به نام خدا

امیرعلی احددوست

ابتدا شبکه را با patternnet(2) که دارای 2 بخش در لایه مخفی هست train میکنیم

برای این از patternnet استافده کردیم که خروجی شبکه عضبی ده تا است به نوعی ده کلاس برای خروجی شبکه عصبی داریم و سپس بخاطر این که تارگت و هدف خود را تبدیل به برداری با ده خروجی کنیم باید از متد ind2vec استفاده کنیم اما این تابع مقادیر صفر را نمیپذیرد پس ابتدا یک ارایه دیگر گرفته و تمامی مقادیرر trainY را یکی اضافه کرده تا دیگر صفری نباشد و به ان ارایه اد میکنیم و شبکه را train میکنیم سپس testX را به سبکه train شده میدهیم و بردار b max را به کمک متد های اماده پیدا میکنیم و چون قبلا به همه یکی اضافه کرده بودیم از تمامی مقادیر حاضر در b max یکی کم میکنیم و سپس همانند بخش یک پارت اول counter را حساب کرده و به کمک ان دقت شبکه عصبی را بدست میاوریم

تیکه کد مربوط به توضیحات بالا:

t=trainY+1;

t=full(ind2vec(t));

x=trainX;

net = train(net,x,t);

y=net(testX);

[a,b]=max(y);

b=b-1;

counter=0;

for i=1:10000

if b(1,i)==testY(1,i)

counter=counter+1;

end

end

جدول عملکرد

|  |  |  |
| --- | --- | --- |
| درصد | counter | patternnet |
| 41.63 | 4163 | Patternnet(2) |
| 79.68 | 7968 | Patternnet(3) |
| 88.41 | 8841 | Patternnet(5) |
| 89.98 | 8998 | Patternnet(6) |
| 87.46 | 8746 | Patternnet(7) |

پس Patternnet(6) بهترین عملکرد را دارد

برای بخش دوم نیز تابع های train متفاوتی برای patternnet(2) تست شده است که دقت درستی انها در جدول قابل مشاهده اند

|  |  |
| --- | --- |
| درصد دقت | تا بع train |
| 41.663% | train |
| 12.18% | traingd |
| 13.35% | traincgb |
| 23.02% | traingdx |
| 49.96% | traincgp |