**NÖRAL AĞ İLE LOJİK VE ARİTMETİK**

**DEVRELERİN GERÇEKLENMESİ**

**ÖZET**

Matlab ortamında ToolBox kullanmadan geri beslemeli Yapay Sinir Ağı(Nöral Ağ) tasarlanarak 2-3 girişli XOR kapısı, 4 girişli AND kapısı ve Tam Toplayıcı Devre öğretildi.

**Anahtar Kelimeler:** Geri Beslemeli Nöral Ağ; Matlab; Lojik Kapılar; Tam Toplayıcı

**ABSTRACT**

Designed FeedForward Neural Network on Matlab without ToolBox. Learned 2-3 input XOR gate, 4 input AND gate and Full Adder Circuit.

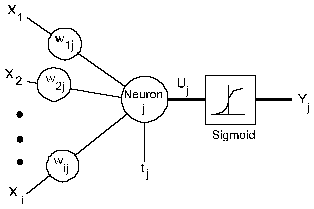
**Keywords:** FeedForward Neural Network; Matlab; Lojic Gates; Full Adder

**1.GİRİŞ**

Yapay Sinir Ağları, beynin fizyolojisinden yararlanılarak oluşturulan bilgi işleme modelleridir. Literatürde 100’den fazla YSA modeli vardır. Bazı bilim adamları, beynimizin güçlü düşünme, hatırlama ve problem çözme yeteneklerini bilgisayara aktarmaya çalışmışlardır. Bazı araştırmacılar ise, beynin fonksiyonlarını kısmen yerine getiren birçok modeli oluşturmaya çalışmışlardır.

Yapay sinir ağlarının hesaplama ve bilgi işleme gücünü, paralel dağılmış yapısından, öğrenebilme ve genelleme yeteneğinden aldığı söylenebilir. Genelleme, eğitim ya da öğrenme sürecinde karşılaşılmayan girişler için YSA’ların uygun tepkileri üretmesi olarak tanımlanır. Bu üstün özellikleri, YSA’ların karmaşık problemleri çözebilme yeteneğini gösterir.

Yapay sinir hücreleri, YSA’nın çalışmasına esas teşkil eden en küçük bilgi işleme birimidir. Geliştirilen hücre modellerinde bazı farklılıklar olmasıyla birlikte genel özellikleri ile çok girişli bir nöron modeli Şekil-1’de gösterilmiştir. Bu hücre modelinde girişler, ağırlıklar, toplam fonksiyonu, aktivasyon fonksiyonu ve çıkış görülmektedir. Girişler (**X**) diğer hücrelerden ya da dış ortamlardan hücreye giren bilgilerdir. Bilgiler, bağlantılar üzerindeki ağırlıklar (**w**) üzerinden hücreye girer ve ağırlıklar ilgili girişin hücre üzerindeki etkisini belirler. Toplam fonksiyonu (**U**), bir hücreye gelen net girdiyi hesaplayan bir fonksiyondur ve genellikle net girdi, girişlerin ilgili ağırlıkla çarpımlarının toplamıdır. Hücre modellerinde net girdiyi artıran +1 değerli ya da azaltan -1 değerli eşik girişi (**t**) bulunabilir. Aktivasyon fonksiyonu (**Sigmoid**) ise toplama fonksiyonundan elde edilen net girdiyi bir işlemden geçirerek hücre çıktısını (**Y**) belirleyen bir fonksiyondur. Nöron çıkışı seçilen özel aktivasyon fonksiyonuna bağlıdır. Böylece nöron giriş/çıkış ilişkilerinin bazı özel amaçlarda kullanılabilmesi için w ve b parametreleri öğrenme kuralları tarafından ayarlanabilir.

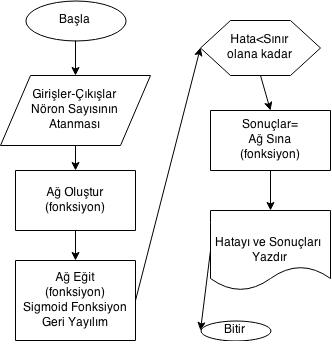


Şekil 1. YSA Yapısı

**2. Materyal ve Yöntem**

Tamamen elle kodlanmış Nöral Ağ tasalandı. Giriş, çıkışlar ve değişken sayıda nöron için dizilerin kullanılması gerekliydi. Dinamik matrisleri ve hazır matematik fonksiyonları olduğu için Matlab üzerinde tasarım yapıldı.

Program iskeleti fonksiyonlar üzerine kuruldu. Dışarıdan alınan parametrelerle ağ oluşturan ve belirli hata sınırına kadar eğiten fonksiyonlar sayesinde anlaşılması ve kullanılması kolay bir program olmasına çalışıldı.



Şekil 2. Akış Diyagramı

2.1. Ana Program

Nöron sayısı, girişler ve çıkış tanımlandıktan sonra nöral ağ için yazılmış fonksiyonları çağıran ana program

|  |
| --- |
| adet=[2 2 1]; % katmanlardaki noron sayisi(giris, gizli, cikis)  girisler=[0 0;0 1;1 0;1 1];  cikislar=[0;1;1;0];  hata\_payi=1e-1; % 0.1  [W ,B]= ag\_olustur(adet); % agirliklarin sayisini belirliyor ve rasgele degerler atiyor  [W ,B, hata]= ag\_egit(girisler,cikislar,W,B,adet,hata\_payi); % agirliklari ayarlayip geri donduruyor  sonuc=zeros(size(cikislar));  for i=1:size(sonuc,1)  sonuc(i,:)=ag\_sina(girisler(i,:),W,B,adet,'yazdir'); % verilen girisleri ve agirliklari kullanarak sonuc uretiyor  end |

Kaynak Kod 1. Ana Program(XOR)

2.2. Ağ Sınama Fonksiyonu

Parametre olarak sınanacak girişleri ve ağırlıkları alarak ağın ürettiği sonucu döndüren ve konsola yazdıran fonksiyon

2.3. Ağ Oluşturma Fonksiyonu

Parametre olarak gelen nöron sayı dizisine göre ağırlıkların adetini belirleyen ve rastgele ağırlık değerleri üreten fonksiyon. Geri dönüş olarak ağırlıkları döndürüyor.

|  |
| --- |
| function [W,B]= ag\_olustur(adet) % W,B = fonksiyonun geri dönüş değerleri (ağırlıklar)  B=rand(sum(adet),1); % rand=0-1 aralığında rasgele sayı, sum=toplam  toplam=0;  for i=1:(size(adet,2)-1)  toplam=toplam+(adet(i)\*adet(i+1));  end  W=rand(toplam,1);  End |

Kaynak Kod 2. ag\_olustur fonksiyonu

2.4. Ağ Eğitme Fonksiyonu

Parametre olarak girişleri, çıkışları, ağırlıkları, nöron sayısını ve hata eşiğini alarak ağı eğiten(ağırlık güncellemesi), performans ve ağ sonuçlarını ekrana yazdıran fonksiyon. Geri dönüş olarak güncellenmiş ağırlıklar ve hata(iteratif) döndürür.

|  |
| --- |
| Nöronların değeri=Sigmoid(()+T)  (T=1 sıfırdan kurtar)  Hata(çıkış nöronları için)=  Hata(ara nöronları için)=  Ağırlık\_değişimi(i)= |

Formül 1. ag\_egit fonksiyonu matematiksel formülleme(Kaynak.1,2)

2.5. Hata Hesaplama Fonksiyonu

Parametre olarak nöron çıkış değerlerini ve beklenen değerleri alark hatayı döndüren fonksiyon.

|  |
| --- |
| function donus=hata\_hesapla(x,beklenen) % x=nöron çıkış değerleri  i=size(beklenen,1)\*size(beklenen,2);  j=size(x,1)\*size(x,2);  hata=zeros(i,1);  while i>0  hata(i)=abs(x(j)-beklenen(i)); % abs=mutlak değer  i=i-1;  j=j-1;  end  donus=max(hata(:));  end |

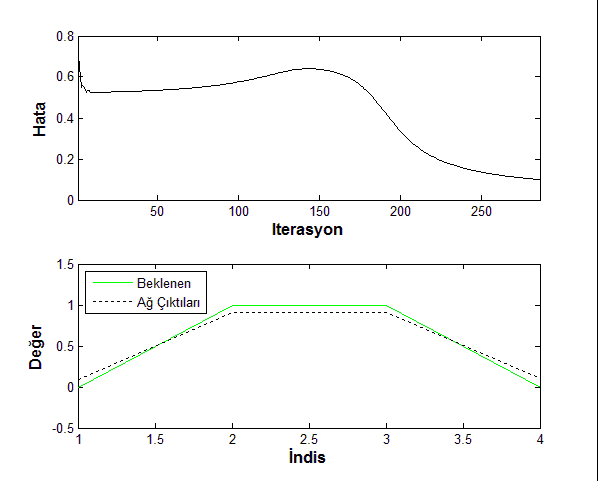
Kaynak Kod 3. hata\_hesapla fonksiyonu

**3. PERFORMANS ANALİZİ**

Ağın eğitimi sırasında hata eşiği olarak belirlenmiştir.

3.1. XOR Kapısı ( 2 Giriş )

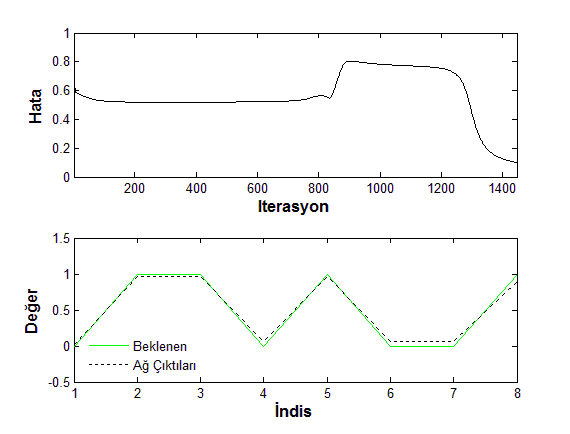
Nöron sayısı 2 giriş, 2 gizli, 1 çıkış olarak belirlenmiştir.



Şekil 3. XOR öğrenmesi

3.2. XOR Kapısı ( 3 Giriş )

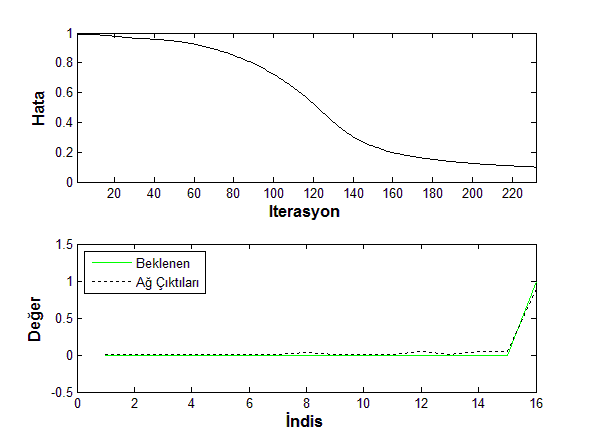
Nöron sayısı 3 giriş, 6 gizli, 1 çıkış olarak belirlenmiştir.



Şekil 4. XOR öğrenmesi(3 giriş)

3.3. AND Kapısı ( 4 Giriş )

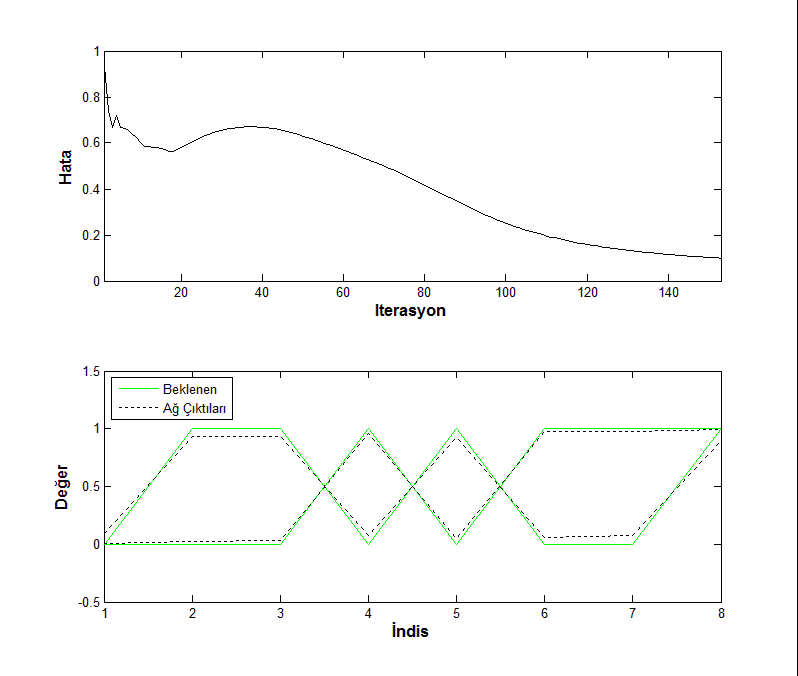
Nöron sayısı 4 giriş, 4 gizli, 1 çıkış olarak belirlenmiştir.



Şekil 5. AND öğrenmesi

3.4. Tam Toplayıcı Devre

Nöron sayısı 3 giriş, 6 gizli, 2 çıkış olarak belirlenmiştir.



Şekil 6. Tam Toplayıcı Öğrenmesi

**4. SONUÇLAR**

Ağın eğitimi sırasında hata eşiği olarak belirlenmiştir.

4.1. XOR Kapısı ( 2 Giriş )

Nöron sayısı 2 giriş, 2 gizli, 1 çıkış olarak belirlenmiştir.

|  |  |  |  |
| --- | --- | --- | --- |
| **Giriş 1** | **Giriş 2** | **Beklenen** | **Ağ Sonucu** |
| 0 | 0 | 0 | 0.09070 |
| 0 | 1 | 1 | 0.91164 |
| 1 | 0 | 1 | 0.91153 |
| 1 | 1 | 0 | 0.09864 |

Tablo 1. XOR

4.2. XOR Kapısı ( 3 Giriş )

Nöron sayısı 3 giriş, 6 gizli, 1 çıkış olarak belirlenmiştir.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Giriş 1** | **Giriş 2** | **Giriş 3** | **Beklenen** | **Ağ Sonucu** |
| 0 | 0 | 0 | 0 | 0.01055 |
| 0 | 0 | 1 | 1 | 0.97623 |
| 0 | 1 | 0 | 1 | 0.95823 |
| 0 | 1 | 1 | 0 | 0.05438 |
| 1 | 0 | 0 | 1 | 0.96559 |
| 1 | 0 | 1 | 0 | 0.06580 |
| 1 | 1 | 0 | 0 | 0.05536 |
| 1 | 1 | 1 | 1 | 0.90142 |

Tablo 2. XOR(3 giriş)

4.3. AND Kapısı ( 4 Giriş )

Nöron sayısı 4 giriş, 4 gizli, 1 çıkış olarak belirlenmiştir.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Giriş 1** | **Giriş 2** | **Giriş 3** | **Giriş 4** | **Beklenen** | **Ağ Sonucu** |
| 0 | 0 | 0 | 0 | 0 | 2.1215e-05 |
| 0 | 0 | 0 | 1 | 0 | 5.2556e-05 |
| 0 | 0 | 1 | 0 | 0 | 9.0569e-05 |
| 0 | 0 | 1 | 1 | 0 | 0.00068002 |
| 0 | 1 | 0 | 0 | 0 | 3.9772e-05 |
| 0 | 1 | 0 | 1 | 0 | 0.00030306 |
| 0 | 1 | 1 | 0 | 0 | 0.0050271 |
| 0 | 1 | 1 | 1 | 0 | 0.03932 |
| 1 | 0 | 0 | 0 | 0 | 0.00010211 |
| 1 | 0 | 0 | 1 | 0 | 0.00064831 |
| 1 | 0 | 1 | 0 | 0 | 0.0011089 |
| 1 | 0 | 1 | 1 | 0 | 0.044423 |
| 1 | 1 | 0 | 0 | 0 | 0.00048145 |
| 1 | 1 | 0 | 1 | 0 | 0.042887 |
| 1 | 1 | 1 | 0 | 0 | 0.046713 |
| 1 | 1 | 1 | 1 | 1 | 0.9033 |

Tablo 3. AND

4.4. Tam Toplayıcı Devre

Nöron sayısı 3 giriş, 6 gizli, 2 çıkış olarak belirlenmiştir.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Giriş 1** | **Giriş 2** | **Giriş Elde** | **Beklenen Toplam** | **Beklenen**  **Elde** | **Ağ Toplam** | **Ağ Elde** |
| 0 | 0 | 0 | 0 | 0 | 0.025317 | 0.0039203 |
| 0 | 0 | 1 | 1 | 0 | 0.95547 | 0.023871 |
| 0 | 1 | 0 | 1 | 0 | 0.95944 | 0.030244 |
| 0 | 1 | 1 | 0 | 1 | 0.05047 | 0.95648 |
| 1 | 0 | 0 | 1 | 0 | 0.96482 | 0.045593 |
| 1 | 0 | 1 | 0 | 1 | 0.056755 | 0.97267 |
| 1 | 1 | 0 | 0 | 1 | 0.076726 | 0.97823 |
| 1 | 1 | 1 | 1 | 1 | 0.9013 | 0.9958 |

Tablo 4. Tam Toplayıcı

**5.KAYNAKLAR**

1. Matlab Documentation, <http://www.mathworks.com/help/nnet/ref/logsig.html>
2. Prof. Dr. Ercan Öztemel, Yapay Sinir Ağları, 3. Basım, 2012