T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

FARKLI MAKİNE ÖĞRENMESİ ALGORİTMALARININ KARŞILAŞTIRILMASI

Ebru PEKEL

YÜKSEK LİSANS TEZİ

TC

ONDOKUZ MAYIS ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

FARKLI MAKİNE ÖĞRENMESİ ALGORİTMALARININ KARŞILAŞTIRILMASI

EBRU PEKEL

AKILLI SİSTEMLER MÜHENDİSLİĞİ ANABİLİM DALI

SAMSUN

2018

TEZ ONAYI

Ebru PEKEL tarafından hazırlanan "FARKLI MAKİNE ÖĞRENMESİ ALGORİTMALARININ KARŞILAŞTIRILMASI" adlı tez çalışması 11/01/2017 tarihinde aşağıdaki jüri tarafından Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü Akıllı Sistemler Mühendisliği Anabilim Dalı'nda Yüksek Lisans Tezi olarak kabul edilmiştir.

Danışman Prof. Dr. Sermin Elevli
Akıllı Sistemler Mühendisliği Anabilim Dalı

İkinci Danışman Yrd. Doç. Dr. Tuncay Özcan (Eş Danışman)
İstanbul Üniversitesi
Endüstri Mühendisliği Anabilim Dalı

Jüri Üyeleri

Başkan Prof. Dr. Sermin ELEVLİ

Ondokuz Mayıs Üniversitesi

Endüstri Mühendisliği Anabilim Dalı

Üye Yrd. Doç. Dr.Naci Murat

Ondokuz Mayıs Üniversitesi

Endüstri Mühendisliği Anabilim Dalı

Üye Yrd. Doç. Dr. İhsan EROZAN

Dumlupınar Üniversitesi

Endüstri Mühendisliği Anabilim Dalı

Yukarıdaki sonucu onaylarım. 11/01/2018

Prof. Dr. Bahtiyar ÖZTÜRK

Enstitü Müdürü

ETİK BEYAN

Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

11/01/2018

Ebru PEKEL

ÖZET

Yüksek Lisans Tezi

FARKLI MAKİNE ÖĞRENMESİ ALGORİTMALARININ KARŞILAŞTIRILMASI

Ebru PEKEL

Ondokuz Mayıs Üniversitesi

Fen Bilimleri Enstitüsü

Akıllı Sistemler Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Sermin Elevli

Eş Danışman: Yrd. Doç. Dr. Tuncay Özcan

Makine Öğrenmesi, matematiksel ve istatistiksel yöntemler kullanarak mevcut verilerden çıkarımlar yapan, bu çıkarımlarla bilinmeyene dair tahminlerde bulunan bir veri madenciliği yöntemidir. Probleme yaklaşımlarına göre farklılık gösteren (sınıflandırma, tahmin, kümeleme) ve bu yüzden farklı problemlerde farklı başarılara sahip olan birçok makine öğrenmesi yöntemi bulunmaktadır. Geçmiş verilerin hangi sınıftan olduğu biliniyorsa, yeni gelen verinin hangi sınıfa ait olacağı makine öğrenmesi algoritmaları kullanılarak tespit edilebilmektedir. Bu tez çalışmasında bir sınıflandırma problemi üzerinde durulmuştur. Çalışmada, dört temel sınıflandırma algoritması (Karar Ağacı, Destek Vektörü Makineleri, Naive Bayes, Yapay Sinir Ağları) sunulmuş ve hazır veri setindeki performansları karşılaştırılmıştır. Naive Bayes Algoritmasının, veri setinde uygulanan diğer sınıflandırma yöntemlerinden daha iyi performans (%70,29) gösterdiği tespit edilmiştir. Çalışmada ayrıca temel sınıflandırma algoritmalarının performansını artırmak amacıyla, genetik algoritma ile melez modelleri önerilmiş ve performansları karşılaştırılmıştır. Genetik Algoritmalı Karar Ağacı (GA-KA) Algoritmasının en yüksek performans değerine (%92,57) sahip melez model olduğu tespit edilmiştir.

Ocak 2018, 76 sayfa

Anahtar Kelimeler:Sınıflandırma, Makine Öğrenmesi, Veri Madenciliği, Melez Modeller.

ABSTRACT

Master's Thesis

COMPARISON OF DIFFERENT MACHINE LEARNING ALGORITHMS

Ebru Pekel

Ondokuz Mayis University

Graduate School of Sciences

Department of Industrial Engineering Departmant

Supervisor: Prof. Dr. Sermin Elevli

Co-Supervisor: Ass. Prof. Dr. Tuncay Özcan

Machine learning is a methods that make inferences from existing data using mathematical and statistical methods and that are inferred from these inferences. They may differ according to the probing approaches and thus may have different successes in different problems (classification, prediction, clustering). In this thesis study, a classification problem is emphasized. If the former data (information) is known in which class,it is possible to identify to which class the data is to be included by using certain machine learning algorithms. In this thesis, it is presented 4 classification algorithms (Decision Tree, Naive Bayes, Support Vector Machine and Artificial Neural Network) and compared their performance on the standard dataset which was obtained from a database. Naive Bayes Algorithm performed better performance (70.29%) than the other methods. In addition to these traditional classification algorithms, their hybrid algorithm models with Genetic Algorithm were presented and were compared. Results of the four classification algorithms were evaluated in terms of classification performance. According to the findings, Decision Tree with Genetic Algorithm (GA-DT) model performed better performance (92.57%) the other classification methods applied on dataset.

January 2018, 76 pages

Keywords: Classification, Machine Learning, Data Mining, Hybrid Models.

ÖNSÖZ VE TEŞEKKÜR

Tez çalışmam sırasında kıymetli bilgi, birikim ve tecrübeleri ile bana yol gösterici ve destek olan değerli danışman hocalarım, sayın Prof. Dr. Sermin Elevli ve ilgisini, değerli bilgilerini ve önerilerini göstermekten kaçınmayan sayın Yrd. Doç. Dr. Tuncay Özcan'a sonsuz teşekkür ve saygılarımı sunarım.

Çalışmalarım boyunca yardımını hiç esirgemeyen değerli arkadaşım Zeynep Ceylan ve abim Engin Pekel'e teşekkürü bir borç bilirim.

Çalışmalarım boyunca maddi manevi destekleriyle beni hiçbir zaman yalnız bırakmayan aileme de sonsuz teşekkürler ederim.

Ocak 2018, Samsun

Ebru PEKEL

İÇİNDEKİLER DİZİNİ

KISALTMALAR

GA: Genetik Algoritma

NB: Naive Bayes

DVM: Destek Vektörü Makineleri

YSA: Yapay Sinir Ağları

KA: Karar Ağaçları

GA-KA: Genetik Algoritmalı Karar Ağacı

GA-NB: Genetik Algoritmalı Naive Bayes

GA-DVM: Genetik Algoritmalı Destek Vektörü Makineleri

GA-YSA: Genetik Algoritmalı Yapay Sinir Ağları

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

Günümüzde şirketler, bilişim ve veri depolama sistemlerine düşük maliyetlerde sahip olabilmekte ve bu sayede internet ağları üzerinden hızlı bir şekilde bilgiler yayılabilmektedir. Bilgisayar sistemlerinin kullanımının bu denli hızla yaygınlaşmasıyla birlikte sayısal veri üretiminin artmış ve veri depolama teknolojilerinin gittikçe güçlenmesi nedeni ile de veri tabanlarında daha fazla veri depolanmaya başlanmıştır. Daha fazla verinin ağlarda toplanması ile oluşan kirlilikten dolayı bu veri yığınının yönetilmesi, bu verilerin anlamlı hale getirilmesi ve işe yarar bilgilerin çıkarılması konusunda ciddi boyutta sorun oluşmaya başlamıştır. Çünkü veri tabanlarının bilgiyi sadece saklamak için dizayn edilmesinden dolayıbilişim yöntemleri ile üretilen bu veriler tek başlarına anlamsızdır. Bu veriler belli bir örüntüye uydurulup işlendiği zaman anlamlı hale gelmektedir.

Verilerin analiz edilip yönetilmesi Veri Madenciliği alanının oluşmasını sağlamıştır. Örneğin eskiden hastanelerde süreç hastanın sıraya girerek muayene olmasından ibaretti. Hastanın muayene sırasında tıbbi verileri değerlendirilir, fiziksel olarak dosyalanır ve sonuca varılırdı. Günümüzde ise böyle bir muayene işleminden ziyade sağlık sektörünün veri tabanı sayesinde bu hareketin bütün detayları saklanabilmektedir. Bu binlerce hasta ve onların kan değerleri vatandaşlık numaraları ile kodlanmış bir şekilde saklanabilmektedir. Bu sayede de bir hastanın zaman içindeki verilerine ulaşmak ve analiz etmek olasıdır.

Veri tek başına değersizdir. Bu nedenle bir nesnenin geçmiş verilerini ve güncel verilerini kapsayandesenin tespiti üzerinden yorumlama yapılmalıdır. Veriler genellikle tanımlanmamış kullanım ve sorguları içeren ham gerçekleri sunarlar. Bilgi, seçenekleri etkileyen işlenmiş veri olmak üzere elde edilmelidir. Verileringüncellenme, filtrelenme ve özetlenmesi ile karar verme amacına dönük bilgi üretimi gerçekleşir. Veriyi bilgiye çevirme işlemine veri analizi denir. Veri analizi yaparak bir bireyin sonraki yapacağı alışveriş tahminleri çıkarılabilir, müşteriler alışveriş alışkanlıklarına bağlı olarak gruplanabilir, yeni bir ürün için potansiyel müşteriler belirlenebilir; müşterilerin zaman içindeki hareketleri incelenerek onlara tamamlayıcı hizmetlerin neler olabileceği tahmin edilebilir. Binlerce müşterinin olabileceği düşünülürse böylesi analizlerin elle yapılamayacağı, teknolojik bir yapı ile yapılmasının gerektiği ortaya çıkar. Makine öğrenimi konusu burada devreye girer.

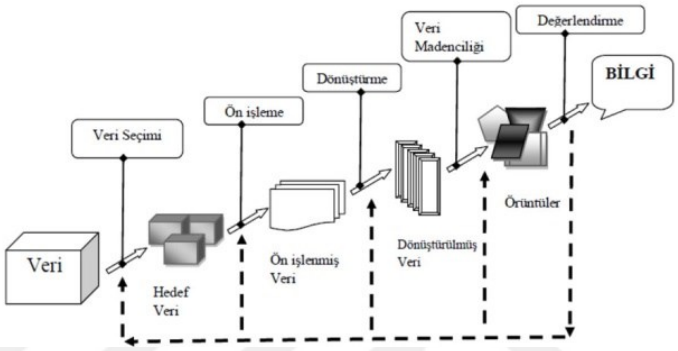
Makine öğrenmesi veri madenciliği alanında kullanılan, büyük miktarda veri içinden gelecekle ilgili tahmin yapılmasını sağlayacak örüntü ve kuralları inceleyen akademik bir disiplindir. Veri madenciliğinde kullanılan algoritmaların çoğunluğumakine öğrenmesi çalışmaları sonucu oluşmuştur.

Makine öğreniminde, geçmiş bilgilerin hangi sınıftan olduğu biliniyorsa, yeni gelen verinin hangi sınıfa dahil olacağı belirli bir takım makine öğrenmesi algoritmaları kullanılarak tespit edebilmek mümkündür. Aynı zamanda temel makine öğrenmesi algoritmalarının çeşitli optimizasyon teknikleriyle melezleyerek performans değerleri daha yukarıya çıkarmak mümkün olabilmektedir. Bu tez çalışmasında, temel sınıflama algoritmaları ile bunların melez modelleri sunulmuş ve diyabet teşhisine dönük hazır veri seti üzerinden elde edilen performansları karşılaştırılmıştır.

2. VERİ MADENCİLİĞİ

2.1. Veri Madenciliğinin Tanımı

Büyük miktardaki veriler arasından gerekli olanları çıkarmaya Veri Madenciliği denir. Veriler üzerinde analizler yapmak ve veriyi analiz edip bilgiyi çıkarabilmek amacıyla veri madenciliği teknikleri ortaya çıkmıştır (Şekil 2.1). Veri madenciliği uygulamaları, bu açıdan bakıldığında sadece bir sorgulama süreci veya istatistik programlarıyla yapılmış bir çalışma değildir.



Şekil 2.1. Veri madenciliği

Veri madenciliği, milyonlarca veri ve çok fazla değişken değerleri ile ilgilenir. Zamanla verilerin daha karmaşık yapıya bürünmesi bir takım temel kavramları da beraberinde getirmiştir. Veri analizleri ile ilgili temel kavramlar aşağıda açıklanmaktadır.

- Veri: Bir araştırmada, bir tartışmada, bir akıl yürütmede sonuca ulaşabilmek için gereken ilk bilgidir.
- Veri Ambarı: Birçok veri tabanından alınarak derlenen verilerin toplandığı depolardır.
- Veri Tabanı: Birbirleriyle ilişkili bilgilerin depolandığı alanlardır.
- Veri Madencisi: Hedefi oluşturma, veriyi elde etme, veriyi hazırlama, modeli uygulama, sonuçları değerlendirme gibi önemli alanları yöneten kişidir.

Veriyi yönetmek için "veri ambarı", verileri çözümleyip bilgiye ulaşılabilmesi için "veri madenciliği" yöntemleri ortaya çıkmıştır.

Bilimsel gelişmeler dünyada gerçekleşen birçok işlemin elektronik olarak veri ambarına kayıtlanmasını, bu kayıtların gerekli şekilde saklanabilmesini ve gerektiğinde

ulaşılabilmesini hem kolaylaştırmakta, hem de bu süreçlerin her geçen gün daha az maliyet gereksinimi sağlamaktadır. Ancak, veri ambarlarında saklanan birçok veriden analiz için anlamlı çıkarımlar sunabilmek bu verilerin yetkin uzmanlarca analiz edilmesini gerektirmektedir.

Veri madenciliği, uygulamalarında veri saklamanın öneminden dolayı alt yapı gereksinimi *veri ambarı* sayesinde sağlanmaktadır. 1991 yılında ilk olarak William H. Inmon tarafından ileri sürülen veri ambarı kavramı, örneğin bir işletmede yönetimin kararlarını desteklemek amacı ile bazı kaynaklardan çıkardıkları bilgileri zaman değişkeninihesaba katarak veri toplama olarak tanımlanmaktadır. Veri ambarlarının özelliği veri madencilerine farklı detay seviyelerinde bilgi sağlayabilmesidir. Detayın taban düzeyinde arşivlenen kayıtların kendisi ile ilgili bilgiler içerirken, daha üst düzeyler zaman değişkeni gibi daha fazla bilginin derlenmesi ile ilgilidir. Veri ambarları, mali olarak da ciddi yatırımlar gerektirmekte ve uygulanması bir yıl veya daha fazla zaman alabilmektedir (Mackinnon ve Glick, 1999).

Veri ambarları, verilerin üzerine bilgi eklemeye ve verilerde değişiklikler yapabilmek için değil sadece okumaya yönelik olarak veri sunabilmektedir. Bu sebeple, veri ambarında veriler, uygulamaları kolaylaştıran bir formatta tutulmaktadır. Burada uygulama; raporlar, karar destek sistemleri,sorgular veya istatistikselişlemleri kapsamaktadır. Birbiriyle entegre olmayan uygulamaların entegre edilmesine olanak sağlar. Veri Ambarları, tıp sektöründen bilişim sektörüne, işletmelerin pazarlama bölümünden üretim bölümlerine, geleceğe dönük analizleryapmada, sonuç çıkarımında ve işletmelerin yönetim stratejilerini değerlendirmede kullanılmakta olan bir sistemdir.

Veriyi yönetmek için “veri ambarı”, verileri çözümleyip bilgiye ulaşılabilmesi için “veri madenciliği” yöntemleri ortaya çıkmıştır.

2.2. Veri Madenciliğinin Gelişim Süreci

Bilgisayarların etkin kullanımında ilk adım verilerin depolanmasıdır.Önceleri karmaşık işlemleri yapmaya yönelik geliştirilen bilgisayarlar, kullanıcı istekleri doğrultusunda veri depolama işlemleri için de kullanılmaya başlanmıştır. Bu depolama işlemleri paralelinde veri tabanları ortaya çıkmıştır. Veri tabanlarının genişlemesi donanımsal olarak bu verilerin tutulacakları ortamların da genişlemesini gerektirdi. Veri ambarı kavramının ortaya çıkışı bu dönemlere denk gelmektedir. Kaybedilmek istenmeyen veriler, bir ambar misali fiziksel sürücülerde tekrar kullanılmak üzere saklanmaktaydı.

Gittikçe büyüyen veri tabanlarının güncellenmesi, düzenlenmesi ve yönetimi de buna bağlı olarak zor bir hal almaya başladığında, veri modelleme kavramı ortaya çıkmıştır.İlk etapta basit veri modelleri olan Hiyerarşik ve Şebeke veri modelleri geliştirildi.

Günümüz teknolojisi trilyonlarca bit veriyi küçük çip belleklerde tutmak mümkün hale getirmiştir. Veriyi depolama ihtiyacı her ne kadar teknolojiyi ciddi anlamda ileri götürse de yanında bir takım sorunları da getirmektedir. Her ne kadar verilerin depolanması, düzenlenmesi, yönetilmesi bir problem gibi görünmese de bu kadar çok veri ile sadece istenilen bilgiye ulaşmak başlı başına bir sorun haline gelmiştir. Veri madenciliği, kavramsal olarak 1960'lı yıllarda, bilgisayarların veri analiz problemlerini çözmek için kullanılmaya başlamasıyla ortaya çıkmıştır. O dönemlerde yeterli bir tarama yapıldığında, istenilen bilgilere bilgisayar yardımıyla ulaşmanın mümkün olacağı düşünülmeye başlanmıştı. Ancak o zamanlarda bu işlemler bütününe veri madenciliği yerine veri taraması (data dredging) ve veri yakalanması (data fishing) gibi isimler verilmiştir. Veri madenciliği ismi ilk olarak 1990'lı yıllara gelindiğinde bilgisayar mühendisleri tarafından ortaya atılmıştır. Burada bilgisayar mühendislerinin amacı, geleneksel istatistiksel yöntemler yerine, veri analizinin algoritmik bilgisayar modülleri tarafından analizinin mümkün olduğunu vurgulamaktır (Çetin, 2009).

Bu aşamadan sonra veri madencileri, veri madenciliğine farklı yaklaşımlar getirmeye başlamışlardır. Bu yaklaşımların kökeninde istatistik, makine öğrenimi, veritabanları, otomasyon, pazarlama, araştırma gibi disiplinler ve kavramlar yatmaktadır.

Bilgisayarların veri analizi için kullanılmaya başlamasıyla istatistiksel çalışmalar da yaygınlaşmaya ve çeşitlenmeye başlamıştır. Veri madenciliğinin varlığı ile daha önce yapılması olası olmayan istatistiksel araştırmalar yapılabilmesi mümkün hale gelmiştir. 1990'lardan sonra istatistik, veri madenciliği ile ortak bir platforma taşınmıştır. Verinin, yığınlar içerisinden çekip çıkarılması ve analizinin yapılarak kullanıma hazırlanması sürecinde veri madenciliği ve istatistik birlikte kullanılmaktadır.

Bunun yanı sıra veri madenciliği, veri tabanları ve makine öğrenimi disipliniyle birlikte yol almıştır.

2.4. Veri Madenciliğinin Uygulama Alanları

Veri madenciliği, veri işlemedurumlarında;sınıflandırma,sınama, güncelleme, özetleme, kaydetme, istatistiksel veya mantıksal hesaplama, depolama, üretme, erişim ve iletme adımlarını içeren süreç gibi işlemektedir, ayrıca bilgiyi keşfetme adımının bir parçasını oluşturur. Veri madenciliği, bilişsel tekniklerle edinilen bilgilerin dağıtılması ve yorumlanması amacıyla; önişleme, seçme, yorumlama, dönüştürme adımlarının ardından veri madencisi ve veri tabanlarıyla etkileşimi destekleyen bir adımdır. Bu adımdan sonra, veri seti üzerindeki örüntüler gözden geçirilerek yorumlama adımına geçilebilmektedir.

Son yıllarda otomatik veri derleme ve analizi yöntemlerinin kolaylaşmasıyla, edinilen veri miktarı ve tipleri de çoğalmıştır. Veri derleme yöntemleri ve veri madenciliği teknolojisindeki gelişmeler, veri depolarındaki çok sayıda bilginin saklanmasını ve analizini gerektirmektedir. Bu denli çok miktardaki verinin de geleneksel yöntemlerle işlenebilmesi mümkün değildir.

Veri madencisi içinalınan kararın tutarlılığı, onun teknik ve bilgi birikimine olduğu gibi karar aşamasında kullandığı verilerin yeterliliğine de bağlıdır. Kararın doğruluğu ve tutarlığında, verilerin doğru saklanması, doğru analiz edilmesi, doğru seçilmesi, doğru yorumlanması oldukça önemlidir.

Teknolojik gelişmeler ile birlikte veri hacimleri de giderek artmış ve bu durum veri depolarının büyüklüğünü artırmış ve geleneksel yöntemlerle işlenemeyecek ölçülere ulaştırmıştır.

Karar vericilerin yanlış kararlar alma riskini azaltabilmek için, mümkün olabildiğince çok sayıda veri toplanması gerekmektedir. Ayrıca uygulamalarının yaygınlaşması, rekabet koşulların zorlaşması, kar marjlarının değişkenliği ve teknolojik gelişmelerin etkisiyle artan esnek müşterileri taleplerinin tam olarak karşılanabilmesi içindoğru karar vermek analiz sürecinde en önemli adımı oluşturmaktadır. Bu nedenle veri madenciliği bilişim, finans, sağlık, müşteri ilişkileri yönetimi başta olmak üzere birçok alanda yaygın olarak kullanılmaya başlanmıştır (Abu Nimeh, 2007; Amasyalı, 2008).

Doğru ve gerekli veriye ulaşmak, verinin nitelikleri kadar hayati bir öneme sahiptir.

Veri madenciliği, karar vermede önemli bilgilerin ortaya çıkarılmasını sağlayan stratejik bir araçtır.

Karar verme dışında veri madenciliği bankacılık, pazarlama, finans, sağlık gibi farklı disiplin çalışmalarında da kullanılmaktadır.

Pazarlama uygulamalarında,

- Müşteri profilinin daha iyi tanınması,
- Müşterilerin doğru segmentlere ayrılması,
- Müşterilere ait davranış örüntüleri çıkarılması,

Bankacılık ve finans uygulamalarında,

- Kredi kartı kullanımlarına göre müşterleri belirli gruplara ayırmak,
- Kredi ve hesap açtırma taleplerinin uygunluğunun değerlendirilmesi,
- Kredi talep edecek müşterilerin öncedenöngörülmesi,
- Müşterilerin geçmiş dönemlerdeki ödeme performanslarının analiz edilerek, aynıperformansa sahip diğer müşteriler için risk haritasının oluşturulması şeklindeörneklendirilebilir (Davis, 1992).

Bankalar tarafından düzenlenecek çeşitli kredi kampanyaları için mevcut müşteri segmentinin seçimi ve müşterilerin ödeme performanslarına yönelik kredi paketleri hazırlanmasına yardımcı olur.

Sağlık uygulamalarında,

- Hastalar için risk haritalarının oluşturulması,
- Hastalık ve anormalilik tespiti,
- Hastalıkların geleceğe dönük yayılımlarının analizleri gibi problemlerde veri madenciliği kullanılabilmektedir (Boyle, 2001; Çakmak, 2017; Chao vd, 2010).

2.5. Makine Öğreniminde Bilgi Keşfi Süreci

Günümüz teknolojisinin temelini veri, bilgi ve enformasyon kavramları oluşturmaktadır, sistemlere ilişkin stratejik kararların temelini veri ve veriden derlenen bilgi oluştururken alınan kararların amacına uygun ve efektif olabilmesi için güvenilir, güncel, net ve zamanında ulaşan veriye ihtiyaç duyulmaktadır. Bu kavramlar hayatımızın her evresinde karşımıza çıkmakta ve sağlık verileri, alışveriş verileri, otomasyon verileri, oldukça hızlı bir şekilde artarken oluşan veritabanlarını uygun bir şekilde saklayabilmek de işletmeler açısından büyük bir zorunluluk haline gelmektedir. Bu verilerin depolanması ve saklanması ambarlardaki veriler büyüdükçe bir takım

problemler de meydana gelmiştir ve araştırmacılar bu problemlere veri tabanları ve dosya sistemlerindeki gelişmelerle çözüm bulma yoluna gitmişlerdir.

Bu büyüklükteki veri tabanlarını herhangi bir ara yüz kullanmadan analize tabi tutabilmesi ve bu verilerin karar destek aşamasında kullanımı imkansız olmaktadır. Bu aşamada veritabanından bilgi keşfi kavramı ortaya çıkmıştır. Veritabanından bilgi keşfi; veriden yararlı bilgi çıkarma sürecidir.

Veri tabanlarında bilgi keşfinin uygulanabilmesi için verilerin veri ambarı gibi bir veri tabanı düzeninde olması gerekmektedir. Veri ambarları işlemsel olduğu kadar işlemsel olmayan verileri de içermektedir. Bu yapılar veritabanında bilgi keşfinin altyapısını oluşturmaktadır. Veri ambarının düzenli olması, makine öğrenimi uygulamalarını oldukça kolaylaştıracak bir etkendir (Özdemir vd, 2009).

2.5.1. Problemin tanımlanması

Makine öğrenimi çalışmalarında sağlıklı bir neticeye ulaşabilmek için, uygulamanın hangi amaç için yapılacağının açık bir şekilde tanımlanması gerekmektedir. Belirlenen amaç, problem üzerine odaklanmış ve açık bir dille ifade edilmiş olmalı, elde edilecek sonuçların başarı düzeylerinin nasıl ölçüleceği diğer bir deyişle performans göstergeleri de net bir şekilde tanımlanmalıdır.

2.5.2. Verilerin hazırlanması

Model kurulması sürecinde ortaya çıkabilecek problemler, bu adıma tekrar geri dönülerek verilerin düzenlenme sürecine girmesine neden olmakta ve zaman kaybına sebebiyet vermektedir. Verilerin hazırlanması ve modelin kurulması aşamaları toplam, veri keşfi sürecindeefor ve zamanının %50 - %85'ine karşılık gelmektedir. Verilerin hazırlanması aşaması kendi içerisinde toplama, değer biçme, birleştirme ve temizleme, seçme ve dönüştürme adımlarından meydana gelmektedir.

2.5.2.1. Toplama

Tanımlanan problem için gerekli olduğu düşünülen verilerin ve bu verilerin toplanacağı veri kaynaklarının belirlenmesi adımıdır. Verilerin toplanmasında çalışmanın kendi veri kaynaklarının dışında, farklı analiz çalışmalarının veya veri pazarlayan kuruluşların veri tabanlarından faydalanılabilir. Günümüz teknolojisinde veriler birçok farklı ortamda depolanabilmektedir. Bu aşamadaki işlemi özü, veri tabanlarından veya veri ambarlarından yapılacak uygulama için uygun verileri çekebilmektir. Veri madenciliği

uygulamalarında, veriler test ve eğitim veri seti olarak iki gruba ayrılmalıdır. Genellikle yapılan uygulamalarda verilerin %80'i analiz %20'si ise test verisi olarak ayrılmakla beraber her çalışmanın yapısına uygun olarak bu oranlar büyük ölçüde değişebilmektedir.

2.5.2.2. Değer biçme

Veri madenciliğinde veriler genellikle birden fazla farklı kaynaklardan toplanmaktadır ve bu durum doğal olarak veri tiplerinde uyumsuzluklara neden olabilmektedir. Veri uyumsuzluklarından en sık karşılaşılan örnekler; farklı zamanlara ait olmaları, kodlama farklılıkları (örneğin bir veri tabanında cinsiyet özelliğinin e/k, diğer bir veri tabanında 0/1 olarak kodlanması) ve farklı ölçü birimleridir. Ayrıca verilerin toplanırken vuku bulunan çevrenin koşulları da veriler arasında bir örüntü oluşturabileceği sebebiyle uyumsuzluklara yol açabilmektedir. Bu nedenlerle, iyi sonuç alınacak modeller ancak iyi verilerin üzerine kurulabileceği için, toplanan verilerin ne ölçüde uyumlu oldukları bu adımda incelenerek değerlendirilmelidir(Çetin, 2009).

2.5.2.3. Birleştirme ve temizleme

Birleştirme ve temizleme aşamasında çeşitli depolardan derlenen verilerde bulunan ve bir önceki aşamada bahsedilen problemler mümkün olduğu kadar giderilerek veriler tek bir veri kaynakta toplanır fakat basit metotlar ile ve baştan savma şekilde sorun giderme işlemlerinin, sonraki süreçlerde daha karmaşık sorunlara sebep olabilmektedir. Bu aşamada yapılam işlemlerin amacı, veri setindeki uygunsuz, eksik veya hatalı verileri temizlemektir.

2.5.2.4. Seçim

Seçim aşamasında kurulacak modele bağlı olarak veri seçimi yapılır. Örneğin tahmin edici bir model için, bu adım bağımlı ve bağımsız değişkenlerin ve modelin eğitiminde kullanılacak veri kümesinin seçilmesi anlamını taşımaktadır. Çalışmanın çıktısı üzerinde herhangi anlamlı bir etkisi olmayan ve diğer değişkenlerin modeldeki ağırlığının çarpıtılmasına da neden olabilecek değişkenlerin modele alınmaması gerekmektedir. Bağımsız değişkenlerin seçilmesinde verilerin görselleştirilmesine olanak sağlayan bazı yazılımlar ve bunların sunduğu ilişkiler önemli faydalar sağlayabilir. Veri setinin çok büyük hacimli veri içermesi durumunda rassallığı bozmayacak şekilde örneklem seçilmesi uygun olmaktadır. Günümüzde veri analizi

teknikleri gelişmiş olmasına rağmen, büyük hacimli veri setlerindeki analizler çok fazla zaman almaktadır. Bu nedenle, rassal olarak seçilmiş bir veri tabanı örneklemi üzerinden en doğru ve tutarlı modelin ayıklanması önem arz etmektedir.

2.5.2.5. Dönüştürme

Veri dönüşümünde amaç, mevcut veriyi farklı formatlara veya değerlere dönüştürebilmektir. Mesela hastalık riskinin tahmini için geliştirilen bir modelde, hastalanma sayısı gibi önceden hesaplanmış bir oran yerine, birbirinden bağımsız olarak hastaya ait birtakım verilerin kullanılması tercih edilmelidir. Ayrıca modelde kullanılan algoritma, verilerin dönüştürülmesinde de önemli rol oynayacaktır. Örneğin bir uygulamada bir destek vektörü makinelerinin kullanılması durumunda ikili değişken değerlerinin evet/hayır olması; bir karar ağacı algoritmasının kullanılması durumunda ise örneğin gelir değişken değerlerinin yüksek/orta/düşük olarak gruplanmış olması modelin etkinliğini artıracaktır.

2.5.3. Modelin kurulması ve değerlendirilmesi

Eldeki probleme en uygun tekniklerinseçilmesi, yeterli sayıdakitekniklerin denenmesi ile mümkün olabilir. Bu sebeple veri önişleme ve model kurma aşamaları, en iyi sonucu veren tekniğeulaşıncaya kadar tekrarlayan bir süreçtir. Modelleme aşaması, denetimli (Supervised) ve denetimsiz (Unsupervised) öğrenimin kullanıldığı tekniklere göre farklılık göstermektedir. Literatürde eğitimli öğrenme olarak da bilinen denetimli öğrenmede, veri madencisi tarafından veri setindeki sınıflar önceden belirlenerek her sınıf için farklı örnekler verilir. Bu öğrenmede analiz sürecinde verilen örneklerden yola çıkarak her bir sınıfa ilişkin niteliklerinçıkarılması ve bu niteliklerinkurallar ile tanımlanmasıdır. Bu süreç tamamlandığında, tanımlanan kurallarseçilen yeni örneklere uygulanır ve yeni örneklerdeki nesnelerin hangi sınıfa ait olduğu oluşturulanteknik tarafından belirlenir. Denetimsiz öğrenme süreçlerinde, var olan örneklerin özellikleri arasındaki benzerliklerden yola çıkarak sınıfların tanımlanması amaçlanmaktadır.

Denetimli öğrenmede seçilen tekniğe uygun hazırlanan verilerin, ilk aşamada bir kısmı modelin öğrenimi, diğer kısmı ise modelin performansının test edilmesi için ayrılır. Modelin eğitilmesi, eğitim verisi üzerinde uygulandıktan sonra, test verisi ile modelin doğruluk oranı belirlenir. Bir modelin performansının test edilmesinde kullanılan en basit teknik basit geçerlilik testidir. Bu teknikte, veri kümesinin yaklaşık olarak %5 ile %33 aralığındakiveriler test verileri olarak ayrılır ve modelin öğrenimi

sağlandıktan sonra kalan veriler üzerinde test işlemi yapılır. Bir sınıflandırma tekniğinde yanlış sınıflanan nesne sayısının, tüm nesne sayısına bölünmesi ile hata oranı, doğru olarak sınıflanan nesne sayısının tüm nesne sayısına bölünmesi ile ise doğruluk oranı elde edilir. Sınırlı miktarda veriye sahip olunması durumunda, kullanılabilecek diğer bir yöntem çapraz geçerlilik (Cross Validation) testidir. Bu yöntemde veri kümesi Öğrenim Kümesi ve Test Kümesi olmak üzere tesadüfi olarak iki eşit parçaya ayrılır. İlk parça üzerinde model eğitilirken diğer parçası üzerinde test işlemi gerçekleştirilir. Sonrasında parçalar değiştirilerek model tekrar çalıştırılır ve test edilir. Elde edilen iki farklı hata oranlarının ortalaması kullanılır. Bir diğer doğrulama yöntemi ise, verilerin n gruba ayrıldığı n katlı çapraz geçerlilik (N-Fold Cross Validation) yöntemidir. Verilerin genellikle 10 gruba(10-Fold) ayrıldığı bu yöntemde, ilk aşamada birinci grup test, diğer gruplar öğrenim için kullanılır. Bu süreç her defasında bir grubun test, diğer grupların öğrenim amaçlı kullanılması ile sürdürülür. Sonuçta elde edilen on hata oranının ortalaması, kurulan modelin tahmini hata oranı olmaktadır.

Model kuruluşu çalışmalarının sonucuna bağlı olarak, aynı teknikle farklı parametrelerin kullanıldığı veya başka algoritma ve araçların denendiği değişik modeller kurulabilir. Öngörü neticesinde, hangi tekniğin en uygun olduğuna karar verebilmek güçtür. Bu nedenle farklı modeller kurulması, doğruluk derecelerine göre en uygun modelin seçilmesi için sayısız deneme yapılması gerekmektedir. Özellikle sınıflama çalışmalarında kurulan modellerin doğruluk derecelerinin değerlendirilmesinde basit ancak önemli bir özet sunan *Karışıklık (Confusion)* matrisi kullanılmaktadır. Karışıklık matrisi, Çizelge 2.1'de görüleceği üzere örnek kümesindeki gerçek sınıf etiketi ile tahmin edilen sınıf etiketi sayılarını içerir (Han vd, 2011).

Çizelge 2.1'de a değeri *True Pozitive (TP)*, b değeri *False Negative (FN)*, c değeri *False Pozitive (FP)* ve d değeri *True Negative (TN)* olarak adlandırılmaktadır. Sonrasında bu değerler Bölüm 4.3'te kullanılacaktır.

Çizelge 2.1. Karışıklık Matrisi

		Tahminlenen	
		Sınıf=1	Sınıf=0
Gerçekleşen	Sınıf=1	a	b
	Sınıf=0	c	d

2.5.4. Modelin kullanılması

Kurgulanan ve performansı kabul gören bir tekniğin tek bir uygulama olabileceği gibi, diğer uygulamanın alt parçası olarak kullanılabilir. Kurulan modellerde kredi değerlendirme, dolandırıcılık tespiti,risk analizi gibi uygulamalarda direkt kullanılabilmektedir.

2.5.5. Modelin izlenmesi

Genel olarak tüm sistemlerde zamana bağlı olarak verilerde ortaya çıkan değişikliklerden dolayı, kurulan modellerin sürekli olarak güncelliğinin kontrol edilmesi ve gerekiyorsa yeniden düzenlenmesi gerekmektedir. Tahmin edilen ve gerçekleşen değişkenler arasındaki farklılığı vurgulayan performans göstergeleri model sonuçlarının izlenmesini olağan kılmaktadır.

2.6. Makine Öğrenmesi Metodolojisi

Araştırmacılar, yoğun bir şekilde yeni makine öğrenmesi yöntemleri geliştirmektedir. Bu gelişen alanlara yeni bilgi türlerinin araştırılması, çok boyutlu makine öğrenmesi problemleri, diğer disiplinlerden alınan tekniklerle entegre modeller gibi süreçler örnek olarak verilebilir. Buna ek olarak, makine öğrenmesi metodolojileri veri belirsizliği, gürültü ve veride eksiklik gibi olumsuzluklara da çözümler üretebilmelidir. Bazı madencilik yöntemleri, keşfedilen modellerin anlamlılığını değerlendirmek ve keşif sürecini yönlendirmek için veri madencisinin performans değerlerini nasıl kullanılabileceğini de araştırmaktadır. Makine öğrenimi metodolojileri tek boyutlu problemlerde makine öğrenimi, çok boyutlu problemlerde makine öğrenimi olarak farklı uygulamalar içermektedir.

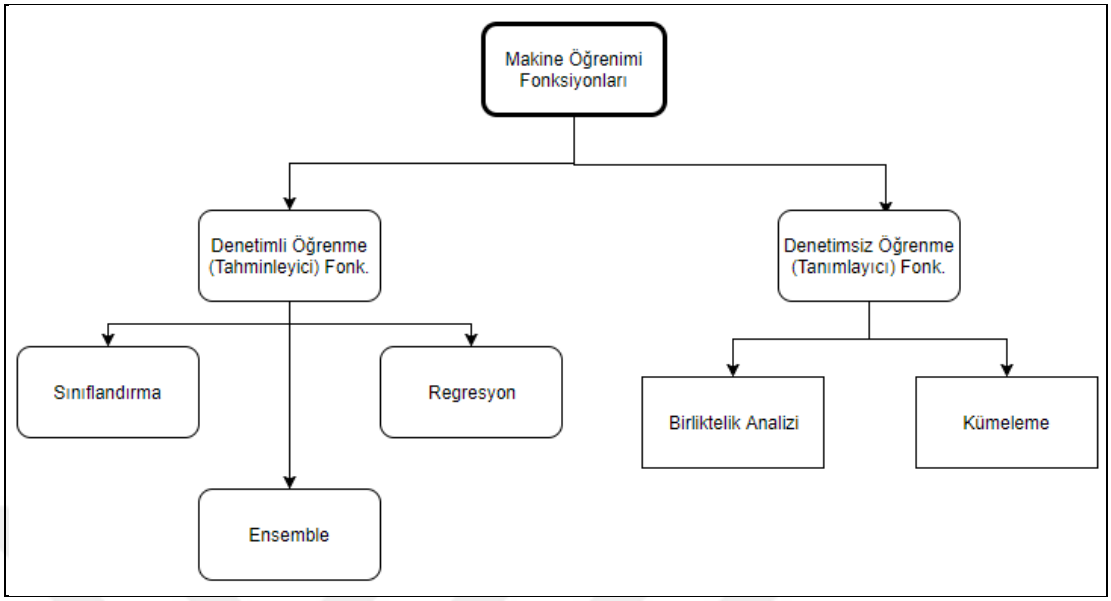
Tek boyutlu problemlerde makine öğrenimi, veri analizi ve bilgi bulma görevleri, veri karakterizasyonu ve farklılıkları ilişkilendirme ve korelasyon analizi, sınıflandırma, regresyon, kümeleme, outlier analizi, sıra analizi ve trend ve evrim analizi gibi konuları içermektedir. Bu görevler, aynı veri seti üzerinde farklı algoritmalar ile denererek sonuçlanmaktadır. Uygulamaların çeşitliliğe açık yapısı sebebiyle, yeni veri madenciliği yöntemleri de ortaya çıkmaya devam etmektedir ve veri madenciliğini dinamik ve hızlı büyüyen bir alan haline getirmektir.

Çok boyutlu problemlerde makine öğrenimi, geniş veri dünyası içerisinde gerekli bilgiyi ararken, çok boyutlu uzaydaki verileri de keşfetmeyi içermektedir. Diğer bir deyişle, verisetindeki nitelikler arasındaki kombinasyonları birer örüntüye sahip olduğu farkedilebilir. Bu analiz süreci, (keşif amaçlı) çok boyutlu veri madenciliği olarak da bilinir. Birçok durumda, veriler bir araya getirilebilir veya çok boyutlu bir veri küpü olarak görüntülenebilir. Küp alanındaki madencilik bilgisi, makine öğreniminin güç ve esnekliğini önemli ölçüde artırabilir (Han vd, 2011).

2.7. Makine Öğrenimi Fonksiyonları

Makine öğrenimi, yapılan analizde bir sonucu tahmin etmek (tahminleyici) ya da belirli bir sonucu tanımlamak (tanımlayıcı) amacıile kullanılmaktadır. Bu nedenle makine öğrenimi fonksiyonları, tahmin edici ve tanımlayıcı olmak üzere iki ana görev altında toplanmaktadır.

Tahminleyicifonksiyonlarda amaç, bir nesnenin bir özelliğinin diğer özelliklerinden yola çıkılarak öngörülmesidir. Öngörülen değer sürekli sayısal bir değer olabileceği gibi kategorik bir değeri de olabilir. Öngörü aşamasında model geçmiş verileri kullanarak eğitilir ve bu sebeple bu fonksiyonlar ***Denetimli Öğrenme*** yöntemleri olarak bilinirler (Şekil 2.2). Tanımlayıcı görevler ise, makine öğrenimi teknikleri aracılığıyla, özellikler arasındaki ilişkilerin ortaya çıkarılmasını amaçlar ve bunu yaparken model bir eğitim sürecine tabi tutulmaz. Bu nedenle tanımlayıcı görevler ***Denetimsiz Öğrenme*** olarak bilinmektedir.



Şekil 2.1. Makine öğrenmesi fonksiyonları

2.7.1. Denetimli öğrenme (supervised) fonksiyonları

Gelecek ile ilgili bir sonucu öngörebilmek için geçmiş verilerden yararlanan fonksiyonlardır. Yeni bir değişkenin niteliklerini analiz etme ve bu değişkenin belirlenmiş bir grubasınıflandırılması aşamasında geçmişte gerçekleşen değişkenlerin hangi sınıfa ait olduğu bilgilerinden faydalanır. Modellemelerinde muhtemel sonucu öngörmeye yarayan faktörler ve sonuç yer alır. Model kurulurken geçmiş verilerde, faktörlerin aldığı değerlere göre elde edilen sonuçlar girdi olarak kullanılır. Beklenen sonuç; "Evet-Hayır" şeklinde ikili veya kategorik değer veya rakamsal değerdir. Tahmin edilen sonuçların performans değeri tahmin edilen sonuç kadar önemlidir. Çoğunlukla tahmin edilen sonuç ile birlikte, bu sonucun kalitesine yönelik; güven aralığı, olasılığı, doğruluk oranı vb. değerleri belirlenir. Tahmin edici modellerde, sonuçları bilinen verilerden hareket edilerek bir model geliştirilmesi ve kurulan bu modelden yararlanılarak sonuçları bilinmeyen veri kümeleri için sonuç değerlerin tahmin edilmesi amaçlanmaktadır. Tahminleyici fonksiyonlarda amaç veri tabanındaki bazı alanların diğer alanlara bağlı olarak tahmin edilmesidir. Tahmin edilecek alan eğer sayısal (sürekli) bir değişken ise tahmin problemi bir regresyon problemi olacaktır. Eğer tahmin edilecek alan kategorik bir değişken ise sınıflama problemidir.

Sınıflama ve regresyon için kullanılan çok fazla sayıda değişken bulunmaktadır. Tahmin edici modellerde problem; diğer alanlardaki (girdiler), her gözlem için hedef değişken değerinin verilmiş olduğu eğitim veri seti ve problem hakkında önceden sahip olunan bilgileri yansıtan varsayımların kümesinin verilmesi durumunda tahmin edilecek değişkenin alabileceği muhtemel değerin belirlenmesi şeklinde yorumlanabilir (Çetin, 2009).

2.7.1.1. Sınıflandırma (classification)

Sınıflandırma, farklı veri sınıflarını gruplandırabilen bir veri analizi biçimidir. Sınıflandırıcılar adı verilen bu tür modeller, kategorik (ayrık, sırasız) sınıf etiketlerini tahminler. Örneğin, banka kredisi uygulamalarını güvenli veya riskli olarak kategorize etmek için bir sınıflandırma modeli oluşturulabilir. Bu tür bir analiz, büyük verilerin daha iyi anlaşılmasına yardımcı olmaktadır. Literatürde makine öğrenimi, kalıp tanıma ve istatistik konularında birçok sınıflandırma yöntemi önerilmiştir. Çoğu algoritma, genellikle küçük bir veri boyutu kullanarak uygulanmaya başlamaktadır.

Son zamanlarda veri madenciliği fonksiyonlarına ilişkin daha derin çalışmalar yapılmış ve büyük miktarda yerleşik veriyi yönetebilen ve ölçeklenebilir sınıflandırma ve tahmin teknikleri geliştirilmiştir. Sınıflandırma, dolandırıcılık tespiti, pazarlama, performans tahmini, imalat ve tıbbi teşhis dahil olmak üzere çok sayıda uygulamaya sahiptir. Bir banka kredisi sorumlusu, banka için hangi kredi başvuranlarının "güvenli" ve "riskli" olduğunu öğrenmek için her bir verinin analiz edilmesine ihtiyaç duymaktadır. Örneğin bir teknoloji mağazasında pazarlama müdürü, belirli bir profili olan bir müşterinin yeni bir bilgisayar satın alacağını tahmin etmesine yardımcı olmak için veri analizine ihtiyaç duyar. Ya da bir tıbbi araştırmacı, bir hastanın alması gereken üç spesifik tedaviden hangisinin daha anlamlı olacağını öngörmek için meme kanseri verilerini analiz etmek isteyebilmektedir. Veri analizinin görevi, kredi başvurusu verileri örneğinde "güvenli" veya "riskli" gibi sınıf (kategorik) etiketlerini tahmin etmek iken; pazarlama örneğinde "evet" veya "hayır"; veya hastalık teşhisi örneğinde "tedavi A", "tedavi B" veya "tedavi C" olarak değişkenleri sınıflandırmaktır.

Bu kategoriler, değerler arasındaki sıralamada bir anlam taşımayan ayrı değerlerle temsil edilebilir. Örneğin, 1, 2 ve 3 değerleri, A, B ve C işlemlerini temsil etmek için kullanılabilir; burada, bu tedavi sınıflarının ne ile adlandırıldığının herhangi bir hükmü yoktur (Han ve ark., 2011).

Makine Öğrenmesi çalışmalarında sınıflandırma problemlerinde kullanılan başlıca algoritmalar aşağıda sıralanmaktadır.

- ✓ Ensemble Methods
- ✓ Logistic Regression
- ✓ Clustering Algorithms
- ✓ Principal Component Analysis
- ✓ Singular Value Decomposition
- ✓ Independent Component Analysis
- ✓ Support Vector Machine (SVM)
- ✓ Multi Layer Perceptron (MLP)
- ✓ Decision Tree (J48, C4.5)
- ✓ Naive Bayes (NB)
- ✓ Ordinary Least Squares Regression

2.7.1.2. Regresyon (eğri uydurma)

Regresyon analizi, aralarında sebep-sonuç ilişkisi bulunan bir veya daha fazla sayıda tahmin değişkenleri arasındaki ilişkinin niteliğini saptamayı hedeflemektedir. Regresyon analizi, kurulan ilişkiyi kullanarak ilgili tahminler (estimation) ya da kestirimler (prediction) yapmak amacıyla kullanılmaktadır. Tahmin aşamasında tek değişken kullanılırsa basit regresyon, iki veya daha fazla değişken kullanıldığı durumlarda çoklu regresyon analizi olarak ikiye ayrılmaktadır. Buradaki amaç tahmin değişkeninin ölçüt değişkenindeki toplam değişmeye olan katkısının saptanması ve dolayısıyla tahmin değişkenlerinin doğrusal kombinasyonunun değerinden hareketle ölçüt değerinin tahmin edilmesidir. Daha genel bir tabir ile regresyon analizi, aralarında sebep-sonuç ilişkisi bulunan iki veya daha fazla değişken arasındaki ilişkiyi belirlemek ve bu ilişkiyi kullanarak o konu ile ilgili tahminler ya da kestirimler yapabilmek amacıyla yapılır. Bu analiz tekniğinde tek (basit regresyon) veya daha fazla değişken (çoklu regresyon) arasındaki ilişki açıklamak için matematiksel bir model kullanılır ve bu model regresyon modeli olarak adlandırılır. Değişkenler arasındakidesençıkarıldıktan sonra, bağımsız değişkenlerden çıkarımlar yaparak bağımlı değişkenin değeritahminlenebilir. Regresyon analizinde, açıklanan ya da tahmin edilen veri grubu Bağımlı Değişken (y) olarak tanımlanır. Bağımsız Değişken (x), regresyon modelinde açıklayıcı veri grubu olup; bağımlı değişkenin değerini tahmin etmede kullanılır. Bağımlı değişkenin bağımsız değişken ile ilişkili olduğu varsayılır. Bir tek bağımsız değişkenin kullanıldığı regresyon tek değişkenli regresyon analizi, birden fazla

bağımsız değişkenin kullanıldığı regresyon analizi de çok değişkenli regresyon analizi olarak adlandırılır.

Tek Değişkenli Regresyon Analizi; bir bağımlı değişken ve bir bağımsız değişken arasındaki ilişkiyi incelemekte iken, Çok Değişkenli Regresyon Analizi'nde bir bağımlı değişken ve birden fazla bağımsız değişkenler arasındaki ilişkiyi incelemektedir. Ayrıca bağımlı ve bağımsız iki değişken arasında eğrisel bir ilişki var ise değişkenler arasındaki ilişki eğrisel regresyon modeli ile açıklanır.

Regresyon modellerinde kullanılan başlıca teknikler şunlardır:

- ✓ Yapay Sinir Ağları,
- ✓ Doğrusal Regresyon,
- ✓ Genetik Algoritmalar,
- ✓ Karar Destek Makineleri,
- ✓ Bellek Temelli Nedenleme,
- ✓ Naive Bayes,
- ✓ Karar Ağaçları.

2.7.1.3. Ensemble

Ensemble, eğitilebilir ve daha sonra tahminlerde bulunulabilir olmasından dolayı denetimli bir öğrenme algoritmasıdır. Ensemble, diğer öğrenme algoritmalarında olduğu gibi tek bir hipotezi temsil eder ancak diğer yöntemlerden farklı olarak birden fazla sınıflayıcı yöntemlerden oluşmaktadır. Ensemble ile temsil edilen hipotez, onun oluşturulduğu sınıflayıcıların hipotezleri birebir olarak kapsamak zorunda değildir. Bu nedenle, Ensemble algoritması diğer makine öğrenmesi algoritmalarına göre daha esnek tahminler üretebilmektedir. Bu esneklik, teorik olarak, eğitim verilerinin tek bir modelde daha hızlı öğrenme performansı sağlar, ancak pratikte bazı Ensemble teknikleri (özellikle bagging), eğitim verilerinin aşırı öğrenme gibi problemlere meyillidirler.

Genel olarak, Ensemble, kendisini oluşturan modelleri arasında anlamlı bir çeşitlilik olduğunda daha iyi sonuçlar verir. Birçok Ensemble yöntemi, bu nedenle, birleştirildikleri modeller arasında çeşitliliği teşvik etmeyi amaçlamaktadır. Belki de sezgisel olmayan, daha rastgele algoritmalar (rastgele karar ağaçları gibi) çok kasıtlı algoritmalara (entropi azaltıcı karar ağaçları gibi) kıyasla daha güçlü bir Ensemble oluşturmaktadır.

2.7.2. Denetimsiz öğrenme (unsupervised) fonksiyonları

Bazı uygulama senaryolarında, "normal" veya "aşırı" olarak etiketlenmiş nesneler mevcut değildir. Daha basit bir tabirle önceden niteliği belli olan nesneler mevcut değildir. Böyle durumlarda, denetlenmeyen bir öğrenme yöntemi kullanılmalıdır.

Denetimsiz aykırı tanımlama yöntemleri bir varsayım üzerine çalışırlar: Normal nesneler bir miktar "kümelenmiş" olan nesnelerdir. Diğer bir deyişle, denetimsiz bir aykırılık tespiti yöntemi, uydurabilmek için normal nesnelerin aykırı kalıplardan çok daha sık izlediği deseni tespit etmeye çalışır.

Normal nesnelerin yüksek benzerliği paylaşan bir gruba girmesi gerekmez. Bunun yerine, onlar her grubun belirgin özelliklere sahip olduğu birden fazla grup oluşturabilir. Bununla birlikte, normal nesnelerin bu gruplarından herhangi birinden özellik alanından uzakta bir çıkarsamanın olması beklenmektedir. Bu varsayım, her zaman doğru olmayabilir.

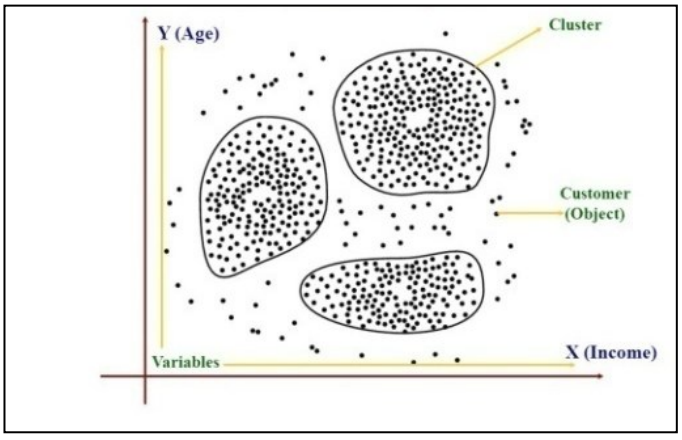
Denetimsiz yöntemler, bu türden aykırı değerleri etkili bir şekilde tespit edemez. Bazı uygulamalarda, normal nesneler çeşitli şekilde dağıtılır ve birçok nesne belirgin örüntüleri takip etmez. Örneğin, bazı saldırı tespiti ve bilgisayar virüsü bulma problemlerinde, normal faaliyetler çok çeşitlidir ve birçoğu yüksek kaliteli kümelere girmez. Bu tür senaryolarda, denetlenmeyen yöntemler yanlış pozitif bir orana sahip olabilir - birçok normal nesneyi normal dışı etiketler olarak (bu uygulamalardaki müdahaleler veya virüsler) yanlış etiketleyebilir ve birçok gerçek aykırı değer tespit edilmemiş olabilir. Böyle durumlarda, müdahalelerle virüsler arasındaki yüksek benzerliğe (yani, hedef sistemlerde önemli kaynaklara saldırmak zorunda olmaları) sahip olan nesneler denetlenen yöntemlerle modellemek çok daha etkili olabilir. Çoğu kümeleme yöntemi, denetimsiz tespit yöntemleri gibi davranmak üzere uyarlanabilir. Bu konudaki ortak fikir, önce kümeler bulmaktır ve sonra herhangi bir kümeye ait olmayan nesneleri aykırı değerler olarak tespit etmektir. Bununla birlikte, bu yöntemlerin iki sorunu vardır. İlk olarak, herhangi bir kümeye ait olmayan bir veri nesnesi, bir aykırı değer değil de bir gürültü verisi olabilir. İkincisi, öncelikle kümelenmeleri bulmak ve sonra aykırı değerler bulmak pahalı ve zaman alıcı olabilir.

Genellikle, aykırı nesnelerde normal nesnelerden daha az sapma olduğu varsayılır. Normal olan nesneler tanımlanmadan önce hedeflenmeyen veri girdilerinin büyük bir popülasyonunu işleme koymak uygulamada pratik olmayabilmektedir. Literatürde, son

olarak denetimsiz aykırılık tespit yöntemleri, açık ve kesin kümeler bulmadan doğrudan aykırı değerleri tespit etmek gibi çeşitli fikirler öne sürülmektedir (Han vd, 2011).

2.7.2.1. Kümeleme (clustering)

Veri madenciliğinden kullanılan önemli tekniklerinden biri kümeleme analizidir. Kümeleme analizinin amacı, nesnelerin temel niteliklerini dikkate alarak gruplara ayırmak ve araştırmacıya özetleyici bilgiler sunmaktır. Kümeleme analizi, araştırmada gözlenen birimlerin, ölçülen tüm değişkenler üzerindeki değerlerini hesaplayarak birbirine benzeyen birimleri aynı küme içinde sınıflandırır. Analiz, ortaya çıkacak kümelere ve gruplara odaklanmaktadır ve elde edilen kümelerin kendi içlerinde homojen, kendi aralarında ise heterojen bir yapıda olmaları beklenir. Birim ve bu birimlere ait değişkenlerin sınıflamaları hakkında kesin bilginin bulunmadığı bir popülâsyondan alınan n tane birimin, p tane değişkene ilişkin gözlem sonuçları ile ilgilenir. Kümelemede yukarıda değinildiği gibi homojen nesneleri birbiri ile birleştirerek heterojen gruplar oluşturulur ve birimler hiyerarşik bir düzene sokulur. Sınıflandırma yapmak gözlem sonuçlarının çok az bir kayıpla bir araya toplanmasını sağlar. Şekil 2.3'te görüldüğü üzere ortaya çıkacak kümelerin kendi içlerinde homojen, kendi aralarında ise heterojen bir yapıda olmaları beklenir.



Şekil 2.2. Kümeleme örneği

Kümeleme analizinin temel amacı, gözlenen birey ya da nesneler arasındaki benzerlikleri ya da uzaklıkları tespit etmektir. Benzerlik iki nesne veya iki özellik arasındaki ilişkinin kuvveti olarak açıklanır. Bu nicel değer alınan ölçeğe veya veri tipine göre değişik yollardan elde edilir. Uzaklık ise, iki nesne arasındaki zıtlık ya da uyumsuzluğun bir ölçüsü olan farklılıkları ölçer. Benzerlik ve uzaklık ölçümleri

gözlemlerin birbirinden ayırt edilmesini sağlar ve bu sayede gözlemler gruplara ayrılır. Uzaklık ölçümü, verilerin nicel veya karışık veriler olmasına göre farklılık göstermektedir. Nicel veriler için uzaklık ölçümü çeşitli matematiksel uzaklık hesabı yöntemler ile yapılmaktadır. Ancak bu konu tez kapsamında olmadığı için detaylandırılmamıştır.

Kümeleme modellerinde kullanılan başlıca teknikler ise şunlardır:

- Bölme yöntemleri,
- Grid tabanlı yöntemler,
- Hiyerarşik yöntemler,
- Hiyerarşik olmayan yöntemler,
- Yoğunluk tabanlı yöntemler,
- İki aşamalı kümeleme tekniği,
- Model tabanlı yöntemler .

2.7.2.2. Birliktelik analizi / ilişki kuralları (association rules)

Veri madenciliğinde Birliktelik Kuralları, 1993 yılında Agrawal ve arkadaşları tarafından önerilmiştir. Birliktelik Kuralları, büyük veri kümeleri arasındaki ilişkileri çıkaran ve nesnelerin birlikte gerçekleşme ihtimallerini analiz eden denetimsiz veri madenciliği yöntemidir. Birliktelik Kuralları; pazarlama eğitim, sağlık, ekonomi, telekomünikasyon, e-ticaret, bilişim gibi pek çok alanda yaygın olarak kullanılmaktadır.

Birliktelik analiz problemlerine verilen bilinen örnek çocuk bezi ve bira örneğidir. Walmart ismindeki süpermarket zincirinin yaptığı analiz şu şekildedir; "Yeni çocuk sahibi olan ebeveynler eğlenmek için vakit bulamayıp cuma günlerini partiye gitmek yerine eve bira alarak evde geçirmek zorunda kalıyorlar. Bu yüzden bebek bezi alan baba çoğunlukla yanında bir de bira alır," gibi bir çıkarımda bulunmaktadırlar. Bu doğrultuda bir market bebek bezi ve birayı yakın yerlere koyarak satışlarını arttırmayı hedefleyebilmektedir.

Birliktelik analizinin tanımlanmasından kullanılan diğer bir örnek ise internet üzerinden yapılan alışverişler gösterilebilmektedir; müşteriye satın aldığı kıyafetten yola çıkarak daha önce bu kıyafet ile satın alınma ihtimali en yüksek olan tamamlayıcı kıyafetler önermektedir. Bu analiz hala pek çok web sitesinde aktif olarak uygulanmaktadır.

Birliktelik Kuralları'nda kullanılan bazı temel kavramlar aşağıda verilmiştir;

- Bir veya daha çok öğeden oluşan kümeler *Öğeler Kümesi* adlandırılmaktadır.
- Öğeler kümesindeürün gruplarının görülme sıklığı *Destek Sayısı* olarak ifade edilmektedir.
- Destek: Veride bağıntının ne kadar sık olduğunu tanımlamaktadır ve öğeler kümesinin içinde bulunduğu birlikteliklerin toplam birliktelik sayısına oranı ile hesaplanmaktadır. Destek(A=>B) şeklinde ifade edilmektedir.
- Destek değeri eşik değerden büyük ya da eşit olan öğeler kümesine *Yaygın Öğeler* denmektedir.
- Güven: A malını almış bir kişinin B malını alma olasılığını vermektedir. Öğeler arasındaki birlikteliklerin doğruluğunu ifade etmektedir. Güven(A=>B) şeklinde gösterilmektedir.

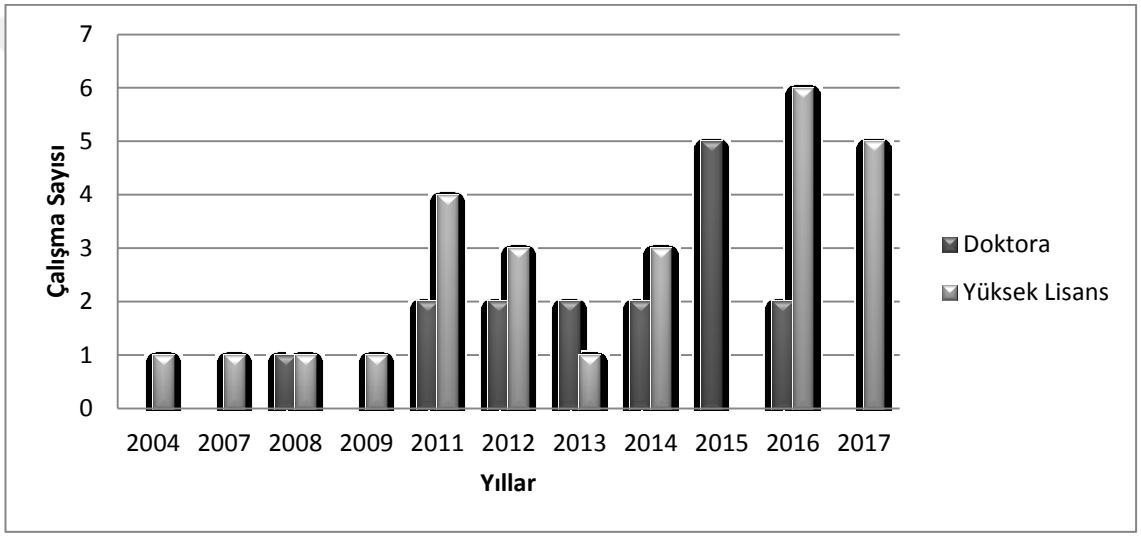
Kavramlara ilişkin bir örnek ile açıklamak gerekirse: Bir markette 1 paket süt alanlar aynı zamanda yoğurt ürününü da alıyorlarsa, bu durum Süt Yoğurt [destek = %3, güven = %70] şeklinde ifade edilir. Buradaki destek ve güven değerleri, birliktelik kuralının farklılık değerleridir. %3 oranındaki bir destek değeri, analiz edilen tüm 50 alışverişlerden %3'sinde süt ile yoğurt satışlarının birlikte olduğunu göstermektedir. %70 oranındaki güven değeri ise 1 paket süt alanların %70'inin aynı zamanda yoğurt da satın aldığını ortaya koymaktadır. Veri madencisi tarafından belirlenen eşik güven değerini aşan nesneler için birliktelik kuralı vardır denilmektedir. Geniş veri setlerinde birliktelik kuralları tespit edilirken, 2 adımdan oluşan bir süreç izlenmektedir. İlk adımda sıklıkla yinelenen nesneler bulunur: Bu nesnelerin her biri en az, önceden belirlenen eşik destek sayısı kadar sık tekrarlanırlar. İkinci aşamada sık tekrarlanan nesneler arasından güçlü birliktelik kuralları oluşturulur. Birliktelik analizi modellerinde kullanılan başlıca yöntem, Apriori algoritmasıdır.

3. LİTERATÜR TARAMASI

Bu aşamada, makine öğrenmesi algoritmalarının kullanıldığı lisansüstü tezler ve literatürdeki çalışmalar irdelenmiştir.

3.1. Makine Öğrenmesi Algoritmalarının Tez Çalışmalarındaki Yeri

YÖK Tez Tarama Merkezinde yapılan taramada, makine öğrenmesi algoritmalarının kullanıldığı 42 adet tez çalışması incelenmiştir. Bu çalışmalardan, makine öğrenmesinin sağlık sektörü ağırlıklı olmak üzere birçok farklı sektörde kullanım alanı bulduğu anlaşılmaktadır (Burakgazi, 2017; Doğrusöz, 2007; Gülden, 2014). Şekil 3.1'te tez çalışmalarının yıllara göre artan bir trende sahip olduğu görülmektedir.



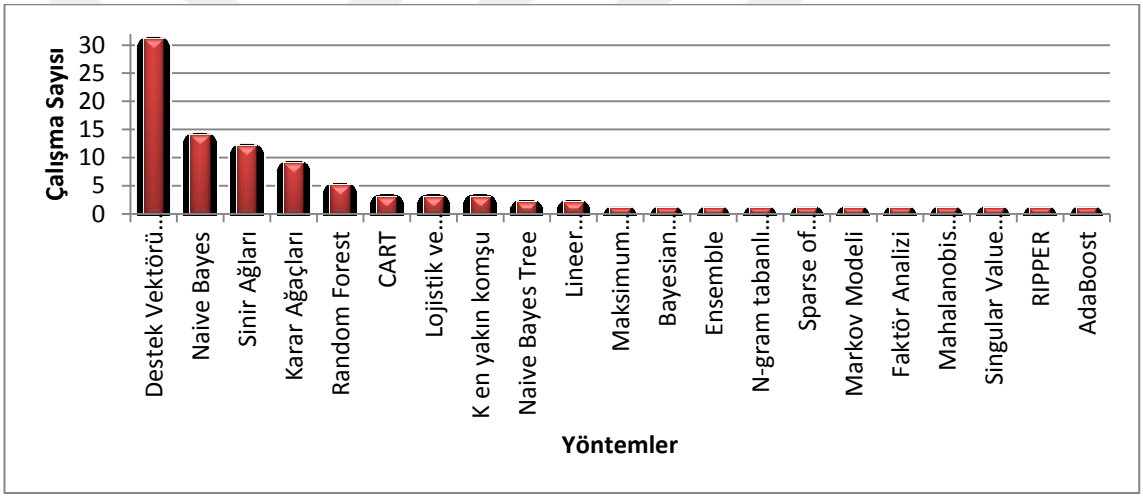
Şekil 3.1. Yıllara göre lisansüstü tez sayıları

Makine öğrenmesi algoritmalarına dair olan uygulamalar hem doktora hem de yüksek lisans tezlerinde yer edinmiştir. Bu tezlerin %95'i Fen Bilimleri Enstitüleri tarafından yürütülmüştür. Tezler incelendiğinde temel makine öğrenmesi algoritmalarının yaygın bir şekilde kullanıldığı ve melez modellerin henüz yeterli ilgiyi çekmediği anlaşılmaktadır. Bu tez çalışmasında bu eksikliğe binaen, melez modellerin kullanımı gerçekleştirilmiştir.

3.2. Yaygın Olan Algoritmalar

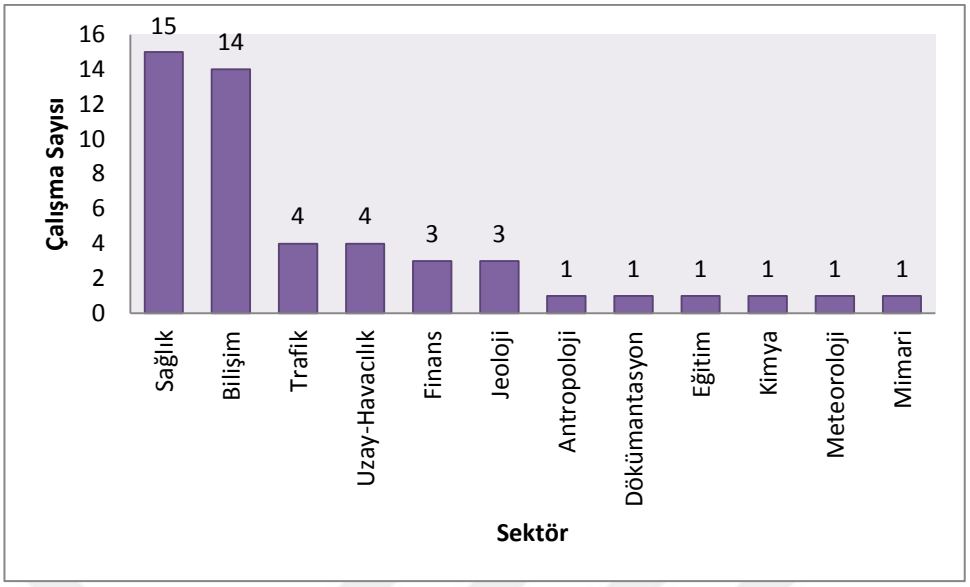
Bu aşamada tez kapsamında kullanılacak makine öğrenmesi algoritmalarını tespit edebilmek ve bu algoritmalar üzerinde melez modeller kurgulayabilmek amacıyla hangi modellerin yaygın olduğu araştırılmıştır.

Bu doğrultuda Google Scholar üzerinden yapılan araştırmalarda yüksek oranda atıf alan akademik çalışmalar (Çizelge 3.1) incelenmiş ve Şekil 3.2'de kullanım sayılarına göre özetlenmiştir. Grafik göz önüne alındığında en sık kullanılan 4 farklı makine öğrenmesi algoritması sırasıyla; Destek Vektörü Makineleri, Naive Bayes Sınıflandırıcısı, Sinir Ağları ve Karar Ağaçları olarak tespit edilmiştir. Bu nedenle tez kapsamında tespit edilen 4 farklı algoritma üzerinde yoğunlaşılmış ve bu algoritmaların kendi aralarındaki performans değerleri kıyaslanmıştır.



Şekil 3.2. Makine öğrenmesi algoritmalarının kullanım sıklığı

Çizelge 3.1'deki çalışmalara göre makine öğrenmesi algoritmalarının en yaygın kullanıldığı sektörler Şekil 3.3'de verilmektedir. Sağlık ve Bilişim sektörlerinde yapılan çalışmaların çoğunlukta olduğu anlaşılmaktadır. Sağlık sektöründe hastalıkların teşhisine karar vermede kullanılırken, bilişim sektöründe ise eldeki gerekli ve önemli bir takım verileri belirli bir sınıfa ayırma durumda kullanılmaktadır (Gray, 2013; Güvenç, 2016). Bu çalışma kapsamında da elde edilen bulguların karşılaştırma imkanı sağlaması açısından diyabet teşhisine (sağlık sektörü) odaklanılması kararlaştırılmıştır.



Şekil 3.3. Sektörlere göre makine öğrenme

Çizelge 3.1. Literatürde farklı makine öğrenmesi çalışmaları

Yazar(lar)	Naive Bayes	Maksimum Entropi	Destek Vektörü Makineleri	Karar Ağaçları	Bayesian Network	Naive Bayes Tree	Sinir Ağları	Lineer Diskriminant Analizi	Random Forest	CART	Ensemble	N-gram tabanlı Karakter Model	Rule Sets	Lojistik ve Lineer Regresyon	K en yakın komşu	Sparse of Network of Winnows	Markov Modeli	Faktör Analizi	Mahalanobis Distance	Singular Value Decomposition	RIPPER	AdaBoost	Diğer
(Davis vd., 1992)							x																
(Joachims, 1998)			x																				
(Drucker vd, 1999)			x	x																			
(DeFries vd, 2000)				x																			
(Drezet ve Harrison, 2001)			x																				
(Pang vd, 2002)	x	x	x																				
(Sebastiani, 2002)			x																				
(Shipp vd, 2002)			x																				
(Zhang ve Lee, 2003)	x		x	x											x	x							
(Cho ve Won, 2003)			x																				
(Byvatov vd, 2003)			x				x																
(Li vd, 2003)			x																				
(Pan vd, 2003)			x																				
(Rosenthal, 2004)	x		x	x										x									
(Statnikov vd, 2004)			x																				
(Melgani ve Bruzzone, 2004)			x																				
(Müller vd, 2004)																							x
(Wei vd, 2005)	x		x			x																	
(Zander vd, 2005)	x																						
(Kim vd, 2005)			x																				
(Wang vd, 2005)			x																				
(Zhang ve Zhou, 2005)			x												x								
(Pal, 2005)			x				x																
(Pal ve Mather, 2005)									x														
(Williams vd, 2006)	x			x	x	x																	

Çizelge 3.1. Literatürde farklı makine öğrenmesi çalışmaları (devamı)

Yazar(lar)	Naive Bayes	Maksimum Entropi Sınıflandırması	Destek Vektörü Makineleri	Karar Ağaçları	Bayesian Network	Naive Bayes Tree	Sinir Ağları	Lineer Diskriminant Analizi	Random Forest	CART	Ensemble	N-gram tabanlı Karakter Model	Rule Sets	Lojistik ve Lineer Regresyon	K en yakın komşu	Sparse of Network of Winnows	Markov Modeli	Faktör Analizi	Mahalanobis Distance	Singular Value Decomposition	RIPPER	AdaBoost	Diğer
(Erman vd, 2006)	x																						
(Cruz ve Wishart, 2006)			x																				
(Larranaga vd, 2006)							x																
(Dai vd, 2007)	x		x				x		x	x				x									
(Romero vd, 2008)																							
(Statnikov vd, 2008)			x						x														
(Ye vd, 2009)	x		x									x											
(Tuia vd, 2009)			x																				
(Chen vd, 2009)	x																						
(Guzella ve Caminhas, 2009)	x		x				x							x									
(Mannini ve Sabatini, 2010)																	x						
(Khandani vd, 2010)										x													
(Otukei ve Blaschke, 2010)			x	x																			
(Pennacchiotti ve Popescu, 2011)								x															
(Figueiredo vd., 2011)							x											x	x	x			
(Kiritchenko ve Matwin, 2011)	x		x																				
(Stallkamp vd, 2012)			x	x			x	x	x		x												
(Kourou vd, 2015)	x		x	x			x																
(Park ve Bae, 2015)	x			x																	x	x	
(Naghibi vd, 2016)									x	x													
(Taravat vd, 2015)			x				x																
(Alby ve Shivakumar, 2016)																							x
(Luo, 2016)							x																
Toplam	**14**	**1**	**31**	**9**	**1**	**2**	**12**	**2**	**5**	**3**	**1**	**1**	**0**	**3**	**3**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**2**

3.3. Melez Model Seçimi

Makine öğrenmesi algoritmalarının performans değerlerinin iyileştirilmesi amacıyla son yıllardan optimizasyon algoritmalarıyla melezlenmesi yaklaşımından yararlanılmaktadır (Kose, 2016; Utomo, 2014). Şekil 3.2'teki grafikte verilmiş olan makine öğrenmesi algoritmaları yerini melez modellere bırakmaya başlamıştır. Makine öğrenmesi algoritmalarının optimizasyon algoritmaları ile melezlenebileceği görüşü ilk olarak Davis (1990) tarafından ileri sürülmüştür. Daha sonrasında ise bu konu ilk çalışma Kelly ve Davis (1991) tarafından yütürülmüştür. Yazarlar çalışmalarından K- en yakın komşu algoritması ile Genetik Algoritmayı melezlemiş ve performans değerlerinin arttığını göstermişlerdir. Kelly ve Davis'in literatüre yeni bir fikir ileri sürmesi 1990'larda başlamış olsa da tam olarak fikrin yaygınlaşması 2000'lerden itibaren mümkün olabilmiştir. Bundan sonraki bazı çalışmalar Çizelge 3.2'de kronolojik bir şekilde özetlenmiştir.

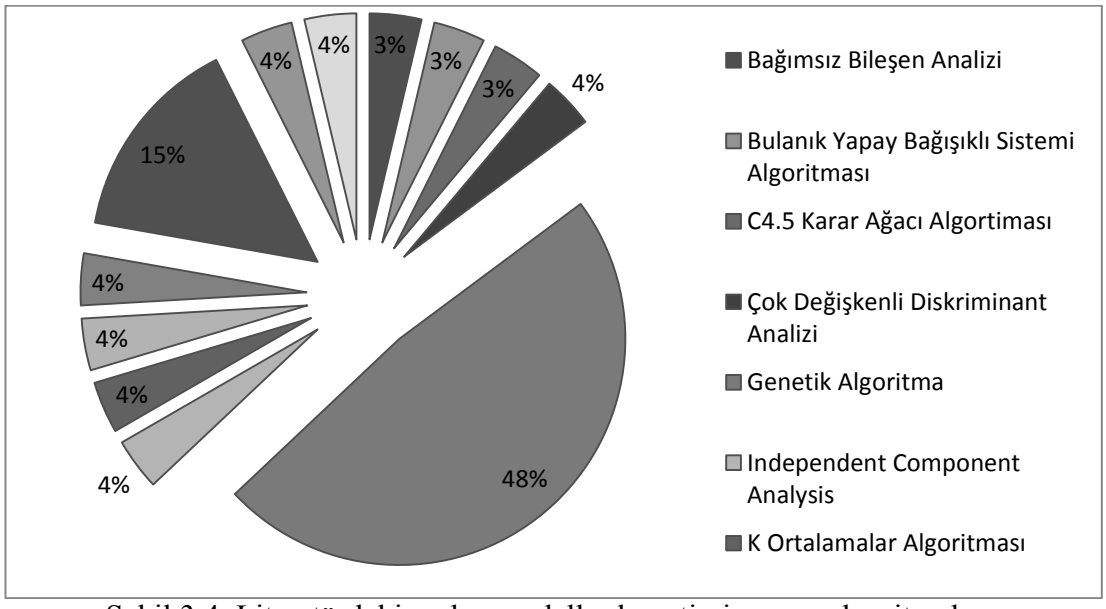
Çizelge 3.2. Literatürde bazı melez modeller

Çalışma	**Ana Model**	**Melezi**
(Lee vd, 1996)	Yapay Sinir Ağları	Çok Değişkenli Diskriminant Analizi
(Pan vd, 2003)	Yapay Sinir Ağları	C4.5 Karar Ağacı Algortiması
(Kim ve Shin, 2007)	Yapay Sinir Ağları	Genetik Algoritma
(Sivagaminathan ve Ramakrishnan, 2007)	Yapay Sinir Ağları	Karınca Kolonisi Optimizasyonu
(Tsai ve Wang, 2009)	Yapay Sinir Ağları	Karar Ağacı
(Aziz vd, 2013)	Yapay Sinir Ağları	Genetik Algoritma
(Qi vd, 2001)	Destek Vektörü Makineleri	Bağımsız Bileşen Analizi
(Li vd, 2005)	Destek Vektörü Makineleri	Genetik Algoritma
(Min vd, 2006)	Destek Vektörü Makineleri	Genetik Algoritma
(Huerta vd, 2006)	Destek Vektörü Makineleri	Genetik Algoritma
(Shon ve Moon, 2007)	Destek Vektörü Makineleri	Genetik Algoritma
(Huang ve Dun, 2008)	Destek Vektörü Makineleri	Parçacık Sürü Optimizasyonu
(Choudhry ve Garg, 2008)	Destek Vektörü Makineleri	Genetik Algoritma
(Kharrat vd, 2010)	Destek Vektörü Makineleri	Genetik Algoritma
(Yang vd, 2010)	Destek Vektörü Makineleri	Independent Component Analysis
(Sartakhti vd, 2012)	Destek Vektörü Makineleri	Tavlama Benzetim
(Basari vd, 2013)	Destek Vektörü Makineleri	Parçacık Sürü Optimizasyonu
(Zheng vd, 2014)	Destek Vektörü Makineleri	K Ortalamalar Algoritması

Çizelge 3.2. Literatürde bazı melez modeller (devamı)

Çalışma	Ana Model	Melezi
(Kelly ve Davis, 1991)	K en yakın komşu	Genetik Algoritma
(Şahan vd, 2007)	K en yakın komşu	Bulanık Yapay Bağışıklı Sistemi Algoritması
(Tahir vd, 2007)	K en yakın komşu	Tabu Araması Algoritması
(Aci vd, 2010)	K en yakın komşu	Genetik Algoritma
	Bayes Metodları	
(Yang vd, 2009)	K Harmonik Ortalamalar	Parçacık Sürü Optimizasyonu
(Niknam ve Amiri, 2010)	K Ortalamalar Algoritması	Parçacık Sürü Optimizasyonu
(Carvalho ve Freitas, 2004)	Karar Ağacı	Genetik Algoritma
(Davis, 1990)	Makine Öğrenmesi Algoritmaları	Genetik Algoritma
(Bies vd, 2006)	NONMEM	Genetik Algoritma

Çizelge 3.2'de taranan literatürde melezleme aşamasında kullanılan optimizasyon algoritmaları Şekil 3.4'de verilen pasta grafiğinde görülmektedir. Buna göre melezleme aşamasında en yaygın olarak kullanılan yöntemin Genetik Algoritmalar olduğu anlaşılmaktadır. Bu tez kapsamında da seçilmiş olan makine öğrenmesi algoritmalarının genetik algoritmalı melez modelleri üzerinde çalışılmıştır.



Şekil 3.4. Literatürdeki melez modellerde optimizasyon algoritmaları

3.4. Diyabet Tahmini Üzerine Yapılan Çalışmalar

Sağlık alanında özellikle teşhis tahmini için yapılan çalışmalarda makine öğrenmesinden yaygın bir şekilde yararlanılmıştır. Bu çalışmalarda özellikle diyabet teşhisinin ön plana çıktığı görülmektedir. Tez aşamasında gerek diğer çalışmalarla karşılaştırma yapma olanağı vermesi gereksede hazır veri kullanımı mümkün olduğundan dolayı makine öğrenmesi yöntemleri ve melez modellerin karşılaştırmasının diyabet teşhisi üzerine yapılmasına karar verilmiştir. Bu aşamada diyabet üzerine yapılan çalışmalar aşağıdaki Çizelge 3.3'de özetlenmiştir.

Çizelge 3.3. Literatürde diyabet çalışmaları

Yazarlar	Algoritma	Pima Indian Veri Seti	Performans (%)
(Acar, 2011)	Yapay Sinir Ağları	+	82,10
(Alby ve Shivakumar, 2016)	Genel Regresyon Sinir Ağları	+	93,40
(Aslam, 2013)	Genetik Algoritmaları Destek Vektörü Makineleri	+	-
(Baraka vdt, 2010)	Destek Vektörü Makineleri	+	94,00
(Diwani ve Sam, 2014)	Naive Bayes	+	76,30
(Doğantekin, 2010)	ANFIS	+	84,61
(Jaafar vd, 2005)	Yapay Sinir Ağları	+	-
(Jhaldiyal ve Mishra, 2014)	Temel Bileşen Analizi ile Destek Vektörü Analizi	+	-
(Kala vd, 2009)	EvrimselYapay Sinir Ağları, ANFIS	+	77,38
(Karegowda vd, 2011)	Genetik Algoritmalı Sinir Ağları	+	84.71
(Kayaer ve Yıldırım, 2003)	Genel Regresyon Sinir Ağları	+	80,21
(Khan vd, 2013)	Yapay Sinir Ağları	+	0.094(RMSE)
(Köse vd, 2016)	Destek Vektörü Tabanlı Bilişsel Gelişim Optimizasyon Algoritması	+	87,50
(Kumari ve Chitra, 2013)	Destek Vektörü Makineleri	+	78,00

Çizelge 3.3. Literatürde diyabet çalışmaları (devamı)

Yazarlar	Algoritma	Pima Indian Veri Seti	Performans (%)
(Polat, 2007)	Sugeno ANFIS	+	89,47
(Sanidad vd, 2012)	Karar Ağacı	+	73,83
(Sarvar, 2014)	Yapay Sinir Ağları	+	96,00
(Saxena, 2014)	K en yakın komşu	+	75,00
(Senthilkumar ve Paulraj, 2013)	Çok Değişkenli Uyarlamalı Regresyon Modeli	+	-
(Sharifi vd, 2008)	Hiyerarşik Takagi-Sugeno Bulanık Sistem	+	78,75
(Smith vd, 1988)	ADAP Öğrenme	+	76,00
(Temurtaş vd, 2009)	Olasılıksal Sinir Ağları	+	82,37
(Thangarajul vd, 2014)	Kümele Yöntemleri	+	0,03(MAPE)
(Ahmed, 2012)	Karar Ağacı		-
(Chen, 2014)	Destek Vektörü Makineleri		99,00
(Chen, 2015)	Yeniden Örnekleme Tabanlı Sinir Ağları		-
(Georga vd, 2013)	Destek Vektörü Makineleri		-
(Heydari vd, 2016)	Yapay Sinir Ağları		97,44
(Huang, 2007)	Naive Bayes, Karar Ağaçları		95,26
(Karahoca, 2009)	Sugeno ANFIS		0,17(RMSE)
(Luangruangrong vd, 2012)	Geri Beslemeli Sinir Ağları		83,65
(Suhaimi ve Ismail, 2012)	Yapay Sinir Ağları		78,40
(Supramani, 2012)	Naive Bayes, Lojistik Regresyon, CART, Random Forest, K en yakın komşu, Destek Vektörü Makineleri		-
(Upadhyay ve Patel, 2016)	Bulanık Sınıflayıcılar		98,79

Literatürdeki diyabet çalışmalarının büyük çoğunluğu PIMA Indian veri seti üzerinden yürütülmüştür. Bu veri seti üzerindeki performans göstergeleri %73,83 ile %96,00 aralığında değişim göstermektedir. Farklı kaynaklardan toplanan veri

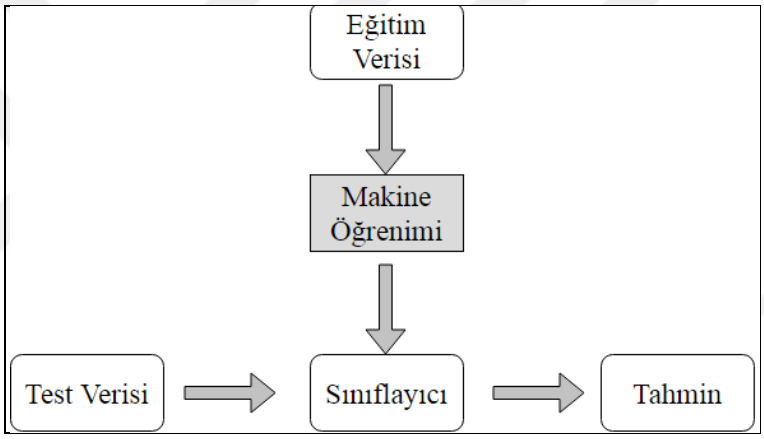
setlerinde yürütülmüş olan diyabet tahmin ve sınıflandırma problemlerin deperformans göstergeleri daha yüksek seyretmektedir.

4. YÖNTEM

Bu başlık altında tez çalışması kapsamında kullanılan temel makine algoritmalarından bahsedilmiştir. Önerilen melez modeller de bu bölümde anlatılmış ve melez modellerin kurulumunda kullanılan optimizasyon modeli olan Genetik Algoritmaya dair genel bilgiler aktarılmıştır.

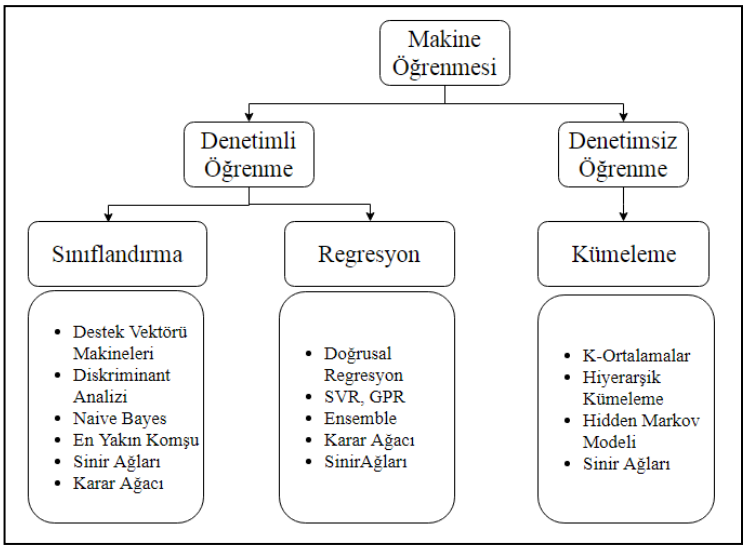
4.1. Temel Makine Öğrenmesi Algoritmaları

Makine öğrenmesi disiplini ilk olarak 1950'li yıllarda veri madenciliğine ait sayısal yorumlama ve model tanıma gibi problemlere yaklaşımlardan esinlenerek ortaya çıkmıştır. Makine öğrenmesi, eğitilebilen ve veriler üzerindeki örüntüyü analiz edebilen algoritmaların çalışma prensibini araştıran bir alt daldır (Ünal, 2015) (Şekil 4.1).



Şekil 4.1. Makine öğrenimi genel işleyişi

Bölüm 2'de bahsedildiği gibi makine öğrenmesi yöntemleri uygulanış şekline göre Denetimli ve Denetimsiz öğrenme olmak üzere ikiye ayrılırlar. Her iki öğrenme yöntemine ait temel makine öğrenmesi algoritmaları Şekil 4.2'daki gösterilmektedir.



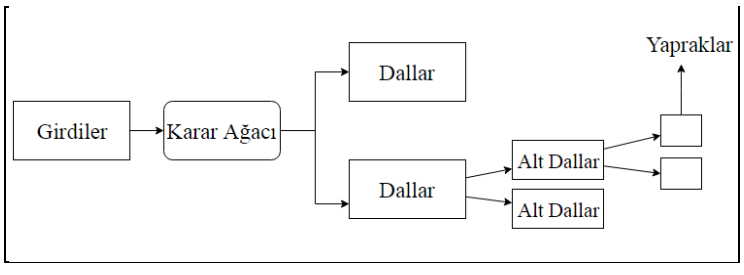
Şekil 4.2. Öğrenme türlerine göre makine öğrenimi algoritmaları

Denetimli öğrenmede eğitim verisi kullanılarak girdi ve çıktı verisine dayalı olarak veriler gruplandırılır. Bu tez çalışmasında modelin eğitilmesi süreci de sürece dahil edildiği için denetimli öğrenme tekniklerinden faydalanılmıştır.

Bölüm 3.2'de sınıflama amacıyla en çok kullanılan 4 makine öğretmesi algoritması Destek Vektörü Makineleri, Karar Ağacı, Yapay Sinir Ağları ve Naive Bayes sınıflandırıcısı olarak tespit edilmişti. Tez çalışmasında bu yöntemlerin karşılaştırma amacıyla kullanılması kararlaştırıldığından bu bölümde yalnızca bu yöntemler açıklanmıştır.

4.1.1. Karar ağaçları

Karar ağacı, öğrenme aşamasında, edindiği bilgiyi bir ağaç üzerine modelleyerek ilerleyen bir algoritmadır. Karar ağacında Şekil 4.3'de görüldüğü gibi yapraklar sınıf etiketlerini temsil eden bir ağaç yapısı oluşturulmaktadır.



Şekil 4.3. Karar ağacı çalışma mantığı

Ağacın öğrenme aşaması sırasında, üzerinde eğitim yapılan veri seti çeşitli niteliklere göre alt veri gruplarına bölünür. Bu işlem, özyinelemeli olarak tekrarlanır

ve yinelenmelerin tahmin üzerindeki etkisi yeterince azalayıncaya kadar (özyinelemeli parçalama) devam eder. Genelde makine öğreniminde verinin modele gelme şekli (4.1) nolu eşitlikte gösterildiği gibidir.

$$(x,Y) = (x_1,x_2,x_3,x_4,..., Y) \tag{4.1}$$

Eşitlik (4.1)'de x_1'den x_n'e kadar olan değişkenler, sistemin girdi değişkenleri iken, Y değeri sistemin çıktı değişkeni olarak tanımlanmaktadır. Örneğin, diyabet hastası olup olmama durumu Y değeri olarak, "soy ağacında diyabetli olup olmaması", "yaşı", "alkol kullanımı", "yaşam tarzı" gibi bilgiler ise sistemin girdi değişkenleri olan x olarak tanımlanacaktır.

Karar ağaçları metodolojisinde sınıflandırma işlemi "bilgi kazanımı" ölçütüne göre işlenir. Herhangi bir satırı sınıflandırmak için gereken bilgi kazanımı (I), (4.2) nolu eşilikteki gibi hesaplanmaktadır.

$$I(s_1,s_2,...,s_m) = -\sum_{i=1}^{m} \frac{s_i}{s} \log_2 \frac{s_i}{s} \tag{4.2}$$

Burada *s* değeri niteliklere ait sınıfların sayılarını ifade etmektedir.

Bir değişkene ait bilgi kazanımının net bir şekilde hesaplanabilmesi için değişkenlerin kendi içerisindeki entropisinin (düzensizliğin) hesaba katılması gerekmektedir. Bir A değişkenin $\{a_1,a_2,\ldots,a_v\}$ değerleri ile düzensizliği Eşitlik (4.3)'teki formüle göre hesaplanmaktadır.

$$E(A) = \sum_{j=1}^{v} \frac{s_{1j} + ... + s_{mj}}{s} I(s_{1j},..., s_{mj}) \tag{4.3}$$

A değişkeni kullanılarak ağacın dallanmasıyla kazanılan bilgi ise nihai olarak (4.4) nolu eşitlikteki gibi hesaplanmaktadır.

$$Kazanç(A) = I(s_1, s_2,..., s_m) - E(A) \tag{4.4}$$

4.1.2. Naive bayes

Naive Bayes metodu her bir değişken için Bayes Teoremi'ne göre bir kestirim yapar ve bir değişkenin sınıf üyelik fonksiyonunu hesaplar. Bu nedenle Naive Bayes sınıflandırıcısının kullandığı formüller Bayes Teoremi'nden gelir. Çıkan olasılık değerlerine göre değişkenin hangi sınıfa ait olduğuna karar verilir.

Bayes Teoreminde değişkenlerin sınıflara aidiyetlik olasılıkları (4.5) nolu eşitlikte gösterildiği gibi hesaplanır.

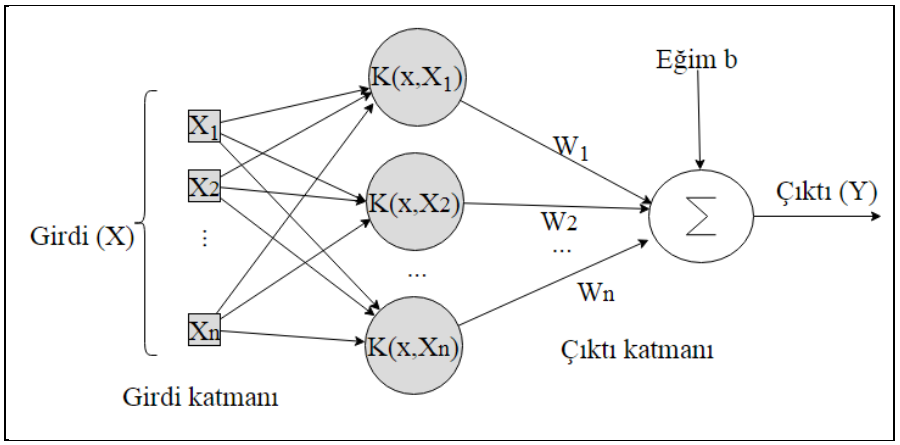
$$P(C_j|\, x) = \frac{P(x|C_j)P(C_J)}{P(x)} = \frac{P(x|C_j)P(C_j)}{\sum_k P(X|C_k)P(C_k)} \tag{4.5}$$

- $P(x/C_j)$: Sınıf j'ye ait bir nesnenin x olma olasılığı
- $P(C_j)$: Sınıf j'nin ilk olasılığı
- P(x) : Bir nesnenin x olma olasılığı
- $P(C_j/x)$= x olan bir örneğin sınıf j'den olma olasılığı olarak tanımlanmaktadır.

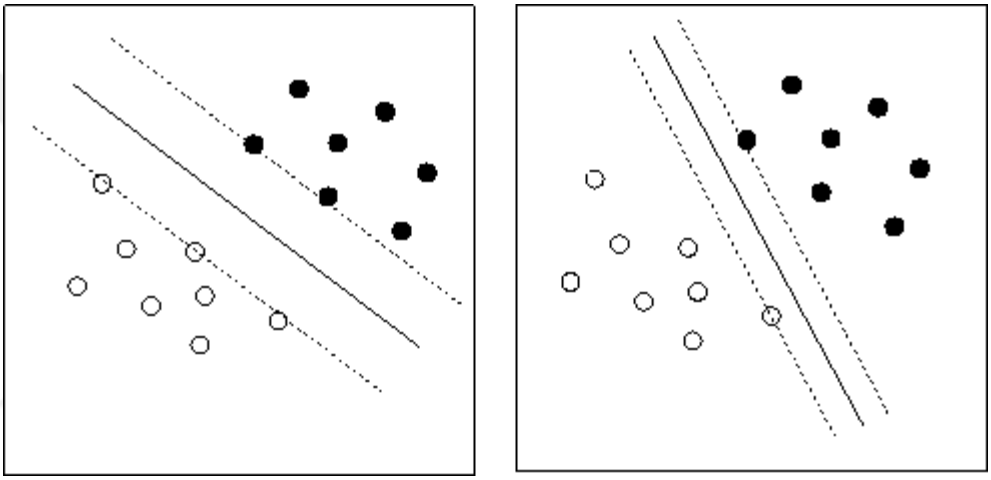
Eşitlik (5)'e göre hesaplanan olasılıkta en büyük değer aranır. Değişkenin en büyük sınıf aidiyetlik olasılığının olduğu sınıfa atama yapılır.

4.1.3. Destek vektörü makineleri

Destek Vektörü Makineleri (DVM) iki sınıflı doğrusal veya doğrusal olmayan verilerin sınıflandırılmasında kullanılan bir yöntemdir (Tong ve Koller, 2001). Çalışma prensibi iki sınıfı birbirinden ayıran en uygun karar fonksiyonunun (hiperdüzlemin) tahmin edilebilmesi ve hatanın enküçüklenebilmesidir. Destek Vektörü Makineleri ile ilgili genel süreç Şekil 4.4'de özetlenmiştir. DVM ara süreçlerinde Kernel fonksiyonlarını kullanılır ve nihai toplamında eğim değerleri (b) toplama eklenir. Marjin uzaklıklarına bakılarak sınıflandırma yapılır. Marjinlerin birbirinden uzak olması sınıflandırma performansını olumlu olarak etkilemektedir. Marjinler birbirinden ne kadar uzak ise sınıflandırma performansı o kadar o kadar iyi olmaktadır (Şekil 4.5).



Şekil 4.4. Destek vektörleri makineleri çalışma mantığı



Şekil 4.5. Marjinlerine göre nesnelerin sınıflandırılması

4.1.4. Yapay sinir ağları

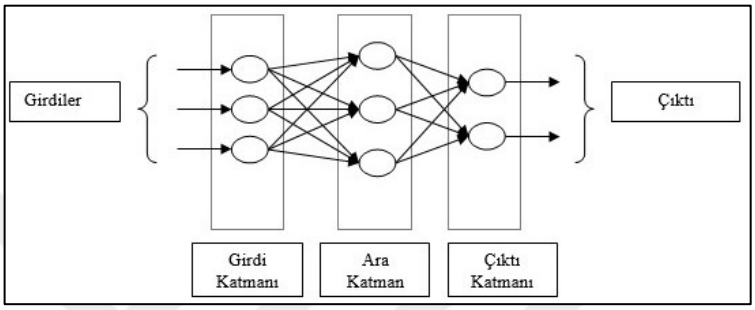
Yapay Sinir Ağı (YSA), insanların sinir sistemi mekanizmasından ve düşünebilme yeteneğinden esinlenerek ileri sürülmüş bir makine öğrenmesi tekniğidir. YSA'lar karmaşık öğrenme özelliğinden dolayı geleneksel makine öğrenmesi teknikleri için oldukça karmaşık ve çözümsüz kalan problemlere çözüm sağlayabilmektedir. Yine öğrenme yeteneği sayesinde, bilinen örnekleri kullanarak daha önce karşılaşılmamış durumlara genelleme yapabilmektedir.

Yapay Sinir Ağları, sadece sayısal verilerle çalışmaktadır. Geçmiş verileri kullanarak öğrenmekte ve gerçekleşmeyenolaylaradair bilgi üretebilmektedir. Bu özelliği sayesinde Yapay Sinir Ağları, mühendislik ve sağlık uygulamalarından, sanayi uygulamalarına pek çok alanda uygulanabilmektedir. Örneğin, sinir ağları yardımı ile bir sistemde veya cihazda olması muhtemel arızaların önceden

öngörülebilme olanağı olmaktadır. YSA'dan genellikle *tahmin*, *sınıflandırma* ve*veri kümeleme* işlemlerinde yararlanılmaktadır.

YSA'nın genel yapısı Şekil 4.6'da verilmektedir. Girdi verileri gizli nöronlara geçmeden önce ağırlıkları ile çarpılırlar ve daha sonra gizli nöronlarda belirleme fonksiyon işlemlerine tabi tutulurlar.

Bu işlemler sonucunda çıkan veriler bir sonraki katman için girdi verileri olur ve güncellenen ağırlıklar yardımıyla da çıktı verileri oluşturulur.



Şekil 4.6. Yapay sinir ağları çalışma mantığı

4.2. Melez Modeller

4.2.1. Genetik algoritmalı makine öğrenmesi

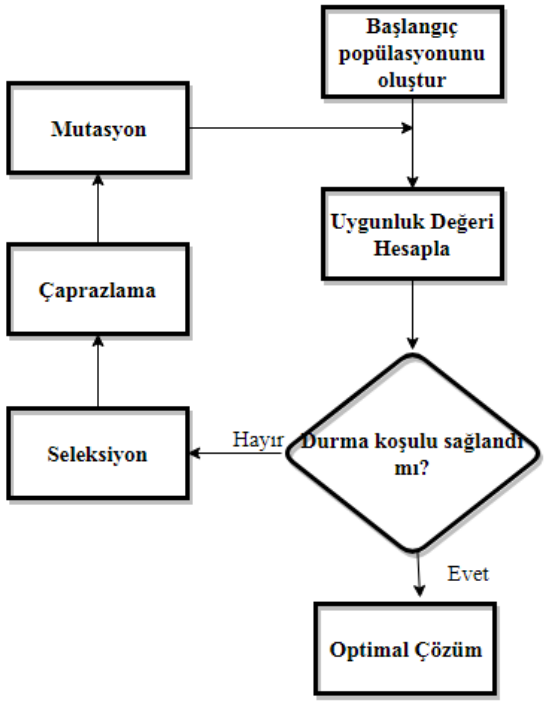
Genetik Algoritma, John Holland'ın 1975 yılında veri analizi üzerine yaptığı çalışmalarda doğadaki canlıların evriminden ve değişiminden esinlenerek, bilgisayar ortamına aktarmasıyla ortaya çıkmıştır. Genetik Algoritma'ya ilişkin temel kavramlar aşağıda açıklanmıştır.

- Gen: Bireylere ait genetik bilgileri taşıyan en küçük birimdir.
- Kromozom: Bir yada birden fazla genin bir araya gelmesiyle oluşurlar. Probleme ait tüm bilgileri içerirler.
- Popülasyon: Kromozomlar veya bireyler topluluğudur. Popülasyon üzerinde durulan problem için alternatif çözümler kümesidir.
- Seleksiyon: Uygunluk fonksiyon değeri daha güçlü olan bireylerin gelecek nesillere geçebilmesi için yapılan seçim işlemleridir.
- Çaprazlama: Ebeveyn kromozomun yerlerini değiştirerek çocuk kromozomlar üretmek ve böylelikle uygunluk değeri yüksek olan ebeveyn kromozomlardan daha yüksek uygunluklu çocuk kromozomlar üretmektir.

- Mutasyon: Kromozomların kendi genleri veya genleri oluşturan küçük birimleri üzerinde değişiklik yapılmasını sağlayan işlemcidir.

Genetik Algoritma, geleneksel veri madenciliği teknikleriyle çözülmesi güç olan, özellikle sınıflandırma ve karmaşık optimizasyon problemlerine, daha kolay ve hızlı çözüm sunmaktadır. Genetik algoritmalar doğada geçerli olan en iyinin yaşaması kuralına dayanarak sürekli iyileşen çözümler üretir. Bunun için "iyi"nin ne olduğunu belirleyen bir *uygunluk* fonksiyonu ve yeni çözümler üretmek için *yenidenkopyalama*, *değiştirme* gibi operatörleri kullanır (Kim vd, 2007).

Genetik algoritmalar makine öğrenmesine, canlıların evrimine dayanan yaklaşım sunmaktadır. Uygun çözümün aranmasına başlarken öncelikle rassal bir başlangıç popülasyonu üretmektedir. Mevcut popülâsyonun bireyleri rastgele mutasyon ve çaprazlama gibi işlemlerle bir sonraki nesli oluşturmaktadır. Her adımda hâlihazırdaki popülâsyondaki bireyler önceden verilmiş bir uygunluk değerine göre evrilmektedirler. Bu işlem en uygun sonuçların bir sonraki popülasyonu oluşturacak kromozomlar olarak seçilmesiyle olmaktadır. Genetik algoritmanın uygulanma süreci Şekil 4.7'de verilmiştir.



Şekil 4.7. Genetik algoritma işleyiş şeması

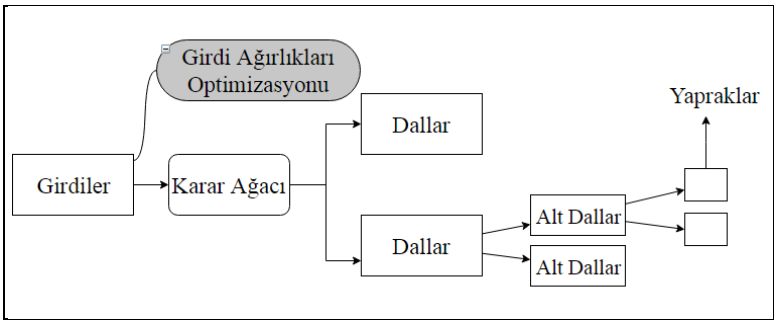
Genetik algoritmalar çeşitli öğrenme görevlerine ve diğer optimizasyon problemlerine başarıyla uygulanmıştır (Aziz vd, 2013).

Makine öğrenmesi çalışmalarından bir optimizasyon aracı olarak kullanılacak olan GA'nın performansını popülasyon büyüklüğü, mutasyon oranı, çaprazlama oranı gibi değerler etkilemektedir.

4.2.2. Genetik algoritma ile karar ağacı melezlenmesi

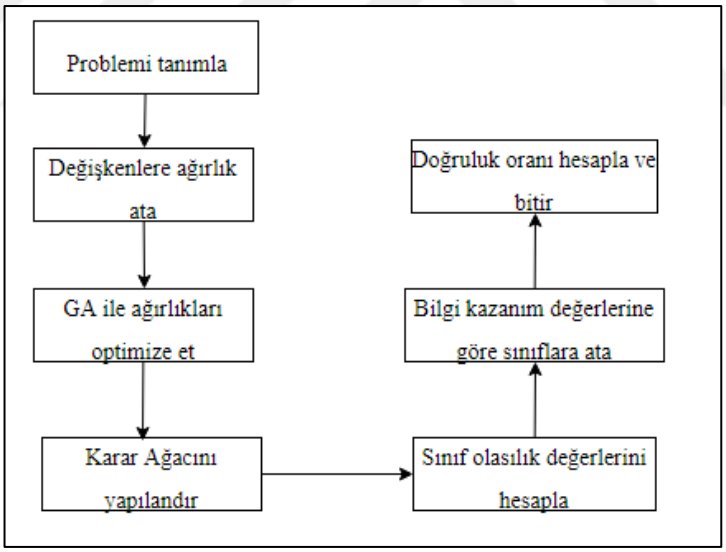
Karar ağaçlarında tüm nitelikler ağaç yapısına dâhil edilirse; test aşamasında bazı örnekler için yanlış sonuçlar çıkabilmektedir. Bu nedenle karar ağacı oluşturulurken her nitelik bir dala açılmak zorunda değildir. Aksi takdirde aşırı öğrenme denilen durum meydana gelir ve modelin tahmin ediciye sunduğu doğruluk oranı düşük olacaktır. Genetik Algoritmalı Karar Ağacı (GA-KA)'nda hesaplamalardaki belirginliği ve modelin sunacağı doğruluk oranını anlamlı bir şekilde artırabilmek için veriler işleme alınmadan önce ağırlıklandırılma yoluna gidilmektedir.

Genetik Algoritma ile karar ağacının oluşturulabilmesi için veriler en optimal ağırlık değerleriyle güncellenir ve karar ağacı modeline dahil edilir (Şekil 4.8).



Şekil 4.8. Genetik algoritma ile karar ağacı yapısı

Klasik karar ağacı modellerinde mevcut data direkt olarak dallanma işlemine alınmaktadır. Önerilen Genetik Algoritma ile Karar Ağacı modelinde ise girdi değişkenlere belirli bir ağırlık yüklenerek ağaca olan etkileri, modelin sunacağı doğruluk oranını optimize edecek şekilde türetilmektedir. Klasik bir Karar Ağacı algoritmasında tüm girdi değişkenler eşit ağırlığa sahiptir. Genetik Algoritma ile ağırlıklar türetildiğinde her girdinin çıktıya etkisi göz önünde bulundurulmakta ve elde edilen performans değeri artırılmaya çalışılmaktadır. Önerilen modelin genel adımları Şekil 4.9'daki akış diyagramında gösterilmiştir.



Şekil 4.9. GA-KA adımları

Veri setinde 8 girdi değişkeni bulunmaktadır. Her birine birer ağırlık atadığımızda $\{w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8\}$ şeklinde bir ağırlık vektör elde edilir. Genetik algoritma ile optimizasyon aşamasında; $\forall$ $w_i \in [0,1]$ olmak üzere $Doğruluk\ Oranı = \frac{Doğru\ Tahmin\ Sayısı}{Toplam\ Veri\ Sayısı}$ denkleminin maksimizasyonunun yapılması istenmektedir. Bu problemde Genetik Algoritmadaki amaç fonksiyonu direkt olarak

popülasyon bireyleri ile ilişkili değildir. Rassal olarak atanan ağırlık değerleri popülasyonu Karar Ağacı ile eğitilir ve doğruluk oranı hesaplanır, daha sonrasında bu doğruluk değerleri fonksiyon uygunluk oranı olarak değerlendirilerek gelecek nesiller üretilir. Model en iyi uygunluk değerinde değişme sabitlenene kadar devam eder.

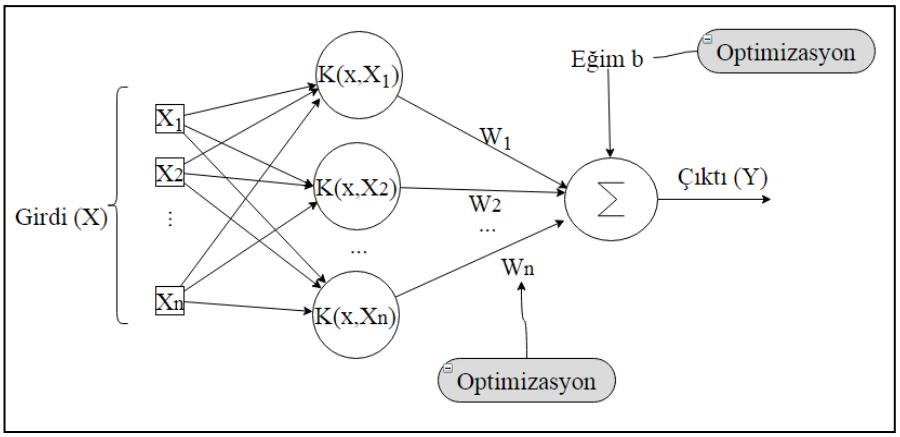
4.2.3. Genetik algoritma ile destek vektörü makineleri melezlenmesi

İkili sınıflandırma probleminde doğrusal olarak ayrılabilen bir veri setinin olduğu düşünülürse, bu veri setini ayırabilen sonsuz sayıda hiper-düzlem vardır. DVM karar yüzeyini oluştururken iki sınıfa olan uzaklığı maksimum (en fazla) yapmaya çalışır. Bu düzlemler arasında maksimum sınıra sahip sadece bir hiper-düzlem bulunmaktadır. Sınır genişliğini sınırlandıran noktalara ise destek vektörleri adı verilir. Destek vektör algoritması en büyük sınır genişliğine sahip ayırıcı hiperdüzlem ile sınıflandırma yaparak eğitim hatasını minimize etmeye çalışır. Bunu yaparken sınırı maksimum yapacak bir model üzerinden gidilir; eşitlik (4.6)' deki amaç fonksiyonunu, (4.7)' deki kısıtlar altında minimize etmeye denktir. (4.7) ifadesindeki w, ağırlık parametreleri vektörü; b ise öteleme değeridir.

Şekil 4.10'da görüldüğü üzere Destek Vektörü Makineleri algoritmasının girdi değişkenlerini eğiterek bir çıktıya ulaştırma aşamasında kullandığı iki farklı parametre vardır. Bu parametrelerden ilki niteliklere ait ağırlıklar vektörü iken, diğeri b ile gösterilen modelde bir eşik değer görevini gören eğim(bias) değeridir.

$$Z = \min \frac{1}{2} \|u\|^2 \tag{4.6}$$

$$y_i(w^T x_i + b) \geq 1 \tag{4.7}$$

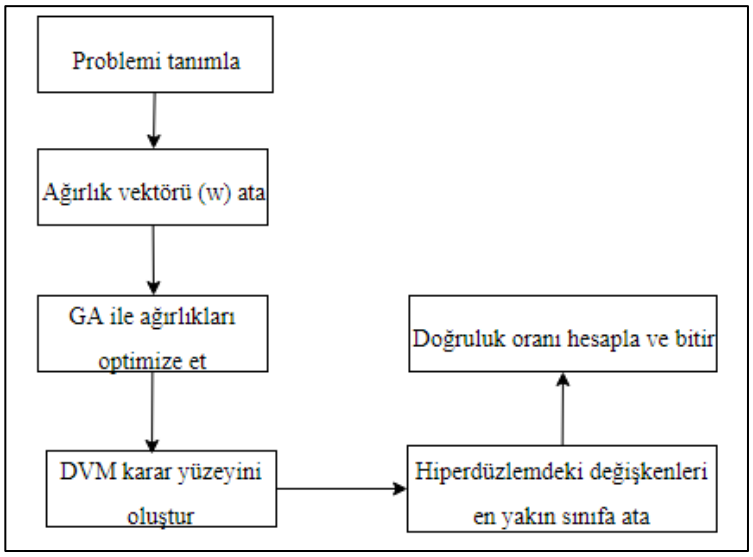


Şekil 4.10. Genetik algoritma ile destek vektörü makineleri yapısı

Çalışmada kullanılan veri setinde 8 adet girdi bulunmaktadır. Alt katmanda değişkenlerden çıkan ağırlıklar $\{w_i\}$ ve b olarak bir ağırlık vektörü ve eğim değeri elde edilir. Genetik algoritma ile optimizasyon aşamasında;

$\forall w_i \in [0,1]$ ve $b \in [-\infty,\infty]$ olmak üzere $Doğruluk\ Oranı$ denkleminin maksimizasyonunun yapılması istenmektedir. Diğer melez modellerde de olduğu gibi Genetik Algoritmadaki amaç fonksiyonu direkt olarak popülasyon bireyleri ile ilişkili değildir. Rassal olarak atanan ağırlık ve eğim değerleri popülasyonu Destek Vektörü Makineleri ile eğitilir ve doğruluk oranı hesaplanır, daha sonrasında bu doğruluk değerleri fonksiyon uygunluk oranı olarak değerlendirilerek gelecek nesiller üretilir. Model en iyi uygunluk değerinde değişme sabitlenene kadar devam eder.

Önerilen modelin genel adımları Şekil 4.11'deki akış diyagramında gösterilmiştir.



Şekil 4.11. GA-DVM adımları

4.2.4. Genetik algoritma ile naive bayes melezlenmesi

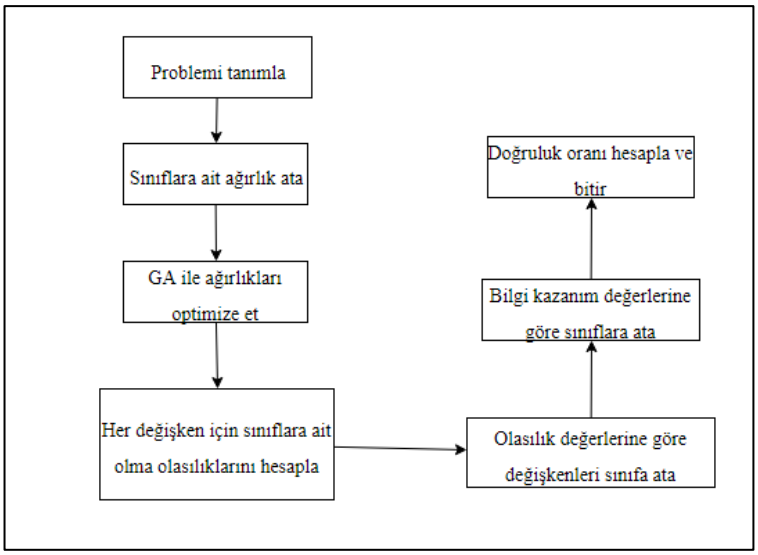
Temel Metodolojiler bölümünde değinildiği gibi, Naive Bayes metodu her bir değişken için Bayes Teoremi'ne göre bir kestirim yapar ve bir değişkenin sınıf üyelik fonksiyonunu hesaplar. Çıkan olasılık değerlerine göre değişkenin hangi sınıfa ait olduğuna karar verilir.

Bölüm 4.1.2'de de anlatıldığı üzere bir nesnenin hangi sınıfa ait olduğu Eşitlik (4.8)'e göre hesaplanmaktadır.

$$P(C_j|\,x) = \frac{P(x|C_j)P(C_J)}{P(x)} = \frac{P(x|C_j)P(C_j)}{\sum_k P(X|C_k)P(C_k)} \tag{4.8}$$

olarak hesaplanırdı ve bu olasılıkta en büyük değer aranırdı. Daha sonrasında ise değişkenin en büyük sınıf aidiyetlik olasılığının olduğu sınıfa atama yapılır.

Burada Genetik Algoritma ile Naive Bayes melez modelinde optimizasyon parametresi olarak $P(C_J)$ değerleri alınmıştır. Önerilen modelin genel adımları aşağıdaki akış diyagramında gösterilmiştir (Şekil 4.12).



Şekil 4.12. GA-NB adımları

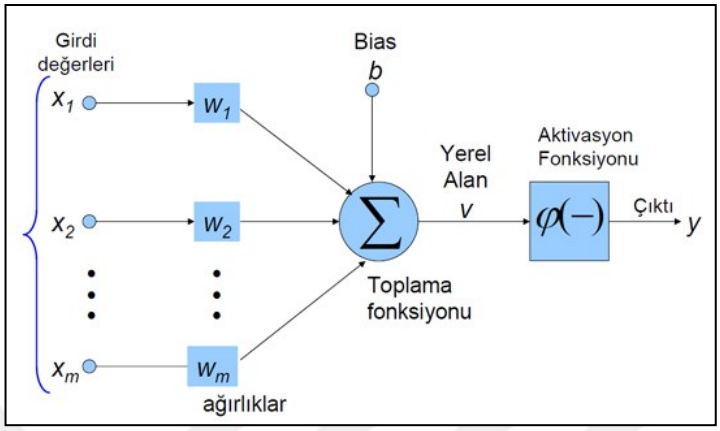
Geleneksel Naive Bayes yönteminde Eşitlik (4.8)'deki $P(C_J)$ değerleri veri setindeki sınıf sayıları oranı olarak alınmaktadır. Örneğin, 300 bireyin bulunduğu bir veri setinde diyabetli sayının 200, sağlıklı bireylerin sayısınında 100 olduğu durumda $P(C_0)$=0,33 ve $P(C_1) = 0{,}66$ olarak hesaba katılarak sınıflandırma yapılacaktır.

Genetik Algoritma ile Naive Bayes modelinde ise veri setinde sınıf sayısından bağımsız bir şekilde türetilen ve uygunluk değerini en büyükleyen sınıf olasılık değeri $P(C_J)$ önerilmiştir. Bu tez çalışmasındaki veri setinde iki adet sınıf bulunduğu için $\{w_{1,}w_2\}$ olarak bir ağırlık vektör elde edilir. Genetik algoritma ile optimizasyon aşamasında; $\{w_{1,}w_2\} \in [0,1]$ olmak üzere $Doğruluk\ Oranı$ denkleminin maksimizasyonunun yapılması istenmektedir. Diğer melez modellerde de olduğu gibi Genetik Algoritmadaki amaç fonksiyonu direkt olarak popülasyon bireyleri ile ilişkili değildir. Rassal olarak atanan sınıf ağırlıkdeğerleri popülasyonu Naive Bayes algoritması ile eğitilir ve doğruluk oranı hesaplanır, daha sonrasında bu doğruluk değerleri fonksiyon uygunluk oranı olarak değerlendirilerek gelecek nesiller üretilir. Model en iyi uygunluk değerinde değişme sabitlenene kadar devam eder.

4.2.5. Genetik algoritma ile yapay sinir ağları melezlenmesi

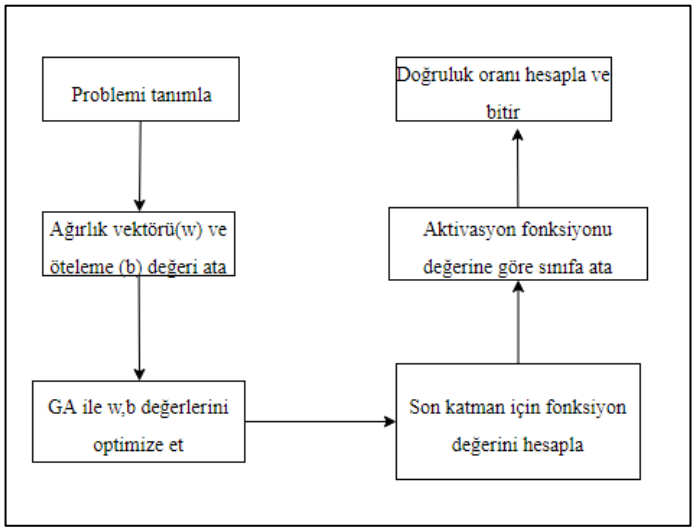
Sezgisel algoritmalar, yapay sinir ağının eğitilmesi (local minimum takılma) gibi komplex optimizasyon problemlerin çözümünde oldukça başarılı sınıflandırma performansı göstermektedir.

YSA'da girdi verileri gizli nöronlara geçmeden önce ağırlıkları ile çarpılırlar ve daha sonra gizli nöronlarda belirlenen fonksiyon işlemlerine tabi tutulurlar.Bu işlemler sonucunda çıkan veriler bir sonraki katman için girdi verileri olur ve güncellenen ağırlıklar yardımıyla da çıktı verileri oluşturulur (Şekil 4.13).



Şekil 4.13. Yapay sinir ağları yapısı

Genetik Algoritmalı Yapay Sinir Ağları girdi ağırlık parametreleri, sınıflandırma performansını en efektif kılacak şekilde Genetik Algoritma ile belirlenmektedir. Önerilen modelin genel adımları aşağıdaki akış diyagramında gösterilmiştir (Şekil 4.14).



Şekil 4.14. GA-YSA adımları

Ara katmanlardaki işlemler sonlandıktan sonra ağırlık parametreleri (4.9) nolu eşitlikteki toplama fonksiyonuna göre çıktı katmanına ilerler. Toplama fonksiyonu sonucuna göre değişkenlerin gerekli sınıfa ataması yapılır.

$$y_i = f(\sum w_{ji} x_i) \tag{4.9}$$

İleri beslemenin sonucunda oluşan hata kareler ortalaması (MSE) aktivasyon fonksiyonunun döndürdüğü değer olacaktır (Eşitlik 4.10 ve 4.11). Buradaki amaç her iterasyonda MSE değerini aşağı çekerek sıfıra yaklaştırmaktır.

$$E = \frac{1}{2} \sum (y_{dj} - y_j)^2 \tag{4.10}$$

$$MSE = \frac{E}{Eğitim\ Veri\ Seti} \tag{4.11}$$

y_{dj} , gerçekleşen değerleri belirtirken y_j değerleri ağın tahmin ettiği değerleri belirtmektedir.

Destek vektörü makinelerinde olduğu gibi yapay sinir ağlarında da ağırlıklar bir önceki katmandan çıkan değişkenlerden gelmektedir. Ancak yapay sinir ağlarında farklı olarak birden fazla katman kullanılmaktadır. Çalışmanın uygulamasında yapay sinir ağları 2 katman *(i)* ve her katmanda da 150 nöron *(j)* barındıracak şekilde kurulmuş ve katmanlardaki ağırlıklar $\{w_{ij,}\}$ ve eğim b şeklinde parametreler kullanılmıştır.

Genetik algoritma ile optimizasyon aşamasında; $\forall\ w_{ij} \in [0,1]$ ve $b \in [-\infty,\infty]$ olmak üzere, Eşitlik (4.11)'in minimizasyonun yapılması istenmektedir. Diğer melez modellerde de olduğu gibi Genetik Algoritmadaki amaç fonksiyonu direkt olarak popülasyon bireyleri ile ilişkili değildir. Rassal olarak atanan ağırlık ve eğim değerleri popülasyonu YSA ile eğitilir ve doğruluk oranı hesaplanır, daha sonrasında bu doğruluk değerleri fonksiyon uygunluk oranı olarak değerlendirilerek gelecek nesiller üretilir. Model en iyi uygunluk değerinde değişme sabitlenene kadar devam eder.

4.3. Makine Öğrenmesinde Performans Göstergeleri

Makine öğrenmesinde aşağıda sıralanan performans göstergeleri kullanılmaktadır.

- **Doğruluk oranı**, sınıflandırılmış örnek sayısının (TP +TN), toplam örnek sayısına (*TP*+*TN*+*FP*+*FN*) oranıdır. Hata oranı ise bu değerin 1'e tamlayanıdır. Diğer bir ifadeyle yanlış sınıflandırılmış örnek sayısının (*FP*+*FN*), toplam örnek sayısına (*TP*+*TN*+*FP*+*FN*) oranıdır (Eşitlik 4.12). Bu değerlerin Karışıklık Matrisinden elde edildiği Bölüm 2.5.3'de anlatılmıştır. Burada *TP*: doğru pozitif, *TN*: doğru negatif, *FP*: yanlış pozitif ve *FN*: yanlış negatif değerlerini göstermektedir.

$$\text{Doğruluk Oranı} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \tag{4.12}$$

- **Özgüllük (Ö)**, Sınıfı 0 olan değişkenin analiz sonucunda 0 olarak sınıflandırılması ihtimalidir (Eşitlik 4.13).

$$\text{Özgüllük} = \frac{TN}{FP+TN} \tag{4.13}$$

- **Duyarlılık (D)**, doğru sınıflandırılmış pozitif örnek (*TP*) sayısının, toplam pozitif örnek sayısına (*TP*+*FN*) oranıdır (Eşitlik 4.14).

$$\text{Duyarlılık} = \frac{(TP)}{(TP+FN)} \tag{4.14}$$

- **Pozitif Prediktif Değer:** Test pozitif olduğunda hastalığın varlığı olasılığı (Eşitlik 4.15).

$$PPD = \frac{TP}{TP+FP} \tag{4.15}$$

- **Negatif Prediktif Değer:** Testin negatif olması durumunda hastalığın bulunma ihtimalidir (Eşitlik 4.16).

$$NPD = \frac{TN}{FN+TN} \tag{4.16}$$

5. UYGULAMA

5.1. Veri Seti

Diyabet, pankreasın yeterli düzeyde insülin hormonu üretememesi ya da ürettiği insülin hormonunun sağlıklı bir şekilde kullanılamaması durumudur. Diyabet, ciddi komplikasyonlara neden olabilen ve daha ciddi sağlık problemlerini de beraberinde getiren metabolizma bozukluğudur. Kişi, yediği besinlerden kana geçen glikozu kullanamamakta ve kan şekeri değeri istenen düzeyde tutulamamaktadır.

HbA1c diyabetik hastanın takibinde değerli olup üç aylık kan şekeri ortalaması hakkında fikir vermektedir. Diyabetik hastalarda HbA1c için ideal düzey < % 6,5 olarak kabul edilmektedir. Diyabet, beraberinde körlük, kalp krizi ve felç gibi ciddi hasarlara yola açabiliyor olmasından dolayı erken diyabet teşhisi ve tedavisi oldukça önemlidir. Zamanında uygulanan tedavi ve iyi kan şekeri kontrolü ile oluşabilecek ciddi hasarların önüne geçebilinmektedir.

Bu tez çalışmasında bazı tıbbi ve demografik verileri alınan hastaların diyabetlilik durumuna ilişkin sınıflandırma problemi ele alınmıştır. UCI Machine Learning Repository veri tabanı sitesinden Pima Indian Diabetes başlıklı veriler kullanılmıştır. Veri seti, en küçüğü 21 yaşındaki kadın bireylerden oluşmaktadır (Lichman, 2013).

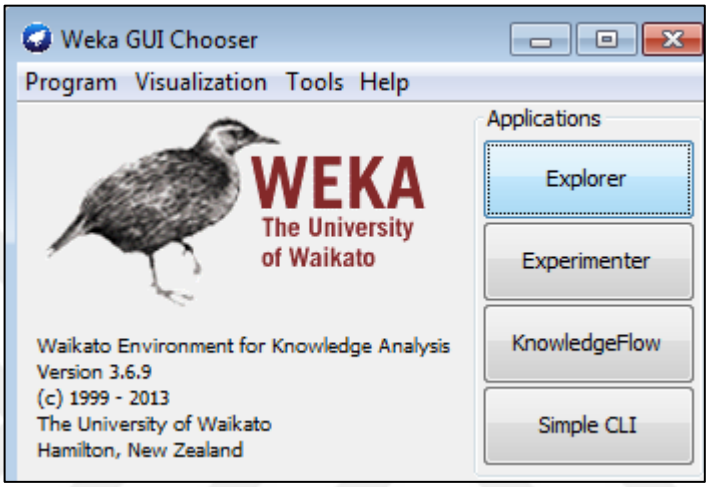
8 girdi değişkeni ve 1 çıktı değişkeni bulunmakta olup toplam 768 veri mevcuttur. Girdi değişkenleri Çizelge 5.1'de verildiği gibidir ve çıktı değişkeni bireyin diyabetli olup olmadığı durumudur.

Çizelge 5.1. Problemin girdileri

Girdi Değişkenleri	**Veri Tipi**
Gebelik Sayısı	Nümerik
2 Saatlik Glikoz Toleransı	Nümerik
Diyastolik Tansiyon	Nümerik
Triceps Deri Alt Kalınlığı	Nümerik
2 Saatlik Serum İnsülin Değeri	Nümerik
Vucüt Kitle İndeksi	Nümerik
Diyabetli Soy Ağacı	Nümerik
Yaş	Nümerik

5.2. Temel Algoritmaların Sonuçları

Weka (Waikato Environment for Knowledge Analysis) java platformu üzerine geliştirilmiş açık kaynak kodlu bir veri madenciliği programıdır (Frank, 2004). Yeni Zelanda'da bulunan Waikato Üniversitesi tarafından geliştirilmiştir. Bünyesinde hazır bir şekilde kullanılabilen 76 sınıflandırma ve 6 tane kümelele algoritması barındırmaktadır. Şekil 5.1'de bu tez çalışmasında kullanılan 3.6 sürümünün arayüzü görülmektedir.

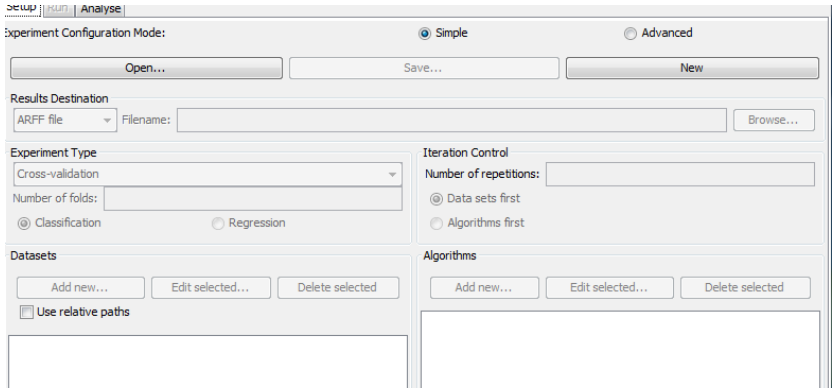


Şekil 5.1. WEKA arayüz

Explorer ile her bir makine öğrenmesi algoritmasının performans değerini analiz edilmek mümkün iken, Experimenter aracı ile birden fazla makine öğrenmesi algoritması eş zamanlı olarak çalıştırılabilmektedir.

Bu bölümde aynı veri seti üzerinde diyabet durumunu teşhis etmek ve belirlemek için kullanılan farklı sınıflandırma yöntemlerinin sonuçları verilmektedir. Destek Vektörü Makineleri, Naive Bayes Algoritması, Karar Ağacı ve Yapay Sinir Ağları Algoritması olmak üzere dört farklı sınıflandırma yöntemi sunulmuştur. Algoritmaların analizine geçmeden önce WEKA'da veri girişinin nasıl yapıldığı aşağıdaki açıklanmaktadır.

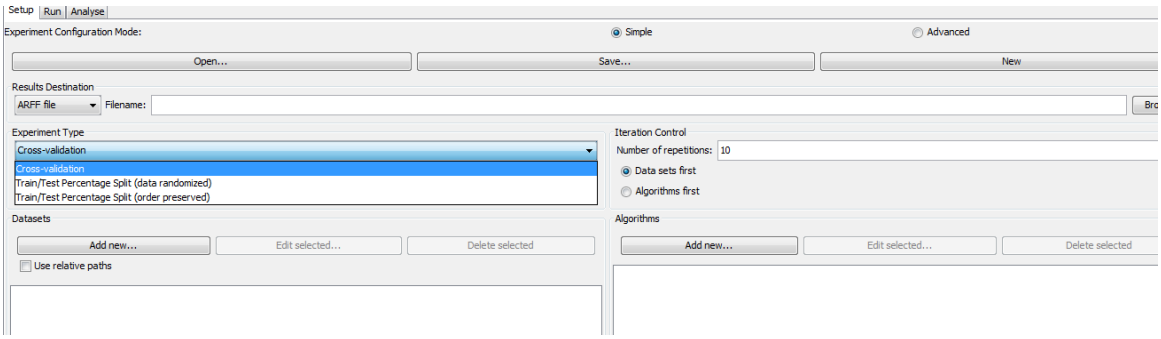
Şekil 5.2'de Experimenter aracının ekranı görülmektedir. Burada New tuşuna basılarak yeni model için bir sayfa açılır.



Şekil 5.2. WEKA kullanımı

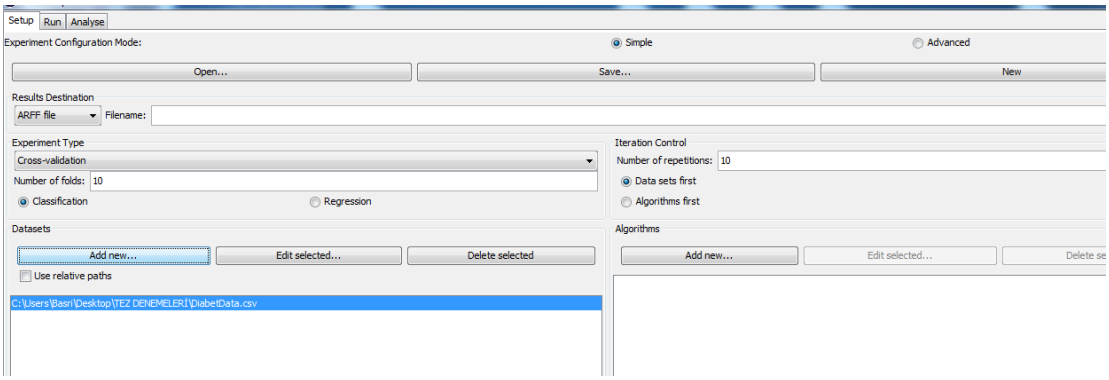
Experiment Configuration Mode olarak Simple düğmesi kullanılmış ve Experimenter aracının default özellikleri ile analiz gerçekleştirilmiştir.

Açılan yeni ekranda Experiment Type bölümünde istenilen doğrulama yöntemi seçilir (Şekil 5.3). Bu çalışmada Weka içerisinde yapılan analizlerde performans değeri ölçümünde 10 Katmanlı Çapraz Doğrulama yöntemi kullanılmıştır. 10 katmanlı çağraz doğrulama işleminde, veri seti rassal olarak 10 gruba ayrılır. 1. grup test için ayrılırken geriye kalan gruplarla algoritmalar kurgulandığı üzere çalıştırılır. Çalıştırılan model test için ayrılan veriler üzerinde tahminlenir ve doğruluk oranı hesaplanır. Süreç 10 defa tekrar eder ve modelin doğruluk oranı, 10 tane doğruluk oranının ortalaması kadar değer alır.



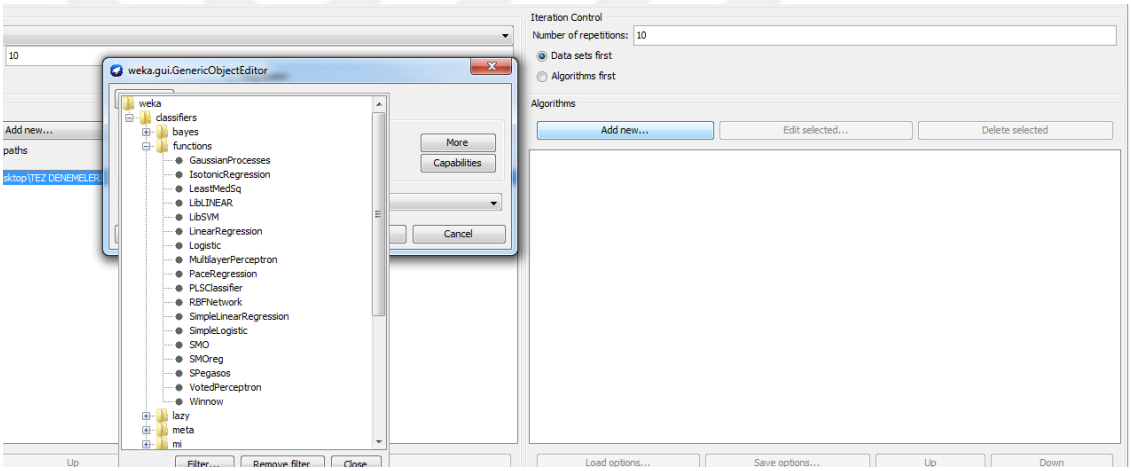
Şekil 5.3. WEKA experiment type

Doğrulama yöntemi seçildikten sonra Datasets bölümünde Add New butonuna tıklanarak veri setinin .csv uzantılı olarak dosyası yüklenir (Şekil 5.4).



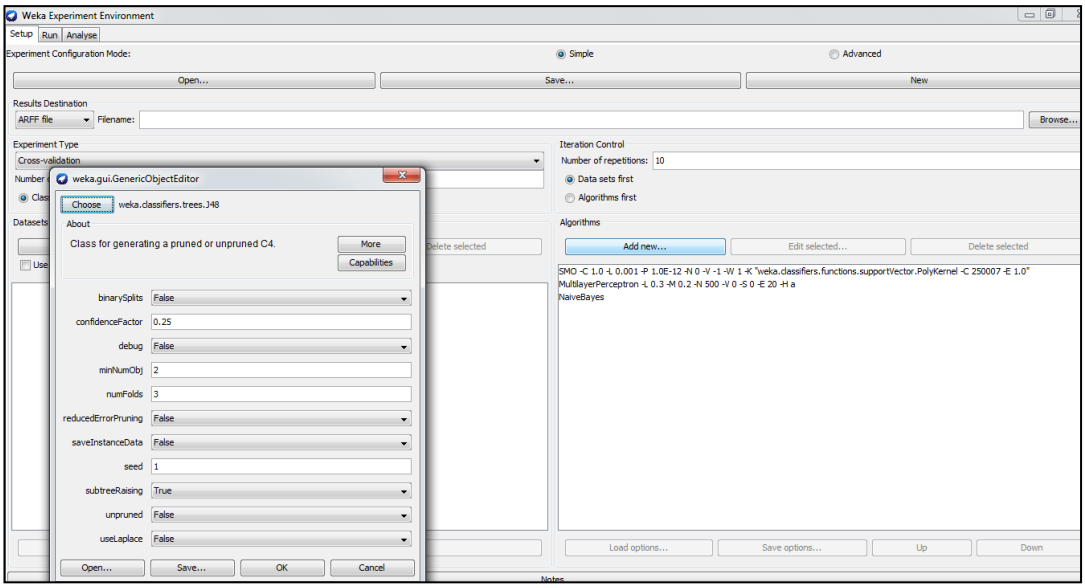
Şekil 5.4. WEKA veri seti girişi

Veri setinin yüklenmesinden sonra analizde kullanılmak istenen algoritma seçilmektedir. Algorithms bölümünden Add New butonundan istenilen algoritmalar yüklenmiştir. Şekil 5.5'de Destek Vektörü Makineleri ile Yapay Sinir Ağları algoritmalarının seçildiği görüntü gösterilmektedir.



Şekil 5.5. Algoritma girişi

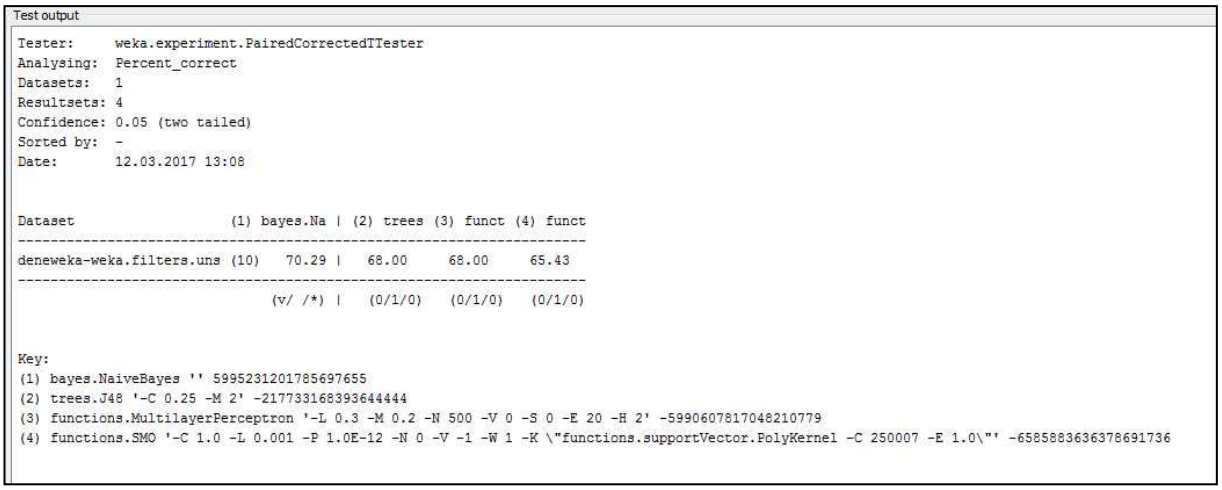
Tüm algoritmaların WEKA yazılımına yüklenmiş olduğu görünüm Şekil 5.6'da gösterilmektedir.



Şekil 5.6. WEKA'ya algoritmaların tanımlanması

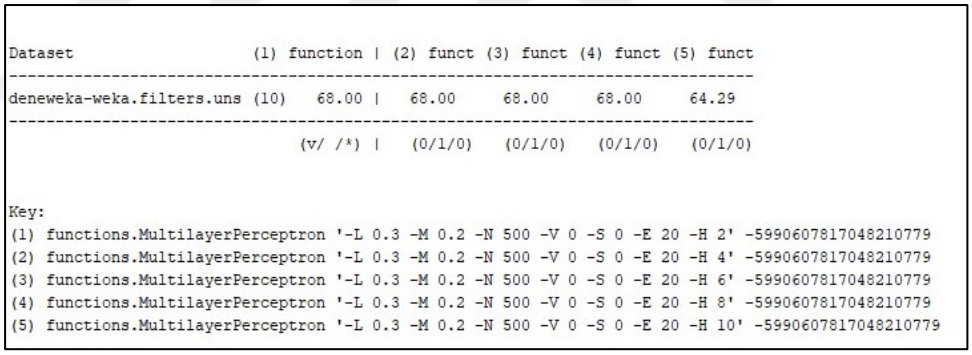
Şekil 5.7'de verilen performans değerlerine göre Naive Bayes %70,29 doğruluk yüzdesi ile en iyi tahmin etme performansını sergilemiştir. En kötü performansı Destek Vektörü Makineleri %65,43'lik bir doğruluk ile gerçekleştirirken Karar Ağacı ile Yapay Sinir Ağları Algoritmaları %68'lik gibi bir doğruluk ile aynı performansı gerçekleştirmiştir. Algoritmaların basit Weka 3.6 ile gerçekleştirilmiş olup analizin sonuç değerleri Şekil 5.7'de verilmiştir.

Şekil 5.7'deki (1) numaralı algoritma Naive Bayes, (2) numaralı algoritma Karar Ağacı, (3) numaralı algoritma Yapay Sinir Ağları ve son olarak (4) numaralı algoritma Destek Vektörü Makineleridir.



Şekil 5.7. WEKA'da yöntemlerin performans değeri

Aynı zamanda YSA için farklı katmanlarda koşumlar yapılmış ve katman sayısı arttıkça YSA'nın doğruluk oranının düştüğü görülmüştür. Şekil 5.8'de (1) numaralı algoritma 2 katmanlı, (2) numaralı algoritma 4 katmanlı, (3) numaralı algoritma 6 katmanlı, (4) numaralı algoritma 8 katmanlı ve (5) numaralı algoritma ise 10 katmanlı YSA'ları ifade etmektedir.



Şekil 5.8. YSA farklı katman sonuçları

5.3. Genetik Algoritma ile Melez Modellerin Sonuçları

Genetik algoritma ile melez model oluşturma sürecinde popülasyon büyüklüğü, çaprazlama oranı ve mutasyon oranı seçimi sınıflandırmanın doğruluk oranı üzerinde doğrudan etkilidir. Bu aşamada doğruluk oranının en yüksek olduğu kombinasyonu bulabilmek amacıyla, 3 faktörlü 2 seviyeli (2^3) faktöriyel deney tasarımından yararlanılmıştır. Belirlenen faktör seviyeleri Çizelge 5.2'de görülmektedir.

Çizelge 5.2. Faktör seviyeleri ve tasarım matrisi

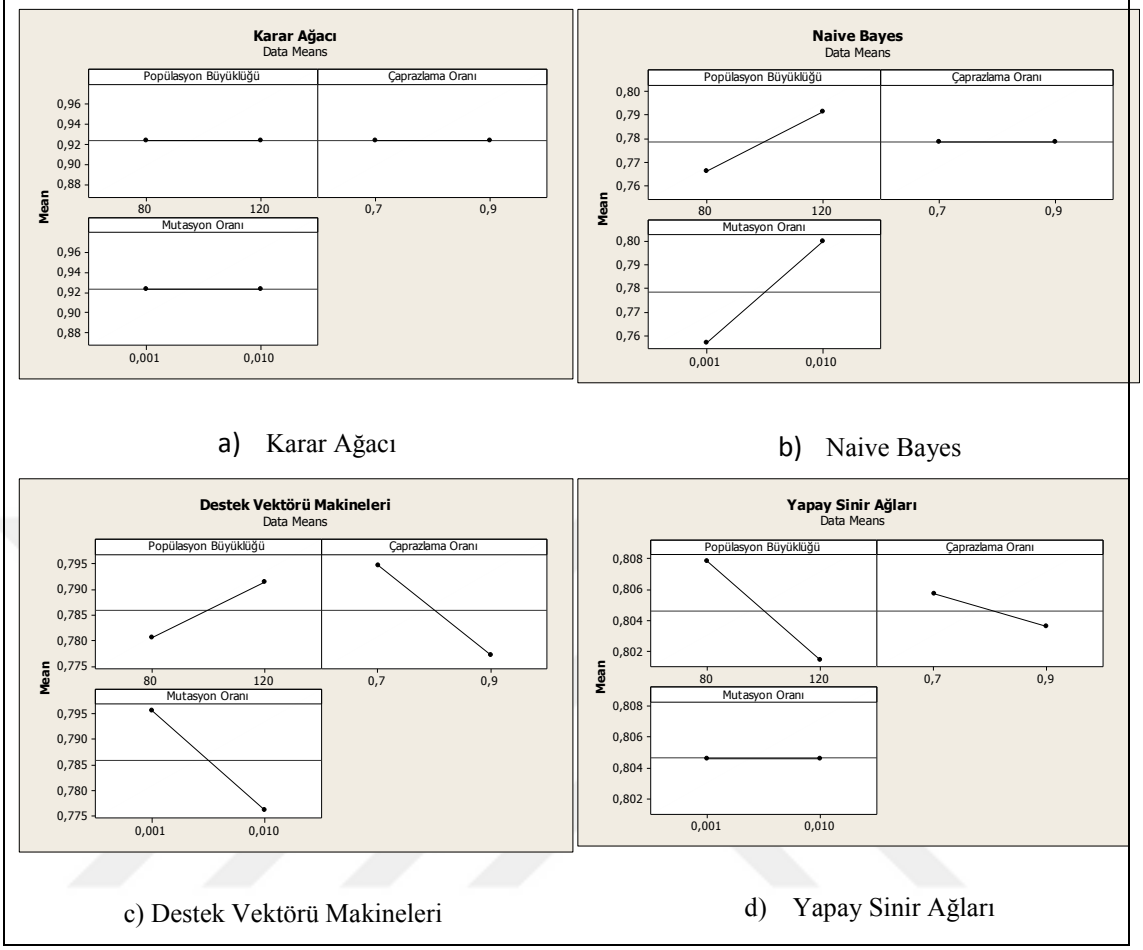
Faktör	Sembol	Faktör Seviyeleri	
		Düşük Seviye (-1)	Yüksek Seviye (+1)
Popülasyon Büyüklüğü	A	80	120
Çaprazlama Oranı	B	70	90
Mutasyon Oranı	C	0,1	1

Rassallık ilkesine uygun olarak yapılan denemelerde her bir kombinasyon 2 kez tekrar edilmiştir. MATLAB'da yapılan 16 deneme sonucu Çizelge 5.3'de verilmiştir. Her bir melez makine öğrenmesi algoritması için yazılmış MATLAB kodları Ek 1, 2, 3 ve 4'te verilmiştir. Melez modellerde genetik algoritma yapılandırılırken seçim yöntemi olarak rulet-çember seçimi yöntemi kullanılmıştır.

Çizelge 5.3. Deneme sonuçları

Deneme No	A	B	C	Ortalama Doğruluk Oranı(%)			
				Karar Ağacı	Naive Bayes	Destek Vektörü Makineleri	Yapay Sinir Ağları
1	80	70	0,1	92,28	78,26	73,57	79,71
2	120	70	0,1	92,28	82,17	75,71	82,71
3	80	90	0,1	92,57	82,17	78,57	80,42
4	120	90	0,1	92,28	75,65	75,00	79,00
5	80	70	1	92,28	76,95	78,57	82,14
6	120	70	1	92,57	80,43	83,57	77,71
7	80	90	1	92,28	74,78	75,71	80,85
8	120	90	1	92,28	78,26	82,14	81,14

Veri analizi aşamasında Minitab 17 istatistiksel yazılımından yararlanılmıştır. Şekil 5.9'da her bir genetik algoritma faktörünün cevap değişkeni (doğruluk oranı) üzerindeki bağımsız etkisini gösteren ana etki grafikleri verilmiştir. Bir faktörün ana etkisi; faktör yüksek seviyede ve düşük seviyede iken hesaplanan ortalama cevap değişkenleri arasındaki farktır. Ana etki grafiğinde, bir faktörün seviye değişimlerinin cevap değişkeni üzerinde yaratacağı fark ne kadar büyük ise, seviyeleri birleştiren çizgi o kadar dik olacaktır.



a) Karar Ağacı

b) Naive Bayes

c) Destek Vektörü Makineleri

d) Yapay Sinir Ağları

Şekil 5.9. Ana etki grafikleri

Ana etki grafiklerine göre en yüksek doğruluk oranı veren genetik algoritma faktörleri [popülasyon büyüklüğü, çaprazlama oranı, mutasyon oranı] her bir makine öğrenmesi algoritması için Çizelge 5.4'de özetlenmiştir.

Çizelge 5.4. Genetik algoritma

Makine Öğrenmesi Algoritması	[pb, ço, mo]	Doğruluk Oranı
Karar Ağacı	[80;90;0,1]	92,57
Naive Bayes	[120;70;1]	80,43
Destek Vektörü Makineleri	[120;70;0,1]	75,71
Yapay Sinir Ağları	[80;70;1]	82,14

pb: Popülasyon Büyüklüğü, ço: Çaprazlama Oranı, mo: Mutasyon Oranı

5.3.1. Genetik algoritma ile karar ağacı

Karar Ağacı algoritmasını veri seti üzerinde kullanmadan önce veri setinde bulunan değişkenlere ağırlıklar atanmış ve bu ağırlıklar sonucu en iyileyecek şekilde genetik algoritma ile optimize edilmiştir.

Önerilen model, geleneksel karar ağacı modeline göre daha iyi sonuç vermiş olup her iki durumda elde edilen performans değerleri Çizelge 5.5'de sunulmuştur.

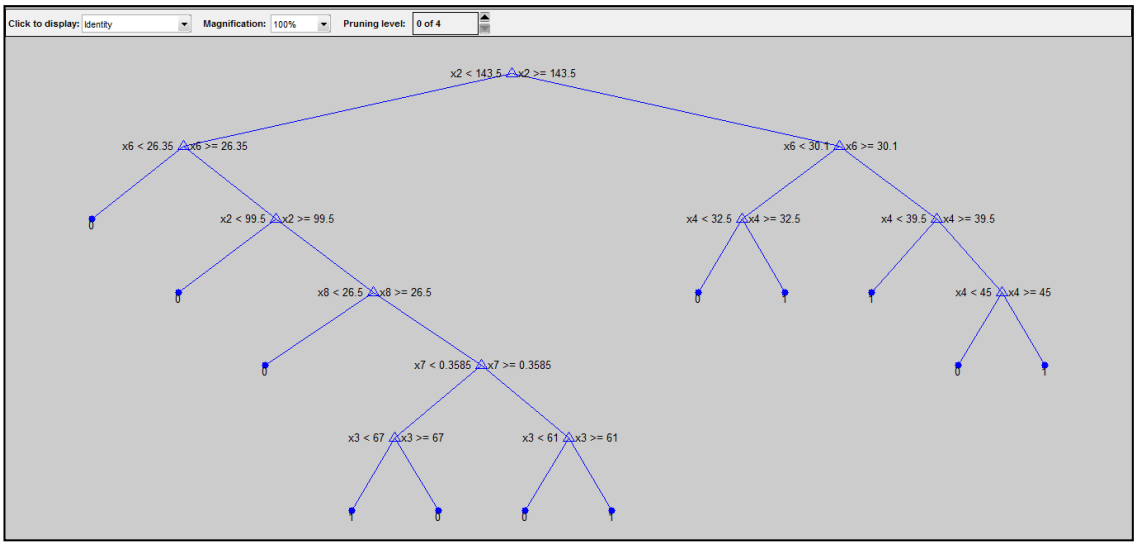
Çizelge 5.5. GA-KA doğruluk oranı

Yöntem	**Doğruluk Oranları**
Karar Ağacı	68,00
GA-KA	92,57

GA-KA: Genetik Algoritmalı Karar Ağacı

Burada bahsedilen melezleme işlemi KA algoritmasının uygulanmasına başlanmadan önce, verilerin optimum sonucu verecek şekilde düzenlemesinden oluşmaktadır.

Probleme ilişkin Matlab 2011 ile oluşturulan KA modeli Şekil 5.10'da gösterilmiştir. Şekil 5.10'da en sol çizgileri takip ettiğimizde 2 numaralı nitelik değerinin 143,5'den küçük olan bireylerde aynı zamanda 6 numaralı nitelik değeri 26,35'den küçükse bu birey sağlıklı olarak sınıflandırılmaktadır. Ancak 6 numaralı nitelik değeri 26,35'den büyük ve 2 numaralı nitelik değeri 99,5'den küçükse yine birey, sağlıklı olarak sınıflandırılacaktır. En sağ çiziden gidildiği durumda 6 numaralı nitelik değeri 30,1'den küçük ve 4 numaralı nitelik değeri 32,5'den büyükse birey diyabetli olarak sınıflandırılmaktadır. Karar ağacı yapısı bu şekilde yorumlanabilmektedir. Niteliklere dair bilgiler Çizelge 5.6'da gösterilmektedir.



Şekil 5.10. MATLAB karar ağacı

Çizelge 5.6. Değişken kısaltmaları

Girdi Değişkenleri	**Veri Tipi**	**Kodu**
Gebelik Sayısı	Nümerik	X1
2 Saatlik Glikoz Toleransı	Nümerik	X2
Diyastolik Tansiyon	Nümerik	X3
Triceps Deri Alt Kalınlığı	Nümerik	X4
2 Saatlik Serum İnsülin Değeri	Nümerik	X5
Vucüt Kitle İndeksi	Nümerik	X6
Diyabetli Soy Ağacı	Nümerik	X7
Yaş	Nümerik	X8

5.3.2. Genetik algoritma ile naive bayes sonuçları

Geleneksel Naive Bayes algoritmalarında çoğunlukla $P(C_J)$ değeri sınıfların örnekler üzerindeki sıklık değerleri olarak alınmaktadır. Bir başka alternatif yaklaşımda ise sınıfların öncelik ağırlıklarının düzgün dağılıma uyduğu da varsayılmaktadır. Böyle durumlarda ise iki sınıflı problemlerde sınıfların ağırlık değerleri (0,5 ; 0,5) olarak alınmaktadır.

Genetik Algoritma ile Naive Bayes Algoritması modelinde sınıf öncelik ağırlıkları veri setindeki frekanslardan bağımsız olarak ve modelin performansını en büyükleyecek şekilde Genetik Algoritma ile türetilmiştir. Gerçek hayat problemlerinde de genel olarak sınıflar arasındaki ayrım homojen olmadığından problem çözümündeki etkileri sınıfların etkileri ve sonuçları gibi etkenler dikkate alınarak yapılmalıdır. GA-NB modelinin sonuçlarına bakıldığında, en iyi tahminleme

performansını sağlayan koşullarda Diyabetli olmama durumunun algoritma üzerinde etkisi 0,3214 olarak belirlenmişken, Diyabetli olma durumunun etkisi 0,6786 olarak belirlendiği görülmektedir. Matematiksel bir ifadeyle optimum koşullar bazında sınıf olasılıkları $P(C_J)$ = (0,3214 ; 0,6789) olarak belirlenmiştir.

Genetik Algoritma ile Naive Bayes melez algoritmasının bu problem üzerinde sunmuş olduğu en büyük avantaj tahminleme sonuçlarının Diyabetli olma durumuna karşı daha hassas sonuçlar vermesidir. Örneğin, sınıf etiketlerinin (0,3214 ; 0,6789) olduğu durumun sınıf etiketlerinin (0,5 ; 0,5) olduğu duruma göre bireylerin diyabetli olma olasılığı daha yüksek seyretmektedir ve bu da diyabetli bir bireyin sağlıklı olarak sınıflandırma oranını düşürmektedir. Diyabetli olma durumunun ilgili problemde en kötü koşul olduğu göz önüne alındığında bu sonuçlar geleneksel Naive Bayes algoritmalarının sunduğu tahmin sonuçlarına göre daha önleyici etkiye sahiptir.

Naive Bayes ile ilgili kurulan modellerin doğruluk oranlarının karşılaştırılması Çizelge 5.7'de gösterilmektedir.

Çizelge 5.7. GA-NB doğruluk oranı

Yöntem	**Doğruluk Oranları**
Naive Bayes	70,29
GA-NB	80,43

GA-NB: Genetik Algoritmalı Naive Bayes

Klasik Naive Bayes algoritması %70,29 oranında bir doğruluk sağlarken Genetik Algoritma ile melezlenen Naive Bayes algoritması %80,43 oranında bir doğruluk sağlamıştır.

5.3.3. Genetik algoritma ile destek vektörü makineleri sonuçları

Destek Vektörü Makineleri modelinin performansını en büyüklenecek düzeyde oluşturulabilmesi için bahsedilen iki parametre değerleri (ağırlıklar ve eğim) Genetik Algoritma ile optimize edilmiştir.

Destek Vektörü Makineleri algoritmasının girdi değişkenlerini eğiterek bir çıktıya ulaştırma aşamasında kullandığı iki farklı parametre vardır. Bu parametrelerden ilki niteliklere ait ağırlıklar vektörü iken, diğeri b ile gösterilen modelde bir eşik değer olan eğim(bias) değeridir. Genetik algoritmalı Destek

Vektörü Makinelerinde ise ağrılık vektörü ve eğim parametre değerleri optimize edilerek melez model elde edilmiştir.

DVM ile ilgili kurulan modellerin karşılaştırılması Çizelge 5.8'de gösterilmektedir.

Çizelge 5.8. GA-DVM doğruluk oranı

Yöntem	**Doğruluk Oranları**
Destek Vektörü Makineleri	65,43
GA-DVM	75,71

GA-DVM: Genetik Algoritmalı Destek Vektörü Makineleri

Klasik Destek Vektörü Makineleri algoritması %65,43 oranında bir doğruluk sağlarken Genetik Algoritma ile melezlenen Destek Vektörü Makineleri algoritması %75,71 oranında bir doğruluk sağlamıştır.

5.3.4. Genetik algoritma ile yapay sinir ağları sonuçları

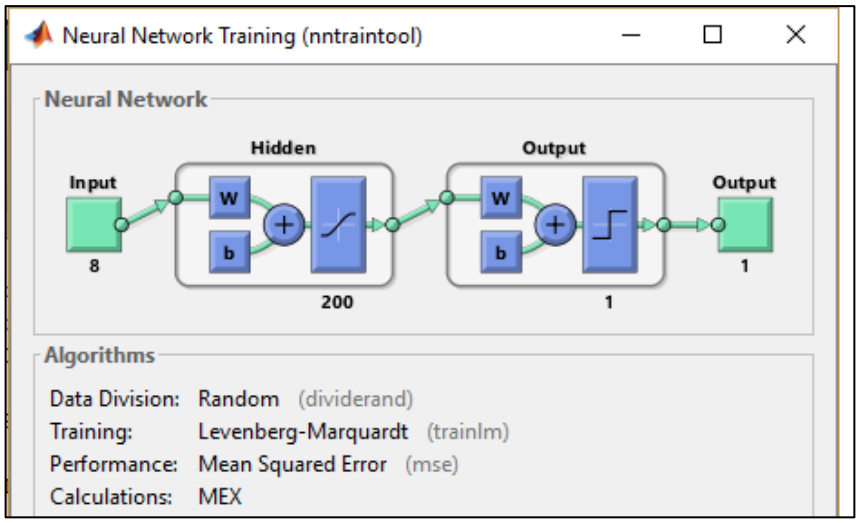
Yapay Sinir Ağları algoritmasının performansını en büyükleyecek şekilde oluşturulabilmesi için Bölüm 4.2.5'de bahsedilen iki parametre değerleri (ağırlıklar ve eğim) genetik algoritma ile optimize edilmiştir.

Yapay Sinir Ağları algoritmasının girdi değişkenlerini eğiterek bir çıktıya ulaştırma aşamasında kullandığı iki farklı parametre vardır. Bu parametrelerden ilki niteliklere ait ağırlıklar vektörü iken, diğeri b ile gösterilen modelde bir eşik değer olan eğim(bias) değeridir.

Yapay Sinir Ağları algoritmasında kullanılan model parametreleri Çizelge 5.9'da verilmiş ve Şekil 5.11'de problemdeki YSA modeli gösterilmiştir.

Çizelge 5.9. YSA parametreleri

Parametreler	**Optimum Değerler**
Gizli nöron sayısı	150
Gizli katman	2
Öğrenme metodu	Levenberg Marquardt
Öğrenme Oranı	0,25
Transfer Fonksiyonu	Hard Limit Transfer Fonksiyonu
Performans	Doğruluk Oranı



Şekil 5.11. Problemin MATLAB'te sinir ağları modeli

Geleneksel Yapay Sinir Ağları algoritmasının ve Genetik Algoritma ile Yapay Sinir Ağları melez algoritmasının sonucu Çizelge 5.10'da verilmektedir.

Çizelge 5.10. GA-YSA doğruluk oranı

Yöntem	**Doğruluk Oranları**
Yapay Sinir Ağları	68,00
GA-YSA	82,14

GA-KA: Genetik Algoritmalı Yapay Sinir Ağları

6. TARTIŞMA VE SONUÇ

Bu çalışmada diyabet teşhisi üzerine çeşitli algoritmalar önerilmiş ve test edilmiştir. Bölüm 3.2'de yapılan literatür çalışmasıyla Naive Bayes, Karar Ağaçları, Destek Vektörü Makineleri ve Yapay Sinir Ağları algoritmalarının en sık kullanılan algoritmalar olduğu tespit edilmiş ve bu algoritmalar Bölüm 5.2'de diyabet veri seti üzerinde sınıflandırma problemi için kullanılmıştır. Karşılaştırma yapabilmek açısından modellerin uygulanması için WEKA programı'nın Experimenter aracı kullanılmıştır. Programdan elde edilen sonuçlara göre diyabet durumu sınıflandırması hakkında Naive Bayes algoritmasının daha iyi sonuç verdiği anlaşılmıştır. En kötü performansı Destek Vektörü Makineleri sergilerken Yapay Sinir Ağları ile Karar Ağacı algoritmaları aynı performans değerini vermiştir. En iyi sınıflamayı yapan Naive Bayes algoritmasında bile doğruluk oranı 0,7029 düzeyinde kalmıştır. Bu nedenle bir optimizasyon algoritmasıyla melez model oluşturma ihtiyacı doğmuştur.

Bölüm 3.3'de yapılan literatür araştırmasında şu ana kadar uygulanmış olan makine öğrenmesi algoritmalarının en çok Genetik Algoritma ile melezlendiği tespit edilmiştir. Bu nedenle uygulaması yapılan 4 farklı temel makine öğrenmesi algoritmasının Genetik Algoritma ile melez modelleri önerilmiştir. Melez modellerin uygulanması aşamasında Matlab programından faydalanılmıştır. Kullanılan temel algoritmaların ve onların melez modellerinin elde edilen performans değerleri Çizelge 6.1'de sunulmaktadır.

Çizelge 6.1. Çalışmada kullanılan modellerin performans değerleri

	Doğruluk Oranı	**İyileştirme(%)**
KA	68,00	36,13
GA-KA	92,57	
NB	70,29	14,42
GA-NB	80,43	
DVM	65,43	15,71
GA-DVM	75,71	
YSA	68,00	20,79
GA-YSA	82,14	

En iyi iyileşme performansı Karar Ağacı ve Naive Bayes algoritmalarının melez modellerinden elde edilmiştir. Karar Ağacı ilk durumda %68 doğruluk oranı

sunarken GA ile melezlenmesi sonucu doğruluk oranı %92,57'e yükselmiş ve %36,13 oranın bir iyileştirme sağlanmıştır. KA'dan sonra en iyi iyileştirme Yapay Sinir Ağları algoritmasında elde edilmiştir. Yapay Sinir Ağları algoritmasında doğruluk oranı %68'den %82,14'e yükselmiş ve önerilen model ile %20,79 oranında iyileştirme sağlanmıştır. Genetik Algoritma ile Destek Vektörü Makineleri'nden %15,71, Genetik Algoritma ile Naive Bayes sınıflandırıcısından %14,42 oranında iyileştirme sağlanmıştır.

Aynı veri seti üzerinde yapılan diğer çalışmalara bakıldığında (Çizelge 3.3), Genetik Algoritmalı Karar Ağacı modeli literatürdeki en iyi sonucu vermese de oldukça iyi bir sınıflandırma performansı sergilemiştir.

Bu tez çalışması kapsamında geleneksel makine öğrenmesi algoritmalarının sınıflandırma problemlerindeki etkinliklerinin güçlendirilmesi için optimizasyon algoritmalarıyla birlikte kullanımı önerilmiştir. Elde edilen bilgi kadar o bilginin ne derece doğru bir kazanım sağladığı da oldukça önemli bir olgudur. Kullanılan geleneksel yöntemler uygunluk değeri daha sağlam olacak şekilde evrildiklerinde kullanılan yöntemlerin sağladıkları bilgiler de daha sağlam olacaktır.

Sağlık uygulamalarında daha sağlıklı kararlar vermek için hem doğru bilgiyi temin etmek hem de bilgiyi doğru ve faydalı çıkarımlara dönüştürmek gereklidir. Bu tez içerisinde hazır veri seti üzerinde diyabet tahmini yapılmış ve önerilen modelle iyi bir sınıflandırma performansı elde edilmiştir. İleriki çalışmalarda sadece aynı veri seti üzerinde kalmayıp önerilen algoritmaların farklı veri setlerinde ve farklı problemlerde performansı analiz edilecektir.

KAYNAKLAR

Abu-Nimeh, S., Nappa, D., Wang, X., Nair, S. (2007). A comparison of machine learning techniques for phishing detection. In Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit (pp. 60-69). ACM.

Acar, E., Özerdem, M. S., Akpolat, V. (2011). Diabetes Mellitus Forcast Using Various Types of Artificial Neural Networks. In 6th International Advanced Technologies Symposium (pp. 196-201).

Aci, M., İnan, C., Avci, M. (2010). A hybrid classification method of k nearest neighbor, Bayesian methods and genetic algorithm. Expert Systems with Applications, 37(7), 5061-5067.

Ahmed, K. A. W. S. A. R., Jesmin, T. A. S. N. U. B. A., Fatima, U. S. H. I. N., Moniruzzaman, M., Emran, A. A., Rahman, M. Z. (2012). Intelligent and effective diabetes risk prediction system using data mining. Orient J Comput Sci Technol, 5, 215-221.

Alby S., Shivakumar B. L. (2016). A PREDICTION MODEL FOR TYPE 2 DIABETES RISK AMONG INDIAN WOMEN, ARPN Journal of Engineering and Applied Sciences, 11 (3), 2037-2043.

Amasyalı, M. F. (2008). Yeni makine öğrenmesi metotları ve ilaç tasarımına uygulamaları, Doktora Tezi, Yıldız Teknik Üniversitesi.

Aslam, M. W., Zhu, Z., Nandi, A. K. (2013). Feature generation using genetic programming with comparative partner selection for diabetes classification. Expert Systems with Applications, 40(13), 5402-5412.

Aziz, A. S. A., Hassanien, A. E., Hanaf, S. E. O., Tolba, M. F. (2013). Multi-layer hybrid machine learning techniques for anomalies detection and classification approach. In Hybrid Intelligent Systems (HIS), 2013 13th International Conference on (pp. 215-220). IEEE.

Barakat, N., Bradley, A. P., Barakat, M. N. H. (2010). Intelligible support vector machines for diagnosis of diabetes mellitus. IEEE transactions on information technology in biomedicine, 14(4), 1114-1120.

Basari, A. S. H., Hussin, B., Ananta, I. G. P., Zeniarja, J. (2013). Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization. Procedia Engineering, 53, 453-462.

Bies, R. R., Muldoon, M. F., Pollock, B. G., Manuck, S., Smith, G., Sale, M. E. (2006). A genetic algorithm-based, hybrid machine learning approach to model selection. Journal of Pharmacokinetics and Pharmacodynamics, 33(2), 195-221.

Boyle, J. P., Thompson, T. J., Gregg, E. W., Barker, L. E., Williamson, D. F. (2010). Projection of the year 2050 burden of diabetes in the US adult population: dynamic modeling of incidence, mortality, and prediabetes prevalence. Population health metrics, 8(1), 29.

Burakgazi, Y., (2017). Identification of breast cancer sub-types by using machine learning techniques", Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi.

Byvatov, E., Fechner, U., Sadowski, J., Schneider, G. (2003). Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. Journal of chemical information and computer sciences, 43(6), 1882-1889.

Carvalho, D. R., Freitas, A. A. (2004). A hybrid decision tree/genetic algorithm method for data mining. Information Sciences, 163(1), 13-35.

Chen, H., Tan, C., Lin, Z., Wu, T. (2014). The diagnostics of diabetes mellitus based on ensemble modeling and hair/urine element level analysis. Computers in biology and medicine, 50, 70-75.

Chen, J., Huang, H., Tian, S., Qu, Y. (2009). Feature selection for text classification with Naïve Bayes. Expert Systems with Applications, 36(3), 5432-5435.

Chen, L. S., Cai, S. J. (2015). Neural-Network-Based Resampling Method for Detecting Diabetes Mellitus. Journal of Medical and Biological Engineering, 35(6), 824-832.

Cho, S. B., Won, H. H. (2003). Machine learning in DNA microarray analysis for cancer classification. In Proceedings of the First Asia-Pacific bioinformatics conference on Bioinformatics 2003-Volume 19 (pp. 189-198). Australian Computer Society, Inc.

Choudhry, R., Garg, K. (2008). A hybrid machine learning system for stock market forecasting. World Academy of Science, Engineering and Technology, 39(3), 315-318.

Cruz, J. A., Wishart, D. S. (2006). Applications of machine learning in cancer prediction and prognosis. Cancer informatics, 2, 59.

Çetin, M. (2009). Bir üretim işletmesinde veri madenciliği uygulaması, Doktora Tezi, Sakarya Üniversitesi.

Dai, W., Xue, G. R., Yang, Q., Yu, Y. (2007). Transferring naive bayes classifiers for text classification. In AAAI (Vol. 7, pp. 540-545).

Davis, L. (1990). Hybrid genetic algorithms for machine learning. In Machine Learning, IEE Colloquium on (pp. 9-1). IET.

Kelly Jr, J. D., Davis, L. (1991). A Hybrid Genetic Algorithm for Classification. In IJCAI (Vol. 91, pp. 645-650).

DeFries, R. S., Chan, J. C. W. (2000). Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. Remote Sensing of Environment, 74(3), 503-515.

Diwani, S. A., Sam, A. (2014). Diabetes Forecasting Using Supervised Learning Techniques. Advances in Computer Science: an International Journal, 3(5), 10-18.

Dogantekin, E., Dogantekin, A., Avci, D., Avci, L. (2010). An intelligent diagnosis system for diabetes on linear discriminant analysis and adaptive network based fuzzy inference system: LDA-ANFIS. Digital Signal Processing, 20(4), 1248-1255.

Doğrusöz, A. (2007). Makine öğrenmesi teknikleri ile metinlerin otomatik olarak sınıflandırılması, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi.

Drezet, P. M., Harrison, R. F. (2001). A new method for sparsity control in support vector classification and regression. Pattern Recognition, 34(1), 111

Drucker, H., Wu, D., Vapnik, V. N. (1999). Support vector machines for spam categorization. IEEE Transactions on Neural networks, 10(5), 1048-1054.

Erman, J., Mahanti, A., Arlitt, M. (2006). Qrp05-4: Internet traffic identification using machine learning. In Global Telecommunications Conference, 2006. GLOBECOM'06. IEEE (pp. 1-6). IEEE.

Figueiredo, E., Park, G., Farrar, C. R., Worden, K., Figueiras, J. (2011). Machine learning algorithms for damage detection under operational and environmental variability. Structural Health Monitoring, 10(6), 559-572.

Frank, E., Hall, M., Trigg, L., Holmes, G., Witten, I. H. (2004). Data mining in bioinformatics using Weka. Bioinformatics, 20(15), 2479-2481.

Georga, E. I., Protopappas, V. C., Ardigò, D., Marina, M., Zavaroni, I., Polyzos, D., Fotiadis, D. I. (2013). Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression. IEEE journal of biomedical and health informatics, 17(1), 71-81.

Guzella, T. S., Caminhas, W. M. (2009). A review of machine learning approaches to spam filtering. Expert Systems with Applications, 36(7), 10206-10222.

Han, J., Pei, J., Kamber, M. (2011). Data mining: concepts and techniques. Elsevier.

Heydari, M., Teimouri, M., Heshmati, Z., Alavinia, S. M. (2016). Comparison of various classification algorithms in the diagnosis of type 2 diabetes in Iran. International Journal of Diabetes in Developing Countries, 36(2), 167-173.

Huang, C. L., Dun, J. F. (2008). A distributed PSO–SVM hybrid system with feature selection and parameter optimization. Applied soft computing, 8(4), 1381-1391.

Huang, Y., McCullagh, P., Black, N., Harper, R. (2007). Feature selection and classification model construction on type 2 diabetic patients' data. Artificial intelligence in medicine, 41(3), 251.

Huerta, E. B., Duval, B., Hao, J. K. (2006). A hybrid GA/SVM approach for gene selection and classification of microarray data. In Workshops on Applications of Evolutionary Computation (pp. 34-44). Springer, Berlin, Heidelberg.

Jaafar, S. F. B., Ali, D. M. (2005). Diabetes mellitus forecast using artificial neural network (ANN). In Sensors and the International Conference on new Techniques in Pharmaceutical and Biomedical Research, 2005 Asian Conference on (pp. 135-139). IEEE.

Jhaldiyal T. ve Mishra P. K. (2014). Analysis and Prediction of Diabetes Mellitus Using PCA, REP and SVM, International Journal of Engineering and Technical Research (IJETR), 2 (8), 164-166.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. Machine learning: ECML-98, 137-142.

Kala, R., Shukla, A., Tiwari, R. (2009). Comparative analysis of intelligent hybrid systems for detection of PIMA indian diabetes. In Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on (pp. 947-952). IEEE.

Karahoca, A., Karahoca, D., Kara, A. (2009). Diagnosis of diabetes by using adaptive neuro fuzzy inference systems. In Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control, 2009. ICSCCW 2009. Fifth International Conference on (pp. 1-4). IEEE.

Karegowda, A. G., Manjunath, A. S., Jayaram, M. A. (2011). Application of genetic algorithm optimized neural network connection weights for medical diagnosis of pima Indians diabetes. International Journal on Soft Computing, 2(2), 15-23.

Kayaer, K., Yıldırım, T. (2003). Medical diagnosis on Pima Indian diabetes using general regression neural networks. In Proceedings of the international conference on artificial neural networks and neural information processing (ICANN/ICONIP) (pp. 181-184).

Kelly Jr, J. D., Davis, L. (1991). A Hybrid Genetic Algorithm for Classification. In IJCAI (Vol. 91, pp. 645-650).

Khan, N., Gaurav, D., Kandl, T. (2013). Performance evaluation of Levenberg-Marquardt technique in error reduction for diabetes condition classification. Procedia Computer Science, 18, 2629-2637.

Khandani, A. E., Kim, A. J., Lo, A. W. (2010). Consumer credit-risk models via machine-learning algorithms. Journal of Banking & Finance, 34(11), 2767-2787.

Kharrat, A., Gasmi, K., Messaoud, M. B., Benamrane, N., Abid, M. (2010). A hybrid approach for automatic classification of brain MRI using genetic algorithm and support vector machine. Leonardo Journal of Sciences, 17(1), 71-82.

Kim, H. J., Shin, K. S. (2007). A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets. Applied Soft Computing, 7(2), 569-576.

Kim, H., Howland, P., Park, H. (2005). Dimension reduction in text classification with support vector machines. Journal of Machine Learning Research, 6(Jan), 37-53.

Kiritchenko, S., Matwin, S. (2011). Email classification with co-training. In Proceedings of the 2011 Conference of the Center for Advanced Studies on Collaborative Research (pp. 301-312).

Kose, U., Guraksin, G. E., Deperlioglu, O. (2016). Cognitive development optimization algorithm based support vector machines for determining diabetes. BRAIN. Broad Research in Artificial Intelligence and Neuroscience, 7(1), 80-90.

Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. Computational and structural biotechnology journal, 13, 8-17.

Köse, İ. (2015). "İnteraktif makine öğrenmesi kullanılarak, aktör-metâ ilişkisinin analizi ile sağlık geri ödeme sistemlerinde proaktif suistimal tespiti modeli", Doktora Tezi, Gebze Teknik Üniversitesi.

Kumari, V. A., Chitra, R. (2013). Classification of diabetes disease using support vector machine. International Journal of Engineering Research and Applications, 3(2), 1797-1801.

Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Robles, V. (2006). Machine learning in bioinformatics. Briefings in bioinformatics, 86-112.

Lee, K. C., Han, I., Kwon, Y. (1996). Hybrid neural network models for bankruptcy predictions. Decision Support Systems, 18(1), 63-72.

Li, L., Jiang, W., Li, X., Moser, K. L., Guo, Z., Du, L., Rao, S. (2005). A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset. Genomics, 85(1), 16-23.

Li, S., Kwok, J. T., Zhu, H., Wang, Y. (2003). Texture classification using the support vector machines. Pattern recognition, 36(12), 2883-2893.

Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

Luangruangrong, W., Rodtook, A., Chimmanee, S. (2012). Study of Type 2 diabetes risk factors using neural network for Thai people and tuning neural network parameters. In Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on (pp. 991-996).

Luo, G. (2016). Automatically explaining machine learning prediction results: a demonstration on type 2 diabetes risk prediction, Health Information Sciences and Systems.

Mackinnon, M. J. Glick N. (1999). Applications: Data Mining and Knowledge Discovery in Databases- An Overview. Australian and New Zealand Journal of Statistics, 41(3), 255-275.

Mannini, A., Sabatini, A. M. (2010). Machine learning methods for classifying human physical activity from on-body accelerometers. Sensors, 10(2), 1154-1175.

Melgani, F., Bruzzone, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. IEEE Transactions on geoscience and remote sensing, 42(8), 1778-1790.

Min, S. H., Lee, J., Han, I. (2006). Hybrid genetic algorithms and support vector machines for bankruptcy prediction. Expert systems with applications, 31(3), 652-660.

Müller, K. R., Krauledat, M., Dornhege, G., Curio, G., Blankertz, B. (2004). Machine learning techniques for brain-computer interfaces.

Naghibi, S. A., Pourghasemi, H. R., Dixon, B. (2016). GIS-based groundwater potential mapping using boosted regression tree, classification and regression tree, and random forest machine learning models in Iran. Environmental monitoring and assessment, 188(1), 44.

Niknam, T., Amiri, B. (2010). An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. Applied Soft Computing, 10(1), 183-197.

Otukei, J. R., Blaschke, T. (2010). Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. International Journal of Applied Earth Observation and Geoinformation, 12, S27-S31.

Özdemir, A., Aslay, F. Y., Handan, Ç. A. M. (2009). Veri Tabanında Bilgi Keşfi Süreci: Gümüşhane Devlet Hastanesi Uygulaması. Sosyal Ekonomik Araştırmalar Dergisi, 1(20), 347-366.

Pal, M. (2005). Random forest classifier for remote sensing classification. International Journal of Remote Sensing, 26(1), 217-222.

Pal, M., Mather, P. M. (2005). Support vector machines for classification in remote sensing. International Journal of Remote Sensing, 26(5), 1007-1011.

Pan, Z. S., Chen, S. C., Hu, G. B., Zhang, D. Q. (2003). Hybrid neural network and C4. 5 for misuse detection. In Machine Learning and Cybernetics, 2003 International Conference on (Vol. 4, pp. 2463-2467).

Pang, B., Lee, L., Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10 (pp. 79-86). Association for Computational Linguistics.

Park, B., Bae, J. K. (2015). Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data. Expert Systems with Applications, 42(6), 2928-2934.

Pennacchiotti, M., Popescu, A. M. (2011). A Machine Learning Approach to Twitter User Classification. Icwsm, 11(1), 281-288.

Polat, K., Güneş, S. (2007). An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. Digital Signal Processing, 17(4), 702-710.

Qi, Y., Doermann, D., DeMenthon, D. (2001). Hybrid independent component analysis and support vector machine learning scheme for face detection. In Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on (Vol. 3, pp. 1481-1484).

Romero, C., Ventura, S., Espejo, P. G., Hervás, C. (2008). Data mining algorithms to classify students. In Educational Data Mining 2008.

Rosenthal A. D. (2004). Body fat distribution and risk of diabetes among Chinese women, International Journal of Obesity, 28, 594-599.

Sanidad, J. G., Bandala, A., Sanidad, B. G., Marfil, S. S. (2012). Prediction of Diabetes on Women using Decision Tree Algorithm.

Sarıman, G. (2011). Veri Madenciliğinde Kümeleme Teknikleri Üzerine Bir Çalışma: K-Means ve K-Medoids Kümeleme Algoritmalarının Karşılaştırılması. Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 15(3).

Sartakhti, J. S., Zangooei, M. H., Mozafari, K. (2012). Hepatitis disease diagnosis using a novel hybrid method based on support vector machine and simulated annealing (SVM-SA). Computer methods and programs in biomedicine, 108(2), 570-579.

Saxena K. (2014). Diagnosis of Diabetes Mellitus using K nearest Neighbor Algorithm, International Journal of Computer Science Trends and Technology (IJCST), 2(4), 36-43.

Sebastiani, F. (2002). Machine learning in automated text categorization. ACM computing surveys (CSUR), 34(1), 1-47.

Senthilkumar, D., Paulraj, S. (2013). Diabetes Disease Diagnosis Using Multivariate Adaptive Regression Splines, International Journal of Engineering and Technology, 5(5), 3922-3929.

Sharifi, A., Vosolipour, A., Sh, M. A., Teshnehlab, M. (2008). Hierarchical Takagi-Sugeno type fuzzy system for diabetes mellitus forecasting. In Machine Learning and Cybernetics, 2008 International Conference on (Vol. 3, pp. 1265-1270).

Shipp, M. A., Ross, K. N., Tamayo, P., Weng, A. P., Kutok, J. L., Aguiar, R. C., Ray, T. S. (2002). Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nature medicine, 8(1), 68-74.

Shon, T., Moon, J. (2007). A hybrid machine learning approach to network anomaly detection. Information Sciences, 177(18), 3799-3821.

Sivagaminathan, R. K., Ramakrishnan, S. (2007). A hybrid approach for feature subset selection using neural networks and ant colony optimization. Expert systems with applications, 33(1), 49-60.

Smith, J. W., Everhart, J. E., Dickson, W. C., Knowler, W. C., & Johannes, R. S. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In Proceedings of the Annual Symposium on Computer Application in Medical Care (p. 261). American Medical Informatics Association.

Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C. (2012). Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural networks, 32, 323-332.

Statnikov, A., Aliferis, C. F., Tsamardinos, I., Hardin, D., Levy, S. (2004). A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics, 21(5), 631-643.

Statnikov, A., Wang, L., Aliferis, C. F. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. BMC bioinformatics, 9(1), 319.

Suhaimi, N., Ismail, A. (2012). Comparing the Performance of Logistic Regression and Artificial Neural Networks Models: An Application to Type 2 Diabetes Mellitus.

Şahan, S., Polat, K., Kodaz, H., Güneş, S. (2007). A new hybrid method based on fuzzy-artificial immune system and k-nn algorithm for breast cancer diagnosis. Computers in Biology and Medicine, 37(3), 415-423.

Tahir, M. A., Bouridane, A., Kurugollu, F. (2007). Simultaneous feature selection and feature weighting using Hybrid Tabu Search/K-nearest neighbor classifier. Pattern Recognition Letters, 28(4), 438-446.

Taravat, A., Del Frate, F., Cornaro, C., Vergari, S. (2015). Neural networks and support vector machine algorithms for automatic cloud classification of whole-sky ground-based images. IEEE Geoscience and remote sensing letters, 12(3), 666-670.

Temurtas, H., Yumusak, N., Temurtas, F. (2009). A comparative study on diabetes disease diagnosis using neural networks. Expert Systems with applications, 36(4), 8610-8615.

Thangaraju, P., Deepa, B., Karthikeyan, T. (2014). Comparison of Data mining Techniques for Forecasting Diabetes Mellitus. International Journal of Advanced Research in Computer and Communication Engineering, 3(8).

Tong, S., Koller, D. (2001). Support vector machine active learning with applications to text classification. Journal of machine learning research, 2, 45-66.

Tsai, C. F., Wang, S. P. (2009). Stock price forecasting by hybrid machine learning techniques. In Proceedings of the International MultiConference of Engineers and Computer Scientists (Vol. 1, No. 755, p. 60).

Tuia, D., Ratle, F., Pacifici, F., Kanevski, M. F., Emery, W. J. (2009). Active learning methods for remote sensing image classification. IEEE Transactions on Geoscience and Remote Sensing, 47(7), 2218-2232.

Upadhyay A., Patel V. R. (2016). Comparative Study - Prediction of Diabetes and Heart Disease using Data Mining Approaches, International Journal of Engineering Technology, Management and Applied Sciences, 4(1), 70-76.

Utomo, C. P., Kardiana, A., Yuliwulandari, R. (2014). Breast cancer diagnosis using artificial neural networks with extreme learning techniques. International Journal of Advanced Research in Artificial Intelligence, 3(7), 10-14.

Ünal, Y. (2015). Makine öğrenmesi yöntemleriyle bel bölgesi rahatsızlıklarının tanısı, Doktora Tezi, Selçuk Üniversitesi.

Wang, Y., Tetko, I. V., Hall, M. A., Frank, E., Facius, A., Mayer, K. F., Mewes, H. W. (2005). Gene selection from microarray data for cancer classification—a machine learning approach. Computational biology and chemistry, 29(1), 37-46.

Wei, L., Yang, Y., Nishikawa, R. M., Jiang, Y. (2005). A study on several machine-learning methods for classification of malignant and benign clustered microcalcifications. IEEE transactions on medical imaging, 24(3), 371-380.

Williams, N., Zander, S., Armitage, G. (2006). A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification. ACM SIGCOMM Computer Communication Review, 36(5), 5-16.

Yang, F., Sun, T., Zhang, C. (2009). An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization. Expert Systems with Applications, 36(6), 9847-9852.

Yang, H., Liu, J., Sui, J., Pearlson, G., Calhoun, V. D. (2010). A hybrid machine learning method for fusing fMRI and genetic data: combining both improves classification of schizophrenia. Frontiers in human neuroscience, 4.

Ye, Q., Zhang, Z., Law, R. (2009). Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert systems with applications, 36(3), 6527-6535.

Zander, S., Nguyen, T., Armitage, G. (2005). Automated traffic classification and application identification using machine learning. In Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on (pp. 250-257).

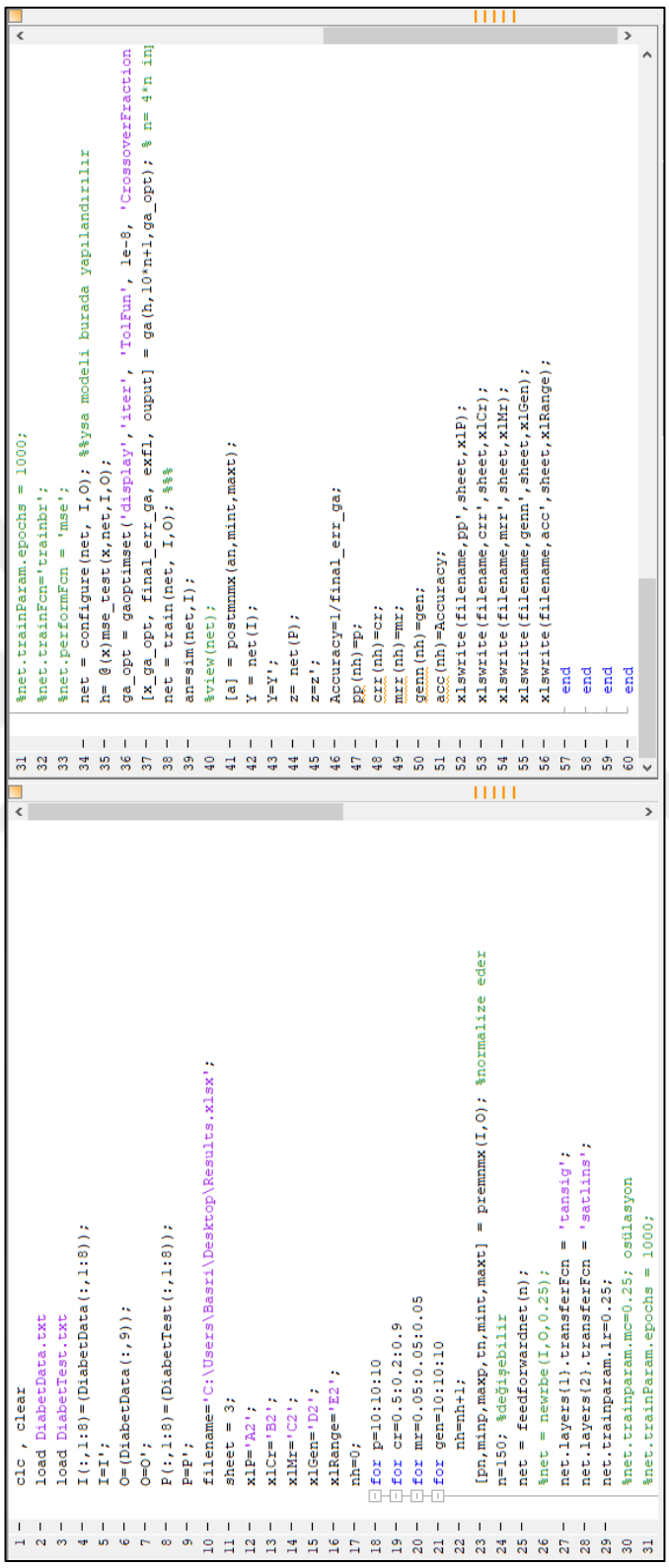
Zhang, D., Lee, W. S. (2003). Question classification using support vector machines. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (pp. 26-32).

Zhang, M. L., Zhou, Z. H. (2005). A k-nearest neighbor based algorithm for multi-label classification. In Granular Computing, 2005 IEEE International Conference on (Vol. 2, pp. 718-721).

Zheng, B., Yoon, S. W., Lam, S. S. (2014). Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms. Expert Systems with Applications, 41(4), 1476-1482.

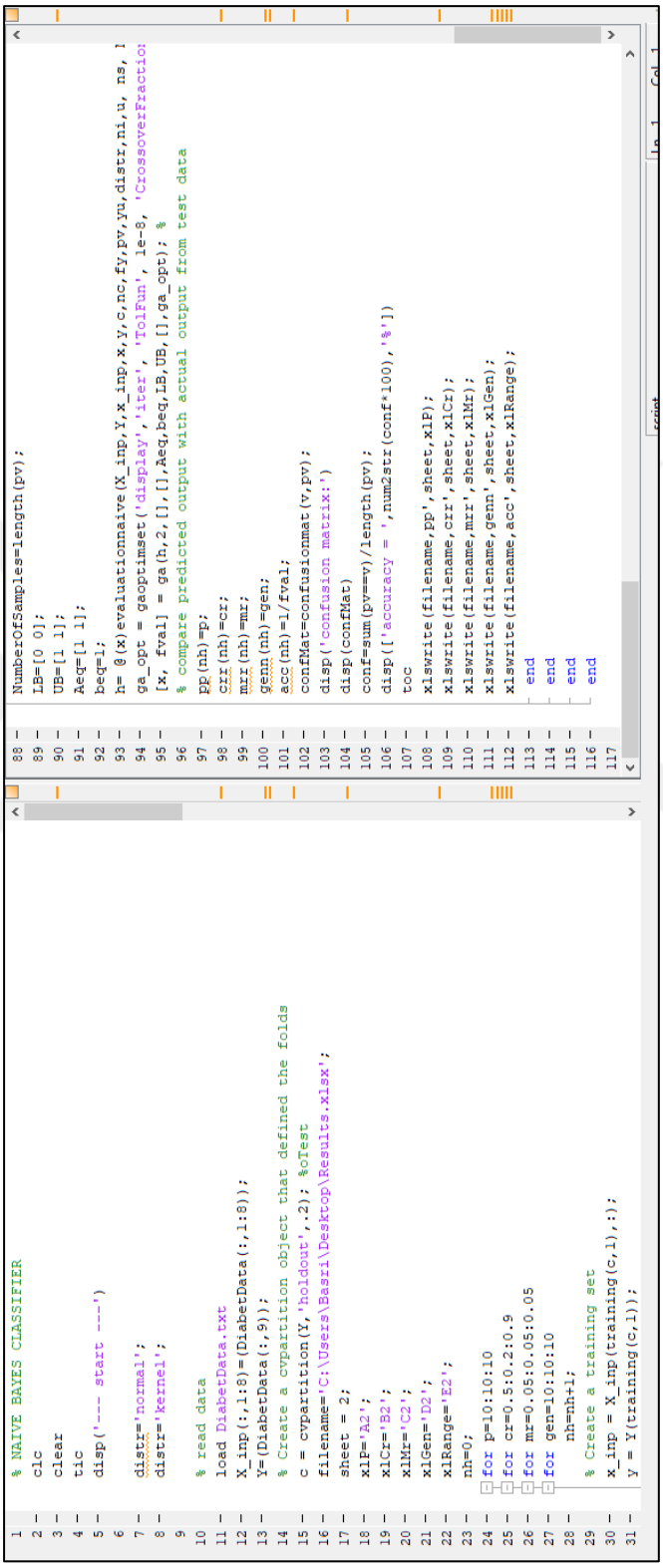
EKLER

EK I. GA-YSA



```
clc , clear
load DiabetData.txt
load DiabetTest.txt
I(:,1:8)=(DiabetData(:,1:8));
I=I';
O=(DiabetData(:,9));
O=O';
P(:,1:8)=(DiabetTest(:,1:8));
P=P';
filename='C:\Users\Basri\Desktop\Results.xlsx';
sheet = 3;
xlP='A2';
xlCr='B2';
xlMr='C2';
xlGen='D2';
xlRange='E2';
nh=0;
for p=10:10:10
for cr=0.5:0.2:0.9
for mr=0.05:0.05:0.05
for gen=10:10:10
    nh=nh+1;
[pn,minp,maxp,tn,mint,maxt] = premnmx(I,O); %normalize eder
n=150; %değişebilir
net = feedforwardnet(n);
%net = newrbe(I,O,0.25);
net.layers{1}.transferFcn = 'tansig';
net.layers{2}.transferFcn = 'satlins';
net.trainparam.lr=0.25;
%net.trainparam.mc=0.25; osilasyon
%net.trainParam.epochs = 1000;
%net.trainParam.goal = 1e-5;
%net.trainFcn='trainbr';
%net.performFcn = 'mse';
net = configure(net, I,O); %%ysa modeli burada yapilandirilir
h= @(x)mse_test(x,net,I,O);
ga_opt = gaoptimset('display','iter', 'TolFun', 1e-8, 'CrossoverFraction
[x_ga_opt, final_err_ga, exfl, ouput] = ga(h,10*n+1,ga_opt); % n= 4*n in
net = train(net, I,O); %%%
an=sim(net,I);
%view(net);
[a] = postmnmx(an,mint,maxt);
Y = net(I);
Y=Y';
z= net(P);
z=z';
Accuracy=1/final_err_ga;
pp(nh)=p;
crr(nh)=cr;
mrr(nh)=mr;
genn(nh)=gen;
acc(nh)=Accuracy;
xlswrite(filename,pp',sheet,xlP);
xlswrite(filename,crr',sheet,xlCr);
xlswrite(filename,mrr',sheet,xlMr);
xlswrite(filename,genn',sheet,xlGen);
xlswrite(filename,acc',sheet,xlRange);
end
end
end
end
```

EK II. GA-NB

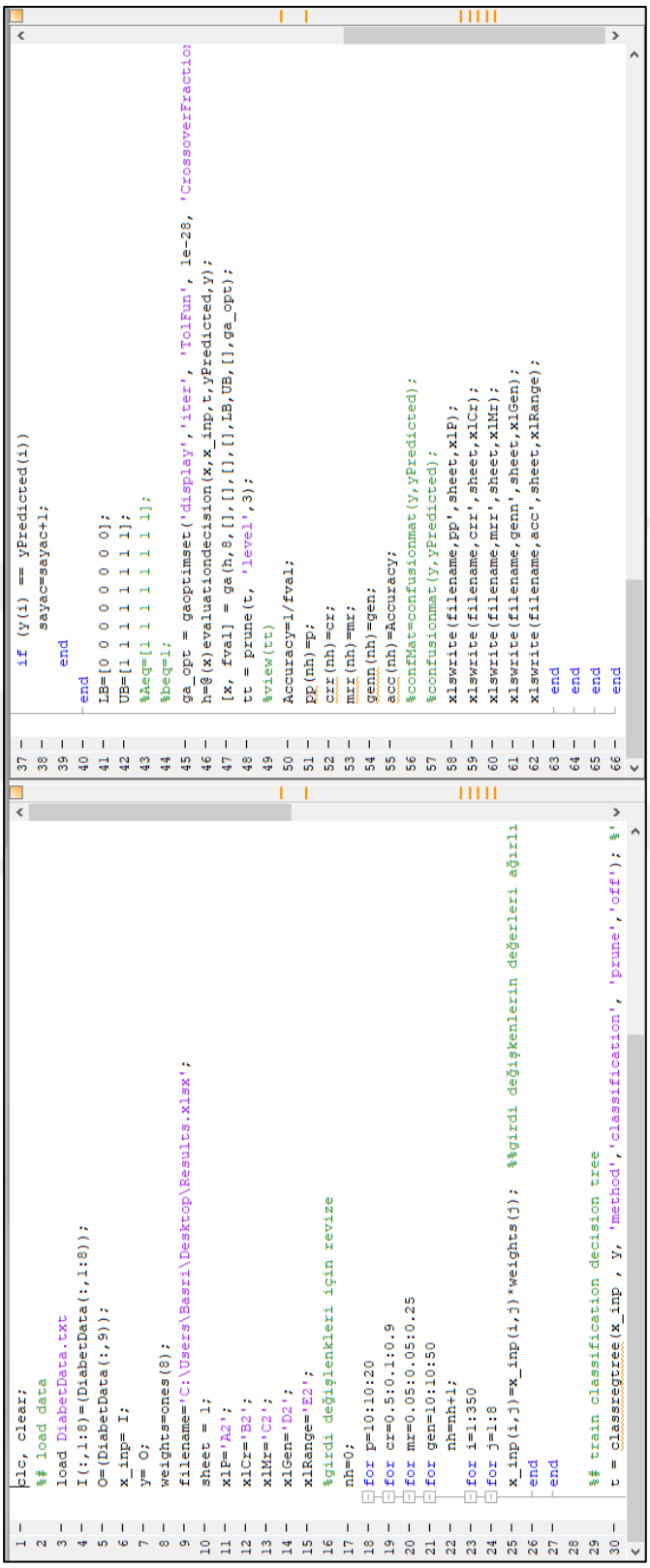


```
% NAIVE BAYES CLASSIFIER
clc
clear
tic
disp('---- start ----')

distr='normal';
distr='kernel';

% read data
load DiabetData.txt
X_inp(:,1:8)=(DiabetData(:,1:8));
Y=(DiabetData(:,9));
% Create a cvpartition object that defined the folds
c = cvpartition(Y,'holdout',.2); %otest
filename='C:\Users\Basri\Desktop\Results.xlsx';
sheet = 2;
xlP='A2';
xlCr='B2';
xlMr='C2';
xlGen='D2';
xlRange='E2';
nh=0;
for p=10:10:10
for cr=0.5:0.2:0.9
for mr=0.05:0.05:0.05
for gen=10:10:10
    nh=nh+1;
% Create a training set
x_inp = X_inp(training(c,1),:);
y = Y(training(c,1));
NumberOfSamples=length(pv);
LB=[0 0];
UB=[1 1];
Aeq=[1 1];
beq=1;
h= @(x)evaluationnaive(X_inp,Y,x_inp,x,y,c,nc,fy,pv,yu,distr,ni,u, ns, 1
ga_opt = gaoptimset('display','iter', 'TolFun' , 1e-8, 'CrossoverFractio
[x, fval] = ga(h,2,[],[],Aeq,beq,LB,UB,[],ga_opt); %
% compare predicted output with actual output from test data
pp(nh)=p;
crr(nh)=cr;
mrr(nh)=mr;
genn(nh)=gen;
acc(nh)=1/fval;
confMat=confusionmat(v,pv);
disp('confusion matrix:')
disp(confMat)
conf=sum(pv==v)/length(pv);
disp(['accuracy = ',num2str(conf*100),'%'])
toc
xlswrite(filename,pp',sheet,xlP);
xlswrite(filename,crr',sheet,xlCr);
xlswrite(filename,mrr',sheet,xlMr);
xlswrite(filename,genn',sheet,xlGen);
xlswrite(filename,acc',sheet,xlRange);
 end
end
end
end
```

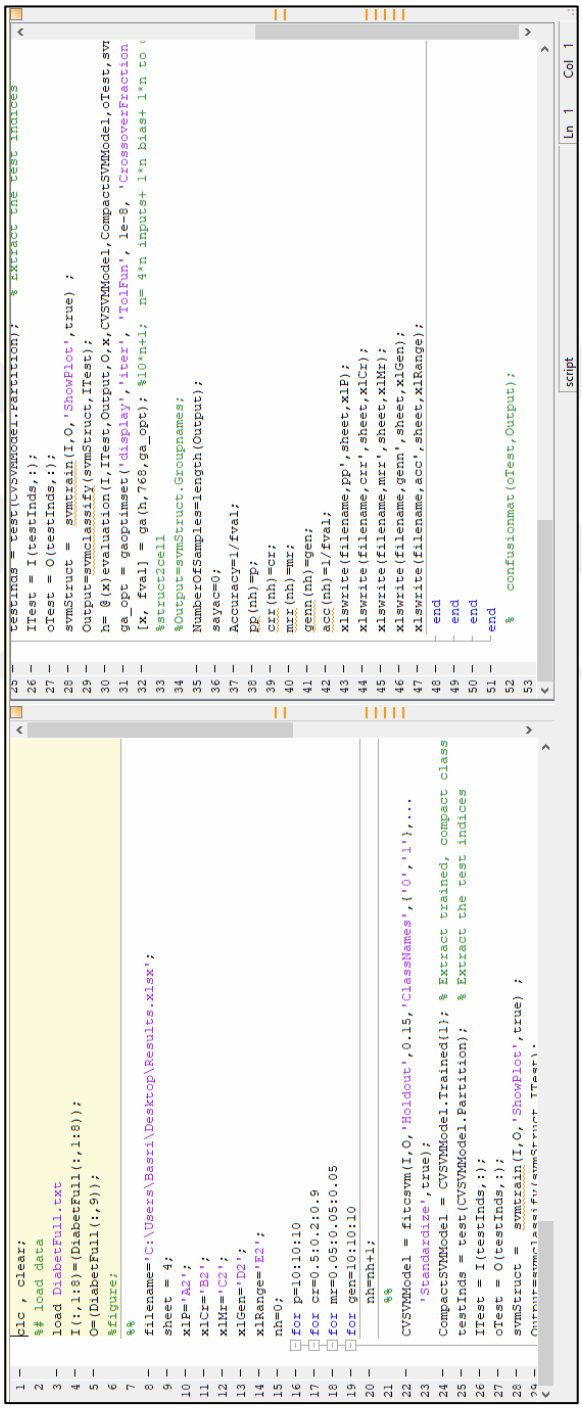
EK III. GA-KA



```
clc, clear;
%# load data
load DiabetData.txt
I(:,1:8)=(DiabetData(:,1:8));
O=(DiabetData(:,9));
x_inp= I;
y= O;
weights=ones(8);
filename='C:\Users\Basri\Desktop\Results.xlsx';
sheet = 1;
xlP='A2';
xlCr='B2';
xlMr='C2';
xlGen='D2';
xlRange='E2';
%girdi değişkenleri için revize
nh=0;
for p=10:10:20
for cr=0.5:0.1:0.9
for mr=0.05:0.05:0.25
for gen=10:10:50
    nh=nh+1;
for i=1:350
for j=1:8
x_inp(i,j)=x_inp(i,j)*weights(j);  %%girdi değişkenlerin değerleri ağırlı
end
end

%# train classification decision tree
t = classregtree(x_inp , y, 'method','classification', 'prune','off'); %'
    if (y(i) == yPredicted(i))
        sayac=sayac+1;
    end
end
LB=[0 0 0 0 0 0 0 0];
UB=[1 1 1 1 1 1 1 1];
%aeq=[1 1 1 1 1 1 1 1];
%beq=1;
ga_opt = gaoptimset('display','iter', 'TolFun', 1e-28, 'CrossoverFraction
h=@(x)evaluationdecision(x,x_inp,t,yPredicted,y);
[x, fval] = ga(h,8,[],[],[],[],LB,UB,[],ga_opt);
tt = prune(t, 'level',3);
%view(tt)
Accuracy=1/fval;
pp(nh)=p;
crr(nh)=cr;
mrr(nh)=mr;
genn(nh)=gen;
acc(nh)=Accuracy;
%confMat=confusionmat(y,yPredicted);
%confusionmat(y,yPredicted);
xlswrite(filename,pp',sheet,xlP);
xlswrite(filename,crr',sheet,xlCr);
xlswrite(filename,mrr',sheet,xlMr);
xlswrite(filename,genn',sheet,xlGen);
xlswrite(filename,acc',sheet,xlRange);
end
end
end
```

EK IV. GA-DVM



ÖZGEÇMİŞ

Adı Soyadı : Ebru PEKEL

Doğum Yeri : Samsun

Doğum Tarihi : 29.11.1992

Yabancı Dili : İngilizce

Eğitim Durumu

Lise : Bayrampaşa Sağmalcılar Lisesi (2010)

Lisans : İstanbul Üniversitesi Mühendislik Fakültesi Endüstri Mühendisliği (2014)

Çalıştığı Kurum/Yıl

Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü Araştırma Görevlisi / Nisan 2017 – Halen

İstanbul Kültür Üniversitesi Mühendislik Fakültesi Araştırma Görevlisi / Eylül 2015-Eylül 2016

Yayınlar

Ceylan, Z., Pekel, E. (2017). Comparison of Multi-Label Classification Methods for Prediagnosis of Cervical Cancer. International Journal of Intelligent Systems and Applications in Engineering, 5(4), 232-236.

ÖZTÜRK H., PEKEL E., ELEVLİ B. (2017). ANP ve ELECTRE Yöntemleri Kullanılarak Tedarikçi Seçimi: Kablo Sektörü Uygulaması. SAÜ Fen Bilimleri EnstitüsüDergisi, 1-1., Doi: 10.16984/saufenbilder.315330

PEKEL EBRU, BALTA B. (2015). The Evaluating Shopping Mall Locations Based on Fuzzy AHP Integrated Fuzzy VIKOR. Fuzzyss-15 (Tam MetinBildiri/SözlüSunum)(Yayın No:3598102)

PEKEL E., ÖZCAN T. (2017). DIAGNOSIS OF DIABETES MELLITUS USING STATISTICAL METHODS AND MACHINE LEARNING ALGORITHMS. The 4th International Management Information Systems Conference (IMISC2017)

PEKEL E., ELEVLİ S., (2017). COMPARISON OF FUNDAMENTAL MACHINE LEARNING ALGORITHMS ON EVALUATION STUDENT SURVEY. 1st

International Conference on Computational and Statistical Methods in Applied Sciences.

PEKEL E., CEYLAN Z. (2017). A HYBRID DECISION TREE - GENETIC ALGORITHM (DT-GA) STUDY FOR DIAGNOSIS OF KIDNEY DISEASE. XVIII. Uluslar arası Ekonometri, Yöneylem Araştırması ve İstatistik Sempozyumu 2017

PEKEL E., CEYLAN Z., (2017). PERFORMANCE OF MACHINE LEARNING ALGORITHMS FOR DIAGNOSING CHRONIC KIDNEY DISEASE. XVIII. Uluslararası Ekonometri, Yöneylem Araştırması ve İstatistikSempozyumu 2017

PEKEL E., ELEVLİ S., (2017). THEORY of CONSTRAINTS with SIMULATION MODELING: CASE STUDY. ICEBSS'17

PEKEL E., ELEVLİ S., (2017). The Evaluation of Manufacturing Machines Based on Fuzzy AHP Integrated Fuzzy TOPSIS. ICEBSS'17