SİVAS CUMHURİYET ÜNİVERSİTESİ
Sosyal Bilimler Enstitüsü
Yönetim Bilişim Sistemleri Ana Bilim Dalı

MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE ÇOKLU ETİKETLİ VERİLERİN SINIFLANDIRILMASI

Yüksek lisans Tezi

Mikail BARAN

Sivas
Şubat 2020

SİVAS CUMHURİYET ÜNİVERSİTESİ
Sosyal Bilimler Enstitüsü
Yönetim Bilişim Sistemleri Ana Bilim Dalı

MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE ÇOKLU ETİKETLİ VERİLERİN SINIFLANDIRILMASI

Yüksek lisans Tezi

Mikail BARAN

Tez Danışmanı
Doc. Dr. Mehmet Ali ALAN

Sivas
Şubat 2020

KABUL VE ONAY

Üniversite:	**: Sivas Cumhuriyet Üniversitesi**
Enstitü	**: Sosyal Bilimler Enstitüsü**
Ana Bilim Dalı	: Yönetim Bilişim Sistemleri
Tezin Başlığı	**: Makine Öğrenmesi Yöntemleriyle Çoklu Etiketli Verilerin Sınıflandırılması**
Savunma Tarihi	**: 23.01.2020**
Danışmanı	**: Doç. Dr. Mehmet Ali ALAN**

	Unvanı - Adı Soyadı	**İmza**
Jüri Başkanı	**: Doç. Dr. Ali Rıza İNCE**	
Üye	**: Doç. Dr. Ersin KARAMAN**	
Üye	**: Doç. Dr. Mehmet Ali ALAN**	

Oy Birliği ☒

Oy Çokluğu ☐

Mikail BARAN tarafından hazırlanan Makine Öğrenmesi Yöntemleriyle Çoklu Etiketli Verilerin Sınıflandırılması başlıklı tez, kabul edilmiştir. …./…./…….....

Prof. Dr. Ahmet ŞENGÖNÜL
Enstitü Müdürü

ETİK İLKELERE UYGUNLUK BEYANI

Cumhuriyet Üniversitesi Sosyal Bilimler Enstitüsü bünyesinde hazırladığım bu Yüksek Lisans/Doktora/Sanatta Yeterlik tezinin bizzat tarafımdan ve kendi sözcüklerimle yazılmış orijinal bir çalışma olduğunu ve bu tezde;

1- Çeşitli yazarların çalışmalarından faydalandığımda bu çalışmaların ilgili bölümlerini doğru ve net biçimde göstererek yazarlara açık biçimde atıfta bulunduğumu;

2- Yazdığım metinlerin tamamı ya da sadece bir kısmı, daha önce herhangi bir yerde yayımlanmışsa bunu da açıkça ifade ederek gösterdiğimi;

3- Başkalarına ait alıntılanan tüm verileri (tablo, grafik, şekil vb. de dahil olmak üzere) atıflarla belirttiğimi;

4- Başka yazarların kendi kelimeleriyle alıntıladığım metinlerini, tırnak içerisinde veya farklı dizerek verdiğim yine başka yazarlara ait olup fakat kendi sözcüklerimle ifade ettiğim hususları da istisnasız olarak kaynak göstererek belirttiğimi,

beyan ve bu etik ilkeleri ihlal etmiş olmam halinde bütün sonuçlarına katlanacağımı kabul ederim.

Mikail BARAN

TEŞEKKÜR

Lisans ve Yüksek Lisans eğitimim boyunca fikirleri ve desteğiyle beni doğru yönlendiren kıymetli danışman hocam Doç. Dr. Mehmet Ali ALAN' a, hayatımın her aşamasında yanımda olan değerli aileme, takıldığım her noktada bana zamanlarını ayırarak yardımcı olan değerli hocam Arş. Gör. Yunus Emre IŞIK' a, ve son olarak hiçbir zaman maddi ve manevi desteğini esirgemeyen sevgili eşime teşekkürlerimi bir borç bilirim.

Mikail BARAN

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

TABLOLAR DİZİNİ

KISALTMALAR

BR : Binary Relevance

CC : Clasisifier Chains

ÇES : Çok Etiketli Sınıflandırma

ECC : Esemble Clasisifier Chains

EML : Esemble Machine Learning

IDF : Inverse Document Frequency

LL : Lazy Learning

LP : Label Poverset

LR : Logistic Regression

ML : Machine Learning

NB : Naive Bayes

SVM : Support Vector Machine

TF : Terim Frekansı

YSA : Yapay Sinir Ağları

ÖZET

Günümüzde internetin gelişmesiyle beraber teknolojik altyapının sağlanması ve erişimin kısalması pek çok sektörde yeni alanların doğmasına olanak sağlamıştır. Artan ve bilgi içeren bu alanların birer kaynak haline gelmesi yenilik ve çeşitlilik açısından büyük fırsatlar doğurmaktadır. Veri olarak adlandırılan bu kaynakların hızla artması makine öğrenmesi alanındaki çalışmalara çeşitlilik ve önem kazandırmaktadır.

Yaygın olarak kullanılan verinin tasnif edilmesi, işe yarar bir duruma getirilmesi ve yeni bilgiler üretebilmesi adına bir sınıflandırmaya tabi tutulması gerekmektedir.

Çoklu etiket sınıflandırması, sınıfların birbirini dışlamadığı geleneksel tek etiketli sınıflandırmanın bir uzantısıdır ve her örnek aynı anda birkaç sınıfa atanabilir. Haberlerin sınıflandırılması ve görüntülerin sınıflandırılması gibi çeşitli modern uygulamalarda görülür.

Bu tezde ilk olarak teknolojinin gelişmesiyle artan verinin tasnifinde kullanılan çok etiketli sınıflandırma tanıtılmış, daha sonra makine öğrenmesi algoritmaları kullanılarak çok etiketli sınıflandırma bir film veri seti üzerinde modellenerek optimum başarının sağlanmasına yönelik testler yapılmıştır.

Anahtar Kelimeler: Makine Öğrenmesi, Sınıflandırma, Çok Etiketli Sınıflandırma.

ABSTRACT

Nowadays, with the development of the internet, the provision of technological infrastructure and the shortening of access have enabled the creation of new areas in many sectors. The fact that these increasing and knowledgeable areas become a source creates great opportunities for innovation and diversity. The rapid increase of these so-called data sources gives diversity and importance to the studies in the field of machine learning.

Commonly used data needs to be classified, put into a useful state, and subjected to a classification in order to produce new information.

The multi-label classification is an extension of the traditional single-label classification in which classes are not mutually exclusive, and each instance can be assigned to several classes at the same time. It is seen in various modern applications such as the classification of news and the classification of images.

In this thesis, firstly, the multi-label classification used in the classification of the increasing data with the development of technology is introduced, and then the tests are performed to achieve optimum success by modeling the multi-label classification on a film dataset using machine learning algorithms.

Keywords: Machine Learning, Classification, Multi - Label Classification.

BİRİNCİ BÖLÜM

1. GİRİŞ

Teknoloji ve bilim gibi yaşamın gelişen dinamiklerine bağlı olarak bilginin de pek çok faktör ile ilişkisi bulunur. Tarih boyunca insanlar, bir ihtiyaçtan diğerine adım adım geçen talepleri karşılamak için yaşamdan öğrendiklerini bilgiye aktarma ve daima geliştirme çabası içerisinde olmuşlardır. İnsan doğasına, bağlı olarak ihtiyaca göre bilgi toplamak ve topladığı bilgiyi kaydetmek her zaman bir gereksinim olmuştur. Yerleşik hayata geçildikten sonra yapılan araç ve gereçlerin nasıl yapıldığı ve nerede gereksinim olacağını gibi pek çok bilgiyi kaydetmek, işaret koymak, resim çizmek gibi ihtiyaçlar doğrultusunda bilgiye verilen değer artmıştır. Özellikle yazının icadından sonra kişiler arası sosyal durumlar, devletler arası anlaşmalar gibi pek çok yenilik insanları yeni keşifler yapmaya ve bilgiyi önemsemeye mecbur bırakmıştır. Yapılan sözlü anlaşmaların kayda alınması, matematiksel hesaplamalar, astronomi merakı gibi pek çok olay büyük kayıtların yapıldığı mekanlar olan kütüphanelerin oluşmasına zemin hazırlamıştır. Sanayi devriminden sonra özellikle iletişim araçlarının artması ve ulaşımda yapılan yeniliklerin, mesafelerin kısaltılmasını doğrudan etkilemiştir. Her şeyden öte insanoğluna özel bir güç ve yetenek olan merak etme duyusu sanayi devriminden sonra çok daha geniş bir boyuta ulaştığı görülmektedir.

İkinci dünya savaşı itibariyle ilk defa bilgisayarların kullanılması savunma sanayide çok büyük hesaplama yapacak araçların gelişmesi bilginin değerini ve önemini yeniden ortaya koymuştur. Nasıl oluşuyor sorusunun yanında nasıl oluştururum algısı oluşmaya başlamıştır. Sanayi devrimiyle ile beraber gündem olan üretme, 1970'li yılların sonundan itibaren ürettiğimi en iyi şekilde nasıl iyileştirerek satarım çabasına dönüştüğü görülmekte, bunu yaparken de mevcutta olan ürünlerin incelenmesi ve detayları analiz etme girişimleri devam etmiştir.

2000'li yılların başından itibaren internetin yaygınlaşması, teknolojik sosyal güçlerin sağlanması ve erişimin kısalması pek çok sektörde yeni alanların doğmasına olanak sağlamıştır. Her yeni olan teknolojik ürünün eskidiği gibi yerine daha gelişmiş bir ürün ve hizmetin ortaya konması, geçmişe olan bağlılığı bitirmek ile beraber yeniyi

arzulamaya sevk etmektedir. Teknolojinin kolay erişilebilir hale gelmesi kendiyle beraber sosyal sıkıntılar getirmiş olsa da teknolojiye düşkünlüğün artması ve tabiri caizse vazgeçilmezi haline gelmesi insanoğlunun bu durumdan pekte şikayetçi olduğu söylenememektedir. Tüketimin hat safa da olduğu bir dönemde teknolojik tüketimden bahsetmek çokta yanlış olmayacaktır. Teknolojik tüketimin sağlık-psikoloji tarafının önemli olmasının yanında, bu tüketimin kullanılabilir nasıl bir üretim ortaya çıkardığı herkes tarafından küçümsenmeyecek derece önemlidir.

Dünyada 7 milyar insanın doğrudan veya dolaylı olarak internete erişebiliyor olması dünyanın sosyal anlamda küçüldüğünü ancak fiziksel olmayan bir yığının ortaya çıktığı anlamına gelmektedir. Veri olarak nitelendirilen bu yığın, farkında olmadan milyonlarca insanın ürettiği bir bilgi haline gelmektedir. Yeni doğan çocukların dahi ilgisini çekmeye başlayan internet, her yaşa cevap verecek şekilde veri üretmekte ve veri ürettiği kişiden kendisine veri üretmesi için sosyal bir bağ kurmaktadır. Pek çok alanda dijitalleşme veya yenilikçi adımların atılması teknoloji çağında, ayakta kalabilmek için yeniliklerin günü birlik takip edilmesi ayrıca veri üretimine bir olanak sağlamaktadır. Buna bağlı olarak üretimin artması artan üretimin gereksinimlere cevap vermesi ve ihtiyaçların yan sektörleri geliştirmesi başlı başına birer veri konusu olarak ele alınmaya başlamıştır.

Bankalar, finans kurumları, eğitim kurumları, havayolu şirketleri, savunma sanayi kuruluşları adete bir teknoloji üreticisi haline gelmiş durumdadır. “Veriye sahip olan güce sahip olur” anlayışı ilerleyen dönemde verinin önemini ve ne kadar etkili bir silah olabileceğini göstermektedir. Yaşantımızın her safhasında veri ile iç içe olduğumuz aşikârdır. Mesajlaşırken atılan ses kaydı, atılan mailler, çekilen fotoğraflar, online yapılan alışverişler, yapılan para transferleri, hastanelerde yapılan testler ve daha birçok örnek, günlük olarak her bireyin ürettiği veri yığınına birer örnek olarak gösterilebilir.

Her veri kendiyle beraber yeni bir verinin ortaya çıkmasına neden olabilmektedir. Özelikle teknolojik cihazların etkisiyle veri üretiminin artması ve bu verilerden yeni bilgiler elde etmek için pek çok girişim yapılmaktadır. Veriyi işleme, veriden çıkarımlarda bulunma ve elde edilen bilgileri en ideal şekilde üretimden, son kullanıcıya kadar her aşamada bir veri incelemesi söz konusu haline gelmiştir. Veri

miktarının çoğalması ve insanoğlunu çoğalan bu verileri saklama, kullanma, analiz etme ve bu verilerden en iyi şekilde nasıl yararlanabiliriz sorusuna cevap olacak yeni alternatif çözümleri aramaya teşvik etmiştir. Her alanda veriye erişim kolaylaşırken, elde edilen verilerin ne kadarı kullanılabilir veya ne derece önemli olduğu sorusu her zaman canlı durmuştur. İnsanoğlu bilgiyi elde etmenin yanında günlük rutin yaptığı işlerde dahi veri tüketmekte ve tükettiği verinin pek çok katı kadar da veri üretir durumdadır.

Verinin bu kadar yaygın bir şekilde kullanılması veriyi tasnif ederek, işe yarar bir duruma getirilmesi ve yeni bilgiler üretebilmesi adına bir sınıflandırmaya tabi tutulması gerekmektedir. Verileri manuel olarak sınıflandırmak imkânsız hale gelmiştir. Örnek teşkil etmesi ve anlaşılması adına sadece 2017 yılında amazon.com'da listelenen ürün sayısının 400 milyondan fazla olduğu amazon tarafından belirtmiştir. Çeşitliliğin bu kadar fazla olduğu ve daha da artacağından şüphe duyulmamaktadır. Geliştirilen akıllı teknolojiler ile mevcutta olduğu gibi bu ürünleri sınıflandırma ihtiyacı doğmasına farklı bir neden ise mevcut ürünlerin birbiri ile olan ilişkisini ortaya koymaktır. Hangi ürün ne kadar satıldığı, satılan ürünün yanında nelerin satıldığı, gelecekte de bu verileri sınıflandırma açısından hayati bir rol oynayacaktır.

Sınıflandırma, bir ürünün veya verinin belirlenen ayırt edici özelliklere göre algoritmalar aracılığıyla gruplandırılmasıdır. Bu gruplandırmalar işletme kuralları, sınıf sınırları veya bir matematiksel fonksiyon ile tanımlanabilir (Alan 2014). Çok etiketli sınıflandırma ise bir ürünün veya verinin sonucunu beklenen gözlem sonuçlarını birden fazla etikete veya gruba atıldığı durumdur. Sınıflandırma günlük hayatımızda sık karşılaştığımız bir ifade olmakla beraber rutin olarak kullandığımız bir işlemdir. Mutfak için yiyecek sınıflandırılması yapıldığı gibi okul için ders başlıkları vb. sık kullanılmaktadır. Sınıflandırılan veri bir hastanenin son bir yılda hastalardan elde ettiği kan sonuçları olabileceği gibi bir bankanın bir yıl içerisinde bir müşterisine ait verilerin sınıflandırması da olabilmektedir. Sınıflandırılan verilerin nitelikleri sınıflandırılma amacı ile orantılıdır. Her verinin sınıflandırılmaya uygun olmadığı gibi her sınıflandırılan veriden de bir sonuç çıkarmak veya analiz yapabilmek doğru olmayabilir. Algoritmaların çok fonksiyonel çalışması veriler üzerinde sınıflandırma yapabilmeyi kolaylaştırmıştır. Bununla beraber bir veriyi sınıflandırmak

için pek çok algoritmaya tabi tutmak farklı sonuçlar üretebilmesini sağlayabilmektedir. Her gün dinlenilen müziklerin sınıflandırması, izlenilen filmlerin türüne göre çok etiketli sınıflandırılması, okunulan haberlerin veya kitapların türüne göre sınıflandırılması gibi pek çok yerde farkında olmadan çok etiketli sınıflandırma yapılmakta veya yapılmış şekilde karşılaşılmaktadır. Çok etiketli sınıflandırmaya örnek olarak, düzenli bir şekilde sinemada veya evde film izleyen bir kişinin izlediği filmleri kategorize etmek istenildiğinde; manuel olarak tek tek, filmlerde ortak oynayan oyuncuya göre sınıflandırması doğru olmayacağı gibi aynı yönetmen tarafından çekilen filmleri ya da aynı yıllarda çekilen filmleri not etmek de çok doğru olmayacaktır. Böyle bir sınıflandırmada başvurulması gereken husus izlenilen filmlerin türlerine göre ayırmaktır. Bir film ya drama ya da komedi veya macera konulu olabilmektedir. Eğer filmin drama, komedi, macera diyerek 3 sınıfa ayırılırsa tek etiketli sınıflandırma yapılmış olur. Bir film tek bir sınıfa ait olabileceği gibi iki farklı sınıfa da ait olabilmektedir. Eğer aynı film iki ayrı sınıfa ayrılırsa burada çok etiketli sınıflandırma söz konusudur. Filme hem drama etiketine hem de macera etiketine atanacaktır.

Çok etiketli sınıflandırma pek çok algoritma ile modellenebildiği yapılan çalışmalarda görülmektedir. Çok etiketli sınıflandırmada hangi algoritmanın nasıl bir veri setine uygulanması gerektiğini bilmek ise sadece veriyi analiz edecek kişiye kalmıştır. Her algoritmanın tüm veri setlerine aynı doğruluk oranını veremeyeceği gibi veriyi modellemek için kullanılan algoritmaların veri seti üzerinden çok düşük çıkması muhtemeldir. Bu doğrultuda çok etiketli sınıflandırma ile bir metin sınıflandırmasının planlaması durumunda, sınıflandırmada örnek olarak NB(Naive Bayes) ve SVM(Destek Vektör Makineleri) algoritmalarının kullanılması, veri setinde NB'nin %40 ve SVM'nin %70 oranında başarı sağlanması, NB'nin modelinin bu veri seti için uygun olmadığını ve SVM'nin bu model için uygun olduğunu göstermektedir. Farklı bir çalışma Duygu analizi yapmak için müzik veri setin de ise NB algoritması çok daha iyi bir modelleme başarısı sağlayabilir. Bu doğrultuda, sınıflandırmanın ne kadar önemli olduğunu ve içerisinde bulunduğumuz teknolojik gelişmenin etkisiyle daha da önemli olacağı görülmektedir. Literatürde pek çok çalışmada çok etiketli sınıflandırılma kullanılmıştır.

Elisseeff and Weston ‘nin 2001 yılında yaptığı çalışma da makine öğrenmesi çok etiketli sınıflandırma alanında yapılan ilk çalışmalardan biridir. Çok etiketli sınıflandırma için bir çekirdek metodu olarak yayınlanan makalede maya veri setinden faydalanmıştır. Veri seti 20,417 genden oluşmaktadır. 1500 geni eğitim seti olarak kullanırken 917 geni ise test setinde kullanmıştır. İkili ve sıralı SVM ile sınıflandırma da kullanılan çalışma da 1000 iterasyonda modelleme sonucunda %71 oranında başarı sağlanmıştır. Biyoenformatik ve metin madenciliği alanında birçok soruna neden olduğunu düşündüğü sıralamaya dayalı bir SVM tanımı getirmektedir. Sıralama problemlerine özellik seçimi yapmak için SVM’ nin genişletilebildiği vurgulanmaktadır. Biyoenformatikte denenen bu çalışmanın çok etiketli sistemler kullanarak fiziksel rahatsızlıkları ayırt edici bir dizi genden daha iyi bir şekilde faydalanılabileceğini belirtmiştir.

Yapılan çalışmalardan bir tanesi Matthew R. Boutella, M. Brown tarafından 2004 yılında çok etiketli bir resim sınıflandırması üzerinde yapılmıştır. Çalışmada eğitim verilerini daha etkili kullanılması amaçlanmıştır. Bu çalışmada görüntülerde bulunan dağ, deniz, orman gibi nesneleri çok etiketli bir sınıfa atamayı amaçlamıştır, yapay sinir ağlarının ve radyal temelli fonksiyonların kullanılmasının yanında SVM gibi algoritmalar ile eğitilerek bir model oluşturulmuştur. Yapılan testte modelin %81 oranında çok etiketli sınıflandırmayı doğru yaptığı gözlemlenmiştir. Radyal temelli fonksiyonların sahne sınıflandırma verilerinde bazı birleştirilmiş sınıflar için seyrek olduğunu ve uygulanan teknik ile performans artışı gözlemlemektedir

Makine öğrenmesi çok etiketli sınıflandırma alanında önemli çalışmalar yapan bir diğer isim Grigorios Tsoumakasdır. 2007 yılında çok etiketli sınıflandırmaya genel bakış başlıklı çalışmasında 3 farklı veri setinden oluşan; gen, maya, resim veri setleri üzerinden çok etiketli sınıflandırma yapmıştır. Gen sınıflandırmasında etiket sayısı 27, maya veri setinin etiket sayısı 14 ve resim de ise 6 etiketten oluşmaktadır. KNN ve NB algoritmalarından faydalanan Grigorios Tsoumakasdır gen data setinde KNN algoritmasından %90 üzeri başarı sağlarken, NB de daha düşük bir oran çıkmıştır. Resim ve maya veri setlerinde ise iki algoritma oransal olarak birbirine yakın ve %60 oranın test başarısı sağlamıştır. Yazar literatürde var olan yöntemlerin toplu olarak sunumunu yapmak ile beraber bazı metodları karşılaştırarak deney sonuçlarını ortaya koymaktadır.

2007 de Zhang and Zhou tarafından yapılan çalışmada üç farklı sınıflandırma ele alınmış maya geninin fonksiyonel analizi, doğal resim sınıflandırması ve metin sınıflandırması, yapılan çalışmada KNN gibi bir algoritma ile model oluşturulmuş. 2000 görüntü içerisinden başarı oranı %80, web sayfalarından elde edilen metin sınıflandırmasında başarı oranı %43 ve maya geninin fonksiyonel analizinde ise %76 oranında başarı sağlamıştır. 3 farklı gerçek dünya verisini çok etiketli öğrenme ile denenmesinden sonra ML-KNN algoritmasının bazı noktalar da iyi etiketlenmiş çok etiketli öğrenme algoritmalarından daha iyi bir performans gösterdiğini sonucunu çıkarmıştır.

2011 yılında Chou vd. tarafından" Singlepleks ve multipleks ökaryotik proteinlerin subselüler" lokalizasyonunu öngören çok etiketli bir sınıflandırıcı adlı yapılan çalışma da ise veri seti olarak 7,766 farklı ökaryotik protein dizisini içeren bir sınıflandırma üzerinden çalışmıştır. Yapılan çalışmada proteinler hem tek lokasyonlu hem de birden fazla lokasyona sahip değişebilen proteinleri içeren veri seti için KNN algoritmasını kullanmıştır. 22 etiketleme oluşturulan modellemede başarı oranı %71 olarak kayda geçmiştir.

Çok etiketli sınıflandırma alanında yapılan başka bir çalışma ise 2006 yılında Min-Ling Zhang ve Zhi-Hua Zhou tarafından yapılan fonksiyonel genomik ve metin kategorizasyonu uygulamaları ile çok etiketli sinir ağları çalışmasıdır. Çalışmada BPMLL algoritmasını kullanılmıştır. Bu çalışmada 21,578 reuters yayınlardan çıkan makalelerin çok etiketli sınıflandırma yöntemiyle sınıflandırılmasıdır. Bu makalelerin sınıflandırılması için yapay sinir ağlarından faydalanan Zhou modellemeden %96 oranında başarı sağlayabilmiştir. BPMLL sinir ağı algoritmasının öneren yazar, çok etiketli sınıflandırma öğrenmenin özeliklerini yakaladığını yani bir örneğe ait etiketlerin örneğe ait olmayan etiketlere göre daha üst seviyede tuttuğunu savunmaktadır. BPMLL'nin iyi etiketlenmiş çok etiketli öğrenme yöntemlerinde üstün bir performans sağladıklarını savunmuşlardır.

2010 yılında Huang vd. tarafından yapılan sınıflandırma için makine öğrenmesine dayalı optimizasyon yöntemi adlı çalışmada lösemi hastalarından alınan örneklerden oluşan iki ayrı veri seti kullanılmıştır. İlk veri seti 72 örneklemden oluşmakta ve her numune de 7,129 gen üzerinde ölçüm yapılmıştır. Diğer veri seti ise 22 normal 40 tümör dokusu örnekleminden oluşmakta ve her örnek 2,000 gen

içermektedir. Bu veri setlerini modelleyebilmek için ELM ile destek vektör makinelerinden yararlanılmıştır. Klasik SVM algoritmasından faydalanarak oluşturulan model de başarı oranı %84,36 iken ELM ve SVM'nin kullanıldığı teste başarı oranı %89,06 olduğu tespit edilmiştir. Bu makalede yazar ELM ve SVM algoritmalarının arasındaki ilişkiden ve optimize edilerek çalıştırılabileceğini ve uyumlu olduğunu tespit etmişlerdir. ELM'de ki sonuç ağırlık normlarının temelde iki farklı sınıfı ayırma marjının en üst seviyeye çıkarmayı başarmıştır.

Bir başka çalışma ise 2008 yılında Turnbull vd. Tarafından yayınlanan anlamsal açıklama müzik ve ses efektleri'nin alınması adlı çalışmadır. Bu çalışmada ses parçalarından anlamlı kelimeler türetmek ve etiketlenmemiş metinlerin okunması sırasında içerdiği sesleri tanıyabilme ve o sesleri seçip etiketleme üzerinde bir çalışma yapılmıştır. Yapılan bu çalışmada 500 adet batıda bilinen müzik parçalarından 1,700 adet ise insan kaynaklı oluşan bir veri kümesinden oluşmaktadır. Söylenen her kelimeyi GMM gauss karışım modeli ile eğiten yazar ortalama %78 test oranında modelleyebilme başarısını sağlamıştır.

2008 yılında Konstantinos Trohidis, Grigorios Tsoumakas, George Kalliris, Ioannis P Vlahavas yazarların ortak çıkardığı makale de müziğin çok etiketli olarak duygulara sınıflandırılması adlı çalışmada bir müziğin birden fazla duyguya sahip olabileceğini, tek etiketli sınıflandırma ile ya da regresyon ile bu kadar çok etiketli bir durumu modellenemeyeceğini savunmuşlardır. 6 duygu türü olarak etiketlenmiş 593 şarkının 72 müzik özelliği eklenerek yeni bir veri seti oluşturulmuştur. Çok etiketli sınıflandırmanın pek çok modeli ile test edilen çalışmada ikili sınıflandırma etiket güç seti KNN ve gibi algoritmalar ile test edilen veri setlerinden farklı sonuçlar elde edilmiştir. 4 farklı algoritmadan modelleyebilme oranı ortalama olarak %79 oranında başarı sağlanmıştır.

Grigorios Tsoumakas tarafından 2010 yılında yapılan başka bir çalışmada ise rastgele K etiketlerin çok etiketli sınıflandırılmasıdır. Bu çalışmada etiket kümelerinin Rassal olarak farklı etiket alt grup olarak farklı etiket kümeleri hedef alan LP etiket güç setini sınıflandırıcısını öğrenen RAKEL adında yeni bir yöntem sunulmaktadır. LP'nin eğitim örneklerinde bulunan etiket sayılarını ve hesaplamalarında basit ve performansının düşük olduğunu ortaya koyarak RAKEL ile LP'nin güçlendirilmesi çalışması yapılmıştır. Yapılan ölçümler sonucunda 2,407 resim 2,417 maya verisi

6,000 reuters metin dökümü ve daha birçok veri setini aynı anda SVM ve NB algoritmaları ile modellenmiştir. Bu modellemede SVM algoritmasının NB'ye göre daha iyi bir sonuç çıkardığı görülmüştür.

2019 yılında Yong Dai-Yi Li ve Li-Jun Liu tarafın yapılan önemli çalışmalardan bir tanesi çok etiketli sınıflandırma ile ürün fotoğraflarıyla yapısal tanımlama getirme çalışmasıdır. Her gün binlerce ürünün alım ve satışının gerçekleştiği online pazarlarda her alınan ürüne bir tanımlama getirmek hem maliyetli hem de zahmetli olmaktadır. Kurumların bu maliyeti düşürme çabalarına cevap niteliğinde olan bu çalışma ele alınan ürünlerin verileriyle oluşturulan model sayesinde bir tanımlama getirmektedir. Etiket güç seti ile çok etiketli sınıflandırma altında karar ağaçları ile oluşturulan modelde yaklaşık olarak %90 oranında başarı düzeyi sağlanırken, bunun yanı sıra ikili alaka düzeyi ile SVM ile oluşturulan modelde %96 oranında başarı düzeyi yakalanmıştır. İki farklı yaklaşımda da hemen hemen tüm özelliklerin oldukça yüksek bir puan aldığı görülmektedir. Çalışmada çok etiketli sınıflandırma ile ürünlere getirilen tanımlamaların tatmin edici olduğunu göstermişlerdir.

Khalil Laghmari, Christophe Marsala,Mohammed Ramdani tarafından 2017 yılında özet derecelendirme, çoklu etiket sınıflandırma (GMLC) üzerinde yapılan bir çalışmadır. İki farklı veri setini ele alan çalışmada online sitelerde tavsiye sisteminin kullanıcının 1 ila 5 arasında yaptığı oylama ile diğer kullanıcılara tavsiye edilen filmin sıralamada iyi bir tavsiye olamayacağının üzerinde durulmuştur. Hamming kayıp oranını düşük tutmayı hedefleyen çalışmada, GMLC ile iyi bir dereceye sahip olduğu halde beklenilen türe ait olmayan filmlerin sınıflandırılması noktasında yeni bir yöntem sunmuştur. 2 farklı veri seti 3 farklı dosya halinde ele alınmıştır. 6,040 kullanıcıya ait 1 ila 5 arasında 1 milyon değerlendirme ve son olarak 3,883 film üzerinden çalışma yürütülmüştür. Hamming, kaybını en düşük seviye tutmayı amaçlayan çalışmada iki kademeli çoklu etiket sınıflandırma modeline dayanan bir öneri sistemi sunmuştur. Kademelerden biri kullanıcıların özelliklerini ve derecelendirmesini ele alırken, diğeri örnekleri değerlendirip örneklerin özelliklerini ve derecelendirmesini niteler. Böylece yapılan bu çalışmada hamming kaybını 0,25 oranında tutabilmişlerdir.

İKİNCİ BÖLÜM

2. MAKİNE ÖĞRENMESİ

Terim olarak makine öğrenmesi günlük hayatta pek çok yerde insanların karşısına çıkmaya başlamıştır. Düşünce bazlı olarak makine öğrenmesi, 13.yy'a kadar dayanmaktadır. 1949 yılında Edmund Berkeley, Giant Brains (Büyük Beyinler-düşünen makinalar) adlı eserinde: "Son zamanlarda, devasa hız ve beceri ile bilgi toplayabilen büyük makineler hakkında çok sayıda haber var. Bu makineler bir beyininkine benzeyen kompleks yapılardan oluşuyor. Bir makine bilgi işleyebilir; hesaplayabilir, sonuçlandırabilir ve seçebilir; makul işlemleri bilgi ile gerçekleştirebilir. Bu nedenle bir makine düşünebilir" ifadesine yer verir (Berkeley, 1949 :31-65).

"Makine öğrenmesi" terimini ise 1959 yılında Arthur Samuel ortaya atmıştır. Bu terimi açıklamak için ise; "Makineler, program yazan kişinin oynayabileceğinden daha iyi bir drama oyunu oynamayı öğrenecek" demiştir. Bu yaklaşım ve tarihi süreçlerden sonra günümüze kadar pek çok bilim insanı makine öğrenmesi hakkında yeni ve farklı yaklaşımlar ortaya atmıştır (Akay 2018).

Günümüzde pek çok teknoloji kurum ve kuruluşları bilime katkı sağlamak için tıpkı bilim insanları gibi çalışmaktadır. Makina öğrenmesi alanında öne çıkan bu kuruluşların makine öğrenmesi hakkında tanımlarını ele almakta ve incelemekte fayda vardır. Bunlardan sadece birkaçı olan Nvidia, Standford gibi üst düzey çalışmalar sağlayan kuruluşlar aşağıdaki tanımlamaları yapmışlardır.

"En temelde Makine Öğrenmesi, verileri ayrıştırmak, ondan yeni bilgiler öğrenmek ve daha sonra dünyadaki herhangi bir şey hakkında bir belirleme veya tahmin yapmak için algoritmalar kullanma uygulamasıdır." (Nvidia Machine Learning 2019)

"Makine öğrenmesi, bilgisayarların açıkça programlanmadan ve herhangi bir işleme tabi tutulmadan harekete geçme bilimidir." (Stanford Machine Learning 2019)

"Makine öğrenmesi, belirlenen kurallara dayalı programlamaya dayanmadan verilerden öğrenebilecek algoritmalara dayanır." (McKinsey An Executives Guide 2019).

“Makine öğrenmesi algoritmaları, örneklerden genelleştirerek önemli görevlerin nasıl gerçekleştirileceğini çözebilir.” (Daniel 2019). Yukarıda sayılan kurum ve kuruşların yaptığı makine öğrenmesi tanımları, en çok bilinen ve yaklaşımları en doğru olan kabul edilen tanımlamalardır. Bu tanımlardan sonra en çok karşılaşılan tanım ise “Makine öğrenmesinin temelde makinanın tüm zorluk ve sınırlandırmalara rağmen insanlar gibi öğrenmelerini, düşünmesini, davranmalarını ve kendi sınırları içerisinde gerçek dünya verilerini gözlemleyerek veri ve bilgilerle beslenerek öğrendiklerini zaman içerisinde geliştirmesini sağlayan bilimdir”. Makine öğrenmesi daha önce gözlemlenmiş durumları ve tanımlanmış problemleri, belirli bir veri kümesine bağlı olarak makine öğrenme teknikleriyle çözüme kavuşturma işlemidir (Aydın 2018)

Teknolojinin ilerlemesine bağlı olarak makine öğrenmesi her sektöre girmiş ve etkili olmuştur. Girişte verinin önemine ve ne kadar büyük bir veriye sahip olduğundan bahsedilmiştir, makine öğrenmesinin temelinde mevcut verilerden yararlanmak vardır. Bu veri dinamiğini tıpta herhangi bir hastalık teşhisinde kullanılabildiği bildiği gibi otomasyonların gelişmesine bağlı olarak, kendi kendini süren araçların ortaya çıkmasına kadar pek çok yerde karşılaşılmaktadır. Her alanda etkisi görülen makine öğrenmesinin işlenebilir verilere sahip olması günlük yaşamda olabildiğince kolaylık sağlamaktadır (Büber 2018).

Makineye veriyi öğretecek olan algoritmaları tasarlarken ya da algoritmayı seçerken, algoritmanın öğrenme kapasitesinin en iyi performansa sahip olunması gerektiğine vurgu yapılır. 30 yıl önce gelişim süreci yavaş iken makine öğrenmesi sayesinden neredeyse 5 yılda bir yeni bir teknolojik döneme adım atılmaktadır. Veriye ulaşmak hiçbir dönemde bu kadar kolay olmadığı gibi elde edilen veriyi, analiz etmek ise makine öğrenmesi sayesinde bir hayli kolaylaşmıştır. Teknolojinin hızlı gelişmesine bağlı olarak yeni alanlar, yeni sektörler faaliyet göstermeye başlamıştır. Özellikle makine öğrenmesinin gelişmesi; kameralara takılan yüzlerin kayıt altına alınması, kapalı ortamlarda konuşan seslerin tutuluyor olması, kullanılan telefonlar ile çekilen fotoğraflar üzerinde çeşitli oynamalar yapabiliyor olması doğrudan veya dolaylı olarak makine öğrenmesinin oluşturduğu etkilerdir.

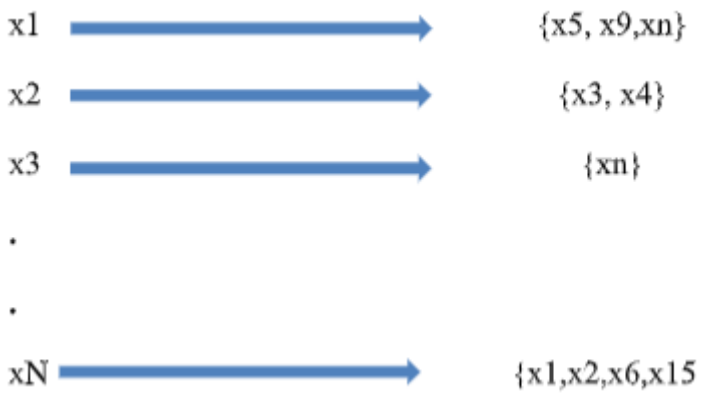
Arama motorlarında yapılan aramaların kayıt altına alınması, her an internet ortamına yeni bir veri yüklenmesi, arama motorunda ara butonuna basıldıktan sonra sayısal olarak kaç milyon sonuç getirdiği bunlara birer kanıt niteliğindedir. Google, Apple, Twitter gibi teknolojik şirketler milyarlarca dolar yeni makine öğrenmesi ve sinir ağlarını geliştirmek için harcamaktadırlar. Adeta bir bilgisayar olan telefonlarda sesli komutların veriliyor olması, çekilen fotoğrafların lokasyonu dahil her bilgiyi bilmesinin yanı sıra hangi fotoğraflarda kimin olduğunu bilmesi veya hangi sesin kime ait olduğunu dahi söyleyecek algoritmaların gelişmesi bu yatırımların neticesidir (Atalay vd. 2017).

Londra ve İsrail'de kişi başı düşen kamera sayısı neredeyse bir iken Çin'de bulunan kamera sayısı pek çok devletin nüfusundan fazla olduğu tahmin edilmektedir (Fatih 2019). Görüntü işlemenin ne kadar değerli bir kazanım olduğunu ortaya koymaktadır. Ülkeler elde ettikleri görüntüler ile herkesin bir nevi vesikalık fotoğraflarına sahip olduğun anlamına gelmektedir. Önceden sadece medyanın günlük olarak çıkardığı gazetelere nazaran, bir medya organının metin üretme sayısı, yüzlerce katına çıkmaktadır. Çıkan bu haberlerin tasnif edilmesi ve analizinin yapılması, bir kişinin bu işlemleri manuel olarak yapması neredeyse imkansız hale gelmiştir. Herkesin sosyal medyada bir haber üretiyor konuma gelmesi bu zorluğun göstergesidir. Kendi kendini geliştirebilen makinaların ortaya çıkması veri ve makine öğrenmesinin ortak bir sonucudur. İnsan hayatını kolay kılmak ile beraber olumsuz yönlerinin uzun vadede ortaya çıkması beklenmektedir. Son yıllarda bu gelişimlerin pek çok sektörde etkili olduğunu belirtilmesiyle beraber havacılık alanında yapılan gelişmeler makine öğrenmesinin ve sinir ağlarının etkisiyle neredeyse her yıl uzaya gidilmekte her ülke kendi insansız hava aracını üretebiliyor duruma gelmektedir. Makine öğrenmesi denetimli makine öğrenmesi denetimsiz makine öğrenmesi olmak üzere iki başlık altında toplanmıştır.

2.1. Denetimsiz Makine Öğrenmesi

$\{(X_i: i = 1, \dots, \dots, \dots, n)\}$ Denetimsiz makine öğrenmesi çıktısı olmayan ve verileri kendi içerisinde herhangi bir sebep-sonuca veya giriş-çıkışa bağlı olmadan etiketleme veya sınıflama yapmadan veri içerisinde var olan ilişkilerin ve birlikteliklerin öğrenilmesidir. Denetimsiz öğrenme, sistemlerin etiketlenmemiş

verilerden gizli bir yapı tanımlayan bir işlevi nasıl ortaya çıkardığını araştırır (Yılmaz 2019). Denetimsiz öğrenmede veri olarak girdi yalnızca X değeridir. Veri olarak girdi sağlayan durumun sadece bir öznitelik içeren vektörlerden oluşmaktadır. Denetimsiz öğrenme; temel olarak verinin sınıflandırılması, boyutlarının belirlenmesi ve verideki düzensizliklerin ortaya çıkarılması amaçları ile kullanılmaktadır (Şekerli 2019). Çıktının ne olduğu veya ne olacağı tahmin edilmemesi denetimsiz öğrenmenin doğrudan bir regresyona uygulanamadığını ortaya koymaktadır (Guru 2019).



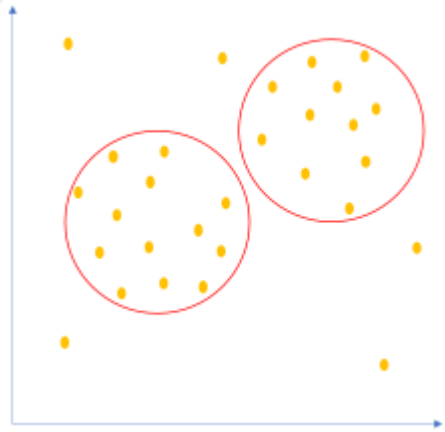
Şekil 1. Denetimsiz Öğrenme Yaklaşımı

Denetimsiz öğrenmenin temelde hedeflenen girdi olarak X verilerden daha fazla bilgi elde edebilmek için verilerde bulunan temel yapıları kendi içerisinden karmaşık bağlantıları ve dağılımı anlaşılabilir bir şekilde modelleyebilmesidir. Denetimsiz öğrenme denilmesinin altında algoritmaların verilerde ki tahmin edilemeyen yapıları keşfetmek ve bu yapıları sunmak için kendi aygıtlarının kullanılmasına olanak bırakılmıştır. Denetimli öğrenmede Örnek olarak, kedi ve köpek resimlerini algoritmaya girmesi halinde, algoritma kedi ve köpek resimleri ile eğitilir. Kendisine yeni bir resim verilmesi durumunda sonuç olarak size ya kedi ya da köpek sonucu verebilmektedir. Yani sonuçları ya da çıktı olarak etiketleri tahmin etmek zor olmayacaktır. Ancak denetimsiz öğrenmede kedi köpek resmini verdiğinde algoritma bu veri setlerini öğrenememektedir. Veri setleri üzerinde görüntüler arasında benzerliklerden faydalanarak kendi bağlantılarını kurup sonuç olarak benzerlik görevini kendi bağlamsal yapılar ile gerçekleştirmektedir (Birbil 2018).

Denetimsiz öğrenmenin kullanıldığı pek çok alan olmakla beraber bu alanlara örnek olarak; işletmelerin alışveriş miktarına göre müşteri gruplaması ve buna

Segmantasyon da denilmektedir. Müşterilerin yaptıkları alışverişe bakılarak müşterinin nasıl bir kesimden oluştuğunu anlamak için yapılan bir çalışmadır. Eğer segmentasyonu sağlanabilirse yani denetimsiz öğrenmeden faydalanabilirse hitap edilen kesim daha iyi tanınmış olacak ve bu müşteri kesimine göre pazarlama stratejileri belirlenecektir. Denetimsiz öğrenme tıpkı bir insanın kendi tecrübelerinden elde ettiği bilgi ve birikime göre düşünmeyi öğrenmesi gibi bir benzetme yanlış olmayacaktır (Microsoft 2019).

Makine öğrenmesinde denetimsiz öğrenme yöntemini çıktı değerinin olmaması uygulama esnasında gelecek yeni bir veri hakkında yorum yapılamayacağı anlamına gelmemektedir. Daha önce alışveriş yapan bir müşterinin gelmesi durumunda önce yapılan alışverişlere bakıldığında bir tespit söz konusu ise segmentasyon çalışması yapıldıysa aldıkları ürüne bağlı olarak müşterinin emekli mi öğrenci mi diye sınıflandırma yapmak mümkün olacaktır. Bir başka örnek ise bankaların hesap hareketlerinden yola çıkılarak bir müşterinin kara para aklama durumu nedir veya yolsuzluk yapabilme olanağını nedir gibi denetimsiz makine öğrenmesi konusu olarak seçilebilir (Birbil 2018).



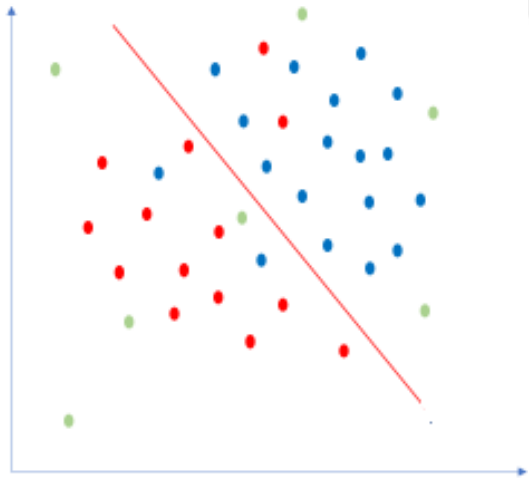
Şekil 2. Denetimsiz Öğrenme Gruplama

Verinin niteliğine bağlı olarak öznitelik sayısı algoritmanın performansını belirlemektedir. Denetimsiz öğrenmenin önemli noktalarından biri olarak görülen bu durum öznitelik sayısını azaltmadır. Öznitelik sayısı azaltılırsa tahmin edilen gruplama sınıflama veya kümeleme daha kolay yapılabilir. Denetimsiz öğrenmeyi etkili kılan üzerinde çalıştığı verilerin etkin olmayan ve herhangi bir kategorizeye tabi tutulmamış verilerin olmasıdır. Denetimsiz öğrenmede hedef değişkenin olmaması algoritma

olarak gizli yapıları öğrenmek ve analitik ilişkilerini çözmek için kullanışlı bir öğrenme yöntemidir. Veriler arasındaki örgütsel ilişkileri bulmaya çalışan denetimsiz öğrenme algoritmaları pek çok noktada denetimli öğrenmeden farklı çalışmaktadır. Örneklerin özniteliklerini ezberlemek yerine arasındaki karmaşık ilişkiyi anlamaya çalışmaktadır.

2.2. Denetimli Makine Öğrenmesi

$\{ (Xi, Yi) : 1 \ldots, \ldots, .. n \}$Eğitim verilerini göz önünde bulundurarak bir fonksiyon üreten makine öğrenmesi tekniği olarak bilinmektedir. Girdi verilerinin değerleri kullanılarak çıktı değerleri tahmin etmeye ve öğrenmeye çalışmaktadır. Bu süre zarfında öncelikle sonuçları bilinen veriler üzerinde bir sınıflama yapılır ve sonuçları bilinmeyen veri kümesi için sonuçlar tahmin edilmeye çalışılır (Aydın vd. 2015). Bu öğrenme tekniğinde her örnek bir giriş **(X)** nesnesinden ve istenen bir çıkış **(Y)** nesnesinden meydana gelen çift değerli öğrenme yöntemleridir. Ve girdi verileri ile beklenilen çıktı verileri arasında dengeli bir eşleşme yapan **Y= f(X)** fonksiyon ya da fonksiyonlar üreten öğrenme tekniği olarak bilinmektedir.



Şekil 3. Denetimli Öğrenme Gruplama

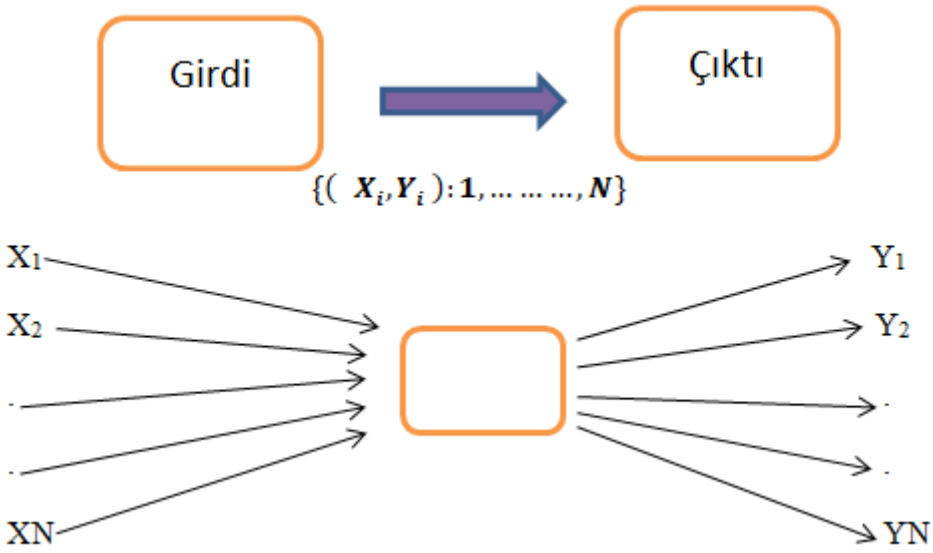
Denetimli bir öğrenme algoritması, mevcut eğitim verilerinin analizini yapar ve gelen yeni örnekleri kullanılabilir düzeye getirecek şekilde onları haritalamak ve onlardan bir çıkarım elde etmek için bir işlev üretmektedir. Belirlenen senaryonun aktif olarak çalışabilmesi için algoritmanın görünmeyen örneklerin sınıf etiketlerini doğru bir şekilde belirlemesini sağlar (Şanlı 2018).

Denetimli öğrenme teknik olarak eğitim aşaması ve test aşaması olmak üzere iki aşamada gerçekleşmektedir. Eğitim aşamasında makine öğrenmesi için mevcutta bulunan verilerin bir kısmını alır ve bunları çalışmanın şekline göre belirlenen denetimli öğrenme algoritmalarıyla inceler, veri için tahmin yapılacaksa, eğitim verileri üzerinden tahminde bulunmaktadır. Test verisi, eğitim verisini denetimli makine öğrenmesi algoritmasından geçirdikten sonra tahminlerin doğruluğunu test etmek için verinin bir bölümü bu alana ayırmaktadır. Eğitim ve test için veri setin ne kadar kullanılacağı kullanıcıya ya da analizi yapacak kişiye bağlıyken genel olarak tercih edilen bu oran eğitim verisi için %80 ve test verisi için %20 sini ayırabilmektedir. Böylece öğrenme tamamlandıktan sonra %20 test verisiyle test yapıp doğruluk oranını yükseltmeyi amaçlamaktadır.

Şekil 4. Denetimli Makine Öğrenmesi de Train ve Test Aşaması

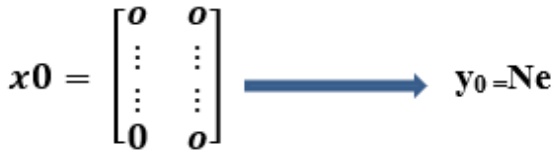
Denetimli öğrenmede en zaman alıcı süreç eğitim verilerin hazırlanması ve analizlerinin yapılması süreci olarak bilinir. Eğitim verileri hazırlanırken küçük detayların göz önünde bulundurulmaması eğitim verisiyle eğitilmiş olan sistemin beklenilenden uzak bir kötü tahminler yapması kaçınılmaz olarak görülmektedir. Örneğin, bir sistemin kendisine öğretilen resimlerde kedi, köpek ve balık olup olmadığını gösterecek bu durumu anlaşılır hale getirmek için belirtilen makine öğrenmesi algoritmalarına ihtiyaç duyulmaktadır. Makineye resimlerde kedi, köpek veya balık olup olamadığını tahmin etmek için algoritmalar aracılığıyla veri seti olarak sisteme bu resimlerin tanıtılması gerekmektedir. Bunun yanında resimlerde yüz tanıma, hasta bilgilerine teşhis önerme, bir bankaya müşterinin ne sıklıkla geldiğini vb. bu gibi ham kullanılabilir veri setlerine eğitim (train data) veri seti denilmektedir.

Eğitim veri setin de bulunan fotoğrafların kediye ait olduğunu belirtilmesi gerekmekte başka bir ifade ile etiketlenmesi gerekmektedir. Bu eğitim veri setleri sayesinde algoritma eğitilir. Veri setinde izlenilen yol aynı şekilde köpek ve balık eğitim setleri için de yapılmalıdır. Sistemi eğitmek için kullanılan veriden farklı bir veri seti daha oluşturulmaktadır. Test veri seti olarak bilinen bu verisine kedi, köpek ve balık fotoğrafları eklenmektedir, önceden tanımlanan fotoğraflar gibi hangisinin kedi, köpek ve balık olduğunu belirtilmediği gibi bir tanımlama da yapılmamaktadır. Yüklenilen test verisi fotoğraflar ile sistem test edilmektedir. İzlenilen bu yol sayesinden sistemimiz eğitim veri setlerinden elde ettiği kedi, köpek ve balık fotoğraflarına bakarak yeni gelecek olan resimlerde bu hayvanlara ait bir fotoğrafı gördüğünde bu hayvanlara ait olduğunu veya olmadığını söyleyebilecektir. Denetimli öğrenmenin grafiksel ifadesi daha anlaşılır olacaktır (Birbil 2018).



Şekil 5. Denetimli Öğrenme Girdi ve Çıktı Değerleri

Denetimli öğrenmede temel gereksinim; özniteliğin ne olduğuna bakarak test verisiyle çıktıyı tahmin etmeye çalışmaktır. Buna istinaden, burada **tahmin** kavramını ve **çıkarım** kavramını ele almak doğru olacaktır. Burada NE sorusuna cevap aradığını söylemek yanlış olmayacaktır yani öznitelik olarak ne alındı, buna bağlı olarak ne olması tahmin edilmektedir, öznitelik olarak kedi verildiyse sonuçların test verisinden sonra kedi veya değil olmasını beklenmektedir (Birbil 2018).



Çıkarımda ise soru NASIL olmalıdır. Ele aldığımız test verilerini bir sınıfa veya bir etikete attığımızı varsayalım bu işlem içerisinde hangi değişkenin ne tür bir rolü üstlendiği nasıl bir etkiye sahip olduğunu ve nasıl bir sonuca yol açtığını anlamak ve kavramak için kolaylaştıracaktır.

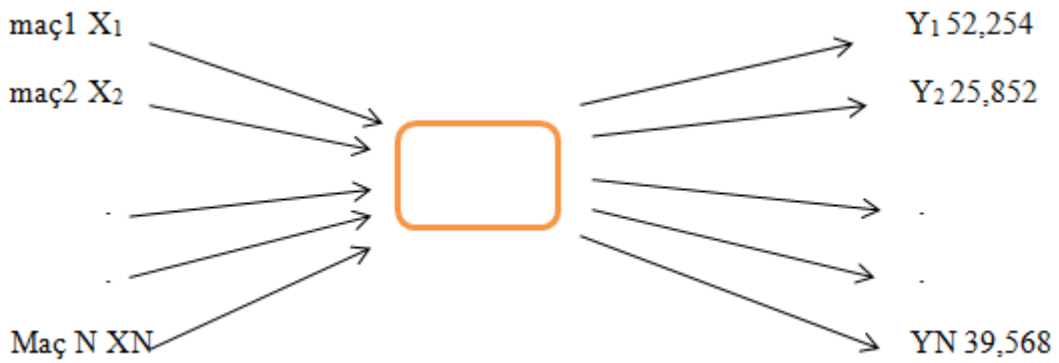
Denetimli öğrenmede çıktı değerinin türüne göre, regresyon ve sınıflandırma gibi iki ana gruba ayrılıyor; (Birbil 2018).

2.2.1. Regresyon Yöntemi

Regresyon analizi, değişkenler arasındaki neden-sonuç ilişkisinin bulunmasına imkân veren bir analiz yöntemidir (Gök 2017). $\{(Xi, Yi): 1 \ldots, \ldots, \ldots, n\}$ Giren özniteliğe bağlı olarak sonuç üreten ve test verilerini, tahmin çıkarımında kullanıldığını yukarıda belirtilmişti, fakat sonuca bağlı olarak tahmin edilen veya beklenilen değerlerin ne olduğu önem kazanmaktadır. Başka bir deyişle, bir veya birden fazla tahmine ait değişkenin (x) değerini baz alan devamlı olarak çıktı değişkenini (y) olarak tahmin edilmesine olanak sağlayan bir çeşit makine öğrenme yöntemidir. Regresyon analizi yönteminde değişkenler arası ilişkileri incelemede sıklıkla kullanılan istatistiksel yöntemlerden biri olarak kullanılmaktadır (Şenel vd. 2014).

Regresyon yönteminin amacı Y'yi X değişkenlerinin bir fonksiyonu olarak tanımlamak ve akabinde Matematiksel bir denklem oluşturmaktır. Oluşturulan bu denklem sonucunda Y'yi X yeni değerlerinin temelinde tahmin etmek ve kullanma amacı güder. Basit bir örnek ile ifade etmek gerekirse her yıl düzenli olarak yapılan bir müsabaka düşünüldüğünde bundan önce yapılan müsabakalara giden seyircilerin

ortalamasını ve stadın büyüklüğü göz önünde bulundurulduğundan dolayı maça katılacak kişiyi tahmin etmek için oluşturulan modelde eğer tahmin edilen sonucun veya beklenilen sonucun aşağıda belirtildiği gibi sayısal verilerden oluşuyorsa burada bir regresyon dan bahsetmek mümkündür. (Birbil 2018)



Şekil 6. Regresyon yöntemi Girdi ve Çıktı Değerleri

Regresyon analizlerinde sürekli bir Y değişkeninin tahmin edilmesini yardımcı olduğu gibi gerçek dünyada farklı kıtalarda ve sahalarda hava durumunu, üretimini ve geliri artırmayı düşünen işletmelerin satış tahminini ve buna bağlı olarak pazarlama eğilimlerini gibi gelecekteki öngörülere ihtiyaç duyulan çeşitli senaryolara bağlı olarak regresyon analizi pek çok kullanıldığı alanlar mevcuttur. Regresyon yönteminin kullanılmasının bazı avantajlardan birkaçı (Analytics 2015):

- Bağımsız değişken olan X ile hedef olan Y arasındaki muhtemel davranışları ve ilişkileri tahmin eder.
- Verilerin kullanılmak istenen amaca bağlı olarak eğilimlerini inceler ve arasındaki ilişkiyi sayısal olarak ifade eder.
- Verilerden elde edilen sonuçların gerçek ve sürekli olarak öngörmeye yardımcı olur.
- Regresyon metodunu kullanarak veriler içerisinde sonucu etkileyen en önemli faktörü buna karşılık en az önemli olan faktörü ve her bir faktörün diğer faktörleri nasıl ne kadar etkilediğini ve buna bağlı olarak bu faktörlerin sonuca nasıl bir etki ettiğini görebiliriz.

Aşağıdaki şekilde de görüldüğü gibi pek çok farklı türü olan regresyonun kullanım alanı gün geçtikçe artmaktadır. Petrol rezerv tahminlerinden kullanıldığı gibi hava sıcaklık durumunu ölmek için de kullanılır.

Şekil 7. Regresyon Türleri

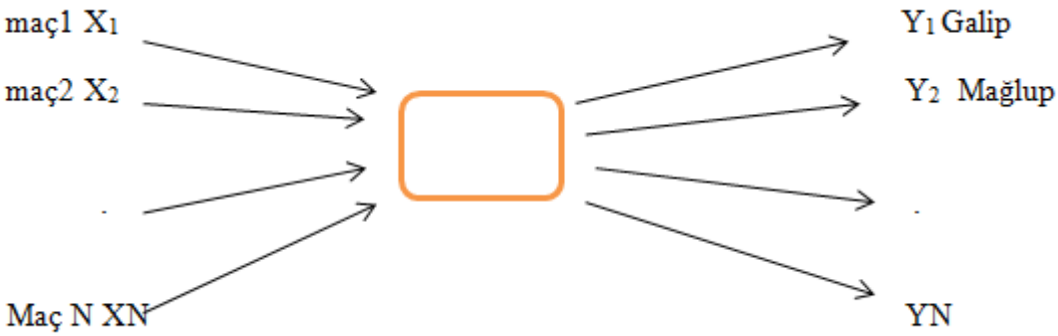
Regresyonun analizinin kullanılması bağımlı değişken ile bağımsız değişken arasında bulunan anlamlı ve örtülü ilişkileri göstermektedir (Orhun vd. 2017). Sayısı birden fazla olan bağımsız değişkenin bağımlı değişken üzerindeki etkisini göstermeye yardımcı olur. Regresyon analizi dönemler arasında değişkenlik gösteren faliyetlerin arasında ölçüm yapma olanağı tanır. Örneğin, gecen yıl kış mevsiminde tüketilen doğalgazın seviyesi ile bu yıl tüketilen gaz seviyeli karşılaştırmalı olarak ölçüm olanağı sunar.

2.2.2. Sınıflandırma Yöntemi

Sınıflandırma bir veri kümesinde bulunan değerlerin bazı özellikler göz önünde bulundurularak hedeflenen işleme en uygun algoritma veya algoritmalar aracılığıyla

belli kategorilere ayırma işlemidir. Sınıflandırma, farklı sınıflardaki, değişik öğeleri ayırma sürecidir (Alan 2014).

$\{x1, \ldots, xn\}$ X örneklerin etki alanı olmasını sağladığı ve n'nin bazı dağılımlara göre oluşturulan bir eğitim örnekleri kümesidir. Y ise hedef sınıflar kümesidir amaç görülmeyen her bireyin bağlı olduğu sınıfı öngören bir sınıflandırıcı bulunmasıdır. $\{(Xi\ , Yi\ ) : 1, \ldots \ldots \ldots, n\}$ Eğitim veri setinde bulunan $Xi.$ değer maçın çeşitli özniteliklerini içeren birer vektör olarak bilinir. Mevcutta bir futbol müsabakasına ait veri bulunduğunu varsayarak; verinin girdisi maç çıktısı ise farklı sınıflardan oluşacaktır (Birbil 2019).



Şekil 8. Sınıflandırmada Girdi ve Çıktı

Sınıflandırma diğer ismiyle etiketleme ise elimizdeki verinin sonucunda sayısal bir değer üretmek ya da elde etmek yerine bir etikete veya sınıfa atılmaktadır. Kategorik değişkenler olarak ta bilinen etiketleme modeli maç için kullanılan özniteliklerin sonucunda kimin kazandığını kimin berabere veya kimin mağlup olduğunu elde edilebilir.

Sınıflandırma genellikle yapısal veya yapısal olmayan veriler üzerinden yapılmaktadır. Yapısal veriler; belirli bir veri modeli üzerinde hangi bilginin nereden nasıl geldiğini bildiğimiz ve kullanırken sorulara cevap bulmakta zorlanılmadığı yapılardır. Nispeten yapısal veriler üzerinde işlem yapmak ve veriyi şekillendirmek kolaydır. Yapısal olmayan veriler; belirli bir veri modelinin olmadığı veri üzerinde bilgilerin karmaşık olduğu ve analiz yapmakta zorlanılan veri türleridir. Sınıflandırma ve regresyon arasında pek çok fark vardır bunlardan birkaçı şöyledir (Brownlee 2017);

a) Sınıflandırma işleminde ayrı sınıf etiketlerinden meydana geldiği ve tahmin yapılması bu model üzerine kurulu olan bir yöntemdir. Regresyon da ise devamlı olarak miktarı öngören bir modeli meydana getirme sürecidir.
b) Sınıflandırma da ele alınan veriler ayrık değerlerdir. Regresyonda ise sürekli değerledir.
c) Sınıflandırmada tahminin doğruluğu ölçülerek elde edilirken regresyonda kök ortalama kare hatası gibi yöntemler kullanılarak doğruluk ölçülür.

Sınıflandırma ikili sınıflandırma çok sınıflı sınıflandırma olarak ikiye ayrılmaktadır.

2.2.2.1. İkili Sınıflandırma

İkili sınıflandırma; verileri 0 veya 1 olarak iki kategoriye göre sınıflandırmasıdır. Bir skor tahmini üzerinden yapılan bu sınıflandırma pozitif veya negatif olarak sonuçlandırabilmek için bir değere (threshold) ihtiyaç duyulur. Karşılaştırma yolunu izleyen bu yöntemde eğer belirlenen eşik değerinden daha düşük olan veriler negatif **(0)** olarak sınıflandırılırken, eşik değerinden daha büyük olması halinde pozitif olarak sınıflandırılır **(1).** İkili sınıflandırmanın görüntü işleme üzerine savunma sanayi, insansız hava araçları gibi pek çok yerde kullanılmaktadır. Aşağıda Marmara Erdek sahili görselinde ikili sınıflandırmaya örnektir.

Şekil 9. İkili Sınıflandırma da 0-1 Değerleri

Modelden beklenilen veriler ile eldeki verilerin test edilmesi sonucunda modelin geçerliliği ve performansı ortaya çıkmış olur. İkili sınıflandırma modelinin tahminleri test verileri ile kıyaslandığında; doğru pozitif tahminler, doğru negatif tahminler, yanlış pozitif tahminler, yanlış negatif tahminler olmak üzere 4 farklı tahmin grubuna ayrılır (Amazon 2019).

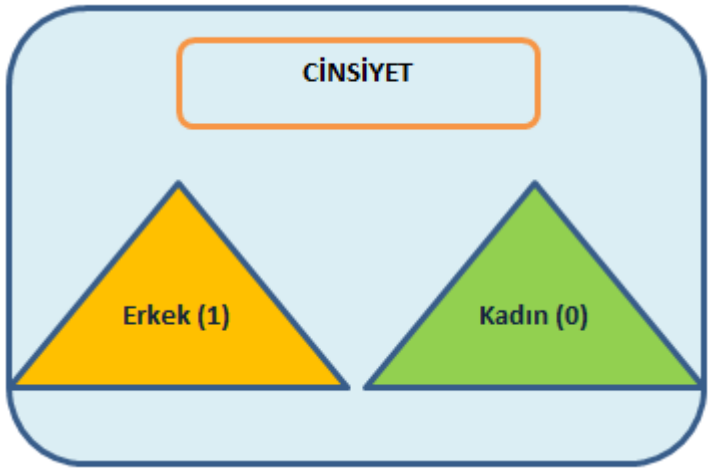
İkili sınıflandırmaya örnek bir bölgede nüfusun cinsiyet üzerinde oransal olarak nasıl bir ilerleme kaydettiğini ve geçmiş doğum oranlarının incelenmesi sonucu gelecekte nasıl bir kadın-erkek nüfus artışı izleneceğini varsayılabilir. Geçmişe dönük doğrumlar incelediğinde nüfus artış hızının yüksek olduğunu kabul edelim ve aralarında çok az bir oran ile erkek nüfusunun fazla olduğunu kabul ederek yol alınması ve aşağıda belirtilen beklentileri 4 farklı tahmin grubuna göre ayrılması ile;

a) sadece erkek çocuk doğma oranı yüksek

b) kız çocuk doğma oranı yüksek

c) hem erkek hem kız çocuk doğma oranı yüksek

d) her ikisinin doğma oranı düşük

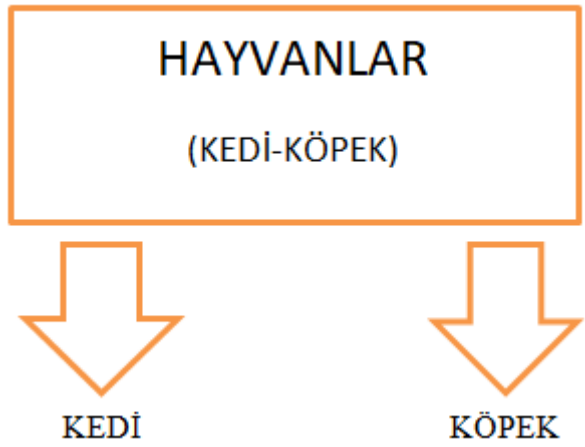


Şekil 10. 2 Sınıflı Sınıflandırma

- Doğru Pozitif tahmin(**c**)
  Doğru kabul edilen tahmin.
- Yanlış pozitif tahmin (**b**)
  Doğum oranı her iki cinsiyet için yüksek kabul edilmektedir
- Doğru negatif tahmin(**a**)
  Doğru ancak eksik bilgi bulunmaktadır doğum oranı yüksek az bir fark ile erkek oranı daha yüksek.
- Yanlış negatif tahmin(**d**)
  Her iki cinsiyet için doğum oranı yüksek kabul edilmektedir.

2.2.2.2 Çok Sınıflı Sınıflandırma

Denetimli çok sınıflı sınıflandırma her girdi için bir sınıf etiketi atamayı amaçlamaktadır. $yi \in \{1, \ldots, K\}$ i bir sınıf etiketidir (Aly 2005). En az iki veya daha fazla sınıfı sınıflandırma problemidir. Geniş anlamda bir sınıfa ait sınıflandırma söz konusu ise ve bu sınıflandırmada birden fazla veri var olduğu kabul edilir ve bu farklı veriler tespit edildikten sonra her biri kendi türünden tek bir etikete atanır bu doğrultuda tek sınıfın altında pek çok sınıf meydana gelir.



Şekil 11. Çok sınıflı Sınıflandırma

Eğer hayvanları bir sınıf olarak kabul edilirse kendi sınıfları içerisinde kedi ve köpekleri de birer sınıf olarak alınabilmektedir. Hem köpek hem de kedi birer hayvandır. Ancak her kedi bir köpek veya her köpek bir kedi değildir ikisinin aynı sınıfta olması söz konusu olamaz bunu genişletmekte mümkün olabilmektedir. Kedi veya köpekleri kendi cinsleri arasında sınıflandırabilir. Veri setinde pek çok tür kedi olabileceği gibi çok fazla köpek türü de olabilir önce kedi ve köpek sınıflarına ayrılır ardından her iki türün kendi altında sınıflar oluşturulup yeni kedi ve köpek türleri eklenebilir.

2.3. Makine Öğrenmesi Algoritmaları

Algortima belirlenen bir problemi çözmek için veya belirlenen bir hedefe ulaşmak için tasarlanmış en ideal yol olarak ifade edilmektedir. Makine öğrenmesi sayesinde, önceki tecrübelere veya örnek veri setlere dayanan bir işlemi optimize etmek için bilgisayarlar programlanabilir (Kızılkaya vd. 2018). Algoritmanın önemi

yapacağı işin değerine ve sürdürülebilir olmasına bağlıdır hayatımızın her tarafında algortima ile işlev gören makine ve cihazlar bulunmaktadır. Bir makinaya bir veri aktarmak makinanın veriyi analiz etmesi onu ezberlemesi ve yeni gelen verileri inceleyerek ezbeledikleri ile benzeyip benzemediğini anlaması gerekmekte ve bir makine bütün bu süreçleri algortimalar sayesinde yapılmaktadır. Kullanılacak olan algoritma ne ile sınırlandırılmalı veya hangi veriye hangi algoritmanın kullanılması gerektiği pek çok faktöre bağlıdır. Algoritmanın kullanıldığı sahaya uygun boyutta veriler olması gerekir (Utku vd. 2019).

Makine öğrenmesi çıktı değerleri kabul edilebilir ve belirlenen bir aralıkta girdi verilerini alan ve analiz eden algoritmalar kullanmaktadır. Mevcut verilerden beklenilen cevaba bağlı olarak seçilen algortimanın ne kadar doğru sonuçlar verdiği ne kadar parametre kullandığı algortima açısından önemlidir. Bu doğrultuda, doğru algoritmayı tercih etmek algoritmayı kullanılacak alan ve kendisinden beklentisi olan kurum ve kuruluşlara doğru yanıtlar verebilmesi önemlidir. Hangi algoritmanın hangi veriye uygulanması halinde tahmin edilen değerlerin alınması deneyimlenmeden elde edilmesi doğru olmayacağı gibi her algortima her veriye uygulanamayabilir. Makine öğrenmesinde pek çok algoritma vardır. Çıktıya bağlı olarak algoritmaların veri üzerinden modellenebilmesi ve başarı elde edilmesi için en ideal algortimanın tercih edilmesi önemlidir. Algoritmaları kendi içerisinde denetimli öğrenme algortimaları ve denetimsiz öğrenme algortimaları olarak ikiye bölünmektedir (Fumo 2017).

2.3.1. Denetimsiz Öğrenme Algoritmaları

Denetimsiz öğrenme çıktı değerinin eğitim sırasında bilinmediği gibi tahmin de edilememesi olarak ifade edilmiştir. Denetimsiz öğrenme algoritmalarını kümeleme ve birliktelik kuralı olarak iki başlıkta incelenmektedir;

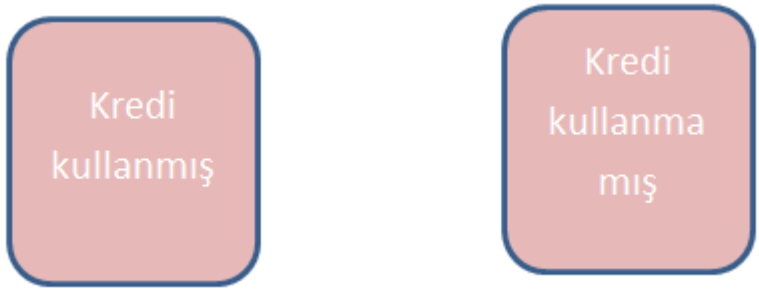
2.3.1.1. Kümeleme

Kümeleme yapısal olarak verilerin benzerliklerinin saptanması sonucunda birlikte gruplandırılma safhasıdır. Denetimsiz öğrenme veri noktaları arasındaki ilişkiden faydalanarak benzer noktalarını bulmak ve bu benzerliklerin kesiştiği noktaların arasında bir gruplama yapmaktır. Denetimsiz öğrenmede amaç, başlangıçta verilen ve henüz sınıflandırılmamış bir küme, veriyi anlamlı alt kümeler oluşturacak

şekilde öbeklemektir (Sarıman 2011). Gruplama yapmak benzer niteliklere sahip verileri incelemek ve analizi yapmak için olanak sağlamaktadır. Etiketlenmemiş veriler içerisinde nasıl bir gruplama yapılması istenildiğine bağlı olarak bu isteğe göre adım atılması veya karar verilmesi doğru olacaktır. Kümeleme işlemi tamamen gelen verinin özelliklerine göre yapılmaktadır (Dinçer 2006:22). Yapılan kümeleme veya gruplamanın en ideal olan kümeleme olduğunu söylemek doğru olmadığı gibi neden kümeleme gereksinimi duyduğumuz ile de ilişkilidir. Yani kümeleme yapacak verilerin hangi ihtiyaca nasıl bir cevap verdiği önemlidir. Kavramak için örnekleme yapmak gerekirse; milyonlarca müşterisi olan bir bankanın müşterilerine kredi vermek istiyor olması ve hangi müşterisine ne kadar kredi vereceğini hesaplamak için tüm müşterilerini tek tek manuel olarak kontrol etmesi hem çok zaman kaybetmek demek hem de çalışanlarının bu işe sevk etmesi maliyetini artırmak demektir.

Şekil 12. Gelire Bağlı Müşteri Kümelemesi

Yukarıdaki şekilde kümelemesi kredi kullandıracak olanları kullanmayacak olanlardan hızlı bir şekilde ayırmasını sağlamak için tüm müşterilerini gelirlerine göre gruplandırmaya gitmesi banka açısından tasarruf sağlayacaktır. Bu gruplandırmanın içerisinde de gruplandırma yapılabilir.



Şekil 13. Kümelemenin Bölünmesine Örnek

Örnek olarak yüksek gelirli müşteriler arasında (Şekil 13); Kredi kullanmış veya kullanmamış olarak da gruplanabilir. Yüksel gelirli müşteriler arasında böyle bir gruplandırmanın yapılması kredi kullanmış ve ödemeleri düzgün olan müşterilere tekrar kredi kullandırma olanağı ve teklifi sunulabilir. Bunun yanında, kredi kullanmamış olan müşterilere de kredi teklifi sunulabilir. Yukarı da ki şekillerin içerisinde bulunan kümelerin verilerin niteliklerine bağlı olarak hangisinin daha iyi bir kümeleme olduğunu savunmak yanlış olacağı gibi istenilen kümelemenin nasıl bir ihtiyaca cevap vermesi gerektiğini bilmekte fayda vardır. Bu doğrultuda, Kümeleme işlemlerini daha iyi kategorize etmek ve istenilen gruplandırmaları iyileştirmek için iki farklı algoritma tercih edilmektedir. Bunlar; K-Ortalama algoritması ve hiyerarşik kümeleme algoritmasıdır (Kırmızıbiber vd. 2019).

2.3.1.2. K- Ortalama Algoritması (K-Means)

En yaygın kullanılan denetimsiz öğrenme yöntemlerinden birisi olan K-means'in atama mekanizması, her verinin sadece bir kümeye ait olabilmesine izin verir (Çamurcu, Işık 2007). Algoritmanın temelinde birbirine benzeyen verilerin aynı kümeye alınmasıyla oluşur. K- ortalama algoritması sadece girdi vektörlerini kullanarak benzerlikleriyle oluşan veri kümelerinden çıkarımlar yapan bir algoritmadır. K değeri birbirine benzeyen verilerden oluşan kime sayısını belirlediği gibi ve K değerini bir parametre olarak alması da gerekmektedir. K- ortalama algoritmasının denklem sel olarak ifade edilmesi;

$$J(V) = \sum_{J=1}^{C} \sum_{J=1}^{Ci} (\|xi - vj\|)^2$$

- "$\|xi - vj\|$" x ve y arasındaki Öklid mesafesi.
- Ci, i. Kümede ki veri noktalarının sayısını ifade etmekte.
- C ise sadece küme merkez sayılarını ifade etmekte.

K-ortalama kümesinin algoritma olarak adımları:

$X = \{ x1 , x2 , x3 \ \dots\dots , xn \}$ Kümesi mevcut veri noktaları

$V = \{v1, v2, v3, \dots\dots vc\}$ ise merkez noktalarının kümesi olarak kabul edelir.

- Rassal olarak 'c' nin küme merkezleri seçilir.
- Kullanılan her veri ile küme merkezlerinin arasındaki mesafe hesaplanır.
- Kümelerin merkezleri ile arasındaki mesafe, diğer küme merkezleriyle olan mesafeden daha az olan verileri hangi kümeye daha yakın ise o küme merkezine atılır.
- Elde edilen küme yeni küme merkezleri aşağıdaki denklem ile tekrar hesaplamaya tabi tutulur.

$$Vi = (\frac{1}{ci}) \sum_{j=1}^{ci} xi$$

- Her elde edilen veri noktasıyla yeni küme merkezleri arasındaki uzaklığı yeniden hesaplanır.
- Eğer herhangi bir veri noktası ataması yapılmadıysa, aynı işlemin üçüncü adımından itibaren işlem tekrarlanır (Takaoğlu vd. 2019).

K-ortalama algoritmasının elde edilen sonuçların başarı düzeyini değerlendirmek için pek çok yöntem bulunmak ile beraber bunların en sık o-kullanılan yöntemi karesel hata toplamıdır (SSE = Sum of Squared Error). SSE Kümenin içerisinde bulunan elamanların merkez noktalarına olan uzaklığını ifade eder. SSE değerinin düşük olması beklenir. SSE nin değerinin düşük olması demek kümele işleminin başarı oranın yüksek olduğu anlamına gelir. K-ortalamanın bilinen en önemli dezavantajı veriyi kaç kümeden oluşturalacağını tahmin etmenin zorunluğudur.

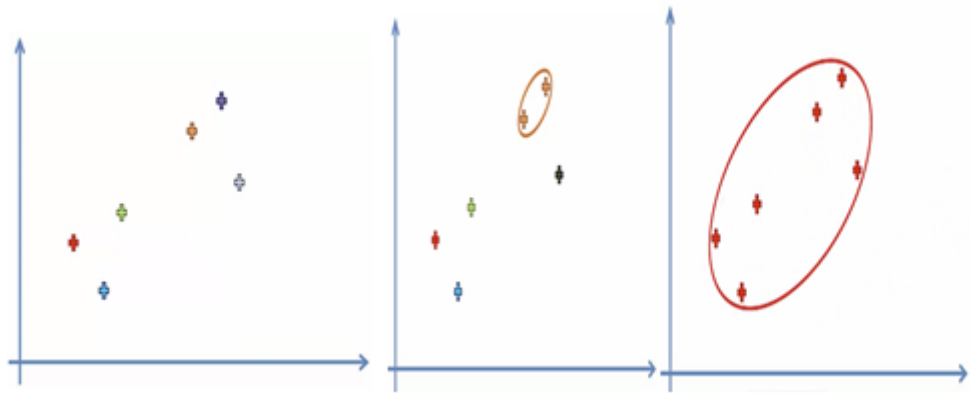
2.3.1.3. Hiyerarşik Kümeleme Algoritması

Anlaşılması kolay ve teknik olarak bilinen bir kümeleme algoritması yöntemidir. Tüm elemanlar tek bir küme gibi varsayılarak kümeleri birbirinden ayırarak farklı küme grupları oluşturmak veya her bir elemanını başlangıçta bir küme kabul edip birbirlerine benzer olan kümeleri tek bir küme altında toplamaktır (Delibaşoğlu 2016). Literatürde agglomerative(parçadan bütüne) ve divisive (bütünden parçaya) olmak üzere iki gruba ayrılır;

2.3.1.3.1. Parçadan Bütüne(Yığınsal)

Parça dan bütüne tekniği başlangıçta her veri noktası tamamen ayrı birer küme olarak kabul edilir. Benzer kümeler tek küme oluncaya kadar ya da K küme oluncaya kadar tek bir kümede birleşmeye devam ederler. Agglomerative izlenilen adımlar şöyledir;

- Her veri noktası bir küme gibi davranır. Böylece başlangıçtaki küme sayısı K olarak kabul edilir. K Veri noktalarını temsil eden birer tamsayıdır.



Şekil 14. Hiyerarşik Kümeleme Parçadan Bütüne

- K veri noktası bir kümeyi temsil etmek ile beraber kendisine en yakın küme ile bir küme oluşturur.
- İki kümenin birleşmesi ile ortaya yeni bir K kümesi meydana geldi. Benzerlik gösteren Kendisine en yakın küme ile birleşme işlemi veri sayısınca ve tek bir küme haline gelene kadar devam eder (Cory 2019).

2.3.1.3. 2. Bütünden Parçaya

Bütünden parçaya yöntemi Tüm veri noktalarının tek bir kümede toplanmasıyla başlamaktadır. Her biri sadece bir örnek içerene kadar kümeyi tekrarlı olarak çok küçük parçalara böler (Victor 2019).

Kümeler arasında mesafeyi ölçmek için farklı hesaplama yöntemleri kullanılır. Bunlardan öne çıkan iki hesaplama yöntemi Öklid ve manhattandır.

2.3.1.3.2.1. Öklid Hesaplama Yöntemi

$X = (a, b)$

$Y = (c, d)$

X ve y arasındaki Öklid mesafesi : $\sqrt{((\boldsymbol{a}-\boldsymbol{c})^{\boldsymbol{2}}+(\boldsymbol{b}+\boldsymbol{d}))^{\boldsymbol{2}}}$

2.3.1.3.2.2. Manhattan Hesaplama Yöntemi

$X = (a, b)$

$Y = (c, d)$

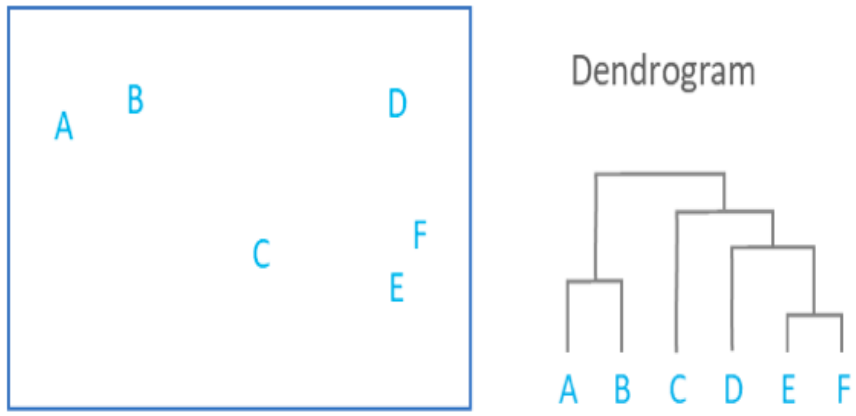
X ve y arasındaki manhattan mesafesi : $|\boldsymbol{a}-\boldsymbol{c}| + |\boldsymbol{b}-\boldsymbol{d}|$

ölçüm sırasında izlenilen yollar:

- İki küme arasında bulunan birleşme mesafesi ölçülür.
- İki küme arasındaki bulunan en uzak mesafa ölçülür.
- İki kümeye ait sendroidler arasındaki mesafa ölçülür.
- İki küme arasında bulunan olası tüm birleşim sağlayabilecek noktalar arası mesafe ölçülür(Usman 2018).

2.3.1.3.3. Dendrogram (Öbek Ağacı)

Benzer veri kümeleri arasındaki ilişkiyi gösteren veya hiyerarşik kümelenmeyi gösteren bir ağaç diyagramıdır. Bir veri kümesi içerisinde kaç adet küme oluşturacağımızı gösteren ağaç diyagramıdır. Öbek ağacın da en uzun bacaktan çizilen yatay çizgi kaç adet küme olduğunu göstermektedir.



Şekil 15. Dendrogram Kümelemesi

Yukarıda ki şekilde dendrogram şemasını kümelenmesi aşağıdaki gibidir.

- E---F bir kümedir(EF)
- A---B bir kümedir(AB)
- D---EF bir kümedir(DEF)
- C---DEF bir kümedir(CDEF)
- AB---CDEF bir kümedir(ABCDEF) (Ekrem 2018)

2.3.1.4. Birliktelik Kuralı

İlk defa 1993 yılında agrawal ve arkadaşları tarafından gündeme getirilen birliktelik kuraları piyasaya bağlı olarak gelişim göstermiştir. Veri madenciliği ve bilgiyi elde etme alanında kullanılan bir denetimsiz öğrenme yönteminin bir parçası olarak kabul edilir. Büyük veri kümeleri arasında birliktelik davranışlarını bulmaya yaramaktadır. Bu kurallar, bilinmeyen ilişkilerin tespit edilmesine ve daha etkin sonuçların çıkması için uygun karar vermeye imkân sağlamaktadır (Liao vd. 2007). X ve Y arasında bir ilişki söz konusu ise X'in Y'yi içerdiğine dair eğilimler ve bulgular olabilir buna mukabil Y içinde geçerlidir. Birliktelik kuralları potansiyel ürün ve hizmet grupları satışının söz konusu olduğu herhangi bir sektörde karşılaşılabilir. Pazar odaklı çalışma yapan kurum ve kuruluşların yaptıkları analiz, ürünler arasındaki ilişkileri göstermek için kullandıkları önemli tekniklerden biri olarak kabul edilir. Perakendecilerin, müşterilerin alışveriş sırasında sıklıkla beraber satın aldıkları ürünleri tespit etmeleri sağlar. Örnek olarak A ve B maddeleri sık satın alınan ürünler

olarak varsayalım nitelikli olarak birliktelik kuralına göre satışı artırmak için atılması gereken birkaç örnek adım;

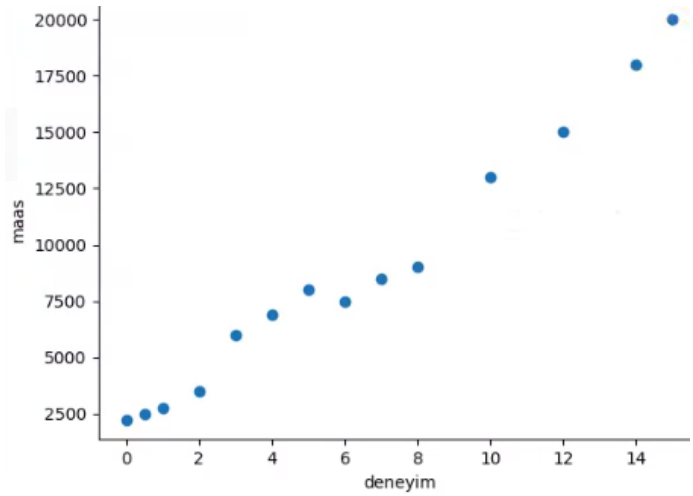
- A ve B ürünleri sık satılan ürünler ise mesafe olarak yan yana yerleştirilmesi halinde bir müşteri bir ürünü satın aldığında diğerini satın almak için çaba sarf etmez.
- Satışı artırmak adına A ve b ürünleri beraber paketlenebilir.
- Her ikisinin satın alınması halinde indirim uygulanabilir.
- Herhangi bir ürünü satın alan kişilere diğer ürünü satın almaları için görsel veya işitsel reklam kampanyası yürütülebilir.
- Aynı üründen iki tane alana yanında diğer üründen biri hediye edilebilir

2.3.2. Denetimli Öğrenme Algoritmaları

Denetimli öğrenme algoritması, girdi olarak verinin özelliklerini tutup hedeflenen çıktının nasıl bir sonuç doğurduğunu ve girdi ile tahmin edilen sonuç arasında nasıl bir ilişki olduğunu modelleyebilecek algoritmalar olarak geçer. Bu işlem sonucunda algoritma önceden kendisine sunulan ve öğretilen veriler arasındaki ilişkiye dayanarak çıktı değerlerini tahmin etmemize olanak sunar. Pek çok denetimli öğrenme algoritmaları bulunsa da biz en yakın komşu, karar ağaçları, doğrusal regresyon, naive bayes, lojistik regresyon, destek vektör makinaları ve yapay sinir ağlarını bu algoritmalara örnektir.

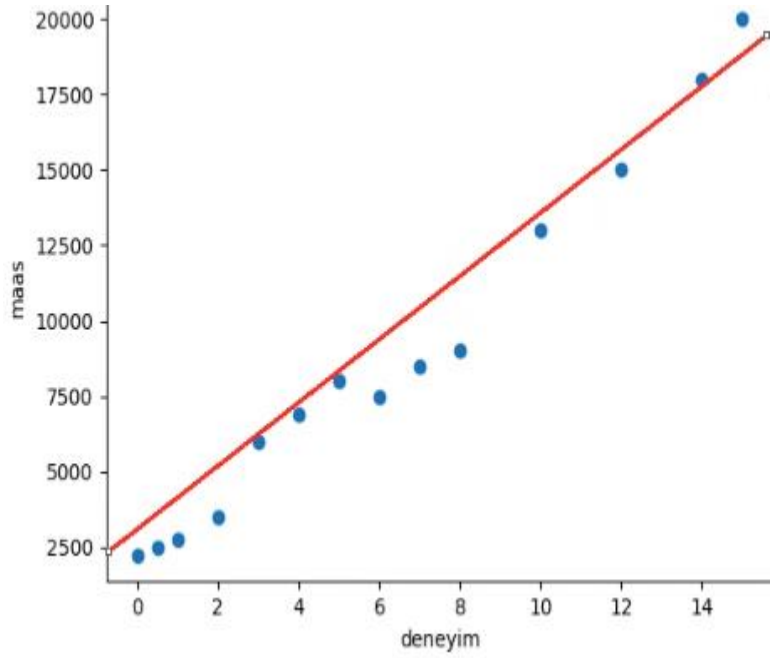
2.3.2.1. Doğrusal Regresyon

Doğrusal regresyon, sürekli değişkenler arasındaki ilişkiyi gösteren ve bu değişkenlerin arasındaki ilişkiyi istatistiksel olarak öngören bir makine öğrenmesi tekniğidir (Simple Linear Regression 2019). Bağımlı değişken X ekseni ile bağımsız değişken Y ekseni arasında doğrusal bir ilişki olduğunu varsayar. Tek bir giriş değişkeni (X) olduğunda method'a basit doğrusal regresyon birden fazla (x) değişkeni girişi olduğunda ise çoklu doğrusal regresyon denir. Basit bir doğrusal regresyon modellemek için çalışanlara ait deneyime bağlı maaş artırımını gösteren bir model aşağıda gösterilmiştir (Şahin 2018).



Şekil 16. Doğrusal Regresyon Maaş-Deneyim Grafiği

Tecrübe arttıkça maaş artan bu grafikte göz kararıyla bir kişinin 11 yıl çalışan bir kişinin tecrübesine bağlı olarak yaklaşık olarak 13,000 Tl maaş alacağını varsayabilir. Bu tahmini yaparken noktalar arasında bir ilişkinin olduğunu ve bu ilişkiden yola çıkılarak yapısal olarak bir model oluşturulmuştur. Doğrusal regresyonda da belirtildiği gibi, bir model oluştururken noktalar arasında ki ilişkiden yola çıkarak bir doğrunun çizilmesi beklenir.



Şekil 17. Maaş- Deneyim Doğrusal Çizimi

Grafikte maaş ve deneyime bağlı olarak çizilen doğrusalın veri setimizde bulunan değerlerin yani mavi noktalara doğru orantılı olarak çizildiği görülmektedir. Kırmızı doğrusalın matematikte ki karşılığı:

$$y = b0 + b1 * x$$

Y= y ekseni maaş

X= x ekseni deneyim

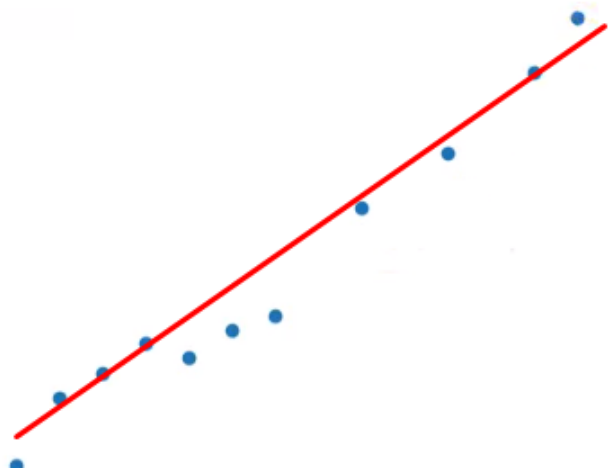
$B0$= sabit değer(bias) yani y ekseninde kestiği noktamız yukarıda ki grafikte 2500 kestiği noktadır

$B1$=katsayı yani bizim eğimimizdir

Kısaca $maaş = b0 + b1 * deneyim$ demekte herhangi bir yanlışlık olmayacaktır. Her oluşturulan regresyon modelinde tahmin modelinin performansı değerlendirmek kaçınılmazdır. Modeli oluşturmak için kullanılmamış test verisinin sonucunu öngörmede modelin ne derece başarılı olduğunu değerlendirmemiz gerekir. Tahmini regresyon modelinin performansını değerlendirmek için yaygın olarak iki ölçüm kullanılır;

2.3.2.1.1. Ortalama Karesel Hata

Sonucun gözlemlenerek bilinen değerleri ile model tarafından edilen değerler arasındaki ortalama farkıdır. Ortalama karesel hata(MSE) ne kadar düşük olursa model o kadar başarılıdır**.** Yukarıda çizilen grafikte fit edilen doğrusalda temel amaç belirtilen maaş ve deneyime bağlı olarak noktalara en yakın şekilde çizilmesi beklenilir. Bu doğrultuda aşağıda ki noktaların bazılarında fit edilen doğrusalın bazı noktaların tam olarak merkezinden geçerken bazılarına ise değmediğini görülmektedir.



Şekil 18. Doğrusalın Ortalama Kare Hatası

$$MSE = \frac{1}{N}\sum_{\text{İ}=1}^{N}(Yi - \widehat{Yı})^2$$

Matematiksel olarak ifade edilen değerde:

n = nokta sayısı

Yi = grafikte bulunan noktalar

$\widehat{Yı}$ = tahmin edilen değer

Noktalar ile tahmin edilen değerler arasındaki farkı bulmaktaki temel amaç, sapma hatasını bulmaktır. Tahmin edilen değerin üzerindeki bulunan değerlerden (Y) tahmin edilen değer $\widehat{Yı}$ çıkarıldığında pozitif bir sonuç bulunurken tahmin edilen değerin altından bulunan değer(Y) ile tahmin değeri $\widehat{Yı}$ arasındaki fark ise negatif bir sonuç bulunmaktadır, toplanan bu değerlerin (negatif ve pozitif) birbirini götürmemesi için kareleri alınır her farkın karesi alındıktan sonra toplanılır ve nokta sayısına (n) bölünür böylelikle MSE bulunmuş olur.

2.3.2.1.2. R-kare

Sonucun gözlemlenerek bilinen değerleri ile model tarafından ön görülen değerler arasında kare korelasyonu ifade eder. R2 ne kadar yüksek olursa model o kadar başarılı olduğu anlamına gelir (Kassam 2018).

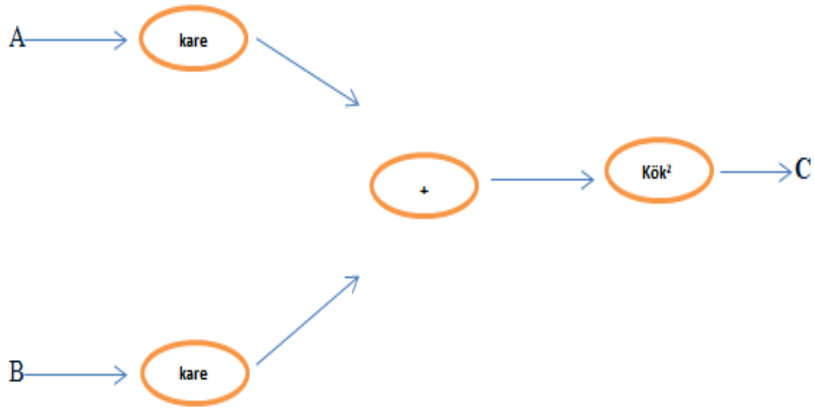
2.3.2.2. Lojistik Regresyon

Lojistik regresyon literatürde regresyon olarak geçse de temelde sınıflandırma problemlerini çözmek için kullanılan bir denetimli öğrenme algoritmasıdır. En az değişkeni kullanarak en iyi uyuma sahip olacak şekilde bağımlı ile bağımsız değişkenler arasındaki ilişkiyi tanımlayabilen kabul edilebilir bir model kurmaktır (Bircan 2004). Kullanılabilirlik ve bilinirlik açısından doğrusal regresyondan sonra gelmektedir. Pek tabii olarak linear(doğrusal) regresyondan farklı çalışmaktadır. Tahmin sonuçlarının genellikle **ikili sınıflandırma** olarak verdiği bir algoritma türüdür. Sonuç olarak üretilen etiketlerin binary classification (0-1, evet- hayır, doğru – yanlış, spam – spam değil) değil gibi etiketlerin üretildiği bir sınıflandırma yöntemidir. İkili sınıflandırmanın yanında pek az da olsa **multinominal** (çok terimli) olarak kullanılabilir bağımlı değişken sayısı eğer 3 ve üzerinde ise buna örnek olarak "A tipi" veya "B tipi" ya da "C tipi" olarak da kullanılabilir. Buna mukabil, **sıralı** şeklinde de kullanılabilir değişken etiketleri; "iyi" "kötü" "çok iyi" veya "mükemmel" gibi ardarda takip eden elde edilmek istenen sonuçların bir devamı olarak kabul edilir. Matematiksel olarak lojistik regresyon modeli X'in bir fonksiyonu olarak P(Y=1)'i tahmin eder (TutorialsPoint 2019). Olasılık kavramları üzerinde çalışan ve bir tahmine dayalı analiz algoritmasıdır. Lojistik regresyona daha derinlemesine geçmeden önce hesaplamalı grafiği ne olduğunu bilmekte fayda olacaktır.

2.3.2.2.1. Hesaplamalı Grafik

Hesaplamalı grafik geriye yayılmalı makine öğrenmesi algoritmalarında çok kullanılan bir yöntemdir. Lojistik regresyonu daha iyi anlamak için matematiksel ifadeleri ve modelleri görselleştirmek adına anlamaya ve uygulamaya yardımcı olacaktır. Örnek olarak:

$$c = \sqrt{a^2 + b^2}$$



Şekil 19. Matematiksel İfadelerin Görselleştirilmesi

Yukarıda ki denklemi ifade eden görselde a ve b'nin karesini aldıktan sonra a ve be 'i önce toplamını ardından kare kökünü alıp C 'ye eşit olduğunu ifade etmektedir.

Lojistik regresyon algoritması bir tahminde bulunmak için bağımsız tahmin edici yöntemler içeren doğrusal denklemler kullanmaktadır. Öngörülen değer negatif sonsuz ile pozitif sonsuz değerler arasından yer almaktadır. Algoritmanın uygulandığı verilerden elde edilen çıktının sınıf değişkeni ya da sayısal etiket (0-hayır, 1- evet) olması gerekmektedir. Bu doğrultuda, doğrusal denklemin 0 ila 1 arasında olmasını istenir. Tahmin etmek istenilen değeri 0-1 arasında görmek için sigmoid işlevini kullanarak bir sıkıştırma meydana getirilir (Kızrak 2019).

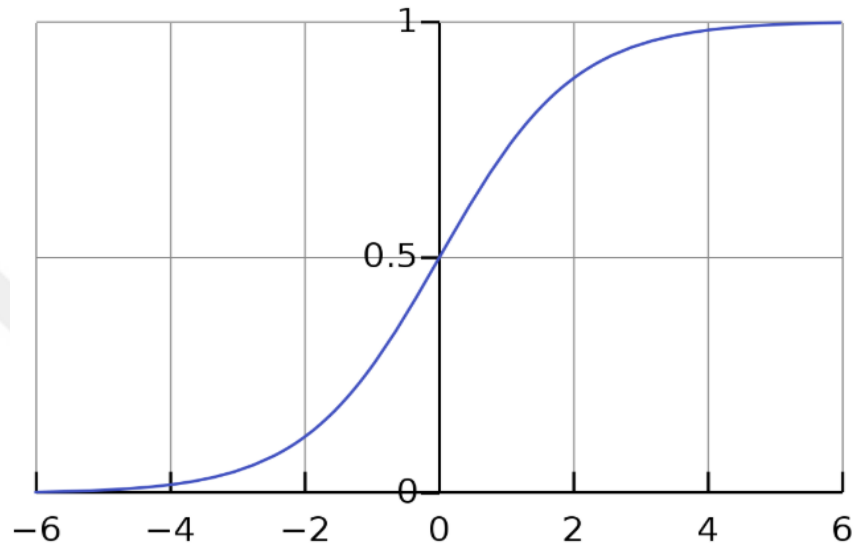
$$Z = 00 + 01.x1 + 02.X2 + \ldots$$

$$\boldsymbol{f\,(x)\; =\; (1\;/\;1 + e\text{^}(-z))}$$

$$\boldsymbol{h = \; f(z) = \; (1\;/\;1 + e\text{^}(-z))}$$

- $f(x) = 0$ ile1 arasında bir tahmin değeri
- x = işleve girdi
- e = doğrusal algoritmanın tabanı

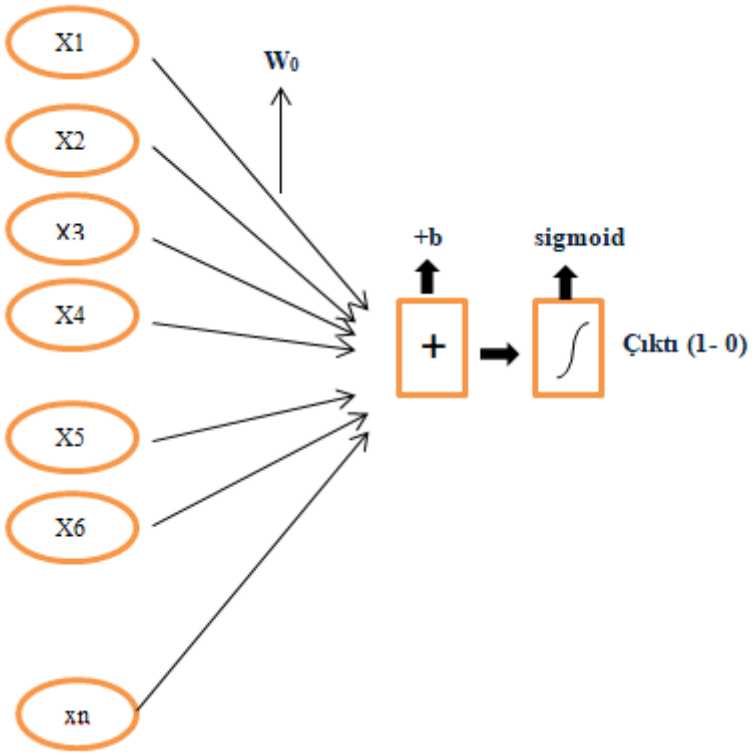
Verilerden elde edilmek istenen doğrusal denklem çıktısı olarak (z) almış h sıkıştırılmış bir ifade döndüren g(x) fonksiyonuna dahil edildiğinde h değerinin 0 ile 1 arasında olması beklenilir. Sigmoid fonksiyonun aldığı değerleri sonsuz negatif ile sonsuz pozitif değerler arasında nasıl bir sıkıştırma meydana getirdiği anlamak için sigmoid grafiği;



Şekil 20. Sigmoid Fonksiyonu

Veriyi kullanmak ve istenen sonuçları tahmin etmek gibi durumların söz konusu olduğu bir analiz veya çalışmada sigmoid grafiğini nasıl kullanıldığını bilinmelidir. Örnek olarak yüzlerce haber metnin bulunduğunu varsayalım ve ikili etiketleme sonucunda haberin spor ya da siyaset olarak verecek bir model olduğunu ve bu doğrultuda işlemlerin yapıldığını varsayarak ;

Spor=1
Siyaset=0



Şekil 21. Verinin Modellemesi

Veri işleme sırasında istenilen sonuç ve tahminlere yaklaşmak için veride karşılaşması muhtemel bazı problemleri ortadan kaldırmak adına parametre ekleme veya eklenilen parametreler de oynama söz konusu olabilir. W_0 ve B birer parametredir. Haber metni örneği yukarıdaki şablona koyulduğunda her bir metin bir $x_1, x_2, x_3 \ldots x_n$ değeri olarak geçmektedir her X değeri bir W ile çarpılıp ardından B ile toplanmaktadır. $Z = b + x1w1 + x2w2 + x3w3 \ldots\ldots xnwn$ Olarak işlenilir. Sonraki adım olan ve yukarıda bahsi gecen sigmoid'e girer değerimizi 0 veya 1 olarak çıkmasını istenilmektedir. Sigmoid türevi alınabilen bir fonksiyondur (Kaan 2018). Bunun veri için önemli tarafı ise türevi alınabilen değerlerde parametrelerin güncellenebiliyor olmasıdır. Doğrusal regresyon da veriye bağlı olarak geliştirilen modelde $y= a_0+a_1*x$ fonksiyonu göz önünde tutularak a_0 ve a_1'lerin bilinmesi önemlidir. Çünkü modelleyebilmek için a_0 ve a_1 değerinin bilinmesi gerekir. Burada bilinmesi gereken önemli hususlardan biri de modelleme yapabilmek için W ve B'lerin değerinin ne olduğunun bilinmesi ve başlangıç değerlerin ne olduğu buna göre bir model ortaya koyulmalıdır.

2.3.2.2.2. Kayıp (Loss) Fonksiyonu

Kayıp fonksiyonu; eğer haber veri seti içerisinde spor haberi değer olarak girildiyse, sonucunda tahmin edilen değer 1 yani spor haberi olması burada kayıp fonksiyonun 0 çıkması anlamına gelmektedir. Bunun aksi durumu olması halinde ise yani haber olarak spor girilmesi durumunda fakat oluşturulan model çıktı olarak siyaset vermesi burada tahmin edilen değer 1 olması gerekirken 0 vermesi kayıp fonksiyonu yüksek olduğu anlamına gelmektedir (Kaan 2019).

Kayıp fonksiyonu = $-(1 - y)\log(1 - \hat{y}) + y\log\hat{y}$

Her verinin bu fonksiyondan geçtikten sonra toplam kayıp fonksiyonu alınır ve bir hata maliyeti çıkarılır. Buna bağlı olarak toplam hata maliyeti yüksek ise bu modelin iyi olmadığı anlamına gelir ya da beklenilenin altında bir hata maliyeti çıkması durumunda elde edilen değerlere ilişkin geliştirilen model işlevsel olarak iyi bir modeldir. Hata maliyetini düşürmek istenildiğinde yapılması gereken W ve B'lerin güncellenmesidir. Başlangıç değerinden farklı olarak sürekli farklı değerler verilmesi modelin şekillenmesine ve buna göre hata maliyetini düşürmesine yardımcı olacaktır. Hata maliyetini azaltmak için denklem:

$$w := w - a\frac{\partial j(w,b)}{\partial(w,b)}$$

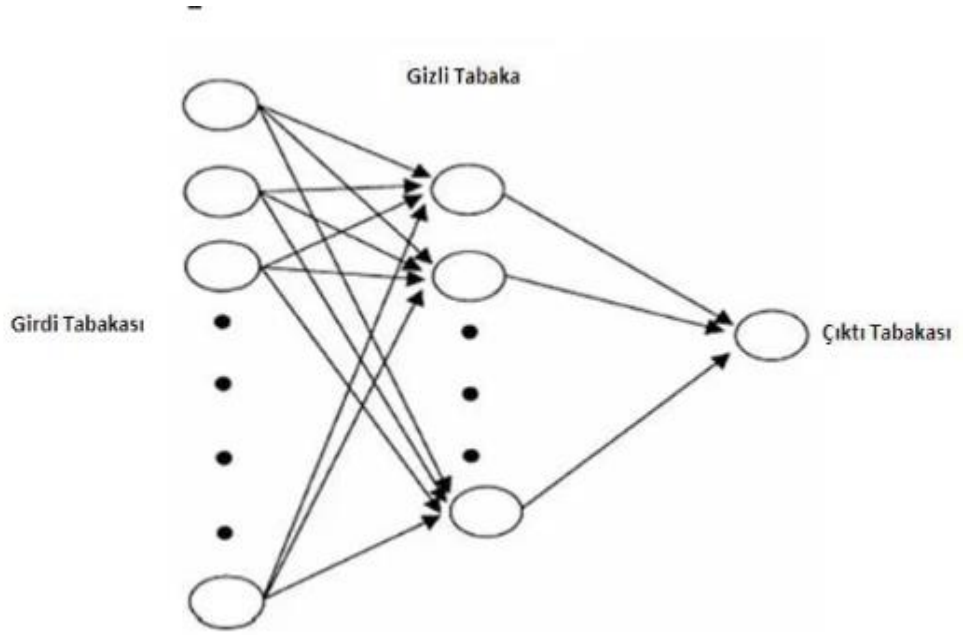
J maliyet fonksiyonunu göstermektedir. Denklem maliyet fonksiyonunun ağırlığa göre türevini alır.

2.3.2.3. Yapay Sinir Ağları

Yapay sinir ağları (YSA), insan beyin ve sinir sisteminin çalışma şeklinden yola çıkarak oluşturulan, girdilere dinamik şekilde yanıt verme yoluyla bilgiyi işleyen ve bağlantılarını ortaya çıkartan hiyerarşik bir organizasyondur (Işık 2018:22-23). İnsan beyninden sonra eğitilmesi halinde kendini modelleyebilen bir makine öğrenmesi yöntemidir. Bir algoritma ile mevcut bir makinaya yeni veriler ekleyerek öğrenmesini ve kendini geliştirmesini sağlayan sinir bir ağ oluşturur. Bilindiği üzere beynin temel birimi görüntü işleme sinyal işleme söylenenleri ve sesleri tanımaya

yardımcı olan pek çok uygulama için temel oluşturma mahiyetindedir. Bu işlemleri yapabilecek pek çok algoritma olmasına rağmen sinir ağlarına entegre edilmesi halinde öğrenme ve öğrendiklerini uygulama noktasında algoritmalara nazaran daha başarılı olmaktadır.

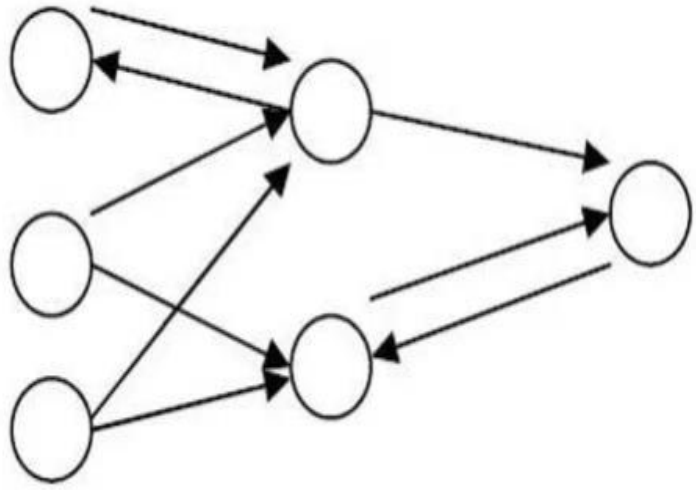
Yapay sinir ağını kullanan bir bilgisayar daha önceden tanımlanmış olan çıktı etiketlerinin eğitim örneklerini analiz ederek kendisinden beklenen görevi yapmayı öğrenir. Bir sinir ağının derin öğrenmeyi kullanarak gerçekleştirmesini beklendiği basit örneklerin başında nesneyi tanıyabilme görevidir. Sinir ağlarının daha önceden kendisine tanımlanan araba, ağaç, insan gibi nesneleri birbirinden ayırması nesne tanımlamasına örnek verilebilir. Yapay sinir ağları kendisine sunulan görüntülerde aynı nesnelerin ard arda gelmesi düzenli ve yapısal olacak şekilde analizinin yaparak yeni görüntüleri kategorize etmeyi öğrenir ve bu etiketlemenin sonucunda kendisine önceden tanımlanan bir nesneye benzer bir nesnenin gösterilmesi durumunda kayıt edilmiş görüntüler ile karşılaştırması sonucunda bir cevap vermesi beklenir. Sinir ağını bir bebeğin beyin yapısına benzetmekte fayda vardır çocuğun gelişimine bağlı olarak bilgiyi öğrenme öğrenilen bilgiler arasında doğrudan veya dolaylı olarak ilişki kurma gibi fonksiyonlara benzer bir şekilde sinir ağı da aynı süreçleri yürütmektedir. Matematiksel olarak ifade etmek gerekirse yapay sinir ağları (YSA)



Şekil 22. Yapay Sinir Ağı Örnek Modeli

Şekil de nümerik değişkenleri şekil içerisinde giriş verisi olarak alıp modeller ve belirlenen fonksiyonlardan geçirdikten sonra nümerik çıktılar üreten çok yönlü bir ağ yapısı haline dönüşür. (Enriquez vd. 2016) aktivasyon fonksiyonu olarak da adlandırılan bu fonksiyon nöronlar arasındaki bağlantıların ağırlıklarını aktivasyon fonksiyon ile tekrar edilmesi ile yapay sinir ağı öğrenme modelleme işlemini gerçekleştirir. Yapısal olarak oluşturulan yapay sinir ağları kullanım amacına bağlı olarak tek katmanlı çok katmanlı olarak iki farklı şekilde kullanılır.

Tek katmanlı ağ, Bir adet girişi ve bir adet çıkıştan meydana gelmektedir. Çok katmanlı ağ da ise bir adet giriş katmanı en az bir adet gizli katman ve en az bir adet çıktı katmanından oluşur. Yapay sinir ağlarından pek çok eğitme modelleri vardır bunlardan en sık kullanılanlardan bir tanesi ise geri beslemeli modeldir. Standart geri beslemeli algoritması, ağ ağırlıklarının, performans fonksiyonunun negatif gradyanı yönünde ilerlediği gradyan iniş algoritmasıdır (Hamzaçebi vd. 2004). Geri beslemeli modelde giriş verileri bir ağ ile eğitilirken ağdan gecen veriler çıktı olarak gelen veriler, beklenilen veriler ile karşılaştırılır. Çıktı değerleri ile beklenilen değer arasında oluşan farka hata denilir. Ağ çıktısında meydana gelen bu hataya tekrar ağ üzerinden çıktıdan geriye doğru yayılarak nöronlar arasındaki ağırlıklarını değiştirilecek şekilde yeniden hesaplama yapılması beklenir. Burada yapılmak istenen arada oluşan hatayı yok etmek ya da en düşük seviye de tutmaktır. şekil geri beslemeli modele örnektir.



Şekil 23. Yapay Sinir Ağları Geri Beslemeli Model

2.3.2.4. Naive Bayes

Olasılıklara dayanan bir denetimli makine öğrenmesi algoritmasıdır. Öngörüler arasında bağımsız bir varsayıma dayanan bir sınıflandırma yöntemidir. Bir sınıfa ait belirli özelliklerin varlığı diğer herhangi bir özelliğin varlığından bağımsız olduğunu varsayar. Naive bayes etkili olmasının yanında oldukça hassastır. Bu nedenle özellik değerlendirme de metriklerin çalışması oldukça gereklidir (Houkuan vd. 2009). Bir meyve kırımızı, yuvarlak ve çapı 3 inç ise elma şeftali kabul edilebilir. Bu sayılan tüm özellikler birbirlerine veya diğer özelliklerin varlığına bağlı olsa bile bu özeliklerin tümü bağımsız bir şekilde bu kırmızı meyvenin bir şeftali olma olasılığına dahil olur. Naive bayes teoreminin en önemli özeliklerinden bir tanesi hızlı ve kolay eğitilebilir olmasıdır. Naive bayes veri üzerinde modellemek için uygulandığında parametre tahmini için en üst düzey olasılık tahmini yöntemini kullanır. Bu veri herhangi bir olasılığı kabul etmeden veya herhangi bir bayes yöntemine başvurmadan naive bayes yöntemiyle çalışabileceğin göstergesidir.

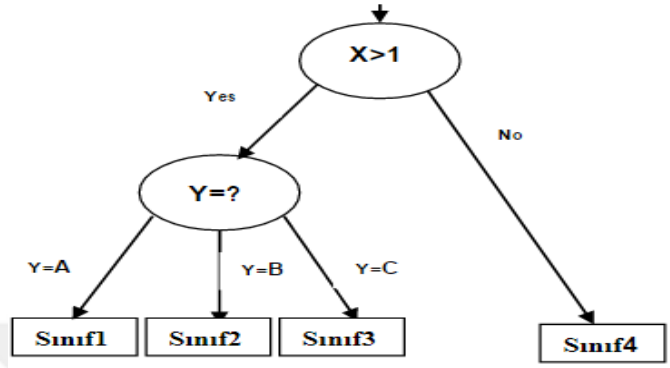
$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

- "P" olasılığı temsil eder.
- P (A | B) = A olayının (hipotez) olayı, B'nin (kanıtların) meydana geldiği göz önüne Alındığın da ortaya çıkma olasılığı.
- P (B | A) = A olayı (hip.) oluştuğu için gerçekleşen B olayının (delil) olasılığı.
- P (A) = B olayının (hip.) oluşma olasılığı.
- P (B) = A olayının (kanıt) ortaya çıkma olasılığı.

2.3.2.5. Karar Ağaçları Algoritması

Karar ağacı algoritması, veri madenciliği sınıflandırma algoritmalarından biridir ve bilgi teorisi ilkelerine dayanmaktadır (Alan 2014). Karar ağaçları algoritması ele alınan verinin modellenerek hangi sınıfa ait olduğunu belirlemek için daha önce işlediği verileri kullanarak sınıflandırma yapan denetim öğrenme algoritmalarından biridir. Daha önceden tanımlanmış bir hedef değişkine sahip olmasının yanında yapısı itibariyle tüm verilere uygulanabilecek bir algoritmadır. Karar ağacı algoritmasının

çalışma mantığında çok büyük veri gruplarını veya kümelerini belirlediği kuralları uygulayarak daha küçük kümelere bölmek için kullanılan bir yapıya sahiptir (Irmak vd. 2017).

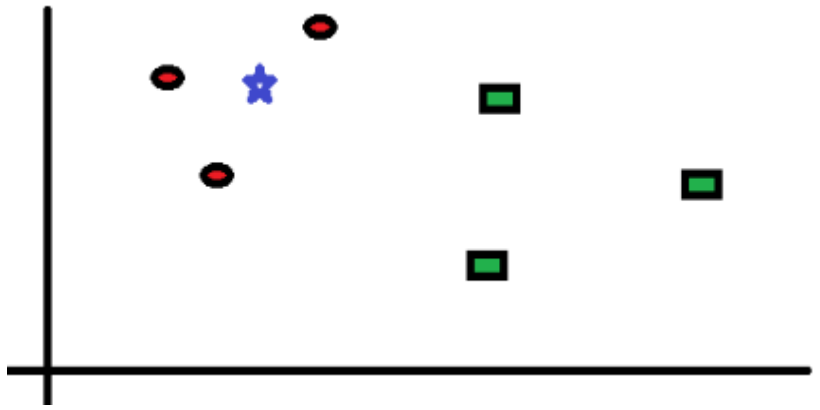


Şekil 24. Karar Ağaçları Modeli

En üsteki düğüm karar düğümü olarak bilinir. Öznitelik değeri kökte bölümlenmeyi öğrenir. Ağacı tekrarlı olarak bölümlendirir. Her dalın bir karar kuralını temsil ettiği gibi her bir yaprak düğümü ise sonucu temsil etmektedir. Bu şekilde akış şeması karar verme noktasında yardımcı olur. Farklı şekillerde görselleştirildiği için karar ağaçlarının anlaşılması ve yorumlanması kolaydır. Karar ağaç yapısının amacı veri setini ile en az düğüm oluşturarak en yüksek başarıyı sağlamaktır (Işık 2018:23-24).

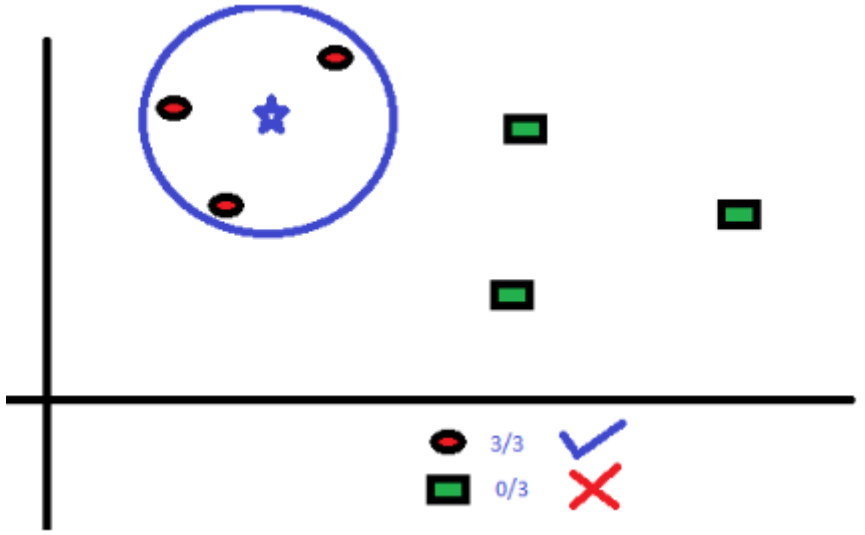
2.3.2.6. KNN Algoritması

Yeni veri noktalarının değerlerini tahmin etmek için kullanılır. KNN algoritması seçilen bir çeşit özelliğin kendine en yakın olan özellikle arasındaki yakınlığı kullanarak sınıflandırma yapılır. Veri noktalarının değerlerini benzerlik özelliğini kullanarak tahmin eder. Bunun yanında, yeni veri noktalarının eğitim setinde bulunan veri noktalara ne kadar yakın bulunduğuna bakarak bir değer ataması yapar (Kılınç vd. 2016).



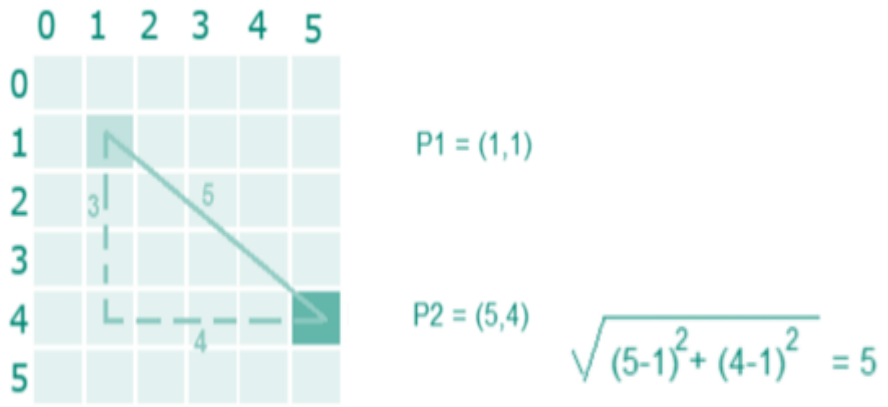
Şekil 25. KNN Algortiması Veri Dağılımı

K değeri komşu sayısını ifade etmektedir. Kırmızı ve yeşil olarak iki sınıftan oluşan bir verinin olduğunu varsayalım ve mavi ile boyanmış yıldızın hangi sınıfa ait olduğuna aşağıda görülmektedir; KNN algoritması ile K ya bir değer verilir. Verilen değer *K*'nın alan olarak kapsadığı veri noktasının sayısıdır K=3 olarak kabul edilen;



Şekil 26. KNN Algortiması K Değer Seçimi

Mavi yıldızın etrafında 3 adet noktayı kendi içerisine almaktadır. KNN algoritması için K parametresinin seçtiği değerin uzaklığı önemi büyüktür. K etrafından bulunan yerin uzaklık hesaplamasını 3 farklı yol ile yapmaktadır. Öklid, manhattan ve minkowski bu hesaplama türleri arasından en çok bilinen ve uygulanan Öklid hesaplama yöntemidir.



Şekil 27. KNN Algortiması K Değer Uzaklık Hesaplaması

$$d(p,q) = d(q,p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

2.4. Çok Etiketli Sınıflandırma

Çok etiketli sınıflandırma analiz sürecinde çok başvurulan bir sınıflandırma türüdür. Çok etiketli sınıflandırma yönteminde problem dönüştürme yöntemleri kullanılarak önce tek etiketliye dönüştürülür ardından sınıflandırılır. Tek etiketli sınıfa dönüştürme işlemi olarak bilinen problem dönüştürme yöntemleri aşağıda ki bölümlerde anlatılacaktır.

2.4.1. Tek Etiketli Sınıflandırma

Tek etiketli sınıflandırma (single-label), var olan m sınıftan her bir metin için ilgili olduğu belirlenen sadece bir tanesinin seçildiği ve belirlenen sınıfa ait etiketin metne "yapıştırıldığı" ya da "atandığı" uygulamadır. Tek etiketli çok sınıflı sınıflandırma olarak da bilinen geleneksel tek etiketli sınıflandırmada, her veri noktası yalnızca bir kategoriye aittir (Ding vd. 2010). Tek etiketli veya geleneksel sınıflandırma olarak da bilinen sınıflandırma yöntemi temelde iki problemli kategoriye ayrılır. İkili (binary) sınıf problemi ve çoklu sınıf(multi-class) problemi bu iki

kategoriden biri olan ikili sınıf problemi bir durumun olup-olmadığı, olumlu-olumsuz gibi peki çok zıt durumu ifade eden ve matematiksel karşılığı (+1) veya (-1) olarak ifade edilir. Örnek olarak; doğan bir bebek ya erkektir ya da kız, bir kadın hamile olup olmadığını öğrenmek için doktora gittiğinde test sonucu ya hamiledir ya da değildir. Tek etiketli sınıflandırmanın diğer problemi olan çoklu sınıf problemi ise hedef sınıfların tamamen birbirinden bağımsız kendine özel olduğu varsayılmakta ve her elemanın tamamen bir sınıfa ait olduğu kabul edilmektedir.

Çok sınıflı sınıflandırma pek çok alanda karşılaşılması mümkündür; Örneğin, bir bireyin kan grubu dört ayrı grup da sınıflandırılır. A, B, AB ve 0 ve bir hasta sadece bir kan grubuna sahiptir. Bu dört kan grubundan birine ait olabilir ancak her ikisine veya ikiden fazla kan grubuna sahip olamaz. Çoklu sınıflandırma problemlerini çözmek için mevcut ikili sınıflandırıcıları çoğaltabilen pek çok algoritma mevcuttur. İleride de aralarından kullanacağımız olan birkaç algoritmaya örnek; karar ağaçları, k en yakın komşular (k-nearest neighbors), naif bayes (naive Bayes), destek vektör makinaları gibi pek çok teknik geliştirilmiştir.

2.4.2. Çok Etiketli Sınıflandırma

Çok etiketli sınıflandırma, her bir durumun veya gözlemin sadece bir durum ve gözlem yerine aynı anda birden fazla sınıfa atabileceği gerçek dünyadaki verilerin üzerinde çalışmalar yapılabildiği ve uygulama alanlarında bulunan bir sınıflandırma sorunudur. Çok etiketli sınıflandırma da örneğin kaç tane sınıfa atanacağı konusunda herhangi bir kısıtlama yoktur. Uygun yöntemler kullanarak pek çok alanda farklı çözüm getirilebilir örneğin e-postalar, bloglar, RSS yayınları sosyal medya gibi devasa verilerin gerçek zamanlı olarak doğru ve nitelikli analizler yapılmasına olanak sağlayabilir. Çok etiketli sınıflandırma problemi gerçek veriler üzerinde pek çok noktada sorun olarak karşımıza çıkmaktadır. Metinlerin kategorize edildiği, makale, web sitesi sınıflandırma, müzik türüne veya çıkarılacak sese bağlı olarak kategorize edildiği, biyoenformatik gibi pek çok alanda karşılaşılmaktadır. Özellikle metin belgeleri genellikle birden fazla kavramsal sınıfa ait olması örneğin; Yeni Zelanda devlet başkanının camii saldırısı sonrasında yaptığı açıklamada haber sayfalarında çok etiketli sınıflandırmasına örnek teşkil edebilir. Atanan etiketlerin siyaset, terör, din,

olduğu gibi, bir resimde birden fazla nesnenin olması bir müzikte birden fazla farklı sesin kullanılması gibi örnek sayısı artırılabilir.

Şekil 28. Çok Etiketli Sınıflandırma

Sınıflandırmada bir den fazla etiket atılmasına örnek olarak Marmara Erdek sahilinin hem kıyı hem de orman gibi iki etiketten oluşan bir sınıf olacağı gibi şehir, kıyı ve ormandan oluşan bir sınıf da olabilir. Etiketler arasındaki gizli korelasyonlar bir sınıfın etiket kümesini tahmin etmede yardımcı olabilir örneğin, yüksek tansiyona sahip bir hastanın kalp hastalıklarından birine yakalanma olasılığı yüksek iken aynı hastanın kas hastalıklarından birine yakalanma olasılığı daha düşüktür. Bu doğrultuda, çok etiketli sınıflandırma sosyal hayatın her karesinde karşımıza çıkması muhtemeldir. Çok etiketli sınıflandırmayı yapabilmek için pek çok alternatif algoritma mevcut iken en sık kullanılan algoritmalar, SVM, karar ağaçları, KNN'dir.

2.4.3. Çok Etiketli Öğrenme Yaklaşımı

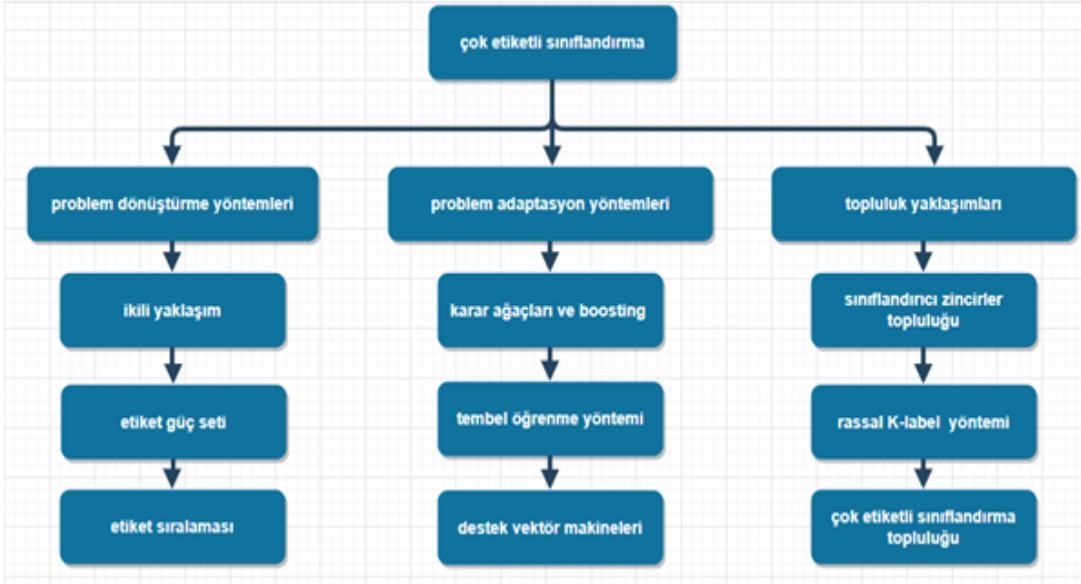
Çok sınıflı sınıflandırmada karşılaşıldığı gibi sadece bir gözlem ele alacak şekilde her gözlem için birden fazla etiket atanabileceği bir sınıflandırma sorunudur (Schratz 2018). Makine öğrenmesi çalışma tekniğinde mevcut verileri bölümlendirdikten sonra eğitim örnekleri olarak ayrılan ve üzerinde herhangi bir çalışma yapılmadan önce bilinen bir dizi örnek kullanarak bir sonuca ulaşılabilir. Veri üzerinden işleme tabi tutulduktan sonra bir modelin ortaya çıkması ve bu modellerin pek çok veri ve uygulamada kullanılabilir olması gerektiği önceki bölümlerde belirtilmiştir. Modelin veriye uygulanabildiği ve veri ile ilişkilendirme, kümeleme, optimizasyon, sınıflandırma ve regresyon gibi görevleri yerine getirebilir.

Çok etiketli sınıflandırma da $\{ xi, yi \}$ birer $f(x)$ fonksiyondur. xi ile yi arasında sınıfsal bir ilişki vardır. İ $= 1,2, \ldots \ldots, n$ her n değeri birer eğitim örneğidir. Veriyi bir kere eğitime tabi tuttuktan sonra modelin tahmin etme fonksiyonu kesinlik kazanır. Sınıflandırma problemlerinde belirtilen sınıf sayısına göre sınıflandırılabileceği gibi eğitim sırasında birer örnek olacak şekilde etiket ataması yapılması halinde belirtilen etiketlere göre sınıflandırma meydana gelir. Örneğin, kullanılabilir çok sayıda belge olduğunu varsayalım ve her belgenin sadece bir dilde yazılabileceğini göz önünde bulundurarak mevcut yazılacak dillerin ise sadece Türkçe, İngilizce, İspanyolca ve Çince olduğunu varsayalım. Her belge sadece bir sınıfta sınıflandırılacaktır. Her belgenin giriş örneği sadece olası sınıflardan bir tanesine atanır. Makine öğrenmesinde sınıflandırma problemlerinin pek çoğu tek etiketli sınıflandırmadır. Çok etiketli sınıflandırma probleminde meydana gelen sınıflandırma sorunu bir örnek veya öznitelik birden fazla sınıfa ait olabilir. Çok etiketli öğrenme problemi literatürde en sık metin kategorizasyon da karşılaşılmaktadır. Eğer bir problemin N sınıfa sahip olduğunu biliniyorsa p_i, $1 \leq$ i $\leq$ N, olmalıdır. Ve her sınıf temel olarak $0 \leq$ pi ≤ 1 arasında yer alır. Tek etiketli sınıflandırma içi bir kısıtlama meydana gelirken $\sum pi = 1$, çok etiketli sınıflandırma kısıtmala kabul etmez. Çok etiketli yöntemleri atama yapabilmek için pek çok algoritma kullanılabilir. Çok etiketli bir sınıflandırma problemi genel itibariyle mevcut çok etiketli bir problemi tek etiketli bir probleme dönüştürerek ele alır. Bu dönüştürme sağlanırken verinin niteliğine veya etiketin niteliğine bakılarak dönüştürme yapılır (Monard vd. 2011).

Literatürde çok etiketli sınınıflandırma ile ilgili problemleri çözmek için pek çok sayıda yöntem geliştirilmiştir. Kullanılan algoritmalar 3 farklı kategoride toplanmıştır bunlar: problem dönüştürme yaklaşımları,problem adaptasyon yöntemleri ve topluluk yaklaşımları (Tsoumakas 2007).

İlk kategoride çok etiketli bir sınıflandırma problemini bir veya daha fazla gruba bölerek problemi geleneksen yöntemler kullanarak çözmeye çalışır. ikinci kategoride doğrudan çok etiketli veriler ile başa çıkmak için tek etiketli algoritmaları genelleştirir. Son olarak ,üçüncü kategoride yukarıda bahsedilen iki yöntemin temelleştirmesi durumudur. Aşağıda ki başlıklarda bu bahsettiğimiz üç kategorinin

detaylı bir şekilde anlatacağız ve bu yöntemlere karşılık gelen metodların. Aşağıdaki şekil 29 da bu yöntemlerin alt başlıkları ve ait oldukları sınıfları göstermektedir.



Şekil 29. Çok Etiketli Sınıflandırma Algortimalarının Kategorizasyonu

2.4.4. Problem Dönüştürme Yöntemleri

Problem dönüştürme yöntemleri mevcut çok etiketli problemi tek bir etiketli probleme dönüştürme görevidir. Sınıflandırmanın olduğu pek çok yerde etiket atamasının gerçekleştiği bilinmekle beraber etiketlemenin iç içe girmiş verilerin analizini kolay bir şekilde yapılmasını sağlar. Problem dönüşümünün sağladığı önemli faktörlerden biri tek etiketliye dönüşen problemin sınıflandırma özelliklerinden soyutlanabilmesi ve modellenebilen etiketlerin oluşturmasıdır. Ve korelasyonlara çok etiketli alanlar ile ilgili konulara odaklanmasını sağlar ve daha genel bir uygulanabilir şartlar oluşturur. Örnek, gün rutin çıkan bir gazetenin haberlerini problem dönüştürme yöntemlerinden faydalanarak tek etiketliye dönüştürme yapılırsa her çıkan haberin bir veri olduğunu ve verinin içerisinde etiket ataması belirlendikten sonra haberin neyi gündeme aldığı anlaşılabilir ve Spor, siyaset, ekonomi gibi etiket ataması gerçekleşebilir. Bir haber sadece bir etikete atanması gibi bir durum söz konusu olamamaktadır. Futbol sahasında bir şiddet olayı yaşanıyor ve ölümler gerçekleşiyorsa bu durumda iki etiket ataması olacaktır bunlar spor ve şiddet.

2.4.4.1. İkili Alaka Düzeyi

Her etiketi bölümlendirerek tek bir sınıflandırma problemi olarak işleyebilen en basit tekniktir. Her q çok etiketli öğrenme problemi olarak ikili sınıflayıcıları eğiterek çözmeyi hedefler ve q tek etiketli sınıflara ayırarak mevut olan q sınıflandırması orijinal veri kümesi üzerinde eğitilerek ele alınan etiket için kendi alaka düzeyini belirler. BR yöntemi çok etiketli problemleri ele almak için genelleştirilmiştir. BR yönteminin kullanılması hem kolay hem de olası karmaşıklık sayısı, etiketlerin sayısı ile orantılıdır. Her etiketi bağımsız olarak ele alan BR yöntemi her label(etiket) için (L) bir veri setin de toplam sınıf sayısını belirtir buna bağlı olarak q=|L| yani orijinal veri seti olan L'yi q'a dönüştürür. İkili ilişkilendirme de kullanılan yöntem etiketler L sayısınca parçalara bölünerek işlenmektedir (Jain 2017).

A	Y_1	Y_2	Y_3	Y_4	Y_5
X_1	1	0	1	0	1
X_2	1	1	0	1	1
X_3	0	0	0	0	0
X_4	1	0	1	1	1
X_5	1	1	1	1	1

Şekil 30. İkili Alaka Düzeyi (BR) Yaklaşımı

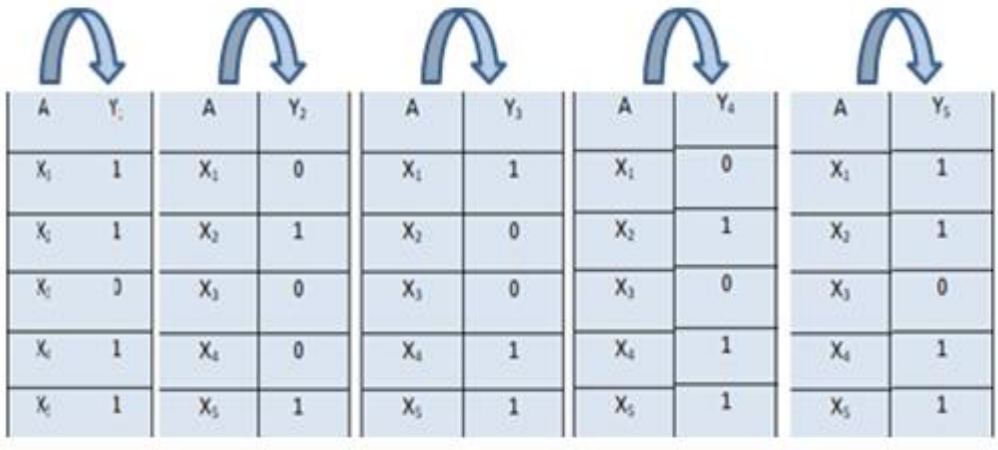
$$H_{BR} = \{ C_{yi}\left((x, yi)\right) => yj \ \in \{0,1\} | y_i \ \in L : j = 1 \ldots . q\}$$

$$L = \{ y_1, y_2, \ldots \ldots \ldots, y_q \}$$

$$D_{yi}, j = (1, \ldots \ldots \ldots, q\ )$$

$Dyi =$ Gerçek çok etiketli tüm örneklerini içeriyor.

$H_j(E), j = 1, \ldots \ldots, q\ ,\ \ Dyi$ eğitim verisini kullanarak oluşturulmuştur. X'in bağımsız birer özellikken Y birer hedef değişkenidir. İkili alaka düzeyi aşağıdaki şekil de görüldüğü gibi 5 farklı tek sınıf sınıflandırma problemine dönüştürülür.



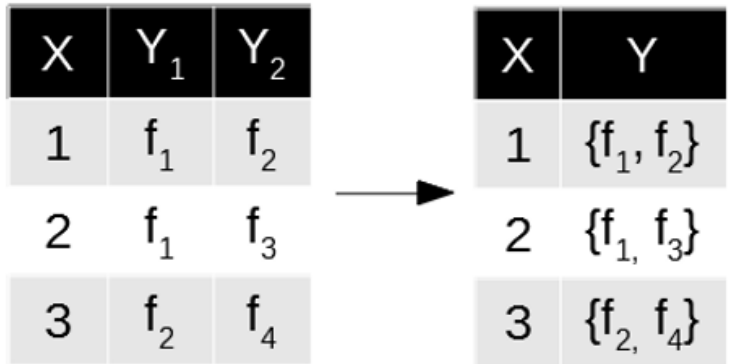
Şekil 31. İkili Alaka Düzeyinin Tek Sınıflı Sınıflandırmaya Dönüşmesi

Her veri kümesi için bir oluşturulan ve yeniden sınıflandırılacak ikili alaka düzeyi çıktıları için q tarafından pozitif olarak etiketlerin birleştirilerek oluşturulmasıdır.

2.4.4.2. Etiket güç seti (LP)

N sayıda örnekleri barındıran bir D veri kümesinde bulunan her benzersiz etiket kümesi için tek bir etiket olarak kabul eder ve ardından tek bir etiket gibi işler. Veri setindeki sınıf sayısına sınır getirilmiştir. Etiket gücü yönteminde karmaşıklık sayısı etiket sınıflandırıcının karmaşıklığına bağlıdır (DeutSchman 2019).

Etiket birleştirme gücü olarak da bilinen (Labelcombination method) Etiket gücü eğitim verisinde bulunan tüm benzersiz etiket kombinasyonları göz önünde bulundurarak birbirine benzer yapıda olan etiketleri bir grupta toplayıp yeni bir etiket sınıfı oluşturmaktır. Kullanarak Dönüşüm ve LP'den Sıralama elde etme örneği;



Şekil 32. Etiket Güç Seti

Bu yöntemin eksik olarak kabul edilen tarafı ise çok sayıda sınıfa bağlı dengeli olamayan veri kümelerine yol açabilmesidir (Everton vd. 2011). Yani Etiket gücünde birbirine benzer çok etiketin olması durumunda öğrenme sürecini zorlaştırır ve sınıf dengesizlik problemini ortaya çıkarır. Bu probleme çözüm ise sadece etiket kombinasyonları dikkate alınarak giderilebilir. Veri setindeki tek etiketli sınıflandırıcı bir sınıf değeri olarak kabul edilir.

2.4.4.3. Etiket Sıralaması

Etiket sıralaması yöntemi için de ki olası tüm etiket örneklerinden bir eşleşme öğrenir, alaka düzeyini göz önünde bulundurarak tek etiketli sınıflandırma bu örneklem için en uygun etiketi seçer. Bir değer atama fonksiyonu olarak kabul edilen etiket sıralaması $f : X\ x\ Y\ \rightarrow R$ modülüne bakarak bir puan ya da değer ataması yapar (Sawsan 2017:18). Her etiket ataması kendinden bir önceki atamaya puan ekleyerek devam eder. İşleyiş biçimleri ikili alaka düzeyi ile oldukça birbirine yakın olmasına rağmen sıralama mantığı ayırt edici özelliktir. Öğrenme sürecinde verileri daha iyi işleme ve kalıtsal özelliğini koruması en büyük avantajıdır.

2.4.5. Problem Adaptasyon Algoritmaları

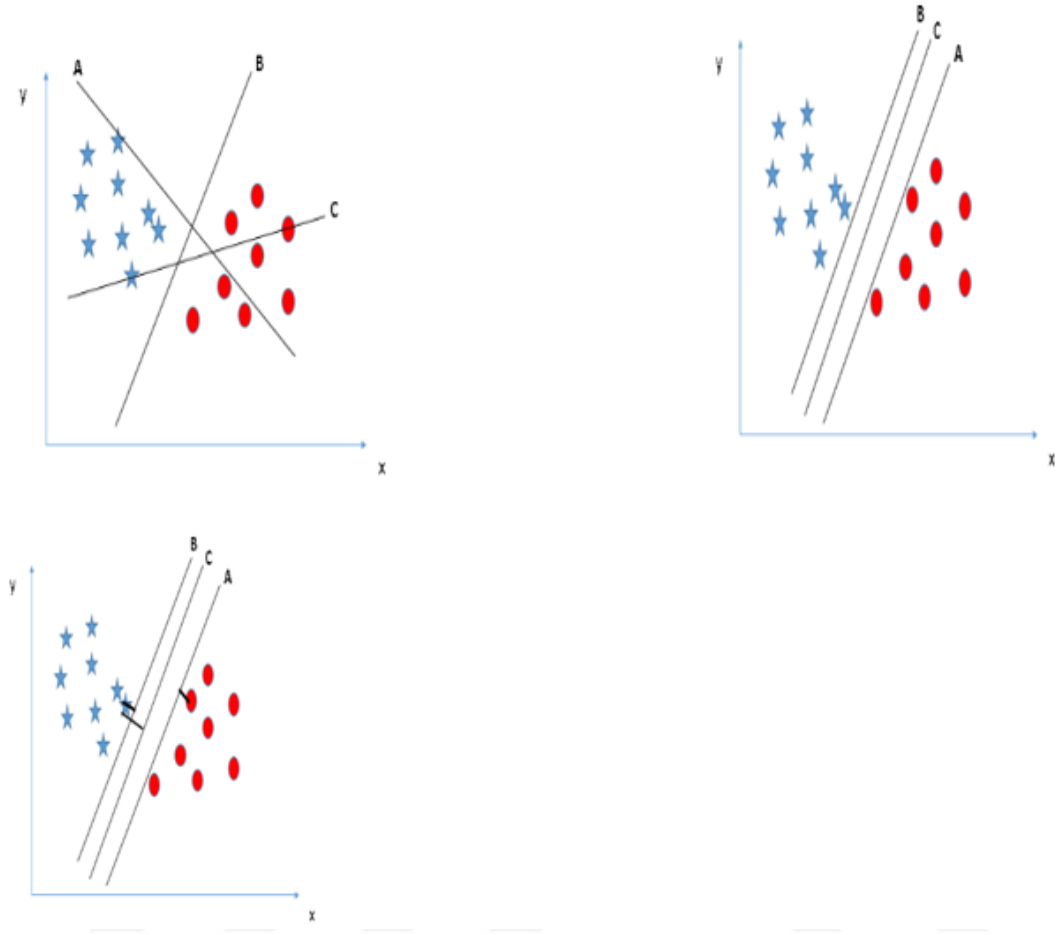
Problem uyarma algoritmaları geleneksel makine öğrenmesi yöntemlerini daha etkili işlemek için doğrudan kişiselleştiriyor. Bu yöntemlerin bize sağladığı avantajlardan bir tanesi kullanılan algoritmaya tamamen odaklanabilmek ve veri seti hakkında istenen sonuçlara şeffaf bir şekilde yaklaşabilme olanağı sunmaktadır bir diğer avantaj ise çok etiketli sınıflandırıcısını eğitmek için tüm eğitim veri setini (örnekler ve etiketler) aynı anda kullanılmasıdır(Sawsan 2017:20). Genel anlamda öğrenme algoritmalarının pek çok uyarlamaları önerilmiştir bunlardan bazıları karar ağaçları, destek vektör makinaları algoritmaları tüm etiketleri ve örnekleri aynı anda genişletme fırsatı sunarken diğer bir örnek olan K-NN algoritması ise her sınıfı ayrı olarak düşünür. Bu yöntemlere ve tabloda yer almayan ancak kullanım sahası olarak çok yaygın olan algoritmaların ve yöntemlerin birkaçına aşağıda kısa bir şekilde değinilmiştir (Sawsan 2017:20).

2.4.5.1. Karar Ağaçları Ve Güçlendirme

Karar ağaçları makine öğrenmesi sınıflandırmaları içerisinde en çok bilinen ve uygulanan yöntemlerindendir. Veri setindeki eğitim örnekleri tamamından başlayarak bir karar ağacı oluşturma ve eğitim veri kümesini göz önünde bulundurarak bir ön gürü sağlayacak ağaç yapı modeli inşa etmeye çalışır. Ağaç yapısında bulunan düğümler kendisinden sonra gelecek olan dallara bağlı pek çok özelliği temsil eder. Düğümler göz önünde olmayan her örnek için dalların yolunu izleyerek kök düğümden terminal yaprağına kadar izleyerek tahmin etmeye çalışır. Her farklı etiket için pek çok sayıda kombinasyonları birbirinden farklı olan yaprak düzeni oluşturur. Bu yöntem kullanılan düzende yanlış sınıflandırmalara daha büyük bir maliyet atayabilir. İlk olarak fonksiyonel genomikten gelen çok etiketli veriler üzerinde değerlendirildi. Yükseltme çoklu etiketli öğrenmeye ve özellikle de metin kategorizasyonuna uygulandı (Schapire 2000). Bu yöntem iki farklı versiyonu içerir. MHMR bunlardan ilki olan Adaboosting.MH bir sorgu değeri için tüm doğru etiketlerin kümesini tahmin etmeyi amaçlarken ikincisi olan Adaboosting.MR en yüksek orantıda doğru biçimde etiketleri sıralayan sınıflandırıcıdır.

2.4.5.2. Destek Vektör Makineleri

Destek Vektör Makineleri güçlü istatistiksel teoriler üzerine inşa edilmiş bir makine öğrenmesi yöntemidir (Güner vd. 2010). Destek vektör makineleri temelde iki sınıfa ait verileri birbirinden en uygun şekilde ayırmak için kullanılmaktadır. İki sınıfa ait verileri, eğitim verisini kullanarak elde edilen bir karar fonksiyonu yardımıyla doğrusal olarak birbirinden ayrılmasını amaçlamaktadır. Sınıfları birbirinden en uygun şekilde ikiye ayıran doğruya karar doğrusu denilmektedir. Sınırsız adet doğru çizilebilmesine rağmen optimal doğruyu çizmek amaçlanmaktadır (Ulgen 2017).



Şekil 33. Destek Vektör Makineleri (SVM) Optimal Doğru Çizimi

A, B, C doğruları yıldız ve daireyi sınıflandırmak için oluşturulan doğrulardır. Solda bulunan ilk grafikte en doğru şekilde B doğrusu olduğu görülmektedir. İkinci grafikte ise 3 hiper düzlem arasında en optimal olanının nasıl tespit edildiği hususunda bakılacak olursa grafikte bulunan en yakın veri noktası veya sınıfların arasında bulunan mesafeleri maksimize etmek en doğru karar verilmesine yardımcı olacaktr. Sağda bulunan üçüncü grafikte ki mesafeler gösterilmektedir. Bu mesafelere marj denilmektedir. Üçüncü şekilde C hiper düzlemin hem A ya göre hem de B ye göre sınıflar ile olan mesafesi yaklaşık aynı orantıda bu durumda C hiper düzlemin en ideal düzlem olduğunu söylemek yanlış olmayacaktır. Diğer iki durumda yani A veya B'nin seçilmesi durumunda ise marjın düşük tutulması anlamına gelir sonuç itibariyle sınıflandırmanın yanlış olma ihtimalini yükseltmektedir.

SVM yaygın olarak çok boyutlu uzaylarda kullanılmaktadır. Uzayda ki boyut sayısı örneklem sayısından fazla olduğunda etkili olur. Veri setinin doğrusallık

durumuna bağlı olarak doğrusal destek vektör makinaları ve doğrusal olmayan destek vektör makinaları diye ikiye ayrılır ve Destek Vektör Makineleri (SVM), çoklu etiketli öğrenme problemini çözmek için, özellikle BR yaklaşımıyla birlikte, problem dönüştürme yöntemleriyle yaygın olarak kullanılmıştır (Shantanu vd. 2004). Svm algoritmasını, çok etiketli problemi doğrudan ele almak için genişletilerek kullanılması gerekir ve bunun için sadece bir yöntem olan RANK-SVM kullanılmıştır. RANK-SVM bir etiket seti ne kadar geniş boyuta sahip olmasını belirleyen doğrusal bir sıralama sitemine dayanan modeldir. Q doğrusal ayırt edici fonksiyon;

$$(q \in \{1, \dots, Q\}), fq(x) = hwq, xi + bq$$

Wq = ağırlık vektörü

Bq =yanlılık vektörü

Olarak bu bileşen de tanımlanmıştır. İki farklı şekilde çalışan bu model ilk olarak etiketlerin çıktı değerine bakarak onları sistematik bir sıralama düzenine sokar ve diğeri ise ilgili etiketleri diğerlerinden ayırt etmek için bir set boyut belirleyicisi görevini üstlenir. Fonksiyonel olarak $\big(t\,(x)\big), (fq\,(x) > t(x))$ tanımlaması bu şekildedir. Bu tanımlamanın temel amacı sıralama kaybını en aza indirmektir.

2.4.5.3. Tembel Öğrenme

Makine öğrenmesinde tembel öğrenme olarak bilinen lazy learning eğitim verilerinin detaylı bir şekilde ele almak yerine teorik olarak daha genelleştirilmiş şekilde kullanılmasını sağlanmaktadır. Yavaş öğrenme eğitim verilerini ve sorgularını sisteme uygulamadan önce daha genele indirgemeyi amaçlanmaktadır. Pek çok çevrim içi öneri sistemleri tarafından tercih edilen en yakın K komşusu algoritmasında benzer $k - nn$ algoritması eğitim verilerinden ayırıcı bir işlev öğrenmek yerine eğitim veri setini ezberlemeyi tercih eder. Örneğin lojistik regresyon algoritması veri setini işlediği müddetçe model ağırlıklarını yani parametrelerini öğrenirken K-NN algoritmasının bir veri seti işlemesinin süresinin olmaması parametre öğrenme durumunu egale etmektedir.

Bu yöntemde etiketlenmesinde elde edilen istatistiksel bilgiler göre, bir X test örneğine atanacak uygun etiket kümesini belirlemek için en yüksek miktarda bir arka

tahminini kullanılır (Kanj 2017:20). Y'deki her ωq için, x'in ωq'ye ait ya da ait olmadığı arka olasılıkları hesaplamak için her olası sınıfa ait komşu örneklerin sayısı kullanılır. Bu olasılıkların hangisinin daha büyük olduğuna bağlı olarak, x örneğini test etmek için ωq sınıfını atamaya veya almaya çalışır. Her etiket için karar bağımsız olarak verilir. Yukarıda da belirtildiği gibi iki ihtimal hesaplanırken etiketler arasındaki korelasyon kullanılır (Kanj 2017:21).

Çok etiketli sınıflandırma için k-NN yönteminin iki farklı uzantısı sunulmuştur. Çoğunlukla belge kategorizasyonuna uygulanan bu iki sürüm, birbirinden farklı pek çok kategorideki belgelerde bulunan eş anlamlı olan kelimeli dikkate alır. Yeni bir örnek geldiğinde ise en yakın komşudaki ilgili etiketler ele alınır ve bu etiket komşuların her birinde varsa her bir etikete karşılık gelen sayaçta bir artırmaya gidilir ve bundan dolayı test örneği en yüksek N ağırlıklı sayı olarak en fazla olan ile sınıflandırılır. N örnekte bulunan etiket sayısına göre seçilir yeni bir etiket ele alındığında etiket sayısı tanımlanamadığında bu yöntemlerin uygun olmadığını anlaşılır.

2.4.6. Topluluk Yöntemleri

Bir veri setinde aynı anda birden fazla sınıflandırıcı bilgisini tutarak ve her sınıflandırıcının kendi kararlarını bir çatı altında fikir birliği olarak uygulamaya alınabilir. Genel durumda bu yöntem tek bir sınıflandırıcıdan daha iyi performans gösterir. Birden fazla sınıflandırıcının alınan tahmin ve kararlarına dayanarak örnekleri sınıflandırma tekniğine topluluk öğrenme sınıflandırması denir. Topluluk yöntemleri veri setinden kullanılırken problem dönüştürmesi ve sınıflandırıcılar için problem adaptasyonunu sağlamak için kullanılır. Bu yöntemlerin pek çoğu yeni bir sınıflandırıcı ortaya koymak için bir arada tutulur. Birkaç topluluk sınıflandırma yöntemleri vardı.

2.4.6.1. Sınıflandırıcı Zincirler Topluluğu

Zincirler topluluğunda (ECC) eğitimde temel sınıflayıcılar olarak sınıflandırıcı zincirler (CC) kullanılır. Sınıflandırıcı zincirler topluluğunun, sınıflandırıcı zincirler üzerindeki etkisini azaltmak ve sınıflandırmak için önerildi. Görünmeyen bir örneği varsaydığımızda her bir etiket sınıflandırıcılar için tahminler toplanılır ve birleştirilir

böylece sınıflandırılan her etiket böylece birden fazla oy alır (Madjaroc vd. 2012). Son tahlilde kümelenen etiketlerin çıktılarını almak için bir eşik kullanılır. Bu topluluk yönteminin de herhangi bir problem dönüştürme yöntemlerinden biri temel sınıflayıcı olarak kullanılabiliyor olmasının yanında BR ve CC den hesaplama olarak daha iyi bir performans göstermektedir.

2.4.6.2. Rastgele K Etiket Setleri

Rastgele k etiket setleri metodu özellikle LP(label powerset) ve diğer problem dönüştürme yöntemleri ile ilişkilidir(Grigorios vd. 2009). Sınıf başına pek çok örnek barındırması çok sayıda etiket tarafından da meydana gelmesi yeni bir problem olarak RAKEL LP 'yi oluşturur. Bu yöntem tüm etiketlerden K kadar bir büyüklükte rastgele bir alt küme belirlemesini ve LP türünde çok etiketli bir sınıflandırıcı ile alt kümelerini eğitir. Yeni bir örnek geldiğinde ise LP'nin tüm sınıflandırıcıları son etiket kümesini belirlemek için basit yapıda bir oylama işlemi kullanarak birleştirir. Rakel en üst düzeyde bir performans elde etmek için optimize edilmesi gereken pek çok parametreye sahip bir yöntemdir. Veri setinde eğitim örnekleri sayısı yeterli olmadığı taktirde kullanılması zor olabilmektedir.

2.4.6.3. Çok Etiketli Sınıflandırıcıların Topluluğu

Çok etiketli sınıflandırıcıların sergiledikleri mevcut performansını artırmak ve etiketler arasında ki denge problemine çözüm üretmek için kullanılır. Amacı çok etiketli bir topluluk oluşturmak olan bu yöntemin H etiket sınıflandırıcısı görünmeyen bir örnek için topluluktan bir etiket ile eşleştirme yapmaya çalışır. Çok etiketli sınıflandırıcılar topluluğunu eğitirken sınıflandırıcı topluluğu olan $(H_1, H_2, \dots, \dots, \dots, H_q)$ tüm q modellerini farklı olmasıyla beraber beklenilenin dışında tahminlerde yürütebilir. Görünmeyen her bir x modeli için N boyutlu vektör $P_k = [P_{1k,} P_{2k,} \dots \dots \dots, P_{nk}]$ üretir. Yukarıda belirtilen q sınıflandırıcıların pek çıktılarını ortak bir paydada birleştirmenin pek çok yöntemi vardır. Bunların arasına $MEAN, MAX, MIN$ gibi yöntemler çıktıların olasılık puanlarını hesaplayarak birleştirmenin en çok bilinen yollarıdır (Gayar vd. 2010). Bu birleştirmeleri yaparken kullanılacak çok fazla parametresi olan bu yöntem yüklenemeyen kombinasyonları ağırlıklı oylama yöntemleri kullanarak çok etiketli sınıflandırma sistemlerinden

bireysel seçim yapma potansiyeline sahiptir. Sınıflandırıcılar tarafından duruma göre statik ve dinamik olmak üzere iki farklı yöntem ile ağırlık değeri belirlenir. Dinamik yöntemde bireysel sınıflandırıcılara atanan ağırlık puanı her test modelinde değişkenlik göstermekteyken statik yöntemde kullanım aşamasında her sınıflandırıcı ya verilen ağırlıklar hesaplanır ve hesaplama sırasında ağırlık sınıflandırması sabit tutulur.

2.4.6.3.1. (EML(a)) olasılıkların ortalaması

Ortalama olasılıkların kullanım açısından en eski yöntemdir. Karar verme, tanımlama ve sınıflandırma gibi durumlarda yaygın olarak kullanılması ile beraber, ortalama alma ve farklı örnekler için topluluk kararını göz önünde bulundurarak sonuçlandırmayı en nihai şekilde yapması yöntemin en önemli özelliklerindendir. Aşağıda ki denklem X örneği için yb etiketi baz alınmıştır.

$$\mu_b(x) = \frac{1}{q}\sum_{k=1}^{q} p_{bk}(x)$$

2.4.6.3.2. Çoklu Etiketleme ile Olasılık Ortalaması ve Eşik seçimi

Karar eşiklerinin geleneksel olarak kabul edilen 0,5 oranından farklı olarak bir değer seçimi ve bu seçimin uygun olarak ayarlanabilmesi durumunda çok etiketli bir sınıflandırıcının mevcut bir sınıflandırıcıya göre performansının geliştirilmesinin daha mümkün olduğunu bildirmiştir. Q modellerinden oluşan ve olasılıklarından elde edilen toplamının vektöre edilmesi

$$W = (\theta_1,,,,,,\theta_N) \in \quad R^N \qquad \theta_b = \sum_{k=1}^{q} P_{bk}$$

w daha sonra 0,1 de bulunan dağılımını temsilen W norm göre normalize etmeye çalışır. X s yi test Xt ise eğitim dizini olarak kabul eder bu dizinden sonra eşik değerini yukarıda belirtilen denklem kullanılarak seçilir. LCard(label cardinality) çoklu etiketlenmişliğin standartlaşmış ölçüsüdür. (Tsoumakas 2009) Kullanılan her bir vaka ile ilgili ortalama etiket sayıları

$$LCard(x) = \frac{\sum_{i=1}^{x} |Ei|}{|x|}$$

Olarak verilir ve bu bağlamda Ei eğitim seti için gerçek etiket seti olarak kullanılırken test setinde ise t eşiğinin altında kalan etiket değerlerini öngören bir settir. Bu yöntemde gerçek test etiketleri kendisine etiketlerin kullanılması için eşik seçim yönteminin başarılı olmadığını kanıtlar. T eşikleri sadece önceden tahmin edilen etiket kümelerinin kullanılmasıyla hesaplanır.

2.4.6.3.3. N Katlı Çapraz Doğrulama (EMLS) ile Statik Ağırlıklandırma

Statik ağırlandırma metodunda kullanılan her sınıflandırıcının ağırlık durumları eğitim aşamasında hesaplanılır**.** Hesaplanan her bir sınıflandırıcının ağırlıkları sınıflandırıcının kalite ölçüsüne oranla öğrenilebilmesi için pek çok strateji önerilirken en yaygın kullanılan ve hesaplamada ortalama kesinlik durumunu veren yöntem N katmanlı çapraz doğrulamadır ve yaygın olarak N=5 olarak kabul edilir.

2.4.6.3.4. Dudani Kuralı ile Dinamik Ağırlıklandırma

Yeni modellerin sınıflandırılması için KNN ağırlıklı bir kural önerilmiştir. Bu yaklaşımda kullanılan ağırlıklandırma yöntemi en yakın komşunun k oylarının test düzenine olan mesafelerin göz önünde bulundurularak ağırlandırmasıdır. Bu yapıda temel amaç daha yüksek mesafeli bir KNN komşusuna değil daha ağır ancak daha küçük mesafeli bir komşunun sınıflandırılmasıdır.

Yapıda en yakın komşunun ağırlığı 1 olarak alınırken en uzak komşunun ağırlığı 0'dır. Ve diğer ağırlıkların değerlerinin ölçüsünü almak için doğrusal ölçeklendirme yöntemi kullanılır (Valdovinos vd. 2009). Dudanin ağırlık hesaplama yöntemi;

$$w_j = \begin{cases} \frac{d_k - d_j}{d_k - d_1} \\ 1 \end{cases}$$

Eğer $d_k \neq d_1$ olması durumunda 1 olarak seçilir. Bu denklemde dj, jth'nin en yakın komşunun mesafesini belirtirken, d1 en yakın komşunun mesafesi ve dk en uzaktaki komşunun mesafesini belirtmektedir. Dudanin ağırlıklandırma fonksiyonu sınıflandırıcı füzyonu içinde kullanılabilir. Model de belirtilen k değerinin aşağıda gösterilen q modellerinin sayısıyla değiştirildiğin de ve her bir sınıflandırıcı için x

bilinmeyen ve örneğinde q mesafeleri artan bir düzen sıralama $(d1, d2, \ldots, dq)$ şeklinde olur. Böylece denklem

$$weight\ (M_j) = \begin{cases} \dfrac{d_q - d_j}{d_q - d_1} \\ 1 \end{cases}$$

Ve eğer $d_q \neq d_1$ değer 1 olarak alınır. D1, x in en yakın komşularının q mesafeleri arasındaki en küçüğüdür ve aynı zamanda dg bu mesafelerin en büyüğüdür. Denklemin eşitliğinde bulunan $w(mj)$ çoklu X örneğinde etiket sınıflandırıcısının merkezi J ağırlığıdır (Tahir vd. 2012).

2.4.6.3.5. Shepard kuralı ile dinamik ağırlıklandırma (EMLP))

1987 yılında shepard ın yaptığı çalışmada başka bir ağırlık fonksiyonu önermiş ve çok etiketli sınıflandırıcıların ağırlık merkezlerini ölçmeye karar vermiştir. Shepard göre yeni bir uyarıcıya genelleme yapabilmesi için mevcuttaki uyarıcının alaka düzeyinin herkes tarafından kabul edilen bir alaka düzeyi olması gerektiğini savunuyor. Bahsi gecen alaka düzeylerinin belirlenebilmesi yapılacak oylama işlemi şekillendirir ve bu oylama işlemi aşağıdaki yolun izlenmesi ile elde edilir.

$$weight\ (M_j) = e^{-xd_j^{\beta}}$$

Üssel fonksiyonun eğim ve gücünü belirlemesi durumunda ilk olarak X ve ß sabit tutulur ve daha sonra X ve ß değer olarak 1 ataması yapılır.

ÜÇÜNCÜ BÖLÜM

3. UYGULAMA

3.1. Sınıflandırma Öncesi Ön İşlemler

Çok etiketli sınıflandırma işlemi birden fazla adımdan oluşan süreçtir. Bu sürecin amacı sınıflandırma yapılacak olan veri setinin algoritmalar tarafından daha iyi bir şekilde sınıflandırılabilir hale getirilmesidir. Bundan dolayı çok etiketli sınıflandırma işlemi bazı işlere tabi tutulur. Süreç içerisinde yapılan bu işlemlerin temel hedefi sınıflandırma yapmadan önce veri setini kullanılabilir bir standarda sokmak hem de veri seti içerisinde bulunan gereksiz kelime boşluklardan arındırmaktır. Yapılacak olan bu işlemlerin sonucunda başarıya etkisi olmayan kelimelerin veri setine dâhil edilmemiş olmasının yanında algoritmalardan elde edilecek performansın daha iyi sonuçlar vermesini sağlayabilir.

3.1.1. Gereksiz Kelimelerin Çıkarılması

Gereksiz kelime çıkartımı Veri seti içerisinde herhangi bir anlam ifade etmeyen fakat metnin veya veri setinin içerisinde sıkça gecen edat, bağlaç gibi kelimelerin temizlenme sürecidir. Gereksiz kelimelerin temizlenmesi model için uygulanacak algoritmanın daha hızlı ve efektif çalıştırılması sağlanmış olur. Bunun yanında, gereksiz kelimelerden kaynaklanan hata paylarının azaltılması hedeflenir.

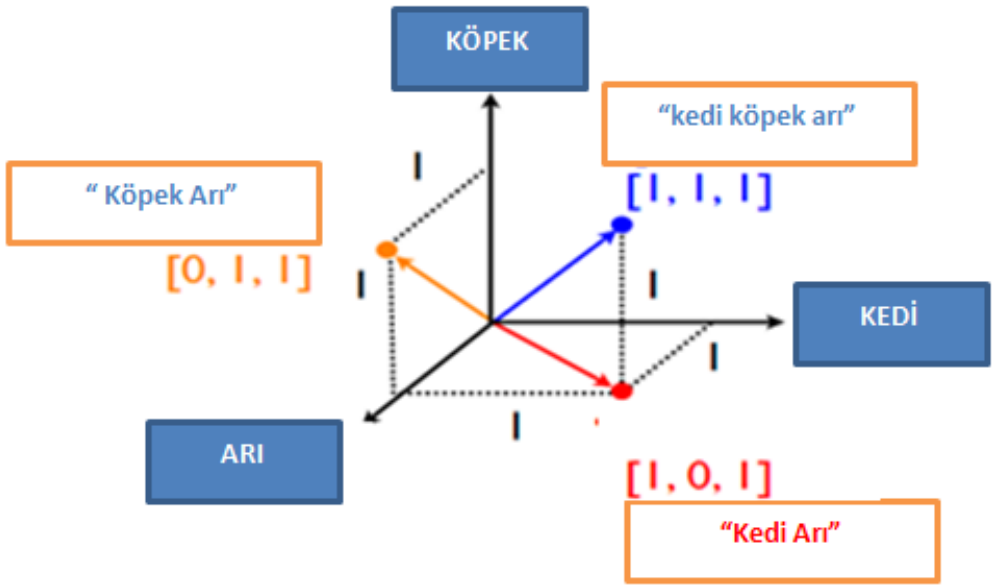
3.1.2. Vektörleştirme

Bir önceki işlemler gerçekleştirildikten sonra verinin içerisinde bulunan gereksiz ve başarıyı olumsuz olarak etkileyebilecek faktörler giderilmiş olur. Bu işlemlerden sonra sınıflandırılmada kullanılacak verinin, algoritmaların anlayabileceği şekilde temsil edilmesi gerekmektedir. Yapılan bu işleme vektörleştirme denilir. Yapılacak olan vektörleşme işlemi için öncelik olarak vektörlerin gösterileceği uzay modelinin kurulması gerekir. Vektör uzay modeli metinlerin, kelimelerin yeterli büyüklüğe sahip bir boyutta temsil edebilmemiz sağlayan matematiksel bir yöntemdir. Oluşturulan vektör uzayına iz düşürülen kelimelerin arasında bulunan ilişkiyi, temsili

vektörler kullanılarak bulunmaya çalışılır. Kelime çantası(bag-of-words) olarak bilinen bu yöntem de kelimelere ait dil bilgisi yapısı ve bulunduğu metinlerde ki geçiş sırası önemsiz olarak kabul edilmektedir (*Vector Space Model* 2017).

	Kedi	Köpek	Arı
Dök1	1	1	1
Dök2	1	0	1
Dök3	0	1	1

Şekil 34. Örnek Dokümanlar ve Kelime Frekans Değerleri



Şekil 35. Örnek Dokümanlara Ait Vektör Uzayı

Örneğin yukarıda bulunan şekil 35'teki gibi 3 farklı dokümandan oluşan veriler mevcut olduğunu varsayalım. Eğer ilgili terim dokümanda bulunuyorsa "1" bulunmuyorsa "0" değerini alsın. Matrisin her satırı ilgili dokümanı temsil eden vektör olarak adlandırılmaktadır. Vektörlerden elde edilen bilgilerden yola çıkılarak bir uzay modeli oluşturuldu. Vektör uzayı oluşturulduktan sonra her bir kelimeye dokumanda ki önemini temsil edecek bir biçimde ağırlık veya yukarıda belirtildiği gibi değer verilir bu ağırlıklandırma 4 farklı şekilde yapılır.

3.1.2.1. Binary (ikili) Ağırlıklandırma

Eğer kelime dokümanın içerisinde bulunuyorsa 1, bulunmuyorsa 0 değerini alacaktır.

3.1.2.2 Geçme Sıklığı

Kelimelerin ağırlığı yukarıda ki şekilde de gösterildiği gibi mevcut dokümanda ne kadar yer alıyorsa o kadar olarak belirlenmesidir.

Dök1	Hafsa kitap okur
Dök2	Hafsa güzel kitap okur
Dök3	Hafsa kitabı okur

Şekil 36. Kelimelerin Geçme Sıklığı

Terim-frekans matrisi	Hafsa	Kitap	Kitabı	Güzel	Okur
Dök1	0	1	0	0	1
Dök2	1	1	0	1	1
Dök3	1	0	1	0	1

Şekil 37. Geçme Sıklığı ve Kelime Ağırlıklandırma Yöntemi

3.1.2.3. Terim Frekansı

Terim frekansı kelimelerin ağırlığını, dokümanın içerisinde geçme sayısını, doküman uzunluğuna bölerek hesaplar. Bu yöntem sık kullanılan ağırlıklandırma yöntemlerinden biridir.

Terim-frekans matrisi	Hafsa	Kitap	Kitabı	Güzel	Okur
Dök1	0,33	0,33	0	0	0,33
Dök2	0,25	0,25	0	0,25	0,25
Dök3	0,33	0	0,33	0	0,33

Şekil 38. Terim Frekansı Kelime Ağırlıklandırma Yönntemi

3.1.2.4. Terim Frekansı-Ters Doküman Frekansı ile Ağırlıklandırma

Bir kelimenin bir veri kümesinde veri için ne kadar önemli olduğu gösteren bir ölçüdür. Temelde TF-IDF, belge arama ve kelimelerin yapısı ile ilgili bil oluşturmak için kullanılan bir yöntemdir. Terim frekansı yukarıda şekil ile ifade edildiği gibidir. Ters doküman frekansı ise bir dizi belge içerisinde seçilen bir kelimenin tüm belge içerisinde ne kadar yaygın veya ne kadar az kullanıldığı anlamına gelir. Frekans değeri sıfıra ne kadar yakınsa kelimenin yaygınlık oranı o kadar fazladır anlamına gelir (Bruno 2019). Bu ölçü toplam doküman veya belge sayısını alarak bir kelimeyi içeren belge sayısına bölerek ve logaritmasını alarak hesaplanır. Örneğin, 100 tanesinde " kitap " kelimesi geçen 1000 doküman bulunsun.

$$w_{i,j} = tf_{i,j} * idf\left(\frac{N}{df_i}\right) = \left(\frac{10}{200}\right) * \log\left(\frac{1000}{100}\right) = 0.2$$

Bunların içerisinden 200 kelimeden oluşan bir dokümanda da 10 kere "*kitap*" kelimesi geçsin. Bu durumda ilgili kelimenin ilgili doküman için TF-IDF değeri eşitlik 41 ile hesaplanır (Işık 2018:54).

3.3.3. Çapraz Doğrulama (Cross Validation)

Bir verinin modellenmesi verinin istatistiksel analizinin yapılması anlamına gelmektedir. Oluşturulan model ile bağımsız bir veri kümesinin nasıl genel bir yapıya oturulacağı ve oluşturulan bu yapının bir değerlendirme aşamasından geçmesi gerekmektedir. Çapraz doğrulama modeli değerlendirmek için kullanılan bir doğrulama tekniğidir. Çapraz doğrulamanın amacı, modelin hızlı öğrenmesinden veya

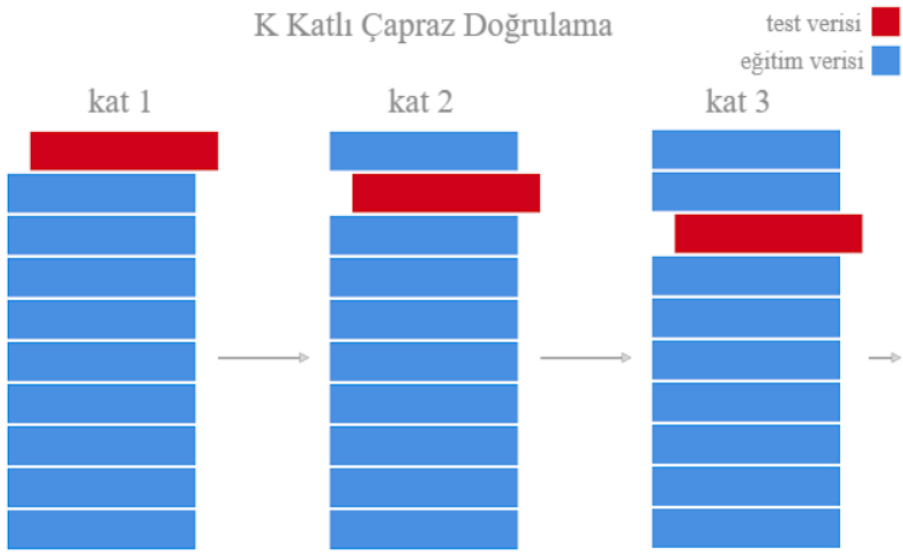
yavaş öğrenmesinden bağımsız bir şekilde herhangi veri kümesini istenilen bir şekilde genelleştirmek için model hakkında fikir sahibi olmaktır. Çapraz doğrulama temelde üç adımda gerçekleşir (Drakos 2018).

- Veri kümesinin bir kısmı ayırtılır.
- Veri kümesinin kalan kısmı modeli eğitmek için kullanılır.
- Veri setinden ayırılan kısım ile de model test edilir.

Çapraz doğrulama yöntemleri arasında en fazla tercih edilen K katlı çapraz doğrulama yöntemidir.(K- Fold cross validation)

3.1.3.1. K Katlı Çapraz Doğrulama (K- Fold cross validation)

Bir modeli tam anlamıyla eğitmek için her zaman veri noksanlığı gibi bir problem olarak ortaya çıkmaktadır. Verinin sadece bir kısmını ayırmak genelde yetersiz uyum sorununu meydana getirmektedir. Ve buna bağlı olarak Eğitim verilerini azaltmak yanlılığın neden olacağı hatayı artırmaktadır. Bu gibi sorunlar modeli oluşturmak için eğitim verisinin geniş bir yapıya sahip olması gerekmektedir. Ve K katlı çapraz doğrulama tam olarak bunu yapmaktadır. K katlı çapraz doğrusal yönteminde veri kümesini K sayıda al kümeye parçaladıktan sonra tüm alt kümeler üzerinde eğitim verilmekte , fakat eğitim için ayrılan modelin değerlendirilmesi için K için bir alt(k-1) küme bırakılmaktadır. (Cross Validation 2019).



Şekil 39. K Katlı Çapraz Doğrulama Yöntemi

Yukarıda ki şekilde veri K katlı parçalara bölündüğünde aşırı öğrenme veya yetersiz öğrenme problemini önemli ölçüde azaltmaktadır. Böyle bir yapı yapılacak analizde veri sayısının az olmasından kaynaklanan çalışmalarda da optimal sonuçlar elde etmek için önemli bir yolu olarak bilinmektedir. Yapılacak bölünme genel olarak K=5 veya K=10 olarak seçililmektedir. Uygulama kısmında tercih bizimde tercih edeceğimiz veriyi K=10 şeklinde ayırmak olacaktır. Çünkü yapılan bu tercih yüksek varyans ve yüksen yanlılıktan kaynaklanan problemleri giderecektir.

3.2. Değerlendirme Ölçütleri

Genellikle tek etiketli sınıflandırma çalışmalarında kullanılan değerlendirme ölçütleri bir verinin sınıflandırılmasıyla ilgili sadece iki olası durumu göz önünde tutmaktadır. Veriye uygulanan algoritma ile oluşturulan model neticesinde ya doğru sınıflandırılmıştır ya da yanlış sınıflandırılmıştır. Çok etiketli sınıflandırmada kullanılan metrikler, tek etiketli sınıflandırmada kullanılan metriklerden genel itibariyle farklı olarak kullanılır. Bazı durumlarda ise çok etiketli sınıflandırmada kullanılan önlemlerin, tek etiketli sınıflandırma da kullanılan önlemlerin uyarlamaları olduğu görülmektedir (Everton, Maria, Jean 2011). Çok etiketli sınıflandırılmada kullanılan bazı değerlendirme ölçütleri; Kesinlik(precision), hassasiyet(recall), Doğruluk(accuracy)dir.

3.2.1. Kesinlik (Precision)

Bir veri setinin olumlu tespitinin ne kadar gerçeğe yakın olduğunu ortaya koymak için kullanılır. Hassasiyetin yüksek olması beklenilir hassasiyet ne kadar yüksek olursa algılama mekanizmasının o kadar yüksek olacaktır. Hassasiyetin yanlış ölçülmesi halinde yapılan tespitin yanlış olma ihtimali yükseltir. Örneğin kanser teşhisi üzerine yapılan bir çalışmanın hassasiyet oranı %10 çıkması tüm hastaların kanser olduğu kanısına varılmasına sağlamaktadır.

$$\boldsymbol{Kesinlik\ P = \frac{TP}{TP + FP}}$$

Pozitif olarak etiketlenen örneklerin sayısının, Pozitif olarak sınıflandırılmış toplam örneklere oranıdır (Kızılkaya vd. 2018).

3.2.2. $Hassasiyet$ (Recall)

Duyarlılık oluşturulan model tarafından doğru bir şekilde tahmin edilen bir sınıftan numunelerin bir kesri olarak tanımlandığı bir ölçüttür.

$$\boldsymbol{Hassasiyet} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FN}}$$

Pozitif olarak etiketlenmiş örneklerin, Pozitif olan örneklerin toplan sayısına oranıdır.

3.2.3. Doğruluk (Accuracy)

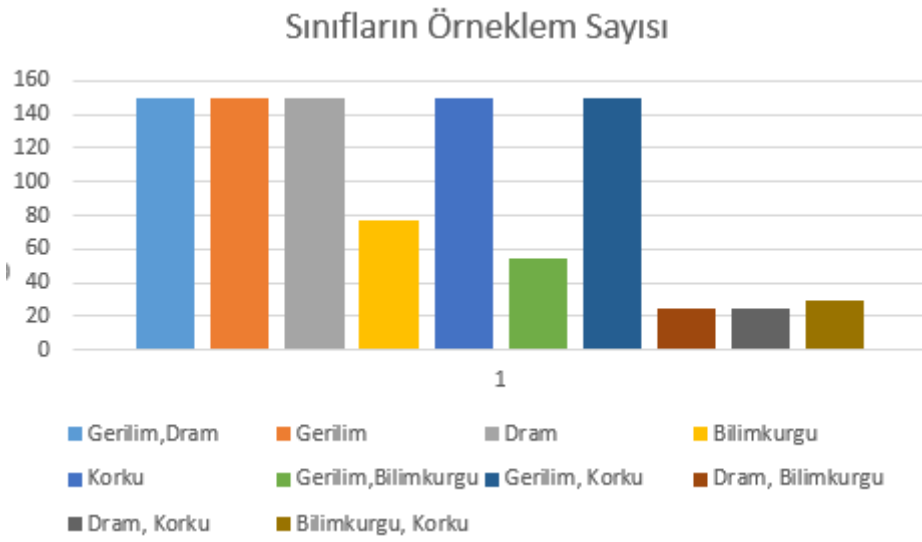
İyi bir şekil de dengelenmemiş çarpık olmayan sınıflandırma problemlerinde kullanılan bir ölçüttür. Modelleme de ana performans ölçütünü doğruluk olarak kullanmanın dezavantajı sınıflar arasında ciddi bir dengesizlik var ise ölçütün iyi sonuçlar vermemektedir.

$$\boldsymbol{Doğruluk} = \frac{\boldsymbol{TP} + \boldsymbol{TN}}{\boldsymbol{TP} + \boldsymbol{TN} + \boldsymbol{FP} + \boldsymbol{FN}}$$

Sınıflandırma işlemlerinden en çok tercih edilen ölçümdür. Doğru olarak sınıflandırılmış örneklerin toplam örnek sayısına oranıdır.

3.3. Veri Seti

Çok etiketli sınıflandırma için kullanılacak olan veri seti Türkiye de faaliyet gösteren uluslararası bir film ve dizi içeriği olan web sitesinden alınmıştır. Python'a ait BeautifulSoup paketi kullanılarak veri seti elde edilmiştir. Bu veri seti filmin adı, türü ve özetine ait bilgiler içermektedir. Film; Gerilim, Dram, Bilim Kurgu, Korku, (Gerilim, Dram), (Gerilim, Bilim Kurgu), (Gerilim, Korku), (Dram, Bilim Kurgu), (Dram, Korku), (Bilim Kurgu, Korku) sınıflardan oluşmaktadır.



Şekil 40. Sınıfların Örneklem Grafiği

Bazı filimler sadece bir sınıftan oluşurken diğerleri ise iki konu işlediği için çok sınıflı sınıflandırmalardan meydana gelmektedir. Veri seti 10 sınıftan oluşmakla beraber toplam örneklem sayısı 958, toplam öznitelik sayısı 20,505dir.

Veri Setinde ki her tür ve tür kombinasyonlarında eşit miktarda film sayısının olmaması nedeniyle dengesiz veri seti problemi oluşmuştur. Bu problemin çözümü için veri setinde dengesizliğe neden olacak tür ve kombinasyonları dışında sadece ["Aksyion","Gerilim","Dram","Bilimkurgu","Korku"] ve bu türlerin birbirleri arasındaki kombinasyonlarına (örneğin Aksiyon ve Dram) ait filmler ayıklanmıştır. Ayıklanan bu film özetlerinde bulunan noktalama işaretleri ve gereksiz kelimeleri temizlenmiş ve özetler işlenmeye hazır hale getirilmiştir.

Veri setini sınıflandırmadan önce veri setinin üzerinde bazı ön hazırlıklar yapılmalıdır. Bu hazırlıklar veriden veriye değişkenlik göstermekle beraber benzerlikler de içerebilmektedir. Ön işlemenin büyük bölümü kodlama ile giderilmektedir. Veri öncelikle başarı oranını etkileyecek boş değer ve gereksiz bilgilerden arındırılmıştır. Bu gereksiz bilgilerin içerisinde filme ait, seyirciler tarafından yapılan puanlama ve web sitesine ait filme verilen puan gibi bilgiler içermekteydi. Aynı anlama gelen fakat farklı şekilde etiketlenmiş veriler silinerek veride ki ikilikler ile beraber sonuca olumsuz etki edecek etiket setleri giderilmiştir. Veride mevcut olan bağlaç ve edat gibi cümlenin anlamın da herhangi bir değişiklik meydana getirmeyecek kelime ve gelime grupları veriden çıkarılmıştır. Bunun

yanında, cümle içerisinde geçen gereksiz boşluk, latince karakter ve sayılar gibi modeli olumsuz etkileyecek ifadeler temizlenmiştir. Yapılan bu işlemden sonra veri seti vektörize edilmesi, yani etiket türlerinin ikili sınıflandırmaya ayrılması gerekmektedir. Vektörleştirme işlemi için CountVectorizer kütüphanesi kullanılmıştır. Bu kütüphane, her örneklem için veri setinde bulunan eşsiz kelime kadar bir vektör oluşturmaktadır. Örneklem içerisinde geçen kelimeye denk gelen vektör değeri, kelimenin geçme sayısıyla değiştirilir diğerleri ise 0 olarak kalır. Böylelikle her bir örneklem vektör uzayına aktarılmış olur.

Bu işlem iki aşamada gerçekleşmektedir; İlk adım LabelEncoder ikinci adım ise OneHotEncode. İlk işlem olan LabelEncode çalışma yapısında veri setinde bulunan etiketlerin model tarafından anlaşılması için sayısal verilere dönüştürülmesi işlemidir. Aslında LabelEncode yaptığı işlem etiket setlerini birer sayısal değere dönüştürme işlemidir. Yani, [komedi, polisiye, animasyon] veri setini [0, 1, 2] dönüştürmüştür. İkinci adım olan OneHotEncode işlemi ise modelin istediği etiketlerin atanabileceği ikili sınıflandırma sistemine çevirmesidir. OneHotEncode çalışma şekli ise etiketlenmiş ve kategorik olarak ayrılmış veriyi, sütunu toplam label sayısınca bölmektedir. Sayıların hangi sütuna ve hangi değere sahip olduğunu bağlı olarak 1 veya 0 olarak atmaktadır (Sunny 2018).

Lojistik regresyon, Destek Vektör Makineleri (SVM) ve Naive Bayes algoritmalarının multilabel sınıflama özelliği bulunmadığı için, 5 sınıf ayrı ayrı tek sınıfmış gibi düşünülerek eğitilmiştir. Her eğitim sonrası ilgili sınıfların tahmin değerleri birleştirilerek her bir örneklem için multilabel sınıf vektörü elde edilmiştir.

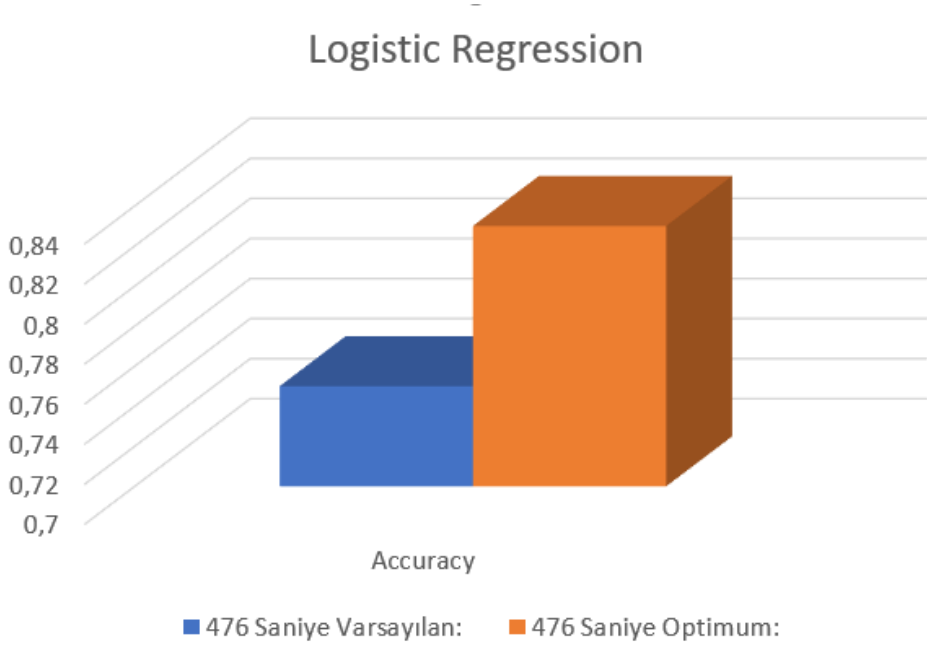
Tüm sınıflama algoritmaların hiper parametreleri python scikit-optimize kütüphanesinin Bayes optimizasyon tekniği kullanılarak optimize edilmiştir. Her bir parametrenin hangi aralıkta optimize edildiği ilgili sınıflama yöntemiyle beraber anlatılmıştır.

Lojistik regresyon deneyi için Scikit-learn kütüphanesinin LinearSVC modülü kullanılmıştır. Bu sınıflama algoritmasının düzenleyici parametresi (C, regression paramater) 0,0001 ila 100 arasında bayes yaklaşımıyla optimize edilmiştir. Varsayılan parametrelerle ve optimizasyon sonucunda elde edilen başarım değerleri çalışma süreleriyle tablo 1 de gösterilmiştir.

Tablo 1. Logistic Regression Algortiması Sonuç Tablosu

Logistic Regression			
Çalışma Süresi		C	Accuracy
476 Saniye	Varsayılan:	2	0,75
	Optimum:	0,01208	0,83

Lojistik regression algoritmasının kısa süre içerisinde belirgin bir optimum başarı sağlaması

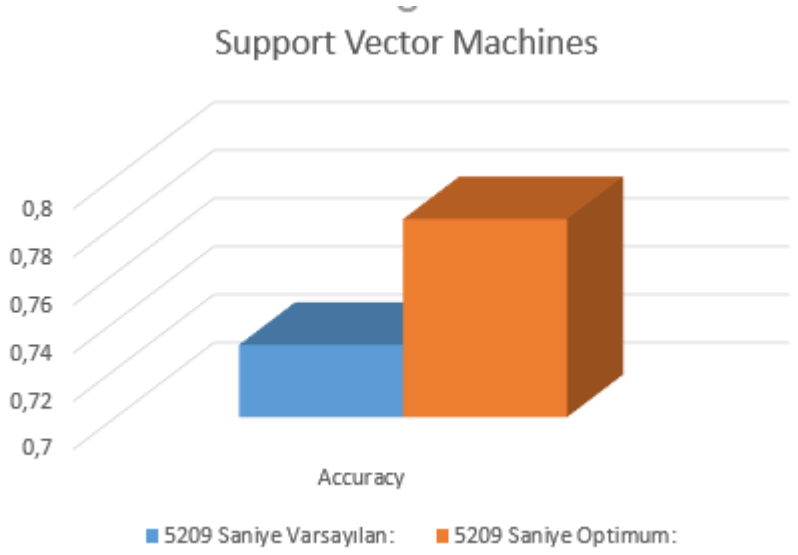


Şekil 41. Logistic Regression Algoritması Optimum Başarı Grafiği

Benzer parametrelere sahip olan Destek Vektör makineleri için de düzenleyici parametre(C) 0,0001 ila 100 arasında bayes yaklaşımıyla optimize edilmiştir. Elde edilen sonuçlar Tablo 2 de gösterilmektedir.

Tablo 2. SVM Algoritması Sonuç Tablosu

Support Vector Machines			
Çalışma Süresi		C	Accuracy
5209 Saniye	Varsayılan:	2	0,73
	Optimum:	97,384	0,78

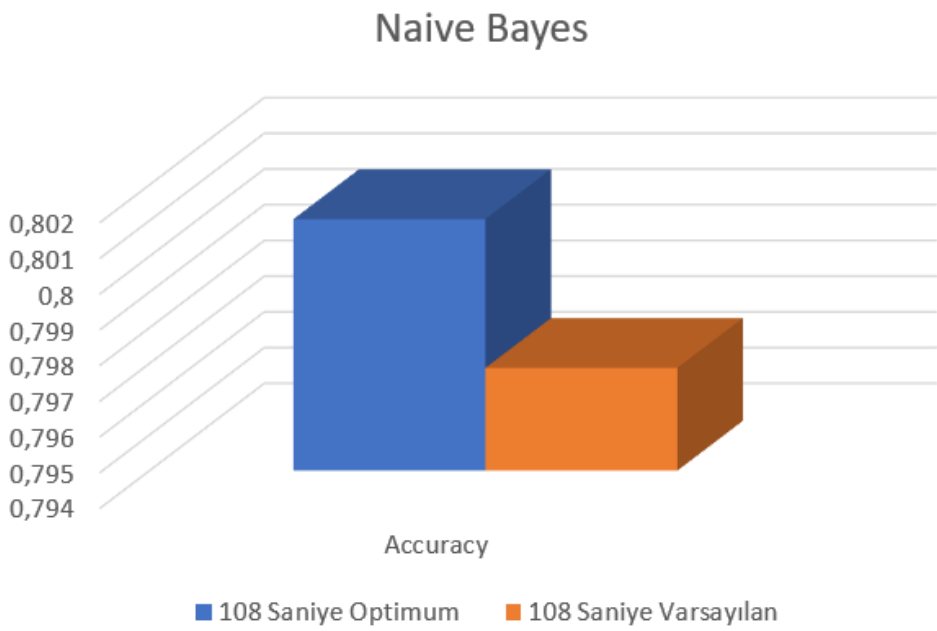


Şekil 42. SVM Algoritması Optimum Başarı Grafiği

Naive Bayes algoritması için metin sınıflama çalışmalarına uygunluğu kabul edilmiş Multinominal Naive Bayes algoritmasından yararlanılmıştır. Multinominal naive bayes, Gaussian Naive Bayes benzer şekilde çalışır. Genelde multinominal naive bayes sınıflandırıcısı veri setleri dağınık olduğunda kullanılır. Naive Bayes sınıflandırıcısı, modeldeki özelliklerin her birinin koşullu bağımsızlığını ifade eden genel bir terimken, Multinomial Naive Bayes sınıflandırıcısı, her özellik için çok terimli bir dağıtım kullanan bir Naive Bayes sınıflandırıcısının belirli bir örneği olarak kabul edilir. Bu algoritmanın laplace düzenleme parametresi olan alpha değeri 1 ile 50 arasında uniform olacak şekilde optimize edilmiştir. Elde edilen başarım değerleri tablo 3'te gösterilmiştir.

Tablo 3. Naive Bayes Algoritması Sonuç Tablosu

Naive Bayes			
Çalışma Süresi		alpha	Accuracy
108 Saniye	Optimum	1,386	0,801041
	Varsayılan	1	0,796875

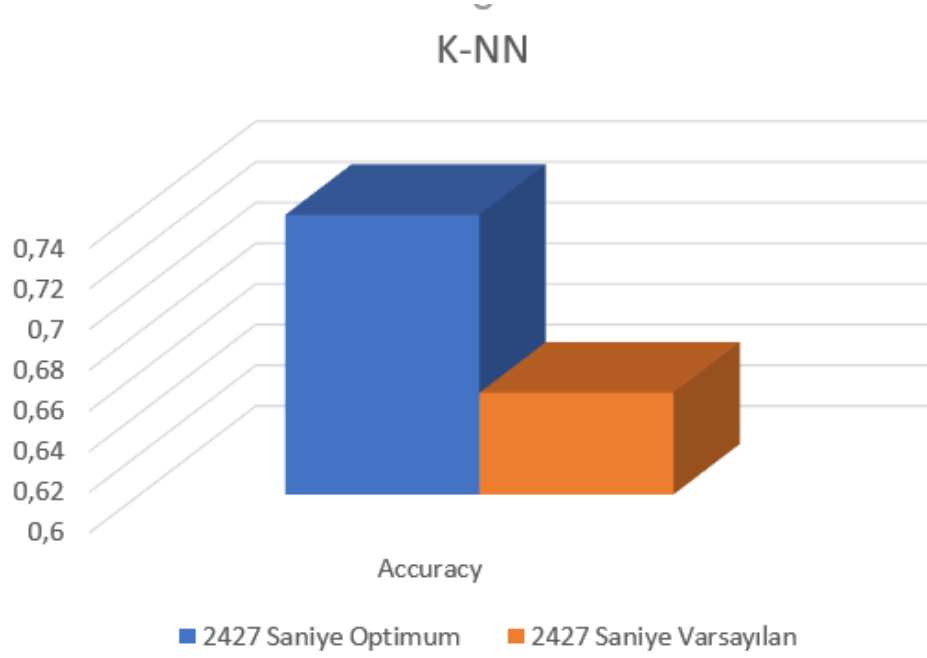


Şekil 43. Naive bayes Algoritması Optimum Başarı Grafiği

En yakın komşu (KNN) algoritması için 4 farklı hiper parametre optimize edilmiştir. Bunlar sırasıyla komşu sayısı, hesaplama algoritması, yaprak sayısı ve minkowski parametreleridir. Komşu sayısı 2 ila 700, yaprak sayısı 10 ila 200, minkowski parametresi ise 2 ila 50 arasında optimize edilmiştir. Algoritmalardan ise ball_tree, kd_tree ve brute denenmiştir. Elde edilen en iyi ve varsayılan sonuçlar tablo 4'te gösterilmiştir.

Tablo 4. KNN algoritması Sonuç Tablosu

KNN						
Çalışma Süresi		Komşu	Algoritma	Yaprak Sayısı	Minkowski Katsayısı	Accuracy
2427 Saniye	Optimum	492	ball_tree	48	21	0,7375
	Varsayılan	5	ball_tree	30	2	0,65

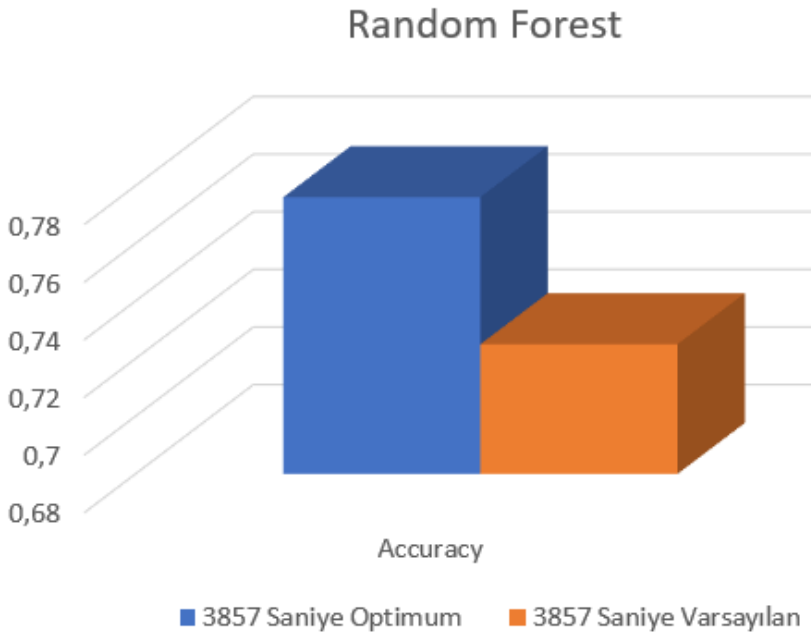


Şekil 44. K-NN algortiması Optimum Başarı Grafiği

Random Forest algoritması için uygulamada 4 parametre kullanılarak optimize edilmiştir. Bunlar sırasıyla tahmin edici değerler, rastgele orman değerinin Max. Derinliği, Min. Bölme sayısı ve Min. Yaprak sayısından oluşmaktadır. Tahmin edici değerler 10 ila 1000 arasında, Max. Derinliği 10 ila 1000 arasında, Min. Bölme sayısını 2 ila 100 arasında ve Min. Yaprak sayısı 1 ila 100 arasında optimize edilmiştir. Elde edilen en iyi ve varsayılan sonuçlar tablo 5'te gösterilmiştir.

Tablo 5. Random Forest Algortiması Sonuç Tablosu

Random Forest						
Çalışma Süresi		Tahmin edici	Maks. Derinlik	Min. Bölme Sayısı	Min Yaprak Sayısı	Accuracy
3857 Saniye	Optimum	1000	1000	100	1	0,77604
	Varsayılan	10	10	2	1	0,725

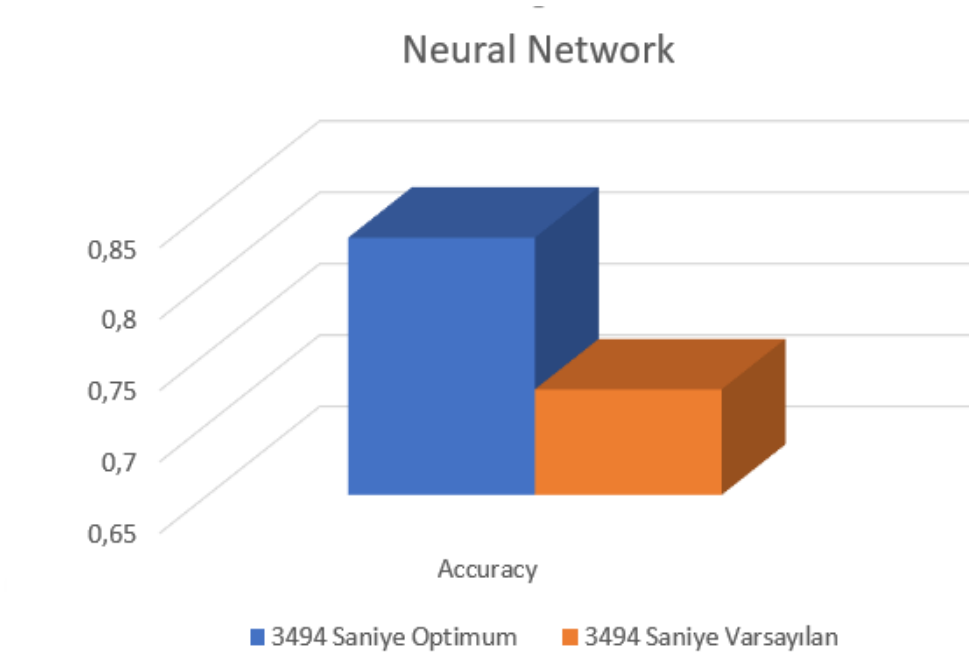


Şekil 45. Random Forest algoritması Optimum Başarı Grafiği

Neural Network algoritması uygulamada optimize etmek için belirlenen aralıklar öğrenme hızı 1e-4 ila 0,75 değerleri arasında, gizli katman sayısı 1 ila 5 arasında, gizli nöron sayısı 10 ila 512 arasında, aktivasyon türü Relu(doğrusallaştırılmış doğrusal ünite) model eğitiminin bir yinelemede (yani bir degrade güncellemesinde) kullanılan örnek kümesini optimize eden batch boyutu ise 1 ila 16 değerleri arasında seçilmiştir. Elde edilen optimim sonuç tablo 6'da gösterilmiştir.

Tablo 6. Neural Network Algortiması Sonuc Tablosu

Neural Network							
Çalışma Süresi		Learning Rate	Gizli Katman Sayısı	Gizli Nöron Sayısı	Aktivasyon Türü	Batch Boyutu	Accuracy
3494 Saniye	Optimum	0,0004773	4	413	Relu	6	0,8302
	Varsayılan						0,7239



Şekil 46. Neural Network Algoritması Optimum Başarı Grafiği

SONUÇ

İnternetin yaygınlaşmasıyla beraber veriye ulaşmak ve veriyi analiz etmek diğer zamanlara göre daha kolay bir hal almıştır. Ulaşılan verilerin nitelik ve niceliği önem kazanmakla beraber pek çok aşılması gereken soruna da yol açmıştır. Verinin kullanılacağı sahaya bağlı olarak, çok etiketli sınıflandırma, bir verinin ne kadar işe yaradığı ve o verinin tasnifinin mümkün olup olmadığı bakımından önem arz etmektedir. Özellikle son yıllarda metin kategorizasyonu, resim analizi, biyoenformatik, duyguya bağlı olarak müzik sınıflandırması gibi gerçek dünya verilerinde çok etiketli sınıflandırma etkin bir şekilde kullanılmaya başlanmıştır. Siyaset ile ilgili olan bir haberin aynı zamanda ekonomi ile etiketlenebilmesi çok etiketli sınıflandırma için örnek olarak gösterilebilir. Literatürde çok etiketli sınıflandırma ile ilgili problemleri çözmek için pek çok farklı yöntem önerilmiştir.

Çok etiketli bir sınıflandırma probleminde başarı sağlanabilmesi için problem dönüştürme yöntemleri ve algoritma adaptasyon yöntemleri olmak üzere iki farklı yol izlenmektedir. Problem dönüştürme yöntemleri çoklu etiket problemlerini bir dizi ikili sınıflandırma problemine dönüştürmektedir. Tek sınıflı sınıflandırıcılar kullanılarak problem dönüştürme yapılmaktadır. Algoritma adaptasyon yöntemleri ise algoritmaları doğrudan çoklu etiket sınıflandırması yapmak üzere uyarlanmaktadır. Başka bir ifadeyle, sorunu daha basit bir probleme dönüştürmek yerine sorunu tam manasıyla ele almaktadır (Nooney 2018).

Çok etiketli sınıflandırmada tartışılan ve kullanılan bu yöntemler, veri setlerinin verimli olmasının ve verilerin özelliklerine bağlı olarak sonucu etkileyebilir. Bu tezin amacı iki yönlüdür; ilk olarak teknolojinin gelişmesiyle artan verinin tasnifinde kullanılan ve son zamanlarda ilgi duyulan çok etiketli sınıflandırmayı tanıtmak, ikincisi olarak makine öğrenmesi algoritmalarını kullanarak çok etiketli sınıflandırmayı bir film veri seti üzerinde modelleyerek optimum başarının nasıl yakalayacağını test etmektir.

Veri setlerinde bulunan sınıf sayılarının, başarı oranının yüksek olmasında veya düşük olmasında etkin bir rol aldığı görülmektedir. Yapılan bu çalışmada film türlerinin sayısı fazlayken başarı oranının bir hayli düştüğü görülmektedir. Ve Sınıf

sayılarını azaltarak daha iyi bir sonuç elde edilebileceği görülmüştür. Sınıf sayısıyla beraber öznitelik sayısı gibi faktörlerde başarı oranını etkilemektedir.

Bunun yanında algoritmaların çalışma sürelerinde de gözle görülür bir fark meydana geldiği gözlemlenmektedir. Özellikle tablo 2, tablo 5 ve bu tabloların grafiklerinde de görüldüğü üzere SVM ve Random Forest algoritmalarında bu sürelerin daha fazla olduğu görülmektedir. Bu algoritmalarda sürenin uzun olmasının bir başka nedeni ise öznitelik sayısının fazla olmasından kaynaklanmaktadır. SVM ve Random Forest algoritmaları öznitelik bazlı işlem yapan algoritmalar olduğu bilinmektedir.

Yapılan çalışmada en iyi sonuçlar Logistic Regression ve Neural Network algoritmalarında elde edilmiştir. Logistic Regression algoritmasından elde edilen optimum başarı oranı 0,8305 olarak görülmektedir. Neural Network algoritmasından ise elde edilen başarı oranı 0,8302'dir. Bu çalışma için Logistic Regression ve Neural Network algoritmalarında görülen bu başarı her iki algoritma için farklı nedenlerden kaynaklanmaktadır. Özellikle Lojistik regresyon algoritması bir tahminde bulunmak için bağımsız tahmin edici yöntemler içeren doğrusal denklem kullanmaktadır. Ve bu veri seti bağımsız tahmin edici yöntemler için uygun bir veri seti olduğu analiz sonuçlarından anlaşılmaktadır. Neural Network algoritmasının iyi bir performans göstermesinin pek çok faktör olmasının yanında en önemli faktör olarak bu algoritmanın parametrik modeller ile çalışabilecek veri setlerini daha iyi analiz edebiliyor olmasıdır.

Verinin artması ve teknolojinin giderek gelişmesi bu alanda yapılacak yeni çalışma sahaları oluşturmaktadır. Kıymetli madenlerin kulanım alanına bağlı olarak çok etiketli sınıflandırılması çalışılacak konulardan biri olabilir.

KAYNAKLAR

Akay Ebru Çağlayan, "Ekonometride Yeni Bir Ufuk: Büyük Veri ve Makine Öğrenmesi", Social Sciences Research Journal, 7/2:(2018), s. 41-53).

Alan Mehmet Ali, "Karar Ağaçlarıyla Öğrenci Verilerinin Sınıflandırılması", Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi, 28/4: (2014), s.103

Alan Mehmet Ali, "Sivas Erzincan Kalkınma Projesi (SEKP) Verilerinin Veri Madenciliği ile Sınıflandırılması ve Kümelenmesi", Manas Sosyal Araştırmalar Dergisi, 03/10: (2014), s.132

Alatlı Betül ve Şenel Selma, "Lojistik Regresyon Analizinin Kullanıldığı Makaleler Üzerine Bir İnceleme", Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi, 5/1: (2014), 35-52.

Aly Mohamed (2005), "Survey on Multiclass Classification Methods", Erişim tarihi 27 kasım 2019, http://citeseerx.ist.psu.edu/viewdoc/download?doi= 10.1.1.175.107&rep=rep1&type=pdf adresinden erişildi.

Analıytıc Vıdhya (2015), "7 Regression Techniques you should know", Erişim Tarihi 15 Aralık 2019, https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression/ adresinden erişildi.

Atalay Muhammet ve Çelik Enes, "Büyük Veri Analizinde Yapay Zekâ Ve Makine Öğrenmesi Uygulamaları", 9/22: (2017). S. (155-172)

Aydın Can, "Makine Öğrenmesi Algoritmaları Kullanılarak İtfaiye İstasyonu İhtiyacının Sınıflandırılması", European Journal of Science and Technology, 14: (2018). s. (169-175)

Aydın Sinan ve Özkul Ali Ekrem, "Veri Madenciliği ve Anadolu Üniversitesi Açık öğretim Sisteminde Bir Uygulama", Journal of Research in Education and Teaching, 4/3: (2015), s.38

Bara Kassam (2018), "Linear Regression Essentials in R" ErişimTarihi: 25 Kasım 2019, http://www.sthda.com/english/articles/40-regression-analysis/165-linear-regression-essentials-in-r/ adresinden erişildi.

Berkeley, C. Edmund (1949). GIANT Brains, Erişim Tarihi :10 Ocak 2019,https://books.google.com.tr/books?id=VhSU4JffJB0C&pg=PA56&lpg=PA56&dq=edmunt+berkeley+grant+brains&source=bl&ots=8vOrANDFEF&sig=ACfU3U0T9P5QHHAc71xjQil0NNaLf-aefA&hl=tr&sa=X&ved=2ahUKEwitsf6zg_vmAhXwwcQBHXhLBV0Q6AEwD3oECAsQAQ#v=onepage&q=edmunt%20berkeley%20grant%20brains&f=false adresinden erişildi.

Birbil İlker (2018), "Tahmin ve Çıkarım (Makine Öğrenmesi? Yapay Öğrenme?" Erişim tarihi: 21 Temmuz 2019, http://www.veridefteri.com/2018/04/04/tahmin-ve-cikarim-makine-ogrenmesi-yapay-ogrenme/ adresinden erişildi.

Birbil İlker (2018), "Tahmin ve Çıkarım 2 – Performans Ölçümü", Erişim Tarihi: 28 Temmuz 2019, http://www.veridefteri.com/2018/04/13/tahmin-ve-cikarim-2-performans-olcumu/ adresinden erişildi.

Bircan Hüdaverdi, "Lojistik Regresyon Analizi: Tıp Verileri Üzerine Bir Uygulama", Kocaeli Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, 2:(2004), s. (185-208).

Börteçin Ege, "Rastlantının Bittiği Yer Big Data", Bilim ve Teknik, 550: (2015), s. (22-26).

Brownlee Jason (2017), "Difference Between Classification and Regression in Machine Learning" Erişim Tarihi 29 Eylül 2019, https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/ adresinden erişildi.

Büber Ebubekir (2018), "Günlük Yaşamda Makine Öğrenmesi Uygulama Örnekleri", Erişim Tarihi: 25 Ekim 2019, https://medium.com/@EbubekirBbr/g%C3%BCnl%C3%BCk-ya%C5%9Famda-makine-%C3%B6%C4%9Frenmesi-uygulama-%C3%B6rnekleri-3aba1cd49f36 adresinden erişildi.

Can Kaan (2019), "Deep Learning Tutorial for Beginners", Erişim Tarihi: 20 Ağustos 2019, https://www.kaggle.com/kanncaa1/deep-learning-tutorial-for-beginners adresinden erişildi.

Chen Jingnian, HUANG Houkuan, TIAN Shenggreng ve QU Youli, “Feature selection for text classification with Naïve Bayes”, Expert Systems with Applications, C: (2009), S. (5433-5434)

Cherman Everton Alvares, MONARD Maria Carolina ve METZ Jean (2011), “Multi-label Problem Transformation Methods: a Case Study”, Erişim Tarihi: 25 Ekim 2019 CLEIElectronicJournalhttp://www.scielo.edu.uy/scielo.php?script=sci_arttext&pid=S0717-50002011000100005#x1-4001r1 adresinden erişildi.

Cherman Everton Alvares, MONARD Maria Carolina ve Metz Jean (2011) “Multi-label Problem Transformation Methods: a Case Study” Erişim Tarihi: 25 Ekim 2019 CLEIElectronicJournalhttp://www.scielo.edu.uy/scielo.php?script=sci_arttext&pid=S0717-50002011000100005#x1-4001r1 adresinden erişildi.

Corry Maklin (2019), “ome pros and cons of Hierarchical Clustering”, Erişim Tarihi : 18 eylül 2019, https://towardsdatascience.com/machine-learning-algorithms-part-12-hierarchical-agglomerative-clustering-example-in-python-1e18e0075019 adresinden erişildi.

Çamurcu Ali Yılmaz ve IŞIK Meltem, “K-means, k-medoıds ve bulanık c-means algoritmalarının uygulamalı olarak performanslarının tespiti”, İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 6/11: (2007), s.(31-45).

Daı Honghua, Srikant Ramakrishnan, Zhang Chengqi (20, “Advances in Knowledge Discovery and Data Mining”, Pacific-Asia Conference, (2004) s. (22-30).

Delibaşoğlu İbrahim (2016), “hiyerarşik kümeleme”, Erişim Tarihi :18 Eylül 2019 Erişim adresi.http://ibrahimdelibasoglu.blogspot.com/2016/04/hiyerarsik-kumeleme.html adresinden erişildi.

Deutschman Zuzanna (2019), “Multi Label Text Classification”, Erişim Tarihi 31 Aralık 2019, https://towardsdatascience.com/multi-label-text-classification-5c505fdedca8 adresinden erişildi.

Dinçer Esra, "Veri Madenciliğinde K-Means Algoritması Ve Tıp Alanında Uygulanması", Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Kocaeli, S.101 : (2006).

Dıng Chris, WANG Hua, HUANG Heng, "Multi-label Linear Discriminant Analysis", Springer-Verlag Berlin Heidelberg, 63/16: (2010), s. (126-139).

Drakos Georgios (2018), "Cross-Validation", Erişim Tarihi : 23 aralık 2019 https://medium.com/@george.drakos62/cross-validation-70289113a072? Adresinden erişildi.

Enrıquez Fernando, TROYANO José A. ve LÓPEZ-Solaz Tomás, "An approach to the use of word embeddings in an opinion classification task". *Expert Systems with Applications* 66 (Supplement C),1: (2016).

Faggella Daniel (2019), **"**What is Machine Learning?", Erişim tarihi : 15 temmuz 2019, https://emerj.com/ai-glossary-terms/what-is-machine-learning/ adresinden erişildi.

Fumo David (2017), "Types of Machine Learning Algorithms You Should Know", Erişim Tarihi: 14 Kasım 2019, https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861 adresinden erişildi.

Gayar Neamet, Kittler Josef, ROLİ Fabia "Multiple Classifier Systems" Multiple Classifier Systems: 9th International Workshop, y: (2010), S. (14-15).

Geeksforgeeks (2019), "Cross Validation in Machine Learning", Erişim Tarihi :23 aralık 2019, https://www.geeksforgeeks.org/cross-validation-machine-learning/ adresinden erişildi.

Godbole Shantanu ve SARAWAGİ Sunita, "Discriminative Methods for Multi-labeled Classification", Springer-Verlag Berlin Heidelberg, 3056: (2004), s.(22-30).

Gök Murat, "Makine Öğrenmesi Yöntemleri İle Akademik Başarının Tahmin Edilmesi", Gazi üniversitesi Fen Bilimleri dergisi, C: Tasarım ve Teknoloji 5/3: (2017), s. (139 – 148).

Guru (2019), “Supervised vs Unsupervised Learning: Key Differences”, Erişim Tarihi 15 Kasım 2019, https://www.guru99.com/supervised-vs-unsupervised-learning.html adresinden erişildi.

Güner Necdet ve ÇOMAK Emre, “Mühendislik Öğrencilerinin Matematik I Derslerindeki Başarısının Destek Vektör Makineleri Kullanılarak Tahmin Edilmesi”, Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 17/2: (2010), s. (87-96).

Hamzaçebi Coşkun ve KUTAY Fevzi, “Yapay Sinir Ağları ile Türkiye Elektrik Enerjisi Tüketiminin 2010 Yılına Kadar Tahmini”, Gazi Üniv. Müh. Mim. Fak. Der. 19/3: (2004), s. (227-233).

Hatipoğlu Ekrem(2018), “Machine Learning — Clustering (Kümeleme)— K-Means Algorithm —Hierarchical Clustering (Hiyerarşik Kümeleme)— Part 13”, Erişim Tarihi : 19 eylül 2019 https://medium.com/@ekrem.hatipoglu/machine-learning-clustering-k%C3%BCmeleme-k-means-algorithm-part-13-be33aeef4fc8 adresinden erişildi.

Irmak Sezgin, ERCAN Uğur, “Karar Ağaçları Kullanılarak Türkiye Hanehalkı Zeytinyağı Tüketimi Görünümünün Belirlenmesi”, Uluslararası Yönetim İktisat ve İşletme Dergisi, 13/3: (2017), S. (553-564).

Işık Fatih (2019), “ülkelere göre kişi Başına Düşen Güvenlik Kamera Dağılımı” Erişim tarihi 20 Aralık 2019, https://www.technopat.net/2019/12/13/ulkelere-gore-kisi-basina-dusen-guvenlik-kamerasi-dagilimi-aciklandi/ adresinden erişildi.

Işık Yunus Emre (2018), “Otomatik Doküman Özetleme Yöntemlerinin Karşılaştırılması”, Sosyal Bilimler Enstitüsü, Yüksek Lisans Tezi, Sivas, Erişim Tarihi :15 Kasım 2019. 2018, s.(22-23).

Jaın Shubham (2017), “Solving Multi-Label Classification problems (Case studies included)”, Erişim Tarihi 1 ağustos 2019, https://www.analyticsvidhya.com/blog/2017/08/introduction-to-multi-label-classification/ adresinden erişildi.

Javapoınt (2019), “Simple Linear Regression in Machine Learning”, Erişim Tarihi: 17 Aralık 2019, https://www.javatpoint.com/simple-linear-regression-in-machine-learning adresinden erişildi.

Javatpoınt (2019), “Regression Analysis in Machine learning”, Erişim Tarihi : 20 Eylül 2019, https://www.javatpoint.com/regression-analysis-in-machine-learning adresinden erişildi.

Kaan Can (2018), “Data ScienceTutorial for Beginners”, Erişim Tarihi : 20 Ağustos 2019, https://www.kaggle.com/kanncaa1/data-sciencetutorial-for-beginners adresinden erişildi.

Kanj Sawsan (2017), “Learning Methods for Multi-label Classification”, Erişim Tarihi: 25 Ağustos 2019, Erişim adresi https://hal.archives-ouvertes.fr/tel-01435796/document adresinden erişildi.

Kılınç Deniz, Borandağ Emin, Yücalar Fatih, Tunalı Volkan, Şimşek Macit, Özçift Akın, (2016), “KNN Algoritması ve R Dili ile Metin Madenciliği Kullanılarak Bilimsel Makale Tasnifi”, Marmara Fen Bilimleri Dergisi, 3/1: (2016), s. (89-94).

Kırmızıbiber Abdullah ve Gökgöz Türkay (2019), “Hiyerarşik ve K-Ortalamalar Yöntemleriyle Grid Noktalarının Kümelenmesi”, TMMOB Harita ve Kadastro Mühendisleri Odası, 17. Türkiye Harita Bilimsel ve Teknik Kurultayı 25-27 Nisan 2019 Ankara.

Kızılkaya Yusuf Murat ve Oğuzlar Ayşe, “Bazı denetimli öğrenme algoritmalarının r programlama dili ile kıyaslanması”, journal of socıal ınquıry, y : (2017), s. (135-150).

Kızrak Ayyüce (2019), “Derin Öğrenme İçin Aktivasyon Fonksiyonlarının Karşılaştırılması”, Erişim Tarihi :27 kasım 2019, https://medium.com/@ayyucekizrak/derin-%C3%B6%C4%9Frenme-i%C3%A7in-aktivasyon-fonksiyonlar%C4%B1n%C4%B1n-kar%C5%9F%C4%B1la%C5%9Ft%C4%B1r%C4%B1lmas%C4%B1-cee17fd1d9cd adresinden erişildi.

Lıao Chia When, PERNG Yeng Horng (2007), “Data mining for occupational injuries in the Taiwan construction industry”, Safety Science, 46/7: (2008), s. (1091-1102).

Madjarov Gjorgji, Kocev Dragi, Gjorgjevikj Dejan, Dzeroski Saso (2012), “An extensive experimental comparison of methods for multi-label learning”, Pattern Recognition, 45/9 (2012), S. (3084-3104).

Mckınsey (2019), “An Executives Guide to Machine Learning”, Erişim Tarihi 21 Eylül 2019, https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/an-executives-guide-to-machine-learning adresinden erişildi.

Mıcrosoft (2019), “Makine Öğrenmesi nedir?”, Erişim Tarihi : 10 Aralık 2019, https://azure.microsoft.com/tr-tr/overview/what-is-machine-learning-platform/ adresinden erişildi.

Monard Maria Carolina, CHERMAN Everton Alvares, METZ Jean , **“**Multi-label Problem Transformation Methods: a Case Study”, CLEI Electronic Journal, 717/5000: (2011).

Nooney Kartik (2018), “Deep dive into multi-label classification..! (With detailed Case Study)”, Erişim Tarihi 10 Ocak 2019, https://towardsdata science.com/journey-to-the-center-of-multi-label-classification-384c40229bff adresimden erişildi.

Nvıdıa (2019), “Machine Learning and Analytıcs”, Erişim Tarihi 21 Eylül 2019, https://developer.nvidia.com/machine-learning adresinden erişildi.

Orhunbilge Neyran, “Uygulamalı Regresyon ve Korelasyon Analizi” Erişim Tarihi: 10 kasım 2019, http://w3.balikesir.edu.tr/~bsentuna/wpcontent /uploads/2013/03/Regresyon-Analizi.pdf adresinden erişildi.

Robert Schapıre, Yoram Sınger (2000), “BoosTexter: A Boosting-based System for Text Categorization”, Erişim Tarihi : 27 Ağustos 2019, https://www.cis.upenn.edu/~mkearns/finread/boostexter.pdf adresinden erişildi.

Roman Victor (2019), “Unsupervised Machine Learning: Clustering Analysis”, Erişim Tarihi :21 Eylül 2019, https://towardsdatascience.com/unsupervised-machine-learning-clustering-analysis-d40f2b34ae7e adresinden erişildi.

Sarıman Güncel, “Veri Madenciliğinde Kümeleme Teknikleri Üzerine Bir Çalışma: K-Means ve K-Medoids Kümeleme Algoritmalarının Karşılaştırılması”, Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü Dergisi, 15/3 :(2011), s.(192-202).

Schratz Patrick (2018), “Multilabel Clasification”, Erişim Tarihi :20 kasım 2019, https://mlr.mlr-org.com/articles/tutorial/multilabel.html adresinden erişildi.

Srınıdhı Sunny (2018), “Label Encoder vs. One Hot Encoder in Machine Learning”, Erişim tarihi :16 aralık 2019, https://medium.com/@contactsunny/label-encoder-vs-one-hot-encoder-in-machine-learning-3fc273365621 adresinden erişildi.

Standford (2019), “machine Learning”, Erişim Tarihi : 21 Eylül 2019, https://online.stanford.edu/courses/cs229-machine-learning adresinden erişildi.

Stecanella Bruno (2019), ”Machinle Learning TF-IDF”, Erişim tarihi 19 aralık 2019, https://monkeylearn.com/blog/what-is-tf-idf/ adresinden erişildi.

Şahin murat (2018), “ML Linear Regression (Theory)”, Erişim tarihi 14 aralık 2019, https://veribilimi.net/linear-regression/ adresinden erişildi.

Şanlı Erdem (2018), “Yapay Sinir Ağı Kontrollü Otonom Rc Araç Uygulaması”, Erişim Tarihi 17 kasım 2019, http://acikerisim.gelisim.edu.tr /xmlui/bitstream/handle/11363/1181/529394.pdf?sequence=1&isAllowed=y adresinden erişildi.

Şekerli, Eyüp Bayram (2019), “Ticari Havayolu Taşımacılığı Sektöründe Makine Öğrenmesi Uygulamalarının İncelenmesi” Erişim Tarihi 04 Ocak 2019 Selçuk Üniversitesi Sosyal Bilimler Meslek Yüksekokulu Dergisi, 22/2: (2019), s. (405-419).

Tahir Muhammad Atif, Kittler Josef, Bouridane Ahmed (2012), “Multilabel Classification Using Heterogeneous ensemble of Multi-label Classifiers”

Erişim Tarihi 16 kasım 2019, https://fdocuments.in/document/multilabel-classification-using-heterogeneous-ensemble-of-multi-label-classifiers.html adresinden erişildi.

Takaoğlu Mustafa, Takaoğlu Faruk, "k-means ve hiyerarşik kümeleme algoritmanın weka ve matlab platformlarında karşılaştırılması", istanbul aydın üniversitesi dergisi c11,s3: (2019), s. (303-317).

Tsoumakas Grigorios ve Vlahavas Ioannis, "Random k-Labelsets: An Ensemble Method for Multilabel Classification", Erişim Tarihi : 18 Eylül 2019 c, 4701 : (2007), s. (406-417).

Tutorıalspoınt (2019), **"**Machine Learning - Logistic Regression" Erişim Tarihi: 3 ekim 2019, https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_classification_algorithms_logistic_regression.htm adresinden erişildi.

Ulgen E. Kaan (2017), **"**Makine Öğrenimi Bölüm-4 (Destek Vektör Makineleri)", Erişim Tarihi :19 Aralık 2019, https://medium.com/@k.ulgen90/makine-%C3%B6%C4%9Frenimi-b%C3%B6l%C3%BCm-4-destek-vekt%C3%B6r-makineleri-2f8010824054 adresinden erişildi.

Usman Malik (2018), "Hierarchical Clustering with Python and Scikit-Learn", Erişim Tarihi :19 eylül 2019, https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/ adresinden erişildi.

Utku Anıl, AKCAYOL M. Ali (2019), "Akan Veri Karakterizasyonu, Üretimi ve Analitiği Üzerine Kapsamlı Bir İnceleme", Erzincan üniversitesi Fen Bilimleri Enstitüsü Dergisi, 12/1: (2019), s. (379-410).

Valdovınos Rosa Maria ve SANCHEZ Josep Salvador, "Combining Multiple Classifiers with Dynamic Weighted Voting", Conference: Hybrid Artificial Intelligence Systems, 4th International Conference, (2009), s. (10-12).

Vector Space Model - Wikipedia (2013), Erişim Tarihi 22 Kasım 2019 tarihinde *https://en.wikipedia.org/wiki/Vector_space_model* adresinden erişildi.

Yılmaz Veysel (2019), “Finansmanı Öğrenen Makinalar” Erişim Tarihi”, SETSCI Conference Proceedings 4/8 : (2019), s.(187-192).

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı:	Mikail BARAN
Uyruğu:	T.C.
Doğum Tarihi ve Yeri:	01.12.1990 GERGER
e-posta:	mikailbaran7@gmail.com

EĞİTİM

Derece	Kurum	Mezuniyet Yılı
Yüksek Lisans	Cumhuriyet Üniversitesi	2020
Lisans	Cumhuriyet Üniversitesi	2016

İŞ TECRÜBESİ

Tarih	Kurum	Görev
11.2016 - Devam Ediyor	Kuveyttürk Katılım Bankası	Operasyon
05.2016 - 11.2016	Fidan Yazıcıoğlu kültür merkezi	Bilgi işlem birimi

YABANCI DİL BİLGİSİ

Yabancı Dilin Adı	YÖKDİL(52,75)