DÜZCE ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

Derin Öğrenme 1.Ödevi

Grup Üyeleri

- **161001061 Yusuf Sina BAKAN**
- **161001043 Furkan SEVGİLİ**
- **161002212 Mehmed Akif AY**

Mart 2020

Veri Setinin Oluşturulması

Resimlerin okunması

```
fid = fopen('train-images', 'r');

A = fread(fid, 1, 'uint32');
magicNumber = swapbytes(uint32(A));

A = fread(fid, 1, 'uint32');
totalImages = swapbytes(uint32(A));

A = fread(fid, 1, 'uint32');
numRows = swapbytes(uint32(A));

A = fread(fid, 1, 'uint32');
numCols = swapbytes(uint32(A));

trainImageCellArray = cell(1, totalImages);
for k = 1 : totalImages
    A = fread(fid, numRows*numCols, 'uint8');
    trainImageCellArray{k} = reshape(uint8(A),
numCols, numRows)';
end
fclose(fid);
trainImageArray=cell2mat(trainImageCellArray);
```

Hedeflerin okunması

```
fid = fopen('train-labels', 'r');
    trainId=[];

    A = fread(fid, 1, 'uint32');
    magicNum = swapbytes(uint32(A));


    A = fread(fid, 1, 'uint32');
    totalId = swapbytes(uint32(A));


for t=1:totalId
       A = fread(fid, 1, 'uint8');
        trainId = [trainId swapbytes(uint8(A))];
end


fclose(fid);
```

Bize verilen veri setleri big endian ve idx biçimindedir.

En önemli bit sağda olacak şekilde kodlanmıştır. Bu yüzden okurken ‘swapbytes’ fonksiyonu ile tersine çevrildi.

‘fread’ fonksiyonu her çağrıldığında bir sonraki hücreyi okur.

Resim depolayan dosyalarda, baştan ikinci hücre dosyadaki resim sayısını üçüncü ve dördüncü hücreler resimlerin boyutlarını tutar. geri kalan tüm hücrelerin her biri bir resimi tutar.

Hedef verilerini depolayan dosyalarda boyutlara ihtiyaç olmadığı için sadece ikinci hücrede hedef sayısı tutmuştur. Geri kalan tüm hücreler hedeflere ayrılmıştır.

Resimlerden özelliklerin çıkarılması

```
[boy,~]=size(trainImageArray);
Features=[];
i=0;
for i=1:28:boy
    A=trainImageArray(i:i+27,:);
    A=A(:);
    B=dct(single(A));
    Features=[Features B];
end
```

Resimler üzerinde Discrete Cosine Transform yapılır.

Elde edilen 784 özellik TrainFeatures içinde tutulur.

Hedeflerin DNN ağının anlayacağı şekle getirilmesi

```
TrainTarget=[];
[boy,en]=size(trainId);
for i=1:boy
   switch trainId(i,en)
      case 0
         TrainTarget=[TrainTarget; 1 0 0 0 0 0 0 0 0 0];%0
      case 1
         TrainTarget=[TrainTarget; 0 1 0 0 0 0 0 0 0 0];%1
      case 2
         TrainTarget=[TrainTarget; 0 0 1 0 0 0 0 0 0 0];%2
      case 3
         TrainTarget=[TrainTarget; 0 0 0 1 0 0 0 0 0 0];%3
      case 4
         TrainTarget=[TrainTarget; 0 0 0 0 1 0 0 0 0 0];%4
      case 5
         TrainTarget=[TrainTarget; 0 0 0 0 0 1 0 0 0 0];%5
      case 6
         TrainTarget=[TrainTarget; 0 0 0 0 0 0 1 0 0 0];%6
      case 7
         TrainTarget=[TrainTarget; 0 0 0 0 0 0 0 1 0 0];%7
      case 8
         TrainTarget=[TrainTarget; 0 0 0 0 0 0 0 0 1 0];%8
      case 9
         TrainTarget=[TrainTarget; 0 0 0 0 0 0 0 0 0 1];%9
   end
end
```

Eğitimin hızlandırılması için eğitimde kullanılacak olan özelliklerin seçilmesi

```
F=[];
for i=1:70000
F=[F;Features(i,1:7) Features(i,29:35)
Features(i,57:63) Features(i,85:91)
Features(i,113:119) Features(i,141:147)
Features(i,169:175)];
end
```

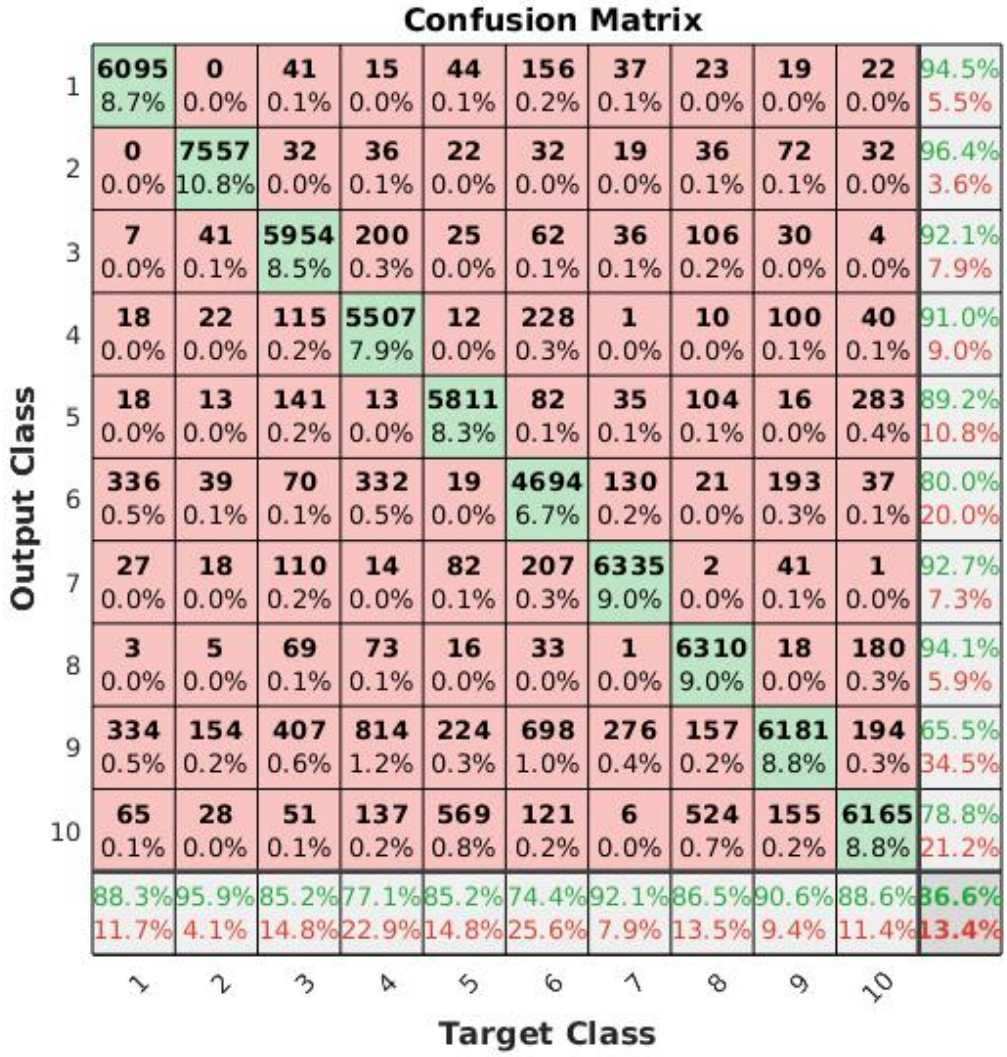
Discrete Cosine Transform sonucunda resimi oluşturan özelliklerin en önemlisi sol üstte en önemsizi sol altta olacak şekilde 28x28 dizi içinde tutulur.

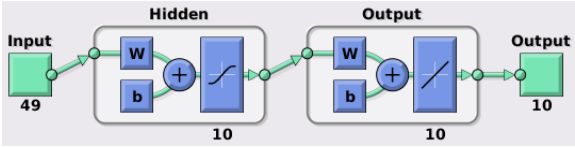
Eğitimde kullanılmak için bu dizinin sol üstünden başlayarak 7x7 lik kısım alınmıştır.

Sonuç olarak eğitimde resim başına 49 özellik kullanılmıştır.

Eğitim ve sonuçlar

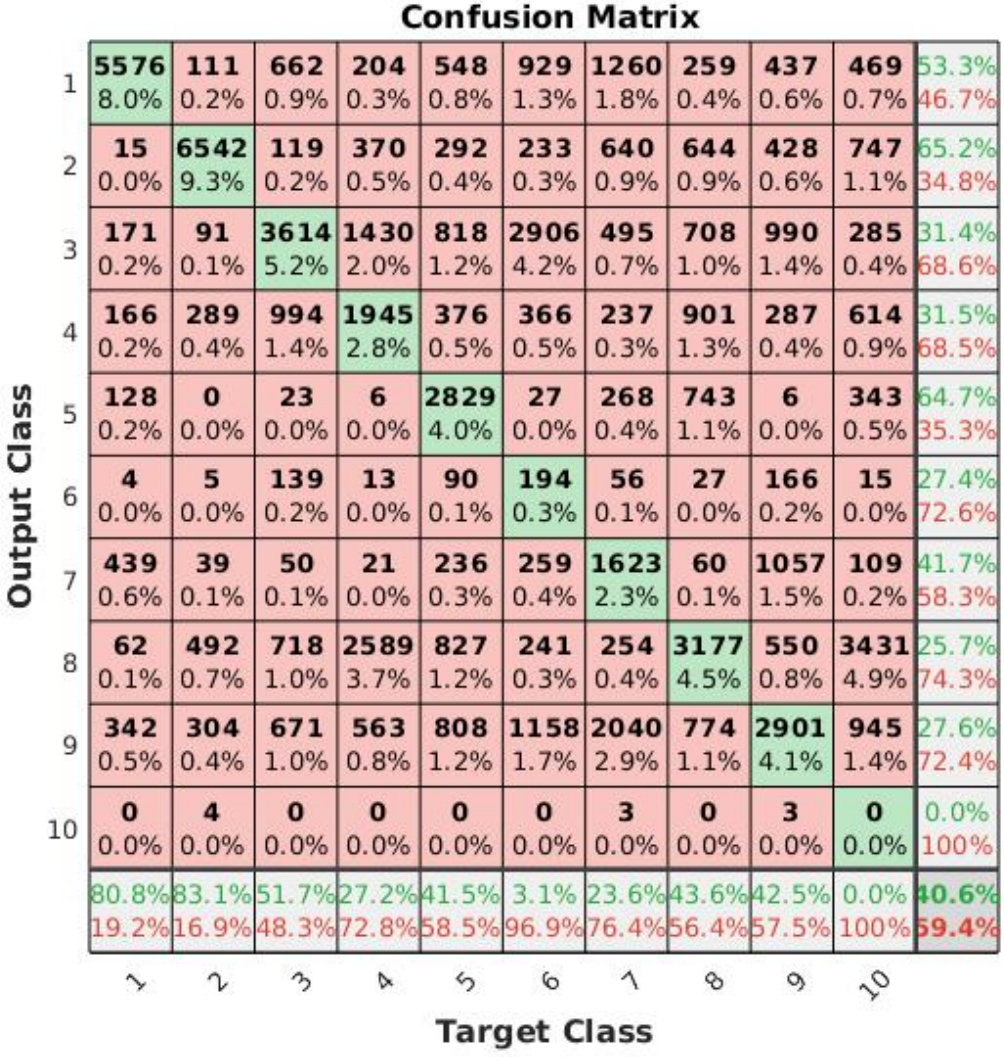
Matlab da diziler 1 den başladığı için aşağıdaki confusion matrix lerde 1, sıfır, 2 , bir vb. anlamına gelir.

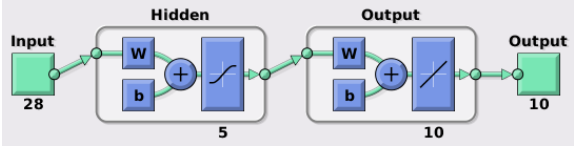





```
net = feedforwardnet();
net=train(net,P,T);
```

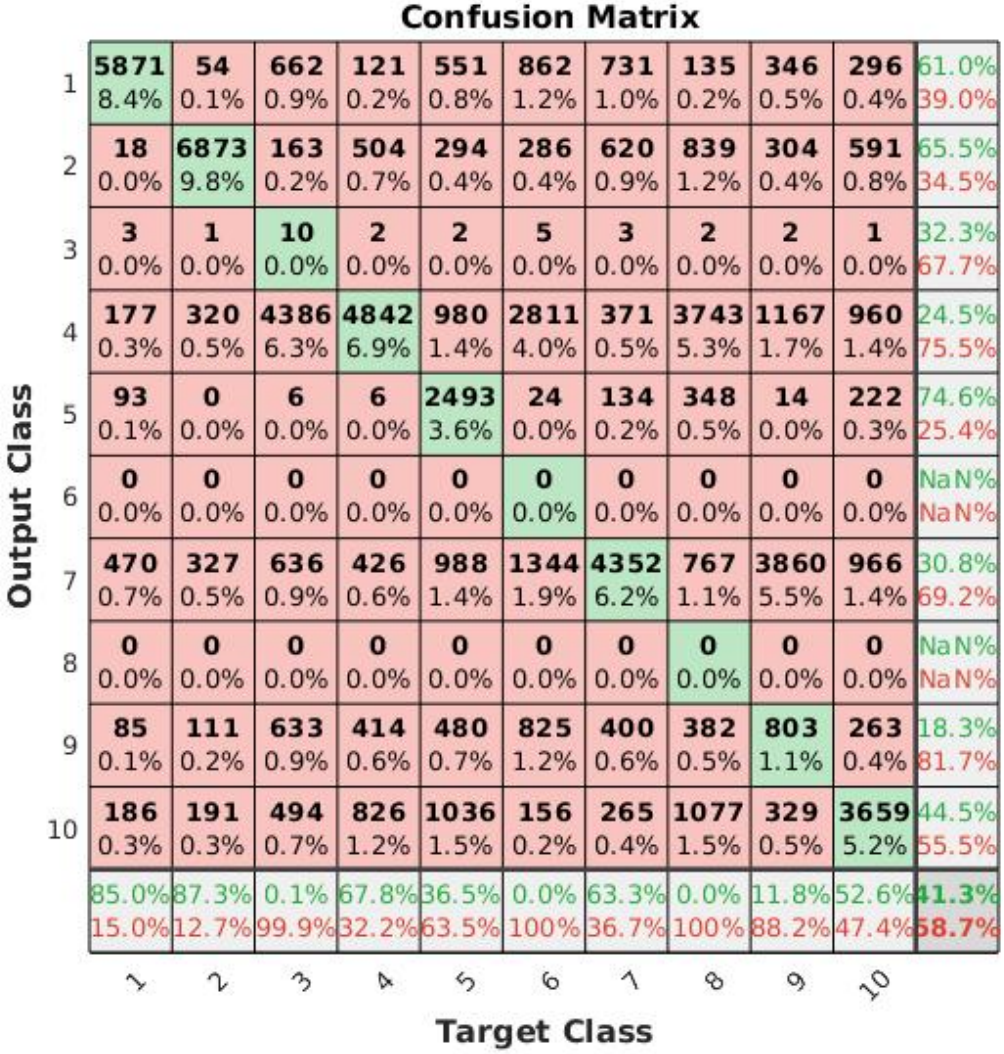
Test 1

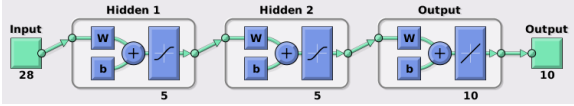





```
net = feedforwardnet(5);
net=train(net,P,T);
```

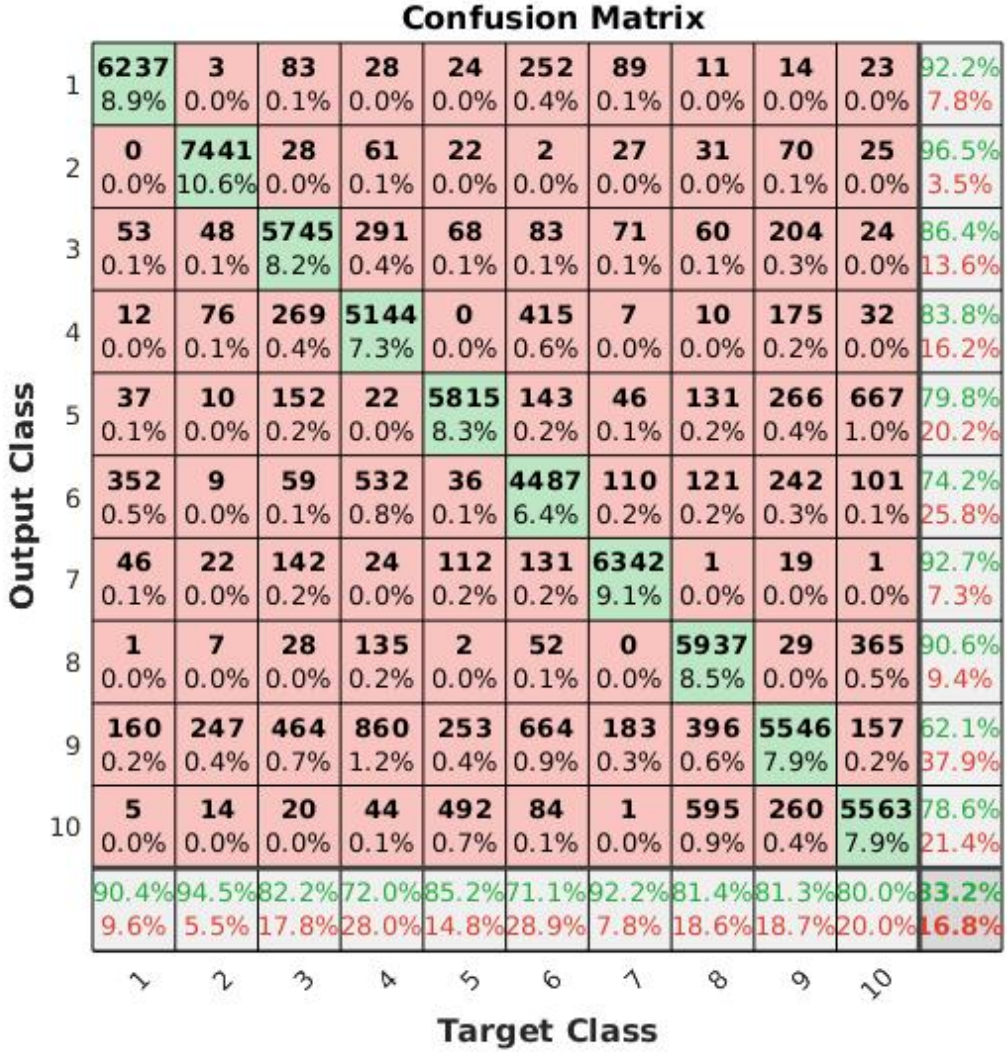
Test 2

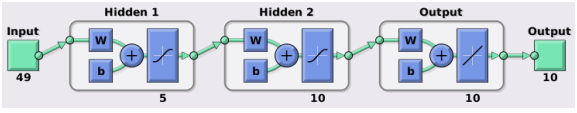





```
net = feedforwardnet([5 5]);
net=train(net,P,T);
```

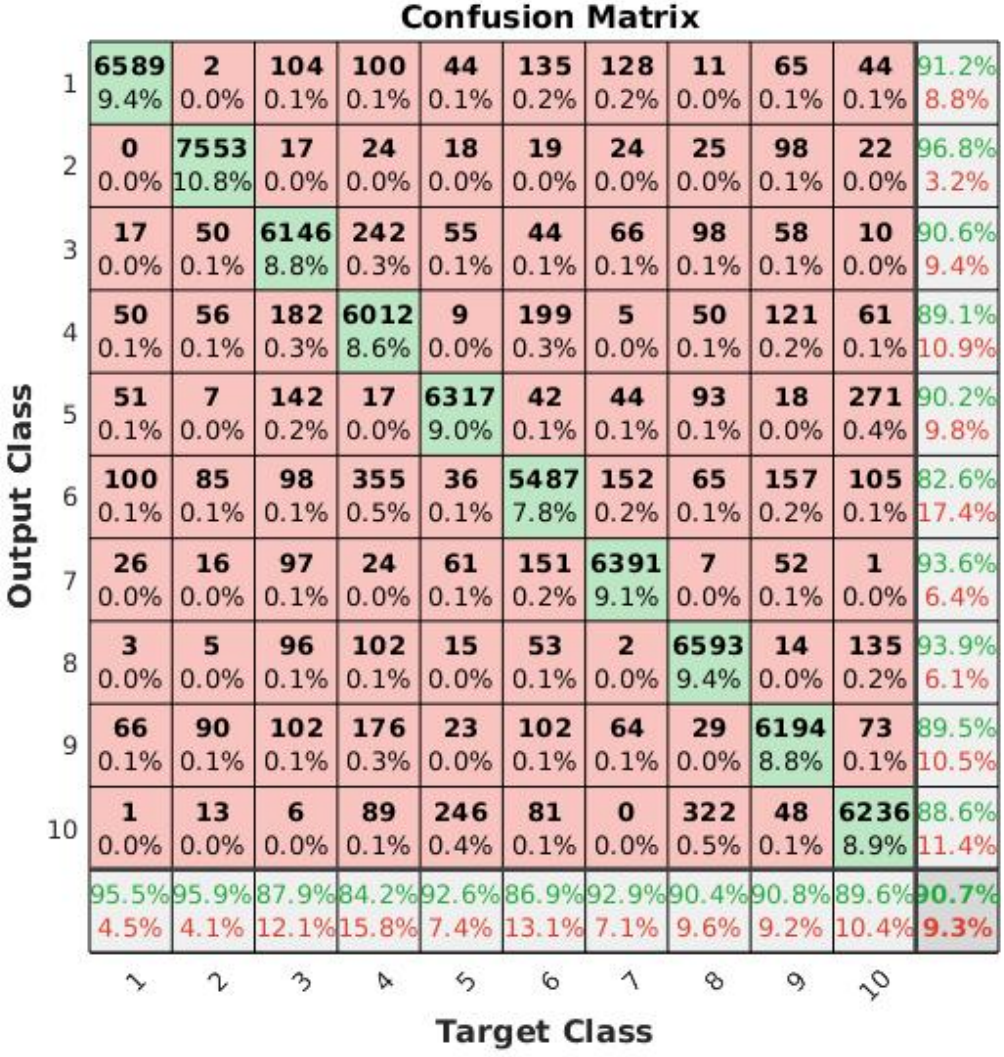
Test 3

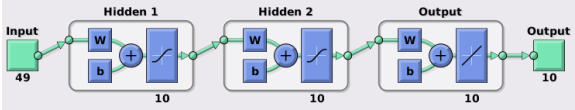





```
net = feedforwardnet([5 10]);
net=train(net,P,T);
```

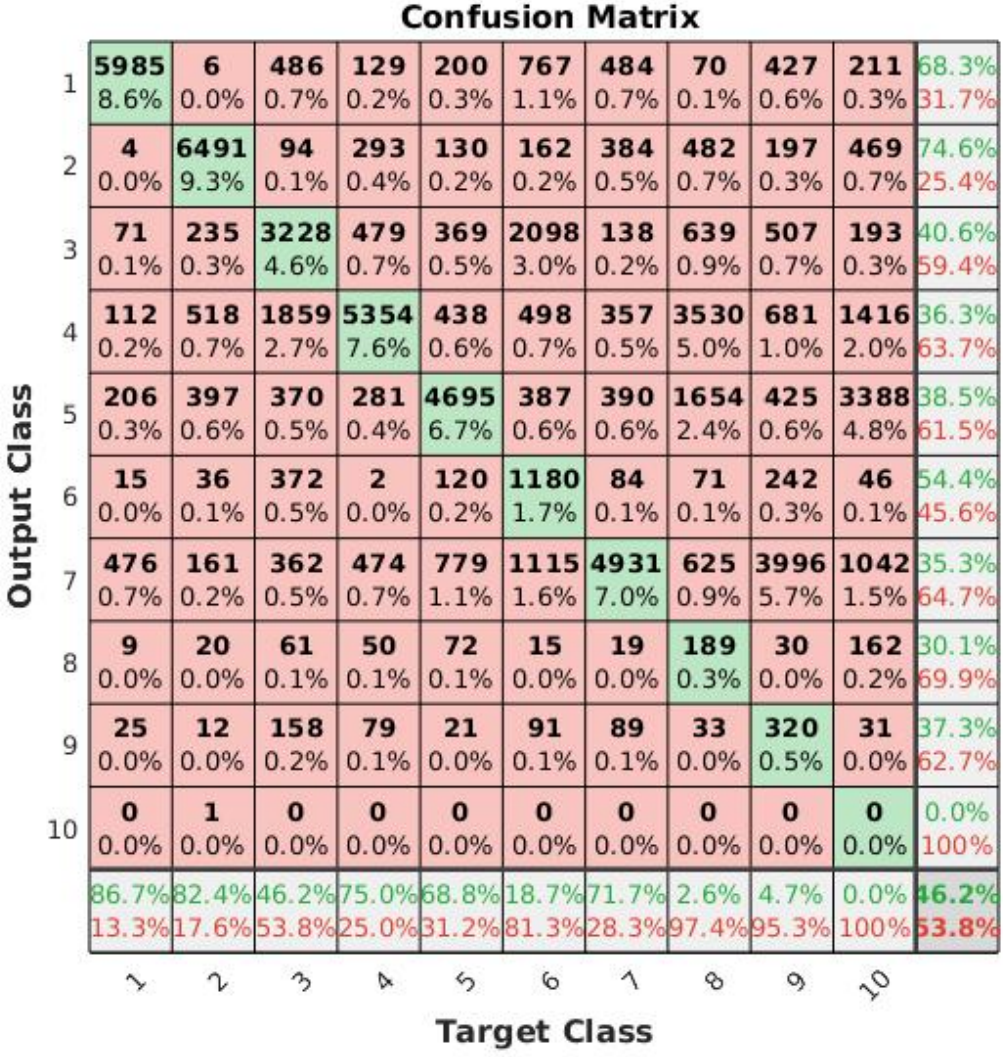
Test 4

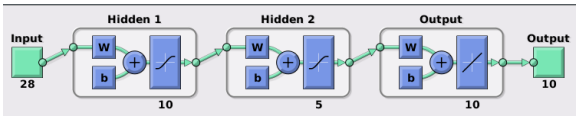





```
net = feedforwardnet([10 10]);
net=train(net,P,T);
```

Test 5

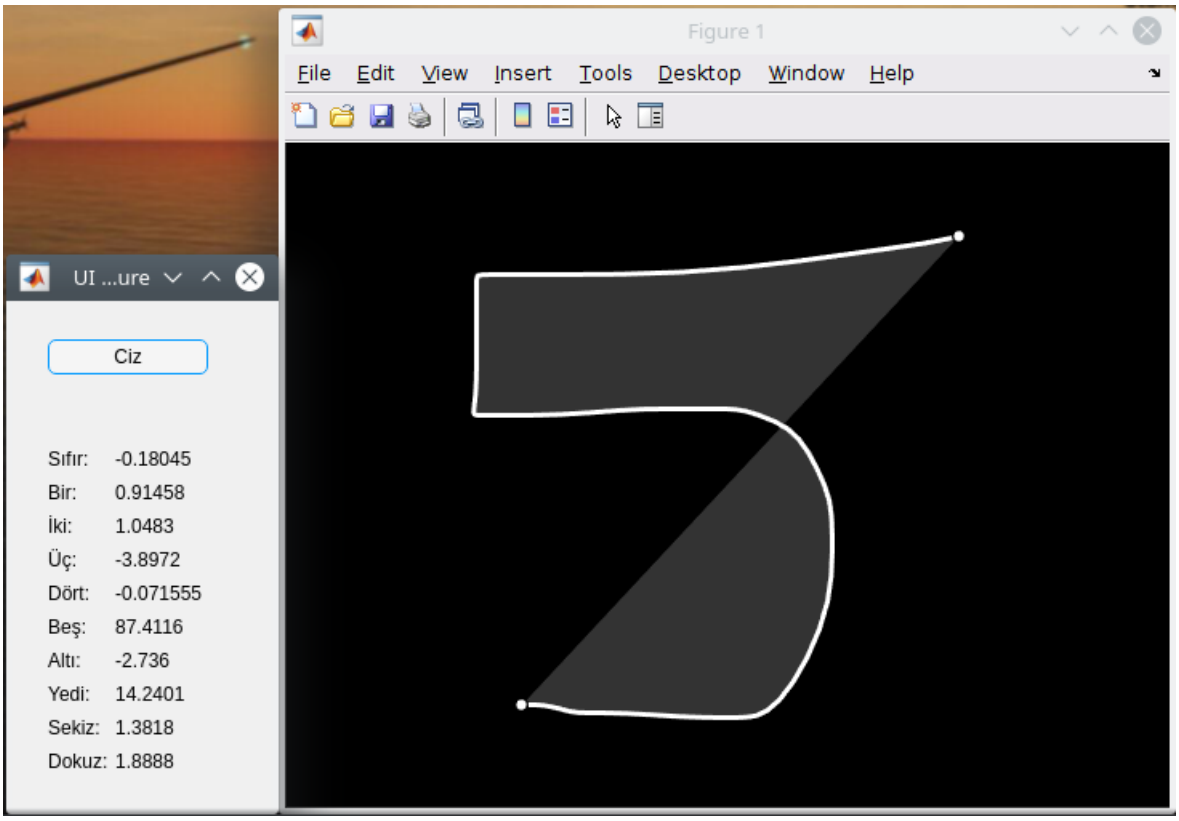





```
net = feedforwardnet([10 5]);
net=train(net,P,T);
```

Test 6

Arayüz




```
ax=axes('Color','Black','Position',[0, 0, 280, 280]);

drawfreehand(ax,'Color','White','Closed',false,'Waypoints',false,'DrawingArea','auto');
saveas(ax,'image','jpeg');
        im=imread('image.jpg');

im=not(im2bw(im));
se = strel('line',11,90);
im=imdilate(im,se);
im=imresize(im,[28 28],'nearest');
imd=im(:);

imdct=dct(single(imd));
imdct=reshape(imdct,[28 28]);
imFeat=imdct(1:7,1:7);
imFeat=imFeat(:);

a=sim(net,imFeat);
```

Yüksek çözünürlükte çizilmiş olan resim küçültüldüğünde özelliklerini kaybetmemesi için. ‘imdilate’ ile genişletilir.

Üzerinde DCT uygulanır ve sol üst kısmından 7x7 dizi alınır.

Daha önceden eğitilmiş olan ağ verilir ve alınan sonuç ekrana yazdırılır.