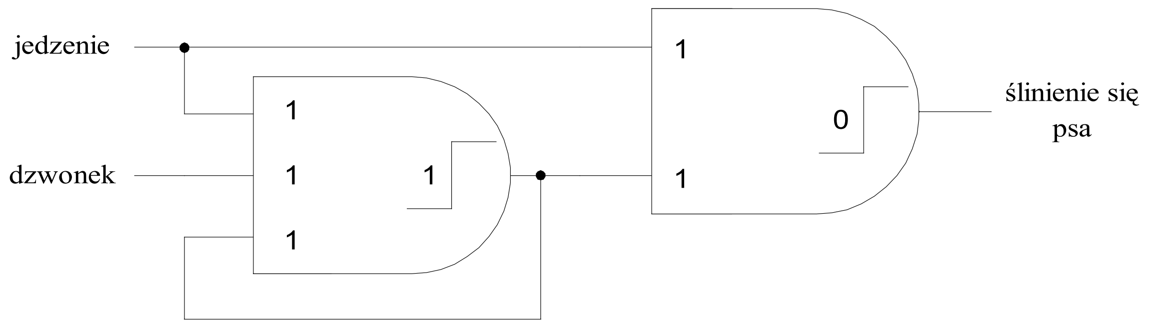
1. Wprowadzenie, sieć naśladująca „psa Pawłowa”

Czy można zbudować sieć neuronową naśladującą **PSA PAWŁOWA**? Trudno udzielić na to pytanie natychmiastowej odpowiedzi zakładając, że korzystalibyśmy tylko i wyłącznie z sieci jednokierunkowych. Pokuśmy się więc na próbę zaprojektowania czegoś na własną rękę w oparciu o poniższą tabelę i o pewne doświadczenie związane ze sposobem działania neuronów (tabela opisuje pewną sekwencję odruchów naszego psa).

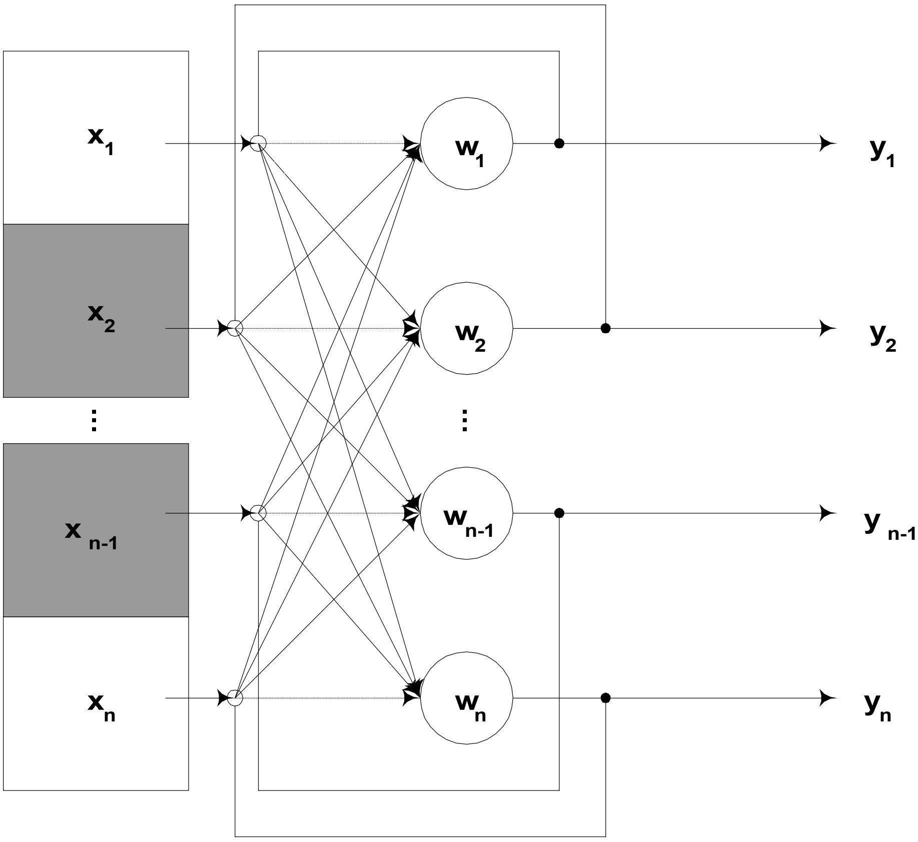
WEJŚCIE		(J,D)	WYJŚCIE	
1.	jedzenie	(1,0)	ślinienie	(1)
2.	dzwonek	(0,1)	brak reakcji	(0)
3.	jedzenie	(1,0)	ślinienie	(1)
4.	jedzenie, dzwonek	(1,1)	ślinienie	(1)
5.	jedzenie	(1,0)	ślinienie	(1)
6.	dzwonek	(0,1)	ślinienie	(1)

Sieć przedstawiona poniżej, choć nie jednokierunkowa, nie dosyć, że działa zgodnie z przedstawioną już tabelą, to ma dodatkowy atut: jest jedną z najprostszych konstrukcji realizujących przedstawiony problem. Schemat tej sieci różni się od dotychczas omówionych sieci jednokierunkowych. Nasuwa się więc pytanie, czy jest jakiś pożytek z zastosowania sprzężeń zwrotnych w sieciach neuronowych? Pomijając oczywiście podany na wstępie przykład, wszak psa po prostu można kupić... Odpowiedź na to pytanie jest twierdząca, Wyczerpujący opis różnych sieci rekurencyjnych stanowi już klasykę wśród teorii sieci neuronowych. Sieci Hopfielda opublikowane w 1982 roku wskrzesiły zainteresowanie sieciami neuronowymi.



2. Dyskretna sieć Hopfielda

Dyskretne sieci Hopfielda są przykładem najprostszych sieci neuronowych, w których zastosowano sprzężenia zwrotne. Wyjścia neuronów połączone są z wejściami tej samej warstwy. Informacja wyjściowa traktowana jest więc jako sygnał pośredni, aż do momentu spełnienia pewnego kryterium stopu. Wówczas wartości wyjść neuronów stanowią odpowiedź sieci na pierwotny sygnał wejściowy. Ze względu na charakter zastosowań sieć Hopfielda często nazywana jest pamięcią autoasocjacyjną. Sieci tego typu potrafią zapamiętać pewną liczbę wzorców i odtworzyć je, przypadku gdy pierwotny sygnał



wejściowy (dalej obraz wejściowy) jest niekompletny. Z tego też powodu najczęstszym zastosowaniem sieci tego typu jest odszumianie sygnału wejściowego.

3. Budowa dyskretnej sieci Hopfielda

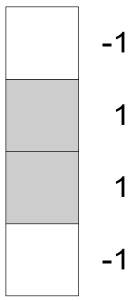
OGÓLNA CHARAKTERYSTYKA	
budowa	jednowarstwowa, ze sprzężeniem zwrotnym; neurony z dyskretną funkcją aktywacji o zerowej wartości progowej: $y_i^{(t+1)} = \begin{cases} 1 & \sum_{j=1}^{n} w_{ij} x_j > 0 \\ -1 & \sum_{j=1}^{n} w_{ij} x_j < 0 \\ y_i^{(t)} & \sum_{j=1}^{n} w_{ij} x_j = 0 \end{cases}$ pierwotny obraz wejściowy $\mathbf{x} = (x_1, x_2, \ldots, x_n)^t$ obraz wyjściowy w chwili t $\mathbf{y}^{(t)} = (y_1, y_2, \ldots, y_n)^t$
rozmiar warstwy przetwarzającej	rozmiar warstwy odpowiada rozmiarowi obrazów (n), które mają być pamiętane przez sieć.
we/wy	sygnał dyskretny (-1,1).
„uczenie”	zapamiętanie wzorcowych obrazów — każdy z obrazów do zapamiętania koryguje wagi sieci jeden raz. Sieć pozwala na uczenie inkrementacyjne.
odczyt	wielokrotne przetwarzanie przez warstwę neuronów, przy czym jako $\mathbf{y}^{(0)}$ traktujemy pierwotny sygnał wejściowy — obraz, dla którego chcemy znaleźć najbardziej podobny z tych które zostały przez sieć zapamiętane. ($\mathbf{y}^{(0)}=\mathbf{x}$). Odczyt uznajemy za zakończony, gdy żaden z neuronów nie zmienia wartości wyjścia ($\mathbf{y}^{(t+1)}=\mathbf{y}^{(t)}$).

4. Proces uczenia sieci Hopfielda

W przeciwieństwie do omówionych architektur sieci neuronowych proces uczenia sieci Hopfielda obejmuje jednorazową korektę wag dla każdego wzorca, który ma być prawidłowo przez sieć rozpoznawany. Oznaczmy wzorce do zapamiętania jako $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$,..., $\mathbf{x}^{(P)}$; na każdy ze wzorców składa się n sygnałów dyskretnych. Przez w_{ij} oznaczymy j-tą wagę w i-tym neuronie i obliczamy w następujący sposób:

$$w_{ij} = \begin{cases} \sum_{p=1}^{P} x_i^{(p)} x_j^{(p)} & i \neq j \\ 0 & i = j \end{cases} \qquad \Leftrightarrow \qquad \mathbf{W} = \left[w_{ij} \right] = \sum_{p=1}^{P} \mathbf{x}^{(p)} \mathbf{x}^{(p)^t} - p\mathbf{I} \qquad (1)$$

Poniżej jako przykład podano pewien czterobitowy wzorzec i odpowiadającą mu macierz wag sieci, w której został zapamiętany. Nie przypadkiem przedstawiona macierz wag, jest macierzą symetryczną. Każda macierz wag obliczana na podstawie wzoru (1) jest macierzą symetryczną ze względu na przemienność mnożenia.



$$\mathbf{W} = \begin{bmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & 1 & -1 \\ -1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 \end{bmatrix} \qquad (2)$$

5. Proces odczytu sieci Hopfielda

I w tym przypadku, odwrotnie niż w sieciach jednokierunkowych, obrazem wyjściowym, nie jest zbiór sygnałów otrzymanych w wyniku jednokrotnego przetworzenia sygnałów wejściowych (**$\mathbf{y_0}$=x**) przez warstwę neuronów. Ze względu na sprzężenia zwrotne sygnały uzyskane na wyjściu warstwy przetwarzającej są tylko sygnałem pośrednim ($\mathbf{y}^{(1)}$, $\mathbf{y}^{(2)}$,..., $\mathbf{y}^{(k-1)}$, $\mathbf{y}^{(k)}$) Uznane zostają jako obraz wyjściowy dopiero gdy kolejne uruchomienie wartswy neuronów nie powoduje żadnych zmian na wyjściu warstwy przetwarzającej (kryterium stopu definiowane jest więc jako $\mathbf{y}^{(k-1)}=\mathbf{y}^{(k)}$).

5.1. Stopień podobieństwa obrazów

Obraz wyjściowy jest obrazem do którego najbardziej podobny jest pierwotny wzorzec wejściowy. Stopień podobieństwa określany jest miarą Hamminga:

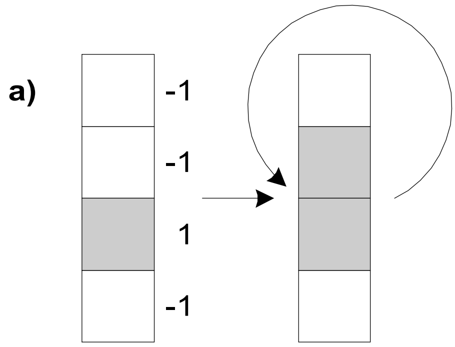
$$\rho_H(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} \tfrac{1}{2}\left|x_i - y_i\right| \qquad (3)$$

Odległość Hamminga dwóch obrazów oznacza liczbę bitów na których te dwa obrazy się różnią. O pierwotnym obrazie wejściowym sieci i odpowiadającym mu obrazie wyjściowym możemy powiedzieć, że ze wszystkich obrazów zapamiętanych przez sieć obraz wyjściowy jest najbardziej podobny do obrazu wejściowego, co oznacza, ze odległość Hamminga obrazów wejściowego i wyjściowego jest najmniejsza.

$$\rho_H(\mathbf{x},\mathbf{y}^{(k)}) = \min_{1\le p\le P} \rho_H(\mathbf{x},\mathbf{x}^{(p)}) \qquad (4)$$

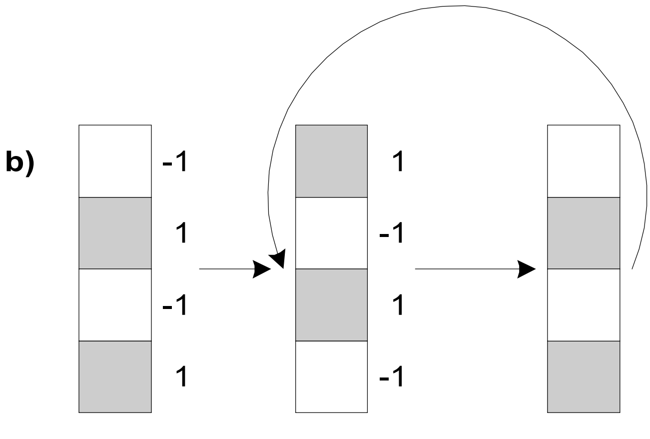
5.2. Uruchamianie sieci synchronicznie

Zacznijmy od uruchomienia sieci, której wagi policzyliśmy w punkcie 2.2 dla wzorca, który różni się od zapamiętanego na jednej pozycji.



Uruchomimy sieć zgodnie z naszym dotychczasowym doświadczeniem, tak jak sieci jednokierunkowe. W pierwszym kroku z czterech neuronów tylko drugi zmieni wartość sygnału wyjściowego. Kolejne uruchomienia całej warstwy nie powodują zmian na wyjściu sieci. Zbiór sygnałów wyjściowych traktujemy więc jako obraz wyjściowy. Jest on identyczny z obrazem, który miał być zapamiętany w procesie uczenia sieci. Uruchomienie sieci w ten sposób nazywa się synchronicznym, wszystkie neurony warstwy w każdym kroku uaktualniają wartość swojego wyjścia.

Uruchomienie synchroniczne dla obrazu wejściowego różniącego się od zapamiętanego w sieci wzorca na więcej niż jednej pozycji niestety nie jest zgodne z naszymi oczekiwaniami.



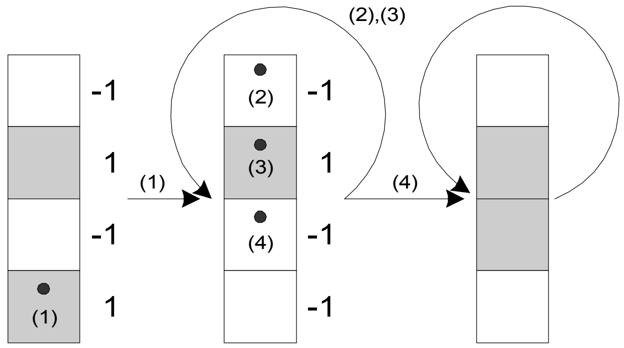
Nie dosyć, że kryterium stopu nigdy nie jest spełnione, każde kolejne uruchomienia sieci powoduje jakieś zmiany w sygnałach wyjściowych neuronów. Dodatkowo obrazy pośrednie tworzą cykl pewnego obrazu i jego negatywu (obrazów komplementarnych). Obraz ten, co gorsza nie ma żadnego związku z zapamiętanym przez nas obrazem.

Punkt ten miał na celu pokazanie jak nie należy uruchamiać sieci Hopfielda. Charakter pracy sieci Hopfielda, proces dochodzenia od wzorca wejściowego poprzez obrazy pośrednie do obrazu wyjściowego stanowiącego jeden z obrazów zapamiętanych w sieci wymaga uruchamiania asynchronicznego.

5.3. Uruchamianie sieci asynchronicznie

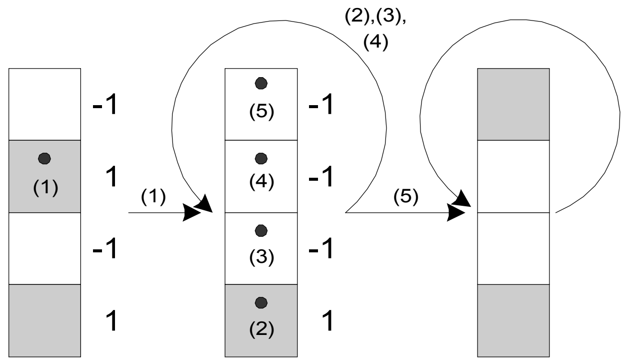
Uruchamianie asynchroniczne sieci oznacza, że w jednym kroku, aktualizuje wartość swego wyjścia tylko jeden, z reguły losowo wybrany neuron.

c)

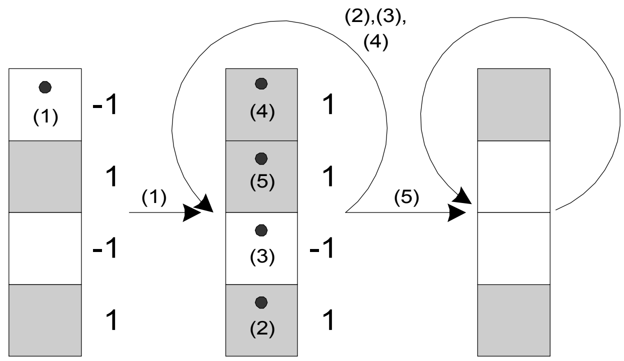


Przyjrzyjmy się więc kilku przykładowym uruchomieniom dla tego samego obrazu wejściowego. Na rysunkach, kropki z numerami w poszczególnych elementach składowych obrazów oznaczają, w którym kroku odpowiedni neuron uaktualniał wartość swego wyjścia.

d)



e)

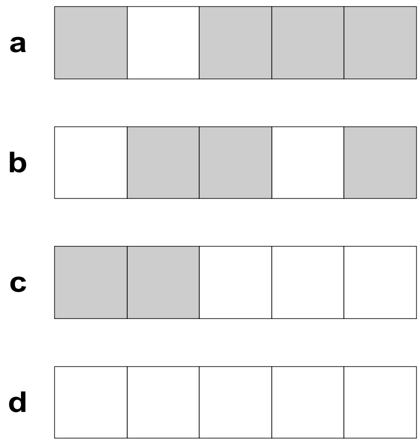


Z otrzymanych wyników, możemy dostrzec, że pomimo faktu, że w sieci zapamiętaliśmy tylko jeden wzorzec (p.2.2.), obrazem wyjściowym jest albo ów wzorzec, albo obraz stanowiący jego negatyw, co sugerowałoby, że w sieci zapamiętaliśmy są dwa wzorce. Konsekwencją charakteru zapisu wag w sieci Hopfielda jest fakt, że sieć zapamięta nie tylko wzorzec, który chcemy zapamiętać, ale także jego komplementarny obraz (negatyw wzorca), oba obrazy powodują taką samą korektę wag. Wpływ na to, który z wzorców otrzymamy jako obraz wyjściowy ma sekwencja wyborów, który z neuronów ma uaktualnić stan swego wyjścia.

6. Zapamiętane wzorce a pamiętane wzorce w sieci Hopfielda

Pytanie jakie pojawia się, gdy w sieci pamiętamy więcej niż jeden wzorzec jest następujące. Czy wszystkie wzorce, które zapamiętaliśmy w sieci poprzez odpowiednią korektę wag są pamiętane? Wzorzec jest pamiętany (praktyczny sposób) gdy w odpowiedzi na obraz wejściowy identyczny z owym wzorcem, obraz wyjściowy jest taki sam jak obraz wyjściowy. Nie ma bowiem, żadnego innego wzorca z zapamiętanych, który mogłby być “bardziej podobny”, różni się bowiem od obrazu wejściowego minimum jednym sygnałem składowym. Wzorce pamiętane przez sieć nazywamy atraktorami, stan w którym znajduje się sieć, gdy na wyjściu jest wzorzec pamiętany, stanem stabilnym.

Zastosujmy ten praktyczny sposób do przykładów przedstawionych poniżej. Najpierw weźmy pod uwagę sieć która ma za zadanie zapamiętać wzorce *a,b,c*. W odpowiedzi na każdy z trzech wzorców, jako obraz wyjściowy otrzymujemy odpowiedni wzorzec. Każdy z trzech wzorców jest pamiętany.



$$\mathbf{W}_{a,b,c} = \begin{bmatrix} 0 & -1 & -1 & 1 & -1 \\ -1 & 0 & -1 & -3 & -1 \\ -1 & -1 & 0 & 1 & 3 \\ 1 & -3 & 1 & 0 & 1 \\ -1 & -1 & 3 & 1 & 0 \end{bmatrix}$$

$$\mathbf{W}_{a,b,c,d} = \begin{bmatrix} 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 2 & 4 \\ 2 & -2 & 2 & 0 & 2 \\ 0 & 0 & 4 & 2 & 0 \end{bmatrix}$$

Weźmy teraz pod uwagę sieć, która ma zapamiętać obrazy *a,b,c,d*. Okazuje się, że wzorce *a,b* są pamiętane. Nie zostały zapamiętane wzorce *c,d*. Wzorzec *c*, bez względu na sekwencję wyborów neuronów uaktualniających wyjście,rozpoznawany jest jako negatyw wzorca *a*. Podobnie atraktorem wzorca *d* jest także obraz komplementarny do wzorca *a*. Podsumowując wzorce *a,b* są stabilne, natomiast *c,d* – niestabilne.

Należy więc zdawać sobie sprawę z faktu, że zapamiętanie wzorców nie oznacza jednocześnie, że wzorce te będą pamiętane. Znalezienie przykładów, gdy sieć nie pamięta części zapamietanych wzorców nie stanowi problemu, wystarczy chocby wziąć ich odpowiednią ilość. W przypadku, gdy rozmiar obrazów wejściowych jest większy (>12), znaleźć można takie przykłady, że sieć pamięta wzorce, które wcale nie zostały zapamiętane.

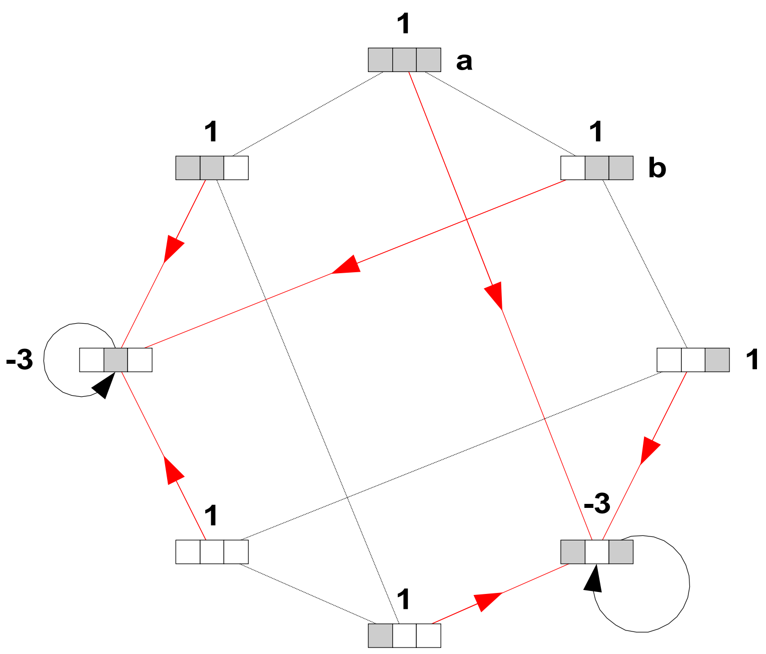
Potrzebujemy więc matematycznej podpórki, która pozwoli nam intepretować stany pośrednie, które pojawiają się między pierwotnym obrazem wejściowym a jego atraktorem.

7. Funkcja energii stanu w sieci Hopfielda

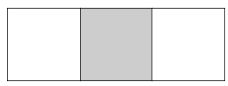
Z każdym możliwym stanem wyjścia (obraz wejściowy,obrazy pośrednie, obraz wyjściowy) możemy skojarzyć funkcję, nazwaną ze względu na swe własności *funkcją energetyczną*. Ponieważ w trakcie odczytu (uruchamiania asynchronicznego sieci) zmiany wartości funkcji są niedodatnie w książkach można spotkać nazwę *funkcja Lapunowa*, w związku z traktowaniem sieci jako układu dynamicznego.

$$E \equiv -\tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i w_{ij} y_j \qquad \Leftrightarrow \qquad E \equiv -\tfrac{1}{2}\mathbf{y}^t\mathbf{W}\mathbf{y} \qquad (5)$$

Stabilny obraz (pamiętany obraz), to taki obraz, który w swoim sąsiedztwie ma minimalną wartość energii. Przez sąsiedztwo rozumiemy te obrazy, których odległość Hamminga od stabilnego obrazu wynosi jeden. Poniżej przedstawiono graf z wszystkimi możliwymi stanami i wartości energii obliczonych według wzoru (5), obrazy połączone są obrazami, które różnią się wartością na jednej pozycji.



zapamiętany obraz



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

Asynchroniczny algorytm aktualizacji wyjść neuronów w procesie odczytu zawsze prowadzi do stanu o minimalnej energii. W przykładach przedstawionych w punkcie 2.3 można było odnieść wrażenie, że gdy sekwencja wyborów kolejnych neuronów uaktualniających wartość wyjścia, będzie jeszcze inna niż te przedstawione otrzymamy na wyjściu inny obraz. Obraz pośredni ulegnie zmianie tylko wtedy, gdy zmianie wartości wyjścia towarzyszy zmiana wartości funkcji energii, na niższą. Na powyższym rysunku przedstawiono wszystkie możliwe przejścia. Nie jest możliwe, aby obrazem pośrednim był obraz *a* czy *b*, z tego względu, że nie ma żadnego obrazu (odległego o jeden), którego wartość energii byłaby mniejsza niż jeden. Z grafu łatwo odczytać jakie obrazy są przez sieć pamiętane, wzorzec, który miał być zapamiętany i jego obraz komplementarny.

7.1 Własności funkcji energii

W punkcie tym pokażemy, że zmianie dowolnej składowej obrazu pośredniego zawsze towarzyszy zmniejszanie się wartości funkcji energii obrazu, co pozwala powiedzieć, że

$$E(\mathbf{y}) = -y_i \sum_{j=1}^{n} w_{ij} y_j - \tfrac{1}{2} \sum_{l=1}^{n} \sum_{j \neq i} w_{lj} y_l y_j \qquad (6)$$

stabilnymi obrazami są te, które znajdują się w minimach lokalnych funkcji energii. Ponieważ w pojedyńczym kroku zmiana obrazu pośredniego obejmuje co najwyżej jedną składową obrazu, funkcję energii (5) możemy zapisać rozdzielając sumę na czynnik zależny i niezależny od składowej, której odpowiadający neuron w kolejnym kroku zmieni wartość wyjścia zmieni wartość:

Rozpatrzmy teraz wpływ zmiany wartości składowej obrazu pośredniego (u_i oznacza pobudzenie łączne *i*-tego neuronu):

$$\begin{aligned} \Delta E = E(\mathbf{y}^{k+1}) - E(\mathbf{y}^{k}) &= -(y_i^{k+1} - y_i^{k}) \sum_{j=1}^{n} w_{ij} y_j \\ &= -(y_i^{k+1} - y_i^{k}) u_i = -\Delta y_i u_i \end{aligned} \qquad (7)$$

Możliwe są trzy przypadki dotyczące zmiany wartości wyjścia *i*-tego neuronu:

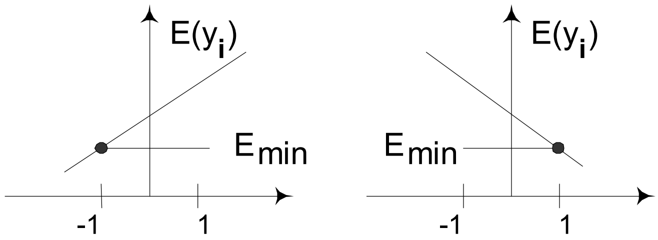
a) $\boldsymbol{\Delta E<0}$, stąd $\Delta y_i u_i>0$, z czego wynika $u_i \neq 0$. Jeśli założymy, że sygnał $y_i^{(k+1)}=1,\ y_i^{(k)}=-1$, wówczas $\Delta y_i>0$, a co za tym idzie $u_i>0$ na podstawie wzoru (7). Dodatnie pobudzenie łączne powoduje, że wartość y_i w chwili $k+1$ zmienia stan z -1 na 1.

   Jeśli założymy, że sygnał $y_i^{(k+1)}=-1,\ y_i^{(k)}=1$, wówczas $\Delta y_i<0$, a co za tym idzie $u_i<0$ na podstawie wzoru (7). Ujemne pobudzenie łączne powoduje, że wartość y_i w chwili $k+1$ zmienia stan z 1 na -1.

b) $\boldsymbol{\Delta E>0}$, stąd $\Delta y_i u_i<0$, z czego wynika $u_i \neq 0$. Jeśli założymy, że sygnał $y_i^{(k+1)}=1,\ y_i^{(k)}=-1$, wówczas $\Delta y_i<0$, a co za tym idzie $u_i<0$ na podstawie wzoru (7). Ujemne pobudzenie łączne powoduje, że wartość y_i w chwili $k+1$ nie zmienia stanu. Analogicznie, nie nastąpi zmiana wartości y_i, gdy $y_i^{(k+1)}=-1,\ y_i^{(k)}=1$.

c) $\boldsymbol{\Delta E=0}$, stąd $\Delta y_i u_i=0$, gdy $u_i=0$ nie nastąpi zmiana stanu ze względu na definicję funkcji aktywacji (patrz p.3). Gdy $u_i \neq 0$, wówczas, $y_i^{(k+1)}= y_i^{(k)}$, co oznacza zachowanie poprzedniego stanu.

Uwzględniając istnienie zerowej przekatnej macierzy wag, można powiedzieć, że funkcja energii, jest funkcją liniowo zależną od składowej y_i obrazu pośredniego. Funkcja ta nie ma więc minimum. Jednak przy założeniu, że sygnał y_i przyjmuje ograniczone wartości (±1), to funkcja ta ma swoje minimum gdy wartość sygnału y_i wynosi 1 przy jednoczesnie dodatniej wartości pochodnej (stała pochodna) oraz gdy wartość sygnału y_i wynosi -1 przy jednocześnie ujemnej wartości pochodnej (rysunek poniżej).



Można więc zdefinować kryterium określające, czy dany stan **y** jest stabilny:

$$\left. \left(y_i \frac{\partial E}{\partial y_i} \right) \right|_{\mathbf{y}} \le 0, \qquad i = 1,2,\ldots,n \qquad (8)$$

8 Pojemność pamięci autoasocjacyjnej

Pytanie, które mogło nasunąć się już w przykładzie z czterema wzorcami, gdzie tylko połowa została prawidłowo zapamiętana brzmi, ile obrazów można efektywnie zapamiętać? Poniżej przedstawiono analizę pojemności sieci.

Załóżmy, że w sieci zapamiętano *m* wzorców. Wówczas spodziewamy się , że łączne pobudzenie *i*-tego neuronu (w stanie stabilnym), gdy obrazem wejściowym jest jeden z zapamiętanych wzorców (*m*-ty wzorzec) jest równe zero, lub takiego samego znaku, co i-ta składowa wzorca $x^{(m)}$.

$$u_i = \sum_{j=1}^{n} w_{ij} x_j^{(m)} = \sum_{j=1}^{n} \sum_{p=1}^{P} x_i^{(p)} x_j^{(p)} x_j^{(m)} = \sum_{j=1}^{n} x_i^{(m)} x_j^{(m)} x_j^{(m)} + \sum_{j=1}^{n} \sum_{\substack{p=1 \\ p \ne m}}^{P} x_i^{(p)} x_j^{(p)} x_j^{(m)} =$$

$$= n y_i^{(m)} + \sum_{j=1}^{n} \sum_{\substack{p=1 \\ p \ne m}}^{P} x_i^{(p)} x_j^{(p)} x_j^{(m)} \qquad (9)$$

Oczekujemy, aby znak u_i był taki sam jak *i*-ta składowa obrazu $\mathbf{x}^{(m)}$ (*i=1,2,...,n*). Stąd na podstawie (9) widać, że *i*-ta składowa zachowuje swoją wartość, gdy znak sumy jest zgodny ze znakiem *i*-tej składowej, lub gdy jest znaku przeciwnego, ale wartość sumy jest mniejsza od *n*. Wprowadzić możemy parametr nazywany *przesłuchem*.

$$c_i^{(m)} = -\tfrac{1}{n} x_i^{(m)} \sum_{j=1}^{n} \sum_{\substack{p=1 \\ p \ne m}}^{P} x_i^{(p)} x_j^{(p)} x_j^{(m)} \qquad (10)$$

Gdy wartość przesłuchu jest mniejsza od jedynki, i-ta składowa obrazu $\mathbf{x}^{(m)}$ jest stabilna. Przy oszacowywaniu tej wartości, zakłada się, że wzorce są niezależnymi dyskretnymi zmiennymi losowymi, przyjmującymi wartości ± 1 z prawdopodobieństwem 0.5. Wielu autorów przeprowadziło złożone analizy pojemności sieci. Krytyczną zależność przedstawili w 1985 roku Amit, Gutfrund i Sompolinsky, podając że $P<0{,}138n$. Dla większych wartości P, niewielka liczba niestabilnych składowych prowadzi do zmiany pozostałych składowych, dając w wyniku błędne odczyty. Na uwagę zasługują zależności pojemności sieci od rozmiaru wzorca, przy dodatkowym założeniu, że wszystkie wzorce mają być pamiętane z prawdopodobieństwem bliskim jedności. McEliece, Posner, Rodemich, Venkatesh (1987) podali oszacowania dla dla dużych wartości n (11).

Podali również rozszerzony wzór określający pojemność pamięci (12). Wzór określa liczbę wektorów wzorcowych P, do których zbieżne będą wektory wejściowe znajdujące się w odległości nie większej niż ρ od wektorów wzorcowych (z prawdopodobieństwem bliskim jedności).

$$P < \frac{n}{4\ln n} \qquad (11) \qquad\qquad P < \frac{(1-2\rho)^2 n}{4\ln n} \qquad (12)$$