BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GEN İFADE VERİLERİ İLE İŞLEMSEL KANSER SINIFLANDIRILMASI

NAMIK BARIŞ İDİL

YÜKSEK LİSANS TEZİ
2009

GEN İFADE VERİLERİ İLE İŞLEMSEL KANSER SINIFLANDIRILMASI

OPERATIONAL CANCER CLASSIFICATION USING GENE EXPRESSION DATA

NAMIK BARIŞ İDİL

Başkent Üniversitesi
Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin
BİLGİSAYAR Mühendisliği Anabilim Dalı İçin Öngördüğü
YÜKSEK LİSANS TEZİ
olarak hazırlanmıştır.
2009

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma, jürimiz tarafından **BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI 'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Başkan	:….....
	Prof. Dr. Ziya AKTAŞ
Üye (Danışman)	:….....
	Doç. Dr. Nizami GASILOV
Üye	:….....
	Doç. Dr. Fatih ÇELEBİ

ONAY

Bu tez 08/06/2009 tarihinde, yukarıdaki jüri üyeleri tarafından kabul edilmiştir.

..../06/2009

Prof.Dr. Emin AKATA

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

TEŞEKKÜR

Yazar, bu çalışmanın gerçekleşmesindeki katkılarından dolayı, aşağıda adı geçen kişi ve kuruluşlara içtenlikle teşekkür eder.

Sayın Doç. Dr. Nizami GASILOV'a (tez danışmanı), çalışmanın sonuca ulaştırılmasında ve karşılaşılan güçlüklerin aşılmasında her zaman yardımcı ve yol gösterici olduğu için…

Sayın Yrd. Doç. Dr. Hasan OĞUL'a, algoritmaların seçimi ve analizleri konusundaki tüm çabaları ve yardımları için...

Sayın Arş. Gör. Mehmet DİKMEN'e, çalışmanın yönlendirilmesi ve kaynak bulma konusundaki yardımları için...

ÖZ

GEN İFADE VERİLERİ İLE İŞLEMSEL KANSER SINIFLANDIRILMASI

Namık Barış İDİL
Başkent Üniversitesi, Fen Bilimleri Enstitüsü,
Bilgisayar Mühendisliği Anabilim Dalı

Son yıllardaki bilgisayar teknolojilerinde elde edilen gelişmeler, özellikle işlemci gücünün artması, önceleri gerçekleştirilebilen sade, doğrusal modeller yerine fiziksel ve gerçek olayları daha iyi yansıtan; ama daha fazla bellek ve zaman gerektiren doğrusal olmayan modellerin kullanılmasına imkan yaratmıştır.

Bu çalışma, A. Statnikov'un, mikrodizi gen ifade verileri kullanarak çok kategorili kanser sınıflandırması ile ilgili çalışması ve bu çalışmadan elde edilmiş sonuçlar üzerine önerilmiş olan optimizasyon çalışmalarını kapsamaktadır [1]. Mikrodizi analizi ile elde edilmiş gen ifade verilerinin üzerinde, destek vektör makinesi ile analiz edilmeden önce, doğrusal ve doğrusal olmayan indirgeme yöntemleri kullanılarak, verilerin eğitilme ve test sürecinin hızlandırılması amaçlanmıştır. Uygulanması amaçlanan indirgeme yöntemleri, bir dizi algoritmanın yanı sıra, bu algoritmaların probleme yönelik yeni yorumlamalarıyla yapılmış, daha sonra bu yöntemler karmaşıklık, kaynak kullanımı ve indirgeme performansı göz önünde bulundurularak test edilmiştir. Böylece, eğitim ve test işlemlerinin performans ve başarı oranlarını kabul edilebilir düzeyin üstünde tutmak koşuluyla, veri kümelerindeki nitelik sayısını küçülterek, işlem hızının arttırılması amaçlanmıştır.

Yapılan testlerin sonucunda, gen ifade verilerinin bulunduğu veri kümesi üzerinden yapılan Bağımsız Bileşen Analizi (BBA), Çekirdek Temel Bileşen Analizi (ÇTBA), İz Düşümü Takip Analizi (İDTA) indirgeme algoritmaları üzerine oluşturulmuş programların, veri kümesindeki nitelik sayısının aşırı yüksek olmasından dolayı kilitlendiği ya da hafıza yetersizliğinden dolayı olağandışı sonlandırıldığı tespit edilmiştir. Diğer algoritmalar olan Temel Bileşen Analizi (TBA), Doğrusal Olmayan Temel Bileşen Analizi (DOTBA), Kendi Düzenlenen Haritalar (KOH), Doğrusal Diskriminant Analizi (DDA) ve Korelasyon Analizi (KA) ile yapılan nitelik indirgemeleri sonucu, karar destek vektör makinesinin eğitim sürelerinin değişken olarak azaldığı görülmüştür. Buna dayanarak, çalışmada kullanılan veri kümesinin içerdiği niteliklerin büyük bir kısmının, veri kümesinin destek vektör makinesindeki

eğitim ve test performansına çok az etkisi olduğu, ayırt edici özellikler taşımadığı veya bazı niteliklerin bir araya gelerek, tüm kümeyi temsil edebilen bir alt grup oluşturabildiğini, bu yüzden etkisiz niteliklerin ya da nitelik alt gruplarının indirgeme algoritmaları kullanılarak orijinal veri kümesinden çıkarılmasının, maliyet ve süre açısından yararlı olacağı anlaşılmıştır.

ANAHTAR SÖZCÜKLER: aritmetik ortalama, bağımsız bileşen analizi, çekirdek bileşen analizi, destek vektör makinesi, DNA mikrodizi, doğrusal olmayan temel bileşen analizi, doğrusal olmayan öğrenim sistemleri, GEMS, gen ifade verileri, iz düşüm takip analizi, kanser, kendi düzenlenen haritalar, korelasyon analizi, nitelik çıkarımı, nitelik indirgeme, standart sapma, temel bileşen analizi.

Danışman: Doç. Dr. Nizami GASILOV, Başkent Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü.

ABSTRACT

OPERATIONAL CANCER CLASSIFICATION USING GENE EXPRESSION DATA

Namık Barış İDİL

Başkent University, Institute of Science,

The Department of Computer Engineering

Recent improvements in computer technologies, especially significant increase in processing power of central processing units, leads to usage of non – linear models which represents physical and abstract problems better but require more memory and time, instead of simple, linear models.

This study focuses on A. Statnikov's article about multicategory cancer classification using of microarray gene expression data and optimization suggestions [1]. Before the training of support vector machines with the gene expression data which is gathered by microarray analysis, it is intented to accelerate the training and test speed process with both linear and non – linear reduction methods. Reduction methods which are intented to be used are both implemented by using some algorithms and new interpretation of these algorithms. After that, these methods are tested according to their complexity, resource allocation and reduction performance. Therefore, by keeping the performance and success ratios of training and testing process above an acceptable treshold, it is intented to reduce the feature size in data sets as it will also increase the overall speed of the process.

The results of the test show that, Independent Component Analysis (ICA), Kernel Principle Component Analysis (KPCA), Projection Pursuit Analysis (PPA) reduction algorithms used on data set failed to give any results due to excessive amount of features in data set by either locking down or terminating itself.

With the usage of other algorithms which are Principle Component Analysis (PCA), Non – Linear Principle Component Analysis (NLPCA), Self Organizing Maps (SOM), Linear Discriminant Analysis (LDA) and Correlation Analysis (CA), it is observed that the training and testing process times of the support vector machine is reduced variably. Taking this into consideration, most of the the features of the data set which is used in this study do not have any differentiative

property and therefore have low - level of effect on the training and testing of the support vector machine. On the other hand, some features may become high – level effective when combined together and form a sub group feature sets. So, by eliminating low – level effective features and revealing high – effective sub group features by feature selection and feature reduction, a significant improvement in both cost and time consume can be established.

KEYWORDS: cancer, correlation analysis, DNA microarray, empirical mean, feature reduction, feature selection, GEMS, gene expression data, independent component analysis, kernel principle component analysis, non – linear learning systems, non – linear principle component analysis, principle component analysis, projection pursuit analysis, self organizing maps, standard deviation, support vector machines.

Supervisor: Ass. Prof. Dr. Nizami GASILOV, Başkent University, Engineering Faculty, Computer Engineering Department.

İÇİNDEKİLER LİSTESİ

Sayfa

İÇİNDEKİLER LİSTESİ

Sayfa

İÇİNDEKİLER LİSTESİ

Sayfa

İÇİNDEKİLER LİSTESİ

Sayfa

ŞEKİLLER LİSTESİ

Şekil **Sayfa**

ÇİZELGELER LİSTESİ

Çizelge **Sayfa**

SİMGELER VE KISALTMALAR LİSTESİ

SVM	support vector machines
DVM	destek vektör makineleri
OVR	one versus rest
OVO	one versus one
GB	Giga Byte
Mhz	Mega Hertz
CA	Correlation Analysis
PCA	Principle Component Analysis
ICA	Independent Component Analysis
KPCA	Kernel Principle Component Analysis
LDA	Linear Discriminant Analysis
SOM	Self Organizing Maps
PPA	Projection Pursuit Analysis
NLPCA	Non Linear Principle Component Analysis
KA	Korelasyon Analizi
TBA	Temel Bileşen Analizi
BBA	Bağımsız Bileşen Analizi
ÇTBA	Çekirdek Temel Bileşen Analizi
DDA	Doğrusal Diskriminant Analizi
KOH	Kendinden Organize Haritalar
İDTA	İz Düşüm Takip Analizi
DOTBA	Doğrusal Olmayan Temel Bileşen Analizi
KKB	Kalana Karşı Bir
BKB	Bire Karşı Bir

1. GİRİŞ

Çağımızın vebası olarak görülen ve henüz tam bir tedavisi geliştirilememiş olan kanser hastalığı, tüm dünya toplumlarını her geçen gün daha da tehdit eder düzeye gelmiştir. Türk Kanser Araştırma ve Savaş Kurumu Derneği'nin son verilerine gore, dünyada 20 milyondan fazla kanser hastası bulunmakta ve her yıl 10 milyondan fazla yeni hasta tespit edilmektedir [2].

Kanser hastalığının kabul edilmiş dört evresi bulunmaktadır ve kanser ne kadar erken evredeyken farkedilirse, tedavi şansı da o derecede artmaktadır. Bu yüzden, kanser hastalığında erken teşhis çok önemlidir.

Gen ifade verileri ile kanser teşhis ve sınıflandırma işlemleri bu noktada büyük önem kazanmaktadır. Bir dokudan alınan tümörün iyi huylu ya da kötü huylu olduğunun anlaşılması, kötü huylu ise once tümörün tipinin, daha sonra bu tümörün alt tipinin belirlenmesi, son olarak da kanserin evresinin belirlenmesi hem zor, hem maliyetli, hem de zaman isteyen süreçlerdir.

Destek Vektör Makineleri ve gen ifade verileri kullanılarak kanser teşhisi daha once A. Statnikov tarafından yapılmış ve başarılı sonuçlara ulaşılmıştır [1]. Öte yandan, kullanılan veri kümelerinin boyutlarının çok yüksek olması işlem hızının düşük olmasına sebep olmuştur.

Kullanılan veri kümesinde bulunan gen örnekleri için kaydedilmiş tüm niteliklerin, kanser teşhisi ve sınıflandırması için kullanılıp kullanılmadığı belirsiz bir durumdur. Eğer gen ifade verilerinden oluşmuş bir veri kümesindeki niteliklerden çıkarılabilme potansiyeli bulunanlar elenebilirse, bu karar destek makinesinin eğitim ve test hızını arttırabilir, ki kanser teşhisi söz konusu olduğunda bu oldukça önemli bir gelişmedir.

Nitelik indirgeme algoritmaları kullanılarak veri niteliklerinden bazılarının veri kümesinden çıkarılması sonucunda bir takım olası sonuçlar ortaya çıkabilir. Öncelikle, bir nitelik, kanser sınıflandırmasına hiçbir etkide bulunmuyor olabilir. Bu durumda, bu veri niteliğinin kümeden çıkarılması eğitim ve test işlemlerini hızlandıracak ve hiçbir performans kaybı yaşanmayacaktır. Diğer bir olası durum ise, bir nitelik, sınıflandırma ve teşhis üzerinde az bir etkiye sahip olabilir. Bu

durumda ise, bu veri niteliğinin kümeden çıkartılması yine eğitim ve test işlemlerini hızlandıracak; ancak az da olsa perfomans kaybına yol açacaktır. Bunun kabul edilebilir olup olmaması, test sonucundaki performans kaybına gore değerlendirilebilir. Bir diğer olasılık ise, bir veri niteliğinin, tüm veri kümesi için büyük bir etki yaratmasıdır ki, böyle bir veri niteliği çekirdek nitelik olarak görülebilir. Bu niteliği kümeden çıkartmak, büyük performans kayıplarına yol açacağından, kabul edilebilir olmayacaktır. Son olarak, bir veri niteliği gürültü, yani tüm veri kümesini kötü olarak etkileyen bir nitelik olabilir. Böyle bir durumda, bu niteliği çıkartmak hem eğitim ve test işlemleri hızlanacak, hem de performans artacaktır.

Veri nitelikleri bu tür etkiler dışında, toplu olarak farklı durumlar ortaya çıkarabilir. Örneğin, bir veri niteliği tek başına performansa çok az bir etki yaratırken, diğer niteliklerle birleştiğinde tüm veri kümesini temsil edecek bir alt grup oluşturulabilir. Bu değişik ve farklı algoritmaların denenmesini gerektirecek bir durumdur.

Tez çalışmasında yapılmak istenilen, veri kümesinin eğitimine ve testine etkisi olmayan, az etkileyen ya da kötü yönde etkileyen nitelikleri bulup, bunları kümeden çıkarmak ya da veri kümesini temsil edebilecek alt gruplar oluşturmaktır; ancak nitelik çıkarma ya da yeniden oluşturma işlemleri sırasında, performansın kabul edilebilir düzeyin altına düşmemesine dikkat edilmelidir.

2. VERİ

Veri sembolik olan veya olmayan, sürekli veya ayrık, geniş boyutlu veya küçük her şekilde olabilir. Verileri barındıran bir küme, yüksek bir hacme ulaştığında, bu küme içerisinden istenilen verileri ayıklamanın yanı sıra, çeşitli sorgulamalar da yapabilmek için gereken yöntemleri sağlamak ancak etkili algoritmalar ile sağlanabilmektedir. Veri analizi teknikleri neredeyse bütün çalışma alanlarında kullanışlı olmakla birlikte, biyoenformatik alanında mikrodizi gen ifadesi verilerinin yanı sıra, gen dizilim verilerinin de ayıklanmasında büyük bir öneme sahiptir.

Verilerin bir çoğu boyutsal, yapısal, anlamsal ve şartsal olarak birbirinden farklıdır. Bazıları analiz öncesi bazı işlemler gerektirirken, bazıları doğrudan analiz edilebilir. Farklı durumlar ve farklı veri şekilleri için farklı algoritmalar vardır. Astronomi ve uzay araştırmaları için iyi sonuç veren bir algoritma, gen/protein verilerinin çıkarılması için hiç işe yaramayabilir. Bir fabrikanın sahip olduğu istatistik verileri kullanılan bir algoritma, bir internet sitesinin istatistik verileri için aynı performansı vermeyebilir. Buna bağlı olarak her algoritmanın farklı çalışma parametreleri, farklı ön şartları ve bunlarla ilişkili farklı avantajları ve dezavantajları vardır.

2.1 Veri Tipleri

Genel olarak veri matrisi üç tür olarak sınıflandırılır:

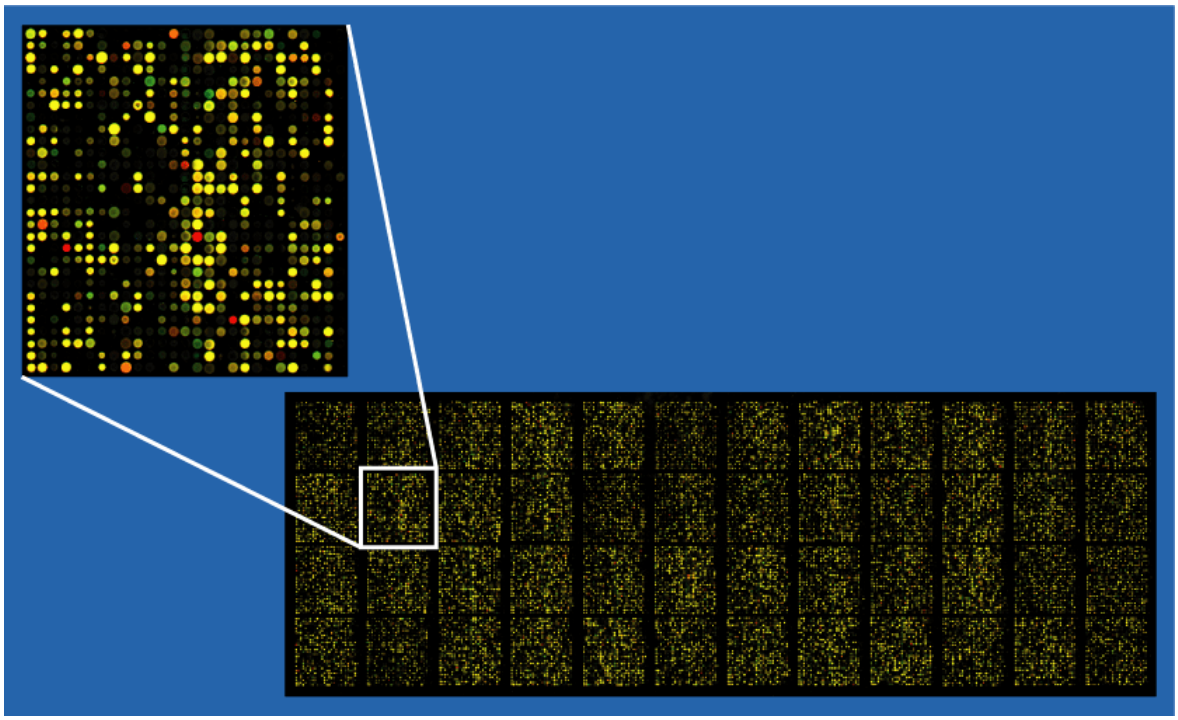
- Sürekli veriler (zaman dizileri), örneğin, bir hastanın her iki saatte bir alınan kan şekeri düzeyleri,
- Parametrik veriler (verinin ait olduğu varlıkla bir ilişkisi olduğu durumlar), örneğin, nüfus veritabanı (boy, ağırlık, görme keskinliği, deri rengi, yaş, saç rengi, göz rengi, vb.),
- Parametrik olmayan veriler (örnekler arasında hiçbir ilişki olmadığı durumlar), örneğin, gen ifade verileri, beş ayrı dersten öğrencilerin aldığı notlar, vb. [3]

3. MİKRODİZİ TEKNOLOJİSİ VE GEN İFADE VERİLERİ

DNA mikrodizisi bir mikroskop lamıdır. Üzerine sabit noktalarda monokataner DNA molekülleri yerleştirilmiştir. Bir robot dizinleme aygıtı kullanılarak yüksek yoğunlukta bir dizi olarak basılmış binlerce ayrı DNA dizisinden oluşurlar. Bu noktalı DNA dizilerinin iki DNA veya RNA örneğindeki göreceli bolluğu, iki örneğin

(veya bir örnek ve bir kontrolün) dizideki diferansiyel hibridizasyonu gözlemlenerek değerlendirilebilir. Bunlar tek flüoresan, çift flüoresan, radyoaktif veya kolorimetrik etiketlerle oluşturulabilir ve her durumda kayıt yöntemleri farklıdır.

Küçük, yoğun substratları bulunan diziler de DNA çipleri olarak anılırlar. Bu sayede gen bilgileri çok kısa bir sürede incelenebilir, çünkü yüzlerce gen analiz edilmek üzere DNA mikrodizisine yerleştirilebilir [4]. Mevcut teknolojilerle 22 nitelikli veya örnekli, 61,000 gen kapasiteli DNA mikrodizileri elde edilebilir.



Şekil 3.1 Bir DNA Mikroçipi ve yakından gösterilmiş bir kesit parçası

Gen ifade verileri, genlerin ne zaman ve nerede devreye girdiklerini, yani kendilerini az veya çok nasıl ifade ettiklerini gösterir. cDNA, tamamlayıcı DNA demektir ve yapay olarak geri dönüştürülmüş durumdadır. mRNA ise orijinal genom DNA'sında bulunan kodlanmayan dizi boşluklarını, ya da intronları içermez. Chee, muhtemelen gen ifadesi verisi tabiri dahi bilinmezken, dünyaya mikrodizi verilerinin nasıl kullanılacağını açıklamıştır [5]. Mikrodizi teknolojisinin kanser tanısı gibi alanlarda kullanımının önemi ilk defa 1998'de tartışılmıştır [6]. Bowtell, mikrodizi deneylerinden gen ifadesi verileri elde etmenin çeşitli yollarını kendi

makalesinde tartışmıştır [7]. Basset, gen ifadesi verilerinin önemini ve gelecekteki uygulamalarını vurgularken [8], White, meyve sineği metamorfozunu açıklamak için mikrodizi verileri uygulamıştır [9]. Golub, kanserin moleküler sınıflandırması üzerine bir makale yayınlamıştır [10]. Bu makale, gen ifadesi gözlemleme yoluyla sınıf saptama ve sınıf tahminleri üzerine olan çalışmaları ile ilgilidir. Slonim, gen ifadesi verilerini kullanarak sınıf tahmini ve saptaması üzerine çalışmalar yürütmüştür [11]. Ramaswami, tümör gen ifadesi imzasını kullanarak çok sınıflı kanser tanısı ile ilgili bir makale yayınlamıştır [12]. Li'nin yayınladığı gibi, daha eski makalelerde ise sadece genlerin aktif olup olmadıkları kaydedilmiştir [13].

Tipik bir mikrodizi deneyi hem çok fazla, hem de çok az bilgi sağlar. Yeni araştırma projelerindeki yaklaşım, az sayıda değişkeni incelemek ve ölçümleri defalarca tekrarlamaktır; ancak mikrodizi deneylerinde incelenen ayrı ayrı genlere karşılık binlerce değişken olabilir. Bu yüzden, çiplerin yüksek maliyeti tekrarlanan gözlemlerin sayıca çok düşük olmasına yol açar.

3.1 Veri Formatı

Dizilerin ve yapısal verilerin aksine, mikrodizi deneylerinden elde edilen verilerin gösterimi için uluslararası kabul edilmiş bir yöntem yoktur. Bunun nedeni deneysel tasarım, deney platformu ve metodolojilerdeki çok çeşitliliktir. Son zamanlarda mikrodizi verilerinin gösterilmesi ve iletilmesi için bir ortak dil geliştirilmesi inisiyatifi önerilmiştir. Deneyler MIAME adı verilen bir standart formatta tarif edilmiş ve standartlaşmış bir veri aktarımı modeli ve XML tabanlı bir mikrodizi biçimleme dili kullanılarak iletilmesi önerilmiştir [14] [15]. Gerçek dünya mikrodizi verileri çok sayıda farklı formatta depolanır, ve gösterim için henüz tekdüze standartlar yoktur. Aşağıda, genel anlamda, mikrodizi gen ifadesi verilerinin temsil edildiği formatlar verilmiştir:

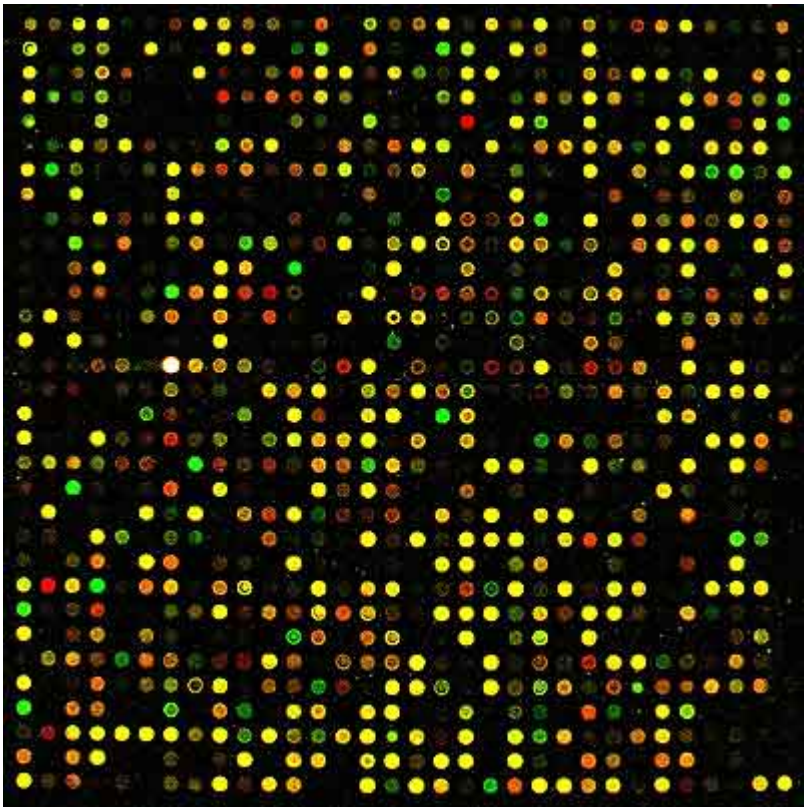
• GenePix

• Affymetrix

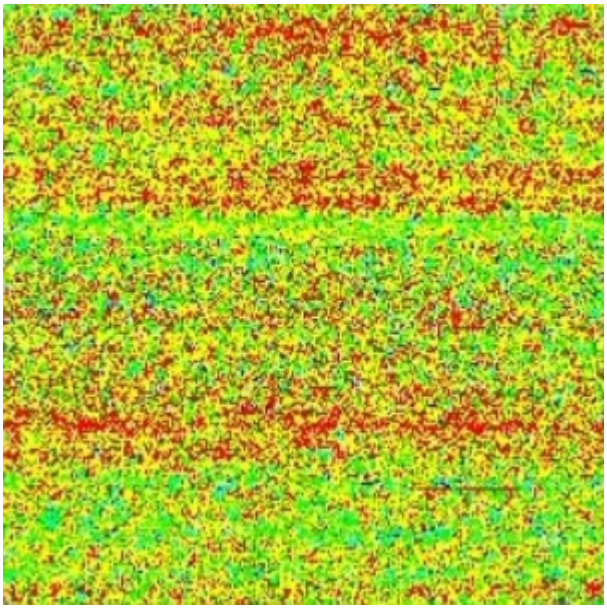
• Agilent

• ScanAnalyze

• ArrayExpres



Şekil 3.2 GenePix veri formatı kullanan bir mikrodizi DNA çipi



Şekil 3.3 Affymetrix veri formatı kullanan bir mikrodizi DNA çipi

3.2 Veri İşleme

Genler normalde olağanüstü ifade düzeyleri gösterir ve onları, kolay çizilmeleri ve görüntülenmeleri için, örnekler bazında ortalama 0 ve standart sapma 1 arasında her gen için ayrı ayrı normalize etmek gerekir. Gen ifadesi örüntü ön işlemleri hakkında Herroro tarafından hazırlanan ayrıntılı bir el kitabı 10 farklı tür ön işlem faaliyeti listelemektedir [16] [17], ancak hepsinin girdi veri setine uygulanması gerekmez. Farklı tür veri setlerine farklı ön işlem algoritmaları uygulanabilir.

Hedenfalk'tan başka [18], diğer kayda değer çalışmalara Khan [19], Hwang [20] ve meme kanserinin tanısı ve/veya tahminiprofil çıkarılması için makine öğrenme tekniklerini kullanan Chen dahildir [21]. Alizadeh, lenfom veri tabanında yaygın B-hücresi analizi için gen ifadesi verilerini kullanmıştır [22]. Çoklu tümör türlerinin sınıflandırılması Yeang tarafından tarif edilmiştir [23].

Bir mikrodizi aşağıdaki özellikleri barındırmalıdır:

• Anlaşılabilirlik

• Sonuç güvenliği ve olasılık yüksek

• Yanlış pozitiflerin varlığında dahi sağlamlık ve devamlılık

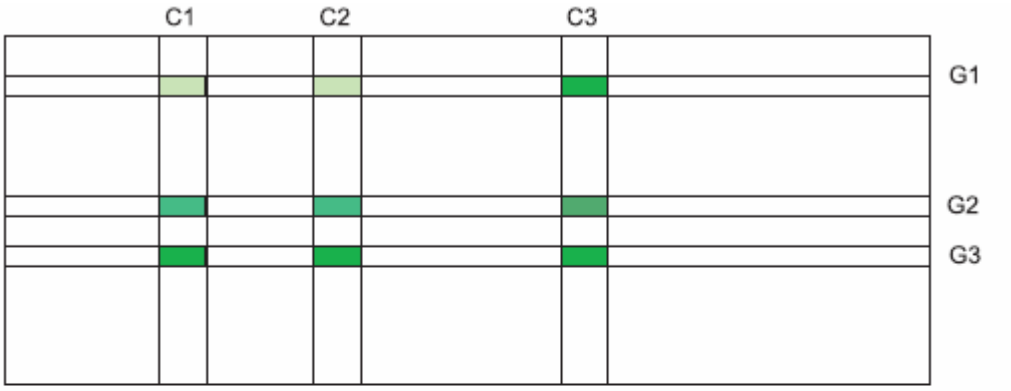
• Hız

Bu işlem sırasında, basit yaklaşımların en sağlam olanlar olduğu hatırlanmalıdır.

3.3 Gen İfadesi Veri Matrisi

Bir matriste sunulan ifade verileri, genleri simgeleyen satırlar (örneğin, çeşitli dokular, gelişim aşamaları ve tedaviler) ayrıca genleri simgeleyen sütunlar ve her hücrede belirli geninin ifade seviyesini simgeleyen numaralar içerir. Bir gen matrisi bir çok örneği (farklı canlılar veya hastalardan) veya aynı varlığın farklı zamanlardaki örneklerini (yani zaman dizileri) karşılaştırmak için kullanılabilir. Örnekler sütunlar halinde düzenlenirler, ve gen ifade değeri satırlardan elde edilebilir. Birincisi genlerin ifade düzeylerinin karşılaştırmalı analizi, ikincisi gen setlerinin değişim (büyüme ve bozulma) örüntülerini saptamak için kullanılır. Bu gibi bir tabloya gen ifade matrisi denmektedir (Şekil 3.4) [24]. Şekil 3.4'te, NC × NG ile ifade edilen veri noktaları, NG (G1, G2, G3) dikey eksende bulunan 3 gen

ve üç deney koşulu NC (C1, C2, C3) yatay eksende işaretlenmiştir. Her bir veri noktasının değişik tonlarda gölgelendirilmesi gen ifadesinin seviyesini temsil etmektedir, mesela daha koyu renkler yüksek ifade seviyelerini göstermektedir.



(a)

Örnekler → / Genler ↓	S1	S2	S3	S4	...	Sn
G1						
G2						
G3						
G4						
. . .						
Gm						

(b)

Şekil 3.4 Üç Deney Sonucunun Birleştirildiği İdeal Anlatım Dizisinin Şeması

4. VERİ MADENCİLİĞİ TEKNİKLERİ

Verilerin hacimsel miktarı, yüksek boyutluluğu ve verilerin heterojen yapısı göz önünde bulundurulduğunda, mevcut veri madenciliği yöntemleri yetersiz kalabilir. Veriler özel olarak veri analizi görevi için toplanmadığından, birkaç deneysel yöntem dışında sıklıkla bir ön işlem gereklidir. Veri ön işleminin gerçekleştirdiği şeyler arasında boyutluluk indirgemesi veya nitelik seçimi, veri analizinde ana sorunlar olarak kabul edilmiştir [25].

Alt boyutlar için iyi performans gösteren veri analizi teknikleri, daha yüksek boyutlardaki veriler için performansta başarısız olmaktadır. Bu artan boyutlar ve tekniklerin nitelik sayısındaki değişiklikleri ele alma konusunda başarısızlığı, boyutluluğun laneti olarak adlandırılmaktadır.

Dünya çapında muazzam miktarlarda veri varken, ya yapay yollarla, ya da doğal süreçlerle, sonuç elde etmek için çok sayıda teknik kullanmıştır. Bu tekniklerin en yaygın olanları kümeleme ve sınıflandırmadır. Veri madenciliği görevleri, tahmin yöntemleri ve betimleme yöntemleri olarak sınıflandırılmıştır. Tahmin yöntemleri bazı değişkenleri başka değişkenlerin bilinmeyen ya da gelecekteki değerlerini tahmin etmek için kullanılır, örneğin sınıflandırma, regresyon ve sapma saptaması. Betimleme yöntemleri veriyi betimleyen ve insanların yorumlayabileceği örüntüleri bulmak için kullanılır, örneğin kümeleme, ilişkilendirme ve sınıflandırma. Yapay zeka, örüntü tanıma, istatistik, veritabanı yönetimi sistemleri ve bilgi görüntüleme gibi diğer bazı alanlardan alınan teknikler birleştirilerek etkili yöntemler haline getirilmiştir. [26].

4.1 Kümeleme ve Sınıflandırma

Bir gen ifadesi matrisindeki her genin bir ifade profili, yani bir dizi örnek boyunca ifade ölçümleri vardır. Mikrodizi verilerinin analizi, bu verilerin benzer ifade profillerine göre gruplandırılmasını içerir. Genleri gruplandırmak için önceden belirlenmiş bir sınıflandırma sistemi kullanılırsa, analiz denetimli olarak adlandırılır. Eğer önceden belirlenmiş bir sınıflandırma yoksa analiz denetimsiz olarak tanımlanır ve kümeleme olarak bilinir.

Kümeleme önce gen ifade matrisinin, benzer ifade profilleri olan genler ile bir arada gruplandırılmasını sağlamak gerekir. Bunu gruplandırma için uzaklık bilgisi kullanılır. Bu genellikle her değer çifti için Öklit uzaklığı olarak bilinir. Kümeleme yöntemleri arasında, hiyerarşik kümeleme, *k* ortalaması kümelemesi ve kendi düzenlenen haritaların derivasyonu gibi bir çok yöntem vardır. Kümeleme, objeler arasındaki benzerliklere göre benzer nesneleri bir araya getirme yöntemidir. Hiyerarşik kümeleme dışındaki Çoğu durumda kümeleme işlemine başlamadan önce gereken küme sayısı belirtilmelidir.

Kümeleme denetimsizdir (yani, başlangıçta bölünme hakkında hiç bir şey bilinmemektedir), buna karşılık sınıflandırma alt türlere bölünmenin baştan bilindiği denetimli bir öğrenme sürecidir. Kümeleme, daha önceden bilinmeyen genlerin tanımlanmasına yardımcı olur. Bu onların fonksiyonları hakkında bir fikir verir, çünkü belli bir kümenin benzer türdeşlik gösteren birimler (veya genler) içerdiği varsayılır. Bu, öğrencilerin otomatik olarak gruplar ya da kümeler oluşturduğu, ve

her kümedeki bir veya iki öğrencinin niteliklerine dayalı olarak diğer öğrencilerin de niteliklerinin tahmin edilebileceği bir sınıfa benzer. Doğru olarak tahmin edilemese dahi, en azından iyi bir fikir vereceğine inanılmaktadır.

4.2 Nitelik İndirimi

Mikrodizi veri setleri çok büyük olduklarından sınıflandırma ve kümeleme son derece yorucu ve bilgisayar kaynakları açısından zorlayıcı olabilir. Bu yüzden bazı verilerin çıkarılması gereklidir. Örneğin, eğer iki veri aynı gen ifadesi üzerinde aynı etkiye sahipse, bu veriler gereksizdir, ve matrisin bir sütunu tümüyle çıkarılabilir. Eğer belli bir genin ifadesi bir dizi boyunca aynı ise, bu geni ilerideki analizlerde kullanmak ne gerekli ne de faydalıdır, çünkü diferansiyel gen ifadelerinde hiçbir faydalı bilgi sağlamamaktadır. Bu durumda bir satır tümüyle çıkarılabilir. TBA/SVD gibi yaklaşımlar böyle gereksiz veya bilgi vermeyen veri setlerini seçmek için kullanılabilir. Gereksiz veriler birleştirilerek tek veya bileşik bir veri seti oluşturulur ve bu yolla gen ifadesi matrisinin boyutları küçültülerek analiz basitleştirilir.

4.3 Nitelik Ayıklama

Nitelik ayıklama, hasarlı ve hasarsız verileri ayırt etme olanağı tanıyan ve ölçülen dinamik tepkiden türetilmiş hasara karşı hassas niteliklerin tanımlanması sürecidir. Dolayısıyla, dinamik tepki zaman dizisi verilerinden bu niteliklerin nasıl aranacağı, bireysel yapısal hasarları tanımlamada en önemli aşama haline gelir. Dahası, sensörlerden ölçülen veriler, gürültü, ısı, vb. gibi yapısal ortam faktörlerine karşı hassas olduğundan tanımlanmaları ve saptanmaları güçtür. Bazı araştırmacılar sensörlerden ölçümlenen sinyallerin işlenmesine daha fazla dikkat etmişlerdir.

5. GEN İFADESİ MODEL SEÇİCİ (GEMS)

Gen İfadesi Model Seçici (GEMS), dizi gen ifadesi verilerinden denetimli bir biçimde tanısal ve sonuç tahminleri oluşturan bir sistemdir. Bu modellerin örnekleri: (a) kanser saptayan modeller, (b) kanserin doğru alt türünü saptayan modeller, veya (c) tedaviden sonra hayatta kalma oranını tahmin eden modellerdir. Bu tür karmaşık karar verme özelliğini destekleyen modellerin gelecek yıllarda tıpta devrim yaratma potansiyeline sahip oldukları yaygın olarak kabul edilmektedir. Karar destekleyen modellerin yanı sıra, GEMS tanı ve/veya sonuç

tahminleri için, veya ondan daha iyi olan az sayıda geni seçmek için kullanılabilir. Bu genler keşif amacı için de faydalıdırlar (yani çeşitli türde kanserlerin nedenlerini veya tedavilerini önerebilirler). Son olarak, GEMS modellerin ilerideki uygulamalarındaki performansları (yani doğrulukları) hakkında bir tahmin sağlarlar (örneğin, modeli oluşturmak için değil, ancak modellerin oluşturmasında kullanan aynı hasta popülasyonundan gelen hastalara uygulandığında), ve kullanıcıların modelleri ayrı ayrı hastalarda uygulamaya izin verirler. Bu gibi modelleri oluşturmak, (a) istatistik ve/veya biyoinformatik ve/veya örüntü tanıma konularında uzmanlık eğitimi gerektirir, (b) tipik akademik ortamda birkaç haftadan birkaç aya kadar sürer, ve (c) eğitim seti için çok iyi olan ancak ileride ayrı ayrı hastalara uygulandığında kötü performans gösteren modeller oluşturabilir. GEMS bu görevleri hızlı, otomatik olarak, aşırı uyarlama yapmadan ve kullanıcının veri analizi konusunda uzman olmasını gerektirmeden yerine getirir [27].

5.1 Avantajları:

1. GEMS'in öğrenme algoritmaları 74 kanser türünü kapsayan 11 halka açık veri seti kullanılarak gerçekleştirilen yaygın algoritmik değerlendirmelerden sonra yaklaşık 20 algoritma arasından seçilmiştir.

2. Kapsamlı doğrulama: (a) GEMS yukarıdaki veri setlerinin tekrar analizi ile test edildi. (b) GEMS 5 "yeni" veri seti ile gerçek uzmanlara karşı çapraz doğrulama yöntemi kullanılarak doğrulandı. (c) GEMS iki çift veri seti ile ve bağımsız (çapraz veri seti) doğrulama yöntemi ile doğrulandı. Doğrulama hem model hem de gen markörü genelleştirilebilme içerdi. Toplam olarak GEMS 16 veri setiyle doğrulandı.

3. Tamamen otomatiktir.

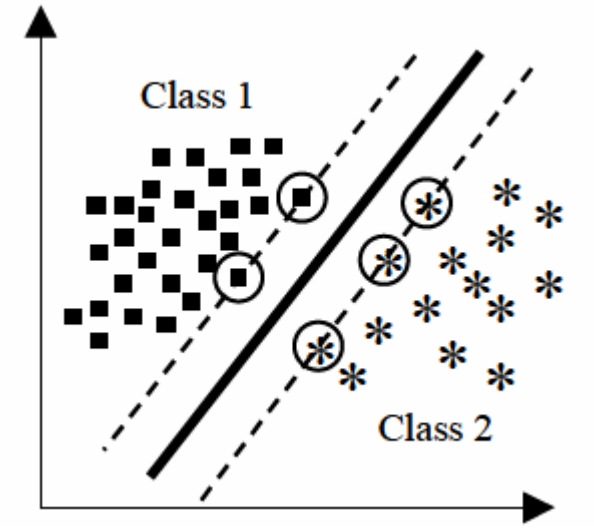
4. İyi tanımlanmış özellikleri, doğruluk için teorik garantileri ve mükemmel performansı olan özel gen seçimi ve nedensel keşif algoritmaları içerir.

5. İstemci – sunucu mimarisi.

6. İKİLİ DESTEK VEKTÖR MAKİNELERİ

İkili DVM'lerin ana fikri çekirdek fonksiyonu ile veriyi daha yüksek boyutsal uzaya eşlemek ve daha sonra çalışma örneklerini ayıran maksimum marjin hiper düzleminin tespiti için optimizasyon problemini çözmektir. Hiper düzlem destek vektörleri olarak adlandırılan sınırlı çalışma durum setini baz alır. Yeni durumlar, uygun düştükleri hiper düzlem tarafına uygun olarak sınıflandırılır (Şekil 6.1). Optimizasyon problemi çoğunlukla, yanlış verilere izin verecek şekilde formüle edilir.

Şekil 6.1'de, sınırlı çalışma örnekleriyle belirlenen hiper düzleme, destek vektörleri denir ve çemberle gösterilmektedir.



Şekil 6.1 Bir İkili DVM'nin Hiper Düzlem Şeması

7. ÇOĞUL SINIFLI DESTEK VEKTÖR MAKİNELERİ

Destek vektör makinesi temel olarak iki sınıflı veri setleri için kullanılır. Bu nedenle, sık sık K > 2 sınıflarıyla ilgili sorunlarla uğraşmak zorunda kalırız. Buna çözüm olarak, çoklu iki sınıf Destek Vektör Makinelerini değişik kombinasyonlarını kullanarak, çok sınıflı bir sınıflandırıcı oluşturması önerilmektedir.

K-ayrı Destek Vektör Makineleri oluşturmak için en yaygın kullanılan ve içinde her seferinde 1 adet pozitif örnek alarak ve geriye kalan K − 1 negativ örnek alarak denendiği yöntem en yaygın kullanılan yaklaşımdır [28].

Bu ayrıca kalana karşı tek yaklaşımı olarak da bilinir. Bu problem bazen yeni girdiler için aşağıdaki formülün kullanımıyla da görülür

$$y(x)=\max y_k(x) \quad (1)$$

Ne yazık ki, bu yaklaşımın; farklı sınıflandırıcıların farklı görevlerde denenmesi ve farklı sınıflandırıcıların yeniden değerlendirilen miktarlarının uygun skalaları (yk(x)) sağlamasının garantisi olmaması gibi problemleri mevcuttur.

Kalana karşı tek yaklaşımının başka bir sorunu da, çalışma setlerinin dengesiz ve biçimsiz olmasıdır. Örneğin, her birinin eşit çalışma verisi noktasının olduğu on sınıfımız var ise, bu durumda veri setlerinde çalışılan tekil sınıflandırıcılar %90 negatif ve %10 oranında pozitif örneklerden oluşacaktırlar, ve orijinal problemin simetrisi kaybolacaktır. Kalana karşı tek yaklaşımında, hedef değerlerin +1 pozitif ve negatif sınıflarınında −1/(K − 1) olması düşünülmüştür. Weston ve Watkins tüm K-DVM'lerin aynı anda çalışabilmesi için, her bir sınıftaki marjinleri maksimuma çıkaracak tek hedef işlevi belirlemişlerdir [29]. Fakat, bu daha yavaş bir çalışmaya neden olabilir çünkü; K-ayrı optimizasyon problemleri, N veri değerleri için toplam maliyeti O(KN2)'dir. Bu yüzden bu sorununu boyutu (K−1)N ve de toplam O(K2N2) maliyeti verilerek çözülmelidir. Başka bir yaklaşım da ise K(K−1)/2 boyut kullanarak, farklı 2-sınıflı DVM'nin tüm sınıf çiftleri üzerine uygulanması ve daha sonra sınıfın en çok oy sayısı aldığı test sonuçlarına göre sınıflandırma yapılması yaklaşımıdır. Bu yaklaşım bazen bire karşı bir olarak adlandırılır. Ayrıca, büyük K için bu yaklaşım, kalana karşı bir yaklaşımına kıyasla daha fazla zamana gerek duymaktadır. Benzer şekilde, test puanlarının değerlendirilmesi için fazla sayım yapılması gerekmektedir. İkinci problem, DAGSVM ile sonuçlanan bir grafik içerisinde eşey sınıflandırıcıların uygulanmaları ile giderilebilir. K sınıfları için, DAGSVM; K(K − 1)/2 boyutlu sınıflandırıcının toplamına eşittir, ve yeni bir test puanı sınıflandırmasının değerlendirilmesi için K − 1 eşey sınıflandırıcıları gerekmektedir.

Dietterich ve Bakiri tarafından çok sınıflı sınıflandırmaya, hata düzeltme çıkış kodlarını baz alarak, farklı bir yaklaşım geliştirmiştir ve vektör makinelerinin

desteklenmesi için Allwein tarafından uygulanmıştır [30] [31]. Bu tekil sınıflandırıcıları çalıştırmak için daha genel sınıf parçacıkları kullanıldığı için bire karşı bir yaklaşımının genellendirilmesi olarak da görülebilir. K sınıfları kendi başlarına seçilen ve iki sınıf sınıflandırıcılardan alınan yanıt setleri olarak temsil edilir ve uygun bir kod çözme şemasıyla, bu tekil sınıflandırıcı çıktılarındaki hatalara ve belirsizliklere karşı sağlamlık sağlar. Her ne kadar da DVM çoklu sınıf sınıflandırma problemleri için yetersiz kalmakta ise de, kalana karşı bir yaklaşımı uygulamadaki kısıtlamalarına rağmen yaygın olarak kullanılmaktadır. Ayrıca olasılık yoğunluğu tahminine ilişkin denetlenme yapmayan öğrenme problemlerini çözen tek-sınıf destek vektör makineleri de bulunmaktadır. Modelleme yerine veri yoğunluğu kullanılarak, bu metotlar yüksek yoğunluklu bölgeleri de kapsayan sağlam sınırlar bulmayı amaçlamaktadırlar. Sınır, yoğunluk dağılımını temsil etmesi için seçilmiştir ki bu da, muhtemelen 0 ile 1 rakamlarıyla gösterilecektir. Tüm yoğunluğu tahmin etmekten daha bazı problemler vardır ancak bazı spesifik uygulamalar bunları engellemeye yeterli olabilir. Bu problemlere destek vektör makineleri kullanarak çözüm sağlayan iki yaklaşım önerilmektedir. Schölkopf algoritması tüm ve özellikle de kökenden elde edilen çalışma verisinin, v fraksiyonunu ayıran ve aynı anda hiper düzlemin köken ile aralığını maksimuma çıkaran hiper düzlem bulmaya çalışmaktadır, diğer tarafta Tax ve Duin nitelik uzayda veri noktalarını içeren en küçük küreyi aramaktadırlar [32] [33] .

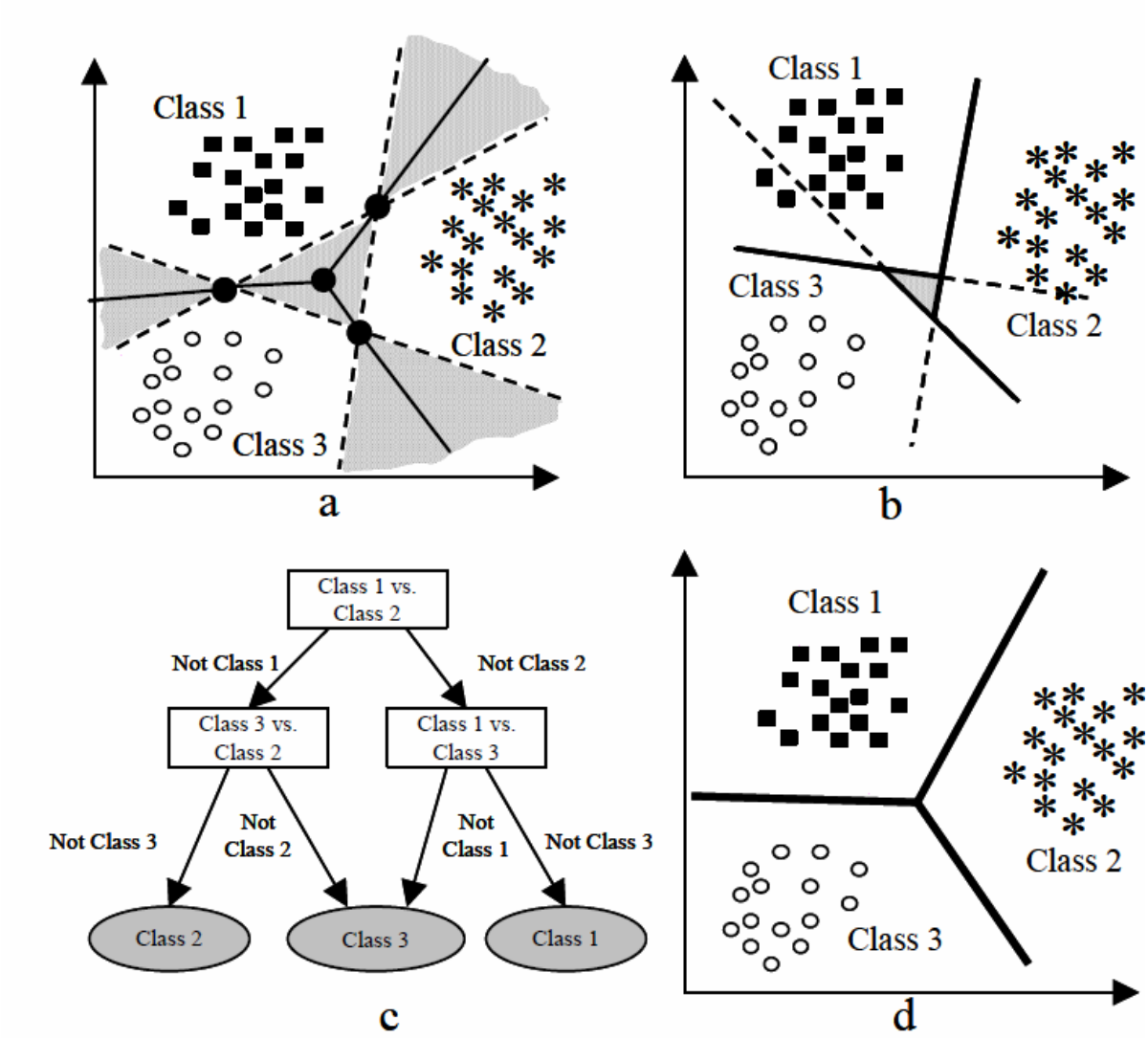
7.1 Kalana Karşı Bir (KKB)

Bu kavramsal olarak en kolay çoklu sınıf DVM metodudur. Burada k ikili DVM sınıflandırıcılarını oluşturur: sınıf 1 (pozitif) tüm diğer sınıflara karaşı (negatif), sınıf 2 tüm diğer sınıflara karşı, … , sınıf k diğer tüm sınıflara karşı (Şekil 7.1a). Birleştirilmiş OVR fonksiyonu daha sonraki pozitif hiper düzlem tarafından belirlenen ikili

k kararı fonksiyonlarına uygun düşen örneklem sınıfı seçer. Böyle yaparak karar düzlemleri, k DVM" tarafından hesaplanır. Ve bu çoklu kategori sınıflandırmanın optimizasyonunu sorgular. Bu yaklaşım, hesaplanması açısından zordur, çünkü bizim için k kuadrik programlama (QP) optimizasyon boyutu n' dir. Ayrıca, bu teknik sağlam öğrenme algoritmasıyla alakalı olan genelleme analizi gibi teorik, doğrulamaya sahip değildir.

7.2 Bire Karşı Bir (BKB)

Bu metot, tüm sınıf çiftlerinin ikili DVM sınıflandırıcılarının oluşturulmasıyla alakalıdır; toplamda üç $\left(\frac{k}{2}\right) = \frac{k(k-1)}{2}$ çift vardır. (Şekil 7.1b). Başka bir deyişle, her bir sınıf çifti için, ikili DVM problemi çözülür. Karar fonksiyonu bir sınıfa bir örnek ataması yapar, ardından atadığı sınıf en yüksek oy sayısına sahiptir ve bu Max Wins stratejisi olarak adlandırılır [34]. Şayet halen bir bağ mevcut ise, daha sonraki hiper düzlem tarafından belirlenen sınıflandırmaya dayalı etikete bir örnek atanır. Bu yaklaşımın faydalarından birisi de her bir sınıf çifti için daha küçük optimizasyon problemiyle uğraşmamız, ve toplamda n'den daha küçük boyutlu k(k-1)/2 QP problemleri çözmemizdir. DVM'ler için kullanılan QP optimizasyon algoritmalarının problem boyutuna göre polinom tipinde olduğunu varsayarsak, zamanda önemli tasarruf sağlar. Ayrıca, bazı araştırmacılar bazı ikili alt problemler ayırabilir iken tüm çoklu kategori problemi ayrıramaz yine de, OVO'nun OVR'ye kıyasla sınıflandırmanın iyileştirmesini sağlayacağını ortaya koymuşlardır [35]. OVR yaklaşımından farklı olarak burada eşitliğin bozulmasında sadece küçük bir rol oynar ve kararın bütününe büyük etkide bulunmaz. Diğer taraftan, OVR'e benzer şekilde, OVO halihazırda genelleştirme hatalarında belirlenen sınırlara sahip değildir.



Şekil 7.1 Üçlü sınıflı teşhis problemine uygulanan MC-SVM algoritmaları. (a) MC-SVM Kalana-Karşı-Bir, 3 ayrı sınıflandırıcıdan oluşur: (1) sınıf 2 ve 3'e karşı sınıf 1, (2) 1 ve 3'e karşı sınıf 2, ve (3) 1 ve 2'ye karşı sınıf 3. (b) MCSVM Bire-Karşı-Bir aynı şekilde 3 ayrı sınıflandırıcıdan oluşur: (1) sınıf 2'ye karşı sınıf 1, (2) sınıf 3'e karşı sınıf 2, ve (3) sınıf 3'e karşı sınıf 1. (c) MCSVM DAGSVM, Bire-Karşı-Bir DVM sınıfladırıcıları bazında bir karar ağacı oluşturur. (d) Weston ve Watkins'in ve Crammer ve Singer'in MC-DVM metotları tüm sınıflar arasında eş zamanlı olarak tek bir sınıflandırıcı oluştururlar.

7.3 DAGSVM

Bu algoritmanın çalışma aşaması OVO yaklaşımına benzer; ancak DAGSVM'nin test aşaması $\left(\frac{k}{2}\right)$ sınıflandırıcılarını (bkz Şekil 7.1c) kullanan ve DDAG' a yönlendirilmiş köklü ikili karar oluşturulması gerekmektedir [36]. Bu grafiğin herbir düğümü bir sınıf çifti için köklü DVM dir(p, q). Topolojik olarak en alt seviyede bulunan k sınıflandırma kararlarına uygun k yaprakları bulunur. Yaprak olmayan her bir düğüm (p, q) iki uca sahiptir – sol uç "p olmayana" uygundur ve sağdaki ise " q olmayana" uygundur. DDAG listesindeki sınıf sırasının seçimi ampirik olarak gösterildiği gibi de raslantısal olarak seçilebilinir. OVO metodundan gelen avantajlara ek olarak, DAGSVM genelleştirme hatasını düzelmedeki başarısı da büyük bir avantaj sağlar.

7.4 Weston ve Watkins (WW) Metodları

Çok sınıflı DVM'lere olan bu yaklaşım bazı araştırmacılar tarafından ikili DVM sınıflandırma probleminin doğal uzantısı olarak görülür (bkz Şekil 7.1d) [37] [38].

7.5 Crammer ve Singer (CS) Metodu

Bu teknik WW'ye benzer (bkz Şekil 7.1d) [39]. Bu, (k-1)n boyutunda bir tekli QP probleminin çözümünü gerektirir, ancak, optimizasyon probleminin sınırlandırmasında daha az yapay değişken kullanılır ve bu nedenle sayısal olarak ucuzdur. WW'ye benzer olarak, ayrıştırmaların kullanımı, optimizasyon probleminin çözümünü önemli miktarda hızlandırır [37]. Ne yazık ki, CS'nin optimizasyonu henüz yapılmamıştır.

8. GEREÇ VE YÖNTEM

8.1 Problem Tanımı

Mikrodizi gen ifade verilerinden elde edilmiş veri kümelerinin çok yüksek sayıda veri niteliği barındırması ve bunun performansa olan etkileri.

8.2 Amaç

Veri kümesinin barındırdığı veri niteliklerini yerleşik algoritmalar ve bunların farklı yorumlanması ile indirgenmesi.

8.3 Veri Kümesi Seçimi

Alexander Statnikov makalesinde 11_Tumors, 14_Tumors, 9_Tumors, Brain_Tumor1, Brain_Tumor2, Leukemia1, Leukemia2, Lung_Cancer, SRBCT, Prostate_Tumor, DLBCL olmak üzere 11 adet veri kümesi üzerinde çalışmıştır [1]. 11_Tumors, bir insanda bulunabilecek 11 çeşit tümörün gen ifade verilerinin belli bir format ile bir dosyada tutulmuş halidir. 14_Tumors, bir insanda bulunabilecek 14 çeşit tümörle, yine bir insanda bulunabilecek 12 normal doku hücresinin heterojen bilgilerinin tutulduğu formatlı bir dosyadır. 9_Tumors, bir insanda bulunabilecek 9 çeşit tümörün tutulduğu formatlı bir dosyadır. Brain_Tumor1 dosyası, 5 tip insan beyin tümörünün bilgilerini tutmaktadır. Lung_Cancer veri kümesi, bir insanda bulunabilen 4 çeşit akciğer tümörü ve normal doku hücreleri bulundurmaktadır. Prostate_Tumor, içerisinde prostat tümörü ve normal doku hücrelerinin bilgilerini bulunduran veri kümesini tutan dosyadır. Brain_Tumor2 dosyası, adet alt tür beyin tümörünü tutan bir dosyadır. Leulemia1 ve Leukemia2 adlı dosyalar, birçok karışık kan kanseri tümörünü barındırmaktadır.

Bu çalışmada sadece bir veri kümesi seçilmiştir ve bu da 11_Tumors dosyasının içinde bulunan veri kümesidir. Bir tane veri kümesi seçilmesinin nedeni, indirgeme algoritmalarının yeterince ağır ve zaman alıcı yöntemler olmasıdır. Aynı zamanda, veri kümelerıı birbirinden tamamen bağımsız ve farklı değil, bazı noktalarda bağıntılıdır. Bu yüzden bir veri kümesi üzerine yapılan bir çalışma, çok fazla yöntem değişikliğine uğramadan rahatlıkla diğer veri kümelerine uygulanabilir.

SRBCT, mavi tümör hücrelerini tutan bir dosya iken, DLBCK ise lenf kanseri ile ilişkili tümörleri tutan bir başka dosyadır.
11_Tumors veri kümesinin seçilmesinin nedeni, bu kümenin en homojen küme olduğu ve diğerlerine göre daha fazla veri barındırdığından kaynaklanmaktadır. Böylece, uygulanan yöntemlerin karşılaştırılması ve doğru sonuç vermesi daha olası olmaktadır. Alexander Statnikov, resmi bir yazışma sırasında bunu kendisi de belirtmiş ve onaylamıştır.

8.4 Analizler

8.4.1 GEMS ve orijinal veri analizi

Alexander Statnikov makalesinde tüm 11 veri kümesini deney ve kontrol setleri olarak kullanmış ve içinde makalede esas ağırlığı bulunan destek vektör makineleri de bulunan bir dizi makine öğrenme ve yapay zeka tekniği kullanarak sınıflandırma yapmış ve bu sınıflandırmaları, süre, maliyet ve performans açısından karşılaştırmıştır. Burada süre, sistemin çalıştığı sistemin işlemin tamamını bitirme süresi, maliyet, kullanılan işlemci ve hafıza gücü ve performans ise, daha sonradan karşılaştırılan test ve kontrol gruplarının doğruluk oranıdır.
Bu çalışmanın esas ağırlığı, destek vektör makinesine sokulan verilerin küçültülmesi, indirgenmesi ya da en azından içsel bir bağıntı bulunması olduğundan, bu çalışma sadece destek vektör makinesine sokulan bir çeşit veri kümesi ile ilgilenmektedir.
Makalede kullanılan destek vektör makinesi bilgisayar yazılımı GEMS adı verilen bir sistemdir. Yine Alexander Statnikov tarafından geliştirilmiş olan bu sistemin oldukça geniş bir arayüz desteği vardır ve bu sayede destek vektör makine alt algoritmalarından daha önce anlatılan 5 ana algoritma dahil birçok yan algoritmaya destek vermektedir. Bunlardan sadece makalede değinilmiş olan 5 ana alt algoritma bu çalışmada kullanılmıştır. Programdaki diğer ayarlamaların hepsi, programın ilk açılışında, yazılımın kendisi tarafından belirlenen ve yine resmi bir yazışma sonucu Alexander Statnikov tarafından belirtildiği üzere, makaledeki analizler yapılırken kullanılan ayarların aynısı olan ayarlar bu çalışmada kullanılmıştır.

Bu çalışma, makaleyi kendine kaynak olarak almasına rağmen, Statnikov'un bulduğu değerleri doğrudan bir referans olarak almak yanlış olacaktır; çünkü bu makaledeki deneylerin yapıldığı bilgisayar sistemi ile bu çalışmada yapılacak olan sistemsel uygulamaların bulunduğu bilgisayar zemini aynı değildir. Bu yüzden, sağlıklı bir karşılaştırma ve analiz için, makaledeki ilgili tümör veri kümesinin destek vektör makinesine bu çalışmanın referans aldığı bilgisayar sisteminde yeniden analiz edilmesi gerekmektedir. Bunun ışığında, dört çekirdekli bir işlemci ve 4 GB 1200 Mhz üzerinden hafızayla çalışan bir bilgisayarda, aşağıdaki süre, maliyet ve performans sonuçları, orijinal veri kümesi için alınmıştır.

8.4.2 Nitelik ortalaması ve standart sapma

Bu yöntem, elimizdeki n * m boyutlu bir matrisi indirgeme algoritmalarına doğrudan sokmadan önce, veriler üzerinden yüzeysel bir indirgeme yapabilmek amacıyla denenmiştir.

8.4.3 Temel bileşen analizi (TBA)

Ana komponent analizinin (TBA) ana fikri, çok sayıda ilişkili değişkenlerden oluşan bir veri setinde mevcut olan varyasyonların mümkün olduğu kadar çoğunu korurken boyutluluğunu küçültmektir. Bu da, ilişkili olmayan, ve ilk birkaçının orijinal değişkenlerin tümünde mevcut olan varyasyonları koruduğu yeni bir dizi değişkene dönüştürülerek sağlanır.

TBA'nın temelini teşkil edecek biçimde, Fisher ve Mackenzie SVD'yi bir tarım denemesinin iki yönlü analizinde kullandılar [40]. Ancak, şimdi TBA olarak bilinen tekniğin en eski tanımlamalarının Pearson ve Hotelling tarafından verildiği yaygın olarak kabul görmektedir [41] [42]. Hotelling'in makalesi iki bölümden oluşuyor. İlk ve en önemli bölümü, Pearson'un makalesi ile birlikte, Bryant ve Atchley tarafından derlenen belgeler koleksiyonunda yer almaktadır [43].

İki belge de farklı yaklaşımlar benimsenmişti. Öte yandan Pearson , *p* boyutlu uzayda bir dizi noktaya en iyi uyan çizgileri ve düzlemleri bulmakla daha çok ilgilenmişti, ve değerlendirdiği geometrik optimizasyon sorunları da sonuçlandırıyordu [41].

Pearson'un yüz yıl önce bilgisayarlar daha yokken, hesaplama ile ilgili olarak yaptığı yorumlar ilgi çekicidir. Yöntemlerinin kolayca sayısal problemlere uygulanabileceğini ifade etmiş, ve dört veya daha fazla değişken için hesaplamaların külfetli olduğunu belirtmekle birlikte gene de geçerli olduğunu söylemiştir.

Pearson ve Hotelling'in makaleleri arasında geçen 32 yıl boyunca bu konu ile ilgili çok az makale yayınlanmış gözükse de, Rao, Frisch'in Pearson'unkine benzer bir yaklaşım benimsediği görülmektedir [44] [45]. Ayrıca, Thurstone'un da Hotelling'e benzer bir çizgide çalışmakta olduğu bilinmektedir [46], ancak Bryant ve Atchley'e dahil edilmiş olan söz konusu makale, TBA'dan daha çok faktör analizi ile ilgilidir. Hotelling'in yaklaşımı da faktör analizinden yola çıkmaktadır, ancak Hotelling'in tanımladığı TBA karakter olarak gerçekten de faktör analizinden farklıdır [43].

Hotelling'in motivasyonu, orijinal *p* değişkenlerinin 'değerlerini tanımlayan' daha küçük 'temel bağımsız değişkenler dizisi' olabileceğidir. Psikolojik literatürde bu gibi değişkenlere 'faktör' dendiğini belirtir, ancak matematikte 'faktör' sözcüğünün farklı kullanımları ile karışıklık yaratmamak için alternatif olarak 'komponent' sözcüğü kullanılır. Hotelling 'komponentlerini' orijinal değişkenlerin varyasyonlarının tamamına olan katkılarını maksimize edecek şekilde seçer, ve bu yolla türetilen komponentleri 'ana komponentler' olarak adlandırır. Bu gibi komponentleri bulan analize de 'ana komponentler yöntemi' adı verilir [42].

Hotelling'in PC derivasyonu yukarıda verilen, Lagrange çarpanları kullanarak bir özdeğer / özvektör problemi ile sonuçlanan yönteme benzemektedir, ancak üç açıdan farklılık gösterir. Birincisi, Hotelling bir ortak varyasyon değil, ortak ilişki matrisi ile çalışır; ikincisi, komponentlerin orijinal değişkenler olarak ifadesi değilde komponentlerin doğrusal fonksiyonları olarak bakar; ve üçüncüsü, matris notasyonu kullanmaz.

Derivasyonu verdikten sonra Hotelling güç yöntemini kullanarak komponentlerin nasıl bulunacağını gösterir. Aynı zamanda, çok değişkenli normal dağılımlarda sabit olasılıklı elipsoidler açısından Pearson'un verdiğinden daha farklı bir geometrik yorumu ileri sürer. Ancak makalesinin büyük bir kısmı, özellikle ikinci bölümü, normal şeklindeki TBA ile değil, faktör analizi ile ilgilidir [41].

Hotelling'in daha sonra yayınlanmış bir yazısında , PC'leri saptamak için güç yönteminin hızlandırılmış bir versiyonunu sunmuştur; aynı yıl içinde Girshick PC'lerin bazı alternatif derivasyonlarını sundu, ve örnek PC'lerin temeldeki popülasyon PC'lerinin azami olasılık tahminleri olduğu fikrini ortaya attı [47] [48].

Girshick PC'lerin katsayılarının ve varyanslarının asimptotik örnekleme dağılımını inceledi, ancak Hotelling'in makalesinin yayınlanmasından hemen sonraki 25 yıl boyunca TBA'nın değişik uygulamalarının geliştirilmesi üzerine çok az çalışma yapıldığı görülmektedir [49]. Ancak o zamandan beri bir yeni uygulamalar ve daha ileri teorik gelişmeler yaşanmıştır. Bu gelişmeler genel olarak istatistiksel literatürdeki artışı yansıtmaktadır, ancak TBA oldukça yüksek hesaplama gücü gerektirdiğinden, kullanımındaki artış elektronik hesap makinelerinin yaygın tanıtımı ile eşzamanlı olmuştur. Pearson'un iyimser yorumlarına rağmen, *p* dört veya daha az olmadıkça TBA'yı elle yapmak pek geçerli değildir. Fakat TBA en çok nitelikle *p*'nin daha yüksek değerleri için yararlı olduğundan, tekniğin tam potansiyelinden bilgisayarların yayılmasından önce faydalanılamazdı.

Bu bölümü bitirmeden önce dört makaleden söz edeceğiz. Bunlar TBA'ya duyulan ilginin genişlemeye başladığı dönemde yayınlandı, ve konuyla ilgili önemli referanslar haline geldiler. Bunlardan ilki, Anderson'a ait olan, dört yazı arasında en teorik olanı idi [50]. Girshick'in daha önceki çalışması üzerine kurulu olarak örnek PC'lerin katsayı ve varyanslarının asimptotik örnekleme dağılımlarını tartışıyordu, ve daha sonraki teorik gelişmelerde sıkça sözü edildi [49].

Rao'nun makalesi, TBA kullanımı, yorumu ve uzantıları ile ilgili ortaya attığı ve yeni çıkıcak fikirler açısından kayda değerdir [44].

Gower , TBA ve çeşitli diğer istatistiksel teknikler arasındaki ilişkileri ortaya attı, ve bir dizi önemli geometrik kavramı açıkladı [51].

Son olarak Jeffers , TBA kullanımının basit bir boyut azaltıcı araçtan öte olduğu iki durum çalışmasını tartışarak konunun gerçekten pratik olan yönüne ivme kazandırdı [52].

Bu önemli makaleler listesine Preisendorfer ve Mobley'in kitabı da eklenmelidir [53]. Her ne kadar zor okunan bir kitap olsa da, TBA ile ilgili, henüz tam olarak araştırılmamış yeni fikirler açısından Rao'ya rakiptir [44]. Kitabın büyük bölümü

yıllar boyunca Preisendorfer tarafından yazılmış, ancak zamansız ölümünün ardından Mobley tarafından düzenlenmiş ve bastırılmıştır [53].

Tekniğin görünürdeki kolaylığına rağmen, TBA konusunda çok sayıda araştırma sürdürülmekte, ve yaygın olarak kullanılmaktadır. Bu, Web of Science'ın (Bilim Ağı) 1999 – 2000 yıllarında, iki yıllık bir süre içinde, adında, özetinde veya anahtar sözcükleri arasında 'ana komponent analizi' veya 'ana komponentler analizi' sözcüklerinin geçtiği yayınlanmış 2,000'den fazla makale tanımlamasından açıkça anlaşılmaktadır. Bu kitaptaki referanslar da TBA'nın uygulandığı alanların geniş çeşitliliğini göstermektedir. Tarım, biyoloji, kimya, iklimbilim, demografi, ekoloji, ekonomi, gıda araştırmaları, genetik, jeoloji, meteoroloji, oşinografi, psikoloji ve kalite kontrolüne uygulamaları içeren kitaplardan söz edilmektedir, ve bu listeye daha da ilaveler yapmak mümkündür [54].

8.4.4 Doğrusal olmayan ana komponent analizi (NLPCA)

Literatürde doğrusal olmayan ana komponent analizi için tamamen farklı iki şekil önerilmiştir. Bunlardan ilki Guttman, Burt, Hayashi, Benzécri, McDonald, De Leeuw, Hill ve Nishisato tarafından bulunmuştur. Buna çoklu benzeşme analizi diyoruz. İkinci şekil Kruskal, Shepard, Roskam, Takane, Young, De Leeuw, Winsberg ve Ramsay tarafından tartışılmıştır. Buna da doğrusal olmayan ana komponent analizi diyoruz. İki şekil Albert Gifi tarafından hem geometrik hem de sayısal olarak ilişkilendirilmiş ve birleştirilmiştir.
Çoklu benzerlik analizi için Cazes ve Lebart veya Hill ve Nishisato'ya ayrı ayrı değinebiliriz [55] [56] [57] [58]. Yalnızca nispeten basit bir durumu tartışıyoruz, özellikle Toulouse'da geliştirilen birçok genellemeler esas olarak bizim çerçevemize de oturtulabilinir. Doğrusal olmayan esas bileşen analizi için Kruskal ve Shepard veya Young'a değinebiliriz [59] [60]. Yine bu teknik ile ilgili basit özel bir durum üzerinde çalışıyoruz. İki yaklaşımı birleştirmek için önceki girişimler için De Leeuw ve Van Rijckevorsel, De Leeuw ve Gifi 'ye değinebiliriz [61] [62] [63].

İlk olarak L1 · · · ,Lm 'nin iç çarpım norm k.k ve birim küre S ile belirtilir. Hilbert alanı H'nin kapalı alt alanları olduğunu düşünelim. Daha sonra Lj ve S kesişimi olan her bir yj 2 LjS eleman seçimi için R(yi, · · · , ym) matrisini r j`(y1, · · · , ym) =< yj, y` > elemanları ile hesaplayabiliriz. Pozitif yarı açıklayıcı olması ve birleşmeye eşit diyagonal elemanlara sahip olması açısından bu matris bir korelasyon

matrisidir. Doğrusal olmayan esas bileşen analizinin (NLPCA) problemi R(yi, · · · , ym) matrisinin p en büyük öz değerlerinin toplamının maksimize edildiği (veya denk olarak m-p en küçük öz değerlerinin toplamının minimize edildiği) bir şekilde $y_j \in L_j S$ 'i bulmaktır. P'nin farklı seçimleri için bunun farklı bir problem olduğunu varsayalım. Bazı durumlarda sadece kriterlerimizi maksimize eden çözümlerle ilgilenmemeliyiz, aksine maksimizasyon problemine karşılık gelen sabit denklemlerin tüm çözümleri ile ilgilenmeliyiz.

NLPCA'nın alt alan Lj'lerin tek boyutlu olan bileşen analizini genelleştirdiği açıktır. Formüllerimiz NLPCA'nın bir açıdan kısıtlı formuyla ilgilenmektedir, çünkü belirttiğimiz literatürün çoğu Lj'nin kapalı konveks koni olduğu düşünerek daha genel bir durumu ele almaktadır.

Tenenhaus da muhtemel olarak sonsuz sayıda konveks koni durumunu tartışır. Çalışmalarımızın çoğunda bir diğer ciddi genelleme kısıtlaması bulunmaktadır [64]. Bu da Lj'nin sınırlı boyutlu olduğudur. Ulusal uyum için dim(Lj)'nin tüm j için aynı olduğunu ummalıyız, fakat bu son tahmin hiçbir şekilde gerekli değildir. Her bir Lj için ortonormal dayanaklar kullanıyoruz ve onları n × q matrislerinde Gj topluyoruz. Burada q, Lj'nin ortak boyutluluğudur ve n H'nin boyutluluğudur. Eğer n sonlu değilse bu durumda Gj basit olarak H'nin elemanlarının düzenli q-değişkenler grubudur ve kullandığımız tüm "matris işlemleri" kendi açık yorumlarına sahiptir. Böylece, sadece q vektörü α_j , $\alpha_j^{'}\alpha_j = 1$ formülüne sağlarsa, $y_j = G_j\alpha_j$, $L_i S$ 'in içindedir. Bu durumda, C_{jl} 'nin, $C_{jl} = G_j^{'}G_l$ tarafından tanımlanan q x q matris olduğu ve de $r_{jl}(y,...,y_m) = \alpha_j^{'}C_{jl}\alpha_l$ formülünü sağlamaktadır. Bütün j'ler için, q dizisinin tanımı olan $C_{jj} = 1$ in doğru olduğunu kabul ediyoruz. n x mq süpermatrisini $G = (G_1 | ... | G_m)$ ve mq x mq süpermatrisini $C = G^{'}G$ olarak tanımlamak da uygundur.

Hedeflerimiz için kullanışlı bir işlem, belirli bir sayıdaki matrislerin *direkt toplamıdır* [65]. Eğer X ve Y a x b ve c x d boyutlarının matrisleriyse, direkt toplam (a + c) x (b + d) matris

$$x + y = \left[\begin{array}{c|c} x & 0 \\ \hline 0 & y \end{array}\right] \text{dir.}$$

İkiden fazla matrisin direkt toplamlarının uzantısı bellidir. Artık A'yı mq×rn matris $\alpha_1 + ... + \alpha_m$ olarak tanımlayacağız. Bu formülde A' x A = I olduğunda R(y1, · · · , ym) =A' x C x A 'dır. Bundan anlaşıldığı gibi, R(y1, · · · , ym) korelasyon matrisinin en büyük p özdeğerlerinin toplamı $\sigma_p(y_1,..., y_m)$ 'dir, burada T, T'T = I'i doğrulayan rn x p matrisinin üzerinde değişebilir halde olduğu aşağıdaki ifade de gözükmektedir.

$$\sigma_p(y_1,..., y_m) = \max\{trT^{'}A^{'}CAT\} \qquad (1)$$

Böylelikle, $y_j \in L_j S$ olduğunda, $\sigma_p(y_1,..., y_m)$'nin maksimizasyonu, T'nin T'T=I'yi doğruladığını düşünürsek. Bütün m x p matrisleri ve A'nın, $A = \alpha_1 + ... + \alpha_m$ olduğu varsayarsak. Ve aynı zamanda $A^{'}A = I$ formülünü sağlayan bütün mq x m matrislerinin geçerli olduğu $trT^{'}A^{'}CAT$ 'ın maksimizasyonu ile de aynı toplama ulaşabiliriz. Eğer A ve T bu kısıtlamaları sağlarsa ve U matris AT'nin mq x p ise U'U = I'dir ve U, q x p boyutlarının ve t_J'nin T'nin j sırası olduğu $U_J = \alpha_J t_j^{'}$ biçiminin de m altmartislerini içermelidir. Böylece U birinci sıranın karşı bloğunda yer alır, ve her bir L_i alt boşluğu bir bloğu tanımlar. Sonuç olarak şu formülü çıkarabiliriz:

$$\sigma_p(L_1,..., L_m) = \max trU^{'}CU \qquad (2)$$

Burada U, birinci sıranın karşı bloğunda yer alan bütün mw x p ortonormal matrisleri üzerinde değişir. NLPCA'nın problemi $\sigma_P(L_1,...,L_m)$ formülünü hesap etmek ve U'nun maksimizörünü istenen biçimde bulmak.

Şimdi NLPCA probleminin sabit denklemlerini çıkaracağız. 1 için Ω'nın p dizisinin simetrik bir matrisi olduğu

$$RT = T\Omega \qquad (3)$$

formülünü elde etmeliyiz.Tanımlama amaçları için Ω'nın diyagonal (köşegen) olduğunu kabul edebiliriz. R = A'CA olduğunu da önceden biliyoruz ve α_j'e göre 1'ın diferansiyeli aşağıdaki denklemi verir

$$\sum_{l=1}^{m} \gamma_{jl} C_{jl} \alpha_l = \theta_j \alpha_j \qquad (4)$$

Burada $\Gamma = \{\gamma_{jl}\}, \theta_j$'ın belirsiz çarpan olduğu ve α_j'nin $\alpha_j^{'} \alpha_j = I$ 'ı sağladığını $\Gamma = TT'$ den elde edebiliriz. Bir $\Theta = diag(T\Omega T')$ çözümündeki 3 ve 4'den anlaşıldığı üzere $(tr)(\Theta) = tr(\Omega)$'dir. NLPCA için çoğu algoritma belirli T için çözümün değiştirilmesi üzerine kuruludur.

3 ve 4 sabit noktalar bulmak için yakınsak algoritmalar kullanılsa da NLPCA probleminin matematiksel yapısına pek az ışık tutar. Örneğin, eğer 3 ve 4 birden fazla çözüme sahipse her şey açık değildir. Ve durum böyleyse, bu çözümler nasıl ilişkilidir?. Eğer örneğin p=1 ise, ve $y_j \in L_j S$'yi $R(y_1,...,y_m)$'ın en büyük öz değerinin maksimize olacağı şekilde seçersek, U'nun birinci kademenin karşı bloğunda bulunma gerekliliği kalkar ve böylece problem U'nun mq x 1'in boyutları olduğundan dolayı U yerine u yazdığımız (u'u = I) formül üzerinde u'nun maksimize edilmesidir. p=1 iken NLPCA probleminin çözümleri netice itibariyle C'nin öz vektörleridir. Örneğin, küçük bir yansıma p=m-1'i gösterir, $R(y_1,...,y_m)$'in en küçük öz değerinin minimize eden $y_j \in L_j S$ çözümü, gene aynı sonucu verir. Eğer v, C.v'v = 1 ve μ öz değerinin bir öz vektörü ise, bunu her biri q öğeli v_j bloklarına bölebiliriz. Eğer bir blok sıfırdan büyük ya da küçük ise $\alpha_j = \frac{v_j}{v_j^{'} v_j}$ belirleriz, eğer bir blok sıfır ise α_j isteğe bağlı uzunluk vektörü olur. Ayrıca bu da $t_j = \sqrt{v_j^{'} v_j}.i.e.u_j = v_j$ 'dir. Bu durumda T'nin m x 1 olduğunu varsayarsak, öğeleri t_j olarak yazılır. Aynı zamanda $\Theta_j = \upsilon v_j^{'} v_j$ ve $w = \mu$ olarakta düşünebilirz. μ'nun, $r_{jl} = \alpha_j^{'} C_{jl} \alpha_l$ öğesine karşılık gelen R matrisinin her zaman en büyük öz değeri olmamasını görmek olanaksızdır. Daha açık olarak belirtmek gerekirse, eğer μ, C'nin en büyük öz değeriyse, w=μ de karşı gelen R'nin de aynı şekilde en

büyük öz değeridir ve aynı şey C'nin en küçük değeri içinde geçerlidir. Ama böyle bir şey ara öz değerleri için doğru değildir [66].

8.4.5. Çekirdek temel komponent analizi (KPCA)

Çekirdek TBA için nitelik alanına çizilmiş, $\Phi(x_1),\ldots,\Phi(x_\ell)$ verilerimizin ortalanmış olduğunu, yani $\sum_{k=1}^{l}\Phi(x_k)=0$ olduğunu varsayalım. Kovaryans matrisi

$$\bar{C}=\frac{1}{l}\sum_{j=1}^{l}\Phi(x_j)\Phi(x_j)^T \tag{1}$$

için TBA yapmak üzere $\lambda V=\bar{C}V$ denklemini değerlendirecek olursak, $\lambda\geq 0$ özdeğerlerini ve $V\in F\setminus\{0\}$ özvektörlerini bulmamız gerekiyor. Bunun yerine (1)'i koyarsak, bütün V çözümlerinin , $\Phi(x_1),\ldots,\Phi(x_\ell)$ aralığında olduğunu görürüz. Bu demektir ki, eşit sistem

$$\lambda(\Phi(x_k).V)=(\Phi(x_k).\bar{C}V) \quad \text{for all k=1,...,l} \tag{2}$$

olarak düşünülebilir, ve $\alpha_1,\ldots,\alpha_l$ gibi katsayıları mevcut sayılabilinir, öyle ki

$$V=\sum_{i=1}^{L}\alpha_i\Phi(x_i) \tag{3}$$

(1) ve (3)'ü (2)'e yerleştirdiğimizde, ve ℓ x ℓ boyutunda K matrisini

$$K_{IJ}:=(\Phi(x_i).\Phi(x_j)) \tag{4}$$

elde ederiz, ayrıca α'nın $\alpha_1,....,\alpha_l$ girdilerini kullanarak sütun vektörünün gösterdiği

$$\ell\lambda K\alpha = K^2\alpha \tag{5}$$

duruma ulaşırız. (5)'in çözümlerini bulmak üzere sıfır olmayan değerler için

$$\ell\lambda\alpha = K\alpha \tag{6}$$

özdeğer problemini çözeriz. Açıkça görüldüğü gibi, (6)'nın tüm çözümleri (5)'i karşılamaktadır. Dahası (6)'nın herhangi bir ek çözümünün (3)'ün açılımında bir değişiklik yapmadığı ve dolayısıyla biz ilgilendirmediğini de söyleyebiliriz.

Sıfır olmayan Özdeğerlere ait α^k çözümlerini, F'de karşı gelen vektörlerin normalize edilmesini sağlayabilirz, yani (V^k * V^k) = 0. (3), (4) ve (6)'nın sayesinde bu,

$$1 = \sum_{i,j=1}^{l} \alpha_i^k \alpha_j^k (\Phi(x_i).\Phi(x_j)) = (\alpha^k.K\alpha^k) = \lambda_k(\alpha^k.\alpha^k) \tag{7}$$

olarak hesaplanır. Ana komponent ayrılması için

$$(V^K.\Phi(x)) = \sum_{i=1}^{l} \alpha_i^k (\Phi(x_i).\Phi(x)) \tag{8}$$ 'e

göre F'deki Özvektörlerin üzerine bir test noktası Φ(x)'in görüntüsünün iz düşümünü hesaplamalıyız.

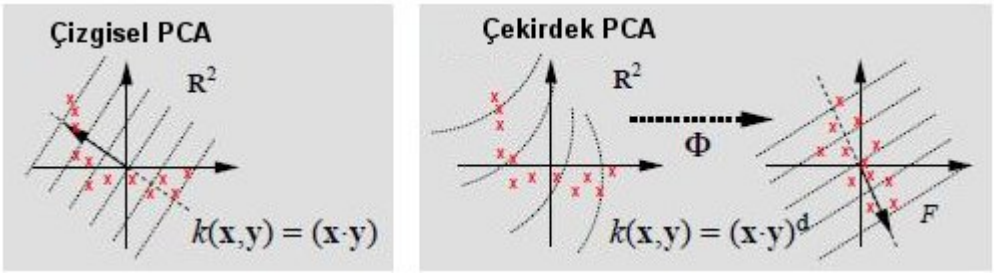
Ne (4)'ün , ne de (8)'in kesin biçmiyle Φ(x)'ye gerek duymadığına dikkat etmeliyiz; onlar sadece nokta ürünlerinde gereklidir. Dolayısıyla, harita da Φ'yi kullanmaya gerek kalmadan bu nokta ürünlerini hesaplamak için çekirdek fonksiyonlarını kullanabiliriz.

Bir k (x,y) çekirdeğinin bazı seçenekleri için fonksiyonel analiz yöntemleriyle (olasılıkla sonsuz boyutlu) k'nın F'deki nokta çarpımını hesapladığı bir harita Φ'nın varlığı gösterilebilir [67]. Destek Vektör Makinelerinde başarıyla kullanılmış olan çekirdeklere polinom çekirdekler denir [68].

$$k(x, y) = (x.y)^d \tag{9}$$

Radyal temel fonksiyonları k (x.y) = exp (||x y||^2 / (2σ^2)), ve sigmoid çekirdekler k(x.y) = tanh (K(x.y) + *(σ)*) buna dahildirler. d derecesinden polinom çekirdeklerin tüm d girdisi ürünleri tarafından kapsandığı ve nitelik alanına bir Φ haritasına karşılık geldiği gösterilebilir, örneğin N = 2.d = 2 durumunda aşağıdaki formül oluşmaktadır.

$$(x.y)^2 = (x_1^2, x_1x_2, x_2x_1, x_2^2)(y_1^2, y_1y_2, y_2y_1, y_2^2)^T \tag{10}$$



Şekil 8.1 Çekirdek TBA'nın Temel Fikri

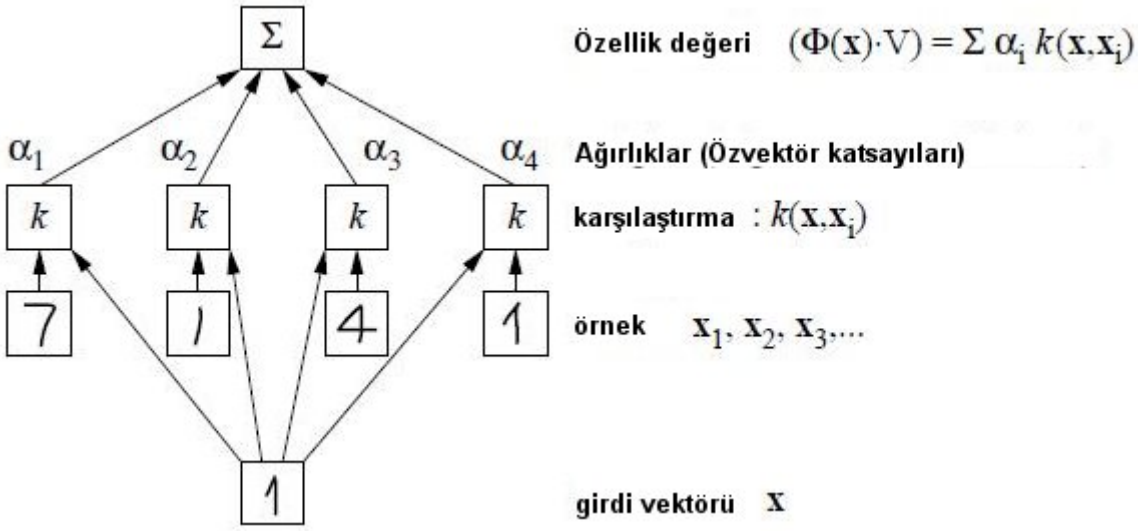
Şekil 8.1'de, standart nokta çarpımı yerine doğrusal olmayan bir çekirdek fonksiyonu kullanarak gerçekleştirilen TBA gösterilmiştir. Noktalı çizgiler, sabit nitelik değerlerinin kontur çizgileridir.

Eğer örüntüler görüntü ise, d piksellerinin tüm alanlarında çalışılınabilinir ve bu yolla TBA'yı gerçekleştirirken daha yüksek kademeli istatistikleri hesaba katabiliriz. (Φ (x) . Φ (y))'nin yerine çekirdek fonksiyonları koyduğumuz zaman çekirdek TBA bir algoritma elde ederiz (Şekil 8.1). Ayrıca nokta ürün matrisini (Denklem (4)) K_{ij} = $(k\ (x_i,k_j))_{ij}$ hesaplarız. (6)'yı, K'yı diyagonalize ederek çözeriz, Ayrıca Özvektör genişleme katsayıları α^n'i Denklem (7) ile normalize ederiz, ve bir test noktası x'in (çekirdek k'ya karşılık gelen) ana bileşenlerinin öz vektörler üzerine iz düşümlerini hesaplayarak (Denklem 8)'i buluruz (Şekil 8.2).

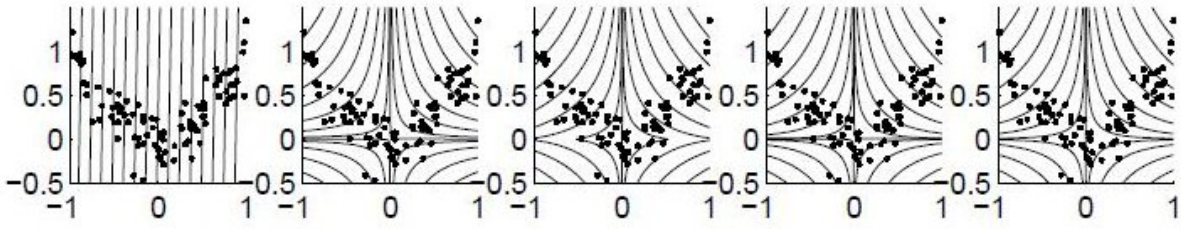
Pratikte, bu algoritmanın F nitelik alanına haritalama yoluyla doğrusal olmayan TBA'ya eşit olmadığını belirtmeliyiz. Ayrıca nokta çarpımı matrisinin sırası, örnek büyüklüğü ile kısıtlı olması ve de boyutluluk engelleyici olması bu matrisi hesaplamamızı imkansız hale getirmektedir. Örneğin, 16 x 16 piksellik girdi görüntüleri ve d = 5 polinom derecesi sonuç olarak 10^{10}'luk bir boyutluluk verir. Çekirdek TBA bu sorunu, otomatik olarak F'nin (K'nın derecesi tarafından belirlenen bir boyutlulukla) bir alt alanını seçerek ve bu alt alanda vektörler arasındaki nokta ürünlerinin hesaplanmasını sağlar. Bu yolla, 10^{10}'luk boyutsal alanda bir nokta çarpımı yerine, girdi alanında ℓ çekirdek fonksiyonu kullanmamız yeterli olucaktır.

Bu konuyu sonlandırırken, $\Phi(x_1)$'in F'de ortalanmış olduğu varsayımından vazgeçtiğimizden kısaca söz edelim. Kesin şekliyle elimizde olmayan bir noktalar

dizisinin ortalamasını hesaplayamıyacağımız için genelde verileri ortalayamıyacağımıza dikkat etmeliyiz. Bunun yerine, $\bar{\Phi}(x_i) := \Phi(x_i) - (1/\ell)\sum_{i=1}^{l} \Phi(x_i)$ kullanarak yukarıdaki cebiri gerçekleştirmek zorundayız. Bu durumda diyagonalize etmek zorunda olduğumuz matrisi (K açısından değerlendirirsek), $\bar{K} = K - 1_\ell K - K 1_\ell + 1_\ell K 1_\ell$ şeklinde ifade edebiliriz [69].



Şekil 8.2 Bir OCR görevi için çekirdek TBA nitelik ayırımı (test noktası x, Özvektör V)



Şekil 8.3 Çekirdek (11, derece d = 1….5) İle TBA.

Şekil 8.3'te de gösterildiği gibi, 100 nokta $((x_i)_1,(x_i)_2),(x_i)_2=(x_i)_1^2+$ gürültü (Gauss, standart sapma 0.2)'den oluşturuldu. Bütün $(x_i)_j$,'ler, $(x_i)_j \mapsto \mathrm{sgn}((x_i)_j).|(x_i)_j|^{1/d}$ 'e göre yeniden boyutlandırıldı. Gösterilen ilk ana bileşenin sabit değerli kontur çizgileridir. Doğrusal olmayan (d > 1) çekirdekler v ortalama varyans yönünde artan nitelikleri gösteriyor. Sonuç olarak doğrusal TBA (d = 1) de bu bakımdan iyi bir sonuç verir, ama sadece doğrusal yönlerde işe yarar.

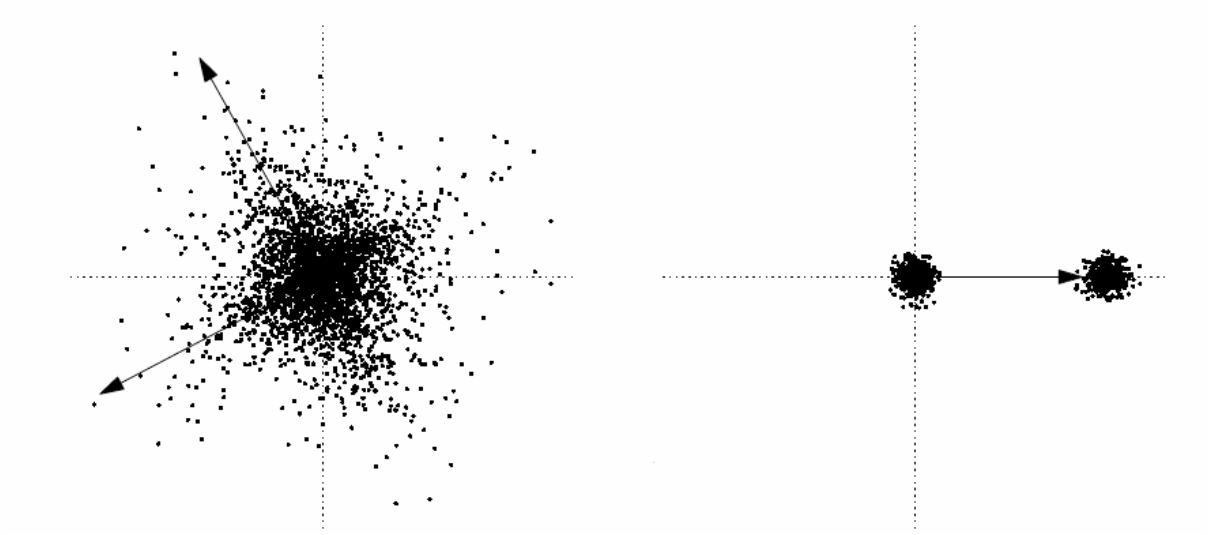
8.4.6 TBA ve ICA'nın karşılaştırılması

Veri madenciliğinde temel bileşen analizi (TBA) ve bağımsız bileşen analizi (ICA) nitelik seçimi için çok yaygın olarak kullanılmaktadır. TBA, ilişkili olma olasılığı olan bir dizi değişkeni ana bileşenler denilen daha az sayıda ilişkisiz değişkene çeviren matematiksel bir işlemdir. Veri setinin boyutluluğunu saptayabilir veya küçültebilir, ve gürültünün etkilerini kaldıracak altta yatan, yeni, anlamlı değişkenler tanımlayabilir. Diğer tarafta çoğunlukla zaman dizisi verilerinde nitelik indirgemesi için kullanılan ICA, rasgele değişkenler, ölçümler ya da sinyallerin altında yatan gizli faktörleri ortaya çıkarmak için bir istatistiksel ve sayısal tekniktir [70].

8.4.7 İz düşüm izleme (PP)

Bu algoritmanın ele aldığı en önemli konu nöron niteliklerinin, çevre değişimi ile değiştirilmesidir. Bu bize biyolojik deneyler için simülasyon yapma şansı vermiştir ve parametre bağımlılığından kaynaklanan bazı öngörüler sağlamıştır. İz düşüm izleme ile bu konuları daha detaylı olarak inceleyelim [71]. Öncelikle denklemleri daha iyi bir hale getirebilmek için nöronları etkileyen çevrenin niteliklerini anlamamız gerekir. Ve çevredeki etkileyen niteliklerin kurallarının da öğrenilmesi

gerekir. İz düşüm izleme yönteminde, nöronlardaki öğrenme sürecini, optimizasyon süreci olarak varsayılır. Sinapsların modifikasyonu bu durumda maliyet fonksiyonunun maksimizasyonu olarak görülebilir ki bu da, bir bakıma, girdilerdeki yapının ölçümü demektir. Maliyetlerin maksimizasyonu şeklindeki ifade düzensiz olsa da, bu ibareyi özellikle dikkate alacağımız maliyet fonksiyonlarıyla uyumlu olduğu için kullanıyoruz. Diaconis ve Freedman yüksek boyutlu verilerin, alçak boyutlu iz düşümlerinin hemen hemen Gaussian olduğunu göstermektedirler [72]. Son bulgularda verilerdeki önemli bilgilerin tek boyutlu iz düşüm dağılımının Gaussian'dan uzak olan yönlere aktarıldığını göstermektedir.Gaussian'dan uzağa terimi benzersiz değildir: bir kimsenin bulmak istediği yapıya bağlıdır. Şekil 8.4'de gösterilen 2 örnek, yüksek kürtosis ve multi modalite yöneleridir. Bu çalışmadaki veri, nöronların girdileridir(x-ekseni). Ayrıca verinin projekte edildiği yön de ağırlık vektörü dür. İz düşüm değeri nöron çıkışı ise y-ekseni'dir. Sigmoid'i daha önce biyolojik olasılık olarak görmüştük, ancak burada onun fonksiyonu maliyet fonksiyonunun şekline bağlıdır. Maliyet fonksiyonu bu durumda, çıktının istatistiğinin bir fonksiyonudur *y*. Sinaptik modifikasyon denklemi, bu maliyet fonksiyonunun ağırlıklar açısından aşamalı olarak elde edilir. Sonuç olarak Nöron, maliyet fonksiyonunun maksimize edebilecek yönleri arar [73].



Şekil 8.4 Yüksek kürtosis Ve Multimodalite Yönleri

8.4.8 Fisher doğrusal ayırtacı

Fisher Doğrusal Ayırtacı, doğrusal sınıflandırma modelini küçültülmüş şekilde görebilmenin tek yoludur. İlk önce iki sınıflı bir durum ele alalım, ve D- boyutsal giriş vektörü olarak x'i alıp, aşağıdaki formüle yerleştirelim.

$$y = w^T x \qquad (1)$$

Şayet y 'yi sınırlandırısak ve $y \geq w^0$ 'ı C1 sınıfı olarak kabul edersek, ve C2'yi de tam tersi sınıflandırırsak, standart doğrusal sınıflandırıcımızı elde ederiz. Genel olarak, tek boyut üzerine yapılan iz düşüm önemli miktarda bilgi kaybına sebep olur, ve orijinal D-boyutlu alanda uygun biçimde dağıtılmış sınıflar tek boyut üzerinde üst üste çakışabilirler. Ancak, ağırlık vektörü(w) bileşenlerini ayarlayarak, sınıf ayırımını maksimum seviyeye çıkaran bir iz düşüm seçebiliriz. Başlangıç için, C1 sınıfının N1 noktalarının ve C2 sınıfının N2 noktalarının bulunduğu ve ana vektörleri aşağıdaki gibi olan iki-sınıflı bir problemi varsayalım.

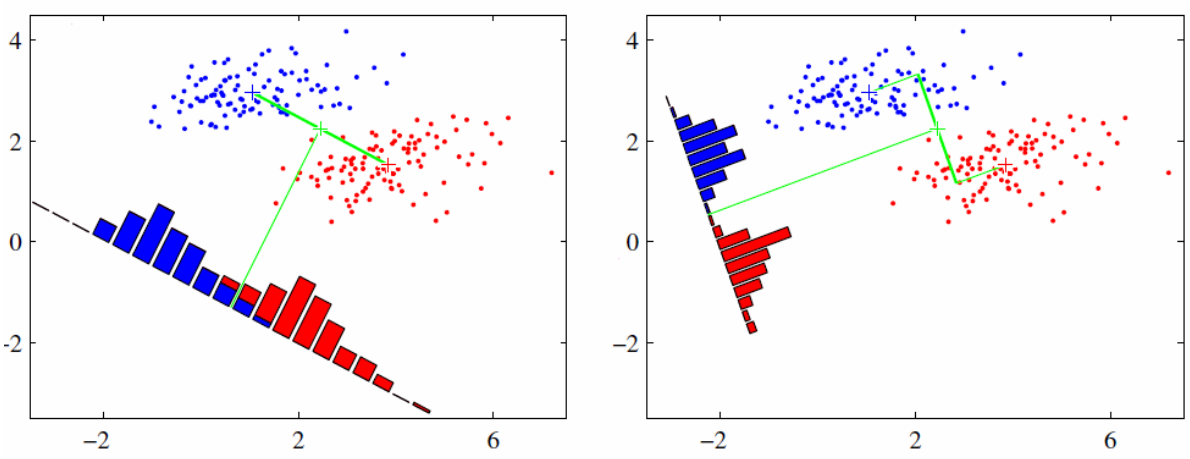
$$m_1 = \frac{1}{N_1} \sum_{n \in C_1} x_n \qquad m_2 = \frac{1}{N_2} \sum_{n \in C_2} x_n \qquad (2)$$

w üzerine formula kurduğumuzda ise, sınıf ayrımının en kolay ölçümü, iz düşüm yapılan sınıf araçlarının ayrılması olarak düşünebiliriz. Bu bize w 'yi seçebilmemizi sağlar ve alttaki denklemi maksimuma çıkarır.

$$m_2 - m_1 = w^T (m_2 - m_1) \qquad (3)$$

Diğer bir deyişle

$$m_k = w^T m_k \qquad (4)$$



Şekil 8.5 Fisher Doğrusal Ayırtacı Histogramları

Şekil 8.5'te, soldaki çizim histogramlar boyunca sınıf araçlarını birleştiren hat üzerinde projekte edilmesine bağlı olarak ortaya çıkan iki sınıftan (kırmızı ve mavi ile gösterilmiştir) örnekler göstermektedir. Değerlendirilen alan içerisinde oldukça fazla sınıf çakışması olduğu unutulmamalıdır. Sağdaki çizim, en yüksek seviyede iyileştirilmiş sınıf ayrımını gösteren Fisher doğrusal ayırtacı tabanlı uygun iz düşümü göstermektedir.

Ck, sınıflardan iz düşümü yapılan verilerin aracıdır. Ancak, bu ifade basitçe, w'nun büyüklüğü artırmak amacıyla rastgele yapılabilinir. Bu sorunu çözmek için, w 'yu birim uzunluk almak için sınırlayabiliriz, dolayısıyla $\sum_i w_i^2 = 1$ şekline gelir. Sınırlanan maksimizasyonu uygulamak için Lagrange kullanabiliriz, daha sonra w'yi $\propto$ olarak buluruz. (m2 –m1). Bu yaklaşımla ilgili Şekil 8.5'de de belirtildiği gibi halen bir sorun bulunmaktadır. Bu orijinal iki boyutlu uzayda ayrılan (x1, x2) malesef araçlarını bağlayan hat üzerinde iz düşümü alındığında, birbirleriyle çakışan iki sınıf olduğunu göstermektedirler. Bu zorluk, sınıf dağılımının aşırı diyagonal olmayan ortak değişkelerinden kaynaklanmaktadır. Fisher tarafından önerilen fikrin amacı ise; herbir sınıf içinde küçük oranda değişim veren, aynı zamanda iz düşümü yapılan sınıf araçları arasında geniş ayırım veren ve dolayısıyla da sınıf çakışmasını en aza indirgeyen fonksiyonu maksimuma çıkarabilmektir. İz düşüm formülü (1) x'de etiketlenmiş veri noktaları setini, tek boyutlu uzay da, y'de etiketlenmiş sete dönüştürür. Sonuç olarak Ck sınıfından dönüştürülen verilerin sınıf içi değişimi şu şekilde gösterilir.

$$s_k^2 = \sum_{n \in C_k} (y_n - m_k)^2 \quad (5)$$

Burada y_n = w.T.x_n dir.

Fisher kriteri; sınıf arası değişiminin, sınıf içi değişimine oranlanması gerektiğini bulmuştur ve aşağıdaki gibidir.

$$J(w) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2} \quad (6)$$

w değişkeni ile (1), (4), ve (5)'i Fisher kriteriyle ve de SB nin sınıf arası eşdeğişken matrisi olacak şekilde aşağıdaki gibi yazılmıştır.

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \quad (7)$$

Daha sona bu formül ile aşağıdaki formülü

$$S_B = (m_2 - m_1)(m_2 - m_1)^T \quad (8)$$

birleştirerek, SW'nin aşağıdaki formüldeki gibi, toplam sınıf içi ortak değişken matrisi olacak şekilde bağ kurabiliriz.

$$S_w = \sum_{n \in C_1} (x_n - m_1)(x_n - m_1)^T + \sum_{n \in C_2} (x_n - m_2)(x_n - m_2)^T \quad (9)$$

(7)'nin türevini aldıktan sonar, w ile oranlarsak, aşağıdaki formülde J(w)'yi maksimuma çıkarmış oluruz.

$$(w^T S_B w) S_w w = (w^T S_w w) S_B w \quad (10)$$

(8)'den, S_B.w 'nun daima (m2−m1) doğrultusunda olduğu görülmüştür. Ayrıca, bizim için w'nun büyüklüğü önemli değildir, yalnızca yönü önemlidir, dolayısıyla

skaler faktörleri olarak $(w^T S_B w)$ ve $(w^T S_w w)$ 'yi düşürebiliriz. (10)'un her iki tarafını da S_w^{-1} ile çarparak aşağıdaki durumu elde ederiz.

$$w \infty S_w^{-1}(m_2 - m_1) \qquad (11)$$

Sınıf içi ortak değişkenin yani S_w'nin birim matrise orantısal olacak şekilde izotropik olması durumunda, yukarıda da belirtildiği gibi w 'nun sınıf araçlarına orantısal olduğu bulunacaktır.

Sonuç olarak (11), Fisher'in doğrusal ayırtacı olarak bilinir, ancak; ayırtaç olmaktan ziyade daha çok bir verinin bir boyuta iz düşümünde belirli bir seçim niteliğindedir. Ancak, tasarlanan veriler bir y_0 tabanı seçerek (C1 eğer y(x) $\geq y_0$ ise ve C2'yi de başka türlü sınıflandırabiliyorsa) ayırtaç oluşturmak için kullanılır.

8.4.9 Çoklu durumlarda fisher ayırtacı

Fischer ayırtacının genellemesini K >2 sınıfları için de dikkate almalıyız, ve giriş uzayında ki D boyutluluğunun sınıflardaki K sayısından büyük olduğunu kabul etmeliyiz. Daha sonra, D' > 1 için linear niteliklerinin $y_k = w_k^T x$ olduğunu , ve burada k = 1, . . . , D' olacağını varsayarsak, bu nitelik değerleri uygun şekilde vektör y oluşturmak için gruplandırılacaktır. Benzer şekilde, $\{w_k\}$ ağırlık vektörleri W matrisinin kolonları olarak varsayılacak, yani

$$y = W^T x \qquad (12)$$

Tekrar, unutulmamalıdır ki y'nin tanımında herhangi bir eğilim parametresi dahil edilmemektedir. Sınıf içi ortak değişken matrisinin, K sınıfı durumlara genellemesini (9)'dan yapılır ve şu sonucu verir.

$$S_w = \sum_{k=1}^{K} S_k \qquad (13)$$

Diğer bir deyişle;

$$S_k = \sum_{n \in C_k} (x_n - m_k)(x_n - m_k)^T \quad (14)$$

$$m_k = \frac{1}{N_k} \sum_{n \in C_k} x_n \quad (15)$$

Ayrıca N_k da, Ck sınıfındaki desen sayısıdır. Sınıf arası ortak değişken matrisi genellemesinde Duda ve Hart ‘dan yola çıkabiliriz ve herşeyden önce toplam ortak değişken matrisini dikkate alırız [74].

$$S_T = \sum_{n=1}^{N} (x_n - m)(x_n - m)^T \quad (16)$$

Burada m toplam veri setinin

$$m = \frac{1}{N} \sum_{n=1}^{N} x_n = \frac{1}{N} \sum_{k=1}^{K} N_k m_k \quad (17)$$

ve $N = \sum_k N_k$ toplam veri sayısını göstermektedir. Toplam ortak değişken matrisi, (13) ve (14) tarafından ve de ilave matris S_B tarafından verilen sınıf içi ortak değişken matrisi miktarına ayrıştırılabilinir.

$$S_T = S_w + S_B \quad (18)$$

Diğer bir deyişle

$$S_B = \sum_{k=!}^{K} N_k (m_k - m)(m_k - m)^T \quad (19)$$

Bu ortak değişken matrisleri x-uzayında belirlenmişlerdir. Şimdi D’-boyutsal y-uzayında tasarlanan benzer matrisleri de beliryebiliriz.

$$s_w = \sum_{k=1}^{K} \sum_{n \in C_k} (y_n - \mu_k)(y_n - \mu_k)^T \qquad (20)$$

ve

$$s_B = \sum_{k=1}^{K} N_k (\mu_k - \mu)(\mu_k - \mu)^T \qquad (21)$$ 'yi

bu şekilde gösterebiliriz.

$$\mu_k = \frac{1}{N_k} \sum_{n \in C_k} y_n \qquad \mu = \frac{1}{N} \sum_{k=1}^{K} N_k \mu_k \qquad (22)$$

Sınıflar arası ortak değişken büyük olduğunda; büyük ve sınıf içi ortak değişken küçük olduğunda; küçük olan bir skaler oluşturmalıyız. Şimdi artık bir çok olası kriteri ele alabiliriz [75]. Bir örnek

$$J(W) = Tr\{s_w^{-1} s_B\} \qquad (23)$$

Bu kriter daha sonra W iz düşüm matrisinin açık bir fonksiyonu olarak
şu şekilde yeniden yazılabilinir

$$J(w) = Tr\{(WS_w W^T)^{-1}(WS_w W^T)\} \qquad (24)$$

Bu türden bir kriteri maksimize edebiliriz, ama bu bir bakıma karmaşıktır,ve Fukunaga'da detaylı olarak anlatılmıştır [75]. Ağırlık değerleri ni hesaplıcak olursak; D'nin en büyük öz değerlerine uygun düşen öz vektörler $S_W^{-1} S_B$ 'yi kullanmamız gerekir.

Tüm bu kriterlerde genel olan ve vurgu yapılması gereken başka bir önemli sonuç daha

vardır. Bu da (19)'da S_B'nin K matris miktarından oluştuğu, ve her birinin iki vektörün dış ürünleri olduğu ve dolayısıyla birinci sırada olduğudur. Ayrıca, bu matrislerden yalnızca (K −1)'in (17) deki sınırlanmasının sonucu olarak

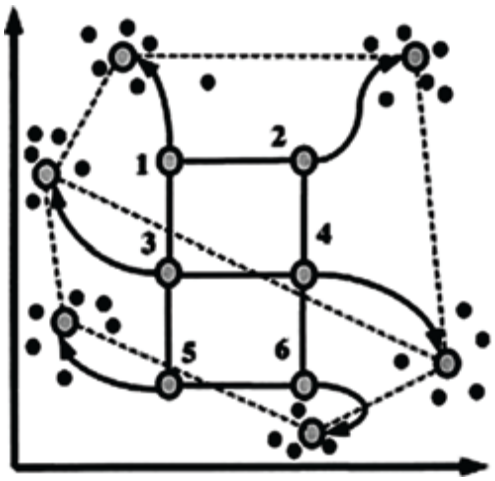
bağımsızdırlar. Böylece, S_B (K −1) ile eşit dereceye sahiptir ve çok fazla (K −1) değeri sıfır olmayan özdeğerler vardır. Bu, (K − 1) boyutsal alt uzayın üzerine yapılan iz düşümünün, S_B özvektörleri tarafından uzatıldığı anlamına gelir. Ayrıca J(w) değerinin değişmediğini ve, bu sayede (K − 1) linear 'niteliklerinden' daha fazla birşey bulamayacağımızı göstermektedir [76].

8.4.10 Kendi düzenlenen haritalar (SOM)

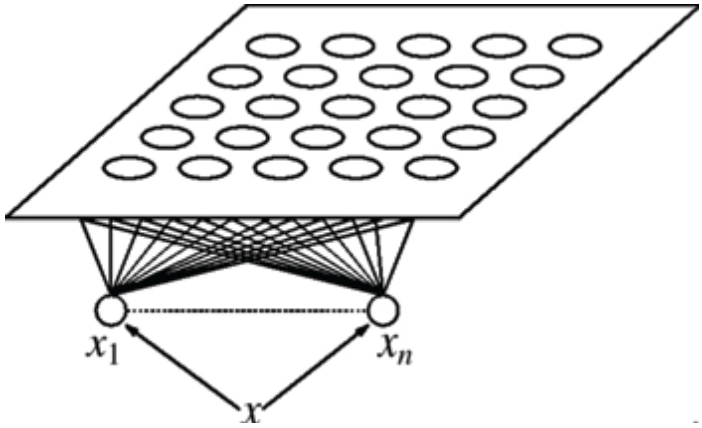
Tamayo, kendi düzenlenen haritalar (Self-organizing Maps – SOM) kullanarak tümör dokularının gen ifadesi verilerinin örüntülerini yorumlamakla ilgili bir makale yayınladı [77]. Ayrıca Ben-Dor gen ifadesi örüntülerinin kümelenmesi için çeşitli tekniklerin uygulanmasını vurgulayan bir makale yayınladı [78]. Carr ise gen ifadesi kümelenmesine bakmak için şablonlar oluşturdu [79]. Tıp alanında Wen de büyük boyutta gen ifadesi haritalama yöntemleri kullanarak merkezi sinir sisteminin gelişimini haritaladı [80]. Yukarıdaki çalışmalara bakarak gen ifadesi verilerinin kümelenmesi, görüntülenmesi ve diğer bir çok alanda SOM'un binlerce uygulaması olduğu görülmektedir.

Bu uygulamalar arasından en önemli sayabileceğimiz ise; Tamayo, öz örgütlemeli haritalar kullanarak tümör dokularında gen ifade verisinin düzenini yorumlama hakkında ki çalışmalarıdır [77] (Şekil 8.6).

SOM da aynı zamanda harici stimulide topolojik ilişkiler korunduğundan kompleks çok boyutlu veri alt alanında temsil edilmeye devam edilebilinir (Şekil 8.7). Genel olarak SOM'un ciddi sorunlarının hiçbirinin hiyerarşik sınıflandırma ve orantısal kümeleme üretmemesi çok büyük bir avantajdır. Fakat bunu sağlıyabilmek için, küme sayısı önceden sabitlenmelidir. Sonuç olarak, SOM herhangi bir ANN gibi *öncül* bilgi gerektirmektedir.



Şekil 8.6 Kendinden Organize Harita Örneği



Şekil 8.7 Nöronların Öğrenme Süreci [81].

SOM mevcut olan kümeleme tekniklerinden yaygın olarak kullanılanlardandır. Aynı zamanda bir veri görüntüleme tekniği olduğundan kendini düzenleyen nöral ağların kullanımı yoluyla da verilerin boyutlarını küçültür.

SOM'un önemli özellikleri şunlardır:

- Kohonen, SOM'u insan beyninin bilgileri mantıklı bir biçimde düzenleme yolunun basit bir analoğu olarak göstermiştir.
- Denetsiz öğrenim sürecine sahiptir.
- Bilgileri topolojik sırayla düzenler.

- Düzenleme sırasında ilişkiler korunur (topoloji koruyucu haritalama).
- Bir, iki veya çok boyutlu diziler şeklindeki bir nöron katmanından oluşur.
- Standart başlangıç geometrisi yoktur.
- Genellikle ilk girdilere bağımlıdır.
- İlk öğrenme sürecini kendi yapamaz; ama sonra denetimsizdir
- SOM çok boyutlu boyutlandırma algoritması ile kümeleme/vektör niceleme algoritması arasında bir yerde konumlandırılmıştır ve de etkili bir hesaplama algoritmasıdır.
- Uyum sağlama gücü zaman içinde sabittir.
- Çıktı haritasını, girdi vektörlerinin dağılımına uyarlamak için yeni hücreler eklenebilir ve var olan hücreler çıkarılabilir.

8.4.11 Korelasyon

Olasılık ve istatistik teorisinde, korelasyon iki rastgele değişken arasındaki linear ilişkinin direncini ve yönünü gösterir. Bu terimin günlük kullanımının aksine, linear olması gerekmez. Genel istatistiki kullanımda, korelasyon yada ko-relasyon iki rastgele değişkenin bağımsızlıktan ayrılmasını kasteder. Bu bağlamda, verinin türüne uyumlu olarak korelasyonun derecesini ölçen bir kaç katsayı vardır.

8.4.12 Pearson'un çarpım-moment katsayısı

Farklı durumlarda belirli sayıda katsayılar kullanılır. En iyi bilineni Pearson çarpım-moment korelasyon katsayısıdır, ve standart sapmaların çarpımları tarafından iki değişkenin ortak değişkesinin bölünmesiyle elde edilir.

8.4.13 Matematiksel Özellikler

İki değişken arasındaki korelasyon katsayısı $\rho_{X,Y}$ 'dir. Ayrıca http://en.wikipedia.org/wiki/Random_variables X ve Y ile beklenen değerler(μ_X ve μ_Y) ve standart sapmalar (σ_X ve σ_Y) olarak aşağıdaki gibi belirlenmiştir:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E((X-\mu_X)(Y-\mu_Y))}{\sigma_X \sigma_Y} \qquad (1)$$

ki burada E beklenen değer operatörü ve cov ortak değişken anlamına gelir. Yaygın şekilde kullanılan alternatif notasyonu ise;

$$corr(X,Y) = \rho_{X,Y} \quad (2)$$ 'dir.

$\mu_X = E(X)$, $\sigma_X^2 = E[(X - E(X))^2] = E(X^2) - E^2(X)$ olduğundan ve Y'yi dahil edersek, $E[(X - E(X))(Y - E(Y))] = E(XY) - E(X)E(Y)$ biçiminde olur, ayrıca şu şekilde yazabiliriz

$$\rho_{X,Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}} \quad (3)$$

Korelasyon, şayet her iki standart sapma da sonlu ve sıfırdan farklı ise belirlenir.

Artan linear ilişkisi durumunda korelasyon 1 dir, azalan da ise −1 dir http://en.wikipedia.org/wiki/Linear_dependence. Katsayı Hem −1 e hem de 1'e ne kadar yakınsa, değişkenler arasındaki korelasyon o kadar güçlüdür.

Mesela değişkenler bağımsız iseler bu durumda korelasyon 0'dır ancak, zıtlık doğru değildir çünkü korelasyon katsayısı yalnızca iki değişken arasındaki linear bağımlılığı tespit eder. Örnek: Rastgele değişken olan X'in −1 ile 1, ve $Y = X^2$ şeklinde olduğunu varsayalım. O zaman Y kesinlikle, X tarafından belirlenecektir, yani X ve Y bağımlıdırlar, ancak korelasyonları 0'dırhttp://en.wikipedia.org/wiki/Uncorrelated.

8.4.14 Örnek korelasyon

Şayet n serisi kadar X ve Y ölçümümüz varsa ve x_i ve y_i yazılmışsa ve burada i = 1, 2, ..., n ise, bu durumda Pearson çarpım-moment korelasyon katsayısı X ve Y 'nin korelasyonunun tahmini için kullanılabilir. Pearson katsayısı ayrıca "örnek korelasyon katsayısı olarak da bilinir". Pearson korelasyon katsayısı bu durumda

X ve Y nin korelasyonunun en iyi tahmini olacaktır. Pearson korelasyon katsayısı şu şekilde yazılır.

$$r_{xy} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y} \qquad (4)$$

Burada $\bar{x}$ ve $\bar{y}$ örnek http://en.wikipedia.org/wiki/Arithmetic_meanX ve Y araçlarıdır, s_x ve s_y isehttp://en.wikipedia.org/wiki/Standard_deviation X ve Y'nin örnek standart sapmalarıdır ve toplam da i = 1'den n'e kadardır. Populasyon korelasyonunda, bunu şu şekilde yazabiliriz.

$$r_{xy} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{(n-1)s_x s_y} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \qquad (5)$$

Yine populasyon korelasyonunda da örnek korelasyonun mutlak değeri 1 den az veya eşit olur. Yukarıdaki formül, örnek korelasyonların hesaplanması için tek geçiş algoritması önerse de, sayısal dengesizliği ile de dezavantaj sağlar.

Örnek korelasyon katsayısının karesi, katsayı belirleme olarak da bilinir. Ayrıca bu y_i deki değişimin fraksiyonudur ve bu x_i'nin y_i'ye linear uyumu ile hesaplanır . Yazılışı da şöyledir.

$$r_{xy}^2 = 1 - \frac{s_{y|x}^2}{s_y^2} \qquad (6)$$

Burada $s_{y|x}^2$http://en.wikipedia.org/wiki/Linear_regressionx_i'nin y_i üzerindeki http://en.wikipedia.org/wiki/Equation y = a + bx eşitliği ile linear regresyon hatasının karesidir:

$$s_{y|x}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - a - bx_i)^2$$

(7)

ve s_y^2 ise yalnızca y'nin değişimidir.

$$s_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2 \quad (8)$$

$$r_{xy}^2 = 1 - \frac{s_{x|y}^2}{s_x^2} \quad (9)$$

Bu denklem ayrıca daha yüksek boyutlar için korelasyon fikri de vermektedir. Örneğin, z = a + bx + cy şeklinde bir düzlemi (x_i , y_i , z_i) verileri koyarak uyarlarsak, bu durumda korelasyon katsayısı z - x ve y olur [82].

$$r^2 = 1 - \frac{s_{z|xy}^2}{s_z^2} \quad (10)$$

9. BULGULAR

Bu çalışmada, öncelikle orijinal veriler, indirgeme yapılmadan analiz edilmiş, daha sonra indirgeme algoritmaları veri kümesi üzerine uygulanmış, yeni veri kümesi tekrar analiz edilmiş ve sonuçlar çizelge halinde kaydedilmiş ve oransal olarak karşılaştırılmıştır. Analizler, dört çekirdekli bir işlemci ve 4 GB 1200 Mhz üzerinden hafızayla çalışan bir bilgisayarda yapılmıştır.

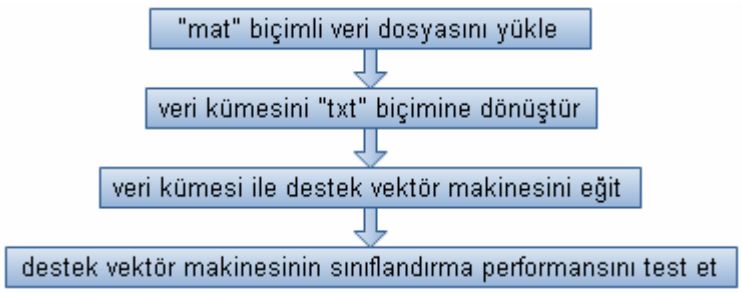
9.1 GEMS ve Orijinal Veri Analizi (GOVA)

Nitelik indirgeme potansiyelini ve başarısını ölçebilmek için, öncelikle hiçbir indirgeme yapılmamış orijinal veri kümesinin girdi olarak GEMS yazılımına verilerek yapılan eğitim ve test işlemlerinin sonuçları kaydedilmiştir (Çizelge 9.1).

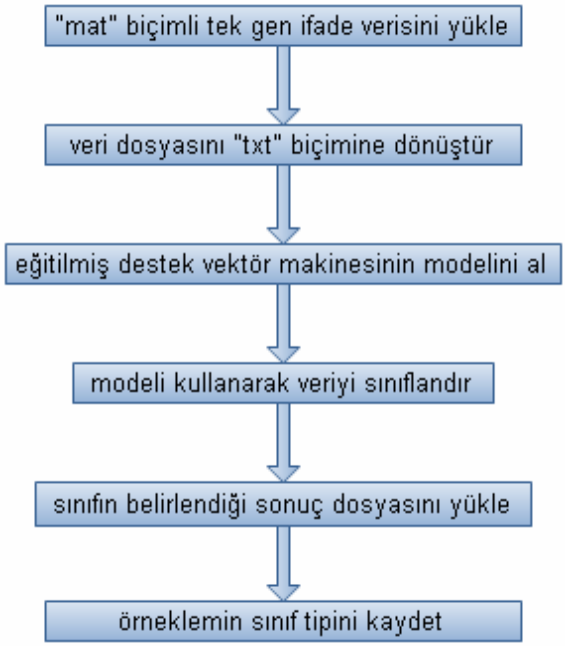
Çizelge 9.1 Herhangi bir indirgeme yapılmadan, veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :			**0**	
Veri Nitelik Sayısı :					**12534**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**0.0.0.0**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	95,54	**0**	1,51	54,1	323,18	**0**
OVO	90,22	**0**	1,48	55,21	288,1	**0**
DAG	90,22	**0**	1,43	56,18	287,87	**0**
WW	95,02	**0**	1,61	53,91	281,39	**0**
CS	95,02	**0**	1,53	50,69	281,85	**0**

9.1.1 Destek vektör makinesinin veri indirgemesi yapılmadan eğitilmesi ve testi



9.1.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.2 Nitelik Ortalaması ve Standart Sapma (NOSS)

Veri matrisinin satırlarının ortalamalarının herhangi bir benzerlik göstermesi beklenmemiştir ve sonuçlar bu beklentiye uygun bir şekilde çıkmıştır. Veri setinin satır ortalamalarında herhangi bir düzen tespit edilememiş ve benzerlik kuracak herhangi bir denklem geliştirilememiştir. Bu yüzden, veri kümesinin sütun ortalamaları analiz edilmiştir.

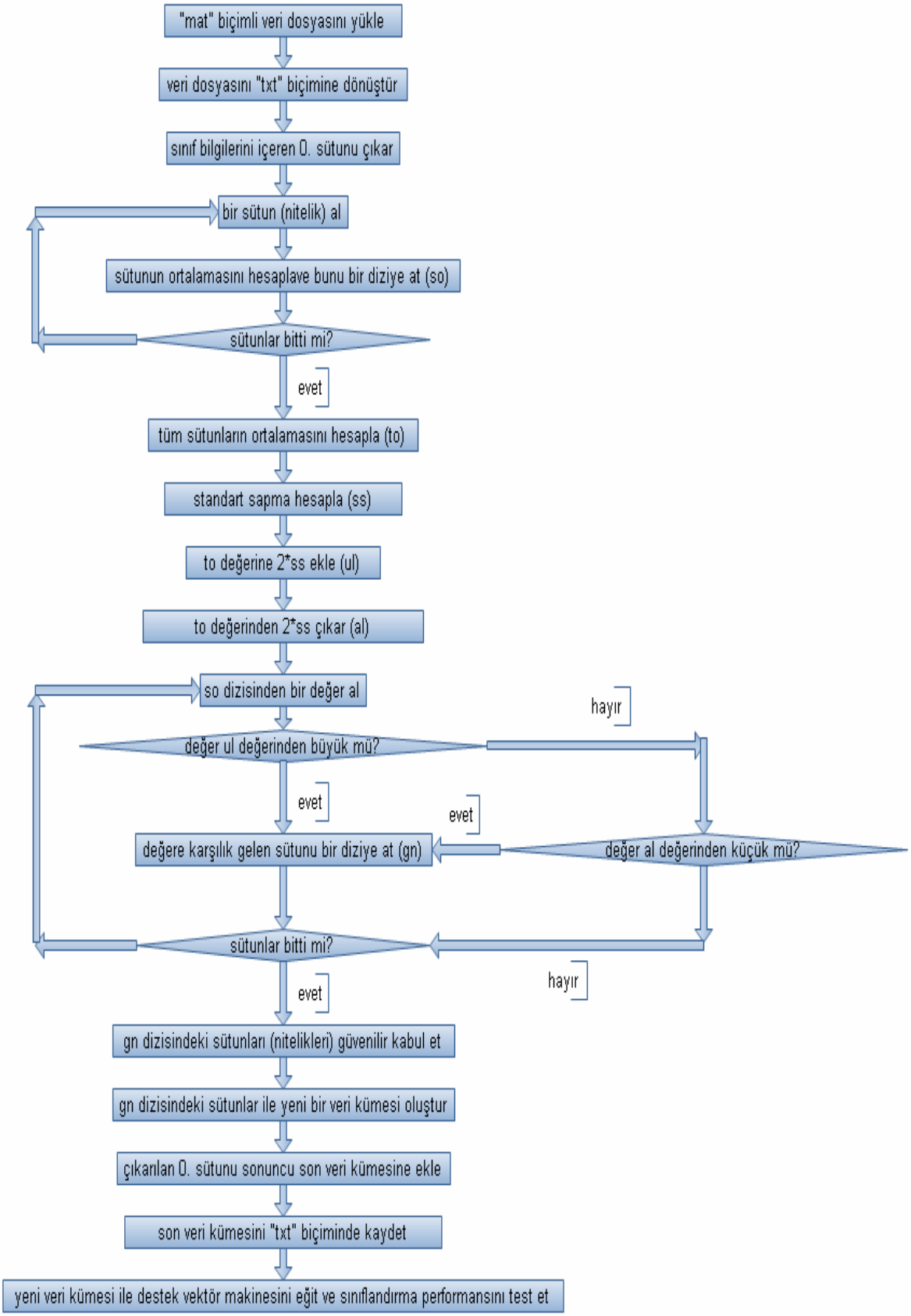
Veri kümesinin sütun ortalamaları da birbirlerinden oldukça farklı çıkmış ve yaklaşık **–**1500 ile -400 arasında bir dağılım göstermiştir. Standart sapma ise 856,1747 çıkmıştır. Bu işlemden sonra, +2sd -2sd aralığınının dışında kalan nitelikler güvenilir olarak kabul edilip, bunlar ile yeni bir veri kümesi oluşturulmuştur. Bu işlem atomik saat ile 23 dakika 17 saniye 12 salise olarak ölçülmüştür. Bu veri kümesi ile destek vektör makinesi eğitilip test edilip, sonuçlar not edilmiştir (Çizelge 9.2).

Çizelge 9.2 Nitelik ortalaması ve standart sapma işlemleri ile indirgeme yapıldıktan sonra, artı – eksi iki standart sapma aralığına girmeyen nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

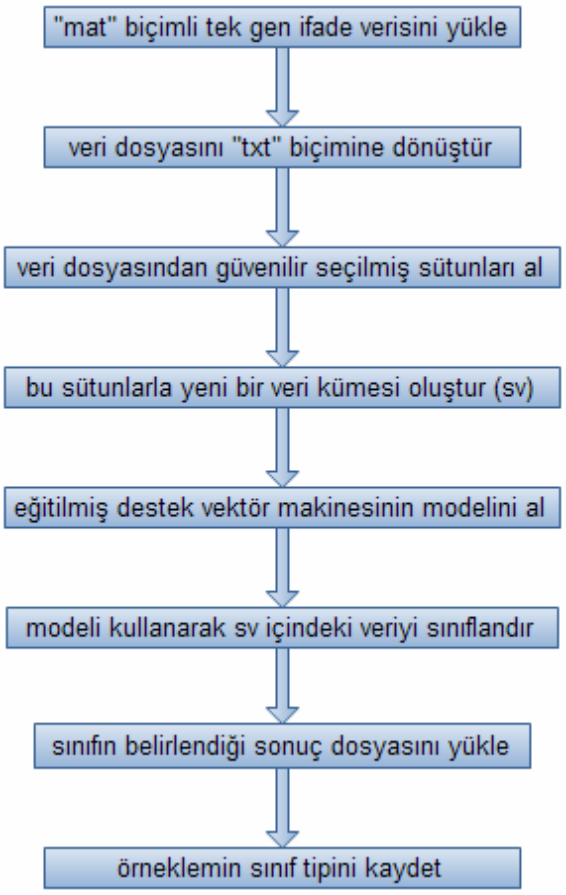
Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :	**95,23**			
Veri Nitelik Sayısı :					**598**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**0.23.17.12**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	91,16	**-4,38**	1,56	52,30	8,25	**97,45**
OVO	88,41	**-1,81**	1,51	51,11	7,59	**97,37**
DAG	88,41	**-1,81**	1,66	55,19	7,52	**97,39**

WW	89,30	**-5,72**	1,45	54,44	8,54	**96,97**
CS	89,34	**-5,68**	1,56	53,23	8,56	**96,96**

9.2.1 Nitelik ortalaması ve standart sapma ile nitelik indirgemesi; destek vektör makinesinin eğitilmesi ve testi



9.2.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.3 Bağımsız Bileşen Analizi (BBA)

Kullanılan veri kümesi üzerine bağımsız bileşen analizi uygulandığında, örneklem sayısı 174 iken sinyal sayısı 12534 olarak alınmıştır. Bu sayılar, veri kümesinin satır ve sütun sayılarına eşittir. Bağımsız bileşen analizi parametreleri varsayılan sabit parametreler olarak alınmıştır. Buna göre analiz, tanh doğrusalsızlığı ile ve (süpergauss verileri için maksimum olasılık tahminleri gibi) paralel olarak yapılmıştır. Bu hesaplama işlemi, dört çekirdekli bir işlemci ve 4 GB 1200 Mhz üzerinden hafızayla çalışan bir bilgisayarda işlenmiş; ancak işlemi tamamlayamadan hafıza yetersizliği yüzünden hata vermiş ve işlem atomik saat ölçümüyle 19,3 saniye içinde olağandışı olarak sonlandırılmıştır.

9.4 Temel Bileşen Analizi (TBA)

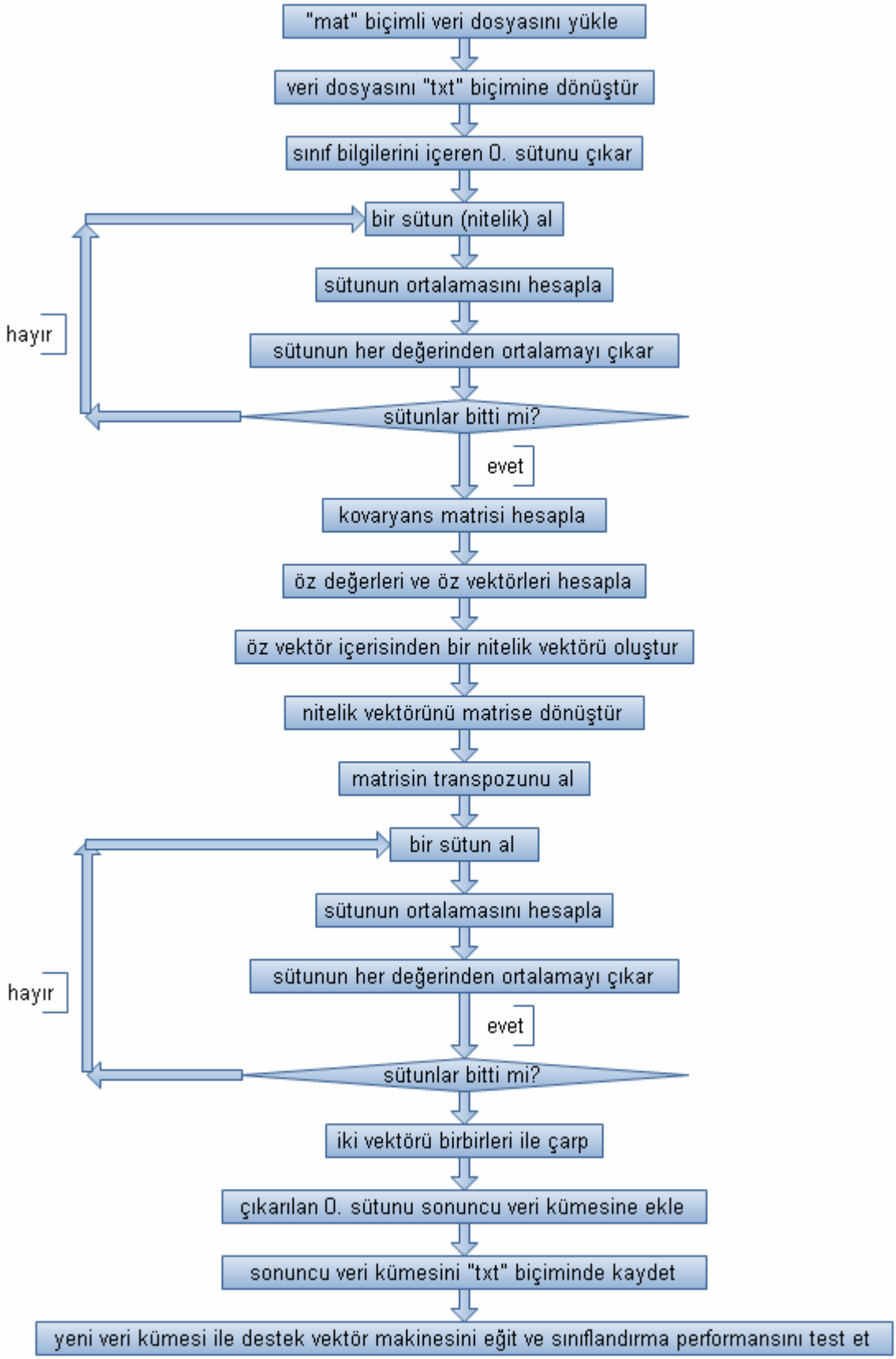
12534 veri kümesi niteliğinden, 12361 nitelik bu yöntem ile indirgenmiş, geriye sadece 173 nitelik kalmıştır. Bu indirgeme işlemi, atomik saat ile ölçüldüğünde 3 dakika, 6 saniye, 26 salise sürmüştür. İndirgeme işlemi sırasında ortalama 3,12 GB hafıza kullanılmış ve yine ortalama %31 işlemci tüketimi yaratılmıştır.

Temel Bileşen Analizi ile ana veri kümesi indirgendikten sonra, yeni veri nitelikleri ile oluşturulmuş yeni veri kümesi kullanılarak GEMS eğitilip test edildiğinde, aşağıdaki sonuçlara ulaşılmıştır (Çizelge 9.3).

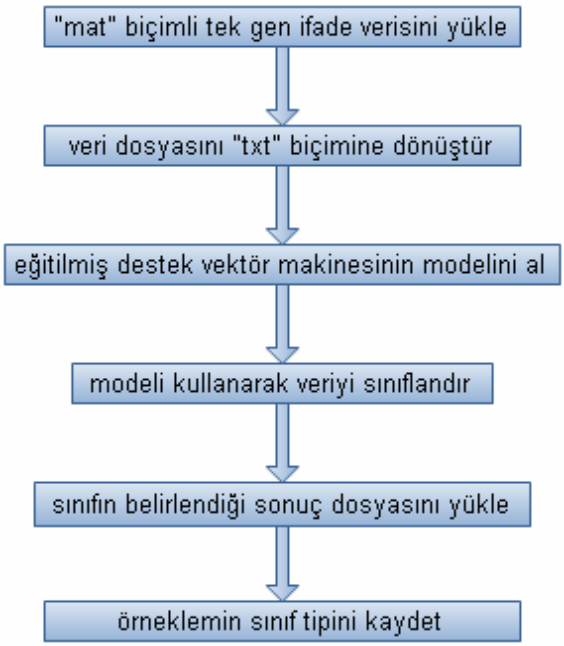
Çizelge 9.3 Temel Bileşen Analizi yapıldıktan sonra, yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :			**98,62**	
Veri Nitelik Sayısı :					**173**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**0.3.6.26**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	63,30	**-32,24**	1,35	51,50	5,05	**98,44**
OVO	59,19	**-31,03**	1,52	55,13	4,58	**98,41**
DAG	57,49	**-32,73**	1,51	51,67	4,40	**98,47**
WW	62,71	**-32,31**	1,57	55,55	6,92	**97,54**
CS	62,71	**-32,31**	1,45	54,90	7,81	**97,23**

9.4.1 Temel bileşen analizi ile nitelik indirgemesi; destek vektör makinesinin eğitilmesi ve testi



9.4.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.5 İz Düşüm Takip Analizi (İDTA)

İz Düşüm Takip Analizi ile indirgeme işlemi yapıldığı zaman, algoritmayı çalıştıran ve işleyen programın, atomik saat ile ölçüldüğünde 11 saat 17 dakika 45 saniye 56 salise sonra yanıt vermeyerek kilitlendiği tespit edilmiştir. Bu süre içerisinde neredeyse hiç hafıza kullanılmamış; ancak %90'lara varan işlemci gücü harcanması not edilmiştir.

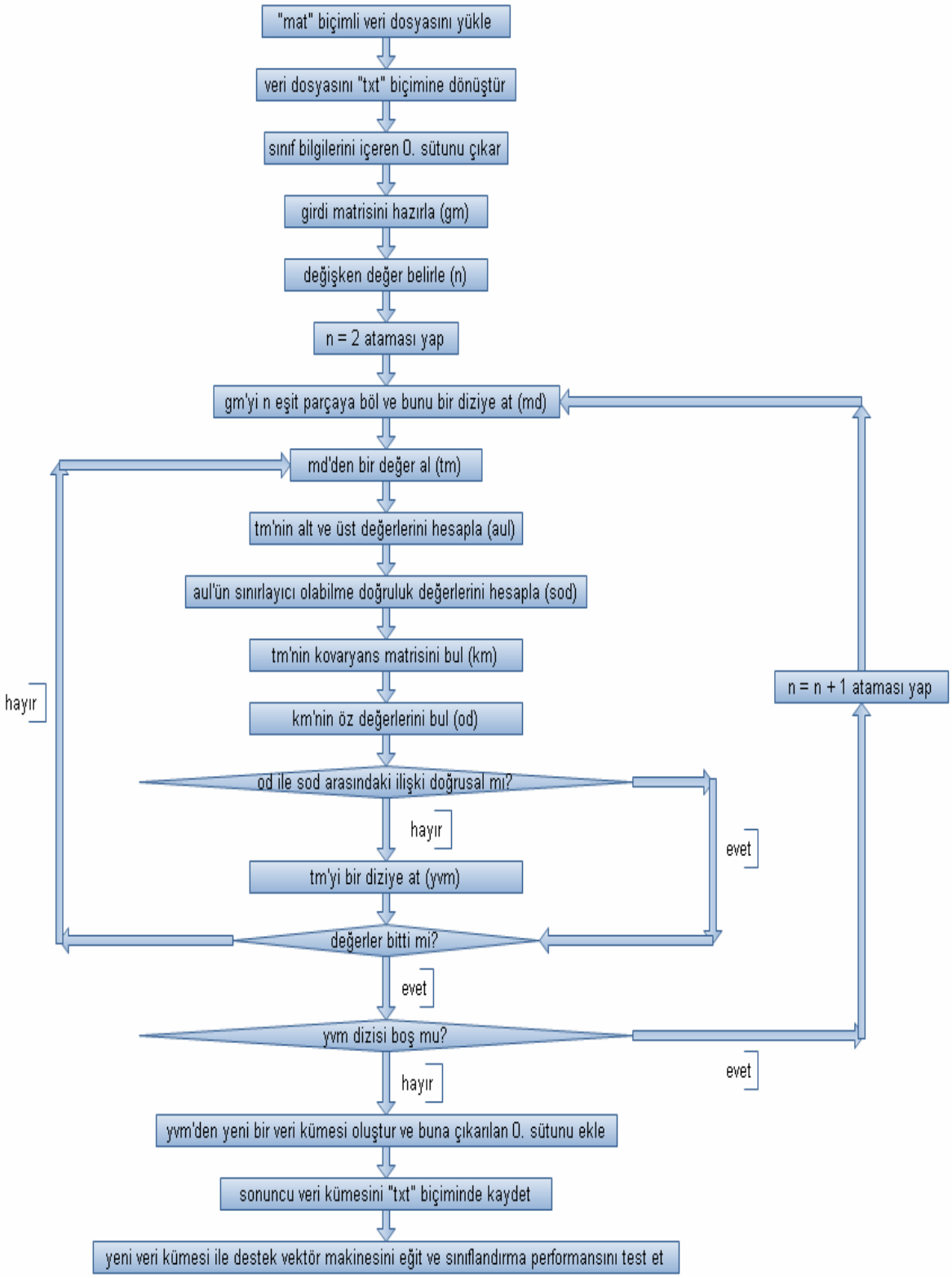
9.6 Doğrusal Olmayan Temel Bileşen Analizi (DOTBA)

Temel Bileşen Analizi'nin bir türevi olan ve doğrusal olmayan yöntemlerle indirgeme yapan bu yöntemi uygulayan program, atomik saate göre 9 saat 27 dakika 18 saniye 63 salise sonunda indirgemeyi bitirmiştir. İşlem sırasında ortalama 2,78 GB hafıza kullanılmış ve ortalama %23 işlemci tüketimi tespit edilmiştir. İndirgeme sonucunda, 348 nitelik ile yeniden bir veri kümesi oluşturulmuştur. Testler sonucunda aşağıdaki çizelge ortaya çıkmıştır (Çizelge 9.4).

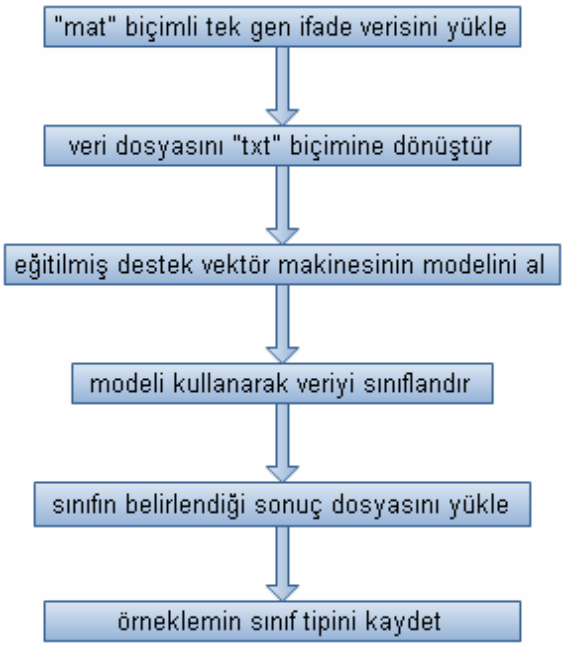
Çizelge 9.4 Doğrusal Olmayan Temel Bileşen Analizi yapıldıktan sonra, yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :		**97,22**	
Veri Nitelik Sayısı :				**348**	
İndirgeme Süresi (saat.dakika.saniye.salise)				**9.27.18.63**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	67,45	**-28,09**	1,57	54,30	7,73	**97,61**
OVO	65,76	**-24,46**	1,44	55,33	6,48	**97,99**
DAG	64,60	**-25,62**	1,55	54,20	5,13	**98,41**
WW	66,15	**-28,87**	1,57	55,11	9,17	**97,16**
CS	66,15	**-28,87**	1,62	57,87	8,88	**97,25**

9.6.1 Doğrusal olmayan temel bileşen analizi ile nitelik indirgemesi; destek vektör makinesinin eğitilmesi ve testi



9.6.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.7 Kendinden Organize Haritalar (KOH)

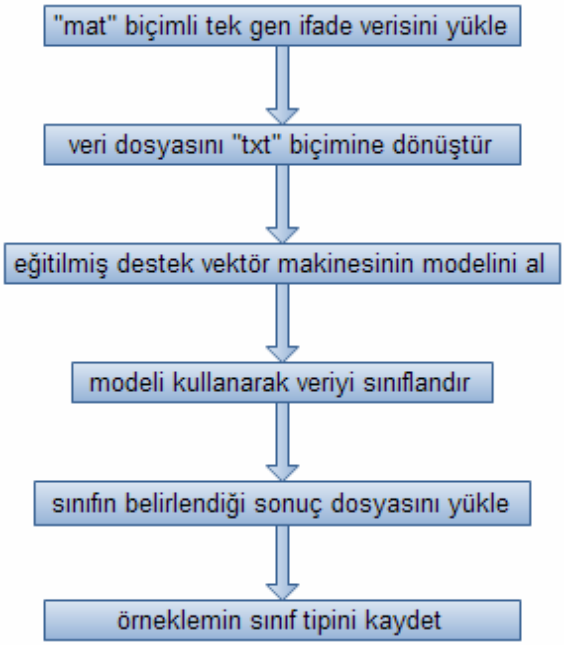
Kendi düzenlenen haritalar yöntemi ile indirgeme yapıldığında, indirgeme performansının iyi olmamasının yanı sıra, perfomans %40'lara kadar düşmüştür. İşlem boyunca ortalama 2,45 GB hafıza kullanılmış ve yine ortalama %45 işlemci kullanımı tespit edilmiştir. İndirgeme 3 saat 55 dakika 15 saniye 73 salise sürmüştür. Elde edilen sonuçlar aşağıda verilmiştir (Çizelge 9.5).

Çizelge 9.5 Kendi Düzenlenen Haritalar Analizi yapıldıktan sonra, yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :	**87,75**			
Veri Nitelik Sayısı :					**1536**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**3.55.15.73**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	45,75	**-49,79**	1,56	55,34	14,45	**95,53**
OVO	44,11	**-46,11**	1,56	54,89	13,11	**95,94**
DAG	45,34	**-44,88**	1,45	56,11	17,65	**94,54**
WW	45,20	**-49,82**	1,78	55,55	18,11	**94,40**
CS	45,20	**-49,82**	1,70	54,10	18,01	**94,43**

9.7.1 Kendi düzenlenen haritalar analizi ile nitelik indirgemesi; destek vektör makinesinin eğitilmesi ve testi

9.7.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.8 Doğrusal Diskriminant Analizi (DDA)

Doğrusal diskriminant analizi, bir tür indirgeme yönteminden çok bir sınıflandırma yöntemidir. Öncelikle, Fisher algoritması ile sınıflandırma yapılır ve aynı ya da

benzer olan nitelikler bir araya toplanır. Burada benzer niteliklerden kasıt, aynı sınıfa ait niteliklerdir. Böylece, birbiriyle benzer olan niteliklerden sadece bir tanesi alınıp, diğerleri veri kümesinden çıkarılır.

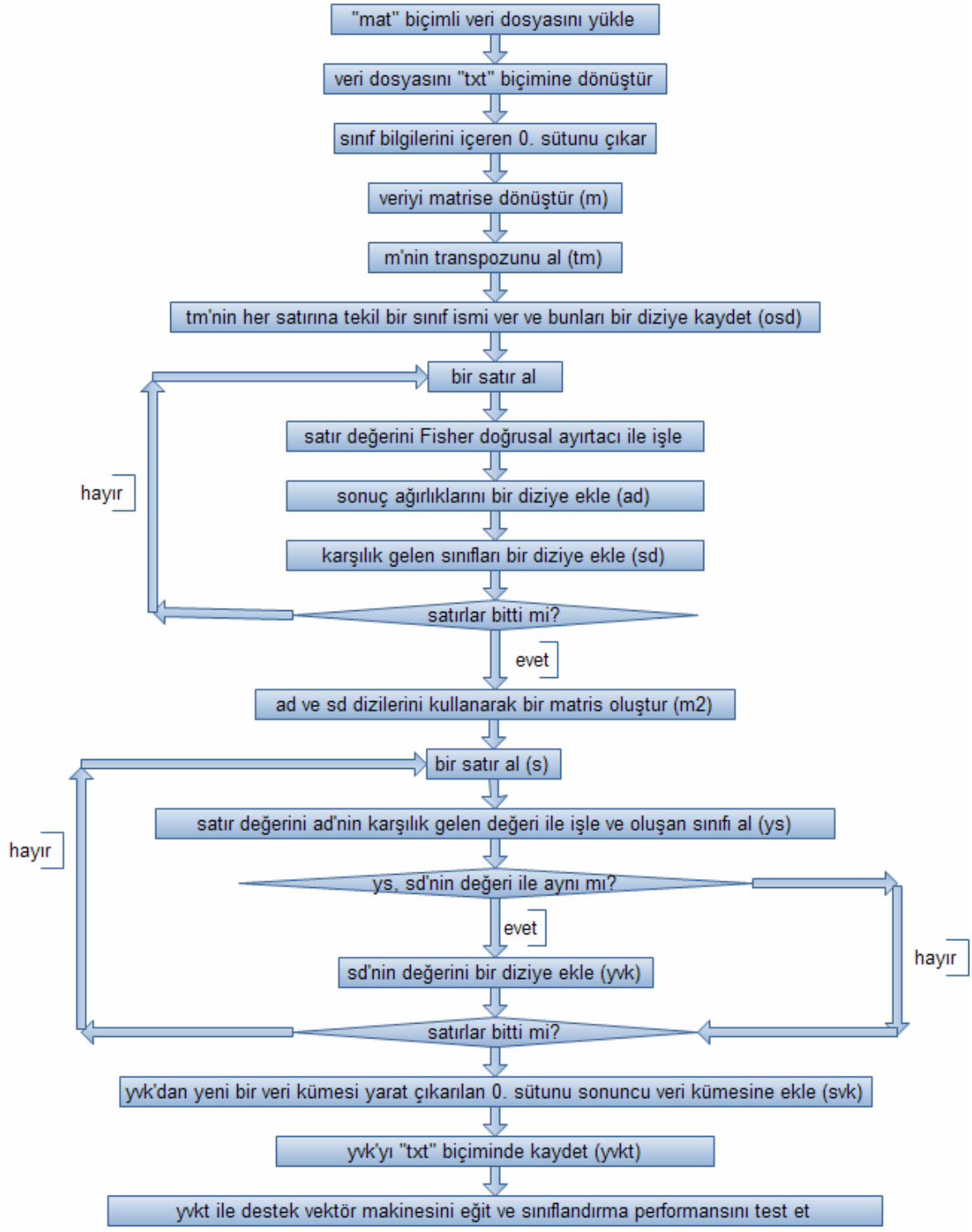
12534 veri kümesi niteliğinden, 6452 nitelik bu yöntem ile indirgenmiş, geriye sadece 6082 nitelik kalmıştır. Bu indirgeme işlemi, atomik saat ile ölçüldüğünde 3 saat, 17 dakika, 54 saniye, 43 salise sürmüştür. İndirgeme işlemi sırasında ortalama 3,77 GB hafıza kullanılmış ve yine ortalama %17 işlemci tüketimi yaratılmıştır.

Sınıflandırmadan sonra oluşturulan küme ile yapılan performans sonuçları aşağıdaki çizelgede verilmiştir (Çizelge 9.6).

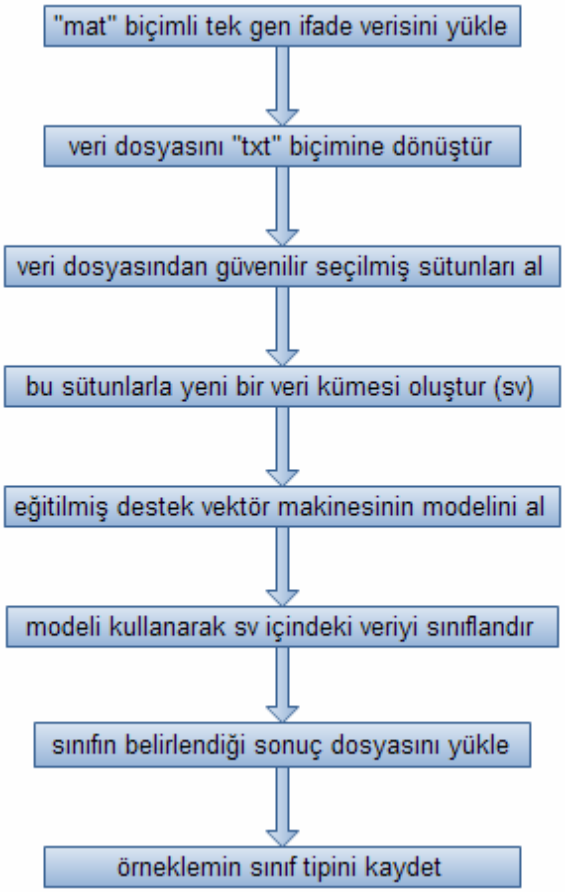
Çizelge 9.6 Doğrusal Diskriminant Analizi yapıldıktan sonra, yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :	**51,48**			
Veri Nitelik Sayısı :					**6082**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**3.17.54.43**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	96,13	**0,59**	1,44	52,30	142,24	**55,99**
OVO	90,75	**0,53**	1,52	54,11	125,91	**61,04**
DAG	90,75	**0,53**	1,49	51,11	125,24	**61,25**
WW	95,02	**0,00**	1,44	54,10	123,86	**61,67**
CS	95,02	**0,00**	1,59	57,44	123,83	**61,68**

9.8.1 Doğrusal diskriminant analizi ile nitelik indirgemesi; destek vektör makinesinin eğitilmesi ve testi



9.8.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.9 Çekirdek Temel Bileşen Analizi (ÇTBA)

Kullanılan veri kümesi üzerine çekirdek temel bileşen analizi uygulandığında, örneklem sayısı 174 iken nitelik sayısı 12534 olarak alınmıştır. Bu sayılar, veri

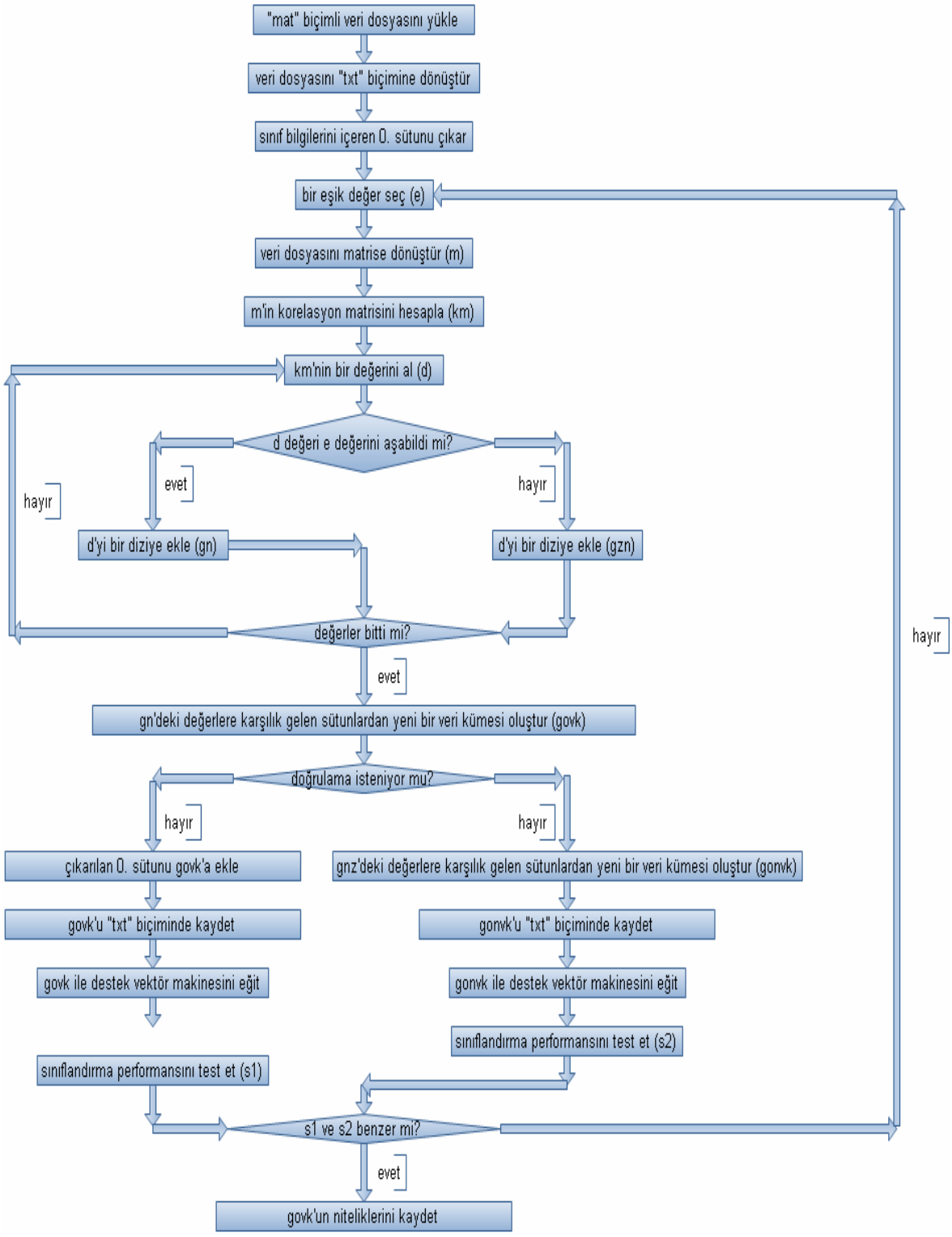
kümesinin satır ve sütun sayılarına eşittir. Çekirdek tipi poly olarak seçilmiştir. Kullanılan bilgisayarın yüksek kapasitesi rağmen, program işlemi tamamlayamadan hafıza yetersizliği yüzünden hata vermiş ve işlem atomik saat ölçümüyle üç dakika on yedi saniye altmış beş salise içinde olağandışı olarak sonlandırılmıştır.

9.10 Korelasyon Analizi (KA)

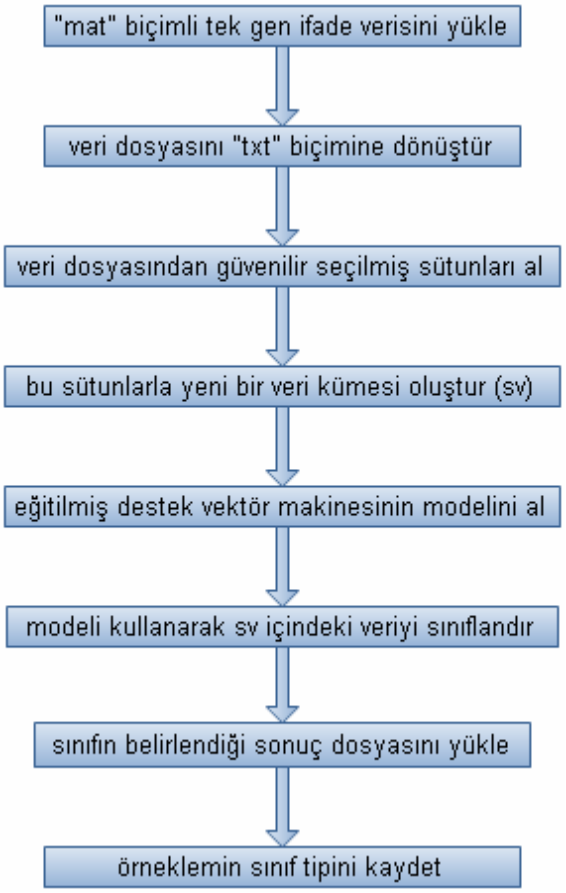
Veri nitelikleri arasında korelasyona bakmak, indirgeme yapmak için iyi bir yol olabilir. Birbirleriyle korelasyona girmiş nitelikler bir araya getirilebilir, ya da birbirlerinin benzeri sayılabilecek nitelikler orijinal veri kümesinden çıkarılabilirse, indirgeme yapmak mümkün olabilir.

Korelasyon analizini yorumlamak birçok farklı yöntemle olabiir. Burada ele alınacak yorumlama ise şöyledir; İki veri niteliği birbirleri ile, belirli bir eşik değer üzerinde korelasyona girdiyse, bu iki veri niteliğine güvenilebilir, aksi halde güvenilir değildir. Bütün korelasyon analizleri bittiğinde ise, güvenilir seçilmiş tüm veri nitelikleri toplanır ve yeni bir veri kümesi oluşturulur. DVM'nin performans – süre analizine göre, korelasyon analizinin eşik değeri yükseltilebilir ya da alçaltılabilir.

9.10.1 Korelasyon analizi ile nitelik indirgemesi; destek vektör makinesinin eğitilmesi ve testi



9.10.2 Eğitilmiş destek vektör makinesine verilen gen ifade verisine göre sınıflandırma



9.10.3 Kesme değeri 0,7 korelasyon

Kullanılan veri kümesi üzerine tam korelasyon uygulandığında, veri kümesinde bulunan 12534 nitelik üzerinden korelasyon matrisi hesaplanmış, bu hesaplamalar

sonucu, korelasyon değeri 0.7 üzeri ve -0.7 altı olan değerler alınmıştır. Bu hesaplama işlemi, atomik saat ile ölçüldüğünde 2 saat 45 dakika 11 saniye 88 salise sürmüştür. Bu süreç içerisinde, bilgisayarın işlemcisi toplam, yaklaşık ve ortalama %30 civarında kullanılmış, yine yaklaşık ve ortalama 1,96 GB hafıza kullanımı tespit edilmiştir.

Korelasyonu 0.7 değerinden kesmek, birbirine benzemeyen bazı nitelikleri birbirlerine benzemeye zorlamıştır. 0,7 eşik değeri normal bir korelasyon için düşüktür ve geniş bir ilişki değer aralığını kapsar. Korelasyon işleminden sonra, 12534 veri niteliği içinden, eşik değeri aşabilen 4731 nitelik parçası tespit edilmiştir. 7803 nitelik 0.7 eşik değerini aşabilen herhangi bir pozitif veya negatif koerelasyon değerine ulaşamamıştır. 4731 nitelikten 1301 tanesi bire bir korelasyon değeri yakalamıştır.

Tüm pozitif ve negatif korelasyon ilişkili veriler, yani 4731 nitelik ile karar destek vektör makinesi üzerinde işlem yaptığımızda, aşağıdaki sonuçları elde ediyoruz (Çizelge 9.7).

Çizelge 9.7 Kesme değeri 0,7 olan korelasyon ile indirgeme yapıldıktan sonra, eşik değeri alan nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :	**62,25**			
Veri Nitelik Sayısı :					**4731**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**2.45.11.88**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	96,13	**0,59**	1,55	0,51	108,59	**66,40**
OVO	90,19	**-0,03**	1,51	0,53	94,98	**67,03**
DAG	90,19	**-0,03**	1,53	0,55	94,95	**67,02**
WW	95,02	**0,00**	1,56	0,51	93,99	**66,60**
CS	93,71	**-1,31**	1,50	0,53	93,49	**66,83**

İlişkiye girmemiş 7803 veri niteliğini alarak, aynı testi uyguladığımızda ise başka bir sonuç çizelgesi ortaya çıkmaktadır (Çizelge 9.8).

Çizelge 9.8 Kesme değeri 0,7 olan korelasyon ile indirgeme yapıldıktan sonra, eşik değeri aşamayan nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :		**11_Tumors**	İndirgeme Oranı (%) :		**37,75**	
Veri Nitelik Sayısı :					**7803**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**2.45.11.88**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	96,10	**0,56**	1,55	51,10	186,23	**42,38**
OVO	90,75	**0,53**	1,51	52,65	165,70	**42,49**
DAG	90,75	**0,53**	1,53	54,65	164,67	**42,80**
WW	95,57	**0,55**	1,56	51,13	161,60	**42,57**
CS	95,57	**0,55**	1,50	53,48	161,89	**42,56**

Diğer bir yaklaşım ise, anlamlı korelasyon değerine sahip olmayan 7803 veriye dokunmadan, anlamlı 4731 veri niteliği üzerinden bir indirgeme yapmaya çalışmaktır. 1301 “bire bir” korelasyon kurmuş veri niteliğini, eksi korelasyon değerleri göz ardı edilerek, ikiye bölebilir ve yaklaşık 600 adet veri niteliğini indirgeyebiliriz. Geriye kalan 3430 veri niteliğinde ise böyle doğrusal bir ayrım yapmak olanaksızdır. Bunun için 122. ve 188. veri niteliklerini örnek verebiliriz. Korelasyon ilişki dosyalarına göre, 122. nitelik, 188. nitelik ile pozitif yüksek korelasyona girmiştir. 7. nitelik tarafından bakıldığında, 7. niteliğin 188. nitelik yerine geçebileceğini ve böylece 188. niteliğin orijinal veri kümesinden çıkarılabileceği varsayılabilir. Ancak, 188. nitelik aynı zamanda 21. nitelik ile pozitif yüksek korelasyona girmiştir. 122. nitelik üzerinden yorum yaparak 188. niteliği kaldırmak, 21. niteliğin boyutsal olarak anlam kaybetmesine neden olacaktır. Bunun engellenmesi için, veri niteliklerinden birbirleri ile bir küme oluşturanlar ayrılıp, tek bir nitelik haline getirilmelidir. Bu işlem için, bütün nitelikler bir ağaç yapısı haline getirilmeli ve her bir olasılık için tüm ağaç düğümleri ve yaprakları dolaşılmalıdır. Bu işlem başarıyla tamamlansa bile, teorik olarak, her birinin özünde tekil olduğu düşünülerek, en fazla 743 nitelik indirgenebilir. Bu oldukça düşük bir indirgeme oranıdır. Bu yöntem, korelasyonun hesaplandığı aynı sistemde beş saatten fazla çalışmış; ama somut bir sonuç verememiştir. Bu

yüzden, bu yöntem bu noktada durdurulmuş ve yöntemin işe yararlılığın düşük, maliyetinin yüksek olmasından dolayı yöntem iptal edilmiştir.

9.10.4 Kesme değeri 0,8 korelasyon

Kullanılan veri kümesi üzerine tam korelasyon uygulandığında, veri kümesinde bulunan 12534 nitelik üzerinden korelasyon matrisi hesaplanmış, bu hesaplamalar sonucu, korelasyon değeri 0.8 üzeri ve -0.8 altı olan değerler alınmıştır. Bu hesaplama işlemi atomik saat ile ölçüldüğünde 2 saat 45 dakika 12 saniye 11 salise sürmüştür. Bu süreç içerisinde, bilgisayarın işlemcisi toplam, yaklaşık ve ortalama %31 civarında kullanılmış, yine yaklaşık ve ortalama 1,86 GB hafıza kullanımı tespit edilmiştir.

Nitelik indirgeme işlemi maliyet açısından çok büyük bir değere ulaşmamasına rağmen, performans açısından yeterli bir küçültme yapmayı başarmıştır. 12534 veri parçası içinden, eşik değeri aşabilen sadece 2447 adet nitelik birbiri ile analiz edilebilir bir korelasyon değerine ulaşmıştır. 10087 nitelik ise analiz edilebilir herhangi bir korelasyon değerine ulaşamamıştır. 2447 niteliklendirilebilir veri niteliğinden, 961 tanesinin “bire bir” korelasyon değerine sahip olduğu ortaya çıkmıştır. 1486 veri niteliği ise birden fazla ilişkiyle korelasyona sahip olmuşlardır.

Bu yöntemi test etmek için, veri nitelikleri içerisinden korelasyona girmeyen tüm veri niteliklerinin sistemden çıkarılıp işlem yapılması gerekmektedir. Böylece, korelasyon oluşturan niteliklerin birbirleriyle ne şekilde değil; ancak ne derece bağımlı oldukları anlaşılacaktır. Ne şekilde bağlı olduklarını söyleyebilmek için henüz erkendir. Bu bir korelasyon izolasyon işlemidir. Bunun sonucunda aşağıdaki çizelge elde edilmiştir (Çizelge 9.9)

Çizelge 9.9 Kesme değeri 0,8 olan korelasyon ile indirgeme yapıldıktan sonra, eşik değeri aşan nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :		**11_Tumors**	İndirgeme Oranı (%) :		**80,48**	
Veri Nitelik Sayısı :					**2447**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**2.45.12.11**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	94,99	**-0,55**	1,48	54,80	56,09	**82,65**
OVO	91,36	**1,14**	1,49	57,11	47,45	**83,53**
DAG	91,36	**1,14**	1,47	55,68	47,33	**83,56**
WW	94,43	**-0,59**	1,55	54,76	47,51	**83,12**
CS	94,43	**-0,59**	1,50	52,66	47,33	**83,21**

Bu sonuca bir anlam yükleyebilmek için, bir test daha gerekmektedir. Bu da, ilk yöntemin tersine, veri niteliklerinin korelasyona girmiş olanlarını dışarıda bırakıp, girmeyenlerden yeni bir veri kümesi oluşturmaktır. Böylece, 10097 veri niteliğine sahip yeni veri kümesine aynı testler uygulanmıştır. Bunun sonucunda başka bir sonuç çizelgesi elde edilmiştir (Çizelge 9.10).

Çizelge 9.10 Kesme değeri 0,8 olan korelasyon ile indirgeme yapıldıktan sonra, eşik değeri aşamayan nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :	**19,52**			
Veri Nitelik Sayısı :					**10087**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**2.45.12.11**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	94,91	**-0,63**	1,48	54,80	247,86	**23,31**
OVO	90,06	**-0,16**	1,49	57,11	220,97	**23,30**
DAG	90,06	**-0,16**	1,47	55,68	221,20	**23,16**
WW	95,36	**0,34**	1,55	54,76	218,51	**22,35**
CS	94,77	**-0,25**	1,50	52,66	216,23	**23,28**

Bu noktada, diğer yaklaşım üzerinde durmak yine anlamsızdır. Tıpkı 0,7 kesmeli korelasyondaki gibi yüksek karmaşıklıklara çıkan bir ağaç yapısı söz konusudur ve bu ağaç yapısını işlemek, çalışmanın esas sebebi olan hız – başarı optimizasyonunu anlamsız bir hale getirecektir.
Korelasyon kesme eğilim değerini daha da yukarıya çekip, farklı indirgeme olasılıkları ortaya çıkarmak da bir başka yaklaşım olabilir.

9.10.5 Kesme değeri 0,9 korelasyon

Kullanılan veri kümesinin üzerine tam korelasyon uygulandığında, veri topluluğunda bulunan 12534 nitelik üzerinden korelasyon matrisi hesaplanmış, bu hesaplamalar sonucu, korelasyon değeri 0.9 üzeri ve -0.9 altı olan değerler alınmıştır. Bu hesaplama işlemi atomik saat ile ölçüldüğünde 2 saat 35 dakika 56 saniye 66 salise sürmüştür. Bu süreç içerisinde, bilgisayarın işlemcisi toplam, yaklaşık ve ortalama %29 civarında kullanılmış, yine yaklaşık ve ortalama 1,87 GB hafıza kullanımı tespit edilmiştir.

Nitelik indirgeme işlemi maliyet açısından çok büyük bir değere ulaşmamasına rağmen, performans açısından yeterli bir küçültme yapmayı başarmıştır. 12534

veri parçası içinden, eşik değeri aşabilen sadece 1196 adet nitelik birbiri ile analiz edilebilir bir korelasyon değerine ulaşmıştır. 11338 nitelik ise analiz edilebilir herhangi bir korelasyon değerine ulaşamamıştır. 1196 niteliklendirilebilir veri niteliğinden, 771 tanesinin “bire bir” korelasyon değerine sahip olduğu ortaya çıkmıştır. 425 veri niteliği ise birden fazla ilişkiyle korelasyona sahip olmuşlardır.

Tıpkı 0,8 Kesmeli Korelasyon gibi, 0,9 Kesmeli Korelasyon ile de yeni bir indirgeme yapıp, ortaya çıkan yeni veri kümesi üzerinde testler yapılmıştır. 0,8 Kesmeli Korelasyon indirgemesinden sonra, aldığımız başarılı sonuçlar, bizi korelasyon kesme değerini daha yukarıya, 0.9 değerine yükseltmemize neden olmuştur. Böylece, veri nitelik sayısı daha da azalacak ve süre iyice kısalacaktır. Bu testlerin sonucunda aşağıdaki çizelge ortaya çıkmıştır (Çizelge 9.11).

Çizelge 9.11 Kesme değeri 0,9 olan korelasyon ile indirgeme yapıldıktan sonra, eşik değeri aşan nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :		**11_Tumors**	İndirgeme Oranı (%) :		**90,46**	
Veri Nitelik Sayısı					**1196**	
İndirgeme Süresi (saat.dakika.saniye.salise)					**2.45.12.11**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	94,40	**-1,14**	1,44	54,50	25,58	**92,08**
OVO	93,00	**2,78**	1,50	57,63	23,41	**91,87**
DAG	93,00	**2,78**	1,46	54,11	22,89	**92,05**
WW	94,96	**-0,06**	1,53	54,75	23,58	**91,62**
CS	94,96	**-0,06**	1,51	54,10	23,55	**91,64**

Seçilen 1196 veri niteliğinin, veri niteliklerinin tamamını temsil ettiğini kanıtlamak amaçlı, geriye kalanlar üzerine de aynı testler uygulanmış ve şu aşağıdaki sonuçlara ulaşılmıştır (Çizelge 9.12).

Çizelge 9.12 Kesme değeri 0,9 olan korelasyon ile indirgeme yapıldıktan sonra, eşik değeri aşamayan nitelikler ile oluşturulmuş yeni veri kümesinin eğitilmesinin ve test edilmesinin istatistiksel sonuçları

Veri Seti :	**11_Tumors**	İndirgeme Oranı (%) :	**9,54**			
Veri Nitelik Sayısı :		**11338**			İndirgeme Süresi (s)	
İndirgeme Süresi (saat.dakika.saniye.salise)					**2.35.56.66**	
Test Algoritması	**Performans Oranı (%)**	**Performans Değişim Oranı (%)**	**Hafıza Kullanımı (GB)**	**İşlemci Kullanımı (%)**	**Analiz Süresi (S)**	**Süre Küçülme Oranı (%)**
OVR	94,91	**-0,63**	1,44	54,50	284,27	**12,04**
OVO	89,57	**-0,65**	1,50	57,63	255,45	**11,33**
DAG	89,57	**-0,65**	1,46	54,11	259,14	**9,98**
WW	94,88	**-0,14**	1,53	54,75	253,40	**9,95**
CS	94,35	**-0,67**	1,51	54,10	254,48	**9,71**

10. TARTIŞMA VE YORUM

10.1 GEMS ve Orjinal Veri Analizi (GOVA)

Sonuç çizelgesinden (Bkz. Çizelge 9.1) görülebileceği üzere, aynı tip tümör gen ifade veri kümesi kullanıldığında, en uzun süren algoritma OVR algoritması olmuştur; ancak, buna karşılık, en iyi performansı da yine OVR algoritması vermiştir. WW ve CS algoritmaları da OVR algoritmasına yakın bir performans elde etmiş; ancak ondan daha kısa sürede eğitim ve test işlemini tamamlamışlardır. Hafıza kullanımı en yüksek WW algoritmasında tespit edilmiş, en yüksek işlemci gücüne de DAG algoritması ulaşmıştır.

Bu çalışma, kabul edilebilir bir performans kaybı yaratarak, veri niteliklerini indirgeyip, süreyi kısaltmak olduğundan, WW ve CS algoritmaları bizim çalışmamıza daha uygun algoritmalardır; ancak yorum yapılabilmesi için tüm testler göz önünde bulundurulmalıdır.

10.2 Nitelik Ortalaması ve Standart Sapma (NOSS)

Nitelik ortalaması ile yapılan analizde, +2sd ve -2sd aralığında yer almayan nitelikler ile yeni bir veri kümesi oluşturulduğunda, performansı çok düşürmeden nitelijk sayısı 600 civarına indirilebilmektedir; ancak bu yöntem çok kullanışlı bir yöntem değildir. Standart sapma hesapladığı için, her veri seti için değişkenlik gösterebilir ve bu da her veri seti için bir kez daha hesaplama işlemi yapma maliyetini ortaya çıkarır (Bkz. Çizelge 9.2).

10.3 Bağımsız Bileşen Analizi (BBA)

Bağımsız Bileşen Analizi, çalışma mantığı gereği bir veri kümesindeki tüm sütunları birer sinyal olarak kabul edip, buna yönelik bir indirgeme işlemi yapmaktadır. Bu analiz, tüm veri kümesi içerisinde anlamlı ve ilişkili alt gruplar bulup, buna göre bir indirgeme yaptığından, ortamda bulunan sinyallerin sayısı arttıkça, analiz edebilme kapasitesi düşmekte ve bu çalışmadaki veri kümesindeki gibi çok yüksek değerlere çıktığında ise tamamen işlevini yitirip, çalışamaz bir hale gelmektedir.

Bağımsız Bileşen Analizi, genellikle birbirine karışmış, birbirinden en az iki boyut düzleminde farklı veri kümeleri içinden yeni veriler veya alt veri kümeleri çıkarmak

içindir. Ne kadar birbirinden bağımsız veri kümesi bir araya geldiyse, o kadar başarılı bir analiz yapılır. Bu yüzden, bu çalışmada kullanılan veri kümesi gibi, aralarında farklı grupsal ilişki bulunmasının beklenmediği ve önceki korelasyon analizler ile de bu fikirin daha da doğrulandığı veri kümelerinde, bu analiz yönteminin yüksek performans gösteremeyeceği rahatlıkla söylenebilir. Bu yüzden, bazı hafıza performans yöntemleri ile bu analiz yöntemini, bu çalışmanın veri kümesi üzerinde çalışması için zorlanması gerekli görülmedi, yöntem başarısız olarak kabul edildi.

10.4 Temel Bileşen Analizi (TBA)

Temel Bileşen Analizi, sonuç çizelgesinde de görülebileceği üzere çok büyük miktarda bir indirgeme yapmıştır. Bu indirgeme, doğrudan bazı veri niteliklerini çıkarmak değil, bu veri niteliklerinden yeni bir veri niteliği kümesi oluşturmaya dayalı bir indirgemedir. Bu yüzden, indirgenmiş veri kümesi incelenirse, orjinal veri kümesinden herhangi bir değeri doğrudan içermediği görülür.

TBA algoritması ile indirgenen veri kümesi oldukça küçülmüştür; ancak performans kayıpları %30 - %32 arasında olmuştur (Bkz. Çizelge 9.3). Bu kayıp oldukça yüksektir ve kabul edilebilir gözükmemektedir; ama tam bir yargıya varabilmek için, diğer indirgeme yöntemlerinin sonuçlarıyla karşılaştırılmalıdır.

10.5 İz Düşüm Takip Analizi (İDTA)

İz Düşüm Takip Algoritması, nitelik sayısını boyut olarak alıp, her bir boyut için bir veri niteliğini çıkarmayı ve bu şekilde tekrarlanarak nitelikleri indirgemeyi amaçlayan bir yöntemdir. Bu çalışmadaki veri kümesi gibi, içindeki bazı niteliklerin, kümenin tamamını temsil edemeyeceğinden şüphelenilen veri kümeleri için çok iyi sonuçlar verebilen bir algoritma olmasına karşın, çok yüksek boyutlu düzlemlerde, tekrarlama işlemi içerisinde tıkanma olasılığı oldukça yüksek bir yöntemdir. Bu çalışmanın kullandığı veri kümesindeki niteliklerin indirgenmesi sırasında benzer bir tıkanmaya maruz kalması da şüphesiz nitelik sayısının çok fazla olmasındandır.

10.6 Doğrusal Olmayan Temel Bileşen Analizi (DOTBA)

Doğrusal Olmayan Bileşen Analizi, türevi olduğu Doğrusal Bileşen Analizi'ne göre daha iy performans vermiştir ve neredeyse onun kadar bir indirgeme oranı yakalamıştır (Bkz Çizelge 9.4). Ancak, indirgeme süresi 10 saate kadar sürmektedir ki bu çok fazladır. Çalışmanın esas amacı olan süre optimizasyonunu tamamen anlamsız bırakmaktadır.

10.7 Kendinden Organize Haritalar (KOH)

Kendi düzenlenen haritalar, NLPCA'ya göre daha hızlı çalışan bir algoritma olmuştur; ancak performansı %40'lara düşürmesi kabul edilemez bir olgudur.

10.8 Doğrusal Diskriminant Analizi (DDA)

Doğrusal Diskriminant Analizi ile yapılan indirgeme, orjinal veri kümesindeki niteliklerin yaklaşık %50'sini elerken, performanstan %1 ile %5 arası bir kayıp olmuştur (Bkz Çizelge 9.5). İndirgeme ve performans açısından başarılı olsa da, doğrusal diskriminant analizinin işlem süresi diğer yöntemlere göre daha fazla olmuştur.

10.9 Çekirdek Temel Bileşen Analizi (ÇTBA)

Çekirdek Temel Bileşen Analizi, Temel Bileşen Analizinin bir uzantısıdır ve doğrusal olmayan bir haritalama ile nitelik indirgeme yapmaya dayalı bir algoritmadır. Veri nitelik sayısının çok fazla oluşu, bunun yanında buna uygulanacak doğrusal olmayan çekirdek metotları için çok fazla hafıza gereksinimi olmuş, maliyet aşırı yükseldiğinden, sistem sonuç veremeden sonlandırılmıştır.

Çekirdek Temel Bileşen Analizi, özellikle çok sınıflı veri kümelerinde iyi sonuçlar vermesine karşın, biyoinformatik alanında kullanılan ve genlerin ifadelerini içeren veri kümelerindeki aşırı yüksek veri nitelik sayıları yüzünden, bu ve Temel Bileşen Analizi algoritmasının buna benzer türev uzantı algoritmaları, bu çalışmanın kullandığı veri kümeleri gibi veri kümelerinde çalışmamaktadır.

10.10 Korelasyon Analizi (KA)

Korelasyon analizi ile veri niteliği indirgeme işlemleri başarılı olmuştur. Bu tür bir indirgeme, diğer yöntemlerin aksine doğrusal indirgeme yöntemidir. Buradaki doğrusal terimi anlamsal değil yapısal olarak alınmalıdır. Diğer algoritmalar, veri kümesindeki nitelikleri kullanarak yeni, orjinal veri kümesinde değer olarak karşılığı bulunmayan niteliklerden yeni bir veri kümesi oluşturur. Korelasyon ise, hangi veri niteliklerinin çıkarılacağını belirterek bir indirgeme yaptığı için, oluşturulan yeni veri kümesindeki değerler, mutlaka orjinal veri kümesinde de mevcuttur.

Hem güvenilir, hem de güvenilir olmayan veri nitelikleri ile ayrı ayrı test yapılmasının nedeni, temsil etme gücünün ortaya çıkarılmasıdır. Mutlaka, öyle ya da böyle her veri niteliği bir şekilde veri kümesini temsil ediyordur; ancak her nitelik aynı güçte temsil etmez. Bir niteliğin temsil ettiği güç ölçüsüne ulaşmak için, belki binlerce başka nitelik birleşmek zorunda kalmaktadır. Bu bağlamda, binlerce niteliği almak yerine, tek bir niteliği almak, şüphesiz oldukça yüksek bir süre kazancını beraberinde getirecektir.

10.10.1 Kesme değeri 0,7 olan korelasyon

0,7 Kesmeli korelasyon ile indirgeme, ana veri kümesindeki veri nitelik sayımızı yaklaşık üçte birine düşürmüş ve performans çok ufak kayıplara uğramıştır. Bu, çalışmanın amacına uygun olmakla beraber başarılı bir sonuçtur; ancak aynı zamanda korelasyon kesme değerinin daha yukarı çekilmesi durumunda daha başarılı sonuçlara ulaşılabileceğinin de bir kanıtı olarak kabul edilebilir.

10.10.2 Kesme değeri 0,8 olan korelasyon

İlk sonuç çizelgesinde dd görülebileceği üzere (Bkz Çizelge 9.9), 0,8 kesmeli korelasyon ile indirgeme işlemi yapıldığında, veri nitelik sayısı yaklaşık %80 oranında azalmış, buna karşın performans oranı +- %1 oranında bir değişim

göstermiştir. Bu kabul edilebilir bir değişimdir. Veri nitelik sayısının böyle büyük bir oranda değişim göstermesi, işlem süresini de oldukça düşürmüştür. Sonuç çizelgesine göre, %82 - %83 arasında bir oran ile süre kısalmıştır. Bu oldukça yüksek bir oran olup, bu oran, çalışmanın amacına uygundur. 0,8 korelasyon ile nitelik indirgeme mümkündür ve oldukça başarılı olmuştur.

İkinci sonuç çizelgesinin (Bkz Çizelge 9.10), ilk testin sonuç çizelgesine (Bkz Çizelge 9.9) performans açısından benzemesi, bizim iki önemli yargıya varmamıza neden olmaktadır. Bunlardan ilki, 0,8 kesimli korelasyon sonucu korelasyon ilişkisi bulunmayan veri niteliklerinin tamamının temsil ettiği düzlemsel değer, korelasyon ilişkisi bulunan veri niteliklerinin temsil ettiği ile aynı kabul edilebilecek kadar yakındır. Bu da gösterir ki, iki aynı veri niteliği grubundan bir tanesini kullanmamız yeterlidir ve hangisinin kullanılacağının bir önemi yoktur. İkinci yargı ise, bu iki veri niteliğinden hangisi seçilirse seçilsin ve buna göre yeni bir veri kümesi oluşturulsun, bu, tüm veri kümesini temsil eden bir veri kümesi olacaktır. Benzer bir ikili test sonuç ilişkisi aynı zamanda 0,7 kesmeli ve 0,9 kesmeli korelasyonlarda da görülmektedir.

10.10.3 Kesme değeri 0,9 olan korelasyon

Sonuç çizelgesinde de görüldüğü üzere (Bkz Çizelge 9.11), 0,9 Kesmeli Korelasyon indirgeme yüzdesini 90 değerlerine çıkarmış, performans değişimi ise %1 - %2 aralığında olmuştur. Süre ise yaklaşık %92 oranında azalmıştır. İkinci sonuç çizelge de (Bkz Çizelge 9.12), ilk testin geçerliliğini kanıtlamıştır.

11. SONUÇ VE ÖNERİLER

Çalışma sonucunda açık bir şekilde görülmüştür ki, gen ifade verileri kullanılarak yapılan DVM sınıflandırmaları yüksek performanslı sonuçlar vermektedir. Öte

yandan, gen ifade verilerinin nitelik sayılarının on binlerce olması, bu sınıflandırma işleminin süre maliyetini oldukça arttırmaktadır.

Çalışmanın başında da belirtildiği gibi, kanser hastalığındaki teşhis hızının kritik olması, bu sınıflandırma işlemlerindeki hızın, en az performans kadar önemli olduğunu ortaya koymuştur. Sistematik bir indirgeme yöntemi, DVM'nin sınıflandırma hızını arttıracak, bu da daha hızlı teşhis imkanı sağlayacaktır.

Bağımsız Bileşen Analizi, Çekirdek Temel Bileşen Analizi, İz Düşüm Takip Analizi algoritmaları, çalışmanın veri kümesi üzerinde sonuç verememişlerdir. Bundaki en büyük etken, nitelik sayısının çok fazla ve örneklem sayısının buna oranla oldukça düşük olmasıdır.

Diğer algoritmalar ise perfomans, maliyet, süre ve kullanılabilirlik olarak farklı sonuçlar vermişlerdir. Standart Sapma ile veri kümesindeki fark yaratan niteliklerden yeni bir veri kümesi oluşturup analiz edildiğinde, nitelik sayısının oldukça düştüğünü, buna bağlı olarak da DVM eğitim ve test süreçlerinin %90'lara varan ölçülerde hızlandığı görülmüştür. Buna karşılık, bu yöntemin kullanılabilirliği, diğer yöntemlere göre daha düşüktür.

Diğer bir yöntem olan Temel Bileşen Analizi, yöntemler arasındaki en fazla indirgemeyi gerçekleştiren algoritma olmuştur. Nitelik sayısını %1'lere indirmiştir ve bu da sürede %95'lere varan bir kazanç sağlamıştır. Ancak, performans kaybı kabul edilebilir düzeyden çok daha aşağılara inmiş ve %60'lara düşmüştür. Bu neredeyse üçte birlik bir performans kaybına karşılık gelmektedir. Kanser teşhisi gibi kritik bir konuda, hız kazancı için bu kadar büyük bir performans kaybı kesinlikle kabul edilememektedir.

Doğrusal Diskriminant Analizi, diğer yöntemlere göre yapısal olarak farklılık göstermektedir. Nitelik indirgemeyi sınıflandırma yaparak gerçekleştirir, ki bu bir tür ön sınıflandırma olarak görülebilir. Öte yandan, sonuçlar göstermektedir ki, doğrusal diskriminant analizi, bu çalışmanın veri kümesi için çok başarılı sonuçlar verememiştir. Niteliklerin sadece yarısını kümeden çıkarabilirken, bu işlem için üç saat harcamıştır ki bu oldukça fazladır. Öte yandan göz önünde bulundurulmalıdır ki, bu bir nitelik indirgeme yöntemidir ve aynı tür bir veri kümesi için sadece bir kere yapılması yeterli olacaktır.

Bir diğer yöntem olan Doğrusal Olmayan Temel Bileşen Analizi, kullandığı daha gelişmiş matematiksel ifadeler sayesinde, türevi olduğu Temel Bileşen Analizi'ne göre çok daha iyi bir performans vermiştir. Süre kazancı onun kadar olmasa da,

ortalamanın çok üstündedir. Ancak, doğrusal olmayan temel bileşen analizi, adından da anlaşılabileceği üzere doğrusal olmayan bir indirgeme yaptığı için, çok uzun sürmektedir. Bu çalışmadaki veri kümesini indirgemesi yaklaşık 10 saat sürmüştür ki bu doğrusal olmayan temel bileşen analizinin tüm artılarını gölgede bırakacak kadar ciddi bir dezavantajdır; çünkü bu bir nitelik çıkarım yöntemidir ve aynı türdeki tüm veri kümeleri için, eğitim, test ya da kullanım fark etmeden her yeni veri kümesi için tekrarlanmak zorundadır.

Kendi Düzenlenen Haritalar yöntemi, çok fazla bir indirgeme yapamamış, süre kazancı sağlayamamış ve büyük performans düşüşlerine neden olmuştur. Bu yüzden, kabul edilebilir bir yöntem olmaktan çok uzaktır.

Son olarak, Korelasyon Analizi ile yapılan indirgeme oldukça başarılı olmuştur. Üç farklı kesme değeri için de diğer algoritmalara göre çok daha iyi sonuçlar vermiştir. En iyi sonuç 0,9 kesmeli tam korelasyon sonucu ortaya çıkmıştır. Nitelik sayısını onda birine düşürmüş, buna oranla yüksek bir hız kazancı sağlamıştır. Bu oran Doğrusal ve Doğrusal Olmayan Temel Bileşen Analizi'ne göre daha düşüktür; ancak korelasyonun birçok avantajı vardır. Öncelikle, korelasyon analizi ile yapılan indirgeme bir izolasyon işlemidir. Yani, veri kümesinin içeriğine yönelik değil, yapısına yönelik bir indirgeme yapar. Bu da indirgeme işleminin bir veri kümesi için sadece bir kere yapılmasını sağlar. Öte yandan, Temel Bileşen Analizi bir çıkarım indirgemesi yapmaktadır. Bu, veri kümesinin indirgenmesinden sonra tamamen değişmesi demektir ki bunun sonucunda, veri kümesi bir kez değil, kullanılacağı eğitim, test ya da yeni bir veriyle sınıflandırma dahil her şekilde indirgeme işleminin tekrarlanması gerekir. Doğrusal olmayan bileşen analizi bu noktada işlevini tamamen yitirirken, her ne kadar doğrusal temel bileşen analizi korelasyondan çok daha hızlı çalışsa da, her seferinde bu analizi yapmak uzun vadede büyük zaman kayıplarına yol açacaktır.

Çalışmanın sonucunda görülmektedir ki, indirgeme işlemi çevresel faktörelere göre farklılık gösterebilir ve farklı durumlarda farklı algoritmalar kullanılabilir; ancak genel anlamda, bir veri kümesindeki nitelik sayısını düşürmek için kullanılacak en performanslı yöntem korelasyodur. Korelasyon kullanılarak yapılan indirgeme işleminin her veri kümesi için sadece bir kere yapılması, hızlı ve fazla maliyetli olmaması ve belki de en önemlisi, değişik eşik değerleri ile farklı süre – performans değerlerine sahip olabilmesi, korelasyonu diğer yöntemlerin bir adım önüne taşımaktadır.

Çalışma bu yöntemlerle ve analizlerle sınırlı kalmayabilir. Korelasyon başta olmak üzere, diğer yöntemlerin bazılarının uçları açıktır ve geliştirilmeye devam edilebilir. Korelasyon bir adım daha ilerletilip, tam korelasyon yerine sınıfsal korelasyon yapılabilir. Daha gelişmiş veri madenciliği teknikleri kullanılabilir ya da mevcut olanlar geliştirilebilir. Bu çalışmada tıkanan, kilitlenen ya da sonuş vermeyen yöntemler optimize edilip, tekrardan denenebilir.

12. KAYNAKLAR

[1] Alexander Statnikov, Constantin F. Aliferis, Ioannis Tsamardinos, A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis, Bioinformatics, 21(5): 631-43, 2005.

[2] Prof. Dr. M.Tezer Kutluk, 2006, Tıp Dünyası Kanserle Mücadelede Yalnız Değil, Türk Kanser Araştırma ve Savaş Kurumu Derneği, http://www.turkcancer.org/news.php?id=74.

[3] M.H. Fulekar, Bioinformatics: Applications in Life and Environmental Sciences. sf:77-100, 2009.

[4] Cho S.B. and Won H.H., Machine learning in DNA microarray analysis for cancer classification, Conferences in Research and Practice in Information Technology, 19s., 2003.

[5] Chee M., Accessing genetic information with high-density DNA arrays, Science, 274(5287):610-4, 1996.

[6] Szallasi Z., Gene expression patterns and cancer, Nature Biotechnology, 16: 1292 - 1293, 1998.

[7] Bowtell D., Options available - from start to finish - for obtaining expression data by microarray, 1999.

[8] Bassett D. Jr, Eisen M.B. and Boguski M.S., Gene Expression Informatics, 1999.

[9] White K.P., Rifkin S.A., Hurban P. and Hogness D.S., Microarray analysis of Drosophila development during metamorphosis. Science, 286(5447): 2179-2184, 1999.

[10] Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J. P., Coller H., Loh M. L., Downing J. R., Caligiuri M. A., Bloomfield C. D. and Lander E. S., Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science, 286: 531-537., 1999.

[11] Slonim D., Tamayo P., Mesirov J., Golub T.R. and Lander E., Class prediction and discovery using gene expression data, Proceedings of RECOMB , 2000.

[12] Ramaswamy S., Tamayo P., Rifkin R., Mukherjee S., Yeang C.H., Angelo M., Ladd C., Reich M., Latulippe E., Mesirov J.P., Poggio T., Gerald W., Loda M., Lander E.S. and Golub T.R., Multiclass cancer diagnosis using tumor gene expression signatures, 2001.

[13] Li L., Weinberg C.R., Darden T.A. and Pederson L.G., Gene selection for sample classification based on gene expression data: Study of sensitivity to choiceof parameters of the GA/KNN method, Bioinformatics, 17(12), 1131-1142, 2001.

[14] Brazma A., Hingamp P., Quackenbush J., Sherlock G., Spellman P.T., Stoeckert C., Aach J., Ansorge W., Ball C.A., Causton H.C., Gaasterland T., Glenisson P., Holstege F.C.P., Kim I.F., Markowitz V., Matese J.C., Parkinson H., Robinson A., Sarkans U., Schulze-Kremer S., Stewart J., Taylor R., Vilo J. and Vingron M., Minimum information about a microarray experiment (MIAME) - toward standards for microarray data, Nature Genetics, 29: 365-371, 2001.

[15] Spellman P.T., Miller M., Stewart J., Troup C., Sarkans U., Chervitz S., Bernhart D., Sherlock G., Ball C.A., Lepage M., Swiatek M., Marks W.L., Goncalves J., Market S., Lordan D., Shojatalab M., Pizarro A., White J., Hubley R., Deutsch E., Senger M., Aronow B.J., Robinson A., Bassett D., Stoeckert J. Jr. and Brazma A., Design and implementation of microarray gene expression markup language (MAGE-ML), Genome Biology, 3(9), 2002.

[16] Herroro J., Valencin A. and Dopazo J., A hierarchical unsupervised growing neural network for clustering gene expression patterns, Bioinformatics, 17: 126-136., 2001.

[17] Dopazo J., Microarray data processing and analysis In*:* Microarray Data Analysis II, Kluwer Academic Publications, 43-63, 2002.

[18] Hedenfalk I., Duggan D., Chen Y., Radmacher M., Bittner M., Simon R., Gene-expression profiles in hereditary breast cancer, New England Journal of Medicine, 344: 539-548, 2001.

[19] Khan J., Wei J.S., Ringner M., Saal L.H., Ladanyi M., Westermann F., Berthold F., Schwab, M., Antonescu C.R., Peterson C. and Meltzer P.S., Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, Nature Medicine, 7(6): 673-679, 2001.

[20] Hwang K.B., Cho D.Y., Park S.W., Kim S.D. and Zhang B.T., Applying machine learning techniques to analysis of gene expression data: Cancer Diagnosis. In: Methods of Microarray Data Analysis, Kluwer Academic, 167-182, 2001.

[21] Chen D., Chang R.F. and Huang Y.L., Breast cancer diagnosis using selforganizing map for sonography, Ultrasound Medical Biology, 26(3): 405-411, 2000.

[22] Alizadeh A.A., Eisen M.B., Davis R.E., Ma C., Lossos I.S., Rosenwald A., Boldrick J.C., Sabet H., Tran T., Yu X., Powell J.I., Yang L., Marti G.E., Moore T., Hudson J. Jr., Lu L., Lewis D.B., Tibshirani R., Sherlock G., Chan W.C., Greiner T.C., Weisenburger D.D., Armitage J.O., Warnke R., Levy R., Wilson W., Grever M.R., Byrd J.C., Botstein D., Brown P.O. and Staudt L.M., Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, Nature, 403(3): 503-511, 2000.

[23] Yeang C.H., Ramaswamy S., Tamayo P., Mukherjee S., Rifkin R.M., Angelo M., Reich M., Lander E., Mesirov J. and Golub T., Molecular classification of multiple tumor types. Bioinformatics, 17: 316S-322S, 2001.

[24] Westhead D.R., Parish J.H. and Twyman R.M. (eds)., Instant Notes on Bioinformatics, BIOS Scientific Publishers Ltd., Oxford, UK, 2003.

[25] Talavera L., Dependency-Based Feature Selection for Clustering Symbolic Data, Intelligent Data Analysis, 4: 19-28, 2000.

[26] Vipin Kumar, Data Mining Algorithms, Tutorial at IPAM 2002, Presentation slides, 2002.

[27] Constantin Aliferis M.D., Ph.D. Department of Biomedical Informatics, Office of Technology Transferand Enterprise Development, Gene Expression Model Selector (GEMS), 2009.

[28] Vapnik V. N., Statistical learning theory, Wiley, 1998.

[29] Weston J. and C. Watkins., Multi-class support vector machines, In M. Verlysen (Ed.), Proceedings ESANN'99, Brussels, D-Facto Publications, 1999.

[30] Dietterich T. G. and G. Bakiri, Solving multiclass learning problems via error-correcting output codes, Journal of Artificial Intelligence, Research 2, 263–286, 1995.

[31] Allwein E. L., Schapire R. E. and Y. Singer, Reducing multiclass to binary: a unifying approach for margin classifiers, Journal of Machine Learning Research 1, 113–141, 2000.

[32] Schölkopf B. J., Platt J., Shawe - Taylor A., Smola and Williamson R. C., Estimating the support of a high-dimensional distribution, Neural Computation 13(7), 1433–1471, 2001.

[33] Tax D. and Duin R., Data domain description by support vectors, In M. Verleysen (Ed.), Proceedings European Symposium on Artificial Neural Networks, ESANN, D. Facto Press, pp. 251–256., 1999.

[34] Friedman J, Another approach to polychotomous classification, Technical report, Stanford Univeristy, 1996.

[35] Kressel U., Pairwise classification and support vector machines, In Advances in Kernel Methods: Support Vector Learning, Chapter 15, MIT Press, 1999.

[36] Platt J., Cristianini N. and Shawe-Taylor J., Large margin dags for multiclass classification, Advances in Neural Information Processing Systems 12, 547-553 s., MIT Press, 2000.

[37] Hsu, Chih-Wei and Chih-Jen Lin, A Comparison of Methods for Multi-class Support Vector Machines, IEEE Transactions in Neural Networks, 13(2) 415-425, 2002.

[38] Weston J. and Watkins C., Support Vector Machines for Multi-Class Pattern Recognition, Proceedings of the Seventh European Symposium On Artificial Neural Networks, 1999.

[39] Crammer K. and Singer Y., On the Learnability and Design of Output Codes for Multiclass Problems, Proceedings of the Thirteen Annual Conference on Computational Learning Theory (COLT), 2000.

[40] Fisher R. A. and Mackenzie W. A, The manurial response of different potato varieties, Journal of Agricultural Science, xiii. 311-320, 1923.

[41] Pearson K., On lines and planes of closest fit to systems of points in space, Philosophical Magazine, 2:559-572, 1901.

[42] Hotelling H., Analysis of a complex of statistical variables into principal components, Journal of Educational Psychology, 24:417-441, 498-520, 1933.

[43] Bryant and Atchley, Integr. Comp. Biol., SUMMER, 15: 833 – 835, 1975.

[44] Rao, Canunical variate analysis and related methods for reduction of dimensioanilty and graphical representation, 1964.

[45] Frisch, Ragnar, Correlation and scatter in statistical variables, Nordic Statistical Journal, 1*:* 36–102, 1929.

[46] Thurstone L.L., Measurement of social attitudes, Journal of Abnormal and Social Psychology, (26), 249-269, 1931.

[47] Hotelling, H., Relations between two sets of variants, Biometrika, 28, 321-377, 1936.

[48] Girshick, M.A., Principal components, Journal of the American Statistical Association, 31, 519-528, 1936.

[49] Girshick M.A., On the sampling theory of the roots of determinantal equations, Ann. Math. Stat., Volume 10, pp. 203-224, 1939.

[50] Anderson, Loretta and Mary Ruth Wise, Contrastive features of Candoshi clause types., In Studies in Peruvian Indian languages, 1, 67-102, Summer Institute of Linguistics Publications in Linguistics and Related Fields, 9. Norman: Summer Institute of Linguistics of the University of Oklahoma, 1963.

[51] Gower J. C., Some Distance Properties of Latent Root and Vector Methods used in Multivariate Analysis, Biometrika 53, 325-338, 1966.

[52] Jeffers J.N.R., Two case studies in the application of principal components analysis, Appl. Statist., 16, 225–236, 1967.

[53] Mobley C. D., A numerical model for the computation of radiance distributions in natural waters with wind-roughened surfaces, part 2: Users' guide and code listing, NOAA Tech. Memo, ERL PMEL-8 1 (NTIS PB88-24687 1), 1988.

[54] Jolliffe I.T. Jolliffe, Extracting Principal Component Analysis, Second Edition, sf:1-6, 2002.

[55] Cazes P, Baumerdier A., Bonnefous S. and Pagès J.P., Codage et Analyse des Tableaux Logiques. Cahiers du B.U.R.O. 27, 1977.

[56] Lebart L., Morineau A. and Tabard N., Techniques de la Description Statistique, Dunod, 1977.

[57] Hill M.O., Correspondence Analysis: a Neglected Multivariate Method, Applied Statistics, 23:340–354, 1974.

[58] Nishisato S., Analysis of Categorical Data: Dual Scaling and its Applications. University of Toronto Press, Toronto, Canada, 1980.

[59] Kruskal J.B. and Shepard R.N., A Nonmetric Variety of Linear Factor Analysis, Psychometrika, 39:123–157, 1974.

[60] Young F.W., Takane Y. and De Leeuw J., The Principal Components of Mixed Measurement Level Multivariate Data: an Alternating Least Squares Method with Optimal Scaling Features, Psychometrika, 45:279–281, 1978.

[61] De Leeuw J. and Van Rijkevorsel J., HOMALS and PRINCALS, Data Analysis and Informatics, North Holland Publishing Company, Amsterdam, 1980.

[62] De Leeuw J., Van Rijckevorsel J. and Van Der Wouden H., Nonlinear Principal Component Analysis Using B-Splines, Methods of Operations Research, 23:211–234, 1981.

[63] Gifi A., Nonlinear Multivariate Analysis, Technical report, Department of Data Theory, University of Leiden, 1981.

[64] Tenenhaus M., Multiple Correspondence Analysis and Duality Schema: a Synthesis of Different Approaches, Metron, 40:289–302, 1982.

[65] McDuffee C.C., The Theory of Matrices, Chelsea, New York, 1946.

[66] Jan de Leeuw, Nonlinear Principal Component Analysis, Department of Statistics, University of California, Los Angeles, 2005.

[67] Aizerman M. A., Braverman E. M., Rozonoer L. I., Theoretical foundations of the potential function method in pattern recognition learning, Automation and Remote Control, 25:821-837, 1964.

[68] Schölkopf B., Burges C., Vapnik V., Extracting support data for a given task, First International Conference on Knowledge Discovery and Data Mining, Menlo Park, CA. AAAI Press, 1995.

[69] Bernhard Schölkopf, Alexander Smola, Klaus-Robert Müller, Kernel Principal Component Analysis, Max-Planck-Institut f. Biol. Kybernetik, Spemannstr. 38, 72076 Tübingen, Germany, 1999.

[70] Luo Zhong, Huazhu Song, Bo Han, Extracting Structural Damage Features and Comparison Between PCA and ICA, School of Computer Science and Technology, Wuhan University of Technology, Wuhan, Hubei 430070, China, 2006.

[71] Friedman J. H., Exploratory Projection Pursuit, 82, 249, 1987.

[72] Diaconis P. and Freedman D., Partial Exchangeability and Sufficiency in Statistics, Applications and New Directions, pp. 205-236, Indian Statistical Institute, Calcutta, 1984.

[73] Brian S, Blais, The Role of the Environment in Synaptic Plasticity:Towards an Understanding of Learning and Memory, Department of Physics at Brown University, chapter3 sf57, 1998.

[74] Duda R. O. and Hart P. E., Pattern Classification and Scene Analysis, Wiley, 1973.

[75] Fukunaga K., Introduction to Statistical Pattern Recognition (Second ed.), Academic Press, 1990.

[76] Bishop Christopher M., Pattern Recognition and Machine Learning, sf:186-189, 2006.

[77] Tamayo P., Slonim D., Mesirov J., Zhu Q., Kitareewan S., Dmitrovsky E., Lander E. and Golub T., Interpreting patterns of gene expression with self-organizing maps, *PNAS*, 96:2907-2912, 1999.

[78] Ben-Dor A., Shamir R. and Yakhini Z., Clustering gene expression patterns, Journal of Computational Biology, 6(3/4): 281-297, 1999.

[79] Carr D.B., Somogyi R. and Micheals G., Templates for looking at gene expression clustering, Stat. Comput. and Stat. Graph. Newsletter, 20-29, 1997.

[80] Wen X., Fuhrman S., Michaels G.S., Carr D.B., Smith S., Barker J.L. and Somogyi R., Large-scale temporal gene expression mapping of central nervous system development, Proc. Natl. Acad. of Sc. USA, 95(1): 334-339, 1998.

[81] Kim H., Microarray analysis II: Whole-genome expression analysis, CISC889: Bioinformatics course, Presentation slides, www.innu.org/~super/dnac/microarray.ppt, 2002.

[82] Cohen J., Cohen P., West S.G., Aiken L.S., Applied multiple regression / correlation analysis for the behavioral sciences, (3rd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates. sf:1-7, sf:23-32, 2003.