

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**YAPAY SİNİR AĞLARI KULLANILARAK CİLT
HASTALIKLARININ TEŞHİS EDİLMESİ İÇİN YARDIMCI BİR
SİSTEM GERÇEKLEŞTİRİLMESİ**

YÜKSEK LİSANS TEZİ

Musa KAPLAN

(092129103)

Anabilim Dalı: Bilgisayar Mühendisliği
Programı: Yazılım
Danışman: Yrd. Doç. Dr. Ahmet ÇINAR

EYLÜL-2016

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YAPAY SİNİR AĞLARI KULLANILARAK CİLT HASTALIKLARININ TEŞHİS
EDİLMESİ İÇİN YARDIMCI BİR SİSTEM GERÇEKLEŞTİRİLMESİ

YÜKSEK LİSANS TEZİ

Musa KAPLAN

(092129103)

Tezin Enstitüye Verildiği Tarih : 14 KASIM 2016

Tezin Savunulduğu Tarih : 11 KASIM 2016

Tez Danışmanı

Yr. Doç. Dr. Ahmet ÇINAR(F.Ü)

Diğer Jüri Üyeleri

:Prof.Dr.Ali KARCI (F.Ü)

Yr. Doç.Dr.Ebubekir ERDEM (F.Ü)

KASIM-2016

ÖNSÖZ

Bu tez çalışması süresi boyunca benden yardımlarını ve bilimsel katkılarını esirgemeyen değerli hocam Yrd.Doç.Dr. Ahmet ÇINAR'a, teşekkür ederim.

Musa KAPLAN

ELAZIĞ-2016

İÇİNDEKİLER

	Sayfa No
ÖNSÖZ	II
İÇİNDEKİLER.....	III
ÖZET	V
SUMMARY	VI
ŞEKİLLER LİSTESİ	VII
ÇİZELGELER LİSTESİ	IX
SEMBOLLER ve KISALTMALAR LİSTESİ	X
1. GİRİŞ.....	1
1.1. Cilt Hastalıkları	1
1.1.1. Akne(sivilce)	1
1.1.2. Psoriasis (Sedef)	2
1.1.3. Tinea (Mantar).....	2
1.2. Tezin Amacı	2
1.3. Tezin İçeriği	2
2. ÖNCEKİ ÇALIŞMALAR	3
3. MATERYAL ve METOT	5
3.1. Öznitelik Çıkarımı.....	5
3.1.1. Gabor Dalgacık Dönüşümü (Gabor Wavelet Transform-GWT).....	5
3.2. Sınıflandırma Yöntemleri.....	10
3.2.1. Yapay Sinir Ağları.....	11
3.2.1.1. Biyolojik Sinir Ağlarının Yapısı	13
3.2.1.2. Yapay Sinir Hücresi ve Elemanları	14
3.2.1.3. Yapay Sinir Ağlarının Genel Özellikleri.....	20
3.2.1.4. Yapay Sinir Ağlarının Avantaj ve Dezavantajları.....	22
3.2.1.5. Yapay Sinir Ağlarının Kullanım Alanları	23
3.2.1.6. Yapay Sinir Ağlarının Eğitilmesi	23
3.2.2. Yapay Sinir Ağlarının Yapılarına Göre Sınıflandırılması.....	24
3.2.2.1. İleri Beslemeli Ağlar	25
3.2.2.2. Geri Beslemeli Ağlar.....	25
3.2.3. Yapay Sinir ağlarının Öğrenme Stratejileri.....	26
3.2.3.1. Eğitici Öğrenme	26

3.2.3.2. Eğiticişiz Öğrenme	27
3.2.3.3. Takviyeli Öğrenme	28
3.2.4. Çok Katmanlı Algılayıcı (ÇKA)	28
3.2.4.1. Geriye Yayılım (Back Propagation) Öğrenme Algoritması	32
3.2.5. Çapraz Doğrulama	33
4. UYGULAMA	35
4.1. Öğrenme Algoritmaları Açısından Sınıflandırma Sonuçları Analizi	41
4.2. Nöron Sayıları Açısından Sınıflandırma Sonuçları Analizi	43
4.3. İterasyon Sayıları Açısından Sınıflandırma Sonuçları Analizi	46
5. SONUÇ	48
6. EKLER	49
7. KAYNAKÇA	50
ÖZGEÇMİŞ	55

ÖZET

Cilt hastalıklarının teşhisi günümüzde dermatoloji alanında büyük bir sıkıntıya neden olmuştur. Cilt hastalıklarının benzerliğinden dolayı bu çoğu zaman yanlış teşhise sebep olmaktadır. Yanlış teşhis ise doğal olarak yanlış tedavi uygulanmasına neden olacaktır.

Bu çalışmada cilt hastalıkları tanınması üzerine bir sistem gerçekleştirilmeye çalışılmıştır. Bu çalışmada materyal olarak insan cilt görüntüleri kullanılmıştır. Bu cilt görüntülerinden öznitelik çıkarmak için Gabor dalgacık dönüşümü kullanılarak her bir imgeden öznitelik matrisi elde edilmiştir. Bu öznitelik matrislerin standart sapma, ortalama ve entropi değerleri hesaplanmıştır. Bu değerlerle her imge için 3 parametrelilik yeni bir öznitelik vektörü oluşturulmuştur. Elde edilen özniteliklerin sınıflandırılması için yapay sinir ağları kullanılmıştır. Elde edilen öz nitelikler farklı öğrenme algoritmaları ile çapraz doğrulama testine tabi tutularak %85'e varan başarı düzeyleri elde edilmiştir.

Anahtar Kelimeler: Yapay Sinir Ağları, Cilt Hastalıkları, Gabor Dalgacık Dönüşümü, Çok Katmanlı Algılayıcı, Çapraz Doğrulama

SUMMARY

DEVELOPING A SUBSYSTEM FOR DERMATOLOGICAL DISEASES DIAGNOSIS USING ARTIFICIAL NEURAL NETWORKS

Nowadays, huge distress in dermatology field is caused by diagnose of skin diseases. Due to similarities of skin diseases, wrong diagnoses are caused. Naturally this condition brings wrong treatment.

In this study, tried to construct a system about skin diseases recognition. In this working, human skin image are used as a material. To reach attributes from these skin images, attribute matrix from each image using Gabor wavelet lug. Standard deviation, mean and entropi values of attribute matrix. With these values, made a new attribute vector that has 3 parameters for each image. To classification of obtained attributes, used artificial neural networks. Obtained attributes are subjected cross validate test with different learning algorithms and finally reached approximately %85 success rate.

Key Words: Artificial Neural Network, Skin Diseases, Gabor Wavelet Lug, Multilayer Perceptron, Cross Validate

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 3.1. Tezde kullanılan örüntü tanıma sistemi	5
Şekil 3.2. Biyolojik sinir hücresi ve bileşenlerinin gösterimi	12
Şekil 3.3. Yapay sinir hücresi.....	12
Şekil 3.4. Sinir sisteminin blok gösterimi[20].....	13
Şekil 3.5. Bir biyolojik sinir hücresinin yapısı	14
Şekil 3.6. Basit bir yapay sinir hücresinin yapısı	15
Şekil 3.7. Eşik değer fonksiyonu.....	16
Şekil 3.8. Adım(Step) aktivasyon fonksiyonu	17
Şekil 3.9. Doğrusal Fonksiyon	17
Şekil 3.10. Kısmi doğrusal fonksiyon	18
Şekil 3.11. Lojistik sigmoid fonksiyonu	18
Şekil 3.12. Hiperbolik tanjant fonksiyonu	19
Şekil 3.13. İleri beslemeli ve çok katmanlı bir YSA gösterimi.....	25
Şekil 3.14. Geri beslemeli ağı gösterimi	26
Şekil 3.15. Eğitici Öğrenme Yapısı[14].....	27
Şekil 3.16. Eğitici Öğrenme blok diyagramı[14].....	27
Şekil 3.17. Takviyeli Öğrenme Blok Diyagramı[14].....	28
Şekil 3.18. Çok katmanlı ağ modeli	29
Şekil 3.19. Bir geri-yayılma ağ gösterimi	32
Şekil 3.20. Çok katmanlı bir algılayıcının geri yayılım akış şeması.....	33
Şekil 4.1. Akne cilt hastalığının gösterimi	35
Şekil 4.2. Sedef cilt hastalığının gösterimi.....	36
Şekil 4.3. Mantar cilt hastalığının gösterimi	36
Şekil 4.4. Her bandın bir birimlik standart sapmaya eşit olduğu normal dağılım eğrisi.....	40
Şekil 4.5. Başarım oranlarının histogramı.....	40
Şekil 4.6. YSA sınıflandırıcılarının öğrenme algoritmalarına göre başarımlarının kutu grafikleri	42
Şekil 4.7. Ara katmanında 25 nöron bulunan YSA sınıflandırıcıların başarımları histogramı	43
Şekil 4.8. Ara katmanında 50 nöron bulunan YSA sınıflandırıcıların başarımları histogramı	44

Şekil 4.9. YSA sınıflandırıcılarının nöron sayılarına göre başarımlarının kutu grafikleri..	45
Şekil 4.10. YSA sınıflandırıcılarının iterasyon sayılarına göre başarımlarının kutu grafikleri	47

ÇİZELGELER LİSTESİ

Sayfa No

Tablo 3.1. Biyolojik ve yapay sinir ağı sisteminin karşılaştırılması.....	14
Tablo 4.1. Çapraz doğrulama doğru sınıflandırma oranı gösterimi.....	39
Tablo 4.2. Doğru Sınıflandırma oranları tablosundan hesaplanan istatistikler.....	39
Tablo 4.3. Öğrenme algoritmalarının doğru sınıflandırma oranlarının istatistikler.....	41
Tablo 4.4. Nöron sayılarına göre YSA sınıflandırıcılarının başarımlar istatistikleri.....	43
Tablo 4.5. İterasyon sayılarına göre YSA sınıflandırıcılarının başarımlar istatistikleri	46

SEMBOLLER ve KISALTMALAR LİSTESİ

ÇKA	:Çok Katmanlı Algılayıcı
GWT	:Gabor Dalgacık Dönüşümü
YSA	:Yapay Sinir Ağları
g	:Gabor Çekirdeği
θ	:Desenin Açısı
λ	:Desenin Dalga Boyu
σ	:Standart Sapma
$G(x)$	:İmge Yoğunluğu
ϑ	:Pikseller Arasındaki Açısı
x_i	:ÇKA ağındaki nöron girişi
y_i	:ÇKA ağındaki nöron çıkışı
a	:Ölçekleme/Yayılma Kat Sayısını
b	:Dönüşüm/Öteleme Kat Sayısını
Ψ	:Ana Dalgacık
$w(a,b)$	:Sürekli Dalgacık Dönüşümü

1. GİRİŞ

Tezin bu bölümünde genel olarak tezin amacı, yapısı ve yapılacak işlemlere genel hatlarıyla değinilmiştir. Çalışmanın amacı cilt hastalıklarının teşhis işlemi olduğu için genel olarak cilt hastalıkları hakkında bilgiler verilmiştir.

Cilt hastalıkların teşhis etme işlemi dermatologların en büyük sıkıntılarından biridir. Cilt hastalıkların türünün çok olması teşhis etme işlemini zorlaştırmaktadır. Bu alanda yeteri kadar çalışma bulunmamakla beraber yavaş yavaş ilgi artmaya başlamaktadır. Çalışmalarda dermatologlar için yardımcı sistemler geliştirilmeye çalışılmaktadır.

Bu çalışmada amaç dermatoloğa (cildiyeceye) teşhis etme işleminde yardımcı bir sistem gerçekleştirmesini sağlamaktır.

1.1.Cilt Hastalıkları

Deri, vücudun tüm yüzeyini saran, vücudu dış etkenlerden koruyan, vücudu koruyup atık maddeleri vücuttan atan bir organdır. Deri kendi içinde üç tabakadan oluşmaktadır. Bunlar; epidermis, dermis ve hipodermistir. İnsanlar arasında en çok görülen enfeksiyon hastalıklarından biri de deri hastalığıdır. Ciltte görülen deri hastalıkları dermatoz olarak adlandırılır. Deri hastalıklarının birçok sebebi vardır. Vücuttaki yaralar, yanıklar, soğuklardan dolayı oluşan çatlaklar fiziksel olarak deri hastalıklarına örnek olarak verilebilir. Bunun dışında parazitlerden (bit, tahtakurusu vs.) dolayı oluşan cilt hastalıklarından bahsedilebilir. Bir diğer cilt hastalık türü alerjik cilt hastalıklarıdır. Deri hastalıklarının çeşitleri çok olduğu için sadece üzerinde çalışacağımız cilt hastalık çeşitleri hakkında genel bir bilgiler verildi. Üzerinde çalışacağımız cilt hastalık türlerinden üç tanesini inceleyeceğiz. Bu cilt hastalıkları;

Akne(sivilce), Psoriasis(sedef) ve Tinea(mantar) hastalıklarıdır.

1.1.1.Akne(sivilce)

Akne ter bezlerinin iltihaplanması sonucu oluşan bir hastalıktır. Bu hastalık tıkalı gözenekler, siyah noktacıklar şeklinde ve iltihaplı kabarıklıklar şeklinde görülen bir hastalık türüdür[51-53].

1.1.2.Psoriasis (Sedef)

Sedef hastalığında kırmızı ve üzerinde sedefimsi beyaz pullar bulunan yaralardan oluşan kalıcı bir hastalıktır. Bu hastalık genellikle kafa derisi, diz ve dirseklerde ortaya çıkar. Hafif olan vakalarda kişiler hastalığın farkına varmazlar. Bazen de vücudun geniş kısımlarını tutabiliyor. Ailesel özelliğine rağmen kişiden kişiye bulaşmıyor[51-53].

1.1.3.Tinea (Mantar)

Mantar vücuttaki deri enfeksiyonlarından biridir. Yaşayan organizmalar olduğu için her insanda çoğalma ihtimalleri vardır. Deri renginde değişiklik yapıp deride pullanmaya sebep olurlar. Mantarlar genelde vücutta yağ yoğunluğu fazla olan yerlerde yaşarlar. Açık renkli kişilerde soluk ve pembe renkli şekillerde olup koyu renkli kişilerde açık ve koyu renkte görünebilirler[51-53].

1.2.Tezin Amacı

Bu tezin amacı yapay sinir ağları kullanılarak cilt hastalığının teşhis edilmesi için yardımcı bir sistem sunmaktır. Cilt hastalıklarının teşhisinde oluşan sıkıntılara çözüm getirilmeye çalışıldı. Tez çalışması için uluslararası kabul görmüş dermatoloji görselleri kullanıldı. Yapılan çalışma ile üç tür cilt hastalığının teşhis işlemi gerçekleştirilecektir. Tez çalışmasında görseller üzerinde boyutlandırma işlemi yapılmış olup aynı boyuttaki görseller üzerinde çalışılmıştır. Çalışmada çapraz doğrulama testi kullanılmıştır. Bu şekilde yapay sinir ağının daha sağlam ve güvenilir bir hale gelmesi sağlanmış olacaktır. Çalışma sonunda doktora teşhis konusunda yardımcı bir sistem gerçekleştirilmiş olacaktır.

1.3.Tezin İçeriği

Birinci bölümde cilt hastalıkları hakkında genel bilgiler verilmiştir. İkinci bölümde daha önce yapılan örnek çalışmalar hakkında genel bir literatür çalışması yapılmıştır. Üçüncü bölümde öz niteliklerin çıkarılması, öz nitelikler çıkarılırken yapılan işlemler anlatılmıştır. Bu amaçla GWT, YSA, Sınıflandırma, Çapraz doğrulama teknikleri anlatılmıştır. Dördüncü bölümde üzerinde çalışılan cilt hastalıklarının teşhisi için yapılan uygulamanın detayları anlatılmıştır.

Son bölümde elde edilen sonuçların değerlendirilmesi ve öneriler yapılmıştır.

2.ÖNCEKİ ÇALIŞMALAR

Örnek çalışma kısmında daha önce genel olarak sınıflandırma ile ilgili yapılan çalışmalar incelenmiştir.

M.Seetha ve arkadaşları tarafından görüntü sınıflandırma için makine öğrenmesi ve algoritmalar diğer yöntemlerle karşılaştırma yapılmıştır. Bu karşılaştırma sonucu farklı parametrelerin sonucu değiştirdiği gözlemlenmiştir. Genetik algoritma ve destek vektör makineleri ile farklı sonuçlar elde edildiği gözlemlenmiştir[16].

P. Görgel tarafından görüntü işleme tekniklerine dayanılarak meme kanseri tanı çalışması yapılmıştır. Bu çalışmada görüntü işleme ve gabor dalgacık dönüşüm yöntemleri kullanılmıştır. Kitleler görüntü bölütleme ile gerçekleştirilmiş olup başarılı sonuçlar elde edilmiştir[17].

M. Sofu tarafından otomatik elma sınıflandırma çalışmasında üç farklı grupta çalışılmıştır. Görüntüler üzerinde görüntü işleme çalışmaları yapılmıştır. Görüntüyü yumuşatma, görüntü üzerindeki gürültüyü kaldırma, kontrast değişiklikleri yapılır. Görüntüdeki ön işleme çalışmasından sonra görüntü bölütlenir. Bu çalışma sonucu makine öğrenmesi ile elmanın rengi, boyutu ve leke durumuna göre farklı sınıflandırılmalar yapılmış olup ortalama %69 başarı elde edilmiştir[18].

Ö. Coşkun tarafından uydu görüntü verilerinin yapay sinir ağları ile sınıflandırılmasında da ön işleme teknikleri kullanılmaktadır. Ön işleme teknikleri olarak öz nitelikler çıkartılmaktadır. Özniteliklerde boyut yeniden ayarlanır. Bunun yanında bazen doku ya da momentum, entropi değerler kullanılmaktadır. Ön işleme niteliklerde amaç performans ve verimliliği artırmaktır. Bu yapılan çalışma çok katmanlı öğrenme algoritmasında farklı öğrenme kuralları ile çalışılmıştır. Her bir kuralda farklı sonuçlar elde edilmiş olup karşılaştırmaları yapılmıştır. Genel olarak başarımlar elde edilmiştir[43].

H. Toyran tarafından yapay zeka görüntü tanıma algoritması ile otomasyon sınıflandırma çalışması yapılmıştır. Yapılan bu çalışmada ön işleme çalışmalarından sonra öz nitelikler elde edilmiştir. Burada gereksiz öznitelikler çıkarılmış olup daha basit ve anlaşılır bir model oluşturulmuştur. Basit bir modelde daha fikir vermesinin yanında performansı daha yüksek olur. Çalışmada destek vektör makineleri ile KNN kullanılmış olup her ikisi karşılaştırılıp farklı sonuçlar elde edilmiştir. Makine görme sistemi gerçek zamanlı olup kısa sürede sınıflandırmaları yaparak başarılı sonuçlar vermiştir[19].

T. Ölmez tarafından yapay sinir ağı ile biyomedikal sınıflandırma işlemi yapılmıştır. Bu çalışmadaki sınıflandırmada genetik algoritmalar kullanarak probleme çözüm geliştirilmeye çalışılmıştır. Üç farklı hücre yapısı incelenmiş olup her biri için farklı uygunluk fonksiyonları kullanılmıştır. En iyi uygunluk dizisi için farklı genetik işlemler yapılmış olup sınıflandırma için de başarılı sonuçlar vermiştir[20].

A. Chawla ve arkadaşları tarafından malign melanom ve tüm cilt kanserli hücrelerinin üzerinde çalışma yapmışlardır. Bu çalışmada melanom benzeri özellikler taşıyan tümörleri 3 kategoride ele almak için yeni bir yapay sinir ağı yaklaşımı sunmuşlardır. Bu yeni yaklaşımla malign ve benign olan tümör görüntülerin sınıflandırılması için bunlar yapay sinir ağına verilip tümör şekline ve bağlı tümör rengine göre diskriminant özellikleri kullanılmaktadır. Bu yaklaşımla makul bir eğitim/test veri setleri için kullanılan gerçek malign ve benign deri tümörü görüntülerinde tümörlerin %80'ini doğru sınıflandırmışlardır[32].

Bakpo ve arkadaşları tarafından yapılan çalışmada bir cilt hastalığı teşhisi için bir çerçeve sunmuşlardır. Bu çalışmada yapay sinir ağlarının teşhis ve tanı %90 lara varan bir başarı elde etmişlerdir. Yapılan bu çalışmaya göre tanı ve teşhis işlemlerinde Ysa'nın başarılı bir yöntem olduğuna karar verilmiştir. Ysa özellikleri sayesinde etken organizmanın mutasyon yeni hastalık durumlarında ya da bilinen bir cilt hastalığının değişimini farketme yeteneğine sahiptir[31].

Andre Esteva ve arkadaşları tarafından yapılan derin öğrenme ağları ile cilt teşhisi ve kanser hastalıklarının sınıflandırması çalışmasında doğrudan görüntüleri elde etmede %60 başarı elde etmişlerdir. İkili cilt kanser hastalığının sınıflandırma çalışmasında ise %80'e varan başarı elde etmişlerdir. Bu çalışmada görüntüler aktivasyon ve harita filtrelerinde 6 farklı kategoride incelenmiştir. Yapılan çalışmada bu filtrelerin gabor benzerliği tespit edilmiş olup bunun yerine renk ve bölge dedektörleri(algılayıcıları) kullanıldığı tespit edilmiştir[41].

3.MATERYAL ve METOT

Başarılı bir örüntü tanıma sistemi için uygun algoritmanın yanında her sınıf için de uygun veri kümelerin elde edilmesi gerekir. Kullanılacak veri miktarı konusunda herhangi bir kural yoktur. Bu tez çalışmasında 3 farklı cilt hastalığına ait 300 tane veri seti kullanılmıştır. Veri kümesi için dermatology resources veri tabanı kullanılmıştır[44].

3.1.Öznitelik Çıkarımı

Tez çalışmasının en önemli aşamalarından biri öznitelik çıkarımıdır. Örüntü tanıma uygulamalarında veriler doğrudan sınıflandırıcının girişine verilmez. Amaç için gerekli olan görüntünün belli kısmı alınır. Bunun için bazı ön işleme aşamalarından geçirilir. Gürültü temizleme, filtreleme, normalizasyon, kontrast yayma, boyutlandırma bu aşamalardan bazılarıdır. Bilgisayardaki imgeler sayısal imgelerdir. Bu imgeler çoğu zaman büyük boyutlarda olurlar. Boyut dönüşümü sağlanarak daha kısa sürede eğitim işleminin bitmesi sağlanmış olur. Boyutu düşürülen örnek uzaylar sınıflandırıcılarda daha az parametre ile çalışmaya başlarlar.



Şekil 3.1.Tezde kullanılan örüntü tanıma sistemi

Bu sayede karar uzay ile örüntü uzay arasındaki dönüşüm daha kısa sürede gerçekleşmiş olur. Öznitelik için çeşitli algoritmalar kullanılmakta olup bu algoritmaların amaçlarından biri de görüntüye filtre uygulamaktır. Öznitelikte temel amaç veriyi en az yer kaplayacak şekilde ve en doğru şekilde kullanmaktır. Özellik çıkarımında çok fazla özellik zararlı olduğu için özellik sayısı az tutulmaya çalışılır. Küçük ve yeterince bilgilendirici olan bir özellik kümesi sayesinde sınıflandırma algoritmasının kompleks yapısı ve bu algoritmayı süreç boyunca yürütecek olan bellek ve zamana olan ihtiyacı azaltacaktır[23,43].

3.1.1.Gabor Dalgacık Dönüşümü (Gabor Wavelet Transform-GWT)

Gabor dalgacık dönüşümü gelişmiş canlılardaki görme sinirlerinin uzaysal frekans ve yönelim seçici niteliklerinden etkilenecek oluşturulmuştur. Gabor filtresi lineer bir filtre olup, itme(impuls) cevabı Gauss fonksiyonun harmonik bir fonksiyonla çarpılması sonucu

elde edilir. Gabor filtresinin itme yanıtı, Gauss fonksiyonun Fourier dönüşümü ile harmonik fonksiyonun Fourier dönüşümünün evrişimlemesidir(konvolüsyon)[27,28]. GWT sinyal işleme, uygulamalı matematik, ses ve görüntü sıkıştırma teknikleri gibi alanlarda sık olarak işlenmektedir. Gabor dalgacıkları görüntüleri etkin bir şekilde analiz etmeyi başaran filtrelerdir. Bir boyutlu sinyallerde bir arada olan zaman-frekans bölgesi belirtilmek zorundadır. GWT deki örnek bir işaret $\psi(t)$ fonksiyonun ölçüm ve kaydırması ile oluşturulmuş olan taban fonksiyonları ve analizlerine dayanır. Bu fonksiyonlar genelde yüksek ve düşük frekanslı fonksiyonlardır. Literatürde çok sayıda dalgacık fonksiyonu olup sinüs ve kosinüs ifadeleri fourier serisinin en temel ifadeleridirler. GWT, doku analizi, ayırma, plaka tanıma, bölütleme ve birçok bilgisayarlarlı görüntü uygulamasında kullanılmaktadır. GWT, rastgele seçilmiş aynı boyutlu bir imge üzerindeki farklı bölgeler arasındaki karakteristik farkını en aza indirmeyi amaçlar. Gabor dalgacıkları bu şekildeki dayanıklı özellikleri sayesinde bu piksel gruplarının aynı resmin parçaları olduğunu ifade etmeye olanak sağlar. Başka bir niteliği de farklı boyutlardaki resimlerin dayanıklı tanıma performanslarını yükseltmektedir. Bu yükseltmeye imkan sağlayan da GWT’ de kullanılan yüksek ve düşük bant filtrelerinin resmin farklı bölgeleri için aynı kalabilme durumudur[24-28].

Gabor dalgacıkları fourier ile analiz edilemeyen uzay ve frekans tabanındaki bilgileri ayrıştırma kabiliyetine sahiptir. Konik biçimli bir Gauss fonksiyonu ile çok yüzeyli bir dalganın çarpımından Gabor dalgacığı oluşturulur. Gauss’ un modüle edilmiş karmaşık üstel fonksiyonu sayesinde zaman ve frekans arasındaki denge sağlanır. Gabor bir boyutlu sinyalleri zaman ve frekans düzlemlerinde en az belirsizlikle tanımlamıştır. Daugman orijinal gabor filtresini iki boyutlu duruma getirmiştir. İki boyutlu gabor filtresi iki boyutlu Gauss ile düzenlenmiş olup belirli bir frekans ve yönelimdeki sinusoidal dalga şeklindedir[24,26].

Gabor dalgacıkları çoklu çözünürlük mantığıyla analiz edilir. Bu sayede belli bir çözünürlükte tespit edilemeyen görüntünün özelliği farklı bir çözünürlükte ortaya çıkarabilmektedir. GWT de ayrıntıyı yakalamak için küçük imgeler ve düşük renk karşıtlığında (kontrast) yüksek çözünürlük tersi durumda ise düşük çözünürlük kullanılır. Hem küçük resim ve düşük renk karşıtlığı ile büyük resim ve yüksek renk karşıtlığının aynı anda kullanıldığı durumlarda ise çoklu çözünürlük kullanılmalıdır. Dalgacık dönüşümü cebirsel olarak;

$$w(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(t) \cdot \overline{\psi\left(\frac{t-b}{a}\right)} dt \quad (3.1)$$

Şeklinde tanımlanır. Burada $a>0$ ve $b \in \mathbb{R}$ olmak üzere;

$a \rightarrow$ ölçekleme/yayılma kat sayısını,

$b \rightarrow$ dönüşüm/öteleme kat sayısını

$f(t) \rightarrow$ işareti ; Ψ , ana dalgacığı $w(a,b)$ ise işaretin sürekli dalgacık dönüşümünü ifade eder.

$$g(x; y, \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi \frac{x'}{\lambda} + \psi\right)\right) \quad (3.2)$$

Verilen (3.2) nolu fonksiyon sayesinde çekirdek matris hesaplanır. Bu fonksiyonun dönüş değeri karmaşıktır.

Gauss elemanları sinüs ve kosinüs tarafından modüle edilmektedir. Modüle edilen bu elemanlar Gabor dalgacığı öznelilik çıkarımında kullanılmaktadır. Gabor dalgacığını tek boyutlu ve iki boyutlu olarak incelemek mümkündür. Tek boyutluyu da gerçek ve imgesel(sanal) şeklinde özelliklere sahiptir. Gauss süzgeci ile modellenmiştir kosinüs sinyali gerçek, sinüs sinyali ise imgesel(sanal) özelliğini belirtir.

Genelde 3.3 ve 3.4 nolu denklemler kullanılarak gerçel ve imajinal değerler hesaplanır. Bu hesaplama sonucunda bu değerlerin toplamının karekökleri alınarak bulunan sonuç kullanılabilir[45].

$$g(x; y, \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (3.3)$$

Verilen(3.3) nolu fonksiyon gerçel değer döndürür.

$$g(x; y, \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \sin\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (3.4)$$

Verilen (3.4) nolu fonksiyon ise imgesel(sanal) değer döndürür. Bu fonksiyonların sonuçlarını kullanabilmek için her iki fonksiyonun değerleri hesaplanır. Hesaplama işlemi değerlerin karelerinin toplamının karekökleri alınarak yapılır.

$$\begin{aligned}x' &= x \cos \theta + y \sin \theta \\y' &= -x \sin \theta + y \cos \theta\end{aligned}\tag{3.5}$$

Yukarıdaki (3.5) nolu fonksiyondaki eşitliği elde etmek için kullanılan dönüşüm matrisi sayesinde (3.7) nolu denklemdeki gibi olur.

$$R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}\tag{3.6}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}\tag{3.7}$$

Θ açısının pozitif ve negatif durumuna göre döndürme işlemi gerçekleşir. Θ açısı pozitif iken saat yönünün tersine döndürme işlemi yapılırken negatif olması durumunda ise saat yönünde döndürme işlemi gerçekleşir.

Sinüs ve kosinüsteki dalgaların frekansına göre süzgecin merkez frekansı belirlenir. İki boyutlu gabor filtresi aşağıdaki denklem kullanılarak elde edilir.

$$\begin{aligned}g(x; y, \lambda, \theta, \psi, \sigma, \gamma) &= \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \\x' &= x \cdot \cos \theta + y \cdot \sin \theta \\y' &= -x \cdot \sin \theta + y \cdot \cos \theta\end{aligned}\tag{3.8}$$

$g \rightarrow$ gabor çekirdeğini ve $g(x,y)$ fonksiyonu gabor fonksiyonunu göstermek üzere;

$\theta \rightarrow$ desenin açısını, $\lambda \rightarrow$ desenin dalga boyunu $\sigma \rightarrow$ standart sapmasını vermektedir. Ayrıca Ψ fazın ofset değerini, γ ise uzamsal görüş açısını belirtir. σ değeri büyük olduğunda λ ve bu değere yakın dalga boylarındaki desenler bulunur. Gabor çekirdeği istenilen deseni bulacak şekilde ayarlanır[27,28,45].

Θ yönelim açısı gabor dalgacık fonksiyonunda paralel çizgilerin yönünü temsil etmektedir. Açılar artırılıp azaltabilir. Açılar arttıkça matrisin döndürme yönüne göre açıları artan paralel çizgiler(doğrular) tespit edilmektedir.

Gabor dalgacığında band genişliği için sıfırdan büyük bir değer kullanılır. Band genişliği değerinin düşük olması durumunda standart sapma değeri artar. Bunun sonucunda süzgeçli görüntülerdeki paralel şeritlerin sayısı artmaktadır.

Gama resmin genişlik ve uzunluk oranı için kullanılan bir değerdir. Gauss fonksiyonu sonucu oluşan eğrideki genişlik ve uzunluk oranını filtrelenen imgeye de aynı oranda uygulanır.

π kosinüs çarpanının faz açısı olarak ta adlandırılmaktadır. Faz açısını değeri 0 ve 180' e eşit iken gauss fonksiyonun eğrişi merkezde konumlanmaktadır. Farklı faz açıları x eksenine göre grafiği sağa ve sola kaydırıp bu eğrinin y eksenine göre asimetrik olmasını sağlar. Asimetrik durum görüntüye de yansıdığı için süzgeçli bir görüntülerdeki paralel olan doğrular da asimetrik olmaktadır.

Gauss Kernel ile karmaşık bir sinüsodin çarpımı sonucu karmaşık bir Gabor filtresi ortaya çıkar. İki boyutlu bir GDD, $G(\vec{x})$ imgesinin evrişimi ile ifade edilmektedir.

$$J_i(\vec{x}) = \int G(\vec{x}') \psi_i(\vec{x} - \vec{x}') d\vec{x}' \quad (3.9)$$

Burada $G(\vec{x})$, $\vec{x}$ imge yoğunluğunu gösterirken $\psi(x)$ ise bir gabor filtresinin kümesini göstermektedir[39,44].

Çekirdek matrisin boyutu elle girilebilir. Kare matris alındığında en iyi optimum n değer $8 \cdot \sigma / \gamma$ ifadesine en yakın en küçük tam sayı belirlenmeye çalışılır[45-46]. Kosinüs çarpanının faz açısı olarak kabul edilir. Değerin sıfır ve yüz seksen olması durumunda gauss eğrisinin de merkezdeki değeri sıfır olur.

Gabor dalgacığın ölçek ve yönelim çiftine ait olmak üzere hesaplama işlemi yapılır. Tez çalışmasında resmin(nesnenin) 8 dalgacık matrislerinin her birine ait istatistiksel parametrelerinin bir vektör meydana getirecek şekilde ve ardışık bağlanması sonucu Gabor öz nitelik vektörleri oluşturulmuştur. GWT matrisinden standart sapma, entropi ve ortalama değerleri hesap edilmiştir. İşlem sonunda her bir resme ait 8 GWT matrisinin istatistiksel değerlerin ardışık eklenip toplam olarak 24 uzunluğundaki öz nitelik vektörü elde edilip sınıflandırıcıya girdi için hesap edilmiştir[23,46].

Gabor Dalgacık Dönüşümlü(Gabor Wavelet Transform-GWT) Değişkenler

Gri tonlamalı nesnelerden elde edilen matrisin boyutu $M \times N$ ve $J(i,j)$ ise kullanılacak matris şu şekilde hesaplanır[41,45].

Ortalama

Bir imgeye ait bütün görüntünün parlaklığını ifade etmektedir.

$$m_{ij} = \frac{1}{MN} \sum_i^N \sum_j^M J(i,j) \quad (3.11)$$

m_{ij} ortalama değeri, MN matrisin boyutunu $J(i,j)$ ise matris içindeki her bir elemanı göstermektedir.

Standart Sapma

Standart sapma veri değerlerinin ortalamadan sapmalarının kareler ortalamasının karekökü olarak tanımlanmaktadır. Ayrıca ikinci dereceden bir Gauss fonksiyonu olup standart sapma değerini belirtip karışıklığın (kontrastın) bir ölçüsü olarak kullanılmaktadır.

$$S_{ij} = \frac{1}{MN} \sum_i^M \sum_j^N (I_{ij} - m_{ij})^2 I_{ij} \quad (3.10)$$

S_{ij} standart sapmayı, m_{ij} ortalama değeri I_{ij} görüntüdeki matrisin noktalarının boyutunu verir.

Entropi

Genel olarak düzensizliği ifade eder. Bir görüntü üzerindeki gri değişiminin bir ölçüsü olarak kullanılmaktadır. Bir imge üzerindeki görüntüye uygulandığında imge üzerindeki belirsizliği ya da gri renk düzeyi hakkında bilgi vermektedir. Yüksek bir entropi değerine sahip bir sistemin durumunu anlatmak için o sistem hakkında daha çok bilgiye ihtiyaç duyulmaktadır. Görüntüdeki tüm piksellerin aynı gri seviye yoğunluğuna sahip olması durumunda en küçük entropi ile, tersi durumda ise en büyük entropi ile ifade edilmektedir.

$$ent_{ij} = -\sum_{i,j} j(i,j) \cdot \log(j(i,j)) \quad (3.12)$$

3.2.Sınıflandırma Yöntemleri

Sınıflandırma örüntü tanıma ve bilgisayar öğrenme sistemlerinde bu sistemlere ait öznitelikler kullanılarak bir karar mekanizması ile sınıfının belirlenme işlemidir. Sınıflandırma veri seti gruplara ayrılarak hangi veri setinin hangi gruba ait olduğunu tahmin edebilmektir. Pek çok sınıflandırma yöntemi kullanılmaktadır. Bu

sınıflandırmalardan bazıları k-NN, yapay sinir ağıları, genetik algoritmalar, destek vektör makineleridir[10,16].

Özniteliklerin seçimi sınıflandırmanın başarımı üzerinde etkilidir. Özniteliklerin yanında eğitim ve test için kullanılacak veri kümelerinin de uygun belirlenmesi sınıflandırmadaki başarı oranını etkiler. Eğitim ve test veri kümelerinin belirlenmesi için rastgele yöntemlerin yanında kullanılan yöntemlerden bazıları da 1/3 veya 2/3 gibi oranlardır.

Bu çalışmada çapraz doğrulama kullanıldığından veri kümemizin tamamı hem eğitim hem de test kümesi için kullanılmıştır. Veri alt kümesinin bir grubunu rastgele test için geri kalanı eğitim için kullanılmıştır. Bu şekilde 10 aşamada verilerin tümü hem eğitim hem de test için kullanılarak YSA çapraz doğrulama testi ile ortalama sonuç elde edilmiştir.

3.2.1. Yapay Sinir Ağları

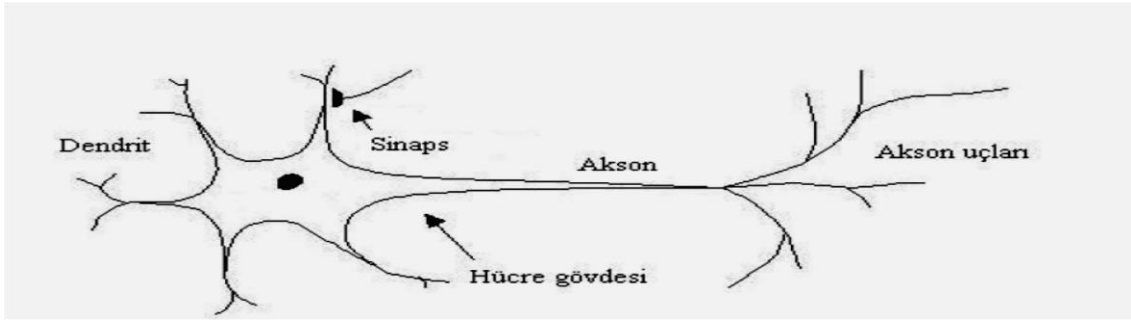
Yapay sinir ağıları, insan beyninin temeli olan nöronlara benzer olarak oluşturulmuş farklı topoloji ve ağ modelleriyle birbirine bağlanmasıyla bir araya gelen karmaşık sistemlerdir. Yapay sinir ağıları birbirleriyle etkileşim halinde olan ve pek çok nörondan oluşan paralel ve hiyerarşik sistemlerdir. Yapay sinir ağı farklı ve bir bilgi işleme tekniği olup algoritmik programlamaya alternatif olarak ortaya çıkmıştır. Yapay sinir ağlarında adım adım program yürütümü yerine sinirsel ağ ilişkilendirmesi ve kendi kurallarını üreterek sonuçları örneklerle düzenleme yoluna gider [6,10,18].

Genel anlamda yapay sinir ağlarındaki işlemciler iki topolojide sınıflandırılmaktadır. Bu topolojiler ileri beslemeli ve geri beslemeli ağlar şeklinde iki ana grupta düşünülebilir. İleri beslemeli ağlarda nöronlar; giriş, ara ve çıkış olarak adlandırılan katmanlar aracılığıyla düzenlenir. Her bir katmandaki nöronlar; bir sonraki katman nöronları ile bağlantı ağırlıkları vasıtasıyla ilişkilidir. Katmanların kendi aralarında herhangi bir bağlantıları olmayıp, bilgi girdi katmanından çıktı katmanına doğru iletilir. Bu aktivasyon yönü şeklinde adlandırılır. Bu tür yapay sinir ağına örnek olarak tek ve çok katmanlı sinir ağları (MLP - Multi-Layer Perceptron ve LVQ - Learning Vector Quantization) verilebilir.

Geri beslemeli ağların en önemli özelliği; katmanlar arasındaki nöronlar biri biriyle bağlantılı olup ve dinamik bir hafızaya sahip olmalarıdır. Bu tür ağlara Kohonen ağı, SOM,

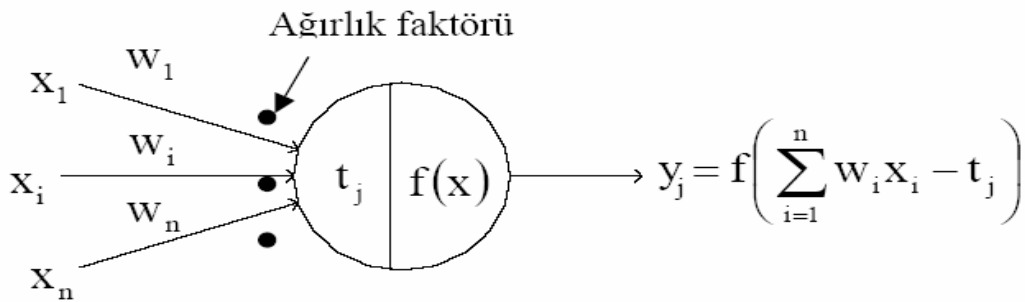
Elman-Jordan ve Hopfield ağı örnek olarak verilebilir. Bu tür ağların eğitilmesi takviyesiz öğrenmeye örnek olarak verilebilir.

Yapay sinir ağlarındaki işleme elemanları biyolojik olarak insan beynindeki nöronlarla benzerlik göstermektedir. Şekil 3.2. de Dendrit adı verilen yapı, diğer hücrelerden bilgiyi alıp bunu aksonların uçlarından diğer uçlara göndermektedir. Bilgiyi diğer hücrelere transfer etme işini aksonlar gerçekleştirmektedir. Bundan dolayı aksonlar nöron çıkışları olarak görev yapmaktadırlar. Sinapslar ise akson ile dendrit arasındaki bağlantıyı gerçekleştirir.



Şekil 3.2. Biyolojik sinir hücresi ve bileşenlerinin gösterimi[20]

Yapay sinir ağlarının işleyişi de bu sisteme benzer olarak geliştirilmiştir. McCulloch ve Pitts 1940 yılında yaptıkları bir çalışmayla nöronun, mantık sistemlerinde basit eş değer yapısıyla modellenebileceğini ortaya koymuşlardır[2]. Yaptıkları bu çalışmanın sonunda Şekil 3.3. 'te altta görüldüğü gibi bir yapay sinir ağı modeli geliştirmişlerdir.



Şekil 3.3. Yapay sinir hücresi [20]

Genel olarak verilen bir örnek veri setine karşılık çıkış verilmiş olan öğrenme kuralına göre ağırlık değerleri otomatik olarak güncellenmektedir. Eğitilen ağ ağırlık değerlerinin son durumuna göre verilen herhangi örnek bir veri setinin sonucunun tahmini değerini yapabilmektedir. Yapay sinir ağı, sinir hücrelerinin ileri beslemeli ve geri

beslemeli bağlantı şekilleri ile birbirine bağlanması sonucu oluşmaktadır. Amaca uygun hedefler için ve farklı alanlarda kullanılmaya uygun birçok yapay sinir ağı modeli geliştirilmiştir. Bu modellerden bazıları Perceptron, Adaline, MLP, LVQ, Hopfield, Recurrent, SOM, ART ve PCA gibi modellerdir. Öğrenme çeşitlerinden öğreticili öğrenme, öğreticisiz öğrenme, destekleyicili öğrenme ve karma stratejiler kullanılmaktadır[4].

Bir sinir n kadar ağırlıklandırılmış girişi toplayıp sonucu lineer olmayan bir fonksiyondan geçirmektedir. Herhangi bir katmandaki j . elemana gelen toplam girdi sayısı önceki katmandaki elemanların y_i çıktılarının bağlantılar üzerindeki w_{ij} ağırlıkları ile hesap edilmiş ağırlıklı toplamdır.

$net_j = \sum_i w_{ij} y_j$ birimin çıkışı, bu değerin bir eşik değerden çıkartılıp lineer olmayan bir fonksiyondan geçirilmesiyle $f(x) = \frac{1}{1+\exp(-\sum_i w_{ij} y_j)}$ olarak hesaplanır.

Sinir sistemi birbiri ile iletişim halinde olan sinir hücrelerinden oluşmaktadır.

3.2.1.1. Biyolojik Sinir Ağlarının Yapısı

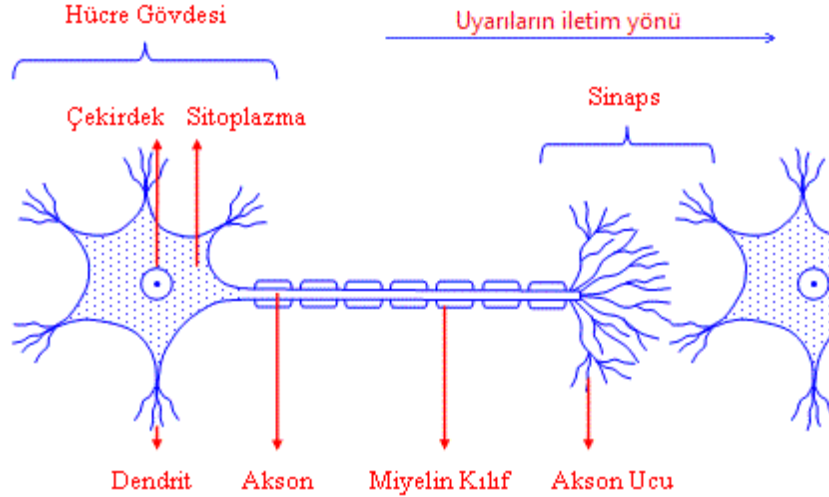
Sinir sisteminin temel fonksiyonel birimi olup beyin çalışmasını sağlayan en önemli öğelerden biridir. Biyolojik sinir ağlarında beyinde bulunan birçok sinir hücresinin bir araya gelmesi ile oluşur. Biyolojik sinir ağları insanın davranışlarını, çevresini tanıması duyu organlar aracılığıyla geliştirdiği anlama ve kavrama mekanizmalarını devreye sokarak olaylar arasındaki ilişki öğrenir. Milyarlarca sinir hücresinin bir araya gelmesiyle sinir ağları oluşmaktadır. Her bir sinir hücresi on binlerce sinir hücreleri ile bağlantı olup birbirlerinden uyarı almaktadırlar. Bir hücreyi gövde(soma), gövdeye giren işaret alıcıları dentrit, iletiler(akson) ve sinapslardan oluşmaktadır. Bir hücreden gelen uyarılar alıcı olarak dentritlerin uçlarından alınıp aksonların uçlarından başka hücelere iletirler.



Şekil 3.4. Sinir sisteminin blok gösterimi[20].

Hücreler arasındaki geçiş noktaları sinaps olarak adlandırılır. Sinapslar ise bilginin iletiminden sorumludurlar. İletim elektriksel olarak gerçekleşmektedir. Sinir hücreleri tek kutuplu, iki kutuplu ve çok kutuplu olarak kümelendirilir. Miyelin kılıf aksonlardaki

yalıtımı sağlayarak uyarı iletim hızını arttırır. Ayrıca nöronlarda enerji tasarrufu sağlar. İnsan beyni geleneksel bilgisayarlara göre daha hızlı olup daha kısa sürede tanıma ve öğrenme gerçekleştirmektedir. Biyolojik sinir ağlarının performansları çok yüksek ve karmaşık olayları işleyebilecek yetenektedir[19-20]



Şekil 3.5. Bir biyolojik sinir hücresinin yapısı[20,23,26]

3.2.1.2.Yapay Sinir Hücresi ve Elemanları

Biyolojik sinir ağlarındaki işleyişe benzer şekilde çalışmaktadır. İnsan beyninin çalışma mantığını temel alan yapay sinir hücresi ile biyolojik sinir hücresi arasındaki benzerlik Tablo3.1. de gösterilmiştir.

Tablo 3.1. Biyolojik ve yapay sinir ağı sisteminin karşılaştırılması

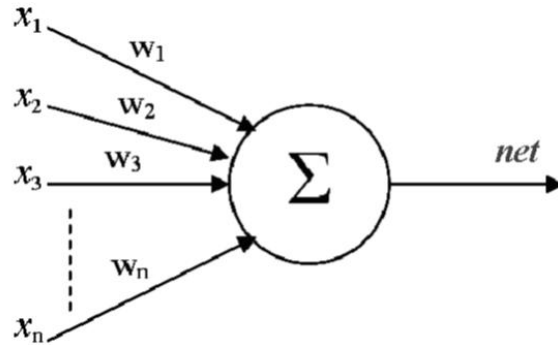
Biyolojik Sinir Sistemi	YSA Sistemi
Nöron(Sinir Hücresi)	İşlem elemanı
Dentrit	Toplama fonksiyonu
Hücre gövdesi	Transfer fonksiyonu
Aksonlar	Eleman çıkışı
Sinapslar	Ağırlıklar

Bir yapay sinir ağını Girdi Katmanı, Ara/Gizli Katman ve Çıktı Katmanı olarak üç kısma ayrılmaktadır. Belirli bir sistemde dizilen katmanlar ağırları oluştururlar. Katmanların

belirli bir düzende dizilmesiyle ağlar oluşur. Ağlar katmanların sayısı, düzeni ve öğrenme algoritmalarına göre isimlendirilirler. Yapay sinir hücreleri süreci içerisinde 5 temel eleman vardır[9].

Bunlar:

- **Girdiler:** Bir yapay sinir hücresine dışardan, başka bir hücreden veya kendi kendisine gelen bilgilerdir. Öğrenmesi istenen örnekler tarafından belirlenirler. X_i giriş değeri olmak üzere, N toplam giriş değeri, i toplam indis değeridir. Buna göre $i=1.....N$ 'dir.
- **Ağırlıklar:** Yapay bir hücredeki bilginin önem derecesi ve hücre üzerindeki tesirini göstermektedirler. Ağırlıklar büyüklük veya küçüklük değerine göre önemli ya da önemsiz anlamına gelmemektedir. Her bir ağırlık bir girişe ait olup sayısal değerdendirler.
- **Toplama fonksiyonu:** Bir hücreye gelen toplam giriş ve ağırlık değerlerini kullanarak net girdiyi hesaplar. Net girdiyi hesaplamak için farklı fonksiyonlar kullanılır. Bu fonksiyonlardan en çok kullanılanı ağırlıklı toplamı bulandır. Bu şu şekilde formülize edilmektedir.



Şekil 3.6. Basit bir yapay sinir hücresinin yapısı[23,26]

$$net = \sum_i^n X_i W_i \quad (3.13)$$

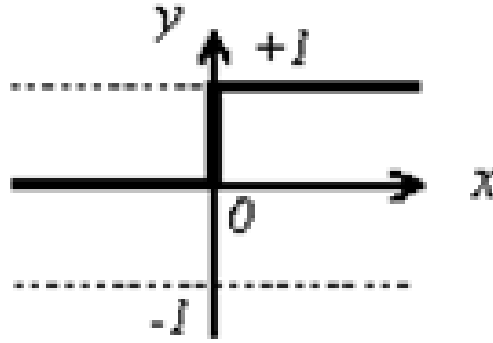
Giriş değerleri kendi ağırlıkları ile çarpılıp elde edilen değerlerle toplanır. Bu şekilde net girdi değeri bulunmuş olur. n bir hücreye gelen toplam giriş(süreç elemanını), X girişleri, W ise ağırlık değerlerini göstermektedir. Bir problem için çoğunlukla deneme yanılma yolu ile toplama fonksiyonu belirlenmektedir[16-17].

Aktivasyon fonksiyonu: Aktivasyon fonksiyonu gelen net girdiye karşı çıktı değerini üretir. Genelde nöronun çıkış genliğini sınırlamak amacıyla kullanılır. [0,1] veya [-1,1] kapalı aralığı bir nöronun normalize edilmiş genliği ifade edilmektedir. Transfer

fonksiyonu genellikle doğrusal olmayan bir fonksiyonlardan seçilir. Çünkü doğrusal fonksiyonlarda girdi ile çıktı değerli orantılıdır. YSA denemelerinin başarısızlıkla sonuçlanmasının nedenlerinden biri de budur. Genelde kullanılan bazı fonksiyonlara değinilmiştir[5].

Eşik Değer Fonksiyonu

Eşik fonksiyonu kullanılarak yapılmış bir sinir hücresi nöron literatürde McCulloch-Pitts modeli olarak adlandırılır. Fonksiyonun grafiği Şekil 7’de gösterilmiştir [2].



Şekil 3.7. Eşik değer fonksiyonu

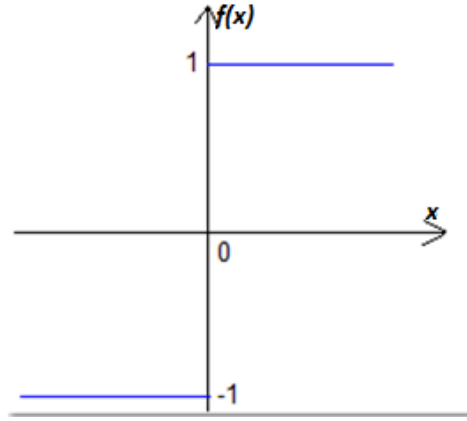
$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (3.14)$$

$y=f(x)$ Eşik fonksiyonu

Burada x giriş fonksiyonu, y ise çıkış fonksiyonudur. Giriş değeri 0’den büyük ve eşit ise çıkış değeri 1, giriş değeri sıfırdan küçük ise çıkış değeri 0 olur.

Adım (Step) Aktivasyon Fonksiyonu

Girilen x girdi değerinin belli bir sınırın altında veya üstünde olmasına göre hücre değeri $[0,1]$ değerini alır. Şekil 3.8 de bu durum verilmiştir.

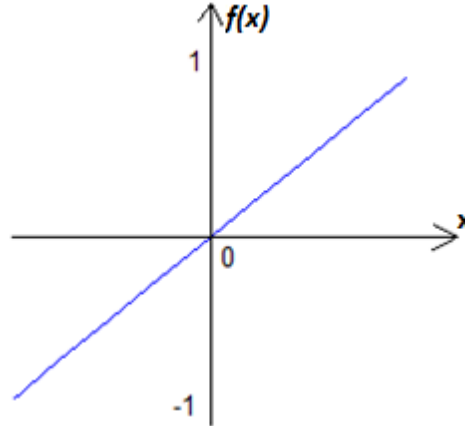


Şekil 3.8. Adım(Step) aktivasyon fonksiyonu

$$f(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (3.15)$$

Doğrusal (Lineer) Aktivasyon Fonksiyon

Doğrusal fonksiyonları çözmek amacıyla kullanılır. Gelen girdiler olduğu gibi sınır hücrenin çıktısı olarak kabul edilir.



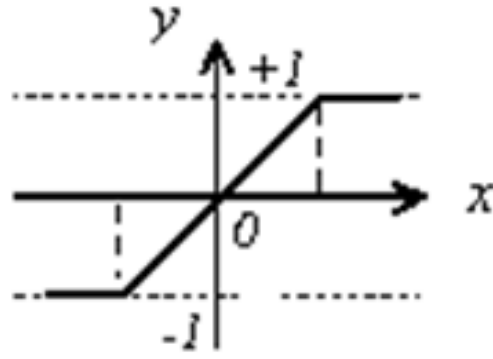
Şekil 3.9. Doğrusal Fonksiyon

$$f(x) = K.x \quad (3.16)$$

K burada herhangi bir sabit sayıyı ifade etmektedir.

Kısmi Doğrusal Fonksiyon

Doğrusal olmayan bir genlik artımı sağlayan bu aktivasyon fonksiyonu Şekil 10'da gösterilmiştir. Eğer doğrusal bölgedeki genlik arttıran katsayı yeterince büyük alınırsa parçalı doğrusal fonksiyon eşik fonksiyonuna dönüşür.



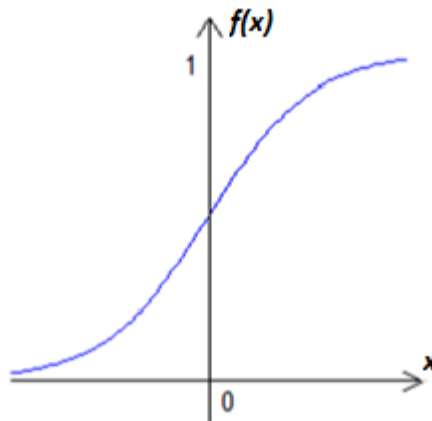
Şekil 3.10. Kısmi doğrusal fonksiyon

$y=f(x)$ kısmi doğrusal fonksiyon

$$f(x) = \begin{cases} 1 & x \geq 1/2 \\ x & -\frac{1}{2} < x < \frac{1}{2} \\ 0 & x \leq -1/2 \end{cases} \quad (3.17)$$

Sigmoid Fonksiyonu

Yapay sinir ağları oluşturulurken sık kullanılan aktivasyon fonksiyonlarından biridir. Doğrusal ve doğrusal olmayan davranışlar arasında denge sağlayan sürekli artan bir fonksiyon olarak tanımlanır. Logsig fonksiyonu olarak ta adlandırılır. 0 ve 1 aralığında dinamik değerler almaktadır. Sigmoid fonksiyona bir örnek logsig fonksiyondur ve Şekil 3.11.'de gösterilmiştir.



Şekil 3.11. Lojistik sigmoid fonksiyonu

Burada x netgirdi değerini ifade etmektedir. Sigmoid fonksiyonun türevi alınabilmektedir. Türev, bir fonksiyon eğrisi üzerinde alınan herhangi bir noktadan çizilen ve o eğriye teğet

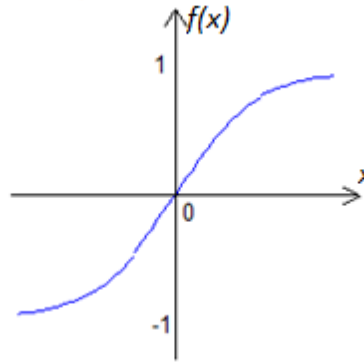
olan doğrunun eğimi olarak ifade edilmektedir. Kısaca bağımlı değişkenler arasındaki değişim olarak adlandırılmaktadır.

$$f(x) = \frac{1}{1+e^{-x}} \quad (3.18)$$

Türev, bir fonksiyon eğrisinin üzerinde alınan herhangi bir noktadan o eğriye çizilen teğettir.

Hiperbolik Tanjant Fonksiyonu

Hiperbolik tanjant fonksiyonu da sigmoid fonksiyon benzer bir fonksiyondur. Şekil 3.12.'de bir hiperbolik tanjant fonksiyonu verilmiştir.



Şekil 3.12. Hiperbolik tanjant fonksiyonu

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3.19)$$

Bu fonksiyonda netgirdi(x) değeri tanjant fonksiyonundan geçirilerek hesaplanır. Sigmoid fonksiyonuna benzer olup çıkış değerleri -1 ile 1 arasında değişmektedir[2-27,47].

Çıktılar

Eşik fonksiyonundan çıkan değer işlem elemanının da çıktısı olup bazı durumlarda işlem elemanının bu çıktıyı bir çıktı fonksiyonu ile bir dönüşüme uğratması gerekebilmektedir. Bu çıktı, ağın yapısına göre, girdi olarak başka bir işlem elemanına veya bir dış bağlantıya gönderilir[1,13].

3.2.1.3.Yapay Sinir Ağlarının Genel Özellikleri

Yapay sinir ağları paralel dağıtılmış işleme yapısı, öğrenebilme ve genelleme yapma yeteneklerinden dolayı üstün hesaplama ve bilgi işleme gücüne sahiptir. Birden çok nöronun eş zamanlı çalışması ve aktif olabilme durumu mevcuttur. Genelleme yeteneği sayesinde eşleştirme ya da öğrenme süresi boyunca karşılaşılmayan girişler için de yapay sinir ağı uygun reaksiyon gösterir.

Bütün modeller için geçerli olan tipik özellikleri aşağıdaki gibi sıralanabilir. Bunlar;

- Yapay sinir ağları makine öğrenmesi gerçekleştirirler. Yapay sinir ağlarının temel işlevlerinden biri bilgisayarların öğrenmesini sağlamaktır. Burada olayları öğrenip benzer olaylar karşısında uygun veya benzeri kararlar vermeye çalışırlar.
- Çalışma yöntemi bilinen programlama yöntemlerine benzememektedirler. Klasik programlama yöntemlerinin uygulandığı bilgi işleme yöntemlerinden farklı bir bilgi işleme yöntemine sahiptirler.
- Bilginin saklanması: Yapay sinir ağlarında bilgi ağın bağlantılarında saklanmaktadır. Klasik programlardaki gibi veriler bir veri tabanında veya programın içinde gömülü değildir. Bilgiler ağın üzerinde saklandığı için bunların ortaya çıkartılması ve yorumlanması zordur.
- Yapay sinir ağları örnek kümeleri kullanarak öğrenme gerçekleştirirler. Yapay sinir ağları örnekleri kullanarak eğitilirler. Kullandığı örnekler sayesinde bağlı olduğu olay hakkında genelleme yapma yeteneğine kavuşturulmuş olurlar. Bütün yönleriyle olay ağı gösterilmez ve örnekler başarılı bir şekilde sunulmazsa başarılı sonuçlar elde edilmez veya çok düşük olur.
- Yapay sinir ağlarının güvenli çalışabilmeleri için öncelikle eğitilip performanslarının test edilmesi gerekmektedir. Yapay sinir ağlarının eğitim işlemleri için var olan örneklerin ağı gösterilmesi ve ağın kendi mekanizmalarını çalıştırarak örnekteki olaylar arasındaki ilişkileri belirlemesi gerekir. Ağı eğitmek için örnekler eğitim ve test veri seti olarak ikiye bölünür.
- Görülmemiş örnekler hakkında bilgi üretebilirler. Ağ öğrenme aşamasından sonra kendisine gösterilen test örnekleri hakkında genellemeler yaparak bilgi üretebilir.
- Algı durumlarında kullanılabilirler. Ağlar daha çok idrak etmeye yönelik bilgileri işlemede kullanılıp başarılı sonuçlar elde ederler.

- İlişkilendirme ve sınıflandırma yapabilirler. Ağlar genel olarak kendisine örnekler halinde verilen örüntüleri ilişkilendirir veya örnekleri kümelendirerek belirli sınıflara ayırarak yeni gelen bir örneğin hangi sınıfa ait olduğuna karar verir.
- Örüntü tamamlama gerçekleştirebilirler. Bazı durumlarda ağa eksik bilgileri içeren bir örüntü (pattern), veya bir şekil verildiğinde ağ bu eksik bilgileri bularak çözüm üretir.
- Kendi kendine organize olma ve öğrenme yeteneğine sahiptir. Yapay sinir ağları örnek veri kümeleri ile kendisine gösterilen yeni durumlara uyum sağlama ve devamlı yeni olayları öğrenebilmesi ve eğitilmesi mümkündür.
- Eksik bilgi ile çalışabilmektedirler. Eğitilen yapay sinir ağları eksik bilgiler ile çalışıp ve yeni gelen örnek verilerde eksik bilgi olmasına rağmen sonuç üretebilmektedirler. Yapay sinir ağlarının eksik bilgilerle çalışması çoğu zaman ağın performansını etkilemez.
- Hata toleransına sahiptirler. Eksik bilgiler olduğu halde yapay sinir ağlarının çalışması hatalara karşı toleranslı olmasını sağlamaktadır.
- Belirsiz, tam olmayan bilgileri işleyebilme yeteneğine sahiptirler. YSA'lar tam olmayan bilgileri işleyebilme yetenekleri sayesinde olayları öğrenip sonra bu belirsiz bilgi işlemleri altında ağlar öğrendikleri olaylar ile ilgili ilişki kurup karar verirler.
- Dereceli bozulma (Graceful degradation) gösterirler. Yapay sinir ağlarının hatalara karşı toleranslı olmaları bozulmalarının da dereceli olmasına neden olmaktadır. Bozulma genelde eksik bilgi veya hücrelerinin bozulmasından kaynaklanmaktadır.
- Dağıtık belleğe sahiptirler. Yapay sinir ağlarında bilgi ağa yayılmış durumdadır. Hücrelerin birbirleri ile bağlantılarının değerleri ağın bilgisini gösterir. Tek bir bağlantının bir anlamı olmayıp, bu tür ağlarda, ağın tamamı öğrendiği olayın bütünü karakterize etmektedir. Bundan dolayı bilgiler ağa dağıtılmış durumdadır.
- Sadece sayısal ile çalışabilmektedirler. Yapay sinir ağları sadece sayısal bilgiler ile çalışmaktadırlar. İmgeler ve sembollerle gösterilen bilgilerin sayısal değerlere çevrilmesi gerekir.

Bu özelliklerin dışında her modelin kendisine has özellikleri olabilmektedir. Klasik bilgisayar yazılım teknolojileri ile çözülemeyen birçok problem yapay sinir ağları ile çözülebilmektedir[14,15,32].

3.2.1.4.Yapay Sinir Ağlarının Avantaj ve Dezavantajları

Ysa'nın sahip olduğu avantajlar şu şekilde sıralanabilir.

- Matematiksel olarak biçimlenmesi güç ve karmaşık problemleri çözme imkanına sahiptir.
- Öğrenebilme özelliği sayesinde bir örnekten yola çıkarak benzer örnekler hakkında genelleme yapma kabiliyetine sahiptirler.
- Bilgiler ağ ağırlıklarında saklanıldığı için ve ağın tamamına yayıldığından dolayı bir veya birden fazla sinir hücresinin işlevini kaybetmesi hataya yol açmaz.
- Ysa sadece probleme ilişkin örneklerle ihtiyaç duyduğundan dolayı olağan bir programlama ile sorunu çözebilmektedir.
- Ysa aynı vakayla ilgili farklı veriler meydana getirdiğinde de yeniden eğitilip ortama uyumu sağlanabilmektedir.
- Sınıflandırma ve örüntü tanıma özelliği sayesinde verileri kümelendirip sonradan gelen verinin hangi kümeye ait olduğunu tahmin edebilmektedir.
- Yapay sinir ağlarının öğrenmesi için ağdaki örneklerin belirlenip hedef çıktılara göre ağın eğitilmesi gerekmektedir. Seçilen örnekler ile ağın başarı düzeyi doğru orantılıdır.
- Ysa 'lar paralel çalışma düzenleri sayesinde eş zamanlı uygulamalarda kolaylıkla kullanılabilirler.

Ysa'nın temel bazı avantajlarından bahsettikten sonra bazı dezavantajları da vardır. Bunlar;

- Model seçim ve ağ topoloji belirleme işlemlerinde bir kural olmaması kullanıcının uzmanlık derecesi ve deneme yanılma yoluna göre seçim yapmaktadır.
- Paralel çalışma mantıklarından dolayı donanım bağımlı çalışmaktadırlar. Bunun içinde paralel çalışan merkezi işlem birimine gereksinim duymaktadırlar.
- Ysa'lar bir sorunu çözmeye çalışırken ağın durumu ile ilgili bilgi vermez.
- Ysa'ya verilecek problemin sayısal değerlere çevrilmesi gerekir. Problemi sayısal değerlere kullanıcılar çevireceği için ve sayısal değerler performansı etkilediği için başarı kullanıcının uzmanlık derecesine bağlıdır.
- Ysa'nın eğitim süresi belli olmadığından örneklerde belli bir hata değerinin altına düşürüldüğünde eğitim tamamlanmış olur[9,17,47].

3.2.1.5.Yapay Sinir Ağlarının Kullanım Alanları

Yapay sinir ağları; sınıflandırma, modelleme ve tahmin amaçlı uygulamalarının yanında başka pek çok alanda uygulanmaktadır. Karakter tanıma, cilt hastalık teşhisi, diller arasında otomatik çeviri yapma, sinyal işleme ve filtreleme, optimizasyon, kontrol, veri sıkıştırma ve filtreleme, üretim uygulamaları, desen tanıma ve doğrusal olmayan sinyal işleme ve sistem modelleme gibi alanlarda kullanılmaktadır.

Ysa'lar bunun yanında pek çok uygulama alanında da kullanılmaktadır. Bunlar; Arıza analizi ve tespiti sayesinde elektrikli makinelerin, uçak veya bileşenlerin ya da parçaların kalan sürelerinin tahmininde kullanılmaktadır. Bunun yanı sıra otomatik yol izleme, yol durumuna göre sürüş analizi, kredi müracaat değerlendirme işlemlerinde, müşteri analizleri, bütçe yatırım tahminleri işlemleri, yeni bir ürün tasarımı, akciğer hastalıklarının erken teşhisi, sinyal işleme, hava tahminleri analizi, robotik işlemlerde, gürültü tahmin analizlerinde, avuç içi ve retina okuma, nesne eşleştirme gibi daha pek çok uygulamada kullanılmaktadır[10,21].

3.2.1.6. Yapay Sinir Ağlarının Eğitilmesi

Yapay sinir ağları insanlar gibi zamanla eğitilirler. YSA var olan bir örnek kümesi üzerinde girdiler ve çıktılar arasındaki bağıntıların ağırlıklarının değiştirilmesi sonucu eğitilirler. Girilen girdi değerleri için transfer fonksiyonu tarafından sağlanan değerlere yanıt olarak hedeflenen çıktı değeri ile üretilen çıktı değerleri arasındaki fark en küçük düzeye düşürülünceye kadar bağıntı ağırlıklarının güncellenmesidir. YSA'da ağırlıklar sabit olmayıp aktivasyon fonksiyonu tarafından ortaya çıkarılan çıktı değerlerine göre güncelleme yapılabilir. Zamanla ağ kendisini istenen çıktıya uyarlayıp düzenleme yapabilmektedir. Bu şekilde öğrenme işlemini yapabilmektedir.

Eğitim uygun sayıda örnek veri kümesinden oluşmalıdır. Çok fazla olması performansı olumsuz etkilediği gibi az olması da yanlış sonuçlara sebep olabilmektedir. YSA' da ağa bir giriş vektörü yönlendirilerek ağdaki her eleman için girdi ve çıktı değerleri üretilir. Her bir elemanın(birimin) çıktısı sonraki ağırlıkların güncelleme işlemlerinde kullanılmaktadır. Çıktı değerlerine sahip olunduktan sonra hedeflenen çıktı ve elde edilen çıktılar karşılaştırılıp çıktı hatası bulunur. Hatalar ağa geri yayılarak, hata oranını azaltmak için birimlerdeki ağırlıklar değiştirilir. Yapay sinir ağını büyük örnek kümeleriyle eğitmek çok daha hassas sonuçlar üretmesine olanak sağlamaktadır.

Değişiklikler durumunda yapay sinir ağları tekrar eğitilebilmektedirler. Ysa yeni örnekler verildiğinde ysa çıktısı sonucu elde edilen çıktı değeri ve gerçek değer birbirine yakınsa ysa öğrenme işini başarmıştır denilebilir.

Ysa'da ağın kullanılacak topoloji yapısı, parametreler, örneklerin toplama işlemi, ağda kullanılacak başlangıç değerleri vd. belirlenmelidir. Yapay sinir ağının devamlı hata düzeyinin üstünde olması durumunda ağın başlangıç ağırlık değerleri değiştirme, ağın topolojik yapısını değiştirme, yeni örnek veri kümeleri ve örnek veri kümesinin sayısı artırılıp azaltılarak işlemler uygulanabilir.

Ağların eğitilmesi bazen çok uzun sürebilmektedir. Bunun sebebi ağın başlangıç değerlerinin çok uzun olmasından kaynaklanmaktadır. Ysa ağında farklı iterasyonlar(tekrarlar) kullanılmaktadır. İterasyon değerine göre başarımlı oranı değişebilmektedir. Bir de ağın ağırlık değerleri için bir standart olmayıp genel olarak 0 ile 1 aralığındaki değerler kullanılmaktadır.

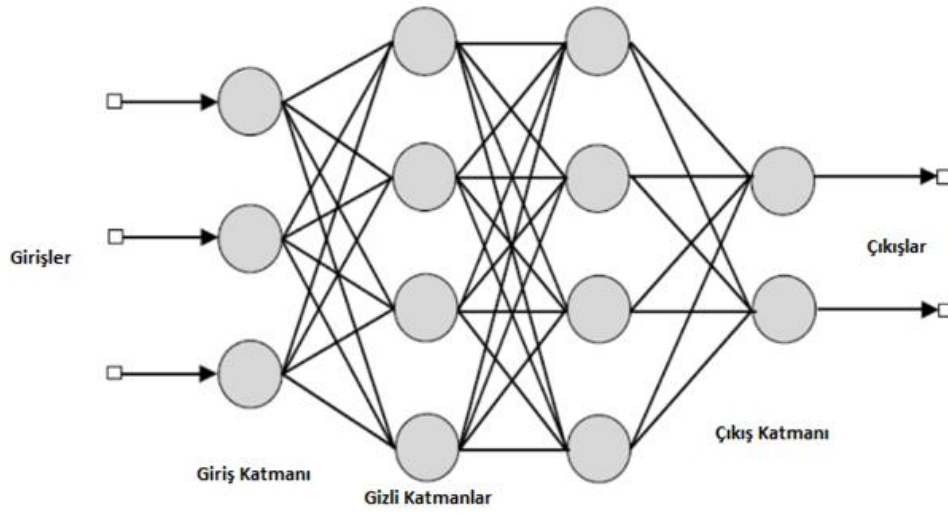
Bu çalışmada sigmoid fonksiyonu kullanıldığı için veriler [0-1] aralığında normalize edilmiştir. Bu çalışmada ileri beslemeli sinir ağı, danışmanlı öğrenme ve geriye yayılım algoritması kullanılmıştır. Bu çalışmada ağın eğitimi ve test edilmesi için toplamda 300 tane imge kullanılmıştır. Bu imgeler eşit sayıda alt veri kümesine ayrılıp bir kısmı eğitime tabi tutulmuştur. Sonra geri kalan imgeler test için sınıflandırıcıya verilmiştir. Çapraz doğrulama testi sayesinde bu döngü 10 kere yapılmıştır. Her defasında bir başarımlı oranını vermiştir. Eğitilen veri kümesine göre test kümesi bir başarımlı oranı üretir. Ayrıca 100, 500, 1000 ve 1500 şeklinde iterasyon değerleri değiştirilerek eğitim ve test etme çalışması yapılmıştır. Elde edilen sonuçlar karşılaştırılmıştır. İterasyon sayısının yanında farklı nöron sayıları ile eğitim ve test etme işlemleri yapıp farklı sonuçlar elde edilmiştir[26,30-31].

3.2.2.Yapay Sinir Ağlarının Yapılarına Göre Sınıflandırılması

Yapay sinir ağları benzerlik göstermelerine karşın belli bir modelleme ve işleme standardı yoktur. Ysa'lar ağ yapıları ve öğrenme algoritmalarına göre sınıflandırılır. Yapay sinir ağındaki nöronlar yapılarındaki bağlantı şekillerine göre ikiye ayrılır[20,25].

3.2.2.1. İleri Beslemeli Ağlar

İleri beslemeli yapay sinir ağlarında bilgi akışı tek yönlü olup giriş katmanından çıkış katmanına doğrudur. Bir katmanın çıkışından alınan bilgi akışı ağırlıklar üzerinden bir sonraki katmana iletilir. Her katman kendisinden sonraki katmanla bağlantısı olup, katmanların kendi içindeki nöronların birbirleriyle ve diğer katmanlardaki nöronlarla bir bağlantısı bulunmamaktadır. Geri yayılım öğrenme algoritması bu tür ağlarda kullanılmaktadır.



Şekil 3.13. İleri beslemeli ve çok katmanlı bir YSA gösterimi

3.2.2.2. Geri Beslemeli Ağlar

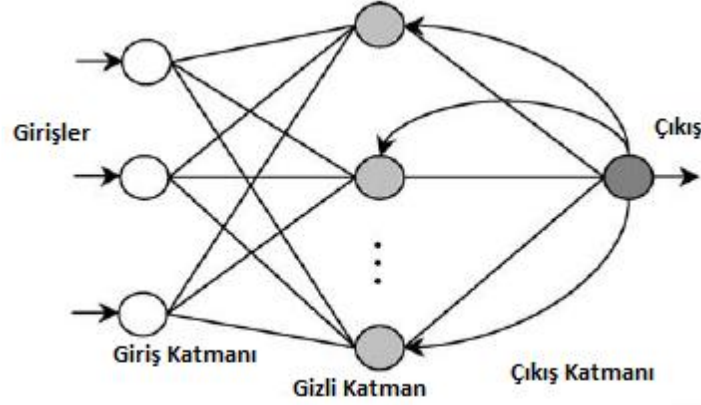
İleri beslemeli ağların tersine dinamik bir yapıya sahiptir. Burada en az bir hücre çıkışı kendi katmanındaki hücreye ya da kendisinden önceki katmanlardaki hücelere giriş verisi olarak bağlanabilmektedir. Bu tür ağlar doğrusal olmayan geri beslemeli dinamik ağ olarak hareket etmektedirler. Ağın bağlanış şekline göre farklı yapı ve davranışta yapay sinir ağı elde dileyebilir. Geri beslemeli ağlarda amaç hatayı düşürmektir. Bunun için girdi değerine karşı üretilen çıktı değeri ile beklenen çıktı değeri karşılaştırılır. Aralarındaki fark hata olarak kabul edilir. Bir sonraki tekrarda hatanın azaltılması ağırlık değerlerine dağıtılır. Hopfield SOM (Self Organizing Map), Elman ve Jordan bu ağlara örnek olarak verilebilir[15,16].

Çıktı katmanındaki n. hücresi için oluşan hata

$$En = Bn - Çn \quad (3.20)$$

Tüm hatalar toplanarak çıktı katmanı için oluşan toplam hata bulunur. Toplam hata formülü (E yi hata olarak gösterirsek);

$$\varepsilon = \frac{1}{2} \sum_m E_m^2 \quad (3.21)$$



Şekil 3.14. Geri beslemeli ağı gösterimi

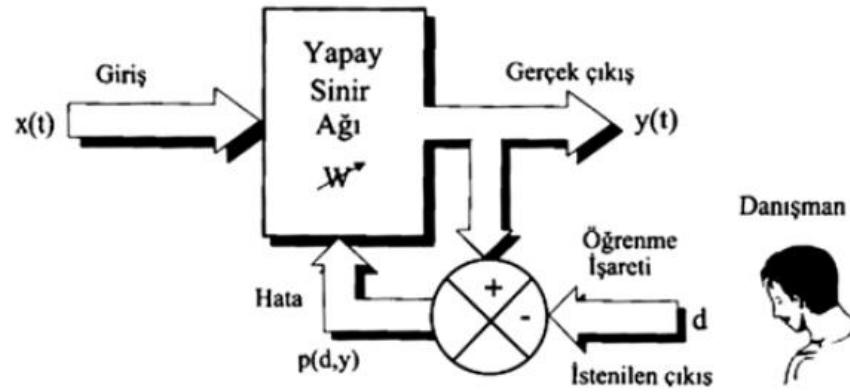
3.2.3.Yapay Sinir ağlarının Öğrenme Stratejileri

Yapay sinir ağlarında öğrenmenin amacı verilen girdi değerlerine çıktı değerlerini üretmektir. Ayrıca ağların tahmin etmedeki hata oranını en aza indirip ağırlık değerlerini elde etmektir. Yapay sinir ağlarında öğrenme eğitici, eğitcisiz ve takviyeli öğrenme olarak 3 grupta incelemek mümkündür.

3.2.3.1.Eğitici Öğrenme

Yapay sinir ağlarında eğitici öğrenmede ağa girdi değerlerinin yanında çıktı değerleri de verilir. Ağı ürettiği çıktı ile verilen çıktı değerleri karşılaştırılarak nöron başına düşen hata oranı hesaplanır. Bu hata oranına göre her nöron kendi ağırlıklarını günceller.

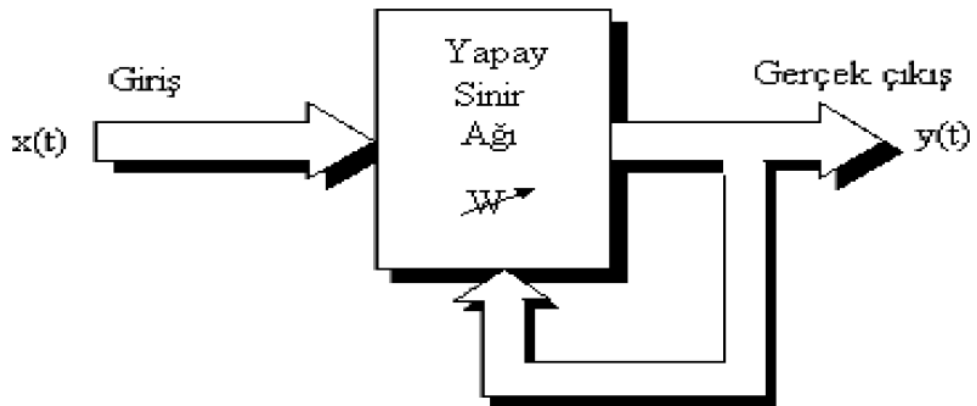
Bu öğrenmede tahmin edilen sonuç bulununcaya kadar döngüye devam edilir. Doğru tahmin oranı ağı eğitime göre başarı gösterir. Eğer ağ iyi eğitilmemişse test aşamasında da doğru sonuç vermemesi mümkündür. Bu öğrenme metoduna örnek olarak delta kuralı, rastsal öğrenme kuralı ve geri beslemeli algoritmalar örnek gösterilebilir[8,16].



Şekil 3.15. Eğitici Öğrenme Yapısı[14]

3.2.3.2. Eğitici Öğrenme

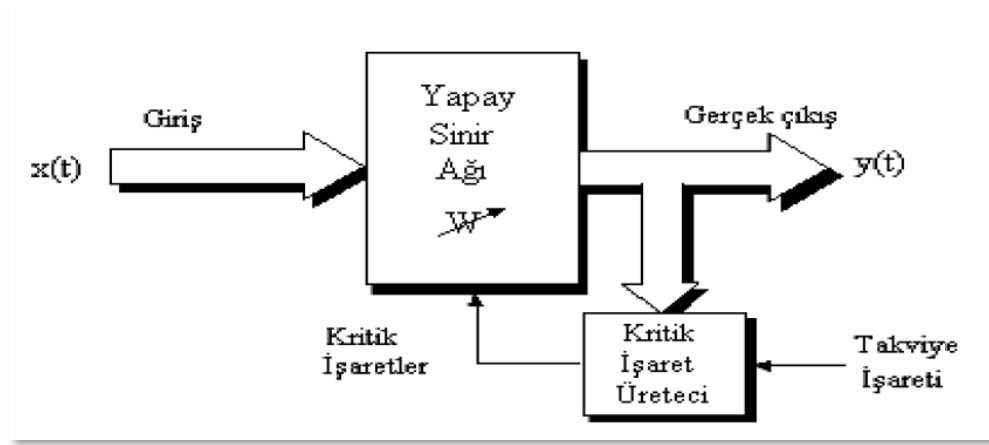
Eğitici öğrenmede ağı yalnızca giriş değerleri verilip çıkış değerlerinin verilmediği bir öğrenme yöntemidir. Bu tür ağlar girilen örnekleri alıp belli ölçütlere göre sınıflandırma yapıp ve giriş bilgilerine göre ağırlık değerlerini düzenlerler. Performanslarını herhangi bir dış etkiye ihtiyaç duymadan içeriden gözlemleyebiliyorlar. Ağ bağlantı ağırlıkları aynı nitelikte olan dokuları ayırabilecek şekilde ayırıp öğrenme işlemini bitirmiş olur. Hebbian öğrenme, Grossberg öğrenme ve Kohonen'in özörgütlemeli harita ağı bu tür ağa örnek olarak gösterilebilir[8,16,24].



Şekil 3.16. Eğitici öğrenme blok diyagramı[14]

3.2.3.3. Takviyeli Öğrenme

Bu yaklaşımda yapay sinir ağına bir çıkış verilmez. Yalnız elde edilen çıkışın verilen girişe karşılık iyilik veya doğruluğunu değerlendiren bir ölçüt kullanılmaktadır. Yani her tekrar sonucunda elde ettiği sonucun durumu hakkında bilgi verir. Bu bilgilere göre ağ kendini yeniden tanzim eder. Ağ tamamen deneme yanılma yoluyla eğitilmektedir. Ağa verilen bir örnek veri aracılığıyla ağ hem öğrenme hem de sonuç çıkararak işlem yapmaya devam eder. Hinton ve Sejnowskinin geliştirdiği Boltzman Kuralı ya da genetik algoritmalar bu ağa örnek olarak verilebilir[7-8].



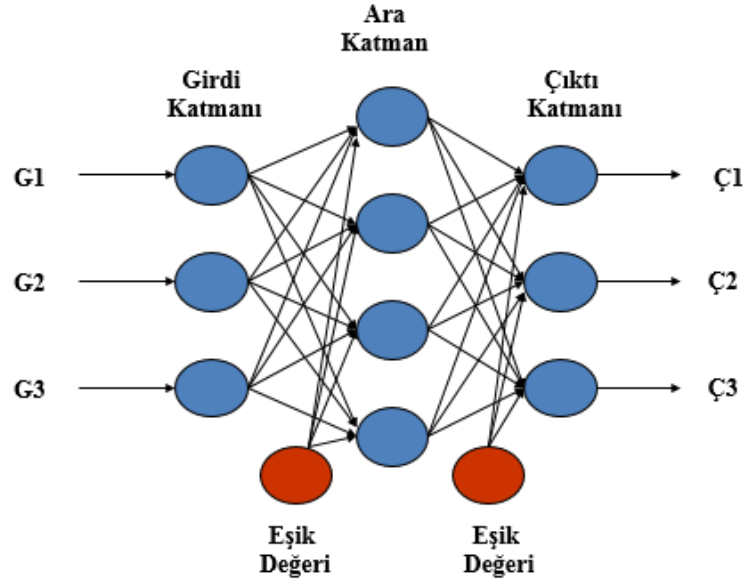
Şekil 3.17. Takviyeli Öğrenme Blok Diyagramı[14]

3.2.4. Çok Katmanlı Algılayıcı (ÇKA)

Ağ yapıları tek katmanlı yapay sinir ağları, çok katmanlı yapay sinir ağları ve döngülü yapay sinir ağları şeklinde 3 farklı grupta incelenir. ÇKA, doğrusal ilişkinin olmadığı durumda kullanılan bir yöntemdir. Bu modele hata yayma modeli veya geriye yayılım modeli denilmektedir. Amaç ağın hedeflenen çıktısı ile üretilen çıktı arasındaki hata farkını en aza düşürmektir. ÇKA bilhassa tanıma, sınıflandırma ve genelleme yapılacak problemlerde çözüm aracı olarak kullanılmaktadır.

ÇKA ağ modelinde bir giriş, bir ya da daha fazla ara katman ve bir çıkış katmanından oluşmaktadır. Her katman kendisinden sonraki katman ile bağlantılıdır. Giriş katmanında bilgi işleme yapılmayıp olduğu gibi kendisinden sonraki katmana gönderilir. Ara katman ise giriş katmanından gönderilen bilgiler işlenerek çıkış katmanına gönderilir. Her katmandaki işlem elemanı kendisinden sonraki katmanlardaki tüm işlem elemanları ile

bağlıdır. Çıkış katmanında ise giriş katmanındaki girdilere karşılık bir çıktı değeri üretilir[7,9,48].



Şekil 3.18. Çok katmanlı ağ modeli

ÇKA ağı eğitici öğrenme stratejisini kullanmaktadır. Ağa örnekler ve hedef çıktılar verildiği için genelleme yaparak bir çözüm uzayı üretir. Bu çözüm uzayına göre benzer yeni örnekler verildiğinde bu çözüm uzayı sonuçlar üretir.

ÇKA ağının öğrenme kuralı en küçük kareler yöntemine dayalı Delta Öğrenme Kuralının genelleştirilmiş halidir. Bundan dolayı bu öğrenme kuralına Genelleştirilmiş Delta Kuralı da denilmektedir. Ağın öğrenebilmesi için örneklerden oluşan eğitim setine gereksinim duyulmaktadır. Bu set içinde bütün örnekler için girdi ve o girdiler için hedeflenen çıktılar belirlenir. Girdi katmanındaki k. Süreç(proses) elemanının çıktısı ζ_k^i şu şekilde hesaplanır.

$$\zeta_k^i = G_k \quad (3.20)$$

Ara katmandaki süreç elemanlarına gelen net girdi (NET_j^a) değeri ise şu şekilde hesaplanır.

$$NET_j^a = \sum_{k=1}^n A_{kj} \zeta_k^i \quad (3.21)$$

Burada A_{kj} k. girdi katmanı elemanını j. ara katman elemanına bağlayan bağlantının ağırlık değerini göstermektedir. j. ara katman elemanın çıktısını göstermektedir. Uygulamada genellikle sigmoid fonksiyonu kullanılmaktadır. Türevi alınabilir bir

fonksiyon amaçlandığı için Sigmoid fonksiyonu kullanılır. Sigmoid fonksiyonu kullanıldığında oluşan çıktı (3.22) denklemde gösterilmiştir,

$$C_j^a = \frac{1}{1 + e^{-(NET_j^a + \beta_j^a)}} \quad (3.22)$$

Burada β_j^a , ara katmanda bulunan j. elemanına bağlanan eşik değer elemanın ağırlığını göstermektedir. Eşik değer ünitesinin çıktısı bir olup sabittir. Ağırlık değeri ise sigmoid fonksiyonunun yönlendirmesini belirlemek üzere konulmuştur. Eğitim esnasında ağ bu değeri kendisi belirlemektedir.

Ağa verilen girdi ve üretilen çıktı değerleri karşılaştırılarak aralarındaki fark hata kabul edilir. Amaç bu hatanın düşürülmesi olduğu için geriye hesaplamada bu hata ağı ağırlık değerlerine dağıtılarak bir sonraki tekrarda hatanın azaltılması sağlanır. Çıktı katmanındaki m. süreç(işlem) elemanı için oluşan hata (E_m).

$$E_m = B_m - C_m \quad (3.23)$$

Burada $b_1, b_2, \dots, b_m$ beklenen değer, $C_1, C_2, \dots, C_m$ çıktı değeri E_m ise hata değeridir.

Bir işlem elemanı için oluşan hata buysa çıktı katmanı için oluşan toplam hatayı bulmak için bütün hata değerlerinin toplanması gerekir. Hata değerlerinin bazıları negatif olacağı için toplamın sıfıra eşit olmasını önlemek amacıyla ağırlıkların kareleri alınarak sonucun karekökü alınır. Toplam hata şu formül ile elde edilir.

$$TH = \frac{1}{2} \sum E_m^2 \quad (3.24)$$

Toplam hatayı en aza indirmek için bu hatanın kendisine neden olan süreç elemanlarına dağıtılıp ağırlıklarının değiştirilmesi gerekir. Bu hesaplamanın ardından hem ara katman ile çıktı katmanı arasındaki ağırlıklar, hem de ara katmanlar arası veya ara katman ile girdi katmanı arasındaki ağırlıklar değiştirilir. Böylece ağı ağırlıklarının tamamı değişmiş olacaktır. Ara katman ile çıktı katman arasındaki ağırlığın değişimi (3.25) formülünde gösterilmiştir

$$\Delta A_{jm}^a(t) = \lambda \delta_m C_j^a + \alpha \Delta A_{jm}^a(t-1) \quad (3.25)$$

Burada λ öğrenme katsayısını, α momentum katsayısını, δ_m m. çıktı ünitesinin hatasını göstermektedir. Şu şekilde hesaplanır.

$$\delta_m = f'(NET).E_m \quad (3.26)$$

Buradaki f' (NET) aktivasyon fonksiyonunun türevidir. Sigmoid fonksiyonunun kullanılması durumunda;

$$\delta_m = C_m(1 - C_m).E_m \quad (3.27)$$

olacaktır. Değişim miktarı hesaplandıktan sonra ağırlıkların t. tekrarındaki yeni değerleri şöyle olacaktır.

$$A_{jm}^a(t) = A_{jm}^a(t-1) + \Delta A_{jm}^a(t) \quad (3.28)$$

Ara katmanlar arası veya ara katman ile girdi katmanı arasındaki ağırlık değişimleri de formüllerde gösterildiği gibi olur.

$$\Delta A_{kj}^i(t) = \lambda \delta_j^a C_k^i + \alpha \Delta A_{kj}^i(t-1) \quad (3.29)$$

olacaktır. Buradaki hata terimi δ_a ise şöyle hesaplanacaktır.

$$\delta_j^a = f'(NET) \sum_m \delta_m A_{jm}^a \quad (3.30)$$

Aktivasyon fonksiyonu olarak sigmoid fonksiyonu düşünülürse bu hata değeri şu şekilde hesaplanacaktır.

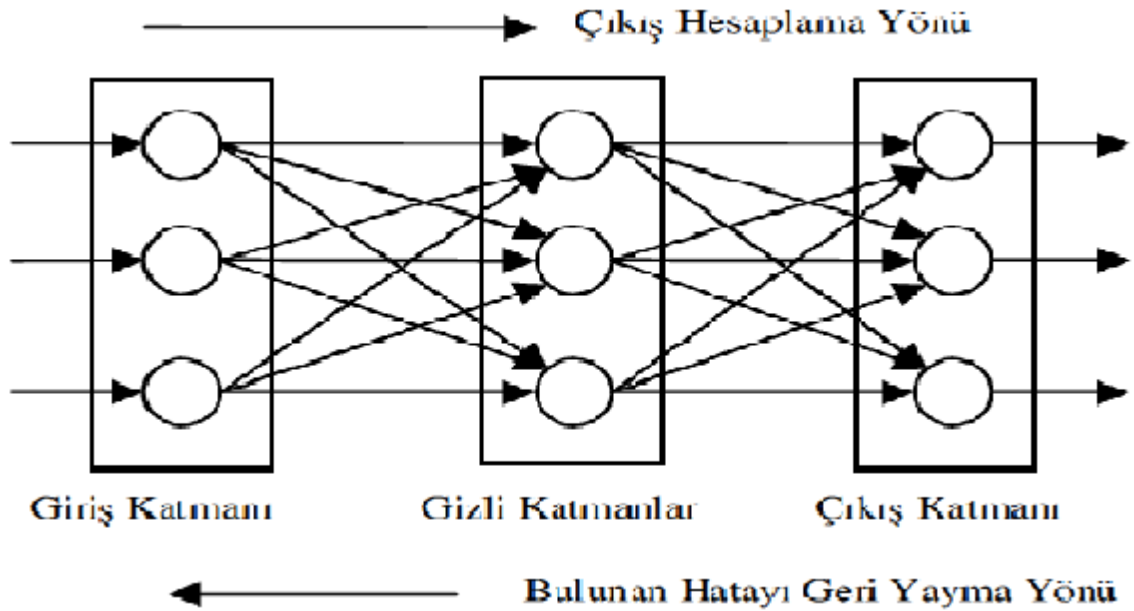
$$\delta_j^a = C_j^a(1 - C_j^a) \sum_m \delta_m A_{jm}^a \quad (3.31)$$

Böylece ağırlıklarının hepsi değiştirilmiş olacaktır. Bir tekrarda hem ileri hem de geriye hesaplamaları yapılarak tamamlanmış olacaktır. İkinci bir örnek verilerek sonraki tekrara(iterasyona) başlanır ve aynı işlemler öğrenme bitinceye kadar devam eder[1,11].

ÇKA ağı parametreleri için de ağı öğrenme katsayısı, proses elemanlarının toplama ve aktivasyon fonksiyonları, momentum katsayısı gibi parametreler belirlenmelidir.

3.2.4.1. Geriye Yayılım (Back Propagation) Öğrenme Algoritması

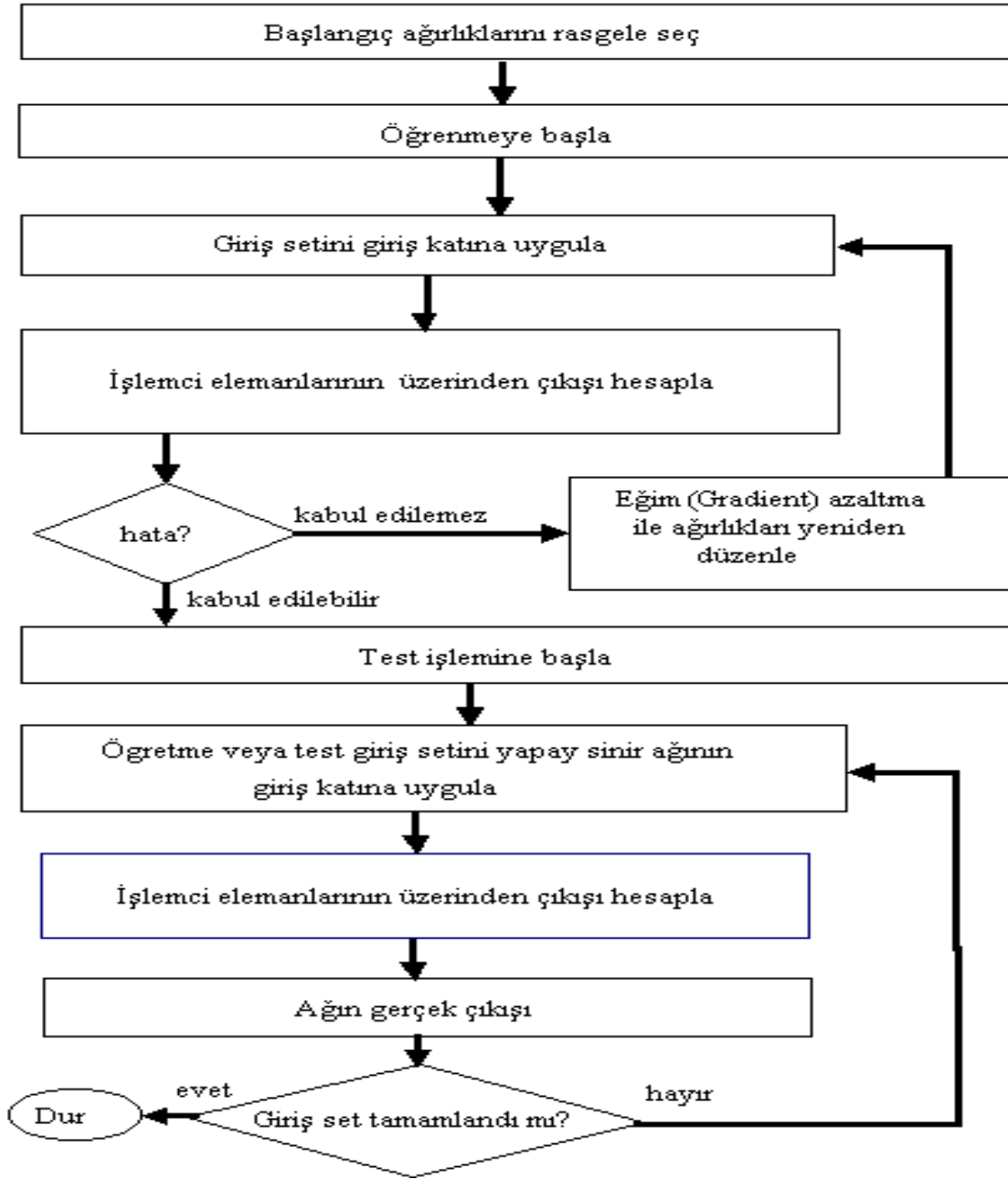
Sinir ağlarında öğrenme girdi ve çıktı kümesi için test edilen ağı, istenilen çıktıyı verecek ağırlık kümesinin belirlenmesidir. Bu ağırlık vektörü istenen çıktı ile ağ çıktısı arasındaki fark en az oluncaya kadar ayarlanıp bulunur. Amaca göre öğrenme grupları için farklı öğrenme algoritmaları geliştirilmiştir. Çok katmanlı algılayıcıların yapısında kullanılan geriye yayılım öğrenme algoritması esasen bir danışmanlı öğrenme algoritmasıdır. Çok katmanlı ağlar giriş, çıkış ve en az bir ara(saklı) katmandan oluşmaktadır. Katmanların kendi içerisindeki nöronlar arasında ve geriye doğru (bir katmandan önceki katmana) bağlantı yoktur.



Şekil 3.19. Bir geri-yayılma ağ gösterimi

Anlaşılması kolay ve matematiksel olarak ispatlanabilir olmasından dolayı uygulamalarda en çok kullanılan öğrenme algoritmasıdır. Bu algoritma, hataları çıkıştan girişe doğru azaltmaya çalışır. Bu işlemi yaparken her katmandaki nöronların ağırlık değerleri güncellenir. Burada üretilen çıktı ile hedeflenen çıktı arasındaki farkı hata olarak geriye doğru dağıtma işlemi gerçekleştirilir.

Geri yayılım algoritmasının MLP' leri eğitmede en çok kullanılan temel bir algoritma olup akış şeması Şekil 3.20'de verilmiştir. Eğitim ve test işlemi bu akışa göre yapılır[2,17,18].



Şekil 3.20. Çok katmanlı bir algılayıcının geri yayılım akış şeması[20,26]

3.2.5. Çapraz Doğrulama

Sınıflandırma çalışmalarında test sonuçlarının başarımlarını doğru hesaplamak önemlidir. Bu nedenle çalışmaların çoğunda çapraz doğrulama yöntemi kullanılmaktadır. Bu yöntem sayesinde test sonuçlarının başarımlarının güvenilirliği artar. Eğitim ve test veri kümesinin belirlenmesinde çoğu zaman rastgele yöntemler kullanılmaktadır. Bu tür yöntemlerde veri setinin bir kısmı eğitim bir kısmı test için kullanılmaktadır. Eğitim ve test kümesinde kullanılan örnek imgelerin benzerlik oranına göre farklı sonuçlar elde edilir. Bu

tür yöntemler sonucu elde edilen test sonuçlarının güvenilirlik sorunu vardır. Çapraz doğrulama testi sayesinde bu güvenilirlik sorunu ortadan kaldırılmaya çalışılmıştır[51-52].

Çapraz doğrulama testinde veri kümesi k kadar ve eşit sayıda alt veri kümelerine ayrılır. $k_1, k_2, k_3, \dots, k_n$ şeklinde parçalara ayrılabilir. Bu veri kümelerin her biri hem eğitim hem de test için kullanılmaktadır. Test için her defasında bir veri kümesi alınır geriye kalan veri grupları eğitim için kullanılmaktadır. Sınıflandırıcı çalıştırılıp bir sonuç elde edilir. Bu defa diğer bir veri kümesi test için seçilip geri kalan veri kümelerin tümü eğitim için kullanılıp sınıflandırıcıyı çalıştırıp ikinci bir sonuç elde edilir. Bu şekilde bu tüm k_n alt kümeleri için çalıştırıp bulunan tüm sonuçlarının toplamının k_n ya bölünmesi sonucu ortalama sonuç bulunur. Literatürde genelde 10 katlı çapraz doğrulama testinin daha uygun olduğu belirtilmektedir[24,33,49-50].

Bu çalışmada 10 k-katlı çapraz doğrulama testi kullanılmış olup veri kümesi 10 tane alt veri kümesine bölünmüştür. Çalışma boyunca bu veri kümelerin her biri hem eğitim hem de test için kullanıldı. Hangi alt veri kümesinin test için alınacağı önemli değildir. Her defasında veri kümelerden biri test geri kalan veri grupları da eğitim için kullanılıp sınıflandırıcıya yönlendirilerek sonuç elde edildi. Bu işlem 10 defa tekrar edilip bulunan sonuçlar toplanıp 10'a bölünerek ortalama sonuç bulunmuş olur.

4.UYGULAMA

Bu bölümde cilt hastalıklarının teşhisi için yapılan uygulama anlatılmıştır. Bu uygulamada 3 farklı cilt hastalığı baz alınmış olup bu hastalık üzerinde yapılan işlemler açıklanmıştır. Bu bölümde sırasıyla yapılan ön işleme ve uygulama işlemleri anlatılmıştır.

Cilt hastalıklarına ait 3 farklı hastalıktan her birinden 100 imge toplanarak toplam 300 tane imge elde edilip GWT metodu ile öz nitelik vektörü oluşturulmuştur. Öznitelik vektörü çıkarılan imgeler yapay sinir ağları ve çapraz doğrulama(cross validation) testinin teknikleri kullanılarak sınıflandırma yapılmaya çalışılmıştır. Kullanılan cilt hastalığına ait imgeler uluslar arası bir veritabanında elde edilmiş olup Cizre devlet hastahanesinin cildiye doktorlarından ve Dicle Üniversitesinin Eğitim Araştırma Hastanesindeki İlgili alanda çalışan dermatologlardan yardım alınarak imgelerin ait olduğu cilt hastalığı grubu üzerine değerlendirme yapılmıştır.

JPEG formatında kullanılan resimler MATLAB programı ile 2 boyutlu gri tonlu resimlere dönüştürülmüştür. Ayrıca uygulamada kullanılan tüm resimlerin boyutu 128x128 piksel olarak ayarlanmıştır. Daha sonra bu resimler bicubic interpolasyon bilinear yöntemi sayesinde 16x16 'lık piksellik resimlere dönüşüm ve histogram eşitleme işlemleri yapılmıştır. Şekil 4.1' de Akne cilt hastalığına, Şekil 4.2' de Sedef cilt hastalığına ve Şekil 4.3'te ise Mantar cilt hastalığına ait örnek imgeler gösterilmiştir.



Şekil 4.1. Akne cilt hastalığının gösterimi



Şekil 4.2. Sedef cilt hastalığının gösterimi



Şekil 4.3. Mantar cilt hastalığının gösterimi

Bu bölümde her bir hastalık için 3. bölümde anlatılan Gabor dalgacık dönüşümü ile Gabor öznitelik vektörleri oluşturulmuştur. Bu öznitelik vektörleri her bir imgeye ait 8 dalgacık matrislerinin istatistiksel değişkenlerinin bir vektör oluşturacak şekilde ardışık eklenmesi sonucu oluşmuştur. GWT matrisinin standart sapma, entropi ve ortalama değerleri hesap edilmiştir. Bu elde edilen 3 değer ve 8 dalgacık dönüşümünün ardışık eklenmesi sonucu her resim için 24 boyutunda öznitelik vektörü elde edilmiştir. Her hastalıktan 100'er resimden oluşan veritabanı düşünüldüğünde 100x24'lük öznitelik matrisine 10 aşamalı çapraz doğrulama(10 fold cross validation) metodu ile sınıflandırma uygulanmaya çalışılmıştır.

Sinir ağılarının, hastalıkları sınıflandırmada hangi parametrelerinin etkili olduğunu saptamaya çalışmak amacıyla 3 faktör ile denemeler yapılarak doğru sınıflandırma oranları ve Confusion matrisler elde edildi.

Bu üç faktör

- 1- Yapay sinir ağının öğrenme algoritmaları.
 - a. Trainscg
 - b. Trainrp
 - c. Trainlm
 - d. Trainbfg
- 2- Ara katmanda bulunan nöron sayıları
 - a. 25 nöron
 - b. 50 nöron
- 3- İterasyon sayıları
 - a. 100 iterasyon
 - b. 500 iterasyon
 - c. 1000 iterasyon
 - d. 1500 iterasyon

Bu faktörlerden biri olan öğrenme algoritmaları hakkında genel bilgiler verilmiştir. Herhangi bir problemi çözebilmek için hangi öğrenme algoritmasının en hızlı olacağını bilmek zordur. Çünkü, öğrenme algoritmasının seçimi birçok faktöre bağlıdır. Tez çalışması kapsamında kullanılan ÇKSA’nda 4 çeşit öğrenme fonksiyonu kullanılmıştır. Bu öğrenme algoritmalarının hepsinin genel amacı hatayı minimize edecek noktayı hata fonksiyonu üzerinden bulmaktır.

Ölçekli Eşlenik Gradyan Algoritması (Scaled Conjugated Gradient)

İleri beslemeli yapay sinir ağları için çeşitli adaptif öğrenme algoritmaları Hinton tarafından 1989 yılında keşfedilmiştir [34]. Bu algoritmaların çoğu optimizasyon teorisinde iyi bilinen gradyan azalan algoritmalarıdır. Hinton tarafından elde edilen bu algoritmalar düşük yakınsama oranına ve kullanıcı tarafından saptanan fakat herhangi bir teorik temeli açıklanmamış parametrelere bağımlıdır. Bunlardan bir tanesi standart geri yayılım (Back Propagation) algoritmasıdır. Bu algoritma geniş ölçekli problemlerde kötü performans göstermekte ve öğrenme oranı, momentum katsayısı gibi kullanıcının karar

verdiği parametrelere göre işlem yapmaktadır. Problemin bu algoritmalarla başarılı bir şekilde çözülmesi, kullanıcıların doğru parametreleri kullanmasına bağlıdır. Bu dezavantajların giderilmesi amacıyla Martin Moller,[35] de eğitici bir öğrenme algoritması sunmuştur. Hinton'un çalışmasında SCG algoritması geniş ölçekli problemlerde BP(Back Propagation) ve Broyden–Fletcher–Goldfarb (BFG) algoritmalarından başarılı olduğu ve kullanıcının belirlediği kritik değerlere bağlı olmadığı ortaya konulmuştur.

Broyden–Fletcher–Goldfarb Algoritması (trainbfg)

Battiti ve Massuli, 1990 yılında optimizasyon literatürüne BFGS diye geçen bir algoritma geliştirdiler[36]. Bu algoritmanın esasını çizgi arama ile birleştirilmiş eşlenik yönler oluşturmuştur. BFG metodu Newton Metodu yaklaşımını esas alarak hatayı minimize etmeye çalışır. BFG metodu ikinci dereceden türevlenebilir bir fonksiyonda hareketsiz noktaları arayan bir tepe tırmanma optimizasyon tekniğidir. Newton metodu ve BFG metodu, kuadratik Taylor açılımına sahip fonksiyon olmadan optimum noktaya yaklaşılmasını garanti etmemektedir. Bu metodların ikisi de hata fonksiyonunun birinci ve ikinci türevlerine ihtiyaç duymaktadır. BFG, düzgün olmayan(non-smooth) optimizasyon problemlerinde yüksek performans göstermektedir[37].

Kendini Toparlayabilen Geri Yayımlı Öğrenme Algoritması (Resilient Backpropagation-Trainrp)

Yapay sinir ağlarında sezgisel olarak öğrenen eğitici öğrenme algoritmalarından biri RP dir. Martin Riedmiller ve Heinrich Braun tarafından 1992 yılında geliştirilmiştir. Bu algoritmada ağırlıklar hata fonksiyonunun davranışına göre güncellenir. RP algoritmasının avantajı ağırlıkların değişim miktarlarını kolayca hesaplayabilmesi ve bu durumun doğal olarak sonuca hızlı gitmesini sağlamasıdır. Manhattan öğrenme metodunun gelişmiş versiyonudur. Kompleks problemlerde RP algoritmasının performansı düşmektedir[38].

Levenberg-Marquardt Algoritması (Trainlm)

Kenneth Levenberg tarafından ilk olarak 1944 yılında geliştirilmiş bir algoritmadır. En lineer olmayan en küçük kareler problemini çözmekte kullanılan yaklaşımı esas almaktadır. Bu algoritma hata fonksiyonuna eğri uydurma problemi gibi yaklaşıp hatayı

minimize edecek değeri tahminlemeye çalışır. LM Algoritması Gauss Newton Algoritması ve gradyan azalan algoritma sonuçlarının interpolasyonunu yaparak hatanın minimum olduğu noktayı bulmaya çalışmaktadır[39]. Levenberg bu algoritmayı daha sonra tanıştığı istatistikçi Donald Marquardt ile 1963 te tekrar geliştirmiştir. Bu algoritma ile lineer olmayan modellerde interpolasyon maksimum komşuluk esasına dayanarak gerçekleştirilmiştir[40].

Çapraz doğrulamadan(Cross Validation) elde edilen doğru sınıflandırma oranları Tablo 4.1 de sunulmuştur. Görüldüğü gibi yapılan denemelerde en yüksek başarıım 50 nöronlu ara katmana sahip, trainscg öğrenme algoritma ile 1500 iterasyonda bulunmuştur.

Tablo 4.1. Çapraz doğrulama doğru sınıflandırma oranı gösterimi

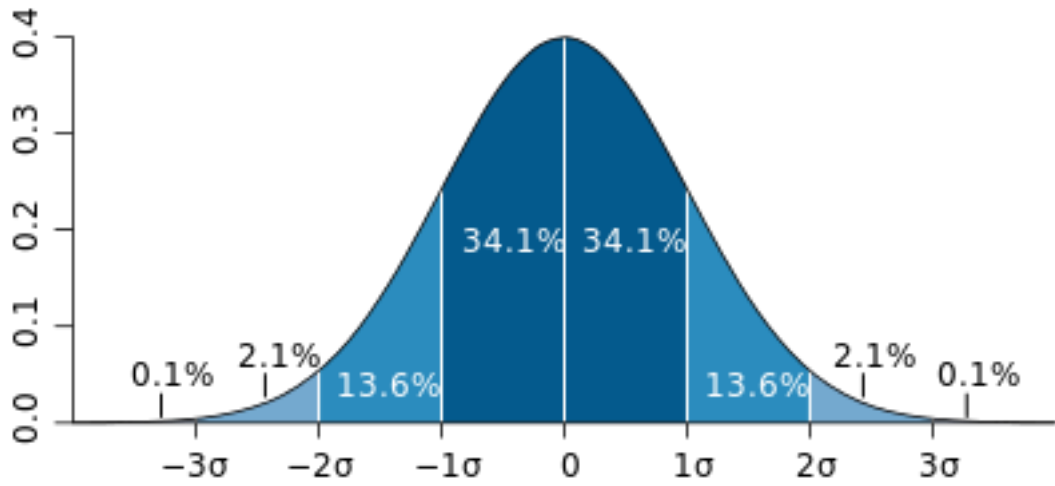
	trainscg		trainrp		trainlm		trainbfg	
epoch	n=25	n=50	n=25	n=50	n=25	n=50	n=25	n=50
100	%72	%81	%70	%74	%76	%79	%69	%76
500	%80	%80	%78	%83	%78	%79	%65	%71
1000	%82	%83	%81	%84	%73	%78	%64	%75
1500	%80	%84,3	%83	%83	%70	%77	%56	%65

Hastalıkların yapay sinir ağları ile sınıflandırma problemine farklı bir açıdan bakabilmek için elde edilen çapraz tablo sonuçları ek-1 de verilmiştir. Ekteki tablodan bazı grafiksel gösterimlerle problem hakkında yorum yapılmaya çalışılmıştır.

Tablo 4.2. Doğru Sınıflandırma oranları tablosundan hesaplanan istatistikler

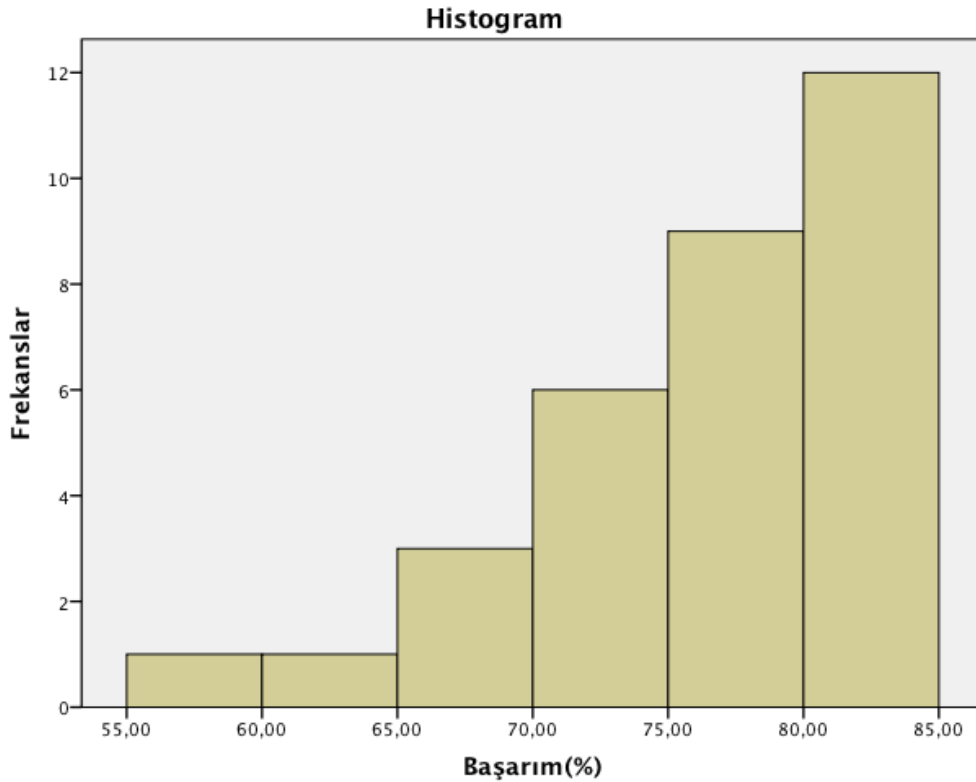
Tanımlayıcı İstatistikler					
	N	Minimum	Maximum	Ortalama	Std. Sapma
Başarım(%)	32	56,00	84,30	75,9781	6,81525

Tablo 4.2. de çalışma boyunca yapılan denemelerden elde edilen ortalama başarıım gösterilmektedir. Normal dağılım gösteren bir örneklemin yüzdelik dilimleri şekil 4.4' te gösterilmiştir.



Şekil 4.4. Her bandın bir birimlik standart sapmaya eşit olduğu normal dağılım eğrisi

Bu verilere bakılarak, bu özneliliklerden elde edilecek yapay sinir ağı sınıflandırıcıların yaklaşık %68'inin başarımları oranı $(75,97-6,81)$ ile $(75,97+6,81)$ arasında değişecektir(%69,%82,78).



Şekil 4.5. Başarım oranlarının histogramı

Şekil 4.5. te görüldüğü gibi başarımların histogramı alındığında sınıflandırma probleminde yapılan denemelerin çoğunluğunun yüzde 80 ile 85 arasında bir doğru sınıflandırma

oranını sağladığı görülmektedir.

4.1. Öğrenme Algoritmaları Açısından Sınıflandırma Sonuçları Analizi

Tablo 4.3.Öğrenme algoritmalarının doğru sınıflandırma oranlarının istatistikler

Öğrenme Algoritmaları	İstatistikler	Std. Hata
trainscg	Ortalama	80,2875
	Standart Sapma	1,30814
	Minimum	3,69998
	Maksimum	72,00
trainrp	Ortalama	84,30
	Standart Sapma	79,5000
	Minimum	5,09902
	Maksimum	70,00
trainlm	Ortalama	84,00
	Standart Sapma	76,2500
	Minimum	3,19598
	Maksimum	70,00
trainbfg	Ortalama	79,00
	Standart Sapma	67,8750
	Minimum	6,72814
	Maksimum	56,00

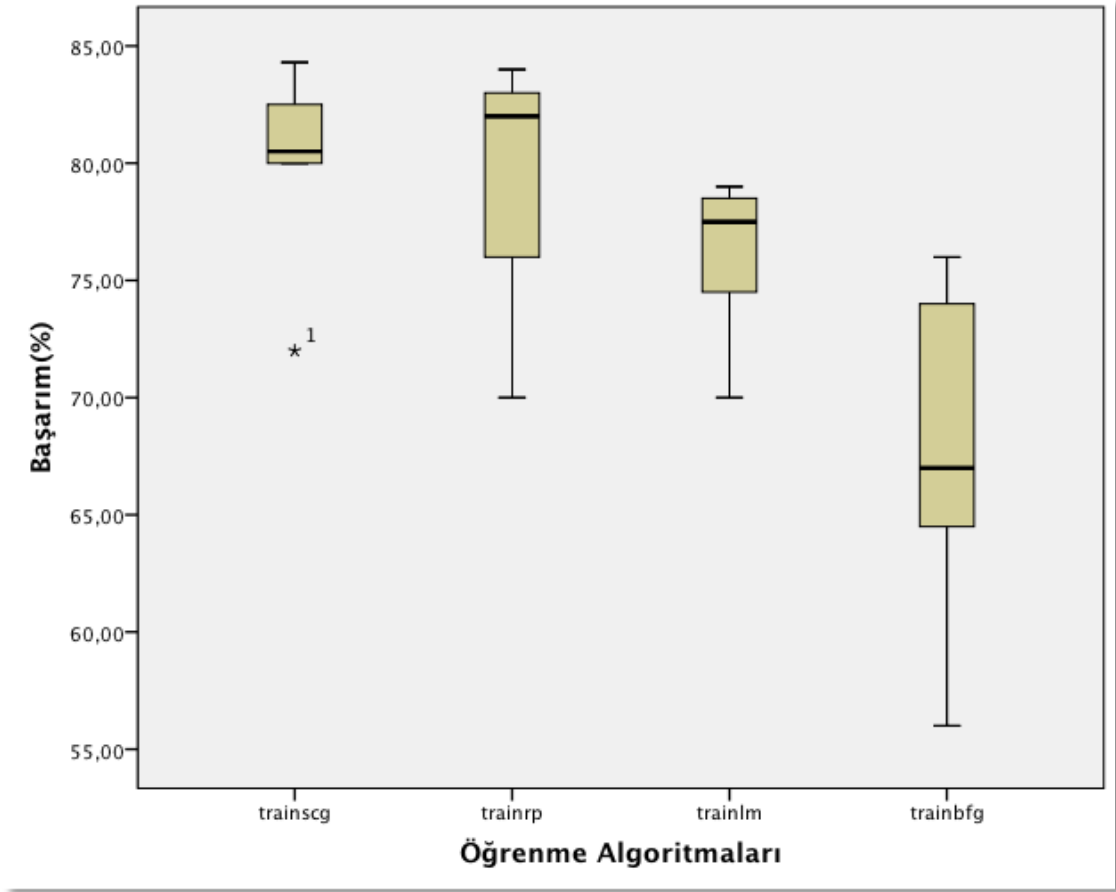
Tablo 4.3. te görüldüğü gibi trainscg öğrenme algoritmasını kullanan YSA sınıflandırıcının öznelilikleri sınıflandırmada elde ettiği doğru sınıflandırma oranları bakımından ortalaması en yüksektir.

Trainbfg öğrenme algoritmasının standart hatasının yüksek olduğuna bakılarak, trainbfg algoritmasını kullanan sınıflandırıcıların iterasyon veya nöron sayılarından etkilendiği yorumu yapılabilir.

Trainlm fonksiyonu her ne kadar trainscg öğrenme fonksiyonundan daha düşük başarımlarına sahip olsa da en düşük varyasyona ve standart hata ya sahiptir. Bu ise çalışmada gerçekleştirilen denemelere bakarak, trainlm öğrenme algoritmasının nöron ve iterasyon sayısı değişikliklerinden daha az etkilendiği sonucuna ulaştırmaktadır.

Şekil 4.6. daki kutu grafik incelendiğinde en yüksek başarıma sahip olan trainscg fonksiyonunun bir tane aykırı değer içerdiği görülmektedir. Bu 25 nöron ve 100 iterasyonlu yapıya sahip olan %72 başarımlı değeridir. Ortalama değer bu tarz aykırı değerlerden etkilendiği düşünüldüğünde bu aykırı değer denemeden çıkarılması

durumunda trainscg algoritmasının başarımı, tablo 4.3'teki değerinden daha yüksek olacaktır.



Şekil 4.6. YSA sınıflandırıcılarının öğrenme algoritmalarına göre başarımlarının kutu grafikleri

Şekil 4.6.'daki grafiğe bakılarak Trainbfg ve trainrp algoritmaları geniş bir aralıkta sonuç vermiştir. Bu algoritmaların, Gabor filtreleri ile elde edilen öznitelikleri kullanarak hastalık sınıflandırma probleminde kullanılması uygun değildir.

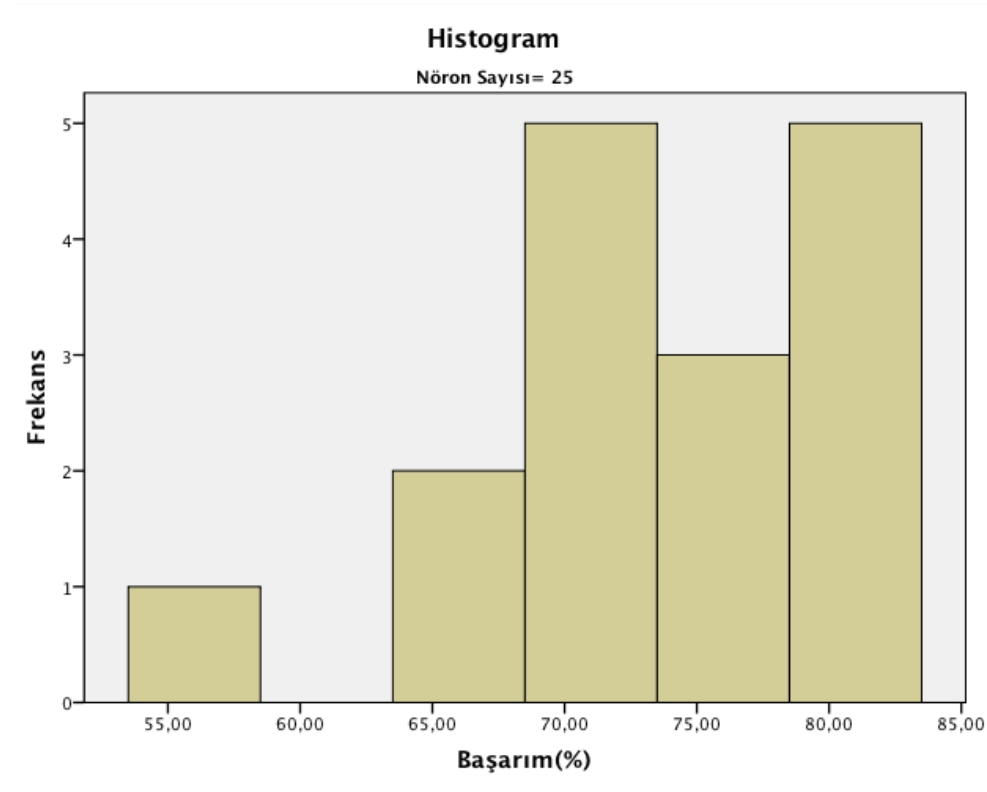
Yüksek başarımı ve düşük varyasyonu sebebiyle trainscg algoritmasının başarılı olduğu söylenebilir.

4.2. Nöron Sayıları Açısından Sınıflandırma Sonuçları Analizi

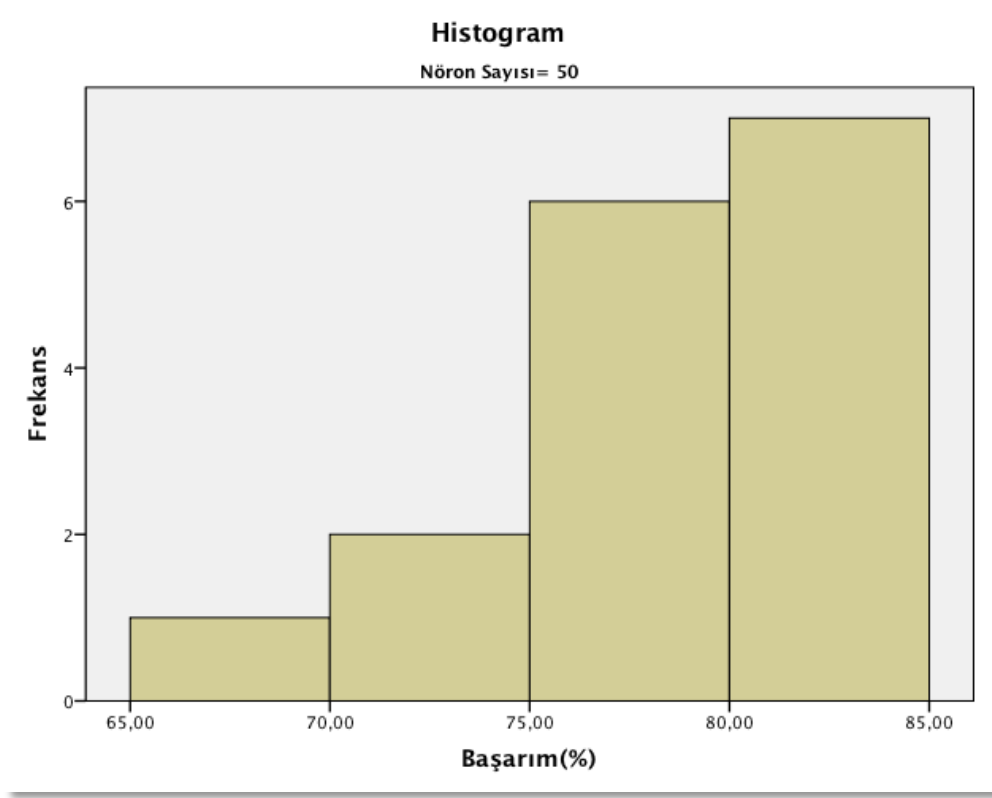
Tablo 4.4. Nöron sayılarına göre YSA sınıflandırıcılarının başarımlar istatistikleri

Nöron Sayıları		İstatistikler	Std. Hata
25 Nöron	Ortalama	73,5625	1,89949
	Standart Sapma	7,59797	
	Minimum	56,00	
	Maksimum	83,00	
50 Nöron	Ortalama	78,3938	1,27019
	Standart Sapma	5,08074	
	Minimum	65,00	
	Maksimum	84,30	

Tablo 4.4. ten nöron sayısı 25'ten 50 ye çıktığında başarımların ortalamalarının arttığı görülüyor. Bu sonucun, 50 nöronlu sınıflandırıcılardan elde edilen başarımların yüksek, standart hatasının düşük olmasına bakarak doğru olduğu söylenebilir.

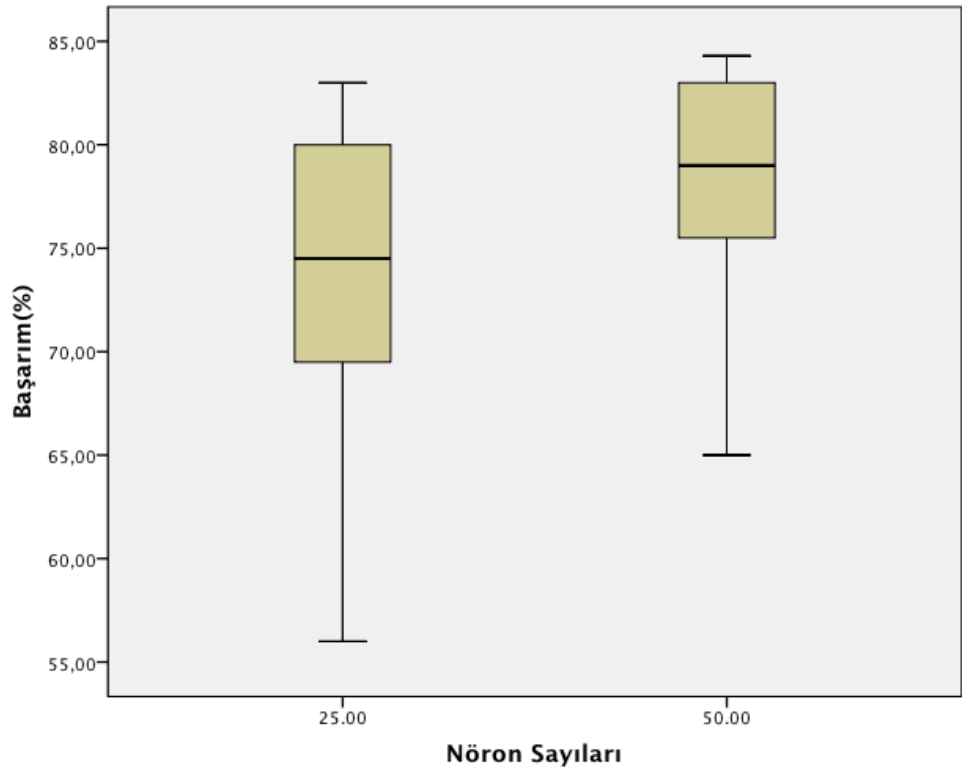


Şekil 4.7. Ara katmanında 25 nöron bulunan YSA sınıflandırıcıların başarımlar oranları histogramı



Şekil 4.8. Ara katmanında 50 nöron bulunan YSA sınıflandırıcıların başarımları histogramı

Şekil 4.7.de ve Şekil 4.8.'de, farklı öğrenme algoritmaları ve farklı iterasyon sayıları ile yapılan denemelerde YSA sınıflandırıcının ara katmanında bulunan nöron sayılarından elde edilen başarımların frekans dağılımları gösterilmiştir. Her iki nöron yapısına sahip YSA sınıflandırıcılardan 16 şar adet kullanılmıştır.



Şekil 4.9. YSA sınıflandırıcılarının nöron sayılarına göre başarımlarının kutu grafikleri

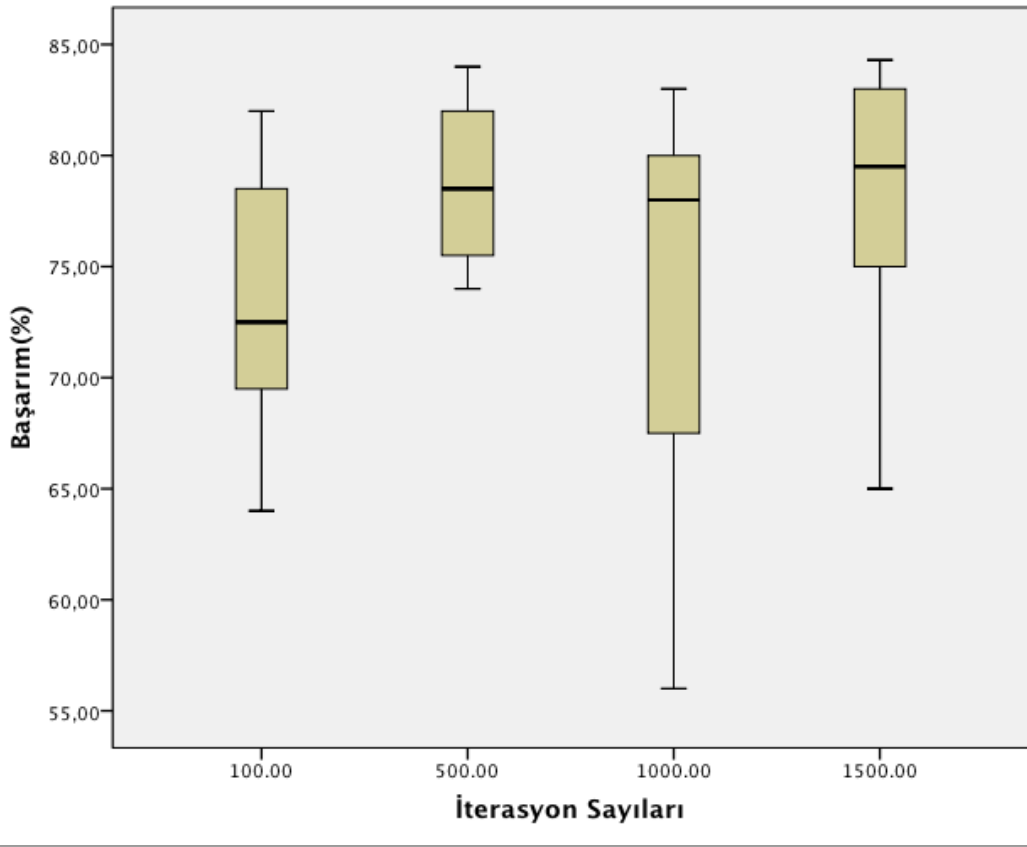
Şekil 4.9’de YSA sınıflandırıcılardan elde edilen kutu grafiklere bakarak, 50 nörona sahip sınıflandırıcıların daha tutarlı ve daha yüksek sonuç ürettiği söylenebilir. Bazı öğrenme algoritmalarının çalışması çok uzun sürdüğünden nöron sayısı olarak farklı yapıdaki YSA sınıflandırıcıları deneme imkanı bulunamamıştır.

4.3. İterasyon Sayıları Açısından Sınıflandırma Sonuçları Analizi

Tablo 4.5. İterasyon sayılarına göre YSA sınıflandırıcılarının başarımlar istatistikleri

İterasyon Sayıları		İstatistik	Std. Hata
100	Ortalama	73,3750	2,15421
	Standart Sapma	6,09303	
	Minimum	64,00	
	Maksimum	82,00	
500	Ortalama	78,7500	1,30589
	Standart Sapma	3,69362	
	Minimum	74,00	
	Maksimum	84,00	
1000	Ortalama	73,7500	3,28824
	Standart Sapma	9,30054	
	Minimum	56,00	
	Maksimum	83,00	
1500	Ortalama	78,0375	2,27439
	Standart Sapma	6,43294	
	Minimum	65,00	
	Maksimum	84,30	

Tablo 4.5.'te her bir iterasyon sayısından 8'er deneme yapılarak elde edilen başarımlar oranları gösterilmiştir. Tabloya bakıldığında 500 iterasyonun diğerlerinden daha yüksek ortalama ve daha düşük standart hataya sahip olduğu görülebilir. İterasyonun artması her zaman sınıflandırıcının performansını arttırmamaktadır.



Şekil 4.10. YSA sınıflandırıcılarının iterasyon sayılarına göre başarımlarının kutu grafikleri

Şekil 4.10. da denemeler sonucunda elde edilen başarımların iterasyon sayılarına göre grafikleri verilmiştir. Değişim aralığının düşük olması ve başarımlar ortalamasının yüksek olmasına bakılarak 500 iterasyon ile sınıflandırma yapılabileceği, iterasyonun artışının düzenli bir performans artımına neden olmayacağı söylenebilir.

5.SONUÇ

Cildiye bölümünde teşhis işlemi doktorların en çok sıkıntı çektiği durumlardan biridir. Cilt hastalıkları birbirine benzediği için yardımcı bir sistem yardımıyla bu sıkıntılar giderilebilir. Cilt hastalığının bir sistem yardımıyla teşhis edilmesi büyük kolaylıklar sağlayacağı gibi doğru teşhis etmeye de yardımcı olacaktır. Bu yardımcı sistem sayesinde doktorların çalışmalarının hızlanmasına da olumlu katkılar sunmaktadır.

Yapılan bu çalışmada Gabor dalgacık dönüşümü ile öznelik çıkarımı yapılarak ve YSA sınıflandırma tekniği kullanılarak cilt hastalığının teşhisi yapılmaya çalışılmıştır. Cilt hastalığının teşhisi 3 farklı hastalık grubu üzerinden farklı ara katmandaki nöron sayısı, iterasyon sayısı ve eğitim fonksiyonları ile yapılmıştır. Her defasında bu 3 faktör ile ysa sınıflandırıcı çalıştırılıp farklı sonuçlar elde edilmiştir. Elde edilen sonuçların çoğunun başarımı, ortalama olarak %80-%85 aralığında değişmektedir. Yapılan denemelerde en yüksek başarımlar 50 nöronlu ara katmana sahip trainscg öğrenme fonksiyonu ve 1500 iterasyonda %84,3 ile bulunmuştur. Yapılan denemelerde bu başarımların yanında ortalama olarak yakın standart sapma değerleri ile sonuçlar elde edilmiştir. Yapılan bu çalışma ile üçlü sınıflandırma yapıldığı için elde edilen başarımların kabul edilebilir bir aralıktadır.

Bu sınıflandırma tekniği ve testi sonrası elde edilen başarımların ileride cilt hastalıklarının teşhis edilmesinde kullanılabileceğini göstermiştir. Bu sistem geliştirilerek cilt hastalığının teşhis edilmesi işlemlerinde dermatoglara(cildiye uzmanlarına) büyük kolaylıklar sağlayacaktır.

6. EKLER

			GERÇEK DEĞERLER																								
		EĞİTİM ALG.	trainscg						trainrp						trainlm						trainbfg						
			25			50			25			50			25			50			25			50			
	NÖRON SAYILARI		Akne	Sedef	Mantar	Akne	Sedef	Mantar	Akne	Sedef	Mantar	Akne	Sedef	Mantar	Akne	Sedef	Mantar	Akne	Sedef	Mantar	Akne	Sedef	Mantar	Akne	Sedef	Mantar	
TAHMİN EDİLEN DEĞERLER	100	Akne	80	13	18	83	8	11	73	11	17	75	7	12	77	7	5	83	3	8	74	14	12	75	9	11	
		Sedef	13	67	13	9	81	11	16	69	14	12	77	17	14	79	22	11	87	24	12	62	16	11	75	10	
		Mantar	7	20	69	8	11	78	11	20	69	13	16	71	9	14	73	6	10	68	14	24	72	14	16	79	
	500	Akne	88	10	10	78	6	7	85	4	14	86	4	9	84	9	8	70	2	6	65	10	13	87	23	30	
		Sedef	3	75	13	7	84	14	5	78	14	6	86	13	4	75	18	10	86	12	25	68	26	4	66	11	
		Mantar	9	15	77	15	10	79	10	18	72	8	10	78	12	16	74	20	12	82	10	22	61	9	11	59	
	1000	Akne	90	9	11	88	5	4	88	5	14	85	3	5	83	8	15	80	6	7	67	12	10	84	16	12	
		Sedef	3	80	13	4	81	15	5	81	13	7	86	13	10	77	26	15	86	25	22	70	35	7	63	11	
		Mantar	7	11	76	8	14	81	7	14	73	8	11	82	7	15	59	5	8	68	11	18	55	9	21	77	
	1500	Akne	82	9	9	91	7	9	87	6	6	91	5	8	80	10	13	84	7	19	118	76	105	65	15	12	
		Sedef	7	81	15	3	85	14	4	82	15	2	80	13	7	68	24	6	76	10	48	91	71	11	68	27	
		Mantar	11	10	76	6	8	77	9	12	79	7	15	79	13	22	63	10	17	71	104	103	94	24	17	61	

7.KAYNAKÇA

- [1]. **Seyed Abolfazl H.**, 2016, neutron spectrum unfolding using artificial neural network and modified least square method, 126,75-84.
- [2]. **Gülbağ A.**, 2006, Yapay sinir ağı ve bulanık mantık tabanlı algoritmalar ile uçucu organik bileşiklerin miktarsal tayini, *Doktora Tezi*, Sakarya Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı, Sakarya.
- [3]. **McCullogh, WS., and Pitts, W.**, A logical calculus of ideas imminent in nervous activity. Bull. Math. Biophysics, 1943;5: 115-33.
- [4]. **Kınacı, A.C., Uğur A.**, 2006, Yapay Zeka Teknikleri ve Yapay Sinir Ağları Kullanılarak Web Sayfalarının Sınıflandırılması, Türkiye'de İnternet Konferansı Bildirileri TOBB Ekonomi ve Teknoloji Üniversitesi, Ankara, 21 - 23 Aralık
- [5]. **Karaboğa, D.**, 2004, Yapay Zeka Optimizasyon Algoritmaları, Nobel Akademi Yayıncılık, Ankara.
- [6]. **Я. А. Туровский, С. Д. Кургалин А. А. Адаменко**, 2016, Сравнительный анализ программных пакетов для работы с искусственными нейронными сетями, Системный Анализ И Информационные Технологии,1,161-168.
- [7]. **Nabiyev V.**, 2003, Yapay Zeka, Seçkin Yayıncılık, Ankara.
- [8]. **Hacıbeyoğlu, M.**, 2006, Çoklu etmen mimarisi ve takviyeli öğrenme, *Yüksek Lisans Tezi*, Selçuk Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Bilimleri Anabilim Dalı, Konya
- [9]. **Grossberg, S.** 1988. Nonlinear Neural Networks: Principles. Mechanisms, and Architectures, Neural Networks, vol. 1, no. 1. 17-61.
- [10]. **Küçükönder, H.**, 2011, Yapay Sinir Ağları ve Tarımda Bir Uygulama, *Doktora Tezi*, Kahramanmaraş Sütçü İmam Üniversitesi Fen Bilimleri Enstitüsü Zootekni Anabilim Dalı, Kahramanmaraş
- [11]. **Shahin, M. A., Tollner, E. W. and Gitaitis, R. D.** Classification of sweet onions based on internal defects using image processing and neural network techniques. Transactions of the ASAE, 2002, 45(5): 1613-1618.
- [12]. **Л. В. Хливненко, Ф. А. Пятакович, В. В. Васильев**, 2016, Формирование нейросетевых «портретов» классов скаттерграмм у больных мерцательной аритмией, Вестник Вгу, Серия: Системный Анализ И Информационные Технологии, УДК 004.032.26, 1, 95-101
- [13]. **McNeill G., Anderson D.**, 1992, Artificial Neural Network Technology

- [14]. **Fausett, L.**, 1994. Fundamentals of neural networks:Architecture, Algoritims and Applications, Prentice Hall
- [15]. **Kargın S.**, 2007, *Yüksek Lisans Tezi*, Yapay sinir ağları ile beyin tümörü tomografi görüntülerinin sınıflandırılması, İstanbul Üniversitesi Fen Bilimleri Enstitüsü Biyomedikal Mühendisliği Anabilim Dalı, İstanbul.
- [16]. **M.Seetha, I.V.Muralikrishn vd.** Artificial Neural Networks and Other Methods Of Image Classification, journal of Theoretical and Applied Information Technology, pp. 1039-1053
- [17]. **Görgel P.**,2011, *Doktora Tezi*, Görüntü işleme teknikleri kullanılarak 2 boyutlu mamografik verilerde kanserli bölge tanısı, İstanbul Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı, İstanbul.
- [18]. **Sofu Mehmet M.**, 2014, *Doktora Tezi*, Otomatik elma sınıflandırma makina tasarımı ve prototip imalatı, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Makine Mühendisliği Anabilim Dalı, Isparta.
- [19]. **Toylan H.**, 2012, *Doktora Tezi*, Yapay zeka örüntü tanıma algoritması kullanarak sınıflandırma otomasyonunun tasarımı, Trakya Üniversitesi Fen Bilimleri Enstitüsü Makine Mühendisliği Anabilim Dalı
- [20]. **Ölmez T.**, 1995, *Doktora Tezi*, Yapay sinir ağları yardımıyla biyomedikal dokuların sınıflandırılması, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektik-Elektronik Anabilim Dalı, İstanbul.
- [21]. **Daugman G.**, “Complete discrete 2-D Gabor transform by neural network for image analysis and compression,” IEEE Trans. On Acoustics, Speech and Signal Processing, 36(7), pp. 1169– 1179, 1988
- [22]. **Gabor D.**1946. Theory of communication. J. IEE, 93(1946): 429-45
- [23]. **Acar E.**, 2012, Yüksek Lisans Tezi, Yapay Zeka Yöntemleriyle Bitki Yaprak İmgelerinde Pas Hastalıklarının Tespiti, Dicle Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Anabilim Dalı
- [24]. **Pun, C., ve Lee, M.**, “Log-Polar Wavelet Energy Signatures for Rotation and Scale Invariant Texture Classification”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Cilt:25, No:5, Mayıs 200
- [25]. **Yeasin, M., Bullot, B., Sharma, R.**, 2006. Recognition of Facial Expressions and Measurement of Levels of Interest from Video, IEEE Trans. Multimedia, Vol. 8, no. 3, pp. 500-507
- [26]. **Şakar M.**, 2011, Yüksek Lisans Tezi, İnsan yüz imgelerinden yüz ifadesi tanıma, Fırat Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı

- [27]. **Manian, V. ve Vasquez, R.A.**1997. Framework for SAR Image Classification: Comparison of CO-occurrence and a Gabor based method. Geoscience and Remote Sensing,1997. IGARS '97. Remote Sensing-A scientific Vision for Sustainable Development. 1997 IEEE International, 1(1997): 335-337
- [28]. **Buciu,I. ve Gacsadi, A.** 2009. Gabor Wavelet Based Features for Medical Image Analysis and Classification. Applied Sciences in Biomedical and Communication Technologies. ISABEL 2009. 2nd International Symposium, 24-27 Nov. 2009, Bratislava. S, 1-4.
- [29]. **Reby, D., Lek, S.,** et al., 1997, Artificial neural networks as a classification method in the behavioural sciences, Behavioural Processes, 40, 35-43.
- [30]. **R. Setiono,** 1996, Extracting rules from pruned networks for breast cancer diagnosis, Artificial Intelligence in Machine, vol. 8, no. 1, pp. 37-51.
- [31]. **Bakpo, F. S., Kabari, L. G, Member,** Diagnosing Skin Diseases Using an Artificial Neural Network, <http://www.intechopen.com/books/artificial-neural-networksmethodological-advances-and-biomedical-applications/diagnosing-skin-diseases-using-an-artificial-neuralnetwork>
- [32]. **F. Ercal, A. Chawla, W. V. Stoecker,** Hsi-Chieh Lee and R. H. Moss, "Neural network diagnosis of malignant melanoma from color images," in *IEEE Transactions on Biomedical Engineering*, vol. 41, no. 9, pp. 837-845, Sept. 1994
- [33]. **Abut, S.,** 2014, Yüz görüntülerinden cinsiyet tahmini, *Yüksek Lisans Tezi*, Fırat Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Anabilim Dalı, Elazığ.
- [34]. **Hinton G.,** 1989, Connectionist learning procedures. Artificial Intelligence, 40, 185-234.
- [35]. **Martin Fodslette Moller.,** A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks, 6:525--533, 1993.
- [36]. **Battiti, R.,** 1990, Optimization methods for back-propagation: Automatic parameter tuning and faster convergence. IJNNC-90- I~ISH.DC. 1,593-596.
- [37]. **Lewis, A. S., & Overton, M. L.,** 2009, Nonsmooth optimization via BFGS. Submitted to SIAM J. Optimiz.
- [38]. **Martin Riedmiller und Heinrich Braun,** Rprop - A Fast Adaptive Learning Algorithm. Proceedings of the International Symposium on Computer and Information Science VII, 1992.

- [39]. **K. Madsen, H.B. Nielsen, and O. Tingleff**, 2004, Methods for Non-Linear Least Squares Problems. Technical University of Denmark.
- [40]. **Marquardt, Donald W.** "An algorithm for least-squares estimation of nonlinear parameters." *Journal of the society for Industrial and Applied Mathematics* 11.2 (1963): 431-441.
- [41]. **Andre Esteva, Brett Kuprel, Sebastian Thrun**, Deep Networks for Early Stage Skin Disease and Skin Cancer Classification YILI....
- [42]. **Haofu, L.**, A Deep Learning Approach to Universal Skin Disease Classification CSC 400 - Graduate Problem Seminar - Project Report.
- [43]. **Coşkun Ö.**,2001, *Doktora Tezi*, Uydu Görüntü verisinin yapay sinir ağları ile sınıflandırılması, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul
- [44]. <http://www.atlasdermatologico.com.br/> DERMATOLOGYATLAS (E.T.:20 Kasım 2015)
- [45]. **S., Arivazhagan, L., Ganesan**, 2003, Texture classification using wavelet transform, Pattern Recognition Letters, 24, 1513-1524.
- [46]. **Tai Sing Lee**, 1996, Image representation using 2d gabor wavelets, Ieee Transactions On Pattern Analysis And Machine Intelligence, 18, 1-13.
- [47]. **Chulwoo Pack, Sung Shin, Hyung-Do Choi** et al.,2016, Optimized multilayer perceptron using dynamic learning rate based microwave tomography breast cancer screening, SAC'16 Proceedings of the 31st Annual ACM Symposium on Applied Computing pages 2171-2175, New York, USA.
- [48]. **Aboul Ella, H., Hossam M. Moftah**, et al., 2014, MRI breast cancer diagnosis hybrid approach using adaptive ant-based segmentation and multilayer perceptron neural networks classifier, Applied Soft Computing, 14-A, 62-71.
- [49]. **Riccardo Taormina, Kwok-wing Chau**, 20015, Neural network river forecasting with multi-objective fully informed particle swarm optimization, Journal Of Hydroinformatics, 17, 99-113.
- [50]. <http://stackoverflow.com/questions/25889637/how-to-use-k-fold-cross-validation-in-a-neural-network/stackoverflow%20/> stackoverflow (E.T: 5 Nisan 2016)
- [51]. **Catherine, S. Yang, Mary T.**, et al., 2016, Inflammatory and glandular skin disease in pregnancy, Clinics in Dermatology, 34.3, 335-343.
- [52]. <http://www.saglikhastalik.com/kategori/cilt-hastaliklari/> Cilt Hastalıkları (E.T.:24 Mart 2016)

- [53]. **BAYKAL, Y.**, 2008, Deri ve mukoza ile ilgili paraneoplastik sendromlar, Türkiye Klinikleri Hematoloji Özel Dergisi, 1.3, 72-83.

ÖZGEÇMİŞ

1981 ŞIRNAK doğumlu Musa KAPLAN ilköğretimi İdilde, lise eğitimini Cizre’de tamamladı. 2001 yılında kayıt yaptığı Kırgızistan-Türkiye “Manas” Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünden 2006 yılında mezun oldu.2008 yılında askerlik yapmıştır. 2009 yılında Şırnak Üniversitesi Bilgi İşlem Daire Başkanlığında Bilgisayar Mühendisi olarak çalıştı. 2010 yılında Şırnak Üniversitesi Cizre Meslek Yüksekokulunda Öğretim Görevlisi olarak işe başladı. 2010 yılında Fırat Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Yazılım Anabilim Dalında yüksek lisans yapmaya hak kazandı. Halen Şırnak Üniversitesi Cizre Meslek Yüksekokulunda Öğretim Görelisi olarak çalışmaktadır