T.C.
KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ

YAPAY ZEKA YAKLAŞIMI İLE SON YILLARDA VE GELECEĞE YÖNELİK MESLEKİ DEĞİŞİMLER VE EĞİLİMLERİN ANALİZİ

EBRU KARAAHMETOĞLU

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

DOKTORA TEZİ

DANIŞMAN

Prof. Dr. Süleyman ERSÖZ

KIRIKKALE-2023

T.C.
KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ

YAPAY ZEKA YAKLAŞIMI İLE SON YILLARDA VE GELECEĞE YÖNELİK MESLEKİ DEĞİŞİMLER VE EĞİLİMLERİN ANALİZİ

EBRU KARAAHMETOĞLU

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

DOKTORA TEZİ

DANIŞMAN

Prof. Dr. Süleyman ERSÖZ

KIRIKKALE-2023

Ebru KARAAHMETOĞLU tarafından hazırlanan “YAPAY ZEKA YAKLAŞIMI İLE SON YILLARDA VE GELECEĞE YÖNELİK MESLEKİ DEĞİŞİMLER VE EĞİLİMLERİN ANALİZİ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalında DOKTORA TEZİ olarak kabul edilmiştir

Danışman: Prof. Dr. Süleyman ERSÖZ
Endüstri Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum/onaylamıyorum.

İkinci Danışman: Doç. Dr. Adnan AKTEPE
Endüstri Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum/onaylamıyorum.

Başkan: Prof. Dr. Cevriye GENCER
Endüstri Mühendisliği Anabilim Dalı, Gazi Üniversitesi
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum/onaylamıyorum.

Üye: Prof. Dr. Ahmet Kürşad TÜRKER
Endüstri Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum/onaylamıyorum.

Üye: Doç. Dr. Atilla ERGÜZEN
Bilgisayar Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum/onaylamıyorum.

Üye: Dr. Öğr. Üyesi Hakan ÖZKÖSE
Yönetim Bilişim Sistemleri Anabilim Dalı, Bartın Üniversitesi
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum/onaylamıyorum.

Tez Savunma Tarihi 10/01/2023

Jüri tarafından kabul edilen bu tezin Doktora Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

Prof. Dr. Recep ÇALIN
Fen Bilimleri Enstitüsü Müdürü

ETİK BEYANI

Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Ebru KARAAHMETOĞLU

12.01.2023

ÖZET

YAPAY ZEKA YAKLAŞIMI İLE SON YILLARDA VE GELECEĞE YÖNELİK MESLEKİ DEĞİŞİMLER VE EĞİLİMLERİN ANALİZİ

KARAAHMETOĞLU, Ebru
Kırıkkale Üniversitesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı, Doktora tezi
Danışman: Prof. Dr. Süleyman ERSÖZ
Ortak Danışman: Doç. Dr. Adnan AKTEPE
Ocak 2023, 107 sayfa

Metin madenciliği, metinsel verileri veri kaynakları olarak kullanan veri madenciliği yöntemidir. Metin madenciliği yönteminde, kaynak olarak alınan metin dilbilimsel olarak, özet, önemli, anahtar veri çıkarımı amaçlı olarak kullanılır. Metin veri madenciliği ile analiz edilen belgeler, öğrenme algoritmaları ile işlenerek analiz konuları ile ilgili tahminler üretilmeye çalışılır. Öğrenme modelleri denetimli ve denetimsiz öğrenme olmak üzere 2 farklı algoritma modeli bulunmaktadır. Bu çalışmada denetimsiz öğrenme algoritmalardan, kümeleşme yöntemlerinden yararlanılarak verilerin içerdiği ilişkiler ve özellikler ile ilgili çıkarımlar yapılmaya çalışılmıştır. Metin veri kaynağı olarak IPA raporları, metin madenciliği teknikleri ile analiz edilmeye çalışılmıştır. Çalışmadaki amacımız, mesleklerle ilgili metinlerden çıkarım yapmak ve bu çıkarımlarla tahminler yapmaya çalışmaktır. Mesleklere yönelik arama kriterlerine göre, internet kaynaklarından edinilen belgeler öğrenme ve tahmin belgeleri olarak 2 başlıkta ele alınmıştır. Belgeler üzerinde metin madenciliği süreçlerini işleterek, öğrenme kümesindeki belgelere, öğrenme algoritmaları uyguladıktan sonra, tahmin belgeleri ile mesleklere yönelik tahminler yaparak, sonuçları sunulmuştur. Çalışmadaki asıl amacımız geleceğin mesleklerine yönelik analizler yapmak ve tahminler üretmektir. Bu doğrultuda, geleceğin meslekleri arama kriteri ile erişilen belgelerde toplu frekans analizi süreci çalıştırılmıştır. Frekans analizi sonuçları, çeşitli kaynaklardan faydalanılarak oluşturulan geleceğin meslekleri listesi ile filtrelenmiştir. Geleceğin mesleklerine yönelik veri kümesi, makine öğrenme algoritmalarına tabi tutularak, tahminler üretilmiştir. Çalışmada elde edilen sonuçlar, çeşitli istatistiki değerlendirme yöntemleriyle değerlendirilerek, grafiksel gösterimleri yapılmıştır.

Anahtar Kelimeler: Metin madencliği, makine öğrenmesi, meslekler, denetimli öğrenme

ABSTRACT

PROFESSIONAL CHANGES AND TRENDS IN RECENT YEARS AND FOR THE FUTURE

KARAAHMETOĞLU, Ebru

Kırıkkale University

Graduate School of Natural and Applied Sciences

Department of Industrial Engineering, Ph. D. Thesis

Supervisor: Prof. Dr. Süleyman ERSÖZ

Co-Supervisor: Assoc. Doç. Dr. Adnan AKTEPE

January 2023, 107 pages

Text mining is a method of data mining in which the data to be analyzed is in text form. In the text data mining method, the text is used linguistically for extracting summary, important, key data. Documents analyzed with text data mining are processed with learning algorithms to produce predictions about analysis topics. There are two different classes of algorithm models in learning models. In supervised learning, there are data related to the model that we will try to teach the system, and by teaching this data to the system, the relationship between data input and output values is tried to be found. In unsupervised learning, there is only data. Just as there is no information on the subject to which the data is related, there is also no feedback on the forecast values. In these algorithms, it is tried to make inferences about the relationships and properties of the data by using clustering methods. In this study, first of all, IPA reports have been tried to be analyzed with text mining techniques. The data obtained in these analysis processes were visualized with frequency analysis and density graphs. Our main purpose in the study is to make inferences from texts about professions and to make predictions with these inferences. At this point, we divided the documents we downloaded according to the search criteria for professions into two classes as learning and estimation documents. By operating text mining processes on the documents, after applying learning algorithms to these documents in the learning set, we presented the results by making predictions about the professions with estimation documents. We evaluated the obtained results with various statistical evaluation methods and graphically displayed them. Documents accessed on the internet for future professions were subjected to text analysis and learning processes. The obtained results were evaluated by evaluation methods.

Keywords: text mining, machine learning, professions, supervised learning

TEŞEKKÜR

Tezimin hazırlanması esnasında hiçbir yardımı esirgemeyen ve bize büyük destek olan, bilimsel deney imkanlarını sonuna kadar bizlerin hizmetine veren, tez yöneticisi hocam, Sayın Prof. Dr. Süleyman ERSÖZ'e, tez çalışmalarım esnasında, bilimsel konularda daima yardımını gördüğüm hocam Sayın Prof. Dr. Ahmet Kürşad TÜRKER'e ve Sayın Araştırma Görevlisi Ali Fırat İNAL'a, fedakarlıklarla bana destek olan eşim Osman KARAAHMETOĞLU'ya ve her konuda desteğini esirgemeyen babam Erdoğan EROĞLU'ya teşekkür ederim.

İÇİNDEKİLER DİZİNİ

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

KISALTMALAR DİZİNİ

AGNES	AGglomerative NEsting
C4.5	Karar Ağacı Oluşturma Algoritması
C5.0	Karar Ağacı Oluşturma Algoritması
CART	Classification And Regression Trees
CHAID	Chi-Squared Automatic Interaction Detector
DIANA	Divise Analysis
ID3	Karar Ağacı Oluşturma Algoritması
J48	Karar Ağacı Oluşturma Algoritması
QUEST	Quick, Unbiased, Efficient Statistical Tree
SEER	Surveillance, Epidemiology, and End Results
SLIQ	Supervised Learning in Quest
SEER	Surveillance, Epidemiology, and End Results
SPRINT	Scalable Parallelizable Induction of Decision Tree
TF-IDF	Term Frequency - Inverse Document Frequency
YSA	Yapay Sinir Ağları

1. GİRİŞ

Bilgisayarların hayatımıza girmesiyle birlikte araştırmacılar için de yeni bir alan doğmuştur. Bu alan, bilgisayarlar tarafından çeşitli ortamlarda toplanan verilerin incelenmesi ve anlamlı sonuçların elde edilmesi işlemlerinden oluşur. Toplanan bu verilerin incelenmesi için çeşitli yöntemler geliştirilmiştir. Bilgisayar teknolojilerindeki gelişmeler, bilgisayar sistemlerinde saklanan veriler üzerinde analizler yapılarak, çıkarımlar yapmaya olanak sağlamıştır. Böylelikle, bilgisayar bilimlerinde "Veri Madenciliği" olarak isimlendirilen veri analizi teknikleri kullanılmaya başlanmıştır.

Veri Madenciliği, 60'lı yıllarda veri analizi sorunlarının bilgisayar ile çözülmeye başlanması ile ortaya çıkmış olup, "Veri Madenciliği" ismi 90'lı yıllarda bilgisayar mühendisleri tarafından ortaya atılmıştır.

Yapay zeka teknolojilerindeki ilerlemelerle birlikte, robot teknolojilerinin gelişmesi mesleklerde makineleşmeyi hızlandırmıştır. İş süreçlerindeki insan gücü ile yapılan işlerin yerini makinelerin, robotların alması, yapay zeka algoritmaları içeren bilgisayar programları ile işlerin tamamın veya belli kısımlarının gerçekleştirilmeye başlanması, gelecekte bazı mesleklerin kısmen veya tamamen ortadan kalkmasına neden olabilecektir. Bununla birlikte değişen iş modelleri ve süreçleri ile yeni meslekler ortaya çıkmaya başlayacaktır.

Mesleklerin değişimi aslında 150 yıldır insan hayatının doğal bir parçası olmuştur. Örneğin bundan 40 yıl öncesine ait olan "arzu halcilik" mesleği ortadan kalkmıştır; meslekler üzerindeki değişimler, bazen doğal süreçte yok olmakta, bazende güncelliğini yitirmektedir.

Özellikle üretim ve tarım alanında robotlaşma ve yapay zekanın endüstri 4.0 kapsamında kullanılmaya başlanmasıyla, bu alanlarda istihdam ihtiyacı azalacak ve bazı meslekler kaybolmaya başlayacaktır. Yapay zeka ve robotlaşma açısından cazip olmayan, örneğin bahçecilik, tesisatçılık gibi bazı mesleklerin, yine insanlar tarafından yapılmaya devam edileceği düşünülebilir. Kitle iletişim, psikologluk vb. insan

iletişimi gerektiren mesleklerin, kısa vadede yapay zeka tarafından geliştirilmeye başlanamayacağı öngörülmektedir. Yapay zeka ve robotlaşma ile değişen iş modellerinde yeni meslekler ortaya çıkacak olup, istihdamda bu alanlara kaymalar olması beklenmektedir. McKinsey raporuna göre, endüstri 4.0 ile gelen üretimde makineleşmeye bağlı olarak, 2030'a kadar, iş yaşamında harcanan işgücünün %30' unun robotlarca yerine getirileceği beklenmektedir. Bu tahminler üretimin içerdiği işgücü faaliyetleri, gerekli mesleki yetenekler ve meslekler açısından ülkeden ülkeye ciddi farklılıklar göstermektedir [1].

2015 - 2030 yılları arasında artan tüketim oranlarına istinaden artan gelirlerin tüketici malları üzerindeki etkisi sonucunda, gelişen ekonomilerde 250-280 milyon yeni mesleğin ortaya çıkacağı düşünülmektedir. Mesleklerdeki artış ve azalışlar sonucu, 75 ile 375 milyon kişinin meslek değiştirmesi beklenmektedir. Makineleşme sonucu iş modellerinde ortaya çıkacak değişimlerle, 2030'lu yıllara kadar, yaklaşık 800 milyon meslek sahibinin yerine getirdiği işlerin, robotlar tarafından geliştirilmeye başlanması öngörülmektedir. Bu nedenle, söz konusu meslek sahiplerinin yeteneklerine göre, yeni işlerde değerlendirilmesi gerekeceği düşünülmektedir. Gelecekte yeterli meslek olup olmayacağı ile ilgili duyulan endişeden ziyade, asıl sorun geleceğin mesleklerini yapabilecek yetenek ve bilgi birikimine sahip olup olunmadığıdır. Meslek değişimindeki bu geçişe ayak uyduramayan ülkelerde, işsizliğin artması, ücretlerin düşmesi ve ülkenin ekonomik olarak geri kalması kaçınılmazdır.

McKinsey raporuna göre, gelişmiş ekonomilerde makineleşme sonucu, istihdamda azalma görülen meslekler ortaöğretim düzeyindeki meslekler olmakta iken, üniversite düzeyindeki mesleklerin artması öngörülmektedir. Hindistan gibi gelişen ekonomilerde, ortaöğretim düzeyindeki mesleklerde de artış olması beklenirken, asıl büyük artış üniversite düzeyindeki mesleklerde olacaktır [1].

Ülke olarak temel önceliğimiz, eğitim programlarımızı, iş dünyasının ve üretim süreçlerinin ihtiyaç duyduğu yetkinlikte işgücü sağlayacak şekilde şekillendirmek ve gelişmiş işgücümüzün de yetkinlik ve bilgi birikimimizi zenginleştirmektir. Bununla birlikte, Bilim ve Sanayi Bakanlığı tarafından hazırlanan meslekler raporuna göre, ülkemizde yoğun bir genç nüfus işsizliği bulunmaktadır. Bu işsizliğin nedeni ise üniversite eğitim programlarının, sanayinin ihtiyaçlarına yönelik olarak tasarlanmış olmamasıdır. Çalışanları, işgücü piyasasanın mevcut ve

değişen iş modelleri sonucu ortaya çıkan mesleklerin gereksinimlerine göre hazırlamak, eğitimden geçmektedir [2].

Gelişen dünya düzeninde, işçilik maliyetlerinin düşük olması yatırımları ülkeye çekmeye yeterli olmamakta, eğitimli ve nitelikli işgücü, sermaye hareketlerinde önemli bir rol oynamaktadır. Artık ülkeler, beşeri sermaye yatırımlarına önem vermekte ve bu vesile ile siyasi, politik ve ekonomik güç elde etmektedir. Kısacası yeni dünya düzeninde, eğitim ve kalkınma arasında sıkı bir ilişki vardır. Modern dünya düzeninde beşeri sermayeler önemli olmakla birlikte, ülkeler bu yönde yatırımlar da yapmalıdır [3].

Günümüzde sanayi devrimin üzerine inşa edilen ve dijital devrim olarak isimlendirilen dördüncü sanayi devrimi yaşanmaktadır. Dördüncü sanayi devrimiyle gelen değişimler, teknoloji, yönetim, üretim vb. alanlardaki geçmiş değişimlere göre, hiç bu kadar yıkıcı olmamıştır [4].

Dördüncü sanayi devrimindeki değişimin temeli imalat sektöründe olup, firmalar ve firma ile müşterileri arasında iletişimi sağlayan teknik platformlar, yeni fırsat alanları oluşturmuştur. İmalat sanayindeki büyümenin ülke ekonomilerine büyük katkısına rağmen ülke ekonomileri büyüyüp zenginleştikçe, GSYH içerisindeki payının artmasıyla tüketim de artmış, hizmet sektörü büyümüş ve imalat sanayinin payı azalmıştır. Dolayısıyla gelişmiş ve ilerlemiş ekonomilerde imalat sanayinin ekseni üretkenlik, verimlilik ve sürdürülebilirliği tetikleyen bir yeniliğe doğru kaymıştır. Bu kayma, yeni müşteriler ve düşük maliyetli üretim sağlayan, gelişmekte olan ülkeler ile küresel bir imalat ortamına önayak olmuştur. Hizmet sektöründeki bu gelişmeleri gören gelişmiş ülkeler, imalat sanayini ucuz işgücüne kaydırarak, bilgi, teknoloji, inovasyon ve bilime yatırım yapmışlardır.[5].

Endüstri 4.0 ile gelen iş süreçlerinde, değişimi üretim süreçleri ve iş modelleri ile etkin olarak pekiştiren ve değişim süreçlerini etkin olarak yöneten ülkelerin, refah düzeylerinin daha yüksek olması kaçınılmazdır. Bu savın göstergesi ise, küresel ölçekte tüketim mallarının değerinin, 2030 yılına kadar, 2015 yılı değerinin iki katına ulaşmasının beklenmesidir [5].

Üçüncü sanayi devriminde sayısal sistemler, üretim sistemine entegre edilerek, bilgisayar kontrollü sistemlerin üretim sisteminde kullanılmaya başlamasıyla, büyük

sıçrama yapılmıştır. Üretim sektöründeki montaj hatlarında, bilgisayarların bazı insani görevleri üstlenmeye başlamasıyla, iş modellerinde değişimler başlamıştır.

Dördüncü sanayi devrimi ile üretimin sayılaştırılması ve iletişim teknolojilerinde tümleşik sistemler kullanan "küçük miktarda bireyselleştirilmiş ürünler" in üretiminin yapılacağı fabrikalara ihtiyaç oluşması beklenmektedir. Kablosuz teknolojiler gibi uculayan yeni teknolojilerin kullanımı sonucu, üretim süreçleri ile ilgili daha fazla ve farklı bilgilerin toplanması, bu bilgiler ile üretim ve planlama süreçlerinin iyileştirilmesi ve üretim miktarlarının artırılması beklenmektedir Toplanan veriler akıllı karar verme süreçlerinde kullanılabilecek olup, üretimdeki robot ve insan iletişiminde önemli rol oynayarak, üretim süreçlerini iyileştirecek ve etkin bir iş modeli sağlayacaktır [2].

Ülkelerin dördüncü sanayi devrimine yönelik hazırladıkları programlara bakılacak olursa, Çin endüstri 4.0 ile gelecek değişimleri önceden görerek, sanayide lider ülke olma ve bilgi teknolojilerinde ilerleme hedefiyle eylem planlarını oluşturmuştur [6], [7]. Gelişmiş ekonomiye sahip Almanya ve Amerika gibi ülkeler ekonomik güçlerine korumak ve değişen iş modellerine ayak uydurmak için, üretimde makineleşme, diğer bir deyişle, üretim süreçlerinde büyük önem teşkil etmesi beklenen veri bilimi ve yapay zeka alanında yatırımları desteklemişlerdir.

Ekonomilerini ve imalat sektörlerini güçlendirmek adına, ABD ve Almanya "günümüz teknolojik devrimi"ni, diğer bir adlandırma ile "sayısal endüstrilerini" destekleyecek teşebbüslerin finansmanında etkin rol almışlardır.

Dördüncü sanayi devrimi iş süreçlerinde büyük değişimlere neden olarak, yeni iş modellerini ortaya çıkaracaktır. İş süreçlerinde bu büyük değişimlerin, ortaya çıkacak olan yıkıcı teknolojiler aracılığı ile olması beklenmektedir. Üretim süreçlerinde lider olmayı planlayan ülkeler şimdiden geleceği yönetmek adına çıkardıkları bu programlarla, yıkıcı teknolojileri yönetmeye, değişimin riskini kontrol altına almaya çalışmaktadır. İş süreçlerindeki bu değişimlerle ortaya çıkan yeni mesleklere, ortadan kalkan mesleklere, kısmi olarak işgücü ve robot paylaşımı ile gerçekleştirilen mesleklere, inovasyon, bilgi aktarımı ve eğitimlerle, işgüçlerini hazırlamaya çalışmaktadır.

Dördüncü sanayi devrimi ile, iş süreçlerindeki ucuz işgücüne sahip olan gelişmekte olan ülkelere üretimi kaydırma eğilimin değişmesi, üretim faaliyetlerinin yeniden

teknolojileri elinde tutan gelişmiş ülkelere geri çağrılması eğiliminin, Türkiye gibi gelişmekte olan ülkeleri etkilemesi beklenmektedir. Üretim süreçlerinde robotlaşma, ucuz işgücü yerine eğitimli, nitelikli işgücüne ihtiyacı artıracak olup, gelişmiş ülkelerin ucuz işgücü ihtiyacı azalacak ve üretimi kendi ülkelerinde gerçekleştirmeye başlayacaklardır. Bu nedenle gelişen ülkelerde, ciddi bir işsizlik sorunu ile karşı karşıya kalınması beklenmektedir.

Bu çalışma kapsamında, yukarıda bahsedilen büyük değişimlerin mesleki alanlardaki etkilerini analiz etmek üzere, metin madenciliği yöntemlerini kullanarak, IPA raporları üzerinde analizler yapılmıştır. Çalışmanın temel aldığı teorik bilgiler ve elde edilen deneysel sonuçlar, ilerleyen bölümlerde sunulacaktır.

2. LİTERATÜR TARAMASI

2.1. Veri Madenciliği Literatürü

Veri madenciliği alanında ulusal ve uluslararası olmak üzere çok fazla sayıda yayın, makale ve kitap bulunmaktadır. 1990'lardan günümüze kadar yayınlanmış birçok makalesiyle David Hand başta gelmektedir. İstatistikle veri madenciliğini ele aldığı makalesinde, veri madenciliğinin istatistiğin bir alt kümesi olduğunu, fakat başka alanlarda da kullanılabileceğini belirtmiştir [8]. İstatistikle veri madenciliği arasındaki ilişkiyi araştıran diğer bir kişi ise Jerome H. Friedman'dır. Makalesinde veri madenciliğinin ve istatistiğin uygulama alanlarını incelemiştir [9]. Bilgisayar teknolojisinin gelişmesiyle birlikte, veri madenciliğine olan ilgi artmış ve bu konuda Han ve Kamber [10], Larose [11], Bramer [12], Nisbet, Elder ve Miner [13] veri madenciliği yöntemlerini, modellerini ve uygulama alanlarını kitaplarında detaylı bir şekilde ele almışlardır.

Son yıllarda ülkemizde de veri madenciliğine olan ilgi artmıştır. Örneğin Akpınar makalesinde veri tabanlarında bilgi keşfi ve veri madenciliği kavramlarını ele alırken [14], Tüzüntürk istatistik ile veri madenciliği arasındaki ilişkiyi ele almıştır [15]. Bir diğer çalışma, Savaş, Topaloğlu ve Yılmaz tarafından gerçekleştirilmiş ve veri madenciliğinin Türkiye'deki uygulama alanları incelenmiştir. Sektördeki çeşitli alanlarda uygulamalara yer verilen çalışmada incelenen araştırmalarda, kurum ve kuruluşların çoğunun müşteri/kullanıcı analizlerine yöneldiği belirlenmiştir [16].

Veri madenciliği alanında birçok lisansüstü çalışma yapılmıştır: Koyuncugil doktora tezinde, bulanık veri madenciliği ve sermaye piyasalarını ele almıştır [17]. Küçüksille doktora tezinde, veri madenciliği yöntemini kullanarak portföy performanslarını değerlendirmiş ve IMKB hisse senetleri üzerine bir uygulama yapmıştır [18]. Bu çalışmaların yanında veri madenciliği ve bilgi keşfiyle ilgili çeşitli kitaplar yayınlanmıştır. Şentürk kitabında veri madenciliği ürünü olan Rapid Miner uygulamasıyla veri madenciliği yöntemlerini incelerken [19], Gürsoy kitabında veri

madenciliği ile ilgili yurtdışında yaptığı araştırmalarından yola çıkarak sektörel veri madenciliği uygulamalarını ele almıştır. Ayrıca Aylık Sanayi Üretim anket verileri ile hesaplanan Sanayi Üretim endeksi hakkında da literatür taraması yapılmıştır [20]. Ekonominin kısa dönemde gelişiminin önemli bir göstergesi olan, Sanayi Üretim Endeksini ekonometrik açıdan inceleyen çok sayıda çalışma bulunmaktadır. Kızılca, yüksek lisans tezinde Türkiye'nin 1980-2001 dönemini incelemiş, sanayi üretim endeksini etkileyen faktörleri belirlemiş, zaman serisi yöntemiyle analizler yapmıştır. Sanayi üretim endeksi bağımlı değişken olmak üzere, sanayi ürünlerin ihracatı, ithalatı, işyeri sayısı, istihdam, reel katma değer ve reel ücret değişkenleri bağımsız değişken olarak belirlenmiştir. Oluşturulan ekonometrik modelle, zaman serisi analizi kullanılarak öngörü modeli oluşturulmuştur [21].

Veri madenciliği yöntemleri ile çok büyük boyutlardaki veriler üzerinde çeşitli analizler yapılabilmektedir. Günümüzde gittikçe artan veri yığınları ve veri madenciliği yöntemlerinin elverişli yapısı nedeniyle çok fazla sayıda araştırma yapılmaktadır.

Ulusal ve uluslararası literatürde, veri madenciliği tekniklerinin çeşitli alanlara uygulandığı birçok çalışma bulunmaktadır. Bu çalışmalar sosyal bilimler alanında yoğunlaşmakta olup, tıp ve mühendislik alanları görece olarak az sayıdadır. Baykasoğlu [22] ve King vd. [23], karar ağaçları algoritmaları, istatistiksel algoritmalar ve yapay sinir ağları teknikleri arasında karşılaştırma yaptıkları çalışmalarında, çeşitli veri kümeleri arasında hangi algoritmaların daha iyi sonuçlar ürettiğini incelemeyi amaçlamışlar ve iyi sonuç üretenlerle, analiz edilen veriler arasında ilişki tespit etmişlerdir.

Özden ve Chen (1999), plastik sanayisinde reçine üretiminde kullanılan enjeksiyonlu döküm işleminde ürün kalitesini arttırmak amacıyla veri madenciliği yöntemlerinden sinir ağları ve karar ağaçlarını kullandıkları çalışmalarında, oluşturulan sinir ağları modellerinin ürün kalitesini arttırıcı bir yöntem olduğunu belirtmişlerdir [24].

Skinner vd. levha üretim işleminin, levhaların kalitesi ve üretim verimindeki etkisini inceledikleri çalışmalarında, veri madenciliği yöntemlerinden CART karar ağacı yöntemini kullanmışlar ve CART karar ağacının düşük verimi engelleyici bir yöntem olduğunu rapor etmişlerdir [25].

Li vd., çalışmalarında, kaplanmış cam üretiminin geliştirilmesi amaçlı mekaniksel, kimyasal, elektriksel ve manyetik işlemlerdeki değişkenleri, CART ve yapay sinir ağları metotları ile incelemişler, CART ve sinir ağları yaklaşımlarının, üretim hattındaki makine ve kalite ölçümleri arasındaki ilişkiyi modellemede ve cam kalitesini artırmada, umut vaat eden sonuç verdiğini bulmuşlardır [26].

Koyuncugil (2006) tez çalışmasında, Bulanık Veri Madenciliği' ne dayalı olarak veri madenciliği yöntemlerinden, k-ortalamalar, bulanık kümeleme, bulanık hedefli CHAID karar ağacı ve algoritması ve önsel birliktelik kuralları algoritmalarını ardarda kullanmış ve hisse senetleri piyasasında manipülasyon tespitine yönelik, bir erken uyarı sistemi geliştirmiştir [17].

Çoban, üretim sektöründe faaliyet gösteren bir şirketin verilerini kullanarak işletmenin tedarikçi seçimine dair yaptığı modelleme çalışmasında, yapay sinir ağları (YSA), CART, CHAID ve QUEST, kümeleme algoritmalarından ise k-ortalamalar ve Kohonen yöntemlerini kullanmıştır. Analizlerde elde edilen etkinlik grafikleri incelendiğinde, YSA algoritmasının başlangıçta, karar ağacı algoritmalarının ise süreç içerisinde daha etkin olduğu gözlenmiştir [27].

Özçınar (2006) çalışmasında, KPSS sınavına girecek öğrencilerin sınav sonuçlarını, veri madenciliği tekniklerinden yapay sinir ağları ve çoklu regresyon tekniklerini kullanarak önceden tahmin etmeye yönelik bir model oluşturmuştur. Modellerin ürettikleri hata değerleri farkı çok yüksek olmasa da, yapay sinir ağı modelinin, çalışmada kullanılan bütün veri kümelerinde, regresyon analizi modelinden daha başarılı olduğu görülmüştür. Bu sonuç, yapay sinir ağlarının, eğitim alanındaki tahmin içerikli çalışmalarda, alışılmış istatistik yöntemlerinin yerine tercih edilebileceğini göstermiştir [28].

Dolgun, vd. çalışmasında, birliktelik kuralları ve karar ağaçları yöntemleriyle öğrenci seçme sınavı verilerini analiz ederek, meslek tercihi eğilimlerini belirlemeye çalışmışlardır [29].

Rajavarman, vd. çalışmalarında, genetik algoritma kullanarak geliştirdikleri sınıflandırma yöntemiyle, ID3, hızlandırılmış ID3 ve yapay sinir ağı algoritmalarının başarım karşılaştırmasını yapmışlardır. Başarım karşılaştırmalarını farklı veri kümelerini kullanarak yapmışlar ve sundukları sınıflandırma modelinin daha iyi sonuçlar ürettiğini belirlemişlerdir [30].

Bozkır, vd. (2008) üniversite öğrencilerinin eğitimle ilgili internet erişimlerini, veri madenciliği teknikleriyle analiz ederek, öğrencilerin düşünce ve davranışlarının eğitim amaçlı internet erişimlerine etkilerinin örüntüsünü çıkarmaya çalışmışlardır [31].

Söylemez çalışmasında, puan kartı veri madenciliği, lojistik regresyon ve karar ağaçları tekniklerini kullanarak bir bankanın kredi değerlendirme sonuçlarını tahmin eden bir puan kartı modeli kurmuştur [32].

Çetin çalışmasında, bir üretim işletmesinde üretilen ürünlerin yanlış ayrılmasının nedenlerini belirlemek ve yapılan hataları azaltacak stratejiler geliştirmek için, yapay sinir ağı ve karar ağacı algoritmalarını kullanarak. modeller geliştirmiştir [33].

Coşkun çalışmasında, göğüs kanseri vaka kayıtlarını içeren SEER veri kaynağını üzerinden J48, naive bayes, lojistik regresyon ve kstar algoritmalarının karşılaştırmasını yapmıştır. Yapılan karşılaştırma sonucuna göre, eldeki veriler üzerinden çalıştırılan J48 algoritmasıyla diğer algoritmalara göre daha iyi sonuçlar elde edilmesine, ancak modellerin birbirine önemli bir üstünlük gösteremediğini ifade etmişlerdir [34].

Bilekdemir çalışmasında, su sayaçlarının üretim süresinin tahmininde yapay sinir ağı, genetik ve karar ağacı gibi sınıflandırma algoritmaları kullanan veri madenciliği teknikleriyle, üretim süresinin tahmin edilebileceğini göstermiştir [35].

Tokmak, geliştirdiği uygulamayla, veritabanında bilgi keşfiyle veri madenciliği arasında bağlantı olup olmadığını belirlemeye çalışmıştır. [36].

Çiftlikçi ve Özyirmidokuz çalışmalarında, halı üretim işletmesinde üretilen halıların kalitesini artırmak amacıyla, sınıflama ağaç modelleri için C4.5 ve hata nedenlerini tahmin etmek için ise C5.0 algoritmalarını kullandıkları çalışmalarında, karar ağaçları yöntemiyle, işlemdeki hataları bulmuş ve hatalı ürünlerin üretimini engellemişlerdir [37].

Ertuğrul vd. çalışmalarında, Pamukkale Üniversitesi Hastane Bilgi Yönetim Sisteminde yer alan verileri kullanarak, veri madenciliği yöntemleri yardımıyla, hastaneye gelen hastaların profilini belirleyen bir uygulama çalışması yapmışlardır [38].

Hanehalkı İşgücü Araştırması ile elde edilen veriler üzerine de çeşitli veri madenciliği yöntemleri uygulanmıştır. Oğuzlar (2004) çalışmasında, Türkiye'nin 2002 hanehalkı

işgücü anketi verileri, CART algoritması kullanarak analiz etmişlerdir. Kişilerin iş arama durumlarını belirleyen özelliklerin, eğitim durumu, cinsiyet, medeni durum ve referans kişiye yakınlık olduğunu, CART karar ağacı oluşturarak sunmuştur [39].

Yılmaz (2012) çalışmasında, 2009 ve 2010 yılları Hanehalkı İşgücü Anketi verilerine CHAID algoritmasını uygulamış ve Türkiye'nin 2009 ve 2010 işgücü istihdam durumunu belirlemeye çalışmıştır [40].

2.2. Metin Madenciliği Literatürü

Literatürde, metin sınıflandırma konusundaki çalışmalar siber güvenlikten sağlık alanlarına kadar farklı alanlarda, farklı teknikler kullanılarak yapılmıştır.

Rong, Yuancheng ve Xiangqian yaptıkları çalışmada, derin inanç ağları kullanarak geliştirdikleri spam sınflandırmayla, internet ortamındaki zararlı içerik ve programları engellemeye çalışmışlardır [41].

Agren çalışmasında, yalan haberlerle insanların yatırım tercihlerinin yanlış yönde etkilendiği, popüler basın organlarında yaygınlaşan yanıltıcı haberlerle sahtekarlıkların arttığını belirtmişlerdir [42].

Cardoso ve Silva çalışmasında, yanlış ve taraflı yorumların, müşterilerin tüketim tercihlerini olumsuz yönde etkilediğini belirlemişlerdir [43].

Kudugunta ve Ferrara çalışmalarında, otomatik olarak oluşturulan ve bir ülkenin seçim ve sağlık sistemini hedef alan, yalan yanlış twitter verilerini belirlemişlerdir [44].

Figueira çalışmasında, basılı ve İnternet ortamında yayınlanan gerçek olmayan haberleri, gizli Markov modelleri kullanarak belirlemeye çalışmıştır [45].

Yao ve Zhi-Min çalışmasında, konu bazlı yazı sınıflandırmasında, destek vektör makineleri, k-en yakın komşu, naive bayes vb. sınıflandırma algoritmalarına göre iyi sonuçlar üreten, DNB adını verdikleri konusal yazı sınıflandırması geliştirmişlerdir [46]. Trstenjak, Mikac ve Donko, metin madenciliğinde, knn makine öğrenme algoritmasının iyi sonuçlar ürettiğini belirttikleri ve belgelerin analiz konusuyla ilgili olup olmadığını belirlemek amacıyla TF-IDF algoritmasını kullandıkları bir çalışma yapmışlardır [47].

Liang ve arkadaşları geliştirdikleri çalışmada, internet ortamından kimya bilimiyle ilgili belgeleri toplamışlar ve bu belgeleri metin madenciliği teknikleriyle analiz etmek için, sözlük yaklaşımını kullanan metin sınıflandırma algoritması geliştirmişlerdir [48].

Kotevska, Padi ve Llbath, Twitter, Facebook vb. sosyal medya ortamlarında yapılan paylaşımların toplum üzerinde önemli etkileri olduğunu ve toplumun nabzının veri madenciliği teknikleri ile belirlendiği duygu analizi algoritmalarına kaynak veri sağlayacağını öngürmüşlerdir. Çalışmada geliştirilen NLP ve rastgele orman algoritmasını kullanan sınıflandırma algoritması, destek vektör makineleri ve naive base algoritmalarına göre daha iyi sonuçlar üretmiştir [49].

Gelişen teknolojiler, günlük ve iş yaşamındaki değişimler, internet ortamında yayılan verilerin hacmini ve belgelerin metinsel büyüklüğünü artırmıştır. Qazi ve Guadar çalışmasında, metin dosyaların sayısal ve boyutsal büyüklüğünün, metin madenciliği analiz süreçlerini zorlaştırdığı, bu durumun sınıflandırma analizlerinin gerçekleştirilmesinde ontoloji tekniklerini kullanan terim ağırlıklandırma yöntemi sunmuşlardır [28] [50].

2.3. Endüstri 4.0 Literatürü

İlk olarak Almanya'da kullanılmaya başlayan endüstri 4.0 terimi, yüksek teknoloji stratejilerini temel almıştır [51]. Rüßmann ve ark. Endüstri 4.0' ın Almanya'da çeşitli sektörlerde büyüme sağlayacağını öngörmüşlerdir [52]. Endüstri 4.0 ile organizasyon yapıları, süreçler, personel ve iş yapış biçimlerinde birçok değişiklik olmuştur [53].

Endüstri 4.0'ın işgücü istihdamına etkileri, meslek ve sektörlere göre farklılık göstermektedir. McKinsey raporuna göre, üretimde makineleşme sonucunda, öncelikle fiziki güçle yapılabilen mesleklerle birlikte veri işleme ve toplama alanındaki mesleklerin de robotlarca gerçekleştirileceği beklenmektedir [1].

Ülkelerin dördüncü sanayi devrimine yönelik hazırladıkları programlara bakılacak olursa, Çin bilgi teknolojileri alanında yetkinliklerini geliştirerek, sanayide lider ülke olmayı hedeflemektedir [54]. Amerika ve Almanya, endüstri 4.0' la gelen değişikliklere hazırlanmak ve ekonomilerini güçlendirmek amacıyla, bilgi teknolojileri faaliyetlerine yatırım finansmanı yapmışlardır. Oxford Üniversitesinden

Frey ve Osborne'ın çalışmasına göre, Amerika'daki güncel işlerin %47' si, Almanya için yapılan bir çalışmaya göre %59' u, OECD ülkeleri için %57' si, İsveç için yapılan bir çalışmaya göre ise %50' sinin bilgisayar teknolojilerince yerine getirilmesi beklenmektedir [55] [56]. Öte yandan, üretim süreçlerinde makineleşme, ucuz işgücü nedeniyle doğu ülkelerinde yapılan üretim faaliyetlerinin, Amerika ve Kuzey Avrupa ülkelerine geri dönmesini sağlayacağı öngörülmektedir [57].

Macurova, Ludvik ve Žwakova' nın 2017 yılındaki çalışmasına göre, organizasyonlar Endüstri 4.0' la gelen değişimlerin gerektirdiği yetenekte işgücü bulamama riskine maruz kalacaklardır. Değişimin hızlı olması nedeniyle, yeni mesleklere yönelik eğitim stratejilerinin uygulanmasında geç kalınacak, dolayısıyla ihtiyaç duyulan yetkinlikte eleman ihtiyacı karşılanamayacaktır [58]. Weber'in 2016 yılında yaptığı çalışmaya göre, Endüstri 4.0 ile vasıfsız işçilerde kısa süreli bir ihtiyaç azalması beklenmektedir. Endüstri 4.0 ile düşük seviye yetenek gerektiren işlerin azalması beklenirken, hizmet sektöründe, özellikle bilgi teknolojileri ve bilimsel meslekler gibi yüksek yetenek gerektiren işgücünde, arzın artması beklenmektedir [59]. Kane vd. çalışmasında, eğitim stratejilerinde ve programlarında yeni mesleklere yönelik değişimler olacağından bahsedilmiştir [60]. Tüm ülkelerin eğitim sistemleri ve vasıflarını, üretimde dijitalleşme kapsamında analiz ederek, güçlü ve zayıf yanlarını belirlemesi gerekmektedir [61]. İlkokuldan başlayarak, eğitim müfredatlarının bilişim derslerine yer verecek şekilde güncellenmesi, değişimin gerektirdiği yetenekte işgücü hazırlamak açısından önemlidir [62].

3. KAVRAMSAL ÇERÇEVE VE YÖNTEMLER

3.1. Veri Madenciliği

İlk defa 1995'te gerçekleştirilen, veritabanlarında bilgi keşfi konferansında bilgi teknolojilerinin oluşturduğu veri yığınları şu şekilde ifade edilmiştir [63]:

"Dünyadaki bilgi miktarının her 20 ayda bir, ikiye katlandığı tahmin edilmektedir. Bu ham veri yığınları ile ne yapılması gerekmektedir? Bilgisayarlar veri çöplüğüne neden olmaktadır."

Winter Corporation' ın işletmelerde kullanılan veritabanlarının kapladığı hacmi belirleyerek, en büyük veritabanını bulma amaçlı olarak gerçekleştirdiği çalışmaya göre Sears, Roebuck ve Co. kuruluşuna ait karar destek veritbanı, 1998'de 4630 gigabyte veri boyutuna ulaşmıştır.

Veritabanı sistemlerinde veri boyutlarındaki bu hızlı artış, veri uzmanlarını yeni yöntemler arayışını itmiştir. Bu noktadan hareketle bilgi keşfi, veri ambarı, veri madenciliği vb. kavramlar ve teknolojiler ortaya çıkmıştır [14].

3.1.1. Veri Madenciliği Nedir?

Veriler büyük hacimlerde yer kaplamalarına rağmen kullanım değerleri fazla değildir. Verilere uygulanan belli dönüşümler sonucu elde edilen çıkarımlara bilgi denir. Bilgi, verilere göre hacim olarak daha az yer tutmakla birlikte, kullanım değeri yüksektir [64].

Günümüz teknolojileri işletme işleri gerçekleştirilmesi için geliştirilen veritabanları sistemlerinde terabyte'larca veri tutulmaktadır. İşletmelerde üst yönetim gelecek projeksiyonları ve stratejilerini belirlemek için, veri yığınının içinde saklı ve önemli bilgileri ihtiyaç duymaktadırlar. Büyük hacimde veri yığınından, çeşitli algoritma, teknik ve süreçlerin kullanımıyla, önemli bilgi ve örüntülerin çıkarımının yapılması, veri madenciliği olarak isimlendirilir.

Piatetsky-Shapiro'ya göre, veri madenciliği, büyük veri yığınlarından, önemli ve anahtar bilgi çıkarımı yapılmasıdır [65].

Veri madenciliği basitçe, insan ve alışılmış tekniklerce büyük emek ve çalışma zamanı harcanarak elde edilebilecek, bilgi ve örüntü çıkarımının çeşitli öğrenme algoritmaları ve analizlerle elde edilmesidir [66].

Diğer bir deyişle, veri madenciliği, büyük miktarda veri içinden, gelecekle ilgili tahmin yapmamızı sağlayacak bağıntı ve kuralların aranmasıdır [67].

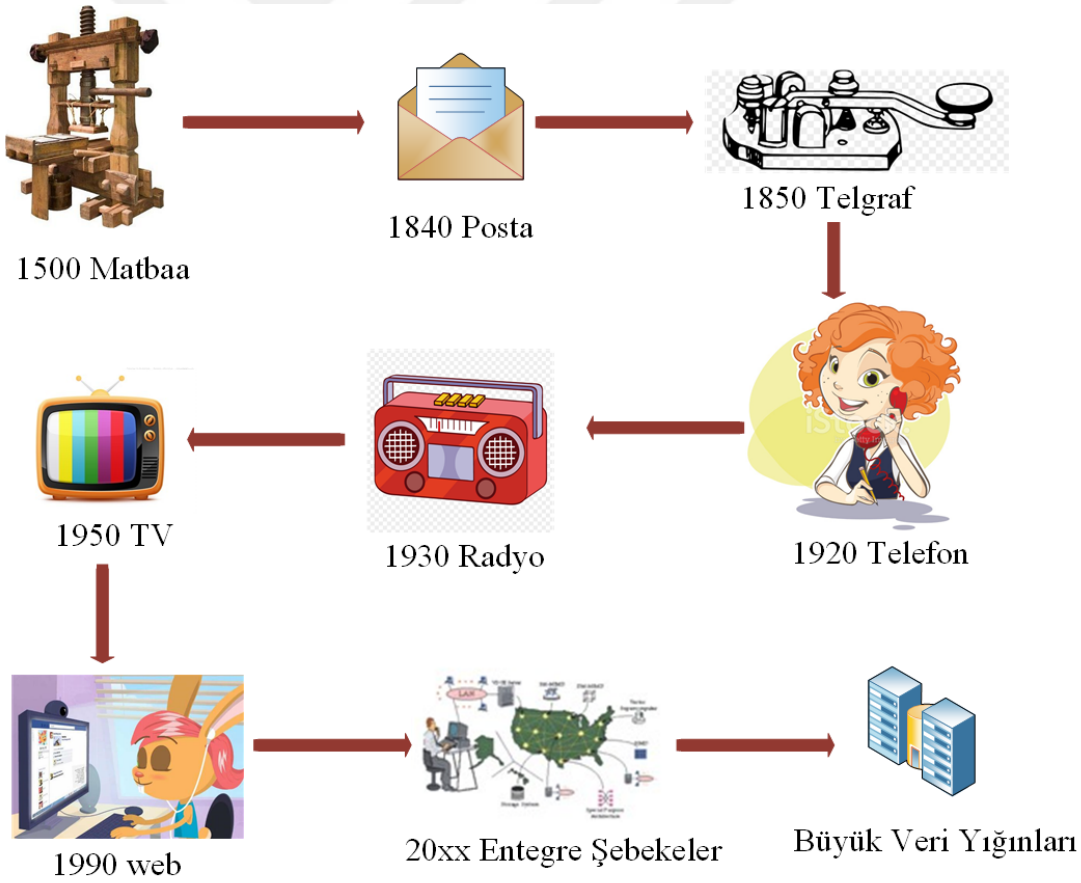
Veri madenciliği, çeşitli analiz teknikleri ve araçlar kullanılarak, analize konu veri yığınlarının içerdiği örüntü ve ilişkilerin çıkarımı yapılarak, tahminler üretilmesidir. Büyük veri yığınları içerisinde bilinmeyen bilgileri belirlemeyi amaçlayan veri madenciliğinde, kontrol veridedir. Analistin kendi düşünce ve beklentilerine göre, analizi yönlendirme şansı yoktur [66].

Veri madenciliği çıplak haliyle bir anlam ifade etmeyen veriden, önemli bilgi çıkarımı yapmayı amaçlar. Diğer bir deyişle büyük bir veri yığınından, işletme için önemli bilginin çıkarımının yapılmasıdır. Kısacası, veri madenciliği farklı kaynaklardan elde edilen kurum verilerini analiz ederek, veriler arasındaki ilişkileri belirleme ve gelecekle ilgili tahminler yapmaya çalışma sürecidir. Kurumun geleceği ile ilgili stratejik açıdan önemli bilgilerin bulunmasını sağlaması açısından, veri madenciliği karar destek sistemleri için önemlidir [67].

3.1.2. Veri Madenciliğinin Tarihsel Keşfi ve Gelişimi

- ✓ Bilgisayar 1950'li yıllarda kullanılmaya başlamıştır.
- ✓ 1960'lı yıllarda, veriler düz ve hiyerarşik dosyalarda saklanarak veri koleksiyonları oluşturulmuştur.
- ✓ Günümüz veritabanı uygulamalarında, verileri sakladığımız ilişkisel veritabanı sistemleri, 1970'li yıllarda kullanılmaya başlamıştır.
- ✓ 1980'li yıllarda veritabanı uygulamalarının kullanımı artmaya başlamıştır. İşletmeler işlerini yerine getirecek otomasyon projelerini geliştirmeye başlamışlardır.

- ✓ Veritabanı uygulamalarınca üretilen ve saklama ortamlarında tutulan veri yığınlarından, işletlemelerin stratejik kararlarına katkıda bulunacak bilgi çıkarımı yapılıp yapılamayacağı, 1990'lı yıllarda sorgulanmaya başlamıştır.
- ✓ 1989 yılında, büyük veri yığınlarından bilgi çıkarımı çalışmaları konusunda, bilgi keşfi grubunun ilk toplantısı yapılmıştır.
- ✓ Piatetsky-Shapiro 1991 yılındaki çalışmasında, karar destek sistemlerinin temel kavram ve tanımları üzerinde durmuştur [65].
- ✓ İİk veri madenciliği uygulaması, 1992 yılında geliştirilmiştir.
- ✓ 1995 yılında veri madenciliği ve bilgi çıkarımına yönelik uluslararası bir konferans düzenlenmiştir.
- ✓ Veri ambarı ve veri madenciliği uygulamalarının, veri analiz projelerinde kullanımı yaygınlaşmıştır.



Şekil 3.1. Veri Madenciliğinin Tarihsel Keşfi ve Gelişimi

Veri madenciliği, kavramsal olarak 1960'lı yıllarda bilgisayarların, veri analizi problemlerinin çözümünde, kullanılmaya başlanmasıyla ortaya çıkmıştır. Bu dönemlerde, bilgisayar ve veritabanlarının, kısıtlı hesaplama ve depolama yetenekleri nedeniyle, veriden bilgi keşfinin, uzun taramalar sonucu olacağı kabul edilmiştir. Bu nedenle, veri taraması veya veri yakalaması kavramları veri madenciliğinde kullanılmıştır [67].

1990'lı yıllarda veri analizlerinde, geleneksel istatistik yöntemleri yerine, algoritmaların kullanılmaya başlanması ile veri madenciliği ismi kullanılmaya başlanmıştır. Bu gelişmeden hareketle, bilim adamları, istatistik, veri bilimi, makine öğrenmesi vb. teknikleri kullanan uygulamalar geliştirerek, veri madenciliği yapmaya çalışmışlardır [67].

Bilgisayarların veri analizinde kullanılmaya başlanması ile birlikte, daha önceden yapılması mümkün olmayan istatistiki yöntemler geliştirilerek, veri analizlerinde kullanılmaya başlanmıştır. Böylelikle verinin keşfi sürecinde, istatistik ve veri madenciliği güçlü bir işbirliği içinde kullanılmaya başlanmıştır [67].

Günümüzde Endüstri 4.0 ile endüstri ve bir çok alanda yapay zeka kullanımının artmasıyla, makine öğrenmesi büyük önem kazanmıştır. Yapay zeka algoritmalarının insan öğrenmesini taklit eden yapısı kullanılarak, özel öğrenme problemlerine yönelik algoritmalar üretilmeye başlanmasıyla, istatistik ve makine öğrenme kavramları, veri madenciliği alanında kullanılmaya başlanmıştır.

3.1.3. Veri Madenciliğinde Verinin Önemi

Veri madenciliği, istatistik, veri tabanları, yapay sinir ağları, örüntü tanıma, makine öğrenmesi, veri analizi ve görselleştirme olmak üzere çeşitli disiplenlerden algoritma ve teknikler içerir. Bu disiplinlerle veri madenciliği iç içe girmiş olup, süreç olarak birbirlerini tamamlamaktadırlar [68].

Veri madenciliğinde kullanılacak verilerin, aşağıda anlatılan ölçütleri sağlaması gerekmektedir:

- ✓ *Verilerin Elde Edilebilirliği:* Analiz konusu ile ilgili veriler, analiz için uygun biçime çevrilir.

- ✓ *Veriler İlgili Nitelikleri Sağlıyor mu?* Elde edilen değişkenler irdelenerek, kayıt edilmemiş değişkenler belirlenmeye çalışılır.
- ✓ *Veriler Gürültülü mü?* Verilerin hatalı veriler içermesi durumudur. Verilerin içerdiği gürültü oranı ne kadar çok olursa, analiz sonuçlarının güvenilirliği o derecede azalır.
- ✓ *Verilerin Yeterliliği:* Veri madenciliğinde verilerin yeterliliği, verilerin büyüklüğüne göre değil, ne kadar çok nitelik içerdiğine göre belirlenir.
- ✓ *Verilerle İlgili Uzman Bilgisi:* Analiz edilecek verinin uzmanlık alanına göre uzman görüşünün alınmasıdır.

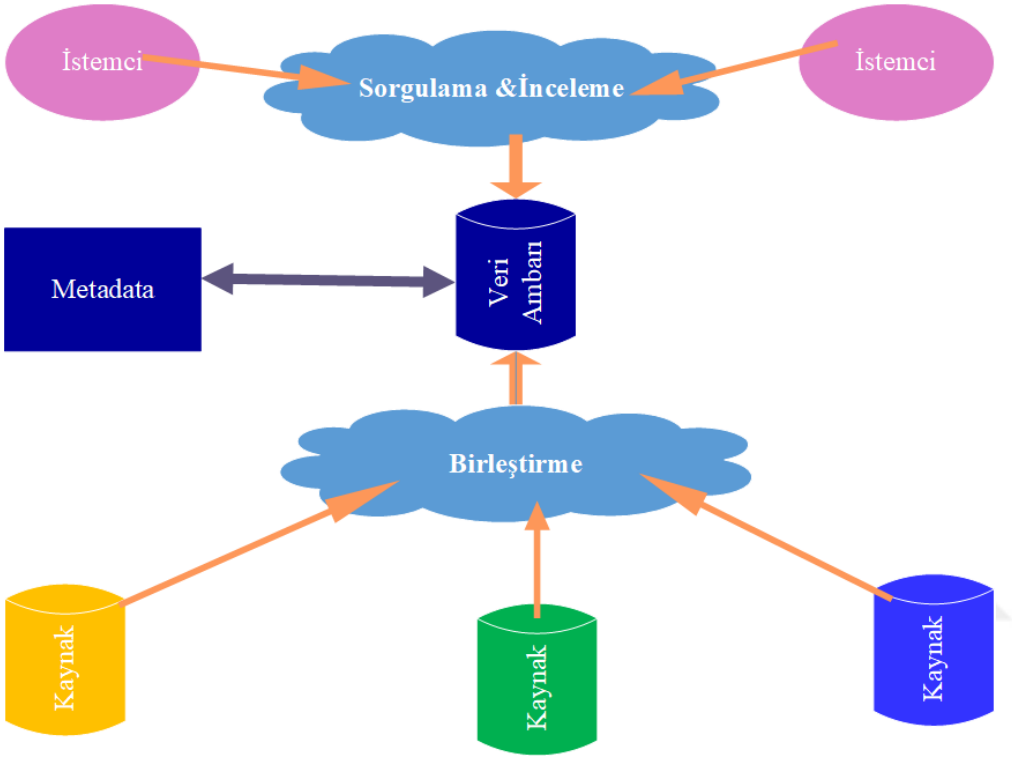
Veri madenciliğinde, istatistikten farklı olarak analiz edilecek veri yığını, büyük ve çok değişken içermektedir. Analiz modelinde kullanılan değişken sayılarının artması durumunda, hesaplama karmaşıklığı ve matematiksel denklemlerin derecesi artmaktadır. Bu da alışılmış istatistik yöntemleri ile kotarılamayacak bir yapı oluşturmaktadır.

Büyük veri kümelerine erişmek, satır, sütün sayısı ve hacim gibi parametre büyüklükleri ve çoğunlukla bu verilerin karışık bir şekilde farklı veri kaynaklarında bulunması nedeniyle zordur. Bunun yanında, büyük veriler, veri toplama süreçlerinin yanında, eksik, kirli ve hatalı veriler içermesi açısından da sorunlar oluşturmaktadır. Veri madenciliği alanında, analize konu verilerin büyüklüğü ve analiz açısından kurulan matematiksel modele göre normalleştirilmemiş ham veriler, alışılmış istatistik tekniklerine göre farklı sorunları gündeme getirmektedir [68], [69].

Veri madenciliğinde, veri analizinde kullanılacak model ve değişkenler belirlendikten sonra, ham veri, modele uygun hale getirilmek üzere, veri temizleme, normalleştirme, eksik veri tamamlama gibi süreçlerden geçirilir. Bunun gibi ön işleme ve veri temizleme süreçleri, analiz için harcanan sürenin %80 kadarını kapsamaktadır [70]. Veri ön işlemede, analizde kullanılmak üzere toplanan veriler, sınıflandırılarak analize uygunluğu belirlenip, veri seçme süreci işletilir. Seçilen veriler, belirlenen model değişkenlerine göre kurulan bir veri yapısında birleştirilir. Birleştirilen veriler eksik veri tamamlama, normalleştirme vb. süreçlerden geçirilerek, analize uygun hale getirilir [71]. Verilerin seçiminde, ham verinin barındırdığı özellikler ve veri büyüklüğünün analize uygunluğu ölçütleri, göz önünde tutulur.

3.1.4. Veri Ambarı

Veri madenciliği uygulamaları için toplanan verilerin, belirli bir biçim kazandırıldıktan sonra depolandığı, özel veritabanlarına veri ambarı denir. Veri ambarlarında, birçok kaynaktan toplanan farklı biçimlerdeki veriler, veri madenciliği uygulamalarının yorumlayabileceği bir şekilde, ortak bir çatı altında birleştirilir ve depolanır. Örnek bir veri ambarı mimarisi aşağıdaki şekilde görülebilir [66].



Şekil 3.2. Örnek Veri Ambarı Mimarisi

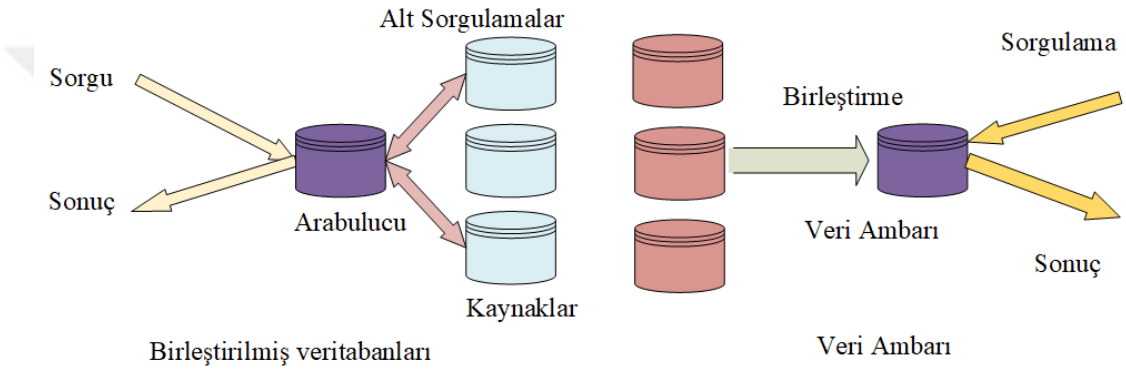
Veri ambarları kurumun stratejik karar süreçlerine önemli bilgi sağlamak üzere, kurumun temel işlem veritabanından ayrı olarak, rapor amaçlı oluşturulan veritabanı yapılarıdır.

Veri ambarları, büyük veri yığınlarından işletmenin durumu ile ilgili özet yönetici raporları sunar. İşletmenin geleceği ile ilgili stratejik kararlara temel oluşturan veri ambarlarının, taşıması gereken özellikler aşağıda listelenmiştir [67]:

- ✓ *Amaca Yönelik:* Analiz yapılacak konuya ilişkin veriler incelenir ve modellenir.

- ✓ *Birleştirilmiş:* Veri ambarlarında farklı kaynaklardan gelen veriler birleştirilerek tutulur. Çeşitli veri temizleme ve birleştirme tekniklerinin kullanımı ile, değişik veri kaynaklarından gelen verilerin tutarlılığı sağlanır.
- ✓ *Zaman Değişkenli:* Veri ambarlarında analiz ve tahminler yapılacağı için, uzun süre ölçekli tarihsel veriler tutulur.
- ✓ *Değişken Değil:* Veri ambarlarında veriler belli aralıklarla farklı bir veritabanına alındığı için, canlı sistemdeki anlık değişimlerden etkilenmez.

Farklı veri kaynaklarını içeren sistemlerden veri çekebilmek için, iki yöntem kullanılabilir. Bunlar, birleştirilmiş veritabanları ve veri ambarıdır.



Şekil 3.3. Farklı Veri Kaynaklarından Veri Çekme Mimarisi

Veritabanlarının birleştirildiği birleştirilmiş veritabanlarında, sorgu alt sorgulara ayrılarak her bir veritabanında çalıştırılır ve sonuçlar birleştirilir.

Veri ambarı çözümünde ise talep ve gereksinimlere göre, veri daha sonra kulanılmak üzere, veri ambarında birleştirilir ve depolanır.

3.1.5. Veri Madenciliği Yöntemleri

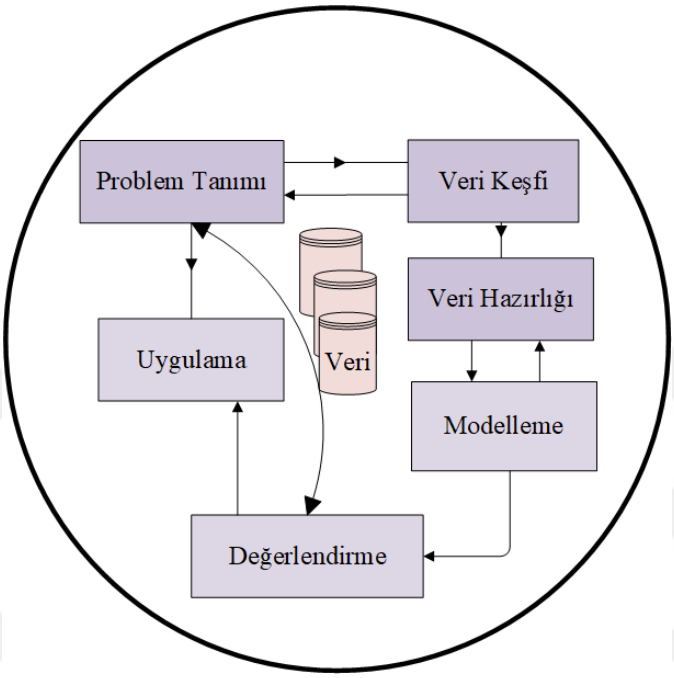
Veri madenciliği sürecinde kullanılan bir endüstri standardı olan CRISP-DM tarafından belirlenen yöntemler, Şekil 3.4'teki grafikte gösterilmiştir.

3.1.5.1. Problem Tanımı

Bu aşamada, iş gereksinimlerine göre proje amaçları belirlenir ve proje amaçlarından problem tanımı belirlenir.

3.1.5.2. Veri Keşfi

Bu aşamada veri toplanır, tarif edilir ve incelenir. Veri kalitesi bu aşamanın önemli etmeni olup, veri kalitesi sorunları bu aşamada belirlenir. Veri keşfi için, betimsel istatistikler gibi geleneksel veri analizleri kullanılır.



Şekil 3.4. CRISP-DM

3.1.5.3. Veri Hazırlama

Bu aşamada analiz modelleme süreci için veri oluşturulur. Veri, çeşitli temizleme ve dönüşümlere tabi tutularak, veri madenciliği araçlarının kullanabileceği biçime çevrilir. Veri hazırlama sürecinin bu aşamasında veri yığınları, veritabanında saklanacak tablo biçimine çevrilmeye çalışılır. Kurulması planlanan model değişkenlerine göre seçilen özelliklerle, kayıtlar oluşturulur ve tablolarda saklanır.

Veri hazırlama süreci model oluşturmada büyük öneme sahiptir. Problemin çözümüne yönelik kurulan modelin eksik, yanlış ve modelle ilişkisi olmayan veriyle çalıştırılması, beklenen sonuçların elde edilememesine neden olacaktır. Bu da sürekli olarak geri dönülerek, verilerin yeniden değerlendirilmesi için çaba harcanmasını gerektirecektir. Veri hazırlama süreci, veri madenciliğinin analistlik aşaması olarak

değerlendirilecek olursa, doğru model, buna bağlı olarak doğru sonuçlar üretebilmek için, bu aşamada tüm çabanın %85'i kadarını harcamak gerekmektedir [14].

3.1.5.4. Modelleme

Problem tanımına göre bilinen matematiksel modeller, verilere uygulanarak çözümler üretilmeye çalışılır. Çözüm için en iyi değerler elde edilene kadar, model parametreleri değiştirilerek, matematiksel model veriye uygulanır. Bu işlemler sonucunda, yüksek kaliteli bir veri modeli kurulmuş olur.

3.1.5.5. Değerlendirme

Veri madenciliği modeli elde edilen sonuç değerlerine göre değerlendirilir. Elde edilen değerler problem için kabul edilebilir değerler değilse, model tekrar değerlendirilir; uygun görülen düzenlemeler yapılır ve veri analiz süreci tekrar işletilir.

3.1.6. Bilgi Keşfi

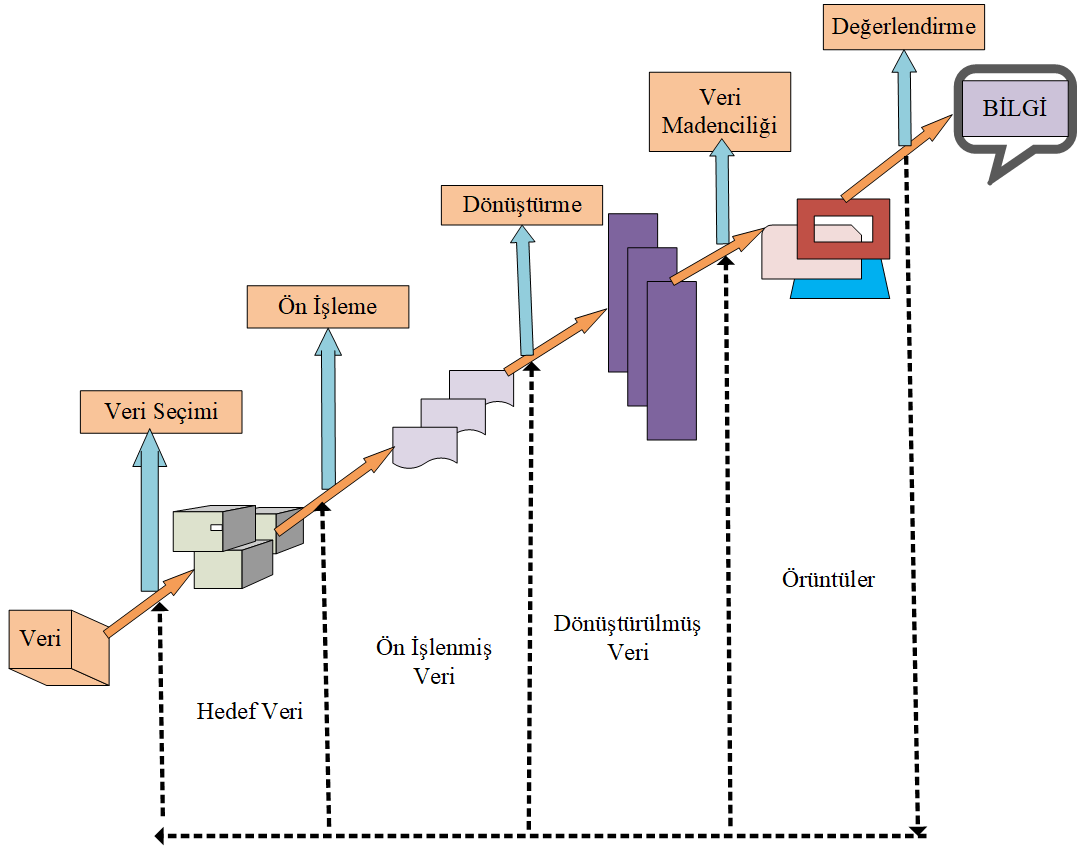
Veri madenciliği, teoride bilgi keşfinin bir aşaması olmasına rağmen, uygulama geliştirmede veri madenciliği ile aynı anlamda kullanılmaktadır. Bilgi keşfi için analiz için belirlenmiş model, veri madenciliği teknikleriyle veriye uygulanır. Böylelikle veri içindeki örüntüler bulunmaya çalışılır.

Veri madenciliği, büyük veri kaynaklarından, gizli, önemli, önceden bilinmeyen, yararlı bilgiyi bulmayı amaçlar.

Veri madenciliğinde bilgi keşfinin aşamaları aşağıda listelenmiştir (bknz. Şekil 3.5):

- ✓ *Veri Temizleme:* Modelle ilişkisi olmayan, verinin genel eğilimleri ile tutarlılık açısından çelişen verilerin ayıklanması sürecidir.
- ✓ *Veri Bütünleştirme:* Farklı veri kaynaklarından sağlanan veriler birleştirilir.
- ✓ *Veri Seçme (İndirgeme):* Analizle ilgili veriler belirlenir ve seçilir. Analize konu verilerin içerdiği genel eğilimlerle uyumlu olmayan, sadece özel olarak gerçekleşmiş, tekrarlanmayan olayları tasvir eden verilerin, model açısından önemi kontrol edildikten sonra, model verilerinden ayıklanır.
- ✓ *Veri Dönüştürme:* Veri, veri madenciliği teknikleri ile analiz edilebilir hale getirilir.

- ✓ *Veri Madenciliği:* Veri madenciliği algoritmaları, veriye uygulanarak, analizler yapılır.
- ✓ *Örüntü Değerlendirme:* Veri analizleri ve ölçümlerle elde edilmiş bilgiyi temsil eden örüntüler geliştirilir.
- ✓ *Sunum ve Değerlendirme:* Veri madenciliği yöntemleri ile elde edilmiş bilginin kullanıcılara sunulmasıdır.



Şekil 3.5. Bilgi Keşfinin Aşamaları

Crisp-DM (bknz. Şekil 3.4) aşamalarında da görüldüğü üzere, veri madenciliği sürecinde model işlemleri başlamadan önce ham veri, veri veri keşfi ve veri hazırlama süreçlerine tabi tutulur. Pyle'ın çalışmasında, veri madenciliği projesi geliştirmeye harcanan toplam zamanın %60'ı veri hazırlamaya ayrılırken, sadece %5'inin modellemeye ayrıldığı iddia edilmektedir [71].

Veri madenciliğinde veri kalitesi, yapılan analizlerin güvenilirliği açısından çok önemlidir. Çünkü hatalı verilerle yapılacak analizler, hatalı sonuçlar üretecektir.

- ✓ Veri ön işleme,

- ✓ Verilerin analizine engel teşkil edecek veri hataların düzeltilmesi,
- ✓ Problemin çözümüne yönelik veri analizi gerçekleştirmek için, verinin içerdiği anlamların belirlenmesi,
- ✓ Analize konu veri yığınının modele uyumlu, model açısından anlamlı bir hale dönüştürülmesi,

amacıyla çalıştırılır.

Veri madenciliği uygulamalarında, çözülecek problem, analiz için toplanan veriler ve kurulan modele göre, çeşitli veri ön işleme tekniklerinin kullanılmasını gerekmektedir. Bu nedenle, modelin anlamlı sonuçlar üretebilmesi için, doğru veri ön işleme tekniklerini kullanmak önem arzetmektedir [72].

CRISP-DM süreçlerindeki, 4 süreçten oluşan veri ön işleme teknikleri,

- ✓ Veri Temizleme
- ✓ Veri Birleştirme
- ✓ Veri Dönüştürme
- ✓ Veri İndirgeme

şeklinde listelenir.

Veri ön işleme tekniklerinin uygulanması sonrasında yapılan analizlerin veri kalitesini artırması, hem analiz sonuçlarının kalitesini, hem de veri madenciliği sürecinde harcanacak zamanı artırmıştır [10].

Genelde veritabanlarında bulunan ham verinin, ön işlemeye tabi tutulmadan önce çeşitli problemleri vardır. Bunlar;

- ✓ Model açısından önem içermeyen alanlar,
- ✓ Eksik değerler,
- ✓ Verinin içerdiği genel eyilimlerle çelişen değerler,
- ✓ Mantıksız değerler,
- ✓ Modelle uyumlu olmayan alanlar, anlamsız değerlerdir.

3.1.6.1. Mantıksız Değerler

Analize tabi tutulacak verilerdeki mantıksız değerlerin, düzeltilerek eksik değerlere dönüştürülmesidir.

3.1.6.2. Eksik Değerler

Kullanıcı veri girişine açık veri tabanları ve bu veri tabanlarındaki verileri kullanarak oluşturulan veri ambarları, eksik, birbirleriyle tutarsız ve gürültülü veriler içerebilir. Literatürde böyle veriler, kirli veriler olarak tabir edilip, veri temizleme süreçlerine tabi tutulması gerekir. Kim vd. nin çalışmasında, kirli verilerin geniş bir taksonomisi hazırlanmıştır [73]. Hatalı veriler nedeni ne olursa olsun veri ön işleme sürecinde düzeltilmelidir [10].

Bir kaydın tüm alanlarının eksik veriden oluşması durumunda, kayıt analiz dışında tutulur. Kaydın bazı alanlarında veri olup, diğer alanlarında verinin eksik olması durumunda, geliştirilecek model ve verilerin içerdiği istatiksel eyilimlere göre, çeşitli eksik veri tamamlama yöntemleriyle veri tamamlanır. Bunlar;

- ✓ Eksik veri yerine sabit bir değerin atanması,
- ✓ Aynı tür verilerin aritmetik ortalaması ile eksik değerin tamamlanması,
- ✓ İstatistiki bir dağılımdan üretilen rastgele bir değerle değiştirilmesi
- ✓ işlemleridir.

Modelin kullandığı analiz algoritmasının doğru ve tutarlı sonuçlar üretebilmesi için, eksik verilerin, model ve mevcut verilerle uyumlu bir şekilde tamamlanması gerekir. Eksik değerlerin, regresyon ve karar ağacı gibi yöntemlerle değiştirilmesi, modele standartlaştırılmış veri sağlayacağı için, daha doğru ve tutarlı analiz sonuçlarının elde edilmesine olanak tanıyacaktır.

3.1.6.3. Anlamsız Değerler

Kullanıcı veri girişiyle veri toplanan sistemlerde, kullanıcının yanlış veri girişi, uygulama hataları vb. nedenlerle veri içerisinde anlamsız değerler oluşabilir. Veri özelliğinin içerdiği değerlerin varyansı hesaplanarak, sapma oranına göre anlamsız, gürültülü değerler belirlenebilir. Kümeleme analizi, regresyon, histogram vb. yöntemler kullanılarak model için, anlamsız veriler belirlenebilir.

3.1.6.4. Veri Birleştirme

Veri madenciliği çeşitli veri kaynaklarından, farklı biçimlerde veriler toplanarak yapılır. Farklı veri kaynaklarından toplanan verilerin, veri madenciliği modeliyle analiz edilebilmesi için, analize uygun olarak oluşturulan bir biçimde, tek bir veri kaynağı olarak birleştirilmesi gereklidir.

3.1.6.5. Veri Dönüşümleri

Değişkenlerin ölçü birimleri arasındaki farklılıklarının giderilmesi amaçlı olarak yapılan dönüşümlerdir. *min-max* ve *z-skor* standartlaştırma algoritmaları, en bilinen iki yöntemdir [74] :

min-max yönteminde veri değerleri mininum 0 ve maksimum 1 olacak şekilde, aşağıdaki eşitlik kullanılarak dönüştürülür:

$$X^* = \frac{X - min(X)}{aralik(X)} = \frac{X - min(X)}{max(X) - min(X)} \tag{3.1}$$

z-skor normalleştirme yönteminde ise, veri 0 aritmetik ortalama, 1 standart sapmaya sahip olacak şekilde dönüştürülür. Bu işlem için veriden ortalamanın çıkarılması ve standart sapmaya bölünmesi yeterlidir.

$$X^* = \frac{X - ortalama(X)}{standartsapma(X)} = \frac{X - \bar{X}_s}{S_x} \tag{3.2}$$

3.1.6.6. Aykırı Değerlerin Tespiti

Aykırı değerler yanlışlıkla tanımlanmış olabilmekle birlikte, veri analizleri üzerinde büyük etkilere sahip olabilecek değerlerdir. Bu değerler standartlaştırma algoritmaları sonucunda elde edilen değerler incelenerek tespit edilebilir. *z-skor* algoritmasına göre +3’ ten büyük veya -3’ ten küçük olarak hesaplanan değerler, aykırı değerler olarak tanımlanır.

3.1.6.7. Veri İndirgeme

Veri indirgeme analizde kullanılacak verinin, içerdiği anlamları kaybetmeden, küçük, özet bir kopyasının oluşturulmasıdır. İndirgenmiş veriler, veri madenciliği modellerinin uygulama hesaplama karmaşıklığını azaltmak, asıl konuya ilişkin

verilerle probleme odaklanmak gibi faydalar sayesinde, veri madenciliği analizlerinin daha iyi sonuçlar üretmesini sağlar.

Veri madenciliğinde aşağıda listelenmiş veri indirgeme yöntemleri kullanılır:

- ✓ Veri birleştirilmesi veya veri küpü oluşturulması
- ✓ Boyut azaltma
- ✓ Verinin sıkıştırılması
- ✓ Verinin kesikli biçime çevrilmesi

3.1.7. Veri Madenciliği Modelleri

Veri madenciliğinde kullanılan model işlevsel olarak üç farklı grupta listelenebilir. Bunlar; [16]

1. sınıflama ve regresyon modelleri,
2. kümeleme modelleri
3. birliktelik kurallarıdır.

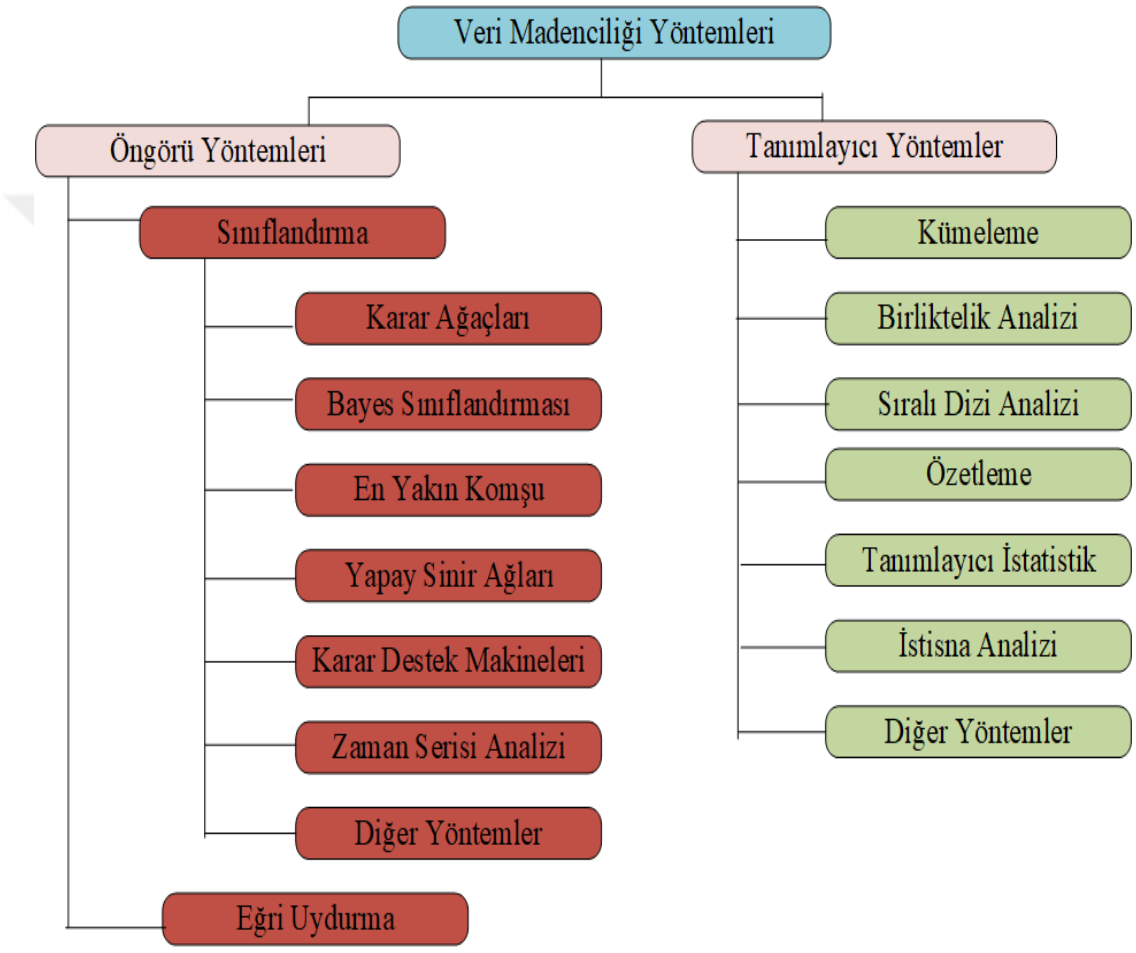
Veri madenciliği tekniklerinde iki farklı öğrenme modeli yaklaşımı kullanılır [14], [75], [76], [77]:

- ✓ Denetimli öğrenme modelinde, gerçek sonuç değeri veri kümesi, öğrenme ve test verisi olarak ikiye bölünür. Giriş ve çıkış değerleri bilinen öğrenme veri kümesiyle, model eğitilir. Modelin eğitimi tamamlandıktan sonra, test veri kümesi ile tahminler üretilir. Test veri kümesi için üretilen tahmin değerleriyle gerçek çıkış değerleri, çeşitli ölçüm metrikleri ile karşılaştırılarak, hata hesaplaması yapılır. Hesaplanan hata değerleri ne kadar küçük olursa, o kadar doğru tahminler üretildiği düşünülür.
- ✓ Denetimsiz öğrenmede, öğrenme sürecine rehberlik edecek sınıf, çıkış değeri bilinen veri olmadığı için, verinin içerisinde örüntüler incelenerek rehberlik edecek bilgi oluşturularak, elde edilmeye çalışılır.

Sınıflama ve regresyon modelleri denetimli öğrenme modelleri, kümeleme ve birliktelik kurallarıysa, denetimsiz modeller olarak tanımlanır [76].

Denetimli modellerde, analiz verileri içerisindeki veri örneklerinin hangi sınıfta olduğu bilinir; bu bilgi öğrenme ve tahmin sürecine rehberlik eder. Denetimsiz öğrenmede ise, böyle bir bilgi bulunmamaktadır. Analizde kullanılan öğrenme algoritması, veri içerisinde örüntü, ilişki ve bağımlılıkları inceleyerek, kendi kendine öğrenme sürecini işletir [77].

Veri madenciliği yöntemlerinin sınıflandırılması, aşağıda şematik olarak gösterilmiştir [67].



Şekil 3.6. Veri Madenciliği Yöntemlerinin Sınıflandırılması

Veri madenciliği yöntemlerinin sınıflandırmasının, denetimli ve denetimsiz model ilişkilendirmesi Tablo 3.1'de verilmiştir.

Tablo 3.1. Denetimli / Denetimsiz Modeller

	Değerlendirme Yöntemi	Denetimli	Denetimsiz
Tahmin	Sınıflandırma	√	
	Regresyon	√	
	Zaman Serisi Analizi	√	
Birliktelik	Bağlantı Analizi		√
	Ardışık Zamanlı		√
Kümeleme	Aykırı Değer Analizi		√

3.1.7.1. Sınıflama ve Regresyon Modelleri

Veri madenciliğinde öğrenme, toplanan verileri bir grupta sınıflayarak yapılmaya çalışılır. Sınıflama ve regresyon algoritmaları denetimli öğrenme modelleri olarak tanımlanır. Denetimli öğrenme modellerinde, veri kümesinin bir kısmi ile eğitilen öğrenme algoritmasından, diğer kısmı ile, sonuç değerini tahmin etmesi beklenir. Regresyon ve karar ağacı algoritmaları, denetimli öğrenme modelleri içinde en önemli algoritmalar olarak bilinmektedir [78], [79].

Sınıflama ve regresyon algoritmaları olarak bilinen önemli yöntemler aşağıda listelenmiştir [75]:

1. Karar Ağaçları
2. Yapay Sinir Ağları
3. K-En Yakın Komşu Algoritması
4. Naive-Bayes Sınıflandırıcı

Karar Ağaçları

Ağaç şeklinde kökten yapraklara kadar kurallar silsilesinden oluşan karar ağaçlarında, öğrenme modelindeki değişkenlerin değerlerine göre, ağaçta dallanarak öğrenme

gerçekleştirilir [80]. Şu özellikler, akış diyagramı yapısındaki karar ağaçlarının güçlü yönleri olarak listelenebilir [79]:

- ✓ Karar ağaçlarında mevcut kurallardan kolaylıkla yeni kurallar türetilebilir.
- ✓ Verileri, kolayca sınıflandırma konusunda yeteneklidir.
- ✓ Kesikli ve sürekli değişkenler içeren modellere uygulanabilir.
- ✓ Model hangi değişkenlerin önemli olduğunu seçebilme yeteneğine sahiptir.

Karar ağaçlarında, düğümler özelliklerin testini gösterirken, düğüm dalları test sonuçlarını, ağaç yaprakları ise ağaçla yapılan sınıflandırmanın etiketlerini gösterir.

Karar ağacı modellerinde, ağacın oluşturulması iki süreç içerir [67]:

1. **Ağaç oluşturma:** Veri örneklerinin özellik değerlerine göre, ağaçta kökten yapraklara dallanmalar ve düğümler oluşturularak ağaç oluşturulur.
2. **Ağaç budama:** Ağaç üzerinde veri madenciliğinin veri temizleme süreci çalıştırılır. Model için önemli olmayan, aykırı ve mantıksız değerler ile bir defaya mahsus gelişen durumları tanımlayan değerler ağaçtan silinir.

Karar ağacı ile sınıflandırmada, örnek veriyi en iyi bölen özellikler belirlenerek sınıflama yapılır. Veriyi en iyi sınıflayan özelliklerin doğru olarak belirlenmesi, modelin doğru kararlar vermesi açısından önemlidir. Örnekleri en iyi bölen özellikler, iyilik fonksiyonu (goodness function) kullanımı ile belirlenebilir.

Veri madenciliği modeli sınıflandırma algoritmalarına göre kullanılan iyilik fonksiyonları aşağıda listelenmiştir:

1. Bilgi kazancı: ID3, C4.5
2. Gini indisi: Her nitelik ikiye bölünür ve tüm olası ikiye bölünmeler sınanır.

Bilgi Kazancı

Bilgi kuramı kavramları kullanılarak, karar ağaçlarının oluşturulduğu yöntemdir. Bütün niteliklerin ayrık değerler aldığı varsayılan yöntem, değişiklik yapılarak sürekli değişkenlere de uygulanabilir. Sınıflandırma işleminde minimum sayıda karşılaştırma yapılmasının hedeflenmesi, sınıflandırmanın önemli özelliklerindendir.

S veri kümesi için, *A* özelliğinin bilgi kazancı,

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (3.3)$$

ifadesiyle hesaplanabilir. Bu ifadede, $Values(A)$, A özelliğinin alabileceği değerler, S_v ise, $A = v$ koşulu için, S' nin alt kümesidir.

Entropi fonksiyonu

Entropi rastgeleliği, belirsizliği ve beklenmeyen durumun ortaya çıkma olasılığını gösterir. Entropinin sınıflandırmada önemi, belirsizliği ölçmesidir. Veri analizinde gerçekleştirilen sınıflandırmanın doğruluk derecesi, entropinin 0'a yakın olmasıyla ölçülür. Sınıflandırmada olayın gerçekleşmesi durumu, entropi değeri 0 olarak tanımlanır. Eşit olasılıklı durumlara sahip sistemler, yüksek belirsizliğe sahiptirler. Entropi bilgi için, rassal bir olayın olması durumunda varolan bilgi ölçütü olarak tanımlanırken, bir süreç için, veri örneklerinin tamamının içerdiği bilginin öngörülen değeri olarak tanımlanır.

Shannon'a göre, bir sistemde gelişen durum değişikliklerinde, entropide meydana gelen değişim, bilgi boyutunu tanımlar [81]. Bu durumda bir sistemin belirsizliği arttıkça, sistem dahilindeki olayların gerçekleşmesi durumunda elde edilecek bilgi de artacaktır. Diğer bir deyişle, risk arttıkça, kazançta artar.

$p_1, p_2, .., p_s$, toplamları 1 olan olasılık değerleri olması koşulu ile, entropi fonksiyonu aşağıdaki eşitlik ile hesaplanır:

$$H(p_1, p_2, .., p_s) = -\sum_{i=1}^{s} p_i log(p_i) \quad (3.4)$$

Gini İndisi Algoritması

Gini algoritmasında, özellik değerleri iki ayrı bölümde hesaplama işlemine tabi tutularak, bölümleme gerçekleştirilir. Sol ve sağ bölümler için ayrı olarak, $Gini_{sol}$ ve $Gini_{sağ}$ değerleri, (3.5) ifadesi ile hesaplanır.

$$Gini_{sol} = 1 - \sum_{i=1}^{k} \left(\frac{|Tsinif_i|}{|B_{sol}|} \right)^2$$
$$Gini_{sag} = 1 - \sum_{i=1}^{k} \left(\frac{|Tsinif_i|}{|B_{sag}|} \right)^2 \quad (3.5)$$

$Tsinif_i$ sol ve sağ bölümdeki sınıf değerlerini, $|B_{sol}|$ sol kısımdaki değerlerin tamamının sayısını, $|B_{sag}|$ sağ kısımdaki değerlerin tamamının sayısını gösterir.

Hesaplamalar sonucunda, bölümler bazında Gini değeri en düşük olan değerler, sonuç olarak seçilir.

Karar Ağacı Algoritmaları

Yapısında karar ağaçlarını kullanan CHAID, QUEST, CRT, SLIQ, ID3 ve C5.0 gibi birçok algoritma vardır. Bu algoritmalardan bazıları ve özellikleri Tablo 3.2'de verilmiştir [82].

Tablo 3.2. Bazı Karar Ağacı Algoritmaları Özellikleri

Karar Ağacı Algoritması	Özellikler
CRT	✓ Gini indisi algoritması ve iki bölümlü işlemi kullanır. ✓ Karar ağacının düğüm bağlı olmayan düğümlerinde iki dal bulunur. ✓ Karar ağacının karmaşıklığına göre budama işlemi yapılır. ✓ Sınıflandırma ve regresyon süreçleriyle uyumludur; bu süreçlere kolaylık sağlar. ✓ Sürekli değişkenler içeren modellere uygulanır. ✓ Ham veri kullanılmaz. Verinin algoritmaya uygun hale getirilmesi için, veri ön işleme süreçlerine ihtiyaç duyar.

C4.5 ve C5.0	✓ Karar ağacı düğümlerden çıkan çok sayıda dallarla oluşturulur. ✓ Ağaçtaki dal sayısı veriyi oluşturan kategorik değişkenlerin sayısına eşittir. ✓ Sınıflandırma modeli, birden fazla karar ağacının birleşiminden oluşur. ✓ Değişkenlerin alt bölümlere ayrılması işlemlerinde bilgi kazancı yöntemi kullanılır. ✓ Karar ağacının yapraklarında, karşılaşılan hata oranına göre budama işlemi yapılır.
CHAID SLIQ	✓ Algoritmadaki bölümleme işlemleri ki-kare testleri ile yapılır. ✓ Ağaçtaki dal sayısı iki ile veriyi oluşturan kategorik değişkenlerin kategori sayısı arasında değerler alır.
SPRINT	✓ Hızlı ve ölçeklenebilir bir sınıflandırma algoritmasıdır. ✓ Algoritma hızlı bir şekilde çalışan ağaç budama işlemine sahiptir. ✓ Büyük veri kümelerinde iyi sonuçlar üretir. ✓ Nitelik değerlerinden en uygun olanı seçilerek, bölme işlemi yapılır. ✓ Algoritmanın bellek kısıtlamalarına rağmen, nitelikler bellekte liste olarak tutulur; hesaplamalar bu veri yapısı üzerinden yapılır.

ID3 Algoritması

Quinlan tarafından sunulan ID3 algoritması kategorik verilerle çalışır [83]. Algoritmada oluşturulan karar ağacı ile çok boyutlu veri, belirlenmiş bir nitelikle

bölümlenir ve her adımda özelliklere göre yapılacak işlemler belirlenir. Karar ağaçları algoritmalarının karmaşıklığı içerdiği düğüm ve yaprak sayısına göre belirlenir. Bu nedenle, ID3 algoritmasında düğüm ve yaprak sayısını azaltacak teknikler kullanılmıştır.

Karar ağaçların bölümlenmenin başlanacağı niteliğin doğru belirlenmesi önemlidir. Bölümlemenin uygun nitelikten başlamaması, karar ağacında çok fazla düğüm ve yaprak olmasına neden olmaktadır.

Dallanma için nitelik belirleme işleminde, aşağıda anlatılan işlem adımları takip edilir [67]:

1. Öncelikle sınıf niteliğinin entropisi hesaplanır.

$$H(T) = -\sum_{i=1}^{s} p_i log(p_i) \tag{3.6}$$

2. Sınıflarla bağıntılı nitelik vektörlerinin entropisi hesaplanır.

$$H(X_k) = -\sum_{i=1}^{n} \frac{|T_i|}{|X_k|} log \frac{|T_i|}{|X_k|}$$
$$H(X,T) = -\sum_{k=1}^{n} \frac{|X_k|}{|X|} H(X_k) \tag{3.7}$$

3. Sınıf niteliği için hesaplanan entropi değeriyle, nitelik vektörlerinin entropisinin farkı alınarak, nitelik bazında kazanç değeri hesaplanır.

$$Kazanç(X,T) = H(T) - H(X,T) \tag{3.8}$$

(3.8) ifadesiyle hesaplanan kazanç değerleri içinden, kazanç değeri en büyük olan vektör, dallanılacak düğüm olarak seçilir.

C4.5 Algoritması

Quinlan'ın geliştirmiş olduğu C4.5 [84] algoritması sayısal değerler alan değişkenlerle karar ağacı oluşturmada kullanılır. ID3 algoritmasının iyileştirilmiş bir sürümü olan C4.5 algoritması, sayısal değerlerin kategorik değişkenlere çevrilmesi açısından farklıdır. Algoritmada aşağıda listeli adımlar takip edilir [84], [67]:

1. Bilgi kazancını maksimum yapacak bir eşik değeri seçilir.
2. Sıralanan değerler ikiye bölünerek, eşik değeri bulunur.
3. $[v_i , v_{i+1}]$ aralığının orta noktası bulunarak, eşik değeri hesaplanır (bknz. (3.9)).

$$\boldsymbol{t_i = \frac{v_i - v_{i+1}}{2}} \tag{3.9}$$

4. Böylelikle eşik değeri orta nokta kabul edilerek, nitelik değerleri iki kategoriye bölünmüş olur.

Twoing Algoritması

1984'te Breiman tarafından bulunan, düğümlerde sadece ikili dallanmaların olduğu CART algoritmasında düğümde dallanmalar, belirli bir ölçüte göre gerçekleştirilir [85]. Bu ölçüt tüm niteliklerin değerleri ve eşleşmeleri göz önüne alınarak belirlenir. Bölünme işlemi sonucunda iki bölüm elde edilir ve bu algoritma ile seçme işlemi yapılır [67]. Twoing algoritmasının adımları aşağıda listelenmiştir:

1. Algoritmada eğitim veri kümesi, her bir işlem adımında sağ ve sol olmak üzere iki ayrı dala bölünür.
2. Sağ ve sol dallar için ayrı ayrı, ilgili sütundaki nitelik değeri yineleme sayısı hesaplanır.
3. Sınıf değerleri meydana gelme olasılığı, sağ ve sol bölümdeki nitelik değerleri için hesaplanır.
4. Düğümlerdeki dallanmalar için uygunluk değeri hesaplanır; nitelik değerine en büyük değer atanır.

$$\Phi(B|d) = 2\frac{|B_{sol}|}{|T|}\frac{|B_{sag}|}{T}\sum_{j=1}^{n}\frac{\left|\left|T_{sinif_j}\right|\right|}{|B_{sol}|} - \frac{\left|T_{sinif_j}\right|}{T\left|B_{sağl}\right|} \tag{3.10}$$

(3.10) ifadesinde, T eğitim kümesi kayıt sayısı, B aday bölünme sayısı, d düğümü, T_{sinif_j}, j.sınıf için hesaplanan değeri, $\Phi(B|d)$ ise hesaplanacak uygunluk değerini gösterir.

K-En Yakın Komşu (K-Nearest Neighbour) Algoritması

Denetimli öğrenme modeli algoritması olarak bilinen k-en yakın komşu algoritmasında, sınıfı yani sonuç değeri bilinen veri kümeleri ile öğrenme süreci işletildikten sonra, modele yeni eklenen gözlem değerleri için sınıf değeri tahmin edilir.

Bu sınıflandırma tekniğinde, veri kümesine eklenen gözlem kayıtlarının birbirine uzaklıkları hesaplanarak, birbirine en yakın k sayıda gözlem kaydı birbirine komşu olarak seçilip aynı grupta sınıflanır.

Gözlem kayıtlarının birbirlerine komşuluklarını hesaplamada kullanılan en bilinen uzaklık hesaplama yöntemi, Öklid uzaklık yönteminin ifadesi (3.11)' de görülmektedir.

$$\Phi d(i,j) = \sqrt{\sum_{k=1}^{p} \left(x_{ik} - x_{jk}\right)^2} \tag{3.11}$$

Uzaklık hesaplamada Öklid uzaklığından farklı olarak Hamming [86] ve Minkowski [87] uzaklıkları da kullanılabilir.

K-en yakın komşu algoritmasının adımları aşağıda listelenmiştir:

1. Öncelikle, algoritmada bir sınıfı oluşturacak en yakın komşu sayısı (k), parametresi seçilir.
2. Referans alınacak bir noktaya göre en yakın komşulukların bulunulacağı algoritmada, uzaklık fonksiyonları kullanılır. Veri kümesi kayıtları ile referans nokta arasındaki uzaklık hesaplanır; uzaklık olarak en yakın olanlar, komşu olarak aynı sınıfda gruplanır.
3. Uzaklık fonksiyonu ile hesaplanan değerler, küçükten büyüğe sıralanır ve en küçük k değer çözüm olarak seçilir.
4. Çözüm olarak seçilen değerlerin sınıfı belirlenerek, en fazla yinelenen sınıf seçilir.
5. Belirlenen sınıf, tahmin için kullanılan veri gözlem örneğinin tahmini sınıf değeri olarak seçilir.

Bayes Sınıflandırıcılar

Bayes teoremini kullanarak yapılan istatistiki tahminlerle, veri örneğinin bir sınıfa üye olma olasılığını tahmin eden bir sınıflandırıcıdır. Naïve Bayesian olasılık teorisini temel alması, veri kümesindeki örnek olayların birlikte gerçekleşme olasılıklarını hesaplaması özellikleri sayesinde, başarılı sonuçlar üreten bir sınıflandırıcıdır [77].

Bayes kuralı,

$p(x|Cj)$: j sınıfına ait bir veri örneğinin x olma olasılığı,

$P(Cj)$: j sınıfınının oluşma olasılığı,

$p(x)$: Veri kümesinden seçilen bir örneğin x olma olasılığı,

$P(Cj|x)$: Veri kümesinden seçilen x verisinin j sınıfına ait olma olasılığı,

olması koşuluyla, (3.12) ifadesi ile hesaplanır.

$$P(C_j \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid C_j)P(C_j)}{p(\mathbf{x})} = \frac{p(\mathbf{x} \mid C_j)P(C_j)}{\sum_k p(\mathbf{x} \mid C_k)P(C_k)} \tag{3.12}$$

Naïve Bayes sınıflandırıcısının işlem adımları aşağıda listelenmiştir:

- ✓ Öğrenme veri kümesi T 'nin bütün elamanlarının n boyutlu veri uzayında tanımlı olması koşuluyla, $X = (x_1, x_2, \dots, x_n)$ olarak ifade edilsin.
- ✓ Veri kümesinde m adet sınıf olsun ve $C_1, C_2, \dots, C_m$ olarak ifade edilsin.
- ✓ Sınıflamada son olasılığı en büyük olan değerler aranır ($max((C_i|X))$).
- ✓ Bayes teoreminden $P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$ ifadesi türetilir.
- ✓ $P(X)$ olasılığının, veri modelindeki tüm sınıflarda sabit bir değere sahip olması nedeniyle, $P(C_i \mid X) = P(X \mid C_i)P(C_i)$ ifadesi için, maksimum olasılık değeri bulunmaya çalışılır.

Sınıflandırma Başarım Değerlendirmesi

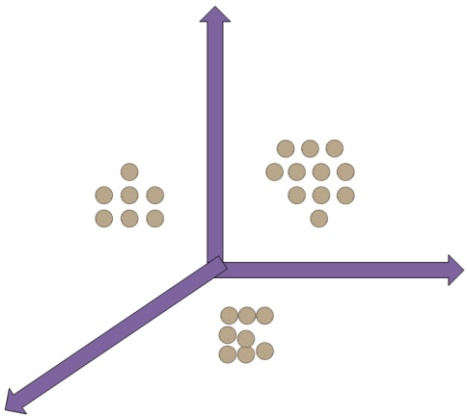
Sınıflandırma teknikleri ile gerçekleştirilen sınıflandırmaların başarımı aşağıda listeli yöntemler ile ölçülebilir [67]:

1. Hız
    a. Model oluşturmaya harcanan süre
    b. Sınıflandırma işlemi için harcanan süre
2. Kararlılık
    a. Analizde kullanılan veri kümesinin, modele aykırı, mantıksız, eksik değerler içermesi durumunda da tutarlı, iyi sonuçlar üretmesidir.
3. Ölçeklenebilirlik
    a. Veri kümesinin büyüklüğünde artışlara cevap verme yetkinliğiyle, büyük boyutlu veri kümelerinde iyi sonuçlar üretmeye devam etmesidir.
4. Anlaşılabilirlik
    a. Sınıflandırma modelinin kullanıcının yorumlayabileceği sonuçlar üretmesidir.
5. Kural bağımlılığı
    a. Birbirine benzer olmayan, ilişkisiz kurallar

3.1.7.2. Kümeleme Modelleri

Denetimsiz bir öğrenme modeli olan kümeleme modellerinde, veri kümesi, birbirleriyle benzer özellikler taşıyan veri elamanlarından oluşur. Aynı kümede olan veri elamanları birbirine yüksek oranda benzerlik gösterirken, farklı kümedeki veri elamanları çok az benzerlik gösterir [77].

Kümeleme modelllerinde Şekil 3.7. de görüldüğü gibi, veri elamanlarının birbirine benzerliklerine göre, birbirine benzemeyen kümelerde gruplanması amaçlanmaktadır [77].



Şekil 3.7. Kümeleme Modelleri

Veri kümesini oluşturan veri elemanlarının birbirlerine benzerlikleri uzaklık fonksiyonları kullanılarak hesaplanabilir. Uzaklık fonksiyonu değerine göre birbirine yakın elemanlar aynı kümede gruplanırken, uzak elemanları farklı kümelerde gruplanır. Kümeleme algoritmalarında kullanılan, bilinen uzaklık hesaplama yöntemleri şunlardır:

- ✓ Manhattan Uzaklığı: p boyutlu uzayda seçilen i ve j noktaları arasındaki uzaklık,

$$\boldsymbol{d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots + |x_{ip} - x_{jp}|} \quad (3.13)$$

ifadesi kullanılarak hesaplanır.

- ✓ Öklid Uzaklığı :p boyutlu uzayda seçilen i ve j noktaları arasındaki uzaklık,

$$(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots + |x_{ip} - x_{jp}|^2} \quad (3.14)$$

ifadesi kullanılarak hesaplanır.

- ✓ Minkowski Uzaklığı: p boyutlu uzayda seçilen i ve j noktaları arasındaki uzaklık,

$$d(i,j) = \sqrt[q]{|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \cdots + |x_{ip} - x_{jp}|^q} \quad (3.15)$$

ifadesi kullanılarak hesaplanır.

Kümeleme modellerinde, veri elamanlarının aynı kümede gruplanmasında kullanılan ikili benzerlik ve uzaklık ölçüsü yöntemi aşağıda anlatılmıştır:

İkili Benzerlik ve Uzaklık Ölçüsü

Analizde kullanılan verilerin, iki sınıftan oluşan kategorik değişkenler içermesi durumunda kullanılan ölçüm yöntemidir. Genellikle 0 ve 1 değerlerinden oluşan bu veri değişkenlerinde, 0 aranan özelliğin olmamasını gösterirken, 1 aranan özelliğin olduğunu gösterir.

Bu yöntemde, Tablo 3.3'te gösterimi verilen, 2*2 boyutunda olasılık tablosu kullanılarak, benzerlik ve uzaklık hesaplamaları yapılır [77], [88].

Tablo 3.3. Binary Benzerlik Ölçüsü

		j Örneği	
i Örneği		0	1
	0	a	b
	1	c	d

a : i ve j örneğinde 0 değerini alan özellik sayısı

b : i örneğinde 0, j' de 1 değerini alan özellik sayısı

c : i örneğinde 1, j 'de 0 değerini alan özellik sayısı

d : i ve j örneğinde 1 değerini alan özellik sayısı

Tablo 3.3' teki değerler kullanılarak aşağıda verilen yöntemlerle, benzerlik ölçüsü hesabı yapılır**.**

Basit Eşleşme Katsayısı: Modelde kullanılan ikili değişkenin simetrik olması durumunda,

$$sim(i,j) = \frac{a+d}{a+b+c+d} \tag{3.16}$$

ifadesi kullanılır.

Jaccard katsayısı: Modelde kullanılan ikili değişkenin asimetrik olması durumunda,

$$sim_{Jaccard}(i,j) = \frac{d}{b+c+d} \tag{3.17}$$

ifadesi kullanılır.

K-Ortalamalar Kümeleme

Analiz veri kümesini, bilinen K sayıda kümeye, istatistiki yöntemleri kullanarak ayıran, basit ve verimli bir kümeleme yöntemidir. Belirli sayıda kümeyle, ortalama hata metriğini minimum yapmak amacıyla, verinin kümelerine ayrılması algoritması aşağıda verilmiştir.

- ✓ N boyutlu uzayda, N örnekli kümelerin verildiği varsayılsın. Bu uzayın $\{c_1, c_2, \dots, c_k\}$ şeklinde K kümeye ayrıldığı düşünülsün. Bu durumda, $\sum n_k = N$ $(k = 1,2, \dots, k)$ olması koşuluyla, C_k kümesi ortalama vektörü M_k, aşağıdaki gibi hesaplanır.

$$M_k = \frac{1}{n_k} \sum_{i=1}^{n_k} X_{ik} c \tag{3.18}$$

- ✓ Burada X_k değerinin, C_k kümesinin bir elamanı olması koşuluyla, C_k kümesindeki karesel hata, C_k örnek veri elamanları ile veri kümesinin merkez noktası arasındaki, Öklid uzaklıklarının toplamı olarak hesaplanır. Aşağıdaki ifade ile hesaplanan hata terimi, küme içi değişim olarak da isimlendirilir:

$$e_i^2 = \sum_{i=1}^{n_k} (X_{ik} - M_k)^2 \tag{3.19}$$

- ✓ Kümeleme algoritmasında toplam karesel hata, tüm veri kümesi içerisindeki küme içi değişimlerin kareleri toplamı alınarak, (3.20) ifadesindeki gibi hesaplanır.

$$E_k^2 = \sum_{k=1}^{K} e_k^2 \tag{3.20}$$

K-ortalamalar kümeleme algoritmasında, K sayıda küme kullanılarak, E_k^2 hata terimini minimun yapacak şekilde, veri kümesinin kümelerine ayrılması amaçlanır. Bu ifadeden hareketle, E_k^2 hata teriminin algoritmanın döngüleri boyunca azalacağı beklenir. K-ortalamalar kümeleme algoritması işlem adımları aşağıda listelenmiştir:

- ✓ Kümeleme işlemlerinde verilerin ayrılacağı, k küme sayısı belirlenir.
- ✓ Analizde kullanılacak veri içerisinden rassal olarak, k merkez noktası belirlenir.

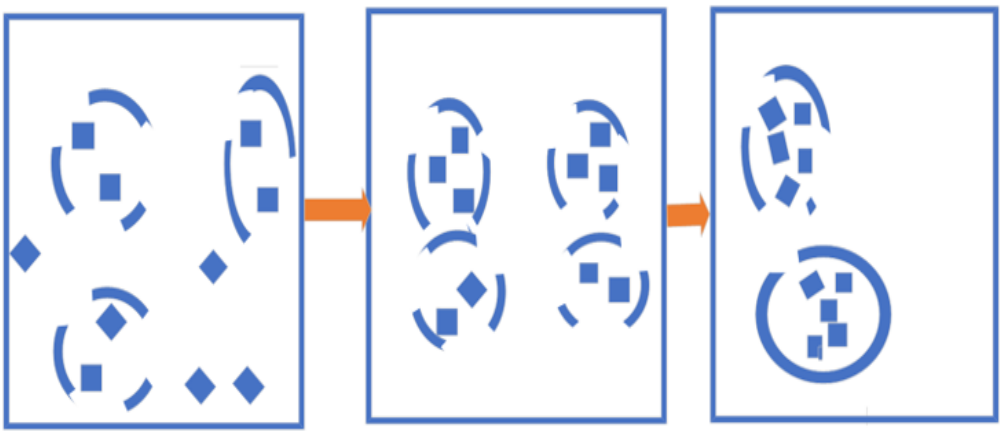
- ✓ Veri kümesi elemanlarıyla, seçilen merkez noktaları arasındaki Öklid uzaklıkları hesaplanır.
- ✓ Yeni veri elamanlarıyla, veri kümeleri için merkez noktaları hesaplanır.
- ✓ Hesaplanan veri kümesi merkez noktalarının önceki merkez noktaları ile aynı olması durumunda algoritma sonlanır. Aksi durumda 3. adımdan çalışmaya devam eder.

Hiyerarşik Kümeleme

Hiyerarşik kümeleme yönteminde, k-ortalamalar kümeleme yönteminden farklı olarak, küme sayısının belli olmasına ihtiyaç duyulmaz; fakat algoritma bir durma koşuluna gereksinim duyar.

1990 yılında Kaufmann ve Rousseeuw tarafından bulunan, Şekil 3.8' de şematik gösterimi yapılmış olan, hiyerarşik kümeleme yönteminin işlem adımları aşağıda listelenmiştir [89]:

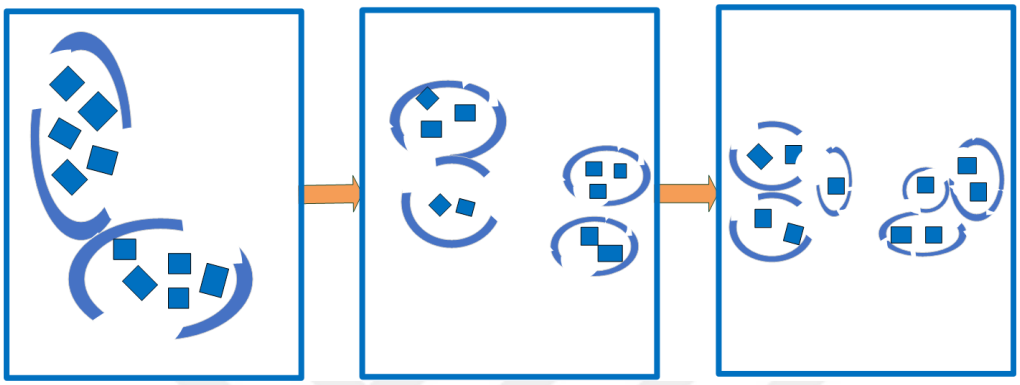
- ✓ Algoritma veri elamanlarının her birini, bir küme kabul ederek çalışmaya başlar.
- ✓ Kümeler arasındaki uzaklıklar hesaplanır; birbirine en yakın olan kümeler birleştirilir.
- ✓ Kümelerin birbirlerine uzaklıkları, tekil bağlantı yöntemi kullanılarak hesaplanır.
- ✓ Veri elamanlarının tümü tek bir kümede toplanana kadar, işlem çalışmayı sürdürür.



Şekil 3.8. Hiyerarşik Kümeleme Modeli (AGNES)

Hiyerarşik Kümeleme: DIANA

İlk defa Kaufmann ve Rousseeuw (1990)' in duyurduğu DIANA algoritması, AGNES algoritmasınına göre ters yönde çalışır. Veri elamanlarının tümü tek bir kümede toplanmış şekilde çalışmaya başlayan algoritma, tüm veri elamanları ayrı tek bir elamanlı kümeye yerleştirilene kadar devam eder [89]. Algoritmanın şematik gösterimi Şekil 3.9.da verilmiştir.



Şekil 3.9. Hiyerarşik Kümeleme Modeli

3.1.7.3. Birliktelik Modelleri

Birliktelik modelleri birden fazla olayın, birbiriyle ilişkili olarak, eş zamanlı gerçekleşmelerini analiz etmeye çalışır. Birliktelik modelleriyle veri madenciliğinde, çeşitli bilgi alanlarının analizi önem arzeder. Örneğin müşterilerin alışveriş alışkanlıklarını belirleyerek, hangi ürünleri birlikte satın almayı tercih ettikleri belirlenebilir ve satış rakamlarının artması sağlanabilir. Birliktelik kuralları ve ardışık zamanlı örüntüler, veri madenciliğini temel alan pazar sepeti analizlerinde, müşterilerin alışverişlerinde ürün tercihi alışkanlıklarını belirlemede kullanılır [14].

Eş zamanlı gerçekleşen, birbirleriyle ilişkili olayların tanımlanmasında kullanılan birliktelik modellerine, bazı örnekler aşağıda listelenmiştir:

1. Marketten kola alan bir müşterinin, %75 olasılık ile patates cipsi alması.
2. Yağsız yoğurt ve az yağlı peynir satın alan müşterinin, %85 olasılıkla %0 yağlı süt alması gibi.

Ardışık zamanlı örüntüler, birliktelik kurallarındaki gibi aralarında bir ilişki olan olayları incelerler; fakat olaylar eşzamanlı olarak gerçekleşmez, ardışık olarak

gerçekleşir. Sonraki olayın oluşma olasılığı, önceki olayın oluşmasına bağlıdır. Ardışık zamanlı örüntülerle ilgili bazı örnekler aşağıda görülebilir.

1. X ameliyatı geçiren bir hasta, 15 güne kadar, %45 olasılıkla Y enfeksiyonuna yakalanabilir.
2. IMKB endeksinde düşüş olması durumunda, A hisse senedinde %15'in üzerinde bir artış olursa, 3 işgününe kadar B hisse senedinde de %60 olasılıkla artış olacaktır.

Örneklerden de görüldüğü gibi olaylar birbirine takip eder; ardışık olarak gerçekleşir.

3.1.7.4. Model Hata Değerlendirme

Denetimli öğrenme modellerinde ham verinin veri ön işleme süreçleriyle işlenmesiyle elde edilen veri, öğrenme ve test verisi olarak iki kümeye ayrılır. Öğrenme veri kümesi ile eğitilen algoritmadan, test verileri ile tahminler üretmesi beklenir. Denetimli öğrenme modellerinde kullanılan test verisinde, giriş değerlerine karşılık gerçek sonuç değeri bilinmektedir. Öğrenme algoritması ile tahmin edilen sonuç değeri, çeşitli ölçüm metrikleri kullanılarak gerçek sonuç değeri ile karşılaştırılır ve modelin doğruluk derecesi belirlenir.

Bir öğrenme modelinin doğruluk derecesinin belirlenmesinde kullanılabilecek en ilkel yöntem, basit geçerleme yöntemidir. Veri kümesinin %5 ila % 33' ünün test verisi, diğer kısmının öğrenme verisi olarak ayrıldığı bir denetimli öğrenme modelinde, öğrenme verisi ile eğitim gerçekleştirildikten sonra, test verileriyle tahminler üretilir. Test verileriyle elde edilen tahmin sonuçları, gerçek sonuçlar karşılaştırılarak, doğru ve yanlış sınıflandırılan olaylar belirlenir. Bu parametreler kullanılarak hata oranı, yanlış tahmin edilen olayların sayısı, bütün olayların sayısına, doğruluk oranı ise yanlış tahmin edilen olayların, bütün olayların sayısına bölünerek hesaplanır. Tahmin edilen değerin doğru veya yanlış olmak üzere, iki durumunun olmasından hareketle, olasılık teorisinin gereği olarak, doğruluk ve hata oranının toplamı 1 olur. Bu durumu açıklayan ifade (3.21)' de verilmiştir.

$$Doğruluk\ Oranı\ =\ 1\ -\ Hata\ Oranı \tag{3.21}$$

Veri miktarının sınırlı sayıda olması durumunda, çapraz geçerleme yöntemleri daha iyi sonuçlar verebilir. Bu yöntemde veri kümesi rassal olarak, a ve b olmak üzere, eşit

sayıda elamanlı iki veri kümesine bölünür. Öncelikle a öğrenme veri kümesi ile eğitilen öğrenme modelinin, b veri kümesi ile tahmin üretmesi beklenir. İkinci aşamada öğrenme ve test veri kümeleri yer değiştirilerek, aynı işlemler tekrar edilir. Her iki aşamada hesaplananan hata değerlerinin ortalaması alınarak hata oranı hesaplanır.

Birkaç bin satır civarı veriden oluşan veri kümelerinde, veri kümesi n gruba bölünebilirse, n katlı çapraz geçerleme yöntemiyle model değerlendirme gerçekleştirilebilir. n katlı çapraz geçerleme yönteminde veri, n veri kümesine bölünür. n veri kümesinden biri test diğerleri öğrenme kümesi seçilerek, denetimli öğrenme süreçleri işletilir. Bu süreç, her aşama diğer veri kümelerinden biri test veri kümesi seçilerek, n aşama devam eder. Her bir aşamada hesaplanan hata oranlarının ortalaması alınarak nihai hata oranı hesaplanır.

Bootstrapping veri kümesi boyutunun küçük olduğu durumlarda tercih edilir. Bu yöntem eğitim veri kümesi, veri kümesi içerisinden tek tek rastgele olarak seçilerek belirlenir. Rassal olarak seçilen veri, her defasında asıl veri kümesine tekrar eklenir. Bu nedenle seçilen bir verinin tekrar seçilebilme durumu olduğu gibi, hiç seçilmeyedebilir. Eğitim veri kümesi ile öğrenme süreci işletildikten sonra, kalan veriler de test kümesini oluşturur. Modelin doğru sonuçlar üretebilmesi için eğitim ve test veri kümesi olarak seçilen örneklemlerin, modeli iyi tarif etmesi gerekir. Modelde n adet örneklem için n kez çalışarak örnekleme işlemi yapılır ve herbir örneklem için öğrenme ve test aşamaları çalıştırılır. Hesaplanan hata oranlarının ortalaması hesaplanarak, nihai hata oranı belirlenir.

3.2. Metin Madenciliği

3.2.1. Metin İşleme

Metin madenciliği, analiz edilecek verinin metin biçiminde olduğu veri madenciliği yöntemidir. Metin veri madenciliği yönteminde, metin dilbilimsel amaçlarla veya özet, önemli ve anahtar veri çıkarımı amaçlı olarak kullanılır.

Metin madenciliğinde yapısal olmayan veriden önceden bilinmeyen, analiz açısından kullanışlı çıkarımlar yapılmaya çalışılır. Elde edilen bilgilerden, metinden çıkarım yapılacak gizli ilişkiler, varsayımlar ve yönelimler olduğu görülür [90], [91].

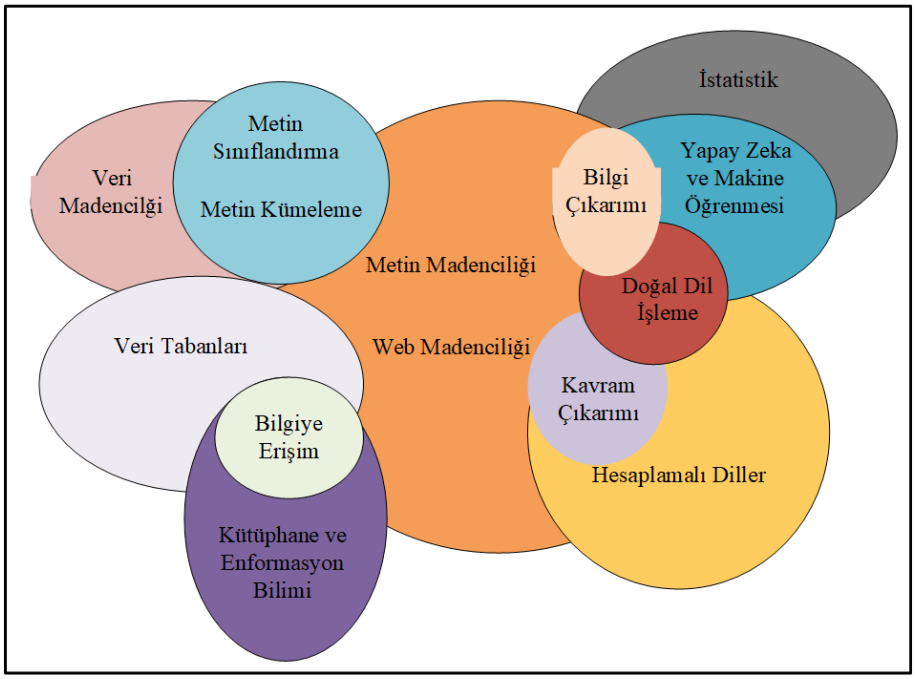
Metin madenciliği, metin biçimindeki veri kaynaklarından bilgi keşfetme olarak da tanımlanabilir [92].

Metin madenciliği teknikleriyle, iki farklı belgenin birbirleriyle benzer olup olmadığına, karar verilebilir. Bu işlem frekans analizi ile yapılabilir. Belgelerin içerisindeki kelimelerin tekrar sayıları birbirleriyle karşılaştırılarak, iki belgenin birbirleriyle benzerliğine karar verilebilir [93].

Metin madenciliği klasik veri madenciliğinin bir parçası olarak düşünülse de, veri kaynakları açısından ciddi bir fark oluşturmaktadır. Metin madenciliğinde dilbilimsel analizler, doğal dil metinleri üzerinde kullanılmaktadır [92].

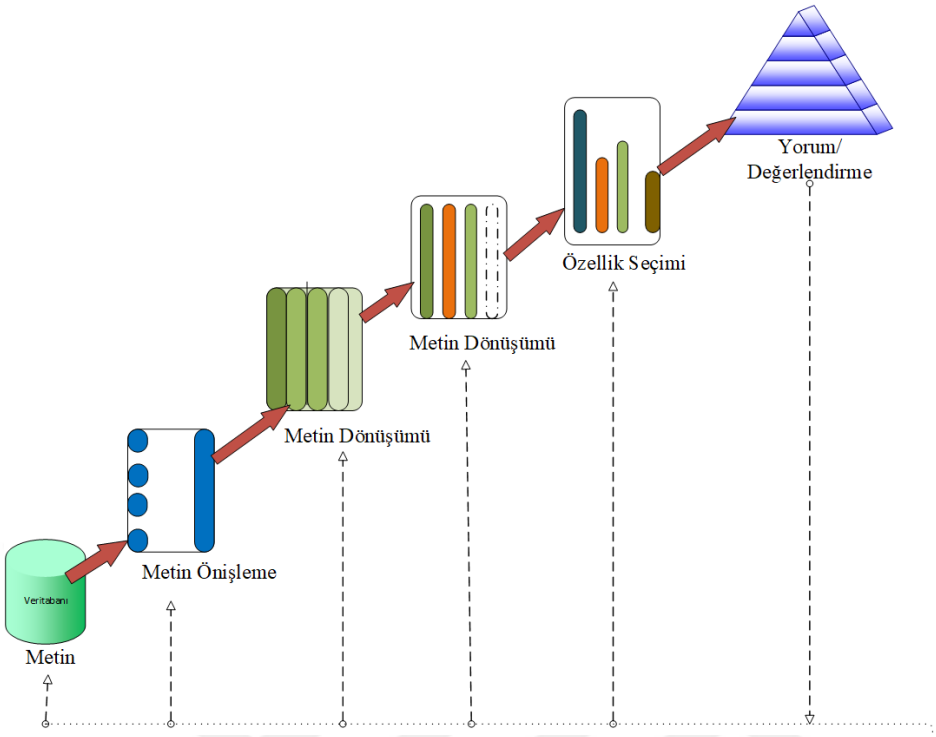
Metin madenciliğinde veri çıkarımı, tokenlaştırma, frekans analizi, duygu analizleri, dilbilimsel, fonetik ve morfolojik analizler, örüntü tanıma vb. analizlerle yapılmaya çalışılır. Metin madenciliği basit anlamda, metinleri sayısallaştırarak, örüntüler çıkarmaya çalışır. İleri seviyede düşünüldüğünde ise metinleri özetleyerek bilgi çıkarımında bulunmaya çalışır.

Miner vd. (2012)' nin çalışmasında, metin madenciliğinin, veri madenciliği, doğal dil işleme, makine öğrenmesi, yapay zeka, istatistik, hesaplamalı diller, kümeleme, sınıflandırma vb. disiplinler ve yöntemler ile ilişkili olduğu sunulmaktadır [94].



Şekil 3.10. Metin Madenciliği İlişkili Olduğu Yöntem ve Disiplinler

Genel olarak, klasik bir metin madenciliği çalışması Şekil 3.11'de özetlenmiştir [95].



Şekil 3.11. Metin Madenciliği Çalışması

Zohar (2002)' ye göre metin madenciliği, dört ana grupta sınıflandırılmıştır [96]:

1. Bilgi Erişimi
2. Bilgi Çıkarımı
3. Web Madenciliği
4. Kümeleme

3.2.1.1. Bilgi Erişimi

Vickery [97], Mooers [98] bilgiye erişimi, bilgilerin tutulduğu bir veri kaynağından getirilmesi olarak tanımlar.

Metinsel bir veri kaynağında bilgiyi bulmak için, konusal bir arama yapılır. Buradaki sorun, kullanıcının ne gibi bilgilerle ilgilendiği ve kullanıcının yaptığı aramaya karşılık ne gibi belgeler sunmak gerektiğidir [99].

Can vd. (2008) çalışmasına göre bilgi ulaşmada iki önemli ölçüt vardır [100]:

1. ***Doğruluk:*** Kullanıcının aradığı bilgi ile ilgili yaptığı sorguya sonuç olarak dönen belgelerin içerisinde, aradığı konu ile ilgili belgelerin sayısının, tüm ilgili belgelerin sayısına oranı, doğruluk oranı olarak gösterilir [100], [101].

$$Doğruluk = \frac{Konuyla\ ilgili\ belge\ sayısı}{Tüm\ ilgili\ belgelerin\ sayısı} \tag{3.22}$$

2. ***Duyarlılık:*** Kullanıcının yaptığı sorgu sonucunda dönen belgeler içerisindeki konuyla ilgili belgelerin, dönen tüm belgelere oranı duyarlılık olarak isimlendirilir. Duyarlılık yapılan çalışmada hızı belirleyen önemli bir etkendir. İlgili belge bulma oranı düşük olduğu sürece, yeterli sayıda ilgili belge bulma süresi uzayacağı için çalışmanın hızı da yavaşlayacaktır [100], [101].

$$Duyarlılık = \frac{Dönen\ \ ilgili\ belge\ sayısı}{Dönen\ belgelerin\ sayısı} \tag{3.23}$$

Bilgi çıkarımında elde edilen bilginin belge içerisinde önem haiz edip etmediği, konuyla ilgili olup olmadığını anlamak için ağırlık (w) verme modelleri geliştirilmiştir. Yerel ağırlıklandırma işleminde terim frekansı kullanılırken, genel ağırlıklandırma işlemindeyse ters belge frekansı kullanılır [100], [101].

Terim frekansı (tf) bir terimin belge içerisindeki tekrarlanma sıklığıdır. Ters belge frekansı (bf) ise terimin belge koleksiyonu (B) içerisindeki önemini gösterir.

Bu tanımlardan hareketle, bilginin önemi belge içerisinde terimin sık gözükmesine bağlıyken, belge kümesi içerisinde az gözükmesine bağlıdır. Bu durumda ağırlık fonksiyonu aşağıdaki ifadedeki gibi hesaplanır.

$$w_i = tf_i log \frac{B}{bf} \tag{3.24}$$

Frekans değeri küçük olan terimlerin ters belge frekansı büyük değerler alırken, büyük olanların ise küçük olur.

TF-IDF, bir terim az sayıda belgede çok sıklıkla görülüyorsa büyük, tüm belgelerde az çok sayıda yer alıyorsa küçük değer alır [100], [101], [102].

3.2.1.2. Bilgi Çıkarımı

Birçok veri kaynağından elde edilen belge yığınından, özet bilgilere erişilmesine bilgi çıkarımı denir. Web sayfalarından bilgiler karşılaştırılarak, geniş ölçekli belgelerden bilgi çıkarımı yapılabilir [103].

Bilgi çıkarımı yöntemi, belge içerisindeki unsur ve varlıkları çıkarır; aralarındaki ilişkileri belirler. Bilgi çıkarımı teknikleri, analiz edilen metindeki varlıklar hakkında bilgi veren önermeleri kullanarak, metnin varlıkları ve aralarındaki ilişkiler hakkında bilgi çıkarımda bulunurlar [104], [105].

Bilgi çıkarımı yöntemleri metnin konusuyla ilgili terimler ve ilişkileri ön plana çıkarması açısından önemlidir. Bu yöntemlerle metin analizlerinde amacın bilinmeyen terimler ve ilişkileri ortaya çıkarnak olması durumunda, bilgi keşfi yöntemlerinden faydalanılır. Bilgi keşfi yöntemleri metinden elde ettiği bilgileri diğer kaynaklarla pekiştirir. Bu yöntemde terim ve terimler arası ilişkilerden daha ileri gidilerek özel yapılar ve fonksiyonlarla, birbirlerine bağlılık gösteren bir bağıntı kümesi oluşturulur. Bu tür sistemlerde metinsel verilerle birlikte yapısal verilerden de faydalanılabilir [106].

Bilgi çıkarım sistemlerinde elde edilen verinin değerlendirilmesi bilgi erişim sistemlerinde olduğu gibi, doğruluk ve duyarlılık ölçütleriyle yapılır. Bilgi çıkarımında tahminler, ölçüm ölçütü olarak kullanılır. Duyarlılık, doğru tahminlerin tüm tahminlere oranıyla hesaplanırken, doğruluk ise doğru tahminlerin, bulunan varlık sayısına bölünmesiyle hesaplanır [107], [108].

3.2.2. Metin Madenciliğinin Adımları

3.2.2.1. Metin Koleksiyonu Oluşturma

Analize konu olacak veriler toplanarak, bir koleksiyon oluşturulmasıdır. Günümüzde genellikle bilinen arama motorları kullanılarak veri toplanır. Buna ek olarak, yapısal veri kaynakları, veritabanı ve bilgisayarda bulunan metin veri kaynakları da kullanılabili [101], [109].

3.2.2.2. Metin Önişleme

Metin önişlemede, metin öncelikle kelimelerine ayrılır. Metin madenciliğinin önemli analizlerinden frekans analizinde, kelimelerin metinde ne kadar geçtiğinin belirlenebilmesi için, kelimenin cümle içinde kullanım şekli, yani cümlenin öğelerine göre kelimenin anlamı bulunur. Türkçe sondan eklemeli bir dil olduğu için, aynı kökten türemiş, aynı anlamlı kelimeler metinde, yazılış olarak farklı şekillerde yer alabilir. Kelimeler köklerine ayrılarak, bu sorunun üstesinden gelinebilir. Analiz açısından herhangi bir önemi olmayan, bağlaç, noktalama işareti, semboller vb. kelimeler metinden ayıklanır. Bunlara ek olarak, Türkçe dilbilgisi kurallarına göre, yazımsal yanlışların belirlenmesi ve düzeltilmesi işlemleri bu süreçte yapılır.

3.2.2.3. Metin Dönüşümü

Hece ve eklerine ayrılmış kelimelerin köklerinin bulunması sürecidir. Kök bulma (stemmer) algoritmaları bu aşamada devreye girer. Türkçe gibi sondan eklemeli bir dilde, kök bulma algoritmalarının başarı oranı düşüktür. Bu aşamada aşağıdaki algoritmalar kullanılır:

- ✓ **Snowball:** Kelime kökü bulmada kullanılan en bilindik algoritmadır.
- ✓ **Kelime Türü:** Kelimenin cümlenin içerisinde geçtiği bölgeye ve türlerine göre etiketlenmesidir. Bu işlemlerde, Pos Tagging olarak isimlendirilen algoritmalar kullanılır.
- ✓ **Stopword İşlemi:** Bir dilde, yazılan metinde sıkça tekrar edilen, fakat tek başına bir anlam ifade etmeyen kelimeler vardır. Bu kelimeler dillere göre stopwords listelerinde yer alır ve analiz yapılmadan önce metinden filtrelenir.
- ✓ **Bagofwords Vektörü:** Analizle ilgili olarak toplanan belgelerin, makine öğrenmesi modelleriyle analiz edilebilmesi, bir matris biçimine getirilmesidir. Belgeler matrisin satırını oluştururken, kelimeler sütunu oluşturur. Bir kelimenin hangi belgede, ne kadar geçtiği ise, hücrelerin değerini oluşturur. Çeşitli ağırlıklandırma yöntemleriyle, kelimelerin belgeyle ne kadar ilgili olduğu derecelendirilebilir.

3.2.2.4 Özellik Seçme

Metin madenciliği ile analiz edilmek için toplanan belgeler, analiz konusu ile ilgili olmayan, hatalar içeren birçok bilgiden oluşabilmektedir. Metin madenciliği işlemlerine başlamadan önce, hatalı, modelle ilişkisi olmayan, birkaç belge ile sınırlı kalan, belgelerin tümüne yayılmayan bilgilerin, verilerin içinden ayıklanması gereklidir. Bu amaçla, veri madenciliği süreçlerinin iş yükünü azaltmak için, veri ön işleme süreçlerinden sonra çalıştırılan özellik seçme sürecinde, veri kümesinin tümüne yayılım gösteren, analiz konusu açısından önemli kelimelerin seçilmesi, bir kısım belgede gözlemlenen, analiz konusu ile pek ilintili gözükmeyen kelimelerin çıkarılması işlemleri gerçekleştirilir [110], [111].

3.2.3. Metin İşlemenin Adımları

Metin madenciliği işlemlerinde dil bağımsız olarak kullanılan işlem aşamaları aşağıda anlatılmıştır:

1. Analizde kullanulacak metnin dilinin ve analiz konusuna ilişkin olarak, metin madenciliğinde kullanılacak belgelerin tanımladığı alanların belirlenmesi. Metin dili ilerleyen aşamalarında kullanılacak, yöntem, parametreler ve algoritmalarda etkili olacaktır.
2. Metin kelime ayrıştırma, metni oluşturan kelime dizilerinin bulunması.
3. Analiz konusu açısından önem içermeyen, metin dilinin yapısı gereği kullanılan, bağlaç, zamir, noktalama işaretleri ve simgeler gibi kelimelerin metinden ayıklanması.
4. Ayıklanma işlemi sonrası elde edilen kelimeler üzerinde standartlaştırma ve kök bulma süreçlerinin çalıştırılması. Özellikle kök bulma süreci, Türkçe gibi sondan eklemeli dillerde önemlidir. Türkçe dilinde aynı kelime, metin içerisinde kullanımına göre sonuna ek olarak farklı şekilllerde görülebilmektedir. Bu da frekans analizi gibi süreçlerde, aynı kelimenin farklı kelimeler olarak sayılmasına ve analiz sonuçlarında yanlışlık oluşmasına neden olabilmektedir. Bu nedenle frekans analizi öncesinde, kök bulma süreci ile kelimelerin kökleri bulunarak, kelimenin metin içerisinde geçme sayısı doğru hesaplanmaktadır.
5. Yazılışları aynı, okunuşları farklı kelimeler belirlenerek, frekans analizi gibi süreçlerde aynı kelime olarak sayılması.

6. Metnin diline göre yazımsal hata içeren kelimelerin düzeltilmesi.
7. İşlem adımları sonucunda elde edilen kelimelerin, geliştirilen metin madenciliği uygulamasının gereklerine göre, kolay erişilebilir şekilde, saklama yapılarına kaydedilmesi.

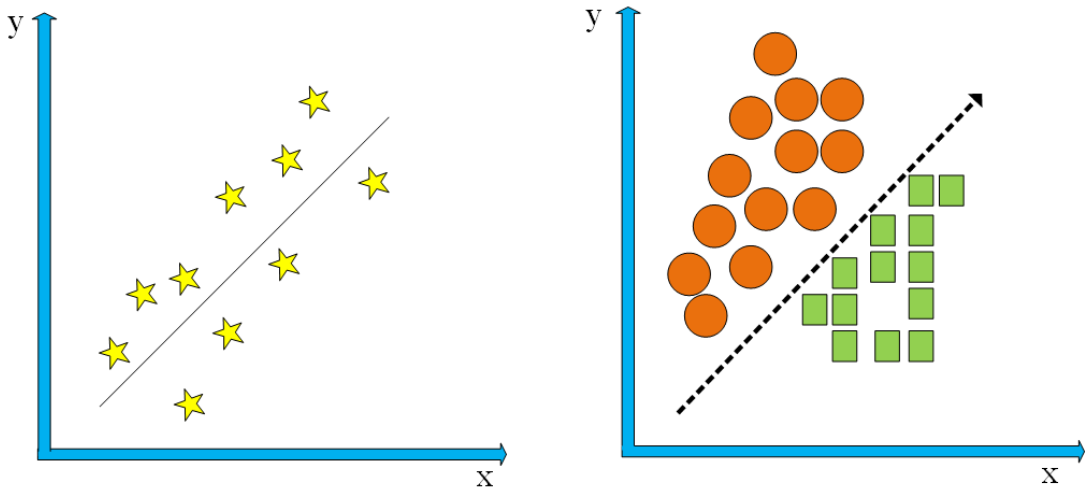
3.3. Makine Öğrenmesi Algoritmaları

Makine öğrenmesi, karmaşık veri yığınlarından analizler yaparak elde edilen çıkarımlarla, öğrenme işleminin gerçekleştirildiği yapay zeka tekniğidir. Veri madenciliğinde, özellikle Endüstri 4.0' la veri biliminde, makine öğrenmesi büyük önem kazanmıştır.

Makine öğrenme algoritmaları kabaca, analiz verilerini sınıflandırmaya çalışır. Makine öğrenmesinde sınıflandırma, denetimli ve denetimsiz öğrenme olmak üzere iki farklı algoritma modeliyle tasarlanır. Denetimli algoritmalarda sisteme öğretmeye çalışılan modele ilişkin veriler bulunmakta olup, bu verileri sisteme öğreterek, veri giriş ve çıkış değerleri arasındaki ilişki bulunmaya çalışılır. Buna ek olarak, öğretilen modeli test etmek amaçlı olarak kullanılan, sonuç değerleri bilinen test veri kümesi bulunmaktadır. Öğrenme veri kümesi ile eğitilen analiz modelinden, test verileri ile tahminler üretmesi beklenir. Test verisi ile yapılan tahmin değerleri ile gerçek değerler karşılaştırılarak, modelin doğruluğu ve hata oranları belirlenir. Eğitim verileri ile eğitilen modeller, sonucu bilinmeyen veriler için de tahminler üretebilir.

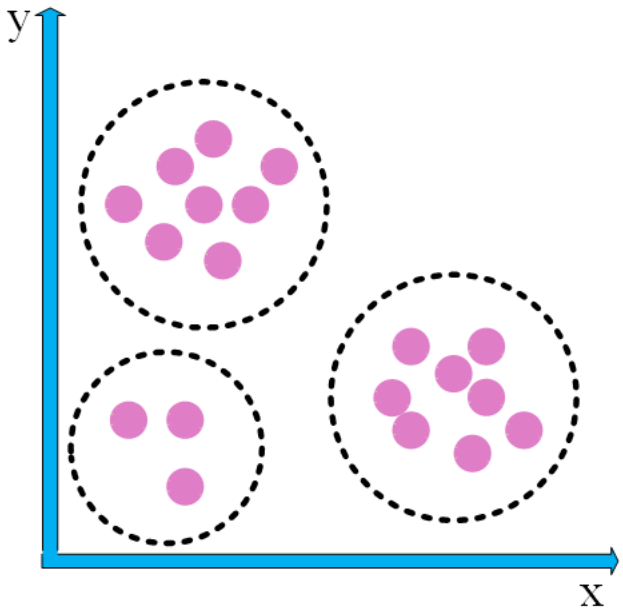
Denetimli öğrenme algoritmaları modelin sürekli veya kategorik değişken içermesine göre, regresyon veya sınıflandırma algoritmaları kullanılarak analiz edilir. Regresyon modelleri sürekli değişkenler içermekte olup, sürekli bir eğri ile uyumlu tahminler üretilir. Sınıflandırma modelleri ise kategorik değişkenlerden oluşmakta olup, veri elamanlarını bu kategorik sınıflarda sınıflayacak tahminler üretilmeye çalışılır.

Regresyon algoritmalarında sürekli bir eyilimde tahminler ürerilmeye çalışılırken, sınıflandırmada ise, çıktılar ayrı sınıflarda tahmin edilmeye çalışılır [112].



Şekil 3.12. Regresyon Algoritmalarında Veri Dağılımı

Denetimsiz öğrenmede ise sadece veri vardır. Verinin neyle ilişki olduğu hakkında bir bilgi olmadığı gibi, tahmin değerleri ile ilgili geribildirim de yoktur. Denetimsiz öğrenme algoritmalarında modelin gözetlenmesi ve yönlendirilmesine gerek yoktur; model verileri kendi kendine öğrenir. Bu algoritmalarda kümeleşme yöntemlerinden yararlanarak verilerin içerdiği ilişkiler ve özellikler ile ilgili çıkarımlar yapılmaya çalışılır [113].



Şekil 3.13. Sınıflandırma Algoritmalarında Veri Dağılımı

3.3.1. K-nn Algoritması

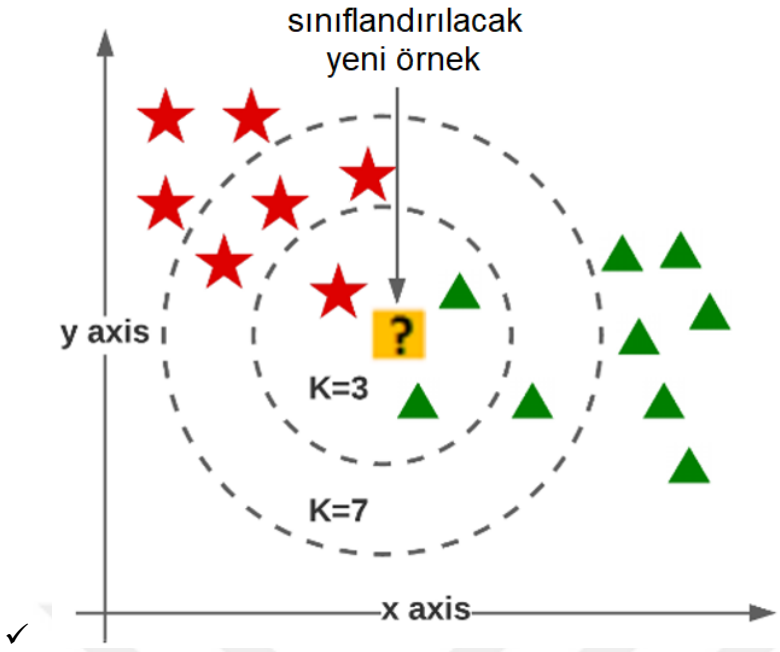
K-nn algoritmasında bir veri kümesi, n sayısında kümelere bölünmeye çalışılarak kümelenir. Veri kümesi içerisinde seçilen küme merkez değerlerine göre; verilerin merkeze uzaklıkları kıyaslanarak kümelenmeye çalışılır [114].

K-nn algoritması denetimli öğrenme algoritması olup, kümeleme problemlerinde olduğu kadar, regresyon tamamlama problemlerinde kullanılır. Genel olarak kullanım alanı kümeleme problemleridir. K-nn algoritmasının iki temel özelliği bulunmaktadır [115]:

1. **Tembel Öğrenme Yöntemi:** K-nn algoritmasında özel bir öğrenme aşaması yoktur. Öğrenme kümeleme aşamasında gerçekleşir. Bu nedenle, K-nn algoritması bir tembel öğrenme algoritmasıdır.
2. **Parametresiz Öğrenme Algoritması:** K-nn algoritması analiz edilen veri üzerinde herhangi bir varsayım yapmadığı için, parametrik olmayan bir algoritmadır.

K-nn algoritmasının aşamaları aşağıda verilmiştir:

- ✓ Algoritma adımlarında öncelikle n adet merkez noktası belirlenir.
- ✓ Veri kümesi elamanları n adet merkez noktasına uzaklıkları hesaplanarak, en yakın oldukları merkez noktasının grubunda kümelenir.
- ✓ n merkez noktası kadar küme oluşturulduktan sonra, kümeler birbirleriyle kıyaslanır. Bu kıyaslama sonucunda küme elamanlarının yer değiştirmesi ve kümelerde değişim olup olmadığı kontrol edilir.
- ✓ Kümelerde değişim varsa, algoritma çalışmaya devam eder, yoksa süreç durur.
- ✓ Kümeleme safhasındaki uzunluk hesaplamada, Öklid, Hamming, Manhattan, Minkowski veya Manchester uzaklığı fonksiyonları kullanılabilir.
- ✓ K-nn algoritması yeni veri noktalarının, yeni değerlerini tahmin ederken, özellik benzerliğini kullanır. Yeni veri noktaları, öğrenme veri kümesine ne kadar yakın olduğuna göre değer alır.



Şekil 3.14. Knn Algoritması

Knn algoritmasının adımları, aşağıda anlatılmıştır [116]:

- ✓ Algoritmada kullanılacak veri kümesi, öğrenme ve test veri kümesi olarak ikiye bölünür.
- ✓ Öğrenme ve test veri kümeleri sisteme yüklenir.
- ✓ Sistem öğrenme veri kümesi ile eğitilir.
- ✓ En yakın veri noktası, *k* nokta değeri seçilir.
- ✓ Tüm test verileri için;

1. Test verisi ve her bir öğrennme verisi arasındaki uzaklık hesaplanır. Uzaklık hesaplamasında, Öklid, Manhattan veya Hamming uzaklığı yöntemleri kullanılır. Bu yöntemlerden en sık kullanılanı Öklid uzaklığıdır.
2. Veri noktaları uzaklık değerlerine göre, küçükten büyüğe sıralanır.
3. Sıralı diziden ilk *k* nokta seçilir.
4. Test noktası en sık karşılaşılan sınıfa göre bir sınıfa atanır.

K-nn algoritması, uygulandığı analiz modelinin sınıflandırma veya regresyon olmasına göre farklı sonuçlar üretir [115]:

K-nn algoritmasının sınıflandırma modellerine uygulanması durumunda elde edilen sonuçlar sınıf üyeliğidir. Sınıf üyeliği yukarıdaki algoritmada anlatıldığı gibi, uzaklık fonksiyonları ile en yakın komşular bulunarak kümeleme ile elde edilir. K-nn algoritmasının regresyon modellerine uygulanması durumunda, model değişkenleri sürekli olacak olup, sonuç değerleri sürekli değişkenin değeri olacaktır. Sonuç değeri, uzaklık fonksiyonlarına göre en yakın komşu olarak gruplanan küme elamanlarının ortalaması alınarak hesaplanır.

K-nn sınıflandırmasında, yakın komşuların kümelemede daha etkin rol oynaması için, hesaplamalarda ağırlıklandırma kullanılır. Yakın komşulara daha yüksek ağırlık değerleri verilirken, uzak komşulara düşük ağırlık değerleri verilir.

Algoritmada kullanılabilecek uzaklık fonksiyonlarının matematiksel gösterimi aşağıda görülmektedir:

Sürekli değişkenler kullanıldığında;

- ✓ Öklid,

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}, \tag{3.25}$$

- ✓ Manhattan,

$$\sum_{i=1}^{k}|x_i - y_i|, \tag{3.26}$$

- ✓ Minkowskii

$$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{\frac{1}{q}}, \tag{3.27}$$

uzaklık fonksiyonları kullanılır.

Kategorik değişkenler kullanıldığında ise Hamming uzaklığı kullanılabilir. Kategorik değişkenler, fonksiyon sonucunun hesaplamalara göre, kategorik sonuçlardan birini oluşturduğu durumlarda kullanılır. Hamming uzunluğu aşağıdaki denklemle ifade edilebilir:

$$D_H = \sum_{l=1}^{k} |x_i - y_i|$$
$$x = y \Rightarrow D = 0$$
$$x \neq y \Rightarrow D = 1 \tag{3.28}$$

K-nn algoritmasında önemli noktalardan biri, k değerinin belirlenmesidir. k değerini belirlemek için veri kümesi incelenmelidir. Büyük k değerleri, veri kümesi içerisinde gürültüyü düşürmek için hassas değerler üretmesi açısından önemlidir. Bağımsız veri kümesi kullanarak geriye doğrulama yapan çapraz doğrulama, k değerlerinin hesaplanması açısından önemlidir. Genel olarak, k, 3 ile 10 arasında değerler alır.

Algoritmanın doğru sonuçlar üretmesi açısından, eğitim veri kümesi üzerinde, öğrenme aşamasından önce normalleştirme süreçlerinin çalıştırılması gereklidir. Özellikle kategorik ve sürekli değişkenlerin birlikte kullanılması durumunda, ölçeklendirme farklılıkları değişkenlerin öğrenme modelinde etkisini göstermesini engelleyecektir. Bu nedenle normalleştirme süreçleri büyük önem taşımaktadır.

3.3.2. Naive Base Algoritması

Naive Base teoremini temel alan olasılık temelli bir sınıflandırıcıdır. Basit varsayımları temel almasına rağmen, gerçek dünya olayları konusunda çok iyi sonuçlar üretebilmektedir.

Naive Bayes algoritması modelin kategorik değişkenlerden oluştuğu durumlarda, iyi sonuçlar verir [117]. 18.yüzyılda ünlü Matematikci Tomas Bayes tarafından bulunan sınıflandırma algoritması, modele istina değişkenlerin, birbirinden bağımsız olduğu varsayımını temel alır. Gerçek dünyada çok olası bir durum olmamasına rağmen, sınıflayıcı karmaşık makine öğrenme problemlerinde iyi sonuçlar verir.

3.3.2.1. Bayes Teoremi

Belirlenen bir rassal değişkenin tabi olduğu koşullu olasılıklarla, marjinal olasılıkların ilişkisini temel alır.

A ve B olaylarının birbirleriyle ilgili iki olay olması koşuluyla, birlikte meydana gelme olasılığı, olasılık teoreminin çarpım kuralı kullanılarak hesaplanabilir.

$P(A|B)$, B olayının oluşması durumunda, A olayının gerçekleşme olasılığı, $P(B|A)$, A olayının oluşması durumunda, B olayının olma olasılığı, $P(A)$ ve $P(B)$, A ve B

olaylarının marjinal olasılıkları olması koşuluyla, $P(A)$ ve $P(B)$ olasılıkları 0' dan büyüktür. Bu durumda, A ve B olaylarının ikisinin birlikte oluşma olasılığı, aşağıdaki ifadelerle hesaplanabilir:

1. $P(B \wedge A) = P(B)P(A|B)$ ifadesi ile iki olayın birlikte oluşma olasılığı, B olayının oluşma olasılığı ile, B olayından sonra, A olayının oluşma olasılığı çarpılarak hesaplanır.
2. $P(A \wedge B) = P(A)P(B|A)$ ifadesi ile iki olayın birlikte oluşma olasılığı, A olayının oluşma olasılığı ile, A olayından sonra, B olayının oluşma olasılığının çarpılmasıyla elde edilir.

Bu iki olayın birlikte oluşması olasılığını hesapladığı için, bu iki olasılık değerinin birbiri ile eşit olması beklenir. Bu varsayımdan hareketle, aşağıdaki ifadeler elde edilir:

$P(A)P(B|A) = P(A \wedge B) = P(B)P(A|B)$ genel olasılık bağıntısı kullanılarak, $P(B|A)$, A olayından sonra, B olayının olma olasılığı ve $P(A|B)$, B olayından sonra, A olayının olma olasılığı hesaplanabilir.

Bu hesaplamalar ışığında aşağıdaki ifadeler elde edilir:

$$P(B|A) = \frac{P(B)P(A|B)}{P(A)} \tag{3.29}$$

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} \tag{3.30}$$

Teorem 3.1. (Olasılıkların Çarpımının Toplamı): S uzayı, $B_1, B_2, \dots, B_n$ karşılıklı olarak birbirine engelleyen olaylar dizisinin birleşiminden oluşsun. B olaylarından birinin mutlaka meydana gelmesi ve bu olaya bağlı olarak A olayının oluşması durumunda, A olayının oluşması olasılığı, aşağıdaki gibi hesaplanır:

$$P(A) = P(B_1).P(A|B_1) + P(B_2).P(A|B_{21}) + \dots + P(B_k).(A|B_k) \tag{3.31}$$

$$P(B_i|A) = \frac{P(B_i).P(A|B_i)}{P(A_i)} = \frac{P(B_i).P(A|B_i)}{P(B_1).P(A|B_1) + P(B_2).P(A|B_{21}) + \dots + P(B_k).(A|B_k)} \tag{3.32}$$

3.3.2.2. Naive Bayes Sınıflandırma Modeli

C değişkeninin, F değişkeninin özelliklerini gösterdiği bir ifadede, naive bayes sınıflandırıcı, (3.31) ifadesinde görüldüğü gibi, koşullu olasılıkların tümünün çarpımı olarak ifade edilebilir.

$$p(C|F_1, \dots F_n) = \frac{P(C)p(F_1, \dots F_n|C)}{p(F_1, \dots F_n)} \tag{3.33}$$

Naive base algoritması, birbirinden bağımsız olmayan olayları içeren bir X veri kümesinin olması durumunda, sonuç olarak oluşacak olayın olasılığını hesaplamaya çalışır.

X nedenler kümesi için, $P(C|X) = X$ olup, C sınıfının oluşma olasılığı, $X =< x_1, \dots, x_k >$ 'dir. Naive bayes algoritmasındaki temel fikir, X veri kümesinden, $P(C|X)$ olasılık değeri maksimum olan sınıfı bulmaktır.

Bayes teoreminden hareketle $P(C|X) = P(X|C) \cdot P(C) \,/\, P(X)$ olasılık hesabı olup, $P(X)$, bütün sınıflar için aynı değerde bir ölçek parametresidir. $P(C)$ ise, C sınıfının sınıflar içindeki bağıl frekansıdır.

MAP (maximum posteriori) Hipotezi

C seçiminin, $P(C|X)$ değerini maksimum yapması durumunda, $P(X|C)$, $P(C)$ olasılık değerleri de maksimum olur. Bu karmaşık hesaplamaları yapmak yerine, değişkenlerin bağımsız olduğu kabul edilebilir. Böylelikle problem naive base algoritmasına dönüşür.

Olay nedenlerinin bağımsızlığının naive kabul edilmesi durumunda, olayların gerçekleşme olasılığı, $P(x_1, \dots, x_k|C) = P(x_1|C) \cdot \dots \cdot P(x_k|C)$ olarak ifade edilir.

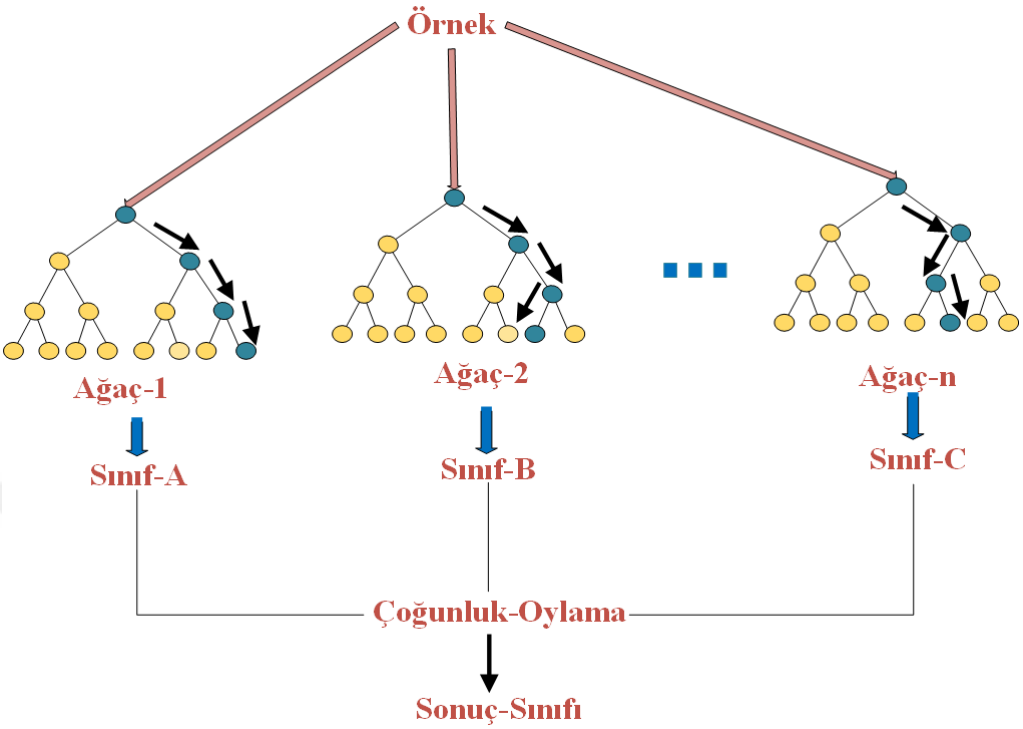
i. etkinin kategorik olması durumunda; $P(x_i|C)$ C sınıfı içindeki x_i değerlerinin bağıl frekansı olarak hesaplanır.

i. etkinin sürekli olması durumunda; $P(x_i|C)$ Gauss yoğunluk fonksiyonu (normal dağılım) ile hesaplanır.

3.3.3. Rastgele Orman Algoritması

Rastgele orman algoritması ağaç bazlı bir karar verme algoritması olup, veri kümesinden rastgele seçilen elamanlarla, karar ağacı oluşturarak öğrenme süreci

işletilir [118]. Rasgele orman algoritmasında, veri kümesinden oluşturulan farklı alt karar ağaçlarından elde edilen değerlerin toplanması ile nihai karar ağacı elde edilir.



Şekil 3.15. Rastgele Orman Algoritması

Rastgele ormanın üyesi olan her bir alt ağacın, x giriş değeri için popüler sınıfın seçiminde bir oyu vardır. Bu oylar nihai sonucun oluşmasında kullanılır [119].

Rastgele orman algoritmasının özellikleri ve avantajları aşağıda listelenmiştir:

1. Bilinen en iyi sonuç üreten öğrenme algoritmasıdır. Çok yüksek derecede doğrulukta sınıflayıcı üretir.
2. Büyük veri tabanlarında çok etkin olarak çalışır.
3. Binlerce giriş veri değişkenini, değişken silmeden kotarır.
4. Sınıflandırma sürecinde, hangi değişkenin önemli olduğu konusunda tahminler verir.
5. Karar ağaçları ormanı üretme işlemi devam ederken, içsel önyargısız genelleştirme hatası tahmini yapar.
6. Eksik veri tahmininde, etkin bir yönteme sahip olmakla birlikte, büyük oranda verinin eksik olması durumunda bile tahmin doğruluğunu korumaktadır.

Rastgele orman algoritmasının dezavantajları aşağıda listelenmiştir:

1. Random orman algoritmasının bazı veri kümeleri için, gürültülü sınıflama ve regresyon görevleriyle, aşırı öğrenmeye (overfit) maruz kaldığı saptanmıştır.
2. Farklı seviyelerde kategorik değişkenler içeren veri kümelerinde, random forest algoritmasının birçok seviyede, bu kategorik değişkenlerin lehinde sonuçlar ürettiği saptanmıştır.

3.3.4. Lasso ve ElasticNet Algoritmaları

Lasso algoritması, doğrusal regresyon için bir daralma ve seçim algoritmasıdır. Lasso algoritması, katsayıların mutlak değerli toplam sınırı ile hata oranlarını minimum yapmaya çalışır [120].

Lasso algoritması, $x_1, x_2, \dots, x_p$ giriş değerleri ve y çkış değeri için aşağıdaki doğrusal modele uymaya çalışır.

$$y_{hat} = b_1 x_1 + b_2 x_2 + \cdots + b_p x_p, \tag{3.34}$$

Veri kümesini doğrusal bir modele uydurmaya çalışırken;

$$min \sum (y - y_{hat})^2 \tag{3.35}$$

ifadesi ile hata kareleri toplamı,

$$\sum |b_j| \leq s \tag{3.36}$$

koşuluna göre minimum yapılır.

Lasso algoritmasında s parametresi algoritmanın etkinliğini belirleyen ince ayar parametresidir. s parametresinin çok yüksek değerlerde olması durumunda, kısıtların çözüm üzerinde bir etkisi olmaz ve algoritma klasik çoklu doğrusal regresyon algoritmasına dönüşür.

$(s >= 0)$ olmak üzere, küçük s değerleri için, çözüm, en küçük kareler tahminlerinin daraltılmış bir şekline dönüşür. Çoğunlukla, katsayılar sıfır değerini alır. s parametresinin seçimi regresyon modelinde kullanılan tahminleyici sayısını seçmeye benzemekle birlikte, s değerini belirlemede, çapraz doğrulama iyi bir araçtır.

Lasso ve ElasticNet doğrusal regresyon algoritması temelli algoritmalardır. Lasso algoritmasında hata düzeltmeleri karesel değerleri kullanırken, ridge regresyonda mutlak değerleri temel alır. ElasticNet algoritması ise iki özelliği birden kullanır [121]. Basit regresyon algoritması, $y = \beta_0 + \beta_1 x_1 + \varepsilon$ ifadesi ile gösterilir. Matris olarak ise,

$$H\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & \dots & x_1 \\ \vdots & \vdots & \vdots \\ 1 & \dots & x_n \end{bmatrix}\begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix} \tag{3.37}$$

şeklinde ifade edilir. Bu noktada β katsayısı, $\beta = (X'X)^{-1}X'Y$ olarak hesaplanır. Bilinen regresyon algoritmasında değişkenler arasındaki yüksek bir korelasyon olması durumunda, regresyon eğrisi ile uyumsuz sonuçlar oluşabilir [122]. Bu nedenle ridge regresyonda, hatayı azaltıcı yönde bir değişken, (3.38)' deki gibi ifadeye eklenir.

$$\beta(ridge) = \arg min\|y - x\beta\|^2 + \lambda\|\beta\|^2 \tag{3.38}$$

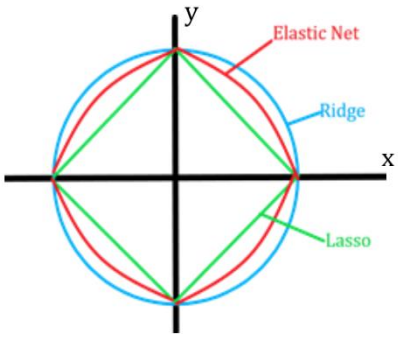
Karesel düzeltmenin yapıldığı, yukarıdaki ifadede $\lambda \geq 0$ değeri düzeltme veya karmaşıklık katsayısı olarak tanımlanır. Lasso algoritmasında da benzer bir şekilde düzeltme vardır. Farklı olarak düzeltme, aşağıdaki ifadeden de görülebileceği gibi mutlak değer şeklindedir.

$$\beta(lasso) = \arg min\|y - x\beta\|^2 + \lambda\|\beta\| \tag{3.39}$$

Mutlak değer hata düzeltmesi, yüksek hata değerlerinin kare işlemi ile katlanmasının engellenmesi açısından önemlidir.

Lasso algoritmalarının sınırlamalarını, bir ceza fonksiyonu kullanımı ile aşan ve Şekil 3.16' da gösterimi verilen, Elasticnet algoritmasında ridge ve lasso algoritmalarındaki iki tekniğin karışımı kullanılır ve aşağıdaki gibi ifade edilir [123].

$$\beta(elasticnet) = \arg min\|y - x\beta\|^2 + \lambda_1\|\beta\|^2 + \lambda_2\|\beta\| \tag{3.40}$$



Şekil 3.16. Lasso ve Elastic Net Algoritması

3.3.5. Stokastik Gradyan İniş (Stochastic Gradient Descent) Algoritması

Stokastik gradyan iniş algoritması, makine öğrenmede popüler bir algoritma olup yapay sinir ağlarına temel oluşturmaktadır [124].

Gradyan bir fonksiyonun eğimidir. Değişkenlerden birindeki değişimin, diğerini ne kadar etkilediğine bakılarak hesaplanır. Gradyan iniş matematiksel olarak, çıkış değeri giriş değerlerinin kısmi türevi olan konveks bir fonksiyondur.

Gradyan iniş algoritması fonksiyonları, eğrilerin eğimiyle ilgilenir. Eğim boyunca aşağı yönlü hareket ederek, en küçük y değerini oluşturan, x değerini bulmaya çalışır.

Gradyan iniş algoritması rastgele bir noktadan başlayarak, aşamalı olarak eğimi aşağı doğru takip ederek, en düşük fonksiyon değerine ulaşmaya çalışır.

Basit bir öğrenme modelinde z, rastgele giriş değeri x ve y çıkış değerinden oluşan (x, y) veri çifti olsun. Kayıp fonksiyonu, y gerçek değer iken, $\hat{y}$' sünü tahmin etmenin maliyetini ölçer. Bu şekilde $\mathcal{F}$ olarak adlandırılan, ağırlık vektörü w tarafından tanımlanan $f_w(x)$ fonksiyon ailesi bulunur. Bu noktadan hareketle $Q(z, w) = \ell(f_w(x), y)$ kayıp fonksiyonunu minimum yapacak $f \in \mathcal{F}$ fonksiyonu araştırılır. Deneysel risk $E_n(f)$,

$$E_n(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(x_i), y_i) \tag{3.41}$$

ifadesi kullanılarak eğitim veri kümesinin başarısı ölçülür [125].

Gradyan iniş algoritması, $E_n(f_w)$ kayıp fonksiyonunun, gradyan iniş kullanılarak minimum yapılması olarak sunulmakta olup, Şekil 3.17' de gösterimi verilmiştir [126]. Gradyan iniş algoritmasında w ağırlık değerleri, $E_n(f_w)$ kayıp fonksiyonunu minimum yapacak şekilde,

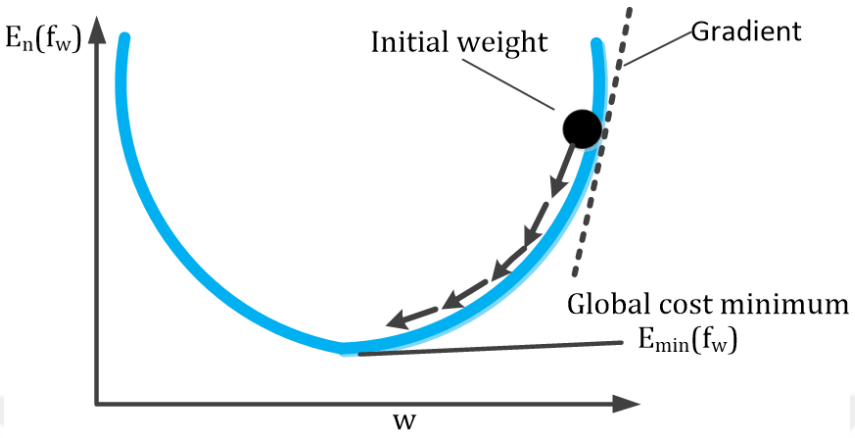
$$w_{t+1} = w_t - \gamma \frac{1}{n}\sum_{i=1}^{n} \nabla_w Q(z_i, w_t) \tag{3.42}$$

ifadesi ile güncellenir.

Stokastik gradyan iniş algoritmasında, her bir adımda, $E_n(f_w)$ kayıp fonksiyonunun tüm gradyan değerleri hesaplanmak yerine, rastgele seçilen tekil z_t örneği temel alınarak tahmin değeri kullanılır. Böylelikle (3.42) ifadesinde gösterimi yapılan ağırlık güncelleme ifadesinde basitleştirme sağlanmıştır:

$$w_{t+1} = w_t - \gamma_t \nabla_w Q(z_i, w_t) \tag{3.43}$$

Stokastik işlem, $\{w_t,\ t = 1, \dots\}$, algoritma adımlarında rastgele seçilen örneğe bağlıdır.



Şekil 3.17. Stokastik Gradyan İniş Algoritması

Algoritmanın aşamaları, aşağıda listelenmiştir:

1. Amaç fonksiyonunun eğimi, her bir parametre ve özelliğe göre bulunmaya çalışılır. Kısacası gradyanlar bulunmaya çalışılır.
2. Parametreler için rastgele başlangıç değerleri seçilir.
3. Parametre değerlerini yerine koyarak gradyan fonksiyonu yenilenir.
4. Her bir özellik için adım büyüklüğü, (Adım büyüklüğü = gradyan * öğrenme oranı) ifadesi ile hesaplanır.
5. Yeni parametreler, (Yeni parametreler = eski parametreler - adım büyüklüğü) ifadesiyle hesaplanır.
6. Gradyan 0 olana kadar, 3-5 adımları tekrarlanır.

Öğrenme oranı, gradyan iniş algoritması için önemli bir parametredir. Öğrenme oranının yüksek bir değer seçilmesi, algoritmanın eğim boyunca büyük adımlarla ilerlemesine ve minimum değer alan noktayı kaçırmasına neden olur. Bu nedenle 0.01 gibi küçük bir değer seçilmesi, algoritmanın doğru değerlere ulaşabilmesi açısından önemlidir. Gradyan iniş algoritması, yüksek bir değerden algoritmaya başlandığında, eğim boyunca büyük adımlarla hareket eder; hedefe yaklaştıkça hedef değerini kaçırmamak için, çok küçük değerlerle hareket eder.

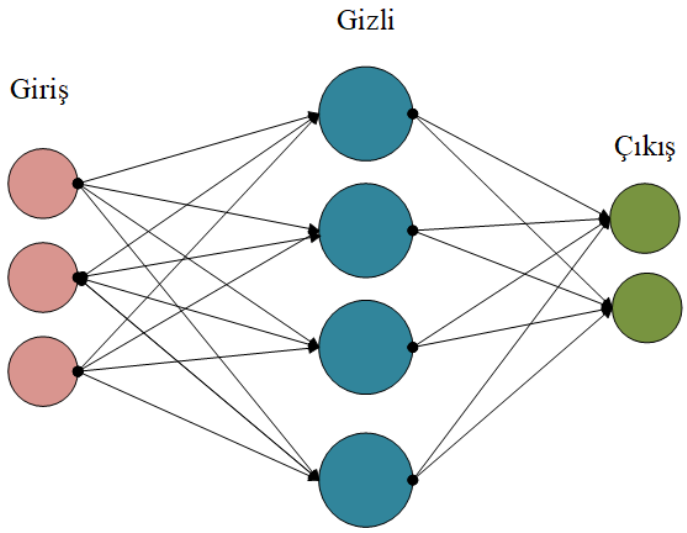
Gradyan iniş algoritması, her bir özellik için, veri noktalarının tümü ile işlem yapması gerektiği için, büyük veri boyutlarında çok yavaş çalışır. Stokastik gradyan iniş algoritması rastgele verilerle çalışma özelliği nedeniyle, veri madenciliğinde yüksek başarım göstermektedir.

Stokastik gradyan iniş algoritmasında, her aşamada veri kümesinden rastgele olarak bir veri seçilir. Bu işlem veri hesaplamaları yükünü büyük oranda azaltır. Her aşamada bir veri yerine, küçük sayıda veri örneklemi de seçilebilir. Mini yığın olarak adlandırılan bu yöntemle, gradyan inişin hem iyi tahmin üretmesi, hem de algoritmanın hızlılığı sağlanır.

3.3.6. Perceptron Algoritması

Yapay sinir ağları, biyolojik nöronları örnek alan bir sistemdir. Bu öğrenme sistemlerinde insan beyninin çalışma şekli örnek alınır. İnsan beyninde birbirleriyle elektromekanikal sinyallerle iletişime geçen, birbirine bağlı 100 milyon sinir hücresi vardır. Sinir hücrelerini birbirine bağlayan bağlantılara sinaps denir. Her bir nöron binlerce nörona bağlı olup, bu nöronlardan sinyaller alır. Bu sinyallerin toplamı belirli bir eşik değerini aşınca, sinir hücresi aksonlar aracılığı ile cevap verir ve öğrenme gerçekleşmiş olur [127].

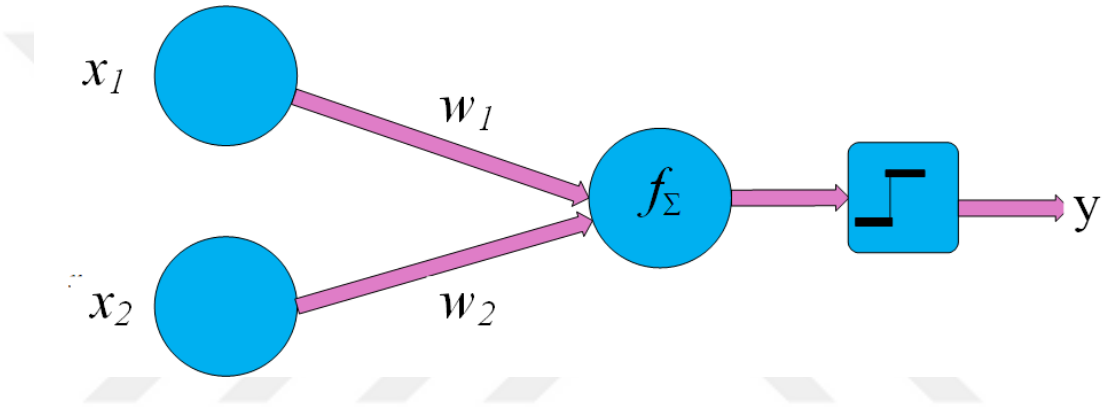
İnsan beynini örnek alan bu sistemde, insan beynindeki karmaşıklıkta nöron iletişimi kurgulamak mümkün olamamaktadır. Bu nedenle, insan beyninin gücünde öğrenme yetenekleri gösteren yapay sistemler tasarlamak henüz mümkün olamamaktadır.



Şekil 3.18. Örnek Yapay Sinir Ağı

Şekil 3.18’ de örnek bir yapay sinir ağı görülmektedir. Yapay sinir ağlarına giriş değerleri, pembe ile gösterilen giriş katmanındaki nöronlardan verilir. Bu sistemde nörönlar sinaps dediğimiz bağlantılarla birbirlerine bağlıdırlar. Her bağlantının bir ağırlığı ve fonksiyonu bulunmakta olup, giriş değerlerine bu fonksiyonlar uygulanarak çıkış katmanlarına doğru ilerletilir. Sonuç olarak, yeşille gösterilen katmanlardan çıkış değerleri alınır ve öğrenme gerçekleşmiş olur.

Perceptron öğrenme algoritması 1943 yılında, Rosenblatt tarafından bulunmuştur [128]. Perceptron bir nöronu en basit şekilde modelleyen matematiksel fonksiyondur. Perceptron bir veya birden fazla giriş değeri, işlemci ve çıkış değerinden oluşmakta olup, Şekil 3.19’da basit bir perceptron’un şematik gösterimi görülmektedir.



Şekil 3.19. Perceptron Algoritması Giriş Fonksiyonu

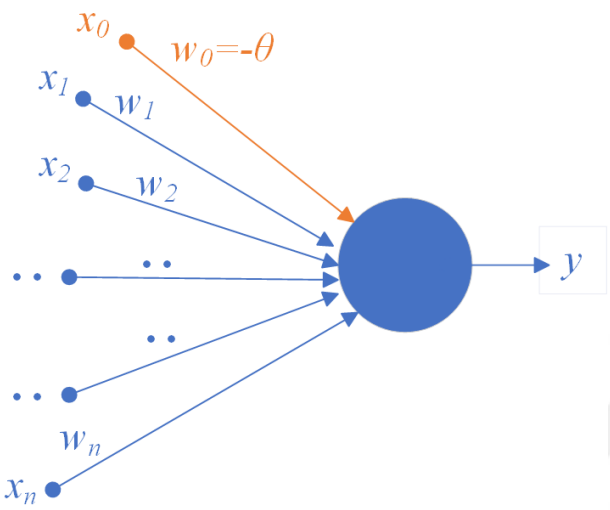
$$f_\Sigma = x_1 w_1 + x_2 w_2$$
$$y = \begin{cases} 0, & f_\Sigma < 0 \\ 1, & f_\Sigma \geq 0 \end{cases} \tag{3.44}$$

Genelleştirilmiş bir perceptron algoritması, n giriş değerinin ağırlıkları çarpımı toplamını hesaplar. Bu toplam değeri, eşik değerinden büyük ise, fonksiyon 1 değerini alırken, aksi durumda sıfır değerini alır [129].

Perceptron gösterimi ve hesaplama modelinin şematik gösteriminin verildiği, Şekil 3.20’de gösterilen perceptron’da, θ ile gösterilen bias değeri perceptron’a, $x_0 = 1$ olarak eklenen giriş değerinin ağırlığı olarak eklenmiştir.

Şekil 3.20’ de gösterilen perceptron matematiksel olarak aşağıdaki gibi ifade edilir:

$$y = 1 \quad \sum_{i=0}^{n} w_i * x_i \geq 0$$
$$y = 0 \quad \sum_{i=0}^{n} w_i * x_i < 0 \tag{3.45}$$



Şekil 3.20. Perceptron Algoritması Öğrenme Modeli

Perceptron Algoritması Adımları

Perceptron algoritması, öğrenme veri kümesini sınıflayan bir hiper düzlem bulmaya çalışır [130].

Giriş Değerleri: (x_1, y_1), (x_2, y_2) , tüm $x_i \in \Re^n, y_i \in \{-1,1\}$ için, öğrenme veri kümesi olsun.

r öğrenme oranı olup, 1 den küçük pozitif değer alır.

$sgn(w^T x)$ tahmin fonksiyonudur. Tahmin fonksiyonuna sabit bir değer olarak, b sabit değeri eklenebilir. Bu durumda fonksiyon değeri $sgn(w^T x + b)$ olur.

$w_0 = 0 \in \Re^n$, başlangıç değeri olarak atanır.

Tüm öğrenme kümesi verisi için (x_i, y_i),

- ✓ $y' = sgn(w_t x_i)$ değeri tahmin edilir.
- ✓ Eğer $y \neq y' ise$
- ✓ $w_{t+1} = w_t + r(y_i x_i)$ olarak güncellenir.

Algoritmanın son ağırlık vektörü çıkış değeri olarak adlandırılır.

3.4. Performans Değerlendirme Yöntemleri

3.4.1. Standartlaşma (Normalizasyon)

Standartlaşma, algoritma modelindeki değişkenlerin aldığı değerlerin normal dağılıma göre yeniden ölçeklendirilmesidir [131].

μ'nin ortalama, σ'nın standart sapma olması durumunda, z-puanları aşağıdaki gibi hesaplanır.

$$\begin{gathered}\mu = 0 \\ \sigma = 1 \\ z = \frac{x - \mu}{\sigma}\end{gathered} \tag{3.46}$$

3.4.1.1. Min-Max Ölçekleme

Sabit bir aralıkta verileri ölçeklendirmeyi amaçlayan bir yöntemdir. Bu aralık genellikle (0-1) arasında seçilir. Min-Max ölçeklemenin matematiksel ifadesi aşağıda görülmektedir.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.47}$$

3.4.1.2. Özellik Ölçeklendirme

Özellik ölçeklendirme, giriş değerlerini minimum, maksimum değerlerine göre aralıklandırarak, yeni bir değer aralığı oluşturulması sürecidir.

Ortalama normalleştirme ise, veri değişken değerlerinin ortalamasının değişken değerlerinde çıkarılmasıyla yeni bir veri kümesi elde edilmesidir.

Standartlaşmalar, μ_i değişkenlerin ortalaması, s_i, değer aralığıdır (maks – min) veya si standart sapma olması koşuluyla, matematiksel olarak aşağıdaki gibi ifade edilebilir:

$$x_i = \frac{x_i - \mu_i}{s_i}, \tag{3.48}$$

3.4.2. Çapraz Doğrulama

Sistemde, öğrenme algoritmasının veriler üzerinde çalıştırılmasından sonra yapılan işlemlerin doğrulanması gereklidir. Bu nedenle veriler eğitim ve test verileri olarak ikiye ayrılırlar. Eğitim verileri içinden de seçilen bu doğrulama verileriyle, eğitilen model üzerinden tahminler yapılmaya çalışılır.

Doğrulama yöntemlerinin en basiti olan hold-out yönteminde, analiz veri kümesi öğrenme amaçlı kullanılan eğitim verisi ve eğitilmiş modeli doğruluğunu test etmek amacıyla kullanılan test verisinden oluşur.

K katmanlı çapraz doğrulama algoritmasında, eğitim kümesindeki verilerin $k.$ kez eğitim işlemine uygulanması durumunda, algoritma $(k-1)$ kezde doğrulama kümesi olarak kullanılır. k değerinin büyük olması durumunda, varyans değerinin azalması ile karşılaşılır.

Tek-çıkışlı doğrulamada, k veri kümesindeki veri nokta sayısı n‘ dir. n nokta içinden bir nokta dışındaki verilerle algoritma eğitilir. İyileştirme fonksiyonu ilgili nokta üzerindeki veri ile test işlemine tabi tutulur.

Bu yöntemlerin değerlendirilmesi yapılırken, hata kareleri yöntemi kullanılarak eşleştirilmesi sağlanır.

3.4.3. Makine Öğrenme Algoritması Değerlendirme

Bir veri kümesine makine öğrenme algoritmasının uygulanması ve sonuçlar elde edilmesi, problemin çözümünün doğrulukla tanımlandığı anlamına gelmez. Bu nedenle çeşitli yöntemlerle, çözümün etkinliğinin belirlenmesi gereklidir. Makine öğrenmesinde kullanılan veri kümesinin belirleyici olduğu birçok metrikle, algoritmanın değerlendirmesi yapılabilir.

Makine öğrenmesi modelinin verimliliğini belirlemekte diğer bir önemli ölçüu, kullanılacak verilerdir. Başarım ölçümünde veri kümesi olarak eğitim veri kümesinin kullanılması, modelde önyargı oluşumuna neden olacaktır. Eğitim süresince modelin kendini eğitim verilerine göre hazırlaması, eğitim verilerinin test verileri olarak kullanılması durumunda, iyi sonuçlar üretecektir. Bu nedenle öğrenme veri kümesini eğitim ve test verileri olarak ikiye bölmek ve eğitim veri kümesi ile öğrenme süreci işletildikten sonra, test verisi ile tahmin yapmak daha doğru olacaktır [132], [133].

Makine öğrenmesi algoritmalarının değerlendirilmesinde, doğruluk ve hassasiyet ölçütleri önem arz etmektedir. Ardı ardına yapılan tahminlerde, gerçek değerlerin tahmin edilen değere yakın, fakat tahmin edilen değerlerin birbirlerine uzak olması, tahmin doğruluğunun yüksek olduğunu gösterirken, hassasiyetin düşük olduğunu gösterir. Tahmin değerlerinin hepsinin birbirine yakın, fakat gerçek değerden uzak olması durumunda, tahmin değerlerinin hassasiyeti yüksek, doğruluğu düşük olur [132], [134].

Makine öğrenmesi algoritmalarında karşılaşılan hatalar genel olarak iki sınfta toplanabilir:

- ✓ ***Sistematik hatalar:*** Bir desen dahilinde ortaya çıkan hatalardır. Sistemsel bir soruna işaret ederler.
- ✓ ***Rastgele hatalar:*** Bir desen dahilinde, ortaya çıkmayan hatalardır.

3.4.4. Metrikler

A_j gerçek değer, P_j tahmin edilen değer, n veri kümesi elaman sayısı olsun. $e_j = A_j - P_j$ ifadesi ile hata hesaplanır.

Ortalama Hata: Ortalama hata tahmin hatalarının ortalaması alınarak, aşağıdaki ifadedeki gibi hesaplanır [135]:

$$ME = \frac{1}{n}\sum_{j=1}^{n} e_j \tag{3.49}$$

Ortalama Yüzde Hata: Hesaplanan tahmin hata yüzdelerinin ortalaması alınarak hesaplanır. Aşağıdaki ifade ile gösterilebilir [135]:

$$MPE = \frac{100}{n}\sum_{i=1}^{n} \frac{e_j}{A_j} \tag{3.50}$$

Ortalama Mutlak Hata: Tahmin hatalarının mutlak değerinin ortalaması alınarak, aşağıdaki ifadedeki gibi hesaplanır [135]:

$$MEA = \frac{1}{n}\sum_{j=1}^{n} |e_j| \tag{3.51}$$

Ortalama Kare Hata: Ortalama kare hatası, noktaların regresyon doğrusuna ne kadar yakın olduğunu söyler. Sıfır değerine yakın değerler makine öğrenmesinin verimli olduğunu ifade eder. Ortama kare hatası aşağıdaki ifade ile hesaplanabilir [135]:

$$MSE = \frac{1}{n}\sum_{j=1}^{n} {e_j}^2 \tag{3.52}$$

Kök Ortalama Kare Hata: Ortalama kare hatasının karekökü alınarak aşağıdaki ifade ile hesaplanır [135]:

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n} {e_j}^2} \tag{3.53}$$

Ortalama Mutlak Yüzde Hata: Tahmin hata ve gerçek değerin mutlak değerlerine göre hata yüzdelerin ortalamasıdır. Aşağıdaki ifade ile hesaplanır [135]:

$$MAPE = \frac{100}{n}\sum_{j=1}^{n} \frac{|e_j|}{|A_j|} \tag{3.54}$$

3.4.5. Hata Matrisi

Hata matrisi iki sınıflandırmalı problemlerde, tahmin değerlerini, gerçekte doğru olan değeri doğru (TP), yanlış olan değeri yanlış (TN), doğru olan değeri yanlış (FP), yanlış olan değeri doğru (FN) olarak sınıflandıran hata matrisidir. Bu sınıflandırma yönteminde gerçek doğru (TP) ve yanlış (TN) sınıfında bulunan tahmin verilerinin değerlerinin yüksekliği iyi olarak nitelenir (bknz. Tablo 3.4).

Tablo 3.4. Hata Matrisi

		Gerçek Değerler: Doğru (1)	Gerçek Değerler: Yanlış (0)
Tahmin Verileri	Doğru (1)	TP	FP
	Yanlış (0)	FN	TN

Toplam veri sayısı, hata matrisindeki tüm sınıfların veri sayısı toplanarak hesaplanır (TP + TN + FP + FN). Gerçek doğruların sayısı (TP + FN), yanlışların sayısı (TN + FP) olarak hesaplanır.

Modelin doğruluk ve duyarlılık ölçümleri, hata matrisi sınıflandırmasını kullanan metriklerle gerçekleştirilir [136]:

Doğruluk: Doğru olarak tahmin edilen verilerin, toplam veri sayısına oranı ile hesaplanır.

$$Doğruluk = \frac{TP + TN}{TOPLAM} \tag{3.55}$$

Geri Çağırma: Tahmin verilerindeki doğru sınıflardan ne kadarının doğru tahmin edildiğini gösterir.

$$Geri\ Çağırma = \frac{TP}{TP + FN} \tag{3.56}$$

Hassasiyet: Gerçek doğru olarak tahmin edilen doğru verilerin sayısının, toplam doğru olarak tahmin edilen veri sayısına oranı olarak hesaplanır.

$$Hassasiyet = \frac{TP}{TP + FP} \tag{3.57}$$

f_1 puanı: İki metrik değerlerini birleştirerek karma bir değerle değerlendirilmek istenirse f_1 puanı kullanılabilir.

$$f_1 = 2 * \frac{Geri\ Çağırma * Hassasiyet}{Geri\ Çağırma + Hassasiyet} \tag{3.58}$$

4. DENEYSEL ÇALIŞMALAR

4.1. Metin Madenciliği Yöntemi İle Meslek Analizleri

4.1.1. Metin Madenciliği İle IPA Raporlarının Analiz Edilmesi

Çalışmada, IPA raporları ntlk kütüphanesi kullanılarak metin madenciliği teknikleri ile analiz edilmeye çalışılmıştır. Bu analiz süreçleri içerisinde elde edilen veriler frekans analizleri ve yoğunluk grafikleri ile görselleştirilmiştir.

Analizde öncelikle, pdf biçimindeki 200 sayfanın üzerindeki belgenin içeriği geliştirilen uygulama ile okunmuştur. Elde edilen içerik kelimelere ayırma yöntemleri kullanılarak, kelimelere ayrıştırılmıştır. Yani içerikte geçen tüm kelimeler, bir dizi olarak ifade edilmiştir.

Ayrıştırılan kelimelerin içerisinde, analiz, dil ve içerik açısından anlam ifade etmeyen, bağlaç vb. kelimelerle birlikte, noktalama işaretleri, Türkçe diline özel, stopwords listeleri ile filtrelenmiştir.

Elde edilen kelime dizisi için frekans analizleri çalıştırılmış, gruplara ayırararak ve toplu olarak frekans analizi grafikleri çıkarılmıştır.

Çalışmanın konusuna yönelik olarak, meslek analizlerini yapabilmek için ISCO meslek sınıflandırma belgesi, sisteme excel dosyasından okunarak aktarılmıştır [137]. Frekans analizlerine göre en büyük değeri alan kelimelerin, hangi meslek sektör grubunda olduğu bilgisi, meslek sınıflandırma belgesinden bulunmaktadır. Grafikte üçüncü boyut olarak kullanılan meslek sektör grubu ile yoğunluk grafikleri oluşturulmuştur.

4.4.1.1. Tokenlaştırma

Tokenlaştırma işleminde, analiz edilen metni oluşturan kelimeler bulunur. Veri bilimi algoritmalarında iki tür tokenlaştırma işlemi ile karşılaşılır:

1. **Kelime Tokenlaştırma:** Kelime tokenlaştırmada, metni oluşturan tüm kelimeler bir dizi olarak elde edilir. Bu dizi içerisinde aynı kelimeler,

metinde geçmiş olduğu sırada yer alır. Tokenlaştırma işlemi, veri analizi yapılan konuyla ilgili kelimeler hakkında bilgi edinmek amacıyla yapılır. Özellikle frekans analizleri ve metin içerisinde önemli, anahtar bilgiyi edinmek bakımından, kelimeler önem arz etmektedir.

Metni kelimelere ayırmaya örnek vermek gerekirse, "Açık işi olan mesleklerde engelli çalıştırılmak istenmesi durumu incelendiğinde; yüzde 9,8 oranında evet denilmiştir.", cümlesinin tokenları, ["Açık", "işi", "olan", "mesleklerde", "engelli", "çalıştırılmak", "istenmesi", "durumu", "incelendiğinde", ";", "yüzde", "9,8", "oranında", "evet", "denilmiştir","."] dizisinde gösterilmektedir.

Çalışma kapsamında, tokenlara ayırma işlemi python programlama dilinin ntlk kütüphanesi kullanılarak gerçekleştirilmiştir.

2. **Cümle Tokenlaştırma:** Metin içerisinde geçen kelimelerin isim mi, fiil mi, nesne mi, vb. mi, olduğunun anlaşılması için kelimenin, cümle içinde kullanımına göre incelenmesi gerekmektedir. Bu nedenle, Lematizer, Pos-Tagger gibi süreçler için metnin cümlelerine ayrılması gerekliliği bulunmaktadır.

   Cümle tokenlaştırmaya örnek vermek gerekirse, "Açık işi olan mesleklerde aranan beceriler incelendiğinde, en fazla "fiziki ve bedensel yeterlilik" ön plana çıktığı, daha sonra "yeterli mesleki/teknik bilgi ve tecrübe" geldiği görülmektedir. "İletişim ve ifade yeteneği" ile "takım çalışması" diğer en fazla aranan becerilerdir." paragrafına cümle tokenlaştırma uygulandığında, ["Açık işi olan mesleklerde aranan beceriler incelendiğinde, en fazla "fiziki ve bedensel yeterlilik" ön plana çıktığı daha sonra "yeterli mesleki/teknik bilgi ve tecrübe" geldiği görülmektedir.", ""İletişim ve ifade yeteneği" ile "takım çalışması" diğer en fazla aranan becerilerdir.""] dizisi elde edilir.

4.4.1.2. Filtreleme

Metin üzerinde yapılan tokenlara ayırma işlemlerinde, metinde geçen ilgili ilgisiz, bağlaç, noktalama işaretleri ve özel karakterlerin tümü, elde edilen token dizisinde yer alır. Bu tokenlar analiz açısından bir öneme sahip olmamakla birlikte, belgede yüksek frekansta yer almaları nedeniyle, analizin yanlış sonuçlar üretmesine neden

olmaktadır. Bu nedenle, çeşitli filtreleme süreçleri ile token dizisinden ayırt edilirler. Bahsedilen filtreleme işlemi, ntlk kütüphanesinde, Türkçe diline özel, stopwords olarak adlandırılan kelime veritabanı kullanılarak gerçekleştirilir.

Frekans analizinde, kelimenin metinde hangi sayıda yer aldığı hesaplanır. Metinde aynı kelimeler, büyük, küçük harfli olarak, farklı şekilde geçebilir. Bu nedenle frekans analizi sürecinden önce, tokenlara ayırma işlemi sonucunda elde edilen kelimeler küçük harfe çevrilirler.

4.4.1.3. Kök Bulma (Stemmer)

Metinleri dosya veya farklı kaynaklardan elde ettikten sonra, analizlere konu olacak kelime verilerini, bir ön inceleme safhası ile temizlemek gereklidir. Metin temizlemede kullanılacak analizler arasında, basit, hızlı ve etkin olması açısından önemli olan kök bulma analizi dikkat çekmektedir. En çok bilinenleri Porter, Snawball, Lancaster olmak üzere birçok kök bulma algoritması vardır [138].

Kök bulma analizinde karşılaşılan kelime, tek bir kök kelimeye dönüştürülür. Kelimenin aldığı ekler temizlenerek kök kelime bulunur. Kısacası bir çok kelime tek bir kelimeye dönüştürülür.

Kök bulma işlemine örnek vermek gerekirse; “okuldakilerden” kelimesi için kök bulma çalıştırılınca “okul” kelimesi elde edilir. “işyerindekilerden” için “işyeri”, “sektörleri” için “sektör” vb. kelimeler elde edilir.

Türkçe sondan eklemeli bir dil olduğu için, kelimelerin sonuna eklenen çeşitli eklerle farklı kelimeler oluşturulabilmektedir. Bu durum frekans analizinde aynı kelimenin farklı isimlerle yer almasına, frekans sayısının düşmesine ve grafikte önem değerinin doğru gözükmemesine neden olmaktadır. Bu sorunu gidermek için, doğal dil işlemede kök bulma algoritmaları kullanılmaktadır.

Türkçe’de kelimeler, genelde bir kök ve buna eklenen en az iki üç tipte ekten oluşur. Türkçe’de ekler, kelime köklerine belirli bir sıralamaya göre eklenir. Türkçe’nin sondan eklemeli bir dil olması, doğal dil işlemede önemli bir yere sahip olmasını sağlamaktadır. Türkçe’nin fonetik özelliği, bu faktörü pekiştirmektedir.

IPA raporu belgesindeki kelime tokenları, Türkçe dilbilgisinin bu ve benzeri kuralları göz önüne alınarak geliştirilen TurkishStemmer kütüphanesi [139] kullanılarak, frekans analizine tabi tutulmadan önce, kelimeler kök bulma sürecine tabi tutulmuştur.

4.4.1.4. Lemmatizer

Kök bulma yönteminde, kelimelerin anlamından bağımsız olarak, sonlarındaki ekler atılarak kök bulunmaya çalışır. Bu yöntem, hız avantajı sağlamakla birlikte, çoğu zaman doğru sonuç vermez [140].

Bu noktada, lemmatizer algoritmaları imdada yetişir. Lemmatizer algoritmaları, kelimeleri, isim, yüklem ve sıfat olarak sınıflandırarak, anlamlandırmaya çalışırlar. Bu nedenle lemmatizer, kök bulmaya göre daha fazla dilbilgisi ve hesaplama gerektirir.

Diğer bir deyişle, lemmatizer, kelimeyi, sözlük anlamına çevirmeye çalışır. Kelimenin anlamlandırma sürecinde, kelimenin metnin hangi cümlesinde, ne şekilde geçtiği önemlidir. Bu da etiketleyiciler (tagger) gibi daha güçlü dilbilimsel özellikler gerektirmektedir.

Yeni bir dilde lemmatizer gerçeklemek, kök bulma gerçeklemeye göre daha çok zaman alır. Lemmatizer için, dilin yapısal özelliklerine hakim olmak, özne, yüklem, nesne vb. cümle öğeleri ilişkilerine göre, kelimeleri etiketleyebilecek algoritmaları geliştirmek gerekir.

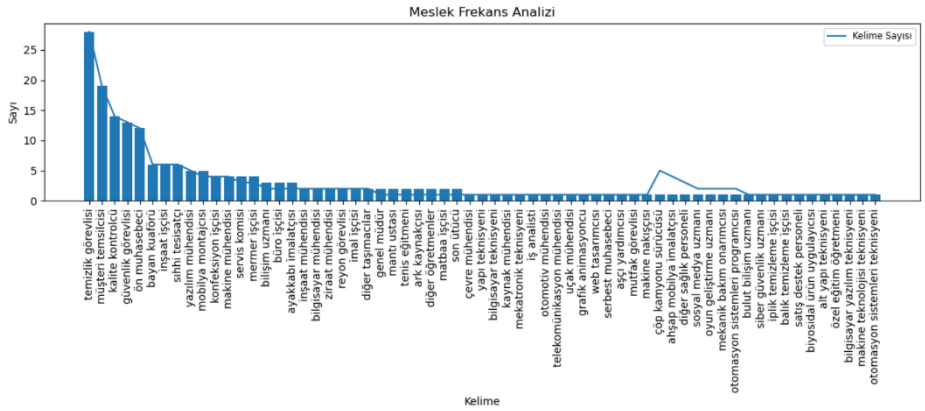
İngilizce dili için lemmatizer algoritması çeşitli kütüphanelerde yer almakla birlikte, Türkçe gibi sondan eklemeli bir dilde, lemmatizer algoritması geliştirmek daha fazla bilgi ve zorlu bir süreç gerektirir [141].

Lemmatizer işlemi sonucunda, “işgücü”, “iş” kelimesine, “açısından” “açı” kelimesine, “dönüşüm”, “dönüş” kelimesine dönüşür. Örneklerden görüldüğü üzere, bazen çalışma konusu ile ilgili kelimeleri bozabiliyor. Bununla birlikte çoğu zaman kök bulmaya göre daha anlamlı kelimeler bulur.

4.4.1.5. Frekans Analizi

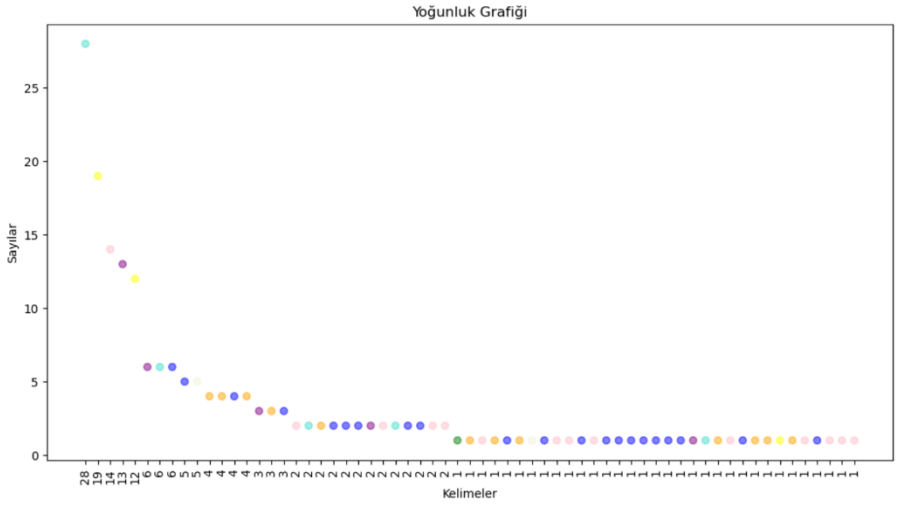
Frekans analizi aşamasında, token dizisinde hangi kelimenin ne kadar geçtiği bilgisi bulunmaya çalışılır. Token dizisi içerisinde daha fazla geçen kelimelerin, metnin konusu ile ilgili bilgileri içermesi beklenir. Çalışmada frekans analizi sonucunda elde edilen frekans dizisinin grafiksel gösterimi, Şekil 4.1’de görülmektedir. Grafik

gösterimi yapılan frekans dizisi, büyükten küçüğe doğru sıralandağı için, x ekseni boyunca, y değerleri azalarak sıfıra doğru giden bir eğri görülmektedir.



Şekil 4.1. Frekans Analizi Sonuçları Grafiği

Çalışmada, frekans analizine, mesleklerin sektörel sınıflandırması, üçüncü boyut olarak eklenerek, dağılım grafiği yöntemiyle, yoğunluk grafiği oluşturulmuştur; elde edilen grafiğin gösterimi, Şekil 4.2'de yer almaktadır.



Şekil 4.2. Yoğunluk Dağılım Grafiği

4.4.1.6. Pos-Tagger (A Part-Of-Speech Tagger)

Herhangi bir dille yazılmış olan metnin, tokenlarını bulduktan sonra, kelimelere, cümlede aldıkları göreve göre, isim, fiil, nesne, zamir vb. öğe etiketleri atayan algoritmalardır [142].

Pos-tagging' le ilgili olarak literatürde birçok algoritma olmakla birlikte, en iyi performans gösteren algoritmalar, gizli Markov modeli [143], maksimum entropi

yaklaşımı [144], ve dönüşüm tabanlı öğrenmedir [145]. Bu üç yöntemde, hemen hemen aynı veri kaynakları, aynı yöntemler ve özellikleri kullandıkları için, aynı seviyede başarım verirler. Oysaki Samuelsen ve Voutilainen (1997) tarafından geliştirilen EngCG algoritması, leksikal analizler, bağlamsal bilgi kaynakları ve istatistiksel etiketleyiciler tarafından kullanılanların ötesinde genellemeler kullanarak, diğer etiketleyici algoritmalarına göre, çok yüksek bir başarım sağlamıştır [146].

4.4.1.7. Kollokasyon

Kollokasyon, metin içerisinde sıklıkla geçen bir dizi veya kelime grubunu bulmak için işletilen bir süreçtir. Bu analiz sonucunda, metin içerisinde sıklıkla karşılaşılan kelime grupları ve frekans değerleri listelenir. Sıklıkla kullanılan kollokasyon analizleri, n-gram, bigram, trigram olarak listelenebilir.

Bu analizin çalışma açısından önemi, metin içerisinde aranan meslek isimlerinin, birden fazla kelimeden oluşabilmesidir. Tokenlaştırma işlemi sonucunda uygulanan frekans analizi, tek kelimeden oluşan frekans dizisi oluşturur. Bu durumda, birden fazla kelimeden oluşan meslek isimlerini, metinden çıkarım yapmak mümkün olmamaktadır. Bu noktada, kollokasyon yöntemleri imdada yetişmektedir.

4.4.1.8. Kelime Bulutu (Word Cloud)

Şekil 4.3. Kelime Bulutu

Kelime bulutu gösteriminde, metinde geçen kelimeler bulunma sıklıkları, yani frekanslarına göre daha büyük punto ve dikkat çekici renkte gözükmektedir. Bir kelimenin puntosunun büyüklüğü ve dikkat çekici renkte olması, kelimenin metinde daha fazla karşılaşıldığını gösterir.

4.1.2. Metin Madenciliği İle Öğrenme ve Tahmin Çalışması

Literatürde son yıllarda yapılan çalışmalar incelendiğinde, arama motorlarının veri madenciliği ve metin madenciliği konusunda ihtiyaç duyulan veri kümelerini oluşturmak için, çok kullanışlı web araçlarına dönüştüğü görülmektedir [147], [148], [149], [150]. Mesleklerle ilgili yapılan araştırmalarda, güncel durumu ortaya koymak açısından, İş Kurumu tarafından yayınlanan IPA raporları bulunmaz bir kaynaktır. Bu nedenle, google arama motoru üzerinde "IPA raporu" ve "Meslekler" arama metinleri ile bulunan belgeler, araştırmaya konu olarak seçilmiştir. Internet üzerinden araştırma kapsamında geliştirilen uygulama ile indirilen belgeler metin madenciliği teknikleri ile analiz edilmiştir.

Internet üzerinden veri hazırlama uygulamasında, "google search" kütüphaneleri kullanılarak arama kriterlerine göre belgeler bulunmuştur. Bulunan belgeler http isteği ile indirilmiştir. Belge biçimi pdf ise, PDFMiner phyton kütüphanesi kullanılarak pdf dosyadan metin çıkarım yöntemiyle içerik elde edilmiştir. Html biçimindeki dosyaları okumada, html etiketlerini ayıklayarak gerçek metne ulaşmak için, Beatifulsoap yönteminden faydalanılmıştır. Beatifulsoap html biçimindeki dosyalardan, veri kazıma teknikleriyle veri çıkarımı yapan Phyton dili kütüphanesidir. Veri hazırlama uygulamasının şematik gösterimi, Şekil 4.4'de verilmiştir.

Şekil 4.4. Veri Hazırlama Uygulaması

Çalışma kapsamında elde edilen 151 adet belgenin 76 adeti öğrenme aşamasında kullanılırken, 75 adeti tahmin için kullanılmıştır. Sınırlı sayıda veri elde edilmesi nedeniyle eğitim ve tahmin veri kümeleri, yeteri kadar tahminleme yapılabilmek için yaklaşık olarak eşit sayıda seçilmiştir. Belgelerde geçen meslekler ISCO (International Standard Classification of Occupations) meslek sınıflandırmasındaki, Tablo 4.1'de verilen, sektör sınıflandırmasına göre sınıflandırılmıştır. Frekans listelerindeki meslek isimlerine göre ISCO meslek listesinden sektör sınıflandırması kullanılmıştır. Frekans listesinde yeralan meslek isimleri modele değişken olarak verilmiştir. "IPA raporu" ve "Meslekler" belgeleriyle yapılan çalışmada 2735 değişken kullanılmıştır. Modelde, belgelerde bulunan meslekler değişken olarak kullanılmıştır. Model giriş matrisindeki değişken değeri, belgede mesleğin bulunma sayısı olarak atanmıştır.

Tablo 4.1. Meslek Sınıflandırma Listesi

Sınıfı	Meslelek Sınıf Adı
1	Silahlı kuvvetlerle ilgili meslekler
2	Yöneticiler
3	Profesyonel meslek mensupları
4	Teknisyenler, teknikerler ve yardımcı profesyonel meslek mensupları
5	Büro hizmetlerinde çalışan elemanlar
6	Hizmet ve satış elemanları
7	Nitelikli tarım, ormancılık ve su ürünleri çalışanları
8	Sanatkarlar ve ilgili işlerde çalışanlar'
9	Tesis ve makine operatörleri ve montajcılar
10	Nitelik gerektirmeyen işlerde çalışanlar'

IPA raporlarından geleceğin meslekleri ile ilgili çıkarım yapma olasılığı çok düşüktür. Bu nedenle google arama motorunda "Geleceğin Meslekleri" arama metni ile ulaşılan belgeler üzerinde çalışılmıştır. Mesleklerin mevcut durumlarının analizinde daha çok pdf türünde belgeler kullanılırken, geleceğin meslekleri çalışmasının belgeleri çoğunlukla html türündedir. Geleceğin meslekleri konusunda yapılan analizler bilgi teknolojisi alanındaki işlerde yoğunlaşmıştır. Çalışma kapsamında elde edilen 71 adet belgenin, 36 adeti öğrenme aşamasında kullanılırken, 35 adeti tahmin için kullanılmıştır. Geleceğin meslekleri YÖK (Yüksek Öğretim Kurumu)' ün açıklamış

olduğu öncelikli meslek gruplarına göre sınıflandırılarak, makine öğrenmesi algoritmaları ile modellenmiştir. Sınıflandırmada kullanılan sınıflar, Tablo 4.2'de listelenmiştir. "Yönetim ve Organizasyon", "Akıllı Şehir", "Etik Uzmanlığı" ve "Diğer" sınıfları, YÖK'ün öncelikli meslek alanları ile sınıflandırılamayan meslekleri, sınıflandırmak için eklenmiştir. Modelde 117 adet değişken kullanılmıştır.

Tablo 4.2. Meslek Sınıflandırma Listesi

Sınıfı	Meslek Sınıf Adı
0	Ağ teknolojileri (5G, Nesnelerin İnterneti)
1	Akıllı Enerji Sistemleri
2	Akıllı ve Yenilikçi Malzemeler
3	Biyomalzeme ve Doku Mühendisliği
4	Biyomedikal ve Biyomedikal Teknolojiler
5	Blokzincir Teknolojisi
6	Cebir ve Kodlama Teorisi
7	Endüstri Mühendisliği (Yöneylem Araştırması; Tedarik Zinciri Yönetimi)
8	İklim Değişikliği
9	İleri Robotik Sistemler ve Mekatronik
10	İleri ve Akıllı İmalat
11	Kuantum Enformasyon ve Kuantum Makina Öğrenmesi
12	Sanal ve Artırılmış gerçeklik teknolojileri
13	Siber Güvenlik/Kriptoloji
14	Sistem Mühendisliği
15	Su Ürünleri ve Balıkçılık Teknolojisi
16	Sürdürülebilir Tarım (Yenilikçi ve İyi Tarım Uygulamaları dahil)
17	Sürdürülebilir ve Akıllı Ulaşım
18	Uzaktan Algılama ve Coğrafi Bilgi Sistemleri
19	Veri Bilimi ve Bulut Bilişim
20	Yakıtlar (Fosil ve Biyo) ve Yanma
21	Yapay Zeka ve Makine Öğrenmesi
22	Yazılım Mühendisliği
23	Yenilikçi Gıda İşleme Teknolojileri ve Gıda Biyoteknolojisi
24	Bilişim Hukuku
25	Eğitimde Dijitalleşme
26	Hemşirelik
27	Moleküler Biyoloji ve Genetik (Gen tedavisi ve Genom Çalışmaları)
28	Rehabilitasyon Tıbbı ve Yardımcı Teknolojiler
29	Rejeneratif Tıp
30	Sağlıklı Beslenme ve Gıda Katkı Maddeleri
31	Yaşlanma ve Yaşlı Sağlığı
32	İnsan Robot İletişimi
33	Sosyal Medya
34	Yönetim ve Organizasyon
35	Akıllı Şehir
36	Etik Uzmanlığı
37	Diğer

Metin madenciliğinde frekans matrisi oluşturma sürecini, veri temizleme süreci takip eder. Veri temizleme süreci fazla boşlukların temizlenmesi, noktalama işaretleri, bağlaç ve edatların kaldırılması, küçük harfe çevirme ve kökten türetme algoritmasını içerir. Çalışmada bilgi kaybına neden olduğu için, kökten türetme algoritması dışındaki tüm süreçler kullanılmıştır.

4.1.2.1. Önerilen Model

Önerilen yöntemde internet ortamından Şekil 4.5'de gösterilen Veri Hazırlama adımı ile elde edilen dosyalar, frekans analizine tabi tutulmuştur. Frekans analizi ile belge içerikleri kelimelerine ayrılmış ve metinde kelimelerden kaçar adet geçtiği belirlenmiştir. Birden fazla kelimeden oluşan meslek isimleri için, n-gram algoritmaları ile ikili (bigram), üçlü (trigram), dörtlü (fourgram) kelime grupları kullanılarak frekans analizi hesaplanmıştır. Kelime grupları metinde yan yana bulunan sözcüklerden oluşmaktadır. Veri Temizleme adımında, kelimelerine ayrılan metinlerde geçen konuyla ilgisiz kelimeler ve alfa karakterler, Türkçe diline özel stopwords veri kümesi ve alfa karakterler kontrolü ile elenmiştir. Metinde geçen meslek isimleri, ISCO meslek listesi ile filtrelenerek elde edilmiştir. Bu frekans grupları meslek listesi ile filtrelenerek metinde geçen meslekler belirlenmiştir. Makine öğrenmesi veri giriş değerleri Bag of Words Vektör Oluşturma adımında matris olarak oluşturulmuştur. Makine öğrenmesi algoritmaları ile üretilen tahminler, ölçüm metrikleri ile değerlendirilmiştir.

Önerilen yöntemin bilgi tabanı matematiksel olarak $\mathcal{K} = (\mathcal{T}, \mathcal{D})$ çifti ile ifade edilsin. Bu modelde, $\mathcal{T} = (t_1, t_2, \dots, t_m)$ olarak meslek isimlerini ifade ederken, $D = (d_1, d_2, \dots, d_n)$ belgeleri ifade eder. Bu ifadelerin ışığında, model bilgi tabanının frekans matrisi, $A \in \mathbb{R}^{m \times n}$ için $A = (a_{ij})$ olarak gösterilir. t_i mesleğinin D_j belgesindeki frekansı, D_j belgesinde ne kadar geçtiği olarak hesaplanır ve $a_{ij} = |\{p | t_{jp} = t_i\}|$ olarak gösterilir. Meslek sınıflarına göre ise, belgede $a_{ij} = |\{p | t_{jp} = mesleksinif(t_i)\}|$ olarak hesaplanır.

Model çıkış değeri meslek sektör frekans analizi kullanılarak şu şekilde hesaplanır. $B = (b_1, b_2, \dots, b_n)$ frekans matrisinin bir satırı olma koşuluyla, y meslek sınıfı değeri,

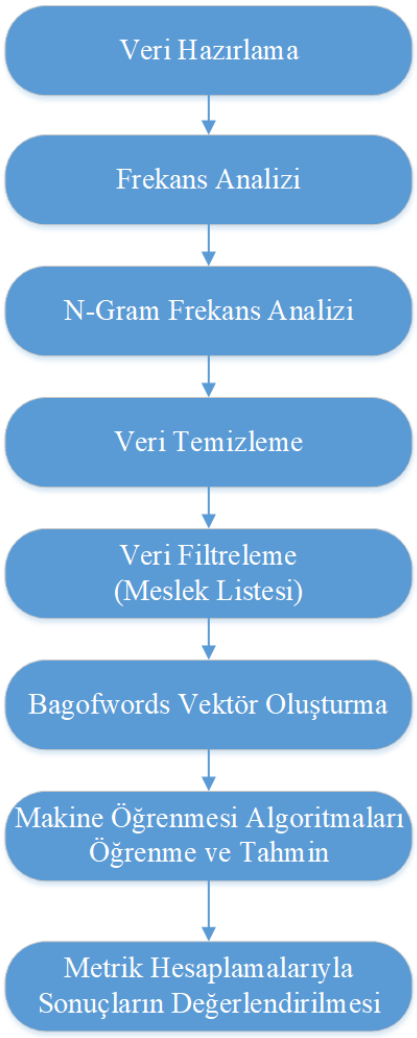
$$y = \frac{\sum_{i=0}^{n} i \times b_i}{\sum_{i=0}^{n} b_i} \quad \textbf{(4.1)}$$

olarak hesaplanır.

Makine öğrenmesi algoritması giriş ve çıkış değerleri, mesleklerin belgelerdeki geçiş frekans değerlerine göre, kelime torbası vektörü olarak düzenlenmiş ve Tablo 4.3'te gösterilmiştir. Frekans analizi ile tüm belgelerden elde edilen kelimeler, bir matris içerisinde değişken bir sayı ile etiketlenmiştir. Kelime torbası vektörü oluşturmada, x' in değişken değeri olarak, kelimenin belgede geçmesi durumunda, kelimenin belgede geçiş sayısı, aksi durumda 0 olarak atanır.

Table 4.3. Bag Of Words Modeli Giriş Matrisi

Belge	Muhasebeci	Temizlik Personeli	Müşteri İlişkileri	Ön Muhasebeci	Şef	Otomasyon Sistemleri	y
Doc 1	30	28	19	12	17	1	6
Doc 2	32	25	4	28	6	0	6



Şekil 4.5. Metin Madenciliği Öğrenme Algoritması.

Algoritma 4.1. Önerilen Model

Giriş: *N* belge , meslek listesi, *M* ml_algoritması, *K* meslek sınıfı
for *i* = 1,2, … , N **do**
wt ← tokenize(*docs(i)*)
wt ← eliminate_lambda_stopwords(wt)
*freq_lis*t ← frequency_analysis (wt)
analysis(i) ← (*docs(i)*, freq_list)
total_frequency_list ← calculate_total_frequency(*freq_list*)
profession_list ← filter_with_occupation_list (*total_frequency_list, occupation_list*)
(*learn_analysis_list , test_analysis_list*) ← split_data_into_two_set (*analysis*)
Splitting data into two set as odd, even, fifty, fifty
for *i* = 1,2, … , *learn_analysis_list.count* **do**
frequency_list ← *learn_analysis_list(1)*
for *j* = 1,2, … , *frequency_list.count* **do**
index ← index_of *(frequency_list(i)(0), occupation_list)*
word_vector(index) = word_vector(index) + frequency_list(i)(1)
bag_vector(i) ← *word_vector*
learn_bag_vector ← *bag_vector*
test_bag_vector ← generate_bag_vector (*test_analysis_list*)
Test bag vector was generated like learn_bag_vector calculation.
for *i* = 1,2, … , *learn_bag_vector.count* **do**
bag ← *learn_bag_vector(i)*
for *j* = 1,2, … , *bag.count* **do**
profession_name ← *profession_list(j)*
profession_class ← find_profession_class **(***profession_name, occupation_list)*
profession_class_count(profession_class) ← *profession_class_count(profession_class) + bag(j)*
toplam1 =0
toplam2 =0
for *k* = 1,2, … , *profession_class_count.count* **do**
toplam1 ← *toplam1 + (k+ 1) * profession_class_count (k)*
toplam2 ← *toplam2 + profession_class_count(k)*
y_train(i) ← *toplam1 / toplam2*
y_test = y_test_from_bag_vector(test_bag_vector)
Test output value were calculated like *y_train*.
for *i* = 1,2, … , *ml_algorithm.count* **do**
learn (*learn_bag_vector, y_train*)
y_pred ← predict (*test_bag_vector, y_test*)
evaluate_metrics (*y_test, y_pred*)
confusion_matrix (*y_test, y_pred*)

Öğrenme algoritması sonuçları ölçüm metrikleri ile tutarlılık ve doğruluk analizlerine tabi tutulur.

Önerilen modelin ayrıntılı algoritması, Algoritma 4.1.den görülebilir. Bu algoritmada, öncelikle elde edilen belgeler kelimelerine ayrıştırılmıştır. Stopwords ve alfa karakterler, kelime dizisinden çıkarılmıştır. Daha sonra tüm belgelerin frekans listesi hesaplanmış ve her bir belge için analiz koleksiyonlarına kaydedilmiştir. Tüm belgelerde karşılaşılan meslekler, meslek listesi ile toplam sıklık listesi filtrelenerek bulunmuştur. Daha sonra analiz verileri öğrenme ve test olmak üzere iki veri kümesine ayrılmıştır. Algoritmanın giriş değerleri olan torba vektörleri, her bir belgedeki mesleklerin sıklığının toplanması ve meslek listesinde karşılaşılan mesleklerin indeksinin belirlenmesi şeklinde hesaplanmıştır. Torba vektörünün çıktı değeri, meslek sayısı ve sınıf indeksinin çarpımlarının toplamının sınıf indeksi toplamına bölünmesiyle hesaplanmıştır. Bu işlem hem öğrenme hem de test veri setleri için gerçekleştirilmiştir. Torba vektörlerine öğrenme ve tahmin sırasına göre makine öğrenmesi algoritmaları uygulanmıştır. Son olarak, elde edilen sonuçlar ve algoritmaların performansı, ölçüm metrikleri ve karışıklık matrisi ile değerlendirilmiştir.

Çalışma kapsamında önerilen modelde kullanılan makine öğrenme algoritmalarının işlevlerini yerine getirmesi açısından önemli olan parametre değerleri, aşağıda listelenmiştir:

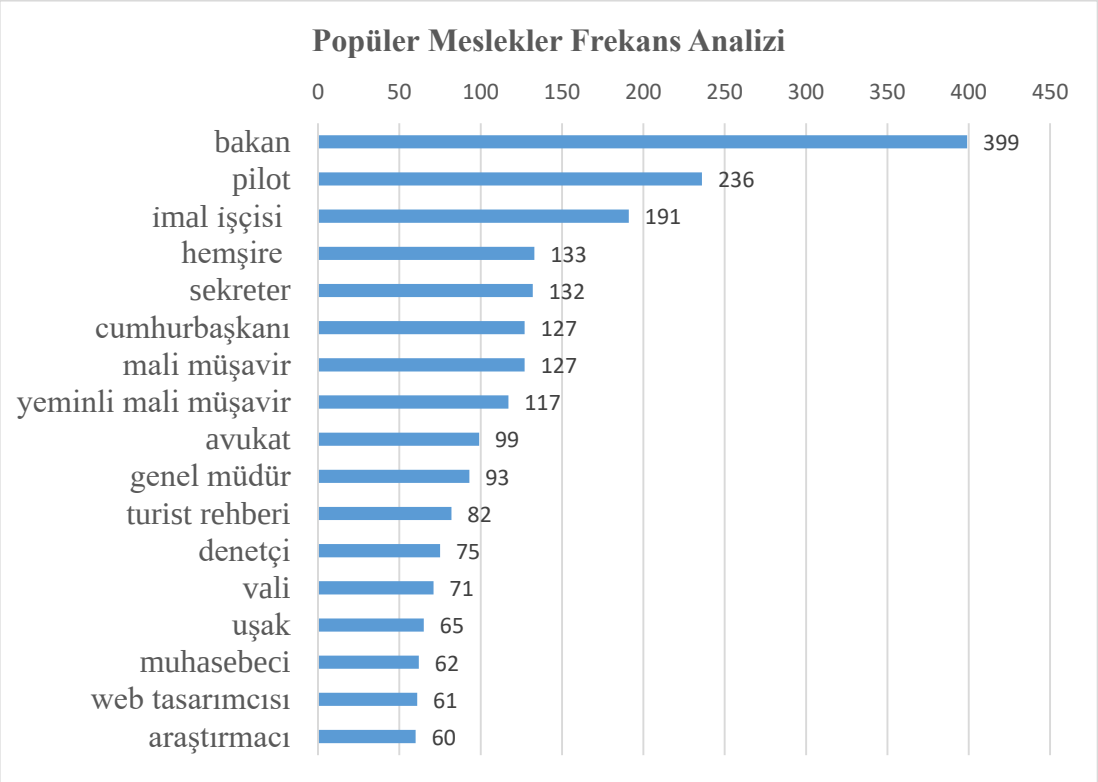
Knn algoritmasında sınıflandırma için önemli olan komşu sayısı parametresi 3, yaprak boyutu, ağaç ve sorgunun yapım hızı 10 olarak alınmıştır. Naive bayes algoritmasında kararlılığı sağlamak için varyanslara, 1e-9 değerinde bir yumuşatma parametresi eklenir. Rastgele orman algoritmasında, 10 ağaçtan oluşan ormanlar kullanılmıştır. LassoLars algoritmasında ceza terim katsayısı 0,01 olarak kullanılırken, Elasticnet algoritmasında, 1 olarak alınmıştır. SGD algoritmasında durdurma koşulu, 1e-3, maksimum döngü değeri, 1000 olarak alınmıştır. Son olarak, perceptron algoritmasında maksimum döngü sayısı olarak 1000 alınmıştır.

4.1.2.2. Günümüz Mesleklerine Yönelik Analizler

Önerilen yöntem ile “IPA raporu” ve “Meslekler” arama metni elde edilen belgeler, phyton dili ile yazılmış uygulama aracılığı ile indirilmiştir. Bu belgeler üzerinde metin

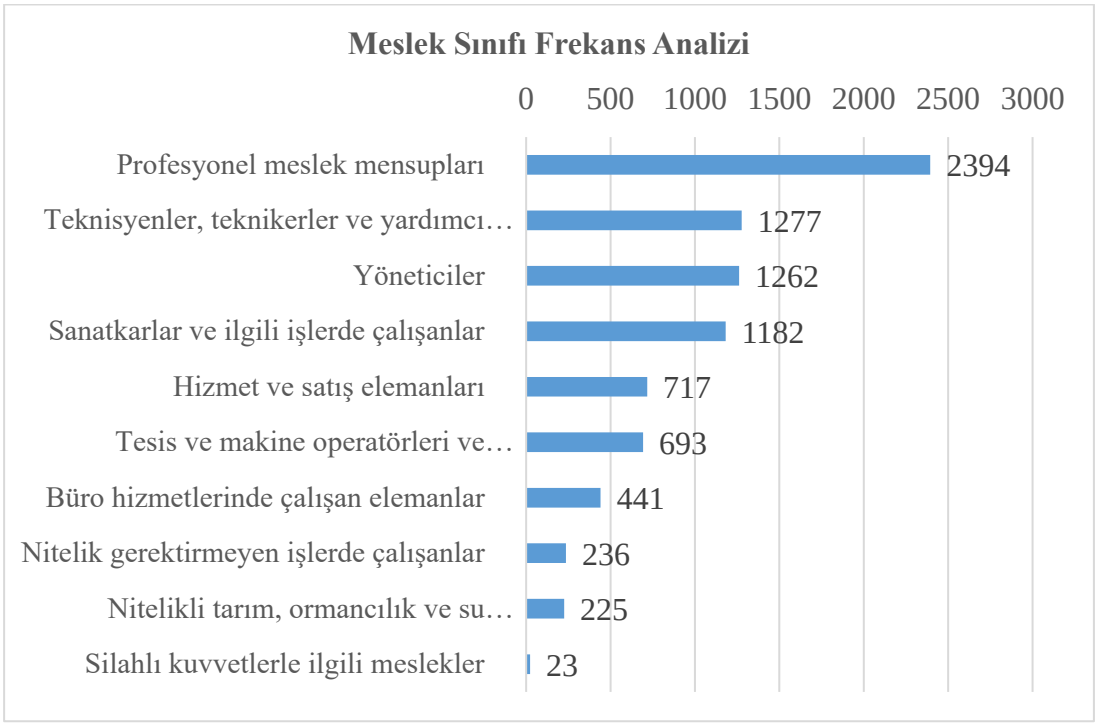
madenciliği süreçlerinden frekans analizi çalıştırılmış, meslekler veritabanında belirtilmiş olan meslekler ile frekans listesi filtrelenmiştir. İnternetten elde edilen 151 belgenin, 76 adedi öğrenme, 75 adedi tahmin verisi olarak kullanılarak, makine öğrenme algoritmaları çalıştırılmıştır. Öğrenme veri kümesi ile eğitilen makine öğrenme algoritmalarından, tahmin veri kümesi ile mesleklerin geleceğine yönelik tahminler üretmesi beklenmiştir.

Analiz kapsamında kullanılan tüm belgelerde geçen meslek isimlerine yönelik olarak oluşturulan, toplu frekans analizi grafiği Şekil 4.6' da gösterilmiştir. Frekans analizi grafiğinde, tüm belgelerde geçen mesleklerden, sayıca fazla değerde bulunan ilk 17' si listelenmiştir. Veriler büyükten küçüğe doğru sıralandığı için, grafik *x* ekseni boyunca sönümlenerek 0'a doğru yaklaşmaktadır.

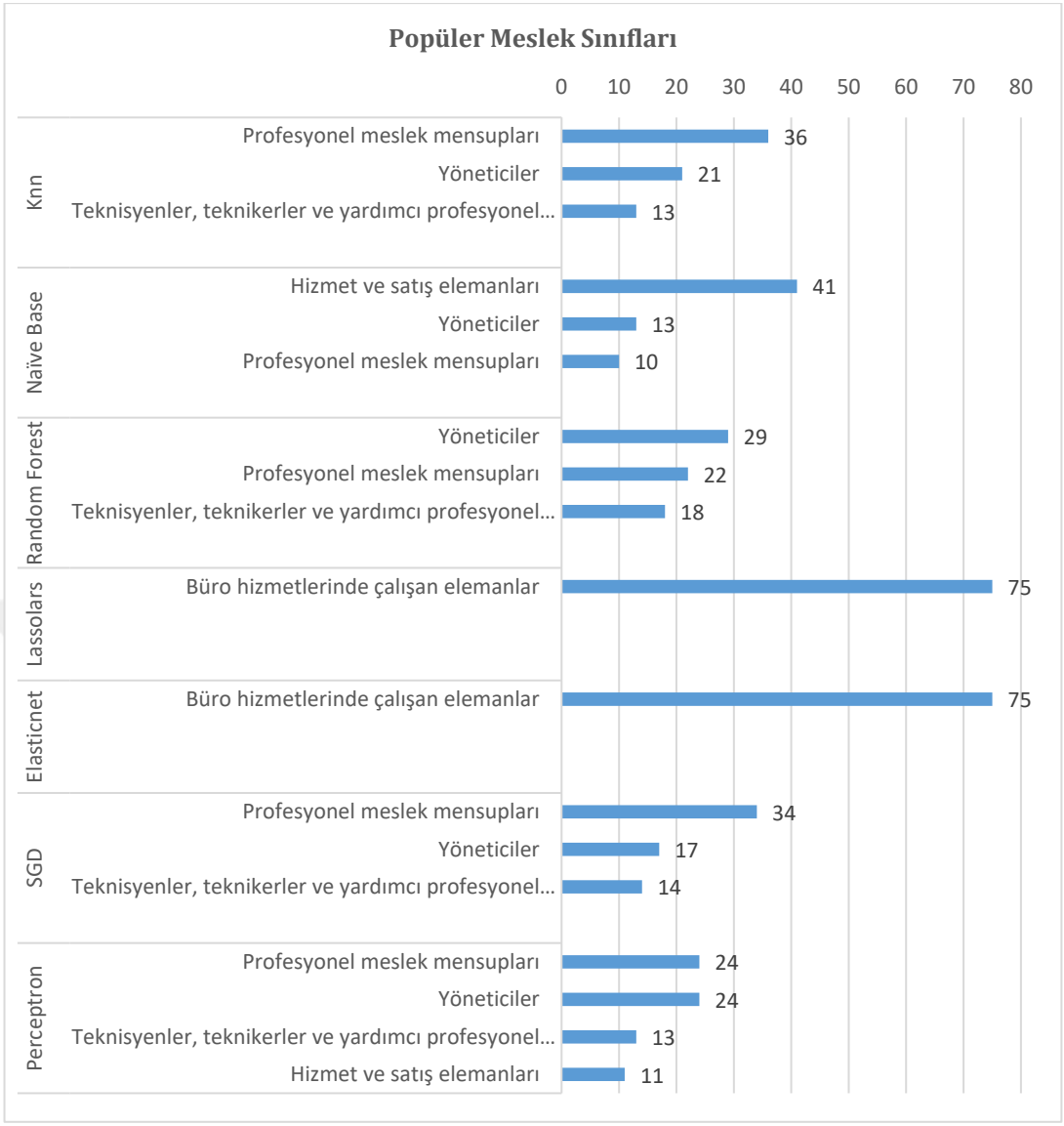


Şekil 4.6. Popüler Meslekler Frekans Analizi

Meslek sınıflarına yönelik toplu frekans analizine göre “Profesyonel meslek mensupları”, “Teknisyenler, teknikerler ve yardımcı profesyonel meslek mensupları”, “Yöneticiler”” ve “Sanatkarlar ve ilgili işlerde çalışanlar” meslek sınıflarının popüler meslek sınıfları olduğu Şekil 4.7’ de görülmektedir. Analiz sonuçlarında kamu mesleklerinin sıklıkla yer alması son dönemlerde artan işsizlikte, kamu görevlerinin cazibesinin arttığını göstermektedir.



Şekil 4.7. Popüler Meslek Sınıfları Frekans Analizi



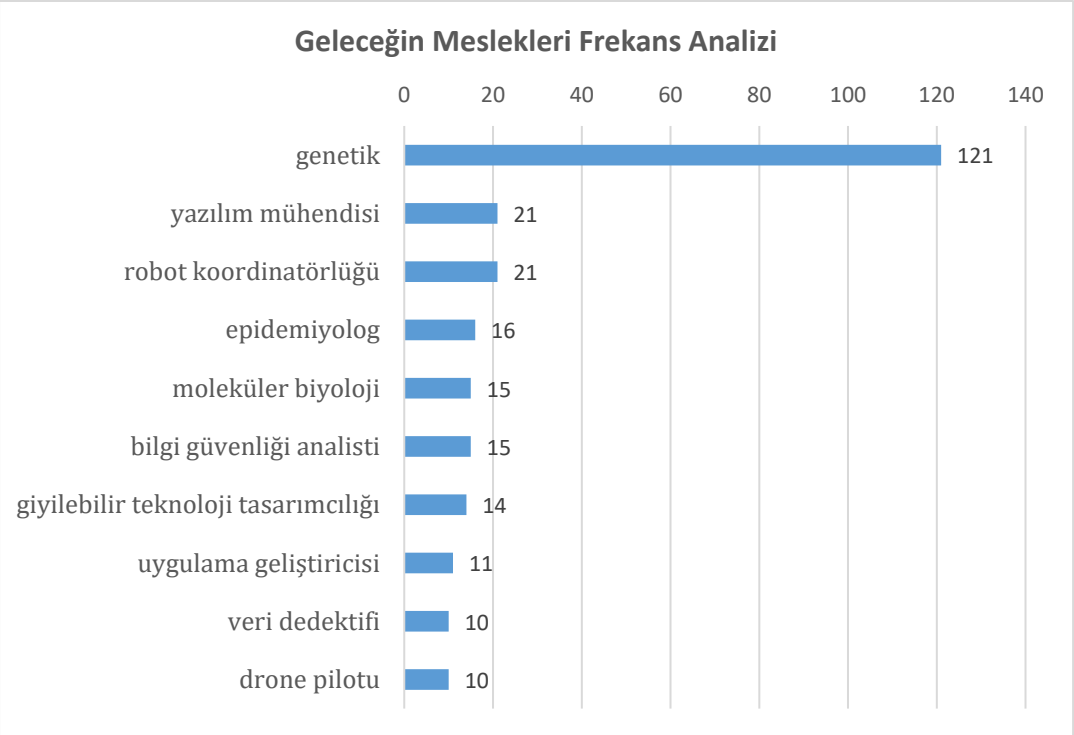
Şekil 4.8. Tahmin Sonuçlarına Göre Popüler Meslek Sınıfları

Makine öğrenmesi algoritmalarına göre tahmin sonuçlarında, "Profesyonel meslek mensupları", "Yöneticiler" ve "Teknisyenler, teknikerler ve yardımcı profesyonel meslek mensupları" meslek sınıflarının ön plana çıktığı Şekil 4.8' de görülmektedir. İş modellerindeki değişimler sonucunda, işleri koordine edecek meslekler ve yüksek yetenek düzeyi gerektiren meslekleri robotların devralamayacağı aşikardır. Tahmin sonuçları da bu önergeyi destekler yöndedir.

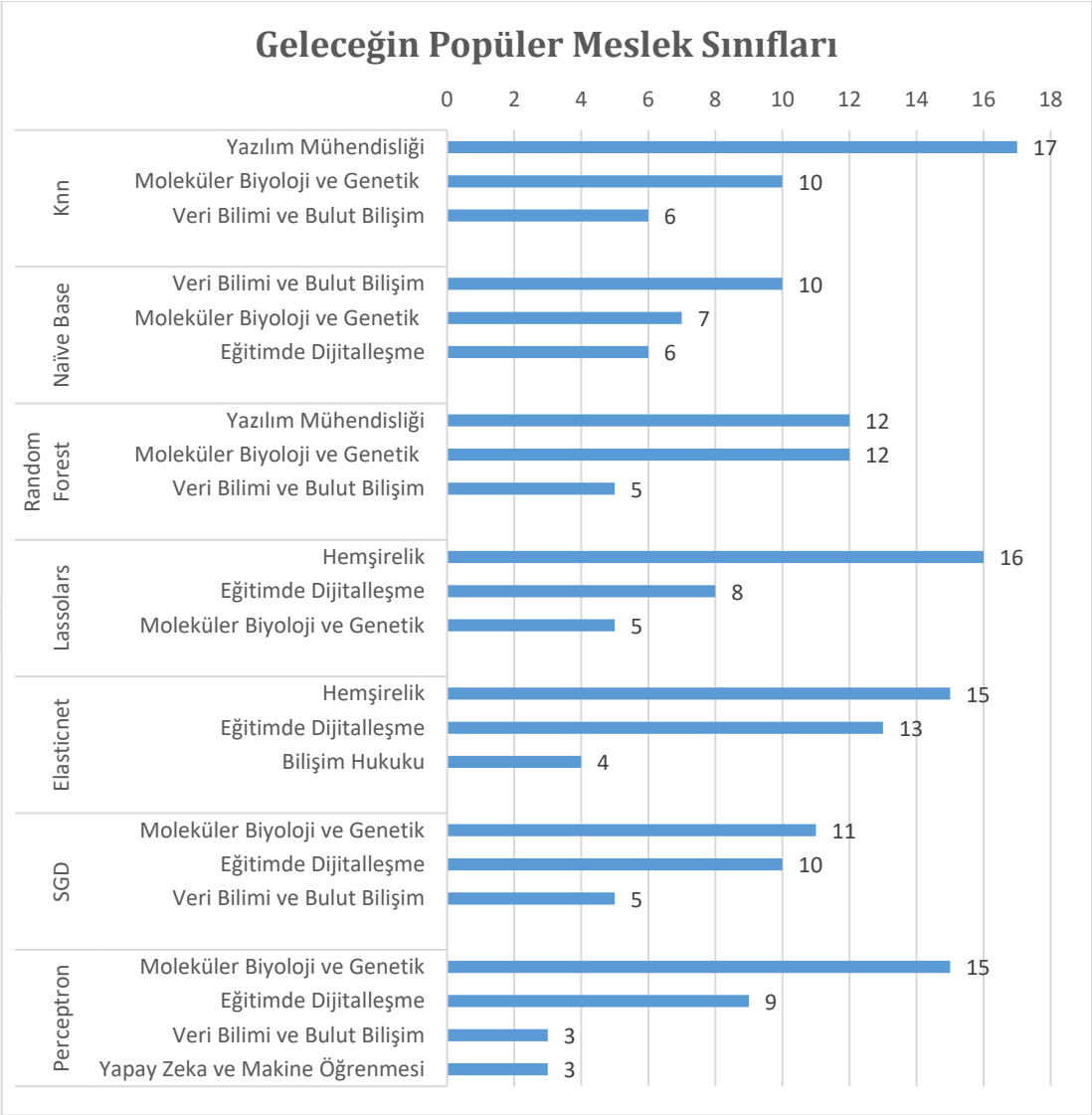
4.1.2.3. Geleceğin Mesleklerine Yönelik Analizler

Çalışmanın bu aşamasında, "Geleceğin Meslekleri" arama metni ile arama yapılarak elde edilen belgelerle analiz süreçleri çalıştırılmıştır. Geleceğin mesleklerine yönelik analiz sürecinde, belgelerin frekans analizleri oluşturulmuş, Özdemir ve Kılınç'ın [151] açıklamış olduğu meslekler listesine, çeşitli meslekler eklenerek oluşturulan meslek listesine göre filtrelenmiştir.

Grafiksel gösterim incelendiğinde genetik mesleği (121 kez), diğer mesleklere göre ciddi bir farkındalık oluşturmuştur. Son zamanlarda tıp alanındaki birçok hastalığa genetik bilimi sayesinde tedavi üretme çalışmaları, bu popülerlikle aynı doğrultuda gözükmektedir. Bilgi teknolojileri mesleklerinin popülerliği, endüstri 4.0 devrimi ile bilgi teknolojileri süreçlerinin, üretim süreçlerinin temel parçası haline geldiğinin göstergesidir. Frekans analizi sonuçlarına göre geleceğin popüler meslekleri Şekil 4.9'da gösterilmiştir.



Şekil 4.9. Geleceğin Meslekleri Frekans Analizi Grafiği



Şekil 4.10. Geleceğin Popüler Meslek Sınıfları

Algoritmalara göre tahmin sonuçlarında, "Yazılım Mühendisliği", "Hemşirelik", ''Moleküler Biyoloji ve Genetik" ve "Eğitimde Dijitalleşme" gibi teknoloji odaklı meslek sınıfları ön plana çıkmaktadır. Algoritma tahmin sonuçlarına göre popüler meslek sınıfları Şekil 4.10' da verilmiştir.

Buraya kadar yapılan çalışmalarda geleceğin mesleği olarak bulunan mesleklerin, YÖK'ün öncelikli meslek grupları [152] öngörüleri ile eşleşmeleri Tablo 4.4' te sunulmuştur. Belirtilmiş eşleşmelere göre uyumlu sonuçlar elde edildiği gözlemlenmiştir. Bu da veri kaynağının yetersiz olmasına rağmen, analiz sonuçlarının doğruluğunun göstergesidir.

Tablo 4.4. YÖK Öncelikli Meslek Alanları ve Meslek Eşleşmeleri.

Öncelikli Meslek Alanları	**Meslek**
İleri ve Akıllı İmalat	Endüstriyel Veri Bilimciler
	Giyilebilir Teknoloji Tasarımcısı
Sürdürülebilir ve Akıllı Ulaşım	Akıllı Şehir Uzmanı
Veri Bilimi ve Bulut Bilişim	Veri Dedektifi
	Veri Analizi Uzmanı
	Kişisel Veri Brokerı
	Endüstriyel Veri Bilimciler
	Bulut Hesaplama Uzmanı
	Yazılım Uzmanı
	Yazılım Geliştirici
Robot Teknolojileri	Robot Koordinatörlüğü
	Robot Koordinasyon Uzmanı
Siber Güvenlik / Kriptoloji	Veri Güvenliği Uzmanı
	Bilgi Güvenliği Analisti
Uzaktan Algılama ve Coğrafi Bilgi Sistemleri	Drone Pilotu
İlaç Çalışmaları	Moleküler Biyoloji
	Biyomühendislik
	Epidimiyolog
Moleküler Farmakoloji ve İlaç Araştırmaları	Moleküler Biyoloji
	Biyomühendislik
	Epidimiyolog
İnsan Beyni ve Nörobilim	Genetik

4.1.2.4. Deneysel Metrik Sonuçları ve Yorumlar

Deneysel ölçüm metrikleri, öğrenme modelindeki tüm algoritmalara uygulanmış, karşılaştırmalı sonuçlar Tablo 4.5. de sunulmuştur. Deneysel metrik sonuçlarına göre SGD algoritması en iyi sonucu üretmiştir. SGD algoritmasının, zeka işlevleri ile donatılmış yapısı ile, kategorik değişkenlerde başarılı sonuçlar üretmesi durumunu açıklamaktadır. Ölçüm metriklerine göre. LassoLars ve ElasticNet algoritmaları en

kötü sonucu üretmiştir. LassoLars ve ElasticNet algoritmalarının regresyon algoritması olması, kategorik değişkenlerde kötü sonuç vermesine neden olmuştur.

Sınıflandırma ile uyumsuz verilerin düzgün belirlenemediği durumlarda doğru sonuçlar üretebilen ortalama mutlak hata, SGD algoritmasında ≅0.64 ile en iyi performansı vermiştir. LassoLars ve ElasticNet algoritması ≅1.52 ile en kötü performansı göstermiştir. LassoLars ve ElasticNet algoritmasının regresyon algoritması olması ve modelin kategorik değişkenler içermesi durumu açıklamaktadır. Ortalama mutlak yüzde hata metriğine göre, yine SGD algoritması ≅0.11 ile en iyi sonucu verirken, ElasticNet ve LassoLars algoritması ≅0.82 ile en kötü sonucu üretmiştir.

Tablo 4.5. Karşılaştırmalı Değerlendirme Sonuçları.

Algoritma	**MAE**	**MAPE(%)**
Knn	0,77	0,15
Naive Base	1,33	0,57
Rand-Forest	0,78	0,16
LassoLars	1,52	0,82
ElasticNet	1,52	0,82
SGD	0,64	0,11
Perceptron	0,70	0,13

Algoritmalara göre hata matrisi sonuçları, Tablo 4.5’ deki karışıklık matrisi değerleri kullanılarak hesaplanmış ve Tablo 4.6’ da gösterilmiştir.

Tahmin sonuçları karışıklık matrisi doğruluk değeri en düşük ≅0,81 olmuştur. Doğruluk değerine göre en yüksek sonucu ≅0,93 değeri ile SGD ve Perceptron algoritması verirken, en düşük sonucu ≅0,81 ile LassoLars ve ElasticNet algoritması vermiştir. Bu sonuçlar Tablo 4.5’ deki ölçüm metrikleri ile uyumlu sonuçlar üretmiştir.

Geri çağırma metriğine göre ≅0,64 ile Perceptron algoritması en iyi sonucu verirken ≅0,04 değeri ile LassoLars ve Elasticnet algoritması en düşük değeri vermiştir. Hassasiyet metriğine görede aynı sonuçlar elde edilmiştir.

Tablo 4.6. Karışıklık Matrisi Değerleri

Algoritma	**TP**	**FP**	**TN**	**FN**
Knn	44	644	31	31
Naive Base	16	616	59	59
Rand-Forest	43	643	32	32
LassoLars	3	603	72	72
ElasticNet	3	603	72	72
SGD	47	647	28	28
Perceptron	48	648	27	27

Tablo 4.7. Karşılaştırmalı Hata Matrisi Sonuçları.

Algoritma	**Doğruluk**	**Geri Çağırma**	**Hassasiyet**	f_1 **puanı**
Knn	0,92	0,59	0,59	0,59
Naive Base	0,84	0,21	0,21	0,21
RandomForest	0,91	0,57	0,57	0,57
LassoLars	0,81	0,04	0,04	0,04
ElasticNet	0,81	0,04	0,04	0,04
SGD	0,93	0,63	0,63	0,63
Perceptron	0,93	0,64	0,64	0,64

Geri çağırma ve hassasiyet metriğine bağlı olarak hesaplanan f_1 puanına göre, en yüksek sonucu Perceptron algoritması ≅0,64 değeri ile verirken, en düşük sonucu ≅0,04 değeri ile LassoLars ve ElasticNet algoritması vermiştir.

Sunulan metrik ölçümlerine göre, ≅0,93 yüksek doğruluk değerinde elde edilen meslek sınıflandırma sonuçları, günümüzün ekonomik ve mesleki istihdam koşulları ile uyumlu gözükmekle birlikte, geleceğe de ışık tutmaktadır. Model veri kümesinin çeşitliliğini artırırak ve veri temizleme süreçlerinde iyileştirmeler sağlayarak, f_1 puanında iyileştirmeler olacağı çıkarımı yapılmıştır.

Tablo 4.8. Karşılaştırmalı Değerlendirme Sonuçları.

Algoritma	**MAE**	**MAPE(%)**	**Doğruluk**
Knn	2,11	0,26	1
Naive Base	2,29	0,35	1
RandomForest	1,88	0,24	1
LassoLars	2,33	0,35	1
ElasticNet	2,43	0,37	1
SGD	2,49	0,29	1
Perceptron	2,11	0,30	1

Geleceğin mesleklerine yönelik analiz değerlendirme sonuçları Tablo 4.8' de listelenmiştir. Algoritmalar yüksek doğrulukta (≅1) sonuçlar vermiştir. Ortalama mutlak yüzdesel hataya göre göre, RandomForest algoritması ≅0.24 ile en iyi sonucu verirken, ElasticNet algoritması ≅0.37 ile en kötü değeri vermiştir. Ortalama mutlak hataya göre, Random-Forest ≅1.88 değeri ile en iyi sonucu vermiştir. En kötü sonucu ise, ≅2.49 ile SGD algoritması vermiştir. Bu durum, modelin kategorik değişkenler içermesi kaynaklıdır.

5. SONUÇLAR VE ÖNERİLER

Öğrenme algoritmalarının uygulanması aşamasında uygulanan algoritmalar denetimli öğrenme algoritmaları olduğu için veri kümesi ikiye bölünmüş, yarısı öğrenme için kullanılırken, diğer yarısı ise tahmin amaçlı olarak kullanılmış ve geleceğin meslekleri, eldeki veri kümesi ile tahmin edilmeye çalışılmıştır.

Çalışmanın algoritma aşamasında, internetten IPA RAPORU ile ilintili Türkçe belgeler indirilmiş ve bu belgeler üzerinde, metinsel araştırma yöntemleri uygulanmıştır. Öncelikle belgeler kelimelere ayrılmış, anlamlı kelimeler belirlenmiş, frekans analizi çalıştırılmış, meslekler listesi ile filtrelenerek, belgelerde hangi meslek isimlerinin ne kadar geçtiği belirlenmiştir. Meslek isimlerinin birden fazla kelimeden oluştuğu da düşünüldüğünde, n-gram algoritmaları kullanarak birden fazla kelimeye göre frekans analizleri oluşturulmuştur. Frekans analizleri ile tüm belgelerden elde edilen kelime dizileri öğrenme algoritmaları ile sisteme öğretilmiş ve tahminler yapılmıştır.

Bugünkü durumun analizi sonucunda “Profesyonel meslek mensupları”, “Yöneticiler” ve “Teknisyenler, teknikerler ve yardımcı profesyonel meslek mensupları” meslek sınıfları diğer sınıflara göre üstünlük göstermiştir. Knn, Naïve Base, Random Forest, SGD ve Perceptron algoritmalarından elde edilen tahmin sonuçlarında, "Profesyonel meslek mensupları", "Yönetici" ve "Teknisyenler, teknikerler ve yardımcı profesyonel meslek mensupları" meslek sınıflarına sıklıkla rastlanmıştır. Bu sonuçlardan iş modellerindeki değişikliklerden, işleri koordine eden meslekler ve yüksek beceri gerektiren mesleklerin etkilenmeyeceği öngörülebilir. Buna ek olarak, son dönemlerdeki ekonomik dengesizlikler nedeniyle, kamu işlerinin popülerlik açısından ön plana çıktığı görülmüştür.

Geleceğin meslekleri analiz sonuçlarına göre bilgi teknolojileri mesleklerinin gelecekte büyük öneme sahip olacağı öngörülmüştür. Bu, bilgi teknolojisi mesleklerinin, Endüstri 4.0 devrimi ile üretim süreçlerinin önemli bir parçası olacağı beklentisi ile açıklanabilir. Ayrıca genetik mesleğinin diğer mesleklere göre ciddi bir

farkındalık yarattığı, frekans grafiklerinden açıkça görülmektedir. Bu durumu, genetik biliminin, tedavisi mümkün olmayan hastalıkların tedavisinde kullanılması açıklamaktadır.

Gelecekteki meslek analizlerinden elde edilen tahmin sonuçlarına göre, "Yazılım Mühendisliği", "Veri Bilimi ve Bulut Bilişim" gibi bilgi teknolojisi meslekleri diğer iş alanlarına göre üstünlük göstermiştir. Ayrıca tıptaki teknolojik gelişmelere paralel olarak "Moleküler Biyoloji ve Genetik" iş alanları da büyük ilgi gördü. Endüstri 4.0 ile birlikte veri biliminin tüm mesleklerde önemli bir rol oynaması beklenmektedir. Veri biliminin etkisi, tahmin sonuçlarında, "Hemşirelik" mesleğinin "Doktorluk" mesleğinden fazla dikkat çekmesinden açıkça görülmektedir.

Çalışmada, modelin doğruluğunu doğrulamak için şimdiye kadar yapılan analizler sonucunda tespit edilen geleceğin meslekleri ile YÖK tarafından açıklanan öncelikli meslek alanları eşleştirilmeye çalışılmıştır. Beklendiği gibi YÖK'ün öncelikli meslek grupları [152] öngörüleri ile Tablo 4.3. te belirtilmiş olan eşleşmelere göre uyumlu sonuçlar elde edildiği gözlenmiştir.

Geleceğin mesleklerinin analizi, YÖK'ün öncelikli meslek grupları tarafından sınıflandırılmayan mesleklerin varlığına işaret etmiştir. Analiz sonuçlarına göre, YÖK tarafından açıklanan öncelikli meslek alanlarına, "Yönetim ve Organizasyon", "Akıllı Şehir", "Etik Uzmanlık" meslek gruplarının eklenmesinin faydalı olacağı çıkarımı yapılmıştır.

Analize göre SGD ve Perceptron algoritması diğer algoritmalara göre üstünlük göstermiştir. Modelin kategorik değişkenlerden oluşması, LassoLars ve ElasticNet algoritmaları, regresyon algoritması olmaları nedeniyle, değerlendirme metriklerine göre kötü performans göstermiştir. Sınıflandırma algoritmaları tarafından üretilen tahmin sonuçlarına göre yapılan performans karşılaştırmasında, ≅%93 doğrulukta tahminler üretilmiştir. Veri çeşitliliğinin ve hacminin artması durumunda, zeka tabanlı algoritmaların diğer algoritmalara göre farklılaşması, daha belirgin hale gelmesi beklenmektedir. Öte yandan, Türkiye'nin mevcut ekonomik koşullarıyla uyumlu sonuçlar elde edilmesi, popüler mesleklerin metin madenciliği teknikleri ile tahmin edilebileceğini göstermektedir.

Sonuç olarak çalışmanın literatüre katkısı, günümüzde işveren anketleri ile belirlenmeye çalışılan, günümüzün popüler mesleklerinin metin madenciliği teknikleri

ile belirlenebileceği ve geleceğe yönelik projeksiyonların, makine öğrenme algoritmaları tarafından üretilen tahminlerle yapılabileceğidir. YÖK'ün öncelikli meslek alanları ile uyumlu sonuçlar elde edilmesi, bu sava pekiştirici bir etki sunmaktadır.

Çalışmada, analizlerin geliştirilmesi, doğruluk ve tutarlılığa etki eden bazı kısıtlarla karşılaşılmıştır. Gelecekteki meslekler için standart bir otorite tarafından yayınlanan bir meslek listesi olmadığı için, çalışmada kullanılan meslek listesi fütüristlerin eserlerinden ve internetten derlenmiştir. Çalışma kapsamında YÖK' ün öncelikli meslek alanlarına göre sınıflandırma yapılmıştır. Ayrıca gelecekteki mesleklerin analizi için yeterli belge bulunamamıştır.

İleride yapılacak araştırmalar için, endüstri 4.0'ın gerektirdiği beceriler belirlenerek ve bu beceriler ile meslekler arasındaki ilişkiler ortaya konularak, çalışma geliştirilebilir. Çalışmada, Türkiye'nin durumu Türkçe belgeler kullanılarak analiz edilmeye çalışılmıştır. Çalışma, İngilizce belgeler kullanılarak, dünyadaki durumu analiz edecek şekilde genişletilebilir.

KAYNAKLAR

[1] J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, ve A. Marrs, *Disruptive technologies: Advances that will transform life, business, and the global economy*. 2013.

[2] Bilim ve Sanayi Bakanlığı, "Mesleklerin Geleceği Araştırma Raporu", 2018.

[3] M. Öztürk, "Üniversitede Eğitim-Öğretim Gören Öğrencilerde Uluslararası Fiziksel Aktivite Anketinin Geçerliliği Ve Güvenirliği ve Fiziksel Aktivite Düzeylerinin Belirlenmesi", Hacettepe Üniversitesi, 2005.

[4] K. Schwab, "The Fourth Industrial Revolution; World Economic Forum: Geneva, Switzerland", Geneva,Switzerland, 2016.

[5] TTGV, "Sanayide Dijital Dönüşüm Platformu", *Sanayide Dijital Dönüşüm Platformu Eğitim Çalışma Grubu*, 2018.

[6] U.S. Chamber of Commerce, "Made in China 2025: Global Ambitions Built on Local Protections", 2017.

[7] M. Huimin *vd.*, "Strategic plan of 'Made in China 2025' and its implementation", içinde *Analyzing the Impacts of Industry 4.0 in Modern Business Environments*, c. 19, sayı May, 2018, ss. 1–23.

[8] D. J. Hand, "Statistics and Data Mining: Intersecting Disciplines", *ACM SIGKDD Explorations Newsletter*, c. 1, sayı 1, ss. 16–19, 1999, doi: 10.1145/846170.846171.

[9] H. J. Friedman, "Data mining and statistics: What's the connection?", içinde *Proceedings of the 29th Symposium on the Interface Between Computer Science and Statistics*, 1997, ss. 3–9.

[10] J. Han, M. Kamber, ve J. Pei, *Data Mining: Concepts and Techniques*. USA, 2011.

[11] D. T. Larose ve C. D. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, Second. Wiley, 2014.

[12] M. Bramer, *Principles of Data Mining*, sayı February. 2007.

[13] R. Nisbet, J. Elder, ve G. Miner, *Handbook of Statistical Analysis and Data Mining Applications*. London: Elsevier, 2009.

[14] H. Akpınar, "Veri Tabanlarinda Bilgi Keşfi ve Veri Madenciliği", *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, c. 29, sayı 1, ss. 1–22, 2000.

[15] S. Tüzüntürk, "Panel Veri Modellerinin Tahmininde Parametre Heterojenliğinin Önemi: Geleneksel Phillips Eğrisi Üzerine Bir Uygulama", *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, c. 21, sayı 2, ss. 1–14, 2010.

[16] S. Savaş, N. Topaloğlu, ve M. Yılmaz, “Veri Madenciliği ve Türkiye’deki Uygulama Örnekleri”, *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, c. 11, sayı 21, ss. 1–23, 2012.

[17] A. S. Koyuncugil, “Bulanık Veri Madenciliği ve Sermaye Piyasalarına Uygulanması”, Ankara Üniversitesi, 2006.

[18] E. Küçüksille, “Veri madenciliği süreci kullanilarak portföy performansinin değerlendirilmesi ve imkb hisse senetleri piyasasinda bir uygulama”, Süleyman Demirel Üniversitesi, 2009.

[19] A. Şentürk, *Veri Madenciliği: Kavram ve Teknikler*, 1. Basım. Ekin Basım Yayın, 2006.

[20] U. T. Ş. Gürsoy, *Uygulamalı Veri Madenciliği Sektörel Analizler*. Ankara: Pegem Akademi, 2012.

[21] Z. Kızılca, “Türkiye’de Sanayi Üretim Endeksini Etkileyen Faktörler ve Zaman Serisi Analizi, Yüksek Lisans Tezi”, İstanbul Üniversitesi, 2007.

[22] A. Baykasoğlu, “Veri Madenciliği ve Çimento Sektöründe Bir Uygulama”, 2005, [Çevrimiçi]. Available at: http://ab.org.tr/ab05/tammetin/171.pdf.

[23] R. D. King, C. Feng, ve A. Sutherland, “StatLog : Comparison of Classification Algorithms on Large Real-World Problems”, *Applied Artificial Intelligence*, c. 9, sayı 3, ss. 289–333, 1995, doi: 10.1080/08839519508945477.

[24] Ö. G. Ali ve Y.-T. Chen, “Design Quality and Robustness with Neural Networks”, *IEEE Transactions on Neural Networks*, c. 10, sayı 6, ss. 518–1527, 1999, doi: 10.1109/72.809098.

[25] K. R. Skinner *vd.*, “Multivariate statistical methods for modeling and analysis of wafer probe test data”, *IEEE Transactions on Semiconductor Manufacturing*, c. 15, sayı 4, ss. 523–530, 2002, doi: 10.1109/TSM.2002.804901.

[26] M. Li, S. Feng, I. K. Sethi, J. Luciow, ve K. Wagner, “Mining production data with neural network & CART”, içinde *Third IEEE International Conference on Data Mining*, 2003, ss. 731–734, doi: 10.1109/ICDM.2003.1251019.

[27] A. Çoban, “İmalat Sanayinde Veri Madenciliği Destekli Tedarikçi Seçimi Uygulaması”, Sakarya Üniversitesi, 2006.

[28] H. Özçınar, “Kpss Sonuçlarının Veri Madenciliği Yöntemleri ile Tahmin Edilmesi”, Pamukkale Üniversitesi, 2006.

[29] S. Dolgun, M. O, Özdemir, T., G., Deliloğlu, “Öğrenci Seçme Sınavında (ÖSS) Öğrencilerin Tercih Profillerinin Veri Madenciliği Yöntemleriyle Tespiti”, 2009.

[30] V. N. Rajavarman ve S. P. Rajagopalan, “Feature Selection in Data-Mining for Genetics Using Genetic Algorithm”, *Journal of Computer Science*, c. 3, sayı 9, ss. 723–725, 2007, doi: 0.3844/jcssp.2007.723.725.

[31] A. S. Bozkır, B. Gök, ve E. Sezer, “Üniversite Öğrencilerinin Đnterneti Eğitimsel Amaçlar İçin Kullanmalarını Etkileyen Faktörlerin Veri Madenciliği Yöntemleriyle Tespiti”, içinde *Bilimde Modern Yöntemler Sempozyumu BMYS’2008, Eskişehir*, 2008, ss. 833–842.

[32] A. Söylemez, “Bireysel Müşterilerin Kredi Değerlendirme Sonuçlarını En İyiTahmin Eden Scorecard Modelinin Oluşturulması”, Mimar Sinan Güzel Sanatlar Üniversitesi, 2009.

[33] M. Çetin, “Bir üretim işletmesinde veri madenciliği uygulaması”, Sakarya Üniversitesi, 2009.

[34] C. Coşkun, “Veri Madenciliği Algoritmaları Karşılaştırılması”, Dicle Üniversitesi, 2010.

[35] G. Bilekdemir, “Veri madenciliği tekniklerini kullanarak üretim süresi tahmini ve bir uygulama”, Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü, 2010.

[36] D. Tokmak, “Bilgi Keşfi ve İris Veri Seti Üzerinde Veri Madenciliği Araçlarının Karşılaştırılmas”, Başkent Üniversitesi, Fen Bilimleri Enstitüsü, 2011.

[37] C. Çiflikli ve E. Kahya-Özyirmidokuz, “Enhancing product quality of a process”, *Industrial Management and Data Systems*, c. 112, ss. 1181–1200, 2012, doi: 10.1108/02635571211264618.

[38] I. Ertugrul, A. Organ, ve A. Savli, “The Determination Of Patient Profile At Pamukkale University As Relater To The Application Of Data Mining”, *Pamukkale University Journal of Engineering Sciences*, c. 19, sayı 2, ss. 97–103, 2013, doi: 10.5505/pajes.2013.68077.

[39] A. Oğuzlar, “Cart Analizi İle Hanehalki İşgücü Anketi Sonuçlarinin Özetlenmesi”, *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, c. 18, sayı 3–4, 2004, doi: 10.16951/iibd.13143.

[40] E. Yilmaz, “İstatistiksel Analiz Yöntemi Olarak Veri Madenciliğinde CHAID Algoritması ve Türkiye’de İşgücü Piyasasının Durumunun ve Bunun Nedenlerinin Belirlenmesine İlişkin Bir Uygulama”, Yıldız Teknik Üniversitesi Sosyal Bilimler Enstitüsü, 2012.

[41] Y. Li, X. Nie, ve R. Huang, “Web spam classification method based on deep belief networks”, *Expert Systems with Applications*, c. 96, Nis. 2017, doi: 10.1016/j.eswa.2017.12.016.

[42] A. Ågren, “Combating Fake News with Stance Detection using Recurrent Neural Networks”, Chalmers University of Technology University of Gothenburg, 2018.

[43] E. F. Cardoso, R. M. Silva, ve T. A. Almeida, “Towards automatic filtering of fake reviews”, *Neurocomputing*, c. 309, Eki. 2018, doi: 10.1016/j.neucom.2018.04.074.

[44] S. Kudugunta ve E. Ferrara, “Deep neural networks for bot detection”, *Information Sciences*, c. 467, ss. 312–322, 2018, doi: 10.1016/j.ins.2018.08.019.

[45] Á. Figueira ve L. Oliveira, “The current state of fake news: Challenges and opportunities”, *Procedia Computer Science*, c. 121, ss. 817–825, 2017, doi: 10.1016/j.procs.2017.11.106.

[46] Z. Yao ve C. Zhi-Min, “An Optimized NBC Approach in Text Classification”, *Physics Procedia*, c. 24, ss. 1910–1914, 2012, doi:

10.1016/j.phpro.2012.02.281.

[47] B. Trstenjak, S. Mikac, ve D. Donko, "KNN with TF-IDF based framework for text categorization", *Procedia Engineering*, c. 69, ss. 1356–1364, 2014, doi: 10.1016/j.proeng.2014.03.129.

[48] C. Y. Liang *vd.*, "Dictionary-based text categorization of chemical web pages", *Information Processing and Management*, c. 42, sayı 4, ss. 1017–1029, 2006, doi: 10.1016/j.ipm.2005.09.001.

[49] O. Kotevska, S. Padi, ve A. Lbath, "Automatic Categorization of Social Sensor Data", *Procedia Computer Science*, c. 98, ss. 596–603, 2016, doi: 10.1016/j.procs.2016.09.093.

[50] A. Qazi ve R. H. Goudar, "An Ontology-based Term Weighting Technique for Web Document Categorization", *Procedia Computer Science*, c. 133, ss. 75–81, 2018, doi: 10.1016/j.procs.2018.07.010.

[51] F. Mosconi, *The new European industrial policy: Global competitiveness and the manufacturing renaissance*, 1. baskı. London, 2015.

[52] M. Russmann *vd.*, "Industry 4.0: World Economic Forum", 2015.

[53] B. Vogel-Heuser ve D. Hess, "Guest Editorial Industry 4.0-Prerequisites and Visions", *IEEE Transactions on Automation Science and Engineering*, c. 13, sayı 2, ss. 411–413, 2016, doi: 10.1109/TASE.2016.2523639.

[54] L. Vinet ve A. Zhedanov, "A 'missing' family of classical orthogonal polynomials", *Journal of Physics A: Mathematical and Theoretical*, c. 44, sayı 8, 2011, doi: 10.1088/1751-8113/44/8/085201.

[55] R. Kurt, "Industry 4.0 in Terms of Industrial Relations and Its Impacts on Labour Life", *Procedia Computer Science*, c. 158, ss. 590–601, 2019, doi: 10.1016/j.procs.2019.09.093.

[56] A. S. Blinder, "Education for the Third Industrial Revolution,Princeton University, Department of Economics, Center for Economic Policy Studies,Working Papers", 2008.

[57] N. S. Pamuk ve M. Soysal, "Yeni Sanayi Devrimi Endüstri 4.0 Üzerine İnceleme", *Verimlilik Dergisi*, sayı 1. ss. 41–66, 2018, [Çevrimiçi]. Available at: https://dergipark.org.tr/tr/pub/verimlilik/issue/34982/388198.

[58] T. Valentiny, J. Gonos, V. Timková, ve M. Košíková, "Impact of selected factors on the formation of regional disparities in Slovakia", *Journal of Applied Economic Sciences*, c. 12, sayı 6, ss. 1626–1638, 2017.

[59] E. Weber, "Industry 4.0 – job-producer or employment-destroyer?", 2016.

[60] G. C. Kane, D. Palmer, A. N. Phillips, ve D. Kiron, "Is Your Business Ready for a Digital Future?", *MIT Sloan Management Review*, c. 56, sayı 4, ss. 37–44, 2015, [Çevrimiçi]. Available at: https://sloanreview.mit.edu/article/is-your-company-ready-for-a-digital-future/.

[61] C. Kleinert, B. Matthes, ve M. Jacob, "Folgen der Digitalisierung für die Arbeitswelt Substituierbarkeitspotenziale von Berufen in Deutschland", *IAB-Forschungsbericht*, 2015.

[62] M. Özkan, A. Al, ve S. Yavuz, "Uluslararası Politik Ekonomi Açısından Dördüncü Sanayi-Endüstri Devrimi'nin Etkileri ve Türkiye", *Siyasal Bilimler Dergisi*, ss. 1–30, 2018, doi: 10.14782/marusbd.418669.

[63] Winter Corporation, "VLDB Survey Program", 1998. test.winter-corp.com/VLDB/I998 VLD Winners/table7.html.

[64] N. Gürsakal, *Sosyal Bilimlerde Araştırma Yöntemleri*. Vipaş, 2001.

[65] G. Piatetsky-Shapiro, "Knowledge Discovery in Real Databases : A Report on the IJCAI-89 Workshop", *AI Magazine*, c. 11, sayı 4, s. 3, 1990.

[66] G. Memiş, "Veri madenciliği 1", *Data Mining*, 2019. http://www.baskent.edu.tr/~gmemis/courses/datamining/DM_1.pdf.

[67] K. Ergün, "Veri madenciliği", *EMM4214-Veri Madenciliği Dersi*. http://kergun.baun.edu.tr/20192020Bahar/EMM4214.html.

[68] D. Hand, H. Mannila, ve P. Smyth, *Principles of Data Mining Cambridge*. Massachusetts, 2001.

[69] A. OĞUZLAR, "Veri Ön İşleme", *Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, c. 0, sayı 21, 2003.

[70] S. Piramuthu, "Evaluating feature selection methods for learning in data mining applications", *European Journal of Operational Research*, c. 156, ss. 483–494, 2004, doi: 10.1016/S0377-2217(02)00911-6.

[71] D. Pyle, *Data preparation for data mining*. Morgan Kaufmann Publishers, 1999.

[72] A. Famili, W.-M. Shen, R. Weber, ve E. Simoudis, "Data preprocessing and intelligent data analysis", *Intelligent Data Analysis*, c. 1, sayı 1–4, ss. 3–23, Oca. 1997, doi: 10.1016/S1088-467X(98)00007-9.

[73] W. Kim, B. J. Choi, E. K. Hong, S. K. Kim, ve D. Lee, "A Taxonomy of Dirty Data", *Data Mining and Knowledge Discovery*, c. 7, sayı 1, ss. 81–99, 2003, doi: 10.1023/A:1021564703268.

[74] R. J. Roiger, *Data Mining A Tutorial-Based Primer*, Second Edi. Chapman and Hall/CRC, 2017.

[75] A. Çalış, S. Kayapınar, ve T. Çetinyokuş, "Veri Madenciliğinde Karar Ağacı Algoritmaları ile Bilgisayar ve İnternet Güvenliği Üzerine Bir Uygulama", *Endüstri Mühendisliği*, c. 25, sayı 3, ss. 2–19, 2014, [Çevrimiçi]. Available at: http://dergipark.org.tr/endustrimuhendisligi/issue/46771/586362.

[76] S. Özekes, "Veri Madenciliği Modelleri ve Uygulama Alanları", *İstanbul Ticaret Üniversitesi Dergisi*, ss. 65–82, 2003.

[77] S. Albayrak, "Sınıflama ve Kümeleme Yöntemleri". https://slideplayer.biz.tr/amp/3030436/.

[78] T. Ersoz, D. Merdin, ve F. Ersoz, "Veri Madenciliği Yöntemi ile Memnuniyet Algısının Araştırılması - A Research Into Satisfaction Perception By Means of Data Mining Method", 2015, [Çevrimiçi]. Available at: VTT Information Technology Research Report.

[79] F. Ersöz, *Veri Madenciliği Teknikleri ve Uygulamaları*. Ankara: Seçkin

Yayınevi, 2019.

[80] C. F. Chien, W. C. Wang, ve J. C. Cheng, “Data mining for yield enhancement in semiconductor manufacturing and an empirical study”, *Expert Systems with Applications*, c. 33, sayı 1, ss. 192–198, Tem. 2007, doi: 10.1016/j.eswa.2006.04.014.

[81] C. E. Shannon, “A Mathematical Theory of Communication”, *Bell System Technical Journal*, c. 27, sayı 3, ss. 379–423, 1948, doi: 10.1002/j.1538-7305.1948.tb01338.x.

[82] C. Bounsaythip ve E. Rinta-Runsala, “Overview of Data Mining For Customer Behavior Modeling”, VTT Information Technology Research Report, 2002.

[83] J. R. Quinlan, “Induction of decision trees”, *Machine Learning*, c. 1, ss. 81–106, 1986, doi: 10.1007/BF00116251.

[84] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1992.

[85] L. Breiman, “Random forests”, *Machine Learning*, c. 45, sayı 1, ss. 5–32, 2001, doi: 10.1023/A:1010933404324.

[86] W. M. Waggener, *Pulse Code Modulation Techniques*, 1. baskı. Springer, 1995.

[87] L. Metcalf ve W. Casey, “Metrics, similarity, and sets”, *Cybersecurity and Applied Mathematics*, ss. 3–22, 2016, doi: 10.1016/B978-0-12-804452-0.00002-6.

[88] D. Alkın, “Kümeleme analizi yöntemlerinin küme geçerlilik indekslerine göre karşılaştırılması”, Necmettin Erbakan Üniversitesi, 2019.

[89] L. Kaufman ve P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. New York: John Wiley & Sons, Inc., 1990.

[90] G. Mecca, S. Raunich, ve A. Pappalardo, “A new algorithm for clustering search results”, *Data and Knowledge Engineering*, c. 62, sayı 3, ss. 504–522, 2007, doi: 10.1016/j.datak.2006.10.006.

[91] I. H. Witten, “Text mining”, içinde *Practical handbook of internet computing*, Florida: Chapman & Hall/CRC Press, 2005, ss. 14-1-14–22.

[92] D. Delen ve M. D. Crossland, “Seeding the survey and analysis of research literature with text mining”, *Expert Systems with Applications*, c. 34, sayı 3, ss. 1707–1720, 2008, doi: 10.1016/j.eswa.2007.01.035.

[93] E. Alpaydın, “Zeki Veri Madenciliği: Ham Veriden Altın Bilgiye Ulaşma Yöntemleri”, *Bilişim 2000 Eğitim Semineri*. ss. 1–10, 2000.

[94] G. Miner, D. Delen, J. Elder, A. Fast, T. Hill, ve R. A. Nisbet, *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. 2012.

[95] S. E. Şeker, “Metin Madenciliği (Text Mining)”, 2014. http://bilgisayarkavramlari.sadievrenseker.com/2014/06/15/metin-madenciligi-text-mining/.

[96] E. Zohar, “Introduction to Text Mining”, *Supercomputing*, 2002.

[97] B. C. Vickery, *On Retrieval System Theory*. London, 1965.

[98] C. N. Mooers, "Application of random codes to the gathering of statistical information", Massachusetts Institute of Technology, 1948.

[99] E. Sezer, "Web Sayfaları İçin Anlamsal Erişim Sistemi", Hacettepe Üniversitesi, 2006.

[100] F. Can, S. Kocberber, E. Balcik, C. Kaynak, H. C. Ocalan, ve O. M. Vursavas, "Information Retrieval on Turkish Texts", *Journal of the American Society for Information Science and Technology*, c. 59, sayı 3, ss. 407–421, 2008, doi: 10.1002/asi.20750.

[101] İ. F. Pilavcılar, "Metin Madenciliği İle Metin Sınıflandırma", Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 2007.

[102] E. Sezer, "Öğrenci Seçme Sınavında (ÖSS) Öğrencilerin Tercih Profillerinin Veri Madenciliği Yöntemleriyle Tespiti", sayı November 2014, ss. 13–15, 2009.

[103] S. Soderland, "Learning information extraction rules for semi-structured and free text", *Machine Learning*, c. 34, sayı 1, ss. 233–272, 1999, doi: 10.1023/A:1007562322031.

[104] R. Daş, "Web Kullanıcı Erişim Kütüklerinden Bilgi Çıkarımı", Fırat Üniversitesi Fen Bilimleri Enstitüsü, Elazığ, 2008.

[105] K. Kaiser ve S. Miksch, "Information Extraction A Survey", Vienna, Avusturya,Vienna University of Technology Institue of Software Technology & Interactive Systems, 2005.

[106] N. Kushmerick, "Wrapper Induction for Information Extraction", University of Washington, 1997.

[107] K. Oflazer, "Türkçe İçin Bir Sonlu Durumlu 'Hafif' Doğal Dil Çözümleyicisi ve Bilgi Çıkarımı Uygulamasının Gerçekleştirilmesi", TÜBİTAK PROJESİ, PROJE NO:199E027, 2002.

[108] A. Güven, "Türkçe Belgelerin Anlam Tabanli Yöntemlerle Madenciliği", Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi., 2007.

[109] B. Oğuz, "Metin Madenciliği Teknikleri Kullanılarak Kulak Burun Boğaz Hasta Bilgi Formlarının Analizi", Akdeniz Üniversitesi Sağlık Bilimleri Enstitüsü, 2009.

[110] G. Cebiroğlu, A. C. Tantuğ, E. Adalı, ve Y. Erenler, "Sentetik Türkçe Sözcük Kökleri Üretimi", içinde *International XII. Turkish Symposium on Artificial Intelligence and Neural Networks TAINN*, sayı 2003.

[111] G. C. Eryiğit, "Türkçe'nin Bağlılık Ayrıştırması", İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi, 2006.

[112] VeriBilimcisi, "Denetimli Öğrenme (Supervised Learning)", 2017, [Çevrimiçi]. Available at: https://veribilimcisi.com/2017/07/12 /denetimli-ogrenme/.

[113] VeriBilimcisi, "Denetimsiz Öğrenme (Unsupervised Learning)", 2017. https://veribilimcisi.com/2017/07/12/denetimsiz-ogrenme/.

[114] TutorialPoints, "KNN Algorithm - Finding Nearest Neighbors, Machine

Learning with Python", 2019. https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_knn_algorithm_finding_nearest_neighbors.htm.

[115] VeriBilimcisi, "K-En Yakın Komşu (K-Nearest Neighbors(KNN)), Makine Öğrenmesi", 2017. https://veribilimcisi.com/2017/07/20/k-en-yakin-komsu-k-nearest-neigh borsknn/.

[116] M. Cover T ve E. Hart P, "Nearest Neighbor Pattern Classification", *IEEE Transactions on Information Theory*, c. 13, sayı 1, ss. 21–27, 1967.

[117] DataLabTR, "Naïve Bayes Algoritması ve R Uygulaması", 2019. https://medium.com/@datalabtr/naïve-bayes-algoritması-ve-r-uygulaması-4d321869d371.

[118] A. Chakure, "Random Forest Classification", 2019. https://towardsdatascience.com/random-forest-classification-and-its-implementation-d5d840dbead0.

[119] Y. Liu, Y. Wang, ve J. Zhang, "New machine learning algorithm: Random forest", içinde *International Conference on Information Computing and Applications. ICICA 2012. Lecture Notes in Computer Science*, 2012, c. 7473, ss. 246–252, doi: 10.1007/978-3-642-34062-8_32.

[120] R. Tibshirani, "A simple explanation of the Lasso and Least Angle Regression", *Tibshirani web page*, 2015. http://statweb.stanford.edu/~tibs/lasso/simple.html.

[121] H. Zou ve T. Hastie, "Regression Shrinkage and Selection via the Elastic Net, with Applications to Microarrays", 2003.

[122] Ö. F. Rençber ve H. Bağcı, "Determination of Factors Affecting Capital Adequacy Using the Elastic Net Regression Method", *OPUS Uluslararası Toplum Araştırmaları Dergisi*, c. 11, sayı 18, ss. 1828–1844, 2019, doi: 10.26466/opus.561915.

[123] Z. Zhang, Z. Lai, Y. Xu, L. Shao, J. Wu, ve G. Xie, "Discriminative Elastic-Net Regularized Linear Regression", *IEEE Transactions on Image Processing*, c. 26, sayı 3, ss. 1466–1481, 2017, doi: 10.1109/TIP.2017.2651396.

[124] A. V Srinivasan, "Stochastic Gradient Descent", 2019. https://towardsdatascience.com/stochastic-gradient-descent-clearly-explained-53d239905d31.

[125] L. Bottou, "Stochastic gradient descent tricks", *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science*, c. 7700, ss. 421–436, 2012, doi: 10.1007/978-3-642-35289-8_25.

[126] D. E. Rumelhart, G. E. Hinton, ve R. J. Williams, "Learning internal representations by error propagation", içinde *Parallel distributed processing: explorations in the microstructure of cognition*, c. 1, sayı V, 1986, ss. 318–362.

[127] N. Chigali, "Simple Perceptron Training Algorithm:Explained", *The University of Utah*, 2018. https://medium.com/@nikhilc3013/simple-perceptron-training-algorithm-explained-7bbfdff2c57d.

[128] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain", *Psychological Review*, c. 65, sayı 6, ss. 386–408,

1958, doi: 10.1037/h0042519.

[129] A. L. Chandra, “Perceptron Learning Algorithm: A Graphical Explanation Of Why It Works”. https://towardsdatascience.com/perceptron-learning-algorithm-d5db0deab975.

[130] S. He, “The Perceptron Algorithm”, *Machine Learning Spring 2018*, 2018. http://www.cs.utah.edu/~zhe/pdf/lec-10-perceptron-upload.pdf.

[131] VeriBilimcisi, “Özellik Ölçekleme ve Normalleştirme (Feature Scaling and Normalization)”. https://veribilimcisi.com/2017/07/18/ozellik-olcekleme-ve-normallestirme-nedir-feature-scaling-and-normalization/.

[132] E. Uzun, “Makine Öğrenmesi”. https://erdincuzun.com/makine_ogrenmesi/makine-ogrenmesi-metotlari/.

[133] VeriBilimcisi, “Tahminlerin Kalitesini Ölçmek”. https://veribilimcisi.com/2017/07/14/makine-ogrenme-algoritmalarini-degerlendirme-metrikleri/.

[134] Veribilimcisi, “Doğruluk, Hassasiyet, Hata (Accuracy, Precision, Error)”. https://veribilimcisi.com/2017/07/14/dogrulukaccuracy-kesinlikprecision-hataerror-nedir/.

[135] VeriBilimcisi, “MSE, RMSE, MAE, MAPE ve Diğer Metrikler”. https://veribilimcisi.com/2017/07/14/mse-rmse-mae-mape-metrikleri-nedir/.

[136] M. Ohsaki, P. Wang, K. Matsuda, S. Katagiri, H. Watanabe, ve A. Ralescu, “Confusion-matrix-based kernel logistic regression for imbalanced data classification”, *IEEE Transactions on Knowledge and Data Engineering*, 2017, doi: 10.1109/TKDE.2017.2682249.

[137] International Labour Organization, “International Standard Classification of Occupations”, [Çevrimiçi]. Available at: https://www.ilo.org/public/english/bureau/stat/isco/docs/publication08.pdf.

[138] H. Heidenreich, “Stemming? Lemmatization? What?”, 2018. .

[139] O. Tunçelli, “Turkish Stemmer for Python”, 2019. https://github.com/otuncelli/turkish-stemmer-python.

[140] H. K. Simsek, “Makine Öğrenmesi Dersleri 6: Doğal Dil İşleme (NLP)”, 2018. https://medium.com/data-science-tr/makine-öğrenmesi-dersleri-6-doğal-dil-işleme-nlp-453c3c6b062a.

[141] A. Köksal, “Turkish-Lemmatizer”, 2018. https://github.com/akoksal/Turkish-Lemmatizer.

[142] K. Toutanova, D. Klein, C. D. Manning, ve Y. Singer, “Feature-rich part-of-speech tagging with a cyclic dependency network”, sayı June, ss. 173–180, 2003, doi: 10.3115/1073445.1073478.

[143] T. Brants, “A Statistical Part-ofSpeech Tagger”, içinde *In Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP 2000)*, 2000, ss. 224–231.

[144] A. Ratnaparkhi, “A maximum entropy model for part-of-speech tagging”, içinde *Proceedings of the Conference on Empirical Methods in Natural*

Language Processing, University of Pennsylvania, 1996, ss. 133–142.

[145] E. Brill, "Some advances in rule-based part of speech tagging", *Proceedings of the Twelfth Annual Conference on Artificial Intelligence*, ss. 722–727, 1994.

[146] C. Samuelsson, A. Voutilainen, ve L. Technologies, "Comparing a Linguistic and a Stochastic Tagger", içinde *ACL '98/EACL '98: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, 1997, ss. 246–253, doi: https://doi.org/10.3115/976909.979649.

[147] F. Hu ve R. H. Trivedi, "Mapping hotel brand positioning and competitive landscapes by text-mining user-generated content", *International Journal of Hospitality Management*, c. 84, 2020, doi: 10.1016/j.ijhm.2019.102317.

[148] M. Vanhala, C. Lu, J. Peltonen, S. Sundqvist, J. Nummenmaa, ve K. Järvelin, "The usage of large data sets in online consumer behaviour: A bibliometric and computational text-mining–driven analysis of previous research", *Journal of Business Research*, c. 106, sayı September 2019, ss. 46–59, 2020, doi: 10.1016/j.jbusres.2019.09.009.

[149] H. Hassani, C. Beneki, S. Unger, M. T. Mazinani, ve M. R. Yeganegi, "Text mining in big data analytics", *Big Data and Cognitive Computing*, c. 4, sayı 1, ss. 1–34, 2020, doi: 10.3390/bdcc4010001.

[150] X. Xie, Y. Fu, H. Jin, Y. Zhao, ve W. Cao, "A novel text mining approach for scholar information extraction from web content in Chinese", *Future Generation Computer Systems*, c. 111, ss. 859–872, 2020, doi: 10.1016/j.future.2019.08.033.

[151] D. Özdemir, Ş., Kılınç, "Geleceğin Meslekleri Listesi", 2019.

[152] Yüksek Öğretim Kurumu, "Geleceğin Meslekleri Çalışmaları", 2019, [Çevrimiçi]. Available at: https://www.yok.gov.tr/Documents/Yayinlar/Yayinlarimiz/2019/gelecegin_meslekleri_calismalari.pdf.

ÖZGEÇMİŞ