

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**KARIŞMIŞ İŞARETLERİN KANONİK KORELASYON ALGORİTMALARI
İLE AYRIŞTIRILMASI**

YÜKSEK LİSANS TEZİ

Hüsamettin ÇELİK

**HAZİRAN 2015
TRABZON**

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**KARIŞMIŞ İŞARETLERİN KANONİK KORELASYON ALGORİTMALARI
İLE AYRIŞTIRILMASI**

Hüsamettin ÇELİK

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"YÜKSEK LİSANS (ELEKTRONİK)"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

**Tezin Enstitüye Verildiği Tarih : 25.05.2015
Tezin Savunma Tarihi : 19.06.2015**

Tez Danışmanı : Yrd. Doç. Dr. Yusuf SEVİM

Trabzon 2015

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalında
Hüsamettin ÇELİK tarafından hazırlanan

KARIŞMIŞ İŞARETLERİN KANONİK KORELASYON ALGORİTMALARI
İLE AYRIŞTIRILMASI

başlıklı bu çalışma, Enstitü Yönetim Kurulunun 02 / 06 / 2015 gün ve 1605 sayılı
kararıyla oluşturulan jüri tarafından yapılan sınavda

YÜKSEK LİSANS TEZİ

olarak kabul edilmiştir.

Jüri Üyeleri

Başkan : Doç. Dr. Ayten ATASOY

Üye : Yrd. Doç. Dr. Yusuf SEVİM

Üye : Yrd. Doç. Dr. Yasin OĞUZ

Prof. Dr. Sadettin KORKMAZ

Enstitü Müdürü

ÖNSÖZ

Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim Dalı, Yüksek Lisans Programı çerçevesinde gerçekleştirilmiş olan bu çalışma “Karışmış İşaretlerin Kanonik Korelasyon Algoritmaları ile Ayrıştırılması” işlemini ele almaktadır.

Öncelikle bu çalışmam esnasında bilgi ve tecrübelerini esirgemeyen danışman hocam Sayın Yrd. Doç. Dr. Yusuf SEVİM’e ve beni bugüne kadar yalnız bırakmayan, desteğini hiçbir zaman esirgemeyen, değerli ailem ve eşim Emine ÇELİK’e bu vesileyle sevgi ve saygılarımı sunarım.

Hüsamettin ÇELİK
Trabzon 2015

TEZ ETİK BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduđum “Karışmıř İşaretlerin Kanonik Korelasyon Algoritmaları ile Ayrıştırılması” başlıklı bu çalışmayı baştan sona kadar danışmanım Yrd. Doç. Dr. Yusuf SEVİM’in sorumluluğunda tamamladıđımı, verileri/örnekleri kendim topladıđımı, deneyleri/analizleri ilgili laboratuarlarda yaptıđımı, başka kaynaklardan aldıđım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiđimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandıđımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiđimi beyan ederim. 25/05/2015

Hüsamettin ÇELİK

İÇİNDEKİLER

Sayfa No

ÖNSÖZ	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VII
SUMMARY	VIII
ŞEKİLLER DİZİNİ.....	IX
TABLolar DİZİNİ.....	XI
1. GENEL BİLGİLER.....	1
1.1. Giriş.....	1
1.2. K�r Kaynak Ayrıştırma	2
1.3. Kokteyl Parti Problemi.....	3
1.4. Karışım Modelleri	4
1.4.1. Anlık Karışım Modeli	6
1.4.2. Yansız Karışım Modeli.....	6
1.4.3. Yansızlı Karışım Modeli	7
1.5. Temel Bileşen Analizi	8
1.6. Bağımsız Bileşen Analizi	10
1.6.1. BBA Algoritmasında Kaynak ve Sens�r Sayı Bağılantısı	12
1.7. Kanonik Korelasyon Analizi.....	12
1.7.1. Kanonik Değişkenler ile Orjinal Değişkenler Arasındaki Korelasyonlar	22
1.8. Çekirdek Öznitelik Uzayı.....	25
1.8.1. Öznitelik Uzayında Öğrenme.....	25
1.8.2. Öznitelik Uzayına Dolaylı Eşleme.....	28
1.8.3. Çekirdeklerin Oluşturulması	32
1.8.4. Çekirdek Hilesi.....	33
1.9. Dalgacık Dönüşümü	35
1.9.1. Sürekli Dalgacık Dönüşümü	37
1.9.2. Ayrık Dalgacık Dönüşümü (ADD)	38

1.9.3.	Ters Dalgacık Dönüşümü.....	40
1.9.4.	Ters Dönüşüm Filtreleme.....	40
1.10.	Doğrusal ve Doğrusal Olmayan Karıştırma Yöntemleri.....	41
2.	UYGULAMALAR VE SONUÇLARI.....	43
2.1.	Doğrusal Karışım Sonuçları.....	44
2.2.	Doğrusal Olmayan Karışım Sonuçları.....	51
3.	SONUÇLAR.....	59
4.	ÖNERİLER.....	60
5.	KAYNAKLAR.....	61
6.	EKLER.....	66
	ÖZGEÇMİŞ	

Yüksek Lisans Tezi

ÖZET

KARIŞMIŞ İŞARETLERİN KANONİK KORELASYON ALGORİTMALARI
İLE AYRIŞTIRILMASI

Hüsamettin ÇELİK

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Danışman: Yrd. Doç. Dr. Yusuf SEVİM
2015,65 Sayfa, 1 Sayfa Ek

Kör kaynak ayrıştırma, en az iki sinyalin karışımını içeren bir veri kümesinden bu karışımı oluşturan her bir kaynağın tahmin edilmesi olarak tanımlanabilir. Bu işlemin kör olarak adlandırılması kaynaklar hakkında hiçbir bilgiye sahip olmadığımızı ifade etmektedir. Kör kaynak ayrıştırma yöntemlerinin başarımı, işlem süresi ve doğruluğu ile belirlenmektedir. Çekirdek kanonik korelasyon analizinin işlem performansı literatürdeki diğer yöntemlere göre iyi olmakla birlikte işlem süresi oldukça fazladır. Yapılan bu çalışmada çekirdek kanonik korelasyon analizinin işlem süresinin azalımı amaçlanmıştır. Bu amaç doğrultusunda karışım işaretleri çekirdek kanonik korelasyon analizine uygulanmadan önce ayrık dalgacık dönüşümü işlemine tabi tutulmuştur. Yapılan bu uygulama işlem süresini azaltmakla birlikte performansında da iyileşmelerin olduğu gözlemlenmiştir. Aynı zamanda, yapılan bu işlemler doğrusal ve doğrusal olmayan karıştırma modelleri ile yapay olarak karıştırılıp değerlendirilmiştir. Önerilen yöntemin başarımlarını analizini görmek için sonuçlar işaret gürültü oranı cinsinden değerlendirilmiştir.

Anahtar Kelimeler: Kör Kaynak Ayrıştırma, Çekirdek Kanonik Korelasyon Analizi, Ayrık Dalgacık Dönüşümü

Master Thesis

SUMMARY

SEPARATING MIXED SIGNALS USING CANONICAL CORRELATION ALGORITHMS

Hüsamettin ÇELİK

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Electric-Electronic Engineering Program
Supervisor: Assist. Prof. Dr. Yusuf SEVİM
2015, 65 Pages, 1 Pages Appendix

Blind source separation can be defined as the estimation of each sources which are composed of a data set including at least two signals. As there is any information about sources, this operation is denominated as blind. The success of blind source separation is determined by the operation time and the reliability. Although the operation performance of kernel canonical correlation analysis is better than other methods in the literature, the operation time is relatively longer. This study aims to decrease the operation time of kernel canonical correlation analysis. For this purpose, mixing signals are subject to Discrete Wavelet Transform before kernel canonical correlation analysis is performed. The corresponding application enables to decrease operation time and to improve the operation performance. At the same time, all these operations are evaluated using linear and nonlinear mixing models through artificial mixing. The analysis results are evaluated in terms of signal noise ratio to examine the success analysis of the proposed method.

Key Words: Blind Source Separation, Kernel Canonical Correlation Analysis, Discrete Wavelet Transform

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1. Kokteyl parti problemi	4
Şekil 2. Yansımasız karışım	6
Şekil 3. Yansımali karışım	7
Şekil 4. TBA algoritmasının çalışma prensibi.....	10
Şekil 5. BBA algoritmasının çalışma prensibi.....	11
Şekil 6. Kanonik Korelasyon Analizinin Genel Yapısı.....	14
Şekil 7. Öznitelik eşlemede sınıflandırma işlemi	26
Şekil 8. Öznitelik uzayına eşlenmiş ikili sınıflandırma problemi örneği. (a) Elips düzlem (b) Hiper düzlem	29
Şekil 9. Dalgacık Dönüşümü zaman-frekans gösterimi	36
Şekil 10. Dalgacık Dönüşümü	36
Şekil 11. Zaman-Frekans gösterimi.....	37
Şekil 12. Ayrık dalgacık dönüşümünün tek seviyeli filtre algoritması	39
Şekil 13. İşaretin yaklaşık ve detay bileşenlerine ayrıştırılması.....	39
Şekil 14. Ters dalgacık dönüşüm şeması.....	40
Şekil 15. Ters dönüşüm filtreleme.....	41
Şekil 16. Piyano sesine ait kaynak işareti.....	43
Şekil 17. Gauss gürültüsüne ait kaynak işareti	44
Şekil 18. Konuşma sesine ait kaynak işareti	44
Şekil 19. Üç adet kaynak sinyalinin üç sensörle kaydedilip karıştırılmış şekli.....	45
Şekil 20. Piyano için Monte Carlo analizi İGO sonuçları	46
Şekil 21. Gauss gürültüsü için Monte Carlo analizi İGO sonuçları	46
Şekil 22. Konuşma sesi için Monte Carlo analizi İGO sonuçları	47
Şekil 23. Piyano sesi için örnek sayısı.....	47
Şekil 24. Gauss gürültüsü için örnek sayısı.....	48
Şekil 25. Konuşma sesi için örnek sayısı	48
Şekil 26. Üç adet kaynak sinyalinin üç sensörle kaydedilip doğrusal olmayan yöntemle karıştırılmış şekli.....	51
Şekil 27. Piyano için Monte Carlo analizi İGO sonuçları (Doğrusal olmayan karışım)	52

Şekil 28. Gauss gürültüsü için Monte Carlo analizi İGO sonuçları (Doğrusal olmayan karışım)	53
Şekil 29. Konuşma sesi için Monte Carlo analizi İGO sonuçları (Doğrusal olmayan karışım)	53
Şekil 30. Piyano sesi için örnek sayısı (Doğrusal olmayan karışım).....	54
Şekil 31. Gauss gürültüsü için örnek sayısı (Doğrusal olmayan karışım).....	54
Şekil 32. Konuşma sesi için örnek sayısı (Doğrusal olmayan karışım)	55

TABLULAR DİZİNİ

Sayfa No

Tablo 1. Doğrusal olarak karışmış algoritmaların 500 veri boyutlu SNR değerleri.....	49
Tablo 2. Doğrusal olarak karışmış algoritmaların 1000 veri boyutlu SNR değerleri.....	49
Tablo 3. Doğrusal olarak karışmış algoritmaların 5000 veri boyutlu SNR değerleri.....	50
Tablo 4. Doğrusal olarak karışmış algoritmaların 10000 veri boyutlu SNR değerleri.....	50
Tablo 5. Doğrusal olarak karışmış algoritmaların 15000 veri boyutlu SNR değerleri.....	50
Tablo 6. Algoritmaların çalışma süreleri	51
Tablo 7. Doğrusal olmayan yöntemle karışmış algoritmaların 500 veri boyutlu SNR değerleri	55
Tablo 8. Doğrusal olmayan yöntemle karışmış algoritmaların 1000 veri boyutlu SNR değerleri	56
Tablo 9. Doğrusal olmayan yöntemle karışmış algoritmaların 5000 veri boyutlu SNR değerleri	56
Tablo 10. Doğrusal olmayan yöntemle karışmış algoritmaların 10000 veri boyutlu SNR değerleri	57
Tablo 11. Doğrusal olmayan yöntemle karışmış algoritmaların 10000 veri boyutlu SNR değerleri	57
Tablo 12. Algoritmaların çalışma süreleri (Doğrusal olmayan karışım).....	58

SEMBOLLER DİZİNİ

A	: Karıştırma Matrisi
ADD	: Ayrık Dalgacık Dönüşümü
b	: Öteleme (Zaman) Parametresi
BBA	: Bağımsız Bileşen Analizi
ÇKKA	: Çekirdek Kanonik Korelasyon Analizi
D	: Yüksek Frekanslı Bileşenler
DD	: Dalgacık Dönüşümü
FD	: Fourier Dönüşümü
İGO	: İşaret Gürültü Oranı
K	: Gram Matris
KKA	: Kanonik Korelasyon Analizi
Kor	: Korelasyon
KZFD	: Kısa-Zamanlı Fourier Dönüşümünü
L	: Lagrange Fonksiyonu
N	: Kaynak İşaret Sayısı
SDD	: Sürekli Dalgacık Dönüşümü
TBA	: Temel Bileşen Analizi
U, V	: Kanonik Değişkenler
Var	: Varyans
X, Y	: Değişken Kümeleri
a_{11}	:1. kaynağın 1.sensöre olan uzaklık katsayısı
a_{12}	:2. kaynağın 1.sensöre olan uzaklık katsayısı
a_{21}	:1. kaynağın 2.sensöre olan uzaklık katsayısı

a_{22}	:2. kaynağın 2.sensöre olan uzaklık katsayısı
$e(t)$: Gauss Dağılımlı Gürültü
μ	: Ortalama Vektör
S_1	:1. Kaynak Sinyali
S_2	: 2. Kaynak Sinyali
$t - \delta_{1j}$: İletim Gecikmesi
x_i	: Karışım Sinyali
λ	: Lagrange Çarpanı
$\psi(t)$: Dalgacık Fonksiyonu

1. GENEL BİLGİLER

1.1. Giriş

Kaynak ayırıştırma; en az iki işaretin karışımlarının gözlenmesi ve bu gözlenen karışımlardan yola çıkılarak, karışımları oluşturan kaynak işaretlerin kestirilmesi işlemine verilen genel addır. Birden fazla işaretten oluşan karışımların ayırıştırılmasında, karışımları oluşturan kaynak işaretler hakkında bazı varsayımlarda bulunmak ve karışım sürecini göz önünde bulundurmak gerekmektedir. Karışım süreci ve kaynak işaretler hakkında bilgi sınırlı olduğunda, yapılan ayırıştırma işlemi kör kaynak ayırıştırma olarak adlandırılmaktadır [1].

Bu alanda yapılan ilk çalışmalar kaynak işaretlerin istatistiksel özelliklerinden yola çıkarak ayırıştırma işlemini gerçekleştirmeyi planlamıştır [2]. Ancak bu durum, kaynaklar hakkında bazı varsayımları gerektirmektedir. Bu varsayımlar [3] kaynak işaretlerin istatistiksel bağımsız olmasını gerektirir.

Ses uygulamalarında kaynak sayısı kadar sensörle (mikrofonla) kaydedilen karışım sinyallerinden orijinal kaynak seslerine ulaşmak için birçok yöntem kullanılmıştır. Her yöntemin kendine göre avantaj ve dezavantajları vardır. Karışım sinyallerinden tekrar kaynak sinyalleri elde etmedeki başarı oranı ve bu zamanda geçen süre oldukça önemlidir.

Bu tezde çekirdek kanonik korelasyon analizi (ÇKKA) [4] baz alınarak kanonik korelasyon analizi (KKA) [5] ve ayrık dalgacık dönüşümü (ADD) [6] algoritmaları kullanılmıştır. ADD algoritması kaynak verilerini düşük ve yüksek frekans bileşenlerine ayırarak işlem yaptığından dolayı işlem süresini önemli ölçüde düşürmektedir. Yapılan çalışmada veriler önce ADD'de işleme tabi tutulup daha sonra ÇKKA'ya uygulanmıştır. Bu kaynak veriler ADD'ye uygulandıktan sonra birde KKA algoritmasına uygulanmıştır. Kaynaklardan elde edilen veriler sensörlerde önce yapay olarak doğrusal yöntemle karıştırılmış ve algoritmalar yardımıyla ayırıştırılmıştır. Aynı veriler daha sonra yapay olarak doğrusal olmayan karıştırma yöntemleri ile karıştırılarak da tekrar ayırıştırılmıştır. Dolayısıyla doğrusal olarak karıştırılan verilerin ayırıştırılmasında ÇKKA, KKA, ADD+ÇKKA ve ADD+KKA algoritmaları kullanılmıştır. Aynı adımlar doğrusal olmayan karıştırma yöntemleri kullanılarak da yapılmış ve kaynak verilerin ayırıştırılması için yine

aynı algoritmalar kullanılmıştır. Bu sayede doğrusal ve doğrusal olmayan karıştırma yöntemleri kullanılarak algoritmaların performansları karşılaştırılmıştır.

Bu uygulamalardaki amacımız algoritmaların performansını artırmak aynı zamanda da işlem sürelerini kısaltmaktır.

Çalışmamız yaygın olarak ortam dinlemelerinde, telefon dinlemelerinde, müzik için enstrüman veya sesin ayrıştırılmasında, canlılar için sağlık problemlerinde, hamile bayanlar için çocuk sağlığında ve gürültülü ortamlarda sesin ayrıştırılmasında yaygın olarak kullanılmaktadır.

1.2. Kör Kaynak Ayrıştırma

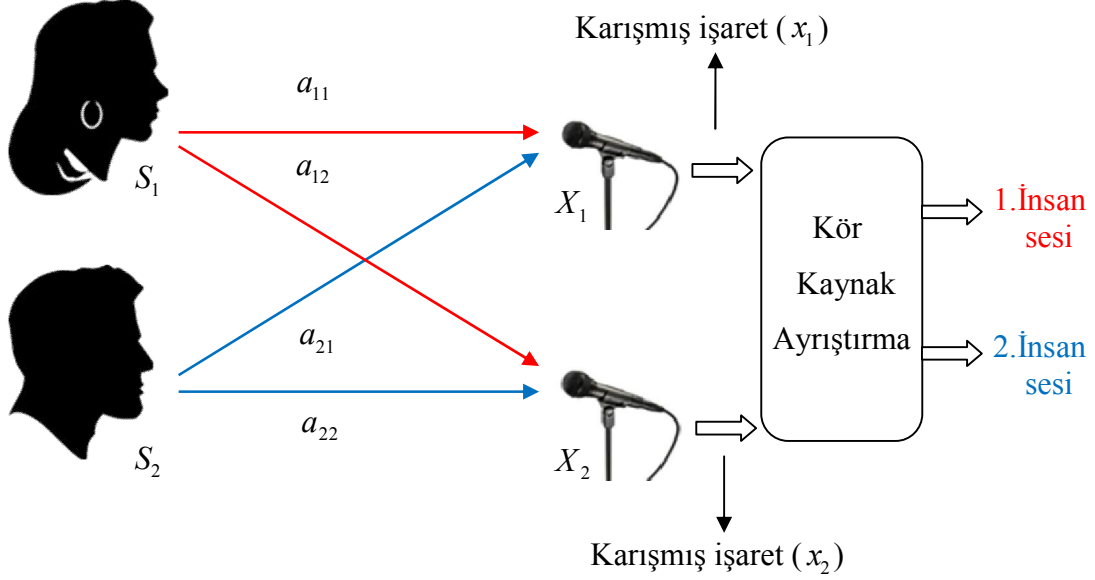
Kör kaynak ayrıştırma son zamanlarda üzerinde çalışılan ve geliştirilen bir konudur. Bu alanda yapılan ilk çalışmalar kaynak işaretlerin istatistiksel özelliklerinden yola çıkarak ayrıştırma işlemini gerçekleştirmeyi amaçlamıştır [7]. Bu amaç doğrultusunda 1986 yılında yapılan bir çalışmada amaç [8] Gauss olmayan bağımsız işaretlerden elde edilmiş, anlık doğrusal, tam-tanımlı karışımı ayrıştırmak planlanmıştır. Ancak bu durum, kaynaklar hakkında bazı varsayımların yapılmasını gerektirmektedir. Bu varsayımlar [3] nolu kaynakta verilmekte olup kaynak işaretlerin istatistiksel bağımsız olması ve Gauss dağılıma sahip olmamasından bahsetmektedir. Yapılan bir çalışmada da karşılıklı bilgi miktarı kullanılarak yapılan ayrıştırma işlemine bir alternatif yaklaşım olarak 1990 yılında maksimum olasılıklı kestirim önerilmiştir [9]. 1994'te yapılan başka bir çalışmada kaynakların birbiriyle bağımsızlığı maksimize edilerek kaynak işaretlerinin belirlenmesi işlemi başarılmış ve bu işleme Bağımsız Bileşen Analizi (BBA) ismi verilmiştir [7]. 1995'te iki araştırmacı BS-Infomax adında, Amari ve diğerlerinin 1996'da önerdiği sade istatistiksel gradyan öğrenme kuralını kullanan [10] nolu kaynakta verilen bir kör kaynak ayrıştırma algoritması geliştirmişlerdir. 1997'de kaynak işaretlerinin Gauss olmaması fikri Hyvarinen ve Oja tarafından yeni bir algoritma geliştirmek için kullanılmıştır [11]. Kör kaynak ayrıştırma algoritmaları kullanılarak müzik işaretleri birbirinden ayrıştırılmış fakat bu uygulamalar yalnızca ses işaretleri ile sınırlı kalmamıştır. Aynı zamanda kör kaynak ayrıştırma ile 1999 ve 2000 yılında yapılan çalışmalarda beyin resimlerinin ayrıştırılması ile olumlu sonuçlar elde edilmiştir [12].

Kör kaynak ayrıştırma problemi genel olarak şu şekilde ifade edilmektedir: N adet kaynak işaretinin, $M \times N$ boyutlu bilinmeyen bir A karışım matrisi yardımıyla birbirine karıştırılması ve karışıma giren kaynak işaretlerin kestirilmesi işlemidir [13].

Karışım işaretlerinin gözlemlerinin yapıldığı ortamdaki çevresel varsayımlar da problemin karmaşıklığını etkilemektedir. Akustik işaretlerin kör kaynak ayrıştırma yöntemleriyle ayrıştırılması genellikle Kokteyl Partisi Problemi (“Cocktail Party Problem”) olarak bilinmektedir [14]. Kör kaynak ayrıştırma, kokteyl parti problemi gibi kontrol edilemeyen akustik bir ortamda birbirine karışmış çok sayıda sesin ayrıştırılması problemidir. Genel olarak kör kaynak ayrıştırma teknikleri, bu zorlayıcı gerçek ortam senaryolarından ayrılarak problemi daha kolay hale getirebilmek için çevresel faktörler hakkında bazı gerçekçi varsayımlar oluştururlar. Bu varsayımlar üç farklı şekildedir. Bunlardan birincisi ve en önemlisi, tüm işaretlerin aynı anda fakat farklı yoğunluklarla sensörlere (alıcıya veya mikrofona) ulaştığı durum olan anlık karışım modelidir. İkincisi, işaretlerin sensörlere ulaşma zamanları arasında bir gecikme söz konusu olduğunda bu durum yansız durum olarak adlandırılır. Üçüncüsü ise, yansız durum göz önüne alındığında verinin her kaynak ve her sensör arasında birden çok yoldan geldiği bilgisi eklenerek yansız durumu oluşturmasıdır [1].

1.3. Kokteyl Parti Problemi

Bağımsız Bileşen Analizi (BBA) yönteminin anlatımı en iyi kokteyl parti problemiyle ifade edilebilir. Kokteyl parti probleminde, aynı ortamda bulunan birden çok kişinin seslerinin birbirinden ayrıştırılması amaçlanmaktadır. Kalabalık bir ortamda, üç farklı kişiden yayılan konuşma sesi bir karışım oluşturmaktadır. Bu konuşma karışımındaki her sesin kaynağı, ayrıştırılması gereken ve birbirlerinden tamamen bağımsız bileşenler olarak tanımlanarak karışımdan ayrıştırılmaya çalışılır. Şekil 1’de temsili bir kokteyl parti problemi gösterilmektedir. Burada S ’ ler kaynak işaretleri, a ’ lar ise karışım katsayılarını ifade etmektedir. Karışım katsayılarının belirlenmesinde, konuşan kişilerin ses yüksekliği, mikrofona uzaklığı gibi faktörler etkilidir. Ayrıca ortamda bulunan gürültü de hesaba katılması gereken önemli bir unsur olarak yer almaktadır [15].



Şekil 1. Kokteyl parti problemi

İstatistiksel analiz yöntemleri tek değişkenli ve çok değişkenli analiz yöntemleri olmak üzere ikiye ayrılmaktadır. Günümüzde yapılan çalışmalar, sağlıklı ve güvenilir sonuçlar verebilmesi açısından bütün yönleriyle ele alınmaktadır, bu yüzden araştırmalarda mümkün olduğunca bütün değişkenler incelenmeye çalışılmakta ve değişkenlerin ayrı ayrı etkilerinin ele alınması gerekmektedir. Bu yüzden tek değişkenli analiz yöntemlerinin kullanımı giderek azalmakta ve çok değişkenli analiz yöntemleri ise ön plana çıkmaktadır. Çok değişkenli istatistiksel yöntemlerden yaygın olarak kullanılanlara; çok değişkenli varyans analizi (multivariate analysis of variance), kümeleme analizi (clustering analysis), temel bileşenler analizi (principle component analysis), ayırma analizi (discriminant analysis), regresyon analizi (regression analysis), kanonik korelasyon analizi (canonical correlation analysis) örnek olarak verilebilir [16,17]. Bu çalışmada önemli bir yere sahip olan kanonik korelasyon analizi yöntemi kullanılmıştır [18].

1.4. Karışım Modelleri

Kör kaynak ayırma probleminin matematiksel olarak gösterimi denklem (1)'de gösterilmiştir.

$$x(t) = As(t) + e(t) \quad (1)$$

Burada $s(t)$;

$$s(t) = \begin{bmatrix} s_1(0) & s_1(1) & \cdots & s_1(n) \\ s_2(0) & s_2(1) & \cdots & s_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ s_N(0) & s_N(1) & \cdots & s_N(n) \end{bmatrix} \quad (2)$$

olmak üzere N adet kaynak işaretinden oluşmaktadır, $x(t)$ ise;

$$x(t) = \begin{bmatrix} x_1(0) & x_1(1) & \cdots & x_1(n) \\ x_2(0) & x_2(1) & \cdots & x_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ x_M(0) & x_M(1) & \cdots & x_M(n) \end{bmatrix} \quad (3)$$

olmak üzere kaynak işaretlerin M adet sensörle kaydedilen n adet gözlemin oluşturduğu karışım işaretleridir. A ise;

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{MN} \end{bmatrix} \quad (4)$$

olmak üzere $M \times N$ boyutlarında bilinmeyen doğrusal karışım matrisidir. $e(t)$ gauss dağılımlı gürültüyü ifade etmek için kullanılır. Karışım matrisinin elemanları olan a_{ij} katsayıları karışım modeline bağlıdır ve modelin anlık, yansız ya da yansılı olacağını belirler [1,2].

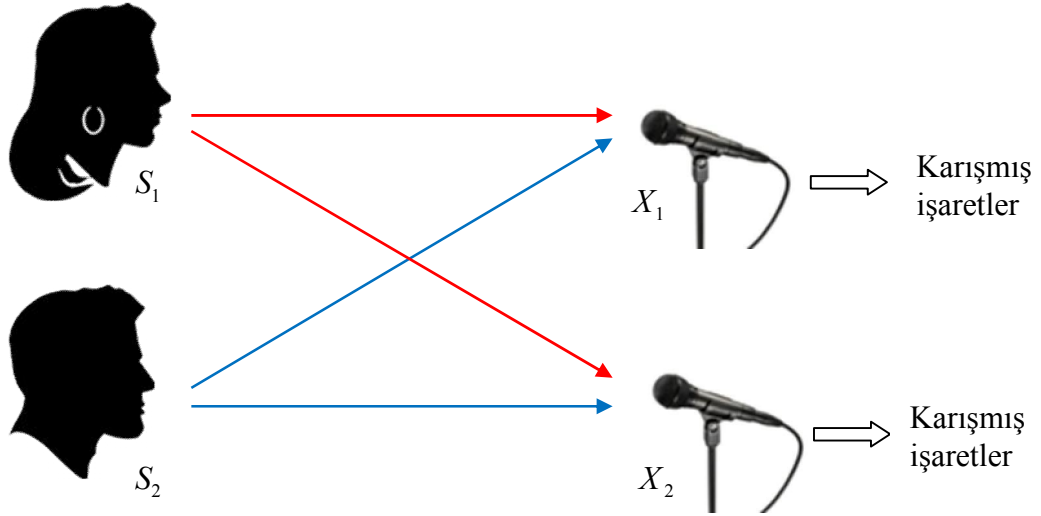
1.4.1. Anlık Karışım Modeli

Anlık karışım modelinde karışım matrisi A basit olarak işaret genliklerini temsil eden sayısal değerlerden oluşmaktadır. Örneğin, iki adet kaynak işaret ve iki adet karışımın olduğu durumda model aşağıdaki gibi ifade edilir [1].

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} S_1(t) \\ S_2(t) \end{bmatrix} \quad (5)$$

1.4.2. Yansımatsız Karışım Modeli

Yansımatsız karışımında anlık karışımından farklı olarak, kaynak işaretlerinin sensörlere olan farklı uzaklıklarından dolayı iletim gecikmeleri ortaya çıkmaktadır.



Şekil 2. Yansımatsız karışım

Yansımatsız işaretlerin ayrıştırılması problemi, her kaynak işaretine ilişkin sönmüleme katsayısının ve göreceli gecikmenin belirlenmesidir. Yansımatsız durum Şekil 2'de gösterilmiştir [1].

Örnek olarak n adet kaynağın ve n adet sensör karışımı göz önüne alacak olursak, karışım modeli aşağıdaki gibi olur.

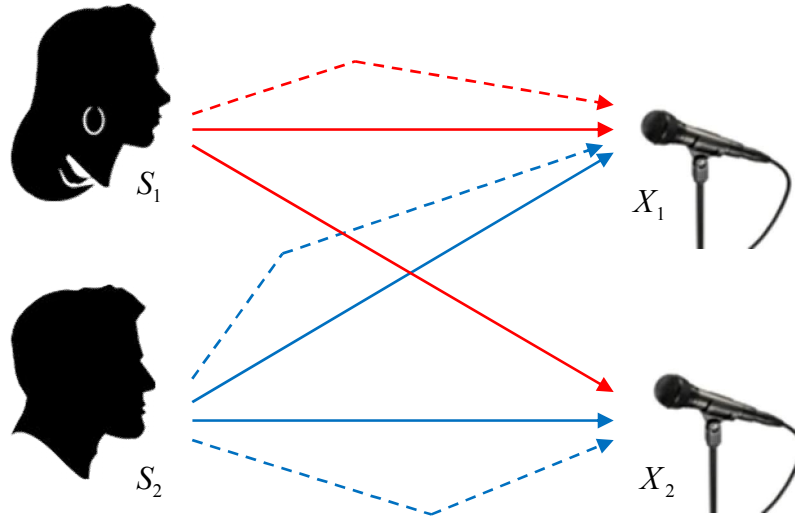
$$x_1(t) = \sum_{j=1}^n a_{1j} s_j(t - \delta_{1j}) \quad (6)$$

$$x_2(t) = \sum_{j=1}^n a_{2j} s_j(t - \delta_{2j}) \quad (7)$$

Burada $(t - \delta_{1j})$ iletim gecikmesini temsil eder.

1.4.3. Yansımalı Karışım Modeli

Yansımalı karışım modelinde ise iletim gecikmelerinin yanı sıra iletim yansımaları da göz önünde bulundurulur. Dolayısıyla bu yaklaşım daha karmaşık bir modelin ortaya çıkmasına sebep olur. Yansımalı karışım modeli Şekil 3'de gösterilmiştir. Şekilde kesikli şekille çizilen dalgalar yansıyan dalgaları göstermektedir. İletim yansımaları ses kaynağından çıkan seslerin birtakım engellere çarparak sensörlere ulaştığı yollar olarak da tanımlanabilir [1].



Şekil 3. Yansımalı karışım

İki karışım ve iki kaynak işaretinden oluşan bir model aşağıda verilmiştir.

$$x_1(t) = \sum_{k=1}^L a_{11}^k s_1(t - \delta_{11}^k) + a_{12}^k s_2(t - \delta_{12}^k) \quad (8)$$

$$x_2(t) = \sum_{k=1}^L a_{21}^k s_1(t - \delta_{21}^k) + a_{22}^k s_2(t - \delta_{22}^k) \quad (9)$$

Burada L , kaynak işaretlerin sensöre ulaştığı farklı yolların sayısıdır.

1.5. Temel Bileşen Analizi

Temel Bileşenler Analizi istatistiksel örüntü tanılama ve sinyal işleme gibi işlemlerde veri boyutu küçültmede ve nitelik çıkarımında kullanılan matematiksel bir yöntemdir [19]. TBA veri boyutunun azaltılması, kayıp veri sıkıştırması, öznelik çıkartımı ve veri temsili gibi işlemlerde de sıkça kullanılabilir [20]. TBA'nin temeli yaklaşık bir asır öncesinden, Pearson'ın [21] çalışmasına uzanmaktadır. Pearson, en küçük karelerin optimizasyonuna dayanan n boyutlu bir doğrusal regresyon yöntemi önermiştir [22]. Ancak, çok boyutlu rastgele değişkenlerin varyans analizi için yeni bir yöntem 1930 yılında Hotelling tarafından yapılmıştır [23,24].

Kaydedilmiş veriye Bağımsız Bileşen Analizi (BBA) algoritması uygulamadan önce bazı ön işlemler yapmakta fayda vardır. Bunun için değişkenin ortalama değerinin sıfıra eşitlenmesi gerekir. Matematiksel ifadesi;

$$x = x - E\{x\} \quad (10)$$

denklemleri ile tanımlanır.

Diğer bir ön işlem ise, işaretin beyazlaştırılmasıdır. Beyazlaştırma, TBA algoritması ile işaretlerin ilintisiz hale getirilmesidir. İlintisiz hale getirilen işaretin ilinti matrisinin matematiksel ifadesi,

$$E\{xx^T\} = I \quad (11)$$

denklemleri ile ifade edilir. İşaretlerin temel bileşenlerinin bulunma işlemi kümesel beyazlatma işlemi olarak da yapılabilir. Kümesel beyazlatma yöntemlerinden biri kovaryans matrisinin,

$$R_x = E \{xx^T\} = EDE^T \quad (12)$$

denklemindeki gibi öz değerlerine ve öz vektörlerine ayrıştırılmasıdır. Denklem (12)'deki E , kovaryans matrisinin öz vektörlerinin ortogonal matrisini ve D 'de öz değerlerinin köşegenel matrisini ifade etmektedir. Bu öz değerler ve öz vektörler elde edildikten sonra beyazlaştırma işlemi,

$$\tilde{x} = ED^{-1/2}E^T x \quad (13)$$

denklemleri ile ifade edilebilir. Burada,

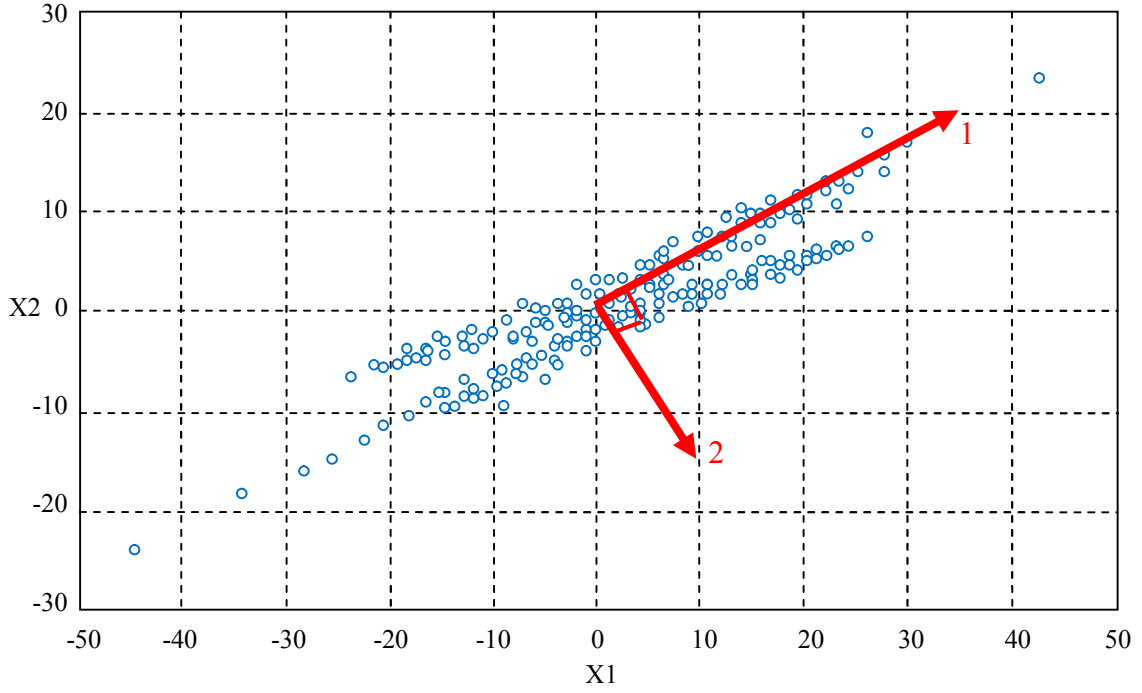
$$D^{-1/2} = \text{diag}(d_1^{-1/2}, \dots, d_n^{-1/2}) \quad (14)$$

işlemi ile bulunur. Beyazlatma işleminin A matrisi üzerine etkisi,

$$\tilde{x} = ED^{-1/2}E^T A s = \tilde{A} s \quad (15)$$

ifadesinde görülebileceği gibi \tilde{A} matrisi artık ortogonal bir matristir.

Çoğu BBA algoritmasında TBA algoritması ön işlem olarak kullanıldığından BBA algoritmasının TBA algoritmasından daha iyi sonuç verdiği kesindir. Fakat bu grafiksel olarak gösterilmek istenirse: Şekil 4'de görüleceği üzere TBA algoritması ilk işlem olarak varyansı en büyük olan bileşenin yönünü bulacaktır (1 numaralı çizgi), ikinci adım olarak da ilk varyansa dik olan ikinci en büyük varyansa sahip bileşeni bulacaktır (2 numaralı çizgi). Bulunan bu ikinci yön dikkat edilirse ikinci en büyük varyansa sahip bileşenin yönünü göstermemektedir [25].



Şekil 4. TBA algoritmasının çalışma prensibi

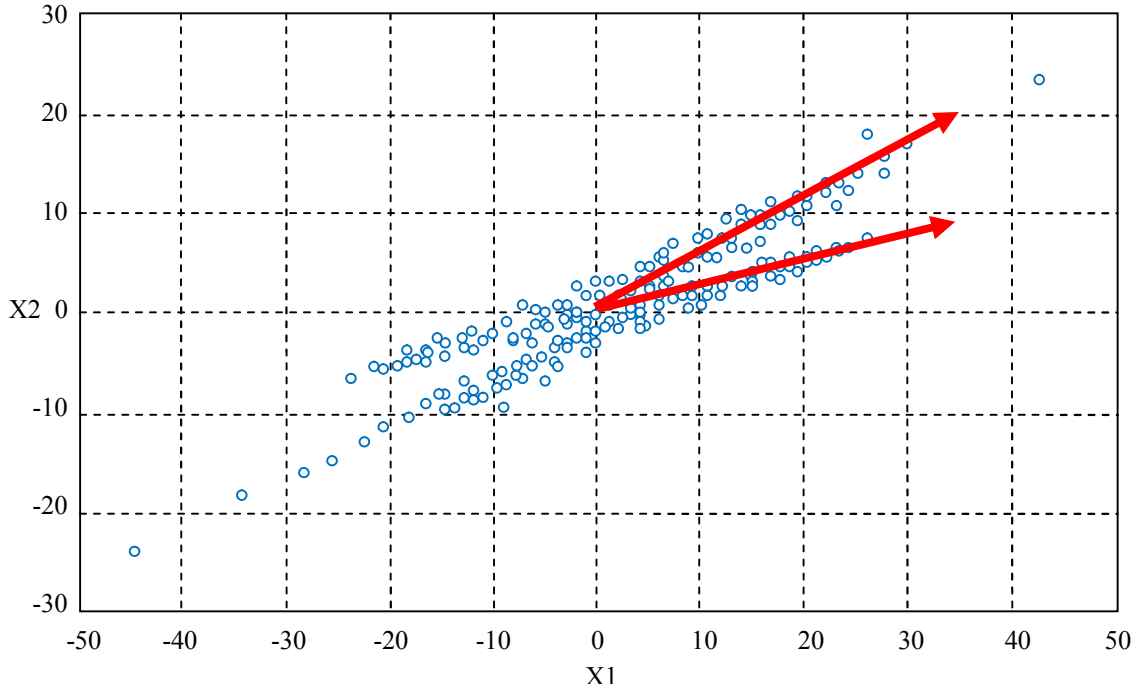
1.6. Bağımsız Bileşen Analizi

BBA'nın, amacı verilen bir veri dizisi içerisinde istatistiksel olarak bağımsız sinyalleri elde etmektir. BBA'da bağımsız bileşenler bilinmeyen bir karıştırma matrisi ile karıştırılarak gözlem verilerini meydana getirirler. Doğrusal BBA'da karıştırma matrisinin kare, kaynakların gaussian olmayan dağılıma sahip olduğu varsayılmaktadır. BBA'da bağımsız bileşenlerin hesaplanması için yüksek dereceden bilgilerin bilinmesi gerekmektedir. Gaussian dağılımda ise bu bilgiler sifıra eşittir. Bu yüzden BBA, gaussian dağılıma sahip verilerde sonuç vermemektedir. Karıştırma matrisinin kare kabul edilmesi, kaynak sayısı ile bağımsız bileşen sayısının eşit olması anlamına gelir. Bu varsayım ayrıştırma matrisinin tahminini kolaylaştırmaktadır [26,27]. Bu tanımlamalara bağlı olarak BBA algoritmalarındaki bağımsız kaynak işaretlerinin karışımının matematiksel ifadesi denklem (16)'da verilmiştir.

$$x_i = a_{i1}s_1 + a_{i2}s_2 + \dots + a_{in}s_n \quad (16)$$

Burada x_i karışım sinyalini gösterirken, s_i bağımsız sinyal kaynağını göstermektedir. a_i ise ölçüm sinyalindeki karıştırma matrisinden kaynaklanan katsayılarıdır. Burada tek bilinen x_i karışım işaretleridir ve bu işaretlerden yola çıkılarak hem a_i karışım katsayıları hem de s_i kaynak işaretleri kestirilmeye çalışılacaktır.

Mikrofonların aldığı işaretler, kaynak işaretlerinin farklı doğrusal kombinasyonlarına sahip olmalı, aralarında gecikme zamanı olmamalı ve ortamda kaynak (işareti) sayısı kadar mikrofon bulunduğu varsayılmalıdır. Bu varsayımların tamamı sağlanmasa dahi BBA genellikle kaynak işaretleri için iyi kestirim değerleri vermektedir. Belirtilen koşulların sağlanması için veri kümesinden beklenen değer çıkarılır ve veri kümesi ortalanan, ardından giriş matrisinin kovaryans değerleri bulunur, bu matrisin öz değer vektörleri azalan sırada dizilerek veri kümesi için beyazlatma işlemi gerçekleştirilir [28,29].



Şekil 5. BBA algoritmasının çalışma prensibi

BBA algoritması TBA algoritmasının aksine Şekil 5’de görüleceği üzere doğrudan ikinci bileşenin yönünü bulacaktır. Çünkü BBA algoritmasının amacı bulunan bileşenlerin istatistiksel olarak bağımsız olmasıdır, TBA gibi birbirlerine dik (ilintisiz) olması değildir.

1.6.1. BBA Algoritmasında Kaynak ve Sensör Sayı Bağlantısı

Genelde BBA algoritmalarında gözlenen sinyallerin sayısı kaynakların sayısına eşit olduğu varsayılır $s(t) = Wx(t)$ ve A karışım matrisi bir kare matris olmaktadır. Karesel durumda ek bir bilgiye gerek kalmadan kaynak işaretleri bulunabilir.

$M > N$, kaynaklarının sayısının kaydedicilerin sayısından az olduğu durumda, A karışım matrisi aşırı-tanımlı bir matris olur ve problem çok kolay bir şekilde, temel bileşen analiz (TBA) algoritmasının kullanılması ile karesel duruma dönüştürülebilir, çözüme devam edilip kaynak işaretleri bulunabilir.

$M < N$ durumunda kaynak sayısı kaydedici sayısından büyük olduğunda A karışım matrisi az-tanımlı bir matris olur ve bu durum tam olmayan durum olarak isimlendirilir. Bu problemin çözümü çok zordur ve ek bilgi olmadan eşsiz bir çözüm bulunamaz. İlk tam olmayan durumlu BBA probleminin çözümüne yönelik yaklaşım Lewicki ve Sejnowski [30] tarafından 1998 yılında yapılmıştır. Daha sonra Theis 2003 yılında geometrik BBA yaklaşımını [31] kullanarak bu problemi çözümlenmiştir fakat sonuçlar çok da iyi değildir.

Karesel durum göz önünde bulundurulursa, elde edilmiş karıştırıcı matris W' ' nin A matrisinin tersine benzer olduğu söylenebilir, fakat aynı değildir. Bundan dolayı yeniden oluşturulmuş kaynakların, orijinal kaynak sinyallerinden sadece ölçeklendirme ve permutasyon ile farklı olduğu söylenebilir. Permutasyon ve ölçeklendirme ifadelerinden yararlanarak bulunan kaynak işaret yaklaşımları,

$$y(t) = Wx(t) \quad (17)$$

olarak elde edilir [25,1].

1.7. Kanonik Korelasyon Analizi

Çok sayıda değişkenden oluşan iki veri kümesi arasındaki ilişki yapısını analiz etmek için kullanılan kanonik korelasyon analizi, Hotelling tarafından 1936 yılında geliştirilmiştir. Kanonik korelasyon analizinin amacı iki değişken kümesi arasındaki ilişkileri, söz konusu değişken kümelerinin doğrusal fonksiyonları arasındaki maksimum korelasyonları bulmaya çalışmaktır [5,32].

Bu analizde amaç her bir kümenin rastlantı değişkenlerinin maksimum korelasyonlu ve birim varyanslı doğrusal bileşenlerini bulmaktır. Diğer amaçlar ise şöyle sıralanabilir [33]:

- İki değişken kümenin istatistiksel olarak birbirinden bağımsız olup olmadığının belirlenmesi.
- Kümeler arası korelasyona en çok katkıda bulunan her iki kümedeki değişkenlerin test edilmesi,
- Bağımsız ve bağımlı değişken kümelerine ait değişkenler arasındaki korelasyonu maksimum yapan doğrusal birleşimlerin bulunması,
- Değişken kümelerinin birindeki doğrusal kombinasyon değerlerine bakarak diğer kümedeki değerlerin tahmin edilmesi.

Doğrusal kanonik korelasyon analizinde bir bağımlı değişken söz konusu olduğunda doğrusal kanonik korelasyon analizi çoklu regresyon analizine dönüşmektedir [34]. Doğrusal kanonik korelasyon analizinde bir tek bağımlı ve bir tek bağımsız değişken olması durumunda ise analiz basit regresyon analizi halini alır. Varyans analizi regresyon analizinin özel bir hali olduğundan dolayı aynı zamanda doğrusal kanonik korelasyon analizinin de özel bir şeklidir [35]. Doğrusal kanonik korelasyon analizi, regresyon analizinin birden fazla bağımlı değişken için genelleştirilmiş şeklidir [36].

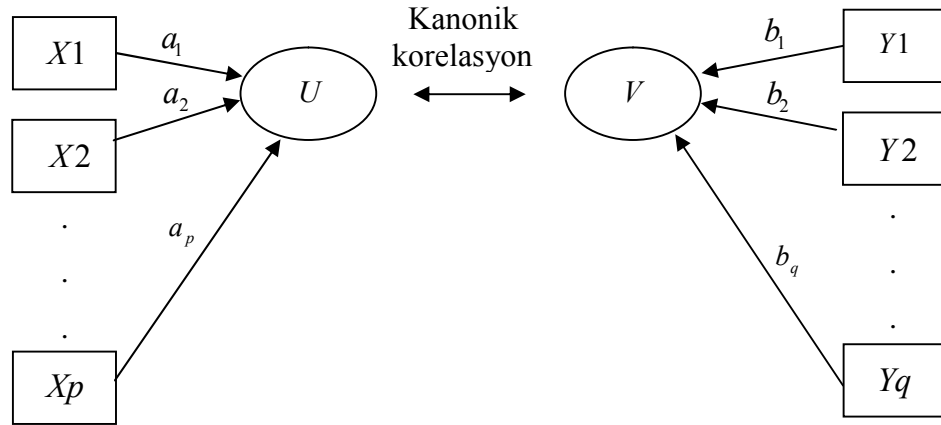
Kanonik korelasyon analizinin uygulanabilmesi için veri kümelerinin ve bu veri kümelerinde yer alan değişkenlerin bazı varsayımlara göre ayarlanması gerekmektedir. Bu varsayımlar sıralandığında;

- Kanonik korelasyon analizi uygulanacak veri matrisinde gereğinden fazla ve problemle ilgisi olmayan değişkenleri içermemesi gerekir. Veri matrisinde bir değişkenin iki kez yer alması, bir değişkenin karesinin, küpünün vb. formlarının birlikte yer alması gibi koşullarda veri aşırı bilgi içermekte ve kanonik korelasyon analizi için kötü koşullama oluşturabilmektedir.
- Veri setlerinde aykırı değerlerin bulunmaması gerekmektedir. Aykırı değerler, değişkenler arasındaki korelasyonları önemli düzeyde etkilediklerinden dolayı aykırı değerlerin önceden saptanarak gerekli düzeltmelerin veya ayıklamaların yapılması gerekmektedir.

- Kanonik korelasyon analizi ile elde edilen sonuçların güvenli olması için kümelerdeki veri sayısının gerektiğinden fazla olması önerilmektedir.
- Kanonik korelasyon analizinde kullanılan değişken kümelerinde yer alan değişkenlerin eşit sayıda olması gerekmez.
- Kanonik korelasyon analizi kovaryans matrisi ya da korelasyon matrisi yardımı ile uygulanmaktadır. Bu yüzden veri kümelerinden elde edilen kovaryans ve korelasyon matrisleri parçalanabilir olmalı ve tersi alınabilir olmalıdır [18,37].

Yukarıda verilen ifadelerle dikkat edilmesi gerekmektedir. Eğer veri matrisi yukarıda bahsedilen varsayımları sağlamıyorsa mümkünse veri kümesinde gerekli düzeltmeler yapıldıktan sonra doğrusal kanonik korelasyon analizine uygulanmalıdır. Düzeltmeler yapılamıyorsa doğrusal kanonik korelasyon analizi kullanılmamalıdır.

KKA'nın genel yapısı Şekil 6'da görülmektedir.



Şekil 6. Kanonik Korelasyon Analizinin Genel Yapısı

Kanonik korelasyon katsayısı, iki değişken arasındaki ilintinin bir ölçüsüdür ve 0 ile +1 arasında değişim gösterir. Kanonik korelasyonun 1 olması mükemmel doğrusal ilişkiyi göstermektedir. Korelasyon katsayısının 0.9 - 1 arasında olması çok yüksek bir ilintiyi, 0 - 0.25 arasında olması ise zayıf bir ilintiyi temsil etmektedir.

Denklem 18’de iki ayrı veri kümesi görülmektedir, X kümesine ait veri matrisi p değişken ve Y kümesine ait veri matrisi de q değişken içermekte ve her bir değişkenin içerisinde de n adet veri bulunmaktadır.

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_p \end{bmatrix}_{p \times n} \quad Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_q \end{bmatrix}_{q \times n} \quad (18)$$

İki değişken kümeleri arasındaki ilişkinin ölçülmesi ile ilgilendiğimizde, birinci değişken kümesi p tane değişken içeren ($p \times 1$) boyutunda X rassal değişken vektörü, ikinci değişken kümesi ise q tane değişken içeren ($q \times 1$) boyutunda Y rassal değişken vektörü ile ifade edilmektedir. Matris işlemlerinin yapılabilmesi için X değişken kümesinin, değişken kümelerinden daha küçük olanını temsil ettiği varsayılmakta, bu nedenle ($p \leq q$) olarak ifade edilmektedir [38].

X değişken kümesi μ_1 ortalama vektörüne, Y değişken kümesi ise μ_2 ortalama vektörüne sahiptir. Bu değişken kümelerine ait ortalama ve kovaryans matrisleri sırası ile denklem (19)’ da görülmektedir [39].

$$\mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}; \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \quad (19)$$

Varyans-kovaryans matrisinin elemanları aşağıda görüldüğü gibi yazılır:

$$\Sigma_{11} = \begin{bmatrix} \sigma_{X_1 X_1} & \sigma_{X_1 X_2} & \cdots & \sigma_{X_1 X_p} \\ \sigma_{X_2 X_1} & \sigma_{X_2 X_2} & \cdots & \sigma_{X_2 X_p} \\ \vdots & \vdots & & \vdots \\ \sigma_{X_p X_1} & \sigma_{X_p X_2} & \cdots & \sigma_{X_p X_p} \end{bmatrix} \quad (20)$$

$$\Sigma_{22} = \begin{bmatrix} \sigma_{Y_1 Y_1} & \sigma_{Y_1 Y_2} & \cdots & \sigma_{Y_1 Y_q} \\ \sigma_{Y_2 Y_1} & \sigma_{Y_2 Y_2} & \cdots & \sigma_{Y_2 Y_q} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{Y_q Y_1} & \sigma_{Y_q Y_2} & \cdots & \sigma_{Y_q Y_q} \end{bmatrix} \quad (21)$$

$$\Sigma_{12} = \begin{bmatrix} \sigma_{X_1 Y_1} & \sigma_{X_1 Y_2} & \cdots & \sigma_{X_1 Y_q} \\ \sigma_{X_2 Y_1} & \sigma_{X_2 Y_2} & \cdots & \sigma_{X_2 Y_q} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{X_q Y_1} & \sigma_{X_q Y_2} & \cdots & \sigma_{X_q Y_q} \end{bmatrix} \quad (22)$$

Σ_{21} matrisi Σ_{12} matrisinin devriğidir. Yukarıda verilen eşitliklerdeki varyans değerleri denklem (23)' deki gibi hesaplanabilir:

$$\sigma_{ij} = \frac{1}{n} \left[\sum_{r=1}^n (X_{ir} - \bar{X}_i)(X_{jr} - \bar{X}_j) \right] \quad (23)$$

Buna göre X ve Y değişken setlerine ait varyans-kovaryans matrisinde, $i = j$ iken varyans değerleri, $i \neq j$ iken kovaryans değerleri hesaplanabilir. Varyans-kovaryans matrisinde köşegende yer alan varyans değerleri, değişkenlerin dağılışı hakkında bilgi verirken, köşegen dışında kalan kovaryans değerleri, değişken çiftleri arasındaki değişim miktarını vermektedir [40].

X ve Y değişken kümelerinden i . kanonik değişken (U_i, V_i) çiftleri aşağıda verilen denklem (24) yardımı ile hesaplanabilmektedir.

$$U_i = a_i' X \quad V_i = b_i' Y \quad (24)$$

Denklem (24)' de verilen a ve b katsayıları, sırası ile $p \times 1$ ve $q \times 1$ ' lik vektörlerdir. Bu katsayılar, X değişken kümesinin çözüm matrisi M_1 ile Y değişken kümesinin çözüm matrisi, M_2 matrislerinin özdeğerlerine karşılık gelen öz vektör elemanlarını temsil eder ve şu şekilde hesaplanır [41];

$$M_1 = \sum_{11}^{-1} \sum_{12} \sum_{22}^{-1} \sum_{21} \quad (25)$$

$$M_2 = \sum_{22}^{-1} \sum_{21} \sum_{11}^{-1} \sum_{12} \quad (26)$$

X ve Y deęişken kümeleri çok deęişkenli normal dağılışa sahip ise, denklem (24)'de verilen U ve V kanonik deęişkenleri de normal dağılışa sahiptir ve kanonik deęişkenleri arasındaki doğrusal ilişki maksimize edilebilmektedir. Eęer, X deęişken kümesi bağımsız deęişken, Y deęişken kümesi bağımlı deęişken olarak ifade edilirse, yani X , Y 'nin sebebi olarak yorumlanırsa, bu durumda U "en iyi tahmin edici", V 'de "en iyi tahmin edilebilir kriter" olarak isimlendirilebilir. Kanonik deęişkenler U ve V 'nin varyans ve kovaryansları denklem (27, 28, 29)'da ifade edilmiştir [42].

$$Var(U) = a' Kov(X) a = a' \sum_{11} a \quad (27)$$

$$Var(V) = b' Kov(Y) b = b' \sum_{22} b \quad (28)$$

$$Var(U, V) = a' Kov(XY) b = a' \sum_{12} b \quad (29)$$

U ve V kanonik deęişkenleri arasındaki korelasyon, yani kanonik korelasyon ise denklem (30)'da verilmiştir [43].

$$r_{uv} = \frac{Kov(U, V)}{\sqrt{Var(U) Var(V)}} = \frac{a' \sum_{12} b}{\sqrt{(a' \sum_{11} a) (b' \sum_{22} b)}} \quad (30)$$

U ve V kanonik deęişkenleri arasındaki korelasyonu maksimize etmek için a ve b katsayılarının maksimum olduęu korelasyon katsayısını bulmamız gerekmektedir. U ve V vektörlerinde yer alan ve birim varyansa sahip olan kanonik deęişken çiftleri $(U_i, V_i \quad i=1, 2, \dots, k)$ korelasyonu maksimize eden deęerlerdir [44]. Bir başka ifade ile,

$$Var(U) = a' \sum_{11} a = 1 \quad (31)$$

$$\text{Var}(V) = b' \sum_{22} b = 1 \quad (32)$$

olduğunda korelasyon maksimum olacaktır. Böylece U ve V kanonik değişken çifti arasındaki maksimum korelasyona birinci kanonik korelasyon adı verilir ve bu korelasyon katsayı denklem (33)'de yazıldığı gibi olacaktır. [45].

$$\max_{a,b} \text{Kor}(U,V) = \rho_1 \quad (33)$$

Burada yapılması gereken bu ifadenin maksimum yapılmasıdır. Bu sebeple katsayıların maksimizasyon problemi olarak düşünülüp ortaya koymak için λ_1 ve λ_2 , Lagrange çarpanları, diğer bir ifade ile Lagrange fonksiyonu denklem (34)' de verildiği biçimde yazılabilir [46].

$$L = a' \sum_{12} b - \frac{1}{2} \lambda_1 (a' \sum_{11} a - 1) - \frac{1}{2} \lambda_2 (b' \sum_{22} b - 1) \quad (34)$$

Bu fonksiyonda a , b , λ_1 ve λ_2 ' ye göre kısmi türev alınıp sifıra eşitlenirse

$$\frac{\partial L}{\partial a} = \sum_{12} b - \lambda_1 \sum_{11} a = 0 \quad (35)$$

$$\frac{\partial L}{\partial b} = \sum_{21} a - \lambda_2 \sum_{22} b = 0 \quad (36)$$

$$\frac{\partial L}{\partial \lambda_1} = (a' \sum_{11} a) - 1 = 0 \Rightarrow a' \sum_{11} a = 1 \quad (37)$$

$$\frac{\partial L}{\partial \lambda_2} = (b' \sum_{22} b) - 1 = 0 \Rightarrow b' \sum_{22} b = 1 \quad (38)$$

denklemleri elde edilir. Denklem (35)'i soldan a' ile denklem (36)'yı da soldan b' ile çarpılırsa

$$a' \sum_{12} b - \lambda_1 (a' \sum_{11} a) = 0 \quad (39)$$

$$b' \sum_{21} a - \lambda_2 (b' \sum_{22} b) = 0 \quad (40)$$

elde edilir. Denklem (37) ve (38)'deki ifadelere göre düzenlenip denklem (39) ve denklem (40) tekrar yazılırsa

$$\lambda_1 = a' \sum_{12} b \quad (41)$$

$$\lambda_2 = b' \sum_{21} a \quad (42)$$

biçiminde bulunmuş olur.

$$a' \sum_{12} b = b' \sum_{21} a \quad (43)$$

olduğundan

$$\lambda_1 = \lambda_2 = a' \sum_{12} b = \rho \quad (44)$$

yazılabilir. Bu bilgilere göre denklem (35) ve denklem (36) aşağıdaki biçimde tekrar yazılabilir.

$$\left. \begin{array}{l} \sum_{12} b - \lambda_1 \sum_{11} a = 0 \\ \sum_{21} a - \lambda_2 \sum_{22} b = 0 \end{array} \right\} \begin{bmatrix} -\rho \sum_{11} & \sum_{12} \\ \sum_{21} & -\rho \sum_{22} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (45)$$

Bu denklem sisteminde a ve b 'nin sıfırdan farklı olabilmesi için ilk matrisin determinantının sıfıra eşit olması gerekmektedir.

$$\begin{vmatrix} -\rho \sum_{11} & \sum_{12} \\ \sum_{21} & -\rho \sum_{22} \end{vmatrix} = 0 \quad (46)$$

Bu determinantın sıfıra eşitlenmesiyle elde edilecek p^2 değeri aşağıdaki denklemlerde yerine konularak a ve b değerleri bulunabilir [39].

$$\left(-\rho^2 \sum_{11} + \sum_{12} \sum_{22}^{-1} \sum_{21} \right) a = 0 \quad (47)$$

$$\left(-\rho^2 \sum_{22} + \sum_{21} \sum_{11}^{-1} \sum_{12} \right) b = 0 \quad (48)$$

Kümelerde yer alan değişkenler farklı ortalama ve varyanslara sahip olması durumunda kanonik korelasyonların korelasyon matrisi kullanılarak ya da değişkenlerin standartlaştırılarak bulunması gerekmektedir [47].

Korelasyon matrisinden faydalanmak üzere kanonik korelasyonların elde edilmesi için H_1 ve H_2 matrislerinden yararlanır. Bu matrisler

$$H_1 = R_{11}^{-1} R_{12} R_{22}^{-1} R_{21} \quad (49)$$

$$H_2 = R_{22}^{-1} R_{21} R_{11}^{-1} R_{12} \quad (50)$$

şeklinde olmaktadır. Birinci kümeye ilişkin kanonik katsayıların bulunmasında H_1 , ikinci kümeye ilişkin kanonik katsayıların hesaplanmasında H_2 matrisinin öz değer-öz vektör çiftlerinden yararlanır.

Öz vektör elemanları e_i ve öz değerleri λ_i olmak üzere $i=1,2,\dots,p$ için birinci kümeye ilişkin standartlaştırılmamış kanonik katsayılar

$$(H_1 - \lambda_i I) e_i = 0 \quad (51)$$

çözümü ile ve f_i öz vektör elemanları olmak üzere $i=1,2,\dots,q$ için ikinci kümeye ilişkin standartlaştırılmamış kanonik katsayılar ise

$$(H_2 - \lambda_i I) f_i = 0 \quad (52)$$

şeklinde ile hesaplanır.

Her bir köke ilişkin öz vektör değerlerinin standartlaştırılması için her küme için $a'a=1$ ve $b'b=1$ koşullarını sağlayacak biçimde katsayılar hesaplanır. Bu işlemler aşağıdaki gibi özetlenebilir.

Birinci kümede i 'inci öz değere ilişkin hesaplanan kanonik değişkenin varyansı $i=1,2,\dots,p$ olmak üzere

$$Var(U_i) = (e_{i1}, e_{i2}, \dots, e_{ip}) R_{11} \begin{bmatrix} e_{i1} \\ e_{i1} \\ \vdots \\ e_{ip} \end{bmatrix} \quad (53)$$

şeklinde hesaplanır. Kümenin standartlaştırılmış kanonik katsayıları ise

$$a_i^* = e_i / \sqrt{Var(U_i)} \quad (54)$$

biçiminde hesaplanır. Benzer şekilde ikinci kümede i 'inci öz değere ilişkin kanonik değişkenin varyansı ve standartlaştırılmış kanonik korelasyon katsayıları $i=1,2,\dots,q$ olmak üzere aşağıdaki gibi hesaplanmaktadır.

$$Var(W_i) = (f_{i1}, f_{i2}, \dots, f_{iq}) R_{22} \begin{bmatrix} f_{i1} \\ f_{i1} \\ \vdots \\ f_{iq} \end{bmatrix} \quad (55)$$

$$b_i^* = f_i / \sqrt{Var(W_i)} \quad (56)$$

Yapılan işlemler sonucunda birinci kümeyle ait standartlaştırılmış kanonik değişkenler

$$U_i = a_{i1}^* X_1 + a_{i2}^* X_2 + \dots + a_{ip}^* X_p \quad (57)$$

şeklinde olur ve ikinci kümeyle ait standartlaştırılmış kanonik değişkenler ise

$$W_i = b_{i1}^* Y_1 + b_{i2}^* Y_2 + \dots + b_{iq}^* Y_q \quad (58)$$

şeklinde hesaplanır.

Kanonik değişkenler ile orijinal değişkenler arasındaki korelasyonlar da analizde önemlidir ve aşağıda açıklanmıştır [48].

1.7.1. Kanonik Değişkenler ile Orijinal Değişkenler Arasındaki Korelasyonlar

X ve Y değişken kümelerinden elde edilen U ve V kanonik değişkenler, hem kendi değişken kümeleri içerisindeki (yani U ile X_1, X_2, \dots, X_p arasında), hem de diğer kümenin orijinal değişkenleri (yani U ile Y_1, Y_2, \dots, Y_q arasında) ile bir ilişkinin olması ve bunun yorumu ile kanonik değişkene herhangi bir orijinal değişkenin ne ölçüde katkı sağladığını ortaya koyma açısından önemlidir [49].

Bu bilgiler doğrultusunda U_i kanonik değişkeni ile kendi kümesindeki X orijinal değişkenler arasındaki korelasyonlar

$$Kor(U_i, X) = \frac{Kov(U_i, X)}{\sqrt{[Köş(Var(U_i))] [Köş(Var(X))]} = \frac{a_i' \sum_{11}}{\sqrt{Köş(\sum_{11})}} \quad (59)$$

denklemleri ile elde edilmektedir. Buna göre U_i kanonik değişkeni ile Y değişken kümesindeki orijinal değişkenler arasındaki korelasyonlar ise

$$Kor(U_i, Y) = \frac{a_i' \sum_{12}}{\sqrt{Köş(\sum_{22})}} \quad (60)$$

şeklinde hesaplanır [39]. V_i kanonik değişkeni ile X değişken kümesindeki orijinal değişkenler arasındaki korelasyonlar

$$Kor(V_i, X) = \frac{b_i' \sum_{21}}{\sqrt{Köş(\sum_{11})}} \quad (61)$$

şeklinindedir. V_i kanonik değişkeni ile Y değişken kümesindeki orijinal değişkenler arasındaki korelasyonlar;

$$Kor(V_i, Y) = \frac{b_i' \sum_{22}}{\sqrt{Köş(\sum_{22})}} \quad (62)$$

formülü ile hesaplanmaktadır. Böylece elde edilen bir kanonik değişkene herhangi bir orijinal değişkenin katkı miktarı da bulunmuş olur. Ayrıca

$$Kor(U_i, Y) = \rho_i Kor(V_i, Y) \quad (63)$$

$$Kor(V_i, X) = \rho_i Kor(U_i, X) \quad (64)$$

eşitliklerine de ulaşılabilir. Bu sonuçları elde edebilmek için i . adım için

$$-\rho_i \sum_{11} a_i + \sum_{22} b_i = 0 \quad (65)$$

$$-\rho_i \sum_{22} b_i + \sum_{21} a_i = 0 \quad (66)$$

yazılıp; ilk eşitlik soldan \sum_{11}^{-1} ile ikinci eşitlik yine soldan \sum_{22}^{-1} ile çarpılıp düzenlenirse,

$$-\rho_i \sum_{11}^{-1} \sum_{11} a_i + \sum_{11}^{-1} \sum_{12} b_i = 0 \quad \Rightarrow \quad \rho_i a_i = \sum_{11}^{-1} \sum_{12} b_i \quad (67)$$

$$-\rho_i \sum_{22}^{-1} \sum_{22} b_i + \sum_{22}^{-1} \sum_{21} a_i = 0 \quad \Rightarrow \quad \rho_i b_i = \sum_{22}^{-1} \sum_{21} a_i \quad (68)$$

$$a_i = \frac{1}{\rho_i} \sum_{11}^{-1} \sum_{12} b_i \quad \text{ve} \quad b_i = \frac{1}{\rho_i} \sum_{22}^{-1} \sum_{21} a_i \quad (69)$$

sonuçları bulunacaktır. Buna göre

$$Kor(U_i, X) = \frac{a_i' \sum_{21}}{\sqrt{Köş(\sum_{11})}} = \frac{a_i' \sum_{12} \sum_{22}^{-1} \sum_{21}}{\rho_i \sqrt{Köş(\sum_{11})}} = \rho_i Kor(U_i, X) \quad (70)$$

$$Kor(U_i, Y) = \frac{a_i' \sum_{12}}{\sqrt{Köş(\sum_{22})}} = \frac{b_i' \sum_{21} \sum_{11}^{-1} \sum_{12}}{\rho_i \sqrt{Köş(\sum_{22})}} = \rho_i Kor(V_i, Y) \quad (71)$$

eşitlikleri gerçekleşmiş olur. Sonuç olarak kitle kovaryans matrisi Σ kullanılarak kanonik korelasyonların ve kanonik değişkenlerin nasıl elde edildiği gösterilmiştir. Bu bilgiler ışığı altında, deneysel veriler için örnek kovaryans matrisi S ' nin

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} = \frac{1}{n} \begin{bmatrix} \sum (X_1 - \bar{X}_1) (X_1 - \bar{X}_1)' & \sum (X_1 - \bar{X}_1) (X_2 - \bar{X}_2)' \\ \sum (X_2 - \bar{X}_2) (X_1 - \bar{X}_1)' & \sum (X_2 - \bar{X}_2) (X_2 - \bar{X}_2)' \end{bmatrix} \quad (72)$$

şeklinde olduğu görülecektir [49].

1.8. Çekirdek Öznitelik Uzayı

Doğrusal öğrenme makinelerinin sınırlı hesaplama gücü 1960'larda Minsky ve Papert [50] tarafından ortaya atılmıştır. Genelde, karmaşık sahici uygulamalar doğrusal fonksiyonlardan daha anlamlı hipotez uzaylarına ihtiyaç duyarlar. Bu problemi görmenin diğer bir yolu ise: verilen niteliklerin basit doğrusal kombinasyonu ile ifade edilemez, fakat genelde verilerden yararlanılmış daha fazla soyut öznitelik ihtiyacı duyulmasıdır. Eşiklenmiş doğrusal fonksiyonların çoklu katmanları bu probleme çözüm olarak önerilmiştir. Bu yaklaşım, sistem için çoklu-katmanlı sinir ağları ve geri yayımlı öğrenme algoritmaları gibi çeşitli yöntemlerin geliştirilmesine olanak sağlamıştır.

Çekirdek gösterimler, doğrusal öğrenme makinelerinin hesaplama yeteneğini arttırmak için verileri yüksek boyutlu bir öznitelik uzayına eşleyerek alternatif çözümler sunarlar. Doğrusal makinelerin ikili gösterimi bu adımı dolaylı olarak gerçekleştirmemize olanak sağlar. Eğitim örnekleri hiç bir zaman tek başına bırakılmış görünmez, her zaman örnek çiftleri arasında iç çarpım biçiminde görünmektedir. Doğrusal makineleri ikili gösterimiyle kullanılmasının avantajı bu gösterimde ayarlanabilir parametre sayısının kullanılan özellik sayısına bağımlı olmamasından kaynaklanmaktadır. İç çarpımı uygun bir şekilde çekirdek fonksiyonu ile değiştirerek, ayarlanabilir parametre sayısını arttırmadan yüksek boyutlu öznitelik uzayına doğrusal olmayan bir eşleme dolaylı olarak gerçekleştirilebilir. Burada çekirdek, iki girişe karşı düşen öznitelik vektörlerinin iç çarpımını gerçekleştirmiş olur. [51].

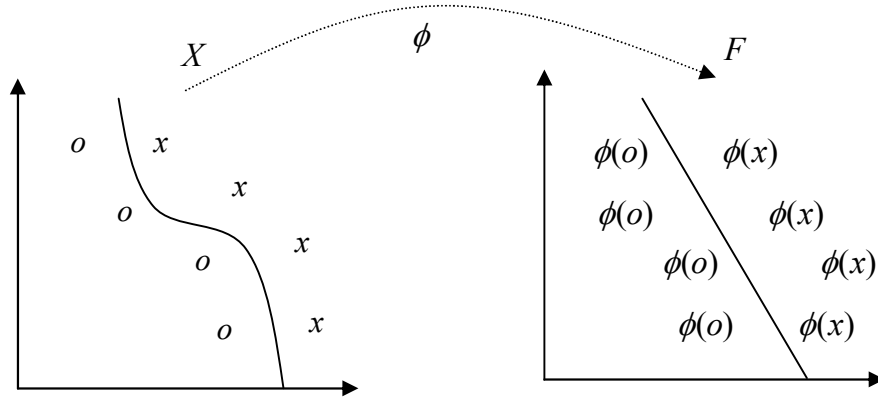
1.8.1. Öznitelik Uzayında Öğrenme

Öğrenilecek hedef fonksiyonun karmaşıklığı temsil edilmesine bağlıdır ve öğrenme işinin zorluğu da bu sebepten dolayı değişiklik gösterebilir. Normalde belirli bir öğrenme problemi için eşleyen gösterim seçilmelidir. Bu yüzden, otomatik öğrenmede yaygın kullanılan bir ön işlem stratejisi veri gösteriminin değiştirilmesine bağlıdır:

$$x = (x_1, \dots, x_n) \mapsto \phi(x) = (\phi_1(x), \dots, \phi_N(x)) \quad (73)$$

Burada X giriş uzayını yeni bir uzaya ($F = \{\phi(x) | x \in X\}$) eşlemeye eşdeğerdir. Sadece verileri diğer bir uzaya eşleme uzun zamandır otomatik öğrenmede bilinen ve birtakım tekniklerin ortaya çıkmasına sebep olan en iyi temsil verisini seçme işini oldukça kolaylaştırır. En uygun gösterimdeki seçme işi, öznelik seçimi olarak bilinir. X uzayı giriş uzayını gösterirken, $F = \{\phi(x) | x \in X\}$ uzayı öznelik uzayı olarak isimlendirilir [51].

Şekil 7, iki boyutlu giriş uzayından iki boyutlu öznelik uzayına bir öznelik eşleme örneği gösterilmektedir. Bu örnekte veriler doğrusal bir fonksiyonla giriş uzayında ayrılamazlarken öznelik uzayında ayrılabilir hale geldikleri görülmektedir.



Şekil 7. Öznelik eşlemede sınıflandırma işlemi

Öznelik seçimi için değişik yaklaşımlar mevcuttur. Orijinal niteliklerde içerilen temel bilgiyi taşıyan en küçük öznelik kümesini tanımlamak için boyut azaltımı olarak bilinen denklem (74) kullanılır.

$$x = (x_1, \dots, x_n) \mapsto \phi(x) = (\phi_1(x), \dots, \phi_d(x)), \quad d < n \quad (74)$$

Bu, bazen öznelik sayısı arttığında hem hesaplama hem de genelleştirme performansının düşmesi açısından çok yararlı olabilmektedir. Yüksek boyutlu öznelik uzayındaki zorluklar, öznelik kümesi geniş oldukça öğrenilecek fonksiyon sayısı da azalmış olacağından bazen gereksiz olabilir.

Diğer bir yöntem öznitelik seçme işlemi için çıkış değerini etkilemeyen ilgisiz özniteliklerin belirlenmesi ve çıkarılmasıdır.

Temel bileşen analizini kullanma, verilerin öznitelik uzayına eşlenmesini sağlar. Bu öznitelik uzayında yeni öznitelikler orijinal niteliklerin doğrusal fonksiyonudurlar ve verilerin her bir yönde sergilediği varyans miktarına göre dizilirler. Boyut azaltımı bazen, sadece verilerin düşük varyansa sahip olduğu yönlere karşı düşen özniteliklerin (her ne kadar bu özniteliklerin hedef sınıflandırmayı gerçekleştirmede gerekli olmadığını garanti edilememesine rağmen) kaldırılması ile gerçekleştirilir.

Burada, ek öznitelik boyutlarının da yararlı olabileceği iki boyutlu giriş uzayı durumunu ele almada fayda vardır. Herhangi bir problem hakkındaki, ilgili bilgilerin 2.dereceden tek terimliler biçiminde kodlandığının tahmini varsayılır. Bu yüzden problemi bu tür bilgilerin açık olarak verildiği ve öğrenme makinesinin kullanımı için hazır olduğu öznitelik uzayında göstermek istenildiğinde olası bir eşleme şu şekilde olacaktır:

$$(x_1, x_2) \mapsto \phi(x_1, x_2) = (x_1^2, x_2^2, x_1x_2) \quad (75)$$

Aynı şekilde, $\binom{n+d-1}{d}$ boyutunda öznitelik uzayına sebep olacak d dereceli özniteliklerin kullanılması da istenebilir. Bu durumda yeterli sayıda nitelik ve öznitelik derecesi için hesaplama uygunsuz hale gelmektedir. Bu tür öznitelik uzayını kullanmak için öznitelik uzayına dolaylı eşlemeyi içeren özel bir tekniğe ihtiyacımız olacaktır.

Hesaba dayalı problemler sadece kullandığımız öznitelik uzayının boyutu ile bağlantılı değildir. Diğer bir zorluk kaynağı, hipotezlerin standart fonksiyon sınıfları için gösterim boyutuna duyarlı olabilen öğrenme makinelerinin geliştirilmesidir.

Öznitelik seçimi öğrenme sürecinin bir parçasıdır ve mümkün olduğunca otomatikleştirilmiş olmalıdır. Diğer yandan bu, altta yatan hedef fonksiyonu hakkındaki tahminleri yansıtan bir adımdır. Öğrenmenin teorik modelleri bu adımı hesaba katmalıdır. Genelleştirme bir şekilde kontrol altına alınamazsa çok büyük öznitelik kümesi oluşturmak aşırı yüklemeye problemlerine yol açabilir [51].

1.8.2. Öznitelik Uzayına Dolaylı Eşleme

Doğrusal olmayan ilişkileri doğrusal bir makine ile öğrenmek için bir grup doğrusal olmayan öznitelik seçmemiz ve yeni gösterimde verileri yeniden düzenlememiz gerekmektedir. Bu, verilerin doğrusal makine tarafından kullanılabilir ve öznitelik uzayına sabit doğrusal olmayan bir biçimde eşlenecektir. Bundan dolayı ele alacağımız hipotezler kümesinin fonksiyon gösterimi:

$$f(x) = \sum_{i=1}^N w_i \phi_i(x) + b \quad (76)$$

biçiminde olacaktır.

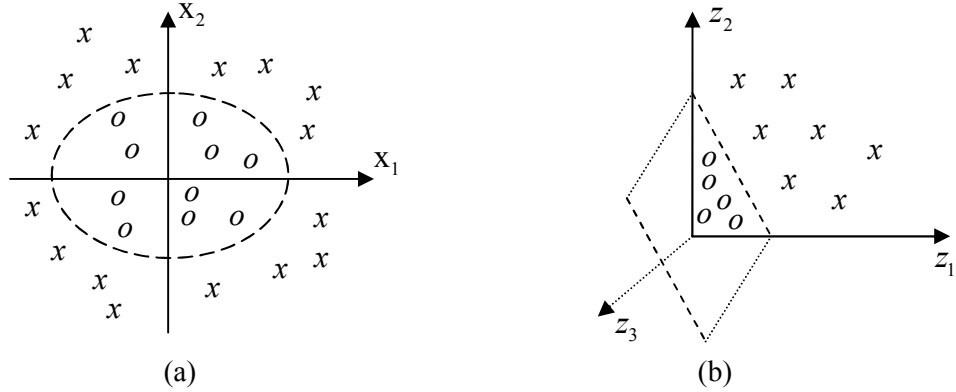
Burada $\phi: X \rightarrow F$, giriş uzayından öznitelik uzayına doğrusal olmayan eşlemedir. Bu, doğrusal olmayan makineleri iki adımda inşa edecek olmamız demektir: önce sabit doğrusal olmayan eşleme, verileri bir F öznitelik uzayına dönüştürür ve ardından bir doğrusal makine onları öznitelik uzayında sınıflandırmak için kullanır.

Doğrusal öğrenme makinelerinin önemli bir özelliği ikili gösterimle ifade edilebiliyor olmalarıdır. Bu aynı zamanda, hipotezler karar kuralının sadece test ve eğitim noktaları arasındaki iç çarpımını kullanarak hesaplanabileceğinden eğitim noktalarının doğrusal kombinasyonu olarak da ifade edilebilir.

$$f(x) = \sum_{i=1}^{\ell} \alpha_i y_i \langle \phi(x_i) \cdot \phi(x) \rangle + b \quad (77)$$

Eğer $\langle \phi(x_i) \cdot \phi(x) \rangle$ iç çarpımını öznitelik uzayında orijinal giriş noktalarının bir fonksiyonu olarak doğrudan hesaplayabiliyorsak doğrusal olmayan öğrenme makinesini inşa etmek için gerekli iki adımı birleştirmemiz mümkün olacaktır. Dolayısı ile bu şekilde doğrudan hesaplama yöntemine çekirdek fonksiyon denilmektedir [51, 52].

Şekil 8'de gösterilen örnekte ayrılamayan problem, giriş verilerinin ϕ_2 ile önişlemeden geçirilmesi sonucu ayırma hiper düzleminin oluşturulmasına dönüştürülmüştür. Bunun hesaplama bakış açısından (hiper düzlemi hesaplamak için etkin bir algoritmanın olması) ve istatistiksel bakış açısından (hiper düzlemin test noktaları için iyi genelleştirileceğinin garanti edilmesi) bakıldığında yararları görülecektir.



Şekil 8. Öznitelik uzayına eşlenmiş ikili sınıflandırma problemi örneği. (a) Elips düzlem (b) Hiper düzlem

Şekil 8-a'da giriş uzayında uygun karar sınırının elips olduğunu varsayılmaktadır. Öğrenme süreci, bu sınırın her iki sınıfa ait eğitim noktalarını içeren gözlemsel verilere dayanılarak tahmin edilmesi sürecidir. Doğrusal olmayan $\phi_2(x) = (z_1, z_2, z_3) = (|x|_1^2, |x|_2^2, \sqrt{2}|x|_1|x|_2)$ eşlemesi ile öznitelik uzayına eşlendiğinde elips düzlem bir hiper düzlem (Şekil 8-b) halini alır. Bu, elipslerin (z_1, z_2, z_3) girişlerine sahip doğrusal eşitlikler olarak yazılabileceğinden dolayı ortaya çıkar. Bu yüzden, öznitelik uzayında problem, eşlenen veri noktalarından hiper düzlemin tahminine indirgenmesini sağlamaktadır. Polinomsal çekirdeklerle üç boyutlu uzaydaki iç çarpım ϕ_2 'yi hesaplamadan da bulunabilir.

Çoğu gerçekçi durumda, örneğin x , elemanları piksel değerleri olan bir görüntüyü temsil ederse, polinomsal çekirdekler $\langle x, x' \rangle^d$ bizim herhangi d piksel değer çarpımı ile uzanan uzayda çalışmamıza imkan tanır. Çalışmamızı eşlenmiş örüntünün herhangi bir açık kullanımını gerektirmeden iç çarpımlarla gerçekleştirmemizi sağlar. Çekirdekleri kullanmakla, zaman ve bellek karmaşıklığına neden olmadan yüksek dereceden istatistiği hesaba katabiliriz [51].

Çekirdek, bütün $x, z \in X$ için denklem (78) koşulunu sağlayan bir fonksiyondur.

$$K(x, z) = \langle \phi(x) \cdot \phi(z) \rangle \quad (78)$$

Burada ϕ , X 'den F öznitelik uzayına eşlemedir.

Çekirdek ismi, çekirdekler ve onların öznitelik uzayları arasındaki ilişki teorilerinin birçoğunun temelini oluşturan integral operatörü teorisinden çıkarılmıştır. Öznitelik vektörlerinin açıkça gösterilmesine gerek olmadığı gibi, çekirdek fonksiyonu ile iç çarpımı hesaplamak için ihtiyaç duyulan işlem sayısı kesin olarak öznitelik sayısı ile orantılı değildir. Çekirdeklerin kullanılması verilerin öznitelik uzayına dolaylı olarak eşlenmesine ve doğrusal makinenin böyle bir uzayda eğitilmesine olanak sağlar. Eğitim örnekleri hakkında kullanılan tek bilgi onların öznitelik uzayındaki Gram matrisidir. Bu matris ayrıca çekirdek matris olarak da adlandırılır ve bu bağlamda onu göstermek için K sembolünü kullanılmaktadır. Bu yaklaşımın anahtar kısmı, etkin bir şekilde hesaplanabilen çekirdek fonksiyonunu bulmaktır. Böyle bir fonksiyona sahip olduğumuzda karar kuralı en fazla çekirdeğin ℓ hesaplaması ile belirlenebilir:

$$f(x) = \sum_{i=1}^{\ell} \alpha_i y_i K(x_i, x) + b \quad (79)$$

Çekirdek kullanmanın ilginç olgularından biri, öznitelik uzayında öğrenmeyi gerçekleştirebilmek için altta yatan öznitelik eşlemeyi bilmemize gerek olmayışıdır.

Çekirdek kavramı doğrudan doğruya sezgisel bir kavram değildir. Öncelikle, çekirdek fikri giriş uzayındaki standart iç çarpımı genelleştirir. Bu iç çarpımın, öznitelik eşlemenin özdeş yapılmasıyla çekirdeğe bir örnek teşkil ettiği açıktır.

$$K(x, z) = \langle x \cdot z \rangle \quad (80)$$

Ayrıca öznitelik eşlemeyi herhangi bir x (A matrisi için Ax) sabit doğrusal dönüşümü olarak ele alabiliriz. Bu durumda çekirdek fonksiyonu denklem (81)'deki gibi ifade edilir.

$$K(x, z) = \langle Ax \cdot Az \rangle = x' A' A z = x' B z \quad (81)$$

Burada $B = A'A$ yarı tanımlı pozitif kare matrisi belirtir. Amacımız doğrusalsızlığı öznitelik uzay eşlemesi ile ortaya koymaktır. Bu yüzden aşağıdaki ilişkiyi dikkate alarak elde edilen doğrusal olmayan eşlemeyi basitçe inceleyelim:

$$\begin{aligned} \langle x \cdot z \rangle^2 &= \left(\sum_{i=1}^n x_i z_i \right)^2 = \left(\sum_{i=1}^n x_i z_i \right) \left(\sum_{j=1}^n x_j z_j \right) \\ &= \sum_{i=1}^n \sum_{j=1}^n x_i x_j z_i z_j = \sum_{(i,j)=(1,1)}^{(n,n)} (x_i x_j) (z_i z_j) \end{aligned} \quad (82)$$

Buradan çekirdek işleminin öznitelik vektörleri arasındaki iç çarpıma eşdeğer olduğu söylenebilir.

$$\langle x \cdot z \rangle^2 = \langle \phi(x) \cdot \phi(z) \rangle, \quad \phi(x) = \left(x_i x_j \right)_{(i,j)=(1,1)}^{(n,n)} \quad (83)$$

Bu durumda özniteliklerin hepsi 2.dereceden tek terimlidir. Fakat şunu da not etmek gerekir ki $i \neq j$ olduğu durumlarda $x_i x_j$ özniteliği iki defa gerçekleştiğinden ağırlığı x_i^2 özniteliğinin ağırlığının iki katı olmaktadır. Daha genel olarak öznitelik uzayı, denklem (84)'deki gibi çekirdeğin hesaba katılmasıyla elde edilir.

$$\begin{aligned} (\langle x \cdot z \rangle + c)^2 &= \left(\sum_{i=1}^n x_i z_i + c \right) \left(\sum_{j=1}^n x_j z_j + c \right) \\ &= \sum_{i=1}^n \sum_{j=1}^n x_i x_j z_i z_j + 2c \sum_{i=1}^n x_i z_i + c^2 \\ &= \sum_{(i,j)=(1,1)}^{(n,n)} (x_i x_j) (z_i z_j) + \sum_{i=1}^n (\sqrt{2c} x_i) (\sqrt{2c} z_i) + c^2 \end{aligned} \quad (84)$$

Burada $\binom{n+1}{2} + n + 1 = \binom{n+2}{2}$ özneliğın hepsi 2'ye kadar olan derecelere sahip tek

terimlidir. Fakat 1 ve 2 dereceliler arasındaki göreceli ağırlıklandırmalar ile öznelikler 0.derecenin kuvvetini belirleyen sabit öznelik c parametresi aracılığıyla kontrol edilmesidir. Benzer türetilmeler çekirdek fonksiyonlar için de yapılabilir.

$$K(x, z) = \langle x \cdot z \rangle^d \text{ ve } K(x, z) = (\langle x \cdot z \rangle + c)^d, d \geq 2 \text{ için} \quad (85)$$

İlk çekirdek $\binom{n+d-1}{d}$ için ayrı özneliğın hepsi, ağırlıkları üstelin yapısına bağılı

olarak değışecek olmasına rağmen d derecelidirler. İkinci çekirdek için bütün tek terimlerin d ve d 'ye kadar olan derecelere sahip olduğı $\binom{n+d}{d}$ ayrı öznelik vardır.

Bu öznelik uzaylarında bir hiper düzleme karşı düşen giriş uzayındaki karar sınırı d dereceli polinomsal eğrilerdir. Bu yüzden çekirdekler çoğunlukla polinomsal çekirdekler olarak adlandırılmaktadır [51].

1.8.3. Çekirdeklerin Oluşturulması

Çekirdek fonksiyonunun kullanımı hesaba dayalı bir kısa yoldur. Eğer bu yaklaşım kullanılmak istenirse önce karmaşıklaştırılmış öznelik uzayının oluşturulması, sonra bu uzayda iç çarpımın ne olacağıın çözülmesi ve son olarak da o değeri orijinal girişler cinsinden hesaplamak için direkt bir yöntemin bulunması gerekmektedir. Pratikte anlatılan bu yaklaşım, doğrudan bir çekirdek fonksiyonu tanımlamaktır. Bu yüzden dolaylı olarak öznelik uzayı tanımlanır. Bu yolla, öznelik uzayından sadece iç çarpımların hesaplanmasında değil aynı zamanda öğrenme makinelerinin tasarımından da kaçınılmış olunur. Giriş uzayı için çekirdek fonksiyonu tanımlamanın çoğunlukla karmaşıklaştırılmış öznelik uzayı oluşturmaktan daha doğal olduğı iddia edilir. Bu yolu izlemeden önce ilk olarak $K(x, z)$ fonksiyonunun bir öznelik uzayı için çekirdek olduğundan emin olmak amacıyla hangi özelliklerinin gerekli olduğunu belirlememiz gerekiyor. Açık bir biçimde, fonksiyon simetrik olmalı,

$$K(x, z) = \langle \phi(x) \bullet \phi(z) \rangle = \langle \phi(z) \bullet \phi(x) \rangle = K(z, x) \quad (86)$$

ve Cauchy-Schwarz eşitsizliğinin sonucu olan aşağıdaki eşitsizliğin sağlanması gerekmektedir.

$$\begin{aligned} K(x, z)^2 &= \langle \phi(x) \bullet \phi(z) \rangle^2 \leq \|\phi(x)\|^2 \|\phi(z)\|^2 \\ &= \langle \phi(x) \bullet \phi(x) \rangle \langle \phi(z) \bullet \phi(z) \rangle = K(x, x)K(z, z) \end{aligned} \quad (87)$$

Bununla birlikte bu koşullar öznitelik uzayının var olduğunun garanti edilmesi için yeterli olmayabilir [51].

1.8.4. Çekirdek Hilesi

Her zaman mevcut veri seti doğrusal ayırlamayacaktır, dolayısı ile bu durumlarda verileri daha yüksek boyutlu bir özellik uzayına aktarmamız gerektiğinden bahsetmiştik. Fakat bu aktarma (gömme) işlemi bazen maliyeti çok yükseltebilir. Çekirdek hilesinin amacı, oluşan maliyetin minimize edilmesi anlamında sisteme büyük bir kazanç sağlamaktır. \mathbb{R}^3 uzayında doğrusal sınıflandırılmayan bir girdi kümesinin daha yüksek boyutlu bir uzaya gömülmesi işleminde maliyetin azaltılması için çekirdek hilesini bilmekte fayda vardır. Verilerimiz \mathbb{R}^3 uzayında olduğunu varsayarsak matematiksel olarak çekirdek hilesini bir noktayı örnekleyerek açıklayalım. Veri setinden bir noktayı ele alalım ve $x_i = (1, 2, 3)$ olsun. Bu noktanın koordinatlarını bir satır vektörü olarak ele alıp, doğrudan dokuz boyutlu bir uzaya gömme işlemini denklem (88) ile tanımlayalım.

$$\varphi(x) = x^T x \quad (88)$$

Buradan,

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \cdot [1 \ 2 \ 3] = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{bmatrix} \quad (89)$$

denklemleri elde edilir ki oluşan sonucu da denklem (90)'daki gibi yorumlayabiliriz.

$$\varphi(x) = [1 \ 2 \ 3] \rightarrow [1 \ 2 \ 3 \ 2 \ 4 \ 6 \ 3 \ 6 \ 9] \quad (90)$$

Veri noktasının iç çarpım fonksiyonu altındaki görüntüsü denklem (91)'deki gibi hesaplanabilir.

$$\kappa(x, x) = \langle x, x \rangle \quad (91)$$

$$\kappa(x, x) = 1.1 + 2.2 + 3.3 = 14 \quad (92)$$

Fakat bu uzayda doğrusal sınıflandırma mümkün olmadığından bu iç çarpım sonucu denklem (92)'deki gibi sınıflandırmaya yaramayacaktır. Şimdi ise elde edilen yeni veri noktasının iç çarpım fonksiyonu altındaki görüntüsünü hesaplayalım [53].

$$\kappa(x, x) = \langle \varphi(x), \varphi(x) \rangle \quad (93)$$

$$\langle \varphi(x), \varphi(x) \rangle = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 + 8^2 + 9^2 = 196 \quad (94)$$

Girdi verisi üç boyutlu bir uzayda olduğundaki maliyet denklemi (92)'de ifade edilirken, dokuz boyuta gömüldükten sonraki maliyet denklemi (94)'deki görüldüğü gibidir ve arasında oluşan fark aşikârdır. Bu sebeple \mathbb{R}^3 de doğrusal sınıflandırılmayan verileri \mathbb{R}^9 da doğrusal sınıflandırabilirsek bile maliyetin yüksek olması sistemin kullanılabilirliğini sınırlamaktadır.

Çekirdek hilesi ile;

$$\langle x, x \rangle^2 = (1^2 + 2^2 + 3^2)^2 = 196 \quad (95)$$

iç çarpım fonksiyonu denklem (95) ile hesaplanıp aynı sonuca ulaşılabilir. Çekirdeklerin sonucu aynı olmasına rağmen maliyetin hala \mathbb{R}^3 de uygulanan iç çarpım kadar olduğu görülür [54,55].

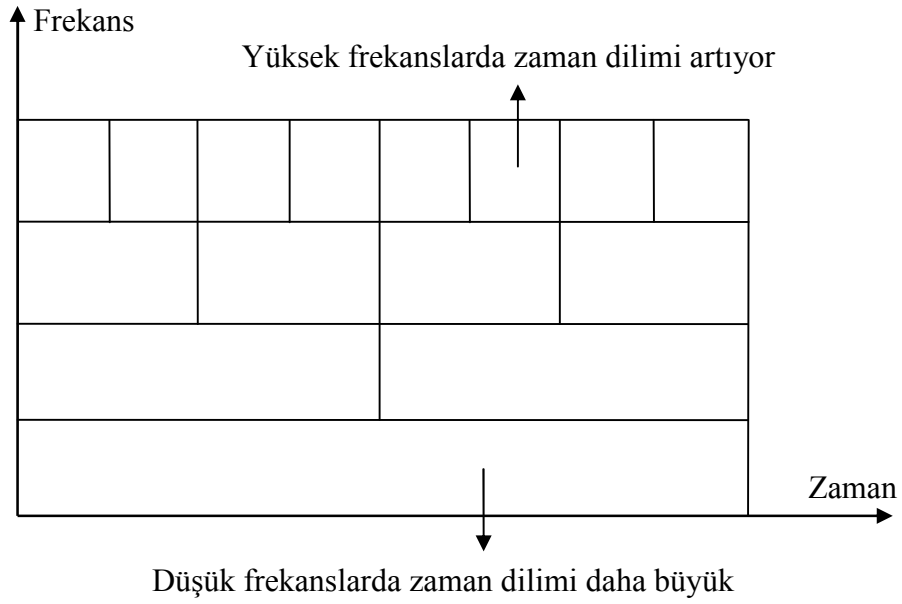
1.9. Dalgacık Dönüşümü

Dalgacıklar, ilk defa Alman matematikçi Alfred Haar [56] tarafından ortaya atılmıştır. Teorik olarak ilk kez, Morlet ve Grossmann [57] idaresindeki bilim adamları tarafından ele alınmıştır. Fakat ana algoritma, Mallat'ın [6] çalışmalarında ortaya çıkmıştır.

Tarihsel yönden dalgacık dönüşümü yeni bir işaret işleme yöntemi olmasına karşılık temeli Joseph Fourier'e kadar uzanmaktadır. Fourier, karmaşık işaretlerin daha basit işaretlerin toplamı şeklinde yaklaşık olarak ifade edilebileceğini belirterek, çok etkin ve çok önemli olduğu ispatlanan frekans dönüşümü ile ilgili düşüncelerini temel olarak ortaya koymuştur. Fourier dönüşümüyle verilen işaretler için frekans bölgesinde analizler yapılabilirken, zaman bölgesine sonsuz uzanan fonksiyonlar için sinyalin içeriği hakkında herhangi bir sayısal bilgi belirlenemediğinden araştırmacıların ilgisi, frekans tabanlı analizden zaman-frekans tabanlı analize zamanla kaymaya başlamıştır [58,59].

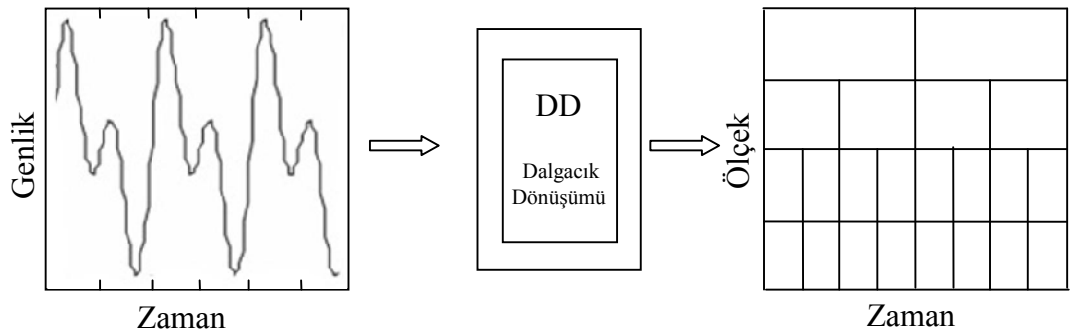
Fourier Dönüşümü (FD), işaretteki herhangi bir anlık değişimi bütün frekans eksenine yansıtır. Bu ise işaretin dinamiklerini yakalamak açısından istenmeyen bir durumdur. Bu eksikliği gidermek için Dennis Gabor , FD'yi zamanın belirli bir bölümünde işaretin bir kısmını incelemek için kullanarak Kısa-Zamanlı Fourier Dönüşümünü (KZFD) geliştirdi. Ancak KZFD de zaman domeninde uygulanan pencerenin uzunluğunun sabit olması, tüm frekanslar için pencerenin aynı kalmasına neden olduğu için hangi zamanda hangi frekans bileşenlerinin ortaya çıkacağı belirlenememektedir. KZFD'deki bu eksikliği gidermek içinde değişen uzunluktaki pencereleri kapsayan pencereleri içeren Dalgacık Dönüşümü (DD) yeni bir işaret işleme yöntemi olarak ortaya çıkmıştır [60].

Belirli bir zaman penceresinde değeri sıfırdan farklı özel salınımsal fonksiyona dalgacık adı verilmektedir. İncelenen işaret dalgacık fonksiyonuyla çarpılarak, her bir kısım için dönüşüm yapılmaktadır. DD'nin en önemli özelliği, Şekil 9'de görüldüğü gibi düşük frekans bilgilerinin istendiği yerde geniş zaman aralığında, yüksek frekans bilgilerinin istendiği yerde ise küçük zaman aralığında işlem yapmasıdır [61].

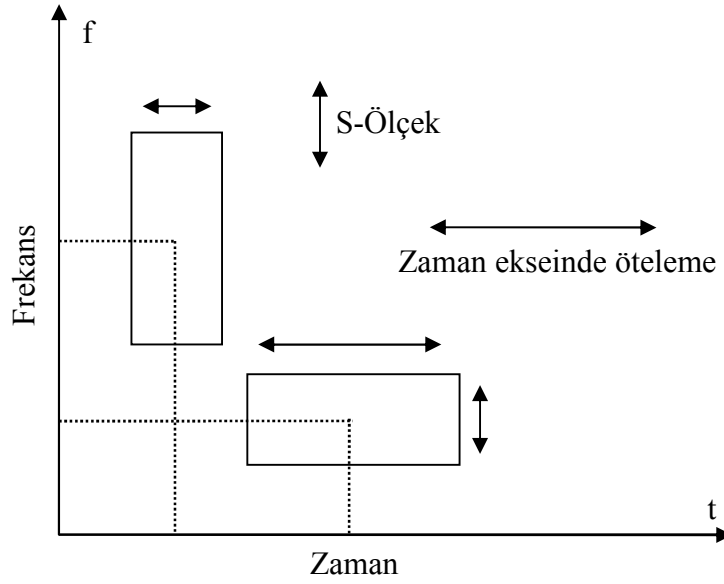


Şekil 9. Dalgacık Dönüşümü zaman-frekans gösterimi

DD, Fourier dönüşümüyle incelenen durağan işaretlerin yanı sıra durağan olmayan işaretlerin incelenmesinde de kullanılan bir işaret işleme yöntemidir (Şekil 10). Dalgacık dönüşümünde, verilen bir işaret için düşük ve yüksek frekanslarda farklı boyutlarda pencereler kullanılarak işaretin zaman düzleminde temsili oluşturulur (Şekil 11) ve diğer zaman-frekans yöntemleriyle belirlenemeyen özellikleri görülebilir [61].



Şekil 10. Dalgacık Dönüşümü



Şekil 11. Zaman-Frekans gösterimi

DD'de işarete bakmak için kullanılan ölçek dalgacık analizinde özel bir rol oynar. KZFD' nin sınırlamalarından biri olan, kullanılan pencerenin sabit olması DD'de ölçeklenebilir bir pencere ile giderilmiştir. Böylece işaret içindeki düşük frekans eğilimlerini açmak için geniş bir pencere, yüksek frekans detaylarını analiz etmek için sıkıştırılmış bir pencere kullanılır. Bunun için, DD ölçeklenebilir temel bir dalgacık fonksiyonu kullanıp sabit çözünürlük problemine çözüm getirerek, işaretin farklı çözünürlüklerde daha esnek bir zaman bölgesi analizini yapar [59].

1.9.1. Sürekli Dalgacık Dönüşümü

Sürekli Dalgacık Dönüşümü (SDD), orijinal işaretin bütün zamanlarda ölçeklenmiş, kaydırılmış dalgacık ile çarpımlarının toplamıdır. SDD'nin sonucunda birçok dalgacık katsayısı oluşur. Bunlar ölçek ve konumun bir fonksiyonudur. Her bir katsayıyı uygun kaydırılmış ve ölçeklenmiş dalgacıklarla çarparak yeniden orijinal işarete ulaşılabilir [6].

$$SDD(s, b) = - \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \psi\left(\frac{t-b}{s}\right) dt \quad (96)$$

Denklemdaki $f(t)$, orijinal işareti, $\psi(t)$ dalgacık fonksiyonunu, b öteleme (zaman) parametresi olup, farklı frekans seviyelerinde ayrışım filtreleri tanımlar. s ise ölçekleme (frekans) parametresi olup her seviye için ayrışım filtrelerini ölçeklendirir. Bir dalgacığın ölçeklendirilmesi ifadesi, basitçe işaretin genişletilmesi veya daraltılması anlamına gelmektedir [61].

1.9.2. Ayrık Dalgacık Dönüşümü (ADD)

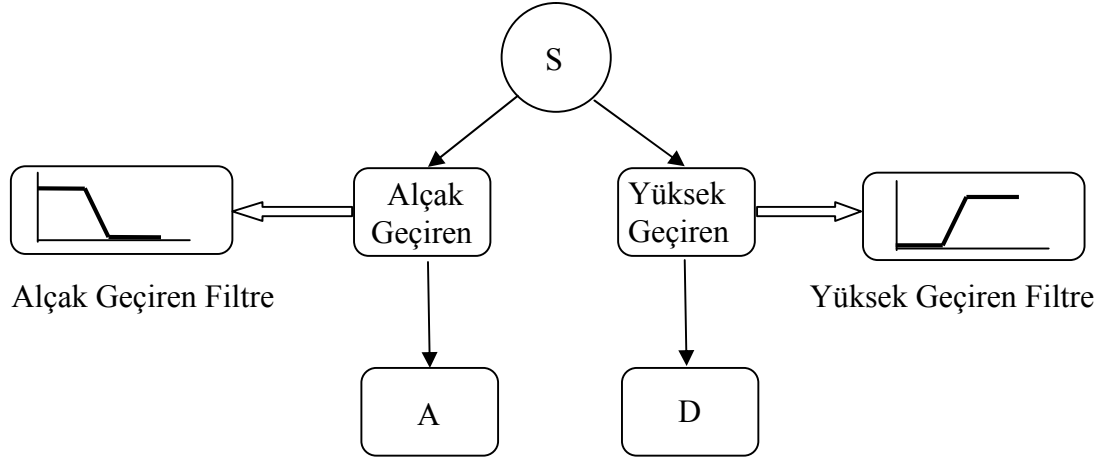
Bütün dalgacık katsayılarını, her ölçek için hesaplama, oldukça büyük miktarda bir iş yüküne ve gereksiz bol miktarda da bilgiye sebep olacaktır. Eğer ölçek ve konum, ikinin katları şeklinde seçilirse yöntem önceki kadar hatasız ve çok daha efektif hale gelecektir. Böyle bir analizi ADD sağlar. Bu yöntemin efektif bir şekilde uygulanmasını, ilk kez 1988 yılında Mallat sağlamıştır. Mallat'ın algoritması, aslında işaret işleyicilerin iki kanal “yan band kodlayıcı” olarak adlandırdığı klasik bir plandır. Bu oldukça hızlı bir dalgacık dönüşüm işlemidir. Bir tarafta bir kutuya giren işaretler öbür taraftan kolaylıkla katsayılarıyla çıkar [62]. ADD denklem (97) ile ifade edilir.

$$ADD(s, n) = 2^{-s/2} \int f(t) \psi(2^{-s} t - n) dt \quad (97)$$

Burada s parametresi ölçeklemeyi (frekansı), n parametresi ise ötelemeyi (zamanı) belirler. Bu işlem, DD'yi hızlandıran çok pratik bir filtre algoritmasıdır [63].

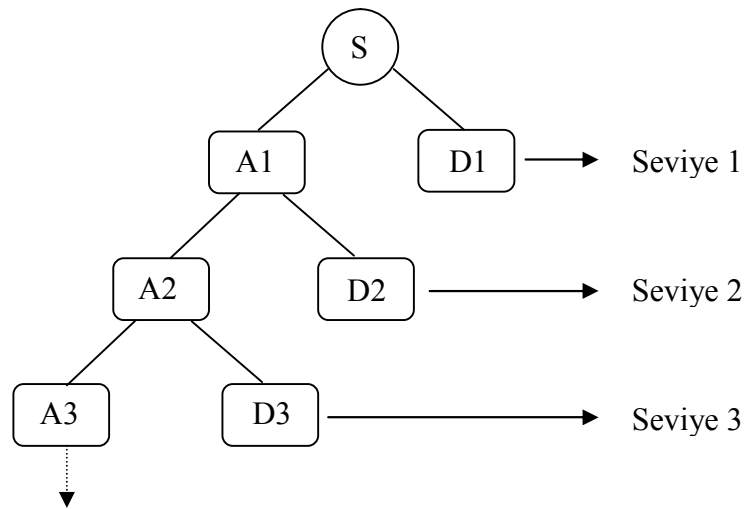
Birçok işaret için, işaretin düşük frekanslı içeriği en önemli kısmıdır. Düşük frekans bileşenleri, işaretler için genellikle tanınma özelliklerini içerir. Diğer taraftan yüksek frekanslı bileşenler ise işaretin kendisiyle düşük frekanslı kısım arasındaki farkı oluştururlar. Örnek olarak insan sesi ele alınacak olursa, sesin yüksek frekanslı bileşenlerinin çıkartılması durumunda ses farklılaşır, fakat hala söylenen kelimeler duyulabilir. Oysa düşük frekanslı bileşenlerin bir kısmı çıkarılacak olursa söylenen sözler duyulamaz. Yukarıda anlatılan kavramlardan yola çıkılarak, ADD için yaklaşık ve detay katsayılarından bahsedilir. DD kullanılan yaklaşıklar, yüksek ölçekli ve düşük frekanslı bileşenler, detaylar ise düşük ölçekli ve yüksek frekanslı bileşenlerdir. Sonuç olarak, incelenecek bir S işareti Şekil 12'da görüldüğü gibi birbirini tümleyen alçak ve yüksek

geçiren filtrelerden geçirilerek alçak olarak A ve yüksek olarak D frekanslı bileşenlerine ayrıştırılır [59].



Şekil 12. Ayrık dalgacık dönüşümünün tek seviyeli filtre algoritması

Şekil 13'de ise işaretin çeşitli kademelerde ayrıştırılması gösterilmiştir. Ayrıştırma işlemi tekrarlanarak sürdürülebilir. Böylelikle bir işaret birçok düşük çözünürlüklü bileşene ayrılır. Bu işleme “dalgacık ayrıştırma ağacı” denir.



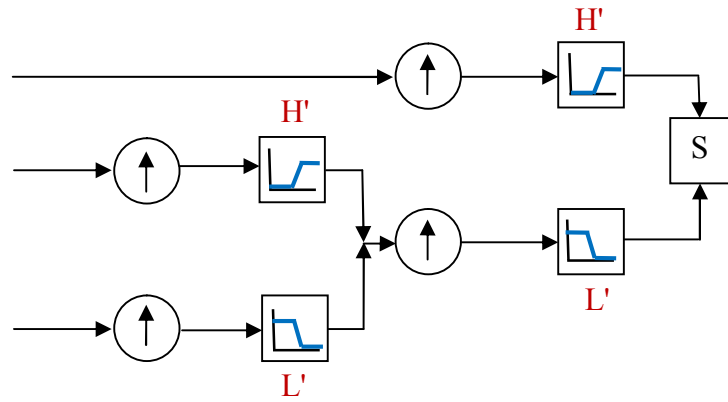
Şekil 13. İşaretin yaklaşık ve detay bileşenlerine ayrıştırılması

Bu ayrıştırma işleminde ayrıştırmanın tekrarlamalı olmasından dolayı, teoride bu işlem sınırsız olarak devam ettirilebilir. Fakat gerçekte ayrıştırma ancak, detaylar bir tek örneğe veya bir tek piksele denk düşene kadar devam ettirilebilir. Pratikte, işaretin yapısına veya entropi gibi bazı kriterlere uygun bir seviye sayısı seçilmelidir [64].

1.9.3. Ters Dalgacık Dönüşümü

Şu ana kadar, işaretin nasıl ayrıştırılacağı izah edildi. Bu kısımda ise, dalgacık dönüşümüyle gelen bileşenlerden orijinal işaretin veri kaybetmeden yeniden nasıl elde edileceği anlatılmaktadır.

Dalgacık analizi için filtreleme ve aşağı örnekleme yapıyordu. Burada ise ters dönüşümde, “yukarı örnekleme” ve filtreleme kullanılır. Yukarı örnekleme, örnekler arasına sıfır eklenmesi ile yapılır (Şekil 14) [65].

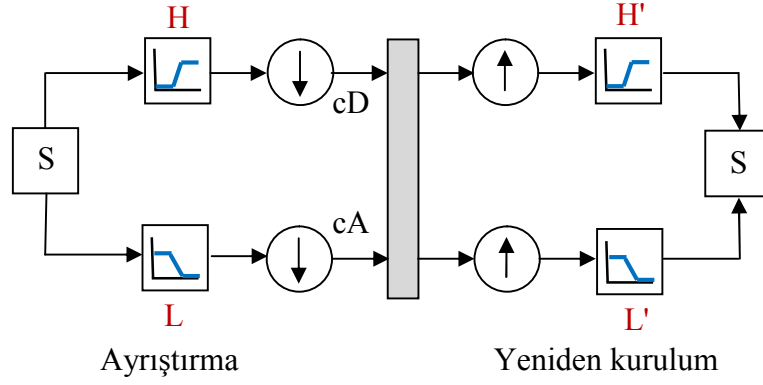


Şekil 14. Ters dalgacık dönüşüm şeması

1.9.4. Ters Dönüşüm Filtreleme

Ters dönüşüm filtreleri, orijinal işaretin seçiminde uygun filtrenin seçiminin önemli olmasından ötürü tartışma konusu olmuştur. Mükemmel bir ters dönüşüm olmayacağı muhakkaktır.

Dönüşüm filtreleri (L ve H), ters dönüşüm filtreleriyle (L' ve H') karesel ayna olarak anılır (Şekil 15) [65].



Şekil 15. Ters dönüşüm filtreleme

1.10. Doğrusal ve Doğrusal Olmayan Karıştırma Yöntemleri

Kaynak işaretleri karıştırma matrisleri ile karıştırırken iki yöntem kullanılır. Bunlardan birincisi doğrusal karıştırma yöntemi olup denklem (98)'de gösterilen kaynak işaretlerin (S), A sabit değer ile çarpılması sonucu karışım matrisini oluşturur. Burada A doğrusal karıştırma matrisini, X ise doğrusal yöntemle karışmış karışım matrisini göstermektedir.

$$X = AS \quad (98)$$

İkinci yöntem olan doğrusal olmayan karıştırma ise denklem (99)'daki şekilde yapılmaktadır.

$$X = Af(X) \quad (99)$$

Burada $f(X)$ doğrusal olmayan karıştırma matrisi olup bu çalışmada $f(X)$ için denklem (100) kullanılmıştır.

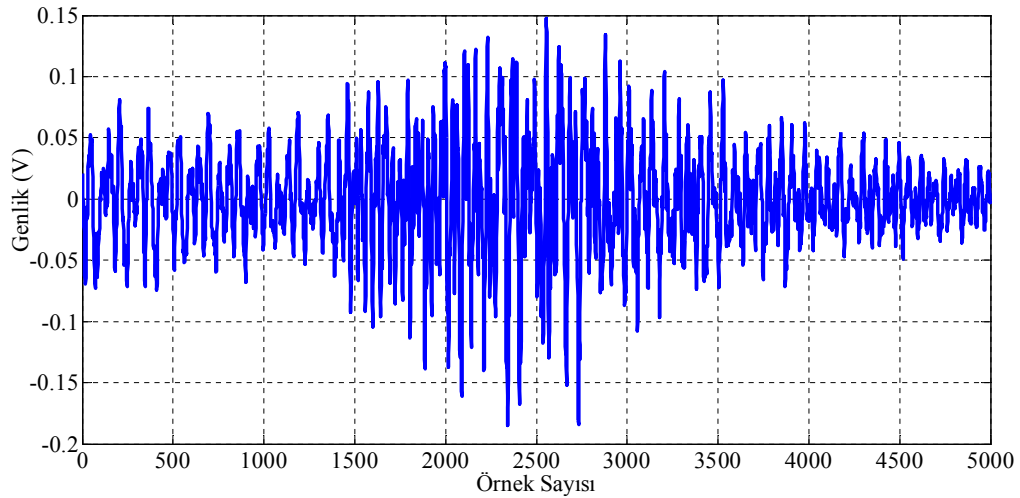
$$\begin{aligned} X_1 &= 0.2 * \exp(S_1) + 0.4 * \exp(S_2) + 0.5 * \exp(S_3) \\ X_2 &= 0.6 * \exp(S_1) + 0.8 * \exp(S_2) + 0.1 * \exp(S_3) \\ X_3 &= 0.8 * \exp(S_1) + 0.7 * \exp(S_2) + 0.3 * \exp(S_3) \end{aligned} \tag{100}$$

Burada X_1 , X_2 ve X_3 her bir sensöre ait doğrusal olmayan karışım matrisleridir.

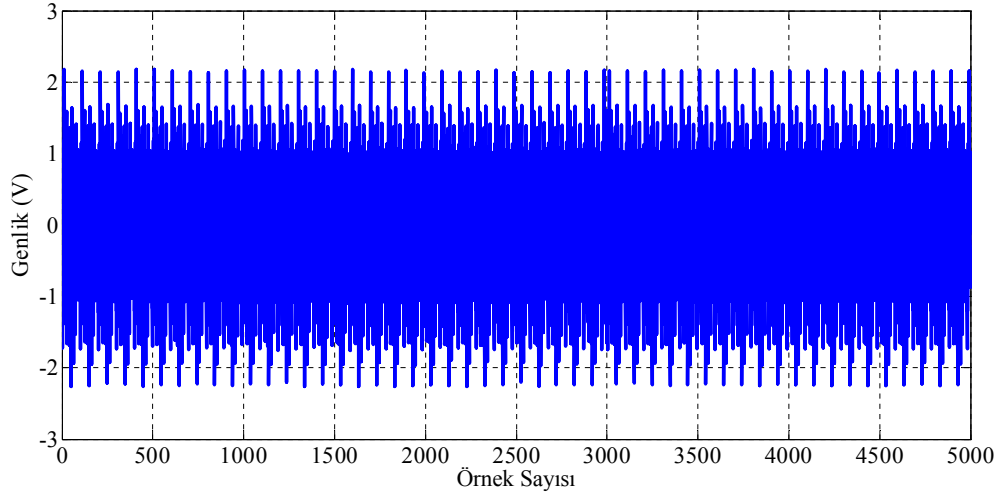
2. UYGULAMALAR VE SONUÇLARI

Karışmış farklı sesleri birbirinden ayırmak için bu uygulama yapılmıştır. Bu çalışmada da üç farklı kaynağın üç sensörle kaydedilip karıştırıldıktan sonra tekrar kaynak verilere ulaşmak için çeşitli algoritmalar ile uygulamalar yapılmıştır. Karıştırma sırasında sensörlerle kaynakların birbirlerine eşit mesafede oldukları varsayılmıştır. Aksi takdirde sonuçların değişeceği muhakkaktır.

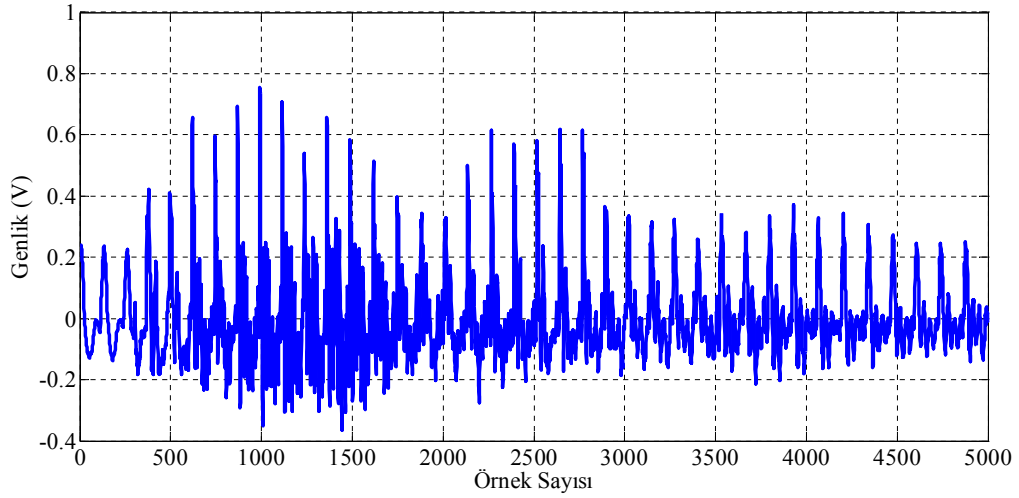
Yapılan deneysel çalışmalarda piyano sesi, gauss gürültüsü ve insan konuşma sesine ait veriler kullanılarak önerilen yöntemlerin simülasyonları yapılmıştır. Bu sinyaller 100'er defa rastgele seçilen doğrusal ve doğrusal olmayan karıştırma matrisleriyle karıştırıldıktan sonra çekirdek kanonik korelasyon analizi (ÇKKA), kanonik korelasyon analizi (KKA), ayırık dalgacık dönüşümlü çekirdek kanonik korelasyon analizi (ADD+ÇKKA) ve ayırık dalgacık dönüşümlü kanonik korelasyon analizine (ADD+KKA) uygulanmış olup bu algoritmalar ile ayrıştırılmıştır. Kullanılan algoritmaların performanslarını değerlendirebilmek için sonuçlar işaret gürültü oranı (İGO) cinsinden ölçülmüş ve sonuçlar karşılaştırılmıştır. Bu işlemler periyodik olarak veri uzunluğu 500, 1000, 5000, 10000 ve 15000 örnek alınarak tekrarlanmıştır. Böylece algoritmaların değişen veri uzunluklarına göre performans analizleri de yapılmıştır. Şekil 16, 17 ve 18'de üç sinyale ait 5000 örnek sayısı için kaynak sinyalleri gösterilmiştir.



Şekil 16. Piyano sesine ait kaynak işareti



Şekil 17. Gauss gürültüsüne ait kaynak işareti

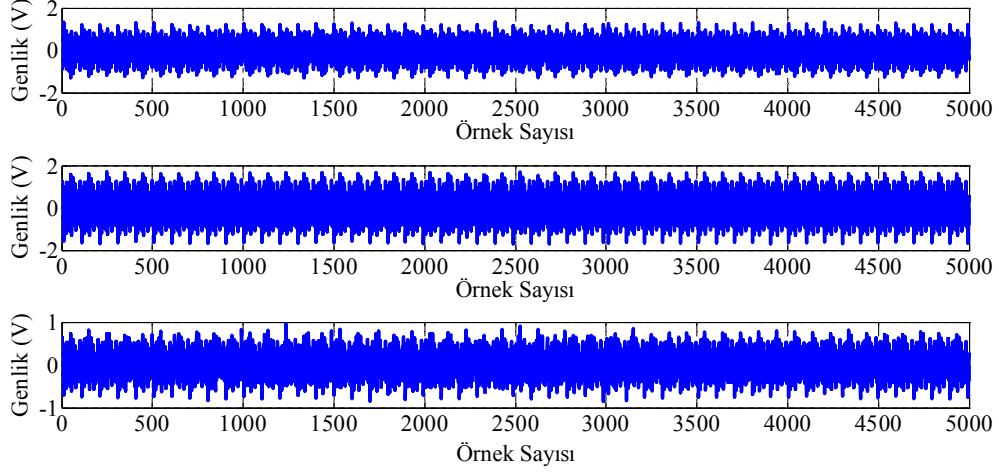


Şekil 18. Konuşma sesine ait kaynak işareti

2.1. Doğrusal Karışım Sonuçları

Uygulamada kaynak işaretler doğrusal ve doğrusal olmayan karıştırma yöntemleri olarak iki grupta incelenmiştir. Önce doğrusal olarak karıştırılmış sinyallerin İGO'ları analiz edilmiştir. Şekil 19'da verilen üç farklı sinyalin üç sensör tarafından kaydedilen

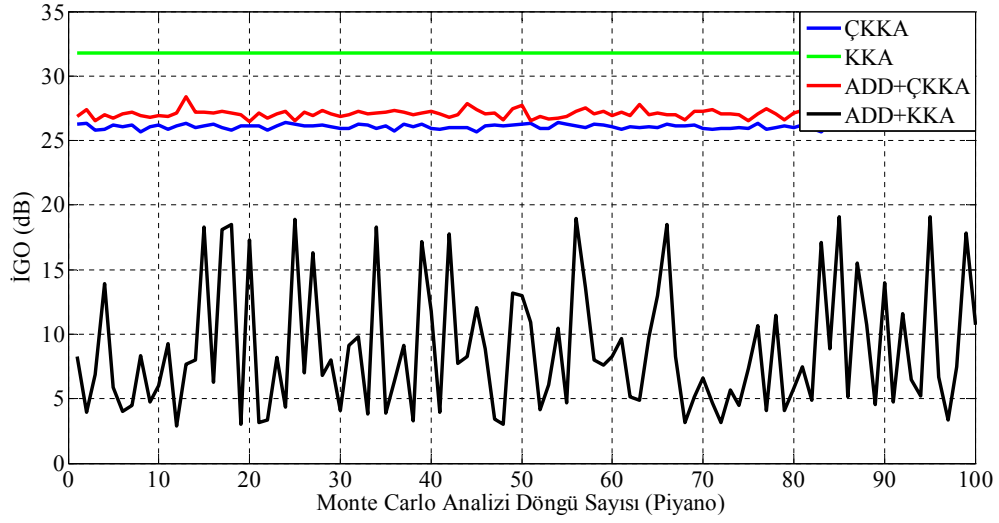
karışmış halleri gösterilmiştir. Burada her bir sensörün kaynağa olan uzaklığı ve seslerin şiddeti o sensörden alınan sinyalin şeklini etkilemektedir.



Şekil 19. Üç adet kaynak sinyalinin üç sensörle kaydedilip karıştırılmış şekli

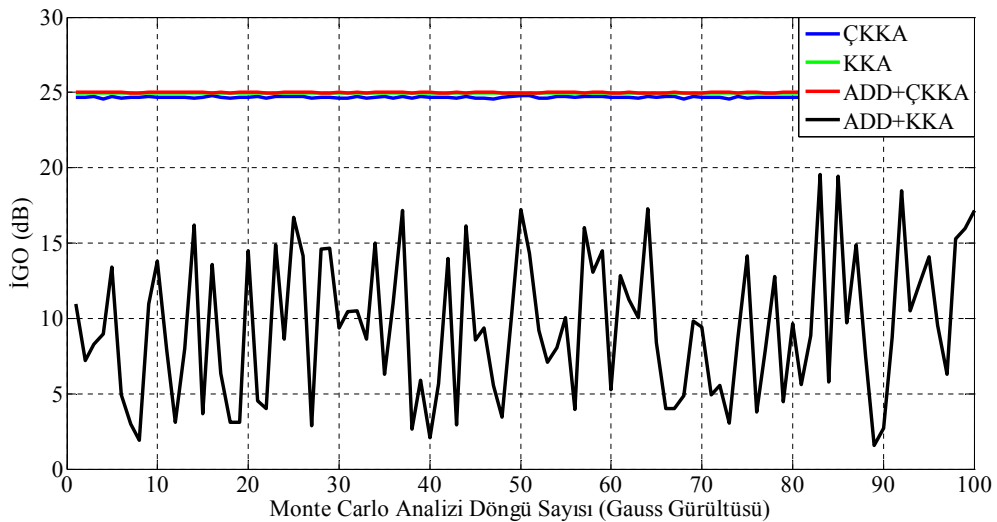
ÇKKA, KKA, ADD+ÇKKA ve ADD+KKA algoritmaların performans analizini daha iyi görebilmek için kaynak sinyaller 100 farklı karıştırma matrisi ile karıştırılmış ve her dört algoritmayla tekrar ayrıştırılmıştır. Bu işlemler kaynak işaretlerinden 500, 1000, 5000, 10000 ve 15000 örnek alınarak tekrarlanmıştır. Şekillerdeki grafiklerde işaret gürültü oranının daha iyi gözlemlenmesi ve yorumlanabilmesi için 5000 örnek adımları kullanılmıştır. Her bir kaynak işareti için sonuçlar ayrı ayrı grafiklerde gösterilmiştir.

Şekil 20’de piyano sesi için elde edilen algoritmaların İGO’su verilmiştir. Kullanılan algoritmalar içerisinde KKA’nın daha iyi sonuç verdiği ve çok stabil çalıştığı gözlemlenmiştir. KKA’ya ADD uygulandıktan sonra ise İGO’nun aşırı derecede düştüğü anlaşılmıştır. Yalnız ADD’nin ÇKKA’yı ise artı yönde etki ettiği görülmüştür. Verilen bu grafikte göze çarpan bir diğer nokta ise ADD+KKA algoritmasının farklı karışım matrisi değerlerinden oldukça fazla etkilendiğidir. ADD+KKA algoritmasının İGO sonuçlarına bakıldığında değer aralığının diğer üç yönteme göre oldukça fazla değiştiği gözlenmektedir.



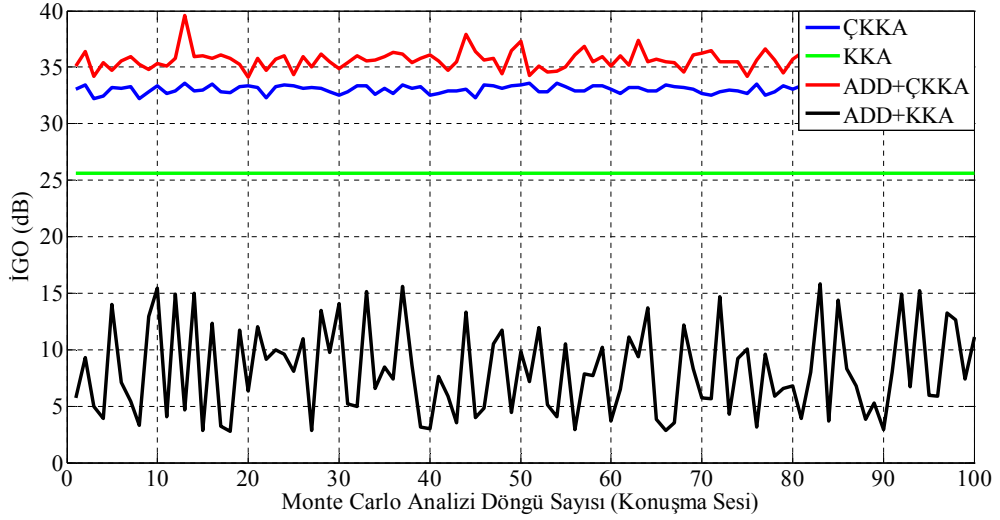
Şekil 20. Piyano için Monte Carlo analizi İGO sonuçları

Gauss gürültüsü için şekil 21’de görüldüğü üzere KKA ile ADD+ÇKKA algoritmalarının İGO’ları birbirleri ile neredeyse aynı olup ÇKKA algoritmasına göre iyi sonuçlar vermektedir. Aynı şekilde ADD+KKA algoritması İGO değer aralığı oldukça değişken değerler göstermiştir.

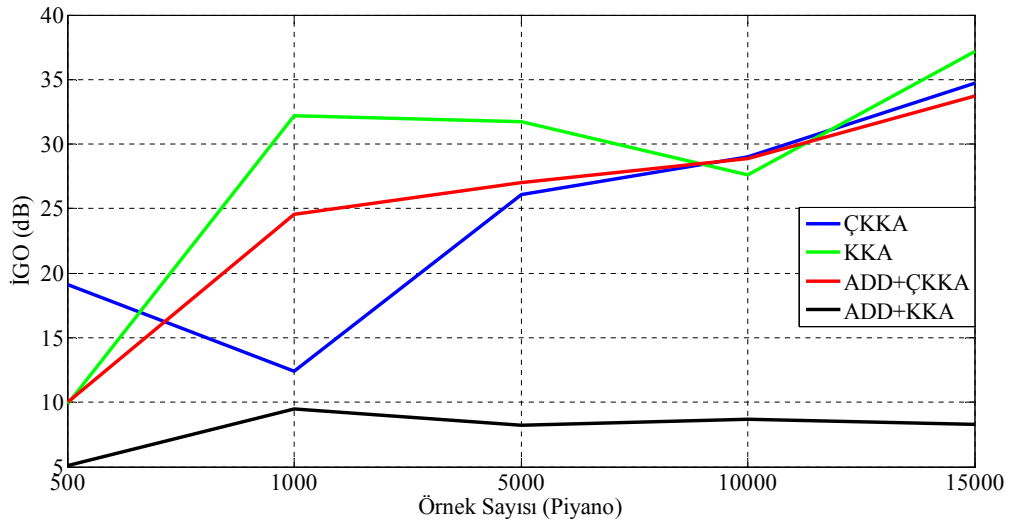


Şekil 21. Gauss gürültüsü için Monte Carlo analizi İGO sonuçları

Şekil 22'de konuşma verisinde ise ADD+ÇKKA'nın diğer algoritmalara göre İGO'nun çok daha iyi sonuçlar verdiği ve özellikle ÇKKA'yı hissedilir derecede başarımlar oranını arttırdığı gözlemlenmiştir. Dikkat edilecek bir nokta ise ADD+ÇKKA'nın her bir veri analizinde de diğer algoritmalara göre daha iyi sonuçlar vermesidir.



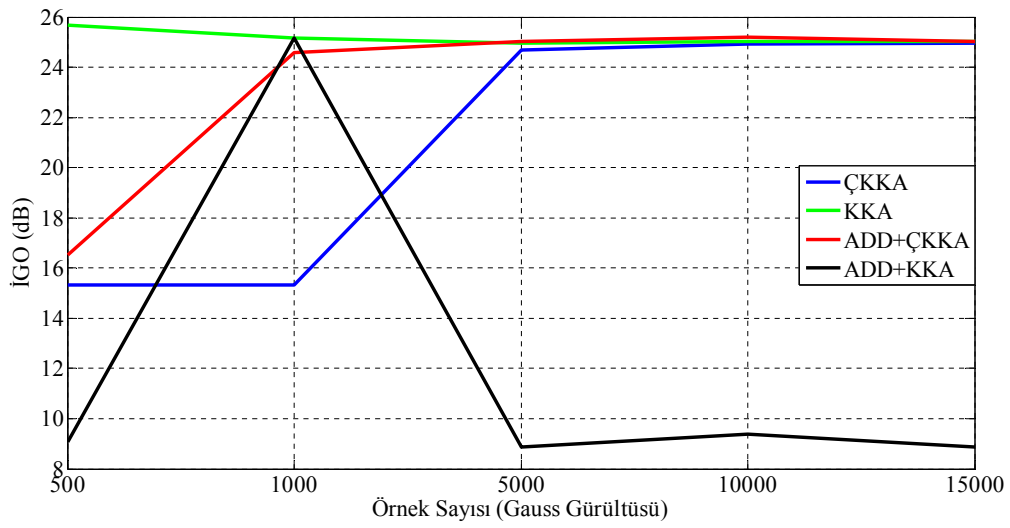
Şekil 22. Konuşma sesi için Monte Carlo analizi İGO sonuçları



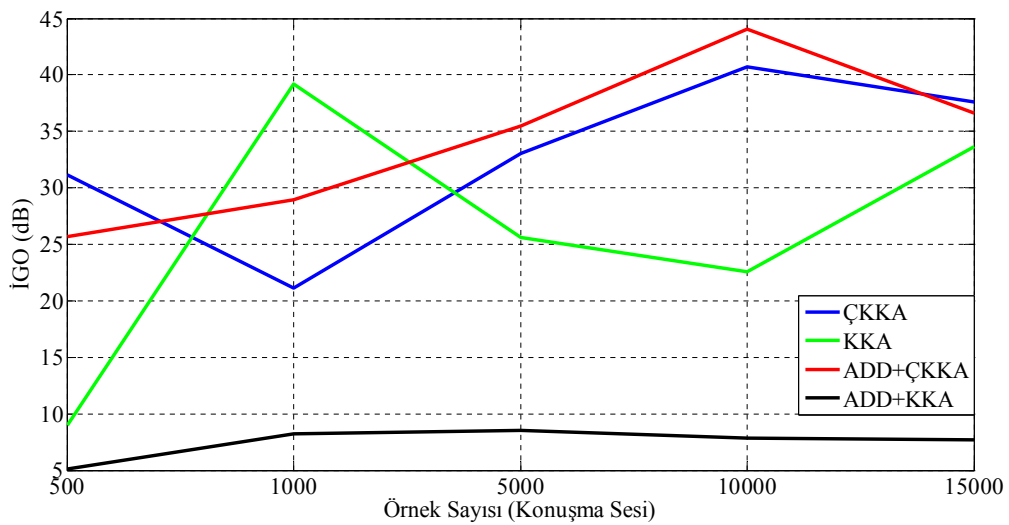
Şekil 23. Piyano sesi için örnek sayısı

Şekil 23'deki algoritmaların örnek sayısına göre İGO'ları incelendiğinde piyano sesi için veri boyutu arttığında İGO'nun arttığı, algoritmalar arasında veri boyutu düşükken en iyi sonucu ÇKKA'nın verdiği gözlemlenmiştir. ADD+ÇKKA'nın ise veri boyutuna paralel olarak arttığı görülmektedir.

Şekil 24'de Gauss gürültüsü için algoritma sonuçları verilmiştir. Şekilden de görüleceği üzere veri boyutu arttıkça üç algoritma aynı İGO'lara sahip olduğu gözlemlenmiştir. Burada da ADD+ÇKKA algoritması örnek sayısına göre devamlı artış göstermiştir.



Şekil 24. Gauss gürültüsü için örnek sayısı



Şekil 25. Konuşma sesi için örnek sayısı

Şekil 25’de ise konuşma sesi için algoritmaların İGO verilmiştir. Genel olarak bakıldığında ADD+ÇKKA algoritması diğer algoritmalara göre daha iyi sonuç vermektedir. Bu yöntem ile ölçüm sonuçlarında ortalama olarak ÇKKA’ya ADD uygulanarak İGO değerlerinde iyileştirme yapıldığı gözlemlenmiştir.

Doğrusal olarak 500, 1000, 5000, 10000 ve 15000 veri boyutları için karışmış sinyaller ÇKKA, KKA, ADD+ÇKKA ve ADD+KKA algoritmaları ile ayrıştırılmış ve veriler Tablo 1, 2, 3, 4 ve 5’de gösterilmiştir.

Tablo 1. Doğrusal olarak karışmış algoritmaların 500 veri boyutlu SNR değerleri

Doğrusal Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 500	1.veri için SNR (piyano)	19.12	9.89	10.04	5.07
	2.veri için SNR (gürültü)	15.31	25.68	16.51	9.06
	3.veri için SNR (konuşma)	31.14	9.01	25.72	5.08

Tablo 2. Doğrusal olarak karışmış algoritmaların 1000 veri boyutlu SNR değerleri

Doğrusal Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 1000	1.veri için SNR (piyano)	12.41	32.17	24.57	9.48
	2.veri için SNR (gürültü)	12.53	25.15	24.59	9.78
	3.veri için SNR (konuşma)	21.15	39.16	28.93	8.23

Tablo 3. Doğrusal olarak karışmış algoritmaların 5000 veri boyutlu SNR değerleri

Doğrusal Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 5000	1.veri için SNR (piyano)	26.06	31.75	27.01	8.2
	2.veri için SNR (gürültü)	24.69	24.96	25.02	8.87
	3.veri için SNR (konuşma)	33.04	25.58	35.49	8.56

Tablo 4. Doğrusal olarak karışmış algoritmaların 10000 veri boyutlu SNR değerleri

Doğrusal Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 10000	1.veri için SNR (piyano)	28.99	27.62	28.88	8.67
	2.veri için SNR (gürültü)	24.93	25.01	25.2	9.4
	3.veri için SNR (konuşma)	40.7	22.57	44.02	7.88

Tablo 5. Doğrusal olarak karışmış algoritmaların 15000 veri boyutlu SNR değerleri

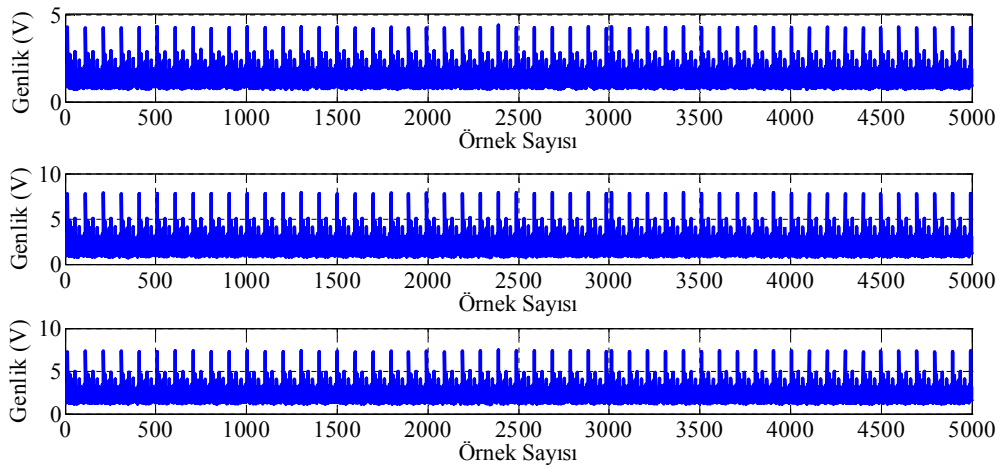
Doğrusal Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 15000	1.veri için SNR (piyano)	34.69	37.2	33.71	8.26
	2.veri için SNR (gürültü)	24.95	25.01	25.02	8.88
	3.veri için SNR (konuşma)	37.61	33.64	36.63	7.66

Tablo 6'da ise bu algoritmaların çalışma süreleri karşılaştırılmıştır. Çalışma süresi aynı zamanda algoritmanın işlem hacmi ile doğrudan orantılıdır. KKA algoritmasının diğer algoritmalara göre çok hızlı sonuç verdiği gözlemlenmiştir. ADD+ÇKKA'da ise çalışma süresinin ÇKKA'ya göre yarı yarıya düştüğü ölçülmüştür. ADD algoritmasını KKA'ya uyguladığımızda ise süre bakımından birbirlerine çok yakın oldukları ölçülmüştür.

Tablo 6. Algoritmaların çalışma süreleri

Adım Sayısı	ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
500	2.17sn	0.1sn	2.21sn	0.1sn
1000	3.65sn	0.11sn	2.04sn	0.11sn
5000	8.49sn	0.11sn	4.19sn	0.12sn
10000	13.61sn	0.12sn	5.7sn	0.14sn
15000	20.69sn	0.13sn	9.92sn	0.16sn

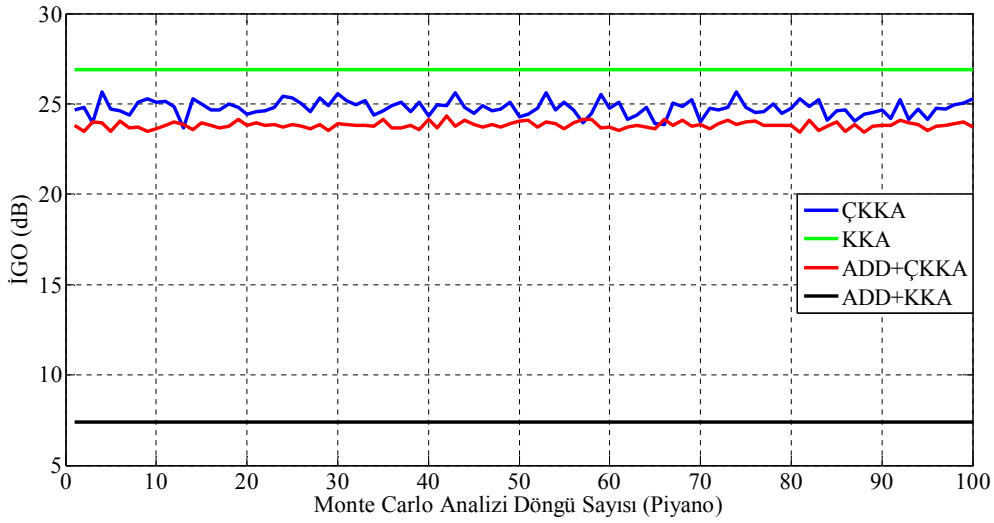
2.2. Doğrusal Olmayan Karışım Sonuçları



Şekil 26. Üç adet kaynak sinyalinin üç sensörle kaydedilip doğrusal olmayan yöntemle karıştırılmış şekli

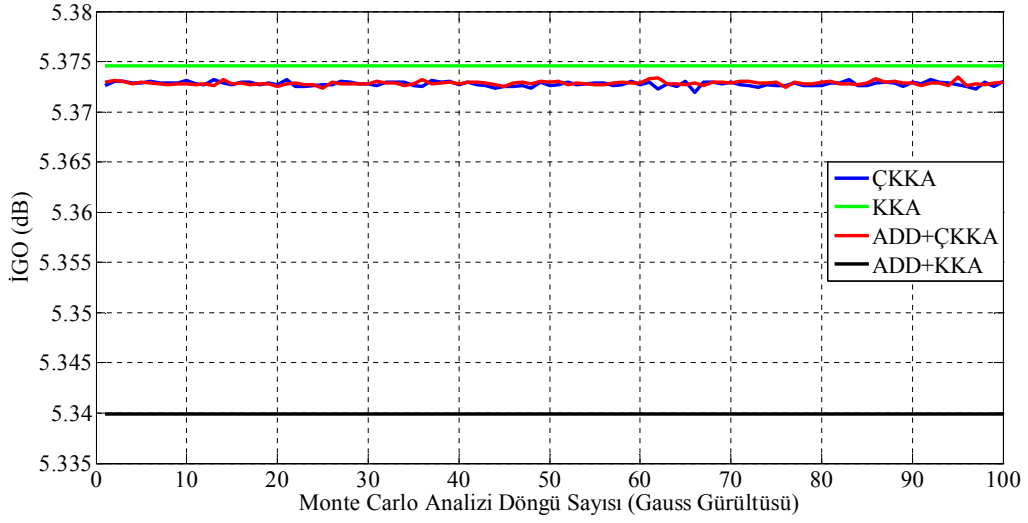
Buraya kadar yapılan işlemlerde kaynak sinyaller doğrusal olarak karıştırılıp dört farklı algoritmalar ile tekrar ayrıştırılmıştı. Bundan sonraki adımlarda ise kaynak sinyaller doğrusal olmayan yöntemlerle karıştırılıp tekrar ayrıştırılarak İGO'ları incelenmiştir. Şekil 26'da kaynak sinyallerin üç farklı sensörle doğrusal olmayan yöntemle kaydedilmiş şekli görülmektedir. Bu şekilde de örnek adımları 5000 olarak alınmıştır.

Şekil 27'de piyano sesi doğrusal olmayan 100 farklı karıştırma matrisi ile karıştırılarak İGO incelenmiştir. Doğrusal olmayan yöntemle karıştırılıp işleme alınan verilerin İGO'larına bakıldığında doğrusal karışıma göre daha stabil değerler çıktığı gözlemlenmiştir. KKA'nın iyi sonuçlar verdiği, ADD uygulandıktan sonra ise sonuçlarda aşırı derece de düşüş görülmektedir. ÇKKA ile ADD+ÇKKA algoritmalarının İGO sonuçları birbirine yakın oldukları gözlemlenmiştir.



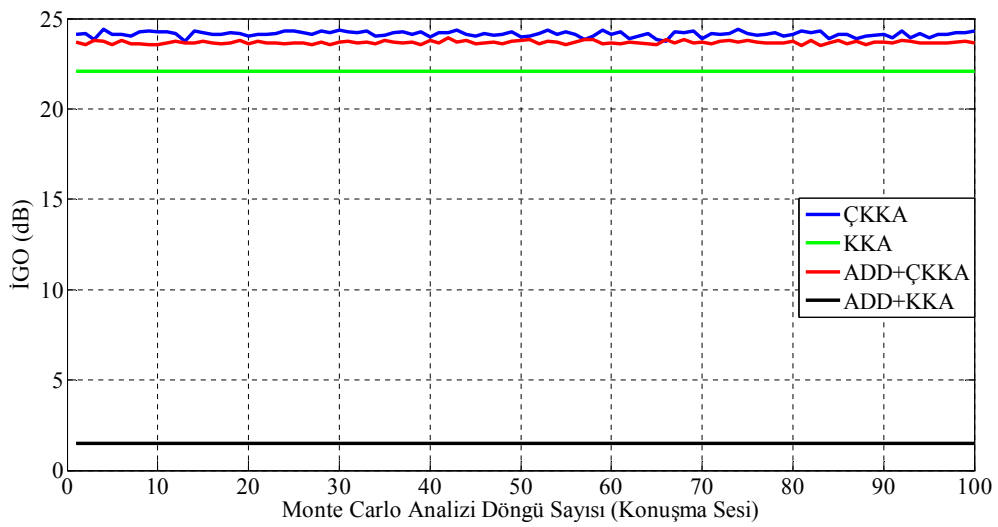
Şekil 27. Piyano için Monte Carlo analizi İGO sonuçları (Doğrusal olmayan karışım)

Piyonu için geçerli olan algoritma sonuçları Gauss gürültüsü için de geçerli olup buradaki İGO değerleri birbirine çok çok yakın sonuçlar verdiği Şekil 28'de görülmüştür.



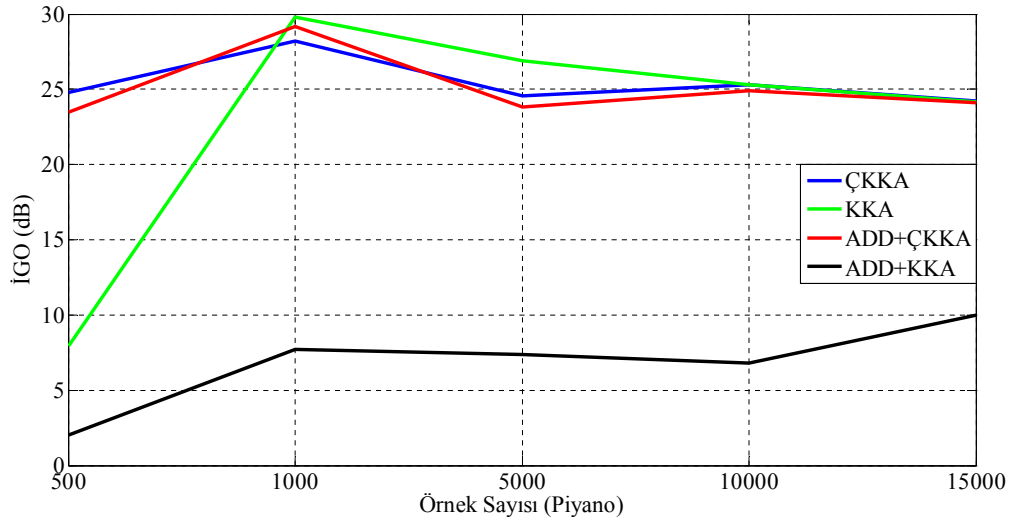
Şekil 28. Gauss gürültüsü için Monte Carlo analizi İGO sonuçları (Doğrusal olmayan karışım)

Şekil 29'da ise konuşma sesinde ÇKKA ve ADD+ÇKKA algoritmaları KKA'dan daha iyi sonuçlar verdiği gözlemlenmiştir. Doğrusal olmayan yöntemle karıştırılan kaynak işaretleri ayırıştırırken KKA ve ADD+KKA algoritmaları çok stabil sonuçlar verdiği grafiklerden anlaşılmıştır. İGO değerleri genellikle ÇKKA ve ADD+ÇKKA algoritmalarında birbirine yakın sonuçlar vermiştir.



Şekil 29. Konuşma sesi için Monte Carlo analizi İGO sonuçları (Doğrusal olmayan karışım)

Şekil 30'daki algoritmaların örnek sayısına göre İGO'ları incelendiğinde piyano sesi için veri boyutu arttığında İGO'nun arttığı, KKA hariç diğer algoritmaların birbirine yakın sonuçlar verdiği gözlemlenmiştir. KKA için veri boyutu attıkça İGO'nun da arttığı gözlemlenmiştir. ÇKKA ve ADD+ÇKKA algoritmalarının ise birbirlerine paralel olarak hareket ettiği görülüyor.



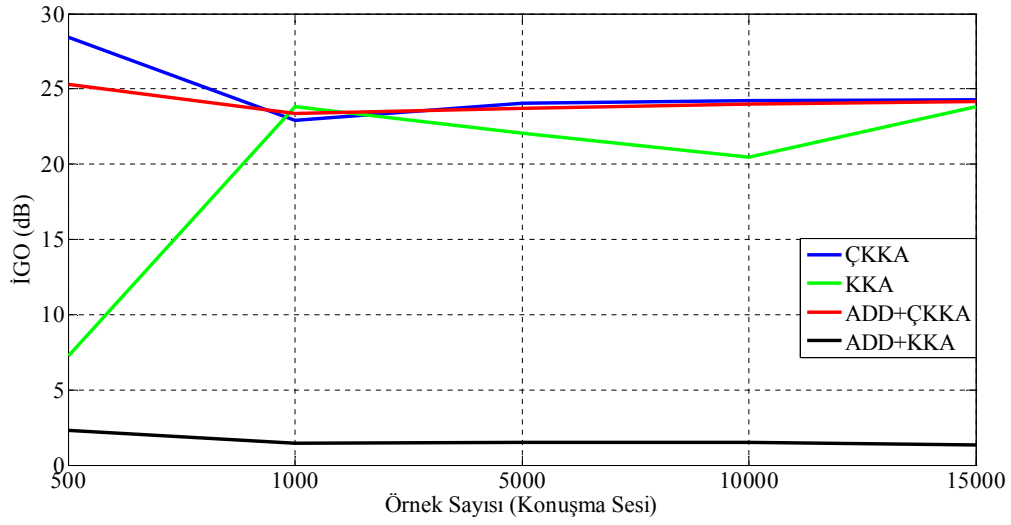
Şekil 30. Piyano sesi için örnek sayısı (Doğrusal olmayan karışım)



Şekil 31. Gauss gürültüsü için örnek sayısı (Doğrusal olmayan karışım)

Gauss gürültüsü için Şekil 31'de İGO aralıklarına bakıldığında birbirlerine çok yakın olduğu ve veri boyutuna göre değişimler olduğu görülüyor. Yalnız düşük veri boyutlarında her dört algoritmanın da İGO sonuçları arttığı önemlidir.

Konuşma sesinde ise düşük İGO sonuçlarında değişimler olduğu ve daha sonra ÇKKA ve ADD+ÇKKA algoritmaları birbirine yakın sonuçlar verdiği gözlemlenmiştir. Doğrusal olmayan yöntemle karıştırılan kaynak işaretlerde ADD+KKA algoritması diğer algoritmalara göre İGO'ları düşük çıkmıştır (Şekil 32).



Şekil 32. Konuşma sesi için örnek sayısı (Doğrusal olmayan karışım)

Tablo 7. Doğrusal olmayan yöntemle karışmış algoritmaların 500 veri boyutlu SNR değerleri

Doğrusal Olmayan Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 500	1.veri için SNR (piyano)	24.79	7.95	23.48	2
	2.veri için SNR (gürültü)	5.37	5.38	5.37	5.28
	3.veri için SNR (konuşma)	28.42	7.25	25.32	2.28

Doğrusal olmayan yöntemle 500, 1000, 5000, 10000 ve 15000 veri boyutları için karışmış sinyaller ÇKKA, KKA, ADD+ÇKKA ve ADD+KKA algoritmaları ile ayrıştırılmış ve veriler Tablo 7, 8, 9, 10 ve 11'de gösterilmiştir.

Tablo 8. Doğrusal olmayan yöntemle karışmış algoritmaların 1000 veri boyutlu SNR değerleri

Doğrusal Olmayan Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 1000	1.veri için SNR (piyano)	28.19	29.82	29.18	7.74
	2.veri için SNR (gürültü)	5.39	5.39	5.39	5.34
	3.veri için SNR (konuşma)	22.94	23.83	23.36	1.46

Tablo 9. Doğrusal olmayan yöntemle karışmış algoritmaların 5000 veri boyutlu SNR değerleri

Doğrusal Olmayan Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 5000	1.veri için SNR (piyano)	24.59	26.88	23.81	7.38
	2.veri için SNR (gürültü)	5.37	5.37	5.37	5.33
	3.veri için SNR (konuşma)	24.07	22.06	23.7	1.5

Tablo 10. Doğrusal olmayan yöntemle karışmış algoritmaların 10000 veri boyutlu SNR değerleri

Doğrusal Olmayan Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 10000	1.veri için SNR (piyano)	25.32	25.31	24.89	6.78
	2.veri için SNR (gürültü)	5.38	5.38	5.38	5.34
	3.veri için SNR (konuşma)	24.22	20.46	23.98	1.51

Tablo 11. Doğrusal olmayan yöntemle karışmış algoritmaların 10000 veri boyutlu SNR değerleri

Doğrusal Olmayan Karışım		ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
Veri boyutu 15000	1.veri için SNR (piyano)	24.21	24.19	24.12	10
	2.veri için SNR (gürültü)	5.37	5.37	5.38	5.34
	3.veri için SNR (konuşma)	24.27	23.81	24.14	1.35

Tablo 12'de ise doğrusal olmayan yöntemle karıştırılan verilerde kullanılan algoritmaların çalışma süreleri karşılaştırılmıştır. Bu sonuçlar doğrusal olarak karıştırılarak kullanılan algoritmaların işlem süreleri ile paralellik göstermiştir. KKA ile ADD+KKA algoritmasının diğer algoritmalara göre çok hızlı sonuç verdiği gözlemlenmiştir. ADD+ÇKKA'da ise çalışma süresinin ÇKKA'ya göre yarı yarıya düştüğü bu uygulamada da ortaya çıkmıştır.

Tablo 12. Algoritmaların çalışma süreleri (Doğrusal olmayan karışım)

Adım Sayısı	ÇKKA	KKA	ADD+ÇKKA	ADD+KKA
500	1.48sn	0.1sn	1.23sn	0.12sn
1000	1.7sn	0.11sn	1.24sn	0.11sn
5000	6.02sn	0.12sn	3.74sn	0.12sn
10000	13.49sn	0.13sn	5.73sn	0.14sn
15000	22.88sn	0.14sn	11.1sn	0.15sn

3. SONUÇLAR

Bu çalışmada; üç kaynaktan elde edilen karışmış işaretleri ÇKKA, KKA, ADD+ÇKKA ve ADD+KKA algoritmaları yardımıyla ayrıştırmadaki başarımları incelenmiştir. Yapılan çalışmalarda veriler yapay olarak doğrusal ve doğrusal olmayan karıştırma yöntemleri ile karıştırılıp ayrı ayrı uygulamalarda işaret gürültü oranlarına göre değerlendirilmiştir.

İlk yapılan doğrusal karıştırma sonuçlarındaki performanslarda örnek sayısı artıkça algoritmaların performanslarında da artış olduğu görülmektedir. Özellikle ÇKKA'ya göre ADD+ÇKKA algoritmasına uygulanan verilerin ayrıştırılmasında iyileştirme ve süre yönünden oldukça hızlandırdığı ortaya çıkmıştır. Dikkat çeken başka husus ise KKA algoritmasının konuşma sesi için İGO değerlerindeki düşüştür. ADD'yi KKA' ya uyguladığımızda ise sonuçlara olumsuz etki ettiği çok açık şekilde görülmektedir. Genel hatları ile incelendiğinde üç veri için KKA algoritması, veri uzunluğuna göre İGO değerlerinin çok değişiklik göstermesi, ADD+ÇKKA'nın ise daha stabil şekilde çalışması görülmektedir.

Bir sonraki çalışma da ise kaynaktan elde edilen veriler doğrusal olmayan karıştırma yöntemleri ile karıştırılıp aynı algoritmalar kullanılarak ayrıştırılmıştır. ADD+ÇKKA ve ÇKKA algoritmaları birbirlerine çok yakın ve yüksek İGO değerleri ile çalışırken ADD+KKA algoritması ise en düşük sonucu verdiği şekillerden (Şekil 30, 31, 32) anlaşılmıştır. Piyo ve konuşma sesinde KKA algoritması hariç diğer üç algoritma özellikle veri boyutu yüksek kaynaklarda çok iyi ve birbirlerine yakın sonuçlar verdiği gözlenmektedir. Süre bakımından ADD+ÇKKA algoritması ÇKKA'ya göre iki kat hızlı çalışmaktadır. ADD+KKA ve KKA'nın çalışma süreleri ise çok çok hızlıdır.

Kaynak sinyalleri ayrıştırırken ÇKKA algoritması bazı verilerde özdeğer ve özvektör ayrıştırma hatası vermektedir. Bu verilerin histogramlarına baktığımız da supergauss sinyaller olduğunda algoritma hata vermektedir. ADD+ÇKKA algoritması ile özdeğer ve özvektör ayrıştırma hatası da ortadan kalkmıştır.

4. ÖNERİLER

Yapılan bu çalışmada üç farklı karışmış verinin daha doğru ve daha hızlı bir şekilde ayrıştırılması için ÇKKA ve KKA'ya ADD algoritması uygulanmıştır. Yapılan çalışmada sensörlerin kaynaklara olan uzaklıkları eşit mesafede alındı ve yapay olarak veriler karıştırılmıştır. Süre önemli olmadığı müddetçe veriler yüksek boyutlarda alınıp uygulama yapılırsa İGO değerleri için daha yüksek sonuçlar elde edilebilir. ADD'nin ÇKKA'ya uygulanması ile performansı ve çalışma süresi arttırılmıştır. Fakat İGO'nun iyileştirilmesi ve daha iyi sonuçlar vermesi için çalışma yapılabilir.

5. KAYNAKLAR

1. Elmas, M., Gözü Kapalı Kaynak Ayırıştırma Seyreklik Tabanlı Yöntemler ve Düet Algoritması, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2010.
2. O'Grady, P. D., Pearlmutter B. A. ve Rickard S. T., Survey of Sparse and Non-sparse Methods in Source Separation, IJIST International Journal of Imaging Systems and Technology, 18 (2005) 78-85.
3. Herault, J. ve Jutten, C., Space Or Time Adaptive Signal Processing By Neural Models, Proceedings AIP Conference: Neural Networks for Computing, (1986) 206-211.
4. Bach, F. R. ve Jordan, M. I., Kernel Independent Component Analysis, Journal of Machine Learning Research, 3 (2002) 1-48.
5. Hotelling, H., Relations Between Two Sets of Variates, Biometrika, 28, 3-4s (1936) 321-377.
6. Mallat, S. G., A Theory For Multiresolution Signal Decomposition: The Wavelet Representation, IEEE Trans. Pattern Anal. And Machine Intelligence, 11 (1989) 674-693.
7. Comon, P., Independent Component Analysis: A New Concept, Signal Processing, 36 (1994) 287-314.
8. Herault, J. ve Jutten, C., Space or Time Adaptive Signal Processing by Neural Models, Proceedings AIP Conference: Neural Networks for Computing, (1986) 206-211.
9. Gaeta, M. ve Lancome, J. L., Source Separation Without Prior Knowledge: The Maximum Likelihood Solution, Proc. Of EUSIPCO'90, 55 (1990) 621-624.
10. Bell, A. J. ve Sejnowski, T. J., An Information-maximization Approach to Blind Separation and Blind Deconvolution, Neural Computation, 7, 6 (1995) 1129-1159.
11. Hyvarinen, A. ve Oja, E., A Fast Fixed-Point Algorithm for Independent Component Analysis, Neural Computation, 9,7 (1997) 1483-1492.
12. Jung, T. P., Makeig, S., Westerfield, M. ,Townsend, J. , Courchesne, E. ve Sejnowski, T. J., Removing Electroencephalographic Artifacts by Blind Source Separation, Psychophysiology, 37 (2000) 163-178.
13. Darmois, G., Analyse Generale Des Liaisons Stochastiques, Rev. Inst. Int. Stat., 21 (1953) 2-8.

14. Cherry, C. E., Some Experiments In the Recognition of Speech, With One or Two Ears, Journal of the Acoustical Society of America, 25 (1953) 975-979.
15. Serdengeçti, Ç., Biyomedikal İşaretlerin Ayrıştırılması, Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir, 2008.
16. Rencher, A. C., Methods of Multivariate Analysis, 2nd ed. John Wiley & Sons, 2002.
17. Hardle, W. K. ve Simar, L., Applied Multivariate Statistical Analysis, 2nd ed. Springer, 2003.
18. Toker, K. G., Kanonik Korelasyon Analizi İle Sistem Tanıma, Yüksek Lisans Tezi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2013.
19. Haykin S., Learning Processes; Single-Layer Perceptrons; Multilayer Perceptrons, Neural Networks A Comprehensive Foundation, 2nd ed. 1998.
20. Büyükkabasakal, K., Kumaş Dokuma Hatalarının Tespiti ve Sınıflandırılması, Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir, 2010.
21. Pearson, K., The Scope of Biometrika, Biometrika, 1 (1901) 1–2.
22. Jwmah, Y. M., Telemedikal Elektrokardiyografi Sinyallerin Yapay Sinir Ağları İle Değerlendirilmesi, Yüksek Lisans Tezi, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2010.
23. Bokman, L., Syungkwon, R. ve Park, F., C., Movement Primitives, Principal Component Analysis, and the Efficient Generation of Natural Motions, Proceedings of the 2005 IEEE International Conference on Robotics and Automation, April 2005, Barcelona, 4630-4635.
24. Manly, B. F. J., Multivariate Statistical Methods A premier, Chapman & Hall, USA, 1992.
25. Sevim, Y., Yapay Toplamsallık Yaklaşımı İle Anne ve Karnındaki Çocuğa Ait Ekg İşaretlerinin Ayrıştırılması, Doktora Tezi, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Trabzon, 2009.
26. Hyvarinen, A., Karhunen, J. ve Oja, E., Independent Component Analysis, John Wiley and Sons Inc., 2001, 481.
27. Yazar, I., Temel Bilesen Analizi ve Bağımsız Bilesen Analizi Yöntemlerini Temel Alan Bazı Görüntü Tanıma Uygulamaları ve Karşılaştırmaları, Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir, 2008.
28. Hyvarinen, A. ve Oja, E., Independent Component Analysis: Algorithms and Application, Neural Networks, 13, 4-5, (2000) 411-430.

29. Çekli, S., Bağımsız Bileşen Analizi ile Ayrıştırılmış Akustik İşaretlerden Varışlar Arası Zaman Farkına Dayanan Küresel İnterpolasyon En Küçük Kareler Yöntemi Kullanılarak Yer Tespiti, IEEE 20. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, Nisan 2012, Muğla, 18-20.
30. Lewicki, M. S. ve Sejnowski, T. J., Learning Nonlinear Overcomplete Representations for Efficient Coding, Advances in Neural Information Processing Systems, 1998, Denver, The MIT Press, 556-562.
31. Theis, F. J., Lang, E. W. ve Puntonet, C. G., A Geometric Algorithm for Overcomplete Linear ICA, Neurocomputing, 56 (2003) 381-398.
32. Borga, M., Learning Multidimensional Signal Processing, Doktora Tezi, Linköping University, Sweden, 1998.
33. Hair, J. F., Anderson, R. E., Tatham R. L. ve Black W. C., Multivariate Data Analysis, 5th edition, 1998, Copyright Prentice Hall, Inc.
34. Yıldırım, H., Albayrak, A. S., Gümüş, M. ve Akalın, T. C., Yüzme Hakemlerinde Örgütsel Bağlılık İle İş Tatmini Arasındaki İlişkinin Kanonik Korelasyon Analizi İle İncelenmesi, Zonguldak Karaelmas Üniversitesi Sosyal Bilimler Dergisi, 7, 13 (2011) 163-186.
35. Giray, S., Doğrusal Olmayan Kanonik Korelasyon Analizi ve Yaşam Memnuniyeti Üzerine Bir Uygulama, Doktora Tezi, Marmara Üniversitesi, Sosyal Bilimleri Enstitüsü, İstanbul, 2011.
36. Golob, T. F., A Non-Linear Canonical Correlation Analysis of Weekly Trip Chaining Behavior, UCI-ITS-AS-WP, 85,4 (1985) 37.
37. Özdamar, K., Paket Programlar İle İstatistiksel Veri Analizi, Kaan Kitabevi, 421-428, 1999.
38. Başaran, E., Kanonik Korelasyon Analizi ve Bir Uygulama, Yüksek Lisans Tezi, Uludağ Üniversitesi, Sosyal Bilimleri Enstitüsü, Bursa, 1998.
39. Çankaya, S., Kanonik Korelasyon Analizi ve Hayvancılıkta Kullanımı, Doktora Tezi, Çukurova Üniversitesi, Fen Bilimleri Enstitüsü, Adana, 2005.
40. Gündüz, M., Fonksiyonel Kanonik Korelasyon Analizi ile İmkb'de İşlem Gören Şirketlerin Kapanış Fiyatları İle İşlem Miktarları Arasındaki İlişkinin Araştırılması, Doktora Tezi, Atatürk Üniversitesi, Sosyal Bilimleri Enstitüsü, Erzurum, 2012.
41. Gao, D.D. ve Huang, R. B., Some Results on Canonical Correlation and Their Applications to a Linear Model, Linear Algebra and Its Applications, 321 (2000) 47-59.

42. Tatar, A. M. ve Eliçin, A., Ile de France x Akkaraman (G1) Melezi Erkek Kuzularında Süt Emme ve Besi Dönemindeki Canlı Ağırlık ve Vücut Ölçüleri Arasındaki İlişkinin Kanonik Korelasyon Metodu ile Araştırılması, Ankara Üniv. Ziraat Fak. Tarım Bilimleri Dergisi, 8, 1 (2002) 67-72.
43. Lee, J., Park, M. ve Kim, Y., An Application of Canonical Correlation Analysis Technique to Land Cover Classification of Landsat Images, ETRI Journal, 21,4 (1999) 41-51.
44. Özdamar, K., Paket Programlar ile İstatistiksel Veri Analizi: Çok Değişkenli Analizler, 7. Baskı, Kaan Kitabevi, Eskişehir, 2010.
45. Anderson, T.W., An Introduction To Multivariate Statistical Analysis. John Wiley & Sons, Inc., 1958, 374, Canada.
46. Tatsuoka, M. M., Multivariate Analysis: Techniques for Educational and Psychological Research. John Willey & Sons, Inc. 1971, New York. 310p.
47. Özdamar, K., Paket Programlar İle İstatistiksel Veri Analizi-2, Yenilenmiş 5. Baskı, Kaan Kitabevi , 2004.
48. Kolukısaoğlu, S., Doğrusal Olmayan Kanonik Korelasyon Analizi ve Depresyon Anksiyete ve Stres Ölçeğine Uygulanması, Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir, 2013.
49. Kaya, L., Birden Fazla Değişken İçeren Setler Arasındaki İlişkinin Kanonik Korelasyon Analizi İle Belirlenmesi, Yüksek Lisans Tezi, Harran Üniversitesi, Fen Bilimleri Enstitüsü, Şanlıurfa, 2008.
50. Minsky, M. L. ve Papert, S. A., Perceptron, Expanded Edition, Cambridge, MIT Press, London, UK, 1969.
51. Aykut, M., Örüntü Tanıma Amaçlı Doğrusal Olmayan Yöntemlerin Geliştirilmesi ve Uygulanması, Yüksek Lisans Tezi , Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Trabzon, 2007.
52. Cristianini, N. ve Taylor, J. S., An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, UK, 2000.
53. Fidan, Ü., Destek Vektör Makineleri İle Doküman Sınıflandırma, Yüksek Lisans Tezi, Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü, Afyon, 2013.
54. Liu, Q., Hanqing L. ve Songde M., Improving Kernel Fisher Discriminant Analysis for Face Recognition. Circuits and Systems for Video Technology, IEEE Transactions On Information And Systems 14,1 (2004) 42-49.
55. Scholkopf, B., The Kernel Trick For Distances. Advances in Neural Information Processing Systems, (2001) 301-307.

56. Haar, A., Zur Theorie Der Orthogonalen Funktionensysteme, Mathematische Annalen, 69, 3 (1910) 331- 371.
57. Goupillaud, P., Grossmann, A. ve Morlet, J., Cycle-Octave and Related Transforms in Seismic Signal Analysis, Geoexploration, 23 (1984-1985) 85-102.
58. Gu, I.Y.H. ve Bollen, M.H.J., Bridge The Gap: Signal Processing for Power Quality Applications, Electric Power Systems Research, 66, 1 (2003) 83-96.
59. Berdibek, M., Güç Sistemlerinde Güç Kalitesinin Ayrık Dalgacık Dönüşümü Kullanılarak İncelenmesi, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ, 2009.
60. Ağalar, S., Alçak Gerilim Güç Sistemlerinde Enerji Kalitesinin Belirlenmesi, Yüksek Lisans Tezi, Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir, 2003.
61. Arı, N., Çolak, Ö. H. ve Özen, Ş., Dalgacık Teorisi, Palme Yayınevi, Ankara, 2008.
62. Chouakri S. A., Reguig F. B., Ahmaidi S., ve Fokapu O., Wavelet Denoising of the Electrocardiogram Signal Based on the Corrupted Noise Estimation, Computers in Cardiology, 2005.
63. Strang, G. ve Nguyen, T., Wavelets and Filter Banks, Cambridge Press, Wellesley, 1996, 390p.
64. Türkoğlu, İ., Durağan Olmayan İşaretler İçin Zaman-Frekans Entropilerine Dayalı Akıllı Örüntü Tanıma, Doktora Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ, 2002.
65. Karakoç, A. O., Kalman Filtresi ve Ayrık Dalgacık Dönüşümü Tekniği Kullanılarak EKG İşaretlerinin Üzerindeki Gürültülerin Temizlenmesi, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2010.

6. EKLER

EK 1

İşaret Gürültü Oranı (İGO)

İGO değeri, çok boyutlu sinyal ayrıştırma işleminde kaynak işaretleri ile gürültü işaretleri arasındaki toplam işaret gürültü oranını göstermektedir olup,

$$İGO=10\log_{10}\left(\frac{E_i\{s_i^2\}}{E_i\{n_i^2\}}\right) \quad (E.1)$$

denklemleri ile hesaplanır. Burada s_i , tüm sensörlerdeki kaynak işaretlerini, n_i ise, tüm sensörlerdeki gürültü işaretlerini temsil etmektedir.

ÖZGEÇMİŞ

Hüsamettin ÇELİK, 20.03.1982 tarihinde Erzincan'da doğdu. İlköğretim öğrenimini Ziya Gökalp İlkokulu'nda orta öğrenimini Salih Erkan Ortaokulu'nda, lise öğrenimini ise Fatih Anadolu Teknik Lisesi'nde 2001 yılında tamamladı. 2001-2002 eğitim öğretim yılında Gazi Üniversitesi Teknik Eğitim Fakültesi Elektronik Öğretmenliği Bölümünü kazandı ve 2006 yılında bu bölümden mezun oldu. 2008 yılında Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı'nda yüksek lisans öğrenimine başladı. Erzincan Üniversitesi Tercan Meslek Yüksekokulu'nda öğretim görevlisi kadrosunda görev yapmaktadır. Çalışma alanları sayısal işaret işleme ve kör kaynak ayrıştırma üzerinedir.