T.C.
KASTAMONU ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MALZEME BİLİMİ VE MÜHENDİSLİĞİ ANA BİLİM DALI

İKİLİ SINIFLANDIRMA PROBLEMLERİNDE ÇOK KATMANLI ALGILAYICI VE DESTEK VEKTÖR MAKİNELERİ SINIFLANDIRICILARININ HİPERPARAMETRELERİNİN EN İYİLENMESİ

Wisam Salem Ali ZANBIL

YÜKSEK LİSANS TEZİ

Doç. Dr. Kemal AKYOL

HAZİRAN - 2022
KASTAMONU

TEZ ONAYI

Wisam Salem Ali ZANBIL tarafından hazırlanan "**İkili Sınıflandırma Problemlerinde Çok Katmanlı Algılayıcı ve Destek Vektör Makineleri Sınıflandırıcılarının Hiperparametrelerinin En İyilenmesi**" adlı tez çalışmasının savunma sınavı **16.06.2022** tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği ile Kastamonu Üniversitesi Fen Bilimleri Enstitüsü **Malzeme Bilimi ve Mühendisliği Ana Bilim Dalı Yüksek Lisans Tezi** olarak kabul edilmiştir.

Danışman	Doç. Dr. Kemal AKYOL Kastamonu Üniversitesi	.....
Jüri Üyesi	Doç. Dr. Salih GÖRGÜNOĞLU Kastamonu Üniversitesi	.....
Jüri Üyesi	Doç. Dr. Abdulkadir KARACI Samsun Üniversitesi	.....

Jüri üyeleri tarafından kabul edilmiş olan bu tez Kastamonu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunca onanmıştır.

Enstitü Müdürü Prof. Dr. İzzet ŞENER .....

TAAHHÜTNAME

Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu; ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını, bilimsel etiğe uygun olarak kaynak gösterildiğini bildirir ve taahhüt ederim.

Wisam Salem Ali ZANBIL

ÖZET

YÜKSEK LİSANS TEZİ

İKİLİ SINIFLANDIRMA PROBLEMLERİNDE ÇOK KATMANLI ALGILAYICI VE DESTEK VEKTÖR MAKİNELERİ SINIFLANDIRICILARININ HİPERPARAMETRELERİNİN EN İYİLENMESİ

Wisam Salem Ali ZANBIL

KASTAMONU ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ
MALZEME BİLİMİ VE MÜHENDİSLİĞİ ANA BİLİM DALI
DANIŞMAN: Doç. Dr. Kemal AKYOL

Makine öğreniminde genellikle sağlam çalışan ve doğru tahminler yapan bir model oluşturulması hedeflenir. Parametre değerlerinin en iyilenmesi bu amaç doğrultusunda iyi bir çözüm sunar. Bu tez çalışmasının amacı, Destek Vektör Makineleri ve Çok Katmanlı Algılayıcı makine öğrenmesi algoritmalarının ikili sınıflandırma problemlerindeki başarılarını arttırmak için parametre en iyilemesi gerçekleştirmektir. Bunun için Rastgele Arama, Bayesian Arama ve Optuna en iyileme yöntemlerinin bu sınıflandırıcılara katkıları ele alınmıştır. Bu çerçevede, üç veri seti üzerinde deneysel çalışmalar gerçekleştirilmiştir. Hem Destek Vektör Makineleri hem de Çok Katmanlı Algılayıcı için Rastgele Arama ve Bayesian Arama metotları çoğunlukla Optuna metodundan daha iyi performans sunmuşlardır. Dahası, deneysel çalışmalar ele alınan probleme bağlı olarak en iyileme yöntemlerinin bu sınıflandırıcılar üzerinde olumlu etkisinin olabileceğini bize göstermiştir.

ANAHTAR KELİMELER: Destek vektör makineleri, çok katmanlı algılayıcı, ikili sınıflandırma, en iyileme, hiperparametre ayarlama.

Haziran 2022, 55 Sayfa,

ABSTRACT

MSC THESIS

TUNING THE HYPERPARAMETERS OF MULTILAYER PERCEPTRON AND SUPPORT VECTOR MACHINES CLASSIFIERS IN BINARY CLASSIFICATION PROBLEMS

Wisam Salem Ali ZANBIL

KASTAMONU UNIVERSITY INSTITUTE OF SCIENCE
DEPARTMENT OF MATERIALS SCIENCE AND ENGINEERING
SUPERVISOR: Assoc. Prof. Dr. Kemal AKYOL

In machine learning, it is generally aimed to create a model that works well and makes accurate predictions. Optimization of parameter values provides a good solution for this purpose. The aim of this thesis study is to perform parameter optimization to increase the success of Support Vector Machines and Multilayer Perceptron machine learning algorithms in binary classification problems. For this, the contributions of Random Search, Bayesian Search, and Optuna optimization methods to these classifiers were handled. In this context, experimental studies were carried out on three datasets. For both Support Vector Machines and Multilayer Perceptron, Random Search and Bayesian Search methods mostly outperformed the Optuna method. Moreover, experimental studies showed us that optimization methods can have a positive effect on these classifiers depending on the problem under consideration.

KEYWORDS: Support vector machines, multi-layer perceptron, binary classification, optimization, hyperparameter tuning.

June 2022, 55 Page,

TEŞEKKÜR

Bu tez çalışmasının belirlemesinde, araştırılmasında, geliştirilmesinde ve sonuçlandırılmasında yapıcı ve yönlendirici katkılarıyla büyük emeği geçen, desteğini ve ilgisini esirgemeyen danışman ve değerli hocam Doç. Dr. Kemal AKYOL'a sonsuz teşekkürlerimi sunarım. Görüş ve bilgileriyle beni yönlendiren değerli hocalarım Doç. Dr. Salih GÖRGÜNOĞLU ve Doç. Dr. Abdulkadir KARACI'ya çok teşekkür ederim. Son olarak, bu tezi yazma aşamasında moralimi ve enerjimi hep yüksek tutarak destekleriyle her zaman yanımda olan aileme ve arkadaşlarıma içtenlikle teşekkür ederim.

Wisam Salem Ali ZANBIL

Kastamonu, 2022

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

Sayfa

TABLOLAR DİZİNİ

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar

DVM : Destek Vektör Makineleri
ÇKA : Çok Katmanlı Algılayıcı
AİK : Alıcı İşlem Karakteristiği
C : Düzenlileştirme Parametresi
DP : Doğru Pozitif
DN : Doğru Negatif
YP : Yanlış Pozitif
YN : Yanlış Negatif
EAA : Eğri Altındaki Alan
RA : Rastgele Arama
BA : Bayesian Arama
OEİ : Optuna En İyileyici

1. GİRİŞ

Makine öğrenmesi, bilgisayar programının kendisine verilen verilerden öğrendiği ve yeni gözlemler veya sınıflandırmalar yaptığı denetimli bir öğrenme yaklaşımıdır. Sınıflandırmanın temel amacı, çeşitli makine öğrenme algoritmaları ile deneysel çalışmalar yaparak en iyi doğruluğu elde etmektir. Sınıflandırma problemleri, verideki tahmin edici ilişkileri temsil eden en iyi modeli bulmanın hedeflendiği bir en iyileme problemi olarak görülebilir (Syarif vd., 2016). Bir veri koleksiyonuna dayalı model oluşturmak için makine öğrenimi teknikleri kullanılır. Makine öğrenimi sınıflandırma, tahminleme ve kümeleme olmak üzere üç tür sorunu çözmek için kullanılır. Mevcut eğitim verilerinin türlerine ve kategorilerine bağlı olarak, uygun makine öğrenimi yöntemini uygulamak için "denetimli öğrenme", "denetimsiz öğrenme", "yarı denetimli öğrenme" ve "pekiştirmeli öğrenme" arasında seçim yapılması gerekir (Ray, 2019). Otomatik model geliştirme sürecine "eğitim" adı verilir ve eğitim için kullanılan verilere eğitim verileri adı verilir. Eğitilen model, girdi değişkenlerinin çıktıya nasıl aktarıldığına dair yeni bilgiler verebilir ve eğitim verilerine dâhil edilmeyen benzersiz girdi değerleri için tahminler üretebilir (Baştanlar ve Özuysal, 2014).

Makine öğrenimi algoritmaları, bir veri setinden modeller oluşturmak için kullanılır ve bu algoritmalar birçok hiperparametreye sahiptirler. Algoritma eğitilirken veriler üzerinde bir dizi en iyi hiperparametre değeri keşfedilmeye çalışılır. Diğer bir deyişle, model parametre değerlerinden en iyi performans sunanı keşfetmek için farklı değer kümeleri üzerinde eğitim ve test süreçlerini içeren deneyler yapılması gerekir Bu, makine öğreniminde kritik bir rol oynayan hiperparametrelerin optimizasyonu veya en iyilenmesi olarak bilinen bir süreçtir (Wu vd., 2019).

En iyi performansı elde etmek için bir sınıflandırıcı parametre değerlerini manuel olarak ayarlamak ve sınıflandırıcıyı tekrar tekrar teste tabi tutan deneyler yapmak zaman ve çaba gerektirir. Sınıflandırıcı algoritmanın herhangi bir parametresi için tanımlı değerlerden en iyisinin tespiti için Rastgele Arama, Bayesian Arama ve Optuna En İyileme gibi otomatik arama yöntemleri kullanılarak gereken süre ve ortaya çıkabilecek hatalar en aza indirilebilir.

Ölçeklenebilir bir yaklaşıma sahip olan Rastgele Arama, hem makul ölçüde verimli olan hem de ızgara aramasının uygulama basitliği ve tekrarlanabilirliğinin avantajlarını koruyan bir yöntemdir (Bergstra ve Bengio, 2012). Bayesian Arama, arama alanını modellemek için "vekil" olarak adlandırılan bir tahmine dayalı modelin kullanıldığı ve mümkün olan en kısa sürede iyi bir parametre değerleri kombinasyonuna ulaşmak için Bayesian en iyilemesini kullanır (Wu vd., 2019). Optuna En İyileyici, esas olarak makine öğrenimi için tasarlanmış otomatik bir hiperparametre en iyileme yazılımı çerçevesidir. Optuna ile yazılan kod yüksek modülerliğe sahiptir ve hiperparametreler için arama alanları dinamik olarak oluşturabilir (Optuna Contributors, 2018a).

Bu tez çalışmasındaki deneyler iki makine öğrenimi algoritması olan Destek Vektör Makineleri ve Çok Katmanlı Algılayıcı üzerinde Rastgele Arama, Bayesian Arama ve Optuna En İyileyici algoritmaları kullanılarak gerçekleştirilmiştir. Birçok çalışmada oldukça sık kullanılan doğruluk, duyarlılık, özgüllük, kesinlik ve F1-ölçütü sınıflandırma metrikleri kullanılarak modellerin performansları karşılaştırılmıtşır.

1.1 İkili Sınıflandırma Problemi

İkili sınıflandırma problemi, iki hedef sınıfı olan bir sınıflandırma problemi olarak değerlendirilir. Hastalık var/yok, orman yangını var/yok bu probleme örnek olarak verilebilir. Genellikle, ikili sınıflandırma problemleri kavram öğrenme olarak ifade edilir (Mitchell, 1997). Bu tür problemlerde, makine öğrenmesi algoritmasının performansı, tahmin edilen sınıfları içeren tahmin seti ile gerçek sınıfları içeren test seti karşılaştırması yapılarak değerlendirilir.

1.2 Çalışmanın Önemi

Hiperparametre en iyilemesi bir modelin hiç görmediği veriler için en iyi sınıflandırma doğruluğunu üreten en iyi parametre değerini bulmak için kullanır. Bu tez çalışmasının odak noktası, en iyileme algoritmalarından yararlanarak ikili sınıflandırma problemlerinde sınıflandırıcıların doğruluğunu artırmaktır. Bu çerçevede, bu tez çalışmasında ikili sınıflandırma problemlerinde Rastgele Arama, Bayesian Arama ve

Optuna En İyileyici metotları kullanılarak Destek Vektör Makineleri ve Çok Katmanlı Algılayıcı algoritmaları için en uygun parametre değerleri araştırılmıştır.

1.3 Motivasyon Ve Katkı

Bu tez çalışmasının amacı, ikili sınıflandırma problemlerinde en iyi performansı elde etmek diğer bir deyişle yanlış pozitif ve yanlış negatifi en aza indirmektir. Bu doğrultuda, varsayılan parametre değerleri ile kullanılan sınıflandırıcı algoritmaların performanslarını iyileştirmek amacıyla çeşitli en iyileme metotlarının katkıları analiz edilmiş ve elde edilen sonuçlar değerlendirilmiştir. Bu çalışma kapsamında öncelikle veri seti üzerinde veri ön işleme ve normalizasyon işlemleri gerçekleştirilmiş, ardından sınıflandırıcı algoritmalara dayalı oluşturulan modellerin performansları karşılaştırılmıştır. Dahası, en iyi genelleme yeteneğine sahip bir model elde etmek için sınıflandırıcının parametre değerleri 5-kat çapraz doğrulama tekniğine başvurularak belirlenmiştir.

Birinci bölüm, bu çalışmanın önemini ve motivasyonu sunar. İkinci bölüm, literatürdeki en iyileme teknikleri ile yapılan bazı çalışmaları sunar. Üçüncü bölüm, bu tez çalışmasında kullanılan veri setleri ve yararlanılan yöntem ve teknikler hakkında bilgiler sunar. Makine öğrenmesi yaklaşımı, Destek Vektör Makineleri ve Çok Katmanlı Algılayıcı hakkında detaylı bilgiler verir. Dahası, hiperparametre en iyileminin önemini vurgular ve bunun yanı sıra, Rastgele Arama, Bayesian Arama ve Optuna En İyileyici metotlarını açıklar. Dördüncü bölüm, ikili sınıflandırma problemleri için bu tez çalışmasında yapılan deneysel çalışmaları içerir. Dahası, elde edilen bulguları değerlendirir ve tartışır. Son olarak, beşinci bölüm bu tez çalışmasının sonuçlarını ve ileride yapılacak çalışmaları sunar.

2. LİTERATÜR TARAMASI

Makine öğrenimi ve hiperparametre ayarlama çalışmalarında amaç en iyi modeli otomatik olarak seçmektir (Shahriari vd., 2016). Bu bölüm, makine öğrenimi ve en iyileme tekniği ile ilgili literatürdeki bazı çalışmaları sunar. Örneğin, Mantovani vd. Destek Vektör Makineleri (DVM) algoritmasının hiperparametrelerini ayarlamak için meta-sezgisel yöntem, rastgele arama ve ızgara aramasını karşılaştırmışlardır. Yazarlar, rastgele aramanın meta-sezgisel ve ızgara arama teknikleri ile eşdeğer performans sunduğunu ve bu yöntemlerden daha düşük hesaplama maliyetine sahip olduğunu göstermişlerdir (Mantovani vd., 2015).

Aljarah vd. (2018), DVM'yi optimize etmek için Çekirge Optimizasyon algoritmasına dayalı yeni bir hibrit yaklaşım sunmuşlardır. Uygun özellikler ve optimal çekirdek parametreleri ile önerdikleri yaklaşımın diğer optimize edicilerden daha iyi genel sınıflandırma doğruluğuna sahip olduğunu rapor etmişlerdir.

Syarif vd. (2016), genetik algoritma (GA) temelli DVM parametre en iyilemesinin ızgara aramasından 15,9 kattan daha hızlı olduğunu göstermişlerdir. Yazarlar, GA'nın ortalama 294 saniye çalışma süresine sahip olduğunu, ızgara aramasının ise yaklaşık 4680 saniye olduğunu rapor etmişlerdir.

Bergstra ve Bengio (2012), rastgele arama ve ızgara aramanın karşılaştırmasını içeren deneyler yapmışlardır. Yazarların çalışmasına göre, ızgara arama çok fazla deneme gerçekleştirirken rastgele arama, çoğu durumda daha iyi modeller bulmuş ve daha az hesaplama süresi gerektirmiştir.

Valarmathi ve Sheela (2021), Rastgele Orman ve Aşırı Gradyan Artırma sınıflandırıcılarının parametrelerinin en iyilenmesini gerçekleştirmişlerdir. Yazarlar, bu sınıflandırıcıların performanslarını ızgara arama, rastgele arama ve genetik programlama ile 10 kat çapraz doğrulama tekniği çerçevesinde değerlendirmiş olup Rastgele Arama ve Rastgele Orman ikilisi ile en yüksek sonucu elde etmişlerdir.

Srinivas ve Katarya çalışmalarında Aşırı Gradyan Artırma (Extreme Gradient Boosting, XGBoost) algoritmasını optimize etmek için Optuna temelli hyOPTXg modelini önermişlerdir. Yazarlar, Cleveland, kalp yetmezliği ve kalp hastalığı kardiyovasküler hastalık veri setleri üzerinde optimize edilmiş bir XGBoost sınıflandırıcısı ile yaptıkları deneysel çalışmalarda en yüksek doğruluğu %94,7 ile Cleveland veri setinde elde etmişlerdir (Srinivas ve Katarya, 2022).

Xia vd. (2017), Aşırı Gradyan Arttırma sınıflandırıcısının hiperparametrelerini ayarlamak için Bayesian hiperparametre en iyileme yöntemini kullanmışlardır. Yazarlar, beş verisetinden dört tanesi üzerinde Bayesian hiperparametre en iyileme metodunun rastgele arama, ızgara arama, manuel arama ve temel modellerden daha iyi performans gösterdiğini ifade etmişlerdir.

Lin vd. (2008), çalışmalarında ele aldıkları problemler için DVM'nin parametre değerlerini belirleme ve özellik seçimi için Parçacık Sürü Optimizasyonu (PSO) tabanlı bir çalışma sunmuşlardır. Yazarlar, sınıflandırma doğruluğunda PSO+DVM olarak adlandırılan önerdikleri yöntemin ızgara aramasından daha iyi performans gösterdiğini ve GA+DVM ile benzer bir sonuca sahip olduğunu rapor etmişlerdir.

Wang vd. (2011), PSO ve ızgara arama algoritmalarının birlikte kullanıldığı bir çalışma sunmuşlardır. Yazarların elde ettiği sonuçlara göre, PSO ve ızgara arama, tek başına ızgara arama yönteminden ve tek başına PSO algoritmasından daha yüksek tanıma doğruluğuna sahiptir. Ayrıca, yazarlar önerdikleri çalışmanın tek başına ızgara aramadan daha hızlı olduğunu ifade etmişlerdir.

Ramadhan vd. (2017), insan sesinin özelliklerine dayalı olarak cinsiyeti tespit etmek için Rastgele Orman sınıflandırıcısı kullanmıştır. Yazarlar, en iyi sonuçları elde etmek için bu sınıflandırıcının 'sqrt' ve ağaç sayısı parametreleri üzerinde ince ayar uygulamışlar ve parametre ayarı için ızgara arama yaklaşımını kullanmışlardır. Yazarlar, parametre ayarı olmadan elde ettikleri %96,75 sınıflandırma doğruluğunu parametre ayarı ile %96,907'ye yükselttiklerini ifade etmişlerdir.

Lacerda vd. (2021), Covid-19 hastalığının teşhisi için yaptıkları çalışmalarında hiper bant yöntemini optimize etmek ve dört hiperparametreyi değerlendirmek için Optuna

çerçevesi içeren bir model sunmuşlardır. Yazarların önerdikleri model, %97 duyarlılığa, %82 kesinliğe ve %88 doğruluğa sahip olup, %94 duyarlılık, %82 kesinlik ve %87 doğruluk sunan temel modelden daha yüksek performans sunmuştur.

Literatürdeki bu çalışmalar, sınıflandırıcı algoritmalar için en iyi hiperparametre değerinin tespitinin önemini ortaya koymaktadır. Bu bağlamda, bu tez çalışmasında üç veri seti üzerinde Destek Vektör Makineleri ve Çok Katmanlı Algılayıcı algoritmalarının en iyi sınıflandırma performanslarını elde etmek için Rastgele Arama, Bayesian Arama ve Optuna En İyileyici metotları ile deneysel çalışmalar gerçekleştirilmiş ve elde edilen sonuçlar karşılaştırılmıştır.

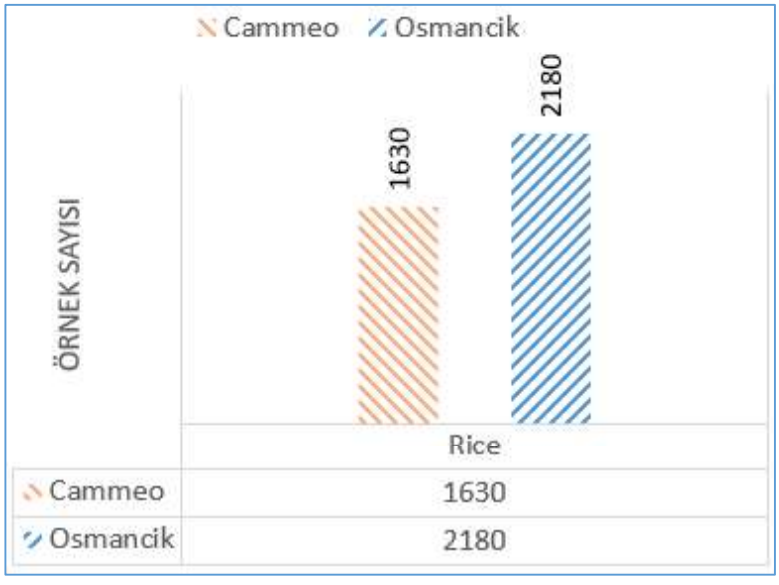
3. MATERYAL VE METOT

3.1 Veri Setleri

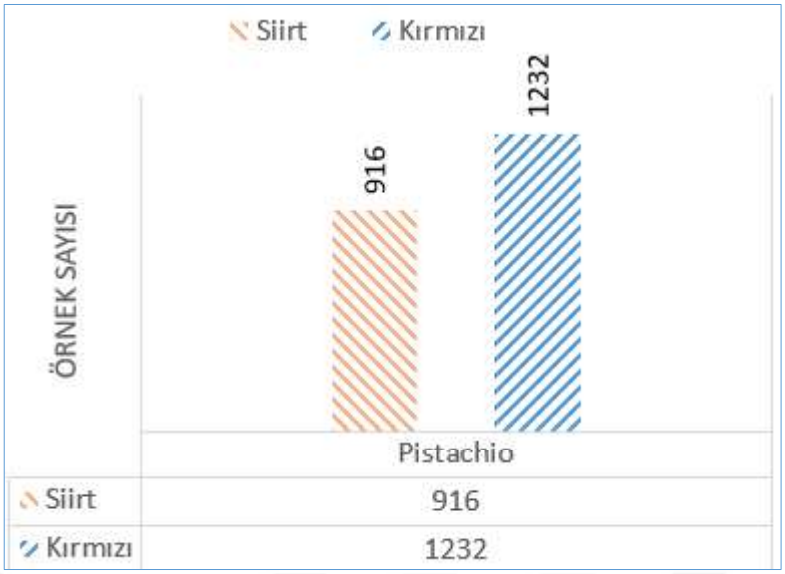
Tablo 3.1'de sunulduğu üzere, bu tez çalışmasında sırasıyla 8, 29 ve 754 öznitelik içeren üç veri seti kullanılmıştır. *Parkinson* (Sakar vd., 2019) ve *Rice* (Cinar ve Koklu, 2019) verisetleri UCI Machine Learning Repository'den indirilmiştir. *Pistachio* (Ozkan vd., 2021) veriseti Kaggle web sitesinden indirilmiştir. Tablo 3.1, bu verisetlerindeki örnek sayılarını ve sınıfları ile birlikte öznitelik sayılarını özetlemektedir. Örneğin, 756 örnek kayıt içeren Parkinson veriseti, 753 öznitelik bilgisi ve 1 sınıf bilgisi olmak üzere 754 boyutludur. Bunun yanı sıra veri setlerinin hedef sınıflarının dağılımı sırasıyla Şekil 3.1, Şekil 3.2 ve Şekil 3.3'te sunulmuştur. Buna göre örneğin Veriseti-1'deki 1630 örnek 'Cammeo' sınıfına, 2180 örnek ise 'Osmancik' sınıfına aittir.

Tablo 3.1 Veri setleri

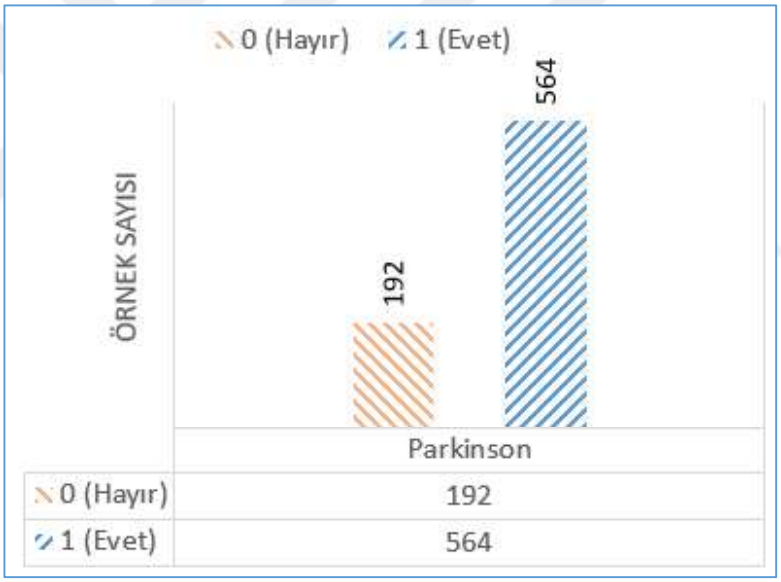
	Veri seti adı	**Örnek Sayısı**	**Boyut**
Veri seti-1	Rice	3810	8
Veri seti-2	Pistachio	2148	29
Veri seti-3	Parkinson	756	754



Şekil 3.1 Veri seti-1 örnek dağılımı



Şekil 3.2 Veri seti-2 örnek dağılımı



Şekil 3.3 Veri seti-3 örnek dağılımı

3.2 Makine Öğrenmesi

Makine öğrenimi, zeki insan davranışının taklidi olan yapay zekânın bir alt alanıdır. Yapay zeka, ayrıca insanlar gibi düşünen ve hareket eden sistemler ve rasyonel düşünen ve hareket eden sistemler olarak ifade edilebilir (Kok vd., 2009). Veri ve bilgiye dayalı öğrenim için bilgisayar algoritmalarının kullanıldığı makine öğrenmesi, bilgisayar bilimlerinde hızla gelişen bir alandır. Örneğin, makine öğrenimi algoritmaları otonom sürüş yapan arabaların insanlarla iletişim kurmasını sağlar

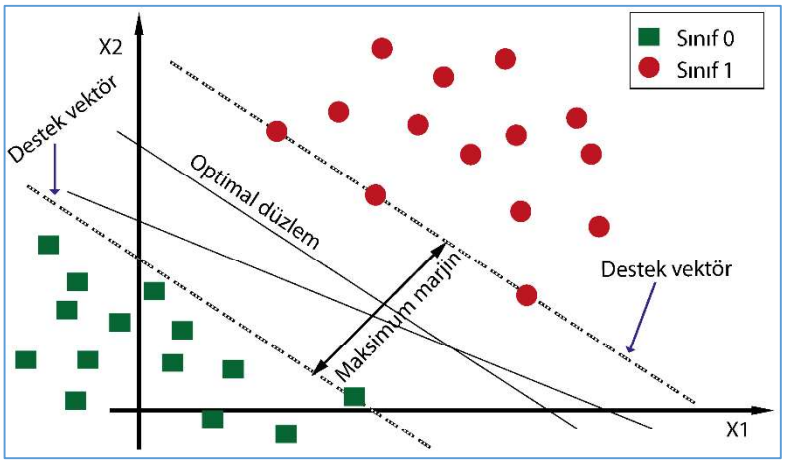
(Bhatnagar, 2018). Ancak, makine öğrenimi tahmin edilebilir sonuçları olan basit bir süreç değildir. Eğitim verilerine dayalı olarak bir makine öğrenmesi algoritma ile başarılı bir model oluşturulabilir. Örneğin, bir makine öğrenmesi algoritması ile tahmine dayalı bir model inşa edilir. Ardından veriler tahmin modeline girdi olarak gönderilir ve modelin eğitimine dayalı bir tahmin gerçekleşir (Hurwitz ve Kirsch, 2018).

Sınıflandırma tipik bir denetimli öğrenme görevidir. Eğitim veri seti, makine öğrenimi algoritmalarını eğitmek için ilişkili etiketlere sahip örnekler içermelidir. Denetimli makine öğrenmesi algoritmaları hem sınıflandırma hem de regresyon için yaygın olarak kullanılmaktadır (Gan, 2020). Denetimli öğrenme, girdi-çıktı çifti örneklerine dayalı olarak bir girdiyi bir çıktıya eşleyen öğrenme işlevidir. Bir dizi eğitim örneğinden oluşan etiketli eğitim verilerinden bir işlev çıkarır. Denetimli makine öğrenimi algoritmaları için veri seti, eğitim ve test veri setlerine bölünür. Eğitim veri seti, tahmin edilmesi veya sınıflandırılması gereken bir hedef sınıf niteliğine sahiptir. Tüm algoritmalar, eğitim veri setinden bazı kalıpları öğrenir ve yaptığı çıkarımları tahmin veya sınıflandırma için test veri setine uygular (Mahesh, 2020).

Denetimli öğrenmede hedef sınıfı belli olan örnekler ile öğrenme gerçekleştirilir iken denetimsiz öğrenmede hedef sınıf etiketlemesi yoktur. Kümeleme algoritmaları ile veriler gruplandırılır. Pekiştirmeli öğrenmede ise, model işlemin doğruluğu için sistemden geri bildirim alır ve böylece sonuçlardan çıkarımlar yaparak öğrenir.

3.2.1 Destek Vektör Makineleri

İstatistiksel öğrenme kuramına dayalı bir denetimli makine öğrenme algoritması olan Destek Vektör Makineleri (DVM), 1963 yılında Vladimir Vapnik ve Alexey Chervonenkis tarafından tanıtılmıştır (Akpınar, 2014). Şekil 3.4'te görüldüğü gibi bu algoritma iki sınıfı en uygun şekilde ayıran bir hiper düzlem tanımlar. Doğrusal olarak ayrılması mümkün olmayan veriler için DVM, çekirdek fonksiyonları ile başarılı bir çözüm sunar (Metlek ve Kayaalp 2020).



Şekil 3.4 Destek Vektör Makineleri sınıflandırıcısının temsili gösterimi

DVM, iki kategori arasındaki optimal hiper düzlemi keşfederek yüksek performans üretmesi amaçlanan bir tür denetimli öğrenme modelidir. (Khan vd., 2018). DVM, iki sınıfı ayıran çok sayıda hiper düzleme sahip olabilir ve mevcut tüm hiper düzlemlerin en yüksek marja sahip en uygun hiper düzlemi dikkate alır. İdeal hiperdüzlem, diğer örneklere bakılmaksızın yalnızca kendisine en yakın örneklere dayalı olarak belirlenir. Bu örnekler, destek vektörleri olarak bilinir (Li vd., 2009; Mubarak vd., 2021).

DVM, sınıflandırma ve regresyon için kullanılan güçlü ve esnek denetimli makine öğrenme algoritmasıdır. Sınıflar arasında kalan bölge margin olarak adlandırılır ve marginin genişliğiyle doğru orantılı olarak örnek veriler daha iyi bir şekilde sınıflandırılır (Kahraman, 2021).

DVM'nin avantajı, yerel minimumlardan etkilenmemesi ve destek vektörlerini kullanımı nedeniyle yüksek boyutluluk sorunundan etkilenmemesidir. (Syarif vd., 2016).

3.2.1.2 Çekirdek

DVM, doğrusal sınıflandırma ile mükemmel bir şekilde çalışır. Hiper düzlemleri ayırmanın kolay olmayacağı doğrusal olmayan sınıflandırma problemleri için doğrusal olmayan verileri yüksek boyutlu özellik uzayına eşlemek için çekirdek işlevi olarak bilinen özel bir işleve ihtiyaç vardır. Basit bir deyişle, çekirdek ayrılamayan sorunları daha fazla boyut ekleyerek ayrılabilir sorunlara dönüştürür (Somvanshi vd., 2016).

Bu çalışmada DVM için kullanılan çekirdek fonksiyonları sırasıyla ‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’, ‘precomputed’’dir. Temel DVM sınıflandırıcısında varsayılan çekirdek fonksiyon, *rbf*’tir (URL-1).

3.2.1.3 Düzenlileştirme

Düzenlileştirme parametresi (C), DVM optimizasyonunda her bir eğitim örneğini yanlış sınıflandırmaktan ne kadar kaçınmak istenildiğini ifade eder. Büyük C değerleri için bir hiperdüzlem tüm eğitim noktalarını doğru bir şekilde sınıflandırmak için daha iyi bir performans gösterirse optimizasyon daha küçük marjlı bir hiperdüzlem seçer. Bir hiperdüzlem daha fazla noktayı yanlış sınıflandırsa bile minimum bir C değeri optimize edicinin daha büyük marjlı bir ayırma hiperdüzlemi aramasını sağlar (URL-2, 2021). Temel DVM sınıflandırıcısında varsayılan C değeri 1.0’dır (URL-1, 2021).

3.2.1.4 Gama

Gama parametresi, düşük değerler 'uzak' ve yüksek değerler 'yakın' olmak üzere tek bir eğitim örneğinin etkisinin ne kadar uzağa ulaştığını belirtir. Düşük gama ile ayırma çizgisi belirlenirken gerçekçi ayırma çizgilerinden uzak konumlar dikkate alınır. Yüksek gama noktaları, hesaplamada makul çizgilere yakın noktaların dikkate alındığı anlamına gelir (URL-2, 2021). Gama parametresi, çekirdek işlevinin genişliğini veya eğimini tanımlar. Gama değeri düşük olduğunda karar sınırının 'eğrisi' çok düşük olur ve karar bölgesini çok geniş hale getirir. Gama yüksek olduğunda karar sınırının 'eğrisi' yükselir, bu da veri noktaları etrafında karar sınırı adaları oluşturur (Al-Mejibli vd., 2020). Bu çalışmada DVM için kullanılan gamma değerleri ‘scale’, ‘auto’ olup varsayılan değer, *scale*’dir (URL-1, 2021).

3.2.2 Çok Katmanlı Algılayıcı

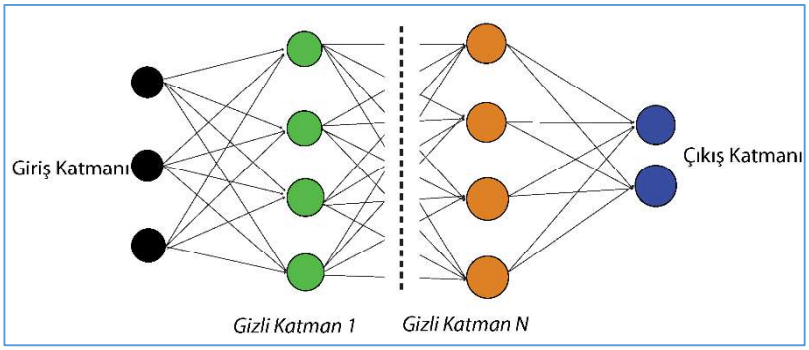
Yapay sinir ağları insan beyninin çalışmasından yola çıkılarak geliştirilmiş bir algoritmadır. İnsan beyninde birbirlerine snapslarla bağlı çok sayıda sinir hücresi vardır. Bu yapı, yapay sinir ağları ile modellenmiştir (Abiodun vd. 2018). Yapay sinir ağlarında nöronlar birbirlerine hiyerarşik olarak bağlı ve paralel olarak çalışabilen yapay hücrelerden oluşur (Öztemel, 2003). Bu hücreler arasındaki bağlantıların her

birinde öğrenilen bilgileri içeren değerler vardır. Bu hücrelerin birbirine bağlanmasıyla giriş katmanı, ara katman ve çıkış katmanı olan yapay sinir ağları ortaya çıkar.

Giriş katmanında ağdaki katmanlara gönderilecek giriş verileri bulunur. Ara katmanlarda girişten gelen veriler işlenir. Ağ tasarımı çerçevesinde farklı sayıda ara katmanlar modelde bulunabilir. Çıkış katmanındaki nöronlarda her giriş verisi için tahmin edilecek sınıflara ait değerler bulunur (Dongare vd., 2012).

Tet katmanlı algılayıcı ile giriş ve çıkış katmanları arasında doğrusal bir ilişki tespit edilmeye çalışılır. Ancak XOR problemi gibi doğrusal ilişkinin olmadığı problemleri çözmek için Rumelhart vd. Çok Katmanlı Algılayıcı (ÇKA) modelini geliştirmiştir (Rumelhart vd., 1985). Bu model sınıflandırma ve tanıma problemlerinde yüksek başarı sunar.

ÇKA, girdi veri setlerini bir dizi uygun çıktıya eşleyen ileri beslemeli bir yapay sinir ağı modelidir. Bir ÇKA, birden çok katmandan oluşur ve her katman bir sonrakine tam olarak bağlanır. Katmanların düğümleri, girdi katmanının düğümleri hariç doğrusal olmayan aktivasyon fonksiyonlarına sahip nöronlardır. Girdi ve çıktı katmanları arasında bir veya daha fazla doğrusal olmayan gizli katman olabilir (Sridevi vd., 2019). Çıkış katmanından önceki katmana kadar işlenen veriler işlenerek her giriş verisi için bir çıkış verisi belirlenir. Şekil 3.5, ÇKA'nın temsili bir formunu sunar.



Şekil 3.5 Çok Katmanlı Algılayıcı sınıflandırıcısının temsili gösterimi

3.2.2.2 Aktivasyon Fonksiyonu

Gizli katmanlardaki nöron çıkışları o nöron için tanımlanmış aktivasyon fonksiyonuna bağlı olarak elde edilir (URL-3, 2021). Birçok aktivasyon fonksiyonu olmakla birlikte, bu çalışmada ÇKA katmanları arasında kullanılan fonksiyonlar; *identity, logistic, tanh,* ve *relu*'dur.

Identity fonksiyonu: Bu fonksiyon, argümanı olarak kullanılan aynı değeri döndüren bir fonksiyondur. Eşitlik 3.1'de görüldüğü üzere, argümanı olarak kullanılan değeri döndürür yani çıktı girdi ile aynıdır (Chen ve Wang, 2020).

$$y = x \tag{3.1}$$

Hiperbolik tanjant fonksiyonu (tanh): ÇKA ağlarında yaygın olarak kullanılan -1 ile +1 aralığında olan simetrik s-şekilli bir fonksiyondur (Chen ve Wang, 2020). Eşitlik 3.2'de verilen tanh fonksiyonu, sigmoid işlevinin ikinci kusurunun üstesinden gelir ve optimizasyon sürecini kolaylaştırır (Zhu ve Song, 2020).

$$y = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{3.2}$$

Lojistik fonksiyon (Lojistik): Bu fonksiyon çıkışının 0 ile +1 aralığında olması nedeniyle tahn fonksiyonundan farklılık gösterir. Lojistik fonksiyonu Eşitlik 3.3'teki gibi ifade edilir (Chen ve Wang, 2020).

$$y = \frac{1}{1 + e^{-x}} \tag{3.3}$$

Rectified Lineer Unit (ReLU): Bu fonksiyon, 0 eşik değeriyle çalışır. Buna göre, $x < 0$ olduğunda 0 verir ve diğer durumda yani $x \geq 0$ olduğunda x değerini verir (Agarap, 2019). ReLU kaybolan gradyanı çözer ve yakınsamayı hızlandırır. ReLU, Eşitlik 3.4 ile ifade edilebilir (Zhu ve Song, 2020).

$$y = max\{0, x\} \tag{3.4}$$

3.2.2.3 Öğrenme Oranı

Öğrenme oranı, ağın eğitimi için gerekli örneklerin ağa gösterilmesi ve ardından her bir nörondaki ağırlık güncelleme miktarını denetler (Kaytan vd., 2020). Bu çalışmada ÇKA için kullanılan öğrenme oranı parametre değerleri *constant, invscaling* ve *adaptive* olup ÇKA temel modeli için varsayılan değer, *constant*'tır. Bu değerler aşağıda sırasıyla açıklanmıştır (URL-4, 2021):

'constant': 'learning_rate_init' tarafından verilen sabit bir öğrenme oranıdır.

'invscaling': Her 't' adımında öğrenme oranını kademeli olarak azaltır.

'adaptive': Eğitim kaybı azalmaya devam ettiği sürece öğrenme oranını "learning_rate_init" olarak sabit tutar.

3.2.2.4 Çözümleyici

Gizli katmanlardaki bulunan nöronlarda ağırlık optimizasyonu için çözücü kullanılır. Bu tez çalışmasında ÇKA için kullanılan çözücüler *lbfgs, sgd* ve *adam* olup varsayılan değer, *adam*'dır (URL-4, 2021):

Broyden–Fletcher–Goldfarb–Shanno (bfgs) kısıtlanmamış doğrusal olmayan optimizasyon problemlerini çözmek için yinelemeli bir yöntemdir (Fletcher, 1981). 'lbfgs', bfgs'nin sınırlı bellekli bir versiyonudur ve özellikle çok sayıda değişken içeren problemler için uygundur (Byrd vd., 1995).

Sgd (stochastic gradient descent - stokastik gradyan inişi) uygun düzgünlük özellikleriyle bir amaç fonksiyonunu optimize etmek için yinelemeli bir yöntemdir. Özellikle yüksek boyutlu optimizasyon problemlerinde çok yüksek hesaplama yükünü azaltan bir yöntemdir (Bottou ve Bousquet, 2012).

'adam', Kingma ve Ba tarafından önerilen stokastik gradyan tabanlı optimize edicidir. Bu metot, yalnızca az bellek gereksinimi olan birinci dereceden gradyanlar gerektiren verimli stokastik optimizasyon için bir yöntemdir (Kingma ve Ba, 2015).

3.3 Hiperparametre En İyilemesi

Makine öğreniminin temel amacı iyi çalışan ve doğru tahminler yapan bir model oluşturmaktır. Bu nedenle, makine öğrenimi algoritmasının optimize edilmesi gerekir. Makine öğreniminde optimizasyon, maliyeti en aza indirmek için bir optimizasyon tekniği kullanarak hiperparametreleri ayarlama işlemidir. Hiperparametre en iyilemesi daha iyi bir tahmin sonucuna ulaşmak adına bir modelin parametresi için tanımlı değer aralığından en uygun olanını tespit etmek üzere gerçekleştirilen arama sürecidir. Amaç, model parametreleri için en uygun değerlerin tespiti ve böylece makine öğreniminin iyileşmesidir. Diğer bir deyişle, parametre en iyilemesi, daha iyi bir performans elde etmek için varsayılan modelin parametrelerinin en iyilenmesini ifade eder.

Modelin yapısı hiperparametrelerle tanımlanır. Modelin parametre değerleri girdilere bağlı olarak değişir ve modelin eğitimi sırasında model parametre değerleri belirlenir. Bir sınıflandırma algoritmasında hiperparametreler tanımlanan varsayılan değerleri ile kullanılır. Bu parametreler için en uygun değeri bulmak daha iyi sonuçlar ve verimlilik sunar (Chon, 2019).

Parametre bir dizi değer içeren bir işlevin argümanıdır. Hiperparametre en iyilemesi, bir makine öğrenimi modelindeki herhangi bir parametre için geçerlidir (Hamida vd., 2020).

Bir model için en iyi parametre değerleri eğitim sırasında belirlenir. Bir modelin parametre değerleri girdilere bağlı olarak değişmektedir. Bir makine öğrenimi algoritmasının hiperparametreleri için uygun bir konfigürasyon elde etmek belirli bir uzmanlık veya deneme yanılma gerektirir. Bu hiperparametrelerin ayarlanması bir optimizasyon problemi olarak görülmektedir (Mantovani vd., 2015). Manuel arama ve otomatik arama yöntemleri iki tür hiperparametre en iyileme yaklaşımıdır. Manuel aramanın dezavantaları şu şekilde sıralanabilir (Wu vd., 2019):

a) Sonuçlar üzerinde en büyük etkiye sahip parametrelerin belirlenmesi uzmanlık gerektirir. Bu alanda uzman olmayanlar için zor bir süreçtir.

b) Hiperparametreleri değiştirme prosedürleri kolayca tekrarlanamaz.

c) Hiperparametrelerin sayısı ve değer aralığı büyüdükçe yüksek boyutlu verileri yönetmek giderek daha zor hale gelir.

Sınıflandırıcı algoritmanın performansını iyileştirmek için yürütülen deneysel çalışmalar uzun soluklu olduğu için deneme yanılma çalışmalarına kıyasla en iyileme yöntemleri tercih edilir. Ayrıca, deneme yanılma çalışmaları ile ortaya çıkacak hatalar veya kayıplar arama teknikleriyle en aza indirilir. Hiperparametreler genellikle olası parametre değerleri arasında bir tür verimli arama olan bir en iyileme metodu kullanılarak tahmin edilir.

3.3.1 Rastgele Arama

Makine öğreniminde ve özellikle hiperparametre en iyilemesinde, Rastgele Arama (RA), arama alanını rastgele veya olasılıkla örnekleyen bir en iyileme tekniği olarak karakterize edilebilir. RA, ızgara aramaya benzemekle birlikte bir model için en iyi çözümü bulmak için hiper-parametrelerin rastgele bir kombinasyonunun kullanıldığı bir tekniktir. RA, farklı hiperparametre kombinasyonları arasındaki korelasyonları kullanmaz (Zhang vd., 2021). Bu tür arama için çapraz doğrulama sürecindeki her yinelemede rastgele parametre kümeleri dikkate alınır ve tahmin edicinin parametreleri çapraz doğrulanmış arama parametresi ayarları ile optimize edilir (Li, 2020).

RA, birçok sınıflandırma probleminde sınıflandırıcı algoritmaların performansını iyileştirmek için oldukça sık kullanılan bir metottur. Örneğin, kalp hastalığının tespiti için yapılan iki farklı çalışmada yüksek başarı elde etmek için RA metodu kullanılmıştır (Javeed vd., 2019; Valarmathi ve Sheela, 2021). Bir başka çalışmada çeliğin çatlak derinliğinin tespitinde hiperparametre en iyilemesi için kullanılan metotlardan biri RA olmuştur (Chen vd., 2022). Sözde kod-1, RA algoritması adımlarını sunar (Rahmadayana ve Sibaroni, 2021).

Sözde Kod-1

1. Yineleme sayısı, k ve arama alanı belirlenir.
2. Arama alanında rastgele parametreler seçilir
3. Kesme kriterine ulaşana kadar aşağıdaki adımlar tekrarlanır.
    1. Arama alanındaki parametreler rastgele seçilir.
    2. Veri kümesini eşit olarak k-kata bölünür.
    3. Bir k katlamasındaki her k için:
        1. k test verisi, geri kalanı eğitim verisi olarak ayarlanır.
        2. Kullanılan parametrelerin kombinasyonu ile bir sınıflandırıcı eğitilir.
        3. Puanlamaya dayalı olarak model performansı değerlendirilir.
        4. Ortalama puan hesaplanır.
4. Ortalama çapraz doğrulama değerlendirme sonuçları, en iyi parametreler ve gerekli diğer ölçümler döndürülür.

3.3.2 Bayesian Arama

Bayesian optimizasyon algoritmasının ana fikri, sonsal dağılım gerçek dağılıma uyana kadar optimize edilmiş amaç fonksiyonunun iç yapısını ve matematiksel özelliklerini bilmeden amaç fonksiyonunun sonsal dağılımına sürekli olarak örnek noktalar eklemektir (Yin ve Li, 2022).

Stratejiye dayalı bilgilendirilmiş bir arama kullanarak daha kısa sürede en uygun hiperparametreleri bulma hedefiyle hiperparametrik ayar prosedürü giderek artarak daha fazla otomatik hale getirilmektedir. Amaç fonksiyonlarını optimize etmek için Bayesian optimizasyonu en iyi seçenektir (Zhang vd., 2021).

Bayesian Arama (BA), Bir "uyum" ve bir "puan" yöntemini ve diğer standart yöntemleri uygular. Tahmin edicinin parametreleri, parametre ayarları üzerinde çapraz doğrulanmış bir arama ile optimize edilir. Belirtilen dağılımlardan sabit sayıda parametre ayarı örneklenir (URL-5, 2021).

BA, birçok sınıflandırma probleminde sınıflandırıcı algoritmaların performansını iyileştirmek için oldukça sık kullanılan bir metottur. Örneğin, Xia vd., kredi puanlamasında aşırı gradyan artırma (XGBoost) sınıflandırıcınının performansını yükseltmek için Bayesian hiperparametre en iyilemesini kullanmışlardır (Xia vd., 2017). Bir başka çalışmada, CIFAR-10 veriseti için kullanılan modelin performansını artırmak için Bayesian hiperparametre en iyileme algoritmasını uygulamışlardır

(Victoria ve Maragatham, 2021). Bir diğer çalışmada, Zhang vd., yumuşak hassas killer için drenajsız kesme dayanımı ve çeşitli temel toprak parametreleri arasındaki ilişkileri yakalamak için XGBoost ve rastgele orman topluluğu öğrenme algoritmalarının parametreleri için BA ile en iyi değeri belirlemişlerdir (Zhang vd., 2021). Sözde kod-2, Bayesian optimizasyon adımlarını sunar (Wu vd., 2019).

Sözde Kod-2

1. For t=1,2, …
2. Kazanç fonksiyonunu u üzerinde fonksiyon f optimize edilerek x_t bulunur:
$$x_t = arg\max_x u(x|D_{1:t-1)}.$$
3. Amaç fonksiyonu örneklenir: $y_t = f(x_t)$.
4. Veriler, $D_{1:t} = \{D_{1:t-1}, (x_t, y_t)\}$ artırılır ve f fonksiyonunun sonsal bilgileri güncellenir.
5. Döngü sonu

Burada, $D_{1:t-1=\{x_n,y_n\}_{n=1}^{t-1}}$ f fonksiyonunun t −1 gözleminden oluşan eğitim veri kümesini temsil eder. Tüm algoritma, sonsal dağılımın güncellenmesi ve kazanç işlevinden oluşur. Gözlemler biriktikçe, sonsal dağılım sürekli olarak güncellenir; kazanç fonksiyonunun maksimize edildiği nokta bulunur ve eğitim veri setine eklenir. Tüm süreç, maksimum iterasyon sayısına ulaşılana kadar veya mevcut değer ile şimdiye kadar elde edilen optimal değer arasındaki fark, önceden tanımlanmış bir eşikten az olana kadar tekrarlanır (Wu vd., 2019).

3.3.3 Optuna En İyileyicisi

Akiba vd. (2019) tarafından tanıtılan Optuna En İyileyicisi (OEİ) özellikle makine öğrenimi için tasarlanmış otomatik ve hızlı bir hiperparametre en iyilemesi sunan açık kaynaklı bir Python kitaplığıdır.

OEİ, hiperparametreleri örneklemek için en yeni algoritmaları onaylayarak ve iyi sonuçlanmayan denemeleri verimli bir şekilde budayarak etkili hiperparametre en iyilemesi sağlar (Optuna Contributors, 2018a).

Örnekleme algoritmaları, kaydın önerilen parametre değerlerini ve değerlendirilen nesnel değerleri kullanarak arama alanını sürekli olarak daraltır ve daha iyi nesnel

değerlere yol açan parametreleri veren optimal bir arama alanı sunar (Optuna Contributors, 2018b).

OEİ ile yazılan kod oldukça modülerdir ve bu teknik ile hiperparametreler için dinamik olarak arama alanları oluşturabilir (Akiba vd., 2019). OEİ'nin önemli özellikleri aşağıda verilmiştir (URL-6, 2021).

1. Daha hızlı sonuçlar için geniş alanları verimli bir şekilde arar ve iyi sonuçlanmayan denemeleri budar.

2. Optimum hiperparametreler için otomatik arama yapar.

3. Kodu değiştirmeden birden çok iş parçacığı veya işlem üzerinde hiperparametre aramalarını paralel olarak gerçekleştirir.

Bir amaç fonksiyonunu maksimize etme veya minimize etme sürecinde bir dizi hiperparametre için OEİ doğrulama puanını döndürür. OEİ, iz nesnesi ile etkileşim yoluyla adım adım hedef işlevini oluşturur. İz nesnesi yöntemleri, amaç fonksiyonunun çalışma süresi boyunca arama alanını dinamik olarak belirler (Srinivas ve Katarya, 2022).

OEİ, birçok sınıflandırma probleminde sınıflandırıcı algoritmaların performansını iyileştirmek için oldukça sık kullanılan bir metottur. Örneğin, Srinivas ve Katarya (2022) Cleveland, kalp yetmezliği ve kalp hastalığı veri setleri üzerinde XGBoost sınıflandırıcısını OEİ ile optimize etmişlerdir. Lacerda vd., Covid-19 hastalığının tespiti için kullandıkları bir derin öğrenme modelini OEİ ile optimize etmişlerdir (Lacerda vd., 2021). Covid-19'u sınıflandırmak için yapılan bir başka çalışmada Barros vd., önerdikleri hibrit derin öğrenme modelinin hiperparamerelerini OEİ ile optimize etmişlerdir (Barros vd., 2021). Son olarak, Ekundayo elektrik enerjisi tüketimini tahmin etmek için önerdiği hibrit modelin hiperparamerelerini OEİ ile optimize etmiştir (Ekundayo, 2020).

3.4 Performans Ölçütleri

Bir modelin veya sınıflandırıcı algoritmanın kendisine verilen görevi ne kadar iyi yerine getirip getiremediğini görebilmek için performansının değerlendirilmesi gerekmektedir. Makine öğrenmesinde performans değerlendirmesi önemli bir husus olup, bu amaçla birçok performans değerlendirme kriteri geliştirilmiştir. Bu kriterlerin hesaplanmasında Tablo 3.2'de verilen karışıklık matrisi kullanılmaktadır. Buradaki, DP, doğru pozitif, DN doğru negatif, YP yanlış pozitif ve YN yanlış negatif anlamına gelir. Farklı modellerin sundukları karışıklık matrislerindeki YP ve YN toplamlarının DP ve DN toplamlarıyla karşılaştırılması yapılarak hangi modelin en iyi performansa sahip olduğu genellemesi yapılabilir.

Tablo 3.2 Karışıklık matrisi

		Tahmin	
		Negatif	Pozitif
Gerçek	Negatif	DN	YP
	Pozitif	YN	DP

Burada 1 pozitif sınıfı ve 0 negatif sınıfı temsil etmek üzere;

a) DP: Test örneğinin sınıfının gerçek değerinin 1 ve model tarafından tahmin edilen değerin 1 olduğu anlamına gelir.

b) DN: Test örneğinin sınıfının gerçek değerinin 0 ve modelin tahmin değerinin 0 olduğu anlamına gelir.

c) YP: Test örneğinin sınıfının gerçek değerinin 0 ve modelin tahmin değerinin 1 olduğu anlamına gelir.

d) YN: Test örneğinin sınıfının gerçek değerinin 1 ve modelin tahmin değerinin 0 olduğu anlamına gelir.

En sık kullanılan performans değerlendirme ölçütleri; Doğruluk (Accuracy), Kesinlik (Precision), Duyarlılık (Sensitiviy, Recall), F1-ölçütü (F1-score)'dür. Karışıklık

matrisinden faydalanılarak hesaplanan bu performans ölçütleri Eşitlik 3.5 ve Eşitlik 3.10 arasında verilmiştir.

$$Doğruluk = \frac{DP+DN}{DP+YN+DN+YP} \tag{3.5}$$

$$Kesinlik = \frac{DP}{DP+YP} \tag{3.6}$$

$$Duyarlılık\ (Doğru\ Pozitif\ Oranı) = \frac{DP}{DP+YN} \tag{3.7}$$

$$Özgüllük = \frac{DN}{DN+YP} \tag{3.8}$$

$$\text{F1} - \text{ölçütü} = 2 * \left(\frac{Kesinlik * Duyarlılık}{Kesinlik+Duyarlılık}\right) \tag{3.9}$$

$$Yanlış\ Pozitif\ Oranı = \frac{\text{YP}}{YP+DN} \tag{3.10}$$

Doğruluk: Girdi veya eğitim verilerine dayalı olarak bir veri setindeki değişkenler arasındaki ilişkileri ve kalıpları en iyi hangi modelin belirlediğini belirlemek için sıklıkla başvurulan bir ölçümdür ve sınıflandırıcının doğru tahmin ettiği örneklerin tüm örneklere oranını verir.

Kesinlik: Doğru pozitif sayısının doğru pozitiflerin ve yanlış pozitiflerin toplamına oranıdır.

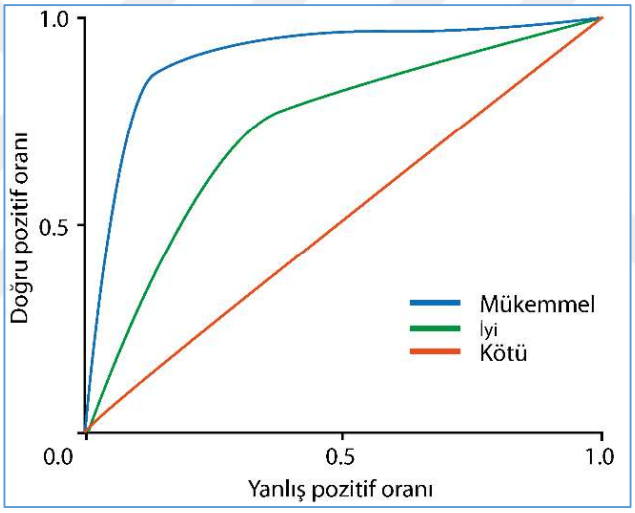
Duyarlılık: Doğru sınıflandırılmış pozitif etiketli örnek sayısının toplam pozitif örnek sayısına oranıdır. Bu metrik, literatürde Geri Çağırma veya Doğru Pozitif Oranı olarak da bilinir.

Özgüllük, doğru sınıflandırılmış negatif etiketli örnek sayısının toplam negatif örnek sayısına oranıdır.

F1-ölçütü: Duyarlılık ve Kesinlik değerlerinin harmonik ortalaması olarak ifade edilir.

Ayrıca, sınıflandırıcı algoritmaların performanslarını değerlendirmede kullanılan bir başka ölçüt, Eğri altındaki alan (EAA, Area Under Curve)'dır. Buradaki eğri, Alıcı İşlem Karakteristiği (AİK, Receiver Operating Characteristic (ROC)) eğrisidir. AİK eğrisi, gerçek pozitif oranına karşı yanlış pozitif oranı olarak ifade edilir ve çeşitli sınıflandırma eşikleri kullanılarak çizilir (Srinivas ve Katarya, 2022). Şekil 3.6'da örneği verilen AİK eğrisinin altında kalan alan olarak ifade edilen EAA metriği ile algoritmanın sınıflandırma başarısı değerlendirilir.

Bir olasılık eğrisi olan AİK'nde eğrinin x koordinatı YP'yi, y koordinatı DP'yi temsil eder. Modelin sınıfları ayırt etme başarısı ile EAA büyüklüğü doğru orantılıdır (Kahraman, 2021).



Şekil 3.6 AİK eğrilerinin türleri (Ferraris, 2018).

EAA'nın sol üst köşeye daha yakın olması doğrudan modelin başarısını gösterir. Bu metrik, 0 ile 1 arasında değerler alır ve aşağıda verilen derecelendirmeler kullanılarak yorumlanabilir (Ertorsun vd., 2009; Ferraris, 2018).

- 0,90 - 1 = mükemmel
- 0,80 - 0,90 = iyi
- 0,70 - 0,80 = orta
- 0,60 – 0,70 = zayıf
- 0,50 – 0,60 = başarısız

4. DENEYSEL ÇALIŞMALAR VE ELDE EDİLEN SONUÇLAR

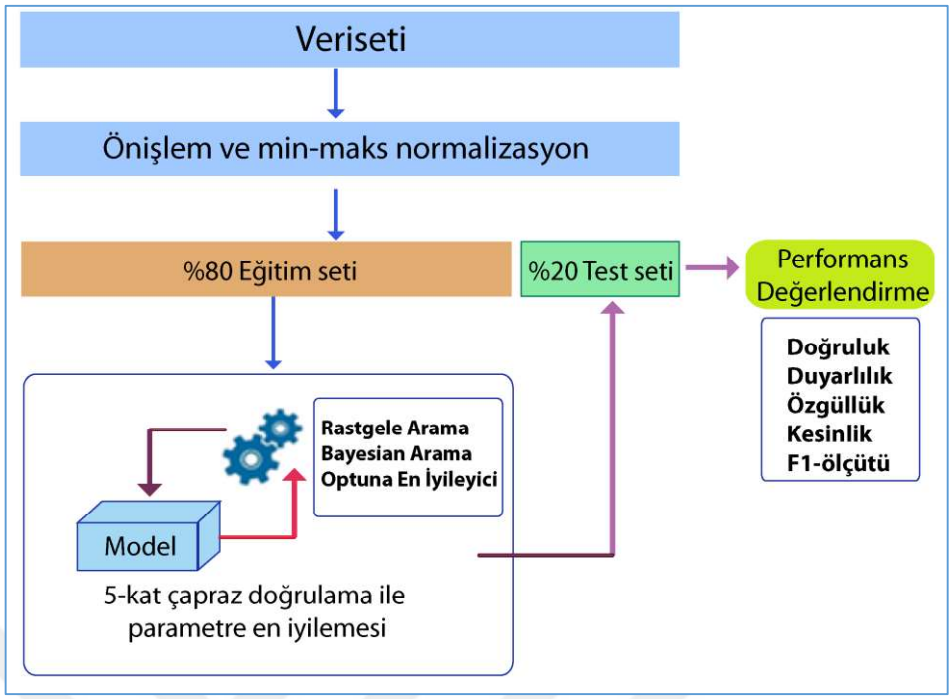
4.1 Deneysel Ortam

Bu tez çalışmasındaki deneyler, 8 GB RAM, Intel 8. nesil i7 işlemci ve Windows 10 işletim sistemine sahip x64 tabanlı bir dizüstü bilgisayar üzerine kurulu Anaconda Spyder platformu üzerinde yürütülmüştür. Kod geliştirme ortamında veri setlerinin yüklenmesi ve okunması işlemleri için *pandas* kütüphanesi kullanılmıştır. Dizi işlemleri için *numpy* kütüphanesi kullanılmıştır. Sınıflandırıcı algoritmalar için *scikit-learn* kütüphanesinden yararlanılmıştır. Son olarak, sonuçların analizi, karışıklıklık matrisi ve AİK çizimlerinde oldukça kullanışlı olan *matplotlib* kütüphanesi ile çalışılmıştır.

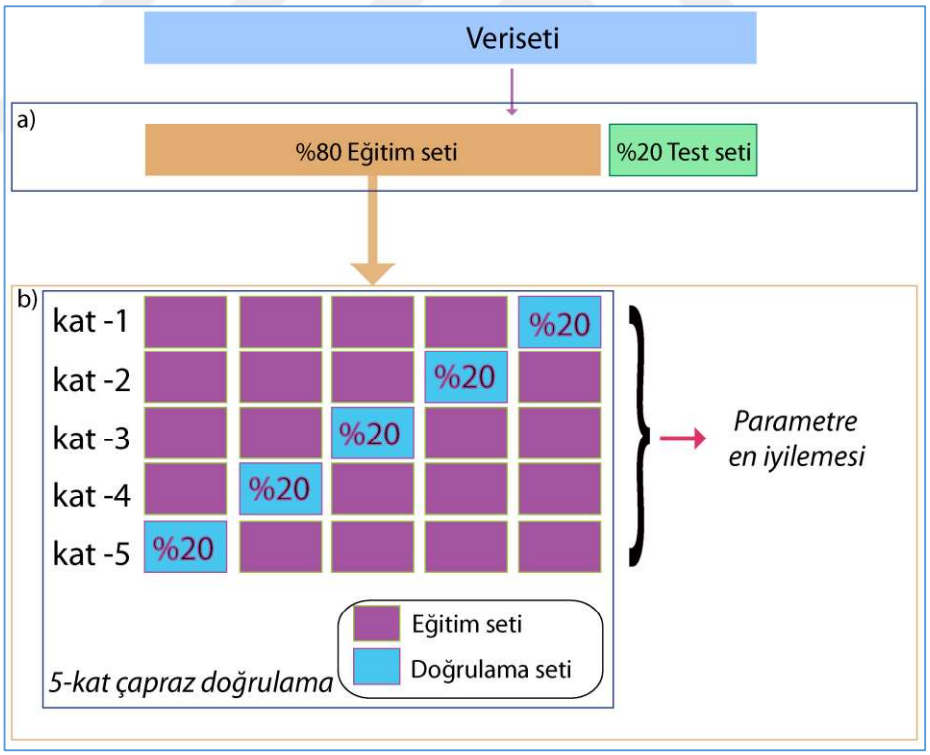
4.2 Model Eğitimi Ve Testi

Şekil 4.1, model eğitimi ve testi çalışmaların genel blok diyagramını sunar. Buna göre, ilk olarak, veri setini içeren Excel dosyası kod geliştirme ortamında açılmış, önişlemler gerçekleştirilmiş ve verisetlerindeki Yes/No gibi sayısal olmayan verilerin 1/0 kategorik değerlerine dönüştürülmüştür. Ardından, makine öğrenmesi aşamasına geçmeden önce veri seti üzerinde min-max normalizasyon tekniği uygulanarak verilerin 0 ve 1 aralığına ölçeklendirilmesi sağlanmıştır. Min-maks normalleştirme, orijinal veri değerleri arasındaki ilişkileri korur (Han vd., 2012). Eşitlik 4.1'de verilen bu normalizasyon tekniğinde; x mevcut değer, x_{min} minimum değer, x_{max} maksimum değer olup, x_{yeni} normalize edilmiş değerdir.

$$x_{yeni} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{4.1}$$



Şekil 4.1 Bu çalışmanın genel blok diyagramı



Şekil 4.2 a) Eğitim ve test setleri, b) Hiperparametre en iyilemesi için 5-kat çapraz doğrulama

Şekil 4.2 (a)'da gösterildiği gibi veri setleri eğitim ve test veri kümesi olmak üzere iki kümeye ayrılmıştır. Temel DVM ve temel ÇKA modellerinin eğitiminde verisetlerinin

%80'lik kısmı kullanılırken, bu sınıflandırıcıların performans testinde kalan %20'lik kısım kullanılmıştır. Ardından, makine öğrenimi algoritmaları eğitim setinde eğitilmiş ve bu algoritmaların performansları test seti üzerinde test edilmiştir. Sonraki aşamada, bu verisetleri üzerinde sınıflandırma modellerinin en iyi performansını sunması için en iyileme yöntemleri uygulanmıştır. Hiperparametre ayarında çapraz doğrulama, algoritmanın verimliliğini artırmak için hiperparametrelerin optimal değerini bulmaya yardımcı olur. Her bir en iyileme yöntemi ile sınıflandırıcı algoritmaların parametre değerlerinin en iyilenmesi için Şekil 4.2(b)'de gösterildiği üzere her bir eğitim seti üzerinde 5-kat çapraz doğrulama uygulanmıştır. Belirlenen en iyi parametre değerlerine sahip modelin performansı test seti üzerinde ölçülmüştür. Buna göre, eğitim veri setleri üzerinde DVM ve ÇKA sınıflandırıcılarının yer aldığı modellerin eğitimleri gerçekleştirilmiştir. Sonrasında, her bir test veri seti için bu modeller ile elde edilen sonuçlar ve ortalamaları ilgili tablolarda sunulmuştur. Dahası, her bir deneyde elde edilen karışıklık matrisleri ve AİK çizimleri ilgili şekillerde sunulmuştur. Sonuç olarak, sınıflandırıcı algoritmaların başarıları performans ölçütleri çerçevesinde ele alınmış ve kapsamlı olarak karşılaştırılmıştır.

Deneysel çalışmalar için ifade edilebilecek bir diğer önemli nokta, eğitim ve test verisetlerinin dağılımından kaynaklanabilecek aşırı uyum problemini önlemek için dışarıda tutma tekniği ile farklı eğitim ve test setleri üzerinde 10 kez denemeler yapılmıştır. Diğer bir deyişle, en yüksek tahmin doğruluğunu sağlamak ve algoritmanın tahmin edebilirliğini eğitim yoluyla kanıtlamak için dışarıda tutma yöntemi 10 kez uygulanmış olup her bir deneyde Veri setinin rastgele %80 ve %20' lik kısımları sırasıyla eğitim ve test için ayrılmıştır. Her bir veri setinde bu iş akışı uygulanarak tüm en iyileme yöntemleri ile sınıflandırıcı algoritmaların parametrelerinin en iyi değerleri tespit edilmiştir. Elde edilen sonuçlar literatürde oldukça iyi bilinen doğruluk, kesinlik, duyarlılık, özgüllük, F1-score ve EAA performans ölçütleriyle değerlendirilmiş ve karşılaştırılmıştır. Deneysel çalışmalarda oluşturulan modellerin bilgileri Tablo 4.1'de özetlenmiştir.

Tablo 4.1 Oluşturulan modellerin listesi

Hiperparametre en iyileme	Model
Hayır	Temel DVM
	Temel ÇKA
Evet	RA+DVM
	RA+ÇKA
	BA+DVM
	BA+ÇKA
	OEİ+DVM
	OEİ+ÇKA

4.3 Deneysel Sonuçlar

Bu çalışmada, DVM ve ÇKA sınıflandırıcıları için en iyileme metotları ile belirlenen en iyi parametre değerleri ile eğitilen modellerin test setlerindeki performansları değerlendirilmektedir. Bu sınıflandırıcıları optimize etmek için RA, BA ve OEİ yöntemleri kullanılmıştır. Bu en iyileme metotlarından ilk ikisi için scikit-learn kitaplığından, OEİ için ise Optuna kitaplığından faydalanılmıştır. Hiperparametre en iyilemesi için ÇKA sınıflandırıcısında gizli katman için *activation* fonksiyonu, ağırlık optimizasyonu için *solver* parametresi ve ağırlık güncellemeleri için *learning rate* parametresi, DVM sınıflandırıcısında *kernel, gama* ve *C* parametreleri ince ayar parametreleri olarak belirlenmiştir. DVM ve temel ÇKA sınıflandırıcıları için bu hiperparametreler sklearn kitaplığındaki varsayılan değerleri ile kullanılmıştır. Buna göre, temel DVM modelindeki C değeri 1.0, çekirdek fonksiyonu 'rbf' ve gamma değeri 3'tür. Temel ÇKA modelindeki gizli katman sayısı 100, aktivasyon fonksiyonu 'relu', çözücü 'adam' ve öğrenme oranı parametresi 'constant' değerine sahiptir.

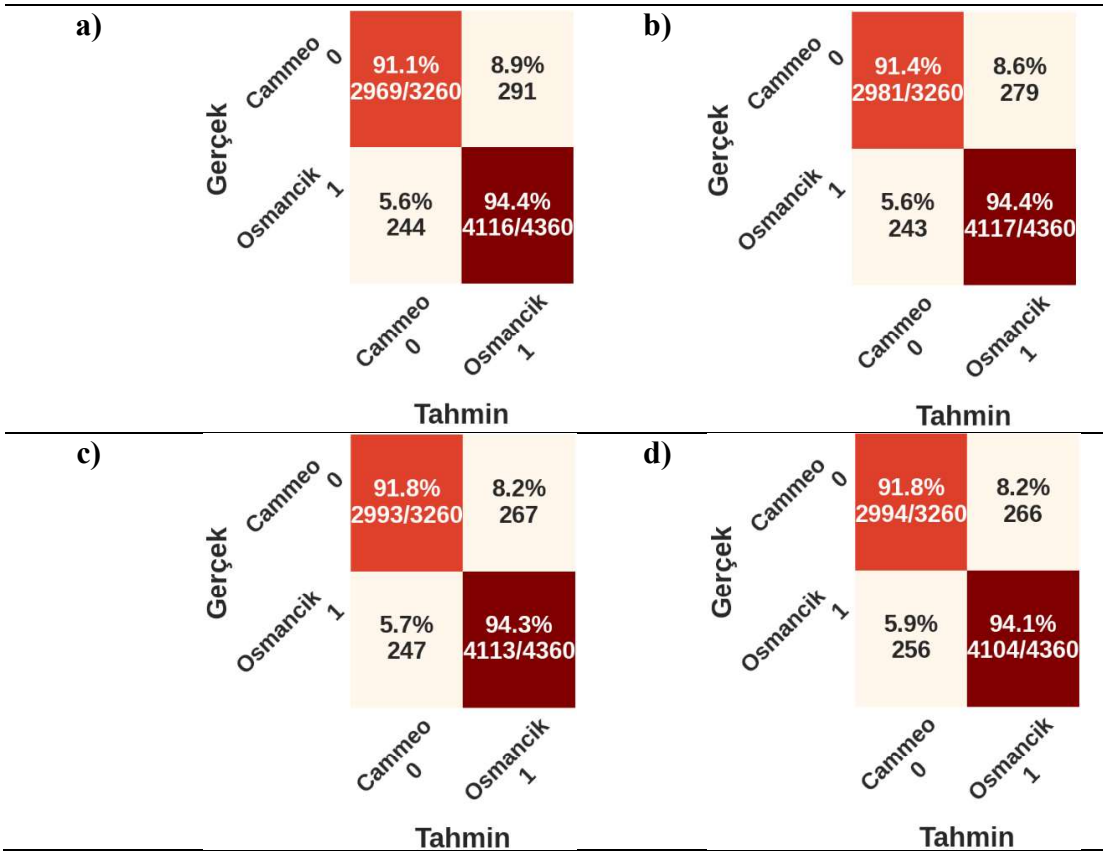
Her bir veri seti üzerinde yapılan deneylerde sınıflandırıcıların eğitimi için gereken çalışma zamanı ortalaması, her bir en iyileme metodu ile her bir veri setinde en iyi sonucun elde edildiği deney için DVM ve ÇKA'nın hiperparametre değerleri ve ayrıca bu sınıflandırıcıların sınıflandırma performansları ilgili tablolarda sunulmuştur. En iyi parametre değerlerinin elde edildiği deney numarası da bu parametre değerlerinin yer aldığı tablolarda ayrıca sunulmuştur. Bunun yanı sıra, DVM ve ÇKA sınıflandırıcı algoritmaları ve üç en iyileme metodu ile inşa edilen modellerin karışıklık matrisleri ve AİK grafikleri sunulmuştur. AİK çizimlerindeki etiket bilgilerinde yer alan 'Exper.'

ve 1 ile 10 arasındaki sayı değeri ilgili deneyi ifade eder. Örneğin, ‘Exper. 1’, 1 nolu deneyi ifade etmektedir.

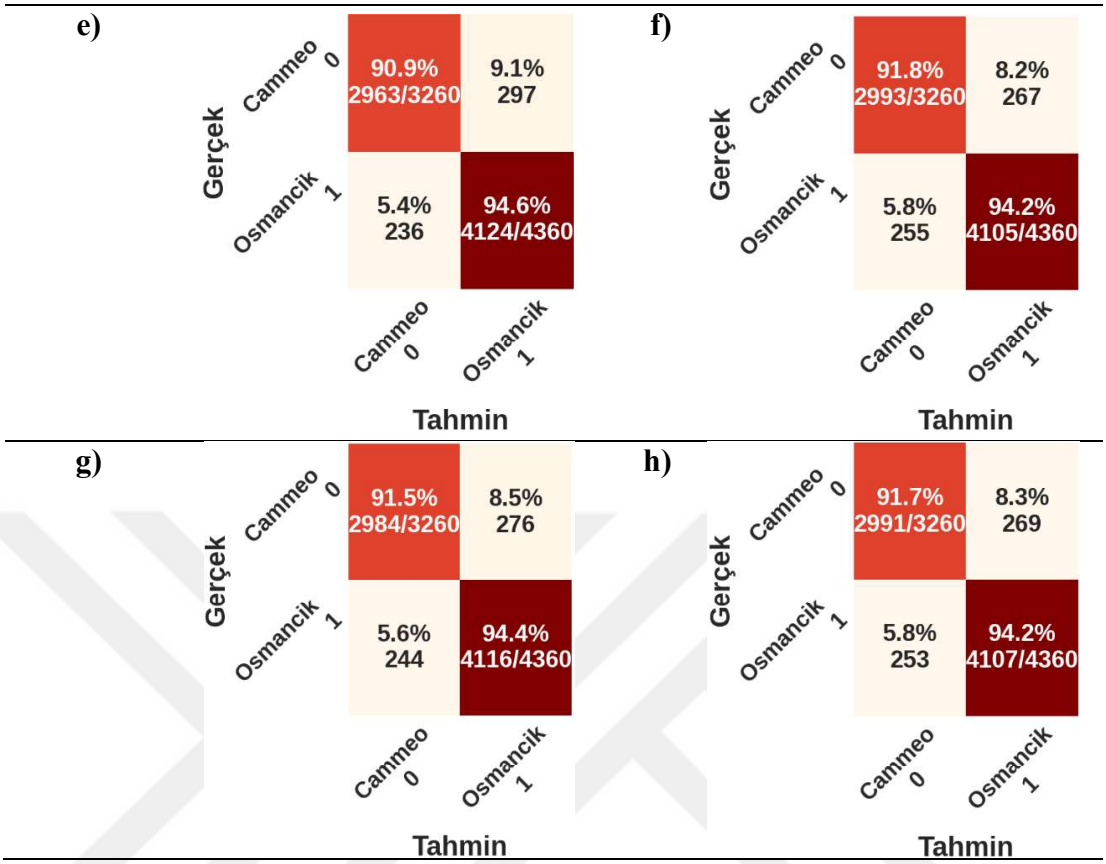
4.3.1 Veri Seti-1 İçin Elde Edilen Sonuçlar

Tablo 4.2, Veri seti-1 için her bir sınıflandırıcı ve en iyileme yöntemleri ile yapılan 10 deney ile elde edilen karışıklık matrislerinin üst üste bindirilmiş sonuç matrislerini sunar. Buna göre, RA, BA ve OEİ en iyileme metotlarının yer aldığı DVM modelleri temel DVM modeline kıyasla daha az hatalı sınıflandırma yapmıştır. Temel model 535 örneği yanlış sınıflandırma yaparken RA+DVM, BA+DVM ve OEİ+DVM sırasıyla 514, 533 ve 520 örneği yanlış sınıflandırmıştır. RA, BA ve OEİ en iyileme metotlarının yer aldığı ÇKA modelleri temel ÇKA ile aynı sayıda hatalı sınıflandırma yapmıştır. Bu veri seti üzerinde yapılan deneysel çalışmalarda sadece DVM sınıflandırıcısı üzerinde tüm en iyileme metotlarının olumlu bir etkisi olduğu görülmüştür.

Tablo 4.2 Veri seti-1 için her bir modelin örtüşen karışıklık matrisleri; a) Temel DVM b) Temel ÇKA, c) RA+DVM d) RA+ÇKA e) BA+DVM f) BA+ÇKA g) OEİ+DVM h) OEİ+ÇKA



Tablo 4.2 Devamı



Tablo 4.3 Veri seti-1 üzerinde her bir model ile yapılan 10 deneyin ortalama sonuçları

Sınıflandırıcı	En iyileme	Doğruluk (%)	Özgüllük (%)	Duyarlılık (%)	Kesinlik (%)	F1-ölçütü (%)
DVM	Hayır	92,98	91,07	94,4	93,4	93,9
	RA	93,25	91,81	94,33	93,91	94,12
	BA	93,01	90,89	94,59	93,29	93,93
	OEİ	93,18	91,53	94,4	93,72	94,06
ÇKA	Hayır	93,15	91,44	94,43	93,67	94,04
	RA	93,15	91,84	94,13	93,92	94,02
	BA	93,15	91,81	94,15	93,9	94,02
	OEİ	93,15	91,75	94,2	93,86	94,03

Tablo 4.3, Veri seti-1 üzerinde tüm modeller ile yapılan deneylerin ortalama sonuçlarını sunar. Görüldüğü üzere, hiperparametre değeri en iyilemesi ile en iyi doğruluk değeri DVM'de 92,98'den %93,25'e yükselmiştir. Burada DVM için RA, BA ve OEİ en iyileme metotlarının pozitif etkisi olduğu görülmektedir. Diğer yandan, temel ÇKA'ya kıyasla tüm en iyileme metotları ile daha iyi doğruluk elde

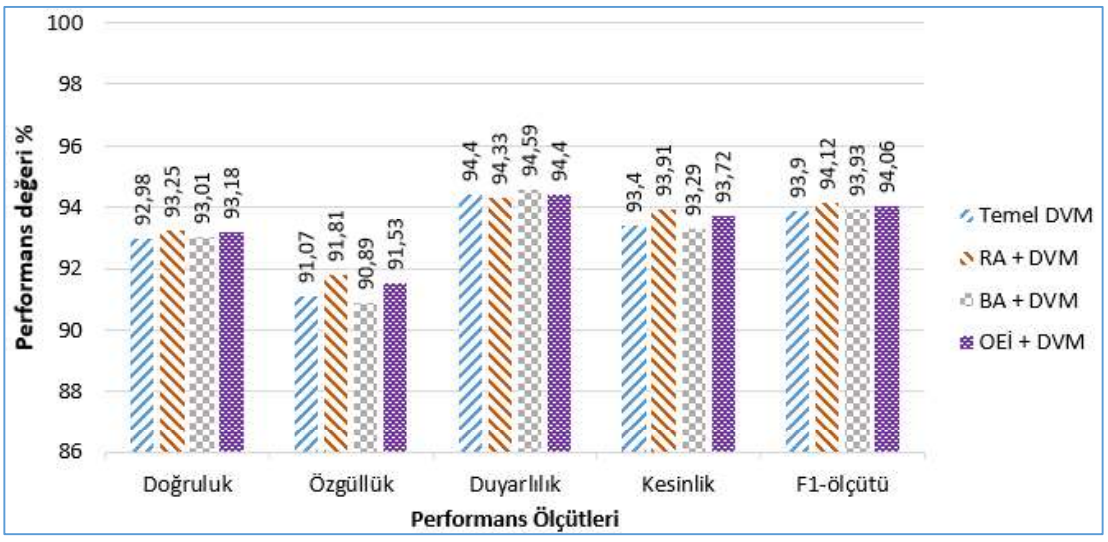
edilememiştir. Özgüllük metriği için RA ve OEİ en iyileme metotlarının DVM'nin performansında iyileşme sağladığı görülmüştür. ÇKA için ise RA, BA ve OEİ en iyileme metotlarının az da olsa iyileşme sağladığı görülmüştür.

Duyarlılık metriği için DVM'nin performansında sadece BA metodunun az da olsa iyileşme sağladığı görülmüştür. Bunun yanı sıra, ÇKA'nın sunduğu duyarlılık ölçümlerinde üç en iyileme yönteminin de bir iyileşme sunmadığı görülmüştür.

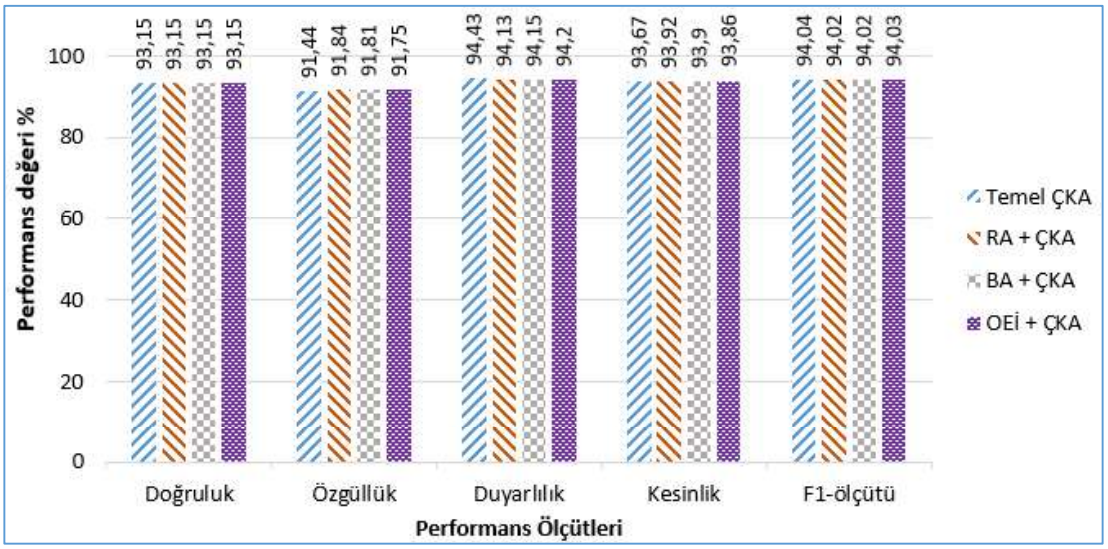
Kesinlik metriği için RA ve OEİ metotlarının DVM'nin performansında iyileşme sağladığı görülmüştür. Buna göre, kesinlik değeri %93,4 olan temel DVM'nin performansını RA ve OEİ metotları sırasıyla %93,91 ve 93,72'ye yükseltmişlerdir. BA+DVM'nin sunduğu kesinlik değeri %93,29 ile temel DVM'ye göre daha azdır. ÇKA'nın sunduğu kesinlik ölçümlerinde üç en iyileme yönteminin de az da olsa olumlu etkisinin olduğu görülmüştür. Buna göre, kesinlik değeri %93,67 olan temel ÇKA'nın performansını RA, BA ve OEİ metotları sırasıyla %93,92, %93,9 ve 93,86'ya yükseltmişlerdir.

F1 ölçütü için RA, BA ve OEİ metotlarının DVM'nin performansında iyileşme sağladığı, ÇKA'nın performansında ise bu üç en iyileme yönteminin pozitif etkisinin olmadığı görülmüştür.

Şekil 4.3 ve Şekil 4.4 elde edilen sonuçları grafiksel olarak sunmaktadır.



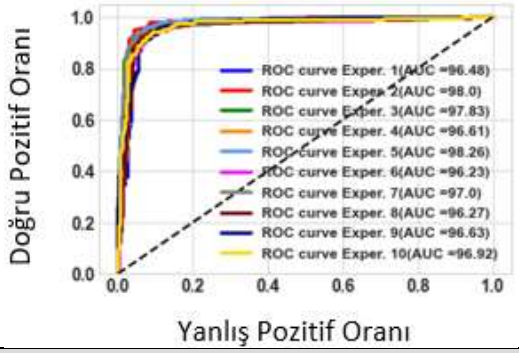
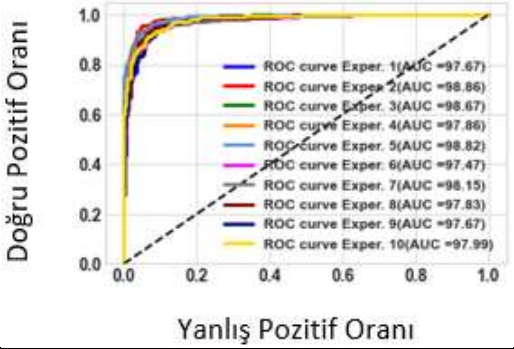
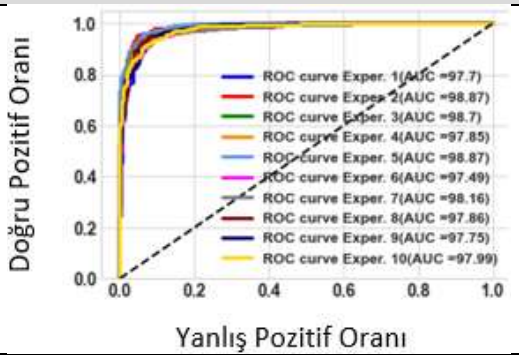
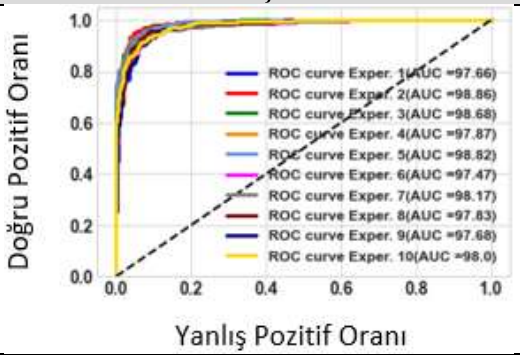
Şekil 4.3 Veri seti-1 için DVM sonuçları



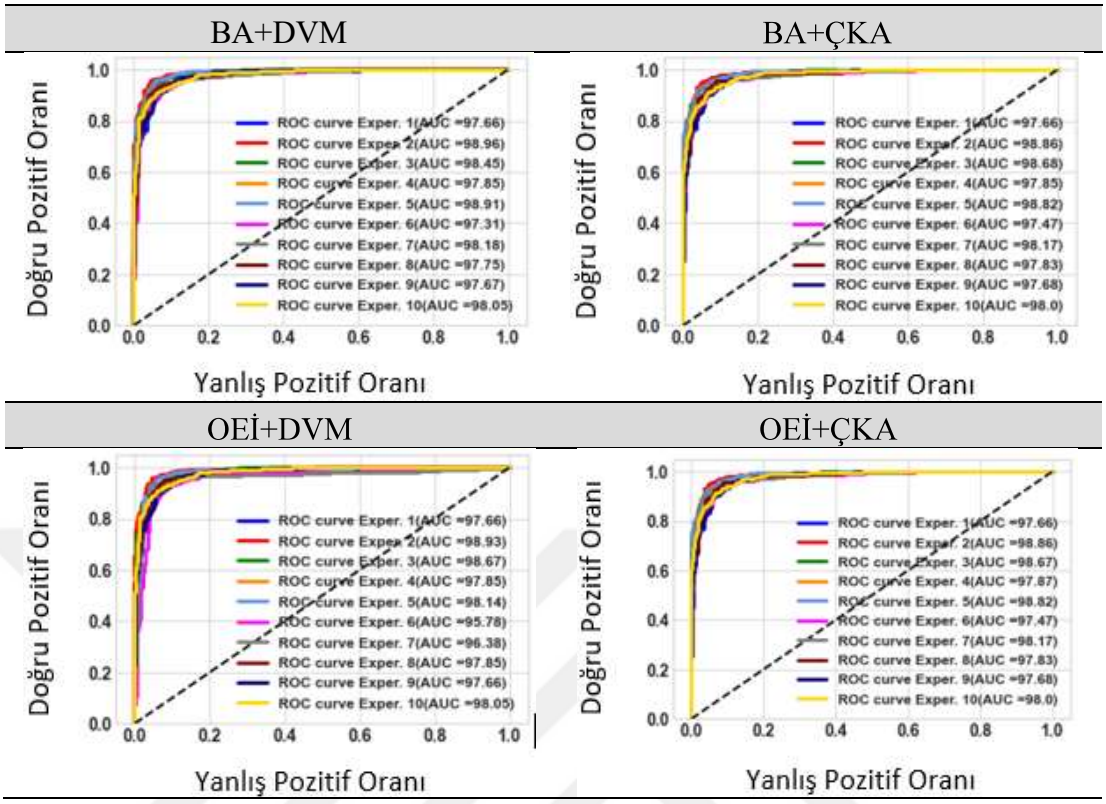
Şekil 4.4 Veri seti-1 için ÇKA sonuçları

Tablo 4.4, Veri seti-1 için oluşturulan modellere ait AİK çizimlerini sunar. Her bir satırda iki şekil bulunmaktadır. Bu şekiller sırasıyla DVM ve ÇKA sınıflandırıcılarına aittir. Her AİK çiziminin üzerinde ilgili model bilgisi bulunmaktadır. Örneğin, 1. satır 2. sütundaki AİK çizimleri temel ÇKA modeline aittir. Her bir AİK çizimlerinde bir modele ait 10 deneyin her birinden elde edilen sonuçlar bir arada verilmiştir.

Tablo 4.4 Veri seti-1 için her bir model ile yapılan 10 deneye ait AİK çizimleri

DVM	ÇKA
	
RA+DVM	RA+ÇKA
	

Tablo 4.4 Devamı



Tablo 4.5, 10 deneyden elde edilen EAA değerlerinin ortalamasını sunar. Bu değerler, en iyileme metotlarının sınıflandırıcı algoritma üzerindeki sınıflandırma etkisini özetlemektedir. Buna göre, Veri seti-1 için %98,124 EAA ile RA+DVM modeli diğer DVM modellerine kıyasla iyi bir performans göstermiştir. Bunun yanısıra, %98,104 ile RA+ÇKA sınıflandırıcısı ÇKA olan modeller içinde en yüksek EAA değerine sahiptir. Sonuç olarak, Veri seti-1 için RA en iyileme yöntemi ile inşa edilen DVM ve ÇKA sınıflandırıcıları diğer modellere kıyasla daha iyi bir performans göstermiştir. Bu sonuçlar, en iyileme metotlarının sınıflandırıcı algoritma üzerindeki pozitif etkisini ayrıca göstermektedir.

Tablo 4.5 Veri seti-1 için her bir model ile yapılan 10 deneyin EAA değerlerinin ortalaması

Hiperparametre en iyileme	DVM (%)	ÇKA (%)
Hayır	97,023	98,099
RA	98,124	98,104
BA	98,079	98,102
OEİ	97,697	98,103

Tablo 4.6, en iyileme metotlarının en iyi doğruluk değeri sunduğu deney için parametre değerlerini ve deney numarasını sunar. Temel sınıflandırıcıların varsayılan parametre bilgileri de ayrıca son sütunda verilmiştir.

Tablo 4.6 Veri seti-1 için en iyi hiperparametre değerleri

Sınıflandırıcı	Hiperparametre	RA	BA	OEİ	Varsayılan değer
DVM	C kernel Gamma	50 rbf auto	1,0 poly scale	45,62 linear auto	1,0 rbf scale
	Deney no.	2	2	2	2
ÇKA	activation learning rate solver	logistic adaptive Adam	logistic invscaling Adam	logistic adaptive Adam	relu constant Adam
	Deney no.	2	2	2	2

Tablo 4.7, tüm modellerinn deneysel çalışmalardaki ortalama yürütme sürelerini gösterir. Buna göre, varsayılan parametreleri ile bir sınıflandırıcının eğitimi için geçen zaman oldukça azdır. RA en iyileme metodu ile inşa edilen modellerin eğitimi için geçen zaman diğer en iyileme metotlarına kıyasla daha az olup bu yöntemi sırasıyla BA ve OEİ takip etmektedir.

Tablo 4.7 Veri seti-1 üzerinde her bir modelin eğitimi için geçen ortalama süre (sn)

Sınıflandırıcı	Temel Model	RA	BA	OEİ
DVM	3,37	79,90	235,23	432,29
ÇKA	87,84	481,01	828,68	1685,65

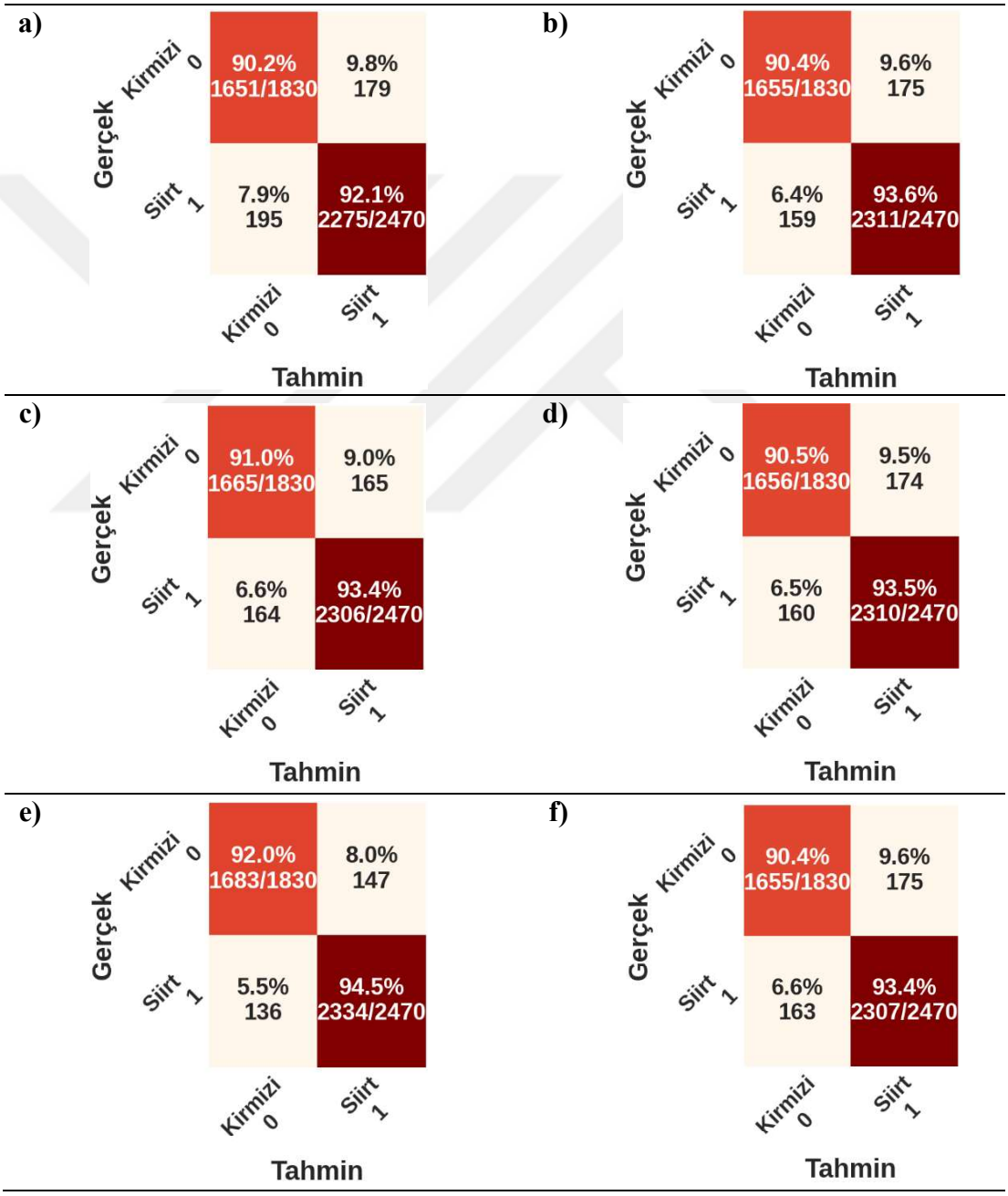
sn: saniye

4.3.2 Veri Seti-2 İçin Elde Edilen Sonuçlar

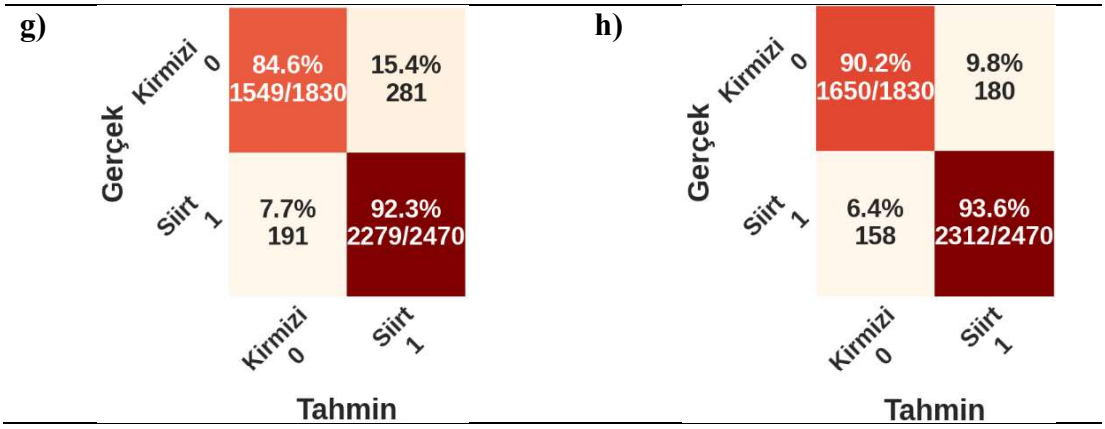
Tablo 4.8, Veri seti-2 için her bir sınıflandırıcı ve en iyileme yöntemleri ile yapılan 10 deney ile elde edilen karışıklık matrislerinin üst üste bindirilmiş sonuç matrislerini sunar. Buna göre, RA ve BA en iyileme metotlarının yer aldığı DVM modelleri temel DVM modeline kıyasla daha az hatalı sınıflandırma yapmıştır. Temel model 374 örneği yanlış sınıflandırma yaparken RA+DVM ve BA+DVM sırasıyla 329 ve 283 örneği yanlış sınıflandırmıştır. OEİ metodu, temel DVM sınıflandırıcısına kıyasla daha fazla örneği yanlış sınıflandırmıştır. Ayrıca, BA ve OEİ metotlarının yer aldığı

ÇKA modelleri temel ÇKA ve RA+ÇKA modellerine kıyasla daha fazla hatalı sınıflandırma yapmıştır. Bu veri seti üzerinde yapılan deneysel çalışmalarda sadece DVM sınıflandırıcısı üzerinde RA ve BA en iyileme metotlarının olumlu etkisinin olduğu görülmüştür.

Tablo 4.8 Veri seti-2 için her bir modelin örtüşen karışıklık matrisleri; a) Temel DVM b) Temel ÇKA, c) RA+DVM d) RA+ÇKA e) BA+DVM f) BA+ÇKA g) OEİ+DVM h) OEİ+ÇKA



Tablo 4.8 Devamı



Tablo 4.9 Veri seti-2 üzerinde her bir model ile yapılan 10 deneyin ortalama sonuçları

Sınıflandırıcı	En iyileme	Doğruluk (%)	Özgüllük (%)	Duyarlılık (%)	Kesinlik (%)	F1-ölçütü (%)
DVM	Hayır	91,3	90,22	92,11	92,72	92,4
	RA	92,35	90,98	93,36	93,33	93,34
	BA	93,42	91,97	94,49	94,09	94,29
	OEİ	89,02	84,64	92,27	89,02	90,62
ÇKA	Hayır	92,23	90,44	93,56	92,97	93,26
	RA	92,23	90,49	93,52	93,01	93,26
	BA	92,14	90,44	93,4	92,96	93,17
	OEİ	92,14	90,16	93,6	92,8	93,19

Tablo 4.9, Veri seti-2 üzerinde tüm modeller ile yapılan deneylerin ortalama sonuçlarını sunar. Görüldüğü üzere, hiperparametre değeri en iyilemesi ile en iyi doğruluk değeri DVM'de 91,3'ten %93,42'ye yükselmiştir. Burada temel DVM'ye kıyasla daha başarılı sonuçları sunan modellerin parametreleri RA ve BA ile elde edilmiştir. OEİ'nin temel DVM modelinin başarısını arttırmadığı görülmüştür. Özet olarak, DVM'nin RA ve BA yöntemleri ile temel DVM'ye kıyasla daha iyi doğruluk sunduğu ifade edilebilir. Diğer yandan, ÇKA için tüm en iyileme yöntemleri ile varsayılan parametrelere kıyasla daha iyi doğruluk elde edilmemiştir.

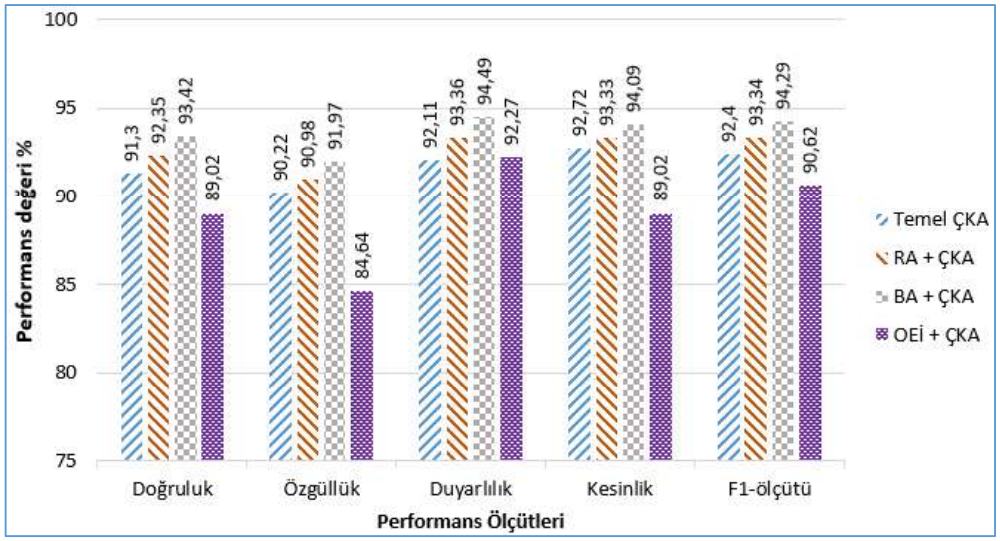
Özgüllük metriği için RA ve BA en iyileme metotlarının DVM'nin performansında iyileşme sağladığı, ÇKA için ise RA'nın çok az iyileşme sağladığı, BA ve OEİ en iyileme yöntemleri ile bir iyileşme sağlanamadığı görülmüştür. Duyarlılık metriği için tüm en iyileme metotlarının DVM'nin performansında az da olsa iyileşme sağladığı

görülmüştür. Bunun yanı sıra, ÇKA'nın sunduğu duyarlılık ölçümlerinde üç en iyileme yönteminden sadece OEİ'nin %0,04 fark ile çok az düzeyde pozitif etkisinin olduğu görülmüştür.

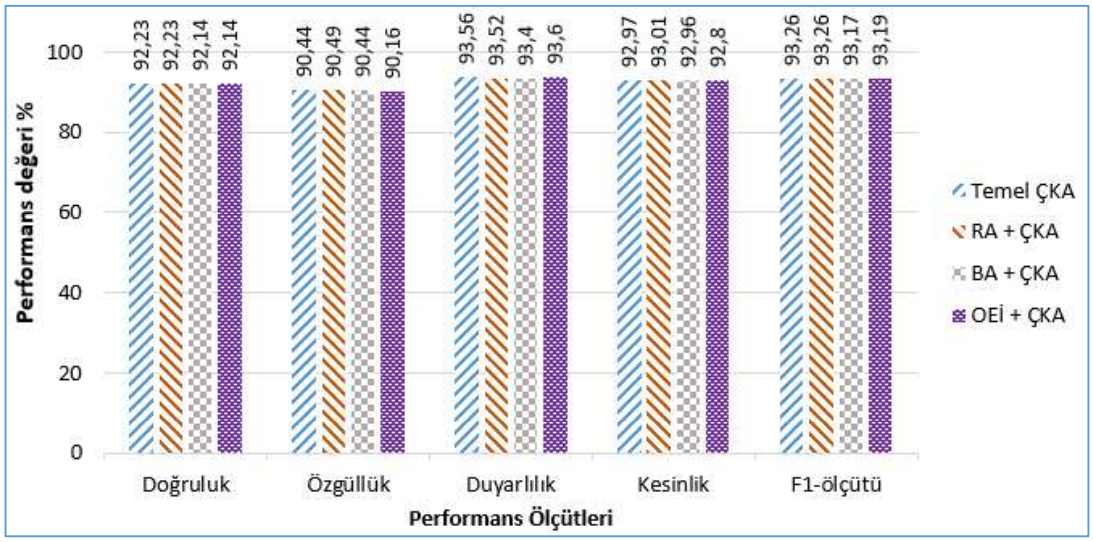
Kesinlik metriği için RA ve BA metotlarının DVM'nin performansında iyileşme sağladığı görülmüştür. Buna göre, kesinlik değeri %92,72 olan temel DVM'nin performansını RA ve BA metotları sırasıyla %93,33 ve 94,09'a yükseltmişlerdir. OEİ+DVM'nin sunduğu kesinlik değeri %89,02 ile temel DVM'ye göre daha azdır. Bunun yanı sıra, ÇKA'nın sunduğu kesinlik ölçümlerinde üç en iyileme yönteminden sadece RA'nın %0,04 fark ile çok az düzeyde pozitif etkisinin olduğu görülmüştür.

F1 ölçütü için RA ve BA metotlarının DVM'nin performansında iyileşme sağladığı, OEİ+DVM'nin temel DVM'ye kıyasla daha düşük bir değer sunduğu görülmüştür. Bunun yanı sıra, bu metrik için ÇKA'nın performansında üç en iyileme yönteminin pozitif etkisinin olmadığı görülmüştür.

Şekil 4.5 ve Şekil 4.6 elde edilen sonuçları grafiksel olarak sunmaktadır.



Şekil 4.5 Veri seti-2 için DVM sonuçları

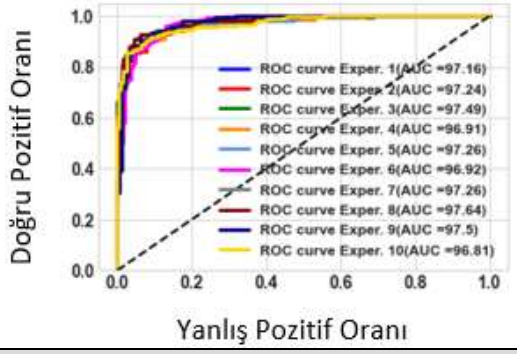
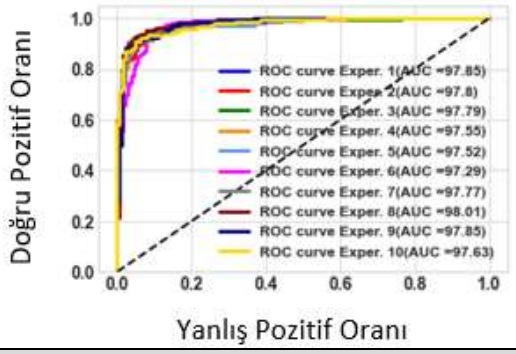
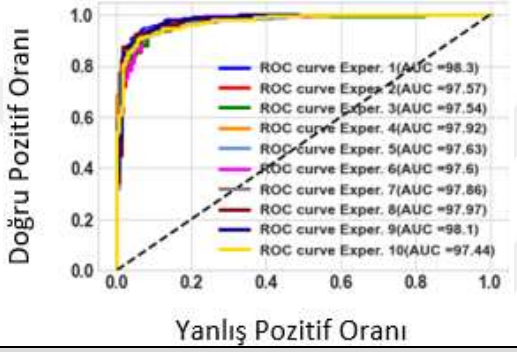
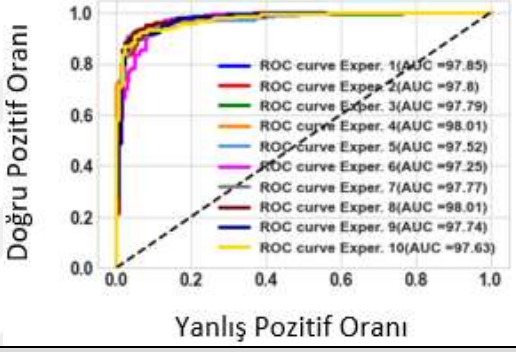
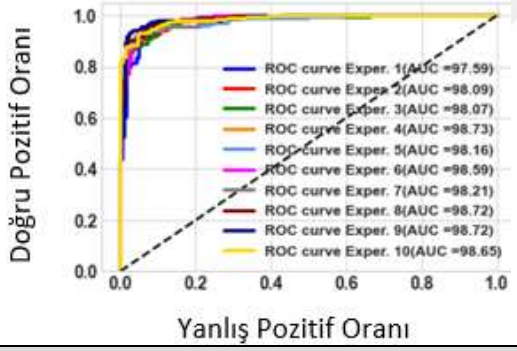
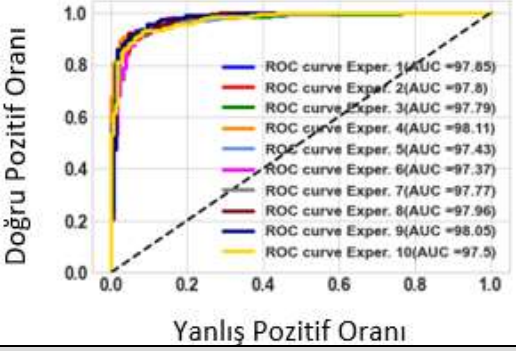
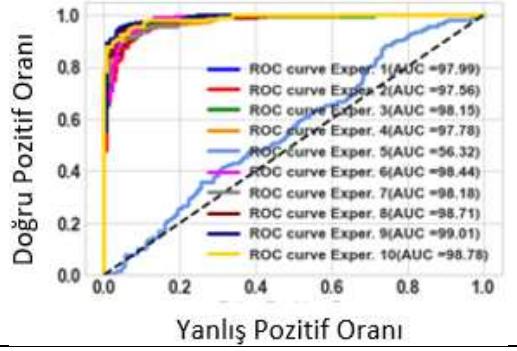
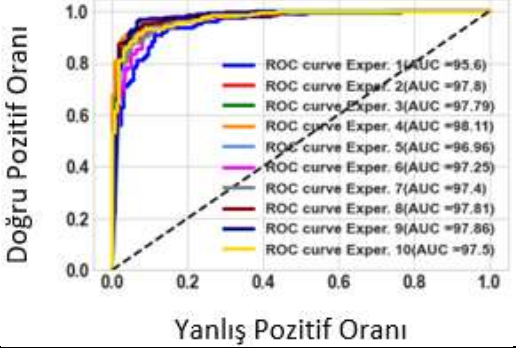


Şekil 4.6 Veri seti-2 için ÇKA sonuçları

Tablo 4.10, Veri seti-2 için oluşturulan modellere ait AİK çizimlerini sunar. Her bir satırda iki şekil bulunmaktadır. Bu şekiller sırasıyla DVM ve ÇKA sınıflandırıcılarına aittir. Her AİK çiziminin üzerinde ilgili model bilgisi bulunmaktadır. Örneğin, 3. satır 2. sütundaki AİK çizimleri BA+ÇKA modeline aittir. Her bir şekilde bir modele ait 10 deneyin her birinden elde edilen AİK çizimleri bir arada verilmiştir.

Tablo 4.11, 10 deneyden elde edilen EAA değerlerinin ortalamasını sunar. Bu değerler, en iyileme metotlarının sınıflandırıcı algoritma üzerindeki sınıflandırma etkisini özetlemektedir. Buna göre, Veri seti-2 için %98,353 EAA ile BA+DVM modeli diğer modellere kıyasla iyi bir performans göstermiştir. Bunun yanısıra, %97,763 ile BA+ÇKA modeli, ÇKA olan modeller içinde en yüksek EAA değerine sahiptir. Dahası, iki sınıflandırıcıyı karşılaştırdığımızda DVM algoritması ÇKA'ya kıyasla daha iyi performans sunmuştur. Sonuç olarak, Veri seti-2 için RA ve BA en iyileme yöntemleri ile inşa edilen DVM ve ÇKA sınıflandırıcıları temel modellere kıyasla daha iyi bir performans gösterrmiştir. Bu sonuçlar, en iyileme yöntemlerinin sınıflandırıcı algoritmalar üzerindeki pozitif etkisini ayrıca göstermektedir.

Tablo 4.10 Veri seti-2 için her bir model ile yapılan 10 deneye ait AİK çizimleri

DVM	ÇKA
	
RA+DVM	RA+ÇKA
	
BA+DVM	BA+ÇKA
	
OEİ+DVM	OEİ+ÇKA
	

Tablo 4.11 Veri seti-2 için her bir model ile yapılan 10 deneyin EAA değerlerinin ortalaması

Hiperparametre en iyileme	DVM (%)	ÇKA (%)
Hayır	97,219	97,706
RA	97,793	97,737
BA	98,353	97,763
OEİ	94,092	97,408

Tablo 4.12, en iyileme metotlarının en iyi doğruluk değeri sunduğu deney için parametre değerlerini ve deney numarasını sunar. Temel sınıflandırıcıların varsayılan parametre bilgileri de ayrıca son sütunda verilmiştir.

Tablo 4.12 Veri seti-2 için en iyi hiperparametre değerleri

Sınıflandırıcı	Hiperparametre	RA	BA	OEİ	Varsayılan değer
DVM	C kernel Gamma	50 linear auto	10 poly Scale	20,41 rbf Scale	1,0 rbf scale
	Deney no.	1	9	9	1
ÇKA	activation learning rate solver	relu adaptive Adam	identity invscaling lbfgs	relu constant lbfgs	relu constant Adam
	Deney no.	8	9	9	8

Tablo 4.13, tüm modellerinn deneysel çalışmalardaki ortalama yürütme sürelerini gösterir. Buna göre, varsayılan parametreleri ile bir sınıflandırıcının eğitimi için geçen zaman oldukça azdır. RA en iyileme metodu ile inşa edilen modellerin eğitimi için geçen zaman diğer en iyileme metotlarına kıyasla daha az olup bu yöntemi sırasıyla BA ve OEİ takip etmektedir.

Tablo 4.13 Veri seti-2 üzerinde her bir modelin eğitimi için geçen ortalama süre (sn)

Sınıflandırıcı	Temel Model	RA	BA	OEİ
DVM	0,30	6,90	7,21	14,53
ÇKA	4,30	101,70	269,44	353,81

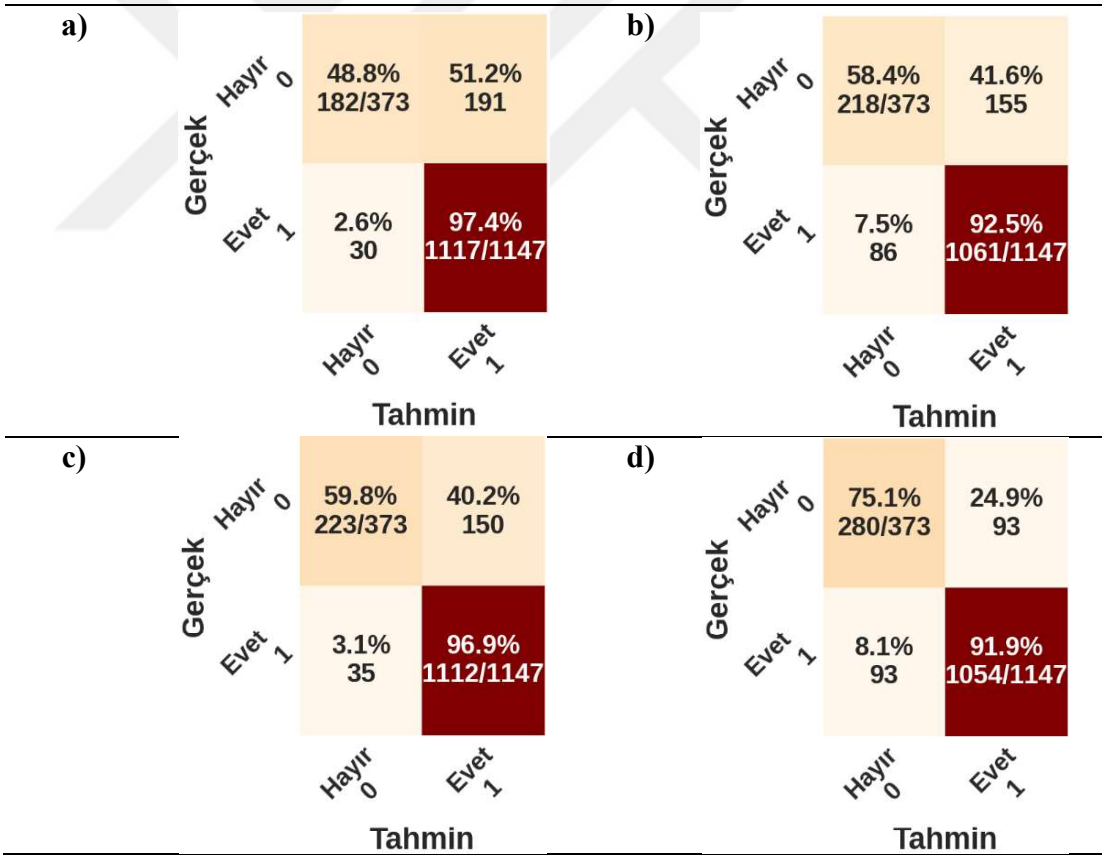
sn: saniye

4.3.3 Veri Seti-3 İçin Elde Edilen Sonuçlar

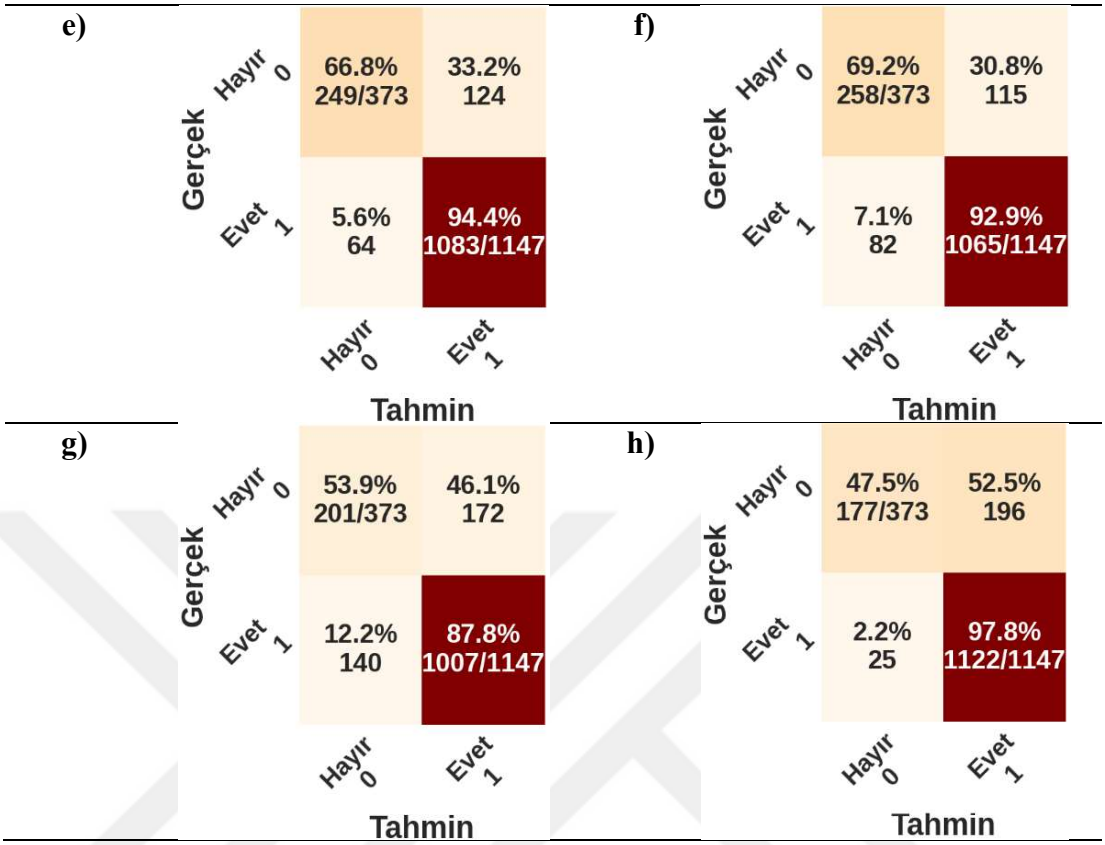
Tablo 4.14'te, Veri seti-3 için her bir sınıflandırıcı ve en iyileme metotları ile yapılan 10 deney ile elde edilen karışıklık matrislerinin üst üste bindirilmiş sonuç matrislerini

sunar. Buna göre, RA ve BA en iyileme metotlarının yer aldığı modeller temel modellere kıyasla daha az hatalı sınıflandırma yapmıştır. Temel model 221 tane yanlış sınıflandırma yaparken RA+DVM, BA+DVM ve OEİ+DVM sırasıyla 185, 188 ve 312 örneği yanlış sınıflandırmıştır. Tüm en iyileme metotlarının yer aldığı ÇKA modelleri temel ÇKA modeline kıyasla daha az hatalı sınıflandırma yapmıştır. ÇKA temel modeli 241 örneği yanlış sınıflandırırken, RA+ÇKA, BA+ÇKA ve OEİ+ÇKA sırasıyla 186, 197 ve 221 örneği yanlış sınıflandırmıştır. Özetle, bu veri seti üzerinde yapılan deneysel çalışmalarda DVM sınıflandırıcısı üzerinde RA ve BA en iyileme metotlarının olumlu etkisinin olduğu ve ÇKA sınıflandırıcısı üzerinde tüm en iyileme metotlarının olumlu etkisinin olduğu görülmüştür.

Tablo 4.14 Veri seti-3 için her bir modelin örtüşen karışıklık matrisleri; a) Temel DVM b) Temel ÇKA, c) RA+DVM d) RA+ÇKA e) BA+DVM f) BA+ÇKA g) OEİ+DVM h) OEİ+ÇKA



Tablo 4.14 Devamı



Tablo 4.15 Veri seti-3 üzerinde her bir model ile yapılan 10 deneyin ortalama sonuçları

Sınıflandırıcı	Metot	Doğruluk (%)	Özgüllük (%)	Duyarlılık (%)	Kesinlik (%)	F1-ölçütü (%)
DVM	Hayır	85,46	49,32	97,36	85,37	90,97
	RA	87,83	60,06	96,89	88,07	92,27
	BA	87,63	67,03	94,39	89,84	92,06
	OEİ	79,47	54,88	88,15	85,42	86,76
ÇKA	Hayır	84,14	60,38	92,62	87,65	90,07
	RA	87,76	75,35	91,92	91,88	91,9
	BA	87,04	69,04	92,83	90,33	91,56
	OEİ	85,46	46,54	97,76	85,21	91,05

Tablo 4.15, Veri seti-3 üzerinde oluşturulan tüm modellerin performanslarını sunar. Görüldüğü üzere, hiperparametre değeri en iyilemesi ile RA+DVM ve BA+DVM modelleri doğruluk değerini yükseltmiş olup OEİ+DVM modeli bir iyileşme sunmamıştır. Temel DVM'nin %85,46 doğruluk değeri RA ile %87,83'e ve BA ile %87,63'e yükselmiştir. Diğer yandan, tüm en iyileme yöntemleri ÇKA sınıflandırıcısında başarıyı yükseltmiştir. Buna göre, %84,14 doğruluğa sahip temel

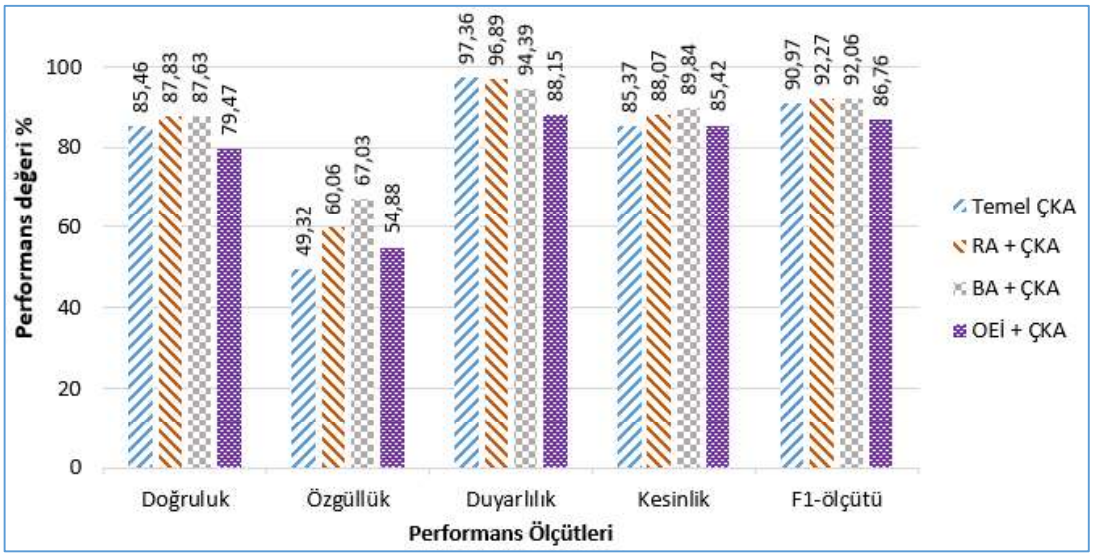
ÇKA, RA ile %87,76'ya, BA ile %87,04'e ve OEİ ile %85,46'ya yükselmiştir. Özet olarak, RA ve BA en iyileme yöntemleri ile oluşturulan DVM modellerinin doğruluk değerleri temel DVM'ye kıyasla ve tüm en iyileme yöntemleri ile oluşturulan ÇKA modellerinin doğruluk değerleri temel ÇKA'ya kıyasla daha yüksektir.

Özgüllük metriği için tüm en iyileme metotlarının DVM'nin performansında iyileşme sağladığı görülmüştür. ÇKA için OEİ hariç diğer iki en iyileme yönteminin iyileşme sağladığı görülmüştür. DVM sınıflandırıcısı için BA yöntemi, ÇKA sınıflandırıcısı için RA yöntemi diğer yöntemlere kıyasla daha başarılı olmuştur. Buna göre, BA+DVM ve RA+ÇKA modelleri sırasıyla %67,03 ve %75,35 özgüllük değerlerine sahiptir.

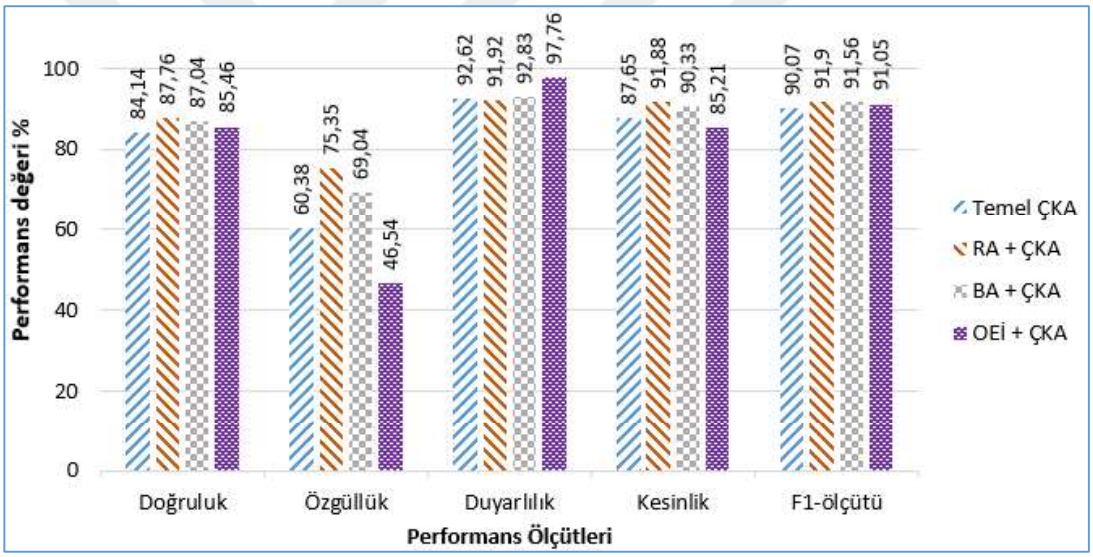
Duyarlılık metriği için DVM sınıflandırıcısı üzerinde en iyileme metotları ile iyileşmeler sağlanamadığı görülmüş olup ÇKA için ise BA %92,83 duyarlılık değeri sunarken OEİ %97,76 ile yüksek duyarlılık değeri sunmuştur.

Kesinlik metriği için DVM sınıflandırıcısı için özellikle RA ve BA'nın OEİ'ye kıyasla daha fazla iyileşme sağladığı, ÇKA sınıflandırıcısı için RA ve BA'nın iyileşme sağladığı, OEİ'nin bu sınıflandırıcı için iyileşme sağlamadığı görülmüştür. Buna göre, DVM ile yapılan deneylerde %89,84 ile BA+DVM modeli ve ÇKA ile yapılan deneylerde %91,88 ile RA+ÇKA yüksek kesinlik değerleri sunmuştur.

F1 ölçütü için DVM sınıflandırıcısında RA ve BA en iyileme yöntemlerinin, ÇKA sınıflandırıcısında ise tüm en iyileme yöntemlerinin iyileşme sunduğu görülmüştür. Dahası, her iki sınıflandırıcı için RA metodunun diğerlerine kıyasla daha yüksek F1-ölçütü değeri sunduğu görülmektedir. Şekil 4.7 ve Şekil 4.8 elde edilen sonuçları grafiksel olarak sunmaktadır.



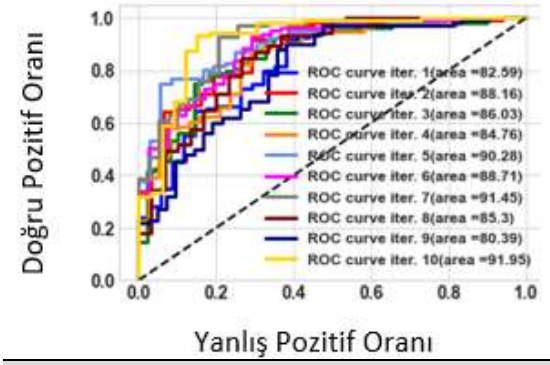
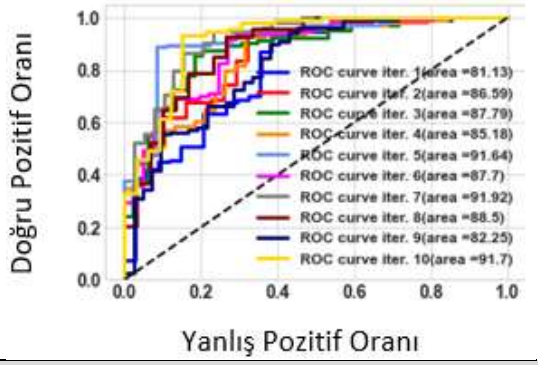
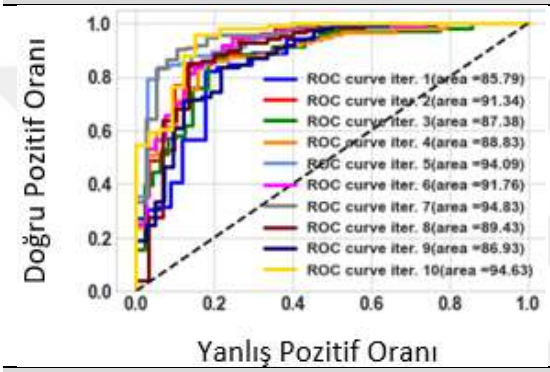
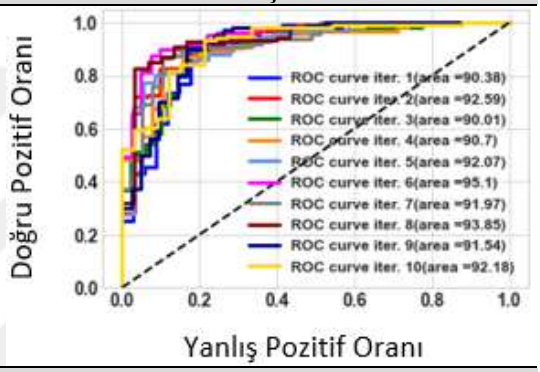
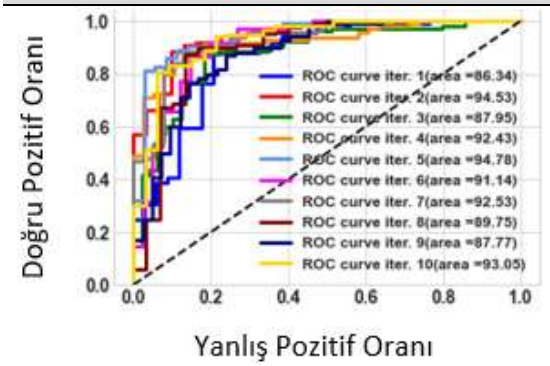
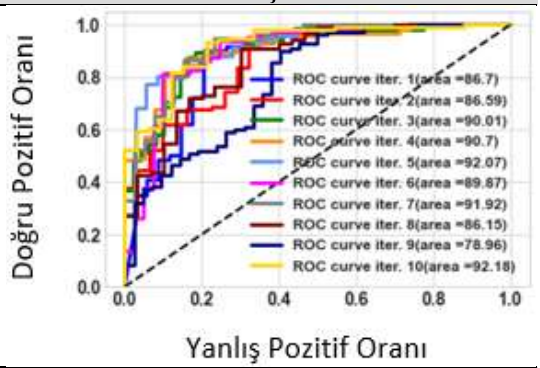
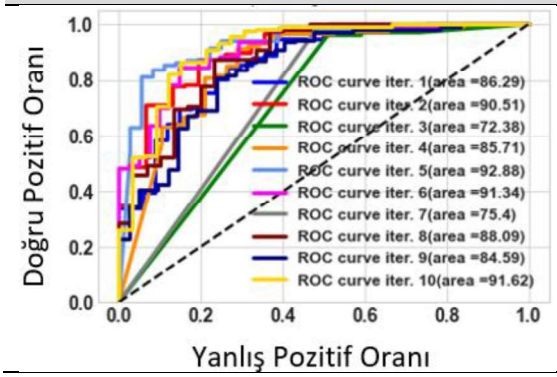
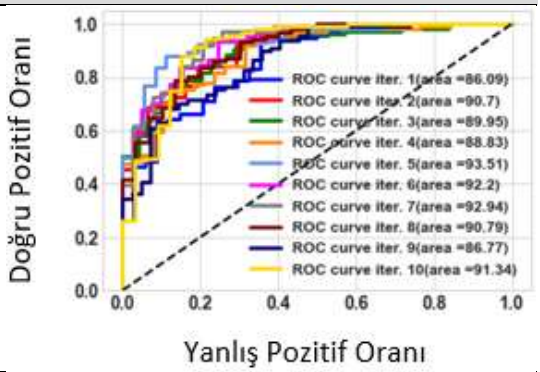
Şekil 4.7 Veri seti-3 için DVM sonuçları



Şekil 4.8 Veri seti-3 için ÇKA sonuçları

Tablo 4.16, Veri seti-3 için oluşturulan modellere ait AİK çizimlerini sunar. Her bir satırda iki şekil bulunmaktadır. Bu şekiller sırasıyla DVM ve ÇKA sınıflandırıcılarına aittir. Her AİK çiziminin üzerinde ilgili model bilgisi bulunmaktadır. Örneğin, 4. satır 1. sütundaki AİK çizimleri, OEİ+DVM modeline aittir. Her bir şekilde bir modele ait 10 deneyin her birinden elde edilen AİK çizimleri bir arada verilmiştir.

Tablo 4.16 Veri seti-3 için her bir model ile yapılan 10 deneye ait AİK çizimleri

DVM	ÇKA
	
RA+DVM	RA+ÇKA
	
BA+DVM	BA+ÇKA
	
OEİ+DVM	OEİ+ÇKA
	

Tablo 4.17, 10 deneyden elde edilen EAA değerlerinin ortalamasını sunar. Bu değerler, en iyileme metotlarının sınıflandırıcı algoritmalar üzerindeki sınıflandırma etkisini özetlemektedir. Buna göre, Veri seti-3 için %91,027 EAA ile BA+DVM modeli diğer DVM modellerine kıyasla iyi bir performans göstermiştir. Bunun yanısıra, %92,04 ile RA+ÇKA sınıflandırıcısı ÇKA olan modeller içinde en yüksek EAA değerine sahiptir. Dahası, iki sınıflandırıcıyı karşılaştırdığımızda RA+ÇKA modeli BA+DVM'ye kıyasla daha iyi performans sunmuştur. Sonuç olarak, Veri seti-3 için RA ve BA en iyileme yöntemleri ile inşa edilen DVM modelleri ve tüm en iyileme yöntemleri ile inşa edilen ÇKA modelleri temel DVM ve temel ÇKA modellerine kıyasla daha iyi bir performans gösterrmiştir. Bu sonuçlar, en iyileme metotlarının sınıflandırıcı algoritma üzerindeki pozitif etkisini ayrıca göstermektedir.

Tablo 4.17 Veri seti-3 için her bir model ile yapılan 10 deneyin EAA değerlerinin ortalaması

Hiperparametre en iyileme	DVM (%)	ÇKA (%)
Hayır	86,962	87,44
RA	90,501	92,04
BA	91,027	88,515
OEİ	85,881	90,312

Tablo 4.18 Veri seti-3 için en iyi hiperparametre değerleri

Sınıflandırıcı	Hiperparametre	RA	BA	OEİ	Varsayılan değer
DVM	C kernel Gamma	50 rbf auto	10 poly scale	14,89 rbf scale	1,0 rbf scale
	Deney no.	10	2	2	10
ÇKA	activation learning_rate solver	identity adaptive Adam	identity invscaling lbfgs	identity adaptive Adam	relu constant Adam
	Deney no.	6	7	10	10

Tablo 4.18, en iyileme metotlarının en iyi doğruluk değeri sunduğu deney için parametre değerlerini ve deney numarasını sunar. Temel sınıflandırıcıların varsayılan parametre bilgileri de ayrıca son sütunda verilmiştir.

Tablo 4.19, tüm modellerinn deneysel çalışmalardaki ortalama yürütme sürelerini gösterir. Varsayılan parametreleri ile bir sınıflandırıcının eğitimi için geçen zaman

oldukça azdır. BA en iyileme metodu ile inşa edilen modellerin eğitimi için geçen zaman diğer en iyileme metotlarına kıyasla daha azdır. Bunu sırasıyla RA ve OEİ takip etmektedir.

Tablo 4.19 Veri seti-3 üzerinde her bir modelin eğitimi için geçen ortalama süre (sn)

Sınıflandırıcı	Temel Model	RA	BA	OEİ
DVM	17,43	133,14	159,21	500,88
ÇKA	1,37	10,52	16,38	32,76

sn: saniye

5. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasının amacı, ikili sınıflandırma problemlerinde sınıflandırıcı modelin doğruluğunu artırmaktır. Doğruluk, sınıflandırıcı algoritmaların optimize edilmesiyle arttırılabilir. Bu çerçevede, DVM ve ÇKA sınıflandırıcı algoritmaları için en iyi hiperparametre değerlerinin tespiti için RA, BA ve OEİ en iyileme yöntemleri ile deneysel çalışmalar gerçekleştirilmiştir.

Deneysel çalışmalarda; Veri seti-1 için %92,98 ile %93,25 arasında ortalama genel sınıflandırma doğrulukları ve %97,023 ile %98,124 arasında ortalama EAA değerleri elde edilmiştir. Buna göre, veri seti-1 için DVM sınıflandırıcısının genel sınıflandırma doğruluğunda en iyi %0,27 artış sağlanmış olup ÇKA'nın genel sınıflandırma doğruluğunda bir iyileşme sağlanamamıştır. Bunun yanı sıra, DVM ve ÇKA sınıflandırıcılarının EAA değerlerinde sırasıyla en iyi %1,101 ve %0,005 artış sağlanmıştır. Veri seti-2 için %89,02 ile %93,42 arasında ortalama genel sınıflandırma doğrulukları ve %94,092 ile %98,353 arasında ortalama EAA değerleri elde edilmiştir. Buna göre, veri seti-2 için DVM sınıflandırıcısının genel sınıflandırma doğruluğunda en iyi %2,12 artış sağlanmış olup ÇKA'nın doğruluğunda bir iyileşme sağlanamamıştır. Bunun yanı sıra, DVM ve ÇKA sınıflandırıcılarının EAA değerlerinde sırasıyla en iyi %1,134 ve %0,057 artış sağlanmıştır. Son olarak, veri seti-3 için %79,47 ile %87,83 arasında ortalama genel sınıflandırma doğrulukları ve %85,881 ile %92,04 arasında ortalama EAA değerleri elde edilmiştir. Buna göre, veri seti-3 için DVM ve ÇKA sınıflandırıcılarının genel sınıflandırma doğruluklarında sırasıyla en iyi %2,37 ve %3,62 artış sağlanmıştır. Bunun yanı sıra, EAA değerinde DVM ve ÇKA için sırasıyla en iyi %4,065 ve %4,6 artış sağlanmıştır.

Genel olarak değerlendirildiğinde, RA ve BA, hem DVM hem de ÇKA için OEİ metodundan çoğunlukla daha iyi performans göstermiştir. Bunun yanı sıra, bazı modellerde en iyileme yöntemlerinin ilgili metrik bağlamında temel modele kıyasla daha iyi performans sunamadığı da görülmüştür. Genel sınıflandırma doğrulukları açısından veri seti-1 için DVM sınıflandırıcısında RA, BA ve OEİ iyileşme sağlarken, ÇKA sınıflandırıcısında en iyileme yöntemleri iyileşme sağlamamıştır. Veri seti-2 için DVM sınıflandırıcısında RA ve BA katkı sunarken ÇKA sınıflandırıcısında en iyileme

yöntemleri iyileşme sağlamamıştır. Son olarak, Veri seti-3 için DVM sınıflandırıcısında RA ve BA en iyileme metotları iyileşme sağlarken, ÇKA sınıflandırıcısında tüm en iyileme metotltarı iyileşme sağlamıştır. Bu deneysel çalışmalar çerçevesinde; ele alınan probleme bağlı olarak DVM ve ÇKA sınıflandırıcıları üzerinde en iyileme yöntemlerinin pozitif etkisinin olabileceği sonucuna varabiliriz. Ayrıca, deneysel çalışmalar en iyileme metotları sınıflandırıcı algoritmalar üzerinde problemden bağımsız olarak her zaman iyileşme sağlar gibi bir sonuca varılamayacağını bize göstermiştir.

En iyileme yöntemleri ile modelin parametre değerlerinin en iyilenmesi için model eğitiminde 5-kat çapraz doğrulama tekniğinin uygulanması ile geçen zaman temel DVM ve temel ÇKA modellerine kıyasla doğal olarak daha fazla olmuştur.

Gelecekte, farklı sınıflandırıcılar üzerinde en iyileme yöntemlerinin etkisinin araştırılması hedeflenmektedir. Ayrıca, Yapay Sinir Ağları ve Derin Sinir Ağları için en uygun gizli katman ve nöron sayılarının da tespiti üzerine çalışmalar yapılması planlanmaktadır. Dahası, öznitelik seçimi ve hiperparametre en iyilenmesi hibrit çalışmalarının da yapılması hedefler arasındadır.

KAYNAKLAR

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., Arshad, H. (2018). *State of the art in artificial neural network applications: A survey.* Heliyon, 4(11), e00938. https://doi.org/10.1016/j.heliyon.2018.e00938.

Agarap, F. A. (2019). Deep Learning using Rectified Linear Units (ReLU), *arXiv:1803.08375v2 [cs.NE]* 7 Feb 2019. https://arxiv.org/abs/1803.08375

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyper-parameter optimization framework. In Proceedings of the 25th *ACM SIGKDD international conference on knowledge discovery & data mining (KDD '19). Association for Computing Machinery, New York, NY, USA,* (pp. 2623-2631). https://doi.org/10.1145/3292500.3330701

Akpınar, H. (2014). Data: Veri Madenciliği Veri Analizi, 1. baskı. Papatya Yayıncılık Eğitim, İstanbul.

Aljarah, I., Al-Zoubi, A. M., Faris, H., Hassonah, M. A., Mirjalili, S., & Saadeh, H. (2018). Simultaneous Feature Selection and Support Vector Machine Optimization Using the Grasshopper Optimization Algorithm. *Cognitive Computation*, 10(3), 478–495. https://doi.org/10.1007/s12559-017-9542-9

Al-Mejibli, I. S., Alwan, J. K., & Abd Dhafar, H. (2020). The effect of gamma value on support vector machine performance with different kernels. *International Journal of Electrical and Computer Engineering,* 10(5), 5497-5506. DOI: 10.11591/ijece.v10i5.

Barros, B., Lacerda, P., Albuquerque, C., & Conci, A. (2021). Pulmonary COVID-19: Learning Spatiotemporal Features Combining CNN and LSTM Networks for Lung Ultrasound Video Classification. *Sensors*, *21*(16), 5486. 1-25. https://doi.org/10.3390/s21165486

Baştanlar Y., Özuysal M. (2014) Introduction to Machine Learning. In: Yousef M., Allmer J. (eds) *miRNomics: MicroRNA Biology and Computational Analysis*. Methods in Molecular Biology (Methods and Protocols), vol 1107. (pp 105-128). Humana Press, Totowa, NJ. https://doi.org/10.1007/978-1-62703-748-8_7

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 281-305. https://dl.acm.org/doi/10.5555/2188385.2188395

Bhatnagar, R. (2018). Machine Learning and Big Data Processing: A Technological Perspective and Review. In: Hassanien, A., Tolba, M., Elhoseny, M., Mostafa, M. (eds) *The International Conference on Advanced Machine Learning Technologies and Applications* (AMLTA2018). AMLTA 2018. Advances in Intelligent Systems and Computing, vol 723. (pp. 468-478). Springer, Cham. https://doi.org/10.1007/978-3-319-74690-6_46

Bottou, L., & Bousquet, O. (2007). The tradeoffs of large scale learning. *Advances in neural information processing systems*, vol 20. (pp. 161-168). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2007/file/0d3180d672e08b4c5312dcdafdf6ef36-Paper.pdf

Byrd, R. H., Peihuang, L., Nocedal, J., Zhu, C. (1995). A Limited Memory Algorithm for Bound Constrained Optimization, *SIAM Journal on Scientific Computing*, 16(5): 1190–1208. https://doi.org/10.1137/0916069

Chen, H., Zhang Z., Yin W., Zhao, C., Wang, F., Li, Y. (2022). A study on depth classification of defects by machine learning based on hyper-parameter search, *Measurement*, 189, 1-11, https://doi.org/10.1016/j.measurement.2021.110660.

Chen, J. C., & Wang, Y. M. (2020). Comparing activation functions in modeling shoreline variation using multilayer perceptron neural network. *Water*, 12(5), 1281. 1-12. https://doi.org/10.3390/w12051281

Chon, S. (2019). *Hyper-Parameter Optimization of a Convolutional Neural Network* (No. Ad1077380). Graduate School of Engineering and Management Air Force Institute of Technology Air University. https://apps.dtic.mil/sti/citations/AD1077380

Cinar, I. and Koklu, M. (2019). Classification of Rice Varieties Using Artificial Intelligence Methods. *International Journal of Intelligent Systems and Applications in Engineering*, 7(3), 188-194. https://doi.org/10.18201/ijisae.2019355381

Dongare, A. D., Kharde, R. R., Kachare, A. D. (2012). Introduction to Artificial Neural Network. *International Journal of Engineering and Innovative Technology (IJEIT), 2*(1), 189–194.

Ekundayo, I. (2020). *OPTUNA Optimization Based CNN-LSTM Model for Predicting Electric Power Consumption.* Masters thesis, Dublin, National College of Ireland. http://norma.ncirl.ie/id/eprint/4440

Ertorsun, A. D., Bağ, B., Uzar, G. ve Turanoğlu, M. A. (2009). *ROC (Receiver Operating Characteristic) Eğrisi Yöntemi ile Tanı Testlerinin Performanslarının Değerlendirilmesi*, XIII. Öğrenci Sempozyumu, Ankara.

Ferraris V. A. (2018). Commentary: Should we rely on receiver operating characteristic curves? From submarines to medical tests, the answer is a definite maybe!. *The Journal of thoracic and cardiovascular surgery*, 157(6), 2354–2355. https://doi.org/10.1016/j.jtcvs.2018.10.100

Fletcher, R. (1981). *Practical Methods of Optimization. Volume 2: Constrained Optimization* (1st ed., Vol. 2). John Wiley & Sons.

Gan, C. L. (2020). Prognostics and Health Management of Electronics: Fundamentals, MachineLearning, and the Internet of Things. *Life Cycle Reliability and Safety Engineering*, 9(2), 225–226. https://doi.org/10.1007/s41872-020-00119-y

Hamida, S., El Gannour, O., Cherradi, B., Ouajji, H., & Raihani, A. (2020, December). Optimization of machine learning algorithms hyper-parameters for improving the prediction of patients infected with COVID-19. In *2020 İEEE 2nd international conference on electronics, control, optimization and computer science* (*İCECOCS*) (pp. 1-6). IEEE Xplore. https://doi.org/10.1109/ICECOCS50124.2020.9314373

Han J, Kamber M, Pei J. (2012). Data Preprocessing. *Data Min,* 83-124. https://doi.org/10.1016/B978-0-12-381479-1.00003-4.

Hurwitz, J., & Kirsch, D., (2018). *Machine learning for dummies*. IBM Limited Edition, John Wiley & Sons, Inc., pp. 1-68. https://www.ibm.com/downloads/cas/GB8ZMQZ3

Javeed, A., Zhou, S., Yongjian, L., Qasim, I., Noor, A., & Nour, R. (2019). An intelligent learning system based on random search algorithm and optimized random forest model for improved heart disease detection. *IEEE Access*, vol 7, pp. 180235-180243. doi: 10.1109/ACCESS.2019.2952107.

Kahraman, G. (2021). *Bilgisayarli Tomografi Görüntülerinden Derin Öğrenme Ve Makine Öğrenmesi İle Covid-19 Hastaliğinin Teşhisi*, Yüksek Lisans Tezi, Çankırı.

Kaytan, M., Yeroğlu, C., Aydilek, İ. B., (2020). Yapay Sinir Ağları Eğitiminde Kullanılan Optimizasyon Yöntemlerinin İncelenmesi ve Kan Nakli Hizmet Merkezi Veri Seti Üzerinden Değerlendirilmesi, *Anatolian Journal of Computer Sciences*, 5(2), 99-113. https://dergipark.org.tr/en/pub/bbd/issue/57870/735975

Khan, M. M. R., Arif, R. B., Siddique, M. A. B., & Oishe, M. R. (2018). Study and observation of the variation of accuracies of KNN, SVM, LMNN, ENN algorithms on eleven different datasets from UCI machine learning repository. In *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)* (pp. 124-129). IEEE. https://doi.org/10.1109/CEEICT.2018.8628041

Kingma, D.P., Ba, J. (2015). Adam: A Method for Stochastic Optimization, *arXiv preprint arXiv:1412.6980* (Published as a conference paper at *the 3rd International Conference for Learning Representations, San Diego*, 2015), 1-15. https://doi.org/10.48550/arXiv.1412.6980

Kok, J. N., Boers, E. J., Kosters, W. A., Van der Putten, P., & Poel, M. (2009). Artificial intelligence: definition, trends, techniques, and cases. *Artificial intelligence- Encyclopedia of Life Support Systems (EOLSS)*, 1, 270-299.

Lacerda, P., Barros, B., Albuquerque, C., & Conci, A. (2021). Hyperparameter Optimization for COVID-19 Pneumonia Diagnosis Based on Chest CT. *Sensors*, 21(6), 1-11. https://doi.org/10.3390/s21062174

Li, B. (2020). *Random Search Plus: A more effective random search for machine learning hyperparameters optimization*. Master's Thesis, University of Tennessee. https://trace.tennessee.edu/utk_gradthes/5849

Li, C., Liu, K., & Wang, H. (2011). The incremental learning algorithm with support vector machine based on hyperplane-distance. *Applied Intelligence*, 34(1), 19–27. ID - Li2011. https://doi.org/10.1007/s10489-009-0176-9

Lin, S. W., Ying, K. C., Chen, S. C., & Lee, Z. J. (2008). Particle swarm optimization for parameter determination and feature selection of support vector machines. *Expert Systems with Applications*, 35(4), 1817-1824. https://doi.org/10.1016/j.eswa.2007.08.088

Mahesh, B. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR)*. 9(1), 381-386. https://doi.org/10.21275/ART20203995

Mantovani, R. G., Rossi, A. L., Vanschoren, J., Bischl, B., & De Carvalho, A. C. (2015 , July). Effectiveness of random search in DVM hyper-parameter tuning. In *2015 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). Ieee. doi: 10.1109/IJCNN.2015.7280664

Metlek, S. ve Kayaalp, K., (2020). *Makine Öğrenmesinde, Teoriden Örnek Matlab Uygulamalarına Kadar Destek Vektör Makineleri*, Ankara: İksad Yayınevi.

Mitchell T (1997) Machine learning. McGraw Hill, New York

Mubarak, A. S., Serte, S., Al-Turjman, F., Ameen, Z. S. I., & Ozsoz, M. (2021). Local binary pattern and deep learning feature extraction fusion for COVID-19 detection on computed tomography images. *Expert Systems*, 39(3), e12842. pp 1-13. https://doi.org/10.1111/exsy.12842

Optuna Contributors, (2018a). Optuna: A hyper-parameter optimization framework - Optuna 2.9.1 documentation. Optuna. *https://optuna.readthedocs.io/en/stable/index.html*

Optuna Contributors. (2018b). Efficient Optimization Algorithms — Optuna 2.10.0 documentation. Optuna. https://optuna.readthedocs.io/en/stable/tutorial/10_key_features/003_efficient_optimization_algorithms.html?highlight=SamplingAlgorithms#sampling-algorithms

Ozkan I.A., Koklu M. and Saracoglu R. (2021). Classification of Pistachio Species Using Improved K-NN Classifier. Progress in Nutrition, 23(2), 1-9. doi:10.23751/pn.v23i2.9686.

Öztemel, E. 2003. *Yapay Sinir Ağlari.* Papatya Yayıncılık.

Rahmadayana, F., & Sibaroni, Y. (2021). Sentiment Analysis of Work from Home Activity using SVM with Randomized Search Optimization. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 5(5), 936-942. https://doi.org/10.29207/resti.v5i5.3457

Ramadhan, M. M., Sitanggang, I. S., Nasution, F. R., & Ghifari, A. (2017). Parameter Tuning in Random Forest Based on Grid Search Method for Gender Classification Based on Voice Frequency. *DEStech Transactions on Computer Science and Engineering, (CECE 2017).* pp. 625-629. https://doi.org/10.12783/dtcse/cece2017/14611

Ray, S. (2019, February). A quick review of machine learning algorithms. In *2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon)* (pp. 35-39). IEEE. http://doi.org/10.1109/COMITCon.2019.8862451

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science. https://apps.dtic.mil/sti/citations/ADA164453

Sakar, C.O., Serbes, G., Gunduz, A., Tunc, H.C., Nizam, H., Sakar, B.E., Tutuncu, M., Aydin, T., Isenkul, M.E. and Apaydin, H. (2019). A comparative analysis of speech signal processing algorithms for Parkinson's disease classification and the use of the tunable Q-factor wavelet transform. *Applied Soft Computing*, 74, 255-263. https://doi.org/10.1016/j.asoc.2018.10.022.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., & de Freitas, N. (2016). Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE*, 104(1), 148–175. https://doi.org/10.1109/jproc.2015.2494218.

Somvanshi, M., Chavan, P., Tambade, S., & Shinde, S. V. (2016, August). A review of machine learning techniques using decision tree and support vector machine. In *2016 international conference on computing communication control and automation (ICCUBEA)* (pp. 1-7). IEEE. https://doi.org/10.1109/iccubea.2016.7860040

Sridevi, N., Varsha, K., & Navin, M., Jr. (2019). Machine Learning Algorithms: Diagnosing Breast Cancer. *International Journal of Recent Technology and Engineering*, *8*(2S6), 849–851. https://doi.org/10.35940/ijrte.b1157.0782s619

Srinivas, P., & Katarya, R. (2022). hyOPTXg: OPTUNA hyper-parameter optimization framework for predicting cardiovascular disease using XGBoost. *Biomedical Signal Processing and Control,* vol 73, 103456. https://doi.org/10.1016/j.bspc.2021.103456

Syarif, I., Prugel-Bennett, A., & Wills, G. (2016). DVM Parameter Optimization using Grid Search and Genetic Algorithm to Improve Classification Performance. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 14(4), 1502-1509. *https://doi.org/10.12928/telkomnika.v14i4.3956*.

URL-1. https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html (Erişim Tarihi: 02.06.2021).

URL-2. https://datascience.foundation/datatalk/basic-overview-of-svm-algorithm, (Erişim Tarihi:05.07.2021)

URL-3. https://www.emo.org.tr/ekler/490c742cd8318b8_ek.pdf (Erişim Tarihi: 03.06.2021).

URL-4. https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.mlpclassifier.html (Erişim Tarihi: 03.06.2021).

URL-5. scikit-optimize contributors. (2017–2020). skopt.BayesSearchCV — scikit-optimize 0.8.1 documentation. https://scikit-optimize.github.io/stable/modules/generated/skopt.BayesSearchCV.html?highlight=bayessearchcv, (Erişim Tarihi: 04.06.2021).

URL-6. Optuna Hiper Optimizasyon Modülü. https://optuna.org/ (Erişim Tarihi: 04.06.2021)

Valarmathi, R., & Sheela, T. (2021). Heart disease prediction using hyper parameter optimization (HPO) tuning. *Biomedical Signal Processing and Control*, 70(103033). 1-10. https://doi.org/10.1016/j.bspc.2021.103033

Victoria, A. H., & Maragatham, G. (2021). Automatic tuning of hyperparameters using Bayesian optimization. Evolving Systems, *Evolving Systems,* 12(1), 217-223. https://doi.org/10.1007/s12530-020-09345-2

Wang, P. P., Mao, Z. L., Chen, J. D., & Pan, F. (2011). Parameter optimization algorithm for support vector machine based on predatory search genetic algorithm. *Jisuanji Yingyong/ Journal of Computer Applications*, 31(2), 498–500. https://doi.org/10.3724/sp.j.1087.2011.00498

Wu, J., Chen, X. Y., Zhang, H., Xiong, L. D., Lei, H., & Deng, S. H. (2019). Hyper-parameter optimization for machine learning models based on Bayesianian optimization. *Journal of Electronic Science and Technology*, 17(1), 26-40. https://doi.org/10.11989/JEST.1674-862X.80904120

Xia, Y., Liu, C., Li, Y., & Liu, N. (2017). A boosted decision tree approach using Bayesianian hyper-parameter optimization for credit scoring. *Expert Systems with Applications,* 78, 225–241. https://doi.org/10.1016/j.eswa.2017.02.017

Yin, J., & Li, N. (2022). Ensemble learning models with a Bayesianian optimization algorithm for mineral prospectivity mapping. *Ore Geology Reviews*, 145, 104916. https://doi.org/10.1016/j.oregeorev.2022.104916

Zhang, W., Wu, C., Zhong, H., Li, Y., & Wang, L. (2021). Prediction of undrained shear strength using extreme gradient boosting and random forest based on Bayesian optimization. *Geoscience Frontiers*, *12*(1), 469–477. https://doi.org/10.1016/j.gsf.2020.03.007

Zhu, J., & Song, J. (2020). An Intelligent Classification Model for Surface Defects on Cement Concrete Bridges. *Applied Sciences*. 10(3), 1-15. https://doi.org/10.3390/app10030972