



به نام خدا



دانشگاه تهران  
دانشکده مهندسی برق و کامپیوتر  
شبکه های عصبی و یادگیری عمیق  
تمرین سری .....

نام و نام خانوادگی	فاطمه حقیقی
شماره دانشجویی	810195385
تاریخ ارسال گزارش	۲۹ خرداد ۱۳۹۹

## فهرست گزارش سوالات

3	سوال 1 – SOM
6	سوال ۲ – MaxNet
7	سوال 3 – Mexican Hat
8	سوال 4 – Hamming Net

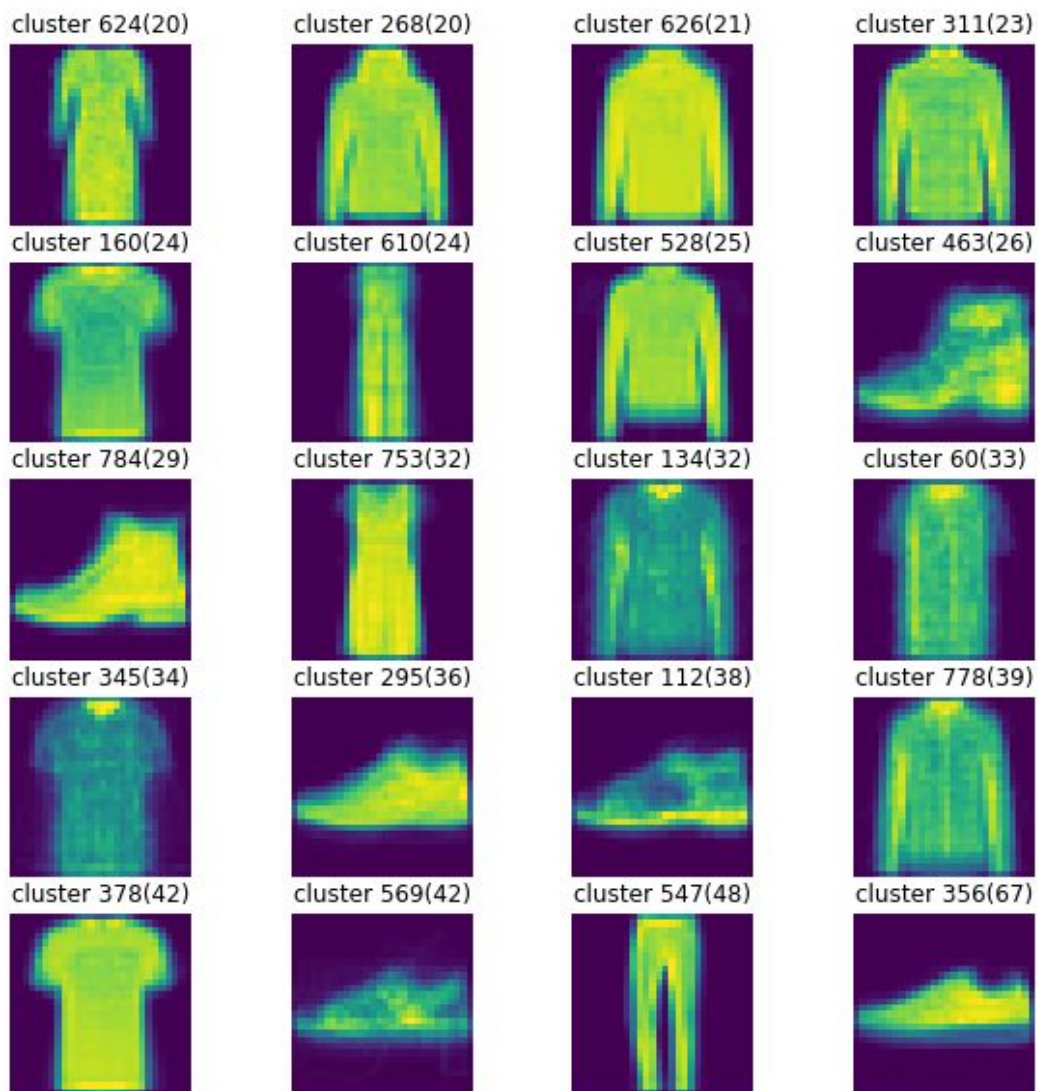
## SOM- 1 سوال

برای این سوال شبکه ی SOM را به گونه ای طراحی کردیم که ۱۰۰۰ داده ی اولیه ی دیتاست fashion mnist را خوشه بندی می کند.

قسمت الف)

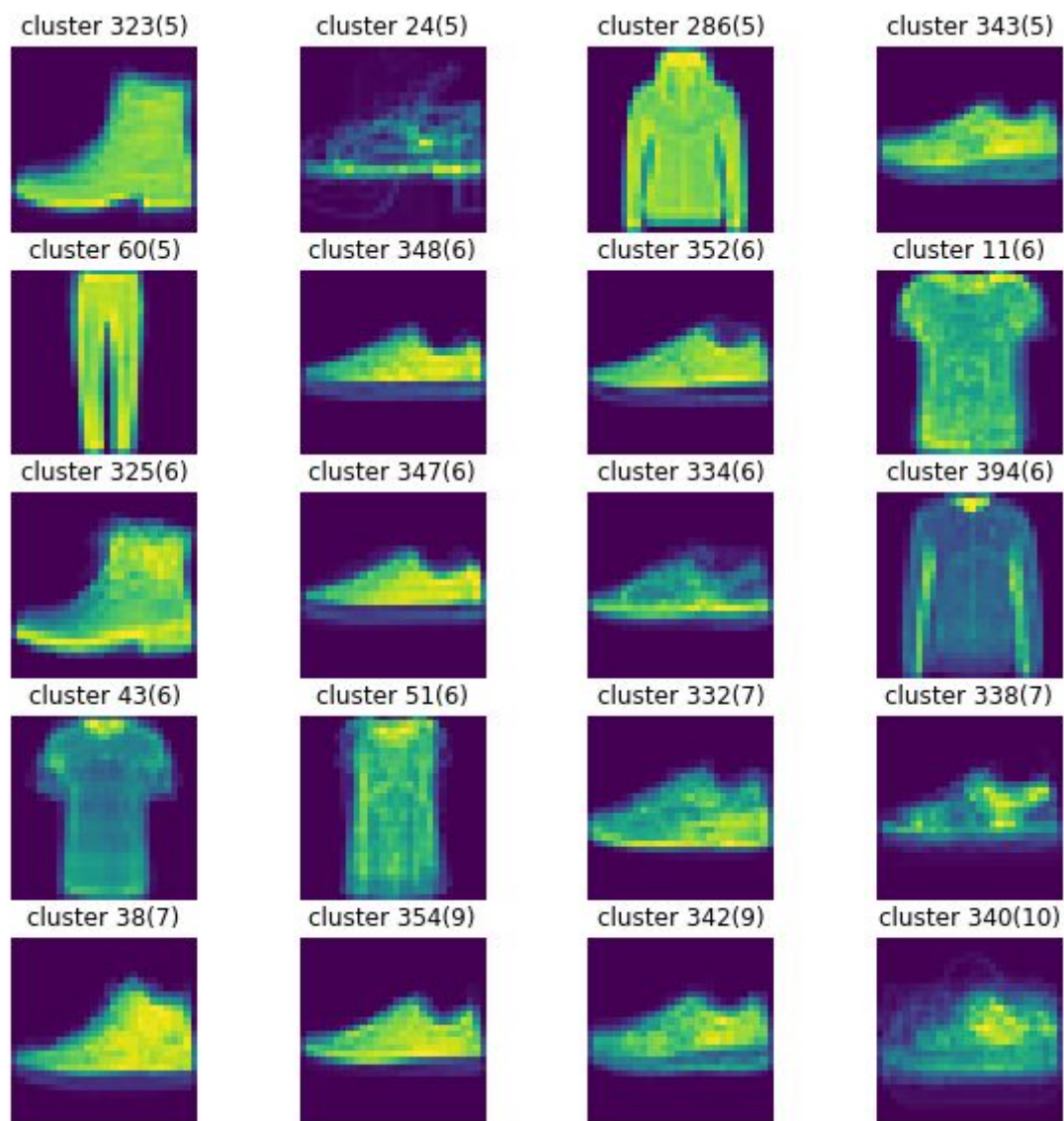
در صورتی که شعاع مجاورن هر نورون را صفر در نظر بگیریم:

در این حالت نورون ها همواره در حال رقابت هستند و در هیچ بازه ای با یکدیگر همکاری نمی کنند. پس از آموزش شبکه، مرکز دسته ی 20 cluster که داده ی بیشتری در آن است به صورت زیر می باشد:



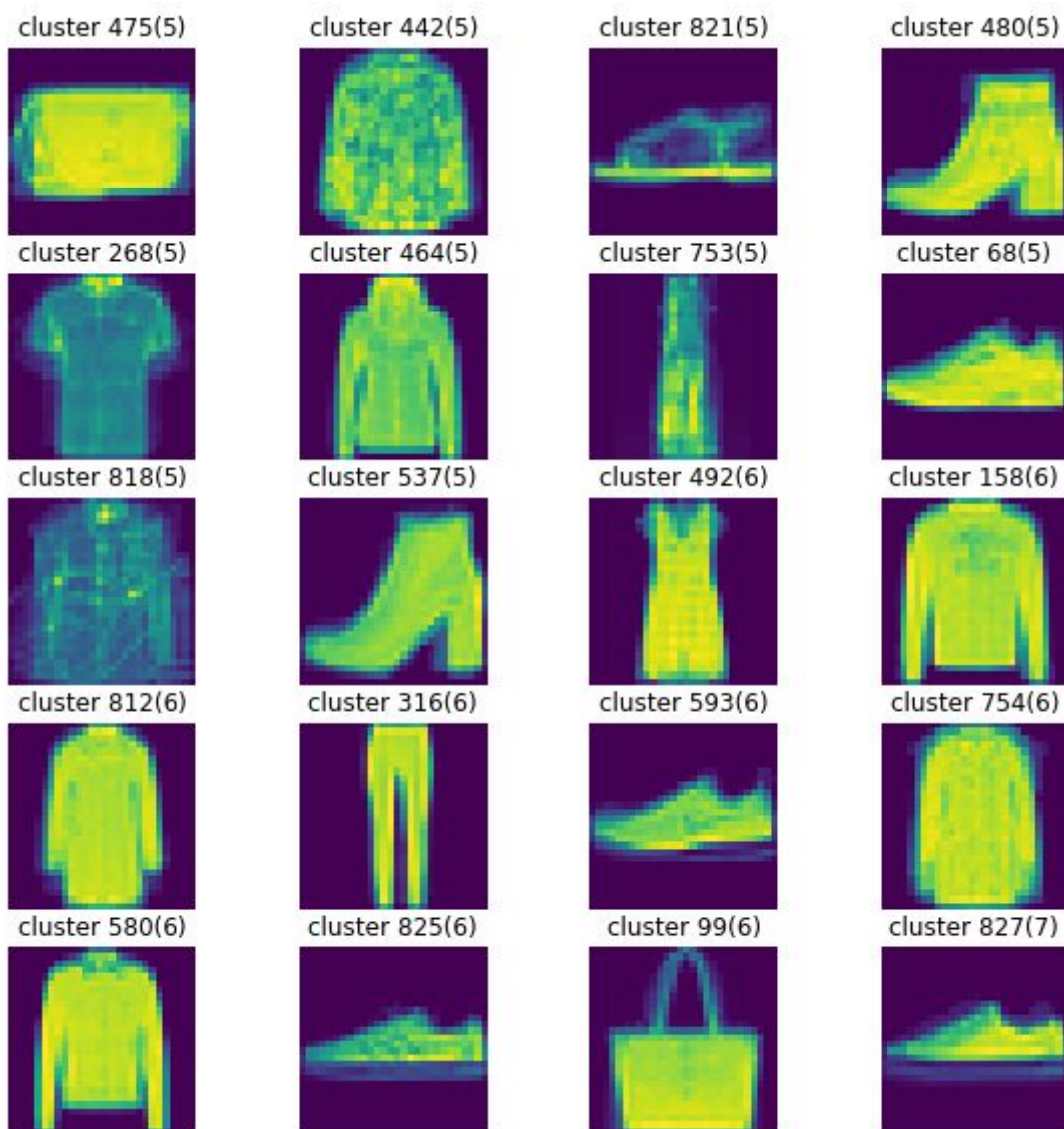
شکل ۱: مراکز دسته ی ۲۰ cluster با داده های بیشتر

قسمت ب) در صورتی که شعاع مجاور هر نورون را یک و آرایش میان نورون ها را خطی در نظر بگیریم، پس از آموزش شبکه، مرکز دسته ی 20 cluster که داده ی بیشتری در آن است به صورت زیر می باشد:



شکل ۲: مراکز دسته ی ۲۰ cluster با داده های بیشتر

قسمت ج) در صورتی که شعاع مجاور هر نورون را یک و آرایش میان نورون ها را مربعی در نظر بگیریم، پس از آموزش شبکه، مرکز دسته ی 20 cluster که داده ی بیشتری در آن است به صورت زیر می باشد:



شکل ۳: مراکز دسته ی ۲۰ cluster با داده های بیشتر

## سوال ۲ – MaxNet

پس از طراحی و اجرای شبکه، مراحل به روزرسانی المنت های داده ی ورودی به صورت زیر می باشد:

in iteration = 1

[1.15 0.95 0.9 0.1 1.1 1.2]

in iteration = 2

[0.3775 0.1475 0.09 0.205 0.32 0.435]

in iteration = 3

[0.197875 0.0- 0.0- 0.0- 0.13175 0.264 ]

in iteration = 4

[ 0.1385125 0.0- 0.0- 0.0- 0.06246875 0.21455625 ]

in iteration = 5

[0.09695875 0.0- 0.0- 0.0- 0.00950844 0.18440906 ]

in iteration = 6

[0.06787112 0.0- 0.0- 0.0- 0.0- 0.16843898 ]

in iteration = 7

[0.04260528 0.0- 0.0- 0.0- 0.0- 0.15825832 ]

in iteration = 8

[0.01886653 0.0- 0.0- 0.0- 0.0- 0.15186752 ]

in iteration = 9

[ 0.0- 0.0- 0.0- 0.0- 0.0- 0.14903754 ]

پاسخ سوال اول:

در صورتی که همه ی اعداد از  $\beta$  که خود عددی حقیقی است، بزرگتر باشند و بخواهیم بزرگترین آن ها را به وسیله ی شبکه ی MaxNet بیابیم، کافی ایست اختلاف این اعداد را با عدد حقیقی  $\beta$  بدست بیاوریم و اعداد حاصل را به شبکه بدهیم. به عبارت دیگر اگر  $a$  یکی از اعداد موجود در این مجموعه باشد،  $b = a - \beta$  را به عنوان مقدار جدید به شبکه MaxNet می دهیم و این عمل را بر روی بقیه ی اعداد نیز انجام می دهیم

پاسخ سوال دوم:

برای آنکه بتوانیم بوسیله ی شبکه ی MaxNet مجموعه ای از اعداد را از بزرگ به کوچک مرتب کنیم، می توان از این شبکه به صورت گردشی استفاده کنیم به این ترتیب در هر بار اجرای آن از میان اعداد باقی مانده بزرگترین عدد پیدا می شود و اعداد را می توان از بزرگ به کوچک مرتب کرد.

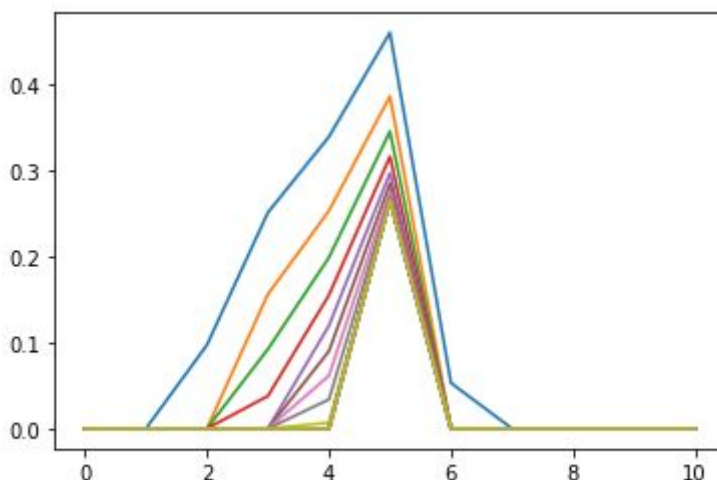
پاسخ سوال سوم:

برای اینکه اعداد را به وسیله ی شبکه ی MaxNet از کوچک به بزرگ مرتب کنیم ابتدا اعداد را منفی می کنیم. سپس به وسیله ی روش گفته شده در پاسخ سوال اول، اعداد منفی شده را تغییر می دهیم. به عبارت دیگر به وسیله ی این روش اعداد منفی را به مقادیر مثبت تبدیل می کنیم. حال به وسیله ی شبکه ی MaxNet مجموعه اعداد حاصل را از بزرگ به کوچک به صورتی گردشی (روش گفته شده در پاسخ سوال دوم) مرتب می کنیم. اعداد مرتب شده از بزرگ به کوچک، مرتب شده کوچک به بزرگ اعداد اولیه می باشد. برای تبدیل اعداد به فرم اولیه ی آن ها مقدار مثبت را از هر کدام کم کرد (اعداد را منفی می کنیم) و سپس هر کدام را منفی می کنیم. حال می بینیم که اعداد اولیه از کوچک به بزرگ مرتب شده اند

### سوال 3 – Mexican Hat

قسمت الف)  $R1 = 0$  و  $R2 = \infty$  :

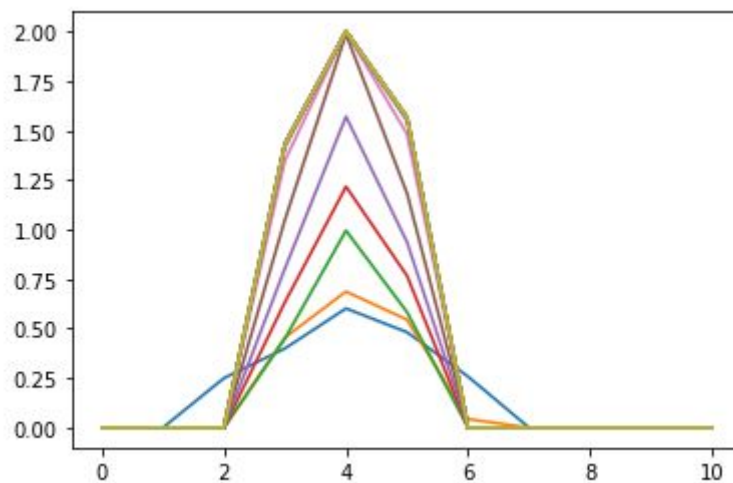
برای این قسمت تعداد iteration ها را برابر با ۲۰،  $c1 = 1$  و  $c2 = -0.1$  در نظر گرفتیم.



شکل ۴: نمودار index

قسمت ب)  $R1 = 1$  و  $R2 = 3$  :

برای این قسمت تعداد iteration ها را برابر با ۲۰،  $c1 = 0.6$  و  $c2 = -0.4$  در نظر گرفتیم.



شکل ۵: نمودار index

#### سوال 4 – Hamming Net

قسمت الف)

شبکه ی Hamming net طراحی شده در این بخش یک شبکه با دو لایه می باشد. در لایه ی اول این شبکه به اندازه ی بعد بردار ورودی نورون داریم. در اینجا بردارها به طول ۶ می باشند پس ۶ نورون در لایه ی اول داریم. در لایه ی دوم تعداد نورون ها به تعداد بردارهای پایه می باشد. میان هر نورون در لایه ی دوم و هر نورون از لایه ی اول یک وزن وجود دارد. بنابراین می توان گفت در این شبکه یک ماتریس وزن  $N \times M$  داریم که  $N$  اندازه ی بعد بردار ورودی و  $M$  برابر با تعداد بردارهای پایه می باشد مقدار وزن های متناظر با هر نورون در لایه ی دوم نصف مقدار بردار پایه ی متناظر با آن نورون است. همچنین به ازای هر نورون در لایه ی دوم یک مقدار بایاس نیز وجود دارد که اندازه ی آن نصف اندازه ی بعد ورودی می باشد.

قسمت ب)

پس از اعمال بردارهای پایه و بردارهای ورودی داده شده در صورت پروژه به شبکه، نتیجه به صورت زیر بدست آمد:



```

original vector is: [ 1 -1 1 1 -1 1]
output vector is: [3. 3. 4.] vector belong to vector number 3
original vector is: [-1 1 1 -1 1 -1]
output vector is: [4. 2. 3.] vector belong to vector number 1
original vector is: [ 1 1 1 -1 -1 -1]
output vector is: [4. 2. 3.] vector belong to vector number 1
original vector is: [-1 -1 -1 1 1 1]
output vector is: [2. 4. 3.] vector belong to vector number 2
original vector is: [1 1 1 1 1 1]
output vector is: [3. 3. 6.] vector belong to vector number 3
original vector is: [-1 -1 1 -1 -1 -1]
output vector is: [4. 2. 1.] vector belong to vector number 1
original vector is: [-1 -1 -1 1 -1 -1]
output vector is: [2. 4. 1.] vector belong to vector number 2
original vector is: [ 1 1 -1 -1 1 1]
output vector is: [3. 3. 4.] vector belong to vector number 3
original vector is: [ 1 1 -1 1 1 1]
output vector is: [2. 4. 5.] vector belong to vector number 3
original vector is: [ 1 1 1 -1 1 1]
output vector is: [4. 2. 5.] vector belong to vector number 3

```

---