

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÖLÇÜLEN KİMYASAL GAZLARIN MAKİNA
ÖĞRENMESİ İLE SINIFLANDIRMASI

YÜKSEK LİSANS TEZİ

Safa EI BEKRI

Enstitü Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ

Tez Danışmanı : Prof. Dr. Cemil ÖZ

Şubat 2020

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ


ÖLÇÜLEN KİMYASAL GAZLARIN MAKİNE
ÖĞRENMESİ İLE SINIFLANDIRILMASI

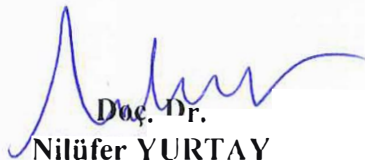
YÜKSEK LİSANS TEZİ


Safa El BEKRI

Enstitü Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ

Bu tez 28/02/2020 tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile kabul edilmiştir.


Prof. Dr.
Cemil ÖZ
Jüri Başkanı


Doç. Dr.
Nilüfer YURTAY
Üye


Dr. Öğr. Üyesi
İsmet KANDİLLİ
Üye

BEYAN

Bu tezdeki tüm çalışmaların akademik kurallarda kendimce yapıldığını, tüm görsel ve yazılı bilgilerin ve sonuçların akademik ve etik kurallara uygun olarak sunulduğunu, diğer kişilerin çalışmalarından yararlanılmasında sunulan verilerde herhangi bir bozulma olmadığını beyan ederim. Bilimsel normlara uygun şekilde yönlendirildiler.

Safa EL BEKRI
28/02/2020

TEŐEKKÜR

Rahman ve Rahim olan Allah'ın adıyla,öncelikle bana vermiş olduđu sađlık ve iş yapabilme yeteneđi verdiđi için ona sonsuz Őükürler diliyorum, burada çalışmamı mümkün kıldıđı için Türkiye burs programına çok minnettarım. Bu yüksek lisans konusunun tamamlanmasına katkıda bulunan herkese ve tez savunma jürimde yer alan jüri üyelerine teşekkür ederim.

Sakarya Üniversitesi Bilgisayar ve Bilişim Bilimleri Fakültesi Bilgisayar Mühendisliđi Bölümündentez danışmanlıđımı yürüten Profesör Dr. Cemil Öz'e tez çalışmam boyunca eğitim önerileri ve teknik tavsiyeleri için teşekkür ederim.

Bu güne kadar olduđu gibi Sakarya Üniversitesindeki yüksek lisans çalışmam boyunca beni teşvik eden, madi ve manevi desteđini esirgemeyen Anne, babam ve kardeşlerime, Sakarya üniversitesinin ilgili birimlerinde ve FBE çalışanlara teşekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR	i
İÇİNDEKİLER	ii
SİMGELER VE KISALTMALAR LİSTESİ	iv
ŞEKİLLER LİSTESİ	v
TABLolar LİSTESİ	vi
ÖZET	Vii
SUMMARY	Vii
BÖLÜM 1.	
GİRİŞ	1
1.1. Amaç	2
1.2. Kapsam	3
BÖLÜM 2.	
ENDÜSTRİ 4.0	4
2.1. Giriş	4
2.2. Endüstride Kullanılan Teknolojiler 4.0	4
2.3. Endüstri 4.0 Modelini Benimsemenin Faydaları	5
2.3.1. Geliştirilmiş verimlilik	5
2.3.2. Verimliliği arttırmak	5
2.3.3. Artan bilgi paylaşımı ve ortak çalışma	6
2.3.4. Maliyetleri düşürür	6
2.3.5. Daha yüksek gelirler	6
2.3.6. Esneklik ve çeviklik	6
2.4. Yapay Zeka ve Endüstri 4.0	7

2.5. Endüstri 4.0'da Makine Öğrenimi	9
2.5.1. Öngörücü bakım	10
2.5.2. Üretimde otonom araçlar ve interaktif makineler	10
2.5.3. Kalite kontrol	10
BÖLÜM 3.	
GAZ SENSÖRLERİ	12
3.1. Sensör	12
3.2. Gaz Sensörü	12
3.3. Farklı Gaz Sensörü Türleri	14
3.4. Gaz Sensörlerinin Uygulamaları	14
3.5. Gaz Sürüklenme Veri Seti	14
BÖLÜM 4.	
KULLANILAN SINIFLANDIRMA ALGORİTMASI	16
4.1. Giriş	16
4.2. Makine Öğrenmesi	16
4.2.1. Tanım	16
4.2.2. Makine öğrenimi adımları	16
4.2.3. Makine öğrenimi ihtiyacı	17
4.2.4. Makine öğrenimi türleri	18
4.3. Sınıflandırma Algoritması	19
4.3.1. Rastgele orman	20
4.3.2. K-en yakın komşular algoritması	25
4.3.3. Destek vektör makinesi	32
4.3.4. Lojistik regresyon	40
BÖLÜM 5.	
GERÇEK ÖRNEK.....	46
5.1. Giriş	46
5.2. Arduino DUE	46
5.2.1. Arm çekirdek faydaları	46

5.2.2. Teknik özellikler	47
5.2.3. Güç	47
5.2.4. Hafıza	48
5.2.5. Giriş ve çıkış	48
5.2.6. Haberleşme	49
5.3. Co2 Gaz Sensörü CDM4161a	49
5.3.1. Özellikleri	49
5.3.2. Giriş çıkış sinyali	50
5.4. Tera Term	51
5.5. Pratik Kısmı	51

BÖLÜM 6.

SENSOR GAZLARIN MAKİNE ÖĞRENME SİNİFLANDIRILMASI	56
6.1. Giriş	56
6.2. Weka	56
6.2.1. Weka veri formatları	56
6.2.2. Weka sınıflandırması	57
6.3. Veri Kümesi	58
6.4. WEKA'da Sınıflandırma Adımları	59
6.4.1. ZeroR	60
6.4.2. Rastgele orman	61
6.4.3. K-en yakın komşular	62
6.4.4. Lojistik regresyon	63
6.4.5. Naif bayes	64
6.4.6. Karar ağacı: (rastgele ağaç)	65
6.4.7. Destek vektör makinesi	66
6.5. Sonuç Tablosu	67

BÖLÜM 7.

TARTIŞMA VE SONUÇ	68
-------------------------	----

KAYNAKLAR	70
ÖZGEÇMİŞ	74



SİMGELER VE KISALTMALAR LİSTESİ

DVM : Destek vektör makinesi

KYK : k-en yakın komşular

LR : Lojistik regresyon

RO : Rastgele orman



ŞEKİLLER LİSTESİ

Şekil 4.1. Makine Öğrenmesi Adımı	16
Şekil 4.2. Rastgele seçim özellikleri sürecini göstermektedir [22]	22
Şekil 4.3. KNN örnek sınıflandırması	28
Şekil 4.4. Örnek sınıflandırması[25]	30
Şekil 4.5. SVM ile sınıflandırma [2]	33
Şekil 4.6. Doğrusal SVM [3]	35
Şekil 4.7. Doğrusal SVM [3]	35
Şekil 4.8. Doğrusal SVM [3]	36
Şekil 4.9. Doğrusal Olmayan SVM [3]	36
Şekil 4.10. Doğrusal Olmayan SVM [3]	37
Şekil 4.11. Doğrusal Olmayan SVM [3]	37
Şekil 4.12. Doğrusal Olmayan SVM [3]	38
Şekil 4.13. Lojistik fonksiyonu [36]	41
Şekil 4.14. Sigmoid aktivasyon fonksiyonu [38]	42
Şekil 5.1. Arduino ve CO2 sensor.....	52
Şekil 5.2. Arduino ide kodu.....	53
Şekil 5.3. Tera term veri çıkmak için ayarlar.....	54
Şekil 5.4. Csv uzantılı dosya kaydetme.....	54
Şekil 5.5. Excelde veri seti.....	55
Şekil 6.1. ARFF biçimi	57
Şekil 6.2. Gaz sensörü veri seti yüklü weka explorer arayüzü	58
Şekil 6.3. Weka tek değişkenli öznitelik dağıtım grafikleri	59
Şekil 6.4. Gaz sensörü veri setindeki ZeroR algoritması için weka sonuçları	60
Şekil 6.5. Zeror algoritmasının karışıklık matrisi	61
Şekil 6.6. Rasgele orman algoritmasının sonuç yürütülmesi	61
Şekil 6.7. Rastgele Ormanı algoritmasının karışıklık matrisi	62

Şekil 6.8. Gaz sensörü veri setindeki K- en yakın komşular algoritması için weka sonuçları	62
Şekil 6.9. K- en yakın komşular algoritmasının karışıklık matrisi	63
Şekil 6.10. Gaz sensörü veri setindeki lojistik regresyon algoritması için weka sonuçları	63
Şekil 6.11. Lojistik regresyon algoritmasının karışıklık matrisi	64
Şekil 6.12. Gaz sensörü veri setindeki naive bayes algoritması için weka sonuçları	65
Şekil 6.13. Gaz sensörü veri setindeki rastgele ağaç algoritması için weka sonuçlar	66
Şekil 6.14. Gaz sensörü veri setindeki Destek vektör makine algoritması için weka sonuçlar	66

TABLolar LİSTESİ

Tablo 4.1. Vaka veri kümesi KNN kullan	31
Tablo 6.1. K'nin farklı değeri	63
Tablo 6.2. Sonuç tablosu.....	67



ÖZET

Anahtar kelimeler: Gaz sensörleri, Sınıflandırma algoritması, Weka, Arduino.

Bu tez çalışmasında amacı endüstri 4.0'ın temel konularından olan gömülü sistem yazılım ve donanımlarını kullanarak elde edilen sensör verilerini makine öğrenmesi algoritmaları ile sınıflandırmak ve sürdürülebilir üretim için gerekli otomasyonun sağlanması için alt yapı oluşturmaktır. Günümüzde endüstride kullanılan tüm aygıtlar akıllı hale geliyor ve üretim sürecinin tüm seviyelerinde üretilen dijital veriler ürün kalitesini, esnekliğini ve fonksiyonelliğini artırmak için kullanılmaktadır. Üretim tesislerinden gelen veri sensörlerle toplanmaktadır. Büyük veri tıfadesi ile bahsedilen bu verilerin insan kullanıcıların değerlendirebilmesi mümkün değildir. Bu veriyi makine öğrenmesi algoritmaları ile değerlendirmek, anlamlandırmak ve sürdürülebilir üretim için kullanmak ancak mümkün olabilir. Bu çalışmada endüstride gerek insan hayatını doğrudan etkilemesi ile gerek ise birçok ürünün üretiminde ortaya çıkması veya kullanmasından dolayı gazların algılanması ve sınıflandırılması üzerinde durulmuştur. Gaz sensörleri, insan sağlığını ve özelliklerini tehdit eden yanıcı, yakıcı ve toksik gazları tespit etmek için endüstride ve yangınla mücadelede yaygın olarak kullanılmaktadır. Önce arduia ile b gaz verilerini toplama ve makine öğrenmesi ile sınıflandırılması üzerine çalışılmıştır. Gazlar ile ilgili yeterince veri elde edilemediği için altı farklı uçucu organik gaz için bir veri kümesi veri seti indirilerek onun üzerinde çalışmalar yapılmıştır. Bu veri setinde; Amonyak, Asetaldehit, Aseton, Etilen, Etanol ve Toluen gazları bulunmaktadır. Üç yıllık bir süre boyunca, 16 metal oksit gaz sensörü kullanılarak elde edilmiştir. Bu veriler 13910 ölçüm ve 129 özellik içerir. Çalışmamızda bu verileri sınıflandırmak için çok sayıda makine öğrenmesi algoritmaları kullanılmıştır. Bunlar; k-en yakın komşular, Destek vektör makinesi, Rastgele orman ve Lojistik regresyon dur. Verileri sınıflandırmak için bu makine öğrenme algoritmaları Weka programında gerçekleştirilmiştir. Yapılan çalışmalar sonucunda k-en yakın komşular algoritmasının en iyi sonucu verdiğini ve sınıflandırma başarısı% 99,48'dır.

PRODUCTION WITH ARTIFICIAL INTELLIGENCE IN INDUSTRY 4.0

SUMMARY

Keywords: Gas sensors, Classification algorithm, Weka, Arduino.

In this thesis, the purpose of this study is to classify the sensor data obtained by using embedded system software and hardware, which is one of the fundamental issues of industry 4.0, with machine learning algorithms and to create the infrastructure for the necessary automation for sustainable production. Today, all devices used in the industry become smart and digital data produced at all levels of the production process is used to increase product quality, flexibility and functionality. Data from production facilities are collected with sensors. It is not possible for human data to be evaluated by these data, which are mentioned with the definition of big data. It is only possible to evaluate this data with machine learning algorithms, to make sense of it and to use it for sustainable production. In this study, the perception and classification of gases are emphasized in the industry both because it directly affects the human life and because it occurs or uses many products. Gas sensors are widely used in industry and fire fighting to detect flammable, combustible and toxic gases that threaten human health and properties. Firstly, it was studied on collecting gas data with Arduino and classifying it with machine learning. Since sufficient data about gases could not be obtained, a dataset was downloaded for six different volatile organic gases and studies were carried out on it. In this dataset; There are ammonia, acetaldehyde, acetone, ethylene, ethanol and toluene gases. It was obtained over a three-year period using 16 metal oxide gas sensors. These data contain 13910 measurements and 129 features. In our study, many machine learning algorithms were used to classify this data. These; k-nearest neighbors, Support vector machine, Random forest and Logistic regression stop. These machine learning algorithms were implemented in the Weka program to classify the data. As a result of the studies, the k-nearest neighbors algorithm gives the best result and the classification success is 99,48%.

BÖLÜM 1. GİRİŞ

Ortamda belirli bir gazın varlığını ve çeşitli konsantrasyonlarını tespit edebilen gaz sensörlerini çevresel izleme, endüstri, tarımsal üretim, askeri, kamu güvenliği ve tıbbi teşhis gibi alanlar için son derece yararlıdır. Gaz sensörleri birçok uygulamada önemli bir rol oynamaktadır ve son çeyrek yüzyılda yoğun bir şekilde geliştirilmiştir. Bu özellikle otomotiv egzoz gazlarının (lambda probu) ve hava kalitesinin (AQ sensörleri) izlenmesi uygulamaları için geçerlidir [1].

Gaz sensörleri, insan sağlığını ve yaşam kalitesini tehdit eden yanıcı, yakıcı ve toksik gazları tespit etmek endüstride ve yangınla mücadelede yaygın olarak kullanılmaktadır. Çeşitli malzemeler (örn., Optik fiberler, konjüge edilmiş malzemeler, karbon malzemeler ve metal oksitler), hedef gazlarla spesifik etkileşimlerine dayanarak gaz sensörleri için algılama malzemesi olarak araştırılmıştır. Bunlar arasında, metal oksit (örn. TiO_2 , V_2O_5 , WO_3 , ZnO ve SnO_2) bazlı gaz sensörleri, düşük üretim maliyetleri ve kullanımlarının basitliği nedeniyle bilim adamlarının ve araştırmacıların sayısız ilgisini çekmiştir [2].

Metal oksit malzemeler tarafından gazların tespiti, algılanması esas olarak gazların emilimi üzerine iletkenlik (veya direnç) değişikliklerine dayanmaktadır [3]. Bu çalışmada, üç yıllık bir süre boyunca altı farklı uçucu organik bileşik için oluşturulmuş bir veri kümesi kullanılmaktadır [4]. Bu veri kümesinde 16 adet metal oksit gaz sensörü kullanılarak, gaz sensörleri ve gaz dereceleri sınıflandırılmıştır.

Endüstri 4.0 veya Nesnelerin İnterneti (IoT) ile birlikte sürdürülebilir üretimde sensörlerden toplanan veriler önemli hale gelmiştir. Üretimde elde edilen yoğun veri,

Büyük veri olarak isimlendirilmektedir. Büyük verini işlenmesi ve anlamlandırılması sürdürülebilir üretim için kullanılması ancak makine öğrenmesi algoritmaları ile mümkün olabilir. Her şey daha akıllı hale geliyor ve üretim sürecinin tüm seviyelerinde üretilen veriler ürün kalitesini, esnekliğini ve üretkenliğini artırmak için kullanılıyor.

Endüstride, sürdürülebilir üretim verileri üreten ve izleme sağlayan sensörlerle sağlanabilir. Bu sensörlerindeki değerlendirmeleri sonucunda sınıflandırma ve karar verme sağlanmakta sistemler akıllı hale gelmektedirler. Endüstri 4.0'nın temelinde bu oluşturmaktadır. [5].

1.1. Amaç

Bu çalışmanın amacı, sürdürülebilir üretim için gerekli verilerin toplanması ve sınıflandırılması için gömülü sistemleri kullanma, değişken veri seti için sınıflandırma yapmak için etkili makine öğrenmesi sınıflandırma algoritmalarını belirlemektir. Arduino gömülü sistemi ile veri toplanmış. Yeterli veri toplanmadığı için hazır veri seti kullanılmıştır.

Çalışmalarımızda sınıflandırma yapmak için çok sayıda makine öğrenmesi algoritmaları kullandık. Bunlar: k-en yakın komşular, destek vektör makinesi, rastgele orman, lojistik regresyon ve bize veri madenciliği görevleri için makine öğrenme algoritmalarıdır. Bu algoritmalar WEKA yazılımında gerçekleştirilmiştir. WEKA programı veri hazırlama, sınıflandırma, regresyon, kümeleme, ilişkilendirme, veri madenciliği ve görselleştirme için araçlar içerir.

1.2. Kapsam

Bu tez çalışmasında giriş bölümünden sonra, planlama aşağıdaki şekilde düzenlenmiştir: sırası ile ikinci bölümde, endüstri 4.0'ü, üçüncü bölüm, gaz sensörleri ve endüstride gaz sensörlerinin kullanımı ve sınıflandırmalarının önemi hakkında bilgi vermektedir. Dördüncü bölümde tezde kullanılan makine öğrenimi sınıflandırma algoritmaları detaylı olarak verilmiştir. Beşinci bölümde gerçek bir örnek verdik ve son bölümde test ve tartışma anlatılmaktadır.



BÖLÜM 2. ENDÜSTRİ 4.0

2.1. Giriş

Endüstri 4.0, Sanayi Devrimi'nde ağırlıklı olarak bağlantı, otomasyon, makine öğrenimi ve gerçek zamanlı verilere odaklanan yeni bir aşamayı ifade eder. Endüstri 4.0, IIoT veya akıllı üretim olarak da adlandırılır, üretim ve tedarik zinciri yönetimine odaklanan şirketler daha bütünsel ve daha iyi bağlantılı bir ekosistem oluşturmak için fiziksel üretim ve işlemleri akıllı dijital teknoloji, makine öğrenimi ve büyük verilerle birleştirir. Temel olarak endüstri 4.0, siber fiziksel sistemler (CPS), nesnelerin interneti (IoT), nesnelerin endüstriyel interneti (IIOT), bulut bilişim, bilişsel hesaplama ve yapay zeka.

2.2. Endüstride Kullanılan Teknolojiler 4.0

Gelişmiş dijital teknoloji zaten imalatta kullanılmaktadır, ancak Endüstri 4.0 ile üretimi dönüştürecektir. Tedarikçiler, üreticiler ve müşteriler ile insan ve makine arasındaki geleneksel üretim ilişkilerinin daha fazla verimliliğe ve değişmesine yol açacaktır. Dokuz teknoloji trendleri Endüstri 4.0'ın yapı taşlarını oluşturmaktadır [6]:

- Büyük veri ve analiz
- Bulut
- Siber güvenlik
- Arttırılmış gerçeklik
- Otonom Robotlar
- Nesnelerin İnterneti (IoT)
- Simülasyon ve Sanallaştırma
- Sistem entegrasyonu
- Katmanlı üretim

2.3. Endüstri 4.0 Modelini Benimsemenin Faydaları

Endüstri 4.0'ın avantajları arasında iyileştirilmiş verimlilik ve verimlilik, daha iyi esneklik ve çeviklik ve artan kârlılık sayılabilir. Endüstri 4.0 da müşteri deneyimini geliştirir. Akıllı Fabrika teknolojileri ilginç ve heyecan verici olsa da, Endüstri 4.0'ın faydaları her zaman tartışmanın merkezinde olmalıdır. Sonuçta, teknolojiye yaptığımız yatırımlar, gelişmiş üretim süreçleri veya gelişmiş sistemler bir getiri sağlamalıdır. Endüstri 4.0'ın üretim hatlarınıza ve işinize getirdiği başlıca faydalar:

2.3.1. Geliştirilmiş verimlilik

Basit bir ifadeyle, Endüstri 4.0 teknolojileri daha azıyla daha fazlasını yapmanızı sağlar. Başka bir deyişle, kaynaklarınızı daha düşük maliyetli ve verimli bir şekilde tahsis ederken daha fazla ve daha hızlı üretebilirsiniz.

Gelişmiş makine izleme ve otomatik / yarı otomatik karar verme özellikleri sayesinde üretim hatlarınız daha az arıza süresi yaşayacaktır. Aslında, tesisiniz bir Endüstri 4.0 Akıllı Fabrikası olmaya yaklaştıkça genel OEE (Genel Ekipman Verimliliği) gelişecektir [7].

2.3.2. Verimliliği arttırmak

Endüstri 4.0 ile ilgili teknolojiler sayesinde üretim hattınızın birden fazla alanı daha verimli hale gelecektir. Bu verimliliklerden bazıları yukarıda belirtilmiştir - daha az makine duruş süresi, daha fazla ürün üretme ve daha hızlı hale getirme yeteneği.

Geliştirilmiş verimliliğin diğer örnekleri arasında daha hızlı toplu değişimler, otomatik takip ve izleme süreçleri ve otomatik raporlama bulunur. NPI'ler (Yeni Ürün Tanıtımları) iş kararları ve daha fazlası gibi daha verimli hale gelir [7].

2.3.3. Artan bilgi paylaşımı ve ortak çalışma

Endüstri 4.0 teknolojileri, üretim hatlarınızın, iş süreçlerinizin ve departmanlarınızın konum, zaman dilimi, platform veya diğer faktörlerden bağımsız olarak iletişim kurmasını sağlar. Bu, örneğin bir tesiste makinedeki bir sensör tarafından öğrenilen bilgilerin kuruluşunuz genelinde yayılmasını sağlar. En iyisi, bunu herhangi bir insan müdahalesi olmadan otomatik olarak, yani makineden makineye ve sistemden sisteme yapmak mümkündür. Başka bir deyişle, bir sensörden gelen veriler, dünyanın herhangi bir yerinde bulunan birden fazla üretim hattında anında bir iyileştirme yapabilir [7].

2.3.4. Maliyetleri düşürür

Akıllı Fabrika olmak bir gecede ve kendi başına gerçekleşmez. Bunu başarmak için yatırım yapmanız gerekir, bu yüzden ön maliyetler vardır. Bununla birlikte, tesislerinizdeki üretim maliyeti, Endüstri 4.0 teknolojileri, yani otomasyon, sistem entegrasyonu, veri yönetimi ve daha fazlası nedenleriyle önemli ölçüde düşecektir.

2.3.5. Daha yüksek gelirler

Yukarıdaki noktaların çoğu, üretim tesisiniz için daha yüksek gelir sağlayabilir. Örneğin, üretim hattınızı tam olarak otomatikleştirerek ve diğer Endüstri 4.0 teknolojilerini uygulayarak, talep artışını karşılamak veya yeni bir sözleşmeyle rekabet etmek için minimum personel maliyetiyle yeni bir değişiklik ekleyebilirsiniz [7].

2.3.6. Esneklik ve çeviklik

Endüstri 4.0'ın avantajları arasında gelişmiş esneklik ve çeviklik de bulunmaktadır. Örneğin, bir Akıllı Fabrikada üretimi aşağı veya yukarı ölçeklendirmek daha kolaydır. Üretim hattına yeni ürünler tanıtmanın yanı sıra tek seferlik üretim

çalışmaları, yüksek karışım üretimi ve daha fazlası için fırsatlar yaratmak da daha kolaydır.

2.4. Yapay Zeka Ve Endüstri 4.0

Büyük veri ve yapay zeka Endüstri 4.0'a büyük bir destek veriyor. Akıllı yazılım çözümleri, daha sonra üretim süreçlerini daha verimli hale getirmek ve enerji tüketimini azaltmak için kullanılacak eğilimleri ve modelleri belirlemek için bir fabrika tarafından üretilen yüksek hacimli verileri kullanabilir. Tesisler sürekli olarak yeni koşullara adapte oluyor ve operatör girdisine ihtiyaç duymadan optimizasyondan geçiyor. Ağ oluşturma düzeyi arttıkça, AI yazılımı “satırlar arasında okumayı” öğrenebilir; bu da, henüz insan gözüyle görülmeyen veya artık belirgin olmayan sistemlerde birçok karmaşık bağlantının ortaya çıkmasına yol açabilir. Yeterince akıllı analitik teknolojiye sahip akıllı yazılım zaten mevcuttur. Ancak veri işlemenin bir bulut çözümü kullanılarak mı, yoksa yerel düzeyde mi (örneğin, Edge bilgi işlem kullanılarak) kullanıcının gereksinimlerine bağlı olacaktır.[8]

Yapay zeka çağımızda, süreçleri dönüştürmek için çok çeşitli endüstriyel uygulamalarda makine ve derin öğrenmeyi kullanıyor. Yapay zeka, endüstriyel dünyayı dönüştürme potansiyeline sahiptir - şu anda devam etmekte olan ve her geçen gün ivme kazanan bir harekettir -. Dünya çapında işletmeler, ürün geliştirme, üretimden tedarik zinciri ve saha operasyonlarına kadar temel endüstriyel süreçleri otomatikleştirmek ve yeniden icat etmek için AI güdümlü sistemleri kullanıyor. Bu eğilimin bir ismi bile var: endüstriyel yapay zeka. Adından da anlaşılacağı gibi endüstriyel AI, yapay zeka teknolojilerini karmaşık endüstriyel operasyonlarla gelen zorlukları uygulamakla ilgilidir [9].

Endüstriyel yapay zekanın gücünü göstermek için, bugün gerçek dünyada kullanılan bazı yenilikçi uygulamalara bakalım. Bunların hepsi, yüksek performanslı bilgi işlem sistemleri tarafından yönlendirilen yapay zekanın gücü olmadan mümkün olmayacak kullanım durumlarıdır.

Tasarım doğrulama süreçlerini optimize etme

Intel'in BT organizasyonu ve ürün geliştirme ekipleri, yeni işlemcilerin tasarımında zaman alan ürün doğrulama sürecini optimize etmek için yüksek oranda bağlı veri kümeleri ve yapay zeka kullanıyor. Bu AI uygulaması, Intel'in tasarım sürecinde daha önce ürün geliştirme sürecinde düzeltmeler yapmanın daha pahalı olacağı zamana kadar keşiften kaçabilecek gizli hataları keşfetmesine yardımcı oluyor. Şirket, AI'nın artık maliyetleri düşürmenin ve pazara sunma süresini hızlandırmanın anahtarlarından biri olduğunu söyledi [10].

Malzemelerin hareketini otomatikleştirme

Ontario merkezli bir şirket olan OTTO Motors, malzeme hareketi işlerini otomatikleştirmek için yapay zekanın gücünü kullanıyor. OTTO, havacılık, otomotiv, e-ticaret ve sağlık hizmetleri de dahil olmak üzere birçok sektörde işletmeler için malzeme elleçleme için esnek ve akıllı otomatik sürüş araçları üretmektedir. Ham maddeleri hatta hatlara ve çapraz yerleştirme paletlerinden süreçler arasındaki hareketli parçalara getirmek, AI, OTTO Motors için önemli bir iş yüküdür ve bir OTTO aracının çevresini analiz etmesini, bu bilgileri içselleştirmesini ve hareket ettikçe hızlı bir şekilde karar vermesini mümkün kılar, endüstriyel zeminde bağımsız olarak [11].

Tahmini bakımı etkinleştirme

Akıllı üreticiler, makine arızalarını tahmin etmek ve önlemek için AI sistemlerini sensörlerden gelen veriler ve Nesnelerin İnterneti (IoT) ile birlikte kullanıyor. Amaç, aksaklığı ve rahatsızlığı en aza indirmek, sorunları önlemek ve sorunları hızlı bir şekilde çözmek için öngörücü bakım kullanmaktır. Üreticiler için geri ödeme çok büyük olabilir. McKinsey & Company tarafından hazırlanan bir rapor, AI güdümlü öngörücü bakımın, varlık verimliliğini yüzde 20'ye kadar artırabileceğini ve bakım maliyetlerini yüzde 10'a kadar azaltabildiğini ve bakım işlerinden kaynaklanan makine duruş süresini büyük ölçüde azaltabileceğini belirtiyor [12].

Büyüme koşullarının otomatikleştirilmesi

New Jersey merkezli bir dikey tarım şirketi olan AeroFarms, iç mekan çiftçilik faaliyetlerine yeni hassasiyet ve verimlilik seviyeleri getirmek için makine öğrenimi, veri analizi, IoT ve ilgili teknolojileri kullanıyor. Bu teknolojiler, şirkete sürekli büyüyen koşulları optimize etme çabalarında yardımcı olmaktadır. Örneğin, AI, bitki besinlerini, ışığı ve diğer faktörleri ayarlamak için görüntü tanıma ve sınıflandırmayı otomatikleştirmek için kullanılır [13].

2.5. Endüstri 4.0'da Makine Öğrenimi

Uzun bir süre, insanlar makinelerin nasıl çalıştığını, nasıl çalıştırılacağını ve mümkün olduğunca verimli nasıl kullanılacağını öğrenmek zorundaydı. Bugün ise tam tersi: Makineler süreçleri anlamayı, çevreleriyle etkileşime girmeyi ve davranışlarını akıllıca uyarlamayı öğreniyor. Robotik, sensör teknolojisi, büyük veri ve yapay zeka, endüstriyel üretimdeki makineleri her zamankinden daha akıllı hale getiriyor.

Özellikle, makine öğrenimi gibi yöntemler son yıllarda kaydedilen önemli ilerlemeden sorumludur. Makine öğrenimi algoritmaları üretim sürecine iki büyük avantaj sağlar:

- Ürün kalitesinin iyileştirilmesi.
- Üretim sürecinin esnekliği.

Makine Öğrenimi, bazı endüstriyel sektörlerde yeniliğin ana itici gücü haline gelmiştir. Endüstri 4.0'da makine öğrenimi için en önemli uygulamalara göz atmak yeterlidir.

2.5.1. Öngörücü bakım

Sensör verileri, makinelerin durumu hakkında değerli bilgiler sağlar. Buna ek olarak, sensörler yıllardır üretimde daha küçük ve daha ucuz hale gelmektedir. Makinelerin izlenmesi şirketler için gittikçe daha ekonomik hale geliyor. Bir üretim makinesinde, binlerce ayrı ölçüm noktası, makinenin mevcut “sağlıklı” durumunun dijital bir görüntüsünü üretebilir. Bu sağlıklı durumdaki veri setleri daha sonra makine öğrenme algoritmalarını eğitmek için kullanılabilir.

Amaç, makine bileşenlerinin, olası bileşenlerin arızalarını veya arızalarını gösteren birçok petabayt sensör verisindeki paternleri tanımlamak için kullanmaktır. Öncelikli amaç, makineleri arızalı olmadan önce onarabilmektir [14].

2.5.2. Üretimde otonom araçlar ve interaktif makineler

Makine Öğrenmesini bu kadar güçlü bir araç yapan özelliği vardır: kalıpları ve düzenleri bağımsız olarak tanıma ve bunları yeni, öngörülemeyen durumlara uygulama yeteneği. Karayolu trafiği, eğitimli kurallar temelinde değerlendirilmesi gereken sürekli olarak yeni durumların ortaya çıktığı birçok ortamdan sadece biridir. Ancak otonom araçlar ve otonom makineler, bunun mümkün kıldığı sadece bir olası senaryodur.

En azından aynı derecede önemli, insanlarla etkileşime girecek kadar akıllı olan işbirliği makineleri olacaktır. Akıllı, etkileşimli sistemler, ağa bağlı fabrikadaki üretim sürecinin dönüşümünün merkezinde yer alır. Böylece “seri büyüklüğü 1” olan ürünlerin üretimi mümkündür [14].

2.5.3. Kalite kontrol

Özellikle Almanya'da endüstriyel üretim bağlamında üretilen ürünlerin kalitesi büyük önem taşımaktadır. Geleneksel olarak, ürünlerin kalitesi sadece üretim sürecinin sonunda kontrol edilir. Makine Öğrenimi bu ilişkiyi tersine çevirir: Sensör

teknolojisinin kullanımı ve verilerin bileşen düzeyinde sürekli değerlendirilmesi, çalışma sırasında iş parçalarının kalitesini kontrol etmeyi ve garanti etmeyi mümkün kılar. Özellikle hata kaynakları önceden tespit edilebildiğinde ve değişkenleri etkilendiğinde, üretim sırasında bireysel ölçüm verileri toplanabilir ve testler üretim sürecine entegre edilebilir.

Makine Öğrenmesine dayalı bir test otomasyonu üretim kalitesini önemli ölçüde artırabilir. Özellikle makine mühendisliği sektörü, makine öğrenimi ve sürekli kalite kontrolünden yararlanabilir [14].



BÖLÜM 3. GAZ SENSÖRLERİ

3.1. Sensör

Miktar veya olaylardaki deęişiklikleri tespit ederek çıktı veren bir cihaz bir sensör olarak tanımlanabilir. Genellikle sensörler, girişlerdeki deęişikliklere karşılık gelen bir elektrik sinyali veya optik çıkış sinyali üretir. Farklı tipte sensörler vardır, örneğin, giriş sıcaklığı deęişikliklerine dayalı bir çıkış voltajı üreten sıcaklık sensörü olarak düşünölebilen bir termokupl düşünöbilirsiniz [15].

3.2. Gaz Sensörü

Gaz sensörü teknolojisi, özellikle endüstriyel proseslerin çevresel etkilerinin daha iyi izlenmesi için son yıllarda önemli ölçüde ilerlemiştir. İki tür ekipman bu ihtiyacı karşıladığını iddia edebilir: pahalı ve karmaşık analizörler ve hassas bir katmandan oluşan daha basit kimyasal sensörler. Yarı iletken sensörler söz konusu olduğunda, bu katman yarı iletkendir ve iletkenliği çevredeki ortamın bir fonksiyonudur. Bu gaz sensörlerinin performansları çoktur, hassasiyet, seçicilik, tepki süresi, iyileşme süresini alıntılalım. Son yıllarda, gaz algılama ve konsantrasyon ölçüm yöntemlerine olan talep önemli ölçüde artmıştır. Bu ilgi, temel olarak çevre, güvenlik veya süreç kontrolü hususlarından kaynaklanmaktadır [16].

Gaz algılama alanında, analizörler (analitik aletler) ve kimyasal sensörler arasında bir ayırım yapılmalıdır. Analizörler genellikle farklı mekanik, kimyasal ve elektriksel elemanları birleştiren nispeten karmaşık sistemlerdir. Montaj genellikle pahalı, hantal ve enerji tüketir, bu da onu yerinde ölçümler için uygun hale getirir. Ek olarak, bu aletler, ya tespit tekniğinin kendisiyle ya da numuneleri işleme gereksinimiyle sıklıkla uzun bir tepki süresinden etkilenir. Bu cihaz kategorisi kromatografı ve

çeşitli spektrometreleri içerir. Bu cihazların ana avantajları, gaz numunesinin tam ve hassas bir şekilde analiz edilmesi olasılığıdır.

Diğer yandan kimyasal sensörler, genellikle etkileştiği gazın ve kimyasal etkileşimi bir elektrik sinyaline dönüştüren bir dönüştürücü sisteminin tanınmasına izin veren hassas bir katmandan oluşan basit sistemlerdir. Genellikle, iki işlev birbirine sıkı sıkıya bağlıdır.

Sensörlerin ana avantajları şunlardır: küçük boyutları, düşük enerji tüketimi, düşük maliyetleri, nispeten kısa tepki süreleri. Tüm bu avantajlar onları yerinde ölçümler ve çevrimiçi proses kontrolü için ideal araçlar haline getirir. Öte yandan, kompleks karışımların analizi, bir veya diğer gaz türlerine duyarlı çok sayıda sensörün kullanılmasını gerektirir. Birkaç farklı sensörden gelen sinyalin yeterli sinyal işleme kullanılarak işlenmesi, "elektronik burun" olarak adlandırılan şeyi oluşturur. Son yıllarda, bu alan yoğun bir faaliyet konusu olmuştur. Hassas katmanın türü ve transdüksiyon prensibi ile ayırt edilen birkaç kimyasal sensör ailesi vardır.

Sensörlerin performansı farklı parametrelerle karakterizedir. Başlıcaları:

- a. Duyarlılık: sensör sinyali varyasyonları ile hedef gaz konsantrasyonu arasındaki bağlantı.
- b. Hassasiyet limiti: hedef gazın minimum saptanabilir konsantrasyonu.
- c. Seçicilik: diğerlerinin dışlanması için hedef gazı tespit etme yeteneği.
- d. Müdahale: sensörden istenmeyen bir yanıtı yol açan diğer gazlar.
- e. Tepki süresi: tespit edilecek gazla temas ettikten sonra çalışma yanıtının% 90'ına ulaşmak için gereken süre.
- f. İyileşme süresi: gazın boşaltılmasından sonra sinyalin temel değerine dönmek için gereken süre.
- g. Ömür: sensörün özelliklerinin yeterli bir hassasiyetle kullanılmasına izin verdiği süre (uygulamanın gerekliliğine bağlıdır).
- h. Sapma: hedef gazın yokluğunda sensör taban çizgisinin evrimi.
- i. Tüketim: çalışma sırasında sensör tarafından emilen güç [17].

3.3. Farklı Gaz Sensörü Türleri

Gaz sensörleri tipik olarak, inşa edildiği algılama elemanının tipine göre çeşitli tiplerde sınıflandırılır. Aşağıda, genellikle çeşitli uygulamalarda kullanılan algılama elemanına bağlı olarak çeşitli gaz sensörü tiplerinin sınıflandırılması verilmiştir.

- Metal Oksit bazlı gaz Sensörü.
- Optik gaz sensörü.
- Elektrokimyasal gaz Sensörü.
- Kapasitans bazlı gaz Sensörü.
- Kalorimetrik gaz Sensörü.
- Akustik tabanlı gaz sensörü [18].

3.4. Gaz Sensörlerinin Uygulamaları

- Endüstride toksik gazların konsantrasyonunu izlemek için kullanılır.
- Acil durumları tespit etmek için evlerde kullanılır.
- Petrol teçhizatının bulunduğu yerlerde, salınan gazların konsantrasyonunu izlemek için kullanılır.
- Müşterilerin sigara içmesini önlemek için otellerde kullanılır.
- Ofislerde hava kalitesi kontrolünde kullanılır.
- Klimalarda CO2 seviyelerini izlemek için kullanılır.
- Yangın tespitinde kullanılır.
- Madenlerde gaz konsantrasyonunu kontrol etmek için kullanılır.
- Nefes analizörü [18].

3.5. Gaz Sürüklenme Veri Seti

"Sınıflandırıcı topluluklar kullanılarak kimyasal gaz sensörü sürüklenme telafisi" başlıklı makalenin okunmasından sonra birlikte çalıştığımız veri tabanını aldık. Veri seti, çeşitli konsantrasyon seviyelerinde 6 gazdan oluşan bir ayırma görevinde sapma dengelemesi için simülasyonlarda kullanılan 16 kimyasal sensörden 13910 ölçüm içerir.

Bu veriler, Ocak 2007 - Şubat 2011 (36 ay) içinde, Kaliforniya Üniversitesi BioCircuits Enstitüsü'nün ChemoSignals Laboratuvarında bulunan bir gaz dağıtım platformu tesisinde toplandı. Ölçüm sistemi platformu, ilgilenilen kimyasal maddelerin istenen konsantrasyonlarını yüksek doğrulukta ve yüksek oranda tekrarlanabilir bir şekilde elde etmek için çok yönlülük sağlar, böylece insan müdahalesinin neden olduğu ortak hataları en aza indirir ve sadece gerçekleri telafi etmek için kimyasal sensörlere konsantre olmayı mümkün kılar [17].

Sonuçta elde edilen veri seti, her biri çok çeşitli konsantrasyon değerlerinde dozlanan altı farklı saf gaz halindeki maddeden, yani Amonyak, Asetaldehit, Aseton, Etilen, Etanol ve Toluen'den kayıtları içerir.

BÖLÜM 4. KULLANILAN SINIFLANDIRMA ALGORİTMASI

4.1. Giriş

Bu bölümde çalışmamızda kullanılan veri madenciliği görevleri için farklı makine öğrenme algoritmalarını sunacağız. Bu bölüm, ilk olarak makine öğrenimini tanımlayan iki bölümden ve ikincisi işimiz için kullanılan her algoritmanın ayrıntıları sunulmaktadır.

4.2. Makine Öğrenmesi

4.2.1. Tanım

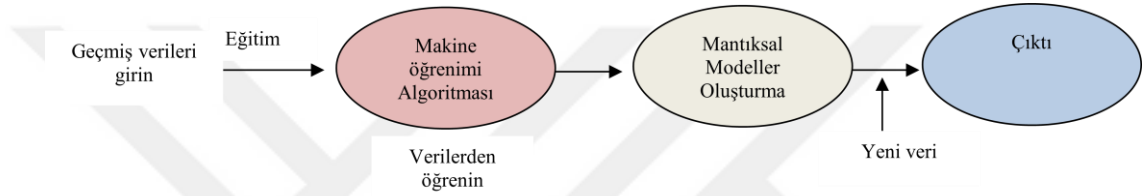
Makine öğrenimi, sistemlere açıkça programlanmadan, deneyimden otomatik olarak öğrenme ve geliştirme yeteneği sağlayan yapay zekanın (AI) bir uygulamasıdır. Makine öğrenimi, verilere erişebilen ve öğrenmeyi kendileri için kullanabilen bilgisayar programlarının geliştirilmesine odaklanır. Gerçek dünyada, öğrenme yetenekleri ile deneyimlerinden her şeyi öğrenebilen insanlar tarafından kuşatılmış durumdayız ve talimatlarımız üzerinde çalışan bilgisayarlarımız veya makinelerimiz var. Özet olarak şöyle tanımlayabiliriz: "Makine öğrenimi, bir makinenin verilerden otomatik olarak öğrenmesini, deneyimlerden performansı iyileştirmesini ve açıkça programlanmadan işleri tahmin etmesini sağlar"[19].

4.2.2. Makine öğrenimi adımları

Bir Makine Öğrenim sistemi geçmiş verilerden öğrenir, tahmin modellerini oluşturur ve her yeni veri aldığı anda bunun çıktısını tahmin eder. Büyük miktarda veri, çıktıyı

daha doğru tahmin eden daha iyi bir model oluşturmaya yardımcı olduğundan, tahmini çıktının doğruluğu veri miktarına bağlıdır.

Bazı tahminler gerçekleştirmemiz gereken karmaşık bir sorununuz olduğunu varsayalım, bu nedenle bunun için bir kod yazmak yerine, verileri genel algoritmalara beslememiz gerekiyor ve bu algoritmaların yardımıyla makine mantığı, veri ve çıktı tahmini yapılmalıdır. Makine öğrenimi sorun hakkında düşünme şeklimizi değiştirdi. Aşağıdaki blok şema Makine Öğrenimi algoritmasının çalışmasını açıklamaktadır: [19].



Şekil 4.1. Makine öğrenmesi adımı

4.2.3. Makine öğrenimi ihtiyacı

Makine öğrenmesi ihtiyacı her geçen gün artmaktadır. Makine öğrenmesi ihtiyacının arkasındaki neden, bir kişinin doğrudan uygulayamayacağı kadar karmaşık görevleri yerine getirebilmesidir. Bir insan olarak, büyük miktarda veriye manuel olarak erişemediğimiz için bazı sınırlamalarımız vardır, bu nedenle bazı bilgisayar sistemlerine ihtiyacımız var bu yüzden işleri kolaylaştırmak için makine öğrenimi geliyor.

Makine öğrenme algoritmalarını onlara büyük miktarda veri sağlayarak eğitebilir ve verileri keşfetmelerine, modelleri oluşturmalarına ve gerekli çıktıları otomatik olarak tahmin etmelerine izin verebiliriz. Makine öğrenimi algoritmasının performansı veri miktarına bağlıdır ve maliyet fonksiyonu ile belirlenebilir. Makine öğrenimi sayesinde hem zamandan hem de paradan tasarruf edebiliriz.

Makine öğrenmenin önemi, kullanım durumları ile kolayca anlaşılabilir, Şu anda, makine öğrenimi, kendi kendine giden otomobillerde, siber dolandırıcılık tespiti, yüz tanıma ve Facebook tarafından arkadaş önerisinde vb. kullanılmaktadır. Netflix ve Amazon gibi çeşitli üst düzey şirketler kullanıcının ilgisini analiz etmek için çok miktarda veri kullanan makine öğrenimi modelleri oluşturmak ve buna göre ürün önermek [19].

4.2.4. Makine öğrenimi türleri

Geniş bir düzeyde, makine öğrenimi üç tipte sınıflandırılabilir: Denetimli öğrenme, Denetimsiz öğrenme ve Takviye öğrenme.

Denetimli Öğrenme

Denetimli öğrenme, makine öğrenme sistemine onu eğitmek için örnek etiketli veriler sağladığımız ve bu temelde çıktıyı tahmin ettiğimiz bir tür makine öğrenme yöntemidir. Sistem, veri kümelerini anlamak ve her veri hakkında bilgi edinmek için etiketli verileri kullanarak bir model oluşturur, eğitim ve işleme tamamlandıktan sonra tam çıktıyı tahmin edip etmediğini kontrol etmek için örnek bir veri sağlayarak modeli test ederiz.

Denetimli öğrenmenin amacı, girdi verilerini çıktı verileriyle eşlemektir. Denetimli öğrenme denetime dayanır ve bir öğrencinin öğretmenin gözetiminde bir şeyler öğrenmesi ile aynıdır. Denetimli öğrenme örneği spam filtrelemedir. Denetimli öğrenme iki algoritma kategorisinde daha fazla gruplanabilir: 'Sınıflandırma' ve 'Regresyon'.

Denetimsiz Öğrenme

Gözetimsiz öğrenme, bir makinenin gözetim olmadan öğrendiği bir öğrenme yöntemidir. Eğitim, makineye etiketlenmemiş, sınıflandırılmamış veya kategorize edilmemiş veri seti ile sağlanır ve algoritmanın herhangi bir denetim olmadan bu

veriler üzerinde hareket etmesi gerekir. Gözetimsiz öğrenmenin amacı, girdi verilerini yeni özelliklere veya benzer desenlere sahip bir grup nesneye yeniden yapılandırmaktır.

Denetimsiz öğrenmede, önceden belirlenmiş bir sonucumuz yoktur. Makine, büyük miktarda veriyle ilgili faydalı bilgiler bulmaya çalışır. Ayrıca iki algoritma kategorisinde sınıflandırılabilir: 'Kümeleme' ve 'İlişkilendirme'.

Takviye Öğrenme

Takviye öğrenme, bir öğrenme temsilcisinin her doğru eylem için bir ödül aldığı ve her yanlış eylem için bir ceza aldığı geri bildirim tabanlı bir öğrenme yöntemidir. Temsilci bu geri bildirimlerle otomatik olarak öğrenir ve performansını artırır. Takviye öğrenmede, ajan çevre ile etkileşime girer ve onu araştırır. Bir ajanın amacı en fazla ödül puanını almak ve böylece performansını arttırmaktır. Kollarının hareketini otomatik olarak öğrenen robot köpek takviye öğreniminin bir örneğidir.

4.3. Sınıflandırma Algoritması

Sınıflandırma algoritması, eğitim verileri temelinde yeni gözlem kategorisini tanımlamak için kullanılan Denetimli Öğrenme tekniğidir. Sınıflandırmada, bir program verilen veri kümesinden veya gözlemlerden öğrenir ve ardından yeni gözlemi birkaç gruba göre sınıflandırır. Evet veya Hayır, 0 veya 1, Spam veya Spam değil, kedi veya köpek vb. Sınıflar hedef / etiket veya kategori olarak adlandırılabilir [19].

Sınıflandırma Algoritmaları Temel olarak iki kategoriye ayrılabilir:

- a) Doğrusal Modeller
 - Lojistik Regresyon
 - Destek Vektör Makineleri
- b) Doğrusal Olmayan Modeller
 - K-En Yakın Komşular

Çekirdek SVM
Naif Bayes
Karar Ağacı Sınıflandırması
Rastgele Orman Sınıflandırması.

Sınıflandırmayı veri kümesine uygulayan algoritma, sınıflandırıcı olarak bilinir. İki tür Sınıflandırma vardır:

a) İkili Sınıflandırıcı:

Sınıflandırma probleminin sadece iki olası sonucu varsa, o zaman İkili Sınıflandırıcı olarak adlandırılır. Örnekler: EVET veya HAYIR, ERKEK veya KADIN, KEDİ veya KÖPEK vb.

b) Çok Sınıflı Sınıflandırıcı:

Bir sınıflandırma sorununun ikiden fazla sonucu varsa, bu soruna Çok Sınıflı Sınıflandırıcı denir. Örnek: Mahsul türlerinin sınıflandırılması, Müzik türlerinin sınıflandırılması [19].

4.3.1. Rastgele orman

Rastgele karar ormanları için ilk algoritma Tin Kam Ho tarafından, Ho'nun formülasyonunda Eugene Kleinberg tarafından önerilen sınıflandırmaya "stokastik ayrımcılık" yaklaşımını uygulamanın bir yolu olan rastgele altuzay yöntemi kullanılarak oluşturulmuştur.

Algoritmanın bir uzantısı Leo Breiman ve "Random Forests" i (2019'dan itibaren Minitab, Inc. şirketine ait) ticari marka olarak kaydeden Adele Cutler tarafından geliştirilmiştir. Eklenti, Breiman'ın "torbalama" fikrini ve önce Ho, daha sonra bağımsız olarak Amit ve Geman tarafından bağımsız olarak kontrol edilen varyanslı bir karar ağaçları koleksiyonu oluşturmak için tanıtılan rasgele özellik seçimini birleştiriyor [20].

4.3.1.1. Rastgele orman algoritması

Öncelikle, Random Forest algoritması denetimli bir sınıflandırma algoritmasıdır. Onu bir şekilde bir orman yaratmak ve rastgele yapmak olan adından görebiliriz. Ormandaki ağaç sayısı ile elde edebileceği sonuçlar arasında doğrudan bir ilişki vardır: ağaç sayısı arttıkça sonuç daha doğru olur. Ancak dikkat edilmesi gereken bir nokta, ormanı oluşturmanın, kararı bilgi kazancı veya kazanım endeksi yaklaşımı ile inşa etmekle aynı olmadığıdır [21].

Karar ağacı bir karar destek aracıdır. Olası sonuçları göstermek için ağaç benzeri bir grafik kullanır. Karar ağacına hedefleri ve özellikleri olan bir eğitim veri kümesi girerseniz, bazı kurallar dizisi formüle edilir. Bu kurallar tahminleri gerçekleştirmek için kullanılabilir. Bu noktayı göstermek için bir örnek kullanabiliriz: kızınızın animasyonlu bir filmi beğenip beğenmeyeceğini tahmin etmek istediğinizi varsayalım, sevdiği, geçmiş animasyon filmlerini toplamalı ve bazı özellikleri girdi olarak almalısınız. Ardından, karar ağacı algoritması aracılığıyla kuralları oluşturabilirsiniz.

Daha sonra bu filmin özelliklerini girebilir ve kızınızın beğenip beğenmeyeceğini görebilirsiniz. Bu düğümleri hesaplama ve kuralları oluşturma süreci bilgi kazancı ve Gini indeksi hesaplamalarını kullanmaktır. Rastgele Orman algoritması ile karar ağacı algoritması arasındaki fark, Rastgele Orman'da kök düğümü bulma ve özellik düğümlerini bölme işlemlerinin rastgele çalışmasıdır [21].

4.3.1.2. Rastgele orman algoritması gerçek hayat örneği

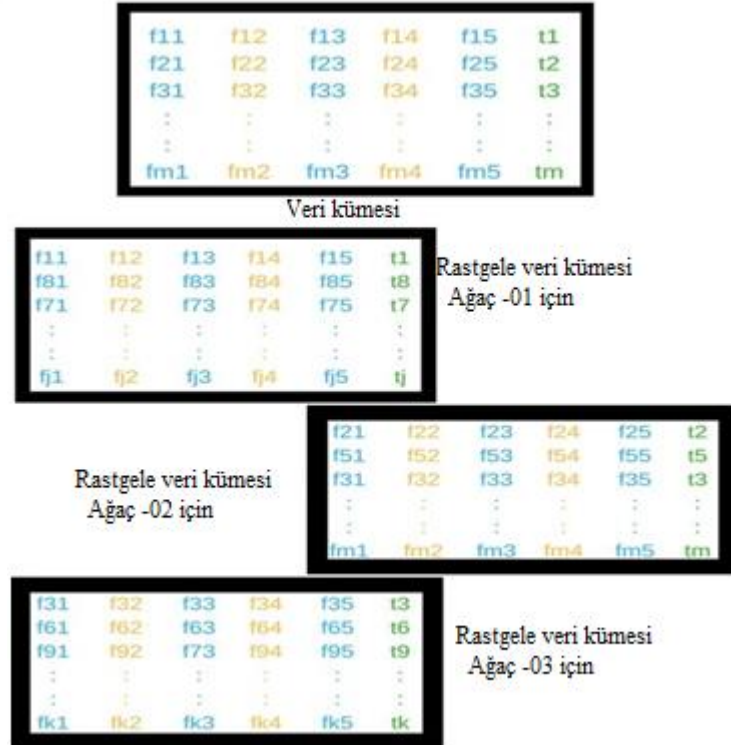
Diyelim ki Mady iki haftalık tatili için hoşuna gidebilecek farklı yerlere gitmek istiyor ve arkadaşından tavsiye istiyor. Arkadaşı zaten nerede olduğunu ve ziyaret ettiği yerleri sevip sevmediğini soracak. Mady'nin cevaplarına dayanarak, arkadaşını tavsiye vermeye başlar. Burada arkadaşını karar ağacını oluşturur. Mady daha fazla arkadaşından tavsiye istemektedir çünkü sadece bir arkadaşının doğru bir karar vermesine yardımcı olamayacağını düşünmektedir. Bu yüzden diğer arkadaşları da

ona rastgele sorular soruyor ve sonunda bir cevap veriyor. En çok oy alan yeri tatil kararı olarak görüyor. Burada yazar bu örnek için bir analiz sunmaktadır.

Bir arkadaşı ona bazı sorular sordu ve cevaplara dayanarak en iyi yerin tavsiyesini verdi. Bu tipik bir karar ağacı algoritma yaklaşımıdır. Arkadaş, cevaplara dayalı olarak kurallar oluşturdu ve kuralları eşleşen kuralları bulmak için kuralları kullandı. Mady'nin arkadaşları da rastgele bir şekilde farklı sorular sordular ve Mady için buranın oyları olan cevaplar verdiler. Sonunda, en yüksek oyu alan yer Mady'nin gitmeyi seçeceği yerdir. Bu tipik Rastgele Orman algoritması yaklaşımıdır [21].

4.3.1.3. Rastgele orman algoritması çalışır

Rastgele orman algoritmasında iki aşama vardır, biri rastgele orman oluşturma, diğeri ilk aşamada oluşturulan rastgele orman sınıflandırıcısından bir tahmin yapmaktır. Tüm süreç aşağıda gösterilmektedir ve şekli kullanarak anlaşılması kolaydır [22].



Şekil 4.2. Rastgele seçim özellikleri sürecini göstermektedir [22]

Rastgele orman algoritması için sözde kod iki aşamaya ayrılabilir. Rastgele orman oluşturma sözde kodu ve oluşturulan rasgele orman sınıflandırıcısından tahmin gerçekleştirmek için sözde kodu.

a. Rastgele Orman sözde kodu:

1. $k \ll m$ olan toplam "m" özelliklerinden rastgele "K" özellikleri seçin.
2. "K" özellikleri arasında, "d" düğümünü en iyi ayrılma noktasını kullanarak hesaplayın.
3. En iyi bölünmeyi kullanarak düğümü çocuk düğümlere ayırın.
4. "l" düğüm sayısına ulaşılan kadar 1-3 arasındaki adımları tekrarlayın.
5. "n" sayı ağacı oluşturmak için "n" sayı kez 1 ila 4 arasındaki adımları tekrarlayarak orman oluşturun.

Rasgele orman algoritmasının başlangıcı, toplam "m" özelliklerinden rasgele "k" özelliklerinin seçilmesiyle başlar. Resimde, rastgele özellikler ve gözlemler aldığımızı gözlemleyebilirsiniz.

Bir sonraki aşamada, en iyi bölünmüş yaklaşımı kullanarak kök düğümü bulmak için rastgele seçilen "k" özelliklerini kullanıyoruz.

Bir sonraki aşamada, aynı en iyi bölünmüş yaklaşımı kullanarak çocuk düğümlerini hesaplayacağız. İlk 3 ağaç kök düğümü ve yaprak düğümü hedefe sahip olana kadar aşamaları yapar.

Son olarak, rastgele oluşturulan ağaçlar oluşturmak için 1 ila 4 aşamayı tekrarlıyoruz. Bu rastgele oluşturulmuş ağaçlar rastgele ormanı oluşturur [22].

b. Rastgele orman tahmini sözde kodu:

- Test özelliklerini alır ve sonucu tahmin etmek için rastgele oluşturulmuş her karar ağacının kurallarını kullanır ve tahmin edilen sonucu saklar (hedef).

- Her tahmin edilen hedef için oyları hesaplayın
- Yüksek oylanan tahmin edilen hedefi rastgele orman algoritmasından elde edilen nihai tahmin olarak düşünün. Sürecin anlaşılması kolaydır ve etkilidir.

Eğitilmiş rasgele orman algoritmasını kullanarak tahmin yapmak için test özelliklerini rastgele oluşturulmuş her ağacın kurallarından geçirmemiz gerekir. Diyelim ki rastgele ormandan 100 rastgele karar ağacı oluşturduk.

Her rastgele orman, aynı test özelliği için farklı hedef (sonuç) öngörür. Daha sonra tahmin edilen her bir hedef oy dikkate alınarak hesaplanır. 100 rastgele karar ağacının bazı 3 benzersiz hedefin x , y , z olduğu tahmin edildiğini varsayalım, o zaman x 'in oyları 100 rastgele karar ağacından kaç ağaç tahmini x olduğunu gösterir.

Benzer şekilde diğer 2 hedef için (y , z) eğer x yüksek oy alıyorsa, 100 rastgele karar ağacından 60 ağaç hedefin x olacağını öngörüyor. Sonra son rastgele orman x 'i tahmin edilen hedef olarak döndürür [22].

4.3.1.4. Rasgele orman algoritması uygulaması.

- Bankacılıktaki uygulama için, Rasgele Orman algoritmasıyla sadık müşterileri bulmak için kullanılır, yani bol miktarda kredi alabilen ve bankaya doğru faiz ödeyebilen müşteriler ve dolandırıcılık müşterileri, bu da krediyi zamanında geri ödeyemeyen veya tehlikeli işlemleri olan kötü kayıtları olan müşteriler anlamına gelir.
- Tıpta uygulama için Random Forest algoritması hem tıptaki bileşenlerin doğru kombinasyonunu tanımlamak hem de hastanın tıbbi kayıtlarını analiz ederek hastalıkları tanımlamak için kullanılabilir.
- Borsadaki uygulama için, bir hisse senedinin davranışını ve beklenen zarar veya kârı tanımlamak için Rastgele Orman algoritması kullanılabilir.

- E-ticaretteki uygulamalarda, benzer müşterilerin deneyimlerine dayanarak müşterinin tavsiye edilen ürünleri beğenip beğenmeyeceğini tahmin etmek için Random Forest algoritması kullanılabilir [21].

4.3.1.5. Random forest algoritmasının avantajları.

- a. Rastgele Orman algoritması aşırı sığdırma problemini önleyecektir.
- b. Hem sınıflandırma hem de regresyon görevi için aynı rastgele orman algoritması kullanılabilir.
- c. Rastgele Orman algoritması, eğitim veri kümesindeki en önemli özellikleri, diğer bir deyişle özellik mühendisliğini tanımlamak için kullanılabilir.

4.3.1.6. Rasgele ormanın dezavantajları

- a. Yorumlanması zor. Birçok ağacın sonuçlarının ortalamasından dolayı, rastgele bir ormanın neden bu şekilde tahminlerde bulunduğunu anlamamız zorlaşıyor.
- b. Rastgele Orman'ın oluşturulması daha uzun sürer. Bir Karar Ağacına kıyasla hesaplama olarak pahalıdır [23].

4.3.2. K-en yakın komşular algoritması

En Yakın Komşu, Denetimli Öğrenme tekniğine dayanan en basit Makine Öğrenimi algoritmalarından biridir. Yeni vaka / veri ile mevcut vakalar arasındaki benzerliği varsayar ve yeni vakayı mevcut kategorilere en çok benzeyen kategoriye yerleştirir.

K-NN algoritması mevcut tüm verileri saklar ve yeni bir veri noktasını benzerliğe göre sınıflandırır. Bu, yeni veriler görüldüğünde K-NN algoritması kullanılarak kolayca bir sınıfta kategorisine sınıflandırılabileceği anlamına gelir.

Bir reklam kampanyası başlatmak istediğinizi ve hedef kitleye bağlı olarak farklı bir reklam mesajı göndermek istediğinizi düşünün. İlk olarak, hedef nüfusu gruplara

ayırmanız gerekir. Her gruptaki bireyler bir dereceye kadar benzerlik gösterecektir (yaş, maaş vb ...) K-Ortalamlar algoritması bunu yapacaktır [24].

Sınıflandırma yapmadan önce alınması gereken iki önemli karar vardır. Birincisi kullanılacak k değeri; buna isteğe bağlı olarak karar verilebilir veya en uygun değeri bulmak için çapraz doğrulamayı deneyebilirsiniz. Bir sonraki ve en karmaşık olan, kullanılacak mesafe metriğidir. Mesafeyi hesaplamamanın birçok farklı yolu vardır, çünkü oldukça belirsiz bir kavramdır ve kullanılacak uygun metrik her zaman veri kümesi ve sınıflandırma görevi tarafından belirlenecektir. Bununla birlikte, iki popüler olan Öklid mesafesi ve Kosinüs benzerliğidir.

Öklid mesafesi

Öklid mesafesi muhtemelen en çok aşına olduğunuz mesafedir. Temel olarak, eğitim veri noktasının sınıflandırılacak noktadan çıkarılmasıyla elde edilen vektörün büyüklüğüdür. Aşağıdaki denkleme bakınız Öklid mesafesi için genel formül (5.1)

$$\sqrt{\sum_{i=1}^K (x_i - y_i)^2} \quad (5.1)$$

Bir diğer yaygın metrik Kosinüs benzerliğidir. Büyüklüğünü hesaplamak yerine, Kosinüs benzerliği iki vektör arasındaki yön farkını kullanır.

4.3.2.1. Knn için farklı alan adı

En yakın komşular, mevcut tüm vakaları saklayan ve yeni vakaları benzerlik ölçüsüne (örneğin, mesafe fonksiyonları) göre sınıflandıran basit bir algoritmadır. KNN, 1970'lerin başında istatistiksel tahmin ve örüntü tanımada parametrik olmayan bir teknik olarak kullanılmıştır [25].

Anlaşılması ve uygulanması çok basit olmasına rağmen, bu yöntem, öneri sistemleri, semantik arama ve anomali tespiti gibi birçok alanda geniş uygulama gördü.

4.3.2.2. Knn uygulamaları ve örnekleri

Kredi derecelendirmeleri: finansal özelliklerin toplanması ve benzer finansal özelliklere sahip kişilerin veritabanıyla karşılaştırılması. Kredi notunun doğası gereği, benzer finansal detaylara sahip kişilere benzer kredi notları verilecektir. Bu nedenle, tüm mevcut hesaplamaları yapmak zorunda kalmadan, yeni bir müşterinin kredi notunu tahmin etmek için bu mevcut veritabanını kullanmak istiyorlar. Banka bireye kredi vermeli mi? Birey kredisinde temerrüde düşebilir mi? Bu kişi, kredilerini temerrüde düşüren veya düşürmeyen kişilere özellik olarak daha yakın mı?

Siyaset biliminde: potansiyel bir seçmeni “oylayacak” ya da “oy vermeyecek” ya da “Demokrat'a oy verecek ya da“ Cumhuriyetçi oy verecek ”olarak sınıflandırmak.

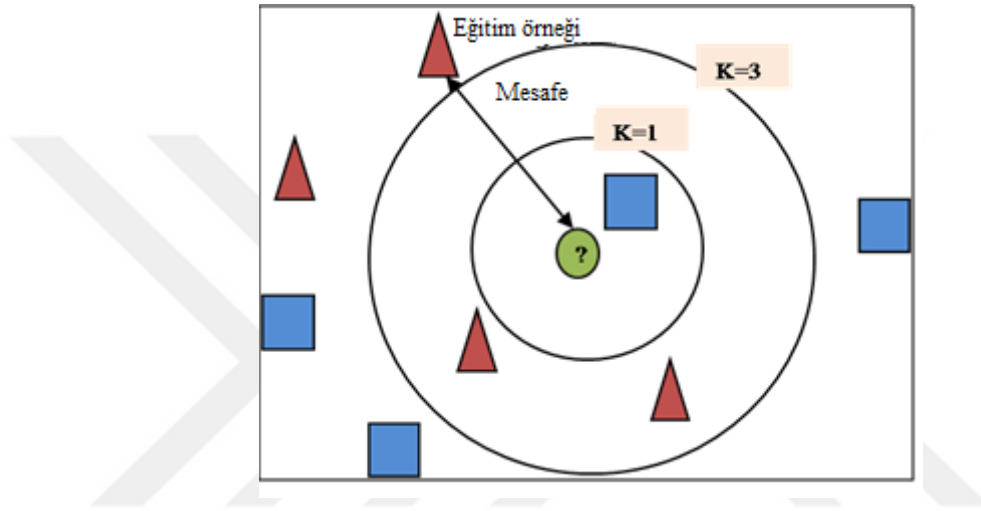
Daha ileri düzey örnekler arasında el yazısı algılama (OCR gibi), görüntü tanıma ve hatta video tanıma sayılabilir [26].

4.3.2.3. Knn algoritması

- a. Verileri yükleyin
- b. K'yi seçtiğiniz sayıda komşu için başlatın
- c. Verilerdeki her örnek için
 - Sorgu örneği ile verilerden geçerli örnek arasındaki mesafeyi hesaplayın.
 - Örneğin mesafesini ve dizinini sıralı bir koleksiyona ekleyin
- d. Sıralı mesafeler ve endeksler toplamını en küçükten en büyüğe (artan sırada) mesafelere göre sıralayın
- e. Sıralanan koleksiyondan ilk K girişlerini seçin
- f. Seçilen K girişlerinin etiketlerini alın
- g. Regresyon varsa, K etiketlerinin ortalamasını döndürün
- h. Sınıflandırma varsa, K etiketlerinin modunu döndürün [27].

4.3.2.4. Knn sınıflandırması örneği

KNN sınıflandırması için basit bir örnek alabiliriz. Test örneği (dairenin içindeki) ya mavi karelerin birinci sınıfına ya da ikinci kırmızı üçgen sınıfına sınıflandırılmalıdır. $K = 3$ ise (dış daire) ikinci sınıfa atanır, çünkü 2 üçgen ve iç dairenin içinde sadece 1 kare vardır. Örneğin, $k = 5$ ise, birinci sınıfa atanır (3 çember vs. dış çemberin dışındaki 2 üçgen) [26].



Şekil 4.3. KNN örnek sınıflandırması

4.3.2.5. K için doğru değeri seçme

Verileriniz için doğru olan K'yi seçmek için, KNN algoritmasını farklı K değerleriyle birkaç kez çalıştırırız ve algoritmanın sahip olduğu veriler verildiğinde doğru tahminler yapma yeteneğini korurken karşılaştığımız hata sayısını azaltan K'yi seçeriz. t daha önce görmedim.

Akılda tutulması gereken bazı şeyler şunlardır:

a) K değerini 1'e düşürdükçe tahminlerimiz daha az istikrarlı hale geliyor. $K = 1$ 'i düşünün ve birkaç kırmızı ve bir yeşil ile çevrili bir sorgulama noktamız var (yukarıdaki renkli arsanın sol üst köşesini düşünüyorum), ancak yeşil en yakın tek komşudur. Makul olarak sorgu noktasının büyük olasılıkla kırmızı olduğunu

düşünürüz, ancak $K = 1$ olduğundan KNN, sorgu noktasının yeşil olduğunu yanlış tahmin eder.

b) Tersine, K değerini arttırdıkça, tahminlerimiz çoğunluk oyu / ortalaması nedeniyle daha kararlı hale gelir ve bu nedenle daha kesin tahminler yapma olasılığı (belirli bir noktaya kadar) artar. Sonunda, artan sayıda hataya tanık olmaya başlıyoruz. Bu noktada K 'nin değerini çok ileri ittiğimizi biliyoruz.

c) Etiketler arasında çoğunluk oyu aldığımız durumlarda (örneğin, bir sınıflandırma probleminde modu seçmek gibi), genellikle K 'yi bir tiebreakçiye sahip olması için tek bir sayı yaparız [27].

4.3.2.6. Avantajları

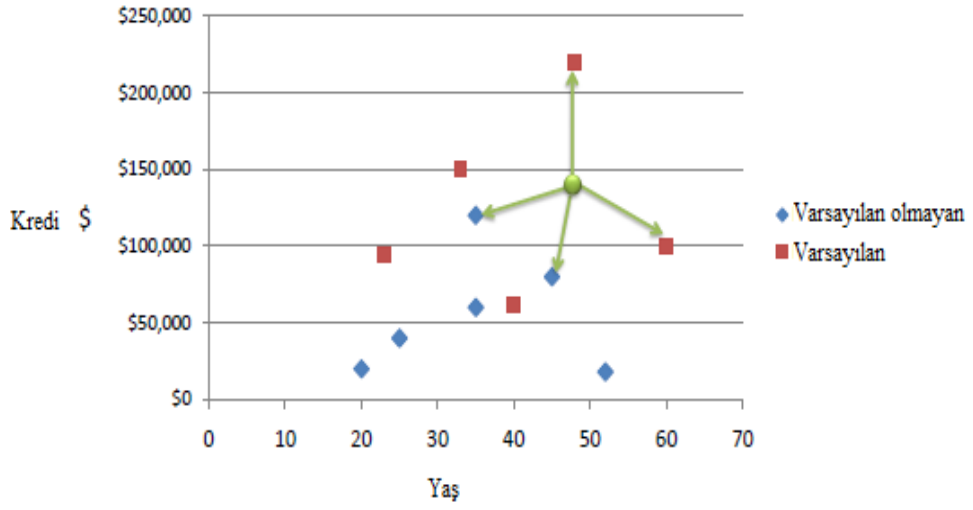
- Verilerle ilgili varsayımlar yok
- Basit algoritma
- Yüksek doğruluk
- Çok yönlü

4.3.2.7. Dezavantajları

- Hesaplama pahalıdır: çünkü algoritma tüm eğitim verilerini saklar.
- Yüksek bellek gereksinimi.
- Eğitim verilerinin tümünü (veya neredeyse tümünü) saklar.
- Tahmin aşaması yavaş olabilir (büyük N ile).
- Alakasız özelliklere ve verilerin ölçeğine duyarlıdır [26].

4.3.2.8. Kullanım örneği

Kredi temerrütüyle ilgili aşağıdaki verileri göz önünde bulundurun. Yaş ve Kredi iki sayısal değişkendir (öngörücüler) ve Varsayılan hedeftir.



Şekil 4.4. Örnek sınıflandırması[25]

Şimdi eğitim setini Öklid mesafesini kullanarak bilinmeyen bir vakayı (Yaş = 48 ve Kredi = 142.000 \$) sınıflandırmak için kullanabiliriz. $K = 1$ ise, en yakın komşu varsayılan = Y ile ayarlanan eğitim setindeki son durumdur. Öklid mesafesi (5.2) [25].

$$D = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \quad (5.2)$$

$$D = \text{Metrekare } [(48-33)^2 + (142000-150000)^2] = 8000.01 \gg \text{Varsayılan} = Y$$

Tablo 4.1. Vaka veri kümesi KNN kullan

Yaş	Kredi	Varsayılan	Mesafe	
25	\$40,000	N	102000	
35	\$60,000	N	82000	
45	\$80,000	N	62000	
20	\$20,000	N	122000	
35	\$120,000	N	22000	2
52	\$18,000	N	124000	
23	\$95,000	Y	47000	
40	\$62,000	Y	80000	3
60	\$100,000	Y	42000	
48	\$220,000	Y	78000	
33	\$150,000	Y	8000	1
48	\$142,000	?		

K = 3 ile, en yakın üç komşunun iki Varsayılan = Y ve bir Varsayılan = N vardır. Bilinmeyen vaka için tahmin yine Varsayılan = Y'dir [25].

4.3.2.9. Endüstride problem sınıflandırmasında kullanılan knn

Yarı iletken üreticileri İçin süreç

FD-kNN kullanan yeni bir arıza tespit yöntemi, çoğu yarıiletken işleminin (doğrusal olmayanlık ve çok modlu yörüngeler) bazı benzersiz özelliklerini açıkça hesaba katmak için geliştirilmiştir.

Geleneksel kNN algoritması, bir işlem modeli, yani normal numunelerin kNN kare mesafelerinin dağılımı için sadece normal çalışma verilerine ihtiyaç duyulacak şekilde uyarlanmıştır. Geliştirilen FD-kNN yöntemi sürecin doğrusallığı hakkında herhangi bir varsayımda bulunmadığından ve yerel mahallelere dayalı bir anormallik

saptadığından, geliştirilen FD-kNN yöntemi doğal olarak işlem doğrusalsızlığı ve çok modlu çevreyi idare etmektedir [28].

Yumuşak sensör modellemesi için

En yakın K komşuları yöntemine (KNN) dayanan bir yumuşak sensör modelleme yöntemi önerilmiştir. Bu yöntem, ikincil değişkenler sınıflamasına KNN uyguladı ve yumuşak ölçüm için bir model oluşturmak için sınıflandırılmış sonuç, temel bileşen analizi (KPCA) ve destek vektör makinesini (SVR) kullandı. KNN analizi ilişkili regresyon modelinden bağımsızdır, ancak model yapısını doğrudan etkilemiştir [29].

Gaz sensörü dizilerinin arıza tespit problemini çözmek için

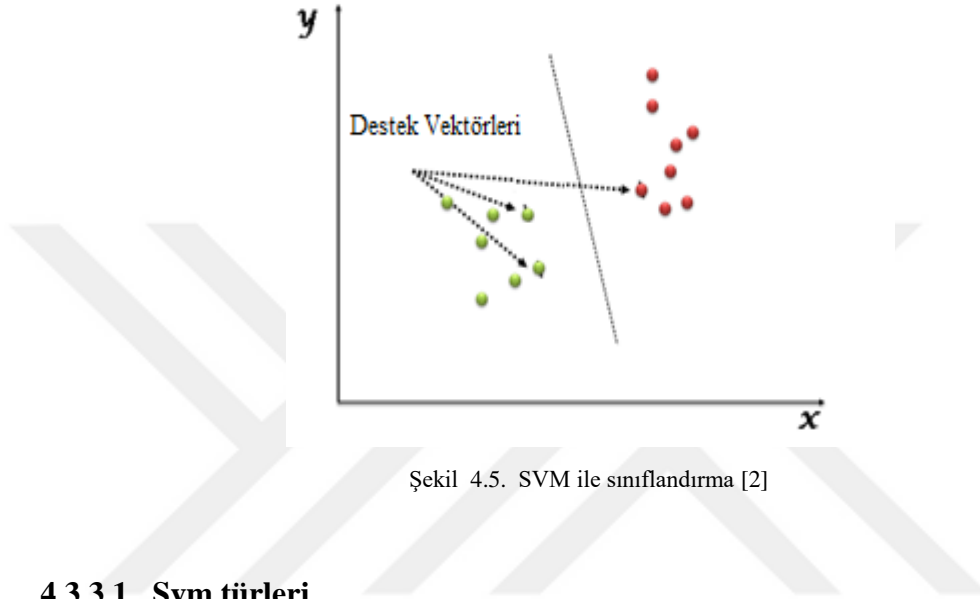
Doğal olarak verilerin doğrusal olmama olasılığını doğal olarak ele alan k-en yakın komşu (kNN) kuralı, gaz sensörü dizilerinin arıza tespit problemini çözmek için verilmiştir. KNN kuralına dayanan geleneksel arıza tespit yöntemlerinde, her yeni test örneğinin tespit süreci tüm eğitim numune setindeki tüm örnekleri içerir [30].

4.3.3. Destek vektör makinesi

Destek vektör makinesi, hem sınıflandırma hem de regresyon zorlukları için kullanılabilen denetimli bir makine öğrenme algoritmasıdır. Daha çok makine öğreniminde sınıflandırma problemlerinde kullanılır. Bu algortmada, her veri ögesini n-boyutlu uzayda bir nokta olarak çiziyoruz (burada n sahip olduğumuz özellik sayısıdır) her özelliğin değeri belirli bir koordinatın değeri olur. Daha sonra, iki sınıfı çok iyi ayıran hiper-düzlemi bularak sınıflandırma yaparız.

SVM algoritmasının amacı, n-boyutlu alanı sınıflara ayırabilecek en iyi çizgi veya karar sınırını oluşturmaktır. Böylece yeni veri noktasını gelecekte doğru kategoriye kolayca koyabiliriz. Bu en iyi karar sınırına hiper düzlem denir. Hiper düzlem kullanılarak sınıflandırılan iki farklı kategorinin bulunduğu aşağıdaki şemada görebildiğimiz gibi [2].

SVM, hiper düzlemi oluşturmaya yardımcı olan uç noktaları / vektörleri seçer. Bu uç durumlar destek vektörleri olarak adlandırılır ve bu nedenle algoritma Destek Vektör Makinesi olarak adlandırılır. Destek Vektörler sadece bireysel gözlemin koordinatlarıdır. Destek Vektör Makinesi, iki sınıfı (hiper-düzlem / çizgi) en iyi ayıran bir sınırdır [3].



Şekil 4.5. SVM ile sınıflandırma [2]

4.3.3.1. Svm türleri

SVM'yi iki tipte tanımlayabiliriz:

Doğrusal SVM: Doğrusal SVM doğrusal olarak ayrılabilir veriler için kullanılır, yani bir veri kümesinin tek bir düz çizgi kullanılarak iki sınıfa sınıflandırılabilmesi durumunda, bu veriler doğrusal olarak ayrılabilir veriler olarak adlandırılır ve doğrusal SVM sınıflandırıcısı olarak adlandırılan sınıflandırıcı kullanılır .

Doğrusal olmayan SVM: Doğrusal olmayan SVM doğrusal olmayan olarak ayrılmış veriler için kullanılır, yani bir veri kümesi düz bir çizgi kullanılarak sınıflandırılmazsa, bu tür veriler doğrusal olmayan veri olarak adlandırılır ve kullanılan sınıflandırıcı Non olarak adlandırılır doğrusal SVM sınıflandırıcısı. [3].

4.3.3.2. Svm uygulamaları

- SVM'ler çeşitli gerçek dünya sorunlarını çözmek için kullanılabilir:
- Görüntü sınıflandırması
- Biyolojik ve diğer bilimler
- Yüz tanıma
- Metin kategorizasyonu,
- Kanser Genomiği [4]
- Biyoinformatik
- Sanayi [6].

4.3.3.3. Svm algoritmasında köprü ve destek vektörleri:

Hiper düzlem:

Sınıfları n boyutlu uzayda ayırmak için birden fazla sınır / karar sınırı olabilir, ancak veri noktalarını sınıflandırmaya yardımcı olan en iyi karar sınırını bulmamız gerekir. Bu en iyi sınır SVM'nin hiper düzlemi olarak bilinir.

Hiper düzlemin boyutları, veri kümesinde bulunan özelliklere bağlıdır, yani 2 özellik varsa hiper düzlem düz bir çizgi olacaktır. Ve 3 özellik varsa, hiper düzlem 2 boyutlu bir düzlem olacaktır. Her zaman maksimum kenar boşluğuna sahip bir köprü oluştururuz, bu da veri noktaları arasındaki maksimum mesafe anlamına gelir.

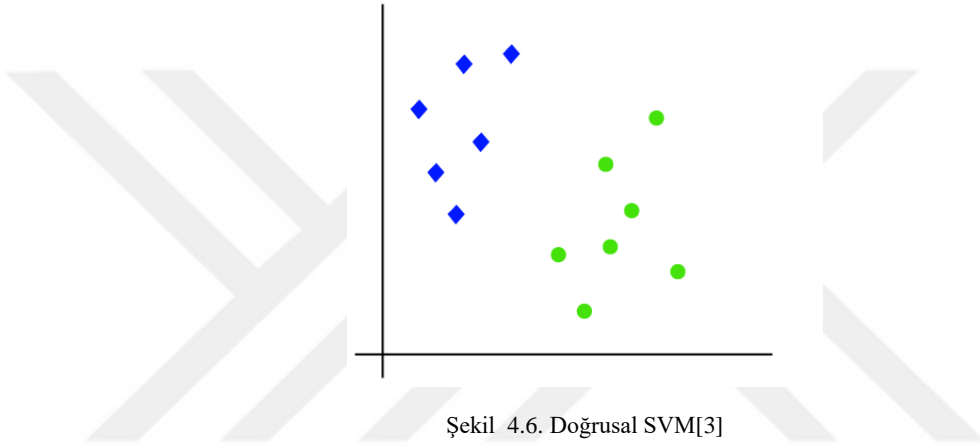
Destek Vektörleri:

Köprü birimine en yakın olan ve köprü düzleminin konumunu etkileyen veri noktaları veya vektörler Destek Vektörü olarak adlandırılır. Bu vektörler hiper düzlemi desteklediğinden, bu nedenle Destek vektörü olarak adlandırılır [3].

4.3.3.4. Svm'nin eserleri

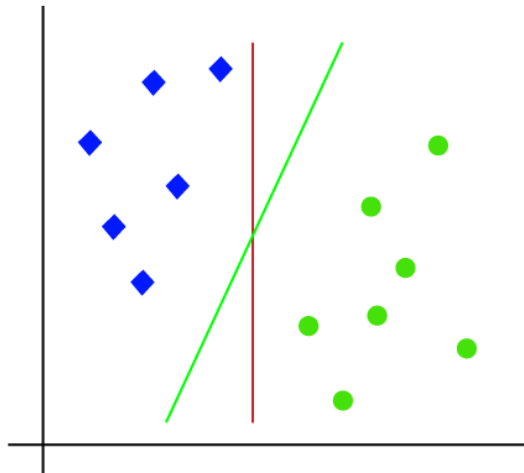
4.3.3.4.1. Doğrusal svm:

SVM algoritmasının çalışması bir örnek kullanılarak anlaşılabilir. İki etiketi (yeşil ve mavi) olan bir veri kümemiz olduğunu ve veri kümesinin x_1 ve x_2 olmak üzere iki özelliği olduğunu varsayalım. Koordinat çiftini (x_1, x_2) yeşil veya mavi olarak sınıflandırabilen bir sınıflandırıcı istiyoruz. Aşağıdaki resim:



Şekil 4.6. Doğrusal SVM[3]

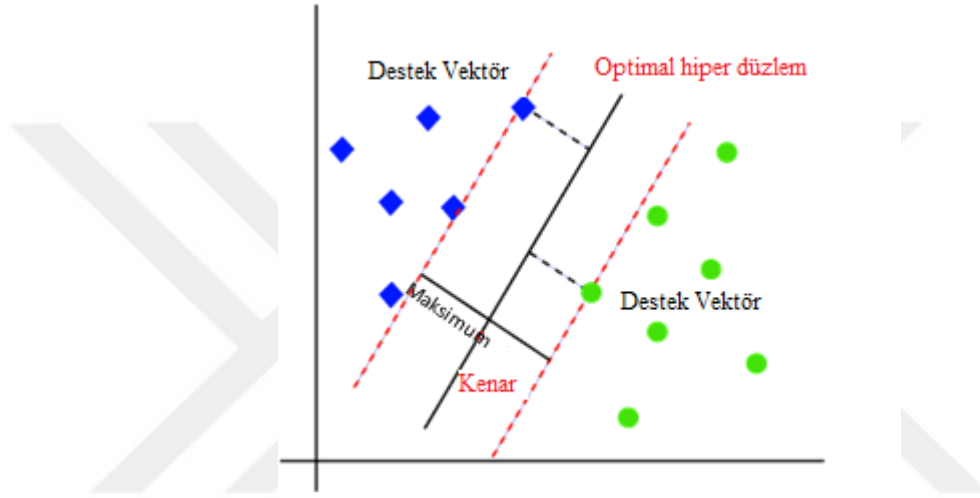
İki boyutlu bir alan olduğundan, sadece düz bir çizgi kullanarak, bu iki sınıfı kolayca ayırabiliriz. Ancak bu sınıfları ayırabilecek birden fazla satır olabilir. Aşağıdaki resim:



Şekil 4.7. Doğrusal SVM [3]

Bu nedenle, SVM algoritması en iyi çizgiyi veya karar sınırını bulmaya yardımcı olur; bu en iyi sınır ya da bölgeye köprü adı verilir.

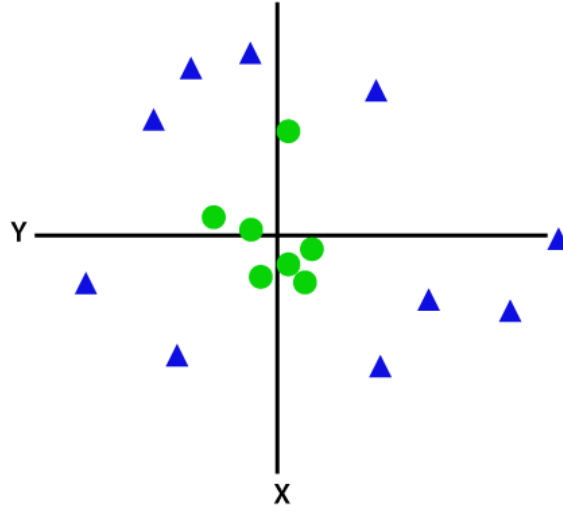
SVM algoritması her iki sınıftan da en yakın çizgiyi bulur. Bu noktalara destek vektörleri denir. Vektörler ve hiper düzlem arasındaki mesafeye kenar boşluğu denir. Ve SVM'nin amacı bu marjı en üst düzeye çıkarmaktır. Maksimum kenar boşluğuna sahip hiper düzlem, optimal hiper düzlem olarak adlandırılır [2].



Şekil 4.8. Doğrusal SVM [3]

4.3.3.4.2. Doğrusal olmayan svm

Veriler doğrusal olarak düzenlenmişse, düz bir çizgi kullanarak ayırabiliriz, ancak doğrusal olmayan veriler için tek bir düz çizgi çizemeyiz. Resimdeki gibi:

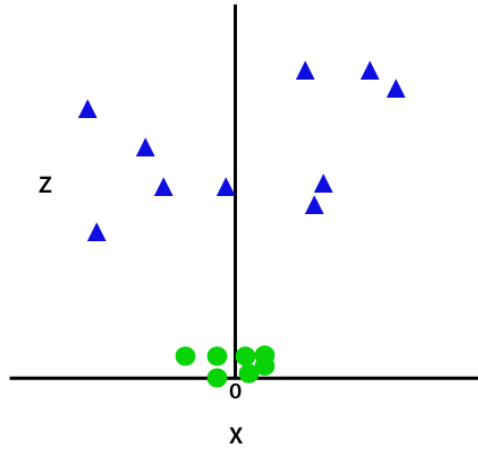


Şekil 4.9. Doğrusal olmayan SVM [3]

Bu veri noktalarını ayırmak için bir boyut daha eklememiz gerekiyor. Doğrusal veriler için iki boyut x ve y kullandık, bu nedenle doğrusal olmayan veriler için üçüncü bir boyut z ekleyeceğiz. Şu şekilde hesaplanabilir [3].

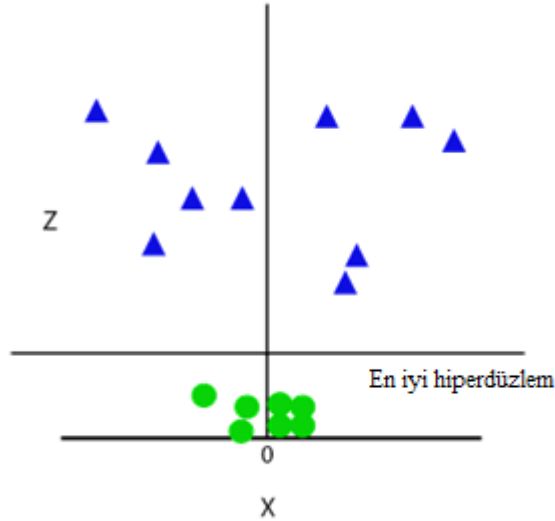
$$z = x^2 + y^2 \quad (5.3)$$

Üçüncü boyutu eklediğinizde, örnek alanı aşağıdaki gibi olacaktır:



Şekil 4.10. Doğrusal olmayan SVM [3]

Şimdi, SVM veri kümelerini aşağıdaki şekilde sınıflara ayıracaktır. Aşağıdaki resmi düşünün:

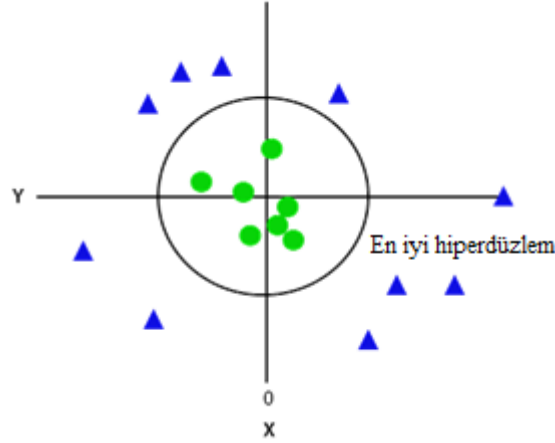


Şekil 4.11. Doğrusal olmayan SVM [3]

SVM'de, bu iki sınıf arasında doğrusal bir hiper düzlemin olması kolaydır. Ancak, ortaya çıkan bir başka yanma sorusu, hiper düzlemi elde etmek için bu özelliği manuel olarak eklememiz gerektiğidir. Hayır, SVM'nin çekirdek hilesi adı verilen bir tekniği vardır.

Bunlar, düşük boyutlu girdi alanını alan ve daha yüksek boyutlu bir alana dönüştüren işlevlerdir, yani ayrılmaz problemi ayrılabilir probleme dönüştürür, bu işlevlere çekirdek denir. Çoğunlukla doğrusal olmayan ayırma probleminde faydalıdır. Basitçe söylemek gerekirse, son derece karmaşık veri dönüşümleri yapar, daha sonra verileri tanımladığınız etiketlere veya çıktılara göre ayırma işlemini öğrenin [2].

Üç boyutlu uzayda bulunduğumuz için x eksenine paralel bir düzlem gibi görünüyor. Eğer onu 2d uzayda $z = 1$ ile dönüştürürsek, o zaman orijinal girdi uzayındaki hiperdüzleme baktığımızda, bir daireye benziyor ya da doğrusal olmayan verilerde 1 yarıçapı bir çevre elde ettiğimizi söyleyebiliriz [3].



Şekil 4.12. Doğrusal olmayan SVM [3]

4.3.3.5. Ayar parametreleri: çekirdek, düzenleme, gama ve marj

Makine öğrenimi algoritmaları için parametre değerini ayarlamak, model performansını etkili bir şekilde artırır. Bunları değiştirerek, makul miktarda daha fazla doğrulukla hatırı sayılır derecede doğrusal olmayan sınıflandırma çizgisine ulaşabiliriz. SVM ile kullanılabilen parametreler listesine bakalım.

Çekirdek

Doğrusal SVM'de hiper düzlemin öğrenilmesi, problemin bir miktar doğrusal cebir kullanılarak dönüştürülmesi ile yapılır. Çekirdek burada rol oynar. Doğrusal çekirdek için, girdi (x) ile her destek vektörü (x_i) arasındaki nokta ürünü kullanarak yeni bir girdi için tahmin denklemi aşağıdaki gibi hesaplanır:

$$f(x) = B(0) + \text{toplam}(a_i * (x, x_i)) \quad (5.4)$$

Bu, eğitim verilerindeki tüm destek vektörleri ile yeni bir giriş vektörünün (x) iç ürünlerinin hesaplanmasını içeren bir denklemdir. B_0 ve a_i katsayıları (her girdi için) öğrenme algoritması ile eğitim verilerinden tahmin edilmelidir.

Düzenlileştirme

Düzenleme parametresi (genellikle python'un sklearn kütüphanesinde C parametresi olarak adlandırılır) SVM optimizasyonuna her bir eğitim örneğini yanlış sınıflandırmaktan ne kadar kaçınmak istediğinizi söyler.

Büyük C değerleri için, eğer o hiper düzlem tüm eğitim noktalarını doğru bir şekilde sınıflandırmak için daha iyi bir iş çıkarırsa, optimizasyon daha küçük bir marjlı köprü seçecektir. Tersine, çok küçük bir C değeri, hiper düzlem daha fazla noktayı yanlış sınıflandırsa bile, optimize edicinin daha büyük bir marj ayıran hiper düzlem aramasına neden olur.

Gama

Gama parametresi, tek bir eğitim örneğinin etkisinin ne kadar uzağa ulaştığını tanımlar, düşük değerler "uzak" anlamına gelir ve yüksek değerler "yakın" anlamına gelir. Başka bir deyişle, düşük gama ile, ayırma çizgisi için hesaplamada makul ayırma çizgisinden uzak noktalar dikkate alınır. Yüksek gama anlamına gelirse, makul çizgiye yakın noktalar hesaplamada dikkate alınır.

kenar

Bir kenar boşluğu, çizginin en yakın sınıf noktalarına ayrılmasıdır. İyi bir fark, bu ayırımın her iki sınıf için de daha büyük olduğu bir farktır. Aşağıdaki resimler iyi ve kötü marjın görsel örneğini vermektedir. İyi bir marj, puanların diğer sınıfa geçmeden kendi sınıflarında olmasına izin verir [7].

4.3.4. Lojistik regresyon

Lojistik regresyon, Denetimli Öğrenme tekniği ile gelen en popüler Makine Öğrenimi algoritmalarından biridir. Y değişkeni ikili kategorik olduğunda kullanılan öngörülü bir modelleme algoritmasıdır. Yani, sadece 1 veya 0 gibi iki değer alabilir.

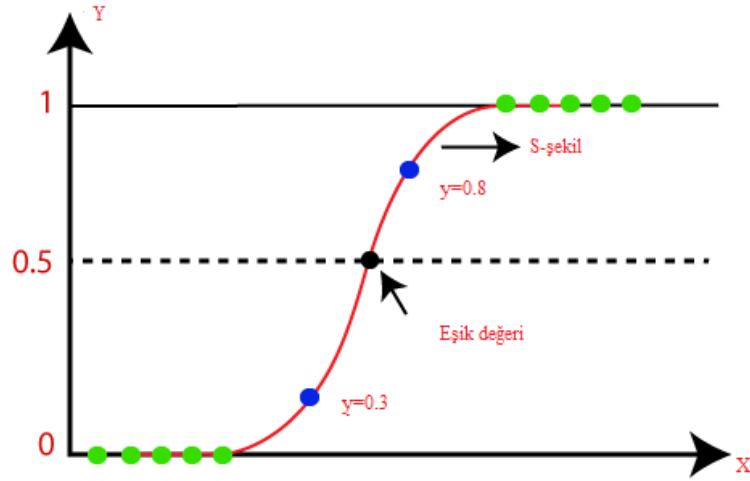
Amaç, olay 1'in olasılığını tahmin etmek için kullanılabilecek bir matematiksel denklemi belirlemektir. Denklem oluşturulduktan sonra, sadece Y'yi tahmin etmek için kullanılabilir. X'ler bilinmektedir.

Lojistik Regresyonu Doğrusal Regresyon modeli olarak adlandırabiliriz, ancak Lojistik Regresyon daha karmaşık bir maliyet fonksiyonu kullanır, bu maliyet fonksiyonu 'Sigmoid fonksiyonu' olarak tanımlanabilir veya doğrusal fonksiyon yerine 'lojistik fonksiyon' olarak da bilinir. Lojistik regresyon hipotezi, maliyet fonksiyonunu 0 ile 1 arasında sınırlama eğilimindedir. Bu nedenle doğrusal fonksiyonlar, lojistik regresyon hipotezine göre mümkün olmayan, 1'den büyük veya 0'dan küçük bir değere sahip olabileceği için bunu temsil edemez.

Lojistik Regresyon, nasıl kullanıldığı dışında Lineer Regresyona çok benzerdir. Doğrusal Regresyon Regresyon problemlerini çözmek için kullanılırken, Lojistik regresyon sınıflandırma problemlerini çözmek için kullanılır [36].

Lojistik regresyonda, bir regresyon çizgisini yerleştirmek yerine, iki maksimum değeri (0 veya 1) tahmin eden "S" şeklinde bir lojistik fonksiyona uyarız. Lojistik fonksiyonundan gelen eğri, hücrelerin kanserli olup olmadığı, bir farenin obez olup olmadığı, ağırlığına dayanma vb.

Lojistik Regresyon, sürekli ve ayrık veri kümeleri kullanarak olasılıklar sağlama ve yeni verileri sınıflandırma yeteneğine sahip olduğundan önemli bir makine öğrenme algoritmasıdır. Lojistik Regresyon, gözlemleri farklı veri türleri kullanarak sınıflandırmak için kullanılabilir ve sınıflandırma için kullanılan en etkili değişkenleri kolayca belirleyebilir. Aşağıdaki görüntü lojistik işlevini göstermektedir:

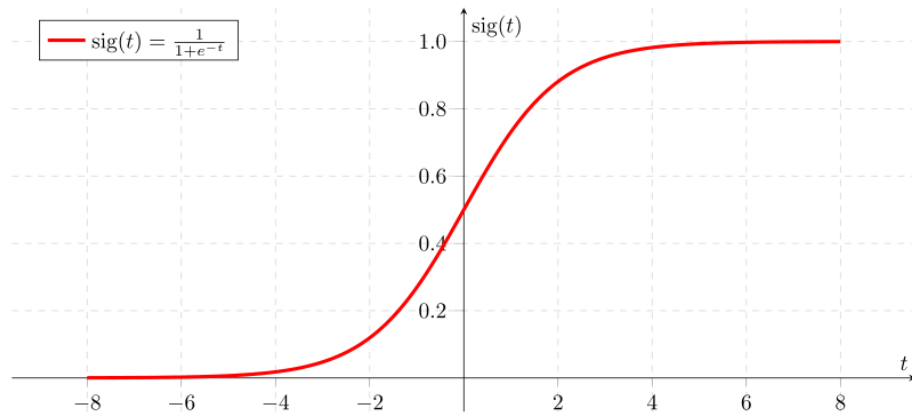


Şekil 4.13. Lojistik fonksiyonu [36]

4.3.4.1. Sigmoid işlevi

Sigmoid işlevi / lojistik işlevi, bir grafik üzerine çizildiğinde “S” şeklinde bir eğriye benzeyen bir işlevdir. 0 ile 1 arasında değerler alır ve bunları üst ve alt kenar boşluklarına doğru “ezer” ve 0 veya 1 olarak etiketler [37]. Sigmoid işlevi için denklem şudur:

$$\text{Sigmoid}(x) = \frac{1}{1+e^x} \quad (5.5)$$



Şekil 4.14. Sigmoid aktivasyon fonksiyonu [38]

- a) Sigmoid işlevi, öngörülen değerleri olasılıklarla eşleştirmek için kullanılan bir matematik işlevidir.
- b) Herhangi bir gerçek değeri 0 ve 1 aralığında başka bir değere eşler.
- c) Lojistik regresyonun değeri 0 ile 1 arasında olmalıdır, bu sınırın ötesine geçemez, bu nedenle "S" formu gibi bir eğri oluşturur. S-form eğrisine Sigmoid işlevi veya lojistik işlevi denir.
- d) Lojistik regresyonda, 0 veya 1 olasılığını tanımlayan eşik değeri kavramını kullanıyoruz. Eşik değerinin üzerindeki değerler 1'e ve eşik değerlerinin altındaki bir değer 0'a eğilimlidir [38].

4.3.4.2. Lojistik regresyon türü

Kategoriler temelinde, Lojistik Regresyon üç tipte sınıflandırılabilir:

- Binom: Binom Lojistik regresyonunda, 0 veya 1, Pass veya Fail, vb. Gibi sadece iki olası bağımlı değişken türü olabilir.
- Çok terimli: Çok terimli Lojistik regresyonunda, "kedi", "köpekler" veya "koyunlar" gibi bağımlı değişkenin 3 veya daha fazla olası sırasız türü olabilir.
- Sıralı: Sıralı Lojistik regresyonda, "düşük", "Orta" veya "Yüksek" gibi 3 veya daha fazla olası sıralı bağımlı değişken türü olabilir [36].

4.3.4.3. İkili sınıflandırma problemleri

Lojistik regresyonu ne tür problemler için kullanabileceğinizi merak edebilirsiniz. İşte ikili sınıflandırma problemlerine bazı örnekler:

- Spam Algılama
- Kredi Kartı Dolandırıcılığı
- Sağlık
- Pazarlama
- Bankacılık

4.3.4.4. Lojistik regresyon denklemi

Lojistik regresyon denklemi, Lineer Regresyon denkleminden elde edilebilir. Lojistik Regresyon denklemlerini elde etmek için matematiksel adımlar aşağıda verilmiştir: Düz çizginin denkleminin şöyle yazılabileceğini biliyoruz: [36].

$$Y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \dots + b_nx_n \quad (5.6)$$

Lojistik Regresyonda y sadece 0 ile 1 arasında olabilir, bu nedenle yukarıdaki denklemi $(1-y)$ ile bölelim:

$$\frac{y}{1-y}; 0 \text{ for } y=0, \text{ ve sonsuzluk for } y=1 \quad (5.7)$$

Ama - [sonsuzluk] ile + [sonsuzluk] arasında bir aralığa ihtiyacımız var, sonra o olacak denklemin logaritmasını ele alalım:

$$\log \left[\frac{y}{1-y} \right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \dots + b_nx_n \quad (5.8)$$

Yukarıdaki denklem Lojistik Regresyon için son denklemdir.

4.3.4.5. Lojistik regresyon kullanımı

Lojistik regresyonun sadece hedef değişkenler ayrık kategorilere düştüğünde kullanılması ve hedef değer bir dizi sürekli değer olması durumunda lojistik regresyonun kullanılmaması gerektiğini anlamak önemlidir. Lojistik regresyonu kullanabileceğiniz durumlara örnekler:

- Bir e-postanın spam olup olmadığını tahmin etme
- Bir tümörün kötü huylu veya iyi huylu olup olmadığı
- Bir mantarın zehirli veya yenilebilir olup olmadığı.

Lojistik regresyon kullanılırken, örneğin bir sınıfa diğer sınıfa göre hangi değerde konulacağını gösteren bir eşik değeri genellikle belirtilir. Spam sınıflandırma görevinde,% 50 veya daha fazla spam olma olasılığı olan bir e-postanın “spam” olarak sınıflandırılma olasılığı ve% 50'den daha az olasılığı olan herhangi bir e-postanın spam değil olarak sınıflandırılmasına neden olacak 0,5 eşiği ayarlanabilir.

Lojistik regresyon en iyi ikili sınıflandırma örnekleri için uygun olsa da, çok sınıflı sınıflandırma problemlerine, üç veya daha fazla sınıflı sınıflandırma görevlerine uygulanabilir. Bunu, “bire karşı” stratejisini uygulayarak başarabilirsiniz [37].

“Bire karşı tüm strateji” nedir? Diyelim ki veri kümenizdeki örneklerin içine girebileceği üç farklı sınıfınız var ve bu üç sınıfınız olsaydı bunları üç farklı ikili sınıflandırma sorunu olarak ele alabilirsiniz.

Örneğin, sadece A Sınıfına ait örnekler ve diğer tüm sınıflara ait tüm örnekler üzerinde bir sınıflandırıcı eğiteceksiniz. Aynı şeyi B Sınıfı ve son olarak C Sınıfı için de yapardınız. Sınıflandırıcılar, seçtikleri sınıfı diğer sınıflardan ayırt etmeyi öğrendikten sonra, girişlerde üç sınıflandırıcıyı çalıştırırsınız ve hangi sınıflayıcı en iyi olursa olsun, bu örnek için doğru sınıfı seçtiğinden emin olursunuz [37].

BÖLÜM 5. GERÇEK ÖRNEK

5.1. Giriş

Bu bölümde, sensörden veri toplamak için arduino kullanan bir gaz sensörü örneği sunacağız, daha sonra Tera terimiyle excel olarak göndereceğiz ve bize en iyi sınıflandırmayı vermek için bu verileri Weka ile sınıflandıracacağız. Bu kısmı yapmak için, bir CO2 gazı sensörü CDM4161a modeli ve arduino DUE kullandık.

5.2. Arduino DUE

Arduino Due, 32-bit ARM çekirdekli mikro denetleyiciye dayanan ilk Arduino kartıdır. 54 dijital giriş / çıkış pini, 12 analog giriş ile güçlü büyük ölçekli Arduino projeleri için mükemmel bir karttır.

Arduino Due, 54 dijital giriş/çıkış pinine (12'si PWM çıkışı olarak kullanılabilir), 12 analog girişe, 4 URT (seri donanım), 84 Mhz'lik saate, USB-OTG uyumlu bağlantıya, 2 DAC (dijitalden analoğa), 2 TWI, güç fişine, SPI başlığına, JTAG başlığına, reset ve silme butonuna sahiptir.

Diğer Arduino kartlarından farklı olarak, Arduino Due 3.3 V ile çalışır. 5V gibi yüksek voltajlarda giriş/çıkış pinleriniz zarar görebilir. 3.3V sinyal seviyesinde çalışan her arduino shield'i Due ile uyumludur [39].

5.2.1. Arm çekirdek faydaları

Due üzerinde 32-bitlik Arm çekirdek işlemci bulunmaktadır. En önemli faydaları;
- 32-bit işlemci bir işlemci saatinde 4 byte'lık veri işlemenize imkan sağlar.

- İşlemci 84 MHz saat hızında çalışır.
- 96 KBytes SRAM
- 512 KByte flash bellek
- DMA kontrolcü, işlemciyi bellek işlemlerini rahatlatır

5.2.2. Teknik özellikler

- Mikrodenetleyici AT91SAM3X8E
- Çalışma Gerilimi 3.3V
- Giriş Gerilimi (önerilen) 7-12V
- Giriş Gerilimi (limit) 6-16V
- Dijital I/O Pinleri 54 (12 tanesi PWM çıkışı)
- Analog Giriş Pinleri 12
- Analog Çıkış Pinleri 2 (DAC)
- Yüm I/O'ların Toplam Çıkış Akımı 130 mA
- 3.3V Çıkış için Akım 800 mA
- 5V Çıkış için Akım 800 mA
- Flash Hafıza 512 KB (SAM3X8E)
- SRAM 96 KB (SAM3X8E)
- Saat Hızı 84 MHz
- Uzunluk 101.52 mm
- Genişlik 53.3 mm
- Ağırlık 36 g [40].

5.2.3. Güç

Arduino Due gücünü usb üzerinden veya harici güç kaynağından alabilir. Harici güç kaynağı AC-DC adaptör olabileceği gibi bataryada olabilir. Adaptör kart üzerindeki 2.1mm merkez-pozitif güç soketinden bağlanabilir. Batarya kart üzerindeki GND ve Vin pinleri üzerinden bağlanabilir.

Kartın çalışması için sürekli olarak usb'nin bağlı olması şart değildir. Kart sadece adaptör veya batarya ile çalıştırılabilir. Bu sayede kart bilgisayardan bağımsız olarak çalıştırılabilir.

Harici güç kaynağı olarak 6-20V arası kullanılabilir. Ancak bu değerler limit değerleridir. Kart için önerilen harici besleme 7-12V arasındadır. Çünkü kart üzerinde bulunan regülatör 7V altındaki değerlerde stabil çalışmayabilir. 12V üstündeki değerlerde de aşırı ısınabilir[40].

5.2.4. Hafıza

SAM3X8E 512 KB'lık flash belleğe sahiptir. 96 KB SRAM bulunmaktadır. SAM3X'in flash belleği kart üzerinde bulunan erase butonu ile silinebilir. Bu sadece yüklediğiniz arduino kodunu silecektir, bootloader yazılımında herhangi bir değişiklik olmayacaktır. Kartta güç varken erase tuşuna birkaç saniye basılı tutmanız yeterlidir.

5.2.5. Giriş ve çıkış

Due üzerindeki 54 adet dijital pinin hepsi giriş veya çıkış olarak kullanılabilir. 12 tane analog giriş pinide bulunmaktadır. Bu analog giriş pinleride aynı şekilde dijital giriş ve çıkış olarak kullanılabilir. Bu pinlerin tamamının lojik seviyesi 3.3V'dur. Kullanılan pine bağlı olarak çıkış akımı pin başına 3 mA veya 15 mA, giriş olarak da 6 mA veya 9mA'dir. Ek olarak, bazı pinlerin farklı özellikleri bulunmaktadır. Özel pinler aşağıda belirtildiği gibidir [40].

- PWM, 2-13: 8-bit çözünürlükte PWM çıkış pinleri olarak kullanılabilir. PWM çözünürlüğü analogWriteResolution() fonksiyonu kullanılarak değiştirilebilir.
- SPI, ICSP Header: Bu pinler SPI haberleşmesi için kullanılır. SPI portu 3×2'lik ICSP header'ı ile dışarı alınmıştır. Ancak diğer arduino kartlarının aksine Due'de bu pinler ICSP tekniği ile SAM3X işlemcisini programlamak için kullanılamaz.

Due üzerindeki SPI portunun bazı ileri özellikleri bulunmaktadır, Extended SPI methods for Due sayfasını inceleyebilirsiniz.

5.2.6. Haberleşme

Arduino Due'nın bilgisayarla, başka bir arduino veya mikrodenetleyici ile veya telefon, tablet gibi cihazlar ile haberleşmesi için bir kaç farklı seçenek vardır. SAM3X8E, 4 tane donanımsal seri port TTL (3.3V) seri haberleşme imkanı sunar.

Programming Port kart üzerinde bulunan Atmega16u2 usb-seri dönüştürücüsüne bağlıdır ve bilgisayarda sanal bir com port açarak SAM3X8E ile bilgisayar arasında bir köprü kurar. Due kartı arduino yazılımı ile programlanırken Programming Port kullanılması önerilir. Pin 0 ve 1 doğrudan kart üzerinde bulunan Atmega16u2 usb-seri dönüştürücüsüne bağlıdır. Yani bilgisayardan karta kod yüklerken veya bilgisayar-due arasında karşılıklı haberleşme yapılırken bu pinler kullanılır. O yüzden karta kod yüklerken veya haberleşme yapılırken hata olmaması için mecbur kalınmadıkça bu pinlerin kullanılmamasında fayda vardır [41].

5.3. Co2 Gaz Sensörü CDM4161a

CDM4161A, Figaro'nun kompakt ve düşük güç tüketimli katı elektrolit CO2 sensörü olan TGS4161 kullanan son derece minyatürleştirilmiş, önceden kalibre edilmiş bir CO2 ünitesidir. Figaro'nun bir mikrobilgisayarla sinyal işleme fikri fikri nedeniyle, bu modül için herhangi bir bakım gerekmez. IR sensörleri kullanan geleneksel CO2 sensör modülleri ile karşılaştırıldığında, Figaro'nun CO2 modülü çok daha düşük maliyetlidir, bu da bu modülü iç mekan hava kalitesi kontrol sistemleri için ideal seçim haline getirir. İç hava kalitesi kontrolü için kullanılan bu sensör [42].

5.3.1. Özellikleri

- Ultra kompakt boyut
- CO2'ye yüksek seçicilik
- Bakım gerektirmeyen

- Düşük güç tüketimi
- Uzun yaşam
- Önceden kalibre
- Düşük maliyetli

5.3.2. Giriş çıkış sinyali

5.3.2.1 Pin n.1

Bu porta ayarlanan voltaj girilmelidir. Giriş voltajı dalgalanırsa, sensörün çıkışı sensörün voltaj bağımlılık özelliklerine göre değişebilir.

5.3.2.2 Pin n.2 konsantrasyon çıkışı ve sorun sinyali çıkışı

Bu bağlantı noktasının çıkış sinyaline bağlı olarak iki rolü vardır: konsantrasyon çıkışı veya sorun sinyali çıkışı. Normal çalışma koşullarında, CO₂ konsantrasyonuna karşılık gelen bir analog voltaj (CO₂ konsantrasyonu / 1000) 0.4 ila 4.0V DC aralığında çıkarılır. Sensörün çıkışı anormalse, bu pinden bir DÜŞÜK sinyali çıkacaktır [43].

5.3.2.3 Pin n.3 kontrol sinyali çıkışı

CO₂ konsantrasyonları 1000 ppm CO₂'yi aştığında, bu porttan YÜKSEK bir sinyal çıkacaktır.

5.3.2.4 Pin n.4 sinyal girişini sıfırla

Temel değer, bu giriş sinyali kullanılarak sıfırlanabilir. YÜKSEK giriş sinyali ile, o anda sensörün çıkışı 400ppm CO₂ (temiz havada temel değer) olarak hafızaya alınır. Isınma süresi sırasında bir YÜKSEK sinyal girildiğinde (ilk iki içinde) açıldıktan sonra, temel voltaj o anda okunur ve modül derhal çalışma moduna geçer [43].

5.3.2.5 Pin n.5

Kaynak güç voltajı (pim) (VSS; GND) NMOS, CMOS ve TTL aygıtları için toprak referans voltaj pini, genellikle diğer tüm aygıt pimleri için referans pini. VSS normalde sistem topraklamasıdır ve VSS terimi genellikle GND terimi ile dönüşümlü olarak kullanılır.

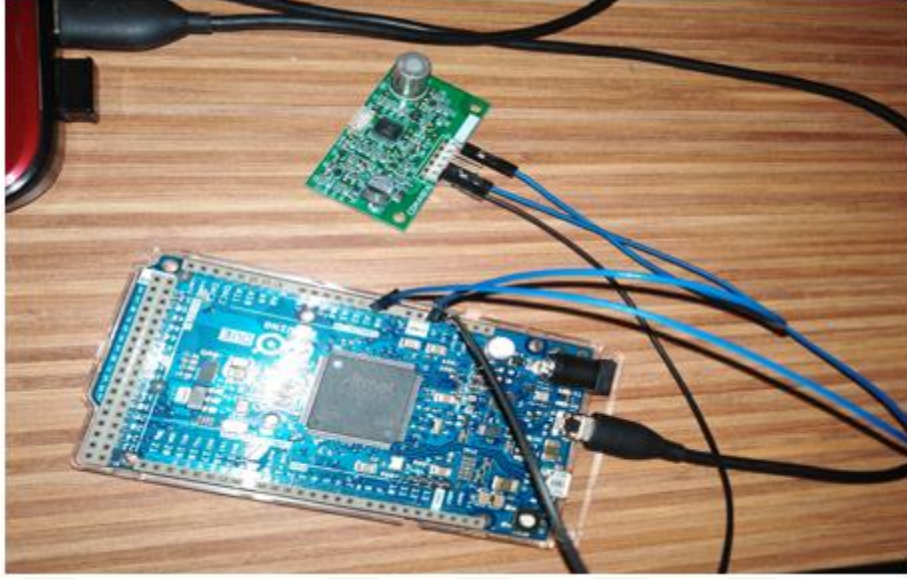
5.4. Tera Term

Tera Term açık kaynaklı, ücretsiz, yazılımla uygulanan bir terminal emülatör (iletişim) programıdır. DEC VT100'den DEC VT382'ye kadar farklı bilgisayar terminalleri öykünür. Telnet, SSH 1 ve 2 ve seri port bağlantılarını destekler. Sayfa terimini excelelemek için Arduino verilerini almak için tera terimini kullanıyoruz [44].

5.5. Pratik Kısım

Bu bölümde şunları kullanacağız:

- Arduino kart DUE
- CO2 dioksit de karbon sensörü
- Veri kablosu
- Kod için Arduino IDE
- Tera Term
- Excel
- Weka



Şekil 5.1. Arduino ve CO2 sensor

İlk Önce Arduino kartını aşağıdaki gibi dioksit de karbon modülü sesnor ile bağlarız:

5v --> vin pin

GND --> GND

A0 --> Vconc.

İkincisi, arduino'yu veri kablosuyla bilgisayara bağlarız. Üçüncüsü, Arduino IDE ile kodlama kısmını geçiyoruz.



```

Gas_sensor | Arduino 1.8.12
Fichier Édition Croquis Outils Aide

Gas_sensor$
int pin5 = 5;
int sensor = A0;
int sensorValue = 0;
void setup() {
  // Initialize the digital pin 5 as an output
  pinMode(pin5, OUTPUT);
  // Initialize serial communication at 9600 bits per second
  Serial.begin(9600);
  Serial.println("Date & Time, co2 $");
}

void loop() {
  delay(1000);
  sensorValue = analogRead(sensor);
  Serial.println(sensorValue, DEC);
  // If sensorValue is greater than 500
  if (sensorValue > 500) {
    digitalWrite(pin5, HIGH);
  }
  else {
    digitalWrite(pin5, LOW);
  }

  Serial.print(",");
  Serial.println(sensorValue);
}

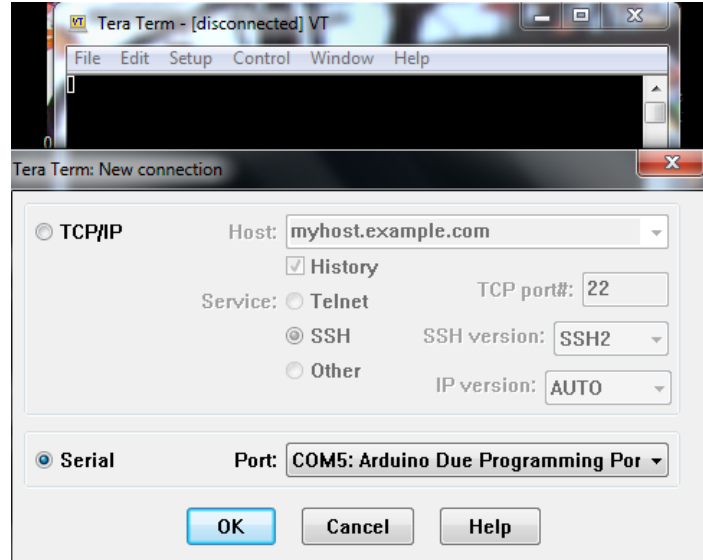
Téléversement terminé
CPU reset.
Erreur durant le téléchargement de https://downloads.arduino.cc/pa

```

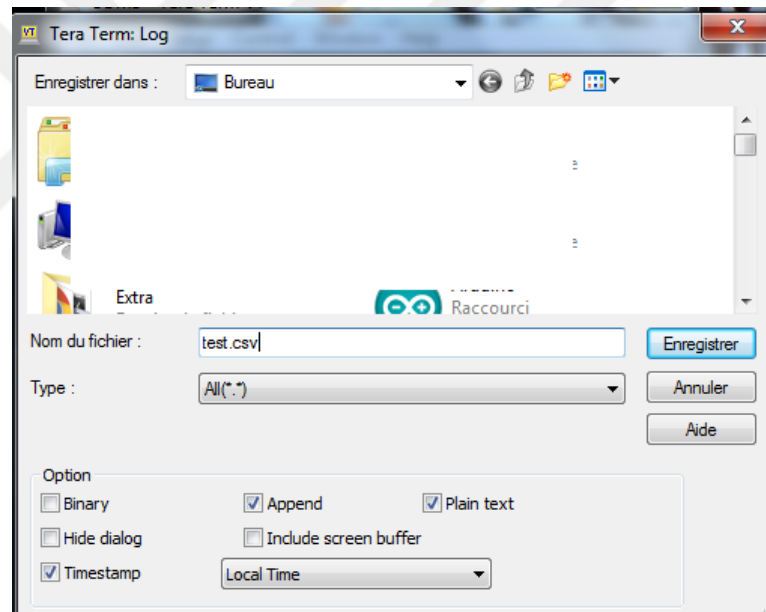
Şekil 5.2. Arduino ide kodu

İlk önce kodun yazılmasından sonra taslağımızı yüklüyoruz, ikinci olarak tera terim yazılımını açıyoruz, COM5 Arduino olan seri ve portu seçiyoruz ,aşağıdaki şekil 3'te görebiliriz. Üçüncüsü, bir excel sayfası oluşturmak için dosyamızı .CSV ile kaydediyoruz ve veri setimizde bize zaman vermek için kutu timestimp'ini kontrol ediyoruz şekil 4'te açıkladı.

Sonra odamızda co2 ölçümüne başlamak için arduino'daki araştırma düğmesine tıklıyoruz. Okuma yaptıktan sonra tera terim yazılımını kapatabilirsiniz. Masaüstüne gidiyoruz ve kaydedilen dosyayı açıyoruz. ve şekil 5'de görebileceğimiz veri seti csv dosyamız var.



Şekil 5.3. Tera term veri çıkmak için ayarlar



Şekil 5.4. Csv uzantılı dosya kaydetme

	A	B		
			7290	[2020-04-14 20:24:04.078]
1	Date & time	CO2 (ppm)	7291	[2020-04-14 20:24:04.569]
2	[2020-04-14 15:54:19.440]	460	7292	[2020-04-14 20:24:04.576]
3	[2020-04-14 15:54:19.492]	460	7293	[2020-04-14 20:24:05.568]
4	[2020-04-14 15:54:19.493]	460	7294	[2020-04-14 20:24:05.577]
5	[2020-04-14 15:54:19.494]	460	7295	[2020-04-14 20:24:06.565]
6	[2020-04-14 15:54:19.495]	460	7296	[2020-04-14 20:24:06.575]
7	[2020-04-14 15:54:19.544]	459	7297	[2020-04-14 20:24:07.568]
8	[2020-04-14 15:54:19.545]	460	7298	[2020-04-14 20:24:07.574]
9	[2020-04-14 15:54:19.546]	460	7299	[2020-04-14 20:24:08.568]
10	[2020-04-14 15:54:19.547]	460	7300	[2020-04-14 20:24:08.573]
11	[2020-04-14 15:54:19.548]	459	7301	[2020-04-14 20:24:09.567]
12	[2020-04-14 15:54:19.548]	460	7302	[2020-04-14 20:24:09.573]
13	[2020-04-14 15:54:19.583]	460	7303	[2020-04-14 20:24:11.431]
14	[2020-04-14 15:54:19.584]	460	7304	[2020-04-14 20:24:11.432]
15	[2020-04-14 15:54:19.585]	460	7305	[2020-04-14 20:24:11.566]
16	[2020-04-14 15:54:19.586]	460		
17	[2020-04-14 15:54:19.586]	459		

Şekil 5.5. Excelde veri seti

Veri seti CO2 gaz sensörü CDM4161a'dan 7310 ölçüm içerir, Pencerelelerin açıldığı sabahları ve odada spor yaptığımız geceleri farklı saatlerde ölçülür (15:54 ve 20: 24 arası). biz farklı co2 ölçüsü var: 460, 455, 453, 320, 1023 ppm vb ..

Veri setimizi csv uzantılı excel sayfasında aldıktan sonra, bu uzantıyı weka'daki bir sonraki " sınıflandırma " için arff uzantısına değiştirmeliyiz. Dosyanın uzantısını çevrimiçi olarak değiştirebiliriz ya da weka'da sorun değil.

Weka yazılımını açtığımız dosyanın uzantısını değiştirdikten sonra dosyamızı açıyoruz, daha sonra sınıflandırmamıza son bölümde açıklanan farklı sınıflandırma algoritması ile başlıyoruz.

Farklı algoritma: KNN, Rastgele orman ve destek vektör makinesi algoritmaları. Bu üç algoritmanın veri setimizde yürütülmesinden sonra,% 95 doğru sınıflandırılmış örneklerle rastgele orman algoritması ile en iyi sınıflandırma.

BÖLÜM 6. SENSOR GAZLARIN MAKİNE ÖĞRENMESİ ALGORİTMALARI İLE SINIFLANDIRILMASI

6.1. Giriş

Bu çalışmada Weka Machine Learning yazılımını kullanacağız. Weka, veri ön işleme, sınıflandırma, regresyon, kümeleme ve daha birçok araç için çeşitli araçlar içerir. Dolayısıyla bu çalışmada verilerimiz için sınıflandırma yapacağız ve hangisinin en iyi sonucu verecek en iyi sınıflandırma algoritması olduğunu göreceğiz.

6.2. Weka

Yeni Zelanda Waikato Üniversitesi'nde geliştirilen Waikato Bilgi Analizi Ortamı. GNU Genel Kamu Lisansı altında lisanslı özgür yazılım ve "Veri Madenciliği: Pratik Makine Öğrenme Araçları ve Teknikleri" kitabına eşlik eden yazılımdır.

Weka, veri madenciliği görevleri için bir makine öğrenimi algoritması koleksiyonudur. Veri hazırlama, sınıflandırma, regresyon, kümeleme, ilişkilendirme kuralları madenciliği ve görselleştirme için araçlar içerir [45].

6.2.1. Weka veri formatları

Weka, veri analizi için varsayılan olarak Öznitelik İlişkisi Dosya Biçimini kullanır. Ancak weka, ODBC kullanan CSV ve veritabanı gibi başka bir dosya türünü destekler. Öznitelik İlişkisi Dosya Biçimi (ARFF); bu biçimin iki bölümü vardır:

üstbilgi bölümü şunları tanımlar: ilişki (veri kümesinin adı), öznitelik adı ve türü. Veri bölümü veri örneklerini listeler. Aşağıdaki şekilde gördüğümüz gibi:

@relation:

Bu, herhangi bir ARFF dosyasındaki başlık bölümünde yazılan, ardından ilişki / veri kümesi adı olan ilk satırdır. İlişki adı bir dize olmalı ve boşluk içeriyorsa, tırnak işaretleri arasına alınmalıdır.

@attribute:

Bunlar adları ve başlık bölümündeki türü veya aralığıyla bildirilir.

@data-Veri bölümünde ve ardından tüm veri segmentlerinin listesinde tanımlanır[46].

```
@relation weather
@attribute outlook {sunny, rainy}
@attribute temperature real
@attribute windy {TRUE, FALSE}

@data
sunny,85,FALSE
sunny,80,TRUE
rainy,70,FALSE
rainy,68,FALSE
rainy,71,TRUE
```

Şekil 6.1. ARFF biçimi

Bu, Weka'daki ARFF dosyasının bir örneğidir; bu, ilk bölümde "hava durumu" veri kümesi adını içeren iki bölüme sahiptir ve sıcaklık gibi türlerine sahip üç öznitelik, gerçek bir tür değerine sahiptir. ve ikinci bölüm veri kümesi örneklerini içerir. Örneğin, hava güneşli olduğunda, sıcaklık = 85 ve rüzgarlı olduğunda ilk satırı alırız.

6.2.2. Weka sınıflandırması

Weka, çok sayıda sınıflandırma algoritmasını kullanıma sunar. İnceleyeceğimiz algoritmalar:

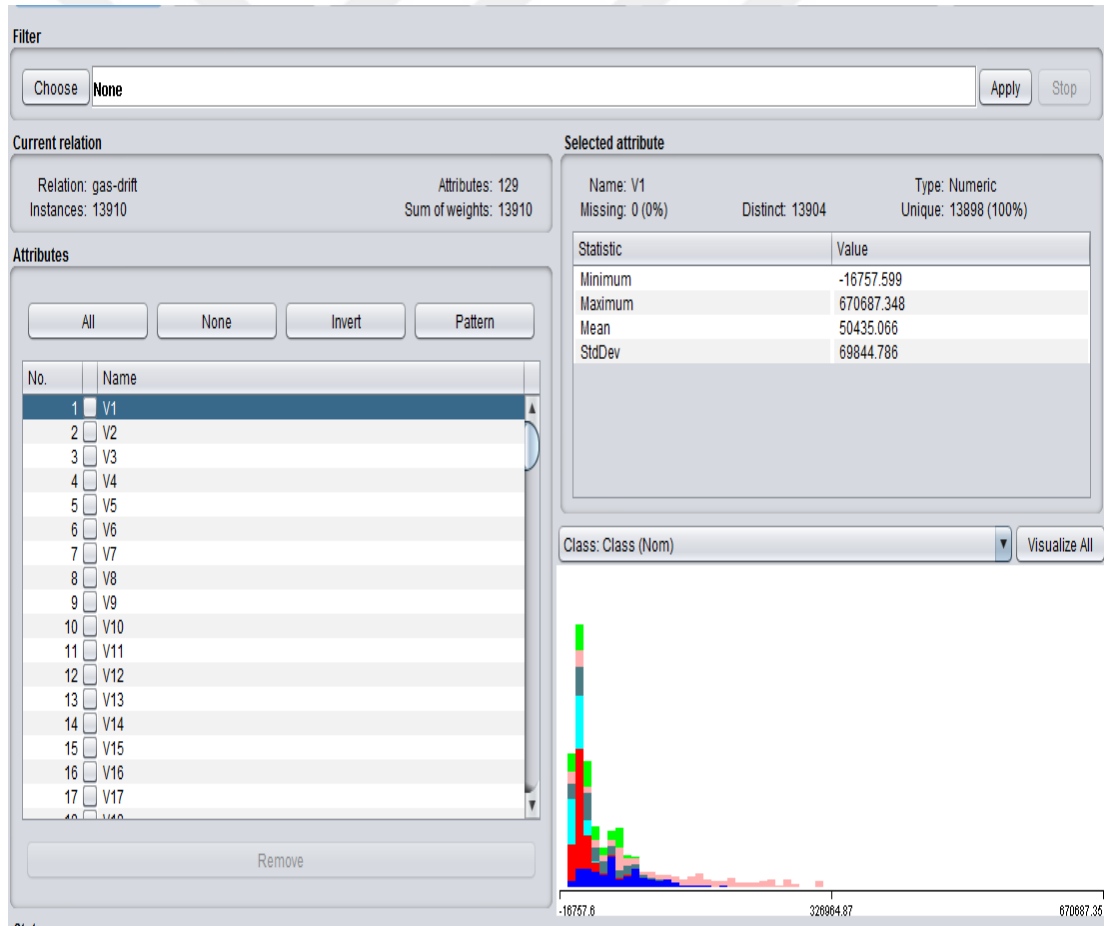
- Lojistik regresyon

- Naif bayes
- Karar ağacı
- k-En Yakın Komşular
- Vektör makineleri desteklemek

Çalışmamızda kullandığımız tüm bu algoritmalar daha sonra açıklayacağız.

6.3. Veri Kümesi

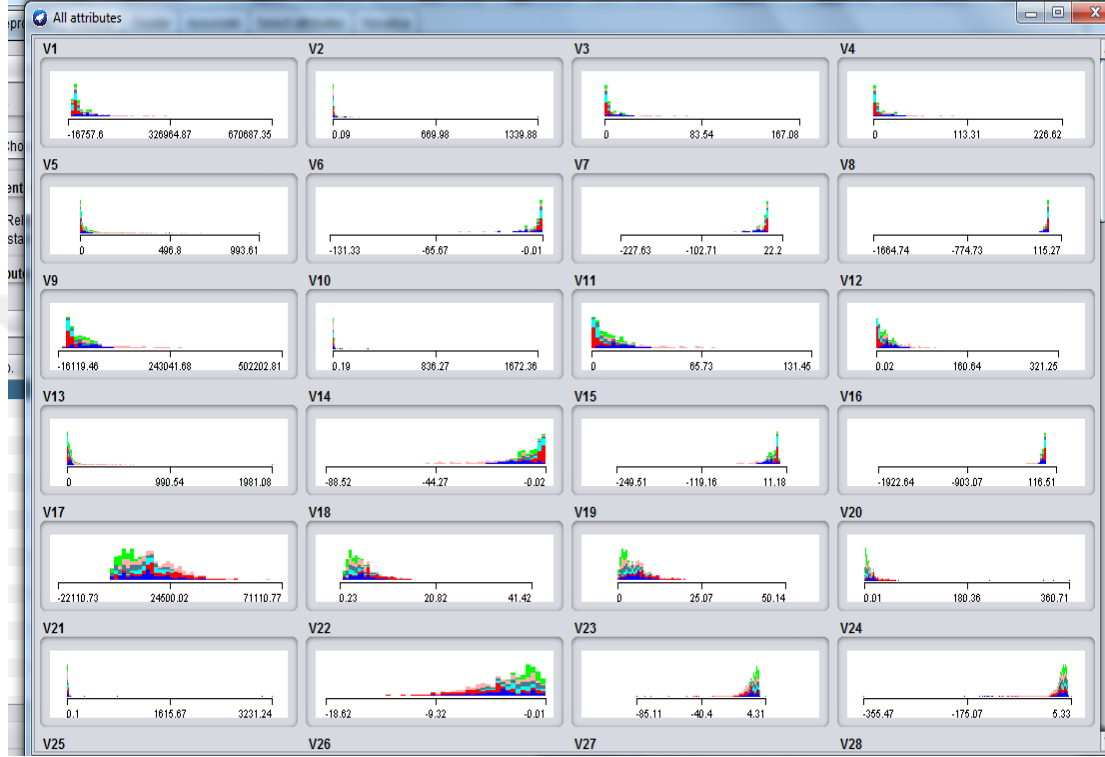
Kullanılan veri kümesi ikinci bölümde tanımlanmıştır veri kümesini UCI'den aldık [47].Tüm algoritma için girdi, gaz tipi drift.arff dosyasındaki aynı veri kümesidir.



Şekil 6.2. Gaz sensörü veri seti yüklü Weka Explorer Arayüzü

Öznitelik Dağılımları

"Tümünü Görselleştir" düğmesini tıklayın ve her bir özelliğin grafik dağılımını gözden geçirelim.



Şekil 6.3. Weka Tek Değişkenli Öznitelik Dağıtım Grafikleri

6.4. WEKA'da Sınıflandırma Adımları:

Her şeyden önce:

- Weka GUI Seçici'yi açın.
- Weka Explorer'ı açmak için "Explorer" düğmesine tıklayın.
- Verfi kümesini data.arff dosyasından yükleyin.
- Sınıflandır sekmesini açmak için "Sınıflandır" a tıklayın.
- "Seç" düğmesini tıklayın ve kullanmak istediğiniz algoritmayı seçin Algoritmayı çalıştırmak için "Başlat" düğmesine tıklayın.

Ve son olarak, bu veri kümesinde kullanılan algoritmanın sonucuna ve çıktısına sahipsiniz. İşte başlıyoruz: ilk olarak ZeroR algoritması ile başladık çünkü varsayılan olarak kontrol edilen ilk algoritmadır.

6.4.1. ZeroR

Sıfır Kuralı olarak da adlandırılan ZeroR algoritması, veri kümenizdeki tüm algoritmaların performans taban çizgisini hesaplamak için kullanabileceğiniz bir algoritmadır. 0-R sınıflandırıcısı, hedef niteliğe ve olası değerlerine bakar. Her zaman verilen veri kümesinde hedef özellik için en sık bulunan değeri verir. 0-R isminden de anlaşılacağı gibi; hedef olmayan özellikler üzerinde çalışan herhangi bir kural içermez.

Aşağıdaki şekil, weka'da sınıflandırma algoritmasının çıktısını sunar, çalışma bilgisine sınıflayıcı model, özet ve karışıklık matrisini verir.

The screenshot shows the Weka software interface with the 'Classifier' tab selected. The 'ZeroR' classifier is chosen. The 'test options' section shows 'Cross-validation' selected with 'Folds' set to 100. The 'Classifier output' panel displays the following information:

```

=== Run information ===
Scheme:      weka.classifiers.rules.ZeroR
Relation:    gas-drift
Instances:   13910
Attributes:  129
             [list of attributes omitted]
Test mode:   10-fold cross-validation

=== Classifier model (full training set) ===
ZeroR predicts class value: 5

Time taken to build model: 0.2 seconds

=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances      3009           21.6319 %
Incorrectly Classified Instances    10901          78.3681 %
Kappa statistic                     0
Mean absolute error                 0.2748
Root mean squared error             0.3707
Relative absolute error              100 %
Root relative squared error         100 %
Total Number of Instances          13910

```

Şekil 6.4. Gaz sensörü veri setindeki ZeroR algoritması için weka sonuçları

Test seçeneklerinin varsayılan olarak 10 katlamayla Çapraz Doğrulmayı seçtiğini de not edeceğiz. Bu, veri kümesinin 10 bölüme ayrıldığı anlamına gelir. İlk 9

algoritmayı eğitmek için kullanılır ve 10 tanesi algoritmayı değerlendirmek için kullanılır. Bu işlem tekrarlanır ve bölünmüş veri kümesinin 10 kısmının her biri, dışarıda bırakılan test seti olma şansı verir.

ZeroR algoritmasının% 21'lik bir doğruluk elde ettiğini görebiliyoruz ki bu, bu algo'nun kullanımının iyi olmadığı anlamına gelen en kötü sonuçtur.

```

=== Confusion Matrix ===
      a  b  c  d  e  f  <-- classified as
0  0  0  0 2565  0 |  a = 1
0  0  0  0 2926  0 |  b = 2
0  0  0  0 1641  0 |  c = 3
0  0  0  0 1936  0 |  d = 4
0  0  0  0 3009  0 |  e = 5
0  0  0  0 1833  0 |  f = 6

```

Şekil 6.5. Zeror algoritmasının karışıklık matrisi

6.4.2. Rastgele orman

```

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      13834      99.4536 %
Incorrectly Classified Instances     76         0.5464 %
Kappa statistic                     0.9934
Mean absolute error                  0.0119
Root mean squared error              0.0483
Relative absolute error              4.3133 %
Root relative squared error          13.029 %
Total Number of Instances           13910

```

Şekil 6.6. Rasgele orman algoritmasının sonuç yürütülmesi

=== Confusion Matrix ===

```

a   b   c   d   e   f   <-- classified as
2554  9   0   2   0   0 |   a = 1
2 2913   2   0   9   0 |   b = 2
0   6 1618   3  14   0 |   c = 3
0   0   0 1926   5   5 |   d = 4
1   2   3   3 2998   2 |   e = 5
2   1   0   4   1 1825 |   f = 6

```

Şekil 6.7. Rastgele ormanı algoritmasının karışıklık matrisi

Rastgele orman algoritması% 99'luk bir doğruluk sağlar ve bu da çok iyi bir sonuçtur.

Bu karışıklık matrisinde, tahmin edilen sınıflara kıyasla gerçek sınıfların bir tablosunu görebiliriz ve a'nın b, a'nın d ve b'nin d.... vb olarak sınıflandırıldığı bir hata olduğunu görebilirsiniz.

6.4.3. K-en yakın komşular

Classifier output

```

=== Classifier model (full training set) ===
IBk instance-based classifier
using 1 nearest neighbour(s) for classification

Time taken to build model: 0.07 seconds

=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances      13839      99.4896 %
Incorrectly Classified Instances     71         0.5104 %
Kappa statistic                     0.9938
Mean absolute error                  0.0018
Root mean squared error              0.0412
Relative absolute error              0.6673 %
Root relative squared error         11.126 %
Total Number of Instances           13910

=== Detailed Accuracy By Class ===

```

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,996	0,001	0,995	0,996	0,996	0,995	0,998	0,992	1
	0,996	0,001	0,996	0,996	0,996	0,995	0,998	0,993	2

Şekil 6.8. Gaz sensörü veri setindeki k- en yakın komşular algoritması için weka sonuçları

```

=== Confusion Matrix ===
      a   b   c   d   e   f  <-- classified as
2556   6   1   1   1   0 |   a = 1
  4 2915   2   0   5   0 |   b = 2
  1   4 1619   3  13   1 |   c = 3
  4   0   0 1930   0   2 |   d = 4
  2   1   9   5 2990   2 |   e = 5
  3   0   0   1   0 1829 |   f = 6

```

Şekil 6.9. K- en yakın komşular algoritmasının karışıklık matrisi

Algoritmanın doğruluğunu görmek için farklı K değeri denedik. K değerini arttırdığımızda algoritmanın doğruluğu azalır.

Tablo 6.1. K'nin farklı değeri

K number	Time	+%	-%
K=1	0,03	99,3	0,6
K=5	0,03	98,9	1,0
K=10	0,02	98,3	1,6
K=50	0,01	95,4	4,3
K=100	0,01	91,5	8,4

6.4.4. Lojistik regresyon

Logistik regresyon algoritması % 98, 28 'luk bir doğruluk sağlar ve bu daiyi bir sonuçtur.

```

Correctly Classified Instances      4648      98.2872 %
Incorrectly Classified Instances     81        1.7128 %
Kappa statistic                      0.9783

```

Şekil 6.10. Gaz sensörü veri setindeki lojistik regresyon algoritması için Weka Sonuçları

```

=== Confusion Matrix ===
      a  b  c  d  e  f  <-- classified as
862   5   3   4   4   0 |  a = 1
  0 990   2   3   4   3 |  b = 2
  0   3 557   0   9   1 |  c = 3
  5   0   0 635   6   0 |  d = 4
  0   4   9   4 966   4 |  e = 5
  4   0   0   2   2 638 |  f = 6

```

Şekil 6.11. Lojistik regresyon algoritmasının karışıklık matrisi

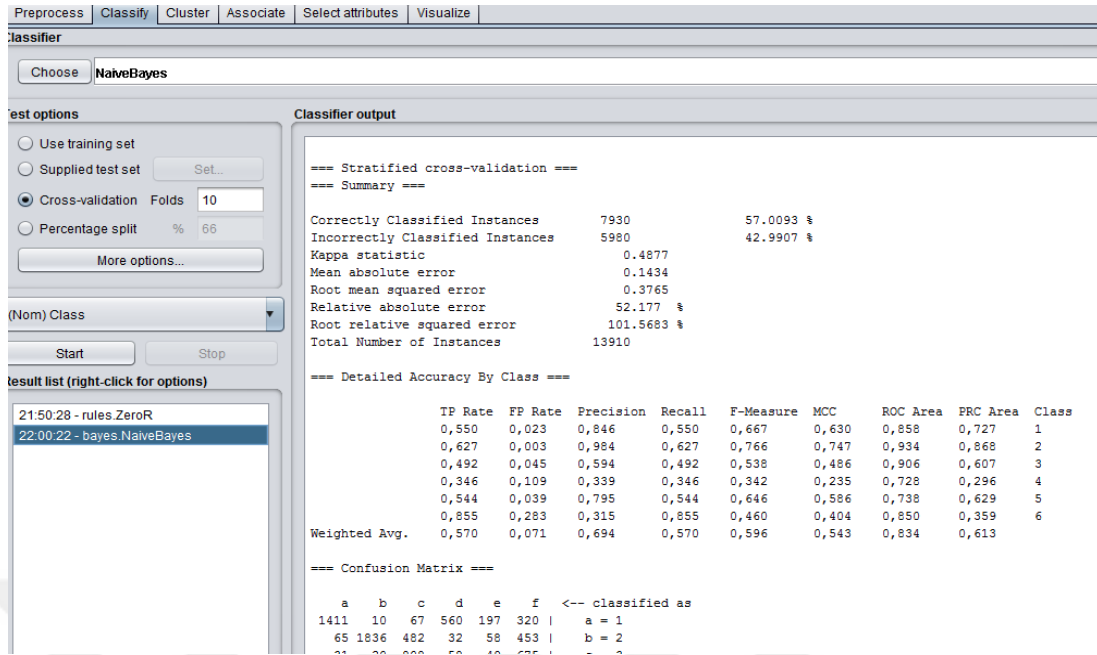
6.4.5. Naif bayes

Tahminciler arasında bağımsızlık varsayımı ile Bayes Teoremine dayanan bir sınıflandırma tekniğidir. Basit bir ifadeyle, Naive Bayes sınıflandırıcısı, bir sınıftaki belirli bir özelliğin varlığının, başka herhangi bir özelliğin varlığıyla ilgisiz olduğunu varsayar. Bu özellikler birbirine veya diğer özelliklerin varlığına bağlı olsa bile, bu özelliklerin tümü bağımsız olarak olasılığa katkıda bulunur.

Naif Bayes modelinin oluşturulması kolaydır ve özellikle çok büyük veri setleri için kullanışlıdır. Basitliğin yanı sıra, Naive Bayes'in son derece sofistike sınıflandırma yöntemlerinden bile daha iyi performans gösterdiği bilinmektedir. Naive Bayes bir sınıflandırma algoritmasıdır.

Geleneksel olarak, sayısal değerlerin bir dağıtım varsayarak desteklenmesine rağmen, giriş değerlerinin nominal olduğunu varsayar. Naive Bayes, Bayes Teoreminin (dolayısıyla naif) basit bir uygulamasını kullanır, burada her sınıf için önceki olasılık eğitim verilerinden hesaplanır ve birbirinden bağımsız olduğu varsayılır (teknik olarak koşullu olarak adlandırılır).

Bu gerçekçi olmayan bir varsayımdır çünkü değişkenlerin etkileşimini ve bağımlı olmasını bekleriz, ancak bu varsayım olasılıkları hızlı ve hesaplamayı kolaylaştırır. Bu gerçekçi olmayan varsayım altında bile, Naive Bayes'in çok etkili bir sınıflandırma algoritması olduğu gösterilmiştir.



Şekil 6.12. Gaz sensörü veri setindeki naive bayes algoritması için weka sonuçları

Naive Bayes'in % 57 oranında bir doğruluk elde ettiğini görebilirsiniz.

6.4.6. Karar ağacı: rastgele ağaç

Karar ağacı, ağaç yapısı şeklinde sınıflandırma veya regresyon modelleri oluşturur. Bir veri setini daha küçük ve daha küçük alt kümelerle bölerken aynı zamanda ilişkili bir karar ağacı kademeli olarak geliştirilir. Nihai sonuç karar düğümleri ve yaprak düğümleri olan bir ağaçtır. Bir karar düğümü iki veya daha fazla dalı vardır ve bir yaprak düğümü bir sınıflandırmayı veya kararı temsil eder.

Bir ağaçta kök düğüm adı verilen en iyi yordayıcıya karşılık gelen en üst karar düğümü. Karar ağaçları hem kategorik hem de sayısal verileri işleyebilir. Bir veri örneğini değerlendirmek, ağacın kökünden başlamak ve kasabayı yapraklara (köklere) bir tahmin yapılar kadar hareket ettirmek için bir ağaç oluşturarak çalışırlar. Bir karar ağacı oluşturma süreci, tahminler yapmak ve ağaç sabit bir derinliğe ulaşmaya kadar işlemi tekrarlamak için en iyi ayırık noktayı seçerek iştahla çalışır.

Ağaç oluşturulduktan sonra, modelin yeni verilere genelleme yeteneğini geliştirmek için budandır. Weka'da bu Sınıf, her düğümde K rastgele seçilmiş öznitelikleri dikkate alan bir ağaç oluşturmak için kullanılır.

```

=== Summary ===
Correctly Classified Instances      13506      97.0956 %
Incorrectly Classified Instances     404        2.9044 %

```

Şekil 6.13. Gaz sensörü veri setindeki Rastgele ağaç algoritması için Weka Sonuçları

Rastgele ağaç algoritması% 97,09'luk bir doğruluk sağlar ve bu da iyi bir sonuçtur.

6.4.7. Destek vektör makinesi

Destek vektör makinesi algoritması% 97,11 'luk bir doğruluk sağlar ve bu da iyi bir sonuçtur.

```

Correctly Classified Instances      13509      97.1172 %
Incorrectly Classified Instances     401        2.8828 %

```

Şekil 6.14. Gaz sensörü veri setindeki Destek vektör makine algoritması için Weka Sonuçları

6.5. Sonuç Tablosu

Tablo 6.2. Sonuç tablosu

Algoritma	Modeli oluşturma süresi / saniye	Doğru Sınıflandırılmış Örnekler	Yanlış Sınıflandırılmış Örnekler
ZeroR	0.01	21.6319 %	78.3681 %
Naif bayanlar	1.52	57.0093 %	42.9907 %
Karar ağacı Rastgele ağaç	1.66	97.0956 %	2.9044 %
Lojistik regresyon	2813.15	98.2872 %	1.7128 %
Destek vektör makinesi	18.49	97.1172	2.8828
Rastgele Orman	50.78	99.4536 %	0.5464 %
K-en yakın komşu	0.07	99.4896 %	0.5104 %

Bu tablodan, en yakın K komşusunun % 99 doğruluk elde ettiğinde bize en iyi sonucu verdiğini görebiliriz. Veri setimiz için sınıflandırma yapmak için en iyi algoritmadır.

BÖLÜM 7. TARTIŞMA VE SONUÇ

Bu çalışmada, Endüstri 4.0'ın temel konularından olan gömülü sistem yazılım ve donanımlarını kullanarak elde edilen sensör verilerini makine öğrenmesi algoritmaları ile sınıflandırmak ve sürdürülebilir üretim için gerekli otomasyonun sağlanması için alt yapı oluşturmaktır. Gaz sensörleri, insan sağlığını ve özelliklerini tehdit eden yanıcı, yakıcı ve toksik gazları tespit etmek için endüstride ve yangınla mücadelede yaygın olarak kullanılmaktadır. Önce arduino ile bir gaz verilerini toplama ve makine öğrenmesi ile sınıflandırılması üzerine çalışılmıştır.

Gazlar ile ilgili yeterince veri elde edilemediği için altı farklı uçucu organik gaz için bir veri kümesi veri seti indirilerek onun üzerinde çalışmalar yapılmıştır. Bu veri setinde; Amonyak, Asetaldehit, Aseton, Etilen, Etanol ve Toluen gazları bulunmaktadır. Üç yıllık bir süre boyunca, 16 metal oksit gaz sensörü kullanılarak elde edilmiştir. Bu veriler 13910 ölçüm ve 129 özellik içerir. Çalışmamızda bu verileri sınıflandırmak için çok sayıda makine öğrenmesi algoritmaları kullanılmıştır. Bunlar; k-en yakın komşular, Destek vektör makinesi, Rastgele orman ve Lojistik regresyon dur. Verileri sınıflandırmak için bu makine öğrenme algoritmaları Weka programında gerçekleştirilmiştir.

Her bir makine öğrenmesi algoritması bize genel itibari ile başarılı sonuçlar vermiştir. Ancak bu çalışmalar sonucunda KNN, Rastgele orman ve lojistik regresyon algoritmaları en iyi sonucu veren üç başarılı algoritma olarak ortaya çıkmıştır.

En başarılı sonuç veren KNN verilerimizi 99,48 doğruluk ile sınıflandırılırken, ikinci en başarılı sonuç veren algoritma Rastgele ormandır ve başarı yüzdesi % 99,45dir.

Üçüncü başarılı sonuç ise lojistik regresyon algoritması tarafından verilmiştir. Bu başarı yüzdesi % 98,28dur. Gaz sensörlerinin ve gazların sınıflandırılmasında makine öğrenmesi algortimaları başarılı sonuçlar vermişler ve sınıflandırmada kullanılmaları önerilir.



KAYNAKLAR

- [1] Rettig, F. and Moos, R., 2013. Semiconducting direct thermoelectric gas sensors. In *Semiconductor Gas Sensors* (pp. 261-296). Woodhead Publishing.
- [2] Lin, C., Xu, W., Yao, Q. and Wang, X., 2019. Nanotechnology on Toxic Gas Detection and Treatment. In *Novel Nanomaterials for Biomedical, Environmental and Energy Applications* (pp. 275-297). Elsevier.
- [3] Rajkumar, K. and Kumar, R.R., 2019. Gas Sensors Based on Two-Dimensional Materials and Its Mechanisms. *Fundamentals and Sensing Applications of 2D Materials*.
- [4] Vergara, A., Vembu, S., Ayhan, T., Ryan, M.A., Homer, M.L. and Huerta, R., 2012. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166, pp.320-329.
- [5] Schütze, A., Helwig, N. and Schneider, T., 2018. Sensors 4.0—smart sensors and measurement technology enable Industry 4.0. *Journal of Sensors and Sensor Systems*, 7(1), pp.359-371.
- [6] Endüstri 4.0, www.bcg.com/en-tr/capabilities/operations/embracing-industry-4.0-rediscovering-growth., Eriřim Tarihi: 20.01.2020.
- [7] Datase, www.openml.org/d/1476., Eriřim Tarihi: 20.11.2019.
- [8] Dataset, archive.ics.uci.edu/ml/datasets/Gas+Sensor+Array+Drift+Dataset#., Eriřim Tarihi: 5.01.2020.
- [9] Endüstri 4.0'ın Faydaları, www.slcontrols.com/benefits-of-industry-4-0., Eriřim Tarihi: 22.01.2020.
- [10] Endüstride yapay zeka, www.hightechsummit.dk/artificial-intelligence-in-the-industry., Eriřim Tarihi: 5.01.2020.
- [11] Endüstriyel Yapay Zekanın Yükseliři, www.cio.com/article/3400951/the-rise-of-industrial-ai.html., Eriřim Tarihi: 18.12.2019.

- [12] Nufar Gaspar, Intel white paper, “Artificial Intelligence Reduces Costs and Accelerates Time to Market,” June 2018.
- [13] Yapay zeka, www.deltechnologies.com/Dell EMC video, OTTO Motors uses AI for a happier workforce., Erişim Tarihi: 25.10.2019.
- [14] McKinsey and Company, “Smartening up with Artificial Intelligence (AI) — What’s in it for Germany and its Industrial Sector?” April 2017.
- [15] CIO.com blog by Dell EMC, “Seeding the Right IoT Architecture to Build a Better Farm,” March 21, 2019. Dell Technologies case study, “Harvesting Insights,” 2018.
- [16] Machine-learning-in-industry-4-0, www.datafestival.de/en/machine-learning-in-industry-4-0-five-use-cases ., Erişim Tarihi: 18.12.2019.
- [17] Sensörler, [www.elprocus.com/types-of-sensors-with-circuits.](http://www.elprocus.com/types-of-sensors-with-circuits), Erişim Tarihi: 02.09.2019;
- [18] Gaz sensörleri , [www.techniques-ingenieur.fr/base-documentaire/mesures-analyses-th1/metrologie-relative-aux-gaz-42539210/capteurs-de-gaz-a-semi-conducteurs-r2385.](http://www.techniques-ingenieur.fr/base-documentaire/mesures-analyses-th1/metrologie-relative-aux-gaz-42539210/capteurs-de-gaz-a-semi-conducteurs-r2385), Erişim Tarihi:22.01.2020.
- [19] Debliquy, M., 2006. Capteurs de gaz à semi-conducteurs. Ed. Techniques Ingénieur.
- [20] Gas sensor, [www.components101.com/articles/introduction-to-gas-sensors-types-working-and-applications.](http://www.components101.com/articles/introduction-to-gas-sensors-types-working-and-applications), Erişim Tarihi:21.01.2020.
- [21] Machine learning, [www.javatpoint.com/machine-learning.](http://www.javatpoint.com/machine-learning), Erişim Tarihi:15.11.2019.
- [22] Random..forest,[www.en.wikipedia.org/wiki/Random_forest.](http://www.en.wikipedia.org/wiki/Random_forest),Erişim Tarihi:08.12.2019.
- [23] Random-forest-algorithm,www.syncedreview.com/2017/10/24/how-random-forest-algorithm-works-in-machine-learning.,Erişim Tarihi: 19.09.2019.
- [24] Random forest, www.dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learning.,Erişim Tarihi: 19.09.2019.
- [25] Random forest, [www.medium.com/greyatom/a-trip-to-random-forest.](http://www.medium.com/greyatom/a-trip-to-random-forest),Erişim Tarihi: 19.09.2019.

- [26] Machine learning, www.mrmint.fr/9-algorithmes-de-machine-learning-que-chaque-data-scientist-doit-connaître.,Erişim Tarihi: 21.09.2019.
- [27] k_nearest_neighbors, www.saedsayad.com/k_nearest_neighbors.,Erişim Tarihi: 21.09.2019.
- [28] k-nearest-neighbors..algorithm, www.blog.usejournal.com/a-quick-introduction-to-k-nearest-neighbors-algorithm.,Erişim Tarihi: 21.09.2019.
- [29] k-nearest neighbors,www.towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm.,Erişim Tarihi: 21.09.2019.
- [30] He, Q.P. and Wang, J., 2007. Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes. IEEE transactions on semiconductor manufacturing, 20(4), pp.345-354.
- [31] Li, Z. and Tian, X., 2008. Soft sensor modeling method based on secondary variables KNN analysis. JOURNAL OF CHEMICAL INDUSTRY AND ENGINEERING-CHINA-, 59(4), p.941.
- [32] Yang, J., Sun, Z. and Chen, Y., 2016. Fault detection using the clustering-kNN rule for gas sensor arrays. Sensors, 16(12), p.2069.
- [33] www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code., Erişim Tarihi:17.12.2019.
- [34] Support-vector-machine, www.javatpoint.com/machine-learning-support-vector-machine-algorithm., Erişim Tarihi:17.12.2019.
- [35] Huang, S., Cai, N., Pacheco, P.P., Narrandes, S., Wang, Y. and Xu, W., 2018. Applications of support vector machine (SVM) learning in cancer genomics. Cancer Genomics-Proteomics, 15(1), pp.41-51.
- [36] Pai, P.F. and Lin, C.S., 2005. Using support vector machines to forecast the production values of the machinery industry in Taiwan. The International Journal of Advanced Manufacturing Technology, 27(1-2), p.205.
- [37] Support vector machine, www.medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72.,Erişim Tarihi:22.01.2020.
- [38] Logistic regression, www.javatpoint.com/logistic-regression-in-machine-learning.,Erişim Tarihi:22.01.2020.
- [39] Arduino DUE, www.store.arduino.cc/arduino-due.,Erişim Tarihi: 15/03/2020.

- [40] Arduino DUE , www.robotiksistem.com/arduino_due_ozellikleri.html.,Erişim Tarihi: 13/03/2020.
- [41] Arduino DUE, www.diyot.net/arduino-due/.,Erişim Tarihi:10/03/2020.
- [42] Sensor,www.thomasnet.com/catalogs/carbon-dioxide-gas-sensors/.,Erisim Tarihi:15/03/2020.
- [43] Sensor, [www.figarosensor.com/technical information for CDM4161A](http://www.figarosensor.com/technical%20information%20for%20CDM4161A).,Erişim Tarihi:15/03/2020.
- [44] Tera term, www.wikipedia.org/wiki/Tera_Term.,Erişim Tarihi:16/03/2020.
- [45] Logistic regression, www.kambria.io/blog/logistic-regression-for-machine-learning.,Erişim Tarihi:22.01.2020.
- [46] Logistic regression,www.mrmint.fr/logistic-regression-machine-learning-introduction-simple.,Erişim Tarihi:22.01.2020.
- [47] Weka, www.cs.waikato.ac.nz/ml/weka/arff.html.,Erişim Tarihi:22.01.2020.

ÖZGEÇMİŞ

Safa El bekri, 26.08.1991 tarihinde Tunus'ta doğdum. İlk ve ikinci okulumu Tunus'ta bitirdim. Bachlor'umu 2011 yılında yüksek Tunus yönetimi enstitüsünde başlattım. Branch IT 2013 yönetiminden mezun oldum. Daha sonra Yüksek Lisans programına Karar Destek için Akıllı Yöntemler İş Zekası dalında başladım ve 2015 yılında mezun oldum. 2015 yılında Türkiye'ye geldim Türkçe öğrenmeye başladım, daha sonra 2017 yılında Sakarya Üniversitesi'nde yüksek lisansım başladı.