T.C.

BİTLİS EREN ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANA BİLİM DALI

MAKİNE ÖĞRENMESİ ALGORİTMALARI KULLANILARAK ÜNİVERSİTE BÖLÜM TERCİHİ TAHMİNİ

YÜKSEK LİSANS TEZİ

HARUN ÇELİK

DANIŞMAN

DOÇ. DR. RIDVAN UMAZ

İKİNCİ DANIŞMAN

DR. ÖĞR. ÜYESİ AYŞE SALİHA SUNAR

TEMMUZ 2023

BİTLİS

T.C.

BİTLİS EREN ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANA BİLİM DALI

MAKİNE ÖĞRENMESİ ALGORİTMALARI KULLANILARAK ÜNİVERSİTE BÖLÜM TERCİHİ TAHMİNİ

YÜKSEK LİSANS TEZİ

HARUN ÇELİK

ORCID: 0000-0002-4979-6922

DANIŞMAN

DOÇ. DR. RIDVAN UMAZ

İKİNCİ DANIŞMAN

DR. ÖĞR. ÜYESİ AYŞE SALİHA SUNAR

TEMMUZ 2023

BİTLİS

T.C.

BİTLİS EREN ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANA BİLİM DALI

YÜKSEK LİSANS TEZİ

ETİK BEYANI

Lisansüstü Eğitim Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı Yüksek Lisans öğrencisiyim. Hazırlamış olduğum "Makine Öğrenmesi Algoritmaları Kullanılarak Üniversite Bölüm Tercihi Tahmini" başlıklı tezde sunduğum akademik ve etik kurallar çerçevesinde elde ettiğimi; tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu; tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi; kullanılan verilerde herhangi bir değişiklik yapmadığımı; bu tezde sunduğum çalışmanın özgün olduğunu bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim. ……./……./2023

İmza

Harun ÇELİK

T.C.

BİTLİS EREN ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

TEZ YAZIM KILAVUZU UYGUNLUK BEYANI

“Makine Öğrenmesi Algoritmaları Kullanılarak Üniversite Bölüm Tercihi Tahmini**”** başlıklı yüksek lisans tezi Bitlis Eren Üniversitesi Eğitim Enstitüsü Tez Yazım Kılavuzuna uygun olarak hazırlanmıştır. ……./……/2023

Tezi Hazırlayan	**Danışman**	**İkinci Danışman**
imza	imza	imza
Harun ÇELİK	DOÇ. DR. RIDVAN UMAZ	DR. ÖĞR. ÜYESİ AYŞE SALİHA SUNAR

Elektrik Elektronik Mühendisliği Ana Bilim Dalı Başkanı

imza

T.C.

BİTLİS EREN ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

TEZ ONAY SAYFASI

Bitlis Eren Üniversitesi Eğitim Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı öğrencisi Harun ÇELİK tarafından hazırlanan "Makine Öğrenmesi Algoritmaları Kullanılarak Üniversite Bölüm Tercihi Tahmini" adlı Yüksek Lisans Tezi ile ilgili tez savunma sınavı, …../…../2023 tarihinde yapılmış ve tezin oy birliği/ oy çokluğu ile kabul / red edilmesine karar verilmiştir.

JÜRİ:	**İMZA**
Danışman: DOÇ. DR. RIDVAN UMAZ	**…….**
Üye: Prof. Dr. Sabir RÜSTEMLİ	**…….**
Üye: Doç. Dr. Erdal BAŞARAN	**…….**

Bitlis Eren Üniversitesi Lisansüstü Eğitim Enstitüsü Yönetim Kurulu …….. tarih ve …… sayılı kararıyla jüri tarafından kabul edilmiş bu çalışmanın yüksek lisans olarak kabulü onaylanmıştır.

…../…../2023

Doç. Dr. Mehmet Bakır ŞENGÜL

Enstitü Müdürü

T.C.

Bitlis Eren Üniversitesi Lisansüstü Eğitim Enstitüsü

Elektrik Elektronik Mühendisliği Ana Bilim Dalı

MAKİNE ÖĞRENMESİ ALGORİTMALARI KULLANILARAK ÜNİVERSİTE BÖLÜM TERCİHİ TAHMİNİ

Yüksek Lisans Tezi

Harun ÇELİK

Danışman: Doç. Dr. Rıdvan UMAZ

İkinci Danışman: Dr. Öğr. Üyesi AYŞE SALİHA SUNAR

Temmuz 2023

ÖZET

Bugünün dünyasında, üniversite bölüm tercihi yapmak öğrenciler için oldukça zorlu bir süreç olabilmektedir. Özellikle de öğrenciler; yetenekleri, ilgi alanları, kişilikleri ve gelecekteki kariyer hedefleri bakımından net bir fikre sahip değillerse, bölüm tercihi için doğru seçimi yapmak daha da zor hale gelebilir. Bu nedenle, öğrenciler üniversite tercihleri hakkında daha objektif ve bilimsel bir bakış açısı kazanmak için farklı kaynaklara başvururlar. Ancak, rehberlik servisleri veya üniversite mezunu arkadaşları ve akrabaları gibi kaynaklardan gelen tavsiyeler genellikle objektif olmayabilir. Bu tavsiyelerin objektif olamamasının nedeni, verilen önerilerin kişisel tercihler, deneyimler veya hedeflerden etkilenebilmesidir. Bu bağlamda, bu çalışma, üniversite bölüm tercihlerinde öğrencilere yardımcı olmak için makine öğrenmesi algoritmalarını kullanan bir uygulama geliştirmeyi amaçlamaktadır. Uygulama, öğrencilerin lise dönemi Matematik, Fizik, Kimya, Biyoloji, Coğrafya, Türkçe, Tarih ve yabancı dil derslerinde aldığı başarı puanlarını kullanarak, hangi bölümlerde başarılı olabileceklerini tahmin etmelerine olanak tanır. Çalışmamızda elde edilen veriler, Türkiye'nin tüm bölgelerinden toplanmış ve temizlenerek veri seti haline getirilmiş, öğrenme modellerinin daha tutarlı ve genelleştirilebilir olması amaçlanmıştır. XGBoost, SVC, KNN ve Gaussian Naive Bayes öğrenme modelleri kullanılarak yapılan çalışmada, en yüksek performans KNN algoritması ile elde edilmiştir.

Bu uygulamanın, öğrencilerin üniversite bölüm tercihlerinde karşılaştıkları zorlukları azaltmaya yönelik önemli bir adım olması amaçlanmıştır. Çünkü bu uygulama, öğrencilere objektif ve bilimsel bir bakış açısı sunarak, kişisel tercihler veya çevresel faktörlerden etkilenmeden, hangi bölümlerde başarılı olabileceklerini tahmin etmelerine yardımcı olacaktır. Bu çalışmanın, gelecekteki araştırmalara ilham verebilmesi ve üniversite bölüm tercihi yapmakta zorluk çeken öğrenciler için faydalı bir kaynak olması hedeflenmektedir.

Anahtar Sözcükler: Makine Öğrenmesi, Üniversite Bölüm Tercihi, Kariyer Tercihi, Makine Öğrenmesi Modelleri, KNN Algoritması.

Republic of Türkiye

Bitlis Eren University Graduate School

Department of Electrical and Electronics Engineering

PREDICTION OF UNDERGRADUATE MAJOR SELECTION VIA MACHINE LEARNING ALGORITHMS.

Master's Thesis

Harun ÇELİK

Superviser: Assoc. Prof. Dr. Rıdvan UMAZ

Second Superviser: Asst. Prof. Dr. Ayşe Saliha SUNAR

July 2023

ABSTRACT

In modern world, making a decision on a major for high school students can be a challenging step in their lives. Especially, if students are not aware of their potential talents, interests, personalities and goals to reach in the future. Therefore, students search and look for various sources in order to obtain the best objective and scientific perspective on their major preferences. However, advices from some guidance services or recommendations from friends and close relatives who graduated from college may not be objective and helpful. This is because these recommendations are shaped with guiders' preferences than taking into consideration of the student's ability.

In this context, this study aims to develop an application that uses machine learning algorithms to assist students in their university major choices. The application allows students to predict which departments they may be successful by using their achievement scores in high school Mathematics, Physics, Chemistry, Biology, Geography, Turkish, History, and foreign language courses. The data obtained in our study were collected from all regions of Turkey, cleaned, and transformed into a dataset to ensure more consistent and generalizable learning models. In the study, different learning models are used such as XGBoost, SVC, KNN, and Gaussian Naive Bayes learning models. Among these models, the highest performance was achieved with the KNN algorithm.

This application aims to be an important step in reducing the difficulties students face in their university major choices. Because this application will provide students with an objective and scientific perspective, helping them predict which departments they may be successful without being influenced by personal preferences or environmental factors. It is aimed that this study can inspire future research and be a useful resource for students struggling with university major choices.

Keywords: Machine Learning, University Major Choice, Career Choice, Machine Learning Models, KNN Algorithm.

TEŞEKKÜR

Öncelikle bu tez çalışması sırasında her türlü bilgi, teşvik ve deneyimleri ile yardımlarını esirgemeyen danışman hocalarım Doç. Dr. Rıdvan UMAZ ve Dr. Öğr. Üyesi Ayşe Saliha Sunar' a teşekkür ederim.

Ayrıca çalışma sürecimde bana hep destek olan ve yardımlarını esirgemeyen sevgili eşim Nimet Çelik'e; dualarıyla her an yanımda olan anne ve babama şükranlarımı sunarım.

ÖNSÖZ

Öğrenciler için üniversite bölüm tercihleri yapmak oldukça zor bir süreçtir, özellikle de kişisel hedefler, ilgi alanları, yetenekleri ve gelecekteki kariyer hedefleri bakımından net bir fikir sahibi olmadıklarında. Bu nedenle, öğrenciler üniversite tercihleri hakkında daha objektif ve bilimsel bir bakış açısı kazanmak için farklı kaynaklara başvururlar. Ancak, bu kaynaklardan gelen tavsiyeler genellikle objektif değildir ve kişisel tercihler, deneyimler veya hedeflerden etkilenebilir. Bu nedenle, bu çalışma, öğrencilere üniversite bölüm tercihleri yaparken yardımcı olmak için makine öğrenmesi algoritmalarını kullanan bir uygulama geliştirmeyi amaçlamaktadır.

Uygulama, öğrencilerin lise dönemi Matematik, Fizik, Kimya, Biyoloji, Coğrafya, Türkçe, Tarih ve yabancı dil derslerinde aldığı başarı puanlarını kullanarak, hangi bölümlerde başarılı olabileceklerini tahmin etmelerine olanak tanır. Veri seti; Türkiye'nin tüm bölgelerinden veriler toplanarak ve temizlenerek hazırlanmıştır. Bu sayede öğrenme modelleri daha tutarlı ve genelleştirilebilir hale getirilmiştir. Bu çalışmada, XGBoost, SVC, KNN ve Gaussian Naive Bayes öğrenme modelleri kullanılmış ve en yüksek performans KNN algoritması ile elde edilmiştir.

Bu uygulamanın, öğrencilerin üniversite bölüm tercihlerinde karşılaştıkları zorlukları azaltmaya yönelik önemli bir adım olması hedeflenmiştir. Çünkü bu uygulama, öğrencilere objektif ve bilimsel bir bakış açısı sunarak, kişisel tercihler veya çevresel faktörlerden etkilenmeden, hangi bölümlerde başarılı olabileceklerini tahmin etmelerine yardımcı olacaktır. Bu çalışmanın, gelecekteki araştırmalara ilham vermesi ve üniversite bölüm tercihi yapmakta zorluk çeken öğrenciler için faydalı bir kaynak olması amaçlanmıştır.

İÇİNDEKİLER

Sayfa

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

KISALTMA LİSTESİ

KDD **:** Veri Tabanlarında Bilgi Keşfi (Knowledge Discovery in Databases)

KNN **:** K - en yakın komşular (K – Nearest Neighbors)

SVM **:** Destek Vektör Makinesi (Support Vector Machine)

CRİSP-ML **:** Makine Öğrenimi aşamaları (Cross-Industry Standard Process for Machine Learning)

SVC **:** Destek Vektör Sınıflandırması (Support Vector Classification)

EVM **:** Eğitsel Veri Madenciliği

TP : True Pozitif

FN : False Negatif

FP : False Pozitif

TN : True Negatif

1. GİRİŞ

Bugünün dünyasında, üniversite bölüm tercihi yapmak öğrenciler için oldukça zorlu bir süreç olabilmektedir. Özellikle de öğrenciler; yetenekleri, ilgi alanları, kişilikleri ve gelecekteki kariyer hedefleri bakımından net bir fikre sahip değillerse, bölüm tercihi için doğru seçimi yapmak daha da zor hale gelebilir. Bu süreçte, öğrenciler üniversite tercihleri hakkında daha objektif ve bilimsel bir bakış açısı kazanmak için farklı kaynaklara başvururlar. Ancak, rehberlik servisleri veya üniversite mezunu arkadaşları ve akrabaları gibi kaynaklardan gelen tavsiyeler genellikle objektif olmayabilir. Bu tavsiyelerin objektif olamamasının nedeni, verilen önerilerin kişisel tercihler, deneyimler veya hedeflerden etkilenebilmesidir.

Bu tezin amacı; üniversite bölüm tercihlerinde öğrencilere yardımcı olmak için makine öğrenmesi algoritmalarını kullanarak bir uygulama geliştirmeyi amaçlamaktadır. Bu uygulamanın, öğrencilerin üniversite bölüm tercihlerinde karşılaştıkları zorlukları azaltmaya yönelik önemli bir adım olması amaçlanmıştır. Çünkü bu uygulama, öğrencilere objektif ve bilimsel bir bakış açısı sunarak, kişisel tercihler veya çevresel faktörlerden etkilenmeden, hangi bölümlerde başarılı olabileceklerini tahmin etmelerine yardımcı olacaktır. Bu çalışmanın, gelecekteki araştırmalara ilham verebilmesi ve üniversite bölüm tercihi yapmakta zorluk çeken öğrenciler için faydalı bir kaynak olması hedeflenmektedir.

Bu amaç doğrultusunda; öğrencilerin lise dönemi Matematik, Fizik, Kimya, Biyoloji, Coğrafya, Türkçe, Tarih ve yabancı dil derslerinde aldığı başarı puanlarını kullanarak, hangi bölümlerde başarılı olabileceklerini tahmin edilmeye çalışılmıştır. Çalışmamızda elde edilen veriler, Türkiye'nin tüm bölgelerinden toplanmış ve temizlenerek veri seti haline getirilmiş bu sayede öğrenme modellerinin daha tutarlı ve genelleştirilebilir olması amaçlanmıştır. Elde edilen verilerde çok sayıda eksik değer içeren veriler çalışmadan çıkarılmıştır. Az sayıda eksik değer içeren veriler, eksik değerler doldurularak veri setinde tutulmuştur. Aynı zamanda veri setinde veri artırma işlemleri yapılarak öğrenme modellerinin performansı yükseltilmiştir. Farklı öğrenme modelleri kullanılarak en yüksek başarı elde edilen KNN algoritması çalışmamızda kullanılmıştır. Her öğrenme modelinde hiperparametre optimizasyonunu gerçekleştirmek için GridsearchCV kullanılmıştır. Aynı zamanda veri setinin tamamının eğitim setinde

kullanılması için K-fold çaprak doğrulama tekniği kullanılmıştır. Öğrenme modellerinin başarıları Accuracy, Precision, Recall ve F1score gibi farklı metriklerle kıyaslanarak en yüksek başarı elde edilen öğrenme modeli çalışmamızda kullanılmıştır. Buna ilave olarak, en yüksek başarı elde edilen öğrenme modeli kullanılarak bir masaüstü uygulaması geliştirilmiştir. Bu sayede çalışma öğrencilere kolay ulaşılabilir hale getirilmiştir.

Bu kapsamda üç bölümden oluşan çalışmamızın düzeni şu şekilde sıralanmıştır:

Birinci bölümde; teze genel bir bakış açısı kazandırmaya yönelik bilgiler verilmiş ve makine öğrenmesi ve eğitim alanında yapılan makine öğrenmesi çalışmalarına yer verilmiştir.

İkinci bölümde; makine öğrenmesi araçları, algoritmaları ve makine öğrenimi aşamalarına yer verilmiştir.

Üçüncü bölümde; çalışmamızda kullanılan veri setinin tanımlanması, kullanılan algoritmalar, elde edilen bulgulara yer verilmiştir.

Dördüncü bölümde; çalışmamızda elde edilen kazanımlar verilerek önerilerde bulunulmuştur. Ayrıca çalışmanın geliştirilmesi için gelecekte yapılabilecek çalışmalar hakkında önerilerde bulunulmuştur.

1.1 Literatür Özeti

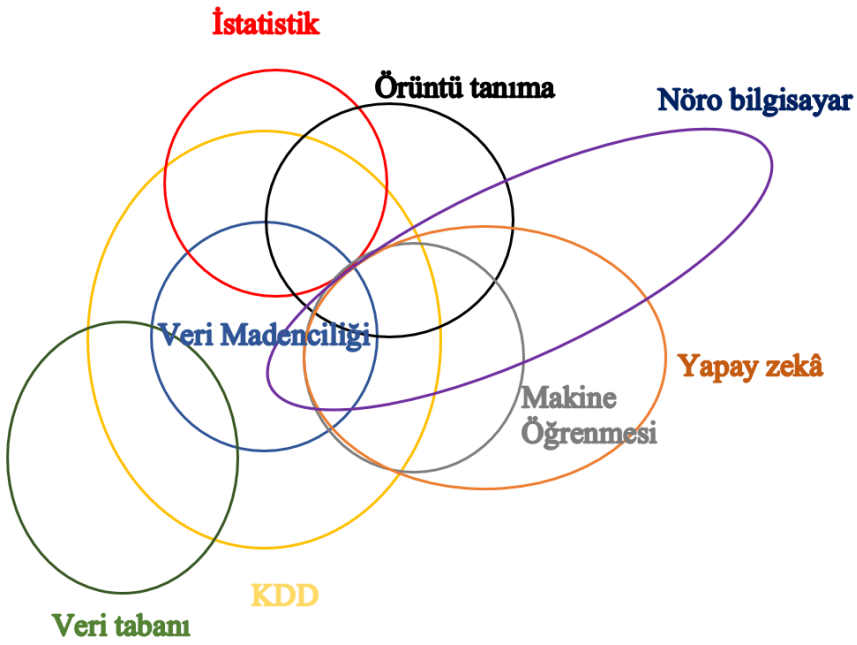
Teknolojinin gelişmesiyle insanlar işleriyle ilgili geçmişteki verileri işleyerek, geleceğe yönelik doğru tahminlerde bulunmak istediler. Ancak bu verileri bir insanın incelemesi ve bu bilgilerle bir tahminde bulunması hayli zor ve tutarsızdı. Bu durumda ortaya makinelerin insanlar gibi bilgileri inceleyip anlamlı ve faydalı çıkarımlarda bulunması ayrıca sonuçlarından beslenerek öğrenmesi ve performansını artırması fikri geldi. Buna da makine öğrenmesi adı verildi.

Makine öğrenmesi tekniklerinin kökeni istatistik ve matematiğe dayanmaktadır. 1940'lı yıllarda bilim insanları yapay bir beyin oluşturma olasılığını konuşmaya başlamış ve 1956 yılına gelindiğinde yapay zekâ araştırmaları doğmuştur [1]. 1952 yılında makine öğrenimi terimi IBM'den Arthur Samuel tarafından çıkmıştır. Samuel Ezberci Öğrenme adı verilen teknikle dama oyunundaki tüm hamleleri programa kaydederek uzman dama oyuncusu Robert Nealey'e karşı 1962 yılında rekabet etmiştir. IBM 7094 bilgisayarında oynanan oyunu program Nealey'e karşı kazanmıştır [2].

1960’lı yıllarda çok katmanlı yapıların kullanılmaya başlanması, sinir ağı çalışmalarında seviye atlatmıştır. 1970’lerde geliştirilen ileri ve geri beslemeli sinir ağları bugün derin öğrenme adıyla verileri eğitmek için kullanılmaktadır. 1980 öncesinde makine öğrenmesi, yapay zekâ için eğitim programı olarak kullanılıyorken 1980’lerde makine öğrenimi ve yapay zekâ kavramları ayrışmıştır.

1990’lı yıllara gelindiğinde makine öğrenimi sinir ağlarına odaklanmaya ve ilişkileri anlamlandırmaya başlamıştır. İnternetin gelişmesi ve verilerin sürekli artması ile sorunları çözme becerisi de artmıştır. Bilgisayarlarda grafik işlem birimlerinin hesaplamalarda kullanılmaya başlanması ve bilgisayar hızlarının artması ile yapay sinir ağları çalışmaları hız kazanmış, hesaplama hızlarında büyük artışlar görülmüştür. Son olarak 2012-2017 yılları arasında sektörün büyük teknoloji firmalarının yatırımları ile makine öğrenmesi ve derin öğrenmedeki gelişmeler hayli hız kazanmıştır.

Makine öğrenmesi, bir yapay zekâ alt dalıdır ve büyük veri kümeleri üzerinde işlem yaparak, veriler arasındaki örüntüleri ve ilişkileri tespit ederek, yeni veriler için tahmin ve sınıflandırma yapabilen modellerin geliştirilmesini sağlar. Dolayısıyla, makine öğrenmesinin gelişiminde büyük veri, oldukça önemli rol oynamaktadır. Büyük veri, yüksek hacimli, hızlı üretilen ve farklı kaynaklardan gelen verileri ifade eder. Bu veriler, sensörler, cihazlar, uygulamalar, web siteleri, sosyal medya platformları vb. gibi birçok kaynaktan gelir. Makine öğrenmesi algoritmaları, büyük veri setleri üzerinde çalışarak, veriler arasındaki ilişkileri keşfedebilir ve bu verileri kullanarak daha doğru tahminler ve sınıflandırmalar yapabilir. Büyük veri, makine öğrenmesi için önemli bir kaynak olmasının yanı sıra, makine öğrenmesinin kendisi de büyük veri analitiği için önemlidir. Makine öğrenmesi modelleri, büyük veri kümelerinin içindeki örüntüleri ve ilişkileri tespit edebilir ve bu verileri daha etkili bir şekilde analiz edebilmek için kullanılabilir. Büyük verinin daha az hata, yüksek oranda doğruluk ve yüksek hızlı bir şekilde çözümleyebilmesi için istatistik ve bilgisayar yöntemlerinin makine öğrenmesi yardımıyla bir arada kullanılması gerekmektedir Makine öğrenmesi Şekil 1’de görüldüğü gibi yapay zekânın alt kümesi ve örüntü tanıma, nöro bilgisayar, veri madenciliği gibi disiplinlerin birleşimidir. Tüm bu disiplinlerin ortak noktaları, veri tabanlarında bilgi keşfi (Knowledge Discovery in Databases-KDD) dir [3].



Şekil 1.1 KDD ve Makine öğrenmesiyle ilişkili disiplinler [4].

Makine öğrenmesi finans, sağlık hizmetleri, ulaşım, tarım ve eğitim gibi birçok alanda kullanılmaktadır. Makine öğrenmesi uygulamalarının kullanıldığı alanların içinde bulunan, eğitim alanında kullanılabilen makine öğrenmesi algoritmaları, Eğitsel Veri Madenciliği (EVM) olarak adlandırılır. EVM, eğitimcilere ve eğitim planlamacılarına büyük ve karmaşık eğitim veri kümelerini daha iyi anlamaları ve bu veri kümelerinden çıkardıkları sonuçlara göre durum tespiti yapmalarına, bunun yanında gelecek için planlamalar yapmalarına olanak sağlar [5].

Eğitim alanında ortaya çıkan büyük verinin analiz edilmesi;

- Eğitimin hedeflerini tahmin etme,
- Eğitimde alınması gereken önlemlerin erken alınması,
- Öğrenenin öğrenme yöntemini ve öğrenme biçimini belirleyerek uygun içerikleri önerme,
- Ölçme değerlendirme sonuçlarına göre eksik kaldığı konuları belirleme gibi alanlarda kullanılmaktadır.

Eğitim, tarih boyunca önemini hiç kaybetmeyen temel konulardan biri olmuştur. Toplumların ekonomik, sosyal ve kültürel konularda ilerlemeleri eğitim düzeyleri ile ilişkilidir [6]. Günümüz dünyasında öğrencilerin seçecekleri bölüm hayatlarının en kritik

dönüm noktalarından biridir. Bölüm seçiminin bilinçli yapılması hem bireyler hem de ülkeler için büyük önem taşımaktadır. Öğrencilerin bölüm seçmesi aynı zamanda yaşam biçimini de seçmesi anlamına gelmektedir. Bu nedenle bireyler tarafından doğru yapılmış bir bölüm tercihi, hayatının devamında mutlu ve başarılı olabilmesinde önemli rol oynamaktadır [7]. Mevcut eğitim yöntemlerini incelemek, analiz etmek; gelecek için yeni ve daha iyi teknikler geliştirmek için kullanılan en önemli yaklaşımlardan biri haline gelmiştir [8]. Üniversitelerde sunulan eğitimin amacına ulaşabilmesi için gerekli olan en önemli faktörlerden biri bölümlerden mezun olan öğrencilerin istihdam düzeylerinin yükseltilmesidir. Bunun sağlanabilmesi için de öğrencilerin kendi özelliklerine uygun ve ilgi duydukları alanlarla ilgili bölüm tercihleri yapmaları gerekmektedir. Çevre baskısı ve ekonomik olanaklar düşünülerek yapılmış bölüm tercihlerinde öğrencilerin başarıları düşük olmakta, bu takdirde öğrencilerin bölüme olan olumlu görüşleri değişmekte ve meslek hayatında özgüvenleri kırılmaktadır.

1.2 Eğitim alanında yapılan yapay zekâ uygulamaları

Yapay zekânın eğitim alanında kullanımına örnek birçok çalışma mevcuttur. Bu çalışmaların güçlü ve zayıf yönlerine çalışmanın bu bölümünde değineceğiz. Yapay zekânın eğitim alanında kullanılması ve bu alanda yapılan çalışmalar çizelge 2.1’ de verilmiştir. Çizelge 2.1’ de çalışma adının yanı sıra kullanılan modeller ve başarı oranları da verilmiştir.

Çizelge 2.1 Eğitim alanında yapay zeka kullanımı.

No	Adı	Tarih	Faktörler	Amaç	Öğrenme Başarısı	Kullanılan Öğrenme Modeli
1	Comparatıve Study Of Machıne Learnıng Knn, Svm, And Decısıon Tree Algorıthm To Predıct Student's Performance [9].	2019	Geçmiş akademik başarılar.	Öğrenci akademik performans tahmini.	Accuracy: %93, F1-score: %82.	KNN, Decision Tree, SVM
2	Machine Learning Based Predicting Student Academic Success [10].	2020	Geçmiş akademik başarılar.	Öğrenci ders başarısı tahmini.	Accuracy: %88, Precision: %97	SVC Elastic Net
3	Predicting Students' Performance Using Machine Learning Techniques [11].	2019	İnternet kullanımı, Sosyal ağlarda geçirilen süre, vb.	Öğrenci başarılarının tahmini.	Accuracy: %77.04, F1-score: %78.47	Yapay Sinir Ağı, Naive Bayes, Karar Ağacı, Lojistik Regresyon
4	Predıctıng Academıc Performance Of Engıneerıng Students Usıng Ensemble Method [12].	2020	Geçmiş eğitim kayıtları, demografik faktörler, aile arka planı vb.	Mühendislik öğrencilerinin başarılarının tahmini.	Accuracy: %82, F1-score: %82	DT, SVM, LR, Voting
5	Modeling and Predicting Students' Academic Performance Using Data Mining Techniques [13].	2016	Öğrenci lisans ders notları	Öğrenci akademik performans tahmini.	Accuracy: %86, Specificity %86.3	Naif Bayes, Yapay Sinir Ağı, Karar Ağacı, C4.5

"Comparatıve Study Of Machıne Learnıng Knn, Svm, And Decısıon Tree Algorıthm To Predıct Student's Performance " [9] başlıklı 2019 yılında yayınlanan çalışmada, öğrenci performansının makine öğrenimi algoritmalarıyla tahmin edilmesi konusunu ele alan bir araştırma yapılmıştır. Çalışma; k-En Yakın Komşular, Destek Vektör Makineleri, Karar Ağaçları dâhil olmak üzere birçok popüler makine öğrenimi algoritmasının performansını karşılaştırarak farklı tekniklerin kapsamlı bir değerlendirmesini barındırmaktadır. Araştırmada, bir üniversitenin Bilgisayar Mühendisliği bölümündeki öğrencilerinin akademik verileri kullanılmıştır. Veri seti, 1530 satır ve 7 öznitelikten oluşmaktadır. İlk 6 değişken, 7. değişkenin tahmin edilmesinde kullanılmıştır. Öğrenci performansı, tahmin edilmek istenen hedef değişkendir. Çalışmada, algoritmaların performansını değerlendirmek için doğruluk, kesinlik ve duyarlılık dâhil olmak üzere birden fazla değerlendirme metriği kullanılmıştır. Ancak çalışmada, yalnızca bir üniversiteden elde edilen veriler kullanılmıştır. Bu durum bulguların diğer üniversitelere veya bölgelere genellenebilirliğini sınırlayabilmektedir.

"Machine Learning Based Predicting Student Academic Success." [10] başlıklı çalışmada, öğrencilerin gelecek bir dersi geçip geçmeyeceğini tahmin etmek amacıyla makine öğrenmesine dayalı bir model geliştirilmiştir. Model, öğrencilerin geçmiş akademik başarı verilerine dayanarak tahmin yapmaktadır. Tahminler, sınavdan belirli bir süre öncesinde yapılır (örneğin, 2-4 ay öncesinde). Veri toplama aşamasında, Nizwa Üniversitesi'nin Sanat ve Bilimler Fakültesi'ndeki Fizik ve Matematik Bilimler Bölümü, Bilgisayar Bilimleri Bölümünden öğrenci verileri kullanılmıştır. Çalışma ayrıca doğruluk, kesinlik ve duyarlılık gibi metrikleri kullanan tahmine dayalı modellerin değerlendirmesini içermektedir. Bununla birlikte çalışma, tek bir üniversite ile sınırlı olduğu için bulgular diğer üniversitelere veya farklı eğitim ortamlarına genellenememektedir.

"Predicting Students' Performance Using Machine Learning Techniques." [11] adlı çalışma, İnsani Bilimler Fakültesi tarafından sunulan Bilgisayar Bilimi dersinde öğrencilerin performansını tahmin edebilen bir sınıflandırıcı oluşturmak için dört makine öğrenimi tekniğinin kullanıldığı bir çalışmadır. Makine öğrenimi modelleri arasında Yapay Sinir Ağları, Naive Bayes, Karar Ağacı ve Lojistik Regresyon bulunmaktadır. Bu araştırma, öğrencilerin öğrenme için interneti kullanma etkisi ve öğrencilerin sosyal ağlarda geçirdikleri zamanın performansları üzerindeki etkisi üzerine yapılmıştır. Bu

etkiler, öğrencinin öğrenme için interneti kullanıp kullanmadığını ve öğrencilerin sosyal ağlarda geçirdikleri zamanı ölçen özellikler kullanılarak tanımlanmıştır. Modeller ROC indeksi performans ölçütü ve sınıflandırma doğruluğu kullanılarak karşılaştırılmıştır. Ayrıca, sınıflandırma hatası, hassasiyet, geri çağırma ve F1 score gibi farklı ölçümler hesaplanmıştır. Modellerin oluşturulmasında kullanılan veri seti, öğrencilere verilen bir ankete ve öğrencilerin not listelerinden toplanan verilere dayanmaktadır. Ancak çalışma, sınırlı sayıda öğrenci verisi kullanır ve bu nedenle sonuçları genelleştirmek için daha fazla veriye ihtiyaç duyulabilmektedir. Ayrıca veri seti oluşturulurken yalnızca bir kurumun verileri kullanılmıştır. Bu sebeple sonuçlar tüm bölgelere genellenememektedir.

"Predıctıng Academıc Performance of Engıneerıng Students Usıng Ensemble Method." [12] başlıklı çalışma mühendislik öğrencilerinin akademik performansını tahmin etmeyi amaçlamaktadır. Geçmiş eğitim kayıtları, demografik faktörler, aile faktörü ve diğer ilgili faktörlere dayanarak bir öğrencinin akademik performansını tahmin etmek için bir model oluşturulmuştur. İlk olarak, Decision Tree, SVM ve Lineer Regresyon gibi geleneksel sınıflandırıcılar kullanılarak bir tahmin modeli oluşturulmuştur. Ardından, bireysel sınıflandırıcıların performansını iyileştirmesiyle bilinen popüler bir Ensemble Yöntemi olan voting uygulanmıştır. Sonuçlar; doğruluk, hassasiyet, hatırlama ve F1 skoru açısından bireysel sınıflandırıcılardan önemli ölçüde daha iyi performans göstermektedir. Çalışmada kullanılan veriler, 2004-2015 yılları arasında Paschimanchal Mühendislik Kampüsü, Pokhara'dan her mezun öğrencinin kişisel dosyasından doğrudan toplanmıştır. Veri kümesinin sadece bir üniversiteden toplanması sonuçların genelleştirilebilirliğini sınırlamaktadır.

"Modeling and Predicting Students' Academic Performance Using Data Mining Techniques." [13] çalışmada öğrencilerin akademik performansını öngörmek ve analiz etmek için veri madenciliği tekniklerini uygulamaktır. Çalışmada, iki lisans dersinden öğrenci verileri toplanmıştır. Veri seti üzerinde üç farklı veri madenciliği sınıflandırma algoritması (Naive Bayes, Yapay Sinir Ağı ve Karar Ağacı) kullanılmıştır. Üç sınıflandırıcı modelin tahmin performansı ölçülmüş ve karşılaştırılmıştır. Naive Bayes sınıflandırıcının diğer iki sınıflandırıcıdan daha iyi performans göstermiş, genel tahmin doğruluğu %86 olarak elde edilmiştir. Çalışmanın öğretmenlere, öğrencilerin akademik performanslarını iyileştirmede yardımcı olması amaçlanmaktadır. Çalışma tek bir bölgede yürütülmüştür, dolayısıyla sonuçlar diğer ortamlara genellenememektedir.

Çizelge 2.1' de verilen çalışmaların güçlü yönleri birden fazla makine öğrenmesi kullanılarak sistemin eğitilmesi ve test edilmesi, en iyi sonuç alınan modelle sistemin çalışmasına devam edilmesidir. Ayrıca başarı ölçütü olarak doğruluk, kesinlik, duyarlılık ve F1 skor gibi farklı ölçütler kullanılarak sistemin başarıları test edilmiştir. Ancak bu çalışmaların zayıf yönleri de vardır. Bunlar çalışmalar bir bölgede, bir üniversitede veya bir kurumda yapılmış ve veri seti oluşturulurken bu alanın dışına çıkılmamıştır. Bu durum da çalışmaların farklı bölgelere veya üniversitelere uygulanabilirliği açısından sorun oluşturabilmektedir.

1.2 Tezin amacı

Birey; yetenek, ilgi ve istekleri doğrultusunda meslek olarak seçtiği alanda başarılı, verimli ve mutlu olur. Öğrenci özelliklerini göz önünde tutmadan rastgele seçim yaptığında başarısız, verimsiz ve mutsuz olur. Bu nedenle birey, meslek seçerken kendi özellikleri ile seçeceği mesleğin nitelikleri arasında uygunluk olmasına dikkat etmelidir [14]. Lise öğrencilerinin meslek seçimi yaparken temel aldığı kriterler üzerine yapılan çalışmada, meslek seçimini etkileyen faktörler şu şekilde sıralanmıştır: meslekte iş bulma imkânı, yetenek, ilgi, değerler, kişilik özellikleri (kendini tanıma), mesleğin getirileri (para, saygınlık, şöhret vb.) ve ailenin isteği şeklindedir [15]. Türkan Sarıkaya ve Leyla Khorshid yaptıkları çalışmada, öğrencilerin 1/3'ü (%33.6) meslekle ilgili olumlu görüşleri, %23.5'i çeşitli nedenlerle çaresizlik yaşadığı ve aldığı puanla açıkta kalmamak için, %28.2'si meslekle ilgili avantajları olduğunu düşündüğü için, %14.7'si ise mezun olduğunda sahip olacağı mesleği seçtiğini belirtmiştir [16]. Yapılan bu çalışmadan da gördüğümüz gibi mezun olduğunda sahip olacağı mesleği dikkate alarak bölüm tercihi yapan öğrencilerin oranı %14,7'de kalmaktadır. Bu şekilde başarı durumuna uygun yapılmayan bölüm tercihlerinde öğrencilerin başarılı olma oranları düşük olmakta, bu durumda mezun olan öğrenciler de mezun oldukları alanı meslek olarak tercih etmemektedirler.

Günümüzde kendi başarı durumlarını ve kişisel özelliklerini dikkate alarak iş hayatında sahip olacağı mesleği tercih eden öğrencilerin oranı çok düşük kalmaktadır. Bu oranı yükseltmek ve öğrencilerin başarı durumlarına göre bölüm önerilerinde bulunmak amacıyla bu tez çalışmasında makine öğrenmesi algoritmaları kullanılarak lise öğrencilerinin ders başarıları ile üniversite ve iş hayatında başarılı olması muhtemel bölüm tercihi tahmini yapılacaktır.

Öğrenciler üniversite bölüm tercihlerini yaparken çevresindeki insanların görüşlerine, mesleğin ekonomik ve sosyal olanaklarına, ayrıca statü avantaj ve dezavantajlarına dikkat etmektedirler. Bu bozucu etkiler öğrencilerin bölüm tercihini etkilemekte, kendi ilgi duydukları ve başarılı olabilecekleri alanları tercih sırasında geri plana itmektedirler. Bu çalışma sayesinde öğrenciler çevrelerindeki insanların görüşlerinden etkilenmeden ve ekonomik kaygı gütmeden sisteme lise dönemine ait Matematik, Fizik, Kimya, Biyoloji, Coğrafya, Türkçe, Tarih ve Yabancı Dil ders başarı puanlarını girerek üniversitede hangi bölümde başarılı olabileceğini tahmin edebilecektir. Bu sayede öğrenciler ilgi duyduğu bölümler içinden başarılı olması muhtemel bölümü tercih edecek ve de kazandığı bölümü hem sevecek hem de başarılı olacaktır.

1.3 Hipotez

Bu çalışmanın hipotezi, lise öğrencilerinin üniversite tercih aşamasında yaşadıkları zorlukları azaltmak için, öğrencilerin ders notları gibi akademik performansları temel alınarak, öğrencilere başarılı olabilecekleri bölümler konusunda öneriler sunulması gerektiğidir. Bu öneriler, mesleklerin sosyo-ekonomik etkileri veya öğrencilerin çevrelerindeki insanların görüşleri gibi dış faktörlerden bağımsız olarak, kendi ders başarılarına dayanacaktır. Böylece öğrenciler, kendilerini daha iyi tanıyabilecekler ve bu sayede rehberlik hizmetlerinin de daha etkili olmasını sağlayabileceklerdir. Bu hipotez doğrultusunda, öğrencilerin lise ders notlarını sisteme girecekleri ve bu sistemin öğrencilere başarılı olabilecekleri bölümler konusunda öneriler sunacağı bir çalışma yürütülecektir.

1.4 Çalışmanın sınırlılıkları

Veri toplama ile ilgili sınırlılıklar;

- Veriler toplanırken katılımcıların mezun oldukları lise türü dikkate alınmamıştır. Ülkemizde mevcut lise türlerinde; müfredattan, ders sürelerinden ve uygulanan sınavlardan kaynaklı farklılıklar çalışmanın dışında tutulmuştur.
- Ayrıca katılımcıların yalnızca mezun oldukları bölüm dikkate alınmış, mezun olduğu üniversite dikkate alınmamıştır.
- Veri toplama işleminde verinin kalitesi ve belirli bir zaman diliminde tamamlanması gerekliliği verilerin toplanmasında katılımcı sayısını ve verilerin doğruluğunu etkilemektedir. Katılımcı sayısının azlığı ve verilere doğru

olmayan bilgilerin karışması uygulamanın çalışmasını olumsuz etkilemektedir. Veri toplama işleminde katılımcı azlığı, bazı öğrenme modellerinde öğrenme başarılarının düşük olmasına neden olmuştur.

- Çalışmamızda kullanılan özellikler sadece öğrencilerin lise notlarına dayanmaktadır, diğer faktörler (örneğin, sosyoekonomik durum, öğrencinin kişisel özellikleri vb.) dikkate alınmamıştır.

2. MATERYAL YÖNTEM

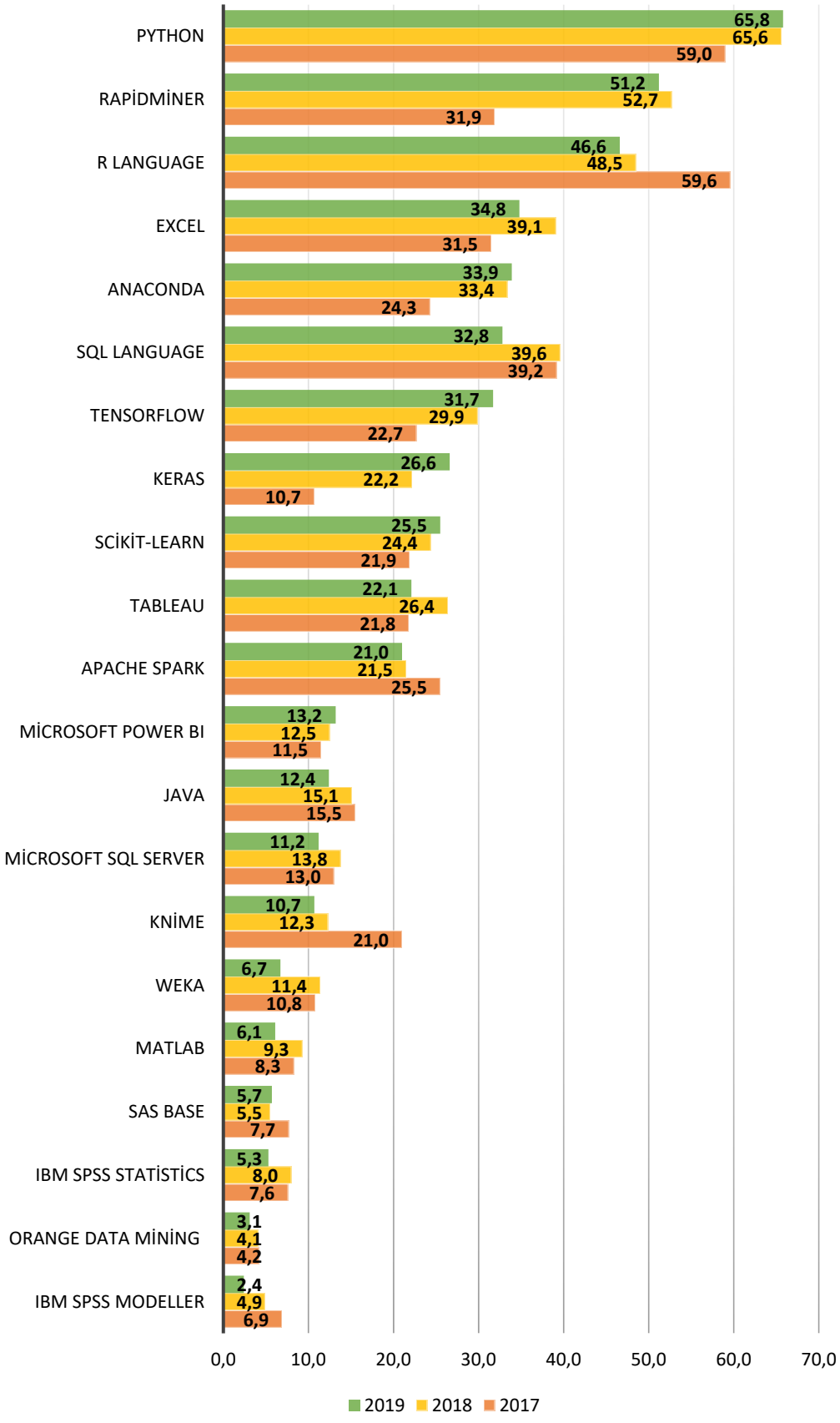
2.1 Makine Öğrenmesi Kavramı

Teknoloji, günümüzün vazgeçilmezi halinde gelmiştir. Teknolojinin gelişmesiyle de verinin toplanması ve işlenmesi kolaylaşmıştır. Toplanan bu veriler tek başına anlam taşımamaktadır, ancak işlendikleri zaman anlamlı olabilir. Bu verileri işlemek ve anlamlı sonuçlar elde edebilmek için Makine Öğrenmesi gibi birçok yöntem bulunmaktadır. Şekil 2.1'de gösterildiği gibi Makine Öğrenmesi ile Bilgisayar Bilimi, Matematik ve İstatistik disiplinleri yakından ilişkilidir. Çalışmamızda veri işleme yöntemlerinden makine öğrenmesi tercih edilmiştir. Makine Öğrenmesi, Bilgisayar Bilimi, Matematik ve İstatistik ve Veri Bilimi ile ilişkilidir.

Şekil 2.1 Makine öğrenmesi ve ilişkili disiplinler [5].

2.2 Makine Öğrenimi Araçları

Şekil 2.2' de 2017-2019 yılları arasında en çok tercih edilen makine öğrenmesi araçları verilmiştir. Burada yazılım dilleri, kütüphaneler ve uygulamalar mevcuttur. Şekil, Piatestsky [17] tarafından 2019'da makine öğrenmesi ve yapay zekâ alanında çalışan yazılımcılara yöneltilen bazı sorulara verilen cevaplar incelenerek oluşturulmuştur. Makine öğrenmesinde en fazla tercih edilen yazılım dilleri ve uygulamaları aşağıda açıklanmıştır.



Şekil 2.2 En yaygın kullanılan makine öğrenimi araçları [17].

2.2.1 Python: En yaygın kullanılan makine öğrenmesi dilidir. Python, birçok makine öğrenmesi kütüphanesi ile birlikte gelir ve kolayca öğrenilebilen bir dil olarak kabul edilir.

2.2.2 R: Özellikle veri analizi ve istatistiksel hesaplamalar için kullanılan bir programlama dili olarak bilinir. R ayrıca makine öğrenmesi için birçok kütüphane sağlar.

2.2.3 Java: Büyük ölçekli uygulamalar içinde programlama dili olarak popülerdir. Makine öğrenmesi için Java kullanmak, özellikle büyük ölçekli projeler için tercih edilir.

2.2.4 C++: Yüksek performanslı uygulamalar için kullanılan bir dildir. Makine öğrenmesi için özellikle sınıflandırma ve kümeleme algoritmaları gibi yoğun hesaplama gerektiren alanlarda kullanılır.

2.2.5 MATLAB: Özellikle mühendislik ve bilimsel araştırmalar için kullanılan bir programlama dili ve hesaplama ortamıdır. MATLAB, makine öğrenmesi için birçok kütüphane sağlar.

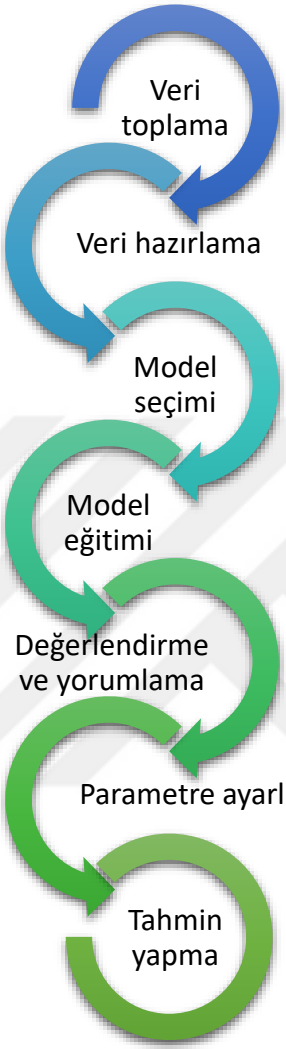
Bu diller arasında Python, en yaygın kullanılan ve özellikle başlangıç seviyesi için uygun bir dil olarak kabul edilir. Ancak, proje ihtiyaçlarına ve kişisel tercihlere göre farklı dillere de başvurulabilir. Çalışmamızda yaygın kullanımı ve kullanım kolaylığı nedeniyle Python programlama dili tercih edilmiştir.

2.3 Makine öğrenmesi aşamaları

Şekil 2.3'de gösterildiği gibi makine öğrenmesinde 7 aşama vardır [18]. Bunlar; Veri toplama, veri hazırlama, model seçimi, model eğitimi, değerlendirme ve yorumlama, parametre ayarlama ve tahmin yapmadır [19]. İlk aşamada toplanan veriler modelin eğitilmesi ve test edilmesi için kullanılmak üzere gruplanır. İkinci aşamada kullanılacak eğitim modeli seçilir. İlk aşamada gruplanan eğitim verileriyle seçilen model eğitilir. Eğitim sonunda modelin test edilmesi için ayrılan veri grubuyla test edilerek sistemin öğrenme oranı değerlendirilir. Değerlendirme sonunda elde edilen veriler yorumlanarak uygun parametre ayarları yapılır. Sistemin durumuna göre eğitim ve test süreci tekrarlanır. Sistem yeterli öğrenme oranına ulaştıktan sonra model tahmin yapmak için hazırdır.

Çalışmamızda da CRISP-ML aşamaları dikkate alınarak ilk olarak veri toplama işlemi gerçekleşmiş sonrasında veriler makine öğrenmesi uygulaması için uygun hale

getirilmiştir. Sonrasında çalışmamıza uygun makine öğrenmesi modelleri seçilmiş ve bu modeller eğitilmiştir. Daha sonra modellerden elde edilen başarılar değerlendirilmiş ve en uygun model ve parametreler kullanılarak sistem hazır hale getirilmiştir.



Şekil 2.3 CRISP-ML Makine öğrenimi aşamaları.

2.4 Veri Toplama

Makine öğrenmesinde veri toplama, bir modelin eğitimi için kullanılan verilerin toplanması işlemidir. Veriler, modelin bir problemi çözmek için öğrenmesi gereken bilgileri içerir.

Veri toplama yöntemleri, kullanılan verilerin türüne ve kaynağına bağlı olarak değişebilir. Bazı yaygın veri toplama yöntemleri şunlardır:

- **Web Scraping:** İnternet sitelerindeki verilerin otomatik olarak toplanması işlemidir. Örneğin, bir ürünün fiyatı gibi bilgileri toplamak için kullanılabilir.
- **Veri Tabanlarından Veri Çekme:** Büyük veri tabanlarından verilerin çekilmesi işlemidir. Örneğin, bir bankanın müşteri verilerinin toplanması.
- **Sensörler:** Bir cihazdan ölçülen verilerin toplanmasıdır. Örneğin, bir otomobilin hızı veya bir kalp atış hızı gibi.
- **Anketler:** Belirli bir konuda sorular sorarak verilerin toplanmasıdır. Örneğin, bir pazar araştırması yaparak müşteri memnuniyeti hakkında bilgi toplanması.
- **Sosyal Medya:** Sosyal medya platformlarından verilerin toplanmasıdır. Örneğin, bir şirketin Twitter hesabındaki kullanıcı yorumlarının toplanması.
- **Kaynakların Birleştirilmesi:** Farklı kaynaklardan verilerin birleştirilerek kullanılmasıdır. Örneğin, bir şirketin müşteri verilerinin farklı veri tabanlarından çekilerek birleştirilmesi.

Çalışmamıza en uygun veri toplama yöntemi ankettir. Bu yöntem içinde en işlevsel bir alternatif olan Google Forms tercih edilmiştir. Kullanılan anket Ek-1'de sunulmuştur. Ayrıca Bitlis Eren Üniversitesi Etik Kurul Kararı Ek-2'de sunulmuştur.

2.5 Veri Hazırlama

Veri hazırlama, makine öğrenmesi sürecindeki en önemli aşamalardan biridir ve verilerin doğru bir şekilde kullanılabilmesi için hazırlanmalarını içerir. Bu aşama, verilerin toplanması, temizlenmesi, özellik mühendisliği, özellik seçimi, özellik ölçeklendirme ve veri bölümü gibi bir dizi işlemi kapsar.

Veri hazırlama aşamalarının detayları şöyle sıralanabilir:

- **Veri Toplama:** Makine öğrenmesi modelinin eğitilmesi için veri toplanması gerekir. Bu veriler doğru ve yeterli olmalıdır. Veri toplama sürecinde, veri kaynakları belirlenir ve veriler toplanır.
- **Veri Temizleme:** Toplanan verilerin kalitesi çok önemlidir. Verilerin içindeki hatalı, eksik veya yanlış veriler, veri analizi sonuçlarını

bozabilir. Bu nedenle, veri temizleme işlemi, verilerin doğruluğunu ve tutarlılığını sağlamak için yapılır.

- **Özellik Mühendisliği:** Verilerin özellikleri, modelin başarısı için çok önemlidir. Bu nedenle, verilerin özellikleri geliştirilir ve düzenlenir. Özellik mühendisliği, verilerin içindeki anlamlı bilgileri ortaya çıkarmak için yapılır.
- **Özellik Seçimi:** Verilerdeki tüm özellikler önemli değildir. Bazı özellikler diğerlerine göre daha önemlidir. Bu nedenle, özellik seçimi yapılır ve model için en önemli özellikler belirlenir.
- **Özellik Ölçeklendirme:** Veriler farklı birimlerde ve farklı aralıklarda olabilir. Bu nedenle, özellik ölçeklendirme işlemi yapılır. Bu işlem, verilerin aynı ölçekte olmasını ve eşit bir şekilde değerlendirilmesini sağlar.
- **Veri Bölümü:** Verilerin bir kısmı modelin eğitilmesi için kullanılır, diğer kısmı ise modelin test edilmesi için kullanılır. Veri bölümü işlemi, verilerin rastgele bir şekilde bölünmesini ve modelin eğitim verilerinde öğrenmesini, test verilerinde ise performansının değerlendirilmesini sağlar.

Bu işlemler, veri hazırlama aşamasının temel adımlarını oluşturur. Doğru veri hazırlama işlemi, modelin doğru sonuçlar vermesini sağlayan en önemli adımlardan biridir.

2.6 Öğrenme Türleri

Makine öğrenmesi ve yapay zekâ çalışmalarında öğrenme türlerinin önemi büyüktür. Yapılacak çalışmaya uygun bir öğrenme türü seçilmesi öğrenme başarısı açısından önemlidir. Makine öğrenmesinde kullanılan öğretim türleri, eğitim alanında da birçok problemin çözümünde kullanılmaktadır.

2.6.1 Denetimli Öğrenme: Sınıflandırma problemlerinin çözümünde yaygın olarak kullanılan bir öğretim türüdür. Eğitim alanında da öğrencilerin notlarını veya başarı durumlarını sınıflandırmak için kullanılabilir.

2.6.2 Denetimsiz Öğrenme: Bu öğrenme türü, verilerdeki yapıları bulmak ve sınıflandırma yapmak için gerekli olan etiketli verilerin olmadığı durumlarda kullanılır.

2.6.3 Yarı-Denetimli Öğrenme: Bu öğrenme türü, hem etiketli hem de etiketsiz verilerin kullanıldığı bir öğrenme türüdür. Eğitim alanında öğrencilerin öğrenme düzeylerinin ölçüldüğü bir sınavda bazı soruların cevapları etiketlenirken, diğer soruların cevapları etiketlenmez.

Çalışmamıza uygun olan öğrenme türü denetimli öğrenmedir. Diğer öğrenme türleri çalışmamız için uygun değildir. Bu nedenle çalışmamızda tercih edilmemiştir.

2.7 Öğrenme Modelleri

Makine öğrenmesi modelleri, veri analizi ve desen tanıma yoluyla otomatik olarak öğrenme yapabilen sistemler oluşturmayı amaçlar. Bu modeller, veri setleri üzerinde çalışarak desenleri, ilişkileri ve trendleri belirler ve bu bilgileri kullanarak gelecekteki verilere dayalı tahminler yapabilir veya kararlar alabilir. Makine öğrenmesi modelleri genellikle Sınıflandırma, Regresyon, Kümeleme, Boyut Azaltma, Öneri Sistemleri ve Doğal Dil İşleme gibi alanlarda kullanılır.

Bunlar sadece bazı örnekler olup, makine öğrenmesi modelleri birçok farklı alanda kullanılmaktadır ve sürekli olarak yeni uygulama alanları ortaya çıkmaktadır.

2.7.1 Support Vector Machine: SVM, sınıfları marjinalleştiren ve aralarındaki mesafeleri daha net bir şekilde ayırt etmek için en üst düzeye çıkaran bir hiper düzlem çizer [20]. SVM, sınıflandırma ve regresyon problemlerinde kullanılan güçlü bir makine öğrenmesi algoritmasıdır. SVM' nin temel fikri, veri noktalarını bir özellik uzayında temsil ederek, sınıflar arasında en iyi ayrımı yapacak bir hiper düzlem bulmaktır. Bu hiper düzlem, veri noktalarını iki sınıfa ayırmak için kullanılır. SVM, bu hiper düzlemi oluştururken, sınıflara ait veri noktalarının marjinalliğini (maksimum ayrımla çevrelenen alan) maksimize etmeye çalışır. Bu şekilde, yeni veri noktalarını doğru şekilde sınıflandırmak için maksimum ayrım elde edilir. SVM, veri noktalarını sınıflandırmak için lineer bir hiper düzlem kullanabilir veya çekirdek fonksiyonları aracılığıyla verileri daha yüksek boyutlu bir uzaya taşıyarak non-lineer ayrımlar yapabilir. Özellikle, çekirdek fonksiyonları sayesinde SVM, daha karmaşık veri setlerinde de etkili bir şekilde çalışabilir. SVM, genellikle sınıflandırma problemlerinde kullanılsa da regresyon problemlerinde de kullanılabilmektedir. Regresyon için kullanılan SVM, destek vektör regresyonu (Support Vector Regression - SVR) olarak adlandırılır ve veri noktalarını bir regresyon çizgisi yerine bir hiper düzlemle tahmin etmeye çalışır.

Doğrusal olarak ayrılabilen iki sınıflı bir problem için karar fonksiyonu denklem 2.1' de gösterilen şekilde yazılabilir.

$$f(x) = sign\left(\sum_{i=1}^{k} \lambda_i y_i (x.x_i) + b\right) \qquad (2.1)$$

Çalışmamızda olduğu gibi birçok problemde verilerin doğrusal olarak ayrılması mümkün değildir. Bu durumda eğitim verilerinin bir kısmının optimum hiper-düzlemin diğer tarafında kalmasından kaynaklanan problem pozitif bir yapay değişkenin (i ξ) tanımlanması ile çözülür. Sınırın maksimum hale getirilmesi ve yanlış sınıflandırma hatalarının minimum hale getirilmesi arasındaki denge, pozitif değerler alan ve C ile gösterilen bir düzenleme parametresi (0<c<∞) tanımlanmasıyla kontrol edilebilir [21].

$$min\left[\frac{\|w\|^2}{2} + c\sum_{i=1}^{r} \xi_i\right] \qquad (2.2)$$

Düzenleme parametresi ve yapay değişken kullanılarak doğrusal olarak ayrım yapılamayan veriler için optimizasyon problemi denklem 2.2'deki gibi yazılabilir.

$$y_i(w.\varphi(x_i) + b) - 1 \geq -\xi_i \qquad (2.3)$$

$$\xi_i \geq 0 \; ve \; i = 1, \ldots, N$$

Buna bağlı sınırlılıklar ise denklem 2.3' deki şekilde ifade edilir.

$$K\left(x_{i,}x_j\right) = \varphi(x).\varphi(x_j) \qquad (2.4)$$

Ayrıca Kernel fonksiyonları kullanılmak istenirse matematiksel olarak denklem 2.4' de ifade edilen şekilde kullanılabilir. Sonuç olarak, kernel fonksiyonu kullanılarak doğrusal olarak ayrılamayan bir problemin çözümünde kullanılacak karar kuralı denklem 2.5' de gösterildiği şekliyle yazılabilir.

$$f(x) = sign(\sum_i \alpha_i y_i \varphi(x).\varphi(x_i) + b) \qquad (2.5)$$

2.7.2 K-Nearest Neighbors (K-En Yakın Komşu): K-En Yakın Komşu (KNN) makine öğrenmesi algoritması, en basit sınıflandırma algoritmalarından biridir. KNN, bir veri noktasının sınıfını belirlemek için diğer veri noktalarıyla olan uzaklıklarını hesaplar. Etiketlenmiş bir veri sınıflandırıcıya verildiğinde kendisine en yakın k-nesneleri için hesaplanan uzaklıklarda en sık görülen etiketi belirleyerek sınıfa atar [22]. Ancak, sadece k adet en yakın veri noktası dikkate alınır. Bu k veri noktası, hesaplanan mesafeye göre, diğer verilerden daha yakın olan verilerdir. Algoritmanın kullanımı için aşağıdaki adımlar izlenir:

1. Eğitim Veri Setinin Oluşturulması: İlk adım, sınıf değerleri bilinen örneklerden oluşan bir eğitim veri seti oluşturmaktır. Her veri örneği, özellik vektörü (x) ve ona karşılık gelen sınıf etiketi (y) ile temsil edilir.
2. K Parametresinin Belirlenmesi: K-en yakın komşu algoritmasında, sınıflandırılacak örneklemin etrafındaki en yakın k komşuyu seçmek için k değeri belirlenir. Bu parametre, genellikle kullanıcı tarafından belirlenir ve deneylerle optimizasyonu yapılabilir.
3. Mesafe Fonksiyonunun Belirlenmesi: K-en yakın komşu algoritması, örneklemler arasındaki benzerliği veya uzaklığı ölçmek için Öklid veya Manhattan mesafesi gibi bir mesafe fonksiyonu kullanır. Mesafe fonksiyonu, örneklemlerin özellik vektörleri arasındaki farkı hesaplar.
4. Örneklemin Sınıflandırılması: Sınıflandırılacak örneklemin sınıfını belirlemek için aşağıdaki adımlar izlenir:
    a. Sınıflandırılacak örneklemin, eğitim veri setindeki tüm örneklerle olan mesafeleri hesaplanır.
    b. Hesaplanan mesafelere göre en yakın k örneği seçilir.
    c. Seçilen bu k örneğin sınıf etiketleri incelenir.
    d. Sınıflandırma için en çok tekrar eden sınıf etiketi, tahmin edilen örneklemin sınıfı olarak atanır.

2.7.3 XGBoost: XGBoost, makine öğrenimi alanında kullanılan bir algoritmadır ve ekstrem gradyan artırma yöntemini temel alır. Sınıflandırma ve regresyon problemlerinde etkili bir şekilde kullanılabilen XGBoost, yüksek performans, ölçeklenebilirlik ve doğru tahmin yeteneği gibi avantajlar sunar. Python, R, Java, C++ gibi programlama dillerinde uygulanabilir ve sınıflandırma, regresyon, sıralama ve anomalik durum tespiti gibi çeşitli görevleri başarabilir. 2015 yılında yapılan 29 Kaggle yarışmasından 17 tanesi XGBoost kullanılarak kazanılmıştır [23]. XGBoost'un geliştirilmesi, veri bilimciler tarafından halen devam etmektedir. XGBoost, makine öğrenimi projelerinde tercih edilen bir algoritmadır.

2.7.4 Gaussian Naive Bayes: Gaussian Naive Bayes, ilgili bir olayın oluşumunu temel alarak bir olayın oluşma olasılığını hesaplar [20]. Bu algoritma, özniteliklerin normal dağılım gösterdiğini varsayar ve Bayes teoremini kullanarak sınıf tahminlemesi yapar.

Basit bir model olmasına rağmen genellikle iyi performans gösteren Gaussian Naive Bayes, hızlı çalışır ve düşük bellek gereksinimleriyle uygulanabilir.

Algoritma işleyişi veri setinin frekans tablosuna dönüştürülmesi ile başlar. Sonrasında olasılıklar hesaplanır ve olasılık tablosu oluşturulur. Ardından her bir sınıfın sonsal olasılığını hesaplamak için aşağıda denklem 2.6' da verilen formül uygulanır. Örneklemin ait olduğu sınıfı bulmak için yapılan tahmin en yüksek sonsal olasılığa sahip sınıf olarak yapılır.

$$P(a|b) = \frac{P(a|b)*P(a)}{P(b)} \qquad (2.6)$$

Şekil 2.6'da yer alan formülde;

- P (A | B) = B olayı gerçekleştiğinde A olayının gerçekleşme olasılığını
- P (A) = A olayının gerçekleşme olasılığını
- P (B | A) = A olayı gerçekleştiğinde B olayının gerçekleşme olasılığını
- P (B) = B olayının gerçekleşme olasılığını temsil etmektedir.

Bu modeller, eğitim alanında kullanılan en yaygın yapay zekâ modelleridir ve öğrenci performansının tahmin edilmesi, öğrencilerin ihtiyaçlarının belirlenmesi ve eğitim programlarının özelleştirilmesi için kullanılabilir.

2.8 Model Seçimi

Makine öğrenmesi uygulamalarında model seçimi, projenin gereksinimlerine ve veri setinin özelliklerine bağlı olarak yapılır. Burada dikkate alınması gereken bazı faktörler şunlardır:

- **Veri seti boyutu:** Veri seti boyutu, hangi modelin seçileceğinde önemli bir faktördür. Büyük veri setleri genellikle karmaşık modelleri gerektirirken, küçük veri setleri daha basit modelleri tercih eder.
- **Veri tipi:** Veri tipi, model seçiminde dikkate alınması gereken diğer bir faktördür. Veri seti, sayısal veya kategorik olabilir. Sayısal veri setleri genellikle regresyon modellerinde kullanılırken, kategorik veri setleri sınıflandırma modellerinde tercih edilir.
- **Veri kalitesi:** Veri kalitesi, model seçiminde dikkate alınması gereken önemli bir faktördür. Eksik veya yanlış veriler, modelin doğruluğunu

etkileyebilir. Bu nedenle, veri seti kalitesini artırmak için veri ön işleme teknikleri kullanılabilir.

- **Hızlı tahmin yapma gereksinimi:** Hızlı tahmin yapma gereksinimi olan uygulamalarda, derin öğrenme modelleri kullanmak yerine daha basit modeller tercih edilebilir.
- **Amaç:** Son olarak, uygulamanın amacı, hangi modelin seçileceğinde önemli bir faktördür. Örneğin, doğrusal regresyon modelleri, bir değişkenin diğer değişkenler üzerindeki etkisini tahmin etmek için kullanılırken, karar ağaçları sınıflandırma problemlerini çözmek için kullanılabilir.

Bu faktörlerin yanı sıra, projenin gereksinimleri ve geliştiricinin deneyimi de model seçiminde etkili olabilir. Bu nedenle, en uygun modeli seçmek için birden fazla model denemek ve performanslarını karşılaştırmak önemlidir. Veri setimizdeki verilerin boyutu az ve sayısal veri tipine sahip olduğu için K en yakın komşu, Gaussian Naive Bayes, XGBoost ve destek vektör makineleri çalışmamız için uygun makine öğrenmesi modelleridir.

2.9 Test

Makine öğrenmesi çalışmalarının başarısını ölçmek için kullanılabilecek birçok ölçüt vardır. Seçilen ölçütler, kullanılan algoritma ve modelin türüne, problemin türüne ve projenin amaçlarına göre değişebilir. Ancak genellikle kullanılan en yaygın ölçütler şunlardır:

- **Confusion Matrix (Hata Matrisi):** Modelin yanlış sınıflandırma yapma hatalarını gösteren ve modelin hangi sınıflara karşı daha iyi veya daha kötü performans gösterdiğini anlamamızı sağlayan metriktir.

		Tahmin Edilen Sınıf	
		Sınıf = 1	Sınıf = 0
Gerçek Sınıf	Sınıf = 1	True Pozitif (TP)	False Negatif (FN)
	Sınıf = 0	False Pozitif (FP)	True Negatif (TN)

Bu değerler kullanılarak Accuracy, Precision, Recall F1 score gibi metrikler hesaplanabilir.

- **Accuracy (Doğruluk):** Modelin tahminlerinin doğruluğunu ölçen temel bir ölçüttür. Doğruluk, doğru tahminlerin toplam sayısının toplam örnek sayısına oranıdır [24].

  $$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (2.7)$$

- **Precision (Hassasiyet):** Modelin, gerçek pozitif sonuçlarının (true positive) toplam pozitif tahminlerine (true positive + false positive) oranını ölçer. Hassasiyet, bir modelin yanlış pozitif sonuçlar (false positive) verme olasılığına karşı direncini ölçer [24].

  $$Precision = \frac{TP}{TP+FP} \quad (2.8)$$

- **Recall (Duyarlılık):** Modelin, gerçek pozitif sonuçların toplam pozitif örneklere (true positive + false negative) oranını ölçer. Duyarlılık, bir modelin yanlış negatif sonuçları (false negative) verme olasılığına karşı direncini ölçer [24].

  $$Recall = \frac{TP}{TP+FN} \quad (2.9)$$

- **F1 Score (F1 Puanı):** Hassasiyet ve Duyarlılığın harmonik ortalamasıdır. F1 puanı, hem hassasiyetin hem de duyarlılığın bir arada göz önüne alındığı ve bu nedenle bir modelin performansının daha kapsamlı bir şekilde ölçüldüğü bir ölçüttür [24].

  $$F1\ Score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (2.10)$$

Makine öğrenmesi algoritmalarının başarı ölçümlerinde Accruracy, Precision, Recall, F1 Score, Auc, Roc Eğrisi gibi birçok metrik bulunmaktadır. Ancak çalışmamızda sıklıkla kullanılan ve öğrenme başarısı konusunda önemli bilgiler veren Doğruluk (Accuracy), Duyarlılık (Recall), F1 Puanı (F1 Score) ve Hassasiyet (Precision) başarı ölçütleri kullanılacaktır.

3. BULGULAR VE TARTIŞMA

Lise eğitimini başarıyla tamamlayarak üniversite tercihi aşamasına gelen tüm öğrencilerin karşı karşıya kaldığı en önemli soru "hangi bölümü tercih etmeliyim?" sorusudur. Çalışmamızda öğrencilerin bu sorusuna öneri sunarak cevap bulmasına yardımcı olmak amaçlanmıştır. Bu amaç doğrultusunda, öncelikle veri seti oluşturmak amacıyla üniversite mezunlarına uygulamak üzere anket hazırlanmıştır. Bu anket katılımcılara uygulanarak veri seti oluşturulmuştur. Daha sonra bu veri seti üzerinde tezin ilerleyen bölümlerinde detayları açıklanan makine öğrenmesi algoritmaları ile çeşitli modeller oluşturulmuş, regresyon işlemleri gerçekleştirilmiş ve yine tezin ilerleyen bölümlerinde detayları açıklanan yöntemler kullanılarak bu modellerin farklı ölçütler bazında performans geliştirilmesi üzerinde durulmuştur. Son olarak yine tezin ilerleyen kısımlarında detayları açıklanan ölçütlere göre performans karşılaştırması yapılmıştır. Çalışmanın geliştirilmesinde Python programlama dili kullanılmıştır.

3.1 Deneysel Kurulum

Bu çalışmada bulguların elde edilebilmesine ilişkin deneysel kurulumu;

1. Üniversite mezunlarına yönelik hazırlanan anketin katılımcılar tarafından doldurulması,
2. Bu sayede elde edilen verilerin temizlenmesi, eksik verilerin tamamlanması ve bozucu etkisi bulunan verilerin çıkartılması ile yeni veri seti oluşturulması,
3. Oluşturulan bu yeni veri setinin eğitim veri seti ve test veri seti olarak 2 gruba ayrılması,
4. Başarılı olması muhtemel bölümün tahmin edilebilmesi için eğitim veri seti ve makine öğrenmesi algoritmaları kullanılarak sınıflandırma modellerinin oluşturulması,
5. Veri setlerinde boyut azaltma ve algoritmalarda parametre ayarlama işlemleri yapılarak modellerin performansının artırılmaya çalışılması,
6. Modellerin test edilmesi,
7. Farklı algoritmalar kullanılarak oluşturulan modellerin performans kriterlerine göre incelenmesi ve başarılı modellere karar verilmesi olarak sunulacaktır.

Ayrıca çalışmada kullanılan programlama dili, kütüphaneler ve geliştirme ortamı ile ilgili detaylı bilgiler verilecektir.

3.1.1 Geliştirme Ortamı

Makine öğrenmesi alanında uygulamaların geliştirilmesi sırasında karşılaşılan sorunların hızlı bir şekilde çözüme kavuşturulması oldukça önemlidir. Bu sebeple, uygulamaların geliştirileceği ortam ve programlama dili seçimi büyük bir titizlikle yapılmalıdır. Makine öğrenmesi alanında, açık kaynak kodlu Python programlama dili sıklıkla tercih edilmektedir. Bu doğrultuda, Python programlama dili Anaconda platformu üzerinden kullanılmaktadır.

Anaconda, veri bilimi, makine öğrenmesi ve yapay zeka gibi alanlarda uygulama geliştirilmesinde tercih edilen Python veya R programlama dillerini kullanmak isteyenler için hazırlanmış bir tümleşik platformdur. İçerisinde birçok hazır paket barındıran Anaconda, veri bilimi, makine öğrenmesi ve yapay zeka çalışmalarında sıklıkla kullanılan kütüphanelere ek olarak Spyder ve Jupiter Notebook gibi geliştirme araçlarını da içermektedir. Bu çalışmada, veri hazırlama ve veri görselleştirme işlemleri için Jupiter notebook aracından yararlanılmıştır. Arayüz oluşturma işlemleri için Spyder aracı tercih edilmiştir.

Bu çalışmada kullanılan bazı Python kütüphaneleri arasında numpy, pandas, matplotlib, seaborn ve scikit-learn gibi popüler kütüphaneler yer almaktadır. Bu kütüphaneler, veri analizi ve makine öğrenmesi uygulamaları için oldukça faydalıdır.

Sonuç olarak, makine öğrenmesi alanında uygulama geliştirirken karşılaşılan problemlere hızlı bir şekilde çözüm bulabilmek için doğru bir geliştirme ortamı seçimi oldukça önemlidir. Python programlama dili ve Anaconda platformu, bu alanda oldukça etkili bir çözüm sunmaktadır. Bu çalışmada kullanılan bazı Python kütüphanelerine aşağıda yer verilmektedir.

3.1.2 Kullanılan Kütüphaneler

Python programlama dili, makine öğrenmesi alanında oldukça kullanışlı bir araç haline gelmiştir. Bu alanda, verilerin işlenmesi ve sınıflandırma algoritmalarının kullanılmasıyla modellerin geliştirilmesi için birçok kütüphane mevcuttur.

Numpy, veri ön işleme sürecinde ve verilerin hazırlanması aşamasındaki matematiksel işlemlerin yapılması amacıyla kullanılır. Genel olarak diziler ve matrisler üzerinde hesaplamalar yapmayı sağlar.

Pandas kütüphanesi, Numpy kütüphanesi ile birlikte verilerin hazırlanması için kullanılır. Veri setinin yüklenmesi, veri formatının ayarlanması ve veri setinden çerçeve oluşturarak işlemlerin yapılması amacıyla kullanılır.

Scikit-Learn kütüphanesi, sınıflandırma algoritmalarıyla model oluşturulması için kullanılır. Sınıflandırma, kümeleme ve regresyon için birçok algoritmayı içermektedir.

Tensorflow, Google tarafından derin öğrenme ve makine öğrenmesi alanlarında kullanılmak üzere geliştirilmiş açık kaynaklı bir kütüphanedir. Ekran kartının özelliğine bağlı olarak cpu veya gpu kullanımına uygun olarak çalışır.

Keras, TensorFlow'un üzerinde çalışabilen ve derin öğrenme modellerinin tanımlanması ve eğitilmesi için kullanılan bir üst düzey sinir ağları uygulama programlama arayüzüdür. Hızlı sonuç elde edilebilmesi için geliştirilmiş ve Python programlama diliyle yazılmıştır.

Bu kütüphaneler sayesinde, makine öğrenmesi alanında verilerin işlenmesi ve modellerin geliştirilmesi oldukça basit hale gelmiştir. Bu nedenle, Python programlama dili makine öğrenmesi alanında sıklıkla tercih edilen bir araçtır.

3.2 Veri Setinin Toplanması

Çalışmamızda veri toplama için Google Forms'dan faydalanılmıştır. Katılımcılara anket linkleri gönderilerek katılımları sağlanmıştır. Ankette katılımcılardan lisede öğrenim gördükleri Matematik, Fizik, Kimya, Biyoloji, Türkçe, Tarih, Coğrafya ve Yabancı Dil' den oluşan 8 derse ait başarı puanlarını 0-5 arasında puanlamaları istenmiştir. Çalışmamızda veri seti oluşturmak için lise notlarını kullanmamızın nedeni, üniversite bölümlerinin sayısal, sözel ve eşit ağırlık olarak gruplanması ve tercih aşamasında bu puanların dikkate alınmasıdır. Ayrıca katılımcılardan üniversitede hangi bölümden mezun oldukları ve mezuniyet puanları istenmektedir. Toplamda 250 katılımcıya erişilmiş elde edilen veriler, veri seti oluşturacak şekilde düzenlenerek test ve eğitim kümesi olarak gruplanmıştır. Eğitim kümesi kullanılarak sistem eğitilmiş ve test kümesi kullanılarak da sistemin öğrenme düzeyi test edilmiştir.

3.3 Veri Ön İşleme

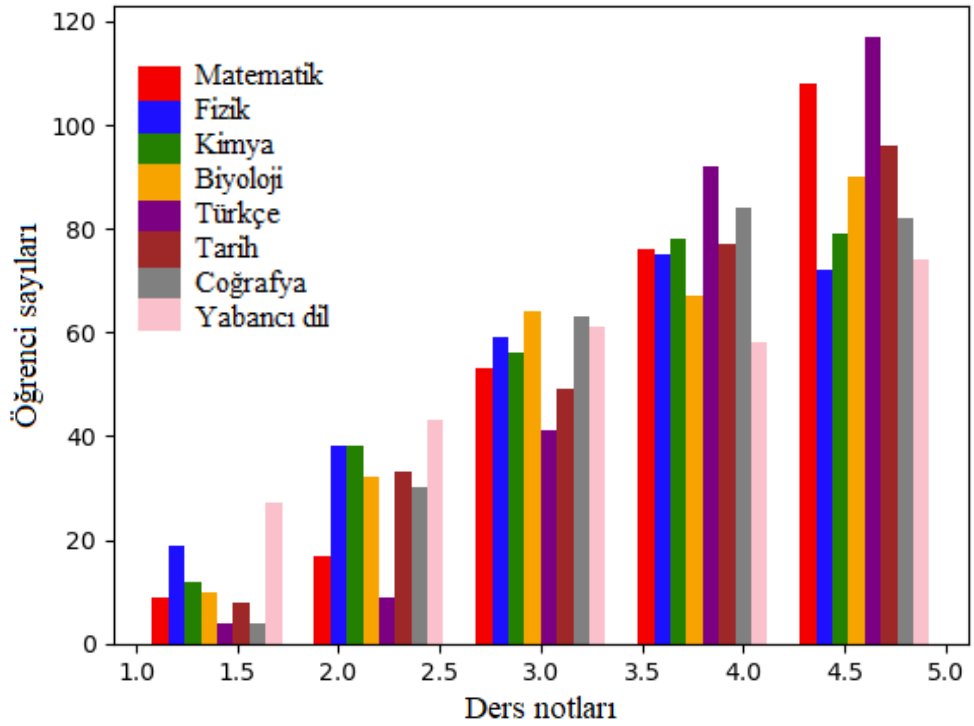
Bu bölümde analize hazır hale getirilmiş veri setleri oluşturmak amacıyla aşağıdaki işlemler uygulanmıştır.

3.3.1 Verinin Betimsel Analizi

	matematik	fizik	kimya	biyoloji	türkçe	coğrafya	tarih	yabancı
matematik	1.00	0.73	0.71	0.45	0.14	0.02	-0.00	0.21
fizik	0.73	1.00	0.68	0.43	0.05	-0.00	-0.01	0.17
kimya	0.71	0.68	1.00	0.66	0.19	0.10	0.11	0.33
biyoloji	0.45	0.43	0.66	1.00	0.26	0.13	0.15	0.37
türkçe	0.14	0.05	0.19	0.26	1.00	0.52	0.39	0.37
coğrafya	0.02	-0.00	0.10	0.13	0.52	1.00	0.63	0.29
tarih	-0.00	-0.01	0.11	0.15	0.39	0.63	1.00	0.28
yabancı	0.21	0.17	0.33	0.37	0.37	0.29	0.28	1.00

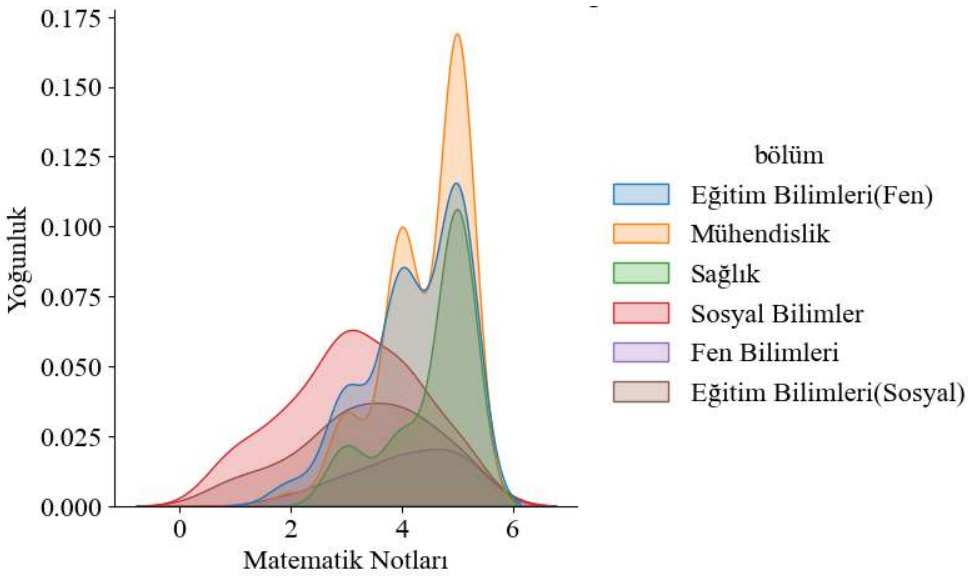
Şekil 3.1 Lise ders notları korelasyon matrisi.

Şekil 3.1'de bir pandas DataFrame'inin korelasyon matrisi hesaplanarak html tablosu şeklinde gösterilmiştir. İlk olarak DataFrame'in sütunları arasındaki Pearson korelasyon katsayıları hesaplanmış ve sonuç olarak bir korelasyon matrisi oluşturularak 2 ondalık basamağa kadar gösterilmiştir. Bu matris, her bir sütunun diğer sütunlarla olan korelasyonunu gösterir ve genellikle veri analizinde kullanılan önemli bir araçtır. Grafikten de anlaşılacağı gibi Matematik, Fizik ve Kimya derslerinin birbirleriyle olan korelasyonları yüksektir. Ayrıca Kimya dersinin Biyoloji dersiyle de olan korelasyonu yüksektir. Son olarak da Tarih ve Coğrafya derslerinin birbirleriyle olan korelasyonu yüksek görünmektedir.



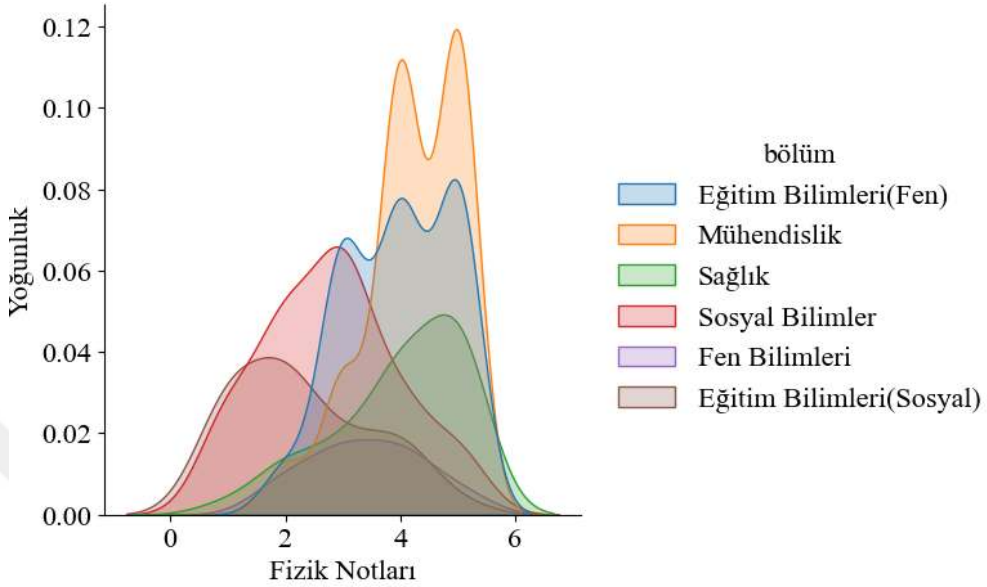
Şekil 3.2 Ders notları dağılımı.

Şekil 3.2' de gösterilen grafikte katılımcıların lise dönemi derslerindeki performansları gösterilmiştir. Grafikten de anlaşılacağı gibi katılımcıların büyük çoğunluğunun Matematik ve Türkçe puanları yüksektir. Coğrafya ve Türkçe notları düşük olan öğrenci sayısı çok azdır. Benzer şekilde diğer derslerde başarılı öğrenci sayısı grafikten anlaşılmaktadır.



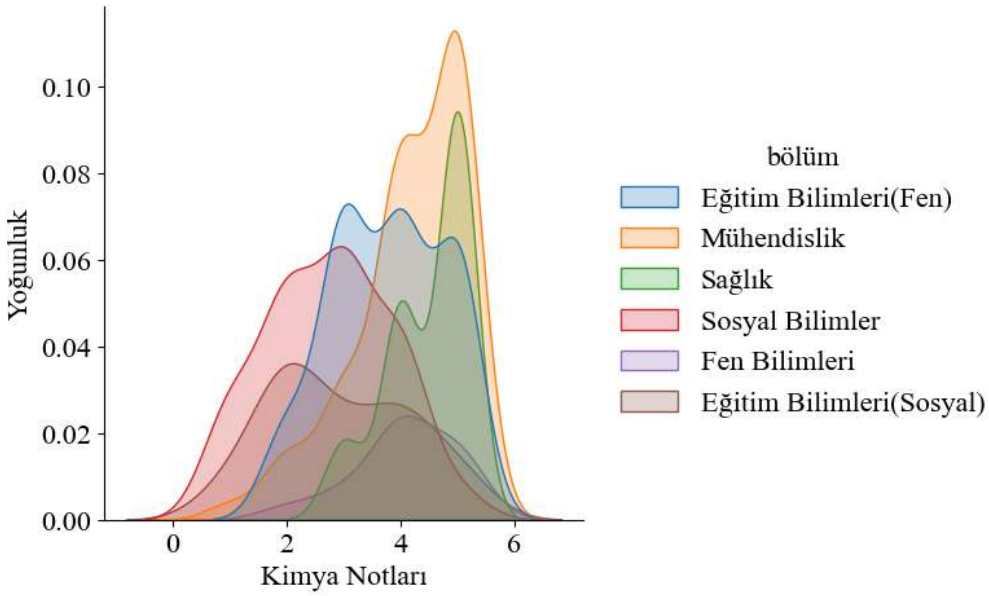
Şekil 3.3 Matematik notlarının bölümlere göre dağılımı.

Şekil 3.3‘ deki grafikte Matematik notlarının bölümlere göre dağılımı grafiği gösterilmektedir. Grafikten de anlaşılacağı gibi lise döneminde Matematik dersi başarısı yüksek olan katılımcıların büyük bölümü Mühendislik, Sağlık ve Eğitim bilimleri(fen) bölümlerinden başarıyla mezun olmuştur.

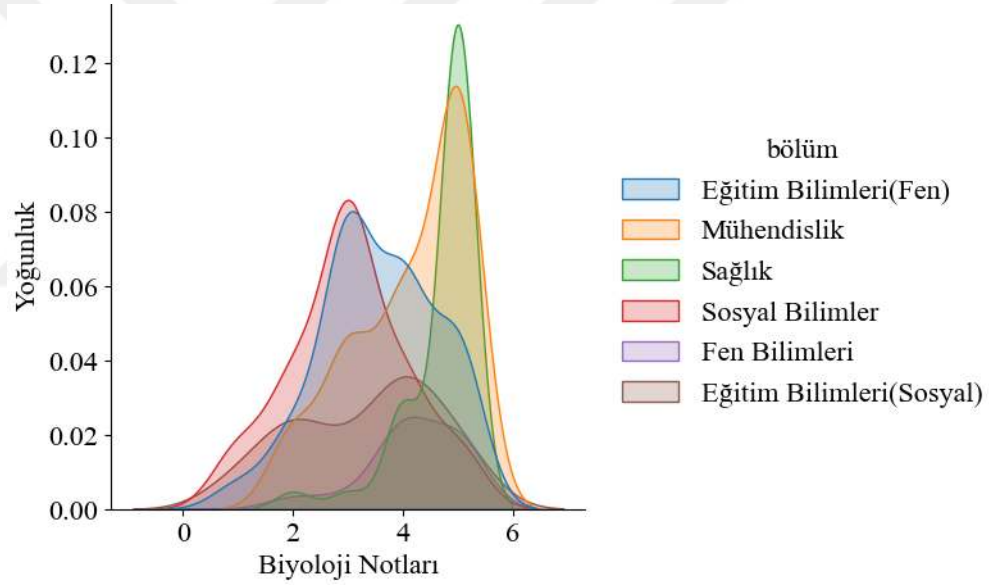


Şekil 3.4 Fizik notlarının bölümlere göre dağılımı.

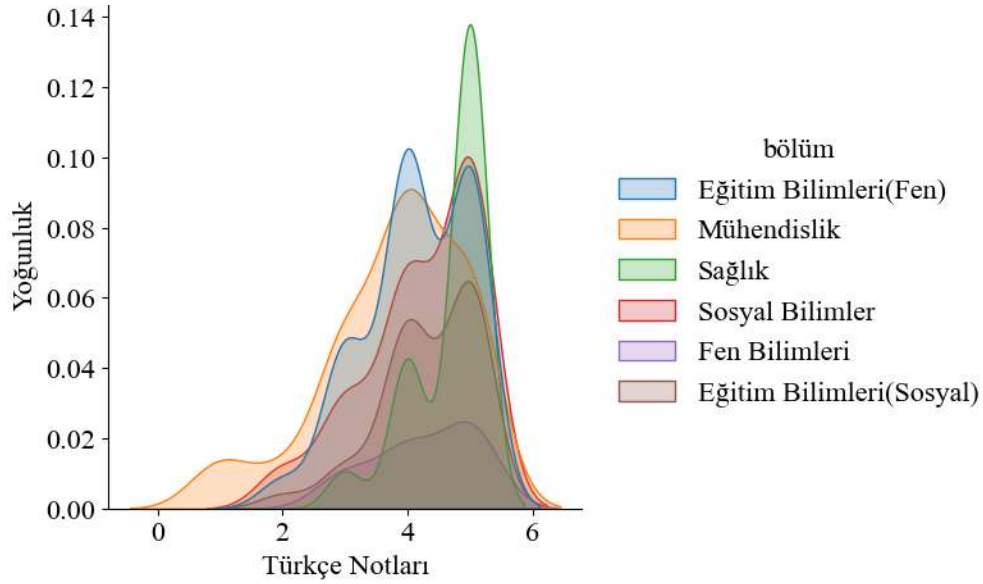
Şekil 3.4’ deki grafikte Lise dönemi Fizik dersi notlarının bölümlere göre dağılımı grafiği gösterilmiştir. Grafiğe göre lise döneminde Fizik dersi başarısı yüksek olan katılımcıların Mühendislik, Eğitim bilimleri (Fen) ve Sosyal bilimler bölümlerinden başarıyla mezun oldukları görülmektedir. Ayrıca Şekil 3.5, şekil 3.6, şekil 3.7, şekil 3.8, şekil 3.9 ve şekil 3.10’ da diğer ders notlarının bölümlere göre dağılım grafiği gösterilmiştir.



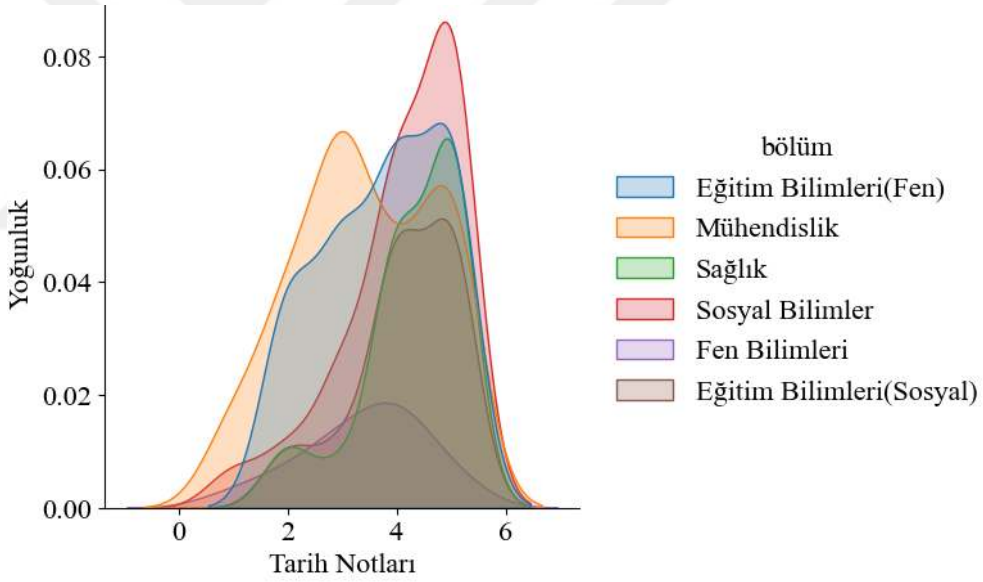
Şekil 3.5 Kimya notlarının bölümlere göre dağılımı.



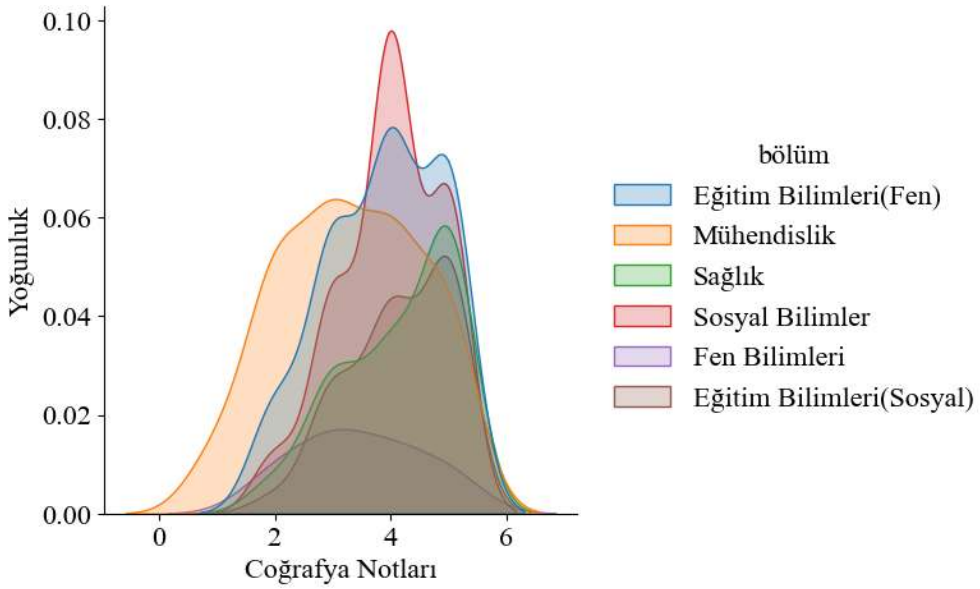
Şekil 3.6 Biyoloji notlarının bölümlere göre dağılımı.



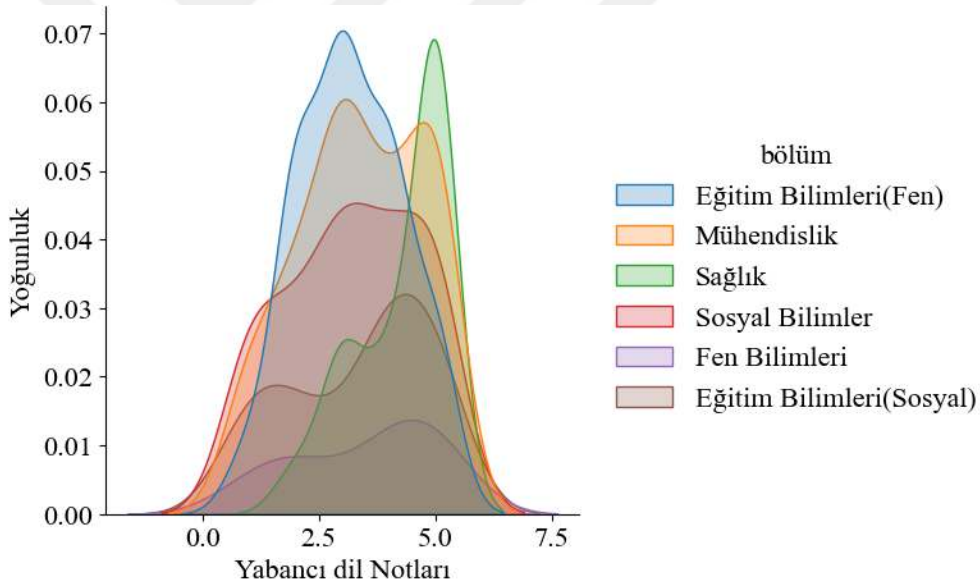
Şekil 3.7 Türkçe notlarının bölümlere göre dağılımı.



Şekil 3.8 Tarih notlarının bölümlere göre dağılımı.



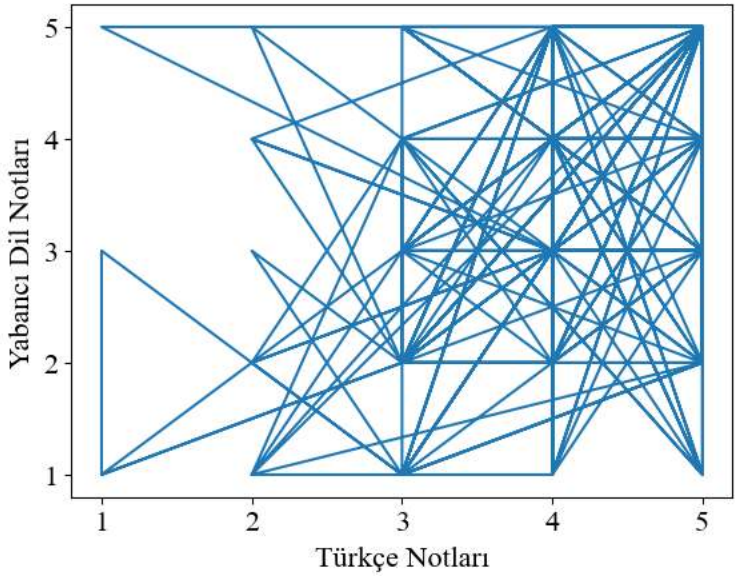
Şekil 3.9 Coğrafya notlarının bölümlere göre dağılımı.



Şekil 3.10 Yabancı Dil notlarının bölümlere göre dağılımı.

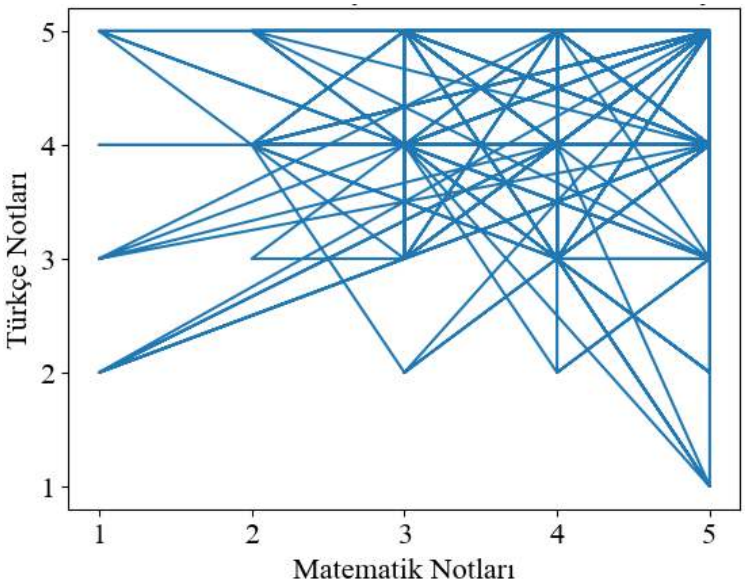
Makine öğrenmesinde veriler arasındaki ilişki, veri analizi ve model oluşturma süreçlerinde oldukça önemlidir. Veriler arasındaki ilişkiyi anlamak, veri setindeki değişkenlerin birbirleriyle nasıl etkileşimde olduğunu ve hangi değişkenlerin hedef değişkeni en iyi açıklayabileceğini belirlemeye yardımcı olur. Veriler arasındaki ilişkiyi anlamak, model oluşturma sürecinde de oldukça önemlidir. Modelde kullanılan değişkenler arasındaki ilişki doğru bir şekilde anlaşılmazsa, model yanıltıcı sonuçlar

üretebilir veya gereksiz değişkenlerin modelde kullanılması nedeniyle performans düşebilir. Veriler arasındaki ilişkinin ölçülmesi, korelasyon analizi gibi istatistiksel yöntemlerle yapılır. Korelasyon analizi, iki değişken arasındaki ilişkiyi ölçmek için kullanılan bir yöntemdir. Korelasyon katsayısı, iki değişken arasındaki doğrusal ilişkiyi ölçen bir ölçüttür.



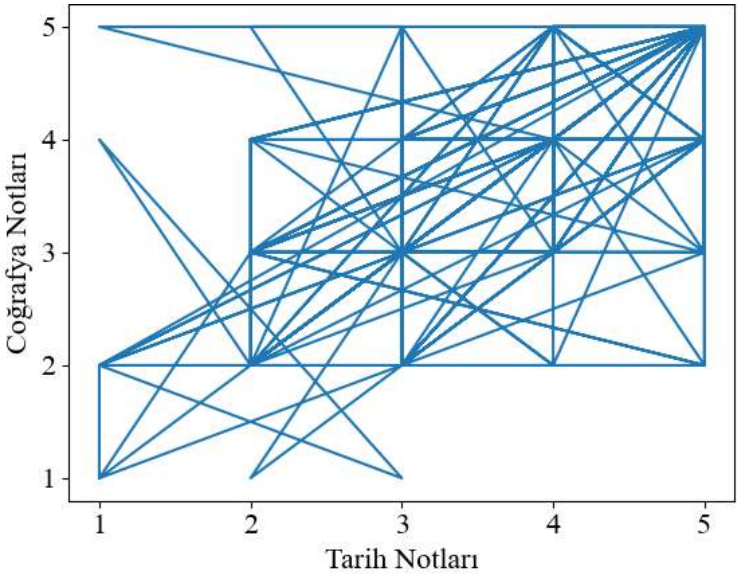
Şekil 3. 11 Türkçe ve Yabancı dil notlarının aralarındaki ilişki.

Şekil 3.11‘ deki grafikte Lise dönemi Türkçe ve Yabancı Dil notlarının aralarındaki ilişki gösterilmektedir. Grafiğe göre Yabancı dil puanı yüksek katılımcıların Türkçe dersi puanının da yüksek olduğu görülmektedir.

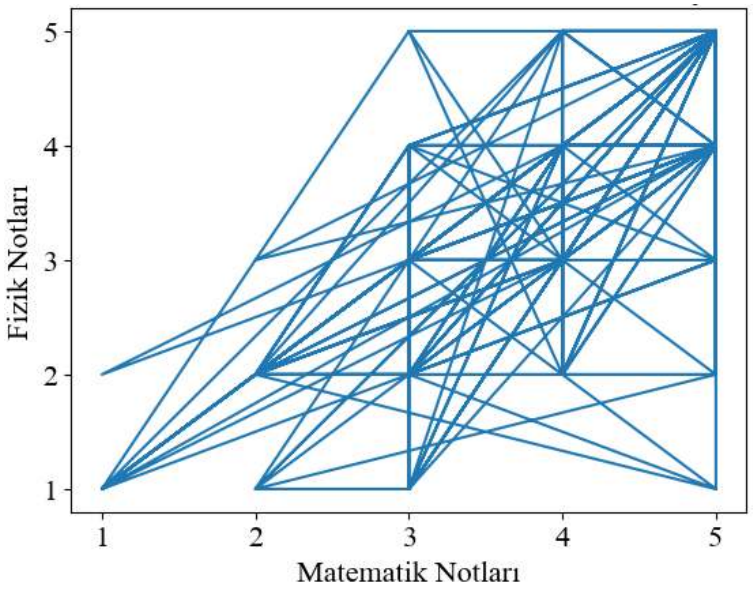


Şekil 3. 12 Matematik ve Türkçe notlarının aralarındaki ilişki.

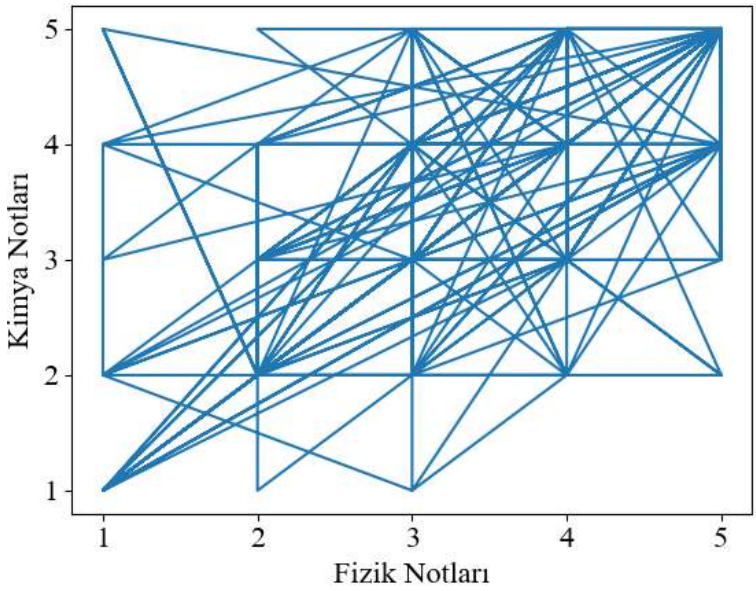
Şekil 3.12’ deki grafikte lise dönemi Matematik ve Türkçe dersi notlarının birbirleriyle olan ilişkisi gösterilmektedir. Bu grafiğe göre Matematik notu yüksek olan katılımcıların Türkçe notlarının da yüksek olduğu görülmektedir. Benzer şekilde şekil 3.13’ de Coğrafya ve Tarih derslerinin, şekil 3.14’ de Fizik ve Matematik derslerinin, şekil 3.15’ de Kimya ve Fizik derslerinin ve şekil 3.16’ da Kimya ve Biyoloji derslerinin aralarındaki ilişki gösterilmiştir.



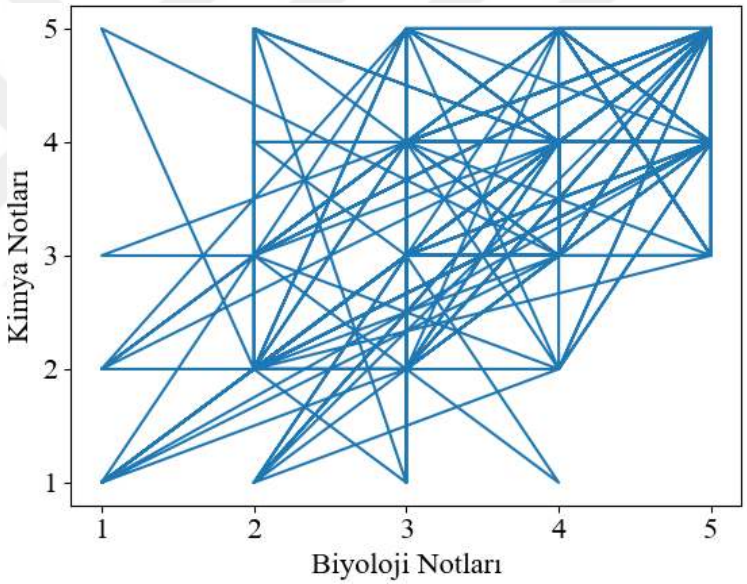
Şekil 3. 13 Coğrafya ve Tarih notlarının aralarındaki ilişki.



Şekil 3. 14 Fizik ve Matematik notlarının aralarındaki ilişki.



Şekil 3. 15 Kimya ve Fizik notlarının aralarındaki ilişki.



Şekil 3. 16 Kimya ve Biyoloji notlarının aralarındaki ilişki.

3.3.2 Veri Setinin Temizlenmesi

Veri analizi süreci, araştırmacıların elde ettikleri verilerin doğru bir şekilde işlenmesi ve yorumlanması için önemli bir aşamadır. Bu süreçte, verilerin doğru bir şekilde temizlenmesi ve işlenmesi, analiz sonuçlarının güvenilirliğini artırmak için kritik önem taşır.

Google Forms; araştırmacıların, ankete katılanların verilerini kolayca toplamalarını sağlayan bir araçtır. Ankette toplanan veriler, Google Forms tarafından

excel formatında indirilebilir. Veri seti daha sonra incelenir ve gerektiğinde veriler düzenlenir. Google Forms tarafında hazırladığımız ankete gelen cevaplar yine Google Forms tarafından excel formatında indirilerek incelenmiş, sonrasında analiz sürecinde fayda sağlamayacak olmasından dolayı ankete katılanların birkaç soruyu cevaplamamış olmaları gibi nedenlerle oluşan eksik veriler veri setinden çıkarılmıştır. Ayrıca üniversite mezuniyet puanı 3,0'ın altında olan katılımcıların verileri diğer katılımcıların verilerini etkileyerek analiz sonuçlarını bozabilir. Bu nedenle, veri seti incelendiğinde bu katılımcıların verileri tespit edilerek, veri setinden çıkarılmıştır. Veri seti incelendiğinde aynı bölüm mezunu katılımcılardan, bozucu etki yapacak yüksek çarpıklıkta değer giren katılımcıların verileri de veri setinden çıkarılmıştır.

3.3.3 Eksik Verilerin Doldurulması ve Veri Çoğaltma işlemi

Eksik verilerin doldurulması işlemi, araştırma veya analiz çalışmalarında önemli bir adımdır. Çok sayıda eksik değer içeren katılımcıların verilerinin veri setinden çıkarılması, doğru ve güvenilir sonuçlar elde etmek için mantıklı bir yaklaşımdır. Bu tür katılımcıların verilerinin çıkarılması, eksik değerlerin yoğun olduğu durumlarda veri setinin güvenilirliğini artırır ve sonuçların yanıltıcı olmasını önler. Öte yandan, az sayıda eksik değer içeren katılımcıların verilerinin eksik değerlerin ortalama değerlerle doldurulması yoluyla veri setinde tutulması, eksik değerleri doldurma tekniği olarak tercih edilen bir yöntemdir. Bu yaklaşım, eksik değerlere sahip olan katılımcıların verilerini korumaya ve analizlerde kullanılabilir hale getirmeye yönelik bir çözümdür. Eksik değerlere sahip olan özelliklerin eksik olmayan değerlerinin ortalaması kullanılarak eksik değerler doldurulur ve veri seti eksikliklerden arındırılmış hale getirilir. Bu veri doldurma yöntemi, eksik değerlerin tahmin edilmesi veya daha karmaşık yöntemlerin kullanılmasına göre daha basit ve hızlı bir çözüm sunar. Ayrıca, eksik değerlerin doldurulmasıyla birlikte veri setinin boyutu ve bütünlüğü korunur.

Veri çoğaltma, makine öğrenimi ve derin öğrenme alanlarında yaygın olarak kullanılan bir tekniktir. Bu teknik, sınırlı miktardaki mevcut veriye dayalı olarak daha fazla eğitim örneği elde etmeyi amaçlar. Veri setinin boyutunu artırarak, modelin daha iyi genelleme yapabilmesi ve daha güvenilir sonuçlar üretebilmesi hedeflenir. Veri çoğaltmanın temel amacı, veri setindeki çeşitliliği artırmaktır. Bu, modelin farklı veri örnekleri üzerinde eğitim yaparak daha iyi bir anlayış geliştirmesini sağlar. Daha çeşitli veri örnekleri, modelin daha genel bir perspektif kazanmasına yardımcı olur ve aşırı uyum

(overfitting) riskini azaltır. Veri çoğaltma için farklı yöntemler mevcuttur. Bunlardan biri, veri noktalarını doğrudan kopyalamaktır. Bu yöntem, mevcut veri noktalarının aynı haliyle çoğaltılmasını sağlar. Böylece veri seti boyutu artar, ancak orijinal veri dağılımı korunur. Diğer bir yöntem ise gürültü eklemektir. Bu yöntemde, mevcut veri noktalarına rastgele değişiklikler uygulanır. Bu sayede yeni veri noktaları elde edilir ve modelin daha fazla çeşitlilikle eğitilmesi sağlanır. Sentetik veri üretimi de veri çoğaltma için kullanılan bir yöntemdir. Bu, sınıf dengesizliği olan veri setlerinde kullanılarak daha dengeli bir eğitim seti elde edilmesine yardımcı olur. Dönüşümler ve değişiklikler de veri çoğaltmada etkili yöntemler arasındadır. Bu yöntemde, mevcut veri noktaları dönüştürülerek veya değiştirilerek yeni veri noktaları elde edilir [25]. Bu şekilde, daha fazla çeşitlilik sağlanır ve modelin daha geniş bir veri yelpazesine adapte olması sağlanır. Veri çoğaltmanın; seçilecek yöntemi, veri setinin özelliklerine, dağılımına ve hedeflenen probleme bağlıdır. Her yöntemin avantajları ve dezavantajları bulunur ve deneyler yoluyla en iyi yöntemler belirlenir. Çalışmamızda veriler, orijinal veri dağılımını etkilememek, aynı zamanda makine öğrenmesi yöntemlerinin uygun çalışabileceği veri değerine ulaşabilmek için veriler kopyalama yöntemi kullanılarak artırılmıştır.

3.3.4 Kategorik Verilerin Dönüştürülmesi

Veri setinde bazı özelliklerin metin veya kategorik formatta olması, bu özelliklerin makine öğrenimi algoritmaları tarafından doğrudan kullanılamamasına neden olur. Bu nedenle, bu özelliklerin uygun bir sayısal formata dönüştürülmesi gerekmektedir. Ancak, çalışmamızda kullanılan bazı veriler (örneğin, ders başarıları ve üniversite mezuniyet puanı gibi), zaten sayısal formatta oldukları için dönüştürme işlemine gerek yoktur. Bu tür veriler doğrudan makine öğrenimi algoritmalarında beslenebilir ve analiz sürecinde kullanılabilirler.

3.4 Modellerin Eğitilmesi ve Test Edilmesi

Makine öğrenmesi modellerinin eğitilmesinde, veri seti olarak bölüm 3.2’ de açıklanan kendi oluşturduğumuz anketten elde edilen veri seti kullanılmıştır. Bu sayede belirli bir bölgeden ve kurumdan değil ülke genelinden veriler elde edilerek, uygulama sonucunun ülkenin tamamına genelleştirilebilmesi sağlanmıştır.

"K-fold çapraz doğrulama" yöntemi, eğitim sürecinde veri setinin k-farklı parçalara bölünmesi ve k-1 parçanın eğitim, k. parçanın test için kullanılması şeklinde

tekrarlanan bir işlemdir. Bu yöntem sayesinde modelin genelleştirilebilirliği arttırılır ve overfitting (aşırı öğrenme) gibi problemlerin önüne geçilir.

GridSearchCV sınıfı, farklı parametre kombinasyonlarına karşı en iyi sonucu veren parametreleri bulmak için kullanılan bir hiperparametre optimizasyon tekniğidir. Bu yöntem, model performansını artırmak ve overfitting gibi problemlerin önüne geçmek için kullanılır. GridSearchCV yöntemi, tüm parametre kombinasyonlarını deneyerek en iyi sonucu veren parametreleri belirler ve bu parametreler kullanılarak model yeniden eğitilir. Bu sayede, en iyi öğrenme performansı elde edilir.

Doğru model seçimi ve modelin en iyi performansını sağlamak, makine öğrenimi uygulamalarında önemlidir. Bu nedenle, birçok farklı modelin performansı test edilerek en iyi sonucun elde edilmesi hedeflenmiştir. Testler sonrasında, eğitim işlemi "K-fold" çapraz doğrulama yöntemiyle tüm veri seti üzerinde katlamalı olarak gerçekleştirilmiştir. Bu yöntem, veri setini k-katmanlı bir şekilde böler, k-1 katmanını eğitim için ve 1 katmanını test için kullanarak modelin doğruluğunu test eder. Bu sayede modelin overfitting veya underfitting gibi problemlerinin önüne geçilir. Bu nedenle, K-fold çapraz doğrulama yöntemi kullanılarak modelin doğruluğu kendi veri setinde değerlendirilir. Ayrıca, öğrenme oranını artırmak amacıyla en uygun parametrelerin belirlenmesi için farklı parametrelerle eğitim ve test işlemi tekrarlanmıştır. Bu amaçla, 'GridSearchCV' sınıfı kullanılmıştır. GridSearchCV yöntemi kullanılarak en iyi parametrelerin belirlenmesi hedeflenmiştir. Bu yöntem, belirtilen parametrelerin farklı değerleri ile modellerin eğitilmesi ve en iyi performansı veren parametrelerin belirlenmesi işlemidir. Bu sayede modelin performansı artırılabilir ve en uygun parametrelerle eğitilerek daha iyi sonuçlar elde edilebilir. Çalışmamızda veriler %70 eğitim ve %30 test kümesine ayrılmış ve XGBoost Classifier, Support Vector Classification, KNN, Gaussian Naive Bayes modellerinde eğitilmiş ve test edilmiştir. Öğrenme modellerinin eğitilmesi ve test edilmesine ait kod parçaları şekil 3.17, şekil 3.18, şekil 3.19, şekil 3.20' de verilmiştir.

Çalışmamızda kullandığımız ilk model Gaussian Naive Bayes modelidir. Bu model, Bayes teoremine dayanarak çalışır ve özellikleri (features) birbirinden bağımsız olarak ele alır. Bu nedenle, özellikler arasındaki korelasyonları dikkate almaz.

```
import pandas as pd
from sklearn.model_selection import train_test_split, cross_val_score,
GridSearchCV
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix, f1_score, precision_score, recall_score

veri = pd.read_excel('veriler.xlsx')
veri = veri.drop('Unnamed: 0', axis=1)
x = veri.drop('bölüm', axis=1)
y = veri.bölüm
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3,
                                                    random_state=110)
nb = GaussianNB(var_smoothing=1e-09)
nb.fit(x_train, y_train)
y_pred = nb.predict(x_test)
params = {'var_smoothing': [1e-09, 1e-08, 1e-07, 1e-06, 1e-05, 1e-04,
                            1e-03, 1e-02, 1e-01, 1]}
grid = GridSearchCV(nb, params, cv=3)
grid.fit(x_train, y_train)

accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')
```

Şekil 3.17 Gaussian Naive Bayes modelinin eğitim ve test kod parçası.

Gaussian Naive Bayes sınıflandırıcı nesnesi "sklearn.naive_bayes" kütüphanesinin "GaussianNB" sınıfı kullanılarak oluşturulur. Model oluşturulurken, 'var_smoothing' parametresi kullanılır. Bu parametre, veri setindeki özelliklerin varyansını düzenleyen bir düzenleyici parametredir. Bu sayede, özelliklerin varyansı çok küçük olduğunda, modelin daha iyi performans göstermesi sağlanır. GridSearchCV yöntemi kullanılarak, 'var_smoothing' parametresi için farklı değerler denenmiş ve en iyi değer ile en iyi öğrenme oranı tespit edilmiştir. Çapraz doğrulama ile 4 katlı doğrulama yapılarak model eğitilmiştir. Bu verilere göre en iyi parametreler: 'var_smoothing': 0.1 şeklindedir. En iyi öğrenme başarısı ise: accuracy: 47.7, precision: 42.92, recall: 40.5, F1score: 39.63 olarak elde edilir. Bu sonuçlar, modelin sınıflandırma performansının ortalamanın altında olduğunu gösterir. Bunun nedeni, Gaussian Naive Bayes algoritmasının özellikleri birbirinden bağımsız olarak ele almasıdır. Bu nedenle, özellikler arasındaki korelasyonları dikkate almayan bu algoritma, bazı veri setlerinde düşük performans gösterebilir.

Çalışmamızda kullandığımız bir diğer makine öğrenmesi algoritması da Destek

vektör sınıflandırıcı (Support Vector Classifier - SVC) makine öğrenmesi algoritmasıdır. Bu algoritma, bir veri kümesindeki örnekleri sınıflandırmak için kullanılır. SVC, bir hiper-düzlem kullanarak verileri sınıflandırır ve bu hiper-düzlem, verilerin en iyi durumda ayrılmasını sağlayacak şekilde bulunur.

```
veri = pd.read_excel('Veriler.xlsx')
veri=veri.drop('Unnamed: 0',axis=1)
x = veri.drop('bölüm',axis=1)
y = veri.bölüm

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3,
 random_state=110)

svm = SVC(kernel='rbf', C=1, gamma='auto')
svm.fit(x_train, y_train)
y_pred = svm.predict(x_test)

basari = cross_val_score(estimator=svm, X=x_train, y=y_train, cv=4)
param_grid = [{'C': [0.05, 0.08, 0.1, 0.2, 0.3], 'kernel': ['linear', 'rbf'],
               'gamma': [0.5, 1, 1.5, 2]}]

gs = GridSearchCV(estimator=svm, param_grid=param_grid, scoring='accuracy',
                  cv=10, n_jobs=-1)
grid_search = gs.fit(x_train, y_train)


eniyi_sonuc = grid_search.best_score_
eniyi_parametreler = grid_search.best_params_
```

Şekil 3.18 SVC modelinin eğitim ve test kod parçası.

SVC nesnesi "sklearn.svm" kütüphanesinin "SVC" sınıfı kullanılarak oluşturulmuştur. Model oluşturulurken 'kernel', 'C', ve 'gamma' parametreleri kullanılmıştır. Kernel parametresi, SVC'nin veri noktalarını bir hiper-düzlemde nasıl ayırdığına karar verir. C parametresi, bir sınıflandırma hatasının ne kadar kabul edilebilir olduğunu belirler ve gamma parametresi ise hiper-düzlemin esnekliğini belirler. Model eğitimi, çapraz doğrulama yöntemi kullanılarak gerçekleştirilmiştir. Bu, veri kümesinin farklı parçalarını kullanarak modelin performansını değerlendirmek için yapılan bir yöntemdir. Çalışmamızda, 4 katlı doğrulama kullanılmıştır. En iyi parametreleri bulmak için GridSearchCV yöntemi kullanılarak farklı parametreler denenmiş ve en iyi parametreler ile en iyi öğrenme oranı tespit edilmiştir. GridSearchCV, bir hiperparametre arama yöntemidir ve aynı zamanda bir çapraz doğrulama yöntemi içerir. Bu yöntem, farklı parametrelerle bir modeli eğiterek, en iyi parametreleri seçer ve bu parametrelerle modelin performansını ölçer. Çalışmamızda en iyi parametreler: 'C': 0.3, 'gamma': 0.5,

'kernel': 'rbf' şeklindedir. Bu parametreler, en iyi öğrenme başarısını sağlayan parametrelerdir. En iyi öğrenme başarısı ise, modelinde accuracy: 60.22, precision: 59.89, recall: 57.59, F1score: 57.34 olarak elde edilmiştir.

Çalışmamızda SVC' den daha fazla başarı gösteren XGBoost algoritması, hızlı ve yüksek performanslıdır. Ayrıca büyük veri kümeleri üzerindeki sınıflandırma ve regresyon problemleri için kullanılabilir.

```
veriler = pd.read_excel(r"Veriler.xlsx")
veriler.drop("Unnamed: 0", axis=1, inplace=True)

veri = veriler[["matematik", "fizik", "kimya", "biyoloji", "türkçe",
                "coğrafya", "tarih", "yabancı", "bölüm"]]

X = veri.drop(["bölüm"], axis=1)
y = veri["bölüm"]

y.replace({
    "Eğitim Bilimleri(Fen)": 0,
    "Eğitim Bilimleri(Sosyal)": 1,
    "Fen Bilimleri": 2,
    "Mühendislik": 3,
    "Sağlık": 4,
    "Sosyal Bilimler": 5
}, inplace=True)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
                                                    random_state=42)

xgb_clf = xgb.XGBRFClassifier()
xgb_clf = xgb_clf.fit(X_train, y_train)

param_grid = {
    'max_depth': [3, 4, 5, 6],
    'learning_rate': [0.07, 0.1, 0.14],
    'n_estimators': [400, 450, 500],
    'subsample': [0.5, 0.6, 0.7],
    'colsample_bytree': [0.6, 0.8, 1.0],
}

grid_search = GridSearchCV(xgb_clf, param_grid=param_grid, cv=4, verbose=0,
                           n_jobs=-1)
grid_search.fit(X_train, y_train)
```

Şekil 3.19 XGBoost Classifier modelinin eğitim ve test kod parçası.

XGBoost algoritmasının sınıflandırma problemleri için özel olarak tasarlanmış bir sınıflandırıcı olan "XGBClassifier" sınıfı kullanılarak oluşturulmuştur. Model eğitimi GridSearchCV metoduyla yapılmış ve 3 katlamalı çapraz doğrulama (3-fold cross-validation) yöntemi kullanılmıştır. Modelin eğitiminde kullanılan en iyi parametreler,

GridSearchCV yöntemiyle bulunmuştur. Bu parametreler arasında, karar ağacı oluşturulurken kullanılacak özelliklerin oranını ifade eden "colsample_bytree": 0.8, modelin her adımda öğrenmesinin ne kadar ağır olacağını belirleyen "learning_rate": 0.07, karar ağacının maksimum derinliğini belirleyen "max_depth": 6, oluşturulacak karar ağaçlarının sayısını belirleyen "n_estimators": 400 ve her ağaç için kullanılacak örneklerin oranını ifade eden "subsample": 0.7 bulunmaktadır. Sonuç olarak, bu XGBoost sınıflandırıcısı nesnesi ile elde edilen öğrenme modeli, verilen veri seti üzerinde oldukça iyi performans göstermiştir. Modelin doğruluk değeri (accuracy) %82.78, hassasiyet değeri (precision) %69.9, geri çağırma değeri (recall) %62.68 ve F1 skoru (F1score) ise %63.83 olarak ölçülmüştür.

Çalışmamızda KNN algoritması en iyi performansı gösteren makine öğrenmesi algoritması olarak belirlenmiştir. KNN algoritması, veri setindeki her bir örneğin diğer tüm örneklerle olan uzaklıklarını hesaplayarak, k-en yakın komşuların etiketlerine bakarak yeni bir örneğin tahminini yapmaktadır.

```
veri = pd.read_excel('Veriler.xlsx')
veri = veri.drop('Unnamed: 0', axis=1)

x = veri.drop('bölüm', axis=1)
y = veri.bölüm

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3,
                                                    random_state=150)

knn = KNeighborsClassifier()

param_grid = {'n_neighbors': [7, 8, 9, 10], 'weights': ['uniform', 'distance'],
              'p': [1, 2, 3]}

grid_search = GridSearchCV(knn, param_grid, cv=3)
grid_search.fit(x_train, y_train)

best_params = grid_search.best_params_
best_knn = KNeighborsClassifier(**best_params)
best_knn.fit(x_train, y_train)
y_pred = best_knn.predict(x_test)
```

Şekil 3.20 KNN modelinin eğitim ve test kod parçası.

K-NN (K en Yakın Komşu) sınıflandırıcısı, Scikit-learn kütüphanesi tarafından sağlanan 'KNeighborsClassifier' sınıfı kullanılarak oluşturulur. Bu sınıf, K-NN algoritmasını uygulayan bir sınıflandırıcıdır. Model oluşturulurken, 'minkowski' parametresi ve bu parametrenin kuvvet değerini belirten 'p' değeri 1 olarak belirlenmiştir.

Ayrıca, kullanılacak olan komşu sayısını belirten 'n_neighbors' parametresi 7 olarak ayarlanmıştır. Her bir yakın komşuya verilen ağırlıklandırma türünü belirten 'weights' parametresine 'distance' değeri atanmıştır. Bu sayede, her bir komşuya veri noktası ve komşu arasındaki manhattan mesafe metriğine göre ağırlık verilir. KNN öğrenme modeli, verilerin sınıflandırılması için kullanılmıştır. Modelin performansını ölçmek için kullanılan ölçütler şunlardır: accuracy (doğruluk): 93.03, precision (kesinlik): 93.81, recall (duyarlılık): 93.03, F1score: 93.12.

3.4.1 Kullanılacak Modelin Seçilmesi

Veri seti üzerinde farklı sınıflandırma modelleri kullanılarak eğitim ve test işlemleri gerçekleştirildikten sonra elde edilen öğrenme başarıları incelenmiş, bu öğrenme başarıları sınıflandırıcıların farklı metrikler üzerinden performanslarının karşılaştırılmasına imkân sağlamıştır. Bu metrikler arasında accuracy, precision, recall ve F1 score gibi metrikler yer alır.

Çizelge 3.1 Öğrenme modelleri başarı karşılaştırması.

Öğrenme modeli	**Accuracy**	**Precision**	**Recall**	**F1 score**	**Kullanılan parametreler**
KNN	93.03	93.81	93.03	93.12	n_neighbors=7, p=1, weights=distance
XGBoost Classifier	82.78	69.9	62.68	63.83	colsample_bytree=0.8, learning_rate= 0.07, max_depth=6, n_estimators=400, subsample=0.7
SVC	60.22	59.89	57.59	57.34	C=0.3, gamma=0.5, kernel=rbf
Gaussian Naive Bayes	47.70	42.92	40.50	39.63	var_smooting=0.1

KNN (K-En Yakın Komşu) algoritması diğer öğrenme modellerine göre bazı avantajlara sahiptir. KNN basit ve anlaşılması kolay bir algoritmadır, bu nedenle kullanımı kolaydır ve hızlı bir şekilde uygulanabilir. Ayrıca, KNN' nin temelsiz (non-parametrik) bir yaklaşımı vardır, bu da veri setinin dağılımı hakkında herhangi bir varsayımda bulunmadığı anlamına gelir. Bu, karmaşık veya belirli bir dağılım yapısına sahip veri setleriyle de başa çıkabilme yeteneği sağlar. KNN' nin bir diğer üstünlüğü iyi genelleme yeteneğidir. Yakın komşuların etiketlerine dayanarak sınıflandırma yapması, veri setindeki lokal yapıyı daha iyi yakalar. Bu sayede KNN algoritması çalışmamızda yüksek başarıyı elde etmiştir. Ancak KNN büyük veri setlerinde ve yüksek boyutlu verilerde zaman ve hafıza açısından maliyetli olabilir.

Çizelge 3.1’ de öğrenme modellerinin Accuracy, Precision, Recall ve F1 score başarı kriterleri bakımından karşılaştırma tablosu verilmiştir. Bu karşılaştırmalar sonucunda, veri seti üzerinde KNN sınıflandırıcısının en yüksek başarıyı elde ettiği tespit edilmiştir. Bu nedenle, KNN sınıflandırıcısının kullanılmasına karar verilmiştir.

3.5 Algoritmanın Oluşturulması ve Sistemin Test Edilmesi

Şekil 3.21’ de sistemi eğitmek ve başarılarını test etmek için kullanılan algoritmalara yer verilmiştir.

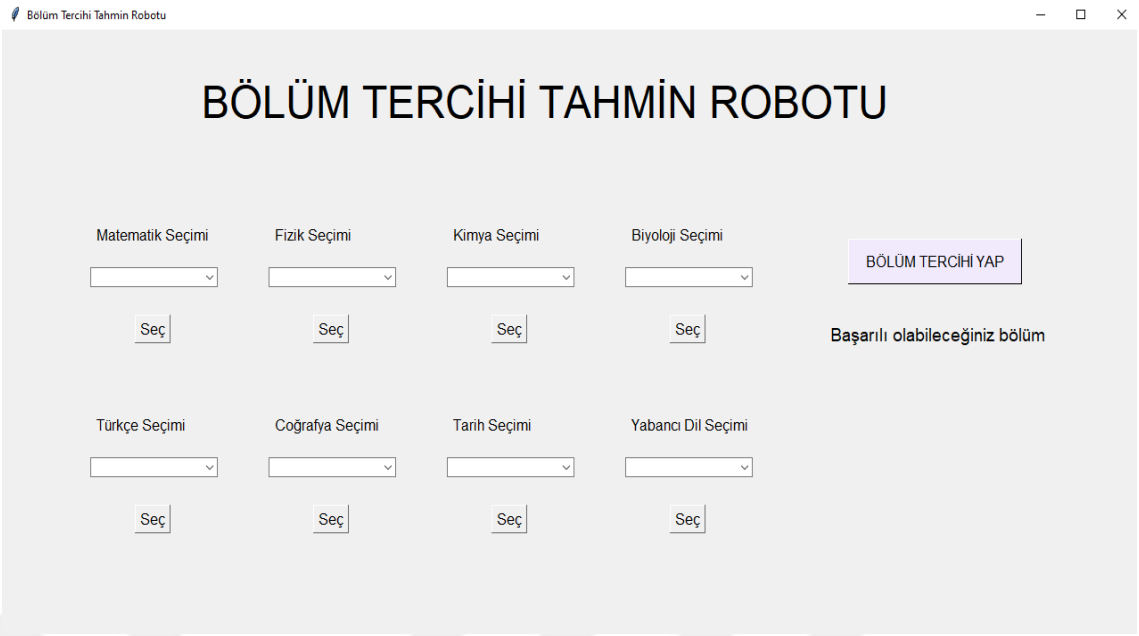
Bölüm 3.4' te algoritmanın oluşturulmasına öncelikle gerekli kütüphanelerin ve nesnelerin yüklenmesiyle başlanmıştır. Ardından veri seti okunarak gereksiz sütunlar atılmıştır. Veri setindeki bağımsız değişkenler “x” ve bağımlı değişken “y” olarak ayrılmıştır. Daha sonra, sınıflar etiketlenerek KNN nesnesi kullanılarak model elde edilen en iyi parametrelerle eğitilmiştir. Eğitim işlemi tamamlandıktan sonra, oluşturulan modelin doğruluğu test edilmiştir. Test sonuçları, başarı değerleri olan doğruluk, hassasiyet, duyarlılık ve F1 puanı olarak yazdırılmıştır. Böylece, en iyi parametrelerle eğitilmiş ve test edilmiş bir KNN nesnesi kullanılarak, veri setindeki verilerin regresyon işlemi tamamlanmıştır.

```
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score, precision_score,
recall_score, f1_score

veri = pd.read_excel('Veriler.xlsx')
veri = veri.drop('Unnamed: 0', axis=1)

x = veri.drop('bölüm', axis=1)
y = veri.bölüm

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3,
                                                    random_state=150)

knn = KNeighborsClassifier()

param_grid = {'n_neighbors': [7, 8, 9, 10], 'weights': ['uniform', 'distance'],
              'p': [1, 2, 3]}

grid_search = GridSearchCV(knn, param_grid, cv=3)
grid_search.fit(x_train, y_train)

best_params = grid_search.best_params_
best_knn = KNeighborsClassifier(**best_params)
best_knn.fit(x_train, y_train)
y_pred = best_knn.predict(x_test)

accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')
print("En iyi başarı değeri (accuracy):", accuracy)
print("En iyi parametreler:", best_params)
print("Model precision:", precision)
print("Model recall:", recall)
print("Model F1 score:", f1)
```

Şekil 3.21 Sistemde kullanılan algoritmalar.

3.6 Sistem Arayüzü Tasarlanması

Arayüz tasarımı, kullanıcıların programı kolayca kullanabilmesi için görsel bir arabirim sağlar. Bu sayede kullanıcılar programı daha etkili bir şekilde kullanabilirler. Arayüz tasarımı, kullanıcılara veri girişi yapmalarına, programı çalıştırmalarına ve sonuçları görsel olarak görüntülemelerine olanak tanır. Ayrıca, programın '.exe' formatına dönüştürülmesi, kullanıcılara programı kolayca yükleyip çalıştırmalarını sağlar. Bu sayede kullanıcılar programın kurulumu veya ayarlarıyla ilgili herhangi bir sorunla karşılaşmadan programı kullanabilirler. Bu şekilde tasarlanmış ve dönüştürülmüş bir sistem, öğrencilerin kolayca kullanabileceği, verimli ve kullanıcı dostu bir ara yüz sunar.



Şekil 3.22 Sistem arayüzü.

Şekil 3.22' de sistem arayüzü için tasarlanan görsel verilmiştir. Bu ara yüzde öğrenciler ders başarı oranlarını seçtikten sonra bölüm tercihi yap butonuna tıkladıklarında, sistem KNN algoritmasını kullanarak öğrenciye uygun bir bölüm tahmini sunacaktır.

Çizelge 3.2' de bölüm 3.2' de verilen çalışmalarla karşılaştırması yapılmıştır. Bu çalışmalarda başarı ölçütleri birbirinden farklıdır. Bu farklı başarı ölçütlerine göre karşılaştırma yapıldığında, çalışmamızın üstünlüğü görülmektedir. Ayrıca diğer tüm çalışmaların ortak eksik noktası, veri setinin tek bir kurum veya bir bölgeden elde edilen verilerle hazırlanmış olmasıdır. Çalışmamızda bu eksikliği gidermek için tüm Türkiye'den veriler toplanarak veri seti oluşturulmuştur. Bu sayede çalışma bir bölgeyle veya kurumla sınırlı kalmamış, tüm ülkeye uygulanabilecek şekilde oluşturulmuştur.

Çizelge 3.2 Çalışmanın diğer çalışmalarla karşılaştırmalı analizi

Çalışmalar	**Başarı metrikleri**
Comparatıve Study Of Machıne Learnıng Knn, Svm, And Decısıon Tree Algorıthm To Predıct Student's Performance [9].	Accucary:93 F1 Score: 82
Machine Learning Based Predicting Student Academic Success [10].	Accucary:88 Precision: 97
Predicting Students' Performance Using Machine Learning Techniques [11].	Accucary:77.04 F1 Score: 78.47
Predıctıng Academıc Performance Of Engıneerıng Students Usıng Ensemble Method [12].	Accucary:82 F1 Score: 82
Modeling and Predicting Students' Academic Performance Using Data Mining Techniques [13].	Accucary:86 Specificity: 86.3
Çalışmamız	Accucary:84.78 Precision: 82.38 Recall: 84.79 F1 Score: 83.26

.

4. SONUÇ

Bu çalışma, üniversite bölüm tercihlerinde öğrencilere yardımcı olmak amacıyla geliştirilen bir uygulama üzerine yapılmıştır. Uygulama, makine öğrenmesi algoritmaları kullanarak öğrencilerin tercih edebilecekleri bölümleri tahmin etmektedir. Çalışmada kullanılan veri seti, Türkiye'nin tüm bölgelerinden toplanarak genelleştirilebilir bir yapıda oluşturulmuştur. Veri seti, temizleme ve eksik verilerin doldurulması işlemleri ile öğrenme modellerinin daha tutarlı ve genelleştirilebilir hale getirilmesi sağlanmıştır.

Çalışmada kullanılan öğrenme modelleri arasında XGBoost, SVC, KNN ve Gaussian Naive Bayes yer almaktadır. Bu modeller arasında KNN algoritması, Accuracy, Precision, Recall ve F1score başarılarına göre en yüksek performansı sergilemiştir. K-fold çapraz doğrulama yöntemi kullanılarak öğrenme başarısı artırılmış ve GridsearchCV ile en uygun parametreler belirlenmiştir.

Bu çalışmanın sonuçlarına göre, geliştirilen uygulama öğrenciler için kullanılabilir bir hale getirilmiştir. Ancak, veri setinin katılımcı sayısı artırılarak öğrenme oranlarında yükselme ve genelleştirme gücünün artırılması mümkündür. Ayrıca, veri setine cinsiyet, okul türü, hobiler gibi değerlerin eklenmesiyle daha uygun sonuçlar elde edilebilir. Uygulama telefon uygulaması haline getirilerek öğrenciler için daha kolay ulaşılabilir hale getirilebilir.

Sonuç olarak, bu çalışma üniversite bölüm tercihlerinde öğrencilerin karşılaştığı zorlukları azaltmak amacıyla önemli bir adım olmuştur. Geliştirilen uygulama sayesinde öğrencilerin bölüm tercihleri konusunda daha bilinçli kararlar alması hedeflenmektedir.

5. KAYNAKLAR

[1]. New Scientist, *Düşünen Makineler*, Say Yayınevi, 2004.

[2]. "A brief history of machine learning. european journal of science and technology," https://www.dataversity.net/a-brief-history-of-machine-learning/ (erişim Nisan 20, 2023).

[3]. "Looking backwards, looking forwards: Sas, data mining, and machine learning,"https://blogs.sas.com/content/subconsciousmusings/2014/08/22/looking-backwards-looking-forwards-sas-data-mining-and-machine-learning/ (erişim Mayıs 5, 2023).

[4] F. Ersöz ve Y. Çınar, "Veri madenciliği ve makine öğrenimi yaklaşımlarının karşılaştırılması: Tekstil sektöründe bir uygulama", *Avrupa Bilim ve Teknoloji Dergisi*, sayı. 29, ss. 397-414, Ara. 2021, doi:10.31590/ejosat.1035124.

[5]. E. Filiz, "Makine öğrenmesi yöntemleri ve eğitim verisi üzerine bir uygulama: Uluslararası matematik ve fen eğilimleri araştırması 2015 Türkiye örneği," Doktora tezi, Fen Bil. Ens., Yıldız Teknik Üniv., İstanbul, Türkiye, 2019.

[6]. Ü. Şengül, H. Pamukçu, B. Zengin, "Bölüm tercihi ile kişilik arasındaki ilişki: Kastamonu turizm işletmeciliği ve otelcilik yüksekokulu örneği," *Kastamonu Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, vol. 7, issue 1, ss. 122-134. 2015.

[7]. M. Pekkaya ve N. Çolak "Üniversite öğrencilerinin meslek seçimini etkileyen faktörlerin önem derecelerinin ahp ile belirlenmesi," *The Journal of Academic Social Science Studies*, vol. 6, issue 2, pp. 797-818, 2013, doi:10.9761/jasss_643.

[8]. E. Tosunoğlu, R. Yılmaz, E. Özeren ve Z. Sağlam "Eğitimde makine öğrenmesi: Araştırmalardaki güncel eğilimler üzerine inceleme," *Ahmet Keleşoğlu Eğitim Fakültesi Dergisi*, vol. 3, issue 2, pp. 178-199, Sep. 2021, doi:10.38151/akef.2021.16.

[9] S. Wiyono ve T. Abidin "Comparative study of machine learning knn, svm, and decısıon tree algorithm to predict student's performance," *İnternational Journal*

of Research - Granthaalayah, cilt. 7, sayı. 1, 2019, doi: https://doi.org/10.29121/granthaalayah.v7.i1.2019.1048.

[10] K.A. Mayahi ve M. Al-Bahri "Machine learning based predicting student academic success," *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, 2020, pp. 264-268.

[11] H. Altabrawee, O. A. . Ali ve S. Q. Ajwi "Predicting students' performance using machine learning techniques," *Journal of University of Babylon, Pure and Applied Sciences*, cilt. 27, sayı. 1, pp. 194-205, 2019, doi:10.29196/jubpas.v27i1.2108.

[12] T. B. Bithari, S. Tahapa ve K.C. Hari "Predıctıng academıc performance of engıneerıng students usıng ensemble method," *A Peer Reviewed Technical Journal*, cilt. 2, sayı. 1, 2020, doi:10.3126/tj.v2i1.32845.

[13] A. Mueen, B. Zafar ve U. Manzoor "Modeling and predicting students' academic performance using data mining techniques," *I.J. Modern Education and Computer Science*, cilt. 8, sayı. 11, 2016, doi: 10.5815/ijmecs.2016.11.05.

[14]. E. Yanıkkerem, S. Altıparmak, G. Karadeniz "Gençlerin meslek seçimini etkileyen faktörler ve benlik saygıları," *Nursing Forum Derg.*, vol. 7, issue 2, ss. 61-62, 2004.

[15]. S. Kıyak "Genel lise öğrencilerinin meslek seçimi yaparken temel aldığı kriterler," Yüksek Lisans Tezi, Sosyal Bil. Ens., Yeditepe Üniv., İstanbul, Türkiye, 2006.

[16]. T. Sarıkaya ve L. Khorshid "Üniversite öğrencilerinin meslek seçimini etkileyen etmenlerin incelenmesi: Üniversite öğrencilerinin meslek seçimi," *Türk Eğitim Bilimleri Dergisi*, vol. 7, issue 2, ss. 393-423, 2009.

[17]. "Python leads the 11 top data science, machine learning platforms: Trends and analysis," https://www.kdnuggets.com/2019/05/poll-top-data-science-machine-learning-platforms.html (erişim Mayıs 20, 2023).

[18]. "Visual analytics and human ınvolvement in machine learning," https://arxiv.org/abs/2005.06057 (erişim Mayıs 10, 2023).

[19]. "The 7 steps of machine learning," https://towardsdatascience.com/the-7-steps-of-machine-learning-2877d7e5548e (erişim Mayıs 20, 2023).

[20]. "Makine öğrenmesi algoritmaları," https://azure.microsoft.com/tr-tr/resources/cloud-computing-dictionary/what-are-machine-learning-algorithms (erişim Mayıs 12, 2023).

[21]. C. Cortes ve V. Vapnik "Support vector network, machine learning," *Intelligent Information Management*, vol. 7, no. 3, ss. 273-297, May 2015, http://dx.doi.org/10.1007/BF00994018

[22]. H. Bhavsar ve A. Ganatra "A comparative study of training algorithms for supervised machine learning." *International Journal of Soft Computing and Engineering (IJSCE),* vol. 2, issue 4, pp. 78-79, 2012.

[23]. Chen T. ve Guestrin C. "Xgboost: A scalable tree boosting system," *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785-794, doi: https://doi.org/10.1145/2939672.2939785.

[24]. H. A. Nizam, "Sosyal medyada makine öğrenmesi ile duygu analizinde dengeli ve dengesiz veri setlerinin performanslarının karşılaştırılması," *XIX. Türkiye'de İnternet Konferansı*, İzmir, 2014, pp. 80-81.

[25]. C. Sharton, T.M. Khoshgoftaar ve B. Furht " Text data augmentation for deep learning" *Journal of Big Data*, vol. 8, issue 101, 2021, doi:10.1186/s40537-021-00492-0.

EKLER

EK-1

Makine Öğrenmesi algoritmaları kullanılarak üniversite bölüm tercihi tahmini

Bu formda kişisel bilgileriniz istenmemektedir. Paylaştığınız bilgiler yalnızca yapay zeka uygulamalarının eğitilmesinde veri seti olarak kullanılacak olup üçüncü şahıslarla kesinlikle paylaşılmayacaktır.

* Gerekli

1. Mesleğiniz. (Gelecekteki muhtemel mesleğiniz.) *

   Öğrenciler ve şu anda mesleği olmayanlar gelecekte muhtemel sahip olacakları mesleği yazabilirler.

2. Mesleğinizden memnuniyet durumunuz. *

   Yalnızca bir şıkkı işaretleyin.

   Hiç memnun değilim.

   1 ○

   2 ○

   3 ○

   4 ○

   5 ○

   Çok memnunum.

3. Gelir ve statü kaygısından bağımsız olarak yapmak istediğiniz meslek nedir? *

_____

4. Öğrenim düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

- ◯ Lise
- ◯ Önlisans
- ◯ Lisans
- ◯ Yüksek lisans

5. Okuduğunuz üniversitenin adı nedir?

_____

6. Üniversitede okuduğunuz bölüm adı nedir?

Bölümünüzün tam adını yazınız.

_____

7. Lisans mezuniyet başarı puanınız nedir? *

Seçeneklerde belirtilen puan aralıklarından uygun olanı seçiniz. Henüz mezun olmayanlar muhtemel mezuniyet puan aralığını girmelidir.

Yalnızca bir şıkkı işaretleyin.

- ◯ 1.0 - 1.5 aralığında
- ◯ 1.5 - 2.0 aralığında
- ◯ 2.0 - 2.5 aralığında
- ◯ 2.5 - 3.0 aralığında
- ◯ 3.0 - 3.5 aralığında
- ◯ 3.5 - 4.0 aralığında

8. Lise **matematik** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız				Pekiyi
	1	2	3	4	5

9. Lise **fizik** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız				Pekiyi
	1	2	3	4	5

10. Lise **kimya** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız				Pekiyi
	1	2	3	4	5

11. Lise **biyoloji** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız				Pekiyi
	1	2	3	4	5

12. Lise **türkçe** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız
1	◯
2	◯
3	◯
4	◯
5	◯
	Pekiyi

13. Lise **coğrafya** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız
1	◯
2	◯
3	◯
4	◯
5	◯
	Pekiyi

14. Lise **tarih** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız
1	◯
2	◯
3	◯
4	◯
5	◯
	Pekiyi

15. Lise **yabancı dil** dersi başarı düzeyiniz. *

Yalnızca bir şıkkı işaretleyin.

	Başarısız
1	◯
2	◯
3	◯
4	◯
5	◯
	Pekiyi

EK-2

Evrak Tarih ve Sayısı: 11.05.2023-E.91190

T.C.
BİTLİS EREN ÜNİVERSİTESİ REKTÖRLÜĞÜ
Etik İlkeleri ve Etik Kurulu

Konu : Etik Kurulu Kara

DAĞITIM YERLERİNE

İlgi :

İlgide kayıtlı dilekçeniz gereği, "Makine Öğrenmesi Algoritmaları Kullanılarak Üniversite Bölüm Tercihi Tahmini" adlı çalışmanız Üniversitemiz Etik İlkeleri ve Etik Kurulunu sayılı kararıyla uygun görülmüştür.

Bilgilerini ve gereğini rica ederim.

Doç. Dr. Yunus Levent EKİNCİ
Kurul Başkanı

Dağıtım:
Sayın Dr. Öğr. Üyesi Ayşe Saliha SUNAR
Sayın Doç. Dr. Rıdvan UMAZ
Sayın Harun ÇELİK

Bu belge, güvenli elektronik imza ile imzalanmıştır.

Bu belge, güvenli elektronik İmza ile imzalanmıştır.