

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DESTEK VEKTÖR MAKİNE TABANLI BULANIK SİSTEMLER,
YENİ BİR GÜRBÜZ SINIFLAYICI VE REGRESÖR TASARIMI**

Ayşegül UÇAR

Tez Yöneticileri
Prof. Dr. Yakup DEMİR
Prof. Dr. Cüneyt GÜZELİŞ

DOKTORA TEZİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ
ANA BİLİM DALI

ELAZIĞ, 2006

T.C.
FIRAT UNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DESTEK VEKTÖR MAKİNE TABANLI BULANIK SİSTEMLER,
YENİ BİR GÜRBÜZ SINIFLAYICI VE REGRESÖR TASARIMI**

Ayşegül UÇAR

Doktora Tezi
Elektrik-Elektronik Mühendisliği
Ana Bilim Dalı

Bu tez, 8 Kasım 2006 tarihinde aşağıda belirtilen jüri tarafından oybirliği ile başarılı olarak değerlendirilmiştir.

Danışman: Prof. Dr. Yakup DEMİR

Danışman: Prof. Dr. Cüneyt GÜZELİŞ

Üye: Prof. Dr. Ferit Acar SAVACI

Üye: Prof. Dr. Mustafa POYRAZ

Üye: Doç. Dr. Erhan AKIN

Üye: Yrd. Doç. Dr. Selçuk YILDIRIM

Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun/...../..... tarih ve sayılı kararıyla onaylanmıştır.

TEŞEKKÜR

Öncelikle Prof. Dr. Cüneyt Güzeliş ve Prof. Dr. Yakup Demir'e onların denetimi altında doktora çalışmama fırsat verdikleri için teşekkür etmek istiyorum.

Tüm çalışmalarım boyunca Prof. Dr. Yakup Demir'in yakın ilgisi, önerileri, sabrı ve anlayışı hem akademik hem de sosyal açıdan gelişmem için sürekli olarak beni teşvik etmiştir. Onunla çalışmak fırsatını yakaladığım için her zaman çok şanslı olduğumu düşünürüm.

Son beş yıl boyunca uzak mesafeye ve yoğun çalışmalarına rağmen, destek ve öğütleriyle her zaman yanımda olan Prof. Dr. Cüneyt Güzeliş sayesinde akademik bakış açım değişmiştir.

Dr. Hatice Doğan'a yakın arkadaşlığı, tez çalışmalarım hakkındaki önerileri ve programlarımı geliştirmemdeki katkıları için çok teşekkür ederim.

Eğitim hayatım boyunca beni destekleyen aileme minnettarım.

Ayşegül UÇAR

İÇİNDEKİLER TABLOSU

	Sayfa
İÇİNDEKİLER	i
ŞEKİLLER LİSTESİ	v
TABLolar LİSTESİ	viii
KULLANILAN BAZI TERİMLER LİSTESİ	ix
KISALTMALAR LİSTESİ	x
SEMBOLLER LİSTESİ	xi
ÖZET	xii
ABSTRACT	xiii
1. GİRİŞ	1
1.2. Tezin Organizasyonu	8
2. GİRİŞ UZAYINDA KÜRE ve ELİPSOİT BİÇİMLİ AYRIŞTIRICI YÜZEYLER KULLANARAK SINIFLAMA	10
2.1. Destek Vektör Makinelerin Bazı Eksiklikleri	10
2.1.1. Küre ve Elipsoit Biçimli Çekirdek Kullanarak Sınıflama	12
2.2. Önerilen Küre ve Elipsoit Biçimli Sınıflayıcıların Tanımı	15
2.2.1. Matematiksel Temel	17
2.2.1.1. Uyarlanı Öğrenme Oranlı Eğim İniş Yöntemi	19
2.2.1.2. Momentum Terimi İçeren Uyarlanı Öğrenme Oranlı Eğim İniş Yöntemi	20
2.2.1.3. Ölçeklenmiş Eşlenik Eğim İniş Yöntemi	21
2.2.1.4. Yarı-Newton Yöntemi	21
2.3. Önerilen İki Aşamalı Formülasyon	23
2.3.1. Aşama-I: Küre Biçimli Sınıflayıcılar	23
2.3.1.1. Önerilen Küre Biçimli Sınıflayıcılar İçin Çözüm Yöntemi	24
2.3.2. Aşama-II: Elipsoit Biçimli sınıflayıcılar	24
2.3.2.2. Önerilen Elipsoit Biçimli Sınıflayıcılar İçin Çözüm Yöntemi	25
2.3.3. Penaltı Parametresinin Seçimi	26
2.4. Uygulamalar	27
2.4.1. Veri Kümeleri	27
2.4.1.2. İki Sınıflı Sınıflayıcılar İçin Uygulamalar	29
2.4.2. Önerilen Sınıflayıcıların Farklı Yöntemlerle Eğitilerek Başarımının İncelenmesi	29

2.4.3. Zambak Çiçeği Veri Kümesi ile İki Sınıflı Sınıflama Problemi	38
2.4.4. CKH, BUPAKB, İyonosfer, WGK ve GYA Veri kümeleri ile Sınıflama Problemi	39
2.5. Çok Sınıflı Sınıflama	45
2.5.1. Aşama-I: Küre Biçimli m-Sınıflı Sınıflayıcılar	45
2.5.1.1. Önerilen Küre Biçimli m-Sınıflı Sınıflayıcılar İçin Çözüm Yöntemi	46
2.5.2. Aşama-II: Elipsoit Biçimli m-Sınıflı Sınıflayıcılar	47
2.5.2.1. Önerilen Elipsoit Biçimli Sınıflayıcılar İçin Çözüm Yöntemi	48
2.6. Çok Sınıflı Sınıflama İçin Uygulamalar	48
3. BULANIK KÜRE ve ELİPSOİT BİÇİMLİ SINIFLAYICILAR	52
3.1. Önerilen Bulanık Küre ve Elipsoit Biçimli Sınıflama Probleminin Tanıtımı	52
3.2. Aşama-I: Bulanık Küre Biçimli Sınıflayıcılar	52
3.2.1. Bulanık Küre Biçimli Sınıflayıcılar İçin Çözüm Yöntemi	53
3.3. Aşama-II: Bulanık Elipsoit Biçimli Sınıflayıcılar	54
3.3.1. Bulanık Elipsoit Biçimli Sınıflayıcılar İçin Çözüm Yöntemi	54
3.4. m-Sınıflı Bulanık Küre ve Elipsoit Biçimli Sınıflayıcılar	55
3.4.1. Aşama-I: Küre Biçimli m-Sınıflı Bulanık Sınıflayıcılar	55
3.4.2. Aşama-II: Elipsoit Biçimli m-Sınıflı Bulanık Sınıflayıcılar	56
3.5. Uygulamalar	57
3.5.1. Yapay Veri Kümesi ile Sınıflama Problemi	57
3.5.2. Zambak Çiçeği Veri Kümesi ile İki Sınıflı Sınıflama Problemi	60
3.5.3. Kanser Veri kümeleri ile Sınıflama Problemi	61
4. RADYAL TABANLI İŞLEVLERİ KULLANARAK SINIFLAMA ve REGRESYON KESTİRİMİ	63
4.1. Problem Tanıtımı	63
4.2. RTİ Dönüşümü Kullanarak Sınıflama	64
4.2.1. RTİ Dönüşümü Kullanarak Sınıflama İçin Çözüm Yöntemi	66
4.3. RTİ Dönüşümü Kullanarak Regresyon Kestirimi	67
4.3.1. RTİ Dönüşümü Kullanarak Regresyon Kestirimi İçin Çözüm Yöntemi	68
4.4. Uygulamalar	70
4.4.1. WGK, CKH ve İyonosfer Veri kümeleri ile Sınıflama Problemi	70
4.4.2. İki Spiralli ve Dört Spiralli Veri kümeleri ile Sınıflama Problemi	71
4.4.3. Sinc İşlevi ile Regresyon Kestirimi	76
5. ÇEKİRDEK İŞLEVLERİ KULLANARAK SINIFLAMA ve REGRESYON KESTİRİMİ	78

5.1. Doğrusal Olarak Ayırıştırma Yapan Sınıflayıcı İçin Önerilen Formülasyon	78
5.1.1. Önerilen Doğrusal Sınıflayıcı Formülasyonunun Çözüm Yöntemi	79
5.2. Çekirdek Tabanlı Sınıflayıcılar İçin Önerilen Formülasyon	80
5.2.1. Çekirdek Tabanlı Sınıflayıcılar İçin Önerilen Formülasyonun Çözüm Yöntemi	81
5.3. m-Sınıflı Sınıflama İçin Önerilen Çekirdek Tabanlı Formülasyon	82
5.4. Regresyon Kestirimi İçin Önerilen Çekirdek Tabanlı Formülasyon	83
5.4.1. Regresyon Kestirimi İçin Önerilen Çekirdek Tabanlı Formülasyonun Çözüm Yöntemi	84
5.5. Uygulamalar	84
5.5.1. CKH, BUPAKB ve İyonosfer Veri Kümeleri ile Sınıflama Problemi	84
5.5.2. Zambak Çiçeği Veri Kümesi ile İki Sınıflı Sınıflama Problemi	87
6. YAPISAL ve DENEYSEL RİSKİ ENAZLAYAN BULANIK MODELLER	92
6.1. Bulanık Tabanlı Modeller	92
6.2. Ele Alınan Bulanık Tabanlı Modelin Yapısı	94
6.3. Regresyon Kestirimi İçin Bulanık Tabanlı Modellerde Önerilen Öğrenme Algoritması	96
6.4. Uygulamalar	99
6.4.1. Sinc İşlevi ile Regresyon Kestirimi Problemi	99
6.4.2. Mackey-Glass Zaman Serisi ile Öngörüm Problemi	101
7. SONUÇLAR	102
KAYNAKLAR	106
EK-1. İSTATİSTİKSEL ÖĞRENME KURAMI	E1-1
E1.1. Örneklerden Öğrenme Kavramı	E1-1
E1.1.2. Deneysel Riskin Enazlanması	E1-2
E1.2. Öğrenmeye Felsefi Yaklaşım	E1-3
E1.3. Temel Niceliklerin Tanımı	E1-4
E1.3.1. Öğrenme İşleminin Tutarlılığı	E1-4
E1.3.2. Öğrenme İşleminin Yakınsama Hız Oranı	E1-5
E1.3.3. VC-boyutu	E1-6
E1.3.4. Öğrenme İşleminin Genelleme Yeteneği	E1-7
E1.3.5. Yapısal Riskin Enazlanması	E1-8
EK2. DESTEK VEKTÖR MAKİNELERİ	E2-1
E2.1. DVM'lere Giriş	E2-1
E2.2. En Uygun Ayırıştırıcı Çok Boyutlu Düzlem	E2-2

E2.2.1. Doğrusal Olarak Ayrılabilen Durumlar	E2-2
E2.2.2. Doğrusal Olarak Ayrılamayan Durumlar	E2-4
E2.3. Çekirdek Tabanlı DVM’ler	E2-5
E2.4. Çok Sınıflı DVM’ler	E2-9
E2.5. Regresyon Kestirimi İçin DVM’ler	E2-10
E2.6. Uygulama Noktaları	E2-12
EK-3. DESTEK VEKTÖR MAKİNE ÇEŞİTLERİ	E3-1
E3.1. Mangasarian’ın Destek Vektör Makine Formülasyonları	E3-1
E3.1.1. Doğrusal Destek Vektör Makine Sınıflayıcılar	E3-2
E3.1.2. LDVM, NLDVM, DDVM ve NDVM Sınıflayıcılar	E3-4
E3.1.2.1. Lagrangian Destek Vektör Makine (LDVM) Sınıflayıcılar	E3-4
E3.1.2.2. Sonlu Newton Lagrangian Destek Vektör Makine (NLDVM) Sınıflayıcılar	E3-5
E3.1.2.3. Düzgün Destek Vektör Makine (DDVM) Sınıflayıcılar	E3-6
E3.1.2.4. Sonlu Newton Destek Vektör Makine (NDVM) Sınıflayıcılar	E3-7
E3.1.2.5. Yakınsal Destek Vektör Makine (YDVM) Sınıflayıcılar	E3-8
E3.2. Joachims’in Destek Vektör Makine Formülasyonu	E3-10
ÖZGEÇMİŞ	

ŞEKİLLER LİSTESİ

	Sayfa
Şekil 2.1 (a) RTİ çekirdek ve (b) ikinci dereceden polinomsal çekirdek için öz nitelik uzayındaki mesafelere karşı giriş uzayındaki mesafe.	12
Şekil 2.2 (a) Daire biçimli çekirdek ve (b) elips biçimli çekirdek için giriş uzayındaki mesafelere karşı öz nitelik uzayındaki mesafe.	14
Şekil 2.3 Fisher'in zambak çiçeği test kümesinin sadece ilk iki karakteristiğine ait verilerin gösterimi. İlk sınıfa ait veriler yıldızlar ile diğer sınıfa ait veriler üçgenler ile gösterilmiştir.	16
Şekil 2.4 Daire biçimli sınıflayıcı.	16
Şekil 2.5 Elips biçimli sınıflayıcı.	17
Şekil 2.6 Aşama-I'de önerilen sınıflayıcının MUÖEİ algoritması ile eğitim başarımı ve karar yüzeyleri.	33
Şekil 2.7 Aşama-II'de önerilen eniyileme probleminin MUÖEİ algoritması ile eğitim başarımı ve karar yüzeyleri.	35
Şekil 2.8 RTİ çekirdekli DVM sınıflayıcının karar yüzeyleri.	37
Şekil 2.9 Aşama-I'de önerilen eniyileme probleminin MUÖEİ algoritması ile eğitim başarımı.	44
Şekil 2.10 Aşama-II'de önerilen eniyileme probleminin MUÖEİ algoritması ile eğitim başarımı.	45
Şekil 2.11 Aşama-I'de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.	50
Şekil 2.12 Aşama-II'de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.	50
Şekil 2.13 Aşama-I'de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.	50
Şekil 2.14 Aşama-II'de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.	51
Şekil 2.15 Aşama-I'de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.	51
Şekil 2.16 Aşama-II'de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.	51
Şekil 3.1 Üyelik işlevlerinin elde edilmesi için merkez yönteminin geometrik gösterimi.	58

Şekil 3.2	YV22, yapay veri kümesi için bulanık elipsoit biçimli sınıflayıcıların sonuçları.	59
Şekil 3.3	MUÖEİ algoritması ile önerilen bulanık sınıflayıcının eğitim başarımının değişimi.	59
Şekil 4.1	(a) Dört spiralli veri kümesi ve (b) iki spiralli veri kümesi için $r=57$ ve 58 ile RTİA yüzeyi kullanarak oluşturulan sınıflayıcıların karar yüzeyleri ve eğitim başarımı. Eğitim noktaları * (bir sınıf) ve + (diğer sınıf) ile gösterilmiştir.	74
Şekil 4.2	(a) Dört spiralli veri kümesi ve (b) iki spiralli veri kümesi için genel DVM sınıflayıcıların karar yüzeyleri. Eğitim noktaları * (bir sınıf) ve + (diğer sınıf) ile gösterilmiştir. Elde edilen destek vektörler ise daire içerisine alınmıştır.	75
Şekil 4.3	(a) $r=20$ ile RTİA yüzeyi kullanarak oluşturulan regresyon kestirimi formülasyonunun yaklaşım sonuçları: ε -duyarsız bölgesi (-.-), gürültülü eğitim verileriyle elde edilen eğri (-.) ve test verilerini kullanarak hesaplanan eğri kalın düz çizgi ile gösterilmiştir, (b) eğitim başarımı.	77
Şekil 5.1	(a) İyonosfer, (b) BUPAKB ve (c) CKH veri kümeleri üzerinde, önerilen yöntemin 10 katlı eğitim başarımlarından rasgele alınan bir sonuç.	87
Şekil 5.2	Zambak çiçeği verisinin (a) 1.sınıfı, (b) 2. sınıfı ve (c) 3. sınıfı üzerinde önerilen çekirdek tabanlı sınıflayıcıların eğitim ve test başarımları – Verilen bir sınıfa ait eğitim noktaları (*) ve test noktaları (x) ile gösterilirken, diğer sınıfa ait eğitim noktaları (o) ve test noktaları ise □ ile gösterilmiştir.	91
Şekil 6.1	Bulanık tabanlı modelin karar verme düzeni.	93
Şekil 6.2	Yaygın olarak kullanılan üç bulanık model yapısı [83].	94
Şekil 6.3	Ele alınan Takagi-Sugeno bulanık tabanlı model.	95
Şekil 6.4	Önerilen bulanık tabanlı modelin öğrenme algoritması.	97
Şekil 6.5	Önerilen bulanık tabanlı sistem ile regresyon kestirimi, (a) eğitim sonuçları, eğitim verileri düz çizgi, ε -duyarsız bölge kesikli çizgi, hesaplanan eğri kalın düz çizgi ile gösterilmiştir (b) test sonuçları.	100
Şekil 6.6	Bilinen bulanık tabanlı sistem ile regresyon kestirimi, (a) eğitim sonuçları, eğitim verileri düz çizgi, hesaplanan eğri kalın düz çizgi ile gösterilmiştir (b) test sonuçları.	100
Şekil 6.7	Kaotik Mackey-Glass zaman serisi.	101

Şekil 6.8 x(501)~x(1000) zaman serisinin benzetim sonuçları (a) $\varepsilon=0.02$ ve 103
öbekleme algoritması kullanarak 10 kuralla oluşturulan bulanık modelin
eğitim ve test başarımları, ε -duyarsız bölge (:), hesaplanan (-.) ve gerçek
değerler düz çizgi ile gösterilmiştir. (b) eğitim ve öngörüm hataları.

TABLÖLER LİSTESİ

	Sayfa
Tablo 2.1 Farklı eğitim algoritmalarının kıyaslanması.	30
Tablo 2.1 Farklı eğitim algoritmalarının kıyaslanması.	31
Tablo 2.2 Önerilen sınıflayıcıların eğitim başarımının küre biçimli ve RTİ çekirdekli DVM ile kıyaslanması.	36
Tablo 2.3 Önerilen sınıflayıcıların ve DVM'nin 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.	38
Tablo 2.4 Önerilen sınıflayıcıların, farklı eğitim algoritmaları kullanılarak elde edilen 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.	41
Tablo 2.5 Önerilen sınıflayıcıların diğer sınıflayıcılar ile 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.	42
Tablo 2.6 Denektaşı veri kümeleri üzerinde BKB, BKD ve önerilen yöntemin kıyaslanması.	49
Tablo 3.1 Önerilen sınıflayıcıların 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.	61
Tablo 3.2 Deneylerde kullanılan veri kümesinin özeti	61
Tablo 3.3 Kanser veri kümeleri için önerilen sınıflayıcının ve DVM'nin doğruluklarının kıyaslanması.	62
Tablo 4.1 RTİA yüzeyi kullanarak oluşturulan sınıflayıcıların ve DVM sınıflayıcıların denektaşı veri kümeleri üzerindeki başarımları.	71
Tablo 4.2 RTİA yüzeyi kullanarak oluşturulan sınıflayıcıların, DVM sınıflayıcıların ve RTİA sınıflayıcıların denektaşı veri kümeleri üzerindeki başarımları.	72
Tablo 5.1 Önerilen çekirdek tabanlı sınıflayıcı ile NLDVM, DVMlight ve DVM'nin 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.	86
Tablo 5.2 Zambak çiçeği veri kümesi üzerinde önerilen çekirdek tabanlı sınıflayıcı ile DVM'nin 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.	88
Tablo 6.1 Önerilen bulanık modelin sinc işlevi üzerinde eğitim ve test başarımı.	100
Tablo 6.2 Mackey-Glass zaman serisinin son 300 verisinin öngörümü problemi üzerinde, farklı yöntemlerin başarımlarının kıyaslanması.	102

KULLANILAN BAZI TERİMLER LİSTESİ

Literature	Bilimsel yazın
Margin	Pay
Kernel	Çekirdek
Feature space	Öznitelik uzayı
Hyperplane	Çok boyutlu düzlem
Minimization	Enazlama
Maximization	Ençoklama
Optimization	Eniyileme
Optimal	En uygun
Support vector	Destek vektör
Empirical error	Deneysel (ampirik) hata
Robust	Gürbüz
Clustering	Öbekleme
Global	Tümel

KISALTMALAR LİSTESİ

Destek Vektör Makine	DVM
Vapnik-Chervonenkis boyutu	VC boyutu
Radyal Tabanlı İşlev	RTİ
Radyal Tabanlı İşlev Ağları	RTİA
Yapay Sinir Ağları	YSA
Genel Eğitim İniş	GEİ
Uyarlanır Öğrenme Oranlı Eğitim İniş	UÖEİ
Momentum terimli Uyarlanır Öğrenme Oranlı Eğitim İniş	MUÖEİ
Eşlenik Eğitim İniş	EEİ
Ölçeklenmiş Eğitim İniş	ÖEİ
Ortalama Karesel Hatanın Karekökü	OKHK
Yarı-Newton yöntemi	YN yöntemi
Bire Karşı Bir yöntemi	BKB yöntemi
Bire Karşı Diğerleri yöntemi	BKD yöntemi
Lagrangian Destek Vektör Makine	LDVM
Sonlu Newton Lagrangian Destek Vektör Makine	NLDVM
Düzgün Destek Vektör Makine	DDVM
Sonlu Newton Destek Vektör Makine	NDVM
Yakınsal Newton Destek Vektör Makine	YDVM
Küre Çekirdekli Destek Vektör Makine	KÇDVM
Bulanık Destek Vektör Makine	BDVM
k-En yakın Komşuluk algoritması	k-EK algoritması
İstatiksel Öğrenme tabanlı Bulanık Modeller	İÖBM
Bulanık Elipsoit biçimli Sınıflayıcılar	BES
Genel Elipsoit biçimli Sınıflayıcılar	GES
Karush-Kuhn-Tucker teoremi	KKT teoremi

SEMBOLLER LİSTESİ

x	Giriş vektörü
y	Etiket vektörü
R	Yarıçap
c	Merkez vektörü
Σ	Kovaryans matrisi
A	Penaltı parametresi
$F(.)$	Amaç ölçütü
$f(.)$	Kayıp işlevi
$\ell(.)$	Karar işlevi
L	Eğitim veri sayısı
M	Sınıf sayısı
σ	Gauss işlevinin genişliği
ξ	Yapay hata değişkeni
Δ	Pay
$L(.)$	Lagrangian işlevi
w	Gerçek ağırlık vektörü
b	Orijinden kayıklık oranı
λ	Lagrange çarpanı
K	Çekirdek matrisi
$\phi(.)$	Öznitelik uzayına dönüşüm işlevi
C	DVM için düzenleme sabiti
ε	Duyarsız bölgenin genişliği
η	Öğrenme oranı
∇	Gradyen matrisi
H	Hessian matrisi
s	Bulanık üyelik değeri
β	Ölçeklenmiş ağırlıklar

ÖZET

Doktora Tezi

DESTEK VEKTÖR MAKİNE TABANLI BULANIK SİSTEMLER, YENİ BİR GÜRBÜZ SINIFLAYICI VE REGRESÖR TASARIMI

Ayşegül UÇAR

Fırat Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği

Anabilim Dalı

2006, Sayfa: 113

Bu tezde; giriş uzayında Destek Vektör Makinelere (DVM'lere) seçenek sınıflayıcıların ve regresörlerin tasarımı yapılmıştır. Bu amaçla, eğitim iniş yöntemleri kullanılarak çözümü yapılabilen eniyileme algoritmaları sunulmuştur.

Bu çalışmada ilk olarak, giriş uzayında DVM'lerdeki gibi hem yapısal hata hem de deneysel hata prensibine dayanarak karar yüzeyleri küre ve elipsoit biçimli olan iki sınıflı sınıflayıcı algoritmaları önerilmiştir. Bu algoritmalar çok sınıflı sınıflama problemlerine basit olarak genişletilmiştir. Önerilen tüm algoritmaların, gürültü ve aykırı verilere karşı gürbüzlüğünü artırmak için, her veriye farklı bir üyelik değeri atanarak yeni bulanık küre ve elipsoit biçimli sınıflayıcılar elde edilmiştir. İkinci olarak, küre ve elipsoit biçimli karar yüzeylerinin dışında, Radyal Tabanlı İşlev Ağlarının (RTİA'ların) veya klasik bir gizli katmanlı Yapay Sinir Ağlarının (YSA'ların) yüzeyleri gibi daha karmaşık karar yüzeyleri kullanılarak sınıflama başarımını arttırmayı amaçlayan bir algoritma önerilmiştir. Önerilen algoritma, regresyon kestirimi için de genişletilmiştir. Üçüncü olarak, RTİA'lar kullanılarak elde edilen sınıflayıcı ve regresör formülasyonları, çekirdek kullanılarak yeniden türetilmiştir. Son olarak, bulanık sistemlerin genelleme yeteneğini artırmak amacıyla, DVM'nin istatistiksel öğrenme algoritmasına benzer bir öğrenme algoritması sunulmuştur.

Önerilen tüm algoritmalarda, penaltı parametresi uyarlanır seçilerek gürültü ve aykırı verilere karşı daha az duyarlı olma özelliği sağlanmıştır. Algoritmaların, bilimsel yazındaki birçok yöntemle göre başarımı farklı veri kümeleri üzerinde eğitim süresi ile eğitim ve test hataları açısından karşılaştırılmıştır.

Anahtar Kelimeler: Destek vektör makineler, bulanık mantık, penaltı işlev yaklaşımı, eğitim iniş algoritmaları, sınıflama ve regresyon kestirimi.

ABSTRACT

PhD Thesis

SUPPORT VECTOR MACHINES BASED FUZZY SYSTEMS, A NEW CLASSIFIER AND REGRESSOR DESIGN

Ayşegül UÇAR

Firat University

Graduate School of Natural and Applied Sciences

Department of Electrical and Electronics Engineering

2006, Page: 113

In this thesis, new classifiers and regressors in the input space as alternatives to Support Vector Machines are designed. For this aim, new optimization algorithms that can be solved using gradient descent methods are proposed.

Firstly, based on the structural and empirical error principles such as SVMs, two class classification algorithms with spherical and ellipsoidal decision surfaces are proposed. The classifiers are extended simply to multi-class classification. To increase robustness against noise and outliers of all proposed algorithms, new fuzzy spherical and ellipsoidal classifiers are constructed by assigned a membership to each point. Secondly, by using more complex decision surfaces such as that of Radial Basis Function Network (RBFN) or one hidden layer Artificial Neural Networks (ANN) except from sphere and ellipse decision surfaces, an algorithm increasing the classifier performance is proposed. The algorithm is extended to regression estimation. Thirdly, classifier and regressor formulations obtained by using RBFN are again derived by using kernel functions. Finally a similar learning algorithm to statistical learning algorithm of SVM is introduced to increase the generalization performance of fuzzy models.

In all proposed algorithms, penalty parameter is determined iteratively. Thus less sensitivity against noise and outlier data is achieved. The performances according to many methods in the literature of the algorithms are evaluated in terms of training time and training and testing correctness on different benchmark data sets.

Keywords: Support vector machines, fuzzy logic, penalty function approach, gradient descent methods, classification, and regression estimation.

1. GİRİŞ

Ölçüm verilerine uygun modellerin kurulması, etiketli veya etiketsiz verilerin karşı düştüğü sınıfların belirlenmesi ve örnek verilere, seçilen bir uzaydaki, en uyan işaretin veya işlevin belirlenmesi; örüntü tanıma, işaret işleme ve kontrol gibi birçok alanın en temel problemleridir. Belirtilen problemler, “verilerle tanımlı bir bağıntıya biçimi bilinen veya öğrenilen bir işlevin parametrelerinin belirlenmesi yoluyla yaklaşılmaları” problemine dönüştürülebilir. İstatistiksel yöntemler, işaret dönüşüm yöntemleri, Yapay Sinir Ağlarına (YSA’lara) dayalı yöntemler, bulanık mantığa dayalı yöntemler farklı türden bağıntılar sağlarlar. Bu tezde incelenen Destek Vektör Makineler (DVM’ler), var olan yöntemlere etkin seçenekler sunan ve günümüzde yapı, tasarım ve uygulama açılarından sürekli gelişim içinde olan doğrusal olmayan cebrik modellerdir.

DVM’ler yapısal olarak, düşük boyutlu bir giriş uzayından alınan vektörleri, yüksek boyutlu bir diğer uzaya doğrusal olmayan bir biçimde taşıyan bir dönüşümdür. Dönüşümü gerçekleyen makine, sistem ya da ağ, dönüşümü belirleyen bir çekirdek ile tanımlanır. Sınıflama problemlerinde, yeteri kadar yüksek boyutlu uzaya taşınan vektörler doğrusal ayrıştırılabilir duruma gelir. En uygun doğrusal ayrıştırıcı, ayrıştıran düzlemler arasında sınıflara uzaklığı en çok olanıdır. Pay olarak adlandırılan en yakın uzaklık; yüzeye en yakın olan vektörlerin belirlenmesi ile bulunur. DVM’ye de adını veren ve destek vektörler olarak adlandırılan bu vektörler, ayrıştıran düzlemi belirler ve DVM’lerin tasarımı için etkin bir yol sunarlar.

Temelleri istatistiksel öğrenme kuramına dayanan DVM’ler, örüntü tanıma ve regresyon problemlerinin çözümünde sağlam ve etkin bir yöntem olarak kullanılmakta ve birçok uygulama alanında ümit veren bir yöntem olarak durmaktadır [25].

Örüntü tanıma alanının geleneksel yöntemleri, eğitim kümesi üzerindeki başarımın ençoklanmasına ve böylece deneysel riskin enazlanmasına dayanırlar. DVM’ler deneysel ve yapısal risklerin ikisini de enazlayacak şekilde eğitilirler. DVM’lerin tasarımında genelleme hatası için verilen bir üst sınır enazlanır.

Yaygın kullanım bulan ve etkin bir yöntem sunan DVM’lerin, yapısal olarak geliştirilme süreci devam etmekte ve artan bir eğilimle yeni uygulama alanlarında denenmektedir. DVM’lerin uygulanmasında, aşağıda belirtilen dolayısıyla geliştirilmesi gereken 5 zayıf yan bulunmaktadır.

Birincisi, DVM’lerin asıl olarak geliştirildikleri iki sınıflı problemlerin [22, 101], çok sınıflı problemlere etkin olarak genişletilmesidir. Bir sınıfa karşı diğer sınıf, bir sınıfa karşı diğer sınıflar ve yönlendirilmiş çevrimsiz çizge DVM’ler bu yöndeki genişletmelere örnek olarak verilebilir [46, 65, 76]. DVM’lerin çok sınıflı sınıflama başarımını arttırmak için,

öncelikle ikili sınıflama yapan DVM'lerin başarımını arttırmak gerekir. Çok sınıflı problemlere doğrudan çözüm öneren formülasyonlar genelde iyi bir başarımlı vermemektedir [106].

İkinci yan, iki sınıflı sınıflamada aşırı uygunluk problemidir. [47]'de belirtildiği gibi, DVM'ler gürültü ve aykırı verilere çok duyarlıdır.

Üçüncü yan, hesaplama ve bellek gereksiniminin eğitim kümesinin eleman sayısı ile orantılı büyümesi ve bu nedenle çözümün çok yavaş olmasıdır [22, 51]. Büyük boyutlu bir karesel programlama probleminin çözümünü gerektirdiği için, DVM'lerin büyük veri kümelerine uygulamaları sınırlı kalır.

Dördüncü yan, veriler çok boyutlu uzaya aktarılırken uzaklık sırasının korunması için, çekirdek ve çekirdek parametresinin seçiminde sorunlarla karşılaşılmasıdır [27, 70, 110]. DVM'ler, yüksek boyutlu uzayda en yakın iki veri arasındaki uzaklığı en çoklayarak sınıflama yapar, ancak birçok çekirdek uzaklık sırasını korumaz, çekirdek parametresi de hesaplanan uzaklığın büyüklüğünü değiştirir. Bu nedenle uygun çekirdek ve çekirdek parametresi seçilmediğinde, yüksek boyutlu uzayda uzaklık sırası korunmadığından veya uzaklıklar arası farklar küçüldüğünden hatalı sınıflamaya yol açılır. Bu problemi gidermek için [3]'te yeni bir çekirdek önerilmiştir.

Beşinci bir yan, DVM'lerin eniyileme problemi olarak formüle edilen tasarım yöntemlerinde, amaç ölçütü içerisinde bulunan penaltı katsayısının uygun değerinin belirlenmesinin başarımı çok etkilemesidir [88].

Bu tezde DVM'nin yukarıda adı geçen zayıf yanlarının yol açtığı problemlerin 5'ini de çözmeye yönelik iki farklı çalışma yapılmıştır. Birinci çalışma, istatistiksel öğrenme kuramına dayanan DVM'lere seçenек olacak yeni sınıflayıcılar oluşturmaktır. İkincisi ise; bu problemleri gidermek için DVM'lerin tasarımının dayandığı eniyileme formülasyonunda ve karşı düşen eğitim algoritmasında değişiklikler yapmaktır.

Birinci çalışmada, DVM'ye seçenек olacak sınıflayıcıların öznitelik uzayı yerine giriş uzayında oluşturulmasıyla, veri yapısının korunması probleminin çözülmesi amaçlanmıştır. Çalışmada, ayrıştırma yüzeyleri olarak, geçmişten beri yaygın olarak kullanılan ve birçok uygulamada yüksek doğrulukla karar vermeyi sağlayan küreler ve elipsoidler kullanılmıştır.

Küre ve elipsoide dayalı sınıflayıcılar, bilimsel yazında birçok araştırmaya konu olmuş ve birçok sınıflama probleminde başarıyla uygulanmışlardır. [74, 81]'de geliştirilen küre biçimli sınıflayıcılarda, her bir sınıf için birden fazla küre kullanılmıştır. Rosen 1965'te, ayrıştırma yüzeyleri olarak elipsoidler kullanmış, en uygun sınıflayıcının tasarımını bir eniyileme problemi olarak önermiştir [84]. Çalışmasında her bir sınıfın örnek örüntülerini içeren en küçük hacimli elipsoid bulmayı amaçlamıştır. Elipsoidin kovaryans matrisinin pozitif yarı tanımlılığı üzerinde hiçbir kısıt kullanmamıştır. Barnes 1982'de, Rosen'in formülasyonu üzerinde Lagrange

çarpanları yöntemini kullanarak kendi yöntemini oluşturmuştur [5]. Açık bir şekilde elipsoidin kovaryans matrisi üzerine pozitif yarı tanımlılık kısıtını eklemiştir. Verinin ayırma sınırının dışında olabilmesi için herhangi bir şart vermemiş veya herhangi yapay değişken tanımlamamıştır. İki boyutlu 7 örüntü içeren küçük boyutlu bir örnek üzerinde uygulama yapmıştır.

1999'da Tax ve Duin, verilen bir veri kümesini tanımlamak için öz nitelik uzayında en küçük hacimli bir küre oluşturmuştur [95, 96]. Burada; penaltı yaklaşımı kullanılarak, hata ve en küçük hacim arasında bir denge kurulmuştur. Önerilen formülasyon, aykırı veri bulma veya tek sınıflı sınıflamaya uygulanmıştır. 2005'te Wang ve diğerleri, Tax ve Duin'in çalışmasından yola çıkarak tek bir küre ile öz nitelik uzayında dairesel sınıflama yapmıştır [104]. Hacim enazlanmasının yanında payın ençoklanmasının amaçlandığı bu çalışmada: California, Irvine Üniversitesi'nin (UCI) makine öğrenme veritabanları arşivinden [72] Wisconsin isimli göğüs kanseri verileri (WGK), İyonosfer radar yansımaları verisi (İyonosfer), Hindistan'daki Pima yerlileri içerisindeki şeker hastaları verileri (Pima), Liver ve Sonar verileri üzerinde elde ettikleri deneysel sonuçlar verilmiştir.

Glineur 1998'de, elipsoitler kullanarak örüntü ayırma problemi için formülasyonlar önermiştir [39]. Bu formülasyonların, ardışıl karesel doğrusal konik programlama tekniği (SQL conic) ile çözülebilmesi için problemi, $n+1$ boyutlu uzaydaki bir probleme dönüştürmüştür. En Büyük Ayırma Oranı (EBAO) formülasyonunda, aynı merkezli iki elipsoit oluşturmuştur, $(a_i - c)^T \Sigma (a_i - c) \leq 1 \quad \forall_i$ ve $(b_j - c)^T \Sigma (b_j - c) \leq \rho^2 \quad \forall_j$. İkinci elipsoit, birincisinden ayırma oranı kadar geniş seçilmiş ve ayrıştırıcı elipsoitlerin kümesi üzerinden ayırma oranı en büyük yapılmak istemiştir. Oluşturulan iki elipsoidin ortasından geçen aynı merkezli elipsoit ayrıştırıcı yüzey olarak alınmıştır. Bu yöntemin dezavantajı, hiçbir ayrıştırıcı elipsoit bulunmadığında (elipsoidin dışında olması gereken veriler içinde ise veya içinde olması gereken veriler dışında ise), yöntemin tüm örüntü uzayını kapsayacak şekilde bir ayrıştırıcı elipsoit, yani,

$$\tilde{\Sigma} = \begin{bmatrix} 1 & 0_{1 \times n} \\ 0_{n \times 1} & 0_{n \times n} \end{bmatrix} \text{sonucunu bulmasıdır. Bu dezavantajdan kurtulmak için, En Küçük Hacimli}$$

Elipsoit (EKHE) formülasyonunu önermiştir. EKHE'de bir elipsoidin hacmine denk olan yarı eksenlerinin uzunluğunun karesi enazlanmıştır. EKHE'nin başarısızlığı, sadece $(a_i - c)^T \Sigma (a_i - c) \leq 1$ kısıtının kullanılması ve tüm b_j 'lerin elipsoidin dışında bulunmasının gerekliliğidir. Sadece bir b_j örüntüsünün, elipsoidin içinde olması, algoritmanın yanlış bir çözüm yapması için yeterlidir. Bu dezavantajdan kaçınmak için, basitçe b_j 'lerin çoğunun elipsoidin dışında olması kısıtını içeren yeni bir yöntem önermiştir. Kısıtı modellemek için, a_i 'ler ve her bir b_k arasındaki ayırma oranları ayrı ayrı hesaplanmış, bu ayırma oranlarının

aritmetik ortalamasının mümkün olduğu kadar büyük olması sağlanmıştır. Bu, ρ_j 'lerin toplamını veya onların karelerinin toplamını en büyük yapmaya denktir. Bu yöntemin ise gerçekte hiçbir ayırıştırıcı elipsoit olmadığı zaman, sonuç bulamadığı ve özellikle karelerinin toplamı alındığında, küçük ayırma oranlarını en çok yapmaktan ziyade, daha çok olanları en çok yaparak ayırma oranlarını artırdığını göstermiştir. Böylece Karesi alınmış Ayırma Oranlarının Toplamını Enazlayan (KAOTE) yeni bir yöntem önermiştir. Bu yöntemde, tüm b_j 'lerin elipsoidin dışında bulunmasını gerektiren kısıt kullanılmasının bir dezavantaj olduğunu belirtmiştir. UCI makine öğrenme arşivinden Zambak çiçeği verileri, WGK, Pima, Boston Housing verileri üzerinde yöntemin başarımı ve bilimsel yazındaki çalışmalarla karşılaştırmalı sonuçlar vermiştir.

2003'te Astorino ve diğerleri, hem öz nitelik uzayında hem giriş uzayında, B gibi bir kümeden A gibi bir kümeyi ayıran (A 'nın tüm noktalarını içeren ve B 'nin noktalarını içermeyen) en küçük hacimli küre bulmayı amaçlamışlardır [1]. Burada; merkez sabitlenerek, formülasyon doğrusal programlama problemi olarak önerilmiştir. Ayrıca iki dönüştürülmüş küme üzerinde çalışarak, öz nitelik uzayında da aynı algoritmanın uygulanabileceği gösterilmiştir. Veriler her zaman küreler ile ayrılmadığı için, problemi daima uygulanabilir kılmak üzere, sınıflama hatasının amaç ölçütü ile hacmin enazlanma ölçütünü birleştirmiştir. Böylece B 'nin bazı verilerinin, A kümesinin verilerini kapsayan küre içerisinde olabilmelerini sağlamak ve aykırı veriler tarafından oluşturulan etkiyi de azaltmak istemiştir. Burada; UCI makine öğrenme arşivinden WGK verisi, Cleveland isimli kalp hastaları (CKH) verisi, İyonosfer verisi, Mantar verisi, Tic-Tac-Toe Endgame (Tic-Tac-Toe) verisi ve [75]'deki galaksi yıldız ayırıştırma (GYA) verisi kullanılarak elde edilen sonuçlar, bilimsel yazındaki DVM temelli yaklaşımlarla karşılaştırmalı olarak verilmiştir. Aynı yazarlar tarafından 2005'te, sadece elipsoitlerle ayrılabilir verilerin giriş uzayında ayrılması amaçlanmıştır [2]. Bu nedenle elipsoidi tanımlayan parametrelere bağlı olan hata işlevi enazlanmıştır. Böylece ayırıştırma tam olarak gerçekleştiği zaman, sıfıra eşit olan bir amaç ölçütü elde edilmiştir. Sınıflamada kayıp işlevi olarak, "maks" alınmıştır. Eniyileme problemi, düzgün olmayan eniyilemeden de faydalanılarak yerel arama türünden bir algoritma ile çözülmüştür. Merkezler öncelikle sabit alınmış, daha sonra en dik iniş tekniği kullanan bir algoritma ile güncellenmiştir. Algoritma, UCI makine öğrenme arşivinden WGK verisi, CKH verileri, BUPA Karaciğer bozuklukları (BUPAKB) verisi ve Amerika Birleşik Devletleri Kongre Oylama Kayıtları örnekleri üzerinde denenmiştir.

Konno ve diğerleri 2002'de, yarı-tanımlı programlama problemi olarak ifade edilen başarısızlık ayırtıcı analizi için yeni bir düzlem kesme algoritması önermiştir [54]. Burada finansal veri, finansal yorum kadar iyi matematiksel özellikleri olan elipsoit biçimli yüzey ile ayrılmıştır. Formülasyonda sınıflama hatasının ağırlıklı toplamı (yani elipsoit biçimli yüzeyden

sapmaları da içeren) esnek elipsoit ayrıştırıcı kısıtlar üzerinden enazlanmıştır. Elipsoit biçimli ayrıştırmanın çok boyutlu düzlemle ayrışmadan, belirgin olarak daha iyi sonuçlara götürdüğü, özellikle DVM yaklaşımında kullanılan yöntem gibi karesel yüzeylerle oluşturulan ayrıştırmadan daha iyi sonuç verdiği gösterilmiştir

Kharechko ve diğerleri 2004'te, Gram-Schmidt dikleştirme tekniği kullanarak ve Glineur'un çalışmasına benzer olarak aynı merkezli iki elipsoit arasındaki ayırma oranını en büyük yapma yöntemi kullanarak, öz nitelik uzayında elipsoit biçimli ayırma gerçekleştirmişlerdir [52]. Bu ayrıştırıcı elipsoitler $n+1$ boyutlu uzayda, kanonik homojen formdadır [52], (yani ayırma oranı aynıdır Σ ve c de basit formüller kullanarak hesaplanabilir). Reuters-21578 veri kümesinin 10 kategorisi üzerinde ayırma sonuçları, DVM ile karşılaştırmalı olarak verilmiştir.

Potra ve diğerleri 2004'te, öngörüm amacı için yarı-tanımlı programlama kullanarak benzer eşmerkezli elipsoitler veya doğrusal programlama kullanarak iki paralel çok boyutlu düzlem kurmuştur [79]. Elipsoit biçimli ayrıştırmayı iki aşamalı olarak yapmıştır. İlk aşamada, Glineur'un yöntemine benzer olarak iki eş merkezli elipsoit kurmuş ve ayırma payını ençoklamıştır. Bu aşamanın Glineur'unkinden farkı, elipsoit denklemini polinomsal olarak vererek İç Nokta yöntemleriyle [88, 108] polinomsal zamanda çözülebilmesidir. İkinci aşamada, yanlış sınıflandırılmış noktaların sayısının enazlanması için ilk bölümde bulunan iki elipsoit arasında üçüncü bir elipsoit bulmuştur.

Bu tezde önerilen küre biçimli ve elipsoit biçimli sınıflayıcılar, bilimsel yazında mevcut olanlardan tamamen farklıdır. Önerilen formülasyonlarda, hacim ve sınıflama hatası aynı anda enazlanmıştır. DVM gibi Lagrange çarpanları yöntemi kullanarak öz nitelik uzayında sınıflama yapan yöntemlerin tersine, önerilen yöntem penaltı parametresi kullanarak giriş uzayında sınıflama yapmaktadır [28]. Böylece fazladan hiçbir yapay değişken kullanılmamaktadır. Çözüm, eğim iniş yöntemi ile yapıldığı için geniş veri kümelerine uygulanabilir. Karesel programlamadan ve [39]'daki konik programlamadan daha hızlıdır. Bunun dışında, Mangasarian ve diğerlerinin [34-36, 58, 66-69]'daki doğrusal denklem takımları ile çözümünü yapabilen DVM tabanlı algoritmalarına hem hız hem de doğruluk olarak yakındır.

İkinci çalışmada, DVM'nin geniş veri kümelerine hızlı bir şekilde uygulanabilmesini sağlamak için, penaltı yaklaşımı kullanarak, birincil formülasyon tabanlı yeni bir eniyileme problemi sunulmuş ve çözümü için eğim iniş yöntemleri önerilmiştir. DVM problemleri için eğim iniş yöntemlerinin uygulanması yeni değildir. DVM'nin doğrudan eğitiminde, şimdiye kadar yapılan çalışmalar ikincil programlama temeline dayanır [22, 34]. Bu durumda da veri sayısı fazla olan sistemlerle ilgilenildiğinde, eğitim süresi ve hesapsal problemler mevcuttur. Chapelle 2002'de, yoğunluk kestirimi problemleri için; en küçük hacimli daire problemlerinde,

parametre hesaplamasında ve sınır tanımlamalarında eğim iniş yöntemlerini kullanmıştır [14]. Chapelle bu tez çalışması ile eş zamanlı olarak ilk defa, [15]'deki çalışmasında birincil biçimi kullanarak sınıflama problemleri için formülasyon önermiştir. Bu tezde ve bu tez sonuçlarının kısmen yayınlandığı [100]'de [15]'den farklı olarak, formülasyonlar hem sınıflama ve hem regresyon için önerilmiştir. Ayrıca yapısal hata ve deneysel hata arasında ödünleşimi sağlayan penaltı sabiti her özyinelemede uyarlanır olarak elde edilmiştir. Böylece hem DVM'lerde seçimi önemli olan bu parametrenin seçiminden kaçınılmış hem de gürültü ve aykırı verilere karşı daha az duyarlı olması sağlanmıştır. Bunun dışında, çok sınıflı sınıflama problemleri için tek bir formülasyon önerilmiştir.

Bulanıklaştırma, insan faktörünün etkili olduğu sistemlerde yararlı sonuçlar vermektedir. Diferansiyel denklemler gibi klasik matematiksel aletler kullanarak sistem modelleme, belirsiz sistemler ve birkaç çözümü olan sistemler ile ilgilenmek için uygun değildir. Karmaşık sistemleri basitleştirmek için kullanılan yöntemlerden biri, modelleme aşamasındaki belirsizlik ve şüphenin mantıklı bir miktarını iptal etmektir. Bu durumda sonuç sistem kesinlikle kusursuz değildir, fakat birçok durumda oluşturulan uygun modeller problemi çözer. Böyle bir belirsizlik, Zadeh'in bulanık mantık sistemleri ile karşılanabilir. Zadeh 1960'ta, geleneksel sistem analizinin birçok gerçek yaşam uygulamasında gereğinden çok kusursuz olduğu duygusunu belirtmiştir. Ardından 1961 yılındaki bir makalesinde, bilinen bir olasılık dağılımına dayalı olmayan tamamen bir önceki bilgi ve pratik deneyimi yansıtan bulanık mantığın temellerini açıklamıştır [83].

İlk olarak sistematik bir şekilde, Takagi ve Sugeno tarafından keşfedilen bulanık modelleme veya tanıma; kontrol, öngörüm ve çıkarım gibi birçok önemli uygulamada kullanılmıştır [55, 57, 83, 105]. Bulanık mantık isminin ifade ettiği gibi, tam olmaktan ziyade yaklaşık mantıktır. Bulanık mantığın önemi, insan mantığının kurduğu çoğu modelin doğal olarak yaklaşık olması gerçeğinde yatar.

Bulanık kuramı, sistemin analizi klasik teknikler ile çok fazla karmaşık ise mevcut bilgi kaynakları belirsiz, tam olmayan ve nitel olarak yorumlanırsa faydalı görünür. Böylece bulanık karar modeli klasik tam matematiksel karar ve insana benzer karar verme arasında uzlaşmaya doğru bir adım olarak bakılabilir.

Bulanık modeli tanımlama işlemi, yapı tanıma ve parametre hesaplama olarak genellikle iki aşamaya ayrılır [4, 18, 21, 57, 83]. Yapı tanıma aşamasında; uygun giriş değişkenleri, giriş değişkenleri arasındaki ilişki, kuralların sayısı, giriş çıkış uzayının bölümlere ayrılması, model parametrelerinin başlangıç değerleri gibi birkaç yapı ögesi belirlenir. Parametre hesaplama aşamasında ise model parametreleri, deneysel risk enazlanarak ayarlanır. Bilimsel yazında öbekleme yöntemi, genetik yöntem, karar ağacı yöntemi, dikgen en küçük kareler yöntemi ve

QR-tekil deęer ayrışım yöntemi gibi birçok yöntem düşünülmesine rağmen, yapı tanıma güçtür ve çok sayıda çözümü vardır [21, 83]. Bu nedenle insan müdahalesi yapı tanımada genellikle gereklidir.

Bilimsel yazında bulanık mantık ve DVM, sınıflama veya regresyon kestirimi için iki farklı şekilde birleştirilmektedir. Birinci şekilde, her bir veri farklı bir üyelik deęerine atanmıştır. [47, 48-50, 62, 71, 97, 107]. İkincisinde ise bulanık sistemlerin yapısının oluşturulması hedeflenmiştir. Chen ve dięerleri 2003'te, üyelik işlevlerinin genel varsayımları altında toplamsal bulanık sistemler ile çekirdek makinelerinin ilişkisini göstererek, verilen eğitim örneklerinden, destek vektör öğrenme yaklaşımı ile bulanık kural tabanlı sınıflayıcı sistemler oluşturmuştur [18]. Uçar ve dięerleri 2003'te sıfırncı ve birinci dereceden bulanık tabanlı sistemler ile DVM'nin eşitliğini çıkararak bulanık sistemlerin başlangıç yapısını oluşturmuş, başarımı arttırmak amacıyla bulanık sistemlerin karma öğrenme algoritmasını kullanmıştır [98]. Uçar ve dięerleri [99]'da öbekleme algoritması kullanarak bulanık sistemler oluşturmuştur. Hao ve dięerleri [2004]'de, [98]'e benzer olarak DVM tabanlı bir bulanık modelleme çerçevesi tanımlamışlardır. Sıfırncı dereceden bulanık tabanlı sistemler ile DVM'lerin eşitliğinden faydalanılarak, DVM öğrenme algoritmasıyla bulanık eęer-ise kurallarını çıkarmışlardır [21]. Fakat bulanık sistemleri tekrar eğitmemişlerdir. Kim ve dięerleri 2006'da, ölçme gürültüsü içeren giriş ve çıkış verisi temeline dayanan doğrusal olmayan dinamik sistemlerin modellenmesi için, olasılık Bayes öğrenme çatısına sahip genişletilmiş ilgililik vektör makineleri kullanarak, eş zamanlı olarak eniyileme uygulayan yeni bir bulanık tabanlı sistem önermiştir [53]. Bu sistemin eğitimi karma öğrenme algoritması ile yapılmıştır. Lin ve dięerleri 2006'da, sınıflama için destek vektör tabanlı bulanık yapay sinir ağı geliştirmiştir. Lin'in çalışmasında, YSA'lar kullanılarak bulanık uyarlanır çekirdek önerilmiştir [63]. Tasarım üç aşamalı olarak gerçekleştirilmiştir. İlk aşamada, başlangıç kural yapısı ve ağı yapısının üyelięi, bulanık öbekleme algoritması ile otomatik olarak oluşturulmuştur. İkinci aşamada, bulanık sistemin katmanları arasındaki ağırlıklar bulanık çekirdekli DVM'nin eğitimi sonucunda elde edilen parametreler kullanılarak hesaplanmıştır. Üçüncü aşamada, gereksiz bulanık kurallar elenerek uygun bulanık kurallar belirlenmiştir. Ancak oluşturulan bulanık sistemlerde, hem deneysel hem de yapısal hata enazlanmamıştır.

Bu tezde iki farklı yaklaşımla, bulanık mantık kavramları kullanılmıştır. Önerilen sınıflayıcı ve regresörlerin iyi bir şekilde genelleştirme yetenekleri ve bulanık küme kuramının insanın düşünmesine benzer olma özelliklerinin birleştirilmesi amaçlanmıştır.

Bu tezde bulanık mantığın içerildięi ilk çalışmada, önerilen küre ve elipsoit biçimli sınıflayıcıların gürültü ve aykırı verilere karşı gürbüzlüğünü artırmak için Bulanık Destek Vektör Makinelere (BDVM'lere) benzer olarak her bir veriye uygun bir üyelik deęeri

atanmıştır. Böylece, yöntemin özellikle aykırı verilere karşı etkisi azaltılmıştır. İkincisinde ise, bulanık tabanlı sistemlerin genelleştirme yeteneğini artırmak, gürültü ve aykırı verilere karşı gürbüz olmasını sağlamak amacıyla başlangıç yapısı bilinen bulanık sistemler için DVM'nin istatistiksel öğrenme algoritması tanıtılmıştır.

1.2. Tezin Organizasyonu

Bölüm 2: Öncelikle giriş uzayında ayırıştırma yapılmasının nedeni incelenerek, yeni küre ve elipsoit biçimli çekirdekler önerilmiş, başarımı ve geçerliliği tartışılmıştır. Daha sonra giriş uzayında iki aşamalı sınıflayıcılar önerilmiştir. İlk aşamada, karar yüzeyinin küre biçimli olduğu kabul edilerek, DVM'lerdeki gibi hem yapısal hatayı hem deneysel hatayı enazlama ilkesine dayanan yeni bir eniyileme problemi önerilmiştir. Çözüm yöntemi olarak eğim iniş yöntemleri önerilmiştir. Farklı eğim iniş yöntemleri, önerilen problemin çözümünde kullanıldığı zaman başarımın ne olacağı örnekler ile tartışılmıştır. İkinci aşamada ise elde edilen başarımı artırmak için karar yüzeyi elipsoitlere genişletilmiştir. Bunun için ilk aşamadan elde edilen merkez ve yarıçap değerleri kullanılarak, ikinci aşamanın başlangıç değerleri oluşturulmuştur. Son olarak, çok sınıflı sınıflama problemleri için tek bir formülasyon, çözüm yöntemi hem küre hem elipsoit biçimli karar yüzeyleri için verilmiştir. Önerilen tüm formülasyonların başarımı eğitim, test hataları ve eğitim süresi açısından farklı veri kümeleri üzerinde elde edilmiş ve bilimsel yazındaki birçok yöntem ile karşılaştırılmıştır. Önerilen küre veya elipsoit biçimli sınıflayıcıların, özellikle eğitim süresi açısından, karesel programlama ile çözümü yapılan genel DVM sınıflayıcılar ile arasında büyük fark olduğu gösterilmiştir. Ayrıca küre veya elipsoit biçimli karar yüzeylerinin, DVM'lerinki gibi karmaşık olmasına rağmen kabul edilebilir bir başarımlar verdikleri gösterilmiştir.

Bölüm 3: Bir önceki bölümde önerilen küre ve elipsoit biçimli sınıflayıcılar, deneysel ve yapısal risk arasında bir ödünleşim sağlayarak sınıflama yaptıkları için gerçekte gürbüz yöntemlerdir. Bunun ötesinde ödünleşim parametresi de her özyinelemede uyarlanabilir olduğu için, genel DVM'lere göre gürültü ve aykırı verilere karşı çok gürbüz olduğu düşünülebilir. Ancak ele alınan veri kümesi, önerilen karar yüzeylerinin merkezlerinden ve her iki sınıftan çok uzakta olan aykırı veriler içerdiği durumda, sınıflama başarımını artırmak için yeni bulanık küre ve elipsoit biçimli sınıflayıcılar bu bölümde tanıtılmıştır. Farklı veri kümeleri ve üyelik işlevleri kullanılarak sınıflama başarımları test edilip tartışılmıştır.

Bölüm 4: Her veri, küre veya elipsoit biçimli karar yüzeyleri ile ayırışmayabilir. Bu nedenle bu bölümde gizli katman uzayı olarak isimlendirilen Radyal Tabanlı İşlev Ağların (RTİA'ların) oluşturduğu yüzeylerin kullanımı amaçlanmıştır. Bu amaçla sınıflama ve

regresyon kestirimi için yeni bir formülasyon önerilmiştir. Örnek uygulamalar üzerinde klasik RTİA ve RTİ çekirdekli DVM'nin başarımları karşılaştırılmış ve tartışılmıştır.

Bölüm 5: Bu bölümde, bir önceki bölümde önerilen yöntem ve formülasyonlar ϕ dönüşümü yerine, çekirdek kullanılarak yeniden türetilmiştir. Elde edilen formülasyonlar eğitim iniş yöntemleri ile çözülmüştür. Böylece karmaşık yüzeyler ile geniş veri kümelerine hızlı bir şekilde uygulanabilme yolu açılmıştır. İkinci olarak, yapısal hata ve deneysel hata arasında ödünleşimi sağlayan penaltı sabiti her özyinelemede uyarlanabilir olarak elde edilerek, hem DVM'lerde seçimi önemli olan bu parametrenin yanlış seçiminin oluşturduğu sorunlardan kaçınılmış hem de gürültü ve aykırı verilere karşı daha az hassas olması sağlanmıştır. Üçüncü olarak, momentum terimli uyarlanırlar öğrenme oranlı eğitim iniş yöntemi kullanılarak, klasik DVM'ye göre hızlı olması sağlanmıştır. Dördüncü olarak, çok sınıflı sınıflama problemleri için tek bir basit formül önerilerek, DVM'lerde halen çözülmeye çalışılan zor bir problem aşılmıştır.

Bölüm 6: Bu bölümde, bulanık sistemlerin genelleştirme yeteneğini artırmak ve gürültü ve aykırı verilere karşı gürbüz olmasını sağlamak için, DVM'nin istatistiksel öğrenme algoritmasına benzer bir öğrenme algoritması önerilmiştir.

Bölüm 7: Sonuç bölümünde, bu tezin bilime yaptığı katkılar özetlenmiş ve gelecek çalışmalar için öneriler ortaya konmuştur.

EK-1: Bu bölümde, öğrenme probleminin formülasyonu tanıtılmış ve eğitim örneklerinin sınırlı sayısı ile öğrenme için istatistiksel öğrenme kuramının temel sonuçları açıklanmıştır.

EK-2: Sınıflama ve regresyon kestirimi için, karesel programlama ile çözümü yapılan genel DVM formülasyonları bu bölümde tanıtılmıştır. Her iki formülasyon için, hem doğrusal hem doğrusal olmayan durumda DVM'nin birincil ve ikincil programlamaları kısaca tanıtılmıştır.

EK-3: Bu bölümde, uygulamaları karşılaştırma amacıyla kullanılan, bilimsel yazındaki yöntemler tanıtılmıştır.

2. GİRİŞ UZAYINDA KÜRE ve ELİPSOİT BİÇİMLİ AYRIŞTIRICI YÜZEYLER KULLANARAK SINIFLAMA

2.1. Destek Vektör Makinelerin Bazı Eksiklikleri

EK-2’de detayları verilen DVM’ler; $(x^1, y^1), \dots, (x^L, y^L)$, $x \in \mathfrak{R}^n$, $y \in \{-1, 1\}$ eğitim örneklerinin L örneği verildiği zaman, verileri yüksek boyutlu uzaya taşıyarak, o uzayda (2.1) ile verilen düzlemi oluşturup sınıflama yapmak üzere tasarlanmıştır,

$$\ell(x) = w^T \varphi(x) + b. \quad (2.1)$$

Burada; $w \in \mathfrak{R}^n$ ve $b \in \mathfrak{R}$ çok boyutlu düzlemi oluşturan parametreler ve $\varphi(x)$ verileri giriş uzayından yüksek boyutlu uzaya dönüştürmek için kullanılan dönüşüm işlevidir [101].

DVM’ler (2.2) ile verilen birincil eniyileme problemini çözerek eğitim hatasını (birinci terim) ve genelleme hatasını (ikinci terim) aynı anda enazlar.

$$L(w, \xi^i) = \frac{C}{L} \sum_{i=1}^L \xi_i + \frac{1}{2} \|w\|^2 \quad (2.2)$$

$$y^i [w^T \varphi(x^i) + b] \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (2.3)$$

Burada; C parametresi genelleme ve eğitim hatası arasındaki ödünleşimi nitelemektedir, ξ_i ise $w^T \varphi(x^i) + b$ ve y^i arasındaki mutlak hatayı göstermektedir.

(2.2)’de $\|w\|^2$ ’nin enazlanması, öz nitelik uzayında zıt sınıfın iki en yakın verisi arasındaki mesafe olarak bilinen payın ençoklanmasına karşılık gelir [EK-2]. Çoğu zaman verileri yüksek boyutlu uzaya dönüştüren $\varphi(x)$ bilinmediğinden, hesaplamalarda kolaylık sağlamak için veriler bu işlevi kullanmadan bir çekirdek yardımıyla örtük olarak yüksek boyutlu uzaya taşınır. Bu amaçla birincil eniyileme problemi, Lagrange çarpanları yöntemi kullanarak ikincil forma çevrilir,

$$L_{ikincil}(\lambda) = -\frac{1}{2} \sum_{i,j=1}^L \lambda_i \lambda_j y^i y^j K(x^i, x^j) + \sum_{i=1}^L \lambda_i \quad (2.4)$$

$$\text{kısıtları: } \sum_{i=1}^L y^i \lambda_i = 0, \quad 0 \leq \lambda_i \leq \frac{C}{L}, \quad i = 1, \dots, L. \quad (2.5)$$

Burada; λ_i Lagrange çarpanlarını ve $K(x^i, x^j)$ çekirdeği $K(x^i, x^j) = \varphi(x^i)^T \varphi(x^j)$ ile hesaplanan iç çarpımı göstermektedir.

Sonuç karar işlevi ise çekirdek ve Lagrange çarpanları kullanarak

$$\ell(x) = \text{sign} \left(\sum_{\text{destek vektörler}} y^i \lambda_i K(x^i, x^j) + b \right) \quad (2.6)$$

ile hesaplanır. Burada; destek vektörler $\lambda > 0$ Lagrange çarpanlarına karşılık gelen x_i değerleridir.

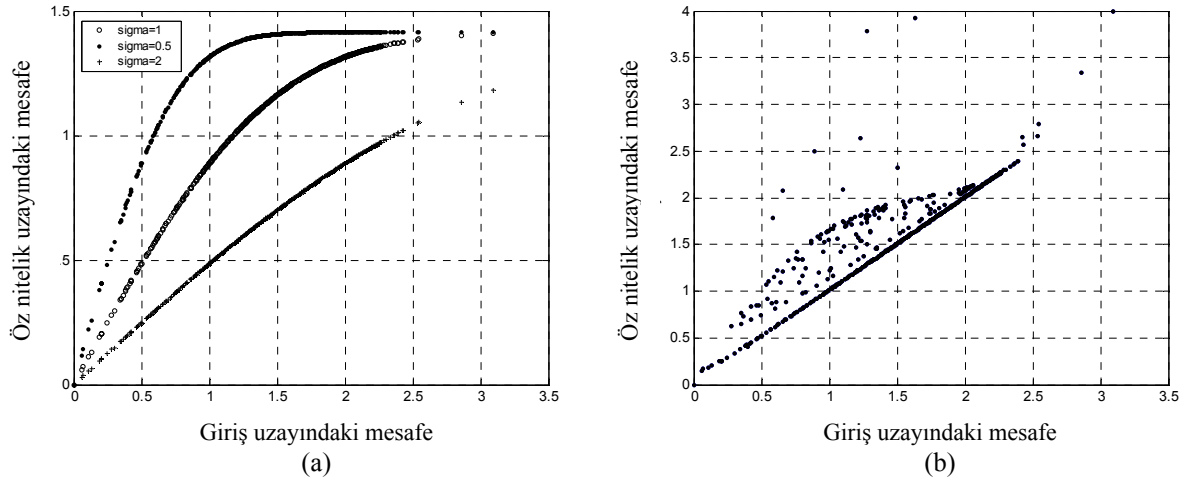
Yüksek boyutlu öz nitelik uzayında her bir örüntü, bir nokta olarak düşünülür ve giriş uzayındaki noktaların bağlı yerini gösterir. Öz nitelik uzayında ayrıştırma yapılırken veri yapısının da korunması gerekir. Veri yapısı, örnekler arasındaki mesafeyi tanımlar ve verinin temelini oluşturan doğal bir özellik olarak düşünülür [70]. Veri yapısının korunması DVM sınıflayıcılarda önemli bir kriter olarak düşünülür [110]. Çünkü DVM, öz nitelik uzayında en yakın veriler arasındaki payı ençoklamayı amaçlar. Eğer veriler öz nitelik uzayında giriş uzayındakine göre farklı yerlerde bulunursa hatalı sınıflama yapılır.

Bu bölümde; giriş ve öz nitelik uzayında veri yapısı incelenerek, DVM'lerin hatalı sınıflama yapıp yapmadığı kontrol edilmiştir. Bu amaçla DVM uygulamalarında en çok kullanılan RTİ ve polinomsal çekirdekler ele alınmıştır. RTİ çekirdek durumunda öz nitelik uzayındaki mesafe,

$$\begin{aligned} \|\phi(x^i) - \phi(x^j)\|^2 &= K(x^i, x^i) - 2K(x^i, x^j) + K(x^j, x^j) \\ &= 2 - 2 \exp\left(-\|x^i - x^j\|^2 / 2\sigma^2\right). \end{aligned} \quad (2.7)$$

ile bulunur [26-28]. Burada; σ giriş uzay mesafesi ile ilgili duyarlılığı düzenleştirir ve seçimi önemlidir. Polinomsal çekirdek durumunda ise, öz nitelik uzayındaki mesafe giriş uzayındaki mutlak mesafeye bağlıdır [56].

Şekil 2.1'de RTİ ve polinomsal çekirdekler için, giriş uzayındaki bir noktaya diğer noktaların mesafesi ile aynı noktanın öz nitelik uzayında diğer noktalara mesafesi arasındaki ilişki verilmiştir. Şekil 2.1a'dan görüldüğü gibi RTİ çekirdek durumunda, sigma parametresi uygun seçilmez ise, giriş ve öz nitelik uzayındaki mesafe arasındaki doğrusal ilişki, mesafe artarken bozularak öz nitelik uzayında veri noktalarını birbirine yaklaştırdığı için sınıflamanın hatalı olma olasılığı artar. Polinomsal çekirdek durumunda ise, öz nitelik ve giriş uzayı mesafeleri arasında ilişki doğrusal değildir. Şekil 2.1b'den görüldüğü gibi, giriş uzayında iki zıt sınıfın en yakın iki verisi ve öz nitelik uzayında en yakın iki veri olmayabilir. Dolayısıyla pay farklı olabilir. Böylece giriş uzayında, gerçek pay ençoklanmadığı için sınıflama hatalı olabilir.



Şekil 2.1 (a) RTİ çekirdek ve (b) ikinci dereceden polinomsal çekirdek için öz nitelik uzayındaki mesafelere karşı giriş uzayındaki mesafe.

Sonuç olarak, yüksek boyutlu uzayda sınıflama yapmak sakıncalı olabilir. Bunun dışında DVM'lerin diğer eksiklikleri ise temel olarak; çekirdek parametrelerinin seçimi, düzenleme sabiti C 'nin seçimi, büyük sayıdaki veriler ile ilgilendiğinde DVM'nin ikinci dereceden programlama kullanılarak bu problemleri çözmemesi veya aşırı hesaplama gerektiği için uzun zaman alması, gürültü ve aykırı veri içeren veri kümelerinde doğru çözüm yapmaması ve başlangıçta ikili sınıflama problemlerini çözmeye yönelik tasarlanması olarak gösterilebilir [64].

Bu tezde, bahsedilen problemlerden kurtulmak için DVM'ye yeni düzeltmeler önermek veya seçenek sınıflayıcılar oluşturmak hedeflenmiştir. Bunun için ilk olarak yeni bir çekirdek oluşturulmuştur.

2.1.1. Küre ve Elipsoit Biçimli Çekirdek Kullanarak Sınıflama

Vapnik Chervonenkis [EK-1] boyutunu enazlayan dolayısıyla genelleme hatasını enazlayan ayrıştırıcı yüzeyler, özelde daireler (küreler) genelde elipslerdir (elipsoitlerdir). [7, 20, 28, 87, 94, 108]'de öbekleme ve tek sınıflı sınıflama problemleri için küre ve elipsoit biçimli yüzeyler kullanılmıştır. DVM ise farklı karar yüzeyleri oluşturarak sınıflama yapabilir. Ancak küre veya elipsoit biçimli karar yüzeylerine sahip ayrıştırıcı yüzeyler kolay yorumlanabilir oldukları için biyoloji, tıp ve benzeri alanlarda öngörüm amaçları için kullanılırlar. Polinomsal çekirdeklerin denklemleri, küre denklemine yakın olmakla birlikte tam olarak bir küre olmadığı için giriş uzayında küreye benzer ayrıştırıcı yüzeyler veremezler. Bu bölümde, DVM sınıflayıcılarda kullanmak için Mercer şartını sağlayan yeni küre ve elipsoit biçimli çekirdekler

bularak, anlamlı yüzeyler elde etmek amaçlanmıştır.

Öncelikle ayrıştırıcı yüzey küre olarak alınırsa, yüzey denklemi

$$a \{ [x - c]^T [x - c] - R^2 \} = 0 \quad (2.8)$$

ile verilir. Burada; a ölçekleme sabitini, $c \in \mathfrak{R}^n$ kürenin merkezini ve $R \in \mathfrak{R}$ kürenin yarıçapını göstermektedir. Denklem açılırsa,

$$a \{ x^T x - 2c^T x + c^T c - R^2 \} = 0 \quad (2.9)$$

$$a \{ x_1^2 + x_2^2 + \dots + x_n^2 - 2c_1 x_1 - 2c_2 x_2 - \dots - 2c_n x_n + c_1^2 + c_2^2 + \dots + c_n^2 - R^2 \} = 0 \quad (2.10)$$

elde edilir. İki boyutlu giriş için daire biçimli karar yüzeyi

$$\begin{bmatrix} a & -2ac_1 & -2ac_2 & ac_1^2 + ac_2^2 - aR^2 \end{bmatrix} \begin{bmatrix} x_1^2 + x_2^2 \\ x_1 \\ x_2 \\ 1 \end{bmatrix} = 0 \quad (2.11)$$

olarak bulunur. (2.11), DVM'nin $w^T \phi(x^i) + b = 0$ yüzeyine benzetilebilir. Burada b eşiği, 1 katsayılı terim olarak düşünülerek sıfır alınabilir. Ancak bu durumda oluşacak çekirdek işlevi Mercer şartını sağlamaz. Mercer şartının sağlanması için (2.11)

$$w^T \phi(x^i) + b = [w_1 \quad w_2 \quad w_3 \quad w_4] \begin{bmatrix} x_1^2 + x_2^2 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \\ 1 \end{bmatrix} + b = 0 \quad (2.12)$$

olarak değiştirilir. (2.12), EK-2'de doğrusal çekirdekli DVM sınıflayıcı için verilen ikinci dereceden programlama problemi ile çözülerek, w ve b parametreleri hesaplanır. Bu değerler kullanılarak ayrıştırıcı yüzeyin R ve c değerleri

$$a = w_1, \quad c_1 = \frac{w_2}{-\sqrt{2}a}, \quad c_2 = \frac{w_3}{-\sqrt{2}a}, \quad (2.13)$$

$$R^2 = (ac_1^2 + ac_2^2 - w_4 - b)/a \quad (2.14)$$

olarak elde edilir. Çözüm çok boyutlu giriş için benzer şekilde genişletilebilir.

Ayrıştırıcı yüzey elipsoit biçimli alınırsa, yüzey denklemi

$$a \{ [x - c]^T \Sigma [x - c] - 1 \} = 0 \quad (2.15)$$

ile verilir. Burada; $c \in \mathfrak{R}^n$ elipsoidin merkezini ve $\Sigma \in \mathfrak{R}^{n \times n}$ elipsoidi tanımlayan pozitif tanımlı kovaryans matrisini göstermektedir. İki boyutlu giriş için elips biçimli ayrıştırıcı denklem

$$a \left\{ \begin{bmatrix} x_1 - c_1 & x_2 - c_2 \end{bmatrix} \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \begin{bmatrix} x_1 - c_1 \\ x_2 - c_2 \end{bmatrix} - 1 \right\} = 0 \quad (2.16)$$

olarak yazılabilir. Bu yüzey, DVM'nin ayrıştırıcı yüzeyi olarak kullanılırsa ve b eşiği eklenirse

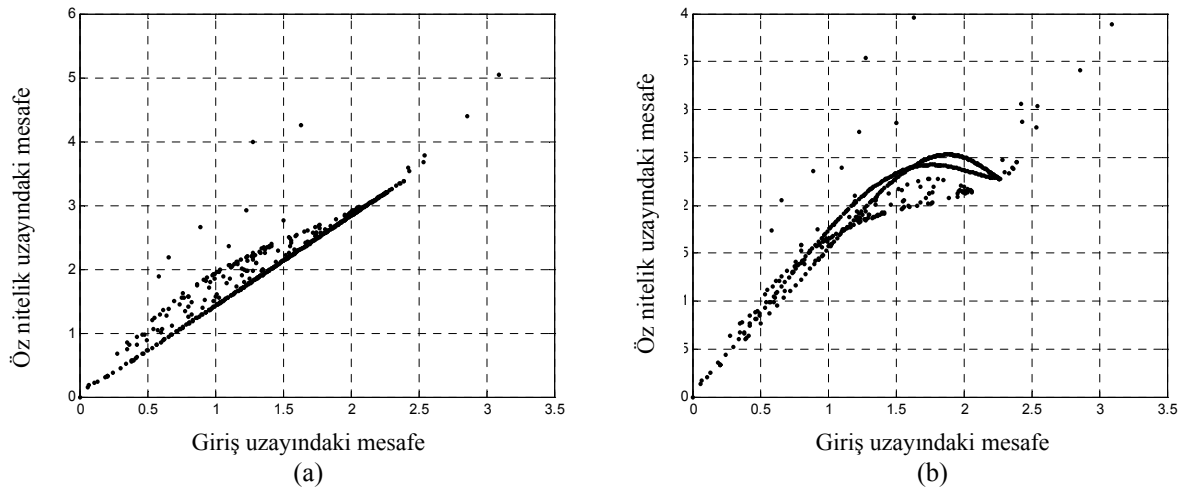
$$w^T \varphi(x^i) + b = [w_1 \quad w_2 \quad w_3 \quad w_4 \quad w_5 \quad w_6] \begin{bmatrix} x_1^2 \\ x_1 x_2 \\ x_2^2 \\ x_1 \\ x_2 \\ 1 \end{bmatrix} + b = 0 \quad (2.17)$$

elde edilir. Küre biçimli ayrıştırıcıya benzer olarak DVM'nin ikinci dereceden karesel programlama problemi çözülürse, w parametreleri elde edilir. Bu değerler kullanılarak ayrıştırıcı yüzeyin parametreleri

$$c_1 = \frac{\left(w_5 - \frac{2w_3 w_4}{w_2} \right)}{-w_2 + \frac{4w_3 w_1}{w_2}}, \quad c_2 = \frac{w_4 + 2c_1 w_1}{w_2}, \quad (2.18)$$

$$\Sigma_{11} = \frac{w_1}{a}, \quad \Sigma_{22} = \frac{w_2}{a} \text{ ve } \Sigma_{21} = \Sigma_{12} = \frac{2w_2}{a} \quad (2.19)$$

olarak hesaplanır.



Şekil 2.2 (a) Daire biçimli çekirdek ve (b) elips biçimli çekirdek için giriş uzayındaki mesafelere karşı öz nitelik uzayındaki mesafe.

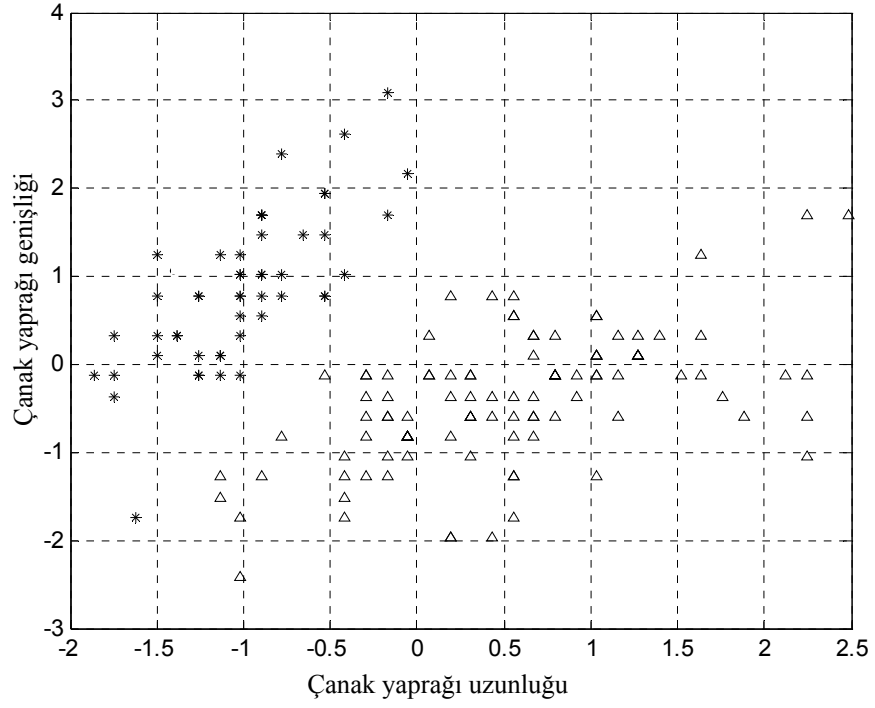
Önerilen küre ve elipsoit biçimli çekirdekler kullanılarak oluşturulan DVM'ler, genel DVM'lerden hesapsal olarak daha kolaydır. Şekil 2.2'de, daire ve elips biçimli çekirdekler için giriş uzayındaki bir noktaya diğer noktaların mesafesi ile aynı noktanın, öz nitelik uzayında diğer noktalara mesafesi arasındaki ilişki verilmiştir. Şekil 2.2a'dan görüldüğü gibi daire biçimli çekirdek durumunda, giriş ve öz nitelik uzayları arasındaki mesafe ilişkisi tam olarak doğrusal

değildir. Ancak polinomsal çekirdek kadar doğrusal olmayan bir ilişki yoktur. Şekil 2.2b’de ise elips biçimli çekirdeğin, tamamen doğrusal olmayan bir işlev verdiği görülür. Bundan dolayı kullanılması uygun değildir ve seçenek sınıflayıcılar oluşturulması gerekir.

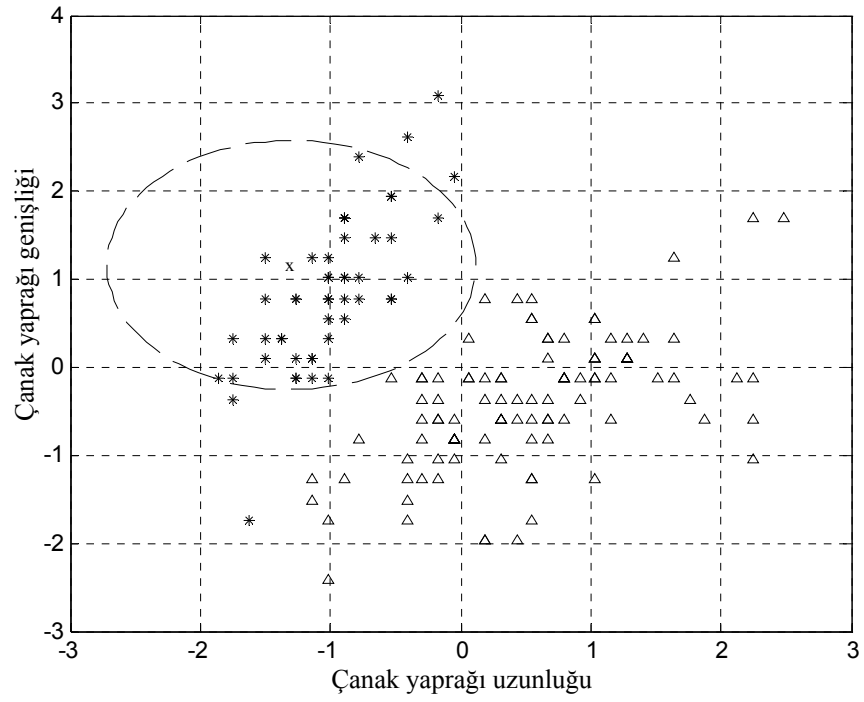
2.2. Önerilen Küre ve Elipsoit Biçimli Sınıflayıcıların Tanımı

Bilinen en basit ve az hata ile ayırıştırma yapan sınıflayıcılar genelde elipsoitlerdir. Bölüm 2.4’te detayları verilecek olan Fisher’in Zambak çiçeği test kümesinin iki sınıfı, Şekil 2.3’de görülmektedir. Bu sınıflar daire biçimli yüzey kullanılarak ayırıştırılmak istenirse, sınıfın tüm verileri daire olarak ayırıştırılamayacağı için Şekil 2.4’deki gibi bir çok veri karar yüzeyinin dışında kalır. Ancak karar yüzeyi elips biçimli olursa, Şekil 2.5’deki gibi veriler tam olarak ayrılabilir. Dolayısıyla bu tezde, giriş uzayında sınıfların ayırıştırıcı yüzeyleri önce küre biçimli olarak düşünülmüş, sonra daha iyi başarımlar elde etmek için sınıflayıcı elipsoit biçimli gösterime genişletilmiştir. Böylece sınıfları ayırştırmak için, sadece küre ve elipsoit biçimli yüzeyleri kullanmak amaçlanmıştır [100]. Yüzeylerin içine ait noktalar bir sınıf, dışındakiler ise diğer bir sınıf alınarak, küre ve elipsoit biçimli sınıflayıcılar oluşturulmuştur.

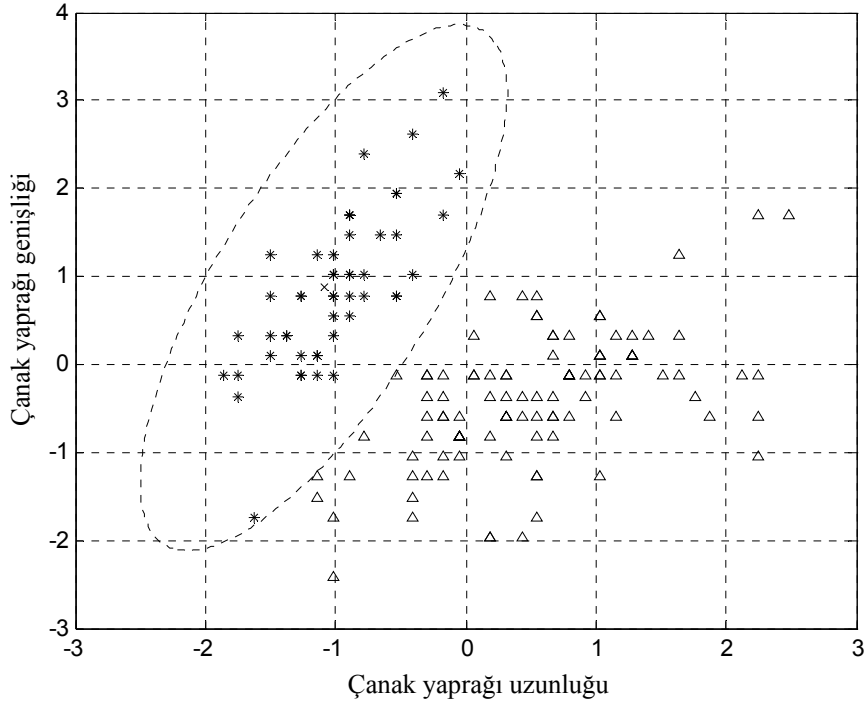
Problem iki aşamalı olarak düşünülmüştür. Önce karar yüzeyi küre alınmış eniyileme problemi önerilmiştir. Önerilen formülasyon eğer problem küre olarak ayırıştırılabiliyorsa, formülasyon %100 doğrulukla çözülecek şekilde oluşturulmuştur. Aksi durumda ise verilerin hatalı sınıflamasına da olanak vererek önerilen eniyileme probleminden beklenen en iyi sonucu vermesi sağlanmıştır. Bu durumda elde edilen doğruluk azalacaktır. Bu nedenle ikinci aşamada, verilerin hatalı sınıflamasına izin vermeden elipsoitler kullanarak ayırıştırılabilme olasılığı düşünülerek, ilk aşamaya benzer şekilde karar yüzeyi elipsoitler olan sınıflayıcılar tasarlanmıştır. Burada ilk aşamada elde edilen kürenin merkez ve yarıçapları, elipsoidin merkez ve kovaryans matrisinin başlatılması için kullanılmıştır. Böylece merkez ve özellikle elipsoidin kovaryans matrisinin başlatılmasında ve güncellenmesinde ortaya çıkan problemlerden kaçılmıştır. Elipsoidin kovaryans matrisinin pozitif tanımlılığı her adımda kontrol edilmiştir. Kötü şartlanma sayısına bakılarak katsayılar düzeltme önerilmiştir [4, 40]. Ancak örneklerde, birinci aşamadan elde edilen merkez veriler kullanıldığı için, bu düzeltme adımına gerek duyulmadığı görülmüştür.



Şekil 2.3 Fisher’in zambak çiçeği test kümesinin sadece ilk iki karakteristiğine ait verilerin gösterimi. İlk sınıfa ait veriler yıldızlar ile diğer sınıfa ait veriler üçgenler ile gösterilmiştir.



Şekil 2.4 Daire biçimli sınıflayıcı.



Şekil 2.5 Elips biçimli sınıflayıcı.

2.2.1. Matematiksel Temel

Bu tezde önerilen, küre ve elipsoit biçimli sınıflayıcılar gerçekte kısıtlı eniyileme problemidir. Bu nedenle, penaltı işlevi yöntemi kullanarak, kısıtlı eniyileme probleminin kısıtsız eniyileme problemi haline dönüştürülmesi gerekir. Bu bölümde, gerekli matematiksel temel sunulmuştur.

Genel kısıtlı eniyileme problemi (2.20)'deki gibi ifade edilir [6, 11, 31]

$$\min \quad f(x) \quad (2.20)$$

$$\text{kısıt} \quad g_j(x) \leq 0, \quad j = 1, \dots, m$$

$$h_l(x) = 0, \quad l = 1, \dots, k$$

$$a(i) \leq x_i \leq b(i), \quad 1 \leq i \leq n \quad (2.21)$$

Burada; $x = (x_1, \dots, x_n) \in R^n$ ve $f(x)$ bir amaç ölçütü, $g_j(x)$ ve $h_l(x)$ sırasıyla eşitsizlik ve eşitlik kısıtlarıdır. $a(i)$ ve $b(i)$ ise x_i için arama uzayı alt ve üst sınırlarıdır.

Penaltı yaklaşımı kullanarak, kısıtlı problem basitçe aşağıdaki gibi kısıtsız probleme dönüştürülebilir.

$$F(x) := \begin{cases} f(x), & g_j(x) \leq 0, \quad h_l(x) = 0 \\ f(x) + AG(x) \end{cases} \quad (2.22)$$

$$G(x) := \sum_{j=1}^m g_j(x) + \sum_{l=1}^k |h_l(x)| \quad (2.23)$$

Burada, $A \in R$ genel bir penaltı parametresini göstermektedir. Bu dönüşüm sonucunda, tüm amaç ölçütü $F(x)$ olmuştur. (2.24)'deki kısıtsız problem çözülerek $F(x)$ 'i enazlayan $x \in R^n$ bulunur.

$$\begin{aligned} \min \quad & F(x) \\ & A > 0 \end{aligned} \quad (2.24)$$

Bu yaklaşım ile ortaya çıkan iki soru vardır. Birincisi kısıtsız eniyileme probleminin, asıl kısıtlı probleme ne kadar iyi yaklaşacağı, ikincisi de önerilen kısıtsız problemin nasıl çözüleceğidir [103]. Öncelikle ilk sorunun cevabı üzerinde düşünülürse, penaltı parametresinin bu yaklaşımın doğruluğunu belirleyebildiği gösterilebilir. A parametresi sonsuza doğru artarken, bu yaklaşım artan bir şekilde kararlı olur. Yani kısıtlı problemin çözümü kısıtsız problemin çözümüne yaklaşır.

Penaltı parametresinin nasıl seçileceği düşünülürse, $F(x)$ 'in ve $G(x)$ 'in değeri büyük ise A 'nın küçük olması gerektiği görülür. Küçük değerdeki A düzgün bir arama ile sonuçlanarak, $F(x)$ 'in değerini aşağıya çekecektir. Ayrıca eğer iki terimin değeri de küçükse, A 'nın arama yönü içinde tümel en uygunu sağlamak için yeterince büyük olması gerekir. Bununla birlikte çoğu durumlarda, değeri önemli olan penaltı parametresinin seçimi probleme bağlıdır ve penaltı işlevinin seçimi için genel bir kural da yoktur [109]. Çapraz geçerlilik testi, L-eğrisi testi veya uygun bir değer bulununcaya kadar “deneme ve yanılma” yöntemi kullanılarak bu parametre seçilebilir [16].

Çapraz geçerlilik testi, düzenlileştirme parametresinin seçimi için uygun bir yöntemdir. Çapraz ilişkinin dayanak noktası, verinin geçerlilik ve eğitim kümesi içine bölünmesidir. Çalışılan algoritmada, bilinmeyen parametrenin sabitlenmiş değeri veri kümesine uygulanır. Daha sonra geçerlilik kümesi, çapraz geçerlilik hatasını hesaplayarak parametre seçiminin uygunluğunu test etmek için kullanılır. İşlem, parametrenin farklı değerleri için tekrarlanır ve en düşük çapraz geçerlilik hatasını veren parametrenin en uygun olduğu düşünülür. Bu mantıkta, çapraz geçerlilik kümesini yorumlayan veya en uygunu hesaplamak için eğitim kümesine izin veren parametreyi seçer. Eğer veri sadece iki alt kümeye bölünürse “hold-out testi” olarak, $\bar{a}$ eşit ölçülü alt kümelere bölünür ve her bir kat için aynı işlemler yapılırsa “ $\bar{a}$ katlı çapraz geçerlilik testi” olarak, $\bar{a}$ eğitim veri sayısına eşit ise bu yöntem “leave-one-out testi” olarak adlandırılır [77, 78].

Penaltı parametresi, veriye uzunluğu ve bir önceki bilgi arasındaki ödünleşimi kontrol eder. Eğer çok küçük seçilirse düşük cezalandırmaya, çok büyük seçilirse aşırı cezalandırmaya götürür. $F(x)$ ve $AG(x)$ 'in logaritmik olarak çizimi, L-eğrisi olarak adlandırılır. Eğer aşırı düzenleme (cezalandırma) varsa eğri gerçekte yataydır, tersi durumunda ise eğri dikeydir. Böyle bir çizim "L" şekilli bir karakteristiğe sahiptir. Düşük ve aşırı cezalandırma arasındaki geçiş, L-eğrisinin köşesidir ve bu köşedeki penaltı parametresinin değeri en iyi olarak düşünülür. Bu iki yöntem, tekrar eden birçok işlem gerektirdiği için fazla hesapsal yük getiren bir iş olarak düşünülür [43, 44].

Önerilen kısıtsız problemin çözümü için, $F(x)$ [23]'deki gibi enerji işlevi olarak düşünülüp, eğim iniş yöntemlerinden biri kullanılarak eniyileme probleminin çözümü yapılabilir [23]. Enerji işlevinin enazlanma problemi, eğim iniş yöntemleri uygulanarak adi diferansiyel denklemlerle sisteme dönüştürülür.

$$\frac{dx}{dt} = -\eta \nabla_x F(x, A), \quad x(0) = x^{(0)}, \quad (2.25)$$

$$A > 0 \quad (2.26)$$

$x(t) \in \mathbb{R}^n$ ve $\eta(x, t)$ uygun olarak seçilen $n \times n$ simetrik pozitif tanımlı integrasyon adıdır. (2.25)'deki diferansiyel denklem sistemi, temel gradyan sistem olarak adlandırılır ve bu yöntem sürekli zamanlı en dik iniş yöntemidir. $-\nabla F(x)$ yönü, herhangi bir noktada $F(x)$ enerji işlevinin en büyük negatif değişiminin yönüdür. Elde edilen bu sistemin ayrık zamanlı eşdeğeri (2.27)'deki fark denklemler kümesi ile gösterilir.

$$x(k+1) = x(k) - \eta(k) \nabla_x F(x(k), A) \quad (2.27)$$

$$\eta(k) > 0 \quad k=0, 1, 2, \dots$$

Burada; $\eta(k)$ öğrenme oranı genellikle sabit alınır. Sıfırdan büyük ancak yakınsamayı sağlamak için sınırlı bir bölge içerisinde seçilmesi gerekir. Her eniyileme problemi için öğrenme oranını doğru seçmek zordur. Yakınsama oranını artırmak için önerilen bazı yöntemler aşağıda anlatılmıştır [23].

2.2.1.1. Uyarlanır Öğrenme Oranlı Eğim İniş Yöntemi

Öğrenme oranı küçük bir değer ile başlatıldıktan sonra, (2.28)'deki gibi özyineli denklem ile değiştirilebilir.

$$\eta(k) = \begin{cases} a\eta(k) & F(x(k)) < F(x(k-1)) \text{ ise,} \\ b\eta(k) & F(x(k)) \geq F(x(k-1)) \text{ ise,} \\ \eta(k) & \text{diğer.} \end{cases} \quad (2.28)$$

Burada; parametrelerin tipik deęerleri $a=1.05$, $b=0.7$ ve $E=1.04$ olarak seilmiřtir [23].

2.2.1.2. Momentum Terimi İeren Uyarlanıř Öğrenme Oranlı Eğim İniř Yöntemi

Uyarlanıř öğrenme oranı ve momentum terimi kullanılarak, yakınsama hızlandırılabilir. Momentum ve uyarlanıř öğrenme oranı ile eğim iniř algoritmasının kullanımı, bir parabol ile hata yüzeyi üzerinde enaza yaklařmaktır [23, 83]. Parametre deęiřimleri için (2.29)'daki kural kullanılır:

$$\Delta x_{ij}(k) = -\eta \left(\frac{\partial F(k)}{\partial x_{ij}(k)} + \gamma x_{ij}(k) \right) + \alpha_{ij}(k) \Delta x_{ij}(k-1) \quad (2.29)$$

Parametrenin deęiřimi üç faktöre baęlıdır. Bunlar; parametreye göre hata iřlevinin türevi, parametrenin gerek deęeri ve parametrenin bir önceki deęiřimidir.

γ katsayısı (tipik olarak $\gamma \approx 10^{-4}$) parametrenin aşırı büyümesini engeller. [83]'de iki farklı momentumlu eğim iniř yöntemi sunulmuřtur. Bunlar arasındaki farklılık, η öğrenme oranı ve α momentum teriminin seiminden dolaydır. Birinde momentum terimi sabit alınır (tipik olarak $\alpha \approx 0.9$) ve öğrenme oranı (2.28)'deki gibi ayarlanır. Dięerinde ise öğrenme oranı, (2.30)'daki gibi iki deęerden birini alır,

$$\begin{cases} \eta = \eta_0 & \left(\frac{\partial F(k)}{\partial x_{ij}(k)} + \gamma x_{ij}(k) \right) > 0 \quad \text{veya} \quad \Delta x_{ij}(k-1) \text{ için} \\ \eta = 0 & \text{dięer.} \end{cases} \quad (2.30)$$

Momentum terimi α her bir adımda her parametre için ayarlanır,

$$\begin{cases} \alpha_{ij}(k) = \alpha_{\max} & \beta_{ij}(k) > \alpha_{\max} \quad \text{veya} \quad S_{ij}(k) \Delta_{ij}(k-1) \beta_{ij}(k) < 0 \\ \alpha_{ij}(k) = \beta_{ij}(k) & \text{dięer.} \end{cases} \quad (2.31)$$

Burada;

$$S_{ij}(k) = \frac{\partial F(k)}{\partial x_{ij}(k)} + \gamma x_{ij}(k) \quad \text{ve} \quad \beta_{ij}(k) = \frac{S_{ij}(k)}{S_{ij}(k-1) - S_{ij}(k)} \quad (2.32)$$

olarak hesaplanır.

Momentum ve uyarlanıř öğrenme oranı ile eğim iniř algoritması, standart eğim iniřten daha hızlıdır ve yerel enaza takılma eğilimi daha azdır.

2.2.1.3. Ölçeklenmiş Eşlenik Eğim İniş Yöntemi

Burada, başlangıç eğimi g_0 ve başlangıç vektörü $p_0 = -g_0$ ile özyineli olarak iki vektör oluşturulur,

$$g_{i+1} = \nabla F(x_{i+1}) \text{ ve } p_k = -g_k + \beta_k p_{k-1}. \quad (2.33)$$

β_k , Polak-Ribiere formülü kullanılarak

$$\beta_k = \frac{|g_k|^2 - g_k^T g_{k-1}}{|g_{k-1}|^2} \quad (2.34)$$

elde edilebilir. Burada; g eğim yönü ve p eşlenik yönü olarak adlandırılır. F 'nin w_{i+1} 'deki enaz noktasına ulaşmak için, doğru enazlama yoluyla w_i 'den p_i yönü boyunca ilerlenir ve enaz noktada g_{i+1} oluşturulur.

Eşlenik eğimin temeli, bulunulan adımı mümkün olduğunca küçük adımlarla bir önceki adıma dikgen yönde giderek, bulunulan adımda önceki adımın sonuçlarını çürütmektir. Eşlenik yönü, salınımı enazlar ve daha uzun olmayan adımlara izin verir. Böylece hata işlevi, en dik iniş yönünde daha hızlı azalmasına rağmen, en dik iniş yönünden daha hızlı yakınsama sağlar.

Ölçeklenmiş eşlenik eğim algoritmaları, eşlenik yönleri boyunca zaman tüketici yön aramadan kaçınan eşlenik eğim algoritmasının bir çeşididir. Bu algoritma [45], geniş parametrelili sistemler için iyi bilinen algoritmalar arasında en hızlı olarak düşünülür. Levenberg-Marquart algoritmasındaki gibi $H = \nabla^2 F$ Hessian matrisini ölçeklemek için skaler ζ içerilir.

$$\Delta x_k = \varepsilon_k p_k \quad (2.35)$$

$$\text{adım ölçüsü } \varepsilon_k = \frac{p_k^T g_k}{p_k^T s_k + \zeta_k |p_k|^2} \quad (2.36)$$

$$s_k = \frac{g(x_k + \alpha_k p_k) - g_k}{\alpha_k} \quad (2.37)$$

$$\Delta_k = \frac{F(x_k) - F(x_k + \varepsilon_k p_k)}{F(x_k) - F(x_k + \varepsilon_k p_k)} \quad (2.38)$$

Burada; F bir gerçel sayıyı, F_q hatanın karesel yaklaşımını, ζ_k her bir adımda F/F_q 'nun 1'e ne kadar yaklaştığına bağlı olarak azalıp-artan skaler bir faktörü göstermektedir.

2.2.1.4. Yarı-Newton Yöntemi

Newton'un yöntemi, hızlı eniyileme için eşlenik eğim yöntemlerine seçenektir [31, 45,

73]. Eğim iniş algoritması, (2.39)'daki güncelleme kuralını kullanır:

$$x_{i+1} = x_i - \lambda_o \nabla F(x_i) \quad (2.39)$$

x_i civarında Taylor serisi kullanılarak $F(x)$ 'in eğimi açılırsa

$$\nabla F(x_{i+1}) = \nabla F(x_i) + (x_{i+1} - x_i)^T \nabla F(x_i) + \text{diğer terimler} \quad (2.40)$$

ve

$$\nabla F(x_i) = 0 \quad (2.41)$$

için denklem çözölür ve yüksek dereceli terimler ihmal edilirse, Newton'un güncelleme kuralı

$$x_{i+1} = x_i - H^{-1}(x_i) \nabla F(x_i) \quad (2.42)$$

olarak elde edilir. Burada; $H(x_i) = \nabla^2 F(x_i)$ parametrelerin bulunulan değerlerinde başarıım indeksinin Hessian matrisidir.

$$\nabla F = \begin{bmatrix} \frac{\partial e_{11}}{\partial x_1} & \frac{\partial e_{11}}{\partial x_2} & \dots & \frac{\partial e_{11}}{\partial x_n} \\ \frac{\partial e_{21}}{\partial x_1} & \frac{\partial e_{21}}{\partial x_2} & \dots & \frac{\partial e_{21}}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial e_{1P}}{\partial x_1} & \frac{\partial e_{1P}}{\partial x_2} & \dots & \frac{\partial e_{1P}}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial e_{NP}}{\partial x_1} & \frac{\partial e_{NP}}{\partial x_2} & \dots & \frac{\partial e_{NP}}{\partial x_n} \end{bmatrix} \quad H = \begin{bmatrix} \frac{\partial F}{\partial x_1^2} & \frac{\partial F}{\partial x_2 \partial x_1} & \dots & \frac{\partial F}{\partial x_n \partial x_1} \\ \frac{\partial F}{\partial x_1 \partial x_2} & \frac{\partial F}{\partial x_2^2} & \dots & \frac{\partial F}{\partial x_n \partial x_2} \\ \dots & \dots & \dots & \dots \\ \frac{\partial F}{\partial x_1 \partial x_n} & \frac{\partial F}{\partial x_2 \partial x_n} & \dots & \frac{\partial F}{\partial x_n^2} \end{bmatrix} \quad (2.43)$$

Hessian matrisini hesaplamak, birçok işlem gerektirdiği için zor ve karmaşıktır. Bu nedenle yarı-Newton olarak adlandırılan algoritmanın farklı bir çeşidi kullanılır. Bu algoritma, her bir k özyinelemede yaklaşık Hessian matrisini günceller. Hessian matrisini yaklaşık olarak hesaplamak için en gözde yöntem, yaklaşık Hessian matrisinin tersini hesaplayan BGFS (Broyden-Goldfarb-Fletcher-Shano) yöntemidir.

$$V_k = V_{k-1} + \left[1 + \frac{r_k^T V_{k-1} r_k}{s_k^T r_k} \right] \frac{s_k s_k^T}{s_k^T r_k} - \frac{s_k r_k^T V_{k-1} + V_{k-1} r_k s_k^T}{s_k^T r_k} \quad (2.44)$$

$$V_k = G_k^{-1}, \quad V_o = 1, \quad s_k = X_k - X_{k-1} \quad r_k = \nabla F_k - \nabla F_{k-1} \quad (2.45)$$

Yarı-Newton algoritması genel olarak daha az adım sayısı ile yakınsamasına rağmen, her bir adımda daha fazla hesaplama gerektirir ve eşlenik iniş yönteminden daha fazla depolama yapar. Yaklaşık Hessian matrisinin boyutu $N_x \times N_x$ 'dir. Fazla parametrelili sistemler için eşlenik iniş algoritmasını kullanmak daha uygundur. Daha az parametrelili ağlar için, etkili bir eğitim yöntemi olabilir.

Yukarıda verilen tüm yöntemler için bir durdurma kriteri gereklidir. Bu çalışmada [73]'deki gibi adım sayısı, hedef başarıım (0), gradyen matrisinin normu ($1.0000e-006 >$) ve

eğitim süresi (sonsuz) kriterlerin tümü aynı anda kontrol edilerek algoritma sonlandırılmıştır.

2.3. Önerilen İki Aşamalı Formülasyon

2.3.1. Aşama-I: Küre Biçimli Sınıflayıcılar

$(x^1, y^1), \dots, (x^L, y^L)$, $x \in \mathfrak{R}^n$, $y \in \{-1, 1\}$ eğitim örneklerinin L örneği için, karar yüzeyi olarak kullanılan $R \in \mathfrak{R}$ yarıçaplı, $c \in \mathfrak{R}^n$ merkezli bir küre

$$D = \left\{ x \in \mathfrak{R}^n \mid \left\| (x^i - c) \right\|^2 - R^2 = 0, \quad i = 1, \dots, L \right\} \quad (2.46)$$

gibi tanımlanır. Kürenin içindeki veri etiketlerinin +1, dışındakilerin ise -1 olduğu kabul edilirse

$$\left\| (x^i - c) \right\|^2 - R^2 \geq 0 \quad y^i = -1, \quad (2.47)$$

$$\left\| (x^i - c) \right\|^2 - R^2 \leq 0 \quad y^i = 1, \quad (2.48)$$

olur, problemin sınıflama hatası da

$$f(x) = f \left(y^i \left(\left\| (x^i - c) \right\|^2 - R^2 \right) \right) \quad (2.49)$$

ile verilir. Burada, f işlevi kullanılarak kürenin içinde olması gereken veri, dışında yerleşmişse veya kürenin dışında olması gereken veri, içinde yerleşmişse amaç ölçütüne pozitif bir terim eklenmesi sağlanır [26-28],

$$\begin{aligned} f(\xi) &= \xi \quad \xi > 0 & f'(\xi) &= 1 \quad \xi > 0 \\ f(\xi) &= 0 \quad \xi \leq 0 & f'(\xi) &= 0 \quad \xi \leq 0. \end{aligned} \quad (2.50)$$

Burada oluşturulan eniyileme probleminin, DVM'ye benzer şekilde sadece formülasyona dayanımlılık kazandıran deneysel hatanın (sınıflama hatasının) değil, aynı zamanda yapısal hatanın da enazlanması istendiği için, en küçük hacimli ayırıştırıcı kısıtı da amaç ölçütüne eklenmiştir [7, 88, 101].

Sonuç olarak, sınıflama hatası enazlanarak x verilerini içeren en küçük hacimli ayırıştırıcı küre problemi, (2.51-2.54) bağıntıları göz önüne alınarak oluşturulur,

$$g(R) = R^2, \quad (2.51)$$

$$f(x) = f \left(y^i \left(\left\| (x^i - c) \right\|^2 - R^2 \right) \right), \quad (2.52)$$

$$F(x) = \sum_{i=1}^L f\left(y^i \left(\|x^i - c\|^2 - R^2\right)\right) + AR^2, \quad (2.53)$$

$$\min_{c,R} \sum_{i=1}^L f\left(y^i \left(\|x^i - c\|^2 - R^2\right)\right) + AR^2 \quad (2.54)$$

$A > 0$, bir önceki bölümde belirtildiği gibi dayanımlılık (doğru sınıflama) ve hacim arasında ödünleşim sağlayan penaltı parametresidir. Bu parametre ve önerilen yaklaşım sayesinde, hem gürültü hem de aykırı veriler esnek bir şekilde dışlanarak gürbüzlük sağlanır.

2.3.1.1. Önerilen Küre Biçimli Sınıflayıcılar İçin Çözüm Yöntemi

(2.54)'de önerilen eniyileme problemi, Bölüm 2.2.1'de verilen eğim iniş yöntemlerinden biri kullanılarak çözülebilir. Eğer en dik eğim iniş yöntemi alınırsa amaç ölçütü F 'nin, (2.55) ve (2.56)'daki gibi c ve R değişkenlerine göre negatif gradyenleri alınır.

$$-\frac{\partial F}{\partial c} = 2 \sum_{i=1}^L y^i (x^i - c) f'\left(y^i \left(\|x^i - c\|^2 - R^2\right)\right), \quad (2.55)$$

$$-\frac{\partial F}{\partial R} = 2 \sum_{i=1}^L y^i f'\left(y^i \left(\|x^i - c\|^2 - R^2\right)\right) R - 2AR, \quad (2.56)$$

(2.57) ve (2.58)'deki gibi, özyineli olarak güncellenebilir,

$$c(k+1) = c(k) - \eta \frac{\partial F}{\partial c}, \quad (2.57)$$

$$R(k+1) = R(k) - \eta \frac{\partial F}{\partial R}. \quad (2.58)$$

Burada; η öğrenme oranını göstermektedir ve her iki güncellemede aynı değer kullanılır. c 'nin başlangıç değeri giriş verileri içerisinde rasgele seçilir ve R 'nin başlangıç değeri ise 1 alınır.

Karar işlevi (2.59)'daki yüzeyle belirlenir

$$\ell(x) = \text{sgn}\left(R^2 - \|x - c\|^2\right). \quad (2.59)$$

Burada; c ve R değerleri (2.57) ve (2.58) kullanılarak elde edilen sonuç değerleri göstermektedir. Bu karar işlevi temeline dayanarak, yüzeyin içinde bulunan noktalar pozitif diğerleri de negatif olarak öngörülür.

2.3.2. Aşama-II: Elipsoit Biçimli sınıflayıcılar

Bu aşamada, karar yüzeyi olarak, $\Sigma \in \mathbb{R}^{n \times n}$ kovaryans matrisli ve $c \in \mathbb{R}^n$ merkezli bir

elipsoit kullanılmıştır,

$$E = \{x \in \mathbb{R}^n \mid ((x^i - c)\Sigma(x^i - c) - 1) = 0, \quad i = 1, \dots, L\}. \quad (2.60)$$

Birinci aşamaya benzer olarak, elipsoidin içindeki verilerin etiketlerinin +1, dışındakilerin -1 olduğu kabul edilirse,

$$((x^i - c)\Sigma(x^i - c) - 1) \geq 0 \quad y^i = -1, \quad (2.61)$$

$$((x^i - c)\Sigma(x^i - c) - 1) \leq 0 \quad y^i = 1, \quad (2.62)$$

olur. Sınıflama hatasını enazlayan ve aynı anda giriş örüntülerini içeren en küçük hacimli elipsoidi arayan sınıflayıcı problemi

$$g(\Sigma) = \frac{1}{\sqrt{|\Sigma|}}, \quad f(x) = f(y^i((x^i - c)\Sigma(x^i - c) - 1)), \quad (2.63)$$

$$F(x) = f(y^i((x^i - c)\Sigma(x^i - c) - 1)) + \frac{1}{\sqrt{|\Sigma|}}, \quad (2.64)$$

$$\min_{c, \Sigma} \sum_{i=1}^L f(y^i((x^i - c)\Sigma(x^i - c) - 1)) + A/\sqrt{|\Sigma|} \quad (2.65)$$

olarak oluşturulur.

2.3.2.2. Önerilen Elipsoit Biçimli Sınıflayıcılar İçin Çözüm Yöntemi

(2.65)'deki eniyileme problemi, Bölüm 2.2.1'de verilen eğim iniş yöntemlerinden birisi kullanılarak çözülebilir. Bunun için öncelikle amaç ölçütü F 'nin, c ve Σ değişkenlerine göre negatif gradyenleri alınır.

$$-\frac{\partial F}{\partial \Sigma} = -\sum_{i=1}^L y^i (x^i - c)(x^i - c)^T f' \left(y^i \left((x^i - c)^T \Sigma (x^i - c) - 1 \right) \right) + \frac{A(\Sigma^{-1})^T}{2\sqrt{|\Sigma|}} \quad (2.66)$$

$$-\frac{\partial F}{\partial c} = 2 \sum_{i=1}^L y^i (x^i - c) f' \left(y^i \left((x^i - c)^T \Sigma (x^i - c) - 1 \right) \right) \quad (2.67)$$

Birinci aşamadan elde edilen merkezler, elipsoidin merkezleri olarak alınır ve elde edilen yarıçaplar kullanılarak $1/R^2$ ile kovaryans matrisinin diyagonal elamanları başlatılır. Eğer en dik eğim iniş yöntemi kullanılmak istenirse, kovaryans matrisi ve c vektörü

$$c(k+1) = c(k) - \eta \frac{\partial F}{\partial c}, \quad (2.68)$$

$$\Sigma(k+1) = \Sigma(k) - \eta \frac{\partial F}{\partial \Sigma}, \quad (2.69)$$

gibi özyinelemeli olarak güncellenebilir. Karar yüzeyi

$$\ell(x) = \text{sgn}(1 - (x^i - c)\Sigma(x^i - c)) \quad (2.70)$$

ile belirlenir. Burada; c ve Σ değerleri (2.68) ve (2.69) bağıntıları kullanılarak elde edilen sonuç değerleri göstermektedir. Karar yüzeyinin içinde bulunan noktalar pozitif, diğerleri de negatif olarak öngörülür.

2.3.3. Penaltı Parametresinin Seçimi

Penaltı parametresinin değeri, bu tezde önerilen eniyileme problemlerinin çözümü için oldukça önemlidir. Penaltı parametresinin uygun değerinin elde edilmesi için, Bölüm 2.2’de verilen testler denenebilir ve bu testlerin sonucunda edinilen deneyimler ışığında penaltı parametresi aşağıdaki gibi seçilebilir:

Eniyileme problemi

$$\min I + A.II. \quad (2.71)$$

olarak tekrar sembolleştirilir. Burada; I , toplam hata terimini ve II , hacim terimini göstermektedir.

$A.II/(I/L)$ birden büyük olursa, hacim terimi ortalama sınıflama hatasından daha büyüktür. Eğer birden küçükse sınıflama hatası çok fazladır. $A.II/(I/L)=1$ ise her iki terim de eşit büyüklüktedir. Bu bölümdeki tüm uygulamalarda her iki terime de eşit önem vermek için, $A.II/(I/L)=1$ alınarak,

$$A = (1/L)I/II \quad (2.72)$$

ile öz yinelemeli olarak hesaplanmıştır: A ’nın başlangıç değeri olarak $A=L$ atanır. Her özyinelemede A ’nın $(1/L_1)$ değerinden küçük olup olmadığı belirlenir. Küçük olduğu durumda A ’nın bir önceki değeri alınır. L_1 burada yüzeyin içinde kalması gereken veri sayısıdır. Uygulamalarda $L_1 = L/2$ değeri alındığında iyi sonuç elde edilmiştir.

Bir seçenek olarak, tamamen aynı olmamakla birlikte gürbüz bulanık öbekleme algoritmalarında yapıldığı gibi A büyük bir değerden başlatılarak adım sayısına göre üstel olarak azaltılabilir [85]. Bu amaçla (2.72)

$$A = (1/L)(I/II)\exp(-zh) \quad (2.73)$$

olarak değiştirilir. Burada; z adım sayısı ve h azalma hızını belirleyen bir sabittir. Buna göre A başlangıçta büyük bir değer seçilerek, penaltı terimine daha fazla önem verilmesi sağlanır. Daha

sonra, (2.73)'e göre $(1/L_1)$ değerine kadar azaltılarak diğer terime önem verilmesi sağlanır. A hiçbir zaman sıfıra götürülmez, bunun nedeni sıfır olması durumunda hacim üzerinde herhangi bir kısıt olamayacağıdır. Bu bölümdeki uygulamalarda, her bir adımda (2.72) ile hesaplanan A değeri, yeni bir A değeri alınarak güncellenmiştir.

2.4. Uygulamalar

2.4.1. Veri Kümeleri

İki sınıflı sınıflama ile ilgili örneklerde, yüksek boyutlu ve veri sayısı farklı olan sekiz veri kümesi kullanılmıştır. Kullanılan veri kümelerinin dördü California, Irvine Üniversitesi'nin (UCI) makine öğrenme veritabanları arşivinden alınmıştır [72]. Diğerleri Fisher'in zambak çiçeği veri kümesi [9], [75]'den Galaksi/Yıldız ayrıştırıcı veri kümesi ve yapay olarak oluşturulan veri kümeleridir. Ayrıca çoklu sınıflama ile ilgili örneklerde UCI'den üç veri kümesi kullanılmıştır. Bu veri kümeleri hakkında ayrıntılar aşağıda verilmiştir:

Wisconsin Göğüs Kanseri (WGK) veri kümesi: W. H. Wolberg tarafından Wisconsin Üniversitesinde toplanan, UCI'deki göğüs kanseri verilerinden biridir. Hastanın göğsünden alınan doku örneğinin, kötü huylu veya iyi huylu olup olmadığını belirleme problemleridir. 13 sayısal özellikli iki sınıf vardır. 41 kötü huylu ve 69 iyi huylu toplam 110 gözlem vardır.

BUPA Karaciğer Bozuklukları (BUPAKB) veri kümesi: Bu UCI veri kümesi, R. S. Forsyth tarafından BUPA Tıp Araştırma Şirketinde toplanmıştır. Bir erkek hastanın, kan testleri ve alkol tüketimine göre toplam 6 sayısal özelliğe dayanarak karaciğer bozukluğunun olup olmadığını belirleme problemleridir. Karaciğer bozukluğunun şiddetine göre iki sınıf vardır. Bunlar, 200'ü bir sınıftan ve 145'i diğer sınıftan olan, toplam 345 hastadır.

Cleveland Kalp Hastalıkları (CKH) veri kümesi: Bu UCI veri kümesi, R. Detrano tarafından Amerika Birleşik Devletleri'ndeki Cleveland kalp merkezinden toplanmıştır. Hasta üzerinde yapılan çeşitli tıbbi testlerin sonuçları verildiği zaman, kalp hastalığının varlığını veya yokluğunu belirleme problemleridir. 7'si sayısal, 6'sı kategorik olmak üzere 13 özellikli iki sınıf vardır. 83'ü kalp hastası ve 214'ü de hasta olmayan, toplam 297 gözlem vardır.

Galaksi/Yıldız Ayrıştırıcı (GYA) veri kümesi: Bu veri kümesi, Odewahn'ın [75]'deki çalışmasından alınmıştır. Minnesota Üniversitesi'nde Otomatik Plaka Tarayıcı (OPT) kullanılarak toplanmıştır. OPT tarafından fark edilen nesnelerin 14 görüntü parametresine dayanarak, galaksi mi yoksa yıldız mı olduğunu tespit ederek onları ayrıştırma problemleridir.

Bu veri kümesinde 2110 galaksi ve 2082 yıldız bulunmaktadır.

İyonosfer veri kümesi: İyonosfer radar yansımaları verisidir. Bu radar verisi Goose Bay, Labrador anten sistemi ile toplanmıştır. Sistem, toplam iletim gücü 6.4 KW olan 16 yüksek frekanslı dizi antenden oluşmaktadır. Hedefler iyonosferdeki serbest elektronlardır. “İyi” radar dönüşleri, iyonosferdeki elektronların varlığını, “Kötü” radar dönüşleri ise yokluğunu göstermektedir. Alınan sinyaller; bağımsız değişkenleri, dönüş sinyali sıra numarası sayısı ve dönüş süresi olan bir öz ilişki işlevi kullanılarak işlenmiştir. Goose Bay sisteminde, 17 dönüş sinyali kullanılmıştır. Bu veri tabanındaki her bir örnek, dönüş sayısı başına değerleri karmaşık sayılar ile verilen iki elektromanyetik sinyalle tanımlanmıştır. Bu nedenle iyonosfer veri kümesi 34 girişle verilmektedir. Toplam 351 örnekten oluşmaktadır. Bu örneklerin 225’i “iyi”, 126’sı “kötü” olarak etiketlenmiştir.

Zambak çiçeği veri kümesi: Zambak çiçeği verisi ilk olarak, 1936’da Fisher tarafından yayınlanmıştır [9]. Bu veri kümesi, her biri 50 örnekten oluşan 3 sınıf içerir. Her bir sınıf zambak çiçeğinin farklı bir tipini göstermektedir. Zambak çiçeğinin farklı tiplerini ayırtmak için 4 karakteristiğine bakmak yeterlidir. Bu karakteristikler; santimetre olarak verilen çanak yaprağı uzunluğu, çanak yaprağı genişliği, taç yaprağı uzunluğu ve taç yaprağı genişliğidir. Zambak çiçeğinin bir sınıfı diğerlerinden doğrusal olarak ayrıştırılabilir, diğerleri ise ayrıştırılamaz.

Yapay Veri kümesi-1 (YV1): Bu veri kümesi, önerilen sınıflayıcıların başarımını göstermek için ilk olarak [100]’de oluşturulmuştur. Birim dairenin içindeki ve dışındaki verileri bulma problemidir. Birim daire içindeki veriler bir sınıftan, dışındakiler ise diğer sınıftan olmak üzere, iki özelliği bulunan iki sınıf vardır. 141 pozitif olarak etiketlenen ve 353 negatif olarak etiketlenen, toplam 494 örnek vardır.

Yapay Veri kümesi-2 (YV2): Bu veri kümesi, YV1 veri kümesinden sadece $x=[0.9\ 0; 0.18\ -0.3; 0.36\ 0.3; -0.18\ 0.15; -0.22\ -0.24; 1.1\ 0; 1.11\ 0.96; 1.17\ -0.915; -0.96\ -1.1]$ $y=[1\ 1\ 1\ 1\ 1\ -1\ -1\ -1\ -1]$; verileri alınarak oluşturulmuştur. Öncekine benzer şekilde birim dairenin içindeki ve dışındaki verileri bulma problemidir.

Ayrıca YV2 veri kümesinin küre olarak ayrıştırılmaması için sırasıyla, 1 etiketli {1.25, 0.75} ve {10, 10} verileri eklenerek yeni veri kümeleri oluşturulmuştur. Elde edilen veri kümeleri uygulamalarda sırasıyla **YV21** ve **YV22** olarak isimlendirilmiştir.

Şarap veri kümesi: Bu veri kümesi, C.Blake tarafından Eczacılık ve Yiyecek Analizi ve Teknolojileri Enstitüsü’nde toplanmıştır. İtalya’nın özel bir alanında üretilen şarapların kimyasal analizinin sonuçları kullanılarak şarapların üç farklı tipini belirleme problemidir. Toplan 178 örneğin her biri için 13 kimyasal analiz yapılmıştır.

Cam veri kümesi: Bu veri kümesi, B. German tarafından USA Adli Bilim Servisi'nde suçun işlendiği olay yerinde bulunan camların kanıt olarak kullanılması amacıyla toplam 214 örnek toplanmıştır. Camın oksit içeriğine bakarak (magnezyum, alüminyum, silikon, potasyum, kalsiyum, baryum, demir, sodyum, ışığı kırma indeksi gibi) işlenmiş veya işlenmemiş bina pencere camı, işlenmiş veya işlenmemiş araba camı, şişe camı, masa camı ve far camı olup olmadığını belirleme problemidir.

Uydu görüntüleri veri kümesi: Bu veri kümesi Avusturya Uzaktan Algılama Merkezi tarafından NASA'dan satın alınmıştır. 4 elektromanyetik izgel (spektral) bant ve komşuluktaki 9 piksel değerlendirilerek alanın kırmızı toprak, gri toprak, nemli gri toprak, çok nemli gri toprak, pamuk ürününe veya bitki örtüsüne sahip toprak olup olmadığını belirleme problemidir.

2.4.1.2. İki Sınıflı Sınıflayıcılar İçin Uygulamalar

Aşağıda verilen ilk iki örnekte, bu tezde önerilen küre ve elipsoit biçimli sınıflayıcıların eğitim yöntemlerinin başarımları incelenmiştir. İlk örnekte, zambak çiçeği verisi ve oluşturulan yapay veri kümeleri üzerinde, yöntemlerin eğitim başarımları ve eğitim süreleri önce birbirleriyle ve daha sonrada RTİ çekirdekli ve küre çekirdekli DVM ile kıyaslanmıştır. İkinci örnekte ise zambak çiçeği verisinin farklı boyutları ile üç sınıf için ayrı ayrı sınıflayıcılar oluşturularak, hem eğitim hem de test başarımları DVM ile kıyaslanmıştır. Üçüncü örnekte ise bilimsel yazında bilinen veriler kullanılarak; önerilen sınıflayıcılar ile hem DVM tabanlı hem de diğer sınıflayıcılar, eğitim ve test kümesi üzerindeki başarımları ve eğitim süreleri açısından kıyaslanmıştır.

2.4.2. Önerilen Sınıflayıcıların Farklı Yöntemlerle Eğitilerek Başarımının İncelenmesi

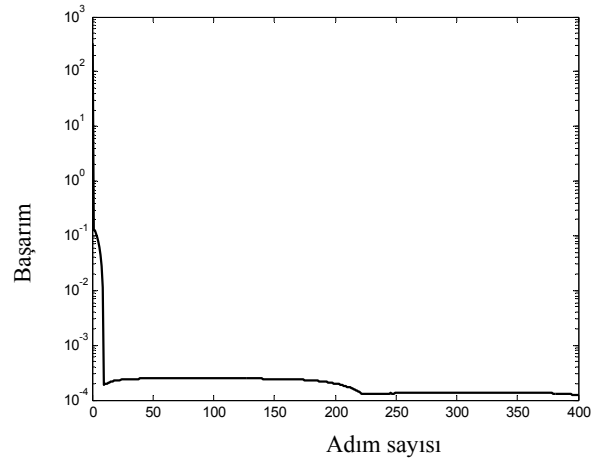
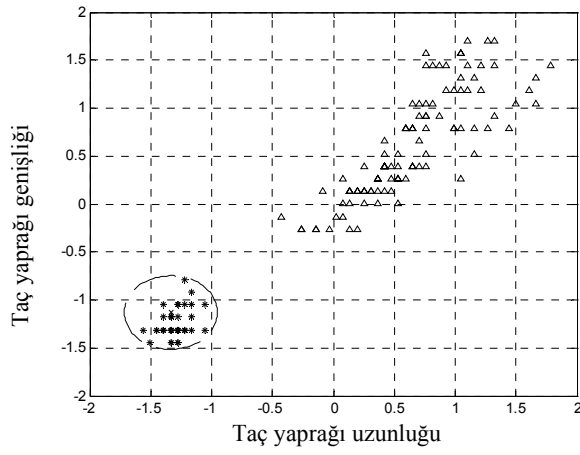
Bu uygulamada, zambak çiçeği verisinin 3. ve 4. karakteristiği alınarak, birinci ve ikinci sınıflarının ayrı ayrı ayrıştırılması problemi düşünülmüştür ve tüm veriler eğitim için kullanılmıştır. Ayrıca oluşturulan yapay veri kümeleri ile ikili sınıflama yapılmıştır. Sınıflayıcıların her biri için ayrılması gereken, yani karar yüzeyinin içinde kalması gereken her bir sınıf 1, diğerleri -1 olarak etiketlenmiştir. Sınıflayıcılar Genel Eğitim İniş (GEİ), Uyarlanı Öğrenme Oranlı Eğitim İniş (UÖEİ), Momentum Terimli Uyarlanı Öğrenme Oranlı Eğitim İniş (MUÖEİ), Eşlenik Eğitim İniş (EEİ), Ölçeklenmiş Eğitim İniş (ÖEİ) ve Yarı Newton (YN) yöntemleriyle eğitilmiştir. Sınıflayıcıların bu eğitim yöntemleri için, eğitim süresi ve doğruluğu açısından başarımlarına bakılmıştır. Tüm eğitim algoritmalarında, c merkez değerleri eğitim

verileri içerisinde her kat için rasgele olarak ve R değerleri 1 alınarak başlatılmıştır.

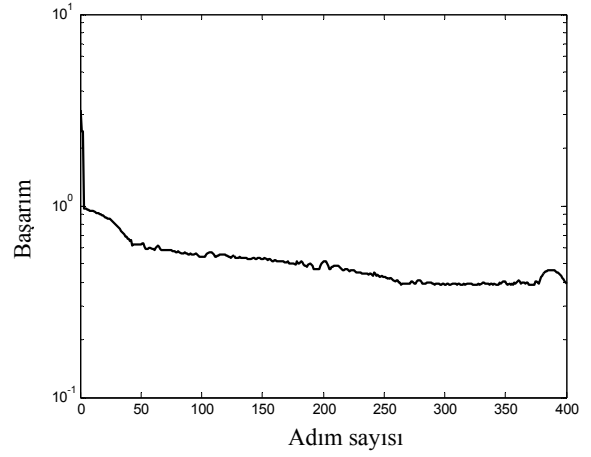
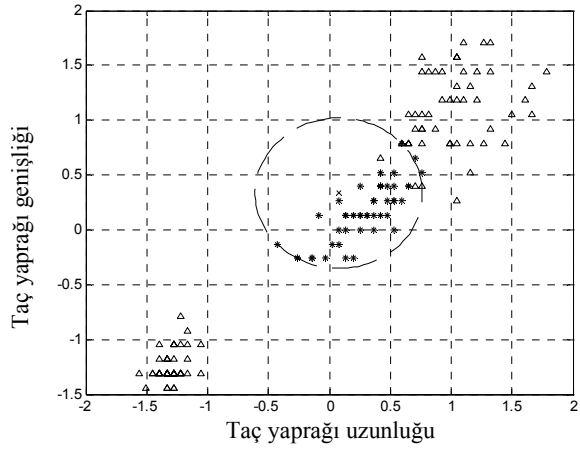
Bu tezdeki tüm hesaplamalar için MATLAB kullanılmıştır [73]. Sonuçlar Intel Centrino 1.6 GHz, 756 MB RAM'e sahip diz üstü bilgisayar kullanılarak elde edilmiştir. Farklı eğitim yöntemlerine ilişkin sonuçlar Tablo 2.1'de verilmiştir. Penaltı parametresi (2.72)'deki gibi $A = (1/L).I/H$ 'e göre hesaplanmıştır. Tüm algoritmalar 400 adım için çalıştırılmış, ancak Bölüm 2.2'de belirtilen sonlandırılma kriterleri sağlandığında özyineleme durdurulmuştur. Bu nedenle Tablo 2.1'de adım sayıları ayrıca verilmiştir. Bu tablodaki bazı sonuçlarda sıfır adım sayısı ile algoritmanın sonlandırıldığı görülmektedir. Bu sonuç algoritmanın sadece verilen başlangıç değerleri için çözüm yaparak durdurulduğunu göstermektedir. Tabloda verilen sonuçlara göre önerilen sınıflayıcılar, MUÖEİ yöntemiyle eğitildiklerinde, eğitim süresi ve özellikle doğruluk açısından yeterince iyi sonuç verdikleri görülebilir. Bu nedenle, sadece bu yöntemin, birinci ve ikinci aşamasına ilişkin eğitim özyineleme sonuçları ve elde edilen karar yüzeyleri Şekil 2.6 ve Şekil 2.7'de verilmiştir.

Tablo 2.1 Farklı eğitim algoritmalarının kıyaslanması.

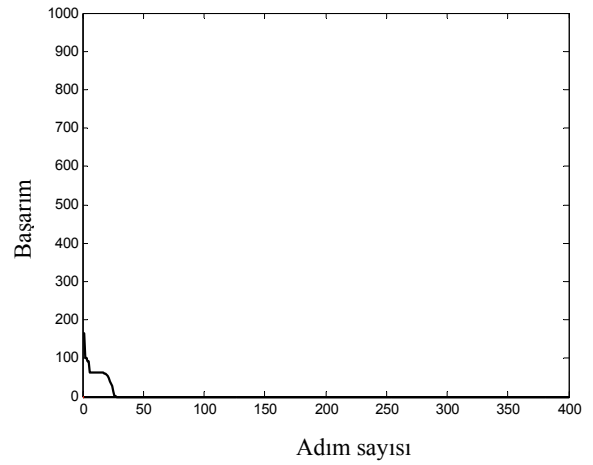
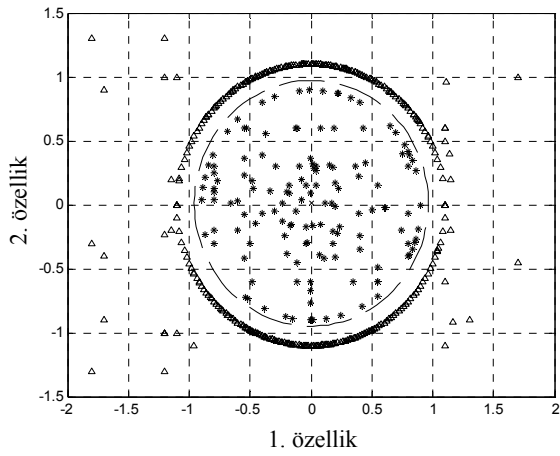
Veri Kümesi	GEİ Öğrenme Oranı Süre [s] Doğruluk [%] Adım Sayısı	UÖEİ Süre [s] Doğruluk [%] Adım Sayısı	MUÖEİ Süre [s] Doğruluk [%] Adım Sayısı	EEİ Süre [s] Doğruluk [%] Adım Sayısı	ÖEİ Süre [s] Doğruluk [%] Adım Sayısı	YN Süre [s] Doğruluk [%] Adım Sayısı
Zambak Çiçeği 2.sınıf Aşama-I	0.01 0.9820 97.333 400	- 0.9810 96.6667 400	0.9820 96.6667 400	0.4710 97.3333 4	0.6610 96.667 100	0.4610 96.6667 7
Zambak Çiçeği 2.sınıf Aşama-II	0.001 0.6610 96.6667 200	1.2345 96.6667 400	1.0800 96.6667 400	0.4330 97.3333 80	0.3910 97.3333 80	0.2310 97.3333 8
Zambak Çiçeği 1.sınıf Aşama-I	0.01 0.9510 100 100	0.96 100 400	1.0110 100 400	0.4810 100 6	0.38 100 2	0.53 100 9
Zambak Çiçeği 1.sınıf Aşama-II	0.001 0.25 100 2	0.1900 100 3	0.1800 100 3	0.5430 100 100	0.4610 100 80	0.01 100 0
YV1 Aşama-I	0.0005 3.9060 100 400	3.8750 100 400	3.9960 100 400	0.4810 100 4	9.4240 100 400	2.4240 100 100
Aşama-II	0.001 0.3100 100 6	0.5674 100 25	0.4810 100 25	2.67 100 100	1.42 100 50	0.01 100 0
YV2 Aşama-I	0.01 0.82 100 30	0.6710 100 400	0.6710 100 400	0.34 100 400	0.4610 100 400	0.3210 100 400
Aşama-II	0.005 0.3010 100 400	0.2600 100 2	0.2600 100 2	0.28 100 50	0.38 100 50	0.01 100 0
YV21 Aşama-I	0.01 0.8420 80 400	0.6510 70 400	0.7820 80 400	0.4110 80 400	0.4800 80 400	0.3200 90 7
Aşama-II	0.005 0.9110 100 400	0.435 100 100	0.3810 100 100	0.48 100 50	0.4710 100 80	0.01 100 0
YV22 Aşama-I	0.01 0.78 50 400	0.5910 70 400	0.8710 70 100	0.4410 60 400	0.4910 60 400	0.4410 60 17
Aşama-II	1e-5 1.4320 80 400	0.9450 70 400	0.8976 80 100	0.4517 60 400	0.5673 60 400	0.16 70 5



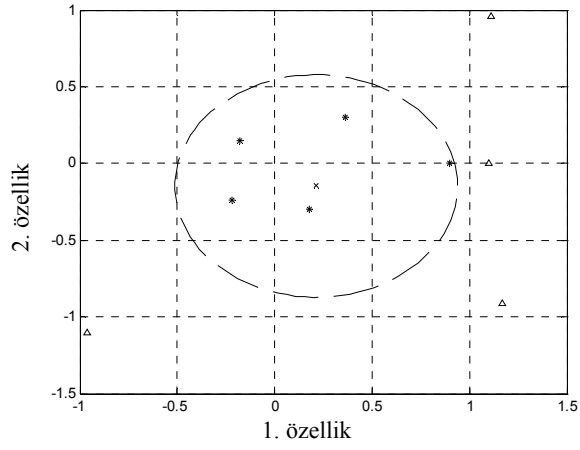
(a) Zambak çiçeği veri kümesinin 3. ve 4. karakteristiği alınarak birinci sınıfın ayrıştırılması



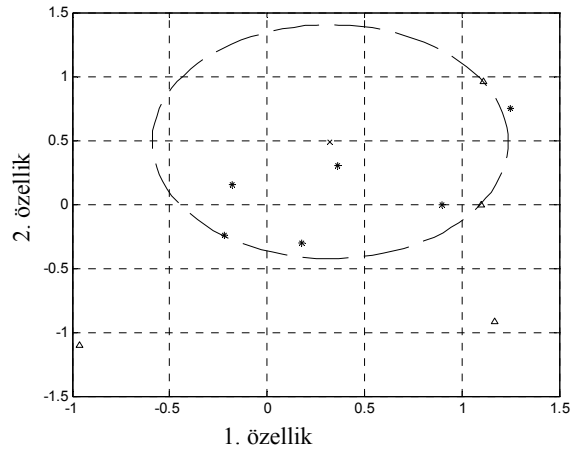
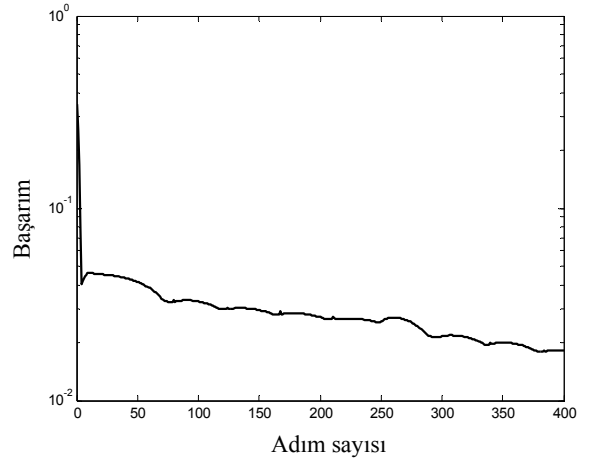
(b) Zambak çiçeği veri kümesinin 3. ve 4. karakteristiği alınarak ikinci sınıfın ayrıştırılması.



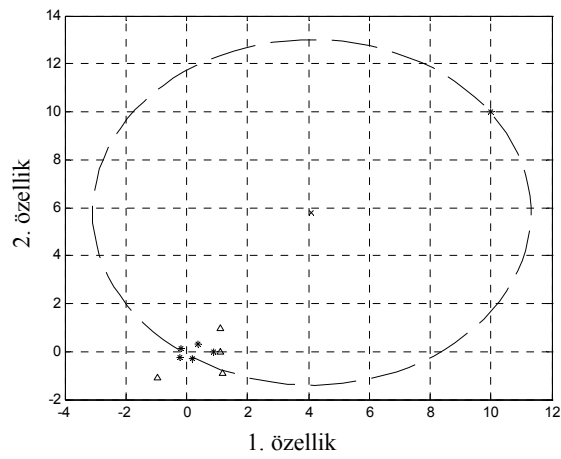
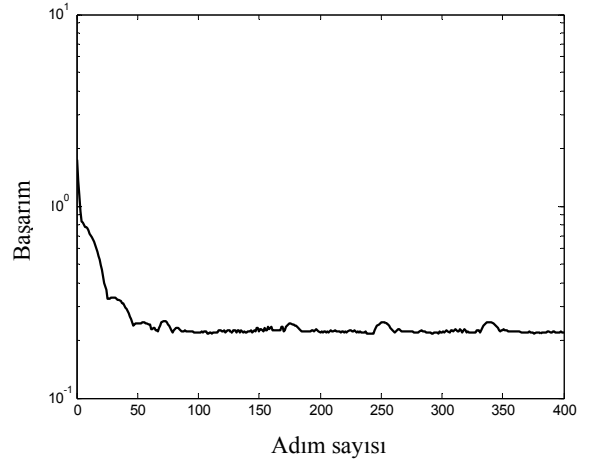
(c) YV1 veri kümesinin ayrıştırılması.



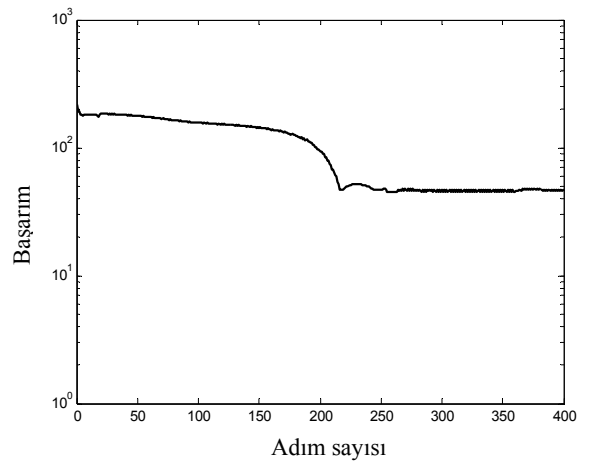
(d) YV2 veri kümesinin ayrıştırılması.



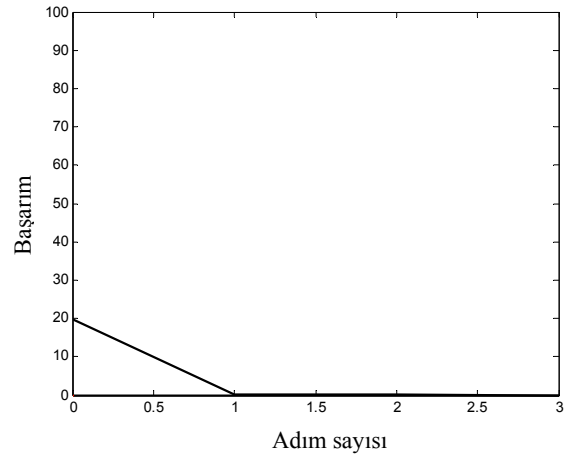
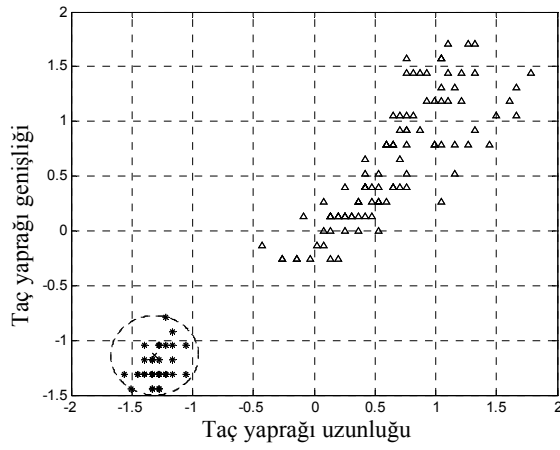
(e) YV21 veri kümesinin ayrıştırılması.



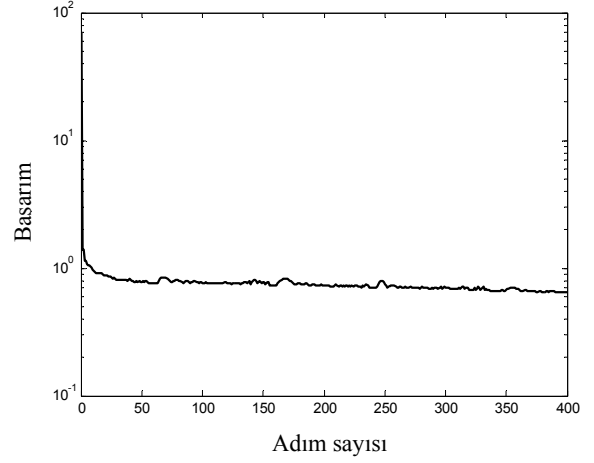
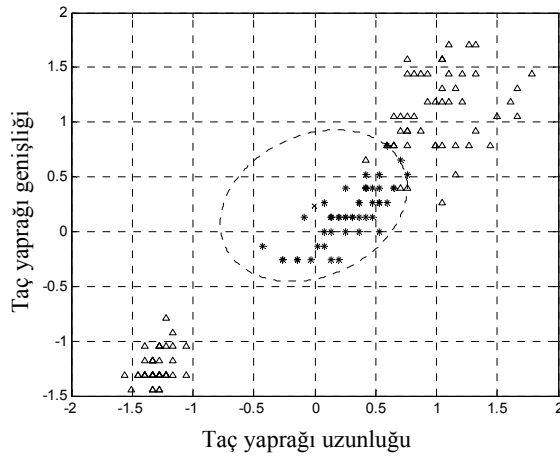
(f) YV22 veri kümesinin ayrıştırılması.



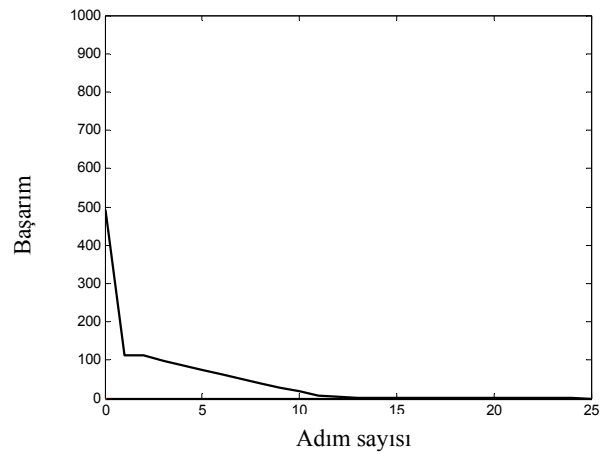
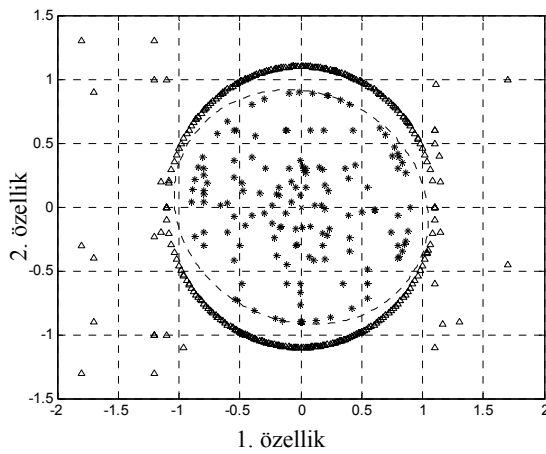
Şekil 2.6 Aşama-I'de önerilen sınıflayıcının MUÖEI algoritması ile eğitim başarımı ve karar yüzeyleri.



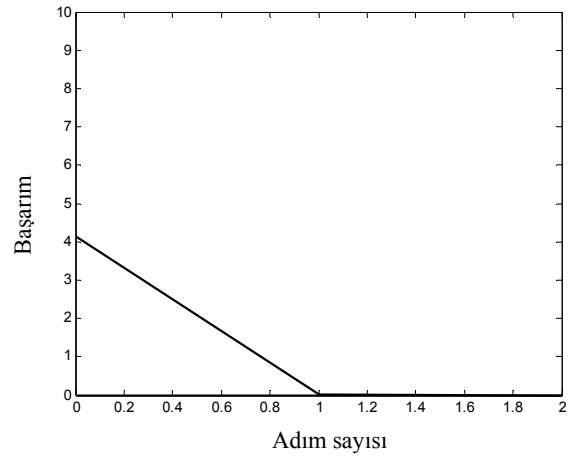
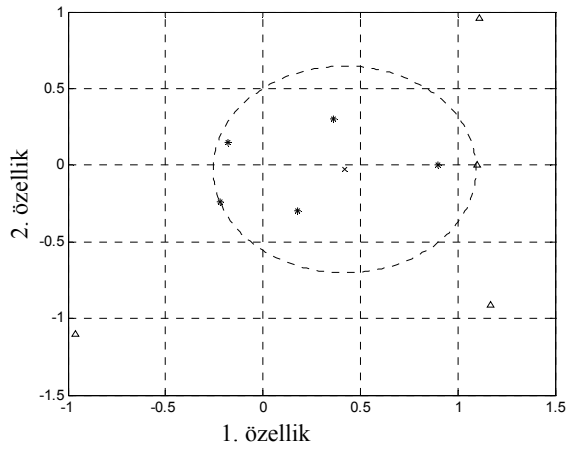
(a) Zambak çiçeği veri kümesinin 3. ve 4. karakteristiği alınarak birinci sınıfın ayrıştırılması.



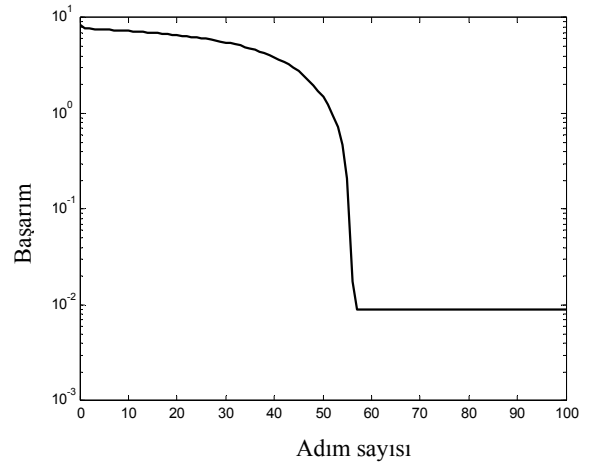
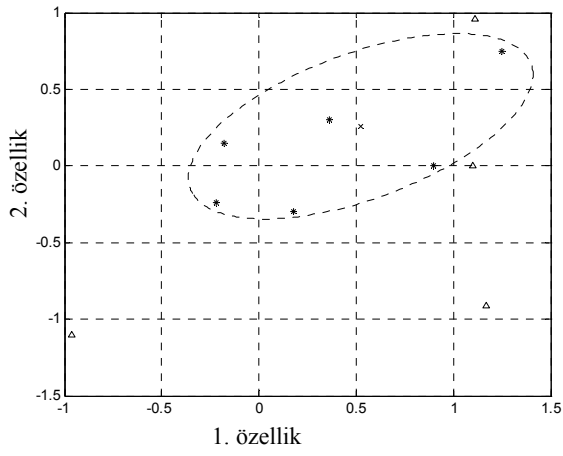
(b) Zambak çiçeği veri kümesinin 3. ve 4. karakteristiği alınarak ikinci sınıfın ayrıştırılması.



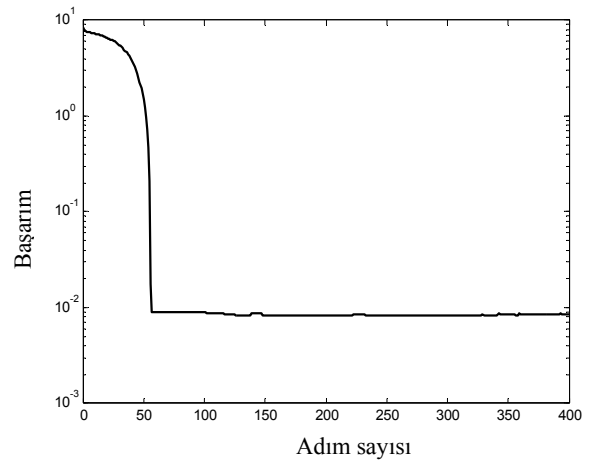
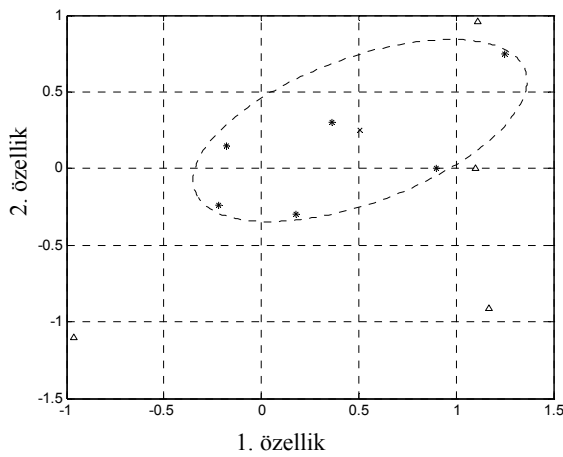
(c) YV1 veri kümesinin ayrıştırılması.



(d) YV2 veri kümesinin ayrıştırılması.



(e) YV21 veri kümesinin ayrıştırılması.



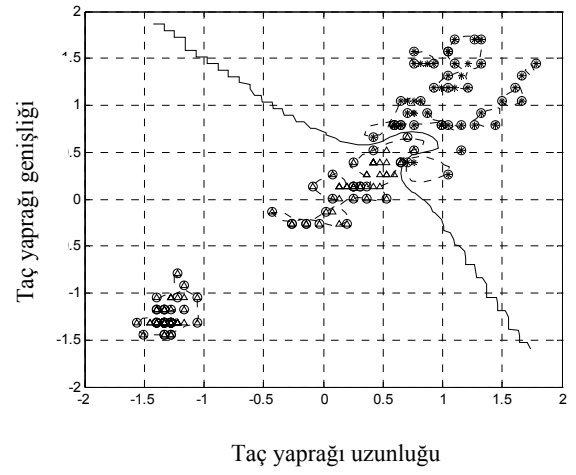
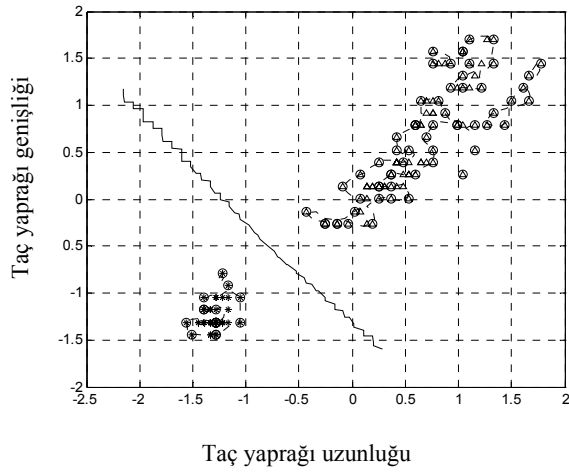
(f) YV21 veri kümesinin ayrıştırılması.

Şekil 2.7 Aşama-II'de önerilen eniyileme probleminin MUÖEI algoritması ile eğitim başarımı ve karar yüzeyleri.

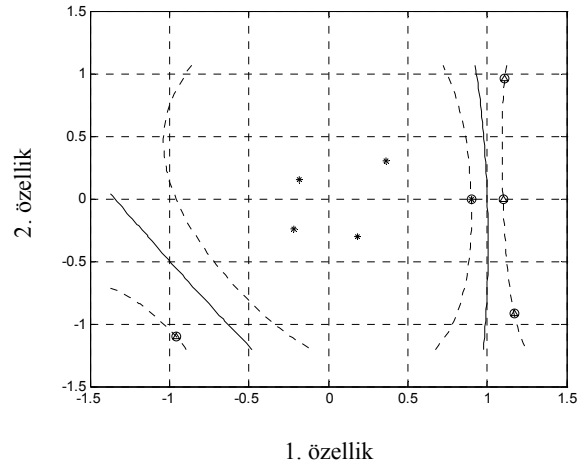
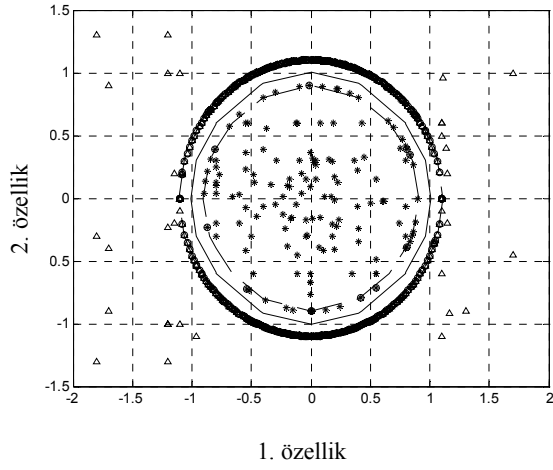
Önerilen sınıflayıcının eğitim başarımı ve karar yüzeyleri elde edildikten sonra, [40]’daki MATLAB kütüphanesi kullanılarak RTİ çekirdekli DVM ve MATLAB’ın “quadprog” komutu kullanılarak karesel programlama ile çözümü yapılan küre biçimli çekirdeğe sahip DVM, eğitim başarımı ve süresi açısından kıyaslanmıştır. DVM’nin C parametresi [88]’de önerildiği gibi $C=10L$ alınmıştır. σ parametresi ise $[0.01:0.001:3]$ dizisi içerisinde en iyi eğitim başarımı ve büyük pay sağlayan değerlerden biri seçilmiştir. Elde edilen sonuçlar, Şekil 2.8 ve Tablo 2.2’de verilmiştir. Sonuçlardan veri sayısı arttıkça DVM’nin önerilen yöntemle göre çok daha yavaş olduğu görülmektedir. Diğer taraftan YV22 veri kümesindeki gibi veriler, elipsoit biçiminde ayrıştırılamadığı durumda DVM’nin eğitim başarımının önerilen sınıflayıcılardan daha iyi olduğu görülmektedir.

Tablo 2.2 Önerilen sınıflayıcıların eğitim başarımının küre biçimli ve RTİ çekirdekli DVM ile kıyaslanması.

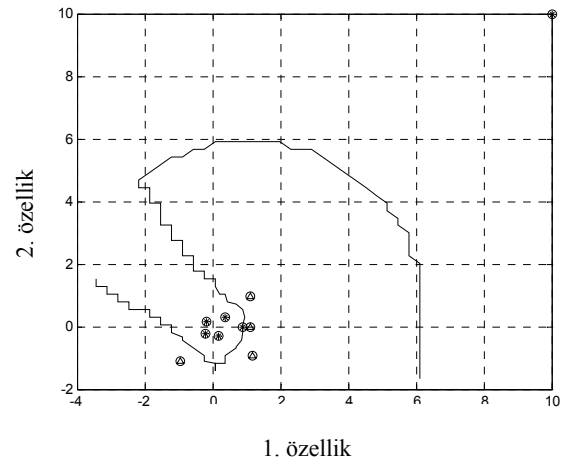
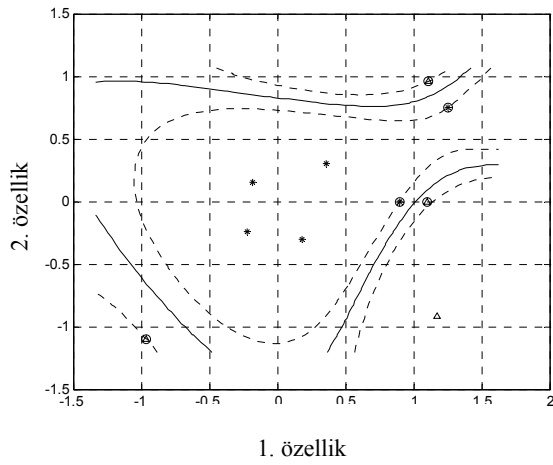
Yöntem	YV1	YV2	YV21	YV22	Zambak Çiçeği 1. sınıf	Zambak Çiçeği 2. sınıf
DVM						
Süre [s]	316.44	0.16	0.14	0.4	2.73	2.29
Doğruluk [%]	100	100	100	100	100	99.33
C	3530	90	90	90	1500	1500
σ	1	0.125	0.125	0.125	0.125	0.125
Küre Çekirdekli DVM						
Süre [s]	104	0.23	0.2	0.2	1.73	1.87
Doğruluk [%]	100	100	100	80	100	96.62
σ	3530	90	90	90	1500	1500
Önerilen Sınıflayıcı (YN)						
Süre [s]	2.42	0.01	0.01	0.16	2.37	0.23
Doğruluk [%]	100	100	100	70	100	97.33



(a) Zambak çiçeği veri kümesinin 3. ve 4. karakteristiği alınarak birinci ve ikinci sınıfın ayrıştırılması.



(b) YV1 ve YV2 veri kümesinin ayrıştırılması.



(c) YV21 ve YV21 veri kümesinin ayrıştırılması.

Şekil 2.8 RTİ çekirdekli DVM sınıflayıcının karar yüzeyleri.

2.4.3. Zambak Çiçeği Veri Kümesi ile İki Sınıflı Sınıflama Problemi

Burada bir önceki örnekten farklı olarak, zambak çiçeği verisi üzerinde önerilen küre ve elipsoit biçimli sınıflayıcıların test başarımı da incelenmiştir. Zambak çiçeği verisi 4 karakteristikli olduğu için, önce 3. ve 4. karakteristikler alınarak iki boyutlu giriş uzayında, daha sonra dört karakteristiği de alınarak dört boyutlu giriş uzayında, üç ayrı küre ve elipsoit biçimli sınıflayıcılar oluşturulmuştur.

Tüm veriler, sıfır ortalamalı standart sapması 1 olarak ölçeklenmiştir. Önce rasgele karıştırılmış, daha sonra 10 eşit alt kümeye bölünerek 4'ü eğitim 6'sı test kümesi olarak kullanılmıştır. Önerilen sınıflayıcılar, 350 adım için MUÖEİ yöntemi ile eğitilmiştir. Algoritma 10 defa çalıştırılarak ortalama eğitim süresi ile hem eğitim hem de test verileri için doğru sınıflanmış noktaların toplam yüzdesi, ortalama doğruluk elde edilmiştir. Elde edilen sonuçlar, karesel programlama ile eğitilen DVM ile kıyaslanmıştır [41]. DVM için RTİ çekirdek kullanılmıştır. Çekirdek parametresi [0.01:0.001:3] dizisi içerisinde en iyi test başarımını veren değerlerden biri olan $\sigma=0.125$ seçilmiştir. DVM'nin C parametresi ise [10:10:3000] dizisi içerisinde en iyi test başarımını veren değerlerden biri olan $C=1000$ alınmıştır. DVM'de 10 defa çalıştırılarak rasgele karıştırılmış veriler için ortalama eğitim ve test başarımları elde edilmiştir. Tablo 2.3'de verilen tüm sonuçlardan görüldüğü gibi, önerilen sınıflayıcılar eğitim süreleri ve test başarımları açısından DVM'ye göre üstündür.

Tablo 2.3 Önerilen sınıflayıcıların ve DVM'nin 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.

Zambak Çiçeği	DVM	Aşama -I-	Aşama -II-	DVM	Aşama -I-	Aşama -II-	DVM	Aşama -I-	Aşama -II-
	1. Sınıf	1. Sınıf	1. Sınıf	2. Sınıf	2. Sınıf	2. Sınıf	3. Sınıf	3. Sınıf	3. Sınıf
n=4									
Eğitim	100	100	100	100	95.00	98.33	100	96.66	100
Doğ. [%]									
Test	100	98.87	100	93.13	88.76	94.38	94.38	93.50	95.13
Doğ. [%]									
Süre [s]	0.13	0.07	0.02	0.15	0.03	0.1	0.15	0.04	0.28
n=2									
Eğitim	100	100	100	98.33	96.62	100	100	96.66	100
Doğ. [%]									
Test	100	100	100	94.38	94.62	95.62	93.25	94.38	95.50
Doğ. [%]									
Süre [s]	0.26	0.04	0.01	0.19	0.03	0.01	0.25	0.04	0.03

2.4.4. CKH, BUPAKB, İyonosfer, WGK ve GYA Veri Kümeleri ile Sınıflama Problemi

Bu örnekte, Bölüm 2.2.1’de verilen eğitim algoritmaları ile eğitilen küre ve elipsoit biçimli sınıflayıcıların, bilimsel yazındaki farklı sınıflayıcılar ile kıyaslamasını yapmak için, UCI makine öğrenme arşivinden CKH, BUPAKB, WGK ve iyonosfer veri kümeleri ile [75]’den GYA veri kümesi ele alınmıştır. Kıyaslama için EK-3’de tanıtılan DVM tabanlı sınıflayıcılardan kısıtsız, türevi alınabilir, dışbükey örtük Lagrangian işlevini Newton yöntemi kullanarak çözen doğrusal, RTİ çekirdekli ve veri sayısı belirli bir Oranda Azaltılan (AO) NLDVM [66, 67] ile DVM formülasyonundaki kısıtları eşitlik durumuna çevirerek, en yakın sınır oluşturan düzlemler oluşturan doğrusal, RTİ çekirdekli ve veri sayısı azaltılmış YDVM kullanılmıştır. Bunlar bilimsel yazında oldukça sık kullanıldıkları ve sınıflama uygulamalarında denektaşı oldukları için tercih edilmiştir. Ayrıca DVM tabanlı olmayan sınıflayıcılardan, elipsoit biçimli karar yüzeyleri sağlayan [39]’daki yöntem ve k-En yakın Komşuluk algoritması (k-EK) [83] kıyaslama için kullanılmıştır. Bunun dışında, Bölüm 2.1’de önerilen Küre Çekirdekli DVM (KÇDVM) de kıyaslama için ele alınmıştır. Veri kümelerine, [66, 67]’deki gibi bir ölçekleme uygulanmıştır.

Ele alınan veriler önce rasgele karıştırılmış, daha sonra 10 eşit alt kümeye bölünerek 4’ü eğitim 6’sı test kümesi olarak kullanılmıştır. [39]’daki sınıflayıcı hariç önerilen ve diğer sınıflayıcıların, seçilen parametrelerle 10 defa çalıştırılmasıyla elde edilen sonuçlar, ortalama eğitim süresi, ortalama adım sayısı ve hem eğitim hem de test verileri için doğru sınıflanma yüzdeleri, ortalama doğruluk olarak iki tablo halinde verilmiştir. Tüm sınıflayıcılar için algoritmanın her başlatılışında sınıflardaki veriler ve veri sayıları rasgele değiştirildiği için [39]’daki sınıflayıcının çözüm yapamadığı durumlar ortaya çıkmıştır. Bu nedenle [39]’daki sınıflayıcı da 10 defa çalıştırılmış ancak elde edilen en iyi sonuçları tabloda verilmiştir. Tablo 2.4’de CKH, BUPAKB ve iyonosfer veri kümeleri üzerinde farklı eğitim algoritmaları için önerilen sınıflayıcıların başarımları verilmiştir. Önerilen sınıflayıcılar, 350 adım için farklı eğitim iniş algoritması ile eğitilmiştir. Tablodan görüldüğü gibi GEİ yöntemi, diğer yöntemlerden eğitim süresi ve doğruluk açısından daha kötüdür. Buna karşılık MUÖEİ yöntemi, bilimsel yazında hem hızlı hem de daha iyi sonuç verdiği bilinen EEİ yöntemleriyle benzer başarımlar vermiştir. Bu nedenle, tezin geri kalanında verilen örneklerde aksi belirtilmedikçe, MUÖEİ yöntemi tercih edilmiştir. Tablo 2.5’de CKH, BUPAKB, WGK, iyonosfer ve GYA veri kümeleri üzerinde bu yöntem ile elde edilen sonuçlar, diğer yöntemlerin sonuçlarıyla kıyaslanmıştır.

Önerilen küre ve elipsoit biçimli sınıflayıcılar, kısa eğitim süreleri ve küçük adım sayılarıyla eğitilmiştir. Her veri kümesi için, Aşama-I’in MUÖEİ yöntemi ile 10 katın rasgele

birinden alınan eğitim sonuçları, Şekil 2.9'da verilmiştir. Aşama-II için ise eğitim sonuçları, Şekil 2.10'da verilmiştir. Görüldüğü gibi elipsoit biçimli yüzeyler ile ulaşılan hata daha düşüktür.

YDVM, NLDVM, AYDVM ve ANLDVM için [35, 36, 66, 67]'deki, DVM için [41]'deki DVMLight için ise [86]'daki MATLAB kütüphanesi kullanılmıştır. [39]'da verilen yöntemin sonuçlarını elde etmek için [92]'deki Sedumi eniyileme kütüphanesi kullanılmıştır.

DVM ve DVMLight için RTİ çekirdek kullanılarak sonuçlar elde edilmiştir. DVMLight ve DVM'nin C parametresi [1: 10: 5000] dizisi içerisinde, σ parametresi ise [0.01:0.01:3] dizisi içerisinde en iyi test başarımı veren değer seçilmiştir. Sonuçlardan görüldüğü gibi önerilen sınıflayıcı karar yüzeyleri sadece küre ve elipsoit biçimli olmasına rağmen, karmaşık karar yüzeyli DVM tabanlı yöntemlerle test başarımları açısından diğer yöntemlerden açık bir farkla üstündür. Süre açısından ise DVMLight, DVM ve [39]'daki yöntemden daha iyidir. Bununla birlikte Mangasarian'ın yöntemleri diğerlerine göre daha hızlıdır.

Sınıflardaki örnek sayısı eşit olmayan veri kümeleri, DVM gibi sınıflayıcılarda düşük doğruluk oranı ile sonuçlanabilir [36]. Bu çalışmada önerilen sınıflayıcılarda ise bu durum sınıflama başarımını etkilemez. Çünkü doğrudan karar sınırının oluşturulması ile ilgileniliyor ve yoğunluk temelli bir amaç ölçütü kullanılıyor. Ayrıca DVM'nin, hafıza probleminden dolayı çözemediği GYA veri kümesi ele alınarak, eğim iniş temelli algoritmalar sayesinde önerilen sınıflayıcıların çok girişli, büyük veri kümelerine uygulanabildiği gösterilmiştir.

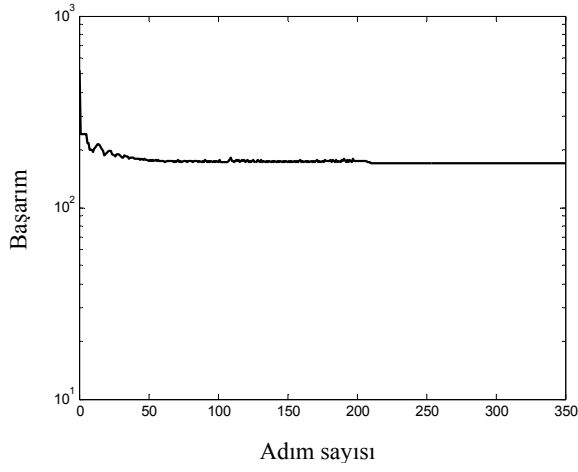
Tablo 2.4 Önerilen sınıflayıcıların, farklı eğitim algoritmaları kullanılarak elde edilen 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.

Veri Kümesi (veri sayısı × boyut)	İyonosfer 351 × 34	BUPAKB 345 × 6	CKH 297 × 13
AŞAMA-I			
GEİ			
Öğrenme Oranı	0.007	0.007	0.007
Eğitim Doğruluğu [%]	94.28	64.84	82.39
Test Doğruluğu [%]	89.04	61.67	74.05
Süre [s]	0.74	0.34	0.42
Adım Sayısı	350	350	350
MUÖEİ			
Eğitim Doğruluğu [%]	98.57	64.49	83.03
Test Doğruluğu [%]	91.42	62.13	75.66
Süre [s]	0.61	0.35	0.40
Adım Sayısı	350	350	350
EEİ			
Eğitim Doğruluğu [%]	96.42	63.04	81.51
Test Doğruluğu [%]	91.42	62.62	80.22
Süre [s]	1.16	0.90	0.68
Adım Sayısı	350/45	350/35	350/35
ÖEİ			
Eğitim Doğruluğu [%]	93.57	65.21	80.67
Test Doğruluğu [%]	92.85	57.76	77.96
Süre [s]	1.72	0.73	0.73
Adım Sayısı	350	350	350
AŞAMA-II			
GEİ			
Öğrenme Oranı	0.0005	10 ⁻⁶	7x10 ⁻⁶
Eğitim Doğruluğu [%]	98.57	67.39	90.75
Test Doğruluğu [%]	91.20	63.10	75.70
Süre [s]	1.77	0.94	0.99
Adım Sayısı	350	350	350
MUÖEİ			
Eğitim Doğruluğu [%]	98.87	73.66	96.78
Test Doğruluğu [%]	95.23	66.16	82.48
Süre [s]	0.61	0.82	0.65
Adım Sayısı	350/83	350	350/161
EEİ			
Eğitim Doğruluğu [%]	98.57	73.08	93.17
Test Doğruluğu [%]	93.33	66.01	79.18
Süre [s]	0.78	0.82	0.65
Adım Sayısı	350/12	350/30	350/14
ÖEİ			
Eğitim Doğruluğu [%]	96.23	75.98	94.63
Test Doğruluğu [%]	77.40	66.93	75.14
Süre [s]	1.43	1.03	1.45
Adım Sayısı	350	350	350

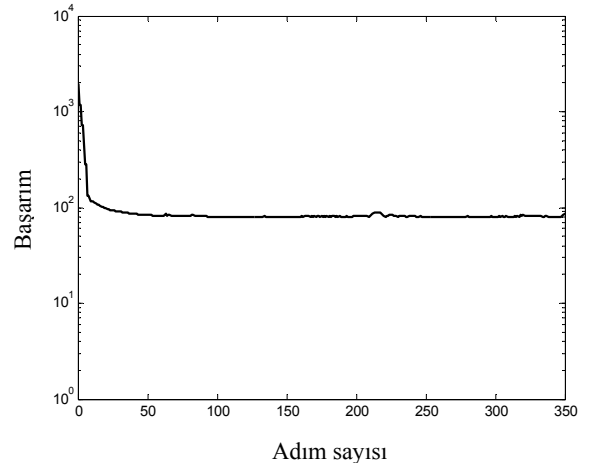
Tablo 2.5 Önerilen sınıflayıcıların diğer sınıflayıcılar ile 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.

Veri Kümesi L × n (veri sayısı × boyut)	İyonosfer 351 × 34	BUPAKB 345 × 6	CKH 297 × 13	WGK 110 x 32	GYA 4192 x 14
Doğrusal Çekirdekli NLDVM					
Eğitim Doğruluğu [%]	94.42	71.44	88.15	77.50	93.11
Test Doğruluğu [%]	85.71	65.48	83.66	64.76	93.59
Süre [s]	0.01	0	0	0	0.03
RTİ Çekirdekli NLDVM (ao=1)					
Eğitim Doğruluğu [%]	95.78	80.72	73.69	67.95	hafıza yetersiz
Test Doğruluğu [%]	96.09	81.74	71.01	66.92	
Süre [s]	0.99	0.32	0.34	0.03	
RTİ Çekirdekli ANLDVM (ao=0.5)					
Eğitim Doğruluğu [%]	95	75.36	72.94	68.40	hafıza yetersiz
Test Doğruluğu [%]	95.76	73.44	71.46	66.61	
Süre [s]	0.32	0.11	0.09	0.02	
RTİ Çekirdekli ANLDVM (ao=0.05)					
Eğitim Doğruluğu [%]	90.64	62.53	73.78	67.04	87.59
Test Doğruluğu [%]	91.61	61.89	70.84	67.84	87.47
Süre [s]	0.02	0	0	0	4.63
Doğrusal Çekirdekli YDVM					
Eğitim Doğruluğu [%]	89.71	69.05	86.47	77.50	92.93
Test Doğruluğu [%]	88.85	71.94	86.04	75.53	92.54
Süre [s]	0	0	0	0	0
RTİ Çekirdekli YDVM (ao=1)					
Eğitim Doğruluğu [%]	95.85	80.65	71.84	65.68	hafıza yetersiz
Test Doğruluğu [%]	94.95	80.09	72.14	68.46	
Süre [s]	0.44	0.16	0.14	0.03	
RTİ Çekirdekli AYDVM (ao=0.5)					
Eğitim Doğruluğu [%]	94.85	71.37	72.77	69.09	hafıza yetersiz
Test Doğruluğu [%]	94.52	73.59	71.63	65.53	
Süre [s]	0.17	0.05	0.05	0.01	
RTİ Çekirdekli AYDVM (ao=0.05)					
Eğitim Doğruluğu [%]	88.50	63.76	72.43	67.72	86.38
Test Doğruluğu [%]	88.23	61.60	71.75	65.23	86.33
Süre [s]	0.01	0.01	0	0	1.24
DVMLight					
Eğitim Doğruluğu [%]	100	100	100	100	96.36
Test Doğruluğu [%]	94.28	65.04	82.79	66.78	95.94
Süre [s]	0.22	0.13	0.74	0.12	3.63
σ, C	1, 1000	0.4, 1000	2, 1000	2.2, 1000	2, 1000

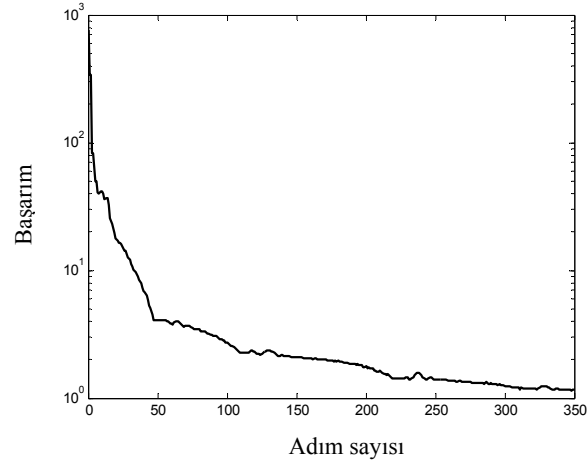
DVM Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] σ , C	100 93.33 5.04 0.2, 1000	99.27 57.76 3.28 1.04, 1000	99.15 77.97 2.22 0.4, 1000	97.73 64.15 0.64 0.0125, 1000	hafıza yetersiz
Küre Çekirdekli DVM Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] C	97.91 91.45 2.81 1	71.73 69.41 1.66 1	92.437 82.48 1.01 1	75.14 65.07 0.73 1	hafıza yetersiz
AŞAMA-I Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	98.57 91.42 0.61	64.49 62.13 0.35	83.03 75.66 0.40	63.63 61.538 0.88	90.041 89.14 45.24
AŞAMA-II Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	98.87 95.23 0.61	73.66 66.16 1.02	96.78 82.48 0.60	68.18 64.61 1.05	93.38 93.67 44.31
k-EK k Test Doğruluğu [%] Süre [s]	3 82.76 0.011	13 61.55 0.006	5 80.50 0.004	7 63.53 0.003	7 94.78 0.601
[39] Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] Formülasyon	100 88.57 21.32 KAOTE	69.56 57.76 0.95 EBAO	100 70.05 1.65 EBAO	82.50 63.38 8.66 KAOTE	-



a) BUPAKB

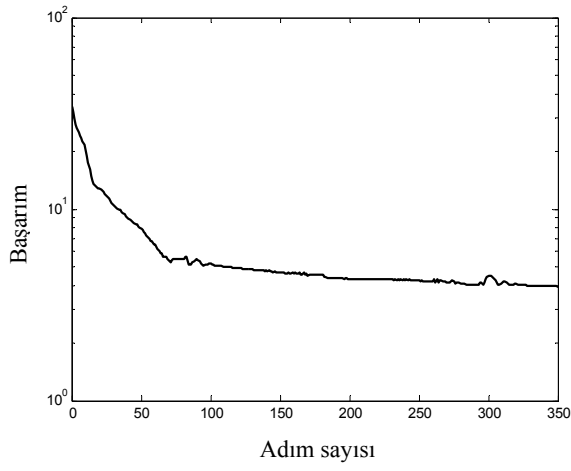


b) CKH

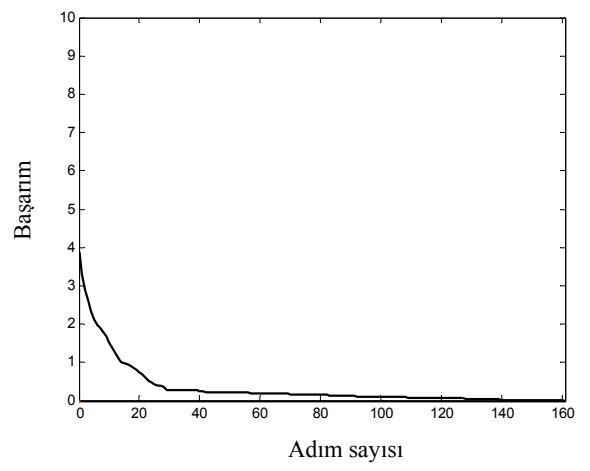


c) İyonosfer

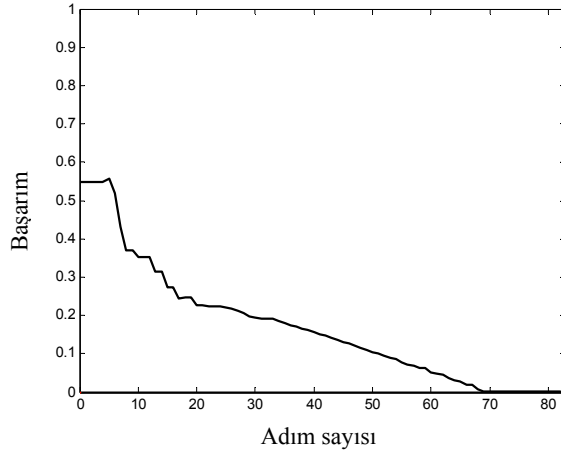
Şekil 2.9 Aşama-I'de önerilen eniyileme probleminin MUÖEİ algoritması ile eğitim başarımı.



a) BUPAKB



b) CKH



c) İyonosfer

Şekil 2.10 Aşama-II’de önerilen eniyileme probleminin MUÖEİ algoritması ile eğitim başarımı.

2.5. Çok Sınıflı Sınıflama

EK-2’de anlatıldığı gibi, DVM’ler iki sınıflı sınıflama problemleri için geliştirilmiştir. Halen DVM’ler için değişken sayısı, doğruluk ve eğitim süresi açılarından kabul edilebilir tek bir çoklu sınıflama formülasyonu geliştirilmemiştir. Çok sınıflı sınıflama problemleri için doğruluğu kabul edilebilir formülasyonlar, gerçekte problemi iki sınıflı sınıflama problemine ayırtırmayı içermektedir ve çok sınıflı problemlerde birden fazla sınıflayıcı aynı anda kullanılmaktadır. Bu bölümde, bu tezde önerilen sınıflayıcılar, doğrudan çoklu sınıflama yapan bir formülasyona genişletilmiştir.

2.5.1. Aşama-I: Küre Biçimli m-Sınıflı Sınıflayıcılar

$m > 2$ sınıflama problemi için, karar yüzeyi olarak kullanılan $R_m \in \mathcal{R}$ yarıçaplı, $c_m \in \mathcal{R}^n$ merkezli küreler,

$$D = \left\{ x \in \mathcal{R}^n \mid \left\| (x^i - c_m) \right\|^2 - R_m^2 = 0, \quad i = 1, \dots, L, \quad m = 1, \dots, M \right\}. \quad (2.74)$$

gibi tanımlanırsa, kürenin içindeki verilerin etiketlerinin +1, dışındakilerin -1 olduğu kabul edilerek

$$\left\| (x_i - c_m) \right\|^2 - R_m^2 \geq 0 \quad y_m^i = -1 \quad (2.75)$$

$$\left\| (x^i - c_m) \right\|^2 - R_m^2 \leq 0 \quad y_m^i = 1 \quad (2.76)$$

problemnin sınıflama hatası,

$$f(x) = f\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right) \quad (2.77)$$

olarak elde edilir. Burada, f işlevi kullanılarak kürenin dışında (içinde) olması gereken veri içinde (dışında) yerleşmişse amaç ölçütüne pozitif bir terim eklenmesi sağlanır,

Sonuç olarak, sınıflama hatasını enazlayan ve sınıfa ait verileri içeren en küçük hacimli m -tane ayrıştırıcı küreden oluşan problem için amaç ölçütü aşağıdaki gibi elde edilir,

$$g(R) = \sum_{m=1}^M R_m^2, \quad (2.78)$$

$$f(x) = \sum_{m=1}^M \sum_{i=1}^L f\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right), \quad (2.79)$$

$$F(x) = \sum_{m=1}^M \sum_{i=1}^L f\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right) + A \cdot \sum_{m=1}^M R_m^2, \quad (2.80)$$

$$\min_{c,R} \sum_{m=1}^M \sum_{i=1}^L f\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right) + A \cdot \sum_{m=1}^M R_m^2 \quad (2.81)$$

Burada, $A > 0$ dayanımlılık (doğru sınıflama) ve m . sınıfa ait dairenin hacmi arasında ödünleşimi sağlayan penaltı parametresidir. A , Bölüm 2.3.3'dekine benzer şekilde

$$A = 1/L \sum_{m=1}^M \sum_{i=1}^L f\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right) / \sum_{m=1}^M R_m^2 \quad (2.82)$$

bağıntısı uyarınca özyineli olarak hesaplanmasının benzetimlerde iyi sonuç verdiği gözlenmiştir.

2.5.1.1. Önerilen Küre Biçimli m -Sınıflı Sınıflayıcılar İçin Çözüm Yöntemi

m -sınıflı problem için önerilen eniyileme formülü, eğim iniş yöntemi kullanılarak çözülür. (2.80)'deki amaç ölçütü F 'nin, (2.85) ve (2.86)'daki gibi c_m ve R_m değişkenlerine göre negatif gradyenleri alınır. Değişkenlerin değerleri, iki sınıflı sınıflamaya benzer şekilde güncellenir,

$$-\frac{\partial F}{\partial c_m} = 2 \sum_{i=1}^L y_m^i (x^i - c_m) f'\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right), \quad (2.83)$$

$$-\frac{\partial F}{\partial R_m} = 2 \sum_{i=1}^L y_m^i f'\left(y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right)\right) R_m - 2A R_m, \quad (2.84)$$

$$c_m(k+1) = c_m(k) - \eta \frac{\partial F}{\partial c_m}, \quad (2.85)$$

$$R_m(k+1) = R_m(k) - \eta \frac{\partial F}{\partial R_m}. \quad (2.86)$$

(2.87)'deki karar işlevinde, güncellenmiş merkez c_m ve yarıçap R_m değerleri kullanılarak verilerin ait olduğu sınıflar belirlenir.

$$\ell(x) = \text{sgn}(R_m^2 - \|x - c_m\|^2) \quad (2.87)$$

2.5.2. Aşama-II: Elipsoit Biçimli m-Sınıflı Sınıflayıcılar

Bu aşamada $m > 2$ sınıflama problemi için, karar yüzeyi olarak $\Sigma_m \in \mathbb{R}^{n \times n}$ kovaryans matrisine sahip, $c_m \in \mathbb{R}^n$ merkezli elipsoitler kullanılır,

$$E = \{x \in \mathbb{R}^n \mid ((x^i - c_m) \Sigma_m (x^i - c_m) - 1) = 0, \quad i = 1, \dots, L, \quad m = 1, \dots, M\}. \quad (2.88)$$

Birinci aşamaya benzer olarak, her elipsoidin içindeki verilerin etiketlerinin +1, dışındakilerin -1 olduğu kabul edilerek,

$$((x^i - c_m) \Sigma_m (x^i - c_m) - 1) \geq 0 \quad y_m^i = -1, \quad (2.89)$$

$$((x^i - c_m) \Sigma_m (x^i - c_m) - 1) \leq 0 \quad y_m^i = 1, \quad (2.90)$$

problemin sınıflama hatası

$$f(x) = \sum_{m=1}^M \sum_{i=1}^L f(y_m^i ((x^i - c_m) \Sigma_m (x^i - c_m) - 1)). \quad (2.91)$$

ve yapısal hatası

$$g(\Sigma) = \sum_{m=1}^M A / \sqrt{|\Sigma_m|} \quad (2.92)$$

olarak tanımlanırsa,

$$F(x) = \sum_{m=1}^M \sum_{i=1}^L f(y_m^i ((x^i - c_m) \Sigma_m (x^i - c_m) - 1)) + \sum_{m=1}^M A / \sqrt{|\Sigma_m|} \quad (2.93)$$

amaç ölçütü elde edilir.

Sonuç olarak, giriş örüntülerini içeren en küçük hacimli elipsoitleri ararken aynı anda sınıflama hatasını da enazlayan sınıflama problemi

$$\min_{c, \Sigma} \sum_{m=1}^M \sum_{i=1}^L f(y_m^i ((x^i - c_m) \Sigma_m (x^i - c_m) - 1)) + \sum_{m=1}^M A / \sqrt{|\Sigma_m|}. \quad (2.94)$$

gibi oluşturulur.

2.5.2.1. Önerilen Elipsoit Biçimli Sınıflayıcılar İçin Çözüm Yöntemi

(2.94)'deki eniyileme problemi, Bölüm 2.2.1'de verilen eğim iniş yöntemlerinden birisi kullanılarak çözülebilir. Bunun için öncelikle (2.93)'teki amaç ölçütü F 'nin, c ve Σ değişkenlerine göre negatif gradyenleri alınır,

$$-\frac{\partial F}{\partial \Sigma_m} = -\sum_{i=1}^L y_m^i (x^i - c_m)(x_i - c_m)^T f'(y_m^i ((x^i - c_m)\Sigma_m(x^i - c_m) - 1)) + A(\Sigma_m^{-1})^T / 2\sqrt{|\Sigma_m|}, \quad (2.95)$$

$$-\frac{\partial F}{\partial c_m} = 2\sum_{i=1}^L y_m^i (x^i - c_m) f'(y_m^i ((x^i - c_m)\Sigma_m(x^i - c_m) - 1)). \quad (2.96)$$

Aşama-I'den elde edilen merkezler, elipsoitlerin merkezleri olarak alınır ve elde edilen yarıçaplar ise $1/R_m^2$ ile kovaryans matrislerin diyagonal elamanları olarak başlatılır. Merkez değerleri ve kovaryans matrisleri

$$c_m(k+1) = c_m(k) - \eta \frac{\partial F}{\partial c_m}, \quad (2.97)$$

$$\Sigma_m(k+1) = \Sigma_m(k) - \eta \frac{\partial F}{\partial \Sigma_m}. \quad (2.98)$$

olarak güncellenir. Elde edilen c_m ve Σ_m değerleri kullanılarak karar yüzeyi belirlenir.

2.6. Çok Sınıflı Sınıflama İçin Uygulamalar

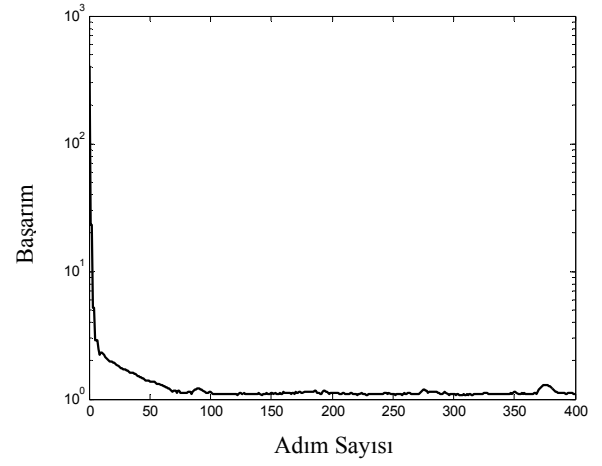
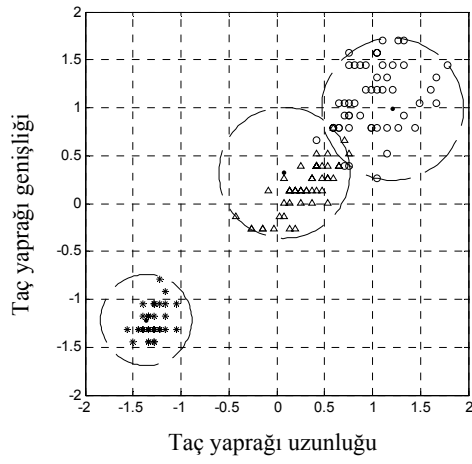
Bu bölümde, görsel ve kıyaslama amaçlı iki uygulama yapılmıştır. İlk uygulamada üç sınıflı Zambak çiçeği verisinin, 3. ve 4. karakteristiği alınarak iki boyutta çok sınıflı ayrıştırma problemi düşünülmüştür. Daha sonra 1., 3. ve 4. karakteristikleri alınarak, 3 boyutlu 3 sınıflı sınıflama problemi düşünülmüştür. Ayrıca [29]'dan 99 veriden oluşan iki boyutlu beşgen verisi alınarak, 5 sınıflı sınıflama problemi ele alınmıştır. Tüm veriler eğitim için kullanılmıştır. Elde edilen karar yüzeyleri ve önerilen yöntemin, MUÖEİ yöntemi ile eğitim başarımları 2.11-2.16 şekillerinde verilmiştir. Hem karar yüzeylerinden hem de eğitim sonuçlarından görüleceği gibi oluşturulan çok sınıflı sınıflayıcı oldukça iyi bir şekilde çalışmaktadır.

İkinci uygulamada, birden fazla DVM'nin aynı anda kullanımını gerektiren çoklu sınıflama yöntemleri olan bire karşı biri (BKB) ve bire karşı diğerleri (BKD) yöntemleri ile kıyaslamak için UCI arşivinden, zambak çiçeği, şarap, cam ve uydu görüntüleri veri kümeleri alınmıştır. Tüm veriler sıfır ortalamalı standart sapması 1 olan verilere ölçeklenmiştir. Uydu görüntüleri veri kümesi hariç, diğer veri kümelerinin test verisi olmadığı için, veriler önce

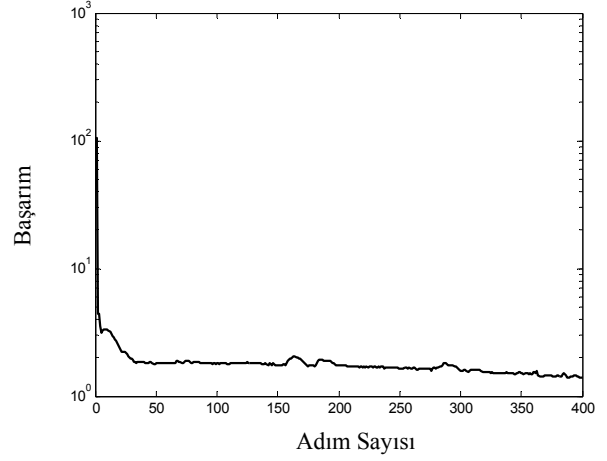
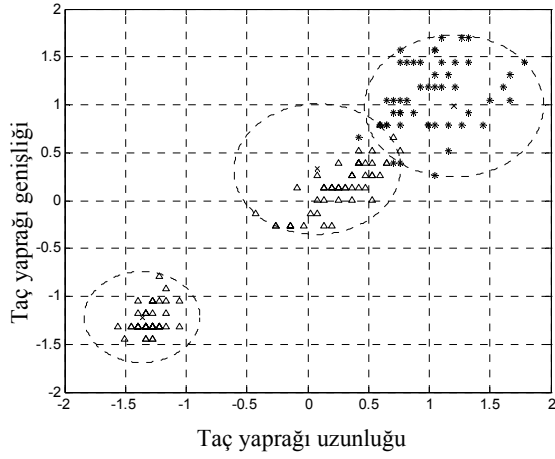
rasgele karıştırılmış daha sonra 10 eşit alt kümeye bölünerek 4'ü eğitim 6'sı test kümesi olarak kullanılmıştır. Uydu görüntüleri veri kümesinin ise 4435 verisi eğitim, 2000 verisi test için kullanılmıştır. DVM'nin C parametresi $[1 \ 10 \ 100 \ \dots \ 10^{12}]$, σ parametresi ise $[0.01: 0.01:10]$ dizisi içerisinde en iyi test başarımı veren değer seçilmiştir. Burada, çoklu sınıflama için BKB ve BKD sonuçları, [86]'daki ardışıl en küçük eniyileme algoritması kullanan MATLAB eniyileme kütüphanesi ile elde edilmiştir. Önerilen sınıflayıcı, 350 adım için çalıştırılmış ve sınıflanmayan veriler ise en yakın küre veya elipsoide ait sınıflara atanmıştır. Önerilen sınıflayıcı ve parametreleri belirlenen DVM için, 10 defa çalıştırılarak elde edilen sonuçlar ortalama eğitim süresi ve hem eğitim hem de test verileri için doğru sınıflanma yüzdeleri ortalama doğruluk olarak Tablo 2.6'da verilmiştir. Buradan görüldüğü gibi, giriş uzayında elipsoit biçimine yakın dağılımı olan veri kümelerinde, önerilen sınıflayıcılarda DVM yöntemlerinden daha iyi sonuçlar elde edilmiştir. Üstelik önerilen yöntemin daha hızlı olduğu açıkça ortaya çıkmıştır.

Tablo 2.6 Denektaşı veri kümeleri üzerinde BKB, BKD ve önerilen yöntemin kıyaslanması.

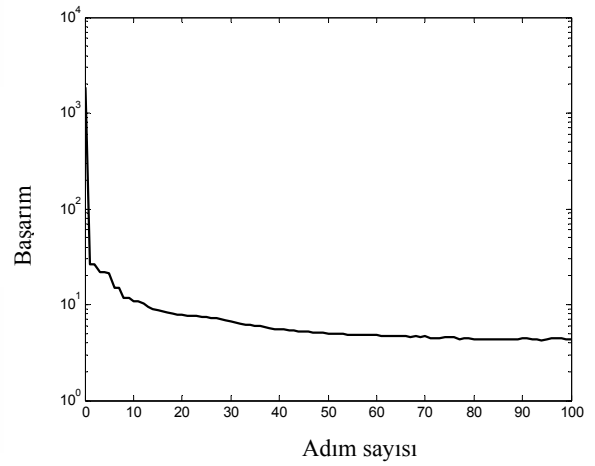
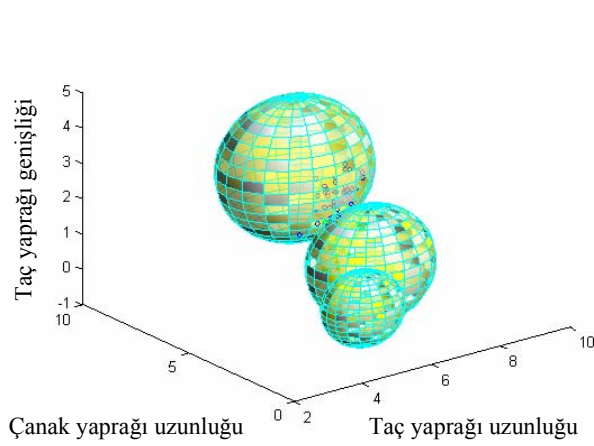
Veri Kümesi (n,L,m)	Aşama-I	Aşama-II	BKB	BKD
Zambak Çiçeği (4,150,3) Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	97 96.40 0.115	99.66 97.45 0.2	100 94.38 0.001 $C=10^3, \sigma=4$	99.66 94.15 0.002 $C=10^3, \sigma=1$
Şarap (13,178,3) Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	98.16 93.09 0.1	100 97.45 0.12	100 97.45 0.002 $C=10^3, \sigma=2$	100 96.98 0.002 $C=10^3, \sigma=2$
Cam (9, 214,7) Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	65.00 55.71 0.31	79.45 61.17 0.34	93.02 60.62 0.15 $C=10^5, \sigma=2.5$	98.95 58.18 0.09 $C=10^5, \sigma=2.5$
Uydu Görüntüleri (36, 6435,6) Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	87.18 79.57 3502.7	90.15 82.68 3345.9	83.06 77.65 3120 $C=10^3, \sigma=1.5$	88.12 81.76 231234 $C=10^3, \sigma=2$



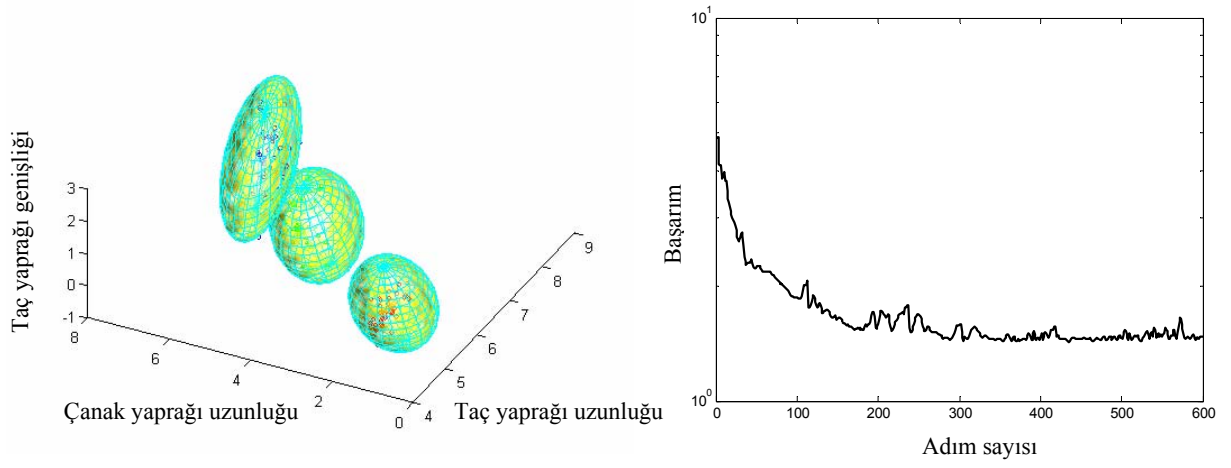
Şekil 2.11 Aşama-I’de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.



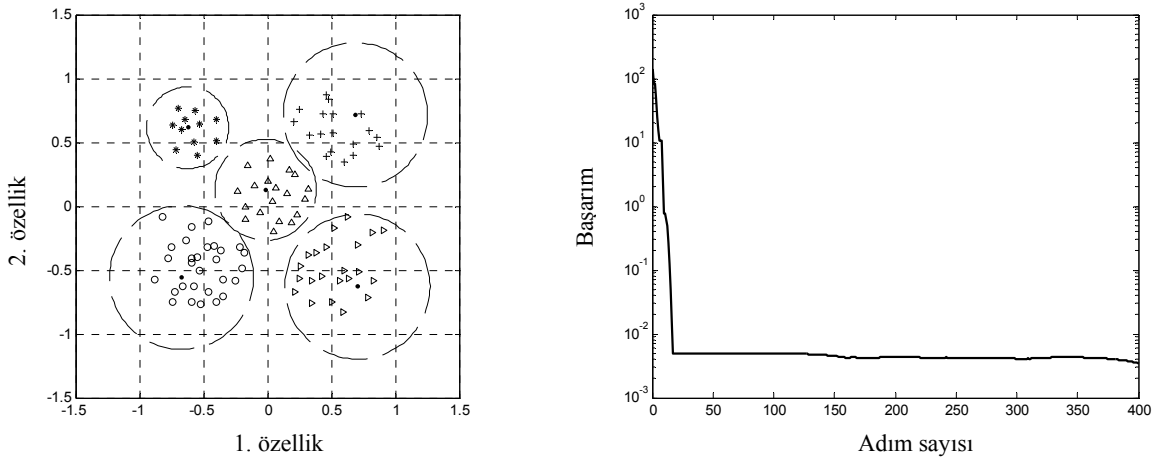
Şekil 2.12 Aşama-II’de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.



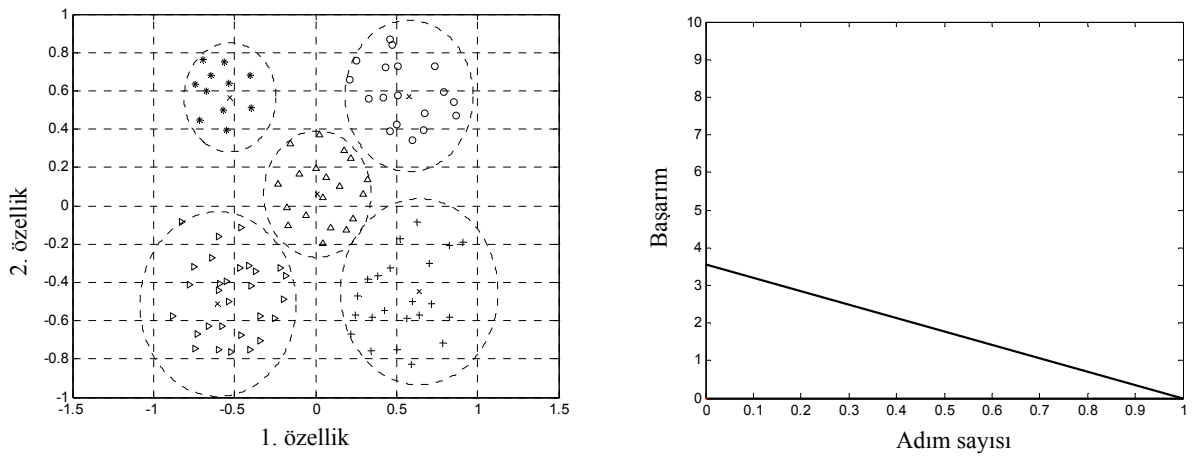
Şekil 2.13 Aşama-I’de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.



Şekil 2.14 Aşama-II’de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.



Şekil 2.15 Aşama-I’de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.



Şekil 2.16 Aşama-II’de önerilen çok sınıflı eniyileme probleminin eğitim başarımı ve karar yüzeyleri.

3. BULANIK KÜRE ve ELİPSOİT BİÇİMLİ SINIFLAYICILAR

3.1. Önerilen Bulanık Küre ve Elipsoit Biçimli Sınıflama Probleminin Tanıtımı

Bölüm 2’de önerilen küre ve elipsoit biçimli sınıflayıcılar, veri kümesini hemen hemen kapsayan en küçük hacimli yapılar olduklarından, gürültü ve aykırı verileri dışlarlar ve onlardan etkilenmezler. Ayrıca küre ve elipslerle de ayrıştırılamayan veri uygulamalarında, sınıflama ve hacim arasındaki ödünleşimin en az sayıda veriyi göz ardı ederek düzgün ayrışım yüzeyleri sağladığı görülmüştür. Bu anlamda gürbüz sınıflayıcılardır. Bu bölümde, Bölüm 2’deki yapılarda oluşacak küçük değişikliklerle, gürültülü ve aykırı değerli problemlerde gürbüzlüğü artırmak için farklı üyelik işlevlerinin kullanımı ile elde edilen sonuçlar verilmiştir.

Bulanık sınıflayıcıların, Bölüm 2’deki sınıflayıcılardan farkı; bulanık sınıflama problemlerindeki her bir eğitim örneğinin karar yüzeyinin tanımlanmasına katkısının, örneğe ait üyelik değeriyle belirlenmesidir. Her verinin üyelik değeri bir olduğu zaman Bölüm 2’de tanımlanan sınıflayıcılar, bulanık sınıflayıcılarla tamamen aynı olur. Bulanık sınıflayıcılarda önce, veriler bulanık üyelik işlevleri kullanılarak bulanıklaştırılır. Böylece, eğitim boyunca eğitim örneklerinin bulanık doğası da hesaba katılır. Sonra, yüksek belirsizlikli örneklerin etkisi, eğitim örneğine atanan üyelik değeriyle ağırlıklandırılarak sınıflama hatası azaltılır. Burada bir önceki ile farklı olması açısından, bu sınıflayıcılar, bulanık küre biçimli ve bulanık elipsoit biçimli sınıflayıcılar olarak adlandırılmıştır.

3.2. Aşama-I: Bulanık Küre Biçimli Sınıflayıcılar

Bulanık sınıflayıcılarda, L tane veri (3.1)’deki gibi hazırlanır,

$$\begin{aligned}\Theta &= \left\{ (x^1, y^1, s^1), (x^2, y^2, s^2), \dots, (x^L, y^L, s^L) \right\}, \\ x^i &\in R^n, \\ y^i &\in \{-1, 1\}, \\ s^i &\in S.\end{aligned}\tag{3.1}$$

Burada, $\rho < s^i < 1$ eğitim noktaları için uygun üyelik işlevleri kullanılarak elde edilen bulanık üyelik değerlerini [47, 48-50, 62, 97, 107], S ise bulanık üyelik değerlerinden oluşan bir kümeyi göstermektedir. Üyelik değeri $s^i = 1$ olan veri, sınıflamada en çok dikkat edilecek veri iken,

üyelik değeri $s^i = \rho$ olan veri ise eğitimdeki en önemsiz, hatalı sınıflandırılacak veri olarak kabul edilir. Farklı üyelik işlevleri probleme uygun olarak seçilebilir.

Bölüm 2’de tanımlandığı gibi karar yüzeyi olarak $R \in \Re$ yarıçaplı, $c \in \Re^n$ merkezli bir küre alınırsa, problemin yeni deneysel hatası.

$$f(x) = f\left(s^i y^i \left(\|x^i - c\|^2 - R^2 \right)\right) \quad (3.2)$$

ile tanımlanır. Bu terim, bulanık BDVM’lerdeki gibi farklı ağırlıklı hata terimleri olarak yorumlanabilir [62]. Yapısal hata olarak

$$g(R) = R^2 \quad (3.3)$$

alınırsa yeni amaç ölçütü

$$F(x) = \sum_{i=1}^L f\left(s^i y^i \left(\|x^i - c\|^2 - R^2 \right)\right) + AR^2 \quad (3.4)$$

olarak bulunur. Sonuç kısıtsız eniyileme problemi

$$\min_{c, R} \sum_{i=1}^L f\left(s^i y^i \left(\|x^i - c\|^2 - R^2 \right)\right) + AR^2 \quad (3.5)$$

şeklinde elde edilir. Burada, $A > 0$ ödünleşimi sağlayan penaltı parametresi

$$A = \frac{1}{L} \frac{\sum_{i=1}^L f\left(s^i y^i \left(\|x^i - c\|^2 - R^2 \right)\right)}{R^2} \quad (3.6)$$

olarak hesaplanır.

3.2.1. Bulanık Küre Biçimli Sınıflayıcılar İçin Çözüm Yöntemi

Önerilen yöntemde, (3.5)’de verilen eniyileme problemi, en dik iniş algoritmaları uygulanarak çözülmektedir. Bunun için öncelikle amaç ölçütü F ’nin, c ve R değişkenlerine göre negatif gradyenleri alınır,

$$-\frac{\partial F}{\partial c} = 2 \sum_{i=1}^L s^i y^i (x^i - c) f'\left(y^i \left(\|x^i - c\|^2 - R^2 \right)\right), \quad (3.7)$$

$$-\frac{\partial F}{\partial R} = 2 \sum_{i=1}^L s^i y^i f'\left(y^i \left(\|x^i - c\|^2 - R^2 \right)\right) R - 2AR. \quad (3.8)$$

Daha sonra seçilen algoritma gereğince özyineli olarak güncellenir. Örneğin en dik iniş algoritması kullanılırsa özyineleme,

$$c(k+1) = c(k) - \eta \frac{\partial F}{\partial c}, \quad (3.9)$$

$$R(k+1) = R(k) - \eta \frac{\partial F}{\partial R}. \quad (3.10)$$

ile yapılır. Özyineleme sonucunda elde edilen R ve c değerleri kullanıldığında,

$$\ell(x) = \text{sgn}\left(R^2 - \|(x - c)\|^2\right) \quad (3.11)$$

olarak karar işlevi elde edilir. Yüzey içinde bulunan noktalar karar işlevinin pozitif değer verdiği, yüzeyin dışındaki noktalar ise negatif değer verdiği noktalar.

3.3. Aşama-II: Bulanık Elipsoit Biçimli Sınıflayıcılar

Bu aşamada, Bölüm 2’de tanımlandığı gibi karar yüzeyi olarak $\Sigma \in \mathfrak{R}^{n \times n}$ kovaryans matrisine sahip, $c \in \mathfrak{R}^n$ merkezli bir elipsoit biçimi kullanılırsa, problemin yeni deneysel hatası

$$f(x) = f(s^i y^i ((x^i - c) \Sigma (x^i - c) - 1)) \quad (3.12)$$

ve yapısal hatası

$$g(\Sigma) = A / \sqrt{|\Sigma|} \quad (3.13)$$

olarak elde edilir. Amaç ölçütü ise

$$F(x) = \sum_{i=1}^L f(s^i y^i ((x^i - c) \Sigma (x^i - c) - 1)) + A / \sqrt{|\Sigma|} \quad (3.14)$$

ile verilir. Birinci aşamaya benzer olarak, aynı anda sınıflama hatasını ve giriş örüntülerini içeren en küçük hacimli, elipsoit biçimli sınıflayıcı problemi

$$\min_{c, \Sigma} \sum_{i=1}^L f(s^i y^i ((x^i - c) \Sigma (x^i - c) - 1)) + A / \sqrt{|\Sigma|} \quad (3.15)$$

gibi oluşturulur. Burada, $A > 0$ ödünleşimi sağlayan penaltı parametresi

$$A = \frac{1}{L} \frac{\sum_{i=1}^L f(s^i y^i ((x^i - c) \Sigma (x^i - c) - 1))}{1 / \sqrt{|\Sigma|}} \quad (3.16)$$

yardımıyla ardışık olarak hesaplanır.

3.3.1. Bulanık Elipsoit Biçimli Sınıflayıcılar İçin Çözüm Yöntemi

(3.15)’deki eniyileme probleminin eğitim iniş algoritmalarıyla çözülebilmesi için, öncelikle amaç ölçütü F ’nin, c ve Σ değişkenlerine göre negatif gradyenleri alınır.

$$-\frac{\partial F}{\partial \Sigma} = -\sum_{i=1}^L s^i y^i (x^i - c)(x^i - c)^T f' \left(s^i y^i (x^i - c)^T \Sigma (x^i - c) - 1 \right) + A \Sigma^{-T} / 2\sqrt{|\Sigma|} \quad (3.17)$$

$$-\frac{\partial F}{\partial c} = 2 \sum_{i=1}^L s^i y^i (x^i - c) f' \left(s^i y^i \left((x^i - c)^T \Sigma (x^i - c) - 1 \right) \right) \quad (3.18)$$

Daha sonra en dik iniş algoritması örneği için, (3.19) ve (3.20)'deki gibi özyineli olarak güncellenir

$$c(k+1) = c(k) - \eta \frac{\partial F}{\partial c}, \quad (3.19)$$

$$\Sigma(k+1) = R(k) - \eta \frac{\partial F}{\partial \Sigma}, \quad (3.20)$$

Başlangıçta, birinci aşamadan elde edilen merkezler, elipsin merkezleri ve elde edilen yarıçaplar ise $1/R^2$ ile kovaryans matrisinin diyagonal elamanları alınır.

(3.21) kullanılarak yüzeyin içinde bulunan noktalar pozitif, dışındakiler ise negatif olarak etiketlenir

$$\ell(x) = \text{sgn} \left(1 - (x^i - c) \Sigma (x^i - c) \right). \quad (3.21)$$

3.4. m-Sınıflı Bulanık Küre ve Elipsoit Biçimli Sınıflayıcılar

3.4.1. Aşama-I: Küre Biçimli m-Sınıflı Bulanık Sınıflayıcılar

$m > 2$ sınıflı sınıflama problemi için, karar yüzeyi olarak $R_m \in \mathfrak{R}$ yarıçaplı ve $c_m \in \mathfrak{R}^n$ merkezli küreler kullanılırsa, problemin deneysel hatası

$$f(x) = f \left(s_m^i y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right) \right) \quad (3.22)$$

ve yapısal hatası

$$g(R) = \sum_{m=1}^M R_m^2 \quad (3.23)$$

olur. Amaç ölçütü

$$F(x) = \sum_{m=1}^M \sum_{i=1}^L f \left(s_m^i y_m^i \left(\left\| (x^i - c_m) \right\|^2 - R_m^2 \right) \right) + A \sum_{m=1}^M R_m^2 \quad (3.24)$$

olarak elde edilir.

Sınıflama hatasını enazlayarak verileri içeren en küçük hacimli m-tane ayrıştırıcı küreden oluşan eniyileme problemi

$$\min_{c,R} \sum_{m=1}^M \sum_{i=1}^L f\left(s_m^i y_m^i \left(\|x^i - c_m\|^2 - R_m^2\right)\right) + A \sum_{m=1}^M R_m^2 \quad (3.25)$$

olarak oluşturulur. Burada, $A > 0$ penaltı parametresi

$$A = \frac{1}{L} \frac{\sum_{m=1}^M \sum_{i=1}^L f\left(s_m^i y_m^i \left(\|x^i - c_m\|^2 - R_m^2\right)\right)}{\sum_{m=1}^M R_m^2} \quad (3.26)$$

olarak hesaplanabilir veya yukarıda verilen formüllere benzer olarak özyineleme sayısına göre üstel olarak azaltılabilir.

Önerilen yöntemde m -sınıflı problem için elde edilen eniyileme probleminin, eğim iniş yöntemi kullanılarak çözülebilmesi için, öncelikle amaç ölçütü F 'nin, c_m ve R_m değişkenlerine göre negatif gradyenleri alınır

$$-\frac{\partial F}{\partial c_m} = 2 \sum_{i=1}^L s_m^i y_m^i (x^i - c_m) f'\left(y_m^i \left(\|x^i - c_m\|^2 - R_m^2\right)\right), \quad (3.27)$$

$$-\frac{\partial F}{\partial R_m} = 2 \sum_{i=1}^L s_m^i y_m^i f'\left(y_m^i \left(\|x^i - c_m\|^2 - R_m^2\right)\right) R_m - 2AR_m, \quad (3.28)$$

daha sonra algoritmaya özgün formüller ile özyineli olarak güncellenir.

3.4.2. Aşama-II: Elipsoit Biçimli m -Sınıflı Bulanık Sınıflayıcılar

Bu aşamada m -sınıflı sınıflama problemi için, karar yüzeyi olarak kovaryans matrisine sahip, $c_k \in \mathbb{R}^n$ merkezli elipsler kullanılarak problemin sınıflama hatası

$$f(x) = \sum_{m=1}^M \sum_{i=1}^L f\left(s_m^i y_m^i \left((x^i - c_m) \Sigma_m (x^i - c_m) - 1\right)\right) \quad (3.29)$$

ve yapısal hatası

$$g(\Sigma) = \sum_{m=1}^M 1/\sqrt{|\Sigma_m|} \quad (3.30)$$

ile tanımlanır. Birinci aşamaya benzer olarak, aynı anda sınıflama hatasını ve giriş örüntülerini içeren en küçük hacimli elipsoit biçimli yüzey ile sınıflama problemi için amaç ölçütü

$$F(x) = \sum_{m=1}^M \sum_{i=1}^L f\left(s_m^i y_m^i \left((x_i - c_m) \Sigma_m (x_i - c_m) - 1\right)\right) + \sum_{m=1}^M A/\sqrt{|\Sigma_m|} \quad (3.31)$$

biçiminde oluşturulur.

$$\min_{c, \Sigma} \sum_{m=1}^M \sum_{i=1}^L f(s_m^i y_m^i ((x^i - c_m) \Sigma_m (x^i - c_m) - 1)) + \sum_{m=1}^M A / \sqrt{|\Sigma_m|} \quad (3.32)$$

Eniyileme probleminin, eğitim iniş algoritmalarıyla çözülebilmesi için, öncelikle amaç ölçütü F 'nin, c ve Σ değişkenlerine göre negatif gradyenleri alınır,

$$-\frac{\partial F}{\partial \Sigma_m} = -\sum_{i=1}^L s_m^i y_m^i (x^i - c_m) (x^i - c_m)^T f'(y_m^i ((x^i - c_m) \Sigma_m (x^i - c_m) - 1)) + A (\Sigma_m^{-1})^T / 2\sqrt{|\Sigma_m|}, \quad (3.33)$$

$$-\frac{\partial F}{\partial c_m} = 2 \sum_{i=1}^L s_m^i y_m^i (x^i - c_m) f'(s_m^i y_m^i ((x^i - c_m) \Sigma_m (x^i - c_m) - 1)) \quad (3.34)$$

daha sonra elde edilen gradyenler kullanılarak, parametreler özyineli olarak güncellenir.

3.5. Uygulamalar

Bu bölümde üç farklı örnek ele alınmıştır. İlk örnekte; sezgisel olarak, düşünülen bulanık sınıflayıcıların sınıflama başarımlarına etkisini göstermek için yapay örnekler kullanılmıştır. İkinci örnekte, denektaş veri kümesi olan Fisher'in zambak çiçeği veri kümesi ile önerilen bulanık sınıflayıcıların başarımları, Bölüm 2'de önerilen sınıflayıcılar ile kıyaslanmıştır. Üçüncü örnekte ise doğal olarak gürültü içeren, gerçek yaşamdaki kanser veri kümeleri ele alınmıştır.

3.5.1. Yapay Veri Kümesi ile Sınıflama Problemi

Bölüm 2.4'te tanıtılan, YV22 yapay veri kümesinde (10,10) verisi iki sınıftan da çok uzak bulunmaktadır. Bölüm 2'de küre biçimli ve elipsoit biçimli sınıflayıcılarla elde edilen sonuç karar yüzeylerinden görüldüğü gibi, bu veri noktası kapsanmaya çalışılmış veya bu noktaya çok yaklaşılmıştır. Burada, üyelik işlevleri tanıtılarak, bu veri noktasının aykırı değer olarak kabul edilmesi amaçlanmıştır. Bu örnekte YV22 veri kümesindeki aykırı veri olarak kabul edilen (10,10) verisinin sınıflayıcının tasarımına etkisini azaltmak için, Şekil 3.1'de gösterilen sınıf merkezi yöntemi temeline dayanan, (3.35)'de verilen üyelik işlevi tercih edilmiştir [62].

$$s^i = \begin{cases} 1 - |x^+ - x^i| / (r^+ + \delta) & y^i = 1 \text{ için} \\ 1 - |x^- - x^i| / (r^- + \delta) & y^i = -1 \text{ için} \end{cases} \quad (3.35)$$

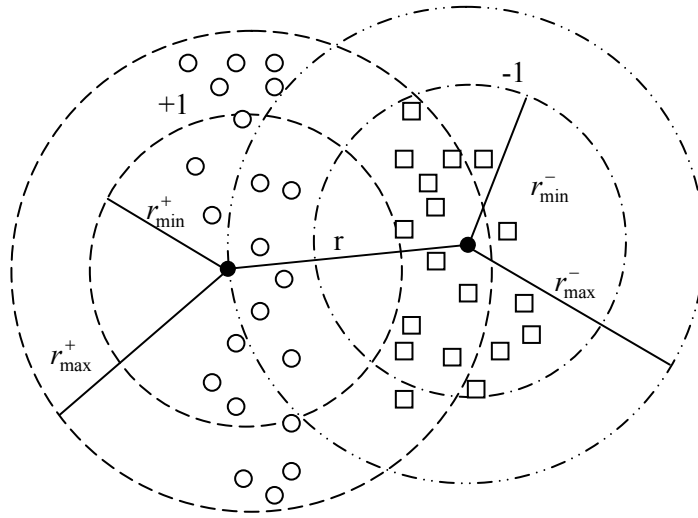
Burada, x^+ , +1 sınıfının ortalamasını ve x^- , -1 sınıfının ortalamasını göstermektedir. +1 sınıfının yarıçapı

$$r^+ = \max_{\{x_i, y_i=1\}} |x^+ - x^i| \quad (3.36)$$

ve -1 sınıfının yarıçapı

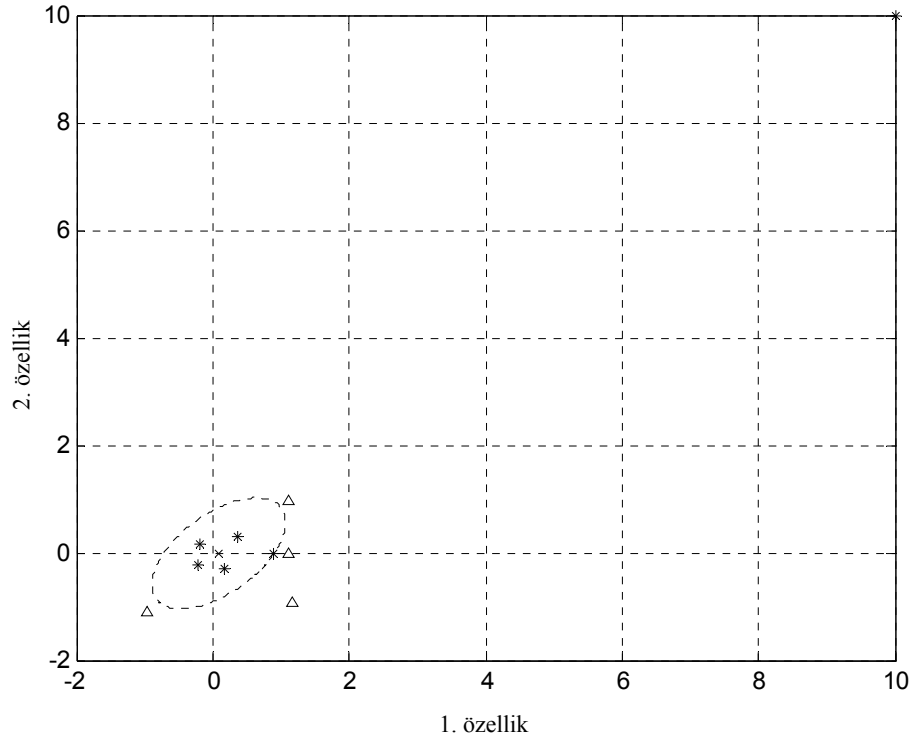
$$r^- = \max_{\{x_i, y_i=-1\}} |x^- - x^i| \quad (3.37)$$

ile hesaplanmıştır. $\delta > 0$, $s^i = 0$ durumundan kaçınmak için kullanılmıştır.

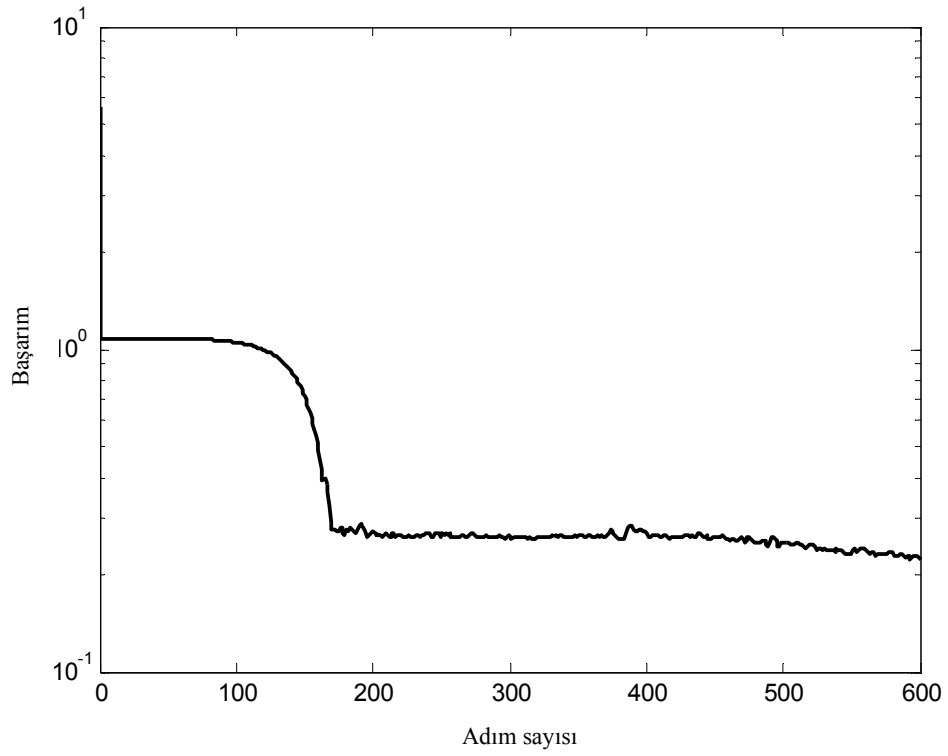


Şekil 3.1 Üyelik işlevlerinin elde edilmesi için merkez yönteminin geometrik gösterimi.

Burada $\delta=0.03$ alınmıştır. Önerilen sınıflayıcılar, üç örnek için de momentum terimli uyarlanırlı öğrenme oranlı eğim iniş algoritması (MUÖEİ) ile eğitilmiştir. Merkez c değerleri, eğitim verileri içerisinde her kat için rasgele ve R yarıçap değerleri ise 1 ile başlatılmıştır. Hem küre hem de elipsoit biçimli bulanık sınıflayıcılar için bu veri, karar yüzeyinin dışında bırakılmıştır. Böylece istenilen amaca ulaşılmıştır. Şekil 3.2’de bulanık elipsoit biçimli sınıflayıcının karar yüzeyi ve Şekil 3.3’de eğitim sonuçları verilmiştir.



Şekil 3.2 YV22, yapay veri kümesi için bulanık elipsoit biçimli sınıflayıcıların sonuçları.



Şekil 3.3 MUÖEİ algoritması ile önerilen bulanık sınıflayıcının eğitim başarımının değişimi.

3.5.2. Zambak Çiçeği Veri Kümesi ile İki Sınıflı Sınıflama Problemi

Bu örnekte, Zambak çiçeği verisinin önce 3. ve 4. karakteristikleri alınarak iki boyutlu giriş uzayı ile daha sonra tüm dört karakteristiği alınarak dört boyutlu giriş uzayı ile ikili sınıflama yapılmıştır. Burada, Keller ve Hunt'un bulanık algılayıcılar için önerdiği üyelik işlevi kullanılmıştır [8]. Bu üyelik işlevi, bulanık DVM sınıflayıcılara da uygulanmıştır [17, 49]. 1. sınıfa ait bir örnek verildiğinde, hem -1 hem de 1 sınıfı için üyelik işlevi

$$s_1^i = 0.5 + \frac{[\exp(e(d_{-1}(x^i) - d_1(x^i)))/d] - \exp(-e)}{2(\exp(e) - \exp(-e))}$$
$$s_{-1}^i = 1 - s_1^i \quad (3.38)$$

ile hesaplanmıştır. Burada, e , üyelik değerlerinin 0.5'e doğru azalma oranını belirleyen sabiti $d_1(x_i)$ ve $d_{-1}(x_i)$ her bir verinin sırasıyla 1 ve -1 sınıflarının ortalaması arasındaki mesafeleri ve d ise iki sınıfın ağırlık merkezi arasındaki mesafeyi göstermektedir.

-1 sınıfına ait üyelik değerleri verildiği zaman, x^i 'nin üyelik işlevi, hem 1 hem de -1 sınıfı için aşağıdaki gibi verilmiştir.

$$s_{-1}^i = 0.5 + \frac{[\exp(e(d_1(x^i) - d_{-1}(x^i)))/d] - \exp(-e)}{2(\exp(e) - \exp(-e))}$$
$$s_1^i = 1 - s_{-1}^i \quad (3.39)$$

Veri kümesi 10 eşit alt kümeye bölünerek 4'ü eğitim 6'sı test kümesi olarak kullanılmıştır. Tanımlanan üyelik işlevi kullanılarak elde edilen sonuçlar, Tablo 3.1'de verilmiştir. Tabloda Bölüm 2'de önerilen Genel Elipsoit biçimli Sınıflayıcılar (GES) ile, bulanıklaştırılan elipsoit biçimli sınıflayıcılar ise BES ile gösterilmiştir.

Tablo 3.1'den görüldüğü gibi, önerilen bulanık ve genel elipsoit biçimli sınıflayıcılar yakın başarımlar vermiştir. Bu da önerilen GES sınıflayıcıların, veriler bulanıklaştırılmadan bile gürbüz olduğunu göstermektedir.

Tablo 3.1 Önerilen sınıflayıcıların 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.

Zambak Çiçeği	1.sınıf		2.sınıf		3.sınıf	
	BES	GES	BES	GES	BES	GES
n=4, e=0.5						
Eğitim Doğruluğu [%]	100	100	100	98.33	100	100
Test Doğruluğu [%]	100	100	95.67	94.38	95.62	95.13
Süre [s]	0.02	0.02	0.11	0.12	0.14	0.28
n=2, e=0.05						
Eğitim Doğruluğu [%]	100	100	100	100	100	100
Test Doğruluğu [%]	100	100	96.62	95.62	96.50	95.50
Süre [s]	0.01	0.01	0.01	0.02	0.02	0.03

3.5.3. Kanser Veri Kümeleri ile Sınıflama Problemi

Bu uygulamada, DVM bilimsel yazınında oldukça fazla kullanılan kolon ve lösemi kanseri veri kümeleri ele alınmıştır [12, 32]. Kolon veri kümesi, 40 tümör ve 22 normal kolon dokusundan oluşmaktadır. Lösemi veri kümesi, lösemi kanserinin iki akut durumunun gen ifade profilini içererek burada akut lymphoblastic lösemi (kısaca ALL) ve akut myeloblastic lösemi (kısaca AML) ALL/AML veri kümesi olarak gösterilmiştir. Veri kümesinin ALL bölümü, B ve T hücreli iki örnek tipinden oluşmaktadır. AML bölümü kemik iliği örnekleri ve ikincil kana ayrılmaktadır. Burada ilk olarak, AML ve ALL örnekleri arasındaki ikili sınıflama problemi daha sonra her bir elemanın ikiye ayrılmasından dolayı üç sınıflı sınıflama problemi (B-hücresi, T-hücresi ve AML) oluşturulmuştur. Ele alınan veri kümelerinin karakteristiği Tablo 3.2’de verilmiştir. Gen seçimi için MATLAB PLS kütüphanesi kullanılmıştır [32]. Hem kolon hem de lösemi veri kümeleri için toplam 500 gen kullanılmıştır. Veri kümesi 10 eşit alt kümeye bölünmüş ve 4’ü eğitim 6’sı test kümesi olarak kullanılmıştır.

Tablo 3.2 Deneylerde kullanılan veri kümesinin özeti

Veri Kümesi	Örnek Ölçüsü	Genler	Sınıflar
ALL-AML-3	72	7129	3
ALL-AML -2	72	7129	2
Kolon	62	2000	2

Uygun bulanık üyelik işlevi seçimi önemlidir. Bu nedenle uygulamalarda farklı üyelik işlevleri ile ilgilenilmiştir. Burada üyelik işlev değerleri [50]’deki gibi tek bir değere atanmıştır. Birinci sınıfa ait olan veriler, bulanıklaştırma derecesi daha yüksek olan $s_1^i = 1$ üyelik değerine atanırken, diğer sınıfa ait olan veriler $s_{-1}^i = 0.4$ üyelik değerine atanmıştır. Önerilen

sınıflayıcılar, 350 özyineleme ile momentum terimli uyarlanır öğrenme oranlı eğim iniş algoritması ile eğitilmiştir. Girişlerin boyutu çok büyük olduğu için elipsin kovaryans matrisinin pozitif tanımlılığı olmadığı adımlar oluşmuştur. Bu nedenle kötü şartlanma sayısına bakılarak, katsayılar düzeltme önerilmiştir. Klasik DVM için [86]’daki MATLAB eniyileme kütüphanesi kullanılmıştır. Çoklu sınıflama için bire karşı bir yöntemi ile sonuçlar elde edilmiştir. DVM parametrelerinin diğer uygulamalardaki gibi en iyi test başarımını veren değerleri aranmıştır. Kolon veri kümesi için $C = 1000$ ve RTİ çekirdek parametresi 6.4, lösemi veri kümesi için ise $C = 10^6$ ve çekirdek parametresi 50 olarak seçilmiştir. Elde edilen sonuçlar Tablo 3.3’te verilmiştir. Tabloda iki ve üç sınıflı veri kümesi sırasıyla ALL-AML-2 ve ALL-AML-3 olarak belirtilmiştir. Görüldüğü gibi bulanık sınıflayıcılar küçük bir farkla, önerilen elipsoit biçimli sınıflayıcıları geçmiştir ve DVM sınıflayıcılar ile yakın sonuçlar elde edilmiştir.

Tablo 3.3 Kanseri veri kümeleri için önerilen sınıflayıcının ve DVM’nin doğruluklarının kıyaslanması.

Yöntem	Kolon		ALL-AML-2		ALL-AML-3	
	Eğitim Doğruluğu [%]	Test Doğruluğu [%]	Eğitim Doğruluğu [%]	Test Doğruluğu [%]	Eğitim Doğruluğu [%]	Test Doğruluğu [%]
DVM	100	80.27	100	82.88	100	97.61
GES	100	79.67	100	84.12	100	97.07
BES	100	80.23	100	85.02	100	97.57

4. RADYAL TABANLI İŞLEVLERİ KULLANARAK SINIFLAMA ve REGRESYON KESTİRİMİ

4.1. Problem Tanıtımı

Önceki bölümlerde küre ve elipsoit biçimli sınıflayıcılar, boyutu ve veri sayısı farklı olan birçok veri kümesinin sınıflanmasında başarılı bir şekilde kullanılmıştı. Ancak her veri, küre veya elipsoit biçimli yüzeyler ile ayrıştırılamaz. Böyle bir durumda, “Hem penaltı yöntemi kullanarak deneysel ve yapısal riski enazlayan, hem de karmaşık yüzeylerle ayrıştırma yapan yeni bir formülasyon önerilebilir mi?” sorusu akla gelmektedir. Bu sorunun yanıtını bulmak için, öncelikle karmaşık yüzeyi seçmek gerekir. DVM iyi bir başarımla verdiği göre, DVM’nin kullandığı yüzeyleri kullanmak ilk bakışta mantıklı gelebilir. Ancak bu durumda da ortaya çıkan problem, DVM’de yüksek boyutlu uzaya dönüşümün açıkça verilmeyip çekirdek yardımıyla ifade edilmesidir. Açıkça verilmeyen ϕ dönüşümünün çekirdek yardımıyla elde edilmesi olanaklıdır. Örneğin ikinci dereceden polinomsal çekirdek,

$$K(x, x^i) = (1 + x^T x^i)^2 \quad (4.1)$$

iki boyutlu giriş $x = [x_1, x_2]^T$ ve $x^i = [x_1^i, x_2^i]^T$ için

$$K(x, x^i) = 1 + x_1^2 (x_1^i)^2 + 2x_1 x_2 x_1^i x_2^i + x_2^2 (x_2^i)^2 + 2x_1 x_1^i + 2x_2 x_2^i \quad (4.2)$$

olarak ifade edilerek

$$\phi(x) = [1, x_1^2, \sqrt{2}x_1 x_2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2]^T \quad (4.3)$$

6 boyutlu bir dönüşüm elde edilebilir [45].

RTİ çekirdek için

$$K(x, x^i) = e^{-\gamma \|x - x^i\|^2} = e^{-\gamma (x - x^i)^T (x - x^i)} = e^{-\gamma x^T x + 2\gamma x^T x^i - \gamma (x^i)^T x^i} \quad (4.4)$$

$$K(x, x^i) = e^{\gamma x^T x - \gamma (x^i)^T x^i} \left(1 + \frac{2\gamma x^T x^i}{1!} + \frac{(2\gamma x^T x^i)^2}{2!} + \frac{(2\gamma x^T x^i)^3}{3!} + \dots \right) \quad (4.5)$$

$$K(x, x^i) = e^{\gamma x^T x - \gamma (x^i)^T x^i} \left(1.1 + \sqrt{\frac{2\gamma}{1!}} x \cdot \sqrt{\frac{2\gamma}{1!}} x^i + \sqrt{\frac{(2\gamma)^2}{2!}} x^2 \cdot \sqrt{\frac{(2\gamma)^2}{2!}} (x^i)^2 + \sqrt{\frac{(2\gamma)^3}{3!}} x^3 \cdot \sqrt{\frac{(2\gamma)^3}{3!}} (x^i)^3 + \dots \right) \quad (4.6)$$

$$K(x, x^i) = \varphi(x)^T \varphi(x^i) \quad (4.7)$$

olarak açılırsa,

$$\varphi(x) = e^{-\gamma x^2} \left[1, \sqrt{\frac{2\gamma}{1!}}x, \sqrt{\frac{(2\gamma)^2}{2!}}x^2, \sqrt{\frac{(2\gamma)^3}{3!}}x^3, \dots \right]^T \quad (4.8)$$

sonsuz boyutlu bir dönüşüm elde edilebilir. Bu durumda sonucu canlandırmak için basitçe, Bölüm 2’de önerilen küre veya elipsoit biçimli sınıflayıcılara benzer olarak, formülasyonda küre (veya elipsoit) yüzeyi yerine, sonsuz boyutlu (veya iyi bir başarımla için oldukça büyük boyutlu) bir yüzey yerleştirdiğimizi düşünürsek, hafıza yetmeyeceği için problemin çözülemeyeceği açıkça görülür.

Bu problemin çözümü için, akla ikinci olarak, “RTİA’ların veya bilinen bir gizli katmanlı YSA’ların yüzeyleri kullanılarak formülasyon oluşturulabilir mi? [111] Oluşturulabilirse yapısal hata nasıl bulunur?” soruları gelmektedir. Küre veya elipsoit sınıflayıcılarda, hacim enazlanarak bu yapılabiliriyordu. “RTİA veya YSA’ların tanımladığı yüzeyler için bu nasıl yapılabilir?” sorusunun bir yanıtı aşağıda verilmiştir.

Birinci durumda, $(x^1, y^1), \dots, (x^L, y^L)$, $x \in \mathfrak{R}^n$ girişleri için, sınıflama veya regresyon problemlerinde r gizli birimli, tek gizli katmanlı RTİA’nın çözümünün,

$$\ell(x) = \sum_{i=1}^r w_i \varphi_i(x) + b \quad (4.9)$$

biçiminde olduğunu düşünelim. Burada, $\varphi(x)$ (4.10)’da verilen gauss işlevini, $c \in \mathfrak{R}^n$ işlevin merkezlerini, $\sigma \in \mathfrak{R}$ işlevin genişliğini w gerçel ağırlık vektörünü ve b ise orijinden kayıklık oranını göstermektedir.

$$\varphi_i(x) = \exp \left(-\frac{\|x - c^i\|^2}{2\sigma_i^2} \right) \quad (4.10)$$

4.2. RTİ Dönüşümü Kullanarak Sınıflama

$x^i \in \mathfrak{R}^n$, $y^i \in \{-1, 1\}$ olmak üzere $(x^1, y^1), \dots, (x^L, y^L)$ eğitim örneği RTİ dönüşümü kullanılarak sınıflanırsa, karar sınırı

$$D = \left\{ x^i \in \mathfrak{R}^n \left| \sum_{j=1}^r w_j \left(e^{-\frac{[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b = 1, \quad j = 1, \dots, r, \quad i = 1, \dots, L \right. \right\} \quad (4.11)$$

olarak alınır ve veri etiketleri (4.12) ve (4.13)'deki gibi belirlenebilir,

$$\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \geq 1, \quad y^i = 1, \quad (4.12)$$

$$\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b < 1, \quad y^i = -1. \quad (4.13)$$

Bu durumda, problemin deneysel hatası Bölüm 2'de türetilen formülasyonlara benzer olarak

$$f(x) = \sum_{i=1}^L f \left(y^i \left(1 - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) \quad (4.14)$$

ile hesaplanır. Burada, f kayıp işlevi (4.15)'deki gibi seçilir.

$$\begin{aligned} f(\xi) &= \xi & \xi > 0 & & f'(\xi) &= 1 & \xi > 0 \\ f(\xi) &= 0 & \xi \leq 0 & & f'(\xi) &= 0 & \xi \leq 0. \end{aligned} \quad (4.15)$$

Problemin yapısal hatası

$$g(w) = \frac{1}{2} \|w\|^2 \quad (4.16)$$

olarak alınırsa, amaç ölçütü

$$F(x) = \sum_{i=1}^L f \left(y^i \left(1 - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) + \frac{A}{2} \|w\|^2 \quad (4.17)$$

olarak elde edilir. Kısıtsız eniyileme problemi, (4.18)'deki gibi önerilir,

$$\min_{w,b} \sum_{i=1}^L f \left(y^i \left(1 - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) + \frac{A}{2} \|w\|^2 \quad (4.18)$$

$A > 0$.

Burada; A penaltı parametresi

$$A = \frac{2}{L} \frac{\sum_{i=1}^L f \left(y^i \left(1 - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right)}{\|w\|^2} \quad (4.19)$$

veya

$$A = \frac{2}{L} \frac{\sum_{i=1}^L f \left(y^i \left(1 - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right)}{\|w\|^2} \cdot \exp(-hz) \quad (4.20)$$

ile hesaplanabilir. Burada z adım sayısını, h ise azalma hızını göstermektedir.

4.2.1. RTİ Dönüşümü Kullanarak Sınıflama İçin Çözüm Yöntemi

(4.18)'de önerilen eniyileme problemi, eğim iniş algoritmalarından birisi kullanılarak çözülmektedir. Bu algoritmalar ile çözüm yapabilmek için, önce amaç ölçütü F 'nin, (4.21) - (4.24)'deki gibi w_j , c_j , σ_j ve b değişkenlerine göre negatif gradyanları alınır. Eğer merkezler tüm veriler alınır ve σ sabit bir değer kabul edilirse, sadece w_j ve b değişkenlerine göre negatif gradyanları alınarak çözüm yapılabilir. Ancak bu durumda eğitim süresi bir önceki duruma göre fazla olur.

$$-\frac{\partial F}{\partial w_j} = \sum_{i=1}^L f' \left(y^i \left(1 - \left(\sum_{j=1}^L w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) y^i \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) - Aw \quad (4.21)$$

$$-\frac{\partial F}{\partial c_j} = \sum_{i=1}^L f' \left(y^i \left(1 - \left(\sum_{j=1}^L w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) y^i \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) \frac{[x^i - c_j]}{\sigma_j^2} \quad (4.22)$$

$$-\frac{\partial F}{\partial \sigma_j} = \sum_{i=1}^L f' \left(y^i \left(1 - \left(\sum_{j=1}^L w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) y^i \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) \frac{[x^i - c_j]^T [x^i - c_j]}{\sigma_j^3} \quad (4.23)$$

$$-\frac{\partial F}{\partial b} = \sum_{i=1}^L f' \left(y^i \left(1 - \left(\sum_{j=1}^L w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \right) y^i \quad (4.24)$$

Daha sonra kullanılan algoritma gereğince özyineli olarak güncellenir. Örneğin (4.25) ve (4.28), en dik eğim iniş algoritmasında kullanılan (k+1). adımındaki güncelleme işlemlerini göstermektedir

$$w_j(k+1) = w_j(k) - \eta \frac{\partial F}{\partial w_j} \Big|_{w_j=w_j(k)} , \quad (4.25)$$

$$b(k+1) = b(k) - \eta \frac{\partial F}{\partial b} \Big|_{b=b(k)} , \quad (4.26)$$

$$c_j(k+1) = c_j(k) - \eta \frac{\partial F}{\partial c_j} \Big|_{c_j=c_j(k)} , \quad (4.27)$$

$$\sigma_j(k+1) = \sigma_j(k) - \eta \frac{\partial F}{\partial \sigma_j} \Big|_{\sigma=\sigma_j(k)} . \quad (4.28)$$

Sonuç olarak, (4.29)'da verilen sınıflayıcı yardımıyla sınıflanır

$$\ell(x) = \text{sgn} \left(-1 + \sum_{i=1}^L \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) . \quad (4.29)$$

4.3. RTİ Dönüşümü Kullanarak Regresyon Kestirimi

Regresyon problemlerinde, $x^i \in \mathcal{R}^n$ girişleri ve $y^i \in \mathcal{R}$ çıkışları göstermek üzere bir $(x^1, y^1), \dots, (x^L, y^L)$ eğitim kümesi verildiğinde, veriler $\varphi(\cdot): \mathcal{R}^n \rightarrow \mathcal{R}^r$ ile yüksek boyutlu uzaya dönüştürülür. Bu uzayda verilerin, Vapnik'in ε -duyarsız kayıp işlevine göre sapmalarını sınırlayarak,

$$|y - \ell(x)| = \begin{cases} 0 & |y - \ell(x)| \leq \varepsilon \\ |y - \ell(x)| - \varepsilon & \text{diğerleri} \end{cases} \quad (4.30)$$

(4.9) ilişkisi oluşturulur.

Regresyon problemi, Bölüm 2'deki gibi kısıtlı eniyileme problemi olarak düşünülüp, deneysel hata

$$f(x) = \sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) \quad (4.31)$$

ve yapısal hata

$$g(w) = \frac{1}{2} \|w\|^2 \quad (4.32)$$

olarak alınırsa, kısıtsız eniyileme probleminin amaç ölçütü

$$F(x) = \sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) + \frac{A}{2} \|w\|^2 \quad (4.33)$$

olarak elde edilir. Sonuç kısıtsız eniyileme problemi (4.34)'deki gibi önerilmiştir,

$$\min_{w,b} \sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) + \frac{A}{2} \|w\|^2 \quad (4.34)$$

$$A > 0$$

Burada; A ödünleşim parametresi

$$A = \frac{2}{L} \frac{\sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right)}{\|w\|^2} \quad (4.35)$$

veya

$$A = \frac{2}{L} \frac{\sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right)}{\|w\|^2} \exp(-hz) \quad (4.36)$$

uyarınca hesaplanır.

4.3.1. RTİ Dönüşümü Kullanarak Regresyon Kestirimi İçin Çözüm Yöntemi

(4.34)'de önerilen eniyileme probleminin, eğim iniş algortimaları kullanılarak çözülebilmesi için öncelikle amaç ölçütü F 'nin, (4.37) - (4.40)'daki gibi w_j , c_j , σ_j ve b değişkenlerine göre negatif gradyanları alınır.

$$-\frac{\partial F}{\partial w_j} = \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} - A.w \right) \quad (4.37)$$

$$\begin{aligned}
-\frac{\partial F}{\partial c_j} = & \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \\
& \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) \frac{[x^i - c_j]^T}{\sigma_j^2}
\end{aligned} \quad (4.38)$$

$$\begin{aligned}
-\frac{\partial F}{\partial \sigma_j} = & \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right) \\
& \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) \frac{[x^i - c_j]^T [x^i - c_j]}{\sigma_j^3}
\end{aligned} \quad (4.39)$$

$$\begin{aligned}
-\frac{\partial F}{\partial b} = & \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \left(e^{\frac{-[x^i - c_j]^T [x^i - c_j]}{2\sigma_j^2}} \right) + b \right) \right)
\end{aligned} \quad (4.40)$$

Daha sonra (4.41) - (4.44) bağıntılarındaki gibi özyineli olarak güncellenir.

$$w_j(k+1) = w_j(k) - \eta \frac{\partial F}{\partial w_j} \Big|_{w_j=w_j(k)} \quad (4.41)$$

$$b(k+1) = b(k) - \eta \frac{\partial F}{\partial b} \Big|_{b=b(k)} \quad (4.42)$$

$$c_j(k+1) = c_j(k) - \eta \frac{\partial F}{\partial c_j} \Big|_{c_j=c_j(k)} \quad (4.43)$$

$$\sigma_j(k+1) = \sigma_j(k) - \eta \frac{\partial F}{\partial \sigma_j} \Big|_{\sigma_j=\sigma_j(k)} \quad (4.44)$$

4.4. Uygulamalar

4.4.1. WGK, CKH ve İyonosfer Veri Kümeleri ile Sınıflama Problemi

Bu uygulamada, RTİA yüzeyi kullanarak oluşturulan sınıflayıcıları; DVM sınıflayıcılara göre, eğitim ve test başarımları ile süre açısından kıyaslamak için, UCI makine öğrenme arşivinden: Cleveland kalp hastalıkları (CKH), Wisconsin göğüs kanseri (WGK) ve İyonosfer verileri seçilmiştir. Veri kümeleri 10 eşit alt kümeye bölünmüş ve ayrılan 4 alt küme eğitim kümesi, geriye kalan 6 alt küme test kümesi olarak kullanılmıştır.

Önerilen algoritma iki farklı şekilde çalıştırılarak sonuçlarına bakılmıştır. Birincisinde K-ortalamlar öbekleme algoritması ile eğitim veri sayısı kadar merkez elde edilmiş ve eğitim boyunca güncellenmemiştir. RTİ çekirdeğin σ parametresi ise $[0.01:0.01:4]$ dizisi içerisinde test kümesi üzerinde en iyi başarımları veren değer seçilmiş ve eğitim boyunca güncellenmemiştir. İkincisinde ise merkez sayısı önceden belirlenerek K-ortalamlar öbekleme algoritması ile σ parametresinin değeri ise $1/(2r)^{(1/n-2)}/\sqrt{2\log(2)}$ [83] ile başlatılmış ve eğitim boyunca güncellenmiştir. w değerleri ise iki biçimde de her kat için rasgele ve b değerleri ise 1 ile başlatılmıştır. Algoritma farklı adım sayısı için momentum terimli ve uyarlanabilir öğrenme oranı içeren eğim iniş yöntemiyle eğitilmiştir. Penaltı parametresi iyonosfer veri kümesi için (4.35)'deki gibi, diğer veri kümeleri için ise (4.36)'daki gibi üstel olarak güncellenmiştir. Azalma hızını belirten h sabiti 3 seçilmiştir.

Merkez ve değişinti parametrelerinin güncellenmediği durumda, program 10 defa çalıştırılarak eğitim süresi ile eğitim ve test verileri için doğru sınıflanmış noktalarının toplam yüzdesi, ortalama doğruluk olarak Tablo 4.1'de verilmiştir. Tüm parametrelerin güncellendiği durumda da program 10 defa çalıştırılmış ancak değişinti değeri uygun bir değerle başlatılmadığı için bazı çalıştırmalarda çok uygun olmayan sonuçlar elde edilmiştir. Bu doğal bir sonuç olduğu için tüm parametrelerin güncellendiği durumda sadece uygun sonuçların toplam yüzdesi, ortalama doğruluk olarak tabloda verilmiştir. DVM'nin sonuçları ise Bölüm 2'den tabloya aktarılmıştır. Sonuçlardan görüldüğü gibi merkez ve değişinti parametre değerlerinin başlangıçta uygun seçilerek daha sonra güncellenmediği durumda, önerilen formülasyon hem hız hem de başarımları olarak DVM'yi geçmiştir. Aynı durumda K-ortalamlar öbekleme algoritması ile başlatılmayıp tüm veriler merkez olarak kullanılsa bile benzer sonuçlar alınabilmesi dikkate değer bir noktadır. Diğer durumlarda ise, merkez ve değişinti parametresinin değeri değişimi her adımda dönüşüm işlevinin hesaplanmasını da gerektiği için eğitim süresi artmıştır. Diğer taraftan birim sayısı r azalmasına rağmen iyi sonuçlar elde edilmiştir.

Tablo 4.1 RTİA yüzeyi kullanarak oluşturulan sınıflayıcıların ve DVM sınıflayıcıların denektaşı veri kümeleri üzerindeki başarımları.

Veri Kümesi (1. ve 2. sınıftaki veri sayıları, n)	WGK (41,69,32)		CKH (83,214,13)		İyonosfer (126,225,34)	
Önerilen RTİA						
Eğitim Doğruluğu [%]	100		100		98.57	
Test Doğruluğu [%]	69.23		80.00		93.38	
Süre [s]	0.11		0.12		0.17	
σ	0.95		2.09		2	
r	44		119		140	
Adım Sayısı	350		350		350	
Önerilen RTİA						
Eğitim Doğruluğu [%]	100	64.36	100	84.87	97.42	94.28
Test Doğruluğu [%]	67.69	62.53	77.57	82.48	93.80	93.90
Süre [s]	2.33	2.28	13.04	2.55	37.73	7.01
r	44	15	119	15	140	15
Adım Sayısı	350	550	350	550	350	550
DVM						
Eğitim Doğruluğu [%]	97.73		99.15		100	
Test Doğruluğu [%]	64.15		77.97		93.33	
Süre [s]	0.64		2.22		5.04	
σ	0.0125		0.4		0.2	
C	1000		1000		1000	

4.4.2. İki Spiralli ve Dört Spiralli Veri Kümeleri ile Sınıflama Problemi

Bu uygulamada, elde edilen sonuçları görsel olarak göstermek için 2 boyutlu 1000 veriden oluşan dört spiralli veri kümesi ile 2 boyutlu 2000 veriden oluşan iki spiralli veri kümesi kullanılmıştır. Veri kümesinin yarısı eğitim, diğeri ise test etmek için ayrılmıştır. Önerilen algoritma, karesel programlama ile eğitilen DVM ve RTİA ağı ile kıyaslanmıştır.

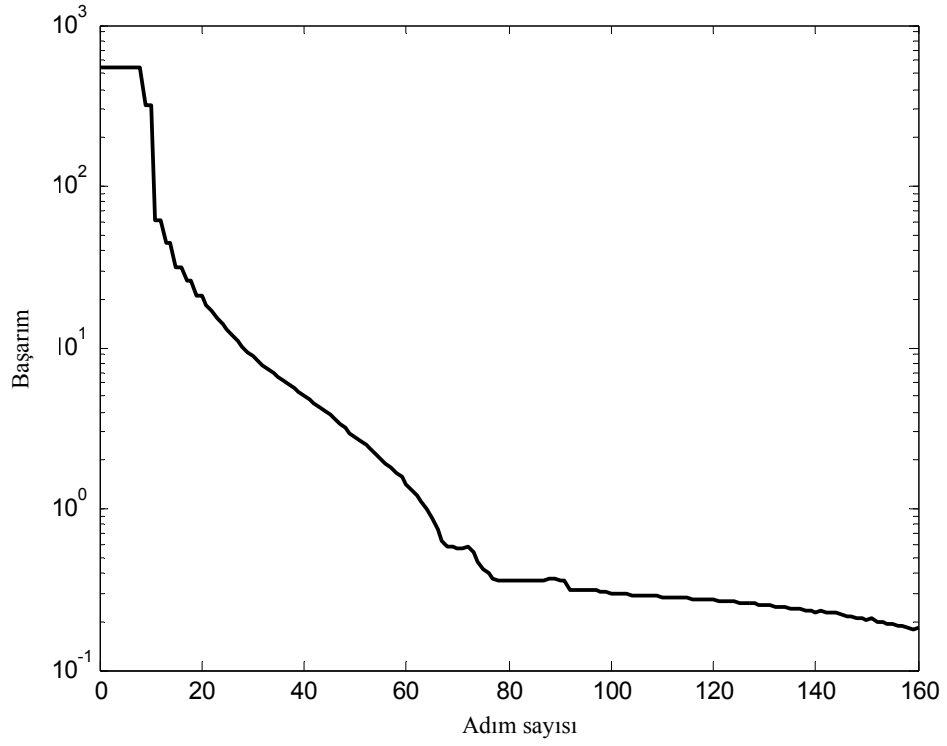
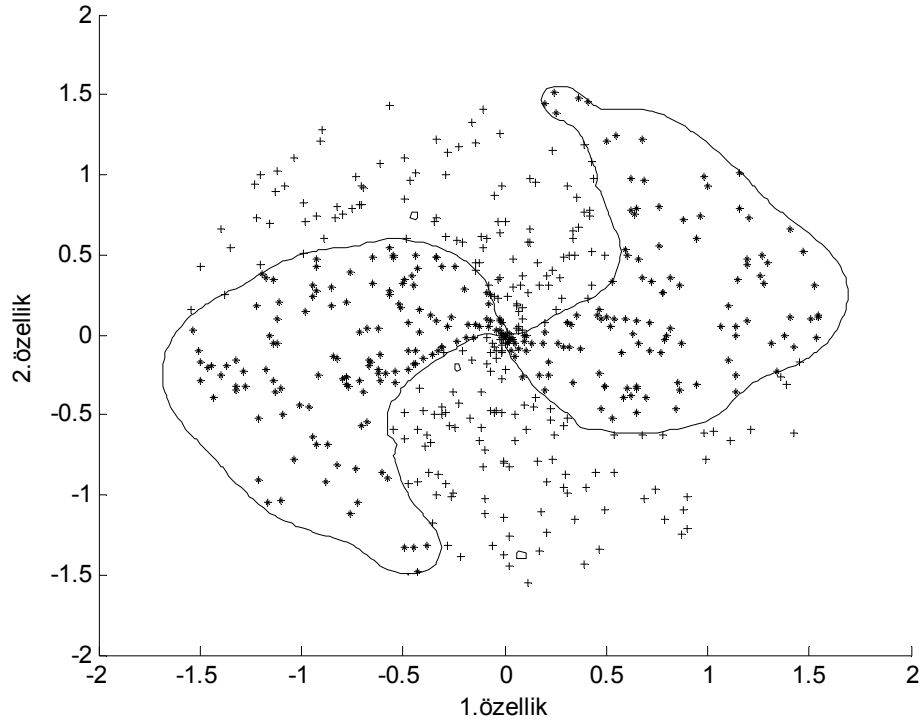
DVM için [41]'deki MATLAB kütüphanesi kullanılmıştır. RTİ çekirdekli DVM'nin C parametresinin en uygun değeri [1 10 100 1000 10000] dizisi içerisinde, σ parametresinin ise [0.01:0.01:3] dizisi içerisinde en iyi test başarımı veren değer olarak belirlenmiştir. En iyi test başarımını veren parametre değerleri içerisinde pay ve eğitim başarımlarına da bakılarak uygun bir değer seçilmiştir. Önerilen algoritma ise önceki örnekteki gibi iki farklı şekilde çalıştırılmıştır. RTİA benzetimleri için MATLAB'ta bir program oluşturulmuştur. Böylece önerilen algoritma ile aynı başlangıç şartları kullanılarak eğitim süresi ve başarımlarının uyumlu olmasını sağlanmıştır.

Program 10 defa çalıştırılarak eğitim süresi ile eğitim ve test verileri için doğru sınıflanmış noktalarının toplam yüzdesi, ortalama doğruluk olarak Tablo 4.2'de verilmiştir.

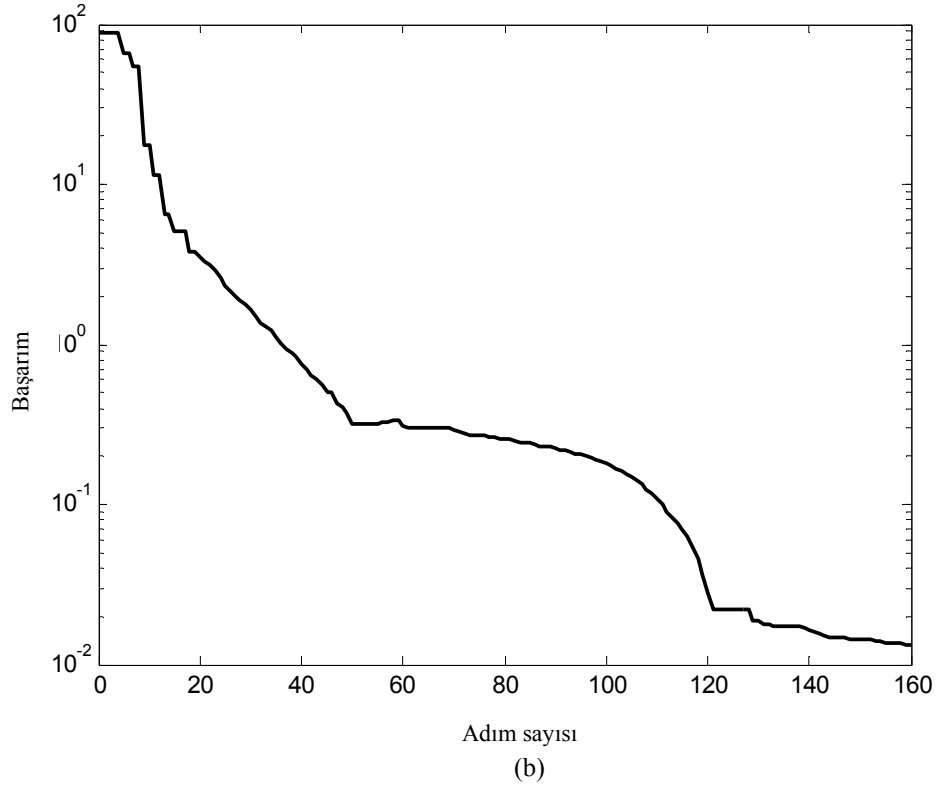
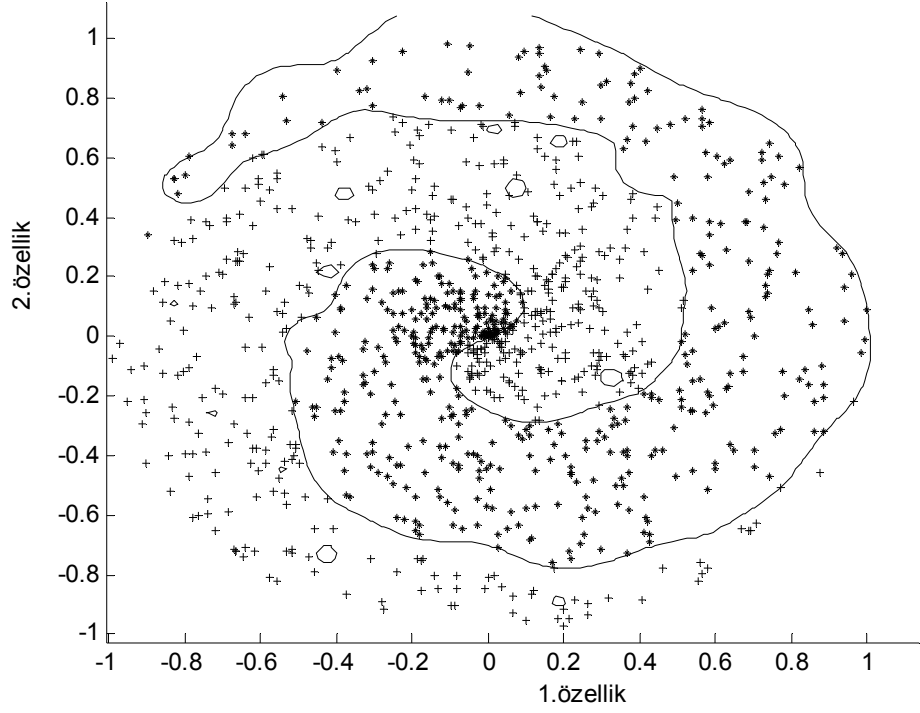
Önerilen sınıflayıcının eğitim başarımı ve karar yüzeyleri, eğitim örneklerini içererek Şekil 4.1’de DVM’nin sonuçları ise destek vektörleri de belirterek Şekil 4.2’de verilmiştir. Görüldüğü önerilen formülasyonun da karar yüzeyleri DVM’nin ki kadar düzgündür. Düzgünlükte değişinti parametresinin etkin olduğu bilinen bir gerçektir. Eğer güncellenmez ise uygun değerinin bulunması DVM’deki gibi önemli bir problem olur. Belirgin olarak küçük değerlerinde karmaşık yüzeyler ile sonuçlanabilir. Başarım açısından ise önerilen formülasyon RTİA’yı geçmiştir, DVM ile de benzer sonuçlar elde edilmiştir.

Tablo 4.2 RTİA yüzeyi kullanarak oluşturulan sınıflayıcıların, DVM sınıflayıcıların ve RTİA sınıflayıcıların denektaşı veri kümeleri üzerindeki başarımları.

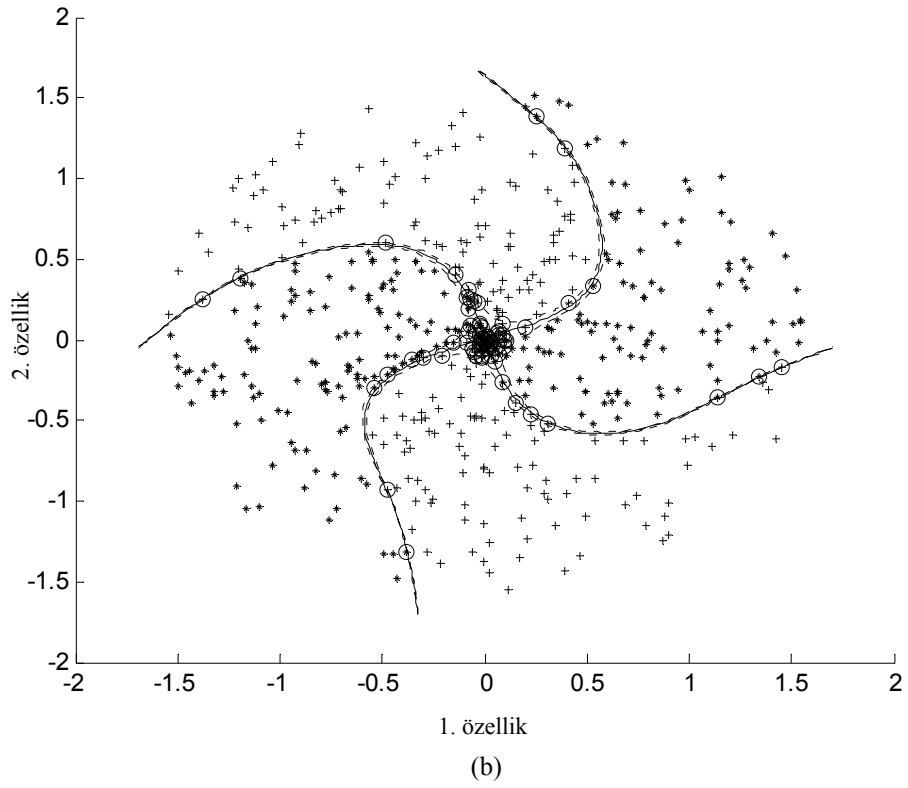
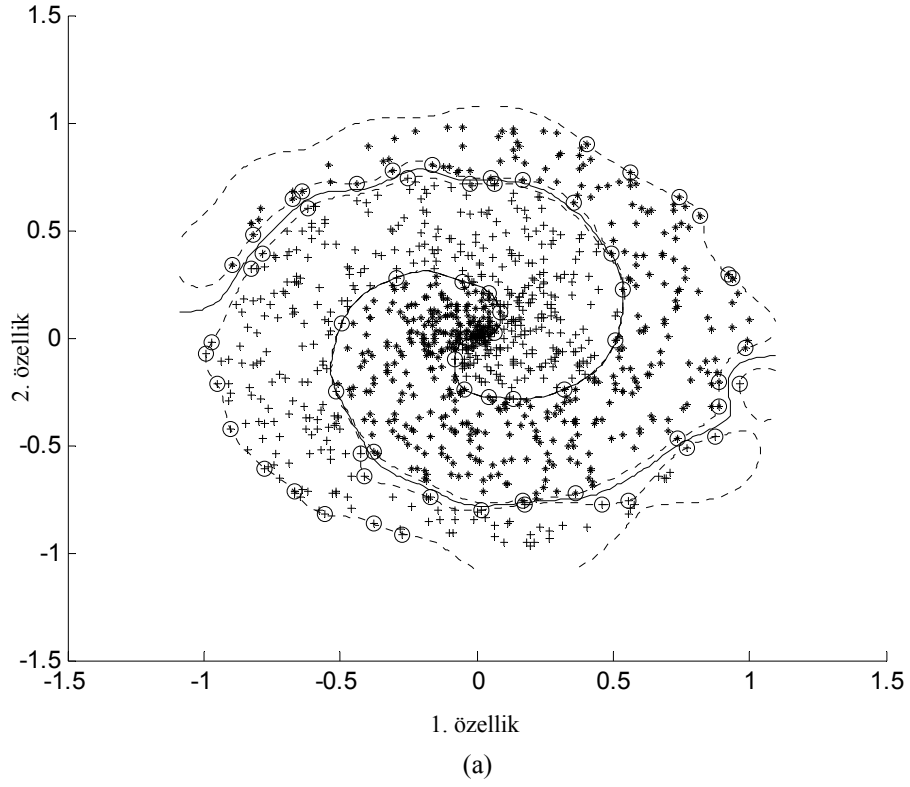
	İki Spiral (2x2000)			Dört Spiral (2x1000)		
Önerilen RTİA						
Eğitim Doğruluğu [%]	99.70			98.20		
Test Doğruluğu [%]	96.40			94.40		
Süre [s]	14.29			3.25		
σ	0.11			0.2		
Adım sayısı	1000			1000		
Önerilen RTİA						
Eğitim Doğruluğu [%]	99.70	99.60	98.40	97.40	98.60	97.40
Test Doğruluğu [%]	96.00	95.60	96.10	93.80	95.00	94.80
Süre [s]	29.08	17.60	10.06	14.11	6.16	4.18
r	170	95	58	154	76	57
Adım sayısı	160	160	160	160	160	160
DVM						
Eğitim Doğruluğu [%]	97.12			98.45		
Test Doğruluğu [%]	96.51			97.32		
Süre [s]	1803			133.77		
C	1000			1000		
σ	0.193			0.333		
RTİA						
Eğitim Doğruluğu [%]	93.90	91.01	91.80	95.80	93.40	94.60
Test Doğruluğu [%]	91.00	90.45	91.50	90.80	90.20	91.40
Süre [s]	30.07	16.78	9.56	14.12	6.92	4.87
r	170	95	58	154	76	57
Adım sayısı	160	160	160	160	160	160



(a)



Şekil 4.1 (a) Dört spiralli veri kümesi ve (b) iki spiralli veri kümesi için $r=57$ ve 58 ile RTİA yüzeyi kullanarak oluşturulan sınıflayıcıların karar yüzeyleri ve eğitim başarımı. Eğitim noktaları * (bir sınıf) ve + (diğer sınıf) ile gösterilmiştir.



Şekil 4.2 (a) Dört spiralli veri kümesi ve (b) iki spiralli veri kümesi için genel DVM sınıflayıcıların karar yüzeyleri. Eğitim noktaları * (bir sınıf) ve + (diğer sınıf) ile gösterilmiştir. Elde edilen destek vektörler ise daire içerisine alınmıştır.

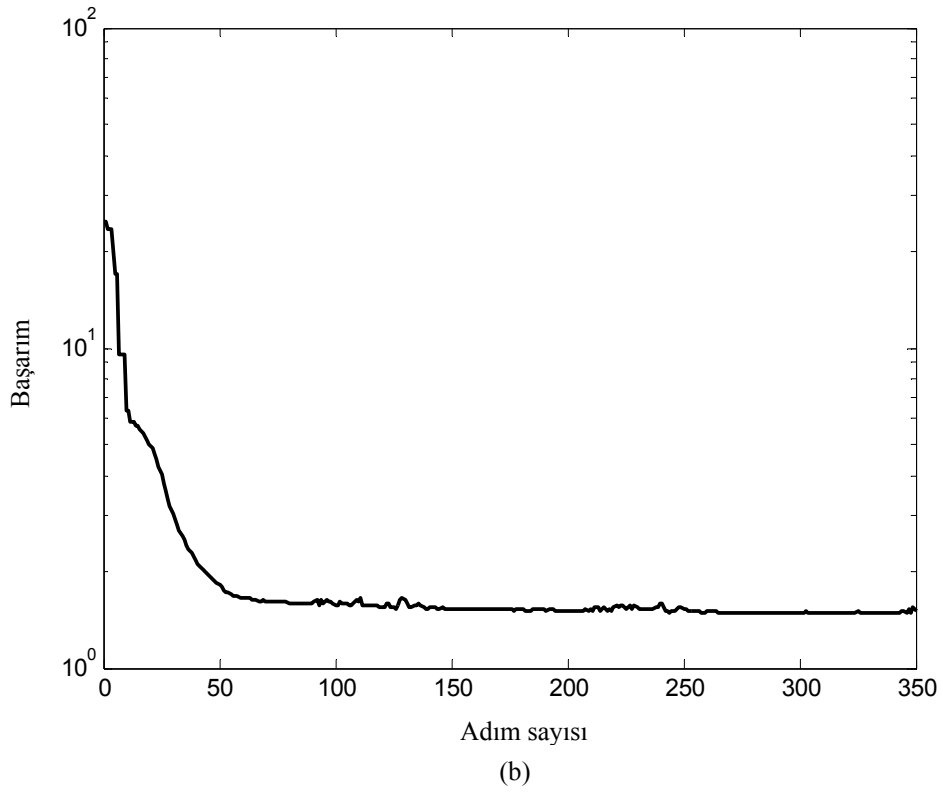
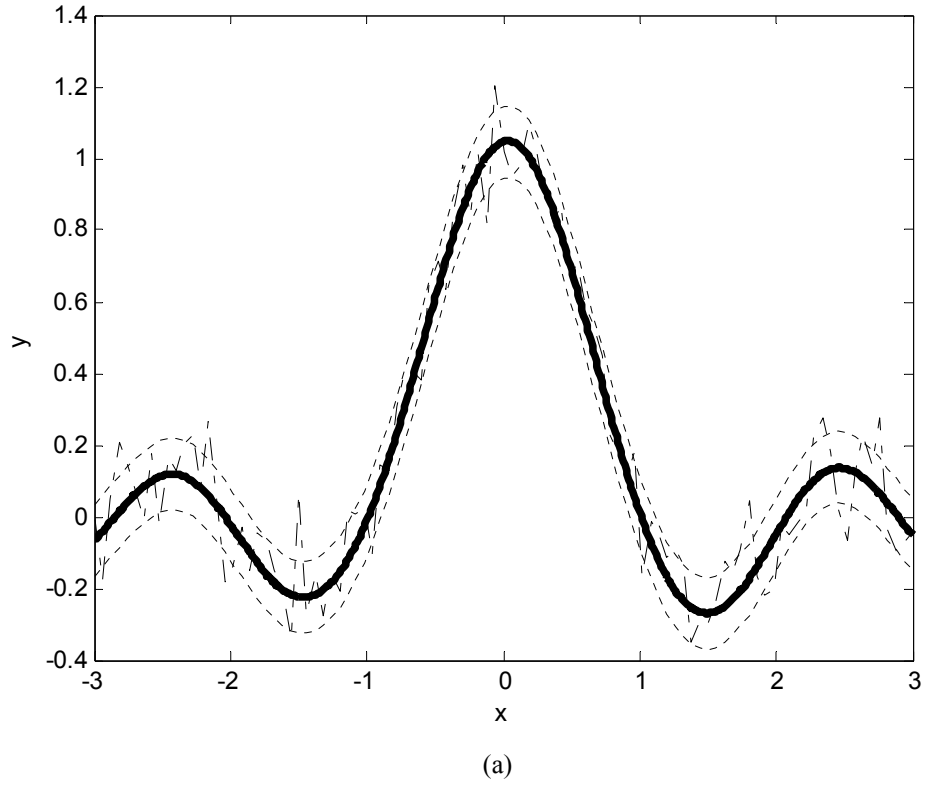
4.4.3. Sinc İşlevi ile Regresyon Kestirimi

Bu uygulamada RTİA yüzeyi kullanarak, oluşturulan regresyon kestirimi formülasyonunun başarımını incelemek için, genellikle bilimsel yazındaki öğrenme algoritmalarının geçerliliğini test etmek için kullanılan

$$y^i = \sin(\pi x^i) / (\pi x^i) + 0.1 n^i, \quad (4.45)$$

doğrusal olmayan gürültülü *sinc* işlevi ele alınmıştır. Burada, $x^i \sim [-3, 3]$ aralığında düzgün bir dağılıma sahip skaler bir girişi, $y^i \in \mathfrak{R}$ çıkış vektörünü, $n^i \sim N(0, 1)$ sıfır ortalamalı ve birim değişimli normal dağılıma sahip gürültü işaretini göstermektedir. Rasgele değişken x^i , n^i 'den bağımsızdır. Eğitmek amacıyla 101 veri ve test etmek amacıyla 600 veri kullanılmıştır.

$\varepsilon=0.1$ ve $r=20$ için algoritma 10 defa çalıştırılmıştır. Eğitim 350 adım için 0.52 işlemci hesaplama süresinde sonuçlanmıştır. Şekil 4.3'de elde edilen uygun sonuçlardan bir tanesi verilmiştir. Verilen şekiller için eğitim ve test hatası, Ortalanmış Hatanın Karelerinin Karekökü (OHKK) olarak sırasıyla 0.038 ve 0.101 elde edilmiştir. Görüldüğü gibi gürültüler öğrenilmeyerek oldukça iyi bir başarıma ulaşılmıştır.



Şekil 4.3 (a) $r=20$ ile RTİA yüzeyi kullanarak oluşturulan regresyon kestirimi formülasyonunun yaklaşım sonuçları: ϵ -duyarsız bölgesi (-.-), gürültülü eğitim verileriyle elde edilen eğri (-.) ve test verilerini kullanarak hesaplanan eğri kalın düz çizgi ile gösterilmiştir, (b) eğitim başarımı.

5. ÇEKİRDEK İŞLEVLERİ KULLANARAK SINIFLAMA ve REGRESYON KESTİRİMİ

5.1. Doğrusal Olarak Ayırıştırma Yapan Sınıflayıcı İçin Önerilen Formülasyon

DVM'lerin, gerçek yaşam uygulamalarında bazı hesapsal problemleri bulunmaktadır. Bunlardan en önemlisi, veri sayısı çok fazla olan problemlerde hafızanın büyümesinden dolayı çözümde karesel programlamanın kullanılamamasıdır. Bu engeli aşmak için farklı formülasyonlar ve çözüm algoritmaları geliştirilmesi için çalışılmaktadır [19, 51, 80, 88].

Bölüm 4'te karmaşık yüzeylerin elde edilmesinde kullanılan ϕ dönüşümü, bir çekirdek ile tanımlanabilir. Bu bölümde, Bölüm 4'teki yöntem ve formülasyonlar çekirdek temel alınarak yeniden türetilmiştir ve eğim iniş algoritması uygulanmıştır. Eğim temelli algoritmaların bilimsel yazında çok girişli, büyük veri kümelerine başarılı bir şekilde uygulanarak dayanımlı ve etkili olduğunun gösterilmesi [45], sınıflama ve regresyon kestirimi için bu algoritmalarla eğitilen eniyileme problemleri önermenin iyi bir fikir olacağını güçlendirmiştir. DVM sınıflayıcılarının çözümü için, eğim iniş yöntemlerinin uygulanması yeni değildir. Ancak şimdiye kadar yapılan çalışmalar, ikincil programlama temeline dayanır [22, 34]. Yoğunluk kestirimi problemleri için en küçük hacimli daire problemlerinin parametre hesaplanmasında ve sınır tanımlamalarında; eğim iniş yöntemlerini kullanmayı başlatanlardan biri olan Chappelle [14], bu tez çalışması ile eş zamanlı olarak ilk defa [15] çalışmasında birincil form kullanarak, DVM sınıflayıcıların eğitimi için formülasyonlar önermiş ve çözümler elde etmiştir. [15]'den ve bu tez sonuçlarının kısmen yayınlandığı [100]'den farklı olarak, seçimi zor ve değeri önemli olan penaltı parametresi her özyinelemede güncellenmiştir. Böylece DVM'nin bir zorluğundan kurtulmak amaçlanmıştır. Ayrıca ilk defa, regresyon kestirimi ve çok sınıflı sınıflama problemleri için eğim iniş yöntemlerinin uygulanabileceği formülasyonlar geliştirilmiştir.

Bu bölümde, önce çok boyutlu uzayda doğrusal ayırıştırma formülasyonu çıkarılmıştır. Daha sonra çekirdek kullanımına imkan veren temsilci teoreminden faydalanılarak, doğrusal olmayan formülasyonlar türetilmiştir.

$x \in \mathcal{R}^n$ ve $y \in \{-1, 1\}$ olmak üzere $(x^1, y^1), \dots, (x^L, y^L)$ eğitim örneklerinin L örneği için, çok boyutlu düzlem biçimindeki karar yüzeyi

$$D = \{x \in \mathcal{R}^n \mid w^T x^i + b = 0, \quad i = 1, \dots, L\} \quad (5.1)$$

ile verilirse, verilerin hangi sınıfa atandığı

$$w^T x^i + b \geq 1 \quad y^i = 1 \quad (5.2)$$

$$w^T x^i + b \leq -1 \quad y^i = -1 \quad (5.3)$$

kullanılarak bulunur. Bu durumda problemin sınıflama hatası, küre ve elipsoit biçimli sınıflayıcı formülasyonunda önerilene benzer olarak, (5.4)'deki gibi hesaplanır

$$f(x) = \sum_{i=1}^L f(1 - y^i (w^T x^i + b)). \quad (5.4)$$

Burada; f ve onun türevi,

$$\begin{aligned} f(\xi) &= \xi \quad \xi > 0 & f'(\xi) &= 1 \quad \xi > 0 \\ f(\xi) &= 0 \quad \xi \leq 0 & f'(\xi) &= 0 \quad \xi \leq 0. \end{aligned} \quad (5.5)$$

olarak seçilir.

Yapısal hata, DVM formülasyonundan (5.6)'da ki gibi $\|w\|$ enazlanarak elde edilir,

$$g(w) = \frac{1}{2} \|w\|^2 = \frac{1}{2} w^T w, \quad (5.6)$$

Kısıtlı eniyileme problemi düzenlenmiştir kısıtsız eniyileme problemine

$$F(x) = \sum_{i=1}^L f(1 - y^i (w^T x^i + b)) + \frac{A}{2} w^T w \quad (5.7)$$

ile dönüştürülürse, amaç ölçütü hem formülasyona dayanımlılık kazandıran deneysel hatayı hem de yapısal hatayı içerir. Sonuç olarak (5.7)'deki amaç ölçütü ile

$$\min_{w,b} \sum_{i=1}^L f(1 - y^i (w^T x^i + b)) + \frac{A}{2} w^T w \quad (5.8)$$

$A > 0$

eniyileme problemi olarak önerilir. Burada; A dayanımlılık (doğru sınıflama) ve hacim arasında ödünleşim sağlayan penaltı parametresini göstermektedir. Bölüm 2'de önerildiği gibi

$$A = \frac{2 \sum_{i=1}^L f(1 - y^i (w^T x^i + b))}{w^T w} \quad (5.9)$$

her özyinelemede güncellenmektedir.

5.1.1. Önerilen Doğrusal Sınıflayıcı Formülasyonunun Çözüm Yöntemi

(5.8) ile verilen eniyileme problemi, önerilen yöntemde herhangi bir eğim iniş algoritmasıyla çözülmektedir. Çözüm için öncelikle amaç ölçütü F 'nin, (5.10) ve (5.11)'deki gibi w ve b değişkenlerine göre negatif gradyenleri alınır

$$-\frac{\partial F}{\partial b} = \sum_{i=1}^L y^i f'(1 - y^i (w^T x^i + b)), \quad (5.10)$$

$$-\frac{\partial F}{\partial w} = \sum_{i=1}^L y^i x^i f'(1 - y^i (w^T x^i + b)) - Aw, \quad (5.11)$$

daha sonra kullanılacak eğitim algoritmasına uygun olarak özyineli olarak güncellenir. Örneğin en dik iniş algoritması ile özyineleme işlemi (5.12) ve (5.13)'deki gibi yapılmaktadır,

$$w(k+1) = w(k) - \eta \frac{\partial F}{\partial w}, \quad (5.12)$$

$$b(k+1) = b(k) - \eta \frac{\partial F}{\partial b}. \quad (5.13)$$

Burada, η öğrenme oranıdır. Bölüm 2'de belirtildiği gibi, eğer momentum terimli uyarlanır öğrenme oranlı eğim iniş algoritması kullanılırsa veya uyarlanır öğrenme oranlı eğim iniş kullanılırsa, bu parametrenin seçimi problemi ortadan kalkmış olur.

Özyineleme sonucunda elde edilen w ve b değerleri kullanılarak, karar işlevi

$$\ell(x) = \text{sgn}(w^T x + b) \quad (5.14)$$

ile belirlenir.

5.2. Çekirdek Tabanlı Sınıflayıcılar İçin Önerilen Formülasyon

Giriş uzayında (5.15)'deki gibi doğrusal olmayan bir yüzey ile ayrıştırma yapılırsa,

$$D = \left\{ x \in \mathbb{R}^n \left| \sum_{j=1}^r w^T \phi_j(x) + b = 0, \quad j = 1, \dots, r \right. \right\}. \quad (5.15)$$

veriler (5.16) ve (5.17)'ye göre sınıflara ayrılır,

$$w^T \phi(x) + b \geq 1 \quad y = 1, \quad (5.16)$$

$$w^T \phi(x) + b \leq -1 \quad y = -1. \quad (5.17)$$

Burada; $\{\phi_j(x)\}_{j=1}^r$ giriş uzayından r boyutlu uzaya doğrusal olmayan dönüşümü, w ayarlanabilir ağırlıkları ve b orijinden kayıklık oranını göstermektedir.

Önerilen formülasyon doğrusal olan durum için kolaylıkla uygulanabilir. Çünkü, küre ve elipsoit biçimli sınıflayıcı formülasyonlarından farklı değildir. Ancak doğrusal olmayan durumda çoğu zaman, $\phi(x)$ ayrıştırıcı yüzeyler bilinmeyip çekirdek işlevi şeklinde örtük olarak verildiği için önerilen formülasyon, çekirdek kullanılarak uygulanabilir duruma temsilci teoremi [37, 38, 90, 91, 102] ile getirilmesi gerekir.

Temsilci teoremi kullanılırsa, çekirdek yardımıyla ifade edilebilen karar işlevi (5.18) biçiminde gösterilebilir [30, 82, 90, 91]

$$\ell(x) = \sum_{i=1}^L v_i K(x, x^i) + b \quad (5.18)$$

Burada; $v \in \mathbb{R}^L$ ağırlık vektörünü ve K çekirdek matrisini göstermektedir.

Bu durumda, yeni eniyileme formülasyonu için deneysel hata

$$f(x) = \sum_{i=1}^L f(1 - y^i (v^T K(x, x^i) + b)) \quad (5.19)$$

ve yapısal hata

$$g(v) = v^T K v \quad (5.20)$$

ile verilirse, kısıtsız amaç ölçütü

$$F = \sum_{i=1}^L f(1 - y^i (v^T K(x, x^i) + b)) + \frac{A}{2} v^T K v \quad (5.21)$$

olarak elde edilir. Burada; f (5.5)'de verilen kayıp işlevini göstermektedir.

Sonuç eniyileme problemi

$$\min_{v, b} \sum_{i=1}^L f(1 - y^i (v^T K(x, x^i) + b)) + A \frac{1}{2} v^T K v \quad (5.22)$$

$A > 0$

ile verilir. Burada; A penaltı parametresi her bir özyineleme adımında

$$A = \frac{2 \sum_{i=1}^L f(1 - y^i (v^T K(x, x^i) + b))}{L v^T K v} \quad (5.23)$$

ile hesaplanmaktadır.

5.2.1. Çekirdek Tabanlı Sınıflayıcılar İçin Önerilen Formülasyonun Çözüm Yöntemi

(5.22)'de önerilen eniyileme problemi de, doğrusal formülasyona benzer olarak herhangi eğim tabanlı algoritma ile çözülmektedir. Bunun için ilk olarak amaç ölçütü F 'nin (5.24) ve (5.25)'deki gibi w ve b değişkenlerine göre negatif gradyenleri alınır.

$$-\frac{\partial F}{\partial b} = \sum_{i=1}^L y^i f'(1 - y^i (v^T K(x, x^i) + b)), \quad (5.24)$$

$$-\frac{\partial F}{\partial v} = \sum_{i=1}^L y^i K(x, x^i) f'(1 - y^i (v^T K(x, x^i) + b)) - A v K. \quad (5.25)$$

Daha sonra v ve b değişkenleri, eğitim algortimaları gereğince öz yineli olarak güncellenir. En dik iniş için örnek bir güncelleme, (5.26) ve (5.27)'de verilmiştir,

$$v(k+1) = v(k) - \eta \frac{\partial F}{\partial v}, \quad (5.26)$$

$$b(k+1) = b(k) - \eta \frac{\partial F}{\partial b}. \quad (5.27)$$

Özyineleme sonucunda elde edilen v ve b değerleri ile (5.28)'deki karar işlevi kullanılarak, verilerin ait olduğu sınıflar belirlenir:

$$\ell(x) = \text{sgn} \left(\sum_{i=1}^L v_i K(x, x^i) + b \right). \quad (5.28)$$

5.3. m-Sınıflı Sınıflama İçin Önerilen Çekirdek Tabanlı Formülasyon

Bu bölümde ikili sınıflandırma için önerilen formülasyon, karar yüzeyi

$$y_m^i (v_m^T K(x, x^i) + b_m) \geq 1 \quad (5.29)$$

alınarak m -sınıflı sınıflama problemlerine genişletilmiştir.

m -sınıflı sınıflama için formülasyon, ikili sınıflamaya benzer olarak

$$\min_{v,b} \sum_{m=1}^M \sum_{i=1}^L f(1 - y_m^i (v_m^T K(x, x^i) + b_m)) + \frac{A}{2} \sum_{m=1}^M v_m^T K v_m \quad (5.30)$$

$A > 0$

ile formüle edilir. Ödünleşimi sağlayan penaltı parametresi

$$A = \frac{2}{L} \sum_{m=1}^M \sum_{i=1}^L f(1 - y_m^i (v_m^T K(x, x^i) + b_m)) \bigg/ \sum_{m=1}^M v_m^T K v_m \quad (5.31)$$

veya

$$A = \frac{2}{L} \exp(-h z) \cdot \sum_{m=1}^M \sum_{i=1}^L f(1 - y_m^i (v_m^T K(x, x^i) + b_m)) \bigg/ \sum_{m=1}^M v_m^T K v_m \quad (5.32)$$

olarak hesaplanabilir. Burada; z adım sayısını, h ise azalma hızını belirleyen sabiti göstermektedir.

DVM'ler ile sınıflama yapıldığı zaman, pozitif ve negatif örnekler arasındaki ayırma payı içinde kalan eğitim örneklerinin sayısı enazlanır [13]. Bu, yapay değişkenlerin gösterge işlevinin yerine kullanıldığı için sadece yaklaşık olarak doğrudur. YSA'lar ise geriye yayılım algortimaları ile öğrenme görevinin (sınıflama veya regresyon kestirimi) ne olduğuna bakmadan, ikinci dereceden kayıp işlevini enazlarlar. Burada da eğitim iniş yöntemlerine benzer şekilde

deneysel hatanın karesi alınarak, ikinci dereceden eğim iniş yöntemleri uygulanabilir. Hem zaman hem de başarımlar açısından küçük farklılıkları olan çözümler elde edilebilir.

5.4. Regresyon Kestirimi İçin Önerilen Çekirdek Tabanlı Formülasyon

x giriş vektörü verilerek buna karşılık gelen, gürültü içeren gerçel y vektörü hesaplanmak istenirse, (5.33)'de verilen doğrusal olmayan regresyon modeli kullanılabilir

$$\ell(x) = \sum_{i=1}^L v_i K(x, x^i) + b. \quad (5.33)$$

Burada; $v \in \mathbb{R}^L$ ağırlık vektörünü ve K çekirdek matrisini göstermektedir.

Regresyon kestiriminde farklı kayıp işlevleri seçilebilir [22, 89], uygulamalarda (5.34) ile verilen ε -duyarsız kayıp işlevi kullanılmıştır,

$$f(y, \ell(x)) = |y - \ell(x)|_{\varepsilon} = \max(0, |y - \ell(x)| - \varepsilon). \quad (5.34)$$

ε -duyarsız kayıp işlevi için regresyon probleminin deneysel hatası

$$f(x) = \sum_{i=1}^L f\left(y^i - \left(\sum_{j=1}^L v_j K(x, x^j) + b\right)\right) - \varepsilon \quad (5.35)$$

ve yapısal hatası

$$g(v) = \frac{1}{2} v^T K v \quad (5.36)$$

olarak bulunur. f kayıp işlevi (5.5)'deki gibi hesaplanır.

Kısıtsız eniyileme problemi için amaç ölçütü ve eniyileme problemi, sırasıyla (5.37) ve (5.38) ile verilir.

$$F(x) = \sum_{i=1}^L f\left(y^i - \left(\sum_{j=1}^L v_j K(x, x^j) + b\right)\right) - \varepsilon + \frac{A}{2} v^T K v \quad (5.37)$$

$$\min_{v, b} \sum_{i=1}^L f\left(y^i - \left(\sum_{j=1}^L v_j K(x, x^j) + b\right)\right) - \varepsilon + \frac{A}{2} v^T K v \quad (5.38)$$

$A > 0$

Burada; A ödünleşimi sağlayan penaltı sabitini göstermektedir. A , (5.39)'daki gibi özyineli olarak hesaplanır:

$$A = \frac{2}{L} \frac{\sum_{i=1}^L f\left(y^i - \left(\sum_{j=1}^L v_j K(x, x^j) + b\right)\right) - \varepsilon}{v^T K v} \quad (5.39)$$

5.4.1. Regresyon Kestirimi İçin Önerilen Çekirdek Tabanlı Formülasyonun Çözüm Yöntemi

(5.38)'de önerilen eniyileme probleminin en dik iniş yöntemi kullanılarak çözülebilmesi için önce, amaç ölçütü F 'nin v ve b değişkenlerine göre negatif gradyenleri alınır.

$$-\frac{\partial F}{\partial v_j} = \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^L v_j K(x, x^i) + b \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^L v_j K(x, x^i) + b \right) \right) K(x, x^i) - A.vK(x, x^i) \quad (5.40)$$

$$-\frac{\partial F}{\partial b} = \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^L v_j K(x, x^i) + b \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^L v_j K(x, x^i) + b \right) \right) \quad (5.41)$$

Daha sonra, önerilen algoritmaya göre öz yineli olarak güncellenir. Örneğin en dik eğim iniş algoritması için güncelleme (5.42-5.43) bağıntıları yapılır

$$v_j(k+1) = v_j(k) - \eta \frac{\partial F}{\partial v_j} \Big|_{v_j=v_j(k)}, \quad (5.42)$$

$$b(k+1) = b(k) - \eta \frac{\partial F}{\partial b} \Big|_{b=b(k)}. \quad (5.43)$$

Bu bölümde benzer formülasyonların tekrarından kaçınmak için, sadece doğrusal olmayan regresyon kestirimi için formülasyonlar çıkartılmıştır. Doğrusal formülasyonun çok daha kolay bir şekilde elde edilebileceği, sınıflama bölümünden açıkça görülmektedir. Basit olarak çekirdek matrisi x vektörüyle yer değiştirir ve yapısal hata sadece ağırlıkların normundan oluşan bir terimle gösterilir.

5.5. Uygulamalar

5.5.1. CKH, BUPAKB ve İyonosfer Veri Kümeleri ile Sınıflama Problemi

Bu uygulamada, önerilen DVM sınıflayıcının hızını ve başarımını farklı sınıflayıcılar ile kıyaslamak için, UCI makine öğrenme arşivinden Cleveland kalp hastalıkları (CKH), İyonosfer ve BUPA karaciğer bozuklukları (BUPAKB) veri kümeleri kullanılmıştır. Ele alınan veriler

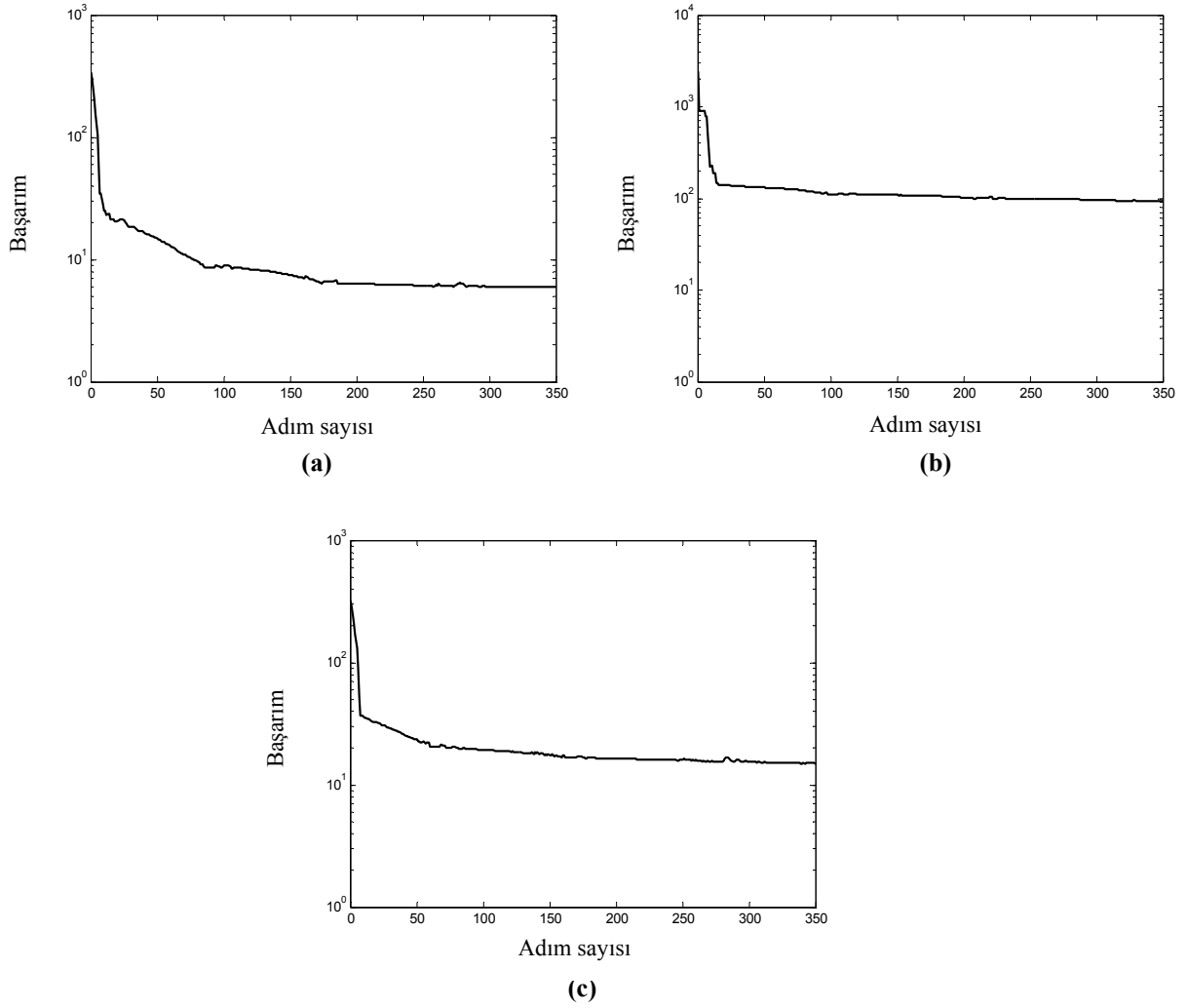
önce rasgele karıştırılmış daha sonra 10 eşit alt kümeye bölünerek 4'ü eğitim 6'sı test kümesi olarak kullanılmıştır.

Önerilen formülasyonda tüm veri kümeleri için RTİ, tanh ve polinomsal çekirdek kullanılmıştır. RTİ çekirdeğin σ parametresi, [0.01: 0.01: 3] dizisi içerisinde, polinomsal çekirdeğin derecesi [1:1:30] dizisi içerisinde ve tanh çekirdeğin parametresi ise [0.01: 0.01: 3] dizisi içerisinde en iyi test başarımı veren değer seçilmiştir. En iyi sonucu veren değer birden fazla olduğu zaman, her bir parametre için algoritma 10 defa çalıştırılmış hem ortalama eğitim de hem de ortalama test de en iyi başarımı veren bir değer kullanılmıştır. Önerilen sınıflayıcı, 350 adım için momentum terimli ve uyarlanır öğrenme oranı içeren eğim iniş algoritması ile eğitilmiştir. w değerleri algoritmanın her çalıştırılışında rasgele ve b değerleri ise 1 ile başlatılmıştır. Algoritma 10 defa çalıştırılarak hem eğitim hem de test verileri için doğru sınıflanmış noktaların toplam yüzdesi, ortalama doğruluk olarak Tablo 5.1'de ve 10 katın birinden alınan eğitim başarımları da Şekil 5.1'de verilmiştir.

Elde edilen sonuçlar, karesel programlama ile eğitilen DVM'nin yanı sıra bilimsel yazında özellikle zaman açısından en iyi sonuçları veren yöntemler olan NLDVM ve DVMLight ile de kıyaslanmıştır. DVM ve DVMLight için RTİ çekirdek kullanılarak elde edilen sonuçlar verilmiştir. DVMLight ve DVM'nin C parametresi [1: 10: 1000] dizisi içerisinde, σ parametresi ise [0.01:0.01:3] dizisi içerisinde en iyi test başarımı veren değer seçilmiştir. Sonuçlardan görüldüğü gibi RTİ çekirdek kullanılarak eğim iniş algoritması ile eğitilen önerilen sınıflayıcı test başarımları açısından, diğer yöntemleri geçmiştir. Süre açısından ise DVMLight ve DVM'den daha iyidir. Bununla birlikte NLDVM diğer yöntemlere göre daha hızlıdır.

Tablo 5.1 Önerilen çekirdek tabanlı sınıflayıcı ile NLDVM, DVMLight ve DVM'nin 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.

Veri Kümesi L × n (veri sayısı × boyut)	İyonosfer 351 × 34	BUPAKB 345 × 6	CKH 297 × 13
RTİ Çekirdekli NLDVM (ao=1) Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s]	95.78 96.09 0.99	80.72 81.74 0.32	73.69 71.01 0.34
DVMLight Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] σ C	100 94.28 0.22 1 1000	100 65.04 0.13 0.4 1000	100 82.79 0.74 2 1000
DVM Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] σ C	100 93.33 5.04 0.2 1000	99.27 57.76 3.28 1.04 1000	99.15 77.97 2.22 0.4 1000
Önerilen Formülasyon, RTİ çekirdek Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] σ	99.21 94.71 0.21 2.23	75.52 70.35 0.22 2.32	98.06 82.87 0.26 2.39
Önerilen Formülasyon, tanh çekirdek Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] σ	95.78 91.09 0.24 0.03	79.71 64.07 0.27 0.95	76.27 72.85 0.27 0.01
Önerilen Formülasyon, polinomsal çekirdek Eğitim Doğruluğu [%] Test Doğruluğu [%] Süre [s] Derece	97.85 94.28 0.27 2	71.84 68.84 0.21 1	92.41 78.31 0.28 1



Şekil 5.1 (a) İyonosfer, (b) BUPAKB ve (c) CKH veri kümeleri üzerinde, önerilen yöntemin 10 katlı eğitim başarımlarından rasgele alınan bir sonuç.

5.5.2. Zambak Çiçeği Veri Kümesi ile İki Sınıflı Sınıflama Problemi

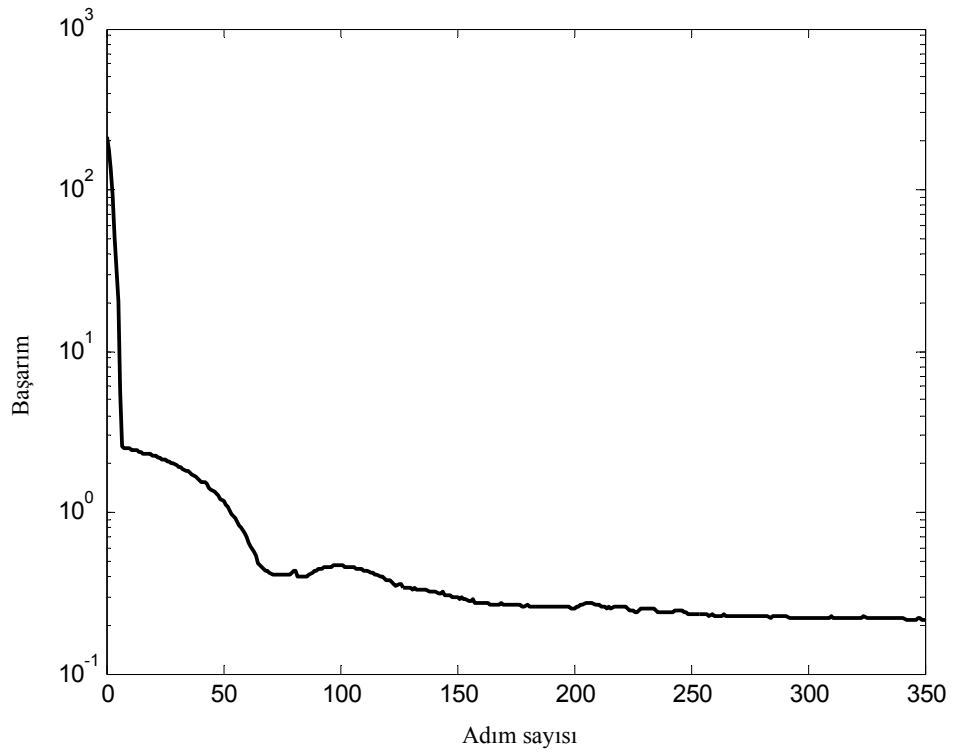
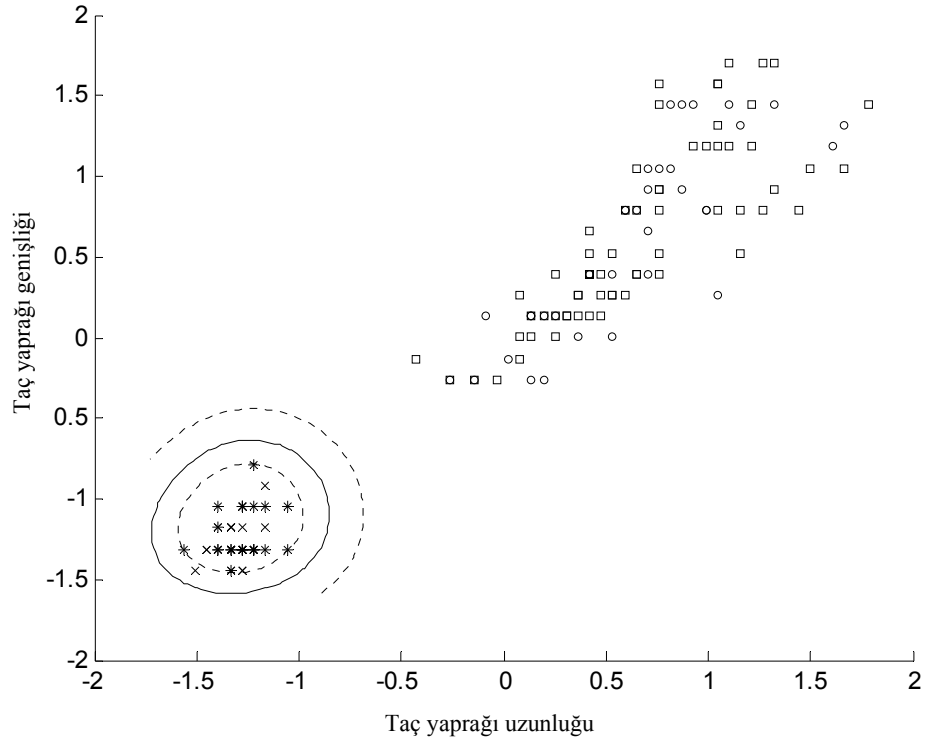
Burada; Zambak çiçeği verisinin 3. ve 4. karakteristikleri alınarak, tüm sınıflar için ayrı ayrı sınıflayıcı tasarlanmıştır. Her bir sınıflayıcı için, karar yüzeyinin içinde kalması gereken veriler 1 ile, diğerleri ise -1 ile etiketlenmiştir.

Veriler, önceki örnekte olduğu gibi önce rasgele karıştırılmış, daha sonra 10 eşit alt kümeye bölünerek 4'ü eğitim 6'sı test kümesi olarak ayrılmıştır. Önerilen formülasyonda tüm sınıflar için RTİ çekirdek kullanılmıştır. RTİ çekirdeğin σ parametresini seçmek için [0.01: 0.01: 3] dizisinin tüm değerleri için algoritma çalıştırılarak test başarımlarına bakılmıştır. En iyi test başarımları veren değerler için algoritma 10 defa çalıştırılmış, hem eğitim hem de test de en iyi başarımları veren bir değer bu uygulama için seçilmiştir. Önerilen algoritma bir önceki örnekteki

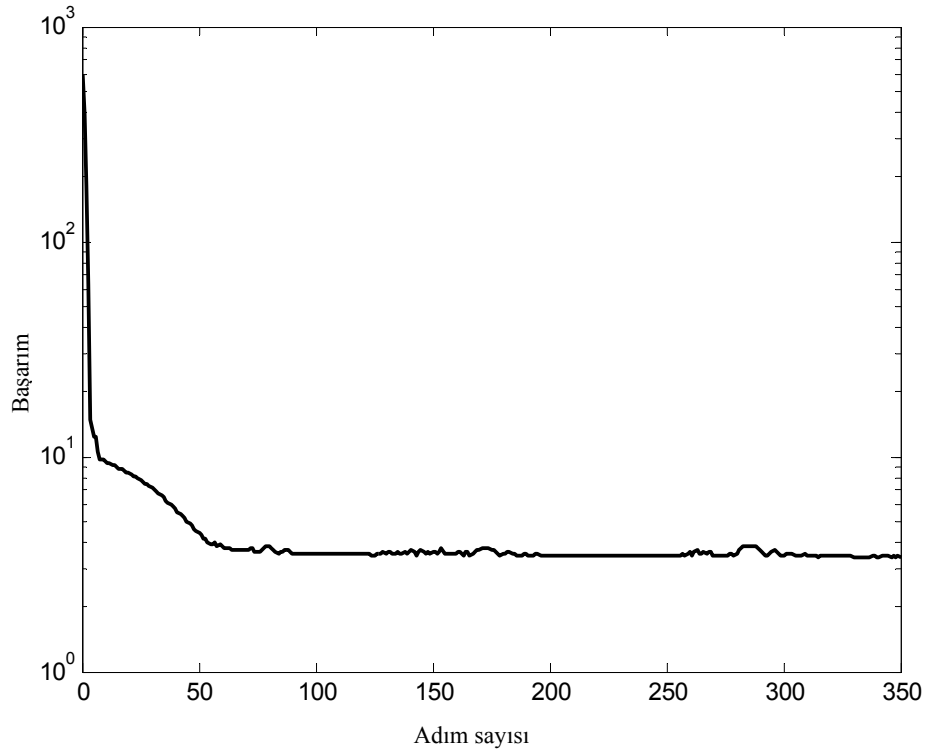
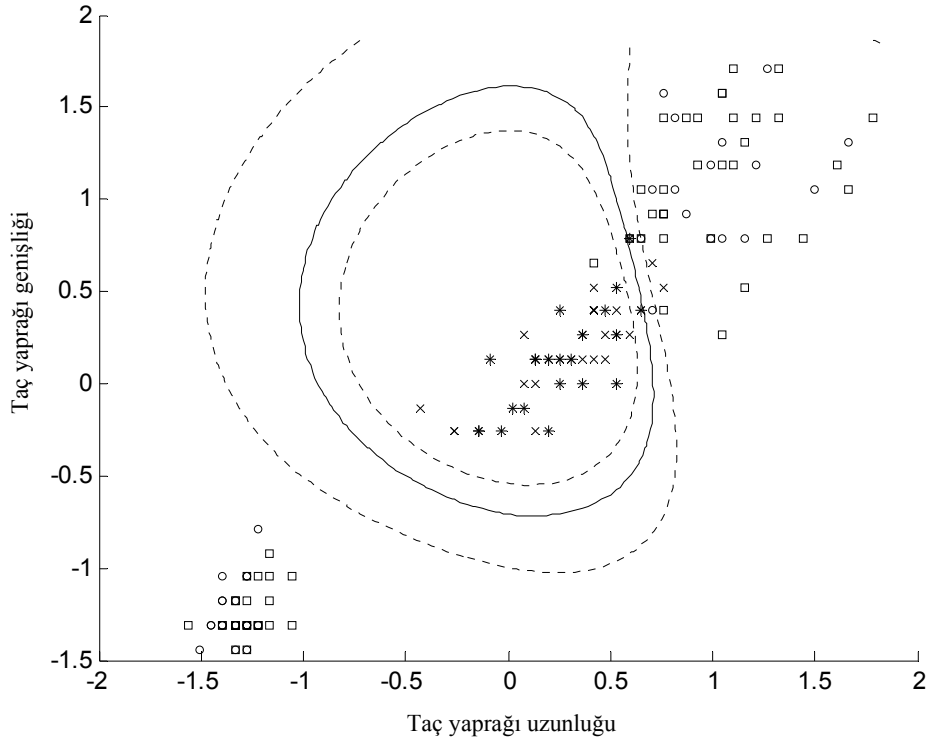
gibi 350 adım için 10 defa çalıştırılarak, hem eğitim hem de test verileri için doğru sınıflanmış noktaların toplam yüzdesi, ortalama doğruluk olarak Tablo 5.2’de verilmiştir. Tabloda verilen DVM sonuçları da Bölüm 2’den alınmıştır. Önerilen formülasyon kullanılarak elde edilen karar yüzeyleri, eğitim ve test örneklerini içererek Şekil 5.2’de verilmiştir. Sonuçlardan görüldüğü gibi önerilen sınıflayıcı test başarımları ve süre açısından DVM’yi geçmiştir.

Tablo 5.2 Zambak çiçeği veri kümesi üzerinde önerilen çekirdek tabanlı sınıflayıcı ile DVM’nin 10 katlı eğitim ve test doğrulukları ile eğitim sürelerinin kıyaslanması.

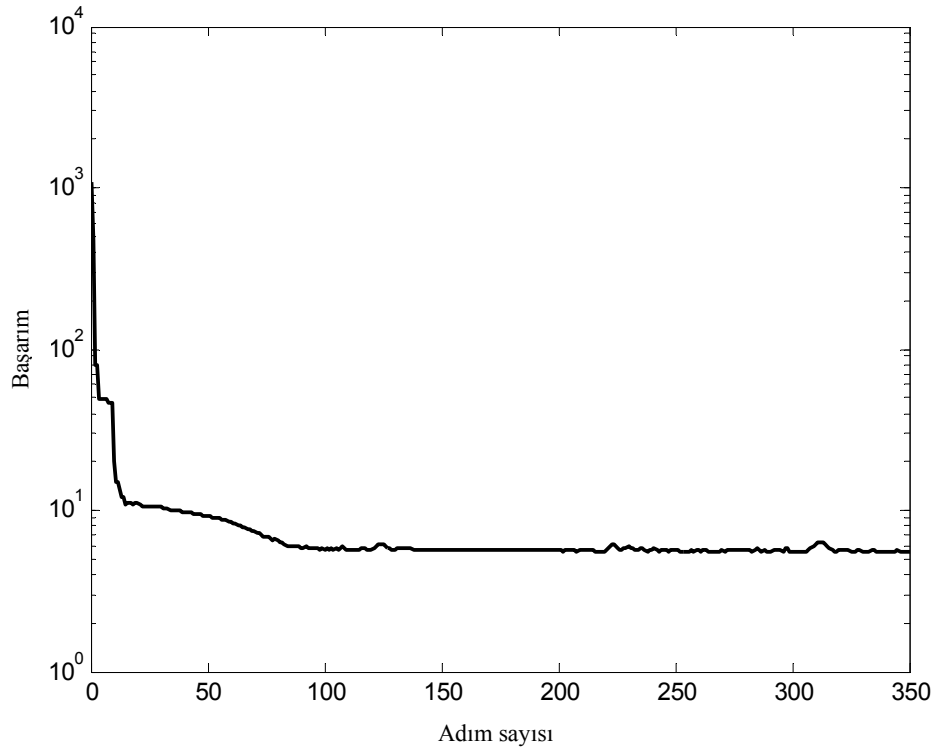
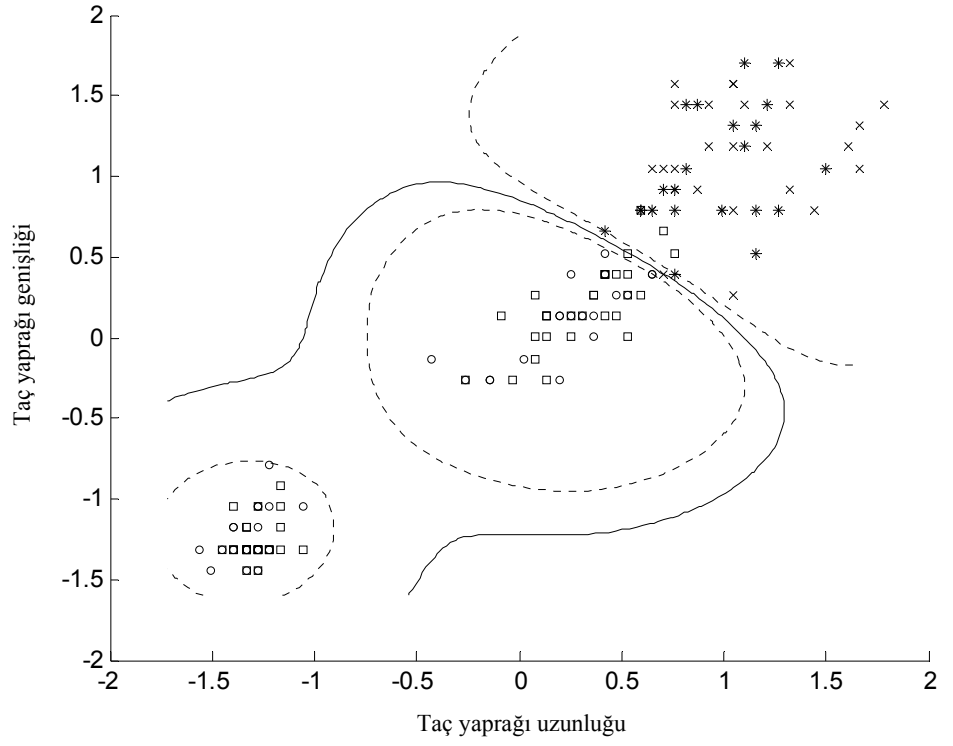
Zambak Çiçeği	1. Sınıf	2. Sınıf	3. Sınıf
Önerilen Formülasyon, RTİ çekirdek			
Eğitim Doğruluğu [%]	100	97.66	99.83
Test Doğruluğu [%]	100	96.17	95.05
Süre [s]	0.13	0.11	0.12
σ	0.29	0.61	0.5
DVM			
Eğitim Doğruluğu [%]	100	98.33	100
Test Doğruluğu [%]	100	94.38	93.25
Süre [s]	0.26	0.19	0.25
σ	0.125	0.125	0.125



(a)



(b)



(c)

Şekil 5.2 Zambak çiçeği verisinin (a) 1.sınıfı, (b) 2. sınıfı ve (c) 3. sınıfı üzerinde önerilen çekirdek tabanlı sınıflayıcıların eğitim ve test başarımları – Verilen bir sınıfa ait eğitim noktaları (*) ve test noktaları (x) ile gösterilirken, diğer sınıfa ait eğitim noktaları (o) ve test noktaları ise □ ile gösterilmiştir.

6. YAPISAL ve DENEYSEL RİSKİ ENAZLAYAN BULANIK MODELLER

6.1. Bulanık Tabanlı Modeller

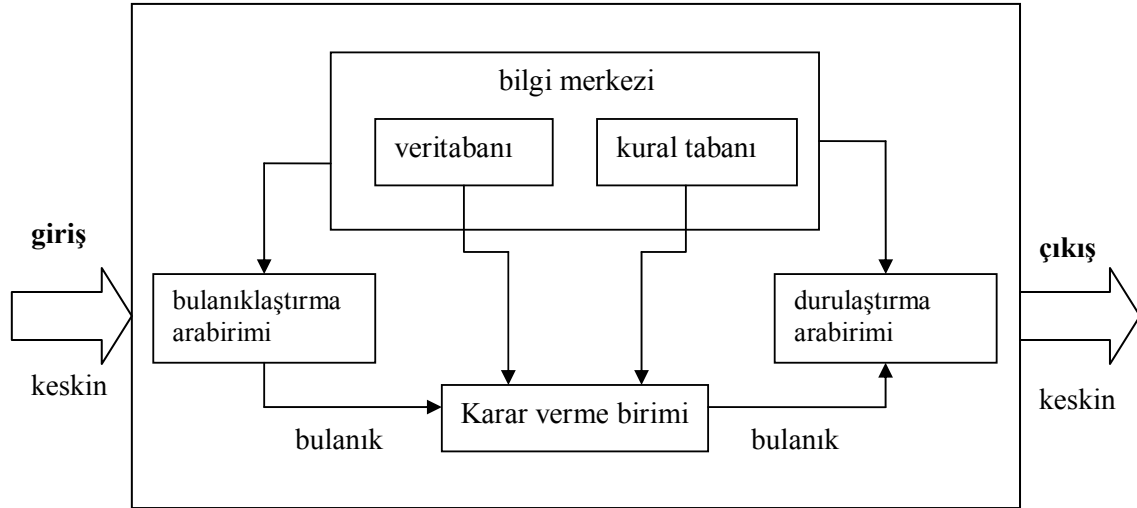
Bu bölümde, bilimsel yazında oldukça iyi bilinen Takagi-Sugeno bulanık tabanlı model yapısında, sadece deneysel riski enazlayan öğrenme algoritmaları yerine, yapısal ve deneysel riski aynı anda enazlayan bir öğrenme algoritmasının türetilmesi açıklanmıştır. Bu yapı ile, birden fazla çıkışlı yapıya genişletilebilme kolaylığı getirilmiştir.

[21]'de, doğrudan DVM'lerin eğitimi sonucunda elde edilen ağırlık değerleri ve destek vektörleri kullanılarak, yapısal riski enazlayan Takagi-Sugeno bulanık tabanlı model yapısı oluşturulmuştur. [59]'da ise bulanık modeller için eniyileme problemi, yapısal hatayı da enazlayacak şekilde DVM'lere benzer olarak oluşturulmuş ve ikincil biçimde çözülerek bulanık model parametreleri elde edilmiştir. Bu tezde ise, DVM'lerin istatistiksel öğrenme algoritması iki şekilde bulanık modellere uygulanmıştır. Birinci uygulamada, yapısı yani kural sayısı ve kurallarının parametreleri bilinen bulanık modeller, genelleme yeteneğini artırmak, gürültü ve aykırı verilere karşı gürbüz olmasını sağlamak için istatistiksel öğrenme algoritması ile eğitilmiştir. İkinci uygulamada, tüm eğitim verileri kullanılarak başlatılan bir bulanık model yapısı, birinci uygulamadaki gibi eğitilmiştir. Fakat, her bir adımda, sadece deneysel hata terimine katkısı bulunan noktalarla hesaplamalar yapılmış ve daha küçük olan bir yapı elde edilmiştir. Böylece bulanık modellerdeki önceden kural sayısının belirlenmesi probleminde kaçınılmıştır.

Temel olarak, bulanık tabanlı sistem Şekil 6.1'den görüldüğü gibi dört işlevsel bloktan oluşmaktadır. Bunlar,

- Bulanık eğer-ise kurallarından oluşan bulanık kural tabanı,
- Kurallar üzerinden sonucu oluşturma işlemlerini uygulayan karar verme makinesi,
- Girişleri dilsel değerlere ve üyelik derecelerine dönüştüren bulanıklaştırma arabirimi ve
- Çıkarımın bulanık sonuçlarını keskin çıkışlara dönüştüren durulaştırma arabirimidir.

Bilimsel yazında yaygın olarak kullanılan üç farklı bulanık model vardır. Modeller arasındaki fark, kural sonuçlarının farklı olmasından, kurallarının birleştirilme yönteminden ve durulaştırma adımlarının da ona göre farklılık göstermesinden kaynaklanır [4, 8, 57, 83]. İki girişli iki kurallı bir sistem için aşağıda kısa tanımları verilen üç bulanık model yapısı Şekil 6.2'de görülmektedir.



Şekil 6.1 Bulanık tabanlı modelin karar verme düzeni.

1. Tsukamoto bulanık modelde, kuralların sonuç bölümü tekdüze (monotonik) üyelik işlevleri ile gösterilir. Her bir kuralın çıkarılmış sonucu, kuralların gerçekleştirme dereceleri ile tanımlanan bir keskin değerdir. Tüm çıkış ise, her bir kuralın keskin çıkışının ağırlıklı ortalaması olarak hesaplanır [83]. Bu yöntemde kullanılan kurallar aşağıdaki gibi ifade edilir:

eğer x_1 , A1 ve x_2 , B1 ise; o halde $Z = C1$ 'dir.

eğer x_1 , A2 ve x_2 , B2 ise; o halde $Z = C2$ 'dir.

2. Mamdani bulanık modelin, kuralların sonuç bölümü farklı üyelik işlevleri ile gösterilebilir [83]. Tüm sistem çıkışı, nitelikli bulanık çıkışlara “maks” işlemi uygulanarak elde edilir. Burada nitelikli bulanık çıkışla, gerçekleşme derecelerinin en küçüğüne göre her bir kuralın çıkış üyelik işlev alanının elde edilmesi kastedilmiştir. Bununla birlikte Mamdani model özel uygulamalarda, farklı durulaştırma yöntemleriyle de kullanılmıştır.

Bu yöntemde kullanılan kurallar aşağıdaki gibidir:

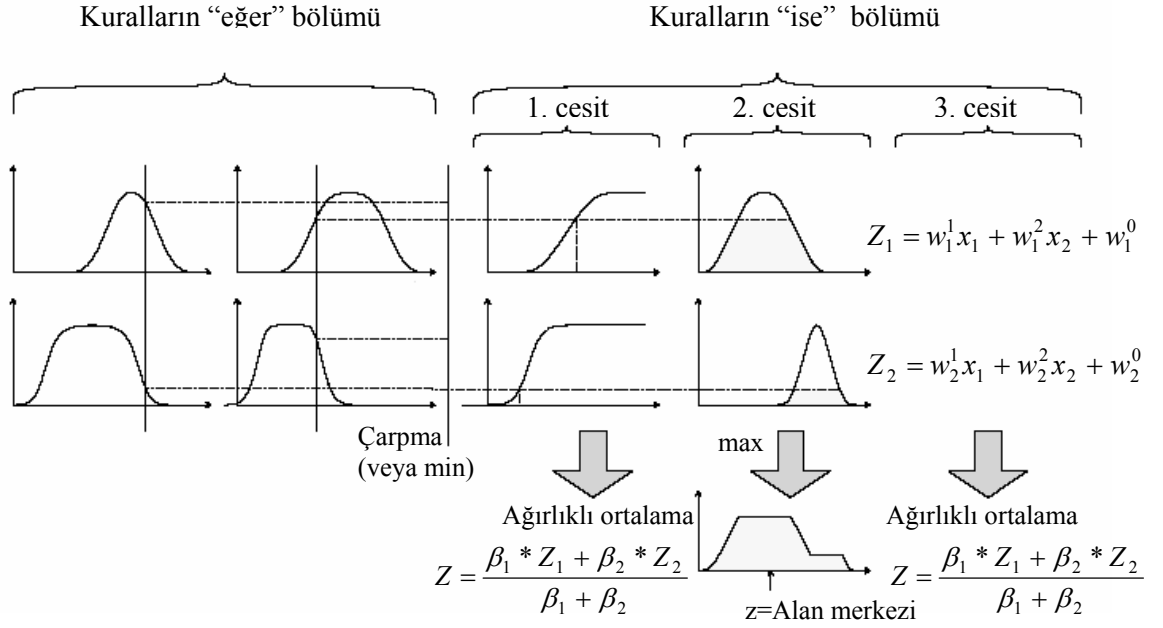
eğer x_1 , A1 ve x_2 , B1 ise; o halde $Z = C1$ 'dir.

eğer x_1 , A2 ve x_2 , B2 ise; o halde $Z = C2$ 'dir.

3. Takagi, Sugeno ve Kang tarafından düşünülen yöntemde, her bir kuralın çıkışı üyelik işlevlerinin yerine, giriş değişkenlerinin doğrusal bir bileşimi ile sabit terimden oluşan polinomsal bir denklem ile verilir. Tüm çıkış ise her bir kural çıkışının ağırlıklı ortalaması alınarak hesaplanır [83]. Bu yöntemde kullanılan kurallar aşağıdaki gibidir:

eğer x_1 , A1 ve x_2 , B1 ise; o halde $Z_1 = w_1^1 x_1 + w_1^2 x_2 + w_1^0$ 'dir

eğer x_1 , A2 ve x_2 , B2 ise; o halde $Z_2 = w_2^1 x_1 + w_2^2 x_2 + w_2^0$ 'dir.



Şekil 6.2 Yaygın olarak kullanılan üç bulanık model yapısı [83].

6.2. Ele Alınan Bulanık Tabanlı Modelin Yapısı

Şekil 6.3'te 5 katmanlı yapısı görülen Takagi-Sugeno bulanık modelde, L veri sayısı olmak üzere, giriş değişkenleri $x = [x_1^1, x_2^2, \dots, x_n^L]^T$ ve hedef değişkeni $y = [y^1, y^2, \dots, y^L]^T$ alınıp, orijine göre mesafeyi temsil eden terimi ifade edebilmek için 1 ilave edilmiş giriş vektörü $X = [x^T \ 1]^T$ olursa sırasıyla kural yapısı ve katmanlarında yapılan tüm işlemler aşağıdaki gibi tanımlanır:

R_1 : Eğer $x_1, \phi(x_1, c_1^1)$ ve ... $x_n, \phi(x_n, c_n^1)$ ise, $Z_1 = w_1 X$ 'dir,

R_2 : Eğer $x_1, \phi(x_1, c_1^2)$ ve ... $x_n, \phi(x_n, c_n^2)$ ise, $Z_2 = w_2 X$ 'dir,

.....

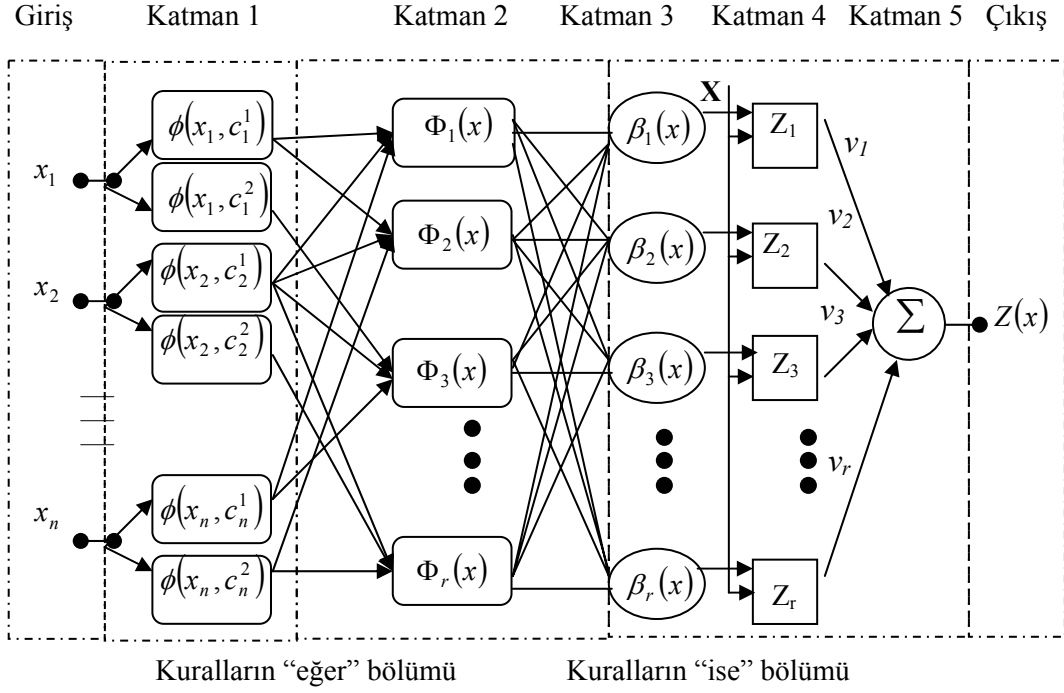
.....

.....

.....

R_r : Eğer $x_1, \phi(x_1, c_1^r)$ ve ... $x_n, \phi(x_n, c_n^r)$ ise, $Z_r = w_r X$ 'dir. (6.1)

Burada, r kuralların sayısını, n giriş değişkenlerinin boyutunu, w_j sonuç parametrelerini ve $\phi(x_i, c_i^j)$ ($j = 1, 2, \dots, r, i = 1, 2, \dots, n$) girişlere ait bulanık kümeyi göstermektedir.



Şekil 6.3 Ele alınan Takagi-Sugeno bulanık tabanlı model.

Katman 1: Bu katmanda her bir giriş değişkeni bir düğüme atanır ve bir sonraki katmana dağıtılır.

Katman 2: Bu katmanda girişler üyelik işlevlerinden geçirilerek bulanıklaştırılır. Uygulamalarda gaus üyelik işlevi

$$\phi(x_i, c_i^j) = \exp\left(-\frac{\|x_i - c_i^j\|^2}{2(\sigma^j)^2}\right) \quad (j = 1, 2, \dots, r, i = 1, 2, \dots, n) \quad (6.2)$$

olarak kullanılmıştır. Burada, c_i^j üyelik işlevinin merkezini, σ^j j . üyelik işlevinin değişim parametresini göstermektedir.

Katman 3: Bu katmanda, üyelik işlevleriyle elde edilen dönüşümlerin bulanık kesişimi hesaplanır. Her bir düğüm için t-norm işlemi olarak cebirsel çarpım yapılırsa,

$$\Phi_j(x) = \prod_{i=1}^n \exp\left(-\frac{\|x_i - c_i^j\|^2}{2(\sigma^j)^2}\right) = \exp\left(-\frac{1}{2} \sum_{i=1}^n \frac{\|x_i - c_i^j\|^2}{(\sigma^j)^2}\right), \quad (6.3)$$

olur ve (6.4)'teki gibi ölçeklenir

$$\beta_j(x) = \frac{\Phi_j(x)}{\sum_{j=1}^r \Phi_j(x)}. \quad (6.4)$$

Katman 4: Ölçeklenmiş β_j ağırlığı, kuralların çıkışlarını gösteren Z_j değişkeni ile çarpılır.

Her bir düğüm çıkışı

$$v_j = \beta_j Z_j$$

$$= \frac{\sum_{j=1}^r \Phi_j(x) w_j X}{\sum_{j=1}^r \Phi_j(x)} \quad (6.5)$$

olarak hesaplanır.

Katman 5: Oluşturulan bulanık modelin tüm çıkışı, ağırlıklı ortalama yöntemi kullanarak durulaştırılır,

$$Z(x) = \sum_{j=1}^r \beta_j(x) Z_j$$

$$= \frac{\sum_{j=1}^r \Phi_j(x) w_j X}{\sum_{j=1}^r \Phi_j(x)} . \quad (6.6)$$

Burada, her bir kuralın sonuç parametreleri $w = \left[(w^1)^T \ (w^2)^T \ \dots \ (w^r)^T \right]^T$ ve ölçeklenmiş girişler $\varphi(X) = \left[\beta^1(x) X^T \ \beta^2(x) X^T \ \dots \ \beta^r(x) X^T \right]^T$ biçimine getirilirse, bulanık sistemin tüm çıkışı kısaca $Z(X) = \varphi^T(X) w$ olarak ifade edilebilir.

6.3. Regresyon Kestirimi İçin Bulanık Tabanlı Modellerde Önerilen Öğrenme Algoritması

Bu bölümde, (6.6)'da verilen bulanık yaklaşım modeli kullanılarak yeni bir öğrenme algoritması türetilmiştir. Algoritma Bölüm 4'de verilen RTİA dönüşümü kullanılarak elde edilen regresyon kestirimi problemine benzer olarak uygulanmıştır.

DVM'lerde kullanılan tüm kayıp işlevleri, önerilen bulanık sistemin öğrenme algoritması için de kullanılabilir. Uygulamalarda, ε -duyarsız kayıp işlevi seçilerek eniyileme problemi

$$\min_{w, c, \sigma} \sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right| - \varepsilon \right) + \frac{A}{2} \|w\|^2 \quad (6.7)$$

$$A > 0$$

olarak önerilir. Burada; A ödünleşim sabiti

$$A = \frac{2}{L} \frac{\sum_{i=1}^L f \left(\left| y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right| - \varepsilon \right)}{\|w\|^2} \quad (6.8)$$

ile hesaplanır.

Regresyon kestirimi formülasyonları, önceki bölümlerde verilen çok çıkışlı sınıflama problemlerine benzer bir şekilde genişletilerek, kolaylıkla çok çıkışlı bulanık modeller elde edilebilir.

$$\min_{w, c, \sigma} \sum_{m=1}^M \sum_{i=1}^L f \left(\left| y_m^i - \left(\sum_{j=1}^r w_{mj} \varphi_j(X^i) \right) \right| - \varepsilon \right) + \frac{A}{2} \sum_{m=1}^M \|w_m\|^2 \quad (6.9)$$

Bu tezde, iki farklı bulanık tabanlı model oluşturulmuştur. Her iki yaklaşımda da Şekil 6.4'te verilen bir plan izlenmiştir. Birinci yaklaşım için önerilen işlemler aşağıdaki gibi özetlenmiştir:

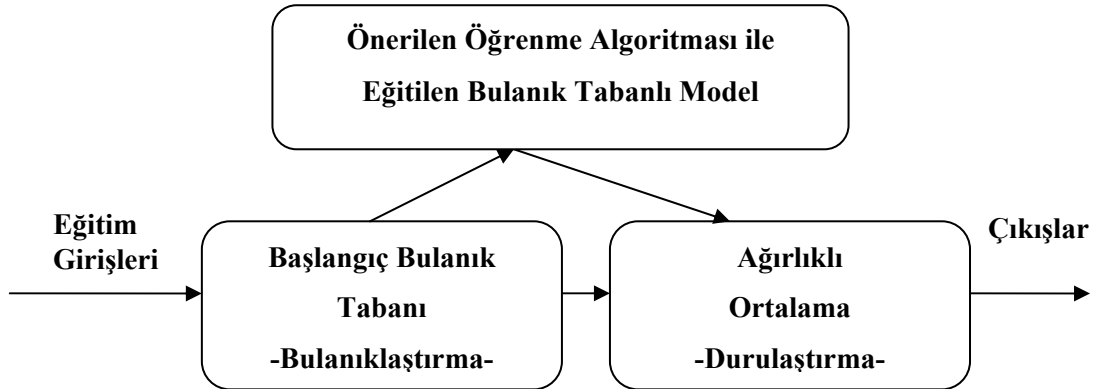
1. İşlem: Başlangıç kural sayısı, merkez ve değişinti parametreleri öbekleme algoritması ile oluşturulur. Bulanık model çıkışı

$$Z(x) = \varphi W \quad (6.10)$$

ile gösterilerse, başlangıç bulanık tabanlı sistem yapısı için çıkış parametreleri

$$W = (\varphi^T \varphi)^{-1} \varphi^T Z. \quad (6.11)$$

olarak hesaplanır. Burada W parametre vektörü, genelleştirilmiş sağ ters işlemi kullanılarak elde edilir.



Şekil 6.4 Önerilen bulanık tabanlı modelin öğrenme algoritması.

2. İşlem: Eğitim için, bilinen deneysel riski enazlayan öğrenme algortimaları yerine, yapısal hatayı da içeren (6.7)'deki eniyileme problemi momentum terimli uyarlanır öğrenme oranlı eğim iniş yöntemi kullanılarak çözülür. Bu amaçla önce (6.12-6.14)'teki gibi w , c ve σ

değişkenlerine göre negatif gradyenleri alınır. Daha sonra da algoritma gereğince özyineli olarak güncellenir.

$$\frac{\partial F}{\partial w_j} = - \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right) \cdot \varphi_j(X^i) + A.w \quad (6.12)$$

$$-\frac{\partial F}{\partial c_j} = \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right) \cdot \varphi_j(X^i) \frac{(x_i - c_j) \left(w_j - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right)}{\sigma^2} \quad (6.13)$$

$$-\frac{\partial F}{\partial \sigma_{ij}} = \sum_{i=1}^L f' \left(\left| y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right| - \varepsilon \right) \operatorname{sgn} \left(y^i - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right) \cdot \varphi_j(X^i) \frac{\|x_i - c_j\|^2 \left(w_j - \left(\sum_{j=1}^r w_j \varphi_j(X^i) \right) \right)}{\sigma_{ij}^3} \quad (6.14)$$

İkinci yaklaşımda ise öbekleme algoritması kullanmadan küçük bir bulanık model yapısı oluşturmak için yapılan işlemler, aşağıdaki gibi özetlenmiştir:

1. İşlem: Kural sayısı eğitim veri sayısı kabul edilir. Merkezlere eğitim verileri, sabit (tek) değişinti parametresine ise rasgele bir değer atanarak bulanık tabanlı sistem yapısı başlatılır.

2. İşlem: Önerilen öğrenme algoritması çalıştırılır. Her adımdaki hesaplamalarda sadece deneysel hata terimine, $|Y - (\varphi^T(X)W)| \geq \varepsilon$ şartını sağlayarak pozitif bir terim ekleyen noktalar kullanılır.

3. İşlem: Bir sonraki adımda $|Y - (\varphi^T(X)W)| \geq \varepsilon$ şartını sağlayan yeni noktalar belirlenir. Belirlenen noktalar, bir önceki adımdaki noktalar ile aynı ise öğrenme sonlandırılır, diğer durumda belirlenen adım sayısı kadar aynı işlemler tekrar edilir. Belirlenen en son noktalar kuralları oluşturmak için kullanılır. Böylece hem hesapsal yük azaltılmış olur hem de kuralları az olan bir bulanık model elde edilir.

6.4. Uygulamalar

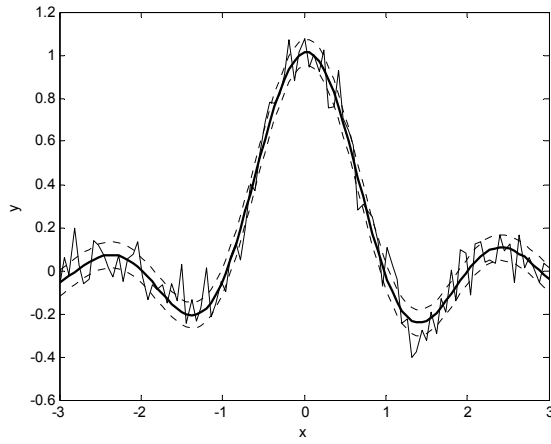
6.4.1. Sinc İşlevi ile Regresyon Kestirimi Problemi

Bu uygulamada, $y^i = \sin(\pi x^i) / (\pi x^i) + 0.1 n^i$ regresyon problemi ele alınmıştır. Burada, $x^i \sim U[-3, 3]$ aralığında düzgün bir dağılıma sahip giriş değerlerini, $n^i \sim N(0, 1)$ sıfır ortalamalı ve birim değişimli normal dağılıma sahip gürültü işaretini ve $y^i \in \mathbb{R}$ hedef değerlerini göstermektedir. Rasgele değişken x^i , n^i 'den bağımsızdır.

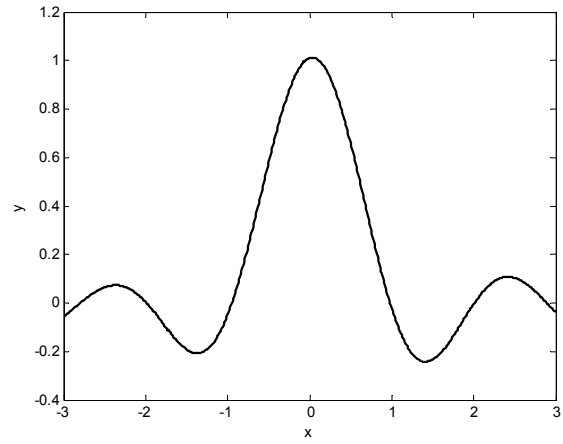
Eğitmek amacıyla 101 veri, test etmek amacıyla 600 veri kullanılarak dört farklı test yapılmış ve önerilen algoritmanın başarımı denenmiştir. İlk üç testte, Gaus üyelik işlevlerinin parametreleri olan merkezler ve değişinti parametreleri, bulanık öbekleme [10] algoritması kullanarak hesaplanmıştır. Bu üç test için 15 kural oluşturulmuştur. Birinci testte, sadece w parametresi eğitim iniş yöntemleriyle güncellenmiştir. Eğitimde merkez ve değişinti parametrelerinin başlangıç değerleri kullanılmıştır. İkinci testte, bu parametreler de güncellenmiştir. Üçüncü testte, bilimsel yazından bilinen bulanık tabanlı sistemlerdeki gibi A ve ε parametrelerinin değeri sıfır alınarak w , c ve σ değerleri güncellenmiştir. Dördüncü testte, tüm eğitim verileri Gaus üyelik işlevinin merkezleri olarak, değişinti parametresinin değeri de 0.7 ile başlatılarak w , c ve σ değerleri güncellenmiştir. Eğitim sonlandırıldığında 9 kural elde edilmiştir. Ortalama Karesel Hatanın Karekökü (OKHK) olarak elde edilen eğitim ve test başarımları, Tablo 6.1'de verilmiştir. İkinci test ile elde edilen eğitim ve test sonuçları, Şekil 6.5'de ve 3. test ile elde edilen eğitim ve test sonuçları Şekil 6.6'da verilmiştir. Tasarlanan bulanık modelin, gürültü ve aykırı verilere karşı gürbüz olduğu ve bilinen bulanık modele göre daha düzgün bir yaklaşım sağladığı Şekil 6.5'den görülmektedir. Buradan anlaşılabacağı gibi bulanık öbekleme algoritması kullanılarak başlatılan modelde iyi bir başarımlar için tüm parametreler eğitilirse, eğitilmeyen duruma göre daha iyi sonuçlar alınabilir. Ayrıca tüm veriler kullanılarak başlatılan bulanık modelde, her adımda sadece eğitime katkıda bulunan noktalar dikkate alınarak, hem fazla hesaplamalardan kaçınılır hem de sonuç bulanık model yapısı kolaylıkla oluşturulur.

Tablo 6.1 Önerilen bulanık modelin sinc işlevi üzerinde eğitim ve test başarımı.

Önerilen Güncelleme	Eğitim Hatası OKHK	Test Hatası OKHK
A değişken ve $\varepsilon=0.06$ iken sadece w 'nin güncellenmesi	0.0501	0.0520
A değişken ve $\varepsilon=0.06$ iken w , c ve σ 'nin güncellenmesi	0.0157	0.0158
$A=0$ ve $\varepsilon=0$, w , c ve σ 'nin güncellenmesi	0.0881	0.0624
Tüm eğitim verileri kuralları oluşturmak için kullanılırsa, A değişken ve $\varepsilon=0.06$ iken w , c ve σ 'nin güncellenmesi	0.0432	0.0441

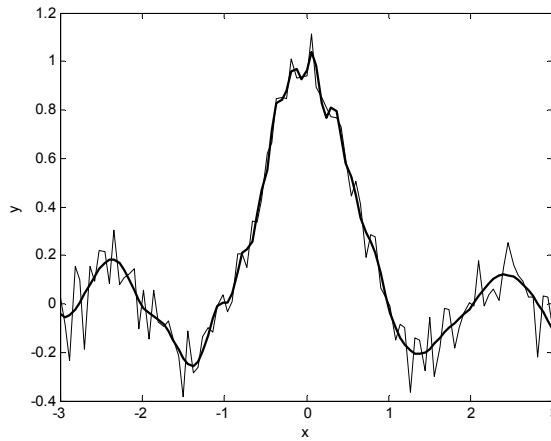


(a)

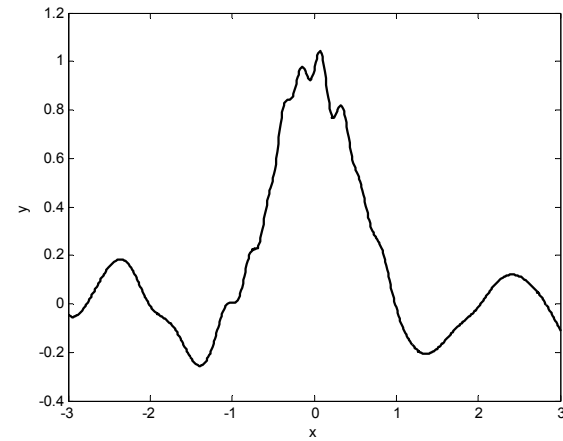


(b)

Şekil 6.5 Önerilen bulanık tabanlı sistem ile regresyon kestirimi, (a) eğitim sonuçları, eğitim verileri düz çizgi, ε -duyarsız bölge kesikli çizgi, hesaplanan eğri kalın düz çizgi ile gösterilmiştir (b) test sonuçları.



(a)



(b)

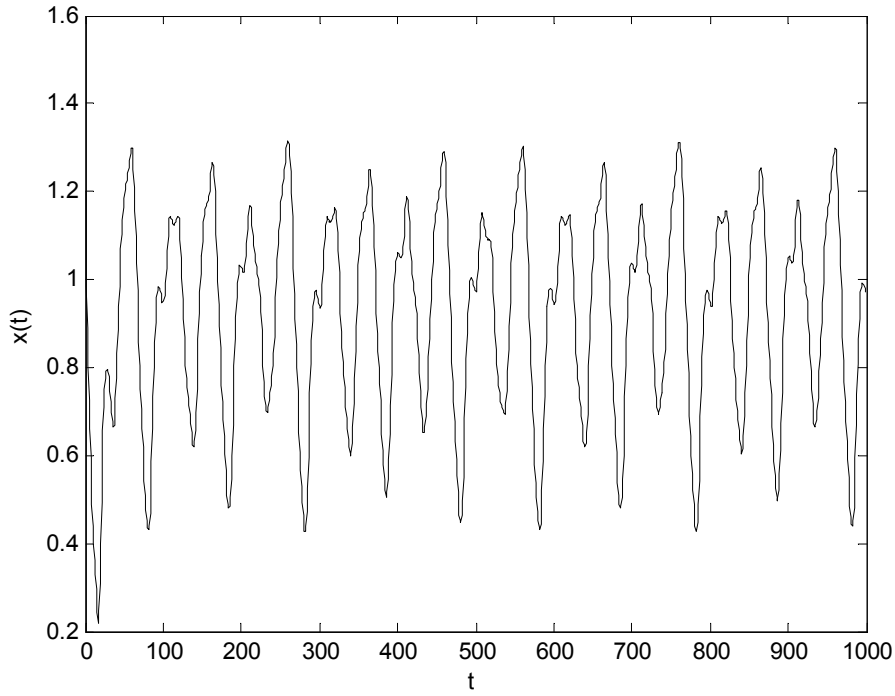
Şekil 6.6 Bilinen bulanık tabanlı sistem ile regresyon kestirimi, (a) eğitim sonuçları, eğitim verileri düz çizgi, hesaplanan eğri kalın düz çizgi ile gösterilmiştir (b) test sonuçları.

6.4.2. Mackey-Glass Zaman Serisi ile Öngörüm Problemi

Bu uygulamada, (6.15) ile verilen kaotik denektaş Mackey-Glass diferansiyel denklemi ele alınmıştır.

$$\frac{dx(t)}{dt} = \frac{bx(t - \mathfrak{T})}{1 + x^{10}(t - \mathfrak{T})} + a x(t) \quad (6.15)$$

Mackey-Glass diferansiyel denklemi $a=-0.1$, $b=0.2$ ve $\mathfrak{T}>17$ iken, kaotik davranış gösterir. Bu çalışmada $\mathfrak{T}=30$ alınarak karşılaştırmalar yapılmıştır. (6.15)'in sayısal çözümünü elde etmek için 4 adımlı Runge Kutta kullanılmıştır. Yöntemde başlangıç zamanı 0.1, başlangıç şartı $x(0)=0.9$ alınmıştır. Bu parametreler önceki çalışmalar ile karşılaştırma yapabilmek için seçilmiştir [21, 60, 105]. Öngörümek istenen Mackey-Glass zaman serisi Şekil 6.7'de görülmektedir. Öngörüm için serinin 1000 verisi kullanılmıştır.



Şekil 6.7 Kaotik Mackey-Glass zaman serisi.

Öngörüm işlemi, bir zaman serisinin önceki değerlerinden gelecek bilgileri doğrusal olmayan bir işlev yardımıyla kestirmek olarak tanımlanabilir. $x(t)$ zaman serisinin $x(t - \wp + 1)$ gibi değerinin öngörümü

$$x(t - \wp + 1) = \phi(p, x(t - \tau), x(t - \tau + 1), \dots, x(t - 1))$$

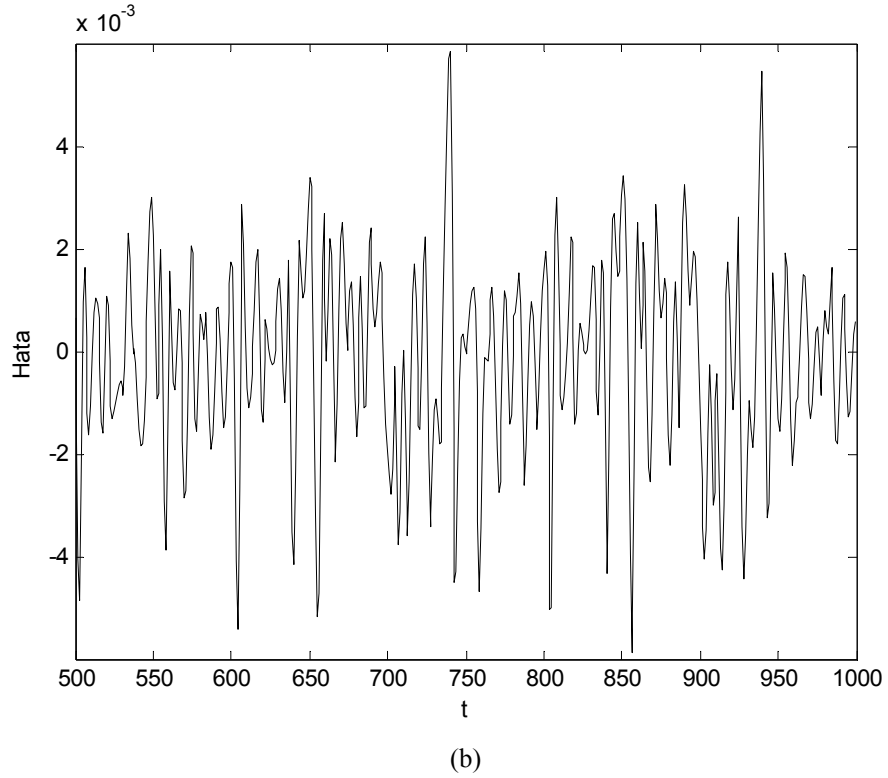
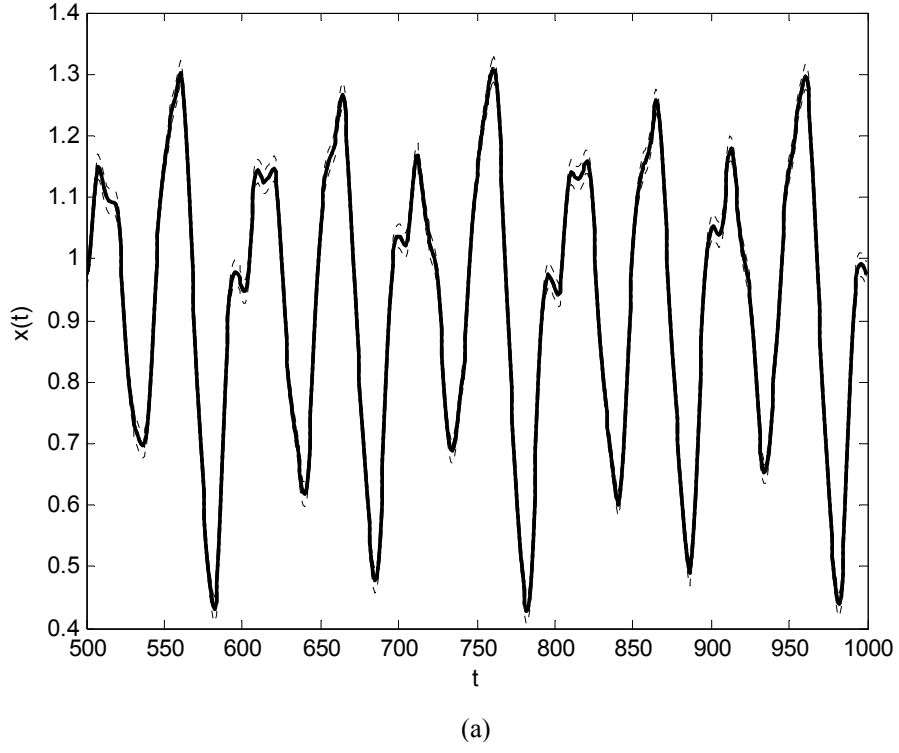
ile verilir. Burada; τ : zaman gecikmesini, p : doğrusal olmayan dönüşüm kabiliyetine sahip olan modelin değiştirilebilir katsayılarını ve $x(t - \tau), \dots, x(t - 1)$ ise model girişlerini göstermektedir. Bu örnekte; $\tau=9$, $\rho=1$ ile bir eğitim kümesi oluşturulmuştur. $x=t+P$ noktasındaki değeri öngörmek için $x=t$ noktasına kadar olan bilinen değerler kullanılmıştır.

Öngörüm için serinin 200 noktası $x(501) \sim x(700)$ eğitim verisi olarak ve serinin son 300 noktası $x(701) \sim x(1000)$ test verisi olarak kullanılmıştır. Önerilen İstatiksel Öğrenme tabanlı Bulanık Modeller (İÖBM) kullanılarak elde edilen kaotik serinin 1000 noktası gösterilmiştir.

$\varepsilon=0.02$ alınarak ve algoritma 350 adım için 10 defa çalıştırılarak elde edilen ortalama test ve eğitim başarımları Tablo 6.2’de verilmiş ve bilimsel yazında önceki çalışmalar ile kıyaslanmıştır. Bu tabloda öbekleme algoritması ile başlatılan birinci bulanık model 1. İÖBM ve tüm veriler kullanılarak başlatılan ikinci bulanık model 2. İÖBM olarak gösterilmiştir. Sonuçlardan görüldüğü gibi öngörüm başarımları tamamen kabul edilebilirdir. Burada oluşturulan bulanık sistem eğitildiği için ve öbekleme algoritması ile başlatılma olanağı bulunduğu için [21]’e göre daha az kural ve daha az hata elde edilmesi doğal bir sonuçtur. Şekil 6.8’de önerilen birinci bulanık model ile elde edilen eğitim ve öngörüm sonuçları verilmiştir.

Tablo 6.2 Mackey-Glass zaman serisinin son 300 verisinin öngörümü problemi üzerinde, farklı yöntemlerin başarımlarının kıyaslanması.

	ε	Kural sayısı	Eğitim Hatası (OKHK)	Test Hatası (OKHK)
[105]	-	121	-	0.0800
[60]	-	22	-	0.0800
[21]	0.0200	30	0.0134	0.0127
1. İÖBM	0.0200	10	0.0050	0.0070
2. İÖBM	0.0200	24	0.0087	0.0100



Şekil 6.8 $x(501) \sim x(1000)$ zaman serisinin benzetim sonuçları (a) $\varepsilon=0.02$ ve öbekleme algoritması kullanarak 10 kuralla oluşturulan bulanık modelin eğitim ve test başarımları, ε -duyarsız bölge (:), hesaplanan (-) ve gerçek değerler düz çizgi ile gösterilmiştir. (b) eğitim ve öngörüm hataları.

7. SONUÇLAR

Bu tezde; giriş uzayında DVM'lere seçenek olarak sınıflayıcı ve regresör tasarımı yapılmıştır. Burada ele alınan konular birçok açıdan incelenmesine rağmen, gelecek çalışmalar için de birçok araştırma konusu ortaya çıkmıştır. Bu bölümde, tüm sonuçlar ve gelecek araştırma konuları sunulmuştur.

Bölüm 2'de, giriş uzayında iki aşamalı sınıflayıcılar önerilmiştir. DVM gibi Lagrange çarpanları yöntemini kullanarak öz nitelik uzayında sınıflama yapan yöntemlerin tersine, önerilen yöntem ile penaltı parametresi kullanarak giriş uzayında sınıflama yapılmıştır. İlk aşamada, karar yüzeyinin küre biçimli olduğu kabul edilerek, DVM'lerdeki gibi hem yapısal hata hem de deneysel hata ilkesine dayanan doğrusal olmayan amaç ölçütü içeren bir eniyileme problemi tanıtılmıştır. Önerilen formülasyon eğer problem küre biçimiyle ayrıştırılabilirse, formülasyon %100 doğrulukla çözülecek şekilde oluşturulmuştur. Ayrıştırılamıyorsa sınıflama doğruluğunun azalmasına yani verilerin hatalı sınıflanmasına olanak vererek, önerilen eniyileme probleminden beklenen sonucun da en iyi olması sağlanmıştır. Çözüm için eğim iniş algoritmaları kullanılmıştır. Parametrelerin (öğrenme oranı ve penaltı parametrelerinin) başlatma problemi yoktur. Çünkü parametreler rasgele başlatılsa bile, her özinyilemede güncellendikleri için, algoritma birkaç adımda yakınsayarak çözüme ulaşmıştır. İkinci aşamada ise sınıflayıcıyı daha genel yapmak için karar yüzeyleri elipsoit biçimli olarak genişletilerek, yeni bir eniyileme problemi önerilmiştir. Birinci aşamaya benzer şekilde çözüm, eğim iniş yöntemleri ile yapılmıştır. Bunun için ilk aşamadan elde edilen merkez ve yarıçap değerleri, ikinci aşamanın başlangıç değerleri olarak atanarak eğitimde başlatma ve güncellemede oluşan güçlükler ortadan kaldırılmıştır. Son olarak, küre ve elipsoit biçimli karar yüzeyleri ile çok sınıflı sınıflama problemleri için tek bir formülasyon önerilmiştir. Önerilen algoritmaların katkısı; giriş uzayında çalışarak veri yapısının değiştirilmemesi, küre ve elipsoit gibi kolay yorumlanabilir karar yüzeyleri ile sınıflama yapılması, doğrudan karar yüzeyi ile ilgilenmeyi sağlayan yoğunluk temelli amaç ölçütü sayesinde sınıflardaki veri sayısı farklı olan problemlerde iyi sonuçlar vermesi ve çözümün DVM ve diğer yöntemler ile karşılaştırıldığında kabul edilebilir derecede hızlı olması olarak özetlenebilir. Bu sınıflayıcılar ile ilgili yeni bir araştırma konusu ise; veri sayısı fazla olan uygulamalarda eğitim süresini hızlandırmak için kayıp işlevinin 1-normu yerine 2-normu kullanılması ve ikinci dereceden bilgiyi içeren çözüm yöntemlerinin uygulanması olabilir.

Bölüm 3'te önerilen küre ve elipsoit biçimli sınıflayıcıların, gürültü ve aykırı verilere karşı gürbüzlüğünü artırmak için her veriye farklı bir üyelik değeri atanarak, yeni bulanık küre

ve elipsoit biçimli sınıflayıcılar tanıtılmıştır. Burada her bir eğitim örneğinin karar yüzeyinin tanımlanmasına katkısı, örneğe ait üyelik değeriyle belirlenmiştir. Böylece yüksek belirsizlikli örneklerin etkisi, eğitim örneğe atanan üyelik değeriyle ağırlıklandırarak azaltılmıştır. Bölüm 2’de önerilen yöntem zaten gürbüz olduğu için, fazla çaba göstermeden gürültülü ve aykırı değerli problemleri çözmek için en basit yöntemlerden birisi, farklı üyelik işlevleri kullanmaktır. Bununla birlikte gelecek çalışmalar için, üyelik değerlerinin verilerinin olasılık dağılımı temeline dayanarak elde edilmesi veya üyelik değerlerinin etkisinin formülasyona farklı bir şekilde karıştırılması önerilebilir.

Bölüm 4’de, küre ve elipsoit biçimli karar sınırlarının dışında daha karmaşık karar yüzeyleri elde ederek, sınıflama başarımını artırmak amacıyla RTIA’ların veya genel bir gizli katmanlı YSA’ların yüzeylerini kullanarak, sınıflama ve regresyon kestirimi için yeni bir formülasyon önerilmiştir. Bu bölümün katkısı, gizli katman uzayında gürbüz sınıflama ve regresyon kestirimi için formülasyonlar önerilmesidir. Gelecekte birden fazla gizli katmanlı ağların yüzeylerinin kullanımı yeni bir araştırma konusudur.

Bölüm 5’de, Bölüm 4’teki yöntem ve formülasyonlar çekirdek işlevi temel alınarak yeniden türetilmiştir. Çözüm için eğim iniş yöntemleri önerilmiştir. Formülasyonlarda penaltı parametresi uyarlanır seçilerek, gürültü ve aykırı verilere karşı daha az duyarlı olması sağlanmıştır. Bu bölümün katkısı ise çekirdek işlevleri kullanılarak ilk defa oluşturulan hem sınıflama hem de regresyon problemlerinin eğim iniş yöntemleri ile çok hızlı bir şekilde çözülmesi, programının kullanıcı tarafından kolaylıkla yazılabilmesi, seçimi zor olan penaltı parametresinin uyarlanabilir seçilmesi ve çoklu sınıflama için tek bir formülasyon önerilerek daha az parametreyle çözümün kolaylıkla yapılabilmesidir. Gelecek çalışmalar için, uygulamalarda karşılaşılan probleme bağlı olarak farklı kayıp işlevlerinin ve farklı normların kullanımı, penaltı parametresinin farklı olarak uyarlanabilmesi araştırmaya değerdir.

Bölüm 6’da bulanık sistemlerin genelleme yeteneğini artırmak, gürültü ve aykırı verilere karşı gürbüz olmasını sağlamak için DVM’nin istatistiksel öğrenme algoritmasına benzer bir öğrenme algoritması türetilmiştir. Bu bölümün katkısı, bilimsel yazında ilk defa istatistiksel öğrenme tabanlı bulanık modelin oluşturulması olarak özetlenebilir. Farklı kayıp işlevleri ve farklı normların kullanılarak, yeni bulanık modellerin oluşturulması ve farklı öğrenme algoritmaları kullanılarak hem başlangıç şartlarına duyarlılığın azaltılması hem de eğitimin hızlandırılması gelecek araştırmalara konu olabilir.

KAYNAKLAR

1. Astorino, A. and Gaudioso, M., 2003, Spherical separation and kernel transformations for classification problems, Technical Report, RT-ICAR-CS-03-16.
2. Astorino, A. and Gaudioso, M., 2005, Ellipsoidal separation for classification problems, *Optimizations Methods and Software*, 20, 12-20.
3. Ayat, N. E., Cheriet, M., Remaki, L., Suen, C. Y., 2001, KMOD- A new support vector machine kernel with moderate decreasing for pattern recognition, *Proceedings of the 6th Int. Conference on Document Analysis and Recognition*, 434-438.
4. Babuska, R., 1998, *Fuzzy Modelling and Control*. Kluwer Academic Publisher, United states of America.
5. Barnes, R., 1982, An algorithm for separating patterns by ellipsoids, *IBM Journal Of Research and Development*, 26, 6, 759-764.
6. Bertsekas, D. P., 1999, *Nonlinear Programming*. Athena Scientific, Belmont, MA, second edition.
7. Ben-Hur, A., Horn, D., Siegelmann, H. and Vapnik, V., 2001, Support vector clustering, *Journal of Machine Learning Research*, 2, 125-137.
8. Bezdek, J. C. and Pal, S. K. editors, 1992, *Fuzzy Models for Pattern Recognition*. IEEE Press, New York.
9. Bezdek, J. C., Keller, J. M., Krishnapuram, R., Kuncheva, L. I. and Pal, N. R., 1999, Will the the real Iris data please stand up?, *IEEE Trans. on Fuzzy Systems*, 7, 3, 368-369.
10. Bontempi, G. and Birattari, M., 1999, Toolbox neuro-fuzzy identification and data analysis for use with Matlab, Iridia, Universite Libre de Bruxelles, <http://iridia.ulb.ac.be>.
11. Boyd, S. and Vandenberghe, L., 2004, *Convex Optimization*. Cambridge University Press.
12. Brown, M. P. S., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W. and Furey, T. S., 2000, Knowledge-based analysis of microarray gene expression data by using support vector machines, *Proceedings of the National Academy of Science*, 97, 262-267.
13. Burges, C. J., 1998, A tutorial on support vector machines for pattern recognition, *Data Mining and Knowledge Discovery*, 2, 2, 121-167.
14. Chapelle, O., Vapnik, V., Bousquet, O. and Mukherjee, S., 2002, Choosing multiple parameters for support vector machines, *Machine Learning*, 46, 1-3, 131-159.
15. Chapelle, O., 2006, Training a support vector machine in the primal, submitted to *Journal*

of Machine Learning Research.

16. Chen, Z. and Haykin, S., 2002, On different facts of regularization theory, *Neural Computation*, 14, 12, 2791-2846.
17. Chen, J. H. and Chen, C. S., 2002, Fuzzy kernel perceptron, *IEEE Trans. on Neural Networks*, 13, 6, 1364-1373.
18. Chen, Y. and Wang, J. Z., 2003, Support vector learning for fuzzy rule-based classification systems, *IEEE Trans. on Fuzzy Systems*, 11, 6, 716-728.
19. Cherkassky, V. and Mulier, F., 1997, *Learning from Data: Concepts, Theory and Methods*. John Wiley & Sons, New York.
20. Chiang, J. and Hao, P., 2003, A new kernel-based fuzzy clustering approach: support vector clustering with cell growing, *IEEE Trans. on Fuzzy Systems*, 11, 4, 518-527.
21. Chiang, J. and Hao, P., 2004, Support Vector learning mechanism for fuzzy rule-based modeling: A new approach, *IEEE Trans. on Fuzzy Systems*, 12, 1, 1-12.
22. Christianini, N. and Taylor, J., 2000, *An Introduction to Support Vector Machines and Other Kernel Methods*. Cambridge University Press, Cambridge.
23. Cichochi, A. and Unbehauen, R., 1992, *Neural Networks for Optimization and Signal Processing*. John Wiley & Sons, New York.
24. Collobert, R. and Bengio, S., 2004, Link between perceptrons, MLPs and SVMs, Technical Report 04-06, IDIAP.
25. Cortes, C. and Vapnik, V. N., 1995, Support vector networks, *Machine Learning* 20, 3, 273-297.
26. Doğan, H. and Güzeliş, C., 2004, Robust spherical clustering as a mixed integer optimization problem and its gradient solution, *Proceedings of the IEEE Conference on Signal Processing and Communication Applications*, Kuşadası, Turkey, 739- 742.
27. Doğan, H., 2004, Gradient networks design for clustering in novel optimization frameworks, PhD. Thesis, Dokuz Eylül University.
28. Doğan, H. and Güzeliş, C., 2006, Robust and fuzzy spherical clustering by a penalty parameter approach, *IEEE Trans. on Circuits and Systems II*, 53, 8, 637-641.
29. Duda, R. O., Hart, P. E. and Stork, D. G., 2001, *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York.
30. Evgeniou, T., Pontil, M. and Poggio, T., 2000, Regularization networks and support vector machines, *Advances in Computational Mathematics* 13, 1, 1-50.
31. Fletcher, R., 1989, *Practical Methods of Optimization*. John Wiley & Sons, New York.
32. Fort, G., 2005, Matlab PLS logistic toolbox-Inference in logistic regression models, <http://www.tsi.enst.fr/%7Egfort/GLM/Programs.html>.

33. Fukunaga, K., 1990, *Introduction to Statistical Pattern Recognition*. Academic Press, San Diego.
34. Fung, G. and Mangasarian, O. L., 2002, Finite newton method for lagrangian support vector machine classification, Technical Report 02-01, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin.
35. Fung, G. and Mangasarian, O. L., 2001, Proximal support vector machine classifiers, In: D. Lee, et al. (Eds.), *Proceedings of the KDD-2001: Knowledge Discovery and Data Mining*, San Francisco, California, Association for Computing Machinery, New York, 77-86.
36. Fung, G. and Mangasarian, O. L., 2004, Multi-category proximal support vector machine classifiers, *Neurocomputing*, 61, 401-411.
37. Girosi, F., Jones, M. and Poggio, T., 1993, Priors, stabilizers and basis functions: From regularization to radial, tensor and additive splines, A.I. Memo No. 1430, MIT, 1993.
38. Girosi, F., 1997, An equivalence between sparse approximation and support vector machines, A.I. Memo No. 1606, MIT.
39. Glineur, F., 1998, Pattern separation via ellipsoids and conic programming, *Mémoire de D.E.A., Faculté Polytechnique de Mons*, Mons, Belgium.
40. Gunn, S. R., 1998, Matlab support vector machine toolbox, University of Southampton, Image Speech and Intelligent Systems Research Group, UK, <http://www.isis.ecs.soton.ac.uk/isystems/kernel/>.
41. Gunn, S. R., 1998, Support vector machines for classification and regression, Technical Report ISIS-1-98, Department of Electronics and Computer Science, University of Southampton.
42. Haasdonk, B., 2003, Feature space interpretation of SVMs with non positive definite kernels, Internal Report 1/03.
43. Hansen, P. C., 1998, *Rank deficient and discrete ill-posed problems: numerical aspects of linear inversion*, SIAM, Philadelphia, PA.
44. Hansen, P. C., 1994, Regularization Tools: A Matlab package for analysis and solution of discrete ill-posed problems, *Numerical Algorithms*, 6, 1-35.
45. Haykin, S., 1999, *Neural Networks: A Comprehensive Foundation*. Prentice Hall, second edition, New Jersey.
46. Hsu, C. W. and Lin, C. J., 2002, A comparison of methods for multi-class support vector machines, *IEEE Trans.on Neural Networks*, 13, 2, 415-425.
47. Huang, H. and Liu, Y.-H., 2002, Fuzzy support vector machines for pattern recognition and data mining, *International Journal of Fuzzy Systems*, 4, 3, 826-835.

48. Inoue, T. and Abe, S., 2001, Fuzzy support vector machines for pattern classification, Proceedings of the International Joint Conference on Neural Networks (July 15-19), Washington DC, 1449-1454.
49. Jayadeva, Khemchandani, R. and Chandra, S., 2004, Fast and robust learning through fuzzy linear proximal support vector machines, Journal of Neurocomputing, 61, 401-411.
50. Jayadeva, Khemchandani, R. and Chandra, S., 2005, Fuzzy linear proximal support vector machines for multi-category data classification, Journal of Neurocomputing, 67, 426- 435.
51. Joachims, T., 1999, Making large-scale support vector machine learning practical, In Schölkopf, B., Burges, C. and Smola, A. editors, Advances in Kernel Methods - Support Vector Learning, MIT Press, 169-184.
52. Kharechko, A. and Shawe Taylor, J., 2004, Text categorization via ellipsoid separation, Learning Methods for Text Understanding and Mining Workshop (26-29 January), Grenoble, France.
53. Kim, J., Suga, Y. and Won, S., 2006, A New approach to fuzzy modeling of nonlinear dynamic systems with noise: relevance vector learning mechanism, IEEE Trans. on Fuzzy Systems, 14, 2, 222-231.
54. Konno, H., Gotoh, J., Uryasev, S. and Yuki, A., 2002, *Failure discrimination by semidefinite programming*. Financial Engineering, Supply Chain and E-commerce, edited by P. Pardalos and V. Tsitsiringos, Kluwer Academic Publisher.
55. Kosko, B. 1996, *Fuzzy engineering*. Prentice Hall, New Jersey.
56. Kruss, M., 2002, Nonlinear multivariate analysis with geodesic kernels, Berlin Technical University, Thesis.
57. Kuncheva, L. I., 2000, *Fuzzy Classifier Design*. Physica-Verlag, New York.
58. Lee, Y.-J. and Mangasarian, O.L., 2001, SSVM: A smooth support vector machine, Computational Optimization and Applications, 20, 5-22.
59. Leski, J. M., 2005, TSK-Fuzzy modeling based on ϵ -Insensitive learning, IEEE Trans. on Fuzzy Systems, 13, 2, 181-193.
60. Lin, C.-J. and Lin, C.-T., 1997, An ART-based fuzzy adaptive learning control network, IEEE Trans. on Fuzzy Systems, 5, 477-496.
61. Lin, H. T. and Lin, C. J., 2003, A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods, Technical report, Department of Computer Science and Information Engineering, National Taiwan University.
62. Lin, C. F. and Wang, S. D., 2002, Fuzzy support vector machines, IEEE Trans. on Neural Networks 13, 2, 464-471.
63. Lin, C.-T., Yeh, C.-M., Liang, S.-F., Chung, J.-F. and Kumar, N., 2006, Support-vector-

- based fuzzy neural network for pattern classification, *IEEE Trans. on Fuzzy Systems*, 14, 1, 31-40.
64. Lyhyaoui, A., Martinez, M., Mora, I., Vaquez, M., Sancho, J. L. and Figueiras-Vidal, A. R., 1999, Sample selection via clustering to construct support vector-like classifiers, *IEEE Trans. on Neural Networks*, 10, 1474-1481
 65. Mayoraz, E. and Alpaydin, E., 1999, Support vector machines for multi-class Classification, *IWANN'99* (June), Alicante, Spain, 833-842.
 66. Mangasarian, O. L., 2002, A finite Newton method for classification problems, *Optimization Methods and Software*, 17, 913-929.
 67. Mangasarian, O. L. and Lee, Y.-J. 2000, Reduced support vector machines, Technical Report 00-07, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin.
 68. Mangasarian, O. L. and Musicant, D. R., 2000, Active support vector machine classification, Technical Report 00-04, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin.
 69. Mangasarian, O. L. and Musicant, D. R., 2001, Lagrangian support vector machines, *Journal of Machine Learning Research*, 1, 161-177.
 70. Mao, K. Z. and Huang, G., 2005, Neuron selection for RBF neural network classifier based on data structure preserving criterion, *IEEE Trans. on Neural Networks*, 16, 6, 1531-1540.
 71. Mill, J. and Inoue, A., 2003, An application of fuzzy support vectors, *Proceedings of the 22nd North American Fuzzy Information Processing Society* (July 24-26), Chicago, Illinois, 302-306.
 72. Murphy, P. M. and Aha, D. W., 1992, UCI repository of machine learning databases, www.ics.uci.edu/~mlearn/METARepository.html.
 73. MATLAB. User's Guide. The MathWorks, Inc., Natick, MA 01760, 1994-2001. <http://www.mathworks.com>.
 74. Marchand, M., Shawe-Taylor, J., 2002, The set covering machine, *Journal of Machine Learning Research*, 3, 723-746.
 75. Odewahn, S., Stockwell, E., Pennington, R. L., Humphreys, R. M. and Zumach, W. A., 1992, Automated star/galaxy discrimination with neural networks, *Astronomical Journal*, 103, 1, 318-331.
 76. Platt, J. C., Cristianini, N. and Shawe-Taylor, J., 2000, Large margin DAG's for multiclass classification, *Advances in Neural Information Processing Systems*, MA: MIT Press, Cambridge, 12, 547-553.

77. Poggio, T., Torre, V. and Koch, C., 1985, Computational vision and regularization theory, *Nature*, 317, 314-319.
78. Poggio, T. and Girosi, F., 1990, Networks for approximation and learning, *Proceedings of the IEEE*, 78, 10, 1481-1497.
79. Potra, F. A. and Liu, X., 2004, Pattern separation and prediction via linear and semidefinite programming with application to cancer research, Technical Report (January), UMBC.
80. Ratsch, G., Mika, S., Schölkopf, B. and Muller, K.-R., 2003, Constructing boosting algorithms from SVMs: an application to one-class classification, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24, 9, 1184-1199.
81. Reilly, D. L., Cooper, L. N. and Elbaum, C., 1982, A neural model for category learning, *Biological Cybernetics*, 45, 35-41.
82. Rifkin, R. M, 2002, Everything old is new again: a fresh look at historical approaches in machine learning, PhD. Thesis, Massachusetts Institute of Technology.
83. Roger, J., Sun, C.-T. and Mizutani, E., 1994, *Neuro-fuzzy and Soft Computing*. Prentice Hall,
84. Rosen, J. B., 1965, Pattern separation by convex programming, *Journal of Mathematical Analysis and Applications*, 10, 123-134.
85. Sahbi, H. and Boujemaa, N., 2005, Validity of fuzzy clustering using entropy regularization, *Proceedings of the IEEE International Conference on Fuzzy Systems* (May 22-25), Reno, USA, 177- 182
86. Schlesinger, M. I. and Hlavac, V., 2002, *Ten Lectures on the Statistical and Structural Pattern Recognition*. Kluwer Academic Publishers.
87. Schölkopf, B., Williamson, R., Smola, A. and Platt, J., 2000, Support vector method for novelty detection, *Advances in Neural Information Processing Systems*, 12, 582-588.
88. Schölkopf, B. and Smola, A. J., 2002, *Learning With Kernels: Support Vector Machines, Regularization and Beyond*. The MIT Press, Cambridge.
89. Smola, A. and Schölkopf, B., 1998, A tutorial on support vector regression, Technical Report NC2-TR-1998-030, ESPRIT Working Group in Neural and Computational Learning II (October).
90. Smola, A., Schölkopf, B. and Müller, K.-R., 1998, The connection between regularization operators and support vector kernels, *Neural Networks*, 11, 4, 637-649.
91. Smola, A. and Schölkopf, B., 1998, From regularization operators to support vector kernels, *Advances in Neural Information Processing Systems*, San Mateo, CA, 10, 343-349.

92. Sturm, J. F., 1999, Using Sedumi 1.02, A Matlab toolbox for optimization over symmetric cones. Optimization Methods and Software, Special issue on Interior Point Methods (CD supplement with software), 11-12, 625-653.
93. Suykens, J., Gestel, T. V., Brabenter, J. D., Moor, B. D. and Vandewalle, J., 2002, *Least Squares Support Vector Machines*. World Scientific, New Jersey.
94. Sun, P., 2004, Computation of minimum volume covering ellipsoids, Operations Research 52, 5, 690-706.
95. Tax, D., 2001, One class classification, Ph.D Thesis, Technische Universit t Delft.
96. Tax, D. and Duin, R., 1999, Support vector domain description, Pattern Recognition Letters, 20, 1191-1199.
97. Tsujinishi, D. and Abe, S., 2003, Fuzzy least squares support vector machines, Proceedings of the International Joint Conference on Neural Networks (July 20-24), Portland, Oregon, 1599-1604.
98. U ar, A., Demir, Y. and G zeli , C., 2003, Support vector machines as zero order and first order adaptive fuzzy inference systems and their applications on system identification, 10th International Fuzzy Systems Association World Congress-IFSA'03 (29 June 2-July), Istanbul Turkey, 421-424.
99. U ar, A., Demir, Y. and G zeli , C., 2003, Fuzzy model identification using support vector clustering method, Springer Verlag, Lecture Notes in Computer Science, 2714, 225-233.
100. U ar, A., Demir, Y. and G zeli , C., 2006, A New formulation for classification by ellipsoids, Springer-Verlag Berlin Heidelberg 2006, TAINN 2005, LNAI 3949, 100-106.
101. Vapnik, V. N., 1995, *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.
102. Vapnik, V. N., 1998, *Statistical Learning Theory*. John Wiley & Sons, New York.
103. Wahba, G., 1987, Three topics in ill-posed inverse problems, M. Engl and G. Groetsch, editors, Inverse and Ill-Posed Problems, Academic Press, Inc., 37-50.
104. Wang, J., Neskovic, P. and Cooper, L. N., 2005, Pattern classification via single spheres, Lecture Notes in Artificial Intelligence, 3735, 241-252.
105. Wang, L. X. and Mendel, J. M., 1992, Fuzzy basis functions, universal approximation, and orthogonal least squares learning, IEEE Trans. on Neural Networks, 3, 807-814.
106. Weston, J. and Watkins, C., 1999, Support vector machines for multi-class pattern recognition, Proceedings of the Seventh European Symposium on Artificial Neural Networks (April 21-23), Bruges.
107. Wu, K. and Yap, K.-H., 2006, Content-based image retrieval a pseudo-label support

- vector machine framework, IEEE Computational Intelligence Magazine (May), 1, 10-16.
108. Yildirim, E. A., 2001, An interior-point perspective on sensitivity analysis in linear programming and semidefinite programming, Ph.D. Thesis, Cornell University.
 109. Yee, P. and Haykin, S., 1999, A dynamic regularized radial basis function network for nonlinear nonstationary time series prediction, IEEE Trans. on Signal Processing, 47, 9, 2503-2521.
 110. Zhang, B., 2001, Is the maximal margin hyperplane special in a feature space? Technical Report, HP Laboratories Palo Alto.
 111. Zhang, L., Zhou, W. and Jiao, L., 2004, Hidden space support vector machines, IEEE Trans. on Neural Networks, 15, 6, 1424-1434.

EK-1. İSTATİSTİKSEL ÖĞRENME KURAMI

E1.1. Örneklerden Öğrenme Kavramı

Öğrenme, basit olarak verilen gözlemlerden (eğitim örneklerinden) bir model (bilgi) kurmayı ifade eder. Böyle bir modeli oluşturmak için verilen gözlemleri kullanarak, bilinmeyen gözlemler için modelin genellenmesi gerekir.

İstatistiksel öğrenme kuramında öğrenme, şu şekilde tanımlanmıştır: Elimizde bilinmeyen $P(z)$ olasılık dağılımından çıkarılan $Z^L = \{z^1, \dots, z^L\}$ eğitim örnekleri varsa, test örneklerini (yani, onlara ait olan etiketleri veya sistem çıkışlarını) tahmin eden $\ell(Z^L, w)$ işlevler kümesinden, en iyi $\ell(Z^L, w_L^*)$ işlevini belirleyen en uygun parametre kümesi w_L^* 'ı bulmaktır. Burada; $w \in W$, $\ell(Z^L, w)$ işlevler kümesinin parametreleri ve W ise tüm parametre kümelerini içerir.

Bu aşamada akla gelen ilk soru, en uygun işlevin ve parametre kümesinin nasıl ve hangi kriterlere göre belirleneceğidir. İstatistiksel öğrenme kuramında, en uygun işlev ve parametre kümesi (E1.1) ile verilen risk işlevi enazlanarak elde edilir.

$$R(w) = \int f(z, w) dP(z) \quad w \in W. \quad (E1.1)$$

Burada; $f(z, w)$ kayıp işlevler kümesidir ve uygulama türüne göre farklı bir işlev ile gösterilir. Aşağıda üç yaygın öğrenme problemi olan örüntü tanıma, regresyon ve yoğunluk kestirimi için kayıp işlevleri tanımlanmıştır.

Örüntü Tanımada Kayıp işlevi

Bu durumda veri, $z = (x, y)$, $x \in \mathcal{R}^n$ giriş ve çıkış veri çiftiyle verilir. Öğrenme sisteminin çıkışı, y ve gösterge işlevler kümesi $\ell(x, w)$, sadece sıfır ve bir değerini alır. Dolayısıyla çıkış ve işlev kümesinin değeri birbirinden farklı ise sınıflandırma hatalı ve bir değerini alır, aksi durumda sıfırdır. Öğrenme problemi, olasılık dağılımı $P(x, y)$ bilinmediği ve sadece eğitim örnekleri verildiği zaman sınıflandırma hata olasılığını enazlayan bir işlev bulmaya dönüşür. Bu nedenle; uygun bir kayıp işlevi aşağıdaki gibi seçilebilir,

$$L(y, \ell(x, w)) = \begin{cases} 0 & y = \ell(x, w) \\ 1 & y \neq \ell(x, w) \end{cases} \quad (E1.2)$$

Regresyon Kestiriminde Kayıp işlevi

Bu durumda, y çıkış değerleri ve $x \in \mathcal{R}^n, w \in W$ olmak üzere $\ell(x, w)$ işlevler kümesinin değerleri gerçel değerlerden oluşur. Dolayısıyla sınıflandırma hatası da gerçel değerler olur. Örneğin, $[0, +\infty)$ aralığında değerler alan uygun bir kayıp işlevi aşağıdaki gibi seçilebilir,

$$f(y, \ell(x, w)) = (y - \ell(x, w))^2. \quad (E1.3)$$

Yoğunluk Kestiriminde Kayıp işlevi

Yoğunluk kestirimi durumunda, $x \in \mathcal{R}^n, w \in W$ olmak üzere $p(x, w)$ yoğunluklarının bir kümesi hesaplanır. $(-\infty, +\infty)$ aralığında değerler alan uygun bir kayıp işlevi aşağıdaki gibi seçilebilir,

$$f(p(x, w)) = -\log p(x, w). \quad (E1.4)$$

E1.1.2. Deneysel Riskin Enazlanması

Klasik olarak, eğitim anında sadece mevcut eğitim kümesi üzerindeki hata enazlanır. En uygun işlevi seçmek için bu ilke, deneysel riskin enazlanması (DRE) olarak tanımlanır. Eğer bu ilke sadece eğitim verisi kullanarak anılırsa, risk işlevi

$$R_{dny}(w) = \frac{1}{L} \sum_{i=1}^L f(z_i, w) \quad (E1.5)$$

ile yer değiştirir. Kısaca deneysel riski enazlayan $f(z^i, w_L)$ ile en iyi kayıp işlevi $f(z^i, w^*)$ yaklaşılır.

Bu ilkenin gelecek test örnekleri üzerinde en az hataya götürdüğü varsayılır. Fakat bu genellikle doğru değildir. Hesapsal öğrenmede en temel problemlerden birisi, deneysel ve gerçek risk arasındaki ödünleşimin nasıl olacağıdır. Bu problemin çözümü farklı öğrenme kuramları ve öğrenme algoritmaları ile farklı tarzlarda yapılabilir. Örneğin; düzenlileştirme (cezalandırma), RTIA'larla öğrenme için; erken durdurma, geriye yayımlı YSA'larla öğrenme için; bayes, model seçiminde bayes tabanlı ağlar ile öğrenme için ve yapısal risk enazlanması, DVM'lerle öğrenme için genelleme başarımını göz önüne alan yöntemler olarak gösterilebilir.

E1.2. Öğrenmeye Felsefi Yaklaşım

Öğrenme kavramının esası, genelleme adımı ve bu adımın nasıl uygulanabileceğidir. “Doğru bilgi (model) nasıl oluşturulur?” sorusunun yanıtı felsefede çok eski bir problemdir. Büyük filozof Immanuel Kant 18. yüzyılda bu soruyu daha keskin bir biçimde ortaya koymuş, “Doğru ve yanlış bilgi arasındaki fark nasıl görülür?” veya “Gözlemlerden genelleme adımı doğru olarak ne zaman uygulanır?” sorularını sormuştur. Bu problem felsefede “sınır çizme” problemi olarak bilinir. Filozof Karl Popper 1934’te bu problem için bir çözüm önermiştir. Öneri, kuram veya modelin ancak ve ancak kuram veya model çürütülebilir ise bilimsel olduğudur. Eğer hiçbir gözlem kuram veya modeli saptırmazsa, o zaman bu kuram bilimsel kabul edilemez [33, 101].

Bu ilkeye bir örnek olarak, astronomiye karşı astrolojinin bilimsel bir disiplin olup olmadığı ele alınabilir. Astronomi her biri çürütülebilir birçok sava açık modele sahiptir. Astronomideki tüm bilimsel modeller, modellerin geçerliliğini koruması için kanıtlanabilecek deney ve gözlemlere dayanır. Diğer yandan astroloji böyle savları içermez. Astrolojide bir model veya savı belirleyen herhangi bir olası deney ve gözlem yoktur. Kuramın yanlış olduğunu tanıtlama olanağı yoktur ve bundan dolayı bu tanımdaki astroloji bilimsel bir kuram değildir.

Aynı ilke istatistiksel öğrenme kuramında benzer bir analojiye sahiptir. Çürütülebilirlik, öğrenme algoritmasında sınırlı öğrenme kapasitesine karşılık gelir. Sistem ancak ve ancak sınırlı öğrenme kapasitesine sahipse ve bundan dolayı hatalar yapma potansiyeline sahipse, doğru olarak öğrenebilir. Vapnik’in önerdiği gibi, sistemin öğrenme kapasitesi Vapnik-Chervonenkis (VC) boyutu ile nitelenir ve VC boyutunun sınırlı öğrenme kapasitesi için sınırlandırılması gerekir.

İstatistiksel öğrenme kuramının amacı, öğrenme sisteminin en iyi öğrenme kapasitesinin belirlenmesi için analitik olarak motive edilmiş otomatik bir prosedür belirlemektir. Bu prosedür aşağıda verilen dört temel soru çözülerek oluşturulur:

- (i) Tutarlılık: DRE ilkesi temeline dayanan öğrenme işleminin tutarlılığı için gerekli şartlar nelerdir?
- (ii) Yakınsama: Öğrenme işleminin yakınsama oranı ne kadar hızlıdır?
- (iii) Genelleme: Öğrenme işleminin yakınsama oranı nasıl kontrol edilebilir?
- (iv) Algoritma: Genelleme yeteneğini otomatik olarak kontrol eden bir öğrenme algoritması nasıl kurulur?

Aşağıdaki bölümlerde bu sorular ile ilgili istatistiksel öğrenme kuramının sonuçları, temel tanımlar da verilerek özetlenmiştir [101].

E1.3. Temel Niceliklerin Tanımı

İkili sınıflandırma problemi için, etiketleri 1 ve -1 olan ve $Z^L = \{z^1, \dots, z^L\}$ ile gösterilen eğitim kümesindeki veriler, farklı şekillerde ayrılabilir ve tüm olası ayırımla sayısı $N^W(Z^L)$ olarak verilsin. Örnek olarak, L tane az sayıdaki nokta çok boyutlu düzlemler ile ayrıştırmak istenirse, tüm olası farklı şekillerde yani, $N^W(Z^L) = 2^L$ kadar farklı düzlem ile bu noktalar ayrılabilir [101]. Bununla birlikte L nokta sayısı büyürken, $N^W(Z^L) < 2^L$ olacaktır [102].

Kuramın sonuçlarını açıklamak için, aşağıdaki nicelikler kısaca tanımlanmıştır.

Verilen veri üzerindeki işlevler kümesinin farklılığının (çeşitliliğinin) beklentisi olarak bilinen VC-entropisi:

$$H^W(L) = E \ln N^W(Z^L). \quad (E1.6)$$

Burada; E , Z^L olasılık dağılımı üzerindedir.

Büyüme (Growth) işlevi,

$$G^W(L) = \ln \sup_{Z^L} N^W(Z^L). \quad (E1.7)$$

Bu nicelik dağılımdan bağımsızdır. Büyüme işlevi ve VC entropisi arasında (E1.8) ile verilen bağlantı vardır,

$$H^W(L) \leq G^W(L). \quad (E1.8)$$

E1.3.1. Öğrenme İşleminin Tutarlılığı

En uygun w_L^* bulunduğu andaki kayıp işlevi $L(z, w_L^*)$, bu kayıp işlevi ile üretilen en uygun deneysel riski $R_{dny}(w_L^*)$ ve aynı kayıp işlevi için gerçek (bilinmeyen) risk $R(w_L^*)$ ile tanımlanırsa, Şekil E1.1'e bakarak ve (E1.9) ve (E1.10) bağıntılarını kullanarak, deneysel risk ilkesi tabanlı öğrenme işlemi için tutarlılığı; $P(z)$ olasılık dağılımı ile örneklerin sayısı sonsuza giderken, gerçek risk ve deneysel riskin aynı limite yakınsaması olarak tanımlayabiliriz

$$R(w_L^*) \xrightarrow[L \rightarrow \infty]{P} \inf_{w \in W} R(w) \rightarrow R(w_L^*), \quad (E1.9)$$

$$R_{dny}(w_L^*) \xrightarrow[L \rightarrow \infty]{P} \inf_{w \in W} R(w) \rightarrow R(w_L^*). \quad (E1.10)$$

1968'de Vapnik ve Chervonenkis Z^L dağılımına bağlı olarak öğrenme işleminin tutarlılığını kanıtlamıştır [101]. Bu çalışmaya göre, bir öğrenme işlemi eğer (E1.11)'deki şart sağlanırsa tutarlıdır,

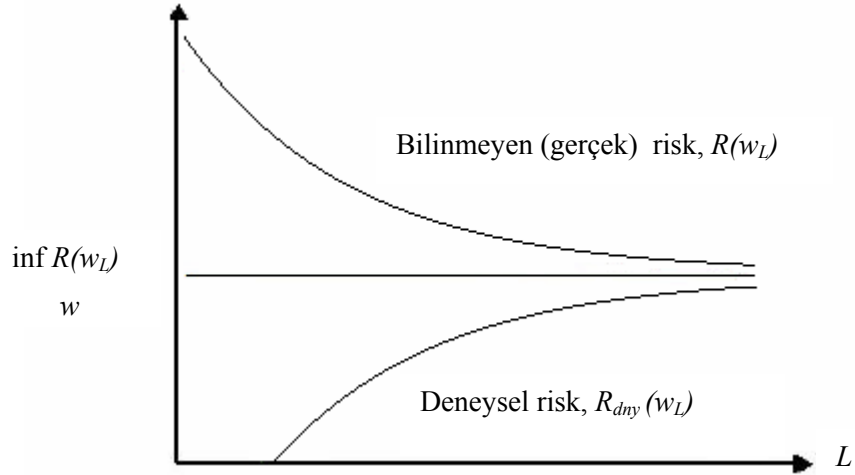
$$\lim_{L \rightarrow \infty} \frac{H^{\omega}(L)}{L} = 0. \quad (E1.11)$$

En son sonuç, gerçekte bilinmeyen $H^w(L)$ 'den dolayı eğitim örneklerinin dağılımına bağlıdır. Bundan dolayı da pratikte çok kullanılabilir bir sonuç değildir. Bununla birlikte 1989'da Vapnik ve Chervonenkis, DRE'nin tek biçimli (uniform) yakınsamasının öğrenme işleminin tutarlılığı için gerekli ve yeterli olduğunu ifade eden teoremlerini kanıtlamışlardır [102],

$$\lim_{L \rightarrow \infty} P\left[\inf_{\omega} (R(\omega) - R_{dny}(\omega)) > \epsilon\right] = 0, \quad \forall \epsilon > 0 \quad (E1.12)$$

Bunun bir sonucu olarak DRE'nin yakınsaması için (E1.13)'teki gerekli ve yeterli şart türetilmiştir,

$$\lim_{L \rightarrow \infty} \frac{G^{\omega}(L)}{L} = 0. \quad (E1.13)$$



Şekil E1.1 Öğrenme işlemi eğer beklenen risk $R(w_L)$ ve deneysel risk $R_{dny}(w_L)$ en küçük olası risk $\inf_{w \in W} R(w)$ 'e yakınsarsa tutarlıdır.

E1.3.2. Öğrenme İşleminin Yakınsama Hız Oranı

Eğer bir üstel oran seçersek, herhangi bir ϵ ve $c > 0$ sabiti için (E1.14)'ün sağlanması gerekir,

$$P\{R(w_L) - R(w_o) > \epsilon\} < e^{-c\epsilon^{2L}}. \quad (E1.14)$$

Vapnik, aynı eşitliğin hızlı yakınsamada gerekli ve yeterli şart olarak tutarlık için

$$\lim_{L \rightarrow \infty} \frac{G^{\omega}(L)}{L} = 0 \quad (E1.15)$$

sağlanmasının zorunlu olduğunu kanıtlar [101].

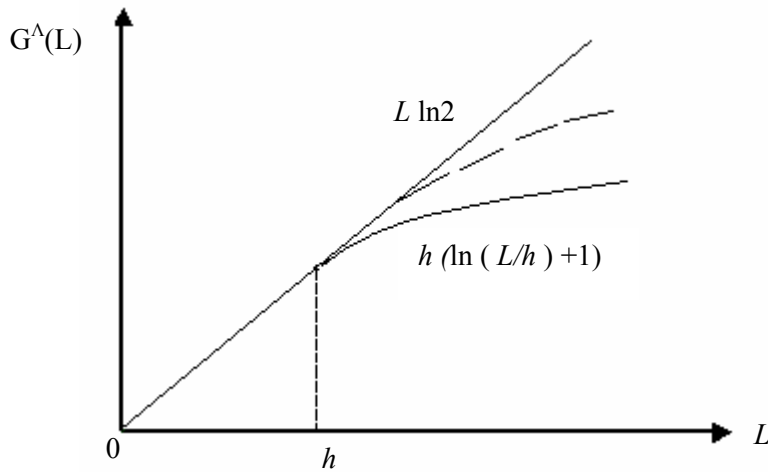
Özetle, sadece en son şartı sağlayarak yaklaştırılan işlevler kümesi, daha fazla eğitim örneği uygulandığı zaman, yeterince hızlı olarak riski enazlayan ve tutarlı olan öğrenme işlemine götürecektir

E1.3.3. VC-boyutu

[101]'de büyüme işlevinin daima (E1.16)'daki biçimde olduğu gösterilmiştir.

$$G^w(L) \begin{cases} = L \ln 2 \\ \text{veya} \\ \leq h \left(1 + \ln \frac{1}{h} \right) \end{cases} \quad (E1.16)$$

Burada; h , VC-boyutu olarak adlandırılır. Eğer $h \neq \infty$ ise, bir önceki bölümde anlatılanlar temelinde tutarlıdır ve öğrenme işleminin yakınsaması garanti edilir.



Şekil E1.2 Büyüme işlevi ya doğrusaldır yada logaritmik bir işlevle sınırlanır. Kesikli eğri gibi davranmaz.

Örnek işlevler kümesi ve VC-boyutları

n boyutlu uzay için doğrusal gösterge işlevlerin VC-boyutu " $h=n+1$ " olur. Örneğin, " $\mathcal{R}^2$ 'de kaç nokta, etiketleri ne olursa olsun kusursuz bir şekilde ayrılabilir?" sorusunun cevabı; 3 nokta, 2^3 farklı şekilde birbirinden ayrılabilir, fakat 4 nokta ayrılamaz, örnek olarak XOR doğrusal bir işlev ile ayrılamaz. $\mathcal{R}^2$ 'deki doğruların VC boyutu 3'tür.

VC-boyutu basit olarak serbest parametre sayısını hesaplamaktan ziyade, farklı bir ölçüdür. Örneğin, aşağıdaki işlev için,

$$\ell(x, w) = \theta(\sin(wx)), \quad w \in \Re \quad (E1.17)$$

Burada; $\theta(x)=1$, $\forall x > 0$; $\theta(x)=-1$, $\forall x \leq 0$, L ne kadar büyük olursa olsun $x^i = 10^{-i}$, $i=1, \dots, L$.

$w = \pi \left(1 + \sum_{i=1}^L \frac{(1-y^i)10^i}{2} \right)$ seçildiği zaman, her bir olası etiket için, $y^1, y^2, \dots, y^L$, $y^i \in \{-1, 1\}$, bir $\ell(x, w)$ vardır.

Dolayısıyla serbest parametre olarak bir w parametresi görünmesine rağmen, VC-boyutu sonsuzdur. Bu örnek, yüksek frekanslı $\sin(wx)$ kullanarak uygun sayıda nokta için, $(-1, +1)$ ile sınırlanan herhangi bir işlevin uygun değerlerinin yaklaştırılabildiğini gösterir [101].

Diğer bir örnek olarak, YSA yazınındaki bilinen işlevler kümesinden biri olan radyal tabanlı işlevi alalım. Bu işlevin parametre kümesinin $w = (c, \sigma, t)$ olduğu düşünülürse, K radyal taban işlevi

$$f(x, c, \sigma, t) = \theta \left(K \left(\frac{\|x - c\|}{\sigma} \right) - t \right) \quad (E1.18)$$

ile verilir. İşlevlerin bu kümesinin VC boyutu $h=n+1$ iken, (E1.18)'deki işlevler kümesinin serbest parametreleri $n+2$ 'dir.

E1.3.4. Öğrenme İşleminin Genelleme Yeteneği

h VC-boyutu verildiği zaman, gelecek örnekler üzerinde beklenen risk, (E1.19)'daki gibi sınırlanır:

$$R(\omega_L) \leq R_{dny}(\omega_L) + \phi \left(\frac{L}{h} \right) \quad (E1.19)$$

Genelleme başarımı, deneysel risk ve yaklaşım işlevlerinin karmaşıklığına bağlı VC-güven olarak adlandırılan terim ile sınırlanır.

(E1.19)'un tam ifadesi, (E1.20)'deki gibidir. En az $1 - \tau$ olasılıkla:

$$R(w_L) \leq R_{dny}(w_L) + \frac{\varepsilon}{2} \left(1 + \sqrt{1 + \frac{4R_{dny}(w_L)}{\varepsilon}} \right) \quad (E1.20)$$

burada

$$\varepsilon = 4 \frac{h \left(\ln \frac{2L}{h} + 1 \right) - \ln \left(\frac{\tau}{4} \right)}{L} \quad (E1.21)$$

ile hesaplanır. Verilen işlevler kümesi üzerinde en iyi gerçekleştirilebilir riske mesafe

$$\Delta(w_L) = R(w_L) - (w_o)$$

$$\Delta(w_L) \leq \psi\left(\frac{1}{h}\right)$$

olur, yada en az $1 - 2\tau$ olasılıkla

$$\Delta(w_L) \leq \frac{\varepsilon}{2} \left(1 + \sqrt{1 + \frac{4R_{dny}(w_L)}{\varepsilon}} \right) + \sqrt{\frac{-\ln \tau}{L}} \quad (E1.22)$$

tam ifadesi elde edilir.

Yapılan çalışmalar bu sınırların oldukça geniş ve herhangi bir dağılım için geçerli olduğunu belirtmesine rağmen, maalesef bu sınır bilinmeyen herhangi bir dağılım için analitik olarak bilinen en küçük sınırdır.

E1.3.5. Yapısal Riskin Enazlanması

Genellemede verilen en son sınırı düzeltmek (yani daraltmak) için deneysel risk azaltılabilir. Bunun dışında, belirli (sabitlenmiş) deneysel risk verildiği zaman, öğrenme işleminin VC-boyutu da enazaltılabilir. Bu yöntem, yapısal riskin enazlanması (YRE) ilkesi olarak bilinir. Bu ilke, belirli bir deneysel risk verildiğinde, en küçük VC-boyutu ile işlevler kümesini seçerek işlevler kümesinin yapısını enazlamak olarak tanımlanır.

L/h büyük ve τ küçük olduğu zaman, VC-güven küçük olur. Gerçek risk, deneysel riskin değerine yakın olur. Bu durumda deneysel riskin küçük değeri beklenen riskin küçük değerini garanti eder. Bununla birlikte L/h küçük ise, $R_{dny}(w_L)$ gerçek değer en küçük değerini garanti etmez. Bu durumda $R(w)$ gerçek riskini kontrol etmek için, (E1.20) eşitsizliğinin sağ tarafı her iki terim üzerinden eş zamanlı olarak enazlanır. Eşitsizliğinin sağ tarafındaki ilk terim işlevler kümesinin özel bir işlevlerine bağlı iken, ikinci terim tüm işlevler kümesinin VC boyutuna bağlıdır. (E1.20) risk sınırının sağ tarafını enazlamak için, VC boyutunu kontrol edilebilir bir değişken yapmak gerekir.

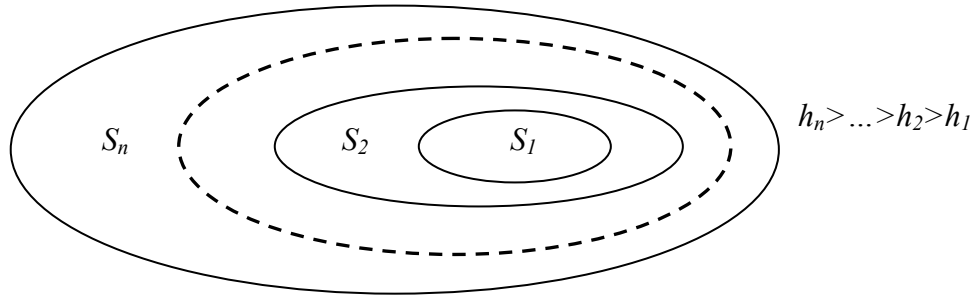
YRE ilkesi, hem deneysel risk hem de VC güvene göre risk işlevini enazlamayı amaçlar.

$\ell(x, w)$, $w \in W$ işlevlerin iç içe alt kümesinden oluşan S İşlevler kümesi

$$S_k = \{\ell(x, w), w \in W_k\} \text{ ile tanımlanırsa,}$$

$$S_1 \subset S_2 \subset \dots \subset S_n \subset \dots,$$

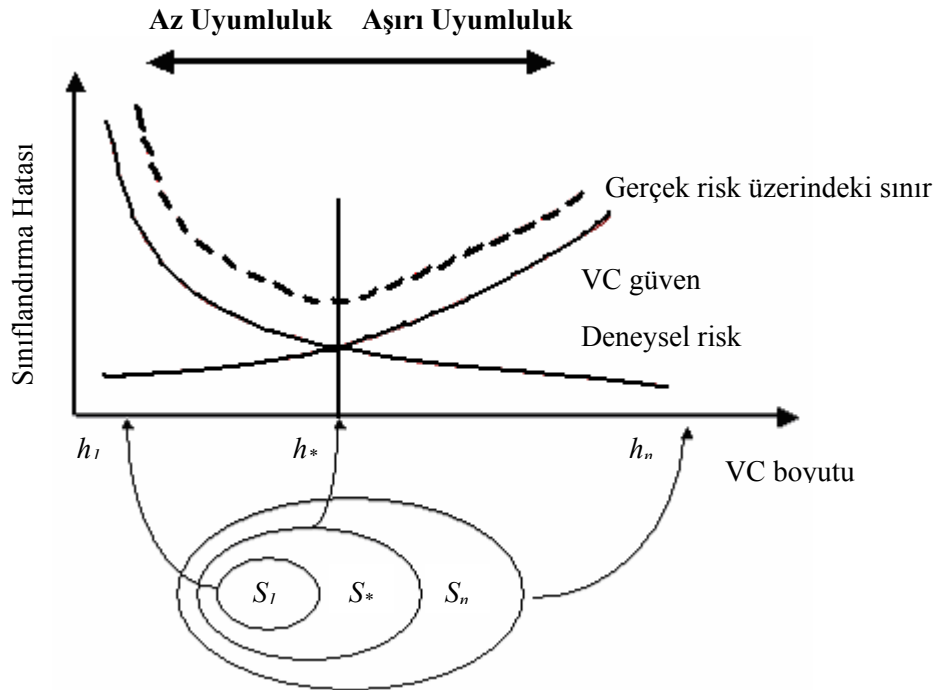
S_k 'nin her bir elemanı Şekil E1.3'te gösterildiği gibi $h_1 < h_2 < \dots < h_n < \dots$ sonlu boyutlu h_k VC boyutuna sahiptir.



Şekil E1.3 VC boyutu ile düzenlenen işlevlere ilişkin Venn diyagramı.

Verilen $x_1, \dots, x_L$ noktalarının verilen kümesi için, YRE ilkesi en küçük riski garanti etmek için S_k alt kümesindeki deneysel riski enazaltarak işlevi seçer.

YRE ilkesi, yaklaşım işlevlerin karmaşıklığı ve verilen verinin yaklaşım kalitesi arasında ödünleşimi tanımlar. Altküme indisi n artarken, deneysel riskin en küçüğü azalır, bununla birlikte güvenden sorumlu terim artar. YRE ilkesi S_* alt kümesini seçerken, Şekil E1.4'te gösterildiği gibi her iki faktörü de göz önüne alır. Böylece deneysel riski azaltarak gerçek risk üzerindeki en iyi sınır elde edilir.



Şekil E1.4 Risk üzerindeki sınır deneysel risk ve VC güvenin toplamıdır. VC güven artırılırken deneysel risk yapının sonlu elemanları ile azaltılır. Riskin en küçük sınırı yapının bazı uygun elemanları üzerinde sağlanır.

Bir öğrenme algoritması kurmak için, yaklaşım işlevler kümesinin h VC-boyutunu hesaplamak ve onu değiştirebilmek gerekir. Bu, birçok işlev gurupları için mümkün değildir, fakat paylı doğrusal işlevler için mümkündür. Bu özellik, DVM ile kullanılır.

EK2. DESTEK VEKTÖR MAKİNELERİ

E2.1. DVM'lere Giriş

DVM'ler başlangıçta sınıflama problemleri için geliştirilmiş, daha sonra regresyon kestirimi için sınıflamaya benzer olarak genişletilmiştir. Bundan dolayı, bu alt bölümde genellikle sınıflama üzerinden bazı bilgiler verilmiştir, regresyon kestirimi için de temel aynıdır.

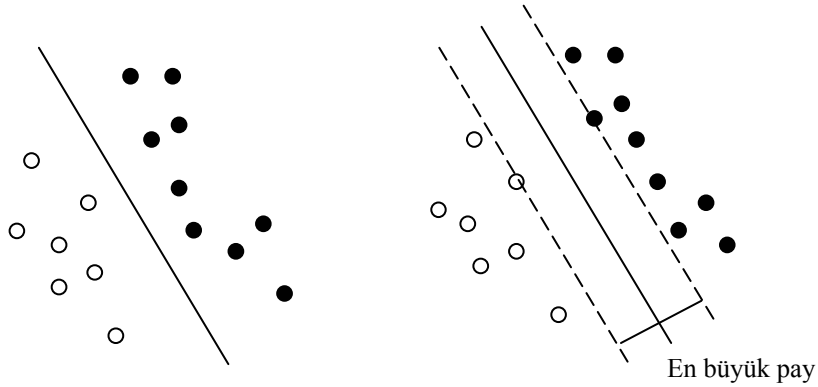
DVM'ler sınıflama yaparken; diğer öğrenme makinelerinin tersine hem yapısal hem de deneysel riski enazaltarak, test örnekleri üzerinde de eğitim örneklerinininkine yakın bir başarımla sağlar [EK-1]. Yapısal risk, yaklaşım işlevleri kümesinin VC-boyutu kontrol edilerek enazlanır. Yaklaşım işlevleri ise, (E2.1)'deki ayrıştırıcı çok boyutlu düzlemler ile verilir,

$$\ell(x) = w^T x + b. \quad (E2.1)$$

EK-1'de, n boyutlu uzayda düzlemler kümesinin VC boyutunun $n+1$ olduğu belirtilmişti. O halde, Şekil E2.1'deki pay içermeyen düzlemin VC boyutu örneğin, $n=10^{13}$ için oldukça büyüktür. Ancak Şekil E2.1'deki pay içeren düzlemlerin VC boyutu, Vapnik'in bir üst sınırı

$$h \leq \min\left(\frac{R^2}{\Delta^2}, n\right) + 1 \quad (E2.2)$$

kullanılarak $\approx 10^3$ 'e kadar azaltılabilir [101]. Burada; R verinin ait olduğu kürenin yarıçapı ve Δ çok boyutlu düzlemin payını göstermektedir. Bundan dolayı; çok boyutlu düzlemler kümesinin payını ençoklayarak, VC-boyutu enazlanabilir. Özetle, DVM'lerde iyi bir genelleme başarımı sağlamak için, belirli bir deneysel risk verildiği zaman, yarıçap pay oranı küçük olan, en uygun çok boyutlu düzlem aranır.



Şekil E2.1 Her iki çok boyutlu düzlem, eğitim örneklerini doğru olarak ayırır. Fakat sağ taraftaki en iyi ayrıştırıcı düzlem geniş paya sahiptir. Bundan dolayı daha düşük riske sahiptir; ayrıca sezgisel olarak bu eğitim kümesi gürültüye daha az hassastır [47].

E2.2. En Uygun Ayırıştırıcı Çok Boyutlu Düzlem

E2.2.1. Doğrusal Olarak Ayrılabilen Durumlar

İki sınıflı sınıflamada, $x \in \mathfrak{R}^n, y \in \{-1, 1\}$ olmak üzere $(x^1, y^1), \dots, (x^L, y^L)$ eğitim örneklerinin L örneği doğrusal olarak ayrılabiliriyorsa, karar sınırı olarak (E2.1)'de verilen çok boyutlu düzlem kullanılır. Çok boyutlu ayırıştırıcı düzlemin aşağıdaki kısıtları sağlaması gerekir [102],

$$w^T x^i + b \geq 1 \quad \text{için } y^i = 1 \text{ ise} \quad (\text{E2.3})$$

$$w^T x^i + b \leq -1 \quad \text{için } y^i = -1 \text{ ise, } i = 1, \dots, L \quad (\text{E2.4})$$

veya bu eşitsizlikler için kapalı biçimde

$$y^i [w^T x^i + b] \geq 1, \quad i = 1, \dots, L \quad (\text{E2.5})$$

olarak gösterilir.

Oluşturulacak modelin genelleme hatasını azaltmak için Δ payını ençoklamak gerekir. Bunun için öncelikle çok boyutlu düzleme olan uzaklık,

$$y \frac{|\ell(x)|}{\|w\|} \quad (\text{E2.6})$$

olarak tanımlanır [93, 102]. Eğer Δ payı kullanılarak ayırıştırıcı düzlemler oluşturuluyorsa, tüm eğitim verileri için (E2.7) sağlanır,

$$y^i \frac{|\ell(x^i)|}{\|w\|} \geq \Delta, \quad i = 1, \dots, L. \quad (\text{E2.7})$$

Bundan dolayı, buradaki problem Δ payını ençoklayan w 'nin bulunmasıdır. Görüleceği gibi, (E2.7)'nin çözümünden, sonsuz sayıda ayırıştırıcı düzleme ve dolayısıyla sonsuz sayıda w 'ye ulaşılabilir. En uygun, tek bir çözüm elde etmek için w ölçeklenir. (E2.8)'deki gibi ölçeklenmiş çözümler göz önüne alınarak çözüm sayısı sınırlanır [102],

$$\Delta \|w\| = 1. \quad (\text{E2.8})$$

Buna benzer çok boyutlu düzlemler doğal biçimli düzlemler olarak tanımlanır.

Δ payını ençoklama, $\|w\|$ 'nin azalanmasına eşittir. Bundan dolayı en uygun çok boyutlu düzlem, (E2.5) kısıtlarını sağlayan ve (E2.9)'u enazlayan eniyileme problemi ile elde edilir,

$$\frac{1}{2} \|w\|^2. \quad (\text{E2.9})$$

Burada, $\frac{1}{2}$ katsayısı ve kare işlemi (E2.10)'daki hesaplamalarda uygunluk için kullanılır. Bu

eniyileme problemini çözmek için Lagrange çarpanları yöntemi kullanılarak, (E2.9)'daki kısıtlı eniyileme problemi, (E2.10)'daki kısıtsız eniyileme problemi olarak tekrar yazılır,

$$L(w, b, \lambda) = \frac{1}{2} (w^T w) - \sum \lambda_i \{y^i (w^T x^i + b) - 1\}. \quad (E2.10)$$

Burada; $\lambda_i \geq 0$ Lagrange çarpanlarını göstermektedir. Problemin çözümü için $L(w, b, \lambda)$ işlevinin eğer noktasının bulunması gerekir. Bu amaçla, $L(w, b, \lambda)$ işlevi $\lambda_i \geq 0$ 'a göre ençoklanır ve w ve b 'ye göre enazlanır.

Eğer noktasındaki çözüm (w^*, b^*, λ^*) , Karush-Kuhn-Tucker (KKT) teoremi olarak bilinen aşağıdaki şartları sağlamak zorundadır [6, 11]

$$\frac{\partial L(w^*, b^*, \lambda^*)}{\partial w} = 0, \quad (E2.11)$$

$$\frac{\partial L(w^*, b^*, \lambda^*)}{\partial b} = 0. \quad (E2.12)$$

(E2.11) ve (E2.12)'nin çözümü sonucunda; Lagrange çarpanları üzerinde (E2.13)'teki kısıt elde edilir ve problem fazla değişkenli olmaktan çıkarak, w parametresi (E2.14)'deki gibi sadece λ_i yardımıyla ifade edilir.

$$\sum_{i=0}^L \lambda_i^* y^i = 0, \quad \lambda_i^* \geq 0, \quad i = 1, \dots, L. \quad (E2.13)$$

$$w^* = \sum_{i=0}^L \lambda_i^* y^i x^i, \quad \lambda_i^* \geq 0, \quad i = 1, \dots, L. \quad (E2.14)$$

(E2.5)'teki eşitsizliğin, karar sınırlarının üzerindeki noktalarda eşitlik haline geldiği açıktır. Şimdi ise Kuhn-Tucker Teoremini de kullanarak, bu noktalarda herhangi bir λ_i^* 'nin sıfırdan farklı olacağı sonucu çıkartılır,

$$\lambda_i^* [y^i (w^T x^i + b) - 1] = 0, \quad i = 1, \dots, L. \quad (E2.15)$$

Böylece, DVM'lere adını veren $\lambda_i^* > 0$ 'a karşılık gelen ve destek vektör olarak adlandırılan x_i değerleri hesaplanabilir. Eğer (E2.10)

$$L(w, b, \lambda) = \frac{1}{2} (w^T w) - \sum_{i=1}^L \lambda_i y^i w^T x^i - b \sum_{i=1}^L \lambda_i y^i + \sum_{i=1}^L \lambda_i \quad (E2.16)$$

olarak tekrar yazılır ve sadece λ_i 'nin bir işlevi olarak Lagrangian işlevini ifade etmek için (E2.13) ve (E2.14)'deki şartlar yerine konulursa, eniyileme probleminin ikincil biçimi

$$L_{ikincil}(\lambda) = -\frac{1}{2} \sum_{i,j=1}^L \lambda_i \lambda_j y^i y^j \left((x^i)^T x^j \right) + \sum_{i=1}^L \lambda_i \quad (E2.17)$$

$$\text{kısıtları: } \sum_{i=1}^L \lambda_i y^i = 0 \quad \lambda_i \geq 0, i=1, \dots, L \quad (\text{E2.18})$$

olarak elde edilir.

(E2.17) ve (E2.18) ile verilen ikincil eniyileme problemi $\lambda_1, \dots, \lambda_L$ parametrelerine göre ençoklanarak çözülür. Sıfır olmayan Lagrange çarpanları elde edildiğinde, sonuç ayrıştırma işlevi

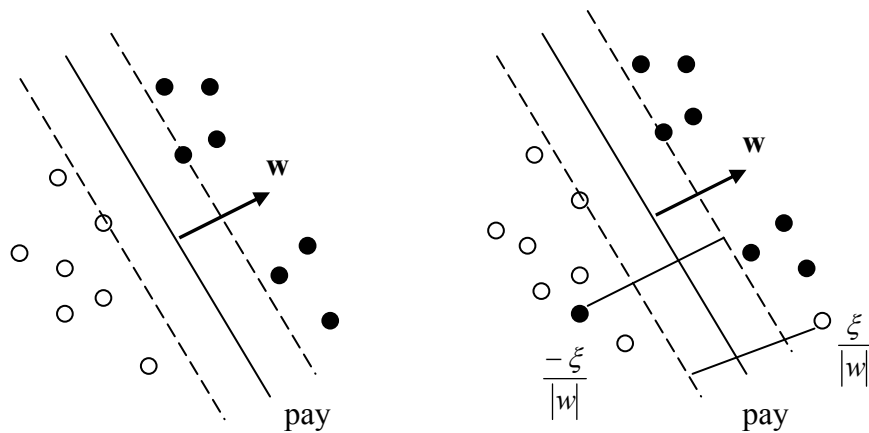
$$\ell(x) = \sum_{i=1}^L \lambda_i^* y^i (x^T x^i) + b \quad (\text{E2.19})$$

olarak bulunur.

İkincil eniyileme probleminin çözümü, hiçbir eşitsizlik kısıtı içermediği için (E2.10)'daki birincil biçimin çözümünden daha kolaydır. Ayrıca diğer bir avantajı, (E2.18)'de veri sadece iç çarpımın bir bölümü olarak ifade edilebildiği için, verinin iç çarpımının bilinmesi yada hesaplanması halinde işlemlerde doğrudan veriyi kullanmak yerine iç çarpımdan oluşan kapalı formun kullanılabilmesidir. Bu ilginç gerçek, çekirdek yöntemlerin kullanımını getirmiştir. Böylece doğrusal DVM'ler ilave hesapsal bedel ödmeden doğrusal olmayan duruma genişletilmiştir [22, 102].

E2.2.2. Doğrusal Olarak Ayrılamayan Durumlar

Veriler her zaman doğrusal olarak ayrılamayabilir. Hatta veri doğrusal olarak ayrılabilir bile, noktaların bazıları pay içerisinde bulunursa çoğu zaman daha geniş bir pay ve bundan dolayı daha düşük bir yapısal risk elde edilebilir.



Şekil E2.2 Sol taraftaki doğrusal olarak ayrılabilir problemde, en iyi ayrıştırıcı çok boyutlu düzlem $\|w\|$ enazlanarak elde edilir. Tek destek vektör en iyi ayrıştırıcı düzlemi tanımlamak için yeterlidir. Sağ taraftaki doğrusal olarak ayrılamayan durumda, en iyi ayrıştırıcı düzlem yapay değişkenler tanıtılarak sağlanır [19].

Payın içerisinde düşen veya karar sınırının hatalı tarafında bulunan veri noktaları için genelleme yapılırsa, (E2.5) kısıtlarını gevşetmek için Şekil E2.2'deki gibi $\xi_1, \dots, \xi_L$ yapay değişkenler tanımlanır:

$$y^i [w^T x^i + b] \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (\text{E2.20})$$

Bu durumda en büyük pay ile çok boyutlu düzlemi bulmak için, (E2.9)'a yapay değişkenlerin toplamı eklenir

$$\frac{1}{2} \|w\|^2 + \frac{C}{L} \sum_{i=1}^L \xi_i \quad (\text{E2.21})$$

ve (E2.20)'deki kısıt üzerinden enazlanarak çözüme tekrar başlanır. Burada; C parametresi öğrenme işlevlerinin karmaşıklığı ve izin verilen ayrıştırılamayan (veya pay içindeki) eğitim örneklerinin oranı arasındaki ödünleşimi niteler

Bu eniyileme problemi ikincil forma dönüştürülürse, aşağıdaki Lagrangian işlevi ençoklanarak çözülür,

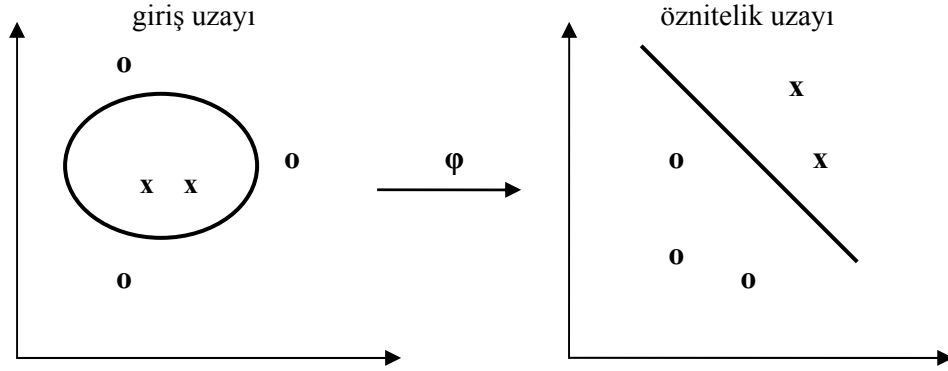
$$L_{ikincil}(\lambda) = -\frac{1}{2} \sum_{i,j=1}^L \lambda_i \lambda_j y^i y^j \left((x^i)^T x^j \right) + \sum_{i=1}^L \lambda_i \quad (\text{E2.22})$$

$$\text{kısıtları: } \sum_{i=1}^L y^i \lambda_i = 0, \quad 0 \leq \lambda_i \leq \frac{C}{L}, \quad i = 1, \dots, L. \quad (\text{E2.23})$$

E2.3. Çekirdek Tabanlı DVM'ler

Şekil E2.3'te görüldüğü gibi veri, asıl uzayda doğrusal olarak ayrıştırılamaz ise, bir çekirdek yardımıyla yüksek boyutlu uzaya örtük olarak taşınarak, o uzayda doğrusal olarak ayrıştırılabilir [101]. Böylece yüksek boyutlu öz nitelik uzayındaki çok boyutlu ayrıştırıcı düzlem, asıl uzayda doğrusal olmayan bir işleve karşılık gelir.

Çekirdekler kullanılarak elde edilen büyüklükler, doğrudan her bir eğitim verisinin işlevi olarak verilmeyip, eğitim verisinin iç çarpımlarının işlevi olarak verilebilir. İç çarpım çekirdekleri, ilk olarak 1964'te Aizerman tarafından çok güzel bir araç olarak bulunmuş, daha sonra doğrusal olmayan DVM'lerin çözümünde tekrar keşfedilmiştir [102].



Şekil E2.3 Giriş uzayının doğrusal sınıflamanın mümkün olduğu çok boyutlu uzaya dönüşümü [89].

Yüksek boyutlu uzaya dönüşüm için, ilk olarak x verisi H yüksek boyutlu uzaya dönüştürülür,

$$\mathcal{R}^d \rightarrow H, x \rightarrow \phi(x). \quad (\text{E2.24})$$

Burada; yüksek boyutlu uzay H bir Hilbert uzayıdır ve sonsuz boyutludur. Sonlu boyutlu durumda, H bir Euclidean uzayı olarak düşünülür.

Çekirdek, genellikle K ile gösterilir ve

$$K(x, y) = \phi(x)^T \phi(y) \quad (\text{E2.25})$$

olarak tanımlanır. H Hilbert uzayında iç çarpım ile $\psi(\phi(x)^T \phi(y))$ niceliği hesaplanmak istenirse, düşük boyutlu giriş uzayında çekirdek hesaplaması yeterlidir. Pratik olarak verinin yüksek boyutlu uzaya dönüşüm işlevini belirlemek için, (E2.26)'daki gibi iç çarpım çekirdeğinin hesaplamasından başlanır ve elde edilen çekirdekte yüksek boyutlu dönüşüm işlevi çıkartılabilir. Örnek olarak üçüncü dereceden polinomsal çekirdek $K(x, y) = (x^T y)^3$ ve uzayın asıl boyutluluğu 2 alınırsa, $x = (x_1, x_2)$ bu çekirdek örtük olarak dört boyutlu uzaya dönüşür:

$$\begin{aligned} K(x, y) &= (x^T y)^3 \\ &= (x_1^3 y_1^3 + 3x_1^2 y_1^2 x_2 y_2 + 3x_2^2 y_2^2 x_1 y_1 + x_2^3 y_2^3) \\ &= (x_1^3, 3x_1^2 x_2, 3x_2^2 x_1, x_2^3) (y_1^3, 3y_1^2 y_2, 3y_2^2 y_1, y_2^3) \\ &= \phi(x)^T \phi(y) \end{aligned} \quad (\text{E2.26})$$

Bu dönüşümün ve H 'nin tek olmadığına dikkat edilmesi gerekir. Diğer dönüşümler de aynı şekilde geçerlidir, örneğin aynı çekirdek için

$$\phi(x) = \left(\frac{1}{\sqrt{2}} (x_1^3 - x_2^3), \sqrt{3} x_1^2 x_2, \sqrt{3} x_2^2 x_1, \frac{1}{\sqrt{2}} (x_1^3 + x_2^3) \right) \quad (\text{E2.27})$$

dönüşümü seçilebilir.

Hafifçe farklı bir çekirdek $K(x, y) = (x^T y + 1)^3$ ve aynı iki boyutlu uzay için, örtük 16 boyutlu bir dönüşüm bulunabilir. Bununla birlikte, RTİ gibi bazı çekirdeklerden sonlu boyutlu bir $\varphi(x)$ dönüşümü bulmak mümkün değildir.

Çekirdekler üzerindeki tek şart Mercer şartlarıyla verilir [88, 102]. Bu şart ile $K(x, y)$ 'nin simetrik olması ve ancak ve ancak $\psi \neq 0$ için aşağıdaki şartların sağlanması gerekir.

$$\int \psi^2(x) dx < \infty \quad (\text{E2.28})$$

$$\iint K(x, y) \psi(x) \psi(y) dx dy > 0 \quad (\text{E2.29})$$

Basit bir örnek olarak, simetrik çekirdek matrisi için $\varphi(x)$ dönüşüm işlevi aşağıdaki gibi oluşturulabilir,

$$\varphi: x^i \mapsto \left(\sqrt{q_t} v_t^i \right)_{t=1}^n \in \mathfrak{R}^n, \quad i = 1, \dots, n.$$

Burada, q_t ve $v_t = (v_{ti})_{i=1}^n$ sırasıyla K 'nin öz değerlerini ve öz vektörünü göstermektedir.

$$\varphi(x^i)^T \varphi(x^j) = \sum_{t=1}^n q_t v_{ti} v_{tj} = (V^T \wedge V)_{ij} = K_{ij} = K(x^i, x^j)$$

Mercer şartı gereğince bir çekirdek matrisinin öz değerlerinin negatif olmaması gerekir. Eğer negatif bir öz değer var ise,

$$\zeta = \sum v_{si} \varphi(x_i) = \sqrt{\wedge} V' v_s$$

öz nitelik uzayındaki bir nokta

$$\|\zeta\|^2 = \zeta^T \zeta = v_s' V \sqrt{\wedge} \sqrt{\wedge} V' v_s = v_s' V \wedge V' v_s = v_s K v_s = q_s < 0$$

gibi negatif bir norma sahip olacaktır ki, buda geometrik olarak anlamsızdır.

Bununla birlikte, son zamanlarda çekirdeklerin pozitif tanımlı olup olmadığını sorgulamadan yeni formülasyonlar ve çözüm önerileri de geliştirilmektedir [42, 61].

(E2.17)'deki Lagrangian formülasyonundaki iç çarpımlar, istenilen doğrusal olmayan çekirdek ile yer değiştirerek

$$x^T y \leftrightarrow K(x, y), \quad (\text{E2.30})$$

yüksek boyutlu öz nitelik uzayında ayrılabilirlik durumu için, yeni ikincil Lagrangian

$$L_{ikincil}(\lambda) = -\frac{1}{2} \sum_{i,j=1}^L \lambda_i \lambda_j y^i y^j K(x^i, x^j) + \sum_{i=1}^L \lambda_i \quad (\text{E2.31})$$

$$\text{kısıtları: } \sum_{i=1}^L y^i \lambda_i = 0, \quad \lambda_i \geq 0, \quad i = 1, \dots, L \quad (\text{E2.32})$$

olarak oluşturulur. Bu formülasyon esnek pay durumuna

$$L_{ikincil}(\lambda) = -\frac{1}{2} \sum_{i,j=1}^L \lambda_i \lambda_j y^i y^j K(x^i, x^j) + \sum_{i=1}^L \lambda_i \quad (E2.33)$$

$$\text{kısıtları: } \sum_{i=1}^L y^i \lambda_i = 0, \quad 0 \leq \lambda_i \leq \frac{C}{L}, i=1, \dots, L \quad (E2.34)$$

ile genellenir.

Örnek çekirdek seçimleri

Polinomsal sistemler için (E2.35)'deki çekirdek seçilebilir:

$$K(x, x^i) = (x^T x^i + 1)^p, \quad (E2.35)$$

Bu durumda elde edilen karar işlevi

$$\ell(x) = \text{sign} \left(\sum_{\text{destek vektörler}} y^i \lambda_i (x^T x^i + 1)^p + b \right) \quad (E2.36)$$

formu ile verilir.

RTİA'lardaki, RTİ çekirdek

$$K(\|x - x^i\|) = \exp \left\{ -\gamma \|x - x^i\|^2 \right\} \quad (E2.38)$$

kullanılırsa, (E2.37)'deki karar işlevi elde edilir,

$$\ell(x) = \text{sign} \left(\sum_{\text{destek vektörler}} y^i \lambda_i K(\|x - x^i\|) + b \right). \quad (E2.37)$$

Burada, DVM ile elde edilen destek vektörlerin sayısı RTİA'larda gizli birimlerin sayısına karşılık gelir. Bundan dolayı RTİA'ların otomatik kurum algoritması olarak gösterilir [102].

Benzer bir yaklaşım, iki katmanlı YSA'yı kurmak için de takip edilebilir. Örneğin, Mercer şartını sağlayacak şekilde uygun v ve c_1 değerleriyle ifade edilen sigmoidal çekirdek seçilirse

$$K(x, x_i) = \tanh(v(x^T x^i) + c_1) \quad (E2.39)$$

ayrıştırmacı yüzey

$$\ell(x) = \text{sign} \left(\sum_{\text{destek vektörler}} y^i \lambda_i \tanh(v(x^T x^i) + c_1) + b \right) \quad (E2.40)$$

olur. Burada, iyi bir genelleme başarımı için destek vektörlerin sayısı, ilk katman (gizli katman) hücrelerinin sayısına eşit alınır. İlk katman hücrelerinin ağırlıkları destek vektörler (x^i), ikinci katman hücrelerinin ağırlıkları ise ($y^i \lambda_i$) alınır [102].

E2.4. Çok Sınıflı DVM'ler

DVM'ler asıl olarak ikili sınıflama problemleri için tasarlanmıştır. Çok sınıflı örüntü tanıma sistemleri, iki sınıflı DVM'leri birleştirilerek elde edilebilir. Bilimsel yazında üç temel yaklaşım vardır [22, 46, 65, 106]:

İlk yaklaşım, M sınıfların sayısı olmak üzere, Bire Karşı Diğer sınıflar (BKD) yöntemidir [46]. Bu sınıflayıcıda her bir makine, bir sınıfa karşı diğer tüm sınıflar alınarak eğitilir ve M tane iki sınıflı sınıflayıcı oluşturulur.

İkinci yaklaşım, çok sınıflı sınıflayıcıyı kurmak için Bire Karşı Bir (BKB) yöntemidir [46]. Burada her makine, bir sınıfa karşı diğer tek bir sınıf alınarak eğitilir ve $M(M-1)/2$ tane sınıflayıcı oluşturulur. Test verisini sınıflamak için, oluşturulan tüm makineler arasında parça-parça yarışma uygulanır, tenis turnuvasına benzer olarak en son kazanan, test veri sınıfını belirler.

Son yaklaşım, M tane düzlem için aynı anda eniyileme problemini oluşturan M -sınıflı DVM yöntemidir [106]. Bu yöntemde $(x^1, y^1), \dots, (x^L, y^L), y^i \in 1, \dots, M$ formunda L eğitim örneği için birincil M sınıflı DVM problemi

$$\frac{1}{2} \sum_{m=1}^M \|w_m\|^2 + \frac{C}{L} \sum_{i=1}^L \sum_{m \neq y^i}^M \xi_i^m \quad (\text{E2.41})$$

$$\text{kısıtları: } \left((w_{y^i})^T x^i \right) + b_{y^i} \geq \left((w_m)^T x^i \right) + b_m + 2 - \xi_i^m \quad (\text{E2.42})$$

$$\xi_i^m \geq 0, \quad i = 1, \dots, L \quad m \in \{1, \dots, M\} \setminus y^i. \quad (\text{E2.43})$$

olarak oluşturulur.

Bu eniyileme probleminin çözümü, ikili sınıflamaya benzer olarak ikincil formülasyon elde edilerek yapılır. Karar işlevi ise,

$$f(x) = \arg \max_M \left[\left((w_i)^T x \right) + b_i \right] \quad i = 1, \dots, M \quad (\text{E2.44})$$

kullanılarak elde edilir.

Pratik olarak, bu yöntemin üstünlüğü büyük bir eniyileme problemi çözdüğü için sınırlıdır. Bilinmeyen değişkenlerin sayısı sınıfların sayısı ile doğrusal olarak artar. Ayrıca yöntem, birçok test veri arşivi üzerinde ilk iki yöntemden üstün değildir. Bu nedenle yöntemin sadece başlıca akademik ve estetik değere sahip olduğu düşünülür.

E2.5. Regresyon Kestirimi İçin DVM'ler

DVM'ler, sınıflayıcılara benzer şekilde regresyon problemlerini çözmek için adapte edilmiştir. Kapasite, boyutluluğa duyarsız nicelik ile kontrol edilirken, doğrusal veya doğrusal olmayan işlev sırasıyla asıl veya öz nitelik uzayında doğrusal öğrenme makinesi ile öğrenilmiştir [41, 89].

DVM'lerde regresyon kestirimi için, giriş uzayında ($\varphi_i(x)=x$) veya öz nitelik uzayında, w parametresine göre doğrusal olan bir işlev kullanılır,

$$\ell(x, w) = \sum_{i=1}^r w_i \varphi_i(x) \quad (E2.45)$$

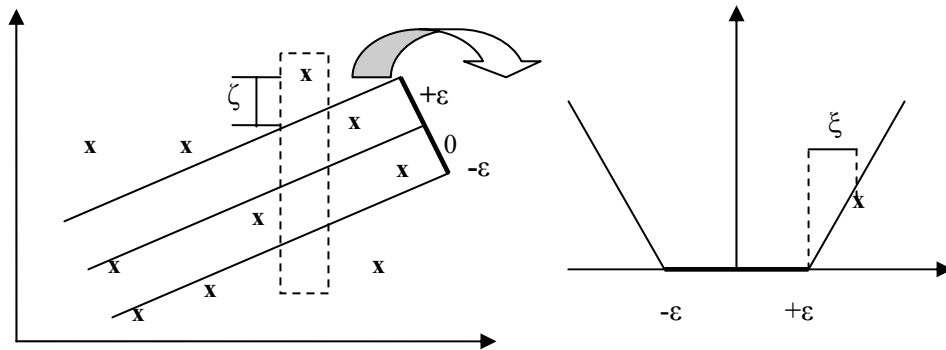
Burada; orijinden mesafeyi gösteren b parametresi düşürülmüştür, fakat sabit bir taban işlevi olarak gösterilebilir. Regresyon problemi için, farklı kayıp işlevleri kullanılabilir [22, 41]. Böylece, farklı formülasyonlar ve faydalar elde edilebilir [34-37, 58, 97] Bu bölümde, en genel DVM ile regresyon formülasyonu için, Şekil E2.4 ile verilen ε -duyarsız kayıp işlevi,

$$L_{\varepsilon}(y, \ell(x, w)) = \begin{cases} |y - \ell(x, w)| - \varepsilon & |y - \ell(x, w)| \geq \varepsilon \\ 0 & \text{diğerleri} \end{cases} \quad (E2.46)$$

tanımlanmıştır. Bu kayıp işlevi gürbüz regresyon için kullanılır. (E2.46) ile ε 'a kadar sapmalar cezalandırılmazken, diğer tüm sapmalar doğrusal bir cezaya uğratılır.

YRE ilkesini kullanarak, yaklaşım işlevler kümesinin yapısı enazlanırsa, her yapının elemanları S_k (E2.47)'deki gibi tanımlanır:

$$S_k = \left\{ \sum_{i=1}^r w_i \varphi_i(x) : (w^T w) \leq c_k \right\} \quad (E2.47)$$



Şekil E2.4 Doğrusal DVM için esnek payın ayarlanması [93].

Deneysel riski enazlayan w parametresi bulunmak istenirse, (E2.48)'deki işlev tanımlanır,

$$R_{dny}(w) = \frac{1}{L} \sum_{i=1}^L L_{1,\varepsilon} \left(y, \sum_{i=1}^r w_i \varphi_i(x) \right) \quad (\text{E2.48})$$

ve YRE ilkesini gerçekleştiren kısıtları

$$(w^T w) \leq c_k \quad (\text{E2.49})$$

ile verilir. Bu problem, gerçekte $\xi_i, \xi'_i, i = 1, \dots, L$ yapay değişkenleri tanımlanarak

$$\sum_{i=1}^L \xi_i + \sum_{i=1}^L \xi'_i \quad (\text{E2.50})$$

$$\text{kısıtları: } y^i - \sum_{i=1}^r w_i \varphi_i(x) \leq \varepsilon + \xi'_i \quad (\text{E2.51})$$

$$\sum_{i=1}^r w_i \varphi_i(x) - y^i \leq \varepsilon + \xi_i \quad (\text{E2.52})$$

$$\xi_i \geq 0 \quad (\text{E2.53})$$

$$\xi'_i \geq 0 \quad (\text{E2.54})$$

gibi deneysel riski enazlayan eniyileme problemine dönüştürülebilir. Eniyileme probleminin hem yapısal hemde deneysel riskin enazlanması için, sonuç eniyileme formülasyonun amaç ölçütü, ikinci dereceden programlama problemi olarak

$$\frac{C}{L} \left(\sum_{i=1}^L \xi_i + \sum_{i=1}^L \xi'_i \right) + \frac{1}{2} (w^T w) \quad (\text{E2.55})$$

gibi formüle edilebilir. Burada, C kullanıcı tarafından belirlenen yeterince büyük bir sabiti göstermektedir.

İkincil problemi elde etmek için aşağıdaki gibi Lagrangian kurularak Kuhn-Tucker teoremi uygulanır. Özet olarak, ençoklanacak Lagrangian:

$$L_{ikincil} = -\varepsilon \sum_{i=1}^L (\lambda_i + \lambda'_i) + \sum_{i=1}^L y^i (\lambda'_i - \lambda_i) - \frac{1}{2} \sum_{i,j=1}^L (\lambda'_i - \lambda_i)(\lambda'_j - \lambda_j) K(x^i, x^j) \quad (\text{E2.56})$$

biçiminde olup, kısıtları

$$\sum_{i=1}^L \lambda_i = \sum_{i=1}^L \lambda'_i \quad (\text{E2.57})$$

$$0 \leq \lambda_i \leq \frac{C}{L} \quad (\text{E2.58})$$

$$0 \leq \lambda'_i \leq \frac{C}{L}, \quad i = 1, \dots, L \quad (\text{E2.59})$$

olarak elde edilir.

E2.6. Uygulama Noktaları

Tüm uygulamalarda, ikinci dereceden problemlerin çözümü tek bir en büyük veya en küçük noktaya ulaşmayı garanti eder. Çözüm için ilk yol eğim iniş uygulamaktır [22]. Çözüme rasgele λ^0 başlangıç değerinden başlanır ve en dik iniş yolu izlenerek bu vektör güncellenir. Güncellemenin uzunluğu η öğrenme oranı olarak alınır. Öğrenme oranının dikkatli olarak seçilmesi gerekir. Eğer çok büyük seçilirse yakınsamayacak, fakat en büyük nokta civarında salınım yapacaktır. Eğer çok küçük seçilirse, algoritmanın yakınsaması uzun sürecektir. Christianini ve Shawe-Taylor, yakınsama için yeterli şartı sağladığını kanıtlayarak

$$\eta_i = \frac{1}{K(x^i, x^i)} \quad (\text{E2.60})$$

gibi her eğitim örneği için farklı bir öğrenme oranı kullanmıştır [22].

Bu problemi çözmek için, standart Newton yöntemi, eşlenik-eğim yöntemi veya birincil-ikincil iç nokta yöntemi gibi eniyileme yöntemleri de uygulanabilir [88, 108]. Bununla birlikte eğitim örneklerinin sayısı arttığı zaman, birçok pratik problem ortaya çıkar. En çok bilinenleri; eniyileme probleminin karmaşıklığı, eğitim örneklerinin sayısı ile karesel olarak değişen çekirdek matrisinin ölçüsü ile büyümesidir. Çoğu kez, çok fazla sayıda eğitim örneği için büyük bir çekirdek matrisin hesaplanması gerekir.

Bu problemin çözümü için, bilimsel yazında parçalama ve ayrıştırma gibi farklı çalışmalar yapılmıştır [88].

- Parçalama (gruplama, kümeleme): Çözüme eğitim örneklerinin keyfi bir alt kümesi ile başlanır ve eniyileyici başlangıç destek vektörleri bulmak için bu küme üzerinde çalışır. Bu başlangıç çözüm, ardıl olarak bu çözümü bozan eğitim örneklerini bulmak için kullanılır. Daha sonraki örnekler ve o andaki destek vektörler, yeni bir alt kümeyi oluşturur. Bu alt küme tekrar tekrar eniyileyiciyi çalıştırmak için kullanılır. Bu prosedür bazı durdurma kriterleri elde edilinceye kadar devam eder.
- Ayrıştırma: Önceki yöntem, alt küme ölçüsünün destek vektörlerin sayısından daha geniş olmasını gerektirir. Bu yöntem, alt küme ölçüsünü sabitler ve bir anda problemin küçük farklı alt kümeleri üzerinde eniyileyiciyi çalıştırır. Ardışıl en küçük eniyileme (sequential minimal optimization), bu yöntem temeline dayanarak çıkartılan bir algoritmadır.

EK-3. DESTEK VEKTÖR MAKİNE ÇEŞİTLERİ

Bilimsel yazında DVM'lerin eğitim hızını azaltmak, hafıza kullanımını artırmak, başarımını azaltmak gibi amaçlarla farklı DVM formülasyonları geliştirilmiştir. Bu tezde karşılaştırma amacıyla Mangasarian ve diğerlerinin geliştirdikleri, Lagrangian Destek Vektör Makine (LDVM) [69], Sonlu Newton Lagrangian Destek Vektör Makine (NLDVM) [34], Düzgün Destek Vektör Makine (DDVM) [58], Sonlu Newton Destek Vektör Makine (NDVM) [66], Yakınsal Newton Destek Vektör Makine (YDVM) [35, 36] ile Joachims'in geliştirdiği DVMlight kullanılmıştır [51]. Bu nedenle, bu bölümde bu altı DVM formülasyonu kısaca tanıtılmıştır.

LDVM, EK-2'de verilen genel doğrusal DVM formülasyonunun tekrar formüle edilmesine dayanmaktadır. n boyutlu gerçel uzayda kısıtsız türevi alınabilir dışbükey işlev, doğrusal olarak yakınsayan özyineli bir algoritma ile enazlanmaktadır [69].

NLDVM, çok geniş yüksek boyutlu giriş uzayı içeren problemler için doğrusal denklem sistemi kullanarak çok hızlı bir şekilde çözüm yapan basit bir algoritmadır [34].

DDVM, tamamen keyfi bir çekirdek kullanarak sınıflama yapan DVM formülasyonuna Newton yöntemini uygulamak için (başka bir deyişle ikinci türevini alarak, Hessian matrisini oluşturmak için) bir düzgünleştirme tekniğinin uygulandığı algoritmadır [58].

NDVM, veri madenciliği ve makine öğrenmenin temel sınıflandırma problemlerini Newton yöntemi kullanarak sonlu özyinelemede, hızlı bir şekilde çözen bir algoritmadır [66].

YDVM, DVM formülasyonunda kısıtlar eşitlik durumuna çevrilerek, en yakın sınır oluşturan düzlemler oluşturan bir algoritmadır [35, 36].

DVMlight, DVM formülasyonunun geniş veri içeren problemlere uygulanmasını sağlamak için yeni düzeltmeler öneren bir algoritmadır.

E3.1. Mangasarian'ın Destek Vektör Makine Formülasyonları

Aşağıda, bu bölümde kullanılan semboller ve özdeşlik hakkında kısa bir özet yapılmıştır. Tüm vektörler, matrisin devriği işlemi kullanılarak satır vektörüne dönüştürülmediği sürece sütun vektörü ile gösterilmiştir. n -boyutlu gerçel uzayda, $\mathbb{R}^n$ 'de, bir vektör olmak üzere x_+ negatif elemanları sıfıra eşitlenen bir vektörü, x_* ise x_+ 'nın gradyenini göstermektedir. Burada $x_i = 0$, $i = 1, \dots, n$ için $(x_*)_i \in [0, 1]$, $x_i < 0$ için $(x_*)_i = 0$ ve $x_i > 0$ için $(x_*)_i = 1$ olmak üzere

adım işlevini tanımlamaktadır. $\mathcal{G}$, doğal logaritmalar tabanını ve $B \in \Re^{L \times n}$ $L \times n$ gerçel matrisi göstermektedir. Böyle bir matris için, B^T , B matrisin devriğidir ve B_i , B 'nin i . satırını ve B_j , B 'nin j . sütununu göstermektedir. 0 veya e , gerçel uzayda keyfi boyutlu sıfır veya birler vektörünü ve I ise keyfi boyutlu birim matrisi göstermektedir.

$\Re^n$ 'de x ve ξ iki vektör olmak üzere $x \perp \xi$ dikliği göstermektedir, yani $x^T \xi = 0$ 'dır. Burada $\| \cdot \|$ tanımı göstermek için kullanılmıştır. x vektörünün ve Q matrisinin normu sırasıyla $\|x\|$ ve $\|Q\|$ ile gösterilmiştir. Eğer F , $\Re^n$ 'de tanımlanan gerçel bir işlev ise, $\nabla F(x)$, x 'deki gradyeni göstermektedir. Ayrıca formüllerde, (E3.1)'deki Sherman-Morrison-Woodbury (SMW) özdeşliğinden faydalanılmıştır,

$$\left(\frac{I}{\nu} + HH^T \right)^{-1} = \nu \left(I - H \left(\frac{I}{\nu} + H^T H \right)^{-1} H^T \right) \quad (\text{E3.1})$$

Burada; ν pozitif bir sayıyı ve H keyfi $L \times k$ matrisini göstermektedir. Bu özdeşlik, büyük bir $L \times L$ matrisinin tersini almak yerine, sadece küçük $k \times k$ matrisinin tersini alarak işlem kolaylığı sağlamaktadır.

E3.1.1. Doğrusal Destek Vektör Makine Sınıflayıcılar

LDVM formülasyonu, doğrusal DVM ile ilişkili olduğu için, burada yazarların sembolleri ve yeni düzenlemeleriyle tekrar tanımlanmıştır. Genel doğrusal çekirdekli DVM ile sınıflama problemi

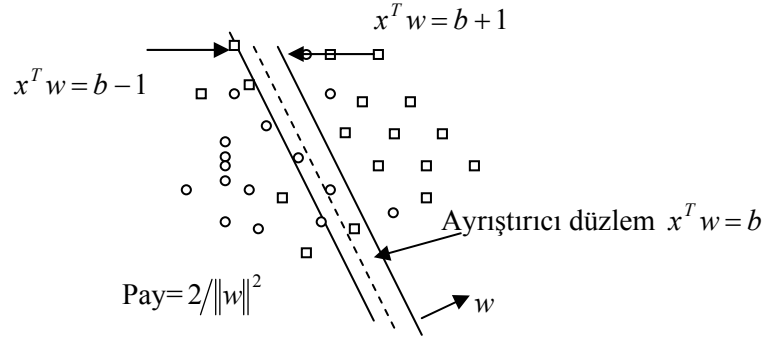
$$\min_{(w,b,\xi) \in \Re^{n+1+L}} \nu e^T \xi + \frac{1}{2} w^T w \quad (\text{E3.2})$$

$$\text{kısıt} \quad D(Bw - eb) + \xi \geq e, \quad \xi \geq 0$$

karesel programlama ile verilmiştir. Burada; D : B_+ veya B_- ile temsil edilen B_i noktasının sınıfına göre diyagonal elemanları 1 veya -1 olan $L \times L$ boyutlu diyagonal matrisi, $\nu > 0$ sabiti: ξ hata değişkeninin çarpanı olan ağırlık faktörünü, w , sınır oluşturan düzlemlere normal olan ağırlık vektörünü ve b : noktaların orijine göre mesafesini göstermektedir [102]. (E3.2)'deki ikinci terim, sınır oluşturan

$$x^T w = b \pm 1 \quad (\text{E3.3})$$

iki düzlem arasındaki mesafenin, 2-normunun karesinin iki katı olarak bilinen $2/\|w\|^2$ payı, birinci terim ise sınıflama hatasını temsil etmektedir.



Şekil E3.1 Esnek pay ile B_+ 'dan B_- 'yi yaklaşık olarak ayıran düzlemler.

Şekil E3.1'den görüldüğü gibi, $x^T w = b + 1$ düzlemi bazı hata olasılıkları ile B_+ sınıfının noktalarını ve $x^T w = b - 1$ düzlemi ise B_- sınıfının noktalarını ayırmaktadır. (E3.4)'teki doğrusal ayırıştırıcı düzlem

$$x^T w = b \quad (E3.4)$$

sınır oluşturan düzlemlerin ortasından geçmektedir. Eğer sınıflar doğrusal olarak ayrılmazlar ise, negatif olmayan hata değişkeni ξ ile belirlenen, bazı hatalar içererek yaklaşık olarak sınırlanır:

$$B_i w + \xi_i \geq b + 1, \quad D_{ii} = 1 \quad \text{için} \quad (E3.5)$$

$$B_i w - \xi_i \leq b + 1, \quad D_{ii} = -1 \quad \text{için}$$

Sonuç olarak, genel karesel doğrusal DVM'nin (E3.2) ile verilen birincil probleminin, ikincil formu yeni semboller ile

$$\min_{\lambda \in \mathbb{R}^L} \frac{1}{2} \lambda^T D B B^T D \lambda - e \lambda \quad (E3.6)$$

$$\text{kısıt } e^T D \lambda = 0, \quad 0 \leq \lambda \leq v e.$$

gibi elde edilir.

EK-2'den bilindiği gibi, birincil problemin (w, b) değişkenleri üsteki ikincil problemin çözümünden elde edilebilir. Ancak; bu formülasyonla, bazı problemler ortaya çıkmıştır. Bunlar; (E3.6)'daki amaç ölçütünde görünen $D B B^T D$ matrisi tipik olarak $m \gg n$ olduğu için, genelde pozitif tanımlı olmaması, sınır kısıtlarına ilave olarak farklı hesaplama kütüphanelerinin kullanımını gerektiren eşitlik kısıtının gelmesi, bir boyutlu eniyileme probleminin (E3.4)'teki ayırıştırıcı yüzeyin b yerini belirlemek için çözülmesi gerekliliğidir.

Bu güçlükleri aşmak için, [69]'da genel DVM formülasyonuna basit fakat önemli düzeltmeler önerilmiştir. ξ 'nin 1 normu, $\xi \geq 0$ kısıtını gereksiz yapan karesi alınmış 2 normu ile değiştirilmiş, $w^T w$ 'ye b^2 ilave edilmiş ve genel DVM'nin tersine düzlemlerin hem konumuna (w) , hem de yönüne (b) , göre eniyilenerek pay ençoklanmıştır. Yeni DVM'nin birincil formülasyonu

$$\min_{(w,b,\xi) \in \mathbb{R}^{n+1+L}} v \frac{\xi^T \xi}{2} + \frac{1}{2} (w^T w + b^2) \quad (\text{E3.7})$$

$$\text{kısıt} \quad D(Bw - eb) + \xi \geq e$$

ve ikincil formülasyonu

$$\min_{0 \leq \lambda \leq \mathbb{R}^L} \frac{1}{2} \lambda^T \left(\frac{I}{v} + D(BB^T + ee^T)D \right) \lambda - e^T \lambda \quad (\text{E3.8})$$

olarak elde edilmiştir. Birincil problemin (w,b) değişkenleri ile ikincil problemin değişkenleri (λ) arasındaki ilişki, ikincil problem çözülerek

$$w = B^T D \lambda, \quad \xi = \frac{\lambda}{v}, \quad b = -e^T D \lambda \quad (\text{E3.9})$$

elde edilmiştir.

İkincil amaç ölçütünde, görünen matris pozitif tanımlıdır, hiçbir eşitlik kısıtı yoktur ve λ ikincil değişkeni üzerinde hiçbir üst sınır yoktur. Şimdiki tek kısıt λ 'nın negatif olmamasıdır. Bu yararlar ile, basit, özyineli ve hiçbir eniyileme paketi gerektirmeyen Lagrangian DVM formülasyonu ortaya çıkmıştır.

E3.1.2. LDVM, NLDVM, DDVM ve NDVM Sınıflayıcılar

E3.1.2.1. Lagrangian Destek Vektör Makine (LDVM) Sınıflayıcılar

[69]'da formülleri basitleştirmek için

$$H = D[B \ -e], \quad Q = \frac{I}{v} + HH^T \quad (\text{E3.10})$$

matrisleri tanımlanarak, ikincil problem

$$\min_{0 \leq \lambda \in \mathbb{R}^L} F(\lambda) := \frac{1}{2} \lambda^T Q \lambda - e^T \lambda \quad (\text{E3.11})$$

olarak elde edilmiştir. Q^{-1} sadece bir kez algoritmanın başlangıcında hesaplanmış, LDVM algoritması içinde (E3.1)'de tanımlanan SMW özdeşliği kullanılmıştır.

LDVM algoritması, Karush-Kuhn-Tucker'in gerekli ve yeterli eniyileme şartları tabanına dayanmaktadır,

$$0 \leq \lambda \perp Q\lambda - e \geq 0. \quad (\text{E3.12})$$

Bağıntı (E3.12)'deki eniyileme şartı herhangi bir α için (E3.13)'teki eşdeğer forma sahiptir.

$$Q\lambda - e = ((Q\lambda - e) - \alpha\lambda)_+ \quad (\text{E3.13})$$

LDVM formülasyonunun çözümü, (E3.14)'teki çok basit özyineleme ile elde edilmiştir.

$$\lambda^{i+1} = Q^{-1} \left(e + \left((Q\lambda^i - e) - \alpha\lambda^i \right)_+ \right), \quad i = 0, 1, \dots, \quad (\text{E3.14})$$

(E3.13)'deki eniyileme şartı, (E3.11) ikincil problemi ile birleşen örtük Lagrangian için de gerekli ve yeterli şarttır.

$$\min_{\lambda \in \mathbb{R}^L} L(\lambda, \alpha) = \min_{\lambda \in \mathbb{R}^L} \frac{1}{2} \lambda^T Q \lambda - e^T \lambda + \frac{1}{2\alpha} \left(\left\| (-\alpha\lambda + Qu - e)_+ \right\|^2 - \|Q\lambda - e\|^2 \right) \quad (\text{E3.15})$$

Algoritma: LDVM sınıflayıcıların eğitim algoritması ve tümel yakınsaması

LDVM formülasyonunda, (E3.10) ile tanımlanan $Q \in \mathbb{R}^{L \times L}$ matrisinin simetrik pozitif tanımlı olduğu kabul edilmiştir. Rasgele $\lambda^0 \in \mathbb{R}^L$ ile başlanarak,

$$0 < \alpha < \frac{2}{v} \quad (\text{E3.16})$$

kolaylıkla sağlanabilen şartı altında, $\bar{\lambda}$ tek çözümüne

$$\|Q\lambda^{i+1} - Q\bar{\lambda}\| \leq \|I - \alpha Q^{-1}\| \|Q\lambda^{i+1} - Q\bar{\lambda}\| \quad (\text{E3.17})$$

doğrusal oranında yakınsanmıştır.

Orta sayıda veri içeren problemlerin, doğrusal olmayan çekirdek kullanılarak LDVM formülasyonu

$$\min_{0 \leq \lambda \in \mathbb{R}^L} \frac{1}{2} \lambda^T \left(\frac{I}{v} + DK(G, G')D \right) \lambda - e^T \lambda \quad (\text{E3.18})$$

$$G = [B \quad -e], \quad Q = \frac{I}{v} + DK(G, G')D \quad (\text{E3.19})$$

olarak verilmiştir. Burada veri sayısı az olduğu için, doğrusal durumun tersine SMW özdeşliğini kullanmaya gerek yoktur.

E3.1.2.2. Sonlu Newton Lagrangian Destek Vektör Makine (NLDVM) Sınıflayıcılar

NLDVM; (E3.15)'deki kısıtsız enazlanma problemine, Newton yönteminin uygulanmasından oluşmuştur. Newton yöntemini uygulamak için, öncelikle $L(\lambda)$ 'nın (E3.20)'deki gibi birinci türevinin

$$\nabla L(\lambda) = \frac{(\alpha I - Q)}{\alpha} ((Q\lambda - e) - ((Q - \alpha I)\lambda - e)_+) \quad (\text{E3.20})$$

daha sonra ikinci türevinin alınması gerekir. Bilindiği gibi, burada elde edilen $\nabla L(\lambda)$ gradyeninin türevi alınamaz ve dolayısıyla Hessian matrisi de oluşturulamaz. NLDVM'de ise $L(\lambda)$ 'nın genelleştirilmiş Hessian, (E3.21)'deki gibi $L \times L$ matris olarak elde edilmiştir [34].

$$\partial^2 L(\lambda) = \frac{(\alpha I - Q)}{\alpha} (Q + \text{diag}((Q - \alpha I)\lambda - e))(\alpha I - Q) \quad (\text{E3.21})$$

Temel Newton adımı, L doğrusal denklemlili sistemi çözmeden oluşmaktadır:

$$\nabla L(\lambda^i) + \partial^2 L(\lambda^i)(\lambda^{i+1} - \lambda^i) = 0 \quad (\text{E3.22})$$

Newton özyinelemesi, (E3.23)'deki gibi basitleştirilmiştir,

$$h(\lambda^i) + \partial h(\lambda^i)(\lambda^{i+1} - \lambda^i) = 0, \quad (\text{E3.23})$$

burada;

$$h(\lambda) := \left(\frac{\alpha I - Q}{\alpha} \right)^{-1} \nabla L(\lambda), \quad (\text{E3.24})$$

$$\partial h(\lambda) := \left(\frac{\alpha I - Q}{\alpha} \right)^{-1} \partial^2 L(\lambda). \quad (\text{E3.25})$$

olup, $\partial^2 L(\lambda)$ 'nun pozitif tanımlı olması gerekir. Ayrıca; NLDVM'de $L \gg n$ doğrusal problemler için, doğrusal sınıflayıcı oluşturmak istendiği her zaman, çok büyük olan $(L+1) \times (L+1)$ matrisinin tersini almak yerine, (E3.1)'deki SMW özdeşliği kullanılarak $(n+1) \times (n+1)$ boyutlu $\partial h(\lambda)$ 'nun tersi alınması gerekir.

Algoritma: NLDVM sınıflayıcılar için Newton algoritması

Rasgele $\lambda^0 \in \mathfrak{R}^L$ ile başlanarak, $\alpha > \|Q\|$ şartı altında tek çözüme doğrusal oranında yakınsanmıştır.

$$(i) \ h(\lambda^i - \partial h(\lambda^i)^{-1} h(\lambda^i)) = 0,$$

$$(ii) \ \lambda^{i+1} = \lambda^i - \beta_i \partial h(\lambda^i)^{-1} h(\lambda^i) = \lambda^i + \beta_i d^i.$$

Burada; β_i armijo adım ölçüsünü göstermektedir.

$$L(\lambda^i) - L(\lambda^i + \beta_i d^i) \geq -\delta \beta_i \nabla L(\lambda^i) d^i$$

Burada; $\delta \in (0, 1/2)$ ve d^i , (E3.23) çözülerek elde edilen Newton yönünü göstermektedir.

$$d^i = -\partial h(\lambda^i)^{-1} h(\lambda^i)$$

(iii) $i=i+1$ al ve i 'inci adıma git.

E3.1.2.3. Düzgün Destek Vektör Makine (DDVM) Sınıflayıcılar

DDVM formülasyonu için, (E3.7)'deki problemin çözümünde,

$$\xi = (e - D(Bw - eb))_+ \quad (\text{E3.26})$$

alınarak, kısıtsız eniyileme problemi

$$\min_{w,b} \frac{\nu}{2} \| (e - D(B w - e b))_+ \|_2^2 + \frac{1}{2} (w^T w + b^2) \quad (\text{E3.27})$$

elde edilmiştir. Bu problem, herhangi bir kısıt olmadan güçlü olarak dışbükey enazlanma problemi olduğu için tek bir çözüme sahiptir. Bununla birlikte; (E3.27)'deki amaç ölçütü; hızlı çözüm için, Newton yönteminin kullanımını engelleyen iki defa türevi alınamayan bir işlev içermektedir. DDVM'de bu problemten kurtulmak için, düzgünleştirme teknikleri uygulanmıştır. Amaç ölçütündeki x_+ terimi ile YSA'ların $1/(1 + \mathcal{G}^{-\alpha})$ sigmoid işlevin integrali olarak verilen $p(x, \alpha)$

$$p(x, \alpha) = x + \frac{1}{\alpha} \log(1 + \mathcal{G}^{-\alpha}), \alpha > 0 \quad (\text{E3.28})$$

yer değiştirilmiştir [58]. Burada; α düzgünleştirme parametresini göstermektedir.

Böylece, yeni DDVM formülasyonu

$$\min_{(w,b) \in \mathbb{R}^{n+1}} Q(w,b) := \frac{\nu}{2} \| p(e - D(B w - e b), \alpha) \|_2^2 + \frac{1}{2} (w^T w + b^2) \quad (\text{E3.29})$$

olarak elde edilmiştir. DDVM, sonsuza yaklaşan α ile (E3.27)'nin tamamen doğru bir çözümünü vermektedir.

Düzgünleştirme yaklaşımı ve genel DVM formülasyonu arasındaki tek fark, karesel programlama yerine doğrusal denklemlerden oluşan sistemin çözülmesi ve tümel olarak tek bir saf çözüme yakınsanmasıdır.

Doğrusal olmayan çekirdekli DDVM için formülasyon

$$\min_{\lambda, b} \frac{\nu}{2} \| p(e - D(K(B, B^T) D \lambda - e b), \alpha) \|_2^2 + \frac{1}{2} (\lambda^T \lambda + b^2) \quad (\text{E3.30})$$

olarak verilmiştir. DDVM'de, herhangi keyfi bir çekirdek kullanıldığında da oldukça doğrusal olmayan ayrıştırıcı yüzeyler oluşturarak türevi alınabilirlik ve güçlü dışbükeylik özellikleri korunur. Bir önceki sonuçların tümü sağlanır ve (E3.30)'u çözmek için doğrudan Newton adımı uygulanabilir.

E3.1.2.4. Sonlu Newton Destek Vektör Makine (NDVM) Sınıflayıcılar

DDVM formülasyonu ile düzgünleştirilerek çözülen (E3.27)'deki problem, NDVM'de doğrusal olmayan çekirdekli kısıtsız DVM problemi [66].

$$\min_{(\lambda, b) \in \mathbb{R}^{n+1}} \frac{\nu}{2} \| (e - D(K(B, B^T) D \lambda - e b))_+ \|_2^2 + \frac{1}{2} (\lambda^T \lambda + b^2) \quad (\text{E3.31})$$

olarak düşülüp, tek bir çözümü olan aşağıdaki parça parça karesel güçlü dışbükey problem

$$\min_{z \in \mathfrak{R}^p} F(z) := \frac{v}{2} \| (Cz - h)_+ \|^2 + \frac{1}{2} (z^T z) \quad (\text{E3.32})$$

olarak önerilmiştir. Burada; $C \in \mathfrak{R}^{L \times p}$, $h \in \mathfrak{R}^L$ ve v sabitlenmiş parametreyi göstermektedir.

(E3.32)'deki eniyileme problemini, Newton algoritması ile çözmek için gradyen vektörü ve Hessian matrisini elde edilmesi gerekir. NDVM'de F 'nin gradyeni:

$$\nabla F(z) = v C^T (Cz - h)_+ + z \quad (\text{E3.33})$$

olarak hesaplanmıştır. Burada verilen gradyen, K sabiti ile tümel olarak Lipschitz sürekliliğidir:

$$\|\nabla F(s) - \nabla F(z)\| \leq K \|s - z\|, \quad \forall s, z \in \mathbb{R}^p, \quad (\text{E3.34})$$

$$K = v \|C^T\| \|C\| + 1 \quad (\text{E3.35})$$

F 'nin Hessiani her yerde mevcut değildir. Bununla birlikte; $\nabla F(z)$ Lipschitzian olduğu için, genelleştirilmiş Hessian her yerde vardır.

[66]'te $F(z)$ 'nin genelleştirilmiş Hessiani

$$\partial^2 F(z) = v C \text{diag}(Cz - h)_* C + I \quad (\text{E3.36})$$

ile hesaplanmıştır.

Algoritma: NDVM sınıflayıcılar için adımsız Newton algoritması

Rasgele $z^0 \in \mathfrak{R}^p$ ile başlanarak

$$\frac{K}{k} = \frac{v \|C^T\| \|C\| + 1}{1} < 2, \quad v \|C^T\| \|C\| < 1 \quad (\text{E3.37})$$

şartı altında, tümel enaz $\bar{z}$ çözümüne

$$\|z^{i+1} - \bar{z}\| \leq v \|C^T\| \|C\| \|z^i - \bar{z}\| \quad (\text{E3.38})$$

doğrusal oranında yakınsanmıştır.

$$(i) \quad z^{i+1} = z^i - \partial^2 f(z^i)^{-1} \nabla f(z^i)$$

$$(ii) \quad \nabla f(z^{i+1}) = 0 \text{ ise dur.}$$

$$(iii) \quad i=i+1. \text{ al ve } i\text{'inci adıma git.}$$

NDVM'nin çözümünde; $F(z)$ eğer (E3.37)'deki şartı sağlamaz ise tümel enazda sonlanmayı sağlamak için Armijo adımı eklenerek çözüm yapılmıştır [66].

E3.1.2.5. Yakınsal Destek Vektör Makine (YDVM) Sınıflayıcılar

YDVM'de (E3.7)'deki DVM formülasyonu

$$\min_{(w, \lambda, b) \in \mathbb{R}^{n+1+L}} v \frac{\xi^T \xi}{2} + \frac{1}{2} (w^T w + b^2) \quad (\text{E3.39})$$

$$\text{kısıt} \quad D(Bw - eb) + \xi = e$$

olarak değiştirilmiştir. Bu değişim çok basit olmasına rağmen, problemi önemli olarak farklılaştırmıştır. Çünkü. $xw^T - b = \pm 1$; artık sınır oluşturan düzlemleri değil, her bir sınıfın öbeklendiği ve $(w^T w + b^2)$ terimiyle mümkün olduğu kadar uzağa itilen noktalar civarındaki, en yakın düzlemleri ifade etmektedir. Bu problemde $w = B^T D \lambda$ alınarak, YDVM formülasyonu

$$\min_{(\lambda, b, \xi) \in \mathbb{R}^{n+1+L}} v \frac{\|\xi\|^2}{2} + \frac{1}{2} (\lambda^T \lambda + b^2) \quad (\text{E3.40})$$

$$\text{kısıt} \quad D(AA^T D \lambda - eb) + \xi = e$$

olarak elde edilmiştir. Doğrusal olmayan çekirdek kullanıldığında BB^T doğrusal çekirdeği $K(B, B^T)$ ile yer değiştirilerek,

$$\min_{(\lambda, \xi, b) \in \mathbb{R}^{n+1+L}} v \frac{\|\xi\|^2}{2} + \frac{1}{2} (\lambda^T \lambda + b^2) \quad (\text{E3.41})$$

$$\text{kısıt} \quad D(K(B, B^T) D \lambda - eb) + \xi = e$$

Lagrangian

$$L(\lambda, b, \xi, v) = \frac{v}{2} \|\xi\|^2 + \frac{1}{2} \left\| \begin{bmatrix} \lambda \\ b \end{bmatrix} \right\|^2 - \alpha^T (D(KD \lambda - eb) + \xi - e) \quad (\text{E3.42})$$

olarak oluşturulmuştur.

Lagrangian'ın (λ, b, ξ, v) değişkenlerine göre türevi alınarak, KKT eniyileme şartları

$$\begin{aligned} \lambda - DK^T D \alpha &= 0, \\ b + e^T D \alpha &= 0, \\ v \xi - \alpha &= 0, \\ D(KD \lambda - eb) + \xi &= e, \end{aligned} \quad (\text{E3.43})$$

bulunmuş ve bu şartlar kullanılarak (E3.43)

$$\begin{aligned} \lambda &= DK^T D \alpha, \quad b = -e^T D \alpha, \quad \xi = \frac{\alpha}{v} \\ \alpha &= \left(\frac{I}{v} + D(KK^T + ee^T)D \right)^{-1} e = \left(\frac{I}{v} + GG^T \right)^{-1} e \\ G &= D[K \quad -e]. \end{aligned} \quad (\text{E3.44})$$

elde edilmiştir.

Sonuç doğrusal olmayan ayrıştırıcı yüzey

$$\begin{aligned} K(x^T, B^T) D \lambda - b &= K(x^T, B^T) D D K(B, B^T)^T D \alpha + e^T D \alpha \\ &= \left(K(x^T, B^T) K(A, B^T)^T + e^T \right) D \alpha = 0 \end{aligned} \quad (E3.45)$$

ile hesaplanmıştır.

Mangasarian ve diğerleri, veri kümesinin küçük bir bölümünü kullanarak, bilinen kare çekirdek matris yerine ince dikdörtgensel bir çekirdek matris oluşturmuşlardır. Bu çekirdek; yukarıda tanıtilan DVM formülasyonlarında uygulanmıştır. Böylece, hesapsal zaman kadar hafıza kullanımının da tam çekirdek kullanan genel DVM'lerden daha az olması sağlanmıştır [67].

E3.2. Joachims'in Destek Vektör Makine Formülasyonu

EK-2'de sınıflama için verilen genel karesel DVM formülasyonu

$$\min \quad L_{ikincil}(\lambda) = \frac{1}{2} \sum_{i,j=1}^L \lambda_i \lambda_j y^i y^j K(x^i, x^j) - \sum_{i=1}^L \lambda_i \quad (E3.46)$$

$$\text{kısıt} \quad \sum_{i=1}^L y^i \lambda_i = 0, \quad (E3.47)$$

$$0 \leq \lambda_i \leq \frac{C}{L}, \quad i = 1, \dots, L. \quad (E3.48)$$

DVMLight'de [51] $Q = y^i y^j K(x^i, x^j)$ alınarak,

$$\min \quad L(\lambda) = -\lambda^T 1 + \frac{1}{2} \lambda^T Q \lambda \quad (E3.49)$$

$$\text{kısıt} \quad \lambda^T y = 0 \quad (E3.50)$$

$$0 \leq \lambda \leq C/L \quad (E3.51)$$

şeklinde düzenlenmiştir. Daha önceden de bahsedildiği gibi bu eniyileme probleminin ölçüsü eğitim örneklerinin sayısına bağlıdır. Q matrisinin ölçüsü L^2 olduğu için, örneğin 10000 veya daha fazla eğitim örneği için, Q matrisini hafızada tutmak imkansızdır. Bir seçenek, Q matrisini hafızada tutmak yerine, her ihtiyaç duyulduğunda tekrar hesaplamaktır. Fakat bu da eğer Q matrisi çok sık kullanılırsa aşırı hesaplama gerektirir.

DVMLight'de çok fazla veri içeren problemler ile ilgilenmek için (E3.49)'daki eniyileme problemi, bir ayrıştırma yöntemi kullanılarak çalışma kümesi olarak adlandırılan aktif ve aktif olmayan bölümlere ayrılmıştır. Bu ayrıştırma ile çok uzun bir eğitim süresine ihtiyaç

duyulmuş ancak eğitim örneklerinin sayısı ve destek vektörlerin sayısı ile doğrusal hafıza kullanımı sağlanmıştır.

(E3.46) ile verilen eniyileme probleminde, Q Hessianı pozitif yarı tanımlı ve tüm kısıtları doğrusal olduğu için dışbükey eniyileme problemidir. Bu problemin uygunluğu için gerekli ve yeterli şartlar olarak aşağıdaki Kuhn-Tucker şartları verilmiştir,

$$g(\lambda) + (\lambda^{eq} y - \lambda^{lo} + \lambda^{up}) = 0, \quad (E3.52)$$

$$\lambda_i^{lo} (-\lambda_i) = 0, \quad \forall i \in [1..n] \quad (E3.53)$$

$$\lambda_i^{up} (\lambda_i - C/L) = 0, \quad \forall i \in [1..n] \quad (E3.54)$$

$$\lambda^{lo} \geq 0, \quad (E3.55)$$

$$\lambda^{up} \geq 0, \quad (E3.56)$$

$$\lambda^T y = 0, \quad (E3.57)$$

$$0 \leq \lambda \leq C/L. \quad (E3.58)$$

Burada; λ^{eq} : (E3.50)'deki eşitlik kısıtı için Lagrange çarpanını, λ^{up} : (E3.51)'deki üst sınır için Lagrange çarpanını ve $g(\lambda)$: kısmi türevler vektörünü göstermektedir. (E3.49) ile verilen eniyileme problemi için

$$g[\lambda] = -1 + Q\lambda \quad (E3.59)$$

olarak tanımlanmıştır

Uygunluk şartları sağlanmadığı durumda, algoritma (E3.49) ile verilen eniyileme problemini ayrıştırarak ortaya çıkan daha küçük problemleri çözmüştür. Bu eniyileme probleminde değişkenler, serbest (BI) ve sabit değişkenler (N) olarak iki kategoriye bölünmüştür.

Algoritma: DVMLight algoritması

- Uygunluk şartları bozulurken,

BI çalışma kümesi için q değişkenleri seçilir. Kalan $L-q$ değişkeni bulunduğu değerde sabit tutulur.

Problem ayrıştırılır ve elde edilen yeni karesel problemi çözülür: BI değişkenler üzerinden $L(\lambda)$ eniyilenir.

- Uygun λ bulunur ve sonlandırılır.

Algoritmada, λ , y ve Q , BI ve N 'e göre aşağıdaki gibi düzenlenmiştir,

$$\lambda = \begin{bmatrix} \lambda_{BI} \\ \lambda_N \end{bmatrix} \quad y = \begin{bmatrix} y_{BI} \\ y_N \end{bmatrix} \quad Q = \begin{bmatrix} Q_{BI BI} & Q_{BI N} \\ Q_{N BI} & Q_{NN} \end{bmatrix}. \quad (E3.60)$$

Q simetrik olduğu için,

$$\min \quad L(\lambda) = -\lambda_{B1}^T (1 - Q_{B1N} \lambda_N) + \frac{1}{2} \lambda_{B1}^T Q_{B1B1} \lambda + \frac{1}{2} \lambda_N^T Q_{NN} \lambda_N - \lambda_N^T 1 \quad (\text{E3.61})$$

$$\text{kısıt} \quad \lambda_{B1}^T y_{B1} + \lambda_N^T y_N = 0 \quad (\text{E3.62})$$

$$0 \leq \lambda \leq C/L \quad (\text{E3.63})$$

olarak yazılmıştır. Burada; $\frac{1}{2} \lambda_N^T Q_{NN} \lambda_N$ ve $-\lambda_N^T 1$ sabit terimleri çözümü değiştirmedikleri için atılarak, oldukça küçük, pozitif tanımlı karesel programlama problemi elde edilmiştir.

Çalışma kümesi seçildikten sonra, $L(\lambda)$ 'nın enazını bulmak için, elemanları sıfır olmayan q en dik iniş yönü elde edilmiştir. Bu elemanlara karşılık gelen değişkenler bulunan adımda çalışma kümesi olarak adlandırılmıştır.

Böylece eniyileme problemi

$$\min \quad V(d) = g(\lambda^{(t)})^T d \quad (\text{E3.64})$$

$$\text{kısıt} \quad y^T d = 0 \quad (\text{E3.65})$$

$$d_i \geq 0 \quad i: \lambda_i = 0 \text{ için} \quad (\text{E3.66})$$

$$d_i \leq 0 \quad i: \lambda_i = C/L \text{ için} \quad (\text{E3.67})$$

$$-1 \leq d \leq 1 \quad (\text{E3.68})$$

$$|\{d_i : d_i \neq 0\}| = q \quad (\text{E3.69})$$

olarak değişmiştir. Burada ilk üç kısıt ile problemin çözülebilir olması, dördüncü kısıt ile problemin ölçeklenmesi, beşinci ve son kısıt ile problemin seyrekliği sağlanmıştır.

(E3.64)'deki eniyileme probleminin çözümü için çok basit bir yol takip edilmiştir. Önce $\omega_i = y^i g_i(\lambda^{(t)})$ tanımlanarak, ω_i azalan düzende sıralanmıştır. Daha sonra bu listenin üstünden, $0 < \lambda_i^{(t)} < C/L$ veya $d_i = -y^i$ 'nin (E3.66) ve (E3.67)'yi sağlaması için $q/2$ eleman ve altından $0 < \lambda_i^{(t)} < C/L$ veya $d_i = y^i$ 'nin (E3.66) ve (E3.67)'yi sağlaması için $q/2$ eleman seçilerek, q değişkenden oluşan çalışma kümesi oluşturulmuştur.

DVMLight ayrıca, DVM'nin eğitimini hızlandırmak için bilimsel yazına iki farklı teknik tanıtmıştır. Birincisi, eğer nokta destek vektör değilse veya o nokta uygunluk şartlarını sağlamıyorsa, hemen eğitim kümesinden uzaklaştırılmasıdır (shrinking). Diğeri ise Q Hessiani elde etmek için q satır kullanmak, çekirdek hesaplamalarında ağır bir yük getirdiği için, bir gizleme (cache) tekniği ile gerekli çekirdek hesaplamalarının sayısının azaltılmasıdır.

DVMLight, başlangıçta birincil-ikincil iç-nokta yöntemleri kullanılarak çözülmüştür, problem şimdi birçok farklı çözücü ile çözülmektedir.

ÖZGEÇMİŞ

Adı Soyadı: Ayşegül UÇAR

Doğum Tarihi: 25 Mayıs 1977

Öğrenim Durumu:

Derece	Bölüm/ Üniversite	Yıl
Lisans	Elektrik-Elektronik Mühendisliği Bölümü, Fırat Üniversitesi	1994-1998
Yüksek Lisans	Elektrik-Elektronik Mühendisliği Bölümü, Devreler ve Sistemler Anabilim Dalı, Fırat Üniversitesi	1998-2000
Doktora	Elektrik-Elektronik Mühendisliği Bölümü, Devreler ve Sistemler Anabilim Dalı, Fırat Üniversitesi	2000-2006

Yüksek Lisans Tez Başlığı ve Tez Danışmanı:

Fuzzy-Nöral ve Yapay Sinir Ağları ile Lineer Olmayan Sistemlerin Modellenmesi ve Kontrolü,

Doç. Dr. Yakup Demir

Görevler:

Görev Unvanı	Görev Yeri	Yıl
Arş.Gör.	Fırat Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü	Kasım 2000-...