Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü Dergisi
Cilt **, Sayı *, **-**, 20**

Süleyman Demirel University
Journal of Natural and Applied Sciences
Volume **, Issue *, **-**, 20**

DOI: 10.19113/sdufbed.01653

Özellik Seçim Yöntemleri ve Yeni Bir Yaklaşım

Hüseyin BUDAK[*1,2]

[1]Mimar Sinan Güzel Sanatlar Üniversitesi, Fen Edebiyat Fakültesi, İstatistik Bölümü, 34360, İstanbul
[2]Tanı Pazarlama ve İletişim Hizmetleri A.Ş., 34700, İstanbul

(Alınış / Received: 19.11.2016, Kabul / Accepted: 06.03.2018, Online Yayınlanma / Published Online: 17.05.2018)

Anahtar Kelimeler
Veri madenciliği,
Özellik seçimi,
Özellik seçim algoritmaları,
Sınıflandırma

Özet: Veri madenciliği sürecinin önemli aşamalarından biri veri boyutunun azaltılması işlemidir. Veri boyutunun azaltılması kısaca, büyük veri kümelerinin depolanması ve analiz edilmesinde karşılaşılan sorunları aşmak için veri kümesinden ilgisiz veya gereksiz değişkenlerin çıkartılması olarak tanımlanmaktadır. Veri boyutunun azaltılması için kullanılan yöntemlerin başında özellik seçimi gelmektedir. Özellik seçimi, orijinal veri setini temsil edebilecek en iyi altkümenin seçimi olarak tanımlanmaktadır. Bu işlem, ilgilenilen problem için en faydalı ve en önemli özellikleri seçerek veri kümesindeki özellik sayısını azaltmayı yani veri boyutunu düşürmeyi amaçlamaktadır. Bu çalışmada, özellik seçim yöntemleri incelenmiş ve alternatif bir yöntem önerilmiştir.

Feature Selection Methods and a New Approach

Keywords
Data mining,
Feature selection,
Feature selection algorithms,
Classification

Abstract: One of important stages of data mining procedure is the process of dimension reduction. The dimension reduction is the process of removing irrelevant or redundant variables from the data set in order to resolve problems encountered in storing big data sets and analyzing them. Feature selection is one of the most popular method among the methods of dimension reduction. Feature selection, is described as the selection of the best subset which can represent the original data set. This process aims to reduce the number of features in the data set by selecting the most useful and important features for the discussed problem. In this study, feature selection methods have been analyzed and an alternative method has been proposed.

1. Giriş

Günümüzde bilişim teknolojilerinde yaşanan gelişmeler, donanımların ucuzlaması ve büyük veri tabanlarının daha ulaşılabilir hale gelmesi gibi nedenler, birçok alanda büyük veri tabanlarının oluşturulmasını ve bu veri tabanlarında depolanan veri miktarının katlanarak artmasını sağlamıştır. Söz konusu veri yığınlarının analiz edilmesinde geleneksel yöntemlerin yetersiz kalması nedeniyle çok sayıda veri madenciliği yöntemi geliştirilmiştir. Veri madenciliği, literatürdeki çeşitli tanımlardan yola çıkarak, farklı araç ve teknolojilerden faydalanarak büyük veri yığınları içerisinde gizli kalmış ilişki, örüntü ve bilgilerin ortaya çıkarılmasını amaçlayan çok aşamalı bir süreç olarak tanımlanabilir. Bu sürecin aşamalarından biri de özellik seçim işlemidir.

Birçok gerçek hayat probleminin çözümü için kullanılan verilerde gereksiz, ilgisiz, gürültülü, yanıltıcı vs değişkenler yer almakta ve problemlerin çözümü için kurulacak modeller hakkında genellikle önsel bilgi bulunmamaktadır. Bağımsız değişken sayısının 20 ve üzeri olduğu durumlarda model için seçilebilecek alternatif değişken sayısı milyonlarca (örneğin 20 değişken için 2^20=1.048.576 seçim) olmaktadır [1]. Bu gibi durumlarda tüm olası alt kümeleri denemek gerek uygulamanın zaman ve maliyeti açısından gerekse kullanılacak veri madenciliği algoritmasının performansı açısından gerçekçi değildir. Bu sebeple, günümüzde kullanılan çok boyutlu verilerde, model kurulumu öncesinde özellik seçimi yapmak çok önemli bir konu haline gelmiştir.

2. Özellik Seçimi

Özellik seçimi (*feature selection*), orijinal veri setini temsil edebilecek en iyi altkümenin seçimi olarak tanımlanmaktadır. Özellik seçimi (diğer adıyla nitelik seçimi veya değişken seçimi), kullanılan algoritmaya göre özellikleri değerlendirerek veri setindeki *n* adet

*İlgili yazar: huseyin.budak@hotmail.com.tr

özellik arasından en iyi *k* adet özelliği seçme işlemidir [2].

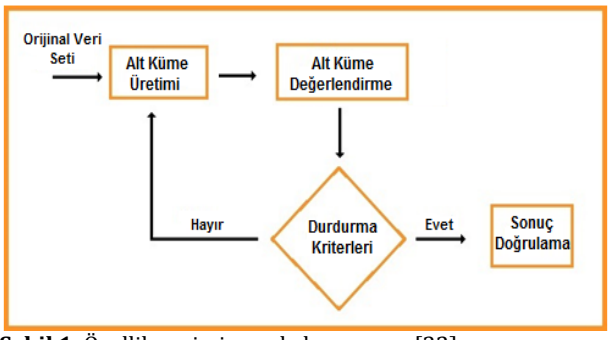
Özellik seçimi, ilgilenilen problem için en faydalı ve en önemli özellikleri seçerek veri kümesindeki özellik sayısının azaltılmasını amaçlamaktadır. Özellik sayısının azaltılması analiz sürecinde uygulamacıya birçok avantaj sağlamaktadır.

Özellik seçme işleminin avantajları [3]:

- Özellik kümesinin boyutunu düşürür ve algoritma hızını arttırır,
- İlgili olmayan ve gürültülü veriyi ortadan kaldırır,
- Veri kalitesini geliştirir,
- Veri kümesini daha basit bir şekilde tanımlanabilir, görselleştirilebilir ve anlaşılabilir hale getirir,
- Veri kümesini oluşturmak için gerekli olan veri toplama işleminde kaynak tasarrufu sağlar,
- Veri depolamak için gerekli olan hafıza miktarını azaltır,
- Elde edilen modelin başarısını arttırır.

2.1. Özellik seçimi genel adımları

Özellik seçimi genel olarak Şekil 1'deki adımlardan oluşan bir süreçtir. Bu süreçte öncelikle orijinal veri setinden bir alt özellik kümesi oluşturulmakta ve daha sonra ele alınan özellikler için farklı formüller aracılığıyla bir değerlendirme yapılarak ilgili özelliğin seçilip seçilmeyeceğine karar verilmektedir. Seçilmesine karar verilen özellik ilgili alt kümeye dahil edilmekte ve algoritmanın durdurma kriteri sağlana kadar süreç devam etmektedir.



Şekil 1. Özellik seçimi genel akış şeması [32]

2.1.1. Alt küme üretimi

Orijinal veri setinden farklı stratejiler kullanılarak daha küçük boyutta veri setleri elde etme işlemi olan alt küme üretimi, esasen değerlendirme yapmak için arama uzayında bulunan aday alt kümeleri belirleyen sezgisel arama sürecidir. Bu sürecin yapısı iki temel konu ile belirlenmektedir. Birinci konu, arama yönünü etkileyen arama başlangıç noktasına (ya da noktalarına) karar verilmesidir. Arama işlemi, ileri yönlü, geri yönlü veya iki yönlü olabilmektedir. İkinci konu, hangi arama stratejisinin kullanılacağına karar verilmesidir. N adet özellik bulunan bir veri seti için, 2^N adet aday alt küme bulunmaktadır. Bu arama uzayı sayısı, N sayısı arttıkça arama işlemini engelleyebilecek seviyeye ulaşmaktadır. Bu nedenle, komple arama, ardışık arama ve rastgele arama gibi farklı stratejiler geliştirilmiştir [4].

Komple Arama

En kapsamlı arama yöntemi olan komple arama, orijinal veri setinden üretilebilecek tüm alt özellik kümelerini üretmektedir. Bu sayede, kullanılan değerlendirme kriterine göre optimal sonucu bulmayı garanti eder. Ancak, arama işleminde tüm alt kümeleri kullanmak işlem yükü açısından gerçekçi bir durum değildir. Optimum sonucu bulma şansını riske atmadan arama uzayını küçültmek için farklı fonksiyonlar kullanılabilmektedir. Bunlardan bazıları, dal ve sınır (*branch and bound*) [5] ve ışın arama (*beam search*) [6] yöntemleridir [4, 40].

Ardışık Arama

Ardışık arama işleminde veri setindeki tüm alt kümeler kullanılmamaktadır. Bu nedenle, en iyi alt kümeyi bulmayı garanti etmez. Optimum sonucu bulabilmek için çeşitli yaklaşımlar kullanılmaktadır [4, 40].

- Ardışık ileri yönlü arama (*sequential forward selection*): Arama işlemine boş bir özellik kümesi ile başlanır. Yapılan değerlendirme sonucunda en iyi olan özellik alt kümeye eklenir. Durdurma kriterleri sağlanıncaya kadar söz konusu özellik ekleme işlemi devam eder.

- Ardışık geri yönlü arama (*sequential backward elimination*): Arama işlemine tam özellik kümesi (orijinal veri seti) ile başlanır. Yapılan değerlendirme sonucunda en kötü olan özellik alt kümeden çıkartılır. Durdurma kriterleri sağlanıncaya kadar söz konusu özellik çıkartma işlemi devam eder.

- İki yönlü arama (*bi-directional selection)*: Ardışık ileri yönlü ve geri yönlü arama yöntemlerini bir arada kullanır. Arama işlemine boş, tam veya rastgele seçilmiş bir özellik alt kümesi ile başlanır. Yapılan değerlendirme sonucunda iyi özellikler alt kümeye eklenir ve kötü özellikler alt kümeden çıkartılır.

Rastgele Arama

Arama işlemine rastgele özellik alt kümesi seçerek başlamakta ve optimum alt kümeyi seçmek için iki farklı yol kullanmaktadır. İlk yol, klasik ardışık arama yöntemlerine rastgeleliğin eklenmesidir. Rastgele tepe tırmanma (*random-start hill-climbing*) ve benzetilmiş tavlama (*simulated annealing*) [6] yöntemleri örnek olarak gösterilebilinir. İkinci yol ise, tamamen rastgele şekilde özellik alt kümesinin üretilmesidir. Las Vegas algoritması [7] örnek olarak gösterilebilinir [4].

2.1.2. Alt küme değerlendirme

Alt küme üretimi aşamasında oluşturulan her yeni alt küme belirli bir değerlendirme kriterleri ile değerlendirilmelidir. Bu adımda, oluşturulan özellik alt kümelerinin değerlendirilmesi yapılarak en iyi özellik alt kümesi bulunmaya çalışılır. Bir alt kümenin iyiliği her zaman belirli bir değerlendirme kriterine göre belirlenmektedir (yani, bir kriter kullanılarak seçilen bir optimum alt küme başka bir kritere göre optimum olmayabilir). Aday alt kümesinin iyiliğini belirlemek için literatürde birçok değerlendirme kriteri önerilmiştir. Bu kriterler, öğrenme algoritmalarına bağımlılıklarına göre bağımsız kriter ve bağımlı kriter olmak üzere genel olarak iki gruba ayrılır [4, 40].

Bağımsız Kriter
Bağımsız kriterler, genellikle filtreleme modeli algoritmalarında kullanılmaktadır. Bu kriterler, herhangi bir öğrenme algoritması içermeksizin eğitim verilerin temel özelliklerini kullanarak bir özelliğin veya özellik alt kümesinin iyiliğini değerlendirmeye çalışmaktadır. Uzaklık ölçütü, bilgi ölçütü, bağımlılık ölçütü ve tutarlılık ölçütü sıklıkla kullanılan bağımsız kriterlere örnektir [4, 40].

Bağımlı Kriter
Sarmal modellerde kullanılan bağımlı kriterler, özellik seçimi için önceden tanımlanmış öğrenme algoritmasının performansına dayanarak hangi özelliklerin seçileceğine karar vermektedir. Uygun özelliklerin seçiminde, bağımsız kriterlere göre genellikle üstün performans gösteren bu yöntemler her bir özellik alt kümesi için tahmin işlemi yaptığından daha fazla hesaplama maliyeti bulunmaktadır. Ayrıca, bağımlı kriterler kullanılarak seçilen özellikler seçim işleminde kullanılan öğrenme algoritmasına bağlı olduğundan başka bir öğrenme algoritması için uygun olmayabilir [4, 40].

2.1.3. Durdurma kriterleri

Durdurma kriterleri, özellik seçimi sürecinin ne zaman durması gerektiğini belirlemektedir. Bazı popüler durdurma kriterleri aşağıdaki gibidir [4, 40].

- Arama işleminin tamamlanmış olması,
- Verilmiş bazı sınırlara ulaşılmış olması (örneğin, minimum özellik sayısı veya maksimum iterasyon sayısı),
- Herhangi bir özelliğin eklenmesi veya çıkarılması daha iyi bir özellik alt kümesi elde ettirmemesi,
- Yeterince iyi bir alt kümeye ulaşılması (örneğin, sınıflandırma hata oranı belirli bir görev için izin verilen hata oranından az ise)

2.1.4. Sonuç doğrulama

Sonuç doğrulama (validasyon) için en basit yol, doğrudan veri hakkındaki önsel bilgiyi kullanarak sonucu ölçmektir. Ancak, gerçek hayat problemlerinde çoğu zaman önsel bilgi bulunmamaktadır. Bu nedenle, özellik değişimine göre algoritma performanslarındaki değişimi izleyen dolaylı yöntemler kullanılmaktadır. Bu yöntemlerde, performans göstergesi olarak genellikle sınıflandırma hata oranı tercih edilmektedir [4].

3. Özellik Seçim Yöntemleri

Özellik seçiminde kullanılan yöntemler, sadece istatistiksel bilgiye dayalı olan filtreleme (*filter*) yöntemleri, özellikler üzerinde arama işlemleri gerçekleştiren sarmal (*wrapper*) yöntemler ve en iyi bölen ölçütünü bulmaya dayalı olan gömülü (*embedded*) yöntemler olmak üzere genel olarak üç grupta toplanmaktadır [8].

Filtreleme yöntemlerinde veri madenciliği algoritması çalışmadan önce özellik seçimi yapılırken, sarmal yöntemlerde veri madenciliği algoritması en iyi özelliklerin seçimi için bir araç olarak kullanılmaktadır. Gömülü yöntemlerde ise, veri madenciliği algoritması ve özellik seçimi algoritması eş zamanlı olarak çalışmaktadır.

Özellik seçim yöntemleri, metin madenciliği [2, 33], kanser teşhisi [9, 38], sahtecilik tespiti [34], kredi skorlama [37], müşteri kaybı analizi [35], spam e-posta tespiti [36] gibi birçok alanda ele alınan probleme ilişkin önemli özelliklerin belirlenmesi işleminde yaygın olarak kullanılmaktadır.

3.1. Filtreleme yöntemleri

Filtreleme yöntemleri veri madenciliğinde kullanılan en eski özellik seçim yöntemleri olarak bilinmektedir. Bu yöntemlerde herhangi bir sınıflandırıcı kullanılmadan uzaklık, bilgi, bağımlılık ve tutarlılık ölçümleri gibi istatistiksel ölçütlere dayalı fonksiyonlar yardımıyla özellik seçimi yapılmaktadır. Benzer mantıkla çalışan bu yöntemlerde, veri kümesinde bulunan her bir özellik için değerlendirme fonksiyonu aracılığıyla bir değer (skor) hesaplanmakta ve hesaplanan bu değerler içerisinde en yüksek değerlere sahip olan özellikler en iyi özellik alt kümesine seçilmektedir. Takip eden alt bölümlerde yaygın olarak kullanılan filtreleme yöntemlerine yer verilmiştir.

3.1.1. Fisher Skor

Fisher Skor yöntemi, her bir sınıf için özelliklere ait ortalama ve standart sapma değerlerini kullanarak bir ilişki skoru hesaplar. Fisher Skor'un hesaplama formülü (1)'deki gibidir [9, 38].

$$F(x_i) = \frac{|\mu_i^+ - \mu_i^-|}{\sigma_i^+ - \sigma_i^-} \quad (1)$$

Formülde bulunan; + ve – işaretleri iki sınıflı bir problem için farklı sınıfları, μ_i^+ ve μ_i^- değerleri sınıfların aritmetik ortalamalarını, σ_i^+ ve σ_i^- değerleri sınıflara ait standart sapma değerlerini göstermektedir.

İki sınıfı birbirinden ayırmaya yardımcı olacak özellikler bulunmaya çalışan bu yöntem ile özellik seçim işlemi, özelliklerin hesaplanan skorlara göre büyükten küçüğe doğru sıralanmasının ardından en üst sıradan başlanarak istenilen sayıda özelliğin seçilmesi şeklinde yapılmaktadır.

Fisher skorunun yüksek olması, ilgili özelliğe ilişkin iki sınıf arasındaki ortalama farkın büyük olduğunu ayrıca ilgili sınıflardaki değerinde küçük sapmalarının olduğunu ifade etmektedir. Bu nedenle, iki sınıfı birbiriden en iyi ayıracak özellikleri seçmek için Fisher skoru yüksek olan özellikler tercih edilmektedir [38].

3.1.2. t-Skor

t-Skor yöntemi, Fisher Skor yöntemine benzer şekilde özellikler için bir ilişki skoru hesaplar. t-Skor yönteminde, Fisher Skor yönteminden farklı olarak hesaplamaya sınıf örnek sayıları da dahil edilir. t-Skor'un hesaplama formülü (2)'deki gibidir [9].

$$t(x_i) = \frac{|\mu_i^+ - \mu_i^-|}{\sqrt{\frac{n_i^+(\sigma_i^+)^2 + n_i^-(\sigma_i^-)^2}{n_i^+ + n_i^-}}} \tag{2}$$

Fisher Skor yöntemine benzer şekilde çalışan bu yöntemle de özellik seçim işlemi, özelliklerin hesaplanan skorlara göre büyükten küçüğe doğru sıralanmasının ardından en üst sıradan başlanarak istenilen sayıda özelliğin seçilmesi şeklinde yapılmaktadır.

3.1.3. Welch t-İstatistiği

Welch t-İstatistiği yöntemi, t-Skor yöntemine benzer şekilde her bir sınıf için özelliklere ait ortalama, standart sapma değerleri ve sınıf örnek sayılarını kullanarak bir ilişki skoru hesaplar. Welch t-İstatistiği'nin hesaplama formülü (3)'deki gibidir [9].

$$WTS(x_i) = \frac{|\mu_i^+ - \mu_i^-|}{\sqrt{\frac{(\sigma_i^+)^2}{n_i^+} + \frac{(\sigma_i^-)^2}{n_i^-}}} \tag{3}$$

Özellik seçim işlemi, en yüksek skora sahip özelliklerin seçilmesi şeklinde yapılmaktadır. Fisher Skor, t-Skor ve Welch t-İstatistiği yöntemleri formülasyon açısından benzerlik gösterdiğinden birbirlerine yakın sonuçlar vermektedirler.

3.1.4. Ki-Kare testi

Gözlenen ve beklenen frekanslar arasındaki farkın anlamlı olup olmadığı temeline dayanan Ki-Kare testi sık kullanılan özellik seçim yöntemlerinden biridir. Yöntem, özellikler (X) ile Y arasında ilişki olup olmadığını test edilmektedir. Yapılan test sonucunda, Y ile ilişkisi olmadığı tespit edilen özellikler veri setinden çıkartılır. Ki-Kare testi için aşağıdaki formüller kullanılmaktadır [10, 11].

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(N_{ij} - \widehat{N}_{ij}\right)^2 \Big/ \widehat{N}_{ij} \tag{4}$$

$$\widehat{N}_{ij} = N_{i.}N_{.j}/N \tag{5}$$

$$d = (I-1)(J-1) \tag{6}$$

Formülerdeki; N_{ij} Y'nin $i.$ ve X'in $j.$ düzeyindeki gözlenen birim sayısı, $\widehat{N}_{ij}$ iki özellik bağımsız iken Y'nin $i.$ ve X'in $j.$ düzeyindeki beklenen birim sayısı, d test istatistiği için kullanılacak Ki-Kare dağılımının serbestlik derecesini göstermektedir.

Belirli bir sayıda özellik seçimi yapılmak istenildiğinde, özelliklerin hesaplanan χ^2 değerine göre büyükten küçüğe doğru sıralanmasının ardından en üst sıradan başlanarak istenilen sayıda özelliğin seçilmesi şeklinde işlem yapılır.

3.1.5. Bilgi kazancı

Y özelliğini tanımak için gereken bilgi ile X özelliği de kullanılarak Y özelliğini tanımak için gereken bilgi arasındaki farkı gösteren Bilgi Kazancı (*Information Gain*) skorunun hesaplanmasında entropi modeli kullanılmaktadır. Entropi, bir sistemdeki belirsizliğin veya tahmin edilemezliğin ölçüsü şeklinde ifade edilir. X özelliğine bağlı olarak Y özelliğinin entropi değerindeki azalmayı gösteren Bilgi Kazancı aşağıdaki gibi hesaplanır [12].

$$H(Y) = -\sum_{y \in Y} p(y)\log_2(p(y)) \tag{7}$$

$$H(Y\backslash X) = -\sum_{x \in X} p(x)\sum_{y \in Y} p(y\backslash x)\log_2(p(y\backslash x)) \tag{8}$$

$$Bilgi\ Kazancı = H(Y) - H(Y\backslash X) \tag{9}$$

Bilgi kazancı simetrik bir ölçüt olup ölçütte X gözlemlendikten sonra Y hakkında kazanılmış bilgi ile Y gözlemlendikten sonra X hakkında kazanılmış bilgi birbirine eşittir. Yöntemin zayıf yanı, daha fazla bilgiye sahip olmasa bile çok çeşitli değerlere sahip özellikler lehine önyargılı şekilde sonuç vermesidir [13].

3.1.6. Kazanç oranı

Kazanç Oranı (*Gain Ratio*) simetrik olmayan bir ölçüt olup, bilgi kazancının çok çeşitli değerlere sahip özellikleri seçme eğiliminin önüne geçmek için kullanılmaktadır. Kazanç Oranı 0-1 aralığında değer almaktadır. Oran 1'e eşit olduğunda X bilgisinin tamamen Y bilgisini tahmin edebildiğini, 0'a eşit olduğunda ise Y ile X arasında hiçbir ilişki olmadığını gösterir. Kazanç Oranı formül (10)'deki gibi hesaplanır [13].

$$Kazanç\ Oranı = \frac{Bilgi\ Kazancı}{H(X)} \qquad (10)$$

3.1.7. Simetrik belirsizlik katsayısı

Simetrik belirsizlik katsayısı (*Symmetrical Uncertainty*), bilgi kazancının olumsuz yanını giderebilmek için bilgi kazancını Y ve X'in entropi değerlerinin toplamına bölmektedir. Simetrik belirsizlik katsayısı da kazanç oranına benzer şekilde 0-1 aralığında değer alır. Simetrik belirsizlik katsayısı 1'e eşit olduğunda X bilgisinin tamamen Y bilgisini tahmin edebildiğini, 0'a eşit olduğunda ise Y ile X arasında hiçbir ilişki olmadığını gösterir [13].

$$SBK = 2\frac{Bilgi\ Kazancı}{H(Y) + H(X)} \qquad (11)$$

3.1.8. Korelasyon tabanlı özellik seçimi

Korelasyon tabanlı özellik seçimi (*Correlation based Feature Selection- CFS*), özellik alt kümelerinin bilgi değerlerini ölçen bir fonksiyonun yanı sıra bir arama algoritması da kullanmaktadır. CFS'nin özellik altkümelerinin değerlerini ölçerken kullandığı yaklaşım her özelliğin sınıf etiketini tahmin etmekteki başarısının yanı sıra aralarındaki iç korelasyon değerlerini de dikkate almaktadır. Bu yaklaşım, iyi özellik altkümeleri ilgili sınıf ile yüksek birbirleri ile düşük korelasyona sahip özelliklerden oluştuğu hipotezine dayanır [12].

$$M_s = \frac{k\bar{r}_{ci}}{\sqrt{k + k(k-1)\bar{r}_{ii}}} \qquad (12)$$

Formüldeki, k altkümedeki özellik sayısı, $\bar{r}_{ci}$ Y ile özellik arasındaki ortalama korelasyonu, $\bar{r}_{ii}$ özelliklerin birbirleri arasındaki ortalama iç korelasyonunu göstermektedir.

3.1.9. Relief

Kira ve Rendell [14] tarafından önerilen Relief yöntemi, özelliklerin değerini aralarındaki bağımlılıkları ortaya çıkartmaya çalışarak bulmaktadır. Yöntem, ikili sınıflandırma problemlerinde özellik seçimi için kullanılmaktadır. Yöntemin mantığı komşuluk algoritmalarına benzemekte olup, özelliğin bulunduğu örneğin ait olduğu ve olmadığı sınıflarda yer alan en yakın örnekleri ağırlıklandırarak çalışmaktadır [14, 15].

Relief algoritmasını oluşturan üç önemli bölüm aşağıdaki gibidir.

1. Aynı sınıfa sahip en yakın örnekteki ilgili özellik değeri ve farklı sınıfa sahip en yakın örnekteki ilgili özellik değerinin belirlenmesi,
2. İlgili özelliğin ağırlığının hesaplanması,
3. Özelliklerin ağırlıklarına göre sıralanması ve belirli bir eşik değeri veren üstteki *k* adet özelliğin seçilmesi

Algoritmanın 2. adımında bahsedilen özellik ağırlıklar (12)'deki formülün *n* kez tekrarlanması yoluyla hesaplanmaktadır [14, 16].

$$W_i = W_{i-1} - (x_i - nearHit_i)^2 + (x_i - nearMiss_i)^2 \qquad (13)$$

Formüldeki; n örnek sayısını, W_i i özelliğinin ağırlığını (önem derecesi), $nearHit_i$ aynı sınıfa sahip en yakın örnekteki ilgili özellik değerini ve $nearMiss_i$ farklı sınıfa sahip en yakın örnekteki ilgili özellik değerini göstermektedir.

3.1.10. One-R

One-R (1R) yöntemi, Holte [17] tarafından önerilmiş basit bir özellik seçim algoritmasıdır. One-R algoritmasında, öncelikle eğitim veri setindeki her özellik için bir kural oluşturulmakta daha sonra oluşturulan her bir kural için sınıflandırma doğrulukları hesaplanmakta ve en az hatalı kurala ait özellik seçilmektedir [13].

One-R algoritma adımları kısaca sözel olarak aşağıdaki gibi ifade edilebilir [18].

Her bir özellik f için,
f nin etki alanındaki her v değeri için,
v değerine sahip f özellik örnek setini seç,
c = seçilen sette en sık rastlanan sınıf değeri,
herbir f özelliği için "f özelliği v değerini alıyorsa sınıfı c" kuralını ekle,
En yüksek sınıflandırma oranına sahip kuralı çıktı olarak üret.

3.2. Sarmal yöntemler

Sarmal yöntemlerde, özellik seçimi için doğru sınıflandırma oranına bağlı olarak verimliliğin ölçüldüğü çeşitli öğrenme algoritmaları kullanılarak en iyi tahmin performansını gösteren özellikler seçilmektedir [19]. Sarmal yöntemler, en iyi alt özellik kümesinin tespit edilmesinde filtreleme yöntemlerine göre daha başarılı olmasına karşın daha yüksek bir hesaplama maliyetine sahiptirler. Takip eden alt bölümlerde yaygın olarak kullanılan sarmal yöntemlere yer verilmiştir.

3.2.1. Ardışık ileri yönde seçim

Ardışık ileri yönde seçim (*Sequential Forward Selection- SFS*) yöntemi, Whitney [20] tarafından önerilen basit ve etkili bir özellik seçim yöntemidir. SFS algoritması, boş bir özellik kümesinden başlayarak her tekrarda daha önce eklenmemiş olan bir özelliğin kümeye eklenmesi ile en iyi özelliklere sahip alt kümeyi bulmayı amaçlamaktadır. Her bir özelliğin alt kümeye dahil edilip edilmemesinde sınıflandırma başarısına olan katkısı dikkate alınır. Algoritma her tekrarda sadece bir tane özelliği alt kümeye ekleyip sınıflandırma oranında artış olmayana kadar devam eder. Herhangi bir tekrarda seçilen özellik daha sonra alt kümeden çıkartılamamaktadır. Ardışık ileri yönde seçim yöntemine ilişkin algoritma kısaca aşağıdaki gibi gösterilmektedir [21].

1. Boş özellik kümesi ile başla
   $Y_0 = \{\emptyset\}$
2. Sıradaki en iyi özelliği seç
   $x^+ = argmax_{x^+ \notin Y_k}[J(Y_k + x^+)]$
3. Eğer $J(Y_k + x^+) > J(Y_k)$ ise
3.1. $Y_{k+1} = Y_k + x^+ ; k = k + 1$ olarak güncelle
3.2. Adım 2'ye git
4. Dur

3.2.2. Ardışık geri yönde seçim

Ardışık geri yönde seçim (*Sequential Backward Selection - SBS*) yöntemi ilk olarak Marill ve Green [22] tarafından önerilmiştir. SBS algoritması, mevcut tüm özellikler içerisinden özellik sayısını azaltarak, sınıflandırma başarısını maksimize edecek şekilde en iyi özelliklere sahip alt kümeyi bulmayı amaçlamaktadır. SFS algoritmasının tersi yönünde çalışan bu algoritma, tüm özelliklerle işleme başlamakta ve sınıflandırma başarısında artış olmayana kadar her adımda bir tane özelliği kümeden çıkartarak devam etmektedir. Seçim sürecinde, özellikler bir kez kümeden çıkartıldıktan sonra kümeye tekrar dahil edilememektedir [3].

3.2.3. L Ekle - R Çıkar

l ekle - *r* çıkar (*plus l – minus r*) yöntemi, ardışık ileri yönde seçim yönteminde kümeye seçilen bir özelliğin bir daha kümeden çıkartılamaması veya ardışık geri yönde seçim yönteminde kümeden çıkartılan bir özelliğin tekrar kümeye dahil edilememesi sorununun belli oranda giderilmesi amacıyla Stearns [23] tarafından önerilmiştir. Algoritma, her adımda öncelikle ileri yönde seçim yöntemiyle l adet özelliği alt kümeye eklemekte ve daha sonra geri yönde seçim yöntemiyle r adet özelliği alt kümeden çıkartmaktadır. Yönteme ilişkin algoritma kısaca aşağıdaki gibi ifade edilmektedir [3, 24].

1. Eğer $l > r$
1.1. ise boş özellik kümesi ile başla
   $Y_0 = \{\emptyset\}$
1.2. değilse tam özellik kümesi ile başla
   $Y_0 = X$
   Adım 3'e git
2. l kez tekrarla (iyi özellik ekleme)
   $x^+ = argmax_{x^+ \notin Y_k}[J(Y_k + x^+)]$
2.1. Eğer $J(Y_k + x^+) > J(Y_k)$ ise
   $Y_{k+1} = Y_k + x^+ ; k = k + 1$ olarak güncelle
3. r kez tekrarla (kötü özellik çıkarma)
   $x^- = argmax_{x^- \notin Y_k}[J(Y_k - x^-)]$
3.1. Eğer $J(Y_k - x^-) > J(Y_k)$ ise
   $Y_{k+1} = Y_k - x^- ; k = k + 1$ olarak güncelle
3.2. Adım 2'ye git
4. Dur

3.2.4. Ardışık ileri yönde kayan seçim

Ardışık ileri yönde kayan seçim (*Sequential Forward Floating Selection- SFFS*) yöntemi, l ekle- r çıkar yöntemine alternatif olarak Pudil vd. [25] tarafından önerilmiştir. *l* ekle - *r* çıkar algoritmasında yer alan l ve r değerleri belirlenirken herhangi bir teorik yapı kullanılmadığından elde edilen sonuç belirlenen alan l ve r değerlerine bağlı kalmaktadır. Bu sorunu giderebilmek adına ardışık ileri yönde kayan seçim algoritmasında alan l ve r değerlerini sabitlemek yerine kayan bir yapı önerilmiştir. Bu sayede, özellik seçiminin herhangi bir adımında mevcut sınıflama başarısı daha yüksek bir değere ulaşıncaya kadar aynı yönde hareket edilir. Ardışık ileri yönde kayan seçim yöntemine ilişkin algoritma kısaca aşağıdaki gibidir [21, 25].

1. Boş özellik kümesi ile başla
   $Y_0 = \{\emptyset\}$
2. Sıradaki en iyi özelliği seç
   $x^+ = argmax_{x^+ \notin Y_k}[J(Y_k + x^+)]$
3. Eğer $J(Y_k + x^+) > J(Y_k)$ ise
3.1. $Y_{k+1} = Y_k + x^+ ; k = k + 1$ olarak güncelle
3.2. En kötü özelliği çıkar
   $x^- = argmax_{x^- \notin Y_k}[J(Y_k - x^-)]$
3.3. Eğer $J(Y_k - x^-) > J(Y_k)$ ise
3.3.1. $Y_{k+1} = Y_k - x^- ; k = k + 1$ olarak güncelle
3.3.2. Adım 3.2'ye git
3.4. Değilse Adım 2'ye git
4. Dur

3.2.5. Ardışık geri yönde kayan seçim

Ardışık geri yönde kayan seçim (*Sequential Backward Floating Selection- SBFS*) yöntemi, Pudil vd. [25] tarafından ardışık ileri yönde kayan yöntemi ile beraber önerilmiştir. Yöntem, ardışık ileri yönde kayan yöntemi ile aynı prensiplere sahip olup tersi yönde hareket etmektedir [25].

3.3. Gömülü yöntemler

Gömülü yöntemler, yapısında hem sınıflandırma algoritması hem de özellik seçimi algoritmasını barındırdığından, sınıflandırma ve özellik seçme

süreçlerini eşzamanlı olarak gerçekleştirirler [19]. Gömülü yöntemler, tıpkı sarmal yöntemlerde olduğu gibi, filtreleme yöntemlerine göre daha yüksek bir hesaplama maliyetine sahiptir. Takip eden alt bölümlerde yaygın olarak kullanılan gömülü yöntemlere yer verilmiştir.

3.3.1. Karar ağaçları

Sınıflandırma problemlerinde yaygın olarak uygulanan algoritmalardan biri olan karar ağaçları özellik seçim işleminde de kullanılmaktadır. Özellik seçimi için karar ağacının yapımında farklı yöntemler kullanılmakta olup en popüleri ID3 algoritmasıdır. Algoritma, bir özelliği seçme ve bu özelliğin değerlerine göre verilen örnek kümesini ayırma işlemini tekrarlanan bir süreç ve bir dizi eğitim kümesi aracılığıyla öğrenmektedir. Buradaki anahtar soru sınıflandırmanın belirlenmesinde hangi özelliğin en etkili olduğu ve bu nedenle hangisinin ilk olarak seçileceğidir. Bu aşamada ID3 algoritması diğer özellikler içerisinden sınıflandırmada en ayırıcı niteliğe sahip özelliği seçmek için entropi kavramını kullanmaktadır. Söz konusu algoritma adımları sözel olarak aşağıdaki gibi ifade edilebilir [3].

- Her bir özellik için entropi değerini hesapla,
- En düşük değerli entropiye sahip özellik değerine göre örnek kümesini böl,
- Tüm özellikler bölünene veya diğer verilen durma kriteri sağlana kadar tüm alt küme örnekleri için bu adımları tekrarla

3.3.2. Destek vektör makineleri-yinelemeli özellik elemesi

Destek vektör makineleri-yinelemeli özellik elemesi (*Support vector machines-Recursive feature elimination / SVM-RFE*), bir çeşit geriye doğru özellik seçim yöntemidir. SVM-RFE yöntemi sınıflandırma performansını optimize eden özellik alt kümesini bulmak için, öncelikle tüm özellikleri bir amaç fonksiyonuna bağlı olarak derecelendirmekte ve daha sonra en düşük skora sahip özelliği özellik kümesinden çıkartmaktadır. Bu işlem en yüksek sınıflama başarısı bulunana kadar devam etmektedir [26]. SVM-RFE algoritmasının çalışma prensibi, SVM sınıflandırıcısı ile eğitim işlemi yapıp, elde edilen sonuç ile özellik ağırlıklarını hesaplayarak en düşük ağırlıklara göre özellikleri elemek üzerinedir. Algoritmanın adımları sözel olarak aşağıdaki gibidir [27].

Eğitim seti : $X_0 = [x_1, x_2, x_3 \dots x_i]^T$
Sınıf etiketleri : $y = [y_1, y_2, y_3 \dots y_i]^T$
Kalan özellikler : $s = [1,2,3, \dots n]$
Sıralanmış özellik listesi : $r = [\ \ ]$

- Eğitim seti üzerinde SVM ile sınıflandırma işlemi gerçekleştir,
  İyi özellik indeksleri için eğitim setini sınırla:
  $X = X_0(:, s)$
  Sınıflandırıcıyı eğit : $\alpha = SVM - train(X, y)$
- Sınıflandırma sonucu ile ağırlık vektörlerini hesapla,
  Ağırlık vektörü : $w = \sum_k \alpha_k y_k x_k$
- Elde edilen ağırlık vektöründen sıralama kriterini hesapla,
  Sıralama kriteri : $c_i = (w_i)^2$, tüm i ler için
- En küçük sıralama kriterine sahip özelliği bul,
  $f = argmin(c)$
- Sıralanmış özellik listesini güncelle,
  $r = [s(f), r]$
- En küçük sıralama kriterine sahip özelliği kümeden çıkart,
- Kalan özellik ve sınırlı eğitim seti ile işlemi tekrarla.

4. Materyal ve Metot

4.1. Önerilen yöntem

Önerilen yöntem, filtreleme yöntemleri içerisinde sıklıkla kullanılan "Fisher Skor" yöntemini temel alarak geliştirilmiştir. Fisher Skor yönteminde her bir sınıf için özelliklere ait ortalama, standart sapma ve örnek sayıları kullanılarak bir ilişki skoru hesaplamakta ve skoru yüksek olan özellikler veri kümesine dahil edilmektedir (yöntemin detayları alt bölüm 3.1.1.'de verilmiştir). Önerilen yöntemde, sınıflandırma başarısına katkısının daha fazla olacağı düşünülen özelliklerin seçilmesi amaçlanmıştır. Bu doğrultuda formüle r_{iy} ve $\bar{r}_{ix}$ terimleri eklenerek söz konusu özelliklere ilişkin skorlar arttırılmaya çalışılmıştır. Önerilen yöntem ile seçilecek özelliklerin sınıf etiketleriyle (y) yüksek ve diğer özelliklerle (x) düşük korelasyona sahip olması istenildiğinden formüle r_{iy} terimi çarpım, $\bar{r}_{ix}$ terimi ise bölüm olarak eklenmiştir. Önerilen yöntem için korelasyonlar hesaplaması yapılırken ilişkinin yönü ile ilgilenilmediğinden r_{iy} ve $\bar{r}_{ix}$ terimleri mutlak değer olarak ele alınmıştır. Böylelikle formül (14)'deki ifadeye ulaşılmıştır. Sınıf etiketleriyle olan korelasyonu diğer özelliklerle olan korelasyonundan yüksek olan bir özelliğe ait $\frac{|r_{iy}|}{\bar{r}_{ix}}$ değeri 1'den büyük çıkacağından önerilen yöntem ile hesaplan skor Fisher Skor yöntemi ile hesaplanandan daha yüksek olacaktır. Bu sayede, yapılan skor sıralamasında söz konusu özellik daha yukarı çıkacağından özelliğin seçilme ihtimali artacaktır. Önerilen yöntem ile özellik seçim işlemi diğer yöntemlerde olduğu gibi özelliklerin hesaplanan skorlara göre büyükten küçüğe doğru sıralanmasının ardından en üst sıradan başlanarak istenilen sayıda özelliğin seçilmesi şeklinde yapılmaktadır.

$$\grave{F}(x_i) = \frac{|\mu_i^+ - \mu_i^-|}{\sigma_i^+ - \sigma_i^-} \frac{|r_{iy}|}{\bar{r}_{ix}} \qquad (14)$$

Formül (14)'de bulunan; + ve – işaretleri iki sınıflı bir problem için farklı sınıfları, μ_i^+ ve μ_i^- değerleri

sınıfların aritmetik ortalamalarını, σ_i^+ve σ_i^- değerleri sınıflara ait standart sapma değerlerini, $|r_{iy}|$ilgili özelliğin sınıf etiketleriyle olan korelasyonun mutlak değerini, $\bar{r}_{ix}$ ilgili özelliğin diğer tüm özelliklerle olan korelasyonun ortalamasını (ortalama hesaplanırken korelasyonların mutlak değerleri kullanılmıştır) göstermektedir.

4.2. Kullanılan metot ve veri setleri

Çalışmanın amacı, özellik seçim yöntemlerinden Fisher Skor yöntemiyle bu yönteme alternatif olarak önerilen özellik seçim yöntemini yüksek boyutlu veri setlerinde uygulayarak elde edilen sonuçlar ile söz konusu yöntemleri karşılaştırmaktır.

Çalışmada deneysel karşılaştırmaları gerçekleştirmek için Neural Information Processing Systems Conference (NIPS) 2003- Feature Selection Challenge yarışmasında kullanılan veri setlerinden Arcene ve Gisette isimli veri setleri kullanılmıştır. NIPS 2003- Feature Selection Challenge web sitesinde [28] yayınlanmakta olan söz konusu veri setlerine ilişkin özet bilgi aşağıdaki gibidir.

Arcene
Arcene, amacı kütle spektrometrisi verileri ile normal ve kanserli hücreleri ayırt etmek olan iki sınıflı bir sınıflandırma problemi verisidir. Verinin orijinal kaynağı National Cancer Institute (NCI) ve Eastern Virginia Medical School (EVMS) kurumlarıdır. Söz konusu veri setinin özellik sayısı 10.000 ve örnek sayısı 100'dür [29, 30].

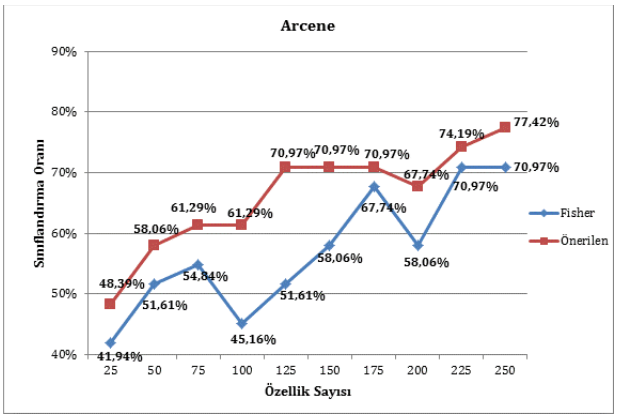
Gisette
Gisette, amacı karıştırılabilir el yazısı ile yazılmış tek haneli rakamları (dört ve dokuz) ayrıt etmek olan iki sınıflı bir sınıflandırma problemi verisidir. Verinin orijinal hali NEC Araştırma Enstitüsünde MNIST verileri kullanılarak Yann LeCun tarafından hazırlanmıştır. Söz konusu veri setinin özellik sayısı 5.000 ve örnek sayısı 6.000'dir [29, 31].

Çalışmada öncelikle, Arcene ve Gisette veri setlerinde yer alan tüm özellikler için excel programı aracılığıyla Fisher Skor ve önerilen özellik seçim yöntemine ilişkin skorlar hesaplanmıştır. Hesaplanan skorlara göre yukarıdan aşağıya doğru tüm veri setleri için ilk 25, 50, 75, 100, 125, 150, 175, 200, 225 ve 250 özellik seçilerek toplam 40 adet veri kümesi oluşturulmuştur. Daha sonra, söz konusu veri kümelerine SPSS Clementine 12.0 programı aracılığıyla Yapay Sinir Ağları (YSA) sınıflandırma yöntemi uygulanarak sınıflandırma doğruluk yüzdeleri elde edilmiştir. YSA, yapay sinir hücrelerinin birbirleri ile çeşitli şekillerde bağlanmalarından oluşmaktadır. Hücre çıktıları, ağırlıklar üzerinden bir sonraki katmandaki tüm hücrelere girdi olarak bağlanmaktadır. Hücrelerin bağlantı şekillerine, öğrenme kuralına ve etkinlik fonksiyonlarına göre çeşitli YSA modelleri bulunmaktadır [32]. Bu çalışmada tek gizli katmanlı MLP modeli kullanılmıştır. Sınıflandırma işleminde öncelikle, veri kümeleri eğitim (%70) ve test (%30) olmak üzere iki ayrı gruba ayrılmıştır. Daha sonra, eğitim kümesi üzerinden model kurulmuş ve kurulan model test kümesine uygulanarak sınıflandırma doğruluk yüzdeleri elde edilmiştir.

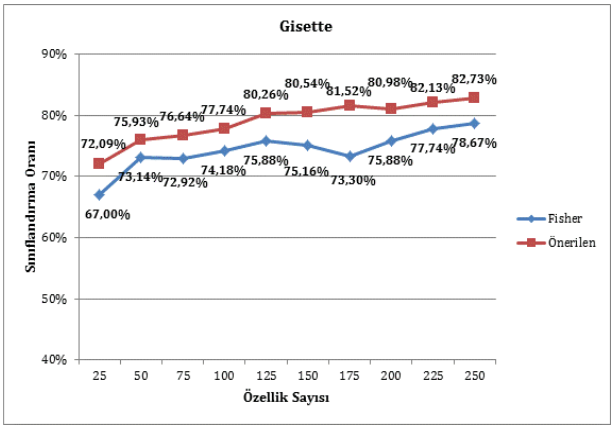
Fisher Skor yöntemi ile önerilen özellik seçim yönteminin karşılaştırılması için hesaplanan sınıflandırma doğruluk yüzdelerinin bu yöntemlere göre istatistiksel olarak anlamlı bir farklılık gösterip göstermediğinin test edilmesi amacıyla IBM SPSS Statistics 21.0 programı aracılığıyla Mann-Whitney U testi uygulanmıştır. Uygulanan tüm testlerde 0,05 anlamlılık düzeyi esas alınmıştır.

5. Bulgular

Arcene ve Gisette veri setlerinden elde edilen veri kümeleri için hesaplanan sınıflandırma doğruluk yüzdeleri aracılığıyla Fisher Skor ile önerilen yönetimi karşılaştıran grafikler Şekil 2 ve Şekil 3'de sunulmuştur. Söz konusu grafikler incelendiğinde, önerilen özellik seçim yöntemi ile seçilen veri kümlerinin tamamında sınıflandırma doğruluk yüzdeleri Fisher Skor yöntemi ile seçilen veri kümlerinin sınıflandırma doğruluk yüzdelerinden fazla olduğu görülmektedir.



Şekil 2. Arcene veri seti için sınıflandırma doğruluk yüzdelerinin karşılaştırılması



Şekil 3. Gisette veri seti için sınıflandırma doğruluk yüzdelerinin karşılaştırılması

Sınıflandırma doğruluk yüzdelerine göre Fisher Skor ve önerilen yöntem arasındaki fark görsel olarak ortaya konulmuştur. Ancak, bu farkın istatistiksel açıdan anlamlı olup olmadığının test edilmesi gerekmektedir. Bu test işleminin uygulanabilmesi için, veri setlerine göre daha önce ayrı ayrı belirtilen sınıflandırma sonuçları bir araya getirilerek her iki yöntem için 20'şer gözlem bulunan yeni bir veri seti oluşturulmuştur. Elde edilen bu veri seti normal dağılım göstermediğinden yapılacak grup karşılaştırmasında Mann-Whitney U testi kullanılmıştır.

Tablo 1. Mann-Whitney U test sonucu

	N	Ort. (%)	SS	U	P
Fisher Skor	20	65,74	0,11	279,0	0,033
Önerilen yöntem	20	72,59	0,09		

Yapılan analiz sonucunda; çalışmada elde edilen sınıflandırma doğruluk yüzdelerinin Fisher Skor ve önerilen özellik seçim yöntemine göre istatistiksel olarak anlamlı bir farklılık gösterdiği tespit edilmiştir ($p=0,033<\alpha=0,05$ olduğu için). Söz konusu farklılık incelendiğinde; önerilen yönteme ait sınıflandırma doğruluk yüzde ortalamasının (%72,59) Fisher Skor yöntemi ortalamasından (%65,74) yüksek olduğu görülmüştür.

6. Tartışma ve Sonuç

Özellik seçimi, orijinal veri setini temsil edebilecek en iyi altkümenin seçimi olarak tanımlanmaktadır. Bu işlem, ilgilenilen problem için en faydalı ve en önemli özellikleri seçerek veri kümesindeki özellik sayısını azaltılmayı yani veri boyutunu düşürmeyi amaçlamaktadır. Günümüzün teknolojik imkanları sayesinde birçok gerçek hayat probleminin çözümü için depolanan verilerde yüzlerce hatta binlerce özellik bulunmaktadır. Bu ölçekteki veriler analiz edilmek istenildiğinde, gerek uygulamanın zaman ve maliyeti açısından gerekse kullanılacak veri madenciliği algoritmasının performansı açısından ciddi sorunlarla karşılaşılmaktadır. Bu nedenle, çok boyutlu verilerde analiz öncesinde özellik seçimi yapmak önemli bir konu haline gelmiştir.

Özellik seçiminde birbirlerine göre avantajları ve dezavantajları bulunan birçok yöntem kullanılmaktadır. Bu yöntemler, istatistiksel bilgiye dayalı olan ve veri madenciliği algoritmasından önce çalışan filtreleme (*filter*) yöntemleri, özellikler üzerinde arama işlemleri gerçekleştiren ve veri madenciliği algoritması en iyi özelliklerin seçimi için bir araç olarak kullanan sarmal (*wrapper*) yöntemler, en iyi bölen ölçütünü bulmaya dayalı olan ve veri madenciliği algoritması ile eş zamanlı çalışan gömülü (*embedded*) yöntemler olmak üzere genel olarak üç grupta toplanmaktadır.

Bu çalışmada, filtreleme yöntemleri içerisinden Fisher Skor yöntemine alternatif olabilecek yeni bir yöntem önerilmiş ve bu yöntemin başarılı olup olmadığını tespit edebilmek amacıyla sınıflandırma doğruluk yüzdeleri kullanılarak karşılaştırma yapılmıştır. Yapılan karşılaştırma sonucunda, önerilen yöntem ile seçilen tüm veri kümelerinden hesaplanan sınıflandırma doğruluk yüzdeleri Fisher Skor yöntemi ile seçilen veri kümelerine ait yüzdelerden daha yüksek olduğu görülmüştür. Ayrıca, iki yönteme ait sınıflandırma doğruluk yüzdeleri arasındaki farkın istatistiksel olarak anlamlı olup olmadığını sınamak amacıyla hipotez testi uygulanmıştır. Söz konusu test sonucunda, uygulamada elde edilen veri kümeleri üzerinden hesaplanan sınıflandırma doğruluk yüzdelerinin Fisher Skor ile önerilen özellik seçim yöntemine göre istatistiksel olarak anlamlı bir farklılık gösterdiği ve önerilen yönteme ait sınıflandırma doğruluk yüzde ortalamasının (%72,59) Fisher Skor yöntemi ortalamasından (%65,74) yüksek olduğu tespit edilmiştir.

Sonuç olarak, filtreleme özellik seçim yöntemleri arasında sıkça kullanılan Fisher Skor yöntemine alternatif olarak önerilen yöntemden elde edilen sınıflandırma sonuçlarının daha başarılı olduğu görülmüştür. Dolayısıyla, önerilen yöntemin özellik seçim işleminde Fisher Skor yöntemine alternatif olarak kullanılabileceği ve daha iyi sonuçlar verebileceği söylenebilir.

Teşekkür

Bu çalışma Mimar Sinan Güzel Sanatlar Üniversitesi Fen Bilimleri Enstitüsü'nde yapılan "Özellik Seçim Yöntemleri ve Yeni Bir Yaklaşım" adlı doktora tezinden üretilmiştir.

Kaynakça

[1] Bozdağ, H. 2004. Intelligent Statistical Data Mining with Information Complexity and Genetic Algorithms, Statistical Data Mining and Knowledge Discovery, Chapman and Hall/CRC, Florida.

[2] Forman, G. 2003. An Extensive Empirical Study of Feature Selection Metrics for Text Classification, Journal of Machine Learning Research, 3, 1289–1305.

[3] Ladha, L., Deepa, T. 2011. Feature Selection Methods And Algorithms, International Journal on Computer Science and Engineering, 3(5), 1787-1797.

[4] Liu, H., Yu, L. 2005. Towards Integrating Feature Selection Algorithms For Classification And Clustering, Knowledge and Data Engineering, IEEE Transactions on Computers, 17(4), 491-502.

[5] Narendra, P., Fukunaga, K. 1977. A Branch and Bound Algorithm for Feature Subset Selection, IEEE Transactions on Computers, 26(9), 917-922.

[6] Doak, J. 1992. An Evaluation of Feature Selection Methods and Their Application to Computer Security, University of California at Davis,Technical Report, California.

[7] Brassard, G., Bratley, P. 1996. Fundamentals of Algorithms,Prentice Hall Professional, New Jersey.

[8] Saeys, Y., Inza, I., Larranaga, P. 2007. A review of feature selection techniques in bioinformatics, Bioinformatics, 23(19), 2507-2517.

[9] Yıldız, O., Tez, M., Bilge, H.Ş., Akcayol ,M.A., Güler, İ. 2012. Meme Kanseri Sınıflandırması için Gen Seçimi, IEEE 20. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 18-20 Nisan, Muğla.

[10] Inc, S. 2007. SPSS Clementine 12.0 Algorithms Guide, SPSS Inc, Chicago.

[11] Ünver, Ö., Gamgam, H. 2006. Uygulamalı Temel İstatistik Yöntemler, Seçkin Yayıncılık, Ankara.

[12] Hall, M. 1999. Correlation-based Feature Selection for Machine Learning, The University of Waikato, PhD Thesis,Hamilton.

[13] Novakavic, J., Strbac, P., Bulatovic, D. 2011. Toward Optimal Feature Selection Using Ranking Methods and Classification Algorithms, Yugoslav Journal of Operations Research, 21(1), 119-135.

[14] http://en.wikipedia.org/wiki/Relief_(feature_selection (Erişim Tarihi: 31.08.2014).

[15] Biricik, G. 2012. Sınıf Bilgisini Kullanan Boyut İndirgeme Yöntemlerinin Metin Sınıflandırmadaki Etkilerinin Karşılaştırılması, IEEE 20. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 18-20 Nisan,Muğla.

[16] Kira, K., Rendell, L. 1992. The Feature Selection Problem: Traditional Methods and a New Algorithm, AAAI-92, 129-134.

[17] Holte, R. 1993. Very simple classification rules perform well on most commonly used datasets, Machine Learning, 11, 63-91.

[18] Holmes, G., Nevill-Manning, C. 1995. Feature selection via the discovery of simple classification rules, To appear in Proceedings of Symposium on Intelligent Data Analysis (IDA–95), 17-19 Ağustos, Baden-Baden.

[19] Guyon, I. Elisseeff, A. 2003. An Introduction to Variable and Feature Selection, Journal of Machine Learning Research, 3, 1157-1182.

[20] Whitney, A. 1972. A direct method of nonparametric measurement selection, IEEE Transactions on Computers, 20(9), 1100-1103.

[21] Pratama, S., Muda, A., Choo, Y., Muda, N. 2011. Computationally Inexpensive Sequential Forward Floating Selection for Acquiring Significant Features for Authorship Invarianceness in Writer Identification, International Journal of New Computer Architectures and their Applications, 1(9), 581-598.

[22] Marill, T., Green, D. 1963. On the effectiveness of receptors in recognition system, IEEE Trans. Inform. Theory, 9, 11-17.

[23] Stearns, S. 1976. On selecting features for pattern classifiers, 3rd International Conference on Pattern Recognition, 8-11 Kasım, Coronado.

[24] http://www.facweb.iitkgp.ernet.in/~sudeshna/courses/ML06/featsel.pdf(Erişim Tarihi: 29.10.2014).

[25] Pudil, P., Novovicova, J., Kittler, J. 1994. Floating search methods in feature selection, Pattern Recognition Letters, 15, 1119-1125.

[26] Eskidere, Ö. 2012. Ses Ölçümlerinden Parkinson Hastalığının Teşhisi İçin Öznitelik Seçme Yöntemlerinin Karşılaştırılması, Sigma, 20, 402-414.

[27] Guyon, I., Weston, J., Barnhill, S., Vapnik, V. 2002. Gene selection for cancer classification using support vector machines, Machine Learning, 46, 389-422.

[28] http://www.nipsfsc.ecs.soton.ac.uk/datasets/ (Erişim Tarihi: 11.01.2015)

[29] http://www.nipsfsc.ecs.soton.ac.uk/papers/NIPS2003-Datasets.pdf (Erişim Tarihi: 11.01.2015)

[30] Petricoin, E. F., Ardekani, A. M., Hitt, B. A., Levine, P. J., Fusaro, V. A., Steinberg, S. M., ... & Liotta, L. A., 2002. Use of proteomic patterns in serum to identify ovarian cancer, The lancet, 359(9306), 572-577.

[31] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. Gradient-based learning applied to document recognition, Proceedings of the IEEE, 86(11), 2278-2324.

[32] Dash, M., Liu, H. 1997. Feature Selection for Classification, Intelligent Data Analysis, Elsevier, 131-156.

[33] Liu, T., Liu, S., Chen, Z. and Ma, WY., 2003. An Evaluation on Feature Selection for Text Clustering, Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), 21-24 Ağustos, Washington D.C.

[34] Pouramirarsalani, A., Khalilian, M. and Nikravanshalmani, A. 2017. Fraud detection in E-banking by using the hybrid feature selection and evolutionary algorithms, IJCSNS, 17(8), 271-279.

[35] Subramanya, KB., Somani, A. 2017. Enhanced feature mining and classifier models to predict customer churn for an E-retailer, Cloud Computing, Data Science & Engineering-Confluence, 2017 7th International Conference on, 12-13 Ocak, Noida.

[36] Mohamad,M. and Selamat, A. 2015. An evaluation on the efficiency of hybrid feature selection in spam email classification, Computer, Communications, and Control Technology (I4CT), 2015 International Conference on, 21-23 Nisan, Kuching.

[37] Wang, D., Zhang, Z., Bai, R. and Mao, Y. 2017. A hybrid system with filter approach and multiple population genetic algorithm for feature selection in credit scoring, Journal of Computational and Applied Mathematics, 329, 307-321.

[38] Bolon-Canedo, V., Sanchez-Marono, N., Alonso-Betanzos, A., Benítez, J.M. and Herrera, F. 2014. A review of microarray datasets and applied feature selection methods, Information Sciences, 282, 111–135.

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

[40] Kumar, V., Minz, S. 2014. Feature Selection: A literature Review, Smart Computing Review, 4(3), 211-229.