

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

ÇEVİRİMİÇİ EĞİTİMDE UYUMLANABİLİRLİK
DÜZEYİNİ TAHMİN EDEN MAKİNE ÖĞRENMESİ
MODELİ

Yüksek Lisans Tezi

Tezi Hazırlayan
Yağmur GÜR

İstanbul, 2024

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

**ÇEVİRİMİÇİ EĞİTİMDE UYUMLANABİLİRLİK
DÜZEYİNİ TAHMİN EDEN MAKİNE ÖĞRENMESİ
MODELİ**

Yüksek Lisans Tezi

Tezi Hazırlayan
Yağmur GÜR

Öğrenci No
2120003028

ORCID ID
0009-0001-4524-5277

Danışman
Doç. Dr. Atınç YILMAZ

İstanbul, 2024

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum “Çevrimiçi Eğitimde Uyumlanabilirlik Düzeyini Tahmin Eden Makine Öğrenmesi Modeli” başlıklı bu çalışmanın, bilimsel ahlak ve geleneklere uygun şekilde tarafımdan yazıldığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, yararlandığım eserlerin tamamının kaynaklarda gösterildiğini ve çalışmamın içinde kullanıldıkları her yerde bunlara atıf yapıldığını, patent ve telif haklarını ihlal edici bir davranışımın olmadığını belirtir ve bunu onurumla doğrularım. 28/08/2024

Yağmur GÜR

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜ
TEZLİ YÜKSEK LİSANS SINAV TUTANAĞI

28.05.2024

Enstitümüz *Bilgisayar Mühendisliği* Anabilim Dalı *Bilgisayar Mühendisliği* Programı yüksek lisans öğrencilerinden 2120003028 numaralı *Yağmur GÜR*'ün "*İstanbul Beykent Üniversitesi Lisansüstü Eğitim – Öğretim Yönetmeliği*"nin ilgili maddesine göre hazırlayarak, Enstitümüze teslim ettiği "*Çevrimiçi Eğitimde Uyumlanabilirlik Düzeyini Tahmin Eden Makine Öğrenmesi Modeli*" konulu tezini, Yönetim Kurulumuzun 28/05/2024 tarih ve 2024/24 sayılı toplantısında seçilen ve On-Line toplanan biz jüri üyeleri huzurunda, İstanbul Beykent Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 29. maddesinin 3. fıkrası gereğince 45 dakika süre ile Zoom programı aracılığıyla on-line olarak aday tarafından savunulmuş ve sonuçta adayın tezi hakkında "*OYBİRLİĞİ*" ile "*KABUL*" kararı verilmiştir.

İşbu tutanak, 2 nüsha olarak hazırlanmış ve Enstitü Müdürlüğü'ne sunulmak üzere tarafımızdan düzenlenmiştir.

DANIŞMAN
Doç. Dr. At*** YI***
(İstanbul Beykent Üniversitesi)

ÜYE
Dr. Öğr. Üyesi Ze*** AL***
(İstanbul Beykent Üniversitesi)

ÜYE
Dr. Öğr. Üyesi So*** SE***
(Kütahya Dumlupınar Üniversitesi)

Adı ve Soyadı : Yağmur GÜR
Danışman : Doç. Dr. Atınç YILMAZ
Derecesi ve Tarihi : Yüksek Lisans (Tezli), 2024
Alanı : Bilgisayar Mühendisliği
Anahtar Kelimeler : Makine Öğrenmesi, Online Öğrenme
Uyumluluğu, Tahmin, Sınıflandırma, Doğruluk
Oranı

ÖZ

ÇEVİRİMİÇİ EĞİTİMDE UYUMLANABİLİRLİK DÜZEYİNİ TAHMİN EDEN MAKİNE ÖĞRENMESİ MODELİ

Günümüzde makine öğrenme modellemeleri pek çok farklı sektörde kullanılıyor. Makine öğrenmesi yöntemleri kullanılan alanlarda yüksek performanslar göstermiştir. Farklı sektörlerde kullanılan makine öğrenmesinin kullanımı teknolojinin hızla gelişmesiyle yaygınlaşmıştır. Farklı sektörlerde kullanılan yöntemler pek çok sorunu görüp çözüme kavuşturmuştur. Teknolojinin gelişimi ile Eğitim sektörü de bundan etkilenmiştir. Uzaktan öğrenme veya çevrim içi öğrenmenin kullanımı Korona dönemi ile artmıştır. Ancak öğrenme yöntemi olarak kullanılan çevrim içi öğrenmenin pek çok örneğinin başarılı olmadığı görülmektedir.

Bu çalışmada amaçlanan, var olan bu sorunun çevrim içi öğrenme yöntemi kullanılabilmesi için bir strateji sunarak örnek bir yöntem uygulamaktır. Böylece teknoloji alanında devam eden çalışmalara katkı sağlamak amaçlanmaktadır. Bu amaç doğrultusunda, 2000 satırlık demografik ve ekonomik özellikleri bulunan veriler incelenerek ön işleme ve görselleştirme işlemlerinden geçirilmiştir. Çalışmada, veri işlemleri basamakları gerçekleştirildikten sonra modeller kurularak öğrencinin çevrim içi öğrenmeye uyum düzeyi belirlenmiştir. Veri setindeki çevrim içi uyum düzeyi bağımlı değişken seçilirken diğer değişkenler bütünsel olarak değerlendirilip her biri çevrim içi uyum düzeyini etkiler. Ardından bu veriler üzerinde bir modelleme oluşturmak Korelasyon ile öznitelik belirlenmiş ve seçilen algoritmalar ile tekrar modellemeler oluşturulmuştur. Bu modellerde eğitim ve veri setleri sırasıyla %60-

%40, %70-%30, %80-%20 ve %90-%10 olarak test edilmiştir. Daha sonra modellerin performansı, f1-skor, kesinlik ve geri çağırma değerleri karşılaştırılmıştır. Oluşturulan modeller ile yapılan testler sonucunda başarı düzeyi en yüksek algoritma Korelasyon sonrası oluşturulan Karar Ağaçları algoritmasında %70 eğitim seti %30 veri seti olarak kullanılan model belirlenmiştir. Çalışma sonucunda, modelin başarı performansı 0.85, f1-skoru 0.85, kesinlik 0.77 ve geri çağırma 0.88 olarak bulunmuştur.

Model sonucuna göre çevrim içi uyum düzeyi en yüksek olan grup 71 kişiyle erkek öğrencilerdir. Erkek öğrencilerin arasında ise 21-25 yaş arası özel okulda okuyan üniversite öğrencilerinin uyumluluk seviyesi yüksektir. Bu verileri daha da özelleştirdiğimizde ise ekonomik durumu düşük, derse giriş aracı olarak bilgisayar ve internet bağlantısı olarak wi-fi, network tipi olarak 4G kullanan grup en yüksek uyumluluğa sahiptir.

Elde edilen sonuçlar incelendiğinde, kişilerin demografik ve ekonomik özellikleri çevrim içi öğrenme uyumluluğunu etkilemektedir. Bu durum öne alınarak belli aralıklarla çalışma yapılması ve öğrenme tasarımının bu parametreyi de kullanarak hazırlanması kişilerin başarısını etkileyeceği öngörülmektedir.

Name and Surname : Yağmur GÜR
Supervisor : Assoc. Dr. Atınç YILMAZ
Degree and Date : Master's (Thesis), 2024
Major : Computer Engineering
Keywords : Machine Learning, Online Learning
Compatibility, Prediction, Classification,
Accuracy Rate

ABSTRACT

MACHINE LEARNING MODEL TO PREDICT THE LEVEL OF ADAPTABILITY IN ONLINE EDUCATION

Today, machine learning modeling is used in many different sectors. Machine learning methods have shown high performance in the areas where they are used. The use of machine learning, which is used in different sectors, has become widespread with the rapid development of technology. The methods used in different sectors have seen and solved many problems. With the development of technology, the Education sector has also been affected by this. The use of distance learning or online learning has increased with the Corona period. However, it seems that many examples of online learning used as a learning method are not successful.

Today, machine learning modeling is used in many different sectors. Machine learning methods have shown high performance in the areas where they are used. The use of machine learning, which is used in different sectors, has become widespread with the rapid development of technology. The methods used in different sectors have seen and solved many problems. With the development of technology, the Education sector has also been affected by this. The use of distance learning or online learning has increased with the Corona period. However, it seems that many examples of online learning used as a learning method are not successful. According to the model results, the group with the highest level of online adaptation is male students with 71 people. Among male students, the compatibility level of university students aged 21-25 studying in private schools is high. When we further customize this data, the group

with a low economic status, using a computer as a means of accessing the course, wi-fi as an internet connection, and 4G as a network type has the highest compatibility.

When the results obtained are examined, people's demographic and economic characteristics affect online learning compatibility. It is envisaged that taking this situation into consideration, working at regular intervals and preparing the learning design using this parameter will affect the success of people.



İÇİNDEKİLER

Sayfa No.

ÖZ

ABSTRACT

TABLolar LİSTESİ	iii
ŞEKİLLER LİSTESİ	iv
FORMÜLLER LİSTESİ.....	v
SÖZLÜK.....	vi
GİRİŞ.....	1

1. KAVRAMSAL ÇERÇEVE

1.1. Benzer Çalışmalar	6
------------------------------	---

2. YÖNTEM VE METARYELLER

2.1. Yapay Zeka Tanımı.....	10
2.1.1. Yapay Zeka'nın Önemi	14
2.2. Makine Öğrenmesi	14
2.2.1. Danışmanlı(Gözetimli) Öğrenme	15
2.2.2. Danışmansız (Gözetimsiz) Öğrenme.....	16
2.2.3. Takviyeli Öğrenme	16
2.2.4. Çevrim içi (Online) Öğrenme.....	17
2.2.5. Çevrim dışı (Offline) Öğrenme	17
2.3. Makine Öğrenmesi Algoritmaları	17
2.3.1. Naive Bayes Algoritmaları	17
2.3.2. Karar Ağaçları Algoritmaları	18
2.3.3. K-NN Algoritmaları	20
2.3.4. Yapay Sinir Ağları.....	21
2.3.4.1. Yapay Sinir Ağının Ana Öğeleri	23
2.3.4.2. Yapay Sinir Ağlarında Öğrenme	24
2.3.4.3. Yapay Yapay Sinir Ağı Modelleri ve Yapıları	26
2.3.4.4. Yapay Sinir Ağının Katmanları	27
2.3.4.5. Tek Katmanlı Sinir Ağları - Perceptron.....	27
2.3.4.6. Çok Katmanlı Sinir Ağları	29

2.3.4.7. İleri Beslemeli Ağlar	30
2.3.4.8. Geri Beslemeli Ağlar	31
2.3.4.9. Yapay Sinir Ağının Çalışması, Öğrenmesi ve Test Edilmesi	32
2.4. Korelasyon ve Öznitelik Belirleme	33
3. DENEYSEL ÇALIŞMALAR	
3.1. Veri Seti.....	35
3.2. Modelin Oluşturulması	42
3.3. K-NN Algoritması Analizi	42
3.4. Naive Bayes Algoritması Analizi.....	44
3.5. Desicion Tree Algoritması Analizi.....	45
3.6. Yapay Sinir Ağları Algoritması Analizi.....	47
3.7. Bagging Trees Algoritması Analizi.....	48
3.8. Çalışmada Kullanılan Makine Öğrenmesi Modellerinin Karşılaştırılması	49
3.9. Korelasyon ile Öznitelik Seçimi	50
3.10. Korelasyon Sonrası K-NN Algoritması Analizi.....	51
3.11. Korelasyon Sonrası Naive Bayes Algoritması Analizi	52
3.12. Korelasyon Sonrası Decision Tree Algoritması Analizi	54
3.13. Korelasyon Sonrası Yapay Sinir Ağı Algoritması Analizi	55
3.14. Korelasyon Sonrası Bagging Tree Algoritması Analizi.....	56
3.15. Oluşturulan Hibrit Modellerinin Karşılaştırılması	58
3.16. En Başarılı Model ve En Başarılı Hibrit Model Karşılaştırması.....	59
SONUÇ	60
KAYNAKÇA.....	62

TABLÖLAR LİSTESİ

	Sayfa No.
Tablo 1. Yapay Zeka için Yapılan Tanımlar	10
Tablo 2. Bazı Karar Ağacı Algoritmaları Özellikleri	20
Tablo 3. Geleneksel Yöntemler ile YSA'nın Özelliklerini Karşılaştırılması	22
Tablo 4. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Cinsiyet.....	36
Tablo 5. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Yaş	37
Tablo 6. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Eğitim Düzeyi.....	38
Tablo 7. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Okul Tipi.....	38
Tablo 8. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Finansal Durum	39
Tablo 9. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – İnternet Tipi	40
Tablo 10. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Network Tipi.....	40
Tablo 11. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Kullanılan Araç...	41
Tablo 12. K-NN Modellerinin Karşılaştırılması	43
Tablo 13. Naive Bayes Modellerinin Karşılaştırılması.....	44
Tablo 14. Desicion Tree Modellerinin Karşılaştırılması.....	46
Tablo 15. Yapay Sinir Ağı Modellerinin Karşılaştırılması.....	47
Tablo 16. Bagging Trees Modellerinin Karşılaştırılması.....	48
Tablo 17. Modellerin Başarı Karşılaştırmaları.....	49
Tablo 18. Korelasyon Tablosu	50
Tablo 19. Korelasyon Sonrası Oluşturulan K-NN Algoritması Analizi Modelinin Karşılaştırılması	51
Tablo 20. Korelasyon Sonrası Oluşturulan Naive Bayes Algoritması Analizi Modelinin Karşılaştırılması.....	53
Tablo 21. Korelasyon Sonrası Oluşturulan Desicion Tree Algoritması Analizi Modelinin Karşılaştırılması.....	54
Tablo 22. Korelasyon Sonrası Oluşturulan YPA Algoritması Analizi Modelinin Karşılaştırılması	55
Tablo 23. Korelasyon Sonrası Oluşturulan Bagging Tree Algoritması Analizi Modelinin Karşılaştırılması.....	57
Tablo 24. Modellerin Başarı Karşılaştırmaları.....	58
Tablo 25. Desicion Tree ve Hibrit Modelin Başarı Karşılaştırması.....	59

ŞEKİLLER LİSTESİ

	Sayfa No.
Şekil 1. Danışmanlı Öğrenme.....	15
Şekil 2. Yapay Sinir Ağı Yapısı	23
Şekil 3. YPA.....	33



FORMÜLLER LİSTESİ

Sayfa No.

Formül 1. NB Formülü 18



SÖZLÜK

Makine Öğrenmesi: Sisteme gönderilen verilere göre öğrenen ve bu veriler üzerinden tahmin yaparak algoritmaların doğru çalışıp çalışmadığını araştıran bir alandır.

Yanlılık: Verinin ne ölçüde yanlış olduğunu belirler.

Yapay Zeka: İnsan davranışlarını taklit eden makineler veya sistemler anlamına gelir.



GİRİŞ

Türkiye’de MEB tarafından eğitim sistemi 3 kademeye ayrılarak toplanda 12 yıllık zorunlu eğitim olarak belirlenmiştir. Öğrenci zorunlu eğitimini tamamladıktan sonra yüksek öğretimini tamamlamak için en az 2 yıl daha okur. Eğer Lisans bölümü tercih ettiysen en az okuyacağı yıl sayısı 4 yıla yükselir. Süre gelen bu eğitim sisteminde geçmişten bu yana radyo, televizyon ile uzaktan öğrenme denemeleri yapılmaktadır. İnternetinde hayatımıza girmesiyle çevrim içi öğrenme denemeleri de yapılmaya başlanmıştır.

Çevrim içi eğitim; eğitimin uzaktan yapılma sürecidir. Uzaktan platformlar aracılığı ile eğitimler bir sisteme yüklenir. Öğrenci bu sisteme yüklenen ders içeriklerini talep eder, sorumluluklarını yerine getirir. Çevrim içi eğitimin diğer bir uygulanış şekli ise uygulamalar yardımı ile bir link üzerinden tüm sınıf aynı toplantıya katılır ve ders buradan işlenir.

Covid19 süreci ile birlikte çevrim içi öğrenmenin hem uygulaması hem de önemi artmıştır. UNICEF, Udemy, Coursera ve Udemix platformların raporları da bunu desteklemektedir.

Eğitim sektörünün değişimi ve yenilikleri adaptasyon sürecine 2020’de tüm dünyayı etkisi altına alan Covid19 hastalığı da hız katarak çevrim içi öğrenmenin tüm dünyada daha fazla kullanılmasına yol açmıştır. Covid19 sürecinde kademe fark etmeksizin ilköğretim, lise ve üniversite öğrencileri uzaktan öğrenmeye geçerek dersleri senkron ve asenkron olmak üzere çevrim içi olarak gerçekleştirmiştir. UNICEF Koronavirüs salgını süresinden uzaktan eğitime ulaşım konusunda bir araştırma yapıp bunu rapor olarak sunmuştur. Bu rapora göre dünyada yaşayan çocukların üçte biri, Covid19 salgını sebebiyle okulların kapanması sonrasında uygulanan uzaktan eğitime erişim sağlayamamıştır.

Bu çalışmanın amacı var olan sorunu araştırarak etkin bir çevrim içi öğrenme yöntemi kullanılabilmesi için bir stratejik bir yöntem sunmak, teknoloji alanında devam eden çalışmalara katkı sağlamak amaçlanmaktadır. Ayrıca eğitim satan şirketler de elde edilecek olan veriler ile çevrim içi öğrenme uyum seviyesine uygun olarak eğitim tasarımı yapabilirler.

Çalışma da Kaggle'dan elde edilen veriler öncelikle Excel'e aktarılmıştır. Veri setinden toplamda 1205 satır ve 9 sütun bulunmaktadır. Veri seti içerisinde doğruluk oranını etkileyecek durumlar göz önünde bulundurularak isnull parametresi ile boş veya eksik bir veri olup olmadığı tespit edilmiştir. Çalışmada veri setini modellemek için yoğun olarak kullanılan makine öğrenmesi algoritmalarından K-En Yakın Komşu (K-NN), Bagged Trees, Karar Ağaçları, Naive Bayes ve Yapay Sinir Ağı algoritmaları ve hibrit model olarak ise Decision Tree-Bagging Tree, Decision Tree-K-NN ve Decision Tree-Naive Bayes modelleri kullanılmıştır. Modelin çevrim içi öğrenme uyumluluk düzeyini tahmin etmek için 9 değişken içerisinde uyumluluk düzeyi bağımlı değişken olarak belirlenmiştir. Bu algoritmalar ile oluşturulan modellere ait doğruluk, geri çağırma, kesinlik ve f1-skor değerleri kıyaslanmıştır.

Bu çalışmada bulunan öncelikle benzer çalışmalara yer verilecek daha sonraki bölümlerde sırasıyla araştırmada kullanılan makine öğrenmesi modellerinin araştırma sonuçları, veri setinin oluşturulması ve modellenmesi, veri seti üzerinde kullanılan deneysel yöntemler, araştırmanın bulguları ve sonuçları yer alacaktır.

Eğitim Alanı ile İlgili Literatür Taraması

Eğitimin birden fazla tanımı bulunmaktadır. Bunların bazıları;

İnsan davranışlarında yetenek, karakter ve bilgi açısından bazı değişiklikler olduğunu veya gelişmeyi sağlamaya yönelik etkenlerin sistem olduğunu söyler.

Çevrim içi eğitim, teknolojiler kullanılarak verilen bir eğitim türüdür. Çevrim içi eğitim iki kategorilere ayrılabilir: senkron ve asenkron öğrenme asenkron uzaktan kumanda eğitim, kağıt üzerindeki talimatlardan öğrenme, kayıtlı dersleri dinleme veya esnek bir zaman diliminde kayıtlı görsel eğitimlerin izlenmesi gibi farklı zamanlarda öğretmenler ve öğrenciler arasındaki etkileşimlere dayanmaktadır. Bunun aksine, eşzamanlı öğrenme, canlı radyo programlarını dinlemek veya canlı çevrimiçi derslere katılmak gibi gerçek zamanlı etkileşim gerektirir. Elbette asenkron uzaktan eğitim, senkron muadillerinden daha eskidir çünkü ikincisi ancak iletişim teknolojilerindeki yeni gelişmelerle birlikte gelişmiştir. Her iki yöntem de popüler ve kullanışlı hale geldi. Dünyada çevrim içi eğitimi farklı üniversitelerde sunulduğunu görürüz. Güney Avustralya Üniversitesi (UniSA) aynı zamanda 32.000'den fazla öğrencisiyle en popüler uzaktan eğitim kurumudur. Eğitim programı olan bir üniversite olarak

görülüyor. Bu üniversitede uzaktan eğitim popüler olmaya devam ediyor. Üniversiteye öğrenci her yıl artıyor. Öğrenci sayısının bir milyona ulaşması bekleniyor. Türkiye’de de Anadolu Üniversitesi başta olmak üzere bir çok üniversite de örnekleri görülmektedir.

Çevrim içi eğitimi etkileyen en büyük faktörlerden biri teknoloji okur-yazarlığına sahip olmaktır. Bunun dışında öğrencinin maddi durumu, internete olan erişebilirliği, yaşı, kullandığı cihazı da etkilemektedir.

UNICEF tarafından Dünya genelinde geçerliliği olan bir araştırma 100 ülkeden veri alınarak gerçekleştirilmiştir. Araştırma kapsamında televizyon, radyo ve internete erişim ile hayatına devam eden okulların bu dönemde öğrencilere müfredatın sunulup sunulmadığına dair veriler toplandı.

Bu rapora göre yoksul hanelerde ve kırsalda hayatına devam eden okul çağındaki çocuklar, okulların kapanması sebebiyle eğitimden uzaklaşma ihtimali en yüksek olan gruptur. Dünya çapında en yoksul hanelerdeki okula giden çocukların %72’si çevrim içi olarak yapılan uzaktan eğitime ulaşamıyor.

Üst ve orta gelirli daha fazla olduğu ülkelerde ise en yoksul hanelerde hayatına devam eden ve uzaktan eğitime ulaşamayan okul çağındaki çocukların oranı %86’ya kadar çıkabiliyor.

Verilerin sonucunda ulaşılan sonuç bize kırsal bölgelerde hayatına devam eden okul çağındaki çocukların dörtte üçü uzaktan eğitime ulaşamıyor.

Raporda farklı yaş gruplarına göre uzaktan eğitime erişim oranları ele alınırken, en küçük yaşta çocukların, hayatlarının en önemli öğrenme ve gelişim evrelerinde olmalarına rağmen, uzaktan eğitime en fazla erişim sağlayamayan grup olduğu belirtiliyor.

Küçük çocuklara yönelik çevrimiçi eğitimin getirdiği zorluklar ve kısıtlamalar, bu eğitim kategorisinde uzaktan eğitim programlarının bulunmaması nedeniyle okul öncesi çağındaki çocukların yaklaşık %70’ine (120 milyon çocuk) erişilemedi. Uzaktan eğitime devam etmek için ev teknolojisi fırsatlarına ihtiyaç var. İlkokul çağındaki çocukların en az %29’una, bir başka deyişle 217 milyon öğrenciye, uzaktan eğitimle

ulaşılamamıştır. Ortaokul yaş grubundaki öğrencilerin en az %24'üne (78 milyon öğrenci) erişim sağlanamamıştır.

Teknolojik imkânların yetersizliği nedeniyle uzaktan eğitime erişemeyen en küçük grubu 48 milyon lise öğrencisinin (%18) oluşturduğu söyleniyor. Kullanımı aynı zamanda kişiyi iletişim açısından da güçlü kılar.

Çevrim içi eğitim sektörünün 2026 yılında tahmini 330 milyar dolar seviyesine ulaşması bekleniyordu. Yıllık büyüme oranı %9 olan çevrim içi eğitim sektörü yaşanan internete ulaşım sorunlarına rağmen Covid19 sürecinde büyük bir atılım yapmıştır. Çevrim içi eğitimde en hızlı çözümü sunan ancak aslında bir online eğitim platformu olmayan Zoom'un piyasa değeri Şubat 2020'de 25 milyar dolar iken bu süreçten sonra piyasa değeri 110 milyar dolar civarını buldu. Bu veriler bize çevrim içi öğrenmenin ne kadar ilgi gördüğünü de göstermektedir.

Bu verilerin yanı sıra aynı zamanda şahsi kayıtvolunabilecek ve çevrim içi eğitim alınabilecek birçok platform bulunmaktadır. Udemy, Coursera, Udacity gibi platformlar kişisel olarak çevrim içi eğitim alabileceğiniz platformlardır. Bu platformlarda tıpkı Zoom gibi 2020 yılında büyük atılımlara imza attı. Platformların kendi araştırma verilerine göre;

Udemy, Nisan 2020 itibarıyla kayıtlı öğrenci sayısında %425, eğitim içeriklerinde %55, işletmeler ve kamu kuruluşları tarafından kullanımda ise %80 artış elde ettiğini duyurdu.

Eylül ayında yayınlanan istatistiklere göre Coursera %60 artışla 70 milyon kullanıcıya ulaştı. Yine Coursera'da öğrenim gören kişiler %164'lük bir artışla 178 milyon saatlik eğitim içeriğiyle meşgul oldu.

Udacity ayrıca 2020'nin ilk çeyreğinde gelirinde %260'lık bir artış olduğunu açıkladı.

2020 yılında Eğitim-Öğretim yılında en çok kullanılan uygulamalardan biri EBA(Eğitim Bilişim Ağı)'ydı Milli Eğitim Bakanlığı'na bağlı tüm dersler bu platform altyapısı ile sunuldu.

EBA üzerinden çevrim içi eğitim, EBA'yı salgından en çok etkilenen yılın ilk döneminde dünyanın en çok ziyaret edilen eğitim sitelerinden biri haline getirdi. EBA, 2020 yılının ilk altı ayında 3,1 milyar ziyaretçiye ulaşarak Türkiye'nin en çok ziyaret edilen 10 internet sitesinden biri oldu. 7 milyondan fazla öğrenci ve 1 milyondan fazla öğretmenin kullandığı EBA, eksikleri olsa da süreç içerisinde Türkiye'yi çevrimiçi eğitime hazırladı.

İlköğretimden üniversiteye kadar her bir kademedeki öğrencilerin Eğitim-Öğretim'in sağlıklı bir şekilde devam edebilmesi için çevrim içi öğrenmeye uyumlu olması önem arz etmektedir.



1. KAVRAMSAL ÇERÇEVE

Bu bölümde tez çalışmasına benzer çalışmalara yer verilmiştir.

1.1. Benzer Çalışmalar

Bu bölümde çevrim içi öğrenme konusuyla ilgili yapılan çalışmalar araştırılmış ancak konu ile ilgili makine öğrenmesi kullanılarak yapılan çalışmalara pek rastlanmamıştır. Bu sebeple farklı alanlarda yapılan ve eğitim konusu ile ilgili yapılan çalışmalara yer verilmiştir.

Sosyal Medyada Makine Öğrenmesi ile Duygu Analizinde Dengeli ve Dengesiz Veri Setlerinin Performanslarının Karşılaştırılması başlıklı bilimsel makalesinde Rastgele Orman (RF), NB, Sıralı Makine Öğrenmesi yöntemlerini inceledi. Minimal Optimizasyon (SMO), denetimli öğrenme yaklaşımları. K-En Yakın Komşular (K-NN) ve Karar Ağacı'nı kullanarak sosyal medyada duygu analizi yaptı. Twitter, bir sosyal medya aracı olarak popülerliği ve karakter sınırlaması nedeniyle veri analizini kolaylaştırdığına inanılması nedeniyle kullanıldı. Çalışma sonucunda ML algoritmalarının Türk film eleştirilerinin duygu analizinde daha güçlü sonuçlar sağladığı ortaya çıktı. Yöntemler arasında başarı-performans karşılaştırması yapıldı ve ortalama %72,33 doğruluk oranıyla SMO algoritması seçildi (Nizam ve Akın, 2014, s. 16).

“Göğüs Kanseri Teşhisinde Farklı Makine Öğrenmesi Tekniklerinin Performans Karşılaştırması” makalesi çalışmasında göğüs kanseri teşhisinde 5 farklı makine öğrenmesi yöntemi kullanılarak algoritmaların performans karşılaştırmasını yapılmıştır. Çalışmasında Destek Vektör Makinesi (SVM), NB, Rastgele Forest (RF), KNN ve Lojistik Regresyon(LR) algoritmalarının test başarılarını karşılaştırmıştır. Test işlemi sonucunda en başarılı modelin %98,24 doğruluk oranıyla Lojistik Regresyon olduğu ortaya çıkartmıştır (Sevli, 2019, s. 19).

Yapay Zeka İle Meme Kanseri Lenf Nodu Analizi, araştırma sonucunda ise gerçekleştirilen prosedürlerin tıbbi güvenilirliğini kontrol etmek için teşhis testleri ve prosedürleri modellerin karşılaştırılması ROC eğrisi kullanılarak yapıldı. Modellerin sonuçlarını teşhis eden testin kullanışlılığını analiz etmek için Man Whitney U testi yapıldı. Normal dağılmayan sonuçlarda anlamlı farklılıklar olup olmadığı bulundu. Bu

test, teşhis testlerine göre anlamlı farklılıkları tespit etmek için yapılır. Doğruluğu, duyarlılığı ve özgüllüğü tanı testleri kullanılarak test edilmiştir. YSA Modelin doğruluk oranı %84,73, duyarlılık oranı %90,38, özgüllüğü ise %78,79'dur. KG-ANFIS modeli için doğruluk oranı %99,01, hassasiyet oranı %100, Spesifite %97,98 idi. AG-ANFIS doğruluk oranı %89,66, hassasiyet oranı değer %89,42 ve özgüllük %89,90 idi. Yapılan işlemlerin tıbbi açıdan güvenilirliğinin sınanması için tanı testleri ve ROC eğrisiyle modellerin karşılaştırılması yapılmıştır. Modellerin sonuçlarının tanı testi için kullanılabilirliğini analiz edebilmek için Man Whitney U Testi yapılmıştır. Normal dağılımda olamayan sonuçların anlamlı farklılıkların olup olmasını saptamak için yapılan bu testte, tanı testlerine uygun olarak anlamlı farklılıklar bulunmuştur. Tanı testleriyle doğruluğu, duyarlılığı, özgüllüğü sınanmıştır. YSA modeli için doğruluk oranı %84,73, duyarlılık oranı %90,38, özgüllüğü %78,79 bulunmuştur. KG-ANFIS modeli için doğruluk oranı %99,01, duyarlılık oranı %100, özgüllüğü %97,98 bulunmuştur. AG-ANFIS doğruluk oranı %89,66, duyarlılık oranı %89,42, özgüllüğü %89,90 bulunmuştur (Süer, 2019, s. 25).

Eğitimde Makine Öğrenmesi Araştırmalardaki Güncel Eğilimler Üzerine İnceleme adlı araştırma makalesinde, NB, DT, SVM algoritmalarından yararlanmıştır. Araştırmadaki amaç, eğitim bilimleri alanında ML algoritmaları kullanılarak 2015-2020 yılları arasında yazılmış makale eğilimlerini incelemiştir. Toplamda 201 makale incelenmiştir. Sorgulama sonucu 184 makale kriterlere uymuştur. Sonuç olarak bu makalelerde RF algoritmasının makalelerde en çok tercih edilen algoritma olduğu sonucuna ulaşılmıştır (Tosunoğlu, 2021, s. 24).

Sosyo Ekonomik Değişkenlerin Öğrencilerin Akademik Başarısı Üzerine Etkileri adlı çalışmada, öğrenci başarısı; aile eğitimi, öğrencilerin cinsiyeti, ekonomik durumu, kardeş sayısı gibi faktörlerin başarıyla nasıl ilişkili olduğuna dair bir çalışma. Veriler öğrencilere gönderilen anket soruları aracılığıyla toplanmış ve analiz edilmiştir. İki kardeşe sahip olmak, iki kardeşe sahip olmaktan daha az başarılıdır. Gelirin akademik başarıya olumlu etkisi var, cinsiyet faktöründe ise erkekler için sonuçların olumlu olduğu görüldü (Aslanargun, 2016, s. 22).

Pısa 2009 Öğrenci Başarı Düzeylerini Etkileyen Faktörlerin Değerlendirilmesi adlı çalışmadaki amaç, Türkiye verilerini istatistiksel yöntemlerini kullanarak

analiz etmektir. Bu amaçla öncelikle toplam 170 okuldan 4.996 15 yaşındaki öğrenci okuma, temel bilim ve matematik eğitimi aldı. Eğitimi etkileyen faktörlerin belirlenmesinde t ve F testleri kullanıldı. O okudu. Daha sonra çok değişkenli istatistiksel analiz tekniği Lojistik regresyon analizini kullanarak fen, matematik ve okuma öğrencileri. Sonuçları etkileyen faktörler belirlenmeye çalışıldı. Sonuçların analizi; Öğrenci başarı oranları cinsiyet, okul yaşı ve veliden etkilenmektedir. Eğitim gibi değişkenlere göre farklılık gösterdiği gösterilmiştir (Dr. Gürsakal, 2012, s. 23).

Eğitimde Yapay Zeka ve Uygulamaları çalışmasında eğitimde yapay zeka konusu üç başlık altında ele alınmıştır. Zeka ve ilgili kavramlar üç soru üzerinden inceleniyor: "Yapay zeka tam olarak nedir?" bir soruyla; Yapay zeka Eğitime nasıl katkı sağlayacak? "Yapay zeka eğitimi nasıl geliştirir?" bir soruyla; Son bölümde "Eğitimde Yapay zekanın uygulamaları nelerdir?" Sorular çerçevesinde eğitim ve öğretimde kullanılan/kullanılabilecek yapay zeka uygulamalar tanıtıldı. Bu çalışma yapay zeka konusunu eğitimin genel çerçevesinde incelemektedir. Alana tanıtarak, öğretmen ve öğrenciler tarafından nasıl kullanılabileceğini keşfederek katkı sağlar (Arslan, 2017, s. 23).

Makine Öğrenmesi Yöntemleri İle Akademik Başarının Tahmini başlıklı araştırma makalesinde, öğrencilerin yaşam koşullarının ve sosyal çevrelerinin Türkçe dilini nasıl etkilediğini belirlemek amacıyla ilk olarak 6, 7 ve 8. sınıf öğrencilerine 24 maddelik bir anket uygulanmış ve etkisi değerlendirilmiştir. Anket soruları ebeveynlerin sağlık durumu, ebeveynlerin çalışma durumu, eğitim durumu, üvey ebeveynlik durumu, öğrenci yaşı, kardeş sayısı, ev durumu, öğretmenlik durumu, oyun sıklığı, haftalık çalışma saatleri ve yaşam memnuniyeti gibi soruları içeriyordu. Bu araştırma makalesinde kullanılan anket sorularından bazıları, araştırma makalesine referans olarak kullanılmıştır (Gök, 2017, s. 139).

Bilişim Alanında Öğrenim Gören Lise Öğrencilerine Yönelik Üniversite Tercihlerinin Yapay Zeka Yöntemleri ile Analizi araştırmasında mesleki ve teknik ortaöğretim okullarında okutulan bilişim dersleri öğrencilerin ne kadarı eğitimlerini tamamladıktan sonra üniversite tercihlerinde bilgisayar bilimleri bölümlerini seçiyor? veya tercihlerini etkileyen faktörler analiz ediliyor ve Lise mezunları üzerinden ortaöğretim sonrası süreç incelendi. Aynı zamanda ilgi alanlarındaki birlikteliklerin

tespiti için yapılmış bir çalışmadır. Sonuçlar bilişim meslek alanları açısından incelenmiştir. Üniversitelerdeki bilişim bölümlerine göre filtreleme yapılmış ve incelenmiştir. Ortaya çıkan sonuçların yorumları yapılmış sadece yapılan ankete katılan 217 kişinin verdiği cevaplar temel alınarak hesaplanmış ve sadece o cevaplara dikkate alınarak çıkarımlar yapılmıştır (Türk, 2021, s. 34).

Makine Öğrenmesi Yöntemi İle Eğitim Başarısının Tahmini Modeli araştırmasındaki amaç; Amaç, Türkçe dersi değerlendirmeleri yoluyla öğrenci başarısını doğrudan etkileyen sorunları tespit etmek ve ortaya çıkarmaktır. Çalışma sonunda program sisteme verilen yeni girişlerle öğrencinin Türkçe puanını tahmin edebilecektir. Çalışmada veri modelleme için makine öğrenimi yöntemleri kullanıldı. K-Nearest Neighbor (K-NN) , Random Forest (RF) , Lineer Regresyon, Bagged Trees, Gradient Boosting Regressor (GBM) ve Karar Ağaçları(DT) kullanıldı. Türkiye tahmin modeli beklendiği gibi Türkçe diploması 13 değişken arasında bağımlı değişken olarak seçilir. Algoritmalar eğitim verilerinin %80'i ile eğitilmiştir.

Modelin eğitim sonrası tahmin başarısını değerlendirmek için model doğrulama prosedürü gerçekleştirildi. Bu bağlamda en büyük performans Random Forest (RF) algoritması seçildi. algoritmanın performansı 0.88 ve R-squared değeri 0.98 olarak belirlenmiştir (Zilyas, 2023, s. 35).

2. YÖNTEM VE METARYELLER

Bu bölümde tez çalışmasında uygulanan yöntem ve materyallerle ilgili bilgi verilmiştir.

2.1. Yapay Zeka Tanımı

Yapay zeka, evrendeki herhangi bir davranışa benzetme yapmayı amaçlayan ve içerisinde birden fazla farklı yaklaşımı ve algoritmayı içeren yöntemler topluluğudur (Yılmaz, 2020, s. 1).

Başka bir deyişle yapay zeka, insan zekasını taklit eder ve insanın hem fizyolojik hem nörolojik olarak modellenmesiyle yaratılır. Bunun sayesinde makine sistemleri ölçülü kararlar alabilir ve insana benzeyen davranışlar sergileyebilir.

Tablo 1. Yapay Zeka için Yapılan Tanımlar

Yazar	Yıl	Tanım
Bellman	1978	İnsan düşüncesiyle bağlantı kurulan faaliyetlerin otomasyonu. Karar yapısı, sorun çözme gibi faaliyetler
Haugeland	1989	Makineleri düşündürmeyi amaçlayan heyecan verici bir çaba. Başka bir ifadeyle zihinsel makineler
Kurzweil	1990	İnsanlar tarafından gerçekleştirildiğinde akıl gerektiren davranışları yapan makineler üretme
Schalkoff	1990	Akıllı davranışları hesaplayan süreçleri açıklayan ve bu davranışları taklit eden bir çalışma alanı
Rich ve Knight	1991	Bilgisayarların insanlardan daha iyi olabileceği şeyleri nasıl yapacağını araştıran çalışma
Winston	1992	Algılama, akıl yürütme ve harekete geçmeyi sağlayan hesaplamaların incelenmesi

Kaynak: Karaboğa, U. (2020). *İşe alım süreçlerinde yapay zeka teknolojilerinin kullanımı* [Yüksek Lisans Tezi]. İstanbul Medipol Üniversitesi.

Yapay Zeka'nın tarihi çok uzun zaman öncesine, tarih öncesi döneme dayanır. Tarihinden kısaca bahsedecek olursak şu şekildedir.

Tarih Öncesi Dönem'de İngiliz araştırmacılarından biri olan Leibniz, dört matematik işlemini yapabilen makine üreterek yapay zekanın temelini atmıştır. Bunun yanında yapay zeka için en önemli adımlardan biri de 1884'te Charles Babbage'in makinelere zeki davranışlar sergiletme üzerine çalışmasıdır (Say, 2018, s. 15).

Yapay zekânın diğer bir önemli gelişmelerinden biri de Turing'in yaptığı çalışmalardır. Turing, 1950'li yıllarda "Makineler düşünebilir mi?" sorusu ile araştırmalara başlamıştır ve Turing Makine adı verilen bir makine ortaya çıkmıştır. Bu çalışmalar bize yapay zekanın mantıksal olarak hesaplamalar sonucu elde ettiği verilerle insan zekasını taklit edebileceğini gösterdi.

Yapay Zeka terimi dünyada ilk kez McCarthy tarafından 1956'da kullanıldı. 1965'te Herbert Simon "makinelere insanların yapabildiği hemen hemen her şeyi yapabileceğini" söyledi. 1972'de Alain Colmerauer, yapay zeka sistemlerini üretebilmek için PROLOG adı verilen bir programlama dili icat etti. Bu durum yapay zeka çalışmalarına hız kattı.

IBM'in ürettiği "Deep Blue" isimli bilgisayar 1997'de dünya satranç şampiyonu oldu. 2011'de "Watson" isimli yapay zekâ kullanan bir bilgisayar televizyon yarışmasında rakiplerine karşı başarılı oldu. 2016'da önemli gelişmelerden biri "Google Deepmind"ın 'Go'da dünya şampiyonu olmasıdır.

Yapay zekanın 20. ve 21. yüzyıldaki tarihi gelişimi şöyle sıralanabilir:

- 1923 yılında Čapek "RUR" (Rossum's Universal Robots) isimli oyununda robot kelimesini kullandı.
- 1943 yılında Sinir ağları çalışmalarında sinir ağları meydana getirildi.
- 1945 yılında Asimov, Robotik terimini kullandı.
- 1950 yılında Turing, "Computing Machinery and Intelligence" isimli çalışmasında Turing Testi'ni denedi ve tanıttı.
- 1956 yılında McCarthy, yapay zeka terimini kullandı.

- 1958 yılında McCarthy, LISP isimli programlama dilini üretti. Yapay zeka çalışmalarında bu dili kullandı.
- 1964 yılında Bobrow, bilgisayarların doğal dilde matematiksel kelime problemlerini çözebilecek yetkinlikte olduğunu ispatlayan tez çalışması hazırladı.
- 1965 yılında Weizenbaum, etkileşimli bir program olan ELIZA'yı geliştirdi.
- 1969 yılında Stanford Araştırma Enstitüsü bilim insanları Shakey isimli, algı, hareket ve problem çözme yetkinliğine sahip robot üretti.
- 1973 yılında Edinburgh Üniversitesi "Assembly Robotics" isimli grup, vizyonu kullanarak modelleri bulmak ve monte etmek için Robot Freddy'yi kurdu.
- 1979 yılında Stanford Cart isimli uzaktan kontrollü otonom araç geliştirildi.
- 1985 yılında Cohen, Aaron isimli çizim programı üretti.
- 1990 yılında Yapay zeka, veri madenciliği, veri bilimi, makine öğrenimi, gibi birçok alanda büyük gelişmeler kaydetti.
- 1997 yılında IBM'in "Deep Blue" isimli bilgisayarı, satranç dünya şampiyonu olan Garry Kasparov'u mağlup etti.
- 2000 yılında Etkileşimli robot ev hayvanları piyasaya çıktı. (Boden, 2018)

Yapay zeka (YZ) alanındaki diğer önemli dönüm noktaları ise şunlardır:

- Yapay Zeka - 2001: Bu dönem yapay zekanın "bahar dönemi" olarak adlandırılmaktadır. Yapay zeka alanında önemli gelişmelerin yaşandığı ve yeniden ilgi gördüğü bir dönemdir.
- Derin Öğrenme - 2006: En az bir YPA'nın kullanıldığı ve bilgisayarın eldeki verilerden yararlanarak yeni veriler bulduğu bir öğrenme yöntemidir. Derin öğrenme, yapay zeka alanında büyük bir ilerleme sağlamıştır.
- Apple Siri - 2011: Siri, bir bilgisayar yazılım uygulaması olarak geliştirilen akıllı kişisel asistan ve bilgi gezgini olarak kullanılan bir yapay zeka

çözümüdür. Siri, soruları cevaplayabilir, tavsiyelerde bulunabilir ve sosyal ağlardaki hizmet eylemlerini gerçekleştirebilir.

- Grafik İşlemcileri Çağı - 2012: Grafik işlemci, grafikleri işlemek ve göstermek için kullanılan yüksek performanslı donanımlardır.
- Amazon Alexa - 2014: Amazon tarafından geliştirilen yapay zeka destekli sanal bir asistan çözümüdür. Alexa, kullanıcının sesli komutlarını anlayarak çeşitli görevleri yerine getirebilir, bilgi sağlayabilir ve ev otomasyonu gibi işlemleri yönetebilir.
- Asilomar - 2017: Asilomar, yapay zekanın günümüzü ve geleceğini tartışmak amacıyla düzenlenen bir konferanstır. Yüzlerce bilim insanının katıldığı bu konferansta, 3 kategori ve 23 maddeden oluşan bir bildirge imzalanmıştır. (Karataş, 2021).
- Üretken Yapay Zeka (OpenAI Üretken Dil Eğitimi) - 2018: Alec Radford'un bir dil modelinin üretken ön eğitimi (GPT) hakkındaki makalesi OpenAI'nin web sitesinde yeniden yayınlandı; bu makale, üretken bir dil modelinin geniş ve çeşitli bir veri kümesi üzerinde ön eğitime dayalı olarak denetimsiz bilgi ve süreç bağımlılıklarını nasıl elde edebileceğini gösterdi. Bu durum Üretken Yapay Zeka'nın gelişiminin de başlangıcı oldu.
- Üretken Yapay Zeka (GPT-2 Dil Modeli) - 2019: OpenAI, Reddit gönderilerinde en az üç olumlu oy alan URL'lerdeki metinler de dahil olmak üzere dokuzdan fazla görev belgesinden oluşan bir veri kümesi üzerinde eğitilen GPT-2 dil modelinin tam sürümünü yayınladı.
- Üretken Yapay Zeka (Chatgpt) -2022: ChatGPT, beş gün içinde bir milyon kullanıcıya ulaşan bir yapay zeka aracı olan GPT-3.5'i piyasaya sürdü.
- Üretken Yapay Zeka (Chatgpt) -2023: Google, kendi üretken yapay zeka sohbet robotu Bard'ı, OpenAI, ücretli bir "premium" seçeneğiyle birlikte botlarının başka bir sürümü olan GPT-4'ü yayınladı.

2.1.1. Yapay Zeka'nın Önemi

Yapay Zeka günümüzde önemli, güçlü ve gelişime açık bir teknolojidir. Sağlık, finans, emniyet, turizm, enerji gibi birçok alanda kullanılmaya başlanmış ve kullanılmaya daha da artarak devam edecektir. Türkiye Yapay Zeka İnsiyatifi'nin 2017 Çalıştay Raporu'na göre yapay zekanın kullanılmadığı alan kalmayacak.

2.2. Makine Öğrenmesi

Makine öğrenmesi, bilgisayarların algılayıcısı veya veri tabanları gibi veri türlerine dayanan öğrenim olanaklı kılan algoritmaların tasarım ve geliştirme süreçlerini konu edinen bir bilim dalıdır. Başka bir tanımla makine öğrenmesi, bilgisayarlara programlama gerektirmeksizin bilgiyi öğrenebilme yetisi kazandıran çalışma alanı olarak ifade edilir.

Makine öğreniminin odaklandığı konular bilgisayarlara karmaşık örüntüleri algılama ve veriye dayalı akılcı kararlar verebilme yetisi kazandırmaktır. Bu, makine öğreniminin istatistik, olasılık kuramı, veri madenciliği, örüntü tanıma, yapay zeka, uyarlamalı denetim ve kuramsal bilgisayar bilimi gibi alanlarla yakından ilgili olduğunu göstermektedir (Yardımcı, 2011, s. 11).

Başka bir deyişle makine öğrenmesi, verilerden öğrenir. Öğrendikleri bilgilerden çıkarım yaparak kullanıcıya uygun sonuçlar ortaya çıkartır. Makine Öğrenmesi ile yapılan programlar sürekli öğrenir ve kendilerini geliştirir.

Makine Öğrenmesi yöntemini kullanırken temelde doğru veriye sahip olmak önemlidir. Kurduğunuz programın sonucu verilerinizin doğruluğuna göre sonuç verecektir. Ne kadar fazla ve doğru veriyi programınıza yüklerseniz program size o kadar doğru sonuç çıkartacaktır.

Günlük hayatımızda; kullanıcıya kitap, müzik vb. önerisi getiren programlar, yüz tanıma uygulamaları, Otonom Araç ve Google Çeviri gibi uygulamalarda görürüz.

Başka tanıma göre ise makine öğrenmesi, E deneyimi ve buna uyan bazı T görev sınıflarından, P performansı ile öğrenmesi söylenen bir bilgisayar programıdır. Eğer T görevlerindeki performansı P ile ölçülebiliyor ise, E deneyimi ile geliştirilebilir. Bu yapıya aşağıdaki örnek verilmiştir.

- Görev T: Dama oynamak

- Performans ölçütü P: Rakiplerle oynanan oyunlardaki kazanma yüzdesi
- Antrenman deneyimi E: Kendisi ile alıştırmaya oyunları oynaması. (Ünsal, 2011).

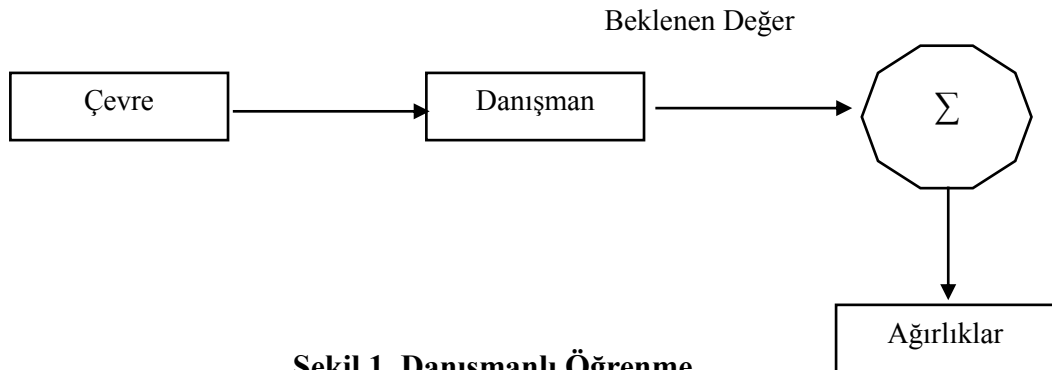
Yapay zekanın makine öğrenmesi çalışma prensibi Google arama motoru üzerinden şu şekilde açıklanabilmektedir: sürekli internet bağlantıları arasında dolaşan yeni siteler arayıp keşfeden Google tarayıcıları, bu web sitelerini birçok veriden yararlanarak belirleyip Google sunucularına iletmekte ve bu bilgiler Google tarafından hatırlanır hale gelmektedir. Google bu şekilde öğrenmekte ve öğrenmesinin ardından herhangi bir arama söz konusu olduğunda konuyu tespit edebilecek bilgiyi algılayıp öğrenme sürecinde edindiği bilgiler ile algoritma sayesinde ilgili internet sitelerini aratana sunmaktadır (Zorluel, 2019, s. 142).

Makine Öğrenmesi kullandığı öğrenme yöntemleri üç ana başlıkta söylenebilir.

2.2.1. Danışmanlı(Gözetimli) Öğrenme

Bu öğrenme modelinde verilerin giriş-çıkış değeri sisteme verilir. Yani sistem giriş verilerine karşılık olan çıkış verilerini de en baştan bilir. Bu sisteme verilen giriş ve çıkış veri setlerine eğitim seti de denir.

Örnek vermek gerekirse bilgisayarların özelliklerinde işlemci tipine karşımıza çıkaran bir program yazmak istiyorsak önce bilgisayarların marka, model, işlemci, RAM, ekran kartı özelliklerini girmemiz gerekir. Böylece programa markasına göre işlemci tipini sorduğumuzda sonucu verecektir.



Şekil 1. Danışmanlı Öğrenme

Kaynak: Yılmaz, A. (2020). *Yapay zekâ*. Kodlab Yayınevi.

Danışmanlı Öğrenme Sınıflandırma, regresyon ve problem sıralama ile ilişkilendirilen en yaygın öğrenme türüdür. (Rastomizadeh ve Talwalkar, 2012, s. 7). Yaygın olarak danışmanlı öğrenme kullanan algoritmalarından bazıları şunlardır:

- YPA
- NB
- K-NN En Yakın Komşu
- Karar Ağaçları

2.2.2. Danışmansız (Gözetimsiz) Öğrenme

Danışmansız Öğrenme’de sisteme sadece giriş verileri verilir. Ortada bir problem vardır, bu problemin girdileri de organize edilmiştir. Sisteme bu şekilde yüklenir ve sistemden sonuç üretmesi beklenir.

Danışmansız Öğrenme, boyut azaltma ve kümeleme denetimsiz öğrenme problemlerine örnek olarak verilebilir (Rastomizadeh ve Talwalkar, 2012, s. 7).

Danışmansız öğrenme YPA algoritmalarında daha çok kullanılır. YPA’ya öğrenme esnasında örnek girdiler verilir. Ancak çıktılar verilmez. Çıktı bilgisi ağdan beklenir. Sınıflandırma problemlerinde kullanılır. Sınıflandırma problemlerinde danışmanlı öğrenme de kullanılır. Danışmanlı öğrenme, danışmanlı öğrenmeye göre daha uzun sürer.

2.2.3. Takviyeli Öğrenme

Takviyeli Öğrenme, sistemlerinin danışmanlı ve danışmansız öğrenme yöntemlerine benzer yanları bulunmaktadır. Sisteme danışmansız öğrenme yöntemlerinde olduğu gibi probleme ait giriş bilgileri verilmektedir. Sistem için bir danışman bulunarak çıkış verilerinin doğru veya yanlış olduğu sonucu verir. Bu şekilde sisteme danışmalı öğrenme de yapılmış olur (Yılmaz, 2022, s. 57).

Makine Öğrenmesi kullanılırken zaman bağlı öğrenme yöntemleri iki ana başlıkta söylenebilir.

2.2.4. Çevrim içi (Online) Öğrenme

Bu öğrenme modeli üzerinde öğrenme bitmez. Sisteme veriler yüklenerek sistem kurulur ancak gerçek zamanlı çalıştığı için dışardaki verileri de alıp öğrenmeye devam eder. Buna örnek olarak öneri sistemlerini verebiliriz.

2.2.5. Çevrim dışı (Offline) Öğrenme

Bu öğrenme modelinde veriler site siteye yüklenir, sistem gerçek zamanlı çalışmaya başlamadan önce son hali verilir ve test edilir. Daha sonra kullanıcıların kullanımına hazır hale getirilir.

2.3. Makine Öğrenmesi Algoritmaları

Bu tez çalışmada kullanılan veri seti sınıflandırılmış bir şekilde tahmin üretmek için kullanılacaktır. Danışmanlı Öğrenme Modelinden yola çıkarak kullanılan algoritmalar arasında karşılaştırmalar yapılacaktır.

2.3.1. Naive Bayes Algoritmaları

NB algoritmasının temeli 18. yüzyılda Thomas Bayes tarafından atılmıştır. NB algoritmalarında olasılıklar üzerinden tahminleme yapılabilir.

NB algoritması sisteme yüklediğimiz verilerden yola çıkar, koşullu olasılık fonksiyonunu kullanarak yeni verilerin sonucunu tahmin etmeye odaklanır. NB algoritmaları hem sayısal hem de sözel metinlerin tahminin de kullanılmaktadır.

Başka bir şekilde tanımlamak gerekirse NB algoritmasında sisteme belli bir sayıda veri öğretilir. Bu veriler sınıflandırılmış verilerdir. Daha sonra NB'ye sorulan yeni veri yani öğretilmemiş veride NB koşullu olasılıktan yararlanarak yeni veriyi sınıflandırır ve sonuç çıkartır. NB'ye ne kadar fazla veri öğretirsek daha doğru sonuç alma olasılığımız artar.

NB algoritması belli bir sınıfa ait nitelik değerinin diğer nitelik değerlerinden bağımsız olduğunu varsayar. NB algoritmasının yaptığı bu varsayıma koşul bağımsızlığı denir. Koşul bağımsızlığı, belirli bir nitelik değerinin var olup olmamasının diğer nitelik değerinin var olup olmayacağına bir etki etmeyeceğine

ifade eder. Bu varsayım, hesaplamaların daha kolay bir şekilde gerçekleştirilmesini sağlar. NB algoritması; hesaplama etkinliği, basit yapısı ve yüksek doğru sınıflandırma başarımı nedeniyle metin madenciliği başta olmak üzere pek çok sınıflandırma çalışmasında başarı ile uygulanmaktadır. (Onan, 2016, s. 28).

Günümüzde NB algoritması ile birçok e-posta ve mesajın içeriğinin gerçekten kişiler tarafından mı gönderildiğini yoksa toplu olarak gönderilen reklam mı yoksa sahte mesaj mı (spam), büyük bir doğrulukla tespit etmek mümkündür.

Aynı zamanda Bayes Karar Algoritmaları, istatistiksel örüntü tanıma için kullanılan en yaygın yöntemdir (Yılmaz, 2022, s. 51).

$$P(A/B) = \frac{P(B/A) P(A)}{P(B)}$$

Formül 1: NB Formülü

Kaynak: Yılmaz, A. (2020). Yapay zekâ. Kodlab Yayınevi.

$P(A | B)$ = B olayı gerçekleştiğinde A olayının gerçekleşme olasılığı

$P(A)$ = A olayının gerçekleşme olasılığı

$P(B | A)$ = A olayı gerçekleştiğinde B olayının gerçekleşme olasılığı

$P(B)$ = B olayının gerçekleşme olasılığı

2.3.2. Karar Ağaçları Algoritmaları

Karar ağaçları, makine öğrenmesi uygulamalarında sınıflandırma ve tahmin için oldukça yoğun kullanılan bir algoritmadır.

Karar ağaçları, büyük kararları küçük kararlara bölerek karar yapılarını oluşturur. Tahmin etme ve tamamlama özelliğine sahiptir. Sinir ağları gibi diğer metodolojilerin de sınıflandırma için kullanılabilmesine rağmen, karar ağaçları, kolay yorumu ve anlaşılabilirliği açısından karar vericiler için avantaj sağlamaktadır.

Karar ağaçları;

- Düşük maliyetli olması,
- Anlaşılmasının, yorumlanmasının ve veri tabanları ile entegrasyonun kolaylığı,
- Güvenilirliklerinin iyi olması gibi nedenlerden

ötürü en yaygın kullanılan sınıflandırma tekniklerinden biridir. (Çalış ve Kayapınar, 2014, s. 6).

Karar ağaçları kapsamında kullanılan algoritmalarından bazıları şu şekildedir: ID3, C4.5, C5.0, CART, CHAID ve QUEST.

Karar ağacı temelli analizlerin yaygın olarak kullanıldığı alanlar şunlardır:

Belirli bir sınıfın olası üyesi olacak elemanların belirlenmesi,

Çeşitli vakaların yüksek, orta, düşük risk grupları gibi çeşitli kategorilere ayrılması,

Parametrik modellerin kurulmasında kullanılmak üzere çok sayıdaki değişkenden en önemlilerinin seçilmesi,

Gelecekteki olayların tahmin edilebilmesi için kurallar oluşturulması,

Sadece belirli alt gruplara özgü olan ilişkilerin tanımlanması,

Kategorilerin birleştirilmesi ve sürekli değişkenlerin kesikli değişkenlere dönüştürülmesidir (Albayrak ve Yılmaz, 2009, s. 40).

Tablo 2. Bazı Karar Ağacı Algoritmaları Özellikleri

Karar Ağacı Algoritması	Özellikleri
ID3	Bu algoritmayı kullanırken ilk olarak sınıfın entropisi hesaplanır. Daha sonra her bir nitelik için ayrı ayrı entropi hesaplanır ve sınıf entropisinden çıkarılır. Böylece en büyük bilgi kazanımına sahip nitelik seçilir ve dallanma başlar.
C4.5, C5.0	ID3 algoritmasının geliştirilmiş halleridir.
CART	Öznitelik seçim ölçüsü olarak Gini indeksini kullanır. İstatistikçi Leo Breiman tarafından önerilmiştir. Hem sayısal hem de kategorik verilerin analizinde kullanılır. Sınıflandırmayı ve regresyonu destekler.
CHAID	Ki Kare Testini bir öznitelik seçim ölçüsü olarak kullanır.
QUEST	İkili bölünmeli bir karar ağacı algoritmasıdır. Uygulanması CART algoritmasına benzerdir.

Kaynak: Yazar tarafından oluşturulmuştur.

2.3.3. K-NN Algoritmaları

K-NN algoritması "k-Nearest Neighbors", "k" en yakın komşu ifadesinin kısaltılmış halidir.

K-NN algoritmaları Danışmanlı Öğrenme Yöntemi içerisinde kullanılır. K-NN algoritması kolay uygulanabilirliği ve basit olma özelliği ile en çok kullanılan makine öğrenmesi algoritmalarından biridir. Regresyan ve sınıflandırma yaparken kullanılır. Adından da anlaşılacağı üzere girilen verinin eldeki verilerin yakınlığına bakarak sonuç çıkarır. Buradaki k bir yakınlık ölçüsüdür (Lantz, 2013).

kNN algoritması bir sınıflandırma algoritması olduğundan veri setindeki özellik bilgilerini belirli bilgi ve özellik etiketleriyle kullanarak gözlemleri sınıflandıracak yapıları modeller. Daha sonra etiketlenmemiş verilerin hangi sınıfa ait olduğunu önemli bir doğrulukla tahmin edebilir. Basit bir senaryo üzerinden kNN algoritmasının nasıl çalıştığını açıklamak daha açıklayıcı olacaktır.

KNN algoritması hesaplama yaparken iki önemli değeri kullanır.

K(Komşuluk Katsayısı): En yakın değer alan kaç komşunun değeri üzerinden hesaplama yapılacağını belirtir. Probleme en iyi şekilde optimize etmek için K'ye vereceğimiz değer önemlidir. Çok küçük veya çok büyük değerler verirsek problemiden uzak sonuçlar elde edebiliriz. K=3 vermek yaygın kullanılan ve bizi en yakın sonuca götürme olasılığı fazla olan değerdir.

Distance (Uzaklık): Değerin diğer değerlere uzaklığı hesaplamasını ifade eden terimdir.

KNN algoritmasında en uygun K değerinin seçilmesi önemlidir. Algoritma, K değerini merkeze alarak sorgu noktası olarak kullanır. Bu şekilde hangi sınıfa ait olduğunu bulur..

K=1 durumunda sorgu noktası “kare” sınıfına girer. K=3 durumunda ise sorgu noktası “üçgen” sınıfına girer.

K değerinin büyümesi bazı sorunlara yol açabilir. Bu sorunlardan biri eğer yeterli eğitim seti içinde yoksa yanlış bir sınıflandırmaya sebep olmasıdır.

Eğitim seti içerisinde verinin K değerini ortalama uzaklıktan daha uzak olması KNN algoritmasını faydalı kılmayabilir. Bunun sebebi verinin kullanılan sınıftan farklı bir sınıfa ait olma olasılığıdır (Kocamaz, 2012, s. 13).

2.3.4. Yapay Sinir Ağları

YPA; yapay zeka yöntemlerinden biri olarak, insan beynini modelleyerek taklit etme çabalarından biridir. Modellemede insan beyninde bulunan biyolojik sinir hücreleri (nöron) baz alınmaktadır (Yılmaz, 2022, s. 61).

Başka bir deyişle insan zekasını matematiksel bir şekilde taklit edilmesidir.

YPA, insan beynine iki şekilde benzerlik göstermektedir:

1.Bilgi, öğrenme süreci yoluyla ağ tarafından elde edilir.

2. Sinaptik ağırlıklar olarak bilinen nöronlar arası bağlantı kuvvetlerini, bilgiyi depolamak için kullanır (Haykin, 1999, s. 36).

İnsanların nasıl biyolojik olarak sinir hücreleri var ise YPA'nın da sinir hücreleri bulunmaktadır. YPA da veri setlerine göre nöronlar kümelendirilir.

Kullanılan yöntemlerden farklı yöntemler sunmaktadır. Tablo 2'de karşılaştırılması verilmiştir (Yılmaz, 2022, s. 65).

Tablo 3. Geleneksel Yöntemler ile YSA'nın Özelliklerini Karşılaştırılması

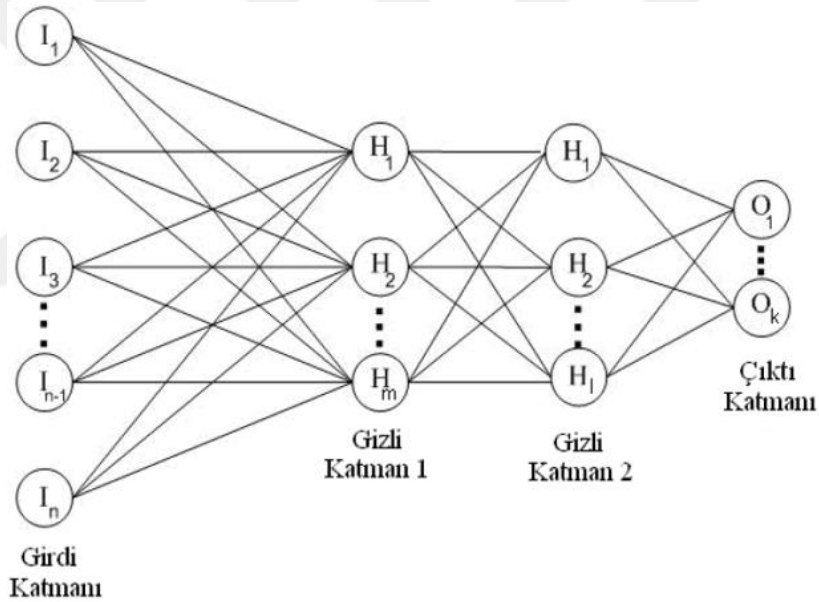
Geleneksel Yöntemlere Sahip Sayısal Bilgisayarlar	YPA
Sonuç üretmek için giriş bilgilerine geleneksel sahip kurallar uygulanır.	Probleme ait giriş ve sonuçlar modelin eğitimi için sisteme verilir. Parametreler ve kurallar modelleyen tarafından belirlenir.
Hesaplama eş zamanlı ve merkezidir.	Hesaplama eş zamansız ve toplu şekilde gerçekleşir. Öğrenmeden sonra paralel işleme gerçekleşir.
Bellek paketlenmiş; yeri adreslenebilir.	Bellek ayrılmış; içerik adreslenebilir.
Hata toleransı yoktur.	Bilgi eksik, karışık, gürültülü ise hata toleransı uygulanabilir.
Hızlıdır.	Yavaştır.
Bilgiler ve kullanılan yöntemler kesindir.	YSA deneyimlerden yararlanır.

Kaynak: Yılmaz, A. (2020). *Yapay zekâ*. Kodlab Yayınevi.

YSA, sınıflandırma, kümeleme ve regresyon gibi çeşitli şekillerde kullanılır. İnsan beyni karmaşık yapıdadır, onlara benzeterek bir algoritmanın geliştirilmesi çok zor gibi görünse de şuan YPA bir çok alanda kullanılmaktadır. Kullanım alanlarının bazıları şu şekilde sıralanabilir:

- Görüntü ve ses tanıma,
- Tahminleme (Hava Durumu)
- Tanıma Sistemleri (Parmak izi)
- Tıp Alanı
- Finans.

2.3.4.1. Yapay Sinir Ağının Ana Öğeleri



Yapay Sinir Ağı

Şekil 2. Yapay Sinir Ağı Yapısı

Kaynak: Altunbaş, C. (2020, 13 Şubat). *Yapay sinir ağları*. 14 Nisan 2024 tarihinde <https://medium.com/türkiye/yapay-sinir-ağları-ysa-caec1cc0bf9f> adresinden edinilmiştir.

Bir YPA birçok nörondan oluşur. 3 ağ veya daha fazla katmandan oluşur. Ağda bir girdi, bir veya birden fazla gizli girdi ve bir çıktı katmanı bulunur. Girdi katmanında ağa girdi bilgisini veren girdi kümesi bulunur. Girdi katmanı ile çıktı katmanı arasında birden fazla ara katman bulunabilir. Bir katmandaki her nöron bir üst

katmanda ki nörona çeşitli ağırlıklar ile bağlıdır. Girdi katmanındaki her bir veri ağırlık değeriyle çarpılarak gizli katmana iletilir. Gizli katmanda bir transfer fonksiyonundan geçirilir ve çıktı katmanına gönderilir. Gizli katmanda 0 ile 1 arasında değişen çıktı elde etmek için transfer fonksiyonu olarak doğrusal sigmoid fonksiyonu kullanılabilir (Kutlu ve Badur, 2009, s. 65).

Yapay sinir ağının süreç elemanları 5 tanedir. Bunlar:

- Girdiler
- Ağırlıklar
- Toplama Fonksiyonu
- Aktivasyon Fonksiyonu
- Çıktı

Girdiler: Dış dünyadan nöronlara gelen verilerdir. Bu veriler ağın öğrenmesi istenen bilgilerdir. YSA, Biyolojik sinir sistemini taklit ederek bilgileri nöron çekirdeklerine iletir.

Ağırlıklar: Bilgini nöron üzerindeki etkisi ve problem için önemini ifade eder. Bilgiler çekirdeğe iletilmeden ağırlık katsayılarıyla çarpılarak iletilir. Eğer ağırlık sıfır ise nöron üzerinde bir etki yaratmaz.

Toplama Fonksiyonu: Nöronlara gelen tüm girdiler ağırlıklar ile çarpıp toplanarak net bir girdiye ulaşılır. Bunu yaparken toplama fonksiyonları kullanılır.

Aktivasyon Fonksiyonu: Hücrelerdeki net girdiyi belirledikten sonra aktivasyon kodu ile çıkışa sonuç iletilir.

Çıktı: Aktivasyon sonucunun sonunda dış dünyaya verilen sonuçtur. Hücrelerin birden fazla girdisi olabilir ama sadece bir tane çıktısı vardır.

2.3.4.2. Yapay Sinir Ağlarında Öğrenme

YPA da dış dünyadan sisteme girdiler verilir. Girdiler sistemde işlenir. Sistemde işlenen veriler ve aktivasyon fonksiyonu ile çıkış verileri üretilir. Sistemden üretilen çıktılar ve eğitim sitemindeki sonuçlar karşılaştırılarak hata bulunur. Sistemin doğru çıktı üretmesi için ağırlıklar güncellenir. Sistem doğru çıktı verene kadar bu şekilde devam eder. Bu süreçte geçen zamana ise öğrenme denir.

Eđitim setinde ıktılar dođru ıktıktan sonra sisteme ıktısı verilmeyen test setleri ile denemeler devam eder. Sistemin ne kadarının eđitim ne kadarının test verisi olacađına dair bir kural yoktur.

YPA’da đrenme řu řekilde ilerlemektedir.

1. Ađın đrenmesi istenen veriler problem alanını kapsayacak řekilde bulunmalıdır ve eđitim setindeki verileri ađın đrenmesi sađlanmalıdır. Ađ problemi đrendikten sonra test verisiyle ađın performansı llr. Daha nce karřılařmadıđı veriler ile karřılařtıđında ađın verdiđi sonular, ađın performansını ortaya ıkarır.
2. Problem zerinde test edilecek ađın temel yapısı belirlenir. Girdi, ara katman, ara katmandaki hcre ve ıktı belirlenir.
3. Aktivasyon ve toplama fonksiyonları, đrenme ve momentum katsayısı gibi parametreler belirlenir.
4. Nronların ađırlık deđerlerinin ve eřik deđerinin bařlangı deđerleri belirlenir. Bu deđerler ile ilgili bir kural yoktur. Rastgele belirlenir ve sistemin hata deđerlerine gre gncellenir.
5. YPA’ya eđitim seti girdi olarak verilir.
6. Verilen girdiler sonrası ıktılar hesaplanır.
7. Eđitim seti deđerlerinde oluřan ıktılar sonrasında hata sonucu belirlenir.
8. Hatanın azalması iin ađırlık deđerleri gncellenir.
9. Bu iřlem ađ đrenmeyi tamamlanıncaya kadar devam eder.

Bazı durumlarda hata istenen seviyenin altına inmez. Bu durumlarda ađın bařlangı deđerleri, eđitim ve veri seti kontrol edilmelidir.

YPA’da en ok uygulanan đrenme kurallarından bazıları řunlardır:

- Hebb kuralı,
- Hopfield kuralı,
- Delta kuralı,
- Eđimli azalan kuralı,
- Kohonen kuralı (Yılmaz, 2022, s. 77).

Hebb Kuralı: Donald Hebb tarafından ortaya konulmuştur. Hebb kuralına göre eğer bir hücre başka bir hücreden giriş alıyorsa ve bu iki hücre matematiksel olarak aynı işaretle ise hücreler arasında kuvvetlendirme yapılmalıdır.

Hopfield Kuralı: Hopfield kuralı, Hebb kuralına benzemektedir. İki kuralın arasındaki fark şu şekildedir: İki hücre arasındaki kuvvetlendirmenin mantığının nasıl belirlendiğidir.

Delta Kuralı: Hebb kuralından sonra ortaya çıkmıştır ve problemlerde en çok kullanılan yöntemlerden biridir. Bu kuralın mantığı, oluşturulan ağ ile beklenen ağ arasındaki farkı azaltmak için bağlantı kuvvetlerinin yeniden yapılandırılmasında yatmaktadır. Ağırlık ayarlamasının amacı ağ hatası miktarını azaltmak, yani ağın toplam hatasını düzeltmektir. Hata bir önceki katmana yayılır ve bu noktaya ulaşıncaya kadar bu şekilde devam eder.

Eğimli Azalan Kuralı: Delta kuralına benzer. Eğimli azalmanın amacı minimum noktayı bulmaktır. Bu kural sabit bir noktaya yakın olsa da en sık kullanılan kuraldır.

Kohonen Kuralı: Biyoloji sistemlerinden esinlenerek oluşturulmuştur. Kohonen öğrenme kuralında hücreler kazanmak için birbirleriyle yarışır. Bu yarışta en büyük çıktıya sahip hücre kazanan olarak belirlenir ve bu hücrenin komşularının ağırlıkları güncellenir.

2.3.4.3. Yapay Yapay Sinir Ağı Modelleri ve Yapıları

YPA'da hücre elemanları birbirlerine farklı şekillerde bağlanabilmektedir. Bağlantıların akışına göre ileri ve geri beslemeli ağlar mevcuttur.

Sinir Ağı: Sinir ağların girdileri ve çıktıları var. Doğrusal olmayan problemleri çözemezler. Bu nedenle karmaşık problemlerde gizli katmanın bulunması gerekir. Bir katmandaki sinirler ile bir sonraki katmandaki sinirler arasında bağlantı vardır ancak katmandaki sinirler arasında bağlantı yoktur.

2.3.4.4. Yapay Sinir Ağının Katmanları

YPA'nın; ağ mimarisinin sinirler katmanında mevcuttur. Giriş katmanına dış dünyadan yada başka bir ağdan bilgiler verilmektedir. Giriş katmanındaki veriler diğer katmanlara gönderilir ve çıkış katmanından veri oluşur. Çok katmanlı bir ay yapısı mevcut ise giriş ve çıkış katmanlarının arasında gizli katmanlar mevcuttur. Gizli katman sayısı ağın performansının en iyi çalışabilecek şekilde belirlenmelidir.

Katmanlar arasında birden farklı bağlantı türleri bulunmaktadır. Bunlar;

- Tam bağlantılı katmanlar,
- Kısmi bağlantılar,
- İleri besleme,
- Çift yönlü bağlantılardır.

Tam bağlantılı: Katmandaki her bir hücre bir sonraki katmanın her sinirine bağlanmış ise bu bağlantı türü tam bağlantılı türdür.

Kısmi bağlantılar: Katmandaki hücrelerden hepsi değil de bir kısmı diğer katmanın sinirlerine bağlı ise bu türe kısmi bağlantılar denir.

İleri besleme: Katmandaki hücreler diğer katmandaki hücrelere çıktı gönderirler ancak bu hücreden geri bir giriş alamazlar. Bu tür ileri besleme türüdür.

Çift yönlü bağlantılar: Katmandaki hücrelerin çıktılarının önceki katmanın hücrelerine bağlayan ağ türüne çift yönlü bağlantıdır.

İleri besleme yada çift yönlü ağların içerisinde tam bağlantı yada kısmi bağlantı olma ihtimali vardır.

2.3.4.5. Tek Katmanlı Sinir Ağları - Perceptron

Bu sinir ağlarında sadece girdi ve çıktılar mevcuttur. Karmaşık problemleri çözme yeteneğine sahip değildirler. Tek Katmanlı Sinir Ağları, Rosentblatt tarafından önerilmiştir. Tek katmanlı sinir ağları, problemlerin çözümü için yetersizdir ancak geri yapılma vb. algoritmalar ile çalışılabilir. Bu açıdan önemlidir. Tek katmanlı sinir ağlarında diğer önemli yapılardan biri eşik değeridir. Eşik değeri, problem modeli

sahibi tarafından belirlenir. Eşik değeri ağın iyi bir sınıflandırma yapmasına yardımcı olur.

Tek katmanlı sinir ağlarında giriş ve çıkış değerleri toplama fonksiyonu ile işleme alınır. İşlemden çıkan sonuç işe eşik değeri karşılaştırılır. Toplam değer eşik değerinden büyük ise sonuç +1, eşit ve küçük olma durumunda ise sonuç -1 olarak belirtilir. Tek katmanlı sinir ağlarında sonuç iki farklı durum için sınıflandırılır.

Tek katmanlı sinir ağları doğrusal olan problemleri çözebilir ancak XOR problemine benzer karmaşık problemleri çözemez. Bu gibi problemler çok katmanlı sinir ağları ile çözülebilir. Tek katmanlı sinir ağları, karmaşık problemleri çözemesinden dolayı YPA çalışmaları duraksamıştır. Bu problemler çok katmanlı sistemlerin ortaya çıkmasıyla çözülmüştür.

Tek katmanlı sinir ağları – Perceptron modelinin algoritması aşağıdaki gibidir.

Adım 1: Tüm öğrenme parametrelerine ve ağırlıklara ilk değer atanır.

$w_i = 0$, $b = 0$ ve μ (öğrenme katsayısı) ($0 < \mu < 1$) başlat.

Adım 2: Belirli bir iterasyon sayısı kadar 3-7 adım aralığı tekrarlanır.

Adım 3: Her girdi(x) ve ilgili girdiye (x) karşılık beklenen sonuç (y) için 4-6 adım aralıkları tekrarlanır.

Adım 4: Girdi değerleri eğitim setinden alınır. ($x_i = k_i$)

Adım 5: Perceptrona gelen toplam sinyal hesaplanır.

Aktivasyon Fonksiyonu sonucu hesaplanır.

Adım 6: Aktivasyon sonucu üzerinden hata kontrolü yapılır. Eğer hata varsa; hata değerine göre ağırlıklar güncellenir.

Eğer ($y \neq t$) ise

$$w_i(\text{yeni}) = w_i(\text{eski}) + \mu * t * x_i$$

$$b(\text{yeni}) = b(\text{eski}) + \mu * t$$

Eğer ($y = t$) ise hata bulunmamaktadır. Güncelleme yapılmaz.

Adım 7: Kontrol et.

Tek katmanlı YPA da öğrenme hata üzerine kurulmuştur. Eğer sistemde hata çıkıyorsa bu öğrenme devam etmekte anlamına gelir. Sistem öğrenerek hatayı en aza indirmeyi hedefler.

2.3.4.6. Çok Katmanlı Sinir Ağları

Tek katmanlı YPA karmaşık problemleri çözemez. Bu model Rumelhart ve arkadaşları tarafından geliştirilmiştir. Geliştirildiğine karmaşık problemler sebebi ile duraksama dönemine giren YPA tekrar gelişmeye başlamıştır. Çok katmanlı yapay sinir ağları genellikle delta öğrenme kuralını kullanır. Bu kuralı Adaline geliştirmiştir.

Çok katmanlı YPA'nın da amacı hatayı en aza indirmektedir. Bu ağlara öğrenme için girdiler ve bu girdiler sonunda üretilmesi gereken veriler iletilir. Amaç her bir girdi için en uygun olan çıktıyı üretmektir. Ağa veriler sunulduktan sonra ağın girdilere göre verdiği sonuçlar ile öğrettiğimiz sonuçlar karşılaştırılır. Sonuçlar arasındaki fark geriye doğru yansıtılır ve hata azalana kadar ağırlıklar değiştirilmeye devam edilir.

Çok katmanlı YPA girdi ve çıktı katmanları ve bu katmanların arasında iletişim sağlayan ara katmanlardan oluşur.

Girdi katmanı dışardan bilgiyi alır ve işlem yapmadan ara diğer adıyla gizli katmana gönderir. Girdi katmanına birden fazla bilgi gelebilir. Bu katman kendin sonraki katmanın tüm hücreleriyle bağlantılıdır.

Ara (Gizli) katmana gelen veriler işlenir. İşlendikten sonra çıktı katmanına veya ağ mimarisine göre başka bir ara katmana gönderilir. Ağ mimarisine göre çok katmanlı YPA'da birden fazla ara katman olabilir. Ara katmanlar içerisinde bir veya birden fazla işlem yapılabilir.

Çıkış katmanı kendisine ara katman tarafından işlenip gönderilen verileri işler ve bir çıktı üretir. Bu katmana birden fazla değer gelebilir veya katman içerisinde birden fazla işlem yapılabilir ancak sistem tek bir çıktı üretir.

Çok katmanlı yapay sinir ağlarında girdi ve çıktıların olduğu veri setine eğitim seti denir. Bu ağlarda genellikle danışmanlı öğrenme kullanılır. Çok katmanlı sinir ağlarında delta öğrenme kuralı kullanılır. İleri yönde hesaplama yöntemi ile sistemin

çıktısı, geriye doğru hesaplama yöntemi ile ise ağırlıklar hesaplanır ve güncellenir. Sinir ağı öğretilen ve verilen örnekler sonucunda problem uzayını temsilen bir çözüm uzayı sunar.

Çok katmanlı YPA şu şekilde çalışmaktadır;

1. Öğrenilmesi hedeflenen problem ile ilgili bütün bilgi ve durumların sonuçlar bulunur. Eğitim ve test seti olarak iki ayrılır. Öğrenme gerçekleşirken sadece eğitim seti gösterilir.
2. Sinir ağının mimari yapısı belirlenir.
3. Toplama ve aktivasyon fonksiyonu, momentum ve öğrenme katsayısı belirlenir.
4. Ağırlıkların başlangıç değeri verilir.
5. Eğitim setindeki veriler ağına gönderilir.
6. İleri yönde hesaplama yapılır.
7. Çıkması beklenen sonuç ve ağına çıktı karşılaştırılır ve hata belirlenir.
8. Geriye doğru hesaplama yapılır ve ağırlıkların değeri tekrar belirlenir.
9. Hata sonucu istenen düzeye geldi ise eğitim sonlanır; gelmedi ise ağırlıkların değerleri güncellenir ve öğrenme devam eder (Adım 5'e dönlür.)

2.3.4.7. İleri Beslemeli Ağlar

İleri beslemeli ağlarda elemanlar katmanlara bölünmüştür. Girdi katmanından çıktı katmanına tek yönlü iletişim sinyali vardır. Katmandaki nöronlar ile bir sonraki katmandaki bütün nöronlara bağlantı mevcuttur ama aynı katman içinde bulunan nöronlar arasında bağlantı yoktur. Çok katmanlı ağlar; ileri beslemeli ağlara örnektir.

İleri beslemeli ağların içerisindeki katmanlardaki hücrelerdeki çıkışlar kendinden sonraki katmanın girişidir. Başka bir deyişle dışardan ağlara verdiğimiz veriler giriş katmanında herhangi bir işleme yapılmadan ara katmana gönderilir. Ara katmanda ve çıktı katmanında işlemler yapılarak çıktı üretilir.

İleri besleme ağlar adını bilgi akışının sürekli ileriye doğru olmasından alır.

İleri beslemeli ağlarda giriş katmanında işlem olmamaktadır. Giriş katmanındaki hücre sayısı belirlenmesi problem için belirlenen modelin giriş sayısı ile

ortaya çıkar. Ara katman ve ara katmandaki hücre sayısı ise problemin yapısına göre deneme-yanılma ile belirlenir. Çıkış katmanındaki hücre sayısı ise yine problemi modellerken belirlenen çıktı sayısına göre belirlenir.

İleri beslemeli ağlar, delta öğrenme kuralını kullanır.

2.3.4.8. Geri Beslemeli Ağlar

Bu ağlar danışmansız (gözetimsiz) öğrenme ile çözülebilecek problemler için uygundur. Geri beslemeli ağlar içerisinde geri besleme işlemi yapılır. Başka bir deyişle çıkış ve ara katmanlar; kendinden bir önceki katmana veri gönderir. Böylece ağ yapısında geriye doğru akış ile geri yönde hesaplama yapılabilir.

Sinir ağının yapısında hem çıkışın hem de bir önceki değerin tutulması sebebi ile dinamik hafız yapıları vardır. Bu özellikleri ile tahmin problemleri için kullanılır.

Bu ağlardaki hücrelerdeki her bir çıktı, kendisinin yada diğer hücrelerden birinin girdisidir. Büyük bir çoğunlukla geciktirme elemanı olarak geçen eleman ile geri besleme işlemi yapılır. Bu işlem aynı katman veya farklı katmanlarda bulunan hücreler arasında uygulanır. Ağ yapısı dinamik ve doğrusal özelliği göstermeyen ağlarda, modellemeye göre farklı davranışlar gösterebilir.

Bu ağlarda, geriye doğru hesaplama yapabilmek için sonuçta hedeflenen değer ve ağın verdiği çıktı karşılaştırılıp fark bulunur. Bu farka hata denir. Bu ağın amacı da hatayı en düşüğe indirmektedir. Ağın verdiği çıktılar sonucu toplanır ve toplam hata ortaya çıkarılır.

Toplam hata alınırken ağın verdiği sonuçların negatif olma olasılığı düşünülerek ağırlık değerlerinin kareleri toplanır ve toplam sonucunun karekökü alınır. Toplam hatayı en düşük değere indirmek için hatayı neden olan hücrelerin bulunup hatanın bu hücreleri dağıtılması şarttır. Bu sebeple hücreler için belirlenen ağırlık değerleri güncellenir.

2.3.4.9. Yapay Sinir Ağı'nın Çalışması, Öğrenmesi ve Test Edilmesi

YPA'da temel amaç ağı sunduğumuz girdilerin işlenerek çıktıya çevrilmesidir. Ağı doğru çıktıları üretebilmesi için ona sunulan girdilerin istenilen şekle gelmesi gerekir. Bu sebeple ağı sunulan örnekler önce vektör haline getirilir.

Ağı sunulan eğitim setindeki örnekler vektör haline getirilir. Vektör olarak sunulan girdilerin çıktı olarak da vektör üretmeleri beklenir. Hem girdi hem çıktıların doğru olması için ağı parametrelerinin aldığı değerlerde bu şekilde düzenlenmelidir. Girdilerin ve çıktıların vektör şeklinde tasarlayacak kişi ağ mimarisini geliştiren kişidir.

Tüm bağlantılara verilen verilen ağırlık değerlerinin en iyi değerinin bulunması ağı öğrenmesine yol açar. Ağırlık değerleri ilk olarak rastgele verilir. Örnekler sunuldukça bu değer güncellenerek en iyi hale gelir.

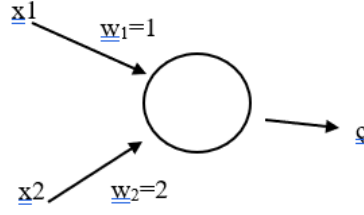
Ağırlık değerleri için en iyi değer bulunduğu YPA diğer örnekler için de geneleme yapabilir. Böylece öğrenme sağlanır. Ağırlık değerleri bazı öğrenme kurallarına göre güncellenir. Eğer ağ üzerindeki örneklerden bazı bilgileri alırsak bu durumda ağ yapısı örneklere yorum getirerek sonuç verir. Bu duruma adaptif öğrenme de denir.

YPA'da öğrenme iki aşamada gerçekleşir. Bunlar şu şekildedir.

1. Aşama: Ağı örnekler gösterilerek verilecek çıkışlar belirlenir.
2. Aşama: Ağırlık değerleri güncellenir.

Bu iki aşama tamamlanıp öğrenme gerçekleştikten sonra ağı bazı örnekler gösterilerek sonuçlar test edilir. Test için kullanılan örnekleri daha önce ağı görmemiş olmaması gerekmektedir. Ağ test edilirken herhangi bir ağırlık değeri güncellenmez.

Örnek Yapay Sinir Ağı



Şekil 3. YPA

Kaynak: Yazar tarafından oluşturulmuştur.

Şekilde gösterilen iki girişli ve tek katmanlı bir YPA'nın nasıl probleminin çözüleceğini görelim. Problemden eğitim setine ait 1. örnekte $(x_1, x_2) = (1, 0)$ girdisine karşılık beklenen değer (t_1) 1 iken 2. örnek için $(x_1, x_2) = (0, 1)$ girdisine karşılık beklenen değer (t_2) 0 olarak verilmektedir. Eşik değeri (b) 1; öğrenme katsayısı (μ) ise 0.5 olarak soruda verilmektedir.

1. Adım : $NET = w_1 * x_1 + w_2 * x_2 = 1 * 1 + 2 * 0 = 1$

$NET = 1$; $b = 1$ $NET \geq b \rightarrow \hat{c} = 1 \rightarrow \hat{c} = t_1$ olduğundan ötürü ağırlık güncellenmez.

2. Adım : $NET = w_1 * x_1 + w_2 * x_2 = 1 * 0 + 2 * 1 = 2$

$NET = 2$; $b = 1$ $NET \geq b \rightarrow \hat{c} = 1 \rightarrow \hat{c} \neq t_2$ olduğundan ötürü ağırlık güncellenir.

3. Adım: $NET = w_1 * x_1 + w_2 * x_2 = 1 * 1 + 0 * 1,5 = 1$

$NET = 1$; $b = 1$ $NET \geq b \rightarrow \hat{c} = 1 \rightarrow \hat{c} = t_1$ olduğundan ötürü ağırlık güncellenmez.

4. Adım: $NET = w_1 * x_1 + w_2 * x_2 = 1 * 0 + 1,5 * 1 = 1,5$

$NET = 1,5$; $b = 1$ $NET \geq b \rightarrow \hat{c} = 1 \rightarrow \hat{c} \neq t_2$ olduğundan ötürü ağırlık güncellenir.

2.4. Korelasyon ve Öznitelik Belirleme

Korelasyon analizi, değişkenler arasındaki ilişki, bu ilişkinin yönü ve şiddeti hakkında bilgi sağlayan istatistiksel bir yöntemdir.

Korelasyon değeri -1 ile 1 arasındadır. 0 değeri değişkenler arasında herhangi bir ilişkinin olmadığı anlamına gelir. Bu değer -1'e doğru ilerledikçe 'negatif mükemmel korelasyonu', +1'e doğru ilerledikçe 'pozitif mükemmel korelasyonu' gösterir.

$(-0,5)$ ile (0) ve (0) ile $(0,5)$ arasındaki değerler, yöne bağlı olarak pozitif veya negatif olarak düşük korelasyon değerleri olarak adlandırılır. Değişkenlerin bir arada gitmediğini düşündüğümüz alandır burası.

$(-0,9)$ - $(-0,5)$ ve $(0,5)$ - $(0,9)$ ve -1 ile 1 aralıklarında pozitif veya negatif yüksek korelasyon değerleri gözlenmektedir. Bunlar değişkenler arasında güçlü bir ilişki olduğunu söylediğimiz kategorilerdir. Yüksek korelasyon durumunda değişkenler birbirleriyle etkileşim halindedir.

Korelasyon yapılarak bağların yüksek olduğu değerler bulunabilir. Bu değerler verilerimizin öznitelikleridir.

3. DENEYSEL ÇALIŞMALAR

Bu bölümde tez çalışmasında kullanılan veri seti ve uygulanan deneysel çalışmalara yer verilmiştir.

3.1. Veri Seti

Veri seti Çalışmada “Kaggle Students Adaptability Level in Online Education” veri seti kullanılmıştır, Veri seti CSV formatındaki metin bilgilerinden oluşmaktadır. Veri setinde 1205 adet kız ve erkek öğrencilerin yaş, eğitim seviyesi, sınıfı ve online eğitime uyumlanıp uyumlanmadığını içeren bilgiler bulunmaktadır.

Veri setinde 8 sütun 1205 satır bulunmaktadır.

Veri setinde 1205 adet kız ve erkek öğrencilerin yaş, eğitim seviyesi, sınıfı ve online eğitime uyumlanıp uyumlanmadığını içeren bilgiler bulunmaktadır.

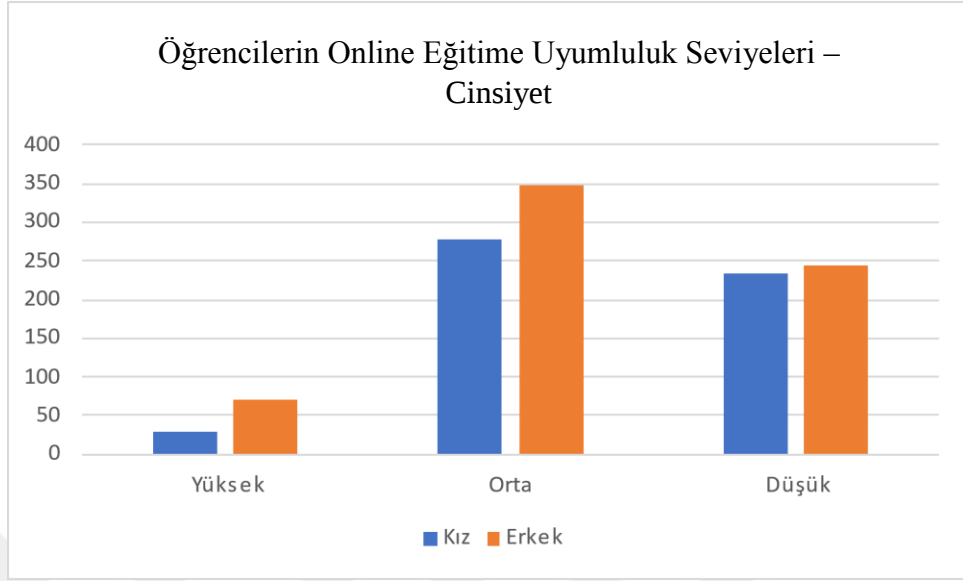
Bu veri setinde yüksek uyumlu olan grup 413 kişiyle erkek öğrencilerdir. Erkek öğrencilerin arasında ise 21-25 yaş arası özel okulda okuyan üniversite öğrencilerinin uyumluluk seviyesi yüksektir. Bu verileri daha da özelleştirdiğimizde ise ekonomik durumu düşük, derse giriş aracı olarak bilgisayar ve internet bağlantısı olarak wi-fi, network tipi olarak 4G kullanan grup en yüksek uyumluluğa sahiptir.

Düşük derecede uyumlu olan grup 345 kişiyle erkek öğrencilerdir. Erkek öğrencilerin arasında ise 21-25 yaş arası devlet bağlı okulda okuyan üniversite öğrencileri düşük derecede uyumludur. Bu verileri daha da özelleştirdiğimizde ise ekonomik durumu standart, derse giriş aracı olarak telefon ve internet bağlantısı olarak mobil veri, network tipi olarak 3G kullanan grup düşük derece uyumluluğa sahiptir.

Orta derecede uyumlu olan grup 447 kişiyle erkek öğrencilerdir. Erkek öğrencilerin arasında ise 21-25 yaş arası özel okulda okuyan üniversite öğrencileri orta derecede uyumludur. Bu verileri daha da özelleştirdiğimizde ise ekonomik durumu standart, derse giriş aracı olarak bilgisayar ve internet bağlantısı olarak wi-fi, network tipi olarak 4G kullanan grup orta derece uyumluluğa sahiptir.

Veri setindeki tüm değerleri ayrı incelediğimizde aşağıdaki tablolar ve sonuçlar ortaya çıkmaktadır.

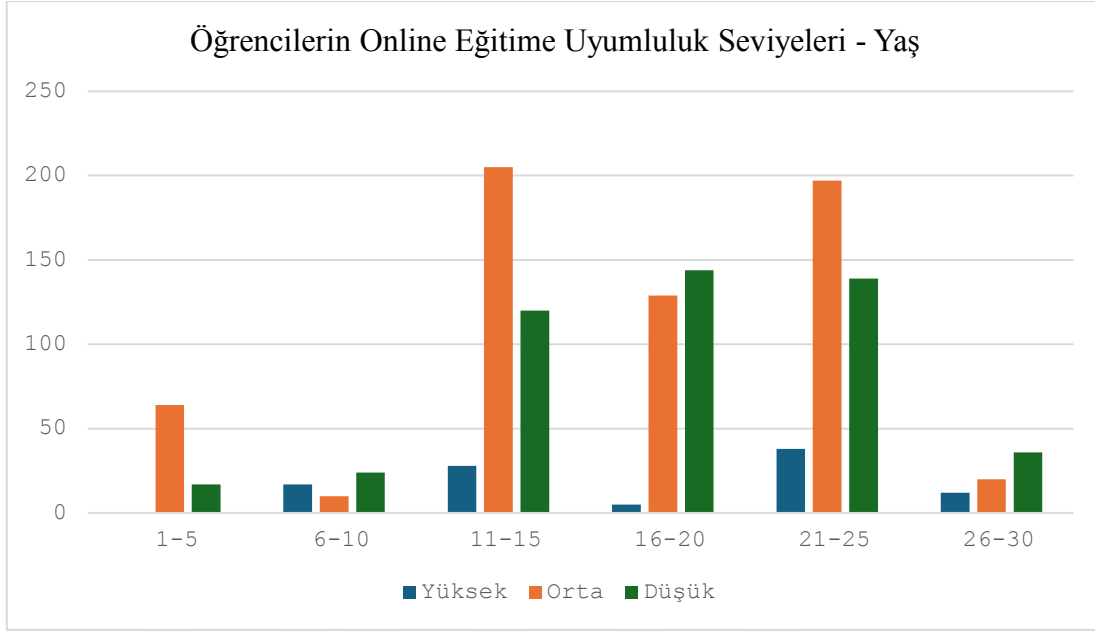
Tablo 4. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Cinsiyet



Kaynak: Yazar tarafından oluşturulmuştur.

663 erkek öğrencinin 71'inin online eğitim uyumluluk seviyesi yüksek, 245'inin düşük, 347'sinin ılımlı; 542 kız öğrencinin 29'unun online eğitim uyumluluk seviyesi yüksek, 235'inin düşük, 278'inin ise ılımlıdır.

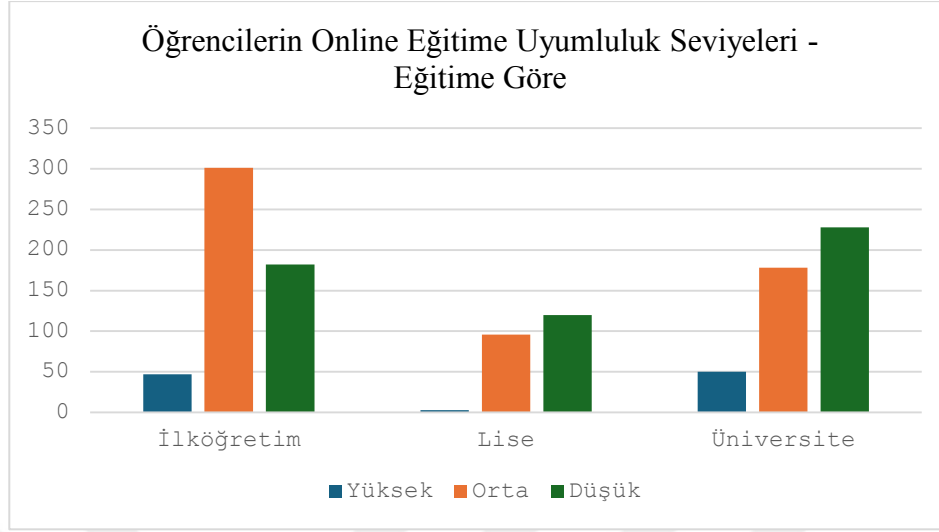
Tablo 5. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Yaş



Kaynak: Yazar tarafından oluşturulmuştur.

353 11-15 yaş aralığında öğrencinin 28'inin online eğitim uyumluluk seviyesi yüksek, 120'sinin düşük, 205'inin ılımlı; 81 1-5 yaş aralığında öğrencilerin 17'inin online eğitim uyumluluk seviyesi yüksek, 64'ünün düşük; 278 16-20 yaş aralığında öğrencilerin 5'inin online eğitim uyumluluk seviyesi yüksek, 144'ünün düşük, 129'u ılımlı; 374 21-25 yaş aralığında öğrencilerin 38'inin online eğitim uyumluluk seviyesi yüksek, 139'unun düşük, 197'inin ılımlı; 68 26-30 yaş aralığında öğrencilerin 12'sinin online eğitim uyumluluk seviyesi yüksek, 36'sının düşük, 20'sinin ılımlı; 51 6-10 yaş aralığında öğrencilerin 17'sinin online eğitim uyumluluk seviyesi yüksek, 24'ünün düşük, 10'un ılımlı olduğunu göstermektedir.

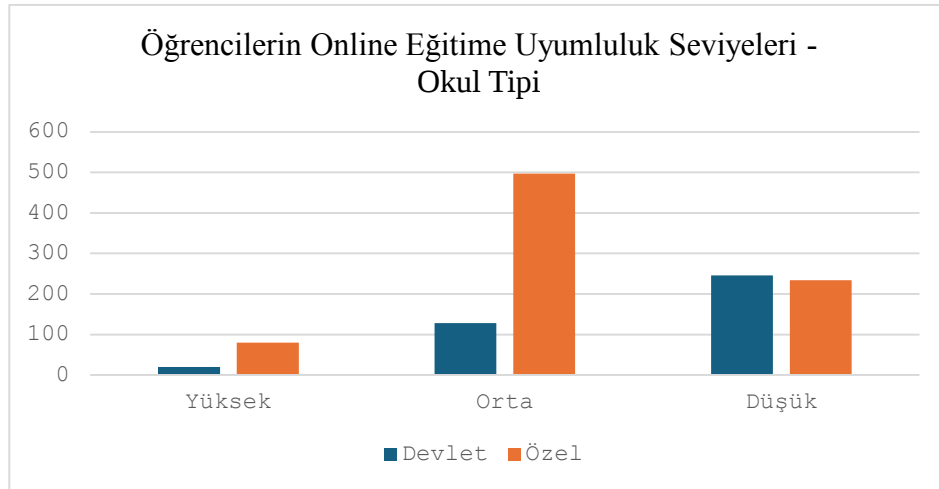
Tablo 6. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Eğitim Düzeyi



Kaynak: Yazar tarafından oluşturulmuştur.

219 lise öğrencisinin 3'ünün online eğitim uyumluluk seviyesi yüksek, 120'sinin düşük, 96'sının ılımlı; 530 ilköğretim öğrencisinin 47'sinin online eğitim uyumluluk seviyesi yüksek, 182'sinin düşük, 301'inin ılımlı; 456 üniversite öğrencisinin 50'sinin online eğitim uyumluluk seviyesi yüksek, 178'sinin düşük, 228'inin ılımlı olduğunu göstermektedir.

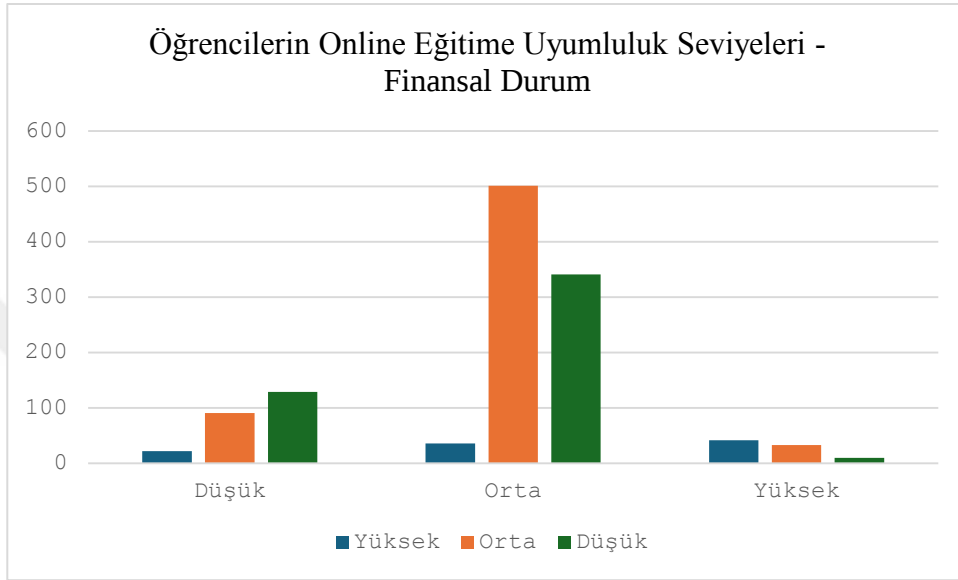
Tablo 7. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Okul Tipi



Kaynak: Yazar tarafından oluşturulmuştur.

382 devlet okulu öğrencisinin 20'sinin online eğitim uyumluluk seviyesi yüksek, 234'ünün düşük, 128'inin ılımlı; 823 özel okulu öğrencisinin 80'inin online eğitim uyumluluk seviyesi yüksek, 246'sının düşük, 497'sinin ılımlı olduğunu göstermektedir.

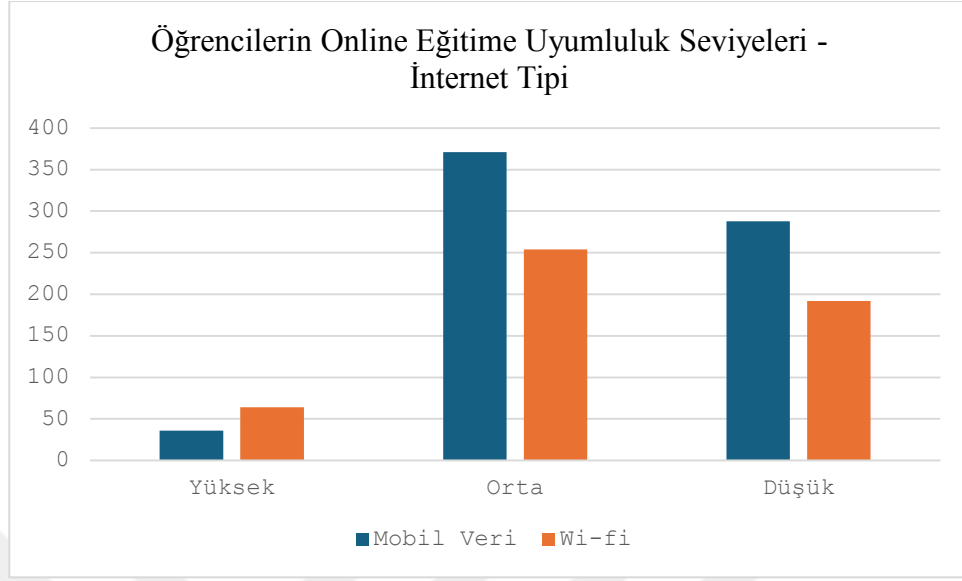
Tablo 8. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Finansal Durum



Kaynak: Yazar tarafından oluşturulmuştur.

Finansal olarak ortalama bütçeye sahip olan 878 öğrencinin 36'sının online eğitim uyumluluk seviyesi yüksek, 341'inin düşük, 501'inin ılımlı; Finansal olarak düşük bir bütçeye sahip olan 242 öğrencinin 22'sinin online eğitim uyumluluk seviyesi yüksek, 129'unun düşük, 91'inin ılımlı; Finansal olarak yüksek bir bütçeye sahip olan 85 öğrencinin 42'sinin online eğitim uyumluluk seviyesi yüksek, 10'unun düşük, 33'ünün ılımlı olduğunu göstermektedir.

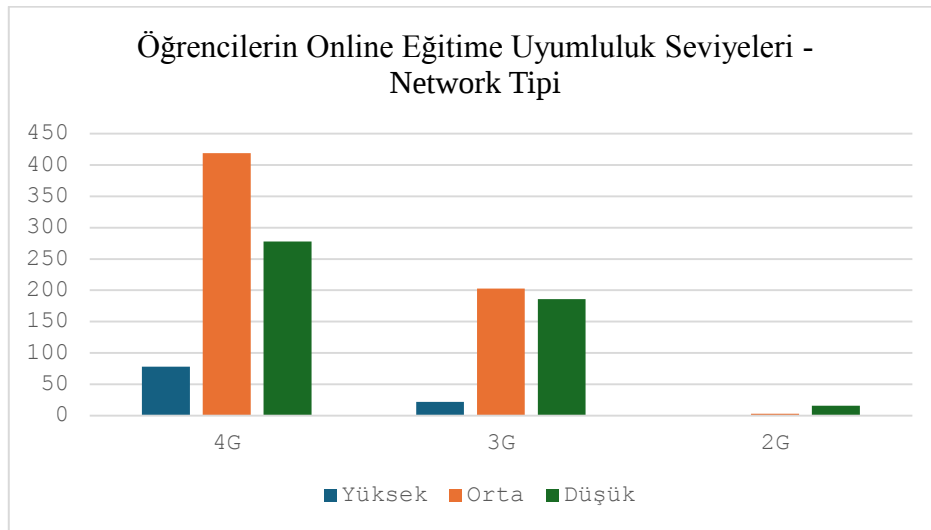
Tablo 9. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – İnternet Tipi



Kaynak: Yazar tarafından oluşturulmuştur.

Mobil veri kullanan 695 öğrencinin 36'sının online eğitim uyumluluk seviyesi yüksek, 288'inin düşük, 371'inin ılımlı; Wifi kullanan 510 öğrencinin 64'ünün online eğitim uyumluluk seviyesi yüksek, 192'sinin düşük, 254'ünün ılımlı; olduğunu göstermektedir.

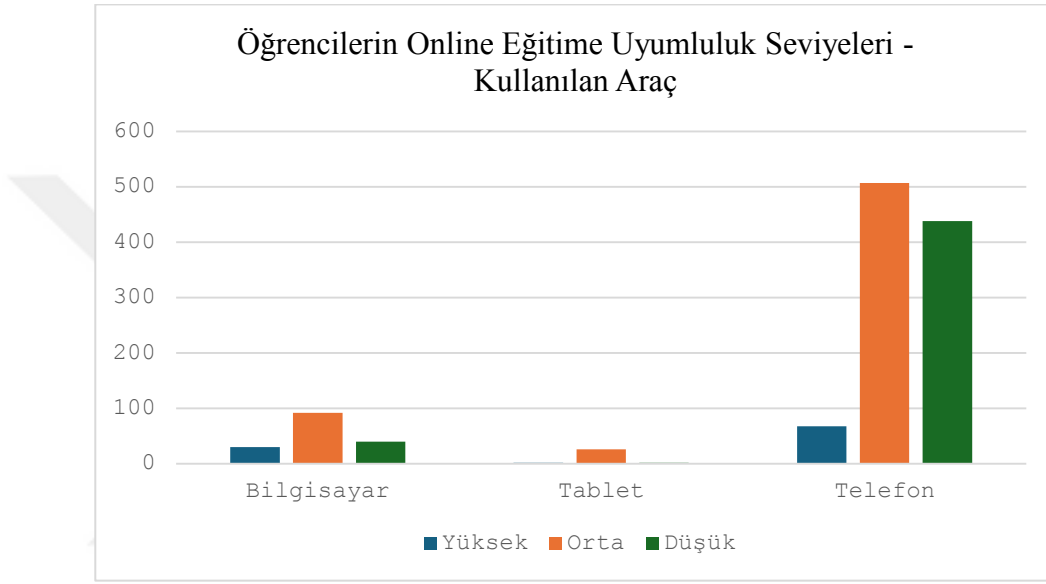
Tablo 10. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Network Tipi



Kaynak: Yazar tarafından oluşturulmuştur.

2G kullanan 19 öğrencinin 16'sının online eğitim uyumluluk seviyesi yüksek, 3'ünün düşük; 3G kullanan 411 öğrencinin 22'sinin online eğitim uyumluluk seviyesi yüksek, 186'sının düşük, 203'ünün ılımlı; 4G kullanan 775 öğrencinin 78'inin online eğitim uyumluluk seviyesi yüksek, 278'inin düşük, 419'ının ılımlı olduğunu göstermektedir.

Tablo 11. Öğrencilerin Online Eğitime Uyumluluk Seviyeleri – Kullanılan Araç



Kaynak: Yazar tarafından oluşturulmuştur.

Online derslere girmek için bilgisayar kullanan 162 öğrencinin 30'unun online eğitim uyumluluk seviyesi yüksek, 40'ının düşük, 92'sinin ılımlı; online derslere girmek için telefon kullanan 1013 öğrencinin 68'inin online eğitim uyumluluk seviyesi yüksek, 438'inin düşük, 507'sinin ılımlı; online derslere girmek için tablet kullanan 30 öğrencinin 2'sinin online eğitim uyumluluk seviyesi yüksek, 2'sinin düşük, 26'sının ılımlı; olduğunu göstermektedir.

Veri setinden kullanılan tüm veri setler string ve kategorik değerde olup tüm veriler Label Encoder ile sayısallaştırılmıştır.

3.2. Modelin Oluşturulması

Değişken sayısının fazla olması modelin başarı düzeyini etkileyen önemli faktörlerdendir. Modelde 9 değişken mevcuttur. Umum düzeyi değişkeni y bağımlı değişken olup diğer 8 değişken ise y bağımlı değişkene etkin eden X değişkenidir. Model tamamlandıktan sonra veri seti X değişkenlerine bağlı olarak y değişkenini tahmin edecektir.

Çalışma için sınıflandırma algoritmalarına ihtiyaç duyulacağı için 5 algoritma kullanılmıştır.

Algoritmaların modellenmesi aşamasında y değişkeni (Uyumlanma Düzwyi) veri setinden çıkartılmıştır. “X_train, X_test, y_train, y_test” parametreleri kullanılarak değişkenler test ve eğitim (train) olarak ayrılmıştır. Veri setinin % 20’si test için %80’i ise eğitim için belirlenmiştir. “Random_state” parametresi 42 olarak belirtilmiştir. Random_state parametresi kod her çalıştığında veri kümesini aynı şekilde bölmektedir.

Modeller aşağıdaki şekilde belirlenmiştir;

DecisionTreeClassifier, MultinomialNB, KerasClassifier, xgboost, KNeighborsClassifier.

Çalışma için Kullanılan Algoritmaların Analizleri

Çalışma için kullanılan yukarıda da belirtildiği üzere 5 adet denetimli algoritma olup bunlar;

K-NN (En Yakın Komşu), NB, YPA, Desicion Tree ve Bagged Trees’dir.

3.3. K-NN Algoritması Analizi

K-NN algoritmasında, KNeighborsClassifier kütüphanesinden yararlanılmıştır. Model “knn_model” ismi ile oluşturulmuştur.

Algoritmadaki komşu sayısı, “n_neighbors” parametresi kullanılarak 1 olarak belirlenmiştir. Bu değer “Cross-Validation Accuracy” parametresiyle belirlenmiştir. Modelde algoritma metriği olarak presicion, recall, f1-score, model eğitim süresi ve

tahmin süresi kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır. Bu modellerin hepsinde en iyi sonucu yukarıda belirtildiği üzere “Cross-Validation Accuracy” parametresiyle komşu sayısı 1 olarak belirlenmiştir.

Bu modellerin başarı performansı aşağıdaki tabloda belirtilmiştir.

Tablo 12. K-NN Modellerinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%75	%76	%76	%76	0.01	0.01
%70-%30	%75	%76	%75	%75	0.01	0.01
%80-%20	%73	%74	%73	%73	0.01	0.01
%90-%10	%70	%70	%70	%70	0.01	0.01

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere K-NN modellerinin başarı performansı sırasıyla şu şekildedir; %60-%40, %70-%30, %80-%20, %90-%10. Bu modellerden en iyi sonucu %60 eğitim seti, %40 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Bu algorithmada en iyi performansı gösteren model kullanılan diğer makine öğrenmesi modellerinden en iyi performansı gösteren modellerle de karşılaştırılacaktır.

En yüksek performansı veren %60 eğitim -%40 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %75, Presicion %76, Recall %76 ve f1-score %76 olarak belirlenmiştir.

3.4. Naive Bayes Algoritması Analizi

NB algoritmasında, veriler kategorik değere sahip olduğu için MultinomialNB kütüphanesinden yararlanılmıştır. Model “nb_model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak presicion, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 13. Naive Bayes Modellerinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%62	%63	%63	%60	0.0030	0.0010
%70-%30	%62	%63	%62	%60	0.0028	0.0010
%80-%20	%59	%61	%59	%57	0.0030	0.0010
%90-%10	%60	%62	%60	%58	0.0028	0.0071

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Naive Bayes modellerinin başarı performansı sırasıyla şu şekildedir; %70-%30, %60-%40, %90-%10, %80-%20. Bu modellerden en iyi sonucu %70 eğitim seti, %30 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Bu model kullanılan modeller arasında en düşük performansı gösteren modeldir. Naive Bayes yapı olarak daha basit problemleri çözme amacıyla kullanılan bir modeldir. Performans düşüklüğünün sebebi bundan kaynaklanmaktadır. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Bu algorithmada en iyi performansı gösteren model kullanılan diğer makine öğrenmesi modellerinden en iyi performansı gösteren modellerle de karşılaştırılacaktır.

En yüksek performansı veren %70 eğitim -%30 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %62, Presicion %63, Recall %62 ve f1-score %60 olarak belirlenmiştir.

3.5. Desicion Tree Algoritması Analizi

Desicion Tree algoritmasında, DecisionTreeClassifier kütüphanesinden yararlanılmıştır. Model “model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak presicion, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 14. Desicion Tree Modellerinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%76	%77	%77	%77	0.0020	0.0010
%70-%30	%78	%79	%79	%79	0.0075	0.0009
%80-%20	%74	%75	%75	%75	0.0015	0.0010
%90-%10	%76	%77	%77	%77	0.0020	0.0013

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Decision Tree modellerinin başarı performansı sırasıyla şu şekildedir; %70-%30, %60-%40, %90-%10, %80-%20. Bu modellerden en iyi sonucu %70 eğitim seti, %30 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Bu modelde kullanılan veri setlerinde başarı performanslarında çok fark olmamakla birlikte dengeli bir dağılım bulunmaktadır. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Bu algorithmada en iyi performansı gösteren model kullanılan diğer makine öğrenmesi modellerinden en iyi performansı gösteren modellerle de karşılaştırılacaktır.

En yüksek performansı veren %70 eğitim -%30 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %78, Presicion %77, Recall %77 ve f1-score %77 olarak belirlenmiştir.

3.6. Yapay Sinir Ağları Algoritması Analizi

YPA algoritmasında, tensorflow kütüphanesinden yararlanılmıştır. Model “model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak precision, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 15. Yapay Sinir Ağı Modellerinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%63	%63	%63	%63	0.0060	0.0009
%70-%30	%63	%64	%64	%63	0.0068	0.0010
%80-%20	%64	%65	%64	%64	0.0065	0.0010
%90-%10	%64	%66	%64	%64	0.0062	0.0009

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Yapay Sinir Ağı modellerinin başarı performansı sırasıyla şu şekildedir; %90-%10, %80-%20, %70-%30, %60-%40. Bu modellerden en iyi sonucu %90 eğitim seti, %10 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Bu modelde kullanılan veri setlerinde başarı performansı çok düşük olsa da dengeli bir dağılım bulunmaktadır. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Bu algorithmada en iyi performansı gösteren model kullanılan diğer makine öğrenmesi modellerinden en iyi performansı gösteren modellerle de karşılaştırılacaktır.

En yüksek performansı veren %90 eğitim -%10 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %64, Presicion %66, Recall %64 ve f1-score %64 olarak belirlenmiştir.

3.7. Bagging Trees Algoritması Analizi

Bagging Trees algoritmasında, DecisionTreeClassifier kütüphanesinden yararlanılmıştır. Model “model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak presicion, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 16. Bagging Trees Modellerinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%76	%76	%76	%76	0.09	0.01
%70-%30	%75	%75	%75	%75	0.09	0.01
%80-%20	%75	%75	%75	%75	0.09	0.01
%90-%10	%76	%76	%76	%76	0.12	0.01

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Bagging Tress modellerinin başarı performansı sırasıyla şu şekildedir; %90-%10, %60-%40, %70-%30, %80-%20. Bu modellerden en iyi sonucu %70 eğitim seti, %30 veri seti ve %60 eğitim seti, %40 veri seti kullanılarak oluşturulan model verse de diğer oranlar da da çok farklılık yoktur. Modelde kullanılan veri setlerinde başarı performansı çok yüksek olmasıyla birlikte dengeli bir dağılım bulunmaktadır. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur.

Bu algoritmada en iyi performansı gösteren model kullanılan diğer makine öğrenmesi modellerinden en iyi performansı gösteren modellerle de karşılaştırılacaktır.

En yüksek performansı veren %60 eğitim -%40 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %76, Presicion %76, Recall %76 ve f1-score %76 olarak belirlenmiştir.

3.8. Çalışmada Kullanılan Makine Öğrenmesi Modellerinin Karşılaştırılması

Çalışmada kullanılan Makine Öğrenmesi modelleri kendi içlerinde karşılaştırıldıktan sonra en iyi sonucu veren modellemeler diğer algoritmalarla karşılaştırılmıştır. Sonuçlar aşağıdaki tabloda paylaşılmıştır.

Tablo 17. Modellerin Başarı Karşılaştırmaları

Algoritma Adı (Eğitim Seti – Test Seti)	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
K-NN (%60- %40)	%75	%76	%76	%76	0.00	0.01
Naive Bayes (%70- %30)	%62	%63	%63	%60	0.0030	0.0010
Desicion Tree (%70- %30)	%78	%79	%79	%79	0.0075	0.0009
Bagging Trees (%60- %40)	%76	%76	%76	%76	0.09	0.0009
Yapay Sinir Ağları (%90- %10)	%64	%66	%64	%64	0.0062	0.0009

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere en yüksek performans sonucunu gösteren %70 eğitim seti, %30 veri seti kullanılarak oluşturulan Desicion Tree modelidir. Desicion Tree, K-N ve Bagging Tress algoritmaları performans sonuçları yüksek olup Naive Bayes ve Yapay Sinir Ağları'nın performans düzeyi düşüktür. Sebebinin veri setinin bu algoritmalar için uygun olmadığı düşünülmektedir.

Çalışmada kullanılan Makine Öğrenmesi modelleri performans yüzdesi açısından doğrulukları şu şekilde sıralanabilir:

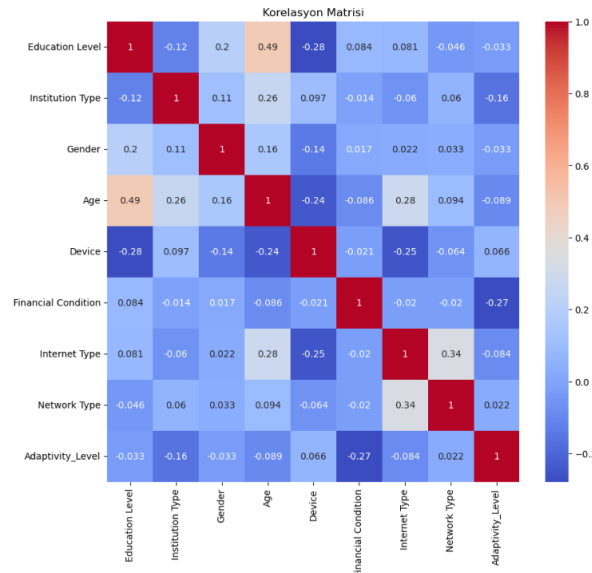
$$DT > BT > K-NN > YSA > NB$$

3.9. Korelasyon ile Öznitelik Seçimi

Veri seti 5 farklı algoritma ile modellenip her bir algoritmanın kendi içinde sonuçlarına ulaşılmıştır. Bu sonuçları optimize etmek adına veri setine korelasyon uygulanarak aradaki bağların 0.2-0.6 eşiği arasında olduğunu görüp öznitelikler belirlenmiştir.

Korelasyon sonrası seçilen öznitelikler: Eğitim Seviyesi, Okul Tipi, Yaş, İnternet Tipi, Ağ Tipi

Tablo 18. Korelasyon Tablosu



Kaynak: Yazar tarafından oluşturulmuştur.

3.10. Korelasyon Sonrası K-NN Algoritması Analizi

Veri setinde öznitelikler belirlendikten sonra tekrar K-NN algoritması uygulanmıştır.

Algoritmadaki komşu sayısı, “n_neighbors” parametresi kullanılarak 1 olarak belirlenmiştir. Bu değer “Cross-Validation Accuracy” parametresiyle belirlenmiştir. Modelde algoritma metriği olarak precision, recall, f1-score, model eğitim süresi ve tahmin süresi kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır. Bu modellerin hepsinde en iyi sonucu yukarıda belirtildiği üzere “Cross-Validation Accuracy” parametresiyle komşu sayısı 1 olarak belirlenmiştir.

Bu modellerin başarı performansı aşağıdaki tabloda belirtilmiştir.

Tablo 19. Korelasyon Sonrası Oluşturulan K-NN Algoritması Analizi Modelinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%76	%77	%77	%78	0.01	0.01
%70-%30	%79	%80	%80	%79	0.01	0.01
%80-%20	%75	%75	%74	%77	0.01	0.01
%90-%10	%71	%73	%75	%72	0.01	0.01

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere K-NN modellerinin başarı performansı sırasıyla şu şekildedir; %70-%30, %60-%40, %80-%20, %90-%10. Bu modellerden en iyi sonucu %60 eğitim seti, %40 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Öznitelik seçimi sonrası uygulanan K-NN'in performansı artmıştır.

En yüksek performansı veren %70 eğitim -%30 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %79, Presicion %80, Recall %80 ve f1-score %79 olarak belirlenmiştir.

3.11. Korelasyon Sonrası Naive Bayes Algoritması Analizi

Veri setinde öznitelikler belirlendikten sonra tekrar K-NN algoritması uygulanmıştır.

NB algoritmasında, veriler kategorik değere sahip olduğu için MultinomialNB kütüphanesinden yararlanılmıştır. Model “nb_model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak presicion, recall ve f1-score kullanılmıştır. Bu metriklerle, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

**Tablo 20. Korelasyon Sonrası Oluşturulan Naive Bayes Algoritması Analizi
Modelinin Karşılaştırılması**

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%67	%67	%65	%66	0.0028	0.0015
%70-%30	%67	%68	%66	%66	0.0025	0.0016
%80-%20	%66	%66	%61	%66	0.0032	0.0013
%90-%10	%62	%63	%63	%60	0.0033	0.0072

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere NB modellerinin başarı performansı sırasıyla şu şekildedir; %70-%30, %60-%40, %80-%20, %90-%10. Bu modellerden en iyi sonucu %70 eğitim seti, %30 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Öznitelik seçimi sonrası uygulanan Naive Bayes’in performansı artmıştır.

En yüksek performansı veren %70 eğitim -%30 test veri setinin oranları şu şekildedir.

Başarı Performans Yüzdesi %67, Presicion %68, Recall %66 ve f1-score %66 olarak belirlenmiştir.

3.12. Korelasyon Sonrası Decision Tree Algoritması Analizi

Veri setinde öznitelikler belirlendikten sonra tekrar Decision Tree algoritması uygulanmıştır.

Decision Tree algoritmasında, DecisionTreeClassifier kütüphanesinden yararlanılmıştır. Model “model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak precision, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 21. Korelasyon Sonrası Oluşturulan Decision Tree Algoritması Analizi
Modelinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%79	%80	%74	%76	0.0000	0.0009
%70-%30	%85	%85	%77	%80	0.0028	0.0010
%80-%20	%84	%85	%77	%80	0.0028	0.0009
%90-%10	%84	%84	%78	%81	0.0047	0.0009

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Decision Tree modellerinin başarı performansı sırasıyla şu şekildedir; %70-%30, %80-%20, %90-%10, %60-%40. Bu modellerden en iyi sonucu %70 eğitim seti, %30 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Öznitelik seçimi sonrası uygulanan Decision Tree’nin performansı artmıştır.

En yüksek performansı veren %70 eğitim -%30 test veri setinin oranları şu şekildedir.

Başarı Performans Yüzdesi %85, Presicion %85, Recall %77 ve f1-score %80 olarak belirlenmiştir.

3.13. Korelasyon Sonrası Yapay Sinir Ağı Algoritması Analizi

Veri setinde öznelilikler belirlendikten sonra tekrar YPA algoritması uygulanmıştır.

YPA algoritmasında, tensorflow kütüphanesinden yararlanılmıştır. Model “model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak presicion, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 22. Korelasyon Sonrası Oluşturulan YPA Algoritması Analizi Modelinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%66	%65	%64	%63	0.0062	0.0010
%70-%30	%64	%66	%65	%64	0.0067	0.0010
%80-%20	%65	%66	%65	%65	0.0065	0.0010
%90-%10	%67	%68	%66	%66	0.0062	0.0008

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Yapay Sinir Ağı modellerinin başarı performansı sırasıyla şu şekildedir; %90-%10, %80-%20, %70-%30, %60-%40. Bu modellerden en iyi sonucu %90 eğitim seti, %10 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Bu modelde kullanılan veri setlerinde başarı performansı artsa da öznitelik belirlemeden yapılan çalışmaya göre büyük bir fark bulunmamıştır.

En yüksek performansı veren %90 eğitim -%10 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %67, Presicion %68, Recall %66 ve f1-score %66 olarak belirlenmiştir.

3.14. Korelasyon Sonrası Bagging Tree Algoritması Analizi

Veri setinde öznitelikler belirlendikten sonra tekrar Bagging Tree algoritması uygulanmıştır.

Desicion Tree algoritmasında, DecisionTreeClassifier kütüphanesinden yararlanılmıştır. Model “model” ismi ile oluşturulmuştur.

Modelde algoritma metriği olarak presicion, recall ve f1-score kullanılmıştır. Bu metriklere, “sklearn.metrics” parametresiyle erişilmiştir.

Modeller oluşturulurken eğitim ve test setleri %60-%40, %70-%30, %80-%20 ve %90-%10 olmak üzere dört farklı şekilde ayrılmıştır.

Tablo 23. Korelasyon Sonrası Oluşturulan Bagging Tree Algoritması Analizi Modelinin Karşılaştırılması

Eğitim Seti – Test Seti	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
%60-%40	%80	%80	%81	%80	0.05	0.01
%70-%30	%79	%80	%79	%80	0.07	0.01
%80-%20	%78	%78	%78	%78	0.06	0.01
%90-%10	%79	%78	%79	%79	0.10	0.01

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere Bagging Tress modellerinin başarı performansı sırasıyla şu şekildedir; %90-%10, %60-%40, %70-%30, %80-%20. Bu modellerden en iyi sonucu %70 eğitim seti, %30 veri seti ve %60 eğitim seti, %40 veri seti kullanılarak oluşturulan model verse de diğer oranlar da çok farklılık yoktur. Diğer BG modelinde olduğu gibi bu modelde de kullanılan veri setlerinde başarı performansı çok yüksek olmasıyla birlikte dengeli bir dağılım bulunmaktadır. Sonuçlar tabloda ayrıntılı bir şekilde sunulmuştur. Bu algorithmada en iyi performansı gösteren model kullanılan diğer makine öğrenmesi modellerinden en iyi performansı gösteren modellerle de karşılaştırılacaktır.

En yüksek performansı veren %60 eğitim -%40 test veri setinin oranları şu şekildedir.;

Başarı Performans Yüzdesi %80, Presicion %80, Recall %81 ve f1-score %80 olarak belirlenmiştir.

3.15. Oluşturulan Hibrit Modellerinin Karşılaştırılması

Çalışmada kullanılan Makine Öğrenmesi modelleri kendi içlerinde karşılaştırıldıktan sonra en iyi sonucu veren modellemeler diğer algoritmalarla karşılaştırılmıştır. Sonuçlar aşağıdaki tabloda paylaşılmıştır.

Tablo 24. Modellerin Başarı Karşılaştırmaları

Algoritma Adı (Eğitim Seti – Test Seti)	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
K-NN (%70-%30)	%79	%80	%80	%79	0.01	0.01
Naive Bayes (%70-%30)	%67	%68	%66	%66	0.0025	0.0016
Decision Tree (%70-%30)	%85	%85	%77	%80	0.0028	0.0010
Bagging Trees (%60-%40)	%80	%80	%81	%80	0.05	0.01
Yapay Sinir Ağları (%90-%10)	%67	%68	%66	%66	0.0062	0.0008

Kaynak: Yazar tarafından oluşturulmuştur.

Tabloda görüldüğü üzere en yüksek performans sonucunu gösteren %70 eğitim seti, %30 veri seti kullanılarak oluşturulan Decision Tree modelidir. Modellerin hem yalnız kullanımda başarılı performansı olması hem de veri setine uygun olması Korelasyon sonrası performansını etkilemiştir. Decision Tree modeli tüm modeller arasında en yüksek sonucu veren algoritmadır.

3.16. En Başarılı Model ve En Başarılı Hibrit Model Karşılaştırması

Modelleri yalnız olarak kullandığımızda en başarılı modelin Desicion Tree modeli olduğu görülmüştür.

Korelasyon yaparak oluşturarak yapılan modellerde ise yine en yüksek modelin Desicion Tree olduğu görülmüştür.

Bu durum Decision Tree modelinin veri setine yatkın bir model olduğunu göstermekte olup öznelilik çıkarıldığında başarı düzeyinin arttığını göstermiştir. Aşağıda verilen tabloda iki modelin en yüksek oranlarının karşılaştırması gösterilmiştir.

Tablo 25. Desicion Tree ve Hibrit Modelin Başarı Karşılaştırması

Algoritma Adı (Eğitim Seti – Test Seti)	Başarı Performans Yüzdesi	Precision (Hassasiyet)	Recall (Duyarlılık)	F1-Score (Denge)	Model Eğitim Süresi (sn)	Tahmin Süresi (sn)
Desicion Tree (%70-%30)	%78	%79	%79	%79	0.0075	0.0009
Korelasyon sonrası Desicion Tree (%70-%30)	%85	%85	%77	%80	0.0028	0.0010

Kaynak: Yazar tarafından oluşturulmuştur.

SONUÇ

Bu bölümde Kaggle veri sitesinden elde edilen veriler kullanılarak farklı makine öğrenmesi algoritmalarının sonuçları değerlendirilmiştir. Bu sonuçlardan en başarılı performansı gösteren algoritma Decision Trees algoritmasıdır.

Modelin başarı performans yüzdesi %85 olarak accuracy_score parametresiyle değerlendirilmiştir.

Bu çalışmanın amacı var olan sorunu araştırarak etkin bir çevrim içi öğrenme yöntemi kullanılabilmesi için bir stratejik bir yöntem sunmak, teknoloji alanında devam eden çalışmalara katkı sağlamak amaçlanmaktadır. Ayrıca eğitim satan şirketler de elde edilecek olan veriler ile çevrim içi öğrenme uyum seviyesine uygun olarak eğitim tasarımı yapabilirler.

Tez çalışmasının literatüre katkısı yeni bir çalışma alanı olarak tanımlanabilir. Alanda bu çalışmaya benzer bir çalışma bulunmamaktadır. Şirketler bu çalışma da gösterilen algoritmalar ile hizmet verdiği kullanıcıların demografik ve ekonomik özelliklerini öğrenerek çevrim içi öğrenmeye olan uyumunu saptayabilir ve kurum içinde yaptığı eğitimlerini bu çerçevede yapabilir. Aynı zamanda çevrim içi uyum düzeyini artırmak için eksikler görülüp bu internet tipini güncelleme, cihazı değiştirme gibi aksiyonlar alınabilir.

Modellemeler sonucu çevrim içi uyum düzeyine sahip üç ana grup bulunmaktadır;

Bu veri setinde yüksek uyumlu olan grup 413 kişiyle erkek öğrencilerdir. Erkek öğrencilerin arasında ise 21-25 yaş arası özel okulda okuyan üniversite öğrencilerinin uyumluluk seviyesi yüksektir. Bu verileri daha da özelleştirdiğimizde ise ekonomik durumu düşük, derse giriş aracı olarak bilgisayar ve internet bağlantısı olarak wi-fi, network tipi olarak 4G kullanan grup en yüksek uyumluluğa sahiptir.

Düşük derecede uyumlu olan grup 345 kişiyle erkek öğrencilerdir. Erkek öğrencilerin arasında ise 21-25 yaş arası devlet bağlı okulda okuyan üniversite öğrencileri düşük derecede uyumludur. Bu verileri daha da özelleştirdiğimizde ise ekonomik durumu standart, derse giriş aracı olarak telefon ve internet bağlantısı olarak mobil veri, network tipi olarak 3G kullanan grup düşük derece uyumluluğa sahiptir.

Orta derecede uyumlu olan grup 447 kiřiyle erkek ğrencilerdir. Erkek ğrencilerin arasında ise 21-25 yař arası zel okulda okuyan niversite ğrencileri orta derecede uyumludur. Bu verileri daha da zelleřtirdiėimizde ise ekonomik durumu standart, derse giriř aracı olarak bilgisayar ve internet baėlantısı olarak wi-fi, network tipi olarak 4G kullanan grup orta derece uyumluluėa sahiptir.

Yapılan alıřmada tm zel sektr kurumlarının alıřanlarına verdiėi eėitim tasarımlarında strateji oluřturmak iin, zel ve devlet okullarında evrim ii ėrenmeye artırmak iin ve kiřisel olarak evrim ii eėitim sana firmaların eėitim tasarımı oluřturmalarında yardımcı olacaėı dřnlmektedir.



KAYNAKÇA

- Albayrak A. ve Yılmaz, K. (2009). Veri madenciliği: Karar ağacı algoritmaları ve imkb verileri üzerine bir uygulama. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 14(1), 31-52.
- Altunbaş, C. (2020, 13 Şubat). *Yapay sinir ağları*. 14 Nisan 2024 tarihinde <https://medium.com/türkiye/yapay-sinir-ağları-ysa-caec1cc0bf9f> adresinden edinilmiştir.
- Arslan, K. (2017). *Eğitimde yapay zeka ve uygulamaları*. *Batı Anadolu Eğitim Bilimleri Dergisi*, 11(1), 71-88.
- Aslanargun, E., Bozkurt, S. ve Sarioğlu, S. (2016). Sosyo ekonomik değişkenlerin öğrencilerin akademik başarıları üzerine etkileri. *Uşak Üniversitesi Sosyal Bilimler Dergisi*, 9(27/3), 201-234.
- Boden, M. A. (2018). Artificial intelligence: A very short introduction. *Oxford University Press*, <https://doi.org/10.1093/actrade/9780199602919.001.0001>.
- Çalış, A., Kayapınar, S., ve Çetinyokuş, T. (2014). Veri madenciliğinde karar ağacı algoritmaları ile bilgisayar ve internet güvenliği üzerine bir uygulama. *Endüstri Mühendisliği Dergisi*, 25(3), 2-19.
- Gök, M. (2017). Makine öğrenmesi yöntemleri ile akademik başarının tahmini. *Gazi University Journal of Science Part C: Design and Technology*, 5(3), 139-148.
- Gürsakaç, D. (2012). Pısa 2009 öğrenci başarı düzeylerini etkileyen faktörlerin değerlendirilmesi. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 17(1), 441-452.
- Karaboğa, U. (2020). *İşe alım süreçlerinde yapay zeka teknolojilerinin kullanımı* [Yüksek lisans tezi]. İstanbul Medipol Üniversitesi.
- Karataş, S. (2021). *Yapay zeka ve açık inovasyon etkileşiminin işletmeler üzerine etkileri* [Yüksek lisans tezi]. Aydın Adnan Menderes Üniversitesi.
- Kocamaz, A. (2012). *Makine öğrenmesi tabanlı bir uzman sistem tasarımı* [Doktora tezi]. Trakya Üniversitesi.

- Nizam, H. ve Akın, S. (2014, 27 Mayıs). *Sosyal medyada makine öğrenmesi ile duygu analizinde dengeli ve dengesiz veri setlerinin performanslarının karşılaştırılması* [Bildiri sunumu]. Türkiye'de İnternet Konferansları, İzmir.
- Onan, A. (2016). *Görüş sınıflandırma için makine öğrenmesi algoritmalarına dayalı bir yöntem tasarımı ve gerçekleştirimi* [Yüksek lisans tezi]. Ege Üniversitesi.
- Say, C. (2018). *50 soruda yapay zeka*. Storyside Yayınevi.
- Sevli, O. (2019). Göğüs kanseri teşhisinde farklı makine öğrenmesi tekniklerinin performans karşılaştırması. *Avrupa Bilim Ve Teknoloji Dergisi*, (16), 176-185. <https://doi.org/10.31590/ejosat.553549>.
- Süer, K. (2019). *Yapay zeka ile meme kanseri lenf nodu analizi* [Yüksek lisans tezi]. İstanbul Beykent Üniversitesi.
- Taşçı, G., ve Çelebi, M. (2020). Eğitimde yeni bir paradigma: “yükseköğretimde yapay zekâ”. *OPUS International Journal of Society Researches*, 16(29), 2346-2370. <https://doi.org/10.26466/opus.747634>.
- Tosunoğlu, E., Yılmaz, R., Özeren, E., ve Sağlam, Z. (2021). Eğitimde makine öğrenmesi: Araştırmalardaki güncel eğilimler üzerine inceleme. *Ahmet Keleşoğlu Eğitim Fakültesi Dergisi*, 3(2), 178-199.
- Türk, E. (2021). *Bilişim alanında öğrenim gören lise öğrencilerine yönelik üniversite tercihlerinin yapay zeka yöntemleri ile analizi* [Yüksek lisans tezi]. Tekirdağ Namık Kemal Üniversitesi.
- Ünsal, Ü. (2011). *Mesleki alan seçimlerinin makine öğrenmesi algoritması kullanılarak belirlenmesi* [Yüksek lisans tezi]. Gazi Üniversitesi.
- Yardımcı, T. (2011). *Makine öğrenmesi teknikleri ile rss besleme yönetimi*. [Yüksek lisans tezi]. Gazi Üniversitesi.
- Yılmaz, A. (2020). *Yapay zekâ*. Kodlab Yayınevi.
- Zilyas, D. (2023). *Makine öğrenmesi yöntemi ile eğitim başarısının tahmini modeli* [Yüksek lisans tezi]. İstanbul Beykent Üniversitesi.
- Zorluel, M. (2019). Yapay zeka ve telif hakkı. *Türkiye Barolar Birliği Dergisi*, 306-356.