Journal of the Faculty of Engineering and Architecture of Gazi University 31:4 (2016) 1017-1025

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Katı kümeleme ve yeni bir geçiş fonksiyonuyla uzman karışımlarında sınıflandırma

Faruk Bulut[1*], M. Fatih Amasyalı[2]

[1]Katip Çelebi Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, Çiğli, İzmir
[2]Yıldız Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Bilgisayar Mühendisliği Bölümü, Davutpaşa, İstanbul

Ö N E Ç I K A N L A R

- Katı kümeleme ile yeni bir Uzman Karışımı modeli
- Yeni bir geçiş fonksiyonun tasarlanması
- Uzman Karışımları ile yüksek sınıflandırma performansı

Makale Bilgileri

Geliş: 10.10.2015
Kabul: 28.03.2016

DOI:
10.17341/gummfd.32784

Anahtar Kelimeler:
Uzman karışımları, geçiş fonksiyonu, sınıflandırma

ÖZET

Uzman Karışımları, Kolektif Öğrenme metotlarından biridir ve sınıflandırma başarısını artırmak için yaygın olarak kullanılmaktadır. Bu algoritmada, veri seti yumuşak kümeleme ile alt bölümlere ayrılır. Her bir bölüm için ayrı bir sınıflandırıcı uzman oluşturulur ve o bölümdeki örneklerle eğitilir. Belirlenen bir geçiş fonksiyonu ile de uzmanların kararları birleştirilerek sınıflandırma işlemi yapılır. Bu çalışmada veri seti, literatürde önerilen yumuşak kümeleme yerine katı kümeleme yöntemiyle alt veri setlerine bölünmüştür ve her bir alt veri seti için ayrı bir sınıflandırıcı atanmıştır. Uzmanların kararları da önerilen yeni bir geçiş fonksiyonu ile birleştirilmiştir. Bu geçiş fonksiyonu sayesinde, her hangi bir test noktası için tüm uzmanların o test noktasına olan uzaklıklarına bağlı olarak verdikleri kararlar ağırlıklandırılarak birleştirilir ve ortak komite kararı hesaplanmıştır. Bir grup veri seti üzerinde yapılan denemelerde, karar ağaçları ve *k* en yakın komşuluk algoritması gibi tekil sınıflandırıcılara göre uzman karışımlarının daha başarılı sonuçlar verdiği gözlemlenmiştir.

Classification in mixture of experts using hard clustering and a new gate function

H I G H L I G H T S

- A new model with hard clustering
- A new gate function design
- High classification performance via Mixture of Experts

Article Info

Received: 10.10.2015
Accepted: 28.03.2016

DOI:
10.17341/gummfd.32784

Keywords:
Mixture of experts, gate function, classification

ABSTRACT

As one of the ensemble methods, Mixture of Experts is used to gain higher prediction performance in classification area. In this technique, a dataset is divided into regions by soft clustering. An expert for each region is assigned and trained with the samples of the corresponding region. In this study, a dataset is divided into regions by a hard clustering method and a decision tree classifier is constructed for each region as an expert. Each expert has their individual decision for a test point. The class prediction for the test point is performed by the proposed gate function which aggregates all the decisions of experts. In calculations of the final decision via the gate function, each expert has different weights according to its distance to the test point. In the experiments on a group of benchmark datasets, better performance has been obtained with the new gate function. This new method gives better results than base classifiers such as decision trees and k Nearest Neighbors.

[*] Sorumlu Yazar/Corresponding author: faruk.bulut@ikc.edu.tr / Tel: +90 232 329 3535

1. GİRİŞ (INTRODUCTION)

RBF (*Radial Basis Functions*) metodunun geliştirilmiş bir versiyonu olan UK (Uzman Karışımları) ilk Jacobs ve Jordan tarafından önerilmiştir [1]. UK, kolektif öğrenme (*Ensemble Learning*) yöntemlerinden biridir. UK'da amaç, veri setini böl-yönet mantığıyla daha az karmaşık bir yapıya indirgeyerek sınıflandırma performansını artırmaktır. UK ile veri seti en uygun şekilde alt kümelere bölünür ve her bir alt küme için bir uzman atanarak o bölgedeki örneklerle eğitilir. Amaç her bir alt bölgenin bir veri seti gibi düşünülüp o bölge için en uygun sınıflandırıcının eğitilmesidir ve toplam sınıflandırma performansının artırılmasıdır. Bir veri setinin alt bölgeleri birbirinden farklı yapılar içeriyor olabilir. Bu farklılıklar, örnek sayısından, sınıf etiketi sayısından, verilerin tipinden, yoğun veya seyrek oluşundan ve değişik istatistiksel ölçütlerden kaynaklanıyor olabilir. Bu farklılıklar sınıflandırma başarısı üzerinde olumlu ya da olumsuz etki oluşturabilir. Kümülatif sınıflandırma başarısını artırmak amacıyla tüm veri kümesi için tek bir model belirlemek yerine her bir alt bölge için farklı modeller tasarlamak ve bu modelleri uygun bir fonksiyonlar birleştirmek gerekebilir. UK algoritmasının başarısı iki temelde dayalıdır. Birincisi veri setinin alt bölümlerinin nasıl belirlendiği, diğeri ise test örneklerinin nasıl etiketlendiğidir. Literatürde veri setinin bölümlere ayrılma işlemi Beklenti Artırımı (*Expectation Maximization*, BA) algoritmasıyla ve etiketleme işleminin de BA yöntemine özgü bir geçiş fonksiyonu (*Gating Function*) ile yapıldığı anlatılmaktadır [2]. Bilindiği üzere BA yöntemi, yumuşak bir kümele (*soft clustering*) yapmaktadır. Diğer bir deyişle veri setindeki her bir örnek belirli bir oranda birden fazla kümeye ait olabilir. UK yönteminde bir test örneğinin etiketlenmesi işlemi ait olduğu her bir kümenin uzmanları tarafından verilen kararların birleştirilmesiyle yapılır. Verilen her bir karar ilgili test örneğinin o kümeye ait olma oranıyla doğru orantılı olacak şekilde ağırlıklandırılır. Karaların bir fonksiyon ile birleştirilmesiyle de komite kararı bulunmuş olur. UK alanında son 20 yıl içerisinde teorik çalışmalar yapılmış, farklı yaklaşımlar sunulmuştur. Bir sınıflandırma modeli olarak UK algoritması, hazır ve ortak kullanıma açık veri setlerinin yanı sıra özel bir amaç için hazırlanmış olan veri kümelerine de uygulanabilmektedir. UK modeli görüntü işleme, yüz tanıma, yüzden cinsiyet ve etnik özellikleri çıkarma, el yazısı tanıma, robot yol bulma, sağlık ve tıp bilişimi, protein etkileşim tahmini, ses tanıma, ses gruplama, 3B nesne tanıma, sinyal işleme, hava tahmin ve raporlama gibi birçok alana uygulanmış ve diğer klasik yöntemlere göre daha başarılı sonuçlar elde edilmiştir. Ayrıca geliştirilmiş değişik stratejilerle UK'nın sınıflandırma başarısı artırılmaya çalışılmıştır. Seniha, Wilson ve Gader 2014 yılında yayınladıkları literatür taramasında [3], UK alanında son 20 yıl içerisinde yapılmış çalışmaları incelenmiştir. Çalışmalarında UK'nın varyasyonlu öğrenme metoduna (*Variational learning of ME*), hiyerarşik UK modellerine, en uygun sayıdaki uzmanın tespit edilmesine, UK ile geliştirilmiş sınıflandırma ve regresyon modellerine, Bayesin UK modeline, UK'nın istatistiksel modellerine, yıllar içerisindeki değişimlerine ve bu alanda yapılan pratik uygulamalara yer verilmiştir. Bir veri setinde sınıf türlerine ait veri dağılımları dengesiz ve asimetrik olabilir. Sınıf türlerine bakılmaksızın örnekler veri uzayında kuyruk yapısında veya kıvrımlı bir şekilde bulunabilir. Genel olarak tüm veriler gauss dağılımına uygun da olmayabilir. Ayrıca belirli özniteliklere ait veriler zamana, mekâna, sıcaklığa ve bazı durumlara bağlı olarak değişiklik gösterebilir. Bir veri seti hem devingen hem de durağan veriler içerebilir. Bu gibi durumlarda hem tekil sınıflandırıcıların [4] hem de UK modellerinin [5] beklenen düzeyde performans gösteremeyeceği literatürde belirtilmiştir. Akben ve Alkan tarafından yapılan çalışmada [4] öznitelikler arasındaki korelasyonun düşük olmasının sınıflandırma başarısını olumsuz etkilediği vurgulanmıştır. Chamroukhi tarafından yapılan çalışmada [5] ise Non-Normal Mixture of Experts (NNMoE) modeli öne sürülmüştür. Dengesiz (*skewed*) ve kuyruklu (*tailed*) veri setleri oluşturmak için ürettikleri sentetik veri setlerinde klasik UK modeline göre regresyon, sınıflandırma ve kümeleme işlemlerinde yüksek düzeyde başarı oranı elde edilmiştir. Normal bir dağılıma sahip veri setinde gürültülü noktalara (*noise*) karşı normal bir UK modeli hassas bir yapıya sahiptir. Nguyen ve McLachlan tarafından yapılan bir çalışmada ise gürültü eklenmiş veri setlerine karşı gürbüz (*robust*) bir yapıya sahip Laplace mixture of linear experts (LMoLE) modelini öne sürmüşlerdir [6]. Lineer bir regresyon modeli ile sınıflandırma yapan LMoLE algoritmasında gürültülü verilere karşı başarılı sonuçlar elde edilmiştir. Bu çalışmada, her bir alt bölgeyi ayrı bir veri seti gibi düşünerek olası bazı olumsuz etkilerin en aza indirilmesi amaçlanmıştır. Bu sayede genel sınıflandırma başarısının artırılacağı düşünülmüştür. Bu amaçla tüm veri seti için tek bir öğrenici yerine, veri setinin her bir alt bölümü için ayrı ayrı öğreniciler inşa edilmiştir. Veri setinin alt bölümleri *k*-means yöntemiyle katı kümeleme (*hard clustering*) yapılarak belirlenmiştir. Normalde UK tasarımlarında, DBSCAN, OPTICS, Fuzzy-Means ve Beklenti Artırımı gibi yumuşak kümeleme yöntemleri kullanılarak gerçekleştirilmektedir. Homojen bir kolektif öğrenme yapısı oluşturmak için her bir alt bölüm ayrı bir Karar Ağacı (*Decision Tree*, KA) ile modellenmiştir. Bir test örneğinin etiketlenmesi için ise tüm uzmanların kararlarını birleştiren yeni bir geçiş fonksiyonu tasarlanmıştır. BA kümeleme algoritması her veri setinde çalışmamaktadır. BA kümeleme işlemi yaparken bazı durumlarda *ill-conditioned* diye ifade edilen kovaryans matrisinin oluşturulamama durumu ortaya çıkmaktadır. Birçok veri setinde karşılaşılabilecek bu durumda GMM (the Gaussian Mixture Model)'nin oluşturulmasında hata meydana gelmekte ve optimizasyon durmaktadır. *k*-means yöntemi, tüm veri setlerine rahatlıkla uygulanabileceği için BA ve GMM'ye göre oldukça avantajlıdır. Bu çalışmanın deneysel sonuçlar içeren ilk hali bir konferansta [7] sunulmuştur. Genişletilmiş ve geliştirilmiş bu çalışmada ise önerilen yöntem ile elde edilen başarılı sonuçlarının

nedenlerine ve hangi durumlarda başarılı olduğunun belirlenmesine odaklanılmıştır. Çalışmamızın geriye kalan kısmında dört bölüm daha vardır. İkinci bölümde önerilen UK yöntemindeki geçiş fonksiyonunun yapısına, çalışma prensibine, uzman seçimine ve önerilen yöntemin zaman karmaşıklığı analizine yer verilmiştir. Üçüncü bölümde pratik uygulama ve deneysel sonuçlara; dördüncü bölümde ise değerlendirmelere ileri çalışmaların neler olabileceğine yer verilmiştir.

2. UK İÇİN YENİ BİR YAKLAŞIM
(A NEW APPROACH FOR MIXTURE OF EXPERTS)

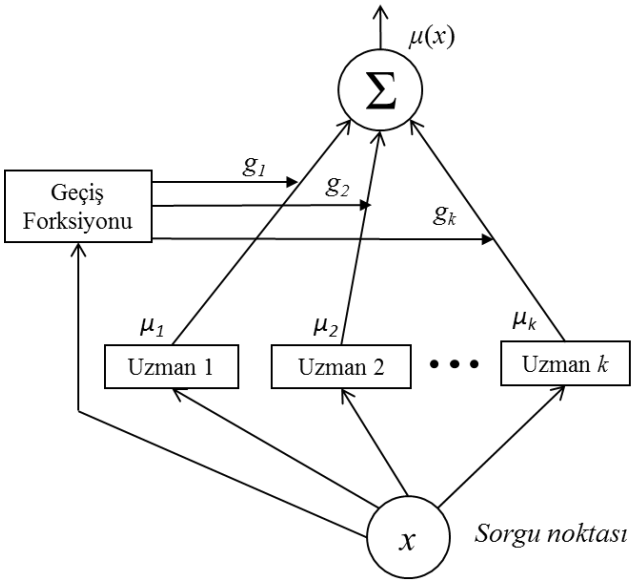
Her hangi bir test noktasının sınıflandırılması yumuşak kümelemeli klasik UK modelinde ait olduğu alt bölgenin uzmanının kararı ile yapılmaktadır. Fakat bu çalışmada bir test noktasının sınıflandırılması, katı kümeleme ile oluşturulan tüm bölgelere ait uzmanların verdikleri kararlarının birleştirilmesiyle yapılmaktadır. Birleştirme işleminde tasarlanan bir geçiş fonksiyonu kullanılmıştır.

2.1. Yeni Bir UK Yaklaşımı ve Geçiş Fonksiyonu
(A New MoE Approach and Gate Function)

Kararların birleştirilmesi işlemi geçiş fonksiyonu adındaki bir mekanizma tarafından yapılmaktadır [8]. Şekil 1'de görüldüğü üzere önerilen yaklaşımın yani UK'nın çalışma mantığı şu aşamalardan oluşur:

1. Veri seti *K* adet alt kümeye ayrılır.
2. Ayrılan her bir alt küme için bir Uzman (*Expert*) atanır ve eğitilir.
3. Gelen *x* test örneğinin etiketlenmesi için tüm alt uzmanlara danışılır ve kararları (μ_i) tek tek alınır.
4. Geçiş fonksiyonu ile tüm kararlar uzmanların test noktasına olan uzaklıklarına bağlı olarak ağırlıklandırılır ($g_i * \mu_i$) ve birleştirilir.
5. Geçiş fonksiyonu test noktasının her bir sınıfa ait olma olasılığı hesaplar.
6. Hesaplanan en yüksek orandaki sınıf türü ile test noktası etiketlenir (*majority voting*).

Çalışmamızda farklı bir uzman tasarımı ve buna bağlı olarak yeni bir geçiş fonksiyonu önerilmiştir. Şekil 1'de görüldüğü gibi geçiş fonksiyonu tarafından, *x* test noktasının etiketlenmesinde her bir uzmanın verdiği karar ağırlıklandırılır. Geçiş fonksiyonun tasarlanmasında yakında olan uzmanların etkisi fazla; uzakta olanların etkisinin az olması gerektiği düşüncesinden yola çıkılarak *Shepard* metodundan [9] yararlanılmıştır. Bir test noktasının sadece en yakında bulunan uzmana danışılarak sınıflandırılması durumunda hatalarının oluşması muhtemeldir. Bu test örneği bulunduğu alt kümenin karar sınırında ise ya da iki alt kümenin orta noktasında ise sınıflandırma işlemi başarısızlıkla sonuçlanabilir. Bu çalışmada bu soruna çözüm aranmış ve bahsedildiği üzere uzmanların uzaklıklarını temel alan yeni bir geçiş fonksiyonu tasarlanmıştır.



Şekil 1. UK'nın çalışma prensibi (Principals of ME)

Şekil 1'de görüldüğü üzere herhangi bir test örneği *x*'in sınıfını bulan $\mu(x)$ fonksiyonunun genel yapışı Eş. 1'de görüldüğü gibidir.

$$\mu(x) = \sum_{i=1}^{K} w_i(x)\mu_i\,(x) \tag{1}$$

$\mu(x)$ fonksiyonun verdiği sınıflandırma kararı $\mu(x) \in [0,1]$, $i=1,\ldots,K$; $j=1,\ldots,C$ şeklindedir. Eş. 1'de kullanılan w_i, *i*. uzmanın hesaplamadaki ağırlığıdır ve Eş. 2'de gösterildiği gibi hesaplanır.

$$w_i = \frac{1}{d_i^2} = \frac{1}{d_i^2(x,E_j)} \tag{2}$$

x test noktası ve E_j arasındaki uzaklık Öklid uzaklık yöntemi Eş. 3 ile bulunmaktadır. E_j, *j*. uzmanın öğrendiği alt kümenin merkez noktasıdır. Merkez noktası bilindiği üzere küme içindeki eğitim örneklerinin aritmetik ortalaması ile bulunmaktadır ve ilgili uzmanı kolektif hesaplamada temsil etmektedir. A, veri uzayın özellik (boyut) sayısıdır ve herhangi bir noktanın özellikleri vektör olarak şu şekilde tanımlanmaktadır: $\langle a_1(x), a_2(x), a_3(x), \ldots a_A(x)\rangle$. Burada $a_r(x)$, *x*'in *r*. özelliğini göstermektedir.

$$d(x,E_j) = \sqrt{\sum_{i=1}^{A}\left(a_i(x) - a_i(E_j)\right)^2} \tag{3}$$

Daha önce Tom Mitchell tarafından önerilen ve üzerinde çalışmalar yapılarak geliştirilen *Distance-weighted-k*-NN sınıflandırıcının [10] çalışma prensibi buraya uyarlanmış ve geçiş fonksiyonu olarak Eş. 4 geliştirilmiştir. g_i, *i*. uzmanın *x* test noktası için hesaplamadaki ağırlığıdır.

$$g_i = \frac{w_i}{\sum_{j=1}^{K} w_j} \tag{4}$$

K'nın alt küme ve aynı zamanda uzman sayısını gösterdiği Eş. 5 her bir uzman için atanan yüzdelik ağırlıklar toplamı yani *G*, 1'e eşittir. Görüldüğü üzere *x* kaydının sınıfı belirlenirken uzaklığa bağlı olarak her bir KA'nın verdikleri kararlar ağırlıklandırılmıştır.

$$G = \sum_{i=1}^{k} g_i = 1 \tag{5}$$

x'in ait olabileceği sınıfların yüzdelik olarak en büyüğü ile etiketleme yapılabilmesi için sınıflandırma fonksiyonu Eş. 6'da, genişletilmiş hali ise Eş. 7'dir.

$$\mu(x) = argmax_{c \in C}\left(g_i . \mu_i^c(x)\right) \tag{6}$$

$$\mu(x) = argmax_{c \in C}\left(\frac{\sum_i^K w_i . \mu_i^c(x)}{\sum_j^K w_j}\right) \tag{7}$$

Burada *C*, veri setindeki tüm sonlu sınıf etiketleri kümesidir ve $\{c_1, c_2, c_3, \ldots, c_s\}$ elemanlarından oluşur. $\mu_i^c(x)$ fonksiyonu ise, *i*. uzmana göre *x*'in *c* sınıfından olma olasılığını hesaplamaktadır. $\mu: R \rightarrow C$ olan μ fonksiyonu tüm sınıf türleri için ayrı ayrı hesaplama yapıp maksimum argümanı sınıf etiketi olarak *x*'e atamaktadır.

2.2. Temel Uzman seçimi (Base Expert Selection)

UK'nın değişik pratik uygulamalarında farklı temel sınıflandırıcılar kullanılmıştır. Çok Katmanlı Algılayıcılar, Destek Vektör Makineleri, *k* En Yakın Komşular yöntemi, Naive Bayes sınıflayıcısı ve lineer ayırıcılar gibi birçok öğrenici uzman olarak tercih edilmiştir. Bu çalışmada temel uzman seçimini, karar ağaçları ile yapılıştır. *k*-means ile oluşturulan *k* tane alt veri setinin her biri ayrı bir veri seti gibi düşünülüp *k* tane KA sınıflandırıcısı inşa edilmiştir. UK uygulamalarında KA kullanımı mantığı ilk olarak Jordan ve Jacobs tarafından anlatılmıştır [11]. Bugüne dek UK yöntemleri üzerinde pek çok çalışma yapılmış ve farklılıkları gösterilmiştir. UK'nın günümüzde olduğu gibi *Radial Basis* fonksiyonlarıyla kullanılmasını ilk olarak Lei Xu önermiştir [12]. Ayrıca KA'lar, kompakt ve sınıflandırmayı kolaylaştırıcı özelliğinden ötürü *Radial Basis* fonksiyon ağlarında TB-RBF (*Decision Tree based Radial Basis Function*) adıyla kullanıldığı da bilinmektedir [13]. Hızlı bir şekilde eğitilebilmeleri, karar verme hızlarının yüksek oluşu ve beyaz kutu (*white box*) yapısındaki özellikleri ile çalışma sistemlerinin kolay analiz edilebilmesi gibi özelliklerinden ötürü KA'lar diğer öğrencilere göre daha avantajlıdır. Kolektif öğrenme yöntemlerinde de genelde temel öğrenici olarak KA'ların tercih edildiği de bir gerçektir [14]. Bu nedenle çalışmamızda karar ağaçları her bir alt bölümde kullanılmak üzere tercih edilmiştir. Karar ağacı algoritmasında, bir ağaç yapısı vardır. Ağacın yaprakları üzerinde sınıflar ve gövdeden yapraklara giden çizgiler üzerinde de işlemeler yer almaktadır. Karar ağacı algoritmasında, ağacın oluşturulması sırasında, üzerinde eğitim yapılan veri seti, özelliklerine göre kümelere bölünür, bu işlem, özyinelemeli bir şekilde (*recursion*) tekrarlanır ve tekrarlama işlemi önemini yitirene kadar sürer. Yapay öğrenmede karar ağaçları iki şekilde kullanılır. Birincisi sınıflandırma ağaçlarıdır ki sorgu noktasını bir sınıfa yerleştirmeyi amaçlar. İkincisi ise Regresyon Ağaçları (*Regression Trees*)'dır ve sorgu noktası için bir sınıf yerine sayısal bir değer bulmayı amaçlar.

2.3. Ağırlıklandırma Yöntemi (Weighting Method)

Eş. 1'de görüldüğü üzere uzmanların kararları uzaklığa bağlı olarak yapılmaktadır. Burada ağırlıklandırma işlemi için kullanılan farklı teknikler vardır: $1/d$, $1/d^2$ ve $1/d^3$. $1/d^2$'li yöntem, $1/d$'ye göre yakındaki öğelerin etkisini çok daha fazla; uzaktakinin etkisini çok daha az almaktadır. $1/d^2$'li geçiş fonksiyonu, yapılan deneysel uygulamalarda diğerlerinden daha başarılı sonuçlar verdiği için tercih edilmiştir.

2.4. Karmaşıklık Analizi (Complexity Analysis)

Uzman olarak inşa edilen bir KA'nın büyük O notasyonuna göre zaman karmaşıklığı Eş. 8 ile açıklanmaktadır [15]:

$$O(A * n * \log n) \tag{8}$$

n örnek sayısını, A ise özellik sayısını göstermektedir. *k*-means kümeleme yönteminin karmaşıklığı ise Eş. 9 ile analiz edilmektedir [16]:

$$O(n * k * I * A) \tag{9}$$

burada *k* küme sayısını, *I* iterasyon sayısını simgelemektedir. Geliştirdiğimiz UK yönteminde *k* tane KA oluşturmanın maliyeti ise Eş. 10 ile hesaplanmaktadır.

$$O\left(k * A * \frac{n}{k} * \log\frac{n}{k}\right) = O\left(A * n * \log\frac{n}{k}\right) \tag{10}$$

Bir veri için yalın bir KA inşa etmek ile *k* tane KA inşa etmek arasındaki zaman karmaşıklığı oranını Eş. 11'de gösterildiği gibi bulunabilir.

$$O\left(\frac{A*n*\log n}{A*n*\log\frac{n}{k}}\right) = O\left(\frac{\log n}{\log\frac{n}{k}}\right) \tag{11}$$

k parametresi her zaman için küçük bir tamsayı olarak alındığından dolayı önemsenmeyebilir. Bu durumda *k* tane KA oluşturmak zaman karmaşıklığı açısından maliyetli bir durum olmadığı sonucuna varılabilir. *I* ve *A* sayılarının her bir uzman için aynı olduğu düşünülüp sabitlenirse, UK yönteminin karmaşıklığı Eş. 12'de gösterildiği şekle dönüşür.

$$O\left(A * n * \log\frac{n}{k}\right) + O(n * k) + O(k) \tag{12}$$

Yukarıdaki zaman karmaşıklığı analizinde ilk iki karmaşıklık *k* tane KA ve *k* tane küme oluşturmayı göstermektedir. En sonda bulunan $O(k)$ karmaşıklığı ise geçiş fonksiyonuna aittir ve *k* küçük bir tamsayı olduğu için önemsenmeyecek etki düzeyindedir.

BA yönteminin karmaşıklık analizi ise [17] Eş. 13'de gösterildiği şekilde olmaktadır.

$$O(I * (A * n + k * n^2)) \quad (13)$$

Görüldüğü üzere iterasyon sayıları (*I*) eşit alınırsa BA'nın karmaşıklığı k-means'e göre oldukça fazladır. Alt bölgelere ayırma işleminin ardından aynı tür sınıflandırıcıların kullanıldığı durumda, alt bölgelere ayırma işleminin etkisi dolayısıyla BA içeren bir yöntemin k-means içeren bir yönteme göre toplam karmaşıklığı fazla olacaktır.

3. DENEYSEL SONUÇLAR (EXPERIMENTAL RESULTS)

Uygulama alanı olarak ortak kullanıma açık UCI (University of California Irvine Machine Learning Respository) [18] veri setlerinde 36 tanesi seçilmiştir. Tüm veri setlerine ait değerler normalize edilmiş, kayıp değerler yer değiştirilmiş, nominal değerler ikili sayısal değerlere dönüştürülerek kullanıma hazır hale getirilmiştir. Tüm algoritmalar MATLAB ortamında, her bir veri seti üzerinde 5x2 çapraz geçerleme ile test edilmiştir. 5x2 kat çapraz geçerleme işlemde 2 kat çapraz geçerleme işlemi 5 defa tekrarlanır. 2 kat çapraz geçerleme işleminde ise eğitim seti rastgele 2 parçaya ayrılır. İlk parça eğitim seti ikinci parça test seti olarak belirlenir. Test setindeki örneklerin yüzde kaçının doğru bilindiği eğitim setine danışılarak hesaplanır. İlk işlem tamamlandıktan sonra test seti eğitim setine; eğitim seti test setine dönüştürülerek yüzdelik başarı tekrar hesaplanır. Bu işlemler 5 defa tekrarlanarak 10 tane yüzdelik başarı sonucu elde edilir. Bu yüzdelik başarıların aritmetik ortalaması çapraz geçerleme işleminin sonucudur. Yapılan uygulamalarda veri setleri *k*-means kümeleme metoduyla (iterasyon sayısı=100) *k* değerleri sırasıyla 2 ile 50 arasında alınarak kümeleme işlemi yapılmıştır. Veri setlerini daha yüksek sayıda kümelere ayırmak hem hesaplama süresini artırmakta hem de seyrek veri setlerinde çok az elemanlı alt kümeler oluşturmaktadır. Bu durumda *k*-means'teki *k* parametresi artırıldığında (veri setindeki eleman sayısına yaklaşıldığında) *distance weighted k*-NN sınıflandırmasına benzer bir yapının ortaya çıktığı gözlemlenmiştir. Aynı veri seti üzerinde iki farklı sınıflandırma işlemi tarafından elde edilen sonuçların istatistiksel anlamlılığını belirleyebilmek için T-Test yöntemi kullanılmıştır. T-Test üç farklı değeri sonuç olarak vermektedir: *win* (başarılı), *loss* (başarısız) ve *tie* (eşit). Bu yöntem sayede önerilen yöntemin başarılı olup olmadığı tespit edilebilmektedir. Tablo 1'de 36 UCI veri seti ile yapılan uygulamaların sonuçları görülmektedir. Geliştirdiğimiz UK yöntemi 3 ayrı sınıflandırıcı ile kıyaslanmıştır. Bunlar 1NN (*En Yakındaki Komşu* algoritması), en iyi sonucu veren *k*-NN (*k – En yakın Komşuluk* algoritması) ve KA sınıflandırıcılarıdır. T-Test sonucuna göre elde edilen başarı, başarısızlık ve eşitlik sayıları Tablo 1'de gözükmektedir. Görüldüğü üzere UK'nın KA ile kıyaslanmasında 11 adet veri kümesinde performans artışı (*win*) sağlanmış, 23'ünde değişme olmamış, sadece 2 tanesinde başarısız olunmuştur. Bu da üzerinde çalışılan yöntemin var olan KA modeline göre daha başarılı sonuçlar elde ettiğini göstermektedir.

Tablo 1. T-Test Sonuçları (T-Test Results)

T-Test Sonucu	UK'yı 1NN ile Karşılaştırma	UK'yı En iyi *k*-NN ile Karşılaştırma	UK'yı KA ile Karşılaştırma
Win	21	11	11
Tie	11	18	23
Loss	4	7	2

Tablo 2'de de görüldüğü üzere T-Test sonucuna göre *audiology* ve *zoo* veri setlerinde, elde edilen başarısızlığın nedeni veri setlerinin yoğunluğu ile alakalıdır. Bu veri kümeleri o kadar küçüktür ki kümelere ayırma işlemi sonucunda oluşan ayrık kümeler içerisinde her bir sınıf etiketini temsil edecek yeteri sayıda örnek bulunmamaktadır. Bu durum oluşturulan alt uzmanların belirli sınıflarda önyargıya sahip olmasına ve doğal olarak genel performansın düşük çıkmasına neden olmaktadır. Tüm veri setleri için oluşturulmuş KA sınıflandırıcısı ile bu uygulamada yaptığımız UK sınıflandırıcısının doğruluk oranları Tablo 2'de verilmektedir. Bu tablo ile tekil bir öğrenci olan KA'nın sınıflandırma performansı ile UK'nın performansı T-Test sonuçlarıyla kıyaslanabilmektedir. İlk sütun tekil KA sınıflandırıcısının veri kümeleri üzerinde verdiği doğruluk değerini; ikinci ve üçüncü sütun ise UK sınıflandırıcısının değerlerini göstermektedir. İkinci sütunda UK için oluşturulan alt küme sayısını yani *k*-means yöntemindeki *k* parametresini; sonraki sütun UK'nın *k* değerine bağlık olarak verdiği doğruluk değerini göstermektedir. Son iki sütunda UK ile KA karşılaştırıldığında yüzdelik artış-azalış ile T-Test sonucu verilmektedir. Yaptığımız UK uygulamasında doğruluk oranının bazı veri setlerinde KA doğruluk oranına göre %10,28'e kadar artış sağladığı gözlemlenmiştir. *k*-means ile 36 adet UCI veri seti üzerinde yapılmış olan bu uygulama yumuşak kümelemeli BA algoritmasıyla tasarlanmış UK modeli ile karşılaştırılmak istenmiştir. Fakat deneysel uygulamalar aşamasında BA'lı UK modeli sadece 4 veri seti üzerinde çalışabilmiştir. Bu nedenle önerilen bu model sadece KA sınıflandırıcısı ile karşılaştırılabilmiştir. Kullanıcı tarafından belirlenen *k* parametresi ile üzerinde çalışılan veri seti için oluşturulacak alt küme sayısı belirlemektedir. *k* değeri tarafından belirlenmektedir ve en uygun değer deneysel sonuçlarla bulunmaktadır. Önerilen bu modelde parametre bağımlılığı olumsuz bir durum olarak algılansa da klasik UK yöntemlerinde de parametre gerekliliğini hatırlatmak isteriz. En uygun *k* değerinin otomatik olarak bulunmasıyla alakalı olarak değişik çalışmalar yapılmıştır [19, 20]. Ayrıca başka bir çalışmada SOM (Kohonen's Self Organized Map) yönteminden yararlanılarak en uygun *k* değeri bulunmaya çalışılmıştır [21]. Değişik k parametreleriyle yapılan deneysel çalışmalarda yoğun bir yapıdaki veri setinin fazla sayıda alt kümelere; seyrek bir yapıdaki veri setinin de az sayıda kümelere bölmenin toplam sınıflandırma başarısını artırdığı gözlemlenmiştir. Burada yoğunluk ölçütü için Eş. 14'te verilen metrik kullanılmıştır.

$$Yoğunluk = \frac{N}{A \times C} \quad (14)$$

Tablo 2. UCI Test Sonuçları (UCI Test Results)

Veri seti Adı	Örnek Sayısı	Sınıf Sayısı	Öznitelik Sayısı	KA Doğruluk Oranı	UK		KA ile UK'yı karşılaştırma	
					k	Doğruluk Oranı	Yüzdelik fark	T-Test Sonucu
abalone	4153	19	10	0,2122	50	0,2231	5,11	*win*
anneal	890	4	62	0,9883	2	0,9879	-0,05	*tie*
audiology	169	5	69	0,8615	2	0,8166	-5,22	*loss*
autos	202	5	71	0,6614	2	0,6426	-2,84	*tie*
balance-scale	625	3	4	0,7811	30	0,8614	10,28	*win*
breast-cancer	286	2	38	0,6692	20	0,7028	5,02	*win*
breast-w	699	2	9	0,9465	4	0,9536	0,76	*tie*
col10	2019	10	7	0,7575	3	0,7559	-0,21	*tie*
colic	368	2	60	0,8092	2	0,8212	1,48	*tie*
credit-a	690	2	42	0,8197	4	0,8533	4,10	*win*
credit-g	1000	2	59	0,6870	20	0,7140	3,93	*win*
d159	7182	2	32	0,9698	2	0,9610	-0,90	*tie*
diabetes	768	2	8	0,7036	10	0,7375	4,81	*win*
glass	205	5	9	0,6527	5	0,6615	1,35	*tie*
heart-statlog	270	2	13	0,7415	35	0,8015	8,09	*win*
hepatitis	155	2	19	0,7923	4	0,8103	2,28	*tie*
hypothyroid	3770	3	31	0,9943	2	0,9932	-0,11	*tie*
ionosphere	351	2	33	0,8547	35	0,8991	5,20	*win*
iris	150	3	4	0,9333	2	0,9400	0,71	*tie*
kr-vs-kp	3196	2	39	0,9901	2	0,9838	-0,63	*tie*
labor	57	2	26	0,8596	2	0,8526	-0,82	*tie*
letter	20000	26	16	0,8234	2	0,8203	-0,38	*tie*
lymph	142	2	37	0,7901	3	0,7775	-1,60	*tie*
mushroom	8124	2	112	1,0000	2	0,9997	-0,03	*tie*
primary-tumor	302	11	23	0,4265	2	0,4351	2,02	*tie*
ringnorm	7400	2	20	0,8842	10	0,9375	6,03	*win*
segment	2310	7	18	0,9436	2	0,9474	0,39	*tie*
sick	3772	2	31	0,9824	2	0,9846	0,22	*tie*
sonar	208	2	60	0,7000	5	0,7231	3,30	*tie*
soybean	675	18	83	0,8824	2	0,8809	-0,17	*tie*
splice	3190	3	287	0,9254	2	0,9066	-2,03	*tie*
vehicle	846	4	18	0,6849	5	0,6835	-0,21	*tie*
vote	435	2	16	0,9361	3	0,9549	2,01	*tie*
vowel	990	11	11	0,6873	9	0,7143	3,94	*win*
waveform	5000	3	40	0,7394	35	0,8051	8,89	*win*
zoo	84	4	16	0,9762	2	0,9214	-5,61	*loss*

Tablo 3. Yoğun ve Seyrek veri setleri (Dense and Sparce Datasets)

					Yüzdelik Değişiklik		
	N	*C*	*A*	Yoğunluk	*k*=5	*k*=10	*k*=20
balance-scale	625	3	4	52,1	3,77	5,29	5,53
breast-w	699	2	9	38,8	1,36	1,33	1,48
waveform	5000	3	40	41,7	0,44	3,24	6,31
autos	202	5	71	0,2	-11,68	-21,11	-30,84
colic	368	2	60	0	-0,53	-5,17	-9,53
labor	57	2	26	0,5	-11,42	-11,83	-9,38

N, *A* ve *C* değerleri sırasıyla veri setindeki örnek sayısı, öznitelik sayısı ve sınıf sayısıdır. Tablo 3'de ilk 3 verinin yoğunluğu olukça fazla, son üç veri setinin de yoğunluğu oldukça azdır. *k*=3 alındığında elde edilen doğruluk başarı değerlerine göre *k*'nın 5, 10 ve 20 alındığında elde edilen başarı değerleri arasında bir karşılaştırma yapılmıştır. Görüldüğü üzere yoğun veri setlerde *k* değerinin artmasıyla daha yüksek başarı, azalmasıyla da daha düşük başarılar elde edilmiştir. Ayrıca KA ve UK sınıflandırıcı sistemlerinde oluşturulan karar ağaçlarının içyapılarındaki

düğüm sayıları da ayrı bir karmaşıklık ölçütü olabilir. Bu nedenle KA ve UK sınıflandırıcılarının içerisinde bulunan düğüm sayıları Tablo 4'de verilmiştir. UK'da *k* değeri 5 alındığında karar ağaçlarında oluşan toplam düğüm sayısı 8662, KA'lardaki toplam düğüm sayıları ise 7368'dir. Birbirine çok yakın olan bu değerler önerilen yöntem ile var olan KA yönteminin karmaşıklık açısından birbirine benzediğini göstermektedir.

Tablo 4. KA ve UK'daki düğüm sayıları
(Number of nodes in DT and MoE)

Veriseti	KA düğüm sayısı	UK düğüm sayısı
abalone	1515	1502
anneal	21	34
audiology	37	42
autos	51	58
balance-scale	69	74
breast-cancer	63	58
breast-w	37	22
col10	295	396
colic	47	50
credit-a	93	106
credit-g	209	198
d159	205	462
lymph	21	28
mushroom	27	48
diabetes	139	138
glass	47	66
heart-statlog	35	66
hepatitis	19	30
primary-tumor	81	70
ringnorm	531	600
segment	91	130
sick	53	76
sonar	35	44
soybean	81	120
splice	167	220
vehicle	151	170
vote	19	26
vowel	191	240
waveform	737	620
zoo	7	10
hypothyroid	23	78
ionosphere	37	46
iris	9	20
kr-vs-kp	73	202
labor	9	12
letter	2413	2600
TOPLAM	7638	8662

4. SONUÇLAR VE TARTIŞMALAR
(RESULTS AND DISCUSSIONS)

Bu çalışmada denetimsiz öğrenme ile veri seti kümelere bölünmüş ve denetimli bir öğrenme gerçekleştirilmiştir. Bu durum önerilen yönteme ait hesaplama süresinin tekil öğrenicilere göre artırmasını sağlamıştır. *k*-means ile yapılan kümeleme işleminde her defasında başka küme grupları oluştuğu için sınıflandırma başarısı kümelerin yerleşimine, eleman sayısına ve yapısına bağlı olarak az da olsa değişiklik göstermektedir. Fakat yapılan T-Test analizlerinde hemen hemen aynı sayıda elde edilen *win*, *tie*, *loss* çıktılarından dolayı bu değişiklik önemsenmeyecek düzeydedir. Ayrıca *k*-means metodu iki boyutlu veri kümelerinde dairesel; üç boyutlu ortamda da küresel tarzda kümeleme yapmaktadır. Bu nedenle oluşan alt kümelerdeki örnekler o kümenin merkez noktası etrafında saçılmış bir şekilde bulunmaktadır. Kısaca *k*-means metodunda oluşturulan kümeler iç içe veya sarmal yapıda olmamaktadır. Bu özellik, UK'da uzaklığa bağlı olarak ağırlıklandırma ile sınıflandırma yapan geçiş fonksiyonu için çok uygun olmaktadır ve yüksek başarı elde edilmesini sağlamaktadır. Her ne kadar literatürde [20] BA algoritmasının uzaklık tabanlı *k*-means tarzı algoritmalara göre daha iyi bir kümeleme yaptığı belirtilse de, kovaryans matrisinin her veri seti için oluşturulamaması önemli bir dezavantaj olduğu kabul edilmektedir. Bu durumda k-means metodunun her veri setine için uygulanabilir olması büyük bir avantaj olarak karşımıza çıkmaktadır. Her hangi bir test noktası, ait olduğu alt kümenin sınırına (*class boundary*) ya da karar sınırına (*decision boundary*) çok yakın bir yerde ise sınıflandırma işleminin nasıl gerçekleşeceği konusu çalışmamızın başında merak edilmiş; bu durumda hangi alt kümenin uzmanı ile etiketleneceği sorusu cevaplanmak istenmişti. Uzaklığa bağlı olarak sınıflandırma yapan geçiş fonksiyonu sayesinde bu sorun ortadan kaldırılmıştır. Sınıflandırma işleminde eğitim setindeki tüm uzmanlara yer verilmiştir. Fakat diğer uzmanlara göre test noktasından daha uzakta bulunan uzmanların etkisi $1/d^2$'den dolayı çok az olmaktadır. Uzaktaki uzmanların mevcut test noktası hakkındaki kararlarının belirlenmesi ve kararlarının geçiş fonksiyonunda hesaplamaya dâhil edilmesi hesaplama süresini artırmaktadır. Bu durumda sadece test noktasının çevresinde ve yakınında bulunan uzmanları sınıflandırma işlemine almak için otomatize edilmiş bir sistem üzerinde çalışma yapılabilir. Ayrıca bu sınıflandırma mekanizmada *k* parametresi kullanıcıdan istenmektedir. Parametrenin kullanıcı tarafından belirlemesi olumsuz bir durum olarak algılansa da diğer birçok sınıflandırma ve kümeleme yönteminin parametreye dayalı çalıştığını unutmamak gerekir. Elbette ki denetimli ve denetimsiz öğrenme yöntemleri için en uygun parametrenin seçimine yönelik birçok bilimsel çalışma vardır. Bu çalışmalardan da yararlanılarak performans artırımı sağlanabilir. *abalone*, w*aveform* ve r*ingnorm* gibi bazı UCI veri setlerinde uzman sayısı artırıldıkça genel sınıflandırma başarısının da arttığı gözlemlenmiştir. Bu durumun bazı dezavantajları vardır. Öncelikle hesaplama süresi bir miktar artmaktadır. Ayni zamanda uzman sayısının artırılması alt kümeler içerisinde kalan eleman sayısının azalması anlamına gelmektedir. Bu durum alt kümelerin saflaşmasına, üzerinde çalıştığımız yöntemin *k*-NN sınıflandırıcısına benzer bir yapıya dönüşmesine sebep olmaktadır. Diğer bir deyişle *k*-means'teki *k* parametresi eleman sayısına yani *N*'e yaklaştıkça sınıflandırma işlemi $1/d^2$-distance-weighted-kNN ile yapılmış gibi olacaktır. Bu durum da hesaplama süresinin fazla olmaması ve *k*-NN benzeri bir

sınıflandırıcının oluşmaması için k değeri N'den oldukça küçük seçilmelidir. Veri seti, *k* adet kümeye bölündüğünde bazı alt kümelerdeki örnek sayısının azaldığını ve bu şekilde alt kümelerin saflaştığı yani tek sınıf türünden örnekler içerdiği gözlemlenmiştir. Bu durum, alt kümenin belirli bir sınıf hakkında önyargılı (*bias*) karar vermesine sebep olmuştur. Bu durumun denetimli öğrenmede olumsuz bir durum olarak tanımlansa da bu çalışmada performans artırımı için olumlu bir faktör olmuştur. Yapılan uygulamalarda, *balance-scale*, *heart-statlog* ve *waveform* gibi bazı veri setlerinde oluşturulan her bir alt bölgenin yoğun olduğu ve aynı veri setindeki diğer alt bölgelerden doğrusal karar çizgileriyle ayrıldığı gözlemlenmiştir. Bu tür veri setlerde yüksek başarı elde edilmiştir. Üzerinde uygulama yapılan diğer veri setlerde geliştirilen bu yöntemin neden başarılı ya da başarısız olduğu üzerinde çalışmalar yapılabilir. Bu amaçla bir veri setinin istatistiksel özelliklerinin yanı sıra geometrik meta öznitelikleri [22] de ele alınarak Meta Öğrenme ile başarının ya da başarısızlığın nedenleri incelenebilir. Veri setlerine ait meta özellik çıkarımı ile alakalı yapılan bazı çalışmalar vardır [23] ve bu çalışmalarda hangi özelliklerin sınıflandırma başarısına etki ettiği incelenmiştir. Ayrıca iki boyutlu daha çok yapay veri seti üretilerek hangi veri setlerinde geliştirilen yöntemin daha iyi çalıştığı Kovaryans Matrisi yardımıyla da anlaşılmaya çalışılabilir. İleri aşamalarda, uzman olarak KA'nın haricinde farklı sınıflandırıcılar ve farklı kümeleme yöntemleri de kullanılarak performans karşılaştırmaları yapılabilir. Ayrıca BA ile yapılan UK çalışmaları ile üzerinde çalıştığımız yöntem değişik açılardan kıyaslanabilir.

5. SONUÇLAR (CONCLUSIONS)

Bir veri setinin bazı bölgelerinin diğerlerinden farklı özelliklere sahip olması genel sınıflandırma performansını düşürebilir. Bu nedenle bu bölgeleri diğerlerinden ayırmak gerekebilir ve bu bölgeler için özel çözümlerin sunulması zorunlu olabilir. Bu amaçla veri setinde önce kümeleme yapılmış, sonra her bir bölge için ayrı bir uzman eğitilmiş, ardından bu uzmanlar bir geçiş fonksiyonuyla birleştirilerek uzman karışımı bir sistem elde edilmiştir. Bu çalışmamızda tüm veriye tek bir model belirlemek yerine, veriyi alt bölgelere ayırıp her biri için farkı modeller belirlemenin sınıflandırma performansını genel olarak artırmaktadır. Geleneksel yumuşak kümelemeli UK modelinden farklı olarak sunduğumuz bu çalışma, tüm veri setlerine uygulanabilmesi açısından oldukça avantajlı bir durumdadır.

KAYNAKLAR (REFERENCES)

1. Jacobs R, Jordan M, Nowlan J, Hinton G.E., Adaptive mixtures of local experts Neural Computer, 3 (1), 79–87, 1991.
2. Alpaydın E, Mixture of Experts, Introduction to Machine Learning E-Book, MIT Press, A.B.D., 301-305, 2014.
3. Yuksel E., Twenty Years of Mixture of Experts, IEEE Transactions On Neural Networks And Learning Systems, 23 (8), 1177-1193, 2012.
4. Akben S.B., Alkan A., Density-Based Feature Extraction to Improve the Classification Performance in the Datasets Having Low Correlation Between Attributes, Journal of the Faculty of Engineering and Architecture of Gazi University, 30 (4), 597-603, 2015.
5. Dempster A.P., Laird N.M., Rubin D.B,. Maximum Likelihood from Incomplete Data viathe EM Algorithm. Journal of the Royal Statistical Society, Series B (39), 1-38, 1977.
6. Chamroukhi, F., Non-Normal Mixtures of Experts, arXiv preprint arXiv:1506.06707, Fransa, 1-61, 2015.
7. Bulut F., Amasyalı M.F., Sönmez A.C., Uzman Karışımlarında Yeni Bir Yaklaşım: Uzman Kararlarının Yeni Bir Geçiş Fonksiyonuyla Birleştirmesi, Akıllı Sistemlerde Yenilikler ve Uygulamaları, ASYU-INISTA 2014, İzmir, 62-66, 2014.
8. Nguyen H.D., McLachlan G.J., Laplace mixture of linear experts, Computational Statistics & Data Analysis, 93 (1), 177-191, 2016.
9. Shepard D., A Two-dimensional interpolation function for irregularly spaced data, Proceedings of the 23rd National Conference of the ACM, A.B.D., 517-523, 1968.
10. Bulut F., Bucak İ.Ö., An urgent precaution system to detect students at risk of substance abuse through classification algorithms, Turkish Journal of Electrical Engineering & Computer Sciences, 22 (3), 690-707, 2014.
11. Jordan M.I., Jacobs R.A., Hierarchical Mixture of Experts and the EM Algorithm, Journal Neural Computation archive, 6 (2), 181-214, 1994.
12. Xu L., RBF nets, Mixture Experts and Bayesian Ying-Yang Learning, Elsevier Neurocomputing 19, 223-257, 1998.
13. Kubat M., Decision Trees Can Initialize Radial-Basis Function Networks, IEEE Transactıon On Neural Networks, 9 (5), 813-821, 1998.
14. Zhou Z.H., Ensemble Methods: Foundations and Algorithms, CRC Press, A.B.D., 270-272, 2012.
15. Witten I.H., Frank E., Data Mining: Practical Machine Learning Tools and Techniques, 3rd Edition, Elsevier, 196-198, 2011.
16. Myatt G.J, Making Sence of Data: A Practical Guide to Exploratory Data Analysis and Data Mining, Wiley, A.B.D., 120-129, 2007.
17. Moore E.J., Bourlai T., Expectation Maximization of Frequent Patterns, a Specific, Local, Pattern-Based Biclustering Algorithm for Biological Datasets, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 13 (5), 812-824, 2016.
18. Bache K., Lichman M., UCI Machine Learning Repository, URL: http://archive.ics.uci.edu/mlIrvine, CA: University of California, School of Information and Computer Science, 2014.

19. Özturk C., Hancer E., Karaboga D., Automatic Clustering With Global Best Artificial Bee Colony Algorithm, 29 (4), 677-687, 2014.

20. Bayá A.E., Granitto P.M., How Many Clusters: A Validation Index for Arbitrary-Shaped Clusters in IEEE/ACM Transactions on Computational Biology and Bioinformatics, 10 (2), 401-414, 2013.

21. Hsu C.C., Lin S.H., Visualized Analysis of Mixed Numeric and Categorical Data Via Extended Self-Organizing Map in IEEE Transactions on Neural Networks and Learning Systems, 23 (1), 72-86, 2012.

22. Ho T.K., & Basu M., Complexity Measures of Supervised Classification Problems, IEEE Transaction on Pattern Analysis and Machine Intelligence, 24 (3), 289-300, 2002.

23. Basu M., Data Complexity in Pattern Recognition, Springer Science & Business Media, Londra, İngiltere, 2010.