

**KARADENİZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**GRI İLİŞKİSEL ANALİZ TABANLI YENİ BİR LVQ YÖNTEMİ: ODUN TÜRÜ  
SINIFLANDIRMA ÜZERİNE BİR UYGULAMA**

**YÜKSEK LİSANS TEZİ**

**End. Müh. Miraç MURAT**

**ARALIK 2017  
TRABZON**



**KARADENİZ TEKNİK ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**GRI İLİŞKİSEL ANALİZ TABANLI YENİ BİR LVQ YÖNTEMİ: ODUN TÜRÜ**  
**SINIFLANDIRMA ÜZERİNE BİR UYGULAMA**

**Miraç MURAT**

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde**  
**"ENDÜSTRİ YÜKSEK MÜHENDİSİ"**  
**Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

**Tezin Enstitüye Verildiği Tarih : 19 / 12 / 2017**

**Tezin Savunma Tarihi : 09 / 01 / 2018**

**Tez Danışmanı : Doç. Dr. Şükrü ÖZŞAHİN**

**Trabzon 2018**

**KARADENİZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**Endüstri Mühendisliği Anabilim Dalında  
Miraç MURAT Tarafından Hazırlanan**

**GRİ İLİŞKİSEL ANALİZ TABANLI YENİ BİR LVQ YÖNTEMİ: ODUN TÜRÜ  
SINIFLANDIRMA ÜZERİNE BİR UYGULAMA**


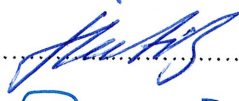

**başlıklı bu çalışma, Enstitü Yönetim Kurulunun 19/ 12/ 2017 gün ve 1732 sayılı  
kararıyla oluşturulan jüri tarafından yapılan sınavda  
YÜKSEK LİSANS TEZİ  
olarak kabul edilmiştir.**

**Jüri Üyeleri**

**Başkan : Doç. Dr. Şükrü ÖZŞAHİN**

**Üye : Doç. Dr. Fatih YAPICI**

**Üye : Doç. Dr. Emrullah DEMİRCİ**

  
.....  
  
.....  
  
.....

**Prof. Dr. Sadettin KORKMAZ**

**Enstitü Müdürü**

## ÖNSÖZ

Tez çalışmam boyunca akademik desteğinin yanı sıra yüksek lisans eğitimim boyunca da hiçbir zaman yardım ve desteğini benden esirgemeyen ve sahip olduğu titiz çalışma adabıyla bana çok kıymetli katkıları olan değerli danışmanım Sayın Doç. Dr. Şükrü ÖZŞAHİN'e, tezin hazırlanması sürecinde kıymetli fikirlerini paylaşan Sayın Prof. Dr. Coşkun HAMZEÇEBİ'ye, tezimin şekillenmesinde önemli katkısı olan değerli hocam Arş. Gör. Hüseyin Avni ES'e, tezin ortaya çıkması sürecinde karşılaştığım zorlukları aşmamdaki destek ve teşvikleri için Arş. Gör. Mehmet SEYHAN ve Arş. Gör. Ahmet ALÇI'ya teşekkür ederim. Ayrıca bu süreçte bana katlanan ve beni her şekilde destekleyen aileme sonsuz şükran ve sevgilerimi sunarım.

Miraç MURAT  
Trabzon 2018

## TEZ ETİK BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum “Gri İlişkisel Analiz Tabanlı Yeni Bir LVQ Yöntemi: Odun Türü Sınıflandırma Üzerine Bir Uygulama” başlıklı bu çalışmayı baştan sona kadar danışmanım Doç. Dr. Şükrü ÖZŞAHİN’in sorumluluğunda tamamladığımı, verileri kendim topladığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma süresince bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 09/01/2018

Miraç MURAT

## İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET .....	VII
SUMMARY .....	VIII
ŞEKİLLER DİZİNİ .....	IX
TABLolar DİZİNİ.....	X
SEMBOLLER DİZİNİ .....	XI
1. GENEL BİLGİLER.....	1
1.1. Giriş.....	1
1.2. Makine Öğrenmesi ve Öğrenme Yöntemleri .....	7
1.2.1. Danışmanlı Öğrenme.....	8
1.2.2. Danışmansız Öğrenme .....	10
1.2.3. Destekleyici Öğrenme .....	10
1.2.4. Yarı Danışmanlı Öğrenme.....	10
1.3. Sınıflandırma .....	11
1.3.1. Sınıflandırma Süreci.....	12
1.3.2. Sınıflandırma Yöntemleri.....	13
1.3.3. Sınıflandırıcı Performansının Değerlendirilmesi .....	14
1.3.4. Sınıflandırıcı Performans Ölçütleri .....	15
1.3.4.1. İkili Karşılaştırma Matrisi .....	15
1.3.4.2. ROC Eğrisi .....	17
1.4. Yapay Sinir Ağları.....	18
1.4.1. Yapay Sinir Hücresi ve Yapay Sinir Ağı .....	19
1.4.2. Çok Katmanlı Yapay Sinir Ağları .....	20
1.4.2.1. Çok Katmanlı Algılayıcı .....	22
1.4.3. Yapay Sinir Ağlarında Öğrenme .....	23
1.4.4. Yapay Sinir Ağlarında Eğitim ve Test Etme.....	24
1.5. Öğrenen Vektör Niceleme (LVQ).....	25
1.5.1. En Yakın Vektör Sınıflandırma.....	25

1.5.2.	LVQ Öğrenme Algoritmaları .....	26
1.5.3.	LVQ1 .....	27
1.5.4.	LVQ2.1 .....	28
1.6.	Gri İlişkisel Analiz .....	31
2.	UYGULAMA .....	34
2.1.	LVQ Sınıflandırma Modeli .....	37
2.2.	Gri LVQ1 Sınıflandırma Modeli .....	39
2.3.	ÇKA Sınıflandırma Modeli .....	40
3.	BULGULAR .....	42
4.	SONUÇ VE ÖNERİLER .....	52
5.	KAYNAKLAR .....	54
6.	EKLER .....	61
	ÖZGEÇMİŞ	

Yüksek Lisans Tezi

ÖZET

GRI İLİŞKİSEL ANALİZ TABANLI YENİ BİR LVQ YÖNTEMİ: ODUN TÜRÜ  
SINIFLANDIRMA ÜZERİNE BİR UYGULAMA

Miraç MURAT

Karadeniz Teknik Üniversitesi  
Fen Bilimleri Enstitüsü  
Endüstri Mühendisliği Anabilim Dalı  
Danışman: Doç. Dr. Şükrü ÖZŞAHİN  
2018, 61 Sayfa, 4 Sayfa Ek

Bu tez çalışması oldukça güçlü tahmin yeteneğine sahip olan yapay sinir ağları ile sınıflandırma modelleri geliştirilmesine odaklanmıştır. Çalışmada yapay sinir ağı tekniklerinden olan ve sınıflandırma uygulamalarında kullanılan Öğrenen Vektör Niceleme (LVQ) ile birlikte yeni bir sınıflandırıcı olan Gri LVQ kullanılmıştır. Odun türü belirleme üzerine bir uygulama yapılarak LVQ yöntemlerinin sınıflandırma performansları ve Çok Katmanlı Algılayıcı (ÇKA) sınıflandırma performansı kıyaslanmıştır.

LVQ sınıflandırma algoritmasında örnek vektör ile referans vektörler arasındaki benzerlikler referans vektörlerin ayrı ayrı değerlendirilmesiyle belirlenir. Gri LVQ algoritmasında ise Gri İlişkisel Analiz yoluyla bütün referans vektörlerinin birlikte değerlendirildiği bir sınıflandırma gerçekleşir. Bu çalışma ile LVQ, Gri LVQ ve ÇKA yöntemleriyle odun anatomisine ait biyometrik özellikleri girdi olarak kullanıp Akçaağaç (*Acer L.*) cinsine ait 4 farklı türün ayrımını yapan sınıflandırıcı modeller geliştirilmiştir. Literatürde fazla çalışılmamış olan, anatomik özelliklerin biyometrik ölçümleri kullanılarak odun türünün belirlendiği çalışmalara yeni bir yaklaşım sunan LVQ ve Gri LVQ yöntemleriyle elde edilen sınıflandırma başarıları ÇKA sınıflandırıcıyla benzer olmuştur. Her üç yöntem de 24 örnekten oluşan veri kümesinde %95,83 sınıflandırma doğruluğuna ulaşılmıştır.

**Anahtar Kelimeler:** Sınıflandırma, Doğrusal Vektör Ayırıcı (LVQ), Gri İlişkisel Analiz, Odun Türü Belirleme



Master Thesis

SUMMARY

A NEW LVQ METHOD BASED ON GRAY RELATIONAL ANALYSIS: AN  
APPLICATION ON CLASSIFICATION OF WOOD SPECIES

Miraç MURAT

Karadeniz Technical University  
The Graduate School of Natural and Applied Sciences  
**Industrial Engineering Graduate Program**  
Supervisor: Assoc. Prof. Şükrü ÖZŞAHİN  
2018, 61 Pages, 4 Pages Appendix

This thesis concentrates on the development of classification models with artificial neural network algorithms which are powerful estimation technique. In the thesis, Learning Vector Quantization (LVQ) one of the artificial neural network techniques is used in addition to Grey LVQ which is a new LVQ method. The classification performances of LVQ methods and Multilayer Perceptron (MLP) are compared with an application based on wood type determination.

The similarities between the sample vector (input vector) and the reference vectors in the LVQ classification algorithm are determined by evaluating the reference vectors individually. In the Gray LVQ algorithm, a classification which evaluates all the reference vectors together using the GRA is performed. To identify 4 different species belonging to genus of *Acer L.*, classifiers were developed by using the biometric features of wood anatomy as input of LVQ, Gray LVQ and MLP algorithms. The classification accuracies of LVQ and Gray LVQ algorithms, which offer a new approach to the rare studies in the literature in which the wood species are identified using biometric measurements of anatomical features are similar to the accuracy of the MLP classifier. In all three methods, 95.83% classification accuracy was achieved in the test data set consisting of 24 samples.

**Key Words:** Classification, Learning Vector Quantization (LVQ), Gray Relational Analysis, Identification Wood Species

## ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1. Makine öğrenmesi ve geleneksel öğrenme arasındaki fark.....	8
Şekil 2. Danışmanlı makine öğrenmesi süreci.....	9
Şekil 3. Farklı makine öğrenmesi sınıflandırma yöntemlerinin kullanım eğilimi.....	14
Şekil 4. Karşılaştırma Matrisi .....	15
Şekil 5. ROC eğrisi .....	17
Şekil 6. Yapay sinir hücresi .....	19
Şekil 7. Çok katmanlı bir yapay sinir ağı örneği .....	21
Şekil 8. Sigmoid (a) ve Hiperbolik Tanjant (b) aktivasyon fonksiyonları .....	22
Şekil 9. (a) 2 sınıf ve 22 örnekten oluşan örnek uzayı, (b) Referans nöronlar ve .....	27
Voronoi bölümleri, (c) Sınıflandırma sonucuna göre sınıf sınırları .....	27
Şekil 10. En yakın doğru ve yanlış referans nöronları .....	29
Şekil 11. 88 Akçaağaç örneği için ölçülen anatomik özellikler. Trahe Teğetsel Çap.....	35
(TTC), Trahe Radyal Çap (TRÇ), Trahe Hücre Uzunluğu (THU), Trahe.....	35
Sayısı (TS), Mültiseri Özışını Yüksekliği (MÖY), Mültiseri Özışını .....	35
Genişliği (MÖG).....	35
Şekil 12. Özışını Sayısı (ÖS), Mültiseri Özışını Genişliği (Hücre) (MÖGh), Lif .....	36
Uzunluğu (LFU), Lif Genişliği (LFG), Lümen Genişliği (LÜMG), .....	36
Lif Çeper Kalınlığı (LÇPK) .....	36
Şekil 13. (a) EK, Libriform liflerle çevrili traheler, paratraheal odun paransimi ve .....	37
küme şeklinde traheler, (b) RK, Homoselüler özışını, (c) TK, Vasküler .....	37
traheid ve uzun libriform lifler.....	37
Şekil 14. LVQ ağ modeli .....	38
Şekil 15. ÇKA ağ modeli.....	40
Şekil 16. Seçilen sınıflandırıcı modelleri için (a) eğitim doğrulukları ve .....	45
(b) test doğrulukları kutu grafikleri.....	45
Şekil 17. YSA sınıflandırıcıları başarı ortalamaları normallik testi sonucu.....	46
Şekil 18. Ortalama doğruluk oranına göre ÇKA ve LVQ başarıları t-test sonucu .....	47
Şekil 19. Ortalama doğruluk oranına göre ÇKA ve Gri LVQ1 başarıları t-test sonucu.....	47
Şekil 20. ÇKA modeli için (a) eğitim ve (b) test veri kümeleri ikili karşılaştırma matrisi 48	48
Şekil 21. ÇKA modeli için eğitim ve test veri kümelerine ait ROC eğrileri .....	48
Şekil 22. LVQ modeli için (a) eğitim ve (b) test veri kümeleri ikili karşılaştırma matrisi 49	49
Şekil 23. LVQ modeli için eğitim ve test veri kümelerine ait ROC eğrileri .....	49
Şekil 24. Gri LVQ1 modeli için (a) eğitim ve (b) test veri kümeleri ikili karşılaştırma ....	49
matrisi .....	49
Şekil 25. Gri LVQ1 modeli için eğitim ve test veri kümelerine ait ROC eğrileri .....	50

## TABLULAR DİZİNİ

	<b><u>Sayfa No</u></b>
Tablo 1. ÇKA sınıflandırıcıları için ağ yapıları, eğitim ve test performansları.....	42
Tablo 2. LVQ sınıflandırıcıları için referans nöron sayıları, öğrenme oranları, eğitim ve test performansları.....	43
Tablo 3. Gri LVQ1 sınıflandırıcıları için referans nöron sayıları, öğrenme oranları,.... eğitim ve test performansları .....	43
Tablo 4. 10 veri kümesi için seçilen uygun modellerin eğitim ve test başarıları (%) ....	44
Tablo 5. 10 veri kümesi için seçilen uygun modellerin eğitim ve test başarıları ..... ortalamaları (%) .....	46
Ek Tablo 1. Sınıflandırmada kullanılan odun örnekleri ve anatomik özellikleri .....	61

## KISALTMALAR DİZİNİ

ÇKA	: Çok katmanlı algılayıcı
DVM	: Destek vektör makineleri
GİA	: Gri ilişkisel analiz
GLCM	: Gri seviye eş oluşum matrisi
GST	: Gri sistem teorisi
LÇPK	: Lif çeper kalınlığı
LFG	: Lif genişliği
LFU	: Lif uzunluğu
LÜMG	: Lümen genişliği
LVQ	: Öğrenen vektör niceleme
MÖG	: Mültiseri özışını genişliği
MÖGh	: Mültiseri özışını genişliği (hücre)
MÖY	: Mültiseri özışını yüksekliği
MSE	: Hata kareleri ortalaması
ÖS	: Özışını sayısı
THU	: Trahe hücre uzunluğu
TRÇ	: Trahe radyal çap
TS	: Trahe sayısı
TTÇ	: Trahe teğetsel çap
YSA	: Yapay sinir ağları

## 1. GENEL BİLGİLER

### 1.1. Giriş

İnsan zekası çevreye duyarlıdır. Değişen koşullara uyum sağlama ve koşullara göre davranma yeteneğine sahiptir ve bunu öğrenme ile gerçekleştirir. Özellikle edindiği tecrübelerden davranış değişikliği sonucunu çıkarmak olan öğrenme insan zekasının en güçlü yönüdür.

Günlük yaşamda insan her an nesnelere, olayları, canlıları, duyguları vb. çeşitli kategorilerden birine ait olarak görür ve davranışlarına buna göre karar verir. Bunu yaparken nesnelere, olayların ya da canlıların çeşitli özelliklerini çok hızlı bir şekilde değerlendirir. Söz konusu bu değerlendirme süreci geçmişte yaşanan tecrübelerle dayanır. Tecrübe edilen olguların çeşitli özelliklerine ve bu özelliklerin belirli şartlara göre aldıkları değerlerine dayalı gözlemler, yeni durumların ait olabilecekleri kategorileri belirlemek için insanın geliştirdiği ve bilinçte saklanan kuralları oluşturur.

Bilgi ve iletişim teknolojilerindeki gelişmeler veri toplamayı ve depolamayı son derece kolaylaştırmış ancak verinin analizinde kullanılan yöntemlerin geliştirilmesi ihtiyacını da beraberinde getirmiştir. Verinin boyutunun giderek büyümesi, içerdiği bilginin yeniden öğrenilmesini sağlayacak veri analizinin önemini artırmıştır.

Veri analizi alanlarından biri olan veri madenciliği insanın öğrenme yeteneği ile birlikte tahmin yapma ve/veya karar verme yaklaşımını birleştirir. Davranışlarını çeşitli bilgiler doğrultusunda iyileştirebilen insanın bu yeteneğinin problemlere çözüm üretmede çok daha hızlı olan bilgisayarlara aktarılması düşüncesi veri madenciliğindeki sınıflandırma sürecinin esin kaynağı olmuştur. Sınıflandırma en genel anlamıyla mevcutta var olan bilgiyi temel alan karar verme ve/veya tahmin yapma işlemidir.

Sınıflandırma, veri madenciliği ve yapay zeka uygulamaları boyutuyla ele alındığında; ait oldukları kategorileri (sınıf) belli olan veriler kullanılarak bir model oluşturup yeni karşılaşılan verilerin hangi kategorilere ait olduğunu belirleme işlemidir. Sınıflandırmanın iki adımı vardır: veriyi analiz edip sınıflandırma kuralları (sınıflayıcı) oluşturmak, oluşturulan kurallar ile yeni verileri gruplandırmak. İlk kez karşılaşılan verinin bilinen sınıflar içinde hangisine ait olduğunu tespit edebilecek bir sınıflayıcı oluşturmak amacıyla sıklıkla başvurulan yöntemlerden biri makine öğrenmesidir.

Makine öğrenmesi de insan davranışları model alınarak geliştirilen bir bilgi işleme yöntemidir. Bilgisayarların çok daha büyük yapı ve karmaşadaki verileri işleyerek çıkarımlar yapması ve bu çıkarımlar ışığında kararlar verebilmesi için modeller geliştirilmesi makine öğrenmesi olarak adlandırılmaktadır. İnsana ait, tecrübelerden davranış değişikliği sonucunu çıkaran öğrenme yeteneğinin bilgisayarlara kazandırılmasını sağlayan makine öğrenmesi insan yeteneklerinin yeterli olamadığı durumlarda önemli bir yöntemdir.

Gerçek dünya problemlerine uyarlanmış sınıflandırma veya tahmin yapmak amacıyla karar ağaçları, Bayes sınıflandırıcılar, ilişki tabanlı sınıflandırıcılar, en yakın komşu sınıflandırıcı, destek vektör makineleri gibi değişik kural çıkarımı algoritmaları geliştirilmiştir. Geliştirilen kural çıkarım algoritmaları içinde en sık kullanılan makine öğrenmesi yöntemi ise Yapay Sinir Ağları (YSA)'dır.

Sınıflandırma yazı, ses ve görüntü tanıma, hastalık ön teşhisi, kredi risk derecelendirme, sahtekarlık tespiti gibi uygulama alanlarına sahiptir. Bu tez çalışması da makine öğrenmesi yöntemlerinin gerçek dünya problemlerine uygulandığı çalışma alanlarından biri olan orman ürünleri üzerine gerçekleştirilmiş ve odun sınıflandırma uygulamalarına ait literatür aşağıda özetlenmiştir.

Çeşitli ağaç türlerinin var olması kalite odaklı fiyatlandırma, yasal odun ticareti, arkeoloji ve tarih araştırmaları gibi konularda odun ya da ağaç türü belirlemeyi önemli bir araştırma alanı olarak ortaya çıkarmıştır (Wang vd., 2013). Türlerin ayrımının yapılabilmesi, turbalık alanlarda keşfedilen korunmuş örneklerin paleocoğrafik yayılımlarının tespitini, arkeolojik bölgelerde bulunan örnek türlerinin belirlenmesini, ahşap heykel ve ahşap eski binaların restorasyonunda aynı tür ile yenileme yapılmasını mümkün kılar (Esteban vd., 2017). Bir ağaç cinsinin ya da türünün belirlenmesi çiçeği, yaprağı veya tohumları dikkate alındığında kolayken odun işlendikten sonra ancak makroskobik ve/veya mikroskobik özelliklerinin belirlenmesi ile mümkün olur (Khalid vd., 2008). Morfolojik ve anatomik özellikleri benzer türleri sınıflandırmanın zor olması güvenilir bir sınıflandırıcı elde etmeye yarayan asıl özellikleri belirleyecek tekniklerin geliştirilmesini zorunlu kılmıştır. Literatürde odun türü analizinde klasik biyolojik sınıflandırmaya alternatif olan bilgisayar destekli sınıflandırma uygulamaları görüntü tabanlı ve spektrum tabanlı olarak iki temel yaklaşımla gerçekleştirilmektedir. Görüntü tabanlı sınıflandırma yaklaşımında odun yapısı, renkleri ve yüzey özellikleri yüksek çözünürlüklü kameralar veya mikroskoplar gibi çeşitli görselleştirme araçları ile incelenir. Spektrum tabanlı sınıflandırmada ise ahşap yüzeyin ses, ışık, sıcaklık gibi uyaranların etkisi sonucu elde edilen yayılma spektrumunun analizi ile

incelenir (Piuri ve Scotti, 2010). Görüntü tabanlı sınıflandırmada, odun numuneleri üzerinden elde edilen görüntülerden çıkarılan yoğunluk veya renk dağılımını gösteren doku tanımlayıcıları sınıflandırma için kullanılan öznitelikleri oluşturur (Nisgoski vd., 2017).

Görüntü tabanlı sınıflandırmada görüntülerin sınıflandırma algoritmalarıyla analizi için sayısal verilere dönüştürülmesi gerekir. Bu kapsamda yapılan çalışmaların hemen hepsi Gri Seviye Eş Oluşum Matrisleri (Grey Level Co-Occurrence Matrices: GLCM) kullanılarak görüntüleri dönüştürür (Piuri ve Scotti, 2010; Tou vd., 2009; Sun vd., 2011; Mallik vd., 2011). GLCM ile görüntü üzerindeki pikseller arasındaki gri seviye farklılıkları karşılaştırılarak görüntü doku karakteristiğini içeren çeşitli özellikler bulunur (Demirhan ve Güler, 2010). Elde edilen özellikler odun sınıflandırma uygulamaları için girdileri oluşturur.

Tou vd. CAIRO veri tabanından alınan 6 farklı odun türü için yaptıkları sınıflandırma uygulamasını k-en yakın komşu algoritması ile gerçekleştirmişlerdir. Görüntülerden doku karakteristiği çıkarmak için GLCM, Gabor filtresi, GLCM-Gabor filtresi birleşimi ve kovaryans matrisi yöntemlerinin kullanıldığı çalışmada en iyi sınıflandırma başarısı olan %85 doğruluk oranına Gabor filtresi tabanlı kovaryans matrisi ile ulaşılmıştır (Tou vd., 2009).

Martins vd. 112 farklı türe (85 cins – 30 familya) ait yapraklı ve iğne yapraklı ağaçlara ait mikroskobik görüntüleri kullanarak benzer çalışmalara örnek bir sınıflandırma veri tabanı oluşturmuşlardır. 2240 görüntü incelenerek yaptıkları çalışmada GLCM ve Bölgesel İkili Örüntü (BİÖ) yöntemleri ile görüntüler dönüştürülerek elde edilen özellikler Destek Vektör Makineleri (DVM), doğrusal ayırma analizi ve k-en yakın komşu sınıflandırma algoritmaları ile kullanılmıştır. İlk bölümünde yapraklı ve iğne yapraklı ağaç ayırımı yapılmış ve sonraki bölümünde de 112 türün sınıflandırması gerçekleştirilmiştir. BİÖ ve DVM hibrit yaklaşımı sayesinde en iyi sınıflandırma sonucu elde edilerek ilk bölümde %98 ve ikinci bölümde %86 başarıya ulaşılmıştır (Martins vd., 2013).

k-en yakın komşu algoritması için  $k$  değeri ve GLCM yöntemiyle elde edilen özellik sayısı parametrelerinin en uygun değerlerini sezgisel yöntemle bulmayı amaçlayan Hasan vd. parçacık sürü optimizasyonu algoritmasını kullanmıştır. 25 farklı odun türüne ait görüntüler üzerinden %68 doğruluğa sahip olan en iyi sınıflandırmanın, özellik sayısının %91,5 oranında azaltılarak  $k$  değeri 5 olduğunda yapılabileceğini göstermişlerdir (Hasan vd., 2013).

Odun türü ayırımı için kullandıkları sınıflandırıcının girdileri olan ve görüntü üzerinden elde ettikleri özelliklerin sınıflandırma başarısı üzerindeki etkisini iyileştirmeyi

hedefleyen Yadav vd., bu amaçla Gabor filtresi yönteminden faydalanmışlardır. Gabor filtresi ile iyileştirilen görüntülerden GLCM yoluyla 44 özellik elde edilmiştir. 25 farklı türün ayrımı için Levenberg-Marquart öğrenme algoritması ile eğitilen YSA modelleri iki farklı veri setinde %88,6 ve %92,6 sınıflandırma doğruluğu yakalamıştır (Yadav vd., 2013).

Doğrudan GLCM yönteminin değil, bu yöntemden esinlenerek iyileştirilen Gabor filtresi yönteminin kullanıldığı bir başka çalışmada sınıflandırma, çıkarılan doku tanımlayıcılarına göre k-en yakın komşu algoritması ile yapılmıştır. 24 farklı ağaç türüne ait örneklerden alınan mikroskobik görüntülerden çıkarılması gereken doku tanımlayıcı sayısı önerilen yeni yöntemle azaltılmıştır. İstatistiksel özelliklere dayalı 6 doku tanımlayıcısıyla sınıflandırma yapılmış ve başarılı sonuçlar elde edildiği gösterilmiştir (Wang vd., 2013).

Khalid vd. 20 farklı tropikal odun türünün ayrımını yapmak amacıyla makroskobik anatomiye gösteren yaklaşık 2000 görüntüden GLCM yoluyla elde edilen 20 özelliği kullanmıştır. Eğittikleri geri yayımlı YSA sınıflandırıcı ile söz konusu 20 tür için %95 sınıflandırma doğruluğu sağlamışlardır (Khalid vd., 2008).

Clark çalışmasında DELTA KEY olarak adlandırılan geliştirilmiş klasik biyolojik sınıflandırma yöntemi ile YSA yöntemini karşılaştırmıştır. 35 farklı *Lithops* türü için geleneksel yöntemlerle elde edilen 16 morfolojik özelliği kullandığı çalışmada karar ağacı benzeri bir sınıflandırma tekniği olan entropi temelli DELTA yöntemini ve ağ yapısı 16 girdi - 35 çıktı nörona sahip olan YSA yöntemini farklı durumlar için karşılaştırmıştır. Sınırlı sayıda verinin varlığı durumunda YSA sınıflandırıcının ayrımı zor olan benzer türler için daha başarılı olduğunu göstermiştir (Clark, 2003).

Daha önce aynı cins için gerçekleştirilen çalışmalardaki iş yükünü azaltmayı amaçlayan Clark vd. sadece yaprak görüntüleri kullanarak yaprak morfolojisine ait 22 özellik ile sınıflandırma yapmıştır. 4 farklı *Tilia* türü için 4 çıktı nöronuna sahip bir YSA sınıflandırması ile ortalama bir başarı elde edilmiştir. Ancak bu başarı sadece yaprak görüntüleri üzerinden otomatik olarak çıkarılan özellikler ile gerçekleştirildiğinden, çiçek ve diğer yapıların özelliklerinin elle çıkarıldığı önceki çalışmalara alternatif olabileceği savunulmuştur (Clark vd., 2012).

Bazı odun türü ayrımı çalışmalarında önerilen sınıflandırma veri tabanı fikrini dikkate alan, farklılık (dissimilarity) vektörü sınıflandırıcı seçimini dinamik olarak gerçekleştiren bir çalışmada klasik doku tanımlayıcısı çıkarım yöntemlerine ek olarak referans nokta odaklı özellik çıkarımı yöntemleri kullanılmıştır. Farklı özellik çıkarım yöntemi ve sınıflandırıcı



birleşimlerinin performanslarının kıyaslandığı çalışmada 112 farklı tür içeren veri tabanındaki örneklerle %93 sınıflandırma başarısına ulaşılmıştır (Martins vd., 2015).

Mikroskobik görüntüleri kullanıp ancak görüntüler üzerinden otomatik çıkarım yapmayan ve çeşitli anatomik özellikleri ölçerek sınıflandırma yapan çalışmalar da mevcuttur. Yaprak yapısının ayırıcı özelliğinin daha güçlü olduğu 3 farklı *Camellia* türü toplam 47 alt tür üzerinde sınıflandırma çalışması yapan Jiang vd. yaprak anatomisine ait 7 farklı özelliğin sayısal değerlerini ölçmüştür. Ölçümlerin YSA ve DVM için girdi olarak kullanıldığı çalışmada eğitim aşamasında %84 ve tahmin aşamasında ise %90.91 başarıyla DVM daha başarılı sınıflandırıcı olduğu sonucuna ulaşılmıştır (Jiang vd., 2013).

7 farklı türün kış odunu örneklerine ait mikroskobik görüntüleriyle yapılan sınıflandırma çalışmasında trahe yapısının 5 özelliği kullanılmıştır: dairesellik, dörtgenlik, trahe sayısı, traheler arası uzaklık ve ortalama trahe alanı. 101 örnek ile gerçekleştirilen çalışmada doğrusal ayırma analizi, lojistik regresyon, kareli (quadratic) sınıflandırıcı, k-en yakın komşu, yalın Bayes, DVM ve YSA sınıflandırıcı olarak kullanılmıştır. Örneklerin ait olduğu türün belirlenmesinde %83 sınıflandırma başarısına sahip yalın Bayes en iyi sınıflandırıcı olurken yapraklı ve iğne yapraklı ağaç ayırımında %93 sınıflandırma başarısıyla k-en yakın komşu ve YSA en iyi başarıyı göstermiştir (Mallik vd., 2011).

Mikroskobik görüntülerden elde edilen odun anatomisine ait çeşitli biyometrik özelliklerin kullanılarak sınıflandırma yapıldığı diğer çalışmalar Esteban ve arkadaşlarına aittir. Esteban vd. benzer yapıya sahip odun türü ayırımının ölçülmüş çeşitli anatomik özelliklerle uygulandığında geleneksel görüntü tabanlı sınıflandırmaya göre daha kolay ve yüksek doğrulukla yapılabileceğini *Juniperus* cinsi iki farklı tür ile göstermiştir (Esteban vd., 2009). İleri beslemeli YSA ile sınıflandırma yaparak *J. cedrus* ve *J. phoenicea* var. *canariensis* türlerinin %92 başarılı ile ayırımının yapıldığı gösterilmiştir. Esteban ve arkadaşları diğer bir çalışmalarında sarıçam ve karaçam türlerinin ayırımı için bir YSA modeli geliştirmiştir (Esteban vd., 2017). Sınıflandırma girdilerini, ölçtükleri odun anatomik özelliklerinden, temel bileşen analizi ile belirledikleri çalışmalarında Resilient (çabuk yakınsayan) öğrenme algoritması ile YSA eğitimi yapmışlardır. Trahe sayısı, trahe çapı, özışın sayısı gibi 8 farklı anatomik özelliğin girdi olarak kullanıldığı sınıflandırma modeli ile test verisinde %84 sınıflandırma başarısı elde edilmiştir.

Anatomik yapının biyometrik analizi ile gerçekleştirilen bir başka çalışma DVM ile *Salix alba*, *Salix caprea* ve *Salix elaeagnos* türlerinin ayırımı üzerine yapılmıştır (Turhan ve Serdar, 2013). 83 örnekten mikroskobik yöntemlerle IAWA (Uluslararası Odun

Anatomistleri Birliđi) standartlarına uygun olarak elde edilen trahe sayısı, lümen çapı, özışın sayısı gibi 13 farklı özellik kullanılarak yapılan odun türü ayırımı ile %95,2 test başarısına ulaşılmıştır.

Farklı YSA türleri kullanılarak, görüntülerden ölçülen yaprak şekil ve damar yapısına ait 31 karakteristik ile gerçekleştirilen sınıflandırma çalışması *Camellia* cinsine ait beş farklı türün ayırımı için yapılmıştır. 93 yaprak örneđi görüntülerinden elde edilen özellikler Öğrenen Vektör Niceleme (LVQ), Dinamik Sinir Ağları ve DVM yöntemleri için sınıflandırma verisi olarak kullanıldığı çalışma ile en yüksek başarıyı %97 doğruluk oranıyla DMV ile elde etmiştir (Lu vd., 2012).

Görüntü işleme yöntemleri ya da görüntülerdeki çeşitli özelliklerin ölçülmesi yoluyla doku çıkarımı yapılan yöntemlere benzer olarak spektrum tabanlı sınıflandırmada da çeşitli tekniklerle elde edilen spektrum grafikleri üzerinden çıkarımı yapılan özellikler kullanılır.

Her odun türünün farklı elastik tepki karakterine sahip olmasından hareketle oduna gönderilen ultrasonik ses dalgalarının yansımalarını kullanan Jordan ve arkadaşları YSA ile sınıflandırma yapmıştır. Gönderilen ses dalgalarını, farklı sönümleyici yapıları nedeniyle farklı yansıtan meşe, kızılâğaç, akçaağaç ve çam cinsi örneklerden elde edilen 120 veri kullanılmıştır. Eğitilen YSA sınıflandırıcı ile test verisi olarak kullanılan 40 veriden sadece bir tanesi doğru olarak sınıflandırılmamıştır (Jordan vd., 1998).

Brandtberg, kızılötesi ışınlarla çektiđi yüksek çözünürlüklü ağaç fotoğraflarını kullanılarak bulanık kümeler ile sınıflandırma yaptıđı çalışmasında fotoğraflardan elde ettiđi 9 farklı özelliđi temel bileşen analizi yöntemi ile dönüştürmüştür. Sınıf üyeliklerini bulanık kümelere üyelik dereceleri ile belirlediđi çalışmasında %67 sınıflandırma doğruluk oranı elde etmiştir (Brandtberg, 2002).

Odun numunesi yüzeylerine gönderilen kızılötesi ışınların odun dokusuna göre yansımalarından faydalanarak odun türünün belirlenmesi amacıyla gerçekleştirilen sınıflandırma çalışmaları da yapılmıştır (Brunner vd., 1996). 8 farklı türün ayırımını yakın kızılötesi ışın spektrum grafikleri ile yapan Tsuchikawa ve arkadaşları spektrum grafiklerinin ikinci derece türevlerini kullanmıştır. Sınıflandırmanın mahalanobis uzaklıđı kullanarak ayırma analizi ile yapıldığı çalışmada iğne yapraklı ağaç ayırımının yapraklı ağaç ayırımından daha zor olduđu tespit edilmiştir (Tsuchikawa vd., 2003). Kızılötesi ışın spektrum grafiklerinin kullanıldığı bir başka çalışmada farklı odun türlerine ait 89 örnek ile geri yayımlı YSA ve genelleştirilmiş regresyon sinir ağlarının eğitimi yapılmış, sırasıyla %97 ve %99 sınıflandırma başarısına ulaşılmıştır (Huang vd., 2012).

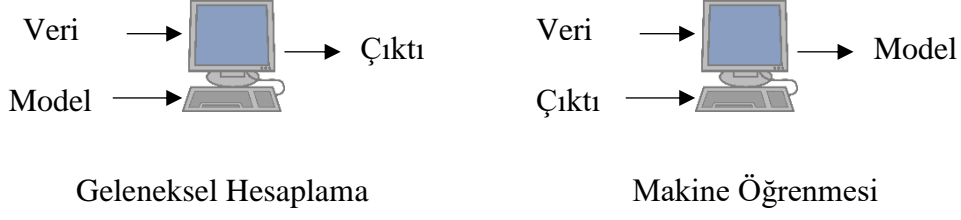
Odon numuneleri üzerine gönderilen lazer ışını yansımalarının bir spektrometre yardımıyla toplanmasıyla elde edilen lazer ışını yoğunluk grafikleriyle odun türü belirleme konusunda çalışan Piuri ve Scotti 4 farklı sınıflandırıcı kullanmıştır. 21 farklı odun türü üzerinde çalışan Piuri ve Scotti lazer ışını spektrumlarından elde ettikleri enerji değerlerini girdi olarak kullandıkları k-en yakın komşu, DVM, doğrusal ve kareli (quadratic) sınıflandırıcı algoritmalarıyla başarılı sonuçlar elde etmiştir (Piuri ve Scotti, 2010).

Spektrum tabanlı odun türü sınıflandırmada ısı işlem gören odunların kütle kayıp grafikleri de kullanılmıştır (Francisco-Fernández vd., 2015; Francisco-Fernández vd., 2012). 7 ayrı tür ile yapılan çalışmalarda termogravimetrik analiz ile elde edilen kütle kayıp eğrileri ve türevleri kullanılmış, doğrusal ayırma analizi, k-en yakın komşu, yalın Bayes, YSA ve DVM yöntemleri ile parametrik olmayan sınıflandırıcı K-NFDA yöntemlerinin sınıflandırma performansları araştırılmıştır. Eğrilerin kendileri kullanılarak yapılan sınıflandırmada doğrusal ayırma analizi, türevleri ile yapılan sınıflandırma uygulamalarında ise DVM en iyi performansa ulaşmıştır.

Bu tezde görüntülerden ölçüm yoluyla elde edilen odunun anatomik özellikleri kullanılarak gerçekleştirilen odun türü belirleme çalışmalarında daha önce uygulanmayan LVQ algoritmaları ile sınıflandırma yapılmış ve sonuçları çok katmanlı algılayıcı YSA ile karşılaştırılarak sunulmuştur. Yeni bir LVQ yöntemi olan Gri LVQ öğrenme yöntemi ile odun türü sınıflandırmada LVQ'nun iyileştirilmesi konusu araştırılmıştır.

## 1.2. Makine Öğrenmesi ve Öğrenme Yöntemleri

Makine öğrenmesi (machine learning) hesaplamalı öğrenme (computational learning) ve örüntü tanıma çalışmalarıyla ortaya çıkmıştır (Angra ve Ahuja, 2017). Bishop, örüntü tanımayı (pattern recognition) bir veride var olan düzenin ortaya çıkarılması ve ortaya çıkarılan bu düzen ile karar verme, tahmin yapma ya da sınıflandırma gibi eylemlerin gerçekleştirilmesi olarak tanımlar (Bishop, 2006). Makine öğrenmesinin amacı verideki gizli örüntüyü bilgisayarlar tarafından otomatik olarak keşfedecek yöntemleri bilgisayarların kendilerinin geliştirmesini sağlamaktır. Makine öğrenmesi bu yönüyle Şekil 1'de ifade edildiği gibi oluşturulmuş bir model doğrultusunda hesaplama yapan geleneksel yöntemlerden ayrılır. Makine öğrenmesi çeşitli model ve algoritma tasarımında büyük boyutlu verideki trend ve ilişkilerin analizini temel aldığından yapay zekanın ayrılmaz bir parçasıdır (Angra ve Ahuja, 2017).



Şekil 1. Makine öğrenmesi ve geleneksel öğrenme arasındaki fark

Makine öğrenmesi yöntemleri kullanılarak çeşitli alanlarda birçok çalışma yapılmıştır. Radar görüntüleri üzerinden petrol sızıntılarının tespiti (Kubat vd., 1998), göğüs kanserinin tekrarlanması belirlenmesi (Jerez vd., 2010), uydu görüntüleri ile bitki örtüsü değişikliğinin ve kırsal bölgelerin haritalanması (Otakei ve Blaschke, 2010), yeni öğrencilerin ilk yılsonunda okulu bırakma nedenlerinin tahmini (Delen, 2010), bina özelliklerine göre iklimlendirme için gerekli enerji tahmini (Tsanas ve Xifara, 2012), depresyondaki ve sağlıklı kişilerin tespiti (Hosseini et al., 2013) makine öğrenmesi alanında gerçekleştirilen çalışmalardandır.

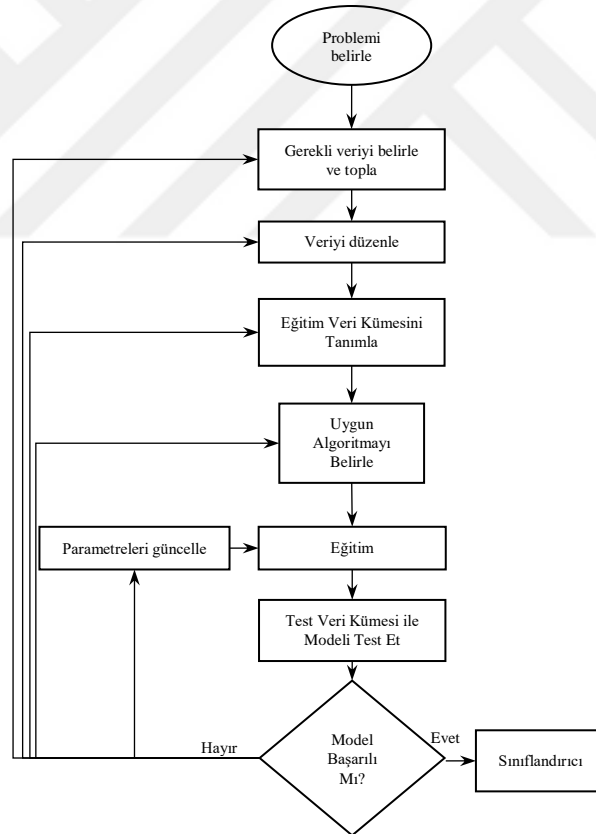
Makine öğrenmesinin dayandığı temel mantık, iletişim kurulan çevreden faydalı bir şeyler öğrenmektir. Simon öğrenmeyi bir sistemin yerine getirdiği görevleri gelecek seferde daha verimli/etkili şekilde yapmasını sağlayacak değişiklikler olarak tanımlamaktadır. Witten ve Frank öğrenmenin performans odaklı olduğunu belirtmiş ve daha iyi bir performans sağlayan davranış değişikliğini öğrenme olarak nitelendirmiştir (Witten ve Frank, 2005). Genel olarak öğrenme yöntemleri dörde ayrılır: Danışmanlı, danışmansız, destekleyici ve yarı-danışmanlı öğrenme.

### 1.2.1. Danışmanlı Öğrenme

Örneklerden öğrenme olarak da ifade edilen danışmanlı öğrenme (supervised learning), geçmiş verilerden (gözlenmiş ya da ölçülmüş örnekler kümesi) gelecekle ilgili tahmin yapmaktır. Danışmanlı öğrenme yüksek doğrulukta tahmin yapmaya odaklanır. Bir veride var olan ve her biri bir çıktıya sahip olan örnekler yardımıyla gelecekte benzer örneklerin sonuçlarını tahmin edecek model oluşturulması amaçlanır. Bu yapılırken mevcut verideki örnekler için üretilen çıktılarda yapılan hatalara göre modelin güncellenmesi öğrenmeyi temsil eder.

Şekil 2 danışmanlı makine öğrenmesi süreç akışını göstermektedir (Kotsiantis, 2007). Probleme uygun verinin toplanması sürecin ilk adımındır. Veri toplama, mümkün tüm özelliklerin bir uzmanın yönlendirmesi doğrultusunda belirli özellikler üzerinden ölçümler ya da gözlemler ile elde edilir. Yanlış, eksik ve gereksiz ölçümlere göre veri düzenlenir ve model geliştirilmesi için kullanılacak bir eğitim verisi oluşturulur. Uygun model seçim adımı gerçekleşen öğrenme ile ilişkilidir. Eğitim verisi ile geliştirilen model test kümesi ile değerlendirilir. Eğer modelin sonuçları geliştiriciyi tatmin etmiyorsa modele ait parametreler güncellenerek ya da başka bir yöntem seçilerek işlemler tekrarlanır.

Tahmin yapma, karar verme ve sınıflandırma problemleri için çok sayıda danışmanlı öğrenme yöntemi geliştirilmiştir. Bu yöntemlerden en sık kullanılanı regresyon, karar ağaçları ve yapay sinir ağlarıdır. Sınıflandırma yöntemi ve bu çalışmada kullanılan danışmanlı öğrenme yöntemleri ilerleyen bölümlerde ayrıntılı olarak anlatılmıştır.



Şekil 2. Danışmanlı makine öğrenmesi süreci

### 1.2.2. Danışmansız Öğrenme

Bazı veri kümelerinde sadece bir referans örnekler varken öğrenmeye katkı sağlayacak çıktılar yoktur. Ancak yine de danışmansız öğrenmeyle (unsupervised learning), veride var olan ve veriyi anlaşılır hale getirecek olan öz yapılar ortaya çıkarılabilir. Danışmansız öğrenme veride birbirinden ayrıştırılabilecek farklı grupların var olup olmadığının belirlenmesine odaklanır (Pérez-Ortiz vd., 2016). Danışmansız öğrenme “kesin doğru” olanı elde etmeye çalışmaz, kullanılan yöntemle göre veride var olması düşünülen yapının varlığını araştırır. Özellikle veri madenciliği alanında birliktelik analizleri, kümeleme, özellik çıkarımı yapmak amacıyla kullanılan kendini örgütleyen haritalar ve k-ortalamlar sık kullanılan danışmansız öğrenme yöntemleridir.

### 1.2.3. Destekleyici Öğrenme

Destekleyici (reinforcement learning) öğrenme öğrenci modelin iletişim halinde olduğu dinamik çevrenin davranışlarını deneme yanılma yoluyla öğrenmesini sağlayan bir öğrenme tekniğidir (Sutton, 1984). Danışmanlı öğrenmenin aksine, sadece doğru sonuçlar üreten verinin kullanılmadığı destekleyici öğrenmede, tüm veri uzayı öğrenciye sunulur. Öğrenci çevrenin durumuna göre olası eylemlerden birini gerçekleştirir, buna karşılık eylemin ne kadar başarılı olduğu konusunda bir destek sinyali alır. Destekleyici öğrenmede amaç, öğrenciye hangi eylemi gerçekleştirmesi gerektiğini değil öğrencinin ne kadar başarılı olduğu söylemektir. Böylece en yüksek başarıyı elde edecek eylem politikasının keşfedilmesi sağlanır (Kotsiantis, 2007; Kober ve Peters, 2012).

### 1.2.4. Yarı Danışmanlı Öğrenme

Büyük kısmı çıktı değerine sahip olmayan veride, sadece çıktısı olan verileri kullanmak yerine çıktı değeri olmayan veriyi de kullanarak örüntülerin keşfedilmesi de mümkündür (Barber, 2012). Danışmanlı ve danışmansız öğrenme yöntemlerinin birleştirildiği bu yöntemde, gerçek hayatta daha kolay elde edilen çıktısı olmayan verilerin öğrenmeye katkısı sağlanır. Yarı-danışmanlı öğrenme, (semi-supervised learning) çıktı

değerine sahip az sayıdaki veri ile geliştirilen modelin çıktı değeri olmayan daha fazla sayıda veri ile iyileştirilmesini hedefler.

### 1.3. Sınıflandırma

Sınıflandırma, nesnelerin *sınıf* olarak adlandırılan birbirinden bağımsız ve kendine özgü özelliklere sahip kategorilerden hangisine ait olduğunun belirlenmesidir. Sınıflandırmayla yeni bir nesnenin hangi sınıfa ait olacağına karar vermek amacıyla veri kümesi üzerinden sınıflayıcı örüntüler ya da kurallar türetilir. Temel görev gelecekte ele alınacak nesnelere söz konusu sınıflardan biriyle etiketleyebilme yeteneği kazanmış bir sınıflandırıcı model geliştirmektir (Pérez-Ortiz vd., 2016).

Sınıflandırma da tahmin gibi geleceğe ait durumlar hakkında karar verebilmeyi sağlayacak modeller çıkarmak için kullanılan bir veri analiz biçimidir. Sınıflandırma kategorik sonuçları, tahmin ise sürekli-değerli sonuçları modellemektedir (Köklü, 2014).

Sınıflandırma yönteminin anlaşılması için genel olarak makine öğrenmesine ait olan çeşitli kavramların bilinmesi gerekir. Makine öğrenmesinde olduğu gibi sınıflandırma yöntemleri de dört temel bileşenden oluşur: Bağımlı değişken, bağımsız değişken, girdi veri kümesi ve test veri kümesi (Lewis, 2007).

Bir sınıflandırıcı doğru sınıflandırmayı nasıl yapacağını girdi veri kümesini (input) oluşturan özgün *örnekleri* (*instances*) analiz ederek öğrenir. Girdi veri kümesi sınıflandırma problemine ait öğrenilecek içeriği oluşturan ve ait oldukları sınıf değerleri bilinen örneklerden oluşur (Witten ve Frank, 2005).

Öğrenilecek içeriğin bir kısmını temsil eden her bir özgün örnek, sınıf değerleriyle ilintili olan ve örneklerin farklı karakteristiklerini tanımlayan aynı bir takım *öznitelikten* (*attribute*) meydana gelir. Özniteliklerin değerleri gözlem veya ölçüm yöntemiyle elde edilirler. Bu nedenle öznitelikler sınıflandırma yöntemlerinde bağımsız değişkendirler. Öznitelikler sayısal ya da nominal (sembolik) değerler olabilirler. Sayısal öznitelikleri sürekli sayısal değerler oluştururken kategorik durumları temsil eden öznitelikleri sayısal ya da metinsel ifadeler oluşturur.

Sınıflandırıcının keşfedeceği örüntülere göre atama yapacağı sınıflar, örnekleri oluşturan özniteliklerle ilişkili olduğundan sınıflandırma yöntemindeki bağımlı değişkenlerdir. Sınıflandırma problemlerinde sınıf değerleri kategoriktir ve sınıf sayıları probleme özgü olarak sınırlıdır. Sınıflar her nesnenin yalnızca tek bir sınıfa ait olacağı

şekilde tanımlanır (Bailey, 1994). Eğer sınıf değerleri iki durumu (evet-hayır) temsil ediyorsa problem ikili (binary) sınıflandırma, ikiden fazla durumu temsil ediyorsa çoklu (multi-class) sınıflandırma problemi olarak adlandırılır. Sınıf değerleri dikkate alındığında sınıflandırmanın, bağımlı değişkenin sürekli sayısal değerler aldığı tahmin yöntemi olan regresyon modeline benzediği söylenebilir (James vd., 2013).

Girdi veri kümesindeki örnekleri doğru sınıflara atamayı yapacak örüntüleri öğrenen sınıflandırıcının gelecekte benzer özellikteki örnekleri de doğru sınıflandırması beklenir. Bir sınıflandırıcının iyi bir model olup olmadığı test veri kümesi için gösterdiği performans ile belirlenir. Test veri kümesi sınıf değerleri bilinen ancak sınıflandırıcının öğrenmesi sürecinde karşılaşmadığı örneklerin kümesidir.

Yukarıda verilen terimler dikkate alındığında sınıflandırma problemi matematiksel olarak aşağıdaki gibi ifade edilir.

$x(x_1, x_2, \dots, x_k) \in X \subseteq \mathbb{R}$   $k$  öznitelikten oluşan bir örnek,  $y \in Y = \{c_1, c_2, \dots, c_Q\}$   $Q$  adet sınıf etiketinden biri ve  $D = \{(x_i, y_i), i = 1 \dots N\}$   $N$  adet  $k$  boyutlu örnekten oluşan  $y$  sınıf değerlerine sahip eğitim verisi kümesi ise sınıflandırma ile  $X$  ve  $Y$  arasındaki ilişkiyi gösteren kural ya da fonksiyon  $g: X \rightarrow Y$  elde edilir.

### 1.3.1. Sınıflandırma Süreci

Sınıflandırma süreci birbiri ardına gerçekleştirilen eğitim (training), değerlendirme (evaluate) ve uygulama aşamalarından oluşur (White, 2000).

Eğitim aşamasında sınıfları birbirinden ayırt etmeyi sağlayacak parametrelerin hangi değerler olması gerektiği araştırılır. Bu aşamada doğru sınıflandırma yapabilen en iyi örüntüler ve kullanılan algoritmanın parametrelerinin en uygun değerleri bulunur. Eğitim sınıf değerleri bilinen eğitim veri kümesi ile gerçekleştirilir. Eğitim veri kümesi problem örnek uzayını temsil eder. Bu nedenle sınıflandırıcı eğitiminin başarılı olması için eğitim veri kümesi hem sık hem de nadir karşılaşılan yeterli sayıda örnekten oluşmalıdır.

Örnekleri oluşturan özniteliklerin seçimi de sınıflandırma başarısını etkiler. Başarıyı artırmak ve uygulama süresini kısaltmak için, her bir örneğin çeşitli ölçülebilir özelliklerini gösteren birbirleriyle özdeş olmayan öznitelikler sınıf değerleri ile ilintili olmalıdır (White, 2000; Kotsiantis vd., 2006). İdeal özniteliklerin belirlenmesi için öneriler getiren çalışmalar, çeşitli öznitelik kombinasyonlarını denemek ya da birleştirme yoluyla yeni öznitelikler



oluşturmak gibi yöntemler üzerinde durmuştur (Saeys vd., 2007). Aynı zamanda bir sınıflandırma algoritması olan lojistik regresyon yönteminin de anlamlı özniteliklerin belirlenmesi için kullanılabileceği önerilmiştir (Lewis, 2007).

Eğitimi gerçekleştirildikten sonra sınıflandırıcının başarısının ölçülmesi gerekir. Başarının ölçülmesi hem sınıflandırıcının kullanılmasına karar vermede hem de geliştirilen farklı sınıflandırıcıların karşılaştırılmasında zorunludur (White, 2000). Yansız bir performans ölçümü elde edilebilmesi için test veri kümesindeki örneklerin eğitim kümesindeki örneklerle aynı özellikleri taşıması gerekir (Barber, 2012).

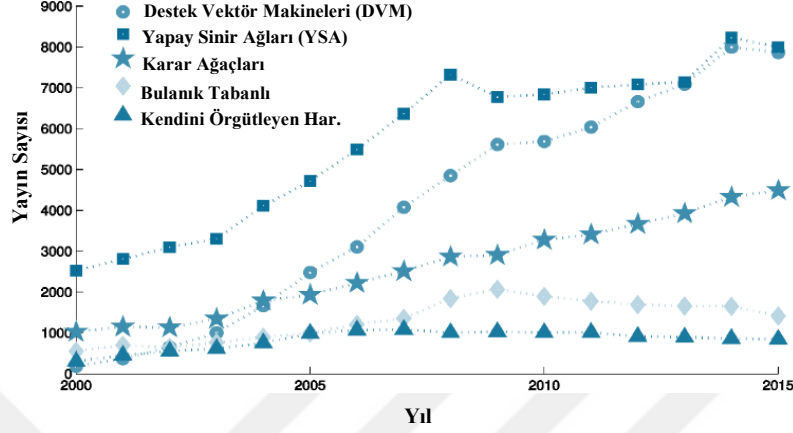
Test örnekleri için sınıflandırıcının yaptığı sınıf atamalarının gerçek sınıf değerleriyle olan uyumu sınıflandırma doğruluğu olarak adlandırılır. Diğer iki önemli performans ölçüm aracı yanlış sınıflandırma oranı ve genel tahmin gücüdür (Olson ve Delen, 2008). Sınıflandırma sonucu değerlendirme yöntemleri “Sınıflandırıcı Performansının Değerlendirilmesi” başlığı altında ayrıntılı olarak anlatılmıştır.

### 1.3.2. Sınıflandırma Yöntemleri

Sınıflandırma çok geniş bir uygulama alanına sahiptir. Bu durum da çok çeşitli sınıflandırma algoritmaları geliştirilmesine yol açmıştır. Olson, matematiksel yöntemler olarak nitelediği ikili karar ağaçları, sinir ağları, doğrusal programlama ve istatistiksel sınıflandırıcıların en sık kullanılan sınıflandırıcılar olduğunu belirtmiştir (Olson ve Delen, 2008). Şekil 3’te 2000 yılından bu yana sınıflandırma problemlerinde kullanılan yöntemler ve kullanım sayıları gösterilmektedir (Pérez-Ortiz vd., 2016).

Geliştirilen algoritmalar birçok çalışmada farklı başlıklara göre sınıflandırılmıştır. Pérez-Ortiz ve arkadaşları sınıflandırma yöntemlerinin genel olarak öğrenme yöntemlerine, model türüne, kullanılan mantığa, kullanım amacına ve öğrenme süreçlerine göre ayrılabilirliğini belirtmiştir (Pérez-Ortiz vd., 2016). Bu tezde, Kotsiantis ve arkadaşlarının önerdiği mantık temelli, istatistiksel ve algılayıcı temelli sınıflandırıcı ayrımı referans alınmıştır (Kotsiantis vd., 2006). Buna göre karar ağaçları ve kural tabanlı sınıflandırıcılar sembolik sınıflandırıcılar olarak da adlandırılan mantık temelli sınıflandırıcılardır. Örneklerin sınıflara ait olması olasılıkları üzerine sınıflandırma yapan Bayesyan ve yalın Bayes sınıflandırma ile k-en yakın komşu sınıflandırma yöntemleri istatistiksel sınıflandırıcılar olarak gruplanır. YSA yöntemlerinden olan Çok Katmanlı Algılayıcı (ÇKA)

ve Öğrenen Vektör Nicelme (LVQ: Linear Vector Quantization) de algılayıcı temelli sınıflandırıcılardan ikisidir.



Şekil 3. Farklı makine öğrenmesi sınıflandırma yöntemlerinin kullanım eğilimi

### 1.3.3. Sınıflandırıcı Performansının Değerlendirilmesi

Bir tahmin modeli geliştirildikten sonra modelin gelecek (geliştirme aşamasında karşılaşılmayan veriler) için nasıl bir sonuç vereceği önemli bir sorudur. Karar verici bir karar verme probleminin çözümü için kullandığı farklı yöntemlerin ve her bir yöntemle elde ettiği modellerin kullanılabilir olup olmadığını veya farklı yöntemleri birleştirerek oluşturduğu hibrit modelin daha iyi sonuç verip vermediğini belirlemek ister. Bu beklentilerin karşılanması modelin başarısının ölçülmesi ile sağlanır.

Sınıflandırma modellerinin başarısının belirlenmesinde en temel ölçüt doğru sınıfların oranını gösteren doğruluk (accuracy) oranıdır. Doğruluk oranı sınıflandırıcının güvenilirliğini gösterir. Bir model için hem eğitim hem de test veri kümesi için doğruluk ölçülebilir. Her iki kümede de yüksek doğruluğa sahip model tahmin başarısı yüksek bir sınıflandırıcıdır (Michie vd., 1994). Sınıflandırıcı başarısının belirlenmesinde 3 temel yöntem kullanılır: Basit Ayırma, Çapraz Doğrulama ve Birini Hariç Tut Yöntemleri.

İlk yöntem, girdi veri kümesini, eğitim ve test veri kümesi olarak iki bağımsız ve özgün kümeye ayırır. Eğitim girdi kümesi genellikle girdi veri kümesinin 2/3'si olarak seçilir ve geri kalan veri, model başarısının belirleneceği test veri kümesi olur. Çapraz doğrulamada ise girdi veri kümesi k boyutlu bağımsız ve özgün kümelere ayrılır. Kümelerin biri model

başarısını ölçmek için test veri kümesi olarak seçilirken kalanları girdi veri kümesi olarak kullanılır. Modelin genel başarısı tüm test veri kümesi başarılarının ortalamasıdır. Birini hariç tut yöntemi, çapraz doğrulama yönteminin özel bir halidir. Bu yöntemde test veri kümesi tek bir örnekten oluşmaktadır. Yöntem hesaplama yükünü artırsa da model başarısının diğerlerine göre daha doğru gösterilmesini sağlar (Kotsiantis, 2007).

İki sınıflandırıcının karşılaştırılmasında en sık başvurulan yöntem istatistiksel karşılaştırmadır. Aynı eğitim veri kümeleri kullanılarak eğitilen farklı iki sınıflandırıcının belirli test kümeleri üzerindeki tahmin başarıları arasındaki fark iki sınıflandırıcının birbirinden farklı olup olmadığını belirlemede kullanılır. Bu farkın varlığı iki model arasında farkın olmadığını söyleyen  $H_0$  hipotezinin doğruluğunu kontrol eden ikili bağımsız  $t$ -test ile gerçekleştirilir (Kotsiantis, 2007).

### 1.3.4. Sınıflandırıcı Performans Ölçütleri

#### 1.3.4.1. İkili Karşılaştırma Matrisi

Sınıflandırma modellerinin başarısının değerlendirilmesi amacıyla kullanılan temel performans ölçütü ikili karşılaştırma matrisi (confusion matrix) ya da kontenjans tablosu (tesadüf matrisi) olarak adlandırılır. Şekil 4'te gerçek sınıf değerleri ile öngörülen sınıf değerleri kullanılarak ikili sınıflandırma için oluşturulmuş kontenjans tablosu gösterilmektedir (Witten ve Frank, 2005).

		Gerçek Sınıflar	
		Pozitif	Negatif
Tahmin Edilen Sınıflar	Pozitif	Doğru Pozitif Sayısı (DP)	Yanlış Pozitif Sayısı (YP)
	Negatif	Yanlış Negatif Sayısı (YN)	Doğru Negatif Sayısı (DN)

Şekil 4. Karşılaştırma Matrisi

Değerlendirilen modelin pozitif ve negatifleri tahmin etme başarısı Doğru Pozitif ve Doğru Negatif oranları ile belirlenir. Doğru pozitif (isabet/rate) oranı Eşitlik (1) ve yanlış pozitif (hatalı alarm) oranı da Eşitlik (2) ile hesaplanır. Modelin doğruluğu (overall accuracy), Eşitlik (3)'teki gibi, doğru pozitif ve doğru negatif sayıları toplamının tüm örneklere oranı ile belirlenir (Olson ve Delen, 2008).

$$\text{Doğru Pozitif Oranı} = \frac{DP}{DP + YN} \quad (1)$$

$$\text{Doğru Negatif Oranı} = \frac{DN}{DN + YP} \quad (2)$$

$$\text{Doğruluk} = \frac{DP + DN}{DP + YN + DN + YP} \quad (3)$$

Sınıflandırma problemi iki sınıflı bir problem değilse, performans ölçütleri her bir sınıfa ait sınıflandırma oranları dikkate alınarak hesaplanır. Böyle bir durumda sınıf doğruluk oranları Eşitlik (4), model sınıflandırma doğruluğu Eşitlik (5) ile hesaplanır.

$$\text{Doğru Sınıf}_i \text{ Oranı} = \frac{\text{Doğru Sınıf}_i \text{ Sayısı}}{\text{Sınıf}_i \text{ Sayısı}} \quad (4)$$

$$\text{Model Doğruluğu} = \frac{\sum_{i=1}^n \text{Doğru Sınıf}_i \text{ Sayısı}}{\text{Toplam Örnek Sayısı}} \quad (5)$$

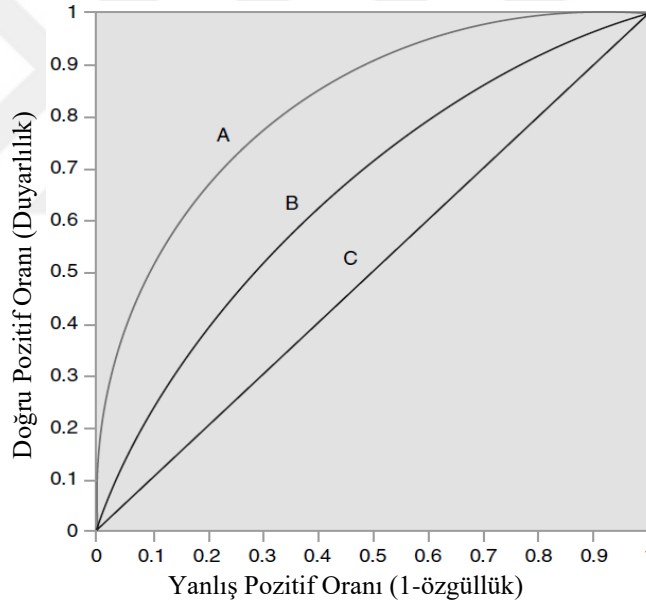
Yüksek doğruluk oranına sahip modellerin, negatif değer alan örnekleri tespit etme performansı (DN) düşük olabilir. Bu nedenle modellerin pozitif ve negatif örnekleri başarıyla belirleyebilme yeteneklerinin aynı anda ölçülmesine ihtiyaç duyulur. Böyle modelleri tespit etmek için model duyarlılığı ve kesinliğini dikkate alan F-ölçütü kullanılır. Duyarlılık, Pozitif olan örneklerin ne kadarının doğru olarak belirlenebildiğini gösteren DP oranına eşittir. Kesinlik ise, Pozitif olarak sınıflandırılan örneklerin ne kadarının gerçekte Pozitif olduğunu gösteren orandır ve Eşitlik (6) ile hesaplanır. Duyarlılık ve kesinlik oranlarının harmonik ortalaması olan F-ölçütünün hesaplaması Eşitlik (7)'de gösterilmiştir. Her iki oranın da yüksek değer alması durumunda yüksek değer alan F-ölçütü modelin performansının iyi olduğunu gösterir (Erdem, 2012; Coşkun ve Baykal, 2011).

$$\text{Kesinlik} = \frac{DP}{DP + YP} \quad (6)$$

$$F = \frac{2 * \text{duyarlılık} * \text{kesinlik}}{\text{duyarlılık} + \text{kesinlik}} \quad (7)$$

### 1.3.4.2. ROC Eğrisi

ROC (Receiver Operating Characteristics) eğrisi sınıflandırıcının sağladığı faydanın (Doğru Pozitif Oranı), neden olduğu kayba (Yanlış Pozitif Oranı) oranını görselleştirir. ROC eğrisi grafiğinde YP oranı (1-özgüllük) x ekseninde yer alırken y ekseninde DP (duyarlılık) oranı yer alır (Tomak ve Yüksel, 2009; Davis ve Goadrich, 2006). Şekil 5'te örnek bir ROC eğrisi grafiği görülmektedir (Olson ve Delen, 2008).



Şekil 5. ROC eğrisi

Duyarlılık (sensitivity), Pozitif değerlerin ne kadarının Pozitif olarak sınıflandırıldığını gösteren DP oranı, özgüllük (specificity) ise gerçekte Negatif olan değerlerin ne kadarının negatif olarak sınıflandırdığını gösteren DN oranıdır.

Grafikte, (0, 0) noktası, hiçbir *pozitif* sınıflandırmanın yapılmadığı durumu gösterir. Tümüyle pozitif sınıflandırmanın yapıldığı, negatif sınıflandırmanın olmadığı durum (1, 1) noktasına gerçekleşir. Mükemmel sınıflandırmanın yapıldığı durum ise (0, 1) noktasıyla

temsil edilir. ROC eğrisi grafiğinde bulunan iki noktadan sol-üst tarafta yer alan için daha yüksek DP ve daha düşük YP değerine sahip olduğu dolayısıyla daha iyi konumda olduğu söylenir (Fawcett, 2006).

ROC eğrisi, sınıflandırıcının pozitif ve negatif etiketli örnekleri doğru sınıflandırma gücünü tek bir performans ölçütüyle göstermesi nedeniyle tercih edilir. ROC eğrisi altında kalan alan (AUC: Area Under the Curve), sınıflandırıcının pozitif ve negatif örnekleri ne kadar başarı ile belirleyebildiğini göstermektedir. AUC değeri 0 ile 1 arasında değişir. Hatasız bir sınıflandırma yapan model için AUC değeri 1 olur. Doğru ve yanlış sınıfları rastgele olarak belirleyen bir sınıflandırma modelinde ROC eğrisi  $y = x$  eğrisi şeklinde oluşur ve AUC değeri 0,5 olur. 0,5 altında AUC değerine sahip sınıflandırma modelinin sınıflandırma yeteneğine sahip olduğu söylenemez. Şekil 5'te gösterilen A, B ve C ROC eğrilerine göre sınıflandırma başarısı en yüksek olan model A ile temsil edilen modeldir.

#### 1.4. Yapay Sinir Ağları

Yapay Sinir Ağları (YSA) insan beyninin öğrenme, birleştirme, uyarlama ve genelleştirme yeteneği ile karmaşık, doğrusal olmayan ve paralel dağılmış bilgi işleme sisteminin taklit edilmesiyle modellenen matematiksel hesaplama sistemleridir (Fausett, 1994). Bilgisayar yazılımları ile biyolojik sinir hücrelerinin benzetimi yapılarak uygulanan YSA, beynin bilgi işleme yönteminden esinlenen öğrenme süreci ile eğitilen bilgiyi toplama, hücreler arasındaki bağlantı ağırlıkları ile bu bilgiyi saklama ve genelleme yeteneğine sahip paralel dağılmış bir işlemcidir (Sağiroğlu vd., 2003). Problem çözümü için önceden tanımlı kurallar ve formüllere dayalı geleneksel hesaplama yöntemlerine alternatif olan yapay sinir ağları, açık çözüm kuralları ve formülleri olmayan problemler için öğrenilen çözümlerin kullanıldığı bir hesaplama yöntemidir (Rafiq vd., 2001).

Yapay sinir ağlarının en önemli özelliği; hatalı veya eksik değerler ya da belirsizlik içeriyor olsalar dahi örnekler üzerinden elde ettiği deneyimlerden öğrenmesi, öğrendiklerini değişen durumlara göre uyarlaması ve genelleme yeteneği kazanmasıdır. YSA'nın sahip olduğu, bir problemin girdi ve çıktıları arasındaki doğrusal ya da doğrusal olmayan ilişkiyi mevcut örneklerden öğrenip, yeni örnekler için öğrendiği bu bilgileri kullanarak çözümler üretebilme yeteneği, biyolojik sinir hücrelerinin taklididir (Rafiq vd., 2001).

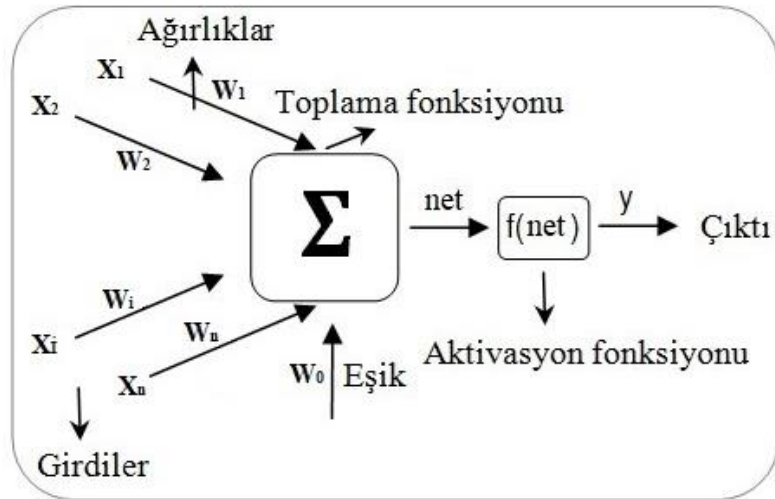
YSA, karmaşık işlemleri eş zamanlı olarak gerçekleştiren çok sayıda paralel dağılmış birbirine bağlı hücreden (işlem birimi) oluşur. Bir problem çözüm tekniği olan "böl ve

fethet” (Bar-Yam, 1997) mantığı ile her bir adım için ayrı hesaplama yapan bu işlem birimleri YSA’ya hızlı bilgi işleme yeteneğine kazandırır. YSA’nın bu yapısı onu, problemin her bir adımı için sadece sırayla hesaplama yapan makine öğrenmesi yöntemlerinden ayırır.

Yapay sinir ağları öğrenme, genelleme yapma, paralel işlem yapabilme ve hatayı tolere etme, yeni durumlara göre uyarlanabilme kabiliyetleri vb. gibi üstünlükleri sayesinde sistemlerin modellenmesinde ve denetiminde geniş uygulama alanı sahiptir.

#### 1.4.1. Yapay Sinir Hücresi ve Yapay Sinir Ağı

Yapay sinir ağlarının temeli, McCulloch ve Pitts’in geliştirdiği basit matematiksel hesaplama yapan, biyolojik sinir hücresinin yapay bir modeline dayanır (Bishop, 1994). Şekil 6 ile gösterilen ve nöron olarak adlandırılan bu ilk sinir hücresi modeli girdiler, ağırlıklar, toplama fonksiyonu, aktivasyon (transfer) fonksiyonu ve çıktıdan oluşur. Bu nöron  $x_1, \dots, x_n$  girdi verisini ağırlıklandırarak elde ettiği  $net = \sum_{i=1}^n x_i w_i + w_0$  toplam girdiyi uygun bir aktivasyon fonksiyonu  $f(.)$  ile sinir hücresinin çıktısı  $y = f(net)$ ’e dönüştürür.



Şekil 6. Yapay sinir hücresi

McCulloch ve Pitts’in geliştirdiği bu nöron, bağlantı ağırlıklarının güncellenmesini öngörmemiştir. Bu ağırlıkların güncellenmesi fikrinin oluşmasına ışık tutan Hebb’in biyolojik sinir hücrelerinin öğrenme mekanizmaları üzerine yaptığı çalışmalar YSA’nın

gelişiminde önemli aşamalardandır. Hebb'in biyolojik sinir hücreleri arasındaki bağlantıların güçlendirilmesi konusundaki tespitleri günümüzde kullanılan Hebb öğrenme kuralını ortaya çıkarmıştır. Buna göre eğer nöron bir girdi için isteneni üretebiliyorsa gelecekte benzer girdiler için de istenen sonucu üretebilme ihtimalini artıracak şekilde, tersi durumda da istenmeyen sonucu üretme ihtimalini azaltacak şekilde ağırlıklar güncellenir. Nörona öğrenme yeteneği kazandıran Hebb öğrenme kuralı Rosenblatt'ın, ağırlıkları istenen çıktı değerlerine göre güncelleyen algılayıcı (perceptron) adındaki en basit YSA modelini geliştirmesini sağlamıştır. Widrow ve Hoft ise ağırlıkların istenen çıktı ile nöronun girdiler için ürettiği çıktılar arasındaki hatayı dikkate alarak güncelleme yapan ADALINE'yi (ADAPtive LINear Element) geliştirmiştir. Minsky ve Papert'in doğrusal olmayan problemleri çözemediği eleştirisinde bulunduğu tek nöronlu modelleri Rumelhart ve arkadaşlarının geliştirdiği hata geri yayılım algoritması izlemiştir. Bu algoritma ve paralel kullanılan nöronlar çok katmanlı algılayıcı olarak adlandırılan YSA'ları ortaya çıkarmıştır (Bishop, 1994).

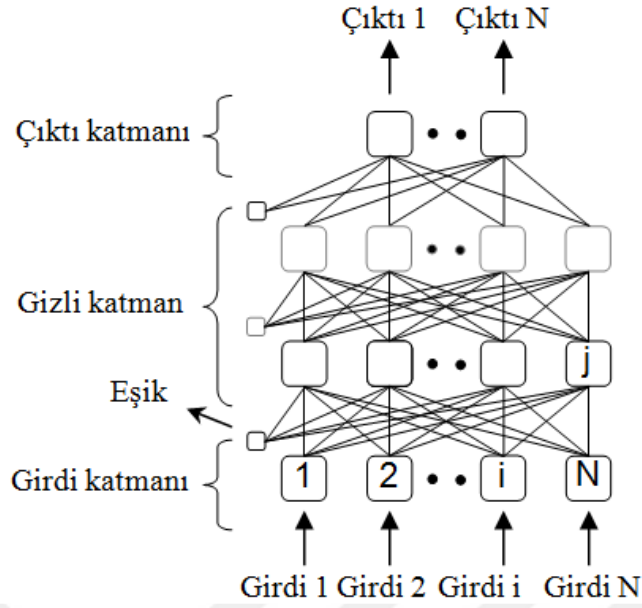
#### **1.4.2. Çok Katmanlı Yapay Sinir Ağları**

Doğrusal olmayan problemlerin çözümü için paralel işlem yapan birden fazla yapay nörona ihtiyaç duyulduğundan çok katmanlı ağ yapılar ortaya çıkmıştır (Öztemel, 2003). Şekil 7'de en basit çok katmanlı ağ yapısı, yapay nöronlar ve katmanlar arası ilişkiler gösterilmektedir (Ozsahin ve Murat, 2017).

Çok katmanlı bir yapay sinir ağı, her biri sonraki katmandaki hücrelerle çeşitli şekillerde bağlantılı yapay sinir hücrelerinin yer aldığı bir girdi katmanı, bir ya da daha fazla gizli katman (ara katman) ve bir çıktı katmanından oluşur. Bir yapay sinir ağındaki her bir nöron girdi, ağırlık, toplama fonksiyonu, aktivasyon fonksiyonu ve çıktıya sahiptir.

Girdi katmanı dışarıdan gelen verilerin YSA'ya alındığı katmandır. Gizli katman/katmanlar, girdi katmanını çıktı katmanına bağlar ve girdi katmanından gelen bilgileri işleyerek çıktı katmanına gönderir. Gizli katmanda, girdi veri kümesindeki örneklere karşılık ağırlık çıktı üretmesi için veri kümesindeki gizli örüntüler açığa çıkarılır. Çıktı katmanı, gizli katmandan gelen bilgileri işleyerek, ağırlık girdi veri kümesindeki örnekler için ürettiği çıktıyı dış dünyaya gönderir (Rafiq vd., 2001).





Şekil 7. Çok katmanlı bir yapay sinir ağı örneği

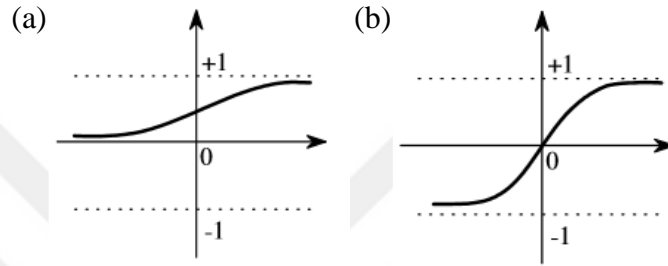
Girdi ( $x$ ), nörona dış ortamdan gelen verilerdir. Ağırlıklar ( $w$ ) nöronları birbirine bağlar ve dışarıdan alınan bilginin nöron üzerindeki etkisini gösterir. Ağırlığın büyüklüğü önemi ile doğru orantılı değildir. Bir ağırlığın değerinin sıfır olması o ağ için en önemli olay olabilir. Değerlerin artı veya eksi olması etkisinin pozitif (arttırıcı) veya negatif (azaltıcı) olduğunu, sıfır olması ise herhangi bir etkinin olmadığını gösterir (Öztemel, 2003). Toplama fonksiyonu nörona gelen girdileri genelde Eşitlik (8) ile gösterildiği gibi ilgili ağırlıklarla çarparak birleştirip nöronun toplam girdisini hesaplayan fonksiyondur. Bir yapay sinir ağındaki tüm nöronların aynı toplama fonksiyonunu kullanmaları gerekmez. Aktivasyon fonksiyonu, toplama fonksiyonuyla elde edilen net girdiyi işleyip nöronun bu girdiye karşılık çıktısını belirler (Sağiroğlu vd., 2003). Türevi alınabilir ve sürekli olan aktivasyon fonksiyonu bir nöron için tek bir çıktı üretir. Bu çıktı ağın çıktısı olabileceği gibi nöronun kendisinin veya sonraki bir nöronun girdisi de olabilir. Çok katmanlı algılayıcı ağlarda en sık kullanılan aktivasyon fonksiyonu gizli katmanda Eşitlik (9) ile gösterilen sigmoid, çıktı katmanında ise doğrusal aktivasyon fonksiyonudur.

$$net_i = \sum_{j=1}^n w_{ij}x_j + w_0 \quad (8)$$

$$f_i(net_i) = \frac{1}{1 + e^{-net_i}} \quad (9)$$

Burada  $x_j$   $i$ . nörona gelen  $j$ . girdiyi,  $w_{ij}$   $j$ . nöronla  $i$ . nöron arasındaki bağlantı ağırlığını ve  $w_0$  eşik değerini gösterir.  $n$   $i$ . nöronun bağlantılı olduğu ve girdilerini sağlayan nöronların sayısıdır.

Çok katmanlı algılayıcı ağında aktivasyon fonksiyonu ile hesaplanan nöron çıktısı ağın bağlantılarının üzerinden diğer nöronlara gönderilir. Aktivasyon fonksiyonu bir nöronun aktivasyon potansiyelini genellikle Şekil 8 (a) ve Şekil 8 (b)'deki gibi  $[0,1]$  veya  $[-1,1]$  aralığına normalize ederek YSA'nın yakınsamasını sağlar.



Şekil 8. Sigmoid (a) ve Hiperbolik Tanjant (b) aktivasyon fonksiyonları

YSA'yı oluşturan katmanların ve katmanlarda bulunan nöronların sayısı problemlerin yapısına göre farklılık gösterebilir. Bu sayıları belirlemek için belirli bir kural yoktur ve en uygun ağ yapısı çok sayıda deneme yapılarak belirlenir. Nöronlarda kullanılan fonksiyonların yanında katman ve nöron sayıları da ağın yapısını etkiler. Gizli katman ve gizli katmanlardaki nöron sayıları az olan ağlar sadece doğrusal yapıdaki problemlerde daha başarılı olurken karmaşık ve doğrusal olmayan örüntüleri ayırt edemeyebilir. Gizli katman ve gizli katmanlardaki yapay nöron sayılarının fazla olması ise ağın ezberlemesine neden olur ve genelleştirme yapmasını engeller (Haykin, 1994; Rafiq vd., 2001).

#### 1.4.2.1. Çok Katmanlı Algılayıcı

Aynı katmanda bulunan nöronların birbirleri ile bağlantılarının bulunmadığı, her bir nöronun sadece kendinden sonraki katmandaki tüm nöronları beslediği en sık kullanılan yapay sinir ağı modeli çok katmanlı algılayıcı (ÇKA)'dır. ÇKA'da bilgi akışı iki yönde gerçekleşir. İleri besleme (feed-forward) olarak adlandırılan ilk bilgi akışı yönü girdi katmanından çıktı katmanına doğrudur. Bu yönde ağa gelen bilgiler, nöronlar ve nöronlar arasındaki bağlantılar ile işlenerek ağın çıktısı hesaplanır. Nöronlar arasındaki bağlantı

ağırlıklarının güncellendiği ikinci bilgi akışı yönü ise çıktı katmanından girdi katmanına doğrudur. Ağın eğitiminin gerçekleştiği bu sürece hatanın geri yayılımı (back-propagation) denir. İleri beslemede, uygun aktivasyon fonksiyonları kullanarak gizli ve çıktı katmanlarındaki her bir nöronun çıktısı hesaplanır. Geri yayılımda ise öğrenme sürecinde ağa sunulan girdi için beklenen ve ağın ürettiği çıktı arasındaki farkı gösteren hata değeri her bir nöron arasındaki bağlantılara dağıtılır.

### 1.4.3. Yapay Sinir Ağlarında Öğrenme

Yapay sinir ağlarında öğrenme, oluşturulan bir ağ topolojisinin çeşitli parametrelerinin ağa gösterilen girdilere karşılık doğru çıktıları üretecek hale gelecek şekilde ayarlanmasıdır. Bir ağ yapısına karar verildikten sonra nöron ve eşik bağlantı ağırlıklarının en uygun değerlerinin belirlenmesi gerekir. Ağın ürettiği çıktı ile istenen çıktı arasındaki farkı en küçükleyecek bu en uygun değerlerin belirlenmesi sürecine eğitim denir. Sonuçları bilinen örneklerin ağa tek tek gösterilerek ağın öğrenme kuralına göre başlangıç ağırlıklarının güncellendiği eğitim süreci sonunda söz konusu problemi öğrenmiş bir yapay sinir ağı elde edilir.

Yapay sinir ağı eğitiminde en sık kullanılan öğrenme kuralı bir danışmanlı öğrenme yöntemi olan geri yayılım algoritmasıdır. Algoritma, her adımda bir örnek için gerçekleşen hatayı çıktı katmanından girdi katmanına doğru dağıtarak toplam hatanın en aza indirilmesini amaçlar.

ÇKA ağlarda kullanılan bir geri yayılım algoritması olan genelleştirilmiş delta öğrenme kuralı Hebb öğrenme kuralının geliştirilmiş halidir. Delta öğrenme kuralında hatalar, ileri besleme gerçekleştirilirken kullanılan bağlantılar üzerinden aktivasyon fonksiyonlarının türevi aracılığıyla geriye doğru yayılır. Ağdaki ağırlıklar ve aktivasyon fonksiyonlarının sonucu olarak gerçekleşen hata, hataların kareleri ortalaması dereceli azaltma (gradient descent) yöntemi kullanılarak en küçüklenir (Öztemel, 2003).

Çıktı nöronlarının her birinin ürettiği hata, sadece o hatanın oluşmasına neden olan nöron ve bağlantılara aktarılır. Böylece ara katmandaki işlem elemanlarının her biri toplam hatanın sorumlu olduğu kısmını içerir. Bu düzenleme tüm verilerin kodlanabileceği bir duruma ağın yakınsamasını sağlar (Öztemel, 2003).

Genelleştirilmiş delta öğrenme kuralına göre  $i$  nöronu ile  $j$  nöronu arasındaki ağırlığın iyileştirilmesi Eşitlik (10)'da gösterildiği gibi gerçekleştirilir.

$$w_{ij}^t = w_{ij}^{t-1} + (\lambda \delta_j z_i + \alpha \Delta w_{ij}^{t-1}) \quad (10)$$

Burada  $\lambda$ , ağırlıkların değişim miktarını belirleyen öğrenme katsayısı ve  $\alpha$  da yerel optimuma takılmayı önleyen momentum katsayısıdır.  $z_i$   $i$  nöronunun çıktısıdır.  $\delta_j$   $j$  nöronunun hata terimini gösterir. Kullanılan aktivasyon fonksiyonunun türevi ile belirlenen bu terim çıktı katmanı nöronu için Eşitlik (11) ve gizli katman nöronları için Eşitlik (12)'de gösterildiği gibi hesaplanır. Eşitlik (11)'de  $E_j$   $j$  nöronunun hatasıdır.

$$\delta_j = \frac{\partial g(z_j)}{\partial z_j} \cdot E_j \quad (11)$$

$$\delta_i = \frac{\partial g(z_i)}{\partial z_i} \sum \frac{\partial g(z_j)}{\partial z_j} w_{ij} \quad (12)$$

#### 1.4.4. Yapay Sinir Ağlarında Eğitim ve Test Etme

YSA örneklerden görererek ileri besleme ve geri yayılım algoritmaları ile öğrenir. Örneklerin tümünü mümkün olan en az hata ile öğrenecek ağırlık değerleri, her bir örnek ağa defalarca gösterilerek bulunur. Eğitim veri kümesindeki tüm örnekler için doğru çıktılar üretilinceye ya da toplam hata değeri belirli bir değerin altına ininceye kadar bu işlemler tekrarlanır.

Eğitimi belirlenen kriterlere uygun olarak tamamlanan ağ eğer eğitimi boyunca görmediği test veri kümesi örneklerini de yüksek başarı oranı ile tahmin edebiliyorsa model eğitiminin başarılı olduğu kararına varılır.

Eğitim veri kümesindeki bu örnekler eğer yeterli sayıda ve problem uzayını temsil edebilir homojenlikte olursa YSA'nın hata oranı azalırken probleme ait genelleme yeteneği artar. Ancak belirlenen bir ağ yapısı için olması gerekenden daha fazla örnek kullanılması durumunda öğrenme değil ezberleme gerçekleşebilir. Bu durum göz önüne alınırsa eğitim veri kümesindeki örnek sayısı için kullanılacak en fazla nöron sayısı Eşitlik (13) ile belirlenebilir (Ozsahin ve Murat, 2017).

$$T_{bağlantı} = (N_g + 1) \times N_{a1} + (N_{a1} + 1) \times N_{a2} + (N_{a2} + 1) \times N_{\zeta} \quad (13)$$

Burada  $T_{bağlantı}$  YSA modelindeki toplam bağlantı sayısı,  $N_g$ ,  $N_a$  ve  $N_ç$  ise sırasıyla girdi katmanı, ara katman ve çıktı katmanındaki nöron sayılarıdır.

### 1.5. Öğrenen Vektör Niceleme (LVQ)

Danışmanlı öğrenme yöntemlerinden biri olan Öğrenen Vektör Niceleme (LVQ: Learning Vector Quantization) bir yapay sinir ağı modeli olarak Kohonen tarafından tanıtılmıştır (Kohonen, 1986). Bir sınıflandırma algoritması olan LVQ farklı sınıfları temsil eden referans (prototip) vektörü adı verilen nöronlardan oluşmaktadır. Referans (prototip) vektörleri, girdi uzayının Voronoi bölümlendirmesini sağlayan sınıf hiper-düzlemlerini tanımlar (Nova ve Estévez, 2014). LVQ, referans vektörleri aracılığıyla sınıf etiketleri bilinmeyen girdiyi sınıflandırmayı amaçlar. Sınıflandırma yeteneği kazanmış bir LVQ ağı, girdiyi en iyi şekilde sınıflandıracak referans vektörlerine sahiptir. LVQ ağının eğitimi bu referans vektörlerinin belirlenmesini sağlamaktır.

Sınıfları en iyi temsil eden referans vektörlerini belirleme ve sınıf etiketi bilinmeyen girdiyi sınıflandırma “en yakın komşu” kuramına göre gerçekleştirilir. Sınıf etiketi belirlenirken referans vektörüne olan metrik uzaklık değeri kullanılır. En yakın komşu kuralı Cover ve Hart’ın önerdiği k-en yakın komşu sınıflandırıcı yöntemi ile benzerdir. LVQ, öğrenme sürecine ihtiyaç duymayan k-en yakın komşu algoritmasından sınıflandırma yöntemiyle farklılaşır. LVQ sınıflandırma sırasında sınıf etiketli örneklerin tamamını kullanmak yerine sadece belirli referans vektörlerini kullanır.

LVQ referans vektörlerini, girdi uzayını girdilerin olasılık dağılımlarına göre ayırarak belirler. LVQ, girdi uzayındaki her bir örneğin rastgele olarak modele sunulduğu, Hebbian öğrenme kuralına dayalı tekrarlı bir öğrenme algoritmasıdır. Doğru sınıflandırma yapma yeteneği kazanmış referans vektörlerinin sayısını artırmak ilkesi ile çalışır (Verleysen vd., 1993). Desino, Kohonen’e göre iyi eğitilmiş bir LVQ ağının eşit kazanma olasılığına sahip referans vektörlerinden oluştuğunu ifade etmiştir (DeSieno, 1988).

#### 1.5.1. En Yakın Vektör Sınıflandırma

Sınıfı bilinmeyen bir örnek  $\xi$ , LVQ ile en yakın referans vektörüne göre sınıflandırılır. Referans vektörleri, girdi vektörleri ile aynı uzayda tanımlanır ve her sınıf için en az bir

referans vektörü tanımlanmış olmalıdır.  $c \in \{1, \dots, C\}$  sınıf değerlerinden biriyle etiketlenen  $n$  boyutlu referans vektörleri Eşitlik (14)'te ifade edilmiştir.

$$W = \left\{ (w_j, c(w_j)) \subset \mathbb{R}^n \times \{1, \dots, C\} \right\} \quad j = 1, \dots, l \text{ ve } l \geq C \quad (14)$$

En yakın referans vektörü, her bir referans vektörü ile örnek arasındaki uzaklık ölçütü  $d(w, \xi)$  ile elde edilir. En bilinen uzaklık ölçütü Eşitlik (15) ile gösterilen Minkowski uzaklık ölçütünün özel bir türü olan Öklit (Euclidean) mesafesidir. “Kazanan hepsini alır (winner-takes-all)” stratejisi ışığında uzaklık ölçütüne göre seçilen referans vektörü örneğin sınıf değerini belirler. Eşitlik (16), en küçük uzaklık ölçütüne sahip referans vektörünün sınıf değerinin, örneğin sınıf değeri olarak atandığını göstermektedir (Schneider, 2010).

$$d(w, \xi) = \sqrt{\sum_{i=1}^n |\xi^i - w^i|^2} \quad (15)$$

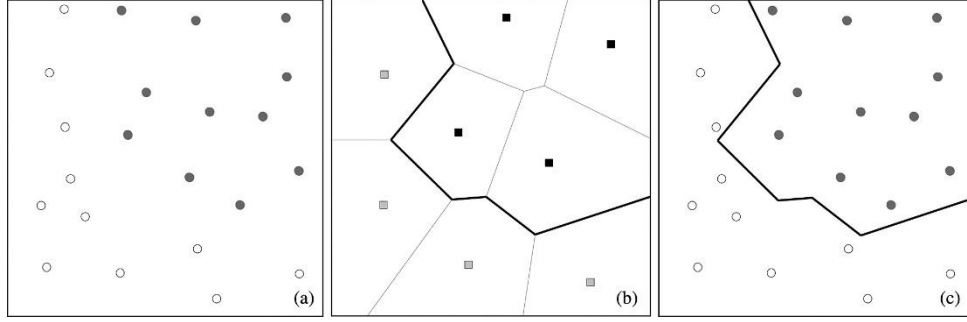
$$\xi \leftarrow c(w_i) \mid w_i = \min_j d(w_j, \xi) \quad (16)$$

Her referans vektörü girdi uzayının bir alt kümesi olan alıcı alanlara sahiptir. Merkezini, girdi verisine diğer vektörlerden daha yakın olan referans vektörünün temsil ettiği alıcı alan  $R_i$  Eşitlik (17) ile ifade edilir. Burada  $X$ ,  $N$  boyutlu girdi veri kümesini göstermektedir.

$$R_i = \{ x \in X \mid \forall w_j (j \neq i) \rightarrow d(x, w_i) < d(x, w_j) \} \quad (17)$$

### 1.5.2. LVQ Öğrenme Algoritmaları

Başlangıçta girdi veri kümesindeki verilere göre rastgele belirlenen referans vektörleri eğitim sürecinde, her bir girdiyi doğru sınıflandırıp sınıflandıramadığına göre güncellenir. LVQ ile sınıflandırmada öğrenme, girdi veri kümesindeki örneklerin sınıf değerlerini doğru olarak belirleyecek referans vektörü kümesi  $W$ 'deki her bir vektörün girdi uzayındaki konumunun bulunmasını amaçlar. İdeal  $W$  kümesi, girdi uzayını Şekil 9'daki gibi Voronoi bölümlendirmesi ile sınıflara göre ayırır (Lloyd vd., 2007).



Şekil 9. (a) 2 sınıf ve 22 örnekten oluşan örnek uzayı, (b) Referans nöronlar ve Voronoi bölümleri, (c) Sınıflandırma sonucuna göre sınıf sınırları

Öğrenme süreci  $N$  boyutlu sınıf değerleri bilinen  $X = \{(x_i, y_i) \in \mathbb{R}\}$  girdi veri kümesi ile gerçekleştirilir. Burada  $x = (x_1, \dots, x_D) \in \mathbb{R}^D$   $D$  boyutlu bir örneği  $y \in \{1, \dots, C\}$  ise örneğin sınıfını göstermektedir.

LVQ eğitiminde kullanılan en sade öğrenme algoritması, girdi veri kümesindeki her örneğin birer birer modele sunulduğu online öğrenme algoritmasıdır (Schneider, 2010). Her tekrarda (iterasyon), modele sunulan örnek için en yakın referans vektör ile yapılan sınıflandırmanın doğruluğuna göre, bir ya da birden fazla referans vektöründe değişiklik yapılır.  $t$  anında güncelleme yapılacak referans vektörlerinin mevcut değerleri Eşitlik (18)'de gösterildiği gibi değiştirilir.

$$w_i^t = w_i^{t-1} + \Delta w_i^t = w_i^{t-1} + \eta(t) f(w_i^{t-1}, x_t, y_t) (x_t - w_i^{t-1}) \quad (18)$$

Denklemden  $\eta$  öğrenme katsayımı gösterir ve zaman içerisinde, giderek yaklaşılacak ideal referans vektörlerinin büyük değişikliklerle bozulmasını önlemek amacıyla, sıfır olacak şekilde azaltılır (Öztemel, 2003). Öğrenme girdi veri kümesindeki tüm örnekler belli sayıda referans vektörüyle temsil edilinceye kadar devam eder.

### 1.5.3. LVQ1

Kohonen'in önerdiği ilk LVQ modeli olan LVQ1, yalnızca en yakın mesafede yer alan referans vektörünün güncellenmesini önerir (Kohonen, 1986). Referans vektörünün nasıl güncellenmesi gerektiği Eşitlik (19) ile ifade edilmiştir. Eğitim sürecinde her iterasyonda modele gösterilen örneğin sınıf değeri ile örneğe en yakın uzaklıktaki referans vektörünün

sınıf değeri eşleşiyorsa referans vektörü örneğe yaklaştırılır. Eğer sınıf değerleri eşleşmiyorsa, sonraki iterasyonlarda benzer örneklerin söz konusu referans vektörü tarafından sınıflandırılmasını önlemek amacıyla, referans vektörü örnekten uzaklaştırılır.

$$\begin{aligned} w_B^t &= w_B^{t-1} + \eta(t)(x_t - w_B^{t-1}) \quad \text{eğer } c(w_B^t) = y \\ w_B^t &= w_B^{t-1} - \eta(t)(x_t - w_B^{t-1}) \quad \text{eğer } c(w_B^t) \neq y \end{aligned} \quad (19)$$

#### 1.5.4. LVQ2.1

Kohonen'in geliştirdiği LVQ2.1 algoritması, LVQ2 algoritması gibi Bayesyan karar sınırlarına çok yakın olan iki referans vektörünü güncelleştirmeyi önerir (Kohonen, 1990). Tanımlanmış bir aralığa düşen, kazanan (winner) ve ikinci (runner-up) olarak adlandırılan iki referans vektörü farklı iki sınıfa aittir. Örnek ile aynı sınıf değerine sahip olan referans vektörü örneğe yaklaştırılırken, örnek ile aynı sınıftan olmayan diğer referans vektörü örnekten uzaklaştırılır.

$y$  sınıfına ait bir  $x$  eğitim veri kümesi örneğine en yakın iki vektör  $w_i$  ve  $w_j$  ise; vektörlerin içine düşmesi gereken aralık Eşitlik (20) ile ifade edilir.

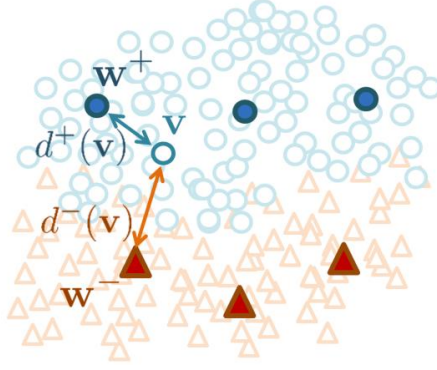
$$\min \left( \frac{d(x, w_i)}{d(x, w_j)}, \frac{d(x, w_j)}{d(x, w_i)} \right) > s \text{ ve } s = \frac{1 - \omega}{1 + \omega} \quad (20)$$

Burada  $c(w_i) \neq c(w_j)$ , farklı iki sınıfa ait vektörlerdir ve  $c(w_i) = y$ 'dir. Eşitlikte  $\omega$ , aktif bölgenin belirlenmesini sağlayan aralık genişliğini ifade eder. Bu aralığa düşen referans vektörleri de Eşitlik (21)'de ifade edildiği gibi güncelleştirilir.

$$\begin{aligned} w_i^t &= w_i^{t-1} + \eta(t)(x_t - w_i^{t-1}) \\ w_j^t &= w_j^{t-1} - \eta(t)(x_t - w_j^{t-1}) \end{aligned} \quad (21)$$

Böylece Şekil 10'da bir örneği gösterilen (Kaden vd., 2014) en yakın doğru ve en yakın yanlış referans vektörleri belirlendikten sonra en yakın kazanan vektör örneğe yaklaştırılır ve en yakın kaybeden vektör örnekten uzaklaştırılır (Schneider, 2010).





Şekil 10. En yakın doğru ve yanlış referans nöronları

LVQ diğer prototip temelli yöntemler gibi basit ve yorumlanabilir uygulamaya sahip olması sayesinde çeşitli uygulama alanları bulmakta ve diğer sınıflandırma yöntemlerine tercih edilmektedir. LVQ ile öğrenme yönteminde herhangi bir değişime ihtiyaç olmadan çok sınıflı veri kümeleri sınıflandırılabilir. Sınıflara ait Bayesyan sınırları temsil eden referans vektörleri girdi verisindeki eksik değerlerden etkilenmeden sadece bilinen özniteliklere göre güncellenebilirler. Referans vektörleri girdi uzayındaki örneklerle aynı özellikte olduğu için veri yapısı hakkında da bilgi sunabilir ve diğer sınıflandırma yöntemleri olan YSA ve DVM'deki gibi kara kutu özelliğine sahip değildir. LVQ'nun bir diğer tercih nedeni de hesaplama yükünün DVM'ye kıyasla daha az olmasıdır. LVQ'da hesaplama yükü belirli sayıdaki referans vektörüne bağlı iken DVM'nin hesaplama yükü girdi verisindeki örnek sayısı ile orantılı destek vektörüne göre değişmektedir (Nova ve Estévez, 2014).

LVQ, yorumlanabilir referans vektörlerden oluşması ve çok sınıflı problemlere doğrudan uygulanabilir olması sayesinde telekomünikasyondan sağlığa, endüstriden robotiğe çok geniş bir alanda uygulanmış ve geliştirilmiştir. Yüz ifadelerinin tanınması (Bashyal ve Venayagamoorthy, 2008), el hareketlerinin sınıflandırılması (Wu vd., 2009), istatistiksel kalite kontrol kartı sınıflandırma (Pham ve Oztemel, 1994), kanserli doku tespiti (Dieterle vd., 2003), internet trafik tanımlama (Yiran ve Suoping, 2010), uydu görüntülerinin sınıflandırılması (Mills, 2011), hibrit araçlarda sürüş tipi sınıflandırma (He vd., 2012), bitki tohumu sınıflandırma (Broséus vd., 2011) LVQ kullanılarak yapılan sınıflandırma çalışmalarıdır.

Nova ve Estevez çalışmasında LVQ algoritmaları gelişimini üç ayrı başlık altında incelemiştirler: Standart LVQ algoritmaları, maliyet fonksiyonu temelli LVQ algoritmaları ve olasılık temelli LVQ algoritmaları (Nova ve Estévez, 2014).

Standart LVQ algoritmaları, Hebbian öğrenme kuralını kullanır ve sezgisel öğrenme temellidir. İlk LVQ ağı geliştirilerek Bayesyan karar sınırlarına daha hızlı ve iyi yaklaşım sağlayan yeni algoritmalar önerilmiştir. Kohonen, LVQ'daki sınır değerlerdeki sınıflandırma hatalarını önlemek için aynı anda birden fazla referans vektöründe güncelleme yapan LVQ2.1 ağını (Kohonen vd., 1988; Kohonen, 1990) geliştirmiştir. Desino aynı referans vektörünün üst üste kazanıp diğerlerinin güncellenmelerinin önüne geçmesini engellemek için kazanma olasılığına dayalı bir cezalandırma yöntemi önermiştir (DeSieno, 1988). Öztemel, referans vektörleri güncellemesini her defasında iki vektörü güncellemek amacıyla global ve yerel kazanan vektörleri belirleyerek yapan LVQ-X modelini geliştirmiştir (Öztemel, 1992).

LVQ algoritmaları, en uygun çözüme yakınsamayı artırmak amacıyla Bayesyan karar sınırları marj maksimizasyonu sağlanarak da iyileştirilmiştir (Boubezoul vd. 2007; Schneider vd. 2009a; Schleif vd. 2011; Hammer vd. 2014). Maliyet fonksiyonuna sahip ilk model genelleştirilmiş LVQ ağıdır (Sato ve Yamada, 1995). Modele eklenen maliyet fonksiyonu en yakın komşu belirlenirken hesaplanan mesafelere dayalı göreceli uzaklık parametresini optimize etmeyi amaçlar. Sınıflandırıcı düzlemleri iyileştirmeyi ve hatayı düşürmeyi amaçlayan bu maliyet fonksiyonu ile en iyi sınıflandırma modellerinin tespiti hedeflenir.

Olasılık temelli olarak ayrılan diğer LVQ algoritmalarının (Hammer vd., 2011; Hofmann vd., 2013; Schneider vd., 2009b) ilk örneği yeni bir öğrenme kuralı öneren robust LVQ'dur (Seo ve Obermayer, 2003). Robust LVQ sınıfların dağılımlarını dikkate alan istatistiksel amaç fonksiyonu kullanarak doğru sınıfların olasılığının yanlış sınıfların olasılığına oranını maksimum yapmayı amaçlar.

Rastgele başlangıç değerleri ile eğitimi gerçekleştirilen LVQ sınıflandırma, bu özelliği nedeniyle başlangıç değerlerine duyarlı olduğundan, bu duyarlılığı azaltmaya amaçlayan çalışmalar da yapılmıştır. Dinamik olarak değişen referans vektörü sayısı kullanımı (Vlassis ve Likas, 2002; Hammer vd., 2005), yeni referans vektörü ekleme (Jirayusakul ve Auwatanamongkol, 2007), harmonik ortalama uzaklık ve minimum uzaklığı birleştiren bir mesafe değişimi yöntemi (Qin ve Suganthan, 2005) ve genelleştirilmiş LVQ (Sato ve Yamada, 1995) gibi yöntemler ile sınıflandırmada yerel optimuma takılmanın engellenebildiği gösterilmiştir.

## 1.6. Gri İlişkisel Analiz

Deng tarafından geliştirilen Gri Sistem Teorisi (GST) (Julong, 1989) sistemleri, haklarında sahip olunan bilginin derecesine göre beyaz, gri veya siyah olarak tanımlar. İçerdiği bilginin tamamı bilinen sistemleri beyaz sistemler, içerdiği bilgi hiç bilinmeyen sistemleri siyah sistemler olarak niteleyen GST, içerdiği bilginin bir kısmına sahip olunan sistemleri ise gri sistemler olarak niteler (Wei, 2011). Belirsizliğin sayısallaştırılmasını sağlayan GST, stokastik veya bulanık yöntemlerle açıklanamayan belirsizliğe sahip sistemlerin davranışlarını, sınırlı sayıda veri yardımı ile tahmin etmek düşüncesiyle ortaya çıkmıştır (Liu ve Lin, 2006).

GİA, faktörler ve onları oluşturan değişkenler arasındaki girift ve örtük ilişkileri inceleyen, Gri Sistem Teorisi araçlarından biridir. GİA bir referans vektör ile alternatif vektörler arasındaki ilişkiyi analiz eder. Vektörleri oluşturan kriterlerin (özniteliklerin) benzerlik ve farklılıklarını matematiksel ifadelerle açıklayan GİA, vektörler arasındaki gri ilişkilerin derecelendirmesi için kullanılan bir yöntemdir.

Gri ilişkisel analiz, ilişkilerin gri sayılarla gösterimi, gri katsayının hesabı, gri ilişki derecelerinin belirlenmesi ve sıralama adımları ile yapılır.

$n$  kriter (öznitelik) ve  $m$  alternatiften (örnekten) oluşan girdi verisindeki  $x_i$  örneklerinin  $x_0$  referans vektörü ile gri ilişkisinin hesaplanmasını gösteren süreç aşağıdaki gibidir.

$$x_i = (x_i(1), x_i(2), \dots, x_i(n)), \quad i = 1, \dots, m \quad (22)$$

$$x_0 = (x_0(1), x_0(2), \dots, x_0(n)) \quad (23)$$

Gri ilişkilerin oluşturulması: Eşitlik (23) ile ifade edilen  $x_0$  referans vektörü, özniteliklerin en küçük, en büyük veya bu değerler arasında ideal bir başka değer olması gerektiğine göre belirlenir. Çok kriterli karar verme problemlerinde genellikle her bir özneliğin en küçük veya en büyük değerleri kullanılır. Bu tez kapsamında  $x_0$  referans vektörü, sınıflandırma amacıyla kullanılacağından ideal değerler kullanılarak belirlenmiştir. Farklı özellikteki özniteliklerin standartlaştırılması anlamına gelen gri ilişkilerin oluşturulması ideal bir değere göre Eşitlik (24)'teki gibidir:

$$x_i'(k) = 1 - \frac{|x_i(k) - x_0(k)|}{\max[\max x_i(k) - x_0(k), x_0(k) - \min x_i(k)]} \quad (24)$$

Burada,  $x_i'(k)$   $i$ . vektörün  $k$ . özniteliği için normalize değeri,  $x_i(k)$  normalize edilmeden önceki değeri,  $x_0(k)$  ideal referans vektör değeri ve  $\max x_i(k)$  ile  $\min x_i(k)$  de vektör özniteliklerinin en büyük ve en küçük değerleridir.

Gri ilişkisel katsayının hesaplanması: Gri ilişkisel katsayı, referans vektör ile diğer vektörlerin her bir öznitelik bazında bire bir ( $x_0'(k)$  ile  $\forall x_i'(k)$ ,  $i = 1, \dots, m$ ) benzerlikleri olarak düşünülebilir (Huang vd., 2008; Bayramoglu ve Hamzacebi, 2016). Her bir öznitelik için Gri ilişkisel katsayı hesaplaması Eşitlik (25)'teki gibidir.

$$\gamma(x_0'(k), x_i'(k)) = \frac{\Delta_{\min} + \zeta \Delta_{\max}}{\Delta_{0i}'(k) + \zeta \Delta_{\max}} \quad (25)$$

Eşitlikte  $\Delta_{0i}'(k)$ ,  $x_0'(k)$  ve  $x_i'(k)$  arasındaki mutlak farkı gösterir.

$\Delta_{\min} = \min_{\forall i} \min_{\forall k} \Delta_{0i}'(k)$  ve  $\Delta_{\max} = \max_{\forall i} \max_{\forall k} \Delta_{0i}'(k)$ , her bir referans öznitelik ile ona karşılık gelen öznitelikler arasındaki mutlak farkların en büyük ve en küçük değerleridir.  $\zeta$  farklılaştırma katsayısı  $\Delta_{0i}'(k)$  ile  $\Delta_{\max}$  arasındaki farkı dengelemek için kullanılır ve  $(0, 1]$  aralığında bir değer alır (Huang vd., 2008).

Gri İlişkisel Derece (GİD)'nin hesaplanması: GİD alternatif vektörlerin özniteliklere göre evrensel bir değerlendirmesini sunar (Bayramoglu ve Hamzacebi, 2016). GİD, referans vektör ve diğer alternatif vektörlerin her biri arasındaki ikili benzerliğin ölçüsünü belirlemek için kullanılır. Daha büyük GİD değeri daha fazla benzerlik anlamını taşır. GİD 0 ile 1 arasında değer alır (Huang vd., 2008). Tüm öznitelikleri referans vektöre özdeş olan alternatif vektör ile referans vektör arasındaki GİD 1'e eşittir. GİD Eşitlik (26) ile hesaplanır.

$$\gamma(x_i, x_0) = \sum_{k=1}^n w_k \gamma(x_0'(k), x_i'(k)), \quad i = 1, \dots, m \quad (26)$$

$\gamma(x_i, x_0)$   $i$ . alternatif vektörün GİD'dir.  $w_k$  vektörleri oluşturan özniteliklerin ağırlıklarını gösterir. Özniteliklerin eşit öneme sahip olduğu durumda  $w_k = \frac{1}{n}$  olur.

GİA, derecelendirme, sınıflandırma ve özellikle karar verme problemlerinde kullanılmaktadır. İş kazaları riski gruplandırma (Zhou vd., 2015), elektrik aktarıcı hatalarının tespiti (Liwei Zhou vd., 2017), uydu görüntüleri sınıflandırma (Zhang vd., 2014), görüntü üzerinden cinsiyet belirleme (Kaya vd., 2017), EKG aritmi sınıflandırma (Lin, 2008; Lin, 2006), kumaş yüzey hatalarının tespiti (Kuo ve Su, 2003), hareket sinyali algılayıcı portatif el (Du vd., 2010), betonarme yapılarda çatlak algılama (Lai vd., 2012), tütün yaprağı kalitesi belirleme (Mao vd., 2010), iş hastalıklarına göre mesleklerin sınıflandırılması (Akay, 2011), yazılım kalitesi sınıflandırma (Chiu, 2009) GİA kullanımına örnek çalışmalardır.

GİA bazı diğer yöntemlerle birlikte de kullanılmıştır. Aktivasyon fonksiyonu olarak gri ilişkisel analizin kullanıldığı tek katmanlı algılayıcı ile sınıflandırma performansı geliştirilmeye çalışılmıştır (Hu, 2008). Bir diğer birleştirme kendini örgütleyen haritalar algoritmasında referans vektörlerin güncellenmesi için gri ilişkisel katsayıların kullanımı ile gerçekleştirilmiştir. Bu birleştirme ile referans vektörü, ilişkinin derecesine göre daha fazla ya da az güncelleme yapılarak iyileştirilmektedir (Hu vd., 2002). Gri ilişkisel analiz, parçacık sürü optimizasyonu ve genetik algoritma gibi sezgisel yöntemlerle birleştirilmiş ve öznitelik ağırlıklarının belirlenmesi için kullanılmıştır (Huang vd., 2008; Lin ve Tzeng, 2010).

GİA'nın diğer yöntemlerle birleştirildiği güncel bir çalışma da Yeh ve arkadaşlarının LVQ ile zaman serisi sınıflandırması amacıyla gerçekleştirdiği çalışmalarıdır (Yeh vd., 2017). Yeh ve arkadaşları LVQ yönteminde benzerlik hesaplama için (GİA) yöntemini kullanmıştır. Çalışmada zaman serileri Fourier serilerine dönüştürüldükten sonra [0, 1] aralığına normalize edilerek LVQ ağına girdi olarak sunulmuştur.

Bu tez çalışmasında LVQ yöntemi ile birlikte Gri LVQ yöntemi de kullanılmıştır. Kullanılan Gri LVQ yöntemi, gerçekleştirilen girdi normalizasyonu ve başlangıç ağırlığı belirleme yöntemiyle Yeh vd. tarafından uygulanan LVQ sınıflandırmasından farklılaşmaktadır.

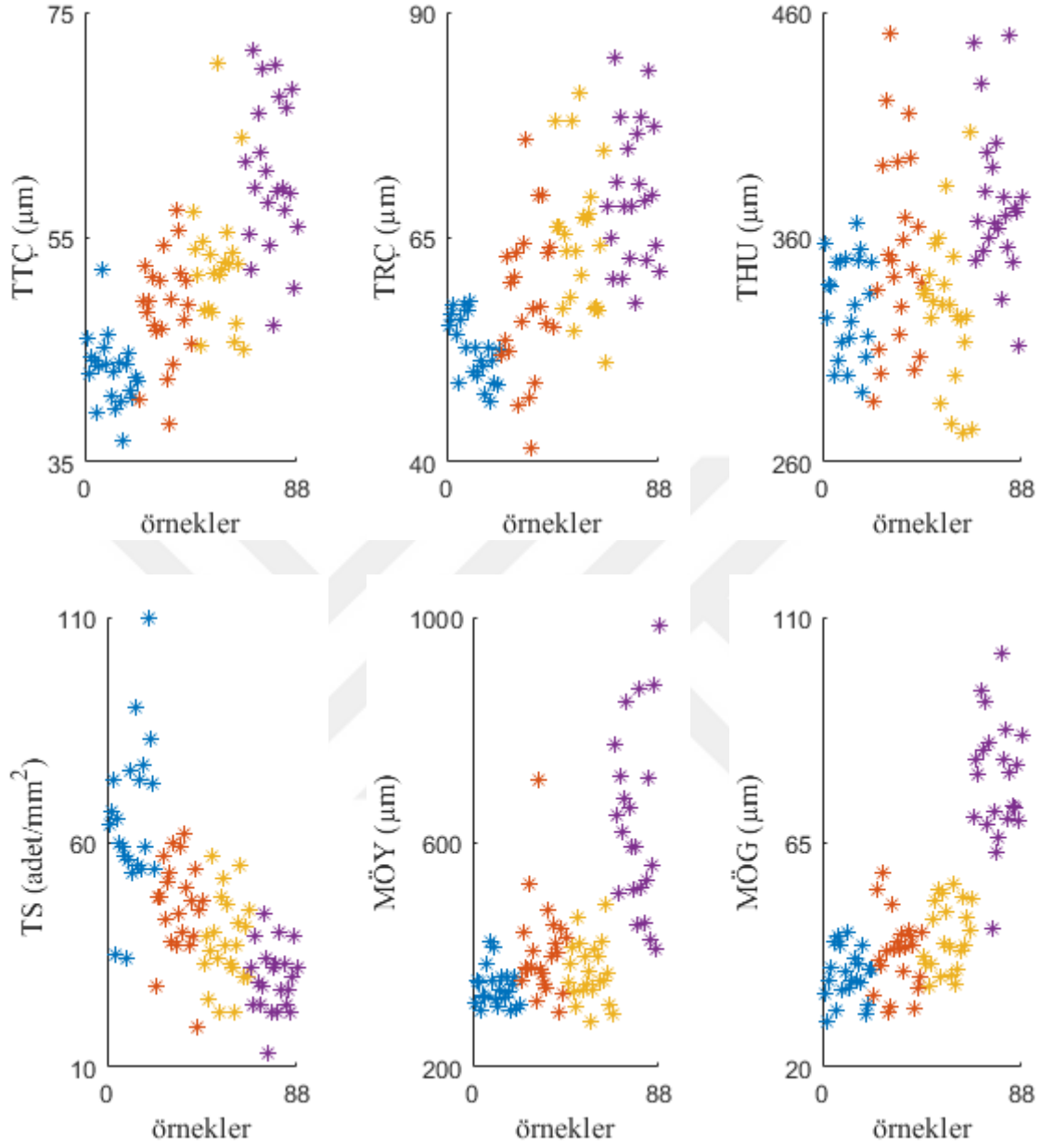
## 2. UYGULAMA

Bu tez çalışması odun örneklerinin belirli türlerden hangisine ait olduğunun belirlenmesi amacıyla makine öğrenmesi yöntemi olan yapay sinir ağları kullanılarak sınıflandırıcı modeller geliştirmek amacıyla yapılmıştır. Bu doğrultuda Birtürk'ün doktora tezinde (Birtürk, 2011) paylaştığı Akçaağaç (*Acer L.*) cinsine ait çeşitli türlerin anatomik özellik verileri kullanılmıştır. Söz konusu veriler ışığında Akçaağaç cinsine ait türlerin ayırımını yapabilmek için veride var olan örüntülerin tespitini yapacak Öğrenen Vektör Niceleme ve Çok Katmanlı Algılayıcı YSA yöntemleri kullanılmıştır. Odun türü sınıflandırma uygulamasında Öğrenen Vektör Niceleme yönteminin iyileştirilmesi amacıyla da yeni bir yöntem olan ve Gri İlişkisel Analiz ile benzerlik hesaplaması yapan Gri LVQ yöntemi kullanılmıştır.

Çalışmada farklı sınıflandırma yöntemlerinin karşılaştırılabilir olması için türlere ait örnek sayısı eşit alınmış ve her türden yeterli sayıda verinin var olmasını sağlamak amacıyla 4 tür seçilmiştir: *Acer campestre*, *Acer cappadocicum*, *Acer platanoides*, *Acer trautvetteri*. İlk iki tür ikişer farklı alt türe ait örnekler içerirken diğer iki tür ise herhangi bir alt türden örnek içermemektedir. Veri kümesi toplam 88 örnek içermekte ve her bir türe ait 22 örnek bulunmaktadır. Çalışmada kullanılan veriler Ek Tablo 1'de, dağılımlarını gösteren grafikler de Şekil 11 ve Şekil 12'de verilmiştir.

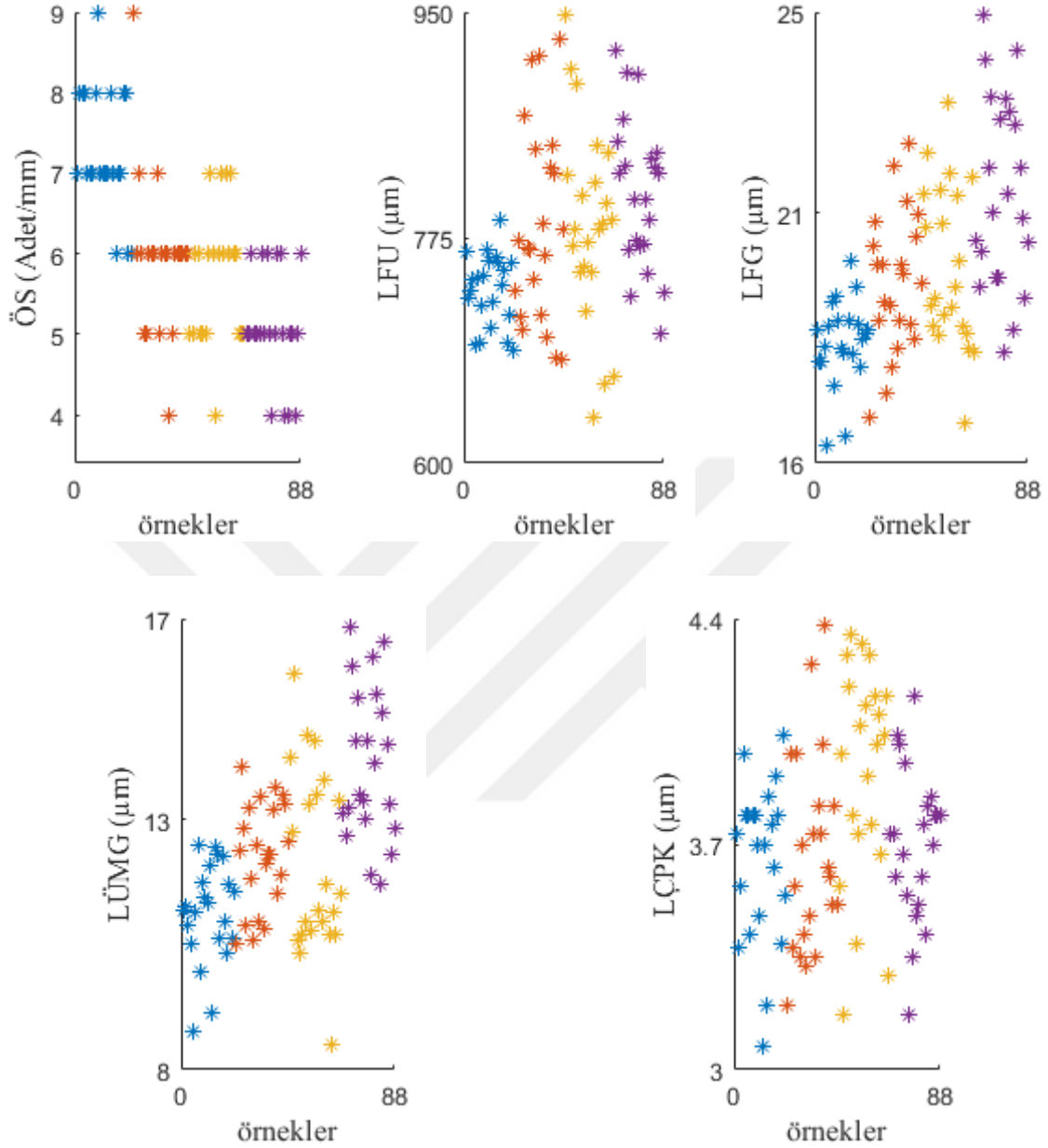
Odun anatomisine ait 11 farklı özellik kullanılarak sınıflandırma modelleri geliştirilmeye çalışılmıştır. Şekil 11 ve Şekil 12'de bu özellikler: Trahe Teğetsel Çap (TTÇ), Trahe Radyal Çap (TRÇ), Trahe Hücre Uzunluğu (THU), Trahe Sayısı (TS), Mültiseri Özışımı Yüksekliği (MÖY), Mültiseri Özışımı Genişliği (MÖG), Özışımı Sayısı (ÖS), Mültiseri Özışımı Genişliği (Hücre) (MÖGh), Lif Uzunluğu (LFU), Lif Genişliği (LFG), Lümen Genişliği (LÜMG), Lif Çeper Kalınlığı (LÇPK) ve her bir örneğe karşılık özelliklerin dağılımı gösterilmiştir. Grafiklerde mavi değerler *Acer campestre* (Sınıf 1), kırmızı değerler *Acer cappadocicum* (Sınıf 2), sarı değerler *Acer platanoides* (Sınıf 3) ve mor değerler de *Acer trautvetteri* (Sınıf 4) türü odun örneklerini göstermektedir. Söz konusu özellikler odun numunelerinin mikroskobik analizi ile elde edilen görüntüler üzerinden ölçülmüştür. Şekil 13, Birtürk'ün çalışmasında çeşitli anatomik özelliklerin ölçümü için kullandığı mikroskobik görüntülere örnektir.

\* *Acer campestre* \* *Acer cappadocicum* \* *Acer platanoides* \* *Acer trautvetteri*



Şekil 11. 88 Akçağaç örneği için ölçülen anatomik özellikler. Trahe Teğetsel Çap (TTC), Trahe Radyal Çap (TRÇ), Trahe Hücre Uzunluğu (THU), Trahe Sayısı (TS), Mültiseri Özışını Yüksekliği (MÖY), Mültiseri Özışını Genişliği (MÖG).

\* *Acer campestre* \* *Acer cappadocicum* \* *Acer platanoides* \* *Acer trautvetteri*

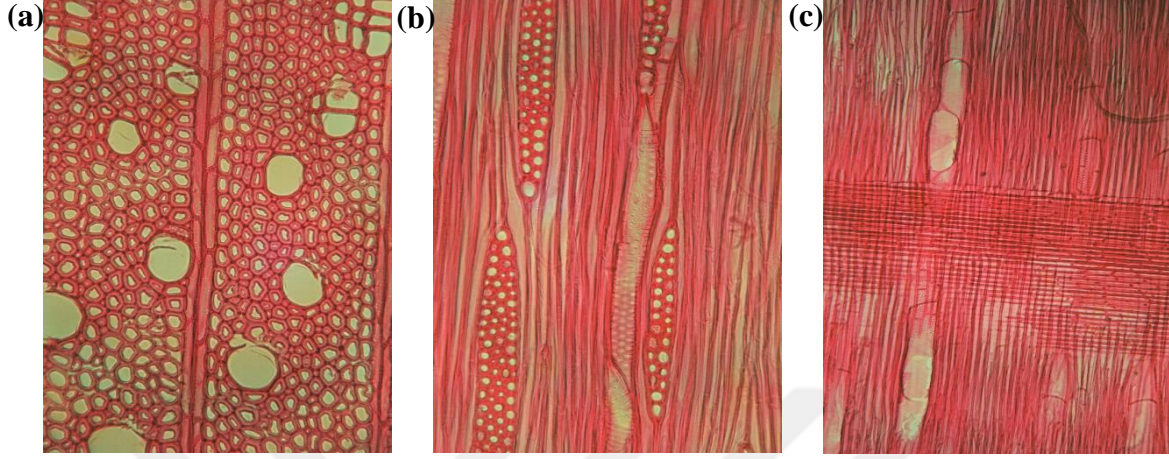


Şekil 12. 88 Akçaağaç örneği için ölçülen anatomik özellikler. Özışını Sayısı (ÖS), Mültiseri Özışını Genişliği (Hücre) (MÖGh), Lif Uzunluğu (LFU), Lif Genişliği (LFG), Lümen Genişliği (LÜMG), Lif Çeper Kalınlığı (LÇPK)

YSA sınıflandırıcı modelleri geliştirilirken ait oldukları odun türleri (sınıf değerleri) bilinen örneklerden oluşan veri kümesi eğitim ve test veri kümesi olarak iki bölüme ayrılmıştır. Veri kümesinin yaklaşık olarak %30'u olan test veri kümesi 24 örnek ile oluşturulurken eğitim veri kümesi 64 örnek ile oluşturulmuştur. Modellerin kıyaslanması ve



geliştirilen Gri LVQ modelin geçerliliğinin ortaya konması amacıyla rastgele olarak oluşturulmuş 10 farklı veri kümesi kullanılmıştır.



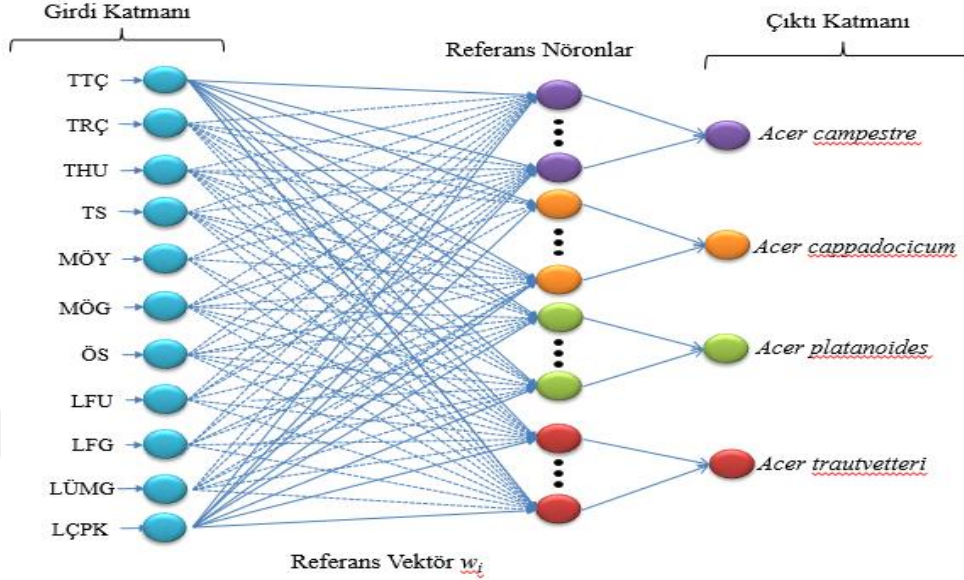
Şekil 13. (a) EK, Libriform liflerle çevrili traheler, paratraheal odun parenşimi ve küme şeklinde traheler, (b) RK, Homoselüler özışını, (c) TK, Vasküler traheid ve uzun libriform lifler

## 2.1. LVQ Sınıflandırma Modeli

LVQ yöntemi ile sınıflandırma modelleri geliştirilirken LVQ1 (learnlv1) algoritması kullanılmıştır. LVQ ağı yapısı belirlenirken her bir sınıfı temsil edecek şekilde en az 4 referans nöron ve en çok da 24 referans nöron denenmiştir. Girdi katmanı, girdi vektörünü oluşturan odun öznelik sayısına denk olarak 11 nörondan, çıktı katmanı ise her biri bir sınıfa etiketleme yapan 4 çıktı nöronundan oluşmaktadır. Kazanan çıktı nöronunun çıktısı 1 iken diğer nöronların çıktısı 0 olur. Her bir referans nöronu kendisine bağlı ağırlık değerleri ile ait olduğu sınıfı temsil etmektedir. Eğitim ve sınıflandırma esnasında girdi ile referans vektörleri arasındaki benzerlik Öklid mesafesi ile hesaplanmış ve en küçük Öklid uzaklığına sahip referans nöron kazanan nöron olarak belirlenmiştir. En iyi ağ yapısı deneme yanılma yoluyla bulunmuştur. Şekil 14 bu yöntemle performansı değerlendirilen bir LVQ ağı modelini göstermektedir.

LVQ ağı ara katmandaki referans nöronu sayısı dört olduğunda, her bir referans nöronu çıktı katmanındaki dört çıktı nöronundan sadece biri ile eşleştirilmiş ve böylece her bir sınıf için bir referans nöronu oluşturulmuş olur. Ara katmandaki referans nöronu sayısı

dörtten fazla olduğunda ise çıktı katmanındaki dört nöronun her biri en az bir referans nöronu ile eşleştirilir.



Şekil 14. LVQ ağ modeli

LVQ ağı eğitiminde girdi veri kümesi ağa gösterilmeden önce Eşitlik (27)'de ifade edilen normalizasyon fonksiyonu ile  $[-1, 1]$  aralığına normalize edilmiştir. Referans nöronları başlangıç ağırlıkları normalize edilmiş her bir özneliğin en küçük ve en büyük değerlerine göre rastgele olarak belirlenmiştir.

$$x' = \frac{(y_{maks} - y_{min}) * (x - x_{min})}{(x_{maks} - x_{min})} + y_{min} \quad (27)$$

Eşitlikte  $y_{maks}$  ve  $y_{min}$ ,  $x$  girdisinin dönüştürüldüğü aralığın sınır değerlerini ve  $x'$ 'in normalize edilmiş değerini gösterir.

LVQ algoritması ağ eğitimi sırasında, kazanan referans nöronuna göre ağırlıklar güncellenirken doğru ağırlıklardan uzaklaşmayı önleyen en iyi  $\eta$  değerinin belirlenebilmesi için 0,5 ile 0,01 arasındaki değerler denenmiş ve her farklı  $\eta$  değerinin 100 iterasyon ile gerçekleştirilen eğitim sonunda sifıra çok yakın bir değere ulaşması sağlanmıştır.

Modellerin performanslarının karşılaştırılmasında “Sınıflandırıcı Performans Ölçütleri” başlığı altında anlatılan performans yöntemlerden doğruluk oranı, modellerin genel performanslarının belirlenmesi ve birbirleriyle kıyaslanması amacıyla, ikili

karşılaştırma matrisi ve ROC eğrileri de her bir yöntem için en iyi modellen belirlenmesi amacıyla kullanılmıştır.

## 2.2. Gri LVQ1 Sınıflandırma Modeli

Gri LVQ1 ağında, girdi vektörü ile referans nöronun temsil ettiği vektör arasındaki benzerliğin Öklid uzaklığı ile belirlendiği LVQ ağından farklı olarak, iki vektör arasındaki benzerlik Gri İlişkisel Analiz (GİA) yöntemi ile belirlenir. Öklid uzaklığı hesaplamasında özniteliklerin sadece ikili karşılaştırması yapılırken GİA ile özniteliklerin bütün değerleri aynı anda dikkate alınarak benzerlik hesaplaması yapılır. Girdi vektörünün referans vektörlerine olan benzerliklerinin referans vektörleri kombinasyonundan etkilendiği düşüncesiyle kullanılan bu yöntem ile benzerliğin farklı şartlar altındaki değerinin belirlenmesi hedeflenmiştir.

LVQ ağının eğitimindeki sürecin aynı şekilde tekrarlandığı Gri LVQ1 eğitiminde, GİA sürecinde normalizasyon gerçekleştirildiğinden girdi veri kümesi normalize edilmeden ağa gösterilmiştir. Başlangıç referans vektörleri normalize edilmemiş özniteliklerin en küçük ve en büyük değerleri arasında olacak şekilde rastgele olarak oluşturulmuştur. Girdi, referans ve çıktı nöronu sayıları LVQ1 ağında denenen sayılarla aynıdır. Gri LVQ1 ağı eğitiminde kazanan referans nöronu, girdi vektörü ile referans vektörleri arasındaki gri ilişki derecesine göre belirlenmiştir. Gri LVQ ağı, girdi vektörünü ideal değer olarak kabul ederek ağı oluşturan referans nöronların temsil ettiği vektörlerin bu ideal değerle olan gri ilişkilerinin belirlenmesi mantığını temel alır. Buna göre girdi vektörü ile en ilgili referans vektörü gri ilişki derecesi en yüksek olan vektördür. Gri LVQ1 ağının eğitiminde referans vektörlerin güncellenmesi LVQ1 ağının anlatıldığı bölümde verilen Eşitlik (19) ile gerçekleştirilmiştir. Ağırlıklar güncellenirken kullanılan öğrenme oranı değişimi Eşitlik (28) ile yapılmıştır.

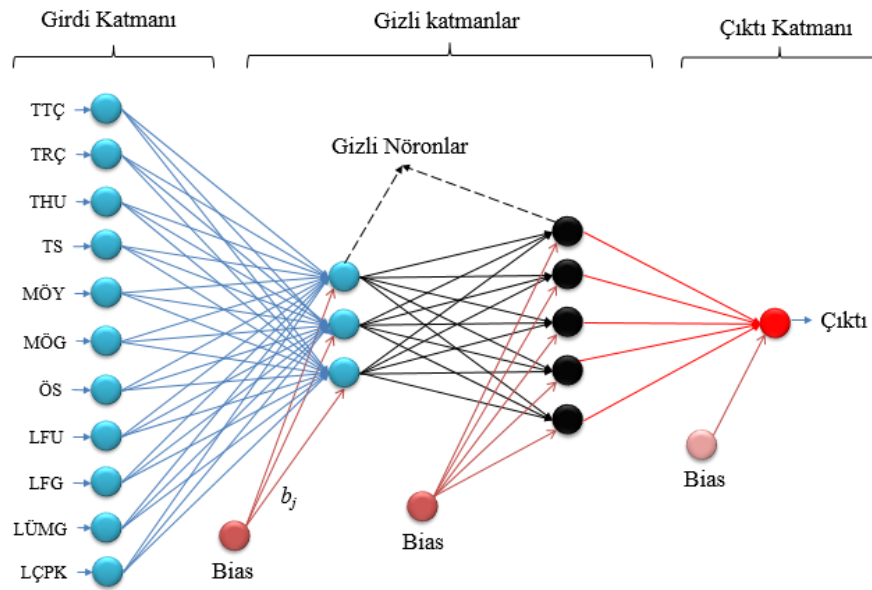
$$\eta_{t+1} = \eta_t * \left(1 - \frac{t}{T}\right) \quad (28)$$

Eşitlikte  $t$  mevcut iterasyon sayısını ve  $T$  maksimum iterasyon sayısını gösterir.

### 2.3. ÇKA Sınıflandırma Modeli

Oluşturulan 10 farklı dizilime sahip veri setinin her biri için en iyi ağ yapısı ve parametrelerinin belirlenmesi için farklı modeller yine deneme yanılma yoluyla belirlenmiştir. ÇKA modellerin sınıflandırma başarıları, kıyaslanabilir olması amacıyla LVQ ağlarında olduğu gibi doğruluk oranı ile hesaplanmıştır.

Şekil 15, çalışmada odun sınıflandırma amacıyla geliştirilen ÇKA sınıflandırma modellerinin genel bir gösterimidir. ÇKA ağı da LVQ ağlarında olduğu gibi 11 girdi nöronundan oluşmaktadır. Tek bir çıktı nöronuna sahip ÇKA ağı bir ya da iki gizli katmanlı olarak denenmiştir. Gizli katmanlardaki nöron sayıları “Yapay Sinir Ağlarında Eğitim ve Test Etme” başlığı altında verilen Eşitlik (13) dikkate alınarak eğitim veri kümesinde yer alan 64 örneği öğrenecek bağlantıyı barındıracak şekilde belirlenmiştir. Buna göre geliştirilen ÇKA modelleri, tek gizli katmanlı ise en fazla 4 gizli nörondan oluşurken iki gizli katmanlı modeller ise en fazla 1-16, 2-9, 3-5 ve 4-2 nörondan oluşmuştur.



Şekil 15. ÇKA ağ modeli

Sınıflandırıcı ÇKA eğitiminde genellikle sınıflandırma problemlerinde kullanılan Resilient ileri besleme ve geri yayılım (trainrp) algoritması kullanılmıştır. Gizli katman nöronlarında hiperbolik tanjant sigmoid aktivasyon (tansig) fonksiyonu kullanılırken çıktı katmanı nöronlarında doğrusal aktivasyon fonksiyonu (purelin) kullanılmıştır. Performans

fonksiyonu olarak Eşitlik (29) ile belirtilen MSE (Hata Kareleri Ortalaması) seçilmiştir. Denklemden  $t_i$ ,  $a_i$  ve  $n$  sırasıyla, hedeflenen çıktıyı, ağıın ürettiği çıktıyı ve eğitim kümesi veri sayısını göstermektedir.

$$MSE = \frac{1}{n} \sum_{i=1}^n (t_i - a_i)^2 \quad (29)$$

ÇKA ağı eğitiminde de girdi veri kümesi LVQ ağında olduğu gibi Eşitlik (27) ile normalize edildikten sonra ağı gösterilmiştir.

ÇKA modellerin çıktı değerleri, doğrudan sınıf değerleri değil sınıf değerlerini temsil eden tam sayı değerlerine yakın rasyonel sayılardır. Sınıflandırma başarısının hesaplanması için modellerin girdiler için ürettiği çıktılar sınıfları temsil eden en yakın tam sayı değerine yuvarlanmıştır.

### 3. BULGULAR

Sınıflandırma modelleri MATLAB R2016b programı kullanılarak geliştirilmiştir. LVQ ağlarından Gri LVQ1 sınıflandırma modelinin kodu elle yazılarak hazırlanmıştır. LVQ ve ÇKA modelleri için ise MATLAB YSA araç kutusu aracılığıyla oluşturulan kod kullanılmıştır. Oluşturulan bu kodlar ile ağ mimarisinin tasarımı, eğitilmesi ve optimizasyonu gerçekleştirilmiştir. Geliştirilen sınıflandırıcıların performanslarının karşılaştırılabilmesi amacıyla kurulan modeller mümkün olduğunca aynı şartlarda çalıştırılmıştır.

Her iki LVQ ağı da girdilerin normalize edilerek ağa sunulma yöntemi ve dolayısıyla başlangıç ağırlıkları hariç aynı koşullara göre modellenmiştir. İterasyon sayısı, öğrenme katsayısı değişimi, maksimum referans nöronu sayısı her iki yöntem için de aynı belirlenerek en iyi modelin aranması sağlanmıştır. ÇKA ağı için de girdiler ağa normalize edilerek sunulmuştur. Ancak ÇKA ağ yapısı ve eğitimi LVQ ile aynı olmadığından ortak şartlar altında çalıştırılmamıştır.

Tablo 1, Tablo 2 ve Tablo 3 üç yöntem için elde edilen sınıflandırıcı modeller için ağ yapısı ve parametreleri ile eğitim ve test veri kümesi doğruluk (%) oranlarını göstermektedir.

Tablo 1. ÇKA sınıflandırıcıları için ağ yapıları, eğitim ve test performansları

Veri Kümesi No	Nöron Sayısı		MSE		Doğruluk Oranı (%)	
	Ara Katman 1	Ara Katman 2	Eğitim	Test	Eğitim	Test
1	4	1	0,0522	0,0798	93,750	95,833
2	1	6	0,1019	0,2806	87,500	83,333
3	3	2	0,0570	0,1796	95,313	87,500
4	3	3	0,0499	0,1349	95,313	91,667
5	4	2	0,0498	0,1080	95,313	91,667
6	3	5	0,0329	0,1508	96,875	87,500
7	3	2	0,0834	0,0908	89,063	91,667
8	3	5	0,0288	0,0851	96,875	91,667
9	4	2	0,0370	0,1824	95,313	91,667
10	2	8	0,0750	0,1591	93,750	87,500

Tablo 2. LVQ sınıflandırıcıları için referans nöron sayıları, öğrenme oranları, eğitim ve test performansları

Veri Kümesi No	Ağ Parametreleri		Doğruluk Oranı (%)	
	Öğrenme Oranı	Referans Nöron Sayısı	Eğitim	Test
1	0,34	19	90,625	91,667
2	0,16	15	89,063	87,500
3	0,19	15	90,625	87,500
4	0,23	21	90,625	95,833
5	0,22	13	93,750	91,667
6	0,04	23	92,187	91,667
7	0,16	11	93,750	91,667
8	0,04	17	90,625	91,667
9	0,37	20	87,500	87,500
10	0,28	12	90,625	87,500

Tablo 3. Gri LVQ1 sınıflandırıcıları için referans nöron sayıları, öğrenme oranları, eğitim ve test performansları

Veri Kümesi No	Ağ Parametreleri		Doğruluk Oranı (%)	
	Öğrenme Oranı	Referans Nöron Sayısı	Eğitim	Test
1	0,10	21	89,063	91,667
2	0,04	20	84,375	87,500
3	0,16	24	89,063	91,667
4	0,10	18	89,063	95,833
5	0,10	22	90,625	87,500
6	0,28	16	89,063	91,667
7	0,05	19	89,063	95,833
8	0,10	23	87,500	95,833
9	0,40	16	87,500	87,500
10	0,16	22	93,750	87,500

Tablo 1 ÇKA modellerinin oldukça düşük MSE değerleri ile eğitildiğini göstermektedir. ÇKA modelleri tek bir çıktı nöronu ile oluşturulduklarından sınıf değerlerini temsil eden 1, 2, 3 ve 4 sayılarını ya da bu sayılara çok yakın ondalık sayıları üretmiştir. Düşük MSE değerleri modellerin çıktı değerine çok yakın değerler ürettiğini göstermektedir.

LVQ ve Gri LVQ1 sınıflandırıcılardan seçilen uygun modellerin öğrenme oranları ve referans nöronlarının gösterildiği Tablo 2 ve Tablo 3 incelendiğinde LVQ modellerinin daha az sayıda referans nöron ile sınıflandırma yaptığı görülmektedir.

Tablo 4 her üç yöntemin eğitiminde kullanılan 10 veri kümesi için eğitim ve test performanslarını bir arada göstermektedir. Tablodaki performans değerleri her bir veri kümesi ve yöntem için seçilen en uygun modelin ulaştığı eğitim ve test veri kümeleri sınıflandırma doğruluğu değerlerini göstermektedir.

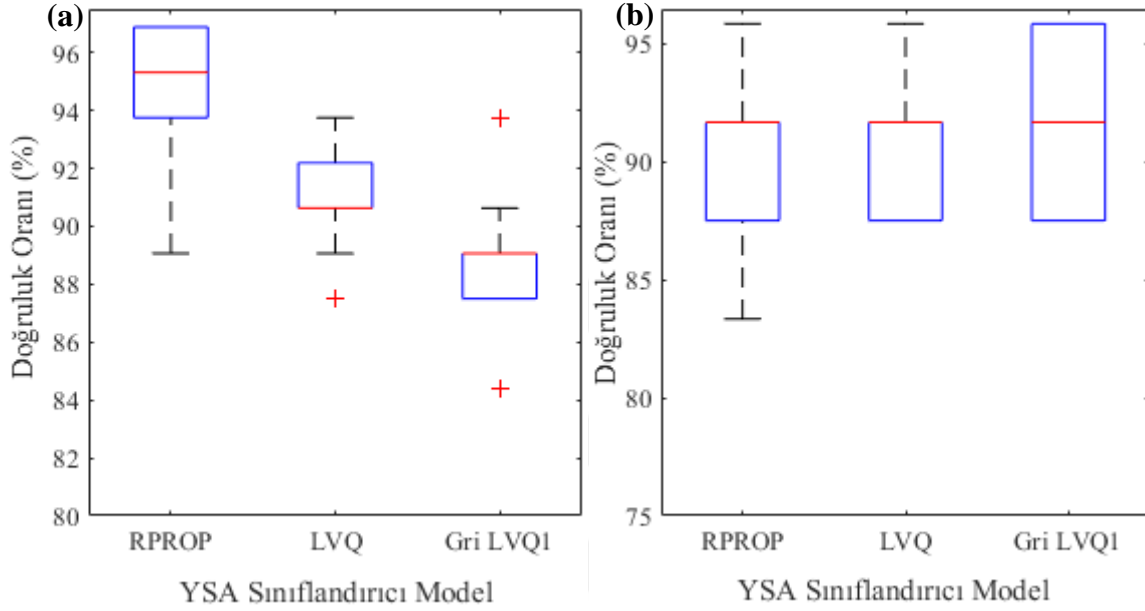
Tablo 4'e göre eğitim performansı ÇKA sınıflandırıcı modellerinde diğer iki modele göre yüksek olmuştur. LVQ sınıflandırıcı hem eğitim hem de test veri kümesi için aynı denilebilecek sınıflandırma başarısına sahip olurken Gri LVQ1 yöntemi diğer iki yönteme göre test sınıflandırma doğruluğu ile bir derece daha iyi başarı göstermiştir. ÇKA modeller, eğitim için en yüksek başarıya sahip olmalarına rağmen test performansları aynı oranda yüksek olmamıştır. Gri LVQ1, ÇKA ve LVQ modellerinin her üçü ile de sınıflandırma da %95,83 başarıya ulaşılabilmiş yani 24 örneğin 23'ünü doğru olarak sınıflandırmıştır.

Tablo 4. 10 veri kümesi için seçilen uygun modellerin eğitim ve test başarıları (%)

Veri Kümesi No	ÇKA Modeli		LVQ Modeli		Gri LVQ1 Modeli	
	Eğitim Başarısı	Test Başarısı	Eğitim Başarısı	Test Başarısı	Eğitim Başarısı	Test Başarısı
1	93,750	95,833	90,625	91,667	89,063	91,667
2	87,500	83,333	89,063	87,500	84,375	87,500
3	95,313	87,500	90,625	87,500	89,063	91,667
4	95,313	91,667	90,625	95,833	89,063	95,833
5	95,313	91,667	93,750	91,667	90,625	87,500
6	96,875	87,500	92,187	91,667	89,063	91,667
7	89,063	91,667	93,750	91,667	89,063	95,833
8	96,875	91,667	90,625	91,667	87,500	95,833
9	95,313	91,667	87,500	87,500	87,500	87,500
10	93,750	87,500	90,625	87,500	93,750	87,500
<b>Ortalama</b>	<b>93,907</b>	<b>90,000</b>	<b>90,938</b>	<b>90,417</b>	<b>88,907</b>	<b>91,250</b>



Şekil 16 (a) ve Şekil 16 (b) üç modelin eğitim ve test performanslarının gösterildiği kutu grafikleridir. Şekillerden de görüldüğü üzere Gri LVQ1 modeli test başarıları medyanı diğer YSA sınıflandırıcılar ile aynı olmuştur ancak test başarıları medyanının her iki tarafında eşit oranda gerçekleşmiştir. LVQ ve ÇKA sınıflandırıcılarının her ikisinde test başarıları medyan altında daha fazla iken ÇKA uç değerde düşük bir test başarıları elde edilmiştir.



Şekil 16. Seçilen sınıflandırıcı modellerin (a) eğitim doğrulukları ve (b) test doğrulukları kutu grafikleri

Modeller Tablo 4'te verilen test başarılarına göre değerlendirildiğinde Gri LVQ1 modeli daha yüksek performansa sahip olduğu görülmektedir. Ancak hem eğitim hem de test başarıları birlikte değerlendirildiğinde ise en başarılı sınıflandırıcı modeller ÇKA'lar olmuştur. Tablo 5 üç yöntemin eğitim ve test başarılarının ortalamalarını göstermektedir.

Çalışmada kullanılan 3 farklı YSA sınıflandırma yönteminin başarılarının kıyaslanması için istatistiksel testlere başvurulmuştur. Odun türü ayrımı için kullanılan bu yöntemlerin Tablo 5'te verilen ortalama sınıflandırma doğruluk oranları arasındaki farklarının anlamlı olup olmadığını belirlemek amacıyla SPSS 22 paket programında Bağımsız-Örnek t-Test yapılmıştır.

Ortalama sınıflandırma doğruluk oranları arasındaki farkların istatistiksel olarak anlamlılığının testi yapılmadan önce her yöntem için ortalama sınıflandırma doğruluk oranlarının normal dağılıp dağılmadığı test edilmiştir. Şekil 17'de verilen ve küçük örneklem

hacmi için kullanılan Shapiro-Wilk normallik testi sonuçlarına göre üç modele ait ortalama doğruluk oranlarının dağılımı, normal dağılımdan anlamlı bir farklılık göstermemektedir.

Tablo 5. 10 veri kümesi için seçilen uygun modellerin eğitim ve test başarıları ortalamaları (%)

Veri Kümesi için Ortalama Başarı			
Veri Kümesi No	ÇKA Modeli	LVQ1 Modeli	Gri LVQ1 Modeli
1	94,792	91,146	90,365
2	85,417	88,282	85,938
3	91,407	89,063	90,365
4	93,490	93,229	92,448
5	93,490	92,709	89,063
6	92,188	91,927	90,365
7	90,365	92,709	92,448
8	94,271	91,146	91,667
9	93,490	87,500	87,500
10	90,625	89,063	90,625
<b>Ortalama</b>	<b>91,953</b>	<b>90,677</b>	<b>90,078</b>

#### Tests of Normality

Sınıflandırıcı	Kolmogorov-Smirnov			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
Ort. Doğruluk Oranı						
ÇKA	,212	10	,200	,848	10	,055
LVQ	,190	10	,200	,913	10	,304
GRI LVQ1	,255	10	,065	,906	10	,253

Şekil 17. YSA sınıflandırıcıları başarı ortalamaları normallik testi sonucu

Normal dağıldıkları istatistiksel olarak tespit edilen ÇKA, LVQ ve Gri LVQ1 modelleri doğruluk oranları ortalamaları arasında anlamlı bir farkın olmadığını belirten  $H_0$  hipotezi ile farkların istatistiksel olarak anlamlı olduğunu iddia eden  $H_1$  hipotezleri, bağımsız örnek t-test ile 0,05 anlamlılık düzeyinde test edilmiştir. Şekil 18 ve Şekil 19’da verilen t-test sonuç tablolarına göre anlamlılık düzeyleri 0,05’ten büyük olduğundan ÇKA, LVQ ve Gri LVQ1 modellerinin 10 veri kümesi için ortalama doğruluk oranları arasındaki farkların istatistiksel olarak anlamlı olmadığını söyleyen  $H_0$  hipotezleri kabul edilmiştir. Buna göre

her üç yöntemden herhangi birinin söz konusu odun türlerini sınıflandırmada diğer bir yönteme üstün olmadığı sonucuna varılmaktadır.

Group Statistics					
	Sınıflandırıcı	N	Mean	Std. Deviation	Std. Error Mean
Ort. Doğruluk Oranı	ÇKA	10	91,9535	2,74640	,86849
	LVQ	10	90,6774	2,04973	,64818

Independent Samples Test			
		Ort. Doğruluk Oranı	
		Equal variances assumed	Equal variances not assumed
Levene's Test for Equality of Variances	F	,149	
	Sig.	,704	
t-test for Equality of Means	t	1,177	1,177
	df	18	16,652
	Sig. (2-tailed)	,254	,256
	Mean Difference	1,27605	1,27605
	Std. Error Difference	1,08370	1,08370

Şekil 18. Ortalama doğruluk oranına göre ÇKA ve LVQ başarıları t-test sonucu

Group Statistics					
	Sınıflandırıcı	N	Mean	Std. Deviation	Std. Error Mean
Ort. Doğruluk Oranı	ÇKA	10	91,9535	2,74640	,86849
	GRI LVQ1	10	90,0784	2,08489	,65930

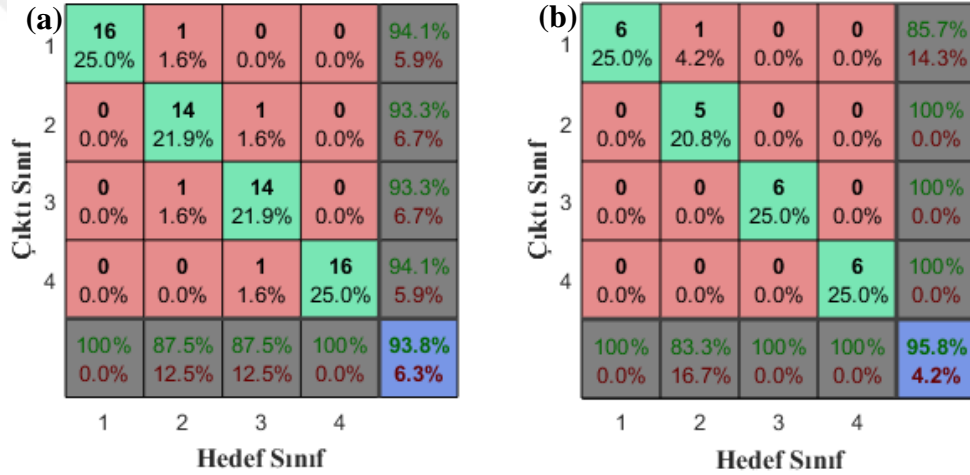
  

Independent Samples Test			
		Ort. Doğruluk Oranı	
		Equal variances assumed	Equal variances not assumed
Levene's Test for Equality of Variances	F	,429	
	Sig.	,521	
t-test for Equality of Means	t	1,720	1,720
	df	18	16,787
	Sig. (2-tailed)	,103	,104
	Mean Difference	1,87505	1,87505
	Std. Error Difference	1,09039	1,09039

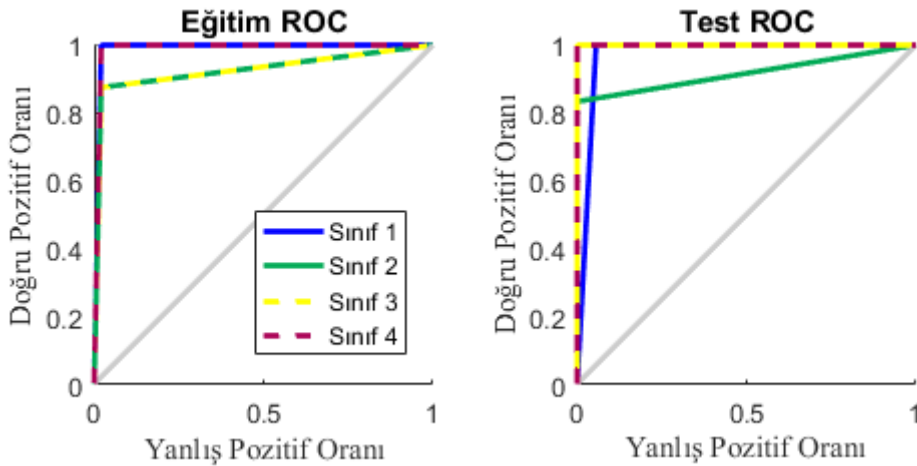
Şekil 19. Ortalama doğruluk oranına göre ÇKA ve Gri LVQ1 başarıları t-test sonucu

YSA sınıflandırıcı modelleri arasında sınıflandırma başarısı açısından fark olmadığı sonucuna ulaşıldıktan sonra Akçağaç (*Acer L.*) cinsine ait 4 farklı türün belirlenmesi için her bir yöntem için seçilen modellerin sınıflandırma sonuçları aşağıda verilmiştir.

Odun türü sınıflandırma için geliştirilen ilk yöntem ÇKA modeline ait eğitim ve test performansları Şekil 20 ve Şekil 21 ile gösterilmektedir. ÇKA modeli, ilki 4 ikincisi 1 nöron bulunduran iki gizli katmandan oluşmuştur. Eğitilen model ile eğitim ve test veri kümeleri için sırasıyla %93,8 ve %95,8 doğruluk elde edilmiştir. Şekil 20 ve Şekil 21'deki ikili karşılaştırma matrisleri ve ROC eğrileri incelendiğinde ise modelin, eğitimde Sınıf 2 ve Sınıf 3'e ait ikişer örneği, test veri kümesinde de 24 örnekten sadece Sınıf 2'ye ait bir örneği başarılı olarak sınıflandıramadığı görülmektedir.

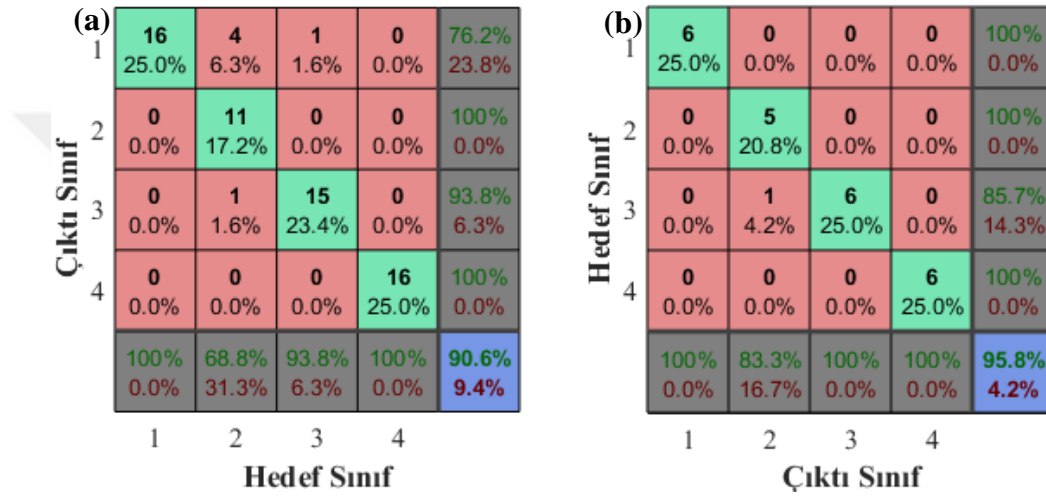


Şekil 20. ÇKA modeli için (a) eğitim ve (b) test veri kümeleri ikili karşılaştırma matrisi

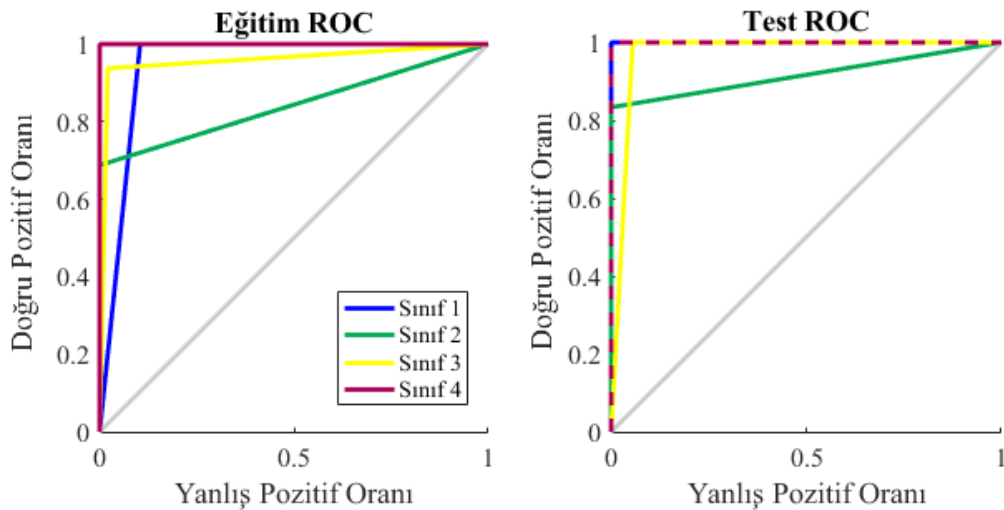


Şekil 21. ÇKA modeli için eğitim ve test veri kümelerine ait ROC eğrileri

Şekil 22 ve Şekil 23 odun sınıflandırma için seçilen LVQ modeline ait eğitim ve test performanslarını göstermektedir. LVQ modeli 21 referans vektörüne sahip bir ağ yapısından oluşmaktadır. 0,23 öğrenme oranı ( $lr$ ) ile eğitilen model, eğitim ve test veri kümeleri için sırasıyla %90,6 ve %95,8 doğruluk sağlamıştır. Şekil 22 ve Şekil 23'teki ikili karşılaştırma matrisleri ve ROC eğrileri incelendiğinde LVQ modelinin de Sınıf 2'ye ait örneklerin ayırımını yapmakta daha başarısız olduğu görülmektedir.

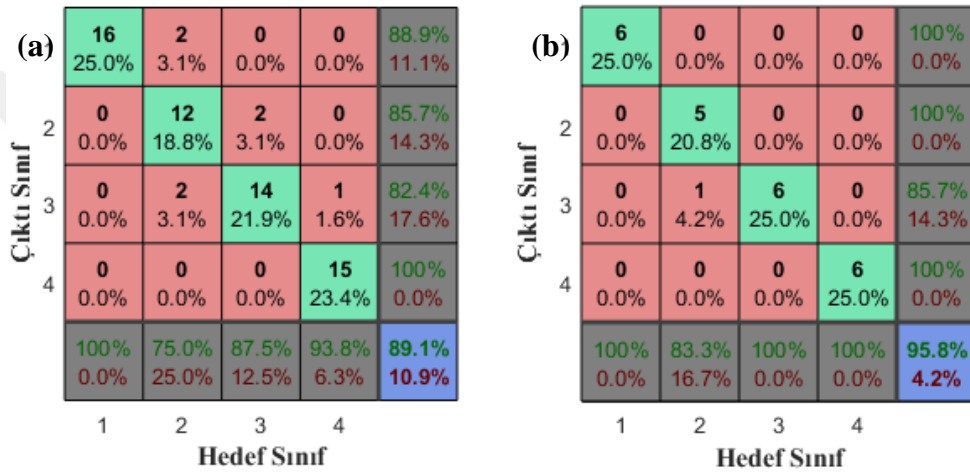


Şekil 22. LVQ1 modeli için (a) eğitim ve (b) test veri kümeleri ikili karşılaştırma matrisi

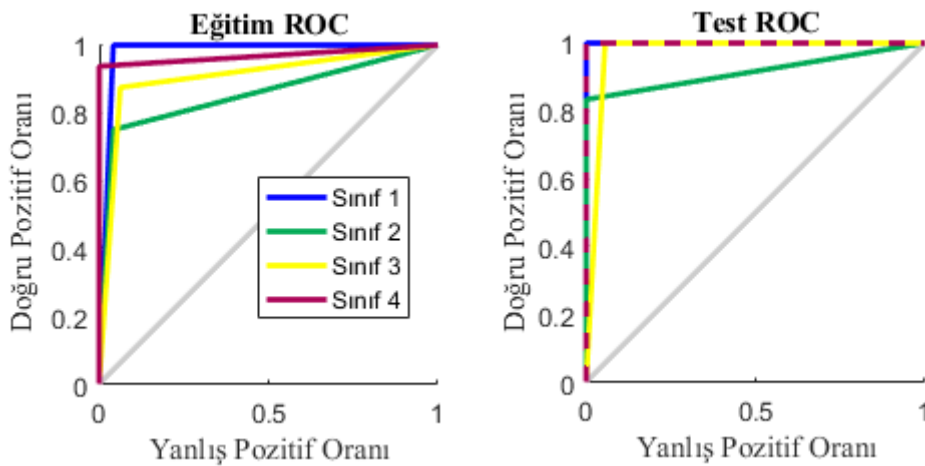


Şekil 23. LVQ modeli için eğitim ve test veri kümelerine ait ROC eğrileri

Şekil 24. Gri LVQ1 modeli için önerilen 18 referans vektörüne sahip bir ağ modelinin eğitim ve test veri kümeleri için sınıflara göre tahmin değerlerini ve başarılarını göstermektedir. Model, Sınıf 1'i temsil eden 3, Sınıf 2'yi temsil eden 6, Sınıf 3'ü temsil eden 5 ve Sınıf 4'i temsil eden 4 referans vektöründen oluşmuştur. Model, öğrenme oranı ( $lr$ ) 0,1 ile eğitilerek eğitim ve test veri kümeleri için sırasıyla %89,1 ve %95,8 doğruluk elde edilmiştir. Şekil 24 ve Şekil 25'teki ikili karşılaştırma matrisleri ve ROC eğrileri ile modelin eğitimde Sınıf 2'ye ait örnekleri öğrenmekte ve örneklerin sınıfını belirlemekte hatalı olduğu görülebilmektedir.



Şekil 24. Gri LVQ1 modeli için (a) eğitim ve (b) test veri kümeleri ikili karşılaştırma matrisi



Şekil 25. Gri LVQ1 modeli için eğitim ve test veri kümelerine ait ROC eğrileri

Her üç modele ait ikili karşılaştırma matrisleri ile ROC eğrileri genel olarak değerlendirildiğinde; ÇKA, LVQ ve Gri LVQ1 yöntemleri ile eğitim ve test aşamalarında, Sınıf 2'ye yani *Acer cappadocicum* türüne ait örneklerin sınıflandırılmasının diğer sınıfların örneklerine göre yanlışa daha açık olduğu sonucuna ulaşılmaktadır.



#### 4. SONUÇ VE ÖNERİLER

Bu çalışmada yapay sinir ağı yöntemleri kullanılarak sınıflandırma uygulaması yapılmıştır. Literatürde farklı açılarla ele alınmış olduğu görülen odun türü belirleme konusu üzerinde odaklanan çalışma iki boyutuyla önemlidir.

Çalışmanın ilk önemli yanı odun sınıflandırmada literatürde çok kullanılmamış olan yöntemle sınıflandırma yapılmış olmasıdır. Çoğunlukla odun numunelerine ait yüksek çözünürlüklü görüntülerden görüntü işleme teknikleri ile elde edilen veriler kullanılarak sınıflandırma yapılan çalışmalardan farklı olarak, literatürde çok fazla örneği olmayan ve daha çok son yıllarda çalışılmış olan odun anatomisine ait biyometrik ölçüm verileri ile sınıflandırma yapılmıştır. Genelde farklı cinslere ait odunların ticari amaçla ayrımının yapıldığı çalışmalara kıyasla daha zor olan, aynı cinsin farklı türlerinin ayrımının yapılmasını sağlayacak bir sınıflandırma çalışması gerçekleştirilmiştir. Literatürdeki 2 veya 3 tür için yapılan odun türü belirleme çalışmalarına benzer olarak 4 farklı odun türünün ayrımını sağlayacak modeller geliştirilmiştir. ÇKA, LVQ yöntemleri kullanılarak %95,8 doğruluk oranı ile sınıflandırma yapılabilmiş ve 24 örnekten oluşan test veri kümesindeki 23 örnek doğru sınıflandırılabilmiştir.

Çalışmanın ikinci önemli yanı ise odun türü belirleme amacıyla LVQ yönteminin kullanılmasıdır. Literatürde örneği olmayan odun anatomisine ait biyometrik verilerin kullanıldığı LVQ yapay sinir ağı yöntemi ile ilk kez odun sınıflandırma yapılmıştır. LVQ yönteminin odun türü sınıflandırma uygulamalarındaki başarısı incelenmiş ve bu yöntemin iyileştirilmesi amacıyla benzerlik hesaplamasını Gri İlişkisel Analiz yöntemi ile yapan yeni bir LVQ yöntemi kullanılmıştır. Gri LVQ1 olarak adlandırılan yeni yöntemle de ÇKA ve LVQ yöntemlerinin başarıları yakalanmıştır.

Bu tez çalışmasında odun anatomisine ait özellikler ile gerçekleştirilen sınıflandırma uygulaması, özellikle jeolojik ve arkeolojik odun buluntuların türlerinin tespiti için etkili bir yöntemdir. Anatomik özellikleri temel alan ve yeterince geniş bir veri tabanı ile geliştirilmiş bir sınıflandırıcı ile paleocoğrafik yayılımların tespiti, arkeolojik bölgelerde bulunan örnek türlerinin belirlenmesi, ahşap heykel ve ahşap eski binaların restorasyonu için aynı türün kullanılabilmesi gibi hedeflere ulaşmak oldukça kolay olacaktır.

Bundan sonraki çalışmalarda, farklı yöntemlerin kıyaslanması amacıyla sınıflara ait eşit sayıda örnek temsili ile hazırlanan veri yapısı yerine farklı sayılarda örnekten oluşan veri



kümeleri ile sınıflandırma gerçekleştirilerek sınıfların dağılımının sınıflandırma performansına etkisi incelenebilir. Gri LVQ yönteminin Gri İlişkisel Analiz bölümünde kriterleri temsil eden öznitelikler ağırlıklandırılarak sınıf değerinin belirlenmesindeki etkili öznitelikler tespit edilebilir. Ayrıca Gri LVQ yöntemi farklı alanlara ait veri setleri üzerinde incelenerek modelin genel sınıflandırma performansı araştırılabilir.



## 5. KAYNAKLAR

- Akay, D., 2011. Grey Relational Analysis Based on Instance Based Learning Approach for Classification of Risks of Occupational Low Back Disorders, Safety Science, 49, 8, 1277-1282.
- Angra, S. ve Ahuja, S., 2017. Machine Learning and its Applications: A review, International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), March, India, 57-60.
- Bailey, K.D., 1994. Typologies and Taxonomies: An Introduction to Classification Techniques, Sage Publications, California.
- Bar-Yam, Y., 1997. Dynamics of Complex Systems, Addison-Wesley Reading, Massachusetts.
- Barber, D., 2012. Bayesian Reasoning and Machine Learning, Cambridge University Press, New York.
- Bashyal, S. ve Venayagamoorthy, G.K., 2008. Recognition of Facial Expressions Using Gabor Wavelets and Learning Vector Quantization, Engineering Applications of Artificial Intelligence, 21, 7, 1056-1064.
- Bayramoglu, M.F. ve Hamzacebi, C., 2016. Stock Selection Based on Fundamental Analysis Approach by Grey Relational Analysis: A Case of Turkey, International Journal of Economics and Finance, 8, 7, 178-184.
- Birtürk, T., 2011. Karadeniz Bölgesinde Doğal Olarak Yetişen Akçaağaç (*Acer L.*) Taksonları Oduklarının Anatomik Özellikleri ve Farklı Yetiştirme Koşullarının Bu Özellikler Üzerine Etkisi, Doktora Tezi, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Trabzon.
- Bishop, C.M., 1994. Neural Networks and Their Applications, Review of Scientific Instruments, 65, 6, 1803-1832.
- Bishop, C.M., 2006. Pattern Recognition and Machine Learning, Springer Science Business Media, New York.
- Boubezoul, A., Paris, S. ve Ouladsine, M., 2007. Improving the GRLVQ Algorithm by The Cross Entropy Method. 17th International Conference on Artificial Neural Networks (ICANN), September, Porto, Proceedings Part I: 199-208.
- Brandtberg, T., 2002. Individual Tree-Based Species Classification in High Spatial Resolution Aerial Images of Forests Using Fuzzy Sets, Fuzzy Sets and Systems, 132, 3, 371-387.
- Broséus, J., Vallat, M. ve Esseiva, P., 2011. Multi-class Differentiation of Cannabis Seedlings in a Forensic Context, Chemometrics and Intelligent Laboratory Systems, 107, 2, 343-350.
- Brunner, M., Eugster, R., Trenka, E. ve Bergamin-Strotz, L., 1996. FT-NIR Spectroscopy and Wood Identification, Holzforschung-International Journal of the Biology, Chemistry, Physics and Technology of Wood, 50, 2, 130-134.

- Chiu, N.-H., 2009. An Early Software-Quality Classification Based on Improved Grey Relational Classifier, Expert Systems with Applications, 36, 7, 10727-10734.
- Clark, J.Y., 2003. Artificial Neural Networks for Species Identification by Taxonomists, Biosystems, 72, 1, 131-147.
- Clark, J.Y., Corney, D.P.A. ve Tang, H.L., 2012. Automated Plant Identification Using Artificial Neural Networks, IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), May, California, 343-348.
- Coşkun, C. ve Baykal, A., 2011. Veri Madenciliğinde Sınıflandırma Algoritmalarının Bir Örnek Üzerinde Karşılaştırılması, XIII. Akademik Bilişim Konferansı, Şubat, Malatya, Bildiriler Kitabı, 51-58.
- Davis, J. ve Goadrich, M., 2006. The Relationship Between Precision-Recall and ROC Curves, Proceedings of the 23rd International Conference on Machine Learning, June USA, 233-240.
- Delen, D., 2010. A Comparative Analysis of Machine Learning Techniques for Student Retention Management, Decision Support Systems, 49, 4, 498-506.
- Demirhan, A. ve Güler, İ., 2010. Özörgütlemeli Harita Ağları ve Gri Düzey Eş Oluşum Matrisleri ile Görüntü Bölütleme, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 25, 2, 285-291.
- DeSieno, D., 1988. Adding a Conscience to Competitive Learning, IEEE International Conference on Neural Networks, July, Piscataway, NJ, 117-124.
- Dieterle, F., Müller-Hagedorn, S., Liebich, H.M. ve Gauglitz, G., 2003. Urinary Nucleosides as Potential Tumor Markers Evaluated by Learning Vector Quantization, Artificial Intelligence in Medicine, 28, 3, 265-279.
- Du, Y.-C., Lin, C.-H., Shyu, L.-Y. ve Chen, T., 2010. Portable Hand Motion Classifier for Multi-Channel Surface Electromyography Recognition Using Grey Relational Analysis, Expert Systems with Applications, 37, 6, 4283-4291.
- Erdem, M., 2012. Mesleki Bel Ağrısı Riskinin Destek Vektör Makineleriyle Sınıflandırılması, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- Esteban, L.G., Fernández, F.G., de Palacios, P. de P., Romero, R.M. ve Cano, N.N., 2009. Artificial Neural Networks in Wood Identification: The Case of Two Juniperus Species From the Canary Islands, IAWA Journal, 30, 1, 87-94.
- Esteban, L.G., de Palacios, P., Conde, M., Fernández, F.G., García-Iruela, A. ve González-Alonso, M., 2017. Application of Artificial Neural Networks as a Predictive Method to Differentiate the Wood of Pinus Sylvestris L. and Pinus Nigra Arn Subsp. Salzmannii (Dunal) Franco, Wood Science and Technology, 51, 5, 1249-1258.
- Fausett, L., 1994. Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, Prentice-Hall, New Jersey.
- Fawcett, T., 2006. An Introduction to ROC Analysis, Pattern Recognition Letters, 27, 8, 861-874.
- Francisco-Fernández, M., Tarrío-Saavedra, J., Mallik, A. ve Naya, S., 2012. A Comprehensive Classification of Wood From Thermogravimetric Curves, Chemometrics and Intelligent Laboratory Systems, 118, 159-172.

- Francisco-Fernández, M., Tarrío-Saavedra, J., Naya, S., López-Beceiro, J. ve Artiaga, R., 2015. Classification of Wood Using Differential Thermogravimetric Analysis, Journal of Thermal Analysis and Calorimetry, 120, 1, 541-551.
- Hammer, B., Hofmann, D., Schleif, F.-M. ve Zhu, X., 2014. Learning Vector Quantization for (dis-)Similarities, Neurocomputing, 131, 43-51.
- Hammer, B., Schleif, F.-M. ve Zhu, X., 2011. Relational Extensions of Learning Vector Quantization, Neural Information Processing, International Conference on Neural Information Processing (ICONIP), November, China, Neural Information Processing, 481-489.
- Hammer, B., Strickert, M. ve Villmann, T., 2005. Supervised Neural Gas with General Similarity Measure, Neural Processing Letters, 21,1, 21-44.
- Hasan, A.F., Ahmad, M.F., Ayob, M.N., Rais, S.A.A., Saad, N.H., Faiz, A., Abidin, Z., vd., 2013. Application of Binary Particle Swarm Optimization in Automatic Classification of Wood Species Using Gray Level Co-Occurrence Matrix and K-Nearest Neighbor, Int. J. Sci. Eng. Res., 4, 50-55.
- Haykin, S., 1994. Neural Networks: A Comprehensive Foundation, Macmillan College Publishing Company, New York.
- He, H., Sun, C. ve Zhang, X., 2012. A Method for Identification of Driving Patterns in Hybrid Electric Vehicles Based on a LVQ Neural Network. Energies, 5, 9, 3363-3380.
- Hofmann, D., Gisbrecht, A. ve Hammer, B., 2013. Efficient Approximations of Kernel Robust Soft LVQ, Advances in Self-Organizing Maps, Advances in Intelligent Systems and Computing, Springer-Verlag, Berlin, 198, 183-192.
- Hosseinfard, B., Moradi, M.H. ve Rostami, R., 2013. Classifying Depression Patients and Normal Subjects Using Machine Learning Techniques and Nonlinear Features from EEG Signal, Computer Methods and Programs in Biomedicine, 109, 3, 339-345.
- Hu, Y.-C., 2008. Nonadditive Grey Single-Layer Perceptron with Choquet Integral for Pattern Classification Problems Using Genetic Algorithms, Neurocomputing, 72, 1, 331-340.
- Hu, Y.-C., Chen, R.-S., Hsu, Y.-T. ve Tzeng, G.-H., 2002. Grey Self-Organizing Feature Maps, Neurocomputing, 48, 1, 863-877.
- Huang, A., Zhang, Z., Xiang, Y. ve Gu, X., 2012. Study on Artificial Neural Network Combined with Near Infrared Spectroscopy for Wood Species Identification, Spectroscopy and Spectral Analysis, 32, 9, 2377-2381.
- Huang, S.-J., Chiu, N.-H. ve Chen, L.-W., 2008. Integration of the Grey Relational Analysis with Genetic Algorithm for Software Effort Estimation, European Journal of Operational Research, 188, 3, 898-909.
- James, G., Witten, D. ve Hastie, T., 2013. An Introduction to Statistical Learning: With Applications in R, Springer-Verlag, New York.
- Jerez, J.M., Molina, I., García-Laencina, P.J., Alba, E., Ribelles, N., Martín, M. ve Franco, L., 2010. Missing Data Imputation Using Statistical and Machine Learning Methods in a Real Breast Cancer Problem, Artificial Intelligence in Medicine, 50, 2, 105-115.
- Jiang, W., Özaktaş, B.B., Mantri, N., Tao, Z. ve Lu, H., 2013. Classification of Camellia

- Species from 3 Sections Using Leaf Anatomical Data with Back-Propagation Neural Networks and Support Vector Machines, Turk. J. Bot., 37, 6, 1093-1103.
- Jirayusakul, A. ve Auwatanamongkol, S., 2007. A Supervised Growing Neural Gas Algorithm for Cluster Analysis, Int. J. Hybrid Intell. Syst., 4, 2, 129-141.
- Jordan, R., Feeney, F., Nesbitt, N. ve Evertsen, J.A., 1998. Classification of Wood Species by Neural Network Analysis of Ultrasonic Signals, Ultrasonics, 36, 1-5, 219-222.
- Julong, D., 1989. Introduction to Grey System Theory, J. Grey. Syst., 1, 1, 1-24.
- Kaden, M., Lange, M., Nebel, D., Riedel, M., Geweniger, T. ve Villmann, T., 2014. Aspects in Classification Learning-Review of Recent Developments in Learning Vector Quantization, Foundations of Computing and Decision Sciences, 39, 2, 79-105.
- Kaya, Y., Ömer, Ertuğrul, F. ve Kaya, B.Y., 2017. Gender Classification from Facial Images Using Gray Relational Analysis with Novel Local Binary Pattern Descriptors, Signal, Image and Video Processing, 11, 769-776.
- Khalid, M., Lee, E.L.Y., Yusof, R. ve Nadaraj, M., 2008. Design of an Intelligent Wood Species Recognition System, International Journal of Simulation System, Science and Technology, 9, 3, 9-19.
- Kober, J. ve Peters, J., 2012. Reinforcement Learning in Robotics: A Survey. In: Wiering M., van Otterlo M. (eds) Reinforcement Learning. Adaptation, Learning, and Optimization, vol 12. Springer, Berlin, Heidelberg, 579-610.
- Kohonen, T., 1990. Improved Versions of Learning Vector Quantization. International Joint Conference on Neural Networks (IJCNN), June, USA, 545-550.
- Kohonen, T., 1986. Learning Vector Quantization for Pattern Recognition, Helsinki University of Technology, Report TKK-F-A601.
- Kohonen, T., Barna, G. ve Chrisley, R., 1988. Statistical Pattern Recognition with Neural Networks: Benchmarking Studies. International Joint Conference on Neural Networks (IJCNN), July, USA, 61-68.
- Kotsiantis, S.B., 2007. Supervised Machine Learning: A Review of Classification Techniques, Informatica, 31, 249-268.
- Kotsiantis, S.B., Zaharakis, I.D. ve Pintelas, P.E., 2006. Machine Learning: A Review of Classification and Combining Techniques, Artif. Intell. Rev., 26, 3, 159-190.
- Köklü, M., 2014. Sınıflandırma Problemlerinde Kural Çıkarımı İçin Yeni Bir Yöntem Geliştirilmesi ve Uygulamaları, Doktora Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya.
- Kubat, M., Holte, R.C. ve Matwin, S., 1998. Machine Learning for the Detection of Oil Spills in Satellite Radar Images, Machine Learning, 30, 2-3, 195-215.
- Kuo, C.-F.J. ve Su, T.-L., 2003. Gray Relational Analysis for Recognizing Fabric Defects, Textile Research Journal, 73, 5, 461-465.
- Lai, W.-C., Chang, T.-P., Wang, J.-J., Kan, C.-W. ve Chen, W.-W., 2012. An Evaluation of Mahalanobis Distance and Grey Relational Analysis for Crack Pattern in Concrete Structures, Computational Materials Science, 65, 115-121.
- Lewis, R.J., 2007. An Introduction to Classification and Regression Tree (CART) Analysis. 2000 Annual Meeting of the Society for Academic Emergency Medicine, May, USA

- Lin, C.-H., 2006. Classification Enhancible Grey Relational Analysis for Cardiac Arrhythmias Discrimination, Med. Biol. Eng. Comput., 44, 4, 311-320.
- Lin, C.-H., 2008. Frequency-Domain Features for ECG Beat Discrimination Using Grey Relational Analysis-Based Classifier, Comput. Math. Appl. 55, 4, 680-690.
- Lin, J.C. ve Tzeng, H.Y., 2010. Applying Particle Swarm Optimization to Estimate Software Effort by Multiple Factors Software Project Clustering, International Computer Symposium (ICS2010), December, Taiwan, 1039-1044.
- Liu, S. ve Lin, Y., 2006. Grey Information: Theory and Practical Applications Springer-Verlag, London.
- Liwei Zhou, Youyuan Wang, Feng Wang, Chunyu Yan ve Jiangang Bi, 2017. A Transformer Fault Diagnosis Method Based on Grey Relational Analysis and Integrated Weight Determination, IEEE Electrical Insulation Conference (EIC), June, USA, 491-494.
- Lloyd, G.R., Brereton, R.G., Faria, R. ve Duncan, J.C., 2007. Learning Vector Quantization for Multiclass Classification: Application to Characterization of Plastics, Journal of Chemical Information and Modeling, 47, 4, 1553-1563.
- Lu, H., Jiang, W., Ghiassi, M., Lee, S. ve Nitin, M., 2012. Classification of Camellia (Theaceae) Species Using Leaf Architecture Variations and Pattern Recognition Techniques, PLoS One, 7, 1, e29704.
- Mallik, A., Tarrío-Saavedra, J., Francisco-Fernández, M. ve Naya, S., 2011. Classification of Wood Micrographs by Image Segmentation, Chemometrics and Intelligent Laboratory Systems, 107, 2, 351-362.
- Mao, P.J., Liu, L., Wang, J. ve Hu, C.Y., 2010. A Study on Gray Relational Analysis of Many Factor Weights in Tobacco Leaves Classification, Advanced Materials Research, 39, 1728-1731.
- Martins, J., Oliveira, L.S., Nisgoski, S. ve Sabourin, R., 2013. A Database for Automatic Classification of Forest Species, Machine Vision and Applications: 24, 567-578.
- Martins, J.G., Oliveira, L.S., Britto, A.S. ve Sabourin, R., 2015. Forest Species Recognition Based on Dynamic Classifier Selection and Dissimilarity Feature Vector Representation, Machine Vision and Applications, 26, 2–3, 279-293.
- Michie, D., Michie, D., Spiegelhalter, D.J. ve Taylor, C.C., 1994. Machine Learning, Neural and Statistical Classification, Ellis Horwood, New Jersey.
- Mills, P., 2011. Efficient Statistical Classification of Satellite Measurements, International Journal of Remote Sensing, 32, 21, 6109-6132.
- Nisgoski, S., de Oliveira, A.A. ve de Muñiz, G.I.B., 2017. Artificial Neural Network and SIMCA Classification in Some Wood Discrimination Based on Near-Infrared Spectra, Wood Science and Technology, 51, 4, 929-942.
- Nova, D. ve Estévez, P.A., 2014. A Review of Learning Vector Quantization Classifiers, Neural Computing and Applications, 25, 3–4, 511-524.
- Olson, D.L. ve Delen, D., 2008. Advanced Data Mining Techniques, Springer-Verlag, Berlin.
- Otukei, J.R. ve Blaschke, T., 2010. Land Cover Change Assessment Using Decision Trees, Support Vector Machines and Maximum Likelihood Classification Algorithms, Int J

Appl Earth Obs Geoinf, 12, 27-31.

- Ozsahin, S. ve Murat, M., 2017. Prediction of Equilibrium Moisture Content and Specific Gravity of Heat Treated Wood by Artificial Neural Networks, European Journal of Wood and Wood Products: 1-10.
- Öztemel, E., 1992. Integrating Expert Systems and Neural Networks for Intelligent on-line Statistical Process Control, Doctoral Dissertation, University of Wales.
- Öztemel, E., 2003. Yapay Sinir Ağları, PapatyaYayincilik, İstanbul.
- Pérez-Ortiz, M., Jiménez-Fernández, S., Gutiérrez, P.A., Alexandre, E., Hervás-Martínez, C. ve Salcedo-Sanz, S., 2016. A Review of Classification Problems and Algorithms in Renewable Energy Applications, Energies, 9, 8, 1-27.
- Pham, D.T. ve Oztemel, E., 1994. Control Chart Pattern Recognition using Learning Vector Quantization Networks, International Journal of Production Research, 32, 3, 721-729.
- Piuri, V. ve Scotti, F., 2010. Design of an Automatic Wood Types Classification System by Using Fluorescence Spectra, IEEE T. Syst. Man. Cy. C., 40, 3, 358-366.
- Qin, A.K. ve Suganthan, P.N., 2005. Initialization Insensitive LVQ Algorithm Based on Cost-Function Adaptation, Pattern Recognition, 38, 5, 773-776.
- Rafiq, M.Y., Bugmann, G. ve Easterbrook, D.J., 2001. Neural Network Design for Engineering Applications, Computers & Structures, 79, 17, 1541-1552.
- Saeys, Y., Inza, I. ve Larrañaga, P., 2007. A Review of Feature Selection Techniques in Bioinformatics, Bioinformatics, 23, 19, 2507-2517.
- Sağıroğlu, Ş., Beşdok, E. ve Erler, M., 2003. Mühendislikte Yapay Zeka Uygulamaları: Yapay Sinir Ağları, Ufuk Kitap, Kayseri.
- Sato, A. ve Yamada, K., 1995. Generalized Learning Vector Quantization, Advances in Neural Information Processing Systems 8 (NIPS), 423-429.
- Schleif, F.M., Villmann, T., Hammer, B. ve Schneider, P., 2011. Efficient Kernelized Prototype Based Classification, Int. J. Neural. Syst., 21, 6, 443-457.
- Schneider, P., 2010. Advanced Methods for Prototype-Based Classification, Doctoral Thesis, University of Groningen.
- Schneider, P., Biehl, M. ve Hammer, B., 2009a. Adaptive Relevance Matrices in Learning Vector Quantization, Neural Computation, 21, 12, 3532-3561.
- Schneider, P., Biehl, M. ve Hammer, B., 2009b. Distance Learning in Discriminative Vector Quantization, Neural Computation, 21, 10, 2942-2969.
- Seo, S. ve Obermayer, K., 2003. Soft Learning Vector Quantization, Neural Computation, 15, 7, 1589-1604.
- Sun, L.J., Ji, Z.W. ve Wang, H.J., 2011. A New Wood Recognition Method Based on Texture Analysis, Applied Mechanics and Materials. 58, 613-617.
- Sutton, R.S., 1984. Temporal Credit Assignment in Reinforcement Learning. Doctoral Dissertation, University of Massachusetts Amherst.
- Tomak, L. ve Yüksel, B.E.K., 2009. İşlem Karakteristik Eğrisi Analizi ve Eğri Altında Kalan Alanların Karşılaştırılması, Deneysel ve Klinik Tıp Dergisi, 27, 58-65.
- Tou, J.Y., Tay, Y.H. ve Lau, P.Y., 2009. A Comparative Study for Texture Classification

- Techniques on Wood Species Recognition Problem, Fifth International Conference on Natural Computation. August, China, 8-12.
- Tsanas, A. ve Xifara, A., 2012. Accurate Quantitative Estimation of Energy Performance of Residential Buildings Using Statistical Machine Learning Tools, Energy and Buildings, 49, 560-567.
- Tsuchikawa, S., Inoue, K., Noma, J. ve Hayashi, K., 2003. Application of Near-Infrared Spectroscopy to Wood Discrimination, Journal of Wood Science, 49, 1, 29-35.
- Turhan, K. ve Serdar, B., 2013. Support Vector Machines in Wood Identification: The Case of Three Salix Species from Turkey, Turk. J. Agric. For., 37, 2, 249-256.
- Verleysen, M., Thissen, P. ve Legat, J.-D., 1993. Linear Vector Classification: An Improvement on LVQ Algorithms to Create Classes of Patterns, New Trends in Neural Computation, Springer, Berlin, 340-345.
- Vlassis, N. ve Likas, A., 2002. A Greedy EM Algorithm for Gaussian Mixture Learning, Neural Processing Letters, 15, 1, 77-87.
- Wang, H., Qi, H. ve Wang, X.-F., 2013. A New Gabor Based Approach for Wood Recognition, Neurocomputing, 116, 192-200.
- Wei, G., 2011. Grey Relational Analysis Model for Dynamic Hybrid Multiple Attribute Decision Making, Knowledge-Based Systems, 24, 5, 672-679.
- White, R.L., 2000. Object Classification as a Data Analysis Tool, Astronomical Data Analysis Software and Systems IX, October, USA, ASP Conference Proceedings, 216, 577-586.
- Witten, I.H. ve Frank, E., 2005. Data Mining: Practical Machine Learning Tools and Techniques, 2nd Edition, Morgan Kaufmann Publishers, San Francisco.
- Wu, H.-Y., Tzeng, G.-H. ve Chen, Y.-H., 2009. A fuzzy MCDM Approach for Evaluating Banking Performance Based on Balanced Scorecard, Expert Systems with Applications, 36, 6, 10135-10147.
- Yadav, A.R., Dewal, M.L., Anand, R.S. ve Gupta, S., 2013. Classification of Hardwood Species Using ANN Classifier. Computer Vision, Pattern Recognition, Fourth National Conference on Image Processing and Graphics (NCVPRIPG), December, India, 1-5.
- Yeh, P.-L., Fahn, C.-S., Lin, Y.-T., Hung, H.-F., Hsu, Y.-L. ve Hsu, Y.-T., 2017. A Novel Prediction Method Based on Grey-LVQ Neural Network, The Journal of Grey System, 29, 1, 34-51.
- Yiran, G. ve Suoping, W., 2010. Traffic Identification Method for Specific P2P Based on Multilayer Tree Combination Classification by BP-LVQ Neural-Network, International Forum on Information Technology & Applications, July, China, 34-38.
- Zhang, D., Chen, X., Han, Y., Cong, L., Wang, Q. ve Wang, X., 2014. Remote Sensing Image Classification with Small Training Samples Based on Grey Theory, Third International Workshop on Earth Observation and Remote Sensing Applications (EORSA), June, China, 190-194.
- Zhou, Z.Q., Li, S.C., Li, L.P., Shi, S.S. ve Xu, Z.H., 2015. An Optimal Classification Method for Risk Assessment of Water Inrush in Karst Tunnels Based on Grey System Theory, Geomechanics and Engineering, 8, 5, 631-647.



## 6. EKLER

Ek Tablo 1. Sınıflandırmada kullanılan odun örnekleri ve anatomik özellikleri

Sıra	Örnek No	Takson	TTC (µm)	TRÇ (µm)	THU (µm)	TS (adet/mm <sup>2</sup> )	MÖY (µm)
1	38	<i>Acer campestre</i> subsp. <i>campestre</i>	43,72	56,85	313,12	34	353,92
2	39	<i>Acer campestre</i> subsp. <i>campestre</i>	43,78	51,31	329,76	55	359,36
3	50	<i>Acer campestre</i> subsp. <i>campestre</i>	42,98	52,62	314,88	53	307,36
4	51	<i>Acer campestre</i> subsp. <i>campestre</i>	44,34	57,47	339,04	74	349,6
5	52	<i>Acer campestre</i> subsp. <i>campestre</i>	36,82	47,45	349,6	54	360,48
6	53	<i>Acer campestre</i> subsp. <i>campestre</i>	39,74	49,57	322,08	90	338,4
7	65	<i>Acer campestre</i> subsp. <i>campestre</i>	43,6	52,62	354,4	77	333,6
8	86	<i>Acer campestre</i> subsp. <i>campestre</i>	44,03	54,17	338,4	35	301,76
9	92	<i>Acer campestre</i> subsp. <i>campestre</i>	43,47	55,85	349,12	60	324,8
10	94	<i>Acer campestre</i> subsp. <i>campestre</i>	40,86	50,07	298,72	76	319,68
11	96	<i>Acer campestre</i> subsp. <i>campestre</i>	45,96	55,17	357,28	64	314,72
12	100	<i>Acer campestre</i> subsp. <i>leiocarpum</i>	40,43	50,69	365,92	74	323,52
13	105	<i>Acer campestre</i> subsp. <i>campestre</i>	40,74	48,76	316,16	83	360
14	106	<i>Acer campestre</i> subsp. <i>campestre</i>	44,59	46,71	291,2	59	302,4
15	107	<i>Acer campestre</i> subsp. <i>campestre</i>	45,09	52,62	348,96	57	421,92
16	121	<i>Acer campestre</i> subsp. <i>campestre</i>	42,79	56,35	323,68	67	354,88
17	122	<i>Acer campestre</i> subsp. <i>campestre</i>	39,31	48,76	298,56	65	328
18	127	<i>Acer campestre</i> subsp. <i>campestre</i>	46,4	57,84	350,72	56	413,92
19	131	<i>Acer campestre</i> subsp. <i>campestre</i>	41,3	51,31	306,4	110	347,36
20	134	<i>Acer campestre</i> subsp. <i>campestre</i>	52,12	57,47	305,44	59	384,96
21	148	<i>Acer campestre</i> subsp. <i>leiocarpum</i>	42,17	52,55	348,48	54	311,68
22	155	<i>Acer campestre</i> subsp. <i>leiocarpum</i>	42,54	48,64	334,88	73	305,6
23	32	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	46,71	46,27	450,56	38	318,56
24	36	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	46,89	64,37	342,08	37	372,16
25	40	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	38,43	41,48	328,8	40	341,28
26	41	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	40,61	51,87	287,2	28	353,6
27	42	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	51,12	63,25	364,48	19	447,04
28	46	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	54,3	75,75	393,12	44	365,92
29	47	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	55,66	57,28	395,04	47	420,48
30	55	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	57,47	69,66	415,04	37	452,96
31	56	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	52,49	62,82	309,92	48	376,32
32	58	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	49,32	53,49	336,64	48	441,12
33	123	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	47,64	55,35	300,96	54	297,12
34	125	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	49,45	57,03	358,88	62	479,04
35	133	<i>Acer cappadocicum</i> var. <i>stenocarpum</i>	42,42	47,14	316,64	59	346,72
36	135	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	48,95	63,87	306,4	45	330,88
37	136	<i>Acer cappadocicum</i> var. <i>stenocarpum</i>	51,81	69,66	345,6	39	402,56
38	145	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	48,32	52,37	298,88	57	372,8
39	147	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	47,08	63,25	352,48	53	405,92
40	149	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	43,72	48,82	368,8	50	397,12
41	165	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	49,32	59,89	392,16	43	527,2
42	170	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	51,43	60,58	420,48	51	376,32
43	171	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	51,19	55,66	349,76	60	711,52
44	172	<i>Acer cappadocicum</i> var. <i>cappadocicum</i>	45,46	55,04	339,68	47	428,32

Ek Tablo 1'in devamı

Sıra	Örnek No	Takson	TTÇ (µm)	TRÇ (µm)	THU (µm)	TS (adet/mm <sup>2</sup> )	MÖY (µm)
45	48	<i>Acer platanoides</i>	63,81	74,57	406,56	30	309,12
46	57	<i>Acer platanoides</i>	54,61	65,31	331,36	57	306,4
47	67	<i>Acer platanoides</i>	52,55	64,12	324,48	41	365,92
48	69	<i>Acer platanoides</i>	55,42	67,48	298,72	22	397,76
49	72	<i>Acer platanoides</i>	48,26	54,61	329,76	48	396,16
50	79	<i>Acer platanoides</i>	53,61	57,09	323,52	42	423,04
51	80	<i>Acer platanoides</i>	70,47	81,04	382,56	37	368,64
52	83	<i>Acer platanoides</i>	51,56	60,83	329,44	46	281,6
53	88	<i>Acer platanoides</i>	52,62	69,35	324,8	37	355,36
54	90	<i>Acer platanoides</i>	53,49	77,87	286,08	22	336,16
55	93	<i>Acer platanoides</i>	51,56	66,11	342,88	25	411,84
56	97	<i>Acer platanoides</i>	53,86	66,05	336,48	39	384
57	101	<i>Acer platanoides</i>	45,03	51,06	274,56	45	293,6
58	109	<i>Acer platanoides</i>	48,45	63,44	356,96	40	464,64
59	110	<i>Acer platanoides</i>	57,22	77,99	334,72	33	351,2
60	128	<i>Acer platanoides</i>	47,39	56,85	313,28	30	489,12
61	130	<i>Acer platanoides</i>	52,06	67,11	276,8	33	338,56
62	132	<i>Acer platanoides</i>	51,75	63,44	338,88	52	346,24
63	138	<i>Acer platanoides</i>	48,7	58,21	359,52	34	420,32
64	143	<i>Acer platanoides</i>	45,71	57,34	272,96	55	352,16
65	154	<i>Acer platanoides</i>	45,28	57,03	324,32	36	333,12
66	157	<i>Acer platanoides</i>	52,8	66,86	351,04	32	411,2
67	30	<i>Acer trautvetteri</i>	60,89	74,76	391,36	13	592,16
68	59	<i>Acer trautvetteri</i>	62,57	60,39	397,76	44	849,6
69	60	<i>Acer trautvetteri</i>	58,03	62,57	366,24	22	514,56
70	61	<i>Acer trautvetteri</i>	69,91	68,42	359,36	34	662,4
71	62	<i>Acer trautvetteri</i>	70,28	76,44	332,32	22	872,8
72	64	<i>Acer trautvetteri</i>	55,91	61,14	377,6	32	983,52
73	70	<i>Acer trautvetteri</i>	54,23	68,42	401,92	32	592,32
74	71	<i>Acer trautvetteri</i>	55,23	64,87	349,76	24	647,52
75	73	<i>Acer trautvetteri</i>	59,33	69,04	355,68	33	530,88
76	74	<i>Acer trautvetteri</i>	65,93	78,24	380,48	28	678,56
77	84	<i>Acer trautvetteri</i>	66,55	83,53	348,96	27	427,36
78	89	<i>Acer trautvetteri</i>	61,7	68,48	446,24	32	773,12
79	99	<i>Acer trautvetteri</i>	68,1	77,37	370,88	30	880,16
80	102	<i>Acer trautvetteri</i>	67,54	78,24	369,44	27	456
81	111	<i>Acer trautvetteri</i>	57,47	62,51	450,08	24	714,72
82	139	<i>Acer trautvetteri</i>	47,2	57,65	364	33	453,76
83	150	<i>Acer trautvetteri</i>	71,59	84,84	428,48	29	718,4
84	152	<i>Acer trautvetteri</i>	50,44	64,12	311,36	39	409,76
85	153	<i>Acer trautvetteri</i>	52,06	60,45	367,36	39	510,4
86	166	<i>Acer trautvetteri</i>	58,84	69,72	372,8	22	557,76
87	167	<i>Acer trautvetteri</i>	59,33	71,15	353,6	24	619,04
88	168	<i>Acer trautvetteri</i>	59,02	70,97	377,6	40	517,6

Ek Tablo 1'in devamı

Sıra	Örnek No	MÖG (µm)	ÖS (Adet/mm)	LFU (µm)	LFG (µm)	LÜMG (µm)	LÇPK (µm)	Sınıf
1	38	35,36	9	744,64	19,34	11,75	3,79	1
2	39	41,76	8	759,36	18,84	12,44	3,2	1
3	50	35,68	7	705,12	18,22	12,06	3,07	1
4	51	37,12	8	734,24	18,03	10,88	3,57	1
5	52	40,64	6	789,28	18,16	10,63	3,76	1
6	53	37,44	7	725,44	16,54	9,14	3,7	1
7	65	36,96	7	738,56	19,53	12,25	3,63	1
8	86	40	8	741,76	18,34	10,51	3,98	1
9	92	31,36	7	743,52	18,72	11,13	3,79	1
10	94	46,88	7	756,96	18,28	11,32	3,48	1
11	96	34,56	7	763,52	18,66	11,19	3,73	1
12	100	37,28	7	754,88	20,02	12,31	3,85	1
13	105	32,64	8	715,2	18,47	11,69	3,39	1
14	106	44,32	7	750,08	18,78	10,94	3,91	1
15	107	44,48	8	722,56	17,54	9,95	3,79	1
16	121	29,12	8	728,32	18,03	11,25	3,38	1
17	122	45,28	7	691,36	16,35	8,76	3,79	1
18	127	39,04	7	764,96	18,84	11,44	3,7	1
19	131	30,72	8	693,28	17,91	10,32	3,79	1
20	134	46,24	7	693,92	19,22	12,5	3,42	1
21	148	39,36	6	687,84	18,65	11,57	3,54	1
22	155	39,68	6	756	18,59	10,63	4,04	1
23	32	32	6	912,96	17,41	10,57	3,42	2
24	36	43,68	7	843,68	17,91	10,94	3,48	2
25	40	46,72	6	785,6	18,84	12,12	3,35	2
26	41	34,4	9	733,44	16,91	10,51	3,2	2
27	42	35,68	6	928,8	20,52	13,49	3,51	2
28	46	45,28	5	915,84	21,95	13,43	4,26	2
29	47	46,08	6	845,92	22,39	13,62	4,38	2
30	55	47,52	5	829,12	21,21	13,18	4,01	2
31	56	41,44	7	714,08	20,83	14,05	3,38	2
32	58	55,68	6	772,48	20,34	12,37	3,98	2
33	123	31,68	6	681,12	18,47	11,88	3,6	2
34	125	39,04	6	761,12	19,96	12,31	3,82	2
35	133	43,52	6	715,52	18,28	10,82	3,73	2
36	135	38,08	6	680,16	20,96	13,31	3,82	2
37	136	44,48	6	824,48	18,78	11,5	3,63	2
38	145	40,32	6	702,88	19,96	12,81	3,57	2
39	147	31,04	6	765,76	19,22	11,81	3,7	2
40	149	44,16	4	697,44	19,78	12,31	3,73	2
41	165	59,04	5	869,6	18,84	10,88	3,98	2
42	170	43,36	5	767,68	19,96	13,24	3,35	2
43	171	52,64	6	742,4	19,15	12,5	3,32	2
44	172	47,04	6	781,44	19,59	12,56	3,51	2

Ek Tablo 1'in devamı

Sıra	Örnek No	MÖG (µm)	ÖS (Adet/mm)	LFU (µm)	LFG (µm)	LÜMG (µm)	LÇPK (µm)	Sınıf
45	48	47,52	5	788,16	21,7	13,37	4,16	3
46	57	49,6	5	780,8	18,72	10,32	4,19	3
47	67	49,92	5	841,44	18,28	10,69	4,04	3
48	69	36,48	7	847,04	21,33	13,8	3,76	3
49	72	54,88	7	752,32	20,77	13,31	3,73	3
50	79	43,36	7	786,24	18,72	10,69	4,01	3
51	80	50,88	4	771,84	23,2	14,55	4,32	3
52	83	43,84	6	747,68	21,77	13,49	4,13	3
53	88	39,52	6	782,08	20,02	11,69	4,16	3
54	90	38,08	6	807,36	21,45	14,67	3,39	3
55	93	36,32	6	906,56	22,2	15,92	3,17	3
56	97	40,64	5	822,88	20,71	12,75	3,98	3
57	101	53,76	5	667,84	18,22	11,5	3,29	3
58	109	53,44	5	894,24	19,28	10,69	4,35	3
59	110	42,08	5	947,2	21,39	14,24	3,57	3
60	128	44,48	6	802,24	18,59	11,13	3,67	3
61	130	38,4	7	635,36	19,09	11,19	3,91	3
62	132	44,64	6	717,76	18,97	10,76	4,07	3
63	138	55,52	5	748,32	18,53	10,94	3,79	3
64	143	54,08	6	660,8	16,79	8,5	4,1	3
65	154	40,8	6	768,48	19,15	10,57	4,29	3
66	157	56,8	6	818,24	19,53	10,94	4,29	3
67	30	47,84	6	804,64	19,71	13,37	3,17	4
68	59	68,64	6	765,28	23,32	15,42	3,95	4
69	60	71,2	5	772,96	19,71	13	3,35	4
70	61	84,96	5	728,8	21,02	13,49	3,54	4
71	62	102,88	6	769,28	23,26	16,23	3,51	4
72	64	86,56	6	732	20,4	12,81	3,79	4
73	70	63,04	4	901,92	22,88	14,55	4,16	4
74	71	81,44	5	848,96	19,53	12,68	3,73	4
75	73	69,6	4	788,96	18,65	11,69	3,42	4
76	74	92,96	5	902,88	21,89	14,55	3,67	4
77	84	72,32	5	829,6	24,25	16,54	3,85	4
78	89	69,92	5	920,48	20,46	13,12	3,73	4
79	99	80,32	4	825,6	20,89	13,31	3,79	4
80	102	87,36	5	746,08	23,01	15,48	3,76	4
81	111	78,88	4	836,48	22,76	15,11	3,82	4
82	139	65,92	5	768,48	18,22	11,88	3,48	4
83	150	95,36	5	866,72	24,94	16,85	4,04	4
84	152	69,12	5	700,16	19,28	12,31	3,79	4
85	153	78,72	6	824,96	20,21	13,24	3,6	4
86	166	71,84	5	840,32	21,89	14,49	3,7	4
87	167	83,36	5	830,88	24,07	16,04	4,01	4
88	168	81,44	6	805,12	21,39	14,11	3,6	4

## ÖZGEÇMİŞ

Miraç MURAT 1989 yılında Trabzon'da doğdu. 2007 yılında Akçaabat Anadolu Lisesi'nden mezun olduktan bir yıl sonra İstanbul Üniversitesi Mühendislik Fakültesi Endüstri Mühendisliği Bölümü'nde lisans öğrenimine başladı ve 2012 yılında mezun oldu. 2014 yılında Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalında yüksek lisans öğrenimine başladı. Aynı yıl Boğaziçi Üniversitesi'nde İngilizce dil eğitimi aldı. 2014 yılı Eylül ayından beri Karadeniz Teknik Üniversitesi Endüstri Mühendisliği Bölümünde araştırma görevlisi olarak çalışmakta olan Miraç MURAT iyi derecede İngilizce bilmektedir.