywacje od innych SN ów zignoruje. Mozna tez wykorzystac refrakcyjnosc do wyłączenia neuronów juz reprezentowanych w ktorej kombinacji poprzez ich aktywacje za posrednictwem powiazan ADEF z neuronami KN , które aktywujemy bezposrednio przed uruchomieniem procesu wyznaczania neuronu *germSN*. Mozna tez wykorzystac tłumienie ASUP za posrednictwem dodatkowych neuronów hamujacych. Znalezienie neuronu zalazkowego *germSN* dla nowej kombinacji ma wiec liniowa zlozonosc obliczeniowa w hybrydowym modelu oraz stala w modelu równoleglym.

$$t_{correl}(degree, no) = \frac{1}{no \cdot degree^2} \quad (1.8.)$$

Następnie rozpoczynany jest proces poszerzania kombinacji reprezentowanej przez KN o kolejne neurony wartosci VN , umożliwiajace jej reprezentacje kolejnych **neuronów wzorców własnych *inSN***, tzn. takich SN ów, które definiuja ta sama klase, dla ktorej budowana jest ta kombinacja. Zeby jakis SN byl reprezentowany przez KN , wszystkie polaczone z nia RNy i UNy musza być równocześnie polaczone z $VNami$ definiujacymi ten SN . Kombinacja jednak nie musi być zdefiniowana przez wartosci wszystkich atrybutów (parametrów) dostepnych w zbiorze uczacych, lecz tylko przez te najbardziej reprezentatywne i zapewniajace jej mozliwie wysoki stopień dyskryminatywnosci dla wszystkich reprezentowanych przez nia SN ów. Rozbudowa kombinacji o kolejne VNy moze spowodowac obnizenie stopnia dyskryminatywnosci calej kombinacji, ale moze tez spowodowac zwiekszenie ilosci reprezentowanych przez nia wzorców. Obydwa te procesy nie musza jednak nastepowac równocześnie i w równym stopniu. Zalezy to w duzym stopniu od ilosci i stopnia obcych korelacji pomiedzy wzorcami reprezentowanej klasy, a wzorcami pozostalych klas. Im wiecej korelacji obcych, tym trudniej bedzie poszerzac kombinacje.

Kolejna trudnoscia jest mnogość mozliwosci rozbudowy kazdej takiej kombinacji. Kombinacje tworzone są stopniowo, poczynajac od jednowartościowych przedzialów lub jednoelementowych podzbiorów określonych na podstawie wartosci atrybutów *germSNa*. Dla kazdego atrybutu o porzadkowalnych wartosciach, dla ktorego istnieja asocjacyjne polaczenia typu ASIM pomiedzy reprezentujacymi go neuronami wartosci VN , przedzial definiujacy kombinacje mozna poszerzac w dwuch kierunkach. Dla przedzialów reprezentowanych przez neurony RN nie ma tez mozliwosci opuszczenia jakiegokolwiek wartosci reprezentowanej przez VN , nawet jesliby definiowala obcy neuron wzorca *outSN* i niekorzystnie wplywala na stopień dyskryminatywnosci kombinacji. Dla parametrów o wartosciach symbolicznych polaczenia typu ASIM nie istnieja, wiec rozbudowa kombinacji zwiazana jest z powiekszaniem podzbiorów neuronów wartosci VN powiazanych z neuronem UN . To oznacza, iz potencjalnie kazdy czesciowo reprezentowany SN w KN poprzez swoje dotad niereprezentowane neurony wartosci VN wyznacza potencjalny kierunek rozwoju tego UNa i KNa , a wiec mozliwosci jest jeszcze wiecej. W ogolnoscii na gruncie klasycznej informatyki nalezaloby rozwazyć wszystkie mozliwe kombinacje w kazdym kroku takiej

rozbudowy z możliwością np. rekurencyjnego wycofywania się z nich, gdyż z góry nie wiadomo, który kierunek rozbudowy okaże się w kolejnych krokach właściwy i najbardziej efektywny. Ponadto wybór jednego VNa często ogranicza możliwość wyboru innego VNa w kolejnym kroku pod rygorem obniżenia stopnia dyskryminatywności całej kombinacji, a więc przeszukiwanie wszystkich możliwości „na ślepo” i szukanie optymalnego rozwiązania miałyby złożoność wykładniczą $O(n!)$. Korzystając z AANGów można jednak wykorzystać ich wiedzę o zbiorze danych ukrytą w parametrach i strukturze połączeń pomiędzy neuronami VN i SN oraz możliwość pobudzania i aktywacji neuronów, które mogą szybko ujawnić najbardziej korzystne możliwości względem aktualnego stanu rozwoju kombinacji KN .

Celem tworzenia takich kombinacji jest reprezentowanie jak największej ilości wzorców jednej klasy przy możliwie wysokim stopniu ich dyskryminacji. Najkorzystniejsze z punktu widzenia kombinacji będzie więc dodawanie takich VN ów definiujących $inSNy$, które są już częściowo reprezentowane w kombinacji, gdyż dla osiągnięcia ich reprezentacji wystarczy dodać mniejszą ilość brakujących VN ów. Oczywiście w pierwszym rzędzie warto rozważyć te $inSNy$, którym brakuje najmniej VN ów. Gdyby dodawanie VN ów nie było związane z możliwością obniżenia stopnia dyskryminacji lub ograniczania możliwości dla innych VN ów, wtedy można byłoby to w miarę prosto zrealizować korzystając z podstawowej struktury AGDS grafów AANGów. Niestety wiele VN ów reprezentuje nie tylko $inSNy$, lecz również $outSNy$. Grafy AANG zawierają jednak wbudowaną możliwość równoczesnego pobudzania neuronów powiązanych z istniejącymi w kombinacji $VNami$ w stosunku do możliwości tylko pasywnego przeglądania struktury AGDS. Istniejące VNy mogą pobudzać $inSNy$ i $outSNy$ z nimi powiązane, zaś $inSNy$ częściowo reprezentowane mogą szybko sprawdzić, czy dodanie ich brakujących VN ów spowoduje obniżenie się stopnia dyskryminacji poprzez ich próbną aktywację nie powodującą jeszcze plastyczności i nawiązanie nowych połączeń kombinacji. Taka próbną aktywacja może pobudzić $outSNy$ w stopniu przekraczającym pożądaną stopień dyskryminatywności kombinacji, co oznacza, iż dodanie tego $inSNa$ jest niemożliwe bez obniżenia tego stopnia. Warto więc w pierwszej kolejności uzupełnić kombinację o te $inSNy$, które nie powodują zmniejszenia się stopnia jej dyskryminatywności oraz w możliwie minimalnym stopniu zmuszają do reprezentacji VN ów powiązanych z najbardziej negatywnie skorelowanymi $outSNami$. Takie $inSNy$ można automatycznie wyznaczyć, sprawdzając równocześnie reakcję wszystkich $outSN$ ów za pośrednictwem powiązanych z nimi CN ów. Gwarantuje to możliwość dodawania kolejnych VN ów w kolejnych próbnych aktywacjach i osiągnięcie możliwie dużego poziomu reprezentatywności kombinacji bez obniżania jej stopnia dyskryminatywności albo zrobienia tego w minimalnym możliwym i akceptowalnym stopniu, jeśli nie ma możliwości dalszej rozbudowy kombinacji bez obniżenia tego stopnia, który jest nadal wysoki. W przypadku realizacji równoległej można równocześnie i w czasie stałym określić wszystkie potrzebne zależności i na tej podstawie wywołać plastyczność kombinacji warunkowo blokowanej przez nadmierne pobudzenie któregośkolwiek $outSNa$ za pośrednictwem CNa , z którym jest powiązany. Połączenia asocjacyjne ADEF pomiędzy neuronami wartości i neuronami wzorców umożliwiają więc prawie natychmiastowe zdobycie informacji o najbardziej pożądanym neuronach VN , które w pierwszej kolejności powinny być dodane z punktu widzenia rozwoju kombinacji i zwiększenia jej reprezentatywności, gdyż najlepiej „kojarzą się” z istniejącą kombinacją reprezentowaną przez KN .

Podobnie od razu można zdobyć informację o neuronach SN , których pobudzenie poprzez reprezentowane w kombinacji VNy od razu ujawnia ich stopień reprezentacji w KN . Informacje te można zdobyć poprzez aktywowanie neuronu kombinacji KN , który równocześnie aktywuje wszystkie powiązane z nim neurony przedziałów RN i neurony podzbiorów UN , a te następnie równocześnie aktywują wszystkie powiązane z nimi neurony VN , które następnie za pośrednictwem połączeń ADEF aktywują lub pobudzą w odpowiednim stopniu neurony SN w zależności od ich aktualnego stopnia reprezentacji w kombinacji. Cały ten proces odbywa się w czasie stałym przy założeniu równoległej realizacji. Realizowana jest więc pewna sekwencja skojarzeń wywołana aktywacją wybranego neuronu KN , która równocześnie dostarcza wszystkie informacje o aktualnym stopniu korelacji wszystkich SN ów grafu AANG z reprezentowaną przez ten KN kombinacją.

W przypadku modelowania czasu i relaksacji neuronów oraz kilkukrotnej aktywacji KNa , najbardziej skorelowane $inSNy$ osiągną najszybciej aktywację i mogą wywołać proces swojej plastyczności połączeniowej z niedawno intensywnie aktywowanym neuronem KN , a aktywowane dzięki temu receptoryczne neurony VN plastyczność z receptorycznym neuronem UN lub RN . Odpowiada to biologicznemu modelowi plastyczności i jest zgodne z logiką powiększania kombinacji. W ludzkim procesie myślowym również pobudzane są wielokrotnie te same neurony, które mają przywołać pożądane skojarzenia. Modelowanie tego zjawiska zostanie jednak opisane dopiero w następnym rozdziale opisującym formowanie się wiedzy w AASach oraz procesy wywoływania skojarzeń.

W przedstawionym hybrydowym modelu nie zastosowano symulacji czasu ani relaksacji, więc decyzje o rozbudowie kombinacji są podejmowane na podstawie stopnia pobudzenia poszczególnych $inSN$ ów po jednokrotnej aktywacji KNa . Najbardziej pobudzone, lecz nieaktywowane neurony są najlepszymi kandydatami do włączenia ich do kombinacji pod warunkiem, że dodanie brakujących do ich reprezentacji neuronów wartości nie spowoduje nadmiernej utraty stopnia dyskryminatywności całej kombinacji. Jednak graf AANG dysponuje również wiedzą o tym, gdyż z tymi kandydującymi neuronami wzorców $inSN$ bezpośrednio są połączone

wszystkie definiujące je neurony wartości VN , zaś z wszystkimi neuronami wzorców obcych $outSN$, które są częściowo reprezentowane w tej kombinacji również są bezpośrednio połączone wszystkie definiujące je neurony wartości VN . A więc jeśli wszystkie niereprezentowane jeszcze neurony wzorców własnych $inSN$ (tzn. pobudzone, lecz nieaktywowane) pobudzą definiujące je neurony wartości VN w stopniu, jaki wynika z ich pobudzenia (przekazą im swoje pobudzenie), oraz równocześnie wszystkie neurony wzorców obcych $outSN$ będą tłumić (hamować) pobudzenia tych VN ów w stopniu, jaki wynika z ich pobudzenia, wtedy najbardziej pobudzone i niewyhamowane zostaną te neurony wartości VN , które są najlepszymi kandydatami do włączenia ich do kombinacji za pośrednictwem ich połączenia z odpowiednim neuronem RN lub UN . Jeśliby to pobudzanie i hamowanie neuronów wartości VN poprzez neurony wzorców SN zachodziło w czasie, wtedy spowoduje najszybszą aktywację neuronów wartości VN , które są najlepszymi kandydatami do rozbudowy kombinacji. Implementacja równoległa tego schematu skojarzeń spowoduje wycenę poziomu pozytywnego wpływu na rozwój kombinacji wszystkich potencjalnych kandydatów spośród wszystkich neuronów VN w czasie stałym.

W przypadku parametrów symbolicznych wystarczy wybrać najbardziej pobudzony neuron wartości VN i dołączyć go do kombinacji, a następnie cały proces powtarzać dopóki najbardziej pobudzony neuron wartości VN przekracza określony próg minimalnej dyskryminacji determinujący osiągnięcie pożądanego stopnia dyskryminacji wszystkich SN ów tej kombinacji. Jeśli żaden spośród neuronów wartości nie osiągnie odpowiedniego poziomu pobudzenia na skutek równoważących go hamowań docierających od $outSN$ ów, oznacza to, że nie można dalej rozbudowywać kombinacji bez obniżenia obecnego jej stopnia dyskryminatywności. Proces rozbudowy kombinacji trzeba zakończyć albo pozwolić na obniżenie jej obecnego stopnia dyskryminatywności pod warunkiem, że jest on większy od jedności, gdyż w odwrotnym razie spowodowałoby to utratę możliwości dyskryminacji wzorców tej kombinacji i nie dałoby 100% jednoznacznego wyniku klasyfikacji.

W przypadku parametrów porządkowalnych poszukujemy przedziałów, a nie podzbiorów neuronów wartości VN , więc nie możemy w taki sposób dołączyć do kombinacji najlepszego kandydata, jeśli nie znajduje się on na granicy dotychczasowego przedziału kombinacji. Wtedy potrzebne jest jeszcze pobudzenie powiązanych z wszystkimi lub potencjalnie najlepszymi kandydującymi (pobudzonymi) neuronami wartości ich sąsiadów poprzez asocjacyjne połączenia $ASIM$ aż do granic obecnych przedziałów ich wartości. Tutaj niestety najlepsi kandydaci ze względu na ich możliwe oddalenie od granic przedziałów mogą się okazać wcale nie być tymi najlepszymi. Mogą nawet zupełnie utracić status kandydatów ze względu na dużą ilość pośredniczących neuronów VN reprezentujących dużą ilość wartości $outSN$ ów, które mogą zupełnie wyhamować pobudzenie nawet najlepszych zbyt odległych kandydatów, powodując ich odrzucenie. Przekazywane pobudzenia są wazone, a więc osłabiane względem wagi tych połączeń. W trakcie przekazywania tych pobudzeń są one sumowane ze stanami pobudzeń i hamowań pośredniczących neuronów VN na drodze do granic wyznaczonych przez wartość minimalną lub maksymalną RNa , a więc potencjalnie również z innymi kandydatami. Najlepszym kandydatem będzie wtedy ten brzegowy neuron wartości VN istniejącego przedziału RN , który osiągnie najmocniejsze pobudzenie. Jeśli żaden neuron nie osiągnie pobudzenia, oznacza to, że proces poszerzania przedziałów kombinacji trzeba zakończyć albo umożliwić obniżenie jej stopnia dyskryminatywności. Poszerzenie zakresu RN ów o kolejne VNy zawsze trzeba robić stopniowo, tzn. po dodaniu jednego VNa o najmocniejszym pobudzeniu poza granicą obecnych RN ów należy dokonać aktualizacji stopnia pobudzenia wszystkich VN ów połączonych za pośrednictwem wszystkich SN ów powiązanych z właśnie dodanym $VNem$ do RNa . Jest to szczególnie ważne w przypadku, gdy dodany VN połączony jest z jakimś $outSNem$, co może spowodować zwiększenie siły hamowania niektórych VN ów, które mogłyby być dobrymi kandydatami przed dodaniem tego VNa . Również połączone $inSNy$ z właśnie dodanym $VNem$ mogą dzięki niemu zwiększyć pobudzenie pozostałych powiązanych z nim VN ów niereprezentowanych jeszcze w kombinacji, a więc ich aktualizacja też jest konieczna, gdyż może spowodować wybranie innego lepszego, granicznego VNa do dalszej rozbudowy zaktualizowanych RN ów po dodaniu ostatniego VNa .

Z powyższego opisu wynika, że cały proces selekcji kandydata do rozbudowy kombinacji realizowany jest poprzez pewną ilość pobudzeń, z których większość może odbywać się równoległe. Osiągnięcie tego możliwe jest dzięki wiedzy grafu $AANG$ uformowanej w postaci odpowiednich połączeń, ich wag oraz wynikowych stanów pobudzeń neuronów. Wiedza ta dynamicznie zmienia się z dodaniem każdego nowego VNa do kombinacji, gdyż dodanie nowych połączeń, zmienia poziomy pobudzeń i hamowań części neuronów, a więc stopniowo zmieniają się również kolejne skojarzenia zachodzące w stopniowo rozbudowującym się grafie $AANG$. Zmieniają się więc również jego wnioski na temat potencjalnych kandydatów VN mogące być wykorzystane do rozbudowy kombinacji, a więc również całego grafu zawierającego je. Zmiana skojarzeń oznacza zmianę sposobu działania grafu $AANG$, a więc odpowiada jego samoistnemu przeprogramowaniu się pod wpływem postawionego mu celu w postaci tworzenia dyskryminatywno-reprezentatywnych kombinacji, inicjowanych poprzez najbardziej negatywnie skorelowane $germSNy$. W porównaniu z możliwymi rozwiązaniami tego problemu z wykorzystaniem klasycznych algorytmów operujących na tabelach pasywnych danych **rozwiązanie asocjacyjne korzystające z aktywnej i dynamicznie zmiennej wiedzy grafu $AANG$ i jego plastyczności** redukuje złożoność obliczeniową z NP trudnego sprawdzania wszystkich kombinacji do zazwyczaj liniowej związanej z dodawaniem kolejnych

potencjalnie najlepszych kandydatów rozbudowujących kombinacje po przeprowadzeniu zwykle kilku równoległych skojarzeń.

W przypadku niekorzystania z równoległości i w związku ze stosowaną tutaj hybrydyzacją obliczeń istnieje też możliwość określenia kierunku poszerzania przedziałów poprzez badanie stopnia pobudzenia kolejnych sąsiadujących z granicą przedziałów neuronów wartości VN , co trochę upraszcza cały proces propagacji pobudzeń przez wszystkie neurony wartości, a następnie wybierany jest kierunek najlepiej rokujący. Trudnością jest to, iż kierunek rozwoju danego przedziału nie można określić tylko na podstawie granicznych, najbliższych sąsiednich neuronów wartości VN , gdyż najbliższe neurony mogą nie być zupełnie wartościowe z punktu widzenia podnoszenia reprezentatywności kombinacji, a nawet mogą być niekorzystne z punktu widzenia obniżenia jej stopnia dyskryminatywności. Mimo wszystko czasami właśnie takie VN y trzeba dołączyć do przedziału, żeby następnie dojść do kolejnych neuronów VN , które taką reprezentatywność podniosą przy akceptowalnym spadku dyskryminatywności kombinacji. Decyzja o wybraniu kierunku rozwoju każdego przedziału kombinacji zależna jest od wszystkich neuronów wartości definiujących wzorce klasy, którą reprezentuje ta kombinacja. Neurony wartości VN muszą więc na podstawie swoich powiązań z jeszcze niereprezentowanymi neuronami wzorców $inSN$ tej klasy określić swoją potencjalną przydatność dla rozbudowy kombinacji na podstawie częściowej reprezentacji powiązanych z nimi neuronów wzorców w kombinacji i informację o tym przekazać neuronom przedziałów RN . Ich przydatność jest określona przez ilość już reprezentowanych wartości w kombinacji dla powiązanych z nimi $inSN$ ów, a więc każdy $inSN$, częściowo reprezentowany w kombinacji może pobudzić taki VN swoją aktualną wartością reprezentacji lub prościej – przekazać taką wartość bezpośrednio. Z kolei każdy $inSN$ zna swoją aktualną reprezentację, gdyż pobudzają go obecne w kombinacji VN y. Wszystko więc można zrealizować poprzez odpowiednie pobudzenia albo bezpośrednio przez przekazywanie ilości korelacji za pośrednictwem zmiennych w hybrydowym modelu obliczeniowym. Informacja o stopniu obcych i własnych korelacji neuronów wzorców dla kombinacji jest na bieżąco aktualizowana po dodaniu każdego kolejnego VNa we wszystkich powiązanych z nim neuronach. To sprawia, iż nie trzeba od początku wyznaczać wszystkich pobudzeń i hamowań, a neurony wartości VN mogą cały czas przekazywać informację o swojej aktualnej przydatności dla rozbudowy kombinacji.

Proces rozbudowy kombinacji oraz proces dodawania do niej kolejnych wartości atrybutów reprezentowanych przez neurony VN zostanie zilustrowany na przykładzie 13-elementowego zbioru uczącego Iris zawierającego wzorce uczące dwóch klas z rysunku 7.3. Rozpoczyna się on od neuronu wzorca $germSN_7$, który został określony jako najbardziej negatywnie skorelowany z wzorcami drugiej klasy. Pierwsza kombinacja reprezentowana jest w grafie AANG przez neuron kombinacji KN_1 (rys. 7.12), który jest inicjalnie łączony tylko z neuronem wzorca $germSN_7$, neuronem jego klasy $VERSI$ oraz czterema neuronami przedziałów RN_1, \dots, RN_4 . Neurony przedziałów inicjalnie są łączone tylko z neuronami wartości VN definiującymi neuron wzorca $germSN_7$. Dla każdego nowego neuronu kombinacji KN wyznaczany jest jego maksymalnie osiągalny stopień dyskryminatywności, który jest równy różnicy łącznej ilości RN ów i UN ów kombinacji oraz maksymalnej ilości korelacji obcych $germSN$ a. Neuron $germSN_7$ jest negatywnie skorelowany z obcym neuronem $outSN_{10}$ w stopniu równym 2, gdyż definiujące go neurony wartości $VN_{4,8}, VN_{1,8}$ równocześnie definiują ten obcy neuron (rys. 7.12). Wobec tego maksymalny inicjalny stopień dyskryminatywności neuronu kombinacji KN_1 równy jest 2 ($maxDiscrim_{KN_1} = 2$), gdyż od ilości wszystkich RN ów równą 4 odejmujemy maksymalną ilość wspólnych neuronów wartości VN liczoną po wszystkich $outSN$ ach, równą 2 zgodnie z następującą zależnością:

$$maxDiscrim_{KN} = \|\{RN: RN \leftrightarrow KN \leftrightarrow CN\} \cup \{UN: UN \leftrightarrow KN \leftrightarrow CN\}\| - \max_{outSN \leftrightarrow CN} \|\{outSN \leftrightarrow VN \leftrightarrow germSN\}\| \quad (1.9.)$$

gdzie

- $\|\dots\|$ – licznosc zbioru $\{\dots\}$;
- \leftrightarrow – relacja dwuargumentowa określająca dwustronne heterogeniczne asocjacyjne połączenie dwóch neuronów;
- \leftrightarrow – relacja dwuargumentowa określająca brak jakiegokolwiek asocjacyjnego połączenia pomiędzy dwoma neuronami;
- RN – neuron przedziału definiujący kombinację KN ;
- UN – neuron podzbioru definiujący kombinację KN ;
- KN – neuron kombinacji;
- CN – neuron klasy definiowanej przez neuron kombinacji KN oraz $germSN$;
- $germSN$ – neuron załączkowy nowej kombinacji KN ;
- $outSN$ – obcy neuron wzorca definiujący inną klasę niż neuron $germSN$ oraz neuron kombinacji KN , tzn. nie istnieje połączenie pomiędzy $outSN$ i CN , oznaczone jako $outSN \leftrightarrow CN$.

Dla zapewnienia jednoznacznej klasyfikacji stopień dyskryminatywności każdej kombinacji musi być większy lub równy jeden. W tym przypadku warunek ten jest spełniony, a nawet istnieje pewien zakres swobody pozwalający zdecydować, czy tworzona kombinacja będzie dążyła do zachowania stopnia dyskryminatywności

równego 2, czy też będzie musiała go obniżyć do wartości 1 dla uzyskania jej lepszej reprezentatywności. Generalnie, stopień dyskryminatywności kombinacji warto obniżać tylko wtedy, gdy reprezentatywność wzorców kombinacji jest niewielka. A więc jeśli nie osiągnięto sensownego jej poziomu, warto umożliwić stopniowe obniżanie stopnia dyskryminatywności, gdyż umożliwia to dołączanie kolejnych VN ów do kombinacji. Dzięki nim zazwyczaj dochodzi do reprezentacji kolejnych $inSN$ ów, a to automatycznie wpływa na poprawę reprezentatywności tej kombinacji poszerzając wewnętrzny obszar rozmytego hiperprostokądnianu obejmującego wzorce tej klasy. Stopień dyskryminatywności każdej kombinacji większy lub równy jeden gwarantuje zawsze w 100% poprawną klasyfikację wszystkich niesprzecznych wzorców uczących, które reprezentuje. Natomiast im wyższy jest ten stopień od jedności przy równocześnie dużej reprezentatywności poszczególnych kombinacji, tym lepsze wyniki uogólniania klasyfikator zapewni. Warto więc ostrożnie obniżać dyskryminatywność poszczególnych kombinacji, które zresztą nie muszą mieć równych stopni dyskryminatywności.

Następnie $germSN$ kombinacji uruchamia (poprzez stosowne pobudzenie) w każdym, definiującym go neuronie wartości VN proces aktualizacji liczników własnych i obcych korelacji z klasą reprezentowaną przez tą kombinację w zależności od tego, ile jest z nim powiązanych neuronów $inSN$ czy $outSN$. Poprzez **korelację własną neuronu wartości VN z neuronem klasy CN** będzie nazywana ilość połączonych z nim neuronów wzorców $inSN$ połączonych z neuronem klasy CN . Poprzez **obcą korelację neuronu wartości VN z neuronem klasy CN** będzie nazywana ilość połączonych z nim neuronów wzorców $outSN$, które nie są połączone z neuronem klasy CN . Poprzez **korelację własną neuronu wzorca $inSN$ z neuronem kombinacji KN** będzie nazywana ilość VN ów równocześnie połączonych z neuronem wzorca $inSN$, jak również neuronem kombinacji KN za pośrednictwem RN ów lub UN ów przy założeniu, że $inSN$ oraz KN są połączone z tym samym neuronem klasy CN ($inSN \leftrightarrow CN \leftrightarrow KN$). Poprzez **korelację obcą neuronu wzorca $outSN$ z neuronem kombinacji KN** będzie nazywana ilość VN ów równocześnie połączonych z neuronem wzorca $outSN$, jak również neuronem kombinacji KN za pośrednictwem RN ów lub UN ów przy założeniu, że $outSN$ oraz KN nie są połączone z tym samym neuronem klasy CN ($outSN \leftrightarrow CN \leftrightarrow KN$). Można to zrealizować przy pomocy odpowiednich pobudzeń i hamowań, lecz tutaj dla ułatwienia, odpowiednio inkrementowane są liczniki przechowywane w $SNach$. W tym hybrydowym modelu liczniki te zliczają, ile asocjacyjnych powiązań ADEF z VN ami określonej kombinacji posiada każdy SN , którego chociaż jeden definiujący go $VN \in RN$ albo $VN \in UN$, w zależności od rodzaju parametru. Na rysunku 7.12 można zauważyć, że dla inicjalnej postaci kombinacji tylko $outSN_{10}$ jest asocjacyjnie powiązany równocześnie z dwoma VN ami kombinacji, a więc definiujące go VN y aktualizują jego licznik $OutCorrCtrs_{SN_{10}} = 2$, oraz $outSN_{12}$, który jest powiązany asocjacyjnie tylko z jednym VN em z tej inicjalnej postaci kombinacji reprezentowanej przez neuron KN_1 , a więc $OutCorrCtrs_{SN_{12}} = 1$. Taki proces aktualizacji liczników korelacji jest wznawiany zawsze po dodaniu do kombinacji nowego VNa . Aktualizowane są zawsze tylko te SN y, które są powiązane z nowymi, dodawanymi do kombinacji VN ami, posiadające z nimi powiązania asocjacyjne typu ADEF, tzn. $VN \leftrightarrow SN$, a więc nic nie trzeba wyszukiwać ani sprawdzać wszystkich SN ów, dlatego cała operacja aktualizacji odbywa się prawie zawsze ze stałą złożonością obliczeniową. Na początku $germSN$ dodaje do kombinacji wszystkie definiujące go VN y, więc operacja ta zostaje wykonana dla nich wszystkich. W przypadku równoległej implementacji, wszystkie VN y mogą równocześnie aktualizować liczniki wszystkich powiązanych z nimi SN ów, więc wtedy aktualizacja zachodzi błyskawicznie.

Liczniki korelacji $InCorrCtrs_{SN_x}$ i $OutCorrCtrs_{SN_y}$ są następnie wykorzystane do obliczenia **liczników dekorrelacji**, określających, ile VN ów, definiujących $outSN$ y, posiadających przynajmniej jedno powiązanie asocjacyjne z kombinacją KN , nie jest powiązanych z RN ami lub UN ami tej kombinacji, a więc dyskryminują te $outSN$ y. Minimalny licznik dekorrelacji każdego z $outSN$ ów powiązanych z kombinacją KN nie może być mniejszy niż jeden, dla zapewnienia 100% poprawnej klasyfikacji wszystkich niesprzecznych wzorców uczących. Jeśli kombinacja zawiera dużo VN ów powiązanych z $outSN$ ami, wtedy w naturalny sposób obniża to jej dyskryminatywność. Dyskryminatywność kombinacji obniża się tym bardziej, im równocześnie więcej VN ów powiązanych $outSN$ ami jest zawartych w kombinacji. Ilustrując to na przykładzie inicjalnej postaci kombinacji przedstawionej na rysunku 7.11, $outSN_{12}$ ma tylko jeden VN zawarty w kombinacji, więc jest lepiej dyskryminowany niż $outSN_{10}$, który ma aż dwa VN y zawarte w kombinacji. Takie VN y kombinacji definiujące wyłącznie albo również $outSN$ y „zachwaszczają” kombinację, więc warto ich możliwie nie dodawać wcale lub nadmiernie. Natomiast te wcześniej dodane można spróbować usunąć razem z częścią mało dyskryminatywnych atrybutów definiujących kombinację, co zostanie opisane później. W celu liczbowego określenia stopnia zachwaszczenia kombinacji przez takie VN y wprowadzono **współczynnik zachwaszczenia kombinacji $Weeds$** , który jest dla każdego RNa lub UNa zdefiniowany następująco:

$$Weeds_{RN} = \sum_{outSN \leftrightarrow CN \leftrightarrow KN} \|\{VN \leftrightarrow RN: VN \leftrightarrow outSN\}\|^2 \quad (1.10.)$$

$$Weeds_{UN} = \sum_{outSN \leftrightarrow CN \leftrightarrow KN} \|\{VN \leftrightarrow UN: VN \leftrightarrow outSN\}\|^2 \quad (1.11.)$$

a dla neuronu KN jest sumą tych wartości liczoną po wszystkich przedziałach i podzbiorach tworzących kombinację reprezentowanych przez RN y i UN y:

$$AllWeeds_{KN} = \sum_{RN \leftrightarrow KN} Weeds_{RN} + \sum_{UN \leftrightarrow KN} Weeds_{UN} \quad (1.12.)$$

Z powyższej definicji wynika, iż współczynniki $Weeds$ rosną nieliniowo z kwadratem ilości VN ów definiujących $outSN$ y powiązanych z RN ami lub UN ami kombinacji. W definicji tego współczynnika zastosowano kwadrat liczności zbioru takich negatywnie skorelowanych VN ów, gdyż każdy kolejny VN należący do kombinacji KN powiązany ADEF z tym samym $outSN$ em jest dużo bardziej dotkliwy dla jej dyskryminatywności i tym bardziej ogranicza jej możliwość rozbudowy o kolejne VN y.

Odnosząc się do inicjalnej postaci kombinacji z rysunku 7.12, na którym uwidoczniło się trzy takie powiązania $outSN$ ów z VN ami kombinacji KN , otrzymujemy następujące wartości tych współczynników dla poszczególnych neuronów RN :

$$\begin{aligned} Weeds_{RN1} &= 0 \\ Weeds_{RN2} &= 1^2 = 1 \\ Weeds_{RN3} &= 2^2 = 4 \\ Weeds_{RN4} &= 2^2 = 4 \\ AllWeeds_{KN1} &= 9 \end{aligned}$$

W podobny sposób można wyrazić liczbowo ilość VN ów powiązanych z $inSN$ ami, które nie są jeszcze reprezentowane przez kombinację, gdyż nie wszystkie definiujące je VN y, są powiązane z RN ami lub UN ami tej kombinacji. Z oczywistych powodów $inSN$ y, dla których niektóre definiujące je VN y są już powiązane z RN ami lub UN ami tej kombinacji, są teoretycznie najlepszymi kandydatami do reprezentacji przez kombinację, jeśli uda się powiązać pozostałe definiujące je VN y z RN ami lub UN ami. Może temu przeszkodzić powiązanie tych VN ów również z $outSN$ ami, których nadmierna reprezentacja w kombinacji może spowodować obniżenie jej stopnia dyskryminatywności, albo inne pośrednie VN y dla RN ów, które trzeba dołączyć do kombinacji, żeby od granicy obecnego przedziału dojść do tych pozostałych VN ów definiujących tych potencjalnie najlepszych kandydatów $inSN$. W przypadku podzbiorów sprawa jest prostsza, gdyż wystarczy przejść poprzez połączenia asocjacyjne ADEF do najbardziej pozytywnie skorelowanych z kombinacją $inSN$ ów, a następnie sprawdzić, czy dodanie pozostałych VN ów, je definiujących, nie spowoduje obniżenie stopnia dyskryminatywności kombinacji poniżej akceptowalnego lub pożądanego stopnia. W przypadku przedziałów jest nieco trudniej, gdyż kolejne VN y muszą być dodawane kolejno według określonego porządku. Takie SN y, których część VN ów jest zawartych w kombinacji, są niejako najlepszymi nasionami (*seeds*) dla dalszego rozwoju kombinacji, a więc również dla nich określono następujące współczynniki dla przedziałów RN lub podzbiorów UN :

$$Seeds_{RN} = \sum_{inSN \leftrightarrow CN \leftrightarrow KN} \|\{VN \leftrightarrow RN: VN \leftrightarrow inSN\}\|^2 \quad (1.13.)$$

$$Seeds_{UN} = \sum_{inSN \leftrightarrow CN \leftrightarrow KN} \|\{VN \leftrightarrow UN: VN \leftrightarrow inSN\}\|^2 \quad (1.14.)$$

oraz dla kombinacji KN :

$$AllSeeds_{KN} = \sum_{RN \leftrightarrow KN} Seeds_{RN} + \sum_{UN \leftrightarrow KN} Seeds_{UN} \quad (1.15.)$$

Odnosząc się do inicjalnej postaci kombinacji z rysunku 7.12, na którym widać, że żaden $inSN$ nie jest powiązany z kombinacją KN oprócz tego reprezentowanego $SeedSN_7$, a więc:

$$\begin{aligned} Seeds_{RN1} &= 0 \\ Seeds_{RN2} &= 0 \\ Seeds_{RN3} &= 0 \\ Seeds_{RN4} &= 0 \\ AllSeeds_{KN1} &= 0 \end{aligned}$$

Jeśli jakiś *inSN* zostanie już reprezentowany w kombinacji *KN*, tzn. wszystkie definiujące go *VNy* są połączone z odpowiednimi *RNami* lub *UNami* kombinacji, wtedy odpowiednie wartości związane z takimi *inSNami* są odejmowane od *Seeds* i *AllSeeds*, gdyż taki *inSN* przestaje być już potencjalnym kandydatem do reprezentacji w kombinacji.

Asocjacyjne poszerzanie kombinacji w przypadku przedziałów będzie polegało na dołączeniu do któregoś z *RNów* kolejnego *VNa*, który jest połączony ASIM z którymś granicznym *VNem* reprezentującym minimalną lub maksymalną wartość atrybutu tego *RNa*. Ilustrując to na przykładzie z rysunku 7.12, do *RN₁* może być dołączony *VN_{5,8}* lub *VN_{6,0}* albo do *RN₂* może być dołączony *VN_{3,0}* lub *VN_{3,3}*, albo do *RN₃* może być dołączony *VN_{4,7}* lub *VN_{5,0}*, albo do *RN₄* może być dołączony *VN_{1,7}* lub *VN_{1,9}*. Decyzja o wyborze nie jest łatwa, gdyż nie można jej podejmować tylko i wyłącznie na podstawie stopnia korelacji takiego powiązanego z granicznym *VNa*, który potencjalnie może być dołączony do któregoś *RNa*. Decyzja musi być podjęta na podstawie możliwie wszystkich *VNów*, które definiują *inSNy*, których część definiujących je *VNów* powiązana jest już z *RNami* lub *UNami* tej kombinacji, ponieważ to właśnie one mają szansę być najszybciej reprezentowane przez nią. Takie *inSNy* powinny więc wywierać odpowiedni wpływ na kierunek rozszerzania przedziałów reprezentowanych przez *RNy*. Dzięki asocjacji typu ADEF mamy bezpośredni dostęp do wszystkich *inSNów*, których niektóre *VNy* są już połączone z *RNami* lub *UNami*. Wystarczy je odpowiednio pobudzić przez te połączenia, a one sumując wszystkie pobudzenia z *VNów*, które są już w kombinacji, przesyłają informację o swoim pobudzeniu wynikającym z ich dotychczasowej reprezentacji do pozostałych swoich *VNów*, których w kombinacji jeszcze nie ma. Z kolei te *VNy*, sumując wszystkie takie pobudzenia, przychodzące od wszystkich *inSNów*, mogą pobudzić sąsiednie *VNy* poprzez ważne połączenia typu ASIM, rozsyłając swoje pobudzenie do sąsiednich *VNów* wskazując tym kierunek najbardziej pożądanym *VNów*. To umożliwi dokonanie kolejnych rozszerzeń przedziałów reprezentowanych przez *RNy*.

Z drugiej strony wszystkie *outSNy*, których część definiujących je *VNów* jest powiązana z *RNami* kombinacji, hamują pozostałe swoje *VNy*. Na skutek tego po zesumowaniu tych hamowań z pobudzającymi je sygnałami od *inSNów* tak wyhamowane *VNy* nie będą już tak intensywnie albo wcale pobudzać sąsiednie *VNy*. Ich wpływ na poszerzenie przedziałów *RN* będzie więc odpowiednio przytłumiony lub zupełnie wyhamowany. Dzięki temu przy prawie stałej złożoności obliczeniowej wszystkie stosowne *VNy inSNów* i *outSNów* zostaną odpowiednio pobudzone i hamowane. Ich ważne połączenia ASIM przekażą tę informację przez kolejne *VNy*, z którymi również dokonają odpowiedniego sumowania pobudzeń, aż do granic istniejących przedziałów reprezentowanych przez *RNy* kombinacji. Najmocniej pobudzony *VN* na granicy przedziałów *RNów* powinien jako pierwszy rozpocząć swój proces plastyczności i dołączania do odpowiedniego *RNa*, co spowoduje rozbudowę kombinacji w najbardziej pożądanym kierunku, gdzie jest szansa uzyskać możliwie maksymalne zwiększenie reprezentatywności przy możliwie minimalnym obniżeniu jej dyskryminatywności.

Jeśli odpowiednio dobrać sposób przesyłania takich informacji dla połączeń typu ASIM i ADEF, ważąc je ich wagami synaptycznymi, można doprowadzić do sytuacji, w której przedziały przestaną się rozszerzać w momencie, gdy stopień dyskryminatywności kombinacji miałby spaść poniżej akceptowalnego. Można też bezpośrednio określić granicę dla stopnia dyskryminatywności i ją sprawdzać, co z punktu symulacji jest rozwiązaniem łatwiejszym. Sumowanie pobudzeń i hamowań zaś jest przydatne w równoległych realizacjach. Przesyłanie informacji przez połączenia typu ASIM można przeprowadzić przy złożoności nieprzekraczającej liniową. Zależne to będzie od ilości wartości danego atrybutu. Jeśli tych wartości jest niewiele w stosunku do ilości wzorców uczących, można przyjąć, iż cały proces zachodzi przy złożoności stałej. Takie założenie zwykle można przyjąć dla dużych zbiorów uczących, dla których wiele wartości danego atrybutu powtarza się dla wielu wzorców uczących, natomiast dla małych zbiorów danych, nawet liniowa złożoność będzie nieznacząca dla czasu obliczeń, więc można powiedzieć, iż niezależnie od wielkości zbioru uczącego zadanie jest **asocjacyjnie dobrze uwarunkowane**. W podejściu hybrydowym można jeszcze dodatkowo uprościć cały proces przesyłania informacji o tych korelacjach wpływających na możliwości rozszerzenia przedziałów, przesyłając informacje zakodowane w postaci konkretnych ilości skorelowanych *VNów* dla poszczególnych *SNów*, które są odpowiednio dodawane lub odejmowane, zamiast korzystania z pobudzeń i hamowań. Przeprowadzone badania pokazały, iż kluczowe dla efektywności obliczeń asocjacyjnych nie zawsze jest dokładne odwzorowanie biologicznych mechanizmów pobudzeń i hamowań, lecz struktura i dynamika połączeń, równoległość, reaktywność, refrakcyjność i relaksacja neuronów, które determinują kolejność ich aktywacji tworzące skojarzenia będące reakcją i odpowiedzią całej sieci neuronowej.

W przypadku przedziałów *RN* warto zawsze poszerzać ich granice o *VNy*, które definiują tylko i wyłącznie *inSNy* reprezentowanej klasy przez rozważaną kombinację, gdyż one zawsze poprawiają jej reprezentatywność, a przy tym nigdy nie obniżają dyskryminatywności. Z takim właśnie przypadkiem mamy do czynienia w przypadku inicjalnych przedziałów rozważanej kombinacji z rysunku 7.12, gdzie graniczne $VN_{3,3} \leftrightarrow RN_2$ oraz $VN_{4,7} \leftrightarrow RN_3$ są połączone tylko i wyłącznie z *inSNami*, więc można je bez przeszkód od razu połączyć z odpowiednimi *RNami* tej kombinacji, otrzymując:

$$RN_1 = [15,9 ; 5,21]$$

$$RN_2 = [9,2 ; 3,9]$$

$$RN_3 = [11,7; 4,19]$$

$$RN_4 = [1,8; 1,19]$$

Poprzez dodanie $VN_{3,3}$ do RN_2 i $VN_{4,7}$ do RN_3 zwiększają się również współczynniki *Seeds* i *AllSeeds* zgodnie z zależnościami (7.13) i (7.15) na następujące:

$$Seeds_{RN1} = 0$$

$$Seeds_{RN2} = 2^2 = 4$$

$$Seeds_{RN3} = 2^2 = 4$$

$$Seeds_{RN4} = 0$$

$$AllSeeds_{KN1} = 8$$

Dodanie tych *VN*ów spowodowało, iż $inSN_4$ jest już częściowo reprezentowany przez tą kombinację, a przez to jego oddziaływanie na kierunki poszerzania przedziałów kombinacji będą większe niż dla pozostałych $inSN$ ów nierepresentowanych w tej kombinacji wcale.

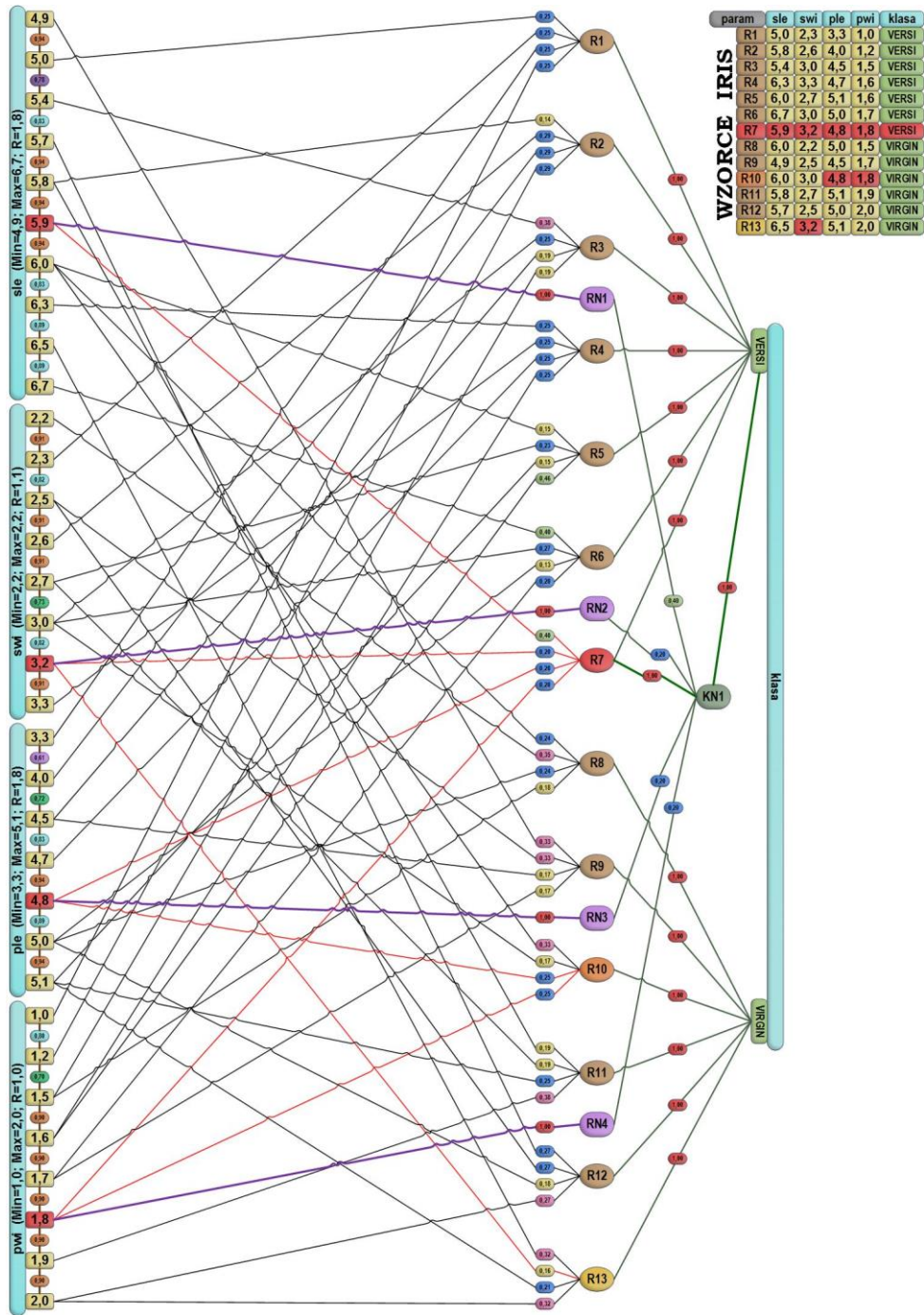
Następnym krokiem będzie określenie kolejnego *VNa*, o który będzie można poszerzyć kombinację KN_1 , możliwie zwiększając jej reprezentatywność. Ze względu na konieczność symulacji, poszczególne kierunki poszerzenia kombinacji będą sprawdzane kolejno. Dla każdego RNa i dwóch kierunków jego poszerzania o kolejne *VNy* zostanie wyznaczony pewien współczynnik (7.13) na podstawie powiązań z dotąd nierepresentowanymi *VNami* $inSN$ ów, które w tym kierunku występują, a z nich zostanie wybrany ten największy połączony z którymś obecnym granicznym *VNem* RNa , na podstawie którego zostanie dołączony do kombinacji kolejny *VN*. Cały ten proces będzie się odbywał tak długo, dopóki dodawanie kolejnych *VN*ów nie będzie powodowało konieczności obniżenia stopnia dyskryminatywności kombinacji. Gdy do tego dojdzie, zanim wszystkie $inSNy$ określonej klasy nie będą reprezentowane w kombinacji, wtedy w zależności od uzyskanej reprezentatywności kombinacji KN_1 zostanie podjęta decyzja:

- o zakończeniu budowy kombinacji KN_1 i rozpoczęciu budowy następnej KN_2 dla tej klasy, jeśli kombinacja KN_1 reprezentuje już sensowną ilość $inSN$ ów lub aktualny stopień jej dyskryminatywności jest równy jeden i uniemożliwia dalsze jego obniżanie,
- o obniżeniu stopnia dyskryminatywności kombinacji o jeden, jeśli był silnie większy od jedności, oraz kontynuowanie poszerzania kombinacji KN_1 , celem poprawy jej reprezentatywności.

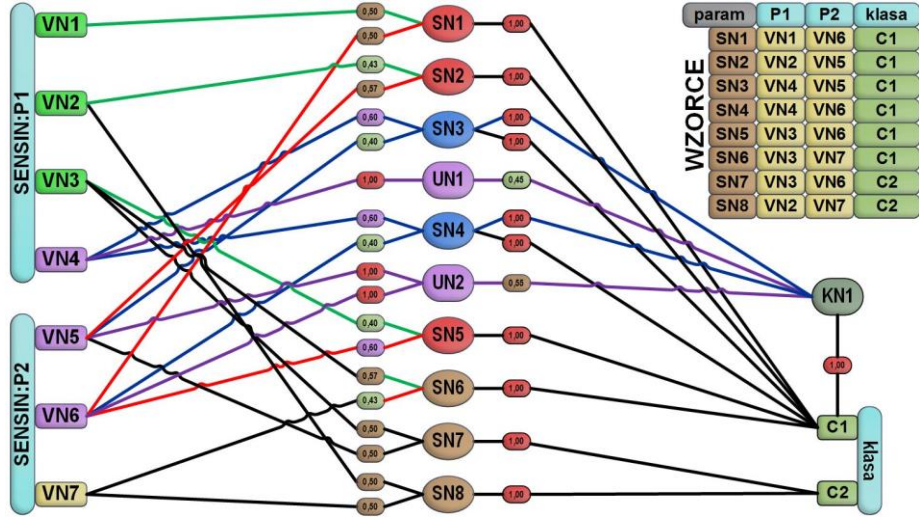
Do hybrydowej, sekwencyjnej realizacji poszerzania przedziałów kombinacji wybrano wyznaczenie współczynnika kierunku poszerzenia kombinacji, który uwzględnia wszystkie $inSNy$ oraz $outSNy$, które są powiązane z *VNami* od obecnych granic określonego przedziału RN kombinacji do rozważanego *VNa* poza granicami RNa . Wybór kierunku (7.16) dla *VN*ów reprezentujących mniejsze wartości lub (7.17) dla *VN*ów reprezentujących większe wartości określonego atrybutu (parametru) *P* oparto:

- o odległość reprezentowanej przez *VNa* wartości ($VN.val$) od odpowiedniej granicy ($RN.min$ lub $RN.max$) obecnego przedziału RN względem zakresu zmienności (7.28) wartości tego atrybutu dla reprezentowanej przez KNa klasy (7.18) (7.19)
- o ilość istniejących reprezentacji $inSN$ ów we wszystkich innych kombinacjach grafu AANG (7.24), żeby ukierunkować cały proces na $inSNy$ jeszcze nie reprezentowane w żadnej kombinacji, a te już reprezentowane dołączać tylko przy okazji,
- o ilość zawartych już w kombinacji *VN*ów (7.25) kandydujących do reprezentacji w kombinacji $inSN$ ów (7.16) (7.17), z którymi rozważane do dołączenia *VNy* są połączone (7.21),
- o ilość zawartych już w kombinacji *VN*ów (7.25) połączonych z $outSNami$ (7.16) (7.17), które przez dołączenie kolejnego połączonego z nim *VNa* mogą potencjalnie spowodować obniżenie stopnia dyskryminatywności kombinacji (7.21).

Im częściej reprezentowany jest już określony $inSN$ w innych kombinacjach (7.24), tym mniejszy wywiera wpływ na poszerzanie kombinacji w tym kierunku (7.20), gdyż w razie jego kolejnej reprezentacji otrzymujemy tylko pewną pozytywną redundancję, jeśli zachodzi ona przy okazji dołączania innych nie reprezentowanych jeszcze w kombinacji $inSN$ ów. Im większa jest odległość *VNa* od obecnych granic przedziału, tym mniejszy wpływ ma na wybór tego kierunku (7.18) (7.19). Im więcej połączonych *VN*ów z $RNami$ lub $UNami$ kombinacji jest równocześnie połączona z kandydującymi $inSNami$ (7.25), tym bardziej jest pożądanym kierunek poszerzania obejmujący brakujące *VNy*. Im więcej połączonych *VN*ów z $RNami$ lub $UNami$ kombinacji jest równocześnie połączona z $outSNami$ (7.25), tym bardziej ten kierunek poszerzania jest niekorzystny, ponieważ może spowodować obniżenie stopnia dyskryminatywności całej kombinacji. Współczynniki określające wybór potencjalnie najbardziej obiecującego kierunku (7.31) albo (7.32) poszerzania poszczególnych RN ów wyznaczone są jako sumy współczynników kierunków określonych dla poszczególnych *VN*ów (7.16) albo (7.17):



Rys. 7.12 Struktura grafu AANG po zainicjowaniu jego rozbudowy przez neuron załączkowy *germSN₇* neuronu kombinacji *KN₁* definiującego klasę VERSI, z którym powiązane są neurony przedziałów *RN₁, ..., RN₄*. Zaznaczono aktywacje i obecne korelacje z neuronami *outSN* z klasy VIRGIN.



Rys. 7.13 Przykładowy graf AANG zawierający dwa atrybuty o wartościach symbolicznych $P1$ i $P2$, w których rozpoczęto proces powiększania podzbiorów reprezentowanych przez neurony $UN1$ i $UN2$.

$$dir_{P,KN}^{VN^-} = \delta_{P,KN,VN^-}^{distmin} \cdot \left(\sum_{\substack{inSN \leftrightarrow VN^- \wedge \\ inSN \leftrightarrow CN \leftrightarrow KN}} \delta_{inSN}^{repr} \cdot \delta_{inSN,KN}^{cont} - \sum_{\substack{outSN \leftrightarrow VN^- \wedge \\ outSN \leftrightarrow KN}} \delta_{outSN,KN}^{cont} \right) \quad (1.16.)$$

$$dir_{P,KN}^{VN^+} = \delta_{P,KN,VN^+}^{distmax} \cdot \left(\sum_{\substack{inSN \leftrightarrow VN^+ \wedge \\ inSN \leftrightarrow CN \leftrightarrow KN}} \delta_{inSN}^{repr} \cdot \delta_{inSN,KN}^{cont} - \sum_{\substack{outSN \leftrightarrow VN^+ \wedge \\ outSN \leftrightarrow KN}} \delta_{outSN,KN}^{cont} \right) \quad (1.17.)$$

$$\delta_{P,KN,VN^-}^{distmin} = \left(1 - \frac{|RN.min - VN.val|}{Range_P^{KN}} \right)^2 \quad (1.18.)$$

$$\delta_{P,KN,VN^+}^{distmax} = \left(1 - \frac{|RN.max - VN.val|}{Range_P^{KN}} \right)^2 \quad (1.19.)$$

$$\delta_{inSN}^{repr} = \left(\frac{1}{1 + q_{inSN}^{repr}} \right)^2 \quad (1.20.)$$

$$\delta_{SN,KN}^{cont} = \left(\frac{1 + q_{SN,KN}^{vn}}{\|\{RN: RN \leftrightarrow KN\}\| + \|\{UN: UN \leftrightarrow KN\}\|} \right)^2 \quad (1.21.)$$

$$RN.min > VN^-.val \in P \quad (1.22.)$$

$$RN.max > VN^+.val \in P \quad (1.23.)$$

$$q_{inSN}^{repr} = \sum_{KN \in AANG} \|\{inSN: inSN \leftrightarrow KN\}\| \quad (1.24.)$$

$$q_{SN,KN}^{vn} = q_{SN,KN}^{vnrn} + q_{SN,KN}^{vnun} \quad (1.25.)$$

$$q_{SN,KN}^{vnrn} = \sum_{RN \leftrightarrow KN} \|\{VN: VN \leftrightarrow RN \wedge VN \leftrightarrow SN \leftrightarrow KN\}\| \quad (1.26.)$$

$$q_{SN,KN}^{mun} = \sum_{UN \leftrightarrow KN} \|\{VN: VN \leftrightarrow UN \wedge VN \leftrightarrow SN \leftrightarrow KN\}\| \quad (1.27.)$$

$$Range_P^{KN} = Max_P^{KN} - Min_P^{KN} \quad (1.28.)$$

$$Max_P^{KN} = \max_{VN.val \in P} \{VN.val : \exists_{SN \leftrightarrow CN \leftrightarrow KN} VN \leftrightarrow SN\} \quad (1.29.)$$

$$Min_P^{KN} = \min_{VN.val \in P} \{VN.val : \exists_{SN \leftrightarrow CN \leftrightarrow KN} VN \leftrightarrow SN\} \quad (1.30.)$$

$$dir_{P,KN}^- = \frac{1}{\gamma_P} \cdot \sum_{VN^-} dir_{P,KN}^{VN^-} \quad (1.31.)$$

$$dir_{P,KN}^+ = \frac{1}{\gamma_P} \cdot \sum_{VN^+} dir_{P,KN}^{VN^+} \quad (1.32.)$$

Chcąc wykorzystać **najtańsze atrybuty (w sensie ekonomicznym)**, wystarczy podstawić koszt ich pozyskania za γ_P w (7.31), (7.32) i (7.33). To zapewni preferowanie tych najtańszych oraz późniejszą łatwą redukcję drogiego atrybutów, które w wyniku tego są przy budowie grafu rzadko wykorzystywane lub wcale. W przypadku braku małej istotności kosztu pozyskania poszczególnych atrybutów należy przyjąć takie same wartości dla nich wszystkich, np. $\gamma_P = 1$ dla każdego atrybutu P .

Współczynniki (7.16), (7.17) są wyznaczone do granic zmienności wartości poszczególnych atrybutów (parametrów) P (7.29) i (7.30.) dla określonej klasy reprezentowanej przez $CN \leftrightarrow KN$. Każdy VN jest powiązany bezpośrednio z $inSNem$, więc dostępność informacji o aktualnym stopniu reprezentacji VN ów $inSNa$ w kombinacji jest natychmiastowa. Ze względu na konieczność przejścia poprzez kolejne VNy połączone ASIM ze sobą, złożoność wyznaczenia wszystkich współczynników kierunków dla wszystkich RN ów w danym kroku poszerzania kombinacji jest maksymalnie liniowa. Budowa wszystkich kombinacji dla wszystkich klas ma łącznie złożoność kwadratową w tym sekwencyjnym, hybrydowym modelu, ponieważ każda kombinacja poszerza swoje przedziały tylko w ograniczonym zakresie. Rzeczywiste obliczenia są jednak zwykle dużo szybciej przeprowadzane, niż by wynikało z tych teoretycznych, maksymalnych i pesymistycznych ograniczeń, które prawie nigdy w praktyce nie zachodzą.

Dla parametrów symbolicznych nieposiadających porządku, których VNy nie są powiązane połączeniami typu ASIM, poszukiwane są podzbiory VN ów łączone z $UNami$ zamiast z $RNami$. W takim przypadku do kombinacji dodawane są w pierwszym rzędzie VNy ($VN1$, $VN2$, $VN3$ na rys. 7.13) pozwalające osiągnąć pełną reprezentację kolejnych $inSN$ ów ($SN1$, $SN2$, $SN5$ na rys. 7.13) w kombinacji, które na razie są częściowo reprezentowane przez pewną ilość VN ów ($VN4$, $VN5$, $VN6$ na rys. 7.13) powiązanych z $UNami$ ($UN1$, $UN2$ na rys. 7.13) kombinacji. Dodawanie tych VN ów odbywa się tak, aby najpierw dodawać te VNy ($VN1$ na rys. 7.13), które wcale nie posiadają połączeń z $outSNami$, następnie te ($VN2$ na rys. 7.13), które nie obniżą stopnia dyskryminatywności kombinacji, a jeśli KN nie osiągnie sensownej reprezentatywności, wtedy w dalszej kolejności te VNy , które obniżą stopień dyskryminatywności kombinacji w minimalnym stopniu, lecz nie poniżej pożądanej lub akceptowalnej granicy ($VN3$ na rys. 7.13), która by uniemożliwiła jednoznaczny dyskryminację wzorców poszczególnych klas. Do takich VN ów ($VN1$, $VN2$, $VN3$ na rys. 7.13) docieramy pośrednio poprzez $inSNy$ ($SN1$, $SN2$, $SN5$ na rys. 7.13), które są już częściowo reprezentowane w kombinacji KN ($KN1$ na rys. 7.13) przez inne VNy ($VN4$, $VN5$, $VN6$ na rys. 7.13). Dla takich neuronów VN_i ($KN \leftrightarrow inSN \leftrightarrow VN_i \leftrightarrow UN \leftrightarrow KN$) (rys. 7.13) wyznaczamy wartość współczynnika kierunku rozwoju przedziału (7.33). Rozpoczynamy więc sprawdzanie $inSN$ ów od tych, które są już częściowo reprezentowane przez możliwie maksymalną ilość VN ów (7.25), gdyż to one mogą najszybciej zwiększyć reprezentatywność kombinacji, jeśli uda się je w całości reprezentować bez obniżania stopnia dyskryminatywności.

$$dir_{P,KN}^{VN_i} = \frac{1}{\gamma_P} \cdot \left(\sum_{\substack{inSN \leftrightarrow VN_i \wedge \\ inSN \leftrightarrow KN}} \delta_{inSN}^{repr} \cdot \delta_{inSN,KN}^{cont} - \sum_{\substack{outSN \leftrightarrow VN_i \wedge \\ outSN \leftrightarrow KN}} \delta_{outSN,KN}^{cont} \right) \quad (1.33.)$$

Następnie spośród obliczonych kierunków wybierany jest ten o największej wartości współczynnika kierunku (7.31) albo (7.32) dla *RN*ów oraz (7.33) dla *UN*ów, który zarazem nie obniża stopnia dyskryminatywności kombinacji *KN*. Jeśli żaden kierunek nie spełnia tego kryterium, a osiągnięta reprezentatywność kombinacji jest niewielka, zaś stopień dyskryminatywności kombinacji jest silnie większy od jedności, wtedy automatycznie pozwala się na obniżenie tego stopnia o jeden i wznowiany jest proces wybierania kierunku o największej wartości. Cały proces jest powtarzany tak długo, aż właśnie rozbudowywana kombinacja nie osiągnie sensownej reprezentatywności albo wszystkie *inSN*y rozważanej klasy zostaną już reprezentowane, albo stopnia dyskryminatywności kombinacji równego jedności, którego dalej już nie sposób obniżyć, ponieważ spowodowałoby to utratę możliwości jednoznacznej klasyfikacji dla części wzorców uczących. W takim przypadku proces rozbudowy tej kombinacji jest zakończony i sprawdza się, czy pozostały jeszcze jakieś neurony *inSN*, które nie są jeszcze reprezentowane przez którąś kombinację. Jeśli tak, wtedy wybierany jest kolejny najbardziej skorelowany z *outSN*ami *germSN*, który staje się załącznikiem kolejnej kombinacji, dla której tworzony jest następny neuron *KN*. Ten proces powtarzany jest tak długo, dopóki wszystkie *inSN*y każdej klasy nie zostaną reprezentowane przynajmniej jednokrotnie przez którąś kombinację.

Pod koniec procesu tworzenia kombinacji może się okazać, że osiągnięta dyskryminatywność kombinacji jest niepotrzebnie duża, szczególnie gdy wzorce uczące zdefiniowane są przy pomocy wielu atrybutów. Tą wysoką dyskryminatywność kombinacji można wykorzystać do redukcji mało dyskryminatywnych atrybutów (parametrów), usuwając je z kombinacji (rys. 7.14). Za mało dyskryminatywne atrybuty uznamy te, które w zakresie lub podzbiorze reprezentowanych przez nie wartości zawierają wiele *VN*ów, które są połączone z wieloma *outSN*ami, a więc wpływają na zachwaszczenie kombinacji, tzn. wartość *Weeds* ((7.10) albo (7.11)) jest dla nich największa. Usuwanie mało dyskryminatywnych neuronów przedziałów *RN* i neuronów podzbiorów *UN* często powoduje obniżenia dyskryminatywności kombinacji, więc usuwanie można przeprowadzać maksymalnie do momentu osiągnięcia pewnej pożądanej lub minimalnej (równej jedności) dyskryminatywności, która dalej zapewnia poprawną i jednoznaczną klasyfikację wszystkich wzorców uczących. Przykład usunięcia dwóch mało dyskryminatywnych neuronów *RN*₃ i *RN*₂ zilustrowano na rysunku 7.14. W wyniku tego kombinacja *KN*₁ wykorzystuje tylko najbardziej dyskryminatywne wartości atrybutów reprezentowane przez neurony *RN*₁ i *RN*₄. Usuwanie mało dyskryminatywnych receptorów z kombinacji jest więc zjawiskiem pozytywnym, gdyż umożliwia oparcie pracy klasyfikatora ASONN na kombinacjach tylko tych przedziałów i podzbiorów wartości, które są równocześnie najbardziej reprezentatywne i dyskryminatywne, co decyduje o jakości uogólniania. Jeśli równocześnie wszystkie kombinacje utworzone dla danego zbioru uczącego usuną swoje *RN*y lub *UN*y dla tego samego atrybutu (parametru), co często się zdarza, wtedy dojdzie do sytuacji zilustrowanych na rysunku 7.19 i 7.20. W tych przypadkach część atrybutów została całkowicie wykluczona z procesu klasyfikacji, a więc klasyfikator ASONN automatycznie zredukował przestrzeń danych do tych atrybutów, które uznał za najbardziej reprezentatywne i dyskryminatywne do przeprowadzenia poprawnej i jednoznacznej klasyfikacji oraz dla poprawnego uogólniania. Oprócz automatycznej minimalizacji przestrzeni danych względem pożądanego stopnia dyskryminacji uzyskujemy jeszcze dodatkowo informację o tym, które atrybuty (parametry, cechy) wzorców uczących są mało wartościowe z punktu widzenia klasyfikacji. Dla zbioru Wine z ML Repository jako mało przydatne atrybuty zostały określone: 4. Alcalinity, 5. Magnesium, 6. Phenols, 8. Nonflavanoid i 9. Proanthocain (rys. 7.19).

Z powyższego opisu wynika, iż odnalezienie wszystkich kombinacji dla wszystkich klas reprezentujących wszystkie niesprzeczne wzorce uczące odbywa się przy maksymalnie kwadratowej złożoności obliczeniowej dzięki wykorzystaniu struktury AGDS oraz grafom AANG. Osiągnięcie tego możliwe było ze względu na to, iż struktura AGDS łączy ze sobą dane w specyficzny sposób, zapewniając natychmiastowy dostęp do potrzebnych danych w trakcie obliczeń, tzn. bez konieczności ich wyszukiwania. ASONN w przedstawionej tutaj hybrydowej postaci, niewykorzystującej równoległości ani innych cech asocjacyjnych neuronów dowodzi, iż bardzo ważne znaczenie dla obniżenia złożoności obliczeniowej ma sama specyficzna grafowa struktura AGDS. Powyżej opisany sposób realizacji ASONN nie jest jedynym możliwym, a w miarę możliwości wykorzystania sprzętu posiadającego odpowiednią ilość rdzeni działających równoległe można te obliczenia znacznie przyspieszyć.

Budowa asocjacyjnego klasyfikatora ASONN

Odnalezienie kombinacji, reprezentujących wszystkie niesprzeczne wzorce uczące reprezentowane przez neurony wzorców *SN*, umożliwia przeprowadzenie dalszej optymalizacji aktywnego asocjacyjnego grafu neuronowego AANG, do którego dodano stosowne neurony *KN*, *RN* i *UN* oraz ich asocjacyjne połączenia z

dotychczasowymi neuronami tego grafu VN , SN i CN (rys. 7.14). Neurony przedziałów RN i neurony podzbiorów UN zostaną przekształcone w neurony receptoryczne, reagujące na wartości pewnego parametru z określonego przedziału lub podzbioru wartości. Neurony kombinacji KN będą wyznaczały sumę ważoną z pobudzeń przekazanych im przez ich RN y i UN y w sieci ASONN. Neurony KN są połączone z neuronami klas CN , których wynikowe pobudzenia będą odpowiedzią klasyfikatora określającą dla poszczególnych klas stopień podobieństwa wartości atrybutów podanych na wejścia sensoryczne sieci, na które reagują RN y i UN y. Jeśli dojdzie do aktywacji jakiegoś neuronu CN oznacza, iż wzorzec został sklasyfikowany jako w pełni zgodny z klasą reprezentowaną przez ten neuron. Pozostała część grafu AANG, składająca się z neuronów VN i SN , nie będzie już potrzebna i może zostać usunięta z wynikowej topologii klasyfikatora ASONN (rys. 7.16), który jest podgrafem grafu AANG. Ze względu na to, iż wartości reprezentowane przez neurony receptoryczne RN i UN mogą być w różny sposób skorelowane z $outSN$ ami, wagi połączeń pomiędzy nimi i neuronami KN wyznaczone są w taki sposób, żeby te mniej skorelowane z $outSN$ ami miały większe wagi niż te bardziej skorelowane. To pozytywnie wpływa na lepsze wyniki uogólniania klasyfikatora ASONN. W tym celu dla każdego neuronu receptorycznego wyznaczany jest współczynnik (7.34) albo (7.35) odzwierciedlający ilość reprezentowanych neuronów wzorców SN oraz ilość pozostałych niewykorzystanych nasion (*Seeds*) (7.13) i (7.14), a także pozostałych chwastów (*Weeds*) (7.10) i (7.11) w każdej kombinacji:

$$\Phi_{RN_k} = \left(1 - \frac{Weeds_{RN_k}}{\varphi_{SN}^{discr}}\right) \cdot \frac{\varphi_{SN}^{repr} + Seeds_{RN_k}}{\varphi_{SN}^{repr} + AllSeeds_{KN}} \quad (1.34.)$$

$$\Phi_{UN_k} = \left(1 - \frac{Weeds_{UN_k}}{\varphi_{SN}^{discr}}\right) \cdot \frac{\varphi_{SN}^{repr} + Seeds_{UN_k}}{\varphi_{SN}^{repr} + AllSeeds_{KN}} \quad (1.35.)$$

gdzie φ_{SN}^{repr} oznacza ilość reprezentowanych $inSN$ ów przez kombinację przemnożoną przez kwadrat ilości atrybutów służących do ich reprezentacji w kombinacji:

$$\varphi_{SN}^{repr} = \|\{inSN: inSN \leftrightarrow KN\}\| \cdot \|\{RN: RN \leftrightarrow KN\} \cup \{UN: UN \leftrightarrow KN\}\|^2 \quad (1.36.)$$

φ_{SN}^{discr} oznacza ilość wszystkich $outSN$ ów przemnożoną przez kwadrat ilości atrybutów służących do ich reprezentacji w kombinacji:

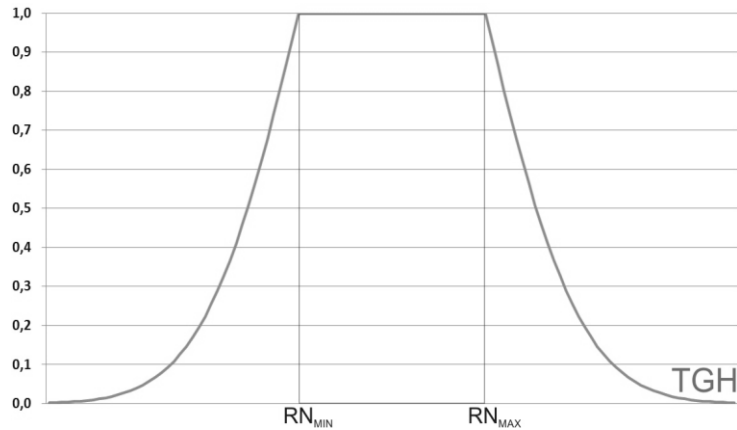
$$\varphi_{SN}^{discr} = \|\{SN: SN \not\leftrightarrow CN \leftrightarrow KN\}\| \cdot \|\{RN: RN \leftrightarrow KN\} \cup \{UN: UN \leftrightarrow KN\}\|^2 \quad (1.37.)$$

Na podstawie współczynników (7.34) wyznaczone są wagi połączeń pomiędzy odpowiednimi neuronami receptorycznymi RN (7.38) albo neuronami przedziałów UN (7.39) oraz neuronami kombinacji KN :

$$w_{RN_k \leftrightarrow KN} = \frac{\Phi_{RN_k}}{\sum_{RN_n \leftrightarrow KN} \Phi_{RN_n} + \sum_{UN_m \leftrightarrow KN} \Phi_{UN_m}} \quad (1.38.)$$

$$w_{UN_k \leftrightarrow KN} = \frac{\Phi_{UN_k}}{\sum_{RN_n \leftrightarrow KN} \Phi_{RN_n} + \sum_{UN_m \leftrightarrow KN} \Phi_{UN_m}} \quad (1.39.)$$

Wynikowy klasyfikator ASONN wraz z wartościami wag zobrazowano na rysunku 7.17 . Wagi powiązań asocjacyjnych ADEF pomiędzy neuronami receptorycznymi i neuronami kombinacji mieszczą się w zakresie (0; 1]. Można to na gruncie biologii zinterpretować jako różny stopień wrażliwości synaptycznej na pobudzenie presynaptyczne. Neurony kombinacji KN tworzą w przestrzeni danych hiperprostopadłościanny, których wymiar jest mniejszy lub równy wymiarowi przestrzeni danych określonej przez zbiór uczący. Wszystkie kombinacje wartości atrybutów, które znajdują się wewnątrz tych hiperprostopadłościannów dają zawsze wynik klasyfikacji równy jedności odzwierciedlając 100% podobieństwo do klas definicyjnych. Wewnątrz hiperprostopadłościannów zachodzi proces uogólniania.



Rys. 7.15 Funkcja ściętego kapelusza Gaussa (7.40) modelująca wrażliwość neuronu receptorycznego RN klasyfikatora ASONN, która przyjmuje wartości maksymalne w zakresie reprezentowanego przedziału wartości, natomiast poza nim stopniowo spada od punktów przegięcia funkcji Gaussa.

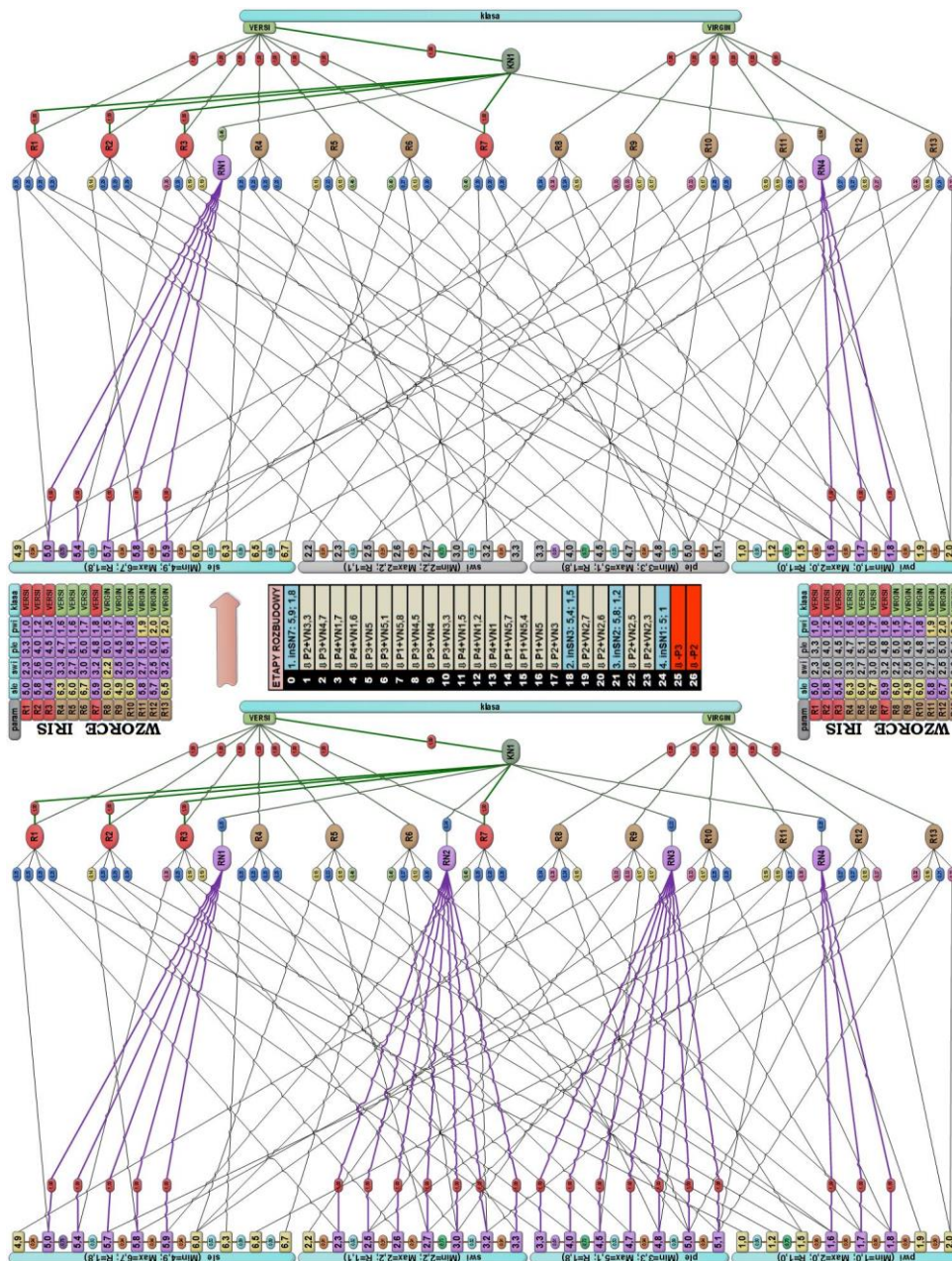
W celu uzyskania uogólnienia klasyfikatora ASONN również poza granicami hiperprostokądnianów rozmyto ich granice poprzez zdefiniowaną poniżej funkcję **ściętego kapelusza Gaussa (TGH - truncated Gauss hat)** (rys. 7.15). Jest ona zbudowana na bazie funkcji Gaussa (nazywanej też kapeluszem Gaussa) ściętej w jej punktach przegięcia oraz przeskalowanej tak, żeby wartości funkcji w tych punktach były równe jedności. Góra ściętego kapelusza Gaussa odpowiada przedziałom reprezentowanym przez neurony RN.

$$TGH(x) = \begin{cases} 1 & RN_{Min} \leq x \leq RN_{Max} \\ e^{-\frac{2 \cdot (x - RN_{Max} - RN_{Min})^2}{RN_{Max} - RN_{Min}}} & x < RN_{Min} \vee x > RN_{Max} \end{cases} \quad (1.40)$$

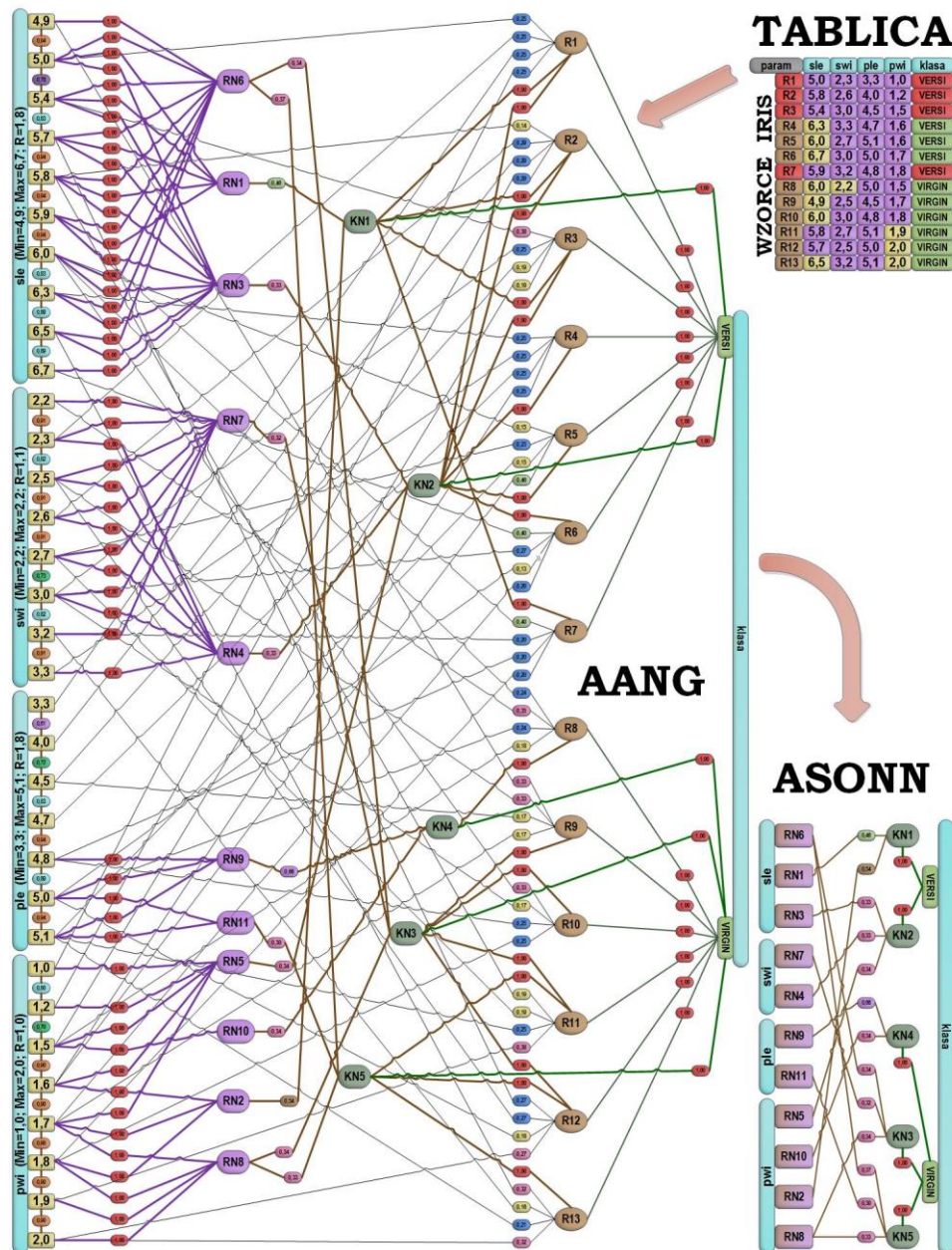
Rysunki 7.14, 7.16 i 7.17 ilustrują proces tworzenia klasyfikatora ASONN w grafie AANG dla rozważanego w tym rozdziale 13-elementowego zbioru Iris definiującego dwie klasy Virginica i Versicolor. Tworzą go mocno skorelowane wzorce, na których można zilustrować przebieg tworzenia się kilku kombinacji. Rysunek 7.14 przedstawia 26 etapów tworzenia się w grafie AANG pierwszej kombinacji dla klasy Versicolor reprezentowanej przez CN_1 . Neuronem zależkowym tej kombinacji został najbardziej negatywnie skorelowany $germSN_7$. Najpierw przedziały są poszerzane o nowe wartości umożliwiające reprezentację kolejnych $inSN$ ów. W 17. etapie wszystkie neurony VN definiujące neuron wzorca $inSN_3$ zostały reprezentowane w kombinacji, a więc w 18. etapie neuron ten został dołączony do kombinacji. Podobnie w kolejnych etapach zostały dołączone $inSN_2$ i $inSN_1$. W 24. etapie budowy kombinacji neuron kombinacji KN_1 reprezentował już 4 neurony wzorców klasy Versicolor. Kombinacja KN_1 nie była w stanie się dalej rozwijać bez utraty jej minimalnego stopnia dyskryminatywności, więc proces jej rozbudowy został zakończony. Następnie została utworzona kombinacja KN_2 , która rozpoczynając od kolejnego najbardziej skorelowanego neuronu wzorca $germSN_5$ z pozostałych niereprezentowanych SN ów klasy Versicolor przez KN_1 . Kombinacja ta w 30 kolejnych etapach umożliwiła reprezentację wszystkich pozostałych wzorców tej klasy wraz z pewną pozytywną redundancją już reprezentowanych wzorców w pierwszej kombinacji: SN_1 , SN_2 , SN_3 , SN_4 , SN_5 , SN_6 .

Na rysunku 7.14 widać końcowy efekt tworzenia się pierwszej kombinacji, z której zostały usunięte najmniej dyskryminatywne przedziały dla atrybutów (parametrów) P_2 i P_3 . Klasyfikator ASONN korzystając ze współczynników dekorrelacji *Seeds* (nasiona) i *Weeds* (chwasty) jest w stanie dokonać optymalizacji ilości atrybutów tak, żeby z kombinacji usunąć możliwie dużo chwastów (*Weeds*), tzn. połączeń z $outSN$ ami poprzez usunięcie części neuronów VN , które są z $outSN$ ami połączone. Takie negatywnie skorelowane VN y niekorzystnie wpływają na klasyfikację i uogólnianie, więc jeśli nie obniża to stopnia dyskryminatywności kombinacji poniżej akceptowanego, takie mało reprezentatywne przedziały warto usunąć. Dzięki temu klasyfikator opiera więc swoje działanie na najbardziej reprezentatywnych dla danej kombinacji i klasy atrybutach oraz ich wartościach.

Na rysunku 7.16 zilustrowano końcową postać rozbudowanego grafu AANG oraz klasyfikatora ASONN. Struktura tego grafu wydaje się być bardzo skomplikowana w stosunku do niewielkiej tabeli wzorców uczących. Graf AANG reprezentuje jednak wiele uniwersalnych powiązań asocjacyjnych pomiędzy różnymi danymi i ich kombinacjami, które mogą być wykorzystane w różny sposób w zależności od postawionego celu, a nie tylko do klasyfikacji. Podobnie zresztą jest w BASach, które posiadają wiele połączeń i bogatą funkcjonalność. Natomiast wydzielenie z grafu AANG podgrafu reprezentującego neurony realizujące zadanie klasyfikacji doprowadziło do uzyskania niewielkiej sktury klasyfikatora ASONN, która poprawnie i jednoznacznie klasyfikuje wszystkie wzorce uczące.



Rys. 7.14 Prezentacja 26 etapów plastycznej rozbudowy i optymalizacji kombinacji KN_1 przeprowadzonych przez graf AANG dla przyjętego stopnia dyskryminatywności kombinacji równego 1. W wyniku tego wszystkie wzorce drugiej klasy CN_2 są dyskryminowane minimalnie przez jedną wartość nie reprezentowaną przez RN_y kombinacji KN_1 . Zdyskryminowane neurony wartości VN zaznaczono kolorem żółtym. Neuron kombinacji KN_1 reprezentuje 4 neurony wzorców $inSN_7, inSN_3, inSN_2, inSN_1$ powiązane z neuronem klasy CN_1 . Do ich reprezentacji w kombinacji doszło odpowiednio w 0., 18., 21. i 24. etapie. W ostatnich dwóch etapach zredukowano mało dyskryminatywne atrybuty: swi i ple.



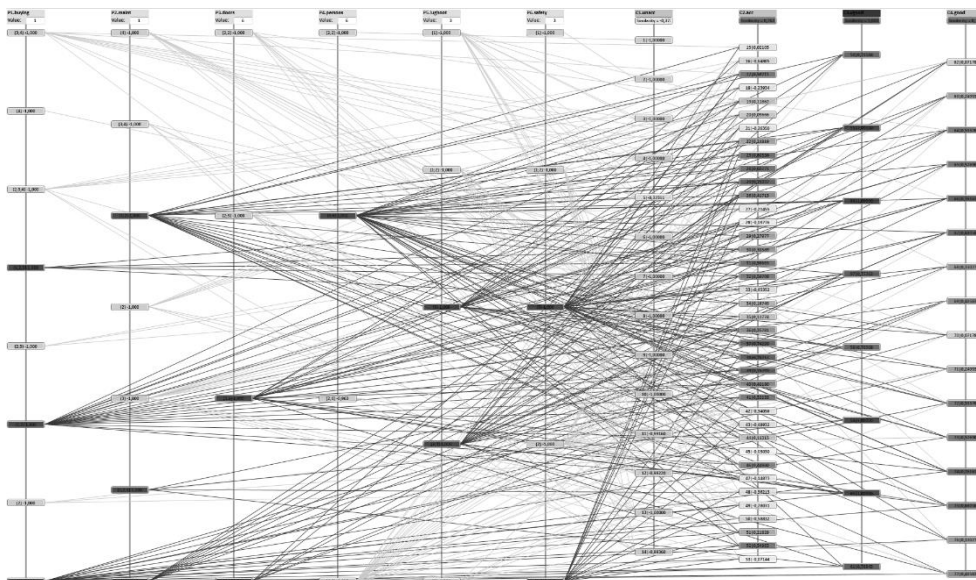
Rys. 7.16 Przekształcenie tabeli reprezentującej zbiór danych uczących do grafu AANG rozbudowanego o kolejne neurony *KN* i *RN* oraz połączenia pomiędzy nimi i pozostałymi elementami jego podstawowej struktury utworzonej na podstawie ADGS (rys. 7.4). Neurony *KN* i *RN* stworzyły wynikową postać klasyfikatora ASONN zawierającego 11 neuronów receptorycznych *RN*, 5 neuronów *KN*, 13 odpowiednio ważonych połączeń *RN* ↔ *KN* oraz 2 efekторы reagujące na ich wyjściu, które przekazują na swoje wyjścia, zapewniając poprawną i jednoznaczną klasyfikację wszystkich wzorców uczących oraz stopień ich dyskryminacji nie mniejszy niż 1.

Asocjacyjny klasyfikator ASONN posiada następujące cechy:

- ✓ Nie wymaga wstępnej konwersji danych, a nawet jej brak jest korzystny z punktu widzenia jego działania.
- ✓ Nie wymaga stosowania metod redukcji wymiaru danych (np. PCA – *principal component analysis*, ICA – *independent component analysis*), gdyż sam radzi sobie z jego redukcją, mogąc ją wykonywać automatycznie również z uwzględnieniem kosztu pozyskania danych dla poszczególnych atrybutów (parametrów).
- ✓ Nie wymaga inicjacji struktury sieci, wag synaptycznych, progów aktywacji ani żadnych innych parametrów poza możliwością opcjonalnego podania kosztu (w sensie ekonomicznym) pozyskania danych dla poszczególnych atrybutów.
- ✓ Korzystając z właściwości grafów AANG i wiedzy na temat klas jest w stanie automatycznie określić, który parametr w zbiorze danych jest najprawdopodobniej klasą, a które są atrybutami danych.

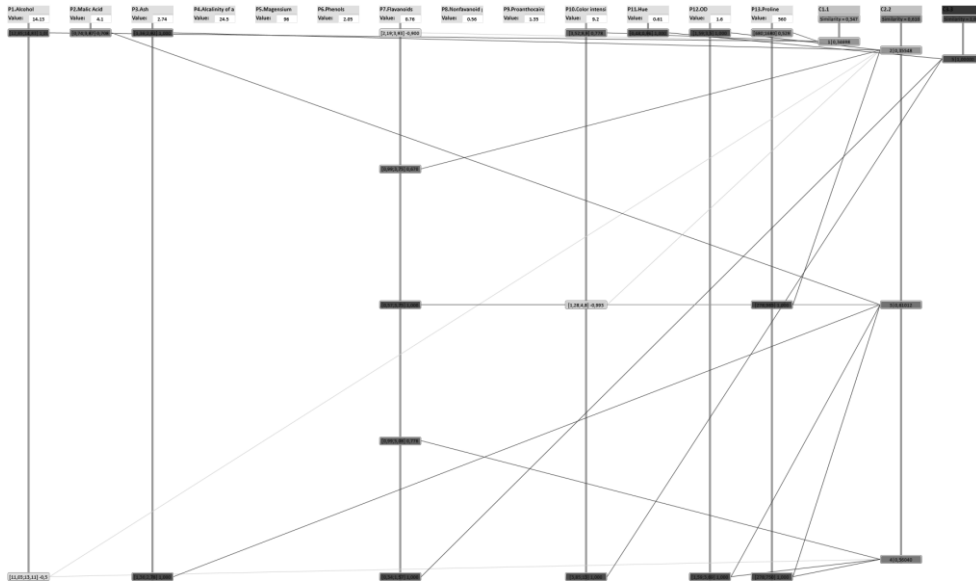
- ✓ Stosuje odmienne podejście do wartości atrybutów porządkowalnych i symbolicznych, odwzorowując ich odmiennosc w strukturze grafu AANG oraz wykorzystując ją do optymalizacji swojego działania.
- ✓ Tworzy swoją architekturę całkowicie automatycznie wykorzystując do tego aktywne asocjacyjne grafy neuronowe AANG oraz strukturę AGDS, dostosowując i optymalizując ją dla zadania określonego przez dany zbiór uczący.
- ✓ Nie wymaga określania żadnych parametrów uczenia się, gdyż sam może szukać sensownego kompromisu pomiędzy reprezentatywnością kombinacji i jej stopniem dyskryminatywności, aczkolwiek możliwe jest podanie pożądanego stopnia dyskryminacji wzorców.
- ✓ Nie jest narażony na utknięcie w żadnym minimum lokalnym.
- ✓ Jego proces tworzenia się nie jest narażony na przeuczenie się (*overfitting*).
- ✓ Przeprowadza cały proces jego konstrukcji przy maksymalnie kwadratowej złożoności obliczeniowej dzięki zastosowaniu asocjacyjnych powiązań.
- ✓ Mimo dużej hybrydyzacji jego modelu obliczeniowego i znacznego uproszczenia modelu jego elementów, szczególnie neuronów ASN, można było wykorzystać asocjacje ASIM i ADEF pomiędzy danymi i ich kombinacjami. Było to możliwe dzięki temu, iż tego rodzaju zbiory danych nie wymagają odwzorowania następstwa wzorców uczących, lecz tylko niezależnych kombinacji wartości poszczególnych atrybutów definiujących wzorce uczące.
- ✓ Wyniki klasyfikacji oparte są o przewidywalne uogólnienia bazujące na rozmytych hiperprostokątach utworzonych na podstawie kombinacji najbardziej reprezentatywnych i dyskryminatywnych przedziałów i podzbiorów wartości atrybutów danych. Przedziały i podzbiory odwzorowane są w neuronach receptorycznych reagujących na ich pojawienie się na odpowiednich wejściach sensorycznych utworzonych dla poszczególnych atrybutów danych.
- ✓ Osiąga zawsze 100% poprawne i jednoznaczne wyniki klasyfikacji dla niesprzecznych wzorców uczących, a wyniki jego uogólniania dla walidacji krzyżowej porównywalne są z najlepszymi klasyfikatorami obecnie stosowanymi na świecie. Porównania różnych metod przedstawiono w tabelach 7.2-5.
- ✓ Potrafi automatycznie wykorzystać najtańsze (w sensie ekonomicznym) atrybuty (parametry), dokonując na ich podstawie poprawnej i jednoznacznej klasyfikacji z uwzględnieniem zróżnicowanego kosztu pozyskania poszczególnych atrybutów.
- ✓ Jego działanie jest stabilne, a uzyskiwane wyniki cechują się wysoką jakością uogólniania niezależnie od zbioru danych uczących i korelacji wzorców uczących.

Na rysunku 7.17 została zilustrowana wygenerowana przez aplikację komputerową struktura sieci ASONN zbudowana dla zbioru uczącego Cars z ML Repository składającego się z 1728 wzorców uczących, z których każdy zdefiniowany jest przy pomocy 6 wartości atrybutów. Graf AANG utworzył 77 kombinacji dla tych wzorców w przeciągu kilku sekund, tworząc klasyfikator ASONN, który zgodnie z założeniami sklasyfikował wszystkie wzorce uczące w 100% poprawnie i jednoznacznie. Ponadto zawsze odpowiednie efekторы podawały na swoich wyjściach wartość 1, która oznacza pełne podobieństwo klasyfikowanego wzorca uczącego do jego klasy definicyjnej określonej dla niego w zbiorze uczącym.

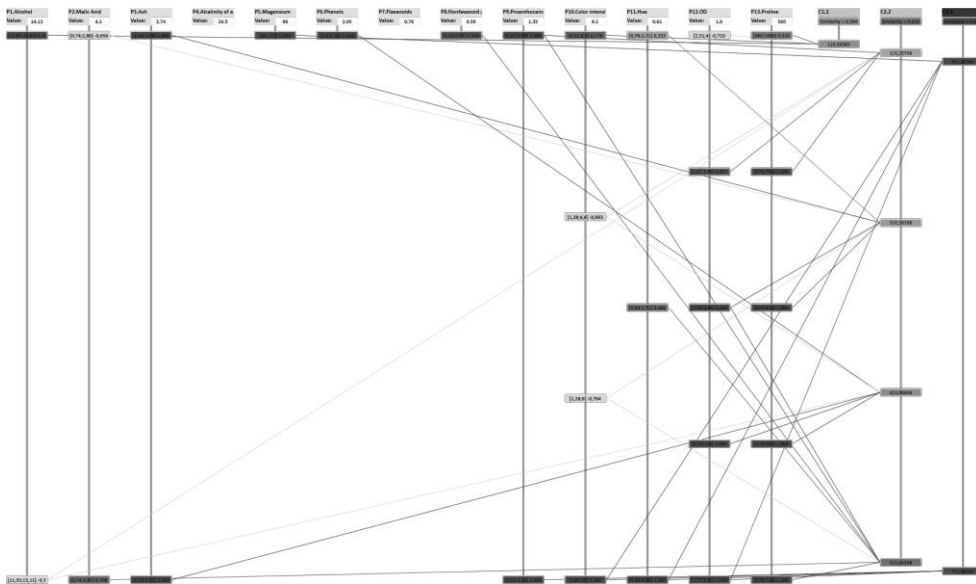


Rys. 7.17 Struktura grafu klasyfikatora ASONN wygenerowana przez aplikację komputerową dla zbioru Cars z ML Repository składająca się z 33 neuronów receptorycznych, 77 neuronów kombinacji i 4 efektorów.

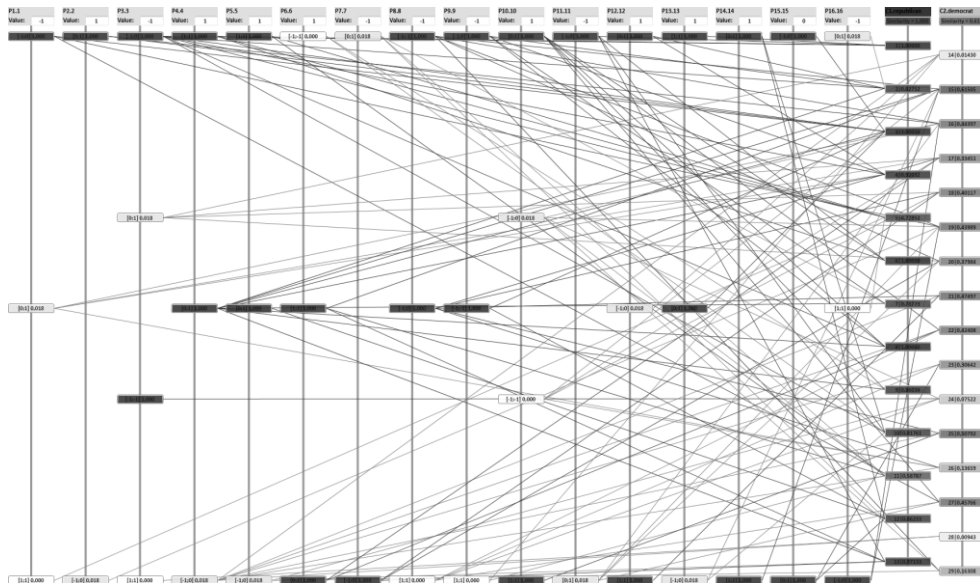
Kolejny rysunku 7.18 prezentuje strukturę sieci ASONN dla zbioru Wine z ML Repository, składającego się z 178 wzorców uczących zdefiniowanych przy pomocy 13 parametrów. Klasyfikator zawiera 19 neuronów receptorycznych, 5 neuronów kombinacji oraz 3 efektory zbudowane przez graf AANG, umożliwiające w pełni poprawną i jednoznaczną klasyfikację wszystkich wzorców uczących, jak w poprzednim przypadku. Automatyczna optymalizacja atrybutów w trakcie tworzenia klasyfikatora spowodowała wykorzystanie tylko 8 z 13 atrybutów, dla których wartości określono w zbiorze uczącym. Klasyfikator zredukował 5 pozostałych atrybutów: 4. *Alcalinity*, 5. *Magnesium*, 6. *Phenols*, 8. *Nonflavanoid* i 9. *Proanthocain*.



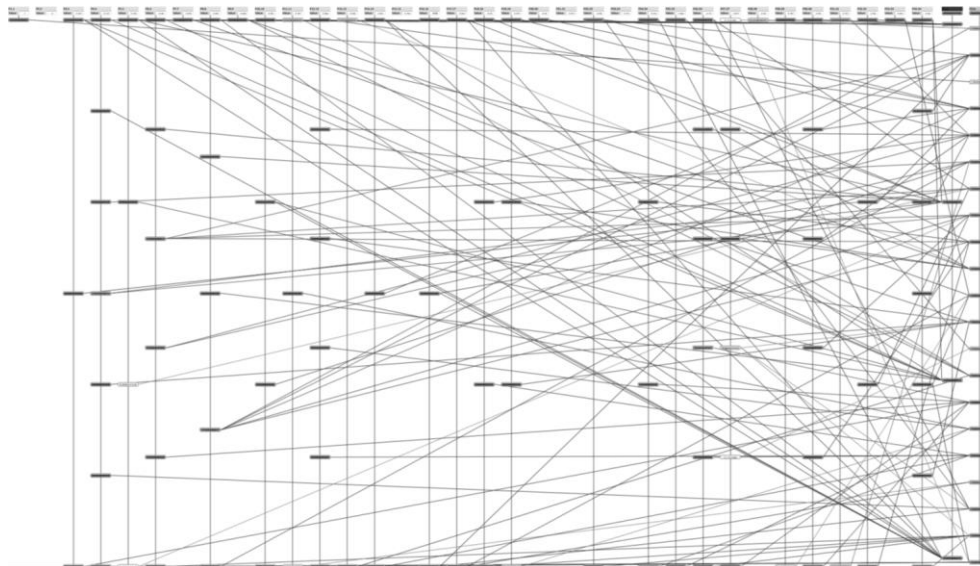
Rys. 7.18 Struktura grafu klasyfikatora ASONN wygenerowana przez aplikację komputerową dla zbioru Wine z ML Repository składająca się z 19 receptorów, 5 neuronów kombinacji i 3 efektorów.



Rys. 7.19 Struktura grafu klasyfikatora ASONN wygenerowana przez aplikację komputerową dla zbioru Wine z ML Repository składająca się z 28 neuronów receptorycznych, 7 neuronów kombinacji i 3 efektorów. Została ona zbudowana po znacznym zwiększeniu kosztu pozyskania atrybutu P_7 (*Flavonoids*) (7.31), (7.32.) i (7.33), który stanowił trzon działania klasyfikatora ASONN z rysunku 7.18, gdyż utworzono dla niego największą ilość dobrze dyskryminatywnych neuronów receptorycznych. Po zwiększeniu jego kosztu nie został on wykorzystany do budowy klasyfikatora. 5 neuronów receptorycznych dla atrybutu P_7 z rysunku 7.18 zostało więc zastąpionych poprzez inne neurony receptoryczne dla pozostałych atrybutów i inne ich kombinacje, umożliwiając dalej poprawną klasyfikację.



Rys. 7.20 Struktura grafu klasyfikatora ASONN wygenerowana przez aplikację komputerową dla zbioru Congress Voting z ML Repository składająca się z 45 neuronów receptorycznych, 29 neuronów kombinacji i 2 efektorów.



Rys. 7.21 Struktura grafu klasyfikatora ASONN wygenerowana przez aplikację komputerową dla 351 wzorców ze zbioru Ionosphere z ML Repository składająca się z 103 neuronów receptorycznych, 25 neuronów kombinacji i 2 efektorów. ASONN sklasyfikował tylko 350 wzorców poprawnie ze względu na to, iż w tym podzbiornie występowały dwa wzorce sprzeczne, z których jeden został uznany za poprawny po zbudowaniu klasyfikatora w oparciu o pozostałe 349 wzorców reprezentowane przez niezahibernowane neurony wzorców.

Następny rysunek 7.19 przedstawia strukturę sieci ASONN dla tego samego zbioru Wine. Dla zilustrowania optymalizacji atrybutów w sensie ekonomicznym przyjęto, iż koszt parametru P_7 jest bardzo wysoki (7.31), (7.32.) i (7.33). Klasyfikator ASONN został więc zbudowany bez jego wykorzystania w stosunku do struktury grafu pokazanej na rysunku 7.18. W trakcie optymalizacji struktury ASONN graf AANG wygenerował tym razem 7 neuronów kombinacji dla 28 neuronów receptorycznych. W tym przypadku klasyfikator również w 100% poprawnie i jednoznacznie sklasyfikował wszystkie wzorce uczące.

Rysunki 7.20-21 przedstawiają struktury sieci ASONN dla kolejnych zbiorów uczących Congress Voting i Ionosphere ze zbioru ML Repository. Zbiór Ionosphere zawiera 2 wzorce sprzeczne, tzn. dwa zestawy (kombinacje) identycznych wartości atrybutów definiujących dwie różne klasy. W takim przypadku graf AANG rozpoznaje tę sprzeczność ze względu na istnienie dwóch połączeń ADEF neuronu wzorca SN z neuronami klas CN . Graf AANG więc hibernuje neuron wzorca SN , który reprezentuje tę podwójną kombinację danych, oraz tworzy neurony kombinacji bez jego udziału. Utworzony klasyfikator ASONN, dokonując klasyfikacji tych dwóch sprzecznych wzorców uczących, potwierdził poprawność klasy definicyjnej jednego z nich, a drugi uznał za niepoprawny.

Tabele 7.2-5 zawierają porównania wyników uczenia i walidacji krzyżowej otrzymane dla następujących klasyfikatorów:

- **kNN – metoda k najbliższych sąsiadów** (*k nearest neighbours*) – w której zastosowano metodę głosowania na równych prawach (*majority voting*) oraz metrykę euklidesową. Polega ona na określeniu wyniku klasyfikacji na podstawie k najbliższych wzorców ze zbioru treningowego dla zastosowanej metryki [143].
- **RBFN – sieci radialnych funkcji bazowych** (*radial basis function networks*), które są sieciami bez sprzężeń zwrotnych, składających się z trzech warstw: pierwszej przekazującej sygnały wejściowe do warstwy ukrytej, w której wykorzystuje się radialne funkcje bazowe, z których w warstwie wyjściowej obliczana jest pewna suma ważona i jest obliczana wartość wyjściowa. Do porównań wykorzystano dwie funkcje radialne: funkcję gausowską oraz wielokwadratową [107,146].
- **SVN Linear – maszyna/metoda wektorów nośnych/wspierających** (*support vector machine*) **z liniową funkcją jądrową**, bazującą na odnalezieniu hiperpłaszczyzny. Metoda dzieli przestrzeń cech na podprzestrzenie, próbując zdyskryminować wzorce jednej klasy od pozostałych [1,9,122,130,169].
- **SVN RBF – maszyna/metoda wektorów nośnych/wspierających** (*support vector machine*) **z gausowską funkcją jądrową**, bazującą na odnalezieniu nieliniowej hiperpłaszczyzny. Metoda dzieli przestrzeń cech na podprzestrzenie, próbując zdyskryminować wzorce jednej klasy od pozostałych [1,9,122,130,169].
- **SVN Polynomial – maszyna/metoda wektorów nośnych/wspierających** (*support vector machine*) **z wielomianową funkcją jądrową**, bazującą na odnalezieniu hiperpłaszczyzny. Metoda dzieli przestrzeń cech na podprzestrzenie, próbując zdyskryminować wzorce jednej klasy od pozostałych [1,9,122,130,169].
- **PNN – probabilistyczne sieci neuronowe** (*probabilistic neural networks*) zainspirowane przez bayesowskie metody klasyfikacji stanowiąc ich rozwinięcie. Stosuje aproksymację funkcji gęstości prawdopodobieństwa rozkładu wektorów ze zbioru treningowego [120,147].
- **SONN-3 – samo- optymalizujące się sieci neuronowe w wersji 3** (*self optimizing neural networks*), będące poprzednikami asocjacyjnych klasyfikatorów ASONN. Strategie tworzenia i optymalizacji sieci SONN-3 i ASONN znacznie się różnią, aczkolwiek obydwie w pełni automatycznie tworzą strukturę sieci, wyznaczają parametry wagowe oraz wykorzystują mechanizm oceny stopnia przydatności poszczególnych przedziałów wartości atrybutów dla całego procesu dyskryminacji [35,54,55,59,66,74,75,76,77,78,79,80,81].
- **AANG – aktywne asocjacyjne grafy neuronowe** (*active associative neural graphs*) wykorzystane w roli klasyfikatora bazującego na ich podstawowej strukturze AGDS uzyskanej na podstawie zbioru uczącego, zaś wszystkie parametry automatycznie wyznaczono w sposób opisany w rozdziale 5.-7. tej monografii [59].
- **ASONN – asocjacyjne samo- optymalizujące się sieci neuronowe** (*associative self optimizing neural networks*) opisane rozlegle w tym rozdziale, dla których na sztywno określono pożądany stopień dyskryminacji oraz stopień rozmycia granic przedziałów definiujących hiperprostokądniany, w celu zilustrowania ich wpływu na końcowy wynik klasyfikacji i uogólniania [59].

Poszczególne klasyfikatory porównano pod kątem poprawności klasyfikacji i uogólniania wyrażonej przez różne rodzaje błędów określonych dla zbiorów uczących i 10-krotnej walidacji krzyżowej, czasu tworzenia i ewaluacji klasyfikatora oraz zajętości pamięci operacyjnej. W tym celu wykorzystano:

Procent poprawnie sklasyfikowanych wzorców (Q_COR – correct percentage):

$$Q_COR = \frac{\text{ilość wzorców sklasyfikowanych poprawnie}}{\text{ilość wzorców w zbiorze danych}} \cdot 100\% \quad (1.41.)$$

Suma kwadratów błędów (Q_SSE – Error Sum of Squares):

$$Q_SSE = \sum_{i=1}^n (f(x_i) - y_i)^2 \quad (1.42.)$$

Błąd średniokwadratowy (Q_MSE – Mean Squared Error):

$$Q_MSE = \frac{1}{n} \sum_{i=1}^n (f(x_i) - y_i)^2 = \frac{Q_SSE}{n} \quad (1.43.)$$

Pierwiastek błędu średniokwadratowego (Q_RMSE – Root Mean Squared Error):

$$Q_{\text{RMSE}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (f(x_i) - y_i)^2} = \sqrt{Q_{\text{MSE}}} \quad (1.44.)$$

Czas tworzenia klasyfikatora (E_{TTR} – *training time*) – określający, ile czasu trwał proces jego budowy, tzn. jego struktury oraz określenia wszystkich jego parametrów. Struktura może być podana z góry lub ontogenicznie tworzona w trakcie jego budowy. Określenie parametrów może przebiegać na wiele różnych sposobów, np. poprzez ich obliczenie według wzorów lub stopniowe dostrajanie w trakcie procesu uczenia.

Czas klasyfikacji (E_{TCL} – *classification time*) – określający, ile czasu trwał proces klasyfikacji niezależnie od czasu tworzenia klasyfikatora. Z punktu widzenia praktycznych zastosowań może to mieć istotne znaczenie, gdyż klasyfikator zwykle tworzony jest jednokrotnie dla pewnego zbioru danych, a następnie jest wielokrotnie wykorzystywany.

Czas całkowity operacji (E_{TOT} – *total time*) – określający, ile czasu zajęło danej metodzie utworzenie klasyfikatora i wykonanie wszystkich operacji związanych z klasyfikacją. Wskaźnik ten jest niestety również uzależniony od pracy systemu operacyjnego i wielkości dostępnej pamięci RAM, a więc uzyskane wyniki mogą być niedokładne lub częściowo przekłamane.

Do porównania klasyfikatorów wykorzystano często stosowane zbiory uczące z ML Repository:

- **Zbiór uczący Iris** składa się ze 150 wzorców treningowych, 4 atrybutów, definiujących 3 klasy, z których każda reprezentowana jest przez tą samą ilość wzorców treningowych.
- **Zbiór uczący Wine** składa się ze 178 wzorców treningowych, 13 atrybutów, definiujących 3 klasy, z których każda reprezentowana jest przez różną ilość wzorców treningowych.
- **Zbiór uczący Congressional Voting Records** składa się ze 435 wzorców treningowych, 16 atrybutów, definiujących 2 klasy, z których każda reprezentowana jest przez różną ilość wzorców treningowych.
- **Zbiór uczący Ionosphere** składa się z 29289 wzorców treningowych, 34 atrybutów, definiujących 2 klasy, z których każda reprezentowana jest przez różną ilość wzorców treningowych. Zbiór ten zawiera wiele sprzecznych wzorców. Ze względu na ograniczenia sprzętowe (pamięć RAM i szybkość działania procesora) oraz długi czas działania różnych klasyfikatorów zredukowano ilość wzorców uczących do 351, pozostawiając w nim dwa wzorce sprzeczne, tzn. należące do obu klas definicyjnych.

Na podstawie otrzymanych wyników można stwierdzić, iż klasyfikator ASONN w porównaniu do innych klasyfikatorów, dla których rozrzut wyników jest duży i zależny od parametrów uczenia oraz wybranej funkcji aktywacji neuronów, prawie zawsze osiąga dobre wyniki uogólniania, które tylko lekko się wahają w zależności od wskazanego metodzie pożądanego stopnia dyskryminacji oraz stopnia rozmycia granic hiperprostopadościanów. Działanie ASONNa jest stabilne niezależnie od danych oraz dobrze uwarunkowane.

Zadania klasyfikacji określone przy pomocy klasycznych zbiorów uczących, np. z ML Repository, zwykle są kolekcją pewnej ilości wzorców uczących, których kolejność zazwyczaj nie odgrywa żadnego znaczenia. Wzorce uczące są więc pewnymi prostymi kombinacjami danych, dla których określono pewną klasę w przypadku problemów klasyfikacji. Z tego też powodu w klasyfikatorze ASONN nie trzeba modelować asocjacyjnych relacji następstwa ASEQ i kontekstu ACON. Nie jest też potrzebna skomplikowana symulacja czasu, gdyż większość obliczeń można w prosty sposób zrealizować korzystając z grafów AGDS oraz innych pomocniczych struktur danych. Wobec tego model działania neuronów receptorycznych, neuronów, efektorów, grafów AANG oraz obliczeń asocjacyjnych dostosowano do potrzeb ASONNa i takiej klasy problemów, żeby niepotrzebnie nie zwiększać złożoności obliczeniowej ich działania na skutek symulacji czasu. Umożliwiło to sprawdzenie, czy możliwe jest stosowanie samej struktury AGDS oraz grafów AANG bez symulacji czasu i funkcji relaksacji neuronów ASN do problemów wymagających jedynie powiązań ASIM i ADEF. Klasyfikator ASONN potwierdził, że zastosowany tutaj kompromis jest skuteczny i przynoszący wymierne korzyści dla takiej klasy problemów. Osiągnięte wyniki klasyfikacji i uogólniania dla ASONN (tab.7.2-5) są zresztą porównywalne z najlepszymi wiodącymi klasyfikatorami stosowanymi na świecie w ramach inteligencji obliczeniowej. Wykorzystanie w taki sposób uproszczonego modelu obliczeń asocjacyjnych oczywiście nie przekreśla możliwości przekształcenia klasyfikatora ASONN do postaci umożliwiającej działanie na zbiorach uczących, dla których ma sens wykorzystanie asocjacyjnych powiązań ASEQ i ACON czy też wymagające zastosowania multiklasyfikacji.

W kolejnym rozdziale zaprezentowane zostaną wyniki badań dotyczące automatycznego formowania się wiedzy w grafach AANG i systemach AAS korzystających z możliwości asocjacyjnych obliczeń wykorzystujących symulację czasu oraz asocjacyjne relacje ASEQ i ACON. Badania te zapoczątkowane były badaniami nad klasyfikatorem ASONN, który również jest pewną skondensowaną formą reprezentacji wiedzy o danych zawartych w zbiorze uczącym, które go uformowały dzięki grafowi AANG.

Tab. 7.4 Porównanie wyników uczenia i 10-krotnej walidacji krzyżowej dla zbioru Congressional Voting składającego się ze 435 wzorców. Dane posortowano względem średniej walidacyjnej wartości błędu Q-ERR (avg), która jest najistotniejsza z punktu widzenia uogólniania i praktycznych zastosowań.

Algoritm i parametry	UCZENIE Q.COR (avg)	UCZENIE Q.COR (min)	UCZENIE Q.COR (max)	WALID Q.COR (avg)	WALID Q.COR (min)	WALID Q.COR (max)	UCZENIE Q.SSE (avg)	WALID Q.SSE (avg)	UCZENIE Q.MSE (avg)	WALID Q.MSE (avg)	UCZENIE Q.RMSE (avg)	WALID Q.RMSE (avg)	UCZENIE E.T.CT (avg)	WALID E.T.CT (avg)	UCZENIE E.TOT (avg)	WALID E.TOT (avg)	UCZENIE&WALID E.MEM (avg)	WALID E.FTR (avg)	
SVM Multiclass Linear (cost=50;eps=0.1)	96,78	95,91	97,70	96,55	93,02	100,00	12,60	1,50	0,032	0,035	0,179	0,162	0,024	0,032	21,702	21,694	78,568	21,692,9	
SVM Multiclass RBF (cost=10;sigma=5;eps=0.1)	98,60	98,47	98,98	96,08	90,70	100,00	5,50	1,70	0,014	0,039	0,118	0,174	0,174	0,161	61,32	60,71	78,365	60,63,9	
SVM Multiclass Linear (cost=1;eps=0.1)	97,16	96,68	97,70	96,08	88,37	100,00	11,10	1,70	0,028	0,039	0,168	0,161	0,028	0,028	51,00	50,00	78,568	49,8,6	
SVM Multiclass Linear (cost=20;eps=0.001)	97,47	96,93	97,95	96,08	86,05	100,00	9,90	1,70	0,025	0,039	0,159	0,158	0,025	0,025	13,907	13,898	78,568	13,897,3	
SVM Multiclass Linear (cost=100;eps=0.1)	96,65	95,92	97,45	96,07	88,37	100,00	13,10	1,70	0,034	0,039	0,183	0,170	0,027	0,025	20,413	20,403	78,568	20,402,4	
RBFN (k=15;funk-radial=1;met.wybor.cent=2)	95,96	94,88	96,68	96,07	86,05	100,00	15,80	1,70	0,040	0,039	0,200	0,150	0,038	0,037	51	37	65,108	35,8	
SVM Multiclass Linear (cost=20;eps=0.1)	97,16	96,43	97,70	95,86	90,70	100,00	11,10	1,80	0,028	0,041	0,168	0,177	0,030	0,025	81,25	81,14	78,568	81,13,4	
SVM Multiclass Linear (cost=10;eps=0.1)	97,04	96,17	97,96	95,85	90,70	100,00	13,60	1,80	0,030	0,042	0,171	0,187	0,028	0,030	39,87	39,77	78,568	39,75,9	
RBFN (k=20;funk-radial=1;met.wybor.cent=2)	96,53	95,40	97,45	95,85	90,70	100,00	11,60	1,80	0,035	0,042	0,186	0,167	0,028	0,030	62	45	65,903	43,6	
SVM Multiclass Linear (cost=20;eps=0.01)	97,47	96,68	97,96	95,85	86,05	100,00	9,90	1,80	0,025	0,042	0,159	0,174	0,027	0,023	14,284	14,275	78,568	14,273,7	
SVM Multiclass RBF (cost=10;sigma=10;eps=0.1)	96,73	96,16	97,45	95,18	90,70	100,00	12,80	2,00	0,033	0,046	0,181	0,187	0,028	0,028	45,670	45,661	52,7533	45,659,4	
ASONN (discrimlevel=1;cutlevel=2)	100,00	100,00	100,00	95,18	90,91	100,00	62,51	7,21	0,160	0,166	0,400	0,407	0,028	0,028	20,76	20,76	78,365	20,69,1	
RBFN (k=15;funk-radial=2;met.wybor.cent=2)	95,94	95,41	96,68	95,17	90,70	100,00	15,90	2,10	0,041	0,048	0,201	0,191	0,034	0,028	17	5	63,044	4,2	
RBFN (k=10;funk-radial=1;met.wybor.cent=2)	95,12	94,13	95,92	94,92	83,72	100,00	19,10	2,20	0,049	0,051	0,221	0,202	0,034	0,035	39	27	64,313	25,4	
ASONN (discrimlevel=2;cutlevel=1)	100,00	100,00	100,00	94,72	90,70	100,00	61,01	7,14	0,156	0,164	0,395	0,405	0,032	0,030	51,549	51,538	54,2760	51,537,0	
100,00	100,00	100,00	94,72	90,70	100,00	61,02	7,12	0,156	0,164	0,395	0,404	0,034	0,030	51,056	51,044	54,2646	51,042,2		
ASONN (discrimlevel=3;cutlevel=1)	100,00	100,00	100,00	94,48	88,37	100,00	61,25	7,13	0,157	0,164	0,396	0,405	0,035	0,035	54,334	54,321	55,3826	54,319,7	
ASONN (discrimlevel=Autocutlevel=1)	100,00	100,00	100,00	94,48	88,37	100,00	61,10	7,13	0,156	0,164	0,395	0,405	0,033	0,032	54,209	54,209	55,3189	54,207,1	
ASONN (discrimlevel=3;cutlevel=2)	100,00	100,00	100,00	94,48	88,37	100,00	60,69	7,09	0,155	0,163	0,394	0,404	0,036	0,037	54,445	54,432	55,3236	54,430,6	
RBFN (k=10;funk-radial=2;met.wybor.cent=2)	94,23	92,60	95,92	94,01	83,72	100,00	22,60	2,60	0,058	0,060	0,231	0,223	0,030	0,028	14	4	62,939	2,5	
RBFN (k=5;funk-radial=1;met.wybor.cent=2)	94,05	92,35	95,15	93,32	88,37	97,73	23,30	2,90	0,060	0,067	0,243	0,250	0,033	0,035	29	18	63,518	16,0	
PNN Simple (sigma=0.1)	100,00	100,00	100,00	93,10	88,37	97,73	0,00	2,72	0,000	0,063	0,000	0,0245	0,861	0,851	337	37	12,5272	0,0	
K.nn (k=1)	100,00	100,00	100,00	92,88	88,64	97,73	0,00	3,10	0,000	0,071	0,000	0,261	0,993	0,979	389	43	65,196	0,1	
RBFN (k=10;funk-radial=2;met.wybor.cent=1)	92,95	91,56	95,15	92,86	81,40	100,00	27,60	3,10	0,071	0,071	0,265	0,243	0,034	0,025	16	4	62,917	2,6	
AAONN ()	100,00	100,00	100,00	92,63	88,37	100,00	86,45	10,04	0,221	0,231	0,470	0,480	0,435	0,439	320	169	43,2773	149,4	
K.nn (k=3)	92,49	91,84	93,88	92,39	83,72	97,73	9,11	2,08	0,023	0,048	0,153	0,212	1,143	1,142	447	50	65,196	0,0	
PNN Simple (sigma=1)	98,75	98,47	99,23	92,18	86,05	100,00	3,18	2,38	0,008	0,055	0,090	0,223	0,829	0,830	325	36	12,5272	0,0	
K.nn (k=5)	91,16	89,29	92,60	91,93	81,40	100,00	12,49	2,06	0,032	0,048	0,179	0,206	1,431	1,389	560	61	65,196	0,1	
K.nn (k=10)	91,24	90,54	92,07	90,32	81,40	97,67	15,40	2,11	0,039	0,049	0,198	0,207	2,132	2,142	835	93	65,196	0,0	
RBFN (k=5;funk-radial=2;met.wybor.cent=2)	90,96	89,51	92,35	89,87	79,07	100,00	35,40	4,40	0,090	0,101	0,300	0,290	0,027	0,037	12	3	62,834	1,1	
K.nn (k=15)	89,22	86,99	91,33	89,20	76,74	100,00	17,21	2,21	0,044	0,051	0,210	0,213	2,814	2,764	1102	120	65,196	0,0	
PNN Simple (sigma=100)	89,35	88,75	90,56	88,95	76,74	97,67	97,78	10,87	0,250	0,250	0,500	0,500	0,830	0,841	325	37	12,5272	0,0	
PNN Simple (sigma=10)	89,40	88,78	90,82	88,95	76,74	97,67	89,15	9,91	0,228	0,228	0,477	0,477	0,832	0,830	326	36	12,5272	0,0	
SVM Multiclass RBF (cost=10;sigma=1;eps=0.1)	88,48	86,73	89,80	88,94	79,07	95,45	20,51	2,45	0,052	0,057	0,229	0,224	5,352	5,368	2095	234	65,196	0,0	
SVM Multiclass RBF (cost=10;sigma=1;eps=0.01)	100,00	100,00	100,00	85,98	75,00	93,02	0,00	6,10	0,000	0,140	0,000	0,367	0,349	0,354	3144	3023	78,365	3007,6	
SVM Multiclass RBF (cost=10;d=3;eps=0.01)	100,00	100,00	100,00	78,39	67,44	90,70	0,00	9,40	0,000	0,216	0,000	0,459	0,383	0,384	6546	6413	78,365	6396,4	
SVM MulticlassP (cost=10;d=4;eps=0.01)	73,69	37,34	79,85	75,21	50,00	86,05	117,60	13,00	0,301	0,298	0,525	0,518	0,120	0,113	17,304	17,262	78,679	17,257,0	
SVM MulticlassP (cost=10;d=6;eps=0.01)	55,27	37,34	58,93	57,43	46,51	72,73	189,70	20,70	0,485	0,476	0,688	0,679	0,118	0,117	26,117	26,076	78,679	26,070,7	
SVM MulticlassP (cost=10;d=6;eps=0.01)	55,27	37,34	58,93	57,43	46,51	72,73	189,70	20,70	0,485	0,476	0,688	0,679	0,120	0,115	26,334	26,292	78,679	26,286,5	
NAJLEPSZY WYNIK:	100,00	100,00	100,00	96,55	93,02	100,00	0,00	1,50	0,000	0,035	0,000	0,150	0,024	0,023	12	3	62,834	0,0	
RBFN (k=5;funk-radial=2;met.wybor.cent=1)	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
RBFN (k=15;funk-radial=2;met.wybor.cent=1)	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
RBFN (k=20;funk-radial=2;met.wybor.cent=1)	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
RBFN (k=20;funk-radial=2;met.wybor.cent=2)	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Tab. 7.5 Porównanie wyników uczenia i 10-krotnej walidacji krzyżowej 351 wzorców ze zbioru Ionosphere składającego się ze 29289 wzorców. Dane posortowano względem średniej walidacyjnej wartości błędu Q-ERR (avg), która jest najistotniejsza

z punktu widzenia uogólniania i praktycznych zastosowań. Metoda SVM Multiclass RBF osiągnęła lepsze wyniki niż ASONN, gdyż była zoptymalizowana dla problemów multiklasyfikacji, zaś ASONN w prezentowanej tutaj wersji nie.

Ionsphere (29289 wzorców, 34 parametrów)		TRAIN	VALID	TRAIN	VALID	TRAIN	VALID	TRAIN	VALID	TRAIN	VALID	TRAIN	VALID	TRAIN	VALID	UCZENIE&VALID			
Algorytm (parametry)		Q_COR (avg)	Q_COR (min)	Q_COR (max)	Q_COR (avg)	Q_COR (min)	Q_COR (max)	Q_SE (avg)	Q_SE (min)	Q_SE (max)	Q_NSE (avg)	Q_NSE (min)	Q_NSE (max)	E_TOT (avg)	E_TOT (min)	E_TOT (max)	E_TTR (avg)	E_TTR (min)	E_TTR (max)
SVM Multiclass RBF (cost=10;sigma=1;eps=0.1)		100.00	100.00	100.00	97.14	94.30	88.57	97.14	97.14	100.00	0.00	2.00	0.00	0.231	0.368	0.379	3782	3679	109399
SVM Multiclass RBF (cost=10;sigma=5;eps=0.1)		97.06	96.52	97.47	100.00	94.30	85.71	100.00	94.30	2.00	0.029	0.057	0.171	0.216	0.249	0.254	8388	8318	109399
ASONN (discrimlevel=2;rcutlevel=2)		99.27	98.42	100.00	91.75	88.57	85.71	97.14	73.58	8.26	0.233	0.483	0.058	0.057	0.04708	0.04692	8298779	304692	304690.1
AASNN ()		100.00	100.00	100.00	91.46	85.71	85.71	97.14	69.13	8.26	0.219	0.235	0.468	0.485	2.254	2.262	1133	501	2848195
ASONN (discrimlevel=3;rcutlevel=1)		99.21	98.10	100.00	90.87	80.00	80.00	100.00	71.96	8.10	0.228	0.231	0.477	0.480	0.080	0.080	450566	9934229	450563.6
ASONN (discrimlevel=2;rcutlevel=1)		99.11	98.10	100.00	90.87	77.14	77.14	97.14	72.72	8.16	0.230	0.233	0.480	0.482	0.056	0.066	3154426	3154411	8303382
RBFN (k=15;funk-radial=2;met.wybor.cent=2)		90.88	88.92	92.09	90.32	85.71	85.71	97.14	28.80	3.40	0.091	0.097	0.302	0.303	0.071	0.063	70	50	98375
RBFN(k=15;funk-radial=2;met.wybor.cent=2)		91.04	90.19	92.09	90.31	77.14	77.14	97.14	28.30	3.40	0.090	0.097	0.299	0.297	0.071	0.066	27	7	94151
SVM Multiclass RBF (cost=10;sigma=10;eps=0.1)		94.08	93.35	94.94	90.30	82.86	80.00	97.14	18.70	3.40	0.059	0.099	0.243	0.303	0.298	0.296	3384	3300	109399
SVM Multiclass RBF (cost=10;sigma=10;eps=0.01)		96.36	95.25	98.73	90.30	80.00	80.00	97.14	9.60	3.50	0.030	0.030	0.099	0.171	0.300	0.172	0.154	2561284	2561235
SVM Multiclass Linear (cost=10;eps=0.1)		93.95	92.72	95.57	90.02	77.14	77.14	97.14	19.10	8.32	0.264	0.237	0.484	0.487	0.032	0.026	230672	230663	6401901
ASONN (discrimlevel=1;rcutlevel=1)		98.54	97.78	99.37	89.75	80.00	80.00	97.14	73.91	8.50	0.061	0.100	0.245	0.305	0.057	0.057	15527	15512	109890
RBFN (k=20;funk-radial=2;met.wybor.cent=1)		91.52	90.82	92.72	89.74	80.00	80.00	97.14	26.80	3.60	0.085	0.103	0.291	0.309	0.074	0.080	30	10	94234
RBFN (k=15;funk-radial=2;met.wybor.cent=1)		91.04	87.66	92.41	89.73	82.86	77.14	97.14	28.30	3.60	0.090	0.103	0.299	0.308	0.070	0.068	27	7	94129
SVM Multiclass Linear (cost=50;eps=0.1)		92.72	91.46	93.99	89.73	77.14	77.14	94.44	23.00	3.60	0.073	0.103	0.269	0.312	0.054	0.063	54962	54948	109890
ASONN (discrimlevel=3;rcutlevel=3)		99.15	98.10	100.00	89.17	77.14	77.14	100.00	73.46	8.26	0.233	0.235	0.482	0.485	0.081	0.083	363591	363569	9876919
RBFN(k=20;funk-radial=2;met.wybor.cent=2)		91.83	90.19	93.35	89.17	77.14	77.14	97.14	25.80	3.80	0.082	0.108	0.285	0.316	0.077	0.077	31	10	94256
ASONN (discrimlevel=3;rcutlevel=2)		99.21	97.78	100.00	89.17	77.14	77.14	100.00	73.04	8.21	0.231	0.234	0.481	0.484	0.082	0.071	372142	372118	9867998
SVM Multiclass RBF (cost=10;sigma=0.5;eps=0.1)		100.00	100.00	100.00	82.86	80.00	80.00	97.14	0.00	4.10	0.000	0.117	0.000	0.334	0.454	0.447	5825	5698	109399
SVM Multiclass Linear (cost=20;eps=0.1)		93.80	92.41	95.25	88.88	77.14	77.14	97.14	19.60	3.90	0.062	0.111	0.248	0.320	0.053	0.048	34023	34008	109890
SVM Multiclass Linear (cost=20;eps=0.001)		94.87	93.35	96.84	88.88	74.29	74.29	97.14	16.20	3.90	0.051	0.111	0.225	0.317	0.054	0.060	44006	43991	109890
ASONN (discrimlevel=1;rcutlevel=2)		98.54	97.78	99.37	88.61	77.14	77.14	97.14	74.11	8.34	0.235	0.238	0.484	0.487	0.036	0.028	199997	199986	6382218
RBFN (k=10;funk-radial=1;met.wybor.cent=2)		89.97	87.34	91.77	88.32	80.00	80.00	91.43	31.70	4.10	0.100	0.117	0.316	0.338	0.060	0.068	53	37	96860
SVM Multiclass RBF (cost=10;sigma=0.5;eps=0.1)		100.00	100.00	100.00	86.90	74.29	74.29	94.29	25.00	4.50	0.079	0.128	0.281	0.351	0.052	0.060	100704	100689	109890
K nn (k=1)		100.00	100.00	100.00	86.90	74.29	74.29	97.14	36.20	4.60	0.115	0.131	0.337	0.352	0.061	0.068	22	5	94024
SONN-3 (Adrian Horzyk)		97.28	94.62	100.00	86.62	77.14	77.14	94.29	6.08	4.78	0.019	0.136	0.127	0.365	0.078	0.088	278	256	349612
ASONN (discrimlevel=Autocutlevel=1)		98.96	97.78	100.00	86.59	71.43	71.43	94.44	72.23	8.19	0.229	0.233	0.478	0.483	0.194	0.188	458665	458610	14371678
PNN Simple (sigma=1)		94.30	93.67	94.94	86.33	74.29	74.29	94.29	13.87	3.95	0.044	0.113	0.209	0.331	0.935	0.915	295	32	235703
SVM Multiclass (cost=10;d=3;eps=0.01)		97.88	97.47	98.42	86.02	77.14	77.14	94.44	6.70	4.90	0.021	0.140	0.145	0.368	0.186	0.183	1838383	1838331	109713
K nn (k=3)		85.82	84.49	87.97	83.48	74.29	74.29	91.43	15.90	3.97	0.050	0.113	0.224	0.326	1.049	1.057	331	37	95698
RBFN (k=5;funk-radial=1;met.wybor.cent=2)		84.33	76.51	91.77	83.25	63.89	63.89	94.29	49.50	5.90	0.157	0.168	0.389	0.393	0.056	0.057	40	24	95345
RBFN (k=10;funk-radial=2;met.wybor.cent=2)		86.99	73.42	93.35	82.90	62.86	62.86	94.29	41.10	6.00	0.130	0.171	0.353	0.398	0.065	0.057	24	5	94046
K nn (k=5)		82.40	80.06	86.39	80.35	68.57	68.57	91.43	23.22	4.08	0.074	0.116	0.271	0.332	1.261	1.262	398	44	95698
SVM Multiclass (cost=10;d=6;eps=0.01)		81.16	54.75	93.67	80.35	54.29	54.29	91.43	59.80	6.90	0.188	0.197	0.421	0.432	0.135	0.145	321330	321293	109713
RBFN (k=5;funk-radial=2;met.wybor.cent=1)		80.41	70.25	87.03	80.33	71.43	71.43	91.43	61.90	6.90	0.196	0.197	0.438	0.439	0.059	0.057	20	4	93919
RBFN (k=5;funk-radial=2;met.wybor.cent=2)		100.00	100.00	100.00	78.36	68.57	68.57	88.57	66.50	7.10	0.211	0.203	0.456	0.444	0.072	0.063	24	4	93941
K nn (k=10)		78.32	74.68	80.70	76.92	65.71	65.71	85.71	31.76	4.52	0.101	0.129	0.317	0.353	1.762	1.758	557	62	95698
K nn (k=15)		75.25	70.89	77.85	74.08	65.71	65.71	85.71	36.40	4.84	0.115	0.138	0.339	0.367	2.182	2.180	689	77	95698
K nn (k=30)		69.39	67.41	71.52	67.26	57.14	57.14	80.00	46.54	5.66	0.147	0.161	0.384	0.398	3.779	3.772	1194	132	95698
PNN Simple (sigma=10)		65.53	64.24	66.35	65.27	55.25	55.25	74.29	77.24	8.59	0.245	0.245	0.495	0.490	0.931	0.920	294	33	235703
PNN Simple (sigma=100)		65.31	64.24	66.35	64.98	54.29	54.29	74.29	78.96	8.77	0.250	0.250	0.500	0.500	0.931	0.920	294	32	235703
THE BEST VALUES/RESULTS:		100.00	100.00	100.00	94.30	88.57	88.57	100.00	0.00	2.00	0.000	0.057	0.000	0.216	0.032	0.026	20	4	93919