GEN AÇIKLAMA VERİLERİNİN SINIFLANDIRILMASINDA YENİ BİR ÖZELLİK SEÇİMİ YÖNTEMİ

A NOVEL FEATURE SELECTION METHOD FOR CLASSIFICATION OF GENE EXPRESSION DATA

DERYA TURFAN

PROF. DR. ÖZGÜR YENİAY

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

İstatistik Anabilim Dalı için Öngördüğü

DOKTORA TEZİ olarak hazırlanmıştır.

2020

ÖZET

GEN AÇIKLAMA VERİLERİNİN SINIFLANDIRILMASINDA YENİ BİR ÖZELLİK SEÇİMİ YÖNTEMİ

Derya TURFAN

Doktora, İSTATİSTİK Bölümü

Tez Danışmanı: Prof. Dr. Özgür YENİAY

Temmuz 2020, 67 sayfa

DNA mikrodizi veri kümelerinin ortaya çıkışı hem biyoinformatikte hem de makine öğrenmesi alanlarında önemli bir araştırma konusunu canlandırmıştır. Doku veya hücre örneklerinden alınan bu tip veriler, hastalık teşhisinde ya da spesifik tümör tiplerini ayırt etmede faydalı olabilecek bilgileri toplamak için oldukça önemlidir. Gen açıklama verileri olarak bilinen bu veri kümelerindeki en önemli zorluk düzinelerle ifade edilebilen örneklem sayısına karşın binlerce gen bilgisini içermesidir. Bu durum verilerin doğru sınıflandırılması açısından büyük bir dezavantaj oluşturmaktadır.

Binlerce gen ve az sayıda örneklemden oluşan gen açıklama verilerinde sınıflandırma yöntemlerinin etkili bir şekilde uygulanması, hastalıkların tanı ve tedavisinde çok önemli bir rol oynamaktadır. Büyük boyutlu verilerde, en ilişkili ve bilgilendirici özellikleri seçerek sınıflandırma performansını artırmak için bir ön işleme adımı olan özellik seçimininin (feature selection) kullanımı kaçınılmazdır. Özellik seçimi yöntemleri literatürde filtre (filter), sarmal (wrapper) ve gömülü (embedded) olmak üzere üç temel başlıkta incelenmektedir. İstatistiksel yöntemler olarak da bilinen filtre yöntemler, sınıflandırma algoritmasından bağımsız olarak özellikleri ayrı ayrı inceleyen, belli bir

değerlendirme ölçütüne dayanarak en iyi özellik alt kümesini seçmeyi amaçlayan özellik seçimi yöntemleridir.

Bu tez çalışmasında, özellik seçimi için yeni bir filtre yöntemi olarak "Etkin Aralıklara dayalı Özellik Seçimi Algoritması" (Feature Selection Algorithm based on Effective Ranges-FSAER) adlı bir yaklaşım önerilmiştir. Önerilen yöntem, literatürde bulunan "Etkin Aralığa dayalı Gen Seçimi" (Effective Range based Gene Selection-ERGS) ve "Etkin Aralığa dayalı Geliştirilmiş bir Özellik Seçimi" (Improved Feature Selection based on Effective Range-IFSER) yöntemlerinin dikkate almadığı bir eksikliği gidermek üzere geliştirilmiştir. Etkin aralıklara dayanan ERGS ve IFSER algoritmalarının en büyük eksikliği, tüm ayrık aralıklara aynı ağırlık değerini atamalarıdır. FSAER; ERGS ve IFSER yöntemlerinin avantajlarına sahiptir ve ayrıca ayrık etkili aralıkları da hesaba katarak yeni bir toplam alan tanımlamaktadır.

Önerilen algoritmanın etkinliğini doğrulamak amacıyla, erişime açık altı farklı gen açıklama veri seti kullanılarak, bilinen beş farklı filtre yöntemi ve FSAER ile farklı büyüklükteki gen alt kümeleri seçilmiştir. Seçilen genler kullanılarak farklı sınıflandırma yöntemlerinin (Naif Bayes, Destek Vektör Makinesi, k-En Yakın Komşu) uygulanması sonucunda sınıflandırma doğrulukları elde edilmiştir. Bu deneylerden elde edilen sonuçlar incelenmiş, FSAER algoritmasının diğer yöntemlerle karşılaştırıldığında sınıflandırma doğrulukları bakımından oldukça etkili sonuçlar verdiği görülmüştür.

Anahtar Kelimeler: Etkin Aralık, Filtre Yöntemler, Gen Açıklama Verileri, Özellik Seçimi, Sınıflandırma Yöntemleri.

ABSTRACT

A NOVEL FEATURE SELECTION METHOD FOR CLASSIFICATION OF GENE EXPRESSION DATA

Derya TURFAN

Doctor of Philosophy, Department of STATISTICS

Supervisor: Prof. Özgür YENİAY

July 2020, 67 pages

Emergence of DNA microarray datasets started up a crucial research subject for both bioinformatics and machine learning. This type of data is obtained from tissue or cell samples and used to collect information that may be useful for disease diagnosis or distinguishing specific types of tumors. The biggest difficulty about this type of data – which is known as gene expression data – is that it includes information of thousands of genes whereas sample sizes are limited in a few dozens. This causes a disadvantage to correct classification of data.

Effective use of classification methods on gene expression data with thousands of genes and a small amount of sample size plays a vital role in diagnosis and treatment of illnesses. In large datasets like these, it is helpful to use feature selection which is a pre-processing step to increase the classification performance by selecting most related and informative features. Feature selection methods are described in three categories in the literature as filter, wrapper, and embedded methods. Filter methods are statistical feature selection

methods that aim to select best feature subsets based on a certain evaluation measurement, independent from the classification algorithm

In this thesis, a new filter method for feature selection is suggested, namely “Feature Selection Algorithm based on Effective Ranges (FSAER)”. The suggested method aims to improve two current methods in the literature, namely “Effective Range based Gene Selection (ERGS)” and “Improved Feature Selection based on Effective Range (IFSER)”. ERGS and IFSER methods assign equal weight values to all discrete ranges. FSAER defines a new total area by taking discrete ranges into consideration in addition to having the advantages of ERGS and IFSER.

FSAER and five current filter methods are applied to six different open access gene expression datasets in order to validate the effectiveness of the suggested algorithm. Then, several classification methods (support vector machine, Naive Bayes, k-nearest neighbor) are employed to obtain the classification accuracies of the selected gene subsets. Findings of the applications are examined and FSAER is found to have highly effective results with regards to classification accuracy compared to the other methods.

Keywords: Classification Methods, Effective Range, Feature Selection, Filter Methods, Gene Expression Data.

TEŞEKKÜR

Meslek hayatına atıldığım andan itibaren yanımda olan, her daim güler yüzünü ve yardımlarını eksik etmeyen değerli hocam ve aynı zamanda danışmanım olan Prof. Dr. Özgür YENİAY'a teşekkürü borç bilirim.

Tez sürecim boyunca tanıma fırsatı bulduğum ve iyi ki tanıdığım, her zaman sorularıma en içten şekilde yanıt veren, tezimin şekillenmesi sürecinde en yoğun zamanlarında bile kapısını çalabildiğim ve zor zamanlarımda en büyük desteği veren insanlardan biri olan değerli hocam Prof. Dr. Bülent ALTUNKAYNAK'a teşekkür eder, şükranlarımı sunarım.

Tez izleme komitelerinde soru ve önerileri ile her zaman beni aydınlatan ve yol gösteren, nazik tavırlarıyla sorularıma cevap alabildiğim ve benim için bu süreci rahatlatacak her desteği vermekten kaçınmayan değerli hocam Doç. Dr. Diyar AKAY'a teşekkürlerimi sunarım.

Tez savunma sınavında jüri olarak yer alan, soru ve önerileriyle tezimin son halini almasında desteklerini sunan, içinde bulunulan zor koşullar nedeniyle uzaktan yaptığımız savunma sınavını en sorunsuz şekilde atlatmamda yardımlarını esirgemeyen çok değerli Prof. Dr. Hasan ÖRKÇÜ ve Dr. Öğr. Üyesi Burkay GENÇ hocalarıma teşekkür ederim.

Bu zorlu yola başlarken konu seçimi hakkında bana fikir veren, belki farkında olmadan kilometrelerce uzaktan yardımıma koşan, bizlere kattıklarından sonra şimdi olduğu yerde çok fazla genç insana ilham verdiğinden emin olduğum değerli hocam Doç. Dr. Haydar DEMİRHAN'a teşekkür etmeyi bir borç bilirim.

Doktora eğitimine başladığım andan itibaren hayatın bana sunduğu tüm olumsuzluklara birlikte göğüs gerdiğimiz, en zor zamanlarımda sadece yanımda olmasından bile güç aldığım, kendisinden çok şey öğrendiğim ve öğrenmeye de devam edeceğim hem meslektaş hem yol arkadaşım olan Göksu UĞURLU'ya en içten teşekkür ve sevgilerimi sunarım.

Tanıdığım günden bu yana her zaman desteğini üzerimde hissettiğim, tüm iyi ve kötü zamanlarımda yanımda olup kahrımı çeken, hem meslektaş hem sırdaşım olan Hatice ÖNCEL ÇEKİM'e en içten teşekkürlerimi sunarım.

Tez sürecimde tanışarak aynı odayı paylaştığım, her zaman hoşgörülü ve yardımsever tavırlarıyla yanımda olduğunu bildiğim sevgili meslektaşım ve oda arkadaşım olan Erhan PİŞİRİR'e çok teşekkür ederim.

Her anıyla öğretici olan doktora sürecimde, en zor zamanlarımda hep desteğini hissettiğim ve bu tezi bitirmemde manevi katkılarını göz ardı edemeyeceğim Hacettepe Üniversitesi İstatistik Bölümü'nün tüm değerli hocalarına ve çalışanlarına teşekkür etmeyi bir borç bilirim.

Yıllardır en büyük destekçilerim olan, birlikte gülüp birlikte ağladığımız, beraber büyüdüğümüz (bazen büyüyemediğimiz) eskimeyen dostlarım Merve Gülşah ULUSOY ve Pınar NİMETOĞLU'na hayatımda olduklarını ve olmaya devam edeceklerini bildiğim için teşekkür ederim.

Artık bizimle olamasa da varlığını her daim kalbimde hissettiğim; tüm hayatını ailesine adayan; benim gözümde karşılıksız sevmenin, sevilmenin, iyiliğin ve dürüstlüğün, cesaretin vücut bulmuş hali olan canım babam Mehmet TURFAN'a sonsuz teşekkürlerimi sunar ve ona olan minnettarlığımı her zaman ruhunda hissedebilmesini dilerim.

Birlikte tüm zorluklara karşı el ele verdiğimiz, hayattaki her şeyimi borçlu olduğum başta babam olmak üzere çekirdek ailemin her bir ferdine; canım annem Yüksel TURFAN, ablalarım Emine Zehra KAHYAOĞLU ve Hülya KÖKTÜRK'e her anımda yanımda olup, desteklerini asla esirgemedikleri için ve dünyanın en şanslı teyzesi olmamı sağladıkları için sonsuz teşekkürlerimi ve sevgilerimi sunarım.

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

SİMGELER VE KISALTMALAR

Simgeler

K	Kernel Fonksiyonu
μ	Ortalama
n	Toplam Örnek Sayısı
φ	Örtüşme Aralığı
ϕ	Ayrık Aralık
p	Olasılık
r^-	Etkin Aralığın Alt Sınırı
r^+	Etkin Aralığın Üst Sınırı
R	Etkin Aralık
σ	Standart Sapma
γ	Çebişev Sabiti
w	Ağırlık Katsayısı
ψ	İçerme Aralığı
χ^2	Ki-kare
$\Lambda(w)$	Hata Fonksiyonu

Kısaltmalar

AA	Ayrık Alan
AC	Alan Katsayısı
ALL	Akut Lenfoblastik Lösemi
AML	Akut Miyeloid Lösemi
CV	Çapraz Geçerlilik
DVM	Destek Vektör Makineleri
ERGS	Etkin Aralığa Dayalı Gen Seçimi
FSAER	Etkin Aralıklara Dayalı Özellik Seçimi Algoritması
IFSER	Etkin Aralığa Dayalı Geliştirilmiş Bir Özellik Seçimi
IA	İçerme Alanı
IG	Bilgi Kazancı
k-fold CV	k-Katlı Çapraz Geçerlilik
kNN	k-En Yakın Komşu
LOOCV	1-Eksiltme Çapraz Geçerlilik
NAC	Normalize Edilmiş Alan Katsayısı
NB	Naif Bayes
OA	Örtüşme Alanı
SRBCT	Küçük Yuvarlak Mavi Hücreli Tümörler
TA	Toplam Alan

1. GİRİŞ

Son yıllarda, DNA mikrodizi veri kümelerinin ortaya çıkması biyoinformatik (bioinformatics) ve makine öğrenmesinde (machine learning) yeni ve aktif bir araştırma alanının ortaya çıkmasına neden olmuştur. Biyomedikal ve DNA mikrodizi teknolojisinin hızlı gelişimi sayesinde binlerce genin açıklama seviyelerinin (gene expression levels) eş zamanlı ölçümü, çok büyük boyutlara ulaşan gen açıklama veri kümelerini (gene expression dataset) ortaya çıkarmıştır (Ang vd., 2016).

Bir gen, genetik bilginin ifadesi olan belirli bir proteini kodlayan bir DNA dizisinden oluşur. Bir deoksiribonükleik asit (DNA) molekülü, nükleotid adı verilen dört temel moleküler birimden oluşan çift sarmallı bir polimerdir. Her nükleotid, bir fosfat grubu, bir deoksiriboz şeker ve dört nitrojen bazından birini içerir. DNA'da bulunan dört farklı baz adenin (A), guanin (G), sitozin (C) ve timindir (T).

İki zincir, nitrojen bazları arasındaki hidrojen bağları ile bir arada tutulur ve baz çifti aşağıdaki kurala göre gerçekleşir: C ile G, ve T ile A eşleşir. Bir DNA molekülü dört harfli bir alfabeden oluşurken, proteinler yirmi farklı tipte amino asit dizisidir.

DNA molekülünde depolanan genetik bilginin ifadesi iki aşamada gerçekleşir: Birinci kısımda DNA molekülündeki baz dizisinin tek zincirli komplementer (tamamlayıcı) bir kopyası olan haberci ribonükleik asite (messenger (m)RNA'ya) transkripsiyonun yapıldığı "transkripsiyon" aşaması sırasında timin bazının yerini urasil (U) alır; (ii) İkinci kısım ise mRNA molekulunu baz alarak protein üretilmek için çevrildiği çeviri aşamasıdır. Böylece her bir gendeki DNA dizisindeki dört harfli alfabe önce mRNA'ya, oradan da proteine çevrilir. Her bir gen dizisindeki her uç baz, proteinin yapıtaşı olan bir aminoasite çevrilir. Sonuç olarak 4 DNA bazı ile yirmi harfli aminoasit alfabesi arasındaki örtüşme, nükleotid üçlülerini amino asitlerle ilişkilendiren genetik kodla belirlenir.

Bir hücrede her genden ne kadar çoğunlukta olduğunu saptamak için, elde edilen komplementer cDNA mikrodizileri, bir cam mikroskop lamı üzerinde yüksek yoğunluklu bir dizide basılmış binlerce ayrı DNA sekansının yer etmesiyle ya da yapıştırılması ile oluşur. Bu DNA sekanslarının iki farklı lam üzerinde iki farklı örnekten elde edilen DNA veya cDNA örneğindeki görece bolluğu, iki örneğin dizideki sekanslara farklı hibridizasyonunun izlenmesiyle değerlendirilebilir.

Bu amaçla, iki DNA numunesi veya hedefi, farklı floresan boyalar (örn. Kırmızı floresan boya Cy5 ve yeşil floresan boya Cy3) kullanılarak etiketlenir, ardından dizilmiş DNA sekansları veya probları ile karıştırılır ve hibridize edilir.

Bu rekabetçi hibridizasyondan sonra, dizi üzerindeki her noktada her boya için ayrı ayrı floresan ölçümleri yapılır. Örnekten elde edilen DNA ne kadar lam üzerindeki cDNA'ya bağlanırsa, o kadar parlak Cy3 ya da Cy5 floresan çıkarır. Her nokta bakımından floresan yoğunluğunun oranı, iki örnekte karşılık gelen DNA sekansının göreceli bolluğunun göstergesidir.

Miktarları ile orantılı olarak etiketlenmiş moleküller bir ışık yardımıyla aydınlatılır ve böylece floresan yaymaları sağlanır. Bunun nedeni, mikrodizide hibridize olmuş örnekteki gen miktarını belirlemektir. Bir tarayıcı, floresanları yakalar. Tarayıcı böylece her biri bir proba (cDNA) karşılık gelen parıldayan noktalardan oluşan bir gridi içeren görüntüyü verir. Nihayetinde, görüntü sayısal hale getirilebilir ve böylece analiz edilebilecek veri ortaya çıkmış olur (Dash & Patra, 2011).

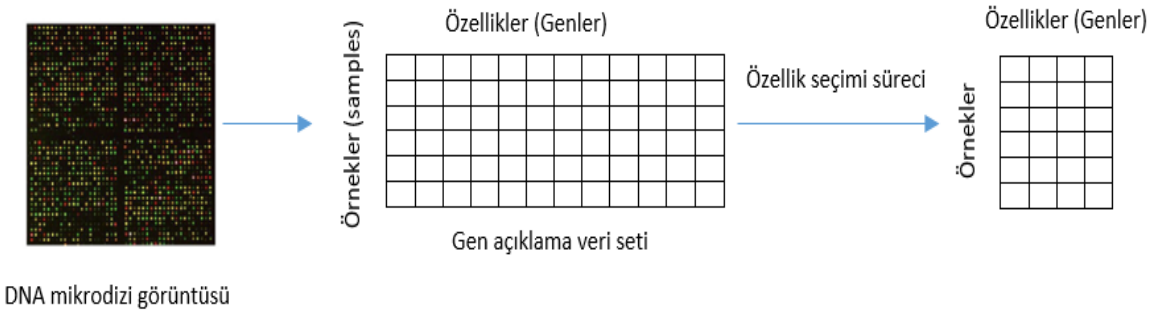
Bu tarz veriler genellikle az sayıda örnekleme karşın birçok özelliğe (gene) sahip olmasıyla karakterize edilir. Az sayıda örneklem için bu kadar çok sayıda özelliğin ele alınması, tahmin modelini oluştururken veya ilgili genleri seçerken rastgele ortaya çıkabilecek "yanlış pozitifler (false-positives)" olasılığı nedeniyle araştırmacılar için bir zorluk yaratmaktadır (Piatetsky-Shapiro & Tamayo, 2003).

Mikrodizi deneyleri sonucu ortaya çıkan gen açıklama verileri sayısal değerlerle ifade edilmekte, binlerce özellikle temsil edilen az sayıda örneklem ve iki veya daha fazla sınıf sayısından oluşmaktadır. Bu veri kümelerinin doğru olarak sınıflandırılması; ilaç geliştirmede (Liang vd., 2018), kanser hastalıklarının tanısı, tedavisi ve önlenmesinde (Chen, 2003; Das vd., 2018; Golub vd., 1999; Nguyen & Nahavandi, 2016; S. Zhang vd., 2018) oldukça önemli rol oynamaktadır. Bu veriler sınıflandırılırken, az örneklem ve çok sayıda özelliğin bulunması performans açısından bir dezavantaj oluşturmaktadır (Buza, 2016; Scott, 2015).

Bu durumda, gereksiz ve alakasız özellikleri ortadan kaldırmak ve uzmanların gen açıklama verileri ile belirli bir hastalık arasındaki temel ilişkileri tespit etmesine yardımcı olmak için özellik seçimi (feature selection) yapılması kaçınılmazdır. Özellik seçmenin genler üzerinde uygulanması gen seçimi olarak da bilinmekte, her iki yöntem de orijinal veriden bir alt küme seçmeyi amaçlamaktadır.

Golub *vd.* (1999), bir DNA mikrodizisinde bulunan az sayıda genin bir sınıflandırma problemini çözmek için yeterli olduğunu göstermektedir. Yapılan biyolojik araştırmalara dayanarak, biyolojik süreçte ve hastalıkların gösterilmesinde az sayıda genin kritik rol oynadığı doğrulanmıştır (Guo vd., 2017; Liao vd., 2003), kalan genler ise genellikle gereksiz veya gürültülüdür.

Gen açıklama verilerinde bulunan özelliklerin içindeki gürültülü kısımlar sınıflandırma başarısını olumsuz yönde etkilemektedir. Buna ilaveten, yüksek boyutlu gen açıklama verilerinin işlenmesi sınıflandırma algoritmalarının performansını düşürmekle kalmamakta, aynı zamanda hesaplama zamanı yükünü de arttırmaktadır. Bu nedenle, gen açıklama verilerinin boyutunun, daha iyi sınıflandırma sonuçları veren bir gen alt kümesi seçilerek azaltılması gerekmektedir (Algamal vd., 2018; Song vd., 2016; C. Tang vd., 2018; Yang vd., 2012; Zheng vd., 2019). Özellik seçiminin gen açıklama verilerine uygulanması süreci genel hatlarıyla Şekil 1.1'de gösterilmiştir.



Şekil 1.1. Özellik seçiminin gen açıklama verilerine uygulanması süreci

Literatürdeki çalışmalar incelendiğinde gen verilerinde sınıflandırma öncesi özellik seçimi yapmanın oldukça etkili ve çarpıcı sonuçları olduğu görülmektedir (Abusamra, 2013; Ahmad vd., 2019; Al-Rajab vd., 2017; Bonilla-Huerta vd., 2016; Golub vd., 1999; Kang vd., 2019; Kira & Rendell, 1992; Z. Li vd., 2018; Huiqing Liu vd., 2002; Zeebaree vd., 2018; Zhou & Mao, 2005). Örneğin; 7129 gen ile çalışılan bir mikrodizi veri analizinde sınıflandırma performansını güçlendirmek için özellik seçimi sonucunda yalnızca iki gen ile çalışmanın yeterli olduğu görülen çalışmalar mevcuttur (Bolón-Canedo vd., 2014).

Özellik seçim yöntemleri filtre (filter), sarmal (wrapper) ve gömülü (embedded) yöntemler olmak üzere üç başlık altında toplanmaktadır (Ahmad vd., 2019; Blum & Langley, 1997). Filtre yöntemler, sınıflandırma algoritmasından bağımsız olarak özellikleri ayrı ayrı inceleyen, belli bir istatistiksel değerlendirme ölçütüne göre en iyi özellik alt kümesini seçmeyi amaçlayan özellik seçimi yöntemleridir (Sánchez-Maroño vd., 2007). Sarmal yöntemler, özellik alt kümelerinin değerlendirilmesinde belli bir sınıflandırma algoritmasının performansına bağlı olan, özellikler arası etkileşimi dikkate alan yöntemlerdir (Xiong vd., 2001). Gömülü yöntemler de sarmal yöntemler gibi sınıflandırma algoritmasına bağlıdır. Sarmal yöntemlerde özellik seçme işlemi için her adımda bir sınıflandırma algoritmasına ihtiyaç duyulurken, gömülü yöntemlerde ise özellik alt küme seçimi sınıflandırıcının eğitim sürecinin bir parçasıdır. Yani gömülü yöntemler özellik seçimi ile sınıflandırma performanslarını eş zamanlı olarak

gerçekleştirirler. Bu bakımdan sarmal yöntemlere göre daha düşük hesaplama maliyeti olduğu söylenebilir (Remeseiro & Bolon-Canedo, 2019).

Filtre yöntemler öğrenme algoritmasından bağımsız oldukları için diğer yaklaşımlara göre hesaplama maliyetleri daha düşüktür. Bu durum, mikrodizi verileriyle uğraşırken özellikle önemlidir, çünkü az sayıda örnek, sarmal yöntemlerin kullanımını mantıksız kılacak aşırı öğrenmeye (over-fitting) neden olabilmektedir. Bu nedenle mikrodizi verileri için filtre yöntemlerinin kullanımı oldukça yaygındır (Radovic vd., 2017; Remeseiro & Bolon-Canedo, 2019).

İstatistiksel tabanlı yöntemler olarak da bilinen bu değerlendirme ölçütlerinden bazıları Z-skoru (Thomas vd., 2001), t-testi (Kuo vd., 2007; Long vd., 2001), ki-kare (chi-square) (Jin vd., 2006), bilgi kazancı (information gain) (Yang vd., 2012), ortak bilgi (mutual information) (Cai vd., 2009) ve Relief-F (Robnik-Šikonja & Kononenko, 2003) yöntemleridir.

Mevcut bilinen istatistiksel yöntemlerden farklı olarak Chandra ve Gupta (2011), "Etkin Aralığa dayalı Gen Seçimi" (Effective Range based Gene Selection, ERGS) adı verilen, her sınıf için özelliklere etkin bir aralık tanımlayan istatistiksel bir yöntem önermişlerdir. Bu yeni ve etkin bir özellik seçimi yaklaşımı olan ERGS algoritmasının altında yatan temel prensip, sınıfları açıkça ayıran özelliklere daha çok ağırlık verilmesidir. Wang *vd.* (2014), 2011 yılında Chandra ve Gupta tarafından önerilen ERGS adlı algoritmanın eksik buldukları bir yanını geliştirerek "Etkin Aralığa Dayalı Geliştirilmiş Bir Özellik Seçimi" (Improved Feature Selection based on Effective Range, IFSER) yöntemini önermişlerdir. Her iki çalışmada da bilinen farklı veri kümeleriyle çalışılarak, mevcut filtre yöntemlerle karşılaştırmalar yapılmış ve doğru sınıflandırmada önemli iyileştirmeler olduğu görülmüştür. Çebişev eşitsizliğine dayanan etkin aralıkların kullanıldığı bu iki yöntemin

de dikkate almadığı durum ise tamamen ayrık olan sınıflarda, sınıfların birbirine olan uzaklığı dikkate alınmadan hepsine aynı değerin atanmasıdır.

Bu tez çalışmasında, ayrık etkin aralıkları da ayırt eden yeni bir özellik seçim algoritması "Etkin Aralıklara dayalı Özellik Seçimi Algoritması" (Feature Selection Algorithm based on Effective Ranges, FSAER) önerilmektedir.

Tez çalışmasının izleyen ikinci bölümünde, özellik seçimi hakkında genel bilgiler verilerek özellik seçim yöntemleri tanıtılmıştır. Bu çalışmanın odak noktası olan filtre yöntemlerden literatürde bilinen beş farklı filtre yöntemi algoritmalarıyla birlikte detaylı olarak incelenmiştir.

Üçüncü bölümde, çalışmada önerilen özellik seçimi algoritması "FSAER" tanıtılarak bu yöntemi geliştirmede etkili olan motivasyon açıklanmıştır. Örneklerle gerekliliği anlatılan bu yöntemin algoritması aktarılmıştır.

Dördüncü bölümde, uygulanan özellik seçim yöntemlerinin değerlendirilmesini sağlayacak olan sınıflandırma yöntemleri ile geçerlilik ölçütleri aktarılmıştır. Bu çalışmada kullanılan üç farklı sınıflandırma yöntemi ve en çok bilinen bazı geçerlilik ölçütleri incelenmiştir.

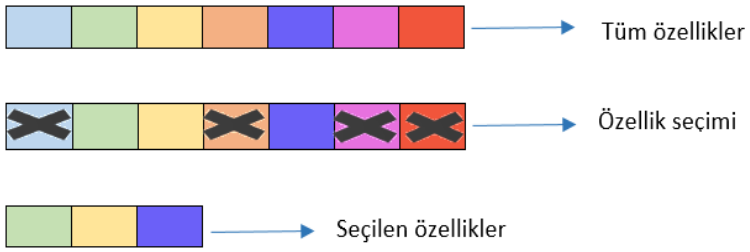
Beşinci bölümde öncelikle çalışmada kullanılan gerçek veri kümeleri tanıtılmıştır. Daha sonra yapılan uygulamalardan elde edilen sonuçlar çizelgeler ve grafikler halinde aktarılmıştır. Farklı sınıflandırma yöntemlerinden elde edilen sonuçlar karşılaştırılmıştır.

Son olarak altıncı bölümde genel değerlendirme yapılarak tezin katkısına değinilmiştir.

2. ÖZELLİK SEÇİM YÖNTEMLERİ

Son yıllarda makine öğrenmesi ve veri madenciliği alanlarındaki verilerin boyutu önemli ölçüde artış göstermiştir. Büyük boyutlara sahip veriler mevcut öğrenme algoritmalarının performansında ciddi sorunlara yol açabilmektedir (Huan Liu & Motoda, 2007; J. Tang vd., 2014).

Özellik seçimi, başlangıç özelliklerinin sayısının azaltıldığı ve daha iyi performans elde etmek için yeterli bilgiyi tutan özelliklerin bir alt kümesinin seçildiği süreçtir. Şekil 2.1'de özellik seçimi ile yeni bir özellik alt kümesi seçmeye örnek gösterilmiştir.



Şekil 2.1. Özellik seçimi ile yeni bir özellik alt kümesi seçmeye örnek

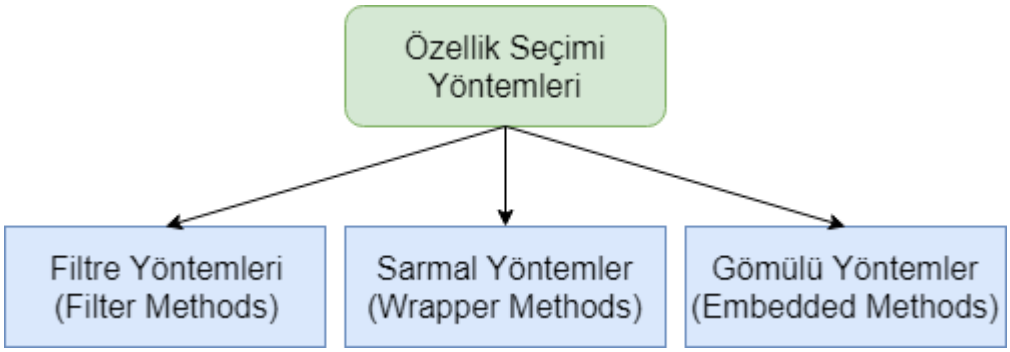
Çok boyutlu veri kümelerinin giderek arttığı ve ilgi kazandığı günümüzde kümeleme, regresyon ve sınıflandırma gibi birçok alanda özellik seçimine olan ilgi artmaktadır. Bu tez çalışmasında, sınıflandırma yöntemlerinde özellik seçiminin uygulanması çalışılmıştır.

Özellik seçimi konusundaki çalışmalar 1960'lara dayanmaktadır (Abend & Harley, 1969). Günümüzde büyük veri kümelerinin artmasıyla birlikte birçok farklı alanda özellik seçimine olan ilgi artmaktadır (Huan Liu, 2018).

Mikrodizi teknolojisi, başta kanser hastalıkları olmak üzere tıp biliminin birçok alanında araştırmacılara fayda sağlayacak olan binlerce gen açıklama verisine aynı anda ulaşabilme imkanı sağlamıştır (S. Liu vd., 2018). Bu teknoloji ile ortaya çıkan ve çok büyük boyutlardan oluşan gen açıklama verilerinde araştırmacılar, binlerce gen ile temsil

edilen az sayıda örneklem sorunuyla karşı karşıya kalmışlardır. Burada ilgilenilen gen sayısının çok fazla olması nedeniyle sınıflandırmada bazı dezavantajlar söz konusu olabilir. Özellik seçimi ile bu dezavantajlar yok edilerek pek çok avantaj sağlanması söz konusu olacaktır.

Özellik seçimi, bilgilendirici olmayan ve sınıflandırma performansını iyileştirmeyen gereksiz ve alakasız özellikleri kaldırmak için kullanılır. Uygun özelliklerin seçilmesi ile model parametrelerini azaltarak düşük karmaşıklıkta bir model sağlamak, aşırı öğrenmeyi önlemek ve genelleme performansını iyileştirmek, özellik sayısını azaltarak model eğitimi için gereken süreyi azaltmak, veri toplama ve depolama maliyetini azaltmak mümkün olabilmektedir (Ahmad vd., 2019; Jirapech-Umpai & Aitken, 2005; Remeseiro & Bolon-Canedo, 2019). Bu faydalar dikkate alındığında araştırmacılar, binlerce gen içerisinden hastalıkları belirlemede önemli rol oynayan genlerin belirlendiği yani bir başka deyişle özellik seçimi yöntemlerinin uygulandığı ve sonrasında seçilen bu genler ile sınıflandırmanın yapıldığı çalışmalara yönelmişlerdir. Sınıflandırma açısından, özellik seçme yöntemleri Şekil 2.2'de görüldüğü üzere filtre, sarmal ve gömülü yöntemler olarak üç ana başlıkta incelenmektedir (Ahmad vd., 2019).



Şekil 2.2. Özellik seçimi yöntemleri

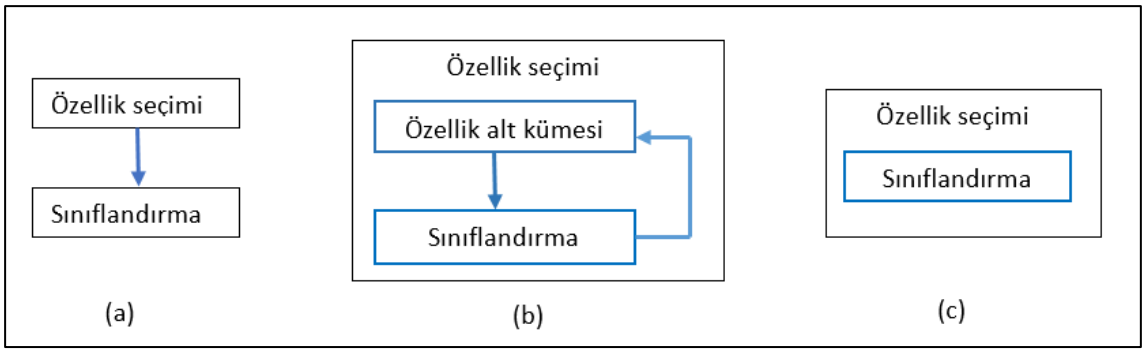
Sınıflandırma öncesi bir ön işlem olarak kullanılan filtre yöntemlerde, sınıflandırıcıdan bağımsız olarak belli bir değerlendirme ölçütü ile genler seçilir. Literatürde sıkça kullanılan bu değerlendirme ölçütlerinden bazıları Z-skoru (Thomas vd., 2001), t-testi (Kuo vd., 2007; Long vd., 2001), ki-kare (Jin vd., 2006), bilgi kazancı (Yang vd., 2012), ortak bilgi (Cai vd., 2009) ve Relief-F (Robnik-Šikonja & Kononenko, 2003)

yöntemleridir. İlk olarak Golub *vd.* (1999) genlerin artı ve eksilerini değerlendirmek için sinyal-gürültü oranı (signal-to-noise ratio) fonksiyonunu önermişlerdir. Daha önce özellik seçimi için literatürde kullanılan Relief-F (Robnik-Šikonja & Kononenko, 2003) ve MRMR (Minimum Redundancy Maximum Relevance) (Peng vd., 2005) yöntemleri gen seçimi için birlikte kullanılarak yeni bir yöntem önerilmiştir (Y. Zhang vd., 2008). Sun, Lu ve Li (2018) mikrodizi verilerinin sınıflandırılmasında çapraz entropi (cross-entropy) temelli çoklu bir filtre yöntemi önermişlerdir.

Sarmal yöntemler en iyi gen alt kümesini seçerken genellikle sınıflandırma doğruluğunu bir gösterge olarak kullanırlar. Genler arasındaki etkileşimi de inceleyen sarmal yöntemler bu yönüyle genleri ayrı ayrı inceleyen filtre yöntemlerden ayrılırlar. Filtre yöntemlerinin aksine sınıflandırma algoritması ile de etkileşim halindedirler. Binlerce gen verisi ile çalışılması nedeniyle sarmal yöntemler daha yüksek hesaplama maliyetine sahip oldukları için filtre yöntemlere göre daha az tercih edilirler (Bolón-Canedo vd., 2014). Literatürde sıklıkla kullanılan sarmal yöntemlerden bazıları destek vektör makineleri - DVM (support vector machine - SVM) (Duan vd., 2005; Zhou & Tuck, 2007), genetik algoritmalar (genetic algorithms) (C. P. Lee vd., 2011), tavlama benzetimi (simulated annealing) (Gheyas & Smith, 2010), ardışık ileriye doğru seçim ve ardışık geriye doğru elemedir (sequential forward and backward selection) (Inza vd., 2002).

Gömülü yöntemler en iyi gen alt kümesini seçerken gen seçimi ve sınıflandırma işlemini eş zamanlı olarak gerçekleştiren yöntemlerdir. Literatürde kanser verilerini sınıflandırmak için gen seçiminde en bilinen gömülü yöntem çalışmalarından biri Guyon vd. (2002) tarafından önerilen SVM-RFE (Support Vector Machine based on Recursive Feature Elimination) yöntemidir. Bunun dışında gen seçimi için kullanılan birçok gömülü yöntem çalışması mevcuttur (Canul-Reich vd., 2012; Kang vd., 2019; Maldonado vd., 2011).

Şekil 2.3'te filtre, sarmal ve gömülü yöntemlerin yukarıda bahsedilen çalışma prensipleri genel olarak özetlenmiştir.



Şekil 2.3. (a) Filtre, (b) sarmal ve (c) gömülü yöntemlerin çalışma prensipleri

Bu tez çalışmasında önerilen yöntem filtrelemeye dayalı bir gen seçim yöntemi olduğu için takip eden alt bölümde seçilen filtre yöntemleri ile ilgili daha detaylı bilgi verilmiştir.

2.1. Filtre Yöntemler

Bu bölümde, özellik seçimi literatüründe sıklıkla kullanılan filtre yöntemlerinden olan ki-kare (χ^2) istatistiği, Relief-F ve bilgi kazancı yöntemleri açıklanmıştır. Ayrıca tez çalışmasının temelini oluşturacak olan ERGS ve IFSER filtre yöntemleri de algoritmaları ile birlikte ayrıntılı olarak incelenmiştir.

2.1.1. Ki-kare (χ^2) İstatistiği

Ki-kare filtre yöntemi ki-kare test istatistiğine dayanmaktadır. Ki-kare test istatistiğinin değeri sınıflara göre her bir özellik için ayrı ayrı hesaplanır. Test istatistiği hesaplanmadan önce sayısal değerlere sahip her bir özellik kesiklileştirilmelidir. Her X_i özelliği için ki-kare test istatistiği Eşitlik 2.1 ile tanımlanır (Chandra & Gupta, 2011):

$$\chi^2 = \sum_{x \in X_i} \sum_{c \in C} \frac{(n_{(x \in X_i \& c \in C)} - e_{(x \in X_i \& c \in C)})^2}{e_{(x \in X_i \& c \in C)}} \tag{2.1}$$

Burada $n_{(x \in X_i \& c \in C)}$ ifadesi değeri x olan X_i'deki (c sınıfı için) örnek sayısını gösterir. Beklenen sıklık $e_{(x \in X_i \& c \in C)}$ Eşitlik 2.2 ile tanımlanır:

$$e_{(x \in X_i \& c \in C)} = \frac{n_{x \in X_i} * n_{c \in C}}{n} \quad (2.2)$$

Burada $n_{x \in X_i}$; x değerli X_i'deki örnek sayısını gösterir ve $n_{c \in C}$; c sınıfındaki örnek sayısını temsil eder. n ise toplam örnek sayısıdır.

İstatistiksel analizlerde, kategorik değişkenlerin birlikteliğinin bir ölçütü olarak Cramer's Phi katsayısı kullanılmaktadır. Bu katsayıyı, özellik seçimi yaparken bir filtre olarak kullanmak mümkündür. Söz konusu ölçüt Eşitlik 2.3 ile ifade edilmektedir:

$$\phi_C = \sqrt{\frac{\chi^2}{n(k-1)}} \quad (2.3)$$

Görüldüğü üzere bu ölçüt ki-kare test istatistiği tanımına bağlı olarak elde edilir. Burada k, satır ya da sütun sayılarından hangisi az ise onu ifade etmektedir.

Özellikler, her bir özellik için ki-kare test istatistiği yardımıyla hesaplanan Cramer's Phi katsayısının sıralanmış değerlerine dayanılarak seçilir. Bu katsayının büyük değer aldığı özelliklere daha yüksek ağırlık verilerek seçim yapılmaktadır.

2.1.2. Relief-F

Relief algoritması özellikleri aralarındaki ilişkiye göre ağırlıklandıran, iki sınıflı veri kümeleri için önerilmiş bir özellik seçme algoritmasıdır (Kira & Rendell, 1992). Relief-F algoritması gürültülü, tamamlanmamış ve çok sınıflı veri kümeleri ile başa çıkabilmek için Relief algoritmasının bir uzantısı olarak geliştirilmiştir (Chandra & Gupta, 2011). Relief-F algoritmasının temel mantığı, özelliklerin değerlerinin farklı sınıfların örnekleri arasında ne kadar iyi ayrıştırma yapabildiğine göre ve aynı sınıfın örneklerini ne kadar iyi kümeleyebildiklerine göre özellikleri oranlamasıdır (Y. Wang vd., 2005).

Bu yöntemde her özelliğe bir "ilişki" ağırlığı atanır. Rastgele olarak, n tane örneklemden bir r örneklemi seçilir. Seçilen örnekler (r) ile aynı (H) (en yakın hedef) ve farklı (M(c))

(c sınıfının en yakın ıskası) sınıfların en yakın örnekleri arasındaki farka bağlı olarak ilişki değerleri güncellenir. Farklı sınıfların komşularından örneği ayrıştıran özelliklere daha fazla ağırlık verilir. Ağırlıklar en yakın ıskaların M(c) ortalama katkısı göz önünde bulundurularak güncellenir. Ortalama katkı aynı zamanda her bir sınıfın ilk olasılığını (P(C)) dikkate alır. i'nci özelliğinin ağırlığı Eşitlik 2.4 ile güncellenir (Chandra & Gupta, 2011):

$$w_i = w_i - \frac{\eta(X_i, r, H)}{n} + \sum_{C \neq C_r} \frac{P(C) * \eta(X_i, r, M(c))}{n} \tag{2.4}$$

Burada, $\eta(X_i, r, H)$ fonksiyonu seçilen örnekler (r) ile en yakın hedef (H) veya en yakın ıskalar (M(c)) arasındaki uzaklığın hesabıdır.

2.1.3. Bilgi Kazancı (IG)

Bilgi kazancı, karar ağaçlarında bir özellik seçme kriteri olarak sıklıkla kullanılmaktadır (Quinlan, 1986). Liu, Li ve Wong (2002) bu yöntemi gen seçme kriteri olarak da kullanmışlardır.

Bilgi kazancı entropi kavramına dayanmaktadır. Bir değişkendeki belirsizliği ölçmek için kullanılan entropi değeri 0 ise belirsizlik yoktur denir. Yani bu durumda değişkenin bütün değerleri birbirine eşittir denilebilir. Ancak değişken tamamen birbirinden farklı değerler alıyorsa entropi alabileceği maksimum değere ulaşır (Han vd., 2012).

Örneğin n tane örnek $N = \{1, 2, ..., n\}$, d tane özellik $X = \{X_1, X_2, ..., X_d\}$, $i = 1, 2, ..., d$ ve l tane sınıftan oluşan bir veri kümesi (D^{n*d}) olsun.

D veri kümesinin –sınıf etiketleri için- entropisi Eşitlik 2.5 ile hesaplanabilir (Lai vd., 2016).

$$Entropi(D) = -\sum_{j=1}^{l} \frac{n(j)}{n} * (log_2 \frac{n(j)}{n}) \tag{2.5}$$

Burada $n(\mathrm{j})$; j'inci sınıfa ait örnek sayısını ($j = 1,2,...,l$), n ise toplam örnek sayısını göstermektedir.

Veri matrisinin gözelerinde bulunan gen açıklama verileri sürekli değerlerle ifade edildiğinden, IG yöntemini bu veri kümelerine uygulayabilmek için ki-kare yönteminde de olduğu gibi ilk olarak kesiklileştirme yapılmalıdır. Bu kesiklileştirme işlemi sonrasında eğer bir X_i geni $V = \{v_1, v_2, ..., v_m\}$ düzeylerine sahipse, X_i geni için entropi hesabı şu şekilde olmaktadır:

Adım 1: Eşitlik 2.5 kullanılarak veri kümesinin sınıf etiketlerine ait entropi değeri hesaplanır.

Adım 2: Her bir özellik için X_i, i=1,2,…,d entropi değeri $Entropi(\mathrm{X}_i)$ hesaplanmalıdır. Ancak bu yapılırken ilgili özelliğin her bir sınıfı ayrı ayrı incelenmelidir. Burada $\mathrm{C}_j, j = 1,2,...,l$ j'inci sınıfa ait örneklerin kümesidir. Özellikler için entropi değerleri her bir sınıf kümesi (C_j) ayrı bir veri kümesi gibi düşünülerek, tüm sınıflar için ayrı ayrı $V = \{v_1, v_2, ..., v_m\}$ düzeyleri üzerinden Eşitlik 2.6 ile hesaplanabilir (j=1, 2, …, l).

$$\begin{aligned}
Entropi((\mathrm{X}_i)_1) &= -\sum_{k=1}^{m} \frac{n_1(\mathrm{k})}{n_1} * (log_2 \frac{n_1(\mathrm{k})}{n_1}) \\
Entropi((\mathrm{X}_i)_2) &= -\sum_{k=1}^{m} \frac{n_2(\mathrm{k})}{n_2} * (log_2 \frac{n_2(\mathrm{k})}{n_2}) \\
&\vdots \\
Entropi((\mathrm{X}_i)_l) &= -\sum_{k=1}^{m} \frac{n_l(\mathrm{k})}{n_l} * (log_2 \frac{n_l(\mathrm{k})}{n_l})
\end{aligned} \tag{2.6}$$

Burada $(\mathrm{X}_i)_l$; i'inci özelliğin *l'*inci sınıfı için entropi değerini göstermektedir. $n_l(\mathrm{k})$; i'inci özelliğin *l'*inci sınıfının içinde k düzeyine sahip örnek sayısını, n_l ; i'inci özelliğin *l'*inci sınıfının örnek sayısını temsil etmektedir.

Adım 3: $Entropi(X_i)$ değeri, Eşitlik 2.6'da farklı sınıflar için bulunan değerler kullanılarak Eşitlik 2.7 yardımıyla bulunur.

$$Entropi(X_i) = \sum_{j=1}^{l} \frac{n_j}{n} * Entropi((X_i)_j) \quad (2.7)$$

Adım 4: Sonuç olarak, X_i geni için bilgi kazancı değeri Eşitlik 2.8 ile hesaplanabilir.

$$Bilgi\ Kazancı(X_i) = Entropi(D) - Entropi(X_i) \quad (2.8)$$

Gen seçimi yapılırken genlerin sıralanması için bir filtre olarak bilgi kazancı değeri sıklıkla kullanılmaktadır. Bir genin yüksek bilgi kazancı değerine sahip olması o genin daha fazla bilgi sağladığı anlamına gelmektedir.

Hem bilgi kazancı hem ki-kare istatistiği yöntemleri, özellikleri ayırmada yüksek performansa sahip yöntemlerdir (Y. Wang vd., 2005).

2.1.4. ERGS (Etkin Aralığa Dayalı Gen Seçimi)

Chandra ve Gupta (2011) her bir özellik için etkin aralıklar hesaplayarak bu etkin aralıkların örtüşme değerlerine dayalı bir filtre yöntemini özellik seçimi için önermişlerdir (Chandra & Gupta, 2011).

ERGS algoritması, istatistiksel çıkarsama teorisi kullanılarak tanımlanan etkin bir aralığa dayalıdır. İstatistiksel çıkarsama teorisine göre verilen bir anlamlılık seviyesinde güven düzeyi, bir aralık tahmininin güvenilirliğini göstermektedir. Buna göre bir sınıf dağılımın güven aralığı, aykırı değerlerin ve yüksek sınıf içi varyansın varlığında daha geniş bir aralık gösterebilir. Bu çalışmada, aykırı değerler ve yüksek sınıf içi varyans problemlerinin üstesinden gelebilmek için istatistiksel olarak etkin bir aralık tanımlanır.

ERGS algoritması verilen bir özelliğin her sınıfına ait etkin aralıklarını tanımlamada aralık tahmini konseptini kullanır. Bu etkin aralık da, dağılım bilgisine gerek duyulmadan (dağılım bilinmediğinde) kullanılabilen Çebişev eşitsizliğine dayanır. Etkin aralığın tanımlanmasında sınıfların ilk olasılıkları da hesaba katılır. Belirli bir özellik için, her sınıfının etkin aralıkları kullanılarak o özelliğin ağırlığı hesaplanır. Sınıflar arası karar sınırları uzaksa yani sınıfları açık bir şekilde ayırıyorsa o özelliğe daha yüksek ağırlık verilir. Yani bu da şu anlama gelmektedir; yüksek ağırlık verilen bir özelliğin etkin aralıkları örtüşmez ya da daha az örtüşme alanına (Overlapping Area, OA) sahip olur.

Bazı özelliklerdeki gen açıklama verilerinin değerleri 0-10 arasında iken, bazılarınınki 0-10000 arasında olabilmektedir. Bu da demektir ki, yüksek veri aralıklarına (data range) sahip özelliklerin örtüşme alanları daha geniş olacaktır. Bu durumun etkisini yok etmek için ERGS algoritmasında, örtüşme alanı o özelliğin veri aralığına bölünür ve bu "alan katsayısı" (area coefficient) olarak adlandırılır.

Tüm özelliklerin alan katsayıları hesaplandıktan sonra, bunlar aynı ölçekte ölçülebilsinler diye maksimum alan katsayısıyla normalleştirilir.

Özelliklerin ağırlıkları üretilirken, daha az normalleştirilmiş alan katsayısı olan özelliğe daha yüksek ağırlık verilir. Bu ağırlıklar azalan sırada sıralanır. Eğer bir özelliğin ağırlığı verilen bir eşik değerinden daha fazlaysa bu özelliğin anlamlı olduğu söylenebilir.

2.1.4.1. Etkin Aralıkların Hesaplanması

d tane özelliğin olduğu bir veri kümesi (D^{n*d}) için özellikler kümesi $X = \{X_i\} = \{X_1, X_2, ..., X_d\}$ ile gösterilsin (*i= 1,2,...,d).* Sınıf sayısı *l* olmak üzere *j.* sınıftaki örnek (gözlem) sayısı n_j, *j = 1,2,...,l* olsun. Bu durumda toplam örnek sayısı,

$$n = \sum_{j=1}^{l} n_j$$

olur ve *j.* sınıfın olasılığı

$$p_j = \frac{n_j}{n}$$

şeklinde elde edilir. *i.* özelliğin *j.* sınıfı için ortalama ve standart sapma sırasıyla μ_{ij} ve σ_{ij} olmak üzere etkin aralık Eşitlik 2.9 ile tanımlanır.

$$R_{ij} = [r_{ij}^{-}, r_{ij}^{+}] = [\mu_{ij} - (1 - p_j)\gamma\sigma_{ij}, \mu_{ij} + (1 - p_j)\gamma\sigma_{ij}] \quad (2.9)$$

Burada r_{ij}^{-} ve r_{ij}^{+} etkin aralığın sırasıyla alt ve üst sınırlarını, γ ise Eşitlik 2.10'da verilen Çebişev sabitini gösterir:

$$P\left(\left|X - \mu_{ij}\right| \geq \gamma\sigma_{ij}\right) \leq \frac{1}{\gamma^2} \quad (2.10)$$

Verinin en az 2/3'ünü içeren etkin aralık için $\gamma = 1.732$ olarak alınır. Eşitlik 2.9'da yer alan $(1 - p_j)$ değeri yüksek olasılıklı sınıfların etkisini azaltmak için varyansı küçültme amaçlı kullanılmıştır.

ERGS algoritması her bir özellik için Eşitlik 2.9'da verilen aralıkların örtüşme değerini hesaplar. Yüksek ayırt ediciliğe sahip özelliklerde sınıflar arası örtüşmenin az olması beklenir.

2.1.4.2. ERGS Algoritması

x_i özelliği için ERGS algoritmasının adımları aşağıdaki şekilde verilebilir.

1. Her bir sınıf için etkin aralıkları (R_{ij}) hesapla

2. Etkin aralıklar alt sınırlarına (r_{ij}^{-}) göre küçükten büyüğe doğru sırala

3. Örtüşme alanını

$OA_i = \sum_{j=1}^{l-1} \sum_{k=j+1}^{l} \varphi_i(j,k)$ hesapla

Burada;

$$\varphi_i(j,k) = \begin{cases} r_{ij}^{+} - r_{ik}^{-} & eğer\ r_{ik}^{-} < r_{ij}^{+}\ ise \\ 0 & \text{diğer durumlarda} \end{cases}$$

4. Alan katsayısını hesapla

$$AC_i = \frac{OA_i}{\max(r_{i1}^{+}, r_{i2}^{+}, ..., r_{il}^{+}) - r_{i1}^{-}}$$

5. Normalize edilmiş alan katsayısını hesapla

$$NAC_i = \frac{AC_i}{\max(AC_s)}, s = 1, 2, ..., d$$

6. Ağırlık katsayısını hesapla

$$w_i = 1 - NAC_i$$

7. $w_i \geq \theta$ şartını sağlayan özelliği seç

2.1.5. IFSER (Etkin Aralığa Dayalı Geliştirilen Özellik Seçimi)

Wang *vd.* (2014), ERGS algoritmasını bir sınıfa ait aralığın başka bir sınıfa ait aralığı içermesi durumunu da dikkate alacak şekilde geliştirmişlerdir.

ERGS, yalnızca her özellik için her bir sınıfın etkin aralıkları arasındaki örtüşme alanını düşünmektedir. Ancak, bu etkin aralıkların ilişkilerini içerme problemini çözmede yetersiz kalmaktadır. Bu sınırlamanın üstesinden gelebilmek için, IFSER isimli yeni bir etkin istatistiksel özellik seçimi yaklaşımı önerilmiştir. IFSER'de etkin aralıkların ilişkilerini içeren bir içerme alanı (Including Area, IA) tanıtılmıştır.

Her bir özellik için sınıflarda bulunan örneklemlerin oranı (samples' proportion) her iki yöntemde de (OA ve IA) hesaba katılmıştır.

x_i özelliği için algoritmanın adımları aşağıdaki şekilde verilebilir.

1. Her bir sınıf için etkin aralıkları (R_{ij}) hesapla

2. Etkin aralıklar alt sınırlarına (r_{ij}^{-}) göre küçükten büyüğe doğru sırala

3. Örtüşme alanını

$$OA_i = \sum_{j=1}^{l-1} \sum_{k=j+1}^{l} \varphi_i(j,k) \text{ hesapla}$$

4. İçerme alanını

$$IA_i = \sum_{j=1}^{l-1} \sum_{k=j+1}^{l} \psi_i(j,k) \quad \text{hesapla}$$

Burada

$$\psi_i(j,k) = \begin{cases} r_{ik}^{+} - r_{ik}^{-} & eğer \;\; r_{ik}^{+} < r_{ij}^{+} \; ise \\ 0 & \text{diğer durumlarda} \end{cases}$$

5. Alan katsayısını hesapla

$$AC_i = \frac{OA_i + IA_i}{\max(r_{i1}^{+}, r_{i2}^{+}, ..., r_{il}^{+}) - r_{i1}^{-}}$$

6. Normalize edilmiş alan katsayısını hesapla

$$NAC_i = 1 - \frac{AC_i}{\max(AC_s)}, s = 1, 2, ..., d$$

7. $s = 1, 2, ..., d$ için

$$NH_i = 1 - H_{ij} / \max(H_{sj})$$

ve

$$GH_i = 1 - G_{ij} / \max(G_{sj})$$

değerlerini hesapla. Burada H_{ij} ve G_{ij} değerleri j. sınıfı için OA_i ve IA_i aralıklarına düşen örnek sayısıdır.

8. Ağırlık katsayısını hesapla

$$w_i = NAC_i \times (NH_i + GH_i)$$

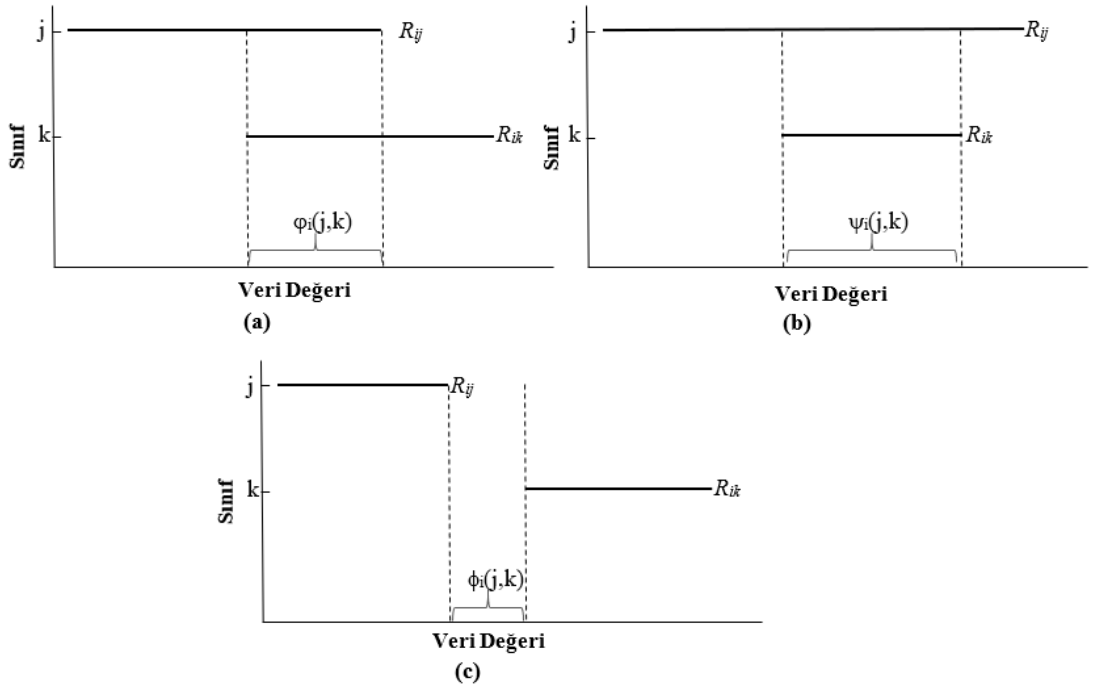
9. $w_i \geq \theta$ şartını sağlayan özelliği seç

3. ÖNERİLEN ÖZELLİK SEÇİMİ ALGORİTMASI (FSAER)

Tez çalışmasının bu aşamasında, önceki kısımlarda özellik seçim yöntemleri olarak bahsedilen ERGS ve IFSER filtre yöntemlerinden yola çıkılarak FSAER (Etkin Aralıklara dayalı Özellik Seçimi Algoritması) isimli yeni bir filtre yöntemi yaklaşımı önerilmiştir.

3.1. Motivasyon

Etkin aralıklara dayalı olan ERGS ve IFSER algoritmalarının en büyük eksikliği ayrık olan aralıkların (Şekil 3.1(c)) hepsine aynı ağırlık değerini atamasıdır. ERGS algoritması sadece Şekil 3.1(a)'da tanımlanan örtüşme alanlarını dikkate alarak özellik seçimi yapmaktadır. Algoritma Şekil 3.1(b)'de verilen durumu kısmen hesaba katarken Şekil 3.1(c) durumuna uyan aralıkların hepsi için aynı ağırlık değerini kullanmaktadır. IFSER algoritması Şekil 3.1(b) durumunu da dikkate alarak ERGS algoritmasında iyileştirme yapmıştır ancak bu algoritma da tıpkı ERGS gibi Şekil 3.1(c)'de verilen duruma uyan aralıkları kendi içinde ayırt etmemektedir.



Şekil 3.1. Etkin aralıkların birbirlerine göre durumları (a) Örtüşme (b) İçerme (c) Ayrık

Bu durumu bir örnek üzerinden göstermek için Golub verisini dikkate alalım. Golub verisi 2 sınıf ve 3051 özellikten oluşan ve toplamda 38 gözlem içeren bir veri kümesidir (Golub vd., 1999). Bu veri kümesinde etkin aralıklar hesaplandıktan sonra Şekil 3.1'de verilen a) örtüşme (b) içerme (c) ayrık durumlarının sayısı Çizelge 3.1'de özetlenmiştir.

Çizelge 3.1. Golub verisinde (38×3051) etkin aralıklar hesaplandıktan sonra OA, IA ve AA durumlarının sayısı

Örtüşme alanı (OA) sayısı	İçerme alanı (IA) sayısı	Ayrık alan (AA) sayısı
1151	1713	187

Çizelge 3.1'de görüldüğü gibi 187 özellik için $\phi_i(j,k)$ değeri sıfırdan büyüktür. ERGS ve IFSER algoritmaları $\phi_i(j,k)$ değerinin büyüklüğü ne olursa olsun bu özelliklerin hepsine aynı ağırlık değerini vermektedir. Örneğin bu 187 özellik için $\max\left[\phi_{829}(1,2)\right]=1.461251$ iken $\min\left[\phi_{1206}(1,2)\right]=0.0001502558$ dır ve ERGS algoritmasındaki OA_i değeri ile IFSER algoritmasındaki OA_i ve IA_i değerleri bu iki özellik için de 0'dır. Oysa, büyük $\phi_i(j,k)$ değerleri ayırt etme gücünü arttırabilir.

Bu çalışmada $\phi_i(j,k)$ ile gösterilen ayrık etkin aralıkları da ayırt eden yeni bir özellik seçim algoritması (FSAER) önerilmektedir. Bu algoritmanın adımları aşağıda verilmiştir.

3.2. FSAER Algoritması

x_i özelliği için algoritmanın adımları aşağıdaki şekilde verilebilir.

1. Her bir sınıf için etkin aralıkları (R_{ij}) hesapla
2. Etkin aralıklar alt sınırlarına (r_{ij}^{-}) göre küçükten büyüğe doğru sırala
3. Toplam alanı (TA)

$$TA_i=\sum_{j=1}^{l-1}\sum_{k=j+1}^{l}\varphi_i(j,k)+\psi_i(j,k)-\phi_i(j,k) \quad \text{hesapla}$$

Burada

$$\varphi_i(j,k) = \begin{cases} r_{ij}^{+} - r_{ik}^{-} & \text{if } r_{ik}^{-} < r_{ij}^{+} \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$\psi_i(j,k) = \begin{cases} r_{ik}^{+} - r_{ik}^{-} & \text{if } r_{ik}^{+} < r_{ij}^{+} \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$\phi_i(j,k) = \begin{cases} r_{ik}^{-} - r_{ij}^{+} & \text{if } r_{ij}^{+} < r_{ik}^{-} \\ 0 & \text{diğer durumlarda} \end{cases}$$

4. Alan katsayısını hesapla

$$AC_i = \frac{TA_i}{\max(r_{i1}^{+}, r_{i2}^{+}, ..., r_{il}^{+}) - r_{i1}^{-}}$$

5. Normalize edilmiş alan katsayısını hesapla

$$NAC_i = \frac{AC_i}{\max(AC_j)}, \qquad j = 1, 2, ..., d$$

6. Ağırlık katsayısını hesapla

$$w_i = 1 - NAC_i$$

7. $w_i \geq \theta$ şartını sağlayan özelliği seç

4. ÖZELLİK SEÇİM YÖNTEMLERİNİN DEĞERLENDİRİLMESİ

DNA mikrodizi görüntülerinden verilerinin sayısallaştırılması ile elde edilen gen açıklama verileri, genleri ve açıklama düzeylerini içeren bir matris biçimindedir. Bu matrislerde bulunan az sayıda örneğin binlerce gen bilgisi, önce gen seçimi (özellik seçimi) yapılarak azaltılır. Seçilen bu genlerin açıklama düzeylerine göre birbirlerine benzer olanları ya da olmayanları ise sınıflandırma sonucunda belirlenebilir. Bu işlem sonrası farklı koşullarda veya farklı zamanlarda benzer değişim gösteren genler belirlenebilmekte, hastalıkların tanı ve tedavisinde etkili olabilecek gen bilgisine ulaşılabilmektedir.

Özellik seçim yöntemlerinin karşılaştırılabilmesi için bu yöntemler sonucu elde edilen genlerin sınıflandırma yapılarak değerlendirilmesi gerekmektedir. Bu noktada doğru sınıflandırmanın yalnızca tek bir sınıflandırma algoritmasına bağlı olmadığını göstermek adına farklı sınıflandırıcılarla yöntemler karşılaştırılmaktadır.

Bu bölümde ilk olarak kullanılan sınıflandırma yöntemleri tanıtılmış, ikinci kısımda ise doğru sınıflandırma yapabilmek için gerekli olan geçerlilik ölçütlerine yer verilmiştir.

4.1. Sınıflandırma Yöntemleri

Bu çalışmada, önerilen filtre yöntem ile diğer filtre yöntemlerden elde edilen sonuçlar doğru sınıflandırma oranları açısından karşılaştırılmıştır. Sınıflandırma yöntemi olarak filtre yöntemlerini karşılaştırmada sıklıkla tercih edilen DVM, Naif Bayes sınıflandırıcısı (Naive Bayes Classifier, NB) ve k-En yakın komşu (k-Nearest Neighbors, kNN) algoritmaları kullanılmıştır (Bommert vd., 2020; Chandra & Gupta, 2011; Shukla vd., 2018; J. Wang vd., 2014; Zheng vd., 2019).

4.1.1. Destek Vektör Makineleri (DVM)

DVM (Cortes & Vapnik, 1995) genellikle sınıflandırma problemlerinde kullanılan denetimli (supervised) bir öğrenme algoritmasıdır. Yapılan çalışmalar mikrodizi verilerini sınıflandırmada DVM'nin oldukça başarılı sonuçlar verdiğini göstermektedir (Furey vd., 2000; Y. Lee & Lee, 2003; Ramaswamy vd., 2001).

DVM karar sınırı olarak maksimum kenar boşluğu prensibine göre özellik uzayında optimal hiper düzlemler oluşturarak sınıflandırma işlemini gerçekleştirir (Bommert vd., 2020).

Başka bir deyişle DVM temel olarak; doğru, düzlem ya da hiper düzlemler (karar sınırları) yardımıyla farklı sınıflara ait verileri birbirinden en uygun şekilde ayırmayı amaçlar.

Optimal bir hiper düzlem oluşturmak için, $\Lambda(\mathrm{w})$ hata fonksiyonunu minimize edecek iteratif bir öğrenme algoritması kullanılır. Bu hata fonksiyonu $\Lambda(\mathrm{w})$ Eşitlik 4.1 ile tanımlanabilir (Chandra & Gupta, 2011):

$$\Lambda(\mathrm{w}) = \frac{1}{2} w^T w + C \sum \xi_i \tag{4.1}$$

kısıtlar:

$$y_i \left[w^T K(\mathrm{x}_i) + \mathrm{b} \right] \geq 1 - \xi_i \ ve \ \xi \geq 0, \ i = 1, 2, ..., n$$

Burada w katsayılar vektörünü, b sabit bir sayıyı ve $\xi_i, \ i = 1, 2, ..., n$ yanlış sınıflamaya neden olabilecek parametreleri göstermektedir. Her i örneği için, x_i ; y_i sınıf etiketleriyle temsil edilen bağımsız değişkenlerdir. Kernel fonksiyonu (K) girdi verilerini ileri boyutlu özellik uzayına dönüştürür. Bu da doğrusal olmayan karar sınırlarını oluşturmada kullanılır (Izenman, 2008).

Bu çalışmada, dönüştürme için doğrusal Kernel fonksiyonu kullanılmıştır. C parametresi, aşırı öğrenmenin kontrol edilmesi için bir yol olarak görülebilir. C değeri ne kadar büyük olursa hata o kadar cezalandırılır.

Standart DVM iki sınıflı problemlere uygulanabilir. Çok sınıflı problemler ise ya ikili sınıflandırıcılardan çok sınıflı sınıflandırıcı yaratılarak ya da doğrudan çok sınıflı DVM uygulanarak çözülür (Jiang vd., 2005; H. Li vd., 2005).

DVM yöntemini bu çalışmada uygulamak için R programlama dili içerisindeki Caret paketinden yararlanılmıştır. Caret paketi girdi parametrelerinin optimizasyonunu kendi içerisinde yapmaktadır.

4.1.2. Naif Bayes Sınıflandırıcısı (NB)

NB sınıflandırıcısı, Bayes teoremine dayanan, özellikler arasında bağımsızlık olduğunu varsayarak olasılık ilkelerine göre çalışan bir sınıflandırıcıdır (Friedman vd., 1997; Rish, 2001). Domingos ve Pazzani (1997), bu varsayımın düşünülenden daha az etkili olduğunu bulmuşlardır. NB'nin gen açıklama verilerinde diğer sınıflandırma yöntemlerine göre genellikle daha iyi sınıflandırma doğruluğu sağladığı görülmüştür (Chandra & Gupta, 2011).

Bayes olasılığı, koşullu olasılığın k tane ayrık olay için genelleştirilmiş halidir. Bu olasılık Eşitlik 4.2 ile tanımlanabilir (Altunkaynak, 2017):

$$P(\mathrm{C}_j / x) = \frac{P(x / \mathrm{C}_j)\,\mathrm{P}(\mathrm{C}_j)}{P(\mathrm{X})} \tag{4.2}$$

Burada;

$P(x / \mathrm{C}_j)$: Sınıf j'de x durumunun ortaya çıkma olasılığı

$\mathrm{P}(\mathrm{C}_j)$: j sınıfının ortaya çıkma olasılığı (ilk olasılık)

$P(x)$: x durumun ortaya çıkma olasılığı

$P(\mathrm{C}_j / x)$: x durumu biliniyorken sınıf j'den olma olasılığı (son olasılık) olarak tanımlanabilir.

C_j, $j \in \{1,...,J\}$ olmak üzere J tane sınıftan birini temsil eden ayrık bir değişkendir. X özelliği p adet özellikten oluşan $X = (x_1,...,x_p)$ özellik vektörü ile ifade edilmektedir. NB sınıflandırıcı sınıflandırmada son olasılığı en büyütmeyi amaçlar (max $P(C_j / X)$. Eşitlik 4.2'deki Bayes teoreminden yola çıkılırsa;

$P(C_j / X) = \frac{P(X / C_j) P(C_j)}{P(X)}$ elde edilir.

$P(X)$ olasılığı tüm sınıflar için sabit olacağından yalnızca $P(C_j / X) = P(X / C_j) P(C_j)$ olasılığı için en büyük değer aranacaktır. Bu olasılık değerindeki tüm özellikler bağımsızsa $P(X / C_j)$ Eşitlik 4.3'deki gibi yazılabilir:

$$P(X / C_j) = \prod_{k=1}^{p} P(x_k / C_j) = P(x_1 / C_j) * P(x_2 / C_j) * ... * P(x_p / C_j) \quad (4.3)$$

NB yöntemini bu çalışmada uygulamak için R programlama dili içerisindeki Caret paketinden yararlanılmıştır. Caret paketi girdi parametrelerinin optimizasyonunu kendi içinde yapmaktadır.

4.1.3. k -En Yakın Komşu Yöntemi (kNN)

kNN (Altman, 1992) makine öğrenmesi alanında en çok kullanılan sınıflandırma algoritmalardan biridir (Coomans & Massart, 1982; Raniszewski, 2010; Sánchez vd., 2007). Bağımsız değişkenlerin sayısal olduğu (gen verilerinde de olduğu gibi) durumlara daha uygun olan bu yöntem, gözlemler arasındaki uzaklıklara bağlı olarak sınıflandırma işlemini gerçekleştirir (Bolón-Canedo vd., 2014).

kNN sınıflandırmasında çıktı bir sınıf üyeliğidir. Bir gözlem, komşularının çoğunluk oyu (majority vote) ile sınıflandırılır ve en yakın komşuları arasında en yaygın olan sınıfa atanır (Zheng vd., 2019).

Yöntemin işleyiş aşamaları aşağıdaki gibi özetlenebilir (Harrington, 2012):

1. k değerinin (burada k pozitif bir tamsayıdır) belirlenir.
2. Sınıf değeri bulunmak istenen gözlemin veri kümesinde bulunan tüm gözlemlere uzaklığı hesaplanır.
3. Bulunan uzaklıklara göre gözlemler küçükten büyüğe doğru sıralanır.
4. k en küçük uzaklık değerine sahip gözlemler alınır.
5. k tane gözlemde en fazla tekrar eden sınıf tahmini sınıf değeri olarak bulunur.

Yöntemin düzgün çalışması, komşuların sayısını temsil eden k parametresi ve kullanılan uzaklık ölçütü gibi parametrelerin seçimine bağlıdır. Seçilen k değeri çok küçük olursa sonuçlar aykırı değerlere karşı aşırı duyarlı olurken, çok büyük olduğunda ise seçilen komşular birçok sınıfın elemanlarından oluşarak sınıflandırma performansını olumsuz yönde etkileyecektir. Bu gibi problemlerin üstesinden gelebilmek için literatürde k değerinin ideal seçimi için bir ön çalışma yapılarak çapraz geçerliliklerin incelenmesi önerilmektedir.

kNN yöntemini bu çalışmada uygulamak için R programlama dili içerisindeki Caret paketinden yararlanılmıştır. Caret paketi girdi parametrelerinin optimizasyonunu kendi içinde yapmaktadır.

4.2. Geçerlilik Ölçütleri

Sınıflandırma algoritmalarında genellikle veriler başlangıçta eğitim ve test kümelerine dağıtılır. Eğitim kümesi özellik seçim sürecini gerçekleştirirken, test kümesi seçimin uygunluğunu değerlendirmek için kullanılır. Burada dikkat edilmesi gereken eğitim ve test kümelerinin nasıl ayrılacağının belirlenmesidir. Yüzdesel bölme, k-katlı çapraz geçerlilik (k-fold Cross-Validation, k-fold CV) ve 1-eksiltme çapraz geçerlilik (Leave-One-Out Cross-Validation, LOOCV) bu alanda kullanılan temel yaklaşımlardandır (Bolón-Canedo vd., 2014; Zaki & Meira, 2013).

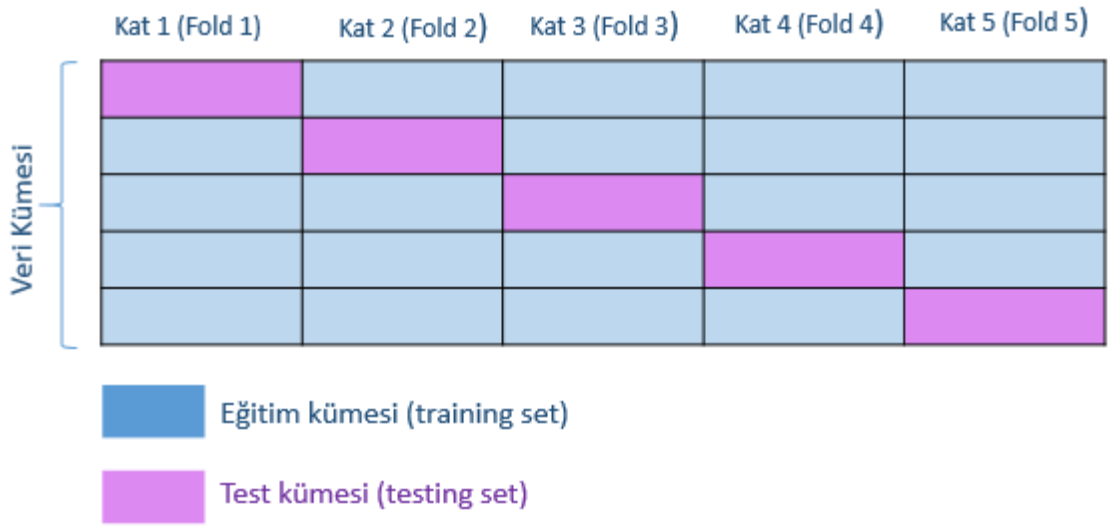
4.2.1. Yüzdesel Bölme (Percentile Split)

Bu yöntem mevcut verilerin eğitim ve test olmak üzere iki gruba bölünmesinden oluşur. Bu bölünme birçok farklı şekilde yapılabilir, yaygın olarak kullanılan oranlardan bazıları sırasıyla eğitim ve test kümeleri olacak şekilde 1: 1, 2: 1, 70:30 veya 60:40 oranlarıdır (Hand, 2007).

Buradaki başlıca dezavantajlardan biri, örnek sayısının az olduğu durumlarda verileri belirli oranlarda ayırmanın zor olmasıdır. Bir diğer olumsuzluk ise, eğitim ve test kümesi ayrımı bir kez yapıldığı için uygun olmayan bir ayrım söz konusu olduğunda hata oranı yanıltıcı olabilmektedir. Bu olumsuzluklar, tek bir deneme yerine çok sayıda deneme içeren yöntemler kullanılarak giderilebilir.

4.2.2. k-Katlı Çapraz Geçerlilik (k-fold CV)

k-fold CV, özellikle örnek sayısı az olduğu durumlarda tercih edilen ve en çok bilinen geçerlilik ölçütlerinden biridir (Hand, 2007). Veri kümesi, eşit büyüklükte k tane alt kümeye bölünür. İlk olarak bu k tane alt grubun 1 tanesi test kümesini, (k-1) tanesi ise eğitim kümesini oluşturur. Bu işlem, k alt kümelerinin her biri için sırayla tekrarlanır yani diğer bir deyişle k tane test yapılır. Bu sayede her örnek mutlaka hem eğitim kümesi hem de test kümesi içinde yer alacaktır. Çapraz geçerlilik hatası, bu şekilde elde edilen tahmin hatalarının ortalaması alınarak hesaplanır (Bolón-Canedo vd., 2014).



Şekil 4.1. 5-fold CV (k=5) işleyişi

4.2.3. 1-eksiltme Çapraz Geçerlilik (LOOCV)

LOOCV yöntemi, k'nın örnek sayısına eşit olduğu özel bir k-katlı çapraz geçerlilik biçimidir. Örnek sayısı N ile gösterilecek olursa, k-katlı çapraz geçerlilik yönteminde olduğu gibi burada da, 1 tane örnek, test amacı ile kullanılır, kalan (N-1) tane örnek ise eğitim kümesini oluşturur. Böylelikle her bir örnek test için tam olarak bir kez kullanılmaktadır (Ding & Peng, 2003; Khan vd., 2001; Wong, 2015).

Bu tez çalışmasında, veri kümelerindeki örnek sayısı az olduğundan, değerlendirme için LOOCV kullanılmıştır.

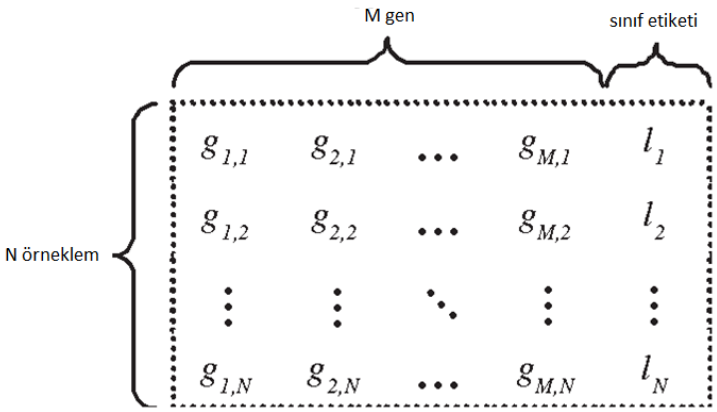
5. DENEYSEL SONUÇLAR

Önerilen özellik seçme algoritmasının (FSAER) performansı; **Golub_1** (Golub vd., 1999), **Golub_2** (Zhu vd., 2007), **Kolon** (Alon vd., 1999), **Lösemi** (Chiaretti vd., 2004), **Prostat** (Singh vd., 2002) ve **SRBCT** (Khan vd., 2001) olmak üzere literatürde bilinen altı farklı gen açıklama veri kümesi kullanılarak değerlendirilmiştir. Seçilen özelliklerin (FSAER kullanılarak) sınıflandırma performanslarının ölçülmesinde DVM, NB sınıflandırıcısı ve kNN sınıflandırma yöntemleri olarak kullanılmıştır. LOOCV uygulanarak elde edilen doğruluk oranları çizelgeler halinde detaylı olarak sunulmuştur.

ERGS, IFSER ve FSAER algoritmaları R programlama dilinde kodlanmıştır. Ki-kare, Relief-F ve IG algoritmaları için Biocomb R paketi, doğru sınıflandırma oranlarının elde edilmesi için ise Caret R paketi kullanılmıştır. İlgili kodlamalar EK 1'de verilmiştir.

5.1. Kullanılan Veri Kümeleri

Gen açıklama veri kümeleri sayısal ifadelerden oluşmakta, satırlarda örneklem sütunlarda ise genlerden oluşan matrislerle ifade edilmektedir. Son sütunda da sınıf etiketi içeren veri kümelerinin genel gösterimi Şekil 5.1'de verilmiştir.



Şekil 5.1. Gen açıklama veri kümelerinin matris halinde genel gösterimi

Kullanılan veri kümeleri ve özellikleri Çizelge 5.1'de verilmiş olup, veriler ile ilgili ayrıntılı bilgiler alt başlıklar halinde sunulmuştur.

Çizelge 5.1. Kullanılan veri kümelerinin özellikleri ve erişilebilir kaynakları

Veri Kümesi	Örneklem Büyüklüğü	Gen Sayısı	Sınıf Sayısı	Kaynak
Golub_1	72	7129	2	https://www.rdocumentation.org/packages/datamicroarray/versions/0.2.3
Golub_2	72	7129	3	http://csse.szu.edu.cn/staff/zhuzx/Datasets.html
Kolon	62	200	2	https://www.rdocumentation.org/packages/datamicroarray/versions/0.2.3
Lösemi	111	12625	2	https://www.rdocumentation.org/packages/datamicroarray/versions/0.2.3
Prostat	102	12600	2	https://www.rdocumentation.org/packages/datamicroarray/versions/0.2.3
SRBCT	83	2308	4	https://cran.r-project.org/web/packages/plsgenomics/plsgenomics.pdf

5.1.1. Golub_1 (ALL-AML)

Golub_1 gen açıklama veri kümesi, 47'si Akut Lenfoblastik Lösemi (ALL) ve 25'i Akut Miyeloid Lösemi (AML) olmak üzere 72 hastanın bilgilerini içermektedir. 72 hastanın her birinden tanı sırasında elde edilen kemik iliği örnekleri mevcuttur. 2 sınıflı bu veri kümesinde 72 hastaya ait 7129 gen bilgisi bulunmaktadır.

5.1.2. Golub_2 (ALL-AML_3)

Golub_2 gen açıklama veri kümesi, 25 Akut Miyeloid Lösemi (AML), 38 B-hücreli Akut Lenfoblastik Lösemi (ALL) ve 9 T-hücreli Akut Lenfoblastik Lösemi (ALL) olmak üzere toplam 72 hastanın bilgilerini içermektedir. 72 örnek ve 3 sınıfa sahip bu veri kümesinde 7129 gen bilgisi bulunmaktadır.

5.1.3. Kolon

Kolon gen açıklama veri kümesi, 62 farklı kolon kanseri hastasından alınan kolon epitel hücre örneklerini içermektedir. Örnekler aynı hastanın kolonlarının sağlıklı kısımlarından toplanan normal biyopsiler ve tümörlerden toplanan tümörlü biyopsileri içermektedir. Tümörlü ve sağlıklı olmak üzere 2 sınıflı olan bu veri kümesinde 62 hastaya ait 2000 gen bilgisi bulunmaktadır.

5.1.4. Lösemi (ALL)

Lösemi gen açıklama veri kümesi, T-hücreli ve B-hücreli Akut Lenfoblastik Lösemi (ALL)'ye sahip toplam 111 hastanın bilgilerini içermektedir. 111 örnek ve 2 sınıflı bu veri kümesinde 12625 gen bilgisi bulunmaktadır.

5.1.5. Prostat

Prostat gen açıklama veri kümesi, 52 tane tümörlü prostat örneği ve 50 tane tümörsüz prostat örneği olmak üzere yüksek kaliteli olduğu rapor edilen toplam 102 radikal prostatektomi türlerinden oluşmaktadır. 102 örnek ve 2 sınıfa sahip bu veri kümesinde 12600 gen bilgisi bulunmaktadır.

5.1.6. SRBCT (Küçük Yuvarlak Mavi Hücreli Tümörler)

SRBCT gen açıklama veri kümesi, çocukluk çağı kanseri çalışmalarından elde edilmiştir. Bu veri kümesinde 29'u Ewing Sarkomu (EWS), 11'i Burkitt Lenfoma (BL), 18'i Nöroblastom (NB) ve 25'i Rabdomiyosarkom (RMS) olan toplamda 83 örnek vardır. 83 örnek ve 4 sınıfa sahip bu veri kümesinde 2308 gen bilgisi bulunmaktadır.

5.2. Uygulama

Yöntemlerden R programlama dili yardımıyla elde edilen sonuçlar Çizelge 5.2, 5.3 ve 5.4'te sunulmuştur.

Çizelge 5.2. Farklı özellik seçme yöntemleri ile seçilen (10-100) ve farklı veri kümeleri kullanılarak elde edilen NB sınıflandırıcısı sınıflandırma doğrulukları (%) (LOOCV)

		Seçilen Özellikler					
Veri Kümesi	**Filtre Yöntemi**	**10**	**20**	**40**	**60**	**80**	**100**
Golub_1 (72×7129) 2 sınıflı	χ^2	95,83	95,83	95,83	95,83	95,83	95,83
	Relief-F	90,28	93,05	94,44	95,83	95,83	95,83
	IG	95,83	95,83	95,83	95,83	97,22	97,22
	ERGS	100	98,61	98,61	98,61	98,61	98,61
	IFSER	100	98,61	98,61	98,61	98,61	98,61
	FSAER	95,83	98,61	100	98,61	98,61	98,61
Golub_2 (72×7129) 3 sınıflı	χ^2	81,94	91,67	94,44	97,22	97,22	98,61
	Relief-F	94,44	93,06	95,83	95,83	97,22	97,22
	IG	95,83	95,83	95,83	97,22	97,22	97,22
	ERGS	97,22	95,83	97,22	97,22	97,22	97,22
	IFSER	97,22	98,61	98,61	97,22	97,22	98,61
	FSAER	98,61	97,22	97,22	97,22	97,22	98,61
Kolon (62×2000) 2 sınıflı	χ^2	85,48	85,48	85,48	87,10	85,48	87,10
	Relief-F	82,26	85,48	87,10	87,10	87,10	88,71
	IG	87,10	87,10	83,87	82,26	82,26	83,87
	ERGS	80,64	77,41	77,41	79,03	79,03	79,03
	IFSER	75,80	79,03	80,64	77,41	79,03	75,80
	FSAER	80,64	77,41	77,41	79,03	79,03	79,03
Lösemi (111×12625) 2 sınıflı	χ^2	82,88	79,28	81,08	79,28	76,58	77,48
	Relief-F	58,82	70,58	67,65	66,67	79,41	78,43
	IG	81,08	77,48	81,08	77,48	77,48	75,68
	ERGS	84,68	82,88	85,58	86,49	85,58	85,58
	IFSER	84,68	83,78	86,49	85,58	84,68	85,58
	FSAER	85,58	81,98	85,58	86,49	85,58	85,58
Prostat (102×12600) 2 sınıflı	χ^2	92,17	92,17	89,22	92,17	93,14	93,14
	Relief-F	56,86	60,78	72,55	72,55	70,59	70,59
	IG	92,16	91,18	91,18	93,14	93,14	93,14
	ERGS	94,12	94,12	93,14	92,17	92,17	93,14
	IFSER	95,10	94,12	93,14	92,17	92,17	91,18
	FSAER	95,10	95,10	93,14	92,17	92,17	93,14

SRBCT (83×2308) 4 sınıflı	χ^2	95,18	96,38	97,59	98,79	100	100
	Relief-F	60,24	68,67	90,36	92,77	90,36	92,77
	IG	97,59	97,59	100	100	100	100
	ERGS	98,79	98,79	100	100	100	100
	IFSER	85,54	96,38	98,79	100	100	100
	FSAER	98,79	100	100	100	100	100

Çizelge 5.3. Farklı özellik seçme yöntemleri ile seçilen (10-100) ve farklı veri kümeleri kullanılarak elde edilen DVM sınıflandırıcısı sınıflandırma doğrulukları (%) (LOOCV)

		Seçilen Özellikler					
Veri Kümesi	**Filtre Yöntemi**	**10**	**20**	**40**	**60**	**80**	**100**
Golub_1 (72×7129) 2 sınıflı	χ^2	88,88	93,05	94,44	95,83	98,61	98,61
	Relief-F	87,50	94,44	93,06	97,22	97,22	95,83
	IG	97,22	91,67	98,61	97,22	98,61	98,61
	ERGS	98,61	97,22	95,83	98,61	100	100
	IFSER	98,61	97,22	95,83	97,22	100	100
	FSAER	95,93	93,05	95,83	98,61	100	100
Golub_2 (72×7129) 3 sınıflı	χ^2	93,05	93,05	95,83	97,22	94,44	95,83
	Relief-F	91,67	94,44	98,61	98,61	98,61	98,61
	IG	94,44	94,44	95,83	95,83	95,83	95,83
	ERGS	97,22	97,22	98,61	98,61	95,83	97,22
	IFSER	93,05	97,22	95,83	95,83	95,83	97,22
	FSAER	97,22	95,83	98,61	97,22	97,22	97,22
Kolon (62×2000) 2 sınıflı	χ^2	83,87	79,03	83,87	87,10	85,48	83,87
	Relief-F	79,03	87,09	77,42	77,42	82,26	85,48
	IG	83,87	82,25	72,58	80,64	82,26	87,10
	ERGS	85,48	82,25	85,48	80,64	80,64	79,03
	IFSER	75,80	85,48	85,48	82,25	77,41	75,80
	FSAER	85,48	82,25	85,48	80,64	80,64	79,03
Lösemi (111×12625) 2 sınıflı	χ^2	85,58	81,08	75,67	72,07	77,48	84,68
	Relief-F	65,69	70,59	84,31	80,39	81,37	81,37
	IG	82,88	77,48	79,28	81,98	78,38	82,88
	ERGS	90,09	93,69	90,10	89,19	92,79	93,69
	IFSER	89,19	93,69	91,89	87,39	90,09	89,19
	FSAER	88,29	92,79	92,79	93,69	91,89	92,79

Prostat (102×12600) 2 sınıflı	χ^2	97,06	93,13	94,11	93,13	95,10	94,11
	Relief-F	55,88	65,69	76,47	69,61	69,61	82,35
	IG	96,08	89,22	91,18	93,14	94,12	95,10
	ERGS	93,14	96,08	95,10	92,17	92,17	93,14
	IFSER	92,17	96,08	92,17	89,26	90,20	91,18
	FSAER	92,17	97,06	92,17	92,17	92,17	94,12
SRBCT (83×2308) 4 sınıflı	χ^2	96,38	96,38	96,38	98,79	98,79	100
	Relief-F	63,85	77,11	92,77	96,38	97,59	100
	IG	96,38	96,38	100	100	100	100
	ERGS	100	100	100	100	100	100
	IFSER	89,15	98,79	100	100	100	100
	FSAER	96,38	100	100	100	100	100

Çizelge 5.4. Farklı özellik seçme yöntemleri ile seçilen (10-100) ve farklı veri kümeleri kullanılarak elde edilen kNN sınıflandırıcısı sınıflandırma doğrulukları (%) (LOOCV)

		Seçilen Özellikler					
Veri Kümesi	**Filtre Yöntemi**	**10**	**20**	**40**	**60**	**80**	**100**
Golub_1 (72×7129) 2 sınıflı	χ^2	86,11	93,05	94,44	91,67	93,06	93,06
	Relief-F	88,89	94,44	94,44	91,67	90,27	91,67
	IG	90,28	95,83	95,83	93,05	94,44	93,05
	ERGS	98,61	98,61	97,22	98,61	97,22	95,83
	IFSER	98,61	98,61	97,22	95,83	95,83	95,83
	FSAER	95,83	97,22	97,22	98,61	97,22	95,83
Golub_2 (72×7129) 3 sınıflı	χ^2	94,44	91,67	93,05	95,83	95,83	95,83
	Relief-F	90,28	90,28	91,67	94,44	94,44	95,83
	IG	90,28	93,05	95,83	95,83	97,22	95,83
	ERGS	97,22	97,22	95,83	95,83	95,83	97,22
	IFSER	97,22	93,05	95,83	93,05	97,22	95,83
	FSAER	94,44	95,83	97,22	94,44	95,83	95,83
Kolon (62×2000) 2 sınıflı	χ^2	83,87	85,48	85,48	85,48	82,26	82,26
	Relief-F	83,87	87,10	87,10	87,10	85,48	85,48
	IG	85,48	85,48	83,87	83,87	87,10	85,48
	ERGS	85,48	85,48	88,70	85,48	87,10	85,48
	IFSER	77,41	85,48	87,10	87,10	87,10	85,48
	FSAER	85,48	85,48	88,70	85,48	87,10	85,48

Lösemi (111×12625) 2 sınıflı	χ^2	87,39	82,88	83,78	83,78	85,58	82,88
	Relief-F	58,82	62,74	76,47	82,35	84,31	82,35
	IG	86,49	81,98	86,48	83,78	81,08	79,28
	ERGS	86,49	87,39	87,39	87,39	85,58	86,49
	IFSER	87,39	89,19	89,19	88,29	89,19	88,29
	FSAER	86,49	87,39	87,39	86,48	84,68	88,29
Prostat (102×12600) 2 sınıflı	χ^2	90,20	89,22	88,24	89,22	92,17	92,17
	Relief-F	54,90	55,88	76,47	76,47	79,41	78,43
	IG	90,20	89,22	88,24	89,22	94,11	94,11
	ERGS	92,17	94,18	90,20	90,20	90,20	90,20
	IFSER	91,18	90,20	90,20	90,20	90,20	90,20
	FSAER	91,18	93,14	90,20	90,20	92,17	90,20
SRBCT (83×2308) 4 sınıflı	χ^2	96,38	97,59	98,79	98,79	98,79	100
	Relief-F	54,22	54,22	84,34	90,36	90,36	89,16
	IG	96,38	98,79	98,79	98,79	100	100
	ERGS	97,59	98,79	98,79	100	100	100
	IFSER	84,33	91,56	96,38	98,79	98,79	98,79
	FSAER	96,38	98,79	97,59	98,79	100	100

5.3. Sonuçların Karşılaştırılması

FSAER algoritmasının etkinliğini araştırmak için, bu algoritma kullanılarak seçilen özelliklerin sınıflandırma doğrulukları ile diğer özellik seçme algoritmalarından elde edilen doğruluklar hesaplanarak karşılaştırılmıştır. Daha iyi bir karşılaştırma yapabilmek için sınıflandırma doğruluklarının hesaplanmasında LOOCV kullanılmıştır.

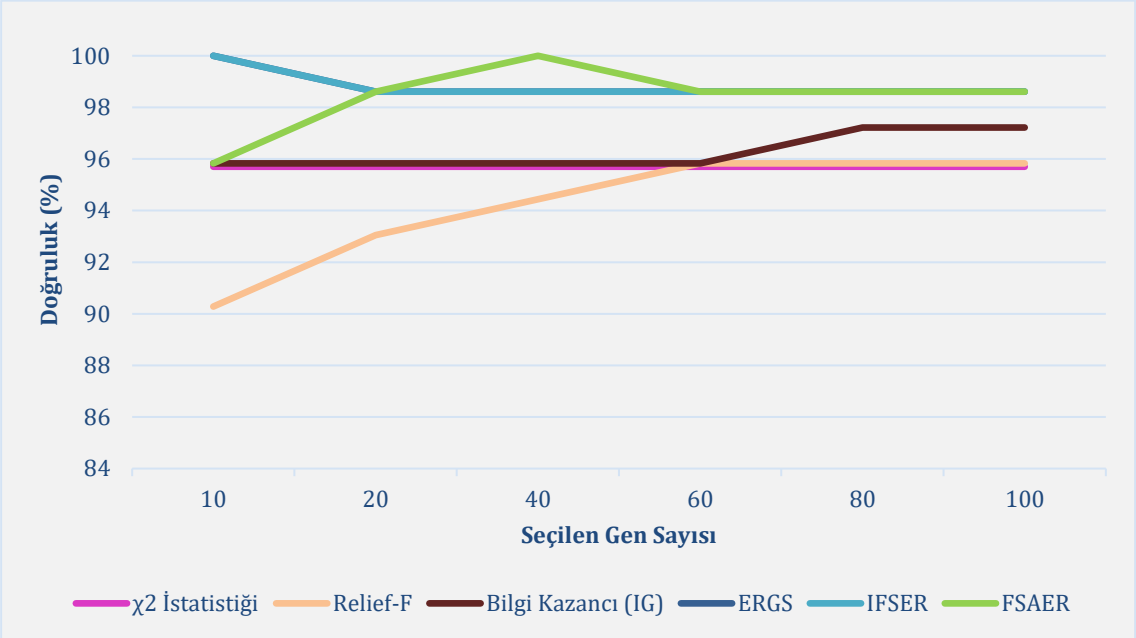
Altı farklı veri kümesi üzerinde, seçilen farklı büyüklükteki özellik (gen) alt kümeleri (10-100 arasında) için χ^2 istatistiği, Relief-F, IG, ERGS, IFSER ve FSAER yöntemlerinin uygulanmasından elde edilen sınıflandırma doğrulukları Çizelge 5.2, 5.3 ve 5.4'te sunulmuştur.

5.3.1. NB ile Elde Edilen Sonuçlar

Çizelge 5.2, altı farklı gen açıklama veri kümesi için özellik seçme yöntemlerinden elde edilen farklı büyüklükteki özellik alt kümelerinin (10-100) NB sınıflandırıcısı kullanılması sonucunda bulunan sınıflandırma doğruluklarını göstermektedir.

Çizelge 5.2'ye genel olarak bakıldığında tüm veri kümeleri için FSAER yönteminin en yüksek doğruluk oranlarına bazen tek başına bazen de diğer yöntemlerle birlikte ulaştığı görülmektedir. Sonuçları ayrıntılı değerlendirebilmek için bu çizelgedeki veri kümeleri tek tek incelenmiştir:

Golub_1 veri kümesi için elde edilen sonuçlar incelendiğinde, en yüksek doğruluk oranı olan 100 (%) değerine FSAER yönteminin 40 tane özellik seçildiğinde ulaştığı görülmektedir. Ayrıca seçilen özellik büyüklükleri 20, 40, 60, 80 ve 100 olduğu durumlar için de FSAER algoritması diğer yöntemlere göre oldukça iyi sonuçlar vermiştir. Golub_1 veri kümesi için sonuçlar Şekil 5.2'de gösterilmiştir.

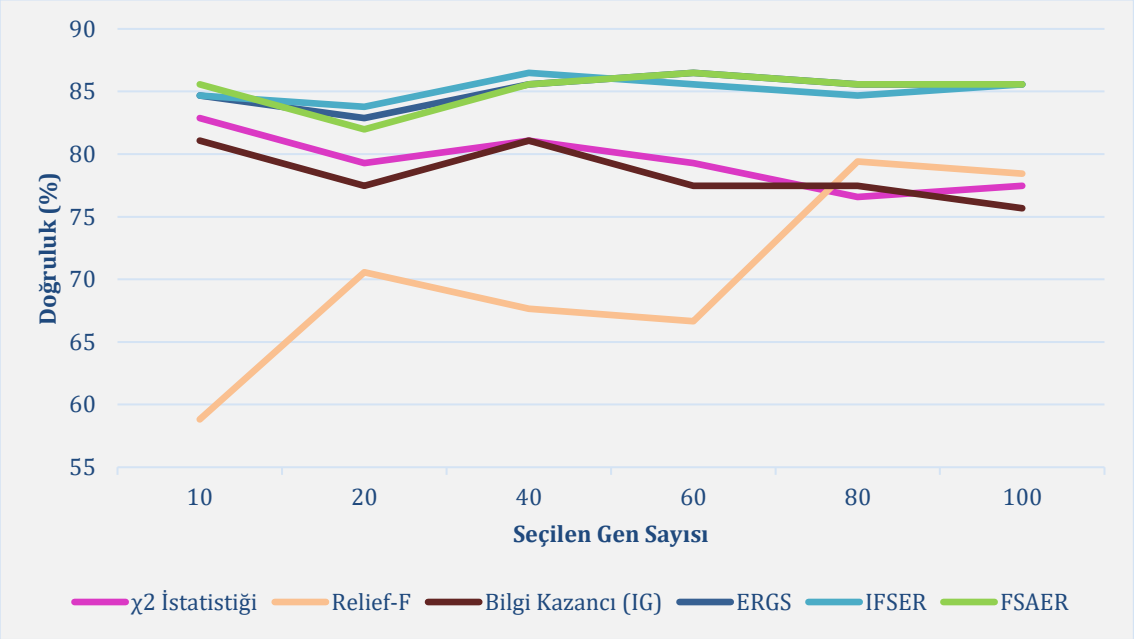


Şekil 5.2. Golub_1 veri kümesi için NB sınıflandırma doğrulukları (%)

Golub_2 veri kümesi için elde edilen sonuçlar incelendiğinde, en yüksek doğruluk oranı olan 98.61 (%) değerine FSAER yönteminin 10 ve 100 tane özellik seçildiği durumlarda ulaştığı görülmektedir.

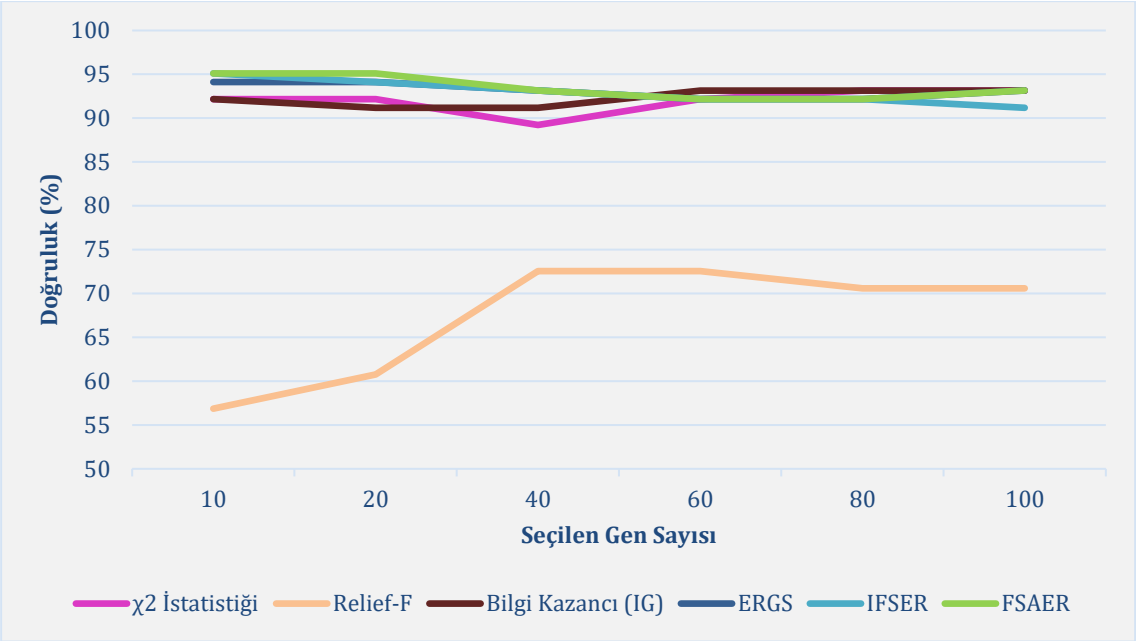
Kolon veri kümesi için en yüksek doğruluğu Relief-F yöntemi sağlarken, FSAER ve ERGS yöntemlerinin sonuçlarının aynı olduğu görülmüştür.

Lösemi veri kümesinde, Şekil 5.3'te görüldüğü gibi tablonun en yüksek doğruluk oranı olan 86.49 (%) değerini 60 özellik seçildiğinde ERGS ve FSAER birlikte vermişlerdir. 10 özellik seçildiği durumda ise FSAER diğer tüm yöntemlerden daha iyi sonuç vermiştir.



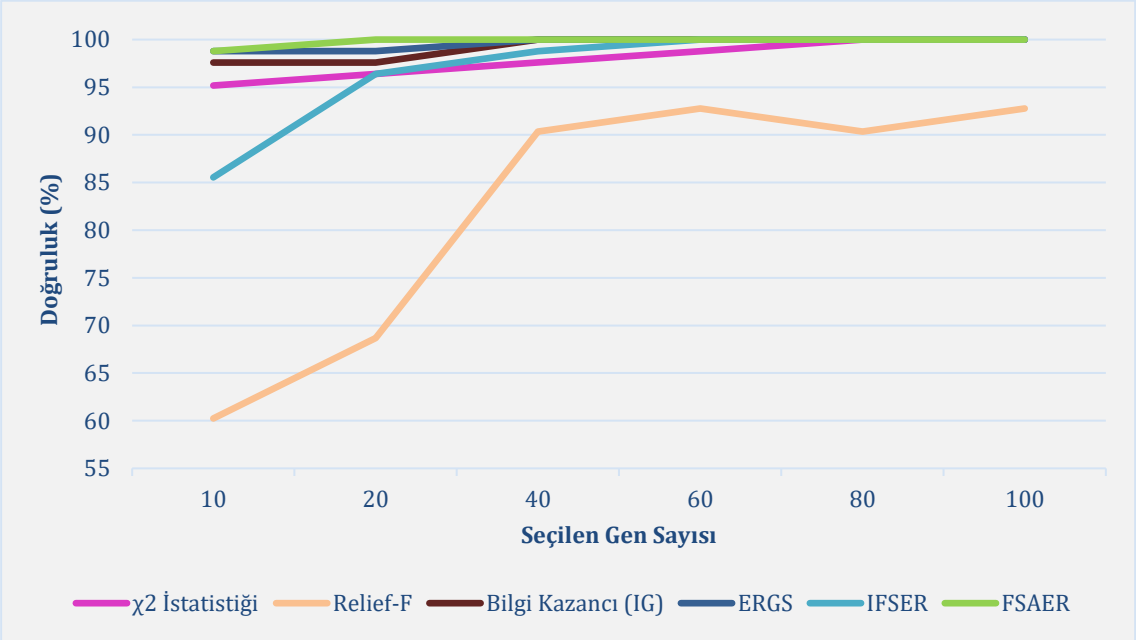
Şekil 5.3. Lösemi veri kümesi için NB sınıflandırma doğrulukları (%)

Prostat veri kümesi incelendiğinde, tablodaki en yüksek doğruluk oranı olan 95.10 (%) değerine FSAER yönteminin 10 ve 20 gen seçildiğinde ulaştığı görülmektedir. Özellik büyüklükleri arttıkça doğruluk oranlarında bir azalma olsa da FSAER için diğer durumlarda da sınıflandırma doğrulukları oldukça yüksektir. Bu veri kümesi için yöntemlerin sonuçları Şekil 5.4'te görülebilir.



Şekil 5.4. Prostat veri kümesi için NB sınıflandırma doğrulukları (%)

SRBCT veri kümesinde (4 sınıflı), tüm özellik alt küme durumlarına bakıldığında kaç gen seçildiği fark etmeksizin FSAER yönteminin en yüksek doğruluk oranlarına ulaştığı Şekil 5.5'te görülmektedir.

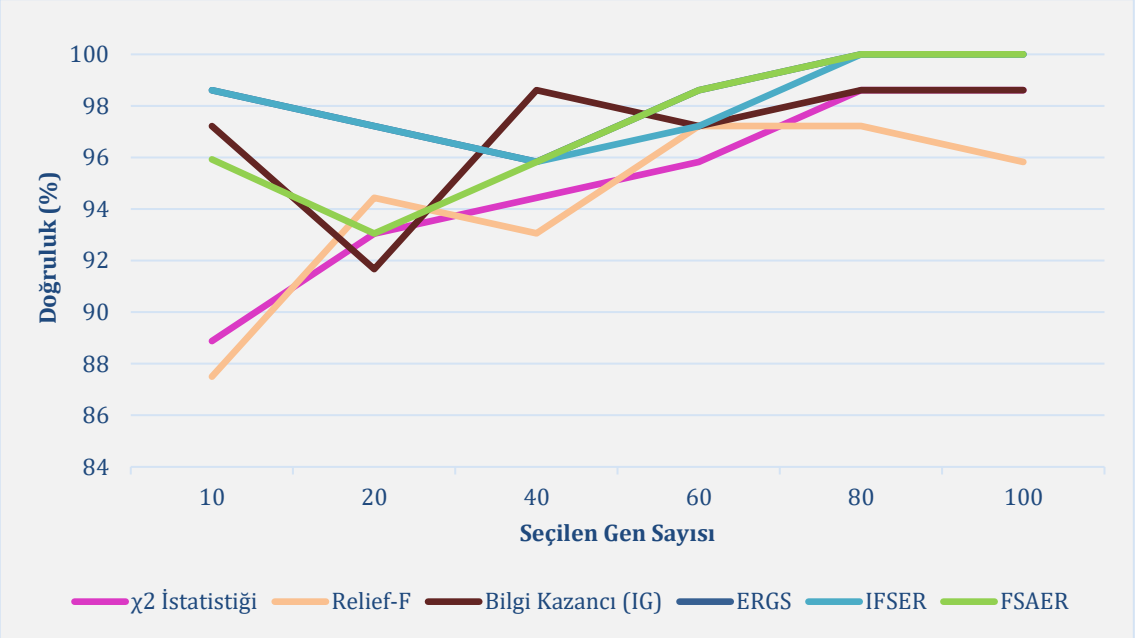


Şekil 5.5. SRBCT veri kümesi için NB sınıflandırma doğrulukları (%)

5.3.2. DVM ile Elde Edilen Sonuçlar

Çizelge 5.3, altı farklı gen açıklama veri kümesi için özellik seçme yöntemlerinden elde edilen farklı büyüklükteki özellik alt kümelerinin (10-100) DVM sınıflandırıcısı kullanılması sonucunda bulunan sınıflandırma doğruluklarını göstermektedir. Çizelge 5.3'e genel olarak bakıldığında tüm veri kümeleri için FSAER yönteminin en yüksek doğruluk oranlarına bazen tek başına bazen de diğer yöntemlerle birlikte ulaştığı görülmektedir.

Golub_1 veri kümesi için elde edilen sonuçlar incelendiğinde, en yüksek doğruluk oranı olan 100 (%) değerine FSAER yönteminin ERGS ve IFSER yöntemleriyle birlikte 80 ve 100 tane özellik seçildiğinde ulaştığı görülmektedir. Ayrıca seçilen özellik büyüklüğü 60 olduğu durum için FSAER ve ERGS yöntemleri en yüksek doğruluk oranını vermiştir. Golub_1 veri kümesine ait sonuçlar Şekil 5.6'da verilmiştir.

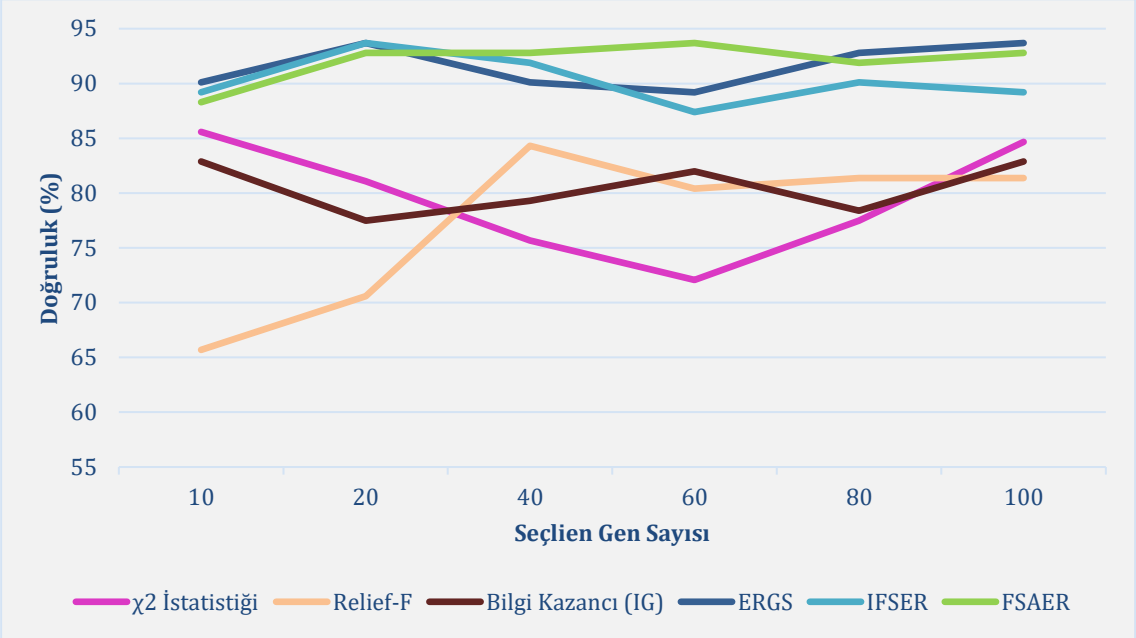


Şekil 5.6. Golub_1 veri kümesi için DVM sınıflandırma doğrulukları (%)

Golub_2 veri kümesi için elde edilen sonuçlar incelendiğinde, en yüksek doğruluk oranı olan 98.61(%) değerine FSAER ve ERGS yöntemlerinin 40 tane özellik seçildiği durumda ulaştığı görülmektedir.

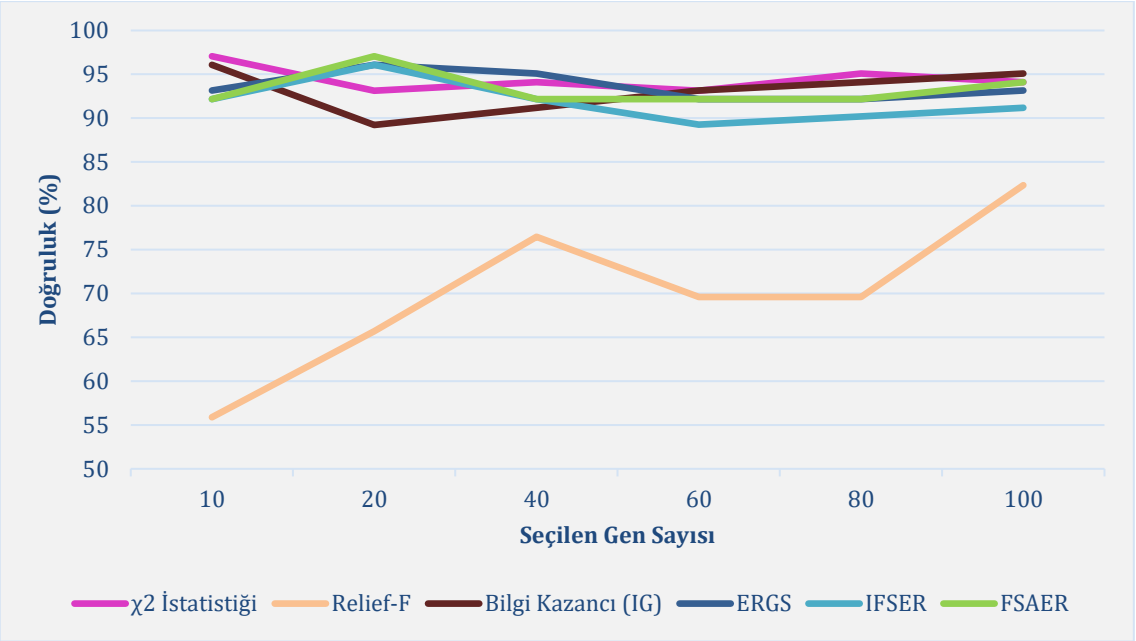
Kolon veri kümesi için en yüksek doğruluk oranı olan 85.48 (%) değerine FSAER ve ERGS yöntemlerinin 10 tane özellik seçildiği durumda ulaştığı görülmektedir. Ayrıca seçilen özellik büyüklüğü 40 olduğu durum için FSAER, ERGS ve IFSER yöntemleri en yüksek doğruluk oranını vermiştir.

Lösemi veri kümesinde, tablonun en yüksek doğruluk oranı olan 93.69 (%) değerini 20 özellik seçildiğinde ERGS ve IFSER yöntemleri verirken, 60 özellik seçildiğinde FSAER yöntemi en yüksek doğruluk oranına ulaşmıştır. Sonuçlar Şekil 5.7'de görülebilir.



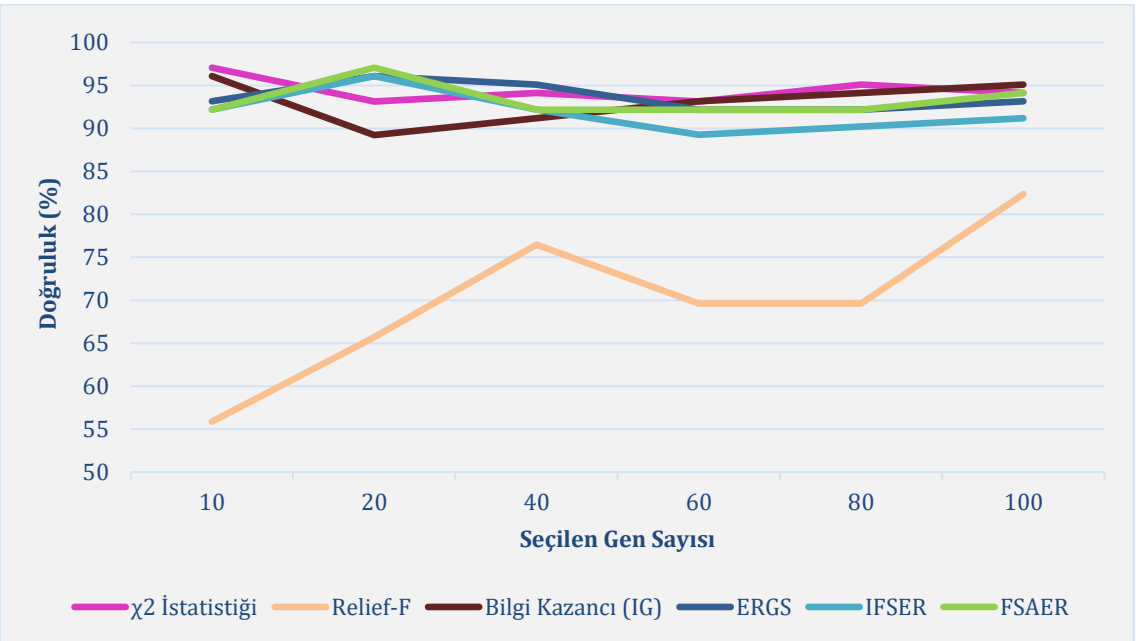
Şekil 5.7. Lösemi veri kümesi için DVM sınıflandırma doğrulukları (%)

Prostat veri kümesi incelendiğinde, Şekil 5.8'te görüldüğü gibi, tablodaki en yüksek doğruluk oranı olan 97.06 (%) değerine 10 özellik seçildiğinde χ^2 istatistiği yöntemi ulaşırken, 20 özellik seçildiğinde FSAER yöntemi ulaşmıştır.



Şekil 5.8. Prostat veri kümesi için DVM sınıflandırma doğrulukları (%)

SRBCT veri kümesinde (4 sınıflı) kullanılan yöntemlerden elde edilen sonuçlar Şekil 5.9'da vermiştir.

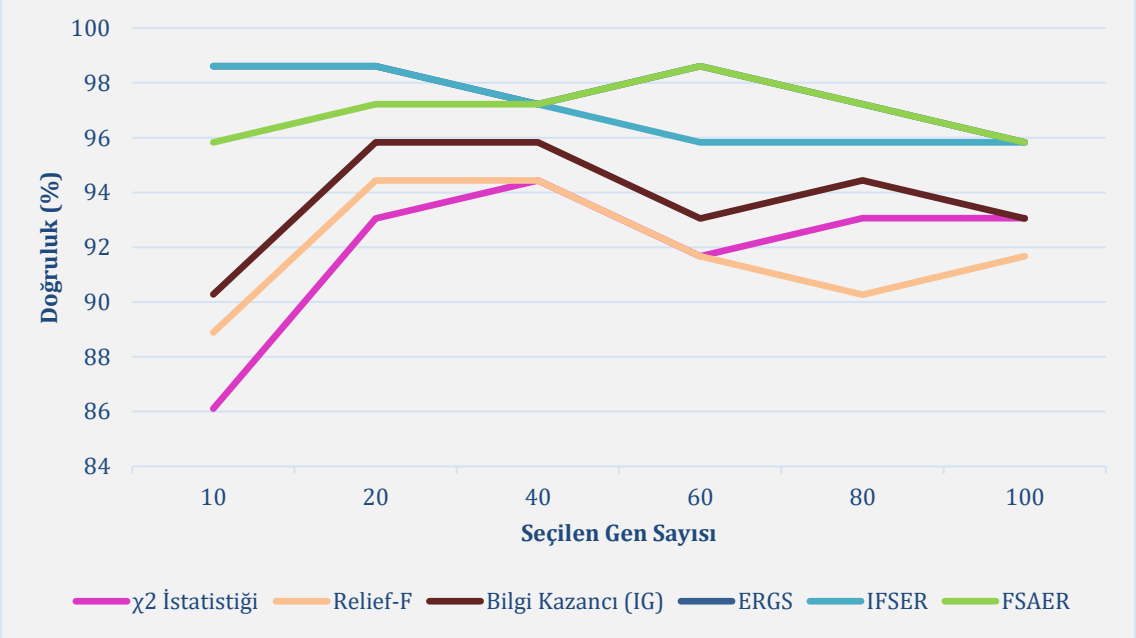


Şekil 5.9. SRBCT veri kümesi için DVM sınıflandırma doğrulukları (%)

5.3.3. kNN ile Elde Edilen Sonuçlar

Çizelge 5.4, altı farklı gen açıklama veri kümesi için özellik seçme yöntemlerinden elde edilen farklı büyüklükteki özellik alt kümelerinin (10-100) kNN sınıflandırıcısı kullanılması sonucunda bulunan sınıflandırma doğruluklarını göstermektedir. Çizelge 5.4'e genel olarak bakıldığında tüm veri kümeleri için FSAER yönteminin en yüksek doğruluk oranlarına bazen tek başına bazen de diğer yöntemlerle birlikte ulaştığı görülmektedir.

Golub_1 veri kümesi için elde edilen sonuçlar incelendiğinde (Şekil 5.10), en yüksek doğruluk oranı olan 98.61 (%) değerine 60 tane özellik seçildiğinde FSAER yönteminin ERGS yöntemiyle birlikte ulaştığı görülmektedir.

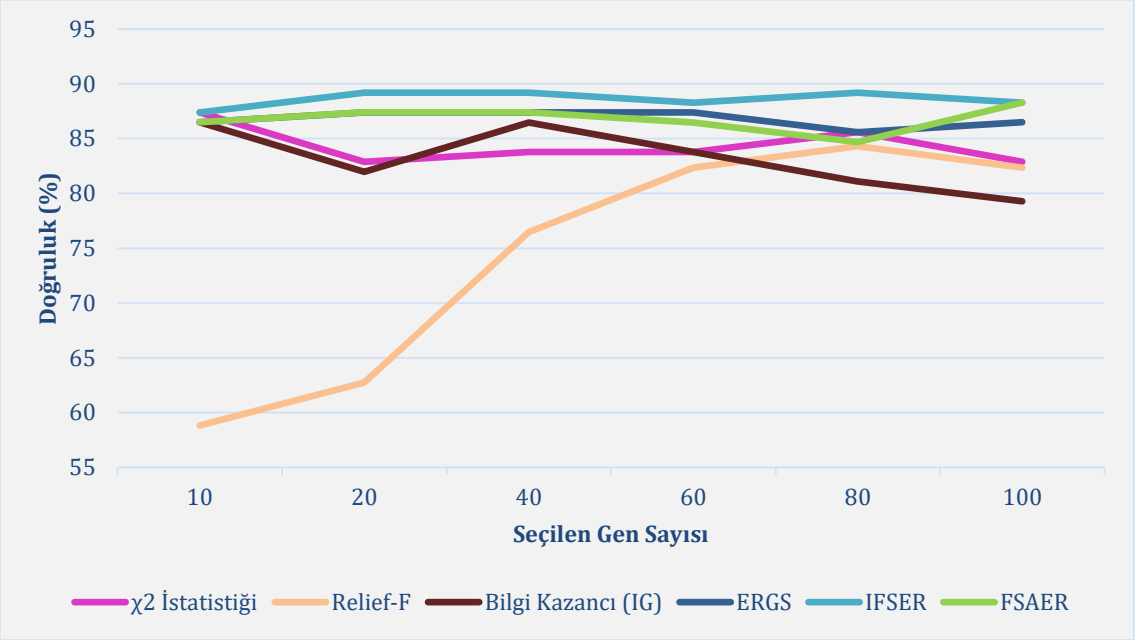


Şekil 5.10. Golub_1 veri kümesi için kNN sınıflandırma doğrulukları (%)

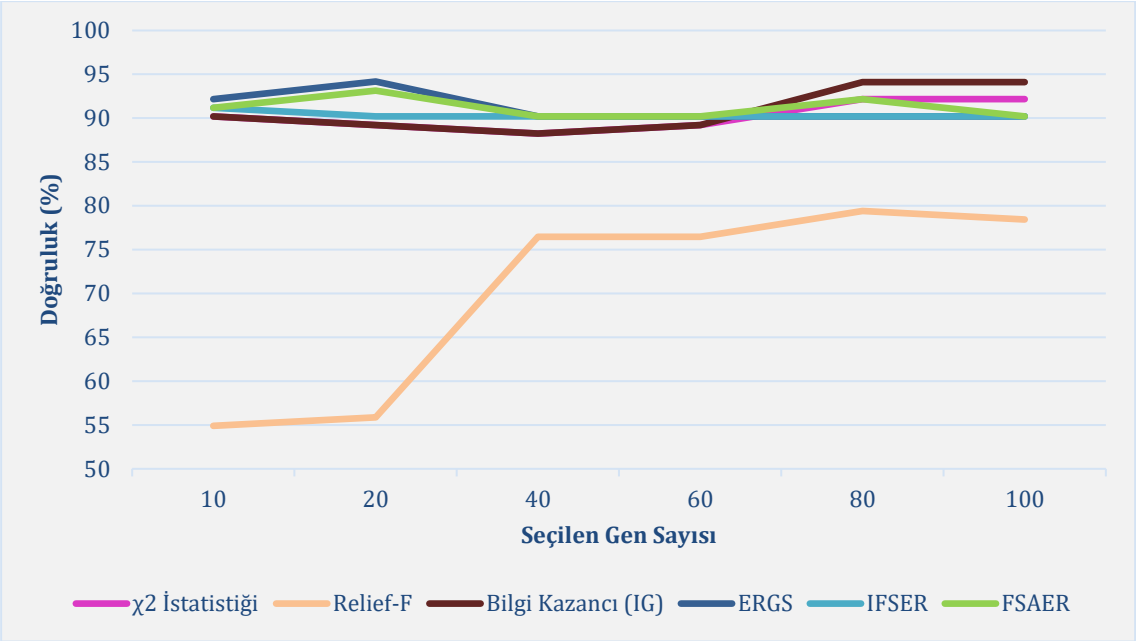
Golub_2 veri kümesi için elde edilen sonuçlar incelendiğinde, en yüksek doğruluk oranı olan 97.22 (%) değerine FSAER yönteminin 40 tane özellik seçildiğinde ulaştığı görülmektedir.

Kolon veri kümesi için en yüksek doğruluk oranı olan 88.70 (%) değerine FSAER ve ERGS yöntemlerinin 40 tane özellik seçildiği durumda ulaştığı görülmektedir. En iyi sonucu iki yöntem birlikte vermişlerdir.

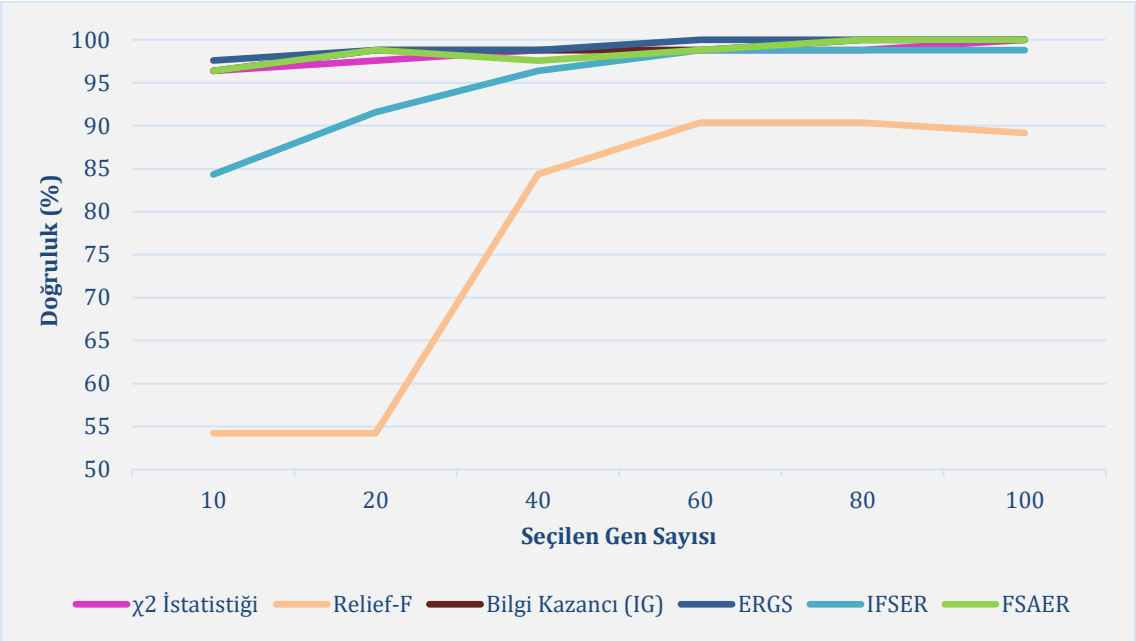
Lösemi, prostat ve SRBCT veri kümeleri için kullanılan yöntemlerin sonuçları, sırasıyla Şekil 5.11, Şekil 5.12 ve Şekil 5.13'te verilmiştir.



Şekil 5.11. Lösemi veri kümesi için kNN sınıflandırma doğrulukları (%)



Şekil 5.12. Prostat veri kümesi için kNN sınıflandırma doğrulukları (%)



Şekil 5.13. SRBCT veri kümesi için kNN sınıflandırma doğrulukları (%)

6. SONUÇ VE TARTIŞMA

Bu tez çalışmasında, FSAER adlı yeni bir istatistiksel özellik seçme yaklaşımı önerilmiştir. Literatüre daha önce katkıda bulunmuş ERGS ve IFSER adlı algoritmaların bir uzantısı olarak önerilen FSAER yöntemi, diğerlerinden farklı olarak verilerin dağılımında ayrık alan bulunmasını dikkate alırken aynı zamanda diğer iki yöntemin avantajlarını da sağlamaktadır.

Erişime açık altı farklı gen açıklama veri kümesine, bilinen beş farklı filtre yöntemi ve FSAER tarafından seçilen özellikler kullanılarak üç farklı sınıflandırma yöntemi (NB, SVM ve kNN) yardımıyla sınıflandırma doğrulukları elde edilmiştir. Tüm yöntemlerin uygulanması sonucu elde edilen doğru sınıflandırma oranları karşılaştırıldığında genel olarak oldukça yüksek doğruluk oranları görülse de aradaki küçük farklılıklar bile gen verileri söz konusu olduğunda hayati önem arz etmektedir. Bu nedenle FSAER yönteminin diğer yöntemlerden daha iyi sonuç verdiği durumlar çalışmanın önemini göstermektedir. Ayrıca uygulamada farklı büyüklükte seçilen gen alt kümeleri seçenekleri de (10, 20, 40, 60, 80, 100) gen seçimi yaparken alternatifler sunmaktadır.

Uygulamada kullanılan Golub_1 (72x7129, 2 sınıflı) ve Golub_2 (72x7129, 2 sınıflı) veri kümeleri aynı boyutlarda olan iki veri kümesidir ancak sınıf sayıları farklıdır. İki veri kümesinin sonuçları incelendiğinde FSAER yönteminin hem 2 sınıflı hem de 3 sınıflı veri kümesi için oldukça etkili sonuçlar verdiği görülmektedir. Ayrıca NB sınıflandırıcısı ile elde edilen sonuçlar incelendiğinde Şekil 5.5'te görüldüğü üzere SRBCT (83x2308, 4 sınıflı) gibi çok sınıflı bir veri kümesinde seçilen gen sayısına bakılmaksızın FSAER yönteminin en iyi sonuçlara ulaştığı görülmektedir. Bu da yöntemin hem iki sınıflı hem de çok sınıflı veri kümeleri için etkin çalıştığını göstermektedir.

Kullanılan veri kümeleri incelendiğinde Lösemi (111x12625, 2 sınıflı) ve Prostat (102x12600, 2 sınıflı) veri kümelerinin diğerlerine göre gen sayıları daha büyüktür. Şekil 5.3 ve Şekil 5.4'e bakıldığında ise (NB sınıflandırıcısı kullanıldığında) FSAER oldukça başarılı olmuştur. Bu durum bize önerilen FSAER yönteminin veri boyutu arttığında da etkin sonuçlar verdiğini göstermektedir.

SRBCT veri kümesi için Şekil 5.5 (NB), Şekil 5.9 (DVM) ve Şekil 5.13 birlikte incelendiğinde en iyi sonuçların Şekil 5.5'te yani NB sınıflandırıcısı kullanıldığında FSAER ile elde edildiği görülmektedir. Yani bu veri kümesi için en iyi sonuçlar NB sınıflandırıcısı kullanıldığında önerilen yöntem ile elde edilmiştir.

Prostat veri kümesi için Şekil 5.4 (NB), Şekil 5.8 (DVM) ve Şekil 5.12 birlikte incelendiğinde en iyi sonuçların Şekil 5.4'te yani NB sınıflandırıcısı kullanıldığında FSAER ile elde edildiği görülmektedir. Yani bu veri kümesi için de en iyi sonuçlar NB sınıflandırıcısı kullanıldığında önerilen yöntem ile elde edilmiştir.

Lösemi veri kümesi için FSAER yönteminin verdiği sonuçlara bakıldığında (Şekil 5.3, Şekil 5.7 ve Şekil 5.11) NB, DVM ve kNN sınıflandırma yöntemleri karşılaştırıldığında DVM ile daha iyi sonuçlara ulaşıldığı görülmektedir. Burada bahsi geçen farklı durumlardan dolayı çalışmanın tek bir sınıflandırıcı kullanılarak değil farklı alternatiflerle çalışılmış olması avantaj sağlamaktadır.

Çalışmanın sonucunda sınıflandırma doğruluklarının elde edilmesinin yanı sıra "seçilen özellik alt kümeleri" bilgisi de elde edilebilmektedir. Bu sayede, ilgili hastalıkların tanı ve tedavisinde ihtiyaç duyulan genlerin bilgisi sağlanabilmektedir.

Özellik seçim yöntemlerinin amacı, çok boyutlu veri kümelerinde sınıflandırma yapılmadan önce özelliklerin boyutunu azaltmaktır. Önerilen yöntem bunu özelliklere ağırlık atayarak yapar. Bu ağırlıklar Çebişev eşitsizliğine dayandığı için herhangi bir varsayım gerektirmemektedir. Bu sayede FSAER yönteminin sadece gen açıklama veri kümelerinde değil, aynı zamanda diğer birçok büyük boyutlu veri kümelerinde de uygulanabilir olduğu söylenebilir.

Üç temel başlıkta toplanan filtre, sarmal ve gömülü özellik seçim yöntemlerinin haricinde literatürde bu yöntemlerin farklı kombinasyonlarından oluşan hibrit yöntemler de

çalışılmaktadır. Bu çalışmada önerilen FSAER isimli filtre yönteminin daha sonraki çalışmalarda hibrit yöntemler için de kullanılması, bu sayede var olan filtre yöntemlerinin kullanıldığı hibrit yöntemlerden daha iyi sonuçlar elde edilebileceği düşünülmektedir. Bu durumda sadece sınıflandırma doğrulukları değil hesaplama zamanı gibi kriterler ile de karşılaştırmalar yapmak mümkün olacaktır.

7. KAYNAKLAR

Abend, K., & Harley, T. J. Comments "on the Mean Accuracy of Statistical Pattern Recognizers". IEEE Transactions on Information Theory, IT-15(3) **(1969)** 420–423.

Abusamra, H. A comparative study of feature selection and classification methods for gene expression data of glioma. Procedia Computer Science, 23 **(2013)** 5–14.

Ahmad, S. R., Bakar, A. A., & Yaakub, M. R. A review of feature selection techniques in sentiment analysis. Intelligent Data Analysis, 23(1) **(2019)** 159–189.

Al-Rajab, M., Lu, J., & Xu, Q. Examining applying high performance genetic data feature selection and classification algorithms for colon cancer diagnosis. Computer Methods and Programs in Biomedicine, 146 **(2017)** 11–24.

Algamal, Z. Y., Alhamzawi, R., & Mohammad Ali, H. T. Gene selection for microarray gene expression classification using Bayesian Lasso quantile regression. Computers in Biology and Medicine, 97 **(2018)** 145–152.

Alon, U., Barka, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A. J. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proceedings of the National Academy of Sciences of the United States of America, 96(12) **(1999)** 6745–6750.

Altman, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. American Statistician, 46(3) **(1992)** 175–185.

Altunkaynak, B. Veri Madenciliği Yöntemleri ve R Uygulamaları. Ankara: Seçkin Yayıncılık, **2017**.

Ang, J. C., Mirzal, A., Haron, H., & Hamed, H. N. A.. Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 13(5) **(2016)** 971–989.

Blum, A. L., & Langley, P. Selection of relevant features and examples in machine learning. Artificial Intelligence 97(1–2) **(1997)** 245–271.

Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A., Benítez, J. M., & Herrera, F. A review of microarray datasets and applied feature selection methods. Information Sciences, 282 **(2014)**111–135.

Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., & Lang, M. Benchmark for filter methods for feature selection in high-dimensional classification data. Computational Statistics and Data Analysis, 143 **(2020)** 106839.

Bonilla-Huerta, E., Hernández-Montiel, A., Morales-Caporal, R., & Arjona-López, M.. Hybrid Framework Using Multiple-Filters and an Embedded Approach for an Efficient Selection and Classification of Microarray Data. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 13(1) **(2016)** 12–26.

Buza, K. Classification of gene expression data: a hubness-aware semi-supervised approach. Computer methods and programs in biomedicine, 127 **(2016)** 105–113.

Cai, R., Hao, Z., Yang, X., & Wen, W. An efficient gene selection algorithm based on mutual information. Neurocomputing, 72(4–6) **(2009)** 991–999.

Canul-Reich, J., Hall, L. O., Goldgof, D. B., Korecki, J. N., & Eschrich, S. Iterative feature perturbation as a gene selector for microarray data. International Journal of Pattern Recognition and Artificial Intelligence, 26(5) **(2012)** 1260003.

Chandra, B., & Gupta, M. An efficient statistical feature selection approach for classification of gene expression data. Journal of Biomedical Informatics, 44(4) **(2011)** 529–535.

Chen, X. W. Gene selection for cancer classification using bootstrapped genetic algorithms and support vector machines. Proceedings of the 2003 IEEE Bioinformatics Conference, CSB 2003, 46(1–3) **(2003)** 504–505.

Chiaretti, S., Li, X., Gentleman, R., Vitale, A., Vignetti, M., Mandelli, F., … Foa, R. Gene expression profile of adult T-cell acute lymphocytic leukemia identifies distinct subsets of patients with different response to therapy and survival. Blood, 103(7) **(2004)** 2771–2778.

Coomans, D., & Massart, D. L. Alternative k-nearest neighbour rules in supervised pattern recognition. Part 3. Condensed nearest neighbour rules. Analytica Chimica Acta, 138(C) **(1982)** 167–176.

Cortes, C., & Vapnik, V. Support-vector networks. Machine learning, 20(3) **(1995)** 273–297.

Das, S., Rai, A., Mishra, D. C., & Rai, S. N. Statistical approach for selection of biologically informative genes. Gene, 655 **(2018)** 71–83.

Dash, S., & Patra, B. N. Reliability analysis of Classification of Gene Expression Data. Proceedings of the International Conference on Bioinformatics & Computational Biology (BIOCOMP), **2011**, s. 1.

Ding, C., & Peng, H. Minimum redundancy feature selection from microarray gene expression data. Proceedings of the 2003 IEEE Bioinformatics Conference, CSB 2003, 3(02) **(2003)** 523–528.

Domingos, P., & Pazzani, M. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. Machine Learning, 29(2–3) **(1997)** 103–130.

Duan, K. B., Rajapakse, J. C., Wang, H., & Azuaje, F. Multiple SVM-RFE for gene selection in cancer classification with expression data. IEEE Transactions on Nanobioscience, 4(3) **(2005)** 228–233.

Friedman, N., Geiger, D., & Goldszmidt, M.. Bayesian Network Classifiers. Machine Learning, 29(2–3) **(1997)** 131–163.

Furey, T. S., Cristianini, N., Duffy, N., Bednarski, D. W., Schummer, M., & Haussler, D. Support vector machine classification and validation of cancer tissue samples using microarray expression data. Bioinformatics, 16(10) **(2000)** 906–914.

Gheyas, I. A., & Smith, L. S. Feature subset selection in large dimensionality domains. Pattern Recognition, 43(1) **(2010)** 5–13.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., … Lander, E. S. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science, 286(5439) **(1999)** 531–527.

Guo, S., Guo, D., Chen, L., & Jiang, Q. A L1-regularized feature selection method for local dimension reduction on microarray data. Computational Biology and Chemistry, 67 **(2017)** 92–101.

Han, J., Kamber, M., & Pei, J. Data Mining : Concepts and Techniques : Concepts and Techniques (3rd Edition). Data Mining. Elsevier, **2012**.

Hand, D. J. Principles of data mining. MIT Press **2007**.

Harrington, P. Machine Learning in Action. Machine Learning (C. 37). Manning Publications Co. **2012**.

Inza, I., Sierra, B., Blanco, R., & Larrañaga, P. Gene selection by sequential search wrapper approaches in microarray cancer class prediction. Journal of Intelligent and Fuzzy Systems, 12(1 SPEC.) **(2002)** 25–33.

Izenman, A. J. Modern Multivariate Statistical Techniques. Artificial Neural Networks, 10 **(2008)** 101–118.

Jiang, N., Wu, W. X., & Mitchell, I. Protein fold recognition using neural networks and support vector machines. Lecture Notes in Computer Science, 3578(4) **(2005)** 462–469.

Jin, X., Xu, A., Bie, R., & Guo, P. Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), **2006**, ss. 106–115.

Jirapech-Umpai, T., & Aitken, S. Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. BMC Bioinformatics, 6(1) **(2005)** 148.

Kang, C., Huo, Y., Xin, L., Tian, B., & Yu, B. Feature selection and tumor classification for microarray data using relaxed Lasso and generalized multi-class support vector machine. Journal of Theoretical Biology, 463 **(2019)** 77–91.

Khan, J., Wei, J. S., Ringnér, M., Saal, L. H., Ladanyi, M., Westermann, F., … Meltzer, P. S. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nature Medicine, 7(6) **(2001)** 673–679.

Kira, K., & Rendell, L. A. A Practical Approach to Feature Selection. Machine Learning Proceedings, **1992**, ss. 249–256.

Kuo, L., Yu, F., & Zhao, Y. Statistical Methods for Identifying Differentially Expressed Genes in Replicated Microarray Experiments: A Review. Statistical Advances in the Biomedical Sciences: Clinical Trials, Epidemiology, Survival Analysis, and Bioinformatics, **(2007)** 341–363.

Lai, C. M., Yeh, W. C., & Chang, C. Y. Gene selection using information gain and improved simplified swarm optimization. Neurocomputing, 218 **(2016)** 331–338.

Lee, C. P., Lin, W. S., Chen, Y. M., & Kuo, B. J. Gene selection and sample classification on microarray data based on adaptive genetic algorithm/k-nearest neighbor method. Expert Systems with Applications, 38(5) **(2011)** 4661–4667.

Lee, Y., & Lee, C. K. Classification of multiple cancer types by multicategory support vector machines using gene expression data. Bioinformatics, 19(9) **(2003)** 1132–1139.

Li, H., Qi, F., & Wang, S. A comparison of model selection methods for multi-class support vector machines. Lecture Notes in Computer Science, 3483(IV) **(2005)** 1140–1148.

Li, Z., Xie, W., & Liu, T. Efficient feature selection and classification for microarray data. PLoS ONE, 13(8) **(2018).**

Liang, F., Li, Q., & Zhou, L. Bayesian Neural Networks for Selection of Drug Sensitive Genes. Journal of the American Statistical Association, 113(523) **(2018)** 955–972.

Liao, J. C., Boscolo, R., Yang, Y. L., Tran, L. M., Sabatti, C., & Roychowdhury, V. P. Network component analysis: Reconstruction of regulatory signals in biological systems. Proceedings of the National Academy of Sciences of the United States of America, 100(26) **(2003)** 15522–15527.

Liu, H. Feature Engineering for Machine Learning and Data Analytics. Feature Engineering for Machine Learning and Data Analytics. CRC Press, **2018**.

Liu, H., & Motoda, H. Computational Methods of Feature Selection. Computational Methods of Feature Selection. CRC Press, **2007**.

Liu, Huiqing, Li, J., & Wong, L. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. Genome informatics. International Conference on Genome Informatics, 13 **(2002)** 51–60.

Liu, S., Xu, C., Zhang, Y., Liu, J., Yu, B., Liu, X., & Dehmer, M. Feature selection of gene expression data for Cancer classification using double RBF-kernels. BMC Bioinformatics, 19(1) **(2018)** 52–57.

Long, A. D., Mangalam, H. J., Chan, B. Y. P., Tolleri, L., Hatfield, G. W., & Baldi, P. Improved statistical inference from DNA microarray data using analysis of variance and A Bayesian statistical framework. Analysis of global gene expression in Escherichia coli

K12. Journal of Biological Chemistry, 276(23) **(2001)** 19937–19944.

Maldonado, S., Weber, R., & Basak, J. Simultaneous feature selection and classification using kernel-penalized support vector machines. Information Sciences, 181(1) **(2011)** 115–128.

Nguyen, T., & Nahavandi, S. Modified AHP for Gene Selection and Cancer Classification Using Type-2 Fuzzy Logic. IEEE Transactions on Fuzzy Systems, 24(2) **(2016)** 273–287.

Peng, H., Long, F., & Ding, C. Feature selection based on mutual information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(8) **(2005)** 1226–1238.

Piatetsky-Shapiro, G., & Tamayo, P. Microarray data mining: facing the challenges. ACM SIGKDD Explorations Newsletter, 5(2) **(2003)** 1–5.

Quinlan, J. R. Induction of Decision Trees. Machine Learning, 1(1) **(1986)** 81–106.

Radovic, M., Ghalwash, M., Filipovic, N., & Obradovic, Z. Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. BMC Bioinformatics, 18(1) **(2017)** 9.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C. H., Angelo, M., … Golub, T. R. Multiclass cancer diagnosis using tumor gene expression signatures. Proceedings of the National Academy of Sciences of the United States of America, 98(26) **(2001)** 15149–15154.

Raniszewski, M. Sequential reduction algorithm for nearest neighbor rule. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), **2010**, ss. 219–226).

Remeseiro, B., & Bolon-Canedo, V. A review of feature selection methods in medical applications. Computers in Biology and Medicine, 112 **(2019)** 103375. https://doi.org/10.1016/j.compbiomed.2019.103375

Rish, I. IBM Research Report An empirical study of the naive Bayes classifier. Science **(2001)** 41–46).

Robnik-Šikonja, M., & Kononenko, I. Theoretical and Empirical Analysis of ReliefF and RReliefF. Machine Learning, 53(1–2) **(2003)** 23–69.

Sánchez-Maroño, N., Alonso-Betanzos, A., & Tombilla-Sanromán, M. Filter methods for feature selection - A comparative study. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), **2007**, ss. 178–187.

Sánchez, J. S., Mollineda, R. A., & Sotoca, J. M. An analysis of how training data complexity affects the nearest neighbor classifiers. Pattern Analysis and Applications, 10(3) **(2007)** 189–201.

Scott, D. W. The Curse of Dimensionality and Dimension Reduction. Multivariate Density Estimation: Theory, Practice, and Visualization, **(2015)** 217–240.

Shukla, A. K., Singh, P., & Vardhan, M. A hybrid gene selection method for microarray recognition. Biocybernetics and Biomedical Engineering, 38(4) **(2018)** 975–991.

Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., … Sellers, W. R. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell, 1(2) **(2002)** 203–209.

Song, H., Zhang, X., Shi, C., Wang, S., Wu, A., & Wei, C. Selection and verification of candidate reference genes for mature microRNA expression by quantitative RT-PCR in the tea plant (Camellia sinensis). Genes, 7(6) **(2016)** 25.

Sun, Y., Lu, C., & Li, X. The cross-entropy based multi-filter ensemble method for gene selection. Genes, 9(5) **(2018)** 258.

Tang, C., Cao, L., Zheng, X., & Wang, M. Gene selection for microarray data classification via subspace learning and manifold regularization. Medical and Biological Engineering and Computing, 56(7) **(2018)** 1271–1284.

Tang, J., Alelyani, S., & Liu, H. Feature selection for classification: A review. Data Classification: Algorithms and Applications **(2014)** 37–64.

Thomas, J. G., Olson, J. M., Tapscott, S. J., & Zhao, P.. An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles. Genome Research, 11(7) **(2001)** 1227–1236.

Wang, J., Zhou, S., Yi, Y., & Kong, J. An improved feature selection based on effective range for classification. The Scientific World Journal, 2014 **(2014)**.

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

Wong, T. T. Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. Pattern Recognition, 48(9) **(2015)** 2839–2846.

Xiong, M., Fang, X., & Zhao, J. (2001). Biomarker identification by feature wrappers. Genome Research, 11(11), 1878–1887. https://doi.org/10.1101/gr.190001

Yang, C. H., Chuang, L. Y., Li, J. C., & Yang, C. H. A hybrid BPSO-CGA approach for gene selection and classification of microarray data. Journal of Computational Biology, 19(1) **(2012)** 68–82.

Zaki, M. J., & Meira, M. J. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press, **2013**.

Zeebaree, D. Q., Haron, H., & Abdulazeez, A. M. Gene Selection and Classification of Microarray Data Using Convolutional Neural Network. ICOASE 2018 - International

Conference on Advanced Science and Engineering, **2018**, ss 145–150.

Zhang, S., Wang, J., Ghoshal, T., Wilkins, D., Mo, Y. Y., Chen, Y., & Zhou, Y. lncRNA gene signatures for prediction of breast cancer intrinsic subtypes and prognosis. Genes, 9(2) **(2018)** 65.

Zhang, Y., Ding, C., & Li, T. Gene selection algorithm by combining reliefF and mRMR. BMC Genomics, 9(SUPPL. 2) **(2008)** 27.

Zheng, X., Zhu, W., Tang, C., & Wang, M. Gene selection for microarray data classification via adaptive hypergraph embedded dictionary learning. Gene, 706 **(2019)** 188–200.

Zhou, X., & Mao, K. Z. LS Bound based gene selection for DNA microarray data. Bioinformatics, 21(8) **(2005)** 1559–1564.

Zhou, X., & Tuck, D. P. Erratum: MSVM-RFE: Extensions of SVM-RFE for multiclass gene selection on DNA microarray data (Bioinformatics (2007) vol. 23 (9) (1106-1114)). Bioinformatics, 23(15) **(2007)** 2029.

Zhu, Z., Ong, Y. S., & Dash, M. Wrapper-filter feature selection algorithm using a memetic framework. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 37(1) **(2007)** 70–76.

Web References

https://www.rdocumentation.org/packages/datamicroarray/versions/0.2.3

(Accessed on 20.10.2019)

http://csse.szu.edu.cn/staff/zhuzx/Datasets.html

(Accessed on 20.10.2019)

https://cran.r-project.org/web/packages/plsgenomics/plsgenomics.pdf

(Accessed on 20.10.2019)

EKLER

EK 1 – Uygulamada kullanılan R Kodları

```
                              ----ERGS----
ergs <- function(ClassName, data, threshold=1, nsf=10,na.rm =
TRUE,verbose=TRUE){

  dp=as.character(ClassName)

  if (na.rm){
    completeObs <- complete.cases(data)
    data <- data[completeObs,]
  }

  if (any(colnames(data)==dp[[2L]])==FALSE) stop("The name of class
variable does not match the variable names in the data. The class
variable must be one factor.")
  if (!is.factor(data[, dp[[2L]]])) stop("The class variable must be a
factor.")
  data<-cbind(data[,! names(data) %in% dp[[2L]]],data[,dp[[2L]]])
  colnames(data)[NCOL(data)]=dp[[2L]]
  group=data[, dp[[2L]]]
  y = data[,! names(data) %in% dp[[2L]]]
  NO<-NROW(data)                                    # Number of
conditions
  NG<-NCOL(data)-1                                  # Number of
features
  NC<-NROW(levels(factor(group)))                   # Number of
classes
  databac<-data
  c.n <- NULL
  CC=1.732                                          # Chebychev
Coefficient
  SRANGE<-list()

  c.levels <- levels(factor(group))

    for (i in c.levels) {
    c.n[i] <- sum(group==i)
    }

  PC=c.n/sum(c.n)
  RL<-matrix(0,NC,NG)
  RU<-matrix(0,NC,NG)
  MEAN<-aggregate(y, list(group), mean)
  SD<-aggregate(y, list(group), sd)


  for (i in 1:NC){ # Lower and Upper bound for each class
    for (j in 1:NG) { # Lower and Upper bound for each genes
```

```
      RL[i,j]<-MEAN[i,j+1]-(1-PC[i])*CC*SD[i,j+1]
      RU[i,j]<-MEAN[i,j+1]+(1-PC[i])*CC*SD[i,j+1]
      }}
  AC<-c()
  s1<-0
  for (i in 1:NG){
    AC[i]=0
    RANGE<-cbind(RL[,i],RU[,i])
    SRANGE[[names(data)[i]]]<-RANGE
    RANGE<-RANGE[order(RANGE[,1]),]
    for (j in 1:(NC-1)){
      for (k in (j+1):NC){
        if (RANGE[k,1]<RANGE[j,2])
        {s1=s1+1
          AC[i]<-AC[i]+(RANGE[j,2]-RANGE[k,1])/(max(RANGE)-
min(RANGE))}
      }}}
  W<-c()
  for (i in 1:NG){
    W[i]<-1-AC[i]/max(AC)}
  W<-t(W)
  W<-data.frame(W)
  W[,NG+1]<-NA
  databac[NO+1,]<-W
  DataOrder<-databac[order(databac[NO+1,],decreasing=TRUE)]
  OrderedData<-DataOrder[,! names(DataOrder) %in% dp[[2L]]]
  SelectedFeatures<-OrderedData[1:NO,1:nsf] # Selecting features
  # SelectedFeatures<-subset(OrderedData,select =
OrderedData[NO+1,]>=threshold) # Selecting features
  NewData<-as.data.frame(c(SelectedFeatures[1:NO,],data[dp[[2L]]]))

  if (verbose) {
    cat("------------------------------------------------------------
","\n", sep = " ")
    cat("  Effective Range based Gene Selection (ERGS) Algorithm",
"\n\n",sep = " ")
    cat("  Threshold =",threshold, "\n", sep = " ")
    cat("  Number of selected features =",dim(SelectedFeatures)[2],
"\n", sep = " ")
    cat("  Names of selected features: \n", sep = " ")
    cat(" ",colnames(SelectedFeatures), "\n", sep = " ")
    cat("------------------------------------------------------------
","\n\n", sep = " ")
  }

  result <- list()
  result$selectedData <- NewData
  result$selectedFNames <- colnames(SelectedFeatures)
  result$weight <- W
  result$ranges <- SRANGE
  result$RankedData <- OrderedData[1:NO,]
  attr(result, "class") <- "owt"
  invisible(result)
}
```

----IFSER----

```
ifser <- function(ClassName, data, threshold=1, nsf=10,na.rm =
TRUE,verbose=TRUE){

  dp=as.character(ClassName)

  if (na.rm){
    completeObs <- complete.cases(data)
    data <- data[completeObs,]
  }

  if (any(colnames(data)==dp[[2L]])==FALSE) stop("The name of class
variable does not match the variable names in the data. The class
variable must be one factor.")
  if (!is.factor(data[, dp[[2L]]])) stop("The class variable must be a
factor.")
  data<-cbind(data[,! names(data) %in% dp[[2L]]],data[,dp[[2L]]])
  colnames(data)[NCOL(data)]=dp[[2L]]
  group=data[, dp[[2L]]]
  y = data[,! names(data) %in% dp[[2L]]]
  NO<-NROW(data)                          # Number of conditions
  NG<-NCOL(data)-1                        # Number of features
  NC<-NROW(levels(factor(group)))         # Number of classes
  databac<-data
  c.n <- NULL
  CC=1.732                                # Chebychev Coefficient
  SRANGE<-list()

  c.levels <- levels(factor(group))

    for (i in c.levels) {
    c.n[i] <- sum(group==i)
    }

  a=1
  A<-list()
  for (i in c.levels)
  {A[[a]] <- subset(data,group==i)
  a=a+1}

  PC=c.n/sum(c.n)
  RL<-matrix(0,NC,NG)
  RU<-matrix(0,NC,NG)
  H<-matrix(0,NG,NC)
  G<-matrix(0,NG,NC)
  MEAN<-aggregate(y, list(group), mean)
  SD<-aggregate(y, list(group), sd)


  for (i in 1:NC){ # Lower and Upper bound for each class
    for (j in 1:NG) { # Lower and Upper bound for each genes
      RL[i,j]<-MEAN[i,j+1]-(1-PC[i])*CC*SD[i,j+1]
      RU[i,j]<-MEAN[i,j+1]+(1-PC[i])*CC*SD[i,j+1]
      }}
```

```
  AC<-c()
  s1<-0
  for (i in 1:NG){
    AC[i]=0
    RANGE<-cbind(RL[,i],RU[,i])
    SRANGE[[names(data)[i]]]<-RANGE
    RANGE<-RANGE[order(RANGE[,1]),]
    for (j in 1:(NC-1)){
      for (k in (j+1):NC){
        if (RANGE[k,1]<RANGE[j,2]) # Overlap
        {
          AC[i]<-AC[i]+(RANGE[j,2]-RANGE[k,1])/(max(RANGE)-min(RANGE))
          H[i,j]<-H[i,j]+sum(A[[j]][i]>=max(RANGE[j,1],RANGE[k,1]) &
A[[j]][i]<=min(RANGE[j,2],RANGE[k,2]))
          H[i,k]<-H[i,k]+sum(A[[k]][i]>=max(RANGE[j,1],RANGE[k,1]) &
A[[k]][i]<=min(RANGE[j,2],RANGE[k,2]))
        }

        if (RANGE[k,2]<RANGE[j,2]) # Including
        {
          AC[i]<-AC[i]+(RANGE[k,2]-RANGE[k,1])/(max(RANGE)-min(RANGE))
          G[i,j]<-G[i,j]+sum(A[[j]][i]>=RANGE[k,1] &
A[[j]][i]<=RANGE[k,2])
          G[i,k]<-G[i,k]+sum(A[[k]][i]>=RANGE[k,1] &
A[[k]][i]<=RANGE[k,2])
        }
}}}

  NAC<-c()
  for (i in 1:NG){
    NAC[i]<-1-AC[i]/max(AC)}

  H<-colSums(t(H)/c.n)
  SNH<-1-H/max(H)

  G<-colSums(t(G)/c.n)
  SNG<-1-G/max(G)

  W<-NAC*(SNH+SNG)

  W<-t(W)
  W<-data.frame(W)
  W[,NG+1]<-NA
  databac[NO+1,]<-W
  DataOrder<-databac[order(databac[NO+1,],decreasing=TRUE)]
  OrderedData<-DataOrder[,! names(DataOrder) %in% dp[[2L]]]
  SelectedFeatures<-OrderedData[1:NO,1:nsf] # Selecting features
  # SelectedFeatures<-subset(OrderedData,select =
OrderedData[NO+1,]>=threshold) # Selecting features
  NewData<-as.data.frame(c(SelectedFeatures[1:NO,],data[dp[[2L]]]))

  if (verbose) {
    cat("------------------------------------------------------
","\n", sep = " ")
```

```
    cat("  Improved Feature Selection Algorithm based on Effective
Range (IFSER)", "\n\n",sep = " ")
    cat("  Threshold =",threshold, "\n", sep = " ")
    cat("  Number of selected features =",dim(SelectedFeatures)[2],
"\n", sep = " ")
    cat("  Names of selected features: \n", sep = " ")
    cat(" ",colnames(SelectedFeatures), "\n", sep = " ")
    cat("------------------------------------------------------------
","\n\n", sep = " ")
  }

  result <- list()
  result$selectedData <- NewData
  result$selectedFNames <- colnames(SelectedFeatures)
  result$weight <- W
  result$ranges <- SRANGE
  result$RankedData <- OrderedData[1:NO,]
  attr(result, "class") <- "owt"
  invisible(result)
}
```

----FSAER----

```
fsaer <- function(ClassName, data, threshold=1, nsf=10,na.rm =
TRUE,verbose=TRUE){

  dp=as.character(ClassName)

  if (na.rm){
    completeObs <- complete.cases(data)
    data <- data[completeObs,]
  }

  if (any(colnames(data)==dp[[2L]])==FALSE) stop("The name of class
variable does not match the variable names in the data. The class
variable must be one factor.")
  if (!is.factor(data[, dp[[2L]]])) stop("The class variable must be a
factor.")
  data<-cbind(data[,! names(data) %in% dp[[2L]]],data[,dp[[2L]]])
  colnames(data)[NCOL(data)]=dp[[2L]]
  group=data[, dp[[2L]]]
  y = data[,! names(data) %in% dp[[2L]]]
  NO<-NROW(data)                                    # Number of
conditions
  NG<-NCOL(data)-1                                  # Number of
features
  NC<-NROW(levels(factor(group)))                   # Number of
classes
  databac<-data
  c.n <- NULL
  CC=1.732                                          # Chebychev
Coefficient
  SRANGE<-list()

  c.levels <- levels(factor(group))

    for (i in c.levels) {
    c.n[i] <- sum(group==i)
    }

  PC=c.n/sum(c.n)
  RL<-matrix(0,NC,NG)
  RU<-matrix(0,NC,NG)
  MEAN<-aggregate(y, list(group), mean)
  SD<-aggregate(y, list(group), sd)

  enb=0;enk=100000000
  for (i in 1:NC){
    for (j in 1:NG) {
      RL[i,j]<-MEAN[i,j+1]-(1-PC[i])*CC*SD[i,j+1]
      RU[i,j]<-MEAN[i,j+1]+(1-PC[i])*CC*SD[i,j+1]
      }}
  AC<-c()
  s1<-s2<-s3<-0
  for (i in 1:NG){
    AC[i]=0
```

```
    RANGE<-cbind(RL[,i],RU[,i])
    SRANGE[[names(data)[i]]]<-RANGE
    RANGE<-RANGE[order(RANGE[,1]),]
    for (j in 1:(NC-1)){
      for (k in (j+1):NC){

        if (RANGE[k,1]<RANGE[j,2])   # Overlap
        {s1=s1+1
          AC[i]<-AC[i]+(RANGE[j,2]-RANGE[k,1])/(max(RANGE)-
min(RANGE))}

        if (RANGE[k,2]<RANGE[j,2])       # Including
        {s2=s2+1
          AC[i]<-AC[i]+(RANGE[k,2]-RANGE[k,1])/(max(RANGE)-
min(RANGE))}

        if (RANGE[k,1]>RANGE[j,2])   #Disjoint
        {s3=s3+1
          #if ((RANGE[k,1]-RANGE[j,2])>enb) {enb=RANGE[k,1]-
RANGE[j,2]; adres1=i}
          #if ((RANGE[k,1]-RANGE[j,2])<enk) {enk=RANGE[k,1]-
RANGE[j,2]; adres2=i}
          AC[i]<-AC[i]-(RANGE[k,1]-RANGE[j,2])/(max(RANGE)-
min(RANGE))}
    }}}
  W<-c()
  for (i in 1:NG){
    W[i]<-1-AC[i]/max(AC)}
  W<-t(W)
  W<-data.frame(W)
  W[,NG+1]<-NA
  databac[NO+1,]<-W
  DataOrder<-databac[order(databac[NO+1,],decreasing=TRUE)]
  OrderedData<-DataOrder[,! names(DataOrder) %in% dp[[2L]]]
  SelectedFeatures<-OrderedData[1:NO,1:nsf] # Selecting features
  # SelectedFeatures<-subset(OrderedData,select =
OrderedData[NO+1,]>=threshold) # Selecting features
  NewData<-as.data.frame(c(SelectedFeatures[1:NO,],data[dp[[2L]]]))


  if (verbose) {
    cat("------------------------------------------------------------
","\n", sep = " ")
    cat("  Feature Selection Algorithm based on Effective Ranges
(FSAER)", "\n\n",sep = " ")
    cat("  Threshold =",threshold, "\n", sep = " ")
    cat("  Number of selected features =",dim(SelectedFeatures)[2],
"\n", sep = " ")
    cat("  Names of selected features: \n", sep = " ")
    cat(" ",colnames(SelectedFeatures), "\n", sep = " ")
    cat("------------------------------------------------------------
","\n\n", sep = " ")
  }
```

```
  result <- list()
  result$selectedData <- NewData
  result$selectedFNames <- colnames(SelectedFeatures)
  result$weight <- W
  result$ranges <- SRANGE
  result$RankedData <- OrderedData[1:NO,]
  attr(result, "class") <- "owt"
  invisible(result)
}
```

----Uygulama adımındaki farklı senaryolara bir örnek----

```
# Results
# library("datamicroarray")

data("golub")
data<-as.data.frame(cbind(golub$y,golub$x))
names(data)[1]<-paste("V1")
data$V1<-as.factor(data$V1)
dp=as.character(V1~.)
SF<-c(10,20,40,60,80,100)

train_control <- trainControl(method="LOOCV")

res1<-ergs(V1~., data, threshold=1, nsf=10,na.rm = TRUE,verbose=TRUE)

cat("  Effective Range based Gene Selection (ERGS) Algorithm",
"\n",sep = " ")
for (i in 1:6)
  {
    NewData<-as.data.frame(c(res1$RankedData[1:SF[i]],data[dp[[2L]]]))
    model1 <- train(V1~., NewData, trControl=train_control,
method="nb")
    cat("  Selected Feature =",SF[i],"Accuracy
=",max(model1$results$Accuracy),"\n", sep = " ")
  }

res2<-ifser(V1~., data, threshold=1, nsf=10,na.rm = TRUE,verbose=TRUE)

cat("  Improved Feature Selection Algorithm based on Effective Range
(IFSER)", "\n",sep = " ")
for (i in 1:6)
{
  NewData<-as.data.frame(c(res2$RankedData[1:SF[i]],data[dp[[2L]]]))
  model2 <- train(V1~., NewData, trControl=train_control, method="nb")
  cat("  Selected Feature =",SF[i],"Accuracy
=",max(model2$results$Accuracy),"\n", sep = " ")
}

res3<-fsaer(V1~., data, threshold=1, nsf=10,na.rm = TRUE,verbose=TRUE)

cat("  New Feature Selection Algorithm based on Effective Ranges
(FSAER)", "\n",sep = " ")
for (i in 1:6)
{
  NewData<-as.data.frame(c(res3$RankedData[1:SF[i]],data[dp[[2L]]]))
  model3 <- train(V1~., NewData, trControl=train_control, method="nb")
  cat("  Selected Feature =",SF[i],"Accuracy
=",max(model3$results$Accuracy),"\n", sep = " ")
}
```

EK 2 – Tezden Türetilmiş Yayınlar

Turfan, D., Altunkaynak, B., Yeniay, O., A Novel Feature Selection Algorithm based on Effective Ranges for Classification of Gene Expression Data. Applied Computing and Informatics **(Submitted)**.

EK 3 – Tezden Türetilmiş Bildiriler

Turfan, D., Yeniay, O., “A Comparision of Filter Methods for Classification of High-Dimensional Data Sets”. 14th Applied Statistics 2017, Slovenia, September 24-27, 2017.