ÖZELLİK SEÇİMİ PROBLEMLERİ İÇİN POLİHEDRAL KONİK FONKSİYONLAR TEMELLİ ÇÖZÜM YAKLAŞIMI

Yüksek Lisans Tezi

Öznur AY

Eskişehir 2019

ÖZELLİK SEÇİMİ PROBLEMLERİ İÇİN POLİHEDRAL KONİK FONKSİYONLAR TEMELLİ ÇÖZÜM YAKLAŞIMI

Öznur AY

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Refail KASIMBEYLİ

Eskişehir
Eskişehir Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Ocak 2019

JÜRİ VE ENSTİTÜ ONAYI

Öznur AY'ın "Özellik Seçimi Problemleri için Polihedral Konik Fonksiyonlar Temelli Çözüm Yaklaşımı" başlıklı tezi 17/01/2019 tarihinde, aşağıdaki jüri tarafından "Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği" nin ilgili maddeleri uyarınca Endüstri Mühendisliği Anabilim dalında Yüksek Lisans tezi olarak kabul edilmiştir.

		Unvanı-Adı-Soyadı	İmza
Üye (Tez Danışmanı)	:	Prof. Dr. Refail KASIMBEYLİ	……..
Üye	:	Doç. Dr. Gürkan ÖZTÜRK	……..
Üye	:	Doç. Dr. Burak ORDİN	……..

Prof. Dr. Ersin Yücel
Enstitü Müdürü

ÖZET

ÖZELLİK SEÇİMİ PROBLEMLERİ İÇİN POLİHEDRAL KONİK FONKSİYONLAR TEMELLİ ÇÖZÜM YAKLAŞIMI

Öznur AY
Endüstri Mühendisliği Anabilim Dalı
Eskişehir Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ocak, 2019
Danışman: Prof. Dr. Refail KASIMBEYLİ

Bu çalışmada makine öğrenmesinin önemli dallarından biri olan özellik seçimi problemleri için çok yüzlü konik fonksiyonlar temelli bir gömülü yöntem önerilmiştir. Çok yüzlü konik fonksiyonlar algoritması, özellikle dışbükey örtüleri ayrık olmayan kümeler için geliştirilmiş ve literatürde yayımlanan çalışmalarla da sınıflandırma alanında çok yüksek başarılar elde ettiği gösterilmiştir. Bu çalışmanın amacı özellik seçimi problemleri için de çok yüzlü konik fonksiyonların kullanılmasını içeren bir algoritma geliştirilmesidir. Hiperdüzlemlerle ayırma teorisine dayanan klasik algoritma, özellik seçimi durumunu da içerecek şekilde yeni bir amaç fonksiyonu ve durdurma kriteri tariflenerek tekrar düzenlenmiştir. Çok yüzlü konik fonksiyonlar kullanılarak oluşturulan yeni amaç fonksiyonu sayesinde eş zamanlı bir şekilde hem özellik seçimi hem de sınıflandırma yapılabilmektedir. Ortaya konulan yeni algoritma, literatürde iyi bilinen güncel veri kümeleri üzerinde test edilmiş ve elde edilen sonuçlar yorumlanarak bilinen diğer yöntemlerle karşılaştırılmıştır. Ayrıca öğrenci başarı tahminlemesi problemine dair bir veri kümesi kullanılarak açıklayıcı bir örnek sunulmuştur.

Anahtar Sözcükler: Makine Öğrenmesi, Özellik Seçimi, Çok Yüzlü Konik Sınıflandırıcılar, Optimizasyon.

ABSTRACT

A SOLUTİON APPROACH BASED ON POLYHEDRAL CONIC FUNCTIONS FOR FEATURE SELECTION PROBLEMS

Öznur AY

Department of Industrial Engineering

Eskişehir Technical University, Graduate School of Science, January 2019

Supervisor: Prof. Dr. Refail KASIMBEYLİ

In this study, an embedded method based on polyhedral conic functions has been proposed for feature selection problems, which is one of the important branches of machine learning. The polyhedral conic functions algorithm is a supervised classification algorithm developed for separating the sets which of convex hulls are intersect. It is reported on different studies in literature that PCF algorithm gives competitive classification accuracies. The purpose of this study is to develop an algorithm that includes polyhedral conic functions for feature selection problems. The classical algorithm based on the separation theory has been rearranged by defining a new objective function which includes a feature selection term and a new termination criteria. Thanks to the new objective function both feature selection and classification can be performed simultaneously. The proposed algorithm is applied to solve classification problems on some well known real world data sets and compared with other well known classifiers. Finally, an illustrative example is presented using a student performance data set.

Keywords: Machine Learning, Feature Selection, Polyhedral Conic Classifiers, Optimization.

TEŞEKKÜR

Tez danışmanlığımı üstlenerek çalışmamın her aşamasında katkılarını esirgemeyen değerli hocam Prof. Dr. Refail KASIMBEYLİ'ye ve bu günlere gelmemde en büyük katkıya sahip olan sevgili aileme teşekkürü borç bilirim.

Öznur AY

OCAK 2019

17/01/2019

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan "bilimsel intihal tespit programı"yla tarandığını ve hiçbir şekilde "intihal içermediğini" beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

.....

Öznur AY

İÇİNDEKİLER

Sayfa

TABLOLAR DİZİNİ

Sayfa

ŞEKİLLER DİZİNİ

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

e : Birler vektörü

ε : Doğal logaritmanın tabanı

Enk : En küçüklemek

ÇKF : Çok yüzlü konik fonksiyonlar

GDP : Gürbüz doğrusal programlama

KÖS : Konkav özellik seçimi

ÖS-ÇKF : Önerilen özellik seçimi yöntemi

ÖS-MÇKF : Önerilen özellik seçimi yönteminin modifiye versiyonu

1. GİRİŞ

Son yıllarda büyük boyutlu verilerin miktarı özellikle internet ve sosyal medyanın kullanımına bağlı olarak katlanarak artmaktadır. Geleneksel internet kullanıcılarının rolü eskiden sadece içerik tüketicisi iken, özellikle sosyal medyanın gelişimi ile bu durum kullanıcıların hem içerik tüketicisi hem de içerik üreticisi konumuna gelmelerine sebep olmuştur. Bu yeni kullanıcı profilinin tükettiği ve ürettiği içerik mükemmel veriden, istenmeyen veriye hatta kötüye kullanım içeriğine kadar çeşitlilik göstermektedir. Dolayısıyla günümüzde internet veya sosyal medya üzerinden elde edilen veriler yüksek düzeyde gürültü barındırmaktadır [1]. Kuşkusuz bu kadar büyük ve gürültülü veriden yararlı bilgi ve örüntüleri çıkarmak oldukça zor bir iştir. Bu nedenle, bu denli büyük boyutlu ve gürültü barındıran veri kümelerinden anlamlı sonuçlar çıkarabilmek için veri madenciliği teknikleri geliştirilmiştir. Veri madenciliği; istatistik, yapay zeka ve makine öğrenmesi olmak üzere üç ana disiplin altında toplanabilir. Veri madenciliğinde en sık ortaya çıkan problemler, çoğunlukla verilerin anlamlı bütünler halinde kategorize edilmesi istendiğinden, sınıflandırma ve kümelemedir. Sınıflandırma problemlerinde, hangi verinin hangi veri kümesine ait olduğunu gösteren etiket bilgileri bilindiğinden denetimli öğrenme, kümeleme problemlerinde ise etiket bilgileri bilinmediğinden denetimsiz öğrenme olarak adlandırılır.

Büyük ölçekli verilerle çalışırken makine öğrenmesi yöntemlerini etkili bir şekilde kullanabilmek için bir veri ön işleme yöntemi uygulamak gerekir. Boyut azaltma teknikleri en sık kullanılan veri ön işleme yöntemlerinden biridir ve son zamanlarda makine öğrenme sürecinin vazgeçilmez bir parçası haline gelmiştir. Boyut azaltma teknikleri genel olarak özellik çıkarımı ve özellik seçimi olarak kategorize edilebilir. Özellik çıkarımı yaklaşımları, özellikleri daha düşük boyutlu yeni bir özellik kümesine yansıtır ve yeni özellikler genellikle orijinal özelliklerin kombinasyonu şeklinde oluşturulur. Özellik çıkarımı yöntemleri Temel Bileşenler Analizi, Lineer Diskriminant Analizi ve Kanonik Korelasyon Analizi'nden oluşmaktadır. Özellik seçimi yaklaşımları ise gereksiz özellikleri minimize eden ve bağımlı değişkenle (sınıflandırmada kullanılan sınıf etiketleri gibi) ilgili özellikleri maksimize eden küçük bir özellik kümesi seçmeyi amaçlar [1].

Hem özellik çıkarımı hem de özellik seçimi yaklaşımları öğrenme performansını artırma, hesaplama karmaşıklığını düşürme, daha iyi genelleştirilebilir modeller oluşturma ve gerekli depolama alanını azaltma yeteneğine sahiptir. Özellik çıkarımı,

orijinal özellik kümesinin kombinasyonlarını kullanarak, orijinal özellik kümesini daha düşük boyutlu bir özellik kümesine eşler. Fakat özellik çıkarımı teknikleri ile elde edilen dönüştürülmüş veriler fiziksel bir anlam taşımadığından yeni özellikler ile veriler üzerinde ileri analizler yapmak oldukça zordur. Özellik seçimi ise orijinal özellik kümesinden herhangi bir dönüşüm yapmadan orijinal özellikler ile fiziksel bağlantıyı sürdürecek şekilde bir özellik alt kümesi seçer. Bu nedenle, özellik seçimi okunabilirlik ve yorumlanabilirlik açısından özellik çıkarımından daha üstündür. Bu durum, belirli bir hastalığa ilişkin ilgili genleri bulma ve duygu analizi için bir duygu sözlüğü oluşturma gibi birçok pratik uygulamada önem kazanmaktadır [1].

Sınıflandırma problemlerinde özellik seçimi ile bağımlı değişkeni açıklama kabiliyeti yüksek özelliklerin bir alt kümesinin seçilmesi amaçlanır. Başka bir deyişle, farklı sınıflara ait olan örnekleri ayırma yeteneğine sahip özelliklerin seçilmesi amaçlanır. Bu nedenle, sınıflandırma problemlerinde etiket bilgilerinin varlığından dolayı özelliklerin ilişkisi farklı sınıfları ayırt edebilme yeteneği olarak değerlendirilir.

Tez kapsamında, Bradley ve Mangasarian tarafından geliştirilmiş olan Konkav Özellik Seçimi (KÖS) [2] yaklaşımı yardımıyla, Gasimov ve Öztürk tarafından ortaya atılan Çok Yüzlü konik Fonksiyonlar (ÇKF) [3] algoritmasına özellik seçimi uygulanmıştır. Yapılan çalışmada, çok yüzlü konik fonksiyonlar algoritmasındaki alt problemin amaç fonksiyonuna konkav özellik seçimi yaklaşımında özellik seçimi için kullanılan terim bir ağırlıklandırma parametresi ile eklenerek gömülü bir özellik seçimi yöntemi elde edilmek istenmiştir. Yöntem, genel itibariyle çok yüzlü konik fonksiyonlar algoritmasının mantığını koruyarak eklenen terim ve yapılan diğer değişiklikler ile eş zamanlı olarak hem özellik seçimi hem de sınıflandırma yapabilecek şekilde tariflenmiştir. Ayrıca bu yönteme ek olarak Gasimov ve Öztürk tarafından geliştirilmiş modifiye ÇKF algoritmasından yararlanılarak bir versiyon daha ortaya atılmıştır.

Tez çalışmasının izleyen bölümleri şu şekilde organize edilmiştir: İkinci bölümde sınıflandırma problemi ve literatürde sıkça kullanılan çözüm yöntemleri ile ilgili genel bilgiler verilmiştir. Üçüncü bölümde, özellik seçiminin ve sınıflandırma problemlerinde özellik seçiminin ne ifade ettiği tariflenmiş ve literatürde kullanılan yöntemlerden bahsedilmiştir. Dördüncü bölümde sınıflandırma problemleri için geliştirilmiş matematiksel programlama temelli çözüm yöntemlerinin bir kaçından bahsedilmiştir. Beşinci bölümde konkav özellik seçimi yaklaşımı ve tez kapsamında önerilen özellik seçimi yöntemi anlatılmıştır. Altıncı bölümde önerilen yönteme ait hesapsal sonuçlar ve

diğer yöntemlerle olan karşılaştırmalar ile önerilen yöntemin iki versiyonu arasında detaylı bir karşılaştırma sunulmuştur. Yedinci bölümde ise önerilen özellik seçimi yöntemi ile öğrenci başarı tahminlemesine yönelik açıklayıcı bir uygulama sunulmuştur. Son olarak, sekizinci bölümde ise yapılan çalışma ile ilgili değerlendirme ve öneriler ortaya koyulmuştur.

2. SINIFLANDIRMA

Sınıflandırma, hangi sınıfa ait olduğu bilinen gözlemlerden oluşan bir eğitim kümesini temel alarak elde edilen yeni bir gözlemin hangi sınıfa ait olduğunu belirleme problemidir. Bu tanıma göre birçok gerçek hayat problemi sınıflandırma problemi olarak modellenebilir. Örneğin, belirli bir e-postayı "istenmeyen" veya "istenmeyen değil" olarak ayırmak veya belli bir hastaya gözlenen özelliklere göre (cinsiyet, kan basıncı, belirli semptomların varlığı veya yokluğu vb.) tarif edildiği gibi teşhis koymak. Eğitim kümesindeki verilerin etiket bilgileri önceden bilindiğinden sınıflandırma problemleri denetimli öğrenme olarak adlandırılır. Eğitim kümesindeki verilerin etiket bilgilerinin bir kısmının bilindiği durum yarı denetimli, hiçbir verinin etiketinin bilinmediği durum ise denetimsiz öğrenme olarak adlandırılır.

Sınıflandırma probleminin çözümü eğitim aşaması ve tahmin aşaması olmak üzere iki aşamadan oluşur. Eğitim aşamasında, literatürde eğitim kümesi olarak adlandırılan belli sayıdaki etiketli veriden oluşan küme üzerine öğrenme algoritması uygulanarak, algoritmanın eğitilmesi amaçlanır. Burada, verilerin özellikleri kategorik olabilir (örneğin, kan grubu için "A", "B", "AB" veya "O"), sırasal olabilir (örneğin, "büyük", "orta" veya "küçük"), tam sayı değerli olabilir (örneğin, bir e-postanın kaç kelimeden oluştuğu) veya gerçek değerli olabilir (örneğin, kan basıncı ölçümü). Sınıflandırma algoritmalarının verilerin sahip olduğu özelliklere göre etiket bilgilerini kullanarak ayırıcı bir fonksiyon öğrenmesi amaçlanır [1].

Tahmin aşamasında, literatürde test kümesi olarak adlandırılan algoritmanın daha önce karşılaşmadığı etiket bilgisi bulunmayan verilerden oluşan küme ile öğrenme algoritmasının genelleştirme başarısı sınanır. Bu nedenle sınıflandırma algoritmalarının genelleştirme başarısının yüksek olması beklenir, aksi halde algoritma *aşırı uyum* ile sonuçlanır. Aşırı uyum kavramı ilerleyen bölümde detaylı bir şekilde ifade edilecektir.

Literatürde pek çok sınıflandırma yöntemi bulunmaktadır. Bu yöntemler genel olarak ayırma analizi, Bayes sınıflandırma algoritması, en yakın komşu, karar ağaçları, yapay sinir ağları ve destek vektör makineleri olarak kategorize edilebilir.

Ayırma problemi ilk kez Fisher tarafından 1930'larda ortaya atılmıştır. Ayırma analizi bir X veri kümesindeki örneklerin iki veya daha fazla gerçek gruba ayrılmasını ve örneklerin p tane özelliğini ele alarak doğal ortamdaki gerçek gruplarına en iyi şekilde atanmasını sağlayacak fonksiyonlar üreten bir yöntemdir [4].

Bayes sınıflandırma algoritması, adını Matematikçi Thomas Bayes'den alan bir sınıflandırma algoritmasıdır. Bayes sınıflandırması olasılık ilkelerine göre tanımlanmış bir dizi hesaplama ile sisteme sunulan verilerin sınıfını tespit etmeyi amaçlar.

En yakın komşu algoritmasına göre sınıflandırma sırasında çıkarılan özelliklerden, sınıflandırılmak istenen yeni verinin daha önceki verilerden k tanesine yakınlığına bakılmasıdır. Örneğin k = 3 için yeni bir veri sınıflandırılmak istensin. Bu durumda eski sınıflandırılmış verilerden en yakın 3 tanesi alınır. Bu veriler hangi sınıfa ait ise yeni veri de o sınıfa dahil edilir. Uzaklık hesaplarken genellikle öklid uzaklığı kullanılır [5].

Karar ağaçlarında, bir ağaç yapısı oluşturularak ağacın yaprak düğümleri seviyesinde sınıf etiketleri bulunur. Bu sınıf etiketlerine ulaşmak için kök düğümden başlanarak her seviyede sorular sorulur ve cevapların yönlendirmesiyle ara düğümlerden yapraklara ulaşılır [5].

Yapay sinir ağları, insan beyninden esinlenerek geliştirilmiş, ağırlıklı bağlantılar aracılığıyla birbirine bağlanan ve her biri kendi belleğine sahip işlem elemanlarından oluşan paralel ve dağıtılmış bilgi işleme yapıları; bir başka deyişle, biyolojik sinir ağlarını taklit eden bilgisayar programlarıdır [6].

Sınıflandırma için bir düzlemde bulunan iki veri kümesi arasında bir ayırıcı yüzey çizilerek iki kümeyi ayırmak mümkündür. Bu ayırıcı yüzeyin çizileceği yer ise iki kümenin de verilerine en uzak olan yer olmalıdır. **Destek vektör makineleri** bu ayırıcı yüzeyin nasıl çizileceğini belirler.

2.1 Model Seçme Yöntemleri

Geliştirilen herhangi bir makine öğrenmesi yönteminin performansını doğru bir şekilde analiz edebilmek için kurulan tahminleme modelinin yapısı oldukça önemlidir. Bunun için çeşitli model seçme yöntemleri geliştirilmiştir. Literatürde mevcut olan ve tez çalışması kapsamında kullanılan yöntemlerden aşağıda kısaca bahsedilmiştir.

Bir eğitim bir test kümesi (Holdout): Bu yöntemde veri, eğitim kümesindeki eleman sayısı daha çok olacak şekilde eğitim ve test kümesi olmak üzere iki bölüme ayrılmaktadır. Eğitim kümesi, kullanılan sınıflandırma yöntemini eğitmek amacıyla, test kümesi ise sınıflandırıcının performansını değerlendirmek amacıyla kullanılır. Bu yöntemin dezavantajları, eğer az miktarda veri mevcut ise test kümesi için ayrılan veri

sayısının yeterli olmaması ve tüm verinin eğitim kümesi, test kümesi olarak yalnızca bir kez ayrılmasından kaynaklanan hataların ortaya çıkabilmesidir [7].

k-kat çapraz doğrulama (k-fold cross validation): Bu yöntemde, öncelikle veri kümesi eleman sayıları olabildiğince birbirine yakın ya da mümkünse eşit olacak şekilde k adet alt kümeye bölünür, daha sonra her bir alt küme test kümesi ve kalan $k-1$ adet alt kümenin tamamı eğitim kümesi olacak şekilde sınıflandırma yöntemi uygulanır. Elde edilen değerlendirme ölçütleri tutularak model, bir sonraki alt küme test kümesi ve kalan alt kümeler eğitim kümesi olacak şekilde yeniden oluşturulur. Bu şekilde devam edilerek tüm alt kümeler ayrı ayrı test kümesi olarak kullanılır. Kurulan modelin genel başarısı, tüm modellerden elde edilen sonuçların ortalaması alınarak bulunur. k sayısı ise literatürde en çok 10 olarak tercih edilmektedir [7].

Katmanlı k-kat çapraz doğrulama (Stratified k-fold cross validation): Bu yöntem k-kat çapraz doğrulama yönteminin alt kümelere bölme işleminde farklı bir yaklaşım kullanılmasıyla oluşturulmuştur. Katmanlı k-kat çapraz doğrulama yönteminde, her bir alt küme, orijinal eğitim kümesindeki sınıf dağılımlarının oranı olabildiğince korunacak şekilde oluşturulur. Alt kümelerin bu şekilde seçilmesiyle gerçek veriyi daha iyi yansıtacak modeller elde edilir [8].

Biri dışarıda kalsın (Leave one out): Bu yöntemde, n elemanlı bir küme için her seferinde yalnızca bir örnek test kümesi ve kalan $n-1$ örnek ise eğitim kümesi olarak kullanılacak şekilde n tane model oluşturulur. Bu yöntem genellikle örnek sayısı az olan veri kümeleri için tercih edilmektedir [8].

2.2 Değerlendirme Ölçütleri

Sınıflandırma yöntemlerinin nasıl bir performans sergilediğini anlayabilmek ve değerlendirebilmek için kullanılan birtakım ölçütler mevcuttur. Bu ölçütler, sınıflandırma işlemi sonrasında test kümesine ait örnekler kullanılarak oluşturulan hatalı sınıflandırma matrisi yardımıyla hesaplanır. Tablo 2.1.'de görüldüğü gibi, hatalı sınıflandırma matrisinin her bir satırı öngörülen sınıfa ait örnekleri temsil ederken her bir sütunu gerçek sınıfa ait örnekleri temsil etmektedir.

Gerçek pozitif, sınıfı A olarak tahmin edilen bir örneğin A sınıfına ait olduğunu, gerçek negatif, sınıfı B olarak tahmin edilen bir örneğin B sınıfına ait olduğunu, yanlış pozitif (Tip-1 hata), sınıfı A olarak tahmin edilen bir örneğin B sınıfına ait olduğunu,

Tablo 2.1. *Hatalı sınıflandırma matrisi*

		Gerçek Sınıf	
		A	**B**
Tahmin Edilen Sınıf	**A**	Gerçek pozitif (gp)	Yanlış pozitif (yp)
	B	Yanlış negatif (yn)	Gerçek negatif (gn)

yanlış negatif (Tip-2 hata), sınıfı B olarak tahmin edilen bir örneğin A sınıfına ait olduğunu ifade etmektedir. Bu değerler kullanılarak aşağıdaki formüller yardımıyla hesaplanan ölçütler sayesinde kullanılan sınıflandırma yönteminin başarısı hakkında değerlendirme yapılabilmektedir.

Tahminleme Başarısı: Gerçek pozitif ve gerçek negatif tahmin sayılarının toplamının tüm tahminlere bölünmesiyle bulunan bu değer modelin yeni veya önceden görülmemiş verilerin sınıf etiketini doğru şekilde tahmin etme yeteneğini ifade eder.

$$Tahminleme\ başarısı = \frac{gp + gn}{gp + yp + gn + yn} \tag{2.1}$$

Hatalı Sınıflandırma Oranı: Yanlış pozitif ve yanlış negatif tahmin sayılarının toplamının tüm tahminlere bölünmesiyle bulunan bu değer sınıflandırma yönteminin ne sıklıkla hata yaptığını gösterir. Aynı zamanda $(1 - Tahminleme\ başarısı)$ değerine eşittir.

$$Hatalı\ sınıflandırma\ oranı = \frac{yp + yn}{gp + yp + gn + yn} \tag{2.2}$$

Gerçek Pozitif Oranı: Gerçek pozitif tahmin sayısının gerçek pozitif ve yanlış negatif tahmin sayılarının toplamına bölünmesiyle bulunan bu değer sınıflandırma yönteminin pozitif bir örneği ne sıklıkla pozitif olarak tahminlediğini ifade eder.

$$Gerçek\ pozitif\ oranı = \frac{gp}{gp + yn} \tag{2.3}$$

Yanlış Negatif Oranı: Yanlış negatif tahmin sayısının yanlış negatif ve gerçek pozitif tahmin sayılarının toplamına bölünmesiyle bulunan bu değer sınıflandırma yönteminin negatif bir örneği ne sıklıkla pozitif olarak tahminlediğini ifade eder.

$$Yanlış\ negatif\ oranı = \frac{yn}{yn + gp} \tag{2.4}$$

Kesinlik: Gerçek pozitif tahmin sayısının gerçek pozitif ve yanlış pozitif tahmin sayılarının toplamına bölünmesiyle bulunan bu değer sınıflandırma yöntemi tarafından pozitif olarak tahminlenen bir örneği ne sıklıkla doğru tahminlediğini ifade eder.

$$Kesinlik = \frac{gp}{gp + yp} \tag{2.5}$$

f değeri: Kullanılan sınıflandırma yönteminin performansını değerlendirmek için tek bir değerin kullanılması bazı durumlarda daha yararlı olabilir. Bu gibi durumlarda gerçek pozitif oranı ile kesinliğin harmonik ortalaması alınarak hesaplanan f değeri kullanılabilir.

$$f = \frac{2 \times gerçek\ pozitif\ oranı \times kesinlik}{gerçek\ pozitif\ oranı + kesinlik} \tag{2.6}$$

3. ÖZELLİK SEÇİMİ

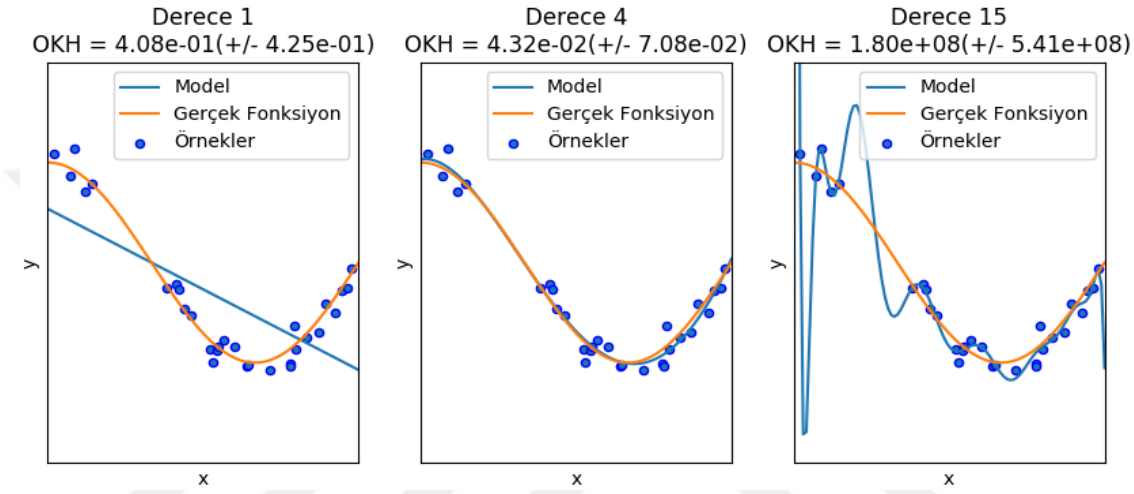
Gerçek hayat problemlerinde çoğunlukla birbiriyle ilişkili özellikler hakkında yeterli bilgi bulunmaz. Bu nedenle, verilerin kaynağını daha iyi temsil edebilmek için pek çok aday özellik belirlenir. Ancak bu durum da gereksiz özelliklerin seçimi ile sonuçlanır. Gereksiz özellikler bağımlı değişkenle direkt olarak ilişkili olmayan özelliklerdir ancak öğrenme sürecini etkiler ve amaca herhangi bir katkı sağlamaz. Günümüzde pek çok sınıflandırma probleminde kullanılan veri büyük boyutlu olduğundan istenmeyen özellikler çıkarılmadan iyi bir sınıflandırıcı elde etmek zordur. Gereksiz ya da ilgisiz özelliklerin sayısının düşürülmesi öğrenme algoritmasının hem çalışma süresini kısaltır hem de genelleştirme başarısının daha yüksek olmasını sağlar. Böylelikle gerçek hayat sınıflandırma problemine daha iyi bir yaklaşım ve bakış açısı geliştirilmiş olur.

Özellik seçimi problemi, orijinal veri kümesinin sahip olduğu özelliklerden, öğrenme algoritması sadece bu özellikleri içeren veri üzerine uygulandığında mümkün olan en yüksek tahminleme başarısına ulaşacak sınıflandırıcıyı oluşturan özellik alt kümesini seçmeyi amaçlar. Özellik seçimi var olan özellikler arasından bir özellik alt kümesi seçer, dolayısıyla özellik seçiminde özellik çıkarımı veya özellik oluşturma aşamaları bulunmaz.

Özellik seçimi genellikle sınıflandırma probleminin eğitim aşamasını kapsar. Özellikler belirlendikten sonra, öğrenme algoritmasını tüm özelliklere uygulamak yerine uygun özelliklerden oluşan alt kümeyi elde edebilmek için öncelikle özellik seçimi gerçekleştirilir ve daha sonra veriler belirlenen uygun özelliklere göre öğrenme algoritması ile işlenir. Özellik seçimi aşaması, filtre yöntemde olduğu gibi öğrenme algoritmasından bağımsız olabilir veya sarmal yöntemde olduğu gibi seçilen özelliklerin amaç ile ilgili olup olmadığını değerlendirmek için öğrenme algoritmasını iteratif olarak kullanabilir. Sonuç olarak, en son seçilen özellikler ile tahmin aşaması için bir sınıflandırıcı oluşturulur.

Özellik seçimi problemi hangi özelliklerin seçileceğine ve seçilen özelliklerin hangi kombinasyonunun kullanılacağına karar vermek olarak iki alt kısma bölünebilir. Hangi özelliklerin seçileceği sorusu özelliklerin ilgisinin neye göre ve nasıl tariflendiği ile alakalıdır. Literatürde özellik seçimi için özelliklerin ilgisi *güçlü ilgili, zayıf ilgili* ve *ilgisiz* olmak üzere üç farklı dereceli tanımlama yapılmıştır [9]. Herhangi bir X özelliği için özellik alt kümesinden bu özelliğin çıkarılması sınıflandırma algoritmasının performansında düşüş ile sonuçlanıyorsa bu özelliğe amaç ile *güçlü ilgili* denir. Herhangi

bir X özelliği güçlü ilgili değilse ve sınıflandırma algoritmasının $S \cup \{X\}$ özellik alt kümesindeki performansı S özellik alt kümesindeki performansından daha iyi olacak şekilde bir S alt kümesi varsa bu özelliğe *zayıf ilgili* denir. Son olarak bir özellik ne güçlü ilgili ne de zayıf ilgili tanımına girmiyorsa bu özelliğe amaç ile *ilgisiz* denir. Özetlemek gerekirse, güçlü ilgili bir özellik alt kümeden çıkarıldığında tahminleme başarısının düşmesi kaçınılmazdır, zayıf ilgili bir özellik ise bazı durumlarda tahminleme başarısına pozitif yönde etki edebilir [9].



Şekil 3.1. *Üç farklı dereceli polinom kullanılarak oluşturulan modeller* [10]

Optimal özellik alt kümesi biricik olmak zorunda değildir, çünkü bazı durumlarda aynı tahminleme başarısına farklı bir özellik alt kümesi kullanılarak da ulaşılabilir. Örneğin; amaç üzerinde birbirine çok benzer etki yaratan iki farklı özellik birbiri yerine kullanılabilir [9].

3.1 Sınıflandırma Problemlerinde Özellik Seçimi

Son otuz yıldır, veri madenciliği ve makine öğrenmesinde kullanılan verilerin boyutsallığında çarpıcı bir artış meydana gelmektedir. Verinin büyük boyutlu olması kurulan modelin gereksiz bileşenleri içermeye açık olmasını kolaylaştırdığından öğrenme algoritmasını *aşırı uyuma* eğilimli hale getirir. Aşırı uyum sorunu algoritmanın eğitim kümesindeki gürültülü noktalara uyumlu hale gelmesini ifade eder ve algoritmanın aslında tüm veride var olmayan gereksiz bir örüntü (çıkarım) elde etmesine sebep olur, bu da test kümesinden elde edilen başarının düşmesine yol açar. Ayrıca aşırı uyum oluştuğunda göz önüne alınan sınıflandırma problemi daha basit bir model ile yeteri kadar ifade edilebilecek iken oldukça karmaşık modellerin elde edilmesine sebep olur.

Şekil 3.1.'de bir sınıflandırma problemi için sırasıyla 1, 4 ve 15. dereceden polinomlar kullanılarak oluşturulan üç farklı model görülmektedir. Kurulan modeller arasından veriyi en iyi ifade eden 15. dereceden bir polinomdan oluşturulan model gibi gözükse de gerçek fonksiyona göre hesaplanan ortalama karesel hatanın en büyük değeri bu model ile elde edilmiştir.

Bu durum algoritmanın eğitim kümesinde bulunan gürültülü noktalara uyumlu hale gelmesi ile olması gerekenden daha esnek bir model oluşturduğunu gösterir. Şekilde de görüldüğü gibi model, 15. dereceden bir polinom yerine 4. dereceden bir polinom ile ifade edildiğinde en düşük hata değerine ulaşılmaktadır.

Sınıflandırma yapılacak veri kümesinin boyutsallığı çok büyük değerlere ulaştığında (özellikle boyut sayısının veri sayısından daha fazla olduğu durumlarda) algoritma, veriyi doğru tahminleyecek yeterli bilgiye ulaşamadığından gereksiz örüntüleri elde etmeye daha yatkın hale gelir. Başka bir ifadeyle, eğitim aşamasında yeterli veri bulunmadığından eğitim kümesindeki gürültülü noktaların kullanılmasıyla tüm veride var olmayan örüntüler elde edilir. Bu örüntüler test aşamasında algoritmanın daha önce karşılaşmadığı veri ile sınandığında tahminleme başarısının düşmesine neden olur. Bu nedenle aşırı uyum sorunu ortaya çıktığında eğitim başarısı test başarından çok daha yüksek olan sonuçlar elde edilir. Dolayısıyla özellik seçimi tekniklerini kullanmak gürültülü ve ilgisiz verileri olabildiğince uzaklaştıracağından başarı oranı yüksek tahminlemeler yapabilmek için oldukça önemlidir.

Özellik seçimi, özelliklerin bağımlı değişken ile ilgisini değerlendiren bir kritere göre orijinal özellik kümesinden küçük bir alt küme seçmeyi amaçlar. Böylece daha iyi öğrenme performansı (sınıflandırma için yüksek eğitim başarısı), daha düşük hesaplama maliyeti ve daha kolay yorumlanabilir modellerin elde edilmesi amaçlanır.

Özellik seçimi algoritmaları eğitim kümesindeki verilerin etiketlerinin bilinip bilinmemesine göre denetimli, denetimsiz ve yarı-denetimli olmak üzere kategorize edilebilir. Denetimli özellik seçimi verilerin sahip olduğu etiketlerin bilindiği ve özelliklerin ilişkilerinin bu etiket bilgilerine göre değerlendirildiği yöntemlerdir. Denetimsiz özellik seçimi, kümeleme ölçütlerine dayanan sınıf etiketlerinin bulunmadığı daha az kısıta sahip bir arama problemidir.

Denetimli özellik seçimi yöntemleri genel olarak filtre yöntem, sarmal yöntem ve gömülü yöntem olarak kategorize edilebilir. **Filtre yöntem,** özellik seçimini sınıflandırıcıdan ayıran, öğrenme algoritması ile özellik seçimi algoritmasının etkileşime

girmesine izin vermeyen bir yöntemdir. Filtre yöntemleri, her özelliğe bir ağırlık atayabilmek için istatistiksel bir ölçü kullanır. Özellikler ağırlıklarına göre sıralanır ve veri kümesinde kalıp kalmama durumlarına karar verilir. Kullanılan istatistiksel ölçümler aşağıdaki tablodaki gibidir.

Tablo 3.1. *Filtre yöntemlerde kullanılan istatistiksel ölçümler*

Özellik\Çıktı	**Sürekli**	**Kategorik**
Sürekli	Pearson Korelasyon	LDA
Kategorik	Anova	Ki-Kare

Sarmal yöntem, seçilen özelliklerin bağımlı değişkeni açıklama kabiliyetini belirlemek için önceden karar verilmiş bir öğrenme algoritmasının tahminleme başarısını kullanır. Yöntem, her adımda bir önceki modelin çıkarımlarına dayanarak elde edilen alt kümedeki özellikleri eklemeye veya çıkarmaya karar verir. Böylelikle problem bir arama problemine dönüşmüş olur. Bu nedenle, sarmal yöntemlerin çok sayıda özellik içeren verilerde kullanımı oldukça maliyetlidir. Sarmal yöntemde kullanılan en yaygın arama yöntemleri; *ileriye doğru özellik ekleme, geriye doğru özellik eleme* ve *rekursif özellik eleme*dir.

- **İleriye doğru özellik ekleme:** İleriye doğru özellik ekleme, hiç özellik bulunmayan bir model ile başlanan tekrarlı bir yöntemdir. Her iterasyonda, modelin performansını iyileştirmeyen yeni bir değişkene kadar modeli en çok iyileştiren özellik eklenmeye devam edilir.
- **Geriye doğru özellik eleme:** Geriye doğru özellik eleme yönteminde tüm özelikler ile başlanır ve modelin performansını iyileştiren en az öneme sahip özellik her iterasyonda modelden çıkarılır. Özelliklerin modelden çıkarılmasında herhangi bir iyileşme gözlenmeye dek bu işleme devam edilir.
- **Rekursif özellik eleme:** Rekursif özellik eleme yöntemi en iyi performans gösteren özellik alt kümesini bulmayı amaçlayan bir açgözlü optimizasyon algoritmasıdır. Modeli tekrar tekrar oluşturur ve her iterasyonda en iyi ya da en kötü performans gösteren özellikleri modelden çıkarır. Tüm özellikler bitene kadar bir sonraki modeli kalan özellikler ile oluşturur. Sonrasında özellikler eleme sırasına göre sıralanır.

Gömülü yöntemde, hangi özelliklerin seçileceğine model oluşturulurken karar verildiğinden öğrenme algoritması ile özellik seçimi birbirinden ayrılamaz. Başka bir deyişle, sınıflandırma ve özellik seçimi eş zamanlı olarak gerçekleştirilir. Bu nedenle, gömülü yöntemlerde kullanılan sınıflandırma algoritması büyük öneme sahiptir.

Denetimli özellik seçiminde özelliklerin ilişkisi etiket bilgilerine göre belirlenir ama iyi bir özellik seçimi algoritması elde edebilmek için yeterli sayıda etiketli veri bulunması gerekir bu da zaman alıcı bir işlemdir. Denetimsiz özellik seçiminde ise etiketleri bulunmayan veri ile çalışılır ama özelliklerin ilişkisini değerlendirmek zordur. Günümüzde genellikle veri kümeleri büyük boyutlu ancak az sayıda etiketli örneklem içeren kümelerden oluşur. Etiketli örnek büyüklükleri küçük olan yüksek boyutlu veriler çok fazla kısıtlamaya sahip ancak çok büyük bir hipotez alanına sahiptir. İki veri özelliğinin kombinasyonu yeni bir araştırma zorluğu ortaya çıkarmaktadır. Etiketli ve etiketsiz verilerin aynı popülasyondan örneklendiği varsayımıyla, yarı denetimli özellik seçimi, özelliklerin birbirleriyle ilişkisini tahmin etmek için hem etiketli hem de etiketsiz verileri kullanmaktadır.

Özellik ağırlıklandırma, özellik seçiminin genelleştirmesi olarak düşünülebilir. Özellik seçiminde 1 özelliğin seçildiğini, 0 ise seçilmediğini ifade edecek şekilde her bir özelliğe ikili değerlendirme uygulanır. Ancak özellik ağırlıklandırmada her bir özelliğe $[0,1]$ veya $[-1,1]$ aralıklarından bir değer atanır. Bu değer ne kadar büyük ise özellik o kadar baskındır. Pek çok özellik ağırlıklandırma algoritması her bir özelliğe global bir ağırlık değeri atar. Ancak özelliklerin birbiriyle ilişkisi ve gürültü yerel olarak önemli farklılıklar gösterebilir. *kNN* gibi tembel öğrenme algoritmalarında yaygın olan, özellik seçiminin bir test örneğine göre yerel olarak yapıldığı yerel özellik seçimi algoritmaları da vardır. Burada, özellik seçimi veya ağırlıklandırma sınıflandırma ile birlikte yapılır (eğitim aşaması yerine), çünkü test örneği bilgisi özellikleri seçme yeteneğini arttırır.

4. SINIFLANDIRMA PROBLEMLERİ İÇİN MATEMATİKSEL PROGRAMLAMA TEMELLİ ÇÖZÜM YÖNTEMLERİ

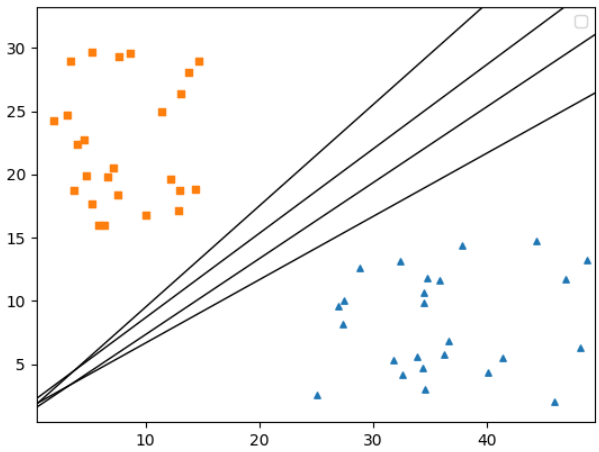
4.1 Gürbüz Doğrusal Programlama Yöntemi

Bennett ve Mangasarian tarafından geliştirilen bu yöntem [11], doğrusal ayrılamayan ve kesişimleri boş küme olan sonlu iki nokta kümesi için yanlış sınıflandırılan noktaları minimize edecek şekilde bir ayırıcı hiperdüzlem bulunmasını amaçlamaktadır.

$\mathcal{A}, \mathcal{B} \in \mathbb{R}^n$ olmak üzere sırasıyla m ve k elemanlı kesişimleri boş küme olan sonlu iki nokta kümesi olsun. Bu kümeleri $A \epsilon \mathbb{R}^{mxn}$ ve $B \epsilon \mathbb{R}^{kxn}$ matrisleri ile ifade edelim. Bu durumda A matrisindeki her bir satır $\mathcal{A}$ kümesine ait bir noktaya ve aynı şekilde B matrisindeki her bir satır da $\mathcal{B}$ kümesine ait bir noktaya karşılık gelir. Eğer $\mathcal{A}$ ve $\mathcal{B}$ kümelerinin konveks örtülerinin kesişimi boş küme ise bu kümeler doğrusal ayrılabilirdir. Matematiksel olarak ifade edersek, bu iki kümenin doğrusal ayrılabilmesi için gerek ve yeter şart,

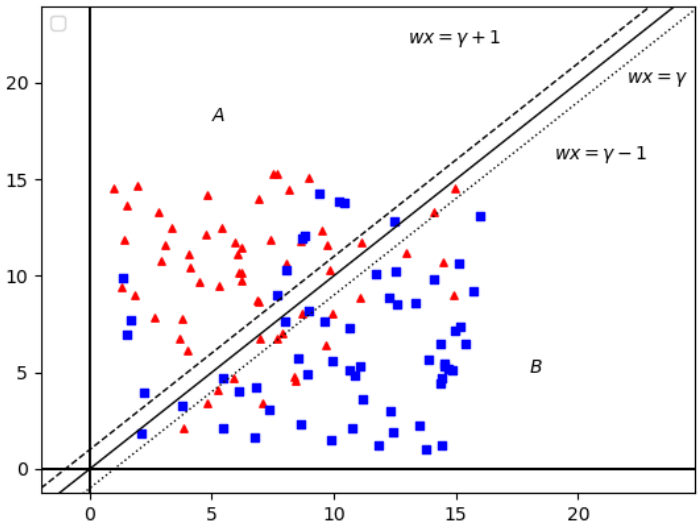
$$\underset{1 \le i \le m}{enk}\ A_i\, v > \underset{1 \le i \le k}{enb}\ B_i v\,, \qquad v = (v_1, \dots, v_n) \epsilon \mathbb{R}^n \tag{4.1}$$

olmasıdır. Şekilde de görüldüğü gibi doğrusal olarak ayrılabilen iki nokta kümesi için birden fazla ayırıcı yüzey bulabilmek mümkündür.



Şekil 4.1. *Doğrusal ayrılabilen veri kümeleri*

Doğrusal ayrılabilen iki nokta kümesi için her durumda en az bir ayırıcı hiperdüzlem bulmak mümkün olsa da gerçek hayat problemlerinin pek çoğu doğrusal ayrılamayan kümeler içermektedir. Aşağıdaki şekilde doğrusal ayrılamayan iki nokta kümesi görülmektedir. Şekilde de görüldüğü gibi bu iki kümenin en büyük değeri ve en



Şekil 4.2. *Doğrusal ayrılamayan veri kümeleri*

küçük değeri ile yukarıdaki eşitsizlik sağlanmaz. Dolayısıyla şekilde görülen kümeleri hatalı noktalar olmaksızın ayıracak bir hiperdüzlem bulmak mümkün değildir.

Şekilde görülen $\mathcal{A}$ ve $\mathcal{B}$ kümelerini, hatalı noktaları minimize edecek şekilde ayıran P hiperdüzlemi aşağıdaki gibi tariflenebilir.

$$P = \{x | x \in \mathbb{R}^n, x^T w = \gamma\} \tag{4.2}$$

Burada, $w = (w_1, \dots, w_n)$ ayırıcı yüzeyin normalini ifade eden n boyutlu vektör ve γ ayırıcı yüzeyin alt sınırını ifade eden reel sayıyı göstermektedir. O halde, P hiperdüzlemini belirleyebilmek için w, γ değerleri bulunmalıdır.

P hiperdüzlemi $P^+ = \{x | x \in \mathbb{R}^n, x^T w > \gamma\}$ ve $P^- = \{x | x \in \mathbb{R}^n, x^T w < \gamma\}$ olmak üzere iki farklı açık yarı uzay tanımlar. Bu yarı uzaylardan P^+, daha çok $\mathcal{A}$ kümesine ait noktaları, P^- daha çok $\mathcal{B}$ kümesine ait noktaları içerecek şekilde ve yarı uzaylar arasındaki mesafe genişletilerek aşağıdaki eşitsizlikler elde edilir,

$$Aw \geq e\gamma + e, \qquad e\gamma - e \geq Bw, \qquad w \epsilon \mathbb{R}^n, \qquad \gamma \epsilon \mathbb{R} \tag{4.3}$$

Burada e ifadesi uygun boyuttaki birler vektörünü göstermektedir. Bu eşitsizlikler doğrusal ayrılabilen kümeler için sağlanır, doğrusal ayrılamayan kümeler içinse Bennett ve Mangasarian tarafından yanlış sınıflandırılan noktaların ortalama sayısını minimize edecek şekilde bir hata fonksiyonu tariflenmiştir. Her iki küme için sırasıyla yanlış sınıflandırılan noktalar izleyen şekilde tanımlanmıştır.

$\mathcal{A}$ kümesi için yanlış sınıflandırılan herhangi bir A' noktası $x^T w = \gamma + 1$ düzleminin alt tarafında kalan bir A noktasıdır ve $\mathcal{B}$ kümesi için yanlış sınıflandırılan

herhangi bir B' noktası ise $x^T w = \gamma - 1$ düzleminin üst tarafında kalan bir B noktasıdır. O halde bu noktalar,

$$A'w \leq \gamma + 1, \qquad B'w \geq \gamma - 1 \tag{4.4}$$

eşitsizliklerini sağlar.

Amaç bu hatalı noktaların sayısını minimize etmek olduğundan hata fonksiyonu aşağıdaki gibi oluşturulmuştur.

$$\underset{w,\gamma}{enk}\, f(w,\gamma) = \underset{w,\gamma}{enk} \frac{1}{m}\|(-Aw + e\gamma + e)_+\|_1 + \frac{1}{k}\|(Bw - e\gamma + e)_+\|_1 \tag{4.5}$$

Burada x_+ ifadesi herhangi bir x vektörü için $x_+ = \text{enb}\{0, x_i\}$ olacak şekilde tariflenmiştir. Buna göre yukarıdaki fonksiyonda parantez içinde kalan ifadelerin sıfırdan küçük değerleri sırasıyla $\mathcal{A}$ ve $\mathcal{B}$ kümelerine ait doğru sınıflandırılmış noktalardır. Tariflenen x_+ fonksiyonu sayesinde doğru sınıflandırılmış noktalar için 0 değeri, yanlış sınıflandırılmış noktalar için o noktanın hata değeri elde edilir. 1-normu ile de bu hata değerlerinin toplamı bulunmuş olur. Böylece tüm noktalar için elde edilen toplam hata değeri kümelerdeki eleman sayıları ile ağırlıklandırılarak enküçüklenmiş olur.

İlgili makalede [11] yukarıdaki gibi bir norm enküçükleme problemi için aşağıdaki şekilde norm içermeyen bir kısıtlı optimizasyon problemi yazılabileceği gösterilmiştir,

$$\underset{w,\gamma,y,z}{enk} \left\{ \frac{ey}{m} + \frac{ez}{k} \middle| Aw - e\gamma + y \geq e, -Bw + e\gamma + z \geq e, y \geq 0, z \geq 0 \right\}. \tag{4.6}$$

Böylece sınıflandırma problemi için açık şekli aşağıdaki gibi olan gürbüz bir doğrusal programlama modeli elde edilmiştir.

$$\begin{gathered} -Aw + e\gamma + e \leq y, \\ Bw - e\gamma + e \leq z, \\ y \geq 0, z \geq 0 \\ \text{kısıtları altında} \\ \underset{w,\gamma,y,z}{enk} \frac{e^T y}{m} + \frac{e^T z}{k} \end{gathered} \tag{4.7}$$

Böylece doğrusal programlama modelinin amaç fonksiyonu, A kümesine ait hatalı noktaların $x^T w = \gamma + 1$ hiperdüzlemine, B kümesine ait hatalı noktaların $x^T w = \gamma - 1$ hiperdüzlemine olan toplam ortalama uzaklıklarını ($\|w\|'$ vektörü ile ağırlıklandırarak) enküçükleyecek şekilde oluşturulmuştur. Optimal ayırıcı yüzey ise bu iki hiperdüzlemin ortasından geçen, onlara paralel olan ve aynı parametreler kullanılarak oluşturulan $x^T w = \gamma$ hiperdüzlemidir.

4.2 Çok Yüzlü Konik Fonksiyonlar Algoritması

Kasımbeyli ve Öztürk tarafından geliştirilen bu yöntem, kesişimleri boş küme olan sonlu iki farklı nokta kümesi için ayırıcı yüzeyleri çok yüzlü konik fonksiyon (çkf) olarak adlandırılan fonksiyonlar yardımıyla tarifleyen matematiksel programlama temelli aç gözlü bir algoritmadır [3].

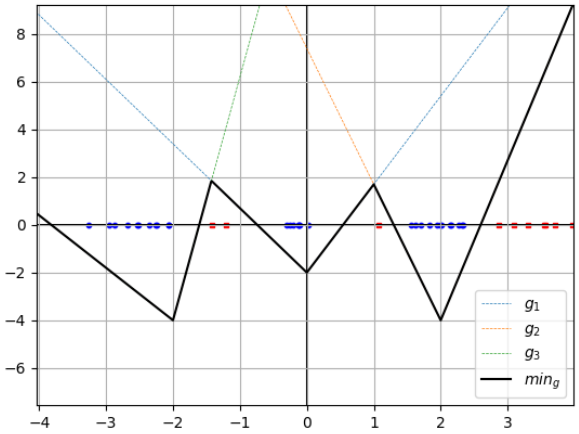
$A, B \in \mathbb{R}^n$ gibi kesişimleri boş küme olan iki farklı küme için algoritma, A kümesinden herhangi bir noktayı (merkez) seçerek başlar. Bu noktaya göre olabildiğince A kümesine ait noktaları içeren ve B kümesine ait hiç nokta içermeyen bir konik fonksiyon oluşturmak amaçlanır. Bir sonraki adımda A kümesinden koninin iç kısmında kalan, doğru sınıflandırılan A noktaları, çıkarılarak küme güncellenir ve kalan A noktalarından yeni bir merkez noktası seçilir. Algoritma bu şekilde A kümesinde hiç nokta kalmayana kadar devam eder ve oluşturulan konik fonksiyonların en küçük değeri alınarak optimal ayırıcı yüzey elde edilir.

Algoritmada kullanılan çok yüzlü konik fonksiyon $g_{(w,\xi,\gamma,\alpha)} : \mathbb{R}^n \to \mathbb{R}$ olmak üzere

$$g_{(w,\xi,\gamma,a)}(x) = w'(x-a) + \xi\|x-a\|_1 - \gamma \tag{4.8}$$

şeklinde tanımlanmaktadır. Burada $w, a \in \mathbb{R}^n,\ \xi, \gamma \in \mathbb{R}, w'x = w_1x_1 + \cdots + w_nx_n$ ifadesi w ile x vektörlerinin skaler çarpımını ve $\|x\|_1 = |x_1| + \cdots + |x_n|$ ifadesi ise x vektörünün 1-normunu göstermektedir.

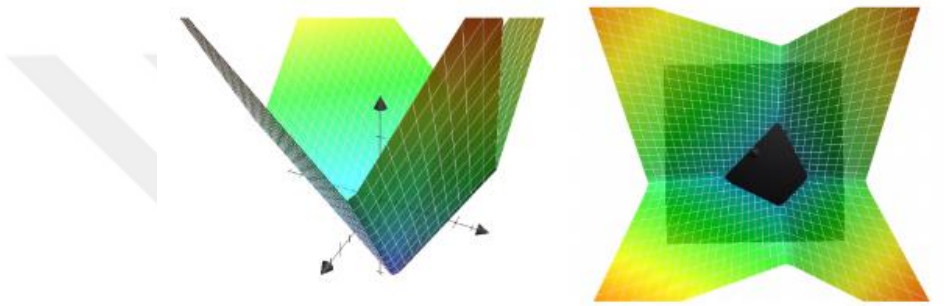
Böyle bir $g_{(w,\xi,\gamma,a)}(x)$ fonksiyonunun grafiği tepe noktası $(a, -\gamma) \in \mathbb{R}^n \times \mathbb{R}$ olan bir konidir ve bu fonksiyonun her seviye kümesi konveks bir polihedrondur. Şekil 4.3.'de aşağıda denklemleri görülen $\mathbb{R}^2$ de tanımlı üç farklı çok yüzlü konik fonksiyona ait grafikler görülmektedir. Şekil 4.4.'de ise $\mathbb{R}^3$ te tanımlı birçok yüzlü konik fonksiyona ait grafik ve seviye kümesi görülmektedir [12].



Şekil 4.3. *Üç farklı çkf'ye ait grafikler ve onların minimumu*

$$g_1 = 0{,}5(x-0) + 3{,}2\|x-0\|_1 - 2$$
$$g_2 = 0{,}5(x-2) + 6{,}2\|x-2\|_1 - 4$$
$$g_3 = 4(x+2) + 6{,}2\|x+2\|_1 - 4 \quad (4.9)$$

Algoritma, çok yüzlü konik fonksiyonların oluşturduğu bu seviye kümelerini kullanarak veri noktalarını ayırmayı amaçlar. İlk şekilde görülen veri noktaları için A kümesine ait noktalar oluşturulan çok yüzlü konik fonksiyonların içinde kalan, seviye kümesinde olan, noktalardır. B kümesine ait noktalar ise oluşturulan çok yüzlü konik fonksiyonların dışında kalan, seviye kümesinde olmayan, noktalardır. Buna göre sırasıyla A kümesine ait noktalar -1, B kümesine ait noktalar 1 etiketi kullanarak sınıflandırılır.



Şekil 4.4. *Üç boyutlu bir çkf grafiği* [12]

Tanımlanan ve özellikleri ifade edilen çok yüzlü konik fonksiyona göre oluşturulan ikili sınıflandırma algoritması izleyen şekildedir,

A ve B kümeleri $\mathbb{R}^n$ de

$$A = \{a^i \in \mathbb{R}^n : i \in I\}, I = \{1, \dots, m\},$$
$$B = \{b^j \in \mathbb{R}^n : j \in J\}, = \{1, \dots, p\},$$

şeklinde tanımlı iki küme olsun.

Başlangıç Adımı: $l = 1, l_1 = I, A_l = A$ atamalarını yap ve Adım 1'e git.

Adım 1: a^l noktası, A_l kümesinin herhangi bir noktası olsun. P_l problemini çöz:

$$w^T(a^i - a^l) + \xi \left\|a^i - a^l\right\|_1 - \gamma + 1 \le y_i, \quad \forall_i \in I_l, \quad (4.10)$$
$$-w^T(b^i - a^l) - \xi \left\|b^i - a^l\right\|_1 + \gamma + 1 \le 0, \quad \forall_j \in J, \quad (4.11)$$
$$y = (y_1, \dots, y_m) \in R_+^m, w = (w_1, \dots, w_n) \in R^n, \xi \in [0, \infty), \gamma \in [1, \infty)$$

kısıtları altında;

$$(P_l) \quad enk\left(\frac{ye_m}{m}\right) \quad (4.12)$$

P_l probleminin bir çözümünü $(w^l, \xi^l, \gamma^l, y^l)$ olacak şekilde bul. Bu çözüme karşılık gelen çok yüzlü konik fonksiyonu aşağıdaki gibi oluştur,

$$g_l(x) = w^l(x - a^l) + \xi^l \|x - a^l\|_1 - \gamma^l \tag{4.13}$$

ve Adım 2'ye geç.

Adım 2: $I_{l+1} = \{i \in I_l : g_l(a^i) + 1 > 0\}, A_{l+1} = \{a^i \in A_l : i \in I_{l+1}\}, l = l + 1$ güncellemelerini yap. Eğer $A_l \neq \emptyset$ ise Adım 1'e git.

Adım 3: A ve B kümelerini ayıran $g(x)$ fonksiyonunu aşağıdaki gibi tanımla ve dur.

$$g(x) = \underset{l}{enk}\, g_l(x) \tag{4.14}$$

Algoritma her l adımında, A_l kümesinden herhangi bir a^l elemanını seçerek başlar ve P_l problemini çözüp (w^l, ξ^l, γ^l) parametrelerini oluşturur. P_l probleminin çözümüyle elde edilen bu parametreler g_l çok yüzlü konik fonksiyonunun oluşturulması için kullanılır. Algoritma (4.10) kısıtı sayesinde, içerisinde olabildiğince A_l kümesine ait noktalar bulunan ve aynı zamanda (4.11) kısıtı ile içerisinde B kümesine ait hiçbir nokta bulunmayan bir g_l fonksiyonu oluşturur. $\xi \in [0, \infty), \gamma \in [1, \infty)$ kısıtları sayesinde kolları yukarı bakan ve tepe noktası $z = 0$ hiperdüzleminin altında kalan bir koni elde edilmesi sağlanır. Böylece A kümesine ait noktaların istenilen seviye kümelerinde olması sağlanır. Herhangi bir noktanın hangi kümeye ait olduğunu belirlemek için, bu noktanın elde edilen sonlu sayıdaki çok yüzlü konik fonksiyon değerine bakılır. Bu değer, eğer sıfır ya da sıfırdan küçükse noktanın A kümesine ait olduğu, eğer sıfırdan büyükse noktanın B kümesine ait olduğu ifade edilir.

Çok yüzlü konik fonksiyonlar algoritması, ortaya konulduğu makaleden ve daha sonraki çalışmalardan görülebileceği üzere sınıflandırma problemlerinin çözümünde oldukça başarılı sonuçlar vermektedir. Ayrıca algoritma detaylı olarak incelendiğinde, zayıf yönlerinin merkez seçimine duyarlılığı ve aşırı uyum sorununa yatkınlığı olduğu görülebilir.

Algoritma seçtiği bir nokta yardımıyla tepe noktasını belirlediği çok yüzlü konik fonksiyonun, yine bu nokta etrafında kalan A kümesine ait noktaları olabildiğince içerisine almaya çalışarak iki kümeyi ayırmayı amaçladığından merkez seçimi oldukça önemlidir. Merkez seçimini daha verimli hale getirebilmek için ilgili makalede ayrıca bir yöntem sunulmuştur. Bu yöntem ilerleyen kısımlarda detaylı bir şekilde ele alınacaktır.

Algoritmanın aç gözlü yapısı gereği eğitim aşamasında, A kümesine ait her bir noktayı içerisine alacak şekilde çok yüzlü konik fonksiyonlar oluşturulur. Bu durumda

eğitim kümesindeki gürültülü noktalar dahil her nokta öğrenildiğinden eğitim başarısı %100 bulunur. Fakat test aşamasına geçildiğinde, eğitim kümesinde bulunup test kümesinde bulunmayan örüntüler için gereksiz bir çıkarım yapıldığından algoritmanın test başarısı düşer. Dolayısıyla algoritma gürültülü noktalara uyum sağlar ve eğitim başarısı ile test başarısı arasında büyük farklar meydana geldiğinden aşırı uyum sorunu ortaya çıkar.

Aşırı uyum sorunu ile paralel olarak gelişen eğitim başarısı ve test başarısı arasındaki büyük farkı azaltabilmenin en pratik yolu kısıt (4.11)'i değiştirip B kümesine ait noktalar için hataya izin vermektir. Böylelikle algoritma B kümesine ait her noktayı dışarıda bırakacak şekilde çok yüzlü konik fonksiyon çizmeye zorlanmaz ve tüm veri kümesinde bulunmayan gereksiz örüntülere uyum sağlaması azaltılmış olur. Fakat bu durumda da test başarısında düşüş meydana gelir. ÇKF algoritmasında, merkez seçimine olan duyarlılığı azaltmak, aşırı uyum sorununu ortadan kaldırmak ve daha yüksek tahminleme başarısı elde etmek için Kasımbeyli ve Öztürk tarafından aynı çalışma kapsamında [3] geliştirilmiş farklı bir versiyon daha bulunmaktadır. Sözü edilen algoritma modifiye ÇKF algoritması olarak adlandırılmıştır ve ÇKF algoritmasının bu versiyonu sayesinde daha verimli merkez seçimi, daha az alt problem çözme ve daha yüksek sınıflandırma başarısı elde etme sonuçlarına ulaşılmıştır. Bir sonraki bölümde, çok yüzlü konik fonksiyonlar algoritması için önerilen bu daha verimli merkez seçme prosedürü basit bir ön işleme adımı tariflenerek tez kapsamında önerilen özellik seçimi algoritmasına uyarlanmıştır ve her iki özellik seçimi yöntemi arasında detaylı bir karşılaştırma sunulmuştur.

5. SINIFLANDIRMA PROBLEMLERİ İÇİN ÖZELLİK SEÇİMİ YÖNTEMLERİ

5.1 Matematiksel Programlama Temelli Özellik Seçimi Algoritması

Bradley ve Mangasarian tarafından geliştirilen bu yöntem [2] bir önceki bölümde detaylı bir şekilde anlatılan gürbüz doğrusal programlama yaklaşımı esas alınarak n boyutlu uzayda sonlu iki farklı nokta kümesini olabildiğince az sayıda özellik kullanarak ayırmayı amaçlamaktadır. Gürbüz doğrusal programlama yöntemi ile iki küme arasındaki optimal ayırıcı yüzey hiperdüzlem olarak tariflendiğinden özellik seçimi işlemi bu hiperdüzlemin normaline göre yapılmalıdır. Bunun için (4.7) ile verilen doğrusal programlama modelinin amaç fonksiyonuna, ayırıcı yüzeyin normalinin bileşenlerini olabildiğince sıfıra götürecek yeni bir terim eklenmiştir. Böylece optimal ayırıcı yüzeyin normalinin sıfıra eşit olan bileşenlerine karşılık gelen özellikler problemden çıkartılmış olur.

Yeni amaç fonksiyonu, eklenen özellik seçimi terimi, $\lambda \in [0,1)$ olmak üzere λ parametresi ile orijinal amaç fonksiyonu ise $(1-\lambda)$ ile ağırlıklandırılarak izleyen şekilde oluşturulmuştur:

$$\begin{gathered} -Aw + e\gamma + e \leq y, \\ Bw - e\gamma + e \leq z, \\ y \geq 0, z \geq 0, \lambda \in [0,1) \\ \text{kısıtları altında} \\ \underset{w,\gamma,y,z}{enk}\ (1-\lambda)\left(\frac{e^T y}{m} + \frac{e^T z}{k}\right) + \lambda e^T |w|_* \end{gathered} \tag{5.1}$$

Burada herhangi bir $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ vektörü için $|x|_*$ ifadesi, x vektörünün sıfırdan farklı bileşenleri için 1 değerini, sıfıra eşit olan bileşenleri için 0 değerini alacak şekilde tanımlanmıştır. O halde, e uygun boyutlu birler vektörü olmak üzere amaç fonksiyonuna eklenen $e^T|w|_*$ terimi ile ayırıcı yüzeyin normali, w vektörünün, sıfırdan farklı bileşenlerinin sayısı elde edilir. λ parametresi ile de iki farklı amaç, sınıflandırmadaki hatayı ve orijinal problemin özellik sayısını enküçüklemek arasında denge sağlanır. Ayrıca eğer w vektörünün herhangi bir bileşeni sıfıra eşit ise o bileşene karşılık gelen özellik problemden çıkarılmış olur.

Amaç fonksiyonu, eklenen özellik seçimi teriminde bulunan mutlak değerden kurtulabilmek için $|w|$ ifadesini temsil edecek şekilde $v = (v_1, \dots v_n) \in \mathbb{R}^n$ olan yeni bir değişken ve kısıt eklenerek aşağıdaki şekilde düzenlenir,

$$
\begin{gathered}
-Aw + e\gamma + e \leq y, \\
Bw - e\gamma + e \leq z, \\
-v \leq w \leq v, \\
y \geq 0, z \geq 0, \lambda \in [0,1), \\
\text{kısıtları altında} \\
\underset{w,\gamma,y,z}{enk}\ (1-\lambda)\left(\frac{e^T y}{m} + \frac{e^T z}{k}\right) + \lambda e^T v_*
\end{gathered} \tag{5.2}
$$

Bu doğrusal programlama modeli sayesinde orijinal problemden daha düşük boyutlu bir özellik kümesi elde edilir. Problemin çözümünde kullanılan özellik sayısı, $w = e^T v_*$ vektörünün boyutu, başlangıçtaki n değerinden daha küçüktür. (5.2) modelinden $\lambda = 0$ için gürbüz doğrusal programlama modeli elde edilir, $\lambda = 1$ için w vektörünün tüm bileşenleri çıkarılmaya çalışıldığından iyi bir sonuç elde edilmez. Bu nedenle λ parametresi, (5.2) doğrusal programlama probleminin çözümü ile elde edilen w, γ değerleriyle oluşturulan $x^T w = \gamma$ optimal ayırıcı yüzeyin genelleştirme başarısını enbüyükleyecek şekilde $(0,1)$ aralığından çapraz doğrulama yapılarak seçilebilir. İlgili makalede [13] (5.2) özellik seçimi probleminin her bir $\lambda \in [0,1]$ için bir çözümü olduğu gösterilmiştir.

(5.2) modelinde özellik seçimi için kullanılan ve $e^T v_*$ ile ifade edilen basamak fonksiyonu sürekli olmadığından problemi çözebilmek için, bu ifadeye yakınsayacak sürekli olan başka bir fonksiyona ihtiyaç vardır. Basamak fonksiyonu terimine yakınsayan bu fonksiyon

$$
v_* \approx t(v, \alpha) \coloneqq e - \varepsilon^{-\alpha v}, \alpha > 0, \tag{5.3}
$$

şeklinde tanımlanmıştır. Buna göre model

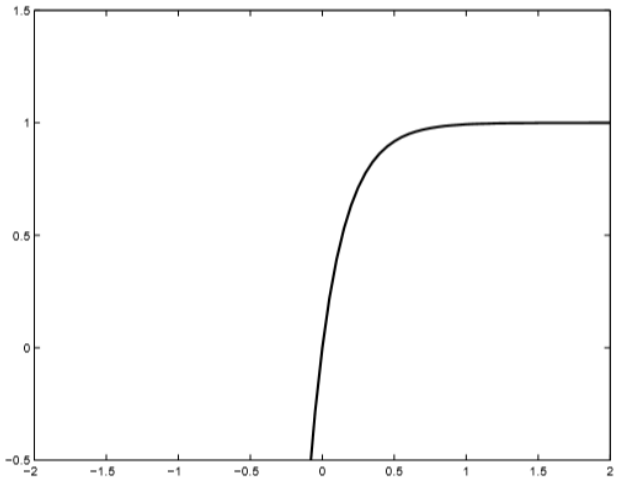
$$
\begin{gathered}
-Aw + e\gamma + e \leq y, \\
Bw - e\gamma + e \leq z, \\
-v \leq w \leq v, \\
y \geq 0, z \geq 0, \lambda \in [0,1), \\
\text{kısıtları altında} \\
\underset{w,\gamma,y,z}{enk}\ (1-\lambda)\left(\frac{e^T y}{m} + \frac{e^T z}{k}\right) + \lambda e^T (e - \varepsilon^{-\alpha v})
\end{gathered} \tag{5.4}
$$

şeklini alır.

Burada e uygun boyuttaki birler vektörünü, ε doğal logaritmanın tabanını ve α ise pozitif bir sayıyı göstermektedir. Önerilen $t(v, \alpha)$ fonksiyonunun seçilmesinde iki önemli özelliğinin bulunması rol oynamaktadır. Bu özellikler;

i. $t(v,\alpha)$ fonksiyonunun 0 noktasındaki değeri $t(0,\alpha)=0$ olduğundan $e^T v_*$ basamak fonksiyonuna tam olarak yakınsar [14].

ii. $t(v,\alpha)$ fonksiyonu şekil 5.1 ile görüldüğü gibi içbükey olduğundan (5.4) problemi çokyüzlü bir küme üzerinde içbükey bir amaç fonksiyonunu enküçükleme halini alır. Böylece sonlu adımda sonuçlanabilen bir algoritma yardımıyla çözülebilir [14].



Şekil 5.1. $t(x,5)=1-\varepsilon^{-5x}$ *fonksiyonunun grafiği* [14]

Bu probleme global bir çözüm bulmak oldukça zor olduğundan bunun yerine sonlu adımda gerekli optimallik koşullarını sağlayan bir kritik noktaya ulaşacak şekilde Frank-Wolfe algoritmasından esinlenen bir çözüm yaklaşımı önerilmiştir. Tarif edilen çözüm yaklaşımı aşağıda sunulmuştur:

Herhangi bir rastgele $(w^0,\gamma^0,y^0,z^0,v^0)$ başlangıç çözümü ile başla. Bu çözümü $(w^i,\gamma^i,y^i,z^i,v^i)$ olarak al, aşağıdaki lineer problemi çöz ve (5.6) eşitliğini kullanarak optimal $(w^{i+1},\gamma^{i+1},y^{i+1},z^{i+1},v^{i+1})$ çözümünü belirle.

$$\begin{gathered}-Aw+e\gamma+e\le y,\\ Bw-e\gamma+e\le z,\\ -v\le w\le v,\\ y\ge 0, z\ge 0\\ \text{kısıtları altında}\end{gathered}$$

$$\underset{w,\gamma,y,z,v}{enk}\ (1-\lambda)\left(\frac{e^T y}{m}+\frac{e^T z}{k}\right)+\lambda\alpha\left(\varepsilon^{-\alpha v^i}\right)^T\left(v-v^i\right) \qquad (5.5)$$

Eğer

$$(1-\lambda)\left(\frac{e^T\left(y^{i+1}-y^i\right)}{m}+\frac{e^T\left(z^{i+1}-z^i\right)}{k}\right)+\lambda\alpha\left(\varepsilon^{-\alpha v^i}\right)^T\left(v^{i+1}-v^i\right)=0 \qquad (5.6)$$

denklemi sağlanıyor ise dur.

Buradaki, uygun α parametresi Bradley ve Mangasarian tarafından 5 olarak bulunmuştur [2], λ parametresi ise $(0,1)$ aralığından çapraz doğrulama yöntemi kullanılarak genelleştirme başarısını enbüyükleyecek değer olarak seçilir.

Algoritma öncelikle herhangi bir rastgele belirlenmiş *uygun* çözüm ile başlar. Daha sonra (5.5) lineer problemi çözdürülerek bir çözüm daha elde edilir ve bu iki çözüm kullanılarak (5.6) eşitliği kontrol edilir. Eğer bu eşitlik sağlanırsa eldeki $(w^{i+1}, \gamma^{i+1}, y^{i+1}, z^{i+1}, v^{i+1})$ çözümü optimaldir, eşitlik sağlanmazsa $\left(w^i, \gamma^i, y^i, z^i, v^i\right) = \left(w^{i+1}, \gamma^{i+1}, y^{i+1}, z^{i+1}, v^{i+1}\right)$ ataması yapılarak (5.5) problemi eşitlik sağlanana dek çözdürülür.

5.2 Önerilen Özellik Seçimi Yöntemi

Bu bölümde, tez kapsamında önerilen gömülü özellik seçimi yöntemi ve onun farklı bir versiyonu sunulmuştur. Önerdiğimiz özellik seçimi yöntemi çok yüzlü konik fonksiyonlar algoritmasını temel almaktadır, aynı zamanda özellik seçimi terimi için de Bradley ve Mangasarian tarafından geliştirilmiş özellik seçimi yöntemindeki fonksiyonu ve çözüm yaklaşımını kullanan iteratif bir algoritmadır. İfade edilen fonksiyon ve çözüm yaklaşımı bir önceki bölümde detaylı bir şekilde ele alınmıştır. Çok yüzlü konik fonksiyonlar algoritması ile birlikte özellik seçimi terimi olarak bu fonksiyonun kullanılması, her iki model de lineer programlama problemi olduğundan uygun görülmüştür. Önerilen özellik seçimi yönteminin kısaca çalışma prensibi şu şekildedir: Her iterasyonda iki farklı alt problemin çözümüyle elde edilen parametreler kullanılarak birçok yüzlü konik fonksiyon oluşturulur, daha sonra ÇKF algoritması ile aynı şekilde her iki veri kümesi de güncellenir ve son olarak optimal ayırıcı yüzey yine ÇKF algoritmasında olduğu gibi belirlenir.

A ve B kümeleri $\mathbb{R}^n$'de verilmiş iki farklı küme olsun:

$$A = \{a^i \in \mathbb{R}^n : i \in I\}, \qquad I = \{1, \dots, m\}$$
$$B = \{b^j \in \mathbb{R}^n : j \in J\}, \qquad J = \{1, \dots, p\}.$$

Bu durumda önerilen özellik seçimi algoritması izleyen şekilde ifade edilir.

Başlangıç Adımı: $l = 1, I_l = I, A_l = A, J_l = J, B_l = B$ ve e uygun boyutlu birler vektörü olsun. Adım 1'e git.

Adım 1: $k = 1$, A_l kümesinden herhangi bir c elemanını seç ve aşağıdaki P_l altproblemini çöz:

$$w^T(a^i - c) + \xi \left\|a^i - c\right\|_1 - \gamma + 1 \leq y_i, \quad \forall_i \in I_l, \tag{5.7}$$

$$-w^T(b^j - c) - \xi \left\|b^j - c\right\|_1 + \gamma + 1 \leq z_j, \quad \forall_j \in J_l, \tag{5.8}$$

$$y = (y_1, \dots, y_{m)} \in R_+^m, z = (z_1, \dots, z_p) \in R_+^p, w = (w_1, \dots, w_n) \in R^n,$$

$$\xi \in [0, \infty), \gamma \in [1, \infty),$$

kısıtları altında;

$$(P_l) \quad \underset{w,\xi,\gamma,y,z}{enk} \frac{e^T y}{m} + \frac{e^T z}{p} \tag{5.9}$$

P_l probleminin bir çözümü (w, ξ, γ, y, z) olsun.

$$(w^{lk}, \xi^{lk}, \gamma^{lk}, y^{lk}, z^{lk}, a^l) = (w, \xi, \gamma, y, z, c)$$

atamasını yap ve $v^{lk} = \left(v_1^{lk}, \dots, v_n^{lk}\right) \in R^n$ değerini rastgele belirle.

Adım 2: Aşağıda verilen P_{lk} altproblemi çöz:

$$w^T(a^i - a^l) + \xi \left\|a^i - a^l\right\|_1 - \gamma + 1 \leq y_i, \quad \forall_i \in I_l, \tag{5.10}$$

$$-w^T(b^j - a^l) - \xi \left\|b^j - a^l\right\|_1 + \gamma + 1 \leq z_j, \quad \forall_j \in J_l, \tag{5.11}$$

$$-v_t \leq w_t \leq v_t, t = 1, \dots, n,$$

$$y = (y_1, \dots, y_{m)} \in R_+^m, z = (z_1, \dots, z_p) \in R_+^p,$$

$$w = (w_1, \dots, w_n), v = (v_1, \dots, v_n) \in R^n$$

$$\xi \in [0, \infty), \gamma \in [1, \infty), \lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}_+$$

kısıtları altında;

$$(P_{lk}) \quad \underset{w,\xi,\gamma,y,z,v}{enk} \lambda_1 \frac{e^T y}{m} + \lambda_2 \frac{e^T z}{p} + \lambda_3 \alpha \left(\varepsilon^{-\alpha v^{lk}}\right)^T (v - v^{lk}). \tag{5.12}$$

Bu problemin optimal bir çözümü $(w^{l,k+1}, \xi^{l,k+1}, \gamma^{l,k+1}, y^{l,k+1}, z^{l,k+1}, v^{l,k+1})$ olsun. Eğer

$$\lambda_1 \frac{e^T(y^{l,k+1} - y^{lk})}{m} + \lambda_2 \frac{e^T(z^{l,k+1} - z^{lk})}{p} + \lambda_3 \alpha \left(\varepsilon^{-\alpha v^{lk}}\right)^T (v^{l,k+1} - v^{lk}) = 0 \tag{5.13}$$

denklemi sağlanırsa aşağıda verilen konik fonksiyonu oluştur:

$$g_l(x) = w^{l,k+1}(x - a^l) + \xi^{l,k+1} \|x - a^l\|_1 - \gamma^{l,k+1} \tag{5.14}$$

ve Adım 3'e git, diğer durumda (yani eşitlik sağlanmazsa) $k = k + 1$ atamasını yap ve Adım 2'ye dön.

Adım 3: A_l ve B_l kümelerini aşağıdaki gibi güncelle:

$$I_{l+1} = \{i \in I_l : g_l(a^i) + 1 > 0\}, A_{l+1} = \{a^i \in A_l : i \in I_{l+1}\},$$
$$J_{l+1} = \{j \in J_l : g_l(b^j) + 1 > 0\}, B_{l+1} = \{b^j \in B_l : j \in J_{l+1}\}.$$

$l = l + 1$ atamasını yap. Eğer $A_l \neq \emptyset$ ise Adım 1'e git, diğer durumda Adım 4'e git.

Adım 4: Ayırıcı fonksiyonu aşağıdaki gibi tanımla ve dur.

$$g(x) = \underset{l}{enk}\ g_l(x) \tag{5.15}$$

Algoritma her l. iterasyonda, A_l kümesinden herhangi bir a^l elamanını seçerek başlar ve (P_l) alt problemini çözerek (w, ξ, γ, y, z) parametrelerini belirler. Bu parametreler rastgele seçilen her bir a^l elemanı için başlangıç çözümü olarak kullanılır. Daha sonra (P_{lk}) alt problemi çözülerek $(w^{l,k+1}, \xi^{l,k+1}, \gamma^{l,k+1}, y^{l,k+1}, z^{l,k+1}, v^{l,k+1})$ parametreleri belirlenir. Eğer, bu iki parametre kümesi ve rastgele belirlenen v^0 ile (5.13) denklemi sağlanırsa g_l çok yüzlü konik fonksiyonu $(w^{l,k+1}, \xi^{l,k+1}, \gamma^{l,k+1}, a^l)$ parametreleri kullanılarak oluşturulur. Belirlenen g_l fonksiyonuna göre ÇKF algoritmasındaki prosedür kullanılarak A_l ve B_l kümeleri güncellenir. Kısıt (5.8) ve (5.11) ile B kümesine ait elemanlar için hataya izin verildiğinden B kümesi de güncellenmektedir. Eğer, sırasıyla (P_l) ve (P_{lk}) alt problemlerinin çözümü ile elde edilen parametreler (5.13) denklemini sağlamazsa, (P_{lk}) alt problemi v^{lk} kullanılarak tekrar çözdürülür. Bu prosedüre, A_l kümesinden rastgele seçilen her bir a^l elemanı için elde edilen çözümlerle (5.13) denklemi sağlanana kadar devam edilir ve algoritma A_l kümesinde hiç bir eleman kalmadığında sonlanır. Optimal ayırıcı fonksiyon oluşturulan çok yüzlü konik fonksiyonların noktasal en küçüğü alınarak bulunur.

Önerilen algoritmada, (P_k) alt probleminin amaç fonksiyonunda ve (5.13) denkleminde λ_1, λ_2 ve λ_3 parametreleri sınıflandırma ve özellik seçimi arasındaki dengeyi sağlamak için kullanılmıştır. Herhangi bir alt problemde A kümesine ait hata olarak tariflenen bir nokta sonraki iterasyonların birinde seçilen merkez noktasına göre doğru sınıflandırılabilir. Böyle bir durumda A kümesine ait hataları cezalandırmak anlamlı olmaz. O halde $\lambda_1 = 1$ ve $\lambda_2, \lambda_3 \epsilon \mathbb{R}_+$ olarak seçilmelidir. Ancak, bu durum birden fazla iterasyon ile çözülen problemler için geçerlidir ve her veri kümesi için birden fazla (P_k) alt problemi çözülmeyebilir, diğer bir deyişle önerilen algoritma tek bir iterasyonda sonlanabilir. Bu durumda, her iki kümeden de hatalı olarak sınıflandırılmış noktaların doğru sınıflandırılabileceği bir sonraki iterasyon olmadığından A ve B veri kümelerinden

oluşabilecek hatalara eşit ağırlık verilmesi anlamlı olur. Her iki veri kümesine ait hatalı noktaların eşit olarak ağırlıklandırılması durumunda amaç fonksiyonu ve (5.13) denklemi aşağıdaki gibi olur;

$$enk\ \ (1-\lambda)\left(\frac{e^T y}{m}+\frac{e^T z}{p}\right)+\lambda\alpha\left(\varepsilon^{-\alpha v^{lk}}\right)^T (v-v^{lk}) \qquad (5.12a)$$

$$(1-\lambda)\left(\frac{e^T(y^{k+1}-y^k)}{m}+\frac{e^T(z^{k+1}-z^k)}{p}\right)+\lambda\alpha\left(\varepsilon^{-\alpha v^{lk}}\right)^T (v^{l,k+1}-v^{lk})=0 (5.13a)$$

$$\lambda\,\epsilon\,(0,1).$$

Parametrelerin iki farklı şekilde düzenlenmesiyle oluşan iki farklı amaç fonksiyonu ve durdurma kriterine ait denklem kullanılarak elde edilen sonuçlar bir sonraki bölümde detaylı bir şekilde sunulmuş ve değerlendirilmiştir.

Bir önceki bölümde detaylı bir şekilde bahsedilen nedenlerden dolayı önerilen özellik seçimi algoritmasına modifiye ÇKF algoritmasının merkez seçimi yaklaşımı aşağıdaki gibi tariflenen ön adım yardımıyla uygulanmıştır:

Merkez Seçme Adımı: Aşağıda verilen (P_l) alt problemini her $l \in I = \{1, \dots, m\}$ ve her $a_l \in A$ elemanı için çöz.

$$\begin{aligned} w^T\left(a^i-a^l\right)+\xi\left\|a^i-a^l\right\|_1-\gamma+1 \le y_i, \quad & \forall_i \in I, \\ -w^T\left(b^j-a^l\right)-\xi\left\|b^j-a^l\right\|_1+\gamma+1 \le z_j, \quad & \forall_j \in J, \\ y \in R_+^m, z \in R_+^p, w \in R^n, \xi\ \in [0,\infty), \gamma \in [1,\infty) & \end{aligned} \qquad (5.16)$$

kısıtları altında;

$$(P_l) \quad \underset{w,\xi,\gamma,y,z}{enk} \ \frac{e^T y}{m}+\frac{e^T z}{p}. \qquad (5.17)$$

P_l probleminin $a_l \in A_l$ elemanına karşılık gelen çözümü $(w^l, \xi^l, \gamma^l, y^l, z^l)$ olsun ve N_l de aşağıda verilen çok yüzlü konik fonksiyon sayesinde A kümesine ait doğru sınıflandırılmış eleman sayısını göstersin.

$$g_l(x) = g_{(w^l,\xi^l,\gamma^l,a^l)}(x) \qquad (5.18)$$

A kümesine ait her a^l elemanını, o elemana karşılık gelen N_l sayısına göre büyükten küçüğe doğru sırala ve önerilen algoritma içinde merkez seçimini bu sıralamaya göre yap.

Merkez seçme adımı tamamlandıktan sonra önerilen özellik seçimi algoritması merkezlerin bulunan sıraya göre seçilmesi dışında bir fark olmadan devam eder.

6. HESAPSAL SONUÇLAR

Tez kapsamında önerilen özellik seçimi yöntemini gerçek hayat verileri ile test edebilmek için "UCI repository of ML" [15] veri tabanından erişilebilen orta ölçekli ve görece daha büyük ölçekli veri kümeleri seçilmiştir. Seçilen orta ölçekli veri kümeleri önerilen özellik seçimi algoritmasını (ÖS-ÇKF) ve merkez seçme adımı eklenmiş diğer versiyonunu (ÖS-MÇKF), gürbüz doğrusal programlama (GDP) ve konkav özellik seçimi (KÖS) yöntemi ile karşılaştırmak için kullanılmıştır. Görece daha büyük ölçekli olan veri kümeleri ise önerilen özellik seçimi yöntemini literatürde iyi bilinen diğer sınıflandırma algoritmaları ile karşılaştırmak için kullanılmıştır. Kullanılan orta ölçekli veri kümeleri ile ilgili sayısal bilgiler Tablo 6.1.'de, görece büyük ölçekli veri kümeleri ile ilgili sayısal bilgiler ise tablo 6.2.'de ve tüm veri kümeleri ile ilgili detaylı bilgiler aşağıda sunulmuştur. Herhangi bir algoritma, orta ölçekli bir veri kümesi ile test edilirken katmanlı 10-kere çapraz doğrulama yöntemi kullanılırken görece büyük ölçekli veri kümeleri tablo 6.2.'deki değerler kullanılarak eğitim ve test kümesi olarak bölünmüştür.

Tablo 6.1. *Orta ölçekli veri kümelerine ait sayısal bilgiler*

Veri kümeleri	Özellik sayısı	Sınıf sayısı	Örnek Sayısı
1. BUPA	6	2	345
2. WBCD	9	2	683
3. WBCP	32	2	194
4. Ionosphere	34	2	351
5. Cleveland	13	2	297
6. Pima	8	2	768

BUPA liver: Bu veri kümesi 345 erkek hastaya ait belirli kan testlerinin sonuçlarını ve günlük tüketilen alkollü içecek sayısını göstermektedir. Bu veriler ile kişinin karaciğer bozukluğu olup olmadığı tahminlenir.

WBCD (Wisconsin göğüs kanseri tanısı veri kümesi): Bu veri kümesi Wisconsin Üniversite Hastanesi'ndeki katı meme kitleli hastalardan elde edilen çeşitli tarama sonuçlarını içermektedir. Bu sonuçlardan yola çıkarak kitlenin iyi huylu ya da kötü huylu olduğuna karar verilir.

WBCP (Wisconsin göğüs kanseri tedavi veri kümesi): Bu veri kümesi Wisconsin Üniversite Hastanesi'nde tedavi gören hastalardan elde edilen çeşitli verileri

içermektedir. Bu verilere göre hastalığın tekrarlanıp tekrarlanmama durumuna karar verilir.

Ionosphere: Johns Hopkins Üniversitesi tarafından oluşturulan bu veri setinde Goose Bay radar sistemi ile iyonosferden toplanan sinyallere ait veriler bulunmaktadır. İyi ve kötü olarak kategorilenen verilerden iyi olanlar iyonosferdeki bazı yapı tiplerinin kanıtını göstermektedir.

Cleveland heart: Cleveland kliniğindeki hastalara ait verilerden oluşan bu veri kümesindeki örneklerin hasta ya da sağlıklı olarak tahminlenmesi istenir.

Pima diabetes: En az 21 yaşında olan kadın hastalardan elde edilen verilerden oluşan bu veri kümesi ile kişinin diyabet hastalığı olup olmadığı belirlenmeye çalışılır.

Tablo 6.2. *Görece büyük ölçekli veri kümelerine ait sayısal bilgiler*

Veri kümeleri	Eğitim/test	Özellik sayısı	Sınıf sayısı	Örnek Sayısı
1. Spambase (SB)	3682/919	57	2	4601
2. Landsat satellite image (LSI)	4435/2000	36	6	6435
3. DNA	2000/1186	180	3	3186
4. Isolet (ISO)	6238/1559	617	26	7797
5. Optical recognition of handwritten digits (HD)	3823/1797	64	10	5620

Spambase (SB): Bu veri kümesi ile e-postaların istenmeyen (spam) ya da değil olarak sınıflandırılması amaçlanmaktadır. Niteliklerin çoğu herhangi bir e-postada belirli bir kelime veya karakterin sık sık meydana gelip gelmediğini gösterir.

Landsat satellite image (LSI): Bu veri tabanındaki veriler NASA'ya ait Landsat uydusu tarafından elde edilmiş verilerdir. Bu sayısal veriler kullanılarak ortaya çıkacak görüntülere dair bir sınıflandırma yapmak amaçlanmaktadır.

DNA: Bu veri kümesinde amaç bir DNA dizilimindeki eklem bağlantılarının donör olarak adlandırılan ekson/intron sınırları, alıcı olarak adlandırılan nitron/ekson sınırları ya da bunlardan hiçbiri olarak tariflenen üç kategoriden hangisine düştüğünü belirlemektir.

Isolet (ISO): Bu veri kümesinde amaç hangi harfin telaffuz edildiğini belirleyebilecek basit bir sınıflandırma prosedürü oluşturmaktır. Bunun için, 150 farklı spiker tarafından alfabenin her harfi ikişer kez söylenerek beş farklı hoparlör tarafından veriler toplanmıştır.

Optical recognition of handwritten digits (HD): Bu veri kümesinde amaç el yazısı ile yazılmış rakamların tahminlenmesidir. Bunun için, 43 farklı kişiden alınan yazı örneklerinin 30'u eğitim kümesi kalanları da test kümesi olarak tariflenmiştir.

Önerilen algoritmada kullanılan λ_2 ve λ_3 parametreleri $P = \{10^i : i = -4, -3, \dots, 0, \dots, 3, 4\}$ kümesinden ızgara arama tekniği [16] kullanılarak her veri kümesi için en yüksek tahminleme başarısına ulaşan değerler seçilmiştir. Parametre arama aralığına $P' = \{10^i : i = -7, -6, \dots, 0, \dots, 6, 7\}$ kümesi kullanılarak başlanmıştır. Fakat, bu kümenin uç noktalarında herhangi bir veri kümesi için anlamlı sonuçlar gelmediğinden aralık daraltılmıştır. λ parametresi de $P'' = \{0.05, 0.10, 0.20, \dots, 0.95\}$ kümesinden yine ızgara arama tekniği kullanılarak her veri kümesi için en yüksek tahminleme başarısına ulaşan değerler seçilmiştir. Ayrıca, α parametresi de Bradley ve Mangasarian tarafından önerildiği şekilde 5 olarak alınmıştır.

Karşılaştırma yapmak için kullanılan GDP, KÖS algoritmaları ve tez kapsamında önerilen ÖS-ÇKF ve ÖS-MÇKF yöntemlerinde lineer problemlerin çözümü için Gurobi [17] optimizasyon paketi kullanılarak Python programlama dili [18] ile implemente edilmiştir. Bu algoritmalara dair sunulan tüm deneyler 3.5 GHz işlemcili ve 16 GB RAM'e sahip bir bilgisayarda uygulanmıştır. Ayrıca, orta ölçekli verileri test etmek için kullanılan katmanlı 10-kere çapraz doğrulama yöntemini uygulamak için scikit-learn [10] makine öğrenmesi paketinden yararlanılmıştır.

Tablo 6.3. *ÖS-ÇKF algoritmasında farklı parametreler kullanılarak elde edilen eğitim ve test başarı yüzdeleri*

Veri kümeleri	Çok Parametreli		Tek Parametreli	
	Eğt.	**Test**	**Eğt.**	**Test.**
1. BUPA	81,2	68,4±4	63,8	63,5±4
2. WBCD	99,7	97,1±2	91,5	90,4±5
3. WBCP	85,2	80,9±4	86,2	81,0±5
4. Ionosphere	93,3	92,9±6	89,8	87,0±6
5.Cleveland	91,8	80,8±5	80,0	75,0±3
6. Pima	81,4	74,2±4	67,3	67,1±1

Tablo 6.3.'de ÖS-ÇKF algoritmasının alt problemine ait amaç fonksiyonunda ve durdurma kriteri olarak kullanılan denklemde tek parametre ve birden fazla parametre kullanılarak elde edilen test ve eğitim başarı yüzdeleri verilmiştir. Tabloda görüldüğü gibi

tek parametre kullanıldığı durumda (yani sınıflandırmadaki toplam hata ile özellik seçimi dengelendiğinde) eğitim ve test başarıları arasındaki fark oldukça azdır. Bu değerlere bakılarak aşırı uyum sorununun ortadan kaldırıldığı söylenebilir. Diğer taraftan birden fazla parametre kullanıldığı durumda (yani sadece *B* kümesine ait hata cezalandırılarak sınıflandırma ve özellik seçimi dengelendiğinde) ise test başarılarının yaklaşık %5 daha yüksek olduğu görülmektedir. Bu nedenle merkez seçme adımı ve diğer yöntemlerle olan karşılaştırmalar çok parametreli model kullanılarak yapılmıştır.

Tablo 6.4. *Farklı yöntemler kullanılarak orta ölçekli veri kümeleri ile elde edilen eğitim ve test başarı yüzdeleri*

	GDP		**KÖS**		**ÖS-MÇKF**		**ÖS-ÇKF**	
Veri kümeleri	**Eğt.**	**Test**	**Eğt.**	**Test**	**Eğt.**	**Test**	**Eğt.**	**Test**
1. BUPA	68,5	66,4±5,61	66,6	64,6±5,29	81,2	**69,0**±6,73	81,2	68,4±4
2. WBCD	97,5	97,2±1,90	97,2	96,2±2,68	100	**97,5**±4,11	99,7	97,1±2
3. WBCP	89,4	77,5±8,91	69,4	67,9±8,99	89,5	**81,1**±5,25	85,2	80,9±4
4. Ionosphere	94,8	87,3±7,28	86,5	83,5±4,50	93,0	92,6±6,36	93,3	**92,9**±6
5.Cleveland	85,1	82,5±5,95	83,2	80,2±6,36	92,4	80,7±6,11	91,8	80,8±5
6. Pima	76,6	75,5±5,13	76,5	**75,7**±3,56	80,8	73,4±4,96	81,4	74,2±4

Tablo 6.4.'de GDP, KÖS, ÖS-ÇKF ve ÖS-MÇKF yöntemleri ile elde edilen eğitim ve test başarıları sunulmuştur. Kalın rakamlarla ifade edilen değerler her bir satıra ait elde edilen en iyi sonucu göstermektedir. Tablo 6.4.'deki sunulan sonuçlardan görülebileceği üzere çalışma kapsamında önerilen ÖS-ÇKF ve ÖS-MÇKF yöntemleri test edilen altı farklı veri kümesinin dördünde en iyi test başarısına ulaşmıştır. Ayrıca tüm veri kümeleri için diğer yöntemlerle kıyaslanınca kabul edilebilir tahminleme değerleri elde edildiği görülmektedir. Eğitim ve test başarılarına bakıldığı zaman, ÇKF algoritmasından elde edilen değerler ile kıyaslandığında önerilen her iki yöntemin de aşırı uyum sorununu azalttığı dolayısıyla genelleştirme başarısını arttırdığı görülmektedir.

Tablo 6.5.'de önerilen özellik seçimi yöntemini test etmek amacıyla kullanılan orta ölçekli veri kümeleri için Adım 2'de ortaya çıkan tekrar sayısı ve algoritma sonucunda seçilen özellik sayıları sunulmuştur. Tabloda sunulan tüm değerler katmanlı 10-kere çapraz doğrulama sonucunun ortalamasıdır. Tablodaki sonuçlar incelendiğinde, kullanılan tüm test kümeleri için Adım 2'de ortalama olarak 2 ila 5 tekrar sonucunda durdurma kriterinin sağlandığı görülmektedir.

Tablo 6.5. *Orta ölçekli veri kümelerinden elde edilen Adım 2'ye ait tekrar sayıları ve seçilen ortalama özellik sayıları*

Veri kümeleri	Adım 2 için ortalama tekrar sayısı	Seçilen ortalama özellik sayısı
1. BUPA (6)	2,01	5,97
2. WBCD (9)	4,05	0,75
3. WBCP (32)	4,31	13,10
4. Ionosphere (34)	2,00	0,00
5.Cleveland (13)	2,08	12,66
6. Pima (8)	2,81	7,84

Seçilen ortalama özellik sayıları incelendiğinde, boyut sayısı daha az olan veri kümeleri için, örneğin BUPA ve Pima veri kümeleri gibi, seçilen özellik sayılarının daha fazla olduğu söylenebilir. Diğer taraftan, özellik sayısı daha çok olan veri kümelerinde seçilen özellik sayılarının oldukça az olduğu görülmektedir. Ionosphere veri kümesi için özelliklerin tamamının bastırıldığı ve burada optimal ayırıcı fonksiyonu oluşturabilmek için ξ ve γ parametrelerinin kullanıldığı söylenebilir.

Tablo 6.6. *Farklı yöntemler kullanılarak görece büyük ölçekli veri kümeleri ile elde edilen test başarı yüzdeleri*

Veri kümeleri Sınıflandırıcılar	SB	LSI	DNA	ISO	OD
NB (kernel)	76,17	82,10	93,34	84,22	90,32
PART	91,40	85,25	91,06	82,81	89,54
LibSVM (LIN)	90,97	85,05	93,09	96,02	96,55
J48	92,93	85,35	92,50	83,45	85,75
Logistic	92,06	83,75	88,36	-	92,21
ÖS-ÇKF	92,06	90,50	93,17	90,19	92,49

Görece büyük ölçekli veri kümeleri ve farklı sınıflandırma algoritmaları kullanılarak elde edilen test başarıları Tablo 6.6.'da sunulmuştur. Çalışma kapsamında önerilen ÖS-ÇKF yöntemi dışında diğer sınıflandırıcılara ait sonuçlar [12] makalesinden alınmıştır. Burada, geniş bir karşılaştırma yapabilmek için olabildiğince farklı yapıya sahip olan sınıflandırıcılar seçilmeye çalışılmıştır. Seçilen sınıflandırma yöntemleri şu şekilde sıralanabilir: olasılık temelli bir algoritma olan Naif Bayes (NB), kural temelli bir algoritma olan PART, destek vektör makinelerini kullanan bir algoritma Lineer LibSVM,

karar ağaçları temelli bir algoritma olan J48 ve Lojistik Regresyon (Lojstik). ÖS-ÇKF algoritmasında bulunan λ_2 ve λ_3 parametrelerini belirlemek için orta ölçekli verilerde olduğu gibi aynı *P* kümesinden ızgara arama tekniği kullanılarak en yüksek test başarısına ulaşan değerler seçilmiştir. Diğer sınıflandırıcılara ait sonuçların WEKA platformu kullanılarak varsayılan parametreler ile denendiği ilgili çalışmada ifade edilmiştir [12].

Tablo 6.6.'da sunulan sonuçlara göre ÖS-ÇKF yöntemi karşılaştırılan diğer tüm yöntemlerin arasında LSI veri kümesi için en iyi tahminleme başarısını elde etmiştir. Ayrıca, ÖS-ÇKF yönteminin DNA ve Spambase veri kümelerinde elde ettiği test başarısı da diğer yöntemler tarafından elde edilen en iyi değere çok yakındır.

6.1 Merkez Seçme Adımının Etkisi

Bu bölümde, çalışma kapsamında önerilen yönteme uygulanan merkez seçme adımının etkisi yapılan deneylerle elde edilen sonuçlara göre tartışılmıştır. Yapılan bu deneyler daha önceki bölümde anlatıldığı gibi aynı donanıma sahip bir bilgisayar kullanılarak ve katmanlı 10-kat çapraz doğrulama yapılarak elde edilmiştir. Her iki versiyonu detaylı olarak karşılaştırabilmek için yine bir önceki bölümde detayları paylaşılmış olan orta ölçekli veri kümeleri kullanılmıştır. Çalışma kapsamında önerilen özellik seçimi yönteminin ÖS-MÇKF ve ÖS-ÇKF versiyonlarına ait yüzdesel olarak ortalama eğitim başarısı ve test başarısı değerleri Tablo 6.7.'de, eğitim süreleri ve ortalama oluşturulan çok yüzlü konik fonksiyon sayıları Tablo 6.8.'de sunulmuştur.

Tablo 6.7. *ÖS-MKÇF ve ÖS-ÇKF yöntemleri kullanılarak orta ölçekli veri kümeleri ile elde edilen eğitim ve test başarı yüzdeleri*

	ÖS-MÇKF		**ÖS-ÇKF**	
Veri kümeleri	**Eğitim**	**Test**	**Eğitim**	**Test**
1. BUPA	81,19	69,00	81,19	68,40
2. WBCD	100,00	97,52	99,70	97,09
3. WBCP	89,46	81,12	85,23	80,93
4. Ionosphere	93,00	92,58	93,32	92,85
5.Cleveland	92,39	80,74	91,81	80,76
6. Pima	80,84	73,44	81,40	74,20

Modifiye ÇKF algoritmasının çıkış noktası bir önceki bölümde detaylı olarak anlatılmıştır. Modifiye ÇKF algoritmasının orijinal ÇKF algoritmasına göre daha yüksek

tahminleme başarısına ulaştığı ve önemli derecede daha az sayıda alt problem çözdüğü ilgili çalışma kapsamında gösterilmiştir [3].

Tablo 6.8. *ÖS-MKÇF ve ÖS-ÇKF yöntemleri kullanılarak orta ölçekli veri kümelerinde ortaya çıkan eğitim süreleri ve oluşturulan ortalama çkf sayıları*

	ÖS-MÇKF		**ÖS-ÇKF**	
Veri kümeleri	**Eğitim süresi (s)**	**Oluşturulan ortalama ÇKF sayısı**	**Eğitim süresi (s)**	**Oluşturulan ortalama ÇKF sayısı**
1. BUPA	22,41	111	6,00	154
2. WBCD	109,41	40	4,68	39
3. WBCP	32,73	6	3,16	8
4. Ionosphere	109,47	200	34,73	201
5.Cleveland	22,04	34	3,67	61
6. Pima	146,73	191	75,19	338

Bu nedenle, modifiye ÇKF algoritmasından yola çıkılarak ÖS-ÇKF yöntemine uygulanan merkez seçme adımı ile de buna benzer sonuçlar elde edilebileceği öngörülmüştür. Ancak, Tablo 6.7'deki sonuçlar incelenerek her iki versiyona ait test ve genelleştirme başarıları değerlendirilirse, iki versiyon arasında birbirine üstünlük sağlayacak anlamlı bir fark görülmemektedir. Oluşturulan ortalama çok yüzlü konik fonksiyon sayısı açısından değerlendirmek için Tablo 6.8 incelendiğinde, ÖS-ÇKF yöntemine uygulanan merkez seçme adımı ile üç veri kümesi dışında (BUPA, Pima ve Cleveland) anlamlı bir gelişim gözlemlenmemiştir. Tablo 6.7 ve Tablo 6.8'deki sonuçlar birlikte değerlendirildiğinde ÇKF algoritması ile özellik seçiminin uygulanması yani ÖS-ÇKF yöntemi sayesinde, modifiye uygulaması ile öngörülen sonuçlara zaten ulaşıldığından bunun üzerinde anlamlı bir iyileşme oluşmamaktadır. Buna ek olarak, Tablo 6.8'deki eğitim süreleri incelendiğinde uygulanan merkez seçimi adımının herhangi bir performans kriterinde iyileşme garanti etmeden önemli ölçüde daha fazla eğitim süresi gerektirdiği görülmektedir. Verilen tüm bu sonuçlardan ve değerlendirmelerden yola çıkılarak ÖS-ÇKF yöntemi ile merkez seçme adımının uygulanması gerekli görülmemektedir.

7. ÖNERİLEN ÖZELLİK SEÇİMİ YÖNTEMİ İLE ÖĞRENCİ BAŞARI TAHMİNLEMESİ

Geleneksel öğretim ortamlarında, eğitimciler ile öğrenciler arasında kurulabilen yüz yüze etkileşimler sayesinde eğitimcilerin öğrencilere öğrenme deneyimleri hakkında geri bildirimlerde bulunmaları mümkündür [19]. Fakat, eğitim sistemlerinin gelişmesiyle internet tabanlı uzaktan öğretim programlarının yaygınlaşması eğitimciler ile öğrenciler arasındaki etkileşimi kısıtlamaktadır. Aynı zamanda herhangi bir örgün öğretim programında da eğitimci başına düşen öğrenci sayısının fazla olması gibi durumlarda eğitimciler ile öğrenciler arasındaki etkileşim sınırlanabilir. Oysa, öğrencilerin eğitim hayatlarına sağlıklı bir şekilde devam edebilmeleri, kendi yetenekleri doğrultusunda ilerleyebilmeleri için eğitimcilerin öğrencileri hakkında yeterli bilgi birikimine sahip olmaları gerekmektedir. Bunun için eğitimcilerin de diğer tüm sistemlerdeki karar vericiler gibi halihazırda var olan verilerden kolaylıkla görünmeyen düzenleri ortaya çıkarabilmeleri, öğrencilere dair ileri dönük doğru tahminlerde bulunup geri bildirimler sunabilmeleri için yardımcı bir sisteme ihtiyaç duyulabilir. Bütün bu nedenlerden dolayı eğitim alanında da veri madenciliği uygulamalarının kullanımı yaygınlaşmaktadır.

Eğitim alanındaki en yaygın veri madenciliği problemlerinden bir tanesi öğrenci başarı tahminlemesi olarak görülebilir. Bu problemde, öğrencilerin birtakım demografik ve akademik bilgileri kullanılarak bir derse ait ya da genel başarı durumlarının tahmin edilmesi amaçlanır. Bununla birlikte, tahminleme yapılırken öğrencilere ait kullanılan bilgilerden hangi özelliklerin öğrenci başarısı üzerinde daha etkili olduğu, hangilerinin etkisiz olduğu ya da daha az etkili olduğu gibi bilgiler de ortaya çıkarılmaya çalışılır. Böylece, karar vericilerin tahminleme sonucunda ortaya çıkan düzenlere göre öğrencilere veya velilere uygun geri dönüşler sağlaması amaçlanmaktadır.

Öğrenci başarı tahminlemesi probleminde, öncelikli olarak öğrencilere ait veriler toplanır. Uzaktan öğretim sistemlerinde çoğunlukla öğrencilere ait çok çeşitli bilgilerin saklandığı bir veri tabanı bulunmaktadır, örgün öğretim sistemlerinde ise bu bilgilere ulaşabilmek için okul kayıtlarından veya öğrenci, eğitimci ve veliler ile yapılan anketler gibi çeşitli veri toplama yöntemlerinden faydalanılmaktadır [20]. Bir sonraki adım toplanan verilerin, uygulanacak olan tahminleme algoritmasına uygun hale getirilmesidir. Bu aşamada, veriler üzerinde temizleme, dönüştürme ve sınıf dağılımlarına göre dengesiz olan veri kümesini dengeli hale getirme gibi işlemler yapılabilir [20]. Bir sonraki aşama tahminleme algoritmasının hazır durumdaki veri kümesi üzerine uygulandığı adımdır.

Son aşamada ise tahminleme algoritmasından elde edilen model, öğrencilerin başarısını tespit edebilmek için analiz edilir.

Yapılan tez çalışması kapsamında önerilen özellik seçimi yönteminin bir gerçek hayat problemi üzerine kurgulanmasını ifade edebilmek için öğrenci başarı tahminlemesi problemi ele alınmıştır. Öğrenci başarı tahminlemesi probleminin ÖS-ÇKF algoritmasıyla uygulanabilmesi için "UCI repository of ML" [15] veri tabanından erişimi olan bir küme seçilmiştir. Bu veri kümesi, Portekiz'in Alentejo bölgesindeki ortaokul seviyesinde eğitim veren iki farklı devlet okulundan toplanan 2005-2006 öğretim yılına ait matematik dersindeki başarıyı tahminlemeye yönelik verilerden oluşmaktadır [21]. Verilerin kaynağını yapılan anketlerle ulaşılan bilgiler ve okulların tuttuğu raporlar oluşturmaktadır. Ele alınan veri kümesi 395 örnek ve 32 özellikten oluşmaktadır, detaylı bilgiler Tablo 7.1.'de sunulmuştur.

Tablo 7.1. *Öğrenci başarı tahminlemesi veri kümesine ait özellikler* [21]

Özellikler	**Açıklaması**
okul	öğrencinin okul bilgisi (okul-1 ya da okul-2)
cinsiyet	öğrencinin cinsiyeti (kadın ya da erkek)
yaş	öğrencinin yaşı (15-22)
adres	öğrencinin ev adres tipi (kent ya da kırsal)
aileBoyutu	ailesinin büyüklüğü (≤3 ya da >3)
EbvDurumu	ebeveynlerin birlikte yaşama durumu (birlikte ya da ayrı)
Aeğt	öğrencinin annesinin eğitim durumu (0- eğitimsiz, 1- ilkokul, 2- ortaokul, 3- lise, 4- üniversite)
Beğt	öğrencinin babasının eğitim durumu (0- eğitimsiz, 1- ilkokul, 2- ortaokul, 3- lise, 4- üniversite)
Aiş	öğrencinin annesinin mesleği (öğretmen, sağlıkçı, memur, çalışmıyor ya da diğer)
Biş	öğrencinin babasının mesleği (öğretmen, sağlıkçı, memur, çalışmıyor ya da diğer)
neden	bu okulun seçilme nedeni (eve yakın olması, okulun başarısı, ders tercihi ya da diğer)
veli	öğrencinin velisi (anne, baba ya da diğer)
seySüresi	evden okula seyahat süresi (1- <15 dk, 2- 15'den 30 dk'ya kadar, 3- 30'dan 1 saate kadar ya da 4- > 1 saat)
çalSüresi	haftalık ders çalışma süresi (1- <2 saat, 2- 2 ile 5 arası, 3- 5 ile 10 arası ya da 4- >10 saat)
ders	önceki yıllarda başarısız olduğu ders sayısı (1-4)
okulDes	okul dışında eğitim desteği (var ya da yok)
aileDes	aile tarafından eğitim desteği (var ya da yok)
ödemeliDers	ders içinde önerilen ödemeli ekstra dersler (var ya da yok)
aktivite	ders dışı aktivite (var ya da yok)

Tablo 7.1. (Devam) *Öğrenci başarı tahminlemesi veri kümesine ait özellikler* [21]

anaokulu	anaokuluna gidip gitmeme durumu (evet ya da hayır)
devam	bir sonraki öğretim aşamasına devam etmeyi isteyip istememe (evet ya da hayır)
internet	evde internet erişimi (var ya da yok)
ilişki	romantik ilişki durumu (var ya da yok)
aileİlişkisi	aile ilişkilerinin kalitesi (1-5: 1-çok kötü, 5-çok iyi)
boşZaman	okuldan sonra boş zaman süresi (1-5: 1- çok az, 5- çok fazla)
arkadaş	arkadaşları ile dışarı çıkma sıklığı (1-5: 1- çok az, 5- çok fazla)
gAlkol	günlük alkol tüketimi (1-5: 1- çok az, 5- çok fazla)
hAlkol	haftalık alkol tüketimi (1-5: 1- çok az, 5- çok fazla)
sağlık	mevcut sağlık durumu
devamsızlık	devamsızlık yaptığı gün sayısı (0-93)
N1	ilk dönem notu
N2	ikinci dönem notu

Tablo 7.1. incelendiğinde, veri kümesinde hem kategorik hem de sayısal değerli özelliklerin bulunduğu görülmektedir. Bu veri kümesini kullanarak ÖS-ÇKF algoritmasıyla tahminleme yapılabilmesi için özelliklerin tamamının sayısal değerlere dönüştürülmesi gerekmektedir. Verilerin dönüştürme işlemi kategorik olan her bir özellik için izleyen şekilde yapılmıştır; kategorik bir özelliğe karşılık gelen her bir değer için sıfırlardan oluşan yeni bir sütun oluşturulmuş ve verinin sahip olduğu değere 1 atanmıştır, örneğin cinsiyet özelliği için kadın ve erkek değerlerine karşılık gelen ve sıfırlardan oluşan iki yeni sütun tariflenip kadın olan bir veri için ‘kadın’a karşılık gelen sütuna 1 ataması yapılmıştır. Sayısal değerli özelliklere herhangi bir işlem yapılmadan olduğu gibi kullanılmıştır. Tarif edilen veri dönüştürme işleminin sonucunda kümenin boyutu 50’ye ulaşmıştır. Etiketler ise izleyen şekilde belirlenmiştir; öğrencilerin 0 ile 20 arasında değişen final notlarından 10 ve 10’un üzerinde olanlar başarılı diğerleri ise başarısız sayılmıştır [21]. Bu karara göre toplam 395 örneğin 265’i başarılı 130’u ise başarısız olarak etiketlenmiştir.

Öğrenci başarı kümesine yapılan veri dönüştürme işleminin ardından ÖS-ÇKF algoritmasının uygulanması ile elde edilen test ve eğitim başarı yüzdeleri ile eğitim süreleri ve ortalama seçilen özellik sayıları Tablo 7.2.’de sunulmuştur. Tablo 7.2.’deki eğitim ve test başarı sonuçları incelendiğinde ÖS-ÇKF algoritmasının hem tek

parametreli hem de çok parametreli durumda birbirine yakın sonuçlar elde ettiği gözlemlenmiştir.

Tablo 7.2. *ÖS-ÇKF algoritması ile öğrenci veri kümesinden elde edilen sonuçlar*

Tek Parametreli				**Çok Parametreli**			
Eğt.	**Test**	**Eğt. sür. (sn)**	**Seç. öz.**	**Eğt.**	**Test**	**Eğt. sür. (sn)**	**Seç. öz.**
95,0	90,6±5	6,2	6	98,8	90,1±1	4,9	7

Her iki durumda da hem seçilen özellik sayıları hem test başarıları hem de eğitim süreleri birbirine yakındır. Sadece eğitim başarıları arasında küçük bir farkın oluştuğu ve tek parametreli durumda genelleştirme başarısının biraz daha iyi olduğu görülmektedir.

Tablo 7.3. *ÖS-ÇKF algoritması ile öğrenci başarı tahminlemesi sonucu seçilen özellikler*

Seçilen Özellikler	
Tek Parametreli	**Çok Parametreli**
N2, Biş, aileİlişkisi, çalSüresi, veli, neden	N2, Beğt, çalSüresi, devamsızlık, yaş, okul, boşZaman, arkadaş

ÖS-ÇKF algoritması ile öğrenci başarı tahminlemesi sonucunda her iki durumda da seçilen özelliklerin detayları Tablo 7.3.'de paylaşılmıştır. Burada, veri kümesine ÖS-ÇKF algoritması uygulanmadan önce bir veri dönüştürme işlemi yapıldığı için seçilen özelliklerin belirlenmesi de bu dönüştürme işlemine göre yapılmıştır. Dönüştürme işlemi sonucunda veri kümesinde kategorik olan her bir özelliğe karşılık gelen değere yeni bir sütun açıldığından bu sütunlardan herhangi birisinin seçilmesi ile o özelliğin seçildiği anlaşılmıştır. Örneğin, öğrencinin babasının mesleği özelliğinin seçilmesi bu özelliğe karşılık gelen değerler "öğretmen, sağlıkçı, memur, çalışmıyor ya da diğer" için açılan sütunlardan herhangi birinin seçilmesi ile belirlenmiştir.

Tablo 7.3.'e göre seçilen özellikler incelendiğinde, ilk olarak beklendiği gibi not bilgisinin tahminleme için önemli olduğu görülmektedir. Bununla birlikte, baba ile ilgili kullanılan özelliklerin her ikisi birden hem babanın mesleği hem de eğitim durumu öğrenci başarısı üzerinde etkili olduğu ortaya çıkmıştır. Aile ilişkilerinin ne kadar kaliteli olduğunun da seçilmiş olması öğrenci başarısı üzerinde aile faktörünün önemli bir etken olduğunu ortaya koymaktadır. Çalışma süresi yani öğrencinin bu derse ayırdığı zaman ve devamsızlık yapılan gün sayılarının seçilmiş olması beklendiği gibi ders çalışmaya ve

okula ayrılan sürenin öğrenci başarısı üzerinde etkili olduğunu ifade eder. Veli değişkeninin seçilmesi öğrencinin akademik durumuyla ne kadar ilgilenildiğinin ya da ilgilenen kişinin öğrenciyi motive edici bir faktör olarak önemli olduğu yorumu yapılabilir. Öğrencinin hangi okula devam ettiği, o okulu hangi nedenlerle seçtiği ve okuldan sonraki boş zaman süresi bir bütün olarak yorumlanırsa okulla ilgili niteliklerin ve okuldan beklentilerin de öğrenci başarısı üzerinde etkili olduğu söylenebilir. Yaş ve arkadaşlar ile dışarı çıkma sıklığı da birlikte yorumlanırsa beklenildiği gibi öğrencilerin yaş grubu ve arkadaşları ile okul dışı aktivitelerinin sıklığı başarı üzerinde etkili olmaktadır.

8. DEĞERLENDİRME VE ÖNERİLER

Yapılan çalışma kapsamında, ÖS-ÇKF adı verilen çok yüzlü konik fonksiyonlar temelli bir gömülü özellik seçimi yöntemi ve bu yöntemin bir varyasyonu olan ÖS-MÇKF geliştirilmiştir. Önerilen yöntemde, çok yüzlü konik fonksiyonlar ile özellik seçimini gerçekleştirebilmek için konkav özellik seçimi yaklaşımından yararlanılmıştır. Yöntem, çok yüzlü konik fonksiyonlar algoritmasının genel yapısı korunarak özellik seçimi kısmı için konkav özellik seçimi yaklaşımındaki terim kullanılarak oluşturulmuştur. Böylece, her adımda rastgele belirlenen bir merkez noktasına göre hem A ve B kümesine ait noktaları olabildiğince ayıracak hem de uygun özellikleri seçecek şekilde bir çok yüzlü konik fonksiyon oluşturan iteratif bir yöntem elde edilmiştir.

Önerilen özellik seçimi yöntemi literatürde bu alanda sıkça kullanılan hem orta ölçekli hem de görece büyük ölçekli veri kümeleri üzerinde denenmiş ve iyi bilinen diğer yöntemlerle kıyaslanarak oldukça başarılı sonuçlar elde ettiği gözlemlenmiştir. Ayrıca, çok yüzlü konik fonksiyonlar algoritmasının zayıf yönleri, merkez noktasına olan hassasiyet ve aşırı uyum sorununa yatkınlık üzerine tartışılmıştır. Buradan yola çıkılarak daha önce bu sorunları ortadan kaldırmak için geliştirilmiş olan modifiye ÇKF algoritması önerilen yönteme basit bir ön adım ile uyarlanarak ÖS-MKÇF yöntemi oluşturulmuştur. ÖS-ÇKF ve ÖS-MÇKF yöntemleri arasında, yapılan deneylerden elde edilen sonuçlarla detaylı bir kıyaslama sunulmuştur. Buna göre, hem ÖS-ÇKF hem de ÖS-MÇKF yöntemi sayesinde ÇKF algoritmasında ortaya çıkan aşırı uyum sorunu azaltılmıştır. Ayrıca ÖS-ÇKF algoritmasının görece büyük ölçekli veri kümeleri üzerinde de oldukça başarılı sonuçlar verdiği gözlemlenmiştir. ÖS-MÇKF algoritması orta ölçekli veri kümeleri için bile çok yüksek eğitim süresi gerektirdiğinden görece büyük ölçekli veri kümeleri üzerinde denenmemiştir. Son olarak, ÖS-ÇKF algoritması ile öğrenci başarı tahminlemesi üzerine bir uygulama yapılarak elde edilen sonuçlar detaylı bir şekilde yorumlanmış ve bu alanda da başarılı sonuçların ortaya çıktığı görülmüştür.

Önerilen ÖS-ÇKF ve ÖS-MKÇF yöntemleri ile beklenildiği gibi genelleştirme başarısının iyileştiği elde edilen test sonuçları ile görülmüştür. Yöntemde kullanılan ağırlıklandırma parametreleri için daha verimli bir parametre seçme yöntemi ve özellik seçimi için kullanılan terim üzerine daha detaylı çalışmalar yapılması planlanmaktadır.

KAYNAKÇA

[1] Tang, J., Alelyani, S., and Liu, H. (2014). Feature selection for classification: A review. *Data classification: Algorithms and applications*, 37.

[2] Bradley, P. S. and Mangasarian, O. L. (1998). Feature Selection via Concave Minimization and Support Vector Machines. *Proceedings of the Fifteenth International Conference on Machine Learning*, 1 (98), 82-90.

[3] Gasimov, R. and Öztürk, G. (2006). Separation via Polyhedral Conic Functions. *Optimization Methods and Software*, 21(4).

[4] Öztürk, G. (2007). *Sınıflandırma Problemleri İçin Yeni Bir Matematiksel Programlama Yaklaşımı*. Doktora Tezi. Eskişehir: Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü.

[5] Beniwal, S. and Arora, J. (2012). Classification and feature selection techniques in data mining. *International journal of engineering research and technology*, 1(6).

[6] Elmas, Ç. (2003). *Yapay Sinir Ağları*. Ankara: Seçkin Yayıncılık.

[7] Kartal, E. (2015). *Sınıflandırmaya dayalı makine öğrenmesi teknikleri ve kardiyolojik risk değerlendirmesine ilişkin bir uygulama.* Doktora Tezi. İstanbul: İstanbul Üniversitesi, Fen Bilimleri Enstitüsü.

[8] Ceylan, G. (2017). *Çok yüzlü konik sınıflandırıcılar için marj en büyüklenmesi.* Yüksek Lisans Tezi. Eskişehir: Anadolu Üniversitesi, Fen Bilimleri Enstitüsü.

[9] Kohavi, R., ve John, G. H. (1997). Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2), 273-324.

[10] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. and Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct), 2825-2830.

[11] Bennet, K. P. and Mangasarian, O. L. (1992). Robust linear programming discrimination of two linearly inseparable sets. *Optimization methods and software*, 1(1), 23-34.

[12] Ozturk, G., Bagirov, A. M. and Kasimbeyli, R. (2015). An incremental piecewise linear classifier based on polyhedral conic separation. *Machine Learning*, 101(1-3), 397-413.

[13] Mangasarian, O. L. (1996). Machine learning via polyhedral concave minimization. *In Applied Mathematics and Parallel Computing* (pp. 175-188). Physica-Verlag HD.

[14] Bradley, P. S. (1998). *Mathematical programming approaches to machine learning and data mining.* Doktora Tezi. Madison: University of Wisconsin, Computer Sciences.

[15] Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[16] Hsu, C. W., Chang, C. C. and Lin, C. J. (2003). *A practical guide to support vector classification.* Department of Computer Science, National Taiwan University.

[17] Gurobi Optimization Inc. (2010). *Gurobi Optimizer Reference Manual Version 3.0*. Houston, Texas: Gurobi Optimization.

[18] Rossum, G. (1995). *Python tutorial, Technical Report CS-R9526*. Amsterdam: Centrum voor Wiskunde en Informatica (CWI).

[19] Romero, C., and Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005. *Expert systems with applications*, 33(1), 135-146.

[20] Márquez-Vera, C., Morales, C. R., and Soto, S. V. (2013). Predicting school failure and dropout by using data mining techniques. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, 8(1), 7-14.

[21] Cortez, P., and Silva, A. (2008). Using data mining to predict secondary school student performance. In A. Brito & J. Teixeira (eds.), *EUROSIS*, (pp.5-12).

ÖZGEÇMİŞ

Adı Soyadı : Öznur AY

Yabancı Dil : İngilizce

Doğum Yeri ve Yılı : Finike / 28.04.1992

E-Posta : oznuray26@gmail.com

Eğitim ve Mesleki Geçmişi:

- 2015-2018, Anadolu Üniversitesi, Mühendislik Fakültesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı (Yüksek Lisans Programı)
- 2010-2014, Eskişehir Osmangazi Üniversitesi, Fen-Edebiyat Fakültesi, Matematik-Bilgisayar Bölümü (Lisans Programı)

Yayınları:

- Ay, Ö. ve Kasımbeyli, R. (2018). Çok yüzlü konik fonksiyonlar temelli özellik seçimi algoritması. *38. Yöneylem Araştırması Endüstri Mühendisliği Ulusal Kongresi (YAEM2018 Kongresi)'*nde sunulan bildiri. Eskişehir, Anadolu Üniversitesi.