T.C.
PAMUKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

TEMEL VERİ MADENCİLİĞİ ALGORİTMALARININ BAŞARIMLARININ ENDOKRİN VERİ SETİ ÜZERİNDE KARŞILAŞTIRILMASI

YÜKSEK LİSANS TEZİ

SİNEM CEYLAN

DENİZLİ, NİSAN - 2023

T.C.
PAMUKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

TEMEL VERİ MADENCİLİĞİ ALGORİTMALARININ BAŞARIMLARININ ENDOKRİN VERİ SETİ ÜZERİNDE KARŞILAŞTIRILMASI

YÜKSEK LİSANS TEZİ

SİNEM CEYLAN

DENİZLİ, NİSAN - 2023

Bu tez çalışmasında Pamukkale Üniversitesi Girişimsel Olmayan Klinik Araştırmalar Etik Kurulu tarafından 21.12.2021 tarihli E-60116787-020-44741 sayılı izin ile Pamukkale Üniversitesi İç Hastalıkları Polikliniğine ait 2021 yılı hasta verileri kullanılmıştır.

Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bilimsel etiğe ve akademik kurallara özenle riayet edildiğini; bu çalışmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etiğe uygun olarak kaynak gösterildiğini ve alıntı yapılan çalışmalara atfedildiğine beyan ederim.

SİNEM CEYLAN

ÖZET

TEMEL VERİ MADENCİLİĞİ ALGORİTMALARININ BAŞARIMLARININ ENDOKRİN VERİ SETİ ÜZERİNDE KARŞILAŞTIRILMASI

YÜKSEK LİSANS TEZİ

SİNEM CEYLAN

PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

(TEZ DANIŞMANI:PROF. DR. SERDAR İPLİKÇİ)

DENİZLİ, NİSAN - 2023

Gelişen teknoloji olanaklarının artması ile birlikte, birçok alanda veri depolanmaktadır. Elde edilen verilerden yol çıkılarak, anlamlı, yorumlanabilir ve insanlığın faydasına yönelik kullanılabilmesi için veri analiz yöntemlerine ve çözümlerine ihtiyaç duyulmaktadır. Teknolojinin ilerlemesiyle birlikte tıp alanında da büyük ve karmaşık veri tabanları oluşmaktadır. Veri madenciliği yöntemleri ile bu karmaşık veri tabanları içerisinden anlamlı verileri tespit etmek; bir altyapı oluşturmak, problemi tespit etmek, problemi çözmek veya bir hastalık teşhisinde daha hızlı ve çeşitli bakış açısı kazandırmaktadır.

Bu tez çalışmasında Pamukkale Üniversitesi Hastanesi İç Hastalıkları Polikliniğine başvurmuş hastaların kan testleri bilgilerini içeren veri seti ele alınarak hasta profili belirlenmeye çalışılmıştır. Üzerine çalışılan hastalıklardan birine sahip olduğu bilinen bir kişinin, çalışılan diğer üç hastalık ile ilişkisi incelenmiş ve bu dört hastalık arasındaki ilişkinin gelecekte oluşabilecek rahatsızlıkların ön teşhisinde kullanılabileceği düşünülmüştür. Ayrıca çalışmada kullanılan Apriori, ECLAT, FP-Tree ve H-Mine algoritmaların veri seti üzerindeki performansları incelenmiş ve birbirleri arasında performans farkları değerlendirilmiştir.

ANAHTAR KELİMELER: Endokrinoloji, Veri Madenciliği, Büyük Veri

ABSTRACT

COMPARISON OF THE PERFORMANCE OF DATA MINING ALGORITHMS ON THE ENDOCRINE DATA SET

MSC THESIS

SİNEM CEYLAN

PAMUKKALE UNIVERSITY INSTITUTE OF SCIENCE

ELECTRICAL AND ELECTRONICS ENGINEERING

(SUPERVISOR:PROF. DR. SERDAR İPLİKÇİ)

DENİZLİ, APRIL 2023

With the increase in developing technology facilities, data can be stored in many areas. Data analysis methods and solutions are needed to use it for meaningful, interpreted, and used for the benefit of humanity by way of the data obtained. With the advancement of technology, large and complex databases are formed in the developing medical. With data mining methods, detecting meaningful data from these complex databases, creating an infrastructure, detecting the problem, solving the problem, or providing a faster and various perspective in the diagnosis of a disease.

In this study, the patient's profile was tried to be determined by considering the data set containing the blood test information of the patients who applied to the Internal Diseases Policlinic of Pamukkale University Hospital. A person who is known to have one of the diseases worked on the relationship between the other three diseases studied and the relationship between these four diseases is thought to be used in the preliminary diagnosis of future disorders. In addition, the performance of Apriori, ECLAT, FP-Tree, and H-Mine algorithms used in the study have been examined on the dataset and their performance differences have been evaluated against each other.

KEYWORDS: Endocrinology, Data Mining, Big Data

İÇİNDEKİLER

Sayfa

ŞEKİL LİSTESİ

Sayfa

TABLO LİSTESİ

Sayfa

SEMBOL ve KISALTMALAR LİSTESİ

ARM	**:**	Association Rule Mining (İlişkisel Veri Madenciliği)
Ck	**:**	Aday Öğe Seti
Conf	**:**	Güven Değeri (Confidence)
ECLAT	**:**	Equivalence CLAss Transformation
FIM	**:**	Frequent Itemset Mining (Sık Öğe Seti Madenciliği)
Fk	**:**	Sık Öğe Seti
FP-Tree	**:**	Frequent-Pattern Tree
k-NN	**:**	k-Nearest Neighbor
LCM	**:**	Least Common Multiple
MB	**:**	Mega-Bayt
s	**:**	Saniye
Supp	**:**	Destek Değeri (Support)
TID	**:**	İşlem Veri Tabanı
VM	**:**	Veri Madenciliği

ÖNSÖZ

Lisans ve yüksek lisans eğitimlerim boyunca beni yönlendiren, çalışmalarımın her aşamasında engin bilgilerini ve yardımlarını esirgemeyen değerli danışman hocam Prof. Dr. Serdar İPLİKÇİ'ye, yönlendirmeleri ve yeni fikirleriyle beni destekleyen değerli hocam Dr. Öğr. Üyesi Bedri BAHTİYAR'a, bu alanda çalışmamı destekleyen ve her zaman yardımıma koşan değerli hocam Doç. Dr. Şenay TOPSAKAL'a sonsuz teşekkür ederim.

Ayrıca bu çalışma sürecinde desteği ve yardımlarıyla her zaman beni motive eden canım arkadaşım Leyla TÜLÜ'ye, her konuda ve koşulda benimle yürüyen sevgili eşim Gökhan KONAK'a, tüm eğitim hayatım boyunca yanımda olan, desteklerini hiç esirgemeyen ve asla haklarını ödeyemeyeceğim canım kardeşime, anneme ve babama sonsuz teşekkür ederim.

1. GİRİŞ

Gelişen teknoloji, akıllı sistemler, bilgisayar sistemlerinin hayatın her alanında yaygınlaşması sonucunda, birçok ham veri depolanmaktadır. Tek başına bir anlam ifade etmeyen bu verilerin mevcut olan verilerden yola çıkılarak gelecek nesiller için insanoğlunun ihtiyaçlarını karşılayacak şekilde anlamlı hale getirilmesi için yeni analiz yöntemlerine ihtiyaç vardır. Örneğin, sosyal medya platformlarında oluşan verilerin kaydedilmesi oldukça büyük bir veri tabanı yapısı oluşturmaktadır. Bu büyük ve anlamsız verilerin içerisinde bulunan anlamlı veriler analiz edilerek, insanlığın ihtiyacına ve faydasına yönelik kullanılmalıdır. Veri madenciliği, bu alanda analiz yapanları destekleyen, karar vericilere kolaylık sağlayan ve fikir veren, sorunlar oluşmadan önlem almaya fırsat tanıyan yapısı ile birçok çalışma alanında yerini almış, güvenirliliği kanıtlanmış disiplinler arası bir çalışma modeli olarak karşımıza çıkmaktadır. Bu sebeple, ham veri tabanlarını, faydalı bilgi haline dönüştürerek anlam kazandıran veri madenciliği teknikleri son yıllarda büyük önem kazanmıştır.

Veri madenciliği, farklı kaynaklardan ve farklı metotlarla elde edilen veri tabanlarının, belirli bir strateji ve teknoloji yardımıyla analizleri yapılarak elde edilen bulgular sayesinde ekonomik, siyasi ve teknoloji gibi alanlarda kullanılmasına ve yaygınlaşmasına imkân tanımaktadır. Bilgiyi en uygun şekilde kullanan araştırmacılar elde etmek istedikleri bulguya kısa yoldan ve hızlı bir şekilde veri madenciliği yöntemleriyle ulaşmaktadır.

Sağlık sektöründe kullanılan teknoloji ve bilişim sistemlerinin gelişimi, bu alandaki uygulamaların elektronik ortama aktarılabilmesi sayesinde bir kişinin sağlığı ile ilgili tüm bilgilerin kaydedilmesine ve bu sayede birçok verinin oluşmasına ve depolanmasına olanak sağlamaktadır (Yücel ve diğ. 2018). Sağlık sektöründe kullanılan Sağlık Bilişim Sistemleri, hastalığın teşhisi ve tedavisinde hekimlere karar verme kolaylığı sağlamaktadır. Hastalık teşhisinin doğru yapılmasıyla hastalığa konulan erken teşhis ve doğru tedavi, iyileşme süresini ve bu süreçte oluşan maliyetleri azaltmaktadır. Veri madenciliği yöntemleri birçok alanda olduğu gibi sağlık alanında da son yıllarda yaygın olarak kullanılmaya başlanmıştır.

1.1 Amaç

Günümüzde her alanda olduğu gibi sağlık sektöründe de ciddi oranda veri bulunmaktadır. Hastanelere gelen her bir hastanın testleri bir veri bankasında toplanmakta ve bu verilerin anlamlı hale getirilerek hastaya ne konusunda teşhis konulacağı hususunda yardımcı bilgilere ihtiyaç duyulmaktadır. Bunların en başında günümüzde yaygınlaşmaya başlayan yapay zekâ teknikleri ve veri madenciliği gelmektedir. Çalışmanın birincil amacı temel veri madenciliği algoritmalarını tanımlayarak, bu algoritmaların performanslarının karşılaştırmaktır. Çalışmanın ikincil amacı ise, endokrin polikliniğine gelen hastalara ait testlerin sonuçları kullanılarak bu temel veri madenciliği algoritmalarında elde edilen ilişkisel kuralları analiz etmektir. Bu ilişkiler, endokrin hastaları için; a) Diyabet, b) Tiroit, c) Hipofiz ve d) Obezite hastalıkları arasındaki gizli ilişkilerdir. Çalışmanın özel amaçları ise aşağıda verilmiştir.

i) Veri madenciliği konusunu kısaca tanımlamak ve literatür taraması yapmak,
ii) Veri madenciliği algoritmalarını endokrin veri seti üzerinde uygulayarak, 4 tipte bulunan endokrin hastaları için her bir tip arasındaki gizli ilişkileri ortaya koymak
iii) Endokrin hastaları için incelenen test sonuçlarına bağlı oluşan örüntüleri kullanarak teşhis konulmasına yardımcı olmaktır.

1.2 Materyal ve Metot

Çalışmada, literatürden yola çıkılarak, veri madenciliği ve veri madenciliği algoritmaları tanımlandıktan sonra algoritmaların performansı kıyaslanmış ve performansı en iyi algoritmanın seçilmesi ile, endokrin hastalarından elde edilen verilere uygulanarak sonuçlar elde edilmiştir. Kullanılan yöntem, algoritmaların PYTHON ortamında gerçekleştirilerek çıktıların EXCEL ortamına istatistiksel olarak ortaya konulmasıdır. Konu ile ilgili aşağıda yöntem hakkında kısaca bilgi sunulmuştur.

Kan testi sonuçları, operasyon raporları, kişisel şikayetler ve hekimin görüşleri gibi verilerden oluşan veri tabanları, veri madenciliği algoritmaları tarafından analiz

edilip, gelecekteki teşhis ve tedavi hizmetlerinin kalite artırımını sağlayan bir yöntem olarak değerlendirilebilir.

Endokrin sistemi vücudun kontrol ve düzenleme sistemidir, endokrin bezler olarak tanımlanan organlar ve doku tarafından oluşturulmuştur. Bu bezlere, hipofiz bezi, tiroit bezi, böbrek üstü bezleri, pankreas gibi bezler örnek olarak verilebilir (Koz ve diğ 2016). Endokrin bezler tarafından salgılanan hormonlar, hücreler ve dokulara giderek vücudun temel fonksiyonlarını kontrol etmektedir.

Bu çalışmada Pamukkale Üniversitesi Hastanesi İç Hastalıkları Polikliniğine başvurmuş hastaların kan testleri sonuçlarını içeren veri seti ele alınarak hasta profili belirlenmeye çalışılmıştır. Veri setinde genel hastalık tanısı olarak dört (Tip 1 ve Tip 2 diyabet, tiroit, hipofiz ve obezite) adet, ICD-10 kodlarına bağlı alt tanılarla birlikte toplam 35 hastalık tanısı bulunmaktadır.

Temel veri madenciliği algoritmalarından dört tanesi (Apriori, ECLAT (Equivalence CLAss Transformation), FP-Tree (Frequent-Pattern Tree) ve H-mine) seçilmiş ve bu algoritmaların kullanılan veri seti üzerindeki performans analizleri yapılmış ve bu performanslar değerlendirilmiştir. Ayrıca üzerine çalışılan diyabet, tiroit, hipofiz veya obezite hastalıklarından birine sahip olduğu bilinen bir hastanın, diğer üç hastalık ile ilişkisi ve bir hastalığın içerisinde gizli kalmış ilişkiler incelenmiştir.

1.3 Kapsam

Bu tez beş bölümden oluşmaktadır. Giriş bölümünde veri madenciliği genel olarak tanımlanmış ve sağlık sektöründeki önemi vurgulanmıştır. Bu tez, endokrin hastalarından elde edilen verilerin veri madenciliği algoritmaları ile analiz edilmesini ve en iyi performans gösteren algoritmanın seçilmesini amaçlamaktadır. Bu çalışmanın yöntemi, PYTHON ortamında gerçekleştirilen algoritma analizleri sonucu elde edilen verilerin istatistiksel olarak EXCEL ortamına aktarılmasıdır.

İkinci bölümde, sağlık sektöründe veri madenciliği ile ilgili yapılan çalışmaların bir literatür taraması yapılmıştır. Sağlık sektörü, büyük miktarda veri

üretmektedir ve bu verilerin doğru bir şekilde analiz edilmesi, hastalıkların daha iyi anlaşılmasına, tedavi yöntemlerinin geliştirilmesine ve sağlık hizmetlerinin kalitesinin artırılmasına yardımcı olabilir. Bu bağlamda, sağlık sektöründe veri madenciliği, hastalıkların teşhisinde, tedavi yöntemlerinin belirlenmesinde, epidemiyolojik çalışmalarda ve hastane yönetiminde kullanılmaktadır. Bu bölümde, sağlık sektöründe veri madenciliği ile ilgili yapılan çalışmaların bir özeti sunulmuştur.

Üçüncü bölümde, veri madenciliğinin tanımını, kullanım alanlarını ve farklı modellerini açıklanmaktadır. Veri madenciliği, büyük veri setlerinden anlamlı bilgilerin keşfedilmesine olanak tanıyan bir süreçtir. Sağlık, işletme, pazarlama, finans, eğitim ve diğer birçok alanda kullanılır. Farklı modeller, veri tiplerine ve amaçlarına göre tasarlanmıştır ve veri setlerinin analizi için kullanılabilirler. Bunlar arasında birliktelik kuralları, sınıflandırma, kümeleme ve regresyon gibi modeller bulunmaktadır.

Dördüncü bölümde, tez çalışmasında kullanılan verinin nasıl hazırlandığı ve algoritmada analiz edildiği belirtilmiştir. Veri hazırlama aşaması, verilerin temizlenmesi, önişleme işlemlerinin yapılması, eksik verilerin tamamlanması ve özelliklerin belirlenmesi gibi adımları içermektedir. Hazırlanan veri, veri madenciliği algoritmalarına uygulanarak analiz edilmiş ve kurallar çıkarılmıştır. Elde edilen sonuçlar, gerçek ilişki olasılığına göre değerlendirilmiş ve bu ilişkilerin hangilerinin anlamlı olduğu belirlenmiştir. Kullanılan algoritmaların performansları analiz edilmiş ve kıyaslanma yapılmıştır.

Beşinci bölüm olan sonuçlar ve öneriler kısmında, önce tez çalışmasının genel bir özeti verilmiş ve elde edilen sonuçların literatüre katkısı açıklanmıştır. Ayrıca, bu çalışmanın sınırları ve gelecekteki çalışmalar için öneriler de sunulmuştur. Sonuçlar kısmında, kullanılan algoritmaların performansı değerlendirilmiş ve sonuçların ne kadar başarılı olduğu belirtilmiştir. Ayrıca, veri madenciliği teknikleri kullanılarak elde edilen sonuçların, incelenen konu hakkında bilgi sahibi olmak isteyen kişiler ve kurumlar için faydalı olabileceği vurgulanmıştır. Öneriler kısmında ise, bu çalışmadan elde edilen sonuçların kullanım alanlarının artırılması için çeşitli öneriler sunulmuştur. Örneğin, elde edilen sonuçların ticari uygulamalarda kullanılması veya veri madenciliği yöntemlerinin diğer konulara uygulanması gibi öneriler yer almaktadır.

Ayrıca, bu çalışmanın sınırları da belirtilerek, gelecekteki çalışmalarda bu sınırların dışında yapılabileceklerden öneriler sunulmuştur.

2. LİTERATÜR TARAMASI

2.1 Veri Madenciliği Çalışmaları

İletişim ve teknolojinin gelişmesi ile tüm dünyada olduğu gibi ülkemizde de ciddi oranda ham veri bulunmaktadır. Bu veriler fen bilimlerinden sosyal bilimlere kadar birçok alanı kapsamaktadır. Tüm sorun, ham şekilde bulunan verilerin depolanması, depolanan verilerden yola çıkılarak yapay zekâ teknikleri kullanılarak sorumlulara (idareciler, doktoralar, vb.) en kısa sürede yardımcı olabilecek bilgileri sunmaktır. Bu kapsamda son yıllarda veri madenciliği üzerine çalışmalar yoğunlaşarak birçok alanda gelişmeler sağlanmıştır. Bunların başında, otomotiv, gıda, elektronik, mekatronik, ulaşım, sağlık, vb. alanlardır. Çalışmada, sağlık sektöründe yapılan ver madenciliği çalışmaları irdelenmiştir.

Koyuncugil ve Özgülbaş (2009), büyük ölçekteki veriler içinden, gizlenmiş, kıymetli, başka alanlarda fayda sağlayabilecek bilgileri gün yüzüne çıkarmak ve stratejik konularda karara bağlama ve destek vermek maksadıyla faydalanılan Veri Madenciliğinin büyük ölçekte veriyle alakalı sıkıntı olan dallarda çözüm bulabildiği belirtmişleridir. Sağlık sektöründe de sağlık ile ilgili verilerin faydalanılmasında yepyeni bir bakış açısı oluştuğunu ve popülaritesinin giderek arttığını ifade etmişlerdir. Veri madenciliğinin faydaları arasında sağlık çalışanlarının bilgiye en doğru ve güncel şekilde ulaşmasını, en uygun ve objektif cevaplar bularak uygulamasını sağlayan kararları vermesini göstermişlerdir. Ayrıca Veri Madenciliğini gelecek adına sayısal olarak karar verme ve iş zekâsı yönetimi aracı olarak betimlemişlerdir. Kaynaksal açıdan daha etkin ve verimli kullanımın, sağlık olanaklarının daha noktasal kullanımının ve bilimsel dayanaklarının olmasının veri madenciliği sayesinde olacağını savunmuşlardır. Aynı zamanda veri madenciliğinin hasta akış planlamasında, tıbbi tedavi süreçlerinin optimize edilmesinde, ilaç birim maliyetlerinin hesabında, hastaya göre ilaç ve tedavi profili oluşturulmasında yararlar sağlayacağını ifade etmişlerdir.

Özdemir ve diğ. (2010), bilgi keşfinin çok büyük ölçeklere sahip olan verilerin içinde gizlenmiş, doğru, tutarlı, değerli ve fayda sağlayacak bilgileri gün ışığına çıkaracak ve stratejik olarak bizlere yol gösterecek olan bir olgu olduğunu

savunmuşlardır. Geniş bir yelpazesi olan veriyi fayda sağlanabilir duruma çeviren ve pek çok alanda değişik pencerelerden bakmamızı sağlayan bir yöntem olduğunu da ifade etmişlerdir. Metodun günler ilerledikçe yayılmaya başladığını ve kullanımının arttığını belirtmişlerdir. Sağlık sektörünün son yıllarda ortaya çıkan gelişmeler sayesinde dünya ekonomisi açısından kıymetinin anlaşıldığı belirtilmiştir. Sağlık alanında oluşturulan veri tabanlarının ve sağlık sektöründe faydalanılan bilgi sistemleriyle beraber gelen yeniliklerin daha fazla ve farklı verinin muhafaza edilmesine olanak tanıdığı söylenmiştir. Bununla beraber bilgi keşfinin de bir gereksinim olabileceği savunulmuştur. Günümüz şartlarında profesyonelce iş yapan pek çok kuruluşunun bu metot sayesinde rakiplerinden bir adım önce olduğu ifade edilmiştir.

Can ve diğ. (2017), Dünya genelinde uydu verileri, alışveriş verileri, otomasyon verileri vb. sektörlerde hızlı bir şekilde artış gösterildiğini ancak elde edilen verilerin toplanmasının veya depolanmasının sorun teşkil ettiğini belirtmişleridir. Ortaya çıkan bu tarz sorunlar için veri tabanları ve dosya sistemlerinde yapılan yenilikler ile kolaylık arandığını ancak insanların doğru şekilde kullanmaması yüzünden istenilen verimin alınamadığı ifade edilmiştir. Tam da bu noktada "Veri Madenciliği" adı verilen bir yeniliğin gündeme geldiğini açıklamışlar. Özellikle mühendislik dallarında sık sık tercih edilen veri madenciliğinin günümüzde tıp dallarında elde edilen verilerin çok büyük boyutlara çıkması hasebiyle bu sektörde de kullanılmaya başlandığını ve çalışan kişilere büyük kolaylık sağladığını belirtmişlerdir.

Önder ve diğ. (2019), çağımızda gelişmeye devam eden teknolojik faaliyetlerden en mühim metotları ve verimleri sağlık alanında kendisini gösterdiğini belirtmektedir. Bulut bilişim teknolojisi, büyük veri, nesnelerin interneti, giyilebilir teknoloji vb. pek çok gelişmeye devam eden teknoloji metotlarında kullanıldığını ifade etmiştir. Dünya devletlerinin dördüncü kez gerçekleşen sanayi devrimini kaçırmamak adına birbirleriyle rekabete girdiğini ve bunun üzerine ülkelerin sağlık sektörleri de Sağlık 4.0 uygulamalarına konsantre olmaya uğraştığını açıklamışlardır. Sağlık 4.0 uygulamalarına en hızlı şekilde uyum sağlanarak sağlıkla ilgili her konuyu bu hususta yoğunlaştırarak orta ve uzun vadede bu uğraşı vermeyen ülkelere göre daha ileri seviyede olabileceklerini ve ilerleyen yıllarda yapay zekâ, makine öğrenimi ve derin

öğrenme teknolojilerinin doktorların karar verme zorluğu olan konularda hafifleyeceğini belirtmişlerdir.

Sıtkı (2020), bugün verilerdeki artış sebebiyle asıl bilgiye erişmekte çok büyük sorunlar yaşandığını ve bu yüzden veri madenciliği kavramının ortaya çıktığını anlatmıştır. Veri madenciliği kavramının en fazla tıp ve sağlık alanlarında gözlendiğini ifade etmiştir. Veri madenciliğinin, verilerin analiz edilmesiyle ortaya çıkan bilginin karar aşamasında fayda sağladığı bir model olduğunu düşünmektedir. Bu sebeple veri madenciliğinin, karar aşamasında yardımı olan bir metot olarak sağlık hizmetleri verilirken, sağlık kuruluşlarının yönetirken ve sağlık stratejileri ortaya çıkarırken faydalanılmasıyla, sağlık çalışanlarının hatasız ve faydalı yollar tercih etmesine yardım ettiğini belirtmiştir. Yapılan veri analizlerinin %72,2 oranında başarı sağladığı ve bu sayede rahat bir şekilde teşhis koyulduğu ifade edilmiştir.

Güllüoğlu (2011), veri madenciliğiyle sahip olunan verilerin genel anlamda pürüzlü ve net olmadığını, başka bir deyişle üstü kapalı olduğunu ancak kıymetli olabileceğini vurgulamıştır. Veri madenciliği kavramının tek başına bir çözüm yolu olmadığını, çözüme giderken kullanılabilecek bir araç olduğunu ifade etmiştir. İstenilen cevapları elde etmek adına karar verme esnasında yardımcı, sorunu ortadan kaldırmak için lazım olan bilgileri gün yüzüne çıkaran bir araç olarak görmektedir. Aynı zamanda veri madenciliğinin, sağlık sektöründe çalışan profesyonel kişilerin en uygun ve yeni bilgiyi elde etmesinde, en tarafsız ve optimize kolaylıkları uygulanmasında yardımcı olan bir kavram olarak nitelendirmektedir. Yine gelecekte sayısal karar verme ve iş zekâsı olarak görülen veri madenciliğinin bu hususta kendisini yeterince geliştirmiş kişilerce sağlık alanında kullanılmaya başlamasıyla sağlık alanında daha nokta atış, eldeki imkânlardan daha etkin faydalanılmasını sağlayacağını düşünmektedir.

Erdem ve Özdağoğlu (2008), veri madenciliğinin büyük boyutlardaki veri toplulukları içerisinden ilişki bağlantılarını gün yüzüne çıkartan bir metot olduğunu belirtmişlerdir. Aynı zamanda sınıflandırma, gruplandırma ve derecelendirme gibi yollar içinde veri madenciliğinde faydalanıldığını ifade etmişlerdir. Çalışmaları çerçevesinde belli bir zaman dilimi süresince Ege Bölgesinde bulunan bir araştırma hastanesine başvuran 214 bin hastanın verisinden faydalanıldığı açıklanmıştır. Veri madenciliğinde pek çok kez faydalanılan birliktelik kuralı metoduyla, veriler

içerisinde saklı kalmış fakat doktorlar için çok kıymetli olan bilgilerin açığa çıkarılacağını belirtmektedirler. Geçmişe dönük yapılan bu incelemede veri topluluklarının arasındaki bağlantıyı bulmak adına apriori algoritmasından faydalanılarak birlikteliklerin arandığı açıklanmıştır.

Kaya ve diğ. (2003), bilişim teknolojilerinden farklı farklı dallarda yararlanmaya başlandığını ve bunun iyiden iyiye yaygınlaştığını bildirmişleridir. Fayda sağlayan sektörlerden bir tanesinin de tıp olduğunu belirtmişlerdir. Bu alanda yapılan yenilikler sayesinde "Tıp Bilişimi (Medical Informatics)" isminde bir disiplinin ortaya çıkmasını sağladığını ifade etmişlerdir. Veri madenciliğinin ticari faaliyetlerde oldukça sık kullanımının ardından tıp alanında da kendisini gösterdiğini belirtmişlerdir. Veri madenciliği sayesinde tıp alanında birçok klinik inceleme lüzumu olan hem maddi hem de manevi olarak riskler taşıyan tıbbi incelemelerin yerine nispeten de olsa gelebilecek olduğuna inanmaktadırlar. 2003 yılında yapılan bir çalışma olmasına rağmen bu hususstaki fikirleri geleceği görme adına oldukça etkilidir.

Yücebaş (2018), günümüz dünyasında genetik alanında ortaya konulan çalışmalar İnsan Genom Projesi'nin sona ermesiyle beraber hız kazandığını ifade etmiştir. Bu incelemelerin bir dalı da genetik varyasyonları irdeleyerek bunların hastalıklara sebebiyet verip vermediğini inceleyen bütünsel genom ilişkilendirme faaliyetleridir. Bütünsel genom bağlantılandırma faaliyetlerinde meydana gelen verilerin çok büyük ölçekte ve oldukça çeşitli boyutlarda olmasından dolayı, kişilerin hastalıklarla bir bağlantısının olup olmadığı ve bu sonuçlarla teşhis yoluna gidilmesi için çeşitli veri madenciliği metotlarının kullanılmasıyla olduğunu bildirmiştir. Kendisinin yaptığı incelemeler sonucunda 1025 vaka ve 531 kontrol içeren melonom veri kümesi ile 2325 vaka ve 2350 kontrol içeren ve prostat kanseri veri kümesi kullanımını açıklamıştır. Bu hastalıkla alakalı profiller Karar ağacı, Naive Bayes, Destek Vektör Makinesi gibi çeşitli veri madenciliği metotları ile irdelendiğini anlatmıştır. İncelenen hastalılardan her ikisi içinde vektör makinesi metodunun en iyi sonuçları verdiğini bildirmiştir. Bahsi geçen yöntemin hastalık incelemelerinde %75,68 oranında başarı elde ettiğine değinmiştir.

Elbaşı (2006), hastalık teşhisinin doktorlar açısından mühim bir yol ayrımı olduğunu ve genelde görüntü, sayılar kaynaklar sayesinde karar verildiğini bu sebeple de hata yapma olasılıklarının göz ardı edilemeyeceğini ifade etmiştir. Nitekim görüntü

analizlerinde çeşitli durumlar sebebiyle oluşan verilerin yanlış teşhise yol açabileceği bildirilmiştir. Günümüz tıp dünyasında bilgisayar ve teknolojik ürünlerin oldukça rağbet gördüğünü belirtmiştir. Yenilikler sayesinde veri madenciliği kavramının da kullanılmaya başlandığını, bunun sonucunda pek çok hastalığın teşhisinde veri madenciliğinden yararlanıldığını ifade etmiştir. Doktorların en kolay biçimde karar verebilmesi adına meme kanseri, idrar sistemi hastalıkları ve hepatit hastalığı için veri madenciliği metotlarından yararlanıldığını belirtmiştir. Sınıflandırma ve kural çıkartma yoluyla yapılan çalışmalar sonucunda bilgisayar desteğiyle ilerleyen karar verme sisteminin %95 üzerinde başarı elde ettiğine vurgu yapılmıştır.

Gökbay ve Gökçek (2022), sağlık kavramını kişilerin hastalık ve sakatlıklarının olmamasıyla beraber bedensel, ruhsal ve toplumsal olarak tam bir bütün şeklinde ifade etmiştir. Kişilerin sağlıklı ve güzel bir hayat sürmesinin en doğal haklarından bir tanesi olduğunu belirtmişlerdir. Bu sebeple de hastalığın tedavi edilmesinin ya da iyileştirme hizmetlerinin devletin görevi olduğunu belirtmişlerdir. Devletlerin kişilerin karşı karşıya kalabileceği sağlıksal tehlikelere karşı kamu sağlık sistemiyle güven vermeleri gerektiğini ifade etmişlerdir. Endokrin sistem hastalıklarının da halk arasında sıkça rastlanan, kronik olarak ortaya çıkan, teşhis ve tedavisi için uzunca sürelere ihtiyaç olan, yanlış teşhis konulması durumunda hastanın hem maddi hem de manevi olarak yıpranmasına yol açacağını belirtmişleridir. Endokrin hastalıkların halk arasında sıkça görülmesi, pek çok fiziksel ve ruhsal belirtinin rahatlıkla karıştırılabilmesi sebebiyle teşhis ve tedavi sürecinde doktor tarafından yanlış metotlar kullanıldığına vurgu yapılmıştır. Bu sebeple uygun karar ve hızlı adım atılabilmesi için teknolojinin nimetlerinden faydalanılması gerektiği belirtilmiştir. Tam bu noktada klinik karar destek sistemi (KKDS) hekim-hasta ilişkisinin uzman sitemlerden yardım alınarak kolaylıkla halledildiği ifade edilmiştir. Endokrin hastalıklar bakımından belirtilerin benzer olması hasebiyle karışıklıklar yaşandığını, klinik karar destek sisteminin bu konuda işleri büyük ölçüde kolaylaştırdığını belirtmişlerdir.

Akgül ve diğ (2020), hastalık teşhisinin tıp dallarında en fazla rastlanan ve en mühim sorunlardan birisi olduğunu belirtmişlerdir. Belli başlı bir hastalığın çeşitli türlerinin olması ve başka hastalıklarla yakın belirtiler sergilemesi hastalık teşhisinde zor duruma sokan hususlardan birisi olarak nitelendirilmiştir. Tiroit hastalığının bir

çeşidi olan hipotiroidin de bu nedenlerden ötürü teşhisinin geciktiği ve hastanın yaşam kalitesinin düştüğü ifade edilmiştir. Bu noktada yapılan incelemeler sonucunda veri madenciliğinden yararlanılarak bir kolaylık elde etmeyi amaçlamışlardır. Veri madenciliğinin tıp alanında kullanılmaya başlanmasıyla insanların hayat standartlarını yükseltmek için pek çok inceleme yapıldığı ifade edilmiştir. Veri madenciliği sayesinde eldeki bilgiyi yorumlama ve karar verme kabiliyetiyle literatürdeki pek çok hastalık adına incelemeler yapıldığını ve başarılı veriler elde edildiğini bizlere sunmuşlardır. Bu hususta doku düzeyinde tiroit hormonunun yetersiz oluşu ya da ender olarak etkisizliği sonucunda meydana gelen hipotiroidi hastalığıyla ilgili incelemeler yapılmış ve hastalık tanısında %92 seviyesinin de üstünde başarı kazanılmıştır. Yapılan çalışmada çeşitli örnekleme metotlarından yararlanılarak Lojistik Regresyon, K En Yakın Komşu ve Destek Vektör Makinesi Algoritmasıyla hipotiroidi hastalığına teşhis konulmaya çalışıldığı ifade edilmiştir.

Oğuztürk (2018), diyabet olan hasta sayısının dünya çağında gittikçe arttığını ifade ederek ülkemizde bu sayının 10 milyonu aştığını belirtmiştir. Diyabetinde kendi içerisinde farklı çeşitleri olduğunu ve bazılarının erken teşhis edilmesinin hayati derecede önemli olduğunu belirtmiştir. Bunların pre diyabet ve tip 2 diyabet olduğunu da açıklamıştır. Bu sebeple diyabetin teşhis edilmesi adına kolaylıkla yapılabilecek, çabuk ve nokta atış tanı koyma araçlarına gereksinim olduğunu ifade etmiştir. Diyabetin erkenden teşhis edilmesi için makine öğrenimi algoritmalarına dayanan kolay, hızlı ve hassas bir tahmin aracının geliştirilmesi gerekliliğinin kaçınılmaz olduğunu savunmuştur. Yaptığı çalışmalarda toplamda 2657 denekten 1860 algoritmayı geliştirmek için kullanmıştır. Kalan denekleri ise geliştirilen algoritmanın doğruluğunu test etmek için kullanmıştır. Aynı zamanda algoritmanın doğruluğunu kontrol edebilmek adına farklı optimizasyonlarda kullanmıştır. Yapılan çalışmalar sonucunda %97,2'lik bir başarı elde etmiştir.

Ertuğrul ve diğ (2012), sağlık ve tıp alanının bugün en fazla bilgiye muhtaç olan araştırma alanlarından olduğunu belirtmişlerdir. Son yıllarda özellikle sağlık veri şekilleri, standartlar ve kodlama sistemlerinde yaşanan gelişmeler yardımıyla hastanelerde ve diğer sağlık kuruluşlarında mühim yenilikler ortaya çıktığını ifade edilmiştir. Sağlık sektöründe gizlenen bilgilerin ortaya çıkarılması hayati derecede ehemmiyet kazanmıştır. Buna örnek veren yazarlar hastalıkların arasında gizlenmiş

tecrübe ile elde edilecek bilgilerin önceden keşfi gibi hususlarda veri madenciliği önem arz etmektedir. Tıp alanında bu bağlamda pek çok verinin olduğunu ve bu verilerin hayati önem taşıdığı ifade edilmiştir.

Çerkezi (2013), son zamanlarda hastanelerde bulunan hastane bilgi sistemlerinde hastalara ait tıbbi bilgilerin şiştiği ve bu artışın devam ettiğini belirtmiştir. Dijital ortamlarda bu bilgilere sahip olunmasıyla hastalara daha güzel hizmet verilebileceği ifade edilmiştir. Böyle bilgilerin içerisinde mutlaka aralarda saklanmış ancak çok işe yarayacak bilgilerinde var olduğu belirtilmektedir. Bu kıymetli verilerin işlenip kullanılabilmesi için veri madenciliği yolunun seçilmesi gerektiği vurgulanmıştır. Yapılan incelemeler ve çalışmalar sonucunda elde edilen veriler K-en yakın komşu, ağırlıklı oylama KNN ve Bayes algoritmaları ile sınıflandırma yapılmıştır. Veriler ışığında doktorların hastalık teşhisinde büyük kolaylık yaşadığı ifade edilmiştir. Yapılan bütün çalışmalarda yüksek derecede başarı elde edinildiği belirtilmiştir.

Kökver (2012), bugün bir hastalık için yapılacak doğru ve süratli teşhisin çok mühim olduğunu ifade etmiştir. Doktorların etkin ve hızlı bir biçimde teşhis yapabilmelerine yardım edebilmesi adına veri madenciliğinin gözde bir metot olduğunu belirtmiştir. Çalışmasında hastaların demografik değerlerini ve kan değerlerini araştırarak, hasta kişiye cerrahi bir müdahale yapılmadan teşhis koyulabildiği anlatılmıştır. Ayrıca hastaneye farklı bir şikâyetle gidilmesine rağmen hipertansiyon veya başka bir teşhis konulabileceği anlatılmıştır.

Salazar ve diğ (2018), veri madenciliğinin, gelecek yıllarda bir durumun görüntüsünü tahmin etmek için kalıpları, bağlantıları ve biçimleri belirlemek adına, boyutları çok yüksek verilerin incelenmesiyle ortaya çıktığını açıklamışlardır. Ancak bu değerli bilgileri elde edebilmek adına sınıflandırma metotlarını, ilişkilendirme kaidelerini, regresyon çeşitlerini ve küme analizlerini kullanmak gerektiğini ifade etmişlerdir. Günümüzde bilgisayar teknolojilerinin hayatımızdaki yerinin çok önemli olduğunu ve sağlıkta faydalanılması ve otomasyonunun gelişmiş ülkelerde gittikçe artış gösterdiğini ifade etmişlerdir. Hastalıkların teşhis ve tedavisi için kullanılabileceği gibi aynı zamanda sağlık sistemlerinin organizasyonuna da yardımcı olduğunu belirtmişlerdir. Yapılan pek çok çalışmanın, makine öğrenimiyle sadece belli hastalık tahminlerinin yöneltilemeyeceğini; aynı zamanda kişisel sağlıklarıyla

ilgili tahminde bulunulabileceğini anlatmışlardır. Tip 2 Diyabeti tahmin etme gücünün yüksek olduğunu belirtmişlerdir.

Savaş ve diğ. (2012), bütün ülkelerde olduğu gibi Türkiye'de de veri madenciliğine verilen değerin giderek arttığını belirtmişlerdir. Bu sebeple de veri madenciliğin kullanım alanlarının giderek arttığını ifade etmişlerdir. Ülkemizde yapılan veri madenciliği incelemelerinin genelde eğitim, ticaret, mühendislik, bankacılık, borsa ve tıp alanında gerçekleştiğini belirtmişlerdir. Tıp dalında ortaya koyulan araştırmaların hastalık belirtileri ve elde olan belirtilerden bir kalıp oluşturulmaya çalışıldığını anlatmışlardır. Ülkemizde tıp alanında yapılan veri madenciliği çalışmalarının yetersiz kaldığını ve daha fazla çalışma yapılması gerektiğini belirtmişlerdir.

2.2 Tıp Alanında Veri Madenciliği ile Yapılan Çalışmalar

Tıp alanında gelişen teknolojinin ve bilişim sistemlerinin yardımıyla yapılan birçok tıbbi işlemin kayıt altına alınması ve erişilmesi kolaylaşmıştır. Sağlık sektöründe kullanılan Sağlık Bilişim Sistemleri, kalite yönetimi, hastalığın teşhisi ve tedavisi, tıbbi belgelendirme ve bilgi yönetimi gibi konularda veri akışı sağlamaktadır. Bu sistemlerin yardımıyla hekimlerin hastalıkları teşhis etmesinde kolaylık sağlamaktadır (Farboudi 2009).

Hekimlerin, anne karnındaki bir bebeğin down sendromlu olması, bazı kanser türlerinin erken teşhisi, çeşitli otoimmün hastalıkların vücutta semptom göstermediği için hastalık teşhisinin belirlenmesi gibi bazı hastalıklarda teşhis tanısı koyabilmesi zor olabilmektedir (Silahtaroğlu 2020). Hastalık teşhisinin doğru yapılması erken teşhis ve tedaviye ayrılan süreyi ve bu süreçte oluşan maliyetleri azaltmaktadır. Veri madenciliği yöntemleri birçok alanda olduğu gibi sağlık alanında da son yıllarda sıkça kullanılmaya başlanmıştır. Kan testi, operasyon, dış bulgular ve hekimin görüşleri gibi verilerden oluşturulmuş veri tabanları, veri madenciliği algoritmaları tarafından incelenip, gelecekteki teşhis ve tedavi hizmetlerinin kalite artırımını sağlayan bir yöntem olarak değerlendirilebilir.

Tıp alanında veri madenciliği, hastalıkların erken teşhisi, tedavi planlaması, kanser teşhisi ve sağlık hizmetlerinin yönetimi gibi birçok alanda kullanılmaktadır. Tıp alanında veri madenciliği ile yapılan bazı çalışmaların aşağıda verilmiştir.

Kanser Tanısı ve Prognozu: Veri madenciliği, kanser tanısı ve prognozunda kullanılan birçok yöntemde kullanılmaktadır. Örneğin, bir çalışmada, göğüs kanseri hastalarının tedavisi için karar vermede kullanılan bir veri madenciliği modeli geliştirilmiştir. Bu model, hastalık aşamasını, kanser türünü ve hastanın yaşam tarzı faktörlerini kullanarak en uygun tedavi planını belirlemeye yardımcı olmaktadır (Ma, J. ve diğ, 2019).

Diyabet Hastalığının Teşhisi: Diyabet hastalığı, dünya genelinde yaygın bir sağlık sorunudur. Veri madenciliği, diyabet hastalığının teşhisinde ve tedavi planlamasında kullanılabilmektedir. Diyabet hastalarının kan şekeri düzeylerini takip etmek için kullanılan bir cihazdan toplanan verileri kullanarak, diyabet hastalığının erken teşhisinde kullanılabilecek veri madenciliği modelleri geliştirilmiştir.

Sağlık Hizmetlerinin Yönetimi: Veri madenciliği, sağlık hizmetlerinin yönetimi için de kullanılmaktadır. Örneğin, hasta verileri analiz edilerek, hasta bakımının geliştirilmesi ve hastane kaynaklarının daha etkili bir şekilde kullanılması için öneriler sunulabilir.

İlaç Yan Etkilerinin Tespiti: Veri madenciliği, ilaç yan etkilerinin tespiti için de kullanılabilmektedir. Bir çalışmada, bir ilaç firmasının verileri kullanılarak, bir ilacın yan etkilerinin tespit edilmesine yardımcı olan bir veri madenciliği modeli geliştirilmiştir (Sampathkumar,2014).

Tedavi önerileri: Veri madenciliği, hastaların tedavi planlarının oluşturulması ve tedavi önerilerinin geliştirilmesi için kullanılabilir. Örneğin, hastalık belirtileri ve tedavi sonuçları hakkındaki veriler analiz edilerek, tedavi önerileri yapılabilir.

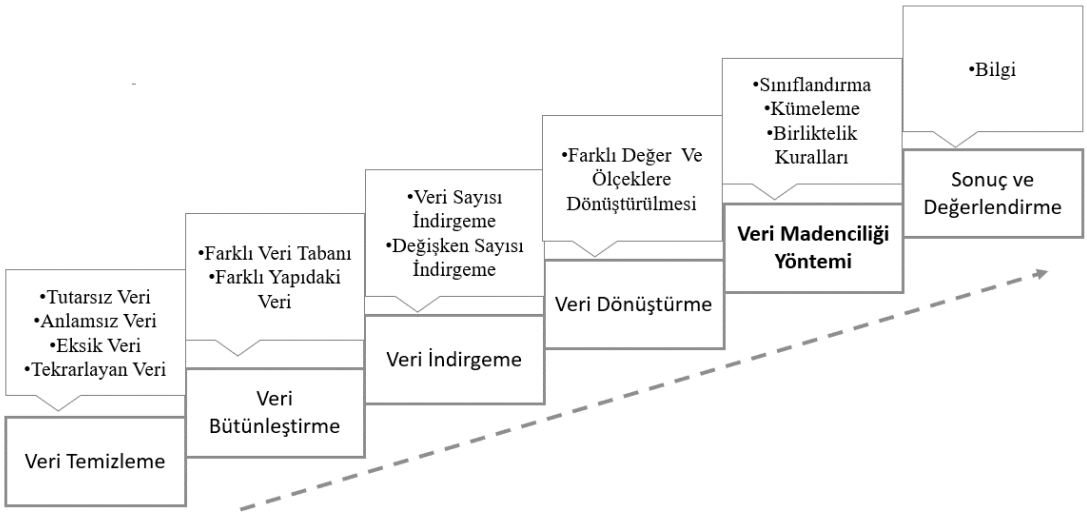
İlaç keşfi: Veri madenciliği, yeni ilaç keşfi için kullanılabilir. Örneğin, ilaç bileşenlerinin analizi ve mekanizmalarının keşfi için büyük veri kümeleri analiz edilebilir (Dara ve diğ,2022).

3. VERİ MADENCİLİĞİ

3.1 Veri Madenciliği Tanımı

Veri Madenciliği (VM); büyük miktardaki veriler içerisinden, anlamlı ve kullanılabilir bilgileri açığa çıkarmaktır. VM, verilerin ilgili olduğu sorun alanlarına yanıt bulmasıyla birlikte araştırmacıya yeni bir perspektif yaratmıştır. Bu yöntemin kullanım yaygınlığı hızla artmaya devam etmiş ve etmektedir (Koyuncugil ve Özgülbaş 2010). Teknolojinin ilerlemesi ile günlük yapılan birçok işlemin elektronik ortamda kayıt altına alınmasına, kolayca depolanmasına ve erişilmesine olanak sağlamaktadır. Veri tabanları içerisindeki verilerden anlamlı bilgiler elde edebilmek için bu verilerin uzman kişiler tarafından geliştirilen analiz algoritmaları ile incelenmesi gerekmektedir (Coşlu 2013). Veri madenciliği, veri tabanları içerisindeki daha önceden fark edilmemiş örüntülerin, ilişkilerin, farklılıkların, benzerliklerin, kuralların ve istatistiksel olarak ilişkili olan yapıların yazılım teknikleri ile keşfedilmesidir (Terzi ve diğ 2011).

Veri madenciliği tek bir adım değil, birçok adımdan oluşan bir süreç olup, araştırmacılar için yeterli derecede zaman, mekân ve yazılım bilgisi gerektirmektedir. İstenilen sonuca daha hızlı ve doğru bir şekilde ulaşabilmek için bu süreçte bulunan adımların doğru bir şekilde yerine getirilmesi gerekmektedir. Üzerinde inceleme yapılan veri tabanının özelliklerine hakim olunmaması durumunda hazırlanan algoritma araştırmacıya yeterince fayda sağlamayacaktır. Bu sebeple, veriyi analiz etmeden önce, veri tabanının özellikleri detaylı olarak işlenmelidir. Veriler, bir veya birden fazla kaynaktan oluşmaktadır. Oluşan bu veriler veri depolarına yerleştirilir ve verinin durumuna göre ön işleme yapılır. Bu ön işleme sürecinde, gereksiz veya aykırı veriyi silme, eksik veriyi ekleme, mevcut veriler üzerinde normalizasyon yapmak gibi işlemler olabilmektedir. Ön işlemesi yapılmış veriler, kurallar veya örüntüler şeklinde çıktı üreten veri madenciliği algoritmasına aktarılmaktadır. Algoritmanın çıktısı yorumlanır ve elde edilen çıktı anlamlı veridir (bkz. Şekil 3.1). Algoritmanın sonucunda anlamlı veri elde edilmediyse, veri hazırlama sürecine geri dönülür ve aynı adımlar tekrar edilir.



Şekil 3.1: Veri madenciliği adımları.

Veri analizi sürecinde izlenen adımlar genellikle şöyledir (Shearer, 2000):

1. Problemin tanımlanması: Çalışmanın hangi hedefe yönelik olacağının ve elde edilecek sonuçların başarı düzeylerinin nasıl ölçüleceğinin tanımlanmasıdır. Yapılacak çalışmanın konusunun açık ve anlaşılır bir şekilde tanımlanması, mevcut durumun değerlendirilmesinde ve çalışma planının oluşturulmasında kolaylık sağlamaktadır.

2. Verilerin hazırlanması: Kullanılacak olan verinin kalitesi, veri madenciliği algoritmasının çıktısını etkileyen önemli konulardan biridir. Algoritma sonucunun başarısının artırılması için veri, ön işleme basamaklarından geçmelidir. Hatalı olabilecek bir algoritma girdisi verileri, kullanıcıyı hatalı ve kullanılamayan çıktıya götürecektir (Kayaalp, 2007). Veri ön işleme basamakları kendi içinde veri temizleme, veri bütünleştirme, veri indirgeme ve veri dönüştürme adımlarına sahiptir. Veri tabanı içerisinde tutarsız, eksik veya tekrarlayan veriler varsa bu hatalar giderilmelidir. Yapılan çalışmada birden fazla veri tabanı bulunabilmektedir, bu veri tabanlarını birleştirmek veya aynı yapıda değerlendirilmek istenebilir. Bu aşamada veri bütünleştirme işlemi yapılır. Çok fazla sayıda olan veriler işlem süresini artırabilir. Sonucu değiştirecek bir durum oluşmuyorsa veri indirgeme yöntemiyle veri sayıları veya değişken sayıları indirgenebilir. Veri dönüştürme, verileri farklı değerlere veya ölçeklere dönüştürme işlemidir. Bu adımlar her zaman yapılmak zorunda değildir.

Gerekli işlemler yapıldıktan sonra veriler ve veri tabanı kullanılacak algoritma için hazır hale gelmiş olacaklardır.

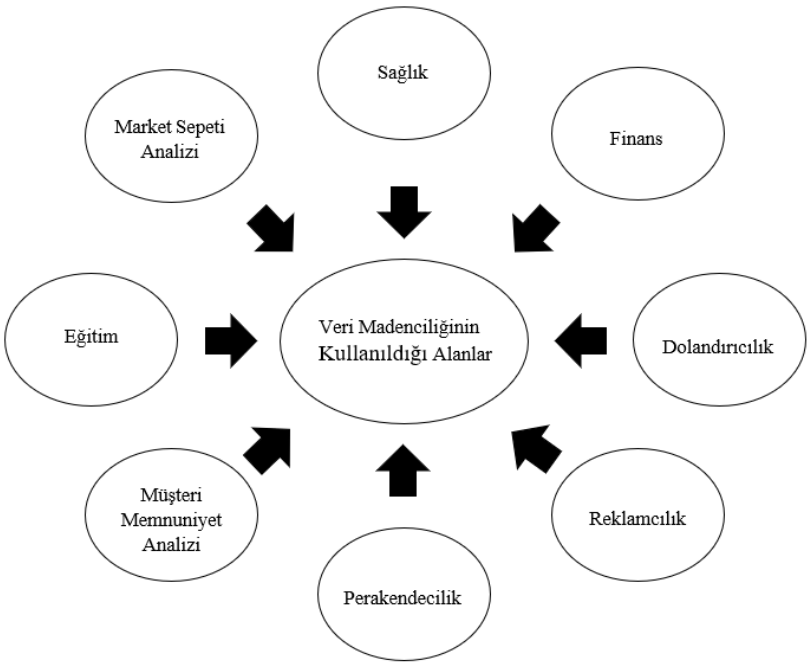
3. Modelin kurulması ve değerlendirilmesi: Veri tabanına göre en uygun modeli kurmak çok sayıda modelin kurularak denenmesi ile mümkündür. Bu sebeple veri hazırlama ve model kurma adımları, en iyi sonuç bulununcaya kadar yinelenebilir (Savaş ve Topaloğlu 2011). Bu aşamada hangi algoritma modelinin en uygun olduğu sorusunu cevaplamak oldukça zordur. Bu nedenle çeşitli algoritmalar oluşturarak doğruluk derecelerine göre en uygun model seçilmelidir.

3.2 Veri Madenciliğinin Kullanıldığı Alanlar

Veri madenciliği birçok alanda, birden fazla uygulamada kullanılmaktadır (bkz. Şekil 3.2). Bu uygulama alanlarından bahsedecek olursak; Müşterilerin satın alma eğilimlerinin belirlenmesi, pazarlama kampanyalarının planlanması, mevcut müşterilerin memnuniyet analizi, yeni müşterilerin kazanılması için geliştirilecek stratejilerinin oluşturulması, market sepeti ve çapraz satış analizlerinin yapılması, müşteri ilişkileri yönetimi ve satış tahminleri gibi pazarlama yönetiminde; kredi kartı dolandırıcılıklarının tespiti, kredi taleplerinin değerlendirilmesi, risk analizleri ve risk yönetimi gibi bankacılık sektöründe; ilaç geliştirme, hastalıkların teşhisi, tedavi sürecinin belirlenmesi, DNA sıra analizi ile hastalıklara neden olan gen sıralamasını belirlemek gibi sağlık ve biyoloji alanında; internet işlemleri dolandırıcılığın tespit edilmesi, bilgisayar sistemlerindeki ve bilgisayar ağlarındaki şüpheli işlemlerin tespit edilmesi, parmak izi ve yüz tanıma sistemleri ile kimlik tespiti ve yapay zekâ uygulamaları gibi bilişim ve mühendislik alanında; duygu analizleri, seçim propagandaları, reklam yönetimi gibi sosyal medya uygulamalarda ve kariyer planlaması, öğrencilerin katalog durumu ve ileri dönemlerde alacak derslerdeki başarı durumu tespiti gibi eğitim alanlarında kullanılmaktadır. Veri madenciliğinin kullanıldığı alanlar kısaca aşağıda verilmiştir:

- Pazarlama: Pazarlama alanında, müşteri davranışlarının analiz edilmesi ve pazarlama stratejilerinin belirlenmesinde kullanılan veri madenciliği yöntemleri bulunmaktadır.

- Sağlık: Hastalık teşhisi, ilaç yan etkilerinin tespiti, hastane kaynaklarının verimli kullanımı gibi alanlarda,
- E-ticaret: Alışveriş tercihlerinin analiz edilmesi, müşteri sadakati, ürün önerileri ve stok yönetimi gibi konularda,
- Finans: Kredi riski analizi, dolandırıcılık tespiti, hisse senedi fiyat tahmini gibi alanlarda,
- Ulaşım: Trafik akışının tahmini, araç konumlandırma, seyahat planlama gibi alanlarda,
- Tarım: Hasat tahmini, toprak verimliliği analizi, ürün kalitesi ve hastalıklarının tespiti gibi konularda,
- Enerji: Enerji tüketimi tahmini, enerji tasarrufu, güneş paneli performansı analizi gibi konularda veri madenciliği kullanılmaktadır.



Şekil 3.2: Veri madenciliğinin kullanım alanları.

3.3 Veri Madenciliği Modelleri

Veri madenciliği, birçok farklı alanda kullanılan bir yöntemdir ve farklı yöntemler ve teknikler kullanılarak verilerden faydalı bilgiler elde edilir. Bu bilgiler daha sonra kullanıcılara öneriler sunmak, tahminler yapmak veya karar verme süreçlerinde yardımcı olmak için kullanılabilir. Veri madenciliği yöntemleri,

genellikle büyük veri kümeleri üzerinde çalışır ve bu verilerin işlenmesi, analizi ve modelleme süreçlerini içerir.

Veri madenciliği modelleri, genel olarak iki temel öğrenme şekliyle sınıflandırılır: gözetimli ve gözetimsiz öğrenme. Gözetimli öğrenme, giriş özniteliklerini kullanarak çıktı özniteliklerini tahmin etmek için bir model oluşturur. Bu öğrenme şekli, veri kümesinin bir kısmının eğitim için kullanılması ve geri kalan kısmının test için ayrılmasıyla gerçekleştirilir. Model, eğitim verilerinden öğrenilen desenleri genelleştirmek için test verilerine uygulanır. Bu şekilde, modelin doğruluğu ölçülür ve iyileştirilir.

Gözetimli öğrenme yöntemi, bir çıktı özniteliği bulunması gerektiği durumlarda kullanılır. Örneğin, bir hastanın hastalık riskini tahmin etmek için, hastalık var veya yok olarak etiketlenmiş bir veri kümesi kullanılır. Model, hastalık riskini tahmin etmek için çeşitli girdi özniteliklerini kullanır

Gözetimsiz öğrenme, bir çıktı özniteliğinin olmadığı durumlarda kullanılır. Bu yöntem, veri kümesindeki yapıları ve kalıpları keşfetmek için kullanılır. Gözetimsiz öğrenme yöntemleri, veri kümesindeki benzer özelliklere sahip verileri gruplandırabilir veya veri kümesindeki benzer özellikleri tespit edebilir. Örneğin, bir müşteri tabanının segmentasyonu, gözetimsiz öğrenme yöntemleri kullanılarak yapılabilir.

Gözetimli ve gözetimsiz öğrenme yöntemleri, veri madenciliğinde yaygın olarak kullanılmaktadır. Veri setlerindeki yapıları ve kalıpları keşfetmek için gözetimsiz öğrenme yöntemleri kullanılırken, gözetimli öğrenme yöntemleri ise öngörü yapmak veya belirli bir sonuç için tahmin yapmak için kullanılır.

Veri madenciliği, otomatik veya manuel modellerle gerçekleşen, araştırmadaki çıkış verisine bağlı yinelemeli bir süreçtir. Bu modellerin temel olarak iki hedefi vardır: Tahminleyici (Predictive) ve Tanımlayıcı (Descriptive). Tahminleyici modeller veri seti tarafından açıklanan sistemin modelini üretirken, tanımlayıcı modeller ise mevcut veri seti içerisindeki kalıpları ve ilişkileri ortaya çıkarır (Kantardzic 2011).

Tahminleyici modeller, gelecekteki bir olayın olasılığını veya bir değerin tahminini yapmak için kullanılır. Bu modeller, sınıflandırma ve regresyon olarak iki

ana gruba ayrılır. Sınıflandırma modelleri, bir veri örneğinin bir sınıfa ait olup olmadığını tahmin etmek için kullanılır. Regresyon modelleri ise sayısal bir değerin tahmin edilmesi için kullanılır (Ayık ve diğ. 2007).

Tanımlayıcı modeller, verilerin mevcut durumunu anlamak ve kalıpları keşfetmek için kullanılır. Bu modeller, kümeleme, faktör analizi, birliktelik kuralları, yoğunluk tahmini ve zaman serisi analizi gibi farklı teknikler içerir. Kümeleme modelleri, benzer özelliklere sahip veri noktalarını gruplandırmak için kullanılır. Faktör analizi, verilerin altında yatan yapıyı anlamak için kullanılır. Birliktelik kuralları, veri kümesindeki ilişkileri keşfetmek için kullanılır. Yoğunluk tahmini, belirli bir bölgedeki yoğunluğu tahmin etmek için kullanılır. Zaman serisi analizi, zaman içindeki değişiklikleri incelemek ve gelecekteki trendleri tahmin etmek için kullanılır.

Tahminleyici modeller "Bu işlemde dolandırıcılık var mıdır?" gibi sorulara yanıt ararken, tanımlayıcı modeller "Çocuk bezi alan bir müşterinin, mama alma olasılığı diğerlerinden üç kat fazladır." gibi sonuçlar çıkarma amaçlı kullanılır (Silahtaroğlu 2008).

Veri madenciliği modelleri kullandıkları veri çeşidine göre kategorilere ayrılırlar. Birçok kaynak veri madenciliği modelleri için farklı gruplandırma yapmışlardır. Han ve diğ (2012) veri madenciliği modellerini beş kategoriye ayırmışlardır;

- Tanımlama ve Ayrılma,
- Sık Öğe Seti, Birliktelik ve Korelasyon Analizi,
- Sınıflandırma ve Regresyon,
- Kümeleme Analizi,
- İstisna Analizidir.

Roiger (2003) dört kategoriye ayırmıştır. Bunlar:

- Market Sepeti Analizi,
- Sınıflandırma,
- Tahminleme, ve Kümelemedir.

Bramer (2007), dört kategoriye ayırmıştır. Bunlar:

- Sınıflandırma,
- Regresyon Analizi,
- Birliktelik Kuralları,
- Kümelemedir.

Bu bölümde VM modelleri için en yaygın kullanılan dört temel model incelenmiştir. Bunlar; Regresyon ve Sınıflandırma, Kümeleme, Birliktelik Analizleri ve Ardışık Zamanlı Örüntülerdir.

3.3.1 Regresyon ve Sınıflandırma

Regresyon analizi, bağımlı değişken ve bir ya da daha fazla bağımsız değişken arasındaki ilişkiyi anlamak için kullanılan bir istatistiksel yöntemdir. Bu yöntem, bağımlı değişkenin sayısal bir değer aldığı durumlarda kullanılır. Regresyon analizinde amaç, bağımlı değişkenin bağımsız değişkenler tarafından açıklanabilen varyasyonunu açıklamaktır. Günümüzde en yaygın olarak doğrusal ve doğrusal olmayan regresyon sistemleri kullanılmaktadır.

Doğrusal regresyon analizi, bağımlı değişken ve bir ya da daha fazla bağımsız değişken arasındaki ilişkiyi doğrusal bir denklemle ifade eder. Bu denklem, bağımsız değişkenlerin katsayıları ve bir sabit terim içerir. Doğrusal regresyon analizi, hem tek değişkenli (tek bağımsız değişken) hem de çok değişkenli (çoklu bağımsız değişken) modellerde kullanılabilir.

Doğrusal olmayan regresyon analizi, bağımlı değişken ve bağımsız değişkenler arasındaki ilişkinin doğrusal olmadığı durumlarda kullanılır. Bu yöntem, veri setindeki doğrusal olmayan yapıları modellemek için kullanılır. Örnek olarak, çok değişkenli doğrusal olmayan regresyon analizi, sinir ağları ve destek vektör makineleri (SVM) gibi yöntemleri içerir.

Tahminleyici modellerde kullanılan teknikler içerisinde en yaygın kullanılan modeller regresyon ve sınıflandırma modelleridir. Sınıflandırmada bağımlı değişken

olan sonuç bölümü kategorize edilirken, regresyon analizinde bağımlı değişken, bağımsız değişkene bağlı sebep-sonuç ilişkisini gösteren bir değerdir.

Sınıflandırma modelleri, bağımlı değişkenin bir kategorik değişken olduğu durumlarda kullanılır. Sınıflandırma modelleri, verileri önceden belirlenmiş sınıflara ayırmak için kullanılır. Bu modeller, öğrenme algoritmaları aracılığıyla, verileri belirli sınıflara ayırmak için belirli özellikleri kullanır.

Regresyon modelleri ise, bağımlı değişkenin sayısal bir değer olduğu durumlarda kullanılır. Bu modeller, verileri önceden belirlenmiş sınıflara ayırmak yerine, sayısal bir değer tahmini yapmak için kullanılır. Örnek olarak, bir regresyon modeli, belli bir evin fiyatını, o evin özelliklerine (boyutu, yeri, vb.) bağlı olarak tahmin etmek için kullanılabilir.

Örneğin, bir sınıflandırma modeli, mağaza müşterilerinin mutlu veya mutsuz olma durumlarını, bir kişinin boy ve kilosuna göre beden numarasını, en basitinden bir evet/hayır problemini kategorize etmek için kurulurken, regresyon modelleri yıllık gelir tahmini, hava durumu tahmini veya borsa tahmini gibi problemleri çözmek için kurulmaktadır.

Sınıflandırma ve regresyon modellerinde yaygın olarak; Bayes sınıflandırma, k-en yakın komşu, karar ağaçları, destek vektör makineleri ve yapay sinir ağları, teknikleri kullanılmaktadır.

3.3.1.1 Bayes Sınıflandırma

Bayes sınıflandırma, örneklerin bir dizi özellikle tanımlandığı ve her özelliğin bir sınıf etiketi ile ilişkilendirildiği bir yöntemdir. Veri tabanında mevcut, sınıflanmış verileri kullanarak yeni girilen verinin bu sınıflardan herhangi birine girme olasılığını hesaplamaktadır. Bayes Teoremi Denklem (3.1) eşitliği şeklinde ifade edilebilir:

$$\mathrm{P}(C_m|x_i) = \frac{P(x_i|C_m)P(C_m)}{P(x_i)} \quad m, i = 1,2,3 \tag{3.1}$$

Burada m adet sınıf, i adet veri olduğu kabul edilmiştir. $P(C_m|x_i)$, x_i 'nin C_m sınıfında olma olasılığını; $P(x_i)$, x_i değerinin veri tabanında bulunma sıklığını, $P(C_m)$ ise C_m sınıfının veri tabanında bulunma sıklığını göstermektedir. Eşitlik Denklem (3.2)'de verilmiştir.

$$P(x_i) = \sum_{j=1}^{m} P(x_i|C_j)P(C_j) \tag{3.2}$$

Bayes sınıflandırması, özellikle çok sayıda özellik ve sınıfı olan veri kümelerinde etkili bir yöntemdir. Ancak, Bayes sınıflandırması, sınıflar arasında kesin sınırlar olmadığı durumlarda ve sınıflar arasındaki özelliklerin örtüştüğü durumlarda doğru sonuçlar veremeyebilir.

3.3.1.2 K-En Yakın Komşu

k-NN algoritması, sınıflandırma ve regresyon problemlerinde kullanılan temel ve basit tekniklerinden birisidir. Temel fikir, bir veri noktasını etiketlendirmek veya değerini tahmin etmek için komşularının etiketlerine veya değerlerine dayanmaktadır. Sınıflandırma yapılırken veri tabanında bulunan her ikili veri arasındaki uzaklığı hesaplar. Bu uzaklık Euclid veya Manhattan mesafesi gibi bir ölçüm yöntemi kullanılarak hesaplanabilir. Veriler arasındaki mesafeyi ölçerken yaygın olarak kullanılan mesafe Euclid uzaklığıdır ve aşağıdaki Denklem (3.3)'te tanımlanmaktadır (Jiang ve diğ. 2007):

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{3.3}$$

Algoritmada *k* değeri önceden seçilmelidir. Bir veri için, diğer verilerden k adet dikkate alınır. Bu *k* değerinin çok büyük seçilmesi birbirine benzemeyen verilerin bir araya toplanmasına, çok küçük seçilmesi ise başka bir sınıfa benzediği halde bu sınıflardan ayrı tutulmasına veya yeni bir sınıf açılmasına neden olmaktadır.

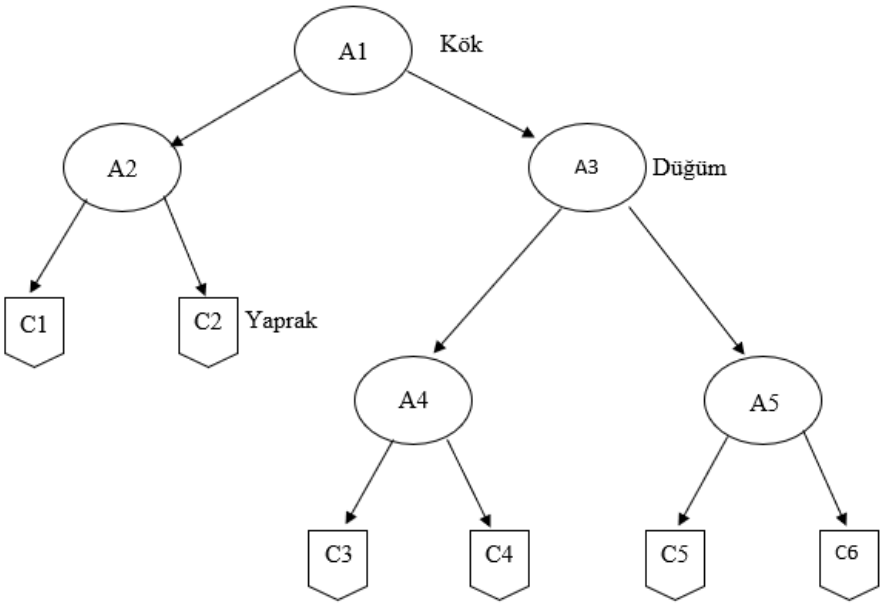
k-NN algoritması oldukça basit ve yüksek doğruluk oranlarına sahip olabilir. Ancak, çok sayıda veri noktası ve boyutlu veri kümeleri için hesaplama gücü gereksinimleri oldukça yüksek olabilir. Ayrıca, veri kümesindeki gürültülü veya yanlış verilerin tahminleri etkileyebileceği ve performansı düşürebileceği bir dezavantajı da bulunmaktadır.

3.3.1.3 Karar Ağaçları

Karar ağaçları regresyon ve sınıflandırma problemlerinde sıklıkla kullanılan ağaç tabanlı bir algoritma türüdür. Veri setindeki verileri kullanarak bir karar ağacı modeli oluşturan ve bir dizi karar düğümü kullanarak veri örneklerini sınıflandıran bir sınıflandırma yöntemidir. Bu yöntemde, veri setindeki örnekler karar ağacının kök düğümünden başlayarak, belirlenmiş olan özelliklerin öncelik sırasına göre birbirleriyle karşılaştırılarak aşağıdaki düğümlere yönlendirilirler. Her düğüm, bir karar kuralına dayanarak veri örneklerini bir alt düğüme yönlendirir veya sınıflandırır. Karar ağacı, en alt seviyedeki düğümlerde, belirli bir sınıfa ait örneklerin sınıflandırılabildiği yaprak düğümlerine ulaşır.

Diğer yöntemlerle kıyasla karar ağaçların uygulanması ve anlaması daha kolaydır (Agrawal,1993). Karar ağaçları aslında birçok "eğer (if then)"lerden oluşmaktadır. Bu algoritma sınıflandırmanın gerçekleşmesi için temel olarak iki adımdan oluşmaktadır; öncelikle ağacın oluşturulması ve daha sonra veri tabanındaki her bir verinin bu ağaca entegre edilmesiyle birlikte çıkan sonucun göre de sınıflandırması şeklindedir. Karar ağaçlarındaki amaç, sınıflandırmaya giden en kısa yolu bulmaktır (Silahtaroğlu 2020).

Karar ağaçları, üç bölümden oluşmaktadır; kök, düğümler ve yapraklar (bkz. Şekil 3.3). Burada $\{A_1, A_2, \ldots, A_n\}$'ler düğümleri ve birer öğeyi, $\{C_1, C_2, \ldots, Cn\}$'ler ise birer yapraktır ve sınıfları temsil etmektedir. Her A_i düğümü hakkında cevabı veri tabanında bulunacak bir sorunun ardından, verilen yanıta göre düğüm iki dala ayrılmaktadır.



Şekil 3.3: Karar Ağaçları.

Karar ağaçları gözetimli öğrenme olduğu için veri tabanında daha önceden belirlenmiş sınıflar mevcuttur. Algoritmanın modellenmesine göre ağacın şekli değişebilmektedir. Kök düğümünün değişmesi, yaprağa ulaşırken izlenilen yolu değiştirecektir.

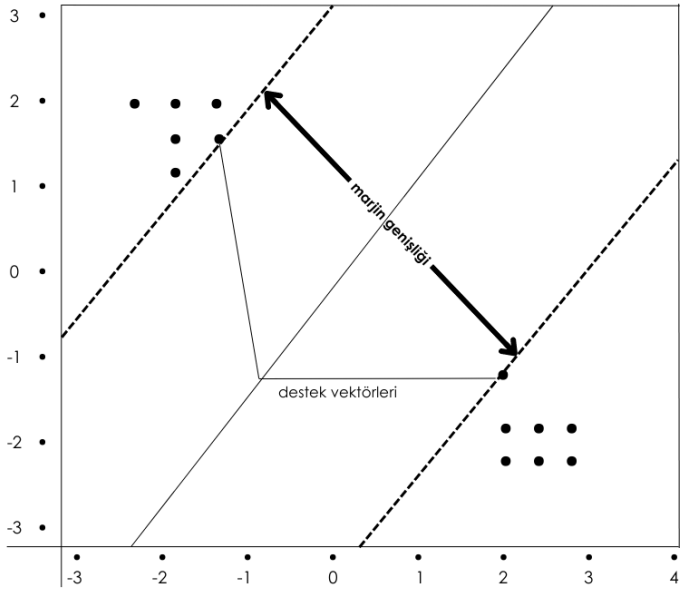
Hangi değişkenin kök düğümü olacağı belirlendikten sonra veri tabanındaki her bir bağımsız değişken birer düğümü temsil eder. Her bir düğümden sonra dallanma oluşturularak diğer düğümlere geçiş sağlanır. Dallara ayırma işlemini yapacak kriterin belirlemesi de önemlidir. Bu kriterler kadın/erkek, evet/hayır gibi ayrıldığı düğümü ikiye ayıracak bir cevap olabilirken; kilo, boy vb. gibi çok fazla değişen bir durum da olabilir. Bu durumda her bir veri için dallandırma yapmaktansa bu veriler gruplandırılabilir; 150<boy<170, 171<boy<170 gibi. Düğümlerin sayısı artış gösterdikçe modelin karmaşıklık seviyesi de artış göstermektedir.

3.3.1.1 Destek Vektör Makinesi

Gözetimli öğrenme yöntemlerinden biri olan destek vektör makineleri (Support Vector Machine) sınıflandırma ve regresyon problemlerinin çözümünde kullanılır. Sınıflandırma problemini çözerken amaç bir düzlem üzerine yerleştirilmiş verileri

ayıran en doğru sınırı çizmektir. Bu sınır, iki sınıfın noktaları için de maksimum uzaklıkta olmalıdır.

Sınıflandırmayı yapabilmek için iki boyutlu düzlemde çizilen doğrunun -1 ve +1 arasında kalan bölgeye Magrin adı verilir. Margin ne kadar geniş olursa ayrıştırılmak istenen iki veya daha fazla sınıf o kadar iyi ayrıştırılır (Rasmussen 2022).



Şekil 3.4: Destek Vektör Makinesi ile sınıflandırma.

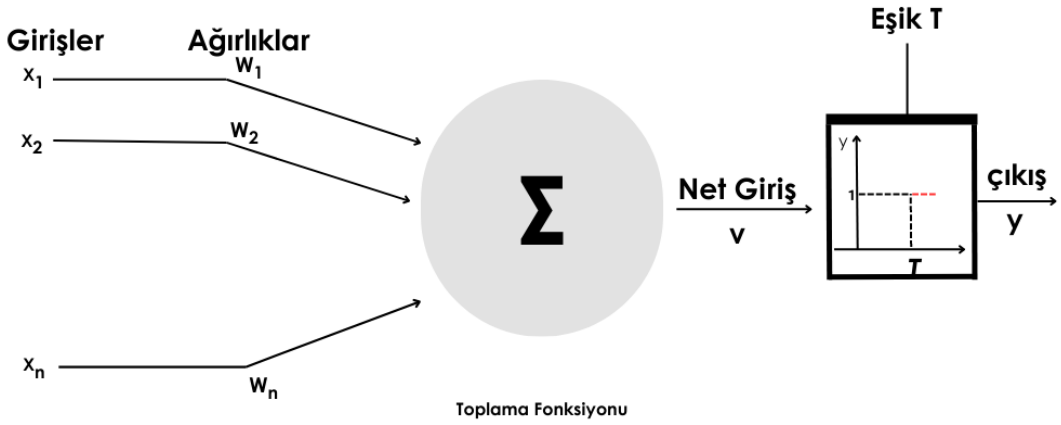
Bazı örnekler Margin bölgesine girebilir. Bu duruma Soft Margin denir. Hard Margin ise verinin doğrusal ayrılabildiği durumlarda çalışır haldedir. Aykırı durum oluştuğunda da çok duyarlıdır. Buna dayanarak bazı durumlarda Soft Margin tercih edilmelidir.

Destek vektör makineleri hem doğrusal hem de doğrusal olmayan sınıflandırma ve regresyon problemlerini çözmek için kullanılabilir. Doğrusal bir destek vektör makineleri, veri kümesindeki örneklerin doğrusal olarak ayrılabilir olduğu durumlarda kullanılırken, doğrusal olmayan destek vektör makineleri, verilerin doğrusal olarak ayrılamadığı durumlarda kullanılır. Doğrusal olmayan destek vektör makineleri, çeşitli çekirdek işlevleri (kernel functions) kullanarak verileri daha yüksek boyutlu uzaylara yansıtarak çalışır.

Destek vektör makineleri, özellikle küçük veri kümelerinde etkili bir şekilde çalışır. Ancak büyük veri kümelerinde hesaplama gücü açısından bazı zorluklarla karşılaşabilir.

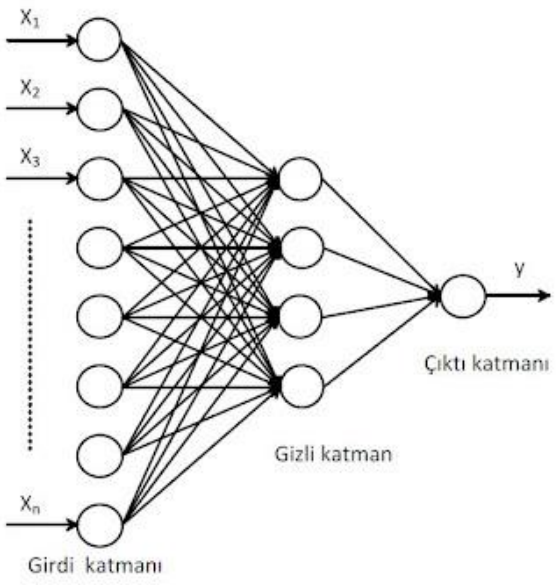
3.3.1.2 Yapay Sinir Ağları

Yapay Sinir Ağları (YSA), insan beyninde bulunan birbirine bağlı yüz milyar nöronun işlevselliğinden yola çıkılarak matematiksel olarak geliştirilmiş bir modeldir (bkz. Şekil 3.5). Yapay Sinir Ağları; örüntü tanıma, tahminleme, optimizasyon, kümele ve sınıflandırma gibi birçok problemi çözmektedir (Jain ve Mao 1996). Yapay Sinir Ağları Şekil 3.6'da görüldüğü üzere nöronların oluşturduğu girdi katmanından, bir veya birden fazla gizli katmandan ve son olarak çıktı katmanından oluşmaktadır.



Şekil 3.5: Bir nöronun matematiksel modeli.

Yapay sinir ağlarındaki veri, ağ içerisindeki bağlantılar ve bu bağlantıların ağırlıkları yoluyla iletilmektedir. Giriş katmanı bilgilerin iletildiği katmandır. Bu katmandan iletilen bilgiler ara katmanda incelenir, işlenir ve buradan çıkış katmanına iletilir. Bilginin işlenmesi, ağa iletilen bilgilerin ağ değerleri yardımıyla çıktıya dönüştürülmesidir.



Şekil 3.6: Yapay Sinir Ağları katman yapısı.

Model kurulduktan sonra sinir ağı eğitilir, modele girilen giriş verilerinin hangi çıkış değerlerini verdiği veri tabanından karşılaştırılmaktadır. Bu karşılaştırmadan nöron fonksiyonlarının yapılan hata miktarı ayarlaması sağlanmaktadır. Veri tabanındaki veriler bu şekilde modele verilerek yapay sinir ağının veri yapısını öğrenmesi sağlanır.

Yapay sinir ağları, veri madenciliği alanında oldukça popüler ve etkili bir tekniktir. Bununla birlikte, yapay sinir ağlarının hem avantajları hem de dezavantajları bulunmaktadır.

Avantajları:

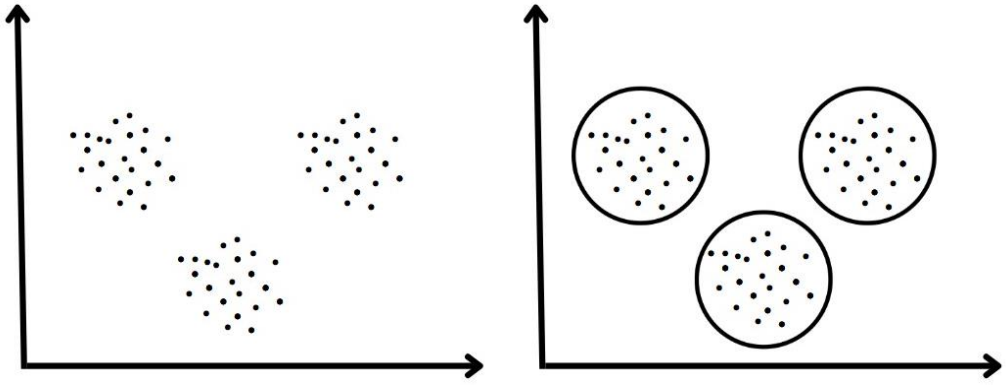
- Yapay sinir ağları, doğrusal olmayan ilişkileri tanıyabilir ve daha genel bir şekilde veri setlerini modellere uyarlayabilir.
- Gürültülü veya eksik verilerle çalışabilir.
- Verilerin özelliklerine ve özellikler arasındaki ilişkilere dayanarak, yeni veriler üzerinde doğru tahminler yapabilir.
- Yapay sinir ağları, öğrenme sürecindeki hataları düzeltmek için geri bildirim döngüsü kullanırlar. Bu sayede öğrenme süreci istenilen doğrulukta sonuçlar verebilir.

Dezavantajları:

- Yapay sinir ağları, modelin karmaşıklığından dolayı yüksek hesaplama gücü gerektirir.
- Modelin karmaşıklığı nedeniyle, aşırı uyum (overfitting) riski taşımaktadır. Bu durumda, model eğitim verilerine çok fazla uyum sağlayarak yeni veriler için yanlış tahminler yapabilir.
- Yapay sinir ağları, eğitim verilerinin doğru sınıflandırılması için yeterince büyük bir veri setine ihtiyaç duymaktadır.

3.3.2 Kümeleme Analizleri

Kümeleme (clustering) algoritmaları veri tabanında bulunan elemanların birbirleri arasındaki benzerliğe göre gruplaştırma yapmaktadır. Sınıflandırma işlemlerinde sınıflar önceden bilinirken kümeleme işlemlerinde sınıflar önceden belirli değildir. Bu nedenle, kümeleme algoritmaları gözetimsiz veri madenciliği yöntemidir. Veri tabanında bulunan verilen hangi kümelere ayrılacağı veri yapıları arasındaki benzerliğe göre belirlenmektedir (Bkz. Şekil 3.7). Algoritmadan alınacak sonucun daha doğru olabilmesi veya algoritmanın işlem yapma süresini kısaltmak için, kullanılacak olan algoritmaya istenilen küme sayısı, her bir kümede bulunması istenilen eleman veya veriler arasındaki minimum-maksimum benzerlik ve uzaklık bilgisi girilebilir.



Şekil 3.7: Kümeleme algoritmasının çıktısı.

Kullanılan verinin türüne, araştırmanın amacına göre veya kümeleme işleminin oluşturulma şekline göre kümeleme analizinde kullanılan algoritmalar farklılık göstermektedir (Han ve Kamber 2012). Başlıca kümeleme yöntemleri şu şekilde sınıflandırılabilir;

- Hiyerarşik Yöntemler;
- Bölümlemeli (Patitioning) Yöntemler;
- Izgara Temelli Yöntemler;
- Kısıtlara Dayanan Yöntemler;
- Kategorik Verinin Yinelenmesine Dayanan Yöntemlerdir.

Kümeleme algoritmaları, veri setindeki benzer verileri bir araya getirerek veri setindeki yapıları, benzerlikleri ve farklılıkları keşfetmeye yardımcı olurlar. Kümeleme analizleri, özellikle müşteri profili analizlerinde, pazarlama alanında, sosyal medya analizi gibi alanlarda kullanıcı analizi ve davranışsal analizler için sıklıkla kullanılmaktadır.

Tez çalışmasında VM regresyon ve sınıflandırma metotları YSA metotları irdelenmiş, ancak bu çalışmada veri yapısına uygun olarak İlişkisel Kural madenciliği (ARM) seçilmiştir. ARM'nin seçilme nedenlerin en başında, endokrin hasatların görülen 4 tip hastalıklar arasındaki gizli ilişkinin bu metotla en iyi şekilde ortaya çıkarılabileceği tespit edildiğinden bu yöntem seçilmiştir.

3.3.3 Birliktelik Kuralları ve İlişki Analizleri

Birliktelik kuralları, veri madenciliği alanında birçok farklı uygulama alanında kullanılan, veriler arasındaki ilişkiyi tanımlayan ve olayların birlikte gerçekleşme olasılığını arayan bir tekniktir. Veri tabanı içerisindeki gizli kalmış örüntüleri ortaya çıkarmayı hedeflemektedir.

Birliktelik kuralları, ürün tavsiyeleri, pazarlama stratejileri ve stok yönetimi gibi alanlarda kullanılır. Bu analiz türünde, birlikte satın alınan ürünlerin kombinasyonlarına dayalı olarak olası kurallar tanımlanır. Yaygın olarak bilinen Market Sepeti Analizinden bir örnekle herhangi bir ürünün yanında başka bir ürünün

birlikte alınma olasılığı birliktelik kurallarını vermektedir. Perakendecilik sektöründe sıklıkla kullanılan birliktelik kuralları algoritmaları, bir müşterinin profilini belirmede yardımcı olmaktadır. Müşteri profilini bilen perakendeci mağazasındaki ürünlerin dizilimleri veya müşterinin bir sonraki olası alışverişi için fikir sahibi olabilmektedir.

Perakendecilik sektörünün yanı sıra sağlık, eğitim, otomotiv, reklamcılık, finans, bankacılık ve sosyal medya sektörlerinde de sıklıkla kullanılmaktadır. Genetik olarak taşınan hastalıkların tespiti, öğrencilerin sınav sonuçları, müşterilerin memnuniyet analizi, dolandırıcılık tespiti gibi örnekler verilebilir. Ancak, büyük veri kümelerinde ve çok sayıda öğe ile çalışırken, birliktelik kuralları analizi hesaplama açısından zorlayıcı olabilir ve sonuçların yorumlanması zorlaşabilir.

Veri analizi sistemleri, büyük veri tabanları içerisinde gizli kalmış, etkili ve kullanılabilir bilgileri ortaya çıkarması sayesinde son zamanlar çok tercih edilen bir yöntem haline gelmiştir. Sık öğe seti madenciliği (Frequent Itemset Mining – FIM) birçok farklı veri analizi ve veri madenciliği görevlerinde, meydana sık gelen olayları, desenleri veya verileri ortaya çıkarmaktadır (Aggarwal ve Han 2014).

Veri madenciliği alanında birliktelik kuralları ve ilişki analizleri sıklıkla kullanılan tekniklerdir. Bu teknikler sayesinde, bir veri kümesindeki öğeler arasındaki ilişkiler belirlenerek örüntüler tespit edilebilir ve bu örüntülerin analizi çeşitli uygulamalarda kullanılabilir. Veri madenciliği yöntemleri arasında sıklıkla kullanılan birliktelik kurallarının temelleri ilk kez Agrawal ve diğ. (1993)'nin yaptığı çalışmada matematiksel olarak gerçekleştirilmiştir. Bu çalışmada, yatay veri tabanı ve genişlik öncelikli arama (Breadth-First Search – BFS) metodu kullanılarak, veri setini birçok kez tarayarak aday öğe setlerini bulmaktadır. Agrawal ve diğ. (1993) tarafından geliştirilen Agrawal Imielinski Swami (AIS) algoritmasının ardından, yaygın olarak kullanılan Apriori algoritması 1994 yılında Agrawal ve Srikant tarafından gerçekleştirilmiştir. Apriori algoritması, önceki aday öğe setlerini kullanarak yeni aday öğe setleri oluşturur ve minimum destek kriterine göre öğe setlerini filtreler. Daha sonra geliştirilen ECLAT (Zaki 2000) ve FP-Growth (Han ve diğ. 2000) algoritmaları dikey veri tabanı ve derinlik öncelikli arama (Depth-First Search – DFS) metodunu kullanmaktadır. Bu algoritmalar, öğe setlerinin sıklığına göre gruplanarak birliktelik kurallarının tespit edilmesine olanak tanır.

Birliktelik kuralları ve ilişki analizleri, veri madenciliği alanında oldukça kullanışlı ve etkili tekniklerdir. Ancak, büyük veri kümelerindeki analizleri gerçekleştirmek için yüksek hesaplama gücüne ve bellek kapasitesine ihtiyaç duyulması dezavantajları arasındadır. Ayrıca, veriler arasında bulunan bazı ilişkilerin anlamsız veya yanıltıcı olabileceği de bir dezavantaj olarak gösterilebilmektedir.

3.3.3.1 Sık Öğe Seti Madenciliği

Sık öğe seti madenciliği, birliktelik kuralları analizinin temelidir. Bu yöntem, bir veri kümesindeki sık tekrarlanan öğe kümelerini (sık öğe setleri) bulmak için kullanılır. Sık öğe setleri, belirli bir destek değerini karşılayan öğelerin birleşimleridir. Destek, bir öğe kümesinin veri kümesinde kaç kez göründüğüne bağlı olarak belirlenir.

İlk FIM algoritmasının ortaya çıktığı 1993 yılından itibaren, her geçen gün bu alandaki algoritmaların performanslarını geliştirmek için çalışılmaktadır. Artık, aşırı büyük veri tabanları bile birkaç saniye içinde analiz edilmektedir. Bu sadece ilerleyen teknolojiyle gelişen donanımlar sayesinde değil, aynı zamanda önerilen algoritmaların sonucudur.

FIM ilk etapta müşteri verilerini analiz etmek için önerilmiş olsa da artık birçok alan için de kullanılmaktadır. Aslında bir veri tabanı genel olarak, öğelerden (items) ve bu öğeleri tanımlayan işlemlerden (transactions) oluşmaktadır. Böylece FIM, bir veri tabanında sık sık birlikte ortaya çıkan öğeleri bulmakla görevlidir denilebilir. FIM algoritmalarının kullanım alanları her geçen gün genişlemektedir. Bu alanlar; biyoinformatik, görüntü kümeleme, trafik ağları analizi, müşteri yorumları analizi, görüntüleme, kötü amaçlı yazılım tespiti gibi alanlardır (Fournier-Viger ve diğ. 2017). Perakende sektöründe müşteri davranışlarının analizi için kullanılabilir. Bu analiz, müşterilerin hangi ürünleri birlikte satın aldıklarını belirleyebilir ve mağaza yöneticilerine hangi ürünlerin bir arada sergilenmesi veya birbirine bağlanması gerektiği konusunda fikir verebilir. Ayrıca, sağlık sektöründe hastalık teşhisinde, tıbbi tedavilerde, ilaç keşfinde ve genomik veri analizinde de kullanılabilir. Sık öğe seti madenciliği ayrıca web sayfaları, sosyal ağlar ve haberler gibi internet içeriği analizinde de kullanılabilir.

FIM algoritmaları aynı zamanla bazı özel durumların tespiti gibi alanlarda da kullanılmaktadır. Bu özel durumlarda FIM algoritmaları, nadir oluşan örüntüleri, ilişkili örüntüleri veya ardışık örüntüleri ortaya çıkarabilmektedir (Fournier-Viger ve diğ. 2017).

Her bir öğe *i* olmak üzere, öğe seti I = $\{i_1, i_2, \ldots, i_m\}$; İşlem veri tabanındaki her bir eşsiz işlem T_q olmak üzere, işlem veri tabanı D = $\{T_1, T_2, \ldots, T_n\}$ olsun. Her T_q işleminin TID (Transaction IDentifier) adlı benzersiz bir tanımlayıcısı vardır (Agrawal ve diğ. 1994).

Tablo 3.1'i inceleyecek olursak veri tabanı beş öğeden(a,b,c,d,e) ve beş işlemden oluşmaktadır. Örnek olarak ilk işlem olan T_1, bir müşterinin a, b ve c ürünlerini satın aldığı bir market fişini temsil etsin.

Tablo 3.1: Yatay veri tabanı (D).

TID	**İşlem**
T_1	{a, b, c}
T_2	{b, c}
T_3	{a, b, c, d, e}
T_4	{b, c, e}
T_5	{a, b, e}

Destek, birliktelik kurallarının belirlenmesinde kullanılan bir ölçüttür ve bir öğe kümesinin veri kümesinde kaç kez göründüğünü belirler. Destek değeri, birliktelik kurallarının belirlenmesinde kullanılan en önemli faktörlerden biridir. Destek değeri, bir öğe grubunun sıklığını ölçmek için kullanıldığından, daha yüksek destek değerleri, öğe grubunun daha sık görüldüğünü ve daha anlamlı olduğunu gösterir.

Bir öğe grubunun destek değeri, veri kümesindeki öğe grubunun toplam sayısının yüzdesi olarak aşağıdaki Denklem (3.4) ile ifade edilir:

$$Sup(X) = \frac{\text{X'inbulunduğu işlem sayısı}}{Toplam\ \text{İşlem}} \tag{3.4}$$

X öğe seti, I'nın bir alt kümesidir. X öğe setinin desteği $Sup(X)$ olarak tanımlanmaktadır. $Sup(X)$, X öğe setinin aslında D veri tabanındaki frekansını temsil

etmektedir. Buna mutlak destek adı verilmektedir. Bağıl destek ($relSup(X)$) ise mutlak desteğin, veri tabanı uzunluğuna bölünmesi ile elde edilmektedir (Denklem (3.5);

$$relSup(X) = \sup(X) / |D| \quad (3.5)$$

Örneğin, Tablo 3.2'de {a, b} öğe setinin sup(X) değeri üçtür; T_1, T_2 ve T_3 işlemlerinde ortaya çıkmaktadır. Ancak relSup(X) değeri 0,6'dır. Sık öğe seti madenciliği veri tabanındaki tüm sık öğe setlerini bulmaktadır. Bir öğe setine, sık öğe seti denilebilmesi için öğe setinin support değeri kullanıcı tarafından belirlenen eşik değerini(minSup) geçmelidir. Yani $\sup(X) \geq minsup$ olmalıdır. Kullanıcı tarafından minSup = 2 değeri verilsin o halde Tablo 3.2 sık öğe setleri oluşmuştur.

Tablo 3.2: Sık öğe setleri.

{a} relSup: 0,6	{a, c} relSup: 0,4	{a, b, c} relSup: 0,4
{b} relSup: 1	{a, e} relSup: 0,4	{a, b, e} relSup: 0,4
{c} relSup: 0,8	{b, c} relSup: 0,8	{b, c, e} relSup: 0,4
{e} relSup: 0,6	{b, e} relSup: 0,6	
{a, b} relSup: 0,6	{c, e} relSup: 0,4	

FIM, karmaşık bir numaralandırma problemidir. Amaç, kullanıcı tarafından girilen minimum eşik değerini karşılayan tüm kalıpları numaralandırmak olduğu için FIM'in her zaman bir tek doğru cevabı bulunmaktadır. Bu problemi çözmek için, en temel çözüm, veri tabanında oluşabilecek tüm öğe setleri bulunarak, bu öğe setlerinin destek değerlerinin eşik değeri kısıtlamasını karşılayıp karşılamadığını incelemektir (Fournier-Viger ve diğ. 2017). Bu yöntem her zaman en etkili yöntem olmayabilir. Eğer bir veri tabanında m adet öğe olması durumunda 2^m-1 adet muhtemel veri oluşacaktır. Böylece, arama uzayı analiz yapılacak veri tabanından daha büyük hale gelecektir.

Sık öğe setlerini verimli bir şekilde bulabilmek için birçok FIM algoritması geliştirilmiştir. Apriori, FP-Growth, ECLAT, H-Mine ve LCM (Least Common Multiple) bunlara örnek olarak verilebilir (Bkz. Tablo 3.3). Tüm bu algoritmaların giriş ve çıkış bilgileri aynıdır. Ancak, veri yapısı ve algoritmanın çözüm stratejileri bakımından farklılık göstermektedir. Bu farklılıkla, algoritmanın BFS veya DFS

olarak arama yapması, dikey veya yatay veri tabanı kullanması, öğe setlerini nasıl oluşturdukları ve ögelerin destek değerlerini nasıl buldukları örnek olarak verilebilir.

Tablo 3.3: Sık kullanılan FIM algoritmaları.

Algoritma	**Arama Türü**	**Veri Tabanı Türü**
Apriori	BFS (Aday Üretimi)	Yatay
ECLAT	DFS (Aday Üretimi)	Dikey (TID Listeleri)
FP-Growth	BFS(*Pattern-Growth*)	Yatay (Prefix-tree)
H-Mine	BFS(*Pattern-Growth*)	Yatay (Köprü Yapısı)
LCM	BFS(*Pattern-Growth*)	Yatay (İşlem Birleştirme)

Apriori algoritması, ilk FIM algoritmalarından biridir ve genellikle ilk tercih edilen algoritmalardan biridir. Algoritma, sık öğe setlerinin keşfi için bir dizi aday öğe seti oluşturur ve sonrasında bu aday öğe setlerini tarama işlemi yaparak sık öğe setlerini bulur.

ECLAT algoritması, veri kümesindeki öğelerin ortak özelliklerine dayanan sık öğe kümelerini bulmak için derinlik öncelikli arama (DFS) yöntemini kullanır.

FP-Growth algoritması, Apriori algoritmasından daha hızlı çalışır ve daha az bellek kullanır. Bu algoritma, veri kümelerindeki sık öğe setlerini bulmak için bir ağaç yapısı oluşturur ve bu ağaç yapısı üzerinde gezinerek sık öğe setlerini keşfeder.

H-Mine ve LCM algoritmaları da sık öğe seti madenciliği için kullanılan diğer algoritmalardır. H-Mine algoritması, yatay olarak tutulan veri kümelerinde çalışır ve çok boyutlu ölçeklendirme için tasarlanmıştır. LCM algoritması ise dikey olarak tutulan veri kümeleri için özel olarak tasarlanmıştır ve sık öğe setlerini keşfetmek için bir derinlik öncelikli arama yöntemi kullanır.

Bu FIM algoritmaları, sık öğe setleri madenciliği alanında oldukça yaygın bir şekilde kullanılmaktadır ve büyük veri kümelerinde verimli bir şekilde çalışabilmektedirler.

3.3.3.2 Apriori Algoritması

Apriori Algoritması, birliktelik kuralları içerisinde en yaygın kullanılan algoritmalardan birisidir. Algoritma girdisi olarak yatay veri tabanı ve minSup değerini kullanmaktadır.

Tablo 3.1'de görülen tablo yatay veri tabanına örnektir.

Algoritma öncelikle tüm öğelerin support değerini bulmak için veri tabanına arama yapar ve arama sonucunda tek elemanlı 1-VeriSeti (1 elemanlı veri seti) bilgisini vermektedir. 1-VeriSeti içerisindeki sık öğeleri bulur ve bu sık öğe setlerinden F_1'i oluşturur. Daha sona Apriori algoritması BFS yöntemiyle daha büyük sık öğe setlerini bulur. Algoritma aramayı yaparken k-1'inci sık öğe setini kullanır. Çünkü, k-1'inci öğe setinde (F_{k-1}), bir öğe sık değilse bir sonraki öğe setinde zaten sık olamaz mantığı ile çalışmaktadır. F_{k-1}'de bulunan öğeleri birbirleri ile k uzunlukta kombinleyerek aday öğe seti bulunur (C_k). Algoritma veri tabanını tekrar tarayarak C_k'da bulunan öğelerin support değerlerini hesaplayarak hangi öğe setlerinin sık öğe seti olduğu bulur. Eşik değerine göre bulunan sık öğe seti F_k'ya eklenir. Hiçbir aday kalmayana kadar algoritma devam eder.

Tablo 3.2'de, F_1 = {{a}, {b}, {c}, {e}} olduğu görülmektedir. Apriori algoritması F_1'deki sık öğe setlerini kullanarak 2 uzunluklu veri setini bulacaktır. Yani, C_2 = {{a, b}, {a, c}, {a, e}, {b, c}, {b, e}, {c, e}} olacaktır. C_k'daki öğeler taranarak support değeri bulunur ve eşik değerini geçen öğeler 2 uzunluklu sık öğe seti kümesi olan F_2'ye ekler. O halde F_2 = {{a, b}, {a, c}, {a, e}, {b, c}, {b, e}, {c, e}} olur. Bu örnek için C_2'de oluşan tüm aday öğe setleri sık öğe setidir. Oluşturulacak aday seti kalmayana kadar bu işlem devam eder.

Apriori algoritması bu algoritmadan sonra gelecek olan birden fazla algoritmaya ilham kaynağı olmuştur. Fakat bu algoritmanın bazı durumları verimliliği düşürmektedir. Bu durumlardan birisi, veri tabanına bakmadan, bir önceki sık öğe seti kullanarak adayların oluşturulmasıdır. Bu durum büyük bir zaman kaybı yaratmaktadır. Bir diğer durum ise, algoritma aday öğe setlerinin support değerini hesaplamak için her seferinde veri tabanını tekrar tarar. Bu da işlem yükünü artırmaktadır.

3.3.3.3 ECLAT Algoritması

ECLAT (Equivalence CLAss Transformation) algoritması, Apriori algoritmasının aksine birçok öğe setini bellekte tutmaktan kaçınmak için DFS yöntemi ve dikey veri tabanı kullanılmaktadır.

Tablo 3.1'de verilen yatay veri tabanı dikey veri tabanı haline getirilerek Tablo 3.4'de verilmiştir. Dikey veri tabanı her öğenin bulunduğu işlem listesini göstermektedir. Her işlemin gösterildiği liste TID-Listesi olarak adlandırılır ve *tid(X)* olarak gösterilir. Dikey veri tabanı bir kez taranarak yatay veri tabanına çevrilebilir. Aynı şekilde yatay veri tabanı, dikey veri tabanına dönüştürmek mümkündür (Fournier-Viger ve diğ. 2017).

Tablo 3.4: Dikey veri tabanı(R).

X	***tid(X)***
{a}	{T_1, T_3, T_5}
{b}	{T_1, T_2, T_3, T_4, T_5}
{c}	{T_1, T_2, T_3, T_4,}
{d}	{T_3}
{e}	{T_3, T_4, T_5}

Dikey veri tabanı kullanımı aşağıda verilen iki özelliğe sahip oldukları için oldukça kullanışlıdırlar.

- Xve Ygibi iki öğe setinden oluşacak yeni öğe seti, Denklem (3.6)'da verilen eşitlik ile orijinal veri tabanı tekrar taranmadan bulunabilir; ve

$$tid(X \cup Y) = tid(X) \cap tid(Y) \quad (3.6)$$

- Bir öğenin support değerini bulmak için veri tabanının tümünün taranması gerekmez. Öğenin TID-Listesine bakmak yeterlidir (Denklem (3.7)).

$$\sup(X) = |tid(X)| \quad (3.7)$$

Örnek olarak {b} ve {c} öğe setlerinden oluşan {b, c} öğe setinin işlem listesi ve support değeri şu şekilde bulunur;

$$tid(\{\mathrm{b}\} \cup \{\mathrm{c}\}) = tid(\{\mathrm{b}\}) \cap tid(\{\mathrm{c}\}) = \{\mathrm{T1},\ \mathrm{T2}, \mathrm{T3}, \mathrm{T4},\} \text{ ve}$$

$$sup(\{b, c\}) = |tid(\{b, c\})| = 4$$

Bu nedenle, bu iki özelliği kullanarak, ECLAT gibi dikey algoritmalar, ilk TID listelerini oluşturmak için veri tabanını yalnızca bir kez tarayarak arama uzayını oluşturabilirler. Aday üretimi ve support sayımı, veri tabanını taramadan yapılmaktadır.

ECLAT algoritması dikey veri tabanını ve minSup değerini girdi olarak kullanmaktadır. Dikey veri tabanında bulunan her X öge seti için bir döngü oluşturulmaktadır. X öğe seti ilk çıktıdır. Ardından X öğe setine bir öğe eklenerek genişletilmektedir. Xöğe-setiyle, son öğe hariç tüm elemanları aynı olan her bir Y öğe-setini birleştirerek $X \cup Y$ öğe-seti elde edilir. $X \cup Y$'nin işlem listesi Denklem 3.5 ile bulunmaktadır. Eğer $X \cup Y$ sık öğe setiyse; X, genişletilmiş öğe seti olan E'ye eklenmektedir. Ardından, algoritma, $X \cup Y$'nin tüm uzantılarını bulmak için E kümesi ile kendi kendini çağırmaktadır. Bu döngü, R'nin içindeki tüm öğe-setleri için tekrarlanmaktadır. Algoritma sona erdiğinde elde edilen tüm sık öğe-setleri ve onların support değerleri çıktısı oluşmaktadır.

ECLAT, veri tabanını birçok kez tarama yapmadığı için genellikle Apriori algoritmasından çok daha hızlı çalışmaktadır. Ancak, ECLAT veri tabanını taramadan adaylar ürettiği için veri tabanında bulunmayan öğe kümelerini üreterek zaman harcayabilmektedir. Ayrıca, TID listeleri yararlı olmasına rağmen, özellikle yoğun veri kümeleri için çok fazla bellek tüketmektedir. ECLAT, Apriori algoritması gibi büyük veri setleri üzerinde çalışırken yüksek bellek gereksinimine sahiptir. Veri kümesi büyüdükçe, algoritmanın bellek kullanımı artar ve performansı düşmektedir.

3.3.3.1 FP-Growth Algoritması

Apriori ve ECLAT gibi algoritmaların ana problemleri, FP-Growth algoritmalarının gelişmesinde önemli rol oynamışlardır. FP-Growth algoritması, Apriori ve ECLAT algoritmalarına göre daha verimli bir şekilde çalışır. Bu nedenle büyük veri kümeleri üzerinde çalışırken avantaj sağlar. FP-Growth algoritmalarındaki temel fikir, öğe setlerini bulmak için veri tabanını taramak ve veri tabanında olmayan aday öğe seti üretmekten kaçınmaktır. Büyük veri tabanlarında verimli bir şekilde çalışmaktadır. Veri tabanını tarama maliyetlerini azaltmaya yönelik mevcut veri tabanını daha küçük veri tabalarına bölerek çıkış verisine ulaşmaktadır (Erpolat, S., 2012).

FP-Growth algoritması, veri setindeki sık öğe setlerini bulmak için ağaç yapısını kullanarak çalışmaktadır. Bu ağaç yapısı, algoritmanın hızlı çalışmasındaki en önemli etkendir. Bu ağaç FP-Tree olarak adlandırılır. FP-Tree yapısı, veri kümesindeki sık öğelerin birbirleriyle olan ilişkisini ve frekanslarını tutar. Ağaç yapısı sayesinde, sık öğelerin bir arada bulunma sıklığına göre öncelik sıralaması yapılarak, daha küçük bir veri kümesi oluşturulur. Bu şekilde, daha verimli bir şekilde sık öğelerin bulunması sağlanır.

FP ağacının her düğümü, öğe kümesinin bir öğesini temsil eder. FP-Growth algoritması için veri tabanı iki kez taranır. Bu algoritma, girdi olarak yatay veri tabanı(D), boş bir öğe seti ve *minSup* değerini kullanmaktadır.

İlk taramada; tüm veri tabanı taranır, her bir eşsiz öğe bulunur ve Support değerleri hesaplanır. Veri tabanındaki 1-öğeli öğe seti oluşturulur. minSup değerinin altında kalan öğeler çıkarılır ve 1-öğeli sık öğe seti bulunur. Her bir sık öğe, en büyük Support değerlerinden, en küçük Support değerine doğru sıralanır ve ağaç yapısı oluşturulmaya başlanır. Ağaç yapısında, kök (root) boş küme olarak oluşturulur.

Veri tabanı ikinci kez taranır, her öğe için; sadece sık kullanılan öğeler ağaca eklenir. Maksimum Support değerine sahip sık öğe en üstte, Support değeri en düşük olan en altta olacak şekilde azalan sayım sırasına göre ağacın dalları oluşturulur. Bir işlemin öğeleri, kendisinden önce gelen işlemlerde mevcutsa bu işlemin öğesi kök ile ortak bir önek (Prefix) paylaşmaktadır. Ayrıca, işlemler ağaca yerleştikçe, öğe

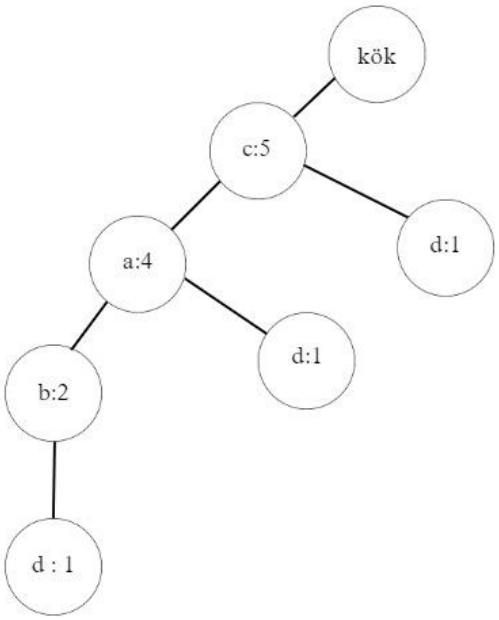
kümesinin sayısı artırılır. Hem ortak düğüm hem de yeni düğüm sayısı, işlemlere göre oluşturuldukça sayı(count) 1 artmaktadır.

Tablo 3.5'te örnek alışveriş listesi verilmiştir. Veri tabanı taranarak öğe setleri bulunur ve Support değerinin altında kalanlar listeden çıkarılmaktadır. En büyük support değerinden başlayarak öğeler sıralanmakta ve veri tabanı düzenlenmektedir.

Tablo 3.5: Örnek alışveriş listesi.

TID	**X**	***Sıralanmış X***
T1	{a, b, c}	{c, a, b}
T2	{a, b, c, d}	{c, a, b, d}
T3	{c, d, e}	{c, d}
T4	{a, c, d}	{c, a, d}
T5	{a, b, c, e}	{c, a, b}

- {a}=4, {b}=3, {c}=5, {d}=3, {e}=2 öğeler ve Support değerleridir. MinSup = 3 olsun. Bu durumda {e} öğesi MinSup değerinin altında kaldığı için sık öğe seti listesinden çıkarılır.
- Öğeler en büyük Support değerinden başlanarak sıralanır; {c}=5, {a}=4, {b}=3, {d}=3.
- Ağaç yapısı oluşturulmaya başlanır. Kök, boş küme olarak atanır. T1, 3 öğeden oluşmaktadır ve en büyük Support değerine sahip olan {c} öğesi köke bağlanır. Sırasıyla {c} öğesinin altına {a} ve {b} öğleleri bağlanır ve sayıları 1 artırılır.
- T2, 4 öğeden oluşmaktadır. {c}, {a} ve {b} öğeleri sırasıyla bir önceki işlemde ağaçta oluşturulduğu için yeniden düğüm oluşturulmaz, mevcut düğümlerin sayısı 1 artırılır. {d} öğesi, {b} öğesine bağlanır.
- Tüm TIDler, ağaca yerleştirilene kadar bu işlem devam eder. En sonda oluşan ağaç, Şekil 3.8'de verilmiştir.



Şekil 3.8: FP-Tree yapısı.

FP-Growth algoritmasının avantajları şunlardır:

- Apriori ve ECLAT algoritmasına kıyasla daha hızlıdır ve daha az bellek gerektirir.
- Sık öğe setleri için bir ağaç yapısı oluşturarak işlem yapar, bu da veri kümesinin boyutunda bağımsız olarak daha iyi performans sağlar.
- Algoritmanın yapısı, diğer veri madenciliği yöntemleriyle (örneğin, karar ağaçları) birleştirilebilir.

FP-Growth algoritmasının dezavantajları şunlardır:

- Apriori algoritmasında olduğu gibi, minimum destek değerinin önceden belirlenmesi gerekmektedir.
- Veri setindeki sık öğe setleri büyükse, ağaç yapısı oluşturma işlemi bellek kullanımını artırabilir.
- Yüksek boyutlu veri kümelerinde performansı düşük olabilir

3.3.3.2 H-Mine Algoritması

H-Mine, veri madenciliği alanında kullanılan bir algoritmadır ve özellikle sık öğe seti analizi işlemlerinde kullanılır. H-Mine algoritması, veri tabanında bulunan sık öğe kümelerinin tespit edilmesinde kullanılır. Sık öğe kümeleri, veri tabanında sıkça görülen öğelerin bir araya gelmesiyle oluşan öğelerdir. Bu yöntemin belirgin bir özelliği, bellek tabanlı ortamda oldukça hızlı çalışmasıdır.

Bu algoritma, "hem her durumda verimli çalışan hem de büyük veri tabanları için gereken belleği küçültme" probleminden yola çıkılmıştır. H-Mine algoritması çok büyük veri tabanlarını bölerek analiz yapmaktadır. Eğer veri seti çok yoğunsa, madencilik analizlerinde devam etmek için FP-Tree algoritmasına geçiş yapabilir.

Algoritma girdisi olarak yatay veri tabanı ve minSup değerleri kullanılır. H-mine algoritması aday öğe seti oluşturmaz ve FP-Growth algoritmasına benzer sık öğeleri bulur. Ayrıca H-Mine herhangi bir sık öğe setini hafızada tutmaya ihtiyaç duymamaktadır. H-mine algoritması arama uzayını sınırlandırır. Diğer FP-Growth algoritmalarının yöntemlerinden farklı olarak bellek yapılarını fiziksel olarak oluşturması gerekmez. İyi organize edilmiş H-Struct yapısında bilgileri tam kullanır ve header tablolarında bu bilgileri toplamaktadır. Bu yöntem alan yönetimi konusunda oldukça tasarruf sağlamaktadır.

H-struct yapısı, H-mine algoritmasının temel veri yapısıdır ve birbirleriyle bağlantılı olan düğümlerden oluşur. H-struct yapısı, FP-tree yapısından farklı olarak, birçok hiper bağlantı (hyper-link) içerir. Hyper-link, bir düğümün diğer düğümlere olan bağlantılarının bir türüdür. H-struct yapısında, bir düğümün doğrudan bağlandığı diğer düğümlerle birlikte, o düğümün diğer düğümlere olan dolaylı bağlantıları da mevcuttur. Hiper bağlantılar, iki düğüm arasındaki mesafeyi azaltabilir ve dolayısıyla madencilik işlemini hızlandırabilir. Bu nedenle, H-mine algoritması, H-struct yapısını kullanarak daha hızlı ve etkili bir şekilde sıklık kuralı madenciliği yapabilir.

3.3.3.3 İlişkisel Kural Madenciliği

İlişkisel Kural Madenciliği (Association Rule Mining – ARM), FIM algoritmaları ile bulunan sık öğe setleri arasındaki ilişkiyi, ilginç korelasyonları veya örüntüleri bulmak için kullanılmaktadır. Bu yöntem telekomünikasyon ağları, market analizi, stok kontrolü vb. gibi çeşitli alanda yaygın olarak kullanılmaktadır (Zhoa ve Bhowmick 2003). İlişkisel Kural Madenciliği (Association Rule Mining), pazarlama, perakende, web madenciliği, tıbbi veri analizi ve finansal analiz gibi birçok alanda kullanılmaktadır. Özellikle perakende sektöründe müşteri satın alma alışkanlıklarını ve ürünler arasındaki ilişkileri anlamak için yaygın olarak kullanılmaktadır. Ayrıca tıbbi veri analizinde de ilaç etkileşimleri veya hastalıkların birbirleriyle olan ilişkileri gibi önemli bilgilerin keşfedilmesi için kullanılmaktadır.

Herhangi iki öğe seti arasındaki muhtemel ilişki $X=>Y$ olarak ifade edilir ve "X öğe setinin bulunduğu bir işlemde Y öğe seti de bulunmaktadır" anlamına gelmektedir. $Sup(X \cup Y)$, X ve Y öğe setlerinin birlikte görüldüğü veri işlemlerinin yüzdesini ifade etmektedir. Burada X ve Y, support değerleri sıfırdan farklı ($sup(X) \neq 0$ ve $sup(Y) \neq 0$) ve kesişim kümesi boş küme olan iki sık öğe setidir. X, öncül (premise) olarak adlandırılırken, Y ise ardıl (consequent) olarak adlandırılmaktadır.

Örneğin, Sık Öğe Seti Madenciliği bölümünde bulunan sık öğe setleri için {b, c} => {e} gibi bir kural olsun. Bu kural "{b, c} öğe setinin bulunduğu bir işlemde {e} öğe seti de vardır" anlamına gelmektedir. Ancak, {b, c} öğe setinin bulunduğu bir işlemde {e} öğe setinin bulunma olasılığının ne olabileceği ve bu kuralın güvenilirliği akla gelebilecek problemlerdir. Bir kuralın güvenilirliğini ifade etmek olasılıksal bir kavram kullanılmaktadır: *güven (confidence)*.

$X=>Y$ şeklindeki bir kuralın güvenilirliği Denklem 3.8'de verilen eşitlik ile hesaplanmaktadır:

$$Conf(X => Y) = \frac{\sup(X \cup Y)}{\sup(X)} \tag{3.8}$$

ve değer aralığı Denklem (3.9)'da verildiği aralıkta

$$0 \leq Conf(X => Y) \leq 1 \quad (3.9)$$

olmalıdır. *Conf(X=>Y)* = %80 değerinin anlamı, *X* öğe setinin bulunduğu bir işlemde, *Y* öğe setinin bulunma olasılığı %80'dir" demektir. Bir öğe setinin, sık öğe seti olabilmesi için *sup(x)* değerinin minimum eşik değerine eşit olması gerektiği gibi *X=>Y* gibi bir ilişkinin *ilişkisel kural* olabilmesi için aynı anda şu iki koşulu sağlaması gerekmektedir:

1) $minSup \leq sup\ (X)$
2) $minConf \leq Conf(X => Y)$

Burada *minConf* değeri kullanıcı tarafından belirlenen, bir kuralın oluşabilmesi için sahip olması gereken en küçük güven değeridir. Birliktelik kuralı, iki parçadan oluşur: bir ön koşul (if) ve bir sonuç (then). Bir ön koşul, veri içinde bulunan bir öğedir. Bir sonuç, ön koşulla birlikte bulunan bir öğedir.

Birliktelik kuralları, sık if-then kalıplarını arayarak oluşturulur ve en önemli ilişkileri belirlemek için support ve confidence kriterlerini kullanır. Destek, öğelerin veride ne sıklıkla göründüğünün bir göstergesidir. Güven, if-then ifadelerinin kaç kez doğru bulunduğunu gösterir. Beklenen güven ile karşılaştırmak için kullanılan lift adlı üçüncü bir ölçüt de bulunmaktadır.

Lift, birliktelik kurallarının gücünü ve ilişki derecesini ölçen bir ölçüdür. Lift, birliktelik kuralının bağımsız olasılık tahminlerine göre ne kadar iyi performans gösterdiğini gösterir. Lift değeri 1'e yakınsa, ürünler birbirleriyle bağımsızdır, lift değeri 1'den büyükse, ürünler arasında pozitif bir ilişki vardır ve lift değeri 1'den küçükse, ürünler arasında negatif bir ilişki vardır.

Lift değeri ne kadar yüksekse, birliktelik kuralı o kadar güçlüdür. Ancak, lift yüksekliği birliktelik kuralının işlevselliği açısından tek başına yeterli değildir. Aynı zamanda destek ve güvenirlik değerleri de hesaba katılmalıdır. Matematiksel olarak lift değeri Denklem (3.10)'da verilen eşitlik ile ifade edilir:

$$lift = \frac{sup\ (X \cup Y)}{sup(X) * sup\ (Y)} \quad (3.10)$$

Interestingness kavramı veri madenciliğinde çıktılarının yararlılığını ölçmek için kullanılan bir terimdir. Veri madenciliği uygulamalarında, çok sayıda keşfedilen kuralların veya birlikteliklerin bulunması mümkündür, ancak bunların hepsi yararlı olmayabilir veya beklenmeyen, basit ve önemsiz sonuçlar olabilir. Bu nedenle, interestingness kavramı, keşfedilen kalıpların veya birlikteliklerin ne kadar ilginç ve yararlı olduklarını ölçmek için kullanılır. İlginçlik ölçütleri, keşfedilen kalıpların sıklığı, kapsamı, tutarlılığı ve diğer özellikleri gibi çeşitli faktörlere dayanabilir. Bu ölçütler, çıktıların daha anlamlı ve değerli hale getirilmesine yardımcı olur ve veri madenciliği uygulamalarının daha iyi sonuçlar vermesini sağlar. Matematiksel formülü çeşitli şekillerde ifade edilebilir, ancak genellikle denklem (3.11)'de verilen eşitlik şeklinde ifade edilir:

$$Interestingness = Sup(X \cup Y) * Conf(X => Y) * lift(X, Y) \quad (3.11)$$

Bu bölümde, veri madenciliği kavramı tanımlanmış ve bu teknolojinin kullanıldığı alanlara örnekler verilmiştir. Veri madenciliği algoritmaları arasında sık öğe seti madenciliği, Apriori, Eclat, FP-Tree, H-Mine ve ilişkisel kural madenciliği (ARM) örnekleri verilmiştir.

Sık öğe seti madenciliği, veri setinde sıkça bulunan öğe gruplarını bulmak için kullanılır. Apriori algoritması, veri setinde sıkça bulunan öğeleri bulmak için kullanılan bir diğer algoritmadır. Eclat algoritması, sık öğe seti madenciliği için geliştirilmiş bir diğer algoritmadır. FP-Tree algoritması, büyük veri kümelerinde etkili bir şekilde çalışabilen bir diğer sık öğe seti madenciliği algoritmasıdır. H-Mine algoritması ise çoklu destek kısıtını kullanarak sık öğe seti madenciliği yapar. Son olarak, ARM, ilişkisel kural madenciliği için kullanılan bir algoritmadır ve iki veya daha fazla öğe arasındaki ilişkileri bulmak için kullanılır. Bu algoritmaların her birinin kendine özgü avantajları ve dezavantajları vardır. Algoritmaların performansı, veri setinin boyutuna, tipine ve analiz amacına göre değişebilir.

4. ÖRNEK UYGULAMA

4.1 Kullanılan Donanımlar ve Yazılımlar

Bu tez çalışması için yüksek performanslı bir bilgisayar kullanılmıştır. Bilgisayar, 12. nesil Intel i7 işlemciye sahiptir ve 64 bit işletim sistemiyle çalışmaktadır. Ayrıca, 64 GB RAM'e sahip olan ASUS marka bir bilgisayardır. Bu özellikler, büyük veri setleri üzerinde yoğun hesaplamalar yapılabilmesine ve veri madenciliği algoritmalarının hızlı bir şekilde çalıştırılabilmesine olanak tanımaktadır. Bu sayede, analizlerin daha hızlı ve etkili bir şekilde yapılması mümkün olmuştur.

Veri madenciliği analizleri yapmak için birçok program ve yazılım dili bulunmaktadır. Bu çalışmada hızlı çalışan ve gelişmiş veri madenciliği kabiliyeti olan PYTHON programlama dili kullanılmıştır. Veri analizi yaparken işin büyüklüğü ile mantığı arasındaki uyumu yürütmekte başarılı olan PYTHON programlama dili, aynı zamanda bu alanda çok sayıda araç ve kütüphane bulundurmaktadır. Analizler yapılırken Numpy, Pandas, Scikit-Learn ve Matplotlib gibi veri işleme, analiz ve görselleştirme konusunda oldukça gelişmiş araçlara kütüphaneler kullanılmıştır.

4.2 Verinin Tanımlanması

Bu çalışmada, 2021 yılında Pamukkale Üniversitesi Hastanesi Endokrinoloji Polikliniği'ne başvuran hastaların sağlık verileri kullanılmıştır. Pamukkale Üniversitesi Hastanesi Bilgi İşlem veri tabanından alınan iki farklı veri dosyası kullanılmıştır: birincisi, hastaların tanı bilgilerini içeren ve hasta kimlik bilgileri, tarihler ve tanı kodları gibi değişkenler içeren bir dosyadır. İkinci veri dosyası ise, aynı hastalardan alınan kan sonuçlarına ilişkin bilgileri içermektedir. Bu dosya, hasta kimlik bilgileri, tarihler, testlerin adları ve sonuçları gibi değişkenler içermektedir.

Toplanan bu veriler, diyabet, tiroit, hipofiz veya obezite hastalık tanısı konulan hastalara aittir. Verilerin analizi, hastalıkların teşhisinde ve tedavisinde kullanılan yöntemlerin geliştirilmesine ve hastaların sağlık durumlarının izlenmesine yardımcı olabilir.

Dosya içerikleri detaylı olarak incelenecek olursa birinci veri dosyası olan tanı verileri dosyası, sırasıyla ‘HASTA_NO, ‘PROTOKOL_NO’, ‘ADI’, ‘SOYADI’, ‘DOGUM_TARIHI’, ‘GELIS_TARIHI’, ‘KOD’, ‘TANI_ADI’ sütunlarından oluşan hasta bilgilerini, ikinci veri dosyası olan kan sonucu verileri ise sırasıyla ‘HASTA_NO’, ‘PROTOKOL_NO’, ‘RESMI_KOD’, ‘GELIS_TARIHI’, ‘ISLEM_TARİHİ’, ‘PARAMETRE_ADI’, ‘SONUC’ sütunlarından oluşan hasta bilgilerini içermektedir.

4.3 Verinin Hazırlanması

Bu veri seti, Pamukkale Üniversitesi Hastanesi Endokrinoloji Polikliniği’ne başvuran hastaların önemli bilgilerini içermektedir. İlk veri dosyası olan tanı verileri dosyası, her hastanın kimlik bilgilerinden (HASTA_NO, PROTOKOL_NO, ADI, SOYADI, DOGUM_TARIHI) ve hastalık bilgilerinden (GELIS_TARIHI, KOD, TANI_ADI) oluşan 8 sütundan ve 26589 satırdan(hasta) oluşmaktadır. İkinci veri dosyası olan kan sonucu verileri dosyası ise, her hastanın kimlik bilgilerinden (HASTA_NO, PROTOKOL_NO) ve kan sonuçlarından (RESMI_KOD, GELIS_TARIHI, ISLEM_TARIHI, PARAMETRE_ADI, SONUC) oluşan 7 sütundan ve 966021 satırdan(hasta) oluşmaktadır. Veri setinde genel hastalık tanısı olarak dört (diyabet, tiroit, hipofiz ve obezite) adet, alt tanılarla birlikte toplam 35 hastalık bulunmaktadır.

Sonuç verileri dosyasında bulunan ‘SONUC’ kolonu haricindeki kolonlar incelendiğinde tekrarlayan satırlar tespit edilmiştir. 966021 satırdan 13837 satırı tekrarlı olduğu için bu satırlar veri setinden atılmıştır. Bir hasta farklı günlerde kan testi yaptırabileceği için bu değerler tekrar eden sonuç olarak değerlendirilmemiştir. Sonuç verileri dosyasında kan testleri bulunan hastaların sonuçları tek bir kolon içerisinde verilmiştir. Analizin yapılabilmesi için her bir hastanın bilgileri ve kan testi sonuçları tek bir satırda olmalıdır. Yapılan düzenleme ile ‘PARAMETRE_ADI’ kolonundaki kan testleri isimleri yeni sütunlara dönüştürülmüştür.

Her iki dosyadaki bilgilerini birleştirebilmek için benzersiz olan ‘PROTOKOL_NO’ ve ‘HASTA_NO’ bilgileri ile ‘GELİS_TARIHI’ sütunları dikkate alınmıştır. Tek bir veri tabanı haline gelen verilerin analiz sonuçlarının daha doğru

çıkabilmesi için herhangi bir ‘PARAMETRE_ADI’ girilmemiş ve kabul edilen kan değerleri sınırlarına aykırı değerleri olan satırlar da veri tabanından silinmiştir. Geriye kalan parametre adları Tablo 4.1’de verilmiştir. Ayrıca cinsiyetler arasında, hamilelik durumu ve kişinin yaş durumu ile farklılık gösteren kan testleri sütunları da silinmiştir. Analiz için geriye 16585 satır ve 21 sütun kalmıştır.

Tablo 4.1: Kullanılan parametreler.

No	Değişken Adı	No	Değişken Adı
1	ALT (Alanin Aminotransferaz)	12	Kolesterol
2	ALP (Alkalen Fosfataz)	13	Kortizol
3	AST (Aspartat Transaminaz)	14	Kreatinin
4	Anti TPO	15	LDL kolesterol
5	Anti Tg(Anti Tiroglobulin Antikor)	16	Serbest T3
6	FSH	17	Serbest T4
7	Glukoz	18	TSH
8	HBA1C (%)	19	Trigliserid
9	HDL Kolesterol	20	VLDL Kolesterol
10	İnsülin	21	Üre
11	Kalsiyum (Ca)		

Her bir kan değerinin düşük, normal veya yüksek olma sınırları farklıdır. Pamukkale Üniversitesi Laboratuvarının kullandığı sınır değerlerine göre her bir sütun kendi içerisinde etiketlenmiştir. Örnek verecek olursak ‘Serbest T4’ hormonunun sınır değerleri 0,91-1,58 ng/dL şeklindedir. Bir hastanın ‘Serbest T4’ değeri bu aralık içerisindeyse **Normal**, 0,91 ng/dL’den düşük ise **Düşük**, 1,58 ng/dL’den yüksek ise **Yüksek** olarak veri bilgisi değiştirilmiştir.

Modellenen algoritmada sözel değerler yerine ikili(binary) değerler kullanılmıştır. Sözel **Düşük, Normal** ve **Yüksek** etiketlerinin dönüştürülmesi için One Hot Encoding yapılmıştır. One Hot Encoding, etiket değerlerinin ikili olarak temsil edilmesi anlamına gelmektedir. Aynı örnek üzerinden devam edilecek olursa, ‘Serbest T4’ sütunu yerine sütunun altında bulunan etiketler için ayrı sütunlar oluşturulmuştur. Bu sütunlar ‘Serbest T4_Normal’, ‘Serbest T4_Düşük’ ve ‘Serbest T4_Yüksek’ olarak isimlendirilmiştir. Yeni oluşturulan sütunlarda hangi etiketi temsil ediyorsa 1, değilse

0 olarak değişmiştir. Tablo 4.2‘de oluşturulan yeni veri tabanından örnek görülmektedir. Aynı işlem ICD-10 kodlarına göre ‘TANI_ADI’ sütununa da uygulanmıştır.

Tablo 4.2: One Hot Encoding için örnek.

‘Serbest T4’		**‘Serbest T4_Normal’**	**‘Serbest T4_Düşük’**	**Serbest T4_Yüksek’**
Düşük	**One Hot Encoding**	0	1	0
Yüksek		0	0	1
Yüksek		0	0	1
Normal	=>	1	0	0
Yüksek		0	0	1
Normal		1	0	0

4.4 Algoritmaların Sonuçları

2021 yılında, Endokrinoloji Polikliniği'ne başvuran, diyabet, tiroit, hipofiz veya obezite hastalığı tanısı konmuş hastalardan 16,585'i analiz için uygun bulunmuştur. Bu hastalara ait yaklaşık 1 milyon veri analiz edilmiştir.

Tablo 4.3’te, MinSup=0.5 ve minConf=0.7 olarak belirlendiğinde, algoritma çıktısı olarak kural çıkaran hastalıkların ICD-10 kodları verilmiştir. Bu kodlar, hastalıkların tanımlanması ve kaydedilmesi için kullanılan uluslararası bir sınıflandırma sistemidir.

Tablo 4.3: Birliktelik kuralı oluşan hastalıkların ICD-10 kodları.

ICD_10 Kodları	Hastalık Adı
E66.0	Aşırı obezite, vücut kitle indeksi (BMI) 30 veya daha yüksek
E66.2	Aşırı obezite, alveoler hipoventilasyonla birlikte
E66.9	Obezite, tanımlanmamış
E03.8	Hipotiroidizm tanımlanmış, diğer
E03.9	Hipotiroidizm tanımlanmamış
E05.9	Tirotoksikoz, tanımlanmamış
E10.9	İnsülin bağımlı diyabetes mellitüs, komplikasyonları olmayan
E11.7	Diyabetli böbrek hastalığı
E11.9	Tip 2 Diyabet, kontrol edilmiş
E14.9	Belirtilmemiş şeker hastalığı
E22.1	Hiperprolaktinemi (Aşırı aktif hipofiz bezi)

Aşağıda ICD-10 kodları ile ilişkili hastalıklar hakkında daha geniş bilgiler verilmiştir (WHO, 2019):

E66: Aşırı obeziteyi ifade eder. Aşırı kilo, sağlık sorunlarına neden olabilen bir risk faktörüdür ve obezite ile ilişkili bir dizi sağlık problemi bulunmaktadır. Bunlar arasında diyabet, kalp hastalığı, yüksek tansiyon, uyku apnesi, kemik ve eklem hastalıkları yer almaktadır.

E03: Hipotiroidizm, tiroid bezinin yeterli miktarda tiroid hormonu üretmediği bir durumdur. Tiroid hormonu, metabolizmanın hızını kontrol eder ve vücuttaki hücrelerin normal çalışmasını sağlar. Hipotiroidizm semptomları arasında yorgunluk, kilo alma, depresyon, soğuğa karşı hassasiyet ve saç dökülmesi yer almaktadır.

E05: Hipertiroidizm, tiroid bezinin aşırı derecede tiroid hormonu ürettiği bir durumdur. Tiroid hormonu metabolizmayı hızlandırır ve vücuttaki hücrelerin normal çalışmasını sağlar. Hipertiroidizm semptomları arasında kilo kaybı, yüksek tansiyon, sinirlilik, terleme ve kalp çarpıntısı yer almaktadır.

E10: İnsüline bağımlı diyabet mellitusu (Tip 1), pankreasın yeterli miktarda insülin üretmediği bir durumdur. İnsülin, kan şekeri seviyesini düzenlemek için

gereklidir. Tip 1 diyabet semptomları arasında aşırı susama, sık idrara çıkma, yorgunluk, kilo kaybı ve bulanık görme yer almaktadır.

E11: İnsüline bağımlı olmayan diyabet mellitusu (Tip 2), vücudun insülini etkili bir şekilde kullanamadığı bir durumdur. Bu, kan şekeri seviyelerinin yüksek kalmasına neden olur. Tip 2 diyabet semptomları arasında aşırı susama, sık idrara çıkma, yorgunluk, kilo artışı, bulanık görme ve yaraların geç iyileşmesi yer almaktadır.

E14: Şeker hastalığı, kan şekeri seviyelerinin yüksek olduğu bir durumdur. Şeker hastalığı tipine bağlı olarak, kan şekeri seviyeleri normal aralıkların üzerinde kalır. Şeker hastalığı semptomları arasında aşırı susama, sık idrara çıkma, yorgunluk, kilo kaybı ve bulanık görme yer almaktadır.

Tablo 4.4'de minSup değeri [0.1, 0,7] aralığında değiştiğinde oluşan sık öğe seti kümesi sayısı ve oluşan kural sayıları gösterilmektedir. minSup ≥ 0.8 olduğunda herhangi bir sık öğe bulunmadığından minSup değeri bu aralıkta alınmıştır. Apriori, ECLAT, FP-Tree ve H-Mine algoritmalarından elde edilen sık öğe setleri aynıdır. Bu algoritmalar, veri kümesindeki sık öğeleri bulmak için farklı teknikler kullanırlar, ancak temel olarak aynı sonuca ulaşırlar. Bu nedenle, aynı veri kümesi üzerinde çalıştıklarında benzer sonuçlar üretirler. Ancak, performans, hız ve bellek kullanımı açısından farklılıklar oluşabilmektedir.

Tablo 4.4: minSup değerine göre oluşan sık öğe setleri ve kural sayısı.

MinSup	Sık Öğe Seti Sayısı	Kural Sayısı
0,1	7393	56504
0,2	1184	6675
0,3	288	1287
0,4	92	361
0,5	35	102
0,6	11	10
0,7	3	0

Tablo 4.5'te minSup=0.5 ve minConf=0.7 olarak belirlendiğinde ortaya çıkan sık öğe setlerini göstermektedir ve veri seti üzerinde yapılan FIM algoritmalarının sonuçlarını içermektedir. Çalışılan dört algoritmada da sık öğe setleri aynı bulunmuştur. Ancak sık öğe setleri incelendiğinde, tüm parametrelerin normal aralıkta olduğu ve olağanüstü bir durum olmadığı gözlemlenmektedir.

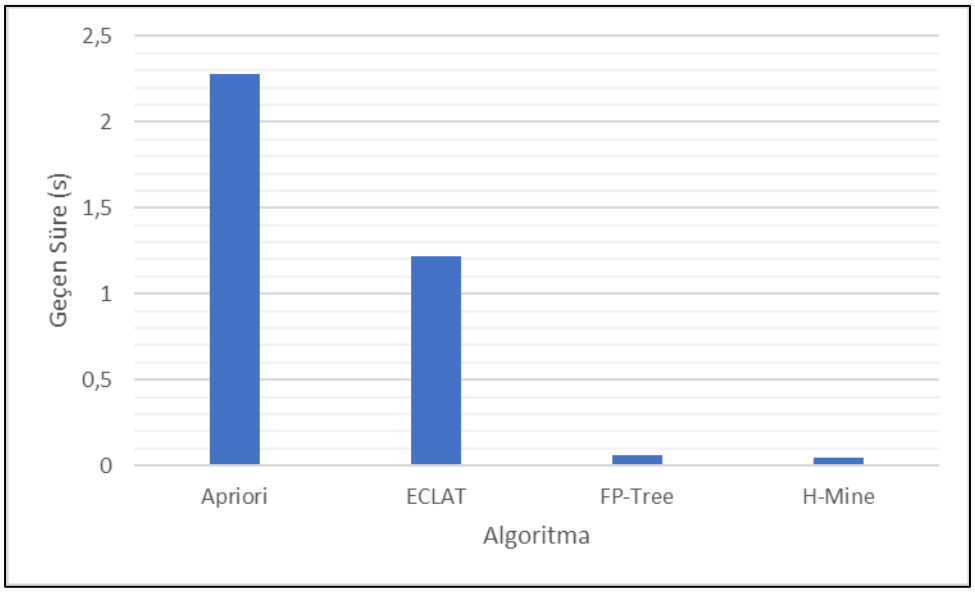
Toplam 35 sık öğe seti bulunmuştur ve bu öğelerin arasında E14_9 kodlu hastalık da yer almaktadır. E14_9(Şeker hastalığı) hastalığı için destek değeri 0,52 olarak hesaplanmıştır. E14_9(Şeker hastalığı) hastalığına ait destek değeri, veri setinde bu hastalığa sahip olan hastaların oranını ifade etmektedir. Yani, veri setindeki tüm hastaların %52'sinin E14_9(Şeker hastalığı) hastalığına sahip olduğu anlamına gelmektedir. Bu destek değeri, sık öğe kümesi analizi için önemlidir çünkü bir öğenin destek değerinin yüksek olması, bu öğenin diğer öğelerle birlikte sık olarak görülebileceği anlamına gelmektedir. Bu da öğeler arasında güçlü bir ilişki olduğunu göstermekte ve potansiyel olarak faydalı bir kuralın oluşabileceğini işaret etmektedir.

Tablo 4.5: Sık öğe setleri.

Support	Sık Öğe Setleri
0,76	Kalsiyum_Normal
0,75	ALT_Normal
0,71	VLDL Kolesterol_Normal
0,65	Trigliserid_Normal
0,69	TSH_Normal
0,52	Kolesterol_Normal
0,52	Kod_E14_9
0,68	Kreatinin_Normal
0,66	Kalsiyum_Normal, ALT_Normal
0,64	VLDL Kolesterol_Normal, Kalsiyum_Normal
0,63	VLDL Kolesterol_Normal, ALT_Normal
0,57	VLDL Kolesterol_Normal, Kalsiyum_Normal, _Normal
0,65	VLDL Kolesterol_Normal, Trigliserid_Normal
0,59	Kalsiyum_Normal, Trigliserid_Normal
0,58	Trigliserid_Normal, ALT_Normal
0,51	Trigliserid_Normal, TSH_Normal
0,53	Trigliserid_Normal, Kreatinin_Normal
0,59	VLDL Kolesterol_Normal, Kalsiyum_Normal, Trigliserid_Normal
0,58	VLDL Kolesterol_Normal, Trigliserid_Normal, ALT_Normal
0,53	Kalsiyum_Normal, Trigliserid_Normal, ALT_Normal
0,53	VLDL Kolesterol_Normal, Kalsiyum_Normal, Trigliserid_Normal, ALT_Normal
0,51	VLDL Kolesterol_Normal, Trigliserid_Normal, TSH_Normal
0,53	VLDL Kolesterol_Normal, Trigliserid_Normal, Kreatinin_Normal
0,60	Kalsiyum_Normal, TSH_Normal
0,59	TSH_Normal, ALT_Normal
0,56	VLDL Kolesterol_Normal, TSH_Normal
0,53	Kalsiyum_Normal, TSH_Normal, ALT_Normal
0,51	VLDL Kolesterol_Normal, Kalsiyum_Normal, TSH_Normal
0,62	Kalsiyum_Normal, Kreatinin_Normal
0,59	Kreatinin_Normal, ALT_Normal _Normal
0,58	VLDL Kolesterol_Normal, Kreatinin_Normal
0,53	TSH_Normal, Kreatinin_Normal
0,55	Kreatinin_Normal, Kalsiyum_Normal, ALT_Normal
0,53	VLDL Kolesterol_Normal, Kalsiyum_Normal, Kreatinin_Normal
0,52	VLDL Kolesterol_Normal, Kreatinin_Normal, ALT_Normal

Şekil 4.1'de minSup=0,5 ve minConf=0,7 olarak belirlendiğinde, veri tabanından sık öğe setlerini bulmak için kullanılan algoritmaların çözümleme süreleri verilmiştir: Apriori 2.2512 s, ECLAT 1,2187 s, FP-Tree 0,0625s ve H-Mine

0,047s'dir. Bu sonuçlara göre, H-Mine algoritmasının diğer algoritmalara kıyasla daha hızlı olduğu görülmektedir.

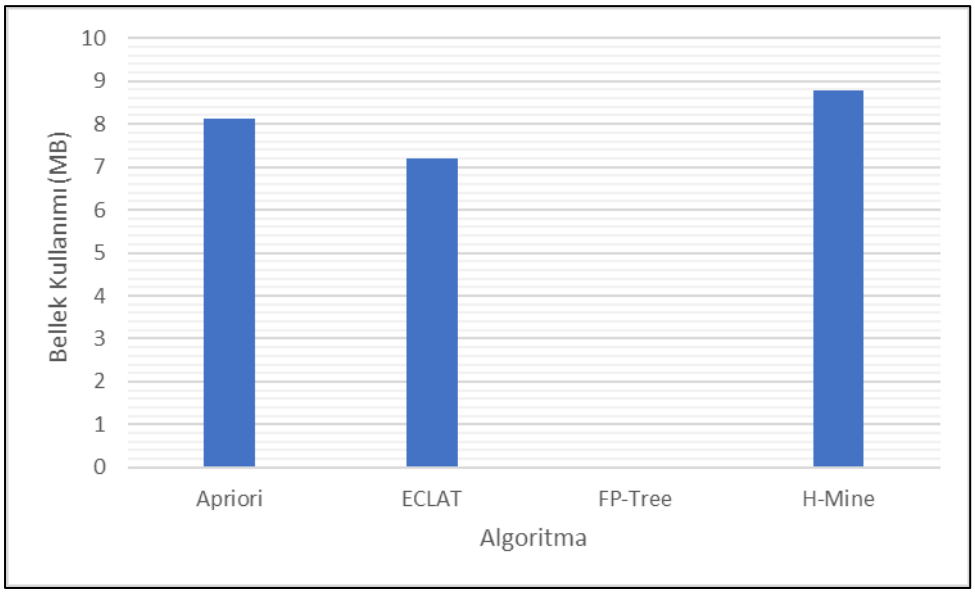


Şekil 4.1: Algoritmaların süre performansları.

Apriori algoritması, diğer algoritmalara göre daha yavaş performans göstermektedir, çünkü tüm olası öğe kümelerinin oluşturulması gerekmektedir. ECLAT ve FP-Growth algoritmaları, veri kümesi büyüdükçe daha yüksek performans gösterirler. Bunun nedeni, bu algoritmaların veri setindeki destek değerini geçemeyen öğeleri elemeleri ve sadece sık öğeler üzerinde işlem yapmalarıdır.

H-Mine algoritması, sık öğe kümesi madenciliği için oldukça hızlı bir algoritmadır. H-Mine, FP-Tree gibi diğer hızlı algoritmalarla kıyaslandığında benzer performans göstermektedir. ECLAT ve Apriori algoritmalarından daha hızlı çalışmaktadır.

Şekil 4.2'de minSup=0,5 ve minConf=0,7 olarak belirlendiğinde, veri tabanından sık öğe setlerini bulmak için kullanılan bellek durumu gösterilmektedir: Apriori 8,12 MB, ECLAT 7,19 MB, FP-Tree yaklaşık 0 MB ve H-Mine 8,77 MB bellek kullanmışlardır. Bu sonuçlara göre, FP-Tree algoritmasının diğer algoritmalara kıyasla çok daha az bellek kullandığı görülmektedir. Apriori ve ECLAT algoritmaları yaklaşık olarak aynı bellek miktarını kullanırken, H-Mine algoritması diğer algoritmalara kıyasla biraz daha fazla bellek kullanmaktadır.



Şekil 4.2: Algoritmaların bellek kullanımları.

ECLAT ve FP-Growth (FP-Tree) algoritmaları, Apriori algoritmasına göre daha az bellek kullanması beklenmektedir. Bunun nedeni, Apriori algoritmasının tüm sık öğe kümelerini bellekte tutması ve her iterasyonda tüm veriyi tarayarak bir sonraki seviyedeki öğeleri hesaplamasıdır. Diğer yandan, ECLAT ve FP-Growth algoritmaları, her seviyede sadece mevcut öğeleri tutarak ve birbirleriyle birleştirerek yeni öğeleri hesaplayarak bellek kullanımını azaltmaktadır.

H-Mine algoritması ise, veri setini bir dizi projeksiyon üzerinde işlemektedir ve bu nedenle bellek kullanımı orta seviyededir. H-Mine algoritması, önceden hesaplanmış özetlerle birleştirerek yeni sık öğe kümeleri bulmaktadır ve veri setini sadece bir kez taramaktadır. Bu nedenle, H-Mine algoritması genellikle çok büyük veri setleri için tercih edilmektedir.

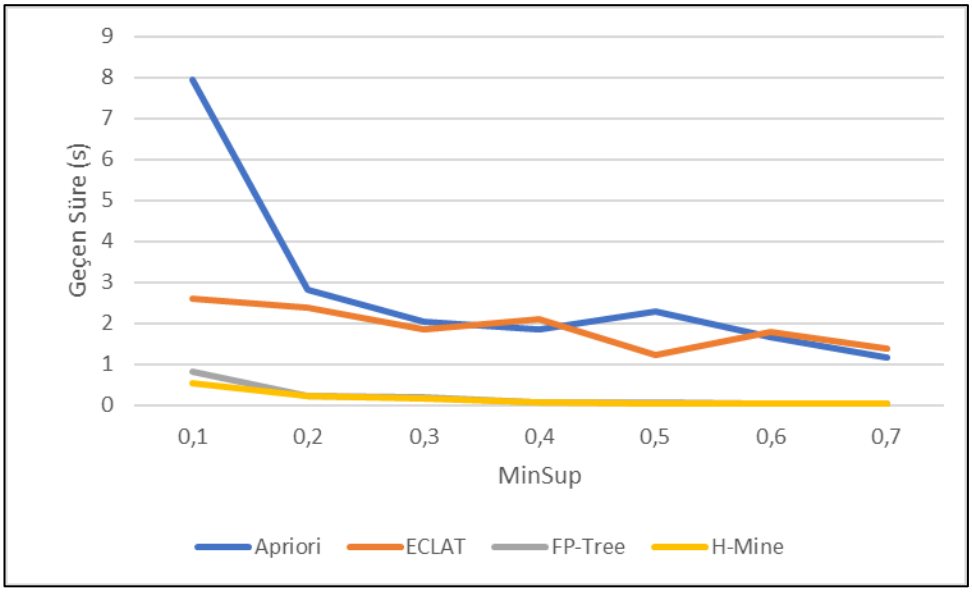
Bellek kullanımı, kullanılan veri kümesinin boyutuna, algoritmanın parametrelerine ve uygulamanın diğer özelliklerine bağlı olarak değişebilmektedir. Ayrıca, bellek kullanımı, algoritma kodunun nasıl yazıldığına da bağlıdır. Bellek yönetimi ile ilgili bazı iyi kodlama uygulamaları, verimli bellek kullanımına yardımcı olabilmektedir. Yapılabilecek başka iyileştirmeler ise verileri mümkün olduğunca küçük tutmak, gereksiz değişkenleri silmek ve gereksiz bellek kopyalamalarını önlemek, bellek kullanımını azaltmaya yardımcı olabilmektedir. Bu çalışmada kullanılan veri seti ve belirlenen parametrelerin dışında, algoritmaların performansları ve bellek kullanımları farklılık gösterebilir.

Tablo 4.6'da minSup değeri [0.1, 0.7] aralığında, algoritmaların analiz süreleri sn cinsinden verilmiştir. FP-Tree ve H-Mine algoritmaları, Apriori ve ECLAT algoritmalarına göre oldukça hızlı analiz yapmaktadır. Bu süreler, bilgisayarın işlemci durumuna göre değişiklik gösterebilir. MinSup değeri artıkça algoritmaların hızlarının düşmesi beklenen bir sonuçtur. Apriori ve ECLAT algoritmalarında, minSup değerine göre beklenmeyen durumlar oluşmuştur. Bunun nedeni, işlemcideki yük ve kodlamadaki değişken durumların olduğu düşünülmektedir.

Tablo 4.6: minSup değerine bağlı algoritmaların süre performansları(s).

MinSup	Apriori	ECLAT	FP-Tree	H-Mine
0,1	7,95	2,6093	0,8125	0,534
0,2	2,8281	2,39	0,23	0,22
0,3	2,0468	1,84	0,1875	0,1718
0,4	1,8593	2,09	0,07812	0,078
0,5	2,28125	1,2187	0,0625	0,047
0,6	1,6718	1,7812	0,03125	0,047
0,7	1,1562	1,375	0,0468	0,047

Şekil 4.3'te, yukarda verilen Tablo 4.6'deki verilerin grafiksel gösterimi verilmektedir. Grafikten görüleceği üzere, ortalama olarak en fazla süreyi Apriori algoritması kullanırken, FP-Tree ve H-Mine algoritmaları oldukça küçük sürelerde analiz yapmaktadır.



Şekil 4.3: Algoritmaların süre kullanımları.

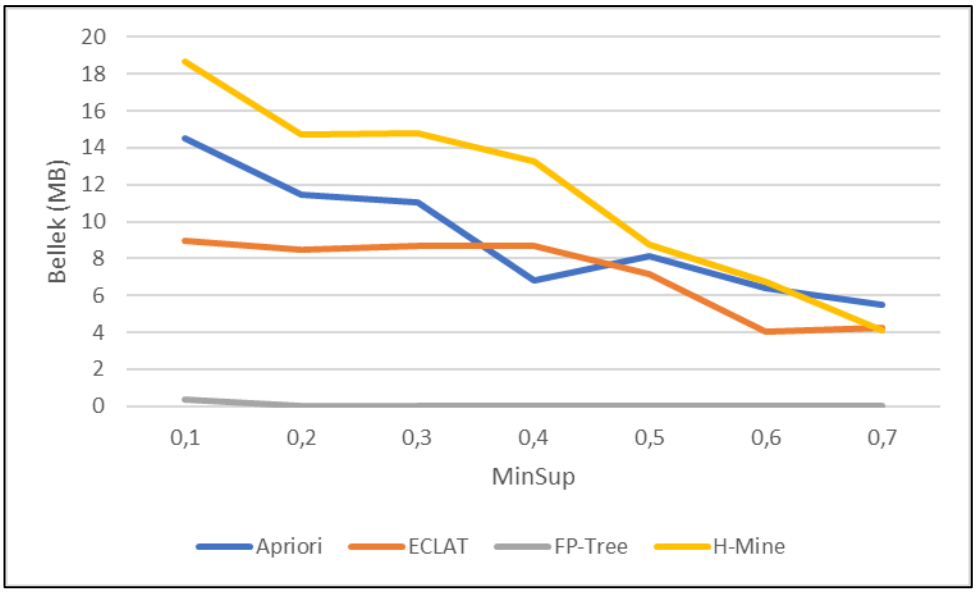
Tablo 4.7'de minSup değeri [0.1, 0.7] aralığında, algoritmaların analiz yaparken kullandıkları bellek miktarı MB cinsinden verilmiştir. FP-Tree algoritması en az bellek kullanımına sahipken, H-Mine algoritması diğerlerine göre biraz daha yüksek bir bellek kullanımına sahiptir. Bellek kullanımı açısından en avantajlı algoritma FP-Tree algoritmasıdır. Bunun nedeni, veri setindeki sık öğe setlerini bir ağaç yapısında tutması ve aynı veri kümesi üzerinde tekrar tekrar tarama yapmadan öğe setlerini oluşturmasıdır.

H-Mine algoritması, diğer algoritmalara göre daha yüksek bellek kullanımı gerektirebileceği durumlar oluşabilir. Bunun nedeni, algoritmanın yapısı gereği birçok frekans matrisi oluşturması ve bu matrislerin birbirleriyle birleştirilmesiyle yeni matrisler oluşturmasıdır. Ancak algoritmaların, bellek kullanımı, veri setine, veri setinin büyüklüğüne, minimum destek değerine ve algoritmanın diğer parametrelerine bağlı olarak değişebilir.

Tablo 4.7: minSup değerine bağlı algoritmaların bellek kullanımları (MB).

MinSup	Apriori	ECLAT	FP-Tree	H-Mine
0,1	14,5	8,99	0,3242	18,6992
0,2	11,48	8,51	0,04296	14,7155
0,3	11,08	8,7	0,0039	14,8024
0,4	6,8	8,71	0,00	13,2886
0,5	8,12	7,19	0,00	8,7798
0,6	6,39	4,02	0,00	6,7737
0,7	5,46	4,23	0,00	4,1362

Şekil 4.4'de, yukarda verilen Tablo 4.7'deki verilerin grafiksel gösterimi verilmektedir. Grafikten görüleceği üzere, en az belleği FP-Tree algoritması kullanmaktadır. MinSup değeri artıkça kullanılan bellek miktarının düşmesi beklenen bir sonuçtur. Ancak, bazı minSup değerlerinde beklenmeyen durumlar oluşmuştur. Bunun nedeni, işlemcideki yük ve kodlamadaki değişken durumların olduğu düşünülmektedir.



Şekil 4.4: Algoritmaların bellek kullanımları.

Bu çalışmada kullanılan veri setine ve parametrelere bağlı olarak, hız ve bellek kullanımına göre en uygun algoritmanın FP-Tree algoritması olduğu belirlenmiştir. Küçük eşik değerlerinde oldukça hızlı çalışması oldukça avantajlıdır.

5. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, temel veri madenciliği algoritmalarında 2021 yılında Pamukkale Üniversitesi Hastanesi Endokrinoloji Polikliniği'ne başvuran hastaların sağlık verileri kullanılmış, bu algoritmaların performans analizleri yapılmıştır. Ayrıca algoritmaların sonuçlarından elde edilen kurallar incelenmiştir. Bu algoritmalardan elde edilen örüntülerin ve kuralların tıp sektöründe bu alanda çalışanlara yardımcı olabileceği düşünülmüştür.

Sağlık sektöründe kullanılan teknoloji ve Sağlık Bilişim Sistemlerinin gelişimiyle birlikte bir kişinin sağlığı ile ilgili tüm bilgiler verinin oluşmasına ve depolanmasına olanak sağlamaktadır. Bu oluşan veri tabanları, tıp alanında veri madenciliğinin gelişmesinde büyük rol oynamaktadır. Veri madenciliği sadece bilimsel alanlarda değil aynı zamanda sağlık sektöründe kullanılan Sağlık Bilişim Sistemleri geliştiren şirketler tarafından da yaygın olarak kullanılmaya başlanmıştır. Veri madenciliği hastalığın teşhisi ve tedavisinde hekimlere karar verme kolaylığı sağlamakta ve hastalık teşhisinin doğru yapılmasıyla hastalığa konulan erken teşhis ve doğru tedavi, iyileşme süresini azaltmaktadır. Sağlık Bilişim Sistemlerinden mümkün olan fazlaca veriyi toplayarak, bu verilerin oluşturduğu veri tabanlarını uzman kişiler inceleyerek, sağlık politikalarına, ilaç üretim firmalarına ve bu alanda çalışma yapan herkese yön verilebilir hale gelmektedir.

Günümüzde her alanda olduğu gibi sağlık sektöründe de ciddi oranda veri bulunmaktadır. Hastanelere giden her bir hastanın testleri bir veri bankasında toplanmaktadır. Bu verilerin anlamlı hale getirilerek hangi hastalığın teşhis konulacağı konusunda yardımcı bilgilere ihtiyaç duyulmaktadır. Verileri anlamlı hale getirmede günümüzde yaygınlaşmaya en yaygın teknik veri madenciliğidir.

2021 yılında Pamukkale Üniversitesi Hastanesi Endokrinoloji Polikliniği'ne başvuran hasta verilerinden algoritmaya uygun 16585 hastanın kan testi sonuçları ve testlerde bulunan 21 adet kan değeri parametreleri kullanılmıştır. Kurallar oluşturulurken minSup değerleri [0.1, 0,7] aralığında alınmış, minConf değeri sabit 0.7 olarak kullanılmıştır. Bu parametreler kullanıldığında ortaya çıkan örüntüler ve kurallardan olağan üstü durumlar elde edilmemiş, hekimlerin tecrübeleri sonucunda bildikleri kurallar dışında bir bilgi bulunamamıştır. Bunun nedeni, güven değerinin

yüksek tutulması, veri sayısının yetersiz olması veya parametrelerin yetersiz olması olabilir.

Temel veri madenciliği algoritmalarından Apriori, ECLAT, FP-Tree ve H-mine algoritmaları bu veri setine uygulanmıştır. Her bir algoritmanın sık öğe seti bulma yöntemi farklıdır ve her bir algoritmanın avantaj ve dezavantajları bulunmaktadır. Algoritmaların sonuçları incelendiğinde, aynı minSup ve minConf değerlerinde oluşturduğu sık öğe setleri ve kurallar aynıdır. Fakat bu algoritmaların kullandığı yöntemlerin farklı olması sebebiyle algoritma hızlarında ve kullanılan bellek miktarında farklılıklar bulunmaktadır.

Algoritmaların hızları ve bellek kullanımları kıyaslandığında, bu çalışmada kullanılan veri setinin analizine en uygun algoritmanın FP-Tree algoritması olduğu belirlenmiştir. Çünkü FP-Tree algoritması, diğer algoritmalara göre daha hızlı çalışmakta ve daha az bellek kullanılmaktadır. Başka bir veri setinde ve giriş parametrelerinde algoritmaların performansları değişebilmektedir.

5.1 Öneriler

Hızla artan veri kayıtları ve bunların anlamlı hale getirilmesi için çeşitli algoritmalar ihtiyaç bulunmaktadır. Bunların içerisinde yaygın olarak kullanılmaya başlanan veri madenciliği algoritmalarına, sadece endokrinoloji polikliniğine başvuran 4 tip hastalık tanısı konulan hastaların verilerine uygulanmıştır. Veri elde edilmesi durumunda, veri madenciliğinin tüm polikliniklerden elde edilen verilere uygulanması gelecek çalışmalarda yapılabilir.

Bu çalışmada, veri madenciliği algoritmaları sadece sayısal verilerin analizi için kullanılmıştır. Ancak, tıbbi uygulamalarda sadece sayısal veriler değil, aynı zamanda hekimlerin yorumları gibi sözel veriler de önemlidir. Bu nedenle, tıbbi verilerin tam anlamıyla anlaşılması için sözel verilerin de analiz edilmesi gerekmektedir. Hekimlerin notlarını, hastaların semptomlarını ve tanılarını içeren tıbbi kayıtları analiz ederek hastalık örüntüleri ve tedavi yöntemleri hakkında bilgi sağlayabilir. Özellikle, sözel verilerin analizi sayesinde, hasta profilinin daha ayrıntılı

bir şekilde belirlenmesi, tedavi planlamasının optimize edilmesi ve tıbbi kararların alınmasında daha doğru ve verimli bir yol izlenebilir.

Veri madenciliği çalışmaları, genellikle uzun vadeli verilerin analizi ile daha kapsamlı sonuçlar elde etmek için kullanılabilir. Bu nedenle, bu çalışmada ele alınan hastalıkların incelenmesi için daha uzun vadede elde edilmiş veriler de kullanılabilir. Bu tür veriler, hastalıkların gelişimini ve ilerlemesini daha iyi anlamak için kullanılabilir. Ayrıca, daha uzun vadeli veriler, hasta tedavisi ve sağlık hizmetlerinin planlanması gibi konularda da kullanılabilir. Ancak, bu verilerin analizi için daha fazla zaman ve kaynak gerekebilir.

Güven değeri, veri madenciliği algoritmalarında önemli bir parametredir ve sıklıkla kullanılan bir ölçüttür. Bu çalışmada, güven değeri 0.7 olarak sabit tutulmuştur, ancak farklı bir güven değeri de seçilebilir. Güven değeri seçimi, oluşturulacak kuralların sayısını ve kalitesini etkilemektedir. Yüksek bir güven değeri seçilmesi, daha az sayıda kural oluşturmaktadır ancak bu kurallar daha güvenilir olmaktadır. Düşük bir güven değeri seçilmesi ise daha fazla kural oluşmasına neden olabilmektedir ancak bu kuralların güvenilirliği daha düşük olacaktır.

6. KAYNAKLAR

Aggarwal, C.C. Bhuiyan, M.A. and Hasan, M.A. "Frequent Pattern Mining Algorithms: A Survey" Springer International Publishing, 19-64, (2014).

Agrawal, R., Imieliński, T. and Swami, A. "Mining Association Rules Between Sets of Items in Large Databases", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 207-216, (1993).

Agrawal, R. and Ramakrishnan, S. "Fast algorithms for mining association rules." 20th International Conference on Very Large Data Bases, Vol. 1215, (1994).

Akgül, G., Çelik, A. A., Ergül Aydın, Z. and Kamışlı Öztürk, Z., "Hipotiroidi Hastalığı Teşhisinde Sınıflandırma Algoritmalarının Kullanımı", Bilişim Teknolojileri Dergisi, 13 (3), 255-268, (2020).

Can, M., Çamur, B., Koru, E., Özkan, Ö. and Rzayeva, Z. "Veri Kümelerinden Bilgi Keşfi: Veri Madenciliği", XIV. Öğrenci Sempozyumu Çalışma Grubu Sunumları, Başkent Üniversitesi, Tıp Fakültesi, sayfa 1-14, (2012).

Coşlu, E, "Veri Madenciliği", Akademik Bilişim, 23-25, (2013).

Çerkezi, M., "Veri Madenciliği Yöntemlerini Kullanarak Diyabetik Retinopati Hastalığının Teşhisi", Yüksek Lisans Tezi, Sakarya Üniversitesi Fen Bilimleri Enstitüsü, Sakarya, (2013).

Erdem, S.H. and Özdağoğlu, G. "Ege Bölgesi'ndeki Bir Araştırma ve Uygulama Hastanesinin Acil Hasta Verilerinin Veri Madenciliği ile Analiz Edilmesi", (2008).

Ersin Elbaşı, "Veri Madenciliği Yöntemleri Kullanarak Hastalık Teşhisi", 4. Mühendislik ve Teknoloji Sempozyumu, 28-29, (2011).

Ertuğrul, İ., Organ, A. and Şavlı, A., "Veri madenciliği uygulamasına ilişkin PAÜ hastanesinde hasta profilinin belirlenmesi". Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 19 (2), 97-103, (2013).

Farboudi, S., "Tıp Bilişiminde İstatistiksel Veri Madenciliği", Yüksek Lisans Tezi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, (2009).

Fournier-Viger, P., Lin, J. C. W., Vo, B., Chi, T. T., Zhang, J., and Le, H. B., "A Survey of Itemset Mining. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery", 7(4), (2017).

Gökbay, İ., Gökçek, S., "Endokrin Hastalıklarının Teşhisinde Klinik Karar Destek Sistemlerin Etkisi, Mühendislikte Güncel Araştırmalar", 47083(1), 117-131, (2022).

Güllüoğlu, S. "Tıp ve Sağlık Hizmetlerinde Veri Madenciliği Çalışmaları: Kanser Teşhisine Yönelik Bir Ön Çalışma". AJIT-e: Academic Journal of Information Technology, 2 (5), 1-7, DOI: 10.5824/1309-1581.2011.4.005.x, (2011).

Han, J., Pei, J. and Tong, H., Data Mining: Concepts and Techniques. Morgan Kaufmann, (2022).

Jain, A. K., Mao, J. and Mohiuddin, K. M., "Artificial Neural Networks: A Tutorial. Computer", 29(3), 31-44, (1996).

Jiang, L., Cai, Z., Wang, D. and Jiang, S., "Survey of Improving k-Nearest-Neighbor For Classification", Fourth International Conference On Fuzzy Systems And Knowledge Discovery, IEEE, Vol. 1. 679-683, (2007).

Kantardzic, M., Data Mining: Concepts, Models, Methods, And Algorithms. John Wiley & Sons, (2011).

Kaya, E., Bulun, M. and Arslan, A., "Tıpta Veri Ambarları Oluşturma ve Veri Madenciliği Uygulamaları", Akademik Bilişim, Çukurova Üniversitesi, Adana, (2003).

Kayaalp, K., "Asenkron Motorlarda Veri Madenciliği ile Hata Tespiti", Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü, Isparta, 1–45 (2007).

Koyuncugil A. and Özgülbaş N., "Veri Madenciliği: Tıp ve Sağlık Hizmetlerinde Kullanımı ve Uygulamaları", Bilişim Teknolojileri Dergisi, 2(2), (2010).

Koz, M. “Egzersizin Endokrin Sistem Üzerine Etkileri ve Hormonal Regülasyonlar.” Türkiye Klinikleri J Physiother Rehabil-Special Topics, 2(1), 48-56. (2016).

Kökver, Y., “Veri Madenciliğinin Nefroloji Alanında Uygulanması”, Yüksek Lisans Tezi, Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü, Kırıkkale, (2012).

Oğuztürk, G.,” Diyabet Hastalığının Makine Öğrenmesi Algoritmaları ile En İyi Doğru Tahminin Elde Edilmesi”, Yüksek Lisans Tezi, Bilgisayar Mühendisliği Anabilim Dalı, Kırıkkale, (2018).

Önder, G., Önder, E. and Özdemir, M., “Gelişmekte Olan Teknolojiler Sonucu Sağlıkta Oluşacak Yeni Meslekler”, Gümüşhane Üniversitesi Sosyal Bilimler Dergisi, 2019 Ek Sayı, 71-80, (2019)

Özdemir, A., Yalçın Aslay, F. and Çam, H., “Veri Tabanında Bilgi Keşfi Süreci: Gümüşhane Devlet Hastanesi Uygulaması”, Sosyal Ekonomik Araştırmalar Dergisi, 10(20), 347-366, (2010)

Pei, J., Han, J., Lu, H., Nishio, S., Tang, S. and Yang, D., “H-mine: Hyper-Structure Mining of Frequent Patterns in Large Databases.” IEEE International Conference on Data Mining, 441-448, IEEE, (2001)

Rasmussen, J. T., “Understanding The Hyperplane Of scikit-learn’s SVC Model”, (2022).

Roiger, Richard J., Data Mining: a Tutorial-Based Primer, Chapman and Hall/CRC, (2017).

Salazar, J., Espinoza, C., Mindiola, A. and Bermudez, V., “Data Mining And Endocrine Diseases: A New Way To Classify”, Archives of Medical Research, 49(3), 213-215, (2018).

Savaş, S., Topaloğlu, N. and Yılmaz, M., "Veri Madenciliği ve Türkiye'deki Uygulama Örnekleri", İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 11 (21), 1-23, (2012).

Savaş, S., and Topaloğlu, N., "Veri Madenciliği Yöntemi ile GSM Şebekelerinin Performans Analizi". Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 26(4), (2011).

Shearer, C., "The Crisp-Dm Model: The New Blueprint For Data Mining" Journal of Data Warehousing, 5(4), 13–23 (2000).

Sıtkı, Y. H., "Tıp Bilişiminde Veri Madenciliği Yöntemleri Kullanılarak Hastalıkların Tahmin Edilmesi", Yüksek Lisans Tezi, İstatistik Bölümü, Ankara, (2020).

Silahtaroğlu, G., Veri madenciliği, Papatya Yayınları, İstanbul, (2008)

Şentürk, Z.K., "Veri Madenciliği ile Kanser Tanısı", Yüksek Lisans Tezi, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, (2011).

Terzi, Ö., Küçüksille, E. U., Ergin, G. and İlker, A., "Veri Madenciliği Süreci Kullanılarak Güneş Işınımı Tahmini", SDU International Journal of Technological Science, 3(2), (2011).

Yücebaş, S. C., "Karmaşık Hastalıkların Teşhisinde Veri Madenciliği Yöntemlerinin Başarım Karşılaştırması". Çanakkale Onsekiz Mart Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 4 (1), 14-27. (2018).

Yücel, Y. B., Aytekin, A., Ayaz, A. and Tüminçin, F. "Bilişim Sistemlerinin Sağlık Sektörü Açısından Önemi", Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi, 5(8), Antalya, 147-155, (2018).

Zaki, M. J., "Scalable Algorithms for Association Mining", IEEE Transactions On Knowledge and Data Engineering, 12(3), 372-390, (2000).

Zhao, Q. and Bhowmick, S. S., "Association Rule Mining: A Survey", Nanyang Technological University, Singapore, 135, (2000).