

ЛАБОРАТОРНА РОБОТА №5

Тривалість роботи – 2 години.

Графічні функції

hintonw(W,maxw,minw) – функція повертає хінтоновський графік матриці вагів, при якому кожен ваговий коефіцієнт відображається квадратом з площею, пропорційній величині даного коефіцієнта. Знак відображається кольором квадрата.

Аргументи:

W – матриця вагів; **maxw** і **minw** – мінімальне і максимальне значення її коефіцієнтів (можуть не задаватися).

Приклад

```
»W = rands(2,3)
```

W=

```
-0.7495  0.8228  0.2340
```

```
-0.6677 -0.7275 -0.4620
```

```
» hintonw(W) (рис. 1.4).
```

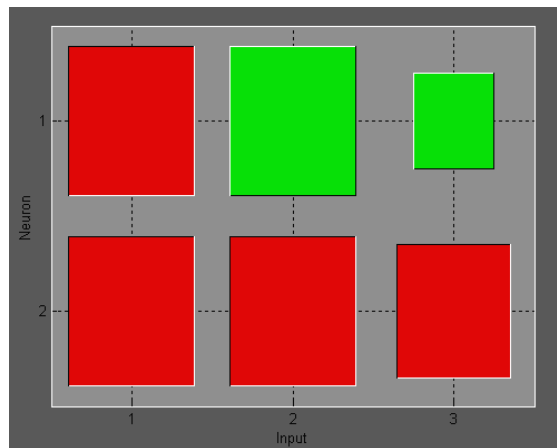


Рис. 1.4. Ілюстрація до виконання функції **hintonw**

hintonwbM(W,b,maxw,minw) – те ж, що і попередня функція, але на графіці відображаються не тільки ваги, але і зсув.

plotbr(tr,name,epoch) – функція повертає графіки зміни критерію якості НМ в процесі навчання при використанні Баєсівського методу.

Аргументи:

tr – запис процесу навчання, **name** – ім'я НМ, **epoch** – кількість циклів навчання (за умовчанням – довжина запису навчання).

Приклад

```
»p = [-1:.05:1];
```

```
»t = sin(2*pi*p) + 0.1*randn(size(p));
```

```
»net = newff([-1 1],[20 1], {'tansig','purelin'},'trainbr'); %Створення нової мережі
```

```
»[net,tr] = train(net,p,t); % Навчання мережі
```

TRAINBR, Epoch 0/100, SSE 228.933/0, SSW 21461.7, Grad 2.33e+002/1.00e-010, #Par 6.10e+001/61

TRAINBR, Epoch 25/100, SSE 0.235423/0, SSW 211.044, Grad 9.43e-002/1.00e-010, #Par 1.35e+001/61

TRAINBR, Epoch 50/100, SSE 0.240881/0, SSW 121.647, Grad 1.87e-001/1.00e-010, #Par 1.23e+001/61

TRAINBR, Epoch 75/100, SSE 0.239867/0, SSW 116.884, Grad 1.62e-002/1.00e-010, #Par 1.22e+001/61

TRAINBR, Epoch 100/100, SSE 0.239762/0, SSW 116.871, Grad 9.60e-003/1.00e-010, #Par 1.22e+001/61

TRAINBR, Maximum epoch reached.

» plotbr(tr) (рис 1.5).

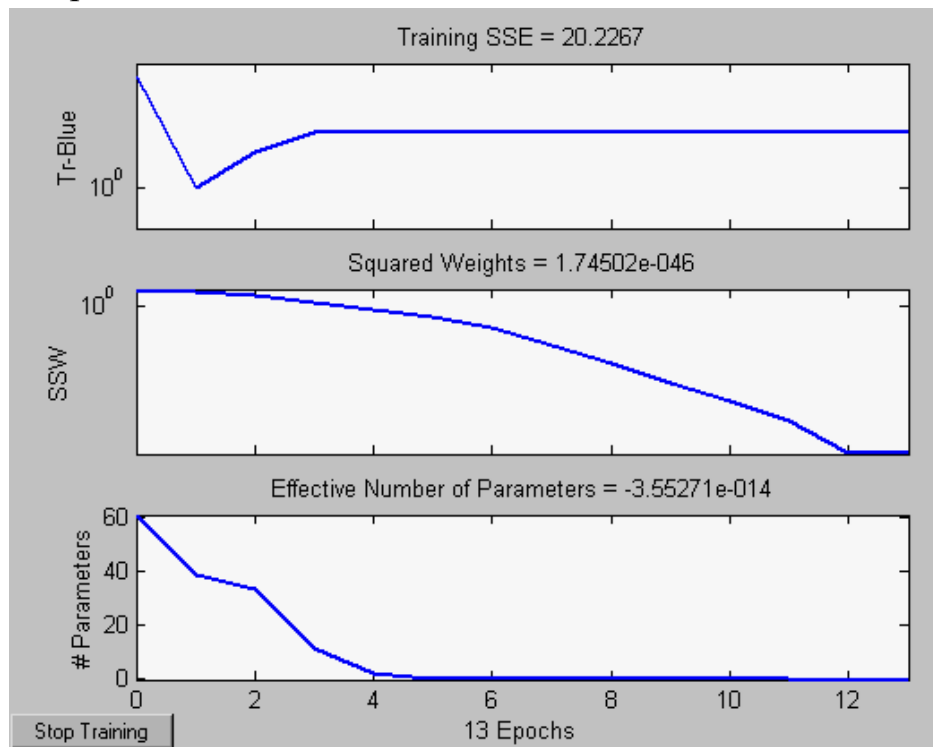


Рис. 1.5. Результат виконання функції **plotbr(tr)**

plotep(w,b,e) – функція відображає позиції вагів і зміщень на поверхні помилки НМ.

Аргументи:

w, **b**, **e** – відповідно, матриці вагів, зсувів і помилок. Повертається вектор, використаний для реалізації графіку, створеного функцією **plotes**.

plotes(wv,bv,e,v) – функція повертає графік поверхні помилки нейрону із одним входом.

Аргументи:

wv, **bv** – відповідно, набори значень ваги і зсуву нейрона, **e** – матриця значень помилки, **v** – опція виду зображення (за умовчанням [-37,5, 30]).

plotpc(W,b) – функція повертає графік лінії розв'язку для персептрона.

Аргументи:

W – матриця вагів, **b** – вектор зсувів. Використовується спільно з функцією **plotpv**.

plotperf(tr,goal,name,epoch) – повертає графік зміни критерію якості НМ в процесі навчання.

Аргументи: **tr** – запис процесу навчання; **goal** – цільове значення критерію; **name** – ім'я НМ; **epoch** – кількість циклів навчання.

plotpv(p,t) – функція повертає графічне відображення вхідних **p** і цільових **t** векторів перцептрона.

Приклад

» **p = [0 0 1 1; 0 1 0 1];**

» **t=[0 0 0 1];**

» **plotpv(p,t)** (рис. 1.6).

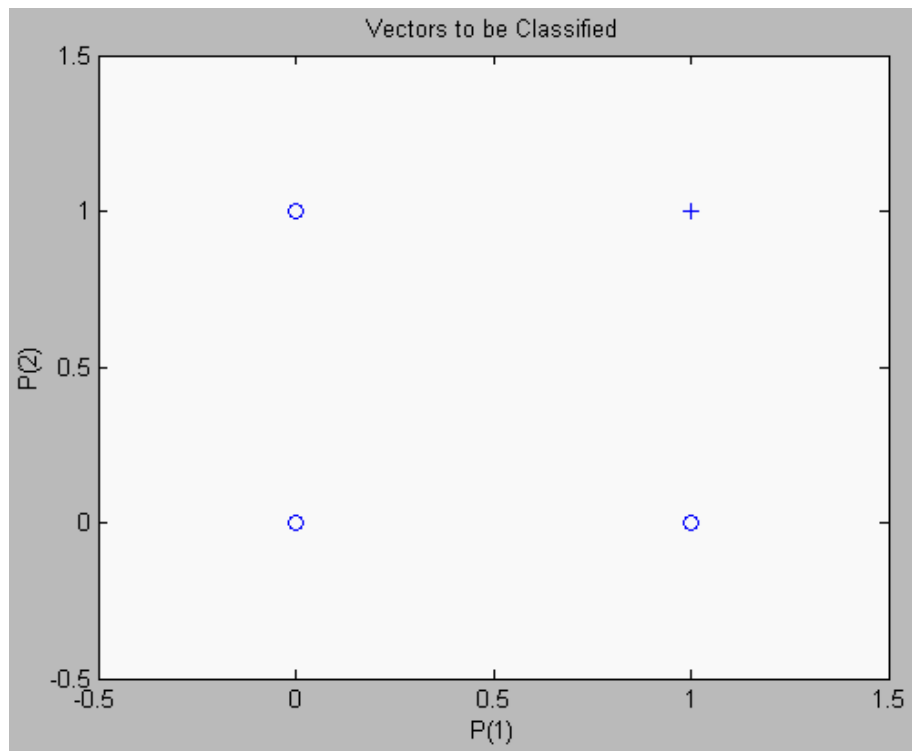


Рис1.6. Результат використання функції **plotpv(p,t)**

plotsom(pos) – функція повертає графічне представлення розташування нейронів у картах, що самоорганізуються (див. функції розміщення нейронів).

plotv(M,t) – функція графічного зображення векторів.

Аргументи: **M** – матриця з двома рядками, стовпці якої асоційовані із векторами, **t** – опція, що задає тип лінії.

Приклад

»`plotv([-0.4 0.7 .2; -0.5 .1 0.5],'-')` (рис. 1.7).

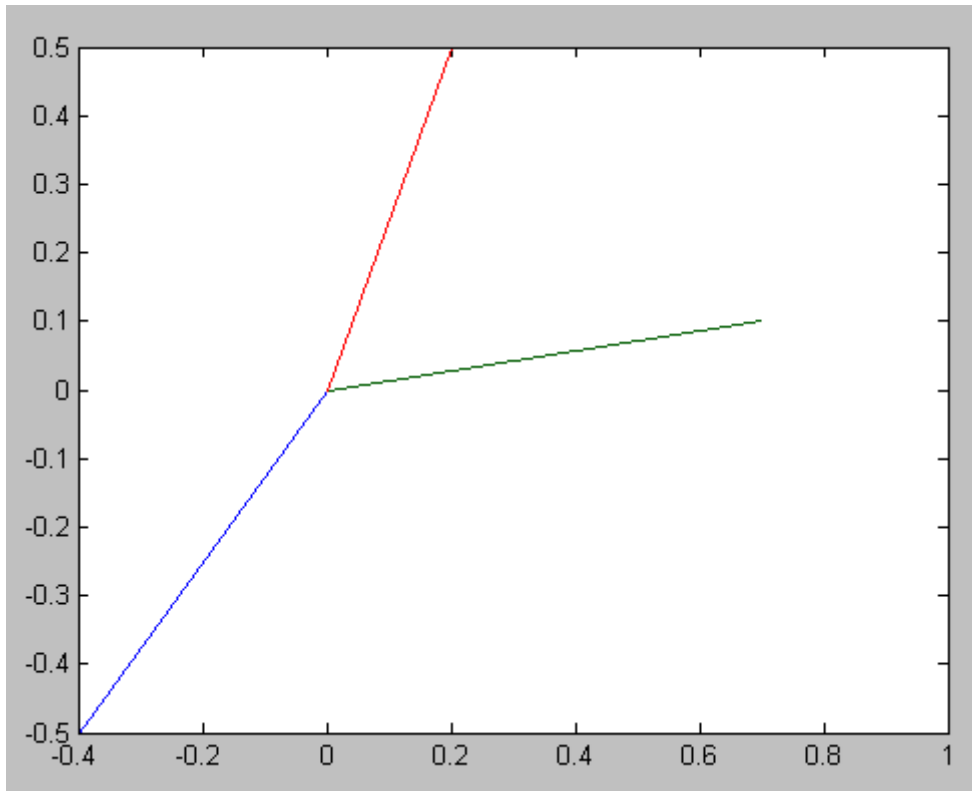


Рис. 1.7. Результат використання функції `plotv(M,t)`

`plotvec(M,C,m)` – функція графічного зображення векторів різними кольорами.

Аргументи:

M – матриця з двома рядками, стовпці якої асоціюються з векторами; **C** – рядок задання кольорів; **m** – тип точок, які вказують на кінці векторів (за умовчанням +).

Завдання

1. Реалізувати у *Neural Networks Toolbox* наведені приклади графічних функцій.
2. Згідно завдання викладача реалізувати 4 графічні функції.

Інші функції

`errsurf(p,t,wv,bv,f)` – функція, що повертає матрицю значень поверхні помилок нейрона з одним входом і одним виходом залежно від значень ваги і зсуву.

Аргументи:

p – вектор значень входу;

t – вектор значень виходу;

wv – набір значень вагів нейрона;

bv – набір значень зсуву;

f – назва функції активації, що реалізовується (рядок).

Розмір матриці, що повертається = (кількість значень bv) * (кількість значень wv).

Приклад

```
» p = [-6.0 -6.1 -4.1 -4.0 +4.0 +4.1 +6.0 +6.1];  
» t = [+0.0 +0.0 +.97 +.99 +.01 +.03 +1.0 +1.0];  
» wv = -1:.1:1;  
bv = -2.5:.25:2.5;  
» es = errsurf(p,t,wv,bv,'logsig');  
» plotes(wv,bv,es[60 30]) (рис 1.8)
```

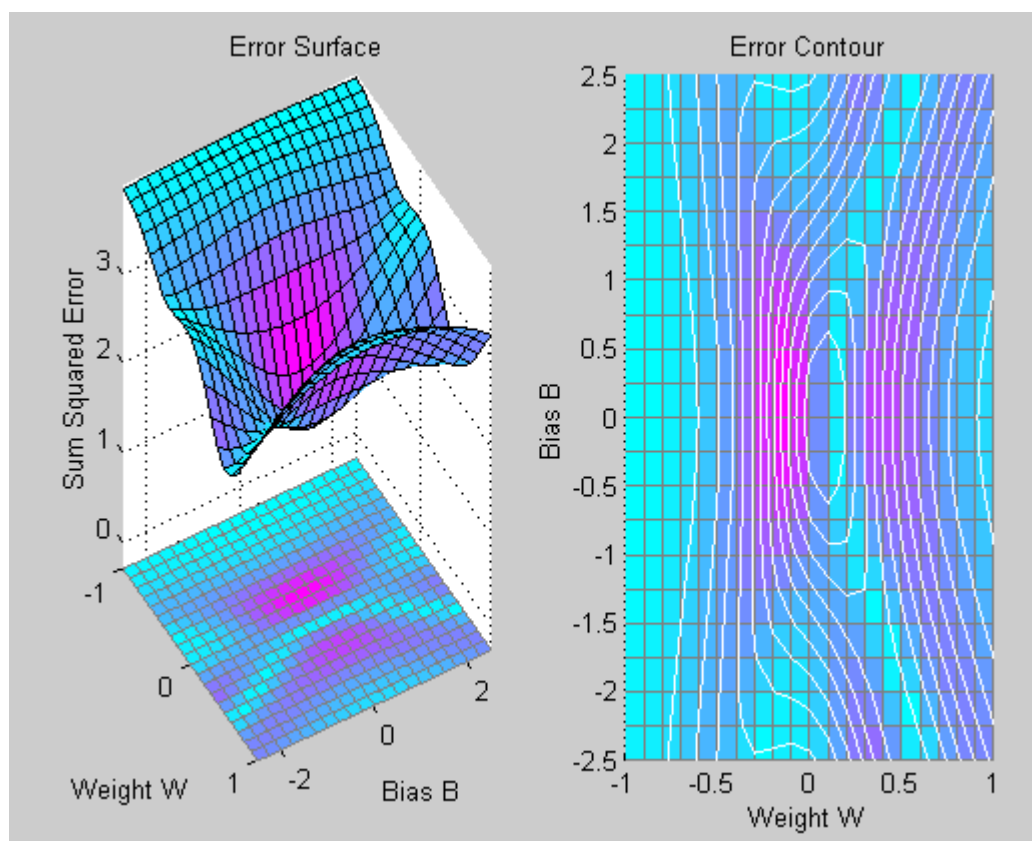


Рис. 1.8. Ілюстрація виконання функції **errsurf**

maxlinlr(P) – повертає максимальну величину коефіцієнта навчання для лінійного шару нейронів. Тут **P** – матриця входів.

При записі у формі **maxlinlr(P,'bias')** функція повертає максимальну величину коефіцієнта навчання для лінійного шару нейронів із зсувом.

gensim(net,st) – функція генерує нейромережвий блок **Simulink** (рис. 1.9) для подальшого моделювання НМ засобами цього пакету.

Приклад

```
» net = newff([0 1],[5 1]); % Створення нової НМ
```

» gensim(net)

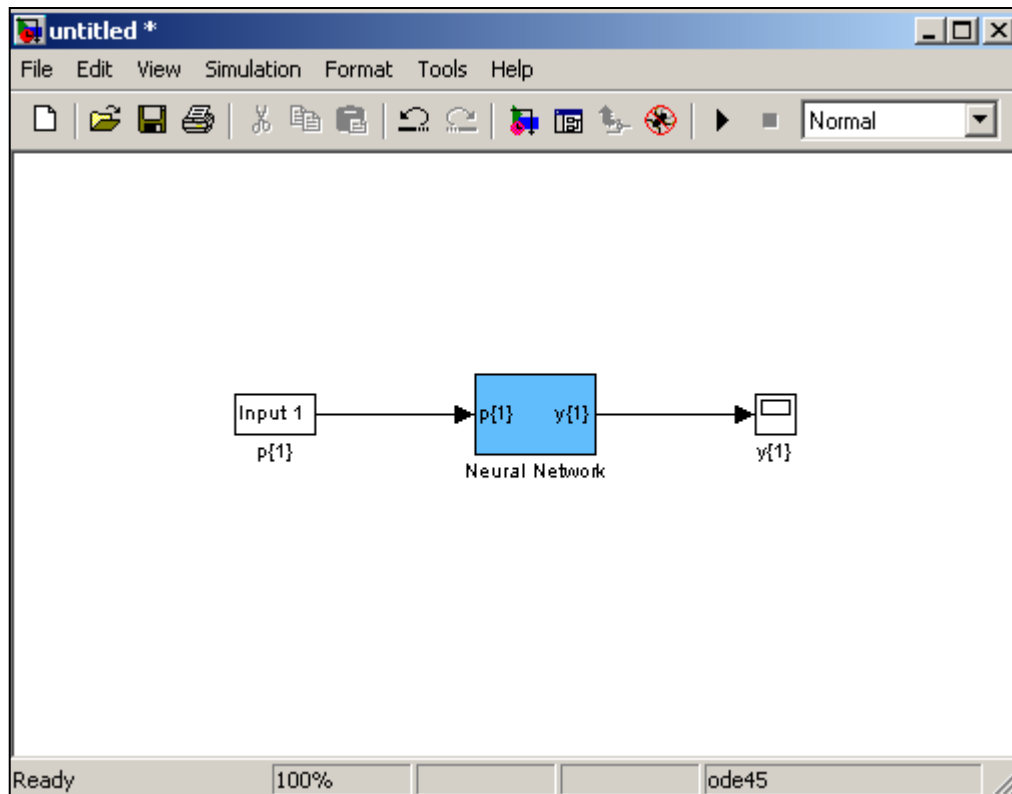


Рис. 1.9. Результат виконання функції **gensim**

initlay(net) – функція ініціалізації шарів нейронної мережі. Як аргумент використовує ім'я (ідентифікатор) **net** НМ. Повертає нейронну мережу, шари нейронів в якій ініціалізовані відповідно до функції **net.layers{i j}.initFcn**.

У формі **initlay(code)**, де змінна **code** може приймати значення **'pnames'** або **'pdefaults'**, функція повертає інформацію про імена або про значення за замовчанням параметрів ініціалізації.

initnw(net,i) – функція ініціалізації шару *p*.

initwb(net,i) – майже те ж, що у попередньому випадку, але ваги і зсув *i*-го шару ініціалізуються відповідно до їх власних функцій ініціалізації.

ddotprod – функція визначення похідної від результату **Z** множення матриці вагів **W** на матрицю входів **P**.

Запис:

```
dZ_dP = ddotprod('p',W,P,Z)
```

```
dZ_dW = ddotprod('w',W,P,Z)
```

Приклади

```
»W = [0 -1 0.2;-1.1 1 0];
```

```
»P = [0.1; 0.6;-0.2];
```

```
» Z = dotprod(W,P)% Обчислення Z=W*P
```

```
Z= -0.6400
```

```
0.4900
```

```
» dZ_dP = ddotprod('p',W,P,Z)
```

```
dZ_dP =
```

```
0      -1.0000  0.2000
-1.1000  1.0000  0
```

```
» dZ_dW = ddotprod('w',W,P,Z)
dZ_dW = 0.1000
        0.6000
       -0.2000
```

Завдання

1. Реалізувати у *Neural Networks Toolbox* наведені функції.
2. Згідно завдання викладача реалізувати 2 функції.