ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО

Илустрација на еден тип невронски мрежи

Пример за една елементарна невронска мрежа

- Околина:
  - еден семафор резултатите ги покажува со помош на цифри претставени со мали сијалички во хоризонтала и вертикала
  - сензорот кој ги чита цифрите е во состојба да открие дали светат сијаличките, но е ограничен на најмногу 5 кои се непрекинати.
  - бројот на хоризонтални сијалички е 5, а на верикални 9.
- Задача: Да се обучи една невронска мрежа за да може да ги препознае
  - сите цифри од резулататот кога семафорот е исправен
  - цифрите од резултатот ако некои сијалички се неисправни

Процесни единици

- 10 различни објекти треба да се претставаат бинарно, значи дека треба да се најдат барем 4 признака кои генерираат 10 различни бинарни броеви.
- Најпросто, оваа задача може да се реши и со 7 процесни единици, при што секоја процесна единица би била една од седумте линии од кои е составена цифрата.
- Но, може да се најдат и комбинациите од 5 сијалички кои се појавуваат на еден или најмногу два правци (ги има барем 18) и потоа да се изврши одделување на четирите најважни, со помош на генетските алгоритми.

Нашиот рачен избор на процесните единици

- Влезни единици:
  - агол горе лево (5 го имаат, 5 го немаат)
  - агол средина горе десно (5 го имаат, 5 го немаат)
  - агол горе десно (7 го имаат, 3 го немаат)
  - црта долу лево (4 ја имаат, 6 ја немаат)
- Излезни единици:
  - Бројот претставен со помош на три цифри кои имаат вредности 0, 1 или 2.

Влезно-излезни парови

- 0: 1 0 1 1 1 0 0
- 1: 0 0 0 0 0 0 0
- 2: 0 1 1 1 0 0 1
- 3: 0 1 1 0 0 0 2
- 4: 0 1 0 0 0 1 0
- 5: 1 0 0 0 0 1 1
- 6: 1 0 0 1 0 1 2
- 7: 0 0 1 0 0 2 0
- 8: 1 1 1 1 0 2 1
- 9: 1 1 1 0 0 2 2

Зошто се измешани претставувањата на броевите со помош на бројниот систем со основа 3?

- Затоа што цифрата 1 е претставена со 4 нули, а тие со било која трансформација би преминале само во нули.
- За да се минимизира грешката (што е целта на обучувањето), цифрата 1 има влезно-излезна асоцијација составена само од нули.
- Според тоа, бројот кој е излез нумерички ќе се претстави како цифра на единиците при премин на следбеникот на бројот од системот со основа 3 во декадниот систем.

Што е целта?

- Ако матрицата на влезот (10 редици, 4 колони) се помножи со тежинската матрица (3 редици, 4 колони), треба да се добие што е можно помала разлика меѓу излезот (тоа е производот на двете матрици) и очекуваната цел (10 редици, 3 колони)
- На ваков начин се дефинира систем од 10 равенки со 12 непознати, кој има:
  - бесконечно многу решенија (што не е еднозначно)
  - противречен (што не ни треба)

Или, со матрици:

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \\ w_{41} & w_{42} & w_{43} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 2 & 0 \\ 0 & 2 & 1 \\ 0 & 2 & 2 \end{bmatrix}$$

Прва итерација: еднакви тежини

- Нека е матрицата W = ½ E_{4*4}.
- Причината за изборот на ½ е што најголемата целна вредност е 2, а таа е збир на четири единици (нив ги има во влезот на цифрата 8).

Излезна матрица

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} ,5 & ,5 & ,5 \\ ,5 & ,5 & ,5 \\ ,5 & ,5 & ,5 \\ ,5 & ,5 & ,5 \end{bmatrix} = \begin{bmatrix} 1,5 & 1,5 & 1,5 \\ 0 & 0 & 0 \\ 1,5 & 1,5 & 1,5 \\ 1,0 & 1,0 & 1,0 \\ ,5 & ,5 & ,5 \\ ,5 & ,5 & ,5 \\ 1 & 1 & 1 \\ ,5 & ,5 & ,5 \\ 2 & 2 & 2 \\ 1,5 & 1,5 & 1,5 \end{bmatrix}$$

Разлика меѓу излезот и целта

$$Output = \begin{bmatrix} 1,5 & 1,5 & 1,5 \\ 0 & 0 & 0 \\ 1,5 & 1,5 & 1,5 \\ 1,0 & 1,0 & 1,0 \\ ,5 & ,5 & ,5 \\ ,5 & ,5 & ,5 \\ 1 & 1 & 1 \\ ,5 & ,5 & ,5 \\ 2 & 2 & 2 \\ 1,5 & 1,5 & 1,5 \end{bmatrix} \qquad Target = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 2 & 0 \\ 0 & 2 & 1 \\ 0 & 2 & 2 \end{bmatrix}$$

Критериум за грешка

- Средно-квадратно отстапување меѓу излезот и целта

- Колона 1: 1,15
- Колона 2: 0,85
- Колона 3: 0,65

$$\begin{bmatrix} 0,25 & 2,25 & 2,25 \\ 0 & 0 & 0 \\ 2,25 & 2,25 & 0,25 \\ 1 & 1 & 1 \\ 0,25 & 0,25 & 0,25 \\ 0,25 & 0,25 & 0,25 \\ 1 & 0 & 1 \\ 0,25 & 2,25 & 0,25 \\ 4 & 0 & 1 \\ 2,25 & 0,25 & 0,25 \end{bmatrix}$$

Рачно нагодување на првата колона според Хебовиот алгоритам*

- Најголема грешка има кај првата колона; нејзината вредност е 1,15
- Нагодувањето се врши за сите три колони истовремено, но во илустрацијата се задржуваме само на првата, бидејќи пристапот е сосема идентичен.
- Доколку сакате, работете рачно за сите три колони, но тоа полесно ќе ви го реши MatLab.

Како да се нагоди првата колона од тежинската матрица?*

- Како прво, се дефинира вредноста за која ќе се врши промената. Таа се вика рата за учење (learning rate). Да земеме дека таа изнесува ¼.
- Потоа, користејќи го Хебовиот пристап, се одредува кој од четирите тежински фактори треда да се зголеми, а кој да се намали.
- Проценуваме дека цифрите 2, 9, а посебно 8 имаат голема грешка. Значи, тежинските фактори што се однесуваат на нивните влезни низи треба да се намалат. Во случајов, тоа се вториот и третиот тежински фактор. Првиот и четвртиот нека останат непроменети.

Новиот излез и очекуваната цел*

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} ,5 \\ ,25 \\ ,25 \\ ,5 \end{bmatrix} = \begin{bmatrix} 1,25 \\ 0 \\ 1 \\ ,5 \\ ,25 \\ ,5 \\ 1 \\ ,25 \\ 1,5 \\ 1 \end{bmatrix} \qquad Target = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Следна епоха (или, итерација)*

- Доколку како излезна функција се земе Хевизајдовата функција од излезот, тогаш по првата итерација, проблематични остануваат грешките на цифрите 2, 6, 8 и 9.
- Во нивната грешка, удел имаат трите десни тежини. Повторно се намалуваат за истата рата, т.е. ¼. За да не се намали точната вредност на излезот за цифрата 0, за истата рата се зголемува првиот тежински фактор.
- Ако Хевизајдовата функција има праг 1, тогаш сите излезни вредности под 1 се 0.

Излезот и целта по втората епоха*

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} ,75 \\ 0 \\ 0 \\ ,25 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ ,25 \\ 0 \\ 0 \\ ,75 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \qquad Target = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Пред почетокот на третата епоха*

- Излезот за цифрите 6, 8 и 9 е неприфатливо голем, па затоа одново се намалуваат трите десни тежини и повторно се зголемува првата.
- Ратата останува ¼.

Излезот и целта по третата епоха*

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} 1 \\ -,25 \\ -,25 \\ 0 \end{bmatrix} = \begin{bmatrix} ,75 \\ 0 \\ -,5 \\ -.5 \\ -.25 \\ 1 \\ 1 \\ 0 \\ ,5 \\ ,5 \end{bmatrix} \qquad Target = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Влез во следна епоха*

- Првиот тежински фактор е неоправдано фаворизиран, посебно затоа што тој придонесува за високата грешка на цифрите 5 и 6, но третата и четвртата тежина може да се зголемат.

Излезот и целта по четвртата епоха*

$$
\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} ,75 \\ -,25 \\ 0 \\ ,25 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -.25 \\ -.25 \\ ,75 \\ 1 \\ 0 \\ ,75 \\ ,5 \end{bmatrix} \qquad Target = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
$$

Следна епоха*

- Повторно се намалува само првата тежина (за да ја намали вредноста на цифрата 6), а се зголемува уделот на третата, за да се зајакне цифрата 0.

Излезот и целта по петтата епоха*

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} * \begin{bmatrix} ,5 \\ -,25 \\ ,25 \\ ,25 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ ,25 \\ 0 \\ -,25 \\ ,5 \\ ,75 \\ ,25 \\ ,75 \\ ,5 \end{bmatrix} \qquad Target = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Постапката е завршена

- Со оглед на тоа што излезната функција е Хевизајдова, се доаѓа до идеално обучување со множеството за обука.
- Ваков резултат не може да се добие, но грешката треба да се намали што е можно повеќе.
- Притоа, ратата за обучување може да има динамичка вредност, т.е. да се менува во тек на следните епохи.
- Алгоритмите за учење се далеку покомплексни, но оваа крајно упростена стратегија е извонредна за да се сфати како работат НМ.

Валидација

- Во нашиов пример, нема простор за валидација, бидејќи множеството за обука е многу мало.
- Сепак, една 10fold вкрстена исправност (научи 9 цифри и провери дали се препознава десеттата), има смисол.

Тестирање

- По завршувањето на обучувањето на сите 12 тежински фактори, претстои тестирање.
- Тестирањето во овој пример би се однесувало на проверка на тоа колку е способна обучената мрежа да ги препознае цифрите од семафорот во кои некои од сијаличките се неисправни.
- Проверете колкава е моќта на нашата обучена мрежа.

Прашања?