T.C.
MANİSA CELAL BAYAR ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ
YAZILIM MÜHENDİSLİĞİ ANABİLİM DALI
YAZILIM MÜHENDİSLİĞİ BİLİM DALI

MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE ORTAÖĞRETİM ÖĞRENCİLERİNİN BAŞARI DEĞERLENDİRİLMESİ

Murat SOLMAZ

Danışman
Dr. Öğr. Üyesi Fatih YÜCALAR

MANİSA–2024

Murat SOLMAZ	**MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE ORTAÖĞRETİM ÖĞRENCİLERİNİN BAŞARI DEĞERLENDİRİLMESİ**	**2024**

TAAHÜTNAME

Bu tezin Manisa Celal Bayar Üniversitesi Lisansüstü Eğitim Enstitüsü Yazılım Mühendisliği Ana Bilim Dalında, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

Murat SOLMAZ

TEZ ONAYI

Murat SOLMAZ tarafından hazırlanan **"MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE ORTAÖĞRETİM ÖĞRENCİLERİNİN BAŞARI DEĞERLENDİRİLMESİ"** adlı tez çalışması … / … /2024 tarihinde aşağıdaki jüri üyeleri önünde Manisa Celal Bayar Üniversitesi Fen Bilimleri Enstitüsü **Yazılım Mühendisliği Anabilim Dalı**'nda **YÜKSEK LİSANS** olarak savunulmuş ve **oyçokluğu / oybirliği** ile başarılı olarak kabul edilmiştir.

Danışman	Dr. Öğr. Üyesi Fatih YÜCALAR Manisa Celal Bayar Üniversitesi
Jüri Üyesi	Prof. Dr. Akın ÖZÇİFT Manisa Celal Bayar Üniversitesi
Jüri Üyesi	Prof. Dr. Aytuğ ONAN İzmir Kâtip Çelebi Üniversitesi

İÇİNDEKİLER

SİMGELER VE KISALTMALAR DİZİNİ

CNN	Convolutional Neural Networks (Evrişimli Sinir Ağları)
EM	Expectation Maximization (Beklentilerin Maksimum Edilmesi)
IDE	Integrated development environments (Entegre Geliştirme Ortamı)
LMT	Logistic Regression Tree (Lojistik Regresyon Ağacı)
LR	Logistic Regression (Lojistik Regresyon)
MCC	Matthews Correlation Coefficient (Matthews Korelasyon Katsayısı)
ML	Machine Learning (Makine Öğrenimi)
NB	Naive Bayes
ROC	Receiver Operating Characteristic (İşlem Karakteristik Eğrisi)
SMO	Sequential Minimal Optimization (Sıralı minimum optimizasyon)
SOM	Self Organization Maps (Kendini Organize Eden Haritalar)
SVM	Support Vector Machine (Destek Vektör Makinesi)
RF	Random Forest (Rastsal Orman)
PCA	Principal Component Analysis (Temel Bileşen Analizi)

ŞEKİLLER DİZİNİ

TABLO DİZİNİ

TEŞEKKÜR

Gerçekleştirmiş olduğum tez çalışmamın her aşamasında bana çok yardımcı olan, yol gösteren, her zaman çalışmaya teşvik eden, kendisi ile çalışmaktan gurur duyduğum danışmanım Sayın Dr. Öğr. Üyesi Fatih YÜCALAR'a, her sıkıntıya düştüğümde yanımda olan ve yüksek lisans tezinde ikinci danışmanım (Lisans tez danışmanım) Sayın Prof. Dr. Aytuğ ONAN'a, geliştirme aşamalarında bilgi ve yardımlarını eksik etmeyen hocam Sayın Prof. Dr. Akın ÖZÇİFT'e, fakültede ilk tanıştığım hocam herkesin yardımcısı, yüce gönüllü insan Sayın Dr. Öğr. Üyesi Emin BORANDAĞ hocama, kendisi ile geç tanıştığım ama güler yüzü pozitif enerjisi ile bizlere desteklerini sunan Sayın Doç. Dr. Nilüfer ATMAN USLU'ya, liseden eski öğrencim, fakülteden asistan hocam benden yardımlarını esirgemeyen sayın Arş. Gör. Tuğba ÇELİKTEN'e, ayrıca gençlik yıllarımdan bu yana bilimi ve bilim sevgisini, adanmışlığı her hareketi ile yaşayarak anlatan Sayın Prof. Dr. Halil ARDAHAN hocama ve her zaman yanımda olan, beni yüksek lisans eğitimime başlamaya teşvik eden kardeşim Dr. Öğr. Üyesi Melike SOLMAZ CİLOŞOĞLU'na ve tabii ki bu süreçte hep yanımda olan aileme başta sevgili eşim Şengül SOLMAZ olmak üzere Anne ve Babama sonsuz teşekkür ederim.

Bu tez çalışmamı başta kendi prenseslerim-kızlarım Mısra, Yağmur, Betül ve tüm öğrencilerime daha başarılı olmaları dileklerimle, ithaf ediyorum.

Murat SOLMAZ
Manisa, 2024

ÖZET

Yüksek Lisans Tezi

Murat SOLMAZ

Manisa Celal Bayar Üniversitesi
Lisansüstü Eğitim Enstitüsü
Yazılım Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Fatih YÜCALAR

Yüzyıllardır insanoğlu gelişmek, ilerlemek ve kendi zamanına kadarki bilim bayrağını hakkıyla temsil etmek için yeni nesillerini iyi yetiştirmek istemektedir. Günümüzde de insanın yaşadığı çağa ayak uydurabilmesi için eğitim alması zorunlu hale gelmiştir. Hatta iyi bir eğitim alabilmesi ve bunu hayatının her alanına yansıtabilmesi modern insanın olmazsa olmazıdır. Eğitim bu yetiştirme çabalarının genel çatı anlamıdır. İyi eğitim denince eğitim-öğretim faaliyetleri sırasında başarılı olmak akla gelir. Maalesef özellikle ülkemizde eğitim ve öğretimi değerlendirme için hala birtakım arayışların sürdüğü görülmektedir. Bu durum birçok alanda olduğu gibi eğitim-öğretim çalışmalarına bilgisayar bilimlerinin katkı vermesi gereğini ortaya koymaktadır. Bu aşamada ortaöğretim öğrencilerinin okul başarısını ve bu başarıya etkisi olabilecek tüm faktörler maddeleştirip, anket çalışmasıyla toplanarak veri seti hale getirilmiştir. 1000 ortaöğretim öğrencisi ile 45 maddelik anket soruları ile toplanan verilerimiz ön işleme aşamalarından geçirilerek düzenli hale getirilmiştir. WEKA platformu aracılığıyla %80'i eğitim, %20'si test verisi olarak ayrılan veri seti üzerinden makine öğrenmesi algoritmaları ile sınıflandırma ile tahminleme işlemleri yapılmıştır. Bu işlemlerde Naive Bayes, Logistic, Simple Logistic, SMO, Kstar, Jrip, PART, Decision Stump, J48, LMT, Random Forest algoritmaları ile meta algoritmalar Bagging, MetaAdaBoostM1, Random SupSpace ve Vote kullanılarak sınıflandırma çalışmaları yapılmıştır. Ele alınan tüm algoritmalar içinde %73,5 doğruluk değeri ile Random Forest ve %74 doğruluk değeri ile Random SubSpace ve Random Forest topluluk öğrenmesi en başarılı algoritmalar olarak tespit edilmiştir.

Anahtar Kelimeler: Makine Öğrenmesi, Sınıflandırma, Topluluk Öğrenmesi, Özellik Seçimi

2024, 69 sayfa

ABSTRACT

M.Sc. Thesis

Murat SOLMAZ

Manisa Celal Bayar University
Graduate School of Applied and Natural Sciences
Department of Software Engineering

Supervisor: Dr. Öğr. Üyesi Fatih YÜCALAR

For centuries, human beings have wanted to develop, progress, and raise their new generations well in order to properly represent the banner of science up to their time. Nowadays, it has become mandatory for people to receive education in order to keep up with the age they live in. In fact, it is a must for a modern person to be able to receive a good education and reflect this in every aspect of his life. Education is the general umbrella meaning of these training efforts. When it comes to good education, being successful during educational activities comes to mind. Unfortunately, it seems that there are still some searches for evaluation of education and training, especially in our country. This situation reveals the need for computer science to contribute to education and training studies, as in many fields. At this stage, the school success of secondary school students and all factors that may affect this success were itemized and collected through a survey and turned into a data set. Our data, collected with 45-item survey questions from 1000 secondary school students, was regularized by going through pre-processing stages. Classification and prediction processes were carried out with machine learning algorithms on the data set, which was divided into 80% training data and 20% test data through the WEKA platform. In these processes, classification studies were carried out using Naive Bayes, Logistic, Simple Logistic, SMO, Kstar, Jrip, PART, Decision Stump, J48, LMT, Random Forest algorithms and meta-algorithms Bagging, MetaAdaBoostM1, Random SupSpace and Vote. Among all the algorithms considered, Random Forest with an accuracy value of 73.5% and Random SubSpace and Random Forest ensemble learning with an accuracy value of 74% were determined to be the most successful algorithms.

Keywords: Machine Learning, Classification, Ensemble Learning, Feature Selection

2024, 69 pages

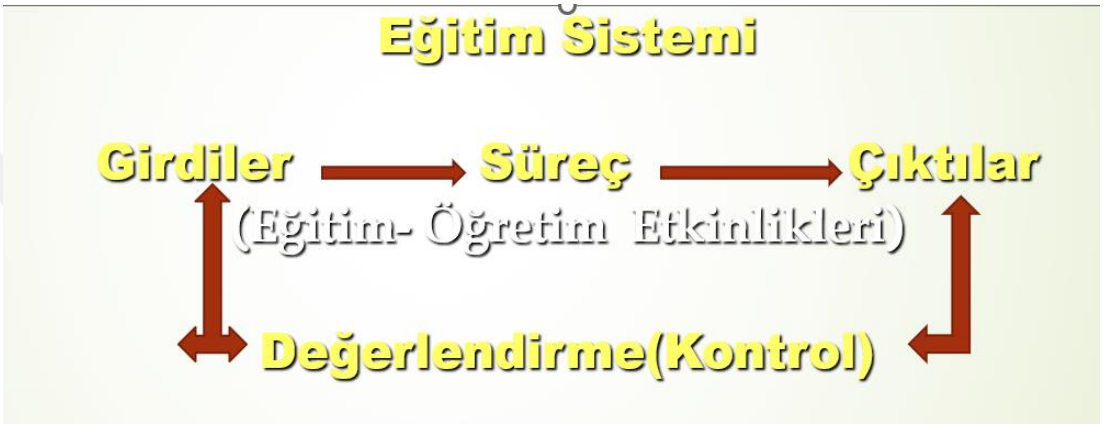
1. GİRİŞ

Eğitim, insan uygarlığı ile yaşıt sayılabilecek bir disiplindir. Eğitimin, yazının keşfiyle birlikte ortaya çıktığı ve buna paralel olarak eğitim sisteminin de geliştiği arkeolojik ve yazılı veriler ile kanıtlanmıştır [1]. Tarih boyunca insanoğlu gelişmek, ilerlemek ve kendi zamanına kadarki bilim bayrağını iyi bir şekilde temsil etmek amacıyla eğitime ihtiyaç duymuştur. Günümüzde de insanların iyi eğitim alabilmeleri ve aldıkları eğitimi yaşam alanlarına yansıtabilmeleri oldukça önemli bir konudur. Eğitim konusu içerisinde eğitim öğretim faaliyetleri kadar, bu faaliyetlerle birlikte elde edilen başarı sonuçlarının doğru ifade edilmesini de barındırmaktadır. Günümüzde, eğitim ve öğretimi değerlendirme hakkında anlaşılmayan metotların varlığı göze çarpmaktadır. Bu durum birçok alanda olduğu gibi eğitim-öğretim çalışmalarına da bilgisayar bilimlerinin katkı vermesi gereğini ortaya koymaktadır.

Eğitimde genel manada, bireylere bilgi, tutum ve becerileri kazandırmakla birlikte yetenek ve davranışların da geliştirilmesi hedeflenmektedir [1]. Eğitim, birey için bilgilendikçe kendi davranışlarındaki isteyerek yaptığı değişikliklerdir. Başka ifadeyle bireyin kültür düzeyinin artışıdır. Eğitim bireyin doğumu ile başlar, hayatının her anında, ailede ve okulda hayatınca var olur. Eğitim çocukken hayata hazırlama, sonrasında ise kendini geliştirme işlevindedir. Bu süreçler insanın içinde ve dışında yaşadıkları ile şekillenir. Eğitim sonunda bir davranış değişimi, istenen amaç doğrultusunda olmalıdır [1].

Öğretim, eğitime göre daha planlı, kontrollü, düzenli ve belirgin bir zaman içerisindeki davranış değişikliklerini ifade eden bir kavramdır [1]. Farklı bir ifadeyle, öğretim; eğitim sisteminin okullar aracılığıyla belli planlarda gerçekleşen formal kısmıdır. Öğretim, bireye sunulan müfredat programını öğrenme ile gerçekleşirken, ilerleyen aşamalarda uzmanlaşmak anlamına gelir [1].

"Eğitimde ölçme ve değerlendirme nedir?" sorusuna cevap olarak; ölçme, herkes tarafından kabul edilen birimlerle karşılaştırma işlemi veya gözlemlenen faaliyetlerin sonuçlarının sayısal karşılık bulmasıdır. Değerlendirme ise ölçme sonuçları ile amaçlanan hedefler arasında ilişkilendirme aşamasıdır. Şekil 1.1'de eğitim sisteminde yer alan ölçme değerlendirme döngüsü görülmektedir.



Şekil 1.1. Eğitim Sistemi

Makine öğrenmesi, bilgisayarlar için girdi verileri, oluşan veri tabanlarını ele alırken; bunlar üzerinden ortaya çıkabilen anlamlı bilgiler üretme aşamalarını konu alan bilim dalıdır. Simon'a göre makine öğrenmesi, bilgisayarları doğru ve anlamlı şekilde programlayarak, bilinmeyen bağlantı ve ilişkileri ortaya çıkarma alt çalışma sahasıdır [15]. Başka bir tanım olarak da makine öğrenmesi, insanların matematik ve istatistik bilimleriyle yaptıkları veri yığınlarını anlamlandırma işlerini (çok daha kısa zamanda) çeşitli algoritmalarla bilgisayarlara yaptırma bilimidir. Bu amaçla kullanılabilecek birçok algoritma ilerideki bölümlerde daha ayrıntılı anlatılacaktır.

Araştırmanın amacı, eğitim öğretim faaliyetlerinde kaliteyi arttırmak ve başarıyı objektif bir şekilde ölçülebilir değerler ile ifade etmeye çalışmaktır. Bu amaç doğrultusunda, teknolojik ve güvenilir ölçme değerlendirme yöntemlerinden biri olan makine öğrenmesi algoritmaları tez verilerinde kullanılmıştır. Altı bölümden oluşan tez çalışmasının ikinci bölümünde literatür incelemesi, üçüncü bölümünde makine öğrenmesinde kullanılan metrikler ve algoritmalar, dördüncü bölümde materyal ve yöntem, beşinci bölümünde bulgular ve son bölümde ise sonuç ve tartışmalara yer

verilmiştir. Araştırmada, başarı ve başarılı olma kriterleri ölçme değerlendirme kıstasları ile ele alındığında tam olarak anlaşılamayan birçok sayısal veri ortaya çıkmıştır. Bu verilerin anlamlandırılmaya çalışılması, yapay zekâ özelinde makine öğrenmesi prensiplerine ve eğitim-öğretim faaliyetlerindeki başarı oranlarının değerlendirilmesine katkı sağlayacağı düşünülmektedir. Bununla birlikte tez çalışması, mühendislik ve eğitim alanlarını içinde barındırmaktadır. Disiplinler arası araştırmalara olan ihtiyacın gittikçe arttığı günümüzde bu çalışmanın makine kullanımı ve eğitim konusunda çalışmalar yapan kişilere de katkı sağlayacağı düşünülmektedir.

2. LİTERATÜR İNCELEMESİ

Literatür araştırması için yerli ve yabancı kaynaklar taranmış, önceki yıllarda yapılmış çalışmalar ve elde edilen sonuçlar incelenmiştir. Bu doğrultuda, eğitim bilimi ile makine öğrenmesinin birlikte ele alındığı çalışmalara ulaşılmıştır. Bu başlık içerisinde, eğitim başarısı ile birlikte makine öğrenmesini de kapsayan çalışmalar aşağıda yer almaktadır.

Tatlıoğlu ve Korkmaz (2015), tarafından gerçekleştirilen çalışmada Konya'da yer alan ilköğretim öğrencilerinin okul başarılarını olumsuz etkileyen nedenler belirlenmeye çalışılmıştır [14]. Bu çalışmanın verileri, 2011/ 2012 öğretim yılı içerisinde, Konya il merkezinde yer alan Milli Eğitime bağlı özel bir eğitim kurumunda eğitim gören öğrencilerden elde edilmiştir. Araştırma survey modelli bir çalışmadır. Çalışmanın örnekleminde; tüm ortaokul grubu sınıflarında eğitim alan 132 erkek ve 140 kız öğrenci olarak toplamda 272 kişi yer almaktadır. Katılımcılara 44 maddeden oluşan "Başarısızlık Nedenleri Anketi" isimli ölçek uygulanmıştır. Araştırma sonucunu olarak başarısızlık sebeplerinin "Aileden, eğitimcilerden, öğrencinin kendisinden, ders konularından, eğitim esnasında kullanılan tekniklerden ve son olarak da çevre faktörlerinden" olabileceğini sonucuna varılmıştır [14].

Ural ve Çınar'ın (2016) çalışmasında, ebeveyn eğitim seviyesinin, çocuklarının matematik derslerindeki başarısı arasında bir korelasyonun olup olmadığını ortaya çıkarılmaya çalışılmıştır [2]. Çalışmanın örnek uzayını, 2012/13 eğitim- öğretim yılı, Burdur ilinin merkezinde yer alan, 7. sınıfta eğitim gören 55 öğrenci oluşturmaktadır. Örneklemdeki katılımcılar tesadüfi olarak belirlenmiştir. Araştırmanın verilerini toplamak için hem öğrencilere hem de anne ile babalara araştırma ölçeği (anket formu) uygulanmıştır. Bununla birlikte öğrenciler tarafından doldurulan ölçekte 1. Döneme ait matematik dersi karne notlarının yazılması istenmiştir. Araştırma sonuçları, istatistiksel anlamda anlamlı olmamakla birlikte anne ile babanın ve özellikle annenin eğitim seviyesinin artması kendi çocuklarının matematik derslerindeki başarısını arttırdığı görülmüştür. Araştırma bulgularında dikkat çekebilen bir diğer husus, ortaokuldan mezun

olan babaların çocukları matematik dersi için başarısı en yüksek grup olarak belirlenmiştir. Bu durum ebeveynlerin eğitim seviyesi arttıkça çocuklarının da başarı oranının artacağına dair olan anlamlı ilişkiyi bozmaktadır [2].

Aslanargun ve Özakça'nın (2015) çalışmaları, akademik anlamda yüksek notlara sahip olan öğrencilerin başarılarında aile büyüklerinin katkıları ve bu katkıların düzeyi belirlenmeye çalışılmıştır [3]. Araştırma yöntemi olarak olgubilim deseni kullanılmıştır. Araştırmanın örneklemini, Düzce ilindeki 3 (üç) farklı okulda eğitim gören ve başarı ortalaması 5 (beş) olan 5 katılımcı (öğrenci) oluşturmaktadır. Çalışmanın örneklerimi, amaçlı örnekleme yöntemi içerisinde yer alan homojen örnekleme tekniği kullanılarak belirlenmiştir. Katılımcılardan yarı yapılandırılmış görüşme tekniği aracılığıyla veri toplanmıştır. Araştırma sonuçlarında, öğrenci başarısında, "yetenek ve içsel güdülenmenin" ailenin çocuklarına karşı olan ilgi ve alakasından daha anlamlı olduğu görülmüştür [3].

Olcay ve Döş (2009) çalışmalarında, ortaöğretim öğrencilerinin başarısızlığına sebep olan unsurları öğrenci açısından belirlemeyi hedeflemişlerdir [4]. Araştırma tarama modellidir. Çalışmanın örneklemini, Gaziantep ili Şahinbey ilçesinde ortaöğretim düzeyinde eğitim gören 690 öğrenci oluşturmaktadır. Örneklem tesadüfi yöntem kullanılarak belirlenmiştir. Araştırma verilerini toplamak için 45 sorudan oluşan anket formunun öğrenciler tarafından doldurulması istenmiştir. Araştırma sonuçları; "öğrencilerin %64'ünün öğretmenlerin anlattıklarından anlamaması", "öğrencilerin %63'ünün başarısızlık kaygıları", "öğrencilerin %55'inin verimli bir şekilde ders çalışma metotlarını bilmemeleri" ve "öğrencilerin %53'ünün ise bazı derslere ilgi duymaması" olmak üzere 4 farklı sebeplerden dolayı derslerde başarısız olduklarını ortaya koymaktadır [4].

Abbasoğlu (2020) çalışmasında, eğitimde veri madenciliği ile eğitim ortamlarındaki farklı ve yeni görülen veri türlerini araştırmak amacıyla yöntem geliştirmeyi, bununla birlikte öğrencileri ve öğrenim ortamlarını iyi anlamak amacıyla bu yöntemleri kullanmak isteyenler için güncel bir disiplin ortaya koymayı amaçlamıştır [5]. Araştırma verileri

Yalova ilinde ikamet eden, farklı sosyal ve ekonomik özelliklere sahip, ortaokul kademesinde eğitim alan 1395 öğrencinin 2019/2020 eğitim ve öğretim yılında "E-okul Sistemi" de var olan bilgileri üzerinden elde edilmiştir. Verilere uygulanan algoritma ve sınıflama teknikleriyle öğrencilerin yıl sonundaki genel başarı ortalamaları tahmin edilmiştir. Çalışma sonuçlarına göre, lojistik algoritmanın en iyi tahmine ulaştıran algoritma olduğu görülmüştür [5].

Çiftçi, Kaleli ve Günal tarafından 2018'de gerçekleştirilen çalışmada eğitmen ve öğrenci sayılarının oldukça çoğaldığı öğretim ekosisteminde başarıyı görebilmek için, paydaşların izlenmesinin gereği üzerinde durulurken; bu faaliyet metin madenciliği desteği ile ölçümlenmeye çalışılmıştır [6]. Şimdiye kadarki araştırmalar çoğunlukla öğrenci çevresinde incelemelerde bulunduğundan, bu çalışma eğitmen üzerine olmuştur. Veri madenciliği çatısında, Gazi üniversitesi öğrenci gruplarıyla anket çalışmaları yapılmış, eğitmen başarısı tahmin edilmeye çalışılmıştır [6].

Web ve diğerleri tarafından gerçekleştirilen 2020 tarihli uluslararası araştırmada ise günlük hayatta fazlaca kullanılmaya başlanan makine öğrenimi ile öğrenciler arasındaki fayda, zarar veya rekabet ilişkisi incelenmiştir [7]. Öğrenciler ve derin öğrenme karşı karşıya getirilirken, oluşabilecek artılar, sürprizler ve belirsizlikler tartışılmaya çalışılmıştır. Her iki tarafın birbirine destek vermesi için gerekli sosyal ve hukuki sorumluluklar çalışma içerisinde ele alınmıştır [7].

Alaybeyoğlu ve Solmaz (2018) tarafından yapılan çalışmada öğrencilerin ders öncesinde, ders sırasında ve ders sonrasında bulundukları durumlar kategorize edilmiştir. Bulanık mantık prensipleriyle ele alınan kural tabanlarına yönelik girdiler uzman sistem tarafından ağırlıklandırılarak ders başarı çıktısına dönüştürülmüştür [8].

Aydoğan ve Karcı (2018) tarafından gerçekleştirilen çalışmada, Bingöl Üniversitesi'ne bağlı Genç Meslek Yüksekokulu öğrencileri örneklem uzayında anket çalışması ile veri toplanmış ve bunlar özelinde öğrencilerin başarıları bulunmaya çalışılmıştır [9]. Oluşan veri tabanı üstünde Python dili ile makine öğrenmesi algoritmaları

çalışılmış; o dönem YÖK'ün almış olduğu sınav geçişin iptali kararının da doğruluğu görülmüştür [9].

Durak ve Bulut (2023) çalışmalarında, bilgisayar programlama alanında eğitim gören öğrencilerin, program başarım seviyelerini belirleme ve sınıflara ayırmaya yönelik makine öğrenmesi modellerini oluşturmayı hedeflemişlerdir [10]. Araştırmanın örneklemini Türkiye'deki lise ve üniversite seviyesinde eğitim gören, bilgisayar programlama derslerinden en azından bir tanesini almış öğrenciler oluşturmaktadır. Araştırma verileri toplanırken, Türkiye'nin bütün bölgelerinin temsil oranı gözetilmiş ve çevrimiçi formlar kullanılmıştır. Araştırma da katılımcılardan kişisel bilgilerin ve 4 (dört) farklı ölçeğin yer aldığı formların doldurulması istenmiştir. Farklı algoritmaların denendiği çalışmada en yüksek değerin %79 ile "lojistik regresyon" olduğu bildirilmiştir [10].

Pek ve arkadaşlarının (2022) yaptığı çalışmada, risk altında bulunan öğrencileri belirlemek ve öğrenci başarısızlığını en aza düşürmek amacıyla, makine öğrenimi algoritmaları kullanılmış ve bulunan performans sonuçlarını tartışılmıştır [11]. Araştırmada sunulan makine öğrenimi algoritmalarının performansları çeşitli metrikler kullanılarak ölçülmüş ve tahmin sonuçlarının en iyisini veren algoritmalar birleştirilerek oluşturulan hibrit model bu araştırmada sunulmuştur. Araştırma için, öğrencilerin demografik özelliklerinin ve akademik bilgilerinin yer aldığı veriler toplanmıştır. Hibrit model, veri özelliklerinin önemini ortaya koymak için, iki farklı veri kümesi kombinasyonu ile test edilmiştir. İlk kombinasyon için, demografik ile akademik veriler kullanılmıştır ve hibrit modelin doğruluk oranı %94,8 olarak bulunmuştur. İkinci kombinasyonda yalnızca akademik veriler kullanılmıştır ve hibrit modelin doğruluk oranı %94,8 olarak elde edilmiştir [11].

Pinto ve arkadaşlarının (2023), makine öğreniminin yüksek öğrenimi nasıl değiştirdiğini belirlemek amacıyla yaptığı derleme çalışmasında son beş yıl içerisinde yayınlanan çalışmalara ulaşmak için sistematik bir literatür taraması yapılmıştır [12]. Bu çalışma verileri, PRISMA tekniği kullanılarak elde edilmiştir. PRISMA yönergeleri takip

edilerek “SCOPUS” indeksli 1887 yayın içerisinden 171 makalenin konu ile ilgili olduğu belirlenmiştir. Araştırma bulguları, makine öğreniminin, yükseköğretimde yaygın olarak araştırılan uygulamasının, öğrencilerin akademik performanslarının ve istihdam edilebilirliğinin tahmini ile ilgili olduğunu göstermektedir. Bu derleme çalışmasından elde edilen sonuçların, ChatGPT çağında, makine öğrenimi ve yapay zekâ teknolojilerindeki gelişmelerin akademik bütünlük bakımından tehlike oluşturmaksızın üniversitelerde ne şekilde ve nasıl kullanılabileceğini keşfetmek için hem araştırmacılara hem de uygulayıcılara katkı sağlayacağı düşünülmektedir [12].

Pande (2023) tarafından gerçekleştirilen çalışmasında ise, bireysel ve toplumsal başarı için eğitimin önemini tartışılarak, Hint eğitim sistemindeki geleneksel öğretim yöntemlerindeki zorluklar ile büyük nüfus nedeniyle öğrenci performansının izlenmesindeki zorluklar karşılaştırılmıştır [42]. Çalışma içerisinde öğrenci ilerlemesini değerlendirmek için standartlaştırılmış yöntemlere ve öğrenci performansını etkileyen faktörleri belirleme karmaşıklığına vurgu yapılmıştır. Araştırma, öğrenci başarısını etkileyen çeşitli faktörleri incelemekte, görselleştirme teknikleri ve K-en Yakın Komşular, Lojistik Regresyon ve Destek Vektör Makinesi gibi makine öğrenimi yöntemlerini kullanarak performansı tahmin etmektedir. SVM modeli, veri kümesi için %84,37 doğruluk elde etmiştir [42].

Kavitha ve arkadaşları (2023) tarafından gerçekleştirilen çalışmada ise eğitim veri madenciliğinin eğitim verilerindeki gizli bağlantıları ortaya çıkarabileceğini ve öğrencilerin akademik başarılarını tahmin edebileceği konusu ele alınmıştır [43]. Kampüs yerleştirmesi aracılığıyla öğrencilerin iş bulma olasılığını tahmin etmek için yeni bir makine öğrenme modeli önermektedir ve bunun için okul sonu sınav notları ve kümülatif not ortalamalarını kullanmaktadır. Rastgele Orman, Destek Vektör Makineleri, Naif Bayes, Sinir Ağı ve K-en Yakın Komşu gibi çeşitli makine öğrenme yaklaşımları, 1105 lisans son sınıf öğrencisinin veri seti kullanılarak değerlendirilmiş ve %75 ila %94 arasında sınıflandırma doğruluğu elde edilmiştir. Tahminler, cinsiyet, sınav notları, öğretim dili ve akademik geçmiş gibi öğrenci bilgilerine dayanmaktadır. Çalışma, yükseköğretim için öğrenme analitiği modelleri geliştirmekte veri odaklı araştırmaların

önemini vurgulamaktadır, bu da karar alma süreçlerini etkileyebilir. Son olarak, başarısız kampüs yerleştirmesi riski altındaki öğrencilerin erken tahmininde etkili makine öğrenme tekniklerini belirlemektedir [43].

3. GENEL BİLGİLER

Bu başlık altında çalışmada kullanılan algoritmalar, bu algoritmaların arka planında çalışan istatistiki bilgilere ve denklemlere, kullanılan araçlar ve genel bilgilere yer verilmiştir.

3.1. Veri Bilimi-Makine Öğrenmesinde Kullanılan Metrikler

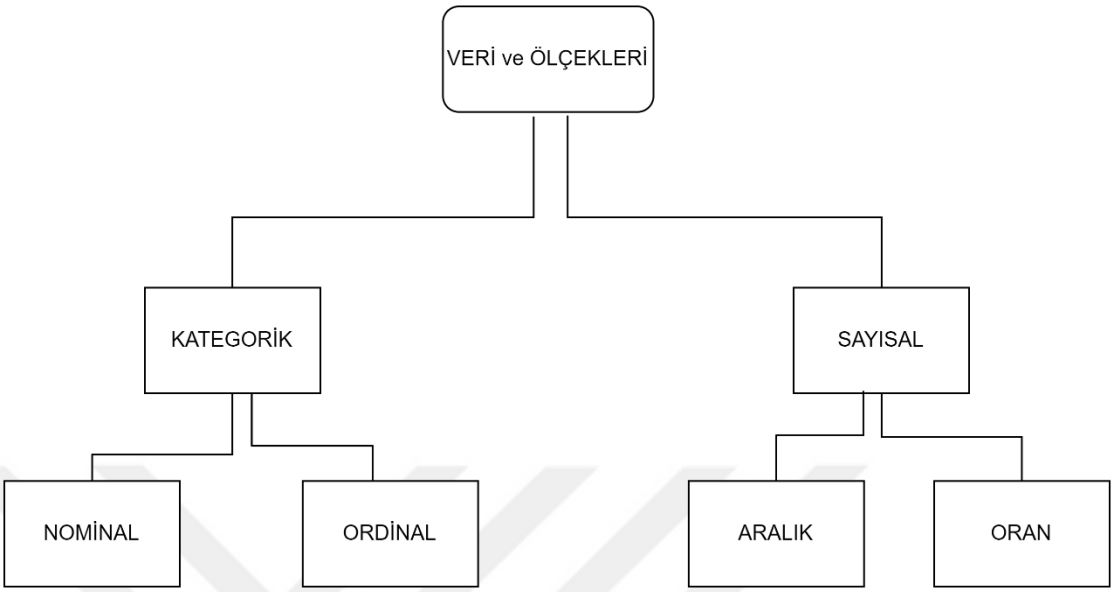
Veri bir konuyla ilgili toplanan veya önümüze gelen tüm bilgiler topluluğudur. Veri, veri tabanlarında iki değişik şekilde bulunur. İlişkisel veri tabanı veya büyük veri (big data) şeklindedir. Veri bilimi ile ilgili proje kim tarafından yapılır denirse veri bilimcidir. Proje nedir? sorusunun karşılığı, veri bilimi projesidir ve veri bilimi projelerinde programlama dilleri ele alınıp, bunlar yardımıyla analiz yapılır. Bu analizin sonucunda da bir modelleme ortaya çıkarılır. Genellikle bu analiz sırasında R veya Payton programlama dilleri kullanılmıştır. Örneğin araç fiyat tahmin modeli projesinde istatistiksel tahminleme veya makine öğrenmesi algoritmaları kullanılarak bir model oluşturulmuştur. Bu modelde amaç, araç fiyat tahmininde etkili olabilecek model, yapılan kilometre değeri v.b. farklı unsurların(parametrelerin) katsayılarının değerleri bulmaktır. Bu bakış açısıyla bir nevi parametre tahmin yöntemleri ya da optimizasyon yöntemleri olarak ele alınabilecektir. Fiyat bilgisini "*y*" değişkeni ile gösterdiğimiz $y = (b0) + (b1).x1 + (b2).x2 +... + (bp).xp$ ifadesinde; *x*'ler fiyatlandırma unsurlarını, *b*'ler ise tahmin yöntemi tarafından bulunmak istenen katsayıları göstermektedir. Bir fiyat eşittir yazıldığında herhangi bir araç için tahmin modeli olarak katsayılara değer verilmesi şeklinde aracın tahmin edilen fiyat bilgisi çıkartılabilir. Eğer bütün bu yapılmış proje bir sitede çalışabilir hale getirilirse, bu modeli "Deploy" etme veya "Deployment" şeklinde adlandırılır. Bu çalışma bize göre veri bilimi projesi adını alırken, takım lideri tarafından makine öğrenmesi projesi, şirketin ceo su tarafından ise yapay zekâ projesi olarak genel çatı anlamı verilir.

Uzmanlık çalıştığımız alanda, çalıştığımız sektörde olası yapılabilecek tüm hataları yapmak ve belli bir tecrübeyle ortaya çıkmak olarak yorumlanabilir. Makine öğrenmesi

projesinin ortaya konma aşamaları olarak; betimleme, burada ne olmuş? Sorusunun cevabının resmedildiği safhadır. Diğer safha teşhis- tanı safhasıdır, olan şeyler neden olmuştur? nasıl olmuştur? ifade edildiği bölümdür. Tahminsel analitik (prodüktivite) bir anlamda gelecek ile ilgili tahminlerde bulunma safhasıdır, bu safhada örnek olarak üretimdeki parçalar ne zaman bozulabilir? kurumsal müşteriler ne zaman terk edebilir? şeklinde, tahmin edilip önlemek için neler yapılabilir? yarın hava durumu ne olacak şeklinde ifadelere cevap aranmaktadır. Son bir safha olarak yönergeli analitik safhası vardır ki; bu aşamada ne olmalı? nasıl olmalı? sorularına cevap aranır. Yine bu safhada ileri istatistik yani makine öğrenmesi katma değer ifadeler ve ne yapmamız lazım, görünen o ki şu parametre de müşteriler terk etmeyi önleyebiliyoruz ama başkası parametrelerde önleyemiyoruz diye sonuçlar ortaya koymaktır. Veri tabanları ilişkisel veri tabanları veya big data şeklinde bulunduğunu söylemiştik R ya da Python programlama dilleri ile bu verileri belli bir derleyici ortamına taşırız ve orada analiz etme imkânı buluruz. Makine öğrenmesi algoritmaları dediğimiz algoritmalar yardımıyla da parametre tahmin yöntemleri geliştirip sonucunda optimizasyon yöntemleri olarak sonuç ifadelerimizde yer verebiliriz. Aslında makine öğrenmesi teknikleri ile yapmış olduğumuz modelleme de bulmak istediğimiz kavramla ilgili daha önce ortaya konan katsayıları anlamlandıran, hep o katsayıların değişik koşullarda hangi değerleri elde edeceğini bulmaya çalışmaktır. Makine öğrenmesi projeleri gerçek hayat problemlerine dayandırmalıdır ki makine öğrenmesi, metin madenciliği gibi bütün veri birimi süreçlerini sonunda genel anlamda yapay zekâ ifadesini taşımalıdır; burada makine öğrenmesi dediğimiz işleme tekniği olarak yer alır.

Veri okuryazarlığı, günlük yaşamda karşımıza çıkan verileri hemen o andaki yorumlama yeteneğidir. Veri okuryazarlığı her tipten veriyi tanımlayabilme ve istatistik biliminin metrikleri ile anlamlandırma kabiliyeti olarak da açıklanabilir. Elimizde doğru veriler olsa dahi, onları sunabiliyor olmamız gerekir. Popülasyon ve örneklem; popülasyon bizim ilgilendiğimiz ana kitlemizdir insan topluluğu, olmuştur örneklem ise bu kitle yani popülasyondan seçilen alt kümedir. Bir başka değişle popülasyon yerine popülasyonu temsil eden alt küme olarak örneklem seçilir. Bu seçim çok iyi temsil kabiliyetine sahip olacak şekilde yansız ve objektif olarak yapılmalıdır. Tabakalı

örnekleme ve rastgele örnekleme şeklinde örneklem belirleme metotları mevcuttur. Gözlem birimi (Observation Unit); araştırmada incelediğimiz birimlerdir. Herhangi bir seçim anketinde mikrofon uzatılan herkes bu araştırma için gözlen birimidir. Makine öğrenmesi veri tablolarında; her satır bir gözlem birimi, her sütun ise bir değişken olarak ifade edilebilir. Bu anlamda gözlem birimi araştırmanın ana odak noktasıdır. Değişken ise; birimden birime farklı değer alan niceliklerdir. Değişken türleri dediğimiz zaman sayısal değişkenler nicel ve kantitatif olarak ikiye ayrılırken, kategorik değişkenler ise nitel ve kalitatif olarak yine ikiye ayrılır. Bu anlamda cinsiyet, erkek veya kadın olarak kategorik bir değişken olup, erkek ve kadın bu değişkenin sınıflarıdır. Bir başka örnekte ise yine kategorik bir değişken olan arabadaki vites türü için manuel ve otomatik seçenekleri bu değişkenin sınıflarıdır. Ölçek türü denince; sayısal değişken için aralık veya oran şekli, kategorik değişken için nominal veya ordinal şekli söz konusudur. Sıcaklık değişkeni, artı ve eksi sayısal değişkenler örneği olarak ifade edilebilir. Başlangıç noktası "0-sıfır" olmayan sayısal değişken olduğundan ölçek türü aralık olur. Aynı şekilde başlangıç noktasını "0-sıfır" kabul eden sayısal değişkenler ölçek olarak oran kullanır. Arabaların kilometre değeri veya fiyat bilgisi, başlangıç değeri sıfır olduğu için ölçek türü oran şeklindedir; ancak sıcaklığın örneğin -4 ve +16 arasında değişmesi ölçek türünü aralık şeklinde yapar. Kategorik değişkenler, karakter formatındadır. Kategorik değişkenler ifade edilirken sınıflar arasında fark yoksa nominal değişken (kadın erkek arasında fark olmaması gibi), ancak sınıflar arasında fark varsa ordinal (askerdeki rütbeler gibi) olarak adlandırılır. Şekil 3.1'de veriler ve ölçek sınıfları görülmektedir.



Şekil 3.1. Veriler ve Ölçeklerin Sınıfları

Merkezi eğilim ölçüleri başlığında; Aritmetik Ortalama, Medyan, Mod, Çeyrekler ve Standart Sapma kavramları yer almaktadır. Aritmetik ortalama; bir veri grubunda var olan tüm veri değerleri toplamının, veri sayısına bölünmesi ile bulunan merkezi eğilim ölçüsüdür.

Medyan ise veri grubunu küçükten büyüğe veya büyükten küçüğe sıralandığında, tam ortadan kalan ve veri grubunu eşit iki parçaya ayıran veri değeridir. Gruptaki veri sayısı tek ve çift ise medyan; Denklem (1)'deki gibi bulunur.

$$n\ tek\ ise\ medyan\ = \left(\frac{n+1}{2}\right).\ terim$$
$$n\ çift\ ise\ medyan = \frac{\frac{n}{2}.\ terim\ +\ \left(\frac{n}{2}+1\right).terim}{2} \quad (1)$$

Aritmetik ortalama, veri dağılımın simetrik (birbirlerine yakın değerlerde) olduğu bilindiğinde kullanılırsa anlamlı olur. Grubun veri dağılımı simetrik değilse, medyan kullanımı daha iyi bir gösterge olur. Simetrik olma özelliği, değişken içinde aykırı değerlerin bulunmamasını işaret eder.

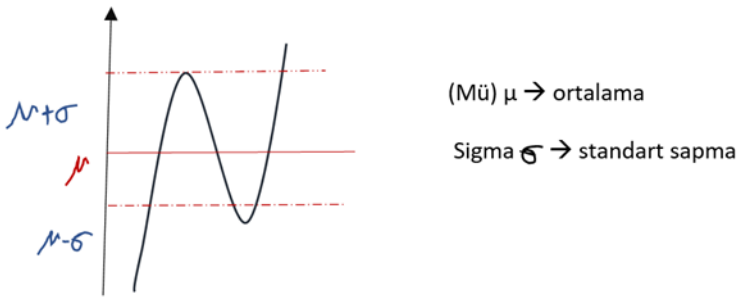
Bir veri grubunda en fazla tekrarlanan değere Mod denir. Küçükten büyüğe sıralanmış bir veri grubunu 4 bölüme ayıran değerlere Çeyrekler denir. Çeyrekler, veri grubunun ortası ile birlikte ortasının sağı veya solu ile ilgili bilgiler sağlar. Hem merkezi hem de dağılım ölçüsüdür. Serilerde temsil yönünü kavrama ve doğru kullanımı için merkezi eğilim ölçüleri çok önemlidir.

Merkezi Dağılım Ölçüleri başlığında ise; Değişim Aralığı, Standart Sapma, Varyans, Çarpıklık ve Baskınlık kavramlarından bahsedilecektir. Değişim Aralığı, en ilkel dağılım ölçüsü olup; bir veri grubundaki en yüksek değerden, en aşağıdaki değerin çıkarılması ile bulunur. Değişim aralığının küçük olduğu yerde verilerin birbirine yakın olduğu (bir açıdan) daha adil olduğu; değişim aralığı büyük olduğunda ise verilerin birbirinden daha uzak olduğu, daha farklı değerler olduğunu gösterir. Standart sapma ise aritmetik ortalamadan sapmaların bir nevi ortalamasıdır. Denklem (2)'de formülü verilmiştir. Şekil 3.2.'de yer alan grafik üstünde incelenebilir.

$$s = \sqrt{\frac{1}{n}} . \sum_{i=1}^{n} (x_i - \bar{x})^2 \qquad (2)$$

X_i: Serideki değerler; $\bar{x}$: Aritmetik Ortalama

(x_i-$\bar{x}$) = Ortalamadan uzaklık (SAPMA)



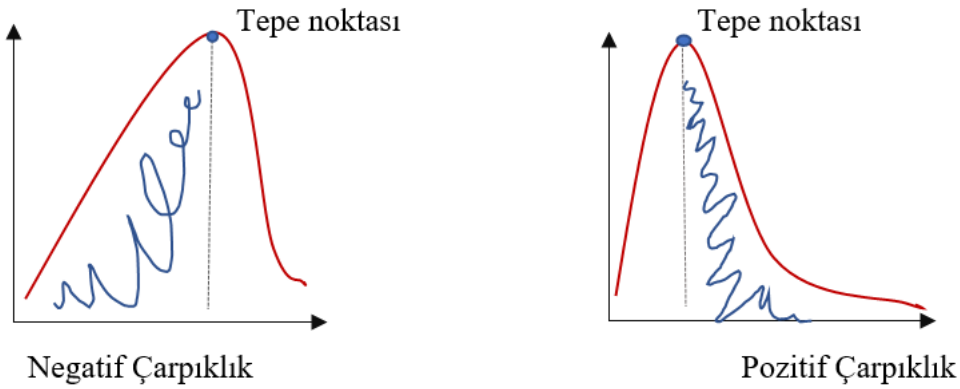
Şekil 3.2. Standart Sapma

Varyans ise Standart sapmanın karesidir. Birden fazla değişkenin dağılımını birbirleriyle kıyaslamak istendiğinde kullanılabilir. Çarpıklık ise, bir değişkenin dağılımının simetrik olamayışıdır. Denklem (3)'de formülü verilmiştir.

$$S=\sqrt{\frac{1}{n}.\sum_{i=1}^{n}(x_i-\overline{x})^2}$$
$$s^2=\frac{1}{n}.\sum_{i=1}^{n}(x_i-\overline{x})^2 \quad (3)$$

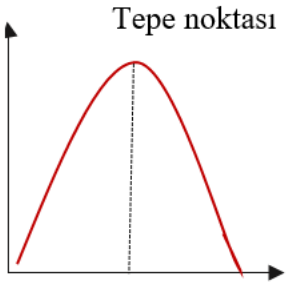
3.1.1. Çarpıklık

Şekil 3.3. ve 3.4.'de görüldüğü gibi;



Şekil 3.3. Negatif ve Pozitif Çarpıklık

Normal olanı tepe noktasının ortada olduğu durumdur. Bu halde dağılım simetriktir.

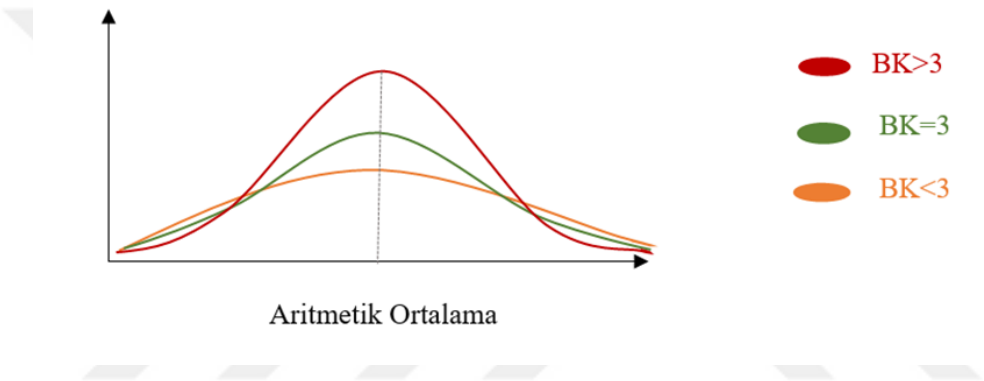


Şekil 3.4. Normal Çarpıklık (Simetrik Dağılım)

Dikkat edilirse, çarpıklık olan verilerde aritmetik ortalama yerine medyan ile temsil etmek daha doğru olur.

3.1.2. Baskınlık

Baskınlık ise dağılımın baskınlığını-sivriliğini Şekil 3.5.'deki gibi gösterir.



Şekil 3.5. Baskınlık

İstatiksel düşünce ekolleri, veri okuryazarlığından yola çıkarak veri bilimine giden yolu belirten yol işaretçileridir. Bireyin veriye ilk dokunduğundan, son aşama olan veri ile ilgili yorumlar yapabilme, çıkarım yapabilme süreçlerin modelleyen teorik çalışmalardır. Baskınlık katsayısı sıralı modellerin, sıralı standart sapmalarının sıralı kuvvetlerine bölümü şeklinde bulunur. B.K. baskınlık katsayısı olmak üzere;

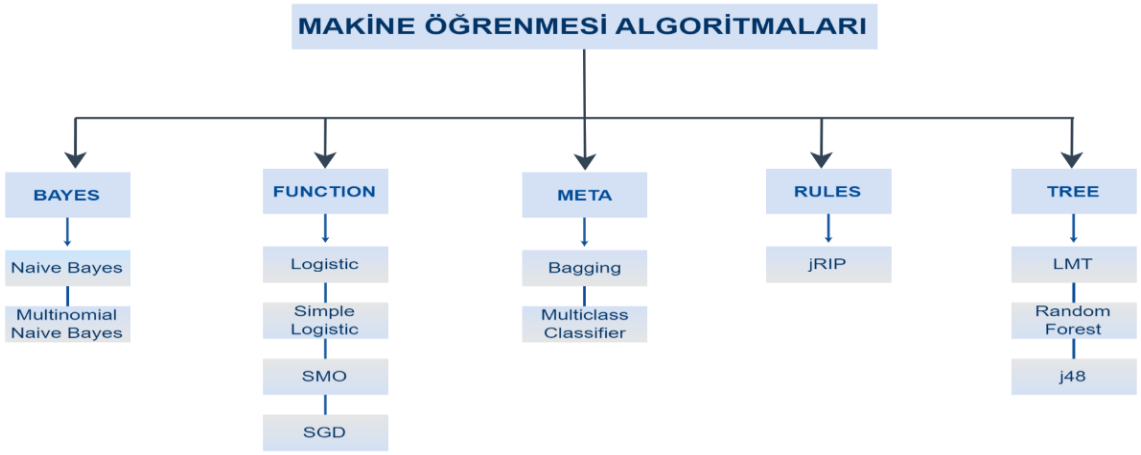
B.K.=3 ise dağılım standart normal dağılıma uygundur. Yeşil grafik ile;
B.K.>3 ise dağılım Sivridir. Kırmızı grafik ile;
B.K.<3 ise dağılım Basıktır. Sarı grafik ile gösterilmiştir.

İstatiksel düşünce ekolleri, veri okuryazarlığından yola çıkarak veri bilimine giden yolu belirten yol işaretçileridir. Bireyin veriye ilk dokunduğundan, son aşama olan veri ile ilgili yorumlar yapabilme, çıkarım yapabilme süreçlerin modelleyen teorik çalışmalardır.

3.2. Makine Öğrenmesi

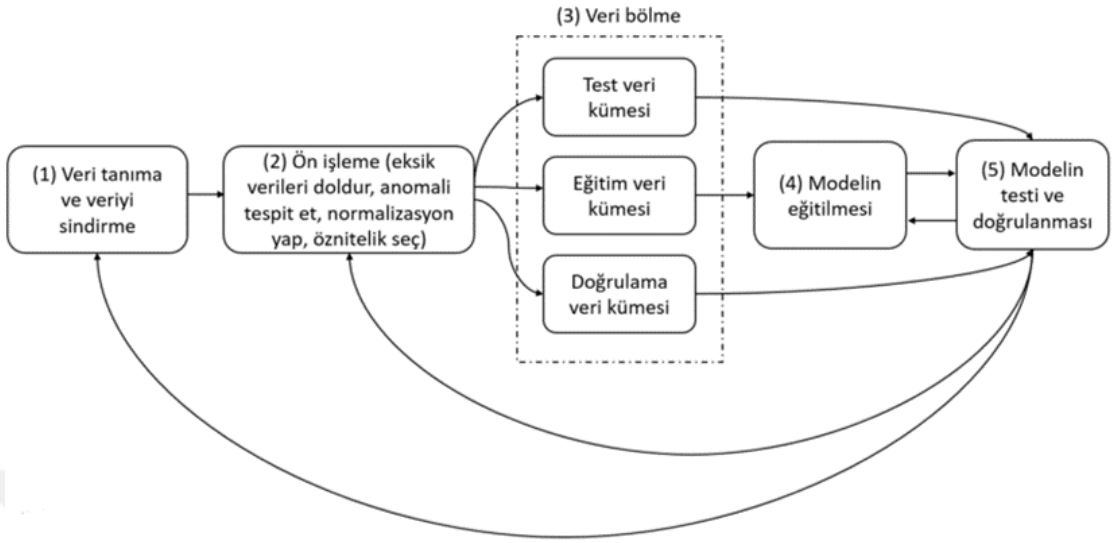
İnsanlık tarihi boyunca insanlar yaptıkları işleri kolaylaştırmanın çarelerini araştırmışlardır. Günlük işlemlerini ya da görevlerini kolay bir şekilde yapabilmek için gözlemlerle, önceki tecrübelerini yeniden düşünerek çeşitli araçlar üretmeyi sürdürmektedirler. Tüm bunlar esasen insanoğlunun ihtiyaçlarına çare bulmak için akıllarını kullanarak geliştirdikleri araçlardır. Başka bir ifade ile insan yaratıcılığı ve aklının ürünleri denebilir. Bu ürünler, günümüzde gelişen teknoloji ile çok daha yüksek seviyelere ulaşmıştır. Birçok iş kolunda insanın yerini gittikçe makineler almakta ve makineler bazı işleri insan kadar yapabilmektedir. Bu ifade aslında makine öğrenmesinin de tanımıdır. Makine öğrenimi, bilgisayarların verilerden öğrenmeyi nasıl yaptığını anlamaya ve geliştirmeye çalışan bilimsel bir disiplindir. Makine öğrenmesi, verilerden ilişkileri anlayarak istatistik öğelerini bilgisayar biliminden öğelerle birleştirerek verileri yönetmek için algoritmalar geliştiren yapay zekanın bir alt dalıdır [41].

Makine öğrenmesi ile geliştirilen uygulamalar insanların işlerini kolaylaştırmanın yanı sıra, iş kazası gibi risklerin insanlara vereceği zararı minimuma indirme için de kullanılabilir. Makine öğrenmesinin kullanıldığı alanlardan bazıları Şekil 3.6'da gösterilmektedir.



Şekil 3.6. Makine Öğrenimi Algoritmaları

Makine öğrenmesi çalışma prensibine Şekil 3.7'de yer verilmiştir.



Şekil 3.7. Makine Öğrenmesi Çalışma Prensipleri

3.2.1. Naive Bayes Algoritması

Naive Bayes, olasılık hesabına dayalı olan ve makine öğreniminde en sık kullanılan algoritmalardan biridir. İsmini İngiliz matematikçi Thomas Bayes'den almıştır. Bayes teoreminden yola çıkılarak, bir sınıftaki bir özelliğin olma olasılığının, diğer özelliklerin olma ya da olmama olasılığından bağımsız olduğunu varsayan bir sınıflandırma algoritmasıdır. Her bir özellik için var olma olasılığı (koşullu olasılık) hesaplanır ve olasılık sonucu en iyi olan özellik sınıfına göre sınıflara ayırma yapılır [20]. Yani bir özellik için her özelliğin olma olasılığı; olduğu varsayılan bir durumun çıktılarının mevcutaki örneklem uzaydan eksiltilmesi ile yeni durumda koşullu olasılık olarak hesaplanır. Makine öğrenimi içindeki basit sınıflandırıcılardan biridir, çok sınıflı ya da dengesiz dağımlı veri setlerinde de yüksek performans gösterir. Bayes teoremi Denklem (4)'de verilmiştir.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{4}$$

$P(A|B)$: B olayı varken A olyının olma olasılığı(A nın B koşullu olasılığı)

$P(A)$: A olayının olasılığı

$P(\mathrm{B})$: B olayının olasılığı

3.2.2. Multinomial Naive Bayes Algoritması

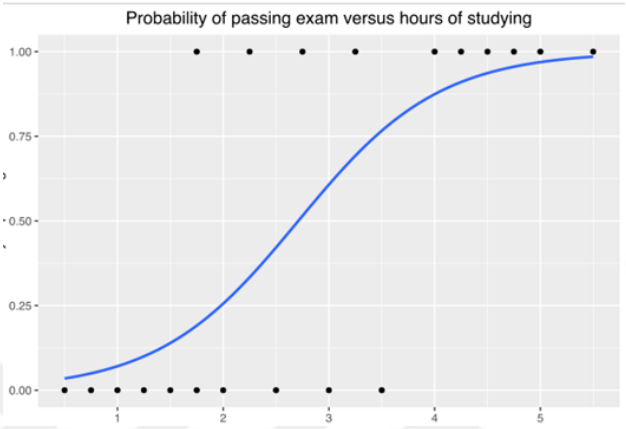
Multinomial Navie Bayes, yüksek tahmin başarısı ve kolay modellenmesi ile metin sınıflandırmada yaygın olarak kullanılan sınıflandırıcılardan biridir. Çok terimli Naïve Bayes, kelimenin bir belgede kaç kez geçtiğini ifade eden terim sıklığı kavramı üzerinde çalışır. Bu model, kelimenin bir belgede geçip geçmediği ve o belgedeki sıklığı olan iki kavramı ele alır. Yeni bir haber makalesinin kategorisini tahmin ederken, makaledeki tüm kelimelerin olma olasılıkları ile değerlendirme yapar. Fakat bir kelimenin bir belgedeki terim sıklığının yüksek olması her zaman bu kelimenin doğru sınıflandırmaya götüreceği anlamına gelmeyebilir, kelime potansiyel olarak belgeye bir anlam katmayabilir. Dolayısıyla multinomial sınıflandırıcı uygulanırken öncelikle bu tür kelimelerin metinden ya da belgeden kaldırılması gereklidir [16]. Denklem (5)'de gösterildiği şekilde bir belgenin w kategorisine ait olma olasılığı ($\mathrm{P}(w)$), tüm belgelerde w belirtecinin sıklığının ($\mathrm{n}(w)$), tüm belgelerdeki toplam terim sayısına ($\mathrm{n}(W)$) oranı ile hesaplanır.

$$\mathrm{P(w)} = \frac{\mathrm{n(w)}}{\mathrm{n(W)}} \tag{5}$$

3.2.3. Lojistik Regresyon

Lojistik regresyon, hem doğru denklemi şeklinde hem de doğru denklemi dışında logaritma vb. denklem şekillerine (Şekil 3.8.'deki gibi) uyan verileri sınıflandırmak için kullanılan gelişmiş bir regresyon tekniğidir. Verileri ikili yanıtlarla modellemek için yaygın olarak kullanılır. Yanıt ikili olduğunda, 0/1 şeklini alır, 1 genellikle bir başarıyı ve 0 bir başarısızlığı gösterir. Ancak 1 ve 0'ın alabileceği gerçek değerler, çalışmanın amacına bağlı olarak büyük ölçüde değişir. İkili sınıfladırmada sınıflardan biri 1 diğeri 0 olarak etiketlenebilir. Lojistik regresyon, verilen girdi değerini alıp girdiyi ağırlık değeri ile çarparak uygulanan bir makine öğrenme tekniğidir [17]. Lineer regresyonda noktaların grafik izdüşümleri üzerinden, model olarak en yakın doğruyu bulabilmek amaçtı.Lojistik regresyonda ise noktalarla doğruların arasını minimize edebilmek için, logaritma fonksiyonunun (Denklem 6) grafiği yaklaşımı sergilenmektedir.

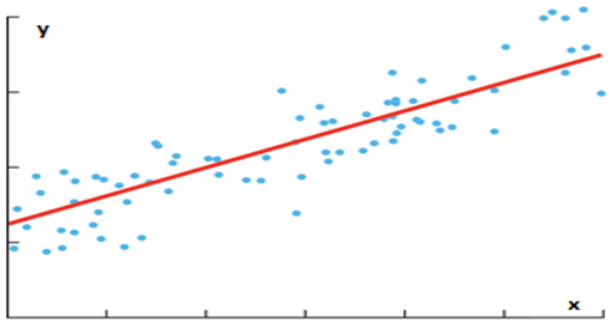
$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}} \tag{6}$$



Şekil 3.8. Logaritmik Fonksiyon

3.2.4. Simple Lojistik

Simple lojistik sınıflandırıcı, veriler doğrusal olduğunda çok iyi performans gösterir, ancak doğrusal olmayan veya karmaşık verilerle performans zayıf olabilir. Bir veri kümesindeki eksik verileri işleyemez [18]. Verileri noktalarının grafik izdüşümü üzerinden doğru formülüne; başka deyişle noktalara en yakın olan doğru formülüne dönüştürme mantığı ile Şekil 3.9'da görüldüğü gibi çalışır.



Şekil 3.9. Lojistik Regresyon

3.2.5. Sıralı Minimum Optimizasyon

Sıralı Minimum Optimizasyon (Sequential Minimal Optimization – SMO), karesel programlama çözücüleri aracılığıyla denetimli makine öğrenme modeli olan destek vektör makinelerini eğitmek için yaygın olarak kullanılan bir sınıflandırma algoritmasıdır. Çeşitli yaklaşımlar arasında SMO, SVM (destek vektör makinelerin)'de bulunan ikinci dereceden programlama problemini çözmenin bir yoludur. SMO tarafından kullanılan yinelemeli algoritma, optimizasyon problemini bir dizi daha küçük alt probleme böler. Bu QP tabanlı alt problemler daha sonra sayısal QP optimizasyonundan tamamen kaçınılarak analitik olarak çözülür [19]. SVM de iki sınıf arasında en fazla aralığı yakaladığı çizgiyi tercih etmek suretiyle, gelen veriyi sınıflandırırken; SMO çalışırken "kernel" denilen çekirdek tercihi ile verileri farklı farklı şekillerde ayırarak belli bir sınıfa ait sayar.

3.2.6. Kstar

K-Star algoritması veri madenciliği çatısında, nesneleri belirleme ve kontrol mekanizmaları gibi farklı alanlar için örnek taban yaklaşımı şeklinde uygulanır. Grafiklerin farklı olarak özel analizi, kullanıcı gizliliğini korurken hassas grafiklerden istatistik yayınlamak için yaygın olarak kullanılır. Ancak mevcut algoritmaların çoğu, güvenilir bir veri küratörünün grafiğin tamamını elinde tuttuğu merkezi bir gizlilik modelindedir. Bu model, küratörün güvenilirliği ve veri ihlali olasılığı gibi bir dizi gizlilik ve güvenlik sorununu gündeme getirdiğinden, algoritmaların, grafiğin tamamını hiçbir sunucunun tutmadığı daha merkezi olmayan yerel bir modelde dikkate alınması arzu edilir. KStar algoritması, ihtimalleri var olan dönüşümlerden rasgele alarak entropi ölçümleri (olasılıkların kendi logaritmaları ile çarpımları toplamı) yapar [20]. Dönüşümler arasında uzaklık metriği entropi olunca, gerçek sonuçlu değerler, sembol değerler ve eksik değerler arasında anlaşılır belirginlik oluşur [21].

3.2.7. Jrip

JRip yaygın olarak kullanılan makine öğrenmesi algoritmalardan biridir. Çalışma prensibi, sınıflar büyüdükçe analiz edilmesi ve gittikçe azalan hata oranları kullanılarak sınıf için ilk kurallar dizisinin oluşturulmasıdır. Bu algoritma, eğitim verisindeki belirli

bir veri setinin tüm örneklerini sınıflandırmak ve o sınıfın tüm üyeleri için geçerli olan bir dizi kural aramak için kullanılır. Aynı kural arama-bulma işlemi mevcut tüm sınıflar için tekrarlanarak inceleme tamamlanır [22].

3.2.8. PART

PART, C4.5 (J48) ve RIPPER algoritmaları üzerinden gelişen kuzen versiyonu olarak bölgesel karar ağacı algoritmasıdır. PART algoritmasının asıl farklılığı, kurallarını oluşturmak için J48 ve RIPPER gibi tüm veri kümesini ağaçlarla ifade etmek zorunda olmamasıdır. Bu tür kural tabanlı sınıflandırma algoritmaları öncelikle özniteliklere göre büyük bir ağaç inşa eder. Sonrası aşamalarda bu ağaçtan budama yaparak, karar verilecek kurala ulaşır. Son olarak en az hataya haiz kural içeren özelliği seçerek sonuç karar ağaç yapısına ulaşır. PART algoritmasında ağaç çok daha büyük oluşacağından, karar ağaç aşaması hata oranı yüksek çıkabilir [23].

3.2.9. Decision Stump

Decision Stump (Karar Kütüğü), tek seviyeli bir karar ağacından oluşan bir makine öğrenme modelidir. Yani, terminal düğümlerine anında bağlanan bir dahili düğüme sahip bir karar ağacıdır. Bir karar kütüğü, yalnızca tek bir girdi özelliğinin değerine dayalı olarak bir tahminde bulunur. Bazen bunlara 1-kurallar da denir [24].

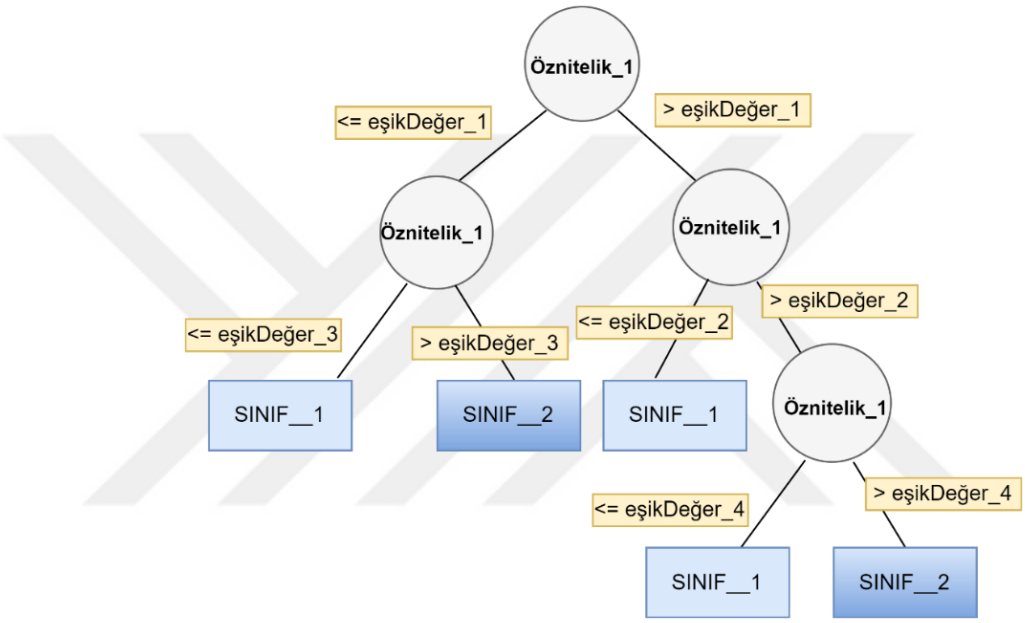
3.2.10. J48

J48, eksik değerler, kural türetme, sürekli öznitelik değer aralıkları gibi özelliklere sahip olan bir makine öğrenimi sınıflandırıcısıdır. Bu algoritma, veriye benzersiz bir kimlik kazandırma kurallarının oluşturulmasında kullanılır. Bu sınıflandırıcıyı kullanmanın amacı, çok yönlülük ve doğruluk arasında bir denge sağlayana kadar karar ağacını kademeli olarak dallandırmaktır [25].

3.2.11. LMT

Lojistik Regresyon Ağacı (Logistic Regression Tree – LMT) temelde lojistik regresyon ile karar ağacı öğrenme metotlarının kombinasyonu şeklindedir. Standart bir karar ağacı yapısındadır. Yaprakları farklı lojistik fonksiyonlar ile ilişkilendirilir. Öznitelik

seçimi ile önemli öznitelikleri seçebilme özelliğine sahiptir. Karar ağaçlarında olduğu gibi düğümlerin alt düğümleri vardır. Bu düğümlerdeki lojistik hesaplamalar neticesinde öznitelik değeri ve eşik değer karşılaştırılması ile dallanmalar yapılır. Sayısal öznitelikler için Şekil 3.10'de görüldüğü gibi öznitelik değeri eşik değerinden küçük ise sol düğüme, büyük ise sağ düğüme doğru dallanma gerçekleşir [26].



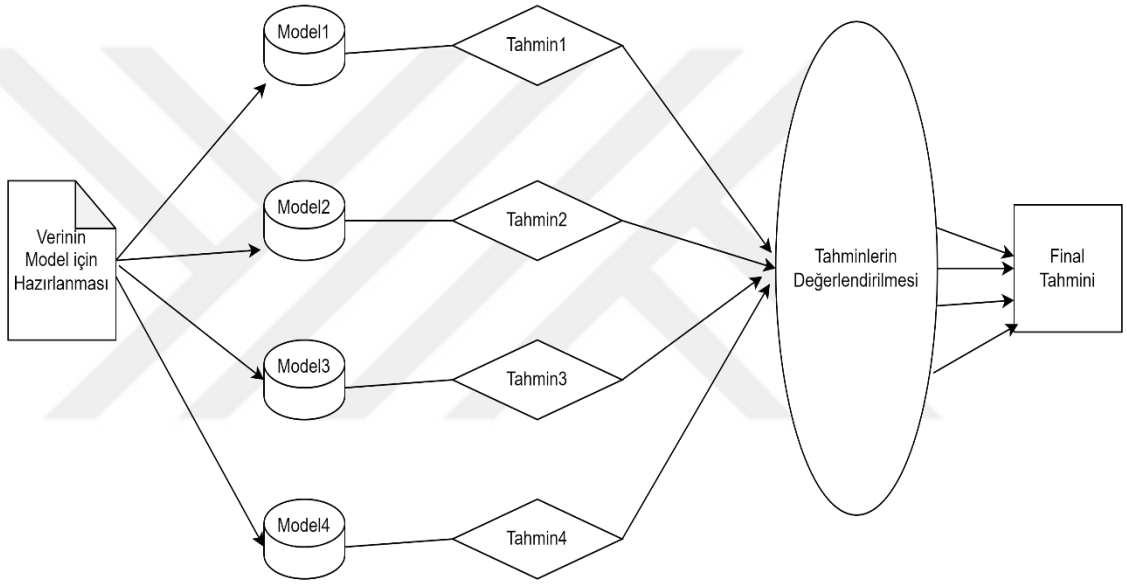
Şekil 3.10. Lojistik Regresyon Ağacı (LMT)

3.2.12. Random Forest

Rastgele Orman (Random Forest – RF), bir veri setindeki farklı alt kümelerde birden fazla karar ağacı kullanan ve tahmin doğruluğunu iyileştirmek için sonuçların ortalamasını alan bir makine öğrenimi algoritmasıdır. Yani tek bir karar ağacına bağlı olmak yerine, sonuçları elde etmek için birden çok ağaçtan tahminler toplar. Her ağaç düğümünde, bir koşul, giriş verilerinin bir veya daha fazla özelliği ile karşılaştırılır. Her ağaç için bir sınıf tahmini vardır ve en fazla tahmin edilen sınıf algoritma sonucunda asıl tahmin edilen sınıftır. RF dengesiz veri setlerinde çok iyi performans gösterir ve sınıflandırma hatalarının sayısı düşüktür [27].

3.2.13. Bagging

Bagging birlikte (ensemble) öğrenme yöntemi ile çalışan bir algoritmadır. Şekil 3.11'te gözlemlendiği, gibi birliktelik içindeki her model veri setinden bir miktar veri ile eğitilir. Modeller birbirlerinden bağımsız oldukları için aynı anda eğitim gerçekleşebilir. Birliktelik modelinin yeni bir test örneğinin sınıflandırılması, sınıflandırıcıların kararları birleştirilerek belirlenir [28]. Amaç birden fazla sınıflandırıcıyla daha yüksek performans elde etmektir.



Şekil 3.11. Ensemble Learning (Topluluk Öğrenmesi)

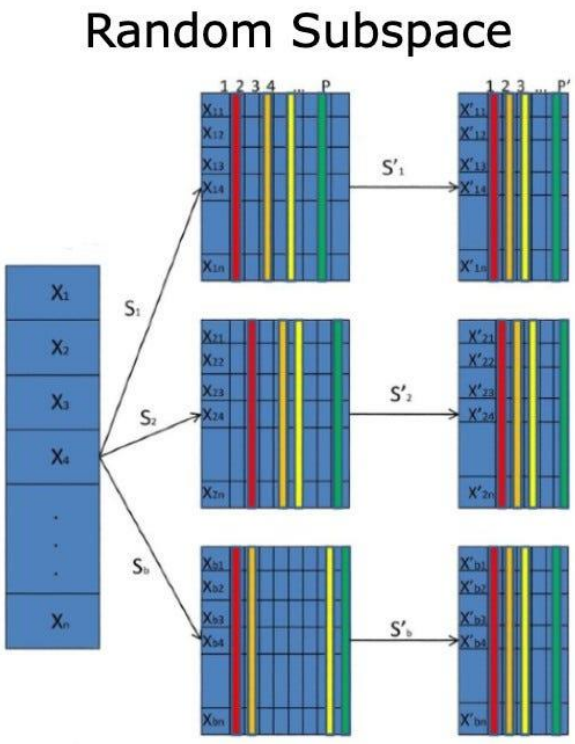
3.2.14. MetaAdaBoostM1(AdaBoost)

MetaAdaBoostM1 de birlikte (ensemble) öğrenme yöntemi ile çalışan bir algoritmadır. Sınıflandırma ve regresyon çalışmaları için uygundur. Çalışırken tüm verilere aynı özgül ağırlığı atar. Sonrasında, bu verilerin karar ağacındaki konumuna göre ağırlıklarını değiştirir. Başka meta yöntemlere göre tahminleme hızı daha iyidir, küçük hafıza tüketir, uygulanması kolaydır ve tercih edilir. Eğitim örnekleri az hata yapılması önceliği ile seçilir. Diğer meta yöntem Bagging'de her tekrarda, eğitim bölümüne alınma ihtimali aynı iken Boosting'de döngüler arası örnekleme seçim ihtimali farklılaşır. Böylece yanlışların düzeltilmesine odaklanma olur. Sistem doğru regresyona doğru itilir.

AdaBoost, güçsüz sınıflandırıcı topluluğu ile çalışır. Etkisiz ayırıcılar veri içinden oluşup, tüm veri seti boyunca öznitelikler kapsamında pozitif ve negatif sonuçların değerleri ile oluşturulur. Hataları en düşük olan zayıf nitelikler diğer gruplardan elenir ve kararlar bu gruplar üzerinden verilmeye çalışılır [29].

3.2.15. Random SupSpace

Random Subspace (rastgele alt uzay) tekniğinde, veri setinde var olan değişken sayısından daha az değişken random olarak seçilir ve Bagging metodu ile oluşturulan ağaç dallanmaların oluşumu sağlanır. Leo Breiman tarafından Bagging tekniğine rastgelelik üzerinden yeni bir yorum katmak amacıyla Şekil 3.12'de görüldüğü üzere geliştirilmiş bir teknolojidir [30].



Şekil 3.12. Random Subspace Algoritması

3.2.16. K-means

Kümeleme, bir veri kümesinde birbirine en yüksek düzeyde benzeyen verilerin aynı bölgede ve mümkün olduğunca az benzeyenlerin farklı bölgelerde yer almasıdır.

Sınıflandırmada belli sınıflar varken, clustering (kümeleme) de sınıflar yok, eldeki verilerin hangileri birbirine yakındır? bu ayrıştırma vardır. K-means kümelemede kullanılan ve kümenin mesafe merkezini odakta tutan bir algoritmadır. Çok kullanılması basit oluşu ve merkezi veriye yakınlık ile çalışmasındandır. Kümenin bölüt sayısına göre, her grup için yeni merkezler seçilir ve bunlara metrik uzaklıkların en yakın oluşu üzerinden kümelere ayırma başlar. K-Means daha ziyade Öklid mesafesine göre çalışır. Ancak, Manhattan, Minkowski, Chebyyshev mesafeleri de özel verilerde kullanılabilir [31].

3.2.17. X-means

K-Means algoritması küme merkezlerini belirlerken, bu işlem için belli bir kıstas bulunmaması ve keyfiliği kendi handikabı olarak su yüzüne çıkmaktadır. Bu sebeple oluşan bölütler her seferde farklı ve objektiflikten uzak kalmaktadır. Yani başarı, merkez noktalarının rastgele iyi seçimine bağlı kalmaktadır. Bu durum ihtimale bırakılmamalıdır. X-means algoritması bu seçimlerle ilgili kara verme mekanizmaları uygulayarak daha iyi kümeleme yapılmasını sağlamak amacıyla kuzen algoritma şeklinde geliştirilmiştir [33].

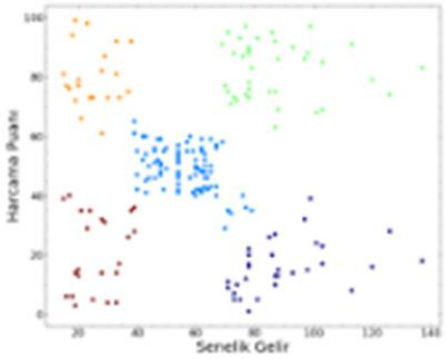
3.2.18. Expectation Maximization

EM (Expectation-Maximization), beklentilerin en iyi hale getirilmesi anlamına gelmektedir. İki tür değer kabul eder; nominal değerler için olasılık, numerik değerler için aritmetik ortalama ve standart sapma ile çalışır. Olasılıksal olarak üyelik belirlerken, kalite için logaritmik hesap yapan log-likelihood algoritmasını yürütür. Eksik olan verili problemlere ve parametre bulma işlemlerini başarıyla yapar [31].

3.2.19. Hierarchical Clustered

Hiyerarşik kümeleme algoritması (Şekil 3.13), K-means ile başlayan merkezi değerlerden en yakın uzaklıktaki verileri kümeleme mantığı ile çalışan algoritmalardan farklıdır. İki temel yaklaşımı vardır. Aşağıdan yukarıya (agglomerative) ve yukarıdan aşağıya (divisive) şeklindedir. Aşağıdan yukarıya yaklaşımda önce bir veri bu veriye en yakın arkadaş veri, sonra bu ikiliye en yakın diğer ikili grup, bu dörtlüye en yakın dörtlü grup şeklinde ilerler. Yukarıdan aşağı yaklaşım ise tüm veri yığını bütün ele alıp

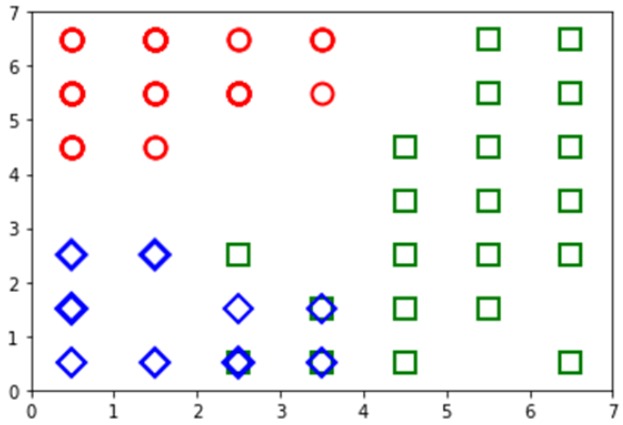
maksimum uzaklıktan her defasında ikiye bölerek devam ediyor. Bu algoritmada diğerlerinden farklı olarak direk kümeleme sayısını değil; maksimum ve minimum kümeleme sayısını kullanıcı belirlemektedir [32].



Şekil 3.13. Hierarchical Clustered

3.2.20. Self Organization Maps

Kendi kendini organize eden harita anlamındadır. Kohonen's SOM olarak literatürde geçmektedir. Aslında bir yapay sinir ağı uygulamasıdır. Her neron (düğüm) bir özelliği tutmaktadır. Veri ünitesi (nöron ile veri tutuluyor) olarak adlandırılır. Denetimsiz öğrenme olarak iki aşamada çalışır. Birinci aşamada training (competit process veya vektör quantization), ikinci aşamada ise mapping (classify) uygulamaktadır. Quantization, parçalı olarak modelleme ve quent (zaman bölümlerine)'lara ayırma şeklindedir. Düğümleri renkler ile ayrı ayrı ifade edersek, anlaşılması daha kolaydır (RGB 3D modelleme Şekil 3.14) [34].



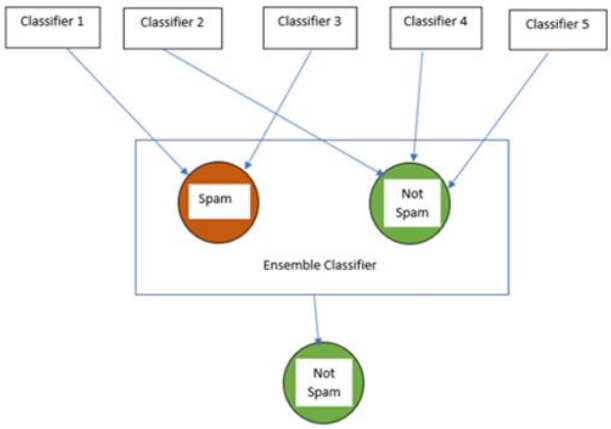
Şekil 3.14. Örnek SOM Renkler ile Kümeleme

3.2.21. LVQ

Öğrenme Vektörü Nicelemesi (LVQ), model bazlı, denetimli öğrenme yapan sınıflama algoritmasıdır. LVQ özel yapay sinir ağıdır. Örneklere ait sınıfın bilinemediği ama model ile karşılaştırılabildiği takviyeli öğrenme ile çalışıp sınıflandırma durumları için uygundur. SOM gibi matris vektörlere haritalama yapar. SOM dan farkı öğrenme katsayısı sıfıra yaklaşırsa, öğrendiklerini unutabilir.

3.2.22. Vote (Çoğunluk Oylaması)

Oy Verme Topluluğu (Vote), çoklu algoritmaları bir arada çalıştırabilen ensemble learning (topluluk öğrenmesi) algoritmasıdır. Topluluk tekniği, birden çok modelden gelen tahmin değerlerini toplayarak, prototip başarısını yükseltmek için kullanılan makine öğrenmesi algoritmasıdır. Oylama topluluğu, tekli algoritma çıktılarından edinilen fikirleri bir araya toplayarak, tüm algoritma çıktılarından faydalanan topluluk güncel yöntemlerden biridir. Çalışma tekniği Şekil 3.15'te gösterilmiştir.



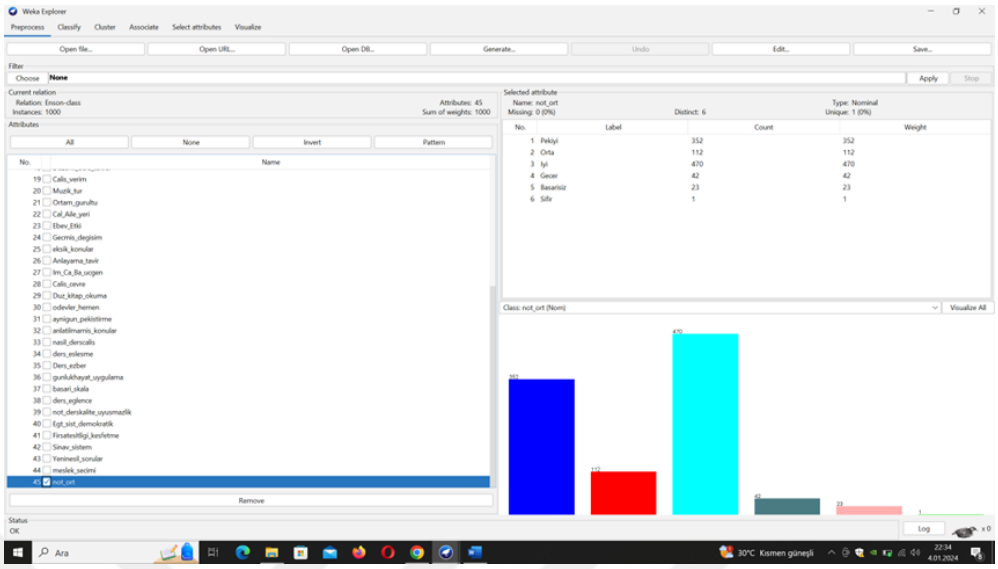
Şekil 3.15. Vote Algoritması Çalışma Tekniği

3.3. Makine Öğrenmesinde Kullanılan Araçlar

3.3.1. Weka

Weka, veri işleme görevlerinde kullanılan, açık kaynak kodlu bir makine öğrenimi aracıdır. Weka, Yeni Zelanda Waikato Üniversitesi'nde Genel Kamu Lisansı'nda geliştirilmiştir. Java programlama dili kullanılarak yazılmıştır ve Grafik Kullanıcı Arayüzü (GUI) veya Java Uygulama Programlama Arayüzü (API) aracılığıyla erişilebilir. Weka ile veri ön işleme, görselleştirme, sınıflandırma, regresyon, kümeleme, özellik seçimi gibi çeşitli işlemler yapılabilir [35]. Ayrıca bu işlemlere ek olarak cross-validation, eğitim-test veri bölümleme gibi özellikler de Şekil 3.16 ve 3.17'de görüldüğü üzere Weka aracılığıyla yapılabilir.

Şekil 3.16. Weka Programı Giriş Ekranı



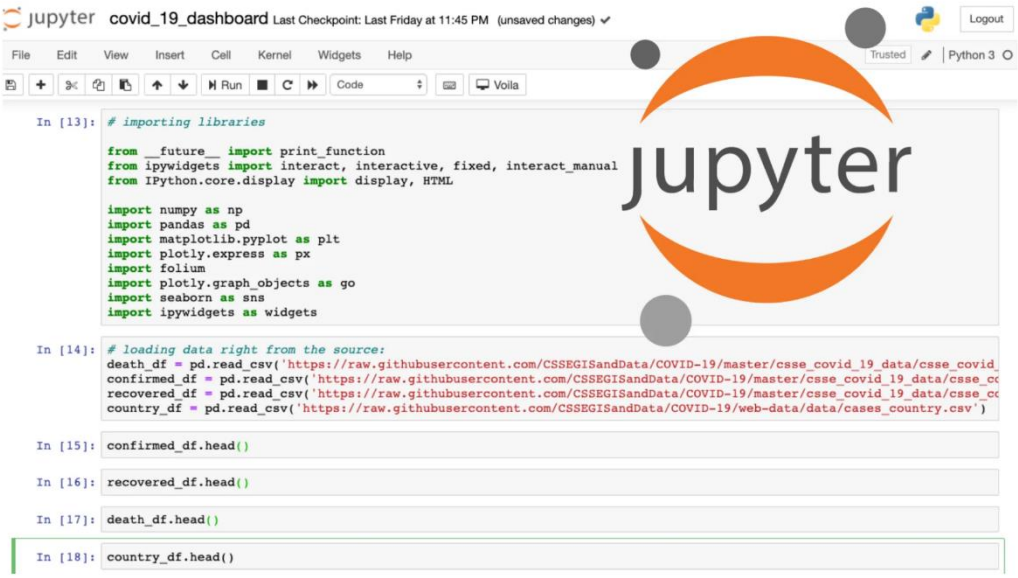
Şekil 3.17. Weka Explorer Ekranı

3.3.2. Jupyter Notebook

Anaconda (Şekil 3.18'de), Python veya R'de kod yazmak için çeşitli ortamlar sunan özel olarak tasarlanmış ücretsiz bir yazılımdır. Entegre geliştirme ortamları (Integrated Development Environments-IDE), kod geliştirmeye katkı sağlayan uygulamalardır. Ayrıca, Anaconda ile entegre geliştirme ortamlarını daha ergonomik hale getirmek için farklı araç setleri (toolkit) indirilebilir. Bu araç setleri ile önceden oluşturulmuş işlevler kullanılabilir. Bu işlevler, kitaplık adı verilen bölümlerde yeniden gruplandırılmıştır ve Anaconda aracılığıyla kolayca indirilebilir. Python'u bilgisayarınıza kurmanın ve kullanmanın birçok farklı yolu vardır, ancak Anaconda, kurulumunda en önemli kitaplıkları ve platformları içeren basit, grafiksel bir kullanıcı arabirimidir (Graphical User Interface – GUI). Anaconda ayrıca tüm bu kitaplıkları güncel tutma sürecini de basitleştirir. Böylece Python'u farklı platformlar, kitaplıklar ve işlevlerle ayrı ayrı kurmak yerine, Anaconda bunu tek bir kurulumda yapabilir. Jupyter Notebook bu platformlardan biridir.

Jupyter Notebook, varsayılan web tarayıcınızı kullanan web tabanlı bir IDE'dir. Jupyter Notebook'un esnek bir kullanımı vardır; bir dosyası içinde yazılan her kod bloğu ayrı ayrı çalıştırılabilir. Aynı Jupyter Notebook dosyası içinde farklı metin türleri kullanılabilir.

Böylece kod çıktıları, görseller, denklemler ve normal metinler tek bir yerde bulunur. Web tabanlı olduğu için kod dosyalarını başkalarıyla paylaşmayı da kolaylaştırır [13].

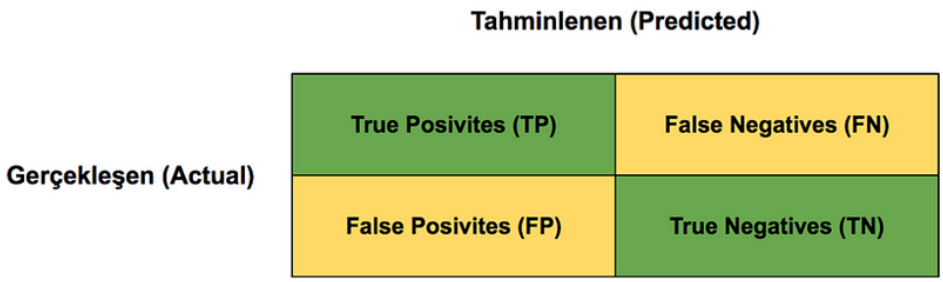


Şekil 3.18. Jupyter Notebook Ekranı

3.4. Makine Öğrenmesinde Kullanılan Performans Metrikleri

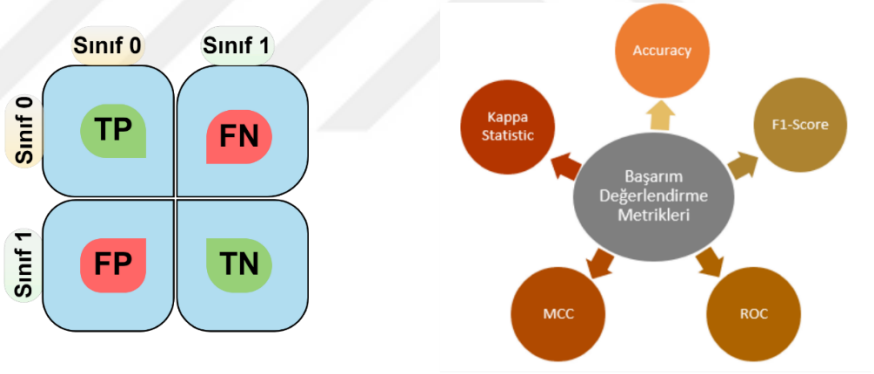
Performans ölçme metrikleri sınıflandırma ve kümeleme gibi makine öğrenmesi sonuçlarını anlamlandırmak için kullanılır [36]. Buradan sonuca ulaşılıp ulaşılmadığı belirlenmesi için bu metriklere ihtiyaç vardır. Her bir metriğin ayrı kriterleri sınamak gibi birer fonksiyonu vardır.

Öncelikle bu metrikler için; güncel eldeki veriler true (doğru) ve false (yanlış) olarak ayrılır. Makine öğrenmesi sonucu tahminlenen verileri ise pozitif (doğru tahminlenen) ve negatif (yanlış tahminlenen) şeklinde ikiye ayırırız. Makine öğrenmesi faaliyeti sonrasında karmaşıklık matrisi (confusion matrix) denilen bir nevi tablo ile doğru olup doğru tahminlenen verilere true-pozitif (TP), doğru olup yanlış tahminlenen verilere false-negatif (FN), yanlış olup doğru tahminlenen verilere false-pozitif (FP), yanlış olup yanlış tahminlenen verilere true-negatif (TN) denilerek tüm veriler sunulmuştur. Şekil 3.19'da karmaşıklık matrisi görülmektedir.



Şekil 3.19. Karmaşıklık Matrisi

Karmaşıklık matrisi kullanılarak diğer değerlendirme metrikleri (precision, recall, F1-score, accuracy, Cohen kappa, vb.) hesaplanabilir. Karmaşıklık matrisi ve yapılan araştırmadaki makine öğrenmesi sınıflandırma ve kümeleme için kullanılan metrikler Şekil 3.20'de gösterilmiştir.



Şekil 3.20. Karmaşıklık Matrisi ve Başarım Ölçütleri

3.4.1. Doğruluk

Doğruluk (Accuracy – Acc), makine öğrenmesinde veya tahmin modellerinde doğru sınıflandırılan veri sayısının, tüm veri sayısına bölümü ile elde edilir. En geçerli, en çok bakılan başarı kriteridir ve Denklem (7)'teki formül ile hesaplanmaktadır [37].

$$Acc = \frac{(TP + TN)}{(TP+TN+FN+FP)} \tag{7}$$

Denklem (4)'te ifade edildiği üzere veri setinin dengeli olmadığı durumlarda doğruluk oranına modeli değerlendirmek için tek başına bakılmaz ve diğer metriklere de ihtiyaç duyulur.

3.4.2. F1-Skor

Doğruluğa göre F1-Skoru dengesiz dağılım gösteren verilere daha uygundur. F1 skorunu hesaplamak için Denklem (8) ve Denklem (9)'da ifade edilen Duyarlılık (Recall) ve Kesinlik (Precision) metrikleri uygulanır ve Denklem (10)'daki gibi hesaplanır. Kesinlik pozitif olarak tahminlenen değerlerin gerçekten kaç adedinin pozitif olduğunu göstermektedir. Duyarlılık ise "Positive" olarak tahmin etmemiz gereken işlemlerin ne kadarını "Positive" olarak tahmin ettiğimizi gösteren bir metriktir. F1 skoru ise Duyarlılık ve Kesinlik değerlerinin harmonik ortalamasıdır. F1 skoru ile uç değerlerin performans başarısını etkilemesinin önüne geçilmiş olur.

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

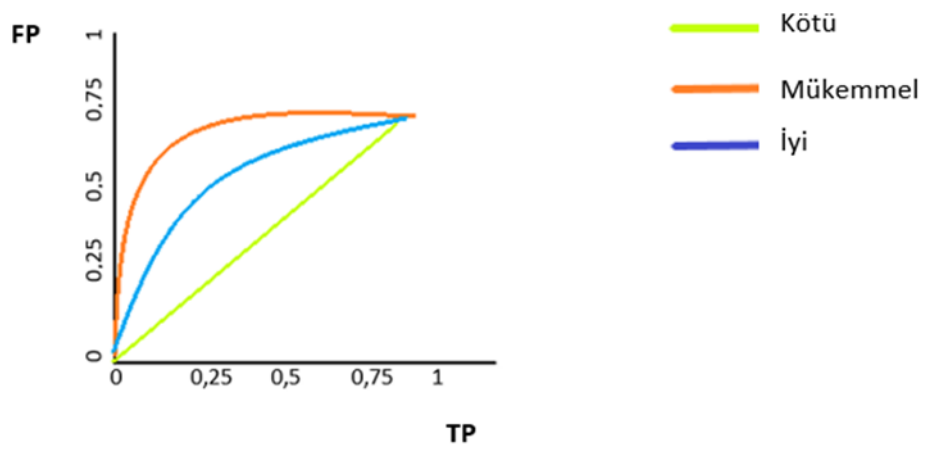
$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{10}$$

3.4.3. ROC Eğrisi

ROC eğrisi, özellikle dengesiz verilerde başarılı ölçümler yapan sınıflandırıcı başarısının iki boyutlu ölçüsüdür. Yani pozitif verileri doğru sınıflandırma olasılığının, negatif verileri yanlış sınıflandırma olasılığına bölümünün karşılaştırmalı çizimi olarak anlatılır [38]. Şekil 3.21'de görüldüğü üzere ROC eğrisini bir grafik ile gösterildiğinde x

ekseni FP, y ekseni ise TP değerlerine karşılık gelmektedir ve ROC eğrisi altında kalan alan 1'e yaklaştıkça model başarımı da doğru orantılı olarak artmaktadır.



Şekil 3.21. ROC Eğrisi

3.4.4. MCC

Bir tahminlemenin başarısını MCC (Matthews Correlation Coefficient) metriği ile değerlendirme denildiğinde karmaşıklık matrisindeki bütün değerler hesaplamaya katılmaktadır. Aktüel veriler ile çıktı verileri arasındaki korelasyona göre, Denklem (11)'de gösterilen şekilde hesaplama yapılmaktadır. Minimum sınır -1, maksimum sınırı ise +1'dir ve değer 1'e yaklaştıkça performansın arttığını, tahminlerin doğru olduğunu göstermektedir [40].

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{11}$$

3.4.5. Kappa

Kappa istatistiği, tahmin edilen değer ile gerçek değer arasındaki uyumu göstermektedir. Ayrıca bu uyumun şans eseri olabileceğini de değerlendirmektedir. MCC metriği gibi -1 ve +1 arasında değer alır ve 1'e ne kadar yakında modelin o kadar başarılı olduğunu gösterir [39]. 0 ise var olan uyumun şans eseri olabileceğini ifade eder.

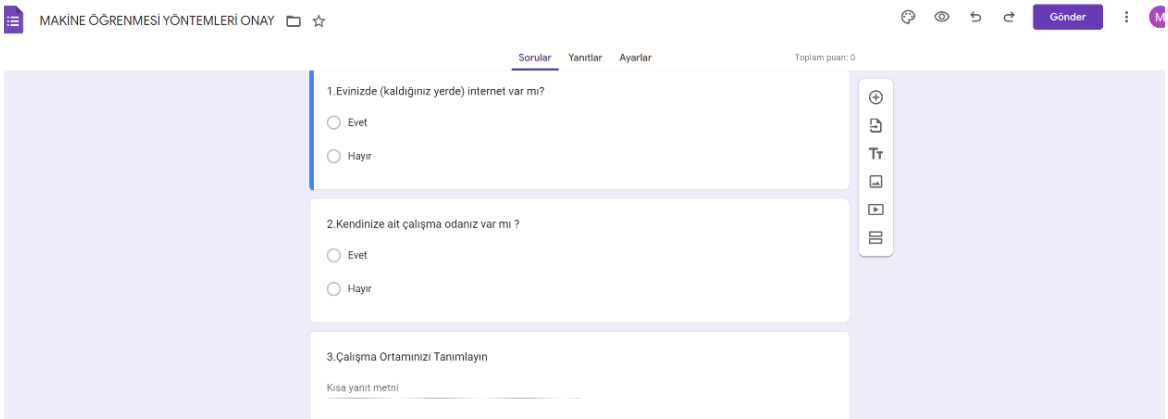
4. VERİ MATERYALİ VE ÇALIŞMA YÖNTEMİ

Bu çalışmada oluşturulan veri seti, veri setini işlemek için kullanılan teknikler ve geliştirilen algoritmalar bu bölüm altında incelenecektir.

4.1. Veri Materyali

Bu çalışmada veriler, yaşları 14 ile 18 arasında değişen lise düzeyi öğrencilerine nicel paradigmalı tarama modeli olarak, ders başarısını etkileyecek soruların bulunduğu (yazılı ve Google forms olarak-Şekil 4.1'de) iki ayrı şekilde anket uygulaması şeklinde sahadan toplanmıştır.

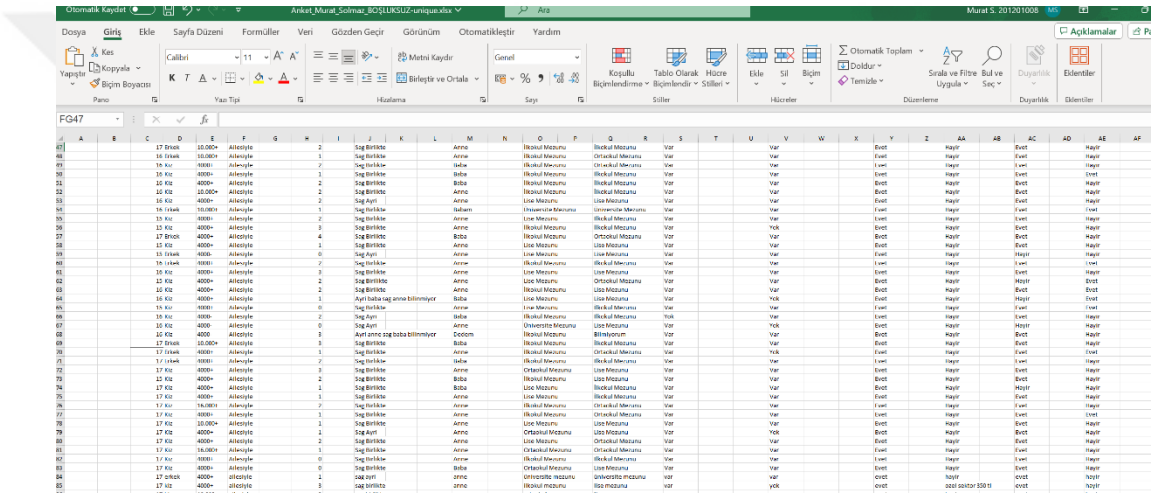
Anketlerde açık uçlu, evet-hayır ve çoktan seçmeli 46 soru bulunmaktadır. Bu anket uygulamaları öncesinde Manisa İl Mili Eğitim Müdürlüğünden gerekli izinler alınmıştır. Anket çalışması Manisa İli Turgutlu İlçesi M.E.B. bağlı Turgutlu Anadolu Lisesi, Senem Aka Anadolu Lisesi, Turgutlu Lisesi ve Niyazi Üzmez Lisesi olmak üzre 1000 öğrencileriyle sınırlı tutulmuştur. Esasen ders başarısının üniversite sınavı vasıtasıyla taçlandığı Lise Düzeyi Öğrencileri konu olarak alınırken (Eğitim Verisi –Doğrulama için Üniversite öğrencileri), İlkokul ve Ortaokul Öğrencileri kapsam dışındadır.



Şekil 4.1. Google Forms Anket Örneği

4.1.1. Veri Seti ve Özellikleri

Anket çalışmaları sonucu oluşan veri setimiz 46 soruya karşılık 46 sütundan, 1000 öğrenciye karşılık 1000 satırdan oluşan 46000 veri içermektedir. Veriler önce Excel'de bir tablo haline getirilmiş, sonra üzerinde ön işleme çalışmaları yapılıp .csv formatına çevrilerek (Şekil 4.2'de), Weka platformunda Sınıflandırma (Classification) ve Kümeleme (Clustering) işlemleri Makine Öğrenmesi Algoritmaları (KNN, Logistic Regression, SVM, Naive Bayes, Decision Tree, Random Forest vb.) deneyleri gerçekleştirilmiştir.



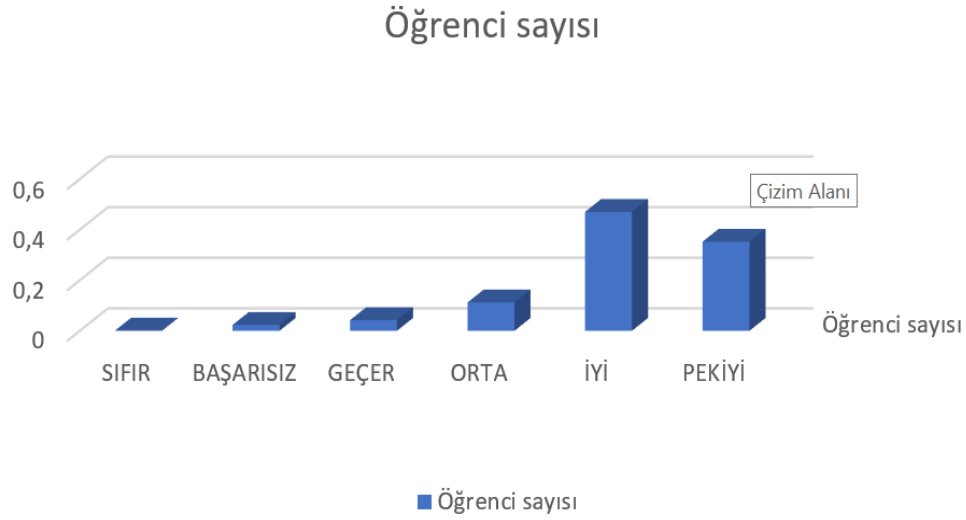
Şekil 4.2. Anketin Exceldeki Tablo Kısmı

Çalışma sırasında asıl tahminleme işlemi, anketteki son soru olan e-okul not ortalaması sütunu üzerindedir. Ankete katılan öğrencilerin not ortalama dağılımları Tablo 4-1'deki gibidir. Görüldüğü üzere veriler sağa doğru çarpıktır.

Tablo 4-1. Not Ortalamalarına Göre Öğrenci Sayıları

Not Ortalaması	Öğrenci Sayısı
PEKİYİ	352
İYİ	470
ORTA	112
GEÇER	42
BAŞARISIZ	23
SIFIR	1

Şekil 4.3. Notlara Göre Öğrenci Sayıları



Şekil 4.4. Notlara Göre Öğrenci Sayılarının Sağa Çarpıklığı

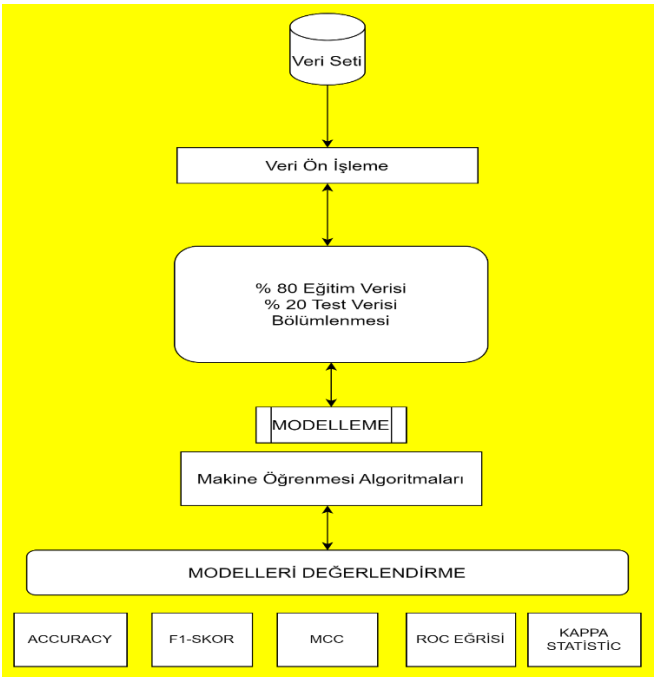
Şekil 4.5. Notlara Göre Öğrenci Sayılarının Sağa Çarpıklığı

4.2. Veri Ön İşleme

Öğrencilerden alınan anket cevapları, Excel tablosuna geçirildikten sonra üzerinde değişiklikler ve düzenlemeler yapılmıştır. Anket sorularından aile geliri araştırma süresince çok değişip, artış göstermiş ama sıralaması değişmemiştir. Bu tarzdaki kategorilere ayrılan çoktan seçmeli soruların cevapları sayısallaştırılarak gruplanmıştır. Aynı işlem evet-hayır cevaplı sorulara da uygulanmıştır. Açık uçlu soruların cevapları analiz edilerek belli ortak cevap kategorilerine ayrılıp, sonra bu kategorilere numara vererek sayısal haline gelmiştir. En son sorudaki (E-okuldaki) not ortalamanız sorusunun cevabı M.E.B.'da uygulandığı gibi pekiyi-iyi-orta-geçer-başarısız ve sıfır şeklinde not sınıflarına ayrılmıştır. Bu sayede Weka platformunda çalışılabilecek hale gelmiştir.

4.3. Sistem Modellenmesi

Weka üzerinde çalışmalarımız veri setini %80'ini eğitim, %20'si test olarak icra edilmiştir. Kümeleme ve sınıflandırma çalışmaları sırsında cross-validation 10 parçalı olaraktan, tüm veri seti bütün olarak (using traning test) alınarak da kullanılmıştır. Tüm süreç aşağıdaki Şekil 4.6 ile modellenmiştir.



Şekil 4.6. Sistem Mimarisi

5. BULGULAR

5.1. Makine Öğrenmesi Deneyleri

5.1.1. Sınıflandırma Deneyleri

Tez çalışmasının birinci aşamasında, %80 eğitim ve %20 test verileri WEKA içindeki tüm makine öğrenmesi algoritmaları ile deneyler yapılmıştır. Sonraki aşamada ise aynı algoritmalara özellik seçimi tekniği uygulanarak her iki çalışmada elde edilen sonuçlar karşılaştırılmıştır. Normal sınıflandırma ile ensemble sınıflandırma algoritmaları ayrı Tablo 5.1. ve 5.2. de gösterilmiştir. Bu algoritmalar içinden en yüksek performansa sahip algoritma sınıflandırmada RandomForest, ensemble sınıflandırmada RandomSubSpace ve Random Forest birlikteliği olmuştur.

Tablo 5-1. Sınıflandırma Algoritmaları Başarım Sonuçları (%)

ALGORİTMA	**%80 Percentage Split**	**Cross-validations (10 Folds)**
Naive Bayes	0,445	0,485
Naive Bayes Multinominal	0,495	0,525
Logistic	0,44	0,515
Simple Logistic	0,465	0,503
SMO	0,48	0,526
SMO Normalize PolyKernel	0,495	0,527
SMO RBFKernel	0,5	0,525
Kstar	0,68	0,702
Jrip	0,58	0,546
PART	0,605	0,616
Decision Stump	0,485	0,51
J48	0,585	0,643
LMT	0,635	0,636
Random Forest	**0,71**	**0,726**

Deney sonuçları incelendiğinde Tablo 5.1.'deki sınıflandırma algoritmaları %44 ile %72,6 arası doğruluk değerleri üretmişlerdir. Random Forest %72,6 (cross-validation) doğruluk değeri ile en başarılı algoritma olurken, Logistic %44 doğruluk değeri ile en az başarı

sahibidir. Toplamda 14 farklı algoritma aslında birbirlerine yakın sayılabilecek başarı değeri üretmiştir. Bunda daha önce bahsedildiği gibi verilerin sağa çarpık olması ve çok çeşitli alanlardan, değişik korelasyona haiz özellikleri belirtmesi büyük etken olmuştur.

Tablo 5-2. Ensemble Sınıflandırma Algoritmaları Başarım Sonuçları (%)

ENSEMBLE LEARNİNG	**%80 Percentage Split**	**Cross-validation (10 Folds)**
MetaAdaBoostM1+Decision Stump	0,484	0,51
MetaAdaBoostM1+Naive Bayes	0,445	0,485
MetaAdaBoostM1+Naive Bayes Multi.	0,495	0,528
MetaAdaBoostM1+Logistic	0,44	0,513
MetaAdaBoostM1+Simple Logistic	0,465	0,503
MetaAdaBoostM1+SMO	0,48	0,526
MetaAdaBoostM1+Kstar	0,66	0,676
MetaAdaBoostM1+JRip	0,625	0,614
MetaAdaBoostM1+PART	0,665	0,674
MetaAdaBoostM1+J48	0,645	0,697
MetaAdaBoostM1+LMT	0,63	0,672
MetaAdaBoostM1+Random Forest	0,705	**0,724**
RandomSubSpace+Rep Tree	0,62	0,615
RandomSubSpace+Naive Bayes	0,485	0,513
RandomSubSpace+Naive Bys.Mnl	0,505	0,514
RandomSubSpace+Logistic	0,505	0,529
RandomSubSpace+Simple Logistic	0,46	0,521
RandomSubSpace+SMO	0,495	0,524
RandomSubSpace+Kstar	0,715	0,721
RandomSubSpace+JRip	0,585	0,551
RandomSubSpace+ PART	0,655	0,696
RandomSubSpace+Decision Stump	0,51	0,519
RandomSubSpace+J48	0,655	0,687
RandomSubSpace+LMT	0,7	0,715
RandomSubSpace+Random Forest	**0,725**	0,72
Vote+Kstar+PART+LMT+Ran.Frst.	0,695	0,709
Vote+N.Bayes M.+Kstar+Ran.Frst.	0,715	**0,72**

Ensemble learning (topluluk öğrenmesi) tarafında (Tablo 5-2.) ise 29 farklı kombinasyon ile 3 farklı meta algoritma (Random SubSpace, MetaAdaBoostM1 ve Vote) bazında sınıflandırma sonuçları üretilmiştir. Bunlarda MetaAdaBoostM1 ve Random Forest %72,4; RandomSubSpace ve Random Forest %72,5 ve de son olarak Vote

algoritması altında Naive Bayes Multinominal, KStar ve Random Forest ile %71,5 şeklinde en başarılı algoritmalar olmuşlardır. Random Forest veri setinin sağa çarpık ve dağınık yapısına ağaç dalları ile ayırma usulü ile en iyi uyum sağlayan algoritma olarak öne çıkmaktadır.

Not ortalaması sınıfına göre sağa çarpık olan veri seti üzerinde, 352 pekiyi ortalamalı veriden 100 tane, 470 tane iyi ortalamalı veriden 100 tane, 112 orta ortalamalı veriden 100 tane veriyi rastgele alarak ve geriye kalan 42 geçer, 23 başarısız ve 1 sıfır ortalamalı veriyi de olduğu gibi alarak "Balans Veri Seti" oluşturulmuştur. Böylelikle yakın nicelikli veriler arasında daha yüksek sınıflandırma başarısı hedeflenmiştir. Balans veri tabanı ile çalışmalar her algoritma için %80 eğitim, %20 test verisi olarak ve 10 folds cross-validation olarak gerçekleştirilmiştir.

Tablo 5-3. Balans Veri Tabanı ile Sınıflandırma Algoritmaları Başarım Sonuçları (%)

ALGORİTMA	**%80 Percentage Split**	**Cross-validation (10 Folds)**
Naive Bayes	39,73	43,45
Naive Bayes Multinominal	31,5	36,88
Logistic	39,73	38,52
Simple Logistic	36,98	37,7
SMO	39,73	40,44
Kstar	42,47	44,54
JRip	26,02	32,51
PART	42,47	47
Decision Stump	26,02	33,06
J48	46,58	40,43
LMT	46,58	43,99
Random Forest	49,32	45,63
Random Tree	49,32	49,3
ENSEMBLE LEARNING		
MetaAdaBoostM1+Decision Stump	26,03	33,06
MetaAdaBoostM1+Naive Bayes	39,73	43,44
MetaAdaBoostM1+Naive Bayes Multi.	31,5	36,89
MetaAdaBoostM1+Logistic	39,73	38,52
MetaAdaBoostM1+Simple Logistic	36,99	37,7
MetaAdaBoostM1+SMO	39,73	40,43
MetaAdaBoostM1+Kstar	43,84	40,98

MetaAdaBoostM1+JRip	26,02	33,61
MetaAdaBoostM1+PART	46,58	50,55
MetaAdaBoostM1+J48	50,68	48,63
MetaAdaBoostM1+LMT	49,32	45,35
MetaAdaBoostM1+Random Forest	49,32	46,72
RandomSubSpace+Naive Bayes	43,84	42,35
RandomSubSpace+Naive Bayes Multi.	34,25	37,16
RandomSubSpace+Logistic	41,1	41,1
RandomSubSpace+Simple Logistic	34,25	39,89
RandomSubSpace+SMO	32,88	39,89
RandomSubSpace+Kstar	49,32	47,54
RandomSubSpace+JRip	28,77	42,62
RandomSubSpace+PART	41,09	46,17
RandomSubSpace+Decision Stump	26,02	34,7
RandomSubSpace+J48	45,2	50
RandomSubSpace+LMT	36,99	47
RandomSubSpace+Random Forest	49,32	47,27
RandomSubSpace+Rep Tree	36,99	43,17
Vote+Kstar+PART+LMT+Random Forest	**58,68**	49,73
Vote+N.Bayes M.+Kstar+Random Forest	43,83	45,9

Bu deneyler yukarıda Tablo 5-3'te görüldüğü üzere en fazla %58,68'lik (Vote+Kstar+PART+LMT+Random Forest algoritmaları) sınıflandırma başarısı sağladığı için istenileni vermemiştir. Bu veri seti üzerindeki satır azaltma işlemi sonrasında, veri tabanında sütun azaltma (öz nitelik seçimi-feature selection) işlemleri yapılarak başarı artışı hedeflenmiştir. Amaç en iyi performans sonucu için, en optimum kolonlarla sınıflandırma yapmaktır. Veri seti üzerinde yaptığımız analiz sonucunda, tüm değerleri birbirlerine yakın olan ve not ortalamasına etkisi düşük kabul edilebilecek bazı öznitelikleri (kolonları) Weka yardımı ile silerek sınıflandırma işlemleri gerçekleştirilmiştir. Önce WEKA üzerinden 17, 20, 22, 24, 26, 27, 31, 32, 34, 36, 39, 41, 43 numaralı sütunlar (öz nitelikler) silinerek deneyler yapılmıştır.

Tablo 5-4. İlk Öznitelik Seçimi Sınıflandırma Algoritmaları Başarım Sonuçları (%)

ALGORİTMA	**%80 Percentage Split**	**Cross-validation (10 Folds)**
Naive Bayes	48,5	48,9
Naive Bayes Multinominal	50	54,4
Logistic	49	50,6

Simple Logistic	47,5	51,5
SMO	50,5	54
Kstar	64	68,3
JRip	53,5	54,5
PART	64	62,9
Decision Stump	48,5	51
J48	63	62,8
LMT	64	65,3
Random Forest	68,5	**72,6**
ENSEMBLE LEARNING		
MetaAdaBoostM1+Decision Stump	48,5	51
MetaAdaBoostM1+Naive Bayes	48,5	48,9
MetaAdaBoostM1+Naive Bayes Multi.	50	54,4
MetaAdaBoostM1+Logistic	49	50,6
MetaAdaBoostM1+Simple Logistic	47,5	51,1
MetaAdaBoostM1+SMO	50,5	54
MetaAdaBoostM1+Kstar	64	67,4
MetaAdaBoostM1+JRip	58	61
MetaAdaBoostM1+PART	68	69,8
MetaAdaBoostM1+J48	66	69,7
MetaAdaBoostM1+LMT	61,5	69,3
MetaAdaBoostM1+Random Forest	70,5	**72,5**
RandomSubSpace+Naive Bayes	51	51,7
RandomSubSpace+Naive Bayes Multi.	50,5	52,8
RandomSubSpace+Logistic	50	53,5
RandomSubSpace+Simple Logistic	50,5	53,6
RandomSubSpace+SMO	48,5	52,2
RandomSubSpace+Kstar	66	71,4
RandomSubSpace+JRip	59	57,9
RandomSubSpace+PART	66,5	67,4
RandomSubSpace+Decision Stump	48,5	52,3
RandomSubSpace+J48	66,5	66,5
RandomSubSpace+LMT	70	68,3
RandomSubSpace+Random Forest	68	71,1
RandomSubSpace+Rep Tree	58,5	62,6
Vote+Kstar+PART+LMT+Random Forest	68,5	70,5
Vote+N.Bayes M.+Kstar+Random Forest	68,7	71,4

Deneyler sonunda Tablo 5.4.'de görüldüğü gibi en başarılı algoritmalar (10 folds cross-validation) %72,6 ile Random Forest ve %72,5 ile MetaAdaBoostM1+Random Forest birlikteliği olmuştur. Tez çalışmasının bulguları bu aşamadan sonra, daha fazla öz

nitelik çıkarması yapılarak WEKA üzerinden 7, 12, 13, 16, 17, 20, 22, 24, 26, 27, 31, 32, 34, 36, 39, 41, 43 numaralı sütunlar (öz nitelikler) silinerek yapılan öz nitelik seçimi sonuçları aşağıda yer almaktadır.

Tablo 5-5. İkinci Öznitelik Seçimi Sınıflandırma Algoritmaları Başarım Sonuçları (%)

ALGORİTMA	**% 80 Percentage Split**	**Cross-validation (10 Folds)**
Naive Bayes	46	47,5
Naive Bayes Multinominal	47,5	50,6
Logistic	50,5	49,5
Simple Logistic	50	49,5
SMO	48	50,4
Kstar	62	68,8
JRip	46	55,1
PART	60	61
Decision Stump	47,5	51
J48	59	58,2
LMT	60	63,2
Random Forest	66	**69,7**
ENSEMBLE LEARNİNG		
MetaAdaBoostM1+Decision Stump	47,5	51
MetaAdaBoostM1+Naive Bayes	46	47,5
MetaAdaBoostM1+Naive Bayes Multi.	47,5	50,6
MetaAdaBoostM1+Logistic	52	49,4
MetaAdaBoostM1+Simple Logistic	50	49,2
MetaAdaBoostM1+SMO	48	51,3
MetaAdaBoostM1+Kstar	61,5	67,6
MetaAdaBoostM1+JRip	55	56,5
MetaAdaBoostM1+PART	63,5	68,2
MetaAdaBoostM1+J48	66	66,4
MetaAdaBoostM1+LMT	63,5	66,6
MetaAdaBoostM1+Random Forest	68	69,9
RandomSubSpace+Naive Bayes	47,5	50,1
RandomSubSpace+Naive Bayes Multi.	51	50,4
RandomSubSpace+Logistic	48,5	51,6
RandomSubSpace+Simple Logistic	47,5	51,1
RandomSubSpace+SMO	47,5	51,1
RandomSubSpace+Kstar	64	70,9
RandomSubSpace+JRip	50,5	56,7
RandomSubSpace+PART	67	67,9
RandomSubSpace+Decision Stump	47	50,4
RandomSubSpace+J48	68,5	67,4

RandomSubSpace+LMT	68,5	69,2
RandomSubSpace+Random Forest	69	70,7
RandomSubSpace+Rep Tree	62,5	69,3
Vote+Kstar+PART+LMT+Random Forest	68	68,7
Vote+N.Bayes M.+Kstar+Random Forest	66	**70,9**
Vote+RandomSubSpace(RandomForest)	69	70,7

Tablo 5-5'teki bulgular incelendiğinde, en başarılı algoritmalar (10 folds cross-validation) %69,7 ile Random Forest ve %70,9 ile RandomSubSpace + Kstar ve Vote + RandomSubSpace (Random Forest) birliktelikleri olduğu görülmüştür. Buraya kadarki yapılan deneylerin sonuçları analiz edildiğinde; ilk veri setimiz üzerinde %72,5 başarı oranı ile RandomSupSpace meta algoritması altında Random Forest algoritmalarının topluluk öğrenmesi ve %72,6'lık başarı oranı ile ilk öznitelik seçimi yapılan veri seti ile Random Forest algoritması olmuştur. Weka'da uygulanmaları sırasında bu algoritmalar ile ilgili daha fazla özellikler üzerine çalışılarak, ağaç sayılarının değişimini sınıflandırma başarısına etkisi incelenmiştir [44]. RandonSupSpace ve Random Forest birlikteliği ile ilgili olarak;

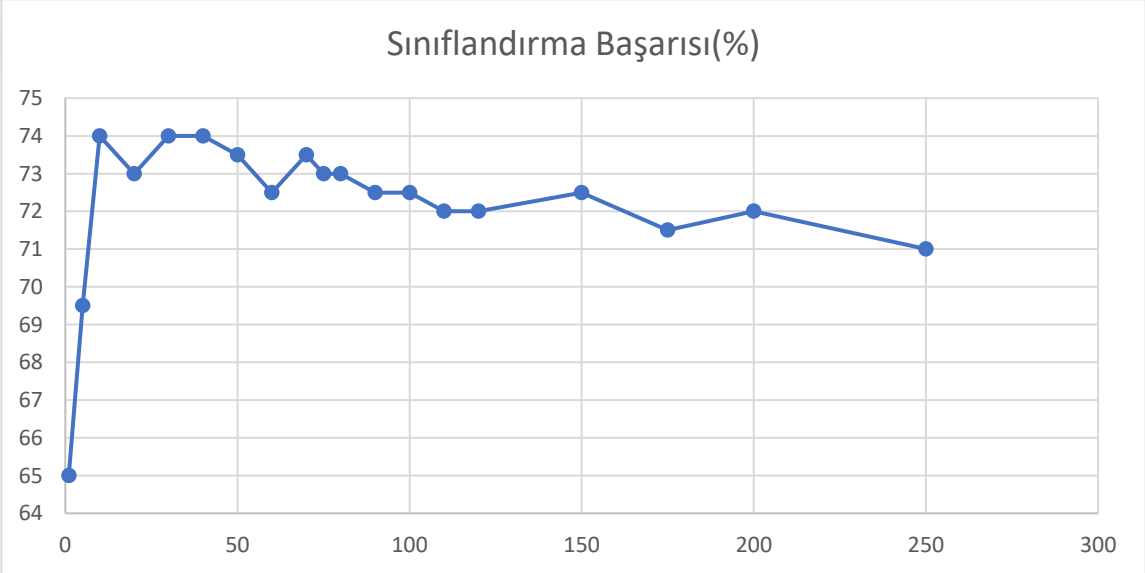
Tablo 5-6. RandomSupSpace ve Random Forest Algoritmaları Başarım Sonuçları (%)

	RandomSupSpace	**Random Forest**	**Sınıflandırma Başarısı(%)**
NumIterations	1	1	50,5
	1	10	61,5
	1	20	64
	1	30	64,5
	1	40	63,5
	1	50	63
	1	100	63,5
	5	1	58
	5	10	72
	5	20	70,5
	5	30	73,5
	5	40	73,5
	5	50	73,5
	5	75	73
	5	100	72,5
	5	150	72
	5	250	72

	10	1	65
	10	5	69,5
	10	10	**74**
	10	20	73
	10	30	**74**
	10	40	**74**
	10	50	73,5
	10	60	72,5
	10	70	73,5
	10	75	73
	10	80	73
	10	90	72,5
	10	100	72,5
	10	110	72
	10	120	72
	10	150	72,5
	10	175	71,5
	10	200	72
	10	250	71
NumIterations	30	1	71
	30	5	72
	30	10	72
	30	20	73
	30	30	71,5
	30	40	71,5
	30	50	72
	30	75	71,5
	30	100	71,5
	40	1	69
	40	5	73,5
	40	10	73
	40	20	72
	40	30	71,5
	40	40	71,5
	40	50	71,5
	40	75	71,5
	40	100	72
	50	1	70,5
	50	5	71,5
	50	10	73
	50	20	72
	50	30	71
	50	40	71

	50	50	71
	50	75	71
	50	100	73

Bu kısımda en başarılı sonuç %74 ile %80 eğitim, %20 test veri uygulaması olarak RandomSupSpace algoritmasının NumIterations 10 olup, Random Forest algoritmasının NumIterasyonun 10, 30 veya 40 olduğu versiyonlardır. Özel olarak bu versiyonun sınıflandırma başarı grafiği Şekil 5.1'deki gibidir.



Şekil 5.1. RandomSupSpace ve Random Forest Algoritmaları Başarım Sonuçları (%)

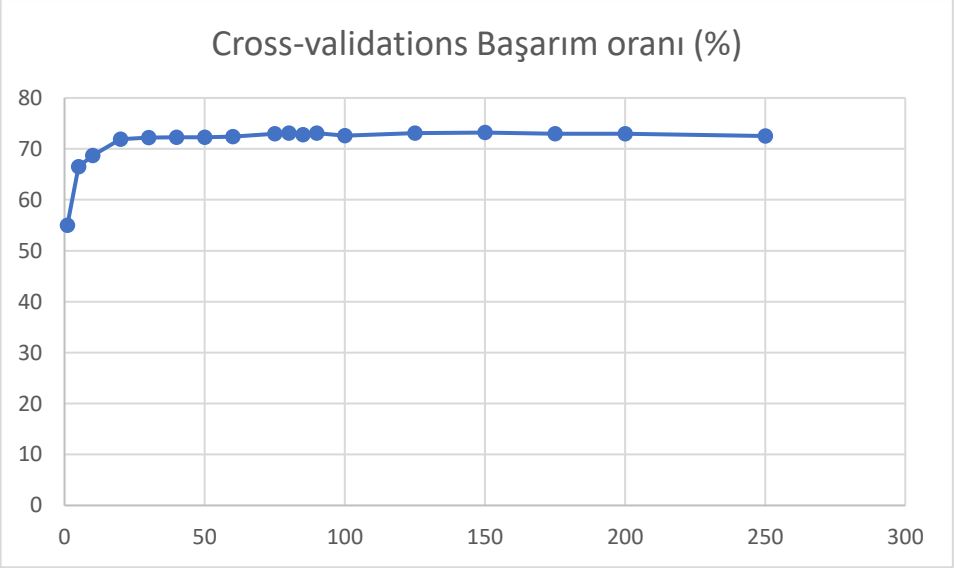
İlk öznitelik seçimi (17, 20, 22, 24, 26, 27, 31, 32, 34, 36, 39, 41 ve 43 numaralı öznitelikler Weka ile silinerek) yapıldıktan sonra, 10 folds cross-validation metoduyla Random Forest algoritmasının ağaç sayılarını değiştirerek aşağıdaki sonuçlar üretilmiştir.

Tablo 5-7. Random Forest Algoritması Başarım Sonuçları (%)

NumIterations	**Cross-validations Başarım oranı (%)**
1	55
5	66,5
10	68,7
20	71,9
30	72,2
40	72,3

50	72,3
60	72,4
75	73
80	73,1
85	72,8
90	73,1
100	72,6
125	73,1
150	**73,2**
175	73
200	73
250	72,5

Bu kısımda en başarılı sonuç %73,2 ile test options cross-validation olarak Random Forest algoritmasında numIterations'ın 150 olduğu versiyon olmuştur. Özel olarak bu versiyonun ağaç sayılarına göre sınıflandırma başarı grafiği aşağıdaki gibidir.



Şekil 5.2. Random Forest Algoritması Başarım Sonuçları (%)

6. SONUÇ VE TARTIŞMA

Bu çalışmada öğrencilerden anket yoluyla elde edilen veriler üzerinden makine öğrenmesi yöntemleri kullanılarak başarı ortalaması tahmin edilmeye çalışılmıştır. Verileri elde etmek için gerçekleştirilen anket çalışmasında örneklem uzayın yani öğrenci kitlesinin anket sorularına yeterli düzeyde konsantrasyon ile cevap veremedikleri görülmüştür. Öğrenci kitlesi cevaplarını örnekler üzerinden vermeye çalışarak, kendilerinden farklı bir cevap katamamışlardır. Bu durum aslında başarısızlık sebebi olarak değerlendirilebilir. Ek olarak, verilen cevaplara göre öğrencilerin aldığı notların pekiyi, iyi, orta, geçer, başarısız ve sıfır not sınıflarını aynı oranda taşımadığı görülmektedir. Yapılan değerlendirmelerde fazladan notlar (muhtemelen sözlü notları ile) verilmiş olduğu veri setinden anlaşılmaktadır. Bu bağlamda makine öğrenmesi vb. yapay zekâ veya otomasyon yapılarının (insan duygusallığı zaaflarını) orta öğretim başarı değerlendirme sistemine dahil edilmesinin gerekliliği ön plana çıkmıştır.

Tez çalışmasının ilk aşamasında veri setinin %80'ini eğitim, %20'si ise test verisi olarak ayrılarak 10 katmanlı çapraz geçerleme (10 folds cross-validation) yöntemi dikkate alınarak WEKA platformu üzerinde tüm makine öğrenmesi algoritmalarının kullanımı ile sınıflandırma deneyleri gerçekleştirilmiştir. Aynı algoritmalar topluluk öğrenmesi (ensemble learning) yöntemi ile meta algoritmalarla kombinasyonlar haline getirilmiştir. Bu şekilde %80 eğitim, %20 test verisi ve 10 katmanlı çapraz geçerleme yöntemi ile Weka'da sınıflandırma deneylerine tabii tutulmuştur. İkinci olarak veri seti üzerinde tahminlenecek sınıfların nicelik olarak eşitlenmesi hedeflenmiştir. Bu şekilde oluşacak yeni veri seti (balans veri seti) ile sınıflandırma deneyleri yapılmıştır. Sonraki aşamada ise aynı algoritmaların kullanımı ile öznitelik seçimi uygulanması düşünülmüştür. Bu bağlamda anket soruları ve cevaplar tek tek incelenerek öğrenci başarısına etkisi az olan bazı kolonlar veri setinden Weka yardımıyla atılarak, topluluk öğrenmesi yöntemi ile sınıflandırma deneyleri yeniden gerçekleştirilmiştir. Son olarak Weka platformunun "Select Atributes" butonu kullanılarak, filtreler bölümünden "Temel Bileşenler Analizi (Principal Component Analysis)" işlemi ile sınıflandırma işlemleri gerçekleştirilmiştir.

İlk aşamada makine öğrenmesi algoritmalarının kullanımı ile gerçekleştirilen sınıflandırma deneyleri sonunda %44 ile %72,6 arasında doğruluk (accuracy) değerleri elde edilmiştir. Random Forest algoritması %72,6'lık (cross-validation) doğruluk değeri ile öğrenci başarılarının sınıflandırılmasında en başarılı algoritma olurken, Logistic %44'lük doğruluk değeri ile en başarısız algoritma olmuştur. Toplamda 14 farklı algoritma aslında birbirlerine yakın sayılabilecek başarı değerleri üretmiştir. Topluluk öğrenmesi yöntemiyle gerçekleştirilen deneysel çalışmalarda ise 29 farklı kombinasyon ile 3 farklı meta algoritması (Random SubSpace, MetaAdaBoostM1 ve Vote) üzerinden sınıflandırma sonuçları elde edilmiştir. Bu algoritmalar içerisinde MetaAdaBoostM1 ve Random Forest %71,9 doğruluk değeri, RandomSubSpace ve Random Forest ile %72,5 doğruluk değeri ve son olarak da Vote algoritması altında Naive Bayes Multinominal, KStar ve Random Forest %71,5'lik doğruluk değeri ile en başarılı algoritmalar olmuşlardır. Random Forest algoritması veri setinin sağa çarpık ve dağınık yapısına rağmen ağaç dalları ile ayırma özelliğine göre en iyi uyum sağlayan algoritma olarak öne çıkmıştır.

Not ortalaması sınıfına göre yukarıda anlatıldığı gibi veri seti üzerinde, 352 pekiyi veriden 100 adet, 470 iyi veriden 100 adet, 112 orta veriden 100 adet veriyi rastgele alarak ve geriye kalan 42 geçer, 23 başarısız ve 1 sıfır ortalamalı veriyi de alarak "Balans Veri Seti" oluşturulmuştur. Böylelikle yakın nicelikli veriler arasında daha yüksek sınıflandırma başarısı hedeflenmiştir. Balans veri seti ile gerçekleştirilen deneysel çalışmalar her algoritma için daha önce yukarıda belirtilen ayrım göz önünde bulundurularak ele alınmıştır. Gerçekleştirilen bu deneysel çalışmalarda en fazla %58,68'lik (Vote+Kstar+PART+LMT+Random Forest) sınıflandırma başarısı elde edildiği için istenilen sonuca ulaşılamamıştır. Bu veri seti üzerindeki satır azaltma işlemi (balans veri seti hazırlanması) sonrasında, veri setinde sütun azaltma (öz nitelik seçimi-feature selection) işlemleri yapılarak başarı artışı düşünülmüştür. Amaç en yüksek sınıflandırma başarısını elde etmek olduğundan, öğrenci başarısına en fazla etki eden kolonlarla sınıflandırma işlemi gerçekleştirilmeye çalışılmıştır. Veri seti üzerinde yapılan inceleme sonrası, cevaplardan değerleri birbirlerine yakın olanlar çıkarılmış ve not başarısına etkisi olmayan bazı öznitelikler (kolonlar) Weka yardımı ile silinerek

sınıflandırma işlemlerine yönelik deneysel çalışmalar yeniden ele alınmıştır. En yüksek sınıflandırma başarısı %72,6'lık doğruluk değeri ile Random Forest algoritması ve %72,5'lik doğruluk değeri ile MetaAdaBoostM1+Random Forest topluluk öğrenmesi yöntemi ile elde edilmiştir.

Weka üzerinden yapılan birbirleriyle en yakın ilişkide bulunan özniteliklerin seçimi, sınıflandırmada %48 ile %51 arasında başarı göstermiştir. Gerçekleştirilen deneysel çalışmalarda en başarılı algoritmalar ve algoritma birliktelikleri Random Forest algoritması ile RandomSupSpace meta algoritması altında Random Forest algoritması olmuştur. Weka'da üzerinde gerçekleştirilen deneysel çalışmalar sırasında bu algoritmalar ile ilgili daha fazla özellikler üzerine çalışılarak, ağaç sayılarının ve iterasyon sayılarının değişiminin sınıflandırma başarısına etkisi incelenmiştir. Öznitelik seçimi yapılan veri seti ile 10 katmanlı çapraz geçerleme yöntemi ile Random Forest algoritmasının ağaç sayılarını değiştirerek %73,2'lik doğruluk değeri ile test options cross-validation olarak Random Forest algoritmasında numIterations'ın 150 olduğu versiyon olmuştur. RandomSupSpace ve Random Forest birlikteliğinde ise, en başarılı sınıflandırma oranı %74 olurken veri seti %80 eğitim, %20 test olarak ayrılmış ve bu halde RandomSupSpace algoritmasının NumIteration'u 10 iken, Random Forest algoritmasının NumIterasyonun 10, 30 veya 40 olduğu versiyon olarak gerçekleştirilmiştir.

Öğrencilerden toplanan cevaplar üzerinden oluşturulan veri seti oldukça yoğun ve çok kategorili bilgiler içermektedir. Yapılan deneysel çalışmalarda başarım düzeyinin en fazla %74'lerde kalması özniteliklerin başarı üzerinde eşit etkide olmadıklarını göstermektedir. Bu bağlamda her özelliğe bir öznitelik olarak farklı ağırlık verilmeli ve bu ağırlıklar sistem tarafında çoklu denemelerle ayrı ayrı bulunmalıdır. Anlaşılabileceği üzere bu durum yapay sinir ağları ve bu ağların çok katmanlısı derin öğrenme alt disiplininde çalışmalar yapılabileceği sonucuna götürmektedir. Yine bu tarz veri seti doğru ağırlıklandırma ve öznitelik seçimleri ile ülkemizde çok eksikliği hissedilen başka bir alana da çözüm getirebilecektir. Bu öğrencilerin lise sonrasında yeteneklerine uygun meslek alanlarına yöneltilmesi, böylece uygun üniversite bölümlerini tercih etmelerinin sağlanmasına yol açacaktır.

KAYNAKLAR

[1] Ertürk, S. (1988). Türkiye'de eğitim felsefesi sorunu. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, *3*(3).

[2] Ural, A., Çınar, F. N. (2016). Anne ve Babanın Eğitim Düzeyinin Öğrencinin Matematik Başarısına Etkisi. Mehmet Akif Ersoy Üniversitesi Eğitim Bilimleri Enstitüsü Dergisi, 3(4), 42-57.

[3] Aslan Argun, E, Özakça, B. (2015). Akademik başarıları yüksek olan öğrencilerin başarı düzeylerine ailelerinin katkıda bulunma biçimleri. Anadolu Üniversitesi Sosyal Bilimler Dergisi, 15 (3), 9-21.

[4] Olcay, A., Döş, İ. (2009). An application to determine the factors affecting the success negatively in respect of students. Gaziantep University Journal of Social Sciences, 8(1), 131-155.

[5] Abbasoğlu, B. (2020). Ortaokul Öğrencilerinin Akademik Başarılarının Eğitsel Veri Madenciliği Yöntemleri ile Tahmini. Veri Bilimi, 3(1), 1-10.

[6] Çiftçi, F., Kaleli, C., & Ünal, S. (2018). Öznitelik Seçme ve Makine Öğrenmesi Yöntemleriyle Eğitmen Performansının Tahmin Edilmesi. Anadolu Journal of Educational Sciences International, 8(2), 419-440. https://doi.org/10.18039/ajesi.454587

[7] Webb, M.E., Fluck, A., Magenheim, J., Malyn-Smith, J., Waters, J., Deschênes, M., Zagami, J. (2021) Machine learning for human learners: opportunities, issues, tensions and threats. Education Tech Research Dev, 69, 2109–2130.

[8] Solmaz, M., Alaybeyoğlu, A. (2018). Bulanık Mantık ile Matlab Projesi Öğrencinin Eğitim ve Öğretim Başarısı. ULEAD 2018 ANNUAL CONGRESS 8TH International Congress of Research in Education 09-10-11 May 2018 Abstract Book.

[9] Aydoğan, M., Karcı, A. (2018). Meslek Yüksekokulu Öğrencilerinin Başarı Performanslarının Makine Öğrenmesi Yöntemleri ile Analizi, 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies, 19-21 October 2018.

[10] Durak, A. ve Bulut V. (2023). Programlama eğitiminde öğrenci performansının makine öğrenmesi algoritmaları ile tahminlenmesi. 5. Uluslararası Uygulamalı Mühendislik ve Doğa Bilimleri Konferansı, Konya.

[11] Pek, R. Z., Özyer, S. T., Elhage, T., Özyer, T., & Alhajj, R. (2022). The role of machine learning in identifying students at-risk and minimizing failure. *IEEE Access*, *11*, 1224-1243.

[12] Pinto, A. S., Abreu, A., Costa, E., & Paiva, J. (2023). How Machine Learning (ML) is Transforming Higher Education: A Systematic Literature Review. *Journal of Information Systems Engineering and Management*, *8*(2).

[13] Langley, P. ve Simon, HA (1995). Makine öğrenimi ve kural çıkarma uygulamaları. *ACM İletişimleri, 38* (11), 54-64.

[14] Tatlıoğlu, K ve Korkmaz, G. (2015). İlköğretim Öğrencilerinin Okul Başarılarını Olumsuz Etkileyen Nedenlerin Belirlenmesine Yönelik Bir Araştırma (Konya Örneği). Karatekin Edebiyat Fakültesi Dergisi, 6(6), 93-116.

[15] Langley, P. ve Simon, HA (1995). Makine öğrenimi ve kural çıkarma uygulamaları. *ACM İletişimleri, 38* (11), 54-64.

[16] Singh, G., Kumar, B., Gaur, L., & Tyagi, A. (2019, April). Comparison between multinomial and Bernoulli naïve Bayes for text classification. In 2019 International Conference on Automation, Computational and Technology Management (ICACTM) (pp. 593-596). IEEE.

[17] Aborisade, O., Anwar, M. (2018, July). Classification for authorship of tweets by comparing logistic regression and naive bayes classifiers. In 2018 IEEE International Conference on Information Reuse and Integration (IRI) (pp. 269-276). IEEE.

[18] Trappey, C. V., Wu, H. Y. (2008). An evaluation of the time-varying extended logistic, simple logistic, and Gompertz models for forecasting short product lifecycles. *Advanced Engineering Informatics*, *22*(4), 421-430.

[19] Asif, A., Majid, M., & Anwar, S. M. (2019). Human stress classification using EEG signals in response to music tracks. Computers in biology and medicine, 107, 182-196.

[20] Madhusudana, C. K., Kumar, H., Narendranath, S. (2016). Condition monitoring of face milling tool using K-star algorithm and histogram features of vibration signal. *Engineering science and technology, an international journal*, *19*(3), 1543-1551.

[21] Cleary, J. G., Trigg, L. E. (1995). KStar: An instance-based learner using an entropic distance measure. In *Machine Learning Proceedings 1995* (pp. 108-114). Morgan Kaufmann.

[22] Ali, A. T., Abdullah, H. S., & Fadhil, M. N. (2021). Voice recognition system using machine learning techniques. Materials Today: Proceedings, 1-7.

[23] Sevimli Deniz S., Kural Tabanlı Sınıflandırma Algoritmalarının Karşılaştırılması Veri Bilim Dergisi, 4(3), 72-80, 2021.

[24] Wang, Y., Zhang, H., Chen, H., Boning, D. ve Hsieh, CJ (2020). Topluluk karar kütükleri ve ağaçların lp-norm sağlamlığı üzerine. *Uluslararası Makine Öğrenimi Konferansında* (s. 10104-10114). PMLR.

[25] Ali, A. T., Abdullah, H. S., & Fadhil, M. N. (2021). Voice recognition system using machine learning techniques. Materials Today: Proceedings, 1-7.

[26] Shiri, F. M., Perumal, T., Mustapha, N., Mohamed, R. (2023). A Comprehensive Overview and Comparative Analysis on Deep Learning Models: CNN, RNN, LSTM, GRU. arXiv preprint arXiv:2305.17473.

[27] Onan, A., Şirket İflaslarının Tahmin Edilmesinde Karar Ağacı Algoritmalarının Karşılaştırmalı Başarım Analizi, Bilişim Teknolojileri Dergisi Cilt:8 Sayı:1 Ocak 2015.

[28] Cahya, R. A., Bachtiar, F. A., & Mahmudy, W. F. (2021). Comparison of Bagging Ensemble Combination Rules for Imbalanced Text Sentiment Analysis. Journal of Information Technology and Computer Science, 6(1), 33-49.

[29] Bulut, F., Sınıflandırıcı Topluluklarının Dengesiz Veri Kümeleri Üzerindeki Performans Analizleri. Bilişim Teknolojileri Dergisi, Cilt :9, sayı:2 Mayıs 2016

[30] Tian, Y., Feng, Y. (2021). RaSE: Random subspace ensemble classification. *Journal of Machine Learning Research*, *22*(45), 1-93.

[31] Şeker, S.E., OptiCRM: Bir CRM, BPM, ERP Uygulaması olarak Odoo ve Yapay Zeka, YBS Anksiklopedi, Cilt :10 Sayı:1 Haziran 2022.

[32] Reddy, C. K., Vinzamuri, B. (2018). A survey of partitional and hierarchical clustering algorithms. In *Data clustering* (pp. 87-110). Chapman and Hall/CRC.

[33] Dhanaraj, R. K., Ramakrishnan, V., Poongodi, M., Krishnasamy, L., Hamdi, M., Kotecha, K., Vijayakumar, V. (2021). Random forest bagging and x-means clustered antipattern detection from sql query log for accessing secure mobile data. *Wireless Communications and Mobile Computing*, *2021*, 1-9.

[34] Ralhan, A., Medium.com, Kendi Kendini Düzenleyen Haritalar, 18 Mayıs 2020.

[35] Sadiq, A. (2021). Intrusion Detection Using the WEKA Machine Learning Tool.

[36] Hossin, M., & Sulaiman, M. N. (2015). A review on evaluation metrics for data classification evaluations. International journal of data mining & knowledge management process, 5(2):1.

[37] Bratko, I. (1997). Machine learning: Between accuracy and interpretability (pp. 163-177). Springer Vienna.

[38] Rakotomamonjy, A. (2004, August). Optimizing Area Under Roc Curve with SVMs. In ROCAI (pp. 71-80).

[39] Özhan, E. (2020). Makine öğrenmesi yöntemleri ile web'den bilgi çıkarımı sürecinin iyileştirilmesi. Afyon Kocatepe Üniversitesi Uluslararası Mühendislik Teknolojileri ve Uygulamalı Bilimler Dergisi, 3(2), 52-59.

[40] Özçift, A., Yücalar F., Borandağ, E., ve diğerleri KNN Algoritmasıve R Dili ile Metin Madenciliği Kullanarak Bilimsel Makale Tasnifi, Marmara Fen Bilimleri Dergisi 2016, (3:89-94)

[41] Atalay, M., Çelik, E., Büyük Veri Analizlerinde Yapay Zeka ve Makine Öğrenmesi Uygulamaları, M. Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi Cilt:9 Sayı:22 Aralık 2017 (s. 155-172)

[42] Pande, S. M. (2023). Machine Learning Models for Student Performance Prediction, International Conference on Innovative Data Communication Technologies, and Application (ICIDCA), Uttarakhand, India, pp. 27-32.

[43] Kavitha, R. K., Rajan Krupa, C., Isabella Menezes, J. (2023). Determining the Factors Influencing the Academic Accomplishment of Students and Predicting their Success Using Machine Learning Techniques, 2nd International Conference on

Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA), Coimbatore, India, 2023, pp. 1-6.

[44] Özçift, A. (2011). Random Forests ensemble classifier trained with data resampling strategy to improve cardiac arrhythmia diagnosis, Computers in Biology and Medicine Volume 41, Issue 5, May 2011, Pages 265-271