

**T.C.
İNÖNÜ ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ**



**DESTEK VEKTÖR MAKİNELERİ VE
LOJİSTİK REGRESYON YÖNTEMİ İLE
SINIFLANDIRMA: TÜRKİYE' NİN İTHALAT
VE İHRACAT ÖRNEĞİ**

YÜKSEK L İSANS TEZİ

**Danışman
Doç. Dr. Bahadır YÜZBAŞI**

**Hazırlayan
Özlem TAYA**

MALATYA 2022

T.C.
İNÖNÜ ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ

**DESTEK VEKTÖR MAKİNELERİ VE LOJİSTİK REGRESYON YÖNTEMİ
İLE SINIFLANDIRMA: TÜRKİYE' NİN İTHALAT VE İHRACAT ÖRNEĞİ**

YÜKSEK LİSANS TEZİ

Hazırlayan
Özlem TAYA

Danışman
Doç. Dr. Bahadır YÜZBAŞI

MALATYA – 2022

ONUR SÖZÜ

Doç. Dr. Bahadır YÜZBAŞI' nın danışmanlığında yüksek lisans tezi olarak hazırladığım **“DESTEK VEKTÖR MAKİNELERİ VE LOJİSTİK REGRESYON YÖNTEMİ İLE SINIFLANDIRMA: TÜRKİYE’ NİN İTHALAT VE İHRACAT ÖRNEĞİ”** başlıklı bu çalışmanın, bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurmaksızın tarafımdan yazıldığını ve yararlandığım bütün yapıtların hem metin içinde hem de kaynakçada yöntemine uygun biçimde gösterilenlerden oluştuğunu belirtir, bunu onurumla doğrularım.

Tarih:

Ad-Soyad: Özlem TAYA

İmza:

ÖNSÖZ

Tez çalışmam süresince bilgi ve değerli fikirlerinden istifade ettiğim, hiçbir zaman desteğini esirgemeyen ve bu tezin oluşmasında büyük rol oynayan değerli tez danışmanım Doç. Dr. Bahadır YÜZBAŞI' ya,

Çalışmalarım sırasında bana maddi ve manevi yardımcı olan değerli arkadaşım Fuad HASANOV' a,

Çalışmam sırasında her zaman yanımda olup bana destek olan canım kardeşim Kardelen TAYA' ya,

Okul hayatım boyunca öğrenimime destek olan ve öğrenimim için her türlü imkânı sunan ve her zaman yanımda olan canım annem ve canım babam'a,

Sonsuz teşekkür eder saygılarımı sunarım.

ÖZET

Bu çalışmada Türkiye'ye ait dış ticaret verileri ile makine öğrenmesi yöntemlerinden yararlanılmış olup sınıflandırma çalışması yapılmıştır. Çalışma kapsamında makine öğrenmesi yöntemlerinden destek vektör makineleri ve lojistik regresyon modelleri kullanılarak model performans karşılaştırması yapılmıştır. Destek vektör makinelerinde radyal ve doğrusal çekirdek fonksiyonları kullanılmıştır. En iyi sınıflandırıcıyı seçmek için karışıklık matrisi değerleri ve ROC eğrisi kullanılmıştır. En iyi sonuç destek vektör makinelerinden olan radyal çekirdek fonksiyonundan elde edilmiştir.

Anahtar Kelimeler: Makine öğrenmesi, Destek Vektör Makineleri, Lojistik Regresyon, ROC Eğrisi, Karışıklık Matrisi, Sınıflandırma

ABSTRACT

In this study, foreign trade data of Turkey and machine learning methods were used and classification study was carried out. Within the scope of the study, model performance comparison was made by using support vector machines and logistic regression models from machine learning methods. Radial and linear kernel functions are used in support vector machines. Confusion matrix values and ROC curve were used to select the best classifier. The best result was obtained from the support vector machine, the radial kernel function.

Key Words: Machine Learning, Support Vector Machines, Logistic Regression, ROC Curve, Confusion Matrix, Classification

İÇİNDEKİLER

ONUR SÖZÜ.....	i
ÖNSÖZ	ii
ÖZET	iii
ABSTRACT.....	iv
İÇİNDEKİLER	v
ŞEKİLLER LİSTESİ	vii
TABLolar LİSTESİ	ix
KISALTMALAR	x
1. GİRİŞ.....	1
1.1. Makine Öğrenmesi Türleri.....	2
1.1.1. Denetimli Öğrenme	2
1.1.2. Denetimsiz Öğrenme	4
2. DESTEK VEKTÖR MAKİNELERİ	5
3. DİĞER SINIFLANDIRMA YÖNTEMLERİYLE DVM ARASINDA Kİ İLİŞKİ	7
3.1. Lojistik Regresyon	7
3.2. Karar Ağaçları	7
3.3. Doğrusal Diskriminant Analizi	7
4. DOĞRUSAL DESTEK VEKTÖR MAKİNELERİ.....	9
4.1. Doğrusal Ayrılabilir Durum (Sert Sınır)	9
4.2. Lagrange yöntemi.....	14
4.3. Krus-Kuhn-Tucker koşulları (KKT)	14
4.4. Tümüyle Doğrusal Ayrılamayan Durum (Yumuşak Sınır).....	16
4.5. Doğrusal Olmayan Destek Vektör Makineleri.....	21
4.5.1. Çekirdek Fonksiyonları (Kernel Functions).....	22
4.5.2. Kernel Fonksiyon Seçimi	25
4.5.3. DVM için Kerneller (DVM'ler de kullanılan kerneller)	25
4.5.3.1. Polinomal Kernel	25
4.5.3.2. Doğrusal kernel.....	25
4.5.3.3. Gaussiyan (radyal tabanlı) kernel	25

5. LOJİSTİK REGRESYON	27
5.1. Logit Dönüşüm ve Lojistik Regresyon Modeli.....	27
6. DEĞERLENDİRME KRİTERLERİ.....	30
6.1. Karışıklık Matrisi	30
6.2. Doğruluk	31
6.3. Duyarlılık (Hassasiyet).....	31
6.4. Özgüllük.....	31
6.5. Kesinlik	31
6.6. ROC Eğrileri	32
6.6.1. ROC Eğrileri Değerlendirilmesi.....	32
6.6.2. ROC Eğrileri Avantajları.....	32
7. UYGULAMA	34
7.1. R Programında e1071 Paketinde svm fonksiyonu	34
7.2. R Programında stats Paketinde glm fonksiyonu.....	34
7.3. Problemin Belirlenmesi.....	34
7.4. Verilerin Toplanması ve Hazırlanması	35
7.5. DVM Kurulumu	35
7.6. Eğitim ve Test Verisinin Ayrılması	40
7.7. Modelinin Oluşturulması	40
7.8. Karışıklık Matrisi Sonuçları.....	41
7.9. Eğitim ve Test Veri Seti Destek Vektörlerinin Belirlenmesi.....	42
8. SONUÇ VE YORUM	46
KAYNAKÇA.....	47
EKLER	49
Ek 1.Çalışmada Kullanılan Veri Seti	49
Ek 2. Çalışmada Kullanılan Kodlar	54

ŞEKİLLER LİSTESİ

Şekil 1.1. Makina öğrenmesi algoritmaları	2
Şekil 4.1. Hiper Düzlemler	9
Şekil 4.2. Maksimum Ayıran Hiper Düzlem	10
Şekil 4.3. Maksimum Marj Sınıflandırıcısı	10
Şekil 4.4. Destek Vektörleri	11
Şekil 4.5. Destek Vektörleriyle Birlikte Dar Marjin(a), Geniş Marjin(b) ve Ayırıcı Hiper Düzlem	12
Şekil 4.6. Kanonik Hiper Düzlemler ve Ayırıcı Hiper Düzlem	12
Şekil 4.7. Maksimal Marj Sınıflandırıcısının Olmadığı Veriler	17
Şekil 4.8. Ayırıcı Hiper Düzlemin Tek Gözlemle Yer Değiştirdiği Grafik	17
Şekil 4.9. Yumuşak Sınır Gösterimi	18
Şekil 4.10. Doğrusal Ayrılama Durumunda Zayıflık Değerleriyle Optimal Ayırıcı Aşırı Düzlem	19
Şekil 4.11. Doğrusal Olmayan Destek Vektör Makineleri	21
Şekil 4.12. Doğrusal Olmayan Veri Setinin Yüksek Boyutlu Uzayda Doğrusal Hiper Düzlem ile Sınıflandırılması	22
Şekil 4.13. Girdi Uzayından Özellik Uzayına Dönüşüm	23
Şekil 5.1. Lojistik Regresyon Grafiği	29
Şekil 7.1. Gıda ve Hammaddeye Göre Dış Ticaret Sınıflandırılması	36
Şekil 7.2. Gıda ve Yakıta Göre Dış Ticaret Sınıflandırılması	36
Şekil 7.3. Gıda ve İmalata Göre Dış Ticaret Sınıflandırılması	37
Şekil 7.4. Gıda ve Sıtc'a Göre Dış Ticaret Sınıflandırılması	37
Şekil 7.5. Hammadde ve Yakıta Göre Dış Ticaret Sınıflandırılması	38
Şekil 7.6. Hammadde ve İmalata Göre Dış Ticaret Sınıflandırılması	38
Şekil 7.7. Hammadde ve Sıtc'a Göre Dış Ticaret Sınıflandırılması	39
Şekil 7.8. Yakıt ve İmalata Göre Dış Ticaret Sınıflandırılması	39
Şekil 7.9. Yakıt ve Sıtc'a Göre Dış Ticaret Sınıflandırılması	40
Şekil 7.10. Gıda ve Yakıt Değişkenine Göre Eğitim ve Test Sınıflandırma Grafikleri	42

Şekil 7.11. Gıda ve İmalat Değişkenine Göre Eğitim ve Test Sınıflandırma	
Grafikleri.....	43
Şekil 7.12. Gıda ve Hammadde Değişkenine Göre Eğitim ve Test Sınıflandırma	
Grafikleri.....	44
Şekil 7.13. ROC Eğrisi	44



TABLÖLAR LİSTESİ

Tablo 6.1. Karışıklık Matrisi Oluşum Tablosu	30
Tablo 7.1. Çalışma Kapsamında Değişkenler ve Açıklamaları	35
Tablo 7.2. Sınıflandırma Yöntemlerinin Karşılaştırılması.....	41
Tablo 7.3. Karışıklık Matrisi Sonuçları.....	41
Tablo 7.4. AUC Değerleri.....	45

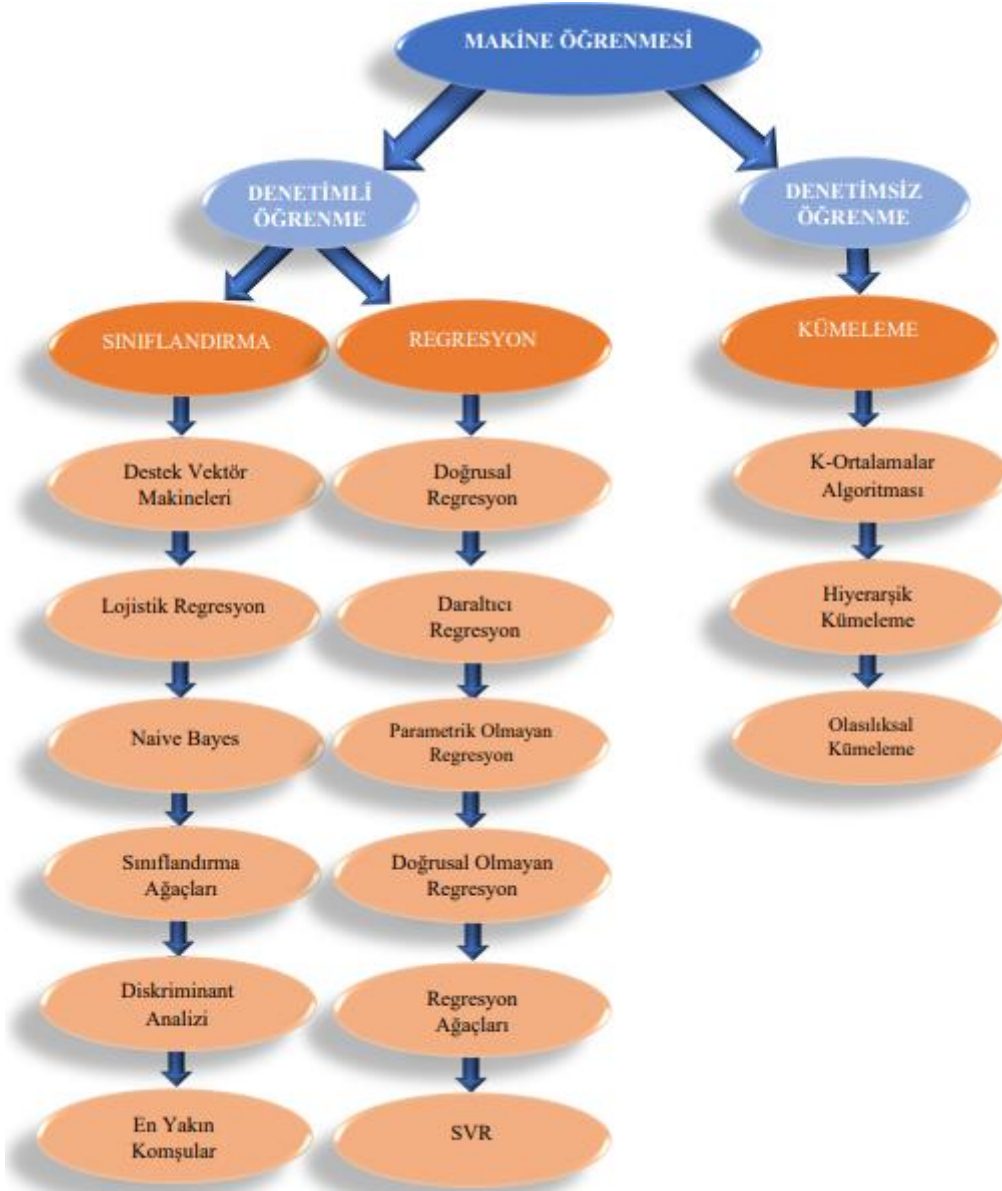


KISALTMALAR

DVM	:Destek Vektör Makineleri
AUC	:ROC Eğrisi Altında Kalan Alan (Area Under the ROC Curve)
ROC	:Alıcı İşletim Karakteristiği (Receiver Operating Characteristic)
TÜİK	:Türkiye İstatistik Kurumu
YSA	:Yapay Sinir Ağları
LR	:Lojistik Regresyon
KKT	:Krus-Kuhn-Tucker Koşulları

1. GİRİŞ

Son zamanlarda büyük veri ile adını sıkça duyduğumuz kavramlardan biri haline gelen makine öğrenmesi yapay zekanın bir alt dalıdır. Makine öğrenmesi matematik ve istatistik bilimlerinden yararlanarak hesaplamalar yapmaya çalışan ve yaptığı bu hesaplamalarla doğru bilgi aktarımını amaçlayan yöntemler bütünüdür. Makine öğrenmesi öğrenebilen algoritmaların kapsamına girmekle birlikte ona sunulan veri tabanlarıyla tahminler yapan sistemler bütünüdür. Makine öğrenmesi için ona belirli bilgiler verilerek öğrenmesi sağlanır daha sonra öğrendiklerini başka veriler üzerinde uygulaması ve nerede yanlış nerede ne kadar doğru tespit yapmış olduğu belirlenir. Bu sebeple makine öğrenmesi yöntemleri geliştirilmiştir. Kızılkaya ve Oğuzlar (2018: 92) Makine öğrenmesinin sağladığı kolaylıklardan biri, insan gücüyle yapılmak istenen işlemlerin çok fazla zaman alacağı ve iş gücü olarak zor olan hesaplamaların, bilgisayarlar tarafından çok kısa sürede kolaylıkla yapılabildiğine değişmiştirler. (Athey, 2019:509) bilgisayar bilimlerinde geliştirilen ve bilgisayar biliminin alt alanlarından oluşan makine öğrenmesi ekonometri, istatistik ve sosyal bilimlerde kullanılmaya başlayan yöntemleri ifade eder. Makine öğrenmesi verilere uygulanmak üzere tasarlanmış algoritmalar geliştiren bir alan olmakla birlikte, kümeleme ve sınıflandırma alanlarıyla yakından ilgili olduğuna değinmiştir. Makine öğrenmesi yöntemi kullanılan alanlara göre farklılık göstermektedir. Bu yöntemler denetimli öğrenme ve denetimsiz öğrenme olarak ikiye ayrılır. Şekil 1. 1. de görselleştirilmiş halini görebiliriz.



Şekil 1.2. Makina öğrenmesi algoritmaları

1.1. Makine Öğrenmesi Türleri

1.1.1. Denetimli Öğrenme

Denetimli öğrenme yöntemi çok sık kullanılan makine öğrenme yöntemidir. Bu öğrenme yönteminde verilerin eğitilmiş süreçlerine dayanarak oluşturulan modelleri ifade eder. Gözetimli öğrenmede sisteme sunulan giriş bilgileri ve modelin öğrenme sayesinde oluşturduğu çıkış bilgileri bulunur. Makineye sunulan girdi bilgileriyle makinelerin ürettiği çıktı bilgilerinin istenen değerle kıyaslanarak ara işlemde kullanılan

bilgi yöntemlerinin ifade biçimidir. Bu bilgiler şu şekilde daha açıklayıcı olabilir. X bağımsız değişkenine Y bağımlı değişkeninin bilgilerinden oluşan eğitim veri seti oluşturulur ki buna verinin eğitilmesi diyebiliriz. Oluşturulan modelin belirli bir kısmı örneğin 0.70 gibi bir oranla eğitim verisi oluşturulur. Geri kalan veri (0.3) oran ile test verisini oluşturur. Test verisi eğitim verisinin ne kadar doğru çalışıp çalışmadığını tespit etmek için kullanır. Bu işlemleri uygulamada kullanmak için denetimli öğrenme konuları olan regresyon ve sınıflandırma olarak iki alt başlıkta ele alınır.

Regresyon

Denetimli öğrenme yöntemlerinden biri olup iki veya daha fazla değişken arasındaki ilişkiyi belirler. Regresyon basit doğrusal regresyon, çoklu doğrusal regresyon olarak iki bölüme ayrılır. Basit doğrusal regresyonda Y_i bağımlı değişkenini sadece bir tane X_i bağımsız değişkeni açıklıyorsa, çoklu doğrusal regresyonda Y_i bağımlı değişkeni birden fazla X_i bağımsız değişkeni tarafından açıklanır. Basit regresyon modeli ekonometrik olarak

$$Y_i = \beta_0 + \beta_1 X_i + u_i, \quad i=1,2,3 \dots n \quad (1.1)$$

şeklinde gösterilir. Modelde yer alan Y_i ile ifade edilen değişken bağımlı değişken olarak ifade edilir. Denkleminde yer alan X_i 'se bağımsız değişken olarak tanımlanır. u_i hata terimidir. β_0 sabit parametredir ve doğrunun Y eksenini kestiği noktayı gösterir. β_1 eğim parametresidir. Bağımsız değişkende meydana gelen bir birimlik değişimin bağımlı değişkendeki etkisini belirtir. i alt-indeksleri belirlenmiş mümkün gözlemi ifade eder.

Çoklu regresyon modeli;

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \dots + \beta_p X_{ip} + u_i \quad i=1,2,3 \dots n \quad (1.2)$$

şeklinde ifade edilir. Y_i bağımlı değişkendir. $X_{1i}, X_{2i}, \dots, X_{ip}$ 'ler bağımsız değişkenlerdir. β_0 bağımsız değişken değerleri sıfır olduğu durumda bağımlı değişkenin alacağı değeri belirten sabit terimdir. $\beta_1 \dots \beta_p$ kısmi regresyon katsayıları olarak tanımlanır ayrıca β_p modelde p tane parametre, $p - 1$ tane açıklayıcı değişken bulunduğunu göstermektedir. u_i hata terimidir. i alt-indeksleri belirlenmiş mümkün gözlemi ifade eder.

Sınıflandırma

Nitel değişkenlerin analiz edildiği denetimli öğrenme algoritmasıdır. Bağımlı değişken nitel olduğu için iki veya daha fazla kodlanmış değişkenleri içerir (0,1). Kodlamanın fazla olduğu durumlara çoklu sınıflama denilir. Denetimli öğrenmede sınıflandırma algoritmaları DVM, naive bayes, karar ağaçları, lojistik regresyon ve diskriminant analizi gibi bazı makine öğrenmesi teknikleri kullanılmaktadır.

Ekonometri alanında literatür incelendiğinde dış ticaret verilerine göre herhangi bir sınıflandırma çalışmasına rastlanmadığından dolayı bu tez çalışmasında destek vektör makineleri ve lojistik regresyon yöntemi kullanılarak Türkiye'nin dış ticaret birim değer endeks verilerine göre ihracat ve ithalat sınıflandırma çalışması yapılacaktır ve en iyi sınıflandırıcı metodu seçilecektir.

1.1.2. Denetimsiz Öğrenme

Denetimsiz öğrenme tanımlanmamış verilerde kullanılır. Denetimsiz öğrenme önceden belirlenmiş verileri gözlemleyerek işlemeye çalıştığı sırada makine öğrenimi algoritmalarını kullanır. Bu işlem aşamasında insan müdahalesi gerekli değildir. Verilerdeki başlıca benzerlik ya da farklılıkları belirlemeye yarayan ve veriler hakkında daha çok bilgiye sahip olmak için kullanılan yöntemdir. Denetimsiz öğrenme algoritmasını denetimli öğrenme algoritmasıyla kıyaslayacak olursak denetimsiz öğrenme daha karmaşık işlemlerle çalışır ve bu çalışmasında doğru bir cevap yoktur. Yani karmaşık olan modelleri tanıyıp doğru gruplara ayırmaya çalışır.

Kümeleme analizi denetimsiz öğrenme algoritmalarının temel başlığı olup, K-ortalımalı kümeleme, hiyerarşik kümeleme, olasılıklı kümeleme gibi birkaç alt başlıkta incelenebilmektedir.

2. DESTEK VEKTÖR MAKİNELERİ

Birçok uygulama alanı olan makine öğrenmesinin denetimli öğrenme algoritması kullanılmaktadır. Bunun sebebi denetimli öğrenme algoritmalarının tahmin ve kestirim gibi sorunlara yoğunlaşıp tutarlı sonuçlar vermesidir. Denetimli öğrenme algoritmalarından ve istatistiksel öğrenme yöntemlerinden biri olan sınıflandırma nitel verilere değerler atayarak işleme alan yöntemdir. Sınıflandırma başlığı adı altında birçok sınıflandırma yöntemi vardır. Bu yöntemler lojistik regresyon, diskriminant analizi, en yakın komşu analizi sınıflandırma ağaçları ve DVM' dir. DVM denetimli öğrenme yöntemlerinden olup sınıflandırma ve regresyon analizlerinde kullanılır. DVM' nin kullanımını arttıran ve diğer metotlardan üstünlüğünün başlıca nedeni öğrenme esnasında bilinmeyen verilerde işlemin az olması ve başarı olasılığının yüksek olmasından kaynaklanır. DVM çok boyutlu veriyi bir hiper düzlemden yararlanarak iki farklı sınıfa ayırmayı amaçlayan sınıflandırma yöntemidir. Literatür incelendiğinde DVM' nin genel sınıflandırma uygulamalarına nazaran daha etkili ve doğru kestirimler yaptığı sonucuna varılmıştır. DVM metin, karakter, hücre, görüntü gibi kullanım alanları oldukça geniş çerçevede incelenebilir ve çalışmalar yapılarak literatüre katkı sağlayabilir bir sınıflandırma türüdür. DVM ile ilgili olarak Cortes ve Vapnik (1995:273) DVM sınıflandırma için kullanılan yeni bir yöntem olup doğrusal olmayan çok yüksek boyutluluk özelliği olan işlemlerde doğrusal karar yüzeyi oluşturur ve sınıflandırma işlemini gerçekleştireceğine değinmiştir. Karagül (2013:174-178) yapmış olduğu çalışmada İstanbul Menkul Kıymetler Borsası 100 (İMKB-100) içinde gıda, tekstil ve çimento sektörlerinde işlem gören şirketleri ele almıştır. Yaptığı çalışmada DVM ile sektör sınıflandırması uygulayıp destek vektör makinelerinde %97,61 başarı sağlamıştır. Kara ve Ecer (2018: 185-195) yayınlamış oldukları BİST Banka Endeksi hareket yönünün tahmininde DVM ile birlikte diğer sınıflandırma yöntemlerinin performanslarını karşılaştırmışlardır. Tayyar ve Tekin (2013 :189-218) DVM ve LR kullanarak İMKB-100 Endeksinin DVM ile Günlük, Haftalık ve Aylık Veriler Kullanarak Tahmin Edilmesini sağlamışlardır.

Yakut vd. (2014: 139-157) yapmış oldukları çalışmada ileri beslemeli yapay sinir ağları ve DVM yöntemleri kullanarak BIST endeksleriyle tahmin gerçekleştirmişlerdir. Elde ettikleri sonuçlara göre DVM ve ileri beslemeli yapay sinir ağlarıyla BIST endeks

çalışması yapılabileceği kanısındadır. Destek vektör makineleriyle yapılan çalışmalarında Niu ve Suen (2012) insan denekler üzerinde yaptıkları çalışmada el yazısıyla yazılmış rakamları tanımak için yeni bit hibrit CNN (Evrışimli sinir ağı) ve DVM (destek vektör makinesi) ile sınıflandırıcı çalışması yapmışlar ve aynı veri üzerinde diğer çalışmalarla kıyaslandığında bu hibrit modelin daha iyi sonuçlar elde ettiğini belirtmişlerdir. DVM' nin kullanılmış olduğu farklı alanlardaki çalışmalarında Güran vd. (2014: 86-93) Duygu analiziyle ilgi çalışmalarında sınıflandırma metotlarından DVM kullanarak farklı parametreler üzerinde DVM başarımının etkili olduğunu belirtmişlerdir. Yapmış olduğu yüksek lisans tez çalışmasında Eray (2008: 1-85) DVM kullanarak ses tanıma yapmıştır ve %91 ve %71 oranla başarı elde etmiştir. DVM geliştirmiş olup başarısını kanıtlayan Akgün (2016: 1-44) Apache spark streaming teknolojisi üzerinde DVM sınıflandırma yöntemi geliştirmiş ve geliştirdiği programı LR ile karşılatırmıştır. Çalışmada DVM yönteminin LR göre daha etkin ve başarılı sonuç alındığı tespit edilmiştir. Literatüre baktığımızda DVM ve diğer sınıflandırma metotlarıyla karşılaştırma yapılmıştır ve alınan sonuçlardan Güneren (2015: 1-44) Gömülü sistem üzerinden yüz tanıma uygulamasında DVM, yapay sinir ağları (YSA) ve K- en yakın komşuluk yöntemini kullanarak sınıflandırma çalışması yapmış ve bu çalışmasında DVM başarısı diğer yöntemlere nazaran %91 e ulaştığı görülmektedir.

DVM genel anlamıyla iki sınıfa ait verileri uygun şekilde ayırabilmek için hiper düzlem kullanır. Destek vektör makineleri' nin asıl amacı sınıflandırmada iki sınırı en iyi şekilde ayıran hiper düzlemi çizmektir. DVM, çizilen bu hiper düzlem ile destek vektör makineleri arasındaki marjın (uzaklık) maksimum olacak şekilde hiper düzlemi kullanarak sınıflandırma işlevini gerçekleştirmeye çalışır. Marjin olarak bahsedilen terim ayırıcı olan hiper düzlemden en yakın veri noktasına olan kısa uzaklığı belirtir. DVM, doğrusal DVM, doğrusal olmayan DVM olmak üzere iki grupta incelenir.

3. DİĞER SINIFLANDIRMA YÖNTEMLERİYLE DVM ARASINDA Kİ İLİŞKİ

Denetimli öğrenme yöntemlerinden olan sınıflandırma için literatür incelendiğinde de birçok yöntem mevcuttur. Bu konunun başlangıcında bahsedildiği üzere DVM diğer sınıflandırma yöntemlerine göre kullanımı daha kolaydır ayrıca diğer metotlara nazaran başarı oranı yüksektir. Ancak bu başlık altında diğer sınıflandırma yöntemlerinden birkaçına değinerek kısa bilgi verilecektir.

3.1. Lojistik Regresyon

LR kullanımına 1945'li yıllarda başlanmış olup sosyal bilgiler alanında yaygın bir şekilde kullanılmaktadır. LR da bağımlı değişken kategorik, bağımsız değişkenler veri halindedir. LR kernel hilesini DVM 'de kullanılmak istenmediği taktirde LR kernel kullanarak işlemleri yapabiliriz.

3.2. Karar Ağaçları

Karar ağaçları sınıflandırma problemlerinde yaygın olarak kullanılmasıyla birlikte regresyon problemlerinde de kullanılmaktadır. Sınıflandırma problemlerinde nicel veriyle çalışmak yerine nitel verileri tahmin etmek için kullanılır. Karar ağaçları niteliksel olarak kukla değişkenlerini oluşturur. James vd.,(2013:303-316) karar ağaçlarının kavranması ve kolayca yorumunun yapılması ayrıca her türden insan tarafından anlaşılabilen bir yapısı vardır ancak karmaşık problemlerde dvm ile kıyaslanacak olursa dvm ye göre daha az tercih edilmelidir diyebiliriz.

3.3. Doğrusal Diskriminant Analizi

Çok değişkenli istatistiksel yöntemlerden olan sınıflandırma da çok eskiden beri kullanılan yöntem olup güvenilirliğini hala korumaktadır. Diskiriminant analizinde verinin içerisinde ki her nitelik için ayrı ayrı işlem yapılarak hangi sınıfa ait olduğunu bulunmaktadır. Diskiriminant analizinin başlıca amacına değinecek olursak ayırma ve sınıflandırma yapmaktır. Diskiriminant analizi çok kullanılan bir yöntem olup çok güçlü teorik bir yapıya sahiptir ancak doğrusal diskiriminant analizinin doğrusal olmayan verilerde kullanılması uygun değildir. DVM ile karşılaştıracak olursak dvm doğrusal olmayan veriler için çözüm üretebilmektedir. Diskiriminant analizi çözüm üretse dahi

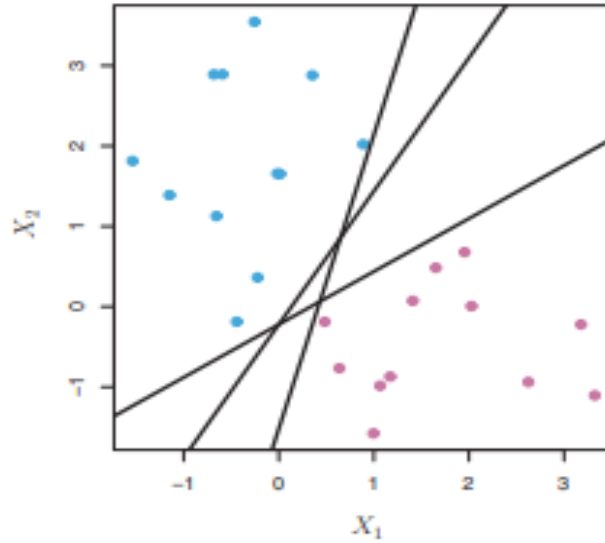
genelleřtirme performansı düşük olacađından analiz ve sonuçları etkili sonuç vermeyecektir.



4. DOĞRUSAL DESTEK VEKTÖR MAKİNELERİ

4.1. Doğrusal Ayrılabilir Durum (Sert Sınır)

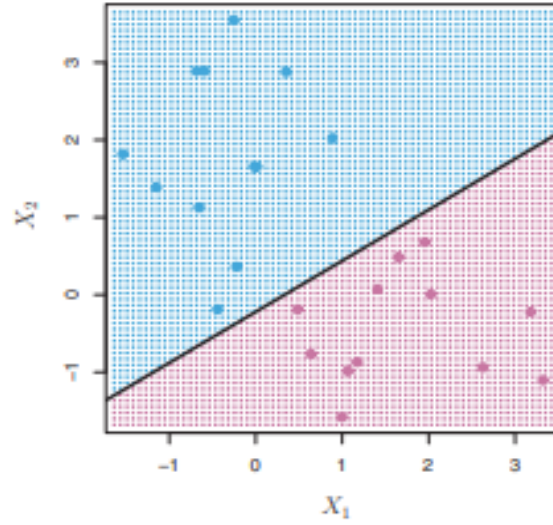
DVM en basit olarak tanınan modelidir. Doğrusal olarak ayrılabilen verilerde kullanılması uygundur. Ancak günümüzde gerçek veri kullanımından kaynaklı olarak bu model pek kullanışlı değildir. Yine de destek vektör makinelerini daha iyi kavrayabilmek için bu konunun anlatılması gereklidir. James vd., (2013: 341) doğrusal olan destek vektör makinelerinde veriler hiper düzlem aracılığıyla problemsiz olarak ayrılıyorsa burada kullanılabilir olan hiper düzlem sonsuz denilecek kadar çok olabilmektedir. Bu işlemin nedeni oluşturulan hiper düzlemin hiçbir gözlemin üzerinden geçmeden kaydırılabilir ya da aşağı yukarı döndürülebilir olduğunu belirtmişlerdir şekil 4. 1. de görebiliriz.



Kaynak: (James vd., 2013: 340)

Şekil 4.1. Hiper Düzlemler

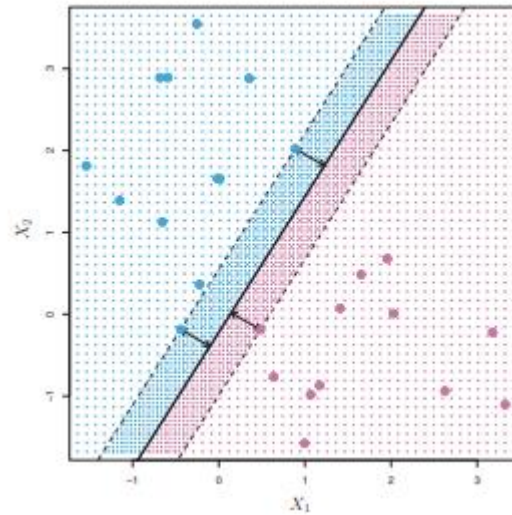
Şekil 4.1. de olası hiper düzlemler görülmektedir. Veriyi ikiye bölen üç tane hiper düzlem görülmektedir. Ama bu hiper düzlemlerin sadece bir tanesi doğru bir şekilde maksimum ayırmaya sahiptir. Bu ayırma düzlemi şekil 4. 2. de gösterilmektedir.



Kaynak: (James vd., 2013: 340)

Şekil 4.2. Maksimum Ayıran Hiper Düzlem

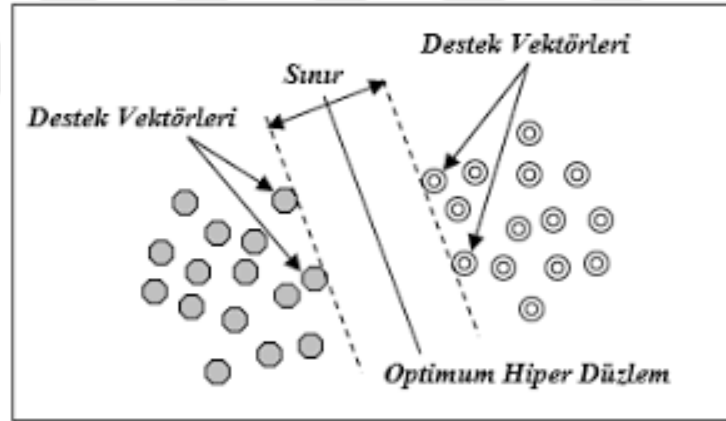
Hangi hiper düzlemin daha iyi ve doğru karar verdiğini görmek için maksimum marj sınıflandırıcısından yararlanılabilir. En kullanışlı ve doğru hiper düzlemin olması için veri kümelerine eşit uzaklıkta olabilmelidir. Yani X veri kümesine daha yakın olup ya da daha uzak olursa doğru bir hiper düzlem bulunmuş olmaz bu sebeple bu tür sorunlarla karşılaşmamak için maksimum marj sınıflandırıcısından yararlanılır.



Kaynak: (James vd., 2013: 342)

Şekil 4.3. Maksimum Marj Sınıflandırıcısı

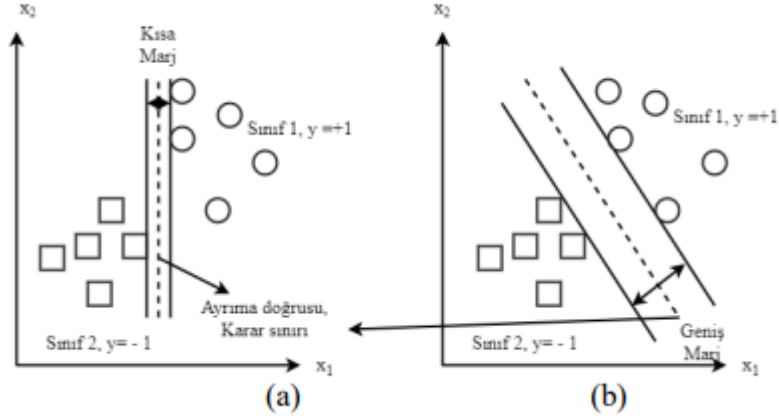
Şekil 4.3'te iki sınıf arasına eklenen ortadaki düz koyu renkli çizgi maksimal marj düzlemidir. Marj (sınır) olarak adlandırılan yer kesikli çizgilerle düz çizgi arasında bulunan mesafedir. Grafikte gördüğümüz iki mavi ve bir kırmızı nokta destek vektörleridir. Destek vektör makinesinde sınıflandırma işleminde sınıflar birbirinden ne kadar uzak olursa yani marjın genişliği ne kadar çok olursa o kadar doğru sınıflandırma olur aksi taktirde sınıflar ne kadar yakın olursa sınıfların doğru ayrılma oranı düşecektir. Maksimum marj sınıflandırıcısı destek vektörleri dışında hiçbir gözleme bağlı değildir. Maksimum marjın tanımını iki sınıf üzerinden belirtecek olursak eğitim verilerimiz $X = (X_1, X_2, \dots, X_n)$ ve $y_i \in \{-1, 1\}, i = 1, 2, \dots, n$ ve $x_i \in R^n$ şeklinde tanımlanır. Hiper düzlemin bir tarafında kalan kısım $y_i = -1$, diğer tarafında kalan kısım ise $y_i = +1$ 'dir. Bu bilgilerden anlaşıldığı üzere hiper düzlem her iki tarafta bulunan sınıflara eşit ve en yüksek mesafede bulunmaktadır. Marja yakın bulunan noktalar ise destek vektörleridir. Marja yakın olan veya marj üzerinde duran destek vektörleri şekil 4.4.'te detaylı olarak inceleyebiliriz.



Kaynak: (Bulut ve Günlü, 2016: 530)

Şekil 4.4. Destek Vektörleri

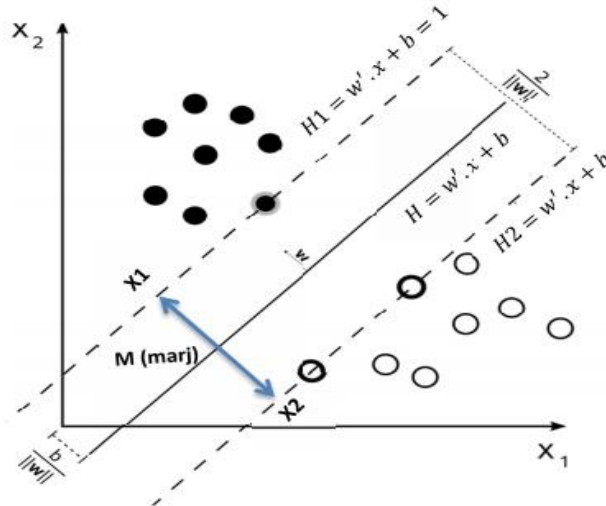
Şekil 4. 1.'de destek vektörleriyle birlikte marjının genişliğini, yakınlığını ve optimal hiper düzlemi detaylı bir şekilde görebiliriz.



Kaynak: (Doğan, 2020: 76)

Şekil 4.5. Destek Vektörleriyle Birlikte Dar Marjin(a), Geniş Marjin(b) ve Ayrıcı Hiper Düzlem

Optimal olarak hiper düzlem $(w' \cdot x) + b = 0$ olarak tanımlanır. Hiper düzlemin karar fonksiyonu $f(x) = (w' \cdot x) + b$ şeklinde ifade edilir. Burada w bir ağırlık vektörünü (veya aşırı düzlemin normalini), x destek vektörlerini, b skaleri de bir sapma değerini ifade etmektedir. Gözlemler pozitif ($y = +1$) ve negatif ($y = -1$) şeklinde sınıflandırılır, burada y sınıf etiketini temsil etmektedir. Bu bilgileri şekil 4. 6. da görebiliriz.



Kaynak: (Karakaynak, 2014: 18)

Şekil 4.6. Kanonik Hiper Düzlemler ve Ayrıcı Hiper Düzlem

Doğrusal destek vektörlerinde ağırlık w ile gösterilir. Destek vektörleri arasındaki şekil uzaklığı $\frac{2}{\|w\|}$ ile hesaplanır ve destek vektörleri arasındaki marjın en geniş uzaklığını hesaplamak için $\frac{\|w\|}{2}$ ifadesi ile mümkün hale gelir. $\|...\|$ bir normdur. Bu şekilde doğru hiper düzlemi bulmak mümkündür ve olası hiper düzlem $(w'.x) + b = 0$ şeklinde yazılır. Kanonik hiper düzlemleri bulmak için aşağıdaki denklem oluşturulur.

$$H_1 = (w'.x) + b = +1 \quad (4.1)$$

$$H_2 = (w'.x) + b = -1 \quad (4.2)$$

Marjı maksimal yapmak için hiper düzlemler (H_1 ve H_2) arasındaki uzaklığı bulmak önemlidir. Bir tane x noktasının H_1 ve H_2 üzerindeki etkisini kullanacak olursak H_1 için x_1 , H_2 için x_2 şeklinde gösterilir. Aynı zamanda geometrik olarak uzaklığın hesaplanmasında aşağıdaki işlemler kullanılır.

$$M = H_1 - H_2 \quad (4.3)$$

$$(w' (x_1 - x_2)) = 2 \quad (4.4)$$

$$\left(\frac{w'}{\|w\|} (x_1 - x_2)\right) = \frac{2}{\|w\|} \quad (4.5)$$

$$M = (x_1 - x_2) = \frac{2}{\|w\|} \quad (4.6)$$

Bu yapılan işlemlerde asıl amacımız marjı en yüksek seçmektir. Yani $\frac{2}{\|w\|}$ formülünün maksimum yapılmasıdır. Özellikle $\|w\|$ ağırlığının minimum yapılması gereklidir. Bu işlem yapılırken destek vektörlerinin ya da verilerin marjın içine düşmemesi gereklidir bunu engellemek için minimizasyon problemi yapılır.

$$y_i = -1 \quad (w' x) + b \leq -1 \quad (4.7)$$

$$y_i = +1 \quad (w' x) + b \geq +1 \quad (4.8)$$

Oluşturulan minimizasyon problemini şekil 4. 6. da görebiliriz. Oluşturduğumuz minimizasyon problemini tek bir denklemde birleştirebilmektedir.

$$y_i(((w' x) + b) - 1) \geq 0, \quad \forall_i \quad (4.9)$$

Yukarıda gördüğümüz denklemde son işlem aşağıdaki gibidir.

Min : $\|w\|$

kısıtlar

$$y_i((w' x) + b) \geq 1 \quad i=1,2,3,\dots,n \quad (4.10)$$

şeklindedir.

Ayrıca bu denklemi çözmek biraz zordur. Bu sebeple

$$\|w\| = \sqrt{w'w} = \sqrt{w_1^2 + w_2^2 + \dots + w_n^2}$$

şeklinde açarak işlemi kolaylaştırabiliriz. Yani ağırlık vektörümüz olan w kareköklerle işlem yapılır. Bu denklemlerle yaptığımız işlemlere optimizasyon problemi diyebiliriz.

4.2. Lagrange yöntemi

1961 yılında Lagrange tarafından geliştirilmiş olup, çok değişkenli optimizasyon problemlerinin çözümü için kullanılan yöntemdir. DVM de optimal hiper düzlemi bulmak için Lagrange yönteminden yararlanılır.

Optimizasyon problemi:

$$Max: f(x, y)$$

$$Kısıtlar: g(x, y) = c$$

Bu optimizasyon problemi için Lagrange metodu 4.11 eşitlikte gerçekleştirilir.

$$L(x, y, \alpha) = f(x, y) - \alpha_i(g(x, y) - c) \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, n \quad (4.11)$$

Yukarıdaki kısıtlarda bulunan eşitlikler için Lagrange çarpanı vardır ve Lagrange metodunun x ve y 'ye göre türevini alıp sıfıra eşitlememizin amacı en küçük ve en büyük değere ulaşmamız içindir. Lagrange yöntemi eşitlik olduğunda başvurduğumuz metotlardandır. Ancak kısıtlarda her zaman eşitlik sağlanamayabiliyor bu durumda Krush-Kuhn-Tucker koşulları yöntemine başvurulur.

4.3. Krush-Kuhn-Tucker koşulları (KKT)

Lagrange yönteminde belirttiğimiz gibi kısıtlar da eşitlik olduğunda Lagrange yönteminden yararlanıyorduk ancak kısıtlarsa eşitlik olmadığı durumlar da görülmektedir. Bu durumlarda KKT koşullarından yararlanabiliriz. KKT koşulları

doğrusal olmayan optimizasyon problemleri için en yararlı ve özellikleri bakımından etkin koşullardır. Lagrange çarpanları eşitsizlik içermeyen koşullarda KKT çarpanları Lagrange çarpanlarına dönüştürülür. Lagrange yöntemini aşağıdaki şekilde yazarak işlemlere başlayabiliriz.

$$L(w, b, \alpha) = \frac{1}{2} w'w - \sum \alpha_i [y_i(w'x_i + b) - 1], \quad \alpha_i \geq 0, \quad i = 1, 2, 3, \dots, n \quad (4.12)$$

Yukarıda belirtilen Lagrange yönetiminin b ve w ye göre birinci dereceden türevini alıp sıfıra eşitlemek gereklidir.

$$\frac{\partial L(w, b, \alpha)}{\partial w} = w' - \sum_{i=1}^n y_i \alpha_i x_i = 0 \quad w = \sum \alpha_i y_i x_i \quad (4.13)$$

$$\frac{\partial L(w, b, \alpha)}{\partial b} = - \sum_{i=1}^n \alpha_i y_i = 0 \quad \sum \alpha_i y_i = 0 \quad (4.14)$$

b ve w ye göre türevini alıp elde ettiğimiz bu sonuçlara KKT koşulları denir. Bu koşullara bağlı olarak KKT koşulları (4.12) de yerine yazılırsa aşağıdaki sonuçlar ortaya çıkmaktadır.

$$\begin{aligned} L(w, b, \alpha) &= \frac{1}{2} \langle \sum \alpha_i y_i x_i, \sum \alpha_j y_j x_j \rangle - \sum \alpha_i [y_i (\langle \sum \alpha_j y_j x_j, x_i \rangle + b) - 1] = \\ &= \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \langle x_i x_j \rangle - \sum_{i=1}^n \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle - b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \alpha_i \\ &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x'_i x_j \end{aligned} \quad (4.15)$$

(4.15)'de gördüğümüz üzere w yi bulmak için Lagrange çarpanını bilmemiz gereklidir. Lagrange çarpanları eşitlik olduğunda kullanılmaktaydı ve KKT ise eşitlik olmadığı durumlarda kullanılmaktadır. Yani (4.15)'de w nin doğrusal kombinasyonları bulmaktayız optimum noktayı bulabilmek için Lagrange çarpanı olan α_i 'i maksimum yapılmalı aynı zamanda w ve b ye göre Lagrange fonksiyonu minimum sırt noktasını bulalım. Bu tür problemler için iki çözüm vardır. Primal ve dual olarak çözümlene bilmektedir. Primal uzayda (w ve b)'ye göre, dual uzayda ise (α)'ya göre çözümlenebilir (Wang,2005).

Genel olarak optimum sırt noktasının problemini çözerken iki çözüm yöntemi de aynı sonuca ulaşmaktadır. Fakat primal çözüm yöntemi dual çözüme göre daha karmaşık yapıya sahiptir. Bu nedenle biz burada dual çözüm uzayı'nı tercih edeceğiz.

$$L_{dual}(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x'_i x_j \quad (4.16)$$

$$\text{Maximize } L_{dual}(\alpha)$$

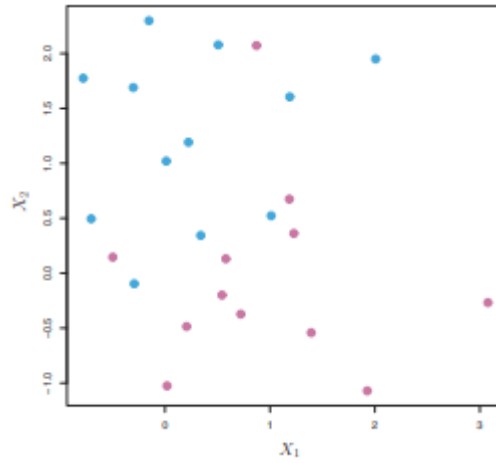
$$\text{Kısıtlar } \alpha_i \geq 0, \quad i = 1, 2, \dots, n$$

$$\sum \alpha_i y_i = 0 \quad (4.17)$$

Bu şekilde oluşturulan dual çözüm problemi optimizasyon probleminin ikinci dereceden çözümü ile yapılır. Genel olarak bakıldığında her bir çözüm eğitim örneği için birer tane Lagrange çarpanı olduğu fark edilir. Lagrange çarpanlarının $\alpha_i \geq 0$ olduğunu ve genel olarak 0 olduğunu bilmekteyiz. $\alpha_i \geq 0$ ve x_i yapıda olan verilere destek vektörleri adını verebiliriz. Destek vektörleri konunun ilk başında bahsedildiği üzere H_1 ve H_2 aşırı düzleminde yer alan noktalardır. Lagrange çarpanları $\alpha_i = 0$ olan x_i değerleri destek vektörleri değildir. Yan Lagrange çarpanları $\alpha_i \geq 0$ olan değerler destek vektörleridir ve aktif kısıtlardır diyebiliriz. H_1 ve H_2 aşırı düzlemin arkasında bulunan yani destek vektör olmayan, Lagrange çarpanları $\alpha_i = 0$ olan değerlerdir.

4.4. Tümüyle Doğrusal Ayrılamayan Durum (Yumuşak Sınır)

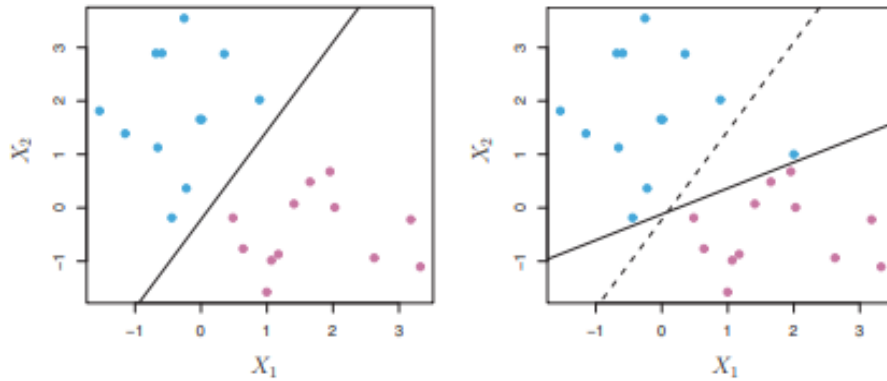
DVM giriş kısmında da belirttiğimiz gibi çoğu değişkenler, veriler doğrusal bir şekilde ayrılmamaktadır bunun sebebi ise gerçek verilerdir. Maksimum marj sınıflandırıcısı ayırma hiper düzleminin olduğu durumda ortaya çıktığından ayırıcı hiper düzlemin olmadığı yerde maksimal marj sınıflandırıcısından bahsetmek mümkün değildir. Bazı durumlarda ise veriyi tam ortadan ayırmak çok güçtür. Bu gibi durumlarda yumuşak marj kullanabiliriz, yani yaklaşık olarak iki sınıfa ayrılabilen hiper düzlemi bulabiliriz.



Kaynak: (James vd., 2013: 344)

Şekil 4.7. Maksimal Marj Sınıflandırıcısının Olmadığı Veriler

Şekil 4. 7. de iki farklı renkte ayrılmış olan iki sınıf vardır ve bu iki sınıf ayırıcı bir hiper düzlem ile ayıramaz ve. Bu durumda maksimal marj sınıflandırıcısından bahsetmek mümkün değildir. Destek vektör sınıflandırıcı olarak adlandırılan tanım ise maksimal marj sınıflandırıcısının kullanılmadığı, hiper düzlemin tam olarak doğrusal ayıramadığı veriler için adlandırılan bir tanımdır diyebiliriz. (James vd., 2013: 344)

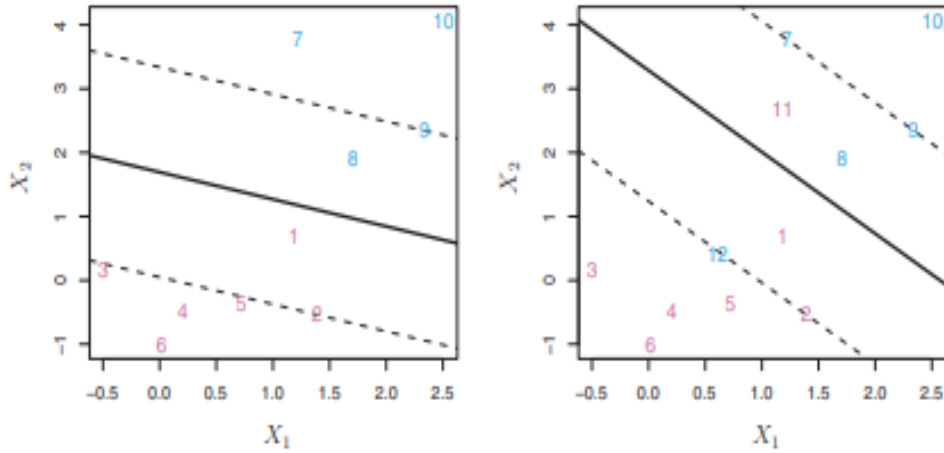


Kaynak: (James vd., 2013: 345)

Şekil 4.8. Ayırıcı Hiper Düzlemin Tek Gözlemle Yer Değiştirdiği Grafik

Şekil 4.8’de görebileceğimiz üzere sınıfa tek bir gözlem eklenmesi veya yer değiştirmesiyle birlikte maksimal marj hiper düzleminin yer değiştirdiği açık bir şekilde görülmektedir. Yer değiştiren hiper düzlemle birlikte marj dar olarak ifade edilir. DVM de marj ne kadar geniş yapıya sahip olursa sınıflandırma gerçek ve doğru şekilde

oluşacaktır. Şekil 4.8’den anlaşılacağı üzere maksimal marj sınıflandırıcısı tek bir gözleme dahi duyarlı bir yapıya sahiptir. Tek bir değişkene dahi duyarlı olan maksimal marj sınıflandırıcısında tek değişkenle yer değiştirmesini veya bu gibi sorunları çözmek için bütün verileri tam olarak doğru bir hiper düzlemle ayırmak sorun teşkil ettiğinden genel olarak eğitim verisinin birçoğunu doğru sınıflandıran hiper düzlemi bularak çözüm üretebiliriz. Bu işlemin başlıca uygulanan metodu destek vektör sınıflandırıcılarıdır. Destek vektör sınıflandırıcılarında bazı değişkenlerin hiper düzlemin diğer tarafında olmasına tolerans sağlayan ama geriye kalan eğitim verilerini doğru tarafta bulunduran sınıflandırıcılarıdır. Bu konunun başlığının ana kaynağı marjının yumuşak olması destek vektör sınıflandırıcılarının toleranslarından kaynaklanmaktadır. Yani bazı gözlemlerin yanlış tarafta olmasından dolayı yumuşak marjın olarak adlandırılır.



Kaynak: (James vd., 2017: 346)

Şekil 4.9. Yumuşak Sınır Gösterimi

Şekil 4.9’a baktığımızda yumuşak sınır örneğini görebiliriz. Sol taraftaki şekilde 1 ve 8 numaralı gözlemler marjın yanlış tarafındadır sağ taraftaki şekilde ise sol taraftaki şekle nazaran 11 ve 12 gözlemleri eklenmiş olup 12 marjın üzerindedir görülmekte ama 11, 1 ve 8 gözlemi marjın yanlış tarafında bulunmaktadır. Bu açıklamaları sadeleştirecek olursak yukarıda bahsettiğimiz gibi marjın yumuşak olması bütün eğitim gözlemlerinin yanlış sınıflandırılmasını tercih etmektense birkaç tane gözlemin yanlış yerde bulunmasını normal karşılayabilir ve bu duruma yumuşak marj denir.

Tümüyle doğrusal ayrılamayan durumlarda hataları ifade etmek amacıyla zayıflık ξ_i değişkenleri modele eklenerek çözüm bulunmaya çalışılır. ξ_i değişkeni pozitif yapıya sahiptir. (4.18) ve (4.19)'da ki eşitlikler zayıflık değişkenine göre düzenlenecek olursa

$$y_i = +1 \quad \text{için} \quad w' \cdot x_i + b \geq +1 - \xi_i \quad \xi_i \geq 0 \quad (4.18)$$

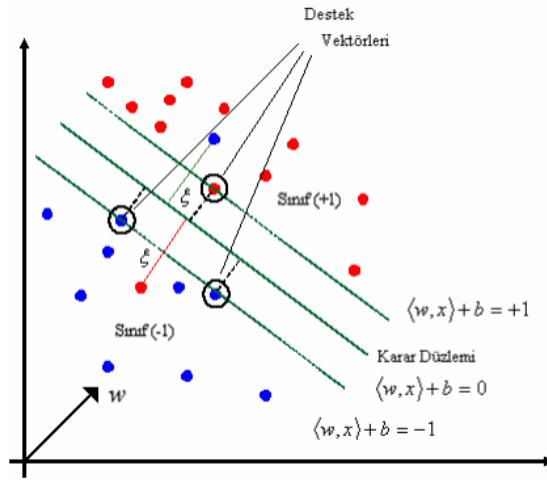
$$y_i = -1 \quad \text{için} \quad w' \cdot x_i + b \geq -1 + \xi_i \quad \xi_i \geq 0 \quad (4.19)$$

eşitsizlikleri ortaya çıkacaktır. Bu eşitsizliklere dayanarak problem şu şekilde tekrardan oluşturulur ise

$$\text{Minimizasyon} \quad f(w, \xi) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i$$

$$\begin{aligned} \text{Kısıtlar:} \quad & y_i(w' \cdot x_i + b) \geq +1 - \xi_i \quad i = 1, 2, \dots, n \\ & \xi_i \geq 0 \quad i = 1, 2, \dots, n \end{aligned} \quad (4.20)$$

oluşur.



Kaynak: (Eray, 2008: 51)

Şekil 4.10. Doğrusal Ayrılamama Durumunda Zayıflık Değerleriyle Optimal Ayrıcı Aşırı Düzlem

Şekil 4.10'da zayıflık değişkenleriyle birlikte optimum ayırıcı aşırı düzlemi görmekteyiz. Sınıflandırma probleminde marjın genişliğinin önemli olduğunu daha önce de ifade etmiştik. Bu sebeple marjın genişliğinin kontrol edilmesi için ve eğitim verilerimizin hep aynı yapıda çalışmasını engellemek amacıyla modele C parametresi

ekleyebiliriz. C parametresi doğru sınıflandırılmayan gözlemler için ölçüm parametresidir. Yani eğitim verilerimiz için ceza parametresi olarak ayrıca marjinin genişliğini ayarlamak için de düzenleyici bir parametre görevi görebilir. C parametresi Lagrange çarpanlarıyla doğrudan ilişkili olup Lagrange çarpanlarının bulunduğu en yüksek aralığı belirtmektedir. Lagrange çarpanları C parametresi ile şu şekilde $0 \leq \alpha_i \leq C$ gösterilir. C parametresinin eklenmesiyle birlikte Lagrange çarpanları aşağıdaki forma dönüşür.

$$L_{primal} = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i \{y_i (x_i \cdot w + b) - 1 + \xi_i\} - \sum_{i=1}^n \mu_i \xi_i \quad (4.21)$$

Bu eşitlikte μ_i parametresi ξ_i zayıflık değişkeninin pozitif olması için kullanılan lagrange çarpanlarından biridir. Bu formda oluşturulan problemi çözmek için KKT koşulları yeniden oluşturulur.

$$\frac{\partial L_{primal}}{\partial w'} = w' - \sum_{i=1}^n \alpha_i y_i x_i = 0 \quad (4.22)$$

$$w' = \sum_{i=1}^n \alpha_i y_i x_i \quad (4.23)$$

$$\frac{\partial L_{primal}}{\partial b'} = \sum_{i=1}^n \alpha_i y_i = 0 \quad (4.24)$$

$$\frac{\partial L_{primal}}{\partial \xi_i} = \sum_{i=1}^n \alpha_i y_i = 0$$

(4.24) eşitliğinin türev işlemi zayıflık değişkenine uygulanır.

$$\frac{\partial L_{primal}}{\partial \xi_i} = C - \alpha_i - \mu_i = 0 \quad (4.25)$$

Zayıflık değişkenlerinin türevinin alınmasıyla oluşan denklem aşağıdaki eşitliği oluşturmaktadır.

$$\alpha_i = C - \mu_i \quad (4.26)$$

Bulduğumuz bu yeni kısıtlar ve değerler Lagrange çarpanlarında yerine koyularak işlem yapılmak istense aşağıdaki dual problem ortaya çıkar.

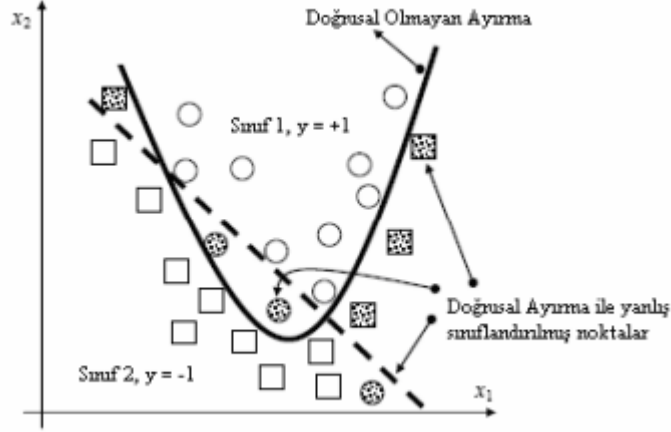
$$\text{Kısıtlar:} \quad 0 \leq \alpha_i \leq C \quad i = 1, 2, \dots, n$$

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad i = 1, 2, \dots, n \quad (4.27)$$

Bu eşitliklerde destek vektörleri x_i olarak gösterilen değerlerdir ve bu değerler lagrange çarpanlarında $0 \leq \alpha_i \leq C$ genişliğinde yer almaktadır. Son eşitlik ise dual problemlerde kullanılacak olan kısıttır.

4.5. Doğrusal Olmayan Destek Vektör Makineleri

Bu bölüme kadar olan süreçte doğrusal destek vektörlerini ve doğrusal olarak ayrılamayan verilere zayıflık değişkeni eklenerek optimizasyon problemlerine çözüm ürettik. Bu bölümde gerçek eğitim verilerimizin doğrusal olarak ayrılamadığı ve birden çok farklı bileşenlerden oluşan ve doğrusal olarak ayrılamayan problemlere çözüm araştıracağız. Doğrusal olmayan sorunların çözümünde çekirdek fonksiyonlarıyla eğitim verilerinin yeni ve yüksek boyutlu uzayda birleştirip yeni çözümler üretilmektedir. Doğrusal olmayan destek vektörlerinin de karar sınırlarını oluşturmak için doğrusal olmayan bir hiper yüzey oluşturmak ve oluşturulan hiper yüzey ile sınıflandırma yapılabilmesidir.



Kaynak: (Eray, 2008: 53)

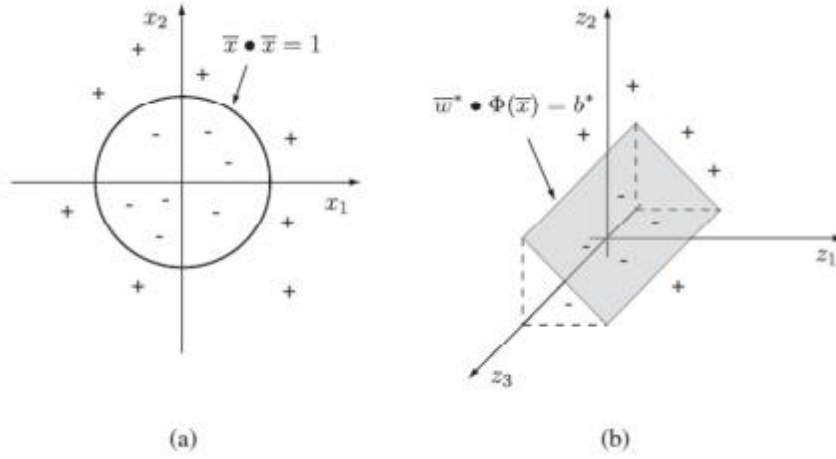
Şekil 4.11. Doğrusal Olmayan Destek Vektör Makineleri

Destek vektörlerinin başlıca amacı ve metodu x girdisini oluşturan $X \in R^n$ vektörlerini

$\Phi(x)$ yönteminde kullanarak daha yüksek bir boyutta F' e yaklaştırarak $(R^n \rightarrow F^n)$ doğrusal olmayan problemi doğrusala çevirerek problemi çözmektir.

$$x \in R^n \rightarrow \Phi(x) = [\phi_1(x), \phi_2(x), \dots, \phi_n(x)] \in R^F \quad (4.28)$$

Oluşan bu fonksiyon sabit bir fonksiyon olup x değişkenlerinin $\Phi(x)$ ' e dönüşmesini sağlar. Girdi vektörü x, x_i gözlemlerden oluşur. Özellik uzayının oluşumunu sağlayan $\Phi(x)$ vektörü $\phi_i(x)$ ' lerden oluşmaktadır.



Kaynak: (Karakaynak, 2014: 34)

Şekil 4.12. Doğrusal Olmayan Veri Setinin Yüksek Boyutlu Uzayda Doğrusal Hiper Düzlem ile Sınıflandırılması

Kernel fonksiyonlarının tamamıyla yapılan çalışmalarda özellikleri ayırıcı bir doğrusallık vardır ve bunu gösterecek karar fonksiyonu aşağıdaki gibidir.

$$d(x) = \text{sign}(\sum_{i=1}^n y_i \alpha_i \Phi(x_i) \Phi(x) + b) = \text{sign}(\sum_{i=1}^n y_i \alpha_i K(x_i, x) + b) \quad (4.29)$$

oluşturulan karar fonksiyonunda,

$k(x_i, x)$ ifadesiyle hem doğrusal hem de doğrusal olmayan, özellik uzayında ise doğrusal sınıflandırıcıları elde eden çekirdek fonksiyonu vardır.

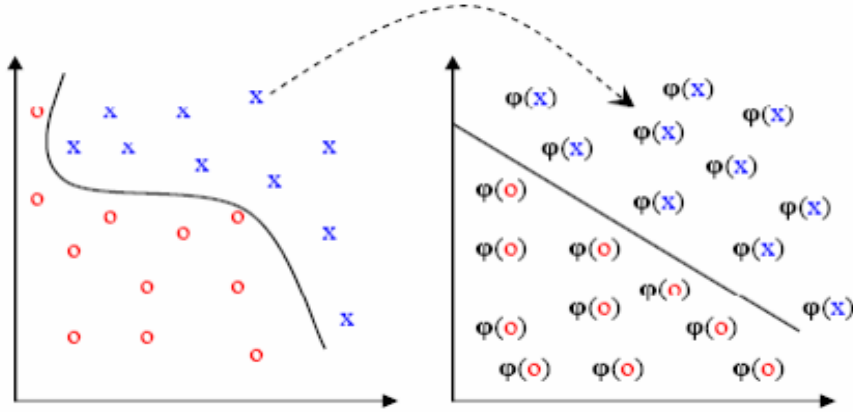
Karar fonksiyonunu yeniden düzenlemek gerekirse

$$\text{sign}(d(x)) = \text{sign}(\sum_{i=1}^n \alpha_i y_i K(x_i, x) + b) \text{ olur.}$$

Bu fonksiyonda kullanılan α_i destek vektörü sayısını ve x_i ise destek vektörlerini belirtmektedir.

4.5.1. Çekirdek Fonksiyonları (Kernel Functions)

Doğrusal olmayan DVM de sınıflandırma yapabilmek için girdi vektörünü olduğundan daha fazla yüksek boyutta yerleştirmek gereklidir. Bu işlemin sonucunda doğrusal olarak ayıramayan verilerin sınıflandırma işlemini yapmak için DVM' nin geçerli koşulları baz alınarak işlem gerçekleştirilir.



Kaynak: (Eray, 2008: 54)

Şekil 4.13. Girdi Uzayından Özellik Uzayına Dönüşüm

Şekil 4.13’de görüldüğü üzere iki boyutlu giriş uzayından, özellik uzayına eşleşmeyi görebiliriz. Sol grafikte verilen doğrusal olarak ayırlamadığını sağ yüksek boyutlu bir özellik uzayında taşındığında doğrusal olarak ayrılabilildiğini görülmektedir.

Kernel fonksiyonunda haritalama için $\phi R^n \rightarrow R^m$, $m > n$ olduğunu varsayarsak dönüşüm fonksiyonu aşağıdaki eşitlik gibi olur.

$$K(x, y) = \Phi(x)\Phi(y) \quad x, y \in R^n \quad (4.30)$$

Oluşturduğumuz bu fonksiyona kernel fonksiyonu veya çekirdek(ler) fonksiyon (ları)’u denir. Kernel nitelikler uzayında iç çarpımdan oluşan bir fonksiyondur. Genel tanımıyla kernel fonksiyonu aşağıdaki gibi ifade edilir.

$$K(x, y) = \Phi(x)\Phi(y) = (x, y)^2 \quad (4.31)$$

Oluşturulan kernel fonksiyonunda $x, y \in R^n$ dir aynı zamanda ikinci dereceden homojen polynominal kernel olarak tanımlanabilir.

Kernel fonksiyonunun birçok çeşitleri vardır. En çok kullanılan kernel fonksiyonlarını açıklamak gerekirse;

- Radyal tabanlı çekirdek(kernel) fonksiyonları

$$K(x_i, y_i) = \exp\left(\frac{-\|x_i - y_i\|}{2\sigma^2}\right) \quad i \quad (4.32)$$

- Doğrusal çekirdek (kernel) fonksiyonu

$$K(x_i, y_i) = (x'_i, x_j) \quad (4.33)$$

- Polinom çekirdek (kernel) fonksiyonu

$$K(x_i, y_i) = (x'_i, x_j)^d \quad (4.34)$$

Kernel fonksiyonları sınıflandırma problemlerinde ihtiyaca göre belirlenmektedir. Problemin zorluk derecesine göre de belirlenir. Yani doğrusal ayrılabilen bir problemde doğrusal kernel $K(x, y) = x \cdot y$ kullanılır ama doğrusal ayrılamayan ve daha karmaşık yapıya sahip olan bir probleme sahipsek gaussiyan (radyal tabanlı) kernel kullanabiliriz. Ya da ikinci dereceden polinomal kernel $K(x, y) = (x, y)^2$ kullanarak doğrusal ayrılamayan sınıfı yüksek boyutta doğrusal olarak ayrabiliriz. Kernel fonksiyonu nitelikler uzayında iç çarpımlardan oluşan bir fonksiyondur. Ancak her fonksiyon bu özelliğe sahip değildir. Bu fonksiyonun nitelikler uzayında iç çarpımların oluşması için mercer koşullarını sağlaması gereklidir.

Mercer koşulları

Oluşturulan fonksiyonun kernel fonksiyonu olabilmesi belirli şartları (koşulları) sağlaması gereklidir ve bu koşullara verilen ad mercer koşullarıdır.

X girdi uzayının fonksiyonu olan $K(x_i, x_j)$ kernel fonksiyonudur.

$$K(x, y) = \Phi(x) \cdot \Phi(y) \quad (4.35)$$

k' nın pozitif ve simetrik özellikleriyle tanımlı ise Mercer koşullarını sağladığı belirtilir.

Koşul 1:

$$K(x, y) = \sum_k^{\infty} \alpha_k \Phi_k(x) \cdot \Phi_k(y), \alpha_k \geq 0 \quad (4.36)$$

Koşul 2:

$$\iint K(x, y) g(x) g(y) d_x d_y > 0 \quad (4.37)$$

Koşulları sağlanıyorsa kernel fonksiyonu nitelikler uzayında iç çarpıma sahiptir diyebiliriz.

4.5.2. Kernel Fonksiyon Seçimi

DVM de sınıflandırma çalışması yapılırken veri setine hangi kernel türü uygulanacağına karar vermek çok önemlidir. Aksi takdirde eğitim verisine kernel fonksiyonlarının tümünü uygularsak aşırı uyum sorununa yol açabilir. Bu sorunların önüne geçebilmek için literatürü iyi araştırmak ve eski çalışmalardan hangi veri tipine hangi kernel uygulanmış araştırmak gereklidir.

4.5.3. DVM için Kerneller (DVM'ler de kullanılan kerneller)

4.5.3.1. Polinomal Kernel

$K(x, y) = (yx'.y + r)^d$ Genellikle doğrusal olmayan verilerde kullanılarak doğrusal olmayan veriler arasındaki ilişkiyi modelleyen yöntemdir. Denklemden $r > 0$ olduğunda homojen olmayan polinomal kernel olur. $r = 0$ olduğunda ise homojen polinomal kernel olarak tanımlanabilir. Polinomal kernel yönteminde aşırı uyum sorunuyla karşılaşmak polinomun derecesi yükseldikçe çok olası bir durumdur. Ivanciuc (2007:291-400) polinomal kernelde polinomun derecesi yükseldikçe aşırı uyum söz konusu olabilmektedir. Polinom derecesi yükseldikçe sınıflandırma probleminin karışıklığı daha da artmakta ve zor bir durum haline alır.

4.5.3.2. Doğrusal kernel

Ivanciuc (2007:291-400) doğrusal kernel hakkında doğrusal sınıflandırıcı olduğundan doğrusal olmama testi olarak kullanılır ve doğrusal olmayan eğitim setinden, doğrusal olmayan çekirdeklerden elde edilen sınıflandırma işlemleri için uygun bir özelliktir. $K(x, y) = x'.y$ doğrusal kerneli tanımlamak için iç çarpımların gerçekleşmesi gereklidir. İç çarpımların gerçekleşmesi için öncelikle Mercer koşulları gerçekleşmelidir. Doğrusallığı test etmede çözüm olarak kullanılan yöntem x ve y nin çarpımlarıyla adlandırılır.

4.5.3.3. Gaussiyan (radyal tabanlı) kernel

Gaussiyan kernel olarak adlandırılır. En çok tercih edilen yöntemlerden biri olup doğrusal olmayan veri setinde kullanılır. Gaussiyan kernel ayırma hiper düzleminin şekliyle doğrudan ilişkili olup DVM sayısının artıp ve azalmasına etki eder.

$$K(x, y) = e^{-\left(\frac{|x-y|^2}{2\sigma^2}\right)} \quad (4.38)$$

$$K(x, y) = e^{-\gamma|x-y|^2} \quad (4.39)$$

$$\gamma = \frac{1}{2\sigma^2} \text{ dir.}$$



5. LOJİSTİK REGRESYON

Lojistik regresyon analizi bağımlı değişkenin genellikle kategorik ve iki düzeyli olduğu durumlarda kullanılmaktadır. Regresyon analizlerinde bağımlı ve bağımsız değişkenler arasında ilişki olup olmadığını incelenmektedir. Doğrusal regresyon modellerine genellikle bağımlı değişken sürekli yapıda olmaktadır. Klasik doğrusal regresyon modellerinde aranan belli başlı varsayımlar mevcuttur. Bu varsayımlar normal dağılımın olması, sabit varyans, bağımlı ve bağımsız değişkenler arasında doğrusallık olmasıdır. Regresyon analizinde bağımlı değişken bazı durumlarda kategorik değer alabilmektedir. Bu durumlarda klasik doğrusal regresyon varsayımları geçerli olmadığından lojistik regresyon ile çalışma devam ettirilir. Lojistik regresyon sınıflandırma yöntemlerinden biridir. Lojistik regresyon da bağımlı değişken kategorik yapıda olduğu için bağımsız değişkenler de herhangi bir zorunluluk yoktur. Sürekli veya kategorik oluşu ile ilgili lojistik regresyon da bağımlı ve bağımsız değişkenler arasında doğrusal bir ilişkinin olması zorunlu aranmamaktadır. Kısacası lojistik regresyon girdi parametrelerini kategorik veriye (0 veya 1), (olumlu veya olumsuz) gibi kodlama yaparak çalışan bir makine öğrenme yöntemidir.

5.1. Logit Dönüşüm ve Lojistik Regresyon Modeli

Doğrusal olasılık modelleri 0 ile 1 arasında değer alabilen modellerdir. i . gözlem için genel doğrusal regresyon modeli, eşitlik (5.1)'deki gibi ifade edilmektedir.

$$y_i = \sum_{k=0}^p \beta_k x_{ik} + u_i \quad i = 1, 2, \dots, n \quad (5.1)$$

Doğrusal regresyon denkleminde bağımsız değişkenler üzerine bir kısıt yok iken bağımlı değişken sürekli olmalıdır. Dolayısı ile elde edilen modelde bağımlı değişken $-\infty$ ile

$+\infty$ arasında yer almalıdır. Bağımlı değişken y_i 0-1 gibi kategorik yapıda değer almasıyla dağılım bozulmaktadır. Bu durumda, i 'inci gözlemin 1 değerini alma olasılığı $P(y_i = 1)$ olmak üzere, beklenen değer

$$E(y_i) = 1 * P(y_i = 1) + 0 * P(y_i = 0) = P(y_i = 1) \quad i = 1, 2, \dots, n \quad (5.2)$$

elde edilir. Elde edilen beklenen değer regresyon denklemi olarak yazılırsa denklem (5.3) gibi olur.

$$E(y_i) = P(y_i = 1) + \sum_{k=0}^p \beta_k x_{ik} \quad i = 1, 2, \dots, n \quad (5.3)$$

arasında olasılık değerleri alabilen denklem (5.3)'teki denklem 'doğrusal olasılık modeli' olarak adlandırılır. Eşitlik (5.3) de gördüğümüz doğrusal olasılık modeline logit dönüşümü yapılabilmesi için öncelikli olarak $P / 1 - P$ olasılık dönüşümü yapılmalıdır. Sonrasında ise elde edilen modeli logaritması alınarak çıktı değişkeninin genişliği $-\infty$ ile $+\infty$ arasında yer almalıdır. Denklem (5.3) de logit dönüşüm yapıldıktan sonra oluşan lojistik model denklemi (5.4) de görülmektedir.

$$E(y_i) = Li = \log \left(\frac{P_i}{1-P_i} \right) \sum_{k=0}^p \beta_k x_{ik} \quad i = 1, 2, \dots, n \quad (5.4)$$

P_i olasılık değeridir ve

$$P_i = \frac{\exp(\sum_{k=0}^p \beta_k x_{ik})}{(1 + \exp(\sum_{k=0}^p \beta_k x_{ik}))} \quad i = 1, 2, \dots, n \quad (5.3)$$

şeklinde elde edilir. Elde edilen denkleme lojistik fonksiyon denilmektedir.

Lojistik regresyon modeline logit dönüşüm uygulanarak doğrusallaştırma yapmak mümkündür.

$$\eta = \ln \left(\frac{P(y_i=1|x)}{1-P_i} \right) \quad i = 1, 2, \dots, n \quad (5.4)$$

$P_i/(1 - P_i) =$ olasılık oranı

Logit dönüşümü uygulandıktan sonra lojistik regresyon modeli denklem (5.5) deki gibi görünmektedir.

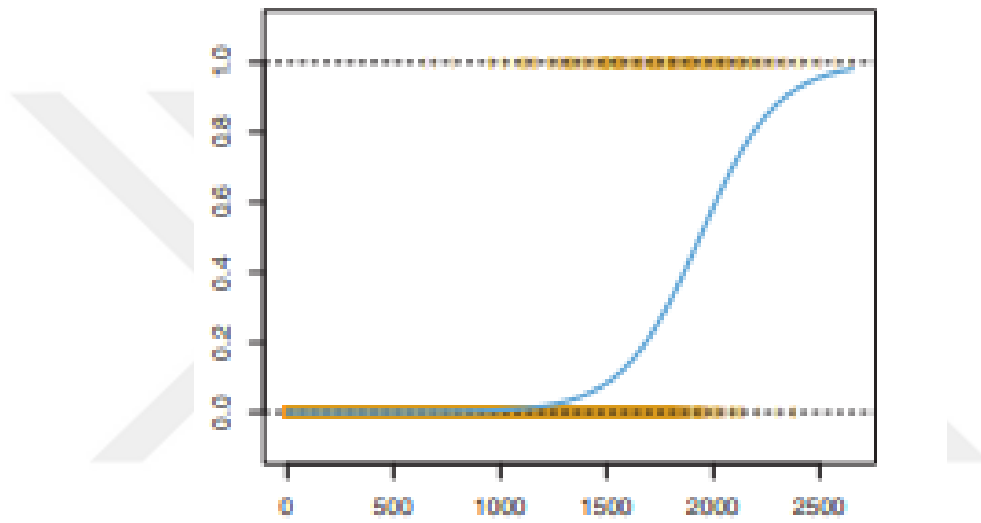
$$E(y_i) = \eta = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \dots + \beta_k x_{ki} \quad i = 1, 2, \dots, n \quad (5.5)$$

Lojistik regresyon modeline dayanarak hesaplanan P_i olasılık değeri 0.5 ten küçük ise

$\hat{y}_i = 0.5$, büyük ise $\hat{y}_i = 1$ şeklinde tanımlanabilir.

$$P_i = \frac{e^{\eta_i}}{1 + e^{\eta_i}} \quad i = 1, 2, \dots, n \quad (5.6)$$

Lojistik regresyon kendi içinde üçe ayrılmaktadır. Bu alt başlıklar şu şekildedir. Lojistik regresyon da bağımlı değişken iki kategori ise ikili (binary), lojistik regresyon da bağımlı değişken ikiden fazla olması durumunda ise çoklu kategorik (multinomial) lojistik regresyon modeli olarak adlandırılmaktadır. Lojistik regresyon da bağımlı değişkenin sıralı şekilde olması ise çoklu sıralayıcı (ordinal) lojistik regresyon olarak adlandırılmaktadır. Lojistik regresyonda grafik doğrusal regresyondan farklıdır. Klasik doğrusal regresyon da grafik doğrusal bir şekil alırken lojistik regresyon grafiği s şeklinde olmaktadır. Aşağıda şekil 5. 1.de lojistik regresyon grafiği görülmektedir.



Kaynak: (James vd., 2013: 131)

Şekil 5.1. Lojistik Regresyon Grafiği

6. DEĞERLENDİRME KRİTERLERİ

6.1. Karışıklık Matrisi

Karışıklık matrisi sınıflandırma çalışmalarında optimal çözümün belirlenmesinde ön planda yer almaktadır. Sınıflandırma çalışmalarında yararlanılan karışıklık matrisi ikili sınıflandırma çalışmalarında 2 x 2 boyutunda olurken, üçlü sınıflandırma çalışmalarında 3 x 3 boyutunda olmaktadır.

Tablo 6.1. Karışıklık Matrisi Oluşum Tablosu

	Gerçek Pozitif	Gerçek Negatif
Tahmin Pozitif	<i>TP</i>	<i>FP</i>
Tahmin Negatif	<i>FN</i>	<i>TN</i>

TP: Gerçek sınıf değeri pozitif olan gözlemlerin doğru bir şekilde pozitif olarak tahmin edilmesi durumunu göstermektedir

FP: Gerçek sınıf değeri negatif olan gözlemlerin doğru bir şekilde pozitif olarak tahmin edilmesi durumunu göstermektedir

TN: Gerçek sınıf değeri negatif olan gözlemlerin doğru bir şekilde negatif olarak tahmin edilmesi durumunu göstermektedir

FN: Gerçek sınıf değeri pozitif olan gözlemlerin tahmin sonucunda negatif olarak yanlış değerlendirildiğini göstermektedir.

Bu dört değer toplamı sınıflandırılacak olan toplam örnek sayısını verir. Karışıklık matrisi makine öğrenmesi sınıflandırma çalışmalarında kullanılan algoritmalarının kalitesinin ölçülmesi için kullanılan yöntemlerin hangisinin daha iyi sonuç verdiğini belirlemede kullanılmaktadır.

6.2. Doğruluk

Coşkun ve Baykal (2011:53) modelin başarılarının ölçümünde kullanılan en basit ve kullanımı kolay olan değerlendirme kriterlerinden biri olduğuna değinmiştir. Doğruluk oranı hesaplamak için karışıklık maddesinden yararlanılır.

$$Doğruluk = \frac{TP + TN}{TP + FP + FN + TN}$$

6.3. Duyarlılık (Hassasiyet)

Pozitif değerlerin ne kadarının pozitif sınıflandırmaya alındığını göstermektedir. Coşkun ve Baykal (2011:54) doğru sınıflandırılmış pozitif örnek sayısının toplam pozitif örnek sayısının sayısına oranı olduğunu belirtmektedir.

$$Duyarlılık = \frac{TP}{TP + FN}$$

6.4. Özgüllük

Sınıflandırmada negatif örnek sayısının toplam negatif örnek sayısına oranı şeklinde hesaplanmaktadır.

$$Özgüllük = \frac{TN}{TN + FP}$$

6.5. Kesinlik

Kesinlik kriteri ile modelde sıfır olarak eklenen grubun tahmin edilen örneklerden gerçekten kaç tanesinin pozitif olduğunu saptanmaktadır.

$$Kesinlik = \frac{TP}{TP + FP}$$

6.6. ROC Eğrileri

ROC eğrisi pozitif yanlış oran ve testlerin duyarlılığından oluşan bir grafikdir. ROC eğrisi genellikle farklı çeşitlerde sınıflandırma problemi için performans değerlendirme biçimidir. ROC eğrisinde bulunan AUC değerleri genel doğruluk değerlerini belirler. ROC eğrisinde kullanılan modellerin ne kadar doğru sınıflandırma yapabildiğini göstermektedir. ROC eğrisinde kullanılan yöntemleri değerlendirmek istendiğinde modelde kullanılan yöntemlerin AUC değerleri 1'e ne kadar yakın veya 1 ise bu değerler en uygun modeli belirlemede yardımcı olur.

6.6.1. ROC Eğrileri Değerlendirilmesi

İçerisinde mükemmel doğruyu elde etmek gerçek hayat verilerine göre oldukça imkânsız bir durum denilebilir. Kumar ve Indrayon (2011: 279) mükemmel doğru için koordinat sisteminde (0,0) başlangıç noktası ile sol üst köşe noktalarının birleşmesiyle aynı zamanda sağ üst köşe noktası ve sol üst noktalarının birleşiminden mükemmel bir şekilde çıkmaktadır. Bu çıkan %100 veya mükemmel sonucu temsil eder. Ancak böyle eğrileri elde etmek oldukça zordur denilebilir. Yapılan analizlerde testin kesinliği veya doğrunun ölçmek için kullanılan kısımlardan bir tanesinde ROC eğrisinin alt tarafında kalan kısımdır. ROC eğrisinin altında kalan kısmın alabileceği değerler aralığı 0 ile 1 arasındadır. ROC eğrisinin altında kalan kısım 0,5 tir. Mükemmel doğruya sahip olan ROC eğrisinin değeri ise 1,0 dir. Yani bu değerlerden anlaşılacağı üzere yapılan testlerin Roc eğrisinde 0,5 ile 1,0 arasında olması gözlenir 0.5 ile 1,0 arasında oluşan eğrilerin yeri büyüdükçe yani 1,0'a yaklaştıkça testlerin kesinliğinde artış meydana gelir.

6.6.2. ROC Eğrileri Avantajları

Kumar ve Indrayon (2011: 280) ROC eğrisinin avantajlarına şu şekilde değinmişlerdir.

1. Olası tüm sonuçları gösterme imkânına sahiptir
2. ROC eğrisi sorunun prevalansından etkilenmez.
3. İki veya ikiden fazla testin ROC eğrisine göre bir grafikte görselleştirilebilir ve değerlendirmeye kolaylık sağlar

4. Bazı durumlarda duyarlılık ön plana çıkabilmektedir veya seçicilik duyarlılığa tercih edilebilir. Bu iki durumla karşılaşıldığında dahi ROC eğrileri uygun kesim noktaları bulur.



7. UYGULAMA

Tez çalışmasının bu bölümünde makine öğrenmesi uygulama alanlarından dış ticaret birim değer endeks verileri üzerinde makine öğrenmesinde DVM ve LR analizi yöntemleri kullanılarak sınıflandırma çalışması yapılmış ve elde edilen sonuçlar değerlendirilmiştir. Çalışma kapsamında TÜİK resmi sitesinden <https://www.tuik.gov.tr/> alınan veriler istatistiksel modelleme, veri görselleştirme gibi etkili bir biçimde kullanılması ve işlem görmesini sağlayan R stüdyo programı kullanılmıştır.

7.1. R Programında e1071 Paketinde *svm* fonksiyonu

svm fonksiyonu destek vektörlerini eğitmek için kullanılan bir fonksiyon olup sınıflandırma ve regresyon çalışmalarında sıklıkla kullanılmaktadır. *svm* fonksiyonu tek değişkenli ve çok değişkenli çalışmaların kullanımına uygundur. *e1071* paketinde uygulamalarda yaygın olarak kullanılan çekirdek isimleri şu şekildedir. *radial kernel*, *doğrusal kernel*, *polinom kernel* en yaygın ve başarılı olan kernel türleridir.

7.2. R Programında stats Paketinde *glm* fonksiyonu

Makine öğrenmesinde sınıflandırma işlemlerinde DVM hangi fonksiyonun kullandığımıza değindik. LR' nin R stüdyo programında uygulaması için *glm* fonksiyonu kullanılır. *glm* genelleştirilmiş lineer modelin kısaltılmış halidir. *glm* fonksiyonu sıradan doğrusal regresyon bir uzantısı şeklindedir. Bağımlı değişken kategorik olduğu durumlarda lojistik regresyon uygulamalarında kullanılmaktadır. *glm* fonksiyonu içerisinde "*binomial*", "*bernoulli*", "*poisson*" gibi geniş kapsamlı dağılım özellikleri bulunan aileler vardır. Bu çalışmamızda *binomial* dağılım kullanılmaktadır.

7.3. Problemin Belirlenmesi

Ülkemizde uluslararası ticaretin gelişmesi ile ithalat ve ihracat oranlarında artış gözlenmektedir. Oluşan bu ithalat ve ihracatını standart ticaret sınıflandırmalarına göre belirlenmesi gereklidir. Bu tez çalışması kapsamında amaç ithalat ve ihracat değerleri kapsamında doğru tahmin yapabilmek için makine öğrenmesi sınıflandırma yöntemlerini kullanarak en iyi sonucu veren yöntemi belirlenmesidir.

7.4. Verilerin Toplanması ve Hazırlanması

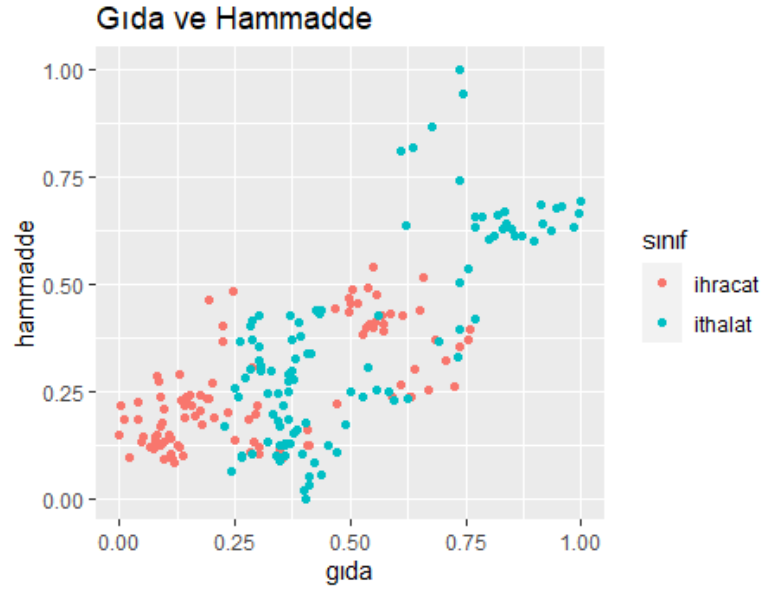
Tez çalışması kapsamında 2013 yılı ocak ayından başlayan ve 2021 yılı Ağustos ayına kadar olan sürede dış ticaret birim değer endeks verileri TÜİK resmi sitesinden alınmıştır <https://www.tuik.gov.tr/>. Veri seti içerisinde eksik veri gözlem bulunmamaktadır. Veri seti içerisinde bağımlı değişken kategorik yapıda bulunmaktadır. Araştırma kapsamında toplanan veri seti içerisindeki özellikler tablo 7.1. 'de detaylı olarak belirtilmiştir.

Tablo 7.1. Çalışma Kapsamında Değişkenler ve Açıklamaları

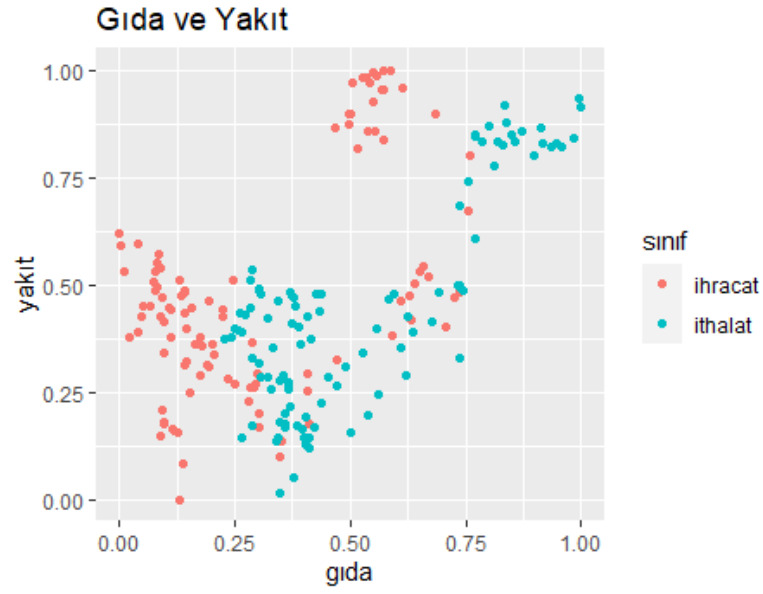
Bağımlı Değişken:	Açıklamalar
Sınıf: Dış ticaret türü	0= İhracat 1= İthalat
Bağımsız Değişkenler:	
Gıda	Gıda, içecek ve tütün
Hammadde	Ham maddeler yakıt hariç
Yakıt	Yakıtlar
İmalat	İmalat (gıda, içecek ve tütün hariç)
Sıtc	SITC'de başka bir yerde sınıflandırılmayan mallar

7.5. DVM Kurulumu

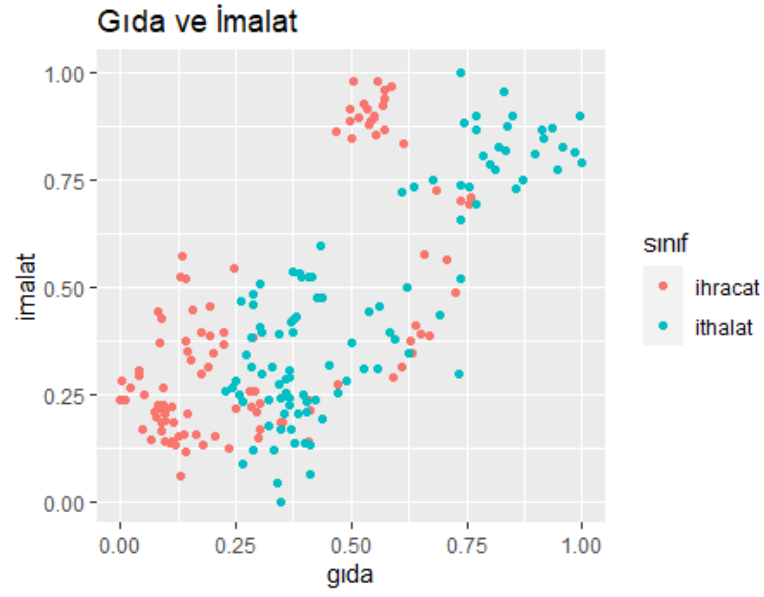
Bu kısımda ileriki çalışmalar için uygulayacağımız radyal ve doğrusal kernel için gerekli çalışma yer alacaktır. Öncelikle Verinin her öz niteliği için ana ekran uzunluklarını ve küçük eksen uzunluklarının görselleşmiş hallerini aşağıdaki şekillerde görebiliriz.



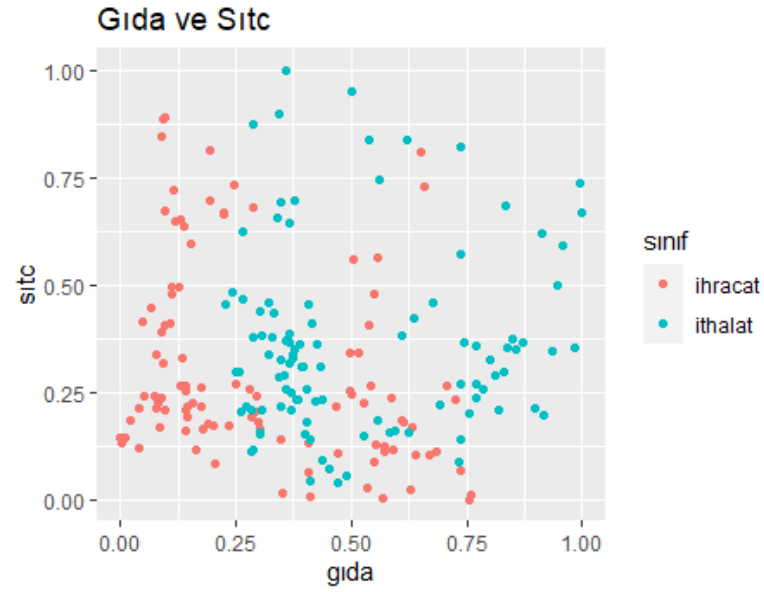
Şekil 7.1. Gıda ve Hammaddeye Göre Dış Ticaret Sınıflandırılması



Şekil 7.2. Gıda ve Yakıtı Göre Dış Ticaret Sınıflandırılması

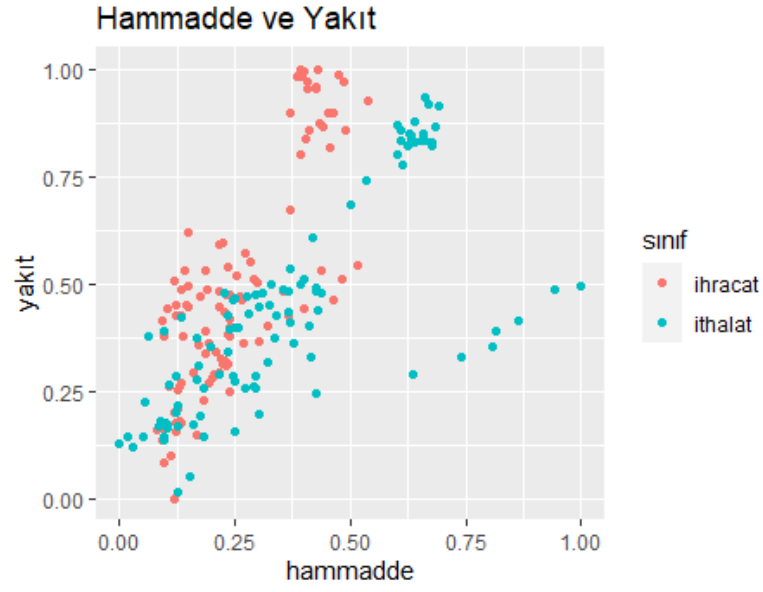


Şekil 7.3. Gıda ve İmalata Göre Dış Ticaret Sınıflandırılması

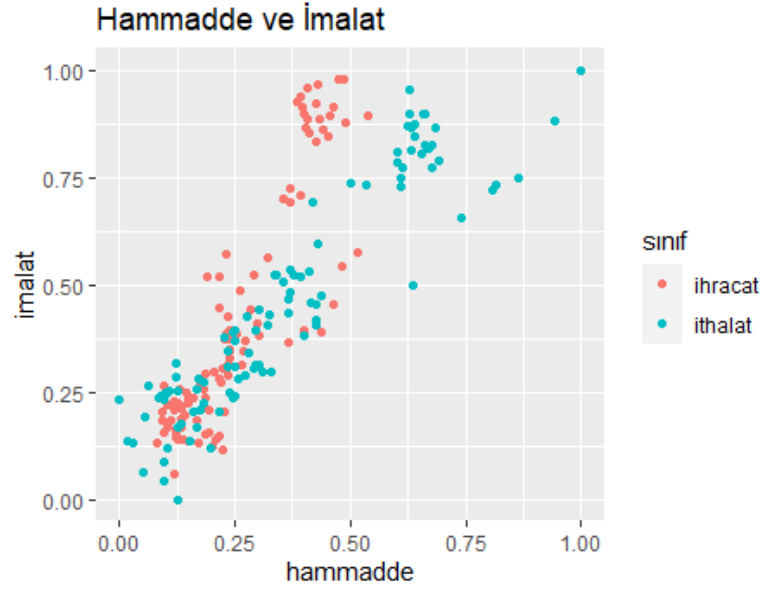


Şekil 7.4. Gıda ve Sıtc'a Göre Dış Ticaret Sınıflandırılması

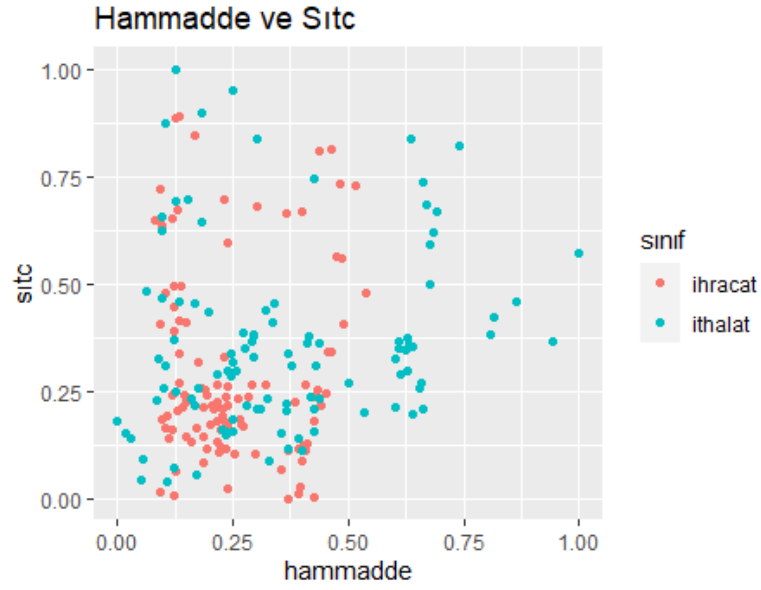
Grafiklerden anlaşılacağı üzere gıda değişkeninin diğer tüm değişkenlerle birlikte iç içe geçmiş bir şekilde sınıflandırıldığı görülmektedir. Şimdi hammadde değişkeninin diğer değişkenlerle kombinasyonunu inceleyelim.



Şekil 7.5. Hammadde ve Yakıtı Göre Dış Ticaret Sınıflandırılması

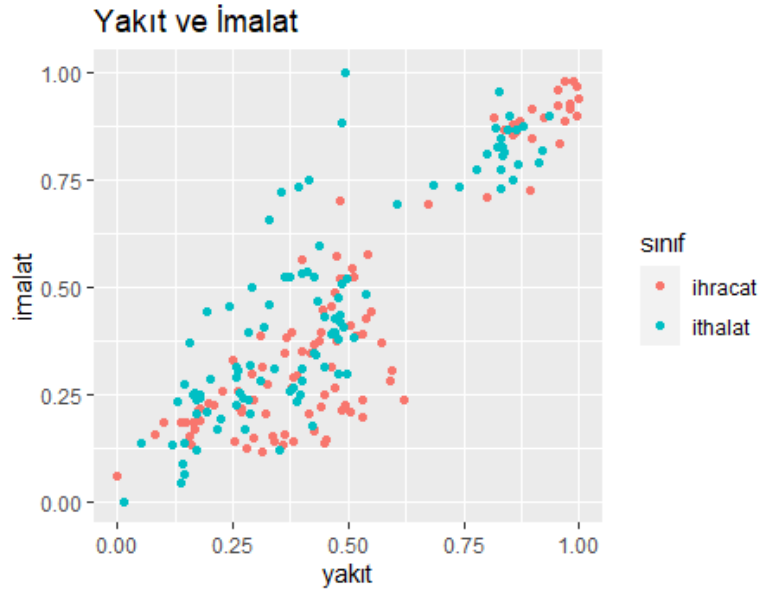


Şekil 7.6. Hammadde ve İmalata Göre Dış Ticaret Sınıflandırılması

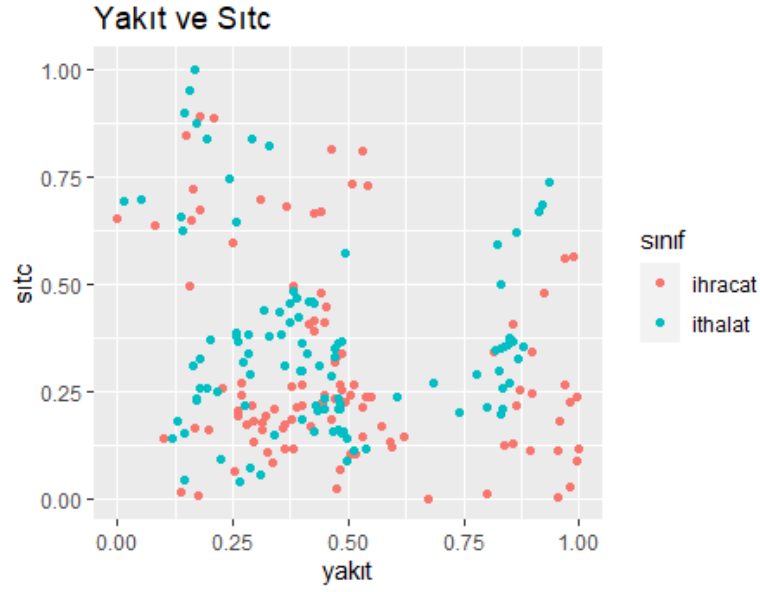


Şekil 7.7. Hammadde ve Sıtc'a Göre Dış Ticaret Sınıflandırılması

Grafiklerden anlaşılacağı üzere hammadde değişkeninin diğer değişkenlerle olan kombinasyonlarına bakıldığında sınıflandırmanın güçsüz olduğu görülmektedir. Şimdi yakıt değişkeninin diğer değişkenlerle kombinasyonunu inceleyelim.



Şekil 7.8. Yakıt ve İmalata Göre Dış Ticaret Sınıflandırılması



Şekil 7.9. Yakıt ve Sıtc'a Göre Dış Ticaret Sınıflandırılması

Verinin tam haliyle grafikler incelendiğine göre dış ticaret grubuyla tam sınıflandırma mevcut değildir. Makine öğrenmesi yöntemleri kullanarak veri seti eğitim ve test seti olarak öğretildikten sonra sınıflandırma çalışması yapılacaktır.

7.6. Eğitim ve Test Verisinin Ayrılması

Veri seti analizini hazırlandıktan sonra makine öğrenmesinde sınıflandırma yöntemleri uygulanabilmesi için eğitim ve test olarak ikiye ayrılması gereklidir. Bu çalışma kapsamında veri rassal olarak %50 eğitim ve %50 test veri seti olarak ayrılmıştır.

7.7. Modelinin Oluşturulması

Çalışma içerisinde belirtilen makine öğrenmesi sınıflandırma tekniklerinden olan DVM ve LR yöntemine uygun modeller R programlama dili kullanılarak oluşturulmuştur. Çalışmada eğitim veri seti kullanılarak DVM' den olan *doğrusal kernel*, *radyal kernel* ve *LR* kullanılmıştır. Model eğitim verisi kullanılarak oluşturulmuş ve test veri seti üzerinde tahmin yapılmıştır. Doğruluk, duyarlılık, özgüllük ve kesinlik gibi değerlendirme kriterlerinin sonuçlarını tablo 7.2 de görebiliriz.

Tablo 7.2. Sınıflandırma Yöntemlerinin Karşılaştırılması

	Doğruluk (Accuracy)	Duyarlılık (Sensitivity)	Özgüllük (Specificity)	Kesinlik (Pos Pred Value)
Doğrusal kernel	0.682	0.727	0.650	0.603
Radyal kernel	0.807	0.840	0.783	0.740
Lojistik regresyon	0.625	0.750	0.533	0.541

Tablo 7.2’de görüldüğü üzere sınıflandırma işlemi gerçekleştirilmiş ve değerlendirme kriterlerine göre radyal kernel 0.807 doğruluk, 0.840 duyarlılık, 0.783 özgüllük ve 0.740 kesinlik değeriyle değeri ile ithalat ve ihracata göre en doğru sınıflandırmayı yapan metot dur. Doğrusal kernel İthalat ve ihracata göre sınıflandırma işleminde ikinci sırada başarıyla sınıflandırıcı olmuştur. LR ise değerlendirme kriterlerine göre üçüncü başarıyla sınıflandırmayı gerçekleştirmiştir.

7.8. Karışıklık Matrisi Sonuçları

Karışıklık matrisi yapılan çalışmada sınıflandırma oranının doğruluğunu göstermektedir. Verideki gözlemlerin kaç tanesinin yanlış kaç tanesinin doğru sınıflandırıldığını belirtmektedir. Yapılan çalışma eğitim verisi kullanılarak oluşturulmuş ve test veri seti üzerinde tahmin edilmiştir DVM’ den doğrusal, radyal kernel ve LR’ a göre sonuçlar oluşmuştur. Aşağıdaki Tablo 7.3.’te karışıklık matrisi sonuçları görülmektedir.

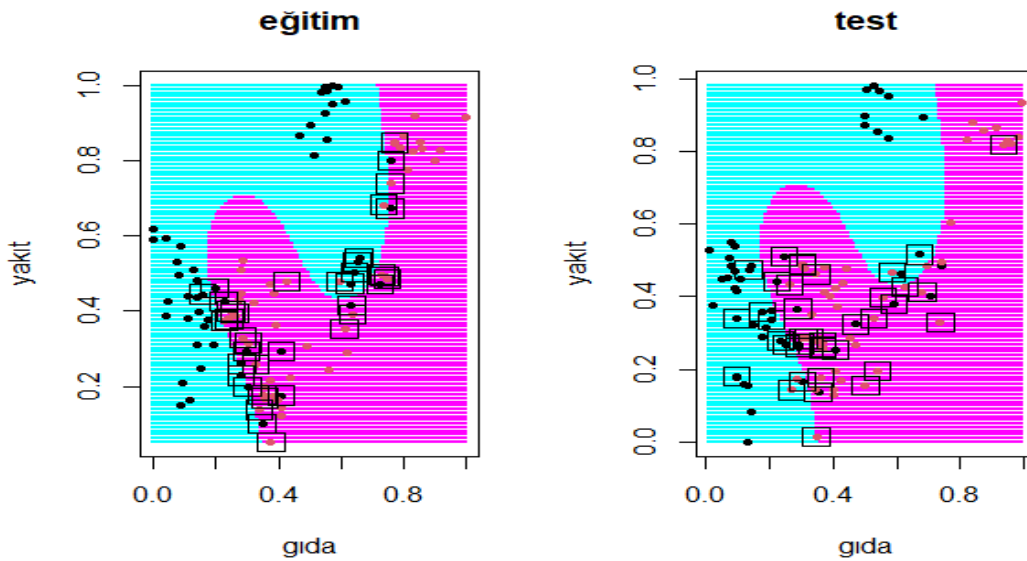
Tablo 7.3. Karışıklık Matrisi Sonuçları

Doğrusal Kernel			Radyal kernel			Lojistik regresyon		
Tahmin	İhracat	İthalat	Tahmin	İhracat	İthalat	Tahmin	İhracat	İthalat
İhracat	32	23	İhracat	37	13	İhracat	33	28
İthalat	12	37	İthalat	7	47	İthalat	11	32

Tablo 7.3'te görüldüğü üzere doğrusal kernel 104 gözleminden 69 tanesini doğru, 35 tanesini yanlış sınıflandırmıştır. Radyal kernel karışıklık matrisi sonuçlarına bakıldığında zaman 104 gözleminden 84 tanesini doğru, 20 tane gözlemi yanlış sınıflandırmıştır. Lojistik regresyonun karışıklık matrisindeki performansı ise 65 gözlemi doğru sınıflandırarak 39 gözlemi yanlış sınıflandırmıştır. Sonuçlardan anlaşılabileceği üzere en iyi performansı veren model 84 gözlemi doğru sınıflandırarak radyal kernel olmuştur.

7.9. Eğitim ve Test Veri Seti Destek Vektörlerinin Belirlenmesi

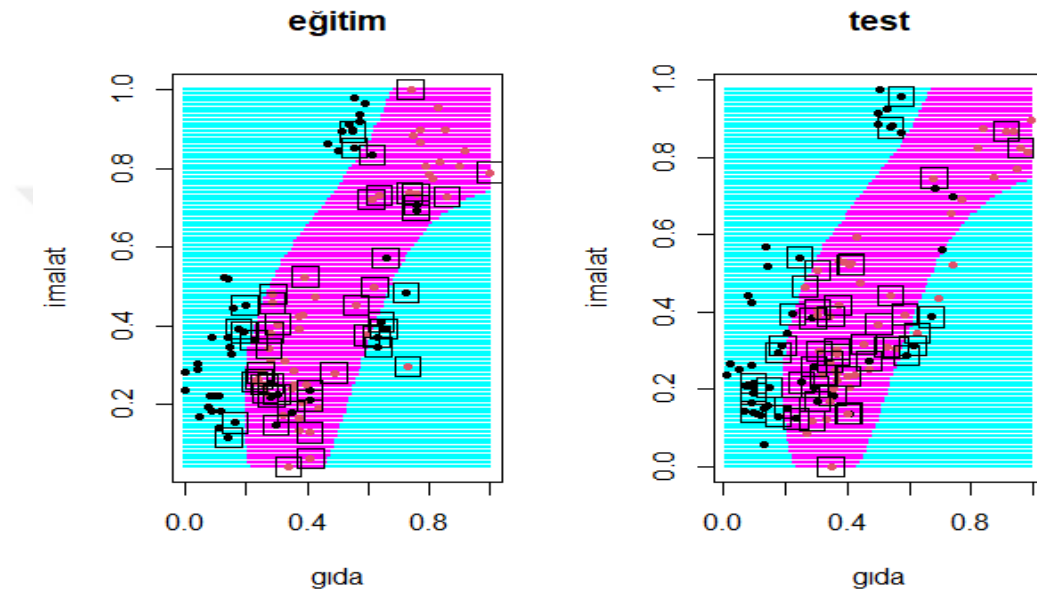
Verilerin %50 eğitim ve test olarak bölünmesiyle gıda ve yakıt değişkenleriyle eğitim ve test veri setinin radyal tabanlı sınıflandırma grafiği çıkarıldı. Oluşturulan grafikte eğitim veri setiyle sınıflandırmadaki destek vektörleri ve hiperplan çizgisi (ayrılma) belirgin bir şekilde görülmektedir. Test veri setinin grafiği incelendiğinde ise eğitim verisinde olduğu gibi sınıflandırmayı doğru bir şekilde yaptığı görülmektedir yani eğitilen model ayrılan veri seti yani test seti ile test edilmiş olup sınıflandırmayı başarılı bir şekilde gerçekleştirmiştir.



Şekil 7.10. Gıda ve Yakıt Değişkenine Göre Eğitim ve Test Sınıflandırma Grafikleri

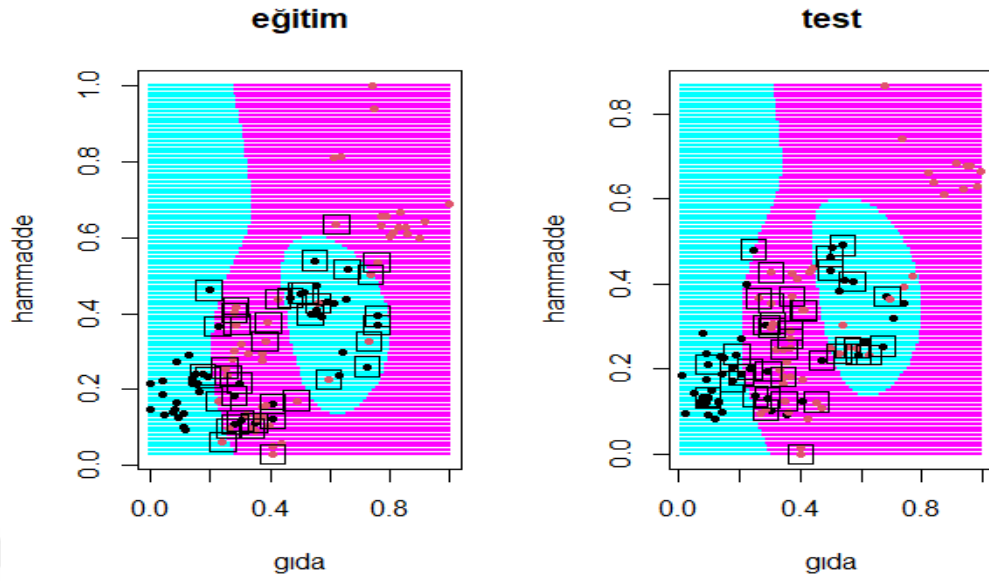
Şekil 7.10'da görüldüğü üzere 1'inci şekil eğitim verilerinin şeklidir ve modelin eğitim gözlemlerini sınıflandırmada ne kadar başarılı olduğunu kontrol edebilmemizi sağlar. Eğitim verisiyle model analiz edilmiştir. Modelin tahmini için ayırdığımız test veri setiyle ne kadar doğru sınıflandırma yaptığı belirlenmiştir. Şekillerde siyah noktalar

ihracatı pembe noktalar ithalatı temsil etmektedir. Dış ticaret her ekonomik uygulama gibi ihtiyaçları karşılamak, fayda sağlamak veya gelir elde etmek için yapılan bir ekonomik faaliyettir. 1. ve 2. Şekillerde ithalat ve ihracat sınıflandırması sağlanmış ancak gıda ve yakıt değişkeni verileri sayısal olarak birbirine çok yakın düzeyde olduğundan ithalat ve ihracat değişkenleri bir arada görülmektedir. Yani Türkiye'nin ithalat ve ihracat verilerine göre gıda ve yakıt değerleri birbirine çok yakındır.



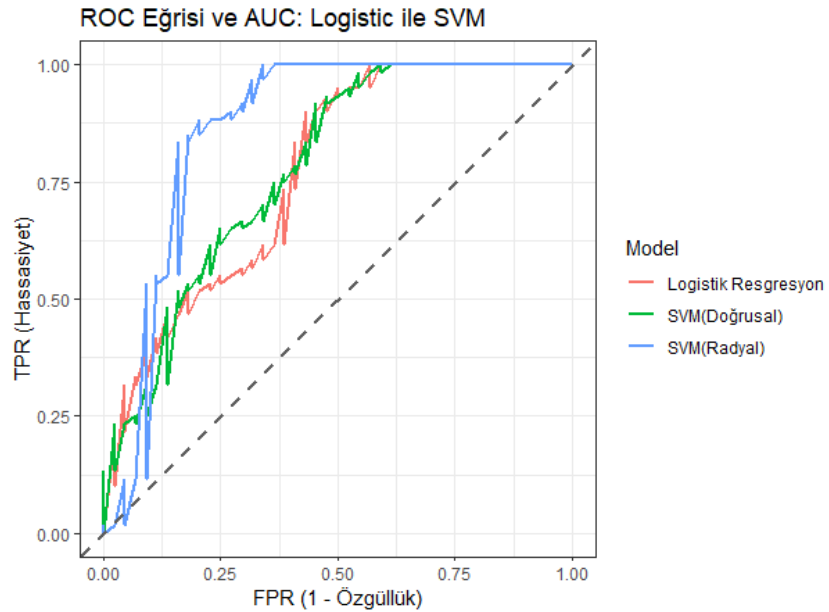
Şekil 7.11. Gıda ve İmalat Değişkenine Göre Eğitim ve Test Sınıflandırma Grafikleri

Şekil 7.11’de görüldüğü üzere 1. ve 2. Şekillerde ithalat ve ihracat sınıflandırması sağlanmıştır ancak Türkiye’nin dış ticaret verilerine göre gıda ve imalat sektöründe ihracat ve ithalat değerleri arasında çok az fark vardır bu yüzden şekillerde destek vektörleri bir arada görünmektedir.



Şekil 7.12. Gıda ve Hammadde Değişkenine Göre Eğitim ve Test Sınıflandırma Grafikleri

Şekil 7.12’de görüldüğü üzere Türkiye’nin dış ticaret verilerine göre gıda ve hammadde değişkenleri arasında gözle görülür düzeyde bir sınıflandırma mevcuttur. Ancak gıda ve hammadde değerleri birbirine yakın olduğunda destek vektörleri bir arada görünmektedir.



Şekil 7.13. ROC Eğrisi

ROC (İşlem Karakteristik Eğrisi) eğrisinde X ekseninde FPR(1-Özgüllük) yanlış pozitif oranı göstermektedir. Y ekseninde bulunan TPR (hassasiyet) gerçek pozitif oranı göstermektedir. ROC eğrisi yorumu yapılırken AUC (ROC eğrisinin altında kalan alan) değerleri dikkate alınarak en iyi sınıflandırıcı model seçilir. Aşağıda Tablo 7. 4.'te çalışmada kullanılan modellerin AUC değerleri görülmektedir.

Tablo 7.4. AUC Değerleri

Lineer kernel	Radyal kernel	Lojistik regresyon
0.780	0.864	0.770

Tablo 7. 4.'te AUC (ROC eğrisinin altında kalan alan) değerleri görülmektedir. AUC ne kadar büyük ise, doğru sınıflandırmanın tahmin edilmesinde söz konusu o metot başarılıdır. AUC 0.5 ile 1 sınırları arasında değer almaktadır. Bu çalışmada kullanılan modellerden en başarılı sonucu veren AUC değeri 0.864 ile radyal kerneldir. Doğrusal kernel 0.780 AUC değeriyle ikinci sıradadır. Lojistik regresyon ise 0.770 değer ile üçüncü sırada yer alır. Sonuç olarak ROC eğrisinin altında kalan alan belirlendi ve en başarılı model radyal kernel olarak belirlenir.

8. SONUÇ VE YORUM

Bu tez kapsamında ilk aşamada makine öğrenmesi yöntemlerinden destek vektör makineleri ve lojistik regresyon metotları kullanılarak sınıflandırma çalışması yapıldı. Bu kapsamda, doğrusal kernel, radyal kernel ve lojistik regresyon sonuçları edilmiştir. Çalışma kapsamında 208 gözlemlilik dış ticaret verisi kullanılmıştır. Veriler rassal bir şekilde %50 eğitim ve %50 test olarak ikiye bölünmüştür. Model eğitim verisi kullanılarak oluşturulmuş ve test veri seti üzerinde tahmin yapılmıştır. Doğruluk, duyarlılık, özgüllük ve kesinlik gibi değerlendirme kriterlerinin sonuçlarına göre en iyi sonucu veren radyal kerneldir. Değerlendireme kriterlerinden olan ROC eğrisi ve AUC değerlerine göre en iyi sınıflandırıcı 0.86 oranıyla radyal kernel seçilmiştir.

Bu çalışmada DVM ve lojistik regresyon metotları kullanılmıştır. Ayrıca, sınıflandırma yapmak için diskriminant analizi ve en yakın komşular yöntemi gibi diğer yöntemler uygulanarak literatüre katkı sağlanabilir.

KAYNAKÇA

- Akgün, Barış, “Apache Spark Tabanlı Destek Vektör Makineleri İle Akan Büyük Veri Sınıflandırma”, *Fen Bilimleri Enstitüsü Yüksek Lisans Tezi*, 2016, ss. 1-44.
- Bulut, Sinan, Alkan Günlü, " Arazi Kullanım Sınıfları İçin Farklı Kontrollü Sınıflandırma Algoritmalarının Karşılaştırılması ", *Kastamonu Üniversitesi, Orman Fakültesi Dergisi*, 2016/16 (2), ss. 528-535.
- CORTES, CORINNA, VLADIMIR, VAPNIK, “Support-Vector Networks”, *Machine Learning*, 20/1995, ss. 273-297.
- Coşkun, Cengiz, Abdullah, Baykal., " Veri Madenciliğinde Sınıflandırma Algoritmalarının Bir Örnek Üzerinde Karşılaştırılması", *İnönü Üniversitesi Akademik Bilişim’11*, 2/2011 (1), ss. 51-58.
- DOĞAN, Seyyide, “Optimal Parametre ve Özellik Seçimi İle Destek Vektör Makinesi Kullanılarak Finansal Başarısızlık Tahmini”, *Gazi Üniversitesi Sosyal Bilimler Enstitüsü Doktora Tezi*, 2020, ss. 1-172.
- Ekmekçi, H., “Türkiye İşgücü Yapısının Özelliklerinin İncelenmesi: Lojistik Regresyon Ve Karar Ağacı Uygulaması”, *Hacettepe Üniversitesi Yüksek Lisans Tezi*, 2019, ss.192.
- Eray, Osman, “Destek Vektör Makineleri İle Ses Tanıma Uygulaması”, *Fen Bilimleri Enstitüsü Yüksek Lisans Tezi*, 2008, ss. 1-85.
- Güneren, Hilal, “Destek Vektör Makineleri Kullanarak Gömülü Sistem Üzerinde Yüz Tanıma Uygulaması”, *Fen Bilimleri Enstitüsü Yüksek Lisans Tezi*, 2015, ss. 1-44.
- Güran, A., M. Uysal, Ö. Doğrusöz, “Destek vektör makineleri parametre optimizasyonunun duygu analizi üzerindeki etkisi”, *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, 2014/16, (48), ss.86-93

- Ivanciuc, O., (2007), "Applications of Support Vector Machines in Chemistry. In: Reviews in Computational Chemistry", *Volume 23*, ed. K. B. Lipkowitz and T. R. Cundari. Wiley-VCH, Weinheim, 291–400
- James, Gareth, vd., (2013) An Introduction to Statistical Learning with Applications in R, (1. baskı), *Springer Yayınları*, New York.
- Kara, İ, Ecer, F., “Bist Endeks Hareket Yönünün Tahmininde Sınıflandırma Yöntemlerinin Performanslarının Karşılaştırılması”, *Akademik Sosyal Araştırmalar Dergisi*, 2018/6 (83), ss. 185-195.
- Karagül, K., “İstanbul Menkul Kıymetler Borsası'nda İşlem Gören Firmaların Destek Vektör Makineleri Kullanılarak Sınıflandırılması”, *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 20 /2014(5), ss. 174-178.
- Kızılkaya, Y. M., A., Oğuzlar, “Bazı Denetimli Öğrenme Algoritmalarının R Programlama Dili İle Kıyaslanması”. *Karadeniz Uluslararası Bilimsel Dergi*, 37/2018, ss. 90-98.
- Kumar, R., Indrayan A., “Receiver Operating Characteristics (ROC) Curve for Medical Researchers”, *Indian Pediatrics*, V.48/2011, ss.277-287.
- Niu, Xiao, Ching Y, Suen, “A novel hybrid CNN–SVM classifier for recognizing handwritten digits”, *Pattern Recognition*, 45/2012, (4), ss.1318-1325.
- Tayyar, Nezi, Tekin, Selin., “İMKB-100 Endeksinin Destek Vektör Makineleri İle Günlük, Haftalık ve Aylık Veriler Kullanarak Tahmin Edilmesi”, *Abant İzzet Baysal Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 2013/13 (1), ss.189-218.
- Wang, L., (2005), Support Vector Machines: Theory and Applications, *Springer, Netherlands*.
- Yakut, E., B., Elmas, S., Yavuz, “Yapay Sinir Ağları Ve Destek Vektör Makineleri Yöntemleriyle Borsa Endeksi Tahmini”, *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 2014/19 (1), ss.139-157.
- TÜİK, (2021) <https://www.tuik.gov.tr/>, (Erişim tarihi: 29.10.2021).

EKLER

Ek 1.Çalışmada Kullanılan Veri Seti

sınıf	gıda	hammadde	yakıt	imalat	sıtc
0	97,2	117,7	177,0	116,9	135,9
0	99,3	116,9	179,3	117,0	136,2
0	99,0	121,4	171,0	114,7	129,3
0	98,6	118,0	161,8	114,3	123,3
0	97,7	115,5	156,4	114,8	117,9
0	96,9	116,2	167,5	115,3	117,8
0	95,7	114,7	163,1	113,9	107,4
0	97,0	114,0	164,0	114,5	110,4
0	97,1	115,4	167,2	113,5	109,8
0	99,8	112,1	159,3	114,0	99,8
0	99,2	112,5	162,1	113,6	100,0
0	99,8	113,6	174,7	115,4	89,7
0	99,9	112,2	174,8	116,4	98,8
0	98,2	110,6	178,6	115,6	108,2
0	98,7	112,3	176,8	114,4	111,5
0	99,0	111,7	180,3	114,8	96,7
0	98,4	111,6	178,4	115,2	91,8
0	99,9	111,2	180,9	115,8	99,1
0	100,5	113,9	180,4	116,6	109,3
0	101,5	113,5	175,6	113,1	104,5
0	104,3	109,7	167,1	110,2	98,8
0	107,2	111,3	154,6	109,8	90,4
0	107,1	109,7	137,6	109,4	89,6
0	106,4	108,6	112,5	109,6	95,2
0	105,1	106,3	101,5	106,0	111,5
0	105,9	102,2	110,9	103,9	109,0
0	102,3	100,6	103,6	100,3	103,5
0	101,4	102,6	109,6	99,4	104,9
0	102,6	104,9	115,2	102,0	98,3
0	103,7	101,7	116,9	101,4	98,3
0	102,1	100,6	111,1	101,0	91,3
0	100,6	100,4	98,9	98,7	99,2
0	95,9	99,4	91,4	98,4	98,5
0	93,5	95,4	87,3	97,4	100,6
0	93,4	92,9	82,1	94,8	94,7
0	93,5	92,8	71,6	96,7	90,2
0	91,3	90,7	66,6	96,0	90,7
0	91,1	92,1	61,8	96,0	101,3

0	89,4	91,4	70,4	95,6	103,2
0	89,3	92,4	74,8	97,1	102,8
0	88,5	91,7	82,9	97,0	105,6
0	88,9	93,4	82,8	98,0	106,4
0	88,4	97,0	78,7	97,9	110,8
0	87,3	93,7	83,9	96,9	111,7
0	89,0	97,7	84,1	96,6	109,6
0	89,2	99,3	87,2	95,1	104,4
0	86,6	98,1	85,4	94,4	103,8
0	85,5	97,2	93,1	95,2	96,5
0	83,9	97,6	96,2	95,2	99,2
0	84,4	96,1	95,9	94,6	103,2
0	83,1	99,7	89,7	94,2	103,0
0	81,3	98,7	93,5	94,9	106,8
0	83,2	99,9	91,1	96,5	105,5
0	84,4	98,4	86,7	99,0	107,5
0	84,9	100,4	90,1	99,4	104,0
0	85,4	102,9	96,4	100,3	104,0
0	84,4	100,8	98,5	101,5	111,1
0	83,2	100,7	101,2	100,3	107,6
0	83,0	100,0	106,1	101,0	106,8
0	83,6	99,1	107,3	102,9	108,1
0	83,0	97,3	112,7	104,8	110,7
0	82,7	100,1	111,3	106,2	117,0
0	83,0	99,3	112,3	104,8	111,7
0	82,5	104,2	116,1	104,9	111,7
0	80,9	100,5	119,9	102,4	109,1
0	80,6	103,9	121,2	102,8	109,3
0	80,8	103,0	124,3	100,9	103,4
0	79,1	99,8	127,3	99,2	99,4
0	77,5	99,2	126,9	98,6	100,4
0	77,5	94,6	130,5	97,4	101,5
0	77,9	97,1	118,7	97,4	101,4
0	78,4	90,9	98,2	98,2	104,8
0	79,1	97,1	99,8	98,8	107,4
0	79,5	94,2	107,8	97,7	109,6
0	80,6	94,6	114,0	97,1	108,3
0	80,6	94,1	118,7	96,3	107,1
0	80,3	92,4	115,4	96,7	109,7
0	81,1	96,4	110,7	98,1	115,8
0	80,5	93,6	112,6	96,8	117,7
0	79,4	93,5	105,0	95,6	123,8
0	80,1	92,7	108,2	94,9	126,7
0	81,7	94,5	107,7	94,7	123,6

0	81,0	92,7	104,7	95,5	122,0
0	81,3	90,6	103,4	96,6	123,2
0	81,9	91,4	106,9	97,0	129,3
0	81,9	93,8	98,6	94,9	130,7
0	82,5	92,9	69,1	95,1	130,5
0	82,9	91,1	59,3	95,3	142,3
0	82,6	92,6	48,4	92,7	143,7
0	82,2	90,0	69,4	94,7	143,2
0	82,0	90,7	70,1	96,0	149,5
0	81,2	93,1	75,9	97,1	163,2
0	81,2	93,5	71,9	96,1	163,3
0	80,9	95,8	68,0	96,0	159,6
0	81,2	93,4	72,2	96,9	145,4
0	83,5	100,8	81,4	99,9	139,0
0	85,0	100,3	89,4	101,3	147,3
0	88,8	105,2	97,0	101,3	145,9
0	86,3	109,5	104,9	100,8	144,7
0	86,2	111,8	107,0	101,5	145,1
0	85,1	116,0	109,6	103,1	156,9
0	103,0	114,3	118,9	101,5	156,6
0	103,3	119,7	120,0	106,3	150,0
0	87,1	117,3	115,9	105,4	150,3
1	110,2	130,2	170,3	112,7	146,4
1	116,5	129,8	172,2	114,8	150,8
1	116,7	131,7	169,6	112,0	145,0
1	113,2	131,2	163,1	114,0	141,1
1	115,0	130,8	157,3	112,9	138,7
1	114,6	130,7	158,4	111,5	131,0
1	111,1	126,1	158,6	110,3	118,6
1	111,6	126,1	162,0	110,9	120,0
1	108,9	125,7	163,5	111,8	116,5
1	110,8	127,5	161,0	114,8	120,7
1	107,7	127,7	160,5	113,9	119,1
1	110,3	128,1	164,9	114,2	119,0
1	108,3	129,3	159,0	112,4	111,0
1	109,5	129,7	158,8	112,9	107,0
1	107,6	129,4	160,9	114,8	112,0
1	110,0	127,3	157,8	116,3	114,1
1	114,1	127,1	157,0	114,0	118,3
1	116,0	127,5	159,6	112,6	118,9
1	113,3	128,3	158,4	113,4	105,9
1	112,7	125,5	154,4	112,4	107,2
1	109,4	126,3	151,4	111,6	113,6
1	107,1	121,0	146,4	110,5	106,1

1	106,3	118,8	138,9	110,6	112,0
1	107,6	113,0	128,8	109,4	109,2
1	106,4	111,2	114,3	104,9	101,1
1	104,6	109,3	112,5	102,6	107,8
1	106,1	106,8	114,4	99,0	96,8
1	102,0	100,3	104,9	100,3	102,7
1	100,8	99,9	111,7	101,1	102,9
1	100,3	101,4	110,3	101,5	102,5
1	99,3	101,5	101,1	99,3	104,7
1	98,2	100,6	93,5	99,3	102,0
1	96,6	96,1	89,2	98,6	94,0
1	95,2	92,7	86,3	99,5	95,5
1	95,9	91,8	83,6	97,8	92,8
1	94,6	88,3	78,1	96,2	97,2
1	93,6	87,8	67,4	92,9	93,3
1	93,5	86,5	64,3	94,6	101,1
1	93,2	85,5	67,6	94,7	102,2
1	93,2	84,4	65,5	97,3	104,4
1	94,0	90,1	70,8	97,4	108,4
1	93,3	96,4	74,0	96,6	111,0
1	92,6	95,3	71,3	96,6	108,8
1	93,0	91,4	70,1	97,8	115,1
1	91,1	90,5	72,1	97,5	116,7
1	91,5	92,9	74,8	98,7	120,1
1	91,5	91,1	71,9	97,7	110,9
1	92,0	93,1	77,1	95,6	110,4
1	91,0	95,8	85,1	95,6	107,6
1	91,4	99,1	86,4	96,5	113,5
1	91,7	101,5	84,7	97,5	115,9
1	90,1	101,2	86,1	97,4	117,5
1	91,8	104,3	83,1	99,2	119,9
1	91,8	103,0	82,3	98,8	121,6
1	90,4	104,6	82,5	99,4	121,1
1	89,4	104,7	86,1	101,5	121,1
1	89,4	106,3	90,2	101,8	125,9
1	88,7	112,8	92,1	103,2	120,8
1	92,9	110,1	96,4	105,0	115,3
1	93,7	107,5	97,9	105,0	123,4
1	93,5	107,6	104,7	104,9	127,4
1	92,1	109,7	102,9	105,2	117,4
1	92,7	112,5	101,4	105,1	119,5
1	94,4	113,7	106,3	106,8	115,3
1	94,1	114,3	111,9	103,7	119,7
1	94,7	114,3	111,8	103,7	109,0

1	92,0	113,4	112,3	102,2	106,9
1	89,3	113,6	113,3	101,9	102,4
1	88,5	111,9	116,0	101,3	98,7
1	88,8	109,8	119,5	103,8	99,1
1	89,4	108,6	112,9	104,5	102,1
1	87,7	109,4	105,8	103,4	106,5
1	92,5	106,7	107,9	102,5	108,7
1	91,0	101,2	109,6	101,4	113,2
1	92,2	103,3	110,8	102,4	118,6
1	92,1	104,7	111,0	101,5	116,8
1	89,5	105,5	111,8	99,0	106,9
1	88,5	105,1	107,6	99,4	106,8
1	88,2	103,5	105,3	100,2	107,7
1	87,2	102,0	101,3	98,6	114,1
1	87,6	100,6	100,8	97,8	114,1
1	86,4	95,9	97,8	98,0	127,3
1	86,9	88,6	98,6	98,1	129,7
1	87,8	91,1	99,9	97,2	128,4
1	90,0	93,5	104,2	95,8	127,6
1	90,4	97,9	95,1	94,3	125,7
1	91,8	96,9	82,6	97,1	142,9
1	92,2	94,9	55,3	94,7	147,3
1	91,0	92,9	50,1	91,2	147,0
1	90,7	91,1	66,6	92,3	144,1
1	87,9	91,0	67,3	93,4	141,3
1	88,7	91,5	71,3	94,3	162,0
1	91,5	93,0	70,7	97,8	172,6
1	90,9	96,8	67,4	98,3	164,1
1	97,1	101,4	68,9	100,9	168,3
1	98,5	105,2	74,2	102,8	159,2
1	99,4	113,6	80,7	103,1	151,5
1	101,8	127,9	86,9	104,3	159,2
1	106,3	135,1	92,0	108,4	157,7
1	101,4	139,8	95,2	110,2	121,1
1	102,4	140,2	100,2	110,4	124,6
1	104,1	143,7	103,1	110,9	127,7
1	106,7	148,8	112,6	114,4	119,9
1	106,4	152,9	113,7	117,5	136,9

Ek 2. Çalışmada Kullanılan Kodlar

```
library(caret)
library(e1071)
library(readxl)
library(ggplot2)
library(tidyverse)

hb<- read_excel("C:/Users/ozlem/Desktop/veri.xlsx") # verinin bilgisayardan çekilmesi

View(hb)

my.df <- hb # my.df olarak atanması

dim(my.df) # verinin boyutunun görülmesi

# Verinin scale edilmesi

my.df_scaled <- my.df %>%

mutate(sınıf= case_when(sınıf == 0 ~ "ihracat", TRUE ~ "ithalat") %>% as.factor())

%>%

mutate_if(is.numeric, function(x) {(x - min(x)) / (max(x) - min(x))})

# Grafikler

ggplot(my.df_scaled, aes(x =gıda , y =hammadde , colour = sınıf )) +

  geom_point() +

  labs(title = 'Gıda ve Hammadde') # Şekil 7. 1. grafiği

ggplot(my.df_scaled, aes(x =gıda , y =yakıt , colour = sınıf )) +

  geom_point() +

  labs(title = 'Gıda ve Yakıt') # Şekil 7. 2. grafiği

ggplot(my.df_scaled, aes(x =gıda , y =imalat , colour = sınıf )) +

  geom_point() +

  labs(title = 'Gıda ve İmalat') # Şekil 7. 3. grafiği
```

```

ggplot(my.df_scaled, aes(x =gıda , y =sıtc , colour = sınıf )) +

  geom_point() +

  labs(title = 'Gıda ve Sıtc') # Şekil 7. 4. grafiği

ggplot(my.df_scaled, aes(x =hammadde , y =yakıt , colour = sınıf )) +

  geom_point() +

  labs(title = 'Hammadde ve Yakıt') # Şekil 7. 5. grafiği

ggplot(my.df_scaled, aes(x =hammadde , y =imalat , colour = sınıf )) +

  geom_point() +

  labs(title = 'Hammadde ve İmalat') # Şekil 7. 6. grafiği

ggplot(my.df_scaled, aes(x =hammadde , y =sıtc , colour = sınıf )) +

  geom_point() +

  labs(title = 'Hammadde ve Sıtc') # Şekil 7. 7. grafiği

ggplot(my.df_scaled, aes(x =yakıt , y =imalat , colour = sınıf )) +

  geom_point() +

  labs(title = 'Yakıt ve İmalat') # Şekil 7. 8. grafiği

ggplot(my.df_scaled, aes(x =yakıt , y =sıtc , colour = sınıf )) +

  geom_point() +

  labs(title = 'Yakıt ve Sıtc') # Şekil 7. 9. grafiği

#####

my.df$sınıf <- as.numeric(as.factor(my.df$sınıf)) # verinin sınıflandırılması

```

```

my.df$sınıf = factor(my.df$sınıf, labels = c(0, 1))

#####

# Gerekli fonksiyonlar

predict_prob <- function(model_train,test_data) {

  pred <- predict(model_train, test_data, probability=TRUE)

  attr(pred,"probabilities") %>%

  as.data.frame() %>%

  pull(1)%>%

  return()

}

predict_prob_logistic <- function(model_train,test_data) {

  pred <- predict(model_train, test_data, type='response') %>%

  as.data.frame() %>%

  return()

}

# ROC ve AUC için gerekli fonksiyonlar

library(pROC)

test_auc <- function(prob,test) {

  roc(test, prob)

}

test_auc_logistic <- function(prob,test) {

  roc(test~prob)

```

```

}

# Veriyi Eğitim ve Test olarak ikiye bölme

set.seed(2021)

index <- 1:nrow(my.df_scaled)

testindex <- sample(index, trunc(length(index)/2))

testing <- na.omit(my.df[testindex,])

training <- na.omit(my.df[-testindex,])

y_train <- training %>% select(sınıf) %>% as.matrix() %>% as.factor()

x_train <- training %>% select(-c(sınıf)) %>% scale() %>% as.matrix()

train_df <- data.frame(sınıf=y_train,x_train)

y_test <- testing %>% select(sınıf) %>% as.matrix() %>% as.factor()

x_test <- testing %>% select(-c(sınıf)) %>% scale() %>% as.matrix()

test_df <- data.frame(sınıf=y_test,x_test)

# svm fonksiyonları

# doğrusal kernel

set.seed(2021)

tune.linear = tune(svm, sınıf ~ . ,

  data = train_df,

  kernel = 'linear',

  ranges=list(gamma=2^(-5:5), cost=2^(-5:10)),

  control = tune.control(sampling="cross", cross=10),

  probability = TRUE)

best.linear = tune.linear$best.model

```

```

predict.linear_test = predict(best.linear,x_test)

confusionMatrix(table(predict.linear_test, y_test))

# radyal kernel

set.seed(2021)

tune.radial = tune(svm, sınıf ~ . ,

                  data = train_df,

                  kernel = 'radial',

                  ranges=list(gamma=2^(-5:5), cost=2^(-5:10)),

                  control = tune.control(sampling="cross", cross=10),

                  probability = TRUE)

best.radial = tune.radial$best.model

predict.radial_test = predict(best.radial,x_test)

confusionMatrix(table(predict.radial_test, y_test))

# Logistik Regresyon

set.seed(2021)

model_glm = glm(sınıf ~., data = train_df, family = "binomial")

model_glm_pred = predict(model_glm, newdata = test_df, type = "response")

confusionMatrix(table(ifelse(model_glm_pred > .5, 1, 0), y_test))

# olasılıklar

pred_svm_linear <- predict_prob(best.linear,test_df)

pred_svm_radial <- predict_prob(best.radial,test_df)

# AUC değerleri oluşumu

auc_logistic <- test_auc_logistic(prob=model_glm_pred,test=test_df$sınıf)

```

```

auc_svm_linear <- test_auc(prob=pred_svm_linear,test=test_df$sınıf)

auc_svm_radial <- test_auc(prob=pred_svm_radial,test=test_df$sınıf)


# ROC eğrisi karşılaştırmak için değerler

df_auc <- bind_rows(data_frame(TPR = auc_svm_linear$sensitivities,

                                FPR = 1 - auc_svm_linear$specificities,

                                Model = "SVM(Doğrusal)"),

                    data_frame(TPR = auc_svm_radial$sensitivities,

                                FPR = 1 - auc_svm_radial$specificities,

                                Model = "SVM(Radyal)"),

                    data_frame(TPR = auc_logistic$sensitivities,

                                FPR = 1 - auc_logistic$specificities,

                                Model = "Logistik Regresyon"))


# ROC eğrisi grafiği

df_auc %>%

  ggplot(aes(FPR, TPR, color = Model)) +

  geom_line(size = 1) +

  theme_bw() +

  coord_equal() +

  geom_abline(intercept = 0, slope = 1, color = "gray37", size = 1, linetype = "dashed")

+

  labs(x = "FPR (1 - Özgüllük)",

       y = "TPR (Hassasiyet)",

       title = "ROC Eğrisi ve AUC: Logistic ile SVM")

```

```

# AUC değerleri gösterimi

lapply(list(auc_svm_linear, auc_svm_radial, auc_logistic), function(x) {x[["auc"]]}))

# grid fonksiyonu oluşturmak

make.grid = function(x, n = 100) {

  grange = apply(x, 2, range)

  x1 = seq(from = grange[1,1], to = grange[2,1], length = n)
  x2 = seq(from = grange[1,2], to = grange[2,2], length = n)

  expand.grid(X1 = x1, X2 = x2)

}

# Şekil 7. 10. grafiği

#define data set and data frame

GıdaYakıt <- data.frame(my.df_scaled[,c(3,5)])

G.Y.SINIF <- as.factor(my.df_scaled$sınıf)

dat2 = data.frame(GıdaYakıt, G.Y.SINIF)

# gıda, yakıt ve bağımlı değişkeni train ve test olara ayırma

set.seed(2021)

intrain<- createDataPartition(y =dat2$G.Y.SINIF, p= 0.5, list = FALSE)

training1 <-dat2[intrain,]

testing1 <-dat2[-intrain,]

dim(testing1)

dim(testing1)

```

```

anyNA(dat2)

#test setine bölünen grafiği dat1 de birleştirme

P_TEST <- data.frame(testing1[,c(1,2)])

P_TRAIN <- data.frame(training1[,c(1,2)])

I_TEST <- as.factor(testing1$G.Y.SINIF)

I_TRAIN <- as.factor(training1$G.Y.SINIF)

dat3 = data.frame(P,I)

#Ftr svm model eğitim

svmfit1 = svm(G.Y.SINIF ~ ., data = training1, kernel = "radial", gamma= 0.8, cost =
10000, scale = FALSE)

print(svmfit1)

plot(svmfit1, dat2)

#ftr svm model test

svmfittest1 = svm(G.Y.SINIF ~ ., data = testing1, kernel = "radial", gamma= 0.8, cost =
10000, scale = FALSE)

print(svmfittest1)

plot(svmfittest1,dat2)

#GRid X1 and X2 values

xgrid_TEST = make.grid(P_TEST)

xgrid_TRAIN = make.grid(P_TRAIN)

xgrid[1:10,]

colnames(xgrid_TEST) <- colnames(xgrid_TRAIN) <- colnames(GıdaYakıt)

# svm fit model tahmini

ygrid_TEST = predict(svmfit1, xgrid_TEST)

```

```

ygrid_TRAIN = predict(svmfit1, xgrid_TRAIN)

#grafikteki noktalar ve destek vektörlerinn belirlenmesi

par(mfrow = c(1, 2)) # 2 x 2 pictures on one plot

plot(xgrid_TRAIN, col = c("#00FFFF", "#FF00FF")[as.numeric(ygrid_TRAIN)], pch =
20, cex = .2,

      main = "eğitim")

points(P_TRAIN, col = I_TEST, pch =20 )

points(P_TRAIN[svmfit1$index,], pch =0.5, cex =2)

plot(xgrid_TEST, col = c("#00FFFF", "#FF00FF")[as.numeric(ygrid_TEST)], pch = 20,
cex = .2,

      main = "test")

points(P_TEST, col = I_TEST, pch =20 )

points(P_TEST[svmfit1$index,], pch =0.5, cex =2)

```

Şekil 7. 11. grafiği

#define data set and data frame

```
Gıdaİmalat <- data.frame(my.df_scaled[,c(3,6)])
```

```
G.İ.SINIF <- as.factor(my.df_scaled$sınıf)
```

```
dat1 = data.frame(Gıdaİmalat, G.İ.SINIF)
```

gıda, imalat ve bağımlı değişkeni train ve test olara ayırma

```
set.seed(2021)
```

```
intrain<- createDataPartition(y=dat1$G.İ.SINIF, p= 0.5, list = FALSE)
```

```
training <-dat1[intrain,]
```

```
testing<-dat1[-intrain,]
```

```

dim(testing)

dim(testing)

anyNA(dat1)

#test setine bölünen grafiği dat1 de birleştirme

W_TEST <- data.frame(testing[,c(1,2)])

W_TRAIN<- data.frame(training[,c(1,2)])

E_TEST <- as.factor(testing3$G.İ.SINIF)

E_TRAIN<-as.factor(training2$G.İ.SINIF)

#Ftr SVM model

svmfit = svm(G.İ.SINIF ~ ., data = training,kernel = "radial", gamma= 0.2, cost =
10000, scale = FALSE)

print(svmfit)

plot(svmfit, dat1)

# plot fit data

svmfittest = svm(G.İ.SINIF ~ ., data = testing,kernel = "radial", gamma= 0.2, cost =
1000, scale = FALSE)

print(svmfittest)

plot(svmfittest, dat1)


#Grid X1 and X2 values

xgrid_TEST = make.grid(W_TEST)

xgrid_TRAIN = make.grid(W_TRAIN)

xgrid[1:10,]

colnames(xgrid_TEST) <- colnames(xgrid_TRAIN) <- colnames(Gıdaİmalat)

```

```

# predict svmfit

ygrid_TEST = predict(svmfit, xgrid_TEST)

ygrid_TRAIN = predict(svmfit, xgrid_TRAIN)

xgrid = make.grid(Gıdaİmalat)

xgrid[1:10,]

colnames(xgrid) <- colnames(Gıdaİmalat)

#predict svmfir

ygrid = predict(svmfit, xgrid)

#grafikteki noktalar ve destek vektörlerinn belirlenmesi

par(mfrow = c(1, 2)) # 2 x 2 pictures on one plot

plot(xgrid_TRAIN, col = c("#00FFFF", "#FF00FF")[as.numeric(ygrid_TRAIN)], pch =
20, cex = .2,

      main = "eğitim")

points(W_TRAIN, col = E_TEST, pch =20 )

points(W_TRAIN[svmfit$index,], pch =0.5, cex =2)

plot(xgrid_TEST, col = c("#00FFFF", "#FF00FF")[as.numeric(ygrid_TEST)], pch = 20,
cex = .2,

      main = "test")

points(W_TEST, col = E_TEST, pch =20 )

points(W_TEST[svmfit$index,], pch =0.5, cex =2)


#Şekil 7. 12. grafiği

#define data set and data frame

GıdaHammade <- data.frame(my.df_scaled[,c(3,4)])

```

```

G.H.SINIF <- as.factor(my.df_scaled$sınıf)

dat3 = data.frame(GıdaHammade, G.H.SINIF)

#gıda hammadde ve bağımlı değişkeni train ve test olara ayırma

set.seed(2021)

intrain<- createDataPartition(y =dat3$G.H.SINIF, p= 0.5, list = FALSE)

training2 <-dat3[intrain,]

testing2 <-dat3[-intrain,]

dim(testing2)

dim(training2)

anyNA(dat2)

#test setine bölünen grafiği dat1 de birleştirme

H_TEST <- data.frame(testing2[,c(1,2)])

H_TRAIN <- data.frame(training2[,c(1,2)])

U_TEST <- as.factor(testing2$G.H.SINIF)

U_TRAIN <- as.factor(training2$G.H.SINIF)

#Ftr SVM model train

svmfit2= svm(G.H.SINIF ~ ., data = training2, kernel = "radial", gamma= 0.8, cost =
10000, scale = FALSE)

print(svmfit2)

plot(svmfit2, dat3)

# ftr svm model test

svmfittest2 = svm(G.H.SINIF ~ ., data = testing2, kernel = "radial", gamma= 0.8, cost =
10000, scale = FALSE)

print(svmfittest2)

```

```

plot(svmfittest2,dat3)

#GRid X1 and X2 values

xgrid_TEST = make.grid(H_TEST)

xgrid_TRAIN = make.grid(H_TRAIN)

xgrid[1:10,]

colnames(xgrid_TEST) <- colnames(xgrid_TRAIN) <- colnames(GıdaHammade)

#predict svmfit

ygrid_TEST = predict(svmfit2, xgrid_TEST)

ygrid_TRAIN = predict(svmfit2, xgrid_TRAIN)

#grafikteki noktalar ve destek vektörlerinn belirlenmesi

par(mfrow = c(1, 2)) # 2 x 2 pictures on one plot

plot(xgrid_TRAIN, col = c("#00FFFF", "#FF00FF")[as.numeric(ygrid_TRAIN)], pch =
20, cex = .2,

    main = "eğitim")

points(H_TRAIN, col = U_TEST, pch =20 )

points(H_TRAIN[svmfit2$index,], pch =0.5, cex =2)

plot(xgrid_TEST, col = c("#00FFFF", "#FF00FF")[as.numeric(ygrid_TEST)], pch = 20,
cex = .2,

    main = "test")

points(H_TEST, col = U_TEST, pch =20 )

points(H_TEST[svmfit2$index,], pch =0.5, cex =2)

```