T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
İŞLETME ANABİLİM DALI
SAYISAL YÖNTEMLER BİLİM DALI

SINIFLANDIRMA SORUNU İÇİN EN UYGUN DEĞİŞKEN ALT KÜMESİ SEÇİMİ ÜZERİNE BİR UYGULAMA

Yüksek Lisans Tezi

EMRAH SEZER

İstanbul, 2018

T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
İŞLETME ANABİLİM DALI
SAYISAL YÖNTEMLER BİLİM DALI

SINIFLANDIRMA SORUNU İÇİN EN UYGUN DEĞİŞKEN ALT KÜMESİ SEÇİMİ ÜZERİNE BİR UYGULAMA

Yüksek Lisans Tezi

EMRAH SEZER

Danışman: Doç. Dr. ÖZGÜR ÇAKIR

İstanbul,2018

T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ

TEZ ONAY BELGESİ

İŞLETME Anabilim Dalı SAYISAL YÖNTEMLER Bilim Dalı TEZLİ YÜKSEK LİSANS öğrencisi EMRAH SEZER'nın SINIFLANDIRMA SORUNU İÇİN EN UYGUN DEĞİŞKEN ALT KÜMESİ SEÇİMİ ÜZERİNE BİR UYGULAMA adlı tez çalışması, Enstitümüz Yönetim Kurulunun 8.03.2018 tarih ve 2018-7/40 sayılı kararıyla oluşturulan jüri tarafından oy birliği /~~oy çokluğu~~ ile Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi 15/03/2018

Öğretim Üyesi Adı Soyadı			**İmzası**
1.	Tez Danışmanı	Doç. Dr. ÖZGÜR ÇAKIR	
2.	Jüri Üyesi	Prof. Dr. ÖMER ÖNALAN	
3.	Jüri Üyesi	Prof. Dr. U.TUĞBA GÜRSOY	

ÖZET

SINIFLANDIRMA SORUNU İÇİN EN UYGUN DEĞİŞKEN ALT KÜMESİ SEÇİMİ ÜZERİNE BİR UYGULAMA

Teknolojik gelişmelerin etkisi ile kaydedilen operasyonel veriler giderek artmaktadır. Veri miktarı ve çeşitliliğindeki artış nedeni ile analiz aşamasında ve analiz sonuçlarının değerlendirilmesi aşamasında bir çok zorluk yaşanmaktadır. İlgili ve ilgisiz birçok verinin analiz aşamasına aktarılmasının sonucunda analizlerin yapılabilmesi için gerekli zaman ve kaynak gereksinimleri artmaktadır. Kaynakların ve zamanın daima sınırlı olacağı aşikardır. Bu çalışmadaki amacımız, bankacılık müşteri verisi üzerinde sınıflandırma için farklı değişken seçimi uygulamaları yaparak ilgisiz değişkenleri elemek ve sınıflandırma çalışmasına katkıda bulunmaktır. Farklı değişken seçimi yöntemleri kullanılarak seçilen değişken alt kümeleri üzerinde sınıflandırma uygulaması yapılmıştır. Sınıflandırma sonuçları karşılaştırılarak değişken seçim yöntemlerinin faydaları araştırılmıştır.

Anahtar Kelimeler : Değişken Seçimi, Sınıflandırma, Veri Madenciliği, Korelasyon Bazlı Değişken Seçimi, Buluşsal Arama.

ABSTRACT

AN APPLICATION ON FEATURE SELECTION FOR CLASSIFICATION

Recorded operational data is gradually increasing with the effect of technological developments. Due to the increase in the amount of data and its diversity, there are many difficulties in the analysis phase and in the evaluation of its results as a result of transferring many relevant and irrelevant data to the analysis stage, the time and resources required for the analysis are increased. It is obvious that resources and time will always be limited. Our aim in this study is to contribute to the study of classifying and eliminating irrelevant features by applying different feature selection methods for classification on a banking customer data. Classification is applied on the selected subsets of features using different feature selection methods. By comparing the classification results, the benefits of the feature selection methods investigated.

Key Words: Feature Selection, Classification, Data Mining, Correlation Based Feature Selection, Heuristic Search.

ÖNSÖZ

Yüksek lisans tez sürecimin tamamında değerli bilgilerini benimle paylaşan ve her konuda destek olan hocam Doç. Dr. Özgür Çakır'a tüm katkılarından dolayı teşekkür ederim.

Yüksek lisans dahil olmak üzere tüm eğitim hayatım boyunca desteklerini benden esirgemeyen annem Sebile Sezer'e, babam Fevzi Sezer'e ve kardeşim Furkan Sezer'e teşekkür ederim.

İstanbul, 2018 Emrah Sezer

İÇİNDEKİLER

ÖZET i
ABSTRACT ii
ÖNSÖZ iii
TABLO LİSTESİ vii
ŞEKİLLER LİSTESİ viii
KISALTMALAR ix
GİRİŞ 1
1. VERİ MADENCİLİĞİ VE SINIFLANDIRMA 3
1.1. Veri Madenciliği 3
1.2. Veri Madenciliği Tarihsel Gelişimi 4
1.3. Başlıca Veri Madenciliği Fonksiyonları 8
1.3.1. Tanımlama ve Özetleme 8
1.3.2. Kümeleme 8
1.3.3. Birliktelik Analizi 9
1.3.4. Sınıflandırma ve Regresyon 9
1.3.4.1. Karar Ağaçları 11
1.3.4.2. C4.5 Algoritması 12
1.4. Weka 16
2. DEĞİŞKEN SEÇİMİ 17
2.1. Değişken Seçimi Bakış Açısı 18
2.2. Sınıflandırma İçin Değişken Seçimi 19
2.3. Sonuçlarına Göre Değişken Seçim Algoritmaları 20
2.3.1. Değişken Sıralama Algoritmaları 20
2.3.2. En Küçük Alt Küme Algoritmaları 21
2.4. Arama Problemi 21
2.5. Arama Yönleri 22
2.6. Arama Yöntemleri 23
2.6.1. Sıralayıcı Arama 23
2.6.2. Tam Arama 23
2.6.3. Buluşsal Arama 23
2.6.4. Sıralı Arama 24

2.6.5. Rastgele Arama 24
2.7. Değişken Seçim Yöntemlerinde Değerlendirme Ölçüleri 24
2.8. Değişken Seçim Modelleri 27
2.8.1. Sarmalayıcı Modeller 27
2.8.2. Filtreleyici Modeller 29
2.8.2.1. Bilgi Kazanımı Bazlı Değişken Seçimi 30
2.8.2.2. Kazanç Oranı Bazlı Değişken Seçimi 31
2.8.2.3. Simetrik Belirsizlik Bazlı Değişken Seçimi 32
2.8.2.4. Korelasyon Bazlı Değişken Seçimi 33
2.8.2.5. Tutarlılık Bazlı Değişken Seçimi 35
2.9. Çalışmada Kullanılan Weka Araçları 38
3. SINIFLANDIRMA SORUNU İÇİN EN UYGUN DEĞİŞKEN ALT KÜMESİ SEÇİMİ ÜZERİNE BİR UYGULAMA 39
3.1. Amaç ve Kapsam 39
3.2. Veri Kümesi 40
3.3. Veri Hazırlama Süreci 41
3.4. Verinin Tanıtılması 42
3.5. Sınıflandırma Uygulamasına Yönelik Değişken Seçimi Uygulamaları 43
3.6. Karar Ağacı Uygulamaları 49
3.6.1. Değişken Seçimi Yapılmadan Oluşturulan Karar Ağacı 49
3.6.2. Yöntem-1 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 50
3.6.3. Yöntem-2 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 51
3.6.4. Yöntem-3 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 51
3.6.5. Yöntem-4 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 52
3.6.6. Yöntem-5 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 53
3.6.7. Yöntem-6 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 53
3.6.8. Yöntem-7 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı 54
3.7. Değerlendirme 55
3.8. Bulgular ve Tartışma 58
SONUÇ 60
EKLER 62
Ek-1: Yöntem-1'e Ait Weka Çıktısı 62
Ek-2: Yöntem-2'e Ait Weka Çıktısı 63
Ek-3: Yöntem-3'e Ait Weka Çıktısı 65

Ek-4: Yöntem-4'e Ait Weka Çıktısı 66
Ek-5: Yöntem-5'e Ait Weka Çıktısı 69
Ek-6: Yöntem-6'e Ait Weka Çıktısı 72
Ek-7: Yöntem-7'e Ait Weka Çıktısı 75
Ek-9 : Yöntem-1 'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 83
Ek-10 : Yöntem-2'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 85
Ek-11 : Yöntem-3'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 92
Ek-12 : Yöntem-4'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 95
Ek-13 : Yöntem-5'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 97
Ek-14 : Yöntem-6'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 100
Ek-15 : Yöntem-7'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı 102
Ek-16 : Değişken Seçimi Yapılmadan Oluşturulan Karar Ağacı..... 108
Ek-17 : Yöntem-1 'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 109
Ek-18 : Yöntem-2'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 110
Ek-19 : Yöntem-3'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 111
Ek-20 : Yöntem-4'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 112
Ek-21 : Yöntem-5'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 113
Ek-22 : Yöntem-6'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 114
Ek-23 : Yöntem-7'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı 115
KAYNAKLAR..... 116

TABLO LİSTESİ

Tablo 1. İkili Kategorik Değişken İçin Hata Matrisi 14
Tablo 2. Kappa Değerlerinin Yorumu 15
Tablo 3. Veri Kümesinde Bulunan Değişkenler 40
Tablo 4. D01 Değişkenine Ait Sıklık Dağılımı 42
Tablo 5. D02 Değişkenine Ait Sıklık Dağılımı 42
Tablo 6. D03 Değişkenine Ait Sıklık Dağılımı 42
Tablo 7. D04 Değişkenine Ait Sıklık Dağılımı 43
Tablo 8. C02 Değişkenine Ait Sıklık Dağılımı 43
Tablo 9. Kullanılan Yöntemlerin İçerdiği Modeller 49
Tablo 10. Tüm Değişkenler İle Oluşturulan Karar Ağacı 49
Tablo 11. Yöntem-1' e Ait Sınıflandırma Çıktısı 50
Tablo 12. Yöntem-2' ye Ait Sınıflandırma Çıktısı 51
Tablo 13. Yöntem-3' e Ait Sınıflandırma Çıktısı 51
Tablo 14. Yöntem-4' e Ait Sınıflandırma Çıktısı 52
Tablo 15. Yöntem-5' e Ait Sınıflandırma Çıktısı 53
Tablo 16. Yöntem-6'ya Ait Sınıflandırma Çıktısı 53
Tablo 17. Yöntem-7'ye Ait Sınıflandırma Çıktısı 54
Tablo 18. Karar Ağaçlarının Karşılaştırılması 56
Tablo 19. Yöntemlerin Karşılaştırılması 57
Tablo 20. Yöntemlerin Sıralanması 57

ŞEKİLLER LİSTESİ

Şekil 1. Veri Madenciliği Tarihçesi 7
Şekil 2. Arama Yönleri 21
Şekil 3. Weka Ara Yüz Görüntüsü - 1 44
Şekil 4. Weka Ara Yüz Görüntüsü - 2 45
Şekil 5. Weka Ara Yüz Görüntüsü - 3 47

KISALTMALAR

CFS : Korelasyon Bazlı Değişken Seçimi
IG : Bilgi Kazanımı
GR : Kazanç Oranı
SU : Simetrik Belirsizlik
GNU : Genel Kamu Lisansı
MAE : Ortalama Mutlak Hata

GİRİŞ

İşletmelerdeki makineleşme ve otomatizasyon çalışmalarının büyük bir hızla artması ile birlikte bilgisayar tarafındaki donanım maliyetlerinin düşmesi, biriktirilen operasyonel verinin boyutlarının ve çeşitliliğinin giderek artmasına imkan sağlamıştır. Bu nedenle işletmeler mümkün olan tüm kayıtları saklama eğilimindedir.

İşletmelerde kaydedilen büyük çaptaki verilerden anlamlı ilişkiler çıkarmak için kullanılan yöntemlerden bir tanesi de veri madenciliğidir. Veri madenciliği ile tanımlayıcı çalışmalar yapılabildiği gibi öngörü modelleri de üretilebilmektedir.

Sınıflandırma, gerçekleşmiş verilerden hareketle kategorik bir hedef değişkene yönelik öngörü modelleri üretmek amacı ile kullanılan bir veri madenciliği yöntemidir. Sınıflandırma uygulamasında kategorik ve sürekli verilerle çalışma imkanı bulunmaktadır. Sınıflandırmanın karar ağaçları, yapay sinir ağları, lojistik regresyon, destek vektör makinesi gibi farklı yapılarda algoritmaları bulunmaktadır.

İşletmeler tarafında geniş çapta tutulan veriler, sınıflandırma analizi için bir araya getirildiğinde analizi olumsuz etkileyebilecek bir çok handikap barındırabilmektedir. İşletme tarafından kullanılan ilişkisel veri tabanları ve veri ambarları gibi kaynaklardan analiz için oluşturulan veri kümesinde hedef değişken ile alakasız veya hedef değişkeni açıklayıcılığı çok düşük değişkenler de bulunabilmektedir. Bu açıklayıcı olmayan değişkenlerin ayıklanması amacı ile bir veri madenciliği uygulaması olan değişken seçimi algoritmaları kullanılmaktadır. Gereksiz değişkenlerin elenmesi ile sınıflandırma analizi ile oluşturulan öngörü modelinin başarısını arttırmak, ortaya çıkan modelin daha sade ve anlaşılabilir olmasını sağlamak ve sınıflandırma analizi için gerekli olan analiz süresini kısaltmak mümkündür.

Ancak geniş bir veri kümesinde bulunan bir çok değişkenden hangilerinin seçilmesi gerektiğine karar vermek değişken seçimi algoritmaları için de her zaman çok kolay olmamaktadır. Değişken uzayının her alt kümesinin, aranan ideal değişken alt kümesi olma ihtimali bulunmaktadır ve geniş veri kümelerinde bu uzay çok büyük olabilmektedir. Bu kadar fazla alternatifin hepsini değerlendirmeye dayalı yöntemler bulunmasına karşın zamanın ve eldeki kaynakların kısıtlı olduğu durumlarda pek kullanışlı değildir. Bu aşamada durum bir optimizasyon problemine dönüştüğü için

rastgele arama yöntemlerine veya buluşsal arama yöntemlerine dayalı algoritmalar bulunmaktadır. Bu yöntemler ile değişken uzayının sınırlı bir bölümü üzerinde değerlendirme yapılır ve hedef değişkeni en iyi açıkladığı tespit edilen değişken alt kümesi sınıflandırma çalışması için önerilir.

Bu çalışmada, gerçek bir bankacılık operasyonel veri kümesi üzerinde veri madenciliği süreçleri içerisinde bulunan sınıflandırma uygulamasına yönelik farklı değişken alt kümesi seçme yöntemlerinden faydalanarak elde edilen değişken alt kümeleri üzerinde sınıflandırma uygulaması yapılarak değişken seçim yöntemlerinin performanslarının karşılaştırılmalı olarak incelenmesi amaçlanmıştır. Tez çalışması üç bölümden oluşmaktadır.

Birinci bölümde, genel hatları ile veri madenciliği kavramı ve süreçleri ile bu çalışmada üzerinde durduğumuz sınıflandırma ve karar ağaçları anlatılmıştır. Ayrıca çalışmada araç olarak kullanılan Weka hakkında bilgi verilmiştir.

İkinci bölümde, sınıflandırma için değişken seçim yöntemleri, değişken seçimi için kullanılan modeller, değişken alt kümesi seçimi için kullanılan arama yönleri, arama yöntemleri, değerlendirme ölçüleri, değişken seçim algoritmaları anlatılmıştır.

Üçüncü bölümde, elimizde bulunan bankacılık verisi üzerinde sınıflandırmaya yönelik yedi farklı değişken seçim uygulaması yapılarak bu uygulamaların sonucunda seçilen değişken alt kümeleri ile sınıflandırma çalışması yapılmıştır. Değişken alt kümesi seçilmeksizin mevcut bulunan veri kümesi ve bu veri kümesinden farklı değişken seçim yöntemleri kullanılarak seçilmiş bulunan 7 adet değişken alt kümesi ayrı ayrı J48 karar ağacı uygulamasına tabi tutulmuş ve sonuçları özetlenerek karşılaştırmalı olarak incelenmiştir.

Tez çalışmasının sonunda ise çalışma boyunca elde edilen bulgular bir sonuç bölümü olarak yer almıştır.

1. VERİ MADENCİLİĞİ VE SINIFLANDIRMA

1.1. Veri Madenciliği

Veri Madenciliği, endüstride sahip olunan büyük miktarda veriden bilgi ve tecrübe elde etme ihtiyacından dolayı son yıllarda ilgi çekmektedir. Bilgi elde etme amacı ile piyasa analizi, olağan dışı durumların tespiti, müşteri sadakati, üretim kontrolü ve bilimsel araştırmalar gibi bir çok uygulamada veri madenciliği kullanılabilmektedir.

Son 30 yılda donanım teknolojilerindeki büyük ilerleyiş sayesinde güçlü ve makul fiyatlı bilgisayarlar kullanılarak büyük boyutlardaki verileri toplama ve saklama imkanları oluştu. Bu teknoloji, veri tabanı ve bilişim sektörünü ilerletti ve veri işleme süreçlerinde çok sayıda veri tabanı ve bilgi havuzu oluşturulmasına ve verilerin analiz edilmesine imkan sağladı[1].

İşletme, bilim ve mühendislik gibi alanların neredeyse tamamında büyük, karmaşık, bilgi açısından zengin veri kümelerini anlama ihtiyacı bulunmaktadır. İş dünyasında bireysel ve kurumsal müşteri verileri stratejik bir varlık olarak kabul edilmektedir. Bu verilere gizlenmiş faydalı bilgileri ayıklayabilmek ve kullanabilmek günümüz dünyasında gittikçe önem kazanmaktadır. Veriden bilgi keşfetme sürecinin tamamının yeni teknikler ile birlikte bilgisayar tabanlı bir metodoloji ile uygulanmasına veri madenciliği adı verilir.

Olası tüm görüşleri kapsamayı amaçlamadan veri madenciliğini bir disiplin olarak kabaca ifade etmek gerekirse, veri madenciliği belirli bir veri kümesindeki çeşitli modelleri, özetleri ve türetilmiş değerleri keşfetme sürecidir. Bazı profesyonel ortamlarda bile uygulamanın bir süreç olduğu göz ardı edilerek veri toplama ve uygulama işleminden ibaret olduğu sanılmaktadır. Bu yaklaşım büyük bir yanılgıdır.

Veri Madenciliğinin faydalı olabilmesi için maliyetleri düşürmek, kar düzeyini artırmak, müşteri memnuniyetini artırmak veya hizmet kalitesini artırmak gibi şirketin ana hedeflerine katkıda bulunması gerekir[2].

[1] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.
[2] Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.

Geniş çapta veriler arasından bilgi çıkarılması işlemi, kayaların arasından altın çıkarma işçiliğine madencilik adı verilmesinden esinlenilerek veri madenciliği olarak adlandırılmıştır.

Genel olarak Veri madenciliği süreçleri, açıklayıcı (descriptive) ve öngörücü (predictive) olarak iki ana başlık altında incelenmektedir. Açıklayıcı veri madenciliği uygulamaları, veri tabanında bulunan verilerin genel özelliklerini açıklamaktadır. Öngörücü veri madenciliği uygulamaları ise öngörüde bulunmak için elde bulunan verilerden faydalanır[3].

Veri Madenciliği bilgisayar dünyasında en hızlı büyüyen alanlardan biridir. Önceleri bilgisayar ve istatistik bilimlerinin ortak ilgi alanı konumundayken kısa süre içerisinde kendi başına bir alan olmuştur.

Misyonu dolandırıcılık ve suç faaliyetlerini tespit etmek olan bir çok veri madenciliği uygulamaları, kolluk kuvvetleri ve özel araştırma birimleri tarafından başarılı şekilde uygulanmıştır[4].

1.2. Veri Madenciliği Tarihsel Gelişimi

1980'lerde yapay sinir ağları, karar ağaçları gibi algoritmaların az sayıda değişken (feature) ve az sayıda örnek içeren küçük veri kümeleri üzerinde kullanılmasına makine öğrenmesi adı verilmiştir. Veri miktarının artması ve gelişmiş veri tabanı sistemlerinin kullanılmaya başlanmasından sonraki dönemlerde ise makine öğrenmesinin geniş veri tabanları üzerinde uygulanmasına veri madenciliği adı verilmiştir[5,6].

1960'lardan beri veri tabanı ve bilgi teknolojisi, ilkel dosya işleme sistemlerinden karmaşık ve güçlü veri tabanı sistemlerine kadar sistematik bir şekilde gelişimini sürdürmüştür. 1970'ler itibari ile ilişkisel veri tabanı sistemlerinin geliştirilmesi ile birlikte veri modelleme araçları ve veriye erişim araçları kullanılmaya başlanmıştır. Son kullanıcılar sorgu dilleri, kullanıcı ara yüzleri, sorgu optimizasyonu uygulamaları ve işlem yönetimi gibi araçlar sayesinde esnek veri erişimi elde etmişlerdir.

---

[3] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.
[4] Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.
[5] Alpaydin, E. 2014. *Introduction to machine learning*. MIT press.
[6] Fürnkranz, J., D. Gamberger & N. Lavrač. 2012. *Foundations of rule learning*. Springer Science & Business Media.

1980'lerin ortalarından itibaren, ilişkisel veri tabanı teknolojinin popüler hale gelmesi ve güçlü veri tabanı sistemleri üzerinde ar-ge faaliyetlerinin artması ile veri tabanı teknolojisi karakterize edilmiştir. Heterojen veri tabanı sistemleri ve internet tabanlı global bilgi sistemleri de ortaya çıkmış ve bilişim endüstrisinde hayati bir rol oynamışlardır.

Bilgisayar donanımı teknolojisinin son otuz yıldaki istikrarlı ve şaşırtıcı ilerleyişi, güçlü ve uygun fiyatlı bilgisayarların, veri toplama ekipmanlarının ve depolama ortamlarının gelişmesini sağlamıştır. Teknolojideki bu ilerleme, veri tabanı ve veri bilimi alanlarında ilerlemelerin yaşanmasına kolaylık sağlamıştır.

Biriken veriyi uygun şekilde yönetebilmek ve etkin şekilde analiz edebilmek için veri ambarı yapısı ortaya çıkmıştır. Veri ambarı teknolojisi, veri temizleme, veri entegrasyonu, özetleme ve birleştirme gibi işlevselliği olan analiz tekniklerinin yanı sıra bilgiyi farklı açılardan görüntüleme imkanı sağlar[7].

Yapay sinir ağları, 1940'lı yıllarda ortaya atılmasına rağmen bu sinir ağlarını eğitmenin hiçbir yöntemi yoktu. Programcılar bu ağların ağırlık matrislerini elle hazırlamak zorundaydılar. Bu işlem yorucu olduğu için, sinir ağları ilk kez kullanılmaz hale geldi.

İlerleyen yıllarda ağırlık matrislerinin otomatik şekilde oluşmasını sağlayan geliştirmeler yapılmasına, 1980'lerde ve 1990'ların başındaki bilgisayarların hesaplama gücünün eklenmesi ile sinir ağlarının görevleri yerine getirmesi kolaylaştı. Ancak bu çağın donanım ve eğitim algoritmaları, sinir ağlarını çok katmanlı bir şekilde eğitemezdi ve ikinci sefer sinir ağları kullanılamaz duruma geldi. Bu süreç ikinci nesil yapay sinir ağları olarak adlandırıldı. İkinci nesil içerisinde çekirdek (kernel) makineleri aracılığı ile yapay sinir ağlarında optimizasyon çalışmaları yapılmıştır. Bir çekirdek makinesi ile eğitim sürecinde, optimum modelin verimli bir şekilde bulunmasının garanti altına alınabilmesi sağlandı. Çekirdek makineleri, temelde yatan fikirleri anlamak için insan operatöre çok fazla ihtiyaç duymadan kendiliğinden çalışacak yazılım uygulamalarına çok uygundur.

2006 yılında Hinton ve arkadaşları tarafından ortaya atılan yüksek hızlı grafik işleme ünitelerindeki son yaklaşım, programcıların sinir ağlarını üç veya daha fazla tabakayla

[7] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

eğitmesine olanak tanıdı ve programcılar derin sinir ağlarının faydalarını fark ettiği için yapay sinir ağları yeniden canlandı. Bu yaklaşımın farklı alanlarda uygulanmaya devam etmesi ile derin öğrenme kavramı konuşulmaya başlandı[8].

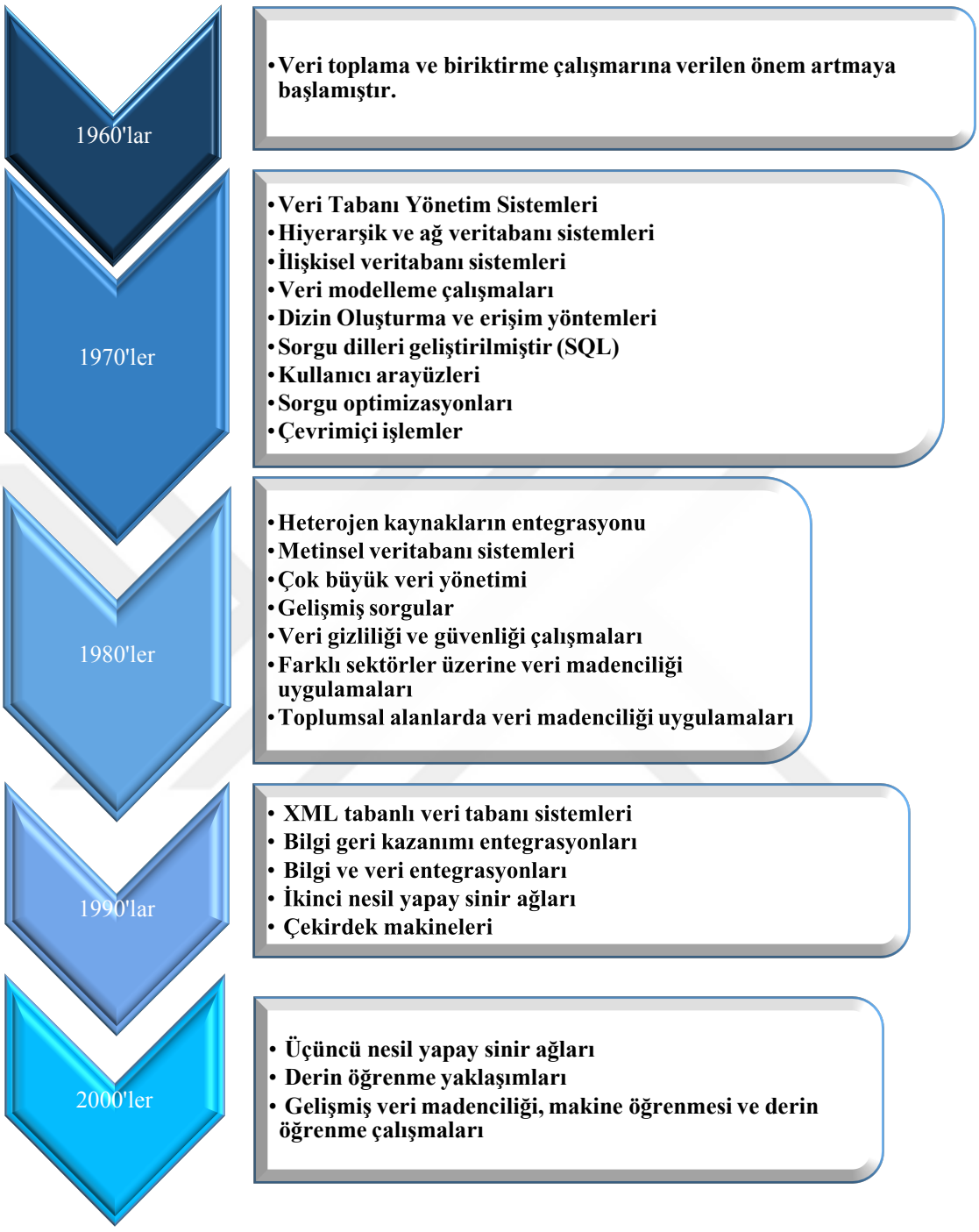
Veriler arasında karmaşık ilişkileri modellemek için çoklu temsil seviyelerini öğrenmek için kullanılan algoritmalara dayalı makine öğrenimi alanındaki bir alt alan olarak tanımlanan derin öğrenme ile, daha komplike değişkenler (features) daha sade olan değişkenlerle tanımlanır ve böyle bir değişken hiyerarşisi ile derin bir mimari oluşturulmuş olur.

Derin öğrenme, denetimli ya da denetimsiz değişken dönüştürme, model analizi ve sınıflandırma gibi birçok doğrusal olmayan bilgi işleme katmanını kullanan bir makine öğrenme teknikleri sınıfıdır[9].

Sinir ağlarının üçüncü neslinin popülaritesi devam etmekle birlikte, derin öğrenme çalışmalarının odağı bu neslin süresi boyunca çarpıcı biçimde değişmiştir. Üçüncü nesil, yeni denetimsiz öğrenme teknikleri ve derin modellerin küçük veri kümelerinden genelleme yeteneği üzerine odaklanmasıyla başlamıştır. Ancak bugün, daha büyük denetimli öğrenme algoritmalarına ve derin modellerin çok sayıda etiketli veri kümeleri ile çalışabilme yeteneğine daha fazla ilgi gösterilmektedir[10].

Bu bölümde anlatılan tarihsel gelişim aşamaları, bölüm içerisinde kullanıldığı belirtilen kaynaklardan faydalanılarak Şekil -1 ile özetlenmiştir.

[8] Hinton, G. E. & R. R. Salakhutdinov (2006) Reducing the dimensionality of data with neural networks. *science,* 313**,** 504-507.
[9] Heaton, J. 2015. *Artificial Intelligence for Humans: Deep Learning and Neural Networks*. Heaton Research, Incorporated.
[10] Bengio, Y., I. J. Goodfellow & A. Courville (2015) Deep learning. *Nature,* 521**,** 436-444.



Şekil 1. Veri Madenciliği Tarihçesi

1.3. Başlıca Veri Madenciliği Fonksiyonları

Genel olarak, veri madenciliği fonksiyonları, tanımlayıcı fonksiyonlar ve öngörücü fonksiyonlar olmak üzere iki kategoriye ayrılabilir. Tanımlayıcı fonksiyonlar, verilerin genel özelliklerini karakterize ederek veri hakkında açıklayıcı yeni bilgiler üretmeyi amaçlarken, öngörücü modeller ise yeni bir bilgi üretmeyip eldeki veriden hareketle öngörüde bulunmayı amaçlarlar[11,12].

Başlıca veri madenciliği fonksiyonları aşağıda belirtilmiştir.

1.3.1. Tanımlama ve Özetleme

Üzerinde çalışılan veriyi açıklamak ve genel veri yapısının karakterini ifade etmek amacı ile uygulanan temel veri madenciliği çalışmalarıdır. Veri kategorizasyonu (data categorization) ve veri ayrımlama (data discrimination) gibi çalışmalar bu başlık altında incelenir. Bu fonksiyonlar açıklayıcı veri madenciliği fonksiyonlarıdır.

Veri kategorizasyonu, veri tabanında bulunan veriler içerisinden açıklanması hedeflenen bir veri sınıfına ait özelliklerin özetlenmesi şeklindedir. Önceki yıla göre satışlarını arttırmış olan ürünlerin veri tabanından SQL sorgusu ile çekilmesi işlemi bir veri kategorizasyonu örneği olarak ifade edilebilir.

Veri ayrımlama ise birbirinden farklı iki veri sınıfının aralarındaki farkların özetlenmesi amacı ile yapılan çalışmalardır. Önceki yıla göre satışlarını arttıran ürünler ile satışlarını düşüren ürünlerin özelliklerinin karşılaştırılması çalışması bir veri ayrımlama çalışması olarak ifade edilebilir[13].

1.3.2. Kümeleme

Kümeleme analizi, üzerinde çalışılan veriyi önceden belirlenmiş herhangi bir sınıflandırmaya bağlı olmaksızın kendi içerisinde homojen parçalara ayırmayı amaçlayan açıklayıcı bir veri madenciliği çalışmasıdır.

---

[11] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.
[12] Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.
[13] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

Kümeleme analizinde, küme içerisindeki benzerliğin (similarity) ve kümeler arasındaki benzemezliğin (dissimilarity) en büyüklenmesi amaçlanmaktadır. Diğer bir deyişle, kendi aralarında benzerlikler bulunanların aynı kümede yer alması istenirken, kümelerin bir birlerine benzer olmaması hedeflenmektedir[14].

1.3.3. Birliktelik Analizi

Birliktelik analizinde, birlikte gerçekleşen olaylar ilişkilendirilerek birliktelik kurallarının ortaya çıkarılması hedeflenmektedir. Birliktelik kurallarının keşfi, veri madenciliğinin ana tekniklerinden biridir.

Birliktelik analizinin, market sepet analizi gibi uygulamaları ile tüketici davranışı açıklanmaya çalışılarak tüketicinin satın alma eğilimine odaklı olarak kampanyalar düzenlenebilmektedir.

1.3.4. Sınıflandırma ve Regresyon

Sınıflandırma analizinde, kesikli bir hedef değişken bulunur ve bu hedef değişkenin her farklı değerine sınıf etiketi adı verilir. Sınıflandırma analizi, eldeki gerçekleşmiş ve sınıf etiketi atanmış veri üzerinden hareketle sınıf etiketi bulunmayan her örnek için doğru sınıf etiketini atamayı amaçlayan bir model ortaya koyma çalışmasıdır.

Regresyon modelleri, prensip olarak sınıflandırma modellerine benzemesine karşın, sınıflandırma modellerinde hedef değişken kesikli iken, öngörü modellerinde hedef değişken süreklidir.

Bir bankanın, müşterilerinin kredi başvurularını değerlendirme aşamasında başvuru yapan müşteriye ait kayıtları ve daha önceki müşterilere ait kayıtları inceleyerek kredi başvurusuna olumlu ya da olumsuz gibi bir sonuç bağlaması işlemi bir sınıflandırma örneğidir. Bu örnekte, olumlu yanıt ve olumsuz yanıt sınıf etiketleridir. Başvuruda bulunan müşteriye ait verilerin, bankanın önceki müşterileri üzerinden oluşturulan

[14] Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.

modele göre değerlendirilerek uygun etiketin belirlenmesi çalışması ise sınıflandırma işlemidir[15].

Sınıflandırma modeli, eğitim verisinin (training data) bir çok analizi ile türetilir. Türetilen model, eğer-öyle ise (IF-Then) kuralları, karar ağaçları, matematiksel fonksiyonlar veya yapay sinir ağları gibi bir çok farklı şekilde görselleştirilebilirler.

Veri tabanları, karar verme süreçlerinde kullanılabilecek gizli bilgileri barındırma yönünden oldukça zengindirler. Sınıflandırma, veri sınıflarını tanımlayan modelleri çıkarmak ve gelecekteki veri eğilimlerini öngörmek için kullanılan bir veri analizi şeklidir. Bu analiz bize büyük çaptaki veriyi daha iyi anlamak için yardımcı olabilir. Örneğin, bir bankanın kredi başvurularını riskli veya güvenli olarak sınıflandırabiliriz. Sınıflandırma, araştırmacılar tarafından makine öğrenmesi, örüntü tanıma (pattern recognition) ve istatistik çalışmalarında kullanılmaktadır. Son dönemlerde Veri madenciliği araştırmacıları tarafından disklerde büyük yer kaplayan veriler üzerinde kullanılmaktadır[16].

Sınıflandırma fonksiyonları, denetimli öğrenme fonksiyonlarıdır ve sınıflandırma modelinin öğrenme kümesi üzerinde sınanması uygun olmayacağından dolayı bir sınama kümesi kullanılmalıdır[17].

Genel olarak sınıflandırma iki aşamadan oluşmaktadır. İlk aşamada veri sınıflarının önceden belirlenmiş bir kümesini tanımlayan bir sınıflandırıcı inşa edilir. Bu aşama, sınıflandırma algoritmasının veri tabanında bulunan veri grupları ve grupları ilgilendiren diğer sınıf başlıklarından öğrenerek sınıflandırıcıyı oluşturduğu öğrenme aşamasıdır.

İkinci aşamada ise model öngörü için kullanılır. Öncelikle sınıflandırıcının öngörü doğruluğu (predictive accuracy) kestirilir. Öğrenme kümesi (training set) sınıflandırıcının doğruluğunu ölçmek için kullanılırsa, sınıflandırıcı veriye uyum göstermeye meyilli olduğundan bu tahmin oldukça iyimser olacaktır. Bu nedenle sınıf başlıkları arasındaki ilişkiyi test etmek için bir sınama kümesi (test set) kullanılır. Bu

[15] Alpaydin, E. 2014. *Introduction to machine learning*. MIT press.
[16] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.
[17] Çakır, Ö. 2008. Veri Madenciliğinde Sınıflandırma Yöntemlerinin Karşılaştırılması "Bankacılık Müşteri Veri Tabanı Üzerinde Bir Uygulama". In *Sosyal Bilimler Enstitüsü*. Marmara Üniversitesi.

sınama kümesi tüm veri içerisinden rastgele seçilir. Sınıflandırıcının doğruluğu ise bu sınama kümesi üzerinde sınıflandırıcının doğru sınıflandırdığı yüzde ile ifade edilir. Sınıflandırıcı doğruluğu kabul edilebilir düzeyde olarak değerlendirilirse sınıflandırıcı, gelecekte elde edilecek sınıf başlığı bilinmeyen verilerin sınıflandırılması için kullanılabilir. Örneğin geçmişteki kredi işlemleri kullanılarak gelecekteki kredi başvurularının kabul edilmesi veya red edilmesine yönelik karar vermede sınıflandırma uygulanabilir[18].

Sınıflandırma modelinin değerlendirilmesi için hata matrisleri, sınıflandırma doğruluk oranı, kappa istatistiği, ortalama mutlak hata, mutlak bağıl hata, bağıl hataların karekökü gibi hesaplamalardan faydalanılabilinir[19].

1.3.4.1. Karar Ağaçları

Karar ağacı, ağaç yapısında bir akış şeması şeklindedir. Bu şema, en üstte yer alan kök düğüm (root node) ile başlar ve daha alttaki her bir değişken için yapılan sınamayı gösteren iç düğümlere (internal nodes) ve en altta bulunan öngörünün sonucunu gösteren yapraklara (leaf nodes) bağlanır. Her bir değişken için sağlanan ayrımları gösteren ve bağlantıyı sağlayan elemanlara dallar (branches) denir.

Bir başka deyişle karar ağaçları, bir kategori ya da değerle sonuçlanan kural serilerini betimleme yöntemidir. Kategorik bağımlı değişkenin kullanıldığı karar ağaçlarına sınıflandırma ağaçları (classification trees), sürekli bağımlı değişken kullanılanlara ise regresyon ağaçları (regression trees) adı verilmektedir[20].

Karar ağaçlarının sınıflandırmada kullanımı ise elde bulunan sınıfları bilinmeyen ilgili veri grupları için kök düğümden yapraklara kadar uzanacak şekilde sınıfların öngörülmesi ile gerçekleştirilir. Karar ağaçları kolaylıkla sınıflandırma kurallarına dönüştürülerek ifade edilebilir.

---

[18] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

[19] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.

[20] Çakır, Ö. 2008. Veri Madenciliğinde Sınıflandırma Yöntemlerinin Karşılaştırılması "Bankacılık Müşteri Veri Tabanı Üzerinde Bir Uygulama". In *Sosyal Bilimler Enstitüsü*. Marmara Üniversitesi.

Karar ağaçları çok boyutlu veriler ile başa çıkabilir. Karar ağaçlarında elde edilen bilginin sunumunun ağaç şeklinde olması anlaşılabilirliği ve sezgisel olarak benimsenmeyi kolaylaştırır. Karar ağaçlarının öğrenme ve sınıflandırma adımları oldukça basit ve hızlıdır. Genel olarak karar ağaçları iyi düzeyde doğruluk oranına sahiptir. Tıp, üretim, finansal analiz, astronomi ve moleküler biyoloji gibi bir çok alanda karar ağaçları ile sınıflandırmadan yararlanılmaktadır[21].

Karar ağaçları, gerçek dünyadaki bir çok veri madenciliği uygulamalarındaki sınıflandırma problemlerinde güçlü bir çözüm olarak uygulanmaktadır[22].

Karar ağacı uygulamalarının başlangıç aşamasında bir takım ön çalışmaların yapılmasında fayda görülmektedir. İlk olarak, karar ağaçları denetimli öğrenme fonksiyonunu kullandıklarından, yüksek öngörü başarısının elde edilmesi için veri kümesinin yeterli büyüklükte ve değişkenlikte olması gerekmektedir. Bununla birlikte, bağımlı değişkenin dağılımının bilinmesi, asimetri ve aykırı değer (outliers) sorunlarının tespiti, değişkenin sıklarının belirlenmesi gibi tanımlayıcı analizlerin gerçekleştirilmesi hem model seçimi hem de öngörü başarısı konusunda olumlu katkılar doğuracaktır. Ve son olarak, en yüksek başarı için en doğru algoritmanın kullanılması gereğinden hareketle, hedefe ve veri yapısına uygun algoritmanın seçimi önem kazanacaktır. En uygun algoritmanın seçilmesinde ise düğümlerde kullanılan bölümleme kriterleri ve ağaç boyutunun küçültülmesine dayanan budama (pruning) yöntemi dikkate alınması gereken iki temel unsur olarak öne çıkmaktadır.

Başlıca karar ağacı ve karar türetme algoritmaları arasında AID (Automatic Interaction Detector), CHAID (ChiSquared Automatic Interaction Detection), CRT (Classification Regression Trees), ID3 yer almaktadır[23].

1.3.4.2. C4.5 Algoritması

C4.5 ve halefi C5.0 algoritmaları, 1970'lerde başlayan 20 yılı aşkın bir periyodun sonunda Ross Quinlan[24] tarafından ortaya çıkarılmış olup, ID3 algoritmasının

[21] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.
[22] Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.
[23] Çakır, Ö. 2008. Veri Madenciliğinde Sınıflandırma Yöntemlerinin Karşılaştırılması "Bankacılık Müşteri Veri Tabanı Üzerinde Bir Uygulama". In *Sosyal Bilimler Enstitüsü*. Marmara Üniversitesi.

geliştirilmiş versiyonlarıdır. Bu algoritmaların ilk versiyonları kategorik veri türleri ile sınırlı olmakla birlikte, sürekli veri türleri üzerinde de uygulama olanağı sağlamaktadır. Weka programında ise C4.5 algoritmasının karşılığı J48 olarak adlandırılmıştır[25].

C4.5 algoritması çoklu bölümlemeye dayanmaktadır ve veri kümelerinin önceden belirlenmiş kategorilere atanması amacıyla, kategorileri tanımlayan örüntüleri keşfetme, bu örüntüleri sınıflandırıcılara dönüştürme ve bu sınıflandırıcıları kullanarak öngörülerde bulunma esasına dayalı olarak kurgulanmıştır.

C4.5 algoritması, yüzlerce değişken ile yüz binlerce kayıt barındıran veri kümeleri üzerinde uygulanabilmektedir. Hem karar ağacı hem de karar kuralları formunda sonuç üretebildiği için yorumlanması, istatistik ya da yapay öğrenme konusunda uzmanlık gerektirmediği için kullanımı son derece kolay bir sınıflandırma aracıdır.

C4.5 uygulaması, bir çok sınıflandırma uygulamasında olduğu gibi, veri kümesinin öğrenme ve sınama amaçlı olarak bölümlenmesine olanak tanımaktadır. Bu sayede, hem inşa edilen sınıflandırıcının öğrenme kümesi üzerindeki kesinliğini sınamak hem de algoritmanın öğrenme kümesi üzerinde aşırı uyum göstermesini engellemek mümkün olmaktadır[26].

C4.5 alt ağaçlarda budama ile ilgili kararlar almak için hata oranı tahminlerini kullanması nedeniyle maliyet karmaşıklığı yöntemine benzeyen ve kötümser budama olarak adlandırılan bir yöntem kullanır. Ancak, kötümser budama bir budama kümesinin kullanımını gerektirmez. Bunun yerine, hata oranlarının kestirimini yapmak için eğitim kümesini kullanır. Eğitim kümesine dayanan bir doğruluk ya da hata kestiriminin aşırı derecede iyimser olmasından dolayı aşırı derecede yanlıdır. Bu nedenle, kötümser budama yöntemi, oluşan yanlılıkla başa çıkmak için bu sıkıntılı durumu öğrenme kümesine ekleyerek elde edilen hata oranlarını düzeltir[27].

---

[24] Quinlan, J. (1993) C4. 5: Programs for Machine Learning Morgan Kaufmann San Mateo.

[25] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.

[26] Çakır, Ö. 2008. Veri Madenciliğinde Sınıflandırma Yöntemlerinin Karşılaştırılması "Bankacılık Müşteri Veri Tabanı Üzerinde Bir Uygulama". In *Sosyal Bilimler Enstitüsü*. Marmara Üniversitesi.

[27] Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

C4.5 algoritması kök düğümden yaprak düğüme uzanan akışla karar kuralları şeklinde de ifade edilebilmektedir. Özellikle çok büyük karar ağaçları yerine karar kurallarını kullanmak insan algısına daha uygun düşmekte ancak veri sayısı arttıkça işlem süresi uzamaktadır[28].

Çalışmada kullanılan C4.5 karar ağacı sonuçlarının değerlendirilmesinde, Kappa istatistiği ve ortalama mutlak hata hesaplamalarından faydalanılmıştır.

Kappa istatistiği, bir veri kümesinin gözlenen ve beklenen durumları arasındaki uyumu (agreement) ölçmek için kullanılır[29]. Hedef değişkeni sıfır ve bir değerlerini alan ikili kategorik değişken olan bir sınıflandırma çalışmasına ait hata matrisi Tablo 1'de gösterilmiştir.

Tablo 1. İkili Kategorik Değişken İçin Hata Matrisi

	1	0	Toplam
1	A	b	m_1
0	C	d	m_0
Toplam	n_1	n_0	N

Tablo 1'de esas köşegen üzerinde yer alan a ve d değerleri iki değerlendiriciye (observer) ait uyumlu (agree) ölçümlerin sayısını gösterir. Esas köşegen haricindeki b ve c değerleri ise uyumsuz (disagree) ölçümlerin sayısını göstermektedir[30].

İki değerlendirici arasındaki uyumun değerlendirilmesi için, 1960 yılında Cohen tarafından kappa istatistiği geliştirilmiştir ve kappa istatistiği

$$\kappa = \frac{P_0 - P_e}{1 - P_e}$$

şeklinde ifade edilir[31]. Burada, P_0 toplam uyum olasılığı olup,

[28] Çakır, Ö. 2008. Veri Madenciliğinde Sınıflandırma Yöntemlerinin Karşılaştırılması "Bankacılık Müşteri Veri Tabanı Üzerinde Bir Uygulama". In *Sosyal Bilimler Enstitüsü*. Marmara Üniversitesi.
[29] Kanik, E. A., S. Erdoğan & G. O. Temel (2015) İki Sonuçlu Tanı Testlerinde İki Hekim Arasındaki Uyum İstatistiklerinin Prevalanstan Etkilenme Durumları+. *Turgut Özal Tıp Merkezi Dergisi,* 19**,** 153-158.
[30] Viera, A. J. & J. M. Garrett (2005) Understanding interobserver agreement: the kappa statistic. *Fam Med,* 37**,** 360-363.
[31] Cohen, J. (1960) A coefficient of agreement for nominal scales. *Educational and psychological measurement,* 20**,** 37-46.

$$P_0 = \frac{a+d}{n}$$

şeklinde hesaplanır. P_e ise şansa bağlı uyum olasılığını ifade etmekte olup,

$$P_e = \left(\frac{n_1}{n}\right) * \left(\frac{m_1}{n}\right) + \left(\frac{n_0}{n}\right) * \left(\frac{m_0}{n}\right)$$

şeklinde hesaplanmaktadır.

Elde edilen κ değerlerini yorumlamak için Landis ve Koch (1977), Tablo 2'de gösterilen yorumu önermiştir[32].

Tablo 2. Kappa Değerlerinin Yorumu

κ değeri	**Yorum**
<0	Şansa bağlı uyumdan daha az uyum
0,01-0,20	Önemsiz düzeyde uyum
0,21-0,40	Zayıf düzeyde uyum
0,41-0,60	Orta düzeyde uyum
0,61-0,80	İyi düzeyde uyum
0,81-1,00	Çok iyi düzeyde uyum

Öngörünün başarısını ölçmek için kullanılan alternatif yöntemlerden biri ortalama mutlak hata (mean absolute error - MAE) dır. Örnek sayısı n, öngörülen değerler $p_1, p_2, \dots, p_n$ ve gözlenen değerler $a_1, a_2, \dots, a_n$ olmak üzere, ortalama mutlak hata,

$$Ortalama\ Mutlak\ Hata = \frac{|p_1 - a_1| + \cdots + |p_n - a_n|}{n}$$

şeklinde hesaplanır[33]. Hedef değişkenin nominal olduğu durumlarda, her örnek satırı için satıra ait sınıflandırma öngörüsünün doğruluk olasılığı üzerinden hata değeri hesaplanır ve tüm satırların hata değerinin ortalaması alınarak ortalama mutlak hata değerine ulaşılır[34].

[32] Landis, J. R. & G. G. Koch (1977) The measurement of observer agreement for categorical data. *biometrics*, 159-174.
[33] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.
[34] WekaForum. Mean absolute error in classification. http://weka.8497.n7.nabble.com/Mean-absolute-error-in-classification-td9440.html. [Erişim 20.02.2018].

1.4. Weka

Weka, Yeni Zelanda'daki Waikato Üniversitesinde geliştirilmiştir. Sistem Java ile yazılmıştır ve Genel Kamu Lisansı (GNU) şartları altında dağıtılmıştır. Program Linux, Windows ve Macintosh işletim sistemlerinde ve hatta cep bilgisayarında test edilmiş ve hemen hemen her platformda çalışmaktadır. Verilen herhangi bir veri kümesinde öğrenme planlarının işlem öncesi hazırlık aşaması ve işlem sonrası değerlendirme aşaması için gerekli metotlarla birlikte bir çok farklı öğrenme algoritmasına yönelik bir arabirim sağlar.

Weka, temel veri madenciliği problemleri ile ilgili regresyon, sınıflandırma, kümeleme, birliktelik kuralı ve değişken seçimi gibi fonksiyonları icra etmeye yarayan araçlar içerir[35].

[35] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.

2. DEĞİŞKEN SEÇİMİ

Teoride daha fazla değişken ile sınıflar arasında daha fazla ayrım yapma gücü elde edilebilecekmiş gibi görünse de, pratikte sınırlı boyuttaki eğitim verisi ve aşırı fazla değişken adedi öğrenme sürecini yavaşlatmanın yanında öğrenme algoritması, alakasız değişkenler nedeni ile sınıflandırıcının eğitim verisine fazlasıyla uyumlu olan sonuçlar üretmesine neden olabilir[36].

Veri madenciliği süreçleri öncesinde veri tabanı, veri ambarı ve diğer ilişkili olabilecek tüm kaynaklardan veriler bir araya getirilmeye çalışılır. Ancak üzerinde çalışılacak veri büyüdükçe yapılmak istenen çalışma için gereken kaynak gereksinimleri (bilgisayar sistemleri, zaman gibi) de veri ile doğru orantılı olarak artmaktadır. Gerçek çalışma alanlarında kaynaklar hiçbir zaman sınırsız olmadığından, ortaya konulmak istenen çalışmanın daha sağlıklı şekilde sonuçlandırılabilmesi için veri madenciliği süreci öncesinde değişken seçimi yapılması faydalı olabilmektedir.

Veri madenciliği süreçlerinin öncesinde ek olarak bir değişken seçim sürecinin uygulanması üç farklı şekilde değerlendirilebilir[37];

- **Hesaplama Süresi (Computing Time)** : Veri madenciliği süreci öncesinde gereksiz değişkenlerin ayıklanması, veri madenciliği sürecindeki işlem süresini, işlemci ve bellek kullanımını düşürecektir. Ancak değişken seçimi için çok fazla zaman harcamak, her zaman makul bir yaklaşım olmayabilir.
- **Öngörü Doğruluğu (Predictive Accuracy) :** Alakasız değişkenlerin, oluşturulan süreci yanlış yönlendirmesi söz konusu olabilmektedir. Sadece ilgili değişkenlerin sürece dahil edilmesi ile ortaya konulacak modelin doğruluğunun artması beklenir. Eğer öngörü doğruluğunu arttırmak asıl amaç ise değişken seçim süreci için uzun süre harcamak zaman zaman kabul edilebilir. Değişken seçimi sonrasında oluşturulan model daha basit ise öngörü doğruluğunda meydana gelebilecek ufak kayıplar göz ardı edilebilir.

[36] Yu, L. & H. Liu (2004) Efficient feature selection via analysis of relevance and redundancy. *Journal of machine learning research,* 5**,** 1205-1224.
[37] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.

- **Sonuçların Sunumu:** Bir çok Veri madenciliği algoritması aç gözlü davranmaya meyilli olduğundan ilgisiz değişkenleri de modele ekleyerek karmaşık sonuçların oluşmasına neden olabilmektedir. Sadece ilgili değişkenlerden oluşturulacak bir model, daha sade sonuçlar üreterek daha net bir öğrenme sürecinin gerçekleşmesini sağlayacaktır.

Bu üç değerlendirme kriterinin aynı anda sağlanabilmesi ideal durumdur. Ancak gerçek veri ile çalışılırken her zaman ideal çözüm elde edilemeyebilir. Çalışmanın amacı ve öncelikleri göz önünde bulundurularak bu kriterlerin bazılarından belirli ölçülerde feragat edilebilir[38].

2.1. Değişken Seçimi Bakış Açısı

İlk olarak, değişken seçimi genellikle algoritmalarla ele alınabilecek en uygun alt kümeyi arama problemi olarak düşünülür.

İkinci olarak, değerlendirme kriterleri incelenir ve onların özellikleri analiz edilir. Farklı ölçü tipleri, kendilerine ait farklı uygulamalara sahiptir.

Üçüncüsü, değişken seçimi, esas olarak değişkenlerin tek değişkenli ve çok değişkenli özellikler olarak değerlendirilmesi şeklinde gözden geçirilir. Değişken seçimi sırasında, tek değişkenli değerlendirme ile, seçilmeyen değişkenler arasındaki en iyi değişkenin seçilen değişkenler kümesine eklenmesini veya seçilen değişkenler kümesinden en az öneme sahip değişkeni silmeyi dikkate alırken, çok değişkenli değerlendirme en uygun alt kümeyi bulmak için tek bir değişken yerine bir değişken alt kümesini değerlendirmeyi dikkate alır.

Sonuncusu, değişken seçimi, sınıflamada değişkenin nasıl dahil edileceğine dayalı olarak incelenir.

Bir değişken seçimi uygulaması, bir sınıflandırıcı için ön işlemci veya performans arttırıcı olarak kullanılabilir. Diğer bir ifadeyle, sınıflandırıcı için bir filtre veya güçlendirici olarak görülebilir.

[38] Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.

2.2. Sınıflandırma İçin Değişken Seçimi

Görülmeyen bir örüntüye ait sınıfı öngörebilen bir sınıflandırıcıyı harekete geçirirken, kesin olarak kestirilebilen ve aynı zamanda yapısı mümkün olduğu kadar basit olan bir sınıflandırıcıya sahip olmak amaçlanmaktadır.

Verilen bir problem için, potansiyel değişkenlerin sayısı oldukça büyük olabilir. Bunun nedeni ise;

- Toplanan veriler sadece veri madenciliği gibi belirli bir görev için değildir. Örneğin, veriler çoğu zaman muhasebecilik gibi belirli bir görev için veya yasa gereği toplanır. Büyük ihtimalle, çok sayıda gereksiz ve alakasız değişken vardır.
- Bazen değişkenlerin belirli bir görev için tasarlandığı bilinse bile, ilgili değişkenler genellikle bilinmez. Bu, aramanın doğası gereği böyledir. Çalışma alanı ve problem hakkında daha fazla şey bilmek istediğimiz için deneyler yapıyor, veriler topluyoruz ve genellikle ihtiyaç duyulan değişkenler (veya çalışma alanı ve problemi tanımlayan boyutlar) hakkında kesin bir fikre sahip değiliz. Bu nedenle, bazıları uzaktan alakalı olsalar bile düşünebildiğimiz kadar çok aday değişken sunmak zorundayız. Dolayısıyla, bunlardan bazıları kaçınılmaz bir şekilde gereksiz ve alakasızdır. Toplanan verileri inceledikten sonra, yalnızca hangi değişkenlerin konu ile ilgili olduğunu biliriz.
- Bir görev, verilerin farklı kaynaklardan gelmesini gerektirebilir. Her kaynaktan toplanan veriler orta büyüklükteyse, bunların birleşimi büyük olabilir. Ancak, ilgili değişkenleri biliyorsak, problemin üzerine gidebiliriz. Genel olarak hangi özelliğin konu ile ilgili olduğu önceden bilinemeyebilir. Konu ile alakası olmayan veya gereksiz değişkenler sınıflama algoritmaları üzerinde aşağıdaki olumsuz etkilere sebep olabilir:
  (1) Farklı sınıflardan elde edilen örüntüler arasındaki istatistiki değişkenliğin sağlanmasına ihtiyaç duyduğumuz için çalışmamızda daha fazla değişken kullanmak isteriz. Çoğu zaman daha fazla değişken ile elde edeceğimiz bilgi için kullanılacak olan sınıflandırma algoritması daha uzun zaman alır.
  (2) Gereksiz ve alakasız değişkenler öğrenme algoritmalarını yanıltabilir ya da verilerin oldukça uyumlu olmasına sebep olur.
  (3) Ek olarak, gereksiz ve alakasız verilerin varlığıyla, elde edilen sınıflandırıcının daha karmaşık olması daha mümkündür.

Aslında, en az ölçüm denemesi ile en yüksek performans elde edebilmek için uygulamada sadece alakalı değişkenlerin kullanılması . Değişken seçimi;

1. Sınıflandırma algoritmasının daha hızlı öğrenilebilmesi için daha az değişken kullanılması,
2. Sınıflandırıcının verilerden daha iyi genelleme yapabilmesi için daha yüksek doğruluk,
3. Anlamak daha kolay olsun diye daha basit sonuçlar,
4. Veri toplamanın bir sonraki aşamasında, mümkünse gereksiz ve alakasız değişkenlerin kaldırılarak tasarruf yapılabilmesi için daha az değişken elde edilmesi,

amaçlanarak yapılır[39].

2.3. Sonuçlarına Göre Değişken Seçim Algoritmaları

Bu bölümde değişken seçim algoritmalarının çalışması sonucunda elde edilen çıktılar incelenecektir.

2.3.1. Değişken Sıralama Algoritmaları

Değişken seçim algoritmalarının değişken sıralama kategorisinde doğruluk, tutarlılık, bilgi, mesafe ve bağımlılık gibi ölçülere göre değerlendirilip sıralanmış bir değişken listesinin oluşturulması beklenir. Bir diğer açıdan, bu tipteki algoritmalar, bir alt kümeyi vermek yerine gerekli değişkenleri diğerleri ile karşılaştırmaktadırlar.

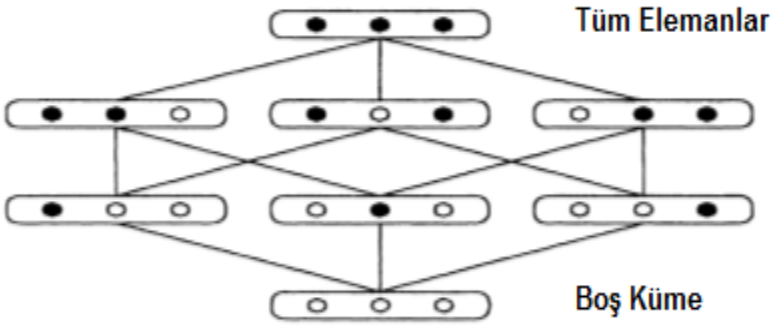
Bazı araştırmacıların, değişken seçimi için bu algoritmaları kullanmalarına karşın, bu şekilde bir kullanım makul olmayabilir ve en küçük değişken alt kümesinin elde edilmesinin amaçlandığı durumlar için önerilmemiştir.

Biyolojide çözümüne yoğun çaba harcanan konulardan biri olan hastalıklı ve sağlıklı bireylerin sahip oldukları genleri incelemede değişken sıralama algoritmaları kullanılabilir[40].

[39] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.
[40] Guyon, I. & A. Elisseeff (2003) An introduction to variable and feature selection. *Journal of machine learning research,* 3**,** 1157-1182.

2.3.2. En Küçük Alt Küme Algoritmaları

İlgili değişkenlerin bilinmemesi sık karşılaşılan bir durumdur. Bu amaca ulaşabilmek için değişkenleri sadece sıralamaktan ibaret olan algoritmalardan farklı algoritmalara ihtiyaç bulunmaktadır. Bu noktada En küçük alt küme algoritmaları kullanılmaktadır. Bu algoritmalar, ilgili değişkenleri barındıran en küçük değişken alt kümesini döndürür. Kümede yer almayan değişkenler ilgisiz olanlardır.



Şekil 2. Arama Yönleri [41]

2.4. Arama Problemi

Değişken seçimi, değişken alt küme uzayındaki (search space) her durumla olası değişkenlere ait alt kümenin belirlendiği bir arama problemi olarak görülebilir. (A_1, A_2, A_3) şeklinde üç değişkenli bir veri kümesini ele alalım. Burada kümeye karşılık gelen iki değerli bir dizi tanımlanabilir ve bu dizide yer alan 1 elemanı, dizide yer alan özelliğin seçildiği anlamına gelir. Örneğin, (1,1,1) üç değişkenin de seçildiği anlamına gelirken, (1,0,0) sadece A_1 değişkeninin seçildiği anlamına gelir.

N, bir veri kümesindeki değişkenlerin sayısı olmak üzere, bu veri kümesinin toplam 2^N adet alt kümesi vardır. Üç değişken söz konusuyken, Şekil 1 de görüldüğü gibi sekiz durum (alt küme) vardır. İki uçta da, iki ekstrem alt küme bulunur. Bunlardan birinde üç değişken de seçilirken, diğerinde hiçbir değişken yer almaz yani boş kümedir. Daha gerçekçi bir durum söz konusu olduğunda en uygun alt küme bu iki uç arasında bir yerde yer alır. Bu durumda, aramaya nereden başlanacağına karar verme bir sorun oluşturmaktadır.

[41] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.

Bu sorun, değişken alt küme uzayı küçük olduğunda önemsiz olur çünkü ya başlangıç ya da bitiş seçilir ve arama hızlıca tamamlanır. Ancak, genellikle bu uzay küçük değildir. Bu durum, özellikle aramanın başlangıç noktası hakkındaki sorunun yerinde olduğunu gösterir ve uygun arama stratejilerine karar verilmesi gerektiğini ortaya koyar [42,43].

2.5. Arama Yönleri

En uygun değişken alt kümesinin, değişken alt kümelerine ait uzayda olduğu bilinmesine rağmen herhangi bir kısıtlama hakkında önceden bilgi sahibi olmadığımız varsayılarak, aramayı hangi uçtan başlattığımızın bir önemi olmayacaktır (boş kümeden ya da tüm değişkenleri içeren kümeden). Yani, çok sayıda sorun için, ortalama olarak, bir yönde arama yapan bir yöntem diğer yönde arama yapan bir yöntem kadar hızlı bir şekilde en uygun alt kümeyi bulacaktır. Arama yönleri, değişken alt kümesinin oluşturulması ile yakından ilgilidir. Arama yönleri genel olarak üç şekilde gruplanabilir;

(1) İleri Yönlü: Boş bir değişken alt kümesi ile başlar. Arama başladığında, değişkenler bu kümeye birer birer eklenir. Her seferinde, seçilmeyen değişkenler arasındaki en iyi değişken, bazı kriterlere göre seçilir.

(2) Geri Yönlü: Arama süreci, tüm değişkenleri içeren küme ile başlar. Değişkenler birer birer kaldırılır. Her seferinde en az öneme sahip olan değişken bazı kriterlere göre kaldırılır. Dolayısıyla, değişken kümesi tek bir değişken kalana kadar daralmaya devam eder.

(3) İki yönlü: Arama süreci her iki yönde de başlar, yani iki arama eş zamanlı olarak devam eder ve bir arama sürecin ortasına gelmeden en iyi değişken alt kümesini bulduğu zaman ya da her iki arama da sürecin ortasına geldiği zaman arama sona erer[44].

[42] Dash, M., H. Liu & H. Motoda. 2000. Consistency based feature selection. In *Pacific-Asia conference on knowledge discovery and data mining*, 98-109. Springer.
[43] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.
[44] Liu, H. & H. Motoda. 1998a. *Feature extraction, construction and selection: A data mining perspective*. Springer Science & Business Media.

2.6. Arama Yöntemleri

Arama yöntemleri, tüm değişken uzayından iyi bir değişken alt kümesi bulana kadar gezinir. Yöntemin başarısı, değişken alt kümesi değerlendirme algoritması tarafından belirlenir. Aşağıda anlatılacak tüm arama yöntemleri, Weka içerisinden seçilebilir.

2.6.1. Sıralayıcı Arama

Sıralayıcı arama (Ranker) yöntemleri, bir değişken alt kümesi üretmek yerine değişkenlerin bir sıralaması şeklinde bir çıktı verdiklerinden dolayı önceleri bir arama yöntemi olarak kabul edilmezdi. Bu yöntemler, her bir değişken için ayrı ayrı skor üreterek önem sırasına göre sıralanmasını sağlarlar. Bu yöntemlerden faydalanılarak bir değişken alt kümesi seçmek istenirse başka bir değişken alt kümesi seçme algoritmasına ihtiyaç duyulmaktadır.

2.6.2. Tam Arama

Tam arama (Complete Search), boş kümeden başlayarak tüm değişkenleri içeren kümeye kadar tüm değişken evrensel kümesi tarayarak en iyi değişken alt kümesinin bulunmasını sağlar. Diğer algoritmalara nazaran daha fazla işlem yükü gerektirir.

2.6.3. Buluşsal Arama

Tam arama yöntemi çok fazla zaman ve kaynak tüketerek en iyi çözümü bulabilmektedir. Alternatif olarak buluşsal yöntemler, daha hızlı çözüm üretirler. Tam tarama ile elde edilebilecek olan en iyi çözümü elde etmeyi garanti etmezler. Hızlı şekilde en iyi çözüme yakın iyi bir çözüm elde edilmek isteniyorsa buluşsal arama kullanılabilir.

Buluşsal (heuristic) bir arama yöntemi olan BestFirst algoritması, başlangıç kümesinden itibaren her arama basamağında elde edilen en iyi sonucu seçer ve sonraki basamağa ilerler. Örneğin boş kümeden başlayarak ileri yönlü yapılan bir aramada, bir elemanlı tüm alt kümeleri inceleyerek en iyisini seçer. Bir sonraki basamakta ise önceki basamakta seçilen bir elemanlı alt kümeye diğer elamanları ekleyerek iki elemanlı alt kümeler oluşturur ve bu kümeleri inceleyerek en iyisini seçerek ilerler. Hangi değişken alt kümesinin nihai olarak seçileceğine ise bir optimizasyon yöntemi olan tepe tırmanışı yöntemini kullanarak karar verir.

Sonuç alınmadan önce kaç adet iyileştirilmemiş düğüm barındırması istendiği seçilebilir. Boş kümeden başlayarak ileri yönlü, tüm değişkenler kümesinden başlayarak geri yönlü veya bir değişken listesi ile sıralanmış şekilde aradan bir noktadan başlayarak, iki yönden birine doğru değişken ekleyerek ve çıkararak ilerleyebilir.

2.6.4. Sıralı Arama

Sıralı Arama (Rank Search) içerisinde bir adet sıralayıcı (ranker) değişken değerlendirme algoritması seçilir. Bu seçilen algoritma değişkenler için bir sıralama oluşturur. Oluşturulan bu sıralama, seçilen bir değişken alt kümesi seçme algoritması tarafından kullanılarak sıralamadaki en iyi, akabinde ikinci en iyi gelecek şekilde ilerleyen bir arama süreci izlenir ve sürecin sonunda değişken alt kümesi seçim algoritmasının kriterlerine göre bir değişken alt kümesi seçilir[45].

2.6.5. Rastgele Arama

Önceki arama yöntemlerinin tamamı aynı sırayı takip ederek çalışan ve ilk çalıştırmanın akabinde tekrar çalıştığında ilk sonuç ile aynı sonucu üreten yöntemlerdir. Rastgele arama (Nondeterministic Search) ise belirli bir sırayı takip etmeksizin çalışır ve her çalıştırılışında farklı bir sonuç üretir[46]. Bu çalışma prensibinden dolayı iyi bir sonuç üretebilir ancak en iyi sonucu ıskalama ihtimali kabul edilerek uygulanır[47].

2.7. Değişken Seçim Yöntemlerinde Değerlendirme Ölçüleri

Sınıflandırma algoritmalarında kullanılmak üzere değişken alt kümesi seçme yöntemleri filtreleyici ve sarmalayıcı olmak üzere iki farklı grupta ele alınmıştır. Sarmalayıcı yöntemler değerlendirme ölçüsü olarak sınıflandırma hata oranını kullanırken, filtreleyici modeller ise dolaylı yollardan ölçüm yapmayı sağlayan sınıflandırma doğruluğundan farklı bir değerlendirme ölçüsü kullanmaktadırlar[48].

---

[45] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.
[46] Dash, M. & H. Liu (1997) Feature selection for classification. *Intelligent data analysis,* 1**,** 131-156.
[47] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.
[48] Blum, A. L. & P. Langley (1997) Selection of relevant features and examples in machine learning. *Artificial intelligence,* 97**,** 245-271.

Son şekli ile ölçü değerlendirme fonksiyonları aşağıdaki gibi beş farklı başlık altında incelenmektedir;

- Mesafe (Distance)
- Bilgi (Information)
- Korelasyon (Correlation)
- Tutarlılık (Consistency)
- Sınıflandırma hata oranı (Prediction error rate)

Mesafe Ölçüleri: Hamming ve Öklid gibi uzaklık fonksiyonlarından faydalanılarak hesaplanan ölçülerdir. İki değişken arasındaki uzaklık sıfır ise iki değişkenden biri ayırt edilemez. Kullanılan veri kümesi içerisinde ilgisiz çok fazla değişken mevcut ise mesafe ölçüleri yanıltıcı olabilir[49].

Bilgi Ölçüleri: Genel olarak bir değişken üzerinden entropi yardımı ile bilgi kazanımının hesaplanması şeklindedir. Değişkenler kendi aralarında karşılaştırılarak bilgi kazanımı fazla olanı diğerine tercih edilir[50].

Korelasyon Ölçüleri: Hedef değişken ile korelasyonu yüksek olan bir açıklayıcı değişken, hedef değişken ile görece daha düşük korelasyona sahip olan değişkene tercih edilir[51].

Korelasyon ölçüsü kullanılan yaklaşımlar için, doğrusal korelasyon kullanılanlar ve bilgi teorisinden faydalananlar olmak üzere iki farklı kullanım şekli mevcuttur. Gereksiz değişkenlerin belirlenmesi konusunda doğrusal korelasyonun oldukça güvenilir olmasına karşın veri setinde bulunan değişkenlerin doğrusal veya sürekli olmaması gibi durumlardan dolayı bilgi teorisinden ve simetrik belirsizlikten faydalanılarak korelasyon hesaplamalarının yapılması daha uygun olabilmektedir[52].

Tutarlılık: Bu değerlendirme ölçülerinin diğerlerinden farklı yönleri bulunmaktadır. Diğer yöntemler öğrenme kümesine (training set) aşırı derece bağımlıdır ve en küçük değişken alt

[49] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.
[50] Ibid.
[51] Dash, M. & H. Liu (2003) Consistency-based search in feature selection. *Artificial intelligence,* 151**,** 155-176.
[52] Yu, L. & H. Liu. 2003. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, 856-863.

kümesini oluşturmaya yanlıdırlar. Bu yöntemde ise tutarlılık hipotezinin belirleyebildiği birkaç değişken, en küçük alt küme yanlılığına tercih edilir. Bu ölçüler, genelde tutarsızlık oranının kabul edilebilir oranda olduğu en az değişkenin seçilmesi şeklinde çalışırlar[53].

Tutarsızlık hesaplaması üç aşamada ifade edilebilir;

1) c_1 ve c_2 hedef değişkene ait iki farklı değer olmak üzere, (0 1 0 c_1) ve (0 1 0 c_2) gibi açıklayıcı değişkenlerin aynı değerleri almasına karşın hedef değişkenin farklı değer aldığı iki farklı örnek tutarsız olarak kabul edilir.

2) (0 1 0 c_1), (0 1 0 c_2), (0 1 0 c_3) şeklinde açıklayıcı değişkenlerin eşit değerler almasına karşın hedef değişkenin farklı değerler aldığı örneklerin bulunduğu bir veri kümesi bulunduğunu durumda sırasıyla hedef değişkenin c_1, c_2, c_3 değerlerinin tüm veri içerisinde tekrarlanma sayıları a_1 ,a_2 ,a_3 olarak ifade edilebilir. Bu durumda veri kümesinde bulunan örnek sayısı n ile ifade edilirse $n= a_1+a_2+a_3$ eşitliği sağlanır. Bu tekrarlanma sayılarının en büyüğünün a_3 olduğunu varsayalım. Bu durumda ise $n - a_3$ değerine tutarsızlık adedi adı verilir.

3) Örnek içerisinde bulunan tüm farklı açıklayıcı değişken kombinasyonu için tutarsızlık adetlerinin toplamının, tüm olası açıklayıcı değişken kombinasyonu adedine bölünmesi ile tutarsızlık oranı hesaplanır[54].

Yukarıdaki değerlendirme ölçülerinin kullanıldığı yöntemler, sınıflandırıcıdan bağımsız olarak değişken seçim yöntemi tarafından seçilmiş değişkenlerdir ve 'Filtreleyici Yöntemler' olarak bilinirler.

Sınıflandırıcı hata oranı ölçüleri: Yukarıdaki filtreleyici metotlardan farklı olarak sınıflandırıcı hata oranı ölçülerini kullanan yöntemler 'Sarmalayıcı Yöntemler' olarak adlandırılırlar ve bir adet sınıflandırıcı, değişken alt kümelerini değerlendirmek için kullanılır. Değişkenler seçildikten sonra sınıflandırıcı olarak kullanılarak sınıflar kestirilir. Öngörü

[53] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.

[54] Dash, M., H. Liu & H. Motoda. 2000. Consistency based feature selection. In *Pacific-Asia conference on knowledge discovery and data mining*, 98-109. Springer.

doğruluğu (prediction accuracy) çok yüksek seviyeye ulaşmasına karşın hesaplama yükü de doğru orantılı olarak diğer yöntemlere nazaran çok yüksek seviyelere ulaşmaktadır[55].

Sürekli verilerden oluşan bir veri kümesi için doğrusal korelasyon kullanılabilirken, kesikli verilerden oluşan veri kümelerinde ise simetrik belirsizlik, entropi, tutarlılık kullanılabilmektedir. Sürekli ve kesikli verileri birlikte barındıran veri kümeleri üzerinde çalışılabilmesi için sürekli verilerin kesikleştirilmesi ve kaba kümelerden faydalanılması gibi yaklaşımlar bulunmaktadır[56].

2.8. Değişken Seçim Modelleri

Değişken seçim algoritmaları genel olarak sarmalayıcı ve filtreleyici olarak iki ana grupta incelenmektedir.

2.8.1. Sarmalayıcı Modeller

Değişken seçim algoritmalarında performans ölçüsü olarak sınıflandırıcının doğruluğunun (accuracy) kullanılması basit bir yaklaşımdır. Bu durumda mümkün olan en yüksek sınıflandırıcı doğruluğunu elde etmeye yönelik bir sınıflandırıcı oluşturulmalıdır ve sınıflandırıcı için en uygun değişkenler seçilmelidir. Bu yöntem "Sarmalayıcı Model" olarak adlandırılır[57].

Sarmalayıcı yaklaşım, Kohavi ve John (1997) tarafından ortaya atılmıştır[58]. Bu yaklaşımda, değişken alt kümeleri tekrarlı şekilde veri kümesi olarak çalıştırılır. Tüm alt kümeler için performans değerlendirmesi yapılır ve en yüksek performans gösteren sonraki aşama için seçilir. Seçilen değişken alt kümesi arama sırasında kullanılmayan bir veri kümesi (sınama kümesi) üzerinde test edilir. Denetimli öğrenme algoritmaları bir test kümesi üzerinde sınıflandırma doğruluğunu en üst düzeye çıkarmayı amaçladığından değişken alt kümesi seçiminde doğruluk ölçüsü kullanılmış olur[59].

---

[55] Dash, M. & H. Liu (2003) Consistency-based search in feature selection. *Artificial intelligence,* 151**,** 155-176.
[56] Jiang, S.-y. & L.-x. Wang (2016) Efficient feature selection based on correlation measure between continuous and discrete features. *Information Processing Letters,* 116**,** 203-215.
[57] Liu, H. & H. Motoda. 1998a. *Feature extraction, construction and selection: A data mining perspective*. Springer Science & Business Media.
[58] Kohavi, R. & G. H. John (1997) Wrappers for feature subset selection. *Artificial intelligence,* 97**,** 273-324.
[59] Liu, H. & H. Motoda. 1998a. *Feature extraction, construction and selection: A data mining perspective*. Springer Science & Business Media.

Makine öğrenmesinde, yüksek öngörü doğruluğu elde etmek istenir. Problemimiz sınıflandırıcı doğruluğunun performansını, elimizdeki veriyi kullanarak nasıl arttıracağımızdır. Sınıflandırıcı performansını arttırmak için değişken seçimi yapmak tercihlerimizden biri iken, diğer tercihimiz sınıflandırıcıya daha kaliteli bir veri sunmaktır. Gürültüden arındırılmış bir veri üzerinde sarmalayıcı model kullanarak sınıflandırıcı doğruluğunu arttırabiliriz.

Öğrenme verileri üzerinden bir sınıflandırıcının tahmini doğruluğu, test verilerindeki doğruluğu göstermeyebilir. Buradaki asıl konu, sınıflandırma doğruluğunun nasıl daha iyi tahmin edileceği konusudur. Bu amaçla çapraz geçerlilik (cross validation) kullanılması makuldür. Bu da değişken seçim sürecini uzatmaktadır. Sarmalayıcı model, sınıflandırıcının öğrenilmesi ve değişkenlerin seçilmesini birlikte gerçekleştirdiğinden dolayı sınıflandırıcı doğruluğunu arttırmak için en iyi yol olarak görünmektedir. Doğru öngörüde bulunmanın zorluğuna rağmen asıl amacımız olan sınıflandırıcıyı iyileştirmek, doğruluğu iyileştirmekten daha fazlasıdır. Sarmalayıcı modelin kısıtlamalarından bir diğeri ise sınıflandırıcı seçimi ile ilgilidir. Hesaplama yüküne karşı eldeki imkanların zayıf olduğu durumlarda sarmalayıcı model kullanılmamalıdır[60].

Zhu ve arkadaşları tarafından Memetik alan üzerinde yapılan çalışmada, sarmalayıcı yöntem kullanılarak yapılan sınıflandırma çalışmaları diğer yöntemlerden faydalanılarak yapılan sınıflandırma çalışmalarından daha yüksek sınıflandırma doğruluğu elde etmiştir[61].

Zhou ve arkadaşları tarafından bioenformatik alanda sarmalayıcı yöntem üzerine yapılan çalışmada, en büyük bilgi katsayısından (maximal information coefficient) faydalanılarak oluşturulan sarmalayıcı yöntem ile klasik sarmalayıcı yöntemler ile elde edilen değişken alt kümesinden daha küçük değişken alt kümeleri elde edilerek klasik sarmalayıcı yöntemin sınıflandırma doğruluğuna yakın sonuçlar elde edilmiştir[62].

[60] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.

[61] Zhu, Z., Y.-S. Ong & M. Dash (2007) Wrapper–filter feature selection algorithm using a memetic framework. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* 37**,** 70-76.

[62] Ge, R., M. Zhou, Y. Luo, Q. Meng, G. Mai, D. Ma, G. Wang & F. Zhou (2016) McTwo: a two-step feature selection algorithm based on maximal information coefficient. *BMC bioinformatics,* 17**,** 142.

2.8.2. Filtreleyici Modeller

Sarmalayıcı modellerdeki iş yükünün fazlalığından ve uzun süre gereksiniminden dolayı farklı dolaylı yollardan performans ölçümleri üzerinde çalışmaların yapılmasına yönelinmiştir. Genelde uzaklık, bilgi, korelasyon gibi ölçüler değişken seçimi için kullanılmıştır. Bu modeller "Filtreleyici Model" olarak adlandırılır.

Bir filtreleyici model iki safhadan oluşmaktadır;

1- Sınıflandırıcıdan bağımsız olarak bilgi, uzaklık, korelasyon gibi bir ölçü ile değişken seçimi yapılan safha

2- Öğrenme verisinden (training data) öğrenilen sınıflandırıcının, seçilen değişkenler ile birlikte sınama kümesi üzerinde uygulandığı safha.

Filtreleyici model bazı niteliklere sahiptir;

1- Sınıflandırıcının belirli bir eğilimine dayanmak yerine verinin özünde bulunan özelliklere dayanır. Değişkenler farklı sınıflandırma tekniklerinde kullanılmak üzere seçilebilir.

2- Bilgi, uzaklık, korelasyon, tutarlılık ölçüleri genel olarak sınıflandırıcı doğruluğu ölçüsüne nazaran daha az çaba gerektirir. Böylece filtreleyici yöntemler daha hızlı şekilde ve zaman tasarrufu sağlayarak bir değişken alt kümesi oluşturabilirler.

3- Ölçülerin sadeliğinden ve zaman tasarrufu sayesinde sınıflandırıcının yapabileceğinden daha geniş veri kümelerine hakim olabilir. Bu durumda sınıflandırıcı geniş bir veri kümesinden direk öğrenilmemiş, boyut (dimension) yönünden bakıldığında da indirgenmiş boyutta bir veri kümesinden öğrenilmiş olur. Bunlarla birlikte, bir filtreleyici model tarafından seçilen değişkenlerin, öğrenme algoritmasının verinin eğilimlerinin tamamından faydalanmasına engel olması gibi bir tehlike söz konusudur[63].

[63] Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.

Büyük veri kümeleri ile çalışılırken daha az kaynak gereksiniminden dolayı filtreleyici yöntemlerin kullanılması daha avantajlı olmasına karşın, sarmalayıcı yöntemler ile sınıflandırma doğruluğunu yükseltme hedefine ulaşmak daha garantidir[64].

Filtreleyici modeller, elde edilebilecek en iyi değişken alt kümesini bulamama riskini barındırırken, sarmalayıcı yöntemler ise öğrenme kümesine aşırı bağımlı (over fitting) kalma riskini barındırır[65].

2.8.2.1. Bilgi Kazanımı Bazlı Değişken Seçimi

Bilgi kazanımı amaçlı değişken seçimi (IG), basit ve hızlı olduğundan oldukça tercih edilen bir değişken seçim algoritmasıdır. Bu algoritma ölçü olarak bilgi ölçüsünü kullanır. Her bir değişken için, değişkenin eklenmesinden önce ve sonra hesaplanan entropi değerlerinin farkı ile hesaplanan değere bilgi kazanımı adı verilir[66].

Weka'da bulunan InfoGainAttributeEval değişken seçim algoritması, değişkenleri değerlendirmeden önce kendi içerisinde bulunan denetimli kesikleştirme yöntemini kullanılarak, öğrenme verisinde (training data) bulunan sürekli değişkenleri kesikleştirerek tamamı kesikli veriden oluşan bir öğrenme verisini çalışma öncesinde kaydeder[67].

A değişkenler kümesini, C ise hedef değişkenin aldığı değerlerin kümesini göstermek üzere, Denklem (1) ve Denklem (2) yardımıyla hesaplanan entropi değerleri, Denklem (3)'de kullanılarak bilgi kazanımı değeri hesaplanır[68,69].

$$H(C) = -\sum_{c \in C} P(c)\ \log_2 P(c) \tag{1}$$

$$H(C|A) = -\sum_{a \in A} P(a) \sum_{c \in C} P(c|a)\ \log_2 P(c|a). \tag{2}$$

[64] Das, S. 2001. Filters, wrappers and a boosting-based hybrid for feature selection. In *ICML*, 74-81.
[65] Ladha, L. & T. Deepa (2011) Feature selection methods and algorithms. *International journal on computer science and engineering,* 3**,** 1787-1797.
[66] Abraham, R., J. B. Simha & S. S. Iyengar (2009) Effective Discretization and Hybrid feature selection using Naïve Bayesian classifier for Medical datamining. *International Journal of Computational Intelligence Research,* 5**,** 116-129.
[67] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.
[68] Shahbaz, M. B., X. Wang, A. Behnad & J. Samarabandu. 2016. On efficiency enhancement of the correlation-based feature selection for intrusion detection systems. In *Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016 IEEE 7th Annual*, 1-7. IEEE.
[69] Hall, M. A. & G. Holmes (2000) Benchmarking attribute selection techniques for data mining.

$$IG_i = H(C) - H(C|A_i) \quad (3)$$

Bilgi kazanımı bazlı değişken seçimi sıralayıcı bir değişken seçim yöntemi olarak kullanılabilir[70].

Yang ve arkadaşları tarafından yapılan metin kategorizasyonu çalışmasında, bilgi kazanımı bazı değişken seçiminin diğer alternatiflere göre daha başarılı sonuçlar elde ettiği ortaya konulmuştur[71]. Uğuz tarafından yapılan çalışmada ise metin kategorizasyonu üzerinde bilgi kazanımı bazlı değişken seçimi ile birlikte temel bileşenler analizi ve genetik algoritmadan faydalanılarak hibrit modeller denenmiş ve bilgi kazanımı bazlı değişken seçiminin temel bileşenler analizi ile birlikte kullanıldığı durumda daha iyi sonuçlar ürettiği gözlemlenmiştir[72].

Oreski D. ve arkadaşları tarafından yapılan çalışmada veri kıtlığı yaşandığı durumlar için bilgi kazanımı bazlı değişken seçiminin tercih edilmesi uygun bulunmuştur[73].

2.8.2.2. Kazanç Oranı Bazlı Değişken Seçimi

Kazanç oranı bazlı değişken seçimi (GR), bilgi kazanımının bir başka versiyonu olarak ifade edilebilir. Kazanılan bilginin sınıfın entropi değerine oranı olarak hesaplanmaktadır. (3) numaralı denkleme benzer olarak,

$$GR_i = \frac{H(C) - H(C|A_i)}{H(C|A_i)}. \quad (4)$$

şeklinde ifade edilir[74,75,76]

[70] Alhaj, T. A., M. M. Siraj, A. Zainal, H. T. Elshoush & F. Elhaj (2016) Feature Selection Using Information Gain for Improved Structural-Based Alert Correlation. *PloS one,* 11**,** e0166017.

[71] Yang, Y. & J. O. Pedersen. 1997. A comparative study on feature selection in text categorization. In *Icml*, 412-420.

[72] Uğuz, H. (2011) A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems,* 24**,** 1024-1032.

[73] Oreski, D., S. Oreski & B. Klicek (2017) Effects of dataset characteristics on the performance of feature selection techniques. *Applied Soft Computing,* 52**,** 109-119.

[74] Abraham, R., J. B. Simha & S. S. Iyengar (2009) Effective Discretization and Hybrid feature selection using Naïve Bayesian classifier for Medical datamining. *International Journal of Computational Intelligence Research,* 5**,** 116-129.

Ibrahim S.E. ve arkadaşları tarafından saldırı tespiti sistemleri üzerine yapılan çalışmada kazanç oranı bazlı değişken seçimi ile birlikte karar ağacı, naive bayes ve yapay sinir ağları kullanılmıştır. Sonuçlar karşılaştırıldığında karar ağacı kullanılarak diğer iki yönteme nazaran daha yüksek sınıflandırma doğruluğu elde edilmiştir[77].

2.8.2.3. Simetrik Belirsizlik Bazlı Değişken Seçimi

Simetrik belirsizlik (SU), değişkenlerin sınıfa göre entropi değerlerini kullanarak bir ölçüm yapar. Simetrik belirsizlik hesaplaması için;

$$SU(X,C) = 2\times\left[\frac{H(X)+H(C)-H(X,C)}{H(X)+H(C)}\right] \quad (5)$$

denklemi kullanılarak her değişken için bir değer hesaplanır ve değişkenler bu değere göre sıralanır. Bu yöntem bir değişken sıralama (ranker) yöntemi olarak kullanılmış olur. Bu yöntem kullanılmadan önce yine kesikleştirme işlemi uygulanır[78,79].

Değişkenlerin ikisinin de kesikli olması durumu sağlandığından Denklem (5) ile verilen simetrik belirsizlik (symmetrical uncertainty) hesaplamasından faydalanılır[80].

Bir değişkene ait Denklem (5) ile elde edilen SU değerinin yüksek olması, bu değişken ile hedef değişken arasında önemli bir ilişkinin bulunduğunu gösterir ve bu değişkenin sınıflandırmada kullanılmasının faydalı olduğu anlamına gelir[81].

---

[75] Karegowda, A. G., A. Manjunath & M. Jayaram (2010) Comparative study of attribute selection using gain ratio and correlation based feature selection. *International Journal of Information Technology and Knowledge Management,* 2**,** 271-277.

[76] Priyadarsini, R. P., M. Valarmathi & S. Sivakumari (2011) Gain ratio based feature selection method for privacy preservation. *ICTACT J. Soft Comput,* 1**,** 201-205.

[77] Ibrahim, H. E., S. M. Badr & M. A. Shaheen (2012) Adaptive layered approach using machine learning techniques with gain ratio for intrusion detection systems. *arXiv preprint arXiv:1210.7650*.

[78] Hall, M. A. & G. Holmes (2003) Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data engineering,* 15**,** 1437-1447.

[79] Ali, S. I. & W. Shahzad. 2012. A feature subset selection method based on symmetric uncertainty and ant colony optimization. In *Emerging Technologies (ICET), 2012 International Conference on*, 1-6. IEEE.

[80] Hall, M. 2000. Correlation Based Feature Selection for Discrete and Numeric Class Machine Learning. In *Proc. 17th Int'l. Conf. Machine Learning*.

[81] Piroonratana, T., W. Wongseree, T. Usavanarong, A. Assawamakin, C. Limwongse & N. Chaiyaratana. 2010. Identification of Ancestry Informative Markers from Chromosome-Wide Single Nucleotide Polymorphisms Using Symmetrical Uncertainty Ranking. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2448-2451. IEEE.

Kannan S.S. ve arkadaşları tarafından yapılan değişken alt kümesi seçme çalışmasında dört farklı değişken alt kümesi seçme yönteminin sekiz farklı veri kümesi üzerinde uygulaması yapılmıştır. Sekiz farklı kümeden yedi tanesinde en küçük değişken alt kümesi simetrik belirsizlik kullanılarak elde edilmiştir[82].

Xu B. ve arkadaşları tarafından, Çin el yazısındaki karakterlerin tespiti üzerinde yapılan çalışmada simetrik belirsizlik bazlı değişken seçimi uygulanmış ve daha önce bu konu üzerine Gao ve Liu[83] tarafından önerilmiş olan doğrusal diskriminant analizi ile karşılaştırma yapılmıştır. Simetrik bazlı değişken seçimi ile daha kısa sürede daha yüksek sınıflandırma doğruluğu elde edilmiştir [84].

Jiang ve arkadaşları tarafından genetik algoritma ile simetrik belirsizlik bazlı değişken seçiminin hibrit bir modeli oluşturulmuş ve bu modelin altı farklı veri kümesi üzerinde sınıflandırma için değişken seçimi uygulaması yapılmıştır. Aynı altı veri kümesi üzerinde simetrik belirsizlik ve sarmalayıcı yöntem kullanılarak sınıflandırma için değişken seçimi uygulamaları yapılmıştır. Yapılan bu uygulamalar karşılaştırıldığında hibrit model altı farklı veri kümesinden beş tanesinde en küçük değişken alt kümesini seçmiştir. Sınıflandırma doğruluklarının karşılaştırılmasında ise hibrit model kullanılarak altı kümeden beş tanesinde en yüksek sınıflandırma doğruluğu elde edilmiştir[85].

2.8.2.4. Korelasyon Bazlı Değişken Seçimi

Bir değişken ile hedef değişken arasındaki korelasyon, değişkenin sınıf ile ilgili olduğunu ortaya koymaya yetecek kadar yüksekse ve bu değişken, sınıf ile ilgili diğer değişkenler

[82] Kannan, S. S. & N. Ramaraj (2010) A novel hybrid feature selection via Symmetrical Uncertainty ranking based local memetic search algorithm. *Knowledge-Based Systems,* 23**,** 580-585.

[83] Gao, T.-F. & C.-L. Liu. 2007. LDA-based compound distance for handwritten Chinese character recognition. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, 904-908. IEEE.

[84] Xu, B., K. Huang & C.-L. Liu. 2010. Similar handwritten Chinese characters recognition by critical region selection based on average symmetric uncertainty. In *Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on*, 527-532. IEEE.

[85] Jiang, B.-n., X.-Q. Ding, L.-T. Ma, Y. He, T. Wang & W.-W. Xie. 2008. A hybrid feature selection algorithm: Combination of symmetrical uncertainty and genetic algorithms. In *The Second International Symposium on Optimization and Systems Biology*, 152-157.

tarafından öngörülemez durumda ise, bu değişken sınıflandırma çalışması için iyi bir değişken olarak kabul edilebilir[86].

Korelasyon bazlı değişken seçimini (CFS) ortaya atan Mark Hall, yaklaşımını şu cümle ile özetlemektedir;

"*İyi bir değişken alt kümesinin barındırdığı değişkenler, sınıf ile yüksek korelasyona sahip iken kendi aralarında korelasyon bulunmayanlardır.*[87]*"*

Korelasyon bazlı değişken seçimi sarmalayıcı değişken seçimi ile karşılaştırıldığında oldukça kısa sürede etkili sonuçlar üretebilmektedir[88].

Korelasyon bazlı değişken seçimi, korelasyon bazlı bir değerlendirme fonksiyonuna göre değişkenleri sıralayan basit bir algoritmadır. Bu fonksiyon hedef değişken ile yüksek korelasyona sahip iken kendi aralarında korelasyona sahip olmayan değişkenleri seçmeye meyillidir. İlgili olmayan değişkenler göz ardı edilmelidir çünkü bunlar, hedef değişken ile düşük korelasyona sahip olacaklardır. Korelasyon bazlı değişken seçimine ait alt küme değerlendirme fonksiyonu;

$$\mathrm{M_S} = \frac{k\,\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}} \qquad (6)$$

$\mathrm{M_S}$: Fonksiyona karşı gelen değer (Merit)

S : Değişken alt kümesi (Subset)

k : Değişken alt kümesinin eleman sayısı

$\overline{r_{cf}}$: Değişken ile hedef değişken arasındaki korelasyon

$\overline{r_{ff}}$: Değişkenler arası korelasyonların aritmetik ortalaması

[86] Yu, L. & H. Liu. 2003. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, 856-863.
[87] Hall, M. A. 1999. Correlation-based feature selection for machine learning. The University of Waikato.
[88] Hall, M. A. & L. A. Smith (1997) Feature subset selection: a correlation based filter approach.

şeklindedir. Hesaplanan M_S değeri korelasyon bazlı değişken seçiminin çekirdeğini oluşturur. Değişken alt kümelerinin uzayında bulunan tüm olası alt kümeleri sıralamak için kullanılır[89,90].

Korelasyon bazlı değişken seçiminde, değişkenlerin kendi aralarındaki ve sınıf ile arasındaki korelasyon hesaplanır ve bu korelasyon hesaplarından yararlanılarak hesaplanan M_S değeri ile arama yönteminin değişken alt kümesi uzayına aktarılır. Arama yöntemi, en yüksek M_S değerine sahip olan alt kümeyi çalışma verisi ve test verisi ile birlikte en son değerlendirme aşamasına gönderir[91,92].

Öğrenme verisinin (training data) bir kopyası çalışma öncesinde kesikleştirilir. Örneğin Weka'da bulunan CfsSubsetEval değişken seçim algoritması, değişkenleri değerlendirmeden önce kendi içerisinde bulunan denetimli kesikleştirme yöntemini kullanır[93].

Değişkenlerin tamamının kesikli olması durumu sağlandığından $\overline{r_{cf}}$ ve $\overline{r_{ff}}$ korelasyon değerleri, Denklem (5) ile verilen simetrik belirsizlik (symmetrical uncertainty) hesaplamasından faydalanılarak kestirilir ve bu kestirimlerden faydalanılarak M_S değeri hesaplanır[94].

Sanchez-Marono ve arkadaşları tarafından yapılan çalışmada, korelasyon bazlı değişken seçiminin gürültüye karşı oldukça duyarlı olduğu ifade edilmiştir[95].

2.8.2.5. Tutarlılık Bazlı Değişken Seçimi

Bu değişken seçim yöntemi, değişkenlerin hedef değişken ile tutarlılıklarının göz önünde bulundurulması ile oluşturulur. Bu değişken seçim yaklaşımında, başlangıç verisinin

[89] Hall, M. A. 1999. Correlation-based feature selection for machine learning. The University of Waikato.
[90] Sun, Y., F. Wang, B. Wang, Q. Chen, N. Engerer & Z. Mi (2016) Correlation Feature Selection and Mutual Information Theory Based Quantitative Research on Meteorological Impact Factors of Module Temperature for Solar Photovoltaic Systems. *Energies,* 10**,** 7.
[91] Hall, M. A. 1999. Correlation-based feature selection for machine learning. The University of Waikato.
[92] Jungjit, S. 2016. New Multi-Label Correlation-Based Feature Selection Methods for Multi-Label Classification and Application in Bioinformatics. University of Kent.
[93] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.
[94] Hall, M. 2000. Correlation Based Feature Selection for Discrete and Numeric Class Machine Learning. In *Proc. 17th Int'l. Conf. Machine Learning*.
[95] Sánchez-Maroño, N., A. Alonso-Betanzos & M. Tombilla-Sanromán (2007) Filter methods for feature selection–a comparative study. *Intelligent Data Engineering and Automated Learning-IDEAL 2007***,** 178-187.

içerisindeki alt kümelerden güçlü değişken kombinasyonları aranır. Bu yaklaşım, hedef değişken ile yüksek tutarlılıkta ve küçük kümeler oluşturmaya meyillidir.

S: Bir değişken alt kümesi

j: S için oluşturulabilecek benzersiz değişken alt küme sayısı

N: Verinin içerdiği satır sayısı

i: Satır numarası

D_i: i inci satırdaki değişkenin aldığı değerin meydana gelme sayısı

M_i: i inci satırdaki değişkenin aldığı değerin en geniş sınıftaki kardinalitesi

olmak üzere, değişken alt kümelerine ait tutarlılık değeri,

$$Consistency_s = 1 - \frac{\sum_{i=0}^{j} |D_i| - |M_i|}{N} \tag{7}$$

şeklinde elde edilir.

Önceki yöntemlerde uygulandığı gibi bu yöntemde de çalışma öncesinde sürekli değişkenler için kesikleştirme çalışması yapılır[96,97,98] .

Yu ve Liu(2003)[99] tarafından on farklı veri kümesi üzerinde yapılan çalışmada, korelasyon bazlı değişken seçimi ile seçilen değişkenler ile tutarlılık bazlı değişken seçimi ile seçilen değişkenler C4.5 sınıflandırma uygulamasına tabi tutulmuştur. On farklı veri için elde edilen sınıflandırma doğruluklarının ortalaması incelendiğinde, korelasyon bazlı değişken seçimi ile daha yüksek ortalama sınıflandırma doğruluğu elde edildiği ortaya koyulmuştur.

[96] Hall, M. A. & G. Holmes (2003) Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data engineering,* 15**,** 1437-1447.

[97] Pedrycz, W., G. Succi & A. Sillitti. 2016. *Computational Intelligence and Quantitative Software Engineering*. Springer.

[98] Abraham, R., J. B. Simha & S. S. Iyengar (2009) Effective Discretization and Hybrid feature selection using Naïve Bayesian classifier for Medical datamining. *International Journal of Computational Intelligence Research,* 5**,** 116-129.

[99] Yu, L. & H. Liu. 2003. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, 856-863.

Jiang ve Wang(2016)[100] tarafından yapılan çalışmada, korelasyon bazlı değişken seçimi ve tutarlılık bazlı değişken seçimi algoritmaları yirmi farklı veri kümesi üzerinde uygulanmış ve sonrasında elde edilen değişken alt kümeleri Naive Bayes ve C4.5 ile sınıflandırma uygulamasına tabi tutulmuştur. Korelasyon bazlı değişken seçimi ile elde edilen değişken alt kümelerinin sınıflandırma hata oranları ile tutarlılık bazlı değişken seçimi ile elde edilen değişken alt kümelerinin sınıflandırma hata oranları karşılaştırılmıştır. Sonuçlar incelendiğinde, tutarlılık bazlı değişken seçiminin C4.5 uygulamasında daha düşük hata oranı ortalaması yakalamasına karşın Naive Bayes uygulamasında daha yüksek hata oranı ortalaması yakaladığı görülmektedir.

De Lima Oliviera ve arkadaşları tarafından yapılan, HIV virüsü taşıyan hastaların uygulanan tedaviye verdikleri yanıtın incelenmesinde değişken seçimi uygulamasında, üç farklı veri kümesi üzerinde bir karar ağacı sınıflandırma algoritması ile sınıflandırma uygulaması yapılmıştır. Daha sonra bu üç veri kümesi üzerinde korelasyon bazlı değişken seçimi yapılarak elde edilen değişken alt kümeleri ile bir karar ağacı sınıflandırma algoritması kullanılarak sınıflandırma uygulaması yapılmıştır. Sonuçlar karşılaştırıldığında üç veri kümesi için, korelasyon bazlı değişken seçimi kullanılarak sınıflandırma doğruluğunun arttırıldığı belirtilmiştir[101].

Fernando ve arkadaşları tarafından yapılan ağ saldırılarının sınıflandırılması çalışmasında tutarlılık bazlı değişken seçimi uygulanmış ve değişkenlerin hangi saldırı tiplerinde etkin rol oynadıkları tespit edilmiştir[102].

Jalali ve arkadaşları tarafından, dokuz farklı veri kümesi üzerinde fazi tutarlılık bazlı değişken seçiminden ve tutarlılık bazlı değişken seçiminden faydalanılarak seçilen değişken alt kümeleri ile yapılan yirmi sekiz farklı sınıflandırma çalışmalarının karşılaştırılmasında, on altı

[100] Jiang, S.-y. & L.-x. Wang (2016) Efficient feature selection based on correlation measure between continuous and discrete features. *Information Processing Letters,* 116**,** 203-215.
[101] de Lima Oliveira, S. E., L. H. de Campos Merschmann & L. E. M. Bouillet. 2011. Identifying Significant Features in HIV Sequence to Predict Patients' Response to Therapies. In *BSB*, 18-25. Springer.
[102] Fernando, Z. T., I. S. Thaseen & C. A. Kumar. 2014. Network attacks identification using consistency based feature selection and self organizing maps. In *Networks & Soft Computing (ICNSC), 2014 First International Conference on*, 162-166. IEEE.

tane uygulamada fazi tutarlılık bazlı değişken seçimi daha yüksek sınıflandırma doğruluğu yakalamıştır[103].

2.9. Çalışmada Kullanılan Weka Araçları

Çalışma içerisinde kullanılan Weka araçlarının yerine getirdikleri görevler aşağıda kısaca açıklanmıştır.

WrapperSubsetEval: Sarmalayıcı model uygulayan bir değişken alt kümesi seçme algoritması.

CfsSubSetEval: Korelasyon bazlı değişken alt kümesi seçme prensibi ile çalışan bir değişken alt kümesi seçme algoritması.

ConsistencySubsetEval: Tutarlılık bazlı değişken alt kümesi seçme prensibi ile çalışan bir değişken alt kümesi seçme algoritması.

InfoGainAttributeEval: Bilgi kazanımı bazlı değişken alt kümesi seçme prensibi ile çalışan bir değişken sıralama algoritması.

GainRatioAttributeEval: Kazanç oranı bazlı değişken alt kümesi seçme prensibi ile çalışan bir değişken sıralama algoritması.

SymmetricalUncertAttributeEval: Simetrik belirsizlik bazlı değişken alt kümesi seçme prensibi ile çalışan bir değişken sıralama algoritması.

BestFirst: Bir buluşsal arama yöntemi.

RankSearch: Sıralyacı arama prensibi ile çalışan bir arama yöntemi.

Forward: İleri yönlü arama yapılmasını sağlayan seçenek.

Backward: Geri yönlü arama yapılmasını sağlayan seçenek.

J48: C4.5 karar ağacı algoritmasının kullanıldığı sınıflandırma fonksiyonu[104].

[103] Jalali, L., M. Nasiri & B. Minaei. 2009. A hybrid feature selection method based on fuzzy feature selection and consistency measures. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, 718-722. IEEE.

3. SINIFLANDIRMA SORUNU İÇİN EN UYGUN DEĞİŞKEN ALT KÜMESİ SEÇİMİ ÜZERİNE BİR UYGULAMA

Bu bölümde ulusal bir bankanın veri tabanından elde edilmiş bir veri kümesi üzerinde sınıflandırma çalışması için değişken seçme uygulamaları yapılacak ve değişken seçimi ile oluşturulan değişken alt kümeleri ile sınıflandırma çalışmaları yapılarak sonuçları karşılaştırılacaktır. Bu şekilde değişken alt kümesi seçme çalışmasının sınıflandırma işlemi üzerindeki etkileri gözlemlenmeye çalışılacaktır ve farklı yöntemlerin performansları karşılaştırılacaktır.

3.1. Amaç ve Kapsam

Veri madenciliği sınıflandırma işlemine yönelik olarak farklı değişken alt kümesi seçimi yöntemleri ile elde edilen farklı değişken alt kümeleri üzerinde sınıflandırma çalışması yapılarak değişken seçimi yöntemlerinin sınıflandırma çalışması üzerideki etkileri gözlemlenmeye çalışılırken, değişken seçimi yöntemleri de kendi aralarında karşılaştırılacaktır.

Üzerinde çalışılan veri kümesinin bir bankaya ait gerçekleşen ve güncel olmayan bir veri tabanından rastlantısal seçilmiş bir veri kümesi olması nedeni ile sektördeki operasyonel bir ihtiyaca cevap verecek bir çözüm ortaya koymaktan ziyade değişken alt kümesi seçme algoritmalarının bir uygulaması şeklinde bir çalışma yapılmıştır.

Hedef değişken olarak belirlenen ürün için, veri kümesinde bu ürünü kullananların oranı yaklaşık %13 düzeyinde olması karar ağacı uygulamasının sonuçlarını, daha ziyade ürünü kullanmayanları açıklamaya yönlendireceğinden ürünü kullanmayan kişilerin içerisinden, ürünü kullanan kişi sayısı kadar rastlantısal kişi seçilmiş ve uygulamalar bu indirgenmiş veri kümesi ile gerçekleştirilmiştir.

Ulusal bir bankanın müşteri verileri içerisinden rastgele seçilmiş bir örnek veri kümesi üzerinde yapılacak olan çalışmada, mevcut veri kümesi üzerinde uygulanabilir olan J48 karar ağacı yöntemi ile sınıflandırma çalışması üzerinde değişken alt kümesi seçim yöntemlerinden faydalanarak iyileştirmeler hedeflenmektedir.

Çalışmadan beklenen iyileştirme sadece var olan bir sınıflandırma çalışmasının iyileştirilmesi

[104] Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.

olduğu için sınıflandırma çalışmasının operasyonel taraf açısından faydası çalışma içerisinde tartışılmamıştır.

3.2. Veri Kümesi

Ulusal bir bankaya ait rastlantısal olarak seçilmiş 13.274 müşterinin 103 değişkenden oluşan bir veri kümesi üzerinde çalışılacaktır. Çalışma içerisinde Microsoft Excel 2013 ve Weka 3.9 Developer Edition kullanılacaktır. Weka yazılımının kurulumunda içerisinde doğrudan bulunmayan değişken alt kümesi seçme algoritmaları, Weka yazılımının kendisine ait "package manager" aracı kullanılarak temin edilmiştir.

Çalışmaya başlanan veri kümesindeki her satır, bir müşteriye ait verileri ifade etmektedir. Sütunlar ise müşterilerin banka tarafından gerçekleştirilen operasyonel hareketlerine ait kayıtları ve müşterilere ait kişisel bilgileri içermektedir. Çalışmaya dahil edilen 103 adet değişken, aşağıdaki Tablo 3'de özetlenmiştir.

Tablo 3. Veri Kümesinde Bulunan Değişkenler

Grup	Değişkenler	Değişken Grubu	Adet
D	D01,D02,D03,D04	Demografik Özellikleri	4
M	M02,M03,M04	Banka Kayıt Bilgileri	3
H	H01,H02,H03,H04,H05,H06,H23	Hesap - Sahip mi	7
	H07,H08,H09,H10,H11	Hesap - Aktif mi	5
	H12,H13,H14,H15,H16,H17	Hesap Sayısı	6
	H18,H19,H20,H21,H22	Hesap Bakiyesi	5
L	L01,L02,L03	Kredi - Sahip mi	3
	L04,L05,L06	Kredi Aktif mi	3
	L07,L08,L09	Kredi Sayısı	3
	L10,L11,L12	Kredi Bakiyesi	3
C	C01,C02,C03,C04,C06	Kredi Kartı - Sahip mi	5
	C07,C08,C09,C10,C11,C12	Kredi Kartı Aktif mi	6
	C13,C15,C16,C18	Kredi Kartı Sayısı	4
	C19,C21,C22,C23,C24	Kredi Kartı Bakiyesi	5
Y	Y01,Y02,Y03,Y04,Y05	Yatırım Ürünü - Sahip mi	5
	Y06,Y07,Y08,Y09,Y10	Yatırım Ürünü - Aktif mi	5
	Y11,Y12,Y13,Y14,Y15	Yatırım Ürünü Sayısı	5
	Y16,Y17,Y18,Y19,Y20	Yatırım Ürünü Bakiyesi	5
U	U01,U02,U03,U04,U05,U12,U21,U22,U23	Hizmet Ürünü - Sahip mi	9
	U06,U07,U08,U09,U10	Hizmet Ürünü Aktif mi	5
	U13,U14,U15,U16,U17	Hizmet Ürünü Sayı	5
	U19,U20	Hizmet Ürünü Tutarı	2
Toplam			103

Değişkenler özetlenirken benzer değişkenler birlikte gruplanmaya çalışılmıştır. Değişken grupları;

- D - Demografik değişkenler
- M - Banka Kayıtları
- H - Müşteri Hesapları
- L - Kredi Kullanımı
- C - Kredi Kartları
- Y - Yatırım Ürünleri
- U - Diğer Bankacılık Ürünleri

şeklinde ifade edilmiştir.

3.3. Veri Hazırlama Süreci

C02 hedef değişken olarak seçilmiş ve bu hedef değişken için veri kümesinin üzerinde çalışmaya uygun hale getirilmesi işleminde;

C08 (C02 ürününü kullanma aktivitesi),

C14 (C02 ürününe sahip olma sayısı) ve

C20 (C02 ürünündeki bakiye)

değişkenleri C02 ile yüksek korelasyona sahip olduklarından dolayı veri kümesinden çıkarılmıştır.

Veri kümesi bu çalışmanın öncesinde de bir temizlik sürecinden geçtiği için bazı veri sorunlarından arındırılmış olarak çalışmaya aktarılmıştır. Ancak yine de bu çalışmanın iyileştirmeye yönelik bazı veri sorunlarının giderilmesi ihtiyacı saptanmıştır. Bazı eksik ve tutarsız verilerin elenmesi uygun bulunmuştur.

Müşterilere ait eğitim durumu ve bankanın daha önceden müşterileri için tanımlamış olduğu segment bilgileri (D04 ve M04 değişkenlerinde) eksik olan müşteriler elenerek çalışma

dışında bırakılmıştır. Bu eleme sürecinin ardından 18 yaş altında müşterilerden hiçbiri çalışma kümesine dahil olamamıştır.

Kayıtların seçilmesi açısından ise C02 ürününe sahip olan 1.233 adet örneğin tümü ile sahip olmayanlardan rastlantısal olarak seçilen 1.233 adet örnek dikkate alınmıştır. Toplamda 2.466 adet örnek ile çalışma gerçekleştirilmiştir.

3.4. Verinin Tanıtılması

Yaş gruplarını ifade eden D01 değişkenine ait tüm veri ve seçilen örneğe ait sıklık dağılımları Tablo 4 ile gösterilmiştir.

Tablo 4. D01 Değişkenine Ait Sıklık Dağılımı

D01 Kod	D01 Açıklama	Sıklık	Göreli
2	18-24	49	%1,99
3	25-34	600	%24,33
4	35-44	826	%33,50
5	45-54	573	%23,24
6	55+	418	%16,95
Toplam		2466	%100,00

Cinsiyet verisini ifade eden D02 değişkenine ait sıklık dağılımı Tablo 5' te gösterilmiştir.

Tablo 5. D02 Değişkenine Ait Sıklık Dağılımı

D02 Kod	D02 Açıklama	Sıklık	Göreli
1	Erkek	1800	%72,99
2	Kadın	666	%27,01
Toplam		2466	%100,00

D03 değişkeni ile ifade edilen evlilik durumu verisinin sıklık dağılımları Tablo 6 ile gösterilmiştir.

Tablo 6. D03 Değişkenine Ait Sıklık Dağılımı

D03 Kod	D03 Açıklama	Sıklık	Göreli
1	Bekar	606	%24,57
2	Evli	1775	%71,98

3	Boşanmış	85	%3,45
Toplam		2466	%100,00

D04 olarak ifade edilen eğitim durumuna ait sıklık dağılımları Tablo 7 ile gösterilmiştir.

Tablo 7. D04 Değişkenine Ait Sıklık Dağılımı

D04 Kod	D04 Açıklama	Sıklık	Göreli
1	İlk	262	%10,62
2	Orta	285	%11,56
3	Lise	649	%26,32
4	Ön lisans	72	%2,92
5	Lisans	888	%36,01
6	Yüksek Lisans	296	%12,00
7	Doktora	14	%0,57
Toplam		2466	%100,00

Sınıflandırma çalışmasında hedef değişken olarak kullanılacak olan C02 için sıklık dağılımları Tablo 8 ile gösterilmiştir.

Tablo 8. C02 Değişkenine Ait Sıklık Dağılımı

C02 Kod	C02 Açıklama	Sıklık	Göreli
0	C02 Kullanıyor	1233	%50,00
1	C02 Kullanmıyor	1233	%50,00
Toplam		2466	%100,00

3.5. Sınıflandırma Uygulamasına Yönelik Değişken Seçimi Uygulamaları

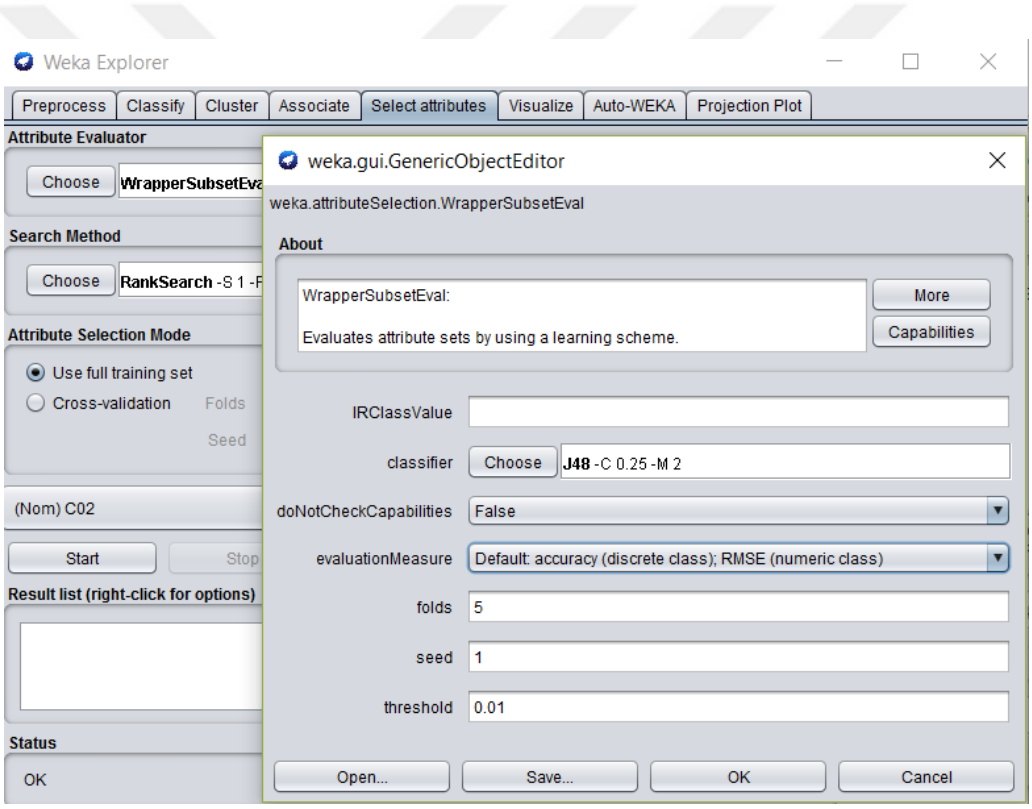
C02 değişkeninin sınıflandırma uygulamasına yönelik değişken seçimi için oluşturulan 100 değişken ve 2466 adet örnekten oluşan veri kümesi kullanılacaktır. Çalışma içerisinde 7 farklı değişken seçim yöntemi ile C02 hedef değişkeni için sınıflandırmaya uygun değişken alt kümelerinin belirlenmesi sağlanacaktır. Ardından elde edilen 7 farklı değişken alt kümeleri ve tüm değişkenleri içeren veri kümesi için ayrı ayrı J48 algoritması ile sınıflandırma çalışması yapılarak 8 farklı karar ağacı oluşturulacak ve sonuçlar karşılaştırılacaktır.

3.5.1 Değişken Seçimi Uygulamaları

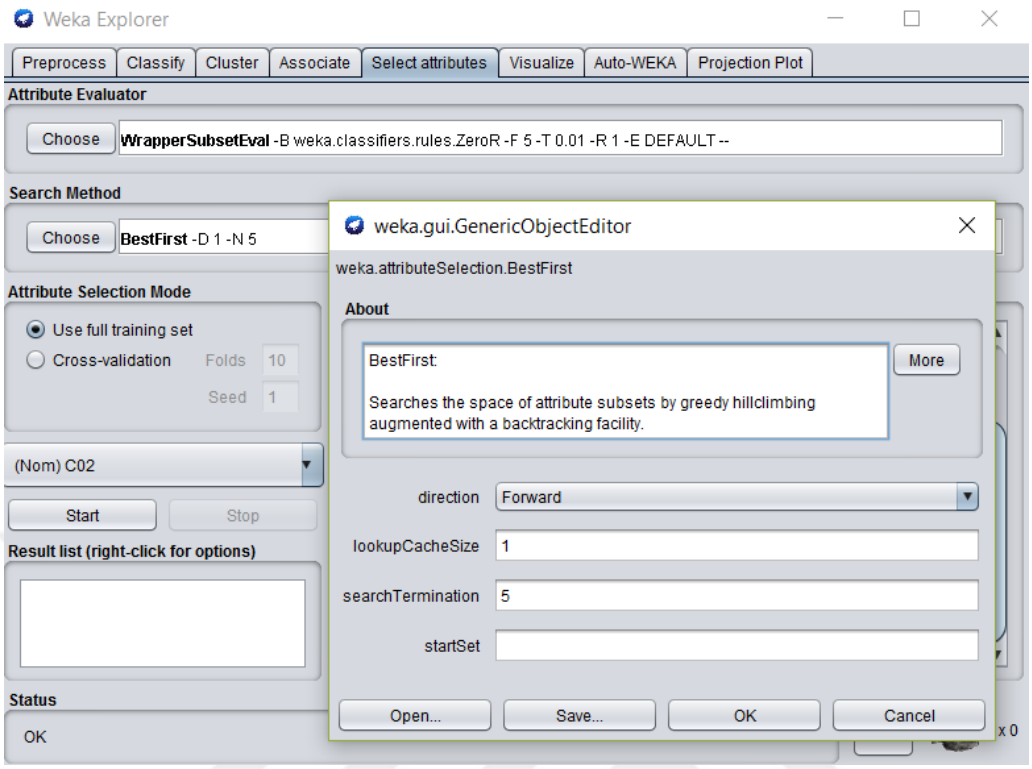
Bu bölümde çalışma içerisinde uygulanan değişken seçimi yöntemlerinin detayları ifade edilecektir.

3.5.1.1 Yöntem-1: WrapperSubsetEval – Forward - BestFirst

Bu yöntemde, sarmalayıcı bir değişken değerlendirme algoritması olan WrapperSubsetEval kullanılacaktır. WrapperSubsetEval algoritmasının çapraz doğrulama yapmak için kullanacağı sınıflandırma algoritması olarak J48 algoritması seçilmiştir (Şekil 3). Arama metodu olarak ileri yönlü BestFirst seçilmiştir (Şekil 4).



Şekil 3. Weka Ara Yüz Görüntüsü - 1



Şekil 4. Weka Ara Yüz Görüntüsü - 2

Yöntem-1, çalışması sırasında 1654 adet değişken alt kümesini değerlendirmiş ve model doğruluğunun bir yerel maksimum noktasını bulduğunda aramasını kesmiştir. Değerlendirme süresi, Weka tarafından sonuç ekranına yansıtılmadığından dolayı yaklaşık sekiz dakika olarak kayıt edilmiştir. Weka, bu arama sonucunda 11 değişkenli bir alt kümeyi seçmiştir. Seçilen değişkenler ise;

D02, M03, H02, H05, H12, H15, H19, C06, C12, Y16, U03

şeklindedir. Yöntem-1'e ait Weka çıktısı Ek-1'de bulunmaktadır.

3.5.1.2 Yöntem-2 : WrapperSubsetEval – Backward - BestFirst

Bu yöntemde, Yöntem-1'den farklı olarak sadece arama yönü, evrensel kümeden boş kümeye doğru çalışmasını sürdüren geriye doğru arama olarak seçilmiş olup diğer tüm seçimler Yöntem-1 ile aynı şekilde uygulanmıştır.

Yöntem-2, 6578 adet alt kümeyi değerlendirerek 56 adet değişkene sahip bir kümeyi yaklaşık olarak dört saat 30 dakikalık bir sürede seçmiştir. Seçilen değişkenler;

D02, D04, D04, M03, M04, H01, H02, H05, H07, H08, H09, H10, H11, H12, H15, H18, H19, H23, L07, L09, C03, C04, C07, C10, C12, C13, C15, C16, C18, C19, C22, Y02, Y03, Y04, Y06, Y07, Y08, Y09, Y11, Y12, Y13, Y16, Y17, U01, U02, U03, U04, U07, U10, U12, U14, U15, U16, U21, U22, U23, U20

şeklindedir. Yöntem-2'e ait Weka çıktısı Ek-2'de bulunmaktadır.

3.5.1.3 Yöntem-3: CfsSubSetEval - BestFirst

Bir diğer BestFirst arama yöntemi ile uygulanan CfsSubSetEval algoritması önceki algoritmalardan farklı olarak sarmalayıcı olarak değil de filtreleyici bir çalışma prensibine sahiptir. İleri yönlü çalışması ile geri yönlü çalışması arasında kayda değer bir fark bulunmadığı için sadece ileri yönlü uygulaması çalışmaya dahil edilmiştir.

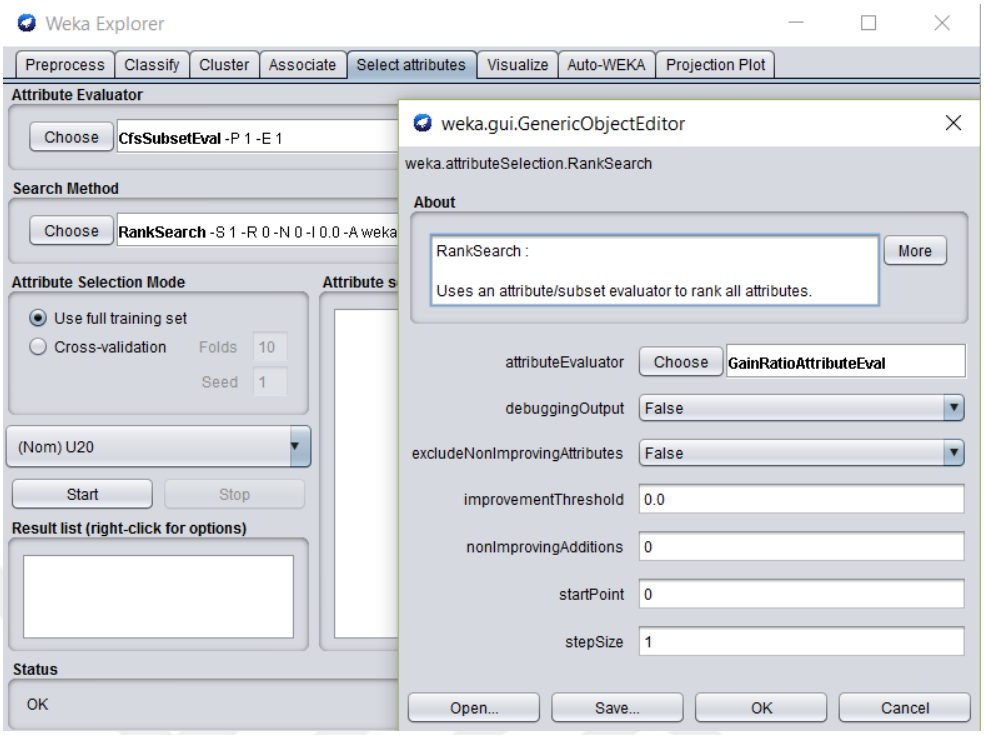
Yöntem-3 Bir dakikadan kısa süre içerisinde 1213 değişken alt kümesini değerlendirerek 11 değişkenli bir alt kümeyi seçmiştir. Seçilen değişkenler;

D04, D04, M03, M04, H02, H14, L12, C06, C12, C18, U03

şeklindedir. Yöntem-3'e ait Weka çıktısı Ek-3'de bulunmaktadır.

3.5.1.4 Yöntem-4: CfsSubSetEval - GainRatioAttributeEval

Önceki uygulamalardan farklı olarak Yöntem-4, Yöntem-5, Yöntem-6 uygulamalarında BestFirst arama yönteminin yerine RankSearch arama yöntemi kullanılmıştır. Bu yöntemin kullanılma amacı değişken alt kümesi seçim algoritmalarının haricinde değişken sıralama algoritmalarını da çalışmada kullanmaktır (Şekil 5).



Şekil 5. Weka Ara Yüz Görüntüsü - 3

Değişken sıralama algoritmaları belirli bir değişken alt kümesi oluşturmayıp sadece değişkenlerin karşısına bir skor atamaktadır. Ancak değişkenlere skor ataması yapıldıktan sonra analize devam edilebilmesi için bir şekilde atanan skordan istifade edilerek değişken alt kümesi seçme mecburiyeti bulunmaktadır. Çalışmada, Weka içerisinde bu işlemi yapabilen RankSearch arama yöntemi kullanılmıştır.

Arama metodu olarak kullanılan RankSearch algoritmasının içerisinde bir adet değişken sıralama algoritması çalışırken bu sıralama algoritmasından bağımsız olarak çalışan bir de değişken alt kümesi seçme algoritması kullanılmaktadır. Sıralama algoritmaları ile çalışırken alt küme seçme algoritması olarak kısa sürede etkili sonuçlar üretebilen CfsSubSetEval kullanılacaktır.

Yöntem 4, bir dakikadan kısa süre içerisinde tüm değişkenleri sıralamış ve bu sıralamayı kullanarak dokuz değişkenden oluşan bir alt küme seçmiştir. Seçilen değişkenler;

M03, M03, M04, H06, H14, L12, C06, C18, U03

şeklindedir. Yöntem-4'e ait Weka çıktısı Ek-4'de bulunmaktadır.

3.5.1.5 Yöntem-5: CfsSubSetEval - InfoGainAttributeEval

Yöntem-4'ten farklı olarak sadece RankSearch içerisindeki değişken sıralama algoritması InfoGainAttributeEval olarak seçilmiş diğer seçenekler Yöntem-4 ile aynı şekilde seçilmiştir.

Yöntem-5, bir dakikadan kısa süre içerisinde tüm değişkenleri sıralamış ve bu sıralamayı kullanarak 12 değişkenden oluşan bir alt küme seçmiştir. Seçilen değişkenler;

D04,D04,M03,M04,H02,H12,H14,L12,C06,C12,C18,U15

şeklindedir. Yöntem-5'e ait Weka çıktısı Ek-5'de bulunmaktadır.

3.5.1.6 Yöntem-6: CfsSubsetEval - SymmetricalUncertAttributeEval

Yöntem-4'ten farklı olarak sadece RankSearch içerisindeki değişken sıralama algoritması SymmetricalUncertAttributeEval olarak seçilmiş diğer seçenekler Yöntem-4 ile aynı şekilde seçilmiştir.

Yöntem 6, bir dakikadan kısa süre içerisinde tüm değişkenleri sıralamış ve bu sıralamayı kullanarak dokuz değişkenden oluşan bir alt küme seçmiştir. Seçilen değişkenler;

M03, M03, M04, H06, H14, L12, C06, C22, U03

şeklindedir. Yöntem-6'e ait Weka çıktısı Ek-6'de bulunmaktadır.

3.5.1.7 Yöntem-7: ConsistencySubsetEval- BestFirst

Yöntem-3'ten farklı olarak CfsSubSetEval yerine ConsistencySubsetEval algoritması kullanılmış, diğer tüm seçenekler Yöntem-3 ile aynı şekilde yapılmıştır.

Yöntem 7, bir dakikadan kısa süre içerisinde tüm değişkenleri sıralamış ve bu sıralamayı kullanarak 21 değişkenden oluşan bir alt küme seçmiştir. Seçilen değişkenler;

D01, D01, D02, D03, D04, M02, M03, M04, H04, H05, H08, H12, H13, H18, C06, C13, C18, C22, Y02, U03, U13

şeklindedir. Yöntem-7'ye ait Weka çıktısı Ek-7'de bulunmaktadır.

Yukarıda uygulanan yedi yöntemin kullandıkları değişken seçim modelleri aşağıdaki Tablo 9'da gösterilmiştir.

Tablo 9. Kullanılan Yöntemlerin İçerdiği Modeller

	Değişken Alt Kümesi Seçim Modeli
Yöntem-1	Sarmalayıcı
Yöntem-2	Sarmalayıcı
Yöntem-3	Filtreleyici
Yöntem-4	Filtreleyici
Yöntem-5	Filtreleyici
Yöntem-6	Filtreleyici
Yöntem-7	Filtreleyici

3.6. Karar Ağacı Uygulamaları

Bu bölümde çalışmanın önceki kısmında kullanılan yöntemler tarafından seçilen farklı değişken alt kümeleri ile karar ağaçları oluşturarak sonuçları karşılaştırılacaktır.

Değişken seçimi yöntemleri ile seçilen değişken alt kümeleri ve değişken seçiminden önce mevcut bulunan veri kümesi üzerinde J48 karar ağacı uygulamaları Weka yardımı ile gerçekleştirilecektir.

3.6.1. Değişken Seçimi Yapılmadan Oluşturulan Karar Ağacı

Veri temizliği sürecinin ardından oluşan 100 değişkenli veri kümesi ile hiç değişken seçimi yapılmaksızın karar ağacı uygulaması yapılacaktır. Bu uygulama, kullanılan yöntemler tarafından seçilen değişken alt kümeleri ile oluşturulacak karar ağaçlarının başarılarını ölçmede yardımcı olacaktır.

100 değişkenli küme ile J48 kullanarak oluşturulan karar ağacına ait sonuçlar aşağıdaki Tablo 10 ile gösterilmiştir.

Tablo 10. Tüm Değişkenler İle Oluşturulan Karar Ağacı

Seçim Öncesi Veri ile oluşturulan Karar Ağacı	
Değişken Sayısı	100

Ağaçtaki dal sayısı	184
Ağaç büyüklüğü	252
Model oluşturma Süresi(Saniye)	0,52 Saniye
Doğru sınıflandırma yüzdesi	%79,6026
Kappa istatistiği	0,5921
Ortalama mutlak hata (MAE)	0,2596
Hataların ortalama karekökü (Root mean squared error)	0,406
Mutlak bağıl hata (Relative absolute error)	%51,9139
Bağıl hataların karekökü (Root relative squared error)	%81,2069
Doğru Sınıflandırılan Örnek	1963
Hata matrisi	a b 945 288 | a = 0 215 1018 | b = 1

Ek-8'de bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-16'da bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.2. Yöntem-1 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-1 ile seçilen 11 değişkene ek olarak hedef değişken C02 ile birlikte 12 değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 11 ile gösterilmiştir.

Tablo 11. Yöntem-1' e Ait Sınıflandırma Çıktısı

Yöntem-1	
Değişken Sayısı	12
Ağaçtaki dal sayısı	15
Ağaç büyüklüğü	29
Model oluşturma Süresi(Saniye)	0,08
Doğru sınıflandırma yüzdesi	%82,11
Kappa istatistiği	0,6423
Ortalama mutlak hata (MAE)	0,283
Hataların ortalama karekökü (Root mean squared error)	0,3797
Mutlak bağıl hata (Relative absolute error)	%56,6042
Bağıl hataların karekökü (Root relative squared error)	%75,9316
Doğru Sınıflandırılan Örnek	2025
Hata matrisi	a b 934 299 | a = 0 142 1091 | b = 1

Ek-9'da bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-17'de bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.3. Yöntem-2 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-2 ile seçilen 56 değişkene ek olarak hedef değişken olan C02 ile birlikte 57 değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 12 ile gösterilmiştir.

Tablo 12. Yöntem-2' ye Ait Sınıflandırma Çıktısı

Yöntem-2	
Değişken Sayısı	57
Ağaçtaki dal sayısı	131
Ağaç büyüklüğü	195
Model oluşturma Süresi(Saniye)	0,22
Doğru sınıflandırma yüzdesi	%80,6569
Kappa istatistiği	0,6131
Ortalama mutlak hata (MAE)	0,2605
Hataların ortalama karekökü (Root mean squared error)	0,3908
Mutlak bağıl hata (Relative absolute error)	%52,1023
Bağıl hataların karekökü (Root relative squared error)	%78,165
Doğru Sınıflandırılan Örnek	1989
Hata matrisi	a b 965 268 | a = 0 209 1024 | b = 1

Ek-10'da bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-18'de bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.4. Yöntem-3 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-3 ile seçilen 56 değişkene ek olarak hedef değişken olan C02 ile birlikte 57 değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 13 ile gösterilmiştir.

Tablo 13. Yöntem-3' e Ait Sınıflandırma Çıktısı

Yöntem-3	
Değişken Sayısı	11
Ağaçtaki dal sayısı	48
Ağaç büyüklüğü	60
Model oluşturma Süresi(Saniye)	0,03
Doğru sınıflandırma yüzdesi	%81,3058

Kappa istatistiği	0,6261
Ortalama mutlak hata (MAE)	0,2743
Hataların ortalama karekökü (Root mean squared error)	0,3795
Mutlak bağıl hata (Relative absolute error)	%54,8627
Bağıl hataların karekökü (Root relative squared error)	%75,8901
Doğru Sınıflandırılan Örnek	2006
Hata matrisi	a b 912 321 | a = 0 140 1093 | b = 1

Ek-11'de bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-19'da bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.5. Yöntem-4 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-4 ile seçilen 9 değişkene ek olarak hedef değişken olan C02 ile birlikte dokuz değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 14 ile gösterilmiştir.

Tablo 14. Yöntem-4' e Ait Sınıflandırma Çıktısı

Yöntem-4	
Değişken Sayısı	9
Ağaçtaki dal sayısı	13
Ağaç büyüklüğü	18
Model oluşturma Süresi(Saniye)	0,03
Doğru sınıflandırma yüzdesi	%80,9813
Kappa istatistiği	0,6196
Ortalama mutlak hata (MAE)	0,2913
Hataların ortalama karekökü (Root mean squared error)	0,3855
Mutlak bağıl hata (Relative absolute error)	%58,2607
Bağıl hataların karekökü (Root relative squared error)	%77,0952
Doğru Sınıflandırılan Örnek	1997
Hata matrisi	a b 898 335 | a = 0 134 1099 | b = 1

Ek-12'de bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-20'de bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.6. Yöntem-5 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-5 ile seçilen 11 değişkene ek olarak hedef değişken olan C02 ile birlikte 12 değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 15 ile gösterilmiştir.

Tablo 15. Yöntem-5' e Ait Sınıflandırma Çıktısı

Yöntem-5	
Değişken Sayısı	12
Ağaçtaki dal sayısı	50
Ağaç büyüklüğü	64
Model oluşturma Süresi(Saniye)	0,05
Doğru sınıflandırma yüzdesi	%81,2247
Kappa istatistiği	0,6245
Ortalama mutlak hata (MAE)	0,2807
Hataların ortalama karekökü (Root mean squared error)	0,3849
Mutlak bağıl hata (Relative absolute error)	%56,1418
Bağıl hataların karekökü (Root relative squared error)	%76,9853
Doğru Sınıflandırılan Örnek	2003
Hata matrisi	a b 928 305 | a = 0 158 1075 | b = 1

Ek-13'de bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-21'de bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.7. Yöntem-6 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-6 ile seçilen 8 değişkene ek olarak hedef değişken olan C02 ile birlikte dokuz değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 16 ile gösterilmiştir.

Tablo 16. Yöntem-6'ya Ait Sınıflandırma Çıktısı

Yöntem-6	
Değişken Sayısı	9
Ağaçtaki dal sayısı	27
Ağaç büyüklüğü	39
Model oluşturma Süresi(Saniye)	0,03
Doğru sınıflandırma yüzdesi	%81,4274
Kappa istatistiği	0,6285
Ortalama mutlak hata (MAE)	0,2785
Hataların ortalama karekökü (Root mean squared error)	0,379

Mutlak bağıl hata (Relative absolute error)	%55,7008
Bağıl hataların karekökü (Root relative squared error)	%75,8096
Doğru Sınıflandırılan Örnek	2008
Hata matrisi	a b 931 302 | a = 0 156 1077 | b = 1

Ek-14'te bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-22'de bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.6.8. Yöntem-7 ile Seçilen Değişkenlerin Oluşturduğu Karar Ağacı

Yöntem-7 ile seçilen 20 değişkene ek olarak hedef değişken olan C02 ile birlikte 21 değişkenli veri kümesi kullanılmıştır. Weka tarafından bu küme üzerinden J48 algoritması ile üretilen karar ağacına ait detaylar aşağıdaki Tablo 17 ile gösterilmiştir.

Tablo 17. Yöntem-7'ye Ait Sınıflandırma Çıktısı

Yöntem-7	
Değişken Sayısı	21
Ağaçtaki dal sayısı	142
Ağaç büyüklüğü	188
Model oluşturma Süresi(Saniye)	0,09
Doğru sınıflandırma yüzdesi	%79,9676
Kappa istatistiği	0,5994
Ortalama mutlak hata (MAE)	0,2783
Hataların ortalama karekökü (Root mean squared error)	0,4019
Mutlak bağıl hata (Relative absolute error)	%55,65
Bağıl hataların karekökü (Root relative squared error)	%80,3899
Doğru Sınıflandırılan Örnek	1972
Hata matrisi	a b 922 311 | a = 0 183 1050 | b = 1

Ek-15'de bu uygulamaya ait sınıflandırma Weka çıktısı bulunmaktadır. Ek-23'de bu uygulamaya ait karar ağacı Weka çıktısı bulunmaktadır.

3.7. Değerlendirme

Bu bölümde 3.6 bölümünde gerçekleştirilen karar ağacı çalışmaları birlikte değerlendirilecektir. Aşağıda bulunan Tablo 18 üzerinde önceki bölümde gerçekleştirilen tüm karar ağacı uygulamalarının sonuçları yan yana getirilerek karşılaştırma imkanı oluşturulmuştur.

Tablo 18 incelenerek en iyi ve en kötü sonuçları üreten algoritmalar Tablo 19 ile gösterilmiştir. Tablo 18 üzerindeki sonuçlara göre yöntemlerin sıralamaları ise Tablo 20 ile gösterilmiştir.

Sonuçlar incelendiğinde, değişken seçim algoritmaları yardımı ile tüm değişkenleri içeren veri kümesi de bulunan 100 değişkenin dokuz ile 57 arasında değişkene indirgenebildiği görülmektedir. Yöntem-2 ile seçilen 57 değişken dışındaki yöntemler ise dokuz ile 21 arasında değişken seçmişlerdir. Yöntem-2 algoritmasının değişkenlerin evrensel kümesinden değişkenleri azaltarak boş kümeye doğru bir arama süreci izlemesi, sınıflandırma doğruluğunu en yüksek yapan bir yerel maksimum değerinde aramasını kesmesi nedeni ile 100 değişkenden başlayarak 57 değişkene kadar indirgeme yaptığı ve tahmin tutarlılığını yüzde 80,6569 düzeyine getirdiğinde aramayı kestiği görülmektedir.

Oluşan karar ağaçlarının yaprak sayıları incelendiğinde, tüm değişkenlerin analize dahil edildiği durumda 184 yaprak oluşurken, oluşturulan değişken alt kümelerinin katıldığı analizler sonucunda oluşan ağaçların yaprak sayılarının 13-131 bandında olduğu görülmektedir. Değişken seçim algoritmalarının amaçlarından bir diğeri olan analizi daha anlaşılır hale getirme konusunda ise yine tüm algoritmaların başarılı olduğu söylenebilir.

Tüm değişkenleri içeren veri kümesinin sınıflandırma işlemine girdiği durumda 0,52 saniyede sonuç alınabilirken değişken sayısının azaltıldığı diğer tüm durumlarda bu süre 0,03-0,22 bandında yer almaktadır. Değişken seçimi algoritmalarının amaçlarından biri olan analiz süresini düşürme konusunda tüm yöntemlerin ürettikleri değişken alt kümeleri ile yapılan analizlerde, sürenin azaltılabildiği gözlemlenmektedir.

Tablo 18. Karar Ağaçlarının Karşılaştırılması

	Değişken Sayısı	Ağaçtaki dal sayısı	Ağaç büyüklüğü	Model oluşturma Süresi(Saniye)	Doğru sınıflandırma yüzdesi	Kappa	Ortalama mutlak hata (MAE)	Hataların ortalama karekökü (Root mean squared error)	Mutlak bağıl hata (Relative absolute error)	Bağıl hataların karekökü (Root relative squared error)	Doğru Sınıflandırılan Örnek	Hata matrisi
Seçim Öncesi Veri	100	184	252	0,52	%79,6026	0,5921	0,2596	0,406	%51,9139	%81,2069	1963	a b 945 288 | a = 0 215 1018 | b = 1
Yöntem-1	12	15	29	0,08	%82,11	0,6423	0,283	0,3797	%56,6042	%75,9316	2025	a b 934 299 | a = 0 142 1091 | b = 1
Yöntem-2	57	131	195	0,22	%80,6569	0,6131	0,2605	0,3908	%52,1023	%78,165	1989	a b 965 268 | a = 0 209 1024 | b = 1
Yöntem-3	11	48	60	0,03	%81,3058	0,6261	0,2743	0,3795	%54,8627	%75,8901	2006	a b 912 321 | a = 0 140 1093 | b = 1
Yöntem-4	9	13	18	0,03	%80,9813	0,6196	0,2913	0,3855	%58,2607	%77,0952	1997	a b 898 335 | a = 0 134 1099 | b = 1
Yöntem-5	12	50	64	0,05	%81,2247	0,6245	0,2807	0,3849	%56,1418	%76,9853	2003	a b 928 305 | a = 0 158 1075 | b = 1
Yöntem-6	9	27	39	0,03	%81,4274	0,6285	0,2785	0,379	%55,7008	%75,8096	2008	a b 931 302 | a = 0 156 1077 | b = 1
Yöntem-7	21	142	188	0,09	%79,9676	0,5994	0,2783	0,4019	%55,65	%80,3899	1972	a b 922 311 | a = 0 183 1050 | b = 1

Tablo 19. Yöntemlerin Karşılaştırılması

	Seçim öncesi	En iyi yöntem		En kötü yöntem	
Değişken Sayısı	100	9	Yöntem-4, Yöntem-6	57	Yöntem-2
Karar ağacındaki dal sayısı	184	13	Yöntem-4	131	Yöntem-2
Model oluşturma Süresi (Saniye)	0.52	0.03	Yöntem-3, Yöntem-4, Yöntem-6	0.22	Yöntem-2
Doğru sınıflandırma yüzdesi (%)	%79,6026	%82,11	Yöntem-1	%79,9621	Yöntem-7
Kappa	0,5921	0,64	Yöntem-1	0,5994	Yöntem-7
Hata matrisi a=0 isabetli sayısı	945	965	Yöntem-2	898	Yöntem-4
Hata matrisi b=1 isabetli sayısı	1018	1099	Yöntem-4	1024	Yöntem-2

Tablo 20. Yöntemlerin Sıralanması

	Değişken Sayısı	Ağaçtaki dal sayısı	Ağaç büyüklüğü	Model oluşturma Süresi(Saniye)	Doğru sınıflandırma yüzdesi	Kappa	Doğru Sınıflandırılan a=0	Doğru Sınıflandırılan b=1	Ortalama mutlak hata (MAE)	Ortalama	Medyan
Yöntem-4	1	1	1	1	5	5	7	1	7	3,22	1
Yöntem-1	4	2	2	5	1	1	2	3	6	2,89	2
Yöntem-6	1	3	3	1	2	2	3	4	4	2,56	3
Yöntem-3	3	4	4	1	3	3	6	2	2	3,11	3
Yöntem-5	4	5	5	4	4	4	4	5	5	4,44	4
Yöntem-2	6	7	7	7	6	6	1	7	1	5,33	6
Yöntem-7	5	6	6	6	7	7	5	6	3	5,67	6

Sınıflandırma başarısının ise tüm değişkenler ile oluşturulan modelde yüzde 79,6026 düzeyinde iken değişken seçme algoritmalarının oluşturduğu alt kümeler ile oluşturulan modellerde yüzde 79,9611-82,11 bandında bulunduğu gözlemlenmektedir. Değişken seçim algoritmaları kullanılırken, ortaya çıkacak modelin başarısını iyileştirmek veya

düşük düzeyde kayıp ile daha sade bir model ortaya koymak olduğu için belirlenen tüm değişken alt kümelerinin bu düzeyi yakaladığı söylenebilir.

Kappa istatistiği incelendiğinde oluşturulan modeller arasında çok belirgin farklar oluştuğu söylenememektedir. Tüm algoritmaların seçtiği değişken alt kümelerinin ortaya koydukları modelleri üzerinden hemen hemen aynı düzeylerde kappa istatistiği hesaplandığı görülmektedir.

Hata matrislerinde hedef değişkenin değerinin bir olma durumunun model tarafından doğru öngörülen adetler incelendiğinde, tüm değişkenler ile oluşturulan model için 945 iken değişken alt kümeleri ile oluşturulan modellerde 965-898 bandında doğru öngörü sayıları gözlemlenmektedir. Hedef değişkenin sıfır olduğu durum için tüm değişkenleri içeren veri kümesinde doğru öngörülen sayı 1018 iken oluşturulan değişken alt kümeleri ile oluşturulan modellerde 1024-1099 bandında doğru öngörüldüğü gözlemlenmektedir. Yapılan çalışmada hedef değişken için bir ve sıfır değerini alan eşit sayıda örnek ile çalışıldığı göz önünde bulundurulursa değişken alt kümesi seçim algoritmalarının, hedef değişkenin değerinin sıfır olarak kestirilmesine meyilli modellerin oluşmasını sağladığı söylenebilir.

3.8. Bulgular ve Tartışma

Tüm sonuçlar karşılaştırıldığında, mevcut veri kümesi ve analiz için kullanılan değişken alt kümesi seçim algoritmalarının yardımı ile oluşturulan değişken alt kümeleri ile elde edilen modele yakın düzeyde tutarlılık sağlanan, yakın düzeyde sınıflandırma doğruluğu elde edilen ve daha sade modeller oluşturulabildiği gözlemlenmektedir.

Eğer model başarısını arttırmaya yönelik bir alt küme belirleme çalışması yapılmak isteniyorsa, yöntem-1 ve yöntem-2'nin model doğruluğunu arttırmayı amaçlayan sarmalayıcı tipte algoritmaları kullanmasından dolayı tercih edilebileceği söylenebilir. Yöntem-1 'in bu konuda yardımcı olduğu söylenebilirken bu çalışma özelinde geri yönlü sarmalayıcı bir algoritmaya sahip olan yöntem-2'nin diğer yöntemlere nazaran çok iyi bir model doğruluğu sunduğu söylenemez. Bir diğer handikap ise, diğer algoritmaların seçim süresi 1 dakikanın altında iken yöntem-1 ve yöntem-2'nin sonuç verme sürelerinin uygulamadaki küçük bir veri kümesi için bile çok daha uzun sürede

sonuç döndürmüş olmasıdır. İleri yönlü çalışan yöntem-1, sekiz dakikada sonuçlanırken, geri yönlü çalışan yöntem-2 ise 4 saatin üzerinde bir sürede sonuçlanmıştır ve tüm veri kümesi ile yapılan analizin 0,52 saniyede sonuçlandığı göz önünde bulundurulursa bu değişken alt küme seçim yöntemlerinin, analizin değişken alt kümesi belirleme süreci ile birlikte tamamlanma süresini astronomik düzeyde arttırabildiği söylenebilir.

Çalışmada, yöntem-4 ile seçilen değişken alt kümesinin, en kısa sürede sonuç veren üç yöntemden biri olması ile birlikte, en sade modeli oluşturması ve tüm değişkenleri barındıran küme ile oluşturulan modelin üzerinde bir doğruluk elde etmesi nedeni ile bu çalışma için en uygun model olduğu söylenebilir.

SONUÇ

Bu çalışmada gerçek bankacılık müşteri verisi üzerinde veri madenciliği sınıflandırma çalışmasına yönelik değişken seçimi süreçleri üzerinde çalışılmıştır.

Çalışmanın başlangıcında veri setinde bulunan 103 adet değişken ve 13.274 banka müşterisine ait veriden oluşmakta olan veri kümesi üzerinden yapılan veri temizliği sürecinin ardından 100 değişken ve 2466 müşteriye ait kayıtlar çalışmaya dahil edilmiştir.

Uygulama aşamasında, iki adet sarmalayıcı değişken seçimi uygulaması ile beş adet filtreleyici değişken seçimi uygulaması olmak üzere toplamda yedi farklı değişken alt kümesi seçim çalışması yapılarak, yedi farklı değişken alt kümesi belirlenmiştir. Bu yedi değişken alt kümesi ve hiç değişken seçilmeden önce mevcut bulunan tüm değişkenlerin içinde bulunduğu veri kümesi ile Weka üzerinde bulunan J48 karar ağacı uygulamaları yapılarak sekiz adet karar ağacı elde edilmiştir. Değişken seçilmeksizin oluşturulan karar ağacı ile yöntemlerce seçilmiş yedi adet değişken alt kümesinden elde edilen değişkenlerin her biri ile ayrı ayrı oluşturulan karar ağaçları karşılaştırmalı olarak incelenmiştir. Bu inceleme ile uygulanan değişken seçim yöntemlerinin etkileri gözlemlenmeye çalışılmış ve bir diğer taraftan ise değişken seçimi yöntemleri kendi aralarında karşılaştırılarak göreceli olarak sonuçlar incelenmiştir.

Değişken seçimi uygulamaları ile, sınıflandırma doğruluğunu arttırma, toplam analiz süresini kısaltma ve daha sade modeller oluşturma amaçlarına ulaşılma durumları incelenmiştir.

Direk olarak sınıflandırma doğruluğunu arttırmayı amaçlayan sarmalayıcı yöntemlerin uygulamalarında, sınıflandırma doğruluğunu arttırma amacına ulaşıldığı gözlemlenmiş olmasına karşın toplam analiz süresinin filtreleyici modellere nazaran çok yüksek olduğu tespit edilmiştir.

Filtreleyici modeller kullanılarak yapılan değişken seçimi uygulamalarında ise toplam analiz süresinin düşürülmesi ve modelin sadeleştirilmesi amaçlanırken sınıflandırma doğruluğunun değişken seçimi yapılmadan önceki sınıflandırma doğruluğundan makul

ölçüde düşük olması kabul edilebilirdir. Bu çalışma kapsamında gerçekleştirilen filtreleyici değişken seçimi uygulamalarının tamamında sınıflandırma başarısının arttırılmıştır ve daha öncelikli hedefler olan kısa sürede seçilen değişkenler ile daha sade model oluşturma hedefine de ulaşılmıştır.

Tüm yöntemler ile gerçekleştirilen uygulamalar incelendiğinde değişken seçimi yöntemi olarak korelasyon bazlı değişken seçimi yönteminin, arama yöntemi olarak ise kazanç oranı bazlı değişken seçimini kullanan sıralayıcı arama yönteminin uygulamasının yapıldığı yöntem-4, diğer uygulamalar ile karşılaştırıldığında en kısa sürede, en az değişken ile, en sade modelin oluşturulmasını sağlamıştır. Toplam analiz süresinin fazla artmasının istenmediği durumlar için bu yöntemin oldukça makul olduğu söylenebilir.

Sınıflandırma doğruluğunun arttırılmasının amaçlanması halinde ise ileri yönlü BestFirst arama yöntemini ile birlikte sarmalayıcı modeli kullanan yöntem-1'in, uygulanan yöntemler arasında en iyi yöntem olarak görünmektedir.

Sonuç olarak uygulanan tüm yöntemlerin sınıflandırma doğruluğunu arttırdığı, sınıflandırma için gerekli süreyi ve oluşturulan modelin sadeleştirilmesi konusunda başarılı olduğu görülmektedir. Analiz için ayrılan süre ve kaynakların sınırlı olması durumunda kısa sürede kabul edilebilir bir iyileştirme sağladığı gözlemlenen değişken seçimi için filtreleyici modellerden korelasyon bazlı değişken seçimi uygulaması ile birlikte kazanç oranı bazlı sıralayıcı arama tercih edilmesi makul görünmektedir. Kaynakların ve analiz için ayrılan sürenin sınırlarının geniş olduğu durumlar için sarmalayıcı yöntemlerin ileri yönlü BestFirst arama yöntemi ile kullanılmasının makul olduğu söylenebilir.

EKLER

Ek-1: Yöntem-1'e Ait Weka Çıktısı

=== Run information ===

Evaluator: weka.attributeSelection.WrapperSubsetEval -B weka.classifiers.trees.J48 -F 5 -T 0.01 -R 1 -E DEFAULT -- -C 0.25 -M 2
Search: weka.attributeSelection.BestFirst -D 1 -N 5
Relation: e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances: 2466
Attributes: 100
[list of attributes omitted]
Evaluation mode: evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:
Best first.
Start set: no attributes
Search direction: forward
Stale search after 5 node expansions
Total number of subsets evaluated: 1654
Merit of best subset found: 0.822

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
Wrapper Subset Evaluator
Learning scheme: weka.classifiers.trees.J48
Scheme options: -C 0.25 -M 2
Subset evaluation: classification accuracy
Number of folds for accuracy estimation: 5

Selected attributes: 2,6,9,12,19,22,26,47,51,75,82 : 11
D02
M03
H02
H05
H12
H15
H19
C06
C12
Y16
U03

Ek-2: Yöntem-2'e Ait Weka Çıktısı

=== Run information ===

Evaluator: weka.attributeSelection.WrapperSubsetEval -B weka.classifiers.trees.J48 -F 5 -T 0.01 -R 1 -E DEFAULT -- -C 0.25 -M 2
Search: weka.attributeSelection.BestFirst -D 0 -N 5
Relation: e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances: 2466
Attributes: 100
[list of attributes omitted]
Evaluation mode: evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:
Best first.
Start set:
1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,1
Search direction: backward
Stale search after 5 node expansions
Total number of subsets evaluated: 6578
Merit of best subset found: 0.817

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
Wrapper Subset Evaluator
Learning scheme: weka.classifiers.trees.J48
Scheme options: -C 0.25 -M 2
Subset evaluation: classification accuracy
Number of folds for accuracy estimation: 5
Selected attributes:
2,4,6,7,8,9,12,14,15,16,17,18,19,22,25,26,30,37,39,45,46,48,50,51,52,53,54,55,56,59,61,62,63,65,66,67,68,70,71,72,75,76,80,81,82,83,86,89,90,92,93,94,95,96,97,100 : 56
D02
D04
M03
M04
H01
H02
H05
H07
H08
H09

H10
H11
H12
H15
H18
H19
H23
L07
L09
C03
C04
C07
C10
C12
C13
C15
C16
C18
C19
C22
Y02
Y03
Y04
Y06
Y07
Y08
Y09
Y11
Y12
Y13
Y16
Y17
U01
U02
U03
U04
U07
U10
U12
U14
U15
U16
U21
U22
U23
U20

Ek-3: Yöntem-3'e Ait Weka Çıktısı

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval -P 1 -E 1
Search:       weka.attributeSelection.BestFirst -D 1 -N 5
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances:    2466
Attributes:   100
              [list of attributes omitted]
Evaluation mode:    evaluate on all training data


=== Attribute Selection on all input data ===

Search Method:
        Best first.
        Start set: no attributes
        Search direction: forward
        Stale search after 5 node expansions
        Total number of subsets evaluated: 1213
        Merit of best subset found:    0.315

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
        CFS Subset Evaluator
        Including locally predictive attributes

Selected attributes: 4,6,7,9,21,42,47,51,55,82 : 10
D04
M03
M04
H02
H14
L12
C06
C12
C18
U03
```

Ek-4: Yöntem-4'e Ait Weka Çıktısı

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval -P 1 -E 1
Search:       weka.attributeSelection.RankSearch -S 1 -R 0 -N 0 -I 0.0 -X -A
weka.attributeSelection.GainRatioAttributeEval --
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances:    2466
Attributes:   100
              [list of attributes omitted]
Evaluation mode:    evaluate on all training data


=== Attribute Selection on all input data ===

Search Method:
        RankSearch :
        Attribute evaluator : weka.attributeSelection.GainRatioAttributeEval
        Attribute ranking :
         47 C06
         51 C12
          6 M03
         55 C18
         49 C09
          8 H01
         59 C22
         45 C03
         98 U17
         57 C21
         53 C15
         42 L12
         21 H14
          7 M04
         82 U03
         93 U15
         87 U08
          9 H02
         30 H23
         60 Y01
         65 Y06
         99 U19
         75 Y16
         19 H12
         29 H22
         24 H17
         13 H06
```

76 Y17
12 H05
23 H16
70 Y11
89 U10
72 Y13
15 H08
90 U12
28 H21
81 U02
92 U14
20 H13
14 H07
4 D04
62 Y03
67 Y08
26 H19
25 H18
88 U09
97 U23
100 U20
95 U21
84 U05
83 U04
86 U07
35 L05
32 L02
38 L08
22 H15
61 Y02
66 Y07
17 H10
71 Y12
11 H04
94 U16
18 H11
73 Y14
52 C13
56 C19
43 C01
48 C07
68 Y09
63 Y04
37 L07
80 U01
34 L04
31 L01

91 U13
39 L09
36 L06
33 L03
96 U22
5 M02
85 U06
69 Y10
64 Y05
74 Y15
54 C16
50 C10
1 D01
46 C04
3 D03
10 H03
2 D02
16 H09
58 C24
78 Y19
77 Y18
27 H20
41 L11
40 L10
79 Y20
Merit of best subset found : 0.301

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
CFS Subset Evaluator
Including locally predictive attributes

Selected attributes: 6,7,13,21,42,47,55,82 : 8
M03
M04
H06
H14
L12
C06
C18
U03

Ek-5: Yöntem-5'e Ait Weka Çıktısı

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval -P 1 -E 1
Search:       weka.attributeSelection.RankSearch -S 1 -R 0 -N 0 -I 0.0 -X -A
weka.attributeSelection.InfoGainAttributeEval --
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances:    2466
Attributes:   100
              [list of attributes omitted]
Evaluation mode:    evaluate on all training data


=== Attribute Selection on all input data ===

Search Method:
        RankSearch :
        Attribute evaluator : weka.attributeSelection.InfoGainAttributeEval
        Attribute ranking :
         55 C18
         47 C06
          7 M04
         51 C12
         59 C22
          4 D04
          6 M03
         19 H12
         20 H13
         93 U15
         82 U03
          9 H02
         87 U08
         25 H18
         26 H19
         81 U02
         92 U14
         70 Y11
         60 Y01
         65 Y06
         75 Y16
          8 H01
         14 H07
         15 H08
         53 C15
         45 C03
         94 U16
```

13 H06
24 H17
12 H05
23 H16
86 U07
83 U04
88 U09
91 U13
52 C13
28 H21
71 Y12
99 U19
22 H15
61 Y02
66 Y07
89 U10
17 H10
84 U05
11 H04
43 C01
80 U01
30 H23
48 C07
76 Y17
56 C19
49 C09
5 M02
72 Y13
85 U06
29 H22
1 D01
18 H11
73 Y14
67 Y08
62 Y03
95 U21
100 U20
98 U17
90 U12
57 C21
97 U23
63 Y04
68 Y09
42 L12
21 H14
37 L07
35 L05

38 L08
32 L02
39 L09
33 L03
36 L06
31 L01
34 L04
96 U22
3 D03
54 C16
50 C10
74 Y15
69 Y10
64 Y05
46 C04
2 D02
10 H03
16 H09
27 H20
79 Y20
78 Y19
58 C24
40 L10
77 Y18
41 L11
Merit of best subset found : 0.314

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
CFS Subset Evaluator
Including locally predictive attributes

Selected attributes: 4,6,7,9,19,21,42,47,51,55,93 : 11
D04
M03
M04
H02
H12
H14
L12
C06
C12
C18
U15

Ek-6: Yöntem-6'e Ait Weka Çıktısı

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval -P 1 -E 1
Search:       weka.attributeSelection.RankSearch -S 1 -R 0 -N 0 -I 0.0 -X -A
weka.attributeSelection.SymmetricalUncertAttributeEval --
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances:    2466
Attributes:   100
              [list of attributes omitted]
Evaluation mode:    evaluate on all training data


=== Attribute Selection on all input data ===

Search Method:
        RankSearch :
        Attribute evaluator : weka.attributeSelection.SymmetricalUncertAttributeEval
        Attribute ranking :
         47 C06
         51 C12
         55 C18
          6 M03
          7 M04
         59 C22
         82 U03
         93 U15
         19 H12
          9 H02
         87 U08
          4 D04
         20 H13
          8 H01
         60 Y01
         65 Y06
         53 C15
         81 U02
         92 U14
         70 Y11
         75 Y16
         45 C03
         26 H19
         25 H18
         15 H08
         14 H07
         24 H17
```

13 H06
23 H16
12 H05
88 U09
86 U07
99 U19
83 U04
28 H21
94 U16
89 U10
30 H23
84 U05
22 H15
71 Y12
66 Y07
61 Y02
17 H10
11 H04
52 C13
76 Y17
49 C09
43 C01
80 U01
72 Y13
91 U13
56 C19
48 C07
29 H22
62 Y03
67 Y08
100 U20
95 U21
5 M02
18 H11
98 U17
73 Y14
90 U12
85 U06
57 C21
97 U23
68 Y09
63 Y04
42 L12
21 H14
1 D01
37 L07
32 L02

35 L05
38 L08
31 L01
39 L09
33 L03
36 L06
34 L04
96 U22
54 C16
3 D03
50 C10
69 Y10
64 Y05
74 Y15
46 C04
10 H03
2 D02
16 H09
27 H20
77 Y18
79 Y20
58 C24
40 L10
78 Y19
41 L11
Merit of best subset found : 0.296

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
CFS Subset Evaluator
Including locally predictive attributes

Selected attributes: 6,7,13,21,42,47,59,82 : 8
M03
M04
H06
H14
L12
C06
C22
U03

Ek-7: Yöntem-7'e Ait Weka Çıktısı

```
=== Run information ===

Evaluator:    weka.attributeSelection.ConsistencySubsetEval
Search:       weka.attributeSelection.BestFirst -D 1 -N 5
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances:    2466
Attributes:   100
              [list of attributes omitted]
Evaluation mode:    evaluate on all training data


=== Attribute Selection on all input data ===

Search Method:
        Best first.
        Start set: no attributes
        Search direction: forward
        Stale search after 5 node expansions
        Total number of subsets evaluated: 2180
        Merit of best subset found:    0.998

Attribute Subset Evaluator (supervised, Class (nominal): 44 C02):
        Consistency Subset Evaluator

Selected attributes: 1,2,3,4,5,6,7,11,12,15,19,20,25,47,52,55,59,61,82,91 : 20
                     D01
                     D02
                     D03
                     D04
                     M02
                     M03
                     M04
                     H04
                     H05
                     H08
                     H12
                     H13
                     H18
                     C06
                     C13
                     C18
                     C22
                     Y02
                     U03
                     U13
```

Ek-8 : Değişken Seçimi Yapılmadan Gerçekleştirilen Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59
Instances:    2466
Attributes:   100
              [list of attributes omitted]
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|  C15 = 0
|  |  U03 = 0
|  |  |  C19 <= -21611
|  |  |  |  M03 = 2: 0 (4.0)
|  |  |  |  M03 = 1
|  |  |  |  |  C04 = 0
|  |  |  |  |  |  Y17 <= 38
|  |  |  |  |  |  |  U06 = 1
|  |  |  |  |  |  |  |  U02 = 0: 0 (6.0/2.0)
|  |  |  |  |  |  |  |  U02 = 1: 1 (2.0)
|  |  |  |  |  |  |  U06 = 0: 1 (9.0)
|  |  |  |  |  |  Y17 > 38: 0 (3.0)
|  |  |  |  |  C04 = 1: 0 (2.0)
|  |  |  C19 > -21611: 0 (893.0/87.0)
|  |  U03 = 1
|  |  |  M03 = 2
|  |  |  |  C19 <= -18753: 1 (3.0/1.0)
|  |  |  |  C19 > -18753: 0 (33.0)
|  |  |  M03 = 1
|  |  |  |  Y16 <= 147736
|  |  |  |  |  Y14 = 0
|  |  |  |  |  |  H23 = 0
|  |  |  |  |  |  |  H13 <= 2
|  |  |  |  |  |  |  |  D03 = 1
|  |  |  |  |  |  |  |  |  U16 = 0
|  |  |  |  |  |  |  |  |  |  M04 = 6: 0 (4.0)
|  |  |  |  |  |  |  |  |  |  M04 = 4: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  M04 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  M04 = 2: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  M04 = 3: 0 (4.0)
|  |  |  |  |  |  |  |  |  |  M04 = -1: 0 (0.0)
```

```
|  |  |  |  |  |  |  |  |  |  M04 = 7: 1 (5.0)
|  |  |  |  |  |  |  |  |  |  M04 = 5
|  |  |  |  |  |  |  |  |  |  |  U07 = 0: 1 (4.0)
|  |  |  |  |  |  |  |  |  |  |  U07 = 1: 0 (7.0/1.0)
|  |  |  |  |  |  |  |  |  |  M04 = 1: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 5: 1 (1.0)
|  |  |  |  |  |  |  |  |  U16 = 2: 1 (1.0)
|  |  |  |  |  |  |  |  |  U16 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 3: 1 (2.0)
|  |  |  |  |  |  |  |  |  U16 = 7: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 1: 0 (2.0)
|  |  |  |  |  |  |  |  |  U16 = 4: 0 (1.0)
|  |  |  |  |  |  |  |  |  U16 = 9: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 18: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 16: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 12: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 10: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 13: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 19: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 14: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 11: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 22: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 27: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 17: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 35: 0 (0.0)
|  |  |  |  |  |  |  |  |  U16 = 15: 0 (0.0)
|  |  |  |  |  |  |  |  D03 = 2
|  |  |  |  |  |  |  |  |  H19 <= 189152: 0 (60.0/20.0)
|  |  |  |  |  |  |  |  |  H19 > 189152: 1 (4.0/1.0)
|  |  |  |  |  |  |  |  D03 = 3: 1 (4.0)
|  |  |  |  |  |  |  H13 > 2
|  |  |  |  |  |  |  |  Y17 <= 65475: 1 (13.0/1.0)
|  |  |  |  |  |  |  |  Y17 > 65475: 0 (2.0)
|  |  |  |  |  |  H23 = 1: 1 (6.0/1.0)
|  |  |  |  |  Y14 = 1: 0 (2.0/1.0)
|  |  |  |  |  Y14 = 5: 0 (0.0)
|  |  |  |  |  Y14 = 2: 1 (2.0)
|  |  |  |  |  Y14 = 4: 0 (0.0)
|  |  |  |  |  Y14 = 7: 0 (0.0)
|  |  |  |  |  Y14 = 3: 0 (0.0)
|  |  |  |  |  Y14 = 6: 0 (1.0)
|  |  |  |  |  Y14 = 9: 0 (0.0)
|  |  |  |  |  Y14 = 10: 0 (0.0)
|  |  |  |  |  Y14 = 8: 0 (0.0)
|  |  |  |  Y16 > 147736: 1 (15.0)
|  C15 = 1
```

```
|  |  Y01 = 0
|  |  |  C07 = 0: 0 (8.0/1.0)
|  |  |  C07 = 1: 1 (6.0/1.0)
|  |  Y01 = 1: 1 (7.0)
|  C15 = 2: 1 (5.0)
C06 = 1
|  M03 = 2
|  |  H19 <= 251
|  |  |  C22 <= -57136: 1 (3.0)
|  |  |  C22 > -57136
|  |  |  |  D03 = 1
|  |  |  |  |  C22 <= -21181: 1 (2.0)
|  |  |  |  |  C22 > -21181: 0 (8.0)
|  |  |  |  D03 = 2: 0 (19.0/2.0)
|  |  |  |  D03 = 3: 1 (3.0/1.0)
|  |  H19 > 251: 1 (2.0)
|  M03 = 1
|  |  C03 = 0
|  |  |  H23 = 0
|  |  |  |  H21 <= 6134421
|  |  |  |  |  C22 <= -44802: 1 (242.0/24.0)
|  |  |  |  |  C22 > -44802
|  |  |  |  |  |  D04 = 3
|  |  |  |  |  |  |  L01 = 0: 1 (193.0/67.0)
|  |  |  |  |  |  |  L01 = 1: 0 (4.0/1.0)
|  |  |  |  |  |  D04 = -1: 1 (0.0)
|  |  |  |  |  |  D04 = 5
|  |  |  |  |  |  |  M04 = 6: 0 (1.0)
|  |  |  |  |  |  |  M04 = 4: 0 (2.0/1.0)
|  |  |  |  |  |  |  M04 = 8
|  |  |  |  |  |  |  |  U10 = 0: 1 (116.0/17.0)
|  |  |  |  |  |  |  |  U10 = 1
|  |  |  |  |  |  |  |  |  Y16 <= 105748: 0 (4.0)
|  |  |  |  |  |  |  |  |  Y16 > 105748: 1 (4.0)
|  |  |  |  |  |  |  M04 = 2: 1 (46.0/9.0)
|  |  |  |  |  |  |  M04 = 3
|  |  |  |  |  |  |  |  H12 <= 1
|  |  |  |  |  |  |  |  |  H19 <= 51
|  |  |  |  |  |  |  |  |  |  U01 = 1: 0 (13.0/1.0)
|  |  |  |  |  |  |  |  |  |  U01 = 0
|  |  |  |  |  |  |  |  |  |  |  C13 = 0
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 0: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 1: 0 (9.0/2.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 2: 1 (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 4: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 3: 1 (2.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 5: 0 (0.0)
```

```
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 7: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 9: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 10: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 1: 1 (2.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 2: 0 (1.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 4: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 3: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  |  H19 > 51: 1 (2.0)
|  |  |  |  |  |  |  |  H12 > 1: 1 (13.0/3.0)
|  |  |  |  |  |  |  M04 = -1: 1 (0.0)
|  |  |  |  |  |  |  M04 = 7
|  |  |  |  |  |  |  |  H20 <= 96
|  |  |  |  |  |  |  |  |  H10 = 0
|  |  |  |  |  |  |  |  |  |  H09 = 0
|  |  |  |  |  |  |  |  |  |  |  U03 = 0
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 0: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 1
|  |  |  |  |  |  |  |  |  |  |  |  |  M02 <= 10: 1 (16.0/2.0)
|  |  |  |  |  |  |  |  |  |  |  |  |  M02 > 10: 0 (2.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 2
|  |  |  |  |  |  |  |  |  |  |  |  |  H07 = 0
|  |  |  |  |  |  |  |  |  |  |  |  |  |  M02 <= 5: 0 (4.0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  M02 > 5
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  H18 <= 909: 1 (7.0/1.0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  H18 > 909: 0 (2.0)
|  |  |  |  |  |  |  |  |  |  |  |  |  H07 = 1: 0 (2.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 4: 1 (1.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 3: 1 (5.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 5: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 7: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 6: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 9: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 8: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  C18 = 10: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  U03 = 1: 1 (45.0/6.0)
|  |  |  |  |  |  |  |  |  |  H09 = 1: 0 (3.0/1.0)
|  |  |  |  |  |  |  |  |  H10 = 1: 0 (6.0/2.0)
|  |  |  |  |  |  |  |  H20 > 96: 1 (10.0)
|  |  |  |  |  |  |  M04 = 5: 1 (47.0/6.0)
|  |  |  |  |  |  |  M04 = 1: 1 (16.0/2.0)
|  |  |  |  |  |  D04 = 6
|  |  |  |  |  |  |  C07 = 0: 1 (118.0/18.0)
|  |  |  |  |  |  |  C07 = 1
|  |  |  |  |  |  |  |  U13 = 1
```

```
|  |  |  |  |  |  |  |  |  H07 = 0
|  |  |  |  |  |  |  |  |  |  H12 <= 1: 0 (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  H12 > 1: 1 (3.0)
|  |  |  |  |  |  |  |  |  H07 = 1: 0 (3.0)
|  |  |  |  |  |  |  |  U13 = 0: 0 (2.0)
|  |  |  |  |  |  |  |  U13 = 2: 1 (2.0)
|  |  |  |  |  |  |  |  U13 = 4: 0 (2.0)
|  |  |  |  |  |  |  |  U13 = 3: 1 (3.0)
|  |  |  |  |  |  |  |  U13 = 18: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 5: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 12: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 7: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 11: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 9: 0 (0.0)
|  |  |  |  |  |  D04 = 1
|  |  |  |  |  |  |  D03 = 1: 1 (7.0/1.0)
|  |  |  |  |  |  |  D03 = 2
|  |  |  |  |  |  |  |  Y12 = 0
|  |  |  |  |  |  |  |  |  U05 = 0
|  |  |  |  |  |  |  |  |  |  H19 <= 546: 0 (24.0/3.0)
|  |  |  |  |  |  |  |  |  |  H19 > 546: 1 (5.0/1.0)
|  |  |  |  |  |  |  |  |  U05 = 1: 1 (2.0)
|  |  |  |  |  |  |  |  Y12 = 1: 0 (4.0)
|  |  |  |  |  |  |  |  Y12 = 2: 1 (1.0)
|  |  |  |  |  |  |  |  Y12 = 3: 0 (0.0)
|  |  |  |  |  |  |  |  Y12 = 4: 0 (0.0)
|  |  |  |  |  |  |  D03 = 3: 0 (2.0/1.0)
|  |  |  |  |  |  D04 = 2
|  |  |  |  |  |  |  Y04 = 0
|  |  |  |  |  |  |  |  U16 = 0
|  |  |  |  |  |  |  |  |  H07 = 0
|  |  |  |  |  |  |  |  |  |  U05 = 0
|  |  |  |  |  |  |  |  |  |  |  C13 = 0: 1 (42.0/15.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 1
|  |  |  |  |  |  |  |  |  |  |  |  C12 = 0: 0 (5.0)
|  |  |  |  |  |  |  |  |  |  |  |  C12 = 1: 1 (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 2: 1 (2.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 4: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 3: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  |  C13 = 6: 1 (0.0)
|  |  |  |  |  |  |  |  |  |  U05 = 1: 0 (4.0)
|  |  |  |  |  |  |  |  |  H07 = 1: 1 (4.0)
|  |  |  |  |  |  |  |  U16 = 5: 0 (1.0)
|  |  |  |  |  |  |  |  U16 = 2: 1 (1.0)
|  |  |  |  |  |  |  |  U16 = 6: 0 (2.0/1.0)
```

```
|  |  |  |  |  |  |  |  U16 = 3: 0 (2.0)
|  |  |  |  |  |  |  |  U16 = 7: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 1: 0 (2.0)
|  |  |  |  |  |  |  |  U16 = 4: 1 (2.0)
|  |  |  |  |  |  |  |  U16 = 9: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 8: 1 (1.0)
|  |  |  |  |  |  |  |  U16 = 18: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 16: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 12: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 10: 0 (1.0)
|  |  |  |  |  |  |  |  U16 = 13: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 19: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 14: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 11: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 22: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 27: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 17: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 35: 1 (0.0)
|  |  |  |  |  |  |  |  U16 = 15: 1 (0.0)
|  |  |  |  |  |  |  Y04 = 1: 0 (2.0)
|  |  |  |  |  |  D04 = 7
|  |  |  |  |  |  |  H19 <= 68871: 0 (2.0)
|  |  |  |  |  |  |  H19 > 68871: 1 (3.0)
|  |  |  |  |  |  D04 = 4
|  |  |  |  |  |  |  Y02 = 0: 1 (21.0/2.0)
|  |  |  |  |  |  |  Y02 = 1
|  |  |  |  |  |  |  |  M02 <= 9: 1 (3.0)
|  |  |  |  |  |  |  |  M02 > 9: 0 (2.0)
|  |  |  |  H21 > 6134421: 1 (32.0)
|  |  |  H23 = 1: 1 (56.0/2.0)
|  |  C03 = 1: 1 (111.0/2.0)

Number of Leaves  :   184

Size of the tree :        252


Time taken to build model: 0.52 seconds

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

Correctly Classified Instances        1963               79.6026 %
Incorrectly Classified Instances       503               20.3974 %
Kappa statistic                          0.5921
Mean absolute error                      0.2596
Root mean squared error                  0.406
```

Relative absolute error 51.9139 %
Root relative squared error 81.2069 %
Total Number of Instances 2466

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,766	0,174	0,815	0,766	0,790	0,593	0,826	0,778	0
	0,826	0,234	0,779	0,826	0,802	0,593	0,826	0,777	1
Weighted Avg.	0,796	0,204	0,797	0,796	0,796	0,593	0,826	0,778	

=== Confusion Matrix ===

a	b	<-- classified as
945	288	a = 0
215	1018	b = 1

Ek-9 : Yöntem-1 'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-
weka.filters.unsupervised.attribute.Remove-V-R2,6,9,12,19,22,26,47,51,75,82,44
Instances:    2466
Attributes:   12
              D02
              M03
              H02
              H05
              H12
              H15
              H19
              C06
              C12
              Y16
              U03
              C02
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|  U03 = 0: 0 (919.0/100.0)
|  U03 = 1
|  |  M03 = 2: 0 (38.0/2.0)
|  |  M03 = 1
|  |  |  Y16 <= 147736
|  |  |  |  H02 = 1
|  |  |  |  |  H05 = 0
|  |  |  |  |  |  D02 = 1
|  |  |  |  |  |  |  H12 <= 2
|  |  |  |  |  |  |  |  H15 <= 0
|  |  |  |  |  |  |  |  |  H19 <= 36800: 0 (23.0/4.0)
|  |  |  |  |  |  |  |  |  H19 > 36800: 1 (6.0/1.0)
|  |  |  |  |  |  |  |  H15 > 0
|  |  |  |  |  |  |  |  |  H15 <= 1: 1 (7.0/1.0)
|  |  |  |  |  |  |  |  |  H15 > 1: 0 (3.0/1.0)
|  |  |  |  |  |  |  H12 > 2: 1 (16.0/3.0)
|  |  |  |  |  |  D02 = 2
|  |  |  |  |  |  |  H12 <= 2: 1 (21.0/4.0)
|  |  |  |  |  |  |  H12 > 2: 0 (4.0)
```

```
|  |  |  |  |  H05 = 1: 1 (17.0/2.0)
|  |  |  |  H02 = 0: 0 (50.0/16.0)
|  |  |  Y16 > 147736: 1 (17.0)
C06 = 1
|  M03 = 2
|  |  H19 <= 251: 0 (35.0/9.0)
|  |  H19 > 251: 1 (2.0)
|  M03 = 1: 1 (1308.0/282.0)

Number of Leaves  :  15

Size of the tree :  29


Time taken to build model: 0.08 seconds

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

Correctly Classified Instances        2025               82.1168 %
Incorrectly Classified Instances       441               17.8832 %
Kappa statistic                          0.6423
Mean absolute error                      0.283
Root mean squared error                  0.3797
Relative absolute error                 56.6042 %
Root relative squared error             75.9316 %
Total Number of Instances             2466

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0,758    0,115    0,868      0,758    0,809      0,648    0,813     0,803     0
               0,885    0,242    0,785      0,885    0,832      0,648    0,813     0,748     1
Weighted Avg.  0,821    0,179    0,826      0,821    0,820      0,648    0,813     0,776

=== Confusion Matrix ===

   a    b   <-- classified as
 934  299 |    a = 0
 142 1091 |    b = 1
```

Ek-10 : Yöntem-2'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2
Relation: e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-weka.filters.unsupervised.attribute.Remove-V-R2,4,6-9,12,14-19,22,25-26,30,37,39,45-46,48,50-56,59,61-63,65-68,70-72,75-76,80-83,86,89-90,92-97,100,44
Instances: 2466
Attributes: 57

D02
D04
M03
M04
H01
H02
H05
H07
H08
H09
H10
H11
H12
H15
H18
H19
H23
L07
L09
C03
C04
C07
C10
C12
C13
C15
C16
C18
C19
C22
Y02
Y03
Y04
Y06
Y07
Y08
Y09
Y11

Y12
Y13
Y16
Y17
U01
U02
U03
U04
U07
U10
U12
U14
U15
U16
U21
U22
U23
U20
C02
Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree

```
C12 = 0
|  C18 = 0
|  |  C15 = 0
|  |  |  U03 = 0
|  |  |  |  C19 <= -23117
|  |  |  |  |  M03 = 2: 0 (4.0)
|  |  |  |  |  M03 = 1
|  |  |  |  |  |  Y17 <= 38: 1 (18.0/5.0)
|  |  |  |  |  |  Y17 > 38: 0 (3.0)
|  |  |  |  C19 > -23117: 0 (894.0/87.0)
|  |  |  U03 = 1
|  |  |  |  M03 = 2
|  |  |  |  |  C19 <= -18753: 1 (3.0/1.0)
|  |  |  |  |  C19 > -18753: 0 (33.0)
|  |  |  |  M03 = 1
|  |  |  |  |  Y16 <= 147736
|  |  |  |  |  |  M04 = 6: 0 (12.0/1.0)
|  |  |  |  |  |  M04 = 4: 0 (0.0)
|  |  |  |  |  |  M04 = 8
|  |  |  |  |  |  |  H10 = 0
|  |  |  |  |  |  |  |  D02 = 1
```

```
|  |  |  |  |  |  |  |  |  H12 <= 2
|  |  |  |  |  |  |  |  |  |  H19 <= 50104
|  |  |  |  |  |  |  |  |  |  |  H18 <= 8: 1 (2.0)
|  |  |  |  |  |  |  |  |  |  |  H18 > 8: 0 (8.0/1.0)
|  |  |  |  |  |  |  |  |  |  H19 > 50104: 1 (3.0)
|  |  |  |  |  |  |  |  |  H12 > 2: 1 (2.0)
|  |  |  |  |  |  |  |  D02 = 2: 1 (3.0)
|  |  |  |  |  |  |  H10 = 1: 1 (6.0)
|  |  |  |  |  |  M04 = 2
|  |  |  |  |  |  |  H09 = 0
|  |  |  |  |  |  |  |  U07 = 0: 1 (3.0/1.0)
|  |  |  |  |  |  |  |  U07 = 1: 0 (5.0)
|  |  |  |  |  |  |  H09 = 1: 1 (2.0)
|  |  |  |  |  |  M04 = 3: 0 (18.0/3.0)
|  |  |  |  |  |  M04 = -1: 0 (0.0)
|  |  |  |  |  |  M04 = 7
|  |  |  |  |  |  |  C04 = 0: 1 (31.0/11.0)
|  |  |  |  |  |  |  C04 = 1: 0 (2.0)
|  |  |  |  |  |  M04 = 5
|  |  |  |  |  |  |  H10 = 0
|  |  |  |  |  |  |  |  Y02 = 0
|  |  |  |  |  |  |  |  |  H08 = 1: 1 (3.0)
|  |  |  |  |  |  |  |  |  H08 = 0
|  |  |  |  |  |  |  |  |  |  U01 = 1
|  |  |  |  |  |  |  |  |  |  |  U07 = 0: 1 (3.0)
|  |  |  |  |  |  |  |  |  |  |  U07 = 1
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 0: 0 (6.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 1: 1 (2.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 2: 0 (1.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 5: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 3: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 4: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  |  |  Y11 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  |  |  U01 = 0: 1 (2.0)
|  |  |  |  |  |  |  |  Y02 = 1: 0 (2.0)
|  |  |  |  |  |  |  H10 = 1: 1 (2.0)
|  |  |  |  |  |  M04 = 1: 1 (7.0/3.0)
|  |  |  |  |  Y16 > 147736: 1 (15.0)
|  |  C15 = 1
|  |  |  Y06 = 0
|  |  |  |  C07 = 0: 0 (8.0/1.0)
|  |  |  |  C07 = 1: 1 (6.0/1.0)
|  |  |  Y06 = 1: 1 (7.0)
|  |  C15 = 2: 1 (5.0)
|  C18 = 1
|  |  M03 = 2: 0 (7.0/1.0)
```

```
|  |  M03 = 1
|  |  |  D04 = 3
|  |  |  |  H05 = 0
|  |  |  |  |  U04 = 0
|  |  |  |  |  |  H02 = 1: 0 (19.0/8.0)
|  |  |  |  |  |  H02 = 0: 1 (18.0/3.0)
|  |  |  |  |  U04 = 1: 0 (2.0)
|  |  |  |  H05 = 1: 1 (3.0)
|  |  |  D04 = -1: 1 (0.0)
|  |  |  D04 = 5
|  |  |  |  Y17 <= 55953: 1 (97.0/23.0)
|  |  |  |  Y17 > 55953
|  |  |  |  |  U02 = 0: 1 (5.0/1.0)
|  |  |  |  |  U02 = 1: 0 (4.0)
|  |  |  D04 = 6
|  |  |  |  D02 = 1
|  |  |  |  |  H08 = 1: 1 (3.0)
|  |  |  |  |  H08 = 0
|  |  |  |  |  |  H18 <= 7324: 0 (10.0/2.0)
|  |  |  |  |  |  H18 > 7324: 1 (5.0)
|  |  |  |  D02 = 2: 1 (12.0)
|  |  |  D04 = 1
|  |  |  |  H19 <= 33: 0 (11.0)
|  |  |  |  H19 > 33
|  |  |  |  |  H08 = 1: 0 (3.0/1.0)
|  |  |  |  |  H08 = 0: 1 (3.0)
|  |  |  D04 = 2
|  |  |  |  C22 <= -8711: 1 (3.0)
|  |  |  |  C22 > -8711
|  |  |  |  |  D02 = 1
|  |  |  |  |  |  H10 = 0
|  |  |  |  |  |  |  U02 = 0: 1 (5.0/1.0)
|  |  |  |  |  |  |  U02 = 1: 0 (2.0)
|  |  |  |  |  |  H10 = 1: 0 (2.0)
|  |  |  |  |  D02 = 2: 0 (9.0)
|  |  |  D04 = 7: 1 (3.0)
|  |  |  D04 = 4: 1 (10.0/1.0)
|  C18 = 2
|  |  M03 = 2: 0 (5.0/1.0)
|  |  M03 = 1
|  |  |  C07 = 0: 1 (47.0/9.0)
|  |  |  C07 = 1
|  |  |  |  H02 = 1: 1 (12.0/1.0)
|  |  |  |  H02 = 0: 0 (7.0/1.0)
|  C18 = 4: 1 (4.0/1.0)
|  C18 = 3
|  |  C07 = 0: 1 (10.0/1.0)
```

```
|  |  C07 = 1: 0 (2.0)
|  C18 = 5: 1 (2.0)
|  C18 = 7: 0 (0.0)
|  C18 = 6: 0 (0.0)
|  C18 = 9: 0 (0.0)
|  C18 = 8: 0 (0.0)
|  C18 = 10: 0 (0.0)
C12 = 1
|  M03 = 2
|  |  H19 <= 251: 0 (23.0/7.0)
|  |  H19 > 251: 1 (2.0)
|  M03 = 1
|  |  C16 = 0
|  |  |  H02 = 1: 1 (634.0/77.0)
|  |  |  H02 = 0
|  |  |  |  U21 = 0
|  |  |  |  |  C22 <= -45289: 1 (101.0/13.0)
|  |  |  |  |  C22 > -45289
|  |  |  |  |  |  C13 = 0
|  |  |  |  |  |  |  M04 = 6
|  |  |  |  |  |  |  |  H05 = 0
|  |  |  |  |  |  |  |  |  D04 = 3: 0 (11.0/4.0)
|  |  |  |  |  |  |  |  |  D04 = -1: 1 (0.0)
|  |  |  |  |  |  |  |  |  D04 = 5: 1 (0.0)
|  |  |  |  |  |  |  |  |  D04 = 6: 1 (0.0)
|  |  |  |  |  |  |  |  |  D04 = 1: 1 (0.0)
|  |  |  |  |  |  |  |  |  D04 = 2: 1 (8.0/2.0)
|  |  |  |  |  |  |  |  |  D04 = 7: 1 (0.0)
|  |  |  |  |  |  |  |  |  D04 = 4: 1 (0.0)
|  |  |  |  |  |  |  |  H05 = 1: 0 (2.0)
|  |  |  |  |  |  |  M04 = 4: 0 (5.0/1.0)
|  |  |  |  |  |  |  M04 = 8: 1 (39.0/6.0)
|  |  |  |  |  |  |  M04 = 2
|  |  |  |  |  |  |  |  C22 <= -17490: 0 (2.0)
|  |  |  |  |  |  |  |  C22 > -17490: 1 (19.0/1.0)
|  |  |  |  |  |  |  M04 = 3
|  |  |  |  |  |  |  |  H01 = 1
|  |  |  |  |  |  |  |  |  C18 = 0: 0 (0.0)
|  |  |  |  |  |  |  |  |  C18 = 1
|  |  |  |  |  |  |  |  |  |  U02 = 0
|  |  |  |  |  |  |  |  |  |  |  C22 <= -26142: 1 (4.0)
|  |  |  |  |  |  |  |  |  |  |  C22 > -26142: 0 (18.0/5.0)
|  |  |  |  |  |  |  |  |  |  U02 = 1: 0 (2.0)
|  |  |  |  |  |  |  |  |  C18 = 2: 1 (10.0/2.0)
|  |  |  |  |  |  |  |  |  C18 = 4: 0 (0.0)
|  |  |  |  |  |  |  |  |  C18 = 3
|  |  |  |  |  |  |  |  |  |  U01 = 1: 0 (4.0)
```

```
| | | | | | | | | | U01 = 0: 1 (2.0)
| | | | | | | | | C18 = 5: 0 (0.0)
| | | | | | | | | C18 = 7: 0 (0.0)
| | | | | | | | | C18 = 6: 0 (0.0)
| | | | | | | | | C18 = 9: 0 (0.0)
| | | | | | | | | C18 = 8: 0 (0.0)
| | | | | | | | | C18 = 10: 0 (0.0)
| | | | | | | | H01 = 0: 0 (2.0)
| | | | | | | M04 = -1: 1 (0.0)
| | | | | | | M04 = 7: 1 (36.0/11.0)
| | | | | | | M04 = 5: 1 (14.0/2.0)
| | | | | | | M04 = 1: 1 (0.0)
| | | | | | C13 = 1
| | | | | | | C18 = 0: 0 (0.0)
| | | | | | | C18 = 1: 0 (40.0/15.0)
| | | | | | | C18 = 2: 1 (11.0/2.0)
| | | | | | | C18 = 4: 1 (1.0)
| | | | | | | C18 = 3: 1 (2.0)
| | | | | | | C18 = 5: 0 (0.0)
| | | | | | | C18 = 7: 0 (0.0)
| | | | | | | C18 = 6: 0 (0.0)
| | | | | | | C18 = 9: 0 (0.0)
| | | | | | | C18 = 8: 0 (0.0)
| | | | | | | C18 = 10: 0 (0.0)
| | | | | | C13 = 2: 1 (12.0/2.0)
| | | | | | C13 = 4: 1 (0.0)
| | | | | | C13 = 3: 1 (0.0)
| | | | | | C13 = 6: 1 (0.0)
| | | | U21 = 1: 0 (3.0/1.0)
| | C16 = 1: 1 (12.0)
| | C16 = 2: 0 (1.0)
| | C16 = 3: 1 (0.0)

Number of Leaves  :  131

Size of the tree :  195


Time taken to build model: 0.22 seconds

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

Correctly Classified Instances        1989               80.6569 %
Incorrectly Classified Instances       477               19.3431 %
Kappa statistic                          0.6131
Mean absolute error                      0.2605
```

Root mean squared error 0.3908
Relative absolute error 52.1023 %
Root relative squared error 78.165 %
Total Number of Instances 2466

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,783	0,170	0,822	0,783	0,802	0,614	0,844	0,814	0
	0,830	0,217	0,793	0,830	0,811	0,614	0,844	0,798	1
Weighted Avg.	0,807	0,193	0,807	0,807	0,806	0,614	0,844	0,806	

=== Confusion Matrix ===

a	b	<-- classified as
965	268	a = 0
209	1024	b = 1

Ek-11 : Yöntem-3'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-
weka.filters.unsupervised.attribute.Remove-V-R4,6-7,9,21,42,47,51,55,82,44
Instances:    2466
Attributes:   11
              D04
              M03
              M04
              H02
              H14
              L12
              C06
              C12
              C18
              U03
              C02
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|   U03 = 0: 0 (919.0/100.0)
|   U03 = 1
|   |   M04 = 6: 0 (34.0/1.0)
|   |   M04 = 4: 0 (3.0)
|   |   M04 = 8: 1 (31.0/7.0)
|   |   M04 = 2: 0 (12.0/4.0)
|   |   M04 = 3: 0 (26.0/5.0)
|   |   M04 = -1: 0 (0.0)
|   |   M04 = 7: 1 (53.0/17.0)
|   |   M04 = 5
|   |   |   D04 = 3: 0 (3.0/1.0)
|   |   |   D04 = -1: 1 (0.0)
|   |   |   D04 = 5: 1 (20.0/6.0)
|   |   |   D04 = 6: 0 (5.0/1.0)
|   |   |   D04 = 1: 1 (0.0)
|   |   |   D04 = 2: 1 (0.0)
|   |   |   D04 = 7: 1 (0.0)
|   |   |   D04 = 4: 0 (3.0/1.0)
|   |   M04 = 1: 1 (12.0/3.0)
C06 = 1
```

```
|   H02 = 1: 1 (837.0/132.0)
|   H02 = 0
|   |   M03 = 2: 0 (29.0/7.0)
|   |   M03 = 1
|   |   |   D04 = 3: 1 (133.0/46.0)
|   |   |   D04 = -1: 1 (0.0)
|   |   |   D04 = 5: 1 (196.0/49.0)
|   |   |   D04 = 6: 1 (58.0/15.0)
|   |   |   D04 = 1
|   |   |   |   C12 = 0: 0 (8.0)
|   |   |   |   C12 = 1
|   |   |   |   |   M04 = 6: 0 (2.0)
|   |   |   |   |   M04 = 4: 0 (2.0)
|   |   |   |   |   M04 = 8: 0 (4.0/1.0)
|   |   |   |   |   M04 = 2: 0 (0.0)
|   |   |   |   |   M04 = 3
|   |   |   |   |   |   C18 = 0: 1 (0.0)
|   |   |   |   |   |   C18 = 1: 0 (3.0/1.0)
|   |   |   |   |   |   C18 = 2: 1 (2.0)
|   |   |   |   |   |   C18 = 4: 1 (0.0)
|   |   |   |   |   |   C18 = 3: 1 (0.0)
|   |   |   |   |   |   C18 = 5: 1 (0.0)
|   |   |   |   |   |   C18 = 7: 1 (0.0)
|   |   |   |   |   |   C18 = 6: 1 (0.0)
|   |   |   |   |   |   C18 = 9: 1 (0.0)
|   |   |   |   |   |   C18 = 8: 1 (0.0)
|   |   |   |   |   |   C18 = 10: 1 (0.0)
|   |   |   |   |   M04 = -1: 0 (0.0)
|   |   |   |   |   M04 = 7: 1 (4.0)
|   |   |   |   |   M04 = 5: 0 (0.0)
|   |   |   |   |   M04 = 1: 0 (0.0)
|   |   |   D04 = 2
|   |   |   |   H14 <= 0
|   |   |   |   |   C12 = 0: 0 (13.0/5.0)
|   |   |   |   |   C12 = 1: 1 (43.0/14.0)
|   |   |   |   H14 > 0: 0 (3.0)
|   |   |   D04 = 7: 0 (1.0)
|   |   |   D04 = 4: 1 (7.0/1.0)

Number of Leaves  :  48

Size of the tree :  60


Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===
```

=== Summary ===

Correctly Classified Instances	2005	81.3058 %
Incorrectly Classified Instances	461	18.6942 %
Kappa statistic	0.6261	
Mean absolute error	0.2743	
Root mean squared error	0.3795	
Relative absolute error	54.8627 %	
Root relative squared error	75.8901 %	
Total Number of Instances	2466	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,740	0,114	0,867	0,740	0,798	0,633	0,853	0,866	0
	0,886	0,260	0,773	0,886	0,826	0,633	0,853	0,794	1
Weighted Avg.	0,813	0,187	0,820	0,813	0,812	0,633	0,853	0,830	

=== Confusion Matrix ===

a	b	<-- classified as
912	321	a = 0
140	1093	b = 1

Ek-12 : Yöntem-4'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-
weka.filters.unsupervised.attribute.Remove-V-R6-7,13,21,42,47,55,82,44
Instances:    2466
Attributes:   9
              M03
              M04
              H06
              H14
              L12
              C06
              C18
              U03
              C02
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|   U03 = 0: 0 (919.0/100.0)
|   U03 = 1
|   |   M04 = 6: 0 (34.0/1.0)
|   |   M04 = 4: 0 (3.0)
|   |   M04 = 8: 1 (31.0/7.0)
|   |   M04 = 2
|   |   |   H14 <= 0: 0 (9.0/2.0)
|   |   |   H14 > 0: 1 (3.0/1.0)
|   |   M04 = 3: 0 (26.0/5.0)
|   |   M04 = -1: 0 (0.0)
|   |   M04 = 7: 1 (53.0/17.0)
|   |   M04 = 5: 1 (31.0/14.0)
|   |   M04 = 1
|   |   |   H14 <= 0: 1 (9.0/1.0)
|   |   |   H14 > 0: 0 (3.0/1.0)
C06 = 1: 1 (1345.0/308.0)

Number of Leaves  :   13

Size of the tree :        18
```

Time taken to build model: 0.03 seconds

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

Correctly Classified Instances	1997	80.9813 %
Incorrectly Classified Instances	469	19.0187 %
Kappa statistic	0.6196	
Mean absolute error	0.2913	
Root mean squared error	0.3855	
Relative absolute error	58.2607 %	
Root relative squared error	77.0952 %	
Total Number of Instances	2466	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,728	0,109	0,870	0,728	0,793	0,628	0,811	0,802	0
	0,891	0,272	0,766	0,891	0,824	0,628	0,811	0,751	1
Weighted Avg.	0,810	0,190	0,818	0,810	0,809	0,628	0,811	0,777	

=== Confusion Matrix ===

a	b	<-- classified as
898	335	a = 0
134	1099	b = 1

Ek-13 : Yöntem-5'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-
weka.filters.unsupervised.attribute.Remove-V-R4,6-7,9,19,21,42,47,51,55,93,44
Instances:    2466
Attributes:   12
              D04
              M03
              M04
              H02
              H12
              H14
              L12
              C06
              C12
              C18
              U15
              C02
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|   U15 = 0: 0 (919.0/100.0)
|   U15 = 1
|   |   M03 = 2: 0 (38.0/2.0)
|   |   M03 = 1
|   |   |   M04 = 6: 0 (13.0/1.0)
|   |   |   M04 = 4: 1 (0.0)
|   |   |   M04 = 8: 1 (31.0/7.0)
|   |   |   M04 = 2
|   |   |   |   H14 <= 0: 0 (8.0/2.0)
|   |   |   |   H14 > 0: 1 (2.0)
|   |   |   M04 = 3: 0 (20.0/5.0)
|   |   |   M04 = -1: 1 (0.0)
|   |   |   M04 = 7: 1 (48.0/14.0)
|   |   |   M04 = 5
|   |   |   |   D04 = 3: 0 (3.0/1.0)
|   |   |   |   D04 = -1: 1 (0.0)
|   |   |   |   D04 = 5: 1 (20.0/6.0)
|   |   |   |   D04 = 6: 0 (4.0/1.0)
|   |   |   |   D04 = 1: 1 (0.0)
```

```
|  |  |  |  D04 = 2: 1 (0.0)
|  |  |  |  D04 = 7: 1 (0.0)
|  |  |  |  D04 = 4: 0 (3.0/1.0)
|  |  |  M04 = 1: 1 (12.0/3.0)
C06 = 1
|  H02 = 1: 1 (837.0/132.0)
|  H02 = 0
|  |  M03 = 2: 0 (29.0/7.0)
|  |  M03 = 1
|  |  |  D04 = 3: 1 (133.0/46.0)
|  |  |  D04 = -1: 1 (0.0)
|  |  |  D04 = 5: 1 (196.0/49.0)
|  |  |  D04 = 6: 1 (58.0/15.0)
|  |  |  D04 = 1
|  |  |  |  C12 = 0: 0 (8.0)
|  |  |  |  C12 = 1
|  |  |  |  |  M04 = 6: 0 (2.0)
|  |  |  |  |  M04 = 4: 0 (2.0)
|  |  |  |  |  M04 = 8: 0 (4.0/1.0)
|  |  |  |  |  M04 = 2: 0 (0.0)
|  |  |  |  |  M04 = 3
|  |  |  |  |  |  C18 = 0: 1 (0.0)
|  |  |  |  |  |  C18 = 1: 0 (3.0/1.0)
|  |  |  |  |  |  C18 = 2: 1 (2.0)
|  |  |  |  |  |  C18 = 4: 1 (0.0)
|  |  |  |  |  |  C18 = 3: 1 (0.0)
|  |  |  |  |  |  C18 = 5: 1 (0.0)
|  |  |  |  |  |  C18 = 7: 1 (0.0)
|  |  |  |  |  |  C18 = 6: 1 (0.0)
|  |  |  |  |  |  C18 = 9: 1 (0.0)
|  |  |  |  |  |  C18 = 8: 1 (0.0)
|  |  |  |  |  |  C18 = 10: 1 (0.0)
|  |  |  |  |  M04 = -1: 0 (0.0)
|  |  |  |  |  M04 = 7: 1 (4.0)
|  |  |  |  |  M04 = 5: 0 (0.0)
|  |  |  |  |  M04 = 1: 0 (0.0)
|  |  |  D04 = 2
|  |  |  |  H14 <= 0
|  |  |  |  |  C12 = 0: 0 (13.0/5.0)
|  |  |  |  |  C12 = 1: 1 (43.0/14.0)
|  |  |  |  H14 > 0: 0 (3.0)
|  |  |  D04 = 7: 0 (1.0)
|  |  |  D04 = 4: 1 (7.0/1.0)

Number of Leaves  :  50

Size of the tree :  64
```

Time taken to build model: 0.05 seconds

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

Correctly Classified Instances	2003	81.2247 %
Incorrectly Classified Instances	463	18.7753 %
Kappa statistic	0.6245	
Mean absolute error	0.2807	
Root mean squared error	0.3849	
Relative absolute error	56.1418 %	
Root relative squared error	76.9853 %	
Total Number of Instances	2466	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,753	0,128	0,855	0,753	0,800	0,629	0,837	0,812	0
	0,872	0,247	0,779	0,872	0,823	0,629	0,837	0,788	1
Weighted Avg.	0,812	0,188	0,817	0,812	0,812	0,629	0,837	0,800	

=== Confusion Matrix ===

a	b	<-- classified as
928	305	a = 0
158	1075	b = 1

Ek-14 : Yöntem-6'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-
weka.filters.unsupervised.attribute.Remove-V-R6-7,13,21,42,47,59,82,44
Instances:    2466
Attributes:   9
              M03
              M04
              H06
              H14
              L12
              C06
              C22
              U03
              C02
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|   U03 = 0: 0 (919.0/100.0)
|   U03 = 1
|   |   M04 = 6: 0 (34.0/1.0)
|   |   M04 = 4: 0 (3.0)
|   |   M04 = 8: 1 (31.0/7.0)
|   |   M04 = 2
|   |   |   H14 <= 0: 0 (9.0/2.0)
|   |   |   H14 > 0: 1 (3.0/1.0)
|   |   M04 = 3: 0 (26.0/5.0)
|   |   M04 = -1: 0 (0.0)
|   |   M04 = 7: 1 (53.0/17.0)
|   |   M04 = 5: 1 (31.0/14.0)
|   |   M04 = 1
|   |   |   H14 <= 0: 1 (9.0/1.0)
|   |   |   H14 > 0: 0 (3.0/1.0)
C06 = 1
|   M04 = 6
|   |   M03 = 2: 0 (16.0/3.0)
|   |   M03 = 1: 1 (58.0/27.0)
|   M04 = 4: 0 (12.0/4.0)
|   M04 = 8: 1 (349.0/61.0)
|   M04 = 2
```

```
|   |   H14 <= 0: 1 (127.0/23.0)
|   |   H14 > 0
|   |   |   C22 <= -3740: 1 (6.0)
|   |   |   C22 > -3740: 0 (13.0/5.0)
|   M04 = 3
|   |   C22 <= -10178: 1 (92.0/26.0)
|   |   C22 > -10178
|   |   |   H06 = 0: 0 (75.0/29.0)
|   |   |   H06 = 1
|   |   |   |   U03 = 0: 0 (6.0/2.0)
|   |   |   |   U03 = 1: 1 (4.0)
|   M04 = -1: 1 (0.0)
|   M04 = 7: 1 (371.0/73.0)
|   M04 = 5: 1 (115.0/16.0)
|   M04 = 1: 1 (101.0/3.0)

Number of Leaves  :   27
Size of the tree :    39

Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances        2008               81.4274 %
Incorrectly Classified Instances       458               18.5726 %
Kappa statistic                          0.6285
Mean absolute error                      0.2785
Root mean squared error                  0.379
Relative absolute error                 55.7008 %
Root relative squared error             75.8096 %
Total Number of Instances             2466

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0,755    0,127    0,856      0,755    0,803      0,633    0,839     0,815     0
                 0,873    0,245    0,781      0,873    0,825      0,633    0,839     0,803     1
Weighted Avg.    0,814    0,186    0,819      0,814    0,814      0,633    0,839     0,809

=== Confusion Matrix ===

   a    b   <-- classified as
 931  302 |    a = 0
 156 1077 |    b = 1
```

Ek-15 : Yöntem-7'de Seçilen Değişkenler ile Yapılan Sınıflandırma Çıktısı

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     e?it_temiz_2466-weka.filters.unsupervised.attribute.Remove-R49,54,59-
weka.filters.unsupervised.attribute.Remove-V-R1-7,11-12,15,19-
20,25,47,52,55,59,61,82,91,44
Instances:    2466
Attributes:   21
              D01
              D02
              D03
              D04
              M02
              M03
              M04
              H04
              H05
              H08
              H12
              H13
              H18
              C06
              C13
              C18
              C22
              Y02
              U03
              U13
              C02
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

C06 = 0
|   U03 = 0: 0 (919.0/100.0)
|   U03 = 1
|   |   M03 = 2: 0 (38.0/2.0)
|   |   M03 = 1
|   |   |   M04 = 6: 0 (13.0/1.0)
|   |   |   M04 = 4: 1 (0.0)
|   |   |   M04 = 8: 1 (31.0/7.0)
|   |   |   M04 = 2
|   |   |   |   D03 = 1: 0 (0.0)
```

```
|  |  |  |  D03 = 2: 0 (8.0/2.0)
|  |  |  |  D03 = 3: 1 (2.0)
|  |  |  M04 = 3
|  |  |  |  D02 = 1
|  |  |  |  |  H13 <= 1: 0 (11.0/2.0)
|  |  |  |  |  H13 > 1: 1 (3.0)
|  |  |  |  D02 = 2: 0 (6.0)
|  |  |  M04 = -1: 1 (0.0)
|  |  |  M04 = 7: 1 (48.0/14.0)
|  |  |  M04 = 5
|  |  |  |  H04 = 0
|  |  |  |  |  H05 = 0
|  |  |  |  |  |  Y02 = 0
|  |  |  |  |  |  |  H08 = 1: 1 (4.0)
|  |  |  |  |  |  |  H08 = 0
|  |  |  |  |  |  |  |  U13 = 1: 0 (5.0)
|  |  |  |  |  |  |  |  U13 = 0: 1 (2.0)
|  |  |  |  |  |  |  |  U13 = 2
|  |  |  |  |  |  |  |  |  H13 <= 0
|  |  |  |  |  |  |  |  |  |  M02 <= 5: 1 (2.0)
|  |  |  |  |  |  |  |  |  |  M02 > 5: 0 (2.0)
|  |  |  |  |  |  |  |  |  H13 > 0: 1 (3.0)
|  |  |  |  |  |  |  |  U13 = 4: 0 (2.0/1.0)
|  |  |  |  |  |  |  |  U13 = 3: 0 (1.0)
|  |  |  |  |  |  |  |  U13 = 18: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 5: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 12: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 7: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 11: 0 (0.0)
|  |  |  |  |  |  |  |  U13 = 9: 0 (0.0)
|  |  |  |  |  |  Y02 = 1: 0 (4.0)
|  |  |  |  |  H05 = 1: 1 (2.0)
|  |  |  |  H04 = 1: 1 (3.0)
|  |  |  M04 = 1: 1 (12.0/3.0)
C06 = 1
|  M03 = 2
|  |  C22 <= -22645: 1 (8.0/2.0)
|  |  C22 > -22645: 0 (29.0/5.0)
|  M03 = 1
|  |  H13 <= 0
|  |  |  C22 <= -45289: 1 (102.0/14.0)
|  |  |  C22 > -45289
|  |  |  |  D04 = 3
|  |  |  |  |  H05 = 0: 1 (90.0/34.0)
|  |  |  |  |  H05 = 1
```

```
|  |  |  |  |  |  Y02 = 0: 1 (3.0/1.0)
|  |  |  |  |  |  Y02 = 1: 0 (4.0)
|  |  |  |  D04 = -1: 1 (0.0)
|  |  |  |  D04 = 5
|  |  |  |  |  M04 = 6: 1 (0.0)
|  |  |  |  |  M04 = 4: 0 (1.0)
|  |  |  |  |  M04 = 8
|  |  |  |  |  |  Y02 = 0: 1 (31.0/4.0)
|  |  |  |  |  |  Y02 = 1
|  |  |  |  |  |  |  H18 <= 5004: 0 (3.0)
|  |  |  |  |  |  |  H18 > 5004: 1 (3.0)
|  |  |  |  |  M04 = 2
|  |  |  |  |  |  C13 = 0: 1 (18.0/2.0)
|  |  |  |  |  |  C13 = 1: 0 (2.0)
|  |  |  |  |  |  C13 = 2: 1 (2.0)
|  |  |  |  |  |  C13 = 4: 1 (0.0)
|  |  |  |  |  |  C13 = 3: 1 (0.0)
|  |  |  |  |  |  C13 = 6: 1 (0.0)
|  |  |  |  |  M04 = 3
|  |  |  |  |  |  C13 = 0
|  |  |  |  |  |  |  U03 = 0
|  |  |  |  |  |  |  |  C18 = 0: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 1: 0 (14.0/1.0)
|  |  |  |  |  |  |  |  C18 = 2: 1 (4.0/1.0)
|  |  |  |  |  |  |  |  C18 = 4: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 3
|  |  |  |  |  |  |  |  |  D01 = 2: 0 (0.0)
|  |  |  |  |  |  |  |  |  D01 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  |  D01 = 4: 1 (2.0)
|  |  |  |  |  |  |  |  |  D01 = 3: 0 (0.0)
|  |  |  |  |  |  |  |  |  D01 = 5: 0 (2.0)
|  |  |  |  |  |  |  |  |  D01 = 1: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 5: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 7: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 6: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 9: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 8: 0 (0.0)
|  |  |  |  |  |  |  |  C18 = 10: 0 (0.0)
|  |  |  |  |  |  |  U03 = 1: 1 (4.0/1.0)
|  |  |  |  |  |  C13 = 1: 1 (8.0/2.0)
|  |  |  |  |  |  C13 = 2: 0 (1.0)
|  |  |  |  |  |  C13 = 4: 0 (0.0)
|  |  |  |  |  |  C13 = 3: 0 (0.0)
|  |  |  |  |  |  C13 = 6: 0 (0.0)
|  |  |  |  |  M04 = -1: 1 (0.0)
|  |  |  |  |  M04 = 7
|  |  |  |  |  |  C18 = 0: 1 (0.0)
```

```
|   |   |   |   |   |   C18 = 1
|   |   |   |   |   |   |   D03 = 1
|   |   |   |   |   |   |   |   H12 <= 1: 0 (6.0/1.0)
|   |   |   |   |   |   |   |   H12 > 1: 1 (4.0/1.0)
|   |   |   |   |   |   |   D03 = 2: 1 (16.0)
|   |   |   |   |   |   |   D03 = 3: 1 (1.0)
|   |   |   |   |   |   C18 = 2
|   |   |   |   |   |   |   U03 = 0
|   |   |   |   |   |   |   |   M02 <= 5: 0 (4.0)
|   |   |   |   |   |   |   |   M02 > 5
|   |   |   |   |   |   |   |   |   D03 = 1: 1 (2.0)
|   |   |   |   |   |   |   |   |   D03 = 2: 0 (3.0/1.0)
|   |   |   |   |   |   |   |   |   D03 = 3: 1 (0.0)
|   |   |   |   |   |   |   U03 = 1: 1 (2.0)
|   |   |   |   |   |   C18 = 4: 1 (1.0)
|   |   |   |   |   |   C18 = 3: 1 (3.0)
|   |   |   |   |   |   C18 = 5: 1 (0.0)
|   |   |   |   |   |   C18 = 7: 1 (0.0)
|   |   |   |   |   |   C18 = 6: 1 (0.0)
|   |   |   |   |   |   C18 = 9: 1 (0.0)
|   |   |   |   |   |   C18 = 8: 1 (0.0)
|   |   |   |   |   |   C18 = 10: 1 (0.0)
|   |   |   |   |   M04 = 5
|   |   |   |   |   |   H05 = 0: 1 (19.0/1.0)
|   |   |   |   |   |   H05 = 1: 0 (3.0/1.0)
|   |   |   |   |   M04 = 1: 1 (1.0)
|   |   |   |   D04 = 6
|   |   |   |   |   D02 = 1
|   |   |   |   |   |   M04 = 6: 1 (0.0)
|   |   |   |   |   |   M04 = 4: 1 (0.0)
|   |   |   |   |   |   M04 = 8: 1 (7.0)
|   |   |   |   |   |   M04 = 2: 1 (5.0/1.0)
|   |   |   |   |   |   M04 = 3: 0 (8.0/2.0)
|   |   |   |   |   |   M04 = -1: 1 (0.0)
|   |   |   |   |   |   M04 = 7
|   |   |   |   |   |   |   D01 = 2: 1 (0.0)
|   |   |   |   |   |   |   D01 = 6: 1 (0.0)
|   |   |   |   |   |   |   D01 = 4
|   |   |   |   |   |   |   |   C22 <= -28698: 1 (2.0)
|   |   |   |   |   |   |   |   C22 > -28698: 0 (4.0/1.0)
|   |   |   |   |   |   |   D01 = 3: 1 (2.0)
|   |   |   |   |   |   |   D01 = 5: 1 (0.0)
|   |   |   |   |   |   |   D01 = 1: 1 (0.0)
|   |   |   |   |   |   M04 = 5: 0 (4.0/1.0)
|   |   |   |   |   |   M04 = 1: 1 (0.0)
|   |   |   |   |   D02 = 2: 1 (14.0/1.0)
|   |   |   |   D04 = 1
```

```
|  |  |  |  |  D03 = 1: 1 (3.0/1.0)
|  |  |  |  |  D03 = 2: 0 (18.0/2.0)
|  |  |  |  |  D03 = 3: 1 (1.0)
|  |  |  |  D04 = 2
|  |  |  |  |  H04 = 0
|  |  |  |  |  |  U13 = 1
|  |  |  |  |  |  |  D01 = 2: 1 (0.0)
|  |  |  |  |  |  |  D01 = 6: 0 (3.0)
|  |  |  |  |  |  |  D01 = 4: 1 (5.0)
|  |  |  |  |  |  |  D01 = 3: 1 (2.0)
|  |  |  |  |  |  |  D01 = 5: 0 (4.0/1.0)
|  |  |  |  |  |  |  D01 = 1: 1 (0.0)
|  |  |  |  |  |  U13 = 0
|  |  |  |  |  |  |  D01 = 2: 1 (0.0)
|  |  |  |  |  |  |  D01 = 6: 1 (4.0/1.0)
|  |  |  |  |  |  |  D01 = 4
|  |  |  |  |  |  |  |  M02 <= 6: 0 (7.0/1.0)
|  |  |  |  |  |  |  |  M02 > 6: 1 (6.0)
|  |  |  |  |  |  |  D01 = 3: 0 (2.0)
|  |  |  |  |  |  |  D01 = 5: 1 (3.0)
|  |  |  |  |  |  |  D01 = 1: 1 (0.0)
|  |  |  |  |  |  U13 = 2: 0 (3.0/1.0)
|  |  |  |  |  |  U13 = 4: 0 (1.0)
|  |  |  |  |  |  U13 = 3: 1 (1.0)
|  |  |  |  |  |  U13 = 18: 1 (0.0)
|  |  |  |  |  |  U13 = 5: 1 (0.0)
|  |  |  |  |  |  U13 = 6: 1 (0.0)
|  |  |  |  |  |  U13 = 12: 1 (0.0)
|  |  |  |  |  |  U13 = 8: 1 (0.0)
|  |  |  |  |  |  U13 = 7: 1 (0.0)
|  |  |  |  |  |  U13 = 11: 1 (0.0)
|  |  |  |  |  |  U13 = 9: 1 (0.0)
|  |  |  |  |  H04 = 1: 0 (3.0)
|  |  |  |  D04 = 7: 0 (1.0)
|  |  |  |  D04 = 4: 1 (7.0/1.0)
|  |  H13 > 0: 1 (829.0/128.0)

Number of Leaves  :  142

Size of the tree :  188


Time taken to build model: 0.09 seconds

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

Correctly Classified Instances	1972	79.9676 %
Incorrectly Classified Instances	494	20.0324 %
Kappa statistic	0.5994	
Mean absolute error	0.2783	
Root mean squared error	0.4019	
Relative absolute error	55.65 %	
Root relative squared error	80.3899 %	
Total Number of Instances	2466	

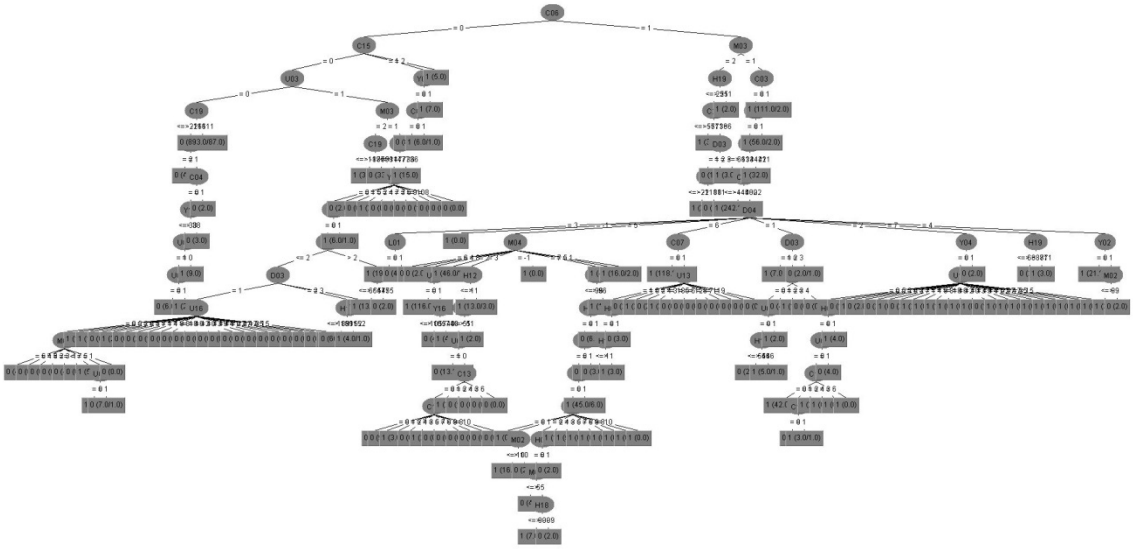
=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,748	0,148	0,834	0,748	0,789	0,603	0,817	0,790	0
	0,852	0,252	0,771	0,852	0,810	0,603	0,817	0,751	1
Weighted Avg.	0,800	0,200	0,803	0,800	0,799	0,603	0,817	0,770	

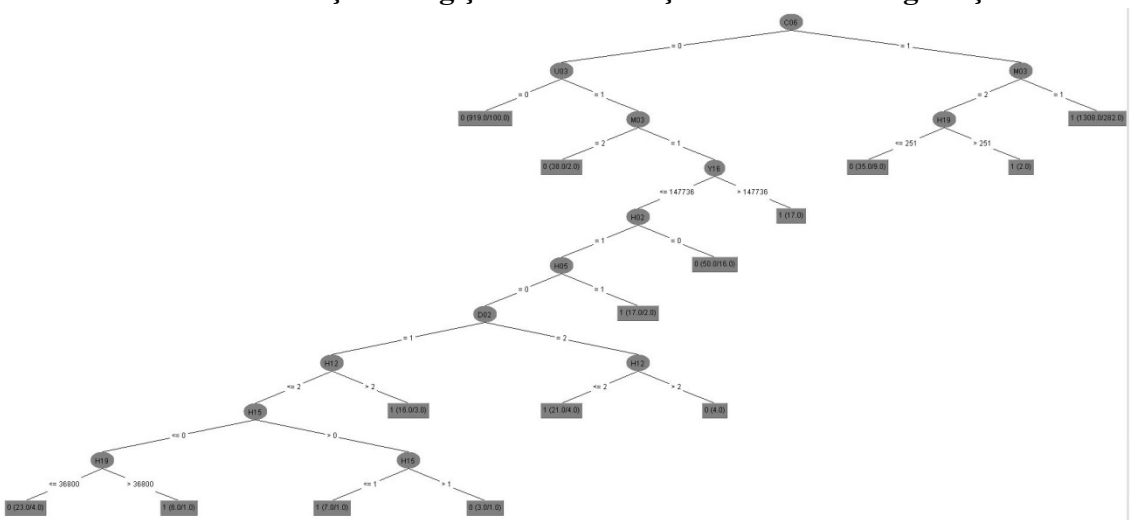
=== Confusion Matrix ===

a	b	<-- classified as
922	311	a = 0
183	1050	b = 1

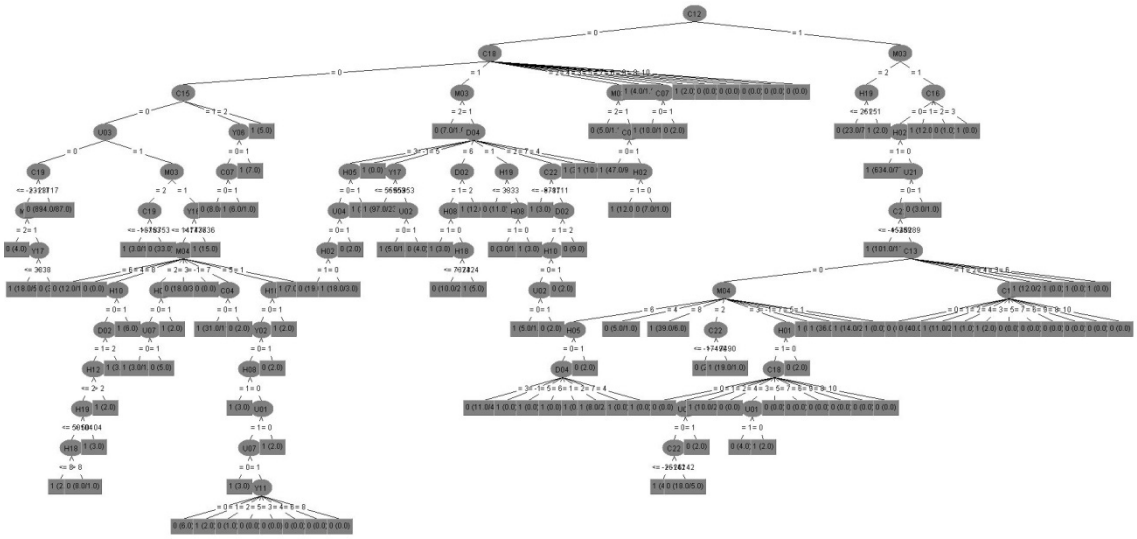
Ek-16 : Değişken Seçimi Yapılmadan Oluşturulan Karar Ağacı



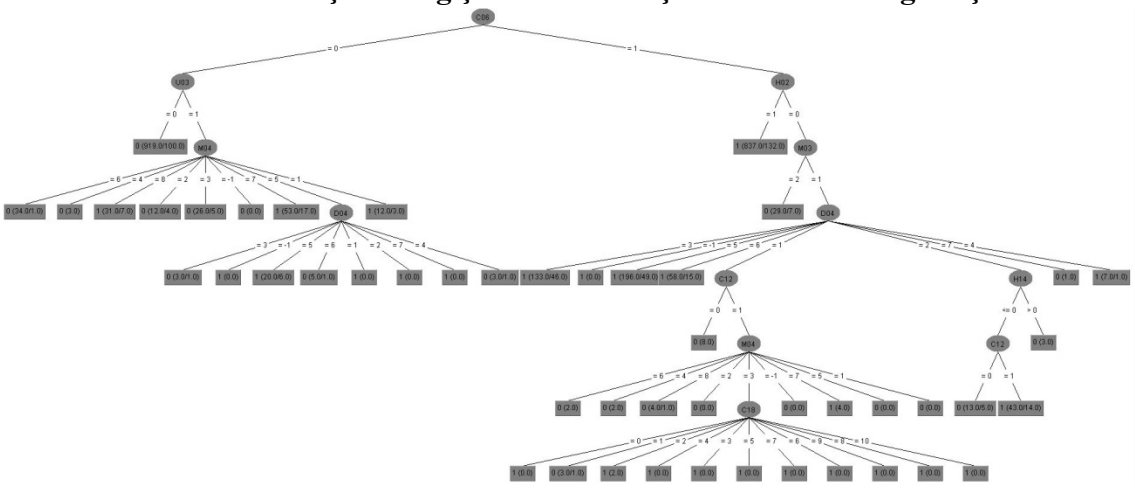
Ek-17 : Yöntem-1 'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



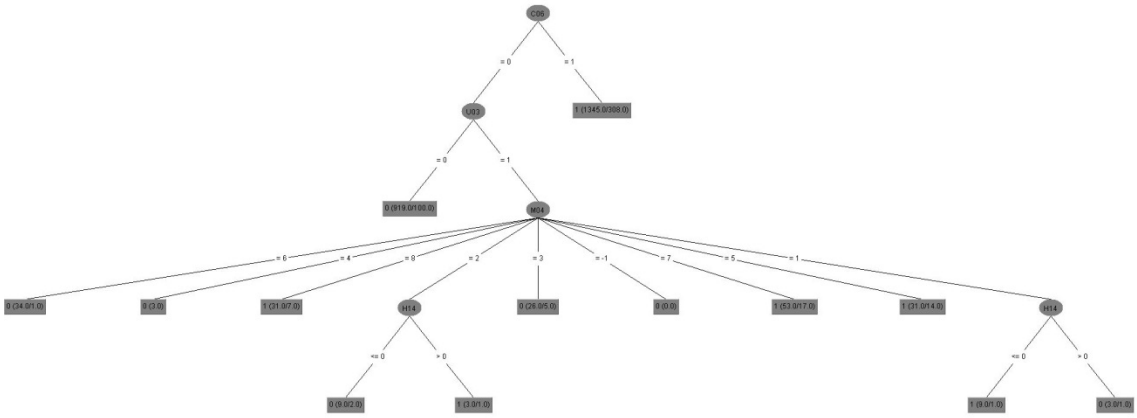
Ek-18 : Yöntem-2'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



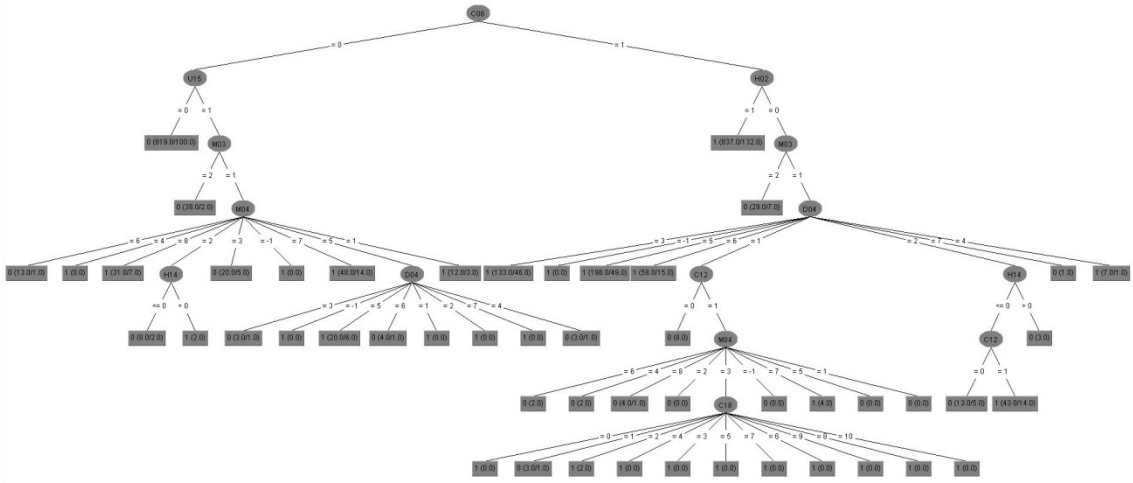
Ek-19 : Yöntem-3'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



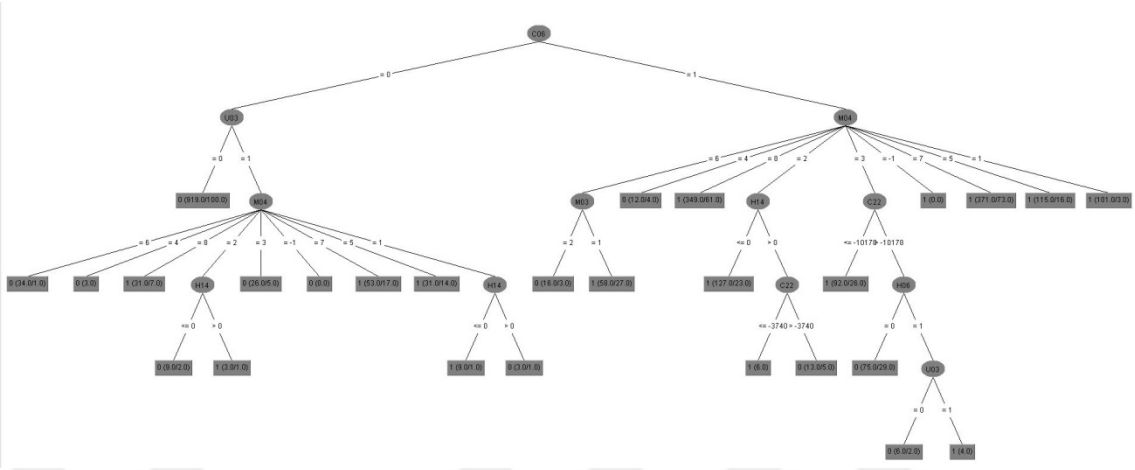
Ek-20 : Yöntem-4'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



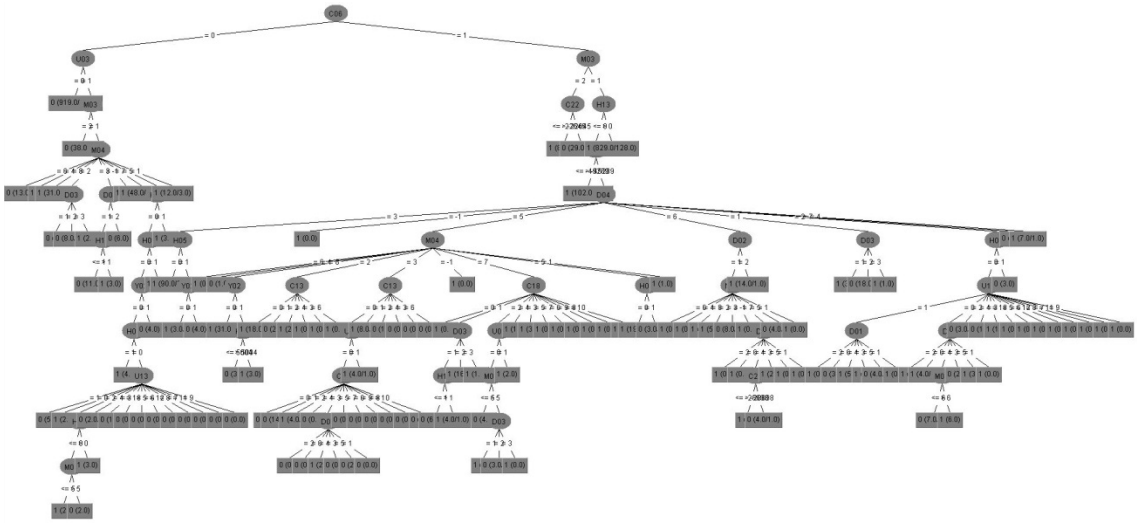
Ek-21 : Yöntem-5'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



Ek-22 : Yöntem-6'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



Ek-23 : Yöntem-7'de Seçilen Değişkenler ile Oluşturulan Karar Ağacı Çıktısı



KAYNAKLAR

Abraham, R., J. B. Simha & S. S. Iyengar (2009) Effective Discretization and Hybrid feature selection using Naïve Bayesian classifier for Medical datamining. *International Journal of Computational Intelligence Research,* 5**,** 116-129.

Alhaj, T. A., M. M. Siraj, A. Zainal, H. T. Elshoush & F. Elhaj (2016) Feature Selection Using Information Gain for Improved Structural-Based Alert Correlation. *PloS one,* 11**,** e0166017.

Ali, S. I. & W. Shahzad. 2012. A feature subset selection method based on symmetric uncertainty and ant colony optimization. In *Emerging Technologies (ICET), 2012 International Conference on*, 1-6. IEEE.

Alpaydin, E. 2014. *Introduction to machine learning*. MIT press.

Bengio, Y., I. J. Goodfellow & A. Courville (2015) Deep learning. *Nature,* 521**,** 436-444.

Blum, A. L. & P. Langley (1997) Selection of relevant features and examples in machine learning. *Artificial intelligence,* 97**,** 245-271.

Cohen, J. (1960) A coefficient of agreement for nominal scales. *Educational and psychological measurement,* 20**,** 37-46.

Çakır, Ö. 2008. Veri Madenciliğinde Sınıflandırma Yöntemlerinin Karşılaştırılması "Bankacılık Müşteri Veri Tabanı Üzerinde Bir Uygulama". In *Sosyal Bilimler Enstitüsü*. Marmara Üniversitesi.

Das, S. 2001. Filters, wrappers and a boosting-based hybrid for feature selection. In *ICML*, 74-81.

Dash, M. & H. Liu (1997) Feature selection for classification. *Intelligent data analysis,* 1**,** 131-156.

--- (2003) Consistency-based search in feature selection. *Artificial intelligence,* 151**,** 155-176.

Dash, M., H. Liu & H. Motoda. 2000. Consistency based feature selection. In *Pacific-Asia conference on knowledge discovery and data mining*, 98-109. Springer.

de Lima Oliveira, S. E., L. H. de Campos Merschmann & L. E. M. Bouillet. 2011. Identifying Significant Features in HIV Sequence to Predict Patients' Response to Therapies. In *BSB*, 18-25. Springer.

Fernando, Z. T., I. S. Thaseen & C. A. Kumar. 2014. Network attacks identification using consistency based feature selection and self organizing maps. In *Networks*

& Soft Computing (ICNSC), 2014 First International Conference on, 162-166. IEEE.

Fürnkranz, J., D. Gamberger & N. Lavrač. 2012. *Foundations of rule learning*. Springer Science & Business Media.

Gao, T.-F. & C.-L. Liu. 2007. LDA-based compound distance for handwritten Chinese character recognition. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, 904-908. IEEE.

Ge, R., M. Zhou, Y. Luo, Q. Meng, G. Mai, D. Ma, G. Wang & F. Zhou (2016) McTwo: a two-step feature selection algorithm based on maximal information coefficient. *BMC bioinformatics,* 17**,** 142.

Guyon, I. & A. Elisseeff (2003) An introduction to variable and feature selection. *Journal of machine learning research,* 3**,** 1157-1182.

Hall, M. 2000. Correlation Based Feature Selection for Discrete and Numeric Class Machine Learning. In *Proc. 17th Int'l. Conf. Machine Learning*.

Hall, M. A. 1999. Correlation-based feature selection for machine learning. The University of Waikato.

Hall, M. A. & G. Holmes (2000) Benchmarking attribute selection techniques for data mining.

--- (2003) Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data engineering,* 15**,** 1437-1447.

Hall, M. A. & L. A. Smith (1997) Feature subset selection: a correlation based filter approach.

Han, J., J. Pei & M. Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

Heaton, J. 2015. *Artificial Intelligence for Humans: Deep Learning and Neural Networks*. Heaton Research, Incorporated.

Hinton, G. E. & R. R. Salakhutdinov (2006) Reducing the dimensionality of data with neural networks. *science,* 313**,** 504-507.

Ibrahim, H. E., S. M. Badr & M. A. Shaheen (2012) Adaptive layered approach using machine learning techniques with gain ratio for intrusion detection systems. *arXiv preprint arXiv:1210.7650*.

Jalali, L., M. Nasiri & B. Minaei. 2009. A hybrid feature selection method based on fuzzy feature selection and consistency measures. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, 718-722. IEEE.

Jiang, B.-n., X.-Q. Ding, L.-T. Ma, Y. He, T. Wang & W.-W. Xie. 2008. A hybrid feature selection algorithm: Combination of symmetrical uncertainty and genetic algorithms. In *The Second International Symposium on Optimization and Systems Biology*, 152-157.

Jiang, S.-y. & L.-x. Wang (2016) Efficient feature selection based on correlation measure between continuous and discrete features. *Information Processing Letters,* 116**,** 203-215.

Jungjit, S. 2016. New Multi-Label Correlation-Based Feature Selection Methods for Multi-Label Classification and Application in Bioinformatics. University of Kent.

Kanik, E. A., S. Erdoğan & G. O. Temel (2015) İki Sonuçlu Tanı Testlerinde İki Hekim Arasındaki Uyum İstatistiklerinin Prevalanstan Etkilenme Durumları+. *Turgut Özal Tıp Merkezi Dergisi,* 19**,** 153-158.

Kannan, S. S. & N. Ramaraj (2010) A novel hybrid feature selection via Symmetrical Uncertainty ranking based local memetic search algorithm. *Knowledge-Based Systems,* 23**,** 580-585.

Kantardzic, M. 2011. *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons.

Karegowda, A. G., A. Manjunath & M. Jayaram (2010) Comparative study of attribute selection using gain ratio and correlation based feature selection. *International Journal of Information Technology and Knowledge Management,* 2**,** 271-277.

Kohavi, R. & G. H. John (1997) Wrappers for feature subset selection. *Artificial intelligence,* 97**,** 273-324.

Ladha, L. & T. Deepa (2011) Feature selection methods and algorithms. *International journal on computer science and engineering,* 3**,** 1787-1797.

Landis, J. R. & G. G. Koch (1977) The measurement of observer agreement for categorical data. *biometrics***,** 159-174.

Liu, H. & H. Motoda. 1998a. *Feature extraction, construction and selection: A data mining perspective*. Springer Science & Business Media.

Liu, H. & H. Motoda. 1998b. *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media.

Oreski, D., S. Oreski & B. Klicek (2017) Effects of dataset characteristics on the performance of feature selection techniques. *Applied Soft Computing,* 52**,** 109-119.

Pedrycz, W., G. Succi & A. Sillitti. 2016. *Computational Intelligence and Quantitative Software Engineering*. Springer.

Piroonratana, T., W. Wongseree, T. Usavanarong, A. Assawamakin, C. Limwongse & N. Chaiyaratana. 2010. Identification of Ancestry Informative Markers from Chromosome-Wide Single Nucleotide Polymorphisms Using Symmetrical Uncertainty Ranking. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2448-2451. IEEE.

Priyadarsini, R. P., M. Valarmathi & S. Sivakumari (2011) Gain ratio based feature selection method for privacy preservation. *ICTACT J. Soft Comput,* 1**,** 201-205.

Quinlan, J. (1993) C4. 5: Programs for Machine Learning Morgan Kaufmann San Mateo.

Sánchez-Maroño, N., A. Alonso-Betanzos & M. Tombilla-Sanromán (2007) Filter methods for feature selection–a comparative study. *Intelligent Data Engineering and Automated Learning-IDEAL 2007***,** 178-187.

Shahbaz, M. B., X. Wang, A. Behnad & J. Samarabandu. 2016. On efficiency enhancement of the correlation-based feature selection for intrusion detection systems. In *Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016 IEEE 7th Annual*, 1-7. IEEE.

Sun, Y., F. Wang, B. Wang, Q. Chen, N. Engerer & Z. Mi (2016) Correlation Feature Selection and Mutual Information Theory Based Quantitative Research on Meteorological Impact Factors of Module Temperature for Solar Photovoltaic Systems. *Energies,* 10**,** 7.

Uğuz, H. (2011) A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems,* 24**,** 1024-1032.

Viera, A. J. & J. M. Garrett (2005) Understanding interobserver agreement: the kappa statistic. *Fam Med,* 37**,** 360-363.

WekaForum. Mean absolute error in classification. http://weka.8497.n7.nabble.com/Mean-absolute-error-in-classification-td9440.html. [Erişim 20.02.2018].

Witten, I. H., E. Frank & M. A. Hall (2011) Data Mining: Practical Machine Learning Tools and Techniques.

Xu, B., K. Huang & C.-L. Liu. 2010. Similar handwritten Chinese characters recognition by critical region selection based on average symmetric uncertainty. In *Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on*, 527-532. IEEE.

Yang, Y. & J. O. Pedersen. 1997. A comparative study on feature selection in text categorization. In *Icml*, 412-420.

Yu, L. & H. Liu. 2003. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, 856-863.

--- (2004) Efficient feature selection via analysis of relevance and redundancy. *Journal of machine learning research,* 5**,** 1205-1224.

Zhu, Z., Y.-S. Ong & M. Dash (2007) Wrapper–filter feature selection algorithm using a memetic framework. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* 37**,** 70-76.