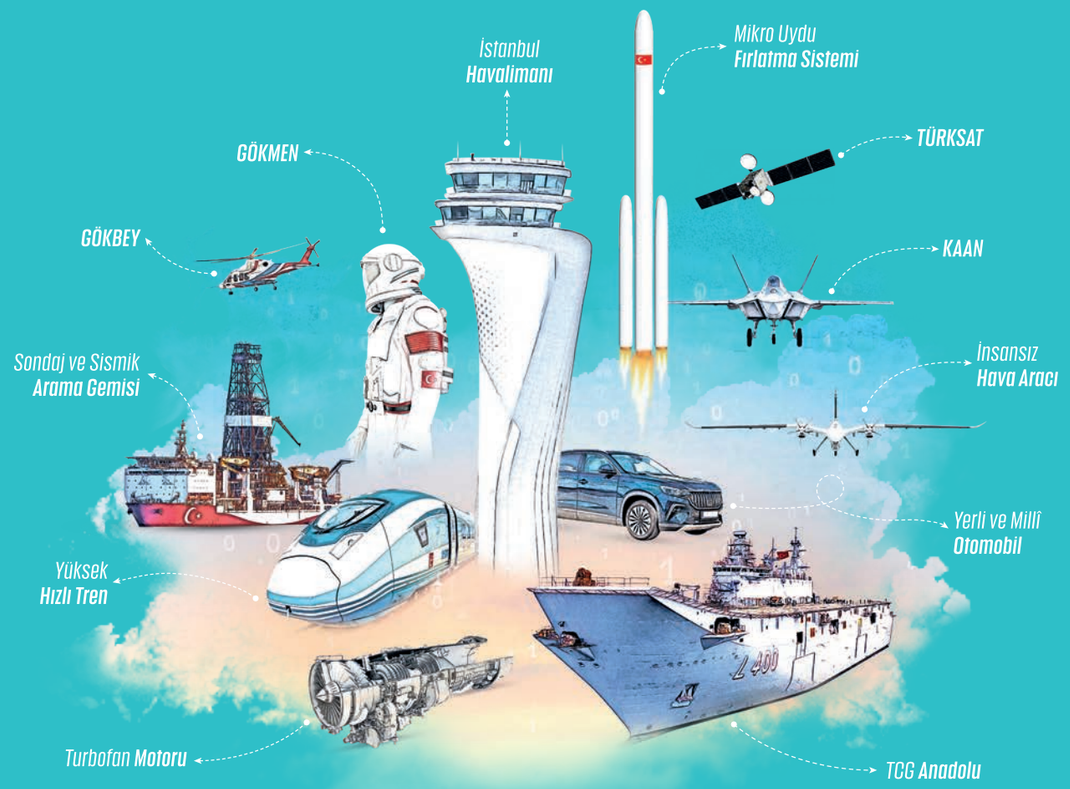
Bu kitaba sığmayan daha neler var!

eba
www.eba.gov.tr

Karekodu okut, bu kitapla ilgili EBA içeriklerine ulaş!

Teknoloji Yolculuğumuz



BU DERS KİTABI MİLLÎ EĞİTİM BAKANLIĞINCA ÜCRETSİZ OLARAK VERİLMİŞTİR.
PARA İLE SATILAMAZ.

ISBN: 978-625-5498-14-4

Bandrol Uygulamasına İlişkin Usul ve Esaslar Hakkında Yönetmeliğin Beşinci Maddesinin İkinci Fıkrası Çerçevesinde Bandrol Taşıması Zorunlu Değildir.

YAPAY ZEKÂ ALANI

YAPAY ZEKÂ TEMELLERİ

9 DERS MATERYALİ

MESLEKİ VE TEKNİK ANADOLU LİSESİ

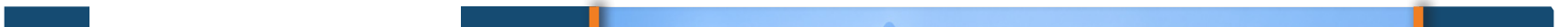
YAPAY ZEKÂ ALANI

YAPAY ZEKÂ TEMELLERİ



9

Ders Materyali



MESLEKİ VE TEKNİK ANADOLU LİSESİ

YAPAY ZEKÂ ALANI

YAPAY ZEKÂ TEMELLERİ

9

DERS MATERYALİ

YAZARLAR

Ayhan YAVUZ

Beyazıt Bestami YÜKSEL

Mehmet KAVAKLI

MİLLÎ EĞİTİM BAKANLIĞI YAYINLARI .....: 11011
YARDIMCI VE KAYNAK KİTAPLAR DİZİSİ .....: 3930

Her hakkı saklıdır ve Millî Eğitim Bakanlığına aittir.
Ders materyalinin metin, soru ve şekilleri kısmen de olsa hiçbir surette alınıp yayımlanamaz.

HAZIRLAYANLAR

Dil Uzmanı	Zülfü MUTLU
Program Geliştirme Uzmanı	Ali DOĞAN
Rehberlik Uzmanı	Başak KARAKÖSE
Ölçme ve Değerlendirme Uzmanı	Serpil DURMUŞ AKGEYİK
Görsel Tasarım Uzmanı	Ayça COŞKUN

ISBN: 978-625-5498-14-4

Millî Eğitim Bakanlığının 24.12.2020 gün ve 18433886 sayılı oluru ile
Meslekî ve Teknik Eğitim Genel Müdürlüğünce ders materyali olarak hazırlanmıştır.



İSTİKLÂL MARŞI

Korkma, sönmez bu şafaklarda yüzen al sancak;
Sönmeden yurdumun üstünde tüten en son ocak.
O benim milletimin yıldızıdır, parlayacak;
O benimdir, o benim milletimindir ancak.

Çatma, kurban olayım, çehreni ey nazlı hilâl!
Kahraman ırkıma bir gül! Ne bu şiddet, bu celâl?
Sana olmaz dökülen kanlarımız sonra helâl.
Hakkıdır Hakk'a tapan milletimin istiklâl.

Ben ezelden beridir hür yaşadım, hür yaşarım.
Hangi çılgın bana zincir vuracakmış? Şaşarım!
Kükremiş sel gibiyim, bendimi çiğner, aşarım.
Yırtarım dağları, enginlere sığmam, taşarım.

Garbın âfâkını sarmışsa çelik zırhlı duvar,
Benim iman dolu göğsüm gibi serhaddim var.
Ulusun, korkma! Nasıl böyle bir imanı boğar,
Medeniyyet dediğin tek dişi kalmış canavar?

Arkadaş, yurduma alçakları uğratma sakın;
Siper et gövdeni, dursun bu hayâsızca akın.
Doğacaktır sana va'dettiği günler Hakk'ın;
Kim bilir, belki yarın, belki yarından da yakın.

Bastığın yerleri toprak diyerek geçme, tanı:
Düşün altındaki binlerce kefensiz yatanı.
Sen şehit oğlusun, incitme, yazıktır, atanı:
Verme, dünyaları alsan da bu cennet vatanı.

Kim bu cennet vatanın uğruna olmaz ki feda?
Şüheda fışkıracak toprağı sıksan, şüheda!
Cânı, cânânı, bütün varımı alsın da Huda,
Etmesin tek vatanımdan beni dünyada cüda.

Ruhumun senden İlâhî, şudur ancak emeli:
Değmesin mabedimin göğsüne nâmahrem eli.
Bu ezanlar -ki şehadetleri dinin temeli-
Ebedî yurdumun üstünde benim inlemeli.

O zaman vecd ile bin secde eder -varsa- taşım,
Her cerîhamdan İlâhî, boşanıp kanlı yaşım,
Fışkırır ruh-ı mücerret gibi yerden na'şım;
O zaman yükselerek arşa değer belki başım.

Dalgalan sen de şafaklar gibi ey şanlı hilâl!
Olsun artık dökülen kanlarımın hepsi helâl.
Ebediyyen sana yok, ırkıma yok izmihlâl;
Hakkıdır hür yaşamış bayrağımın hürriyyet;
Hakkıdır Hakk'a tapan milletimin istiklâl!

Mehmet Âkif Ersoy

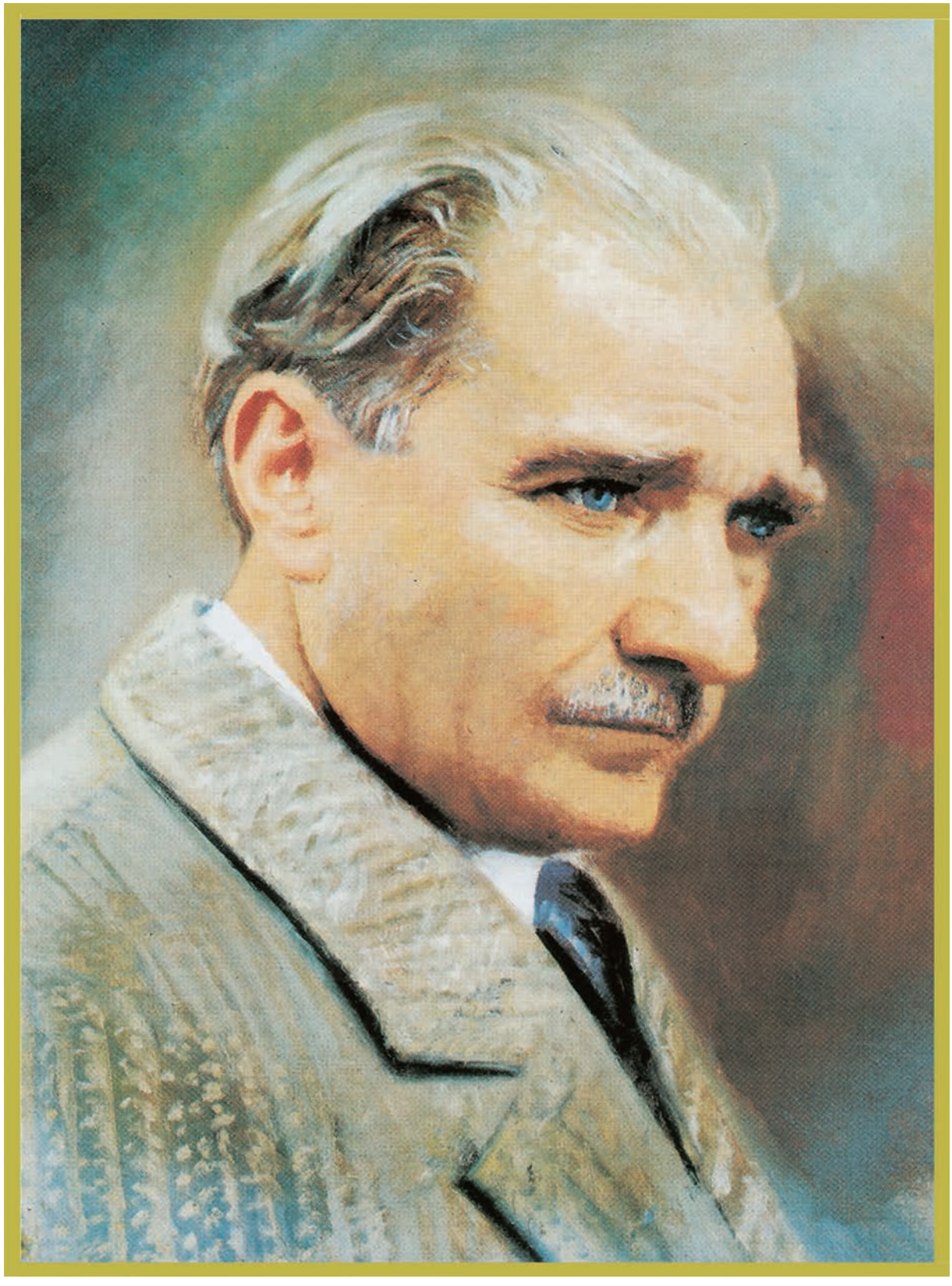
GENÇLİĞE HİTABE

Ey Türk gençliği! Birinci vazifen, Türk istiklâlini, Türk Cumhuriyetini, ilelebet muhafaza ve müdafaa etmektir.

Mevcudiyetinin ve istikbalinin yegâne temeli budur. Bu temel, senin en kıymetli hazinendir. İstikbalde dahi, seni bu hazineden mahrum etmek isteyecek dâhilî ve hâricî bedhahların olacaktır. Bir gün, istiklâl ve cumhuriyeti müdafaa mecburiyetine düşersen, vazifeye atılmak için, içinde bulunacağın vaziyetin imkân ve şeraitini düşünmeyeceksin! Bu imkân ve şerait, çok namüsait bir mahiyette tezahür edebilir. İstiklâl ve cumhuriyetine kastedecek düşmanlar, bütün dünyada emsali görülmemiş bir galibiyetin mümessili olabilirler. Cebren ve hile ile aziz vatanın bütün kaleleri zapt edilmiş, bütün tersanelerine girilmiş, bütün orduları dağıtılmış ve memleketin her köşesi bilfiil işgal edilmiş olabilir. Bütün bu şeraitten daha elîm ve daha vahim olmak üzere, memleketin dâhilinde iktidara sahip olanlar gaflet ve dalâlet ve hattâ hıyanet içinde bulunabilirler. Hattâ bu iktidar sahipleri şahsî menfaatlerini, müstevlîlerin siyasî emelleriyle tevhit edebilirler. Millet, fakr u zaruret içinde harap ve bîtap düşmüş olabilir.

Ey Türk istikbalinin evlâdı! İşte, bu ahval ve şerait içinde dahi vazifen, Türk istiklâl ve cumhuriyetini kurtarmaktır. Muhtaç olduğun kudret, damarlarındaki asil kanda mevcuttur.

Mustafa Kemal Atatürk



MUSTAFA KEMAL ATATÜRK

Ders materyallerinin dijital kopyalarında "İçindekiler" sayfasındaki başlıkların üzerine tıklayarak ilgili sayfalara, iç sayfalarda ise Öğrenme Birimi isimleri veya numaralarının üzerine tıklayarak "İçindekiler" sayfasına ulaşabilirsiniz.

İÇİNDEKİLER

DERS MATERYALİNİN TANITIMI 9

1. ÖĞRENME BİRİMİ: YAPAY ZEKÂYA GİRİŞ 12

1.1. YAPAY ZEKÂNIN TEMEL ÖZELLİKLERİ 14
1.1.1. Yapay Zekânın Özellikleri 15
1.1.2. Yapay Zekânın Tarihsel Gelişimi 16
1.1.3. Yapay Zekânın Alt Dalları 22
1.1.4. Yapay Zekânın Kullanım Alanları 24
1.1.5. Güncel Yapay Zekâ Uygulamaları ve Faydaları 25
1.1.6. Yapay Zekâ Geliştirme Ortamlarının Kurulumu 28
1.2. YAPAY ZEKÂDA VERİ İŞLEME KÜTÜPHANELERİ 39
1.2.1. NumPy Kullanarak Dizi Oluşturma 40
1.2.2. NumPy Kullanılarak Temel Matematiksel İşlemler 43
1.2.3. NumPy Dizilerinde İndeksleme, Dilimleme ve Temel Özellikler 46
1.2.4. Pandas Kütüphanesi 52
1.2.5. Veri Temizleme ve Düzenleme 58
1.3. MAKİNE ÖĞRENMESİ KÜTÜPHANELERİ 65
1.3.1. Scikit-learn Kütüphanesi 67
1.3.2. TensorFlow ve PyTorch Kütüphaneleri 69
1.3.3. Scikit-learn ile Sınıflandırma Uygulaması 71
1.3.4. Model Performansının Değerlendirilmesi 74
1.3.5. Gerçek Yaşamda Makine Öğrenmesi Kütüphaneleri 77
ÖLÇME VE DEĞERLENDİRME 82

2. ÖĞRENME BİRİMİ: ETİK VE SOSYAL ETKİLER 88

2.1. YAPAY ZEKÂNIN ETİK BOYUTLARI 90
2.1.1. Büyük Veri ve Veri Gizliliği 93
2.1.2. Yapay Zekâ ve İş Gücü Piyasası 94
2.1.3. Yapay Zekâ Entegrasyonu ve Verimlilik 96
2.2. YAPAY ZEKÂ YASAL BOYUTLARI 98
2.2.1. Teknolojik Gelişmelerin Doğurduğu Yasal Boşluklar 99
2.2.2. Veri Güvenliği ve Dijital Teknolojilere İlişkin Yasal Düzenlemeler 100
2.2.3. Yapay Zekâ Alanında Yapılan Yasal Düzenlemeler 101
ÖLÇME VE DEĞERLENDİRME 103

3. ÖĞRENME BİRİMİ: YAPAY ZEKÂDA ÖĞRENME TÜRLERİ 106

3.1. YAPAY ZEKÂ ÖĞRENME TÜRLERİNİ KARŞILAŞTIRMA 109
3.1.1. Denetimli Öğrenme 109
3.1.2. Denetimsiz Öğrenme 114

3.1.3. Yarı Denetimli Öğrenme 117
3.1.4. Pekiştirmeli Öğrenme 118
3.2. YAPAY ZEKÂ ÖĞRENME TÜRLERİNİ UYGULAMA 122
3.2.1. Denetimli Öğrenme Uygulamaları 122
3.2.2. Denetimsiz Öğrenme Uygulamaları 135
3.2.3. Yarı Denetimli Öğrenme Uygulamaları 141
3.2.4. Pekiştirmeli Öğrenme Uygulamaları 149
3.2.5. Probleme Göre Uygun Öğrenme Türünü Seçme Uygulamaları 156
ÖLÇME VE DEĞERLENDİRME 162

4. ÖĞRENME BİRİMİ: YAPAY ZEKÂ YÖNTEMLERİ 168

4.1. REGRESYON YÖNTEMLERİ 171
4.1.1. Sayısal Tahmin İçin Regresyon 171
4.1.2. Tahmin Modelinde Giriş ve Sonuç Değişkenleri 171
4.1.3. Veriler Arasındaki İlişkiyi Grafikle Gösterme 172
4.1.4. Lineer ve Polinom Regresyonu Ayırt Etme 172
4.1.5. Doğrusal Regresyon Modeli Oluşturma 173
4.1.6. Model Eğitimi, Testi ve Tahmin Hatası 174
4.2. SINIFLANDIRMA 180
4.2.1. Verileri Sınıflara Ayırma 180
4.2.2. İki Sınıflı ve Çok Sınıflı Problemler 181
4.2.3. Karar Ağacıyla Sınıf Tahmini 181
4.2.4. Doğruluk Oranı ve Karmaşıklık Matrisi 183
4.3. KÜMELEME 195
4.3.1. Benzer Verileri Gruplandırma 195
4.3.2. Veriler Arasındaki Yakınlığı Belirleme 195
4.3.3. K-Means Kullanarak Küme Oluşturma 196
4.3.4. K Değeri ve Elbow Yöntemi 198
4.3.5. Kümeleri Grafik Üzerinde Yorumlama 199
4.4. BİRLİKTELİK KURALLARI 205
4.4.1. Veriler Arasındaki Birlikte Görülme İlişkileri 205
4.4.2. Birliktelik Analizi İçin Veriyi Düzenleme 206
4.4.3. Birliktelik Kurallarını Ölçme 207
4.4.4. Sık Görülen Öğe Kümelerini Bulma 207
4.4.5. Birliktelik Kurallarından Karar Üretme 208
4.5. VAKA ÇALIŞMALARI 217
4.5.1. Başarı Hikâyeleri ve Kullanılan Yöntemler 222
4.5.2. Örnek Vakalar Üzerinde Yapay Zekâ Uygulamaları 225
ÖLÇME VE DEĞERLENDİRME 231
KAYNAKÇA 234
CEVAP ANAHTARLARI KAREKODU 236
GENEL AĞ, GÖRSEL, E-İÇERİK KAYNAKÇASI KAREKODU 236

Ders Materyalinin Tanıtımı



Öğrenme biriminin adını gösterir.

Öğrenme biriminin numarasını gösterir.

Öğrenme birimine uygun görseli içerir.

Öğrenme biriminin tanıtım karekodunu gösterir.

Öğrenme biriminde konu başlıklarını gösterir.

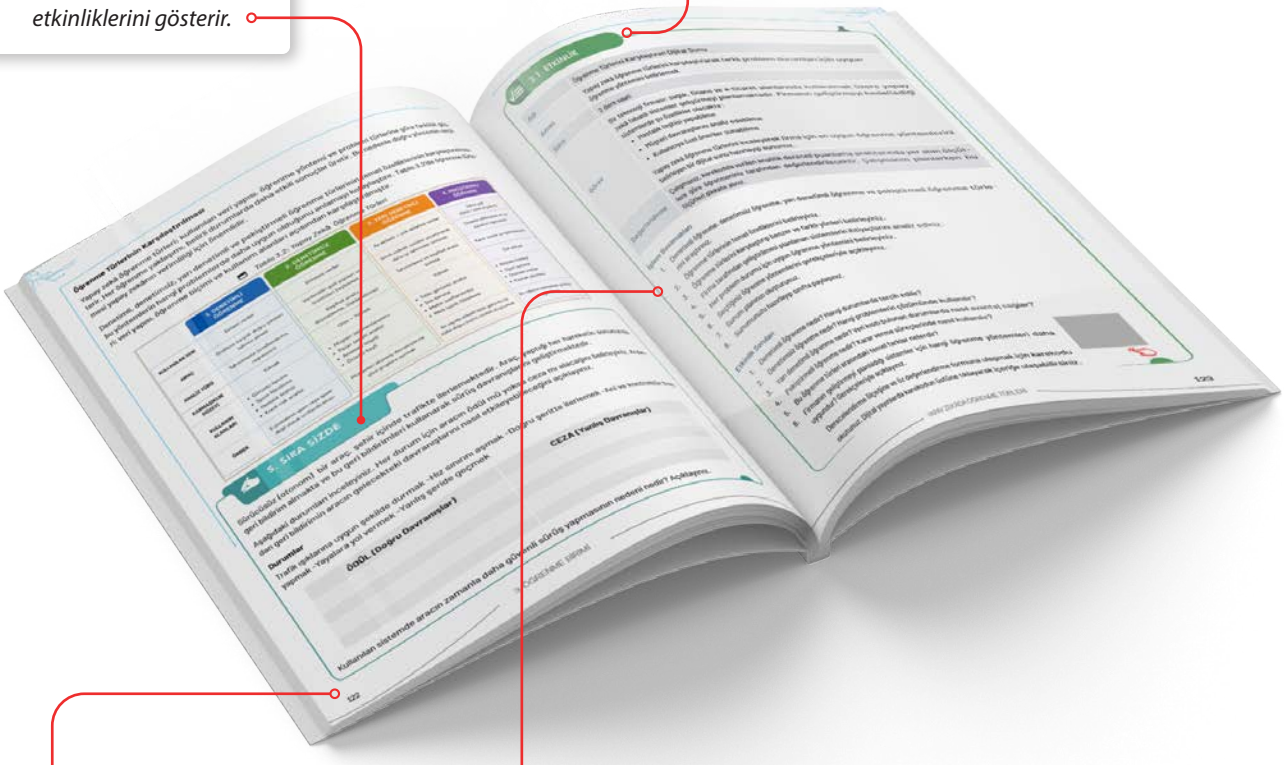
Öğrenme biriminde yer alan anahtar kavramları gösterir.

MATERYAL GENELİNDE KULLANILAN İKONLAR

Konular	Etkinlik	Uygulama
Anahtar Kavramlar	Sıra Sizde	Örnek Uygulama
Beklentiler	Bilgi Kutusu	Düşün, Tartış, Paylaş

Etkinlik numarasını ve etkinlik adını gösterir.

Bireysel çalışmalarda tamamlanacak "sıra sizde" etkinliklerini gösterir.

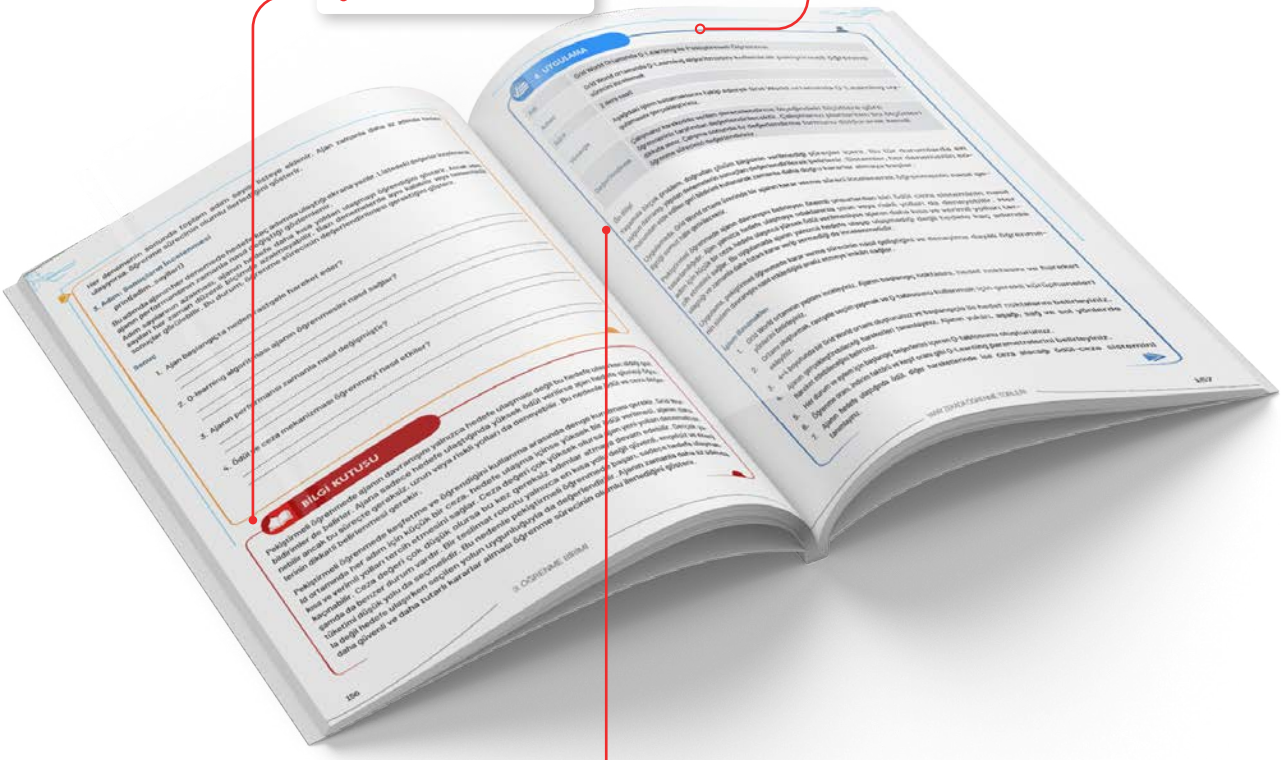


Sayfa numarasını gösterir.

Konuya ilişkin etkinlik adımlarını gösterir.

Uygulama numarasını ve adını gösterir.

Bilgi kutusunu gösterir.



Uygulamada yer alan ön bilgileri gösterir.



1. ÖĞRENME BİRİMİ

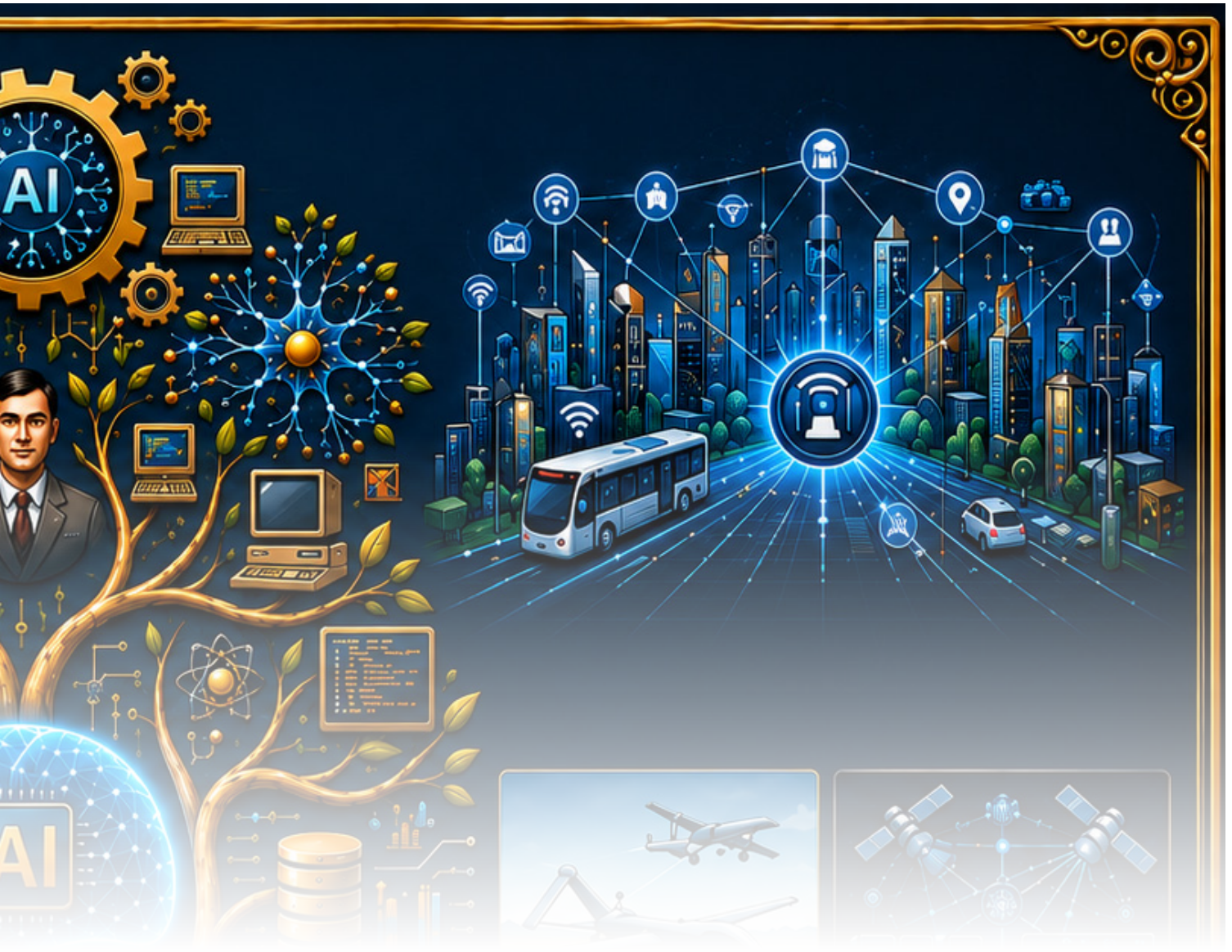
YAPAY ZEKÂYA GİRİŞ

KONULAR

1.1. YAPAY ZEKÂNIN TEMEL ÖZELLİKLERİ
1.2. YAPAY ZEKÂDA VERİ İŞLEME KÜTÜPHANELERİ
1.3. MAKİNE ÖĞRENMESİ KÜTÜPHANELERİ

ANAHTAR KAVRAMLAR

- NumPy
- Pandas
- Python
- PyTorch
- Scikit-learn
- TensorFlow
- veri ön işleme
- yapay zekâ



BEKLENTİLER

Bu öğrenme biriminde sizden

- yapay zekânın tarihsel gelişimi, temel özellikleri ve alt dallarını açıklamanız
- makine öğrenimi, derin öğrenme, sinir ağları, büyük veri, doğal dil işleme gibi yapay zekâ alt dallarını açıklamanız
- yapay zekânın sağlık, finans, ulaşım, eğitim gibi farklı alanlardaki kullanım örneklerini vermeniz
- NumPy (nampay), Pandas ve temel makine öğrenmesi kütüphanelerini kullanarak veri işleme ve yapay zekâ uygulamaları geliştirmeniz

beklenmektedir.

Öğrenme Birimi Karekodu

BAŞLARKEN

Günümüzde yapay zekâ uygulamalarının büyük bölümü, büyük verilerin doğru şekilde işlenip analiz edilmesine dayanır. Yapay zekâ sisteminin devasa veri kümelerini nasıl anlamlandırdığını ve bu veriler üzerinde hangi işlemleri gerçekleştirdiğini anlamak gerekir. Büyük veri kümeleri işlenirken ihtiyaç duyulan gelişmiş yazılımlar, günlük yaşamda kullanılan navigasyon sistemlerinden sosyal medyadaki kişiselleştirilmiş önerilere kadar pek çok sürecin arka planında yer alır. Bu sistemlerde kullanılan yazılım ve fonksiyonel kütüphaneler; verilerin düzenlenmesi, temizlenmesi ve analiz edilmesi süreçlerini daha etkin ve yönetilebilir hâle getirir. Bu öğrenme biriminde yapay zekânın tanımı, ortaya çıkışı ve büyük verinin anlamlandırılması yolculuğunda uygulamaların temelini oluşturan veri işleme süreçleri tanıtılacaktır. Python programlama diliyle NumPy, Pandas gibi kütüphanelerin sunduğu imkânlara ve bu alandaki temel becerilere uygulamalı olarak değinilecektir.

ÖNCE DÜŞÜN

Günlük yaşamda kullanılan sesli asistanlar, sosyal medya platformları ve otonom araçlar; çeşitli öneriler sunarak kullanıcıların kararlarını yönlendirmektedir. Bu teknolojiler, kullanıcı verilerini işleyerek belirli sonuçlar üretip bu sonuçlara göre hareket eder. Ancak bu sistemlerin verileri nasıl işlediği ve bu sonuçları nasıl ürettiği çoğu zaman fark edilmez. Bu süreçte kullanılan verilerin eksik, hatalı veya düzensiz olması, elde edilen sonuçların doğruluğunu etkiler. Bu durum, yapay zekâ sistemlerinin arka planında yürütülen veri işleme sürecinin önemini gösterir.

Yapay zekâ sistemlerinde veri işleme süreciyle ilgili genel bilgilerden yola çıkarak aşağıdaki tabloyu doldurunuz. Tabloda yer alan teknoloji türleri için kullanılabilecek veri türlerini ve bu verilerdeki hataların doğuracağı olası sonuçları tahmin ederek ilgili kutucuğa yazınız.

Teknoloji Örneği	Bu sistem hangi verileri kullanıyor olabilir?	Veri hatalı gelirse sistem nasıl bir karar verebilir?
Sürücüsüz Araç		
Sesli Asistan		
Sosyal Medya		
Sizin Örneğiniz		

1.1. YAPAY ZEKÂNIN TEMEL ÖZELLİKLERİ

Düşün, Tartış, Paylaş

Günlük yaşamda kullanılan birçok teknoloji, kullanıcıların ihtiyaçlarını anlamaya ve onlara uygun öneriler sunmaya çalışır. Akıllı telefonlardaki sesli asistanlar, video izleme platformlarının öneri sistemleri ve navigasyon uygulamaları bu teknolojilere örnek olarak verilebilir. Bu sistemlerin bazıları, kullanıcı davranışlarını analiz ederek zamanla daha doğru sonuçlar üretebilmektedir.

Verilen metne göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Günlük yaşamınızda kullandığınız hangi teknolojilerin akıllı davrandığını düşünüyorsunuz?**
2. **Bir bilgisayar veya makinenin öğrenmesi sizce ne anlama gelmektedir?**
3. **İnsanların yürüttüğü hangi görevler yapay zekâ sistemleri tarafından gerçekleştirilebilir**?

Yukarıdaki sorular değerlendirildiğinde, bazı teknolojik sistemlerin insan benzeri özellikler göstererek öğrenme, karar verme, problem çözme gibi görevleri yerine getirebildiği görülür. Bu sistemlerin nasıl çalıştığının anlaşılması için yapay zekânın temel özelliklerinin incelenmesi gerekir.

Yapay zekâ, insan zekâsına özgü öğrenme, akıl yürütme, karar verme, problem çözme gibi bilişsel süreçlerin bilgisayar ve makineler tarafından taklit edilmesini sağlayan mühendislik ve bilim alanıdır. Geleneksel bilgisayar programları; önceden tanımlanmış, sınırları belirli görevleri ve komutları yerine getiren statik yapıya sahiptir. Yapay zekâ sistemleriyse veri örüntüleri içerisinden kurallar çıkaran dinamik bir yapı sunar. Bu sistemler, uygun veri öğrenme süreçleriyle performansını iyileştirebilir ve yeni durumlara uyum sağlayabilir. Görsel 1.1' de günlük yaşamda yapay zekânın kullanıldığı alanlara ilişkin örnekler verilmiştir.



Görsel 1.1: Yapay zekânın kullanım örnekleri

1.1.1. Yapay Zekânın Özellikleri

Yapay zekâ sistemleri, geleneksel yazılımlardan farklı olarak yalnızca verilen komutları yerine getirmekle kalmaz; öğrenme, akıl yürütme, problem çözme, algılama ve doğal dili anlama gibi insan zekâsına özgü bilişsel yeteneklere de sahiptir. Bu yetenekler sayesinde çevrelerinden bilgi edinebilir, elde ettikleri bilgileri yorumlayabilir ve uygun kararlar verebilir. Yapay zekânın temel işlevlerinin daha iyi anlaşılabilmesi için Şema 1.1' de gösterilen bu beş temel yeteneğin incelenmesi gerekir.



Şema 1.1: Yapay zekânın temel yetenekleri

Öğrenme

Yapay zekâ işlevlerinden biri olan öğrenme, sistemin veri kümelerini analiz ederek belirli görevleri yerine getirmesini sağlayan eğitim sürecini ifade eder. Bu süreçte verinin yapısına ve sistemin çalışma biçimine bağlı olarak denetimli, denetimsiz ve pekiştirmeli öğrenme yaklaşımları yer alır.

Akıl Yürütme

Sistemin eldeki verileri kullanarak çıkarımlar yapmasını ve bu çıkarımlara dayalı sonuçlar üretmesini ifade eder. Modern yapay zekâ sistemleri yalnızca kesin doğrular üzerinden değil olasılıklar üzerinden de çıkarımlarda bulunur. Örneğin tıbbi teşhis yapan yapay zekâ sistemi, hastanın belirtilerini analiz ederek en yüksek olasılıklı hastalığı akıl yürütme süreciyle belirleyebilir.

Problem Çözme

Yapay zekâ sistemleri, mevcut seçenekleri değerlendirerek karmaşık problemlere çözüm önerileri üretebilir. Bu süreçte sistemler, çok farklı senaryoyu kısa sürede değerlendirip hata payını en aza indirecek şekilde uygun çözümü belirleyebilir.

Algılama

Yapay zekâ sistemleri, dış dünyadan gelen ham verileri (görüntü, sensör verisi, ses vb.) çözümleyerek işlenebilir bilgiye dönüştürebilir. Bilgisayar ve makineler, sinyal işleme ve bilgisayarlı görme teknolojileri sayesinde bu verileri işleyerek fiziksel dünyayla etkileşime geçebilir.

Doğal Dil İşleme ve Anlama

Yapay zekâ sistemleri insan diline ait kelime dizimi, bağlam, vurgu, tonlama gibi özellikleri analiz edebilir. Bu sayede yapay zekâ tabanlı sohbet robotları, sorulara anlamlı ve akıcı yanıtlar verebilir. Sesli asistanlar, kullanıcılardan gelen karmaşık talimatları yerine getirebilir ve farklı diller arasında anlık çeviri yapabilir.

BİLGİ KUTUSU

Yapay zekâ sistemleri çoğu zaman bu beş özelliğin hepsini veya bir kısmını bir arada kullanır. Örneğin otonom bir araç sistemi; dış çevreyi algılayarak yolu tanır, trafik kurallarına uygun kararlar verir, yolda engelle karşılaştığında en uygun hareketi planlar, kullanıcıdan gelen sesli komutları yorumlayıp istenen görevleri yerine getirir. Tüm bu deneyimlerinden yararlanarak zamanla performansını da geliştirir.

1.1.2. Yapay Zekânın Tarihsel Gelişimi

Yapay zekâ düşüncesi, modern bilgisayarların icadından çok önce felsefi ve teorik tartışmalarla şekillenmeye başlamıştır. Antik dönemden itibaren insan gibi düşünebilen makineler fikri ele alınmıştır. Yirminci yüzyılın başlarında Alan Turing, hesaplama kuramı gibi çalışmalarla makinelerin bilgi işleme kapasitesine dair bilimsel bir temel ortaya koymuştur. Ancak yapay zekânın modern anlamda bir bilim dalı olarak ortaya çıkışı 1940'lı yıllardan itibaren gerçekleşmiştir. Yapay zekânın günümüzde ulaştığı gelişmiş düzeye bir anda erişmediği, uzun yıllara yayılan bilimsel ve teknolojik çalışmaların bir sonucu olarak şekillendiği görülmektedir. Bundan dolayı yapay zekâ alanının gelişim sürecinin anlaşılabilmesi için tarihsel süreçte yaşanan önemli olay ve kavramların incelenmesiyle alana katkı sağlayan bilim insanlarının tanınması önem taşımaktadır. Yapay zekânın ortaya çıkışından günümüze kadar yaşadığı gelişim; önemli kişiler, kavramlar ve olaylar doğrultusunda kronolojik olarak dokuz ana evrede ele alınabilir. Bu evreler şunlardır:

Kuramsal Temeller ve Sibernetik Dönemi (1940-1956)

Yapay zekânın bilimsel temelleri 1940'lı yıllarda atılmıştır. 1943 yılında Warren McCulloch (Vorın Makkalok) ve Walter Pitts (Voltır Pits), yapay sinir ağlarının matematiksel modelini geliştirerek bu alanın ilk adımını atmışlardır.

1950 yılında Alan Turing, "Makineler düşünebilir mi?" sorusunu ortaya atmıştır (Görsel 1.2). Turing Testi adı verilen bu sistem yapay zekâ tarihinde önemli bir dönüm noktası olarak kabul edilir. Aynı dönemde Norbert Wiener (Norbırt Viner), **Sibernetik** kavramını geliştirerek makinelerde kontrol ve iletişim süreçlerinin fikrini ortaya atmıştır. Bu dönemde yapılan çalışmalar, yapay zekânın bilimsel ve kavramsal temelini oluşturmuş, sonraki çalışmalar için zemin hazırlamıştır.



Görsel 1.2: Alan Turing, "Makineler düşünebilir mi?"

Yapay Zekânın Doğuşu ve Sembolik Yaklaşım (1956-1970)

Yapay zekâ terimi ilk kez 1956 yılında John McCarthy (Con Makkarti) tarafından düzenlenen Dartmouth (Dartmut) konferansında kullanılmıştır. Bu tarih, yapay zekânın bağımsız bilim dalı olarak ele alınması ve bilimsel disiplin olarak kabul edilmesi açısından dönüm noktası olmuştur. Bu dönemde Marvin Minsky (Marvin Minski), Herbert Simon (Hörbırt Saymın) gibi bilim insanları, insan zekâsını mantıksal kurallarla modellemeyi amaçlayan **Sembolik Yapay Zekâ** yaklaşımını geliştirmişlerdir. Bu yaklaşım temelde kural tabanlı sistemler ve mantıksal problem çözme yöntemlerini ön plana çıkarmıştır. Dönemin en önemli özelliği yapay zekânın bağımsız araştırma alanı olarak kabul edilmiş olmasıdır.

Birinci Yapay Zekâ Kışı (1970-1980)

Yapay zekâ alanında yapılan çalışmaların beklentileri karşılayamaması nedeniyle 1970'li yılların başından itibaren duraklama dönemi yaşanmıştır. Marvin Minsky ve Seymour Papert tarafından yayımlanan **Perceptrons** (Persep-trons) adlı çalışma, tek katmanlı sinir ağlarının matematiksel olarak sınırlı olduğunu ortaya koymuş, bu durum yapay zekâ alanındaki çalışmaların yavaşlamasına ve özellikle 1973 yılında yayımlanan Lighthill raporuyla birlikte araştırma fonlarının azaltılmasına neden olmuştur. Bu süreç yapay zekâ araştırmalarında beklentilerin daha gerçekçi biçimde değerlendirilmesinin önünü açmıştır.

BİLGİ KUTUSU

Yapay zekâ tarihi, doğrusal bir gelişim süreci izlememiştir. Bu alanda dönem dönem büyük ilerlemeler ve yüksek beklentiler ortaya çıkarken bazı zamanlardaysa bu beklentilerin karşılanamaması sonucunda ciddi duraklamalar yaşanmıştır. **Yapay zekâ kışları** olarak adlandırılan bu süreçler; araştırma faaliyetlerinin yavaşladığı, yatırımların azaldığı ve alana olan ilginin geçici olarak düştüğü dönemleri ifade eder.

Uzman Sistemler Dönemi (1980-1987)

Edward Feigenbaum öncülüğünde 1980'li yıllarda geliştirilen **uzman sistemlerle** yapay zekâ yeniden yükselişe geçmiştir. Uzman sistemlerin temel amacı, belirli alandaki insan bilgisinin bilgisayar sistemlerine aktarılması ve bu sistemlerin uzmanlaşmasının sağlanmasıdır. Sistemlerin tıp, mühendislik, hukuk gibi alanlarda kullanımına başlansa da sistem bakımının zorluğu ve esnekliğinin yetersizliği nedeniyle aşılması güç sorunlar ortaya çıkmıştır. Dönemin en önemli özelliği, yapay zekânın ilk defa ticari ve pratik alanlarda yaygın biçimde kullanımın başlamasıdır.

İkinci Yapay Zekâ Kışı (1987-1993)

Uzman yapay zekâ sistemlerinin yüksek maliyetli donanımlara ihtiyaç duyması ve karmaşık problemleri çözmede yetersiz kalması nedeniyle 1987-1993 yılları arasında ikinci yapay zekâ kışı yaşanmıştır. Alana yapılan yatırımların azalması ve özellikle Lisp makine pazarının çökmesi, yapay zekâ alanındaki gelişmelerin yavaşlamasına neden olmuştur. İkinci yapay zekâ kışı daha verimli ve esnek yöntem arayışını hızlandırmıştır.

Makine Öğrenmesi ve İstatistiksel Yaklaşımlar Dönemi (1990-2010)

Kurallara dayalı sistemlerden veriye dayalı sistemlere geçişin yaşandığı bu dönemde, yapay zekâ alanında önemli dönüşüm gerçekleşmiştir. 1986 yılında Geoffrey Hinton ve arkadaşları tarafından yeniden popüler hâle getirilen backpropagation (geri yayılım) algoritması, yapay sinir ağlarının çok katmanlı yapılarla eğitilmesini mümkün kılarak önceki sınırların aşılmasını sağlamıştır. 1997 yılında IBM'in Deep Blue sisteminin dünya satranç şampiyonu Garry Kasparov'u yenmesi, yapay zekâ alanında önemli bir dönüm noktası olmuş ve bu sistemlerin hesaplama gücünü ortaya koymuştur (Görsel 1.3). Yapay zekâ alanındaki bu geçiş, kural tabanlı modellerden veriye dayalı modellere yönelimi güçlendirmiştir.



Görsel 1.3: Kasparov ve Deep Blue (1997)

Derin Öğrenme Devrimi (2010-2016)

Büyük veri (Big Data) ve grafik işlemci (GPU) teknolojilerinin gelişmesiyle derin öğrenme alanında hızlı bir yükseliş yaşanmıştır. 2012 yılında Geoffrey Hinton ve arkadaşları tarafından geliştirilen AlexNet modeli, Ima-

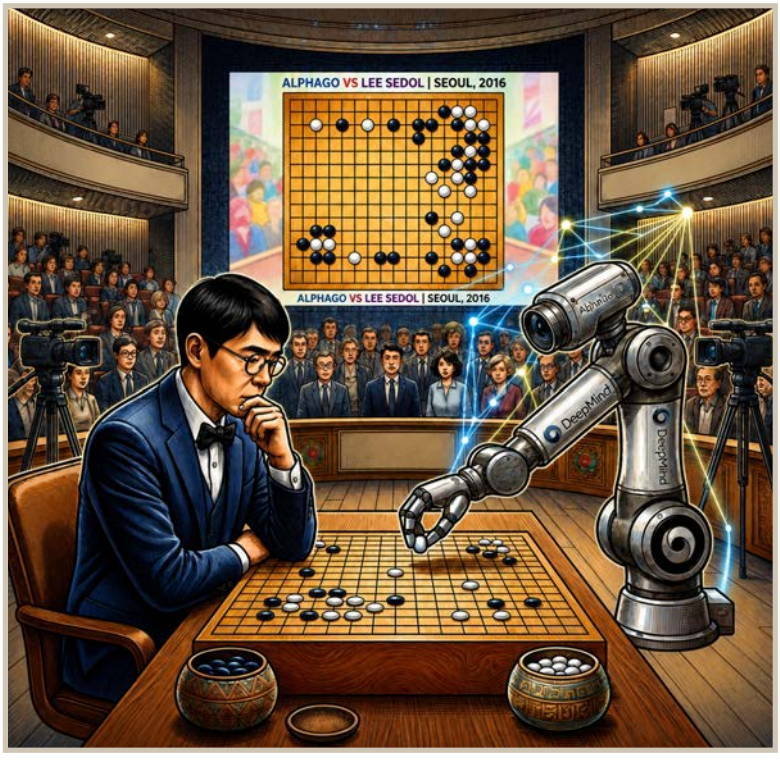
geNet yarışmasını kazanarak görüntü tanıma alanında önemli bir başarıya imza atmıştır (Görsel 1.4). Elde edilen bu başarı, yapay zekâ alanındaki önemli dönüm noktalarından biri olarak kabul edilmiş ve modern yapay zekânın bu alandaki temel yaklaşım hâline gelmesini sağlamıştır. Bu dönemde özellikle görüntü işleme, ses tanıma ve doğal dil işleme alanlarında ilerlemeler gerçekleşmiştir.



Görsel 1.4: AlexNet ve derin öğrenme devrimi (2012)

Transformer Mimarisi ve Stratejik Yapay Zekâ (2016-2020)

Google DeepMind tarafından 2016 yılında geliştirilen AlphaGo, dünya Go şampiyonu Lee Sedol'u yenerek yapay zekânın strateji ve sezgi gerektiren alanlarda insan performansına ulaşabileceğini göstermiştir (Görsel 1.5). Bu olay, yapay zekâ alanında önemli bir dönüm noktası olarak kabul edilir. Bununla birlikte 2017 yılında yayımlanan Attention Is All You Need makalesiyle tanıtılan Transformer mimarisi, doğal dil işleme alanında devrim niteliğinde bir gelişme sağlayarak günümüzdeki büyük dil modellerinin temelini oluşturmuştur.



Görsel 1.5: AlphaGo ve Lee Sedol (2016)

Transformer Mimarisi ve Stratejik Yapay Zekâ (2016-2020)

Google DeepMind tarafından 2016 yılında geliştirilen AlphaGo, dünya Go şampiyonu Lee Sedol'u yenerek

yapay zekânın strateji ve sezgi gerektiren alanlarda insan performansına ulaşabileceğini göstermiştir. Bu olay, yapay zekâ alanında önemli bir dönüm noktası olarak kabul edilir. Bununla birlikte 2017 yılında yayımlanan Attention Is All You Need makalesiyle tanıtılan Transformer mimarisi, doğal dil işleme alanında devrim niteliğinde bir gelişme sağlayarak günümüzdeki büyük dil modellerinin temelini oluşturmuştur.

Üretken Yapay Zekâ Dönemi ve Gelecek (2020-Günümüz)

Yapay zekâ teknolojileri, 2020'li yıllardan itibaren üretken yapay zekâ (Generative AI) dönemine girmiş ve özellikle OpenAI tarafından geliştirilen GPT serisi gibi büyük dil modelleri; metin, görsel ve kod üretme becerileriyle günlük yaşamın parçası hâline gelmiştir. Bu gelişmeler, yapay zekânın günlük yaşamı doğrudan etkileyen bir teknolojiye dönüşmesi açısından önemlidir. Günümüzde çok modlu sistemler gelişmekte ve yapay zekâ teknolojileri yaşamın her alanında giderek daha etkin olmaktadır. Bu durum yapay zekâ kullanımında etik, güvenlik gibi tartışmaları da beraberinde getirir (Görsel 1.6).



Görsel 1.6: Yapay zekânın tarihsel gelişim sürecindeki dönüm noktaları

Yapay zekânın tarihsel gelişimi doğrusal bir ilerleme göstermemiştir. Bazı dönemlerde hızlı gelişmeler yaşanırken bazı dönemlerde duraklamalar olmuştur. Bu süreç, teknolojinin kesintili ancak kümülatif biçimde ilerlediğini göstermektedir.

NOTLAR

1.1. ETKİNLİK

Adı	Yapay Zekânın İnsan Yaşamına Katkıları Sunumu
Süre	1 ders saati
Amaç	Yapay zekânın temel özelliklerini araştırarak bu özelliklerinden birinin insan yaşamına katkılarını açıklamak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, internet erişimi, sunum hazırlama yazılımı (PowerPoint, Google Slides vb.), güvenilir bilgi kaynakları
Değerlendirme	Yapmış olduğunuz çalışmalar öğretmeniniz tarafından karekodda yer alan kontrol listesi kullanılarak değerlendirilecektir. Çalışmalarınızı bu ölçütlere göre planlayınız.

Hazırlık

- Bilgisayarınızı ve kullanacağınız sunum programını açınız.
- Yapay zekânın temel özellikleri hakkında ön bilgi ediniz.
- Çalışma ortamınızın düzenli ve hazır olduğundan emin olunuz.
- İş sağlığı ve güvenliği kurallarına uygun şekilde çalışma ortamınızı düzenleyiniz.
- Sunum dosyanızı oluşturunuz ve uygun isimle kaydediniz.

Etkinlik Sırasında Uyulması Gereken Kurallar

- Araştırma sürecinde güvenilir ve güncel kaynaklardan yararlanınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Kaynaklardan elde ettiğiniz bilgilerin doğruluğunu sorgulayınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Sunum hazırlarken görsel ve metinler arasında denge kurunuz.
- Sunum sırasında açık, anlaşılır ve etkili bir dil kullanınız.

Çalışma Yönergesi

1. Yapay zekânın temel özelliklerini inceleyiniz ve bir özelliği seçerek çalışma planı oluşturunuz.
2. Seçtiğiniz özellik hakkında en az üç güvenilir kaynaktan bilgi toplayınız; bilgilerin doğruluğunu karşılaştırınız.
3. Özelliğin gerçek hayattaki kullanımını ve insan yaşamına katkılarını neden-sonuç ilişkisiyle analiz ediniz.
4. Seçtiğiniz örnek üzerinden katkıları açıklayınız ve bu değeri artırmaya yönelik öneriler geliştiriniz.
5. Sunumunuzda kısa, anlaşılır metinler ve içeriği destekleyen görseller kullanınız.
6. Çalışmanızı süresinde tamamlayarak sunumunuzu açık ve etkili biçimde yapınız.

Kontrol listesine ulaşmak için karekodu okutunuz.
Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

1.1.3. Yapay Zekânın Alt Dalları

Yapay zekâ, ortaya çıktığı günden bu yana bilimsel çalışmaların önemli parçası olmuş; yeni algoritmalar, artan veri miktarı ve donanım gücü sayesinde araştırmacılar tarafından sürekli geliştirilmiştir. Bu yenilenme; yaklaşımların, tekniklerin ve algoritmaların geliştirilmesine katkı sağlamış, aynı zamanda yapay zekâ uygulamalarının günlük yaşamda daha yaygın kullanılmasına olanak tanımıştır. Bu süreçte, farklı görevleri yerine getirebilmek amacıyla geliştirilen yaklaşım ve teknikler, yapay zekânın farklı alt dallarının ortaya çıkmasını sağlamıştır. Her bir alt dal belirli problem türlerine çözüm üretmeye odaklansa da alt dallar çoğu zaman birlikte çalışarak daha kapsamlı ve etkili çözümler sunmaktadır. Yapay zekâ, tek yapıdan oluşan bir alan değildir. Farklı problem türlerine çözüm üretmek üzere gelişmiş çeşitli alt dalları kapsar. Alt dallar kimi zaman ayrı yöntemler olarak kullanılsa da çoğunlukla bir araya gelerek kapsamlı ve etkili çözümler sunar. Yapay zekânın alt dalları Şema 1.2'de verilmiştir.



Şema 1.2: Yapay zekânın alt dalları

Makine Öğrenmesi

Bilgisayarların veriler ve algoritmalar aracılığıyla öğrenmesini sağlayan yapay zekâ alt dalıdır. Bu yaklaşımda sistemler, deneyimlerden yararlanarak veri içerisindeki örüntüleri keşfeder ve ortaya çıkacak yeni durumlar hakkında tahminlerde bulunur. Bu alanın temel amacı, karmaşık verilerden anlamlı sonuçlar çıkarıp tahmin yapmak ve karar verme süreçlerini otomatik hâle getirmektir. Geleneksel programlamadan farklı olarak sonuca ulaşmak için tüm adımların açıkça kodlanması gerekmez. Veriler kullanılarak bir model oluşturulur ve süreç bu model üzerinden yürütülür. Kullanılan verilerin kalitesi, temsil gücü ve modele uygunluğu arttıkça sistemin daha doğru sonuçlar üretme olasılığı artar. Günümüzde makine öğrenmesi; öneri sistemleri, e-posta filtreleme, hava durumu tahmini gibi birçok alanda kullanılmaktadır. Derin öğrenme gibi gelişmiş yöntemleri de kapsayan bu alan, modern teknolojinin verilerden öğrenerek gelişen sistemi olarak öne çıkar.

Bilgisayarlı Görü

Bilgisayarların ve yapay zekâ sistemlerinin görüntü ve videoları analiz ederek anlamlandırmasını sağlayan yapay zekâ alt dalıdır. Bu alanda geliştirilen sistemler; nesneleri tanıyabilir, yüzleri ayırt edebilir, görüntülerdeki karmaşık örüntüleri tespit edebilir ve sahneler hakkında çıkarım yapabilir. Bilgisayarlı görü uygulamaları, kamera ve sensörlerden elde edilen verileri işleyerek çevre hakkında bilgi üretir. Günümüzde bu teknoloji;

yüz tanıma sistemleri, güvenlik uygulamaları, tıbbi görüntü analizi, otonom (sürücüsüz) araçlar, nesne tespit sistemleri gibi birçok alanda kullanılmaktadır. Gelişen yapay zekâ algoritmaları ve artan veri işleme kapasitesi sayesinde bilgisayarlı görü uygulamalarının doğruluk ve başarı düzeyi önemli ölçüde artmıştır.

Robotik

Yapay zekânın fiziksel ortamda somutlaşmış hâlidir. Robotların çevrelerini algılayıp belirli görevleri yerine getirmesini amaçlayan alt alandır. Bu alanın temel amacı, makinelerin sensörler aracılığıyla çevrelerinden veri toplayıp bu verileri işleyerek hareket edebilmesini sağlamak; üretim, sağlık, ulaşım, keşif gibi alanlarda karşılaşılan problemleri çözmektir. Robotlar; ısı, ışık, ses, hareket, engel gibi çevresel bilgileri algılayan sensörlere, bu verileri işleyen işlemcilere ve görevleri yerine getiren mekanik yapılara sahiptir. Bu sayede endüstriyel üretimden tıbbi ameliyatlara, uzay araştırmalarından tarımsal uygulamalara kadar birçok alanda kullanılır. Robotik, yapay zekânın diğer alt dallarıyla da yakından ilişkilidir. Örneğin bir robotta bilgisayarlı görü çevrenin algılanmasına, makine öğrenmesiyse deneyimlerin öğrenilmesine katkı sunar. Böylelikle robot, çalışma verimini sürekli artırır. Diğer yapay zekâ alanlarından farklı olarak fiziksel ortamda doğrudan etkileşim gerektiren robotik, modern teknolojinin gerçek dünyada görev yapan gücü olarak öne çıkar.

Uzman Sistemler

Belirli alanda, uzmanların bilgi ve deneyimlerini taklit ederek karar verebilen yapay zekâ tabanlı sistemlerdir. Sistemin temel amacı, makinelerin uzman bilgisine dayalı olarak akıl yürütmesini sağlamak; tıbbi teşhis, finansal değerlendirme, teknik destek, planlama gibi karar vermeyi gerektiren problemleri çözmektir. Uzman sistemler, belirli bir konuya ilişkin bilgileri içeren bilgi tabanı ve bu bilgileri kullanarak sonuçlara ulaşan çıkarım motoru olmak üzere iki temel bileşenden oluşur. Kullanıcıdan alınan veriler işlenerek uygun öneri veya kararlar üretilir. Bu sistemler, genellikle "eğer–ise–o hâlde" kurallarına dayalı, mantıksal çıkarımlar yaparak çalışır ve belirli alanda derinleşmiş bilgiyle yüksek doğrulukta sonuçlar sunabilir. Bu yönüyle uzman sistemler, yapay zekânın diğer alt dallarıyla da ilişkilidir. Örneğin makine öğrenmesi yöntemleri sistemin bilgi tabanını günceller, uzman sistemlerse mevcut bilgiyi kullanarak doğrudan çözüm üretmeye odaklanır. Veri temelli öğrenmeye dayalı sistemlerden farklı olarak daha çok kural tabanlı çalışan bu alan, modern teknolojinin belirli konularda danışman aklı olarak öne çıkar.

Doğal Dil İşleme

Doğal dil işleme; bilgisayarların insan dilini anlamasını, yorumlamasını ve üretmesini sağlayan yapay zekâ alt dalıdır. Bu alan; yazılı ve sözlü iletişim verilerinin analiz edilmesi, anlamlandırılması ve uygun şekilde işlenmesi üzerine çalışır. Doğal dil işleme sistemleri; metinleri sınıflandırabilir, çeviri yapabilir, soruları yanıtlayabilir ve konuşmaları analiz edebilir. Günümüzde sesli asistanlar, sohbet robotları, makine çevirisi, metin özetleme ve duygu analizi uygulamaları bu teknolojinin yaygın kullanım alanları arasında yer almaktadır. Gelişen yapay zekâ algoritmaları sayesinde doğal dil işleme uygulamaları, insanlara her geçen gün daha akıcı ve içeriğe uygun yanıtlar verir.

Derin Öğrenme

Makine öğrenmesinin alt dalı olup çok katmanlı, yapay sinir ağları aracılığıyla büyük ve karmaşık verilerden anlamlı sonuçlar elde etmeye çalışır. Bu alanın temel amacı, makinelerin verilerdeki gizli ilişkileri otomatik olarak keşfetmesini sağlamak; görüntü tanıma, konuşma tanıma, doğal dil işleme gibi problemleri yüksek doğrulukla çözmektir. Derin öğrenme sistemleri, insan beynindeki nöron yapısından esinlenen çok sayıda katmandan oluşur ve veriyi bu katmanlarda aşamalı olarak işler. Bu sayede geleneksel makine öğrenmesi yöntemlerinden farklı olarak verinin özelliklerini (renk, şekil, ses tonu vb.) insan müdahalesine daha az ihtiyaç duyarak kendisi belirleyebilir. Çok büyük veri setleriyle daha başarılı sonuçlara ulaşan bu yöntem, yüksek işlem gücüne sahip donanımlar özellikle grafik işlemciler (GPU) gerektirir. Bu yönüyle derin öğrenme, yapay zekânın öğrenen beyni olarak bilgisayarlı görü, doğal dil işleme gibi alanların temel itici gücü hâline gelmiştir.

1. SIRA SİZDE

Aşağıda verilen yapay zekâ alt dallarıyla bu alanlara ait özellikleri inceleyiniz. Her alt dalı, doğru açıklamayla eşleştirerek ilgili kutucuğa uygun harfi yazınız.

1	ROBOTİK		A — İnsan beynindeki nöronları taklit eden çok katmanlı yapılar ve yüksek işlem gücü (GPU) gereksinimi
2	MAKİNE ÖĞRENMESİ (ML) (Machine Learning)		B — Geleneksel programlamanın aksine, verilerdeki örüntüleri kullanarak tahmin yürüten temel model
3	DERİN ÖĞRENME (DL) (Deep Learning)		C — "Eğer–ise–O hâlde" kurallarına dayalı, uzman bir kişinin bilgi birikimini kopyalayan sistemler
4	BİLGİSAYARLI GÖRÜ (Computer Vision)		D — Makinelerin görsel verileri (fotoğraf, video, uydu görüntüsü) analiz ederek çevresini anlamlandırması
5	DOĞAL DİL İŞLEME (NLP) (Natural Language Processing)		E — Yazılı ve sözlü insan dilini analiz eden, metinlerden anlam çıkaran "anlayan dil" teknolojisi
6	UZMAN SİSTEMLER (Expert Systems)		F — Sensörler aracılığıyla çevresini algılayan, yapay zekâ destekli mekanik hareketlerle birlikte fiziksel eylem gerçekleştiren teknoloji
			G — Veri güvenliği, ağ yönetimi ve sistem bakımı gibi teknik süreçleri yöneten altyapı çözümleri

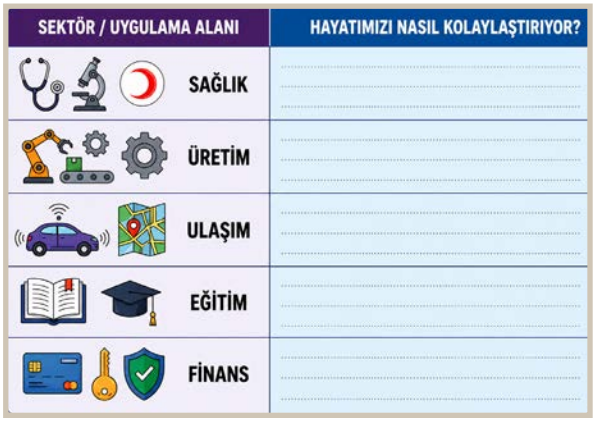
1.1.4. Yapay Zekânın Kullanım Alanları

Günümüzde yapay zekâ, yalnızca teknoloji şirketlerinin kullandığı alan olmaktan çıkmış; eğitimden sağlığa, finans sistemlerinden üretim tesislerine kadar yaşamın birçok alanında kullanılan teknoloji hâline gelmiştir. Yapay zekâ, doktorların çok sayıda tetkik arasında riskli durumları belirlemesine katkı sağlayarak teşhis süreçlerine dâhil olmuş, fabrikalarda üretim bandındaki olası hataları önceden tahmin eden sistemler aracılığıyla üretim süreçlerinde yerini almıştır. Benzer şekilde bankacılıkta çok sayıda finansal işlemi analiz edip güvenliği artırırken öğrenme ihtiyaçlarına ve bireysel yeteneklere göre içerik sunan uygulamalarla eğitim süreçlerine katılmıştır. Bu gelişmeler, yapay zekânın kullanım alanlarının ve sağladığı faydaların daha yakından incelenmesini gerekli kılmaktadır. Bu örnekler yapay zekânın sağlık, eğitim, finans ve daha birçok farklı alanda kullanılan ve günlük yaşamın önemli parçası hâline gelen bir teknoloji olduğunu gösterir.

NOTLAR

2. SIRA SİZDE

Aşağıdaki tabloda yapay zekânın kullanıldığı temel sektörler verilmiştir. Günlük yaşam gözlemlerinizden yararlanarak yapay zekânın bu alanlarda işleyişi nasıl değiştirdiğini ve yaşamı nasıl kolaylaştırdığını ilgili bölümlere yazınız.



1.1.5. Güncel Yapay Zekâ Uygulamaları ve Faydaları

Yapay zekâ alanında yaşanan gelişmeler çeşitli faydalar sunarak bireysel ve kurumsal süreçlere katkıda bulunur. Günümüzde yapay zekâ sağlık, eğitim, finans, ulaşım gibi birçok alanda kullanılmaktadır. Bu alanlara ilişkin örnek uygulamalar şunlardır:

Sağlık-Teşhiste Dijital Dönüşüm

Yapay zekâ, özellikle görüntü işleme yetenekleri sayesinde sağlık sektöründe teşhis süreçlerini güçlendirmiştir. Çok sayıda tıbbi veriyi kısa sürede analiz edebilen sistemler, uzmanların gözden kaçırabileceği detayları belirleyerek hastalıkların erken teşhisine katkı sağlar. Bu durum, tedavi sürecinin daha erken başlamasına imkân tanıyarak sağlık hizmetlerinin verimliliğini artırır.

Eğitim-Bireyselleşen Öğrenme Süreci

Geleneksel eğitimde tüm öğrenciler aynı hızda ilerlerken yapay zekâ destekli sistemler öğrenme sürecini bireysel ihtiyaçlara göre şekillendirir. Öğrencilerin eksik olduğu konuları belirleyen uygulamalar, onlara uygun içerikler sunarak öğrenmeyi daha etkili hâle getirir. Böylece her öğrenci kendi hızına uygun öğrenme deneyimi yaşayabilir.

Finans-Güvenli ve Akıllı Sistemler

Finans alanında yapay zekâ, büyük veri yığınlarını kısa sürede analiz ederek şüpheli işlemleri tespit eder ve güvenliği artırır. Bu sayede dolandırıcılık riski azaltılırken finansal işlemler daha güvenli hâle gelir. Ayrıca bireylerin harcama alışkanlıklarını analiz eden sistemler, bilinçli finansal kararlar alınmasına katkı sağlar.

Ulaşım-Güvenli ve Verimli Hareketlilik

Ulaşım sektöründe yapay zekâ, otonom araçlar ve akıllı trafik sistemleriyle önemli bir dönüşüm sağlar. Sensör ve kameralar aracılığıyla çevresini algılayan sistemler, insan hatasını azaltmayı hedefler. Trafik yoğunluğunu anlık olarak yöneten akıllı sistemlerse ulaşımı daha düzenli hâle getirerek zaman ve enerji tasarrufu sağlar.

Üretim-Akıllı ve Verimli Sanayi

Üretim alanında yapay zekâ, makinelerin performansını sürekli izleyerek verimliliği artırır. Olası arızaları önceden tahmin eden sistemler sayesinde üretim kesintiye uğramaz. Üretim sürecindeki hataları anında tespit eden teknolojilerse ürün kalitesini ve sanayide rekabet gücünü artırır.

Eğlence-Kişiselleştirilmiş Deneyim

Eğlence alanında yapay zekâ, kullanıcıların ilgi alanlarını analiz ederek onlara uygun içerikler sunar. İzlenen içerikler doğrultusunda yapılan öneriler, kullanıcıların zaman kaybetmeden ilgilerine uygun seçeneklere ulaşmasını sağlar. Böylece dijital deneyim daha verimli ve kişiselleştirilmiş hâle gelir.

Yapay Zekânın Somut Zaferleri, Gerçek Dünya Başarı Hikâyeleri

Yapay zekânın sunduğu teorik imkânlar, birçok farklı sektörde başarı hikâyelerine dönüşmüştür. Bu teknoloji bazen bir doktorun gözden kaçırabileceği sağlık detayını yakalamakta bazen de bir fabrikanın üretim hattındaki arızayı günler öncesinden öngörmektedir. Yapay zekâ birçok önemli alanda işleyişi kökten değiştiren ve insan yaşamını kolaylaştıran çözüm ortağıdır (Görsel 1.7).



Görsel 1.7: Yapay zekânın somut zaferleri

BİLGİ KUTUSU

2019 yılında İngiltere'de yapılan rutin bir sağlık kontrolünde, bir hastanın mamografi görüntülerinde ilk incelemede belirgin bir bulguya rastlanmadı. Ancak görüntüler yapay zekâ destekli bir karar destek sistemiyle yeniden analiz edildiğinde, sistem çok küçük ve fark edilmesi güç şüpheli bir alanı işaretledi. Bunun üzerine yapılan ayrıntılı incelemeler sonucunda meme kanseri erken evrede tespit edildi ve tedavi sürecine zaman kaybetmeden başlandı. Bu örnek, yapay zekânın doktorların yerine karar veren bir teknoloji olmadığını; görüntüleri dikkatle tarayarak hekimlere destek olan, erken teşhis olasılığını artıran önemli bir yardımcı sistem olduğunu göstermektedir.

1.2. ETKİNLİK

Adı	Yapay Zekânın Sektörel Kullanım Alanları Sunumu
Süre	1 ders saati
Amaç	Seçtiğiniz bir sektörde (sağlık, eğitim, finans, ulaşım, üretim veya eğlence) yapay zekânın güncel ve potansiyel kullanım alanlarını açıklamak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, internet erişimi, sunum hazırlama yazılımı (PowerPoint, Google Slides vb.), güvenilir bilgi kaynakları
Değerlendirme	Yapmış olduğunuz çalışmalar öğretmeniniz tarafından karekodda yer alan kontrol listesi kullanılarak değerlendirilecektir. Çalışmalarınızı bu ölçütlere göre planlayınız.

Hazırlık

- Bilgisayarınızı ve kullanacağınız sunum programını açınız.
- Yapay zekânın farklı sektörlerdeki kullanım alanları hakkında ön araştırma yapınız.
- Çalışma ortamınızı düzenleyiniz ve gerekli materyalleri hazırlayınız.
- Sunum dosyanızı oluşturunuz ve uygun isimle kaydediniz.

Etkinlik Sırasında Uyulması Gereken Kurallar

- Araştırmalarınızda güvenilir ve güncel kaynaklardan yararlanınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Kaynaklardan elde ettiğiniz bilgilerin doğruluğunu sorgulayınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Sunum hazırlarken görseller ve metinler arasında denge kurunuz.
- Sunum sırasında açık, anlaşılır ve etkili bir dil kullanınız.

Çalışma Yönergesi

1. Yapay zekânın farklı sektörlerdeki kullanım alanlarını inceleyiniz. Bir sektör seçerek çalışma planı oluşturunuz.
2. Seçtiğiniz sektörde yapay zekânın kullanımını en az üç güvenilir kaynaktan araştırınız ve bilgileri karşılaştırınız.
3. Yapay zekânın sektördeki etkilerini ve sağladığı kolaylıkları neden-sonuç ilişkisi kurarak analiz ediniz.
4. Seçtiğiniz sektör için yapay zekânın gelecekteki kullanımına yönelik en az iki öneri geliştiriniz ve günlük yaşama etkilerini açıklayınız.
5. Elde ettiğiniz bilgileri kısa, anlaşılır metinler ile görsel ve grafikler kullanarak sunu hâline getiriniz.
6. Çalışmanızı süresinde tamamlayarak sunumunuzu açık, anlaşılır ve etkili bir şekilde gerçekleştiriniz.

Kontrol listesine ulaşmak için karekodu okutunuz.

Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

1.1.6. Yapay Zekâ Geliştirme Ortamlarının Kurulumu

Önceki bölümde yapay zekânın temel kavramları ele alınmıştır. Bu bölümdeyse yapay zekâ uygulamalarının geliştirilme süreci ve bu süreçte kullanılan araçlar incelenecektir. Yapay zekâ sistemlerini geliştirmek yalnızca bir algoritma yazmaktan ibaret değildir. Bu süreçte farklı amaçlara hizmet eden çeşitli araçlardan yararlanılır. Bu araçlar genel olarak şunlardır:

- Programlama dilleri
- Geliştirme ortamları
- Yazılım kütüphaneleri ve araçları

Yapay zekâ geliştirme sürecinde birden fazla programlama dili kullanılabilir. Bu diller arasında yaygın olarak tercih edileni **Python** programlama dilidir. Python dilinin öne çıkmasının temel nedenleri; öğrenme kolaylığı, anlaşılır söz dizimi ve yapay zekâ çalışmalarına yönelik zengin kütüphane desteğidir.

Bu özellikleri sayesinde Python; veri işleme, makine öğrenmesi ve derin öğrenme uygulamalarının geliştirilmesinde geniş kullanım alanına sahiptir. Bu nedenle, bu eğitim materyalindeki uygulamalarda Python programlama dili kullanılacaktır.

Bu diller, sahip oldukları özelliklere göre farklı problem türlerinde avantaj sağlar. Yaygın kullanılan bazı programlama dilleri ve bu dillerin genel özellikleri şunlardır:

R: İstatistiksel analiz ve veri görselleştirme konusunda güçlüdür. Daha çok akademik çalışmalar ve veri analizi odaklı projelerde kullanılır.

Java: Platformdan bağımsız çalışabilmesi ve yüksek performans sunması nedeniyle büyük ölçekli sistemlerde tercih edilir.

C++ ve C#: Donanıma yakın çalışabilmesi sayesinde yüksek performans gerektiren uygulamalarda kullanılır. Özellikle oyun yapay zekâsı ve gerçek zamanlı sistemlerde etkilidir.

JavaScript: Web ortamında doğrudan çalışır ve kullanıcı etkileşimi gerektiren yapay zekâ uygulamalarında kullanılır.

Scala: Büyük veri ve dağıtık sistemler üzerinde çalışan yapay zekâ uygulamalarında tercih edilir.

Lisp ve Prolog: Mantıksal çıkarım ve sembolik yapay zekâ çalışmalarında kullanılır.

Julia: Bilimsel hesaplamalarda yüksek performans sunar, sayısal analizlerde kullanılır.

Go: Hızlı ve verimli çalışması sayesinde dağıtık sistemlerle altyapı geliştirme süreçlerinde yer alır.

Bu dillerin her biri farklı avantajlara sahip olsa da kullanım alanları problemin türüne göre değişiklik gösterir. Bu nedenle yapay zekâ geliştirme sürecinde önemli olan probleme en uygun dili seçmektir.

NOTLAR

3. SIRA SİZDE

Aşağıda farklı yapay zekâ projelerine ait senaryolar verilmiştir. Her senaryo için en uygun programlama dilini belirleyiniz ve seçiminizi gerekçesiyle açıklayınız.

Senaryolar

1. Araştırma ekibi, büyük bir veri seti üzerinde istatistiksel analiz yaparak sonuçları grafiklerle sunmak istemektedir.
2. Teknoloji firması, milyonlarca kullanıcıya hizmet verecek büyük ölçekli bir yapay zekâ sistemi geliştirmektedir.
3. Oyun geliştirici firma, gerçek zamanlı çalışan ve yüksek performans gerektiren bir yapay zekâ sistemi tasarlamaktadır.
4. Web geliştirici, tarayıcı üzerinde çalışan ve kullanıcıyla etkileşim kuran bir yapay zekâ uygulaması geliştirmek istemektedir.
5. Bir bilim insanı, sayısal hesaplamaların yoğun olduğu bir makine öğrenmesi modeli üzerinde çalışmaktadır.

Yapmanız Beklenenler

- Her senaryo için uygun programlama dilini seçiniz.
- Seçiminizi avantaj, dezavantaj ve kullanım alanı bilgilerine dayanarak gerekçelendiriniz.
- Cevaplarınızı aşağıdaki tabloya yazınız.

Senaryo	Seçilen Programlama Dili	Gerekçe
1		
2		
3		
4		
5		

Python Dilinin Yapay Zekâ Projelerindeki Rolü

Yapay zekâ projelerinde veri işleme, model oluşturma ve sonuçları değerlendirme gibi süreçler belirli aşamalardan oluşur. Bu aşamaların planlı ve sistematik şekilde yürütülmesi, geliştirilen modelin doğruluğunu ve güvenilirliğini etkiler.

Python, bu sürecin tüm aşamalarında etkin biçimde kullanılabilen bir programlama dilidir. Geniş kütüphane desteği sayesinde veri analizi, model eğitimi ve performans değerlendirme işlemleri tek bir ortamda gerçekleştirilebilir. Bu durum, Python dilini yapay zekâ geliştirme sürecinde güçlü bir araç hâline getirir.

Bir yapay zekâ uygulaması geliştirilirken genellikle izlenen adımlar şunlardır:

Veri Setinin Elde Edilmesi: Problemi temsil eden uygun verilerin toplanması

Verinin Hazırlanması: Eksik, hatalı veya düzensiz verilerin düzenlenmesi

Modelin Eğitilmesi: Seçilen algoritmanın veriler üzerinden öğrenmesi

Modelin Test Edilmesi: Modelin daha önce görmediği veriler üzerindeki başarısının ölçülmesi

Sonuçların Değerlendirilmesi: Model performansının analizi

Bu süreci gösteren basit bir uygulama örneği şöyledir:

Örnek

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
veri = load_iris()
X = veri.data
y = veri.target
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
model = KNeighborsClassifier()
model.fit(X_train, y_train)
tahmin = model.predict(X_test)
print (Doğruluk oranı:), accuracy_score(y_test, tahmin))
```

Bu koda göre yapay zekâ geliştirme sürecinin aşamaları şunlardır:

- load_iris() → Veri setinin elde edilmesi
- X ve y → Giriş verileri ve hedef değişkenin belirlenmesi
- train_test_split() → Verinin eğitim ve test olarak ayrılması
- model.fit() → Modelin eğitilmesi
- model.predict() → Tahmin yapılması
- accuracy_score() → Model performansının değerlendirilmesi

Bu yapı, yapay zekâ projelerinde kullanılan sürecin temel örneğini oluşturur. Gerçek uygulamalarda bu süreç daha karmaşık hâle gelebilir. Veri ön işleme, özellik seçimi ve farklı algoritmaların karşılaştırılması gibi ek adımlar da sürece dâhil edilir. Bu tür örnekler, model oluşturmanın yanında modelin nasıl geliştirildiğinin ve değerlendirildiğinin anlaşılmasını sağlar. Bu nedenle Python, yapay zekâ projelerinde yalnızca programlama dili değil aynı zamanda tüm süreci yönetebilen bütünleşik geliştirme aracıdır.

NOTLAR

4. SIRA SİZDE

1. Aşağıda yapay zekâ uygulama geliştirme sürecine ait kavramlar verilmiştir. Bu kavramları inceleyiniz ve uygun aşamalarla eşleştirerek tabloya yerleştiriniz.

Kavramlar

Veri toplama – Veri temizleme – Eğitim verisi – Test verisi – Model eğitimi – Tahmin – Doğruluk oranı

Yapay Zekâ Süreci Aşamaları	İlgili Kavramlar
Veri setinin elde edilmesi	
Verinin hazırlanması	
Modelin eğitilmesi	
Modelin test edilmesi	
Sonuçların değerlendirilmesi	

2. Python programlama dili, yukarıdaki hangi aşamalarda kullanılmaktadır? Açıklayınız.

.....
.....
.....
.....
.....
.....
.....

Yapay zekâ uygulamaları geliştirmek için öncelikle uygun programlama dili seçilir. Ardından geliştirme ortamı ve gerekli araçların kurulumu gerçekleştirilir. Bu süreç, uygulama geliştirme çalışmalarının sağlıklı ve verimli ilerleyebilmesi için önemli bir hazırlık aşamasıdır.

Bu eğitim materyalinde geliştirme ortamı olarak **Anaconda** kullanılacaktır. Anaconda; veri bilimi, yapay zekâ ve makine öğrenmesi çalışmalarında yaygın olarak tercih edilen, içerisinde çok sayıda kütüphane, araç ve geliştirme ortamını (IDE) barındıran bütünleşik bir platformdur.

Anaconda kullanımının sağladığı avantajlar şunlardır:

- Gerekli kütüphaneler tek tek kurulmadan hazır olarak elde edilir.
- Farklı yazılım bileşenleri uyumlu şekilde çalışır.
- Kurulum ve yapılandırma süreci sadeleşir.

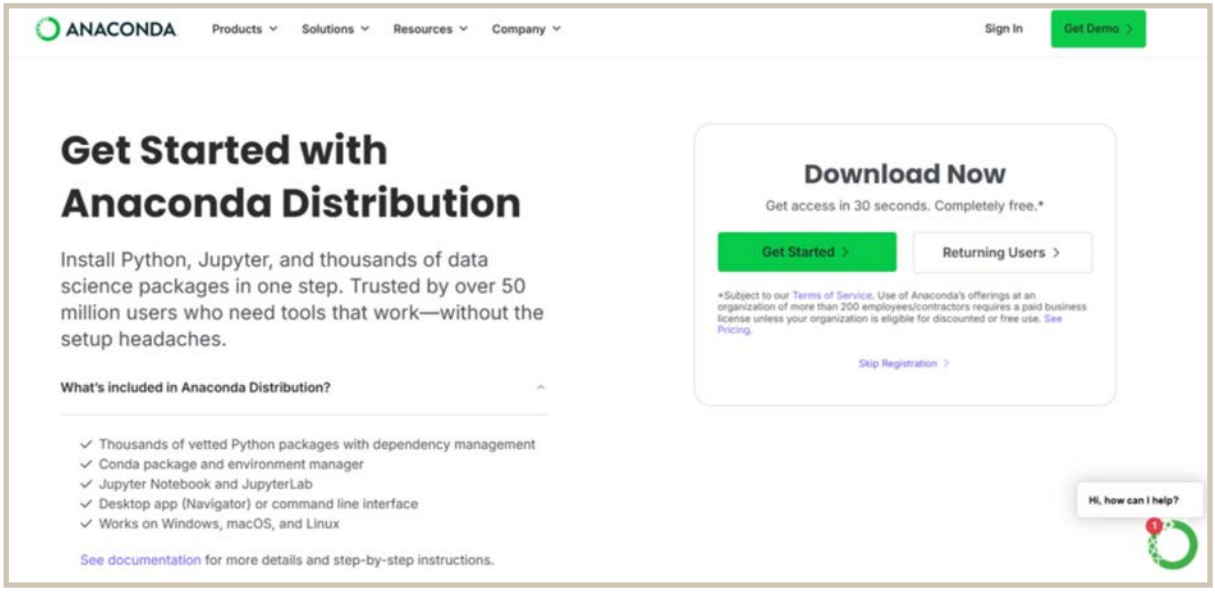
Bu özellikler, uygulama geliştirme sürecinde karşılaşılabilecek teknik sorunları azaltarak doğrudan problem çözmeye odaklanılmasını sağlar.

Yapay zekâ ve makine öğrenmesi uygulamaları geliştirmek amacıyla kullanılan **Anaconda Bütünleşik Geliştirme Ortamının (IDE)** kurulumu için izlenecek adımlar sırasıyla şunlardır:

1. Adım: Anaconda Kurulum Dosyasını İndirme

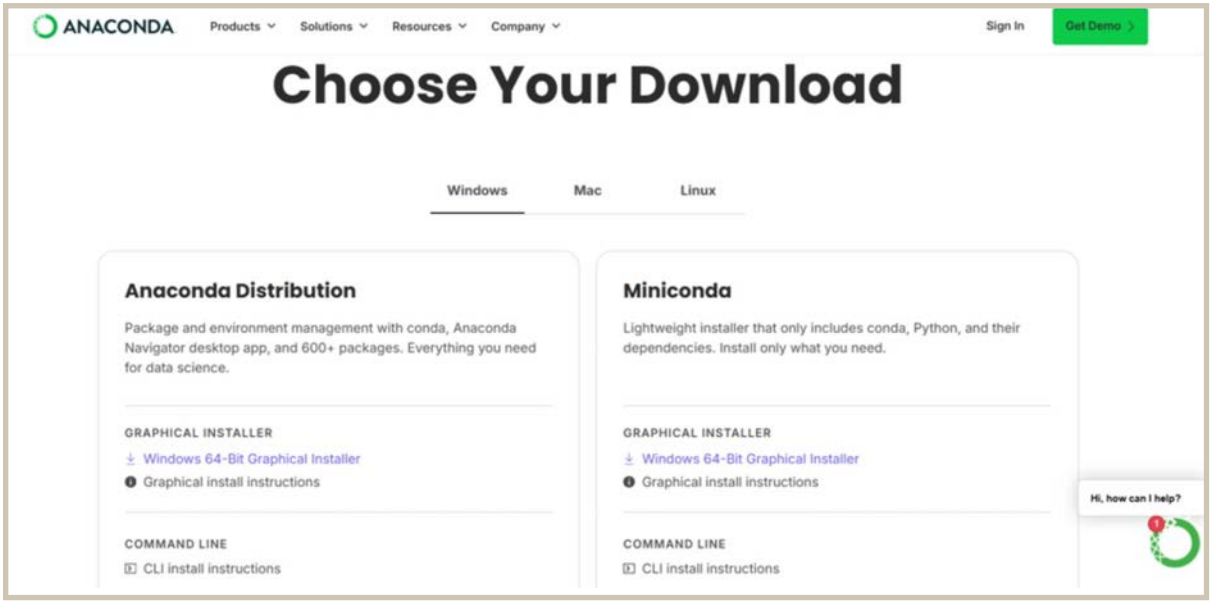
Anaconda kurulum dosyasını indirmek için bir internet tarayıcısı açılır ve adres çubuğuna https://www.anaconda.com/download yazılarak ilgili sayfaya erişim sağlanır.

Kullanılan işletim sistemine (Windows, macOS veya Linux) uygun kurulum seçenekleri açılan sayfada yer almaktadır (Görsel 1.8). İlgili işletim sistemine uygun kurulum dosyası seçilerek indirme işlemi başlatılır.



Görsel 1.8: Anaconda kurulum dosyasının indirildiği web sayfası

İndirme işlemi sırasında kullanıcının kayıt oluşturması istenebilir. Bu durumda **Skip Registration** seçeneği tercih edilerek kayıt adımı atlanır. Program doğrudan indirme sayfasına yönlendirir (Görsel 1.9). Açılan sayfada **Anaconda Distribution** indirme seçeneği görülür. Bu seçenek ilgili kurulum dosyasına erişim içindir. Bu bağlantıya tıklanarak kurulum dosyasının bilgisayara indirilmesi sağlanır.



Görsel 1.9: Anaconda Distribution indirme sayfası

BİLGİ KUTUSU

Anaconda Distribution ve Miniconda, Python kullanarak yapay zekâ uygulamaları geliştirmek için kullanılan iki farklı ortamdır.

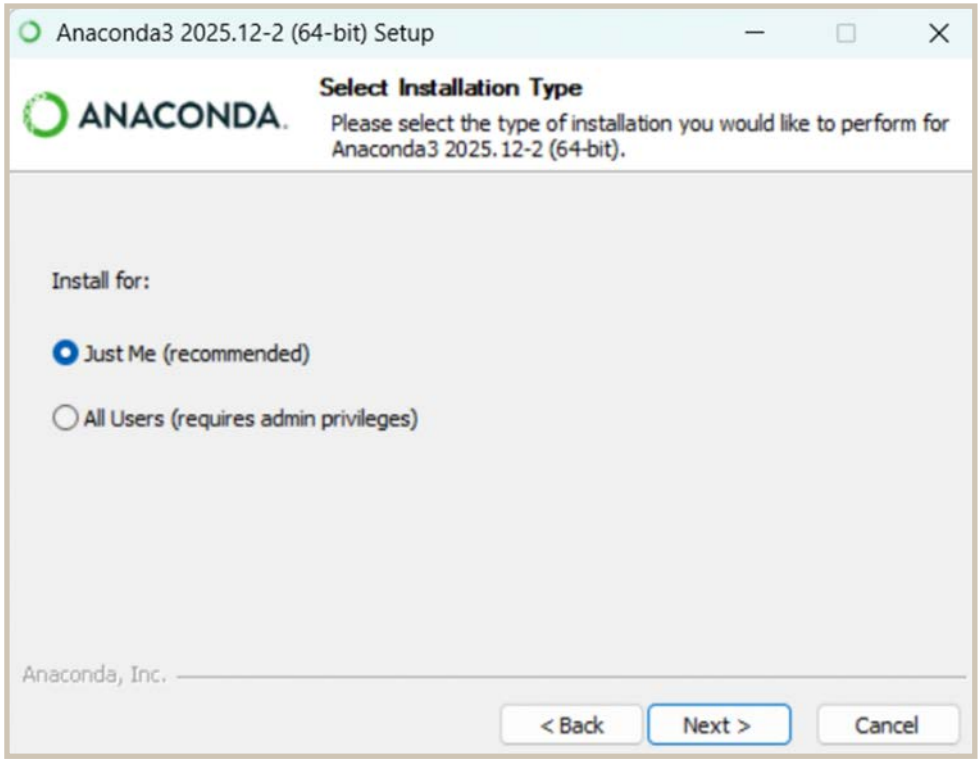
Anaconda Distribution	Miniconda
Avantajları • Hazır kütüphanelerle birlikte gelir. • Kurulduktan sonra hemen kullanılabilir. • Kullanımı kolaydır. Dezavantajı: Kurulduğu yerde fazla yer kaplar.	Avantajları • Sade bir yapıya sahiptir. • Gerekli kütüphaneler sonradan eklenir. • Kurulduğu yerde az yer kaplar. Dezavantaj: Kurulumu ve kullanımı zor olabilir.

Hangisi Tercih Edilmeli?

- Yeni başlanıyorsa → Anaconda Distribution
- Daha kontrollü ve hafif bir yapı isteniyorsa → Miniconda

2. Adım: Kurulum Dosyasını Açma ve Kurulum Türünü Seçme

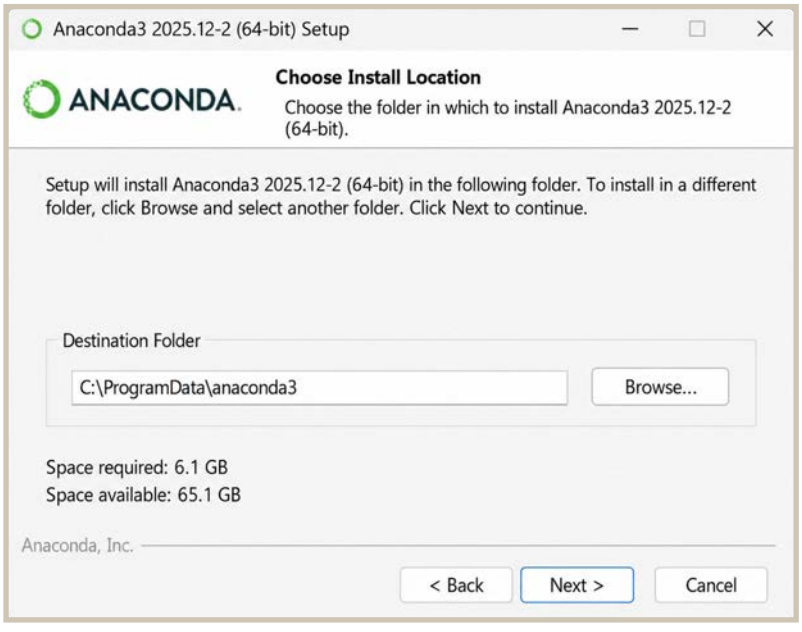
İndirilen kurulum dosyası çalıştırılarak kurulum sihirbazı başlatılır. İlk aşamada ekranda yer alan lisans sözleşmesi okunup kabul edilir. Farklı seçenekler içinden kurulum türü (Görsel 1.10) seçilip sonraki adıma geçilir.



Görsel 1.10: Anaconda kurulum türü seçim ekranı

3. Adım: Kurulum Klasörünü Belirleme

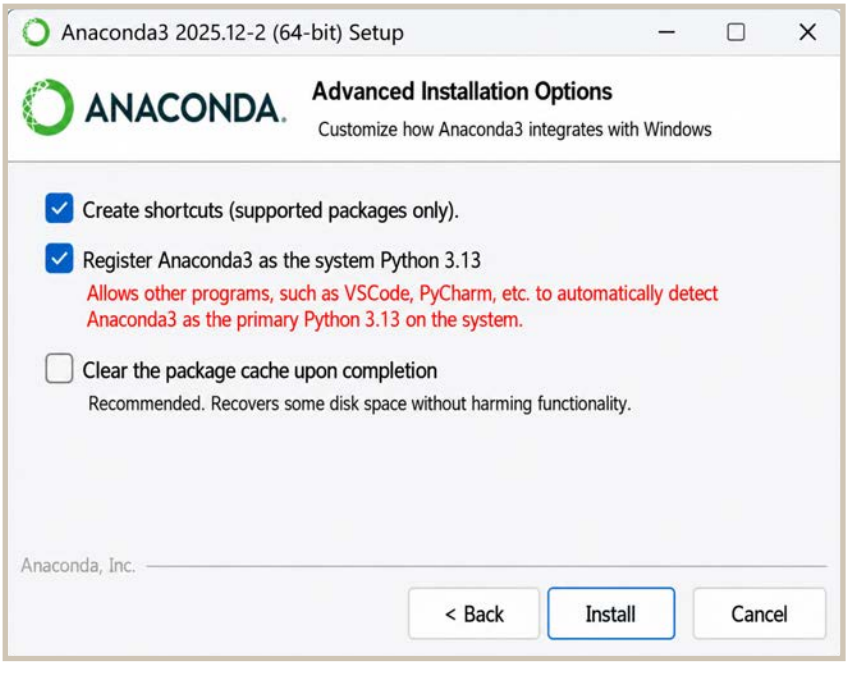
Bu adımda programın bilgisayarda kurulacağı klasör seçilir. Seçim ekranında varsayılan kurulum dizini hazır olarak sunulmuştur (Görsel 1.11). Bu dizin genellikle yeterlidir. Dizinin değiştirilmesine gerek yoktur. Farklı klasör seçilmek istenirse **Browse (Gözat)** butonuna tıklanarak istenen klasör belirlenebilir.



Görsel 1.11: Anaconda kurulum klasörü seçim ekranı

4. Adım: Gelişmiş Kurulum Ayarlarını Yapma

Bu adımda gelişmiş kurulum ayarlarının bulunduğu ekran açılır (Görsel 1.12). Bu ayarlarla Anaconda programının sistemde çalışma özellikleri belirlenir. Seçenekler arasında ortam değişkenleri (Path) ve Anaconda'nın varsayılan Python olarak belirlenmesi gibi ayarlar bulunur. Kurulumun sorunsuz tamamlanması için ayarların **varsayılan hâliyle bırakılması önerilir.** Son olarak **Install (yükle)** butonuna tıklanıp kurulum başlatılır.

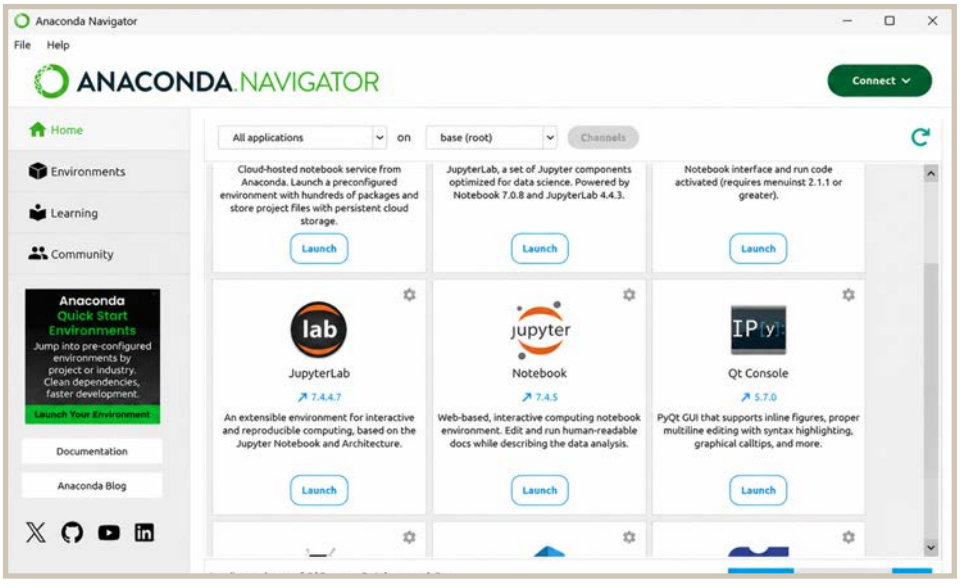


Görsel 1.12: Anaconda gelişmiş kurulum ayarları ekranı

5. Adım: Kurulumu Tamamlama ve Anaconda'yı Çalıştırma

Kurulum işlemi tamamlandığında ekranda kurulumun başarıyla bittiğini gösteren son pencere görüntülenir. Bu aşamada **Finish (bitir)** butonuna tıklanarak kurulum sihirbazı kapatılır. Ardından bilgisayarda **Anaconda Navigator** uygulaması açılır (Görsel 1.13). Anaconda Navigator arayüzünde Jupyter Notebook, Spyder ve benzeri geliştirme araçları yer alır.

Bu arayüz üzerinden **Jupyter Notebook** seçilerek Python kodlarının yazılıp çalıştırılabileceği geliştirme ortamına erişim sağlanabilir. Anaconda Navigator açıldığında ve Jupyter Notebook erişimi sağlandığında kurulum başarıyla tamamlanmış demektir.



Görsel 1.13: Anaconda Navigator ana ekranı

Jupyter Notebook, açık kaynaklı ve web tabanlı geliştirme ortamıdır. Bu ortamda yazılan kodların sonuçları anında görülebilir. Ayrıca veriler görselleştirilebilir ve işlemler adım adım takip edilebilir.

Jupyter Notebookla başta **Python** olmak üzere **R, C#** ve **PHP** gibi farklı programlama dilleri kullanılarak uygulamalar geliştirilebilir.

Jupyter Notebook açıldığında üç temel sekmeden oluşan bir ekranla (Görsel 1.14) karşılaşılır. Ekrandaki sekmeler şunlardır:



Görsel 1.14: Jupyter Notebook arayüzü ve sekmeleri

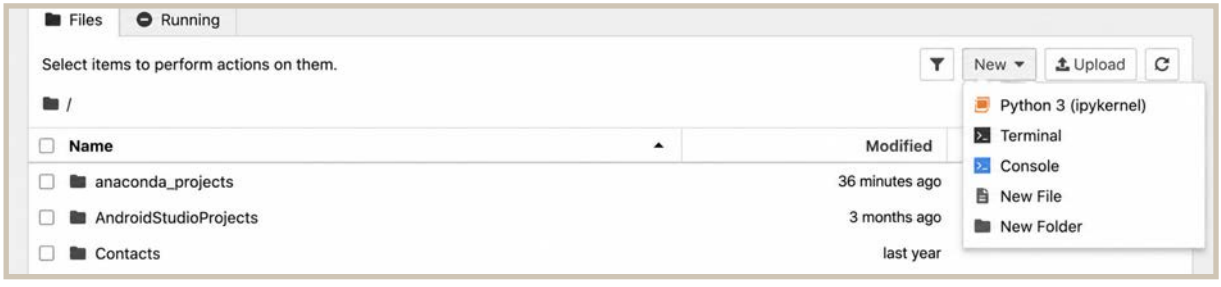
Files (Dosyalar): Bulunulan klasördeki dosya ve klasörleri görüntülemek için kullanılır.

Running (Çalışanlar): O anda açık ve çalışan Notebook dosyalarını gösterir.

Clusters (Kümeler): Notebook'un çalışmasını sağlayan arka plan işlemlerini (kernel) yönetmek için kullanılır.

Jupyter Notebook, kodların **adım adım çalıştırılabilmesi** ve sonuçların **anında görülebilmesi** sayesinde öğrenme sürecini kolaylaştırır. Bu nedenle yapay zekâ ve veri bilimi çalışmalarında yaygın olarak kullanılmaktadır.

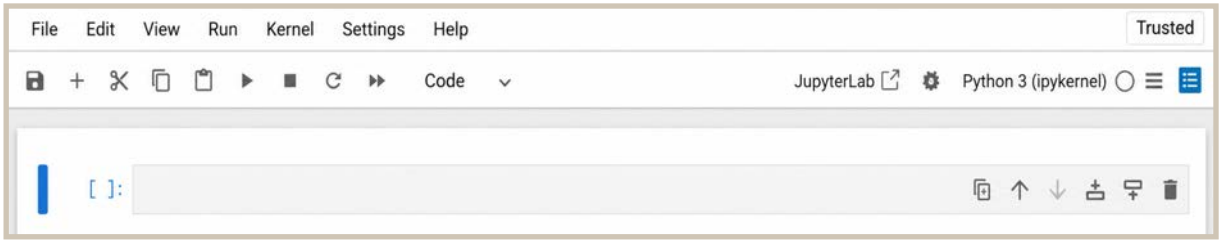
Yeni çalışma başlatmak için arayüzde yer alan **New (yeni)** butonuna tıklanarak yeni bir Notebook oluşturulur (Görsel 1.15).



Görsel 1.15: Yeni Notebook açılması

Bu işlem sonucunda seçilen programlama diline (genellikle Python) ait yeni bir çalışma dosyası açılır ve kod yazımına başlanabilir. Jupyter Notebook, varsayılan olarak **IPython çekirdeği (kernel)** ile çalışır. Çekirdek, yazılan kodları çalıştıran ve sonuçları kullanıcıya ileten temel bileşendir.

Farklı programlama dilleri için farklı çekirdekler kullanılabilir. Ancak bu eğitim materyalinde **Python** kullanılacağı için **IPython çekirdeği** tercih edilmektedir. Yeni bir Notebook oluşturulduğunda IPython çekirdeği otomatik olarak aktif olur ve Python kodları doğrudan çalıştırılabilir (Görsel 1.16).



Görsel 1.16: Yeni Notebook çalışma dosyası

Görsel 1.17 'de, Jupyter Notebook arayüzünde yer alan temel bileşenler numaralandırılmıştır. Bu bölümler şunlardır:



Görsel 1.17: Jupyter Notebook arayüzü

1. **Başlık ve Dosya Adı:** Notebook'un adı ve son kaydedilme bilgisi burada yer alır. Dosya adı bu bölümden değiştirilebilir.
2. **Menü Çubuğu:** File, Edit, View, Run, Kernel gibi menüler bulunur. Tüm işlemler bu menüler üzerinden yapılır.
3. **Araç Çubuğu (Toolbar):** Kaydetme, çalıştırma, hücre ekleme gibi sık kullanılan işlemler için kısayol butonları içerir.

4. **Çekirdek (Kernel) Bilgisi:** Kullanılan programlama dili ve çekirdek bilgisi burada gösterilir (örneğin: Python 3).

5. **Kod Hücresi (Cell):** Kodların yazıldığı ve çalıştırıldığı alandır. Her hücre ayrı ayrı çalıştırılabilir.

Jupyter Notebook ortamında yeni not defteri oluşturma, dosyayı yeniden adlandırma ve temel bir kodu çalıştırma işlemleri için izlenecek adımlar şunlardır:

1. Adım: Yeni Notebook Oluşturma

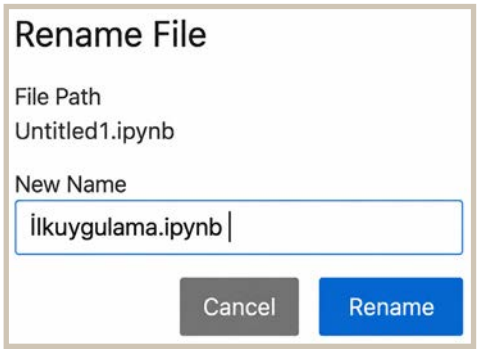
Jupyter Notebook ana sayfasında sağ üst köşede yer alan **New (yeni)** butonuna tıklanır. Açılan seçenekler arasından **Python 3** seçilerek yeni çalışma dosyası oluşturulur.

Bu işlem sonucunda, Python programlama diliyle çalışmaya uygun yeni Notebook dosyası açılır. Yeni sekmede açılan Notebook içerisinde kod yazmaya uygun boş bir hücre (cell) bulunur. Dosya adı varsayılan olarak **Untitled** şeklinde tanımlanmıştır.

Bu aşamada Python kodlarının yazılıp çalıştırılabileceği etkileşimli geliştirme ortamına erişilmiş olur. Açılan bu ekran, yapay zekâ ve veri bilimi uygulamalarının geliştirilmesi için temel çalışma alanını oluşturmaktadır.

2. Adım: Notebook Dosyasını Yeniden Adlandırma

Oluşturulan Notebook dosyasının anlamlı bir isimle kaydedilmesi, çalışma düzeni ve dosya yönetimi açısından önemlidir. Bunun için ekranın üst kısmında yer alan **Untitled (başlıksız)** ifadesine çift tıklanır (Görsel 1.18).



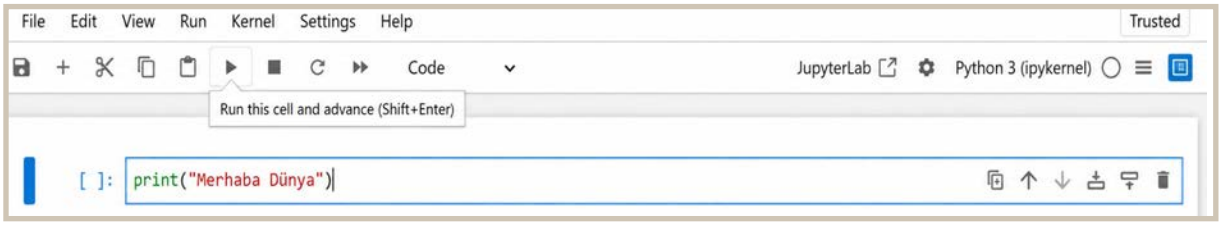
Görsel 1.18: Jupyter Notebook dosya adını değiştirme ekranı

Bu işlem sonucunda açılan pencerede dosya adı düzenlenebilir. Çalışmanın içeriğini yansıtacak bir isim (örneğin ilkuygulama.ipynb) girilir ve ardından **Rename (yeniden adlandır)** butonuna tıklanır.

Bu adımın tamamlanmasıyla Notebook dosyası yeni adıyla kaydedilmiş olur ve çalışma daha düzenli şekilde sürdürülebilir.

3. Adım: Kod Yazma ve Çalıştırma

Notebook dosyası adlandırıldıktan sonra ilk hücreye print("Merhaba Dünya") Python kodu yazılır (Görsel 1.19).



Görsel 1.19: Kod yazımı ekranı

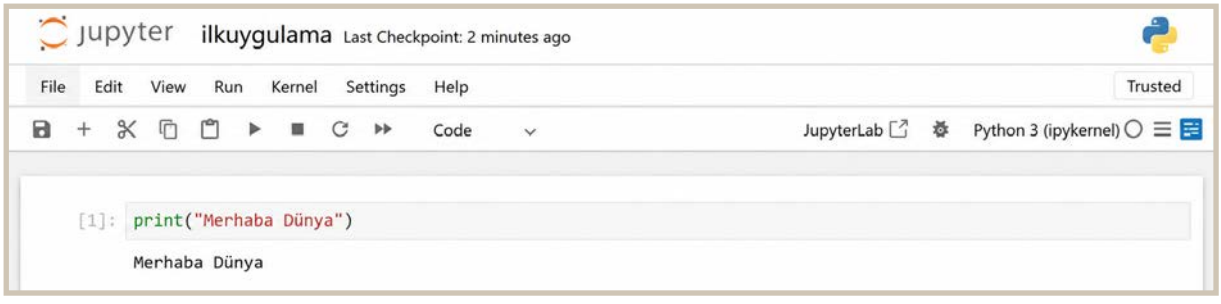
Bu kod, ekrana **Merhaba Dünya** ifadesini yazdıran basit bir Python komutudur. Kod yazıldıktan sonra hücreyi çalıştırmak için şu yöntemlerden biri kullanılabilir:

- Üst menüde yer alan **Run (çalıştır)** butonuna tıklamak
- Klavyeden **Shift + Enter** tuşlarına basmak

Kod çalıştırıldığında gerçekleşen işlemler şunlardır:

- Yazılan komut yürütülür.
- Elde edilen çıktı, hücrenin hemen altında görüntülenir.

Görsel 1.20'de kodun başarıyla çalıştırıldığı ve çıktının doğru şekilde ekrana yazdırıldığı görülmektedir. Bu işlem, Jupyter Notebook ortamında kod yazma ve çalıştırma sürecinin temelini oluşturur.



Görsel 1.20: Kod ekran çıktısı

1. UYGULAMA

Adı	Python Kurulumu ve Merhaba Dünya Kodunu Ekleme
Süre	1 ders saati
Amaç	Python geliştirme ortamını hazırlayarak temel bir uygulamayı çalıştırmak
Yönerge	Aşağıdaki işlem basamaklarını takip ederek Python geliştirme ortamını hazırlayınız ve temel uygulamayı gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğinde yer alan ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Bu uygulamada yalnızca Python ortamını çalıştırmanın yanında çalışma ortamının doğru kurulduğunu kontrol etmek de önemlidir. Kodun çalışmaması durumunda hata mesajı dikkatle incelenmeli; dosya adı, çalışma klasörü ve kullanılan ortam kontrol edilmelidir. Yapay zekâ projelerinde küçük bir kurulum hatası bile sonraki uygulamaların çalışmasını engelleyebilir. Bu nedenle uygulama sonunda "Kod çalıştı mı?", "Çıktı doğru mu?", "Dosya doğru adla kaydedildi mi?" sorularına cevap verilmelidir. Çalışma sürecinde, kurulum ve uygulama adımlarını dikkatli, sıralı ve sistemli şekilde takip ediniz.

İşlem Basamakları

1. Kurulum dosyasını indiriniz.

   Python veya Anaconda kurulum dosyasını kullandığınız işletim sistemine uygun olarak indiriniz. Kurulumu tamamlayınız.

   İndirilen kurulum dosyasını çalıştırınız ve gerekli adımları takip ederek kurulumu tamamlayınız.
2. Geliştirme ortamını açınız.

   Jupyter Notebook, IDLE veya uygun bir Python geliştirme ortamını açınız.
3. Yeni çalışma dosyası oluşturunuz.

   Açılan ortamda yeni bir dosya veya Notebook oluşturunuz.
4. Temel uygulamayı yazınız.

   Aşağıdaki kodu çalışma alanına yazınız.

   print("Merhaba Dünya")
5. Uygulamayı çalıştırınız.

   Yazdığınız kodu çalıştırınız ve çıktının ekranda doğru şekilde görüntülendiğini kontrol ediniz.
6. Dosyayı kaydediniz.

   Oluşturduğunuz çalışma dosyasını uygun bir ad vererek kaydediniz.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

1.2. YAPAY ZEKÂDA VERİ İŞLEME KÜTÜPHANELERİ

Düşün, Tartış, Paylaş

Yapay zekâ uygulamaları, çok büyük veri setleriyle çalışmaktadır. Bu setler milyonlarca sayısal veriden oluşabilir. Böyle durumda, bu veriler üzerinde işlem yapmak önemli bir ihtiyaç hâline gelir.

Bilgisayarların milyonlarca sayıyı saniyeler içinde tek tek toplaması, çarpması veya analiz etmesini sağlayan bazı özel araçlar vardır. Bu araçlar, büyük veri setleri üzerinde hızlı ve doğru işlem yapılmasına olanak sunar.

Verilen metne göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Milyonlarca sayıyı elle işlemeye çalışsaydınız ne gibi zorluklarla karşılaşırdınız?**
2. **Bilgisayarların bu işlemleri çok hızlı yapabilmesi sizce neden önemlidir?**
3. **Bu tür araçlar günlük hayatımızı ve teknolojiyi nasıl kolaylaştırmaktadır?**

Yukarıdaki sorular üzerinde düşünüldüğünde büyük veri setleriyle çalışmanın insan gücüyle zor ve zaman alıcı olduğu anlaşılmaktadır. Bu durum; verilerin hızlı, doğru ve sistemli şekilde işlenmesini sağlayan özel araçlara duyulan ihtiyacı ortaya koyar.

Bu ihtiyaca cevap veren araçlar **veri işleme kütüphaneleri** olarak adlandırılır. Yapay zekâ projelerinde yaygın kullanılan kütüphaneler arasında **NumPy (Numerical Python)** ve **Pandas** yer alır.

NumPy, çok boyutlu dizilerle (array) çalışmayı kolaylaştıran ve sayısal işlemleri yüksek hızda gerçekleştiren bir kütüphanedir. Bu sayede büyük veri setleri düzenli yapıda saklanabilir ve matematiksel işlemler (toplama, çıkarma, çarpma vb.) hızlı şekilde uygulanabilir. Bu özellikleri nedeniyle NumPy, yapay zekâ uygulamalarında veri işleme ve hesaplama süreçlerinin temel araçlarından biridir.

Pandas verilerin düzenlenmesi, temizlenmesi ve analiz edilmesini sağlayan bir kütüphanedir. Özellikle tablo (veri çerçevesi) yapısındaki veriler üzerinde çalışmayı kolaylaştırarak verilerin daha anlaşılır ve yorumlanabilir hâle gelmesini sağlar.

Bu kütüphanelerin sunduğu temel avantajlar şunlardır:

- Büyük veri setleri daha kısa sürede işlenebilir.
- Daha az kodla daha okunabilir ve hızlı işlemler yapılabilir.
- Veriler daha düzenli ve anlamlı hâle getirilir.

5. SIRA SİZDE

Aşağıdaki ifadeleri inceleyiniz ve tabloda uygun başlığın altına yazınız.

Kelimeler

Çok boyutlu diziler	Veri temizleme	Hızlı hesaplama	Tablo yapısı	Eksik veri düzenleme	Sayısal işlemler	Veri analizi	Büyük veri işlemleri	Vaka

NumPy	Pandas
.....	.....
.....	.....
.....	.....
.....	.....

1.2.1. NumPy Kullanarak Dizi Oluşturma

NumPy kullanılarak dizi oluşturulabilmesi için öncelikle kütüphanenin programa dâhil edilmesi gerekir. Bu işlem **import numpy as np** komutuyla gerçekleştirilir.

Bu ifadeyle NumPy kütüphanesi programa **np** takma adıyla eklenmiş olur. Takma ad kullanılması, uzun kütüphane adlarını tekrar tekrar yazma zorunluluğunu ortadan kaldırır ve kod yazımını daha pratik hâle getirir. Bu sayede NumPy fonksiyonları, **np** kısaltması kullanılarak hızlı ve kolay şekilde çağrılabilir.

Tek Boyutlu Dizi Oluşturma

NumPy kullanarak en temel dizi oluşturma yöntemi, bir sayı listesini diziye dönüştürmektir.

Örnek

```
import numpy as np
dizi = np.array([10, 20, 30, 40, 50])
print(dizi)
```

Bu kod çalıştırıldığında, verilen sayılardan bir NumPy dizisi oluşturulur ve ekrana yazdırılır. Bu tür dizilere **tek boyutlu dizi** denir.

İki Boyutlu Dizi Oluşturma

NumPy kullanarak satır ve sütunlardan tablolar da oluşturulabilir. Bu tür diziler **iki boyutlu dizi** olarak adlandırılır.

Örnek

```
import numpy as np
matris = np.array([[1, 2, 3],
                   [4, 5, 6]])
print(matris)
```

Bu örnekte 2 satır ve 3 sütunlu veri yapısı oluşturulmuştur. Bu yapı tablo veya matris şeklindedir.

Çok Boyutlu Dizilerin Önemi

NumPy kullanarak oluşturulan diziler tek boyutlu olmak zorunda değildir. İki ve daha fazla boyuta sahip diziler de oluşturulabilir.

Bu özelliğin sunduğu avantajlar şunlardır:

- Veriler düzenli şekilde saklanır.
- Büyük veri setleri kolayca yönetilir.
- Yapay zekâ uygulamalarında kullanılan karmaşık veri yapıları oluşturulabilir.

1.1. ÖRNEK UYGULAMA

Adı	NumPy Kullanarak Günlük Verileri Dizi Hâline Getirme
Süre	20 dakika
Amaç	Günlük yaşamda elde edilen verileri NumPy kullanarak dizi (array) yapısına dönüştürmek
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook uygulaması, Python programlama dili

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni bir Notebook oluşturunuz ve adını **"NumPy_Dizi"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: NumPy Kütüphanesini Ekleme

```
import numpy as np
```

2. Adım: Tek Boyutlu Dizi Oluşturma

Bir öğrencinin haftalık ders çalışma saatlerini dizi hâline getiriniz.

```
calisma_saatleri = np.array([2, 3, 4, 5, 3, 2, 1])
print(calisma_saatleri)
```

3. Adım: Başlangıç ve Hedef Noktası

Günlük su tüketim miktarları kullanılarak yeni bir dizi oluşturunuz.

```
su_tuketimi = np.array([1.5, 2.0, 1.8, 2.2, 2.0, 1.7, 1.6])
print(su_tuketimi)
```

4. Adım: Çok Boyutlu Dizi Oluşturma

İki öğrencinin verileri karşılaştırılarak tablo biçiminde bir dizi oluşturunuz.

```
veri = np.array([[2, 3, 4, 5, 3, 2, 1],
                 [1, 2, 3, 4, 2, 1, 2]])
print(veri)
```

Çalışmanız tamamlandığında oluşturduğunuz dizileri dikkatlice inceleyip yorumlayınız. Ardından yaptığınız çalışmaları kaydediniz. Son olarak oturumu güvenli şekilde kapatınız ve çalışma ortamını sonraki kullanıcı için temiz ve düzenli bırakınız.

Sonuç

1. **Günlük yaşamda elde edilen verilerin dizi hâline getirilmesi ne gibi kolaylıklar sağlar? Açıklayınız.**

   .....

   .....

   .....

   .....

2. **Tek boyutlu dizilerle çok boyutlu diziler arasındaki farkı oluşturduğunuz örnek üzerinden açıklayınız.**

   .....

   .....

   .....

   .....

1.2.2. NumPy Kullanılarak Temel Matematiksel İşlemler

NumPy kütüphanesi, oluşturulan diziler üzerinde temel matematiksel işlemlerin hızlı ve kolay şekilde yapılmasını sağlar. Bu işlemler, yapay zekâ uygulamalarında veri üzerinde hesaplama yapılması açısından önemlidir. NumPy kullanılarak oluşturulan dizilerde tüm elemanlar üzerinde aynı anda işlem yapılabilir. Bu özellik sayesinde, büyük veri setleri üzerinde işlemler kısa sürede gerçekleştirilir.

Toplama İşlemi

NumPy dizilerinde toplama işlemi dizi elemanlarının her birine aynı değerin eklenmesiyle gerçekleştirilir.

```
import numpy as np
dizi = np.array([10, 20, 30, 40, 50])
sonuc = dizi + 5
print(sonuc)
```

Bu işlem sonucunda dizideki tüm elemanlara 5 eklenmiş olur.

Çıkarma İşlemi

NumPy dizilerinde çıkarma işlemi, tüm elemanlardan belirli bir değerin çıkarılmasıyla yapılır.

```
sonuc = dizi - 10
print(sonuc)
```

Bu işlem sonucunda, dizideki tüm elemanlardan 10 çıkarılır.

Çarpma İşlemi

NumPy dizilerinde çarpma işlemi, tüm elemanların belirli bir sayıyla çarpılması şeklinde gerçekleştirilir.

```
sonuc = dizi * 2
print(sonuc)
```

Bu işlem sonucunda, dizideki tüm elemanlar 2'yle çarpılmış olur.

Bölme İşlemi

NumPy dizilerinde bölme işlemi, dizide yer alan tüm elemanların belirli sayıya bölünmesiyle gerçekleştirilir.

```
print(dizi / 2)
```

Bu işlem sonucunda, dizideki tüm elemanlar 2'ye bölünür ve elde edilen sonuçlar ekrana yazdırılır.

BİLGİ KUTUSU

NumPy kütüphanesinde yapılan bölme işlemleri sonucunda elde edilen değerler genellikle ondalık (float) veri türünde olur. Bu özellik sayesinde matematiksel işlemler daha hassas ve doğru şekilde gerçekleştirilebilir.

NumPy Kullanılarak Yapılan Matematiksel İşlemlerin Önemi

NumPy kullanılarak matematiksel işlem yapmanın avantajları şunlardır:

- Büyük veri setleri üzerinde işlemler hızlı şekilde gerçekleştirilebilir.
- Tüm veriler üzerinde aynı anda işlem yapılabilir.
- Kod yazımı daha kısa ve anlaşılır hâle gelir.
- Hata olasılığı azalır.

6. SIRA SİZDE

Aşağıdaki işlemleri NumPy kullanarak gerçekleştiriniz.

1. Aşağıdaki diziyi oluşturunuz.
   [5, 10, 15, 20, 25]
   .....
2. Dizinin tüm elemanlarına 3 ekleyiniz.
   .....
3. Dizinin tüm elemanlarından 5 çıkarınız.
   .....
4. Dizinin tüm elemanlarını 2'yle çarpınız.
   .....

1.2. ÖRNEK UYGULAMA

Adı	Numpy Kütüphanesiyle Temel Matematik İşlemleri
Süre	20 dakika
Amaç	Basit bir veri seti kullanarak NumPy kütüphanesiyle temel matematiksel işlemleri (toplama, çıkarma, çarpma, bölme) gerçekleştirmek
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook uygulaması, Python programlama dili

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni Notebook oluşturunuz ve adını **"NumPy_Islem"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: NumPy kütüphanesini Ekleme

Aşağıdaki kodu yazarak NumPy kütüphanesi programa dâhil ediniz.

```
import numpy as np
```

2. Adım: Veri Seti Oluşturma

Bir sınıftaki öğrencilerin matematik sınavından aldığı puanları dizi hâline getiriniz.

```
notlar = np.array([50, 60, 70, 80, 90])
print(notlar)
```

3. Adım: Toplama İşlemi

Tüm öğrencilere 5 puan eklenmesi durumunu modelleyiniz.

```
print(notlar + 5)
```

4. Adım: Çıkarma İşlemi

Sınavda yapılan bir hata nedeniyle tüm notlardan 10 puan çıkarılması durumu modelleyiniz.

```
print(notlar - 10)
```

5. Adım: Çarpma İşlemi

Notların 2 ile çarpılarak farklı bir ölçekte değerlendirilmesini sağlayınız.

```
print(notlar * 2)
```

6. Adım: Bölme İşlemi

Notları 2'ye bölerek ortalama hesaplama sürecine hazırlık yapınız.

```
print(notlar / 2)
```

7. Adım: Sonuçları İnceleme

Elde edilen sonuçları karşılaştırınız ve işlemlerin veri üzerindeki etkisini değerlendiriniz.

Çalışmanız tamamlandığında elde ettiğiniz sonuçları inceleyip yorumlayınız. Ardından çalışmalarınızı kaydediniz. Oturumu güvenli şekilde kapatınız, çalışma ortamını sonraki kullanıcı için temiz ve düzenli bırakınız. Bu işlemleri çalışma sürecinize dâhil ediniz.

Sonuç

1. **Bir veri seti üzerinde aynı anda işlem yapılması ne gibi kolaylıklar sağlamaktadır? Açıklayınız.**

   .....
   .....

2. **Toplama, çıkarma, çarpma ve bölme işlemlerinin veri üzerindeki etkilerini karşılaştırınız. Açıklayınız.**

   .....
   .....

1.2.3. NumPy Dizilerinde İndeksleme, Dilimleme ve Temel Özellikler

NumPy kütüphanesiyle oluşturulan dizilerde yer alan verilere erişmek, bu verileri incelemek ve belirli bölümleri kullanmak için bazı temel yöntemler vardır. Bu yöntemler arasında **indeksleme, dilimleme (slicing)** ve **dizi özelliklerinin incelenmesi** önemli yer tutar.

Bu işlemler sayesinde veri setleri daha iyi anlaşılıp veriler üzerinde yapılacak işlemler daha doğru planlanabilir.

NumPy Dizilerinde İndeksleme

İndeksleme, NumPy dizisinde yer alan belirli bir elemana erişmek için kullanılan yöntemdir. NumPy dizilerinde indeks numaraları **0'dan başlar.**

Tek Boyutlu Dizilerde İndeksleme

Örneği inceleyiniz.

Örnek

```
import numpy as np
dizi = np.array([10, 20, 30, 40, 50])
print(dizi[0])
print(dizi[2])
print(dizi[4])
```

Bu kodda dizilerin gösterdiği elemanlar şunlardır:

- dizi[0] → ilk elemanı
- dizi[2] → üçüncü elemanı
- dizi[4] → son elemanı

Negatif İndeksleme

NumPy dizilerinde elemanlara sondan başlayarak da erişilebilir.

Örnek

```
print(dizi[-1])
print(dizi[-2])
```

Bu kullanımda dizilerin gösterdiği elemanlar şunlardır:

-1 → son eleman

-2 → sondan bir önceki eleman

İki Boyutlu Dizilerde İndeksleme

İki boyutlu dizilerde indeksleme yapılırken satır ve sütun birlikte kullanılır.

Örnek

```
matris = np.array([[1, 2, 3],
                   [4, 5, 6]])
print(matris[0, 0])
print(matris[1, 2])
```

Bu kodda dizilerin gösterdiği değerler şunlardır:

- matris[0, 0] → 1. satır 1. sütun
- matris[1, 2] → 2. satır 3. sütun

NumPy Dizilerinde Dilimleme (Slicing)

Dilimleme, dizinin belirli bir bölümünü seçmek için kullanılan yöntemdir. Bu sayede birden fazla elemana aynı anda erişim sağlanabilir.

Kullanım Şekli

```
[başlangıç : bitiş]
```

Tek Boyutlu Dizilerde Dilimleme

Örnek

```
print(dizi[1:4])
print(dizi[:3])
print(dizi[2:])
```

Bu kodda dizilerin gösterdiği elemanlar şunlardır:

- dizi[1:4] → 20, 30, 40
- dizi[:3] → ilk üç eleman
- dizi[2:] → üçüncü elemandan sonrası

```
print(dizi[0:5:2])
```

Bu kullanımda şu işlemler yapılır:

- 0'dan başlanır.
- 2'şer artarak elemanlar seçilir.

İki Boyutlu Dizilerde Dilimleme

Örnek

```
print(matris[0:2, 1:3])
```

Bu kodun seçtiği değerler şunlardır:

- İlk iki satır
- 2.ve 3. sütun

7. SIRA SİZDE

Bir öğretmen, öğrencilerinin bir hafta boyunca çözdükleri soru sayılarını NumPy kullanarak aşağıdaki gibi bir dizi hâline getirmiştir.

```
import numpy as np
sorular = np.array([20, 25, 30, 15, 10, 5, 0])
```

Öğretmenin yaptığı işlemler şunlardır:

- Pazartesi günü çözülen soru sayısını öğrenmek.
- Hafta içi (ilk beş gün) çözülen soru sayılarını birlikte incelemek.
- Son gün çözülen soru sayısını ayrıca kontrol etmek.

Öğretmen bu işlemleri yaparken bazen **tek elemana erişmiş,** bazen de **birden fazla elemanı birlikte seçmiştir.**

Yukarıdaki senaryoya göre yapılan işlemleri inceleyiniz. Aşağıdaki ifadeleri uygun sütuna yerleştiriniz.

İfadeler: tek gün verisine ulaşma, birden fazla günü birlikte inceleme, sorular[0], sorular[0:5], sorular[-1], aralık kullanma, tek elemana erişim, veri grubu seçme

İNDEKSLEME	DİLİMLEME

NumPy Dizilerinde Temel Özellikler

NumPy dizilerinin yapısını incelemek için bazı özellikler kullanılır. Bu özellikler, dizinin boyutu ve veri yapısı hakkında bilgi verir.

Dizi Özellikleri

- **shape (şekil):** Dizinin satır ve sütun sayısını gösterir.
- **ndim (boyut sayısı):** Dizinin kaç boyutlu olduğunu belirtir.
- **size (eleman sayısı):** Toplam eleman sayısını verir.
- **dtype (veri türü):** Elemanların veri türünü gösterir.
- **itemsize (eleman boyutu):** Her bir elemanın bellekte kapladığı alanı gösterir.

Örnek Kullanım

```
print(dizi.shape)
print(dizi.ndim)
print(dizi.size)
print(dizi.dtype)
print(dizi.itemsize)
```

1.3. ÖRNEK UYGULAMA

Adı	NumPy Kullanarak Günlük Telefon Kullanım Verilerinin Analizi
Süre	20 dakika
Amaç	Günlük yaşamda sık karşılaşılan telefon kullanım verilerini NumPy dizisi hâline getirerek, indeksleme ve dilimleme yöntemleriyle veri seçme ve incelemek
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook uygulaması, Python programlama dili

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni Notebook oluşturunuz ve adını **"NumPy_Telefon"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: NumPy Kütüphanesini Ekleme

```
import numpy as np
```

2. Adım: Veri Seti Oluşturma

Bir öğrencinin bir hafta boyunca günlük telefon kullanım sürelerini (saat cinsinden) dizi hâline getiriniz.

```
telefon = np.array([2, 3, 5, 4, 6, 7, 3])
print(telefon)
```

3. Adım: İndekslemeyle Veri Seçme

Belirli günlere ait kullanım sürelerini inceleyiniz.

```
print(telefon[0])  # Pazartesi
print(telefon[3])  # Perşembe
print(telefon[-1]) # Pazar
```

4. Adım: Dilimlemeyle Veri İnceleme

Belirli gün aralıklarını birlikte değerlendiriniz.

```
print(telefon[0:5])  # Hafta içi kullanım
print(telefon[5:])   # Hafta sonu kullanım
print(telefon[::2])  # Gün atlayarak inceleme
```

5. Adım: İki Boyutlu Veri Oluşturma

İki öğrencinin telefon kullanım verilerini karşılaştırınız.

```
veri = np.array([[2, 3, 5, 4, 6, 7, 3],
                 [1, 2, 4, 3, 5, 6, 2]])
print(veri)
```

6. Adım: Çok Boyutlu Veri Seçme

```
print(veri[0, 2])    # 1. öğrenci, Çarşamba
print(veri[1, 4])    # 2. öğrenci, Cuma
print(veri[:, 0:3])  # Her iki öğrencinin ilk 3 günü
```

7. Adım: Dizi Özelliklerini İnceleme

```
print(veri.shape)
print(veri.ndim)
print(veri.size)
print(veri.dtype)
print(veri.itemsize)
```

Çalışmanız tamamlandığında elde ettiğiniz verileri inceleyip yorumlayınız. Ardından çalışmalarınızı kaydediniz. Oturumu güvenli şekilde kapatınız ve çalışma ortamını temiz, düzenli ve sonraki kullanıcıya uygun şekilde bırakınız.

Sonuç

1. **Telefon kullanım verilerini analiz etmek ne gibi bilgiler sağlar? Açıklayınız.**

   .....

   .....

2. **İndekslemeyle tek bir gün incelemesiyle dilimlemeyle birden fazla gün incelenmesi arasındaki fark nedir? Açıklayınız.**

   .....

   .....

3. **Telefon kullanım süresinin fazla olması sizce ne gibi sonuçlara yol açabilir? Açıklayınız.**

   .....

   .....

NOTLAR

2. UYGULAMA

Adı	Numpy Kullanarak Matematiksel İşlemler Yapma
Süre	1 ders saati
Amaç	NumPy dizileri üzerinde temel matematiksel işlemler uygulamak.
Yönerge	Aşağıdaki işlem basamaklarını takip ederek NumPy dizileri üzerinde matematiksel işlemler uygulayınız.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğinde yer alan ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Günlük yaşamda karşılaşılan sayısal veriler üzerinde farklı matematiksel işlemler yapılabilmektedir. Sayısal veriler üzerinde yapılan her işlem, verinin anlamını değiştirebilir. Örneğin bir listedeki değerlere ekleme yapmak artışı, çıkarma yapmak azalmayı, çarpma veya bölme işlemiyse verinin farklı durumlarda nasıl değiştiğini gösterebilir. Bu uygulamada amaç yalnızca işlem sonucunu bulmak değil, yapılan işlemin veri üzerinde nasıl bir etki oluşturduğunu yorumlamaktır.

İşlem Basamakları

1. Veri setini oluşturunuz.
   Bir öğrencinin bir hafta boyunca yaptığı günlük harcamaları NumPy dizisi olarak tanımlayınız.
   Örnek veri: **[20, 35, 40, 25, 50, 60, 30]**
2. NumPy Kütüphanesini ekleyiniz.
   Aşağıdaki komutu kullanınız.
   import numpy as np
3. Harcama verilerini NumPy dizisi hâline getiriniz.
4. Tüm günlerde 10 TL ek harçlık verildiğini varsayarak yeni durumu hesaplayınız.
5. Tasarruf amacıyla her gün 5 TL daha az harcandığı durumu hesaplayınız.
6. Harcamaların iki katına çıktığı durumu hesaplayınız.
7. Harcamaların yarıya düştüğü durumu hesaplayınız.
8. Elde ettiğiniz sonuçları inceleyerek matematiksel işlemlerin veri üzerindeki etkilerini yorumlayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

1.2.4. Pandas Kütüphanesi

NumPy kütüphanesiyle büyük veri setleri üzerinde hızlı matematiksel işlemler yapılabildiği görülmüştür. Ancak gerçek hayattaki veriler çoğu zaman yalnızca sayılardan oluşmaz. Öğrenci listeleri (isim + not), alışveriş kayıtları (ürün + fiyat), telefon kullanım verileri (gün + süre) gibi farklı veri türlerini birlikte içeren birçok veri türü mevcuttur. Bu veriler genellikle tablo yapısındadır.

Pandas kütüphanesi, bu tür karmaşık veri yapılarını yönetmek için geliştirilmiş güçlü bir Python kütüphanesidir. Tablo hâlindeki verileri işleme, düzenleme ve analiz etme süreçlerini kolaylaştırır.

Pandas sayesinde veriler satır ve sütunlardan oluşan düzenli yapılar hâline getirilir ve bu veriler üzerinde hızlı, etkili işlemler yapılabilir. Bu durum, özellikle büyük ve karmaşık veri setleriyle çalışılırken avantaj sağlar.

Temel Veri Yapıları

Pandas kütüphanesinde **Series** ve **DataFrame** adında yaygın kullanılan iki temel veri yapısı bulunur.

Series, tek boyutlu veri yapısıdır. Bir sütun gibi düşünülebilir. Sayısal verileri veya metni saklayabilir.

```
import pandas as pd
veri = pd.Series([10, 20, 30, 40])
print(veri)
```

DataFrame satır ve sütunlardan oluşan tablo veri yapısıdır. Gerçek hayattaki veri tablolarını temsil eder.

```
import pandas as pd
veri = {
  "Ad": ["Ali", "Ayşe", "Mehmet"],
  "Not": [85, 90, 78]
}
df = pd.DataFrame(veri)
print(df)
```

Bu yapı sayesinde farklı veri türleri aynı tabloda birlikte tutulup analiz edilebilir.

Veri Okuma

Pandas kütüphanesiyle veriler dış dosyalardan programa aktarılabilir. En yaygın kullanılan dosya türleri CSV ve Excel dosyalarıdır.

```
df = pd.read_csv("veri.csv")
df = pd.read_excel("veri.xlsx")
```

Bu sayede gerçek hayattaki hazır veri setleri doğrudan analiz sürecine dâhil edilebilir.

Veri İnceleme

Pandas, veri setlerini hızlı şekilde incelemek için çeşitli fonksiyonlar ve özellikler sunar. Bu fonksiyonlar ve özellikler şunlardır:

- head()"fonksiyonu, varsayılan olarak veri setinin ilk 5 satırını gösterir.
- tail() fonksiyonu, veri setinin son birkaç satırını gösterir.
- df.info(): Veri setinde kaç boş (null) değer olduğunu ve sütunların veri türlerini özetler.
- describe() fonksiyonu, sayısal verilerin istatistiksel özetini sunar.

- columns özelliği, sütun isimlerini listeler.
- shape özelliği, veri setinin boyutunu gösterir.

```
print(df.head())
print(df.info())
print(df.describe())
```

Bu işlemler, veri analizi sürecinin ilk adımını oluşturur ve veri yapısının anlaşılmasını sağlar.

Veri Filtreleme

Pandas kütüphanesiyle belirli koşullara uyan veriler seçilebilir.

```
df[df["Not"] > 70]
```

Bu işlemle belirli ölçüte uygun veriler veri seti içerisinden ayrılır. Böylece büyük veri setleri daha anlamlı şekilde incelenebilir.

Veri Sıralama

Veriler belirli bir sütuna göre sıralanabilir.

```
df.sort_values(by="Not", ascending=False)
```

Bu işlem sayesinde veriler büyükten küçüğe veya küçükten büyüğe sıralandığı için karşılaştırma yapılması kolaylaşır.

Veri Manipülasyonu

Pandas kütüphanesi, veri üzerinde değişiklik yapmayı da mümkün kılar. Yeni sütunlar eklenebilir, mevcut veriler güncellenebilir.

```
df["Durum"] = df["Not"] > 70
df["Not"] = df["Not"] + 5
```

8. SIRA SİZDE

Aşağıda Pandas kütüphanesinde sık kullanılan bazı fonksiyonlar ve açıklamalar verilmiştir.

Tabloyu inceleyiniz, fonksiyonları ve özellikleri uygun açıklamalarla eşleştiriniz.

Fonksiyonlar / Özellikler	Açıklamalar
1. head( )	A) Veri setinin ilk 5 satırını tablo hâlinde getirir.
2. tail( )	B) Veri setinin son 5 satırını tablo hâlinde getirir.
3. info( )	C) Veri seti hakkında genel bilgi verir (veri türleri, sütunlar vb.)
4. describe( )	Ç) Sayısal veriler için istatistiksel özet sunar.
5. columns( )	D) Sütun isimlerini listeler.
6. shape( )	E) Veri setinin satır ve sütun sayısını verir.
	F) Eksik (boş) verilerin sayısını gösterir.

Pandas kütüphanesiyle veriler düzenlenebilir, analiz edilebilir ve anlamlı hâle getirilebilir.

Bunun sunduğu avantajlar şunlardır:

- Büyük veri setleri daha kolay yönetilir.
- Veriler üzerinde hızlı analiz yapılır.
- Hata olasılığı azalır.

Yapay zekâ projelerinde veriler genellikle doğrudan kullanılabilir durumda değildir. Verilerin düzenlenmesi, temizlenmesi ve analiz edilmesi gerekir. Pandas kütüphanesi bu sürecin en önemli araçlarından biridir. Bu nedenle Pandas, yapay zekâ ve veri bilimi uygulamalarında veri hazırlama sürecinin temel bileşenlerinden biri olarak kullanılmaktadır.

1.4. ÖRNEK UYGULAMA

Adı	Pandas'la Öğrenci Başarı Verilerini İnceleme
Süre	20 dakika
Amaç	Tablo hâlindeki verileri Pandas kullanarak DataFrame yapısında oluşturmak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook uygulaması, Python ve Pandas kütüphanesi

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni Notebook oluşturunuz ve adını **"Pandas_Basari"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Pandas Kütüphanesini Ekleme

Aşağıdaki kod yazılarak Pandas kütüphanesi programa dâhil ediniz.

```
import pandas as pd
```

2. Adım: DataFrame Oluşturma

Bir sınıftaki öğrencilerin ad, ders ve not bilgileri tablo hâline getiriniz.

```
veri = {
  "Ad": ["Ali", "Ayşe", "Mehmet", "Zeynep", "Ece"],
  "Ders": ["Matematik", "Matematik", "Matematik", "Matematik", "Matematik"],
  "Not": [75, 92, 68, 88, 55]
}
df = pd.DataFrame(veri)

print(df)
```

3. Adım: CSV veya Excel Dosyasından Veri Okuma

Hazır bir veri dosyasını programa aktarınız.

```
df_csv = pd.read_csv("ogrenci_basari.csv")
print(df_csv.head())
df_excel = pd.read_excel("ogrenci_basari.xlsx")
print(df_excel.head())
```

4. Adım: Veri Setini İnceleme

Veri setinin ilk satırlarını, sütunlarını ve genel yapısını inceleyiniz.

```
print(df.head())
print(df.info())
print(df.shape)
print(df.columns)
```

5. Adım: Veri Filtreleme

Notu 70'in üzerinde olan öğrencileri seçiniz.

```
print(df[df["Not"] > 70])
```

6. Adım: Veri Sıralama

Öğrencilerin notlarına göre büyükten küçüğe sıralayınız.

```
print(df.sort_values(by="Not", ascending=False))
```

7. Adım: Temel Veri Manipülasyonları

Veri setine yeni bir sütun ekleyiniz ve öğrencilerin başarı durumu gösteriniz.

```
df["Durum"] = df["Not"] >= 70
print(df)
```

Tüm notlara 5 puan ekleyerek veriyi güncelleyiniz.

```
df["Not"] = df["Not"] + 5
print(df)
```

Yalnızca belirli sütunları seçerek yeni bir görünüm elde ediniz.

```
print(df[["Ad", "Not", "Durum"]])
```

8. Adım: Sayısal Verilerin Özetini İnceleme

Not sütununa ait temel istatistiksel bilgileri görüntüleyiniz.

```
print(df.describe())
```

Çalışmanız tamamlandığında elde ettiğiniz verileri inceleyip yorumlayınız. Ardından çalışmalarınızı kaydediniz. Oturumu güvenli şekilde kapatınız, çalışma ortamını temiz ve düzenli bırakınız.

Sonuç

1. **Pandas kullanarak tablo hâlindeki verileri incelemek ne gibi kolaylıklar sağlar? Açıklayınız.**

2. **Veri filtreleme ve sıralama işlemleri veri analizi sürecinde neden önemlidir? Açıklayınız.**

3. **Bir veri setine yeni sütun eklemek veya mevcut verileri güncellemek ne gibi avantajlar sağlar? Açıklayınız.**

4. **CSV veya Excel dosyasından veri okumanın elle veri girmeye göre üstün yönleri nelerdir? Açıklayınız.**

3. UYGULAMA

Adı	Akıllı Kafe Sipariş Analizi
Süre	1 ders saati
Amaç	Pandas DataFrame yapısını kullanarak veriler üzerinde görüntüleme, filtreleme ve sıralama işlemleri gerçekleştirmek.
Yönerge	Aşağıdaki işlem basamaklarını takip ederek kafe sipariş verileri üzerinde Pandas uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğinde yer alan ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Günlük yaşamda kafe, restoran ve kantinler gün boyunca çok sayıda sipariş almaktadır. Siparişlerden elde edilen veriler; hangi ürünlerin daha çok tercih edildiğini, hangi ürünlerin daha fazla gelir sağladığını ve belirli bir fiyatın üzerindeki ürünlerin satış durumunu anlamak için önemlidir.

Bir işletme için veriler yalnızca sayı listesi değildir; doğru yorumlandığında karar verme sürecini destekleyen önemli bir kaynağa dönüşür. Örneğin en çok satılan ürün ile en yüksek gelir sağlayan ürün her zaman aynı olmayabilir. Bundan dolayı verileri yalnızca sıralamak değil, sonuçları işletme açısından anlamlandırmak da gerekir.

Uygulamada sizden, bir kafeye ait ürün ve sipariş verilerini Pandas DataFrame yapısına dönüştürmeniz; veriler üzerinde okuma, filtreleme ve sıralama işlemleri yaparak elde edilen sonuçları yorumlamanız beklenmektedir. Böylece günlük yaşamda karşılaşılan tablo hâlindeki verilerin bilgisayar ortamında nasıl düzenli ve pratik biçimde işlenebileceğini deneyimleyeceksiniz.

İşlem Basamakları

1. Aşağıdaki kafe sipariş verilerini inceleyiniz ve verilerin içerdiği bilgileri açıklayınız.

Ürün Adı	Kategori	Fiyat (TL)	Satış Adedi
Türk Kahvesi	İçecek	45	32
Latte	İçecek	65	28
Çay	İçecek	20	50
Limonata	İçecek	55	24
Tost	Yiyecek	70	18
Sandviç	Yiyecek	85	15
Kurabiye	Tatlı	40	22
Cheesecake	Tatlı	95	12

2. Verileri Python ortamında uygun veri yapısı kullanarak hazırlayınız.
3. Pandas kütüphanesini ekleyiniz.

   import pandas as pd
4. Hazırladığınız verileri Pandas kullanarak DataFrame hâline getiriniz.
5. Oluşturduğunuz DataFrame'i görüntüleyerek verileri inceleyiniz.
6. Fiyatı 50 TL ve üzeri olan ürünleri filtreleyiniz.
7. Yalnızca içecek kategorisindeki ürünleri listeleyiniz.
8. Ürünleri adet sayılarına göre yüksekten düşüğe sıralayınız.
9. Ürünleri fiyatlarına göre artan sırada sıralayınız ve en pahalı ürünü belirleyiniz.
10. Elde ettiğiniz sonuçları inceleyerek aşağıdaki soruları cevaplayınız.
    - En çok satılan ürün hangisidir?
    - İçecek kategorisinde hangi ürünler bulunmaktadır?
    - 50 TL ve üzeri fiyatlı kaç ürün bulunmaktadır?
    - En pahalı ürün hangisidir?
11. Filtreleme ve sıralama işlemlerinin veri analizi sürecine sağladığı katkıları değerlendiriniz.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

1.2.5. Veri Temizleme ve Düzenleme

Gerçek dünyada kullanılan veriler çoğu zaman düzenli, eksiksiz ve hatasız değildir. Daha önce Pandas kütüphanesiyle verilerin tablo hâlinde düzenlenebildiği ve analiz edilebildiği görülmüştür. Ancak bu veriler doğrudan kullanılabilecek durumda olmayabilir.

Bir veri seti incelendiğinde şu durumlarla karşılaşılabilir:

- Bazı bilgilerin eksik olması
- Yanlış veya hatalı veri girişlerinin bulunması
- Verilerin düzensiz veya karışık olması

Hatalı veri girişlerine bazı örnekler şunlardır:

- Bir öğrenci listesinde bazı öğrencilerin notları girilmemiş olabilir
- Bir alışveriş tablosunda bir ürünün fiyatı yanlış yazılmış olabilir
- Bir veri setinde bazı satırlarda eksik değerler olabilir

Bu tür hatalar içeren veriler kirli veri olarak adlandırılır.

Kirli veri kullanılarak yapılan analizler yanlış sonuçlar verebilir. Bu nedenle veri analizi yapılmadan önce verilerin düzenlenmesi ve temizlenmesi gerekir.

Pandas Kullanarak Veri Temizleme

Veri setleri Pandas ile programa yüklendikten sonra analiz öncesinde temizleme işlemleri yapılır. Bu aşamada eksik değerler belirlenir, gerekli durumlarda silinir veya uygun değerlerle tamamlanır. Böylece veri seti daha düzenli ve analiz edilebilir hâle getirilir.

- Veri setinde eksik değerler olup olmadığı kontrol edilir.

```
print(df.isnull())
print(df.isnull().sum())
```

Bu komutlar, veri setinde eksik değer bulunup bulunmadığını ve her sütunda kaç eksik veri olduğunu gösterir.

```
df = df.dropna()
```

Bu yöntem veri setini temizler ancak bazı satırların tamamen silinmesine neden olabilir. Bu nedenle eksik verilerin silinmesi, veri kaybı açısından dikkatli değerlendirilmelidir.

- Eksik veriler tamamen silinmek yerine uygun bir değerle doldurulabilir.

```
df["Not"] = df["Not"].fillna(df["Not"].mean())
```

Bu işlemde "Not" sütunundaki eksik değerler, aynı sütunun ortalamasıyla tamamlanır. Böylece veri kaybı azaltılır ve analiz süreci daha sağlıklı yürütülür.

- Veri setine yeni bilgiler eklenebilir.

```
df["Durum"] = df["Not"] >= 70
```

Bu işlemle öğrencilerin notlarına göre başarılı olup olmadığı belirlenebilir. Oluşturulan yeni sütun, veri setini yorumlamayı kolaylaştırır.

Veri setinin genel yapısını incelemek için aşağıdaki komutlar kullanılabilir:

```
print(df.info())
print(df.describe())
```

info() komutu sütunların veri türlerini ve eksik değer durumunu gösterir. describe() komutu ise sayısal sütunlara ait ortalama, en küçük değer, en büyük değer ve benzeri temel istatistiksel bilgileri özetler.

Veri temizleme işlemleri tamamlandıktan sonra sayısal sütunlar üzerinde ayrıntılı hesaplamalar yapılabilir. Örneğin "Not" sütunundaki değerlerin ortalamasını hesaplamak için NumPy fonksiyonlarından yararlanılabilir. Bu kullanım, temizlenen veriler üzerinde hızlı ve düzenli sayısal analiz yapılmasına katkı sağlar.

```
import numpy as np
ortalama = np.mean(df["Not"])
print(ortalama)
```

Bu kodda df["Not"] ifadesi DataFrame içindeki "Not" sütununu seçer. np.mean() fonksiyonu ise bu sütundaki değerlerin ortalamasını hesaplar. Böylece tüm değerleri tek tek toplamak ve bölmek yerine ortalama değer kısa yoldan bulunur.

BİLGİ KUTUSU

Veri seti incelenirken yalnızca ortalamaya bakmak her zaman yeterli olmaz. Örneğin iki sınıfın not ortalaması aynı olabilir ancak bir sınıfta notlar birbirine yakınken diğer sınıfta öğrencilerin bazıları çok yüksek, bazıları çok düşük not almış olabilir. Bu nedenle ortalamayla birlikte en küçük değer, en büyük değer ve standart sapma gibi bilgiler de incelenmelidir. Standart sapma, verilerin ortalamadan ne kadar uzaklaştığını gösterir. Bu değer yüksekse veriler arasında büyük farklılıklar olabilir. Veri analizi yalnızca hesaplama yapmakla kalmaz, sonuçları doğru yorumlamayı da gerektirir.

9. SIRA SİZDE

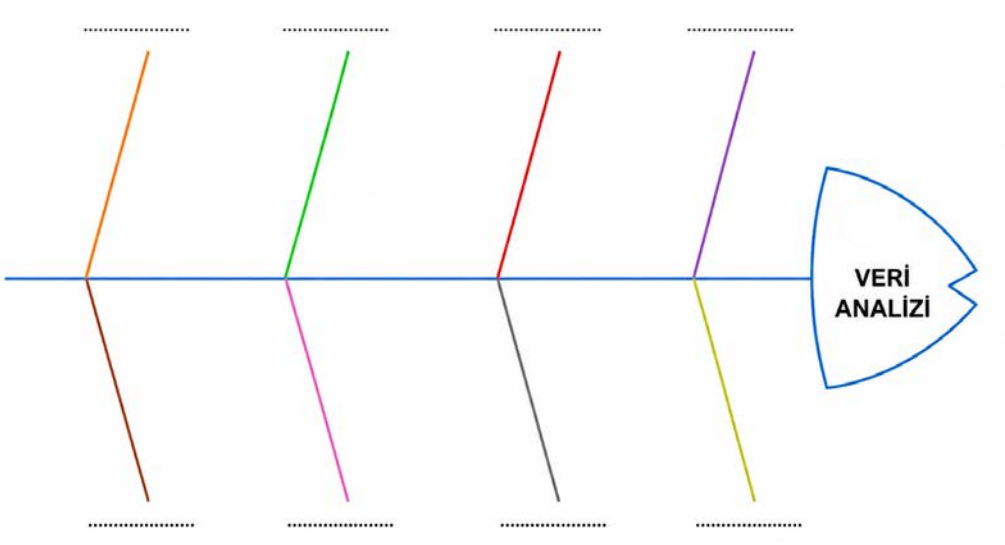
Aşağıda veri temizleme ve sayısal analiz sürecinde kullanılan bazı kavramlar verilmiştir.

Bu kavramları inceleyiniz ve **balık kılçığı diyagramında uygun yerlere yerleştiriniz.**

Kavramlar: Eksik veri, Hatalı veri, df.isnull(), df.dropna(), fillna(), np.mean(), np.max(), np.min(), NumPy

Yönerge

- **Üst kısma** veri temizlemeyle ilgili kavramları yazınız.
- **Alt kısma** sayısal işlemlerle ilgili kavramları yazınız.



NOTLAR

1.5. ÖRNEK UYGULAMA

Adı	Pandas' la Çevrim İçi Market Sipariş Verilerinin Temizlenmesi ve Analizi
Süre	1 ders saati
Amaç	CSV dosyasından veri okumak, eksik değerleri tespit etmek ve uygun yöntemlerle doldurmak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook uygulaması, Python programlama dili, Pandas ve NumPy kütüphaneleri

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni Notebook oluşturunuz ve adını **"Market_Veri_Temizleme"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.
- Veri seti üzerinde yapılan değişikliklerin nedenlerini sorgulayınız ve elde edilen sonuçları değerlendiriniz.

Uygulama Adımları

1. Adım: Pandas ve NumPy Kütüphanelerini Ekleme

Aşağıdaki kod yazılarak Pandas ve NumPy kütüphanelerini programa dâhil ediniz.

```
import pandas as pd
import numpy as np
```

2. Adım: CSV Dosyasından Veri Okuma

Çevrim içi market siparişlerine ait hazır veri dosyasını programa aktarınız.

```
df = pd.read_csv("market_siparisleri.csv")
print(df.head())
```

3. Adım: Veri Setini İnceleme

Veri setinin ilk satırlarını, genel yapısını, sütunlarını ve boyutunu inceleyiniz.

```
print(df.head())
print(df.info())
print(df.shape)
print(df.columns)
```

Bu komutlarla veri setinin yapısı hakkında genel bilgi elde ediniz.

4. Adım: Adım: Eksik Verileri Tespit Etme

Veri setinde eksik değerler olup olmadığını kontrol ediniz.

```
print(df.isnull())
print(df.isnull().sum())
```

Bu komutlarla veri setindeki eksik değerleri ve her sütunda bulunan eksik veri sayısını görüntüleyiniz.

5. Adım: Eksik Verileri Doldurma

Eksik değerleri uygun yöntemler kullanılarak doldurunuz.

```
df["Fiyat"] = df["Fiyat"].fillna(df["Fiyat"].mean())
df["Teslimat_Suresi"] = df["Teslimat_Suresi"].fillna(df["Teslimat_Suresi"].mean())
print(df)
```

6. Adım: Gerekirse Eksik Verileri Silme

Ürün adı veya kategori gibi önemli bilgilerin eksik olduğu satırları veri setinden çıkarınız.

```
df = df.dropna()
print(df)
```

7. Adım: Yeni Sütun Oluşturma

Sipariş tutarını göstermek için yeni bir sütun oluşturunuz.

```
df["Toplam_Tutar"] = df["Adet"] * df["Fiyat"]
print(df)
```

8. Adım: Verileri Sınıflandırma ve İnceleme

Siparişleri kategori bilgisine göre inceleyiniz.

```
print(df["Kategori"].value_counts())
```

9. Adım: Basit İstatistiksel Analiz Yapma

Sayısal sütunlara ait temel istatistiksel bilgileri görüntüleyiniz.

```
print(df.describe())
print("Ortalama fiyat:", np.mean(df["Fiyat"]))
print("En yüksek fiyat:", np.max(df["Fiyat"]))
print("En düşük fiyat:", np.min(df["Fiyat"]))
print( "[ Toplam sipariş tutarı:", np.sum(df["Toplam_Tutar"]))
```

10. Adım: Sonuçları Yorumlama

Elde ettiğiniz çıktıları inceleyiniz ve veri setinin temizlenmeden önceki durumu ile temizlendikten sonraki durumunu karşılaştırınız.

Çalışmanız tamamlandığında elde ettiğiniz verileri inceleyip yorumlayınız. Ardından çalışmalarınızı kaydediniz. Oturumu güvenli şekilde kapatınız ve çalışma ortamını temiz, düzenli ve sonraki kullanıcıya uygun şekilde bırakınız.

Sonuç

1. **CSV dosyasından veri okuma işlemi, hazır verilerle çalışmayı nasıl kolaylaştırır? Açıklayınız.**

.....
.....
.....
.....

2. **Eksik verileri tespit etmek veri analizi sürecinde neden önemlidir? Açıklayınız.**

.....
.....
.....
.....

3. **Eksik verileri ortalamayla doldurmak hangi durumlarda avantaj sağlayabilir? Açıklayınız.**

.....
.....
.....
.....

4. **Yeni sütun oluşturarak toplam sipariş tutarını hesaplamak ne tür kolaylıklar sağlar? Açıklayınız.**

.....
.....
.....
.....

5. **Veri temizleme yapılmadan gerçekleştirilen analizler neden yanıltıcı olabilir? Açıklayınız.**

.....
.....
.....
.....

4. UYGULAMA

Adı	Bireysel Sağlık Verilerinin Temizlenmesi ve Analizi
Süre	2 ders saati
Amaç	CSV dosyasındaki sağlık verilerini inceleyerek eksik verileri temizlemek ve temel veri analizi işlemlerini gerçekleştirmek.
Yönerge	Aşağıdaki işlem basamaklarını takip ederek sağlık verileri üzerinde veri temizleme ve analiz uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğinde yer alan ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Hastaneler, aile hekimlikleri ve mobil sağlık uygulamaları bireylerin sağlık verilerini düzenli olarak kayıt altına alır.

Sağlık verileriyle çalışılırken her eksik değer aynı şekilde ele alınmaz. Bazı eksik veriler ortalamayla doldurulabilirken bazı hatalı değerler analiz sonucunu yanıltabileceği için ayrıca incelenmelidir. Örneğin gerçekçi olmayan bir nabız değeri ölçüm hatasından veya veri girişindeki yanlışlıktan kaynaklanmıştır. Bu

uygulamada veri temizleme işlemi yapılırken "Bu değeri silmek mi, düzeltmek mi, yoksa doldurmak mı daha doğru olur?" sorusu üzerinden karar verilmelidir. Sağlık verilerinin kişisel ve hassas veri olduğu unutulmamalı, uygulamada kurgusal ya da anonimleştirilmiş veri kullanılmalıdır.

Bu uygulamada sizden, bir bireye ait sağlık verilerini CSV dosyasından okumanız, veri setindeki eksik değerleri tespit edip temizlemeniz, ardından veriler üzerinde inceleme yaparak sonuçları yorumlamanız beklenmektedir. Böylece gerçek yaşamda kullanılan sağlık verilerinin bilgisayar ortamında nasıl düzenlenip analiz edildiğini deneyimleyeceksiniz.

Gün	Nabız	Tansiyon	Kilo
Pazartesi	80	120	70
Salı		125	71
Çarşamba	95		70
Perşembe	100	135	
Cuma	85	128	72

İşlem Basamakları

1. Aşağıdaki sağlık verilerini inceleyiniz.
2. Pandas kütüphanesini ekleyiniz.
   import pandas as pd
3. Sağlık verilerini CSV dosyasından okuyunuz.
4. Veri setini görüntüleyerek inceleyiniz.
5. Veri setindeki eksik değerleri belirleyiniz.
6. Eksik değerleri sütun ortalamalarını kullanarak tamamlayınız.
7. Temizlenmiş veri setini görüntüleyiniz.
8. Ortalama nabız değerini ve en yüksek tansiyon değerini belirleyiniz.
9. Elde ettiğiniz sonuçları inceleyerek aşağıdaki soruları cevaplayınız.
   - Veri setinde kaç adet eksik veri bulunmaktadır?
   - Eksik veriler hangi yöntem kullanılarak tamamlanmıştır?
   - Ortalama nabız değeri nedir?
   - En yüksek tansiyon değeri nedir?
10. Veri temizleme işlemlerinin veri analizi sonuçlarına etkisini değerlendiriniz.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

1.3. MAKİNE ÖĞRENMESİ KÜTÜPHANELERİ

Düşün, Tartış, Paylaş

Günümüzde yapay zekâ uygulamaları yalnızca verileri saklamak veya düzenlemekle sınırlı değildir. Birçok yapay zekâ sistemi, geçmiş verileri inceleyerek yeni durumlar hakkında tahminler yapabilmektedir. Yapay zekâ uygulamalarının yapabileceği işlemlere bazı örnekler şunlardır:

- Bir e-posta uygulaması spam mesajları ayırt edebilir.
- Bir alışveriş sitesi kullanıcılara ürün önerilerinde bulunabilir.
- Bir navigasyon uygulaması trafik yoğunluğunu tahmin edebilir.
- Bir sağlık uygulaması bazı risk durumlarını analiz edebilir.

Bu tür işlemlerin yapılabilmesi için yalnızca verileri düzenlemek yeterli değildir. Verilerden öğrenebilen, model oluşturabilen ve tahmin yapabilen özel yazılım araçlarına ihtiyaç duyulur. Bu araçlar makine öğrenmesi kütüphaneleri olarak adlandırılır.

Verilen metne göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

- **Bilgisayarlar geçmiş verilerden nasıl öğrenebilir?**
- **Bir sistemin tahmin yapabilmesi için hangi bilgilere ihtiyaç duyulabilir?**
- **İnsanların öğrenmesiyle makinelerin öğrenmesi arasında ne gibi benzerlikler olabilir?**
- **Günlük yaşamda makine öğrenmesi kullanılan hangi uygulamalarla karşılaşıyorsunuz?**

Önceki bölümde, büyük veri setlerinin işlenmesini sağlayan NumPy ve Pandas gibi veri işleme kütüphaneleri incelenmiştir. Bu kütüphaneler, verilerin düzenlenmesi, analiz edilmesi ve işlemeye uygun hâle getirilmesi için kullanılmaktadır.

Ancak yapay zekâ uygulamalarında yalnızca veriyi düzenlemek yeterli değildir. Hazırlanan verilerden anlam çıkarılması, bu verilerle model oluşturulması ve yeni veriler üzerinde tahmin yapılması gerekir. Bu aşamada makine öğrenmesi kütüphaneleri kullanılır.

Makine öğrenmesi kütüphaneleri; verilerden örüntü çıkarma, model oluşturma, model eğitme, tahmin yürütme ve sonuçları değerlendirme işlemlerini kolaylaştıran yazılım araçlarıdır.

Python ekosisteminde yapay zekâ ve makine öğrenmesi uygulamalarının geliştirilmesinde kullanılan başlıca kütüphaneler ve temel kullanım amaçları Tablo 1.1'de gösterilmektedir.

Tablo 1.1: Yapay Zekânın Temel Yetenekleri

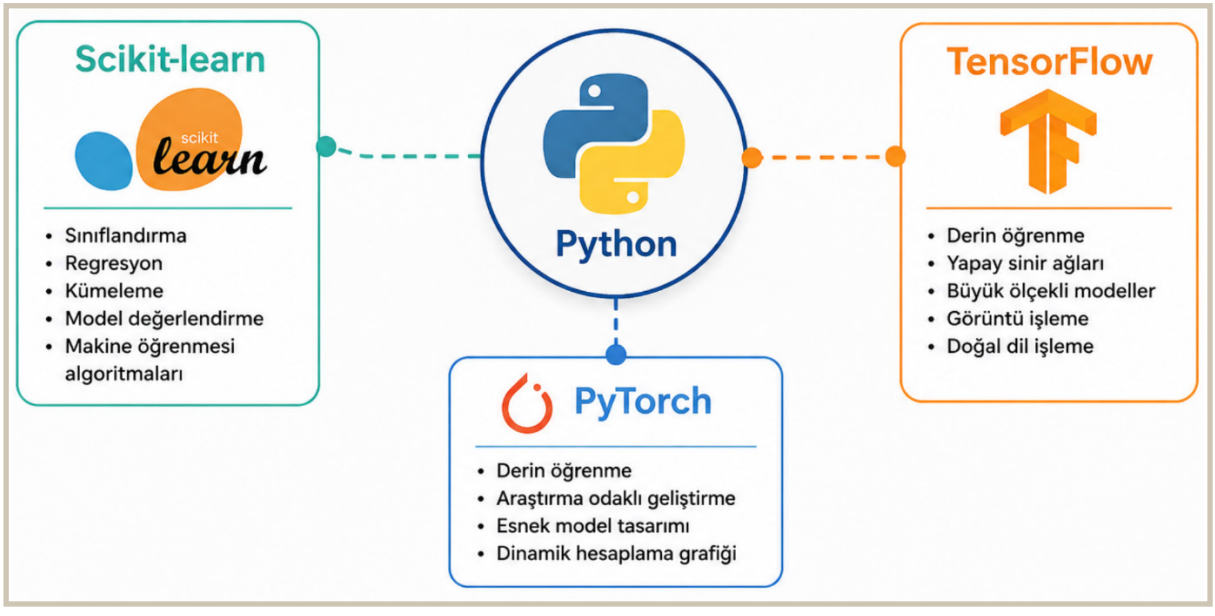
Kütüphane	Kullanım Amacı
Scikit-learn	Temel makine öğrenmesi modelleri oluşturma
TensorFlow	Derin öğrenme ve yapay sinir ağı uygulamaları geliştirme
PyTorch	Derin öğrenme modelleri tasarlama ve araştırma uygulamaları geliştirme

Bu kütüphaneler sayesinde geliştirilebilecek yapay zekâ uygulamalarına bazı örnekler şunlardır:

- Sınıflandırma
- Tahmin

- Model eğitimi
- Model değerlendirme
- Görüntü işleme
- Doğal dil işleme

Makine öğrenmesi kütüphaneleri, yapay zekâ uygulamalarının daha düzenli, hızlı ve anlaşılır şekilde geliştirilmesine yardımcı olur. Bu nedenle yapay zekâ projelerinde verilerin hazırlanmasından sonra uygun makine öğrenmesi kütüphanesinin seçilmesi önemli bir adımdır. Şema 1.3'te Python programlama dili etrafında yaygın kullanılan bazı makine öğrenmesi kütüphaneleri ve temel kullanım alanları gösterilmektedir.



Şema 1.3: Makine öğrenmesi kütüphaneleri genel görünümü

10. SIRA SİZDE

Aşağıdaki ifadeleri inceleyiniz ve karşısına kullanılabilecek uygun kütüphaneyi yazınız.

İfade	Kullanılabilecek Kütüphane
Basit bir sınıflandırma modeli oluşturmak	
Derin öğrenme modeli geliştirmek	
Yapay sinir ağı tabanlı uygulama hazırlamak	
Eğitim amaçlı temel makine öğrenmesi uygulaması yapmak	
Görüntü tanıma gibi gelişmiş yapay zekâ uygulamaları geliştirmek	

1.3.1. Scikit-learn Kütüphanesi

Scikit-learn, Python programlama dili için geliştirilmiş açık kaynaklı bir makine öğrenmesi kütüphanesidir. Kullanımı sade ve anlaşılır olduğu için özellikle eğitim amaçlı uygulamalarda ve başlangıç düzeyindeki makine öğrenmesi çalışmalarında yaygın olarak tercih edilir. Bu kütüphane sayesinde veriler makine öğrenmesi uygulamalarına uygun hâle getirilebilir, makine öğrenmesi modelleri oluşturulabilir ve yeni veriler üzerinde tahmin işlemleri gerçekleştirilebilir. Scikit-learn; sınıflandırma, regresyon, kümeleme, model değerlendirme gibi birçok farklı işlem için hazır araçlar sunmaktadır.

Makine öğrenmesi çalışmalarında genellikle belirli bir süreç izlenir. Öncelikle problemle ilgili veriler toplanır ve düzenlenir. Daha sonra bu veriler kullanılarak bir model eğitilir. Eğitim süreci tamamlandıktan sonra modelin yeni veriler üzerinde tahmin yürütmesi sağlanır. Son aşamadaysa modelin başarısı değerlendirilir.

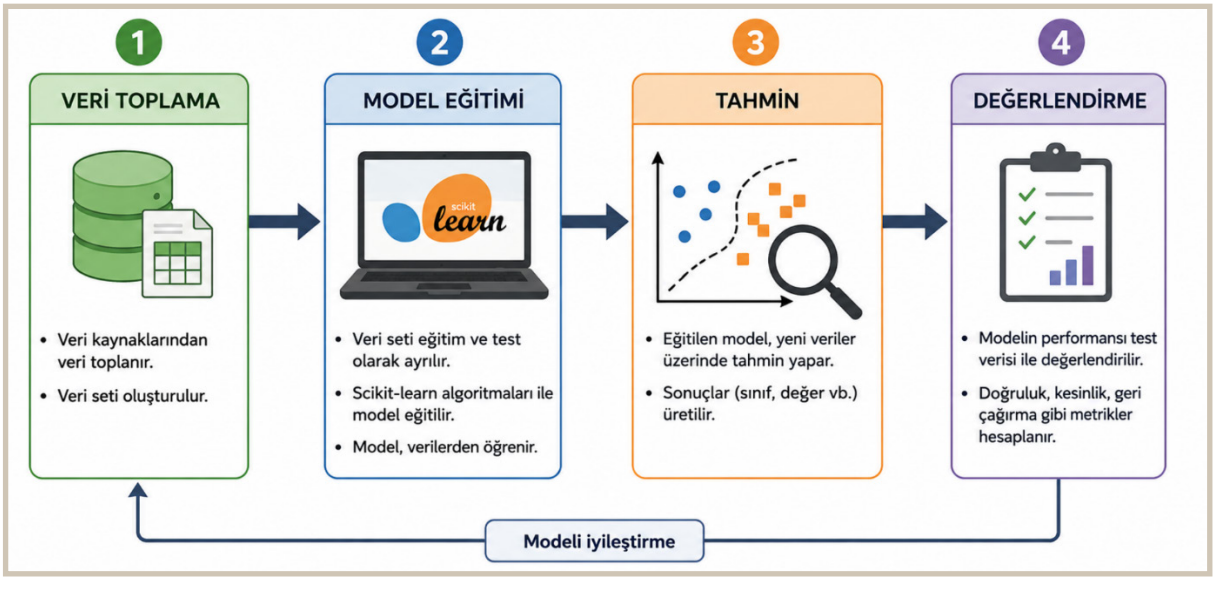
Tablo 1.2'de makine öğrenmesi uygulamalarında takip edilen işlem basamakları yer almaktadır.

Tablo 1.2: Makine Öğrenmesi Sürecinin Temel Aşamaları

Aşama	Açıklama
Veri toplama	Problem ile ilgili verilerin elde edilmesi
Veri hazırlama	Eksik veya hatalı verilerin düzenlenmesi
Model eğitimi	Algoritmanın eğitim verilerinden örüntüleri öğrenerek modeli oluşturması
Tahmin	Eğitilmiş modelin yeni veriler üzerinde tahmin yapması.
Değerlendirme	Model başarısının ölçülmesi

Tablo 1.2'de görüldüğü üzere makine öğrenmesi süreci veri toplama ile başlamakta, veri hazırlama ve model eğitimi aşamaları ile devam etmektedir. Eğitilen model daha sonra yeni veriler üzerinde tahmin yapmak için kullanılmakta ve son aşamada modelin performansı çeşitli değerlendirme ölçütleri kullanılarak analiz edilmektedir. Bu süreç, güvenilir ve başarılı yapay zekâ uygulamalarının geliştirilmesinde temel yaklaşım olarak kabul edilir.

Scikit-learn kütüphanesi bu süreçlerin büyük bölümünü daha kolay ve düzenli şekilde gerçekleştirmeye yardımcı olur. Görsel 1.21' de verilen bu öğrenme sürecini inceleyiniz.



Görsel 1.21: Scikit-learn'le makine öğrenmesi süreci

Scikit-learn kütüphanesi, Python ortamına kurularak ve program içinde içe aktarılarak kullanılabilir. Bunun için öncelikle ilgili modül programa dâhil edilir. Aşağıdaki örnekte doğrusal regresyon modeli için kullanılan LinearRegression sınıfı programa eklenmiştir.

```
from sklearn.linear_model import LinearRegression
```

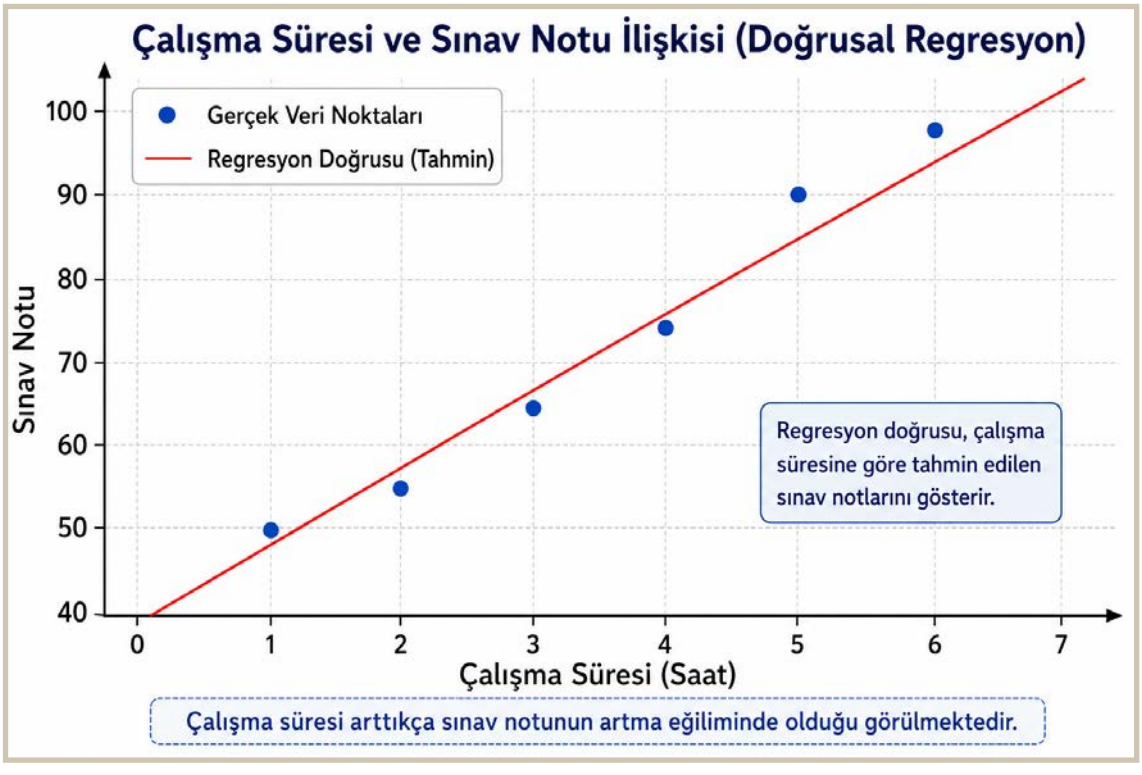
Bu işlemden sonra model oluşturulabilir ve veriler kullanılarak eğitilebilir.

Aşağıdaki örnekte öğrencilerin çalışma süreleriyle sınav notları arasındaki ilişki incelenmektedir.

Örnek

```
from sklearn.linear_model import LinearRegression
import numpy as np
# Çalışma süreleri
X = np.array([[1], [2], [3], [4], [5]])
# Sınav puanları
y = np.array([50, 60, 70, 80, 90])
# Model oluşturma
model = LinearRegression()
# Model eğitimi
model.fit(X, y)
# Tahmin işlemi
tahmin = model.predict([[6]])
print("Tahmini puan:", tahmin)
```

Bu örnekte model, çalışma süreleriyle sınav puanları arasındaki ilişkiyi öğrenmektedir. Daha sonra 6 saat çalışan bir öğrencinin alabileceği tahmini puan hesaplanmaktadır (Görsel 1.22).



Görsel 1.22: Çalışma süresi ve sınav notu ilişkisini gösteren regresyon grafiği

Kod içerisinde kullanılan bazı temel yapılar Tablo 1.3' te açıklanmıştır.

Tablo 1.3: Doğrusal Regresyon Uygulamasında Kullanılan Temel Yapılar ve Görevleri

Kod Yapısı	Açıklama
LinearRegression	Doğrusal regresyon modeli oluşturur.
fit()	Modeli verilen verilerle eğitir.
predict()	Yeni veriler için tahmin yapar.
X	Giriş verilerini temsil eder.
y	Hedef verileri temsil eder.

BİLGİ KUTUSU

Makine öğrenmesi modellerinin başarısı büyük ölçüde, kullanılan verilere bağlıdır. Eksik veya hatalı veriler modelin yanlış sonuçlar üretmesine neden olabilir. Bu nedenle model eğitimi öncesinde verilerin düzenlenmesi önemlidir.

11. SIRA SİZDE

Aşağıdaki işlemlerin makine öğrenmesi aşamalarından hangisinde gerçekleşebileceğini yazınız.

İşlem	Aşama
Öğrenci notlarının toplanması	
Eksik kayıtların düzenlenmesi	
Modelin öğrenme gerçekleştirmesi	
Yeni öğrenci başarısının tahmin edilmesi	
Model başarısının ölçülmesi	

1.3.2. TensorFlow ve PyTorch Kütüphaneleri

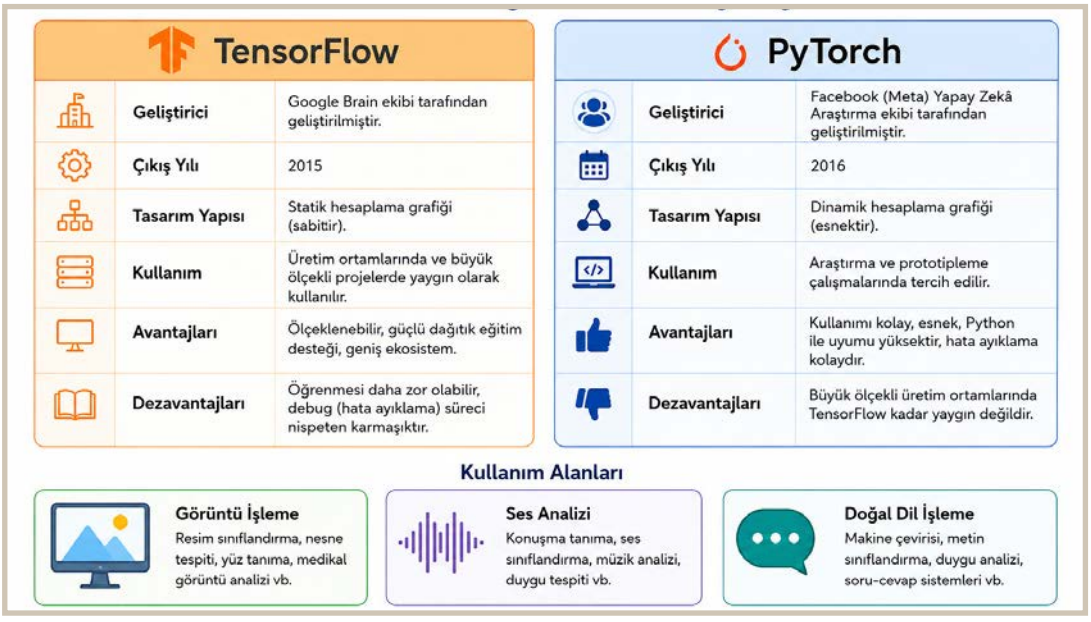
Makine öğrenmesi uygulamalarında bazı problemler daha gelişmiş yapılar gerektirir. Özellikle görüntü tanıma, ses işleme, doğal dil işleme gibi alanlarda derin öğrenme yöntemlerinden yararlanılmaktadır.

Derin öğrenme uygulamaları geliştirmek için yaygın olarak kullanılan kütüphanelerden bazıları TensorFlow ve PyTorch'tur. Bu kütüphaneler, yapay sinir ağları oluşturulmasına ve büyük veri kümeleri üzerinde model eğitilmesine yardımcı olmaktadır.

TensorFlow, açık kaynaklı bir makine öğrenmesi ve derin öğrenme kütüphanesidir. Büyük ölçekli yapay zekâ uygulamalarında yaygın olarak kullanılmaktadır. Görüntü işleme, ses tanıma, veri analizi gibi alanlarda tercih edilir.

PyTorch ise Meta tarafından geliştirilen açık kaynaklı bir derin öğrenme kütüphanesidir. Özellikle araştırma ve eğitim çalışmalarında yaygın olarak kullanılmaktadır. Esnek yapısı sayesinde farklı model tasarımlarının kolayca geliştirilmesine olanak sağlar.

Her iki kütüphane de yapay sinir ağlarıyla çalışmayı kolaylaştırmakta ve gelişmiş yapay zekâ uygulamalarının oluşturulmasına yardımcı olmaktadır. Görsel 1.23 'te TensorFlow ve PyTorch kütüphanelerinin bazı özellikleri karşılaştırılmıştır.



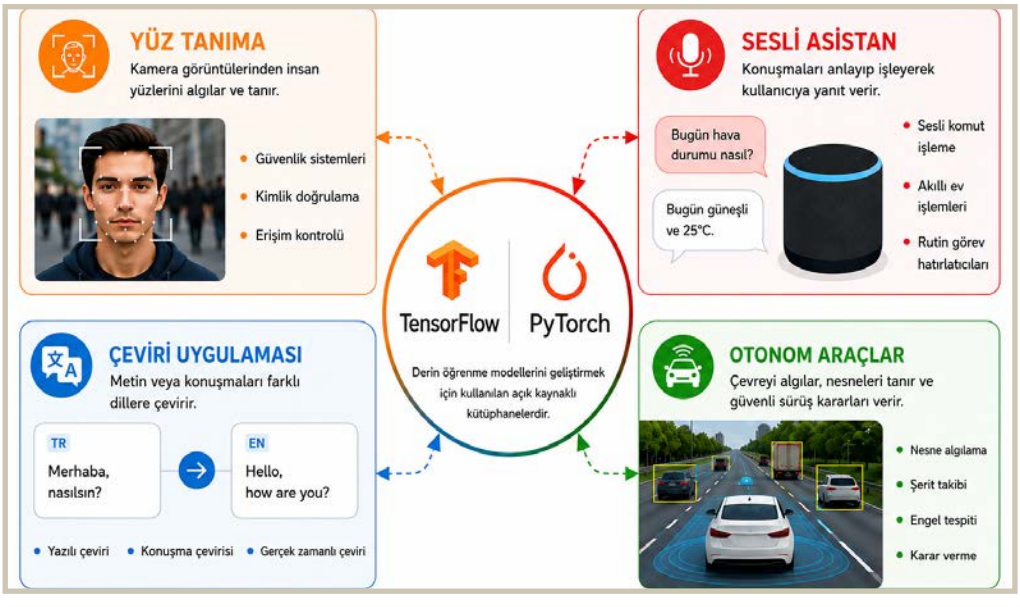
Görsel 1.23: TensorFlow ve PyTorch karşılaştırması

TensorFlow ve PyTorch kütüphaneleri Python programına eklenerek kullanılabilir. Aşağıdaki örneklerde ilgili kütüphanelerin programa nasıl dâhil edildiği gösterilmektedir.

```
import tensorflow as tf
import torch
```

Bu işlemlerden sonra kütüphanelerin sunduğu araçlar kullanılarak yapay zekâ modelleri geliştirilebilir.

TensorFlow ve PyTorch Görsel 1.24'te verilen alanlarda yaygın olarak kullanılmaktadır.



Görsel 1.24: Gerçek yaşamda TensorFlow ve PyTorch kullanım alanları

Günümüzde yüz tanıma sistemleri, sesli asistanlar, öneri sistemleri gibi birçok uygulama bu tür kütüphaneler kullanılarak geliştirilmektedir.

BİLGİ KUTUSU

Derin öğrenme uygulamalarında büyük miktarda veri ve güçlü işlem kapasitesi gerekebilir. Bu nedenle gelişmiş yapay zekâ sistemlerinde yüksek performanslı bilgisayarlar ve grafik işlem birimleri (GPU) kullanılır.

12. SIRA SİZDE

Aşağıdaki alanlar için kullanılabilecek kütüphaneleri yazınız.

Kullanım Alanı	Kullanılabilecek Kütüphane
Görüntü tanıma sistemi geliştirme	
Sesli asistan uygulaması geliştirme	
Yapay sinir ağı modeli oluşturma	
Derin öğrenme uygulaması geliştirme	
Eğitim amaçlı yapay zekâ araştırması yapma	

1.3.3. Scikit-learn ile Sınıflandırma Uygulaması

Makine öğrenmesinde sık kullanılan yöntemlerden biri sınıflandırmadır. Sınıflandırma işlemi, verilerin belirli kategorilere ayrılması anlamına gelir.

Örneğin:

- e-postaların spam veya normal olarak ayrılması,
- öğrencilerin başarılı veya başarısız olarak değerlendirilmesi,
- görüntülerde kedi veya köpek bulunup bulunmadığının belirlenmesi

birer sınıflandırma problemidir.

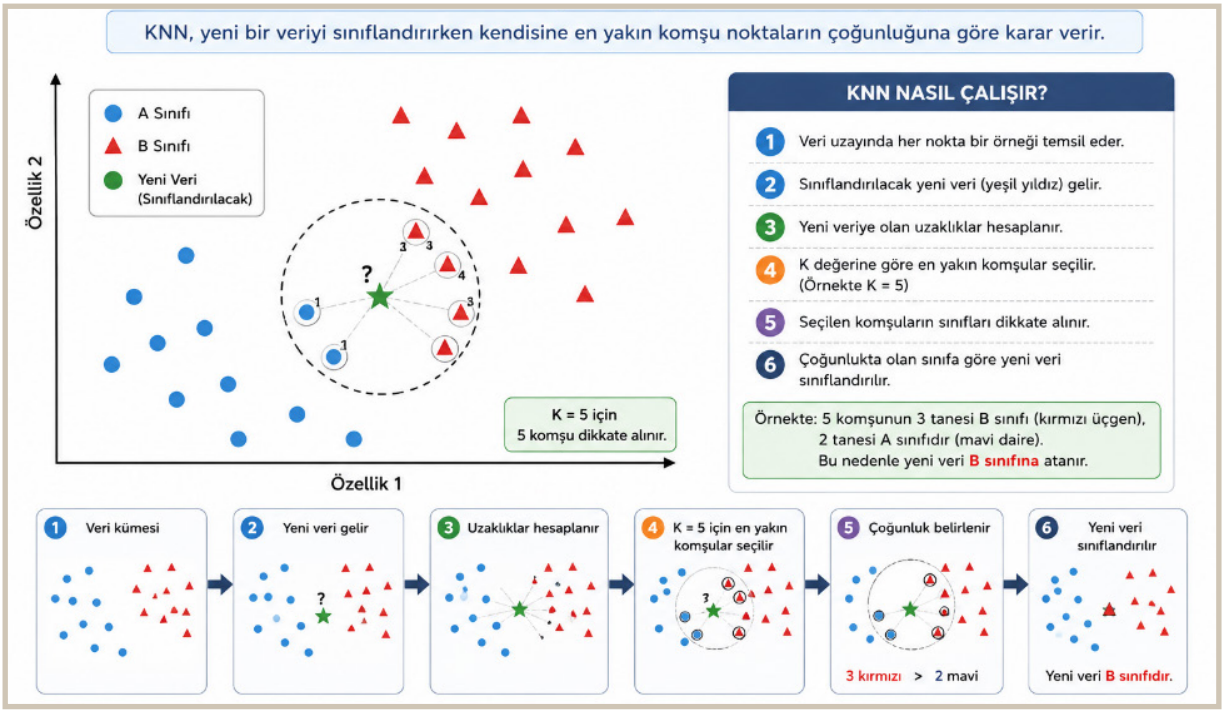
Scikit-learn kütüphanesi, sınıflandırma işlemleri için kullanılabilecek birçok hazır algoritma sunmaktadır. Bunlardan biri K-en Yakın Komşu (KNN) algoritmasıdır.

KNN algoritması, yeni bir veriyi eğitim verileri içindeki en yakın k komşusunun özelliklerine göre sınıflandırır. Örneğin çalışma süresi yüksek olan öğrenciler genellikle başarılıysa, benzer özelliklere sahip yeni bir öğrencinin de başarılı olabileceği tahmin edilebilir. Tablo 1.4'te KNN algoritmasının kullanılabileceği bazı sınıflandırma örnekleri verilmiştir.

Tablo 1.4: KNN Algoritması İçin Sınıflandırma Örnekleri

Problem Durumu	Sınıflar
E-posta analizi	Spam / Normal
Öğrenci başarısı	Başarılı / Başarısız
Görüntü tanıma	Kedi / Köpek
Sağlık analizi	Riskli / Risksiz

KNN algoritmasının çalışma mantığı Görsel 1.25'te gösterilmektedir. Algoritma, yeni bir veri noktasını en yakın komşularının sınıflarını dikkate alarak sınıflandırmaktadır.



Görsel 1.25: KNN algoritmasının çalışma mantığı

Scikit-learn kütüphanesinde sınıflandırma işlemleri için çeşitli modüller kullanılabilir. KNN algoritmasının programa dâhil edildiği bir örnek şu şekildedir:

```
from sklearn.neighbors import KNeighborsClassifier
```

Öğrencilerin çalışma sürelerine göre başarı durumlarını tahmin eden basit bir sınıflandırma modeli örneği şu şekildedir.

Örnek

```
from sklearn.neighbors import KNeighborsClassifier
import numpy as np
# Çalışma süreleri
X = np.array([[1], [2], [3], [4], [5], [6]])
# Başarı durumu
y = np.array([
  "Başarısız",
  "Başarısız",
  "Başarılı",
  "Başarılı",
  "Başarılı",
  "Başarılı"
```

```
])
# Model oluşturma
model = KNeighborsClassifier(n_neighbors=3)
# Model eğitimi
model.fit(X, y)
# Tahmin işlemi
tahmin = model.predict([[2]])
print("Tahmin edilen durum:", tahmin)
```

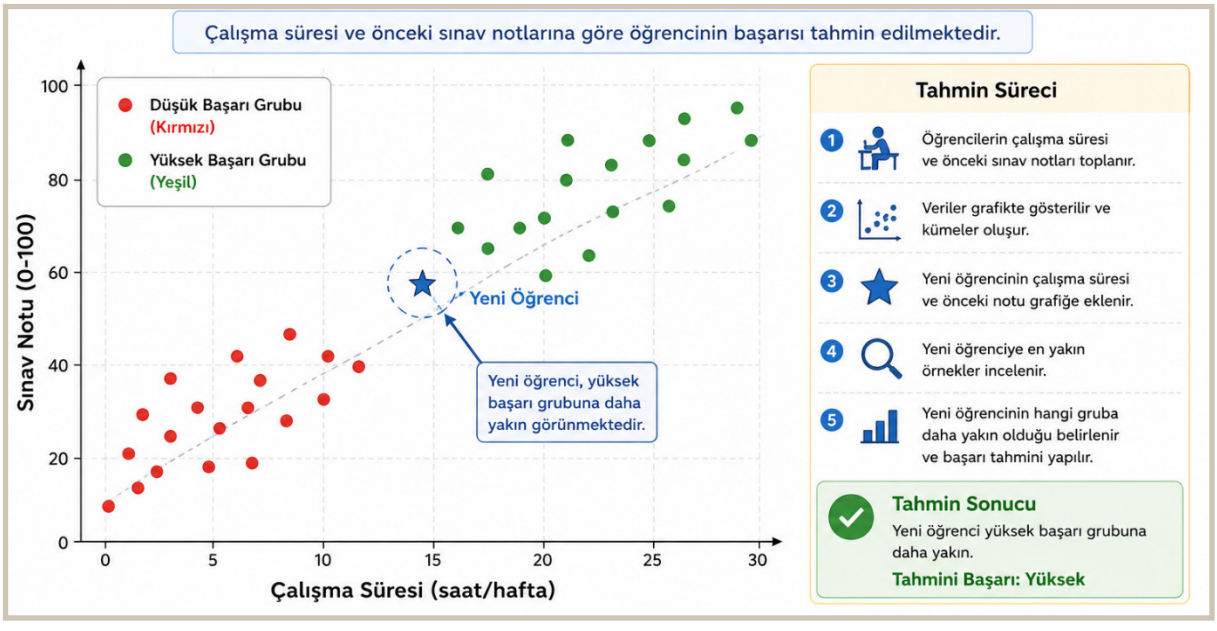
Örnekte model, öğrencilerin çalışma süreleri ile başarı durumları arasındaki ilişkiyi öğrenir. Daha sonra çalışma süresi verilen yeni bir öğrenci için tahmin işlemi gerçekleştirir.

Kod içerisinde kullanılan bazı temel yapılar Tablo 1.5'te açıklanmıştır.

Tablo 1.5: KNN Algoritması İçin Sınıflandırma Örnekleri

Kod Yapısı	Açıklama
KNeighborsClassifier	KNN modelini oluşturur.
n_neighbors	Karar verirken kullanılacak komşu sayısını belirler.
fit()	Modeli verilerle eğitir.
predict()	Başarılı / Başarısız

KNN algoritmasının çalışma mantığı, yeni bir verinin kendisine en yakın komşuların bulunduğu sınıfa atanması esasına dayanmaktadır. Öğrenci başarısı örneğinde çalışma süresi ve sınav notu bilgileri kullanılarak öğrenciler farklı başarı gruplarına ayrılmakta, daha sonra yeni bir öğrencinin hangi gruba daha yakın olduğu belirlenerek başarı tahmini yapılmaktadır. Bu sürecin görsel gösterimi Görsel 1.26'da verilmiştir.



Görsel 1.26: Öğrenci başarı tahmini örneği

BİLGİ KUTUSU

Makine öğrenmesi modelleri, sınıflandırma yaparken veriler arasındaki benzerlikleri dikkate alır. Örneğin okul kütüphanesindeki kitaplar; tür, yazar, konu, okunma sıklığı gibi özelliklerine göre gruplandırılabilir. Benzer şekilde yapay zekâ modeli de verilen özellikleri inceleyerek yeni verinin hangi sınıfa ait olabileceğini tahmin eder. Bu süreç modelin yalnızca ezber yapmadığını, verilerdeki ortak özelliklerden yararlanarak karar verdiğini gösterir.

13. SIRA SİZDE

Aşağıdaki problem durumlarını inceleyiniz. Sınıflandırma uygulaması olanları "✓", sınıflandırma uygulaması olmayanları ise "X" işaretiyle belirtiniz.

Problem Durumu	Sınıflandırma Uygulaması mı?
E-postanın gereksiz olup olmadığını belirleme	
Hava sıcaklığını tahmin etme	
Görüntüde kedi olup olmadığını belirleme	
Öğrenci not ortalamasını hesaplama	
Hastalık risk durumunu belirleme	

1.3.4. Model Performansının Değerlendirilmesi

Eğitilen makine öğrenmesi modellerinin performanslarının değerlendirilmesi gerekir. Bu değerlendirme sayesinde modelin doğru tahmin yapma düzeyi incelenebilir.

Bir modelin performansı değerlendirilirken yararlanılan bazı ölçütler şunlardır:

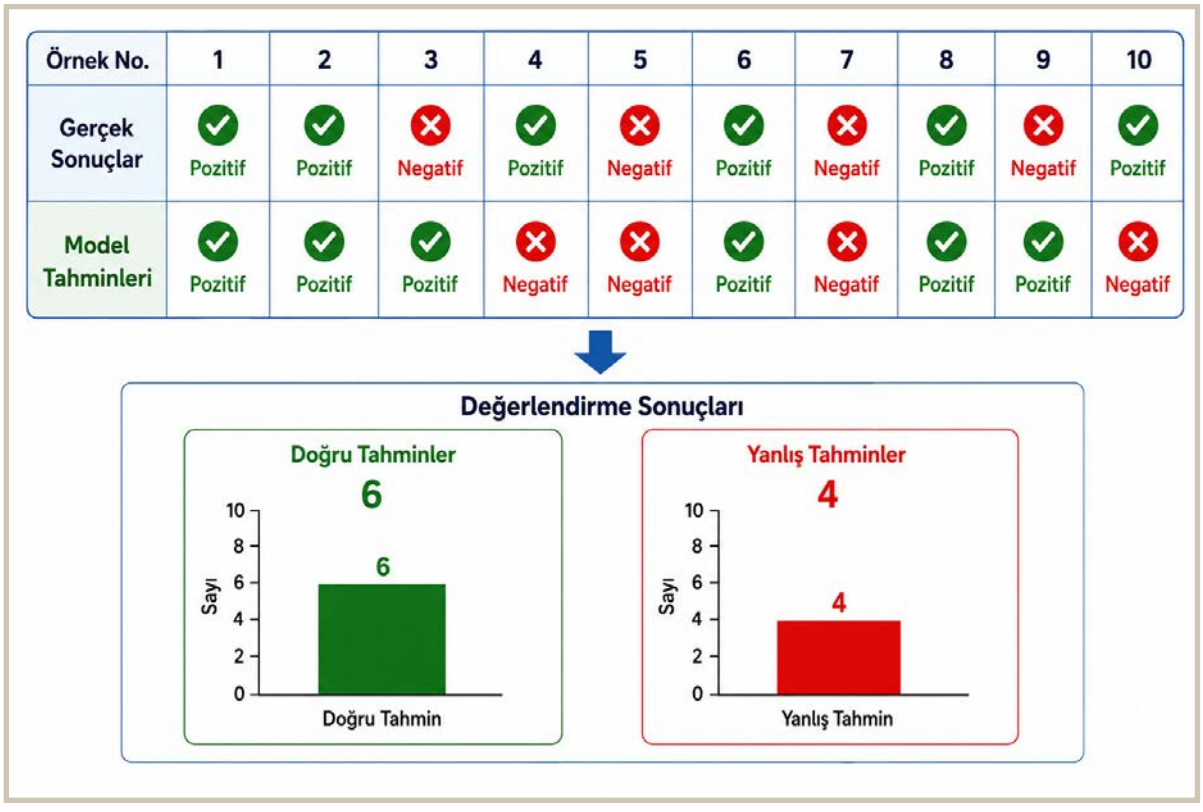
- Doğru tahmin sayısı
- Yanlış tahmin sayısı
- Başarı oranı

Makine öğrenmesinde sık kullanılan ölçütlerden biri doğruluk oranıdır (accuracy). Doğruluk oranı, modelin yaptığı doğru tahminlerin toplam tahmin sayısına oranını ifade eder. Örneğin bir model, gerçekleştirdiği 100 tahminin 90 tanesini doğru yapıyorsa doğruluk oranı %90 olarak hesaplanır. Örnek bir model değerlendirme sonucu Tablo 1.6'da gösterilmiştir.

Tablo 1.6: Örnek Model Değerlendirme Sonuçları

Toplam Tahmin Sayısı	Doğru Tahmin Sayısı	Yanlış Tahmin Sayısı	Doğruluk Oranı
100	90	10	90%

Tablo 1.6 incelendiğinde, modelin gerçekleştirdiği 100 tahminin 90 tanesini doğru yaptığı ve yalnızca 10 tahminde hata yaptığı görülmektedir. Bu durumda modelin doğruluk oranı %90 olarak hesaplanmaktadır. Doğruluk oranı yüksek olan modeller genellikle daha başarılı kabul edilmekle birlikte, özellikle dengesiz veri kümelerinde tek başına yeterli bir değerlendirme ölçütü olmayabilir. Bu nedenle uygulamalarda doğruluk oranının yanı sıra kesinlik (precision), duyarlılık (recall), F1 skoru gibi ek performans ölçütleri de kullanılabilmektedir. Model değerlendirme sürecinin genel akışı Grafik 1.1'de gösterilmiştir.



Grafik 1.1: Model değerlendirme süreci

Scikit-learn kütüphanesi model performansını değerlendirmek için çeşitli araçlar sunmaktadır. Model değerlendirme süreci grafiğindeki örnek için doğruluk oranı hesaplaması şu şekildedir:

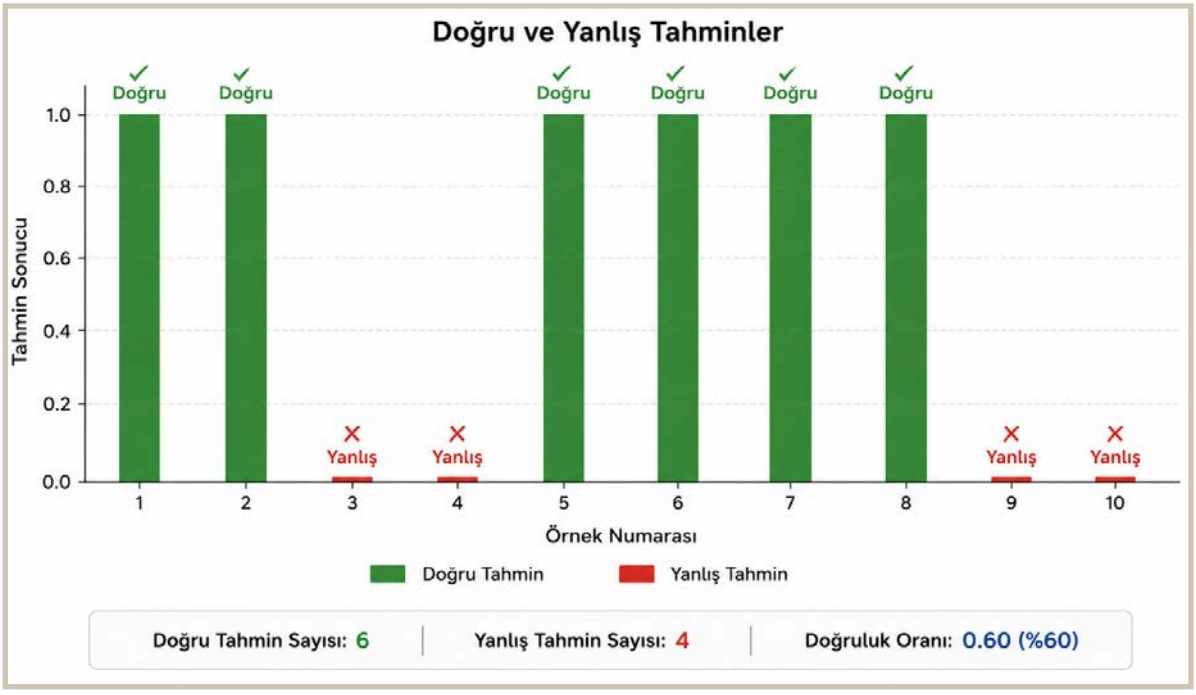
```
from sklearn.metrics import accuracy_score
# Gerçek sonuçlar
gercek = [1, 1, 0, 1, 0, 1, 0, 1, 0, 1]
# Model tahminleri
tahmin = [1, 1, 1, 0, 0, 1, 0, 1, 1, 0]
# Doğruluk oranı hesaplama
sonuc = accuracy_score(gercek, tahmin)
print("Doğruluk oranı:", sonuc)
```

Örnekte modelin yaptığı tahminler ile gerçek sonuçlar karşılaştırılmaktadır. accuracy_score() fonksiyonu kullanılarak modelin doğruluk oranı hesaplanmaktadır. Kod içerisinde kullanılan bazı temel yapılar Tablo 1.7'de açıklanmıştır.

Tablo 1.7: Doğruluk Oranı Hesaplamasında Kullanılan Temel Yapılar

Kod Yapısı	Açıklama
accuracy_score	Doğruluk oranını hesaplar.
gercek	Gerçek sonuçları temsil eder.
tahmin	Model tahminlerini temsil eder.

Tablo 1.7 incelendiğinde, model performansının değerlendirilmesinde gerçek sonuçlar ile model tahminlerinin karşılaştırıldığı görülmektedir. accuracy_score fonksiyonu bu iki veri kümesini karşılaştırarak modelin doğru tahmin oranını hesaplamaktadır. Elde edilen doğruluk oranı, modelin genel başarısını değerlendirmek için kullanılan temel performans ölçütlerinden biridir ve farklı modellerin karşılaştırılmasında önemli rol oynamaktadır. Modelin doğru ve yanlış tahminlerinin dağılımı Grafik 1.2'de gösterilmektedir.



Grafik 1.2: Doğru ve yanlış tahminleri gösteren grafik

BİLGİ KUTUSU

Modelin yüksek doğruluk oranına sahip olması, her zaman kusursuz çalıştığı anlamına gelmez. Örneğin 100 tahminden 90'ını doğru yapan bir model başarılı görünebilir ancak modelin yaptığı 10 hatanın hangi durumlarda ortaya çıktığı da önemlidir. Bu nedenle model değerlendirmesinde yalnızca doğru sonuçlara değil hataların nedenlerine de bakılır.

14. SIRA SİZDE

Aşağıdaki tabloyu inceleyiniz ve modelin doğruluk oranını hesaplayınız.

Öğrenci	Gerçek Durum	Model Tahmini
Ali	Başarılı	Başarılı
Ayşe	Başarısız	Başarılı
Mehmet	Başarılı	Başarılı
Zeynep	Başarılı	Başarısız
Can	Başarısız	Başarısız

Sorular

1. Model kaç doğru tahmin yapmıştır?

   .....

2. Model kaç yanlış tahmin yapmıştır?

   .....

3. Doğruluk oranı yüzde kaçtır?

   .....

1.3.5. Gerçek Yaşamda Makine Öğrenmesi Kütüphaneleri

Makine öğrenmesi kütüphaneleri yalnızca eğitim amaçlı örnek uygulamalarda değil birçok dijital sistemde de kullanılmaktadır.

Günümüzde; öneri sistemleri, görüntü tanıma uygulamaları, sesli asistanlar, sağlık analiz sistemleri, navigasyon uygulamaları gibi birçok teknoloji, makine öğrenmesi yöntemlerinden yararlanmaktadır.

Bu sistemler büyük miktarda veriyi analiz ederek kullanıcı davranışlarını inceleyip yeni durumlar hakkında tahminler üretmektedir.

Makine öğrenmesi kütüphaneleri sayesinde geliştirilen uygulamalar; sağlık, eğitim, ulaşım, finans, sosyal medya gibi birçok alanda kullanılmaktadır. Makine öğrenmesinin farklı alanlardaki kullanım örnekleri Tablo 1.8'de gösterilmiştir.

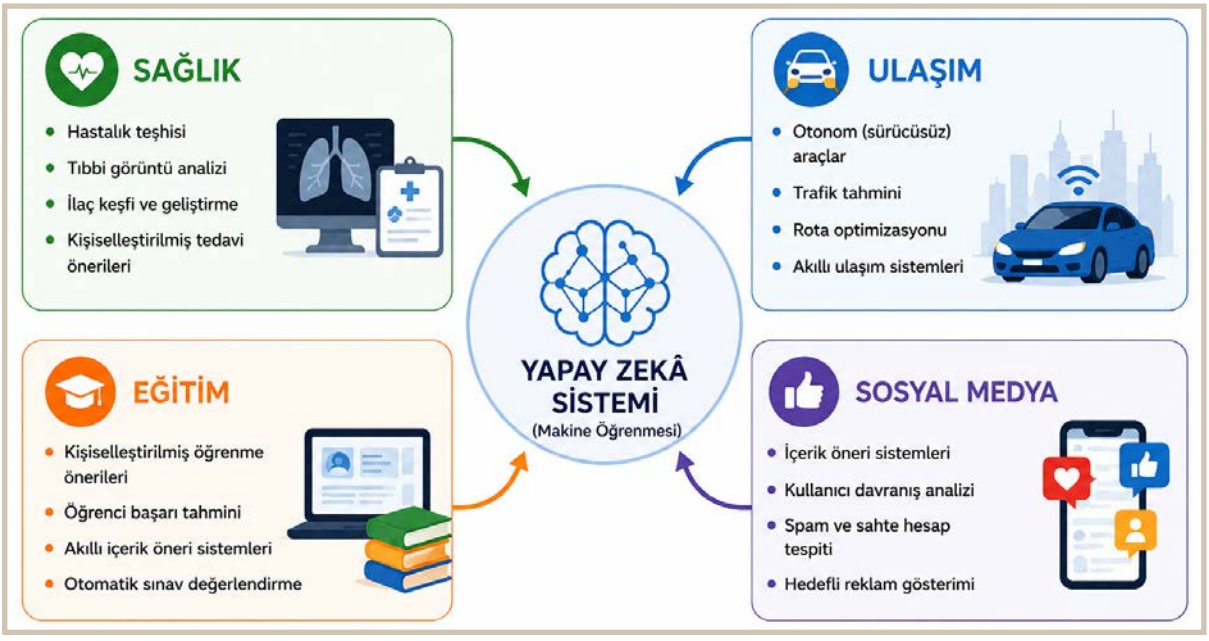
Tablo 1.8: Makine Öğrenmesinin Farklı Alanlardaki Kullanım Örnekleri

Alan	Kullanım Örneği
Sağlık	1. Hastalık risk analizi
Eğitim	2. Öğrenci başarı değerlendirmesi
Ulaşım	3. Trafik yoğunluğu tahmini
Finans	4. Şüpheli işlem tespiti
Sosyal Medya	5. İçerik öneri sistemleri

Tablo 1.8 incelendiğinde, makine öğrenmesi teknolojilerinin günümüzde çok farklı sektörlerde yaygın olarak kullanıldığı görülmektedir. Sağlık alanında hastalıkların erken teşhisine destek sağlanırken eğitim alanında öğrenci performanslarının değerlendirilmesine katkıda bulunulmaktadır. Ulaşım sektöründe trafik yoğunluğu tahminleri yapılabilmekte, finans alanında dolandırıcılık ve şüpheli işlemler tespit edilebilmekte, sosyal medya platformlarında ise kullanıcıların ilgi alanlarına uygun içerik önerileri sunulabilmektedir. Bu durum, makine öğrenmesinin günlük yaşamın birçok alanında önemli bir teknoloji hâline geldiğini göstermektedir.

Makine öğrenmesi teknolojileri, büyük veri kümelerinden anlamlı bilgiler elde edilmesini sağlayarak karar verme süreçlerini desteklemektedir. Günümüzde bireylerin günlük yaşamında kullandığı birçok dijital uygulama, veri analizi ve tahmin mekanizmaları için makine öğrenmesi yöntemlerinden yararlanmaktadır.

Teknolojinin gelişmesiyle birlikte bu uygulamaların kullanım alanlarının daha da genişlemesi beklenmektedir (Görsel 1.27).

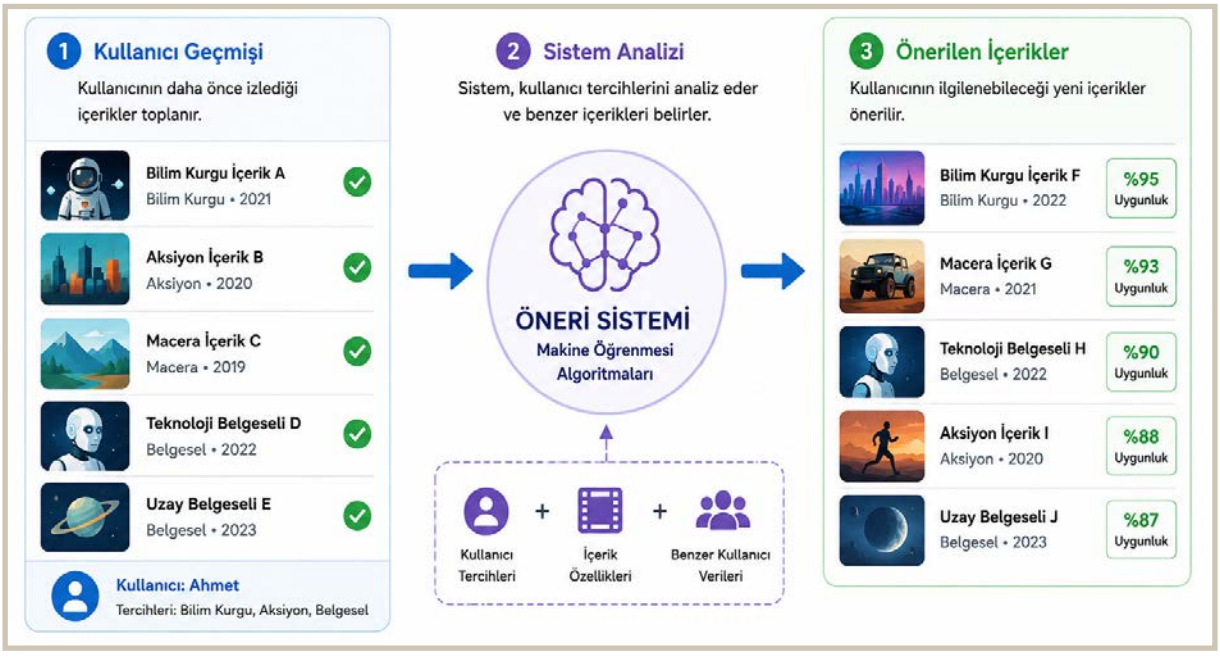


Görsel 1.27: Gerçek yaşamda makine öğrenmesi kullanım alanları

Sosyal medya platformları kullanıcısının izlediği içerikleri, yaptığı aramaları, beğendiği paylaşımları inceleyerek yeni içerik önerilerinde bulunabilir.

Bu uygulamaların büyük bölümü Scikit-learn, TensorFlow, PyTorch gibi makine öğrenmesi kütüphaneleri kullanılarak geliştirilmektedir.

Görsel 1.28 incelendiğinde, sistemin kullanıcının geçmiş tercihlerini ve benzer kullanıcıların davranışlarını analiz ederek yeni içerik önerilerinde bulunduğu görülmektedir. Bu yaklaşım, günümüzde video platformları, müzik uygulamaları ve e-ticaret sitelerinde yaygın olarak kullanılmaktadır.



Görsel 1.28: Öneri sistemi örneği

BİLGİ KUTUSU

Makine öğrenmesi bazı durumlarda kullanıcıdan önce davranarak olası ihtiyaçları tahmin edebilir. Bir uygulamanın içerik önermesi, trafik yoğunluğunu önceden hesaplaması veya riskli bir işlemi fark etmesi buna örnektir. Bu sistemler geleceği kesin olarak bilmez, geçmiş verilerdeki benzerliklerden yola çıkarak en olası sonucu tahmin eder.

15. SIRA SİZDE

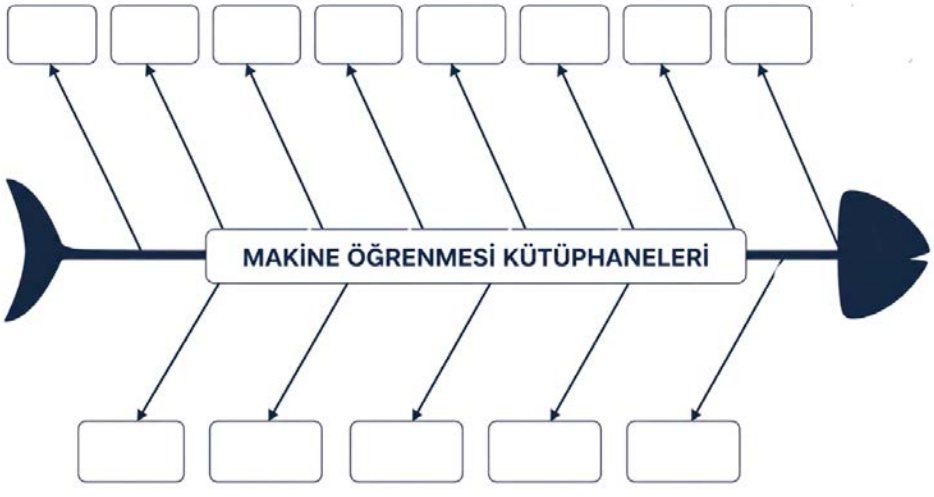
Aşağıda makine öğrenmesi kütüphaneleri ve makine öğrenmesi sürecinde kullanılan bazı kavramlar verilmiştir. Bu kavramları inceleyiniz ve uygun başlıklar altında sınıflandırınız.

Kavramlar

Scikit-learn, TensorFlow, PyTorch, Sınıflandırma, Yapay sinir ağları, fit(), predict(), KNN, Derin öğrenme, accuracy_score(), Regresyon, Görüntü işleme, Tahmin

Yönerge

- Üst kısma temel makine öğrenmesi kavramlarıyla ilgili ifadeleri yazınız.
- Alt kısma derin öğrenme ve gelişmiş yapay zekâ uygulamalarıyla ilgili ifadeleri yazınız.



NOTLAR

1.6. ÖRNEK UYGULAMA

Adı	Scikit-learn Kullanılarak Öğrenci Başarı Tahmini
Süre	1 ders saati
Amaç	Scikit-learn kütüphanesini kullanarak basit bir sınıflandırma modeli oluşturmak, modeli eğitmek ve tahmin yapmak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Python programlama dili, Jupyter Notebook, Scikit-learn ve NumPy kütüphaneleri

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni bir Notebook oluşturunuz ve adını **"Makine_Ogrenmesi_Uygulamasi"** olarak kaydediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç ve gereçleri dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Kodları düzenli, dikkatli ve adım adım yazınız.
- Yapılan tahminlerin nedenlerini sorgulayınız ve elde edilen sonuçları değerlendiriniz.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Aşağıdaki kodu yazarak gerekli kütüphaneleri programa dâhil ediniz.

```
from sklearn.neighbors import KNeighborsClassifier
import numpy as np
```

2. Adım: Veri Setinin Tanımlanması

Öğrencilerin çalışma sürelerini ve başarı durumlarını tanımlayınız.

```
X = np.array([[1], [2], [3], [4], [5], [6]])
y = np.array([
  "Başarısız",
  "Başarısız",
  "Başarılı",
  "Başarılı",
  "Başarılı",
  "Başarılı"
])
```

3. Adım: Model Oluşturma

KNN sınıflandırma modeli oluşturunuz.

```
model = KNeighborsClassifier(n_neighbors=3)
```

4. Adım: Model Eğitimi

Modeli veriler kullanılarak eğitiniz.

```
model.fit(X, y)
```

5. Adım: Tahmin Yapılması

Yeni bir öğrenci için tahmin işlemini gerçekleştiriniz.

```
tahmin = model.predict([[2]])
print("Tahmin edilen durum:", tahmin)
```

6. Adım: Sonuçların İncelenmesi

Modelin verdiği sonuçları değerlendiriniz.

- Çalışma süresi arttığında başarı durumu nasıl değişmektedir?
- Model hangi verilere göre karar vermektedir?
- Farklı veriler kullanılırsa sonuç değişir mi?

Sonuç

1. **Makine öğrenmesi modeli tahmin yaparken hangi verileri kullanmaktadır? Açıklayınız.**

.....
.....
.....
.....
.....

2. **Modelin yanlış tahmin yapmasına neden olabilecek durumlar nelerdir? Açıklayınız.**

.....
.....
.....
.....
.....

3. **KNN algoritması yeni veriyi nasıl sınıflandırır? Açıklayınız.**

.....
.....
.....
.....
.....

4. **Makine öğrenmesi uygulamalarında veri kalitesi neden önemlidir? Açıklayınız.**

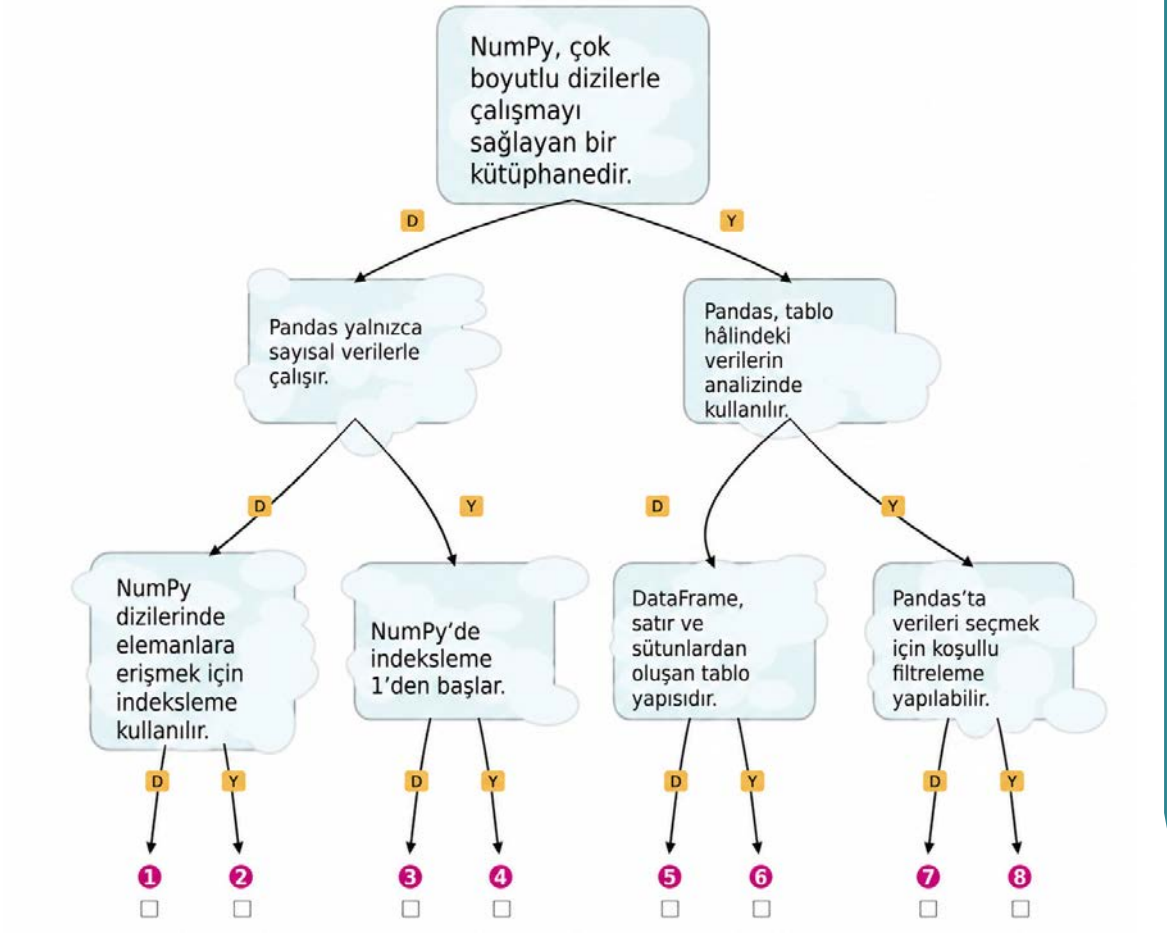
.....
.....
.....
.....
.....

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki tabloyu doldurarak NumPy ve Pandas kütüphanelerini karşılaştırınız.

Ölçüt	NumPy	Pandas
Kullanım Amacı		
Veri Yapısı		
İşlem Türü		
Kullanım Alanı		

B) Aşağıdaki ifadeleri doğru (D) veya yanlış (Y) olarak değerlendiriniz ve uygun yolu takip ediniz.



C) Aşağıdaki kavramları uygun kütüphaneyle eşleştiriniz.

Kavram	Kütüphane
1. Yapay sinir ağları	A) TensorFlow / PyTorch
2. Sınıflandırma	B) Scikit-learn
3. Derin öğrenme	C) TensorFlow / PyTorch
4. Regresyon	Ç) Scikit-learn
5. Görüntü işleme	D) TensorFlow / PyTorch

Ç) Aşağıdaki soruların doğru cevabını işaretleyiniz.

1. Aşağıdaki seçeneklerden hangisi yapay zekâyı en doğru şekilde tanımlar?

A) Bilgisayar sistemlerinin veri işleme ve hesaplama performansını artırmaya yönelik teknoloji lerin genel adıdır.

B) İnsan zekâsına özgü öğrenme, akıl yürütme, problem çözme ve karar verme gibi bilişsel süreç-lerin makineler tarafından gerçekleştirilmesini amaçlayan bilim ve mühendislik alanıdır.

C) Bilgisayar ağlarında veri güvenliğini sağlamak amacıyla geliştirilen yöntem ve teknolojiler bütünüdür.

D) Dijital ortamlarda grafik ve görsel içeriklerin daha gerçekçi üretilmesini sağlayan tekniklerin genel adıdır.

E) Fiziksel görevleri yerine getirebilen robotların tasarımı ve geliştirilmesiyle ilgilenen mühen dislik alanıdır.

2. Büyük veri yığınlarındaki örüntüleri analiz ederek zamanla yetkinlik kazanma, yapay zekânın temel özelliklerinden aşağıdakilerden hangisidir?

A) Algılama

B) Problem çözme

C) Öğrenme

D) Akıl yürütme

E) Görselleştirme

3. Bir otonom robot, sensörlerle önündeki engeli fark edip ardından en güvenli yönde hareket etme kararı vermektedir. **Bu durumda doğrudan birlikte çalışan iki temel özellik aşağıdakilerden hangisidir?**

A) Algılama ve problem çözme
B) Öğrenme ve çeviri
C) Akıl yürütme ve eğlence analizi
D) Görselleştirme ve sınıflandırma
E) Veri depolama ve şifreleme

4. **Aşağıdaki ifadelerden hangisi AlphaGo, Transformer ve üretken yapay zekâ ile ilgili doğru bilgidir?**

A) AlphaGo, Transformer ve üretken yapay zekâ aynı dönemde geliştirilmiş olup benzer amaçlarla kullanılan teknolojilerdir.
B) AlphaGo, yapay zekânın stratejik karar verme yeteneklerini göstermiş; Transformer mimarisi büyük dil modellerinin geliştirilmesine katkı sağlamış; üretken yapay zekâ ise metin, görsel ve kod üretimi gibi uygulamaların yaygınlaşmasına zemin hazırlamıştır.
C) AlphaGo'nun geliştirilmesi, Transformer mimarisinin ortaya çıkmasından sonra gerçekleşmiş ve bu mimariye dayanmaktadır.
D) Üretken yapay zekâ sistemleri, kural tabanlı uzman sistemlerin yöntemlerini kullanarak geliştirilmiştir.
E) AlphaGo, Transformer ve üretken yapay zekâ gelişmeleri ağırlıklı olarak oyun teknolojilerinin ilerlemesine katkı sağlamıştır.

5. Elazığ Havalimanında, yolcuların yüzlerini tanıyarak kimlik kontrolü yapan bir sisteme geçilmiştir. **Bu sistemin merkezinde yer alan yapay zekâ alt dalı aşağıdakilerden hangisidir?**

A) Uzman sistemler
B) Robotik
C) Bilgisayarlı görü
D) Doğal dil işleme
E) Karar teorisi

6. NumPy kütüphanesinin yapay zekâ ve veri analizi süreçlerinde tercih edilmesinin temel nedeni aşağıdakilerden hangisidir?

A) Metin tabanlı verileri düzenlemesi

B) Görsel tasarım yapmaya imkân sağlaması

C) Sayısal veriler üzerinde hızlı matematiksel işlemler yapabilmesi

D) İnternet bağlantısını yönetmesi

E) Kullanıcı arayüzü oluşturması

7. Aşağıdakilerden hangisi Pandas kütüphanesinin temel kullanım amaçlarından biridir?

A) Donanım kontrolü sağlamak

B) Oyun geliştirmek

C) Tablo hâlindeki verileri düzenlemek ve analiz etmek

D) Grafik kartını yönetmek

E) Ağ bağlantılarını kurmak

8. Bir veri bilimci, CSV dosyasındaki verileri Python ortamına aktarmak istemektedir. Bu işlem için aşağıdaki komutlardan hangisi kullanılır?

A) np.array()

B) print()

C) pd.read_csv()

D) input()

E) open()

9. Scikit-learn kütüphanesinin temel kullanım amacı aşağıdakilerden hangisidir?

A) Oyun geliştirmek

B) Temel makine öğrenmesi uygulamaları geliştirmek

C) İnternet bağlantısı kurmak

D) Donanım kontrolü yapmak

E) Grafik tasarımı yapmak

10. NumPy dizilerinde yapılan işlemlerle ilgili aşağıdakilerden hangisi doğrudur?

A) İşlemler sadece tek bir eleman üzerinde yapılabilir.
B) Dizideki tüm elemanlara aynı anda işlem uygulanabilir.
C) Diziler sadece metin verisi içerir.
D) İşlem yapmak için internet gerekir.
E) Diziler sadece iki elemanlı olabilir.

11. Aşağıdakilerden hangisi sınıflandırma problemine örnektir?

A) Hava sıcaklığını ölçmek
B) Öğrenci not ortalamasını hesaplamak
C) E-postayı gereksiz veya normal olarak ayırmak
D) Bilgisayarı yeniden başlatmak
E) Dosya kaydetmek

12. Makine öğrenmesi modellerinde veri kalitesinin önemli olmasının temel nedeni aşağıdakilerden hangisidir?

A) Bilgisayarın rengini değiştirmesi
B) İnternet hızını artırması
C) Daha doğru tahminler yapılmasını sağlaması
D) Yazıcıyı çalıştırması
E) Ekranı büyütmesi

13. Aşağıdakilerden hangisi gerçek yaşamda makine öğrenmesi kullanım alanlarından biridir?

A) Trafik yoğunluğu tahmini
B) Klavye temizleme
C) Monitör taşıma
D) Fare tamiri
E) Kablo bağlama

D) Aşağıdaki metni okuyarak ilgili soruları cevaplayınız.

Akıllı Okul Projesi

Bir lisede yapay zekâ dersi alan öğrencilerden, okul yönetimi için **Akıllı Okul Veri Analiz Sistemi** geliştirmeleri istenir. Bu sistem sayesinde okulda bulunan öğrencilerin başarı durumları analiz edilecek, veriler düzenlenecek ve bazı hesaplamalar otomatik olarak yapılacaktır.

Proje ekibinde yer alan **Ahmet,** veri işleme kısmından sorumludur. Ahmet'in elinde farklı türde veriler bulunmaktadır Bu veriler şunlardır:

- Öğrencilerin sınav notları (sayısal veriler)
- Öğrenci isimleri ve sınıf bilgileri (metinsel + tablo verileri)
- CSV formatında saklanan büyük veri dosyaları
- Çok büyük boyutlu sayısal veri listeleri

Ahmet, bu verileri işlerken Python dilinde kullanılan farklı kütüphanelerden yararlanması gerektiğini fark eder. Ancak hangi durumda hangi kütüphaneyi kullanacağına karar vermesi gerekmektedir.

Bu süreçte Ahmet'in karşılaştığı bazı durumlar şunlardır:

1. Ahmet, elindeki **binlerce öğrencinin sınav notlarını hızlıca toplamak, ortalamasını almak ve çeşitli matematiksel işlemler yapmak** istemektedir.
   - Bu durumda Ahmet hangi kütüphaneyi kullanmalıdır?
   - Bu kütüphane neden daha uygundur?
2. Ahmet, öğrencilerin **ad, soyad, sınıf ve not bilgilerini tablo hâlinde düzenlemek, sıralamak ve filtrelemek** istemektedir.
   - Bu durumda Ahmet hangi kütüphaneyi kullanmalıdır?
   - Bu kütüphane neden daha uygundur?
3. Ahmet, okuldan aldığı büyük bir **CSV dosyasını bilgisayara aktarıp içindeki verileri okumak, eksik verileri tespit etmek ve belirli ölçütleri göre filtrelemek** istemektedir.
   - Bu durumda Ahmet hangi kütüphaneyi kullanmalıdır?
   - Bu kütüphane neden daha uygundur?



2. ÖĞRENME BİRİMİ

ETİK VE SOSYAL ETKİLER

KONULAR

2.1. YAPAY ZEKÂNIN ETİK BOYUTLARI
2.2. YAPAY ZEKÂNIN YASAL BOYUTLARI

ANAHTAR KAVRAMLAR

- uluslararası hukuk
- veri gizliliği
- veri güvenliği
- yapay zekâ etiği



BEKLENTİLER

Bu öğrenme biriminde sizden

- yapay zekânın etik boyutlarıyla veri gizliliği ve güvenliği üzerindeki etkilerini örneklerle açıklamanız
- yapay zekânın veri gizliliği ve güvenliği üzerindeki potansiyel olumsuz etkilerini ve bu etkilere karşı alınabilecek önlemleri sıralamanız
- yapay zekâyla ilgili yasal düzenlemelerin gerekliliğini ve odaklandığı temel alanları açıklamanız
- ulusal veya uluslararası güncel yapay zekâ yasal düzenlemeleri açıklamanız

beklenmektedir.

Öğrenme Birimi Karekodu

BAŞLARKEN

Yapay zekâ teknolojileri, günümüzde pek çok alanda etkisini hissettirmektedir. Akıllı asistanlar, öneri sistemleri, otomasyon uygulamaları gibi işlemlerle günlük yaşamı kolaylaştıran yapay zekâ, üretim süreçlerinde verimliliği artırmakta ve yeni iş alanlarının ortaya çıkmasını sağlamaktadır. Bu gelişmeler bazı tartışmaları da beraberinde getirmektedir. Kaybolabilecek meslekler ve bu durumun olası sosyolojik etkileri, veri güvenliği, etik sorunlar; bu tartışmaların temelini oluşturur. Yapay zekânın çok yönlü etkilerinin anlaşılması için önemi, işlevi ve kullanım alanları bütüncül şekilde değerlendirilmelidir.

ÖNCE DÜŞÜN

Bir teknoloji firması, üretimi hızlandırmak ve güvenliği artırmak amacıyla yapay zekâ destekli sistemleri kullanmaya başlamıştır. Firmanın giriş kapısına yerleştirilen yüz tanıma sistemi sayesinde çalışanların iş yerine giriş çıkışları otomatik olarak kontrol edilmiş ve işletme güvenliği bu yolla sağlanmıştır. Ancak sistem, çalışanların yüz verilerini kaydedip saklamaktadır. Firmanın üretim sürecini iyileştirmek amacıyla kullandığı yapay zekâ destekli sistemler sayesinde daha az iş gücüyle daha hızlı üretim mümkün olmuştur. İş gücü ihtiyacı azalan firma, çalışanların bir kısmını işten çıkararak üretim maliyetini düşürmüştür. Fakat firmanın uyguladığı bu yenilikler; çalışanların kişisel verilerinin korunması ve iş güvencesi gibi konularda tartışmaları beraberinde getirmiştir.

Yukarıdaki metne göre yapay zekâ destekli sistemlerin güvenlik, üretim, kişisel veri ve iş güvencesi açısından olumlu/olumsuz yönlerini gerekçeleriyle tabloya yazınız.

Olumlu Yönler	Gerekçe (Neden?)	Olumsuz Yönler	Gerekçe (Neden?)

2.1. YAPAY ZEKÂNIN ETİK BOYUTLARI

Düşün, Tartış, Paylaş

Yapay zekâ sistemleri günümüzde birçok alanda insanlara yardımcı olmakta ve önemli kararların alınmasına destek sağlamaktadır. Ancak bu sistemlerin kullanımı sırasında kişisel verilerin korunması, adaletin sağlanması ve insanların haklarının gözetilmesi gibi konular da önem kazanmaktadır. Bu nedenle yapay zekânın yalnızca ne yapabildiği değil nasıl kullanılması gerektiği de tartışılmaktadır.

Verilen metne göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Yapay zekâ sistemlerinin hangi kurallara uyması gerektiğini düşünüyorsunuz?**
2. **Yapay zekâ tarafından verilen kararların insanlar üzerindeki etkileri neler olabilir?**
3. **Yapay zekâ kullanılırken bireylerin haklarını korumak neden önemlidir?**

Yapay zekâ teknolojileri, sağladığı faydaların yanında bazı sorumlulukları da beraberinde getirir. Yapay zekâ sistemlerinin güvenilir, adil ve insan haklarına saygılı şekilde geliştirilmesi ve kullanılması büyük önem taşır. Bu doğrultuda yapay zekâ uygulamalarına rehberlik eden etik ilke ve kurallar ön plana çıkar.

Yapay zekâ teknolojileri günümüzde eğitimden ulaşıma, sağlıktan üretime kadar birçok alanda kullanılmaktadır. İş süreçlerini hızlandırarak verimliliği artırıp yaşamı pek çok açıdan kolaylaştırır. Ancak yapay zekâ

teknolojilerinin sunduğu olanaklar, beraberinde çeşitli sorumlulukları da getirir. Teknolojinin hızlı gelişimi, bireysel hakların ve toplumsal değerlerin korunmasını daha da önemli kılmıştır. Kişi ve kurumları bağlayıcı nitelikteki bu sorumluluklar yapay zekâ etiği kavramını ortaya çıkarmıştır.



Görsel 2.1: Yapay zekâ etiğinin temel ilke ve değerleri

Yapay zekâ etiği, yapay zekâ sistemlerinin geliştirilmesi ve kullanımı sürecinde insan haklarına ve toplumsal değerlere uygun hareket edilmesini sağlayan kuralları kapsar. Bu yaklaşım, teknolojinin güçlü şekilde kullanılmasının yanında doğru, güvenilir ve insan odaklı olmasını da mümkün kılar. Yapay zekâ sistemleri insanlığa rehberlik etmeli ancak insani değerlerin ve insan iradesinin yerini almamalıdır. Bu bağlamda yapay zekâ sistemlerinin etik şekilde geliştirilmesi ve kullanılabilmesi için bazı temel ilkelere uyulması gerekir. Bu ilkeler adalet, şeffaflık ve hesap verebilirlik çerçevesinde ele alınır (Görsel 2.1).

Adalet

Yapay zekâ etik ilkelerinin başında gelen adalet kavramı, bu alanda geliştirilen sistemlerin tüm bireylere eşit ve tarafsız şekilde davranmasını ifade eder. Yapay zekâ sistemleri, herhangi kişi ya da gruba din, dil, ırk, cinsiyet gibi unsurlar üzerinden haksız avantaj sağlamamalı ya da onları dezavantajlı duruma düşürmemelidir.

Yapay zekâ sistemleri genellikle eğitildikleri verileri işleyerek öğrenir ve bu doğrultuda çıkarımlarda bulunur. Kullanılan veri setinde ön yargı bulunursa sistem bu ön yargıları öğrenerek adil olmayan sonuçlar üretebilir. Bu durum, yapay zekânın algoritmik ön yargı (bias), ayrımcılık gibi etik sorunlar ortaya çıkarmasına neden olabilir. Veri setlerinden adil değerlendirme ve çıkarımlar yapılması için verilerin dikkatli şekilde seçilmesi ve denetlenmesi gerekir.

Şeffaflık

Yapay zekâ sistemlerindeki karar verme süreçlerinin anlaşılabilir ve açıklanabilir olmasını ifade eder. Bireyler, kendilerini veya yakın çevrelerini etkileyen kararların hangi bilgilere dayanarak verildiğini bilme hakkına sahiptir. Günümüzde yapay zekâ sistemleri çoğu zaman büyük veri setleri ve karmaşık algoritmalar kullanarak karar vermektedir. Bu nedenle bazı yapay zekâ sistemlerinin çalışma biçiminin anlaşılması zor olabilir. Bu durum, söz konusu sistemlerin kara kutu (black box) olarak nitelendirilmesine yol açar. Ancak etik yapay zekâ anlayışı, sistemlerin tamamen kapalı olmamasını, mümkün olduğunca açık ve anlaşılabilir şekilde sunulmasını önerir.

Yapay zekâ teknolojilerinde kullanılan sistemlerin şeffaf olması, bu teknolojiye duyulan güvenin artmasını sağlar. Kararların gerekçeleriyle birlikte açıklanması, kullanıcıların sistemi daha iyi anlamasına ve gerektiğinde kararları sorgulayabilmesine katkı sağlar. Ayrıca şeffaflık, sistemlerin denetlenmesini kolaylaştırarak etik kurallara uygunluğunun kontrol edilmesine yardımcı olur. Bu sayede teknoloji, belirsiz bir mekanizma olmaktan çıkarak hesap verebilir ve güvenilir yapıya kavuşur.

Hesap Verebilirlik

Hesap verebilirlik, yapay zekâ sistemlerinin işleyişinden veya verdiği kararlardan doğacak sonuçların sorumluluğunun net şekilde tanımlanmış olması durumudur. Kullanılan yapay zekâ sistemi hata yaptığında veya hak ihlaline neden olduğunda, bu durumun sorumluluğu makinelere yüklenmemelidir. Sistemi tasarlayan, geliştiren ve kullanıma sunan kişi, kurum veya kuruluşlar; sistemin etik kurallara ve yasalara uygun hareket etmesinden birinci derecede sorumludur.

Yapay zekâ sistemlerinin hesap verebilirlik ilkesi doğrultusunda, yanlış kararların derinlemesine araştırılması ve gerekli düzeltmelerin yapılmasına olanak sağlanmalıdır. Bu süreçte itiraz hakkı önemlidir. Kişi veya kuruluşlar, kendilerini olumsuz etkileyen yapay zekâ kararına karşı itiraz edebilmelidir. Kararın tekrar gözden geçirilmesini veya gerçek kişi tarafından değerlendirilmesini talep edebilmelidir. Bu sürecin işleyebilmesi için yapay zekâ geri bildirim mekanizmaları işletilmelidir. Böylelikle yapay zekâ teknolojileri kontrolsüz güç olmaktan çıkarak daha güvenilir, denetlenebilir ve insan onuruna saygılı yapıya kavuşacaktır.

BİLGİ KUTUSU

Yapay zekâ modellerinin, eğitim verilerindeki örüntüleri yanlış ilişkilendirerek gerçek dışı, mantıksız veya hatalı bilgileri kesin bir dille üretmesine **yapay zekâda halüsinasyon** denir. Üretken yapay zekânın doğasındaki olasılıksal yapıdan kaynaklanan bu durum, bilgi kirliliği ve yanlış yönlendirme riski taşır. Bu riskin yönetilmesi için çıktıların mutlaka insan denetiminden (doğrulama) geçirilmesi, sistemlerin şeffaflık ve hesap verebilirlik ilkeleri çerçevesinde kullanılması önemli bir zorunluluktur.

1. SIRA SİZDE

Aşağıda verilen durumu dikkatlice okuyunuz. Yapay zekâ sistemlerinin yol açabileceği etik sorunları belirleyiniz. Verilen anahtar kavramları kullanarak duruma uygun etik sorunları tabloya yerleştiriniz ve bu sorunların nedenlerini açıklayınız.

Durum

Bir şirket, çalışan performansını değerlendirmek için yapay zekâ destekli bir sistem kullanmaktadır. Sistem, çalışanların verilerini analiz ederek onlara performans puanı vermektedir. Bir süre sonra, sistemin belirli yaş grubundaki çalışanlara daha düşük puan verdiği fark edilir. Çalışanlar, puanların hangi ölçütlere göre verildiğini öğrenmek ister. Şirket yönetimiyse sistemin kararlarını tam olarak açıklayamaz.

Kavramlar

Algoritmik ön yargı (bias), Ayrımcılık, Şeffaflık eksikliği

Etik Sorun	Gerekçe (Neden?)

2.1.1. Büyük Veri ve Veri Gizliliği

Yapay zekâ sistemleri, doğru ve hızlı kararlar verebilmek için büyük veri setlerine ihtiyaç duyar. Büyük veri (big data); farklı kaynaklardan elde edilen ve hızla üretilen büyük hacimli veri topluluklarını ifade eder (Görsel 2.2). Konum bilgisi, alışveriş tercihleri ve kişisel bilgiler büyük veri kümelerinin bir parçası olabilir. Yapay zekâ sistemleri büyük veri kümelerini kullanarak etkili sonuçlar üretir. Bu süreçte verilerin toplanma, saklanma ve işlenme aşamaları kişilerin özel yaşamları açısından risk barındırır.



Görsel 2.2: Yapay zekâ sistemlerinde kullanılan büyük veri kaynakları

Kişi ve kurumlar açısından önem taşıyan veri gizliliği; verilerin yalnızca belirlenen amaçlar doğrultusunda toplanması, işlenmesi, saklanması ve yetkisiz erişime karşı korunmasını ifade eder. Veriler üzerindeki hakların gözetilmesini esas alır. Günümüzde internet ve sosyal medya alanlarında yaşanan hızlı gelişim ve değişimden dolayı kişisel verilere erişim kolay hâle gelmiştir. Veri gizliliği açısından bu sürecin sağlıklı yürütülememesi, verileri paylaşma konusunda bireylerin endişelerinin artmasına ve bunun sonucunda kendilerini güvensiz hissetmelerine neden olmuştur.

Yapay zekâ sistemlerinde büyük verilerin kullanımında dikkat edilmesi gereken bazı temel risk faktörleri vardır. Bu faktörlerden önemlileri şunlardır:

Sınıf Dengesizliği: Yapay zekânın eğitimi sırasında kullanılan verilerde dengesizlik bulunmasıdır. Bu durum, bazı sınıfları yeterince temsil edememesi veya bu sınıflarda düşük performans göstermesi, ileride bireyler hakkında hatalı ve adaletsiz kararlar vermesine neden olabilir.

Güvenlik Açığı: Yapay zekâ sistemlerindeki zayıf noktalar siber saldırılara zemin hazırlar. Örneğin bir görüntü tanıma sistemine yapılan müdahaleyle verilerin değiştirilmesi, sistemin yanlış sonuçlar üretmesine neden olabilir. Ayrıca kişisel verilerin bulunduğu veri tabanlarındaki açıklar, verilerin çalınması veya kötüye kullanılması riskini artırır.

Siyah Kuğu (Black Swan): Geçmiş verilere bakılarak tahmin edilemeyen, gerçekleştiğindeyse büyük etkiler yaratan beklenmedik durumlardır. Yapay zekâ sistemleri bu tür olaylarla karşılaştığında hatalı sonuçlar üretebilir.

Değerlendirme Ölçütleri: Yapay zekâ sistemlerinin güvenirliği, ürettiği sonuçların kalitesine bağlıdır. Modellerin aynı veri üzerinde çalışırken tutarsız sonuçlar üretmesi sistemin güvenirliğini zedeler. Bir sistemden beklenen iki temel özellik vardır. Bu özellikler şunlardır:

Tekrarlanabilirlik: Aynı veri, aynı yöntem ve benzer koşullar kullanıldığında benzer sonuçlara ulaşılabilmesidir.

Tutarlılık: Sistemin benzer durumlarda çelişkili olmayan sonuçlar üretmesidir. Başka bir ifadeyle model, sonucu önceden bilinen yani etiketlenmiş örneklerden yararlanarak yeni veriler için tahminde bulunur.

2.1.2. Yapay Zekâ ve İş Gücü Piyasası

Yapay zekâ teknolojilerinin hızla gelişmesi, çeşitli iş sektörlerinde de büyük değişim ve gelişim sağlamıştır. Bu durum ekonomik ve sosyal hayatı etkileyen, iş alışkanlıklarını değiştiren kapsamlı bir dönüşüm hâline gelmiştir. Yapay zekâ sistemleri, birçok sektörde yapılan işleri daha hızlı ve çoğunlukla daha doğru gerçekleştirdiği için işlerin yapılma biçimlerini büyük oranda değiştirmiştir. Özellikle belirli kurallara dayalı, rutin ve sürekli tekrar eden görevlerin yapay zekâ tarafından üstlenilmesi, bazı alanlarda insan gücü ve bilgisine olan ihtiyacı azaltmıştır. Bununla birlikte yapay zekâ yeni iş alanları ve mesleklerin ortaya çıkmasına da imkân sağlamıştır. Yapay zekâ mühendisliği, veri analizi uzmanlığı, prompt mühendisliği; bu mesleklerden birkaçıdır. Görsel 2.3'te yapay zekânın iş gücü piyasasına etkileri ve farklı sektörlerdeki bazı kullanım alanları örneklenmiştir.



Görsel 2.3: Yapay zekâ ve iş gücü piyasası

Yapay zekânın iş gücü piyasası üzerindeki etkileri sonucunda çalışanların sahip olması gereken beceriler de farklılaşmaktadır. Çalışanların, geçmiş eğitim süreçlerinde edinmiş oldukları bilgiler yeterli olmamakta, teknolojinin hızlı gelişimine ayak uydurabilmeleri için bilgilerini sürekli güncellemeleri gerekmektedir. Bu süreç çalışanların mevcut bilgi ve yeteneklerini günümüz teknoloji dünyasına uygun hâle getirmelerini, başka bir deyişle beceri dönüşümünü zorunlu kılmaktadır. Özellikle yapay zekâ teknolojilerinin henüz yeterli seviyeye ulaşamadığı analitik düşünme, yaratıcı problem çözme ve insani duygularla karar verme yetisi her zamankinden daha değerli hâle gelmiştir.

Yapay zekâ uygulamalarındaki gelişimin neden olabileceği olumsuz durumlardan biri de işe giriş süreçlerinde bazı eşitsizliklerin oluşmasıdır. Teknolojik gelişmelere yeterince ayak uyduramayan veya eğitim imkânlarına erişemeyen bireyler iş bulma süreçlerinde zorlanacaktır. Bu durum gelir farklılıklarını da ortaya çıkaracaktır.

Yapay zekânın doğru ve bilinçli kullanımı iş verimliliğini artırarak toplum için yeni refah fırsatları sunma potansiyeline sahiptir.

Yapay zekâ, iş dünyasında hem risk hem de fırsatlar barındıran büyük bir dönüşüm sürecidir. Bu yeni döneme uyum sağlanabilmesi için bireylerin kendilerini geliştirmesi, eğitim sisteminin güncel ihtiyaçlara göre yenilenmesi ve teknolojinin insani değerleri koruyacak şekilde bilinçli kullanılması büyük önem taşımaktadır.

BİLGİ KUTUSU

World Economic Forum tarafından 2025 yılında yayımlanan Future of Jobs Report'ta (Fyu-çır ov Cobs Riport), 2030 yılına kadar günümüzdeki toplam istihdamın yaklaşık %8'ine karşılık gelen 92 milyon işin ortadan kalkacağı öngörülmektedir. Rapora göre belirli işlerin ortadan kalkma nedeni, iş gücü piyasasında yaşanacak yapısal dönüşümlerdir.

2. SIRA SİZDE

1. Verilen örnek durumu inceleyerek yapay zekânın iş dünyasında oluşturduğu değişimleri tabloya yazınız.

Durum

Bir lojistik şirketi, depo yönetimini daha verimli hâle getirmek amacıyla yapay zekâ destekli robot sistemleri kullanmaya başlamıştır. Bu sistemler ürünleri otomatik olarak tanıyıp sınıflandırmakta ve sevkiyata hazırlamaktadır. Yeni sistem sayesinde işler daha hızlı ve hatasız yürütülmüş, şirketin maliyetleri azalmıştır. Ancak bu değişim bazı çalışanları doğrudan etkilemiştir. Depoda uzun süredir çalışan bazı işçilerin görevleri ortadan kalkmış, bu işçiler işten çıkarılma riskiyle karşı karşıya kalmıştır. Şirket yönetimi, çalışanların bir kısmına robot sistemlerini kontrol edebilecekleri yeni görevler önermiştir. Fakat bu görevler için teknik bilgi ve eğitim gerekmektedir. Bazı çalışanlar bu eğitimi alabileceklerini düşünürken bazılarıysa sürece uyum sağlamakta zorlanmıştır. Bu durum, çalışanlar arasında endişe ve belirsizlik yaratırken şirket içinde de "Verimlilik mi, istihdam mı?" sorusunun tartışılmasına neden olmuştur.

Yapay Zekânın İş Gücüne Etkileri	Örnekten Açıklama
İş kaybı (azalan meslekler)	
Yeni iş alanları	
Beceri değişimi (beceri dönüşümü)	

2. Aşağıdaki soruları arkadaşlarınızla tartışınız ve görüşlerinizi sınıfta paylaşınız.

- Bu durumda yapay zekânın olumlu ve olumsuz etkileri neler olabilir?
- Siz şirket yöneticisi olsaydınız çalışanlar için nasıl bir çözüm üretirdiniz? Görüşlerinizi açıklayınız.
- Sizce gelecekte hangi beceriler daha önemli hâle gelecektir? Nedenini açıklayınız.

2.1.3. Yapay Zekâ Entegrasyonu ve Verimlilik

Yapay zekâ teknolojilerinin sektörel alanda en önemli etkilerinin başında, iş süreçlerini hızlandırması ve insan hatalarını azaltarak verimliliği artırması gelir. Yapay zekâ, büyük veri setlerini kısa sürede analiz edebilmesi ve tekrar eden görevleri otomatik olarak gerçekleştirebilmesi sayesinde işletmelerde daha az kaynakla daha fazla üretim yapılmasına olanak sağlar. Yapay zekâ entegrasyonu, doğru kullanıldığında çalışanların daha nitelikli işlere yönelmesini sağlar ve işletmelerin rekabet gücünü artırır. Yapay zekâ entegrasyonun çeşitli sektörlerde etkili ve verimli kullanılmasına ilişkin örnekler Görsel 2.4'te verilmiştir.



Görsel 2.4: Yapay zekâ entegrasyonunun iş yerinde verimlilik ve üretkenliğe katkıları

2.1. ETKİNLİK

Adı	Yapay Zekâ Etiği ve Veri Gizliliği Sunumu
Süre	1 ders saati
Amaç	Yapay zekâ uygulamalarında etik ilkeleri ve veri gizliliği risklerini açıklamak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, internet erişimi, sunum hazırlama yazılımları, güvenilir bilgi kaynakları
Değerlendirme	Yapmış olduğunuz çalışmalar öğretmeniniz tarafından karekodda yer alan kontrol listesi kullanılarak değerlendirilecektir. Çalışmalarınızı bu ölçütlere göre planlayınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendirebilirsiniz.

Hazırlık

- Bilgisayarınızı ve kullanacağınız sunum programını açınız.
- Yapay zekâ etiği ve veri gizliliği konularında ön bilgi edininiz.
- Çalışma ortamınızın düzenli ve hazır olduğundan emin olunuz.
- İş sağlığı ve güvenliği kurallarına uygun şekilde çalışma ortamınızı düzenleyiniz.
- Sunum dosyanızı oluşturunuz ve uygun bir isimle kaydediniz.

Etkinlik Sırasında Uyulması Gereken Kurallar

- Araştırmalarınızda güvenilir ve güncel kaynaklardan yararlanarak bilgilerin doğruluğunu kontrol ediniz.
- Dijital araçları sorumlu bir şekilde kullanırken kişisel verilerin korunmasına dikkat ediniz.
- Çalışmalarınızı planlanan süre içinde tamamlamaya özen gösteriniz.
- Sunumlarınızda görsel ve metin dengesini koruyarak açık ve etkili bir dil kullanınız.

Çalışma Yönergesi

1. Yapay zekâda etik ilkeler ve veri gizliliği konularını planlayarak çalışma takvimi oluşturunuz.
2. Etik ilkeler ve veri gizliliği hakkında en az üç güvenilir kaynaktan bilgi toplayıp karşılaştırınız.
3. Yapay zekâ uygulamalarındaki veri gizliliği risklerini belirleyerek neden sonuç ilişkisiyle analiz ediniz.
4. Belirlediğiniz sorunlara yönelik uygulanabilir çözüm önerileri geliştiriniz ve günlük yaşama katkılarını açıklayınız.
5. Sunumunuzu kısa, anlaşılır metinler ile görseller kullanarak hazırlayınız.
6. Çalışmanızı süresinde tamamlayıp sunumunuzu açık, anlaşılır ve etkili biçimde gerçekleştiriniz.

Kontrol listesi ve öz değerlendirme formuna ulaşmak için karekodu okutunuz.
Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

2.2. YAPAY ZEKÂNIN YASAL BOYUTLARI

Düşün, Tartış, Paylaş

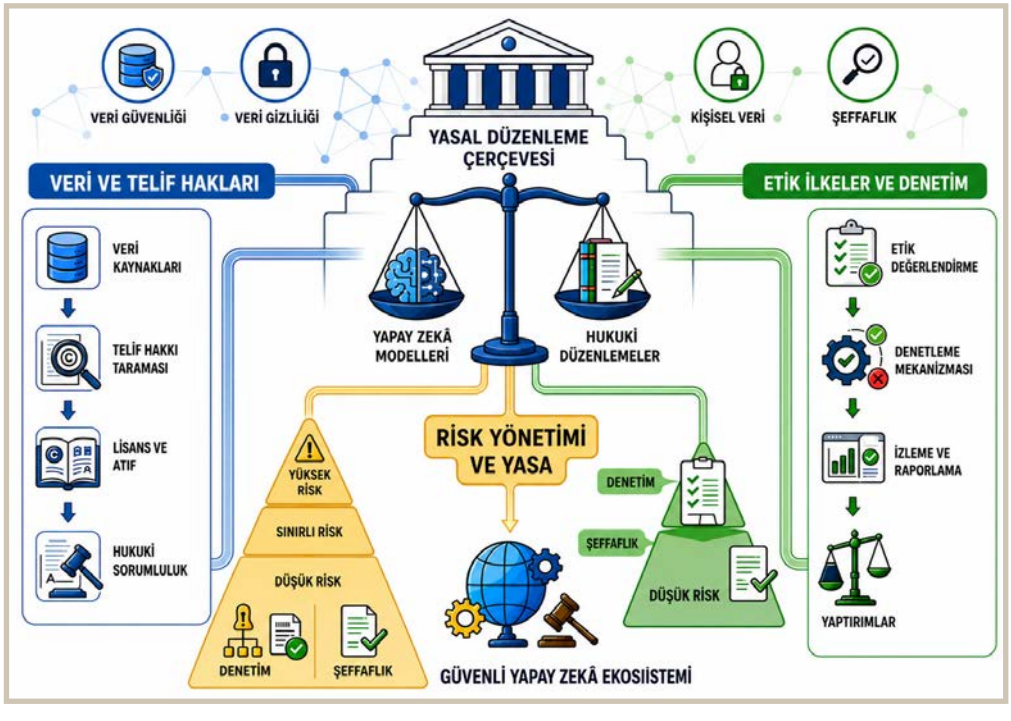
Günümüzde yapay zekâ sistemleri birçok alanda insanlar adına karar verebilmekte ve çeşitli görevleri yerine getirebilmektedir. Teknolojinin kullanım alanları genişledikçe bu sistemlerin nasıl kullanılacağı, hangi kurallara tabi olacağı ve ortaya çıkabilecek sorunlardan kimin sorumlu olacağı gibi sorular da önem kazanmıştır.

Verilen metne göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Yapay zekâ sistemlerinin kullanımıyla ilgili kuralların belirlenmesi neden gereklidir?**
2. **Yapay zekâ sisteminin verdiği yanlış karar sonucunda ortaya çıkan zarardan sizce kim sorumlu tutulmalıdır?**
3. **Teknolojik gelişmelerin hızla değişmesi mevcut yasal düzenlemeleri nasıl etkileyebilir?**

Yapay zekâ teknolojilerinin yaygınlaşmasının yalnızca teknik değil aynı zamanda hukuki sonuçlar da doğurduğu görülmektedir. Yapay zekâ sistemlerinin geliştirilmesi ve kullanılması sürecinde kişi haklarının korunması, sorumlulukların belirlenmesi ve toplumsal güvenliğin sağlanması büyük önem taşır. Bu nedenle yapay zekâ alanında çeşitli yasal düzenlemelere ihtiyaç duyulmaktadır.

21. yüzyılda teknolojinin ve teknolojik alışkanlıkların çok hızlı değişmesi, hukuk sistemlerinin bu değişime aynı hızda uyum sağlamasını zorlaştırmıştır. Özellikle yapay zekâ alanında yaşanan hızlı gelişmeler, kişisel verilerin paylaşımı ve kullanımının giderek artması, hukuki boşluklara yol açmıştır. Bu durum, bireysel haklardan toplumsal güvenliğe kadar birçok alanda riskler doğurmaktadır. Görsel 2.5'te yapay zekâ ekosisteminde etik denetim, veri güvenliği, şeffaflık, hukuki sorumluluk ve risk yönetimi arasındaki ilişki gösterilmektedir.



Görsel 2.5: Yapay zekâ sistemlerinde etik ilkeler, yasal düzenlemeler ve risk yönetimi

Yapay zekâ sistemlerinin birçok alanda otonom kararlar alması, bu sistemlerin hukuki durumunu ve sorumluluk sınırlarını tartışmalı hâle getirmektedir. Son zamanlarda sıkça tartışma konusu olan, yapay zekâ sistemleri tarafından verilen kararların sonuçlarından kimin sorumlu tutulacağı sorusuysa net olarak cevaplanabilmiş değildir. Kimi hukuk çevreleri, sorumluluğu uygulamayı geliştiren kişi ya da kuruma yüklerken kimi hukukçular, kullanan kişi ve kurumları sorumlu tutmaktadır. Bu durum, mevcut hukuk sistemlerinin güncel teknolojilere uyum sağlama konusunda yetersiz kalabildiğini göstermektedir.

3. SIRA SİZDE

Bir yapay zekâ sisteminin hata yaptığı durumlarda yazılım geliştirici, kullanıcı ve kurumların rolünü aşağıdaki tabloyu doldurarak değerlendiriniz.

Sorumlu Olabilecek Taraf	Sorumlu Olma Nedeni
Yazılım geliştirici	
Sistemi kullanan kişi	
Sistemi sunan kurum	
Diğer (Belirtiniz.)	

2.2.1. Teknolojik Gelişmelerin Doğurduğu Yasal Boşluklar

Yapay zekâ ve dijital teknolojilerin gelişmesiyle birlikte bazı alanlarda önemli hukuki sorunlar ortaya çıkmıştır. Bu sorunlar şu başlıklarla incelenebilir:

Yapay Zekâ Sorumluluğu: Yapay zekâ sistemlerinin otonom kararlar aldığı durumlarda hukuki sorumluluğun kime ait olduğu, güncel sorunların başında gelmektedir. Bu bağlamda, yapay zekâ sistemlerinin hukuki olarak hak ehliyetine sahip olup olmadığı konusu tartışılmaktadır. Bu nedenle sorumluluk sınırlarını belirleyen hukuki düzenlemelere ihtiyaç duyulmaktadır.

Kişisel Verilerin İşlenmesi: Büyük verilerin kullanımının artmasıyla birlikte, çeşitli uygulamalar ve yapay zekâ sistemleri tarafından verilerin toplanması, saklanması ve işlenmesi süreçleri hız kazanmıştır. Ancak bu süreçlerin arka planında gerçekleştirilen işlemler hakkında yeterli bilginin sunulmaması ve açık rızanın her zaman sağlanamaması, mahremiyet ihlallerine yol açar. Bu durum, bireylerin özel hayatının ihlalini ve verilerin ticari bir araç hâline gelmesini engellemeye yönelik güçlü yasal düzenlemeleri gerekli kılmaktadır.

Dijital Vatandaşlık ve İçerik Paylaşımı: Sosyal medya ve dijital platformlarda yanlış bilgilerin hızlı ve denetimsiz şekilde yayılması, siber zorbalık, etik olmayan paylaşımlar gibi durumlar; mevcut yasal düzenlemelerin yetersiz kaldığı alanlar arasında yer almaktadır.

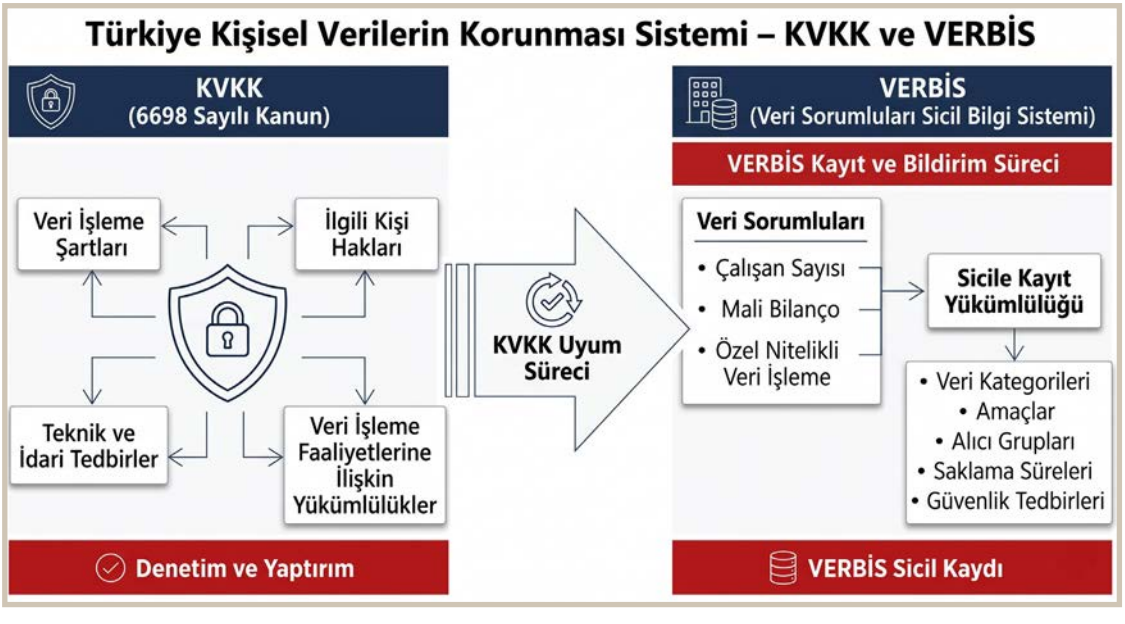
Akıllı Sözleşmeler: Blok zincir teknolojileriyle ortaya çıkan akıllı sözleşmelerin hukuki niteliği ve tarafları bağlayıcılığı konusunda ulusal ve uluslararası düzeyde tam bir görüş birliği yoktur. Bu nedenle yeni teknolojilere uygun, hukuki çerçevelerin oluşturulması bir zorunluluk hâline gelmiştir.

2.2.2. Veri Güvenliği ve Dijital Teknolojilere İlişkin Yasal Düzenlemeler

Veri yönetim sistemleri ve bilgi iletişim teknolojilerinin hızla gelişip yaygınlaşması; veri güvenliği, kişisel mahremiyet ve dijital ortamlarda etik ilkelerin korunması konularını daha önemli hâle getirmiştir. Bu gelişmeler doğrultusunda birçok ülkede, kişisel verilerin güvenli şekilde işlenmesini ve bireylerin dijital ortamlarda korunmasını amaçlayan yasal düzenlemeler yapılmıştır. Türkiye'de bu amaçla 7 Nisan 2016 tarihinde yürürlüğe giren Kişisel Verilerin Korunması Kanunu (KVKK), kişisel verilerin güvence altına alınmasına yönelik önemli bir adımdır. Benzer şekilde Avrupa Birliği tarafından 25 Mayıs 2018 tarihinde yürürlüğe giren General Data Protection Regulation [(GDPR), (Cenırıl Deytı Protekşın Regyuleyşın)], dijital çağın ihtiyaçlarına uygun biçimde veri güvenliğini sağlamayı amaçlayan ve bu alandaki standardı yeniden belirleyen kapsamlı bir yasal çerçeve sunmaktadır.

KVKK ile GDPR yasal metinleri detaylı şekilde incelendiğinde, yapı olarak birbirine benzer yönlerinin olduğu görülür. Yasal metinlerde geçen kişisel veri kavramı; kimlikleri doğrudan ya da dolaylı olarak belirlenebilen gerçek bireylere ait kimlik, konum, sağlık bilgileri gibi her türlü veriyi kapsamaktadır. İki yasal düzenleme de bireylerin açık rızaları alınarak toplanan verilerin belirli amaçlar doğrultusunda işlenmesini ve veri güvenliğinin sağlanmasını amaçlar. Ayrıca bireylere kendi verilerine erişim, verileri düzenleme ve istenmediğinde silme gibi haklar tanınmıştır. Bu sayede bireylerin kendi verileri üzerinde kontrol sahibi olmaları sağlanmıştır. Ancak her iki yasal metin arasında yapı, etki alanı ve yaptırımlar bakımından çeşitli farklılıklar bulunur. KVKK, ülke içerisinde veri işleyen tüm gerçek ve tüzel kişileri, GDPR Avrupa Birliği vatandaşlarına hizmet sunan tüm kişi, kurum ve kuruluşları kapsar. Kişisel veri ihlalleri durumunda uygulanacak maddi yaptırımlar da farklılık gösterir. GDPR kapsamında, ihlalin derecesine bağlı olarak ihlalde bulunan kurumlara yıllık kazançlarının %4'üne kadar para cezası verilebilir. KVKK kapsamındaysa bu yaptırımlar her yıl yeniden düzenlenen sabit tutarlar üzerinden uygulanmaktadır.

Kişisel veri ihlallerinin önüne geçilmesi amacıyla kurumlar tarafından alınan teknik güvenlik önlemleri tek başına yeterli görülmemektedir. Dünya çapında veri güvenliğinin sağlanması amacıyla kurumların ISO 27001 gibi bilgi güvenliği ve yönetim sistemleri sertifikasyon şartlarını yerine getirmeleri beklenmektedir. Ülkemizdeyse KVKK tarafından hizmete sunulan Veri Sorumluları Sicil Bilgi Sistemi (VERBİS) üzerinden, kurum ve kuruluşların işledikleri kişisel verilerin korunmasını sağlayan resmî kayıt sistemi uygulamaya konmuştur. Şema 2.1'de, KVKK ile VERBİS arasındaki ilişki, veri sorumlularının yükümlülükleri ve kayıt süreçleri özetlenmiştir.



Şema 2.1: KVKK ile VERBİS arasındaki ilişki

2.2.3. Yapay Zekâ Alanında Yapılan Yasal Düzenlemeler

Yapay zekâ sistemlerinin her alanda giderek daha etkin olmasıyla birlikte insan hakları, veri güvenliği, şeffaflık ve hesap verebilirlik kavramlarının önemi artmıştır. Birçok ülke ve uluslararası kuruluş, yapay zekâ teknolojilerinin denetlenmesi ve belirli kurallar çerçevesinde geliştirilmesini sağlamak amacıyla yeni yasal düzenlemeler yapmıştır.

Yapay zekâ alanında yapılan çalışmalardan ilk önemli belge 2021 yılında UNESCO tarafından yayımlanan Yapay Zekâ Etiği Tavsiye Kararı'dır. Tavsiye kararlarının yer aldığı bu belgede insan haklarının korunması, etnik ayrımcılığın önlenmesi, çevresel sürdürülebilirlik, şeffaflık gibi ilkeler vurgulanmıştır. Yapay zekâ alanındaki çalışmaların insanlık onuruna yakışır olması ve yapay zekânın insanlık yararına kullanılması ilkeleri benimsenmiştir.

Uluslararası düzeyde yapılan önemli yasal düzenlemelerin başında, 2024 yılında yürürlüğe giren Avrupa Birliği Yapay Zekâ Yasası (EU AI Act) gelir. Bu yasal düzenleme; eğitim, sağlık, güvenlik, hukuk gibi alanlarda kullanılan yapay zekâ uygulamaları için sıkı kurallar getirmiştir. Yasanın hedefi; yapay zekâ sistemlerinin şeffaf ve hesap verebilir olması, kişi hak ve hürriyetlerini korumasıdır.

Türkiye'de yapay zekâ alanında yapılan çalışmaların temelini T.C. Cumhurbaşkanlığı Dijital Dönüşüm Ofisiyle T.C. Sanayi ve Teknoloji Bakanlığı koordinasyonunda hazırlanan Ulusal Yapay Zekâ Stratejisi (2021-2025) oluşturur. Strateji belgesinde; yapay zekâ alanının geliştirilmesi, veri yönetim süreçlerinin güçlendirilmesi ve bu alanda etik ilkelerin oluşturulması hedeflenmiştir. Yapay zekânın veriye dayalı karar alma süreçlerinde etkili şekilde kullanılması ve güvenli, sürdürülebilir bir yapay zekâ ekosisteminin oluşturulması planlanmıştır.

TÜBİTAK tarafından 2026 yılında oluşturulan Yapay Zekâ Politikası, yapay zekânın kurumsal süreçlerde etkin şekilde kullanılması için stratejik çerçeve sunmaktadır. Oluşturulan yasal politika belgesinde; yapay zekânın veriye dayalı karar alma süreçlerinde etkin güç hâline geldiği, operasyonel verimliği artıran bir araç olduğu ve veri yönetim anlayışının geliştirilmesi gerektiği vurgulanmıştır. Ayrıca yapay zekâ kullanımının yalnızca bir teknik değil aynı zamanda bir dönüşüm belgesi olduğu belirtilmiştir.

Millî Eğitim Bakanlığı tarafından 2026 yılında hazırlanan Yapay Zekâ Uygulamaları Etik Rehberi (YAZEK), yapay zekâ teknolojilerinin eğitim süreçlerinde kullanımına dair standartları belirleyen ve bilgi güvenliğini denetleyen bir sistemdir. Hazırlanan rehberde yapay zekânın öğrenme süreçlerini destekleyen yardımcı araç olarak kullanılmasının gerekliliği vurgulanmıştır. Eğitim sürecindeki yapay zekâ kullanımında; öğrenci ve öğretmenlerin veri güvenliği, eşitlik, şeffaflık, kapsayıcılık, insan denetimi gibi ilkelerin ön plana çıkarılmasının gerekliliği belirtilmiştir.

NOTLAR

2.2. ETKİNLİK

Adı	Yapay Zekâ ve Dijital Hukuk Dosyası
Süre	1 ders saati
Amaç	Yapay zekâ alanındaki ulusal ve uluslararası düzenlemeleri araştırarak bu düzenlemelerin amaçlarını ve kapsamlarını analiz etmek, etik ilkeler ve veri güvenliği uygulamalarıyla ilişkilerini değerlendirerek elde edilen bilgileri rapor hâlinde sunmak
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, internet erişimi, güvenilir bilgi kaynakları, kelime işlemci programları, not alma araçları
Değerlendirme	Yapmış olduğunuz çalışmalar öğretmeniniz tarafından karekodda yer alan kontrol listesi kullanılarak değerlendirilecektir. Çalışmalarınızı bu ölçütlere göre planlayınız.

Hazırlık

- Bilgisayarınızı ve kullanacağınız yazılımı açınız.
- Yapay zekâ, veri güvenliği ve dijital hukuk kavramları hakkında ön bilgi edininiz.
- Çalışma ortamınızın düzenli ve hazır olduğundan emin olunuz.
- İş sağlığı ve güvenliği kurallarına uygun şekilde çalışma ortamınızı düzenleyiniz.
- Rapor dosyanızı oluşturunuz ve uygun bir isimle kaydediniz.

Etkinlik Sırasında Uyulması Gereken Kurallar

- Güvenilir kaynaklardan yararlanarak bilgilerin doğruluğunu kontrol ediniz.
- Dijital araçları sorumlu kullanarak kişisel verileri koruyunuz.
- Çalışmalarınızı zamanında tamamlayıp bilgileri düzenli sununuz.
- Akademik dürüstlük kurallarına uyunuz.

Çalışma Yönergesi

1. Yapay zekâ düzenlemelerini veri güvenliği, etik ilkeler, insan hakları ve denetim başlıkları altında incelemek için çalışma planı oluşturunuz.
2. AI Act, UNESCO, KVKK, GDPR, Ulusal Yapay Zekâ Stratejisi ve YAZEK gibi güvenilir kaynaklardan bilgi toplayıp karşılaştırınız.
3. Düzenlemelerin veri güvenliği, şeffaflık, insan denetimi ve hesap verebilirlik açısından hangi riskleri azalttığını analiz ediniz.
4. İncelediğiniz düzenlemeleri karşılaştırarak odaklandıkları alanları ve bireyleri korumayı hedefledikleri riskleri açıklayınız.
5. Raporunuzu kısa ve anlaşılır biçimde; tablo, şema veya görsellerle destekleyerek hazırlayınız.
6. Çalışmanızı belirlenen süre içerisinde tamamlayınız.

Kontrol listesine ulaşmak için karekodu okutunuz.

Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki veri türlerinden büyük veri kapsamında değerlendirilebilecek olanların başına "✓", değerlendirilemeyecek olanların başına "X" işareti koyunuz.

( ) Öğrencinin tek sınav notu
( ) Alışveriş sitesindeki milyonlarca müşteri işlemi
( ) Sosyal medya paylaşımları
( ) Akıllı saatlerden elde edilen sağlık verileri
( ) Trafik sensörlerinden elde edilen anlık veriler

B) Bir lojistik şirketi, depo süreçlerinin bir bölümünde yapay zekâ destekli robot sistemleri kullanmaya başlamıştır. Bu değişiklik sayesinde bazı işler daha hızlı ve hatasız yapılırken bazı çalışanların görevlerinde değişiklikler meydana gelmiştir. **Metne göre aşağıdaki soruları cevaplayınız.**

1. Yapay zekâ destekli robot sisteminin şirkete sağlayabileceği iki avantajı yazınız.
2. Yapay zekâ destekli robot sisteminin çalışanlar açısından ortaya çıkarabileceği iki olası sonucu açıklayınız.
3. Çalışanların yeni çalışma düzenine uyum sağlayabilmeleri için şirket yönetiminin alabileceği önlemleri belirtiniz.

C) Aşağıdaki soruların doğru cevabını işaretleyiniz.

1. Yapay zekâ teknolojilerinin günlük yaşam ve iş yaşamı üzerindeki etkileri düşünüldüğünde, aşağıdakilerden hangisi olumsuz bir durum olarak değerlendirilebilir?

A) Akıllı asistanların yaşamı kolaylaştırması
B) Üretim süreçlerinde verimliliğin artması
C) Yeni iş alanlarının ortaya çıkması
D) Bazı mesleklerin ortadan kalkması
E) Öneri sistemlerinin gelişmesi

2. Yapay zekâ etiği kavramı aşağıdakilerden hangisini ifade eder?

A) Yapay zekâ sistemlerinin daha hızlı ve verimli çalışmasını sağlayan teknik yöntem ve uygulamalar
B) Yapay zekâ uygulamalarının ekonomik başarı ve ticari kazanç ölçütlerine dayalı değerlendirme yaklaşımları
C) Yapay zekâ sistemlerinin insan haklarına ve toplumsal değerlere uygun geliştirilmesini ve

kullanılmasını sağlayan ilke ve kurallar

D) Yapay zekâ sistemlerinde kararların insan müdahalesi olmadan alınmasını savunan yaklaşımlar

E) Yapay zekâ yazılımlarında ortaya çıkan teknik sorunların belirlenmesi ve giderilmesine yönelik çalışmalar

3. Yapay zekâ etiği kavramı aşağıdakilerden hangisini ifade eder?

A) Kararların mümkün olduğunca hızlı verilmesi

B) Tüm bireylere eşit ve tarafsız davranılması

C) Verilerin sınırsız biçimde paylaşılması

D) İnsan müdahalesinin tamamen sona ermesini öngörür.

E) Yapay zekânın yalnızca ekonomik verime göre değerlendirilmesi

4. Belirli yaş grubundaki çalışanlara sistematik olarak daha düşük puan veren bir performans değerlendirme sistemi, aşağıdaki etik sorunlardan hangisiyle doğrudan ilişkilendirilir?

A) Şeffaflığın artmasını sağlamaktadır.

B) Algoritmik ön yargı ve ayrımcılık sorununu ortaya çıkarmaktadır.

C) İnsan denetiminin güçlenmesine katkıda bulunmaktadır.

D) Verimlilik artışını desteklemektedir.

E) İş bölümü çeşitliliğini artırmaktadır.

5. "Kara kutu" olarak adlandırılan yapay zekâ sistemleri, öncelikle aşağıdaki etik ilkelerden hangisinin uygulanmasını zorlaştırır?

A) Adalet

B) Şeffaflık

C) Mahremiyet

D) Tarafsızlık

E) Hesap verebilirlik

6. İnternet ve sosyal medya alanındaki hızlı gelişim, veri gizliliği bakımından aşağıdaki sonuçlardan hangisine yol açmıştır?

A) Kişisel verilere erişimi tümüyle ortadan kaldırmıştır.

B) Veri paylaşımına ilişkin endişeleri azaltmıştır.

C) Kişisel verilere erişimi kolaylaştırmış ve güvensizlik duygusunun artmasına neden olmuştur.

D) Veri koruma tartışmalarını gereksiz hâle getirmiştir.

E) Açık rıza kavramını tamamen işlevsiz duruma getirmiştir.

7. Yapay zekâ nedeniyle ortaya çıkan iş kaybı riski, hangi iş türlerinde daha belirgin hâle gelmektedir?

A) Rutin, tekrarlayan ve kurallara dayalı görevlerde
B) Tamamen rastlantısal ve kuralsız işlerde
C) Yüksek yaratıcılık gerektiren sanat işlerinde
D) Spor antrenörlüğü alanında
E) Tarımsal el işçiliğinde

8. Yapay zekâ alanında hukuki boşlukların ortaya çıkmasının temel nedeni aşağıdakilerden hangisidir?

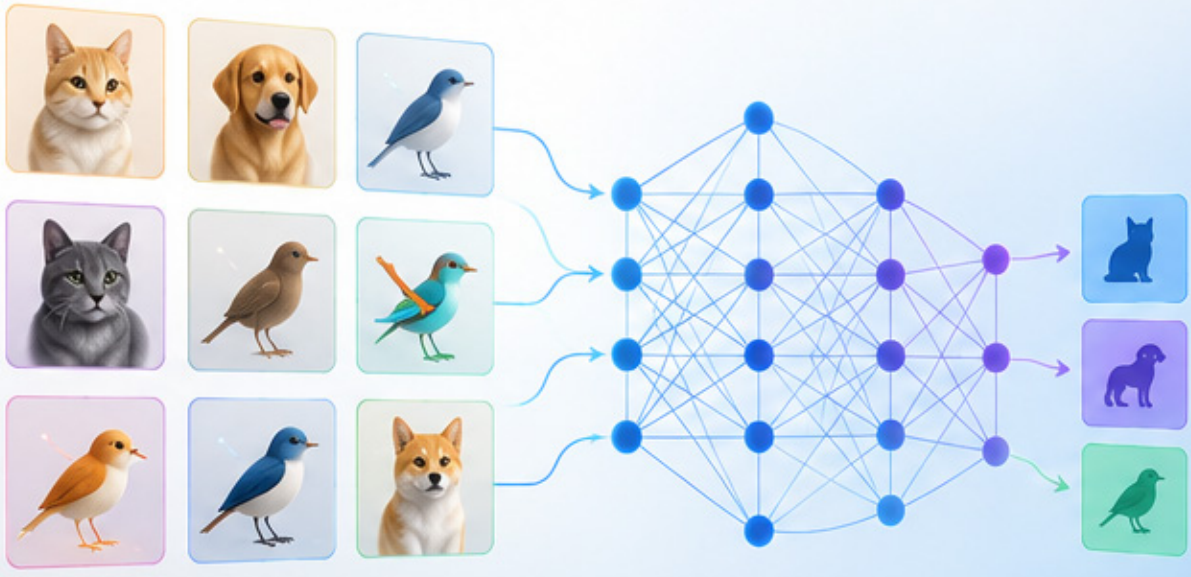
A) Teknolojinin çok yavaş ilerlemesi
B) Hukuk sistemlerinin teknolojik değişime aynı hızda uyum sağlamakta zorlanması
C) Kişisel verilerin artık hiç paylaşılmaması
D) Eğitimin dijitalleşmesi
E) Makine öğrenmesinin tüm ülkelerde yasaklanması

9. Yapay zekâya özgü yeni yasal düzenlemelerin ortaya çıkmasının temel nedeni aşağıdakilerden hangisidir?

A) Uluslararası teknoloji şirketlerinin yapay zekâ pazarındaki tekelci yapısını korumak istemesi
B) İnsan hakları, veri güvenliği, şeffaflık ve hesap verebilirlik gibi alanlarda yeni risklerin ortaya çıkması
C) Geleneksel yazılım geliştirme maliyetlerinin yapay zekâ ile birlikte artış göstermesi
D) Yapay zekâ sistemlerinin artık insan müdahalesine hiçbir şekilde ihtiyaç duymaması
E) Mevcut telif hakkı yasalarının yapay zekânın ürettiği içeriklerin tümünü kamu malı sayması

10. 2021 tarihli UNESCO Yapay Zekâ Etiği Tavsiye Kararı, aşağıdaki ilkelerden hangisini özellikle öne çıkarmaktadır?

A) Ticari rekabet üstünlüğü
B) İnsan hakları, ayrımcılığın önlenmesi, çevresel sürdürülebilirlik ve şeffaflık
C) Donanım üretimi
D) Çevrim içi sınav güvenliği
E) Blok zincir sözleşmelerine ilişkin teknik standartlar





3. ÖĞRENME BİRİMİ

YAPAY ZEKÂDA ÖĞRENME TÜRLERİ

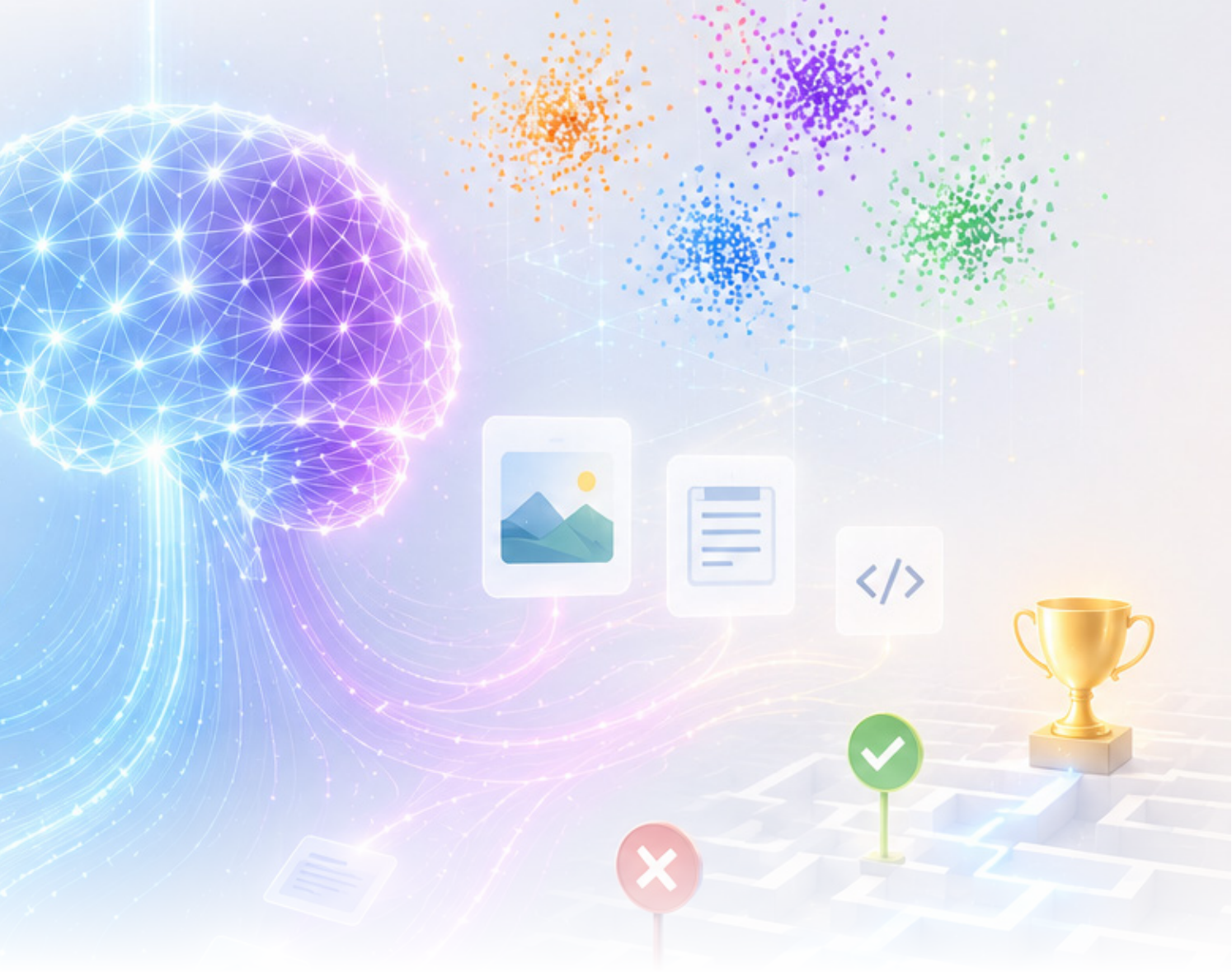
KONULAR

3.1. YAPAY ZEKÂ ÖĞRENME TÜRLERİNİ KARŞILAŞTIRMA

3.2. YAPAY ZEKÂ ÖĞRENME TÜRLERİNİ UYGULAMA

ANAHTAR KAVRAMLAR

- denetimli öğrenme
- denetimsiz öğrenme
- etiketli veri
- etiketsiz veri
- regresyon
- pekiştirmeli öğrenme
- sınıflandırma
- yarı denetimli öğrenme



BEKLENTİLER

Bu öğrenme biriminde sizden

- yapay zekâ öğrenme türlerini tanımlayıp karşılaştırmanız
- probleme göre en uygun yapay zekâ öğrenme türünü seçip uygulamanız
- etiketli veri setleri üzerinde denetimli öğrenme modellerini eğitmeniz ve modelin performansını değerlendirmeniz
- etiketsiz veri setleri üzerinde denetimsiz öğrenme (kümeleme) algoritmalarını uygulayarak sonuçları görselleştirmeniz
- yarı denetimli ve pekiştirmeli öğrenme yaklaşımlarını temel düzeyde modellemeniz **beklenmektedir.**

Öğrenme Birimi Karekodu

BAŞLARKEN

Günlük yaşamda kullanılan dijital uygulamalar, farklı öğrenme yöntemlerinden yararlanarak kullanıcılarına hizmet sunmaktadır. Öneri sistemleri, akıllı asistanlar, oyunlar, navigasyon uygulamaları gibi pek çok sistem; verilerden anlam çıkarma ve karar verme süreçlerinde öğrenme mekanizmalarını kullanır. Bu sistemlerin bir kısmı, doğru cevapların önceden verildiği veri kümeleriyle eğitilirken bir kısmı veriler arasındaki ilişki ve benzerlikleri kendisi keşfeder. Bazı sistemler de deneme yanılma yöntemiyle en uygun davranışı öğrenerek performansını geliştirir.

Yapay zekâ sistemleri, insan öğrenmesine benzer biçimde farklı yöntemlerle öğrenir. Bu öğrenme süreçleri çözülecek problemin özelliklerine, kullanılan verinin türüne ve sistemin amacına göre değişiklik gösterir. Bu nedenle yapay zekâda tek bir öğrenme biçimi yerine farklı durumlara uygun çeşitli öğrenme yaklaşımları kullanılır.

Öğrenmedeki çeşitlilik, yapay zekânın farklı alanlarda etkili ve esnek çözümler üretebilmesini sağlar. Yapay zekâ uygulamalarının geliştirilme aşamasında problem yapısına uygun yöntemlerin seçilmesi önemlidir. Yöntem seçimi, yapay zekâ sistemlerinin başarısını doğrudan etkileyen temel unsurlardan biridir.

Yapay zekâda öğrenme türleri bölümünde; öğrenme türlerini tanıyıp bu türler arasındaki farkları inceleyecek ve problem durumuna göre öğrenme yaklaşımlarını değerlendireceksiniz.

ÖNCE DÜŞÜN

Bir çocuğun yeni bir nesnenin adını bir başkasından öğrenmesiyle nesneleri gözlemleyip benzerliklerine göre gruplandırması arasında nasıl farklar olabileceğini düşününüz.

Bilgi veya beceri öğrenirken başkasının açıklama ve yönlendirmelerinden veya kendi gözlem ve deneyimlerimizden yararlanabiliriz. Öğrenme süreçlerinde bu yöntemler arasında hangi şartlara göre tercih yaparsınız? Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

Aşağıdaki alanlarda yer alan **etiketli veri** ve **etiketsiz veri** kavramlarının etrafındaki noktalı alanlara bu kavramlarla ilgili düşüncelerinizi ve bildiğiniz sözcükleri yazınız. Yazdıklarınızdan yararlanarak kavram arasındaki temel farkı belirleyiniz.

.....
.....
.....
.....
.....
.....
.....

ETİKETLİ VERİ

.....
.....
.....
.....
.....
.....
.....

.....
.....
.....
.....
.....
.....
.....

ETİKETSİZ VERİ

.....
.....
.....
.....
.....
.....
.....

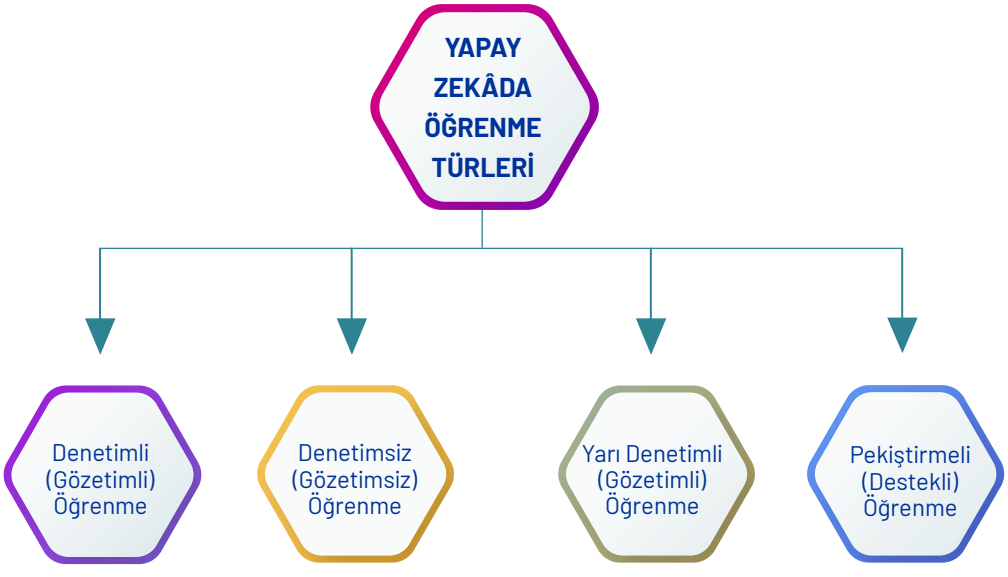
3.1. YAPAY ZEKÂ ÖĞRENME TÜRLERİNİ KARŞILAŞTIRMA

Düşün, Tartış, Paylaş

İnsanlar yeni bilgi ve becerileri farklı yöntemlerle öğrenebilir. Başkasından yardım almak, gözlemlemek, deneyimlemek; bu yöntemlerden bazılarıdır. **Aşağıdaki soruları örnekler vererek cevaplayınız. Cevaplarınızı sınıfta arkadaşlarınızla tartışınız.**

1. **Yeni bir şey öğrenirken hangi yöntemlerin daha etkili olduğunu düşünüyorsunuz?**
2. **Her durumda aynı öğrenme yöntemini kullanmak yeterli olur mu? Neden?**
3. **Bazı durumlarda bir öğrenme yönteminin diğerlerine göre daha yararlı olduğunu düşünüyor musunuz?**

Yapay zekâ sistemleri, kullandıkları öğrenme türlerine göre dört farklı gruba ayrılır (Şema 3.1). Bu gruplandırma, yapay zekânın verileri işleme yöntemlerini ve karar verme mekanizmalarını anlamayı kolaylaştırır.



Şema 3.1: Yapay zekâda öğrenme türleri

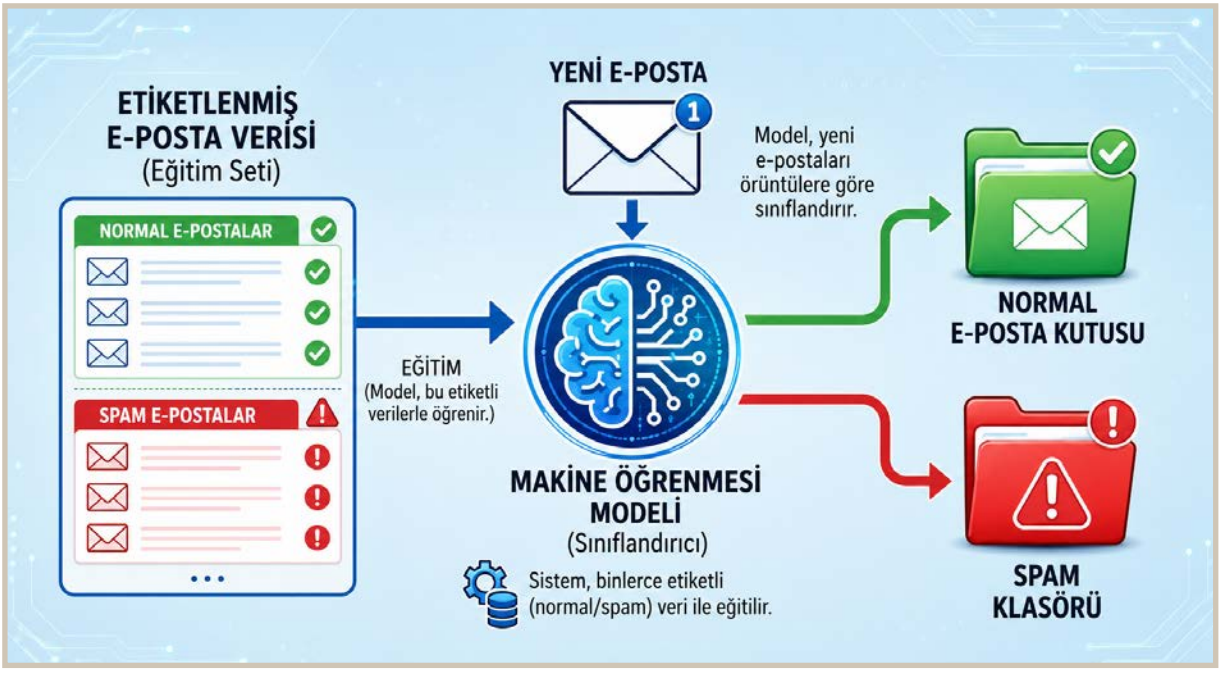
3.1.1. Denetimli Öğrenme

Birçok yapay zekâ uygulaması, geçmişteki verilerden yararlanarak yeni durumlar hakkında tahminde bulunur veya karar verir. E-posta sisteminin gelen mesajları spam (gereksiz) ve normal olarak ayırması, alışveriş sitesinin müşterileri önceden belirlenmiş gruplara göre sınıflandırması veya geçmiş oyun kayıtlarını inceleyen yapay zekâ sisteminin satrançta uygun hamle önerileri üretmesi; bu duruma örnek olarak verilebilir.

Bu tür uygulamalarda yapay zekâ sistemi yalnızca verileri incelemekle kalmaz, aynı zamanda verilere ait doğru sonuçları da öğrenir. Başka bir ifadeyle model, daha önce sonucu bilinen örneklerden yararlanarak yeni veriler için tahminde bulunur. Bu öğrenme yaklaşımı **denetimli öğrenme** olarak adlandırılır.

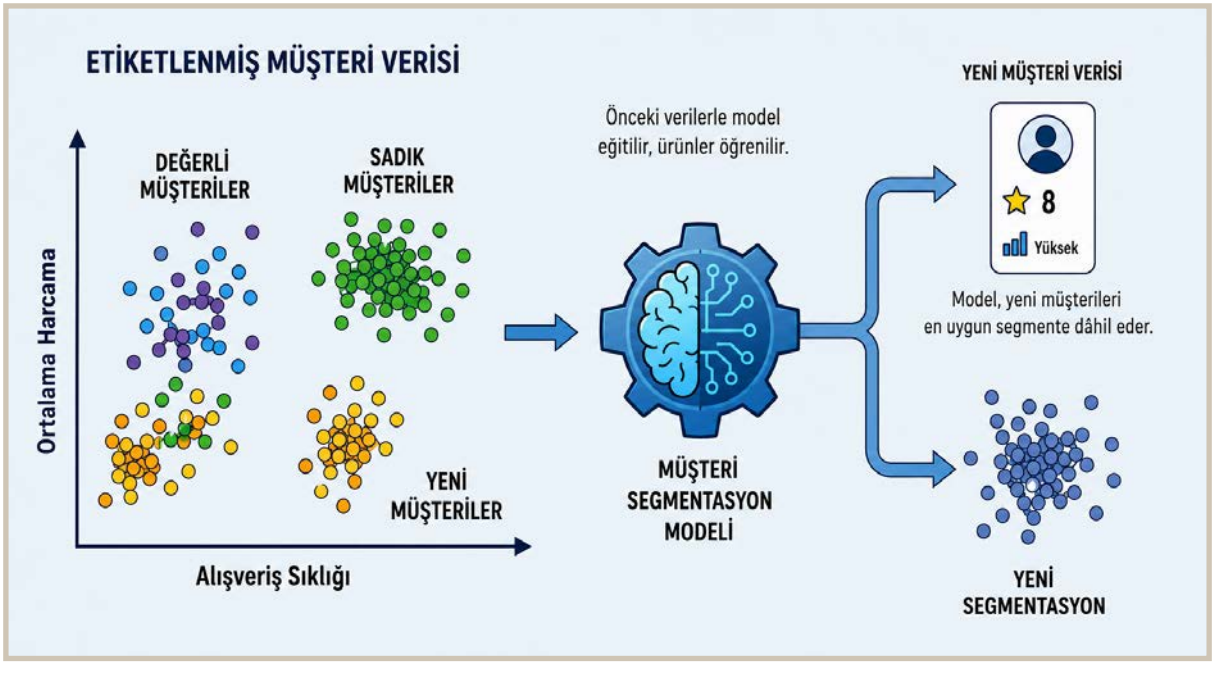
E-postaları ayırması istenen bir yapay zekâ modelinin eğitimi için sistemde daha önce spam ve normal olarak etiketlenmiş çok sayıda e-posta, modele tanıtılır. Model, bu e-postalardaki kelime kullanımı, bağlantı içerme

durumu, konu başlığı, gönderici bilgisi gibi özellikleri inceler. Ulaştığı sonuçları kullanarak yeni gelen e-postanın spam olup olmadığını tahmin etmeye çalışır (Görsel 3.1).



Görsel 3.1: Akıllı e-posta sınıflandırması

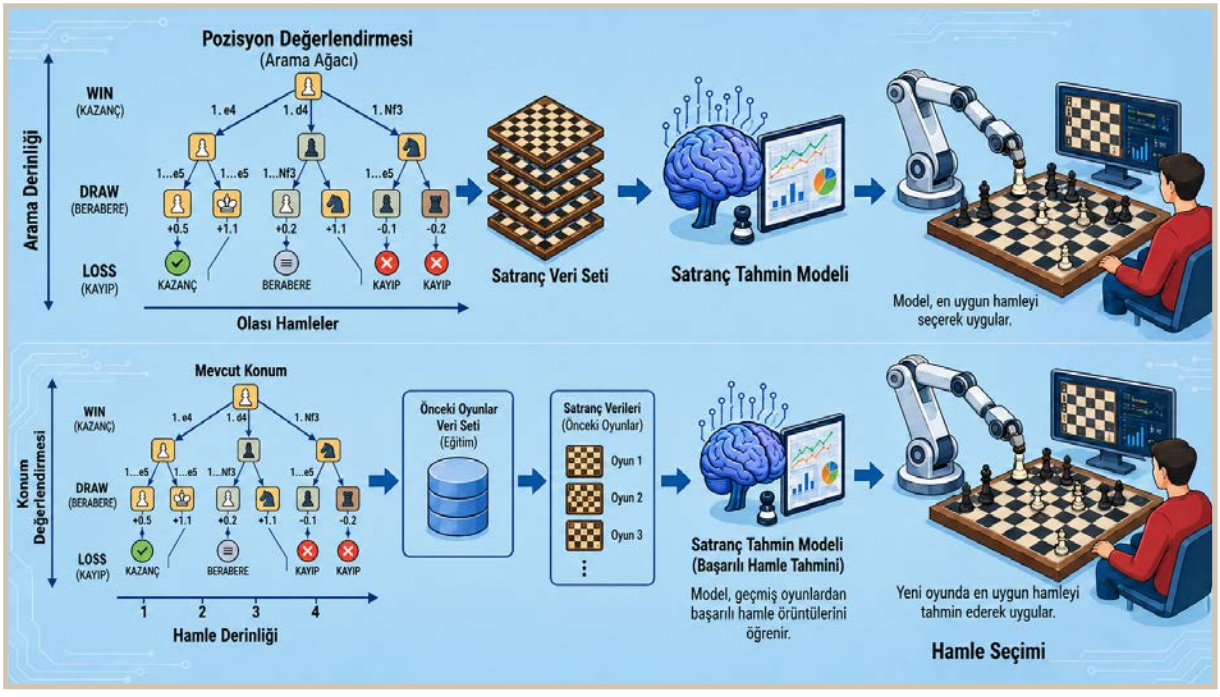
E-ticaret sisteminde de geçmiş müşteri verilerinden yararlanılabilir. Örneğin müşteriler daha önce "sık alışveriş yapan", "kampanyaya duyarlı", "uzun süredir alışveriş yapmayan" gibi gruplara ayrılmış olabilir. Model, etiketlenmiş verilerden öğrenerek yeni bir müşterinin hangi gruba daha uygun olduğunu tahmin eder (Görsel 3.2). Bu sayede işletmeler, müşterilere, uygun kampanya ve öneriler sunabilir.



Görsel 3.2: Müşteri davranışı analizi

Satranç oynayan yapay zekâ sistemlerinde de geçmiş oyun verilerinden yararlanılabilir. Daha önce oynanmış oyunlarda hangi hamlelerin başarılı olduğu biliniyorsa model bu örneklerden öğrenebilir.

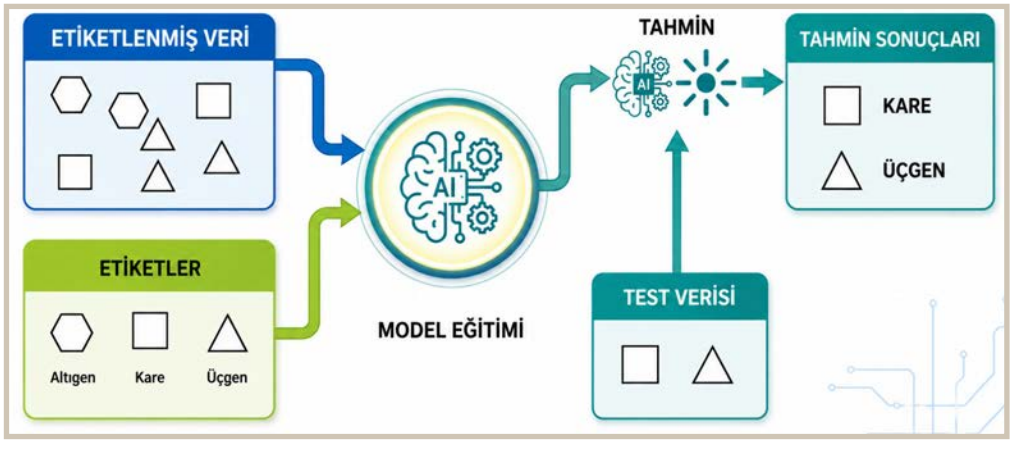
Yeni bir oyun sırasında benzer durumları değerlendirerek uygun hamle önerileri üretebilir (Görsel 3.3).



Görsel 3.3: Stratejik karar veren yapay zekâ

Verilen örneklerin ortak özelliği, doğru sonuçları önceden bilinen verilerle modelin eğitilmesidir. Denetimli öğrenmede veri setindeki her örnek, bir giriş verisi ve bu girişe karşılık gelen doğru sonuçtan oluşur. Doğru sonuçlara **etiket** adı verilir. Model, giriş verileriyle etiketler arasındaki ilişkiyi öğrenerek daha önce karşılaşmadığı veriler için tahminde bulunur.

Denetimli öğrenmenin ilk aşamasında etiketlenmiş veri seti hazırlanır. Daha sonra model bu verilerle eğitilir. Eğitim tamamlandıktan sonra model, test verileri üzerinde tahmin yürütür. Son aşamada tahminler gerçek sonuçlarla karşılaştırılır, modelin başarısı değerlendirilir (Görsel 3.4).



Görsel 3.4: Denetimli öğrenme süreci

Denetimli öğrenmenin temel amacı, veriler arasındaki ilişkiyi temsil eden bir model oluşturmaktır. Model, eğitim sürecinde öğrendiği ilişkileri kullanarak yeni veriler için doğruya en yakın sonucu üretmeye çalışır. Bu süreçte modelin yaptığı tahminler, gerçek sonuçlarla karşılaştırılır. Gerekli durumlarda model yeniden eğitilerek geliştirilebilir.

Denetimli öğrenme süreci genel olarak şu adımlardan oluşur:

1. Etiketlenmiş veri setinin hazırlanması
2. Modelin bu verilerle eğitilmesi
3. Yeni veriler üzerinde tahmin yapılması
4. Tahminlerin gerçek sonuçlarla karşılaştırılması
5. Model başarısının değerlendirilip geliştirilmesi

Yapılan çalışmalar sayesinde yapay zekâ sistemleri; e-posta sınıflandırma, müşteri analizi, hastalık tahmin etme gibi birçok alanda kullanılabilir.

1. SIRA SİZDE

Bir e-posta uygulaması, gelen mesajları **spam** ve normal olarak sınıflandırabilmek için denetimli öğrenme yöntemi kullanmaktadır. Sistem, daha önce **spam** ve **normal** olarak etiketlenmiş çok sayıda e-postayla eğitilmiştir. Böylece yeni gelen e-postaların hangi sınıfa ait olabileceğini tahmin edebilmektedir.

Aşağıda sisteme yeni gelen bazı e-postalar verilmiştir.

No.	E-posta İçeriği
1.	Hesabınız tehlikede! Hemen şifrenizi güncelleyin.
2.	Yarınki ders saat 10.00'a alınmıştır.
3.	Tebrikler! Büyük ödülü kazandınız, hemen tıklayın.
4.	Toplantı notları ektedir, inceleyiniz.

Verilen bilgiler doğrultusunda aşağıdaki soruları cevaplayınız.

1. Yukarıdaki e-postaları spam veya normal olarak sınıflandırınız.

.....

2. Sınıflandırma yaparken hangi ipuçlarını (kelime, ifade, içerik türü vb.) kullandığınızı açıklayınız.

.....

3. Denetimli öğrenme sürecinde aşağıdaki kavramları uygun aşamalarla eşleştiriniz.

.....

Kavramlar

Etiketli veri - Model eğitimi – Tahmin - Yeni veri - Doğruluk kontrolü

Denetimli Öğrenme Aşamaları	İlgili Kavram
Eğitim verisinin hazırlanması	
Modelin öğrenmesi	
Yeni verinin analizi	
Sonuç üretme	
Sonucun değerlendirilmesi	

4. Bir e-posta sisteminin yanlış sınıflandırma yapması durumunda ne gibi sorunlar ortaya çıkabilir? Görüşlerinizi yazınız.

.....

Denetimli öğrenme yalnızca e-postaları ayırmak için kullanılmaz. Bu yöntem, problemin türüne göre farklı görevlerde de uygulanabilir. Denetimli öğrenmede en yaygın kullanılan iki temel görev **sınıflandırma** ve **regresyon**dur.

Sınıflandırma, verilerin belirli gruplara veya sınıflara ayrılmasıdır. Ortaya çıkacak problemlerde modelin ürettiği sonuç bir **kategori** veya **etiket**tir. Örneğin sağlık sisteminde, hastanın tahlil sonuçları incelenerek "Sağlıklı.", "Enfeksiyon riski var.", "Kontrol gerekli." gibi sınıflandırmalar yapılabilir. Model, hastaya ait verileri değerlendirerek onu uygun sınıfa yerleştirir.

Regresyon, sayısal değerin tahmin edilmesini amaçlayan denetimli öğrenme yöntemidir. Bu tür problemlerde modelin çıktısı bir kategori değil sayısal sonuçtur. Örneğin bir emlak uygulamasında konutun metrekaresi, oda sayısı, bulunduğu konum, yaşı gibi bilgiler kullanılarak fiyatı tahmin edilebilir. Model, geçmiş verilerden öğrenerek konutun yaklaşık değerini hesaplar.

Sınıflandırma ve regresyon arasındaki temel fark, modelin ürettiği sonucun türüdür. Model, veriyi belirli bir kategoriye yerleştirirse sınıflandırma; sayısal bir değer tahmin ederse regresyon işlemi gerçekleştirmiş olur. Denetimli öğrenmede kullanılan bu iki temel yaklaşımın özellikleri ve kullanım alanlarına ilişkin örnekler Tablo 3.1'de verilmiştir.

Tablo 3.1: Denetimli Öğrenmede Sınıflandırma ve Regresyon Görevleri

Denetimli Öğrenme Görevi	Açıklama	Örnek
Sınıflandırma	Verileri belirli sınıflara ayırır.	Hasta tahlillerine göre teşhis koyma
Regresyon	Sayısal bir değer tahmin eder.	Evin metrekaresine göre fiyat tahmini yapma

Denetimli öğrenme problemlerinde öncelikle elde edilmek istenen sonucun türü belirlenmelidir. Sonuç bir **kategoriyse** sınıflandırma, **sayısal bir değerse** regresyon yöntemi tercih edilir. Böylece yapay zekâ modeli, problemin yapısına uygun şekilde eğitilir ve daha doğru tahminler yapabilir.

Denetimli öğrenme problemlerinde doğru sonuçlara ulaşabilmek için farklı algoritmalar kullanılır. Bu algoritmalar, veriler arasındaki ilişkileri öğrenerek yeni verilerin tahmin edilmesini sağlar. Problemin türüne ve veri yapısına göre farklı algoritmalar tercih edilir.

Yaygın olarak kullanılan bazı denetimli öğrenme algoritmaları şunlardır:

K-En Yakın Komşu (KNN): KNN yeni veriyi en yakın K komşusuna göre sınıflandırır.

Örnek: E-postanın spam olup olmadığını benzer e-postalara bakarak belirlemek

Karar Ağaçları (Decision Trees): Verileri belirli kurallara göre dallara ayırarak karar verir.

Örnek: Müşterinin geçmiş alışveriş bilgilerine göre kampanyaya ilgi gösterip göstermeyeceğini tahmin etmek

Doğrusal Regresyon (Linear Regression): Sayısal değer tahminlerinde kullanılır. Değişkenler arasındaki ilişkiyi doğrusal bir modelle ifade eder.

Örnek: Günlük sıcaklık değerlerine göre dondurmacının satabileceği ürün miktarını tahmin etmek

Lojistik Regresyon (Logistic Regression): Adında regresyon ifadesi geçmesine rağmen sınıflandırma problemlerinde kullanılır.

Örnek: Müşterinin aboneliğini devam ettirip ettirmeyeceğini tahmin etmek

Destek Vektör Makineleri (SVM): Verileri ayıran sınırı belirlemeye çalışır.

Örnek: El yazısıyla yazılmış rakamları özelliklerine göre sınıflandırmak

Algoritmaların her biri farklı durumlarda avantaj sağlar. Denetimli öğrenme uygulamalarında, problemin yapısına en uygun algoritmanın seçilmesi gerekir.

2. SIRA SİZDE

1. Bir teknoloji firması, farklı yapay zekâ uygulamaları geliştirmektedir. Bu uygulamaların her biri için farklı veri türleri kullanılarak tahmin yapılmaktadır. Aşağıda bu sistemlere ait bazı senaryolar ve senaryo numaralarını içeren tablo verilmiştir. Senaryoları **sınıflandırma** veya **regresyon** görevi olarak ayırınız. Tablodaki senaryo numaralarının karşılarına sonuçları yazınız.

   I. Bir hastanenin yapay zekâ sistemi, hastaların tahlil sonuçlarına bakarak "sağlıklı" veya "hasta" şeklinde sonuç üretmektedir.

   II. Bir emlak uygulaması, konut metrekaresi ve konumuna göre yaklaşık satış fiyatını tahmin etmektedir.

   III. Bir e-posta uygulaması, gelen mesajları "spam" ve "normal" olarak ayırmaktadır.

   IV. Bir uygulama, öğrencilerin çalışma süresine göre sınavdan alacağı puanı tahmin etmektedir.

Senaryo Numarası	Problem Türü (Sınıflandırma / Regresyon)
I	
II	
III	
IV	

2. Her senaryoda modelin ürettiği çıktının neden **kategori** veya **sayısal değer** olduğunu açıklayınız.

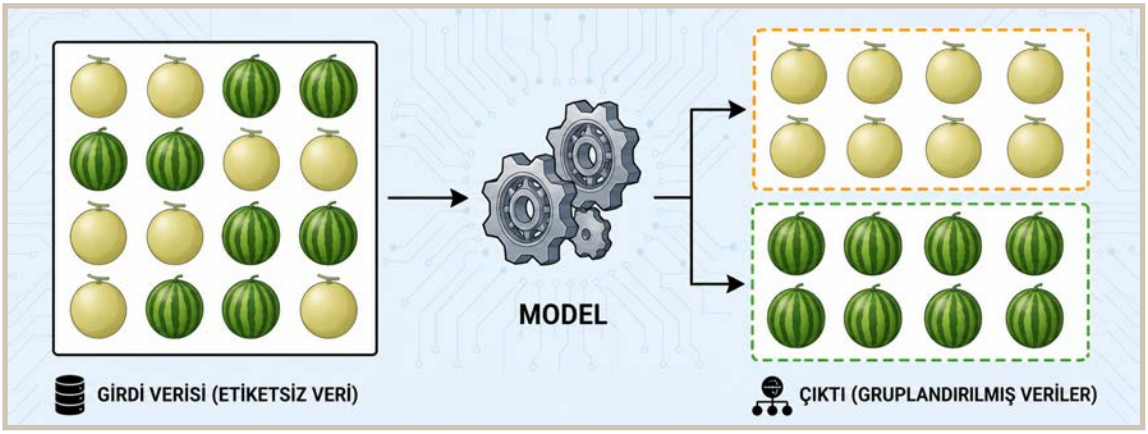
.....
.....
.....
.....

3.1.2. Denetimsiz Öğrenme

Denetimsiz öğrenme, verilerin etiketli olmadığı durumlarda kullanılan yapay zekâ öğrenme türüdür. Bu yöntemde model, veriler arasında önceden belirlenmiş doğru sonuçlar olmadan çalışır. Bunun yerine verilerin yapısını inceleyerek benzerlik ve farklılıkları analiz eder. Böylece sistem, herhangi bir yönlendirme olmadan verilerden anlamlı bilgiler elde edebilir. Denetimsiz öğrenmenin temel amacı, veriler içerisindeki gizli örüntüleri ortaya çıkarmak ve benzer özelliklere sahip verileri gruplandırmaktır.

Karışık hâlde bulunan madeni paralar yalnızca boyutlarına bakılarak gruplandırılabilir. Büyük boyutlu ve küçük boyutlu paralar ayrılabilir. İşlem yapılırken paraların değerlerine dair herhangi bir bilgi kullanılmamış olur. Bu durum, etiketlenmemiş verilerle gerçekleştirilen öğrenme sürecine örnektir. Etiketli bir bilgiye ihtiyaç duymadan yalnızca benzerliklere göre yapılan bir gruplamayı gösterir.

Görsel 3.5'te farklı özelliklere sahip karışık verilerin yapay zekâ modeline verildiği görülmektedir. Veriler herhangi bir etiket içermemektedir. Model, veriler arasındaki benzerlikleri analiz ederek onları belirli gruplara ayırır. Benzer özelliklere sahip veriler aynı kümelerde toplanır. Bu durum, denetimsiz öğrenmede modelin verileri kendi kendine organize ettiğini gösterir.



Görsel 3.5: Denetimsiz öğrenme

Denetimsiz öğrenme birçok alanda kullanılır. Örneğin eğlence platformlarında film, sosyal medyada arkadaş, çevrim içi alışveriş sitelerinde ürün önerileri bu yönteme dayanır. Bu sistemler, davranış ve ilgi alanlarını analiz ederek benzer özelliklere sahip kullanıcıları aynı gruplarda toplar. Kullanıcıların ortak tercihlerini gruba yeni dâhil olanlara öneriler oluşturmak için kullanır.

Alışveriş sitelerinde kullanılan sepet analizi, denetimsiz öğrenmenin önemli bir kullanım alanıdır. Analizde belirli ürünleri birlikte satın alan müşteriler tespit edilir. Bu ürünlerden birini satın alan yeni müşterilere diğer ürün veya ürünler önerilir. Böylece kullanıcı davranışlarına dayalı öneri sistemleri geliştirilebilir.

BİLGİ KUTUSU

Denetimsiz öğrenmede model yalnızca grupları oluşturur, bu grupların ne anlama geldiğini yorumlamak çoğu zaman insana düşer. Örneğin alışveriş sisteminde model, müşterileri benzer davranışlarına göre ayırabilir ancak bu gruplara "sık alışveriş yapanlar", "indirim takipçileri" veya "yeni müşteriler" gibi anlamlı adlar verilmesi için sonuçların incelenmesi gerekir. Denetimsiz öğrenmede önemli olan yalnızca verileri gruplamak değil oluşan grupların gerçek yaşamda neyi ifade ettiğini doğru yorumlayabilmektir. Aynı veri seti farklı özelliklere göre incelendiğinde farklı gruplar ortaya çıkabilir. Bu durum, modele verilen özelliklerin, sonucu doğrudan etkilediğini gösterir.

Denetimsiz öğrenme sürecinde, verileri gruplamak ve örüntüleri ortaya çıkarmak için çeşitli algoritmalar kullanılır. En yaygın kullanılan denetimsiz öğrenme algoritmalarından bazıları şunlardır:

K-Means (K-Ortalamalar) Algoritması: Verileri önceden belirlenen bir küme sayısına göre gruplandırır. Her veri noktası, kendine en yakın küme merkezine atanır. Daha sonra küme merkezleri yeniden hesaplanır ve işlem, veriler en uygun kümelere yerleşene kadar devam eder. Bu algoritma, basit ve hızlı olduğu için sıkça tercih edilir.

Hiyerarşik Kümeleme Yöntemi: Veriler arasında ağaç yapısı (dendrogram) oluşturarak çalışır. Veriler ya küçük gruplardan başlayarak birleştirilir veya büyük bir gruptan başlayarak alt gruplara ayrılır. Oluşan yapı sayesinde veriler arasındaki benzerlik ilişkileri ayrıntılı olarak incelenebilir.

DBSCAN (Yoğunluk Tabanlı Kümeleme) Algoritması: Veri noktalarının yoğunluğuna göre kümeler oluşturur. Birbirine yakın ve yoğun bölgelerde bulunan veriler aynı kümelerde toplanırken seyrek bölgelerde kalan veriler gürültü (aykırı veri) olarak değerlendirilir. Bu algoritma özellikle düzensiz ve karmaşık veri yapılarında etkili sonuçlar verir.

Algoritmaların her biri farklı veri türlerinde ve problem durumlarında avantaj sağlar. Denetimsiz öğrenme uygulamalarında, verinin yapısına ve işlemin amacına en uygun algoritmanın seçilmesi gerekir. Algoritma seçimi, modelin elde edeceği sonuçların doğruluğunu ve anlamlılığını etkiler.

3. SIRA SİZDE

Aşağıda marketlerde bulunan bazı ürünler verilmiştir. Ürünleri inceleyerek soruları cevaplayınız.

Ürünler:

Sandviç – Meyve suyu – Çikolata – Su – Cips – Ayran – Bisküvi – Gazoz

1. Listedeki ürünlerden benzer özelliklere göre en az üç farklı grup oluşturunuz.

.....
.....
.....
.....

2. Gruplandırmayı yaparken kullandığınız ölçütleri (yiyecek / içecek, paketli ürün / paketsiz ürün, sağlıklı / sağlıksız vb.) açıklayınız.

.....
.....
.....
.....

3. Aşağıda verilen denetimsiz öğrenme algoritmalarını, açıklamalarıyla eşleştiriniz.

A) K-Means	1. Verileri belirli sayıda kümeye ayırarak her veri noktasını en yakın merkeze göre gruplandırır.
B) Hiyerarşik kümeleme	2. Veriler arasında ağaç yapısı oluşturarak grupları adım adım birleştirir veya alt gruplara ayırır.
C) DBSCAN	3. Veri noktalarının yoğunluğuna göre kümeler oluşturur, kümeye ait olmayan aykırı verileri belirler.
	4. Etiketlenmiş veriler kullanılarak önceden belirlenmiş sınıflara göre tahmin yapılmasını sağlar.

4. Sizce ürünlerin gruplandırılması için kullanılabilecek en uygun algoritma hangisidir? Gerekçeleriyle açıklayınız.

.....
.....
.....
.....
.....

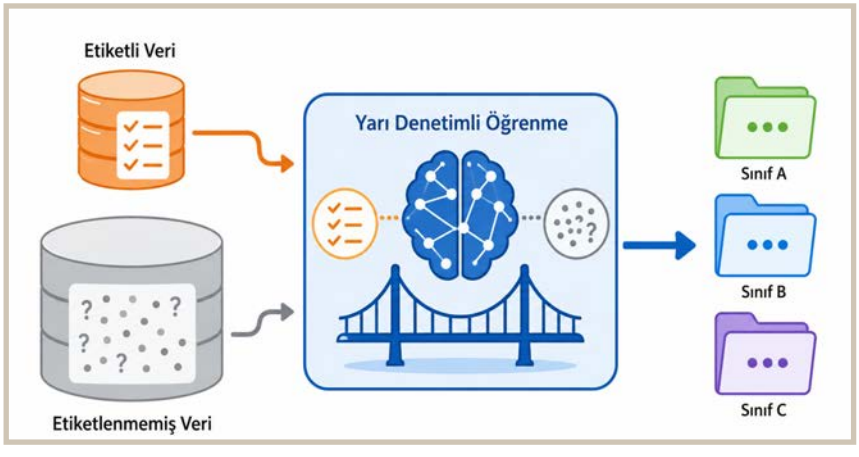
5. Bu etkinlikte yapılan işlemin denetimli öğrenmeden farkını açıklayınız.

.....
.....
.....
.....
.....

3.1.3. Yarı Denetimli Öğrenme

Yarı denetimli öğrenme, **denetimli öğrenmeyle denetimsiz öğrenmenin birlikte kullanıldığı** öğrenme türüdür. Bu yöntemde veri setinin bir kısmı **etiketlenmiş,** büyük bir kısmıysa **etiketlenmemiş verilerden** oluşur. Bu yönüyle yarı denetimli öğrenme, iki öğrenme yaklaşımı arasında bir **köprü görevi** görür (Görsel 3.6).

Bir öğretmenin yalnızca birkaç öğrencinin sınav kâğıdını okuyup değerlendirmesi, geri kalan çok sayıdaki sınavınsa bu örneklerden yola çıkılarak bir sistem tarafından tahmin edilmesi durumunda; değerlendirilen sınav kâğıtları, diğer sınav kâğıtlarının değerlendirilmesi için **rehber** olmuş olur. Bu benzetmede yapay zekâda kullanılan **yarı denetimli öğrenme** yaklaşımıyla aynı işleyiş kullanılmıştır.



Görsel 3.6: Yarı denetimli öğrenmenin işleyişi

Yarı denetimli öğrenme sürecinin aşamaları (Görsel 3.6) genel olarak şu adımlardan oluşur:

1. Etiketlenmemiş veriler, denetimsiz öğrenme yöntemleri kullanılarak benzerliklerine göre kümelere ayrılır.
2. Bu kümeler, etiketli verilerden elde edilen bilgiler yardımıyla anlamlandırılır.
3. Bu aşamada etiketlenmemiş veriler için sözde etiketler (pseudo-labels) oluşturulur.
4. Elde edilen yeni veri seti kullanılarak model yeniden eğitilir.

Bu süreçte etiketli veriler, etiketlenmemiş verilerin yorumlanmasında bir **rehber** görevi görür. Böylece model, hem az sayıdaki doğru bilgiden hem de çok sayıdaki etiketsiz veriden yararlanarak başarılı tahminler yapabilir.

Yarı denetimli öğrenme; **sınıflandırma, regresyon, tahmin** gibi birçok problem türünde kullanılabilir. Özellikle **etiketli verinin az, etiketlenmemiş verininse çok olduğu** durumlarda oldukça etkilidir.

Yapay zekâ yardımı olmadan veri toplamak genellikle kolay olsa da bu verileri etiketlemek zaman alıcı ve maliyetlidir.

Yarı denetimli öğrenme, günümüzde özellikle **bilgisayarlı görü, doğal dil işleme** gibi alanlarda yaygın olarak kullanılmaktadır.

4. SIRA SİZDE

Bir öğretmen, sınıfındaki öğrencilerin sınav kâğıtlarını değerlendirmek istemektedir. Ancak zaman sınırlı olduğu için yalnızca bazı sınav kâğıtlarını ayrıntılı olarak okuyup puanlayabilmiştir. Geri kalan çok sayıdaki sınav kâğıdının değerlendirilmesi için bir yapay zekâ sisteminden yararlanmayı planlamaktadır.

Sistem, öğretmenin değerlendirdiği sınavlardan elde edilen bilgileri kullanarak diğer sınavlar hakkında tahminler yapmakta, benzer cevaplara sahip sınav kâğıtlarını gruplandırarak değerlendirme sürecini desteklemektedir.

Yukarıdaki senaryoda yer alan yarı denetimli öğrenme sürecini dikkate alarak aşağıda verilen kavramları uygun aşamalarla eşleştiriniz.

A) Etiketli veri
B) Etiketlenmemiş veri
C) Kümeleme
Ç) Sözde etiketleme
D) Modelin yeniden eğitilmesi
E) Tahmin
F) Doğruluk kontrolü
G) Veri grubu oluşturma

1. Öğretmenin okuyup puanladığı sınav kâğıtları hangi kavramla ilişkilidir?
   .....
   .....
2. Öğretmenin henüz okumadığı sınav kâğıtları hangi kavramla ilişkilidir?
   .....
   .....
3. Sistem, benzer cevaplara sahip sınav kâğıtlarını kendi içinde gruplandırdığında hangi işlem gerçekleşmiş olur?
   .....
   .....
4. Etiketlenmemiş sınavlara, öğretmenin puanladığı örneklerden yararlanılarak geçici etiketler verilmesi hangi kavramla açıklanır?
   .....
   .....
5. Sistem, hem öğretmenin puanladığı sınavları hem de tahmin edilen sınavları kullanarak kendini geliştirdiğinde hangi aşama gerçekleşir?
   .....
   .....
6. Sistem, yeni bir sınav kâğıdının puan aralığını belirlemeye çalıştığında hangi işlem yapılmış olur?
   .....
   .....

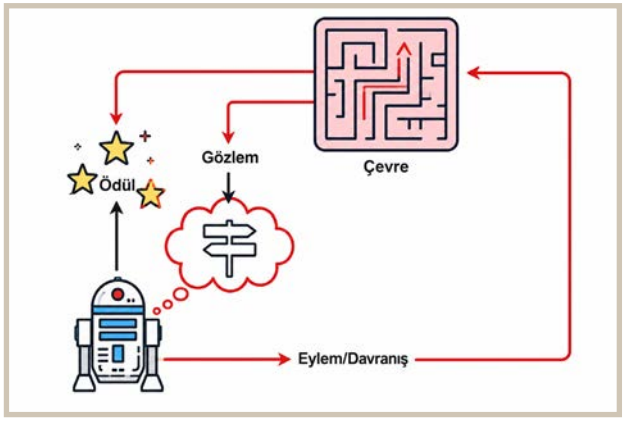
3.1.4. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme, ajanın (agent) ortam (environment) içinde **eylem (action)** seçerek ilerlediği ve bu eylemler sonucunda **ödül (reward)** veya ceza aldığı öğrenme sürecidir. Süreçte amaç, ajanın uzun vadede elde edeceği toplam ödülü en üst seviyeye çıkaracak bir **strateji (policy)** geliştirmesidir.

Bir köpeğe komut öğretilir. Köpek doğru davranışı gerçekleştirdiğinde ödül maması verilir, yanlış davranış sergilediğindeyse ödül verilmez. Zamanla köpek, hangi davranışların ödül getirdiğini öğrenerek doğru komutları uygulamaya başlar. Zamanla köpek, aldığı geri bildirimlere göre öğrenme gerçekleştirir. Bu durum, yapay zekâda kullanılan **pekiştirmeli öğrenme (reinforcement learning)** yaklaşımının temel mantığını yansıtmaktadır.

Ajan çevreyle sürekli etkileşim hâlindedir. Bulunduğu durumu (state) gözlemler, duruma uygun eylem seçer ve eylem sonucunda ortamdan geri bildirim alır. Geri bildirim, ajanın sonraki kararlarını doğrudan etkiler (Görsel 3.7). Bu döngü şu şekilde özetlenebilir:

Durum → Eylem → Ödül → Yeni Durum



Görsel 3.7: Pekiştirmeli öğrenme süreci

Ajan, bu etkileşim sürecini defalarca tekrar ederek hangi durumlarda hangi eylemin daha avantajlı olduğunu öğrenir. Böylece başlangıçta rastgele davranışlar sergileyen sistem, zamanla daha bilinçli ve hedefe yönelik kararlar almaya başlar.

Örneğin bir robotun bir labirentte çıkışı bulmaya çalıştığı durumda, robot her adımda farklı yönleri deneyerek ilerler. Doğru yöne yaklaştığında ödül alırken, yanlış yollara girdiğinde ceza alır. Geri bildirimler sayesinde robot, en kısa ve en doğru yolu keşfeder.

Pekiştirmeli öğrenmenin en önemli özelliği, **doğru cevapların önceden verilmemesidir.** Denetimli öğrenmeden farklı olarak sistem, etiketli veriyle değil, tamamen **geri bildirim mekanizması**yla öğrenir. Olasılıkların önceden tanımlanamadığı dinamik ortamlarda etkili bir yöntemdir.

Pekiştirmeli öğrenme sürecinde ajanın davranışlarını geliştirebilmesi için çeşitli algoritmalar kullanılır. Algoritmalar, ajanın geçmiş deneyimlerini değerlendirerek gelecekte daha iyi kararlar almasını sağlar.

Örneğin **Q-learning algoritması,** her durum-eylem çifti için bir değer (Q-değeri) hesaplar. Değer, o eylemin uzun vadede sağlayacağı toplam ödülü temsil eder. Ajan, zamanla değerleri güncelleyerek en yüksek ödülü sağlayan eylemleri tercih etmeye başlar.

SARSA algoritmasıysa öğrenme sürecinde ajanın mevcut davranış politikasını dikkate alır. Temkinli bir öğrenme süreci sağlar ve özellikle riskli ortamlarda dengeli kararlar alınmasına yardımcı olur.

Karmaşık problemlerde, klasik yöntemler yetersiz kalabilir. Bu durumlarda **Deep Q-Network (DQN)** gibi derin öğrenme tabanlı yaklaşımlar kullanılır. Kullanılan yöntemlerde sinir ağları aracılığıyla çok sayıda durum ve eylem arasındaki karmaşık ilişkiler öğrenilir.

Policy Gradient yöntemleri, ajanın doğrudan en iyi davranış stratejisini öğrenmesini hedefler. Ajan, belirli durumda hangi eylemi seçmesi gerektiğini olasılık hesabıyla öğrenir ve zamanla en yüksek ödülü sağlayan stratejiye ulaşır.

Pekiştirmeli öğrenme, ajanın çevreyle etkileşim kurarak deneyim kazanması ve kazanılan deneyimlere dayalı olarak kararlarını geliştirmesi üzerine kuruludur. Sistem, zamanla daha verimli ve hedef odaklı davranış sergileyerek en uygun çözüm stratejisini oluşturur.

Öğrenme Türlerinin Karşılaştırılması

Yapay zekâ öğrenme türleri; kullanılan veri yapısı, öğrenme yöntemi ve problem türlerine göre farklılık gösterir. Her öğrenme yaklaşımı, belirli durumlarda daha etkili sonuçlar üretir. Bu nedenle doğru yöntemin seçilmesi yapay zekânın verimliliği için önemlidir.

Denetimli, denetimsiz, yarı denetimli ve pekiştirmeli öğrenme türlerinin temel özelliklerinin karşılaştırılması, bu yöntemlerin hangi problemlerde daha uygun olduğunu anlamayı kolaylaştırır. Tablo 3.2'de öğrenme türleri; veri yapısı, öğrenme biçimi ve kullanım alanları açısından karşılaştırılmıştır.

Tablo 3.2: Yapay Zekâ Öğrenme Türleri

	1. DENETİMLİ ÖĞRENME	2. DENETİMSİZ ÖĞRENME	3. YARI DENETİMLİ ÖĞRENME	4. PEKİŞTİRMELİ ÖĞRENME
KULLANILAN VERİ	Etiketli veriler	Etiketsiz veriler	Az etiketli + etiketsiz veriler	Etiket yok (Ödül / ceza sinyalleri)
AMAÇ	Girdilere karşılık doğru çıktıları tahmin etmek	Verilerdeki gizli yapıları ve örüntüleri keşfetmek	Sınırlı etiketli veriden yararlanarak daha iyi tahminler üretmek	Ortamla etkileşerek en iyi kararları öğrenmek
ANALİZ TÜRÜ	Tahminleme (sınıflandırma, regresyon)	Keşifsel analiz (kümeleme, ilişkilendirme)	Tahminleme ve keşifsel analiz birlikte	Karar verme ve optimizasyon
KARMASIKLIK DÜZEYİ	Yüksek	Orta – Yüksek	Yüksek	Çok yüksek
KULLANIM ALANLARI	• Görüntü tanıma • Spam filtreleme • Hastalık teşhisi • Kredi risk analizi	• Müşteri segmentasyonu • Pazar sepeti analizi • Anomali tespiti • Örüntü keşfi	• Tıbbi görüntü analizi • Ses tanıma • Metin sınıflandırma • Web içerik filtreleme	• Robotik kontrol • Oyun oynama • Otonom araçlar • Kaynak yönetimi
ÖRNEK	E-postaların spam veya spam değil olarak sınıflandırılması	Müşterileri alışveriş davranışlarına göre gruplara ayırmak	Az sayıda etiketli tıbbi görüntü ile daha doğru teşhis modeli oluşturmak	Bir robotun labirentten çıkmayı öğrenmesi

5. SIRA SİZDE

Sürücüsüz (otonom) bir araç, şehir içinde trafikte ilerlemektedir. Araç, yaptığı her hareketin sonucunda geri bildirim almakta ve bu geri bildirimleri kullanarak sürüş davranışlarını geliştirmektedir.

Aşağıdaki durumları inceleyiniz. Her durum için aracın ödül mü yoksa ceza mı alacağını belirleyiniz. Ardından geri bildirimin aracın gelecekteki davranışlarını nasıl etkileyebileceğini açıklayınız.

Durumlar

Trafik ışıklarına uygun şekilde durmak -Hız sınırını aşmak -Doğru şeritte ilerlemek -Ani ve kontrolsüz fren yapmak -Yayalara yol vermek -Yanlış şeride geçmek

ÖDÜL (Doğru Davranışlar)	CEZA (Yanlış Davranışlar)

Kullanılan sistemde aracın zamanla daha güvenli sürüş yapmasının nedeni nedir? Açıklayınız.

3.1. ETKİNLİK

Adı	Öğrenme Türlerini Karşılaştıran Dijital Sunu
Amaç	Yapay zekâ öğrenme türlerini karşılaştırarak farklı problem durumları için uygun öğrenme yöntemini belirlemek
Süre	2 ders saati
Görev	Bir teknoloji firması; sağlık, finans ve e-ticaret alanlarında kullanılmak üzere yapay zekâ tabanlı sistemler geliştirmeyi planlamaktadır. Firmanın geliştirmeyi hedeflediği sistemlerde şu özellikler olacaktır: • Hastalık teşhisi yapabilme • Müşteri davranışlarını analiz edebilme • Kullanıcıya özel öneriler sunabilme Yapay zekâ öğrenme türlerini inceleyerek firma için en uygun öğrenme yöntemlerini belirleyen bir dijital sunu hazırlayıp sununuz.
Değerlendirme	Çalışmanız, karekodda verilen analitik dereceli puanlama anahtarında yer alan ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız.

İşlem Basamakları

1. Denetimli öğrenme, denetimsiz öğrenme, yarı denetimli öğrenme ve pekiştirmeli öğrenme türlerini araştırınız.
2. Öğrenme türlerinin temel özelliklerini belirleyiniz.
3. Öğrenme türlerini karşılaştırıp benzer ve farklı yönleri belirleyiniz.
4. Firma tarafından geliştirilmesi planlanan sistemlerin ihtiyaçlarını analiz ediniz.
5. Her problem durumu için uygun öğrenme yöntemini belirleyiniz.
6. Seçtiğiniz öğrenme yöntemlerini gerekçeleriyle açıklayınız.
7. Sunum planınızı oluşturunuz.
8. Sunumunuzu hazırlayıp sınıfta paylaşınız.

Etkinlik Soruları

1. Denetimli öğrenme nedir? Hangi durumlarda tercih edilir?
2. Denetimsiz öğrenme nedir? Hangi problemlerin çözümünde kullanılır?
3. Yarı denetimli öğrenme nedir? Veri kısıtı bulunan durumlarda nasıl avantaj sağlar?
4. Pekiştirmeli öğrenme nedir? Karar verme süreçlerinde nasıl kullanılır?
5. Bu öğrenme türleri arasındaki temel farklar nelerdir?
6. Firmanın geliştirmeyi planladığı sistemler için hangi öğrenme yöntemleri daha uygundur? Gerekçeleriyle açıklayınız.

Derecelendirme ölçeğine ulaşmak için karekodu okutunuz.

Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

3.2. YAPAY ZEKÂ ÖĞRENME TÜRLERİNİ UYGULAMA

Düşün, Tartış, Paylaş

Uygulamalar zamanla gelişip verimliliğini artırabilir ve kullanıcıya daha uygun öneriler sunabilir. **Aşağıdaki soruları cevaplayınız ve düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.**

1. **Telefonunuzdaki fotoğraf uygulaması benzer fotoğrafları nasıl bir araya getiriyor olabilir?**
2. **Müzik veya video uygulamalarının size özel öneriler sunabilmesi için hangi bilgilere ihtiyaç duyduğunu düşünüyorsunuz?**
3. **Günlük yaşamda kullandığınız ve zamanla size daha doğru öneri veya tahminler sunduğunu düşündüğünüz uygulama var mı? Varsa örnek veriniz.**

Yapay zekâ sistemleri, farklı yöntemlerle öğrenerek çeşitli problemleri çözebilir ve kararlar verebilir. Kullanılan öğrenme yöntemi; verinin yapısına, problemin türüne ve amaca göre değişiklik gösterebilir.

3.2.1. Denetimli Öğrenme Uygulamaları

Etiketli veri setlerinin daha iyi anlaşılabilmesi için makine öğrenmesi alanında en yaygın kullanılan örneklerden biri olan **İris veri seti** incelenebilir.

İris veri seti, İngiliz istatistikçi ve biyolog Ronald Fisher tarafından 1936 yılında yayımlanan "The Use of Multiple Measurements in Taxonomic Problems" adlı çalışmada tanıtılmıştır. Veri setinde yer alan ölçümler biyolog Edgar Anderson tarafından toplandığı için bazı kaynaklarda **Anderson's İris veri seti** olarak da adlandırılır. Bu çalışma, iris türleri arasındaki yapısal farklılıkları incelemek amacıyla hazırlanmıştır.

İris veri seti; **İris Setosa, İris Versicolor** ve **İris Virginica** olmak üzere üç farklı çiçek türüne ait verilerden oluşmaktadır. Sette Her türden 50 örnek bulunmakta olup toplamda 150 veri örneği yer alır. Veri setindeki her örnek bir çiçeği temsil eder ve bu çiçeğe ait ölçüm değerleriyle birlikte tür bilgisi (etiket) de bulunur. Bu yönüyle İris veri seti, **etiketli veri setlerine temel örnek** oluşturur.

Görsel 3.8'de veri setinin tablo biçiminde düzenlenmiş hâli görülmektedir. Bu tablonun özellikleri şunlarıdır:

- Her satır bir çiçeğe ait veriyi (örnek) temsil eder.
- Sütunlar çiçeğin ölçülen özelliklerini gösterir.
- Son sütunda çiçeğin ait olduğu tür bilgisi (etiket) yer alır.

Veri seti hem giriş verilerini hem de doğru sonuçları içerir. Böylece yapay zekâ modeli, giriş verileriyle sonuçlar arasındaki ilişkiyi öğrenebilir.

Aynı görselde yer alan diyagram incelendiğinde çiçeğin çanak yaprak (sepal) ve taç yaprak (petal) bölümleri açıkça görülmektedir. İris veri setinde her çiçek için dört temel ölçüm belirlenmiştir. Bu ölçümler şunlardır:

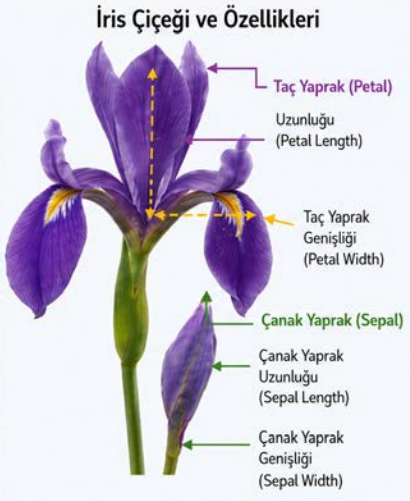
- Çanak yaprak uzunluğu
- Çanak yaprak genişliği
- Taç yaprak uzunluğu
- Taç yaprak genişliği

Ölçümler santimetre cinsinden sayısal değerlerdir ve modelin kullanacağı **giriş verilerini (özellikleri)** oluşturur. Ölçümlerin çiçeğin hangi bölümlerinde yapılacağı da görselde belirtilmiştir (Görsel 3.8).

Ölçülen Özellikler (Bağımsız Değişkenler)

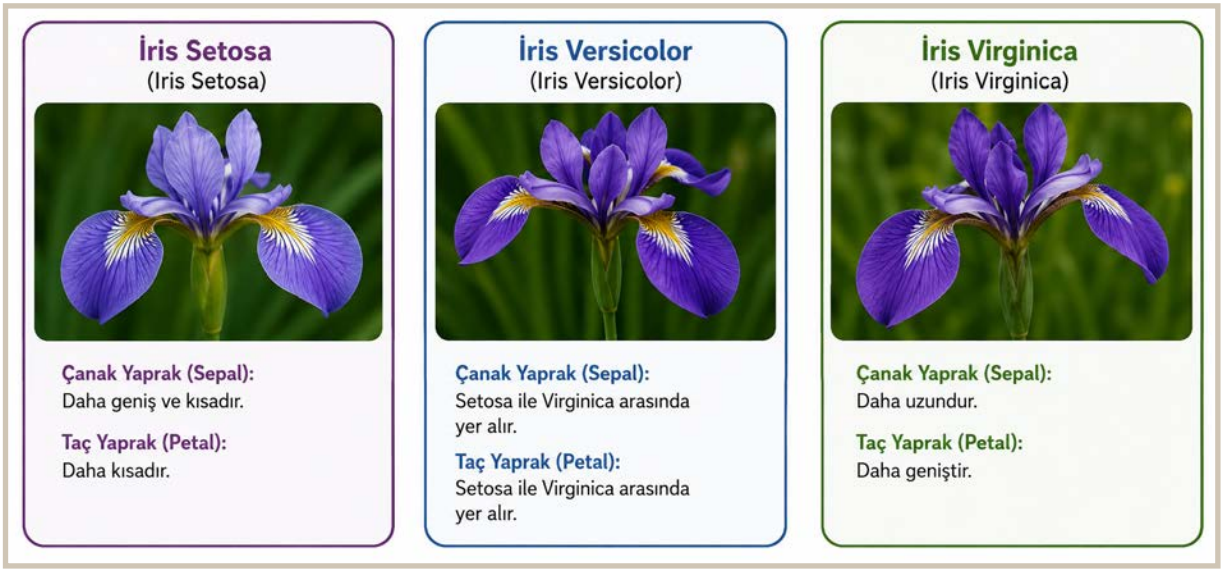
	No	Çanak Yaprak Uzunluğu (sepal length) (cm)	Çanak Yaprak Genişliği (sepal width) (cm)	Taç Yaprak Uzunluğu (petal length) (cm)	Taç Yaprak Genişliği (petal width) (cm)	Hedef (Bağımlı Değişken) Tür (Çiçek Sınıfı)
Örnekler (Gözlemler)	1	5.1	3.5	1.4	0.2	*Iris-setosa*
	2	4.9	3.0	1.4	0.2	*Iris-setosa*
	...	...	...	...	...	...
	50	6.4	2.9	4.3	1.3	*Iris-versicolor*
	...	...	...	...	...	...
	150	5.9	3.0	5.1	1.8	*Iris-virginica*

Bu tabloda her satır bir çiçek örneğini, sütunlar ise çiçeğin ölçülen özelliklerini göstermektedir. Son sütundaki "Tür (Çiçek Sınıfı)" ise her örneğin hangi türe ait olduğunu belirtmektedir.



Görsel 3.8: İris veri setinin yapısı

Görsel 3.9'da veri setinde yer alan üç farklı iris türü görülmektedir. Bu görsellerde türler arasındaki yapısal farklılıklar görülür. Özellikle taç yaprak ve çanak yaprak boyutlarının türlere göre değiştiği dikkat çekmektedir. Bu farklılıklar, modelin sınıflandırma yaparken kullanacağı **ayırt edici özellikleri** oluşturur.



Görsel 3.9: İris türleri

İris veri seti, makine öğrenmesinde **sınıflandırma problemlerini açıklamak** için kullanılan temel örneklerden biridir. Bu tür problemlerde amaç, verilen ölçüm değerlerine bakılarak verilerin belirli sınıflara ayrılmasıdır. Model, etiketli veriler kullanılarak eğitilir ve özelliklerle etiketler arasındaki ilişkiyi öğrenir.

Iris veri seti, basit ve anlaşılır yapısı sayesinde makine öğrenmesine yeni başlayanlar için uygun bir başlangıç örneğidir. Aynı zamanda **K-en Yakın Komşu (KNN), karar ağaçları, destek vektör makineleri (SVM)** gibi birçok sınıflandırma algoritmasının geliştirilmesi ve test edilmesinde yaygın olarak kullanılmaktadır.

Veri seti üzerinde çalışılabilmesi için ilk olarak veri setinin uygun bir ortama yüklenmesi gerekir. Veri seti yüklendikten sonra içerdiği veriler incelenir ve iki temel bileşene ayrılır. Bu bileşenler şunlardır:

- Özellikler (giriş verileri)
- Hedef değişken (etiket)

Özellikler, çiçeğin ölçüm değerlerini, hedef değişkense türünü ifade eder. Bu ayrım yapıldıktan sonra özellikler **X**, hedef değişkense **y** olarak gösterilir.

6. SIRA SİZDE

Aşağıdaki numaralandırılmış kutucuklarda İris veri setiyle ilgili bazı kavramlar verilmiştir. Kutucuk numaralarını kullanarak 1, 2 ve 3. soruları cevaplayınız (Aynı kutucuk numarasını birden fazla cevap için kullanabilirsiniz).

1 Özellik (Feature)	2 Etiket (Label)	3 Veri örneği
4 Sepal uzunluğu	5 Petal genişliği	6 Tür bilgisi
7 Giriş verisi (X)	8 Hedef değişken (y)	9 Ölçüm değeri

1. Modelin giriş olarak kullandığı sayısal değerleri ifade eden kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....
.....

2. Bir çiçeğin hangi türe ait olduğunu gösteren kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....
.....

3. Modelin tahmin etmeye çalıştığı hedef bilgiyi ifade eden kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....
.....

3.1. ÖRNEK UYGULAMA

Adı	İris Veri Setiyle Özellik ve Hedef Değişkenin Ayrılması
Amaç	İris veri setini Python ortamında yüklemek ve veri setindeki özelliklerle (giriş verileri) hedef değişkeni (etiket) inceleyerek birbirinden ayırmak
Süre	20 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook uygulaması, Python programlama dili, scikit-learn kütüphanesi

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- İş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni bir Notebook oluşturunuz ve adını **"Iris_Veri_Seti"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanenin Eklenmesi

Aşağıdaki kodu yazarak İris veri setinin kullanılması için gerekli kütüphaneyi programa dâhil ediniz.

```
from sklearn.datasets import load_iris
```

load_iris fonksiyonu sayesinde hazır bir veri seti Python ortamına kolayca aktarılabilir. Bu adım, veriyle çalışılmaya başlanmadan önce gerekli araçların hazırlanmasını sağlar.

2. Adım: İris Veri Setinin Yüklenmesi

Veri setini programa yükleyiniz.

```
iris = load_iris()
```

Bu adımda İris veri seti programa yüklenir. Veri seti, farklı çiçek türlerine ait ölçüm değerlerini içerir. Modelin çalışabilmesi için verilerin bilgisayar ortamına aktarılması gerekir. Böylelikle veri üzerinde işlem yapılabilir hâle gelinir.

3. Adım: Özelliklerin (Giriş Veri) Ayrılması

Çiçeğe ait ölçüm değerlerini özellikler (X) olarak ayırınız.

```
X = iris.data
print("Özellikler (X):")
print(X[:5])
```

Bu özellikler ölçüleri belirten şu sayısal verilerden oluşur:

- Çanak yaprak uzunluğu
- Çanak yaprak genişliği
- Taç yaprak uzunluğu
- Taç yaprak genişliği

Model, bu bilgileri kullanarak öğrenme gerçekleştirir. Modelin **girdi** olarak kullandığı veriler bu özelliklerdir.

4. Adım: Hedef Değişkenin Ayarlanması

Çiçeğin tür bilgisini hedef değişken olarak ayırınız.

```
y = iris.target
print("\nHedef Değişken (y):)
print(y[ :5 ])
```

Bu adımda çiçeklerin tür bilgisi **hedef değişken (y)** olarak belirlenir. Hedef değişken, modelin tahmin etmeye çalıştığı bilgidir. Model, verilen özelliklere bakarak çiçeğin hangi türe ait olduğunu tahmin etmeye çalışır.

5. Adım: Özellik ve Sınıf İsimlerinin İncelenmesi

Veri setindeki değişkenlerin neyi temsil ettiğini inceleyiniz.

```
print("\nÖzellik İsimleri:")
print(iris.feature_names)
print("\nSınıf İsimleri:")
print(iris.target_names)
```

- Özellik isimleri, ölçüm değerlerinin ne anlama geldiğini gösterir.
- Sınıf isimleriyse çiçek türlerini ifade eder.

Çalışma tamamlandığında elde edilen veriler incelenir ve yorumlanır. Özelliklerle hedef değişken arasındaki fark gözlemlenir. Ardından yapılan çalışmalar kaydedilir. Son olarak oturum güvenli şekilde kapatılır ve çalışma ortamı düzenli bırakılır.

Sonuç

1. **Özelliklerle (X) hedef değişken (y) arasındaki fark nedir? Açıklayınız.**

   .....
   .....
   .....
   .....

2. **Modelin yalnızca özellikleri kullanarak hedef değişkeni tahmin etmesi neden önemlidir? Açıklayınız.**

   .....
   .....
   .....
   .....

Özellikler **(X)** ve hedef değişken **(y)** ayrıldıktan sonra veri seti, bir makine öğrenmesi modeli oluşturmak için uygun hâle gelir. Bu aşamada Python'da yaygın olarak kullanılan **Scikit-learn** kütüphanesi gibi araçlardan yararlanılır. Bu kütüphane; veri işleme, model oluşturma ve değerlendirme süreçlerini kolaylaştıran hazır fonksiyonlar sunar.

Model oluşturma sürecinde veri seti genellikle **eğitim verisi (training set)** ve **test verisi (test set)** olmak üzere ikiye ayrılır. Bu ayrımın yapılmasının temel amacı, modelin yalnızca ezberlemesini engelleyerek **genelleme yeteneğini ölçmektir.**

- **Eğitim verisi,** modelin öğrenme sürecinde kullanılır. Model, bu veriler üzerinden giriş değişkenleriyle hedef değişken arasındaki ilişkiyi öğrenir.
- **Test verisi,** modelin daha önce görmediği verilerden oluşur. Bu veriler kullanılarak modelin ne kadar doğru tahmin yaptığı değerlendirilir.

Süreçte model, eğitim verisi üzerinde öğrendiği kuralları test verisine uygulayarak performansını gösterir. Model test verisi üzerinde de başarılı sonuçlar üretiyorsa bu durum modelin yalnızca veriyi ezberlemediğini, aynı zamanda yeni veriler üzerinde doğru tahminler yapabildiğini gösterir.

Bazı durumlarda veri seti üçe ayrılarak **doğrulama verisi (validation set)** de kullanılabilir. Bu veri seti, modelin parametrelerinin ayarlanması ve en uygun modelin seçilmesi sürecinde kullanılır.

3.2. ÖRNEK UYGULAMA

Adı	Scikit-learn 'le İris Veri Setinde Sınıflandırma Modeli Oluşturma
Amaç	İris veri setini kullanarak K-en Yakın Komşu (KNN) algoritmasıyla basit bir sınıflandırma modeli oluşturmak
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, scikit-learn kütüphanesi

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- Yeni bir Notebook oluşturunuz ve adını **"Iris_Model"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli ve sorumlu şekilde kullanınız.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Öncelikle veri setini yüklemek, veriyi eğitim ve test verisi olarak ayırmak, model oluşturmak ve

modelin doğruluk oranını hesaplamak için gerekli kütüphaneleri programa ekleyiniz.

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
```

Bu adımda kullanılan kütüphaneler, makine öğrenmesi sürecinin temel işlemlerini gerçekleştirmek için gereklidir. load_iris veri setini yüklemek, train_test_split veriyi eğitim ve test olarak ayırmak, KNeighborsClassifier KNN modelini oluşturmak, accuracy_score ise modelin doğruluk oranını hesaplamak için kullanılır.

2. Adım: İris Veri Setinin Yüklenmesi

İris veri seti Python ortamına yükleyiniz.

```
iris = load_iris()
```

İşlemden sonra veri seti iris adlı değişkende saklanır. Bu değişkenin içinde çiçeklere ait ölçüm değerleri, çiçek türleri ve özellik adları yer alır.

3. Adım: Özellikleri ve Hedef Değişkenin Ayrılması

Modelin kullanacağı giriş verilerini ve tahmin etmeye çalışacağı hedef değişkeni birbirinden ayırınız.

```
X = iris.data
y = iris.target
```

Burada X, çiçeklere ait ölçüm değerlerini içerir. Bunlar çanak yaprak uzunluğu, çanak yaprak genişliği, taç yaprak uzunluğu ve taç yaprak genişliğidir. y ise çiçeklerin tür bilgisini içerir. Model, X verilerini kullanarak y değerini tahmin etmeyi öğrenir.

4. Adım: Verinin Eğitim ve Test Verisi Olarak Ayrılması

Modelin öğrenmesi ve daha sonra başarısının ölçülmesi için veri setini iki bölüme ayırınız.

```
X_train, X_test, y_train, y_test = train_test_split(
X, y, test_size=0.2, random_state=42 )
```

Bu kodda veri setinin %80'i eğitim, %20'si test için ayrılır. Eğitim verisi modelin öğrenmesini sağlar. Test verisiyse modelin daha önce görmediği veriler üzerinde ne kadar başarılı olduğunu görmek için kullanılır. random_state=42 değeri, aynı kod çalıştırıldığında aynı veri ayrımının yapılmasını sağlar.

5. Adım: KNN Sınıflandırma Modelinin Oluşturulması

Bu adımda K-en Yakın Komşu algoritmasını kullanarak bir sınıflandırma modeli oluşturunuz.

```
model = KNeighborsClassifier(n_neighbors=3)
```

n_neighbors=3 ifadesi, modelin tahmin yaparken en yakın 3 komşuyu dikkate alacağını gösterir. KNN algoritması, yeni bir verinin hangi sınıfa ait olduğunu belirlerken ona en çok benzeyen örnekleri inceler.

6. Adım: Modelin Eğitilmesi

Eğitim verilerini kullanarak modeli eğitiniz.

```
model.fit(X_train, y_train)
```

Bu adımda model, çiçek ölçümleri ile çiçek türleri arasındaki ilişkiyi öğrenir. Yani hangi ölçüm değerlerinin hangi çiçek türüne daha yakın olduğunu belirlemeye çalışır.

7. Adım: Test Verileri Üzerinde Tahmin Yapılması

Eğitilen modele test verileri üzerinde tahmin yaptırınız.

```
tahminler = model.predict(X_test)
print(tahminler)
```

Bu kod, modelin daha önce görmediği çiçek örnekleri için hangi tür tahmininde bulunduğunu gösterir. Böylece modelin, öğrendiği bilgiyi yeni veriler üzerinde kullanıp kullanamadığı incelenir.

8. Adım: Modelin Doğruluk Oranının Hesaplanması

Modelin yaptığı tahminleri gerçek test etiketleriyle karşılaştırınız.

```
dogruluk = accuracy_score(y_test, tahminler)
print("Modelin doğruluk oranı:", dogruluk)
```

Doğruluk oranı, modelin yaptığı tahminlerin ne kadarının doğru olduğunu gösterir. Örneğin doğruluk oranı 0,96 olarak bulunursa modelin test verilerinin yaklaşık %96'sını doğru sınıflandırdığı söylenebilir.

Sonuç

1. **Eğitim verisiyle test verisi arasındaki fark nedir? Açıklayınız.**

   .....
   .....
   .....
   .....

2. **Modelin doğruluk oranı neyi ifade eder? Açıklayınız.**

   .....
   .....
   .....
   .....

Bu modelde **K-En Yakın Komşu (KNN)** algoritması kullanılmıştır. Model, eğitim verileri üzerinde öğrenme gerçekleştirir ve test verileri üzerinde tahmin yaparak performansı değerlendirilir.

Model oluşturulduktan sonra, elde edilen sonuçların başarısını belirlemek için modelin performansının değerlendirilmesi gerekir. Bu amaçla çeşitli **performans metrikleri** kullanılır.

En temel metriklerden biri doğruluk oranıdır (accuracy). Doğruluk oranı, modelin yaptığı tahminlerin ne kadarının doğru olduğunu gösterir. Ancak bu metrik her zaman tek başına yeterli değildir. Özellikle veri setindeki sınıfların dengesiz olduğu durumlarda modelin başarısını tam olarak yansıtmayabilir.

Bu nedenle daha ayrıntılı bir değerlendirme için farklı metrikler de kullanılır. Bu metrikler şunlardır:

- **Karmaşıklık Matrisi (Confusion Matrix):** Modelin doğru ve yanlış tahminlerini ayrıntılı olarak gösteren tablodur. Bu matris sayesinde hangi sınıfların doğru, hangilerinin yanlış tahmin edildiği analiz edilebilir.
- **Kesinlik (Precision):** Modelin pozitif olarak tahmin ettiği sonuçların ne kadarının gerçekten doğru olduğunu gösterir. Özellikle yanlış pozitiflerin önemli olduğu durumlarda kullanılır.
- **Duyarlılık (Recall):** Gerçek pozitif örneklerin ne kadarının model tarafından doğru şekilde yakalan-

dığını gösterir. Özellikle önemli olan verilerin kaçırılmaması gereken durumlarda kullanılan bir metriktir.

- **F1 Skoru:** Precision ve recall değerlerinin harmonik ortalamasıdır. Modelin genel performansını daha dengeli şekilde değerlendirmek için kullanılır.

Metrikler birlikte değerlendirildiğinde modelin genel doğruluğunun yanında hangi durumlarda başarılı veya yetersiz olduğu da anlaşılabilir. Böylece modelin geliştirilmesi için gerekli iyileştirmeler yapılabilir.

3.3. ÖRNEK UYGULAMA

Adı	Model Performansını Değerlendirme (Confusion Matrix ve Sınıflandırma Raporu)
Amaç	Oluşturulan makine öğrenmesi modelinin performansını doğruluk oranının ötesinde; karmaşıklık matrisi, precision, recall ve F1 skoru gibi metriklerle değerlendirmek
Süre	25 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, scikit-learn kütüphanesi

Hazırlık

- 3.2. örnek uygulamasında oluşturduğunuz modelin bulunduğu Notebook dosyasını açınız.
- Modelin çalıştığını ve **tahminler** değişkeninin oluşturulduğunu kontrol ediniz.
- Çalışma ortamınızı uygulamaya hazır hâle getiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Model performansını değerlendirmek için gerekli metrikleri programa ekleyiniz.

```
from sklearn.metrics import confusion_matrix, classification_report
```

confusion_matrix, modeli ayrıntılı incelemek için, **classification_report** performans metriklerini görmek için kullanılır. Bu kütüphaneler sayesinde modelin doğru tahmin oranının yanında **hangi durumlarda doğru veya yanlış tahmin yaptığı** da analiz edilebilir. Böylece model daha kapsamlı şekilde değerlendirilir.

2. Adım: Karmaşıklık Matrisinin Oluşturulması

Modelin tahminleriyle gerçek değerleri karşılaştırınız.ır.

```
cm = confusion_matrix(y_test, tahminler)
print("Karmaşıklık Matrisi:\n", cm)
```

Karmaşıklık matrisi, modelin doğru tahmin sayısı, karıştırdığı sınıflar ve hata yaptığı durumlar gibi bilgileri tablo hâlinde gösterir. Örneğin model bir çiçeği yanlış sınıflandırdığında bu hata, matris üzerinde açıkça görülebilir. Bu sayede modelin zayıf olduğu noktalar belirlenir.

3. Adım: Sınıflandırma Raporunun Oluşturulması

Modelin performansı daha detaylı inceleyiniz.

```
rapor = classification_report(y_test, tahminler)
print("Sınıflandırma Raporu:\n", rapor)
```

Sınıflandırma raporu şu üç önemli metriği içerir:

Precision (Kesinlik): Modelin yaptığı tahminlerin ne kadarının doğru olduğunu gösterir.

Recall (Duyarlılık): Gerçek verilerin ne kadarının doğru tahmin edildiğini gösterir.

F1 Skoru: Precision ve recall değerlerinin ortalamasıdır.

Bu metrikler sayesinde modelin genel başarısının yanında **her sınıf için ne kadar iyi performans gösterdiği** de anlaşılır.

4. Adım: Sonuçların Yorumlanması

- Hangi sınıflarda doğruluk yüksek?
- Model hangi türleri karıştırıyor?
- Precision ve recall değerleri dengeli mi?

Sonuç

1. **Karmaşıklık matrisi model hakkında hangi bilgileri verir? Açıklayınız.**

.....
.....
.....
.....
.....

2. **Precision ve recall değerleri arasındaki fark nedir? Açıklayınız.**

.....
.....
.....
.....
.....

3. **Bir modelin yalnızca doğruluk oranına bakılarak değerlendirilmesi neden yeterli değildir? Açıklayınız.**

.....
.....
.....
.....
.....

Metrikler sayesinde modelin hangi sınıflarda başarılı olduğu, hangi durumlarda hata yaptığı ayrıntılı olarak analiz edilebilir. Böylece modelin güçlü ve zayıf yönleri belirlenerek gerekli iyileştirmeler yapılabilir.

Bu süreç; veri setinin tanınması, uygun ortama yüklenmesi, özellikler **(X)** ve hedef değişkenin **(y)** ayrılması, modelin oluşturulup eğitilmesi ve performansının değerlendirilmesi adımlarından oluşur. Her adım, makine öğrenmesi sürecinin doğru ve etkili şekilde ilerleyebilmesi için önemlidir. Bu adımlar birlikte değerlendirildiğinde makine öğrenmesinin yalnızca bir model kurma süreci olmadığı; aynı zamanda veri analizi, model geliştirme ve sonuçları yorumlama aşamalarını içeren bütüncül bir süreç olduğu anlaşılmaktadır. Bu yaklaşım, ileri düzey yapay zekâ çalışmalarının anlaşılması için temel bir altyapı oluşturur.

7. SIRA SİZDE

Bir öğrenci, İris veri setini kullanarak çiçek türlerini sınıflandıran bir yapay zekâ modeli geliştirmek istemektedir. Öğrenci öncelikle veri setini inceler, özelliklerle (çiçek yaprak uzunluğu, genişliği vb.) hedef değişkeni (çiçek türü) ayırır. Daha sonra veri setini **eğitim verisi** ve **test verisi** olarak ikiye böler, K-en Yakın Komşu (KNN) algoritmasını kullanarak modelini eğitir. Modelini test ettiğinde %85 doğruluk oranı elde ettiğini görür. Ancak modelin bazı çiçek türlerini karıştırdığını fark eder. Bunun üzerine öğrenci, model performansını daha ayrıntılı incelemek için **karmaşıklık matrisi, precision** ve **recall** değerlerine bakmaya karar verir.

Verilen bilgiler doğrultusunda aşağıdaki soruları cevaplayınız.

1. Öğrencinin veri setini eğitim ve test olarak ayırmasının amacı nedir? Açıklayınız.

.....
.....
.....

2. %85 doğruluk oranı elde edilmesi modelin tamamen başarılı olduğu anlamına gelir mi? Gerekçesiyle açıklayınız.

.....
.....
.....

3. Öğrencinin karmaşıklık matrisi kullanmak istemesinin nedeni nedir? Bu matris model hakkında ne tür bilgiler sağlar? Açıklayınız.

.....
.....
.....

4. Modelin bazı çiçek türlerini karıştırmasının olası nedenleri nelerdir? En az iki neden yazınız.

.....
.....
.....

5. Model performansını artırmak için hangi adımları önerirsiniz? Açıklayınız.

.....
.....
.....

BİLGİ KUTUSU

California Housing veri seti, Amerika Birleşik Devletleri'nin Kaliforniya eyaletinde bulunan farklı bölgelere ait konut bilgilerini içerir. Bu veri seti, makine öğrenmesinde özellikle regresyon problemlerini anlamak için kullanılan temel örneklerden biridir.

Görsel 3.10'da veri setinin tablo biçiminde düzenlendiği görülmektedir. Tabloda her satır bir bölgeyi, sütunlarsa o bölgeye ait özellikleri göstermektedir. Bu nedenle veri setindeki her örnek, belirli bir konut bölgesine ait bilgileri içerir.

Özellikler (Giriş Verileri - X)

No	Medyan Bölge Geliri (ABD Doları)	Konutun Ortalama Yaşı (Yıl)	Ortalama Oda Sayısı (Adet)	Ortalama Yatak Odası Sayısı (Adet)	Bölge Nüfusu (Kişi)	Hane Büyüklüğü (Ortalama Kişi Sayısı)	Enlem (Derece)	Boylam (Derece)	Konutun Medyan Değeri (ABD Doları) (Hedef Değişken)
1	8.32	41.0	6.98	1.02	322.0	2.55	37.88	-122.23	**452,600**
2	8.30	21.0	6.23	0.97	2401.0	2.10	37.86	-122.22	**358,500**
3	11.46	52.0	8.05	1.24	496.0	2.72	37.85	-122.24	**352,100**
4	7.92	52.0	5.74	1.07	558.0	2.80	37.85	-122.24	**341,300**
5	8.71	52.0	6.23	1.14	565.0	2.79	37.86	-122.26	**342,200**
6	9.18	51.0	6.28	1.07	413.0	2.68	37.84	-122.25	**269,700**
7	6.67	22.0	5.41	1.01	1094.0	2.78	37.84	-122.25	**299,200**
...	...	...	...	...	...	...	...	...	...
48	2.28	36.0	4.64	1.02	1005.0	2.28	37.83	-122.26	**115,200**
49	1.91	30.0	4.32	1.03	1118.0	2.21	37.82	-122.27	**112,800**
50	1.78	40.0	4.89	1.12	700.0	2.05	37.82	-122.27	**112,500**
51	2.03	35.0	5.31	1.04	713.0	2.13	37.81	-122.27	**109,300**
52	1.92	23.0	4.58	1.03	1093.0	2.21	37.80	-122.28	**105,200**



Görsel 3.10: California Housing veri setinin yapısı ve özellikleri

Görselde mavi renkle gösterilen sütunlar modelin kullanacağı **özellikleri (giriş verileri - X)** ifade eder. Bu özellikler, bir bölgedeki konutların değerini etkileyebilecek değişkenlerden oluşur.

Veri setinde bulunan temel özellikler şunlardır:

- **MedInc:** Bölgedeki ortalama gelir düzeyi
- **HouseAge:** Bölgedeki konutların ortalama yaşı
- **AveRooms:** Bölgedeki konutların ortalama oda sayısı
- **AveBedrms:** Bölgedeki konutların ortalama yatak odası sayısı
- **Population:** Bölgedeki nüfus
- **AveOccup:** Hane başına düşen ortalama kişi sayısı
- **Latitude:** Bölgenin enlem bilgisi
- **Longitude:** Bölgenin boylam bilgisi

Görselde sağ tarafta yer alan konut ve çevre çizimi, bu özelliklerin gerçek yaşamla ilişkisini gösterir. Örneğin; gelir düzeyi, konut yaşı, oda sayısı, nüfus yoğunluğu ve konum bilgisi konut fiyatını doğrudan etkileyen önemli etkenlerdir.

Tabloda yeşil renkle gösterilen sütunsa **hedef değişkeni (çıktı- y)** ifade eder.

Hedef Değişken

- **MedHouseVal:** Bölgedeki konutların ortalama değeri

Bu veri setinde amaç, bölgeye ait giriş verilerini kullanarak o bölgedeki evlerin ortalama değerini tahmin etmektir. Model, özelliklerle konut değeri arasındaki ilişkiyi öğrenir ve yeni bölgeler için tahmin yapar.

Bu nedenle California Housing veri seti bir **regresyon problemi** için uygundur. Modelin tahmin etmeye çalıştığı çıktı, sınıf etiketi değil **sayısal değerdir.** Bu yönüyle veri seti, yapay zekâ modellerinin gerçek yaşam problemlerinde nasıl kullanılabileceğini gösterir.

Veri Setine Erişim

California Housing veri seti, çeşitli veri bilimi ve makine öğrenmesi kaynaklarında açık veri olarak paylaşılmaktadır. Veri setine erişmeden önce veri setinin adını ve içerdiği özellikleri inceleyerek doğru veri setini seçtiğinizden emin olunuz. Ardından veri setinin orijinal sürümünü indirerek çalışmanızda kullanınız.

Veri setine programlama ortamlarında hazır veri seti araçları kullanılarak da erişilebilir. Bu yöntemler sayesinde veri seti doğrudan çalışma ortamına aktarılabilir ve analiz sürecine hızlı bir şekilde başlanabilir.

Veri setiyle çalışırken CSV dosyasının ve Jupyter Notebook dosyasının aynı klasörde bulunması gerekmektedir.

1. UYGULAMA

Adı	California Housing Veri Setiyle Konut Fiyatı Tahmini
Amaç	California Housing veri setini kullanarak doğrusal regresyon modeli oluşturarak konut fiyatını tahmin etmek.
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek California Housing veri setiyle konut fiyatı tahmini uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Konut fiyatları birçok etkene bağlı olarak değişebilir. Bir konutun değeri; bulunduğu bölgedeki gelir düzeyi, konutun yaşı, oda sayısı, konumu ve nüfus yoğunluğu gibi özelliklerden etkilenir. Ancak aynı özelliklere sahip görünen iki konutun fiyatı; çevre koşulları, ulaşım imkânı, bölgedeki talep veya veri setinde yer almayan başka etkenler nedeniyle farklılık gösterebilir. Bu nedenle yapay zekâ modelinin ürettiği sonuç kesin fiyat değil, geçmiş verilerden hareketle oluşturulan yaklaşık bir tahmindir.

Peki, değişkenler arasındaki ilişkiyi bir bilgisayar sistemi öğrenebilir mi? Bir yapay zekâ modeli, geçmiş verilere bakarak yeni bir konutun yaklaşık değerini tahmin edebilir mi? Bu uygulamada sizden, California Housing veri setini kullanarak bir makine öğrenmesi modeli geliştirmeniz beklenmektedir. Bu süreçte

aşağıdaki işlemler gerçekleştirilecektir.

- Veri setini inceleme
- Özellikler ve hedef değişkeni belirleme
- Model oluşturup eğitme
- Modelin tahmin yapmasını sağlama

Bu uygulama sayesinde, sayısal verilerin bilgisayar ortamındaki analizi ve analizlerin anlamlı tahminlere dönüştürülmesi deneyimlenerek öğrenilecektir. Aynı zamanda, yapay zekâ modellerinin gerçek yaşam problemlerinde kullanımları somut bir örnek üzerinden gözlemlenecektir.

İşlem Basamakları

1. California Housing veri setini inceleyiniz. Veri setinde yer alan özelliklerin konut fiyatını nasıl etkileyebileceğini açıklayınız.
2. Gerekli kütüphaneleri ekleyiniz.

   ```
   from sklearn.datasets import fetch_california_housing
   from sklearn.model_selection import train_test_split
   from sklearn.linear_model import LinearRegression
   from sklearn.metrics import mean_squared_error
   ```
3. Veri setini Jupyter Notebook ortamına yükleyiniz.
4. Veri setini giriş verileri (X) ve hedef değişken (y) olarak ayırınız.
5. Veri setini eğitim ve test verisi olarak ikiye ayırınız.
6. Doğrusal regresyon modelini oluşturunuz.
7. Modeli eğitim verileriyle eğitiniz.
8. Verileri üzerinde tahmin işlemini gerçekleştiriniz.
9. Modelin hata değerini hesaplayınız ve sonucu yorumlayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

3.2.2. Denetimsiz Öğrenme Uygulamaları

Bölümde, daha önce tanıtılan **İris veri seti,** bu kez etiket bilgileri kullanılmadan ele alınacaktır. Veri setinde yer alan çiçek türleri (Setosa, Versicolor, Virginica) model tarafından dikkate alınmayacak; model yalnızca çiçeklere ait ölçüm değerlerini inceleyecektir.

Bu durumda modelin amacı, veriler arasındaki benzerlikleri analiz ederek hangi örneklerin birbirine daha yakın olduğunu belirlemek ve verilen örnekleri aynı gruplar da toplamaktır. Böylece veri setinin iç yapısı, herhangi bir sınıf bilgisi kullanılmadan ortaya çıkarılır.

İşlem için **kümeleme yaklaşımı** kullanılacaktır. Kümeleme sürecinde her veri noktası, sahip olduğu özelliklere göre diğer veri noktalarıyla karşılaştırılır ve benzer veri noktaları aynı kümede toplanır. Bu süreç, özellikle çok boyutlu veri yapılarında örüntülerin ortaya çıkarılmasını sağlar.

Python ortamında **Scikit-learn kütüphanesi** kullanılarak **K-Means kümeleme algoritması** uygulanacaktır. K-Means algoritması, verileri belirlenen küme sayısına göre gruplandırır ve her veri noktasını en yakın küme merkezine atar. Küme merkezleri her adımda güncellenerek verilerin en uygun şekilde gruplanması sağlanır.

İris veri setinde üç farklı çiçek türü bulunmasına rağmen, model bu bilgiyi kullanmadan yalnızca ölçüm değerlerine göre verileri üç kümeye ayırmaya çalışacaktır. Modelin oluşturduğu kümelerin, veri setindeki gerçek yapıyla ne ölçüde örtüştüğü incelenebilecektir. Uygulama sonunda elde edilen kümeler görselleştirilerek analiz edilecektir. Böylece , modelin yalnızca veri özelliklerinden yararlanarak nasıl gruplar oluşturduğu ve bu grupların nasıl yorumlanabileceği gözlemlenecektir.

BİLGİ KUTUSU

K-Means algoritmasında "K" değeri, verilerin kaç gruba ayrılacağını belirler. Bu değer uygun seçilmezse model, verileri gerçekte olduğundan daha az ya da daha fazla gruba ayırabilir. Kümeleme uygulamalarında modelin ürettiği sonuçlar yalnızca kod çıktısı olarak değil görsel ve yorumlanabilir bir yapı olarak değerlendirilmelidir. İris veri setinde üç farklı çiçek türü bulunduğu için uygulamada üç küme seçmek mantıklı bir başlangıçtır ancak gerçek yaşam problemlerinde veri setinin kaç doğal gruba ayrılacağı her zaman önceden bilinmez. Böyle durumlarda grafikler incelenir, kümelerin birbirinden ne kadar ayrıldığına bakılır ve farklı K değerleri denenerek en anlamlı gruplama seçilir.

3.4. ÖRNEK UYGULAMA

Adı	Etiketsiz İris Veri Setiyle K-Means Kümeleme Uygulaması
Amaç	Etiket bilgisi kullanılmadan veri seti üzerinde kümeleme algoritmasını uygulayarak elde edilen kümeleri görselleştirmek ve yorumlamak
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, scikit-learn ve matplotlib kütüphaneleri

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- Yeni bir Notebook oluşturunuz ve adını **"Kumeleme_Iris"** olarak değiştiriniz.
- Gerekli kütüphanelerin kullanılabilir olduğunu kontrol ediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Uygulama için gerekli Python kütüphanelerini programa ekleyiniz.

```
from sklearn.datasets import load_iris
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
```

Uygulamada çiçek türleri (etiketler) kullanılmayacaktır. Amaç, modelin hazır etiketleri kullanmadan veriler arasındaki ilişkileri kendi kendine keşfetmesini sağlamaktır.

load_iris veri setini çağırmak için, **KMeans** kümeleme işlemini yapmak için, **matplotlib** ise sonuçları grafik üzerinde göstermek için kullanılır.

Bu sayede hem veri işleme hem de görselleştirme için gerekli araçlar hazırlanmış olur.

2. Adım: İris Veri Setinin Yüklenmesi

İris veri setini Python ortamına yükleyiniz. Veri setinde çiçeklere ait ölçüm değerlerini bulunuz.

```
iris = load_iris()
```

Uygulamada çiçek türleri (etiketler) kullanılmayacaktır. Amaç, modelin hazır etiketleri kullanmadan veriler arasındaki ilişkileri kendi kendine keşfetmesini sağlamaktır.

3. Adım: Etiket Bilgisi Kullanılmadan Özelliklerin Ayrılması

Yalnızca çiçeklere ait ölçüm değerlerini alınız (X = iris.data).

```
X = iris.data
```

Çiçek türünü gösteren hedef değişken kullanılmaz. Böylece veri seti **etiketsiz hâle getirilmiş** olur ve denetimsiz öğrenme için uygun bir yapı oluşturulur. Model artık yalnızca verilerin benzerliklerine bakarak gruplama yapacaktır.

4. Adım: K-Means Modelinin Oluşturulması

K-Means kümeleme algoritmasını tanımlayınız. Modelin verileri 3 kümeye ayırmasını isteyiniz.

```
model = KMeans(n_clusters=3, random_state=42)
```

K-Means algoritması, veriler arasındaki benzerlikleri kullanarak gruplar oluşturur. Benzer özelliklere sahip veriler aynı kümeye yerleştirilir.

5. Adım: Modelin Veriler Üzerinde Çalıştırılması

Modeli veriler üzerinde çalıştırınız.

```
model.fit(X)
```

Model, veriler arasındaki benzerlikleri analiz eder ve hangi verinin hangi kümeye ait olduğunu belirler.

Sonuç olarak model herhangi bir etiket bilgisi kullanmaz, tamamen kendi hesaplamalarına göre gruplandırma yapar.

6. Adım: Küme Etiketlerinin Elde Edilmesi

Modelin oluşturduğu kümeleri alınız . Her veri örneği için bir küme numarası belirleyiniz (0,1 veya 2 gibi).

```
kumeler = model.labels_
print(kumeler)
```

Bu sayılar, her çiçeğin hangi gruba ait olduğunu gösterir.

7. Adım: Kümelerin Görselleştirilmesi

Verileri grafik üzerinde gösteriniz.

```
plt.scatter(X[:, 2], X[:, 3], c=kumeler)
```

```
plt.xlabel("Taç yaprak uzunluğu")
plt.ylabel("Taç yaprak genişliği")
plt.title("K-Means ile Iris Veri Seti Kümeleme")
plt.show()
```

Her nokta bir çiçeği temsil eder ve renkler farklı kümeleri gösterir. Örneğin aynı renkteki noktalar aynı gruba aittir. Bu görselleştirme sayesinde, modelin verileri nasıl gruplandırdığı daha kolay anlaşılır.

8. Adım: Sonuçların Yorumlanması

Grafikte oluşan kümeleri inceleyiniz. Benzer özelliklere sahip çiçeklerin aynı bölgelerde toplandığını gözlemleyiniz. Şu soruların yanıtlarını düşününüz:

- Kümeler birbirinden net şekilde ayrılıyor mu?
- Bazı kümeler birbirine yakın görünüyor mu?
- Model etiket bilgisi olmadan verileri nasıl gruplandırmış olabilir?
- Denetimli öğrenmeyle denetimsiz öğrenme arasındaki fark bu uygulamada nasıl görülmektedir?

Sonuç

1. **Uygulamada hedef değişken neden kullanılmamıştır? Açıklayınız.**

.....
.....
.....
.....
.....
.....

2. **K-Means algoritması verileri hangi mantıkla gruplandırır? Açıklayınız.**

.....
.....
.....
.....
.....
.....

3. **Grafikte oluşan kümeler birbirinden net şekilde ayrılıyor mu? Açıklayınız.**

.....
.....
.....
.....
.....
.....

4. **Denetimli öğrenmeyle denetimsiz öğrenme arasındaki fark bu uygulamada nasıl görülmektedir? Açıklayınız.**

.....
.....
.....
.....
.....
.....

BİLGİ KUTUSU

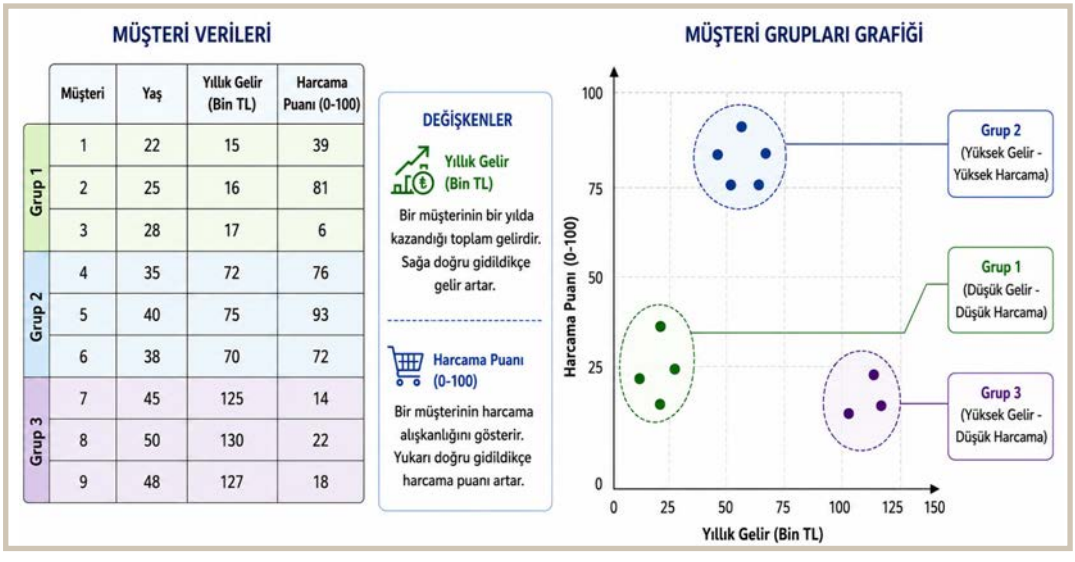
Müşteri veri setleri, işletmelerin müşterilerini daha iyi tanıyabilmesi, alışveriş davranışlarını analiz edebilmesi ve daha etkili pazarlama stratejileri geliştirebilmesi için kullanılan önemli veri kaynaklarıdır. Veri setleri, müşterilerin demografik özellikleriyle harcama alışkanlıklarını birlikte inceleme imkânı sunar.

Uygulamada kullanılan **Mall Customers** veri seti, bir alışveriş merkezine gelen müşterilere ait temel bilgileri içeren ve makine öğrenmesinde özellikle **kümeleme (clustering)** çalışmalarında yaygın olarak kullanılan veri setidir.

Görsel 3.11'deki müşteri segmentasyonunda veri setinin tablo biçiminde düzenlendiği görülmektedir. Bu tabloda her satır bir **müşteriyi (örnek)** temsil eder. Her sütun müşteriye ait bir **özelliği (giriş verisi-X)** gösterir.

Veri seti yaklaşık **200 müşteri kaydı** içermektedir ve şu değişkenlerden oluşur:

- **CustomerID:** Her müşteriye ait benzersiz kimlik numarasıdır (Genellikle modelde kullanılmaz.).
- **Gender (Cinsiyet)**: Müşterinin cinsiyetini belirtir (kategorik veri).
- **Age (Yaş):** Müşterinin yaşını belirtir (sayısal veri).
- **Annual Income (Yıllık Gelir):** Müşterinin yıllık gelirini belirtir (bin dolar cinsinden).
- **Spending Score (Harcama Puanı):** Müşterinin alışveriş davranışını gösteren puandır.



Görsel 3.11: Müşteri segmentasyonu

Görselde özellikle **yıllık gelir ve harcama puanı** değişkenlerinin öne çıktığı görülmektedir. Bu iki değişken, müşterilerin davranışlarının analizi için en belirleyici özelliklerdir.

Grafik incelendiğinde müşterilerin yıllık gelir (x ekseni) ve harcama puanı (y ekseni) değerlerine göre belirli bölgelerde yoğunlaştığı görülür. Bu durum, veri içinde doğal olarak oluşan **müşteri segmentlerinin** varlığını gösterir.

Bu dağılıma göre müşteriler genel olarak şu gruplara ayrılabilir:

- Düşük gelir-yüksek harcama
- Yüksek gelir-yüksek harcama

- Yüksek gelir-düşük harcama
- Düşük gelir- düşük harcama

Müşteri grupları veri setinde hazır olarak verilmez. Model, bu grupları **veriler arasındaki benzerlikleri analiz ederek kendisi oluşturur.** Veri setiyle müşterileri benzer özelliklerine göre gruplandırmak ve müşteri segmentleri oluşturmak amaçlanır.

Bu işlem işletmeler için oldukça önemlidir. Müşteri segmentleri belirlendiğinde kolaylıkla yapılabilecek işlemler şunlardır:

- Hedefli reklam kampanyaları hazırlanabilir.
- Kişiselleştirilmiş öneriler sunulabilir.
- Müşteri davranışları daha doğru analiz edilebilir.
- Pazarlama stratejileri daha verimli hâle getirilebilir.

Bu veri setinde müşterilerin ait olduğu **hazır bir grup (etiket) bilgisi bulunmaz.**

Bundan dolayı veri seti, **etiketsiz veri seti ve denetimsiz öğrenme uygulamaları** için uygundur.

Model, müşterileri **kendi kendine gruplandırır (kümeleme)** ve veri içindeki örüntüleri ortaya çıkarır.

Kümeleme işlemini daha anlaşılır hâle getirmek için genellikle tüm değişkenler yerine şu iki değişken kullanılır:

- **Annual Income (yıllık gelir)**
- **Spending Score (harcama puanı)**

Bu iki değişken sayesinde veri iki boyutlu düzlemde kolayca görselleştirilebilir ve müşteri grupları daha net analiz edilebilir.

Mall Customers veri seti, müşteri özelliklerini içeren açık kaynaklı veri setleri arasında yer almaktadır. Veri setini güvenilir veri paylaşım kaynaklarında araştırarak indiriniz ve içeriğini inceleyiniz. Veri setiyle çalışırken veri dosyasının (.csv) ve Jupyter Notebook dosyasının aynı klasörde bulunması, dosyanın sorunsuz şekilde okunmasını sağlar.

2. UYGULAMA

Adı	Müşteri Veri Setiyle Kümeleme (Müşteri Segmentasyonu)
Amaç	Müşteri veri setini kullanarak K-Means algoritmasıyla müşteri segmentleri oluşturarak elde edilen kümeleri yorumlamak
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek müşteri veri seti üzerinde K-Means kümeleme uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

İşletmeler, çok sayıda müşteriye aynı anda hizmet sunmak zorundadır. Ancak her müşterinin ihtiyaçları, alışveriş alışkanlıkları ve harcama davranışları farklıdır. Bu nedenle işletmeler için "Hangi müşteriler birbirine benzer ve aynı stratejiyle ele alınabilir?" sorusunun yanıtı önemlidir. Kümeleme sonucunda model yalnızca benzer müşterileri aynı grupta toplar; bu grupların ne anlama geldiğini yorumlamak kullanıcıya düşer. Aynı müşteri verisi farklı değişkenlerle incelendiğinde farklı kümeler ortaya çıkabilir. Örneğin yalnızca gelir ve harcama puanı kullanıldığında oluşan gruplar, yaş veya ziyaret sıklığı eklendiğinde değişebilir.

Oluşturulan gruplar sayesinde işletmeler şu işlemleri gerçekleştirebilir:

- Farklı müşteri segmentlerine uygun kampanyalar geliştirmek
- Pazarlama stratejilerini daha verimli hâle getirmek
- Müşteri davranışlarını daha doğru yorumlayabilmek

İşlem Basamakları

1. Veri setinde bulunan değişkenleri inceleyiniz ve geliriyle harcama puanının müşteri davranışlarını nasıl etkileyebileceğini açıklayınız.
2. Gerekli kütüphaneleri programa ekleyiniz.
3. Veri setini Jupyter Notebook ortamına yükleyiniz
4. Veri setinden yıllık gelir ve harcama puanı değişkenlerini seçiniz.
5. Müşterileri 5 gruba ayıracak K-Means modelini oluşturunuz.
6. Modeli veri seti üzerinde çalıştırınız.
7. Her müşterinin ait olduğu kümeyi belirleyiniz.
8. Kümeleme sonuçlarını grafik üzerinde gösteriniz.
9. Elde edilen müşteri gruplarını inceleyerek yorumlayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

3.2.3. Yarı Denetimli Öğrenme Uygulamaları

Yarı denetimli öğrenme uygulamasında daha önce kullanılan **İris veri seti**, bu kez **kısmen etiketli** bir yapıyla ele alınacaktır. Veri setindeki bazı çiçek örneklerinin tür bilgisi korunacak, bazı örneklerinse tür bilgisi modelden gizlenecektir. Böylece modelin, az sayıdaki etiketli veriden yararlanarak etiketi bilinmeyen örnekler hakkında tahmin yapması sağlanacaktır.

Bu yaklaşım, gerçek yaşamda sık karşılaşılan veri durumlarını temsil eder. Birçok alanda veri toplamak kolay olsa da verilerin uzmanlar tarafından tek tek etiketlenmesi zaman alıcı ve maliyetlidir. Örneğin tıbbi görüntülerin sınıflandırılması, sınav kâğıtlarının puanlanması veya ürün yorumlarının etiketlenmesi yoğun emek gerektirir. Buna benzer durumlarda az sayıdaki etiketli veri, çok sayıdaki etiketsiz verinin anlamlandırılmasında rehber olarak kullanılabilir.

Uygulamada **self-training (kendi kendine öğrenme)** yaklaşımından yararlanılacaktır. Model önce tür bilgisi bilinen çiçek örnekleriyle eğitilecek, ardından tür bilgisi gizlenen örnekler üzerinde tahmin yapacaktır. Modelin yüksek güvenle yaptığı tahminler, geçici etiketler olarak değerlendirilerek veri setine eklenecektir. Bu

geçici etiketler **sözde etiket** olarak adlandırılır. Süreç ilerledikçe model, yalnızca başlangıçtaki etiketli veriyle sınırlı kalmaz, güvenilir tahminlerle genişletilmiş veri setinden de yararlanır. Böylece yeniden eğitilerek daha fazla örnek üzerinden öğrenme gerçekleştirir. Bu döngü, modelin veri yapısını daha doğru şekilde öğrenmesini sağlar.

Scikit-learn kütüphanesi kullanılarak yarı denetimli öğrenme süreci oluşturulacaktır. Modelin performansı, yalnızca etiketli verilerle eğitilen bir modelle karşılaştırılarak değerlendirilecektir. Böylece etiketsiz verilerin öğrenme sürecine katkısı somut olarak incelenecektir. Uygulama sonunda, kısmen etiketli bir veri setiyle çalışmanın model başarısı üzerindeki etkisi gözlemlenebilir. Etiketli veri, etiketsiz veri, sözde etiketleme, tahmin ve yeniden eğitim kavramları arasındaki ilişki uygulama üzerinden değerlendirilebilir.

8. SIRA SİZDE

Aşağıda numaralandırılmış kutucuklarda **denetimsiz öğrenme ve İris veri setiyle ilgili kavramlar** verilmiştir. Kutucuk numaralarını kullanarak 1, 2 ve 3. soruları cevaplayınız (Aynı kutucuk numarasını birden fazla cevap için kullanabilirsiniz.).

1 Özellik (Feature)	2 Etiket (Label)	3 Küme
4 Çanak yaprak uzunluğu	5 Tür bilgisi	6 Benzerlik
7 Giriş verisi (X)	8 Hedef değişken (y)	9 Kümeleme
10 Ölçüm değeri	11 K-Means algoritması	12 Veri grubu

1. Modelin denetimsiz öğrenme sürecinde kullandığı giriş verilerini ifade eden kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....

2. Denetimsiz öğrenmede kullanılmayan, yani modelin erişemediği bilgi türünü ifade eden kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....

3. Modelin verileri benzerliklerine göre gruplandırması sonucunda ortaya çıkan yapıyı ifade eden kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....

3.5. ÖRNEK UYGULAMA

Adı	Kısmen Etiketli İris Veri Setiyle Yarı Denetimli Öğrenme (Self-Training) Uygulaması
Amaç	Kısmen etiketli bir veri seti kullanarak yarı denetimli öğrenme yaklaşımını uygulayarak etiketsiz verilerin modele katkısını gözlemlemek ve model performansını değerlendirmek
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, scikit-learn ve matplotlib kütüphaneleri

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- Yeni bir Notebook oluşturunuz ve adını **"Yari_Denetimli_Iris"** olarak değiştiriniz.
- Gerekli kütüphanelerin kullanılabilir olduğunu kontrol ediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

BİLGİ NOTU

Denetimli öğrenmede model tamamen **etiketli veriyle** eğitilirken denetimsiz öğrenmede yalnızca **etiketsiz verilerden** yararlanır. Ancak gerçek yaşamda veri setlerinin büyük bir kısmı etiketsiz olabilir.

Bu uygulamada İris veri seti **kısmen etiketli bir veri seti** hâline getirilecektir. Veri setindeki örneklerin bir kısmı etiketli, bir kısmı etiketsiz olacaktır. Modelin veriler üzerinde tahmin yapması sağlanacaktır. Gelinen süreçte **yarı denetimli öğrenme yaklaşımlarından biri olan self-training (kendi kendine öğrenme)** yöntemi kullanılacaktır. Model önce etiketli verilerle eğitilecek, daha sonra etiketsiz veriler için tahminler yapacak ve güvenilir tahminleri kullanarak kendini yeniden eğitecektir.

Bu uygulamada İris veri seti aşağıdaki şekilde kullanılacaktır:

- Veri setinin bir kısmı → **Etiketli veri**
- Veri setinin bir kısmı → **Etiketsiz veri**

Model yalnızca aşağıdaki özellikleri kullanacaktır:

- Çanak yaprak uzunluğu
- Çanak yaprak genişliği
- Taç yaprak uzunluğu
- Taç yaprak genişliği

Etiketsiz verilerde çiçek türü bilgisi gizlenecek ve model bu veriler için tahmin yapacaktır.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
import numpy as np
```

Bu adımda uygulamanın çalışması için gerekli Python kütüphaneleri programa eklenir. **load_iris** komutu İris veri setinin kullanılmasını sağlar. **train_test_split** komutu veri setini farklı bölümlere ayırmak için kullanılır. **RandomForestClassifier** komutu sınıflandırma modeli oluşturmak için kullanılır. **numpy** kütüphanesiyse verileri birleştirme ve düzenleme işlemlerinde kullanılır. Böylece veri setini yükleme, modeli oluşturma ve yarı denetimli öğrenme sürecini yürütme için gerekli hazırlık yapılmış olur.

2. Adım: İris Veri Setinin Yüklenmesi

```
iris = load_iris()
X = iris.data
y = iris.target
```

Bu adımda İris veri seti programa yüklenir. Veri setinde çiçeklere ait ölçüm değerleri ve çiçek türleri bulunur. **X = iris.data** satırı, modelin kullanacağı giriş verilerini oluşturur. Bu giriş verileri; çanak yaprak uzunluğu, çanak yaprak genişliği, taç yaprak uzunluğu , taç yaprak genişliği gibi ölçüm değerlerinden oluşur. **y = iris.target** satırıysa çiçek türlerini gösteren hedef değişkeni oluşturur. Böylece modelin hangi verilerle çalışacağı ve hangi bilgiyi tahmin etmeye çalışacağı belirlenmiş olur.

3. Adım: Veri Setinin Kısmen Etiketli Hâle Getirilmesi

```
X_labeled, X_unlabeled, y_labeled, y_unlabeled = train_test_split(X, y, test_size=0.7, random_state=42)
y_unlabeled[ : ] = -1
```

Bu adımda veri seti iki bölüme ayrılır. Veri setinin küçük bir kısmı etiketli veri olarak bırakılır, büyük bir kısmıysa etiketsiz veri hâline getirilir. **train_test_split** komutu bu ayırma işlemini yapar. **test_size=0.7** ifadesi, verinin büyük bölümünün etiketsiz veri olarak ayrılacağını gösterir. **y_unlabeled[:] = -1** satırıysa bu bölümdeki çiçek türü bilgisinin gizlendiğini ifade eder. Böylece model, bazı verilerin doğru tür bilgisini görebilirken bazı verilerin tür bilgisini bilmeden tahmin yapmaya çalışır. Bu yapı, gerçek yaşamda az sayıda etiketli ve çok sayıda etiketsiz veriyle karşılaşılan durumları temsil eder.

4. Adım: Modelin Oluşturulması Ve Eğitilmesi

```
model = RandomForestClassifier()
model.fit(X_labeled, y_labeled)
```

Bu adımda **RandomForestClassifie**r kullanılarak bir sınıflandırma modeli oluşturulur. Model, yalnızca etiketli verilerle eğitilir. Yani model, çiçek ölçümleriyle bu ölçümlere karşılık gelen doğru çiçek türleri arasındaki ilişkiyi öğrenmeye çalışır. Bu aşamada etiketsiz veriler henüz eğitimde kullanılmaz. Modelin ilk öğrenmesi, yalnızca doğru cevabı bilinen örnekler üzerinden gerçekleşir. Bundan dolayı bu adım, yarı denetimli öğrenme sürecinin başlangıç noktasıdır.

5. Adım: Self-training (Kendi Kendine Öğrenme) Döngüsünün Uygulanması

```
for i in range(3):
    tahminler = model.predict(X_unlabeled)
    X_labeled = np.vstack((X_labeled, X_unlabeled[:20]))
    y_labeled = np.hstack((y_labeled, tahminler[:20]))
    X_unlabeled = X_unlabeled[20:]
    model.fit(X_labeled, y_labeled)
```

Bu adımda model, daha önce görmediği etiketsiz veriler için tahmin yapar. Modelin yaptığı bu tahminler, geçici etiketler gibi kabul edilir. Bu işleme **sözde etiketleme** denir. Kodda her döngüde etiketsiz verilerin ilk 20 tanesi etiketli veri grubuna eklenir. Veriler, modelin tahmin ettiği etiketlerle birlikte yeni eğitim verisi hâline gelir. Daha sonra model genişletilmiş veriyle yeniden eğitilir. Döngü birkaç kez tekrarlandığı için model her seferinde daha fazla veriyle çalışır. Bu süreç, modelin kendi tahminlerinden yararlanarak öğrenmesini sağlar. Ancak burada dikkat edilmesi gereken nokta, modelin yanlış tahminleri de eğitim verisine ekleyebileceğidir. Bundan dolayı self-training yönteminde tahminlerin güvenilirliği önemlidir.

6. Adım: Model Performansının Gözlemlenmesi

```
print("Eğitim veri sayısı:", len(X_labeled))
```

Bu adımda modelin eğitimde kullandığı veri sayısı ekrana yazdırılır. Başlangıçta model yalnızca az sayıdaki etiketli veriyle eğitilmiştir. Self-training döngüsünden sonraysa etiketsiz verilerin bir kısmı tahmin edilerek eğitim verisine eklenmiştir. Bu nedenle eğitim veri sayısı artar. Öğrenci bu değişimi gözlemleyerek etiketsiz verilerin modele nasıl katkı sağladığını fark eder. Eğitim verisinin artması modelin daha fazla örnekten öğrenmesini sağlayabilir. Ancak eklenen tahminler yanlışsa model hatalı bilgilerden de öğrenebilir. Bu nedenle sonuçlar yorumlanırken modelin fazla veriyle eğitilmesinin yanında bu verilerin doğruluğu da dikkate alınmalıdır.

Sonuç

1. **Bu uygulamada veri seti neden kısmen etiketli hâle getirilmiştir? Açıklayınız.**

   .....
   .....
   .....

2. **Self-training yöntemi hangi mantıkla çalışır? Açıklayınız.**

   .....
   .....
   .....

3. **Etiketsiz veriler model performansını nasıl etkiler? Açıklayınız.**

   .....
   .....
   .....

4. **Yarı denetimli öğrenmenin denetimli öğrenmeye göre avantajı nedir? Açıklayınız.**

   .....
   .....
   .....

Bu uygulamada İris veri seti, **kısmen etiketli** hâle getirilmiştir. Veri setindeki çiçeklerin yalnızca bir bölümünün tür bilgisi modele verilmiş, geri kalan örneklerse etiket bilgisi olmadan kullanılmıştır. Başlangıç aşamasında model, yalnızca etiketli verilerle eğitilmiştir. Bu durumda modelin öğrendiği bilgi sınırlı sayıda örneğe dayandığı için, çiçek türleri arasındaki farkları tam olarak ayırt edememesi beklenir.

Daha sonra **self-training yaklaşım**ı uygulanarak etiketsiz veriler öğrenme sürecine dâhil edilmiştir. Bu süreçte model şu adımları izler:

- Etiketli veriler kullanılarak başlangıç modeli oluşturulur.
- Model, etiketsiz veriler üzerinde tahminler yapar.
- Yüksek güven içeren tahminler sözde etiket olarak kabul edilir.
- Veriler etiketli veri grubuna eklenir.
- Model, genişleyen veri setiyle yeniden eğitilir.

Bu döngü sayesinde model, başlangıçta sahip olduğu sınırlı veriyle yetinmeyip daha fazla örnek üzerinden öğrenme imkânı bulur. Yalnızca etiketli veri kullanıldığında modelin öğrenebileceği örnek sayısı azdır. Bu durum, özellikle birbirine benzer ölçüm değerlerine sahip çiçek türlerinin ayırt edilmesini zorlaştırabilir. Yarı denetimli öğrenme uygulandığında model, etiketsiz verilerden elde edilen sözde etiketler sayesinde daha geniş bir veri setiyle çalışır.

Örneğin model başlangıçta 45 etiketli veriyle eğitilmişse, self-training sürecinden sonra bu sayı artar. Model daha fazla örnek üzerinden öğrenme gerçekleştirdiği için özellikle sınır durumlarda (benzer özelliklere sahip örneklerde) daha başarılı tahminler yapabilir.

Süreçte etiketsiz verilerin modele sağladığı katkıları şunlardır:

- Eğitim veri miktarını artırır.
- Veri içerisindeki örüntülerin daha belirgin hâle gelmesini sağlar.
- Az etiketli veriyle öğrenmeyi destekler.
- Modelin genelleme yeteneğini artırabilir.

Sözde etiketler modelin kendi tahminlerine dayandığı için kesin doğru kabul edilmez. Model hatalı tahminlerde bulunursa, bu hatalar eğitim sürecine dâhil olup modelin performansını olumsuz etkileyebilir. Bundan dolayı self-training sürecinde, modele eklenecek verilerin güvenilirliği önemlidir. Tahminlerin dikkatli değerlendirilmesi, modelin doğru yönde gelişmesini sağlamak açısından önemli bir adımdır.

BİLGİ KUTUSU

Film yorumları veri setleri, kullanıcıların izledikleri filmler hakkında yazdıkları değerlendirme metinlerinden oluşur. Bu veri setleri, yapay zekâ sistemlerinin **duygu analizi** yapabilmesini sağlayan önemli veri kaynaklarıdır. Özellikle sosyal medya, e-ticaret ve içerik platformlarında kullanıcı geri bildirimlerinin anlaşılmasında yaygın olarak kullanılır.

Uygulamada kullanılan **film yorumları veri seti,** metin tabanlı bir veri setidir ve her satır bir kullanıcı yorumunu temsil eder.

Görsel 3.12'de veri setinin iki temel bileşenden oluştuğu görülmektedir. Bileşenler şunlardır:

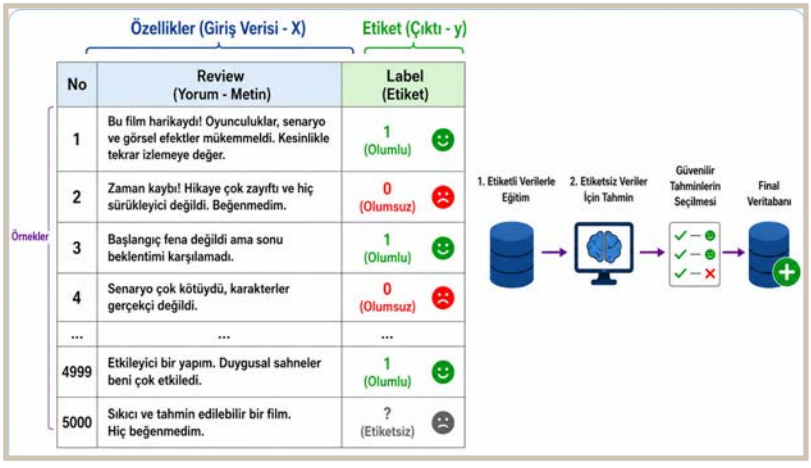
- **Review (Yorum-Giriş Verisi X):**

Kullanıcının film hakkında yazdığı metin

- **Label (Etiket-Çıktı y):**

Yorumun duygu durumunu ifade eder

- 1 ⟶ Olumlu
- 0 ⟶ Olumsuz
- Boş (?) ⟶ Etiketsiz veri



Görsel 3.12: Yarı denetimli öğrenmeyle duygu analizi

Bu yapı, modelin metinle duygu durumu arasındaki ilişkiyi öğrenmesini sağlar.

Veri setinin tamamı etiketli değildir.

- Bir kısmı **etiketlidir** (yorum + doğru duygu bilgisi).
- Bir kısmı **etiketsizdir** (yalnızca yorum metni).

Bundan dolayı veri seti, **kısmen etiketli veri seti ve yarı denetimli öğrenme** için uygundur.

Veri setinin kullanım süreci şu adımlarla gerçekleşir:

1. Model, yalnızca **etiketli yorumlarla** eğitilir.
2. Eğitilen model, **etiketsiz yorumlar** üzerinde tahmin yapar.
3. Yüksek güvene sahip tahminler **sözde etiket** olarak seçilir.
4. Bu veriler, etiketli veri grubuna eklenir.
5. Model, genişletilmiş veri setiyle **yeniden eğitilir.**

Döngü, modelin kendi tahminlerini kullanarak kendini geliştirmesini sağlar ve **self-training** olarak adlandırılır.

Bu veri seti kullanılarak yapılabilecek işlemler şunlardır:

- Film yorumlarının **duygu analizi**
- Etiketsiz verileri **otomatik etiketleme**
- Model performansını daha fazla veriyle **iyileştirme**
- Sosyal medya, müşteri yorumları gibi gerçek problemlere **uyarlama**

Modelleme sürecinde tüm değişkenler kullanılmaz. Genellikle yalnızca **Review ve Label** kullanılır.

Metin verisi doğrudan işlenemeyeceği için model tarafından kullanılmadan **önce sayısal forma dönüştürülür** (örneğin kelime frekansları veya vektör temsilleri kullanılarak).

Film yorumları veri seti, doğal dil işleme ve duygu analizi çalışmalarında yaygın olarak kullanılan açık kaynaklı veri setleri arasında yer almaktadır. Veri setini güvenilir veri paylaşım kaynaklarında araştırarak indiriniz ve içeriğini inceleyiniz. Veri setiyle çalışırken CSV dosyasının ve Jupyter Notebook dosyasının aynı klasörde bulunması gerekmektedir.

3. UYGULAMA

Adı	Film Yorumları Veri Setiyle Yarı Denetimli Öğrenme (Duygu Analizi)
Amaç	Film yorumları veri seti üzerinde yarı denetimli öğrenme yaklaşımını uygulamak
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek film yorumları veri seti üzerinde yarı denetimli öğrenme uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.ız.

Ön Bilgi

Film yorumları, kullanıcıların bir içerik hakkındaki düşüncelerini yansıtan metin verileridir. Uygulamada, yorumların bilgisayar tarafından işlenebilir hâle getirilmesi ve model çıktılarının güvenilirlik açısından değerlendirilmesi amaçlanmaktadır. Modelin ürettiği tahminler incelenerek hangi sonuçların öğrenme sürecine katkı sağlayabileceği gözlemlenecektir.

Metin verilerinde duygu her zaman açık biçimde anlaşılmayabilir. Bir yorumda olumlu ve olumsuz ifadeler birlikte bulunabilir; ironi, abartı veya bağlam eksikliği modelin tahminini zorlaştırabilir. Yarı denetimli öğrenmede modelin kendi tahminlerini tekrar kullanması yararlı olsa da hatalı tahminler sonraki öğrenme sürecini olumsuz etkileyebilir. Bu nedenle uygulamada öncelik yalnızca daha fazla verinin değil güvenilir tahminlerin modele eklenmesidir.

İşlem Basamakları

1. Veri setinde bulunan yorumları inceleyiniz. Yorumların bulunduğu sütunu ve etiket bilgisinin anlamını belirleyiniz. Etiketli ve etiketsiz verilerin model açısından ne ifade ettiğini açıklayınız.
2. Veri setini işlemek, metinleri sayısal forma dönüştürmek ve model oluşturmak için gerekli kütüphaneleri ekleyiniz.
3. Veri setini Jupyter Notebook ortamına yükleyiniz ve ilk satırları inceleyiniz.
4. Veri setindeki olumlu ve olumsuz etiketleri sayısal değerlere dönüştürünüz.

5. Veri setini eğitim ve test verisi olarak ayırınız.
6. Eğitim verisinin bir bölümünü etiketli, kalan bölümünü etiketsiz olacak şekilde düzenleyiniz.
7. Yorumları modelin kullanabileceği sayısal verilere dönüştürünüz.
8. İlk modeli yalnızca etiketli veriler kullanarak oluşturunuz ve eğitiniz.
9. Modeli kullanarak etiketsiz verilerin etiketlerini tahmin ediniz.
10. Güvenilir tahminleri seçiniz ve bu verileri etiketli veri setine ekleyiniz.
11. Genişletilmiş veri setiyle modeli yeniden eğitiniz.
12. Etiketsiz verilerin model performansına etkisini değerlendiriniz.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

3.2.4. Pekiştirmeli Öğrenme Uygulamaları

Uygulamada, bir ajanın basit bir **grid (ızgara) ortamı** içerisinde nasıl hareket ettiği incelenecektir. Ajan, başlangıç noktasından hareket ederek hedefe ulaşmaya çalışır.

İlk aşamada ajan, ortam hakkında herhangi bir bilgiye sahip olmadığı için hareketlerini **rastgele** gerçekleştirir. Bundan dolayı ajanın izlediği yol düzensizdir ve hedefe ulaşma davranışı henüz oluşmamıştır.

Bu aşamada ajanın ortamla ilk etkileşimi ve başlangıç davranışları gözlemlenecektir.

3.6. ÖRNEK UYGULAMA

Adı	Grid Ortamında Ajan Hareketlerinin Gözlemlenmesi
Amaç	Basit bir pekiştirmeli öğrenme ortamını tanıyarak ajanın öğrenme öncesi davranışlarını incelemek
Süre	20 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili

Hazırlık

- Jupyter Notebook uygulamasını açınız.
- Yeni bir Notebook oluşturunuz ve adını **"RL_Ortam"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.

- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Uygulamada kullanılacak Python kütüphanelerini ekleyiniz.

```
import numpy as np
import random
```

Numpy, sayısal işlemler ve veri yapıları için kullanılır. randomsa ajanın rastgele hareket seçmesini sağlar. Uygulamada ajan henüz öğrenme yapmadığı için hareketlerini rastgele seçecektir.

2. Adım: Ortamın Tanımlanması

4x4 boyutunda bir ortam oluşturunuz.

```
boyut = 4
```

Her hücre ajanın bulunabileceği bir durumu temsil eder.

3. Adım: Başlangıç ve Hedef Noktası

Ajanın başlangıç konumunu ve ulaşması gereken hedef noktayı belirleyiniz.

```
baslangic = (0, 0)
hedef = (3, 3)
```

(0, 0) ajanın başlangıç durumudur. (3, 3) ise ajanın ulaşması beklenen hedef durumudur. Bu yapı, ajanın "başlayacağı yer" ve "ulaşmaya çalışacağı yer" bilgisini verir.

4. Adım: Ajanın Hareketleri

```
hareketler = [(0,1), (0,-1), (1,0), (-1,0)]
```

Bu kod satırı, ajanın yapabileceği hareketleri tanımlar. Ajan sağa, sola, aşağıya ve yukarıya hareket edebilir. Her hareket, koordinatlarda bir değişim oluşturur. Örneğin (0,1) sağa hareketi, (1,0) aşağı hareketi temsil edebilir.

5. Adım: Rastgele Hareket

```
x, y = baslangic
for i in range(10):
  dx, dy = random.choice(hareketler)
  x += dx
  y += dy
  print("Ajan konumu:", (x, y))
```

Gelinen süreçte ajan başlangıç noktasından hareket etmeye başlar. random.choice(hareketler) komutu, ajanın her adımda rastgele bir yön seçmesini sağlar. Döngü 10 kez çalışır ve her hareketten sonra ajanın yeni konumu ekrana yazdırılır.

Ajan henüz ödül, ceza veya öğrenme stratejisi kullanmamaktadır. Bundan dolayı ajanın hedefe ulaşması tamamen rastlantısaldır. Ajan bazen hedefe yaklaşıp bazen de hedeften uzaklaşabilir. Ayrıca sınır kontrolü yapılmadığı için ajan teorik olarak grid dışına da çıkabilir. Bu durum, öğrenme öncesi rastgele davranışların sınırlı olma durumunu göstermesi açısından önemlidir.

Ajanın hareketlerinin başlangıçta rastgele olduğu görülür. Bu durum, ajanın henüz doğru hareketleri bilmediğini gösterir.

Süreçte ajanın öğrenmesini sağlamak için **Q-learning algoritması** kullanılacaktır. Ajan, yaptığı her hareketin sonucunda aldığı geri bildirimleri değerlendirerek hangi durumlarda hangi hareketlerin daha uygun olduğunu öğrenmeye başlar.

3.7. ÖRNEK UYGULAMA

Adı	Q-Learning ile Ajanın Hedefe Ulaşmayı Öğrenmesi
Amaç	Q-learning algoritmasının temel mantığını uygulamalı olarak öğrenmek
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, scikit-learn kütüphanesi

Hazırlık

- Jupyter Notebook uygulamasını açınız.
- Yeni bir Notebook oluşturunuz ve adını **"RL_Ortam"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Q-Tablosunun Oluşturulması

Kod ile 4x4 grid ortamı için bir **Q-tablosu** oluşturunuz.

```
Q = np.zeros((4,4,4))
```

Tablo başlangıçta sıfırlardan oluşur çünkü ajan henüz hangi hareketin daha iyi olduğunu bilmemektedir.

Buradaki yapı şu anlama gelir:

- İlk **4,** satır konumlarını temsil eder.
- İkinci **4,** sütun konumlarını temsil eder.
- Üçüncü **4,** ajanın yapabileceği hareketleri temsil eder: sağ, sol, aşağı, yukarı.

Q-tablosu, her hücrede ajanın dört farklı hareket için değer tutmasını sağlar.

2. Adım: Parametrelerin Belirlenmesi

Q-learning algoritmasının nasıl öğreneceğini belirleyen temel parametreleri tanımlayınız.

```
alpha = 0.1
```

Alpha (Öğrenme Oranı): Ajanın yeni öğrendiği bilgiyi ne kadar dikkate alacağını belirler. Değer küçükse öğrenme yavaş, büyükse daha hızlı olur.

Gamma (İndirim Faktörü): Ajanın gelecekte kazanacağı ödülleri ne kadar önemseyeceğini gösterir. 0.9 değeri, ajanın yalnızca anlık ödülü değil, hedefe ulaşmayı sağlayacak gelecekteki ödülleri de dikkate aldığını gösterir.

Epsilon (Keşif Oranı): Ajanın bazen rastgele hareket ederek yeni yollar denemesini sağlarken; ajan farklı hareketleri de keşfeder.

3. Adım: Öğrenme süreci

```
for episode in range(500):
    x, y = 0, 0
    while (x, y) != hedef:
        if random.random() < epsilon:
            action = random.randint(0,3)
        else:
            action = np.argmax(Q[x,y])
        dx, dy = hareketler[action]
        nx, ny = x+dx, y+dy
        if 0 <= nx < 4 and 0 <= ny < 4:
            if (nx, ny) == hedef:
                odul = 10
            else:
                odul = -1
            Q[x,y,action] = Q[x,y,action] + alpha * (
        odul + gamma * np.max(Q[nx,ny]) - Q[x,y,action])
            x, y = nx, ny
```

Bu adımda ajan, **Q-learning algoritmasını kullanarak deneme yanılma yoluyla öğrenmeye başlar.** Süreç çok sayıda tekrarla (episode) yürütülür ve her tekrar ajanın deneyim kazanmasını sağlar.

Bu aşamada ajanın rastgele davranışlardan öğrenmeye geçiş süreci incelenecektir. Tekrar eden denemeler sonucunda ajan, hedefe ulaştıran hareketleri öğrenmeye başlar. Rastgele seçimlerin yerini daha bilinçli ve doğru kararlar alır.

Süreçte ajan daha kısa yolları tercih eder, gereksiz hareketlerden kaçınır, hedefe ulaşma süresini azalttır. Bu durum, pekiştirmeli öğrenmede ajanın zamanla performansını artırdığını gösterir.

3.8. ÖRNEK UYGULAMA

Adı	Ajan Performansının Zamanla Değişimini İnceleme
Amaç	Ajanın öğrenme sürecindeki gelişimini analiz etmek
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, scikit-learn ve matplotlib kütüphaneleri

Hazırlık

- Jupyter Notebook uygulamasını açınız.
- Yeni bir Notebook oluşturunuz ve adını **"RL_Ortam"** olarak değiştiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Adım Sayısını Takip Etme

Ajanın her denemede hedefe kaç adımda ulaştığını kaydetmek için boş bir liste oluşturunuz. Böylece ajanın performansını sayısal verilerle takip ediniz.

```
adim_sayilari = [ ]
```

2. Adım: Eğitim Sürecinin İzlenmesi

```
for episode in range(100):
  x, y = 0, 0
  adim = 0

  while (x, y) != hedef:
    action = np.argmax(Q[x,y])
    dx, dy = hareketler[action]
    x += dx
    y += dy
    adim += 1
  adim_sayilari.append(adim)
```

Bu adımda ajan 100 kez hedefe ulaşmaya çalışır. Her denemede başlangıç noktasından başlar ve Q-tablosunda öğrendiği en iyi hareketi seçerek ilerler. Hedefe ulaşana kadar yaptığı her hareket bir adım olarak sayılır.

Her denemenin sonunda toplam adım sayısı listeye eklenir. Ajan zamanla daha az adımda hedefe ulaşıyorsa, öğrenme sürecinin olumlu ilerlediğini gösterir.

3. Adım: Sonuçların İncelenmesi

Ajanın her denemede hedefe kaç adımda ulaştığını ekrana yazınız. Listedeki değerleri inceleyerek ajanın performansının zamanla nasıl değiştiğini gözlemleyiniz.

```
print(adim_sayilari)
```

Adım sayılarının azalması, ajanın hedefe daha kısa yoldan ulaşmayı öğrendiğini gösterir. Ancak adım sayıları her zaman düzenli biçimde azalmayabilir. Bazı denemelerde aynı kalabilir veya beklenmedik sonuçlar görülebilir. Bu durum, öğrenme sürecinin değerlendirilmesi gerektiğini gösterir.

Sonuç

1. **Ajan başlangıçta neden rastgele hareket eder? Açıklayınız.**

   .....
   .....

2. **Q-learning algoritması ajanın öğrenmesini nasıl sağlar? Açıklayınız.**

   .....
   .....

3. **Ajanın performansı zamanla nasıl değişmiştir? Açıklayınız.**

   .....
   .....

4. **Ödül ve ceza mekanizması öğrenmeyi nasıl etkiler? Açıklayınız.**

   .....
   .....

BİLGİ KUTUSU

Pekiştirmeli öğrenmede ajanın davranışını yalnızca hedefe ulaşması değil bu hedefe ulaşırken aldığı geri bildirimler de belirler. Ajana sadece hedefe ulaştığında yüksek ödül verilirse ajan hedefe gitmeyi öğrenebilir ancak bu süreçte gereksiz, uzun veya riskli yolları da deneyebilir. Bu nedenle ödül ve ceza değerlerinin dikkatli belirlenmesi gerekir.

Pekiştirmeli öğrenmede keşfetme ve öğrendiğini kullanma arasında denge kurulması gerekir. Grid World ortamında her adım için küçük bir ceza, hedefe ulaşma içinse yüksek bir ödül verilmesi, ajanın daha kısa ve verimli yolları tercih etmesini sağlar. Ceza değeri çok yüksek olursa ajan yeni yolları denemekten kaçınabilir. Ceza değeri çok düşük olursa bu kez gereksiz adımlar atmaya devam edebilir. Gerçek yaşamda da benzer durum vardır. Bir teslimat robotu yalnızca en kısa yolu değil güvenli, engelsiz ve enerji tüketimi düşük yolu da seçmelidir. Bu nedenle pekiştirmeli öğrenmede başarı, sadece hedefe ulaşmakla değil hedefe ulaşırken seçilen yolun uygunluğuyla da değerlendirilir. Ajanın zamanla daha az adımda, daha güvenli ve daha tutarlı kararlar alması öğrenme sürecinin olumlu ilerlediğini gösterir.

4. UYGULAMA

Adı	Grid World Ortamında Q-Learning ile Pekiştirmeli Öğrenme
Amaç	Grid World ortamında Q-Learning algoritmasını kullanarak pekiştirmeli öğrenme sürecini incelemek
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek Grid World ortamında Q-Learning uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Yaşamda birçok problem, doğrudan çözüm bilgisinin verilmediği süreçler içerir. Bu tür durumlarda en uygun davranış, yapılan denemelerin sonuçları değerlendirilerek belirlenir. Sistemler, her denemenin sonucundan elde edilen geri bildirimi kullanarak zamanla daha doğru kararlar almaya başlar.

Uygulamada, Grid World ortamı üzerinde bir ajanın karar verme süreci incelenerek öğrenmenin nasıl geliştiği somut hâle getirilecektir.

Pekiştirmeli öğrenmede ajanın davranışını belirleyen önemli unsurlardan biri ödül ceza sisteminin nasıl tasarlandığıdır. Ajan yalnızca hedefe ulaşmaya odaklanırsa uzun veya riskli yolları da deneyebilir. Her adım için küçük bir ceza, hedefe ulaşınca yüksek ödül verilmesiyse ajanın daha kısa ve verimli yolları tercih etmesini sağlar. Bu uygulamada ajanın yalnızca hedefe ulaşıp ulaşmadığı değil hedefe kaç adımda ulaştığı ve zamanla daha tutarlı karar verip vermediği de incelenmelidir.

Uygulama, pekiştirmeli öğrenmede karar verme sürecinin nasıl geliştiğini ve deneyime dayalı öğrenmenin sistem davranışını nasıl etkilediğini analiz etmeye imkân sağlar.

İşlem Basamakları

1. Grid World ortamının yapısını inceleyiniz. Ajanın başlangıç noktasını, hedef noktasını ve hareket yönlerini belirleyiniz.
2. Ortamı oluşturmak, rastgele seçim yapmak ve Q-tablosunu kullanmak için gerekli kütüphaneleri ekleyiniz.
3. 4×4 boyutunda bir Grid World ortamı oluşturunuz ve başlangıçla ile hedef noktalarını belirleyiniz.
4. Ajanın gerçekleştirebileceği hareketleri tanımlayınız. Ajanın yukarı, aşağı, sağ ve sol yönlerde hareket edebileceğini belirtiniz.
5. Her durum ve eylem için başlangıç değerlerini içeren Q-tablosunu oluşturunuz.
6. Öğrenme oranı, indirim faktörü ve keşif oranı gibi Q-Learning parametrelerini belirleyiniz.
7. Ajanın hedefe ulaştığında ödül, diğer hareketlerinde ise ceza alacağı ödül-ceza sistemini tanımlayınız.

8. Q-Learning algoritmasını uygulayınız. Ajanın aldığı geri bildirimlere göre Q-tablosunu güncellemesini sağlayınız.
9. Ajanın öğrenme sürecini gözlemleyiniz. Başlangıçtaki davranışlarıyla sonraki davranışlarını karşılaştırınız.
10. Ajanın ilk denemelerdeki ve öğrenme süreci sonundaki hedefe ulaşma adım sayılarını karşılaştırınız.
11. Ajanın öğrenme sürecini, ödül-ceza sisteminin etkisini ve performansındaki değişimi yorumlayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

3.2.5. Probleme Göre Uygun Öğrenme Türünü Seçme Uygulamaları

Her yapay zekâ problemi aynı tür veriyle çalışmaz ve aynı öğrenme yaklaşımıyla çözülemez. Yapay zekâ uygulamalarında başarılı bir model geliştirebilmek için yalnızca algoritma bilmek yeterli değildir. Öncelikle problemin yapısını doğru anlamak, eldeki verileri incelemek ve modelden beklenen sonucu açık biçimde belirlemek gerekir.

Bir öğrenme türü seçilirken temel karar noktası, **verinin yapısıyla problemin amacı arasındaki uyum**dur. Modelden beklenen işleme göre öğrenme türü seçilir. Sınıf tahmin etme sayısal değer hesaplama, verileri kendi içinde gruplandırma, bir ortamda deneyim kazanarak en uygun davranışı öğrenme gibi farklı işlemler; farklı öğrenme türünün kullanımını gerektirir.

Yapay zekâ problemine başlanmadan önce şu sorulara verilecek yanıtlar dikkate alınmalıdır:

- Veri setinde doğru sonuç veya etiket bilgisi var mı?
- Modelin üretmesi beklenen çıktı kategori mi, sayısal değer mi, grup mu yoksa davranış stratejisi mi?
- Amaç tahmin yapmak mı, verideki örüntüleri keşfetmek mi yoksa en uygun hareketi öğrenmek mi?
- Öğrenme süreci hazır veri seti üzerinden mi,yoksa ajan–ortam etkileşimi üzerinden mi gerçekleşiyor?
- Veri miktarı yeterli mi, etiketli veri sınırlı mı?

Veri setinde giriş verileriyle birlikte doğru sonuç bilgileri bulunuyorsa **denetimli öğrenme** tercih edilir. Bu durumda model, verilen örneklerden giriş ve çıkış arasındaki ilişkiyi öğrenir. İris veri setinde çiçek ölçümleri kullanılarak çiçek türünün tahmin edilmesi veya California Housing veri setinde bölge özelliklerinden ev değerinin tahmin edilmesi bu duruma örnektir. Burada önemli olan, modelin öğrenebileceği açık bir hedef bilginin bulunmasıdır.

Veri setinde hazır sınıf veya sonuç bilgisi bulunmuyorsa **denetimsiz öğrenme** kullanılır. Bu tür problemlerde modelden beklenen, tahmin yapması değil veriler arasındaki doğal benzerlik ve farkları ortaya çıkarmasıdır. Örneğin müşteri veri setinde müşterilerin hangi gruba ait olduğu önceden bilinmez. Model, gelir ve harcama puanı gibi özellikleri inceleyerek benzer müşterileri aynı kümelerde toplar. Bu yaklaşım özellikle segmentasyon, örüntü keşfi ve veri yapısını anlama çalışmalarında kullanılır.

Bazı durumlarda veri setinin yalnızca küçük bir bölümü etiketli, büyük bir bölümü ise etiketsiz olabilir. Böyle durumlarda **yarı denetimli öğrenme** daha uygun bir seçenektir. Model, başlangıçta etiketli verilerden öğrenir, ardından etiketsiz veriler için tahminler üretir. Güvenilir tahminler sözde etiket olarak kullanılarak öğren-

me süreci genişletilir. Film yorumları veri setinde bazı yorumların olumlu veya olumsuz olduğu bilinirken bazı yorumların etiketsiz olması bu yaklaşıma örnek gösterilebilir.

Problemde bir ajan, belirli bir ortamda hareket ediyor ve yaptığı davranışlara göre ödül veya ceza alıyorsa **pekiştirmeli öğrenme** tercih edilir. Bu yaklaşımda amaç, hazır cevaplardan öğrenmek değil deneyimlerden yararlanarak en uygun davranış stratejisini geliştirmektir. Grid World ortamında ajanın başlangıç noktasından hedefe ulaşmaya çalışması bu durumu gösterir. Ajan, zamanla hangi hareketlerin daha fazla ödül getirdiğini öğrenerek daha verimli kararlar almaya başlar.

Yapay zekâ uygulamalarında doğru öğrenme türünün seçilmesi büyük önem taşır. Uygun olmayan bir öğrenme türü seçildiğinde model, verinin yapısına uygun çalışamayabilir veya beklenen çıktıları üretemeyebilir. Örneğin etiket bilgisi bulunmayan bir müşteri veri setinde doğrudan denetimli öğrenme kullanılamaz. Benzer şekilde ödül-ceza ilişkisine dayalı bir oyun veya robot hareketi probleminde yalnızca sınıflandırma yaklaşımı yeterli olmayacaktır.

Bundan dolayı yapay zekâ uygulamalarında ilk adım, veri ve problem analizidir. Doğru öğrenme türü; verilerin etiket durumuna, beklenen çıktının türüne, problemin amacına ve öğrenme sürecinin nasıl gerçekleşeceğine göre belirlenir. Doğru seçim yapıldığında model daha anlamlı sonuçlar üretir ve geliştirilen yapay zekâ sistemi gerçek yaşam problemlerine daha uygun çözümler sunabilir.

Öğrenme türü seçilirken dikkate alınabilecek temel ölçütler Tablo 3.3'te verilmiştir.

Tablo 3.3: Öğrenme Türü Seçiminde Temel Ölçütler

Problem Durumu	Uygun Öğrenme Türü	Örnek
Veriler etiketli Model bir sınıf tahmin edecek.	Denetimli öğrenme	İris çiçek türü sınıflandırma
Veriler etiketli Model sayısal değer tahmin edecek.	Denetimli öğrenme/ Regresyon	Ev fiyatı tahmini
Veriler etiketsiz Model gruplar oluşturacak.	Denetimsiz öğrenme	Müşteri segmentasyonu
Verilerin bir kısmı etiketli, büyük kısmı etiketsiz	Yarı denetimli öğrenme	Film yorumlarında duygu analizi
Ajan, ortamda hareket ederek ödül ceza alacak.	Pekiştirmeli öğrenme	Grid World'de hedefe ulaşma

NOTLAR

9. SIRA SİZDE

Aşağıdaki yapılandırılmış gridde numaralandırılmış kutucuklarda **yapay zekâ öğrenme türleri ve problem özellikleriyle ilgili kavramlar** verilmiştir. Kutucuk numaralarını kullanarak 1, 2, 3 ve 4. soruları cevaplayınız (Aynı kutucuk numarasını birden fazla cevap için kullanabilirsiniz.).

1 Etiketli veri	2 Etiketsiz veri	3 Ajan
4 Sınıflandırma	5 Kümeleme	6 Ödül ceza
7 Regresyon	8 Benzerlik	9 Ortam
10 Giriş verisi (X)	11 Hedef değişken (y)	12 Deneme-yanılma

1. Modelin doğru sonuç bilgilerini kullanarak tahmin yapmasını sağlayan veri türünü hangi kavram (veya kavramlar) ifade eder?

.....
.....
.....
.....
.....
.....
.....
.....
.....

2. Modelin veriler arasındaki benzerlikleri kullanarak gruplama yapmasını sağlayan öğrenme yaklaşımına ait kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....
.....
.....
.....

3. Ajanın bir ortam içinde hareket ederek ödül veya ceza almasıyla öğrenme gerçekleştirdiği süreci ifade eden kavram (veya kavramlar) hangisidir?

.....
.....
.....
.....
.....
.....
.....
.....
.....

3.9. ÖRNEK UYGULAMA

Adı	Probleme Göre Uygun Öğrenme Türünü Seçme
Amaç	Yapay zekâ uygulamalarında kullanılan öğrenme türlerinin farklı problem durumlarındaki kullanımını açıklamak
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	çalışma kâğıdı, kalem, etkileşimli tahta veya projeksiyon

Hazırlık

- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- Problem senaryolarını incelemek için gerekli materyalleri hazırlayınız.
- Gerekirse not almak için defterinizi veya çalışma kâğıdınızı hazır bulundurunuz.
- Öğrenme türlerine ilişkin önceki bilgilerinizi gözden geçiriniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Problem senaryolarını dikkatlice okuyunuz.
- Cevaplarınızı gerekçeleriyle açıklayınız.
- Arkadaşlarınızın görüşlerine saygılı davranınız.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Problem Senaryolarının İncelenmesi

Aşağıdaki problem senaryolarını dikkatlice inceleyiniz.

1. Senaryo: Akıllı E-Posta Filtreleme Sistemi

Bir e-posta uygulaması, kullanıcılara gelen mailleri otomatik olarak "spam" (istenmeyen) ve normal olarak ayırmak istemektedir.

Uygulamanın elinde geçmişte kullanıcılar tarafından işaretlenmiş çok sayıda e-posta bulunmaktadır. Her e-posta; metin içeriği, gönderen bilgisi, başlık gibi özellikler içermektedir.

Amaç, yeni gelen e-postaları doğru şekilde sınıflandırmaktır.

2. Senaryo: E-Ticaret Müşteri Analizi

Bir e-ticaret şirketi, müşterilerini alışveriş davranışlarına göre analiz etmek istemektedir.

Şirketin elinde müşterilere ait şu bilgiler bulunmaktadır:

- Yıllık harcama miktarı
- Siteye giriş sıklığı
- Sepete eklenen ürün sayısı
- Alışveriş sıklığı

Amaç, benzer davranışlara sahip müşterilerin belirleyip bu müşterilere özel kampanyalar oluşturmaktır.

3. Senaryo: Sosyal Medya Yorumlarının İncelenmesi

Bir sosyal medya platformu, kullanıcı yorumlarını analiz ederek kullanıcı deneyimini geliştirmek istemektedir. Platformda çok sayıda kullanıcı yorumu bulunmaktadır. Bu yorumların bir kısmı daha önce incelenmiş ve değerlendirilmiş, büyük bir kısmıysa henüz değerlendirilmemiştir.

Amaç, tüm yorumlar hakkında genel bir değerlendirme yapabilen bir sistem geliştirmektir.

4. Senaryo: Otonom (Sürücüsüz) Araç Sistemi

Bir sürücüsüz araç, şehir içinde trafikte ilerlemektedir. Araç; hız, konum, trafik ışıkları ve çevredeki nesneler gibi bilgileri kullanarak hareket etmektedir. Yaptığı hareketlerin sonucuna göre geri bildirim almaktadır.

Amaç, aracın zamanla daha güvenli ve doğru sürüş davranışları geliştirmesidir.

5. Senaryo: Ev Fiyatı Tahmin Sistemi

Bir emlak platformu, yeni eklenen evlerin tahminî satış fiyatını belirlemek istemektedir.

Platformun elinde daha önce satılmış evlere ait şu bilgiler bulunmaktadır:

- Evin büyüklüğü (m^2)
- Oda sayısı
- Bulunduğu konum
- Bina yaşı

Amaç, yeni eklenen bir evin özelliklerine bakarak tahminî fiyatını belirlemektir.

2. Adım: Problemin Veri Yapısının Analiz Edilmesi

Her senaryo için aşağıdaki sorulara cevap veriniz.

- Veri setinde doğru cevap (etiket) bilgisi var mı?
- Veri seti kısmen mi etiketli?
- Öğrenme süreci bir sistemin çevreyle etkileşimine mi dayanıyor?

3. Adım: Problemin Amacının Belirlenmesi

Her senaryo için modelin amacını belirleyiniz.

- Bir sınıfı mı tahmin ediyor?
- Sayısal bir değeri mi tahmin ediyor?
- Verileri gruplara mı ayırıyor?
- En doğru davranışı mı öğrenmeye çalışıyor?

4. Adım: Uygun Öğrenme Türünün Seçilmesi

Her senaryo için en uygun öğrenme türünü seçiniz.

- Denetimli öğrenme
- Denetimsiz öğrenme
- Yarı denetimli öğrenme
- Pekiştirmeli öğrenme

5. Adım: Sonuçların Tablo Hâline Getirilmesi

Aşağıdaki tabloyu doldurunuz.

Senaryo	Veri Yapısı	Problem Amacı	Öğrenme Türü
I.			
II.			
III.			
IV.			
V.			

6. Adım: Seçimlerin Gerekçelendirilmesi

Seçtiğiniz öğrenme türlerini kısa açıklamalarla gerekçelendiriniz.

Örnek Yapı

"Bu problemde ... öğrenme türü kullanılmalıdır çünkü ..."

Sonuç

1. **Bir yapay zekâ probleminde uygun öğrenme türü nasıl belirlenir? Açıklayınız.**

.....
.....
.....
.....

2. **Denetimli öğrenme hangi tür problemler için uygundur? Açıklayınız.**

.....
.....
.....
.....

3. **Denetimsiz öğrenme hangi durumlarda tercih edilir? Açıklayınız.**

.....
.....
.....
.....

4. **Pekiştirmeli öğrenme hangi problemler için uygundur? Açıklayınız.**

.....
.....
.....
.....

5. **Yarı denetimli öğrenmenin avantajı nedir? Açıklayınız.**

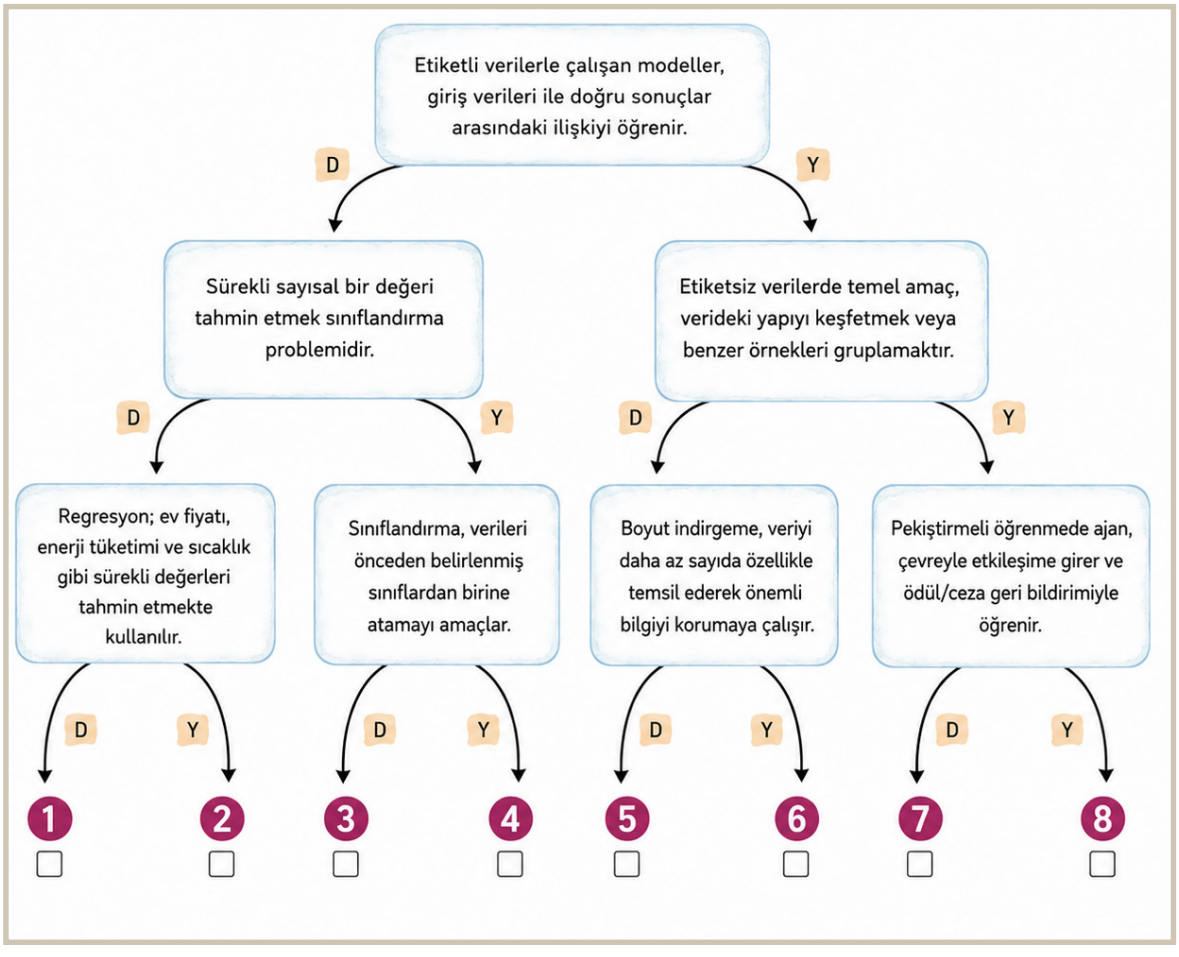
.....
.....
.....
.....

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki tabloyu doldurarak yapay zekâ öğrenme türlerini karşılaştırınız.

Öğrenme Türü	Kullanım Amacı	Veri Türü	Temel Mantık	Örnek Kullanım Alanı
Denetimli Öğrenme				
Denetimsiz Öğrenme				
Yarı Denetimli Öğrenme				
Pekiştirmeli Öğrenme				

B) Aşağıdaki ifadeleri Doğru (D) veya Yanlış (Y) olarak değerlendiriniz ve uygun yolu takip ediniz.



C) Aşağıdaki "Akıllı Şehir 2035" adlı metni dikkatle okuyup soruları cevaplayınız.

Akıllı Şehir 2035

Türkiye'nin büyük şehirlerinden biri, artan nüfus, trafik yoğunluğu, enerji tüketimi ve güvenlik ihtiyaçlarını daha etkili yönetebilmek amacıyla "Akıllı Şehir 2035" projesini başlatmıştır. Bu proje kapsamında şehir genelinde sensörler, kameralar, mobil uygulamalar ve çeşitli dijital sistemler aracılığıyla sürekli veri toplanmaktadır.

Toplanan veriler şunları içermektedir:

- Trafik kameralarından gelen araç görüntüleri
- Toplu taşıma kartlarından elde edilen yolcu verileri
- Hava durumu ve hava kirliliği sensör verileri
- Vatandaşların mobil uygulama üzerinden yaptığı şikâyet ve talepler
- Güvenlik kameralarından elde edilen görüntüler
- Elektrik ve su tüketim verileri

Belediyede veri bilimci olarak çalışan Zeynep, bu verileri kullanarak farklı yapay zekâ sistemleri geliştirmekle görevlendirilmiştir. Ancak her problem için hangi öğrenme türünün (denetimli, denetimsiz, yarı denetimli, pekiştirmeli öğrenme) kullanılacağına karar vermesi gerekmektedir.

Zeynep'in karşılaştığı problemler şunlardır:

1. Problem: Trafik Yoğunluğu Tahmini

Şehrin farklı noktalarına yerleştirilen trafik kameralarından elde edilen görüntüler, daha önce uzmanlar tarafından incelenmiş ve her bir görüntü yoğun, orta ve boş şeklinde etiketlenmiştir. Geliştirilecek sistemin, yeni elde edilen görüntüleri analiz ederek trafik yoğunluğunu doğru şekilde tahmin etmesi beklenmektedir.

2. Problem: Yeni Yolcu Gruplarını Keşfetme

Toplu taşıma sistemlerinden elde edilen yolcu hareket verileri herhangi bir etiket içermemektedir. Bu veriler; yolculuk saatleri, kullanılan hatlar ve durak bilgilerini kapsamaktadır. Belediye, bu verileri analiz ederek benzer ulaşım alışkanlıklarına sahip yolcu gruplarını belirlemek ve ulaşım planlamasını buna göre düzenlemek istemektedir.

3. Problem: Eksik Etiketli Güvenlik Verileri

Şehirdeki güvenlik kameralarından elde edilen görüntülerin yalnızca küçük bir kısmı şüpheli davranış olarak etiketlenmiştir. Geri kalan büyük veri etiketsizdir. Sistem, bu verileri kullanarak güvenlik risklerini daha doğru tespit etmek istemektedir.

4. Problem: Akıllı Trafik Işıkları Sistemi

Şehirdeki bazı kavşaklarda trafik ışıklarının süreleri sabit olarak belirlenmiştir. Ancak trafik yoğunluğunun gün içinde değişkenlik göstermesi nedeniyle bu sistem yetersiz kalmaktadır. Geliştirilecek sistemin, farklı trafik yoğunluklarına göre ışık sürelerini ayarlaması ve zaman içinde en uygun süreleri belirleyerek trafik akışını düzenlemesi hedeflenmektedir.

5. Problem: Enerji Tüketimi Tahmini

Şehirdeki binalardan elde edilen geçmiş elektrik tüketim verileri incelenmektedir. Bu veriler; zaman, hava koşulları ve kullanım yoğunluğu gibi değişkenleri içermektedir. Geliştirilecek sistemin, bu verilerden yararlanarak gelecekteki enerji ihtiyacını tahmin etmesi ve enerji planlamasına katkı sağlaması amaçlanmaktadır.

Sorular

1. **Yukarıda verilen her bir problem için en uygun yapay zekâ öğrenme türünü belirleyiniz. Seçimlerinizi "etiketli veri, etiketsiz veri, tahmin, kümeleme, deneme yanılma" kavramlarını kullanarak gerekçelendiriniz.**

.....
.....
.....
.....
.....
.....
.....
.....

2. **Seçtiğiniz öğrenme türü yerine farklı bir öğrenme türü kullanılması durumunda ortaya çıkabilecek avantaj ve kısıtları değerlendiriniz.**

.....
.....
.....
.....
.....
.....
.....
.....

3. **Bu tür sistemlerin geliştirilme sürecinde etik ve toplumsal sorumluluk açısından dikkat edilmesi gereken hususları açıklayınız.**

.....
.....
.....
.....
.....
.....
.....

Ç) Aşağıdaki soruların doğru cevabını işaretleyiniz.

1. Aşağıdakilerden hangisi denetimli öğrenmenin temel özelliğidir?

A) Etiketli veri kullanarak öğrenmesi
B) Deneme yanılma öğrenmesi
C) Etiketsiz veri kullanması
D) Rastgele veri üretmesi
E) Sadece görsel verilerle çalışması

2. Aşağıdakilerden hangisi denetimsiz öğrenmeye örnektir?

A) Spam e-posta sınıflandırma
B) Konut fiyatı tahmini
C) Müşteri segmentasyonu
D) Sınav notu tahmini
E) Hastalık teşhisi

3. Aşağıdakilerden hangisi yarı denetimli öğrenmede birlikte kullanılan veri türlerini ifade etmektedir?

A) Sadece sayısal
B) Etiketli ve etiketsiz
C) Sadece etiketli
D) Sadece metin verisi
E) Görsel

4. Aşağıdakilerden hangisi pekiştirmeli öğrenmede öğrenmenin gerçekleşme biçimini açıklamaktadır?

A) Etiketli veriyle
B) Kümelemeyle
C) Sadece istatistiksel analizle
D) Deneme yanılma ve ödül cezayla
E) Veri temizlemeyle

5. Aşağıdakilerden hangisi sınıflandırma problemine örnektir?

A) E-postayı spam veya normal olarak ayırma
B) Sınav puanı hesaplama
C) Ev fiyatı tahmini
D) Sıcaklık tahmini
E) Satış miktarı hesaplama

6. Aşağıdakilerden hangisi regresyon problemlerinde modelin çıktısını ifade etmektedir?

A) Kategori
B) Sayısal değer
C) Etiket
D) Metin
E) Görsel

7. Aşağıdakilerden hangisi denetimsiz öğrenmenin temel amacıdır?

A) Doğru cevapları öğretmek
B) Veri etiketlemek
C) Modeli ödüllendirmek
D) Veriler arasındaki gizli yapıyı keşfetmek
E) Kod yazmak

D) Aşağıdaki soruları cevaplayınız.

1. Denetimli öğrenmeyle denetimsiz öğrenme arasındaki temel farkları açıklayınız.

.....
.....
.....
.....
.....
.....
.....
.....
.....

2. **Yanlış sınıflandırma yapan bir yapay zekâ sisteminin gerçek hayatta oluşturabileceği sorunlara örnek veriniz.**

.....
.....
.....
.....
.....
.....
.....
.....
.....

3. **Bir problemin çözümünde uygun öğrenme türünü seçmenin neden önemli olduğunu açıklayınız.**

.....
.....
.....
.....
.....
.....
.....
.....
.....



4. ÖĞRENME BİRİMİ

YAPAY ZEKÂ YÖNTEMLERİ

KONULAR

4.1. REGRESYON YÖNTEMLERİ
4.2. SINIFLANDIRMA
4.3. KÜMELEME
4.4. BİRLİKTELİK KURALLARI
4.5. VAKA ÇALIŞMALARI

ANAHTAR KAVRAMLAR

- Apriori algoritması
- birliktelik kuralları
- doğruluk oranı
- K-Means
- kümeleme
- model
- regresyon
- sınıflandırma
- tahmin
- veri kümesi
- yapay zekâ yöntemleri



BEKLENTİLER

Bu öğrenme biriminde sizden

- sayısal veriler üzerinde regresyon yöntemlerini kullanmanız
- veri kümelerini sınıflandırarak anlamlı sonuçlar elde etmeniz
- benzer verileri kümeleme yöntemleriyle gruplandırmanız
- veriler arasındaki ilişkileri birliktelik kurallarıyla incelemeniz
- gerçek yaşam problemleri üzerinden yapay zekâ yöntemlerini kullanmanız

beklenmektedir.

Öğrenme Birimi Karekodu

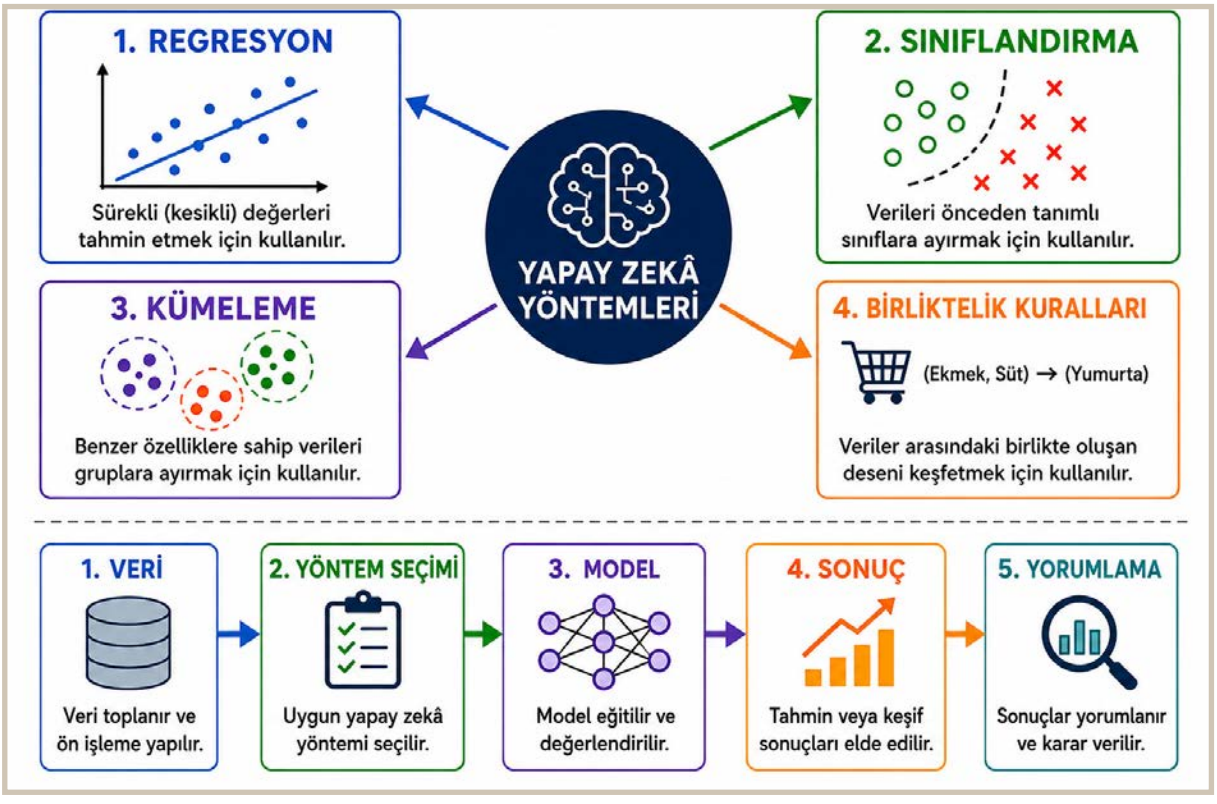
BAŞLARKEN

Yapay zekâ sistemleri problem türlerine göre farklı yöntemler kullanır. Bazı problemlerde sayısal bir değer tahmin edilirken bazı problemlerde veriler belirli sınıflara ayrılır, benzer özelliklere sahip veriler gruplandırılır veya veriler arasındaki ilişkiler incelenir. Öğrenme biriminde regresyon, sınıflandırma, kümeleme ve birliktelik kuralları yöntemleri; problem türüne uygun yöntem seçme, uygulama yapma ve sonuçları yorumlama açısından ele alınacaktır. Amaç, yalnızca bu yöntemlerin ne olduğunu öğrenmek değil hangi problemde neden kullanılabileceğini değerlendirmektir.

ÖNCE DÜŞÜN

Yapay zekâ sistemi geliştirilirken ilk olarak problemin türü belirlenmelidir. Sayısal değer tahmini, veri sınıflandırılması, benzer özelliklerin gruplandırılması; problem türlerine örnektir.

Şema 4.1'de yapay zekâ yöntemlerinin temel türleri ve uygulanma süreci gösterilmiştir.



Şema 4.1: Yapay zekâ yöntemleri genel akış şeması

Şemadaki bilgiler doğrultusunda aşağıdaki soruları cevaplayınız.

1. Bir problemde sayısal tahmin yapılacaksa hangi yöntem tercih edilebilir?
2. Veriler belirli kategorilere ayrılacaksa hangi yöntem kullanılabilir?
3. Benzer özelliklere sahip verilerin gruplandırılması gereken durumlarda hangi yöntem uygun olabilir?
4. Veriler arasındaki birlikte görülme ilişkileri incelenecekse hangi yöntemden yararlanılabilir?

Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

4.1. REGRESYON YÖNTEMLERİ

Düşün, Tartış, Paylaş

Bir okulda geçmiş yıllara ait günlük elektrik tüketim miktarlarının kaydedildiğini düşününüz. Okul yönetimi, gelecek haftalarda ne kadar elektrik tüketilebileceğini önceden tahmin etmek istemektedir. Geçmiş tüketim verilerinin zaman içindeki değişimi incelendiğinde belirli eğilimler fark edilebilir.

Buna göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Geçmiş elektrik tüketim verileri gelecekteki tüketim miktarını tahmin etmek için nasıl kullanılabilir?**
2. **Geçmiş verilerde belirli artış veya azalış eğilimi görülmesi gelecekteki değerler hakkında fikir verebilir mi? Neden?**

Regresyon yöntemleri, veriler arasındaki ilişkilerden yararlanarak sayısal sonuçların tahmin edilmesini sağlar. Bu yöntemlerde verilerin yapısı incelenir, probleme uygun regresyon modeli belirlenir ve elde edilen model yeni veriler üzerinde tahmin yapmak amacıyla kullanılır. Bölümde regresyon yöntemlerinin temel mantığı, farklı regresyon türleri ve model sonuçlarının değerlendirilmesi ele alınacaktır.

4.1.1. Sayısal Tahmin İçin Regresyon

Problemin regresyon problemi olup olmadığını anlamanın en kolay yolu beklenen çıktıya bakmaktır. Çıktı; enerji miktarı, sıcaklık, süre, maliyet, ürün sayısı, puan gibi kesintisiz sayıysa regresyon yaklaşımı uygundur. "Hangi sınıf?" sorusu yerine "Ne kadar?" veya "Kaç?" sorusu sorulur.

BİLGİ KUTUSU

Sonucun sayı biçiminde olması tek başına yeterli değildir. Örneğin 0 ve 1 sayıları "Arıza yok. / Arıza var." sınıflarını temsil ediyorsa problem regresyon değil sınıflandırmadır. Sayının anlamı incelenmelidir.

4.1.2. Tahmin Modelinde Giriş ve Sonuç Değişkenleri

Regresyon modellerinde tahmin işlemi gerçekleştirilirken kullanılan veriler iki grupta incelenebilir. Tahmin yapılmasını sağlayan verilere **giriş değişkenleri,** model tarafından tahmin edilen değere **sonuç değişkeni / değişkenleri** adı verilir. Örneğin bir öğrencinin çalışma süresi, çözdüğü soru sayısı ve deneme sınavı puanları; giriş değişkenleri olarak kullanılabilir. Bu bilgiler kullanılarak öğrencinin dönem sonu başarı puanı tahmin edilmeye çalışılır. Bu durumda dönem sonu başarı puanı sonuç değişkenidir.

Giriş değişkenleri ve sonuç değişkeni arasındaki ilişkinin doğru kurulması, modelin anlamlı tahminler üretebilmesi açısından önemlidir (Tablo 4.1).

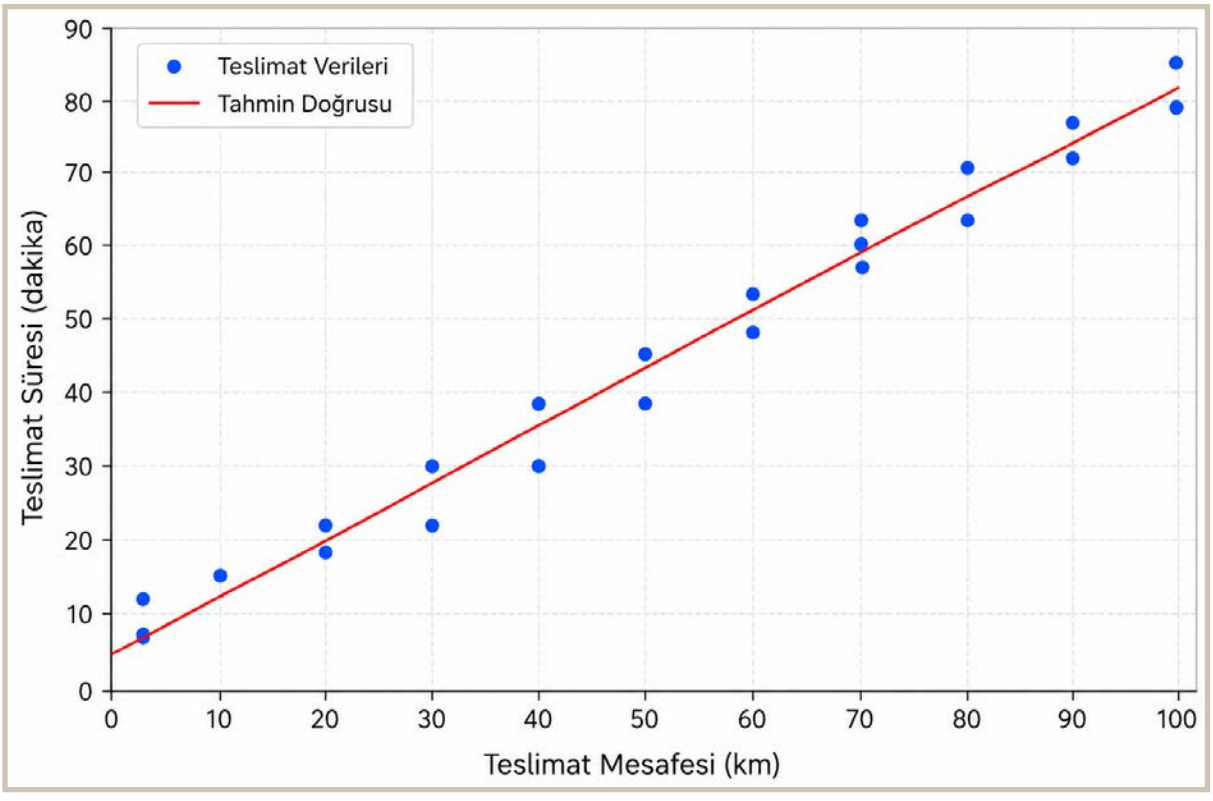
Tablo 4.1: Regresyon Problemlerinde Giriş ve Sonuç Değişkenlerine İlişkin Örnekler

Giriş Verileri	Tahmin Edilen Sonuç
Günlük sıcaklık ölçümleri	1. Sonraki günün hava sıcaklığı
Çalışma süresi	2. Sınav puanı
Geçmiş satış verileri	3. Gelecek satış miktarı
Geçmiş enerji tüketim verileri	4. Gelecek enerji tüketimi

4.1.3. Veriler Arasındaki İlişkiyi Grafikle Gösterme

Giriş değişkeni ile sonuç değişkeni arasındaki ilişki grafik üzerinde incelenebilir. Grafikte yer alan her veri noktası, gözlemlenen bir giriş değerine karşılık elde edilen sayısal sonucu gösterir. Veri noktalarının dağılımı ve genel eğilimi, değişkenler arasında nasıl bir ilişki bulunduğunun anlaşılmasına yardımcı olur.

Şema 4.2'de teslimat mesafesi ile teslimat süresi arasındaki ilişki gösterilmektedir. Grafikteki veri noktalarının genel eğilimi incelendiğinde teslimat mesafesi arttıkça teslimat süresinin de artma eğiliminde olduğu görülmektedir. Regresyon doğrusu, iki değişken arasındaki genel ilişkiyi temsil ederek daha önce gözlemlenmemiş bir teslimat mesafesi için yaklaşık teslimat süresinin tahmin edilmesini sağlar. Trafik yoğunluğu, hava koşulları, teslimat güzergâhı gibi etkenler nedeniyle gerçek süreler regresyon doğrusundan farklılık gösterebilir. Böylece öğrenciler, veri noktalarının her zaman doğru üzerinde bulunmadığını ve regresyonun yaklaşık tahmin ürettiğini de gözlemleyebilir.

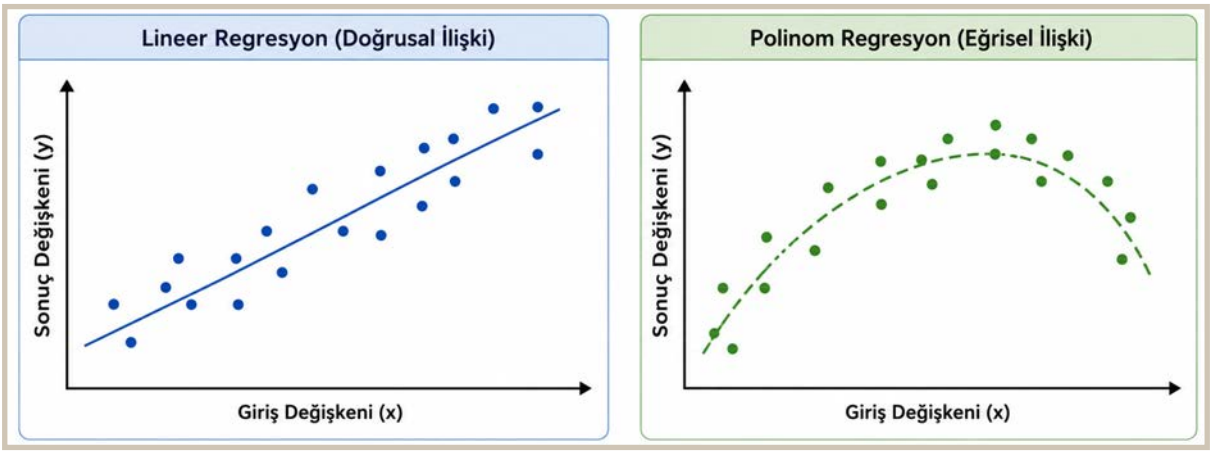


Şema 4.2: Regresyon grafiği örneği

4.1.4. Lineer ve Polinom Regresyonu Ayırt Etme

Regresyon problemlerinde önemli olan yalnızca sayısal tahmin yapmak değil veriler arasındaki ilişkinin nasıl değiştiğini de doğru yorumlamaktır. Bazı veri kümelerinde değişkenler arasındaki ilişki yaklaşık olarak sabit bir artış veya azalış gösterir. Bu tür durumlarda lineer regresyon tercih edilir. Bazı veri kümelerindeyse değişim sabit değildir; artış veya azalışın hızı zamanla farklılaşır ve veri noktaları eğrisel bir görünüm oluşturur. Bu durumlarda polinom regresyon daha uygun sonuçlar verebilir. Örneğin bir kargo şirketinde teslimat mesafesi arttıkça teslimat süresinin de genel olarak artması beklenir. Veri noktaları yaklaşık bir doğru çevresinde toplanıyorsa bu ilişki lineer regresyonla modellenebilir. Buna karşılık bir aracın hızı ile yakıt tüketimi arasındaki ilişki her zaman aynı oranda değişmeyebilir. Hız arttıkça tüketimdeki artış belirli bir noktadan sonra daha belirgin hâle gelebilir. Böyle durumlarda ilişki eğrisel bir yapı gösterdiği için polinom regresyon kullanılması daha uygun olabilir.

Şema 4.3 incelendiğinde, lineer regresyonda veri noktalarının bir doğru etrafında toplandığı; polinom regresyondaysa veri noktalarının eğri boyunca dağıldığı görülmektedir. Bu nedenle model seçimi yapılırken verilerin genel eğilimi dikkatle incelenmeli ve probleme en uygun regresyon türü belirlenmelidir.



Şema 4.3: Lineer ve polinom regresyon örnekleri

4.1.5. Doğrusal Regresyon Modeli Oluşturma

Doğrusal regresyon modeli, giriş değişkeni ile sonuç değişkeni arasındaki ilişkiyi matematiksel bir doğruyla ifade eder. Bu ilişki genel olarak şu denklemle gösterilir:

$$y = ax + b$$

Denklemde x giriş değerini, y modelin ürettiği sonucu, a doğrunun eğimini, b sabit değeri temsil eder. Model, kendisine verilen verileri inceleyerek eğim ve sabit değerlerini hesaplar. Böylece giriş değişkenindeki değişimin sonuç üzerinde oluşturduğu etki sayısal olarak yorumlanabilir.

Bir 3B yazıcının baskı süresi ile tükettiği filament miktarı arasındaki doğrusal ilişkinin modellendiği örnek şu şekildedir:

```
from sklearn.linear_model import LinearRegression
import numpy as np
# Baskı süreleri (dakika)
X = np.array([[15], [25], [35], [45], [55]])
# Tüketilen filament miktarları (gram)
y = np.array([6.5, 10.2, 14.1, 17.8, 21.9])
# Doğrusal regresyon modelini oluşturma
model = LinearRegression()
model.fit(X, y)
# Modelin öğrendiği değerleri belirleme
egim = model.coef_[0]
sabit_deger = model.intercept_
print("Eğim:", round(egim, 3))
print("Sabit değer:", round(sabit_deger, 3))
# Model denkleminden yeni değer hesaplama
```

```
baski_suresi = 65
tahmini_filament = egim * baski_suresi + sabit_deger
print(
  "65 dakikalık baskı için tahmini filament miktarı:",
  round(tahmini_filament, 2),
  "gram"
)
```

Kod çalıştırıldığında modelin oluşturduğu doğrusal ilişki yaklaşık olarak şu denklemle ifade edilir:

Filament miktarı = 0,384 × Baskı süresi + 0,66

Denklemdeki 0,384 eğim değeri, baskı süresindeki her bir dakikalık artışın filament tüketimini yaklaşık 0,384 gram artırdığını gösterir. Model denklemine göre 65 dk. süren bir baskı işleminde yaklaşık 25,62 gram filament kullanılacağı hesaplanır.

4.1.6. Model Eğitimi, Testi ve Tahmin Hatası

Doğrusal regresyon modeli oluşturulduktan sonra modelin yalnızca öğrendiği veriler üzerindeki uyumuna bakmak yeterli değildir. Modelin daha önce karşılaşmadığı veriler için de gerçeğe yakın sonuçlar üretmesi beklenir. Bu nedenle veri kümesinin bir bölümü modelin eğitilmesinde, kalan bölümüyse tahmin başarısının değerlendirilmesinde kullanılır.

Eğitim verileri, modelin giriş ve sonuç değişkenleri arasındaki ilişkiyi öğrenmesini sağlar. Test verileriyse eğitim sırasında modele gösterilmez. Modelin test verileri için ürettiği tahminler gerçek değerlerle karşılaştırılarak öğrendiği ilişkiyi yeni durumlara ne ölçüde aktarabildiği belirlenir. Modelin daha önce görmediği veriler üzerinde de başarılı sonuçlar üretmesine **genelleme** adı verilir.

Regresyon problemlerinde değerlendirme, tahmin edilen değerlerin gerçek değerlere ne kadar yakın olduğu dikkate alınarak yapılır. Gerçek değer ile modelin tahmin ettiği değer arasındaki fark tahmin hatası olarak adlandırılır.

Tahmin hatası = Gerçek değer – Tahmin edilen değer

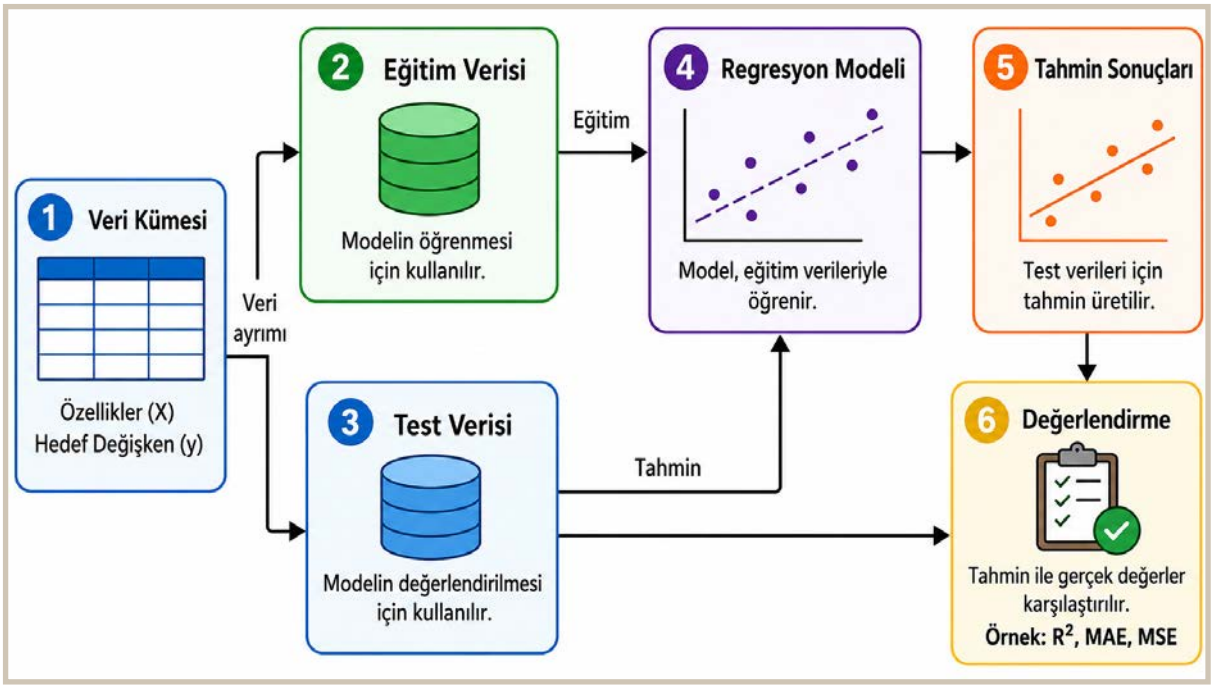
Hatanın pozitif olması modelin gerçek değerden daha düşük, negatif olmasıysa daha yüksek tahmin ürettiğini gösterir. Hatanın mutlak değerinin küçülmesi, tahminin gerçek değere yaklaştığı anlamına gelir. Birden fazla tahmin birlikte değerlendirilirken şu ölçütlerden yararlanılır:

Ortalama Mutlak Hata (MAE): Tahminlerin gerçek değerlerden ne kadar uzaklaştığını ortalama olarak gösterir. Değer küçüldükçe tahminler gerçeğe yaklaşır.

Ortalama Kare Hata (MSE): Büyük tahmin hatalarına daha fazla ağırlık vererek modeldeki önemli sapmaların fark edilmesini sağlar.

R-kare (R^2): Modelin sonuç değişkenindeki değişimi ne ölçüde açıkladığını gösterir. Değerin 1'e yaklaşması, modelin veri üzerindeki açıklama gücünün arttığını belirtir.

Şema 4.4'te veri kümesinin eğitim ve test bölümlerine ayrılması, modelin eğitim verileriyle öğrenmesi ve test verileri üzerinde üretilen sonuçların gerçek değerlerle karşılaştırılması gösterilmiştir. Eğitim verilerinde düşük hata olmasına rağmen test verilerinde hatanın belirgin biçimde yükselmesi, modelin yalnızca eğitim verilerine aşırı uyum sağladığını gösterebilir. Bu nedenle model başarısı eğitim sonuçlarının yanında test sonuçları da dikkate alınarak değerlendirilmelidir.



Şema 4.4: Regresyon modelinin eğitim, test ve değerlendirme süreci

Regresyon modelinin güvenilirliği; verilerin doğru, yeterli ve problemi temsil edecek çeşitlilikte olmasına bağlıdır. Eksik, hatalı veya belirli durumlarda yoğunlaşmış veriler, modelin yeni örnekler üzerinde yanıltıcı tahminler üretmesine neden olabilir. Bu nedenle tahmin hataları değerlendirilirken yalnızca ölçütlerin sayısal değerleri değil kullanılan verilerin niteliği ve problemin koşulları da göz önünde bulundurulmalıdır.

1. SIRA SİZDE

Aşağıdaki problem durumlarının regresyon problemi olup olmadığını gerekçeleriyle yazınız..

Problem Durumu	Regresyon Problemi mi?	Gerekçe
Otobüs hattının gelecek saat taşıyacağı yolcu sayısını tahmin etme		
E-postayı "istenmeyen" veya "normal" olarak belirleme		
Seranın günlük su ihtiyacını litre olarak tahmin etme		
Görseldeki nesneyi kedi, köpek veya kuş olarak belirleme		
Üretim hattında bir sonraki saatte üretilecek ürün miktarını tahmin etme		

4.1. ÖRNEK UYGULAMA

Adı	Sera Sulama Suyu İhtiyacının Tahmin Etme
Amaç	Sıcaklık ve nem verileriyle sulama suyu ihtiyacını tahmin eden regresyon modeli oluşturmak
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlam dili, NumPy ve scikit-learn kütüphaneleri

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Yeni bir Notebook oluşturunuz ve adını **"Sera_Sulama_Tahmini"** olarak değiştiriniz.
- Çalışma ortamınızı uygulamaya hazır hâle getiriniz.
- Gerekli kütüphanelerin kurulu olduğunu kontrol ediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Gerekli kütüphaneleri programa ekleyiniz.

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import (
  r2_score,
  mean_absolute_error,
  mean_squared_error
)
```

2. Adım: Veri Setinin Oluşturulması

Sıcaklık, nem ve sulama suyu verilerini tanımlayınız.

```
sicaklik = np.array([
  18, 20, 21, 23, 24, 25,
```

```
    27, 28, 29, 30, 31, 33
])
nem = np.array([
    80, 75, 72, 68, 65, 62,
    58, 55, 52, 48, 45, 40
])
sulama_suyu = np.array([
    22.5, 27.1, 30.6, 35.1,
    38.9, 41.3, 46.9, 49.3,
    52.9, 55.3, 58.6, 64.2
])
```

3. Adım: Giriş ve Sonuç Değişkenlerinin Belirlenmesi

Giriş ve sonuç değişkenlerini oluşturunuz.

```
X = np.column_stack((sicaklik, nem))
y = sulama_suyu
```

4. Adım: Veri Setinin Eğitim ve Test Verilerine Ayrılması

Veri setini eğitim ve test verilerine ayırınız.

```
X_egitim, X_test, y_egitim, y_test = train_test_split(
    X,
    y,
    test_size=0.25,
    random_state=42
)
```

5. Adım: Modelin Oluşturulması ve Eğitilmesi

Doğrusal regresyon modelini oluşturunuz ve eğitiniz.

```
model = LinearRegression()
model.fit(X_egitim, y_egitim)
```

6. Adım: Test Verileri İçin Tahmin Yapılması

Test verileri için tahmin oluşturunuz.

```
tahminler = model.predict(X_test)
print("Gerçek Değerler:", y_test)
print("Tahminler:", tahminler)
```

7. Adım: Model Performansının Hesaplanması

R-kare, MAE ve MSE değerlerini hesaplayınız.

```
r_kare = r2_score(y_test, tahminler)
mae = mean_absolute_error(y_test, tahminler)
mse = mean_squared_error(y_test, tahminler)
print("R-kare Değeri:", round(r_kare, 3))
print("MAE Değeri:", round(mae, 3))
print("MSE Değeri:", round(mse, 3))
```

Elde ettiğiniz değerleri tabloya yazınız.

Performans Ölçütü	Elde Edilen Değer
R-kare	
MAE	
MSE	
D	
E	

8. Adım: Yeni Veri İçin Tahmin Yapılması

Sıcaklığın 26 °C ve nem oranının %60 olduğu bir gün için tahmin yapınız.

```
yeni_veri = np.array([[26, 60]])
yeni_tahmin = model.predict(yeni_veri)
print(
  "Tahmin Edilen Sulama Suyu Miktarı:",
  round(yeni_tahmin[0], 2),
  "litre"
)
```

Elde ettiğiniz tahmin değerini yazınız.

Tahmin edilen sulama suyu miktarı: ..... litre

9. Adım: Sonuçların Karşılaştırılması

- Gerçek değerlerle tahmin edilen değerleri karşılaştırınız.
- En yüksek tahmin farkını belirleyiniz.
- Performans ölçütlerini birlikte inceleyiniz.
- Modelin sulama planlamasında kullanılabilirliğine karar veriniz.

Çalışmanızı kaydediniz. Oturumu güvenli şekilde kapatınız ve çalışma ortamınızı düzenli bırakınız.

Sonuç

1. **Bu uygulamada modele verilen giriş değişkenlerini ve modelin tahmin ettiği sonuç değişkenini yazınız.**

   .....

2. **Veri setinin eğitim ve test verilerine ayrılmasının modelin değerlendirilmesine nasıl katkı sağladığını açıklayınız.**

   .....

3. **Gerçek değerlerle tahmin edilen değerleri karşılaştırınız. Tahminlerin gerçek değerlere yakın olup olmadığını elde ettiğiniz sonuçlara göre açıklayınız.**

   .....

4. **R-kare, MAE ve MSE değerlerini birlikte inceleyiniz. Bu ölçütlere göre modelin performansını değerlendiriniz.**

   .....

1. UYGULAMA

Adı	Duvar Yüzey Alanına Göre Boya Miktarını Tahmin Etme
Amaç	Yüzey alanı verilerini kullanarak boya miktarını tahmin eden doğrusal regresyon modelini oluşturarak modelin performansını değerlendirmek
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek duvar yüzey alanına göre kullanılacak boya miktarını tahmin eden regresyon uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Bir boya atölyesinde farklı büyüklükteki duvar yüzeyleri için kullanılan boya miktarları kaydedilmektedir. Duvarın yüzey alanı arttıkça ihtiyaç duyulan boya miktarı da genel olarak artmaktadır. Geçmiş uygulamalardan elde edilen veriler kullanılarak yeni bir duvar yüzeyi için gerekli boya miktarı yaklaşık olarak tahmin edilebilir.

İşlem Basamakları

1. Veri setini oluşturunuz.

   Duvar yüzey alanlarını ve bu yüzeylerde kullanılan boya miktarlarını aşağıdaki verilere göre tanımlayınız.

   Yüzey alanı (m^2): [12, 18, 25, 32, 40, 48, 55, 63, 72, 80, 90, 100]

   Boya miktarı (litre): [1.8, 2.5, 3.4, 4.2, 5.3, 6.1, 7.0, 7.9, 9.1, 10.0, 11.2, 12.4]
2. Gerekli kütüphaneleri ekleyiniz.
3. NumPy, train_test_split, LinearRegression, r2_score, mean_absolute_error ve mean_squared_error yapılarını programa dâhil ediniz.
4. Yüzey alanı verilerini NumPy dizisi hâline getiriniz ve giriş değişkeni olarak düzenleyiniz.
5. Boya miktarı verilerini NumPy dizisi hâline getiriniz ve sonuç değişkeni olarak belirleyiniz.
6. Veri setini eğitim ve test verileri olmak üzere iki bölüme ayırınız. Test verisi oranını 0.25, random_state değerini 42 olarak belirleyiniz.
7. Doğrusal regresyon modelini oluşturunuz.
8. Modeli eğitim verilerini kullanarak eğitiniz.
9. est verileri için boya miktarı tahminlerini oluşturunuz.
10. Gerçek boya miktarlarıyla modelin tahmin ettiği boya miktarlarını karşılaştırınız.
11. Modelin R-kare değerini hesaplayınız.
12. Modelin Ortalama Mutlak Hata değerini hesaplayınız.
13. Modelin Ortalama Kare Hata değerini hesaplayınız.

14. R-kare, MAE ve MSE sonuçlarını inceleyerek modelin tahmin başarısını değerlendiriniz.
15. Yüzey alanı 68 m^2 olan bir duvar için gerekli boya miktarını tahmin ediniz.
16. Tahmin edilen boya miktarını gerçek yaşam koşulları açısından yorumlayınız. Boyanın türü, yüzeyin yapısı, uygulanacak kat sayısı gibi etkenleri dikkate alınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

4.2. SINIFLANDIRMA

Düşün, Tartış, Paylaş

Bir okul atölyesindeki el aletlerinin kullanım öncesinde kontrol edildiğini düşününüz. Aletler; sapının sağlamlığı, üzerinde pas bulunması, parçalarının gevşek olup olmaması gibi gözlemlenebilir özelliklere göre "Kullanıma Uygun" veya "Bakım Gerekli" olarak iki gruba ayrılacaktır.

Aşağıdaki örnekleri inceleyiniz.

- A Aleti: Sapı sağlamdır, üzerinde pas yoktur ve parçaları gevşek değildir.
- B Aleti: Sapında çatlak vardır, üzerinde pas bulunmaktadır ve bir parçası gevşektir.

Verilen bilgilere göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **A ve B aletlerini uygun gördüğünüz gruplara yerleştiriniz. Kararınızın nedenini açıklayınız.**
2. **Bir aletin hangi grupta yer alacağını belirlerken başka hangi özellikler incelenebilir?**
3. **Bakım gerektiren aletin yanlışlıkla "Kullanıma Uygun" grubuna yerleştirilmesi hangi sorunlara yol açabilir?**
4. **Alet hakkında yeterli bilgi bulunmadığında doğru karar verebilmek için ne yapılmalıdır?**

Sınıflandırma problemlerinde tahmin edilecek sınıflar ve bu sınıfları birbirinden ayırmada kullanılacak özellikler belirlenir. İki sonuç sınıfının bulunduğu problemler iki sınıflı sınıflandırma, ikiden fazla sonuç sınıfının bulunduğu problemlerse çok sınıflı sınıflandırma olarak adlandırılır. Kullanılacak yöntem, verilerin yapısına ve ulaşılmak istenen sonuca göre seçilir.

Karar ağacı yöntemiyle yeni verilerin ait olduğu sınıf tahmin edilebilir. Modelin başarısı, doğruluk oranı ve karmaşıklık matrisi kullanılarak değerlendirilir. Doğru ve yanlış tahminler karşılaştırılarak modelin başarılı olduğu ve karıştırdığı sınıf belirlenir.

4.2.1. Verileri Sınıflara Ayırma

Sınıflandırma işleminde veri kümesindeki her örnek, sahip olduğu özellikler ve ait olduğu sınıf bakımından incelenir. Özellikler, sınıfın belirlenmesinde kullanılan bilgileri; sınıfsa tahmin edilmek istenen kategoriyi ifade eder. Model, geçmiş örneklerdeki özelliklerle sınıflar arasındaki bağlantıları çözümleyerek yeni bir veriyi

önceden belirlenmiş sınıflardan birine yerleştirir. Örneğin bir başvurunun eğitim durumu, deneyim süresi ve belge bilgileri incelenerek **Uygun** veya **Uygun Değil** sınıfına yerleştirilmesi mümkündür. Bir teknik serviste cihazlar; çalışma sıcaklığı, titreşim düzeyi, kullanım süresi ve hata kodları dikkate alınarak bakım durumlarına göre ayrılabilir. Benzer özelliklerin farklı sınıflarda görülmesi, sınıflar arasındaki ayrımı güçleştirerek hatalı tahminlere neden olabilir. Bu nedenle sınıflandırmada kullanılacak özelliklerin problemle ilişkili ve ayırt edici olması gerekir.

4.2.2. İki Sınıflı ve Çok Sınıflı Problemler

Sınıflandırma problemleri, tahmin edilmek istenen sınıf sayısına göre iki sınıflı veya çok sınıflı olarak ele alınır. Sonucun iki farklı kategoriden biri olduğu durumlarda **iki sınıflı sınıflandırma**, üç veya daha fazla kategoriden biri olduğu durumlardasa **çok sınıflı sınıflandırma** kullanılır.

Bir ürünün kalite kontrol sonucunun **Geçti** veya **Geçemedi** olarak belirlenmesi iki sınıflı sınıflandırmaya örnektir. Bir çağrı merkezine gelen taleplerin **Teknik Destek, Faturalandırma, Satış** veya **Şikâyet** kategorilerinden birine yönlendirilmesiyse çok sınıflı sınıflandırma problemidir.

Tablo 4.2'de iki sınıflı ve çok sınıflı sınıflandırma problemlerine ilişkin örnekler verilmiştir.

Tablo 4.2: Sınıflandırma Problemlerine İlişkin Örnekler

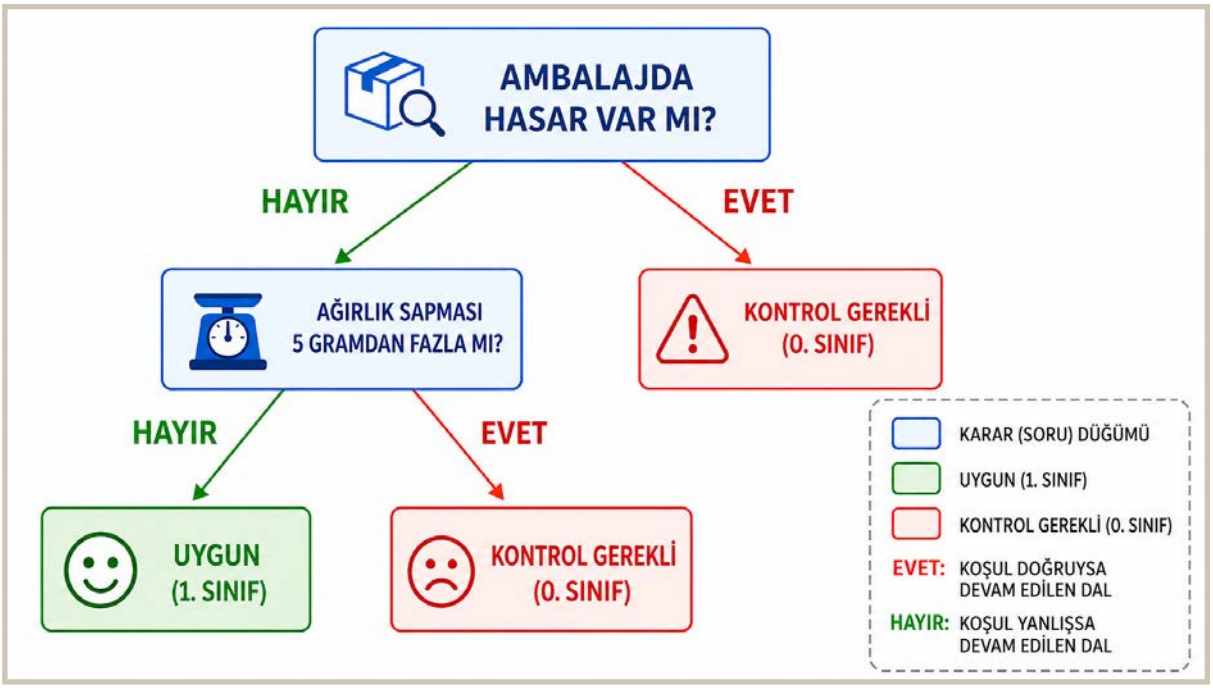
Problem Durumu	Sınıflandırma Türü	Olası Sınıflar
Ürünün kalite kontrol sonucunun belirlenmesi	İki sınıflı	Geçti / Geçemedi
Ödemenin zamanında yapılıp yapılmadığının belirlenmesi	İki sınıflı	Zamanında / Gecikmiş
Atıkların türlerine göre ayrılması	Çok sınıflı	Kâğıt / Plastik / Cam / Metal
Çağrı merkezi taleplerinin ilgili birime yönlendirilmesi	Çok sınıflı	Teknik Destek / Faturalandırma / Satış / Şikâyet

4.2.3. Karar Ağacıyla Sınıf Tahmini

Karar ağacı, verileri art arda uygulanan karar koşullarıyla sınıflara ayırır. Ağacın başlangıcında yer alan ilk koşuldan sonra veriler, koşulun sonucuna göre farklı dallara yönlendirilir. Dalların sonunda bulunan sonuç kutuları, verinin atanacağı sınıfı gösterir.

Karar koşulları oluşturulurken sınıfları ayırt etmede en etkili özellikler kullanılır. Bu yapı sayesinde modelin hangi koşulları izleyerek sonuca ulaştığı incelenebilir. Ancak ağacın gereğinden fazla dallanması, karar sürecini karmaşıklaştırabilir. Bundan dolayı ağacın derinliği probleme ve veri kümesinin yapısına uygun biçimde sınırlandırılır. Örneğin bir üretim hattındaki paketler; **ağırlık sapması** ve **ambalajda hasar bulunma durumu** dikkate alınarak **Uygun** veya **Kontrol Gerekli** sınıflarından birine yerleştirilebilir. Ambalajında hasar bulunan paketler doğrudan kontrol birimine yönlendirilirken hasar bulunmayan paketlerin ağırlık sapması incelenebilir.

Şema 4.5'te ambalaj hasarı ve ağırlık sapması ölçütlerinden yararlanılarak bir paketin kalite durumunun nasıl belirlendiği gösterilmektedir. Her koşuldan sonra izlenen dal, paketin **Uygun** veya **Kontrol Gerekli** sınıfına ulaşmasını sağlar.



Şema 4.5: Paketlerin kalite durumunu belirleyen örnek karar ağacı

Python programlama dilinde karar ağacı modeli oluşturmak için scikit-learn kütüphanesindeki **DecisionTreeClassifier** sınıfından yararlanılabilir. Paketlerin ağırlık sapması ve ambalaj hasarı bilgileri kullanılarak kalite durumunu tahmin eden bir model örneği şu şekildedir:

```
from sklearn.tree import DecisionTreeClassifier
# [Ağırlık sapması (gram), ambalaj hasarı]
# Ambalaj hasarı: 0 = Yok, 1 = Var
X = [
  [2, 0],
  [4, 0],
  [7, 0],
  [3, 1],
  [8, 1],
  [1, 0]
]
# Paketlerin kalite durumları
y = [
  "Uygun",
  "Uygun",
  "Kontrol Gerekli",
  "Kontrol Gerekli",
  "Kontrol Gerekli",
  "Uygun"
```

```
]
# Karar ağacı modelini oluşturma
model = DecisionTreeClassifier(
  max_depth=2,
  random_state=42
)
# Modeli eğitme
model.fit(X, y)
# Yeni paket için sınıf tahmini
# 6 gram ağırlık sapması, ambalaj hasarı yok
tahmin = model.predict([[6, 0]])
print("Tahmin edilen sınıf:", tahmin[0])
```

Model, ağırlık sapması ve ambalaj hasarı özelliklerinin paketlerin kalite sınıflarından ayrılmasındaki etkisini inceler. Ardından yeni pakete ait özellikleri karar koşullarından geçirerek paketin hangi sınıfta yer alacağını belirler. Kod içerisinde kullanılan bazı yapılar Tablo 4.3'te açıklanmıştır.

Tablo 4.3: Karar Ağacı Modeline Özgü Temel Kod Yapıları

Kod Yapısı	Açıklama
DecisionTreeClassifier	Sınıflandırma için karar ağacı modeli oluşturur.
max_depth	Ağacın ulaşabileceği en fazla dallanma düzeyini belirler.
random_state	Modelin aynı verilerle tekrar çalıştırıldığında aynı sonucu üretmesine yardımcı olur.
0 ve 1 değerleri	Ambalaj hasarı gibi iki durumlu bir özelliği sayısal olarak gösterir.

4.2.4. Doğruluk Oranı ve Karmaşıklık Matrisi

Sınıflandırma modelinin başarısı, tahmin edilen sınıflarla gerçek sınıfların karşılaştırılması yöntemiyle incelenir. **Doğruluk oranı,** modelin toplam tahminleri içinde kaç örneği doğru sınıflandırdığını gösterir. Bu oran, hataların hangi sınıflarda gerçekleştiğini tek başına açıklamaz.

Karmaşıklık matrisi, gerçek sınıflarla tahmin edilen sınıfları tablo hâlinde karşılaştırır. Matrisin satırlarında gerçek sınıflar, sütunlarındaysa modelin tahmin ettiği sınıflar bulunur. Böylece modelin hangi sınıflarda doğru sonuç ürettiği ve hangi sınıfları birbiriyle karıştırdığı belirlenebilir. Örneğin güneş panellerini görüntülerine göre inceleyen bir sistemin panelleri **Temiz** veya **Temizlik Gerekli** sınıflarından birine yerleştirdiği düşünülebilir. Modelin tahminleri, uzman incelemesi sonucunda belirlenen gerçek durumlarla karşılaştırılarak sistemin başarısı değerlendirilebilir.

Güneş panellerine ait örnek gerçek sonuçlar ve model tahminleri kullanılarak doğruluk oranı ile karmaşıklık matrisinin hesaplandığı kod örnekleri şu şekildedir:

```
from sklearn.metrics import accuracy_score, confusion_matrix
# Güneş panellerinin gerçek durumları
```

```
gercek = [
  "Temiz",
  "Temizlik Gerekli",
  "Temiz",
  "Temizlik Gerekli",
  "Temiz"
]
# Modelin tahmin ettiği sınıflar
tahmin = [
  "Temiz",
  "Temizlik Gerekli",
  "Temizlik Gerekli",
  "Temizlik Gerekli",
  "Temiz"
]
# Doğruluk oranını hesaplama
dogruluk = accuracy_score(gercek, tahmin)

# Karmaşıklık matrisini oluşturma
matris = confusion_matrix(
  gercek,
  tahmin,
  labels=["Temiz", "Temizlik Gerekli"]
)
# Sonuçları ekrana yazdırma
print("Doğruluk oranı:", dogruluk)
print("Karmaşıklık matrisi:")
print(matris)
```

Kod çalıştırıldığında şu sonuçlar elde edilir:

```
Doğruluk oranı: 0.8
Karmaşıklık matrisi:
[[2 1]
 [0 2]]
```

Örnekte modelin tahminleri gerçek sonuçlarla karşılaştırılmıştır. Beş tahminden dördü doğru olduğu için doğruluk oranı 0,80 olarak hesaplanır. Bu değer, modelin örnek veri kümesindeki tahminlerinin %80'inin doğru olduğunu gösterir. Kod içerisinde kullanılan değerlendirme yapıları ve bu yapıların işlevleri Tablo 4.4'te açıklanmıştır.

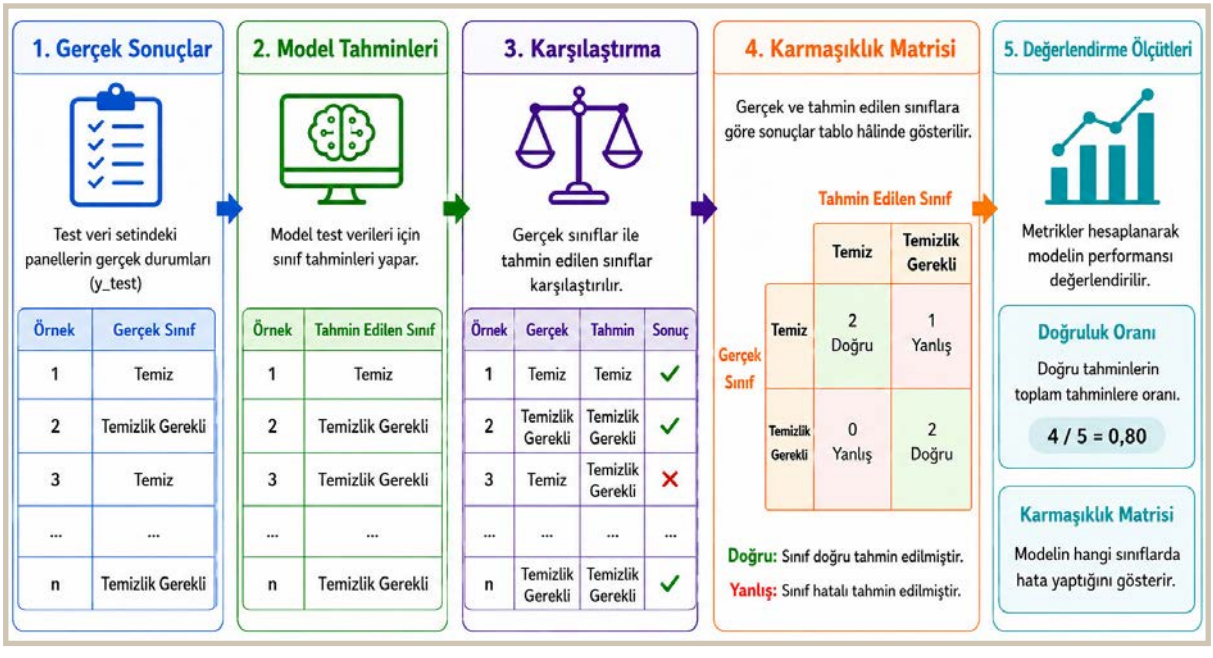
Tablo 4.4: Sınıflandırma Modelinin Değerlendirilmesinde Kullanılan Kod Yapıları

Kod Yapısı	Açıklama
accuracy_score()	Doğru tahminlerin toplam tahminlere oranını hesaplar.
confusion_matrix()	Gerçek sınıflarla tahmin edilen sınıfları karşılaştırır.
labels	Sınıfların matriste gösterilme sırasını belirler.
Matris satırları	Gerçek sınıfları gösterir.
Matris sütunları	Model tarafından tahmin edilen sınıfları gösterir.

Karmaşıklık matrisine göre gerçekte temiz olan iki panel doğru sınıflandırılmış, bir panelse yanlışlıkla **Temizlik Gerekli** olarak tahmin edilmiştir. Gerçekte temizlik gerektiren iki panelin ikisi de doğru sınıflandırılmıştır. Temizlik gerektiren hiçbir panel yanlışlıkla **Temiz** olarak değerlendirilmemiştir.

Güneş panellerinin temizliğinin enerji üretimini etkilediği uygulamalarda yalnızca doğruluk oranı değil hatanın hangi sınıfta gerçekleştiği de dikkate alınmalıdır. Temizlik gerektiren bir panelin temiz olarak sınıflandırılması, bakımın gecikmesine ve panel veriminin azalmasına neden olabilir.

Şema 4.6'da güneş panellerinin gerçek durumları ile model tahminlerinin karşılaştırılması, doğruluk oranının hesaplanması ve karmaşıklık matrisinin oluşturulması aşamaları gösterilmektedir.



Şema 4.6: Güneş paneli temizlik sınıflandırma modelinin değerlendirilme süreci

BİLGİ KUTUSU

Sınıflandırma modeli, çok sayıda örneğin bulunduğu sınıfı sürekli doğru tahmin ederek yüksek doğruluk oranına ulaşabilir. Ancak az sayıda örnek içeren sınıfı çoğu zaman yanlış belirleyebilir. Bu durum, özellikle arıza tespiti, hastalık riski, dolandırıcılık gibi seyrek görülen ancak önemli sonuçlar doğuran sınıflarda yanıltıcı olabilir. Bu nedenle model değerlendirilirken yalnızca genel doğruluk oranına değil karmaşıklık matrisinde her sınıfa ait doğru ve yanlış tahminlerin dağılımına da bakılmalıdır.

2. SIRA SİZDE

Bir tarım işletmesinde domatesler; renk, sertlik ve büyüklük özelliklerine göre "Ham", "Olgun" veya "Aşırı Olgun" sınıflarından birine yerleştirilmektedir. İşletmede kullanılan sınıflandırma modelinin gerçek sonuçları ve tahminleri aşağıdaki tabloda verilmiştir.

Ürün	Renk	Sertlik	Çap (cm)	Gerçek Sınıf	Model Tahmini
A	Yeşil	Sert	5,2	Ham	Ham
B	Kırmızı	Orta	6,4	Olgun	Olgun
C	Koyu kırmızı	Yumuşak	6,1	Aşırı Olgun	Olgun
Ç	Açık kırmızı	Sert	5,8	Ham	Ham
D	Kırmızı	Orta	6,7	Olgun	Olgun
E	Koyu kırmızı	Yumuşak	6,5	Aşırı Olgun	Aşırı Olgun

Tabloyu inceleyerek aşağıdaki çalışmaları yapınız.

1. Modelde kullanılan giriş özelliklerini ve tahmin edilen sonuç değişkenini belirleyiniz.
2. Bu problemin iki sınıflı mı yoksa çok sınıflı mı olduğunu gerekçesiyle açıklayınız.
3. Renk ve sertlik özelliklerini kullanarak ürünleri sınıflandırabilecek iki aşamalı basit bir karar ağacı kuralı oluşturunuz.
4. Modelin doğru ve yanlış sınıflandırdığı ürünleri belirleyerek doğruluk oranını hesaplayınız.
5. Gerçek sınıflarla model tahminlerini karşılaştıran karmaşıklık matrisini oluşturunuz.
6. Modelin hangi sınıfları birbiriyle karıştırdığını belirleyiniz. Bu hataların azaltılması için modele eklenebilecek bir özellik öneriniz.

Elde ettiğiniz sonuçları gerekçeleriyle sınıfta paylaşınız.

NOTLAR

4.2. ÖRNEK UYGULAMA

Adı	Toplu Taşıma Araçlarının Bakım Önceliğini Sınıflandırma
Amaç	Araç verilerini kullanarak bakım önceliğini sınıflandıran bir karar ağacı modeli oluşturarak performansını değerlendirmek
Süre	30 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlam dili, NumPy ve scikit-learn kütüphaneleri

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Yeni bir Notebook oluşturunuz ve adını **"Arac_Bakim_Siniflandirma"** olarak değiştiriniz.
- Çalışma ortamınızı uygulamaya hazır hâle getiriniz.
- Gerekli kütüphanelerin kurulu olduğunu kontrol ediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya uygun şekilde çalıştırınız.
- Dijital araçları dikkatli kullanınız.
- Çalışmalarınızı düzenli olarak kaydediniz.
- Verileri değiştirmeden önce mevcut kodların bir kopyasını saklayınız.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Gerekli kütüphaneleri ve sınıfları programa ekleyiniz.

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
```

2. Adım: Veri Setinin Oluşturulması

Toplu taşıma araçlarına ait motor sıcaklığı, titreşim düzeyi, fren balatası kalınlığı ve bakım önceliği verilerini tanımlayınız.

```
motor_sicakligi = np.array([
  68, 72, 75, 70, 74, 69,
  80, 84, 86, 82, 88, 85,
  92, 95, 97, 94, 99, 96
])
```

```
titresim = np.array([
  1.2, 1.5, 1.7, 1.3, 1.6, 1.1,
  2.0, 2.3, 2.6, 2.2, 2.8, 2.5,
  3.0, 3.3, 3.5, 3.1, 3.7, 3.4
])

fren_kalinligi = np.array([
  12, 11, 10, 12, 10, 13,
  9, 8, 7, 9, 7, 8,
  6, 5, 4, 6, 4, 5
])

bakim_onceligi = np.array([
  "Normal", "Normal", "Normal",
  "Normal", "Normal", "Normal",
  "Kontrol Gerekli", "Kontrol Gerekli",
  "Kontrol Gerekli", "Kontrol Gerekli",
  "Kontrol Gerekli", "Kontrol Gerekli",
  "Acil Bakım", "Acil Bakım", "Acil Bakım",
  "Acil Bakım", "Acil Bakım", "Acil Bakım"
])
```

3. Adım: Özellik ve Hedef Değişkenlerin Belirlenmesi

Motor sıcaklığı, titreşim düzeyi ve fren balatası kalınlığı verilerini özellik değişkenleri olarak düzenleyiniz. Bakım önceliğini hedef değişken olarak belirleyiniz.

```
X = np.column_stack((
  motor_sicakligi,
  titresim,
  fren_kalinligi
))
y = bakim_onceligi
Veri setinin ilk beş satırını inceleyiniz.
Python
Çalıştır
print("Özellik verileri:")
print(X[:5])
print("Hedef sınıflar:")
print(y[:5])
```

4. Adım: Veri Setinin Eğitim ve Test Verilerine Ayrılması

Veri setini eğitim ve test verileri olmak üzere iki bölüme ayırınız.

```
X_egitim, X_test, y_egitim, y_test = train_test_split(
  X,
```

```
    y,
    test_size=0.33,
    random_state=42,
    stratify=y
)
Eğitim ve test verilerindeki örnek sayılarını ekrana yazdırınız.
print("Eğitim verisi sayısı:", len(X_egitim))
print("Test verisi sayısı:", len(X_test))
```

5. Adım: Sınıflandırma Modelinin Oluşturulması ve Eğitilmesi

Karar ağacı sınıflandırma modelini oluşturunuz ve eğitim verileriyle eğitiniz.

```
model = DecisionTreeClassifier(
    max_depth=3,
    random_state=42
)
model.fit(X_egitim, y_egitim)
```

6. Adım: Test Verileri İçin Sınıf Tahmini Yapılması (E3.6, KB2.4, KB2.5)

Test verilerinin bakım önceliğini tahmin ediniz.

```
tahminler = model.predict(X_test)
print("Gerçek Sınıflar:")
print(y_test)
print("Tahmin Edilen Sınıflar:")
print(tahminler)
```

Gerçek sınıflar ile model tahminlerini aşağıdaki tabloya yazınız.

Test Örneği	Gerçek Sınıf	Model Tahmini	Sonuç
1.			
2.			
3.			
4.			
5.			
6.			

7. Adım: Doğruluk Oranının Hesaplanması

Modelin doğruluk oranını hesaplayınız.

```
dogruluk = accuracy_score(y_test, tahminler)
print(
    "Doğruluk Oranı:",
    round(dogruluk, 3)
```

```
)
```

Elde ettiğiniz sonucu yazınız.

Doğruluk Oranı: .....

8. Adım: Karmaşıklık Matrisinin Oluşturulması

Gerçek sınıflar ile model tahminlerini karşılaştıran karmaşıklık matrisini oluşturunuz.

```
siniflar = [
  "Normal",
  "Kontrol Gerekli",
  "Acil Bakım"
]
matris = confusion_matrix(
  y_test,
  tahminler,
  labels=siniflar
)
print("Sınıf sırası:", siniflar)
print("Karmaşıklık Matrisi:")
print(matris)
```

Elde ettiğiniz değerleri aşağıdaki matrise yazınız.

Gerçek Sınıf / Tahmin Edilen Sınıf	Normal	Kontrol Gerekli	Acil Bakım
Normal			
Kontrol Gerekli			
Acil Bakım			

9. Adım: Yeni Bir Araç İçin Sınıf Tahmini Yapılması

Motor sıcaklığı 90 °C, titreşim düzeyi 2,9 ve fren balatası kalınlığı 6 mm olan bir araç için bakım önceliğini tahmin ediniz.

```
yeni_arac = np.array([
  [90, 2.9, 6]
])
yeni_tahmin = model.predict(yeni_arac)
print(
  "Tahmin Edilen Bakım Önceliği:",
  yeni_tahmin[0]
)
```

Elde ettiğiniz tahmini yazınız.

Tahmin Edilen Bakım Önceliği: .....

10. Adım: Sonuçların Değerlendirilmesi

- Gerçek sınıflarla model tahminlerini karşılaştırınız.
- Doğru ve yanlış sınıflandırılan araç sayılarını belirleyiniz.
- Modelin en fazla hangi sınıfları birbiriyle karıştırdığını inceleyiniz.
- Doğruluk oranı ile karmaşıklık matrisini birlikte değerlendiriniz.
- Yeni araç için elde edilen bakım sınıfının ölçüm değerleriyle uyumlu olup olmadığını açıklayınız.
- Modelin araç bakım planlamasında kullanılabilirliğine karar veriniz.

Çalışmanızı kaydediniz. Oturumu güvenli şekilde kapatınız ve çalışma ortamınızı düzenli bırakınız.

Sonuç

1. **Bu uygulamada kullanılan özellik değişkenlerini ve tahmin edilen hedef değişkeni yazınız.**

   .....
   .....

2. **Uygulamadaki bakım önceliği problemi, iki sınıflı mı yoksa çok sınıflı mıdır? Gerekçesiyle açıklayınız.**

   .....
   .....

3. **Modelin test verileri üzerinde kaç doğru kaç yanlış tahmin yaptığını yazınız. Doğruluk oranını bu sonuçlarla ilişkilendirerek açıklayınız.**

   .....
   .....

4. **Karmaşıklık matrisini inceleyiniz. Modelin hangi bakım sınıflarını doğru tahmin ettiğini ve hangi sınıfları birbiriyle karıştırdığını belirtiniz.**

   .....
   .....

5. **Yalnızca doğruluk oranına bakmak yerine karmaşıklık matrisinin de incelenmesi neden gereklidir? Uygulamadaki sonuçlardan yararlanarak açıklayınız.**

   .....
   .....

6. **Motor sıcaklığı 90 °C, titreşim düzeyi 2,9 ve fren balatası kalınlığı 6 mm olan araç için elde ettiğiniz tahmini yazınız. Bu sınıflandırmanın araç bakım sürecinde nasıl kullanılabileceğini açıklayınız.**

   .....
   .....

7. **Modelin sınıflandırma başarısını artırmak için veri setine eklenebilecek başka bir araç özelliği öneriniz. Önerdiğiniz özelliğin bakım önceliğinin belirlenmesine nasıl katkı sağlayacağını açıklayınız.**

   .....
   .....

2. UYGULAMA

Adı	Atıkları Malzeme Türüne Göre Sınıflandırma
Amaç	Atıklara ait fiziksel özellikleri kullanarak malzeme türünü sınıflandıran bir karar ağacı modeli oluşturarak modelin performansını değerlendirmek
Süre	3 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek atıkları **Plastik, Cam** veya **Metal** sınıflarından birine yerleştiren sınıflandırma uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Geri dönüşüm tesisinde atıkların uygun işleme hattına yönlendirilebilmesi için malzeme türlerinin doğru belirlenmesi gerekmektedir. Atıkların ağırlık, mıknatısa tepki verme, ışık geçirme ve esneklik özellikleri; malzeme türlerinin ayrılmasına yardımcı olabilir.

İşlem Basamakları

1. Veri setini inceleyiniz.

   Atıklara ait fiziksel özellikleri ve malzeme sınıflarını aşağıdaki verilere göre belirleyiniz.

Atık	Ağırlık (g)	Mıknatısa Tepki	Işık Geçirgenliği	Esneklik	Malzeme Türü
A1	18	0	0	1	Plastik
A2	25	0	1	1	Plastik
A3	32	0	0	1	Plastik
A4	40	0	1	1	Plastik
A5	22	0	0	1	Plastik
A6	35	0	1	1	Plastik
A7	120	0	1	0	Cam
A8	145	0	1	0	Cam
A9	165	0	1	0	Cam
A10	135	0	1	0	Cam
A11	180	0	1	0	Cam
A12	150	0	1	0	Cam
A13	85	1	0	0	Metal
A14	105	1	0	0	Metal

A15	130	1	0	0	Metal
A16	95	1	0	0	Metal
A17	115	1	0	0	Metal
A18	140	1	0	0	Metal

Not: 0, ilgili özelliğin bulunmadığını, 1'se bulunduğunu gösterir.

2. Gerekli kütüphaneleri ekleyiniz.

   NumPy, train_test_split, DecisionTreeClassifier, accuracy_score ve confusion_matrix yapılarını programa dâhil ediniz.

3. Atıkların fiziksel özelliklerini NumPy dizileri hâlinde oluşturunuz.

   Ağırlık, mıknatısa tepki, ışık geçirgenliği ve esneklik verilerini ayrı diziler hâlinde tanımlayınız.

4. Özellik dizilerini birleştirerek giriş değişkenini oluşturunuz.

   Her atığa ait dört özelliği aynı satırda gösterecek biçimde X değişkenini düzenleyiniz.

5. Malzeme türlerini hedef değişken olarak belirleyiniz.

   Plastik, cam ve metal sınıflarını içeren y değişkenini oluşturunuz.

6. Veri setini eğitim ve test verileri olmak üzere iki bölüme ayırınız.

   Test verisi oranını 0.33, random_state değerini 42 olarak belirleyiniz. Sınıfların eğitim ve test verilerinde dengeli dağılması için stratify=y ifadesini kullanınız.

7. Karar ağacı sınıflandırma modelini oluşturunuz.

   DecisionTreeClassifier sınıfını kullanınız. max_depth değerini 3, random_state değerini 42 olarak belirleyiniz.

8. Modeli eğitim verileriyle eğitiniz.

   Modelin fiziksel özelliklerle malzeme sınıfları arasındaki ilişkileri öğrenmesini sağlayınız.

9. Test verileri için sınıf tahminlerini oluşturunuz.

   Modelin test verilerindeki atıkları plastik, cam veya metal olarak sınıflandırmasını sağlayınız.

10. Gerçek sınıflarla model tahminlerini karşılaştırınız.

    Elde ettiğiniz sonuçları aşağıdaki tabloya yazınız.

Test Örneği	Gerçek Sınıf	Model Tahmini	Doğru / Yanlış
1			
2			
3			
4			
5			
6			

11. Modelin doğruluk oranını hesaplayınız.

    accuracy_score yapısını kullanarak doğru tahminlerin toplam tahminlere oranını belirleyiniz.

    Doğruluk Oranı: .....

12. Karmaşıklık matrisini oluşturunuz.

Sınıf sırasını **Plastik, Cam, Metal** olarak belirleyiniz ve gerçek sınıflarla model tahminlerini karşılaştırınız.

Gerçek Sınıf / Tahmin Edilen Sınıf	Plastik	Cam	Metal
Plastik			
Cam			
Metal			

13. Doğruluk oranını ve karmaşıklık matrisini inceleyiniz.

Modelin hangi sınıflarda doğru sonuç verdiğini ve hangi sınıfları birbiriyle karıştırdığını belirleyiniz.

14. Yeni bir atığın malzeme türünü tahmin ediniz.

Ağırlığı **155 g,** mıknatısa tepki değeri **0,** ışık geçirgenliği **1** ve esneklik değeri **0** olan bir atık için sınıf tahmini yapınız.

Tahmin Edilen Malzeme Türü: .....

15. İkinci bir atık için tahmin yapınız.

Ağırlığı **100 g,** mıknatısa tepki değeri **1,** ışık geçirgenliği **0** ve esneklik değeri **0** olan atığın sınıfını tahmin ediniz.

Tahmin Edilen Malzeme Türü: .....

16. Modelin kararlarını inceleyiniz.

Modelin malzeme türlerini ayırırken hangi özelliklerden daha fazla yararlandığını belirleyiniz.

17. Sınıflandırma sonuçlarını gerçek yaşam açısından değerlendiriniz.

Yanlış sınıflandırılan bir atığın geri dönüşüm sürecinde oluşturabileceği sorunları açıklayınız.

18. Modeli geliştirmeye yönelik öneri oluşturunuz.

Veri setine eklenebilecek bir fiziksel özellik belirleyiniz ve bu özelliğin sınıflandırma başarısına nasıl katkı sağlayacağını açıklayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

4.3. KÜMELEME

Düşün, Tartış, Paylaş

Bir şehirdeki bisiklet paylaşım istasyonlarındaki kullanım durumlarının incelendiğini düşününüz. İstasyonlara ait günlük kiralama sayısı, ortalama kullanım süresi ve en yoğun kullanım saatleriyle ilgili veriler bulunmaktadır. Bazı istasyonların kullanım özelliklerinin birbirine benzediği görülmektedir.

Buna göre aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Benzer kullanım özelliklerine sahip istasyonları belirlemek için hangi veriler incelenebilir?**
2. **İstasyonlar hangi ortak özelliklerine göre gruplandırılabilir?**
3. **Oluşturulan grup sayısı değiştirildiğinde istasyonların gruplara dağılımı değişebilir mi? Neden?**
4. **Benzer istasyonların aynı grupta değerlendirilmesi bisikletlerin dağıtımı ve bakım planlaması açısından ne gibi kolaylıklar sağlayabilir?**

Kümeleme yöntemleri, önceden belirlenmiş sınıfların bulunmadığı veri kümelerinde benzer özelliklere sahip verilerin bir araya getirilmesini sağlar. Bu yöntemlerle veri kümesindeki yapılar incelenir ve veriler arasındaki yakınlıklardan yararlanılarak anlamlı gruplar oluşturulur.

Kümeleme sürecinde veriler arasındaki uzaklıkların belirlenmesi, K-Means algoritmasıyla kümelerin oluşturulması, uygun küme sayısının seçilmesi ve elde edilen sonuçların grafikler üzerinden değerlendirilmesi önem taşır.

4.3.1. Benzer Verileri Gruplandırma

Kümeleme işleminde verilerin hangi gruba ait olduğu önceden belirtilmez. Sistem, veri kayıtlarının sahip olduğu özellikleri karşılaştırır ve birbirine benzeyen kayıtları aynı kümede toplar. Böylece veri kümesi içinde doğal olarak oluşan gruplar belirlenir. Örneğin bir spor merkezindeki üyeler; haftalık ziyaret sayısı, antrenman süresi ve tercih edilen etkinlik türleri dikkate alınarak gruplandırılabilir. Düzenli gelen ve uzun süre antrenman yapan üyeler bir kümede, daha seyrek gelen üyeler başka bir kümede, belirli etkinliklere yoğun ilgi gösteren üyelerse farklı bir kümede yer alabilir.

Aynı kümede bulunan verilerin birbirine benzer, farklı kümelerde bulunan verilerin birbirinden daha farklı özellikler göstermesi beklenir. Kümeleme sonucunda oluşan gruplara başlangıçta ad verilmez. Her kümenin ortak özellikleri incelendikten sonra gruplar anlamlandırılır ve uygun biçimde adlandırılır.

Kümeleme sonucunun güvenilir olabilmesi için kullanılacak özelliklerin problemle ilişkili, ölçülebilir ve verileri birbirinden ayırabilecek nitelikte olması gerekir. Aksi durumda birbirinden farklı kayıtlar aynı kümede toplanabilir veya benzer kayıtlar farklı kümelere ayrılabilir.

4.3.2. Veriler Arasındaki Yakınlığı Belirleme

Kümeleme yöntemlerinde verilerin birbirine ne ölçüde benzediği, seçilen özellikler arasındaki uzaklıklar incelenerek belirlenebilir. Özellik değerleri birbirine yakın olan veri noktalarının benzer yapılar göstermesi, uzak olan veri noktalarının farklı gruplarda yer alması beklenir.

Örneğin hava kalitesi ölçüm istasyonları; sıcaklık, nem, parçacık yoğunluğu ve gaz ölçümleri dikkate alınarak karşılaştırılabilir. Ölçüm değerleri birbirine yakın olan istasyonlar benzer hava koşullarını temsil eden aynı kümede toplanabilir. Değerler arasındaki fark arttıkça istasyonların farklı kümelerde yer alma olasılığı da artar.

Veri noktalarının grafik üzerindeki konumları, birbirlerine olan yakınlıklarının görülmesini kolaylaştırır. Ancak yakınlık değerlendirilirken yalnızca tek bir özellik değil problemle ilişkili bütün özellikler birlikte ele alınmalıdır. Kümeleme algoritmaları bu uzaklıklardan yararlanarak veri kümesindeki benzer yapıları belirler ve kayıtları uygun gruplarda toplar.

Tablo 4.5'te farklı problem durumlarında benzer özelliklere göre oluşturulabilecek küme örnekleri verilmiştir.

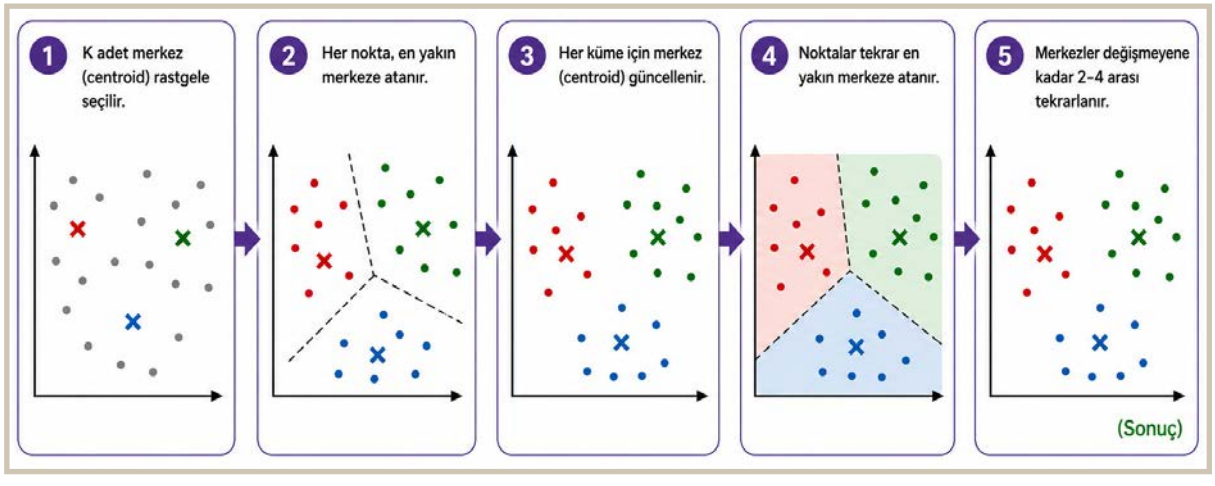
Tablo 4.5: Kümeleme Problemlerine İlişkin Örnekler

Problem Durumu	Oluşturulabilecek Kümeler
Hava kalitesi ölçüm istasyonlarının incelenmesi	Benzer ölçüm değerlerine sahip istasyonlar
Üretim makinelerinin çalışma verilerinin analizi	Benzer titreşim ve sıcaklık değerleri gösteren makineler
Şehir bölgelerindeki enerji tüketiminin değerlendirilmesi	Benzer tüketim düzenine sahip bölgeler
Tarım alanlarındaki toprak özelliklerinin incelenmesi	Benzer nem ve mineral değerlerine sahip alanlar

4.3.3. K-Means Kullanarak Küme Oluşturma

K-Means algoritması uygulanırken öncelikle oluşturulacak küme sayısı belirlenir. Algoritma, bu sayı doğrultusunda veri noktalarını kümelere ayırır ve her küme için bir merkez değeri hesaplar. Küme sayısının doğru seçilmesi, elde edilen grupların anlamlı biçimde yorumlanabilmesi açısından önemlidir.

K-Means algoritmasının merkez seçme, veri noktalarını merkezlere atama ve merkezleri güncelleme aşamaları Şema 4.7'de gösterilmiştir.



Şema 4.7: K-Means algoritmasının çalışma aşamaları

Aşağıdaki örnekte iki özelliğe sahip altı veri noktası, iki küme oluşturacak şekilde işlenmektedir.

```
import numpy as np
from sklearn.cluster import KMeans
X = np.array([
  [1, 2],
  [1, 3],
  [2, 2],
```

```
    [8, 8],
    [9, 8],
    [8, 9]
])
model = KMeans(
    n_clusters=2,
    random_state=42,
    n_init=10
)
model.fit(X)
kume_etiketleri = model.labels_
kume_merkezleri = model.cluster_centers_
print("Küme Etiketleri:", kume_etiketleri)
print("Küme Merkezleri:")
print(kume_merkezleri)
```

Kodda n_clusters=2 ifadesi, iki küme oluşturulacağını belirtir. n_init=10 ifadesi, algoritmanın farklı başlangıç merkezleriyle birden fazla kez çalıştırılmasını sağlar. Bu işlemler sonucunda elde edilen en uygun küme düzeni kullanılır.

Kod çalıştırıldığında aşağıdakine benzer bir küme etiketi çıktısı elde edilir:

[0 0 0 1 1 1]

Çıktıdaki her sayı, veri noktalarının sırasıyla hangi kümeye yerleştirildiğini gösterir. Buna göre ilk üç veri noktası bir kümede, son üç veri noktası diğer kümede yer almaktadır. Küme numaraları yalnızca oluşturulan grupları ayırt etmek amacıyla kullanılır.

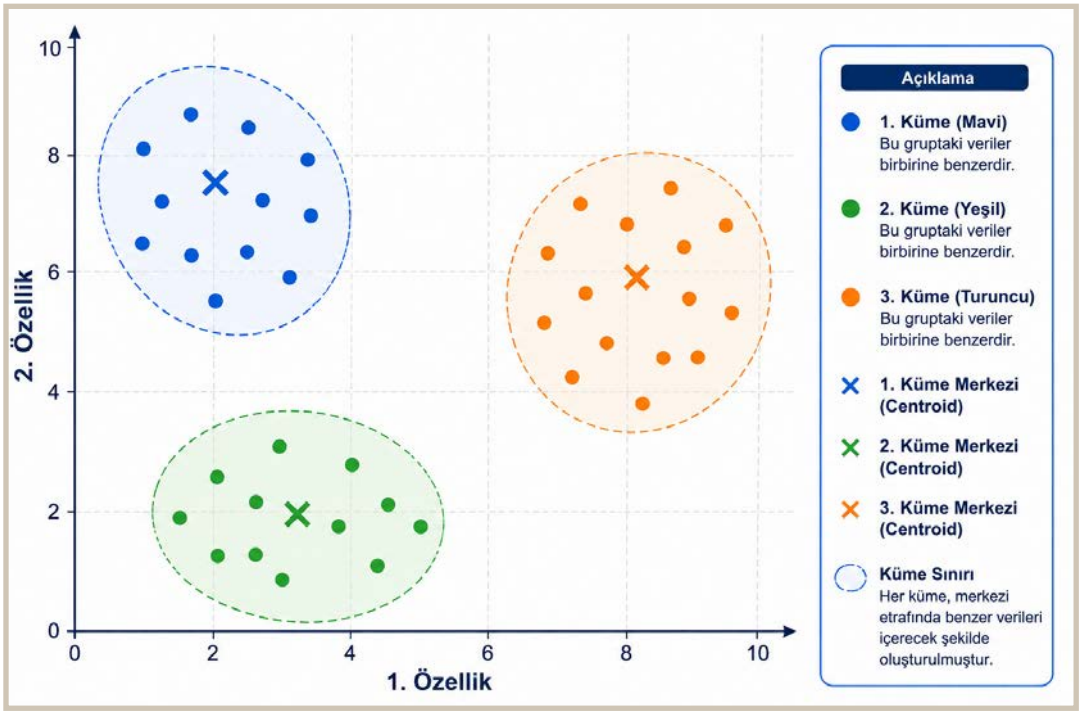
cluster_centers_ çıktısı, her kümenin iki özellik bakımından merkez değerlerini verir. Bu değerler, kümede yer alan veri noktalarının genel konumunu temsil eder.

K-Means algoritmasına özgü kod yapıları Tablo 4.6'da açıklanmıştır.

Tablo 4.6: K-Means Algoritmasına Özgü Kod Yapıları

Kod Yapısı	Açıklama
n_clusters	Oluşturulacak küme sayısını belirler.
n_init	Algoritmanın farklı başlangıç merkezleriyle kaç kez çalıştırılacağını belirler.
labels_	Veri noktalarının atandığı kümeleri gösterir.
cluster_centers_	Kümelerin merkez değerlerini verir.

Kümeleme sonucunun grafik üzerindeki görünümü Şema 4.8'de verilmiştir. Şema 4.8'de aynı renkte gösterilen noktalar aynı kümeye aittir. Çarpı işaretleri kümelerin merkezlerini, eksenlerse veri noktalarına ait iki özelliği göstermektedir. Böylece kodla elde edilen küme etiketleri grafik üzerinde görsel olarak incelenebilir.



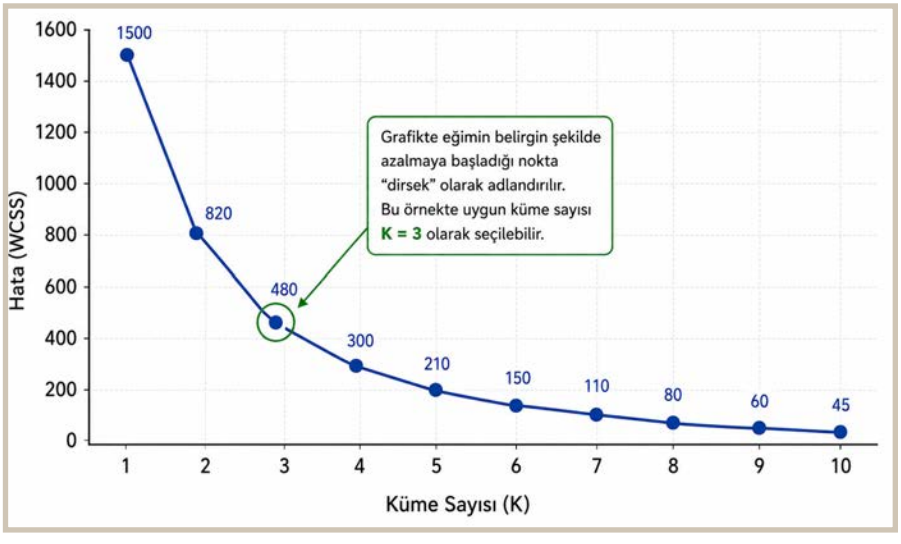
Şema 4.8: K-Means algoritmasıyla oluşturulan kümeler

K-Means uygulamasında küme etiketleri ve merkez değerleri birlikte incelenmelidir. Yalnızca küme numaralarına bakmak, kümelerin hangi özellikleri taşıdığını açıklamak için yeterli değildir. Bu nedenle her kümede yer alan veri noktalarının özellikleri değerlendirilerek kümelerin ne ifade ettiği belirlenir.

4.3.4. K Değeri ve Elbow Yöntemi

K-Means algoritmasında oluşturulacak küme sayısı **K değeri** kullanılarak belirlenir. Küme sayısı gereğinden az belirlendiğinde farklı özelliklere sahip veriler aynı kümede toplanabilir. Gereğinden fazla belirlendiğinde benzer veriler küçük ve gereksiz gruplara ayrılabilir.

Uygun K değerini belirlemek için **Elbow (Dirsek) yöntemi** kullanılabilir. Bu yöntemde farklı K değerleri için küme içindeki veri noktalarının merkezlerine olan uzaklıkları hesaplanır ve sonuçlar grafik üzerinde gösterilir. K değeri arttıkça küme içi kareler toplamı genellikle azalır. Ancak belirli bir değerden sonra bu azalma yavaşlar.



Şema 4.9: Elbow yöntemiyle uygun küme sayısının belirlenmesi

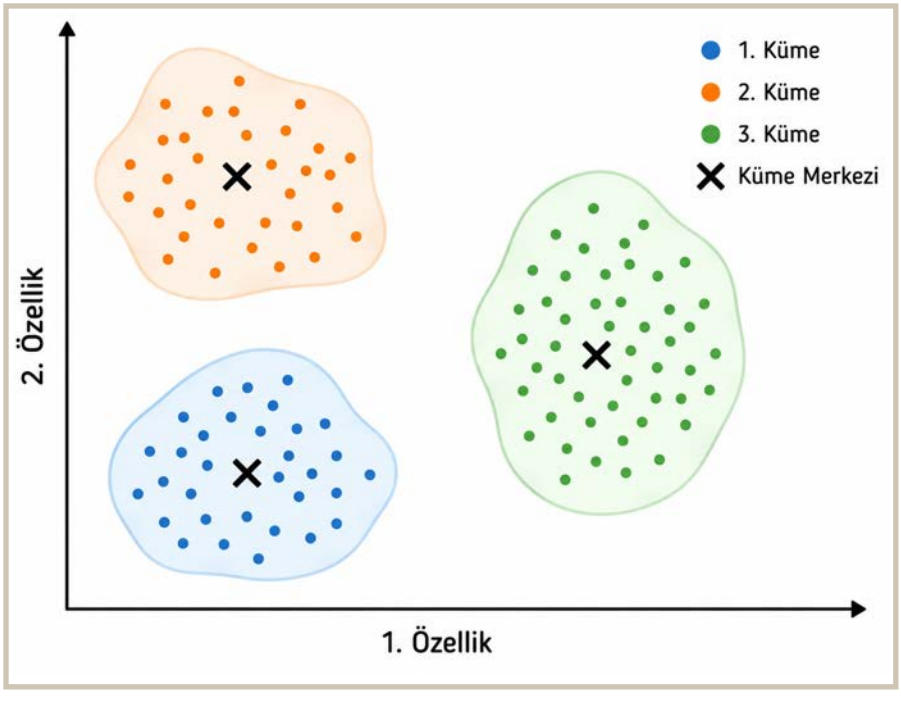
Şema 4.9'da eğrinin belirgin biçimde yön değiştirdiği nokta dirsek noktasıdır. Bu noktadaki K değeri, uygun küme sayısının belirlenmesinde yol gösterici olarak değerlendirilir. Grafikte dirsek noktası **K = 3** olduğundan verilerin üç kümede incelenmesi uygun görü-

lebilir. Bununla birlikte küme sayısı belirlenirken yalnızca grafik değil problem durumu ve oluşan kümelerin yorumlanabilirliği de dikkate alınmalıdır.

4.3.5. Kümeleri Grafik Üzerinde Yorumlama

Kümeleme sonucunda oluşan gruplar, saçılım grafiği üzerinde farklı renk veya sembollerle gösterilebilir. Grafikte birbirine yakın konumlanan ve aynı renkle belirtilen veri noktaları aynı kümeyi temsil eder. Küme merkezleri her grubun genel konumunun ve yapısının değerlendirilmesine yardımcı olur.

Şema 4.10'da hava kalitesi ölçüm noktaları, parçacık yoğunluğu ve nem değerlerine göre farklı kümelerde gösterilmektedir. Noktaların grafik üzerindeki dağılımı incelenerek aynı kümede bulunan ölçüm noktalarının ortak özellikleri, kümeler arasındaki uzaklıklar ve diğerlerinden belirgin biçimde ayrılan veriler belirlenebilir.



Şema 4.10: Kümeleme sonuçlarının saçılım grafiğinde gösterimi

Grafik yorumlanırken yalnızca renkler değil veri noktalarının birbirine yakınlığı, kümelerin yoğunluğu, küme merkezlerinin konumu ve kümeler arasındaki ayrım da dikkate alınmalıdır. Birbirinden belirgin biçimde ayrılan kümeler, veri kümesindeki farklı yapıların daha kolay ayırt edilmesini sağlar. Kümelerin iç içe geçmesi, seçilen özelliklerin grupları ayırmada yeterli olmayabileceğini gösterebilir.

Elde edilen grupların anlamlandırılabilmesi için her kümedeki verilerin özellikleri incelenmeli ve problem durumuyla ilişkilendirilmelidir. Böylece grafik yalnızca görsel bir sonuç olarak değil veriler hakkında çıkarım yapmayı sağlayan bir değerlendirme aracı olarak kullanılabilir.

BİLGİ KUTUSU

Kümeleme sonuçları her zaman kesin ve değişmez gruplar oluşturmaz. Aynı veri kümesi, kullanılan özelliklere göre farklı biçimlerde kümelenebilir. Örneğin şehirler nüfus yoğunluğuna göre gruplandırıldığında bir sonuç, enerji tüketimine göre gruplandırıldığında başka bir sonuç elde edilebilir. Bu nedenle kümeler yorumlanırken yalnızca algoritmanın çıktısı değil seçilen özelliklerin problemle ilişkisi de dikkate alınmalıdır.

3. SIRA SİZDE

Aşağıdaki müzik parçalarının tempo ve enerji düzeylerini inceleyerek soruları cevaplayınız.

Müzik Parçası	Tempo	Enerji Düzeyi
A	60	2
B	65	3
C	70	2
Ç	105	5
D	110	6
E	115	5
F	150	8
G	155	9
Ğ	160	8

Verilen bilgiler doğrultusunda aşağıdaki soruları cevaplayınız.

1. Benzer özelliklere sahip müzik parçalarını gruplandırınız.
2. Kaç küme oluşturdunuz?
3. Aynı kümede yer alan parçaları yazınız.
4. Gruplandırma yaparken hangi özellikleri kullandınız?
5. K değeri 2 olsaydı hangi kümeler birleşebilirdi?
6. Bu kümeler müzik önerisi oluşturmak için nasıl kullanılabilir?

NOTLAR

4.3. ÖRNEK UYGULAMA

Adı	Kütüphane Kullanım Verilerinin K-Means'le Kümeleme
Amaç	Kütüphane kullanım verilerini K-Means'le kümelemek, uygun K değerini belirleyerek sonuçları grafik üzerinden yorumlamak
Süre	25 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python, NumPy, Matplotlib ve Scikit-learn kütüphaneleri

Hazırlık

- Jupyter Notebook uygulamasını açınız.
- Yeni dosyanın adını **"Kutuphane_Kumeleme"** olarak belirleyiniz.
- Gerekli kütüphanelerin kurulu olduğunu kontrol ediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya göre çalıştırınız.
- Her adımda oluşan çıktıyı inceleyiniz.
- Çalışmanızı düzenli olarak kaydediniz.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

```
import numpy as np

import matplotlib.pyplot as plt

from sklearn.cluster import KMeans
```

2. Adım:Etiketsiz Verilerin Oluşturulması

Öğrencilerin aylık kütüphane ziyaret sayısı ile ödünç aldıkları kitap sayılarını tanımlayınız.

```
veriler = np.array([
  [2, 1],
  [3, 2],
  [4, 2],
  [7, 5],
  [8, 6],
  [9, 5],
  [13, 9],
  [14, 10],
  [15, 9]
])
```

Her satır sırasıyla **ziyaret** ve **ödünç alınan kitap sayısını** göstermektedir. Verilerin hangi kümeye ait olduğu önceden belirtilmemiştir.

3. Adım: Farklı K Değerlerinin Denenmesi

```
hatalar = [ ]
for k in range(1, 6):
    model = KMeans(
        n_clusters=k,
        random_state=42,
        n_init=10
    )
    model.fit(veriler)
    hatalar.append(model.inertia_)
```

4. Adım: Elbow Grafiğinin Oluşturulması

```
plt.plot(range(1, 6), hatalar, marker="o")
plt.xlabel("K Değeri")
plt.ylabel("Küme İçi Hata")
plt.xticks(range(1, 6))
plt.grid()
plt.show()
```

Grafikte hata miktarındaki azalmanın yavaşladığı noktayı belirleyiniz.

Uygun K Değeri: .....

5. Adım: K-Means Modelinin Oluşturulması

Küme sayısını 3 olarak belirleyerek modeli çalıştırınız.

```
model = KMeans(
    n_clusters=3,
    random_state=42,
    n_init=10
)
kumeler = model.fit_predict(veriler)
merkezler = model.cluster_centers_
```

Sonuçları ekrana yazdırınız.

```
print("Küme numaraları:")
print(kumeler)
print("Küme merkezleri:")
print(merkezler)
```

6. Adım: Kümelerin Grafik Üzerinde Gösterilmesi

```
plt.scatter(
  veriler[:, 0],
  veriler[:, 1],
  c=kumeler,
  s=90
)
plt.scatter(
  merkezler[:, 0],
  merkezler[:, 1],
  marker="X",
  s=220
)
plt.xlabel("Aylık Ziyaret Sayısı")
plt.ylabel("Ödünç Alınan Kitap Sayısı")
plt.grid()
plt.show()
```

Aynı renkte gösterilen noktalar aynı kümeyi, büyük çarpı işaretleriyse küme merkezlerini temsil eder.

7. Adım: Sonuçların Yorumlanması

- Aynı kümede bulunan öğrencileri belirleyiniz.
- Kümeleri ziyaret ve kitap alma sayılarına göre karşılaştırınız.
- Her kümeye ortak özelliğini yansıtan bir ad veriniz.
- K değeri 2 seçilseydi kümelerin nasıl değişebileceğini açıklayınız.

Çalışmanızı kaydediniz ve oturumu güvenli şekilde kapatınız.

Sonuç

1. **Kümeleme işleminde kullanılan iki özelliğini açıklayınız.**

   .....
   .....

2. **Elbow grafiğine göre uygun K değerini kaç olarak belirlediniz?**

   .....
   .....

3. **Oluşan kümelerin ortak özelliklerini açıklayınız.**

   .....
   .....

3. UYGULAMA

Adı	Müze Sergi Alanlarını Ziyaretçi Davranışlarına Göre Kümeleme
Amaç	Sergi alanlarına ait etiketsiz verileri K-Means algoritmasıyla gruplandırmak, uygun K değerini Elbow yöntemiyle belirleyerek oluşan kümeleri grafik üzerinde yorumlamak
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek müze sergi alanlarını günlük ziyaretçi sayısı ve ortalama inceleme süresine göre kümeleyiniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Bir müzedeki sergi alanlarının ziyaretçi yoğunluğu ve ziyaretçilerin bu alanlarda geçirdiği süre birbirinden farklı olabilir. Hangi sergi alanlarının benzer kullanım özelliklerine sahip olduğu önceden bilinmediğinde veriler kümeleme yöntemiyle gruplandırılabilir.

İşlem Basamakları

1. Veri setini inceleyiniz.

   Müze sergi alanlarına ait günlük ziyaretçi sayısı ve ortalama inceleme süresi verilerini aşağıdaki değerlere göre tanımlayınız.

   Günlük ziyaretçi sayısı.

   [40, 50, 55, 45, 85, 95, 100, 90, 145, 155, 165, 150]

   Ortalama inceleme süresi (dakika).

   [8, 10, 9, 11, 20, 22, 19, 21, 35, 38, 36, 40]
2. Gerekli kütüphaneleri ekleyiniz.

   NumPy, Matplotlib ve Scikit-learn kütüphanesindeki KMeans sınıfını programa dâhil ediniz.
3. Günlük ziyaretçi sayılarını NumPy dizisi hâline getiriniz.
4. Ortalama inceleme sürelerini NumPy dizisi hâline getiriniz.
5. Ziyaretçi sayısı ve inceleme süresi verilerini iki sütunlu bir veri dizisinde birleştiriniz.
6. Veri noktalarını inceleyerek birbirine yakın değerlere sahip sergi alanlarını belirleyiniz.
7. K değerlerini 1 ile 6 arasında değiştirerek her K değeri için K-Means modelini oluşturunuz.
8. Her K değeri için küme içi hata miktarını hesaplayınız ve bir listeye kaydediniz.
9. K değerleri ile küme içi hata miktarlarını Elbow grafiği üzerinde gösteriniz.
10. Grafikte hata miktarındaki azalmanın belirgin biçimde yavaşladığı noktayı inceleyerek uygun K değerini belirleyiniz.

11. Belirlediğiniz K değerini kullanarak K-Means modelini oluşturunuz ve veri seti üzerinde çalıştırınız.
12. Her sergi alanının atandığı küme numarasını ve küme merkezlerini ekrana yazdırınız.
13. Sergi alanlarını saçılım grafiğinde kümelerine göre farklı renklerle gösteriniz ve küme merkezlerini işaretleyiniz.
14. Aynı kümede bulunan sergi alanlarını belirleyerek kümelerin ziyaretçi sayısı ve inceleme süresi bakımından ortak özelliklerini açıklayınız.
15. K değeri 2 veya 4 olarak seçildiğinde oluşabilecek değişiklikleri karşılaştırınız. Elde edilen kümelerin ziyaretçi yönlendirme, görevli planlama ve sergi düzenleme çalışmalarında nasıl kullanılabileceğini yorumlayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

4.4. BİRLİKTELİK KURALLARI

Düşün, Tartış, Paylaş

Bir markette alışveriş yapan müşterilerin sepetlerinde bulunan ürünlerin kaydedildiğini düşününüz. Kayıtlar incelendiğinde bazı ürünlerin farklı müşteriler tarafından sık sık birlikte satın alındığı görülmektedir.

Verilen bilgiler doğrultusunda aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Hangi ürünlerin birlikte satın alındığını belirlemek için hangi bilgiler incelenebilir?**
2. **İki ürünün sık sık aynı sepette bulunması ne anlama gelebilir?**
3. **Birlikte alınan ürünlere göre markette hangi düzenlemeler yapılabilir?**
4. **Bu bilgiler müşterilere ürün önerirken nasıl kullanılabilir?**

Birliktelik kuralları, çok sayıda işlem kaydı içinde tekrar eden birlikte görülme ilişkilerini ortaya çıkaran veri analizi yöntemidir. Bu yöntemde veriler önceden belirlenmiş sınıflara ayrılmaz ve geleceğe yönelik sayısal değer tahmin edilmez. Bunun yerine, "Bir öğe görüldüğünde başka hangi öğeler de görülme eğilimindedir?" sorusuna cevap aranır.

Birliktelik analizinin amacı, yalnızca aynı kayıtta bulunan öğeleri belirlemek değildir. Bu birlikteliğin ne kadar sık gerçekleştiği, ne ölçüde güvenilir olduğu ve rastlantısal bir durumdan farklılık gösterip göstermediği de incelenir. Elde edilen sonuçlar; öneri oluşturma, stok düzenleme, hizmet planlama, kullanıcı davranışlarını anlamlandırma gibi çalışmalarda kullanılabilir.

İşlem kayıtlarının hazırlanması, sık görülen öğe gruplarının belirlenmesi, destek, güven ve kaldıraç ölçütlerinin yorumlanması ile Apriori algoritmasının temel çalışma mantığı ele alınacaktır.

4.4.1. Veriler Arasındaki Birlikte Görülme İlişkileri

Günlük yaşamda bazı durumlar tek başına değil başka durumlarla birlikte ortaya çıkabilir. Bu ilişkiler az sayıda kayıt incelendiğinde kolayca fark edilmeyebilir. Ancak çok sayıda kayıt bir arada değerlendirildiğinde tekrar eden örüntüler belirgin hâle gelir.

Örneğin afet uyarısı öncesindeki alışveriş kayıtları incelendiğinde su, pil ve el feneri gibi temel ihtiyaçların yanında uzun süre saklanabilen ve hazırlama gerektirmeyen bazı yiyeceklerin de sıkça satın alındığı görülebilir. İlk bakışta dikkat çekmeyen bu birliktelikler, işlem kayıtlarının sistemli biçimde incelenmesiyle ortaya çıkarılabilir.

Birlikte görülme, iki durum arasında kesin bir neden sonuç ilişkisi bulunduğunu göstermez. Yalnızca belirli öğelerin aynı işlem kayıtlarında beklenenden daha sık yer aldığını gösterir (Tablo 4.7).

Tablo 4.7: Birlikte Görülme İlişkilerine İlişkin Örnekler

Veri Kaynağı	Belirlenebilecek İlişki
Afet hazırlık kayıtları	Aynı hazırlık paketinde yer alan ürünler
Araç bakım kayıtları	Aynı bakım sırasında değiştirilen parçalar
Sağlık başvuru kayıtları	Birlikte bildirilen yakınmalar
Müze ziyaret kayıtları	Aynı ziyaret sırasında gezilen bölümler
Dijital hizmet kayıtları	Aynı oturumda kullanılan hizmetler

Örneklerin ortak özelliği, her kayıtta birden fazla öğenin bulunması ve bazı öğelerin farklı kayıtlarda tekrar tekrar birlikte görülmesidir.

4.4.2. Birliktelik Analizi İçin Veriyi Düzenleme

Birliktelik analizi yapılabilmesi için verilerin işlem kayıtları biçiminde düzenlenmesi gerekir. Belirli bir zamanda veya tek bir olay kapsamında bir araya gelen öğelerin tamamı bir işlem kaydını oluşturur. Kayıt içinde bulunan her unsur öğe olarak adlandırılır. Örneğin bir afet hazırlık merkezinde aynı pakete konulan malzemeler tek bir işlem kaydı olarak değerlendirilebilir.

```
islem_kayitlari = [
  ["Su", "Pil", "Fener"],
  ["Su", "Konserve"],
  ["Su", "Pil", "Fener", "İlk Yardım"],
  ["Konserve", "İlk Yardım"],
  ["Su", "Pil", "Fener"]
]
print(islem_kayitlari)
```

Veri yapısında her liste bir işlem kaydını; "Su", "Pil", "Fener", "Konserve" ve "İlk Yardım" ifadeleriyse öğeleri temsil eder. Aynı listede bulunan öğeler, birlikte gerçekleşen seçimleri gösterir.

Birliktelik analizinde verilerin sayısal büyüklüklerinden çok, her öğenin işlem kayıtlarında bulunup bulunmadığı incelenir. Bu nedenle işlem kayıtları, öğelerin varlığını doğru yanlış değerleriyle gösteren tablo yapısına dönüştürülebilir. Örnek işlem kayıtlarının bu biçimde düzenlenmiş hâli Tablo 4.8'de verilmiştir.

Tablo 4.8 incelendiğinde "Su", "Pil" ve "Fener" öğelerinin birden fazla işlem kaydında birlikte yer aldığı görülmektedir. Bu düzenleme, sık tekrarlanan öğe ve öğe gruplarının algoritma tarafından daha kolay belirlenmesini sağlar.

Tablo 4.8: İşlem Kayıtlarında Öğelerin Bulunma Durumu

İşlem	Su	Pil	Fener	Konserve	İlk Yardım
1	✓	✓	✓		
2	✓			✓	
3	✓	✓	✓		✓
4				✓	✓
5	✓	✓	✓		

4.4.3. Birliktelik Kurallarını Ölçme

Veriler arasında bir ilişki belirlenmesi, bu ilişkinin her zaman güçlü veya anlamlı olduğu anlamına gelmez. Birliktelik kurallarının değerlendirilmesinde destek, güven ve kaldıraç ölçütlerinden yararlanılabilir. Destek, bir öğenin veya öğe grubunun tüm işlem kayıtları içerisinde ne sıklıkla görüldüğünü gösterir. Güven, bir öğe görüldüğünde başka bir öğenin de görülme olasılığını ifade eder. Kaldıraç, iki öğenin birlikte görülme ilişkisinin rastlantısal bir birliktelikten ne ölçüde farklı olduğunu değerlendirmeye yardımcı olur.

Örneğin bir öğrenme platformunda "Python Temelleri" içeriğini inceleyen öğrencilerin büyük bir bölümünün "Veri Analizi" içeriğini de incelemesi iki içerik arasında bir birliktelik bulunduğunu gösterebilir. Bu ilişkinin ne kadar sık ve güçlü olduğu destek, güven ve kaldıraç değerleri kullanılarak değerlendirilebilir. Birliktelik kurallarının değerlendirilmesinde kullanılan destek, güven ve kaldıraç ölçütleri Tablo 4.9'da özetlenmiştir.

Tablo 4.9: Birliktelik Kurallarının Değerlendirilmesinde Kullanılan Temel Ölçütler

Ölçüt	Açıklama
Destek	Öğe veya öğe grubunun işlem kayıtlarında görülme sıklığını gösterir.
Güven	Bir öğe görüldüğünde diğer öğenin de görülme olasılığını gösterir.
Kaldıraç	Öğeler arasındaki ilişkinin rastlantısal birliktelikten ne ölçüde farklı olduğunu değerlendirmeye yardımcı olur.

Birliktelik kuralı şu biçimde gösterilebilir:

Bilim ⟶ Teknoloji

Bu ifade, "Bilim" öğesinin görüldüğü işlem kayıtlarında "Teknoloji" öğesinin de görülme eğiliminin incelendiğini gösterir.

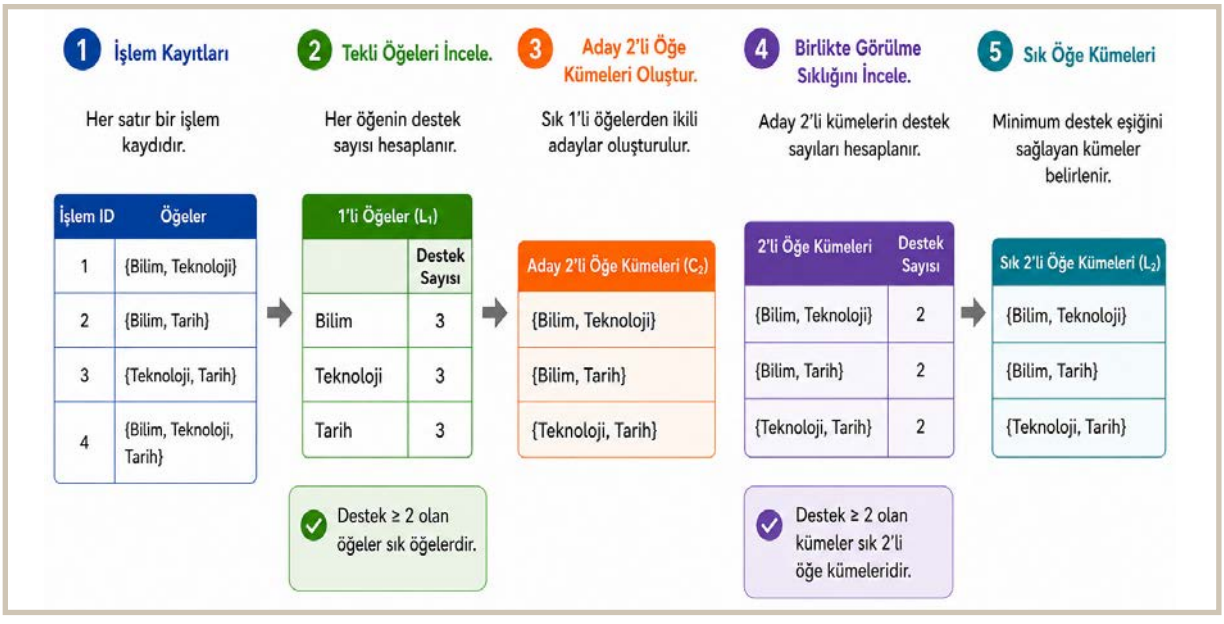
4.4.4. Sık Görülen Öğe Kümelerini Bulma

Birliktelik kurallarının oluşturulmasında yaygın olarak kullanılan yöntemlerden biri **Apriori algoritmasıdır**. Bu algoritma, işlem kayıtlarında belirlenen minimum destek değerini karşılayan öğeleri ve öğe gruplarını aşamalı olarak bulur.

İlk aşamada her öğenin işlem kayıtlarında kaç kez yer aldığı hesaplanır. Minimum destek değerini karşılayan tekli öğeler sık öğe olarak kabul edilir. Ardından bu öğelerden ikili aday gruplar oluşturulur ve her grubun birlikte görülme sayısı belirlenir. Destek eşiğinin altında kalan gruplar elenirken eşiği karşılayanlar sık öğe kümeleri olarak seçilir.

Apriori algoritması, sık olmayan bir öğe grubunun daha büyük gruplar oluşturmak için kullanılmayacağı ilkesine dayanır. Böylece tüm olası birleşimler yerine yalnızca yeterli sıklığa sahip gruplar incelenir ve işlem yükü azaltılır.

Şema 4.11'de, örnek işlem kayıtlarından tekli öğelerin incelenmesi, aday ikili grupların oluşturulması, destek sayılarının hesaplanması ve sık görülen öğe kümelerinin belirlenmesi adım adım gösterilmektedir. Şemadaki örnekte minimum destek değeri 2 olarak alınmış; "Bilim-Teknoloji", "Bilim-Tarih" ve "Teknoloji-Tarih" ikililerinin bu eşiği karşıladığı görülmüştür.



Şema 4.11: Apriori algoritmasıyla sık öğe kümelerinin belirlenmesi

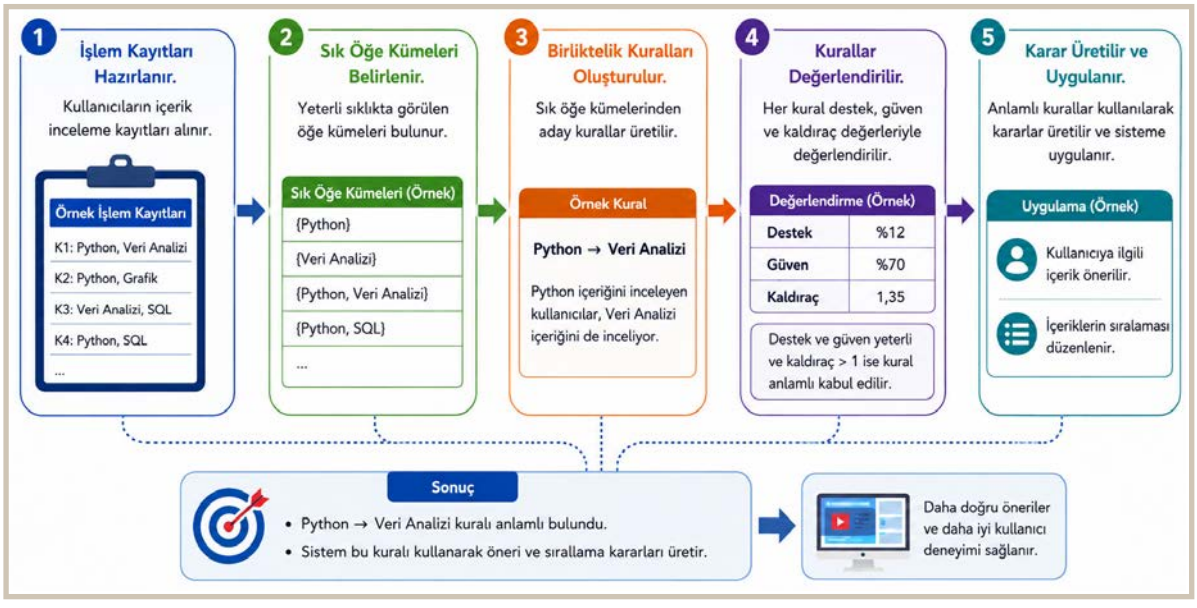
4.4.5. Birliktelik Kurallarından Karar Üretme

Sık görülen öğe kümeleri belirlendikten sonra bu öğeler arasındaki ilişkilerden birliktelik kuralları oluşturulur. Her kuralın anlamlı olup olmadığı destek, güven ve kaldıraç değerleri birlikte incelenerek değerlendirilir. Örneğin bir çevrim içi öğrenme platformunda şu kural elde edilebilir:

Python → Veri Analizi

Kural, "Python" içeriğini inceleyen kullanıcıların "Veri Analizi" içeriğine de yönelme eğiliminde olduğunu gösterir. Kural yeterli destek ve güven değerine sahipse kullanıcıya ilgili içerik önerilebilir veya içeriklerin sıralaması buna göre düzenlenebilir. Kaldıraç değerinin 1'den büyük olmasıyla iki içeriğin rastlantısal beklentiden daha güçlü bir ilişki gösterdiğine işaret eder.

Birliktelik kuralları kesin bir neden sonuç ilişkisi oluşturmaz. Bu nedenle kurallar, veri kümesinin özellikleri ve hesaplanan ölçütlerle birlikte yorumlanmalıdır. İşlem kayıtlarının hazırlanmasından karar üretilmesine kadar izlenen aşamalar Şema 4.12'de gösterilmiştir.



Şema 4.12: Birliktelik kuralı oluşturma ve karar üretme süreci

BİLGİ KUTUSU

Afet uyarılarından önce yapılan alışverişler incelendiğinde yalnızca su, pil, el feneri gibi temel ihtiyaçların değil pişirme gerektirmeyen tatlı atıştırmalıkların da daha fazla tercih edildiği belirlenmiştir. Çok sayıdaki alışveriş kaydında gizlenen bu beklenmedik örüntü, birliktelik analiziyle ortaya çıkarılabilir. Böylece işletmeler, olağan dışı dönemlerde hangi ürünlere daha fazla ihtiyaç duyulabileceğini önceden değerlendirerek stoklarını daha doğru planlayabilir. Ancak birlikte satın alınan ürünler arasında mutlaka neden sonuç ilişkisi bulunduğu düşünülmemelidir.

3. SIRA SİZDE

Bir okulun teknoloji şenliğinde öğrencilerin aynı gün katıldıkları atölyeler kaydedilmiştir. Aşağıdaki tabloda işlem kayıtlarını inceleyiniz. Minimum destek sayısını 3 olarak kabul ediniz.

İşlem Kaydı	Katılınan Atölyeler
K1	Robotik, 3B Tasarım
K2	Robotik, Kodlama, 3B Tasarım
K3	Kodlama, Oyun Tasarımı
K4	Robotik, Kodlama, 3B Tasarım
K5	Robotik, 3B Tasarım
K6	Kodlama, Oyun Tasarımı
K7	Robotik, Kodlama, 3B Tasarım
K8	Robotik, Kodlama

Verilen bilgiler doğrultusunda aşağıdaki soruları cevaplayınız.

1. Veri kümesindeki öğeleri belirleyiniz.
2. Her öğenin işlem kayıtlarında kaç kez yer aldığını hesaplayınız. Minimum destek sayısını karşılayan sık öğeleri yazınız.
3. Sık öğelerden oluşturulabilecek ikili aday kümeleri belirleyiniz. Destek sayısı en az 3 olan sık ikili öğe kümelerini yazınız.
4. **Robotik** → **3B Tasarım** kuralının destek ve güven değerlerini hesaplayınız.
5. Kuralın kaldıraç değerinin 1'den büyük olduğu belirlenmiştir. Bu sonuç, iki atölye arasındaki ilişki hakkında ne gösterir?
6. "Oyun Tasarımı" öğesi minimum destek sayısını karşılamamaktadır. Apriori algoritmasına göre bu öğeyi içeren daha büyük kümelerin neden elenebileceğini açıklayınız.

Sonuçlarınızı sınıfta arkadaşlarınızla paylaşınız.

4.4. ÖRNEK UYGULAMA

Adı	Spor Merkezi Etkinlik Tercihlerinden Birliktelik Kuralları Oluşturma
Amaç	Spor merkezindeki etkinlik tercihlerinde sık görülen birliktelikleri Apriori algoritmasıyla belirleyerek kuralları destek, güven ve kaldıraç değerlerine göre yorumlamak
Süre	25 dakika
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, Pandas ve mlxtend kütüphaneleri

Hazırlık

- Jupyter Notebook uygulamasını açınız.
- Yeni dosyanın adını **"Etkinlik_Birliktelik_Analizi"** olarak belirleyiniz.
- Pandas ve mlxtend kütüphanelerinin kurulu olduğunu kontrol ediniz.
- Çalışmada gerçek kullanıcı bilgileri yerine verilen örnek kayıtları kullanınız.

Uygulama Sırasında Uyulması Gereken Kurallar

- Kodları verilen sıraya göre çalıştırınız.
- Her adımda elde ettiğiniz çıktıları inceleyiniz.
- Destek, güven ve kaldıraç değerlerini birlikte değerlendiriniz.
- Birliktelik kurallarını kesin neden-sonuç ilişkisi olarak yorumlamayınız.
- Çalışmanızı düzenli olarak kaydediniz.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Gerekli kütüphaneleri ve yapıları programa ekleyiniz.

```
import pandas as pd
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori, association_rules
```

2. Adım: İşlem Kayıtlarının Oluşturulması

Spor merkezi üyelerinin aynı gün katıldıkları etkinlikleri tanımlayınız.

```
islem_kayitlari = [
  ["Yüzme", "Pilates", "Sauna"],
  ["Yüzme", "Sauna"],
  ["Pilates", "Yoga"],
  ["Yüzme", "Pilates", "Sauna"],
  ["Pilates", "Yoga"],
  ["Yüzme", "Sauna"],
  ["Yüzme", "Pilates", "Sauna"],
  ["Pilates", "Yoga", "Sauna"]
]
```

Her liste bir işlem kaydını, listedeki ifadeler aynı gün tercih edilen etkinlikleri göstermektedir.

3. Adım: İşlem Kayıtlarının Tabloya Dönüştürülmesi

İşlem kayıtlarını, etkinliklerin bulunma durumunu gösteren tablo yapısına dönüştürünüz.

```
donusturucu = TransactionEncoder()
kodlanmis_veri = donusturucu.fit(
  islem_kayitlari
).transform(islem_kayitlari)
veri = pd.DataFrame(
  kodlanmis_veri,
  columns=donusturucu.columns_
)
print(veri.astype(int))
```

Çıktıda 1 değeri etkinliğin ilgili işlem kaydında bulunduğunu, 0 değeri bulunmadığını gösterir.

4. Adım: Sık Görülen Öğe Kümelerinin Belirlenmesi

Minimum destek değerini 0.30 olarak belirleyerek sık görülen etkinlikleri ve etkinlik gruplarını bulunuz.

```
oge_kumeleri = apriori(
  veri,
  min_support=0.30,
  use_colnames=True
)
oge_kumeleri["Destek (%)"] = (
```

```
    oge_kumeleri["support"] * 100
  ).round(1)
  print(
    oge_kumeleri[
      ["itemsets", "Destek (%)"]
    ]
  )
```

Destek değeri en az %30 olan öğe kümelerini inceleyiniz.

5. Adım: Birliktelik Kurallarının Oluşturulması

Güven değeri en az 0.60 olan birliktelik kurallarını oluşturunuz.

```
  kurallar = association_rules(
    oge_kumeleri,
    metric="confidence",
    min_threshold=0.60
  )
```

Kuralların destek, güven ve kaldıraç değerlerini hesaplayınız.

```
  kurallar["Destek (%)"] = (
    kurallar["support"] * 100
  ).round(1)
  kurallar["Güven (%)"] = (
    kurallar["confidence"] * 100
  ).round(1)
  kurallar["Kaldıraç"] = (
    kurallar["lift"]
  ).round(2)
```

6. Adım: Kuralların Okunaklı Hâle Getirilmesi (KB2.5, KB2.13)

Kuralları daha kolay inceleyebilmek için öğe kümelerini metne dönüştürünüz.

```
  def oge_metni(oge_kumesi):
    return ", ".join(sorted(oge_kumesi))
  kurallar["Kural"] = kurallar.apply(
    lambda satir:
    oge_metni(satir["antecedents"])
    + " → "
    + oge_metni(satir["consequents"]),
    axis=1
  )
  sonuclar = kurallar[
    [
      "Kural",
      "Destek (%)",
      "Güven (%)",
      "Kaldıraç"
```

```
    ]
]
print(sonuclar)
```

7. Adım: Kuralların Değerlendirilmesi

Elde edilen kuralları aşağıdaki ölçütlere göre inceleyiniz.

- Destek değeri yüksek olan kuralları belirleyiniz.
- Güven değeri en yüksek olan kuralı bulunuz.
- Kaldıraç değeri 1'den büyük olan kuralları belirleyiniz.
- Yüzme → Sauna kuralının ne anlama geldiğini açıklayınız.
- Birlikte tercih edilen etkinliklerin kesin neden-sonuç ilişkisi oluşturup oluşturmadığını değerlendiriniz.

8. Adım: Kurallardan Karar Üretilmesi

Elde ettiğiniz sonuçlardan yararlanarak spor merkezi için bir karar öneriniz. Aşağıdaki alanlardan birini seçebilirsiniz.

- Etkinlik saatlerinin düzenlenmesi
- Ortak üyelik paketi hazırlanması
- Salon ve araç gereç planlaması
- Üyelere etkinlik önerilmesi

Önerinizi gerekçesiyle yazınız.

Karar Önerisi

.....

Çalışmanızı kaydediniz ve oturumu güvenli şekilde kapatınız.

Sonuç

1. **Uygulamadaki işlem kayıtları ve öğeler nelerdir? Açıklayınız.**

   .....
   .....

2. **Minimum destek değerini karşılayan sık öğe kümelerinden ikisini yazınız.**

   .....
   .....

3. **Güven değeri en yüksek olan birliktelik kuralını belirleyerek ne anlama geldiğini açıklayınız.**

   .....
   .....

4. **Kaldıraç değerinin 1'den büyük olması öğeler arasındaki ilişki hakkında ne gösterir? Açıklayınız.**

   .....
   .....

5. **Elde edilen kurallar spor merkezinin etkinlik planlamasında nasıl kullanılabilir? Açıklayınız.**

   .....
   .....

4. UYGULAMA

Adı	Elektronik Proje Bileşenleri Arasındaki Birliktelikleri Belirleme
Amaç	Elektronik bileşenler arasındaki ilişkileri Apriori algoritmasıyla belirleyerek sonuçları malzeme planlamasında kullanmak
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek elektronik projelerde kullanılan bileşenlere ait işlem kayıtları üzerinde birliktelik kuralları uygulamasını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Bir okulun teknoloji atölyesinde öğrencilerin hazırladığı elektronik projelerde kullanılan bileşenler kayıt altına alınmaktadır. Bazı projelerde LED ve direnç, bazı projelerde buton ve buzzer gibi bileşenler birlikte kullanılmaktadır. Ancak çok sayıdaki proje kaydı incelendiğinde hangi bileşenlerin anlamlı biçimde birlikte kullanıldığını doğrudan fark etmek zor olabilir.

Her projede kullanılan bileşenler bir **işlem kaydı,** kayıtlarda bulunan LED, direnç, buton, buzzer ve ışık sensörü gibi bileşenler **öğe** olarak değerlendirilebilir.

İşlem Basamakları

1. İşlem kayıtlarını inceleyiniz.

   Elektronik projelerde kullanılan bileşenleri aşağıdaki verilere göre tanımlayınız.

Proje Kaydı	Kullanılan Bileşenler
P1	LED, Direnç, Buton
P2	LED, Direnç, Işık Sensörü
P3	LED, Direnç, Buton, Buzzer
P4	Buton, Buzzer
P5	LED, Direnç
P6	LED, Direnç, Işık Sensörü
P7	Direnç, Buton, Buzzer
P8	LED, Direnç, Buton
P9	LED, Direnç, Işık Sensörü
P10	Buton, Buzzer

Tablodaki işlem kayıtlarını ve veri kümesinde bulunan öğeleri belirleyiniz.

2. Gerekli kütüphaneleri ekleyiniz.
3. İşlem kayıtlarını Python listesi hâlinde oluşturunuz.

   Her proje kaydını ayrı bir liste olacak şekilde düzenleyiniz. Bütün proje kayıtlarını tek bir ana liste içerisinde birleştiriniz.
4. İşlem kayıtlarını tablo yapısına dönüştürünüz.

   TransactionEncoder yapısını kullanarak işlem kayıtlarını bileşenlerin bulunma durumunu gösteren doğru-yanlış değerlerine dönüştürünüz. Elde edilen verilerden bir DataFrame oluşturunuz.
5. Dönüştürülen verileri inceleyiniz.

   Doğru-yanlış değerlerini 1 ve 0 biçiminde ekrana yazdırınız.

- 1 değerinin ne anlama geldiğini açıklayınız.
- 0 değerinin ne anlama geldiğini açıklayınız.
- En fazla proje kaydında yer alan bileşeni belirleyiniz.

6. Sık öğe kümelerini belirleyiniz.

   Apriori algoritmasını kullanarak minimum destek değerini 0.30 olarak belirleyiniz. Bu değeri karşılayan tekli, ikili ve daha büyük öğe kümelerini bulunuz.

   On işlem kaydı bulunduğuna göre 0.30 destek değerinin en az kaç işlem kaydına karşılık geldiğini hesaplayınız.
7. Sık öğe kümelerinin destek değerlerini inceleyiniz.

   Elde edilen öğe kümelerini destek değerlerine göre büyükten küçüğe sıralayınız.

- Destek değeri en yüksek tekli öğeyi belirleyiniz.
- En sık görülen ikili öğe kümesini bulunuz.
- Minimum destek değerini karşılamayan öğe kümelerinin neden elendiğini açıklayınız.

8. Birliktelik kurallarını oluşturunuz.

   Sık öğe kümelerinden birliktelik kuralları oluşturunuz. Kural oluştururken ölçütü confidence olarak, minimum güven değerini 0,70 olarak belirleyiniz.
9. Kuralların ölçüm değerlerini görüntüleyiniz.

   Oluşturulan kurallara ait şu bilgileri ekrana yazdırınız:

- Öncül öğe veya öğeler
- Sonuç öğe veya öğeler
- Destek
- Güven
- Kaldıraç

   Sonuçları destek veya kaldıraç değerine göre sıralayınız.
10. Birliktelik kurallarını yorumlayınız.

    Elde ettiğiniz sonuçlar arasında bulunuyorsa şu kuralları inceleyiniz:

- LED → Direnç
- Buzzer → Buton
- Işık Sensörü → LED

Her kural için aşağıdaki soruları cevaplayınız.

- Kuralın destek değeri neyi göstermektedir?
- Kuralın güven değeri nasıl yorumlanabilir?
- Kuralın yönü ters çevrildiğinde güven değeri değişmekte midir?
- Kuralın kaldıraç değeri, öğeler arasındaki ilişki hakkında ne göstermektedir?

11. Kaldıraç değerlerini değerlendiriniz.

Kuralları kaldıraç değerlerine göre inceleyiniz.

- Kaldıraç değeri 1'den büyük olan kuralları belirleyiniz.
- Kaldıraç değeri 1'e yakın olan bir kuralın nasıl yorumlanacağını açıklayınız.
- Güven değeri yüksek olduğu hâlde kaldıraç değeri düşük olan bir kuralın doğrudan kullanılmasının uygun olup olmadığını tartışınız.

12. Minimum destek değerini değiştiriniz.

Minimum destek değerini 0,30 yerine 0,40 olarak belirleyerek Apriori algoritmasını yeniden çalıştırınız.

İki uygulamanın sonuçlarını karşılaştırınız.

- Sık öğe kümelerinin sayısı nasıl değişmiştir?
- Hangi öğe kümeleri elenmiştir?
- Destek eşiğinin artırılması sonuçları nasıl etkilemiştir?

13. Sonuçlardan karar üretiniz.

Elde ettiğiniz kurallardan yararlanarak teknoloji atölyesi için aşağıdaki alanlardan birine yönelik karar oluşturunuz.

- Birlikte verilecek proje malzemesi setleri
- Bileşenlerin dolaplardaki yerleşimi
- Stokta daha fazla bulundurulacak bileşenler
- Proje hazırlama masalarına bırakılacak ortak malzemeler

Kararınızı dayandırdığınız birliktelik kuralını ve ölçüm değerlerini belirtiniz.

Karar Önerisi

.....

Kararın Dayandığı Kural

.....

14. Kuralların sınırlarını değerlendiriniz.

1. İki bileşenin projelerde sıkça birlikte kullanılması, bir bileşenin diğerinin kullanılmasına kesin olarak neden olduğunu gösterir mi? Açıklayınız.
2. Destek, güven ve kaldıraç değerlerinden yararlanarak açıklayınız.

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

4.5. VAKA ÇALIŞMALARI

Düşün, Tartış, Paylaş

Bir fabrikada üretim bandındaki bir makine, haftalar önce sessiz bir sinyal vermeye başlar. Titreşim değerleri yavaş yavaş yükselir, sıcaklık normalin biraz üstüne çıkar. Deneyimli bir teknisyen bunu fark edip yorum yapabilir ancak gece vardiyasında teknisyen bulunmamaktadır.

Verilen senaryo doğrultusunda aşağıdaki soruları cevaplayınız. Düşüncelerinizi sınıfta arkadaşlarınızla paylaşınız.

1. **Bu problemi çözmek için yapay zekâya hangi verilerin verilmesi gerekir?**
2. **Bu verilere göre regresyon, sınıflandırma, kümeleme yöntemlerinden hangisini seçerdiniz? Nedeninizi açıklayınız.**
3. **Modelin çıktısı ne olmalıdır? Hangi kararı kim vermelidir?**

Yapay zekâ günümüzde yalnızca araştırma laboratuvarlarında geliştirilen bir teknoloji değildir. Sağlıktan eğitime, tarımdan üretime, finanstan ulaşıma kadar birçok alanda gerçek problemlerin çözümünde kullanılmaktadır. Ancak bir yapay zekâ sistemini başarılı yapan yalnızca kullanılan algoritma değildir. Öncelikle çözülmek istenen problemin doğru tanımlanması, uygun verilerin toplanması ve probleme en uygun yöntemin seçilmesi gerekir.

Aşağıdaki sorular üzerinde düşününüz.

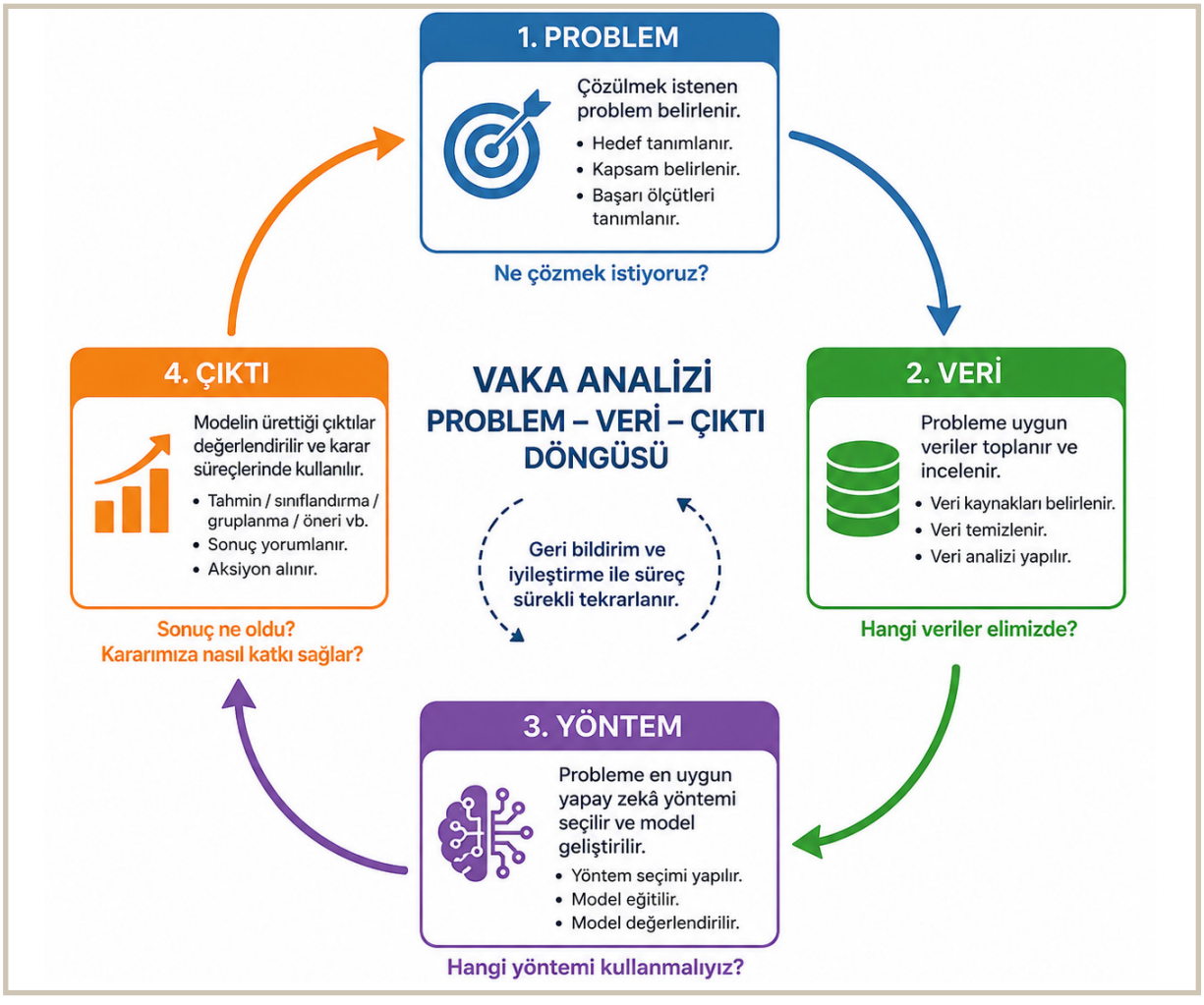
- Doktor olmayan bir yazılım kanseri teşhis edebilir mi?
- Direksiyonu olmayan bir taksiye biner misiniz?
- Bankalar şüpheli kredi kartı işlemlerini saniyeler içinde nasıl fark edebiliyor?

Bu soruların ortak noktası, hepsinin arkasında büyük miktarda veriyi analiz eden yapay zekâ sistemlerinin bulunmasıdır. Ancak bu sistemlerin tamamı aynı yöntemle çalışmaz. Çözülmek istenen probleme göre farklı yapay zekâ yaklaşımları kullanılır. Bazı problemlerde bir değerin tahmin edilmesi gerekirken, bazılarında verilerin sınıflandırılması, benzer özelliklerine göre gruplandırılması veya birlikte gerçekleşen durumların belirlenmesi amaçlanır.

Vaka analizi, gerçek yaşamda karşılaşılan bir problemin sistemli bir şekilde incelenmesidir. Vaka analizi yapılırken aşağıdaki sorulara cevaplar aranmalıdır.

- Çözülmek istenen problem nedir?
- Problemi çözmek için hangi verilere ihtiyaç duyulur?
- Problemin çözümüne en uygun yöntem hangisidir?
- Üretilen sonuçlardan ne elde edilebilir?

Doğru problem tanımı, kaliteli veri, uygun yöntem ve anlamlı çıktı başarılı bir yapay zekâ çözümünün temelidir (Şema 4.13).



Şema 4.13: Vaka analizi döngüsü

Bir problemi çözmeye başlamadan önce "Bu problemde ne tahmin etmeye çalışıyoruz?" sorusu sorulur. Bu sorunun cevabı, büyük ölçüde hangi yöntemin kullanılacağını belirler. Doğru yöntemi seçmek için problemin yapısını analiz edilmelidir (Tablo 4.10).

Tablo 4.10: Problem Durumlarına Göre Kullanılabilecek Yapay Zekâ Yöntemleri

Problem Sorusu	Hedef Değişken Türü	Uygun Yöntem
Üretim hattındaki bir parçanın boyutu kaç mm olacak?	Sürekli sayı	Regresyon
Bu hasta, o ilaca alerjik tepki gösterir mi?	İki kategori (evet/hayır)	Sınıflandırma
Hangi öğrenciler benzer öğrenme güçlükleri yaşıyor?	Kategori yok, etiket yok	Kümeleme
Kütüphanede hangi kitaplar birlikte ödünç alınıyor?	İlişki/örüntü keşfi	Birliktelik Kuralları

Verilen problem durumlarına göre vaka analiz detayları incelenmiştir.

1. Problem: Üretim Hattı- Parça Boyutu Tahmini

Vaka: Bir otomobil fabrikasında üretilen metal parçaların boyut hata oranı	
PROBLEM	Üretim bandındaki sıcaklık, baskı kuvveti ve ham madde yoğunluğu gibi değişkenlere göre üretilen parçanın boyutunun (mm cinsinden) öngörülmesi.
VERİ	Her üretim döngüsüne ait sensör kayıtları (sayısal, etiketli): sıcaklık, baskı, yoğunluk → ölçülen parça boyutu.
YÖNTEM	Lineer Regresyon – Hedef değişken (boyut) sürekli bir sayıdır.
ÇIKTI	Her yeni üretim döngüsü için tahmin edilen parça boyutu. Boyut kabul aralığı dışına çıkıyorsa üretim ayarı yapılır.

2. Problem: Sağlık- İlaç Alerjisi Tahmini

Vaka: Bir hastanenin ilaç reçeteleme sistemi	
PROBLEM	Hastanın yaşı, kilosu, kronik hastalıkları ve geçmiş ilaç tepkileri kullanılarak belirli bir ilaca alerjik tepki gösterip göstermeyeceğinin tahmin edilmesi.
VERİ	Hasta kayıtları (tıbbi geçmiş, demografik bilgiler, geçmiş reaksiyon etiketleri): etiketli, kategorik hedef.
YÖNTEM	Sınıflandırma (Karar Ağacı veya Random Forest) Hedef değişken iki adettir: alerjik tepki var / yok.
ÇIKTI	Her reçete öncesinde "Yüksek risk" veya "Düşük risk" uyarısı. Yüksek risk çıktısında doktor onayı zorunlu hâle getirilir.

3. Problem: Eğitim- Öğrenci Profili Gruplama

Vaka: Bir eğitim platformunun kişiselleştirilmiş içerik sistemi	
PROBLEM	Binlerce öğrencinin soru çözme süresi, hata örüntüsü ve konu bazlı başarı verilerine bakılarak benzer öğrenme güçlükleri yaşayan öğrencilerin gruplanması.
VERİ	Öğrenci etkinlik logları (etiket yok): çözülen soru sayısı, ortalama süre, hata türleri.
YÖNTEM	K-Means Kümeleme Veri etiketsizdir; benzerliğe göre doğal gruplar aranır.
ÇIKTI	Her öğrencinin ait olduğu öğrenme profili grubu. Platforma her gruba özgü ek alıştırmalar ve açıklamalar sunulur.

4. Problem: Kütüphane- Birlikte Ödünç Alınan Kitaplar

Vaka: Bir okul kütüphanesinin kitapları ödünç verme sistemi	
PROBLEM	Üyelerin geçmiş ödünç alma kayıtlarına bakılarak hangi kitapların sıklıkla birlikte alındığının keşfedilmesi ve yeni üyelere öneri yapılması.
VERİ	Ödünç alma işlem logları (her satır bir işlem, sütunlar ödünç alınan kitaplar): etiketsiz, ilişkisel.
YÖNTEM	Birliktelik Kuralları (Apriori) Amaç tahmin değil, birlikte görülen örüntüleri keşfetmektir.
ÇIKTI	"X kitabını ödünç alan üyelerin %65'i Y kitabını da ödünç almıştır" gibi kural setleri. Kütüphane bu kurallara göre öneri listesi oluşturur.

BİLGİ KUTUSU

Bir problem için birden fazla yöntemin uygun görünmesi mümkündür. Örneğin "Bu hastanın bir yıl içinde tekrar hastaneye yatma olasılığı nedir?" sorusuna hem regresyonla (olasılık değeri) hem de sınıflandırmayla (yüksek risk / düşük risk) yaklaşılabilir. Yöntem seçiminde eldeki veri miktarı, modelin yorumlanabilirlik ihtiyacı ve kararın nasıl kullanılacağı belirleyici olur.

4. SIRA SİZDE

Aşağıdaki vaka çalışmalarını inceleyiniz ve soruları cevaplayınız.

1. Vaka Çalışması: Akıllı Sera Yönetimi

Serada sıcaklık, nem, ışık miktarı, toprak nemi ve günlük ürün miktarı gibi veriler düzenli olarak kaydedilmektedir.

Sorular:

1. Günlük ürün miktarını geleceğe yönelik tahmin etmek için hangi yapay zekâ yaklaşımı kullanılabilir? Açıklayınız.
2. Bitkilerin durumu "sağlıklı", "uyarı" ve "riskli" olarak ayrılmak istenirse hangi yaklaşım uygundur?
3. Benzer çevresel koşullara sahip günleri gruplandırmak için hangi yöntem kullanılabilir?
4. Sulama ve gübreleme işlemlerinin birlikte gerçekleşme durumunu incelemek için hangi veri analizi yaklaşımı kullanılabilir?

2. Vaka Çalışması: Robotik Atölyede Bakım Planlama

Atölyedeki makinelerin çalışma süresi, sıcaklık, titreşim düzeyi, enerji tüketimi ve bakım geçmişi kayıt altına alınmaktadır.

Sorular:

1. Bir makinenin ne zaman bakım gerektireceğini tahmin etmek için hangi yöntem kullanılabilir?
2. Makineleri "düşük risk", "orta risk" ve "yüksek risk" olarak sınıflandırmak için hangi yaklaşım uygundur?
3. Benzer çalışma özelliklerine sahip makineleri gruplandırmak için hangi yöntem kullanılabilir?
4. Hangi makinelerin benzer arıza geçmişine sahip olduğunu belirlemek için hangi veri analizi yaklaşımı kullanılabilir?

2. Vaka Çalışması: Akıllı Kütüphane Sistemi

Öğrencilerin ödünç aldığı kitaplar, okuma süreleri, ziyaret sıklıkları ve tercih ettikleri kitap türleri kayıt altına alınmaktadır.

Sorular:

1. Bir öğrencinin ay sonunda kaç kitap okuyacağını tahmin etmek için hangi yöntem kullanılabilir?
2. Öğrencileri okuma alışkanlıklarına göre gruplandırmak için hangi yaklaşım uygundur?
3. Birlikte ödünç alınan kitap türlerini belirlemek için hangi veri analizi yöntemi kullanılabilir?
4. Kütüphane yönetimi bu analiz sonuçlarını hangi karar süreçlerinde kullanabilir?

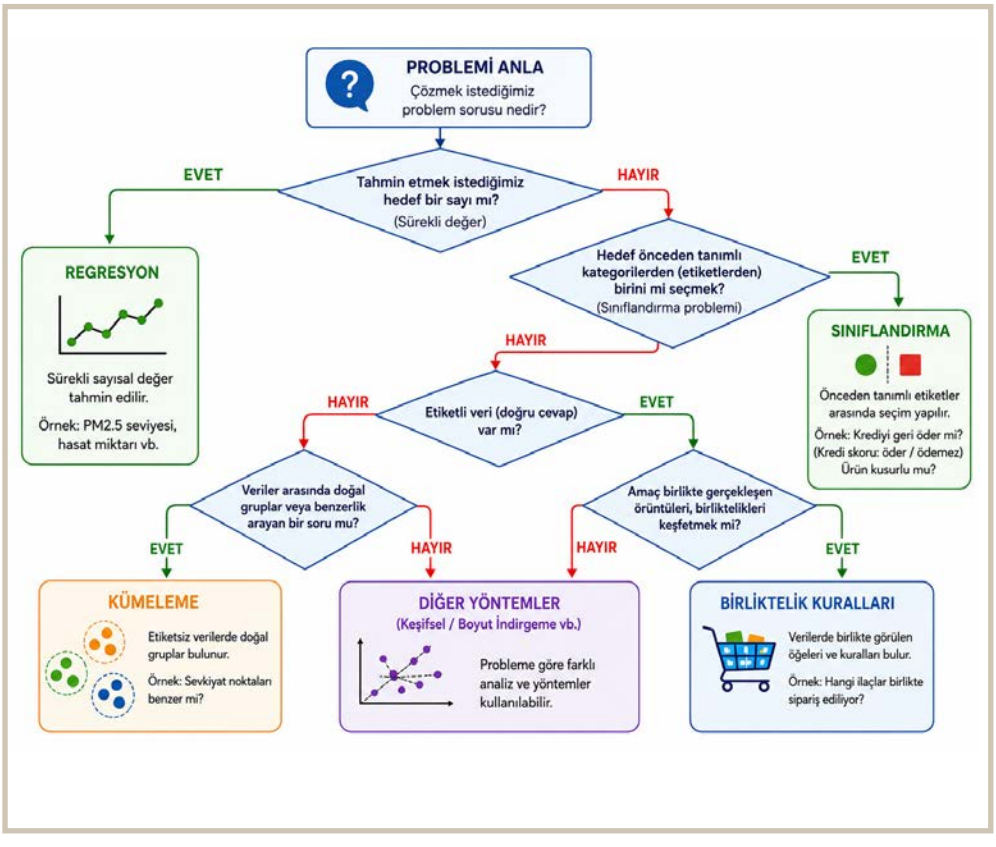
Günümüzde yapay zekâ yöntemleri farklı biçimlerde uygulanmaktadır. Bir uygulamayı değerlendirirken yalnızca "yapay zekâ kullanıyor" demek yeterli değildir. Hangi yöntemin kullanıldığını ve bu yöntemin neden o probleme uygun olduğunu anlamak daha değerlidir.

Tablo 4.11'de gerçek hayattan altı uygulama yer almaktadır. Her uygulama için problem, veri ve yöntem ilişkisini inceleyiniz.

Tablo 4.11: Gerçek Hayat Uygulamaları

Uygulama	Problem Sorusu	Kullanılan Yöntem	Gerekçe
Hava Kalitesi İzleme Sistemi	Yarın sabah PM2.5 seviyesi ne olacak?	Regresyon	Hedef sürekli bir ölçüm değeridir.
Banka Kredi Puanlama	Bu başvuru sahibi krediyi geri öder mi?	Sınıflandırma	İki kategori vardır: öder / ödemez.
Lojistik Merkez Optimizasyonu	Hangi sevkiyat noktaları benzer talep yapısı gösteriyor?	Kümeleme	Etiket yok; doğal gruplar aranıyor.
Çevrimiçi Eczane Sistemi	Hangi ilaçlar birlikte sipariş ediliyor?	Birliktelik Kuralları	Birliktelik örüntüleri keşfediliyor.
Tarımsal Ürün Verimi Tahmini	Bu tarlada bu sezon kaç ton domates hasat edilir?	Regresyon	Hedef sayısal ve süreklidir.

Tablodaki uygulamalar incelendiğinde her problem sorusunun yöntem seçimini doğrudan yönlendirdiği görülür. "Kaç?" sorusu regresyona, "Hangisi?" veya "Var mı?" sorusu sınıflandırmaya, "Hangi gruplar?" sorusu kümelemeye, "Ne ile birlikte?" sorusu birliktelik kurallarına işaret eder. Şema 4.14'te problem sorusu türüne göre yöntem seçimi akış diyagramı verilmiştir. Doğru yöntem seçimi, problemi iyi anlamak ve veriyi doğru değerlendirmekle başlar.



Şema 4.14: Problem sorusu türüne göre yöntem seçim akış diyagramı

5. SIRA SİZDE

Aşağıdaki problem durumlarında hangi yapay zekâ yönteminin kullanılabileceğini belirleyiniz.

Problem Durumu	Kullanılabilecek Yöntem
Bir hastanede acil servis yoğunluğunun saatlik olarak tahmin edilmesi.	
Bir e-ticaret sitesinde iade edilen ürünlerin, benzer özelliklere göre gruplandırılması.	
Bir güvenlik kamerasının, görüntüdeki kişinin çalışan mı yoksa ziyaretçi mi olduğuna karar vermesi.	
Bir müzik yayın platformunda hangi şarkıların birlikte çalma listelerine eklendiğinin analiz edilmesi.	

4.5.1. Başarı Hikâyeleri ve Kullanılan Yöntemler

Yapay zekâ yöntemleri pek çok alanda başarılı sonuçlar üretmiştir. Bu başarı hikâyelerine bakmanın bir amacı da gerçek problemlerin nasıl modellendiğini ve hangi yöntemlerin neden tercih edildiğini anlamaktır. Aşağıdaki başarı hikâyelerini problem-veri-yöntem-çıktı çerçevesiyle inceleyiniz.

1. Başarı Hikâyesi: Almanya, Sanayi-Rulman Arızası Tahmini

Bir Alman endüstriyel makine üreticisi, fabrikalarındaki rulmanların ne zaman arızalanacağını tahmin eden bir sistem geliştirir. Rulmanlar arızalanmadan önce titreşim frekansı, sıcaklık ve ses emisyonunda ölçülebilir değişiklikler gözlemlendiğini keşfeder. Bu değişiklikleri tespit etmek için sensörlerden toplanan zaman serisi verileri üzerinde bir sınıflandırma modeli eğitir.

Vaka: Rulman Arızası Tahmini	
PROBLEM	Rulmanın önümüzdeki 72 saat içinde arızalanıp arızalanmayacağının tahmin edilmesi.
VERİ	Her rulmanın saatlik titreşim, sıcaklık ve ses sensörü kayıtları; geçmiş arıza etiketleri.
YÖNTEM	Sınıflandırma (Random Forest) İki kategori: arıza riski var / yok.
ÇIKTI	Bakım ekibine otomatik uyarı. Plansız duruş süreleri %70 oranında azaldı.

2. Başarı Hikâyesi: Amerika Birleşik Devletleri, Perakende-Talep Tahmini

Büyük bir perakende zinciri, her ürün için hangi mağazada ne kadar stok bulundurması gerektiğini hesaplamak amacıyla bir yapay zekâ sistemi kurar. Geçmiş satış verileri, yerel hava durumu tahminleri, tatil takvimleri ve bölgesel demografik veriler modele girdi olarak verilir. Sistem, her ürün-mağaza kombinasyonu için bir sonraki haftanın satış rakamını tahmin eder.

Vaka: Haftalık Ürün Talep Tahmini	
PROBLEM	Belirli bir ürünün belirli bir mağazada gelecek hafta kaç adet satılacağının tahmin edilmesi.
VERİ	2 yıllık geçmiş satış kayıtları, hava durumu verileri, tatil takvimi, nüfus bilgileri.
YÖNTEM	Regresyon (Gradient Boosting) Hedef değişken: satış adedi (sürekli sayı).
ÇIKTI	Stok fazlalığı %18, ürün tükenmesi kaynaklı kayıplar %23 oranında azaldı.

3. Başarı Hikâyesi: Japonya, Şehir Planlaması-Deprem Hasar Bölgesi Analizi

Japon araştırmacılar, büyük bir deprem sonrasında hasar gören binaların hangi özelliklerde kümelendiğini analiz eder. Binaların yapım yılı, kat sayısı, zemin yapısı, kullanılan malzeme ve konum bilgilerini içeren etiketsiz veri seti üzerinde kümeleme algoritması çalıştırılır. Elde edilen gruplar, gelecekteki şehir planlaması kararlarına ve güçlendirme önceliklerine rehberlik eder.

Vaka: Hasar Gören Binaların Profil Kümelenmesi	
PROBLEM	Deprem sonrası hasar gören binaların yapısal benzerliklerine göre gruplandırılması; risk profili yüksek bina tiplerinin belirlenmesi.
VERİ	Bina envanter verisi: yapım yılı, kat sayısı, zemin sınıfı, malzeme türü, koordinatlar. Etiket yok.
YÖNTEM	Hiyerarşik Kümeleme Benzerliğe dayalı gruplama; kaç küme oluşacağı önceden bilinmiyor.
ÇIKTI	Beş farklı bina risk profili belirlendi. Yüksek riskli gruplara yönelik güçlendirme programı önceliklendirildi.

4. Başarı Hikâyesi: Türkiye, Savunma Teknolojisi-SİHA'da Görüntü Tabanlı Hedef Tanıma

Türk savunma sanayisi üreticisi, silahlı insansız hava aracı (SİHA) teknolojisinde yapay zekâ destekli görüntü işlemenin en dikkat çekici örneklerinden birini sunmaktadır. Hava aracı, görev sırasında kameralarıyla sürekli görüntü akışı toplar. Bir operatörün bu akışı kesintisiz izleyerek tüm nesneleri elle tanımlaması hem yavaş hem de hata payı yüksek bir süreçtir. Üretici bu sorunu yapay zekâ destekli otomatik nesne tanıma sistemiyle ele alır. Sistem, kamera görüntüsündeki her kareyi gerçek zamanlı olarak analiz eder ve görüntüdeki nesneleri önceden eğitilmiş sınıflara göre tanımlar. Bu yaklaşım, operatörün dikkatini tüm görüntü yerine yalnızca sistemin işaretlediği bölgelere yönlendirmesine olanak tanır.

Vaka: SİHA'da Görüntü Tabanlı Hedef Tanıma	
PROBLEM	Hava aracı kamerasından gelen canlı görüntü akışında nesnelerin (taşıt, yapı, vb.) gerçek zamanlı olarak tespit edilmesi ve sınıflandırılması.
VERİ	Farklı irtifa, ışık koşulu ve hava durumunda çekilmiş binlerce etiketli hava görüntüsü. Her görüntüde nesne sınırları ve kategori etiketleri manuel olarak işaretlenmiş (veri etiketleme / annotation).
YÖNTEM	Derin Öğrenme tabanlı Nesne Tespiti (CNN mimarisi; YOLO benzeri yaklaşım) Etiketli görüntü verisi üzerinde eğitilmiş konumu ve sınıfı aynı anda tahmin eden bir sınıflandırma modelidir.
ÇIKTI	Her video karesinde tespit edilen nesnelerin çevresine çizilen sınır kutusu ve sınıf etiketi. Operatör ekranında gerçek zamanlı gösterim; koordinat bilgisi sisteme iletilir.

BİLGİ KUTUSU

Başarı hikâyelerinde dikkat çeken önemli bir nokta modelin çıktısının doğrudan **karar** vermemesidir. Model bir tahmin veya sınıf üretir. Kararı her zaman insanlar verir. İyi tasarlanmış bir yapay zekâ sistemi, insanın karar kalitesini artırır, insanın yerine karar veremez.

4.1. ETKİNLİK

Adı	Güncel Yapay Zekâ Uygulaması Araştırması
Amaç	Güncel bir yapay zekâ uygulamasını inceleyerek uygulamanın çözdüğü problemi, kullandığı veri türlerini ve altyapısındaki makine öğrenmesi yöntemini gerekçesiyle açıklamak
Süre	1 ders saati
Kullanılan Araç ve Gereçler	bilgisayar, internet erişimi, kelime işlemci veya sunum hazırlama yazılımı, güvenilir bilgi kaynakları
Değerlendirme	Çalışmanız, karekodda verilen kontrol listesine göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız.

Hazırlık

1. Bilgisayarınızı ve kullanacağınız belge/sunum programını açınız.
2. Sağlık, tarım, eğitim, üretim, ulaşım veya güvenlik alanlarından ilginizi çeken birini çalışma alanınız olarak belirleyiniz.
3. Çalışma ortamınızın düzenli ve hazır olduğundan emin olunuz.
4. İş sağlığı ve güvenliği kurallarına uygun şekilde çalışma ortamınızı düzenleyiniz.
5. Çalışma dosyanızı oluşturunuz ve uygun isimle kaydediniz.

Etkinlik Sırasında Uyulması Gereken Kurallar

- Araştırma sürecinde güvenilir, resmî ve güncel teknoloji kaynaklarından yararlanınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Kaynaklardan elde ettiğiniz bilgilerin doğruluğunu sorgulayınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Uygulamanın kullandığı yöntemleri açıklarken teknik kavramları doğru ve anlaşılır bir dille ifade ediniz.

Çalışma Yönergesi

1. Belirlediğiniz sektörde (sağlık, tarım vb.) aktif olarak kullanılan güncel bir yapay zekâ uygulaması seçiniz ve çalışma planı oluşturunuz.
2. Seçtiğiniz uygulamanın gerçek hayatta hangi problemi çözdüğünü ve hedeflenen amaca nasıl ulaştığını açıklayınız.

3. Uygulamanın çalışırken girdi olarak ne tür veriler (görüntü, sensör verisi, metin, sayısal ölçümler vb.) kullandığını belirleyiniz.
4. Bu uygulamanın problem çözerken hangi makine öğrenmesi yöntemini (regresyon, sınıflandırma, kümeleme veya birliktelik kuralları) kullanmış olabileceğini mantıksal gerekçeleriyle detaylandırınız.
5. Uygulamanın çıktılarını, sektöre sağladığı faydaları ve kullanımında karşılaşılabilecek olası sınırlılıkları (yanılma payı, veri gizliliği, çevresel faktörler vb.) değerlendirerek çalışmanızı tamamlayınız.



Kontrol listesine ulaşmak için karekodu okutunuz.
Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

4.5.2. Örnek Vakalar Üzerinde Yapay Zekâ Uygulamaları

Gerçek bir vaka üzerinde uygulama geliştirirken aşağıda belirtilen adımlar sırasıyla gerçekleştirilmelidir.

1. Problemi başından sonuna kadar ele almak.
2. Problemi anlamak.
3. Veriyi incelemek.
4. Uygun yöntemi seçip gerekçelendirmek.
5. Python ile bir model oluşturmak.
6. Modelin çıktısını yorumlayarak karar vericiye ne söylenebileceğini belirlemek.

4.5. ÖRNEK UYGULAMA

Adı	Random Forest ile Su Kalitesi Sınıflandırma Uygulaması
Amaç	Etiketli veri seti üzerinde denetimli öğrenme (sınıflandırma) algoritmasını uygulayarak, modelin performansını ölçmek
Süre	1 ders saati
Kullanılan Araç Gereç ve Malzemeler	bilgisayar, Jupyter Notebook, Python programlama dili, pandas, scikit-learn, matplotlib, su_kalitesi.csv dosyası

Hazırlık

- Bilgisayarınızı ve Jupyter Notebook uygulamasını açınız.
- Çalışma ortamınızın hazır olduğunu kontrol ediniz.
- ş sağlığı ve güvenliği kurallarına uyarak ekran başında doğru oturma pozisyonu alınız.
- Yeni bir Notebook oluşturunuz ve adını **"Su_Kalitesi_Siniflandirma"** olarak değiştiriniz.
- Gerekli kütüphanelerin ve "su_kalitesi.csv" veri setinin aynı dizinde kullanılabilir olduğunu kontrol ediniz.

Uygulama Sırasında Uyulması Gereken Kurallar

- Uygulama sürecinde arkadaşlarınızla iş birliği ve uyum içerisinde çalışınız.
- Dijital araç gereci dikkatli ve sorumluluk bilinciyle kullanınız.
- Çalışmalarınızı planlanan süre içerisinde tamamlamaya özen gösteriniz.
- Çalışma sırasında düzenli ve dikkatli davranınız.

Uygulama Adımları

1. Adım: Gerekli Kütüphanelerin Eklenmesi

Uygulama için gerekli Python kütüphanelerini programa ekleyiniz.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
import matplotlib.pyplot as plt
```

Pandas veri setini okumak ve düzenlemek için, scikit-learn model oluşturmak ve test etmek için, matplotlib ise sonuçları grafik üzerinde göstermek için kullanılır.

2. Adım: Veri Setinin Yüklenmesi ve İncelenmesi

Su kalitesi veri setini Python ortamına yükleyiniz. head() komutu ile verinin ilk 5 satırına bakınız. value_counts() ile hedef değişkenimiz olan "Potability" (0: İçilemez, 1: İçilebilir) sınıfında kaçar adet numune olduğunu kontrol ediniz.

```
veri = pd.read_csv("su_kalitesi.csv")
print(veri.head())
print(veri["Potability"].value_counts())
```

3. Adım: Özelliklerin ve Hedef Değişkenin Ayrılması

Öncelikle boş (eksik) verileri o sütunun ortalaması ile doldurunuz . Daha sonra kimyasal ölçümlerin bulunduğu özellikleri X değişkenine, tahmin etmek istediğiniz suyun içilebilirlik durumunu (etiket) y değişkenine aktarınız.

```
veri = veri.fillna(veri.mean())
X = veri.drop("Potability", axis=1)
y = veri["Potability"]
```

4. Adım: Verinin Eğitim ve Test Olarak Bölünmesi

Modelin eğitilmesi ve sonrasında daha önce hiç görmediği verilerle test edilmesi için veri setini ikiye bölünüz. Verilerin %80'ini eğitim, %20'sini test için ayrınız (test_size=0.2).

```
X_egitim, X_test, y_egitim, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

5. Adım: Modelin Oluşturulması ve Eğitilmesi

Makine öğrenmesi modeli olan Random Forest sınıflandırıcısını tanımlayınız.

```
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_egitim, y_egitim)
```

model.fit() komutu ile model, eğitim verileri (kimyasal ölçümler ve etiketler) arasındaki ilişkiyi öğrenir.

6. Adım: Modelin Test Edilmesi ve Değerlendirilmesi

Eğitilen model, etiketlerini gizlediğimiz test verileri üzerinde çalıştırınız (model.predict). Modelin tahminleri ile gerçek etiketleri karşılaştırarak doğruluk oranı ve karmaşıklık matrisini ekrana yazdırınız.

```
tahminler = model.predict(X_test)
print("Doğruluk Oranı:", accuracy_score(y_test, tahminler))
print("Karmaşıklık Matrisi:\n", confusion_matrix(y_test, tahminler))
```

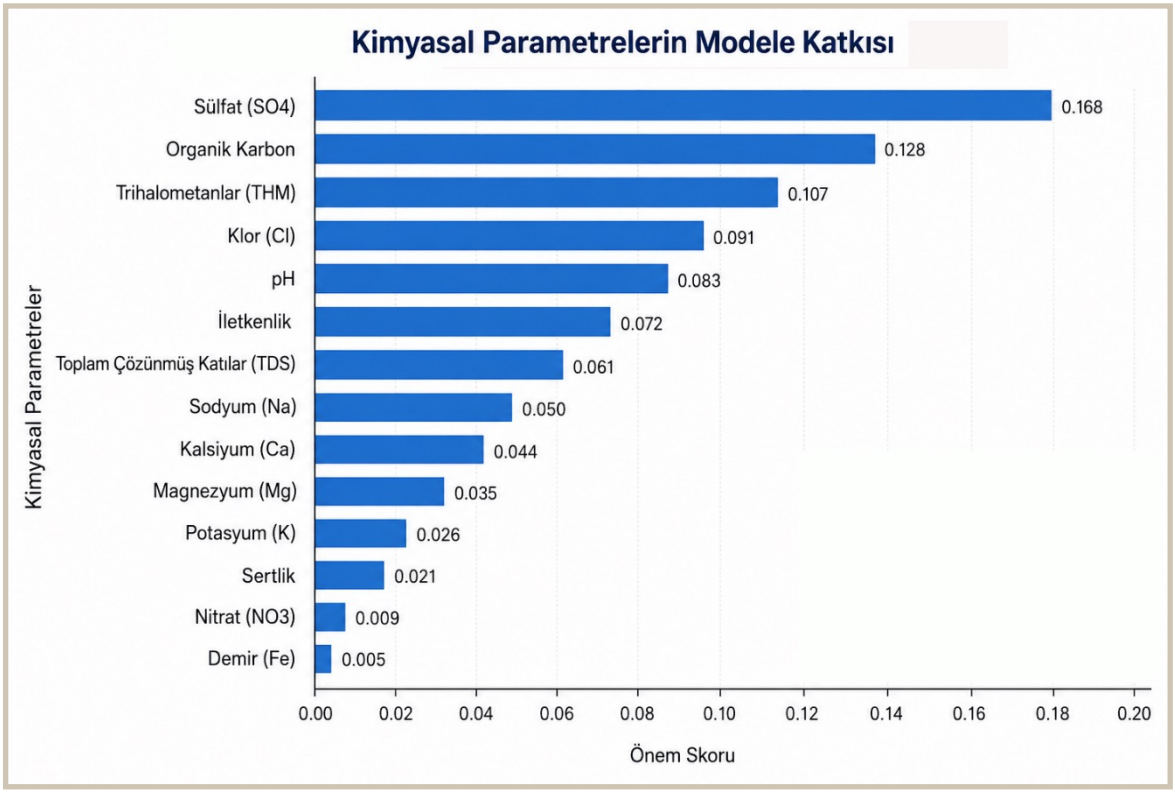
7. Adım: Özellik Önemlerinin Görselleştirilmesi

Modelin tahmin yaparken hangi kimyasal özelliklere daha çok dikkat ettiğini (özellik önemi) hesaplayınız ve bir çubuk grafik üzerinde görselleştiriniz.

```
onem = pd.Series(model.feature_importances_, index=X.columns).sort_values(ascending=False)
onem.plot(kind="barh", figsize=(8, 5))
plt.title("Özellik Önemleri")
plt.xlabel("Önem Skoru")
plt.tight_layout()
plt.show()
```

8. Adım: Sonuçların Yorumlanması

Çıktıda elde edilen doğruluk oranını ve karmaşıklık matrisini inceleyiniz. Matristeki doğru bilinen ve yanlış bilinen (Ör. İçilebilir suyu içilemez olarak tahmin etme) sayılarına dikkat ediniz. Çizdirilen çubuk grafiği inceleyerek şu soruların yanıtlarını düşününüz:



Şema 4.15: Özellik önem çubuğu grafiği - kimyasal parametrelerin modele katkısı

Sonuç

1. **Bu uygulamada model eğitilirken hedef değişken (etiket) neden kullanılmıştır? Açıklayınız.**

   .....
   .....

2. **Veri setini baştan "Eğitim" ve "Test" olarak ikiye bölmemizin temel amacı nedir? Açıklayınız.**

   .....
   .....

3. **Karmaşıklık matrisi (Confusion Matrix) bize doğruluk oranından (Accuracy) farklı olarak hangi detaylı bilgileri sunar? Açıklayınız.**

   .....
   .....

4. **Özellik önemleri grafiğine göre, bir suyun içilebilir olup olmadığına karar vermede en etkili olan ölçümler hangileridir? Bu durum tesis uzmanlarına nasıl bir fayda sağlar? Açıklayınız.**

   .....
   .....

5. **Şema 4.14'ü inceleyiniz. Veri setinden "Nitrat" ve "Demir" ölçümleri tamamen çıkarılsa modelin doğruluk oranında çok büyük bir düşüş yaşanır mı? Nedenini açıklayınız.**

   .....
   .....

Tasarım notu örnek csv dosyası içeriği şu şekilde olabilir:

Sulfat,Organik_Karbon,Trihalometanlar,Klor,pH,Iletkenlik,TDS,Sodyum,Kalsiyum,Magnezyum,Potasyum,Sertlik,Nitrat,Demir,Potability

320.5,12.4,55.2,3.1,7.2,450.0,320.0,45.5,60.2,15.3,4.2,150.0,12.5,0.1,1

380.2,18.1,85.4,5.2,6.1,780.0,650.0,120.0,90.5,35.2,12.4,250.0,45.0,0.8,0

290.1,8.5,42.1,2.8,7.5,380.0,280.0,30.2,50.1,12.0,3.1,130.0,5.2,0.05,1

410.8,22.3,95.6,6.5,5.8,850.0,800.0,150.5,110.0,45.8,18.5,310.0,55.2,1.2,0

310.4,10.2,48.5,2.9,7.4,410.0,290.0,35.0,55.0,14.5,3.8,140.0,8.4,0.08,1

395.5,19.5,88.2,5.8,6.4,810.0,720.0,135.0,95.2,38.4,14.2,280.0,48.5,0.9,0

335.2,14.1,60.8,3.5,7.1,490.0,350.0,55.2,65.8,18.2,5.5,165.0,15.0,0.2,1

405.1,21.0,92.5,6.1,5.9,830.0,760.0,142.8,105.4,42.1,16.8,295.0,52.1,1.1,0

285.6,7.8,38.4,2.5,7.6,360.0,260.0,28.5,48.2,11.5,2.9,120.0,4.5,0.04,1

375.9,17.4,82.1,4.9,6.5,750.0,610.0,115.4,85.6,32.8,11.5,235.0,42.8,0.7,0

345.8,15.2,65.3,3.8,7.0,520.0,380.0,65.1,72.4,21.5,6.8,180.0,18.5,0.3,1

295.4,9.1,45.6,2.7,7.3,395.0,275.0,32.4,52.8,13.2,3.5,135.0,6.8,0.06,1)

6. SIRA SİZDE

Su kalitesi vakasından yola çıkarak aşağıdaki soruları cevaplayınız.

1. Tesisin veri setinde sınıf dengesizliği (örneğin 1000 içilemez, 200 içilebilir numune) olsaydı, bu durum modelin doğruluk oranını nasıl etkilerdi?
2. Bu vakada Random Forest yerine karar ağacı kullansaydık ne gibi avantajları veya dezavantajları olurdu?
3. Tesisin kimyasal ölçüm sayısını azaltması gerekse, özellik önem grafiğine bakarak hangi özelliklerin kaldırılabileceğini önerirsiniz? Nedenini açıklayınız.
4. Bu modelin çıktısına dayanarak tesiste hangi kararı kim vermelidir? Yapay zekânın sorumluluk sınırı ne olmalıdır?

5. UYGULAMA

Adı	Akıllı Sera Sensör Verileriyle Makine Öğrenmesi
Amaç	Sensör verilerini kullanarak regresyon modeliyle sayısal tahmin yaparak sınıflandırma modeliyle durum tespiti gerçekleştirmek
Süre	2 ders saati
Yönerge	Aşağıdaki işlem basamaklarını takip ederek sera verileri üzerinde makine öğrenmesi uygulamalarını gerçekleştiriniz.
Değerlendirme	Çalışmanız karekodda verilen derecelendirme ölçeğindeki ölçütlere göre öğretmeniniz tarafından değerlendirilecektir. Çalışmanızı planlarken bu ölçütleri dikkate alınız. Çalışma sonunda öz değerlendirme formunu doldurarak kendi öğrenme sürecinizi değerlendiriniz.

Ön Bilgi

Bir tarım işletmesi, büyük cam seralarında domates ve biber üretmektedir. Sera içindeki sensörler her gün düzenli olarak hava sıcaklığı, nem oranı, ışık yoğunluğu, toprak nemi ve o günün hasat edilen ürün miktarını kaydetmektedir. Bu verilerin yapay zekâ yöntemleriyle incelenmesi, seranın daha akıllı ve verimli yönetilmesini sağlar.

İşlem Basamakları

1. Aşağıdaki örnek sera sensör verilerini inceleyiniz ve verilerin içerdiği bilgileri açıklayınız.

sicaklik	nem	isik_yogunlugu	toprak_nemi	hasat_miktari_kg	bitki_durumu
24.5	65.2	850	45.1	47.3	saglikli
28.1	55.4	920	38.5	52.1	uyari
32.4	45.1	1100	25.4	31.5	riskli
22.8	70.5	780	48.2	45.0	saglikli

2. Gerekli Python kütüphanelerini çalışma ortamınıza ekleyiniz.
3. "sera_verileri.csv" dosyasını Pandas kullanarak DataFrame hâline getiriniz ve görüntüleyiniz.
4. Günlük hasat miktarını tahmin etmek (Regresyon) için özellik (X) ve hedef (y) değişkenleri ayırınız.
5. Veriyi eğitim ve test olarak bölüp Lineer Regresyon modelini eğitiniz.
6. Hasat tahmini sonuçlarını (R-kare ve MAE) ekrana yazdırarak inceleyiniz.
7. Bitki durumunu tahmin etmek (Sınıflandırma) için hedef değişkeni değiştirerek veriyi tekrar hazırlayınız.
8. Random Forest sınıflandırıcısını eğitip test ediniz ve doğruluk oranını yazdırınız.
9. Elde ettiğiniz sonuçları inceleyerek aşağıdaki soruları cevaplayınız.

A) Hasat miktarını tahmin ederken bulduğunuz R-kare değeri 1'e ne kadar yakındır? Bu durum modelin başarısı hakkında ne ifade eder?

B) Ortalama Mutlak Hata (MAE) değeri dikkate alındığında, modelin günlük kilogram tahmini işletme için yeterince güvenilir midir? Bu tahmin işletmenin hangi planlamalarında kullanılabilir?

C) Sınıflandırma raporuna göre, sistemin bitkileri durumlarına (sağlıklı, uyarı, riskli) ayırma başarısı nasıldır?

Ç) "Riskli" sınıfındaki yanlış negatif tahminlere (gerçekte riskli olan bitkilerin gözden kaçması) neden özellikle dikkat edilmelidir?

D) Sürekli bir sayıyı tahmin etmek ile önceden belirlenmiş kategorilere ayırmak arasındaki mantıksal fark bu uygulamada nasıl görülmektedir?

Derecelendirme ölçeğine ve öz değerlendirme formuna ulaşmak için karekodu okutunuz. Dijital yayınlarda karekodun üstüne tıklayarak içeriğe ulaşabilirsiniz.

NOTLAR

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başında boş bırakılan parantezlere, cümlelerde verilen bilgiler doğru ise "D", yanlış ise "Y" yazınız.

1. ( ) Regresyon yöntemleri, sıcaklık, maliyet, süre ve ürün miktarı gibi sayısal değerlerin tahmin edil mesinde kullanılır.
2. ( ) Sınıflandırma yöntemlerinde verilerin ait olacağı sınıflar model çalıştırıldıktan sonra kendiliğinden oluşur.
3. ( ) K-Means algoritmasında K değeri, oluşturulacak küme sayısını belirtir.
4. ( ) Güven değeri yüksek olan bir birliktelik kuralı, destek ve kaldıraç değerleri incelenmeden doğrudan kullanılmalıdır.
5. ( ) Vaka analizinde kullanılacak yapay zekâ yöntemi, problem ve ihtiyaç duyulan veriler belirlenmeden önce seçilmelidir.

B) Aşağıdaki soruların doğru cevabını işaretleyiniz.

1. Bir serada sıcaklık, nem, ışık miktarı ve toprak nemi verilerinden yararlanılarak ertesi gün elde edilecek ürün miktarı kilogram cinsinden tahmin edilecektir. **Bu problemde aşağıdaki yöntemlerden hangisi kullanılmalıdır?**

A) Birliktelik kuralları
B) Kümeleme
C) Regresyon
D) Sınıflandırma
E) Veri sıralama

2. Bir veri kümesindeki giriş ve sonuç değişkenleri arasındaki ilişki, grafik üzerinde doğru yerine eğrisel bir yapı göstermektedir. **Bu durumda aşağıdaki yöntemlerden hangisinin kullanılması daha uygundur?**

A) Polinom regresyon
B) Lineer regresyon
C) İki sınıflı sınıflandırma
D) K-Means
E) Apriori

3. Bir çağrı merkezine gelen talepler; teknik destek, faturalandırma, satış ve şikâyet kategorilerinden birine yönlendirilecektir. **Bu problem aşağıdaki sınıflandırma türlerinden hangisine örnektir?**

A) Regresyon
B) İki sınıflı sınıflandırma
C) Kümeleme
D) Çok sınıflı sınıflandırma
E) Birliktelik analizi

4. **Kümeleme sonucunda elde edilen gruplarla ilgili aşağıdaki ifadelerden hangisi doğrudur?**

A) Her kümenin adı işlemden önce kesin olarak belirlenir.
B) Aynı kümedeki verilerin benzer özellikler göstermesi beklenir.ştır.
C) Kümeleme yalnızca iki grup oluşturabilir.
D) Kümeleme için bütün verilerin sınıf etiketleri bilinmelidir.
E) Farklı kümelerdeki verilerin tamamen aynı olması beklenir.

5. Bir Elbow grafiğinde eğrinin belirgin biçimde yön değiştirdiği nokta **K = 4** olarak görülmektedir. **Bu sonuç aşağıdakilerden hangisini gösterir?**

A) Veri kümesinin dört özelliği bulunduğunu
B) Dört veri kaydının hatalı olduğunu
C) Dört kümenin uygun bir seçenek olabileceğini
D) Modelin dört kez eğitilmesi gerektiğini
E) Her kümede dört veri bulunması gerektiğini

6. **Bir öğenin veya öğe grubunun bütün işlem kayıtları içerisinde ne sıklıkla bulunduğunu gösteren ölçüt aşağıdakilerden hangisidir?**

A) Doğruluk
B) Güven
C) Kaldıraç
D) Destek
E) Tahmin hatası

7. Bir yapay zekâ vaka çalışmasında izlenmesi gereken süreç aşağıdakilerin hangisinde doğru sıralanmıştır?

A) Yöntem seçimi – çıktı üretme – problem belirleme – veri toplama

B) Veri toplama – çıktı üretme – problem belirleme – yöntem seçimi

C) Problem belirleme – veri inceleme – uygun yöntemi seçme – çıktıyı yorumlama

D) Model oluşturma – problem belirleme – veriyi silme – karar verme

E) Çıktıyı yorumlama – problem belirleme – yöntem seçme – veri toplama

C) Aşağıdaki metni okuyunuz.

Bir belediye, atık toplama hizmetlerini daha verimli hâle getirmek amacıyla yapay zekâ destekli bir sistem geliştirmektedir. Çöp kutularına yerleştirilen sensörlerle doluluk oranı, atık miktarı, konum, sıcaklık ve son boşaltılma zamanı kaydedilmektedir. Belediye, bu verilerden yararlanarak ertesi gün toplanacak atık miktarını tahmin etmek istemektedir. Ayrıca çöp kutularının normal, yakında boşaltılmalı ve acil boşaltılmalı biçiminde belirli gruplara ayrılması planlanmaktadır.

Sistem kapsamında benzer atık üretim özelliklerine sahip mahallelerin belirlenmesi ve geri dönüşüm kayıtlarında hangi atık türlerinin sıklıkla birlikte bulunduğunun incelenmesi de amaçlanmaktadır. Modelin ürettiği sonuçlar, belediyenin ilgili birimlerine sunulacak; toplama araçlarının güzergâhlarına ve çalışma planlarına ilişkin son kararlar görevli personel tarafından verilecektir.

Metne göre aşağıdaki soruları cevaplayınız.

1. Belediye tarafından gerçekleştirilmek istenen aşağıdaki işlemlerde hangi yapay zekâ yöntemlerinin kullanılabileceğini yazınız.

a) Ertesi gün toplanacak atık miktarını tahmin etme

.....

b) Çöp kutularını doluluk durumlarına göre belirli kategorilere ayırma

.....

c) Benzer atık üretim özelliklerine sahip mahalleleri gruplandırma

.....

ç) Aynı kayıtlarda birlikte bulunan atık türlerini belirleme

.....

2. Ertesi gün toplanacak atık miktarının tahmin edilmesinde kullanılabilecek iki giriş değişkeni ile sonuç değişkenini yazınız.

.....

.....

KAYNAKÇA

- Alpaydın, E. (2020). Yapay öğrenme. İstanbul: Boğaziçi Üniversitesi Yayınevi.
- Alpaydın, E. (2023). Introduction to machine learning (5th ed.). MIT Press.
- Géron, A. (2022). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O›Reilly Media.
- McKinney, W. (2022). Python for data analysis (3rd ed.). O'Reilly Media.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
- Raschka, S., Liu, Y., Mirjalili, V., & Dzhulgakov, D. (2022). Machine learning with PyTorch and Scikit-Learn. Packt Publishing.
- Şeker, Ş. E. (2021). Veri madenciliği. Ankara: Seçkin Yayıncılık.
- Cumhurbaşkanlığı Dijital Dönüşüm Ofisi Başkanlığı & Sanayi ve Teknoloji Bakanlığı. (2021). Ulusal Yapay Zekâ Stratejisi 2021–2025. 10 Mart 2026, 12:36 tarihinde https://bilgem.tubitak.gov.tr/wp-content/uploads/sites/8/TR-UlusalYZStratejisi2021-2025.pdf adresinden erişildi.
- Demirkol, Z. (2021). Herkes için yapay zekâ. İstanbul: Genç Destek Yayınları.
- Köse, U. (Ed.). (2021). Yapay zekâ etiği. Ankara: Nobel Akademik Yayıncılık.
- Millî Eğitim Bakanlığı Yenilik ve Eğitim Teknolojileri Genel Müdürlüğü. (2025). Eğitimde yapay zekâ politika belgesi ve eylem planı (2025–2029). 10 Mart 2026, 15:35 tarihinde https://yegitek.meb.gov.tr/meb_iys_dosyalar/2025_06/17092340_egitimdeyapayzekapolitikabelgesiveeylemplani202520291.pdf adresinden erişildi.
- Shah, P. (2024). Yapay zekâ ve eğitimin geleceği: Yapay zekâ çağında öğretim (A. L. Orcan, Çev.). İstanbul: The Kitap Yayınları.
- Bengio, Y. (2009). Learning deep architectures for AI. Foundations and Trends in Machine Learning 2(1), 1-127.
- Bengio, Y., P. Lamblin, D. Popovici and H. Larochelle (2007). "Greedy layer-wise training of deep networks." In NIPS'2006. 14, 19, 200, 323, 324, 530, 532.
- Deng, L., & Yu, D. (2014). "Deep learning: methods and applications". Foundations and Trends in Signal Processing, 7(3–4), 197-387.
- LeCun, Y. (1987). Modèles connexionistes de l'apprentissage, Université de Paris VI. 18, 504, 517.
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). "Deep learning". Nature 521(7553), 436.
- McCulloch, W. S. and W. Pitts (1943). "A logical calculus of the ideas immanent in nervous activity." The Bulletin of Mathematical Biophysics 5(4): 115–133.
- Minsky, M. L. a. P., S. A. (1969). "Perceptrons." MIT Press, Cambridge. 15.
- Rosenblatt, F. (1958). "The perceptron: A probabilistic model for information storage and organization in the brain." Psychological Review 65(6): 386–408.
- Rosenblatt, F. (1962). Principles of Neurodynamics. Spartan, New York. 15, 27.
- Rumelhart, D. E., G. E. Hinton and R. J. Williams (1986). "Learning representations by back-propagating errors." Nature 323(6088): 533–536.
- Rumelhart, D. E., J. L. McClelland and T. P. R. Group (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT Press, Cambridge. 17.
- Widrow, B. and M. E. Hoff (1960). "Adaptive switching circuits." Adaptive Switching Circuits. In 1960.

- Han, J., Kamber, M., & Pei, J. (2022). Data mining: Concepts and techniques (4th ed.). Morgan Kaufmann.
- Mitchell, T. M. (1997). Machine learning. McGraw-Hill.
- Russell, S., & Norvig, P. (2021). Artificial intelligence: A modern approach (4th ed.). Pearson.
- Şeker, Ş. E. (2021). Veri madenciliği. Ankara: Seçkin Yayıncılık.
- Tan, P. N., Steinbach, M., Karpatne, A., & Kumar, V. (2019). Introduction to data mining (2nd ed.). Pearson.
- Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2017). Data mining: Practical machine learning tools and techniques (4th ed.). Morgan Kaufmann.

- Milli Eğitim Bakanlığı (2026) BasımTeknolojileri Alanı. Ankara
- Türk Dil Kurumu. (2012). Yazım kılavuzu. Ankara: Türk Dil Kurumu Yayınları.
- Türk Dil Kurumu. (2023). Bilişim terimleri sözlüğü. Ankara: Türk Dil Kurumu Yayınları. Erişim tarihi: 09 Haziran 2026, http://sozluk.gov.tr
- T.C. Millî Eğitim Bakanlığı. (2025). Türkiye Yüzyılı Maarif Modeli Öğretim Programları Ortak Metni
- Bu ders materyalinde kaynakça, APA 6 kaynak gösterme yöntemine göre düzenlenmiştir.

GENEL AĞ KAYNAKÇASI

T.C. Millî Eğitim Bakanlığı. *TYMM Ortak Metin.* 2 Mart 2026 tarihinde https://tymm.meb.gov.tr/ortak-metin adresinden erişilmiştir.
Türk Dil Kurumu. *Güncel Türkçe sözlük.* 3 Mart 2026 tarihinde https://sozluk.gov.tr/ adresinden erişilmiştir.
Türk Dil Kurumu. *Yazım kılavuzu.* 3 Mart 2026 tarihinde https://tdk.gov.tr/tdk/kurumsal/yazim-kilavuzu/ adresinden erişilmiştir.



CEVAP ANAHTARLARI

Cevap anahtarlarına bu karekodu okutarak ulaşabilirsiniz.

Dijital dokümanlarda karekodun üzerine tıklayarak içeriğe ulaşabilirsiniz.



GENEL AĞ, GÖRSEL, E-İÇERİK KAYNAKÇASI

Kitabın görsel kaynakçasına, genel ağ kaynakçasına ve e-içerik kaynakçasına bu karekodu okutarak ulaşabilirsiniz.

Dijital dokümanlarda karekodun üzerine tıklayarak içeriğe ulaşabilirsiniz.




TÜRKİYE'NİN MÜLKİ İDARE BÖLÜMLERİ İLE
KARA VE DENİZ KOMŞULARI HARİTASI
BAŞKENT
Ankara
İL MERKEZİ
Elazığ
GÖL
ÜLKE SINIRI
ÖZERK BÖLGE SINIRI
AKARSU
0 50 100 200 300 400
Km
KARADENİZ
AKDENİZ
EGE DENİZİ
MARMARA DENİZİ
HAZAR DENİZİ
ROMANYA
UKRAYNA
RUSYA FEDERASYONU
BULGARİSTAN
GÜRCİSTAN
YUNANİSTAN
ERMENİSTAN
AZERBAYCAN
İRAN
SURİYE
IRAK
LİBYA
MISIR
LÜBNAN
İSRAİL
KUZEY KIBRIS TÜRK CUMHURİYETİ
GÜNEY KIBRIS RUM YÖNETİMİ
FİLİSTİN
Batı Şeria
Doğu Kudüs
Gazze Şeridi
NAHÇIVAN O.C. (AZERBAYCAN)
AĞRI O.C.
Sofya
Bükreş
Atina
Tiflis
Erivan
Bağdat
Şam
Beyrut
Lefkoşa
Tel Aviv
Sinop
Samsun
Trabzon
Rize
Artvin
Ardahan
Kars
Ordu
Giresun
Gümüşhane
Bayburt
Erzincan
Erzurum
Ağrı
Iğdır
Tunceli
Bingöl
Muş
Bitlis
Van
Elazığ
Malatya
Diyarbakır
Siirt
Batman
Şırnak
Hakkâri
Mardin
Şanlıurfa
Adıyaman
Kahramanmaraş
Gaziantep
Osmaniye
Adana
Hatay
Kilis
Mersin
Niğde
Kayseri
Sivas
Yozgat
Kırşehir
Nevşehir
Aksaray
Konya
Karaman
Antalya
Isparta
Burdur
Muğla
Aydın
Denizli
Manisa
İzmir
Uşak
Afyonkarahisar
Kütahya
Eskişehir
Balıkesir
Çanakkale
Gökçeada
Bursa
Yalova
İstanbul
Tekirdağ
Edirne
Kırklareli
Bilecik
Sakarya
Kocaeli
Düzce
Bolu
Zonguldak
Bartın
Karabük
Çankırı
Kastamonu
Amasya
Çorum
Tokat
Ankara
Kırıkkale
Tuz Gölü
Van Gölü
Urumiye Gölü
Kura N.
Aras N.
Çoruh N.
Yeşilırmak N.
Kızılırmak N.
Fırat N.
Dicle N.
Seyhan N.
Ceyhan N.
Göksu N.
Asi N.
Meriç N.
Sakarya N.
Büyük Menderes N.
Küçük Menderes N.
Gediz N.
Susuz K.
Nil N.


TÜRK DÜNYASI HARİTASI
Bağımsız Türk Devletleri
Türk Cumhuriyetleri, Türk Bölgeleri
Türk Toplulukları
0 250 500 1,000 1,500 2,000 Km
TÜRK CUMHURİYETLERİ, TÜRK BÖLGELERİ
1 GAGAVUZ YERİ
2 KARAÇAY-ÇERKESYA
3 KABARDİN-BALKARYA C.
4 DAĞISTAN C.
5 ÇUVAŞİSTAN C.
6 TATARİSTAN C.
7 BAŞKURT C.
8 KARAKALPAKİSTAN C.
9 ALTAY C.
10 HAKASYA C.
TÜRK TOPLULUKLARI
a Avrupa'daki Türkler
b Kosova Türkleri
c Yunanistan Türkleri
ç Bulgaristan Türkleri
d Romanya Türkleri
e Suriye Türkmenleri
f Irak Türkmenleri
g İran Türkleri
h Afganistan Türkmenleri
ı Pakistan Türkmenleri
i Hindistan Türkmenleri
j Çin H. C. Türkleri
k Moğolistan Türkleri
l Rusya Federasyonu Türkleri
ÜLKE KISALTMALARI
ARNAVUTLUK ARN
BELÇİKA BEL
BOSNA - HERSEK BH
GÜNEY KIBRIS RUM YÖNETİMİ GKRY
HIRVATİSTAN HRV
İSVİÇRE İS
KARADAĞ KD
KOSOVA KS
KUZEY KIBRIS TÜRK CUMHURİYETİ KKTC
KUZEY MAKEDONYA KMKD
LİHTENŞTAYN LİH
SIRBİSTAN SR
RUSYA FEDERASYONU
KAZAKİSTAN
TÜRKİYE
ÖZBEKİSTAN
TÜRKMENİSTAN
KIRGIZİSTAN
TACİKİSTAN
AZERBAYCAN
MOĞOLİSTAN
ÇİN HALK CUMHURİYETİ
HİNDİSTAN
PAKİSTAN
AFGANİSTAN
İRAN
SAHA (YAKUT) C.
TUVA C.
SİNCAN UYGUR ÖZERK BÖLGESİ
Barents Denizi
Kara Denizi
Norveç Denizi
Kuzey Denizi
Baltık Denizi
Karadeniz
Akdeniz
Hazar Denizi
Kızıldeniz
Umman Denizi
Yemen Denizi
Ohotsk Denizi
Bering Denizi
Tiren Denizi
Adriyatik Denizi
Manş Denizi
Cebelitarık B.