

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**ELEKTROKARDİYOGRAM VERİLERİNİN İYİLEŞTİRİLMİŞ YAPAY ARI
KOLONİSİ (MABC) ALGORİTMASI İLE ANALİZİ**



DOKTORA TEZİ

Selim DİLMAÇ

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Programı

ARALIK 2017

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**ELEKTROKARDİYOGRAM VERİLERİNİN İYİLEŞTİRİLMİŞ YAPAY ARI
KOLONİSİ (MABC) ALGORİTMASI İLE ANALİZİ**

DOKTORA TEZİ

**Selim DİLMAÇ
(504922047)**

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Tamer ÖLMEZ

ARALIK 2017

İTÜ, Fen Bilimleri Enstitüsü'nün 504922047 numaralı Doktora Öğrencisi “Selim DİLMAÇ”, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “ELEKTROKARDİYOGRAM VERİLERİNİN İYİLEŞTİRİLMİŞ YAPAY ARI KOLONİSİ (MABC) ALGORİTMASI İLE ANALİZİ” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Prof. Dr. Tamer ÖLMEZ**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Prof. Dr. Ahmet Hamdi KAYRAN**
İstanbul Teknik Üniversitesi

Prof. Dr. Yekta ÜLGEN
Bahçeşehir Üniversitesi

Prof. Dr. Murat Tayfun GÜNEL
İstanbul Teknik Üniversitesi

Yrd.Doç.Dr. Ali NİZAM
Fatih Sultan Mehmet Vakıf Üniversitesi

Teslim Tarihi : 22 Kasım 2017
Savunma Tarihi : 27 Aralık 2017



ÖNSÖZ

Doktora çalışmamda bana destek olup yol gösteren, nitelikli bir doktora tezi hazırlanması için yönlendiren danışman hocam Prof. Dr. Tamer Ölmez'e teşekkürü bir borç bilirim.

Tez izleme jürimde yer alıp, öneri ve eleştirileri ile beni yönlendiren Prof. Dr. Zümray Dokur, Prof. Dr. Yekta Ülgen ve Prof. Dr. Ahmet Hamdi Kayran'a teze olan katkılarından dolayı teşekkür ediyorum.

Doktora çalışmasına başlamaya beni teşvik eden, yıllar boyunca sabırla ve bilgeliği ile beni destekleyen, eğiten, hakiki bir dost ve ağabey olarak tanıdığım merhum Prof. Dr. Mehmet Korürek'e teşekkürü bir borç biliyorum. Mekânı cennet olsun.

Çalışmalar esnasında deneyimlerini benimle paylaşan ve destek olan Yrd. Doç. Dr. Ali Nizam, Yrd. Doç. Dr. Berat Doğan ve Bilgisayar Y.Müh. Oğuz Dilmaç'a teşekkür ediyorum.

Yine tüm çalışmalar boyunca maddi ve manevi desteklerini esirgemeyen, sabır ve özveri ile yanımda bulunan tüm aileme teşekkür ediyorum.

Bu çalışmanın insanlığa, tıp elektroniği ve yapay zekâ uygulamaları üzerinde çalışma yapan araştırmacılara yararlı olmasını dilerim.

Kasım 2017

Selim Dilmaç
(Elektronik Y. Müh.)



İÇİNDEKİLER

ÖNSÖZ	v
İÇİNDEKİLER	vii
KISALTMALAR	ix
SEMBOLLER	xi
ÇİZELGE LİSTESİ	xiii
ŞEKİL LİSTESİ	xv
ÖZET	xix
SUMMARY	xxi
1. GİRİŞ.....	1
1.1 Tezin Amacı.....	1
1.2 Literatür Araştırması.....	1
1.3 Tezin Katkıları ve İzlenen Yol.....	5
2. ELEKTROKARDİYOGRAM (EKG) VE ARİTMİLER.....	9
2.1 Kalbin Yapısı	11
2.2 EKG İşaretinin Oluşumu ve Başlıca Bölümleri	14
2.3 Kalp Vurularının Tipleri	17
2.3.1 Normal vurular (N).....	17
2.3.2 Aritmik vurular.....	19
2.3.2.1 Kulakçık erken vuruları (APB).....	19
2.3.2.2 Erken karıncık kasılmaları (PVC).....	20
2.3.2.3 Karıncık taşikardisi (VT).....	22
2.3.2.4 Karıncık fibrilasyonu (VF).....	23
2.3.2.5 Sağ dal bloğu (RBBB).....	23
2.3.2.6 Karıncık ve normal vuru füzyonu (F).....	24
2.3.2.7 Paced ve normal vuru füzyonu (f).....	24
2.3.2.8 Anormal kulakçık erken vuruları (a).....	25
2.3.2.9 Jonksiyonel kaçış vurusu (j)	25
3. EKG VERİLERİNİN ANALİZİ.....	27
3.1 Ön İşleme.....	27
3.2 EKG Verileri İçin Öznitelik Çıkartma.....	27
3.2.1 Zaman domenli öznitelikleri.....	28
3.2.2 Frekans domenli öznitelikleri.....	30
3.2.3 Özniteliklerin indirgenmesi	31
3.2.4 Eğitim ve test veri kümeleri.....	33
3.3 EKG Verilerindeki Vuru Tiplerinin Sınıflandırılması.....	34
3.3.1 Öbekleme ve sınıflama.....	34

3.3.2	Klasik sınıflama yöntemleri	36
3.3.2.1	En yakın ortalamalar sınıflayıcı (NMC).....	36
3.3.2.2	K-en yakın komşu sınıflayıcı (KNN).....	37
3.3.2.3	Kohonen'in öz-düzenlemeli harita sınıflayıcısı (SOM).....	38
3.3.3	Doğadan esinlenen sınıflama yöntemleri	40
3.3.3.1	Karınca kolonisi optimizasyonu tabanlı sınıflayıcı (ACO)...	41
3.4	Sınıflama performans ölçütleri.....	44
4.	ABC VE MABC ALGORİTMALARI İLE SINIFLANDIRMA İŞLEMİ..	47
4.1	ABC Algoritması.....	48
4.1.1	ABC algoritmasının yapısı.....	49
4.1.2	ABC algoritması ile sınıflama.....	54
4.2	İyileştirilmiş ABC Algoritması (MABC).....	59
4.2.1	İyileştirilmiş ABC algoritması parametrelerinin kurulması ve ilkleştirilmesi.....	59
4.2.2	İyileştirilmiş ABC algoritması kontrol parametreleri	60
4.2.3	İyileştirilmiş ABC algoritmasının işlem adımları.....	63
4.3	MABC Algoritmasının EKG Vuru Tiplerinin Sınıflandırılmasına Uygulanması	64
5.	BİLGİSAYAR BENZETİM SONUÇLARI.....	67
5.1	MIT-BIH Veri Tabanı.....	67
5.2	EKG Verilerine Ait Zaman ve Frekans Domeni Öznitelikleri	73
5.3	Zaman Domeni Öznitelikleri ile EKG Vurularının Sınıflandırılması ...	85
5.3.1	MABC algoritması ve diğer sınıflayıcılar ile sınıflama sonuçları ..	86
5.4	Zaman ve Frekans Domeni Öznitelikleri ile EKG Vurularının Sınıflandırılması	90
5.4.1	MABC algoritması ve diğer sınıflayıcılar ile sınıflama sonuçları ..	91
5.5	K-Katlı Çapraz Doğrulama ile Sınıflama Sonuçları	95
5.6	Sınıflayıcı Performanslarının Karşılaştırılması.....	97
6	SONUÇLAR	101
	KAYNAKLAR.....	103
	EKLER.....	111
	ÖZGEÇMİŞ.....	121

KISALTMALAR

ABC	: Artificial Bee Colony (Yapay Arı Kolonisi)
Acc	: Accuracy (Sınıflama Başarımı)
ACO	: Ant Colony Optimization (Karıncı Kolonisi Optimizasyonu)
APB	: Atrial Premature Beat (Kulakçık Erken Vurusu)
FN	: False Negative (Yanlış Negatif)
FP	: False Positive (Yanlış Pozitif)
GA	: Genetic Algorithm (Genetik Algoritma)
KNN	: K-Nearest Neighbour (K-En Yakın Komşu)
MABC	: Modified Artificial Bee Colony (İyileştirilmiş Yapay Arı Kolonisi)
Ppr	: Positive Predictivity (Pozitif Öngörü)
PVC	: Premature Ventricular Contraction (Erken Karıncık Kasılması)
PSO	: Particle Swarm Optimization (Parçacık Sürü Optimizasyonu)
Se	: Sensitivity (Duyarlık)
SOM	: Self-Organizing Map (Kendi Kendisine Organize olan Haritalar)
Sp	: Specificity (Özgüllük)
SVTA	: Supra Ventricular Tachycardia Attack (Karıncık Üstü Taşikardi Atağı)
TN	: True Negative (Doğru Negatif)
TP	: True Positive (Doğru Pozitif)

SEMBOLLER

$\mathbf{X}^q(t)$: Yapay Arı Kolonisi algoritmasında t. yineleme adımındaki q. işçi arıya ait konum bilgilerini (öbek merkez koordinatlarını) içeren k-satırlı, p-sütunlu matristir.

$\underline{o}_i^{tj}$: j-inci sınıfa ait i-boyutlu t-inci örnek vektörü

$\hat{\underline{\mu}}_i^j$: j-inci sınıfa ait i-boyutlu örnek vektörlerinin ortalaması

W_i^j : j-inci sınıfın sınıf-içi saçılım matrisi

B_i : sınıflar-arası saçılım matrisi

k : sınıf sayısı

p : öznitelik sayısı

D_i : i-inci boyutta (i. Özniteliğin) diverjans değeri



ÇİZELGE LİSTESİ

	<u>Sayfa</u>
Çizelge 3.1 : Sınıflandırma algoritmasının yapmış olduğu etiketlemelere ait doğruluk çizelgesi.....	44
Çizelge 3.2 : Üç sınıf için sınıflama matrisi.....	46
Çizelge 4.1 : Gerçek arılar ve yapay arılar arasındaki benzetim ve karşılaştırmalar	48
Çizelge 5.1 : Kullanılan MIT-BIH kayıtlarındaki vuru tipleri ve örnek sayıları ..	73
Çizelge 5.2 : Tez çalışmasında incelenmiş olan zaman domeni özniteliklerinin listesi	76
Çizelge 5.3 : En yüksek ayırt ediciliğe sahip ilk 8 öznitelik.....	84
Çizelge 5.4 : VK ve VK_zaman kümelerinde bulunan örnek vektörü sayılarının vuru tiplerine göre dağılımları.....	85
Çizelge 5.5 : VK eğitim veri kümesi üzerinde çeşitli geçerlilik indisi yöntemleri ile elde edilen optimum öbek sayıları	86
Çizelge 5.6 : MABC sınıflama algoritması ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi	86
Çizelge 5.7 : MABC algoritmasının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi.....	87
Çizelge 5.8 : NMC sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi	87
Çizelge 5.9 : NMC sınıflayıcısının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi	88
Çizelge 5.10 : KNN sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi	88
Çizelge 5.11 : KNN sınıflayıcısının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi	88
Çizelge 5.12 : Kohonen SOM algoritması tabanlı sınıflayıcının farklı kontrol parametreleri ile VK_zaman test kümesi üzerinde çalıştırılmasından elde edilen sınıflama başarımları	89
Çizelge 5.13 : Kohonen SOM algoritması tabanlı sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi.....	89
Çizelge 5.14 : Kohonen SOM algoritması tabanlı sınıflayıcının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi	90

Çizelge 5.15	: ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi...	90
Çizelge 5.16	: ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi	90
Çizelge 5.17	: MABC sınıflama algoritması ile VK test kümesi üzerinde elde edilen sınıflama matrisi	91
Çizelge 5.18	: MABC algoritmasının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi	91
Çizelge 5.19	: NMC sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi	92
Çizelge 5.20	: NMC sınıflayıcısının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi	92
Çizelge 5.21	: KNN sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi	93
Çizelge 5.22	: KNN sınıflayıcısının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi	93
Çizelge 5.23	: Kohonen SOM algoritması tabanlı sınıflayıcının farklı kontrol parametreleri ile VK test kümesi üzerinde çalıştırılmasından elde edilen sınıflama başarımları	94
Çizelge 5.24	: Kohonen SOM algoritması tabanlı sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi	94
Çizelge 5.25	: Kohonen SOM algoritması tabanlı sınıflayıcının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi	94
Çizelge 5.26	: ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi	95
Çizelge 5.27	: ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi	95
Çizelge 5.28	: VK-KFold kümesinde bulunan örnek vektörlerinin vuru tiplerine göre dağılımları.....	96
Çizelge 5.29	: VK-KFold, Ex-Tx kümelerinde yer alan, farklı vuru tiplerine ait örnek vektörlerinin sayıları	96
Çizelge 5.30	: VK-KFold, Ex-Tx kümeleri kullanılarak elde edilen sınıflama başarımları.....	96
Çizelge 5.31	: VK_zaman veri kümesi üzerinde farklı sınıflama yöntemleri ile elde edilen sınıflama başarımları	97
Çizelge 5.32	: VK veri kümesi üzerinde farklı sınıflama yöntemleri ile elde edilen sınıflama başarımları	98
Çizelge 5.33	: VK veri kümesi üzerinde farklı sınıflama yöntemlerinin sınıflama süreleri	98

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Dış elektriksel uyarı ve iyon pompaları etkisi ile aksiyon potansiyelinin oluşumu	9
Şekil 2.2 : Aksiyon potansiyelinin aşamaları	10
Şekil 2.3 : Kalbin çeşitli bölümlerindeki aksiyon potansiyelleri ve bunların toplamından EKG işaretinin oluşumu.....	10
Şekil 2.4 : Kalbin başlıca bölümlerinin yapısı.....	11
Şekil 2.5 : Kalbin elektriksel uyarı sistemi ve kalp içindeki konumu.....	12
Şekil 2.6 : Kalpteki ileti sisteminin bölümlerinin uyarı başlatma sıklıkları.....	13
Şekil 2.7 : Kalbin kardiyak vektörünün sembolik gösterimi.....	14
Şekil 2.8 : Willem Einthoven tarafından 1903 yılında geliştirilen elektrokardiyograf cihazı.....	15
Şekil 2.9 : Standart çift kutuplu ve kuvvetlendirilmiş tek kutuplu derivasyonlarda elektrotların bağlanma yerleri.....	15
Şekil 2.10 : Göğüs derivasyonları için elektrotların bağlanma yerleri.....	16
Şekil 2.11 : Normal bir kalp vurusuna ait EKG işareti.....	16
Şekil 2.12 : Çeşitli hastalardan alınmış olan normal vuru örnekleri	18
Şekil 2.13 : APB tipi vuru örnekleri.....	19
Şekil 2.14 : SVTA tipi ritim örneği.....	20
Şekil 2.15 : PVC tipi vuru örnekleri.....	21
Şekil 2.16 : Bigemini tipi vuru örnekleri.....	22
Şekil 2.17 : Karıncık taşikardisi (VT) tipi ritim örnekleri.....	22
Şekil 2.18 : Karıncık fibrilasyonu (VF) tipi ritim örnekleri.....	23
Şekil 2.19 : MIT-BIH veritabanı 212 numaralı kayıttaki sağ dal bloğu (RBBB) tipi vuru örnekleri	23
Şekil 2.20 : Karıncık vurusu ve normal vuru füzyonu (F) tipi vuru örnekleri	24
Şekil 2.21 : Paced ve normal vuru füzyonu (f) tipi vuru örnekleri.....	25
Şekil 2.22 : Anormal kulakçık erken vurusu (a) tipi vuru örnekleri	25
Şekil 2.23 : Jonksiyonel kaçış vurusu (j) tipi vuru örnekleri	26
Şekil 3.1 : Pan&Tompkins algoritmasının başlıca adımları	28
Şekil 3.2 : Zaman domenli özniteliklerine örnekler	29
Şekil 3.3 : KNN sınıflayıcıda komşuluğun etkisi	37

Şekil 3.4 : Kohonen SOM haritasının gösterilimi	38
Şekil 3.5 : SOM yapay sinir ağının yapısı	39
Şekil 3.6 : SOM yapay sinir ağında komşuluk ilişkileri	39
Şekil 3.7 : Karıncaların yuva ile gıda kaynağı arasındaki kısa yolu bulup kullanması	41
Şekil 3.8 : Karıncaların gidilecek yolu seçmeleri süreci	43
Şekil 4.1 : İşçi arıların dans uçuşu ile gıda kaynağının konumu bilgisini diğer arılara aktarması	49
Şekil 4.2 : Dört boyutlu uzayda, üç öbekli veri kümesi için, q . işçi arının (çözümün) konumu	51
Şekil 4.3 : İki boyutlu sanal uzayda, 24 örüntü, 5 işçi ve 5 gözcü arıya ait başlangıç konumları	56
Şekil 4.4 : 50. arama adımında işçi ve gözcü arıların konumları.....	57
Şekil 4.5 : 100. arama adımında işçi ve gözcü arıların konumları	58
Şekil 4.6 : Üç ayrı sınıfa ait örnek vektörleri ve bulunan öbek merkezleri	58
Şekil 4.7 : Aralarında girişim olan üç ayrı sınıfa ait örnek vektörleri ve bulunan öbek merkezleri	59
Şekil 4.8 : IRIS veri kümesinin hazırlanmasında kullanılan çiçek örnekleri	61
Şekil 4.9 : MABC ve ABC algoritmalarının IRIS veri kümesi üzerinde çalışmaları esnasında en yüksek “Uygunluk Değerinin” arama süreci esnasındaki değişimi	63
Şekil 5.1 : MIT-BIH veri tabanındaki EKG verilerinin wview programı ile görüntülenmesi	68
Şekil 5.2 : “wview” programındaki ayar pencereleri	68
Şekil 5.3 : MIT-BIH veri tabanındaki hatalı etiketleme örnekleri.....	69
Şekil 5.4 : “rdsamp” programı ile okunan, 100 numaralı MIT-BIH kaydının ilk 5 saniyelik kısmının görüntüsü	70
Şekil 5.5 : MIT-BIH veri tabanından kayıt 105’e ait EKG çizimleri	70
Şekil 5.6 : MIT-BIH veri tabanından kayıt 116’ya ait EKG çizimleri	71
Şekil 5.7 : MIT-BIH veri tabanından kayıt 209’a ait EKG çizimleri	72
Şekil 5.8 : MIT-BIH veri tabanı, 105 numaralı kayıtdan alınmış olan EKG işareti ..74	
Şekil 5.9 : Gürültülerden temizlenmiş, R-tepeleri belirlenmiş EKG işareti.....	74
Şekil 5.10 : Önemli noktaları belirlenmiş EKG işareti.....	75
Şekil 5.11 : 105 nolu MIT-BIH kaydındaki vurulara ait P-dalgası yükseklikleri ...	78
Şekil 5.12 : 105 nolu MIT-BIH kaydındaki vurulara ait R-tepesi yükseklikleri	79
Şekil 5.13 : 105 nolu MIT-BIH kaydındaki vurulara ait QRS genişlikleri	79
Şekil 5.14 : VK-KFold kümesindeki normalize edilmiş örnek vektörlerine ait öznitelik değerlerinin dağılımı	80

Şekil 5.15 : Normal bir kalp vurusuna ait örneklenmiş veriler	83
Şekil 5.16 : Normal bir kalp vurusuna ait ayırık Fourier katsayıları.....	83
Şekil 5.17 : Diverjansın öznitelik sayısı (boyut) ile değişimi.....	84
Şekil A.1 : MABC tabanlı sınıflayıcı için hazırlanan kullanıcı arayüzünün başlangıç görüntüsü.....	115
Şekil A.2 : Belirlenen kontrol parametreleri ile MABC algoritmasının bir defa çalıştırılması halinde elde edilen sonuçlar.....	116
Şekil A.3 : Sonuç dosyasında, çalışma sayfasında yer alan sonuç verilerinin yapısı	117
Şekil A.4 : Sonuç dosyasında “BasariYuzdeleri” sayfası.....	118





ELEKTROKARDİYOGRAM VERİLERİNİN İYİLEŞTİRİLMİŞ YAPAY ARI KOLONİSİ (MABC) ALGORİTMASI İLE ANALİZİ

ÖZET

Elektrokardiyogram (EKG) işaretlerinin analizi, çeşitli kalp hastalıklarının teşhisinde yaygın olarak kullanılmaktadır. Normalden farklı vurular olan aritmiler, ritm bozukluğu – düzensizliği anlamına gelmektedir. Hastane ortamında kısa süreli alınan kayıtlarda ortaya çıkmayan vuru düzensizliklerinin belirlenmesi amacıyla uzun dönemli elektrokardiyogram kayıtları alınmaktadır. Bir gün süreli bir kayıttaki yüz binin üzerinde vuru yer alabilmektedir. Bu şekilde çok yüksek sayıda vurunun kardiyologlar tarafından incelenmesi zahmetli ve zaman alıcı bir işlem olduğu için, bilgisayar destekli, otomatik EKG analizi yapan yöntemler geliştirilmiş ve geliştirilmeye devam etmektedir.

Normal ve aritmik vuru tiplerinin analizinde çeşitli sınıflama yöntemleri kullanılmaktadır. Sınıflama, öznitelik uzayındaki örnek vektörlerinden, sınıf etiketleri kümesine haritalama yapan, denetimli (supervised) bir sistemdir. Temel olarak, önceden tanımlı sınıf etiketlerini, örnek örüntülere atama işlemini gerçekleştirmektedir. Sınıflama yöntemleri arasında uzun zamandır kullanılmakta olan k-en yakın komşu (K-NN), en yakın ortalama (NMC) gibi çeşitli klasik yöntemler olduğu kadar, son yıllarda geliştirilen yeni yöntemlerin de olduğu görülmektedir. Bunlara örnek olarak, yapay sinir ağlarını esas alan öz-düzenlemeli harita (SOM) yöntemi ve karınca kolonisi optimizasyonu (ACO), yapay arı kolonisi (ABC) gibi doğadan esinlenen (nature inspired) yöntemler verilebilir.

Geliştirilen yöntemlerin performanslarının değerlendirilmesi amacıyla standart bir kaynak olarak çoğu defa Massachusetts Institute of Technology – Beth Israel Hospital (MIT-BIH) veri tabanı kullanılmaktadır. Tez kapsamında da kullanılan bu veri tabanı, PhysioNet tarafından açık kaynak olarak sunulmaktadır.

Yapay zekâ algoritmalarının bir alt kümesi olan doğadan esinlenen yöntemler, son birkaç on yıllık dönemde yaygın bir şekilde kullanılmaktadır. Bal arısı kolonilerinin toplumsal davranışlarından, özellikle gıda kaynağı araştırma ve yararlanma sürecinden esinlenerek, yapay arı kolonisi algoritması geliştirilmiştir. Gerçek hayattaki çok çeşitli mühendislik problemlerinin çözümünde kullanılan yapay arı kolonisi algoritmasının, sınıflama amaçlı öbekleme probleminde de kullanılmasına yönelik çalışmalar literatürde mevcuttur. Ancak bu tez çalışmasına kadar, ABC algoritmasının EKG verilerinin analizine yönelik kullanımı bulunmamaktadır.

Yapay arı kolonisi algoritmasının EKG verilerinin analizi ve kalp vurularının sınıflamasında kullanılırken iyileştirmelere ihtiyaç duyulduğu görülmüştür. Yapılan düzenlemelerle yöntemin performansı artırılmış, elde edilen bu yöntemle “İyileştirilmiş ABC (MABC)” adı verilmiştir.

Orijinal ABC algoritmasındaki en iyi çözümü arama sürecinde kullanılan karar ölçütü, MABC algoritmasında sınıflama başarımını arttıracak şekilde iyileştirilmiştir. Ayrıca, yerel iyi çözümlere takılmadan global en iyi çözümün bulunabilmesini sağlamak için kullanılan işçi arı - izci arı dönüşüm mekanizmasına, izci arı dönüşüm eşliği oranı

(SCTR) kontrol parametresi eklenmiştir. Bu sayede o ana kadar bulunmuş olan iyi çözümlerin hata ile terk edilmesi önlenmiştir.

Kullanılacak olan öznitelik kümesinin belirlenmesi amacıyla, elektrokardiyogram işaretinin başlıca özelliklerinden hareketle zaman domeninde 38, frekans domeninde 60 farklı öznitelik çıkartılmıştır. Bunların içinden en yüksek ayırt ediciliğe sahip olan öznitelikler, diverjans analizi yöntemi ile belirlenmiş ve toplam 15 öznitelik kullanılmıştır.

Tez çalışması kapsamında geliştirilen MABC algoritmasının performansının, eğitim aşamasında kullanılan örnek vektörlerinden bağımsız olduğu, K-Katlı çapraz doğrulama tekniği ile gösterilmiştir. Veri kümesindeki optimum öbek sayısı ise, geçerlilik indisleri analizi ile incelenmiş, sekiz sınıfa ait vuru tipleri için, sekiz öbek olarak doğrulanmıştır.

MABC algoritması tabanlı sınıflayıcı EKG verilerinin analizine uygulandığında, yüksek sınıflama başarımı hızlı bir şekilde elde edilmiştir. Tez çalışmasında MIT-BIH veri tabanındaki “N, j, V, F, f, A, a ve R” tipi olmak üzere toplam sekiz farklı vuru tipi sınıflandırılmıştır. Sınıflama başarımının artırılması amacıyla, zaman ve frekans domenine ait öznitelikler birlikte kullanılmıştır.

Geliştirilen yöntem, EKG verileri analizi dışında, çeşitli diğer veri kümelerine de uygulanmış, olumlu sonuçlar elde edilmiştir. Bu şekilde tez çalışması kapsamında, hızlı çalışan ve yüksek sınıflama başarımına sahip bir genel amaçlı sınıflayıcı elde edilmiştir.

Geliştirilmiş olan MABC algoritmasının, diğer araştırmacıların kolayca kullanmasını sağlamak amacıyla bir grafik kullanıcı arabirimi (Graphic User Interface – GUI) hazırlanmıştır.

ANALYSIS OF ELECTROCARDIOGRAM DATA BY USING MODIFIED ARTIFICIAL BEE COLONY (MABC) ALGORITHM

SUMMARY

One of the most commonly used tools to diagnose the heart diseases is electrocardiogram (ECG signal) analysis. An abnormal heart rhythm is called an arrhythmia. Arrhythmias are caused by problems in heart's electrical system and are indication of some heart diseases. Some of the arrhythmias appear on the electrocardiogram under certain conditions such as effort, excitement etc. during the daily life. In order to detect those rare arrhythmic events, minimum 24 hours long term records must be analysed. However, manual analysis of long records, visual inspection by a cardiologist is too much time consuming and is not practical. Heart beats can have different characteristics on different patients. Because of that reason, there may be an overlap between heart beat types (classes). As a result, arrhythmia classification becomes harder. To make easier and accurate analysis, computer aided heart beat classification system development is an active research topic.

ECG signal consists of certain parts, such as P-wave, QRS complex and T-wave. Characteristics of those waves, such as morphology, area, length, amplitude and time intervals, are different on normal beats and arrhythmic beats. Those characteristics are called time space features. In addition to them, there are some other characteristics of ECG signal in frequency domain, called frequency space features. Some of the arrhythmic beats can be classified by using only time domain features, however some other type beats need both time and frequency domain features.

Using more number of features can help to have higher classification accuracy. However, an increasing number of dimensions in feature space would results in much higher computational time. In order to have the optimum number of features, feature reduction methods can be implemented, such as divergence analysis. In this thesis, ECG signals are taken from the MIT-BIH arrhythmia database, which is developed by the Massachusetts Institute of Technology and ECG recordings obtained by Beth Israel Hospital Arrhythmia Laboratory. In the beginning of the electrocardiogram signal analysis process, a software code is developed to calculate the time domain features. Total 38 time domain features are reduced to 7 features by using divergence analysis. As a second step, discrete Fourier transform (DFT) of a window of data which has 256 samples around the R-peak is used to calculate the DFT coefficients. About 60 of them have non-zero values. They are reduced to 8 frequency domain features by using divergence analysis. In this study, total 15 time and frequency domain features are used for the classification of the heart beat types. Those feature vectors represents the hearts beats to be analysed and classified. Before the classification process, data set which consists of feature vectors should be divided into two subsets, as training set and test set. Training set is used to train the classification algorithm, while test set is used to measure the performance of the classifier. In order to verify that the performance of classification algorithm is independent of the training set, cross

validation techniques can be used. In this thesis, K-Fold cross validation method is used.

Classification is a supervised system, which is a mapping method of assigning pre-defined labels onto objects in feature space. A similar term clustering is an unsupervised learning method that autonomously partitions the data into clusters. Most of the classification methods use clustering as a first stage of classification process.

Various classification methods are used in the literature. In addition to the classical methods such as nearest mean classifier, k-means, k-nearest neighbor, neural network classifier etc, there are many new proposed nature inspired algorithms such as ant colony optimization (ACO) and artificial bee colony (ABC) algorithm.

Nearest mean classifier (NMC) and neural network (NN) classifier are widely used classical methods in data classification problem. NMC is a fast algorithm which uses the similarity between patterns to make the classification. A self-organizing map (SOM) is a type of artificial neural network (ANN). SOM network is trained using unsupervised learning to produce a low-dimensional representation of the input space of the training samples, which is called a map. By using the class information of output nodes in the map, classification of unknown patterns can be performed. K-nearest neighbour (KNN) is fast and simple classification method. It uses all training samples and makes the classification of test samples, based on majority vote of its neighbours in the training set, considering similarity measure. One of the most commonly used distance metric is Euclidean distance.

Nature inspired algorithms may imitate a process in nature or may use an approach simulating some intelligent behaviors of social animal groups in nature such as bird flocks, ant or bee colonies. These kinds of algorithms are called swarm intelligence (SI) algorithms.

Ant colony optimization algorithm is a meta-heuristic, nature inspired method. This algorithm is a multi-agent system, inspired by the collective behavior of ant colonies. It makes the clustering by simulating the interactions between ants through pheromones. By choosing the shortest distance between the nodes, clusters are formed. After that, K-NN can be used for classification phase.

ABC algorithm was first proposed by Karaboga, inspired by foraging behavior of honey bee colony. This algorithm has been used in many application areas as well as data clustering. However, until this thesis, there is no published study about using ABC on electrocardiogram analysis.

ABC is a simple and flexible algorithm, uses few control parameters. It can converge to the global best solution because of its random search mechanism. Honey-bee colony's foraging behaviour, learning, memorizing and information sharing characteristics are the main reasons of the success of this algorithm.

In real life, forager bees randomly search food locations. They evaluate the nectar amount in a food source, memorize it and share this location and food quality information with other colony members in the hive, by using a special communication method.

ABC algorithm is developed based on a model inspired by foraging behaviour of honey bee swarm. Possible solutions of a problem in artificial bee colony are represented by food source locations around the hive in real life. Nectar amount in that

food source represents the fitness value of that solution. There are three types of foraging bees in ABC: scout bees, employed bees and onlooker bees.

Scout bees search new solutions in the solution space randomly, without using any pre-information. It becomes an employed bee when they start to work on a location. Employed bees calculate the fitness of the solution, memorize it and share it with onlooker bees. Number of the employed bees in the hive, represents the number of solutions in ABC algorithm. Onlooker bees receive the information from employed bees; select a location to exploit, depending of the fitness values of the locations. The solutions with higher fitness value are visited by more onlooker bees, because of the probabilistic model in ABC. Each onlooker bee searches new locations near the selected employed bee. Both employed and onlooker bees uses greedy selection mechanism to decide which location is better among the previous and next locations.

If there is no improvement on the fitness of a location during the last “limit” times iterations, the position will be abandoned. That employed bee becomes a scout bee, starts search from a random location. This process enables the algorithm to search global best location, by avoiding converge to local minima. The process explained above is repeated by a pre-defined iteration cycles, called maximum cycle number (MCN).

In classification problem, ABC algorithm performs the clustering during the training phase and provides the best cluster center locations. In the test phase, classification is done by assigning class label to the test sample, according to the shortest Euclidean distance between that sample and the cluster centers.

It has been seen that the algorithm should be improved in the studies conducted within the scope of the thesis. For that purpose, below listed modifications are done. New modified version of algorithm is called Modified ABC (MABC) algorithm.

As an objective function in the fitness value, maximizing the classification accuracy is used, instead of minimizing the sum of distances of the training samples to their cluster centres. Otherwise, when there are much more number of samples in one class than other classes, more than one cluster center approaches towards that class. This problem results in drop of classification accuracy.

A new control parameter called SCTR (Score Conversion Threshold Ratio) is introduced in the algorithm. Otherwise, good solutions, even the best solution, can be abandoned if there is no improvement in the last "limit" time iterations.

In the initialisation stage, the algorithm starts search process from randomly selected training sample vectors instead of random locations. Also, in search space border definitions have been done.

High classification accuracy has been achieved in a short time, when the MABC algorithm is applied on electrocardiogram data analysis. Examined MIT-BIH data set consists of total eight heart beat types “N, j, V, F, f, A, a and R”.

Within the scope of the thesis study, a training data set with a total of 2834 sample vectors was used. A total of 8735 test sample vectors belonging to eight different classes were classified using the obtained cluster center coordinates. The classification performance achieved by the MABC algorithm is 97.18%. With this result, MABC algorithm is among the best performing algorithms.

MABC algorithm has also been applied to other data sets, other than ECG data. High classification accuracy and successful results have been obtained. In this way, within

the scope of the thesis study, a general purpose classifier has been achieved which has a high classification performance and fast working.

A graphical user interface (GUI) has been prepared so that the developed MABC algorithm can be easily used by other researchers.

In future studies, it is advisable to develop a hybrid classification method which can be used together with the MABC algorithm and other classification methods.



1. GİRİŞ

1.1 Tezin Amacı

Bu tezin amacı, elektrokardiyogram (EKG) verilerinin analizinde Yapay Arı Kolonisi (ABC) algoritması kullanılarak, yüksek doğruluk (accuracy) ve duyarlılıkta (sensitivity) sonuç verecek, az sayıda kontrol parametresine sahip, yerel iyi sonuçlara takılmadan en iyi sonuca hızlı bir şekilde yakınsayan bir sınıflama yöntemi geliştirmektir. Bu kapsamda, sınıflama problemlerinde optimum öbek merkezlerinin belirlenmesi için kullanılabilen ABC algoritmasının performansının iyileştirilmesi hedeflenmiştir. Sınıflama başarımının artırılması için, zaman domeni ve frekans domeni öznitelikleri birlikte kullanılmıştır.

1.2 Literatür Araştırması

Kalp hastalıkları başta gelen ölüm nedenlerindendir. Elektrokardiyografik işaretlerin dalga şekilleri ve vuru hızları kalp hastalıklarının belirlenmesinde yaygın olarak kullanılırlar. Normalden farklı vuruların belirlenmesi ve sınıflandırılması, kalp hastalığının teşhisi açısından büyük önem taşımaktadır. Ritm bozukluğu – düzensizliği anlamına gelen aritmiler, hastanın içinde bulunduğu yorgunluk, heyecan vb şartlara bağlı olarak, gün içinde farklı zamanlarda ortaya çıkmaktadır. Bu tür kalp hastalıklarının belirlenmesi amacıyla, uzun dönemli elektrokardiyogram kaydı yapan cihazlar en az 24 saat süreyle hastaya takılı olarak bütün kalp vurularını kaydetmektedir. Dakikada yaklaşık 60 - 90 kalp atışı olduğu durumda, bir günde toplam 86,400 – 129,600 vuru gibi yüksek sayıda kalp vurusunun uzman kardiyolog doktor tarafından incelenmesi çok zaman alıcı ve hatalara açık bir uygulamadır. Bu nedenle EKG verilerinin otomatik olarak analizi ve kalp vurularının sınıflaması ile ilgili çok miktarda çalışmalar yapılmış ve yapılmaya devam etmektedir. Bunlardan bazıları, yayın tarihi sırasıyla aşağıda verilmiştir: EKG işaretinin izdüğümsel ve dinamik özniteliklerini kullanarak kalp vurusu sınıflaması Chen ve diğ. (2017), deterministik öğrenme yöntemi ile EKG vuru sınıflaması Dong ve diğ. (2017), EKG segmentasyon algoritmalarının karşılaştırılması Beraza ve Romero (2017), blok

tabanlı sinir ağı ve parçacık sürü optimizasyonu ile EKG sınıflaması Shadmand ve Mashoufi (2016), istatistiksel ve karışım modelli EKG sınıflaması Afkhami ve diğ. (2016), radyal tabanlı fonksiyon sinir ağı (RBFNN) ile EKG vuru sınıflaması Mateo ve diğ. (2016), EKG işaretlerinin lineer ve lineer olmayan özniteliklerini birlikte kullanarak aritmi tanıma ve sınıflandırma Elhaj ve diğ. (2016), çift ağaç kompleks dalgacık öznitelikleri kullanılarak otomatik EKG aritmi sınıflaması Thomas ve diğ. (2015), otoregresif modelleme ile ventriküler fibrilasyon esnasında EKG işareti sınıflaması Padmavathi ve Ramakrishna (2015), yapay arı kolonisi algoritması tabanlı EKG kalp vuru sınıflama yöntemi Dilmac ve Korurek (2015), EKG işaretinin dinamik modeli ve genetik-bulanık sistem ile EKG sınıflaması Vafaie ve diğ. (2014), makine öğrenimi yaklaşımı ile birden fazla kademeli EKG işareti sınıflaması Li ve diğ. (2014), maliyete duyarlı sınıflayıcı ile EKG vuru sınıflaması Zidelmal ve diğ. (2013), uzmanların karışımı ve negatif ilişkili öğrenme tabanlı modüler yapay sinir ağı ile EKG aritmi sınıflandırma Javadi ve diğ. (2013), EKG öbekleme ve sınıflama yaklaşımı Abawajy ve diğ. (2013), optimum-yol ormanı tabanlı EKG sınıflama Luz ve diğ. (2013), bulanık C-ortalamalar ve hibrit karınca kolonisi optimizasyonu tabanlı bir metot ile ECG vuru sınıflama Doğan ve Korürek (2012), temel eleman analizi ile EKG işaret analizi Martis ve diğ. (2012), genetik tümleşik sınıflayıcılar grubu ile kalp vuruları sınıflama Osowski ve diğ. (2011), kişiselleştirilmiş uzun-dönemli EKG sınıflama – sistematik yaklaşım Kiranyaz ve diğ. (2011), bulanık C-ortalamalar ile kalp vuruları sınıflama Yeh ve diğ. (2010), lokal fraktal boyutlu sınıflayıcı Mishra ve Raghav (2010), karınca kolonisi optimizasyonu yöntemi tabanlı aritmi sınıflama tekniği Korurek ve Nizam (2010), birleştirilmiş sinir ağı modeli ile EKG vuru sınıflama Guler ve Ubeyli (2005), EKG zaman serilerinde otomatik örüntü tanıma Sternickel (2002), hibrit sinir ağı ile EKG vuru sınıflama Dokur ve Ölmez (2001), bilgi tabanlı EKG yorumlama Kundu ve diğ. (2000), EKG analizi için sinir ağı sınıflayıcılar ve RBF tabanlı hibrit mimarilerin tasarımı Silipo ve diğ. (1999), sinir ağları kullanılarak EKG dalga şekillerinin belirlenmesi (Dokur ve diğ, 1997).

Bu çalışmalarda en sık kullanılan iki terim öbekleme ve sınıflama olarak dikkat çekmektedir. Denetimsiz (unsupervised) bir öğrenme yöntemi olan öbekleme, bir veri kümesinin içindeki birbirine benzer örüntüleri, belirli benzerlik ölçütlerine göre, öznitelik domeninde öbeklere ayırma algoritması olarak tanımlanabilir. Sınıflama, öbeklemeye benzer bir terim olmakla birlikte farklı bir anlama sahiptir. Sınıflama,

öznitelik domeninden bir etiket kümesine haritalama yapan, denetimli (supervised) bir sistemdir. Temel olarak, önceden tanımlı sınıf etiketlerini örnek örüntülere atama işlemini gerçekleştirmektedir (Fielding, 2007; Jain, 2010).

Yukarıda belirtilen yayınlar incelendiğinde, ilk dönemlerde geliştirilen çeşitli sınıflama tekniklerinin, son yıllarda birlikte kullanılarak hibrit yöntemlerin oluşturulduğu görülmektedir. Ayrıca, doğadan esinlenerek geliştirilmiş yöntemler kullanılarak hazırlanan algoritmalar da giderek artan miktarda literatürde yer almaktadır. Sadece EKG vurularının sınıflanması değil, aynı zamanda bu amaçla kullanılabilecek özniteliklerin çıkarılması, en yüksek ayırt ediciliğe sahip öznitelik kümesinin oluşturulması konularında da çalışmalar yapılmaktadır. EKG işaretindeki dalga şekillerinin konumlarının hassas bir şekilde belirlenmesi de benzer şekilde üzerinde çalışılan konulardandır.

EKG verilerinin analizi ve sınıflanması konusunda yapılan çalışmalarda büyük ölçüde MIT-BIH veri tabanı kullanılmaktadır (Url-1). Bazı çalışmalarda bu veri tabanında mevcut olan daha fazla sayıda kalp vuru tipinin sınıflandırılması hedeflenmiştir. Diğer bazı çalışmalarda ise vuru tipleri AAMI (Association for the Advancement of Medical Instrumentation) standartlarına uygun olarak gruplandırılarak sınıflandırılmaktadır. Buna göre MIT-BIH veri tabanındaki “N, L, R, e ve j” tipi vurular, “N” olarak; “A, a, x ve J” tipi vurular “S” olarak; “V ve E” tipi vurular “V” olarak, “f ve Q” tipi vurular ise “Q” olarak gruplandırılmaktadır. “F” tipi vurular yine “F” olarak adlandırılır. Bu tez çalışmasında, her bir vuru tipi, kendi orijinal ismi ile sınıflandırma işleminde kullanılmıştır.

Bazı yöntemlerin eğitilmesi ve test edilmesi amacıyla kullanılan örnek sayısı 8-10 bin adet mertebesinde iken, bazı yeni çalışmalarda incelenen örnek sayısının 100 bin mertebesine kadar çıktığı görülmektedir. Ancak sayının artmasının temel sebebi, MIT-BIH veri tabanında çok sayıda bulunan normal vurulara ait örneklerin çoğunluğunun çalışmaya dâhil edilmiş olmasıdır. Çalışmaların büyük bir bölümünde, geliştirilen sınıflama yönteminin performansının diğer yöntemlerle karşılaştırılabilmesi amacıyla, ortak bir veri tabanı olarak MIT-BIH veri tabanı kullanılmıştır. Ancak bu çalışmaların hemen hepsinde farklı hastalardan alınmış EKG kayıtları ve farklı sayılardaki vuru örnekleri kullanıldığı görülmektedir. Bu nedenle, yöntemlerin performanslarının karşılaştırılmalarının sağlıklı bir şekilde yapılması zorlaşmaktadır. Daha güvenilir karşılaştırmalar yapılabilmesi için standart veri kümelerinin kullanılması gerektiği

açıktır. Bu amaçla, aynı veri tabanından alınarak performans değerlendirme ölçütü olacak şekilde, farklı zorluk seviyelerinde standart veri kümeleri oluşturulmalıdır.

Yapılan çalışmalarda, geliştirilen sınıflama yönteminin performansının, seçilen eğitim kümesindeki örneklerden bağımsız olduğunun doğrulanması amacıyla, K-katlı çapraz doğrulama tekniğinin kullanıldığı görülmektedir. Bu tez çalışmasında da kullanılan tekniğin detayları, Bölüm 3.2.4’de açıklanmıştır.

Yapay zekâ algoritmalarının bir alt kümesi olan “doğadan esinlenen (nature inspired)” algoritmalar, son birkaç on yıllık dönemde sınıflama ve diğer çeşitli problemlerin çözümünde yaygın bir şekilde kullanılmaktadır (Karaboğa, 2004; Bolaji ve diğ., 2013). Evrimsel algoritmalar ve sürü zekâsı algoritmaları gibi teknikler, doğadan esinlenen yöntemlerden olup, gerçek hayattaki çeşitli problemlerin çözümünde giderek artan bir şekilde kullanılmaktadır (Karaboğa ve diğ., 2012).

Evrimsel algoritmalar, belli bir gruptaki bireylerin özelliklerini başlangıç değeri olarak alır ve en uygun çözümü arar. Bu bireyler, olası çözümleri simgeler. Arama sürecinde, önceki nesilden yeni nesildeki bireye genetik miras aktarımı, mutasyon ve en uygun özelliklere sahip bireylerin hayatını devam ettirebilmesi gibi mekanizmalar kullanılır (Goldberg, 1989).

Sürü terimi, toplumsal davranışlar gösteren karıncalar, arılar, balıklar, kuşlar gibi hayvanların oluşturduğu toplulukları ifade eder. Sürü zekâsı (SI), bir merkezi kontrol olmaksızın, sürüdeki bireylerin kendi kendisine organize olması olarak tanımlanabilir. Sürüdeki bireylerin birbirleri arasında ve çevreleri ile gerçekleştirdikleri etkileşimler sürüye ait bir zekânın varlığını göstermektedir. Gıda bulmak, yuva inşa etmek, yuva içinde iş bölümü yapmak gibi çeşitli görevleri gerçekleştirirken gösterdikleri davranışlar, bilim insanlarına mühendislik, ekonomi, biyo-informatik ve daha pek çok alanlarda problemlerin çözümü için esin kaynağı olmuştur (Karaboğa ve diğ., 2012). Karınca kolonilerinden esinlenerek geliştirilen karınca kolonisi optimizasyonu (ACO) Dorigo ve diğ. (1991) ve kuş sürülerinden esinlenerek geliştirilen parçacık sürü optimizasyonu (PSO) Kennedy ve Eberhard (1995) algoritmaları bunlardan ikisidir.

Takip eden yıllarda, bir başka sosyal hayvan türü olan arı kolonilerini inceleyen çalışmalar görülmektedir. Arı sistemi (Bee system) Lucic ve Teodorovic (2001), arı kovanı algoritması (Bee hive) Wedde ve diğ. (2004), sanal arı algoritması (Virtual bee) Yang (2005) bu konudaki çalışmalardan bazılarıdır.

İlerleyen yıllarda, bal arısı kolonilerinin toplumsal davranışlarından, özellikle gıda kaynağı araştırma ve yararlanma sürecinden esinlenerek, yapay arı kolonisi (ABC) algoritması geliştirilmiştir. Bu konudaki ilk çalışma Karaboğa tarafından gerçekleştirilmiştir (Karaboga, 2005). Bu tarihten sonra Karaboğa ve diğer araştırmacılar, ABC algoritmasını çeşitli uygulama alanlarında kullanmıştır. Bunlardan bazıları tarih sırasıyla aşağıda belirtilmiştir. Sayısal fonksiyon optimizasyonunda ABC algoritması kullanımı Basturk ve Karaboga (2006), yapay sinir ağlarının eğitiminde ABC algoritmasının kullanılması Karaboga ve Akay (2007), barajların yapısal analizi ve parametre belirlenmesi Kang ve diğ. (2009), metro hatlarının planlanması Yao ve diğ. (2010), tarımsal ürünlerin yetiştirilmesinde hastalıkların belirlenmesi ve hastalık yönetimi Babu ve Rao (2010), kablosuz algılayıcı ağlarında enerji tüketimini minimize edecek yerleşim planlaması Karaboga ve diğ. (2010), haberleşmede frekans tahisi problemi Chaves-Gonzales ve diğ. (2010), kompozit yapıların tasarım optimizasyonunda ABC algoritması kullanılması Omkar ve diğ. (2011), SAR uydu görüntüleri segmentasyonunda ABC algoritması kullanımı Ma ve diğ. (2011), araç rotası planlama probleminde ABC algoritması kullanımı Szeto ve diğ. (2012), video konferans görüşmelerinde hareket kestirimi için blok uyumu algoritması Cuevas ve diğ. (2013), EEG/ERP için adaptif filtre Ahirwal ve diğ. (2014), ECG verilerinin analizi Dilmac ve Korurek (2015), DNA mikrodizilerinin sınıflanması Garro ve diğ. (2016), petrol kuyularının konumlarının optimizasyonu (Nozohour-leilabady ve Fazelabdolabadi, 2016).

Yapay arı kolonisi algoritmasının sınıflama amaçlı öbekleme probleminde de kullanılmasına yönelik çalışmalar literatürde mevcuttur, (Zhang ve diğ. 2010; Karaboga ve Ozturk, 2011; Yan ve diğ. 2012). Bununla birlikte ABC algoritmasının elektrokardiyogram verilerinin analizi ve kalp vurusu sınıflama problemine uygulandığı bir çalışma olmadığı görülmüştür. Doktora tez çalışması kapsamında, ABC algoritması elektrokardiyogram verilerinin analizine uygulanmış ve gerekli bazı iyileştirmeler yapılarak kalp vurularının sınıflanması konusunda tez kapsamında çeşitli makale ve bildiriler yayınlanmıştır.

1.3 Tezin Katkıları ve İzlenen Yol

Son yıllarda geliştirilmiş ve çeşitli sınıflama problemlerine uygulanmış olan Yapay Arı Kolonisi algoritmasının EKG verilerinin analizi ve kalp vurularının sınıflamasında

kullanılırken bazı iyileştirmelere ihtiyaç duyulduğu görülmüştür. Bu tez çalışması kapsamında ABC algoritması tabanlı bir sınıflayıcı geliştirilmiştir. Yapılan düzenlemelerle yöntemin performansı arttırılmış, elde edilen bu yönteme “İyileştirilmiş ABC (MABC)” adı verilmiştir. MABC algoritması tabanlı sınıflayıcı EKG verilerinin analizine uygulandığında, yüksek sınıflama başarımı hızlı bir şekilde elde edilmiştir. Geliştirilen yöntem, EKG verileri analizi dışında, çeşitli diğer veri kümelerine de uygulanmış, olumlu sonuçlar elde edilmiştir. Bu şekilde tez çalışması kapsamında, hızlı çalışan ve yüksek sınıflama başarımına sahip bir genel amaçlı sınıflayıcı elde edilmiştir.

Tez kapsamında, orijinal algoritma üzerinde aşağıda verilen iyileştirmeler gerçekleştirilmiştir. ABC algoritması arama sürecinde yerel iyi sonuçlara takılmamak için, belirli bir arama adımından sonra iyileşme olmamışsa, üzerinde çalışılan çözümü terk etmekte ve rasgele bir konumdan aramaya devam etmektedir. Ancak terk edilen çözümlerin bazı durumlarda iyi çözümler olabildiği, bu nedenle de sınıflama başarımının düşebildiği görülmüştür. Bu problemi gidermek amacıyla “izci arı dönüşüm eşik seviyesi (SCTR)” adında yeni bir kontrol parametresi algoritmaya eklenerek iyi çözümlerin hatalı olarak unutulması önlenmiştir.

Orijinal ABC algoritmasında uygunluk fonksiyonu olarak, örnek vektörlerinin öbek merkezlerine olan Öklid uzaklıklarının toplamı kullanılmaktadır. Bu nedenle, farklı sınıflara ait dengeli sayıda örnek içermeyen veri kümelerinin üzerinde çalışılması halinde, birden fazla öbek merkezi çok sayıda örnek içeren öbeğe yaklaşmakta, bu da ilerleyen arama adımlarında sınıflama başarımını düşürmektedir. Bu problemi çözmek için, arama sürecinde karar ölçütü olarak, toplam Öklid uzaklığı yerine, sınıflama başarımlarının oranı kullanılmıştır.

Sınıflama problemine özel olarak, sınıfların öbek merkezlerinin o sınıfa ait örnek vektörleri civarında bulunacağı dikkate alınarak, arama sürecine rasgele konumlardan başlamak yerine, rasgele seçilen birer örnek vektörünün konumundan başlanılmıştır. Bu uygulama, en iyi konuma daha hızlı bir şekilde ulaşmayı sağlamaktadır.

İyileştirilmiş ABC sınıflama yöntemi ile zaman domeni ve frekans domeni öznitelikleri birlikte kullanılarak toplam sekiz farklı kalp vuru tipinin yüksek sınıflama başarımı ile sınıflanması gerçekleştirilmiştir.

Bu tez çalışmasında, PhysioNet tarafından açık kaynak olarak sunulan, Massachusetts Institute of Technology – Beth Israel Hospital (MIT-BIH) veri tabanından alınan EKG verileri üzerinde çalışılmıştır. Tez çalışmasının başında, Pan-Tompkins algoritması tabanlı bir yazılım ile EKG verileri gürültülerden arındırılmıştır. PhysioNet'in sağlamış olduğu “ecgpuwave, physio toolkit/rdann, rdsamp” yazılım paketleri kullanılarak P, Q-R-S ve T dalgalarının tepe noktalarının konumları, başlangıç ve bitiş noktaları ile genlik değerleri belirlenmiştir. Bu tez çalışması kapsamında hazırlanan bir yazılım ile zaman domeninde 38 farklı öznitelik değerleri hesaplanmıştır. Bunlara ek olarak frekans domeninde öznitelik çıkarma işlemleri yapılarak 128 adet öznitelik değerleri belirlenmiştir. Daha sonra, üzerinde çalışılacak aritmi tiplerine göre en yüksek ayırt ediciliğe sahip öznitelikleri belirlemek için diverjans analizi yapılmıştır. Bu analizin sonucunda kullanılan öznitelik sayısı, zaman domeninden yedi, frekans domeninden sekiz olmak üzere, toplam on beş özniteliğe indirilmiştir.

Sınıflanacak kalp vurularının öznitelik değerlerinin bulunduğu öznitelik matrisinde, her bir kalp vurusunun hangi MIT-BIH kayıt dosyasından alındığı, kaydın kaçınıcı dakika – saniyesindeki vuruya ait olduğu gibi bilgiler de kullanılmış, böylece sınıflanan her bir vurunun izlenebilirliği ve doğrulanabilirliği sağlanmıştır.

Tez çalışması kapsamında, incelenen sınıflama algoritmalarının kontrol parametrelerinin farklı değerleri ve kombinasyonları kullanılarak sınıflama başarımları incelenmiştir.

Kullanılan eğitim ve test vektörlerinin seçimi sınıflama başarımını etkileyebilmektedir. Geliştirilen algoritmanın performansının seçilen örnek vektörlerinden bağımsız olduğunun görülmesi amacıyla K-Katlı çapraz doğrulama (K-Fold cross validation) tekniği kullanılmıştır. Veri kümesindeki optimum öbek sayısını belirleyen geçerlilik indisleri incelenmiştir. Son olarak, elde edilen sınıflama sonuçları, klasik ve doğadan esinlenen sınıflama yöntemlerinin sonuçları ile karşılaştırılarak değerlendirilmiştir.

Tezin ikinci bölümünde elektrokardiyogram işaretinin oluşumu ve başlıca bölümleri anlatılmış, normal ve aritmik kalp vuru tipleri açıklanmıştır. Üçüncü bölümde EKG verilerinin analizi için gerçekleştirilen ön işlemler, öznitelik çıkartma ve öznitelik sayısının diverjans analizi ile indirgenmesi işlemleri anlatılmıştır. Öbekleme ve sınıflama problemleri açıklanmış, sınıflamada kullanılan çeşitli yöntemler hakkında

bilgi verilmiştir. Yine bu bölümde K-katlı çapraz doğrulama tekniği ve sınıflama performans ölçütleri ile optimum öbek sayısı geçerlilik indisleri açıklanmıştır. Dördüncü bölümde ABC algoritması ve sınıflamaya uygulanması verilmiştir. Tez çalışması kapsamında ABC algoritması üzerinde yapılan geliştirme ve iyileştirmeler sonucu elde edilen iyileştirilmiş ABC (MABC) algoritması sunulmuştur. Son olarak bu bölümde MABC algoritması ile EKG verilerinin analizinin nasıl gerçekleştirileceği açıklanmıştır.

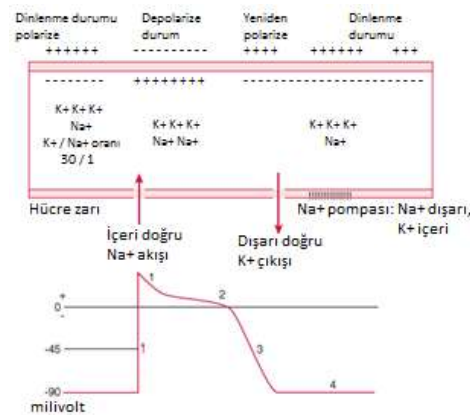
Beşinci bölümde, MIT-BIH veri tabanının yapısı, bu veri tabanından alınan örneklerin detayları verilmiştir. Kullanılan veri kümeleri ile tüm özniteliklerin hesaplanması ve kullanılacak yüksek ayırt ediciliğe sahip olan özniteliklerin seçilmesi anlatılmıştır. Tez çalışması kapsamında geliştirilen MABC algoritması tabanlı sınıflayıcı ile gerçekleştirilen sınıflandırma işlemlerine ait bilgisayar benzetim sonuçları sunulmuştur. Bu sonuçların, diğer yöntemlerle elde edilen sınıflama sonuçları ile karşılaştırılması ve değerlendirmeler verilmiştir. Son bölüm olan altıncı bölümde ise, yapılan çalışmalardan elde edilen çıkarımlar ve öneriler sunulmuştur.

Bu tez çalışmasında elde edilen sonuçlar, SCI'ya giren uluslararası bir dergide Dilmac ve Korurek (2015), ve ayrıca bir başka uluslararası dergide Dilmac ve diğ. (2014) yayınlanmıştır. Çeşitli ulusal ve uluslararası konferanslarda sunulmuştur (Dilmac ve Olmez, 2017; Dilmac ve Korurek, 2014; Dilmac ve Korürek 2013a; Dilmac ve Korurek, 2013b; Dilmac ve diğ. 2013c).

2. ELEKTROKARDİYOGRAM (EKG) VE ARİTMİLER

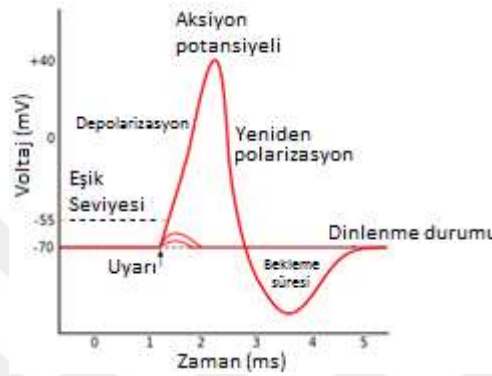
Vücuttaki hücrelerde elektrokimyasal olayların sonucu iyonik akımlar oluşur. Bu akımlar elektrotlar yardımıyla algılanır ve işaret işleme aşamalarından geçirildikten sonra çeşitli hastalıkların teşhisinde kullanılabilir. Kalbin elektriksel aktivitelerinin kaydına elektrokardiyogram, kayıt cihazına elektrokardiyograf, bu kayıt işlemine ise elektrokardiyografi adı verilir. Elektrokardiyogram'ın analizi kalp vurularındaki bozuklukların tanısında kullanılır.

Vücuttaki hücrelerin içi ve hücre zarının hemen dışındaki Na^+ , K^+ ve Cl^- iyonlarının farklı yoğunlukta olmasından ötürü hücrenin içi dışına göre negatif bir gerilim değerine sahiptir. Hücre zarının içi ve dışı arasında, iyon konsantrasyon farkıyla belirlenen bu gerilime dinlenme potansiyeli denilir, -90mV mertebesinde bir değere sahiptir. Dışarıdan uygulanan bir elektriksel uyarı, hücre zarının farklı iyonlara karşı olan geçirgenliğini değiştirir ve Na^+ iyonları hızla hücre içine akarak hücre içinin gerilim seviyesini $+20\text{mV}$ seviyesine çıkarır. Bu potansiyel değişimine aksiyon potansiyeli, olaya ise depolarizasyon adı verilir. Kararlı hale ulaştıktan sonra, hücre zarı tekrar seçici iletken duruma geçer, Sodyum Pompası adı verilen aktif bir iyon pompası aracılığı ile hücre tekrar -90mV 'luk dinlenme durumuna döner. Bu olaya da repolarizasyon adı verilir. Dış elektriksel uyarı ve iyon pompaları etkisi ile aksiyon potansiyelinin oluşumu Şekil 2.1'de görülmektedir.



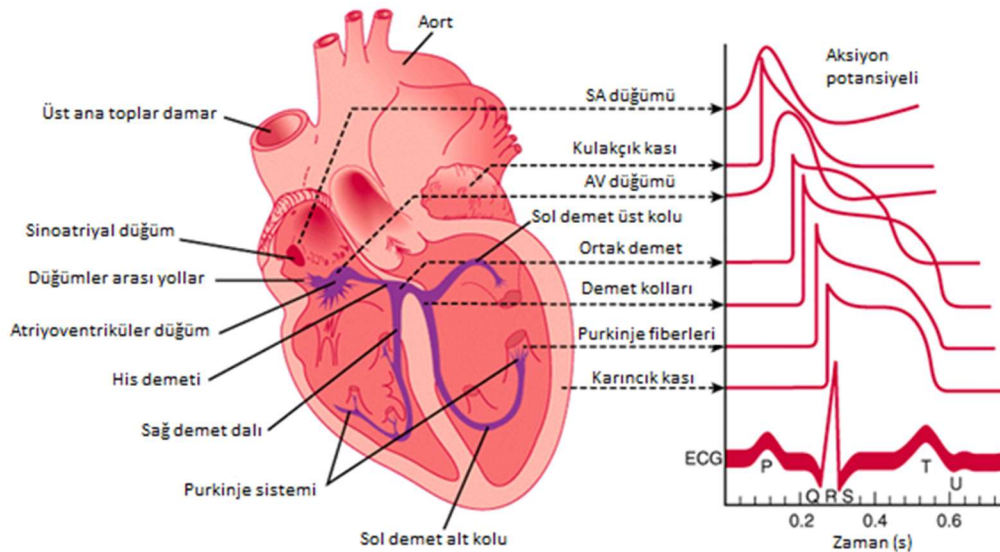
Şekil 2.1 : Dış elektriksel uyarı ve iyon pompaları etkisi ile aksiyon potansiyelinin oluşumu (Khan, 2003).

Bu şekilde oluşan toplam aktivasyon süresi, kas ve sinir hücrelerinde 1-2 milisaniye mertebelerinde iken, kalp kaslarında genellikle 150-300 milisaniye kadardır (Yazgan ve Korürek, 1996). Hücre, bir aksiyon potansiyelinin oluşmasından hemen sonra belirli bir süre boyunca, tekrar gelen uyarılara cevap vermez. Bu süreye bekleme süresi (refractory period) adı verilir. Bu bekleme süresi, aynı hücrenin ters yönde tekrar uyarılmasını önleyerek, uyarının tek yönde ilerlemesini sağlar. Sinir hücrelerinde bu süre 1 ms kadardır. Aksiyon potansiyelinin aşamaları Şekil 2.2’de görülmektedir.



Şekil 2.2 : Aksiyon potansiyelinin aşamaları (Sandoe ve Sigurd, 1991).

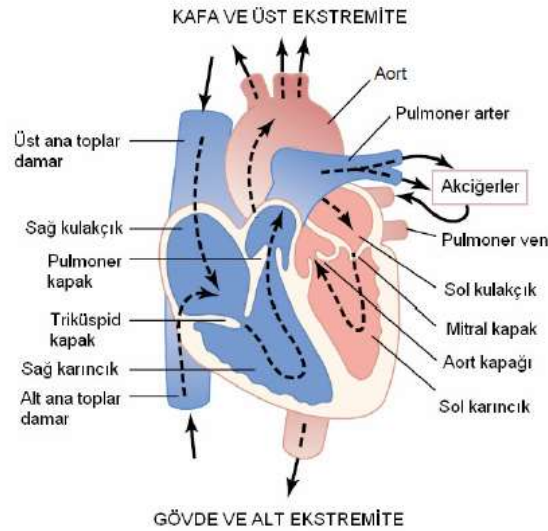
Kalpdeki özelleşmiş uyarıcı ve iletiler hücreler üzerinden kalbin tümüne yayılan aksiyon potansiyellerinin toplamının zaman ekseninde kaydedilmiş haline elektrokardiyogram denilir. Kalbin farklı bölümlerindeki aksiyon potansiyellerinin değişimi ve bunların toplamı olan EKG işareti Şekil 2.3’te görülmektedir.



Şekil 2.3 : Kalbin çeşitli bölümlerindeki aksiyon potansiyelleri ve bunların toplamından EKG işaretinin oluşumu (McPhee ve Hammer, 2010).

2.1 Kalbin Yapısı

Kalp, kan dolaşım sistemi içinde temiz ve kirli kanın tüm vücutta dolaşımını sağlayan bir elektromotor kuvvet kaynağı olan bir organdır. Göğüs kafesi içinde sol tarafta, ters çevrilmiş bir koni şeklindedir. Sol ve sağ tarafında birer adet kulakçık ve karıncık adı verilen toplam dört odacıktan oluşur. Sağdaki odacıklar vücuttan toplanan kirli kanı akciğerlere pompalarken, soldaki odacıklar ise vücudun her noktasındaki organlara ve hücrelere ulaştırılacak oksijen yüklü temiz kanı pompalarlar. Kirli kanın sağ kulakçık–karıncık tarafından akciğerlere pompalanması ve orada CO_2 – O_2 alışverişi ile temizlenen kanın tekrar kalbe gelmesi alt-sistemine pulmoner sistem adı verilir. Temiz kanın sol kulakçık–karıncık tarafından atardamar ağı (arterler) üzerinden vücuttaki hücrelere pompalanması ve kirli kanın toplardamar ağı (venler) üzerinden tekrar kalbe gelmesi alt-sistemine de sistemik dolaşım adı verilir. Kalbin başlıca bölümlerinin yapısı Şekil 2.4’de görülmektedir.

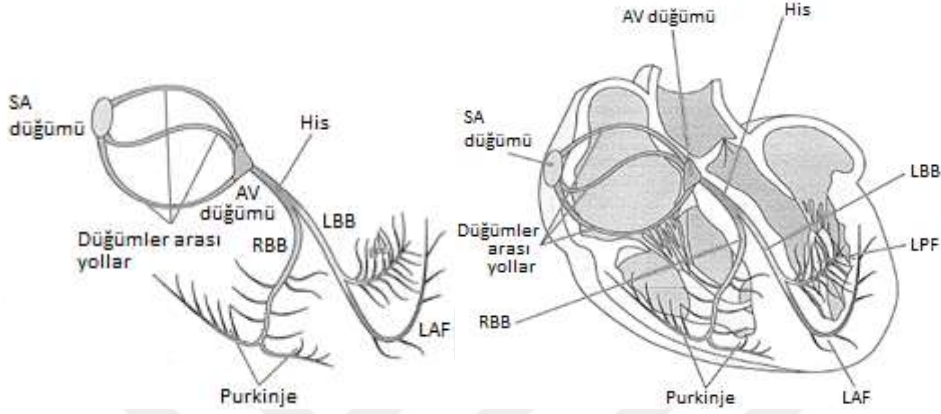


Şekil 2.4 : Kalbin başlıca bölümlerinin yapısı (Guyton ve Hall, 2006).

Vücuttaki hücrelerin ihtiyacı olan oksijen miktarına bağlı olarak, değişen aralıklarla üretilen elektriksel uyarılar, kalp kaslarının kuvvetli bir şekilde kasılmasını ve kalp odacıklarındaki kanın belirli bir sıklıkta pompalanmasını gerçekleştirir. Bu kasılma sıklığına kalp vuru hızı adı verilir. Sağlıklı yetişkinlerde bu değer 60 – 90 vuru/dakika olup çeşitli psikolojik, fizyolojik ve çevresel etkilere bağlı olarak değişiklik göstermektedir. Koşma vb. fiziksel efor harcanması gibi durumlarda vücuttaki hücrelerin oksijen ihtiyacı arttığı için kalp vuru hızı yükselirken; dinlenme, uyku vb

durumlarda düşmektedir. Kalbin bu kasılma sıklığını belirleyen bölümlerine kalp vuru hızı düzenleyici (pacemaker) adı verilir.

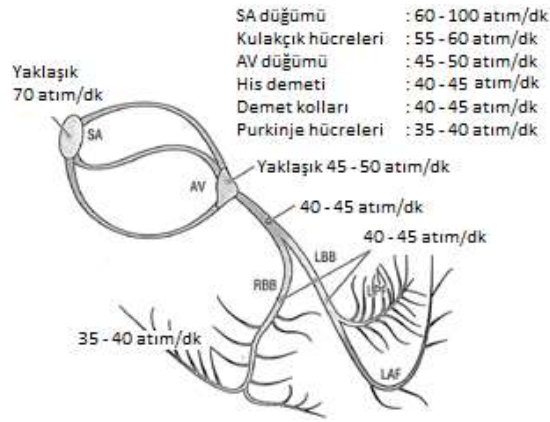
Kalp kaslarının kasılmasını başlatan uyarıları ileten sisteme kalbin elektriksel iletim sistemi denilir. Kalbin elektriksel uyarı sisteminin başlıca bölümleri ve kalbin içindeki konumları Şekil 2.5’de görülmektedir.



Şekil 2.5 : Kalbin elektriksel uyarı sistemi ve kalp içindeki konumu (Garcia ve Miller, 2004).

Kalbin elektriksel iletim sistemi, sinüs (SA) düğümü (Sino-atrial node, SA node), düğüm arası yollar, atriyoventriküler düğüm (atrioventricular node, AV node), his demeti (bundle of his), sol ve sağ demet kolları (left and right bundle branches) ve purkinje liflerinden (purkinje fibres) oluşur. Kalbin elektriksel iletim sistemini oluşturan sinir hücrelerinin elektriksel olarak dört temel özelliği bulunmaktadır: kendi kendisine elektriksel uyarı başlatabilme, dışarıdan elektriksel olarak uyarılabilme, elektriksel uyarıyı iletme ve bekleme süresi (refrakter periyot). Bu elektriksel iletim sistemi üzerindeki tüm bölümlerin kendi kendine uyarı başlatabilme özelliği bulunmaktadır, ancak uyarı başlatma sıklıkları farklı değerlere sahiptir. Değişik bölümlerin uyarı başlatma hızları Şekil 2.6’da görülmektedir. Her bölüm, iletim yolu üzerinde kendinden önce gelen bölümden daha düşük bir hızla uyarı başlatabilmektedir. Bu şekilde, daha hızlı olan vuru hızı düzenleyici (pacemaker), her vuruda kendinden sonra gelen bölümlerin bekleme sürelerini sıfırlamakta ve normal şartlar altında daha düşük hıza sahip bölümler hiçbir zaman uyarı başlatmamaktadır. Ancak herhangi bir nedenle daha hızlı olan pacemaker uyarı başlatamaz, ya da bu uyarı sonraki bölümlere iletilemezse, o zaman bir sonraki en hızlı pacemaker devreye girerek kalbin biraz daha yavaş, ama düzenli bir şekilde atmaya devam etmesini sağlar. Böylece kalp gibi hayati öneme sahip bir organın yapısında, dolaşım sisteminin

devamlılığını sağlayacak birkaç kademe yedek kontrol düzeneği bulunduğu görülmektedir.



Şekil 2.6 : Kalpteki elektriksel ileti sisteminin bölümlerinin uyarı başlatma sıklıkları. (Garcia ve Miller, 2004).

SA düğümü, ortalama 70 (60-100) vuru/dakikalık vuru başlatma hızı ile kalbin öncelikli vuru hızı düzenleyicisidir. SA düğümü ile AV düğümü arasında üç ana düğümler-arası iletim yolu (anterior, middle ve posterior) bulunmaktadır. SA düğümünde oluşan uyarı (aksiyon potansiyeli), kulakçıklarda bulunan bu iletim yolları üzerinden hızlı bir şekilde yayılarak kulakçıkların kasılmasını ve buralardaki kanın karıncıklara pompalanmasını sağlar. Kulakçıklar ile karıncıklar arasında septum adı verilen yağlı bir bölge elektriksel yalıtıma sahiptir ve uyarının karıncıklar bölgesine iletilmesi sadece AV düğümü üzerinden gerçekleşir.

SA düğümünde oluşan bir uyarı yaklaşık 30-50 msn sonra AV düğümüne ulaşır. Bu süre kulakçıklardaki kanın tamamen karıncıklara aktarılması için yeterli değildir. Karıncıkların kasılmasının, odacıklar tam olarak dolduktan sonra olması gerekir. Bu süre, AV düğümünün gelen uyarıyı yaklaşık 0.10 – 0.11 saniye kadar geciktirmesiyle sağlanır. Böylece karıncıklardaki odacıklar tamamen kan ile dolar ve kasılmanın verimli bir şekilde vücuda kan pompalaması gerçekleştirilebilir.

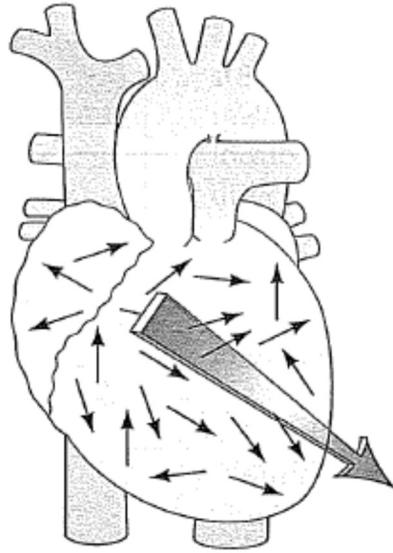
AV düğümünü his demeti takip eder. Sol ve sağ dallara ayrılan his demeti sol ve sağ karıncıklara elektriksel uyarıyı iletir. His demetinin dalları purkinje liflerine dağılır. His demeti ve demet dalları elektriksel uyarıyı iletirken, her biri kalp kası hücrelerine bağlı olan bu purkinje lifi hücreleri, karıncıkları depolarize ederek kalp kaslarının kuvvetli bir şekilde kasılmasını sağlar. Bu kasılma ile de karıncıklardaki kan

pompalanarak, sol karıncıktaki temiz kan tüm vücuda, sağ karıncıktaki kirli kan ise temizlenmek üzere akciğerlere iletilir.

2.2 EKG İşaretinin Oluşumu ve Başlıca Bölümleri

Bölüm 2.1 de açıklandığı şekilde, kalbin elektriksel iletim sistemindeki bütün bölümlerde meydana gelen aksiyon potansiyellerinin toplamı sonucu elektrokardiyogram işareti oluşur.

Kalbin yapısında binlerce kas elamanı bulunmaktadır. Her bir depolarizasyon ve repolarizasyonu anı, çok sayıda hücrenin elektriksel aktivasyonunu temsil eder. Bu elektriksel aktivasyonlar, her biri yön ve genliğe sahip bir vektör kuvveti ile gösterilebilir. Bütün bu vektörlerin toplamı kardiyak vektörü olarak adlandırılır. Kardiyak vektörünün sembolik gösterilimi Şekil 2.7’de görülmektedir.

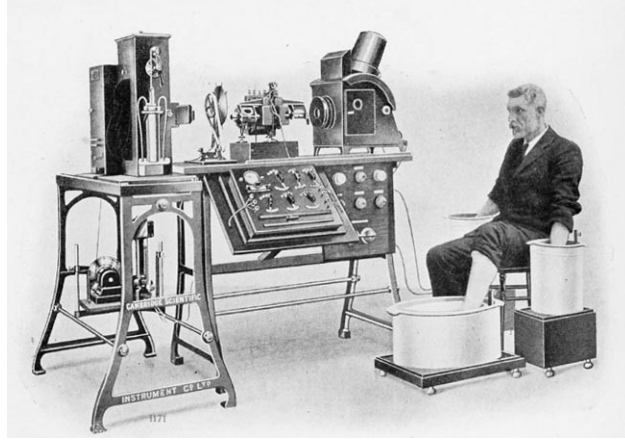


Şekil 2.7 : Kalbin kardiyak vektörünün sembolik gösterimi (Garcia ve Miller, 2004).

Kalp, gövde içine gömülü bir elektrik üretici olarak düşünülürse, bu üreticinin çıkışının ölçülmesi, cerrahi bir ameliyat olmaksızın ancak vücudun üzerine çeşitli noktalara yerleştirilen elektrotlar arasında yapılan potansiyel farkı ölçümleri ile gerçekleştirilebilir.

Standart EKG ölçümü, 12 ayrı derivasyon adı verilen elektrot grupları ile yapılır. Elektrokardiyografi cihazını geliştirip, EKG ölçüm işlemini 1903 yılında gerçekleştiren Willem Einthoven, bu çalışması nedeniyle daha sonra 1924 yılında

Nobel ödülü almıştır. Bu elektrokardiyograf cihazının fotoğrafı Şekil 2.8’de görülmektedir.

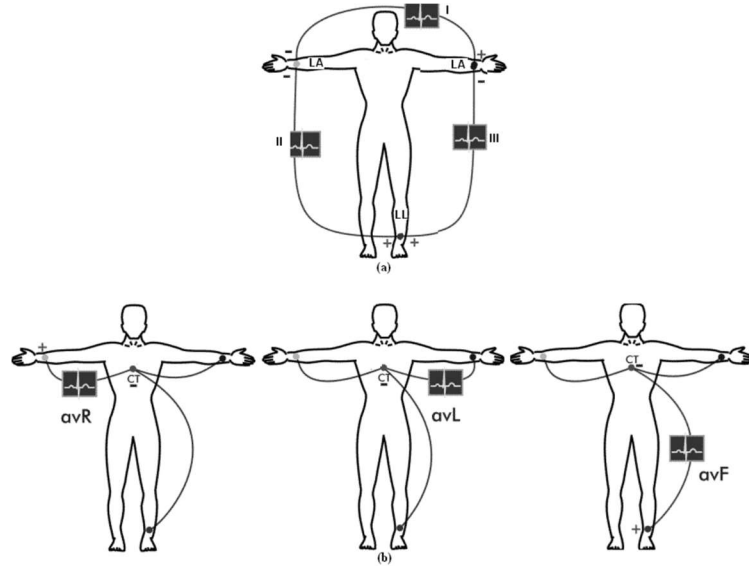


Şekil 2.8 : Willem Einthoven tarafından 1903 yılında geliştirilen elektrokardiyograf cihazı (Gacek ve Pedrycz, 2012).

EKG derivasyonları üç gruba ayrılmaktadır.

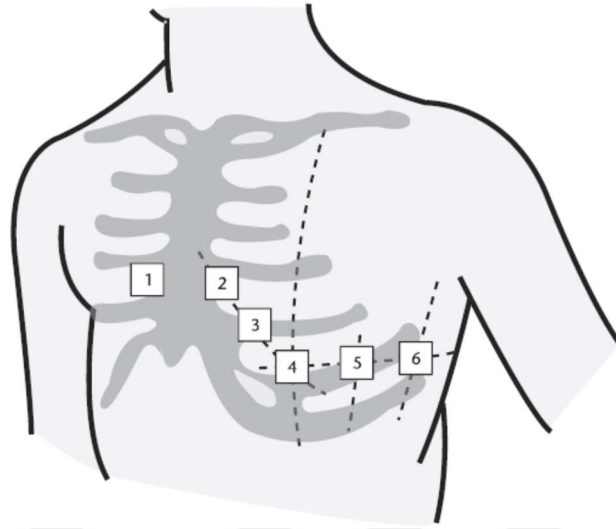
- Standart çift kutuplu derivasyonlar (I, II, III)
- Kuvvetlendirilmiş tek kutuplu derivasyonlar (avR, avL, avF)
- Göğüs derivasyonları (V1, V2, V3, V4, V5, V6)

İlk iki gruptaki derivasyonların elde edilmesi için elektrotların bağlantı yerleri Şekil 2.9’da görülmektedir.



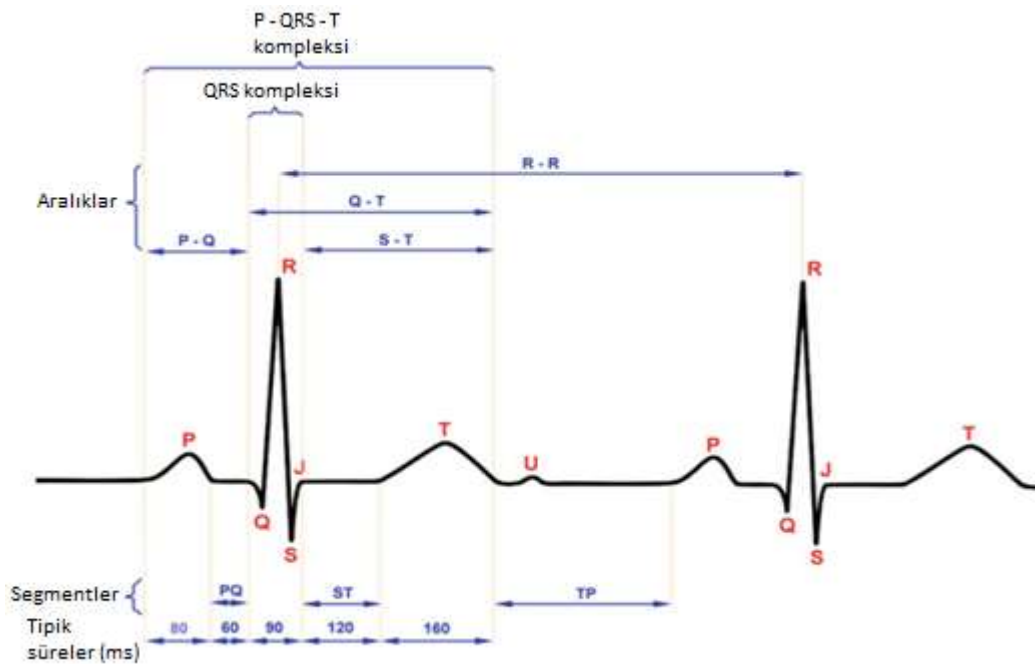
Şekil 2.9 : Standart çift kutuplu ve kuvvetlendirilmiş tek kutuplu derivasyonlarda elektrotların bağlanma yerleri (Yıldız, 2011).

Göğüs derivasyonlarının elde edilmesi için elektrotların bağlantı yerleri Şekil 2.10’da görülmektedir.



Şekil 2.10 : Göğüs derivasyonları için elektrotların bağlanma yerleri (Reisner, 2011).

EKG işareti üzerinde değişik özellikler gösteren bölümler P, QRS, T gibi harfler ile adlandırılır. Normal bir kalp vurusuna ait EKG işaretinin başlıca bölümleri, dalgalar ve aralıklar Şekil 2.11’de görülmektedir.



Şekil 2.11 : Normal bir kalp vurusuna ait EKG işareti (Gacek ve Pedrycz, 2012).

Kulakçıkların depolarizasyonu esnasında P dalgası, karıncıkların depolarizasyonu esnasında ise QRS dalgası oluşur. P dalgasının genliği kulakçık kaslarının fonksiyonel aktivitesini belirtirken, PQ aralığı his demeti iletim süresini gösterir. QRS kompleksinin genliği karıncık kaslarının aktivasyonuna karşı düşer. His demeti ve dallarındaki iletim bozuklukları QRS dalgasında değişikliklere yol açar.

Karıncıkların repolarizasyonu T dalgası şeklinde görülür. Kulakçıkların repolarizasyonu ise yüksek genlikli QRS dalgası ile aynı esnada oluştuğu için ayırt edilemez. QRST dalgaları karıncık aktivasyonunu gösterdiği için ventriküler kompleks olarak adlandırılır. ST segmentinde karıncık kas hücreleri yavaş, T dalgasında ise hızlı bir şekilde repolarize olur.

EKG kayıtlarının yaklaşık dörtte birinde T dalgasından sonra U dalgası adı verilen daha küçük genlikli bir bölüm daha bulunmaktadır. Genellikle takip ettiği T dalgası ile aynı polariteye sahip olan bu dalganın, mekanik-elektriksel geri beslemeler sonucu oluşan sonraki potansiyellerin neden olduğu düşünülmektedir (Gacek ve Pedrycz, 2012).

2.3 Kalp Vurularının Tipleri

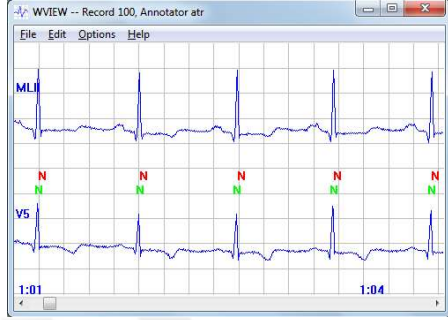
Kalp vuruları normal ve aritmik vurular olmak üzere ikiye ayrılır. Normal vuruların özellikleri ve çeşitli aritmik vurulara ait açıklamalar aşağıda verilmiştir.

2.3.1 Normal vurular (N tipi vurular)

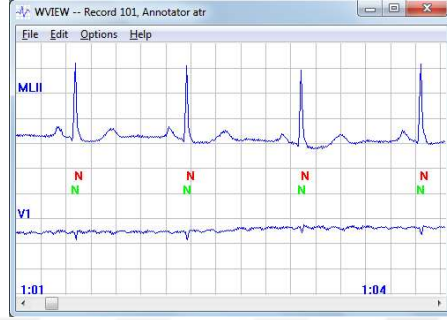
Yetişkinlerde ortalama normal kalp atışı dakikada 60-100 vuru şeklindedir. Vuru hızını belirleyen uyarılar SA düğümünden kaynaklanır. Düzenli bir ritim mevcuttur. Her vuruda P dalgasını 0.12 – 0.20 saniye sonra QRS kompleksi takip eder. Düzenli P-P (veya R-R) aralıkları mevcuttur (Chung, 1989). QRS dalgasının genişliği 0.08 – 0.12 saniye kadardır (Sandoe ve Sigurd, 1991). Vuru hızı, 60 vuru/dakika'nın altına düştüğünde bradikardi, 120 vuru/dakika'nın üzerine çıktığında ise taşikardi olarak adlandırılır. Normal atriyal aktivasyonda II. derivasyonda P dalgası 0.12 saniyeden az bir genişliğe sahiptir.

Normal vurular, kişiden kişiye veya aynı kişide değişik fizyolojik ve çevresel faktörlerin etkisiyle değişik zamanlarda farklılıklar gösterebilir. MIT-BIH veri tabanındaki çeşitli kayıtlarda yer alan normal (N-tipi) vuru örnekleri Şekil 2.12'de

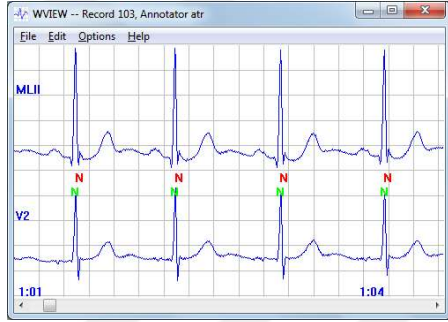
görülmektedir. 100, 101, 103, 106, 108, 112, 113, 117, 202 ve 230 numaralı kayıtlardaki normal vuru örnekleri incelendiğinde, her kayıttaki ortak bulunan II. derivasyondaki (MLII) N-tipi vuruların birbirleri arasında çeşitli farklılık göstermelerine rağmen, uzman kardiyolog tarafından hepsinin N-tipi vuru olarak işaretlenmiş oldukları görülmektedir.



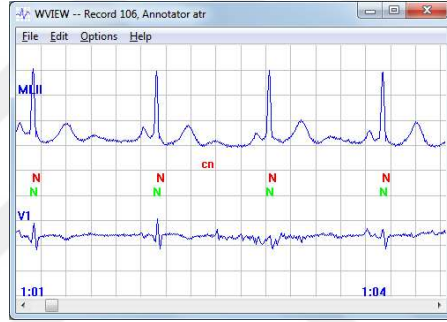
a. 100 no'lu kayıttaki Normal vuru örnekleri



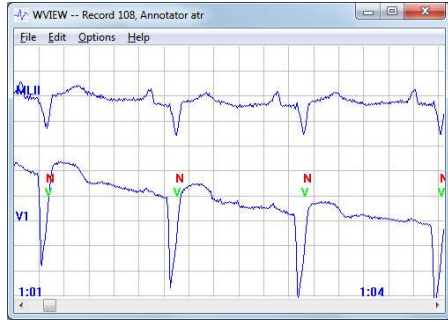
b. 101 no'lu kayıttaki Normal vuru örnekleri



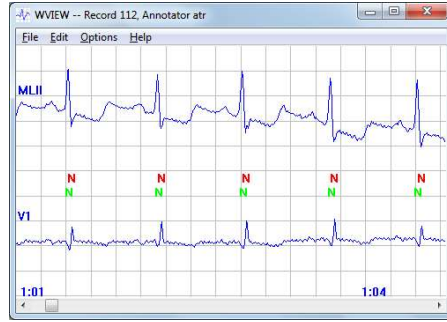
c. 103 no'lu kayıttaki Normal vuru örnekleri



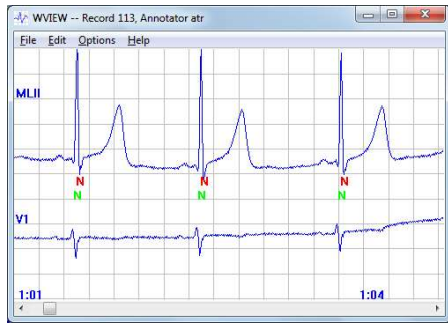
d. 106 no'lu kayıttaki Normal vuru örnekleri



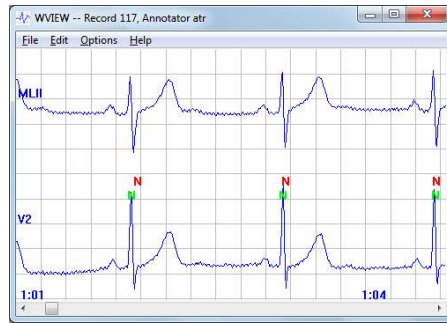
e. 108 no'lu kayıttaki Normal vuru örnekleri



f. 112 no'lu kayıttaki Normal vuru örnekleri

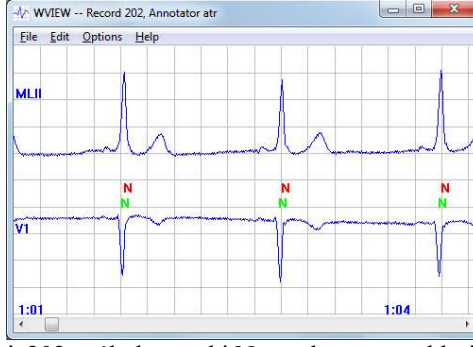


g. 113 no'lu kayıttaki Normal vuru örnekleri

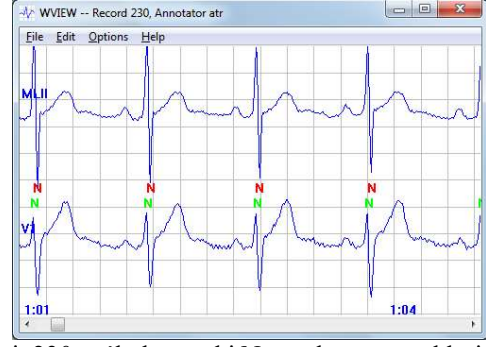


h. 117 no'lu kayıttaki Normal vuru örnekleri

Şekil 2.12 : Çeşitli hastalardan alınmış olan normal vuru örnekleri.



i. 202 no'lu kayıttaki Normal vuru örnekleri



j. 230 no'lu kayıttaki Normal vuru örnekleri

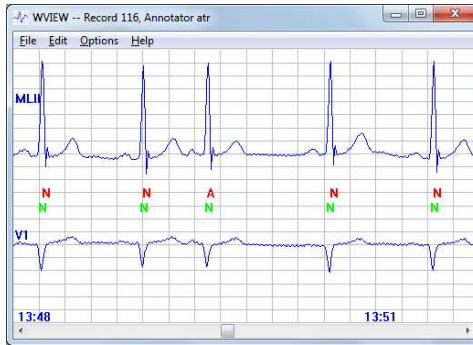
Şekil 2.12 (devam): Çeşitli hastalardan alınmış olan normal vuru örnekleri.

2.3.2 Aritmik vurular

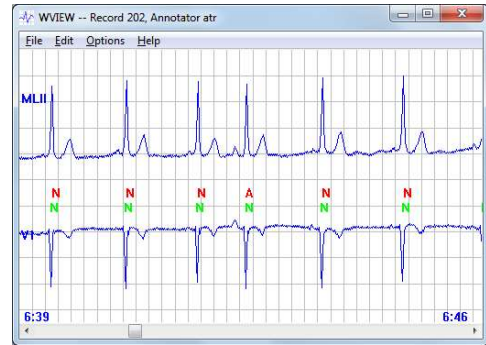
Kalbin düzenli ritmik verusundaki her türlü düzensizlikler veya elektriksel iletim yolundaki düzensizliklerden kaynaklanan şekil bozuklukları genel olarak aritmi olarak adlandırılmaktadır. Normal sinus ritmi mevcut olmakla birlikte, eğer dakikadaki vuru sayısı 60'dan az ise Bradikardi, 120'den çok ise Taşikardi olarak adlandırılır. Bu tip vuruların dışında, çeşitli ritim ve şekil bozukluklarına sahip olan başlıca aritmi türlerine ait bilgiler ve bunlara ait gerçek örnek çizimleri aşağıda verilmiştir.

2.3.2.1 Kulakçık erken vuruları (Atrial premature beat, APB)

Kulakçıklar bölgesindeki ektopik bir kaynaktan yayılan elektriksel uyarı, erken kulakçık vurusu (APB) adı verilen aritmeye neden olur. Her bir P dalgasını, QRS kompleksi takip eder. APB tipi aritmik vuru oluştuğunda R-R aralığı, ortalama R-R aralığından belirgin bir şekilde daha kısadır. APB tipi vurulara örnekler Şekil 2.13'de görülmektedir.



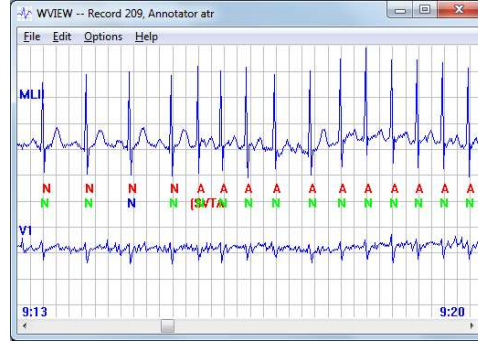
a. 116 no'lu kayıttaki APB tipi vuru örnekleri



b. 202 no'lu kayıttaki APB tipi vuru örnekleri

Şekil 2.13 : APB tipi vuru örnekleri.

Bazı durumlarda tek tek APB tipi vurular oluşurken, bazı durumlarda arka arkaya gelen APB tipi vuru dizileri görülebilir. Bu tür vuruların gerçekleştiği ritimler, karıncık üstü taşikardi saldırısı (Supra Ventricular Tachcardia Attack, SVTA) olarak adlandırılır. SVTA tipi ritim örneği Şekil 2.14’de görülmektedir.



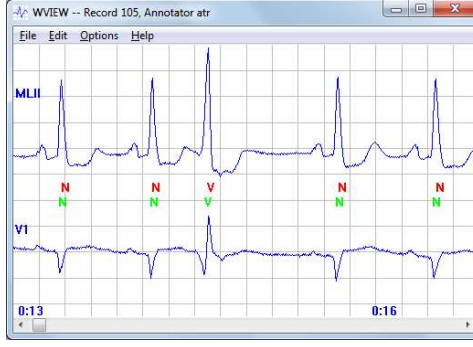
Şekil 2.14 : SVTA tipi ritim örneği.

2.3.2.2 Erken karıncık kasılması (Premature ventricular contraction, PVC)

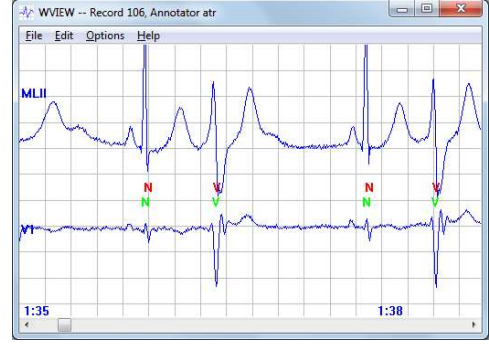
Normal sinüs vurularında elektriksel uyarılar SA düğümünden kaynaklanır ve AV düğümünden geçerek his demeti ve kolları üzerinden karıncıklara yayılır. PVC adı verilen erken karıncık kasılmalarında ise AV düğümünden sonra, karıncık bölgesindeki ektopik (dış kaynaklı) bir noktadan kaynaklanan bir uyarı kalp vurusuna yol açar. Kulakçıkların depolarizasyonuna karşı düşen P dalgası görülmez.

Ortalama bir vuru süresi geçmeden oluşan bu vurudan bir önceki ve bir sonraki vuruların arasındaki süre, ortalama vuru süresinin iki katı şeklindedir. Sonraki vurunun oluşması için geçen süreye kompanze edici faz adı verilir. PVC tipi vurularda PVC kompleksinin genişliği normalden daha uzundur. Genliği ise, normal vurulara göre bazen daha büyük, bazen daha küçük olabilmektedir. Hatta bazen negatif genlikli olur.

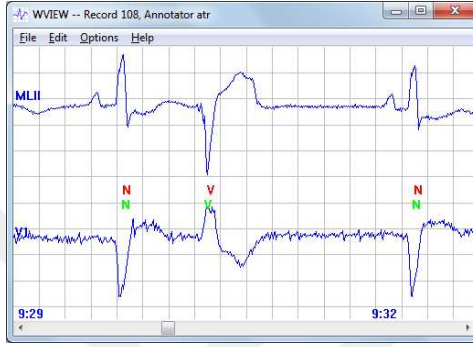
MIT-BIH veri tabanındaki çeşitli kayıtlarda yer alan PVC tipi vuru örnekleri Şekil 2.15’de görülmektedir. Aşağıdaki örneklerde aynı grafik ölçek kullanılmak istenmiş, fakat MIT-BIH veri tabanındaki kayıtların farklı cihazlar kullanılarak yapılmış olması nedeniyle, bazı örneklerde farklı grafik ölçeği kullanılması gerekmiştir.



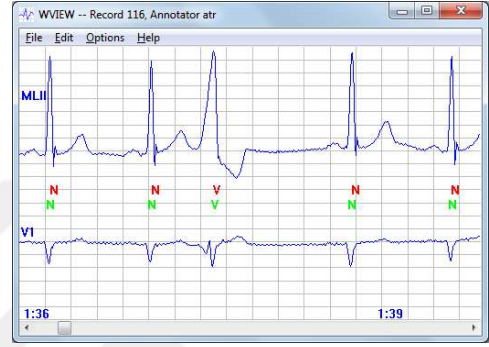
a. 105 no'lu kayıttaki PVC tipi vuru örnekleri



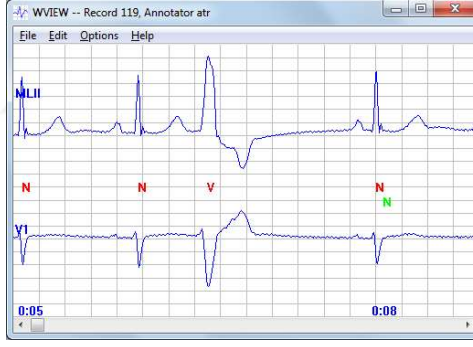
b. 106 no'lu kayıttaki PVC tipi vuru örnekleri



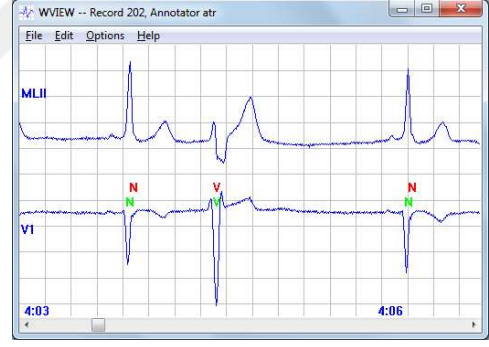
c. 108 no'lu kayıttaki PVC tipi vuru örnekleri



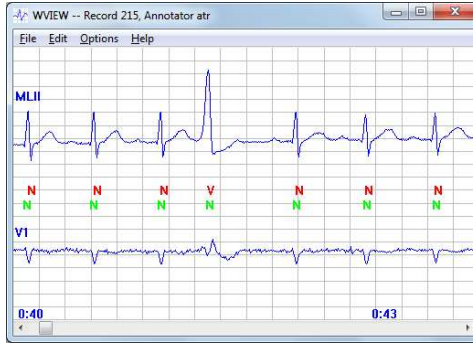
d. 116 no'lu kayıttaki PVC tipi vuru örnekleri



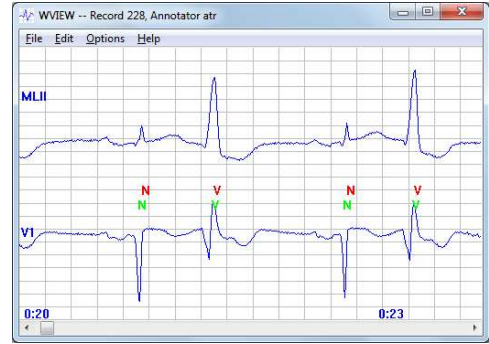
e. 119 no'lu kayıttaki PVC tipi vuru örnekleri



f. 202 no'lu kayıttaki PVC tipi vuru örnekleri



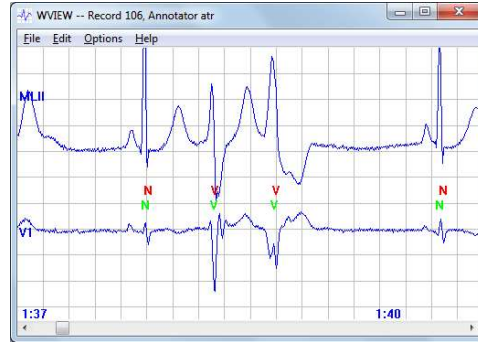
g. 215 no'lu kayıttaki PVC tipi vuru örnekleri



h. 228 no'lu kayıttaki PVC tipi vuru örnekleri

Şekil 2.15 : PVC tipi vuru örnekleri.

Kimi durumlarda tek tek vurular şeklinde oluşan PVC'ler kimi durumlarda ise arka arkaya oluşabilmektedir. İki PVC vurusunun arka arkaya oluştuğu durumlara bigemini adı verilmektedir. Bigemini tipi vuru örneği Şekil 2.16'de görülmektedir.

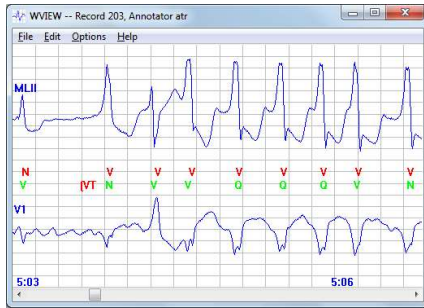


Şekil 2.16 : Bigemini tipi vuru örneği.

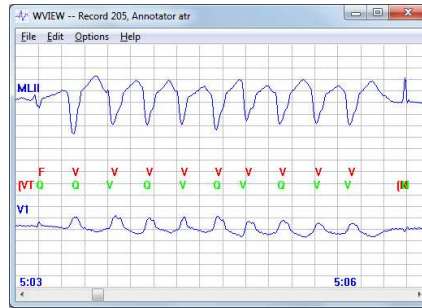
Tek tek oluşan PVC tipi vurular kritik olmamakla birlikte, bir süre sonra karıncık taşikardisi veya karıncık fibrilasyonuna dönüşebildikleri için hayat tehdit edici olarak kabul edilip, dikkatli takip edilmeleri gerekmektedir.

2.3.2.3 Karıncık taşikardisi (Ventricular Tachycardia, VT)

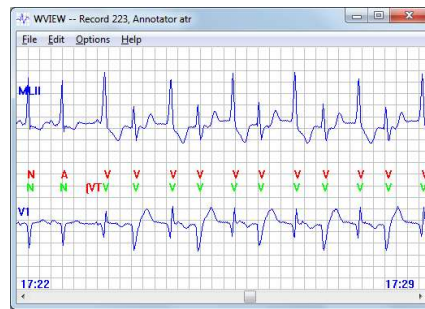
Karıncık bölgesinden kaynaklanan ektopik uyarılarla başlar, hızlı ve ölümcül bir aritmi türüdür. Vuru hızı 70 – 250 vuru/dakika şeklinde olabilir. Çeşitli karıncık taşikardisi (VT) ritim örnekleri Şekil 2.17'de görülmektedir.



a. 203 no'lu kayıttaki VT tipi ritim örnekleri



b. 205 no'lu kayıttaki VT tipi ritim örnekleri

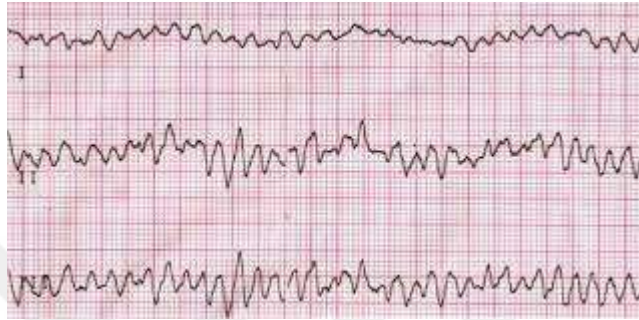


c. 223 no'lu kayıttaki VT tipi ritim örnekleri

Şekil 2.17 : Karıncık taşikardisi (VT) tipi ritim örnekleri.

2.3.2.4 Karıncık fibrilasyonu (Ventricular Fibrillation, VF)

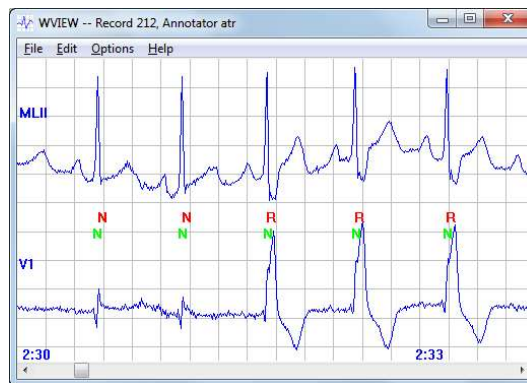
Karıncık bölgesindeki bir veya birden fazla noktadan kaynaklanan ektopik uyarılarla oluşan düzensiz bir ritimdir. Kalp hızı yüksek ve değişkendir. Klasik P, QRS, T dalgaları yoktur. Düzgün ve etkili bir kan pompalama yapılamadığı için, hayat ancak yapay yöntemler ile sürdürülebilir. Aksi takdirde kısa bir süre sonra kalp durur ve hastanın ölümü gerçekleşir. Karıncık fibrilasyonu ritim örneği Şekil 2.18’de görülmektedir.



Şekil 2.18 : Karıncık fibrilasyonu (VF) tipi ritim örnekleri.

2.3.2.5 Sağ dal bloğu (Right Bundle Branch Block, RBBB)

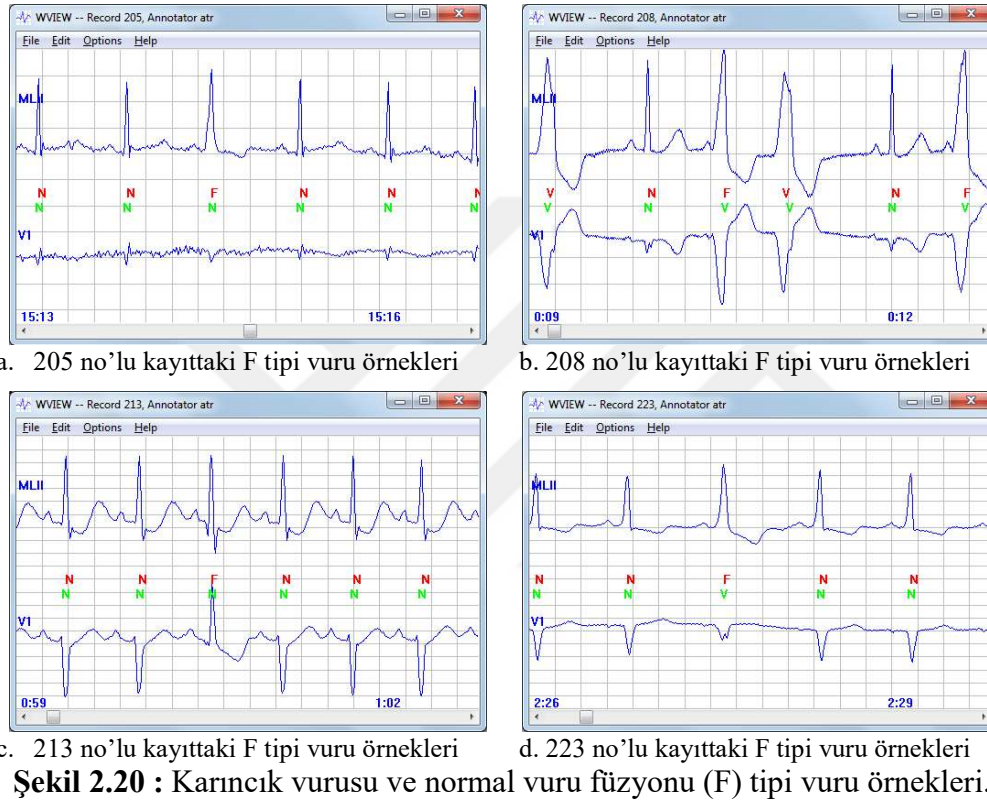
Karıncıklardaki ileti yolu bozukluklarından olan sağ dal bloğu, geniş S dalgası ile karakterize olmaktadır. Bu aritmi tipi oluştuğunda sağ karıncık, sağ dal üzerinden ilerleyen elektriksel uyarı ile uyarılmaz. Fakat sol karıncık sol dal üzerinden düzgün bir şekilde uyarılmaktadır. Bu uyarı AV düğümünden geçerek gecikmeli olarak sağ karıncığı uyarır. Kalp ritmi normal ve QRS genişliği 0.12 saniyeden uzundur. MIT-BIH veritabanı 212 numaralı kayıttaki sağ dal bloğu (RBBB) tipi aritmik vuru örnekleri Şekil 2.19’da görülmektedir.



Şekil 2.19 : MIT-BIH veritabanı 212 numaralı kayıttaki sağ dal bloğu (RBBB) tipi vuru örnekleri.

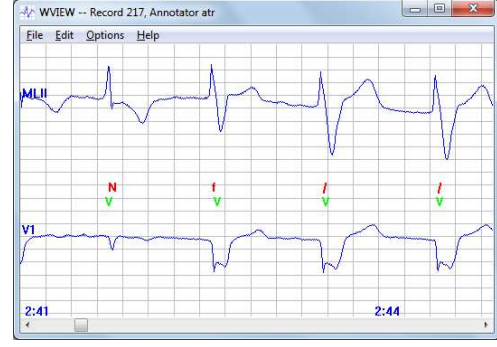
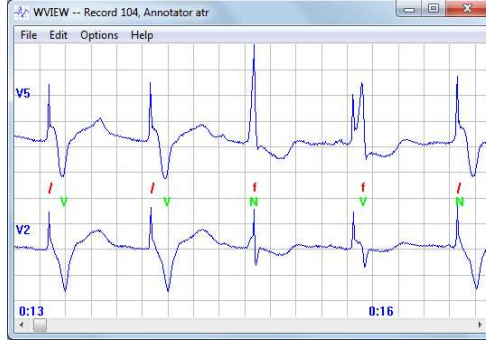
2.3.2.6 Karıncık ve normal vuru füzyonu (F)

Kalp içerisine farklı kaynaklardan yayılan elektriksel uyarıların karşılaşması sonucu oluşan aritmik vurulara Füzyon vurusu (F) adı verilir. Eğer uyarılar karıncık vurusu ile normal vuru karşılaşmasında oluşuyor ise karıncık ve normal vuru füzyonu olarak adlandırılır. Karşılaşan uyarıların kaynağı ve karşılaşma anına göre QRS kompleksinde farklı bozulmalar görülebilir. Çeşitli kayıtlardan alınmış karıncık vurusu ve normal vuru füzyonu (F) tipi vuru örnekleri Şekil 2.20’de görülmektedir.



2.3.2.7 Paced ve normal vuru füzyonu (f)

Kalbe yerleştirilmiş olan bir pacemaker’ın ürettiği uyarı ile normal vuru karşılaşması durumunda, paced ve normal vuru füzyonu adı verilen “f” tipi vurular oluşur. Çeşitli kayıtlardan alınmış f-tipi vuru örnekleri Şekil 2.21’de görülmektedir. Şekilde görülen “/” notasyonu kardiyologlar tarafından MIT-BIH kaydındaki pacemaker vurularını işaretlemek için kullanılmıştır. F tipi vurularda olduğu gibi, f tipi vurularda da farklı dalga şekilleri oluşabildiği görülmektedir.



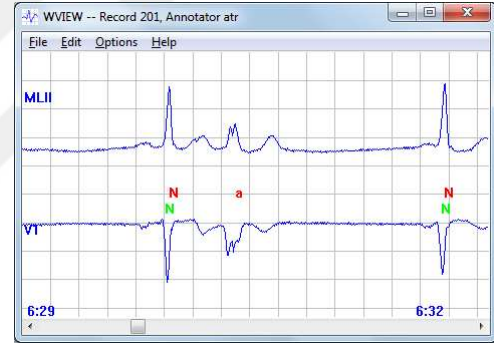
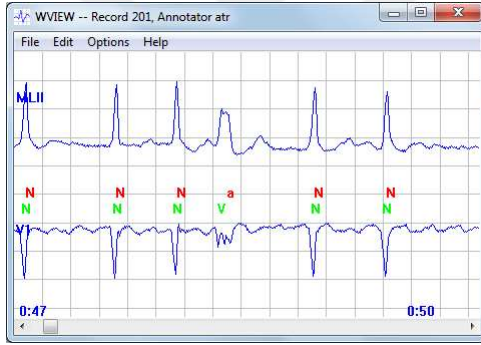
a. 104 no'lu kayıttaki f tipi vuru örnekleri

b. 217 no'lu kayıttaki f tipi vuru örnekleri

Şekil 2.21 : Paced ve normal vuru füzyonu (f) tipi vuru örnekleri.

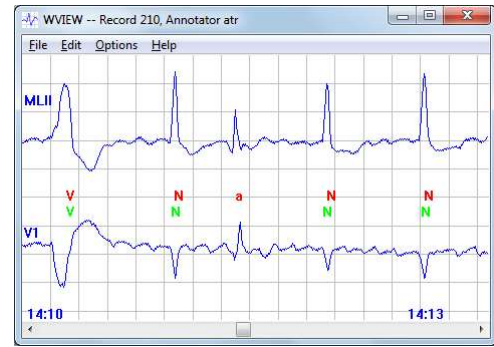
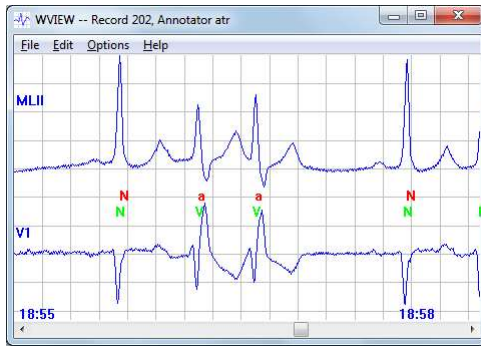
2.3.2.8 Anormal kulakçık erken vuruları (Aberrated atrial premature beat, a)

Kulakçık erken vurusu olmakla birlikte anormal bir yapı gösteren vurular “a” tipi vuru olarak adlandırılır. Bu tip vurularda R dalgasının genliği, normal vuruların R dalgası genliğinden çok daha düşük ve bazı durumlarda R tepesi çentikli olabilmektedir. Çeşitli kayıtlardan alınmış “a” tipi vuru örnekleri Şekil 2.22’de görülmektedir.



a. 201 no'lu kayıttaki a tipi vuru örnekleri

b. 201 no'lu kayıttaki a tipi vuru örnekleri



c. 202 no'lu kayıttaki a tipi vuru örnekleri

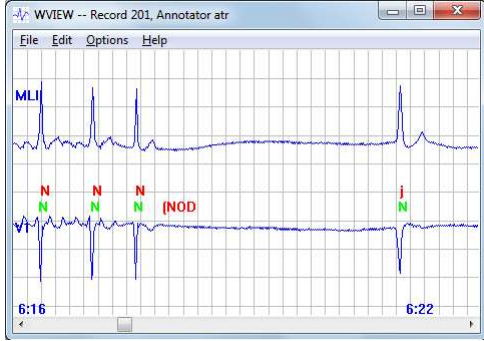
b. 210 no'lu kayıttaki a tipi vuru örnekleri

Şekil 2.22 : Anormal kulakçık erken vurusu (a) tipi vuru örnekleri.

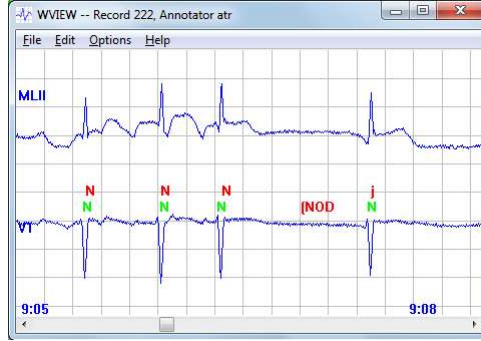
2.3.2.9 Jonksiyonel kaçış vurusu (Junctional escape beat, j)

Atriyal fibrilasyon dönemi sonrası, sino-atriyal düğümünden kaynaklanan vuruların gerçekleşmediği, kalp vurusunun durduğu bazı durumlarda AV düğümünden kaynaklanan bir kaçış vurusu kalbi tekrar çalıştırmaya başlar. Bu tür vurulara “j”

jonksiyonel kaçış vurusu adı verilir. Kendisinden önceki son vuru ile arasındaki uzun aralık, j tipi vuruların karakteristik özelliğidir. Çeşitli kayıtlardan alınmış “j” tipi vuru örnekleri Şekil 2.23’de görülmektedir.



a. 201 no’lu kayıttaki j tipi vuru örnekleri



b. 222 no’lu kayıttaki j tipi vuru örnekleri

Şekil 2.23 : Jonksiyonel kaçış vurusu (j) tipi vuru örnekleri.

3. EKG VERİLERİNİN ANALİZİ

Üzerinde çalışma yapılacak olan verilerin öncelikle analiz edilmeye uygun hale getirilmesi gerekmektedir. İnsan vücudundan alınan elektrokardiyogram işaretinin bilgisayar ortamında işlenebilmesi için, örnekleme yapılarak sayısal verilere dönüştürülmeli, üzerinde bulunan bozucu etkilerden ve gürültülerden arındırılmalıdır.

3.1 Ön İşleme

İşaret işleme uygulamalarında örnekleme frekansı seçiminde, işaretin içinde bulunan maksimum frekans bileşeninin en az iki katı ile örnekleme yapılmalıdır. Teşhise yönelik EKG analiz işlemlerinde 300 Hz ve üzeri örnekleme frekansları yeterli olmaktadır. Sayısal ortama aktarılan EKG verilerinin üzerinde bulunan bozucu etkilerin ve gürültünün çeşitli kaynakları mevcuttur. Ölçümleri kaydeden cihazdan kaynaklanan bozucu etkiler, 50 Hz (ABD’de 60 Hz) şebeke gürültüsü, vücuttaki kasların hareketi ile ilgili elektromiyografik (EMG) işaretler, soluk alıp verme esnasında oluşan taban hattı kaymaları başlıca gürültü kaynaklarıdır. Her bir gürültü kaynağı karakteristik olarak farklı frekans bileşenlerine sahiptir. Örneğin EMG işaretleri yüksek frekans bileşenlerine, taban hattı kayması ise alçak frekans bileşenlerine sahiptir. Bu gürültüleri gidermeye yönelik filtreler kullanılarak istenmeyen işaretlerin elektrokardiyogram üzerindeki bozucu etkisi azaltılır. Ağırlıklı olarak analiz çalışmasının yapılacağı QRS dalgasının frekans aralığı 10 – 25 Hz bölgesinde bulunmaktadır. Buna göre kullanılan filtre karakteristiği, P, QRS ve T dalgalarını geçirecek, gürültü bileşenlerini bastırarak şekilde seçilmiştir (Korürek, 1996).

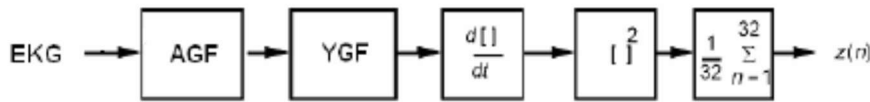
3.2 EKG Verileri İçin Öznitelik Çıkartma

Örüntü tanıma (pattern recognition) problemlerinde, bilgisayarların örüntüleri algılayabilmesi ve üzerinde işlem yapabilmesi için, her bir örüntünün vektörel bir format ile temsil edilmesi gerekmektedir. Farklı sınıflara ait örüntülerin birbirinden en iyi şekilde ayırt edilebilmesi için, sınıfları birbirinden ayıran ve kendi sınıfını en iyi

temsil eden verilerin oluşturulması hedeflenir. Bir örüntüye ait vektördeki her bir bileşen “öznitelik” olarak adlandırılır (Dokur, 2006). Örüntülere ait vektörlerdeki özniteliklerin tamamı, öznitelik domenini oluşturur. Bir vektör “p” adet öznitelik değerinden oluşuyorsa, p-boyutlu öznitelik domeninde çalışılmaktadır.

Her bir kalp vurusuna ait özniteliklerin belirlenmesi için öncelikle her bir vuruya ait referans noktasının, sürekli elektrokardiyogram işareti içindeki yerinin tam olarak bilinmesi gerekir. Bu amaçla ilk aşamada, diğer dalgalara göre daha yüksek genliği ve daha yüksek frekans bileşenleri ile en belirgin bölge olan QRS kompleksinin belirlenmesi gerçekleştirilir. R tepesinin konumu referans nokta olarak alınır ve bu noktadan hareketle gerek zaman domeni öznitelikleri, gerekse frekans domeni özniteliklerinin hesaplanması işlemleri yapılır.

QRS kompleksinin belirlenmesi için literatürde çeşitli algoritmalar yer almaktadır. Bunlardan yaygın olarak kullanılan biri olan Pan&Tompkins algoritması (Pan & Tompkins, 1985) bu tez çalışması kapsamında kullanılmıştır. Pan&Tompkins algoritmasının başlıca adımları Şekil 3.1’de görülmektedir.



Şekil 3.1 Pan&Tompkins algoritmasının başlıca adımları.

Bu algoritmanın dikkat çekici bir özelliği tam sayı katsayılarından oluşan bir bant geçiren filtre aracılığı ile gerçek zamanlı çalışmalarda kullanılmaya uygun olmasıdır. Bir mikroişlemci üzerinde çalıştırıldığında gerçek zamanda hızlı bir şekilde sonuçlar elde edilebilmektedir. Bu tür bir filtre tasarımı kolaylaştırmak için, alçak geçiren ve yüksek geçiren filtreler arka arkaya bağlı bir yapıda kullanılmıştır. Bu filtre, taban hattı kaymalarını, QRS’e göre daha düşük frekanslı olan P ve T dalgalarını bastırmakta, aynı zamanda daha yüksek frekanslı olan şebeke gürültüsünü ve elektromiyografik gürültüleri de bastırmaktadır.

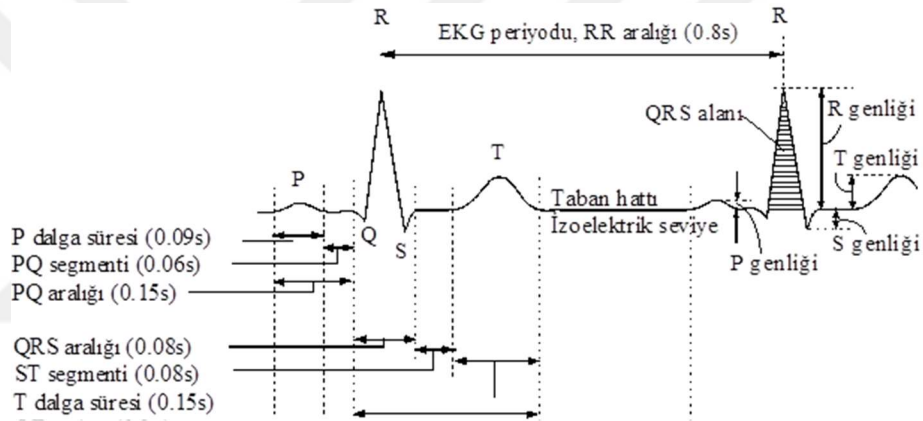
3.2.1 Zaman domeni öznitelikleri

Elektrokardiyogram işaretlerinin analizinde kullanılan zaman domeni öznitelikleri, kalp vurusunun başlıca aşamalarına karşı düşen P, QRS, T dalgalarının genlikleri, genişlikleri, alanları, birbirleri arasındaki süreler gibi değerlerden, ayrıca dalgaların mevcudiyeti ya da yokluğu, normal ya da ters genlikli olması gibi durumlardan oluşur.

Bir vurunun, kendisinden önceki ve kendisinden sonraki vurular arasındaki süreler de önemli zaman domeni özniteliklerdendir. Literatürde yer alan başlıca zaman domeni öznitelikleri şu şekilde sıralanabilir:

- R-tepesi yüksekliği (genliği)
- QRS dalgası alanı
- QRS dalgası genişliği (QRS aralığı)
- P dalgası mevcudiyeti
- P-Q aralığı
- S-T aralığı ve seviyesi
- Bir önceki vuru ile olan R-R aralığı
- Bir sonraki vuru ile olan R-R aralığı

Bazı zaman domeni özniteliklerine dair örnekler Şekil 3.2’de görülmektedir.



Şekil 3.2 : Zaman domeni özniteliklerine örnekler.

Zaman domeni öznitelikleri kullanılırken dikkat edilmesi gereken bir nokta, kullanılan verilerin normalize edilmesinin gerekmesidir. Bunun temel nedeni, normal olarak nitelendirilen EKG değerlerinin kişiden kişiye farklılıklar göstermesidir. Hatta aynı kişiye ait EKG kayıtları, kişinin içinde bulunduğu efor, heyecan vb koşullara bağlı olarak farklılık gösterir. Bunlara ek olarak kayıt cihazının özelliklerinin de EKG ölçüm değerleri üzerinde etkisi bulunmaktadır. Bu nedenlerle, kayıt alınan kişiden, ölçüm koşullarının ve kayıt cihazının etkilerinden bağımsız olarak tüm EKG kayıtlarının birlikte kullanılabilmesi için normalizasyon işlemi yapılmalıdır. Zaman domenindeki özniteliklerin normalizasyonu için yaygın olarak kullanılan bir yöntem, EKG kaydı içindeki bir vuruya ait öznitelik değerinin, son sekiz normal vurunun o öznitelik değerinin ortalamasına göre normalize edilmesidir. Böylece, örneğin iki vuru arasındaki süre, kişinin EKG kayıt işlemi esnasında değişen efor durumuna göre

farklılıklar gösterse de bu şekilde normalizasyon yapılması sayesinde doğru bir şekilde sınıflanabilmektedir.

Tez çalışmasında kullanılan öznitelik hesaplanma yöntemleri ve geliştirilen yazılımın yapısı Bölüm 5.2’de açıklanmıştır.

3.2.2 Frekans domeni öznitelikleri

Elektrokardiyogram, bir zaman domeni işaretidir, ölçülen değerler zamanın bir fonksiyonudur. Ancak bazı kalp vuru tipleri, sadece zaman domeni öznitelikleri kullanılarak diğer vuru tiplerinden başarılı bir şekilde ayırt edilememektedir. Karıncık vurusu ve normal vuru füzyonu (F), paced ve normal vuru füzyonu (f) tipi vurular bu duruma birer örnektir. Böyle durumlarda işaret matematiksel bir dönüşüm uygulanarak farklı bir uzaya taşınıp, elde edilecek yeni özniteliklerle başarılı bir şekilde analiz edilip sınıflandırılabilir. Bu amaçla yaygın olarak kullanılan bir yöntem Fourier dönüşümü ile işaretin frekans spektrumunun (frekans bileşenlerinin) elde edilmesidir. Fourier dönüşümü, aşağıdaki iki denklem ile ifade edilir.

$$X(f) = \int_{-\infty}^{+\infty} x(t) \cdot e^{-j2\pi ft} \cdot dt \quad (3.1)$$

$$x(t) = \int_{-\infty}^{+\infty} X(f) \cdot e^{j2\pi ft} \cdot df \quad (3.2)$$

Fourier dönüşümünde işaret, farklı frekanslara sahip karmaşık üstel fonksiyonlara ayrıştırılır. Eşitliklerde görülen f , frekansı; t ise zamanı göstermektedir. $x(t)$, zaman domenindeki işaret iken, $X(f)$, frekans domenindeki işarettir. Yukarıdaki eşitliklerde görüldüğü şekilde, dönüşüm işlemi esnasında $x(t)$ işareti belirli bir f frekansındaki üstel bir terim ile çarpılıp, $-\infty / +\infty$ aralığında tüm zaman üzerinden integrali alınmakta, böylece zamandan bağımsız bir nitelik kazanmaktadır. f frekansındaki bileşenin her hangi bir anda ortaya çıkması sonucu değiştirmemektedir. Böylece Fourier dönüşümü ile işaretin frekans spektrumu elde edilir.

Bilgisayar ortamında analiz edilen işaretlerin örneklenmiş değerleri ile çalışıldığından, Ayırık Fourier Dönüşümü (AFD) kullanılır. Ayırık Fourier dönüşümü de $-\infty / +\infty$ aralığında tanımlıdır, fakat gerçek hayattaki problemlerde üzerinde çalışılan işaretlere ait sınırlı sayıda örnekler mevcuttur. Bu durumda, işaretin sabit bir pencere aralığında tanımlanması hali için, Ayırık Fourier Dönüşümü aşağıdaki şekilde ifade edilir.

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j2\pi k n / N} \quad (3.3)$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \cdot e^{j2\pi k n / N} \quad (3.4)$$

Eşitliklerde görülen $x(n)$ ayrık zaman domeni işaretini, $X(k)$ ise ayrık frekans domeni işaretini göstermektedir. Daha fazla örnekten oluşan daha büyük pencereler kullanılması halinde frekans eksenini daha iyi çözünürlük ile temsil edilir.

Frekans domenindeki öznitelikler EKG kayıt cihazının kazanç, ofset ayarı gibi özelliklerinden önemli ölçüde etkilenmektedir. Zaman domeninde olduğu gibi, frekans domenindeki öznitelikler için de normalizasyon yapılması gereklidir. Her bir vuruya ait örneklenmiş verilerin bulunduğu pencere üzerinde normalizasyon gerçekleştirilerek frekans domenindeki öznitelik vektörlerinin EKG kayıt cihazının parametrelerine bağımlılığı azaltılır. Bu amaçla yaygın olarak kullanılan bir yöntem, her bir vuruya ait pencere içindeki işaretin enerjisinin bir e eşitlenmesidir (Dokur, 2000).

3.2.3 Özniteliklerin indirgenmesi

Sınıflama işleminde kullanılacak özniteliklerin seçimi sınıflama başarımı açısından çok önemlidir. Genel olarak sınıflama başarımının artırılması için yüksek sayıda öznitelik kullanılmasının tercih edilmesi düşünülebilir. Fakat pratikte, belirli bir ayırt ediciliğe sahip olan öznitelikler kullanıldıktan sonra, eklenen diğer özniteliklerin sınıflama başarımına anlamlı bir olumlu katkı yapmadıkları halde, yapılması gereken hesaplama süresini belirgin bir şekilde arttırdıkları bilinmektedir. Bu durumu önlemek ve en yüksek ayırt ediciliğe sahip olan öznitelik kümesini belirlemek için diverjans analizi yönteminin kullanılması uygundur (Dokur ve Ölmez, 2001).

Diverjans analizinde farklı sınıflara ait olan örnek vektörlerinin öznitelik domeninde saçılımı ve sınıflar arası uzaklık incelenir. Belirli bir öznitelik değeri açısından sınıfiçi saçılımın az olması ve sınıflar arası mesafenin olabildiğince çok olması istenir. Aynı sınıftaki örneklerin incelenen öznitelik değerinin saçılımı arttıkça ve farklı sınıflardaki örnekler birbirine yaklaştıkça, o özniteliğin diverjansı yani ayırt ediciliği düşmektedir. Diverjans hesabının nasıl gerçekleştirildiği denklem 3.5’de görülmektedir (Kohen, 1986).

İncelenen veri kümesi üzerinde, her bir öznitelik için diverjans değeri hesaplandıktan sonra, bu öznitelikler içinden en yüksek ayırt ediciliğe sahip olan bir alt küme seçilerek, bu yeni öznitelik kümesi ile sınıflama işlemi gerçekleştirilir.

$$\begin{aligned}
 W_i^j &= \sum_t (\underline{o}_i^{tj} - \underline{\hat{\mu}}_i^j) \cdot (\underline{o}_i^{tj} - \underline{\hat{\mu}}_i^j)^T \quad j = 1, 2, \dots, K; \quad i = 1, 2, \dots, p \\
 \underline{\hat{\mu}}_i &= \sum_{k=1}^K \underline{\hat{\mu}}_i^k \quad W_i = \sum_{k=1}^K W_i^k \quad B_i = \sum_{k=1}^K (\underline{\hat{\mu}}_i - \underline{\hat{\mu}}_i^k) \cdot (\underline{\hat{\mu}}_i - \underline{\hat{\mu}}_i^k)^T \\
 D_i &= \text{tr}((W_i)^{-1} B_i)
 \end{aligned} \tag{3.5}$$

$\underline{o}_i^{tj}$: j-inci sınıfa ait i-boyutlu t-inci örnek vektörü

$\underline{\hat{\mu}}_i^j$: j-inci sınıfa ait i-boyutlu örnek vektörlerinin ortalaması

W_i^j : j-inci sınıfın sınıf-içi saçılım matrisi

B_i : sınıflar-arası saçılım matrisi

K : sınıf sayısı

p : öznitelik sayısı

D_i : i-inci boyutta (i. Özniteliğin) diverjans değeri

$\text{tr}(\cdot)$: matrise uygulanan iz (trace) işlemi

Gerek zaman gerekse frekans domeni öznitelikleri ile çalışırken kullanılabilecek özniteliklerin sayısı çok yüksek olabilmektedir. Çok fazla sayıda öznitelik kullanılması sınıflama başarımında, bir noktadan sonra belirgin bir iyileşme sağlamamakta, fakat hesaplama süresini aşırı oranda arttırmaktadır. Bu nedenle “m” adet öznitelik içinden, optimal veya optimale yakın sonucu verebilecek “p” adet öznitelik altkümesinin belirlenmesi gerekir ($p < m$). Böyle bir altkümenin seçilmesi, olası tüm farklı kombinasyonlar oluşturulup denenerek gerçekleştirilebilir. Eksiksiz araştırma (exhaustive search) adı verilen bu yöntem basit, fakat özellikle büyük m ve p değerleri için çok uzun hesaplamalar gerektiren bir yöntemdir. Örneğin $m=60$ adet özniteliğin mevcut olduğunu düşünelim. Bu kümenin içinden en iyi 8 öznitelikten oluşan ($p=8$) kümeyi bulmak için,

$$\binom{60}{8} = \frac{60!}{(60-8)! \cdot 8!} = 2.56 \times 10^9$$

adet kombinasyonların denenmesi gerekir ki pratikte bu mümkün değildir.

Eksiksiz araştırma yerine kullanılabilecek en basit yöntem, “En iyi K” (K-best) yöntemidir (Url-2). Bu yöntemde, en yüksek ayırt ediciliğe sahip olan k adet öznitelikten oluşan küme kullanılır. Kimi durumlarda, her biri ayrı ayrı yüksek ayırt ediciliğe sahip olan k adet öznitelikten oluşan küme, en iyi sonucu veren küme olmayabilir.

Optimal öznitelik kümesini bulmaya yönelik diğer bir yöntem ise dinamik programlama’dır (Dynamic programming). Eksiksiz araştırma yöntemine göre daha az sayıda hesaplama gerektiren bu yöntem, çok aşamalı bir optimizasyon tekniğini kullanmaktadır. Dinamik programlama yönteminde, önce bütün özniteliklerin ayırt edicilikleri hesaplanır. En etkin olan öznitelikten geri kalan m-1 özniteliğin içinden hangisinin, ilki ile birlikte kullanıldığında en yüksek ayırt edilebilirlik değerinin elde edildiği kontrol edilir. Örneğin sınıflama problemi için, bu iki öznitelik ile sınıflama başarımı hesaplanır. Bu işleme p adet elemandan oluşan öznitelik kümesi elde edilene kadar devam edilir. Bu yöntemde gerçekleştirilen hesaplama sayısı (HS), denklem 3.6’da görülmektedir.

$$HS = m + (m - 1) + (m - 2) + \dots + (m - p + 1)$$

$$HS = p \cdot m - \frac{p \cdot (p-1)}{2} = p \cdot \left(m - \frac{p-1}{2}\right) \quad (3.6)$$

Örneğin m=60 adet öznitelik içinden, optimal çözüme yakın p=8 adet öznitelik kümesini bulmak için, bu yöntem ile yapılması gereken hesaplama sayısı sadece 452’dir.

3.2.4 Eğitim ve test veri kümeleri

Sınıflama işleminde ilk aşamada, üzerinde çalışılacak olan veri kümesi eğitim ve test kümesi olmak üzere iki alt kümeye ayrılır. Eğitim kümesinde bulunan örnek vektörleri, sınıflama başarımını maksimize edecek şekilde sınıflama algoritmasının eğitilmesinde kullanılır. Eğitim süreci tamamlandıktan sonra, elde edilen haritalama yapısı kullanılarak, test kümesindeki örnek vektörleri üzerinde algoritmanın test edilmesi ve sınıflama başarımının ölçülmesi gerçekleştirilir.

Eğitim kümesinin oluşturulmasında, her sınıfı temsil edecek yeterince örnek içermesine dikkat edilmelidir. Eğitim ve test örneklerinin sayısal dağılımı konusunda literatürde farklı yaklaşımlar kullanıldığı görülmektedir. Bazı çalışmalarda her iki kümede birbirine eşit ya da yakın sayıda örnek kullanılırken, diğer çeşitli çalışmalarda test vektörleri daha az veya daha çok olarak kullanılmaktadır.

Eğitim kümesinde yer alan farklı sınıflara ait örnek vektörlerinin dengeli bir şekilde dağılmış olması da bir başka dikkat edilmesi gereken konudur. Aksi takdirde herhangi bir sınıfa ait çok fazla sayıda örnek kullanılması halinde, özellikle öbekleme yöntemi ile çalışan algoritmaların sınıflama başarımı bu durumdan olumsuz etkilenmektedir. Bu konu ile ilgili daha detaylı açıklama ve bir örnek Bölüm 5.3.1’de verilmiştir.

Her iki kümede kullanılacak örnek vektörlerinin rasgele seçilmiş olması da sınıflama algoritmasının performansının sağlıklı ve doğru bir şekilde değerlendirilebilmesi için önemlidir.

3.3 EKG Verilerindeki Vuru Tiplerinin Sınıflandırılması

Örüntülerin, önceden tanımlanmış gruplara ait olarak etiketlenmesi işlemi sınıflandırma olarak adlandırılmaktadır (Fielding, 2007). Eğitimli bir makine öğrenmesi yöntemi olan bu işlem aynı zamanda “örüntü tanıma” olarak da ifade edilebilmektedir. Öbekleme ve sınıflama kavramları, EKG verileri analiz edilerek incelenen kalp vuru tiplerinin sınıflandırılmasında kullanılan başlıca yöntemler ve performans ölçütleri aşağıda açıklanmıştır.

3.3.1 Öbekleme ve sınıflama

Bir veri kümesini, belirli kurallara göre alt kümelere bölümleme işlemi öbekleme olarak adlandırılır. Bu veri kümesi, çok boyutlu uzayda bir örüntü kümesini temsil etmektedir. Öbekleme esnasında, birbirine benzeyen örüntüler bir arada gruplandırılarak alt kümeler oluşturulur. Böyle alt kümeler “öbek” adı verilir. Bir sınıf, bir veya daha çok öbekten meydana gelebilmektedir. Bu gruptan sonra çeşitli yöntemler kullanılarak sınıflama yapılabilir. Öbekleme ve sınıflandırma algoritmaları arasındaki temel fark, “öğrenme” yöntemidir. Öbeklemede her hangi bir ön bilgi kullanılmadan gruptan yapıldığı için “eğitici”, sınıflamada ise eğitim

kümesindeki örnek vektörlerinin hangi sınıflara ait olduğu bilgisi kullanıldığı için “eğitici” bir öğrenme gerçekleştirilmektedir (Fielding, 2007).

“n” örüntüden meydana gelen “**O**” veri kümesini dikkate alalım ($\mathbf{O} = \{o_1, o_2, \dots, o_n\}$). Bu kümedeki herhangi bir o_i örüntüsü, “p” boyutlu, gerçek değerli bir örnek vektörü ile temsil edilir. Bu şekilde **O** kümesi, n satırlı ve p sütunlu bir veri matrisi ile gösterilir.

Benzer şekilde, “k” öbekten oluşan “**D**” öbektlemesini dikkate alalım ($\mathbf{D} = \{C_1, C_2, \dots, C_k\}$). Öbektleme algoritmalarının amacı, aşağıda verilen üç koşulu sağlayan **D** öbektlemesini belirlemektir ($C_g \neq \emptyset, \forall g; C_g \cap C_h = \emptyset, (\forall g, h \ g \neq h); \bigcup_{g=1}^k C_g = \mathbf{O}$). **D** öbektlemesinde yer alan öbeklerin içinde bulunan örüntülerin birbirine en fazla miktarda benzemesi ve farklı öbeklerdeki örüntülerin mümkün olduğunca birbirinden farklı olması gerçekleştirilmeye çalışılır (Zhang ve diğ, 2010).

Öbektleme işleminde, p-boyutlu bir uzayda, k-adet öbekleri birbirinden ayırt etmek için, “p-1” boyutlu karar yüzeyleri/fonksiyonları tanımlanmalıdır. Üç boyutlu uzayda iki boyutlu yüzeyler, iki boyutlu uzayda tek boyutlu hatlar öbekleri birbirinden ayırt eder. İdeal olarak bu şekilde tüm öbekler bu tür karar yüzeyleri ile birbirinden ayırt edilebilmelidir, fakat gerçek hayattaki çeşitli problemlerde bazı örüntüler, kendi öbeğinin değil de, bir başka öbeğin merkezine daha yakın olurlar. Bir başka ifade ile öbekler örtüşebilir. Bu problemi önlemek için, yüksek ayırt edicilik değerine sahip bir öznelik kümesi kullanılmalıdır. Öbeklerin tam olarak ayrıştırılamadığı durumlarda, yüksek ayırt ediciliğe sahip öznelikler seçilerek örtüşme azaltılmaya çalışılır (Korürek ve diğ, 2010). Eğer örtüşme bu yaklaşımla giderilemiyorsa, verilerin başka bir boyut kümesine aktarılması yöntemi kullanılabilir (Korürek ve Dogan, 2010).

Öbektlemenin başarı oranını ölçebilmek için, bir benzerlik ölçüm yönteminin tanımlanması gerekir. Bu amaçla çeşitli uzaklık ölçüm yöntemleri kullanılabilir. Öklid uzaklığı, Kare Öklid uzaklığı, Chebychev, Manhattan ve Mahalonobis uzaklıkları bunlardan en sık kullanılanlarıdır. Bu tez çalışması kapsamında benzerlik ölçümünde Öklid uzaklığı kullanılmıştır.

D öbektlemesinin performansının ölçülmesi için, aşağıdaki yöntem kullanılabilir. P-boyutlu uzayda, herhangi bir örüntü (o_i) ile, bir öbek merkezi (C_g) arasındaki uzaklık, Öklid uzaklığı olarak denklem 3.7’de görüldüğü şekilde hesaplanır.

$$\text{Öklid Uzaklığı (ÖÜ): } \text{ÖÜ}(\underline{o}_i, \underline{c}_g) = \sqrt{\sum_{m=1}^p (o_{im} - c_{gm})^2} \quad (3.7)$$

$$i=1,2,\dots,n; \quad g=1,2,\dots,k$$

Örüntüleri temsil eden p-boyutlu örnek vektörleri, Öklid uzaklığı ölçümü kullanılarak kendilerine en yakın olan öbeğe atanırlar. Öbekleme probleminin amacı, örüntüler kümesindeki bütün örüntüler ile atandıkları öbeğin merkezi arasındaki uzaklıkların toplamının minimum olmasıdır. Bu Öklid uzaklıkları toplamına, öbeklemenin performansı adı verilir ve denklem 3.8’de görüldüğü şekilde hesaplanır.

$$\text{Perf}(O, D) = \sum_{i=1}^n \min\{\text{ÖÜ}(\underline{o}_i, \underline{c}_g) | g = 1, 2, \dots, k\} \quad (3.8)$$

Öbekleme yapılırken karşılaşılabilecek problemlerden biri, örnek vektörlerinin farklı ölçekte bulunabilmesidir. Bu problemi önlemek için, örnek vektörlerinin değerlerine yukarda bölüm 3.2.1 ve 3.2.2 de açıklandığı şekilde normalizasyonu işlemi uygulanır.

Öbek merkezleri belirlendikten sonra, öbeklerdeki örnek vektörlerinin etiketlenmesi ile sınıflama gerçekleştirilir. Bu amaçla Öklid uzaklığı ölçümü, K-en yakın komşular vb. çeşitli yöntemler kullanılabilir.

3.3.2 Klasik sınıflama yöntemleri

Literatürde yer alan çeşitli klasik sınıflama yöntemlerine en yakın ortalamalar sınıflayıcı (NMC), K-en yakın komşular sınıflayıcı (KNN), Kohonen’in öz-denetimli harita sınıflayıcısı (SOM) vb. örnek verilebilir. Bu yöntemlerin çalışma yapıları aşağıda açıklanmıştır.

3.3.2.1 En yakın ortalamalar sınıflayıcı (NMC)

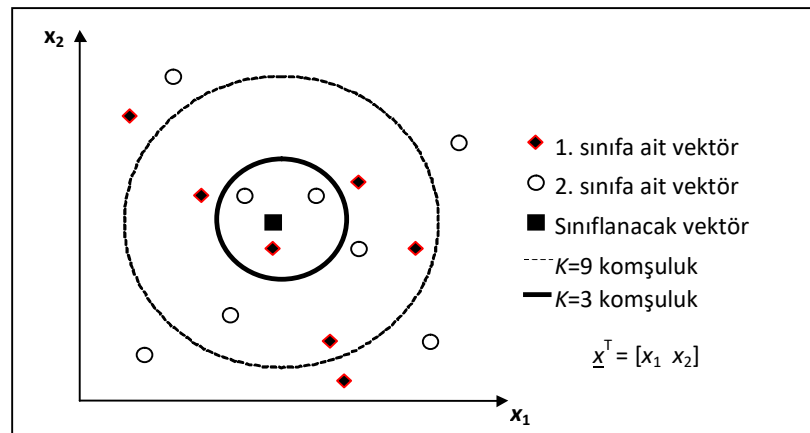
Fukunaga (1990) tarafından geliştirilen en yakın ortalamalar sınıflayıcısı basit ve hızlı bir algoritmadır. Prototip-tabanlı sınıflayıcılardan olan bu yöntemde, öncelikle her sınıfa ait eğitim vektörlerinin ortalaması hesaplanır. Her sınıf, bu ortalama vektör ile temsil edilir (Veenman ve Reinders, 2005). Benzerlik ölçümü için Öklid uzaklığı kullanılır. Sınıflanması istenen yeni bir test vektörü, eğitim örneklerinin belirlediği sınıf prototiplerinden hangisine Öklid uzaklığı olarak daha yakın ise, o sınıfa ait diye etiketlenir.

Farklı sınıflardaki örneklerin birbirleri ile fazla örtüşmediği durumlarda, pek çok uygulama alanlarında başarılı sonuçları hızlı bir şekilde vermektedir (Shin ve Kim, 2009). Fakat özellikle yüksek boyutlu vektörler üzerinde çalışıldığı durumlarda sınıflama başarımı düşebilmektedir (Abdullah ve diğ, 2013).

3.3.2.2 K-en yakın komşu sınıflayıcı (KNN)

Makine öğrenmesi algoritmalarının içinde en basitlerinden biri olan K-en yakın komşu sınıflayıcı, örnek-tabanlı öğrenme yöntemlerinin bir türüdür. Öznitelik domeninde örnek vektörleri arasındaki mesafe ve benzerlik bilgisini kullanarak sınıflamayı gerçekleştirir (Cover ve Hart, 1967). Sınıflandırılması istenen bir örnek vektörü, hangi sınıfa ait olduğu bilinen diğer örnek vektörleri arasında, kendisine en yakın K-adet örnek vektörünün ait olduğu sınıflar dikkate alınarak, en çok komşu örnek hangi sınıfa ait ise, o sınıftan olarak etiketlenir. Mesafe ölçütü olarak genellikle Öklid uzaklığı kullanılır. Bu karar kuralı hesap basitliği sağlar ve çeşitli problemlerde başarılı sonuçlar vermektedir.

Eğitim örnekleri, çok boyutlu öznitelik domenindeki örnek vektörleridir. Algoritmanın eğitim süreci, sadece sınıfları bilinen eğitim vektörlerinin ve sınıf etiketlerinin hafızaya yüklenmesinden ibarettir. Fakat bu yöntemde her bir örnek vektörüne eşit ağırlık veriliyor olması, sınıflara ait örneklerin saçılma gösterdiği uygulamalarda olumsuz sonuç doğurabilmektedir (Dokur, 2006). Kullanıcı tarafından seçilen K-sayısının değerinin belirlenmesi sınıflama başarımını doğrudan etkilemektedir. KNN sınıflayıcıda komşuluğun etkisi Şekil 3.3'te görülmektedir.



Şekil 3.3 : KNN sınıflayıcıda komşuluğun etkisi (Dokur, 2006).

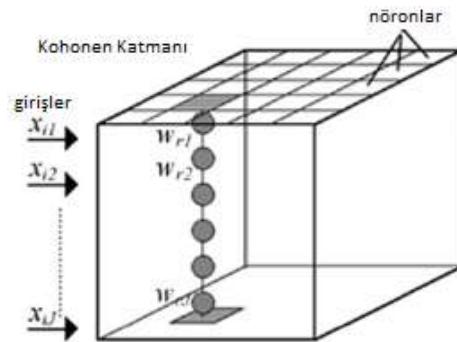
En yakın K-adet örnek vektörüne olan uzaklık değerlerinin dikkate alınmayıp, sadece komşu örnek vektörü sayısının kullanılması da bu yöntemin bir başka olumsuz yanıdır. Sınıflama kararında çoğunluk etkili olduğu için, eğer bir sınıfa ait örnek vektörlerinin sayısı, diğer sınıflara ait örnek vektörlerine göre çok daha fazla sayıda ise, o sınıf baskın çıkmakta ve bu durum sınıflama başarımını düşürmektedir.

K-sayısının en uygun değerinin seçimi, kullanılan veri kümesine bağlıdır. Genel olarak K'nın büyük bir değer seçilmesi sınıflama doğruluğunu arttırabilir. Optimum K-sayısının belirlenmesi için çeşitli araştırmalar mevcuttur (Hall ve diğ., 2008; Samworth, 2012). K-sayısının =1 olarak alındığı durum ise, “en yakın komşu” algoritması olarak adlandırılır.

3.3.2.3 Kohonen'in öz-düzenlemeli harita sınıflayıcısı (SOM)

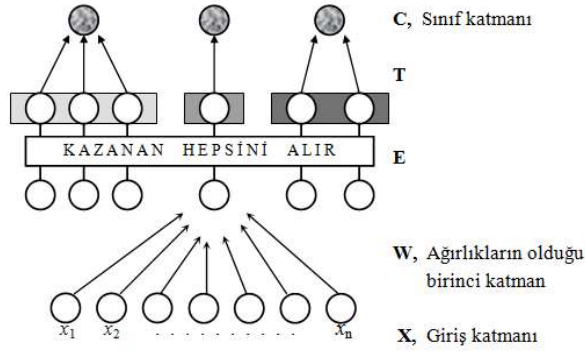
Öz-düzenlemeli harita adlı sınıflayıcı, bir çeşit yapay sinir ağı uygulaması olarak Teuvo Kohonen tarafından geliştirilmiştir. Bu ağı eğitilmesi, denetimsiz bir öğrenme süreci olarak gerçekleşir. Bu sürecin sonunda, çok boyutlu giriş domenindeki eğitim örnekleri, harita adı verilen daha düşük boyutlu, genellikle 2-boyutlu bir yapı ile temsil edilir (Kohonen, 1990). Günümüzde çok çeşitli uygulama alanlarında kullanıldığı görülmektedir. Bunları bazıları görüntü segmentasyonu De ve Guo (2015), atmosferik araştırmalar Rousi ve diğ. (2015) olarak sayılabilir.

Kohonen haritası, genellikle N^2 düğümün bulunduğu bir ızgaradan meydana gelen karesel toroidal bir uzay olarak karakterize edilir. Bu iki boyutlu domenin, her ekseninde N adet düğüm yer almaktadır (Ballabio ve Vasighi, 2012). Bu yapı, Şekil 3.4'de görülmektedir. Ancak bu durum zorunlu olmayıp, bazı uygulamalarda farklı eksenlerde farklı düğüm sayılarından oluşan yapılar da kullanılabilir.



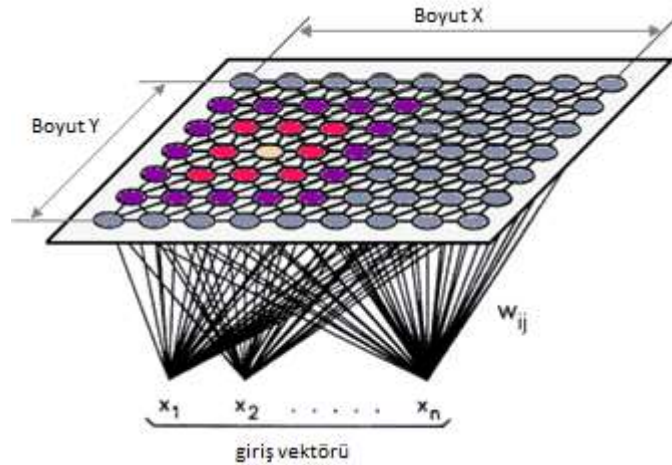
Şekil 3.4 : Kohonen SOM haritasının gösterilimi (Ballabio ve Vasighi, 2012).

Kohonen ağı, temel olarak iki katmandan meydana gelir. İlk katman, düğüm ağırlıkları ile incelenen giriş vektörü arasındaki uzaklıkları ölçmek ve hangisine en yakın olduğunu belirlemek için kullanılır. İkinci katmandaki ağırlıklar, eğitim tamamlandıktan sonra bir etiketleme işlemi ile belirlenir. İkinci katmandaki düğümlerden aynı sınıfa ait olanlar, lojik olarak OR'lanır (Dokur, 2006). SOM yapay sinir ağındaki yapısı ve katmanlar, Şekil 3.5'de görülmektedir.



Şekil 3.5 : SOM yapay sinir ağı yapısı (Dokur, 2006).

Eğitim aşamasında her düğümün ağırlıklarının, bir komşuluk fonksiyonuna göre güncellemesi bakımından, SOM diğer sinir ağlarından farklılık göstermektedir. SOM'un temel bir özelliği, çıkış katmanındaki düğümler arasında yumuşak bir rekabet yapısı bulunmasıdır. Bu şekilde, sadece kazanan düğüm değil, ona komşu olan düğümlerin değerleri de güncellenir. SOM yapay sinir ağında komşuluk ilişkileri Şekil 3.6'da görülmektedir.



Şekil 3.6 : SOM yapay sinir ağında komşuluk ilişkileri (Url-3).

SOM algoritmasındaki eğitim adımları şu şekildedir:

- Ağırlık vektörlerinin başlangıç değeri olarak rasgele seçilen konumları (veya örnek vektörlerinden bazılarının konumlarını) ata
- $t=1$
 - Eğitim kümesinden rasgele bir örnek vektörü " x_i " seç
 - Bu örnek vektörü ile ağıdaki düğümler arasındaki Öklid mesafesini hesapla

$$D_j = \sqrt{\sum_{m=1}^p (x_{im} - w_{jm}(t))^2} \quad (3.9)$$

- En yakındaki düğüm j^* yi bul
- j^* düğümü ve ona komşu olan düğümlerin ağırlıklarını, aşağıdaki 3.10 denklemini kullanarak güncelle.

$$w_{jm}(t+1) = w_{jm}(t) + \beta(t) \cdot (x_{im} - w_{jm}(t)) \quad (3.10)$$

t : yineleme adım numarası

$\beta(t)$: eğitim süreci boyunca değeri azalan, (0,1) aralığında sabit bir sayı

- Yukarıdaki adımları maksimum yineleme sayısına ulaşıncaya kadar tekrarla ($t=t+1$)

Eğitim süreci tamamlandığında, her bir düğüme en yakın olan eğitim vektörleri içinde en fazla hangi sınıftan örnek varsa, o düğüm o sınıfa ait olarak etiketlenir. Sınıflanmak istenen yeni bir test vektörü SOM ağına uygulandığında, bu vektör ile ağı çıkış düğümleri arasındaki uzaklıklar kontrol edilir ve hangi düğüme yakın ise, bu test vektörü, o düğümün sınıfına ait olarak etiketlenir.

3.3.3 Doğadan esinlenen sınıflama yöntemleri

Giriş bölümünde kapsamlı olarak açıklanmış olduğu şekilde, son birkaç on yıllık dönemde doğadan esinlenen algoritmaların sınıflama problemlerinin çözümünde giderek artan bir yaygınlıkta kullanılmakta olduğu görülmektedir. Hayvan topluluklarının davranışlarına benzetim yaparak geliştirilen algoritmalar sürü zekâsı (swarm intelligence, SI) algoritmaları olarak da adlandırılır. Parçacık sürü optimizasyonu (PSO), karınca kolonisi optimizasyonu (ACO), yapay arı kolonisi (ABC) algoritmaları bu tür sürü zekâsı tabanlı, doğadan esinlenen yöntemlere birer

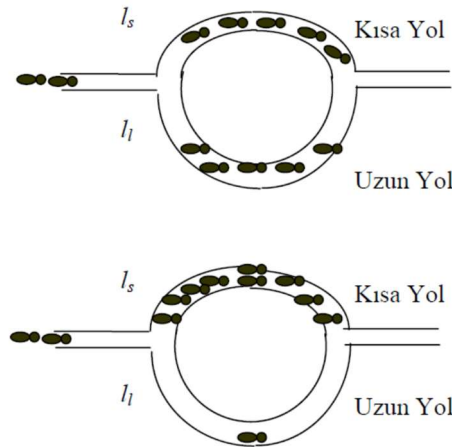
örnektir. Bir başka doğadan esinlenen yöntem grubu olan evrimsel algoritmalar (EA) doğal seçim sürecinden esinlenerek çalışırlar. Bunlara bir örnek olarak, genetik algoritma (GA) verilebilir.

Yapay arı kolonisinin de dahil olduğu doğadan esinlenen algoritmalarından biri olan karınca kolonisi optimizasyon algoritmasının yapısı aşağıda açıklanmıştır.

3.3.3.1 Karınca kolonisi optimizasyonu tabanlı sınıflayıcı (ACO)

Gerçek karınca kolonilerinin kolektif davranışlarından esinlenen Dorigo, karınca sistemi (AS) adını verdiği yöntemi tanıtmış Dorigo ve diğ. (1991) ve daha sonra da karınca kolonisi optimizasyonu adı altında bir algoritma olarak geliştirmiştir (Dorigo ve diğ, 1996; Dorigo ve Stützle, 2004). Karınca sistemi optimizasyon algoritması, kolonideki bireylerin birbirleri ile olan düşük seviyeli etkileşimlerinden hareketle global optimal çözüme yakınsayan, birçok elemanlı sistem uygulamasıdır. Çeşitli problemlerin çözümünde başarı ile kullanılmış olduğuna dair örnekler görülmektedir.

Gerçek hayatta karıncalar, gıda kaynağına giden en kısa yol hakkında her hangi bir ön bilgiye sahip olmamalarına karşın, Şekil 3.7’de görüldüğü gibi yuva ile gıda kaynağı arasındaki en kısa yolu bulabilmektedir.



Şekil 3.7 : Karıncaların yuva ile gıda kaynağı arasındaki kısa yolu bulup kullanması (Nizam, 2008).

Karınca kolonilerinin bu özelliğinin arkasındaki nedenler incelendiğinde, karıncaların yol üzerinde bırakılmış olan feromon (koku) miktarı aracılığı ile birbirleriyle iletişim kurdukları anlaşılmıştır. Her bir karınca ilerlerken yer üzerinde bir miktar feromon bırakmaktadır. Diğer karıncalar da bu kokuyu algılar ve mevcut

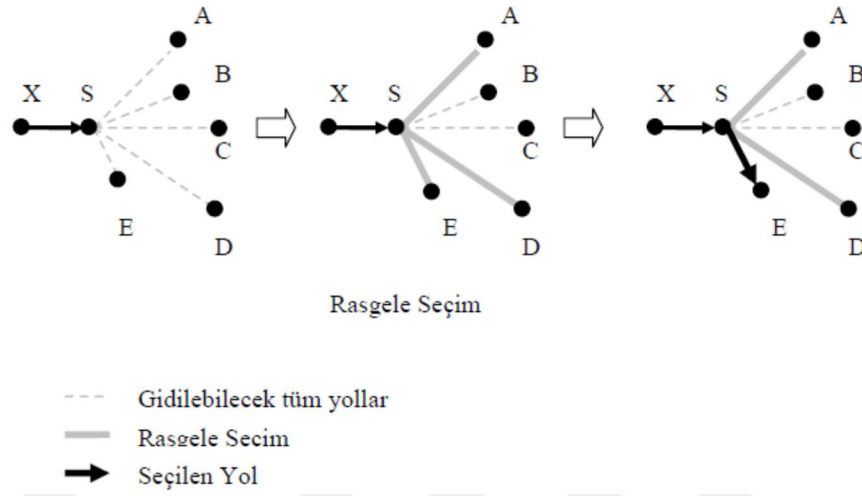
feromon miktarı ile orantılı bir miktarda o yolu takip etme eğiliminde olur. Aynı yolu takip eden diğer karıncalar da kendi feromonlarını bırakarak, o iz üzerindeki toplam koku miktarını arttırıcı etkide bulunur. Doğal ortamda, yer üzerine bırakılan feromonların buharlaşması ile yol üzerindeki koku miktarı zaman geçtikçe azalmakta, yeni karıncaların geçişi ile feromon miktarı takviye edilmezse, bir süre sonra tamamen kaybolmaktadır.

Yuva ile gıda kaynağı arasındaki yol üzerinde bir engel olması halinde, Şekil 3.7’de görüldüğü gibi, bu engelin iki ayrı tarafından dolaşarak hedefe ulaşılması mümkündür. Her iki tarafa da başlangıçta eşit olasılıkla yönelim olacaktır. Karıncaların hızlarının yaklaşık olarak birbiri ile aynı olduğu varsayılarak, birim zamanda kısa yol üzerinden geçen karıncaların sayısı, uzun yoldan geçenlerden daha fazla olacaktır. Her bir karıncanın yol üzerinde yaklaşık olarak eşit miktarda feromon bıraktıkları kabul edilerek, kısa yol üzerinde biriken feromonların miktarı daha fazla olacaktır. Bu şekilde bir pozitif geri besleme süreci yaşanarak, bir süre sonra kısa yolu tercih eden karıncaların sayısı artmakta, onların da eklediği feromonların katkıları ile kısa yol tamamen baskın duruma geçmektedir (Karaboğa, 2004).

Doğadaki karınca kolonilerinin, hedefe ulaşan en kısa yolu bulmak için kullandıkları bu etkileşimden hareketle, karınca koloni optimizasyonu (ACO) adı altında bir algoritma geliştirilmiştir. Gerçek hayatta görülen karınca bireyleri arasında koku aracılığı ile gerçekleşen iletişim, koku miktarının fazla olduğu yolların öncelikle tercih edilmesi ve kısa yollar üzerinde koku miktarının artması gibi özellikler ACO algoritmasında kullanılmıştır. Ancak bu algoritmada yapay karıncalar, gerçek hayattakinden farklı olarak ayrık zamanlı bir çevrede yaşamakta, belirli bir hafızaya sahip olabilmekte ve problemin çözümü ile ilgili bazı ön bilgilere sahip olabilmektedir.

ACO algoritmasını esas alan çeşitli öbikleme yöntemleri literatürde yer almaktadır (Tsai ve diğ, 2004; Kuo ve diğ, 2005). Bunlardan Tsai ve diğ, (2004) sundukları yöntemde, algoritmanın başında karıncalar rasgele noktalara yerleştirilirler. Bir döngü içinde önceden belirlenmiş bir yineleme sayısı kadar, her bir karınca rasgele seçilen diğer noktaları ziyaret eder. Gittikleri yolun uzaklığı ile ters orantılı olarak yolun üzerine feromon bırakırlar. Bu feromon miktarı zamanla azalarak belli bir süre içinde kaybolacak şekilde modellenmiştir. Yineleme adımları ilerledikçe birbirine yakın olan noktalar arasındaki feromon miktarı artar ve o yollar giderek daha fazla karınca tarafından tercih edilir. Bu şekilde ilerleyen adımlarda öbekler belirgin bir

şekilde birbirinden ayrılır. Algoritma içinde, karıncaların gidilecek yolu seçmeleri süreci Şekil 3.8’de görülmektedir.



Şekil 3.8 : Karıncaların gidilecek yolu seçmeleri süreci (Tsai ve diğ, 2004).

ACO algoritmasının başlıca adımları aşağıda verilmiştir:

- Adım 1 : Eğitim kümesindeki verilerin %10’u kadar karıncayı rasgele konumlara yerleştir.
- Adım 2 : Karıncaların ziyaret edebileceği noktaları rasgele seç.
- Adım 3 : Karıncaları, rasgele seçilen yollar içinde feremon miktarı en yüksek olana gönder.
- Adım 4 : Gidilen yol uzunluğu ile ters orantılı olarak feremon miktarını arttır.
- Adım 5 : Tüm yollardaki feremon miktarını zamanla orantılı olarak azalt.
- Adım 6 : Toplam yineleme sayısı kadar adım 2-5’deki işlemleri tekrarla.
- Adım 7 : Yollarda birikmiş olan feremon miktarına göre öbeklemeyi gerçekleştir.
- Adım 8 : Küçük öbekleri, kendilerine yakın olan büyük öbeklerle birleştir.

Öbekleme işlemi tamamlandığında, her bir öbekte yer alan eğitim örnekleri en çok hangi sınıftan ise, o öbek o sınıfa ait olarak etiketlenir. Öbekleme işlemi gerçekleştirildikten sonra sınıflama işleminin yapılması amacıyla K-en yakın komşular sınıflayıcısı kullanılabilir. K-en yakın komşular yönteminde, hangi sınıfa ait olduğu sınıflandırması yapılacak olan test örnekleri, kendisine Öklid uzaklığı olarak en yakın eğitim örneklerine bakılarak, hangi öbeğe ait iseler, o öbeğin sınıfındandır diye etiketlenir.

3.4 Sınıflama Performans Ölçütleri

Sınıflama algoritmalarının performanslarının değerlendirilmesinde kullanılan başlıca ölçütler aşağıda verilmiştir. Bu ölçütlerin hesaplanmasında kullanılan unsurların anlamları ve nasıl hesaplandıkları da açıklanmıştır.

Sınıflamayı gerçekleştiren algoritma, bütün örnek vektörlerine birer sınıf etiketi atar. Bunların bazıları doğru iken, bazıları da hatalı olarak başka bir sınıfa ait olarak etiketlenir. Algoritmanın yapmış olduğu bu işlemler doğru ya da yanlış olarak nitelendirilmekten başka, doğru pozitif, yanlış pozitif, doğru negatif vb. şekilde de nitelendirilir. Bu nitelendirmelerin tümünün yer aldığı doğruluk çizelgesi, Çizelge 3.1’de verilmiştir.

Çizelge 3.1 : Sınıflandırma algoritmasının yapmış olduğu etiketlemelere ait doğruluk çizelgesi.

		Etiketleme	
		P	N
Gerçek	P	TP	FN
	N	FP	TN

Literatürde yer alan performans ölçütlerinde kullanılan unsurlar şunlardır:

TP: Doğru Pozitif (True Positive)

FP: Yanlış Pozitif (False Positive)

TN: Doğru Negatif (True Negative)

FN: Yanlış Negatif (False Negative)

Çizelge 3.1’de görüldüğü şekilde, eğer sınıflandırıcı bir örnek vektörünü doğru olarak o sınıfa ait diye etiketlemişse, bu doğru pozitif bir sınıflamadır. Eğer o sınıftan olduğu halde farklı bir sınıfa ait diye etiketlenmişse bu yanlış negatif bir sınıflamadır. Farklı bir sınıfa ait bir örnek vektörü o sınıfa ait diye etiketlenmişse yanlış pozitif, son olarak da, farklı sınıftan örnekler farklı sınıftan diye etiketlenmiş ise, doğru negatif sınıflama yapılmıştır.

Sınıflama algoritmalarında en yaygın kullanılan performans ölçütü, sınıflama başarımıdır (classification accuracy). Toplam “n” adet örnek vektörünün, “k” farklı sınıfa sınıflandırıldığı bir algoritmada, sınıflama başarımı, doğru sınıflanmış olan örnek vektörlerinin sayısının, toplam örnek vektörü sayısına oranıdır. Yüzde olarak

ifade edilen sınıflama başarımına ait hesaplama yöntemi, denklem 3.11’de görülmektedir.

$$\text{Sınıflama Başarımı} = \frac{\sum_{i=1}^k TP_i}{n} \quad (3.11)$$

Duyarlık (Se, sensitivity), özel olarak her bir sınıfa ait örneklerin hangi doğrulukta bir yüzde ile sınıflandırılmış olduğu gösteren bir performans ölçütüdür. Sınıflama başarımından sonra en yaygın kullanılan performans ölçütüdür. Sınıflama başarımı ölçütü, sınıflayıcının sistem bazında performansını gösterirken, duyarlık ölçütü her bir sınıf için ayrı olarak sınıflama doğruluğunu vermektedir. O sınıfa ait doğru olarak etiketlenen örneklerin sayısı, o sınıfa ait toplam örnek sayısına oranlanarak hesaplanır. Duyarlık yükseldikçe, o sınıftaki örnekler içinden doğru olarak sınıflanmış olan örneklerin sayısı artar. Bu ölçütün %100 olması, o sınıftaki tüm örneklerin doğru olarak etiketlendiği anlamına gelir. Duyarlık hesabı denklem 3.12’de görülmektedir.

$$Se = \frac{TP}{TP + FN} \quad (3.12)$$

Diğer iki performans ölçütü olan özgüllük (Sp, specificity) ve pozitif öngörü (Ppr, positive predictivity) için hesaplama yöntemleri, sırasıyla denklem 3.13 ve 3.14’de verilmiştir. Özgüllük, doğru olarak o sınıftan değil diye etiketlenen örneklerin sayısı, o sınıftan olmayan toplam örnek sayısına oranlanarak hesaplanır. Bu oranın yükselmesi, o sınıftan etiketlenenler içinde başka sınıftan olup da, yanlışlıkla bu sınıftan olarak etiketlenen örneklerin azaldığını gösterir. Özgüllük %100 olursa, hiçbir örnek yanlışlıkla o sınıftan olarak etiketlenmemiştir.

$$Sp = \frac{TN}{TN + FP} \quad (3.13)$$

Pozitif öngörü ise, doğru olarak o sınıftan diye etiketlenmiş örneklerin sayısı, o sınıftan olarak etiketlenen toplam örnek sayısına oranlanarak hesaplanır. Bu oranın yükselmesi, o sınıftan diye etiketlenmiş tüm örnekler içinde doğru olarak o sınıftan diye etiketlenmiş olan örneklerin sayısının arttığını gösterir. Bu oran %100 olursa, o sınıftan diye etiketlenmiş tüm örnekler gerçekten o sınıftandır.

$$Ppr = \frac{TP}{TP + FP} \quad (3.14)$$

Üç ayrı sınıfa ait örneklerin sınıflandırılması sonucu elde edilen sınıflama matrisi (confusion matrix) Çizelge 3.2’de görülmektedir.

Çizelge 3.2: Üç sınıf için sınıflama matrisi.

		Etiketlenen sınıf		
		A	B	C
Gerçek sınıf	A	S_{11}	S_{12}	S_{13}
	B	S_{21}	S_{22}	S_{23}
	C	S_{31}	S_{32}	S_{33}

Üç ayrı sınıfa ait hesaplamalar aşağıdaki şekilde yapılmaktadır (Dilmaç ve diğ, 2014).

$$TP_A = S_{11} \quad TP_B = S_{22} \quad TP_C = S_{33}$$

TP_i : i. sınıfa ait doğru pozitif

$$FN_A = S_{12} + S_{13} \quad FN_B = S_{21} + S_{23} \quad FN_C = S_{31} + S_{32}$$

FN_i : i. sınıfa ait yanlış negatif

$$FP_A = S_{21} + S_{31} \quad FP_B = S_{12} + S_{32} \quad FP_C = S_{13} + S_{23}$$

FP_i : i. sınıfa ait yanlış pozitif

$$TN_A = S_{22} + S_{23} + S_{32} + S_{33} \quad TN_B = S_{11} + S_{13} + S_{31} + S_{33}$$

$$TN_C = S_{11} + S_{12} + S_{21} + S_{22}$$

TN_i : i. sınıfa ait doğru negatif

4. ABC VE MABC ALGORİTMALARI İLE SINIFLANDIRMA İŞLEMİ

Doğadaki gerçek bir arı kolonisinde yaşayan arılar arasında bir iş bölümü vardır. Bazı arılar gıda kaynağı arama ve gıda toplama işlevi gerçekleştirir; aynı zamanda üzerinde çalıştıkları gıda kaynağına ait konum bilgisini kovandaki diğer arılara aktarırlar. Bu araştırmacı davranışa ait minimal modelde üç temel eleman bulunmaktadır: Gıda kaynakları, bir gıda kaynağı üzerinde çalışan işçi arılar ve henüz bir gıda kaynağı üzerinde çalışmaya başlamamış olan arılar. Bu davranışta bir gıda kaynağı üzerinde çalışma ve o gıda kaynağının terk edilmesi olmak üzere iki ayrı aşama yer alır. Üzerinde çalışılan gıda kaynağının değeri, kovana yakınlığı, sahip olduğu nektar yoğunluğu veya zenginliği, nektarın elde edilmesinin kolaylığı gibi çeşitli unsurlara sahiptir.

Her bir işçi arı (employed bee), bir gıda kaynağı üzerinde çalışır. O kaynağa ait nektarı kovana getirirken, aynı zamanda o gıda kaynağının konumu ve gıdanın miktarı bilgilerini hafızasında saklar ve kovanda bekleyen diğer arılara bu bilgileri aktarır.

Henüz bir gıda kaynağı üzerinde çalışmaya başlamamış olan arılar, hangi konumlardaki gıda kaynağından yararlanacaklarını belirlemeye çalışırlar. Kullandıkları gıda kaynağı konumu belirleme yöntemine göre kendi içlerinde iki türe ayrılırlar. Bazıları, işçi arıların aktardığı bilgileri kullanarak hâlihazırda üzerinde çalışılmakta olan kaynaklardan birine giderek o kaynağın üzerinde çalışırlar. Bunlara gözcü arı (onlooker bee) adı verilir. Gözcü arılar, hangi işçi arının kaynağına gideceğine, o kaynağın uygunluk değeri ile orantılı bir olasılık ile karar verir. En yüksek uygunluk değerine sahip kaynağın üzerinde daha fazla sayıda arının çalışması söz konusudur. Fakat düşük uygunluk değerine sahip olan kaynaklar üzerinde de, daha az sayıda olsa da çalışma yapılmaktadır.

Gözcü arılardan farklı olarak, herhangi bir ön bilgi kullanmaksızın rasgele bir şekilde gıda kaynağı arayan arı olarak kâşif arı (scout bee) adı verilir. Herhangi bir gıda kaynağındaki nektar miktarı tükendiğinde, o gıda kaynağı terk edilir ve orada çalışan işçi arı, kâşif arıya dönüşerek rasgele bir şekilde yeni gıda kaynakları araştırır.

Sürü zekâsının önemli bir özelliği olan kendi kendine organize olabilme karakteristiği bu toplulukta şu şekilde açıklanabilir:

Pozitif geri besleme: Bir gıda kaynağındaki nektar miktarı (uygunluğu) ne kadar çok ise, o gıda kaynağının üzerinde çalışmak üzere seçen gözcü arı sayısı da o kadar çoktur.

Negatif geri besleme: Bir gıda kaynağındaki nektar miktarı az ise, benzer şekilde daha az sayıda gözcü arı tarafından seçilir.

Dalgalanmalar: Kâşif arılar, herhangi bir ön bilgi kullanmadan rasgele konumlarda arama yaparak yeni gıda kaynakları bulmaya çalışırlar.

Çoklu etkileşimler: İşçi arılar, yiyecek kaynakları ile ilgili bilgileri kovandaki diğer arılara dans adı verilen özel bir uçuş şekli ile aktarırlar.

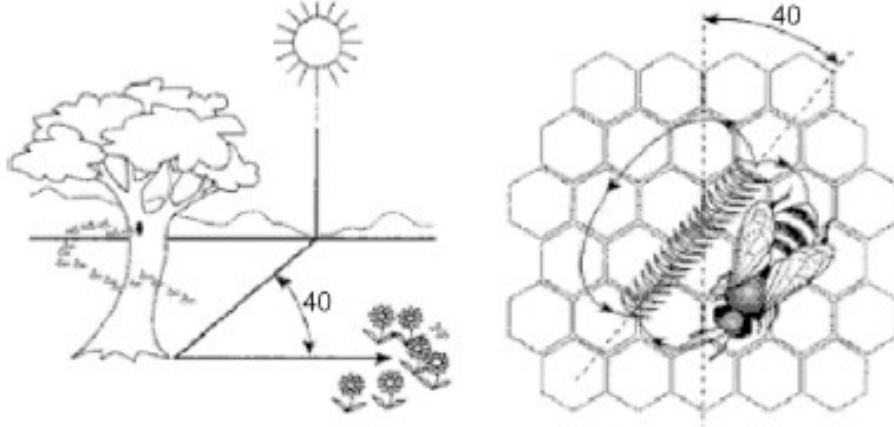
4.1 ABC Algoritması

Bal arısı kolonisinin bu karakteristik özelliklerinden hareketle Yapay Arı Kolonisi (ABC) algoritması adında yeni bir optimizasyon algoritması Karaboğa tarafından geliştirilerek sunulmuştur (Karaboğa, 2005). Bu algoritma, doğadan esinlenen (nature inspired) sürü zekâsı tabanlı çalışan bir algoritmadır. Bal arılarının yiyecek arama davranışları modellenerek geliştirilmiş bir problem çözme yöntemidir. Gerçek arılar ve modellemede kullanılan yapay arılar arasındaki benzetim ve karşılaştırmalar Çizelge 4.1’de verilmiştir.

Çizelge 4.1 : Gerçek arılar ve yapay arılar arasındaki benzetim ve karşılaştırmalar.

Gerçek arılar	Yapay arılar
Sürekli zaman çevrimleri	Ayrık zaman çevrimleri
Hafızada bilgi saklama	Hafızada bilgi saklama
En yüksek miktarda balın elde edilmesi	En uygun çözümün bulunması
Kovanın etrafındaki gıda kaynakları	Problemin olası çözümleri
Nektar miktarı (faydalılık)	Çözümün uygunluğu
Üç boyutlu arama domeni	p-boyutlu arama domeni
Negatif ve pozitif geri beslemeler	Negatif ve pozitif geri beslemeler
Çoklu etkileşimler, dalgalanmalar	Çoklu etkileşimler, dalgalanmalar

İşçi arılar gıda kaynağındaki nektar miktarını öğrenir, hafızasında tutar ve kovandaki diğer arılarla sallanma dansı (waggle dance) adı verilen özel bir uçuş/dans aracılığıyla bu bilgiyi ve gıda kaynağının konumunu diğer arılarla paylaşır (Kang ve diğ, 2009). İşçi arıların bilgi aktarım dans uçuşu Şekil 4.1’de görülmektedir.



Şekil 4.1 : İşçi arıların dans uçuşu ile gıda kaynağının konumu bilgisini diğer arılara aktarması (Kang ve diğ, 2009).

Bütün işçi arıların aktardığı bilgileri değerlendiren gözcü arılar nektar miktarı daha fazla olan (daha yüksek uygunluk değerine sahip olan) gıda kaynağına daha yüksek olasılıkla giderek, o kaynak üzerinde çalışırlar. Bu yöntemle arı kolonisi mümkün olan en kısa süre içinde en çok miktarda gıda temin edebilmektedir.

Bu benzetimde, doğadaki bir gıda kaynağı, problemin olası bir çözümünü temsil eder. Çözümün uygunluk değeri ise, gıda kaynağındaki nektar miktarı ile temsil edilir. Olası çözümler iyi olan ve olmayan bütün çözümler anlamında olup, amaç en iyi çözümün konumunu belirlemektir.

4.1.1 ABC algoritmasının yapısı

Sürü zekâsı tabanlı, en iyi çözümün bulunmasını hedefleyen bir arama süreci olan bu algorithmada toplam N adet olmak üzere, üç çeşit gıda araştırmacı arı bulunmaktadır:

- İşçi arılar: N_e adet
- Gözcü arılar: N_o adet
- Kâşif arılar

Orijinal ABC algoritmasında, N_e ve N_o birbirine eşit sayıda olarak kullanılmaktadır, $N_e = N_o$ 'dur ve ikisinin toplamı kolonideki arı sayısına eşittir $N = N_e + N_o$

(Karaboğa, 2005). Kâşif arılar için ayrıca bir nüfus kullanılmamasının nedeni aşağıda açıklanmıştır.

İşçi arılar: Her bir işçi arı, bir gıda kaynağı üzerinde çalışır. En iyi çözümü bulunmak istenen problemin, olası bir çözümünü temsil eden gıda kaynağının konumu ve uygunluk değeri, o işçi tarafından belirlenir ve hafızasında saklanır. Bu işçi arı, o konumun yakınlarındaki başka gıda kaynaklarını da kontrol eder. Daha yüksek nektar içeren konum hangisi ise o konumun üzerinde çalışmaya devam eder. Bu işçi arı, kovana döndüğünde özel bir dans ile üzerinde çalıştığı gıda kaynağına ait konum ve uygunluk değeri bilgilerini kovanda bekleyen gözcü arılara aktarır. Belirli bir sayıda arama boyunca daha iyi konum bulamayan işçi arı, o konumu terk ederek kâşif arıya dönüşür ve rasgele bir konumdan arama işlemine devam eder.

Gözcü arılar: Kovanda bekleyip işçi arıların getirdiği bilgileri gözleyen arılara gözcü arı adı verilir. Kovana gelen işçi arılardan birini seçip onun üzerinde çalıştığı gıda kaynağına giderek, o kaynak civarındaki başka gıda kaynaklarını kontrol eder. Hangi işçi arının konumunun seçileceği, o konumun uygunluk değeri ile orantılı bir olasılık fonksiyonu ile belirlenir. Bu benzetimde “ q .” işçi arının seçilme olasılığı P^q ile gösterilir. Daha yüksek nektar içeren, yani daha yüksek uygunluğa sahip olan konumlara daha çok sayıda gözcü arı giderek araştırma yapar. Bununla birlikte düşük uygunluğa sahip olan konumlara da düşük olasılıkla da olsa gidilerek oralarda da araştırma yapılması sayesinde bölgesel en iyi konumlara takılarak global en iyi konumun bulunamaması problemi önlenmiş olur.

Kâşif arılar: Herhangi bir ön bilgi kullanmaksızın gıda kaynağı arayan arılar kâşif arı olarak adlandırılır. Bu arılar kovan civarında rasgele olarak araştırma yapar, bir gıda kaynağı bulunca o gıda kaynağını değerlendirmeye başlar ve artık işçi arı olarak adlandırılırlar. Orijinal ABC algoritmasında arama süreci rasgele konumlardan başladığı için, koloninin yarısı kâşif arı olarak rasgele konumlara giderler. Arama sürecinin herhangi bir yineleme adımında oluşan kâşif arı, aynı adım içinde rasgele bir konum üzerinde çalışmaya başlayarak işçi arıya dönüştüğü için, yapay arı kolonisinin belirli bir nüfusunu oluşturmaz.

Yapay arı kolonisindeki arı sayısı “ N ”, kullanıcı tarafından seçilen bir kontrol parametresidir. N ’nin daha büyük bir sayı olarak seçilmesi, daha çok sayıda konumun eş zamanlı olarak araştırılmasına sağlamakta, fakat toplam hesaplama

süresini arttırmaktadır. N_e ile N_o 'nun birbirine eşit olarak seçilmemesinin algoritma sonuçları üzerindeki etkisi bölüm 4.2.4 de incelenmiştir.

Bütün arıların araştırma aktiviteleri bir çevrim içinde yinelenir. Bu çevrimin önceden belirlenen yineleme sayısına “maksimum çevrim sayısı (MCN)” adı verilir. ABC algoritmasındaki bu kontrol parametresinin artan değerlerinde daha yüksek ortalama başarı elde edildiği görülmüştür. Bir işçi arının üzerinde çalışmakta olduğu gıda kaynağındaki nektar miktarı, bir başka deyişle üzerinde çalışılan çözümün uygunluk değeri, belirli bir yineleme boyunca hiç artmıyor ise, bölgesel en iyi konum olduğu kabul edilerek, o konum terk edilir ve rasgele bir konumdan arama sürecine devam edilir. Bu yineleme sayısı “Limit” olarak adlandırılır ve ABC algoritmasında bir kontrol parametresi olarak kullanılır. MCN ile Limit parametreleri arasındaki ilişki ve sonuç üzerindeki etkileri bölüm 4.2.4’de açıklanmıştır.

Bir kâşif arı, p-boyutlu bir uzayda rasgele bir şekilde yeni bir gıda kaynağı konumu $\mathbf{X}^q(t)$ belirleyeceği zaman 4.1 denklemini kullanır. $\mathbf{X}^q(t)$, algoritmanın t. yineleme adımındaki öbek merkez koordinatlarından oluşan, k-satırlı, p-sütunlu bir matristir. Bu matrisin detayları, p=4, k=3 durumu için, Şekil 4.2’de görülmektedir.

$$\mathbf{X}^q = \begin{bmatrix} \underline{x}_1^q \\ \underline{x}_2^q \\ \underline{x}_3^q \end{bmatrix} = \begin{bmatrix} x_{11}^q & x_{12}^q & x_{13}^q & x_{14}^q \\ x_{21}^q & x_{22}^q & x_{23}^q & x_{24}^q \\ x_{31}^q & x_{32}^q & x_{33}^q & x_{34}^q \end{bmatrix}$$

$$q = 1, \dots, N_e$$

$$\underline{x}_1^q = [x_{11}^q \quad x_{12}^q \quad x_{13}^q \quad x_{14}^q]: \text{birinci öbeğin öbek merkez koordinatları}$$

$$\underline{x}_2^q = [x_{21}^q \quad x_{22}^q \quad x_{23}^q \quad x_{24}^q]: \text{ikinci öbeğin öbek merkez koordinatları}$$

$$\underline{x}_3^q = [x_{31}^q \quad x_{32}^q \quad x_{33}^q \quad x_{34}^q]: \text{üçüncü öbeğin öbek merkez koordinatları}$$

Şekil 4.2 : Dört boyutlu uzayda, üç öbekli veri kümesi için, q. işçi arının (çözümün) konumu.

Arama çevriminin başlangıcında $\mathbf{X}^q(1)$, denklem 4.1 kullanılarak belirlenir.

$$x_{gj}^q(t) = o_{jmin} + ((o_{jmax} - o_{jmin}) \times rand[0,1]) \quad (4.1)$$

o_{jmin} : $\mathbf{O}$ kümesindeki örüntülerin j. boyutundaki özniteliklerinin minimum değeri

o_{jmax} : $\mathbf{O}$ kümesindeki örüntülerin j. boyutundaki özniteliklerinin maksimum değeri

$g = 1, 2, \dots, k$; $j = 1, 2, \dots, p$; $q = 1, 2, \dots, N_e$

Arıların arama uzayının dışına çıkmasını önlemek için $\mathbf{X}^q$ 'nın her elemanı $[o_{jmin}, o_{jmax}]$ aralığı ile sınırlandırılmıştır.

Bir arının konumunu gösteren $\mathbf{X}^q$ 'nın uygunluk değerini hesaplamak için çeşitli fonksiyonlar kullanılabilir. Öbekleme probleminde bu amaçla denklem 4.2 kullanılır.

$$F(\mathbf{X}^q) = \frac{1}{\sum_{i=1}^n \min\{\text{ÖU}(\underline{o}_i, \underline{x}_g^q) | g=1,2,\dots,k\}} \quad (4.2)$$

n : örüntü (örnek vektörü) sayısı

k : öbek sayısı

q : işçi arı numarası

Bu denklemde öncelikle her bir örüntünün öbek merkezlerine olan uzaklıkları hesaplanır ve en yakın olan öbeğe olan uzaklığı kaydedilir. Daha sonra bütün örüntülerin en yakın öbeğe olan uzaklıklarının toplamının tersi uygunluk değeri olarak kullanılır. Hedeflenen işlem bu uzaklıklar toplamının minimum olmasıdır.

İşçi arılar her bir yineleme adımında, üzerinde çalıştıkları mevcut konumun yakınlarında daha iyi bir gıda kaynağı ararken, yeni konumun koordinatlarını hesaplamak için denklem 4.3 kullanılır.

$$x_{gj}^q(t+1) = x_{gj}^q(t) + \left(\varphi_{gj} \times (x_{gj}^q(t) - x_{gj}^r(t)) \right) \quad (4.3)$$

$g = 1, 2, \dots, k$ (öbek sayısı); $j = 1, 2, \dots, p$ (çözüm uzayının boyut sayısı);

$q = 1, 2, \dots, N_e$; $r = 1, 2, \dots, N_e$; $r \neq q$;

$\varphi_{gj} : \text{rand}[-1, 1]$

Bir sonraki yineleme adımında incelenecek yeni konumun koordinatlarını hesaplamak için, mevcut konumun koordinatları ile rasgele seçilen bir başka işçi arının (r . İşçi arı) konumunun koordinatları kullanılır. φ_{gj} katsayısı her yineleme adımında, her işçi arı için ve her boyut için rasgele bir değer alır. Her yineleme adımında tek bir boyuttaki değer değiştirilir. Belirlenen bu yeni konumun uygunluk değeri ile bir önceki yineleme adımıdaki uygunluk değeri karşılaştırılır. Hangisi daha yüksek uygunluk değerine sahip ise, o konum ve uygunluk değeri hafızada saklanır ve diğer konum unutulur. Her zaman daha yüksek uygunluk değerine sahip konumları tercih ederek arama süreci gerçekleştirildiği için, bu yaklaşıma “açgözlü seçim (greedy selection)” adı verilmektedir. Her bir işçi arı için yeni konum araştırma işlemi, her yineleme adımında tek bir defa gerçekleştirilir.

Eğer önceden belirlenmiş bir yineleme adımı boyunca (Limit defa aramada) uygunluk değeri daha yüksek bir konum bulunamamış ise, o konum terk edilerek, rasgele bir konumdan arama sürecine devam edilir.

Orijinal ABC algoritmasının adımları aşağıda görülmektedir:

İlk koşullama

- Veri kümesi $\mathbf{O} = \{o_1, o_2, \dots o_n\}$ 'yu kullan.
- Kontrol parametrelerinin (N , N_e , N_o , Limit, MCN) değerlerini seç.
- N_e adet arıyı rasgele konumlara yerleştir, denklem 4.1.
- N_e adet başlangıç konumuna ait uygunluk değerlerini hesapla, denklem 4.2.

döngü=1 (Çevrim'in içinde yinelemelerin başlangıcı)

tekrarla (MCN defa, tüm koloni için)

- Her işçi arı için gerçekleştir (**İşçi arı arama aşaması**)
 - Mevcut konumun yakınlarında yeni bir konum belirle, denklem 4.3.
 - Yeni konumun uygunluk değerini (F^q) hesapla.
 - Bir önceki ve yeni konumun uygunluk değerlerini karşılaştır. Daha iyi olan konumu hafızada sakla.
 - Eğer son "Limit" defa arama işleminde daha iyi uygunluk değeri olan bir konum bulunamamışsa, mevcut konumu terk et ve kâşif arıya dönüş.
- İşçi arıların konumlarına ait olasılık değerlerini (P^q) hesapla, denklem 4.4.
- Her gözcü arı için gerçekleştir (**Gözcü arı arama aşaması**)
 - P^q olasılık değerlerine göre hangi işçi arının konumuna gideceğini belirle.
 - O konumun yakınlarında yeni bir konum belirle, denklem 4.3.
 - Yeni konumun uygunluk değerini (F^q) hesapla.
 - Bir önceki ve yeni konumun uygunluk değerlerini karşılaştır. Daha iyi olan konumu hafızada sakla.
- Bu adımda en az bir işçi arı kâşif arı olmuş ise; her kâşif arı için gerçekleştir (**Kâşif arı arama aşaması**)
 - Kâşif arıyı rasgele bir konuma yerleştir, denklem 4.1.
 - Bu ana kadar belirlenen en iyi konumu hafızada sakla.

döngü=döngü+1

Eğer döngü=MCN olmuş ise dur.

İşçi arı arama aşaması tüm işçi arılar için gerçekleştirildikten sonra, bunlara ait konum ve uygunluk değeri bilgileri gözcü arılarla paylaşılır. Her bir gözcü arı, hangi işçi arının konumunun civarında arama yapacağını, işçi arıların uygunluk değerleri

ile orantılı bir olasılık ile belirler. Hangi işçi arının uygunluk değeri daha yüksek ise, gözcü arılar o işçi arının konumuna daha yüksek bir olasılık ile giderler. Bu olasılığı belirlemek için denklem 4.4 kullanılır.

$$P^q = \frac{F(X^q)}{\sum_{i=1}^{N_e} F(X^i)} \quad (4.4)$$

Bu yöntemle rulet tekerleği (Roulette Wheel) adı verilmektedir (Goldberg, 1989). Uygunluk değeri daha düşük olan konumlara gözcü arılar daha düşük olasılıkla olmakla birlikte yine de gitmekte, böylece tüm çözüm uzayının araştırılıyor olması sağlanmaktadır.

Hangi işçi arının konumu civarında arama yapacağını belirleyen gözcü arı, denklem 4.3'ü kullanarak yeni bir konum ve o konumun uygunluk değerini hesaplar. Bu yeni konum ile önceki adımdaki konumun uygunluk değerleri arasında açgözlü seçim yapılarak daha iyi konumun bilgileri işçi arının hafızasında saklanır. Gözcü arıların ayrıca bir hafızada saklama özelliği yoktur.

Eğer bu şekildeki arama işlemleri esnasında terk edilen konumlar oluşmuş ise, o konum üzerinde çalışan işçi, kâşif arıya dönüşür ve denklem 4.1'i kullanarak rasgele bir konumdan arama işlemine devam eder. Tüm bu işçi, gözcü ve kâşif arı arama aşamaları maksimum yinleme sayısı (MCN) defa tekrarlanır. Bu işlemlerin sonunda döngüden çıkılır ve o ana kadar belirlenmiş olan en iyi konum ve ona ait uygunluk değeri bilgileri algoritmanın çıkışı olarak üretilir.

4.1.2 ABC algoritması ile sınıflama

Orijinal ABC algoritması, öbekleme uygulamasında öbek merkez koordinatlarının araştırıldığı aşamada eğitim örnek vektörlerinin sınıf etiketlerini bilmeden, toplam uygunluk değerini maksimum yapmaya çalışarak bir arama süreci gerçekleştirir. Öbek sayısı bilgisi ise kullanıcı tarafından seçilmektedir. Bu nedenle bu yöntem kullanıcı tanımlamalı, denetimsiz bir öbekleme (user-defined, unsupervised) yöntemi olarak adlandırılabilir.

ABC algoritmasında öbek sayısı, bir giriş parametresi olarak kullanılmaktadır. Öbekleme işleminde doğru sayıda öbek ile çalışılmakta olduğunun belirlenmesi için literatürde Dunn, Davies-Bouldin, Jaccard, Calinski–Harabasz gibi çeşitli geçerlilik indisleri (validity indices) belirleme yöntemleri mevcuttur (Arbelaitz ve diğ, 2013).

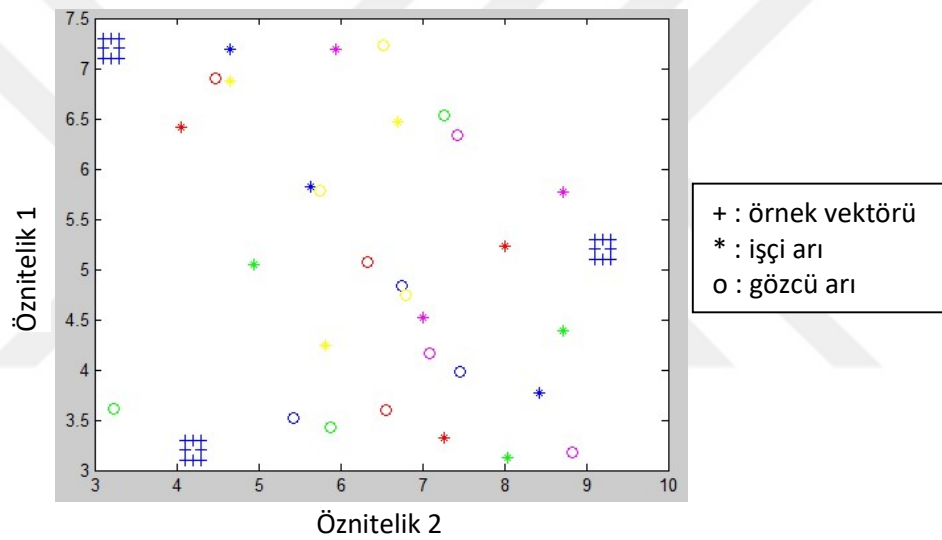
Ancak tüm bu yöntemler genellikle öbeklerdeki verilerin birbirinin içine geçmediği, uzayda ayırık olarak kümelenedikleri durumlarda doğru sonuç vermekte, örtüşen öbeklerin bulunduğu durumlarda optimum öbek sayısını doğru bir şekilde belirleyememektedir (Kovacs ve diğ, 2005). Bu tez çalışmasında Rand, Adjusted Rand, Jaccard, Fowlkes-Mallows, Silhouette, Davies-Bouldin, Calinski-Harabasz, Dunn geçerlilik indisleri kullanılmıştır (Dudoit ve Fridlyand, 2002; Halkidi ve diğ, 2001; Sharan ve diğ, 2003; Chen ve diğ, 2002; Kasturi ve diğ, 2003; Bolshakova ve Azuaje, 2003; Zhao ve Karypis, 2005; Shu ve diğ, 2003).

ABC algoritmasının sınıflama problemine uygulanmasında, öncelikle öbek merkez koordinatları belirlenmektedir. Bu amaçla, “n” örüntüden oluşan “O” veri kümesini ($O=\{o_1, o_2, \dots, o_n\}$) ve “k” öbekten oluşan “D” öbeklemesini ($D=\{C_1, C_2, \dots, C_k\}$) inceleyelim. Bu öbeklerin öbek merkez koordinatlarından oluşan herhangi bir X^q çözümü, çözüm uzayında p-boyutlu, gerçek değerli k adet vektörden oluşur. Örneğin p=4 boyutlu uzayda, k=3 öbekli bir veri kümesi için öbek merkez koordinatlarının gösterilimi Şekil 4.2’de görülmektedir.

ABC algoritmasındaki her bir işçi arı’nın konumu, çözüm uzayındaki öbek merkezlerinin koordinatlarına karşı düşmektedir ve k-satırlı, p-sütunlu bir matris ile temsil edilir. Öbekleme probleminde amaç, sınıflama başarımını en yüksek yapacak şekilde en uygun öbek merkez koordinatlarının belirlenmesidir. Öbek merkez koordinatları belirlendikten sonra her bir örnek vektörü, Öklid uzaklığı olarak kendisine en yakın olan öbek merkezinin bulunduğu öbeğe aittir diye işaretlenir. Her öbek tek bir sınıfı temsil edebileceği gibi, bir sınıf birden fazla öbekle de temsil edilebilir.

Bu tez çalışması kapsamında, ABC algoritması esas alınarak hazırlanan sınıflama yöntemi, ilk aşamada fantom uzay test veri kümeleri üzerinde denenmiştir. Öbekler arasında örtüşme olmadığı durumlarda, algoritmanın 100% başarı ile sınıflama işlemini gerçekleştirebildiği görülmüştür. Veri kümesini oluşturan öbekler arasında örtüşme oluştuğu ise sınıflama başarımında azalma görülmektedir. Bir sonraki aşamada gerçek hayattan alınmış IRIS, PIMA, BUPA gibi örnek veri kümeleri ve en geniş kapsamlı olarak da MIT-BIH veri tabanından alınmış olan ECG vuru örnekleri üzerinde sınıflama işlemleri gerçekleştirilmiştir. Bu denemelerden edinilen deneyimler doğrultusunda orijinal ABC algoritması üzerinde çeşitli değişiklik ve iyileştirmeler gerçekleştirilmiştir. Yapılan bu iyileştirmeler Bölüm 4.2’de verilmiştir.

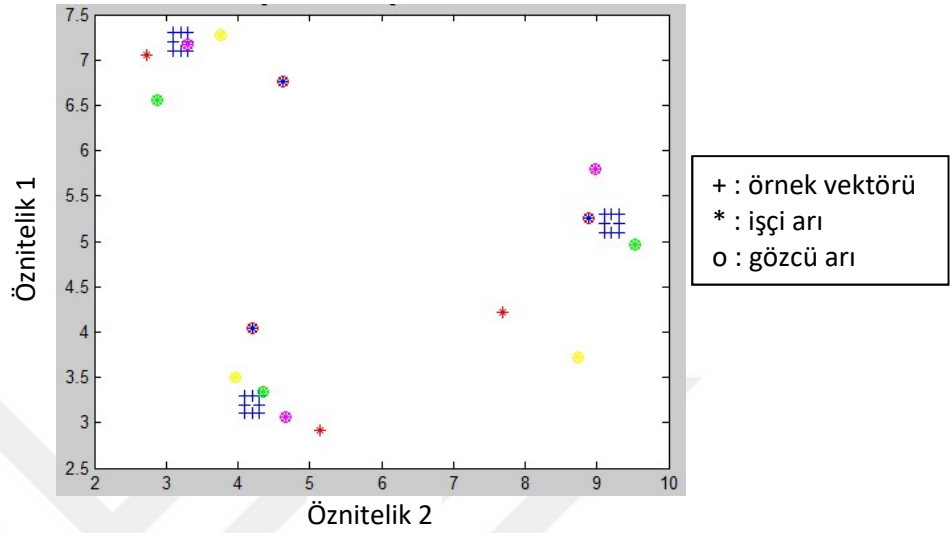
Yukarda açıklanmış olan algoritma yapısını gerçekleştiren bir program hazırlanmıştır. Bu programı test etmek için ilk etapta fantom uzayda basit veri kümeleri kullanılarak denemeler yapılmıştır. İlk aşamada 3 ayrı öbeğe ait toplam 24 örüntünün sınıflanması denenmiştir. Her örüntünün 2 boyutlu öznitelik vektörlerinden oluşan veri matrisi (24x2) giriş olarak verilmiş, 5 işçi ve 5 gözcü olmak üzere toplam 10 arıdan oluşan bir koloni ile çalışılmıştır. Rasgele seçilen ilk konumlardan hareketle, her işçi ve gözcü arı için araştırma yapılmış, optimum çözümlerin hızlı ve çok doğru bir şekilde belirlendiği gözlenmiştir. Gerçekleştirilen bilgisayar benzetiminde arama sürecindeki adımlar ve elde edilen sonuçlar aşağıda şekiller üzerinde verilmiştir. İki boyutlu sanal uzayda, 3 ayrı sınıfa ait 8 örnekten oluşan toplam 24 örüntü, 5 işçi ve 5 gözcü arıya ait başlangıç konumları Şekil 4.3'te görülmektedir.



Şekil 4.3 : İki boyutlu sanal uzayda, 24 örüntü, 5 işçi ve 5 gözcü arıya ait başlangıç konumları.

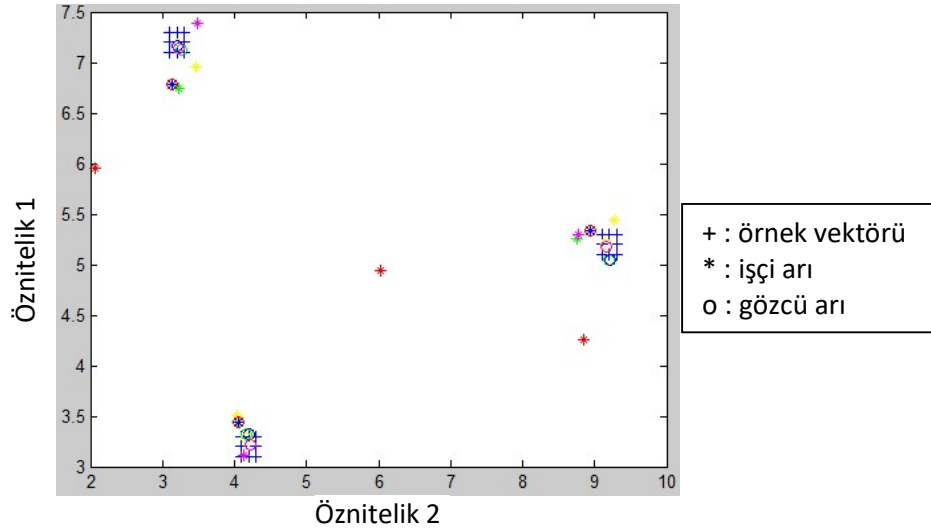
Şekil 4.3'te görülen “+” sembolleri sınıflanacak örüntüleri (örnek vektörlerini) göstermektedir. Üç sınıfa ait veriler birbirine çok yakın ve sınıflar arası uzaklık çok miktarda olduğundan gayet basit bir sınıflama problemini temsil etmektedir. Yapay arı kolonisi algoritmasının çalışma adımlarının görsel olarak incelenmesi amacıyla kullanılmıştır. Orijinal ABC Algoritması gereği, işçi ve gözcü arılar için program tarafından rasgele seçilen başlangıç konumları “*” ve “o” sembolleri ile görülmektedir. İncelenen veri kümesinde 3 öbek bulunduğu için, 3 lü renk ve şekil grupları halinde semboller mevcuttur. Örneğin 3 kırmızı renkli “*”, 3 ayrı öbek merkezinin koordinatlarını göstermekte olup, bir işçi arıyı temsil etmektedir. Benzer şekilde 3 kırmızı renkli “o” sembolü bir gözcü arıyı temsil etmektedir. Ancak renklerin aynı olması, o renkteki gözcü arının, yine o renkteki işçi arının konumuna

giderek o konumun civarında arama yapmasını gerektirmez. Hangi gözcü arının, hangi işçi arı konumunu seçerek o bölgede arama yapacağı denklem 4.4 ile hesaplanan olasılığa göre belirlenir. 50. arama adımında işçi ve gözcü arıların konumları Şekil 4.4’de görülmektedir.



Şekil 4.4 : 50. arama adımında işçi ve gözcü arıların konumları.

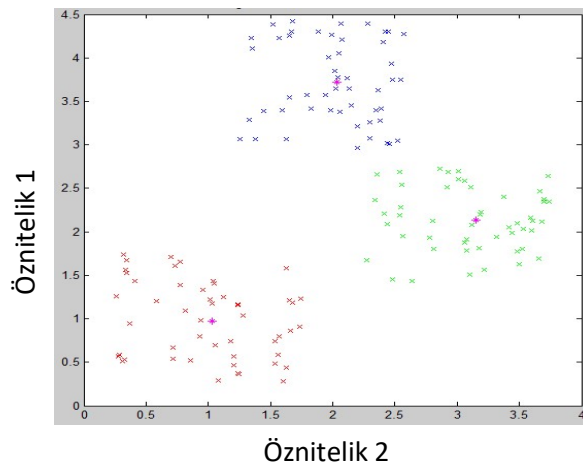
Şekil 4.4 de görüldüğü gibi, 50. arama adımında işçi arılar optimum rasgele başlangıç konumlarından hareket ederek, optimum çözüme yaklaşmış durumdadır. Her arama adımında gözcü arılar ise, işçi arıların uygunluk değeri ile orantılı bir olasılıkla, yüksek uygunluğa sahip işçi arıların konumları civarında arama gerçekleştirmektedirler. Daha düşük uygunluğa sahip olan bazı işçi arıların konumuna hiçbir gözcü arı gitmeyebilir. Bu durumun bir örneği kırmızı “*” sembolü ile gösterilen işçi arıdır. Bu işçi arının üç öbek merkezinden biri öbeklerden uzakta kaldığı için, diğer işçilere göre çok daha düşük bir uygunluk değerine sahiptir. Bu nedenle hiçbir gözcü arı (“o” sembolü), kırmızı işçi arının konumunu seçmemiştir. 100. arama adımında işçi ve gözcü arıların konumları Şekil 4.5 de görülmektedir.



Şekil 4.5 : 100. arama adımında işçi ve gözcü arıların konumları.

Şekil 4.5’de görüldüğü gibi, 100. arama adımında işçi ve gözcü arılar öbeklerin merkezine çok yaklaşmış durumdadır. Dikkat çeken bir durum kırmızı “*” işçi arının durumudur. Bu işçi arının konumu limit defa arama yapıldığı sürece hiç iyileşmemiş, bu nedenle ABC algoritması gereği kâşif arıya dönüşmüştür. Yani mevcut konumunu terk ederek rasgele seçilen yeni bir konumdan aramaya başlamıştır. Bölgesel en iyi konumlara takılma problemini çözmek amacıyla algoritmaya eklenen bu mekanizmanın sakıncaları ve tez çalışması kapsamında bu probleme yönelik geliştirilen çözüm Bölüm 4.2’de açıklanmıştır.

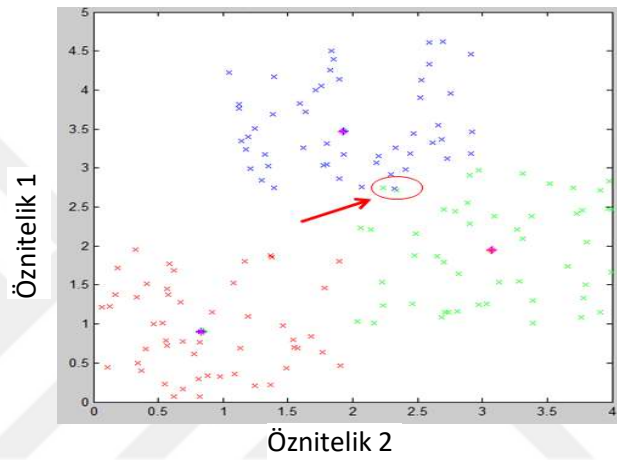
Bundan sonra benzer deneme, daha fazla sayıda örnek vektöründen oluşan bir küme üzerinde tekrarlanmıştır. Bu denemede öbek içi dağılımı büyük, öbekler arası mesafe az, ama sınıflar arası girişim olmayan üç öbekte toplam 150 örnek kullanılmıştır. Üç ayrı sınıfa ait örnek vektörleri ve bulunan öbek merkezleri Şekil 4.6’da görülmektedir.



Şekil 4.6 : Üç ayrı sınıfa ait örnek vektörleri ve bulunan öbek merkezleri.

Şekil 4.6’da görülen örnekte, üç ayrı sınıfa ait örnek vektörleri kırmızı, yeşil ve mavi “x” sembolü ile gösterilmiştir. Görsel basitliği sağlamak amacıyla sadece arama süreci sonunda bulunan en iyi konumu (öbek merkezlerini) gösteren mor “*” sembolü ile tek bir arının konumu şekilde gösterilmiştir. Bu örnekte tüm örnek vektörleri başarı ile sınıflanmıştır, sınıflama başarı oranı %100’dür.

Bir sonraki adımda benzer şekilde öbek içi dağılımı daha da büyük, öbekler arası mesafe az ve hatta sınıflar arası girişim olan 3 öbekte, yine aynı sayıda toplam 150 örnek vektörü üzerinde deneme tekrarlanmıştır. Bu veri kümesindeki örnek vektörleri ve bulunan öbek merkezleri Şekil 4.7’de görülmektedir.



Şekil 4.7 : Aralarında girişim olan üç ayrı sınıfa ait örnek vektörleri ve bulunan öbek merkezleri.

Bu örnekte, iki örnek vektörü Öklit uzaklığı olarak diğer bir öbek merkezine kendi öbek merkezinden daha yakındır. Bu nedenle bu iki örnek vektörü hatalı sınıflanmıştır, sınıflama başarı oranı %98.7’dir.

4.2 İyileştirilmiş ABC Algoritması (MABC)

Tez çalışması kapsamında ABC Algoritmasında çeşitli değişiklik ve iyileştirmeler yapılmıştır. Bu iyileştirmelere dair açıklamalar aşağıda verilmiştir.

4.2.1 İyileştirilmiş ABC algoritması parametrelerinin kurulması ve ilkleştirilmesi

Orijinal ABC algoritmasında, en iyi çözümü arama sürecine başlarken rasgele konumlardan aramaya başlanılmaktadır. Sayısal fonksiyon optimizasyonu vb. problemlerde bu yöntem kullanılabilir. Ancak sınıflama problemine özel olarak, eğitim aşamasında öbek merkez koordinatlarının belirlenmesi için çözüm uzayı

içindeki rasgele konumlardan başlanılmasının gerekli olmadığı açıktır. Öbek merkezleri, örnek vektörlerinin bulunduğu bölgede yer aldığı için, arama sürecine örnek vektörlerinin konumlarından başlanması, en uygun çözümlere daha hızlı bir şekilde ulaşılmasını sağlamaktadır.

Tez kapsamında ABC algoritmasında yapılan bir başka değişiklik, arama uzayına sınırlama getirilmesidir. Orijinal ABC algoritmasında, işçi arı ve gözcü arı arama süreçlerinde, denklem 4.3 kullanılarak yeni bir konum belirlenmesi işleminde herhangi bir sınırlandırma bulunmamaktadır. Oysa sınıflama probleminde öbek merkez koordinatlarının, örnek vektörlerinin bulunduğu bölgenin dışına çıkmayacağı rahatlıkla öngörülebilir. Bu nedenle, yeni konum belirlenirken bir öznitelik boyutunda değeri değiştirilip hesaplanan yeni koordinat, eğer o boyut için geçerli olan $o_{jmin} - o_{jmax}$ aralığının dışına çıkıyor ise, o boyut için mevcut olan o_{jmin} ya da o_{jmax} değeri, o boyut için yeni koordinat değeri olarak kullanılacak şekilde bir değişiklik yapılmıştır.

Buna göre iyileştirilmiş ABC algoritmasında, denklem 4.3 yerine, denklem 4.5 kullanılmaya başlanmıştır.

$$\begin{aligned} x_{gj}^q(t+1) &= x_{gj}^q(t) + \left(\varphi_{gj} \times (x_{gj}^q(t) - x_{gj}^r(t)) \right) \\ x_{gj}^q(t+1) &= o_{jmin} \text{ eğer } x_{gj}^q(t+1) < o_{jmin} \text{ ise} \\ x_{gj}^q(t+1) &= o_{jmax} \text{ eğer } x_{gj}^q(t+1) > o_{jmax} \text{ ise} \end{aligned} \quad (4.5)$$

4.2.2 İyileştirilmiş ABC algoritması kontrol parametreleri

İyileştirilmiş ABC algoritması, temel olarak ABC algoritmasının kontrol parametrelerini (N, MCN, Limit vb) kullanmaktadır. Fakat bunlara ilaveten bazı yeni kontrol parametreleri eklenmiş ve algoritmanın performansı iyileştirilmiştir.

Bu tez kapsamında ABC algoritmasının yapısı üzerinde yapılan en önemli iyileştirmelerden biri kâşif arı dönüşüm eşiği düzenlemesidir. Bölüm 4.1.1’de açıklanmış olduğu şekilde, eğer bir işçi arı bir gıda kaynağı üzerinde çalışırken, önceden belirlenen bir kontrol parametresi olan “Limit” defa arama yapıldığı halde uygunluk değeri iyileşmiyorsa; bu konumun yerel bir en iyi konum olduğu düşüncesi ile terk edilir ve o işçi arı, kâşif arıya dönüşerek rasgele bir konumdan arama işlemine devam edilir. Oysaki bu şekilde terk edilen konumların bazılarının aslında oldukça iyi çözümler, hatta belki de olası en iyi çözüm olabildiği görülmüştür. Böyle

durumlarda iyi konumların terk edilmesi sınıflama sonucunu olumsuz etkilemektedir. Bu problemi gidermek amacıyla, “kâşif arı dönüşüm eşiği (SCTR)” adlı bir kontrol parametresi algoritmaya eklenmiştir. Burada amaçlanan, yüksek uygunluk değerine sahip konumların uygunluk değerleri “Limit” defa aramada iyileşmese bile, o konumların terk edilmesini önlemektir.

SCTR parametresi, $[0, 1]$ aralığında değer alabilecek şekilde tanımlanmıştır. Buna göre, algoritmada arama sürecinin her yineleme adımında o ana kadarki en yüksek uygunluk değerine sahip olan konumun uygunluk değeri 100% olarak kabul edilmiş ve diğer konumların uygunluk değerleri bu değere göre ölçeklendirilmiştir. SCTR’nin örneğin $= 0.7$ seçildiği bir çalışmada, bir işçi arının uygunluk değeri, o ana kadar elde edilen en yüksek uygunluk değerinin en az 30% yakınında ise, o arama adımında, o konum terk edilmeyecektir.

SCTR’nin $= 1$ seçildiği bir çalışmada sadece en yüksek uygunluk değerine sahip konum korunacaktır. Bu parametrenin $= 0$ olarak alınması parametreyi kullanım dışı bırakmak anlamına gelmektedir.

Bu şekilde gerçekleştirilmiş olan değişiklik ve iyileştirmelerin orijinal ABC algoritması üzerindeki etkilerini denemek amacıyla, IRIS veri kümesi üzerinde ABC ve MABC algoritmaları ile testler yapılmıştır. IRIS veri kümesi, “Iris” adlı çiçek türüne ait, Setosa, Virginica ve Versicolor adında üç farklı alt-türden, 50’şer adet olmak üzere toplam 150 örnekten oluşmaktadır. Her örnek vektörü, çanak yaprağı (sepal) ve taç yaprağı (petal) genişlik ve uzunluklarından oluşan dört adet öz nitelik içerir. IRIS veri kümesinin hazırlanmasında kullanılan çiçek örnekleri Şekil 4.8’de görülmektedir.



a. Setosa

b. Versicolor

c. Virginica

Şekil 4.8 : IRIS veri kümesinin hazırlanmasında kullanılan çiçek örnekleri (Url-4).

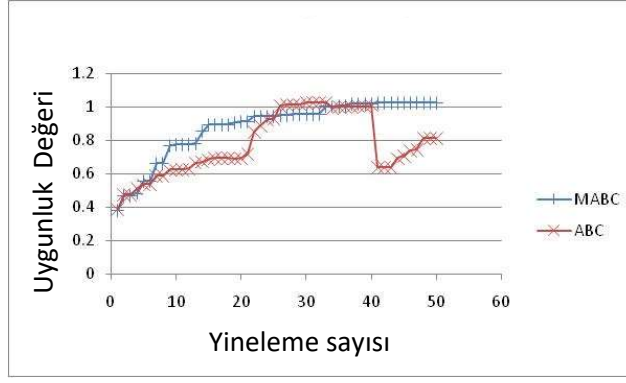
Bu veri kümesi 1936 yılında Ronald Fischer tarafından hazırlanmış (Fischer, 1936), günümüze kadar pek çok algoritmanın performans değerlendirmelerinde yaygın olarak

kullanılmıştır. Bu veri kümesi, sayısal ortamda UCI veri ambarından temin edilebilmektedir (Url-5).

IRIS veri kümesi üzerinde her iki algoritma, MCN ve Limit adlı kontrol parametrelerinin farklı değerlerinden oluşan kombinasyonlarda, toplam 500 defa denenmiştir. MCN = 10, 20, 30, 40 ve 50; Limit = 2, 4, 6, 8 ve 10 olacak şekilde, her bir ikili kombinasyon ile beşer deneme, SCTR kontrol parametresi kullanılarak toplam 125 deneme yapılmıştır. SCTR kontrol parametresi olmadan, sınır kontrol olmadan ve rasgele başlangıç konumlarından arama süreci başlatılarak toplam 375 deneme yapılmıştır. MABC ve ABC algoritmalarının IRIS veri kümesi üzerinde çalıştırılmaları esnasında en yüksek “Uygunluk Değerinin” arama süreci esnasındaki değişimi Şekil 4.9’da görülmektedir.

Her bir işçi arının uygunluk değerleri sıralanıp içlerindeki en yüksek değer, o aşamada algoritmanın belirlemiş olduğu en iyi konuma ait uygunluk değeridir. Bu uygunluk değerinin sürekli artış gösteriyor olması beklenir. Fakat Şekil 4.9’da görüldüğü gibi, ABC algoritmasında “Limit” defa arama yapıldığında iyileşme gerçekleşmediği için o ana kadar belirlenmiş olan en iyi konum olmasına karşın bu çözüm terk edilmiş ve rasgele bir konumdan aramaya devam edilmiştir. Bu durumda takip eden yineleme adımlarında ondan sonra gelen bir sonraki en iyi konumun uygunluk değeri grafikte görülmektedir. Bu ikinci en iyi konumda da “Limit” defa arama yapıldığında iyileşme gerçekleşmeyince o da terk edilmiş ve rasgele bir konumdan aramaya devam edilmiştir. Kırkıncı yineleme (arama) adımından sonra bir miktar iyileşme sağlanmış olmakla birlikte ABC algoritması ulaşılabilecek en iyi değer altında kalmıştır. Buna karşılık MABC algoritmasının en yüksek uygunluk değerinin sürekli artan bir değer izlediği görülmektedir.

İyileştirilmiş ABC algoritmasındaki yeniliklerden bir diğeri de arama sürecinin farklı MCN-Limit çiftleri ile tekrarlanmasıdır. ABC algoritması deterministik yapıda olmadığı için, rasgele arama mekanizması nedeniyle her çalıştırıldığında farklı sonuç vermektedir. Farklı MCN-Limit çiftleri ile arama sürecinin tekrarlanması, daha iyi sonuçların bulunmasını olası kılmaktadır. Hangi MCN-Limit değerleri ile kaç defa arama yapılacağı yeni bir kontrol parametresi olarak MABC algoritmasına eklenmiştir.



Şekil 4.9 : MABC ve ABC algoritmalarının IRIS veri kümesi üzerinde çalışmaları esnasında en yüksek “Uygunluk Değeri” arama süreci esnasındaki değişimi.

4.2.3 İyileştirilmiş ABC algoritmasının işlem adımları

Sınıflama uygulamalarında iyileştirilmiş ABC algoritmasının kullanımında, daha iyi konum arama aşamasında, bir önceki adımda üzerinde çalışılan konum ile yeni hesaplanan konum arasında hangisinin daha iyi olduğuna karar vermek için “performans” değeri F^q yerine, “sınıflama başarımı” Acc^q kullanılmasının daha uygun olduğu görülmüştür.

İyileştirilmiş ABC algoritmasının adımları aşağıda görülmektedir:

İlk koşullama

- Veri kümesi $O = \{o_1, o_2, \dots, o_n\}$ 'yu kullan.
- Kontrol parametrelerinin (N , N_e , N_o , Limit, MCN) değerlerini seç.
- N_e adet arıyı rasgele seçilen örnek vektörlerinin konumlara yerleştir (rasgele konumlara değil).
- N_e adet başlangıç konumuna ait uygunluk değerlerini (Acc^q) hesapla.

döngü=1 (Çevrim’in içinde yinelemelerin başlangıcı)

tekrarla (MCN defa, tüm koloni için)

- Her işçi arı için gerçekleştir (**İşçi arı arama aşaması**)
 - Mevcut konumun yakınlarında yeni bir konum belirle, eşitlik (5.1).
 - Yeni konumun uygunluk değerini (Acc^q) hesapla
 - Bir önceki ve yeni konumun uygunluk değerlerini karşılaştır. Daha iyi olan konumu hafızada sakla.
 - Eğer son “Limit” defa arama işleminde daha iyi uygunluk değeri olan bir konum bulunamamışsa ve üzerinde çalışılan konumun uygunluk değeri, o ana kadar bulunmuş olan en iyi konumun uygunluk değerine yakın bir değerde değilse (*bu arının uygunluk değeri* $< SCTR \times$ *bu ana kadar bulunmuş en yüksek uygunluk değeri* ise) mevcut konumu terk et ve kâşif arıya dönüş.
- İşçi arıların konumlarına ait olasılık değerlerini (P^q) hesapla, eşitlik (4.4).
- Her gözcü arı için gerçekleştir (**Gözcü arı arama aşaması**)

- P^q olasılık değerlerine göre hangi işçi arının konumuna gideceğini belirle.
- O konumun yakınlarında yeni bir konum belirle, eşitlik (5.1).
- Yeni konumun uygunluk değerini hesapla (Acc^q).
- Bir önceki ve yeni konumun uygunluk değerlerini karşılaştır. Daha iyi olan konumu hafızada sakla.
- Bu adımda en az bir işçi arı kâşif arı olmuş ise; her kâşif arı için gerçekleştir **(Kâşif arı arama aşaması)**
 - Kaşif arıyı rasgele bir konuma yerleştir, eşitlik (4.1).
- Bu ana kadar belirlenen en iyi konumu hafızada sakla.

döngü = döngü + 1

Eğer döngü=MCN olmuş ise dur.

4.3 MABC Algoritmasının EKG Vuru Tiplerinin Sınıflandırılmasına Uygulanması

MABC algoritması ile EKG vuru tiplerinin sınıflandırılması için ilk olarak, üzerinde çalışılacak örnek vektörlerinden oluşan veriler, eğitim ve test kümeleri olmak üzere iki parçaya bölünür. Kullanılacak kontrol parametrelerine uygun değerler verilerek, eğitim kümesinde yer alan örnek vektörleri üzerinde algoritma çalıştırılır. Eğitim vektörlerinden hareketle en iyi sınıflama sonuçlarını verecek olan öbek merkez koordinatlarının bulunması işlemi, rasgele arama süreci ile gerçekleştirildiğinden; algoritma her çalıştırıldığında farklı sınıflama başarımları elde edilebilmektedir. Bu nedenle eğitim aşamasında algoritma birden fazla çalıştırılıp, en yüksek sınıflama başarımlarını veren öbek merkez koordinatları kaydedilmektedir. Eğitim kümesindeki örnek vektörleri, hangi öbek merkezine en yakın ise o öbeğe ait olarak atanır. Her bir öbekte, hangi sınıfa ait eğitim örnek vektörü en fazla sayıda bulunuyorsa, o öbek o sınıfa ait olarak etiketlenir.

Test aşamasında, test kümesinde yer alan örnek vektörlerinin her birinin, Öklid uzaklığı olarak bu öbeklerden hangisine en yakın olduğuna bakılır. Hangi öbek merkezine en yakın mesafede ise, o öbeğin ait olduğu sınıfa ait olarak etiketlenir.

Eğitim aşamasında her sınıfa ait öbek merkezleri belirlendikten sonra, test aşamasında çok yüksek sayıda test örnekleri oldukça kısa bir süre içinde sınıflanabilmektedir.

Sınıflama başarımının, kullanılan eğitim vektörlerinden bağımsız olduğunu göstermek için, yaygın olarak kullanılan bir yöntem K-katlı çapraz doğrulama (K-fold cross validation) tekniğidir. Bu yönteme göre, mevcut örnek vektörleri rasgele seçilerek K-eşit parçaya ayrılır. Örneğin K=5 kullanılması durumunda F1, F2, F3, F4 ve F5 adlı alt kümeler elde edilir. Bu parçalardan, aşağıda görüldüğü şekilde beş farklı eğitim ve test kümesi setleri oluşturulur.

$$E1 = F1 + F2 + F3 + F4 \quad T1 = F5$$

$$E2 = F1 + F2 + F3 + F5 \quad T2 = F4$$

$$E3 = F1 + F2 + F4 + F5 \quad T3 = F3$$

$$E4 = F1 + F3 + F4 + F5 \quad T4 = F2$$

$$E5 = F2 + F3 + F4 + F5 \quad T5 = F1$$

Ex kümesi ile eğitilen algoritma, Tx kümesi ile test edilir. Beş denemenin sonunda, her bir set üzerinde elde edilen sınıflama başarımlarının birbirine yakın olması, incelenen sınıflama yönteminin performansının, eğitim kümesindeki örnek vektörlerinin seçiminden bağımsız olduğunu gösterir (Pasolli ve Melgani, 2015). Bu tez çalışmasında gerçekleştirilen K-katlı çapraz doğrulama işlemi ve sonuçları Bölüm 5.5’de verilmiştir.



5. BİLGİSAYAR BENZETİM SONUÇLARI

Tez çalışması kapsamında kullanılan elektrokardiyogram verileri, bunlardan hareketle oluşturulan örnek vektörleri, zaman domeni ve frekans domeni öznitelikleri kullanılarak gerçekleştirilen sınıflama işlemlerine ait bilgisayar benzetim sonuçları aşağıda verilmiştir.

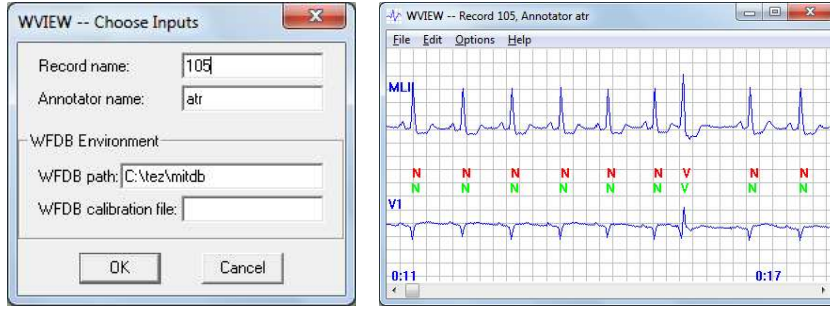
5.1 MIT-BIH Veri Tabanı

Bu tez çalışmasında incelenmiş olan elektrokardiyogram verileri, MIT-BIH aritmi veri tabanından alınmıştır. Beth Israel Hastanesi aritmi laboratuvarında yapılmış olan EKG kayıtları, Massachusetts Institute of Technology bünyesinde yapılan bir çalışma ile sayısallaştırılarak veri tabanı şeklinde hazırlanmıştır. Physionet web sitesi altında (Url-6) tüm araştırmacıların kullanımına sunulmuştur. Veri tabanı, yaklaşık 30 dakikalık 48 kayıttan oluşmaktadır. Değişik yaş gruplarından, çeşitli sağlık durumlarında 25 erkek ve 22 kadın olmak üzere toplam 47 kişiden alınan EKG verilerini içermektedir. İki kayıta aynı kişiden alınan EKG verileri mevcuttur.

Her kayıt dosyasında iki ayrı derivasyon bulunmakta, bunlardan biri II. derivasyon iken, diğeri V1, V2 veya V5 göğüs derivasyonudur. Bu çalışmada II. Derivasyon kayıtları kullanılmıştır. Orijinal analog kayıtlar 9 adet Del Mar Avionics model 445 cihazı ile alınmıştır. Farklı cihazlar kullanılmış olmasının kayıtlar üzerindeki etkisini gidermek amacıyla, tez çalışmasında normalizasyon işlemleri uygulanmıştır.

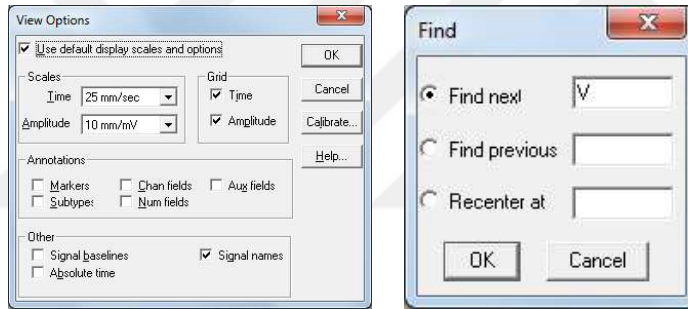
Sayısallaştırma sürecinde analog işaret 360 Hz ile örneklenmiştir. Kullanılan analog/sayısal dönüştürücüler ± 5 mV aralığında 11 bitlik çözünürlüğe sahiptir. Bütün EKG kayıtları uzman kardiyolog tarafından incelenmiş ve her bir kalp vurusu etiketlenmiştir. Kalp vurularının yanı sıra ritim türü etiketleri, işaret kalitesi etiketleri de eklenmiştir. MIT-BIH veri tabanındaki kayıtlarda yer alan elektrokardiyogram işaretinin değişimi, kardiyologların yaptığı vuru ve ritim etiketlemeleri “wview” adlı program kullanılarak görüntülenebilir (Şekil 5.1). Basit bir kullanıcı ara yüzüne sahip olan programda ilk önce, MIT-BIH veri tabanındaki kayıtlardan kaç numaralı

dosyadaki EKG verilerinin görüntülenmek istendiği seçilir. Açılan kayıt içinde ok tuşları yardımı ile zaman ekseninde istenilen bölge ekranda görüntülenir.



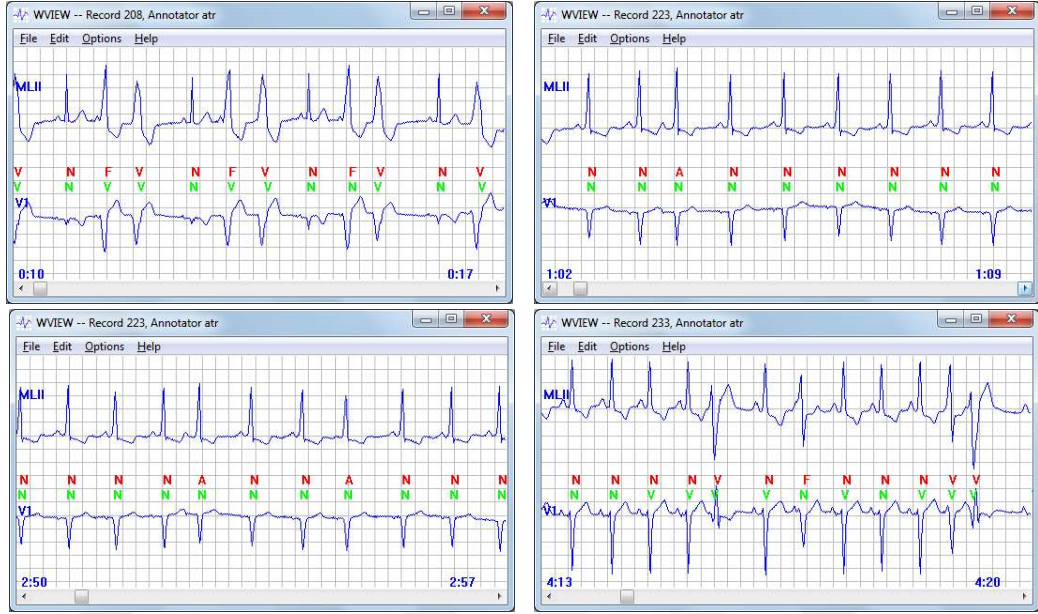
Şekil 5.1 : MIT-BIH veri tabanındaki EKG verilerinin wview programı ile görüntülenmesi.

Görüntülenen ekg verileri üzerinde, kardiyologların belirlemiş olduğu, kalp vurularına ait etiketlerde görülmektedir. Ayrıca, “wview” programında görüntüleme seçenekleri, zaman ve işaret genliği için görüntüleme skalaları, belirli bir vuru tipinin kayıt içinde aranması gibi özellikleri bulunmaktadır (Şekil 5.2).



Şekil 5.2 : “wview” programındaki ayar pencereleri.

MIT-BIH veri tabanındaki bazı etiketlerin hatalı olduğu zaman içerisinde anlaşılmış ve düzeltilmiştir. Örneğin, 1998’de 203 numaralı kayıta bir ritim etiketi düzeltilmiş, 2001’de 209 numaralı kayıta bulunan vuru etiketlerindeki hatalar belirlenmiş ve düzeltilmiş, 2003 yılında 119 numaralı kayıta 26 adet PVC tipi aritmik vuru etiketi düzenlenmiş, ayrıca 214 numaralı kayıta ise bir başka vuru etiketindeki hata düzeltilmiştir. İlerleyen senelerde bu düzeltmeler devam etmiş, 2010 yılında 203 numaralı kayıta vuru etiketlerinde düzeltmeler yapılmıştır (Url-7). Bununla birlikte, halen bazı etiketlemelerde hataların bulunduğu görülmektedir. Bunlardan bazı örnekler, 208, 223 ve 233 numaralı kayıtlarda bulunmakta olup, Şekil 5.3’te görülmektedir. Örneğin 208 numaralı kayıta birinci izde F tipi olarak etiketlenmiş olan vuruların, ikinci izde V tipi veya N tipi olarak etiketlendiği görülmektedir. Benzer şekilde, 233 numaralı kayıta birinci izde N tipi olarak işaretli olan bazı vurular ikinci izde V tipi olarak etiketlenmiştir.



Şekil 5.3 : MIT-BIH veri tabanındaki hatalı etiketleme örnekleri.

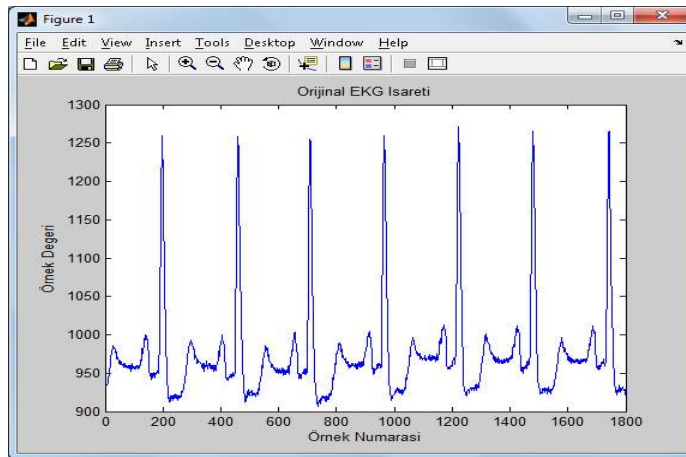
MIT-BIH veri tabanında yer alan kayıtlardan verilerin ve kalp vurularının etiketlerinin okunması için çeşitli yardımcı programlar physio toolkit adı altında, Physionet web sitesinde (Url-8) yer almaktadır. Bunlara örnek olarak “ecgpuwave, physio toolkit/rdann, rdsamp” gibi programlar verilebilir. MIT-BIH kayıtlarından veri okumak için hazırlanmış olan “rdsamp” programı (function) kullanılarak elde edilmiş veriler grafik olarak çizdirilip elektrokardiyogram işareti görsel olarak incelenebilir. “rdsamp” programı ile bir MIT-BIH kaydı içinden EKG verilerinin okunması örneği aşağıda görülmektedir. Hangi MIT-BIH kaydı içindeki, hangi zaman aralığından veri okunacağı, başlangıç ve bitiş anları birer parametre olarak verilmektedir.

```
RecordNo='100';
Baslangic_Zamani='00:00:00';
Bitis_Zamani='00:00:05';

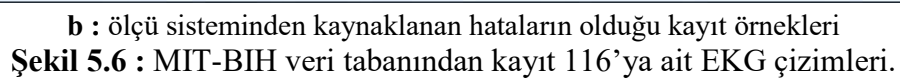
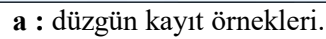
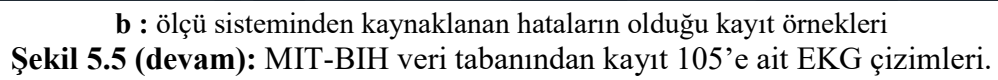
setwfdb('C:\MATLAB7\work\')
data10 = rdsamp(RecordNo, 'begin', Baslangic_Zamani, 'stop', Bitis_Zamani);
x=data10(:,2);

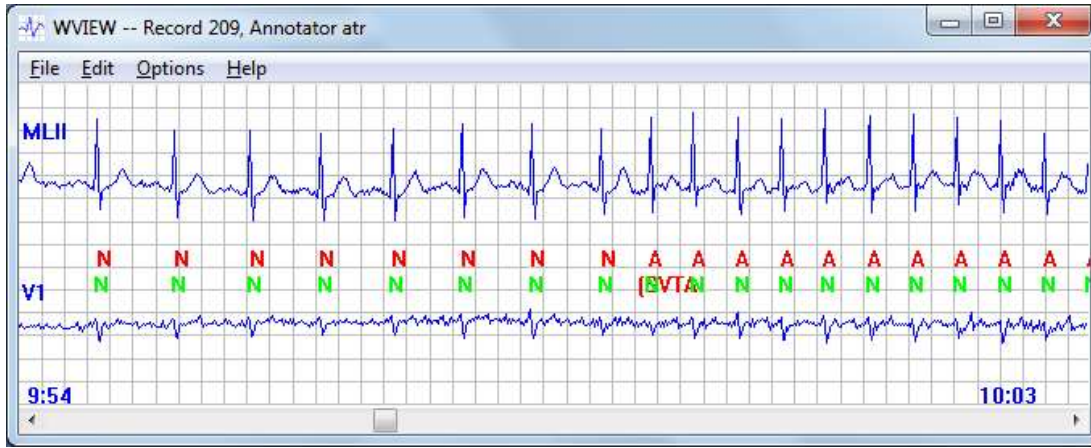
figure, plot(x);title('Orijinal EKG Isareti')
xlabel({'Örnek Numarasi'}); ylabel({'Örnek Degeri'});
```

Bu program ile okunmuş olan, 100 numaralı MIT-BIH kaydının ilk 5 saniyelik kısmının görüntüsü Şekil 5.4’de verilmiştir. Her saniyede 360 örnek bulunduğundan 5 saniyelik bölümde toplam 1800 örnek yer almaktadır.

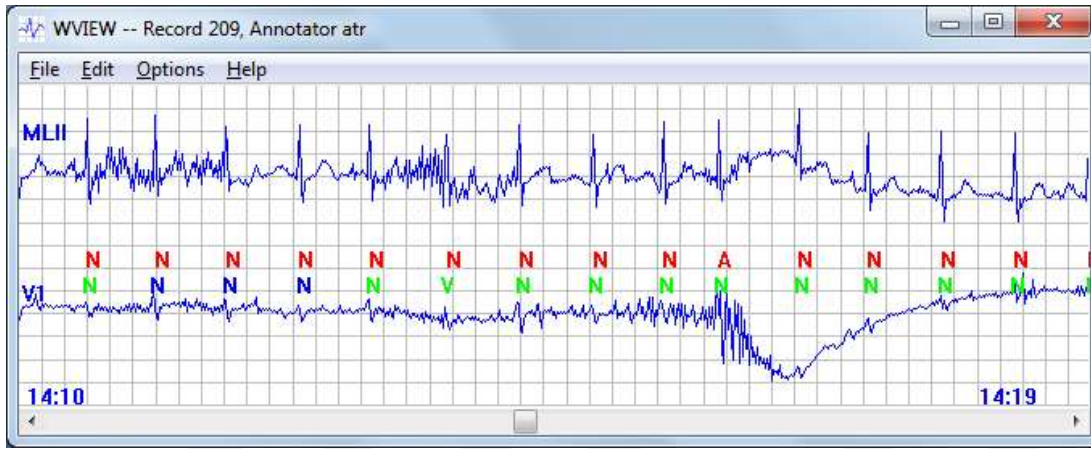


Dikkat edilmesi gereken noktalardan biri, normal elektriksel gürültülerin yanı sıra kayıt sisteminden kaynaklanan hataların olumsuz etkisinin devre dışı bırakılmasıdır. Bu tür kayıt hatalarının elektrokardiyogram işaretine etkisinin görülmesi amacıyla, bazı örnekler aşağıda verilmiştir. MIT-BIH veri tabanındaki kayıtlardan, kayıt 105, kayıt 116 ve kayıt 209'daki düzgün ve hatalı kayıtların bulunduğu bölgelere ait EKG çizimleri Şekil 5.5, Şekil 5.6 ve Şekil 5.7'de görülmektedir. Bu hatalı kayıt bölgelerine ait verilerin, sınıflama işlemlerinde kullanılması halinde sınıflayıcı performansının olumsuz etkileneceği açıktır. Bu nedenle bu tür hatalı kayıt bölgelerinin ayıklanmasından sonra düzgün işaret örnekleri üzerinde çalışma yapılmıştır.





a : düzgün kayıt örnekleri.



b : ölçü sisteminden kaynaklanan hataların olduğu kayıt örnekleri

Şekil 5.7 : MIT-BIH veri tabanından kayıt 209'a ait EKG çizimleri.

Yukardaki şekillerdeki gibi hatalı kayıtların bulunduğu bölgeler inceleme kapsamı dışında bırakılmıştır.

Bu tez çalışmasında MIT-BIH veri tabanında yer alan normal vurular (N), jonksiyonel kaçış vuruları (j), erken karıncık kasılmaları (V), karıncık ve normal vuru füzyonları (F), paced ve normal vuru füzyonları (f), kulakçık erken vuruları (A), anormal kulakçık erken vuruları (a) ve sağ dal bloğu vuruları (R) olmak üzere 8 tip vuru incelenmiştir. Farklı vuru tiplerine ait örnekler, mümkün olduğunca değişik kayıtlardan alınmış olup, hastadan bağımsız olmasına özen gösterilmiştir.

Kullanılan EKG verileri, MIT-BIH veri tabanındaki 100, 105, 116, 119, 201, 202, 205, 208, 209, 210, 213, 217, 222, 223, 231 ve 233 numaralı kayıtlardan alınmıştır. Kullanılan MIT-BIH kayıtlarındaki vuru tipleri ve sayıları, Çizelge 5.1'de görülmektedir.

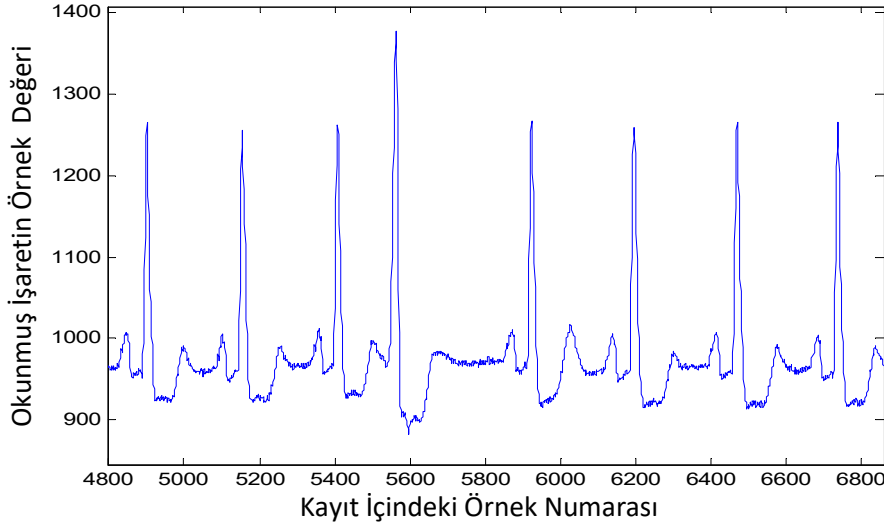
Çizelge 5.1 : Kullanılan MIT-BIH kayıtlarındaki vuru tipleri ve örnek sayıları

Kayıt No:	N	J	V	F	f	A	a	R
100	2239	-	1	-	-	33	-	-
105	2526	-	41	-	-	-	-	-
116	2302	-	109	-	-	1	-	-
119	1543	-	444	-	-	-	-	-
201	1625	10	198	2	-	30	97	-
202	2061	-	19	1	-	36	19	-
205	2571	-	71	11	-	3	-	-
208	1586	-	992	373	-	-	-	-
209	2621	-	1	-	-	383	-	-
210	2423	-	194	10	-	-	22	-
213	2641	-	220	362	-	25	3	-
217	244	-	162	-	260	-	-	-
222	2062	212	-	-	-	208	-	-
223	2029	-	473	14	-	72	1	-
231	314	-	2	-	-	1	-	1254
233	2230	-	831	11	-	7	-	-

Farklı vuru tiplerine ait örnekler, mümkün olduğunca değişik kayıtlardan alınmış olup, hastadan bağımsız olmasına özen gösterilmiştir. Bazı vuru tiplerine ait örnekler (f ve R tipi vurular) az sayıda kayıt içinde bulunmaktadır. Bunların da çoğunluğu normal vuru içermeyen, “pacemaker” tipi vuruların bulunduğu kayıtlardır.

5.2 EKG Verilerine Ait Zaman ve Frekans Domeni Öznitelikleri

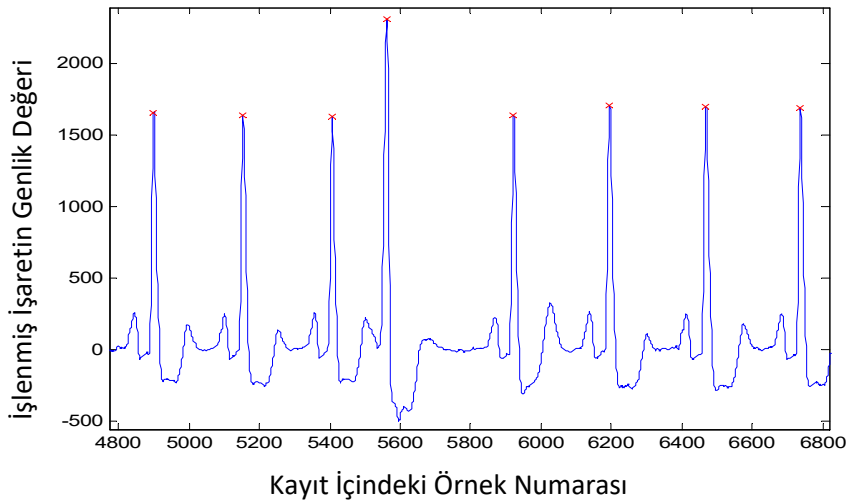
EKG işaretindeki vurulara ait özniteliklerin belirlenmesi amacıyla gerçekleştirilen işaret işleme adımları aşağıda açıklanmıştır. Normal vuruların yanında bir V tipi vuru içeren, MIT-BIH veri tabanı, 105 numaralı kayıttan alınmış olan EKG işaretinin bir bölümü Şekil 5.8’de görülmektedir.



Şekil 5.8 : MIT-BIH veri tabanı, 105 numaralı kayıtdan alınmış olan EKG işareti.

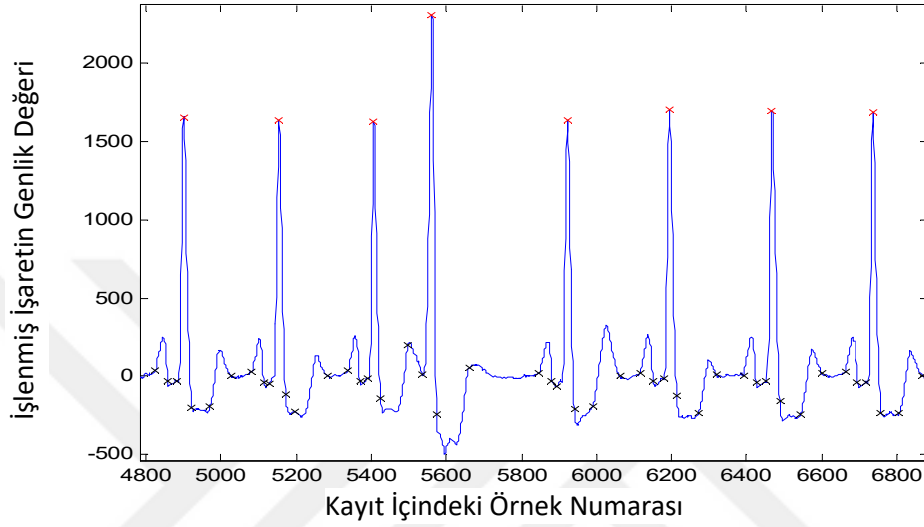
Analog EKG işareti 11 bit çözünürlük ile sayısallaştırılmış olduğu için, sayısal örnekler 0-2047 değerlerini alabilmektedir. Kayıtlarda bulunan örneklenmiş sayısal verilerin sıra numarasından, kayıt başlangıcından itibaren kaçınıcı dakika ve saniyede bulunduğu belirlenebilmektedir. Şekil 5.8’de 105 numaralı kaydın başından itibaren 13. saniye ile 19. saniye arası yer almaktadır.

Bu örnekte görüldüğü gibi, işaretin üzerinde çeşitli elektriksel gürültüler bulunmaktadır. Pan-Tompkins algoritmasının gerçekleştirilmiş olduğu bir yazılım kullanılarak, işaretin üzerindeki 60 Hz (ABD şehir şebekesi frekansı) gürültüsü, taban hattı kayması vb. gürültülerin temizlenmesi ve R-tepelerinin yerlerinin belirlenmesi mümkündür. Bu şekilde gürültülerden temizlenmiş ve R-tepeleri belirlenmiş bir EKG işareti Şekil 5.9’da görülmektedir.



Şekil 5.9 : Gürültülerden temizlenmiş, R-tepeleri belirlenmiş EKG işareti.

Şekil 5.9’da görüldüğü gibi uygulanan işaret işleme yöntemleri sonucu elektriksel gürültüler büyük oranda azaltılmış, işaret kuvvetlendirilmiş, izoelektrik seviye “0” değerine çekilmiştir. Ayrıca Physionet kaynaklı “ecgpuwave” yazılımı kullanılarak P, QRS ve T dalgalarının başlangıç ve bitiş noktalarının EKG işaretinin içindeki konumları belirlenebilmektedir. Bu işlemler sonrası elde edilen, önemli noktaları belirlenmiş EKG işareti Şekil 5.10’da görülmektedir.



Şekil 5.10 : Önemli noktaları belirlenmiş EKG işareti.

Şekil 5.10’da görüldüğü gibi, P, QRS ve T dalgalarının başlangıç ve bitiş noktalarının EKG işaretinin içindeki yerleri belirlenmiştir. Bu ön-işleme adımlarından sonra, kullanılacak olan özneliklerin belirlenmesi gerçekleştirilmiştir.

Bu tez çalışmasında, birinci aşamada sadece zaman domeni öznelikleri kullanılarak, ikinci aşamada ise hem zaman hem de frekans domeni öznelikleri kullanılarak sınıflama işlemleri yapılmıştır. Kullanılan elektrokardiyogram verilerinden hareketle elde edilen öznelikler ve bunların indirgenmesi ile elde edilen optimum öznelik kümesine ait bilgiler aşağıda verilmiştir.

Bölüm 3.2.1’de açıklanmış olduğu şekilde, elektrokardiyogram işaretlerinin analizinde kullanılan zaman domeni öznelikleri, kalp vurusundaki başlıca bölümler olan P, QRS, T dalgalarının genlikleri, genişlikleri, alanları, birbirleri arasındaki süreler gibi değerlerden ve bu dalgaların mevcudiyeti ya da yokluğu, normal ya da ters genlikli olması gibi durumlardan oluşur. Bu tez çalışması kapsamında kullanılan özneliklerin belirlenmesi ve vurulara ait öznelik değerlerinin hesaplanması için aşağıda açıklanan yöntem izlenmiştir.

MIT-BIH veri tabanı dünya genelindeki tüm araştırmacıların serbest kullanımına sunulmuş olmakla birlikte, geliştirilen EKG analiz yöntemlerinin, kalp vurusu sınıflandırma algoritmalarının üzerinde çalıştırılarak test edileceği ortak veri kümesi (öznitelik vektörleri) bulunmamaktadır. Literatürde yer alan ve MIT-BIH veritabanını kullanan yöntemlerin, farklı veri kümeleri üzerinde çalıştırılmış olduğu görülmektedir. Yapılan ön araştırmada, bu tür çalışmalarda ortak olarak kullanılan bir öznitelik vektörleri veri kümesi mevcut olmadığı görülmüştür. Bu nedenle ilk aşamada, geliştirilen sınıflama yöntemi ile diğer çeşitli sınıflama yöntemlerinin üzerinde çalıştırılacağı, elde edilen sonuçların karşılaştırmasının yapılacağı öznitelik vektörleri kümesi hazırlanmıştır.

Bu amaçla öncelikle, Pan-Tompkins algoritmasının gerçekleştirilmiş olduğu bir yazılım kullanılarak, MIT-BIH veri tabanındaki dosyalardan alınan örneklenmiş ham EKG verileri gürültülerden arındırılmış ve R-tepelerinin tam konumları belirlenmiştir. Bundan sonra, Physionet web sitesinde yer alan ve araştırma amaçlı çalışmalarda serbest olarak kullanılabilen “ecgpuwave, physiotoolkit/rdann, rdsamp” yazılımları indirilmiş; bu yazılımlar kullanılarak, P, QRS ve T dalgalarının başlangıç, tepe ve bitiş noktalarının zaman eksenini üzerindeki tam konumları belirlenmiştir. Her iki yazılımla elde edilen R tepelerinin konumlarının aynı olduğu kontrol edildikten sonra, dalgalara ait genlik, alan, mutlak alan gibi özniteliklerin belirlenmesinde gürültülerden arındırılmış örneklenmiş veriler kullanılmıştır. Benzer şekilde, dalgaların genişliği, dalgalar arası süre gibi özniteliklerin belirlenmesinde de zaman eksenini içindeki konumlarının bilgileri kullanılmıştır.

Zaman domeni özniteliklerinin sayısal değerlerinin hesaplanması amacıyla, tez çalışması kapsamında bir yazılım geliştirilmiştir. Bu yazılım kullanılarak her bir vuruya ait, Çizelge 5.2’de listelenmiş olan 38 farklı zaman domeni özniteliği hesaplanmıştır.

Çizelge 5.2 : Tez çalışmasında incelenmiş olan zaman domeni özniteliklerinin listesi

1	P-dalgası mevcudiyeti (1 ya da 0)
2	P-dalgası yüksekliği
3	P-dalgası genişliği
4	P-dalgası alanı
5	P-dalgasının Physionet sayısal sınıflaması (normal, ters, çift-fazlı)

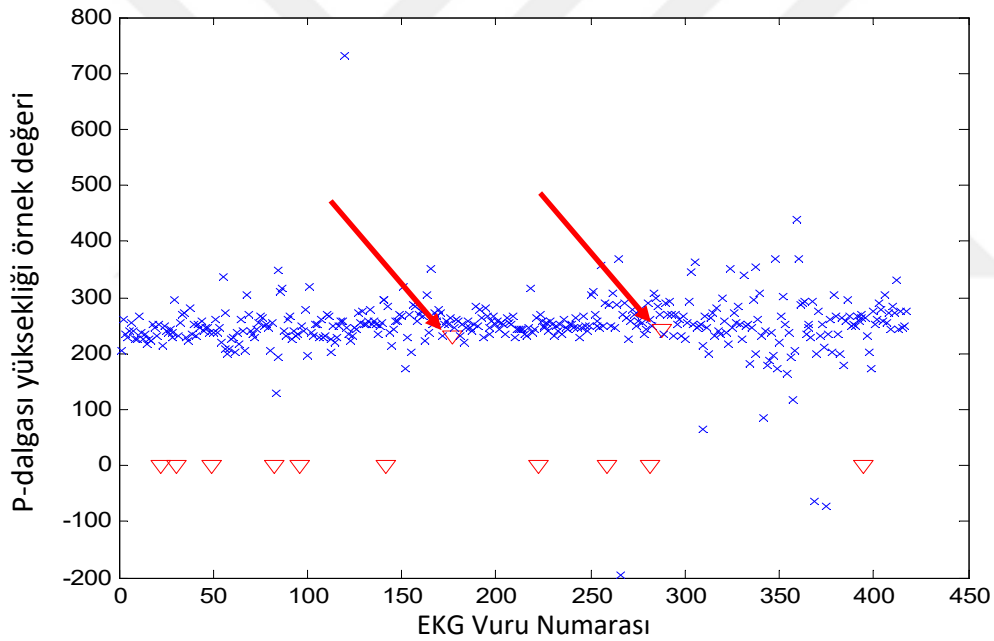
Çizelge 5.2 (Devam): Tez çalışmasında kullanılmış olan zaman domeni öz niteliklerinin listesi

6	QRS kompleksi R-dalgası yüksekliği
7	QRS kompleksi genişliği
8	QRS kompleksi alanı
9	QRS kompleksi mutlak alanı
10	QRS kompleksi pozitif bölge genişliği
11	QRS kompleksi pozitif bölge alanı
12	QRS kompleksi negatif bölge alanı
13	Q-dalgası alanı
14	S-dalgası alanı
15	Q-dalgası yüksekliği
16	S-dalgası yüksekliği
17	QRS kompleksi Physionet sayısal sınıflaması (normal, ters, çift-fazlı)
18	T1-dalgası mevcudiyeti (1 ya da 0)
19	T2-dalgası mevcudiyeti (1 ya da 0)
20	T-dalgası (T1) yüksekliği
21	T-dalgası genişliği (T1+T2 eğer varsa)
22	T-dalgası alanı
23	T-dalgası mutlak alanı
24	T-dalgası pozitif bölge alanı
25	T-dalgası negatif bölge alanı
26	T-dalgası Physionet sayısal sınıflaması (normal, ters, çift-fazlı)
27	T1 Physionet sayısal sınıflaması (normal, ters, çift-fazlı)
28	T2 Physionet sayısal sınıflaması (normal, ters, çift-fazlı)
29	İki adet R tepesi arasındaki minimum değer
30	T2-dalgası yüksekliği
31	PQ1 aralığı (P-dalgası başından, QRS başına kadar olan süre)
32	QT aralığı (QRS kompleksi başından, T-dalgası başına kadar)
33	ST aralığı (QRS kompleksi sonundan, T-dalgası sonuna kadar)
34	PQ2 aralığı (P-dalgasının sonundan, QRS-kompleksinin başına kadar olan süre)
35	PT aralığı (P-dalgasının başından, T-dalgası başına kadar)
36	RR(t)-RR(t-1) aralığı (bu vuru ile önceki vuru arasındaki süre)
37	RR(t+1)-RR(t) aralığı (bu vuru ile sonraki vuru arasındaki süre)
38	RR(t+1)-RR(t-1) aralığı (önceki ve sonraki vuru arasındaki süre)

Vurulara ait bu öznitelik değerleri, o vurunun hangi MIT-BIH kaydında bulunduğu, o kaydın kaçınıcı dakika-saniyesinde yer aldığı ve hangi tip vuru olduğu bilgileri de eklenerek bir veri dosyasına standart bir veri kalıbında kaydedilmiştir. Böylece üzerinde çalışılan örnek vektörleri için geriye dönük izlenebilirlik sağlanmıştır.

Kullanılan işaret işleme yazılımlarından biri olan ecgpuwave çok yararlı bir araç olmakla birlikte, zaman zaman da olsa tanımlamada hata yapabilmekte olduğu görülmüştür. Örneğin V tipi (PVC) aritmik vurularda P-dalgası bulunmamaktadır. Oysaki yapılan incelemelerde bazı PVC tipi vurularda P-dalgası mevcut olarak işaretlenmiştir.

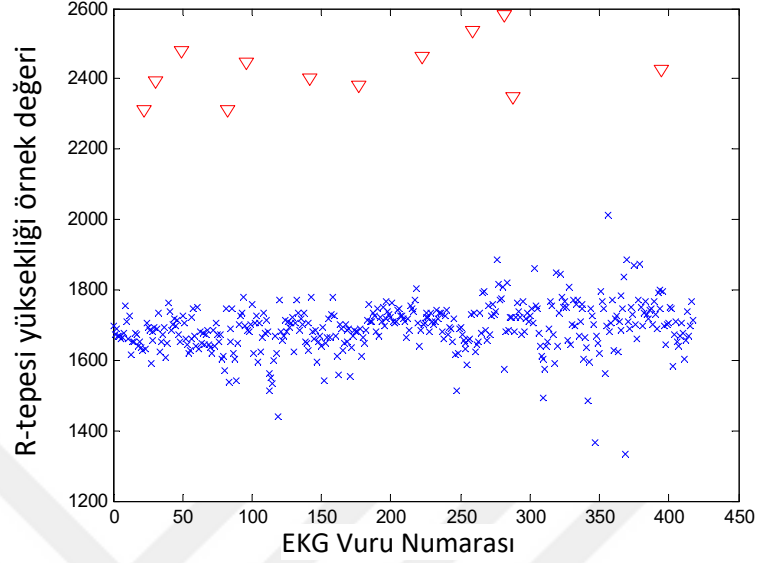
Tez çalışması kapsamında incelenen zaman domeni özniteliklerinden biri P-dalgası olup, 105 numaralı MIT-BIH kaydının ilk 5 dakikalık bölümündeki vurulara ait P-dalgası yükseklikleri 'de görülmektedir.



Şekil 5.11 : 105 numaralı MIT-BIH kaydındaki vurulara ait P-dalgası yükseklikleri.

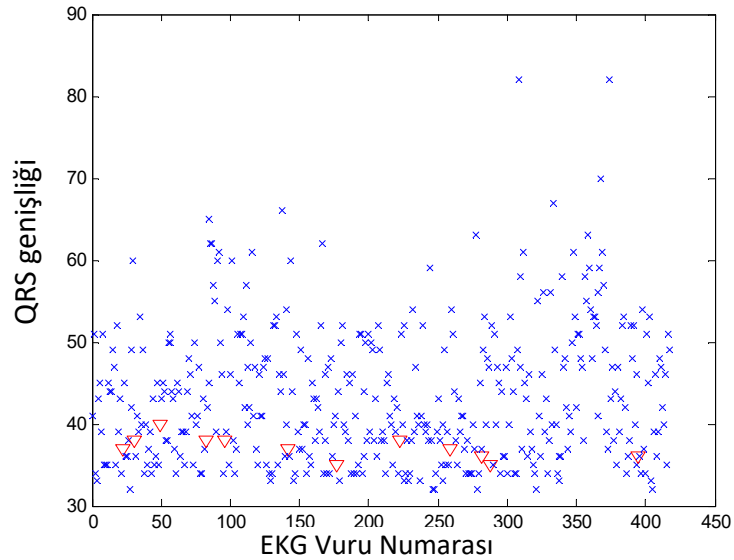
Şekil 5.11’de normal vurulara ait P-dalgası yükseklikleri mavi “x” sembolü ile, V tipi aritmik vurulara ait P-dalgası yükseklikleri ise kırmızı “V” sembolü ile gösterilmiştir. Bu örnekte iki adet V tipi (PVC) aritmik vuruda bir önceki normal vurunun T-dalgası, V tipi vurunun P-dalgası şeklinde hatalı olarak tanımlanmıştır. Bu durum 105 numaralı kayıta ilgili örnek numaralarından hareketle kontrol edilerek tespit edilmiştir.

İncelenen bir diğer zaman domeni özneliği, R-tepesi yüksekliğidir. 105 numaralı kaydın ilk 5 dakikalık bölümündeki vurulara ait R-tepesi yükseklikleri Şekil 5.12’de görülmektedir.



Şekil 5.12 : 105 numaralı MIT-BIH kaydındaki vurulara ait R-tepesi yükseklikleri.

Benzer şekilde bir başka zaman domeni özneliği olarak, QRS dalgası genişlikleri incelenmiştir. 105 numaralı kaydın ilk 5 dakikalık bölümündeki vurulara ait QRS dalgası genişlikleri Şekil 5.13’de görülmektedir.



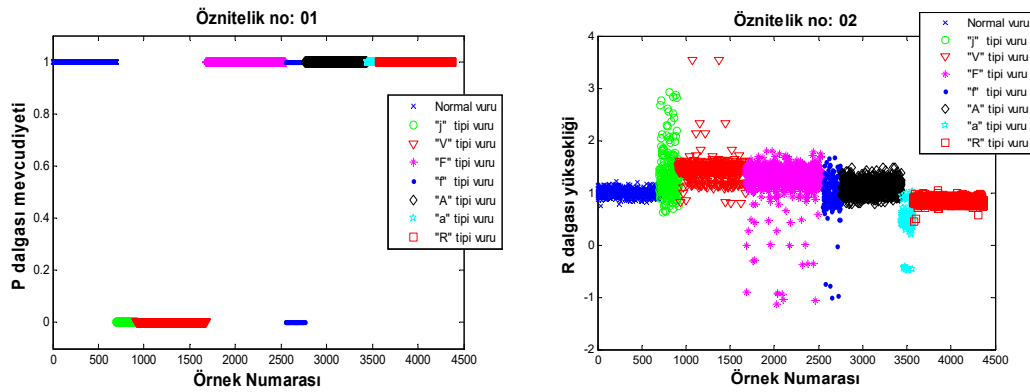
Şekil 5.13 : 105 numaralı MIT-BIH kaydındaki vurulara ait QRS genişlikleri.

Şekil 5.12’ de görüldüğü gibi, R-tepesi yüksekliği normal ve V-tipi aritmik vurular arasında çok belirgin bir ayırt edici özneliktir. Fakat Şekil 5.13 incelendiğinde bu kayıta yer alan vurular için, QRS-genişliği bilgisinin normal ve V-tipi vuruların ayırt

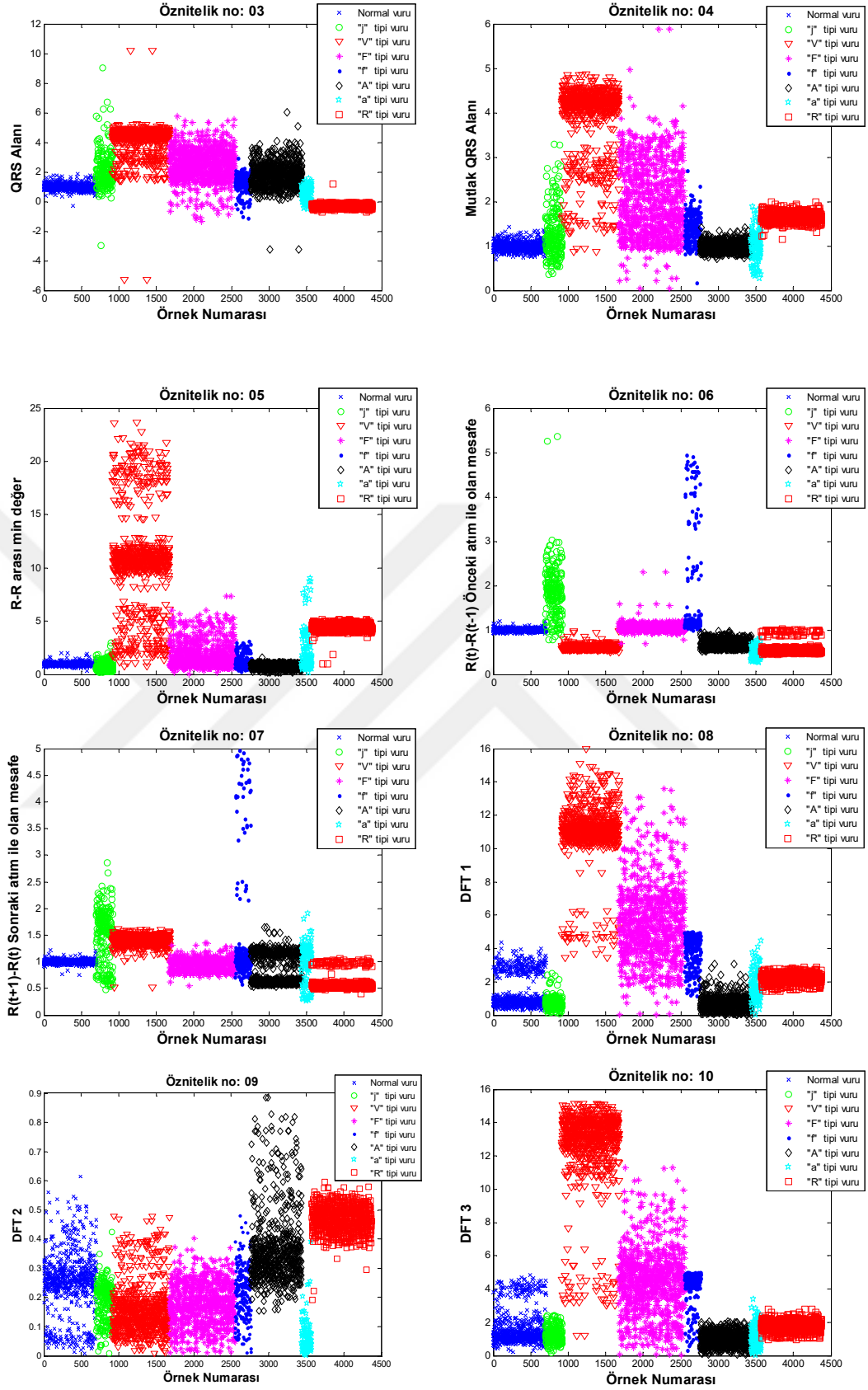
edilmesinde kullanılamayacağı görülmektedir. Bu örneklerde incelenen özniteliklerin ayırt edicilik durumu görsel olarak belirlenebilmektedir. Fakat kullanılan öznitelik sayısı ve incelenen sınıf sayısı arttıkça bu tür bir görsel incelemenin çok zorlaşacağı ve hatta mümkün olmayacağı açıktır. Bunun yerine özniteliklerin ayırt edicilik seviyelerini sayısal olarak hesaplayacak bir yöntemin kullanılması uygun olacaktır. Bu amaçla Bölüm 3.2.3'te açıklanmış olan diverjans analizi kullanılarak, yukarıda belirtilen 38 zaman domeni özniteliği içinden, üzerinde çalışılan veriler için en yüksek ayırt ediciliğe sahip olan 7 öznitelik belirlenmiştir. Bu özniteliklerin listesi aşağıda görülmektedir.

- Mutlak QRS alanı,
- R-R tepeleri arasındaki minimum değer (çoğu durumda s-dalgası genliği),
- $R(t)$ ve $R(t-1)$ arasındaki zaman aralığı (o vuru ve önceki vuru arası),
- $R(t)$ ve $R(t+1)$ arasındaki zaman aralığı (o vuru ve sonraki vuru arası),
- R-dalgası genliği,
- QRS kompleksi alanı,
- P-dalgası mevcudiyeti.

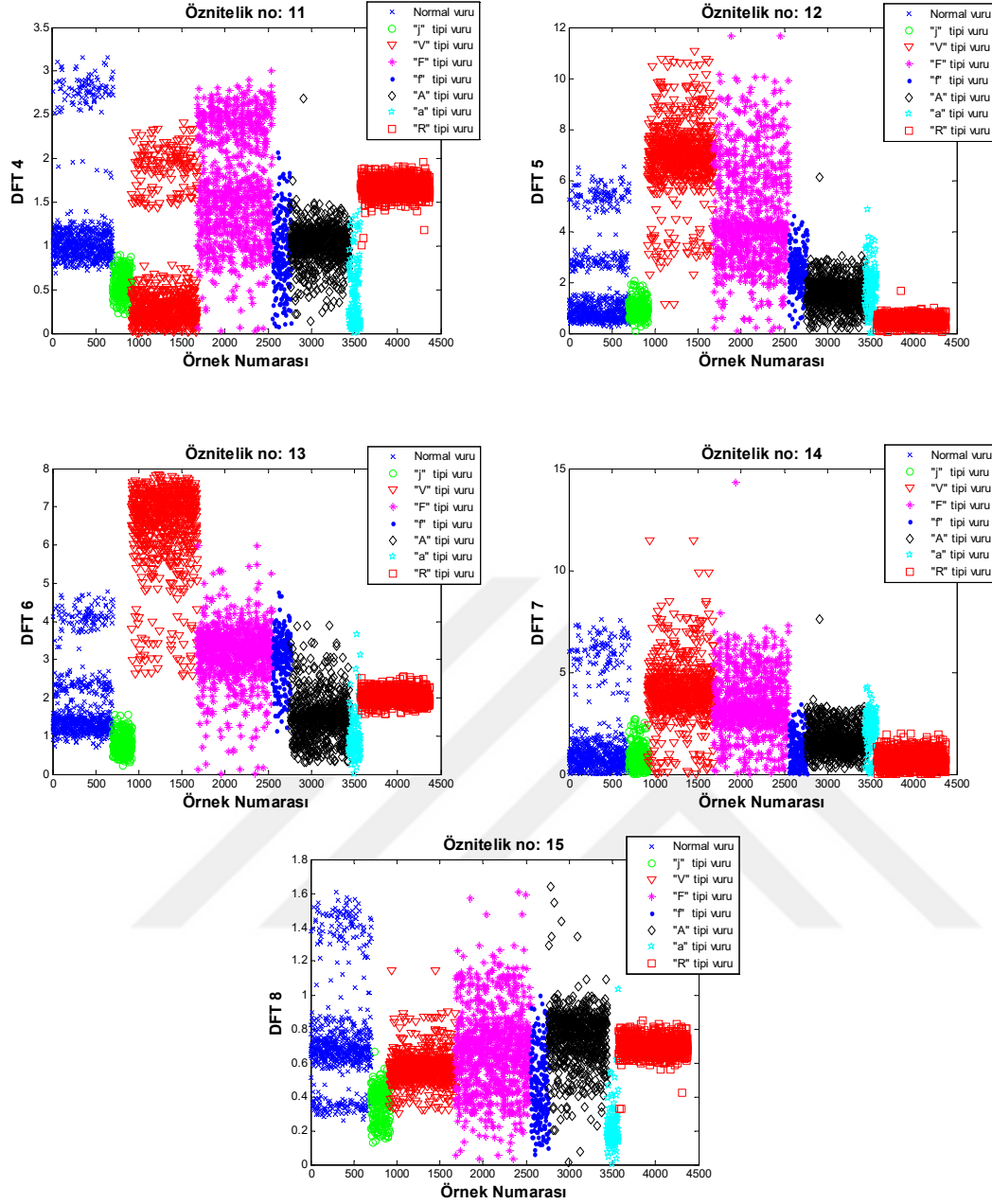
Aşağıda detayları verilen VK_KFold kümesindeki normalize edilmiş örnek vektörlerine ait öznitelik değerlerinin dağılımı Şekil 5.14'de görülmektedir.



Şekil 5.14: VK_KFold kümesindeki normalize edilmiş örnek vektörlerine ait öznitelik değerlerinin dağılımı.



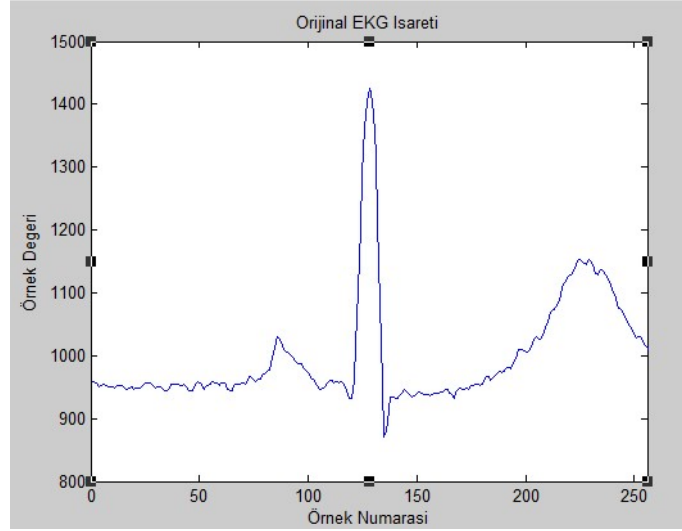
Şekil 5.14 (devam): VK_KFold kümesindeki normalize edilmiş örnek vektörlerine ait öznitelik değerlerinin dağılımı.



Şekil 5.14 (devam): VK_KFold kümesindeki normalize edilmiş örnek vektörlerine ait öznitelik değerlerinin dağılımı.

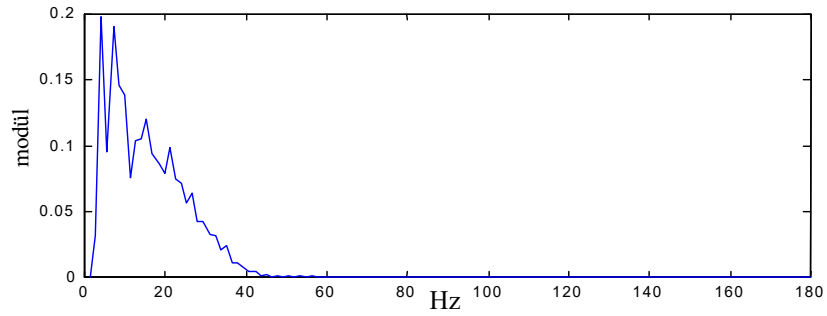
Zaman domeni özniteliklerine ek olarak kullanılan frekans domeni özniteliklerinin hesaplanmasında, Bölüm 3.2.2’de açıklanmış olan Ayrık Fourier Dönüşümü (AFD) kullanılmıştır.

Bu amaçla öncelikle, incelenecek her bir vuruya ait R-tepesinin bulunduğu noktadan önceki 127 ve takip eden 128 noktanın bulunduğu toplam 256 örneklik pencereler alınmıştır. Normal bir kalp vurusuna ait örneklenmiş veriler Şekil 5.15’de görülmektedir.



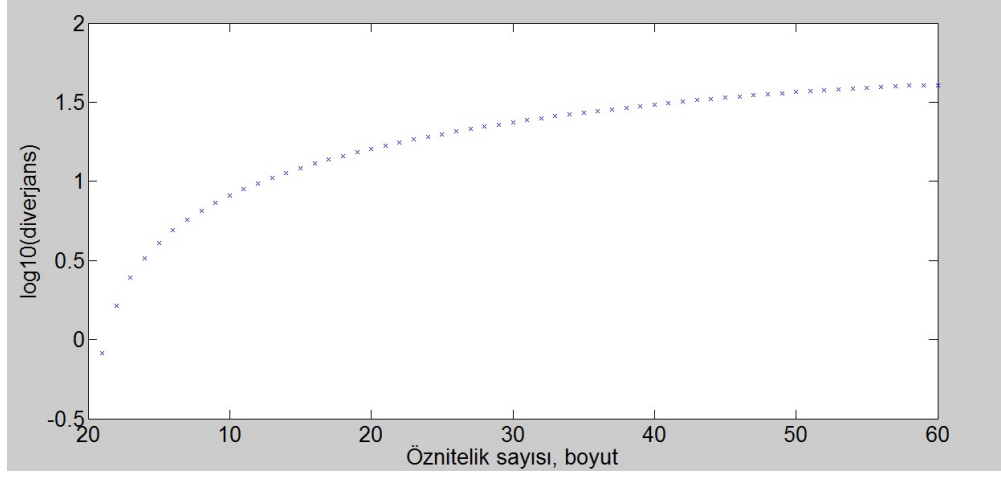
Şekil 5.15 : Normal bir kalp vurusuna ait örneklenmiş veriler.

Bu pencerelerdeki 256 örnekten oluşan veri kümelerine AFD uygulanarak, o kalp vurusuna ait ayrık Fourier katsayıları elde edilmiştir. Bu dönüşüm sonucu elde edilen, normal bir kalp vurusuna ait ayrık Fourier katsayıları Şekil 5.16’da görülmektedir.



Şekil 5.16 : Normal bir kalp vurusuna ait ayrık Fourier katsayıları.

Şekil 5.16’da görüldüğü gibi, ilk 60 katsayıdan sonraki elemanlar yaklaşık olarak sıfır değeri aldığı için dikkate alınmazlar. Bu ilk 60 katsayı içinde en yüksek ayırt ediciliğe sahip öznitelikleri belirlemek için diverjans analizi yapılmıştır. Diverjansın öznitelik sayısı (boyut) ile değişimi Şekil 5.17’de görülmektedir.



Şekil 5.17 : Diverjansın öznitelik sayısı (boyut) ile değişimi.

Hesaplanan diverjans değerlerine göre frekans domeninde en yüksek ayırt ediciliğe sahip 8 öznitelik kullanılmıştır. Bu 8 öznitelik numaraları Çizelge 5.3'te görülmektedir.

Çizelge 5.3 : En yüksek ayırt ediciliğe sahip ilk 8 öznitelik.

Sıra no:	1	2	3	4	5	6	7	8
Öznitelik no:	3	23	2	10	4	5	1	16

Zaman domeni ve frekans domeni öznitelikleri birlikte kullanılarak sınıflama başarımının artırılması hedeflenmiştir. Sınıflama algoritmasında kullanılan öznitelik sayısını arttırmak, diverjans değerini büyütecek, dolayısıyla daha yüksek sınıflama başarımı elde edilebilecektir. Ancak, artan vektör boyutu hesaplama yükünü de önemli oranda arttırmaktadır. Bu nedenle en yüksek ayırt ediciliğe sahip 7 zaman domeni ve 8 frekans domeni olmak üzere, 15 boyutlu öznitelik vektörleri ile çalışılmıştır.

Literatürde yer alan EKG vuru sınıflama algoritmalarına yönelik çeşitli yayınlarda, eğitim kümesi ve test kümesinde birbirine yakın sayıda örnekler kullanıldığı görülmektedir. Örneğin Dokur ve Ölmez (2001) toplam 3000 EKG vurusu üzerinde çalışmış, bunların 1500 adedini eğitim, 1500 adedini ise test amaçlı kullanmıştır. Benzer şekilde Osowski ve diğ. (2008), 6568 vuruyu incelemiş, bunların 3500'ünü eğitim, 3068 adedini test amaçlı kullanmıştır. Homaenizhad ve diğ. (2012), 7185 vuru üzerinde çalışmış, 4035 vuruyu eğitim kümesinde, 3150 vuruyu ise, test kümesinde kullanmıştır.

Bu yaklaşımlar örnek alınarak, eğitim ve test kümeleri oluşturulurken farklı kayıtlardan alınan örnek vektörleri rasgele seçilmiş, aritmik vuruların yarısı eğitim amaçlı, yarısı test kümesinde kullanılmıştır. Normal kalp vurularından çok sayıda örnek bulunduğu için, bunlardan 700 tanesinin eğitim kümesinde kullanılması yeterli görülmüştür. Toplam 8 farklı vuru tipine ait 11269 vuru örneklerinden oluşan ve yalnız zaman domeni özniteliklerini içeren veri kümesine VK_zaman adı verilmiştir. Aynı vuru örneklerine ait hem zaman ve hem de frekans domeni öznitelik değerlerini içeren veri kümesi ise VK veri kümesi olarak adlandırılmıştır. Bu iki veri kümesinde bulunan örnek vektörlerinin sayıları aynı, fakat vektör boyutları VK_zaman'da 8, VK'da ise 15'dir.

VK ve VK_zaman veri kümeleri ile eğitim ve test alt-kümelerinde bulunan örnek vektörü sayılarının vuru tiplerine göre dağılımları Çizelge 5.4'de görülmektedir.

Çizelge 5.4 : VK ve VK_zaman kümelerinde bulunan örnek vektörü sayılarının vuru tiplerine göre dağılımları.

	Toplam	N	j	V	F	f	A	a	R
Toplam	11269	7596	219	757	872	215	677	135	798
Eğitim	2534	700	109	378	436	107	338	67	399
Test	8735	6896	110	379	436	108	339	68	399

5.3 Zaman Domeni Öznitelikleri ile EKG Vurularının Sınıflandırılması

Tez çalışmasının ilk aşamasında sadece zaman domeni öznitelikleri kullanılarak VK_zaman veri kümesindeki EKG vurularının sınıflandırılması işlemi gerçekleştirilmiştir. MIT-BIH veri tabanında bulunan 8 farklı EKG kalp vuru tipi üzerinde çalışılmıştır. Bu vuru tipleri sırayla normal (N), jonksiyonel kaçış vurusu (j), erken karıncık kasılması (V), karıncık vurusu ve normal vuru füzyonu (F), paced ve normal vuru füzyonu (f), kulakçık erken vurusu (A), anormal kulakçık erken vurusu (a) ve sağ dal bloğu (R) şeklindedir.

Bir veri kümesi içinde bulunan öbek sayısının belirlenmesi amacıyla, geçerlilik indisleri (validity indices) analizi uygulanır. Böylece, veri kümesinde bulunan verilerin optimum olarak kaç öbek ile temsil edilebileceği anlaşılır. Bu amaçla Ramd, Jaccard, Dunn vb. yöntemler kullanılabilir. Bölüm 4.1.2'de belirtildiği gibi bu indisler daha çok birbiriyle örtüşmeyen öbeklerin bulunduğu veri kümeleri üzerinde başarılı olmakta, aksi takdirde hatalı sonuç verebilmektedir. VK eğitim veri kümesi

üzerinde geçerlilik indisleri analizi uygulanmış, elde edilen optimum öbek sayıları Çizelge 5.5’de verilmiştir.

Çizelge 5.5 : VK eğitim veri kümesi üzerinde çeşitli geçerlilik indisi yöntemleri ile elde edilen optimum öbek sayıları.

Geçerlilik İndisi Yöntemi	Optimum Öbek Sayısı
Rand	8
Jaccard	8
Fowlkes-Mallows (FM)	8
Silhouette	2
Davies-Bouldin	2
Calinski-Harabasz	3
Dunn	2

Üzerinde çalışılan veri kümesinde sekiz farklı vuru tipi bulunmaktadır. Rand, Adjusted Rand, Jaccard ve FM geçerlilik indisi yöntemleri ile bu veri kümesi incelendiğinde optimum öbek sayısı olarak 8 öbek bulunmaktadır. Diğer yöntemlerin bulduğu öbek sayıları anlamlı değildir. Bu tez çalışması kapsamında öbek sayısı olarak 8 kullanılmıştır

5.3.1 MABC algoritması ve diğer sınıflayıcılar ile sınıflama sonuçları

Bölüm 5.2’de açıklanmış olan VK_zaman veri kümesi üzerinde MABC sınıflama algoritması çalıştırılmış ve Çizelge 5.6’da görülen sınıflama matrisi elde edilmiştir.

Çizelge 5.6 : MABC sınıflama algoritması ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6764	0	0	49	0	80	3	0
	j	1	83	11	5	10	0	0	0
	V	0	1	372	4	1	0	0	1
	F	91	0	2	325	10	1	5	2
	f	15	27	0	10	54	0	1	1
	A	1	3	0	11	0	322	1	1
	a	0	0	0	0	0	9	52	7
	R	0	0	0	0	0	0	0	399

Çizelge 5.6’da görülen sınıflama matrisine göre, VK_zaman test kümesinde bulunan tüm vuru tiplerinden toplam 8735 vurudan, 8371 tanesi doğru olarak sınıflandırılmıştır. Bu değerlere göre sınıflama başarımı %95.83 olup, MABC algoritmasının VK_zaman test kümesi üzerindeki performans matrisi Çizelge 5.7’de verilmiştir.

Çizelge 5.7 : MABC algoritmasının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	98.06	75.45	98.15	74.54	50.00	94.99	76.47	100.0
Özgüllük, Sp (%)	94.13	99.64	99.84	99.05	99.76	98.93	99.88	99.86
Poz.Öngörü, Ppr (%)	98.43	72.81	96.62	80.45	72.00	78.16	83.87	97.08

MABC algoritmasının performansını karşılaştırmak amacıyla, VK_zaman veri kümesi üzerinde diğer sınıflama algoritmaları ile de sınıflama işlemi yapılmıştır. Bu yöntemler: en yakın ortalama sınıflayıcı (NMC), k-en yakın komşular sınıflayıcı (KNN), Kohonen’in öz-düzenlemeli harita algoritması (SOM) ve karınca koloni optimizasyon algoritması (ACO) tabanlı sınıflayıcılardır. NMC, KNN, SOM ve ACO tabanlı sınıflayıcıların VK_zaman veri kümesi üzerinde çalışması ile elde edilen sınıflama matrisleri ve performans matrisleri aşağıda verilmiştir.

NMC en yakın ortalamalar sınıflayıcısının VK3_zaman veri kümesi üzerinde çalıştırılmasıyla elde edilen sınıflama ve performans matrisleri Çizelge 5.8 ve 5.9’da görülmektedir.

Çizelge 5.8 : NMC sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6588	0	0	0	0	290	18	0
	j	11	64	0	17	17	1	0	0
	V	2	1	299	53	0	0	3	21
	F	61	0	0	249	0	117	7	2
	f	34	23	0	2	39	8	2	0
	A	103	0	0	9	0	226	1	0
	a	33	0	0	0	0	3	17	15
	R	0	0	0	0	0	0	0	399

Bu değerlere göre NMC sınıflayıcısının sınıflama başarımının %90.22 olduğu görülmektedir.

Çizelge 5.9: NMC sınıflayıcısının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	95.53	58.18	78.89	57.11	36.11	66.67	25.00	100.0
Özgüllük, Sp (%)	86.73	99.72	100.00	99.02	99.80	95.01	99.64	99.54
Poz.Öngörü, Ppr (%)	96.43	72.73	100.00	75.45	69.64	35.04	35.42	91.30

KNN, K-en yakın komşu sınıflayıcısının VK_zaman veri kümesi üzerinde çalıştırılmasıyla elde edilen sınıflama ve performans matrisleri Çizelge 5.10 ve 5.11’de görülmektedir.

Çizelge 5.10 : KNN sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6547	0	0	289	11	49	0	0
	j	0	99	3	2	6	0	0	0
	V	0	0	379	0	0	0	0	0
	F	17	0	2	411	4	1	1	0
	f	4	1	0	8	95	0	0	0
	A	1	0	0	5	0	333	0	0
	a	0	0	0	2	1	4	60	1
	R	0	0	0	0	0	0	0	399

Bu değerlere göre KNN sınıflayıcısının sınıflama başarımının %95.28 olduğu görülmektedir.

Çizelge 5.11: KNN sınıflayıcısının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	94.94	90.00	100.0	94.27	87.96	98.23	88.24	100.0
Özgüllük, Sp (%)	98.80	99.99	99.94	96.31	99.75	99.36	99.99	99.99
Poz.Öngörü, Ppr (%)	99.67	99.00	98.70	57.32	81.20	86.05	98.36	99.75

Kohonen SOM algoritması tabanlı sınıflayıcı ağ boyutu ve yineleme sayısı olmak üzere iki ana kontrol parametresine sahiptir. Aynı zamanda, arama sürecindeki rasgele

arama özelliğinden ötürü, her çalıştırıldığında farklı sonuç vermektedir. Bu nedenle de her kontrol parametreleri çifti için üçer defa çalıştırılmış ve elde edilen sonuçlar Çizelge 5.12’de verilmiştir.

Çizelge 5.12 : Kohonen SOM algoritması tabanlı sınıflayıcının farklı kontrol parametreleri ile VK_zaman test kümesi üzerinde çalıştırılmasından elde edilen sınıflama başarımları.

Ağ boyutu	Yineleme	Deneme1 (%)	Deneme2 (%)	Deneme3 (%)	Ortalama (%)	Ortalama(Ort) %
10x10	10K	90.02	88.03	91.58	89.88	
10x10	20K	92.64	88.76	80.54	87.31	89.25
10x10	30K	91.10	86.97	93.62	90.56	
14x14	10K	87.47	92.75	90.14	90.12	
14x14	20K	93.16	93.04	92.79	93.00	92.27
14x14	30K	94.01	94.17	92.93	93.70	
20x20	10K	90.58	93.91	94.18	92.89	
20x20	20K	94.40	94.00	94.22	94.21	93.38
20x20	30K	93.59	93.77	91.76	93.04	
30x30	10K	93.49	95.17	94.17	94.28	
30x30	20K	94.36	94.34	94.63	94.38	94.28
30x30	30K	94.24	93.35	94.98	94.28	

Bu denemelerin içinden en yüksek sınıflama başarımına sahip olan denemeye ait sınıflama ve performans matrisleri Çizelge 5.13 ve 5.14’de görülmektedir.

Çizelge 5.13 : Kohonen SOM algoritması tabanlı sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6666	0	0	137	42	51	0	0
	J	12	86	1	0	11	0	0	0
	V	11	0	368	0	0	0	0	0
	F	64	3	3	355	8	3	0	0
	F	21	11	0	7	69	0	0	0
	A	8	1	0	1	0	328	1	0
	A	11	0	0	7	0	1	47	2
	R	5	0	0	0	0	0	0	394

Bu değerlere göre Kohonen SOM algoritması tabanlı sınıflayıcının sınıflama başarımının %95.17 olduğu görülmektedir.

Çizelge 5.14: Kohonen SOM algoritması tabanlı sınıflayıcının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	96.66	78.18	97.09	81.42	63.88	96.75	69.11	98.74
Özgüllük, Sp (%)	92.82	99.82	99.95	98.16	99.29	99.34	99.98	99.97
Poz.Öngörü, Ppr (%)	98.05	85.14	98.92	70.01	53.07	85.63	97.91	99.49

ACO, karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının VK_zaman veri kümesi üzerinde çalıştırılmasıyla elde edilen sınıflama ve performans matrisleri Çizelge 5.15 ve 5.16’da görülmektedir.

Çizelge 5.15 : ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcı ile VK_zaman test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6659	0	0	188	6	43	0	0
	J	0	91	4	3	12	0	0	0
	V	0	0	375	2	2	0	0	0
	F	30	0	4	393	5	1	3	0
	F	8	9	1	6	84	0	0	0
	A	6	0	0	4	0	329	0	0
	A	1	0	0	1	0	4	58	4
	R	0	0	0	0	0	0	0	399

Bu değerlere göre ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının sınıflama başarımının %96.03 olduğu görülmektedir.

Çizelge 5.16: ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının, VK_zaman test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	A	R
Duyarlık, Se (%)	96.56	82.73	98.94	90.14	77.78	97.05	85.29	100.0
Özgüllük, Sp (%)	97.55	99.90	99.89	97.54	99.71	99.43	99.97	99.95
Poz.Öngörü, Ppr (%)	99.33	91.00	97.66	65.83	77.06	87.27	95.08	99.01

5.4 Zaman ve Frekans Domeni Öznitelikleri ile EKG Vurularının Sınıflandırılması

Doktora tez çalışmasının ikinci aşamasında, Bölüm 5.2’de belirtilmiş olan VK veri kümesinde yer alan 8 farklı EKG kalp vuru tipi üzerinde hem zaman domeni, hem de

frekans domeni öznitelikleri kullanılarak, MABC, NMC, KNN, SOM ve ACO algoritmaları ile sınıflandırma işlemi gerçekleştirilmiştir. Elde edilen sonuçlar aşağıda verilmiştir.

5.4.1 MABC algoritması ve diğer sınıflayıcılar ile sınıflama sonuçları

VK veri kümesi üzerinde MABC sınıflama algoritması çalıştırılmış ve Çizelge 5.17’de görülen sınıflama matrisi elde edilmiştir.

Çizelge 5.17 : MABC sınıflama algoritması ile VK test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6807	0	0	17	19	49	4	0
	j	0	105	0	2	0	3	0	0
	V	0	0	354	25	0	0	0	0
	F	69	1	3	346	6	3	6	2
	f	6	3	0	3	94	0	2	0
	A	15	0	0	0	0	324	0	0
	a	1	1	0	0	0	3	60	3
	R	0	0	0	0	0	0	0	399

Çizelge 5.17’de görülen sınıflama matrisine göre, VK test kümesinde bulunan tüm vuru tiplerinden toplam 8735 vurudan, 8489 tanesi doğru olarak sınıflandırılmıştır. Bu sınıflama işlemi 0.31 saniye içinde gerçekleştirilmiştir. Bu değerlere göre sınıflama başarımları %97.18 olup, MABC algoritmasının VK test kümesi üzerindeki performans matrisi Çizelge 5.18’de verilmiştir.

Çizelge 5.18 : MABC algoritmasının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	98.71	95.45	93.40	79.36	87.04	95.58	88.24	100.0
Özgüllük, Sp (%)	95.05	99.94	99.96	99.43	99.71	99.31	99.86	99.94
Poz.Öngörü, Ppr (%)	98.68	95.45	99.16	88.04	78.99	84.82	83.33	98.76

MABC algoritmasının sınıflama performansını değerlendirmek amacıyla, literatürde yaygın olarak yer alan klasik sınıflama yöntemlerinden NMC, KNN sınıflayıcıları,

Kohonen SOM algoritması tabanlı sınıflayıcı ve doğadan esinlenen yöntemlerden olan ACO algoritması tabanlı sınıflayıcı kullanılmıştır.

NMC en yakın ortalamalar sınıflayıcısının VK veri kümesi üzerinde çalıştırılmasıyla elde edilen sınıflama ve performans matrisleri Çizelge 5.19 ve 5.20’de görülmektedir.

Çizelge 5.19 : NMC sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	2897	41	0	863	9	3074	12	0
	j	3	106	0	0	0	0	1	0
	V	0	0	354	13	12	0	0	0
	F	67	2	4	344	11	2	3	3
	f	11	4	0	0	92	0	1	0
	A	21	0	0	0	0	317	1	0
	a	5	1	0	0	0	2	55	5
	R	0	0	0	0	0	0	0	399

Bu değerlere göre NMC sınıflayıcısının sınıflama başarımının %52.25 olduğu görülmektedir. Bu sınıflama işlemi 0.31 saniye içinde gerçekleştirilmiştir.

Çizelge 5.20: NMC sınıflayıcısının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	42.01	96.36	93.40	78.90	85.19	93.51	80.88	100.00
Özgüllük, Sp (%)	94.18	99.44	99.95	89.44	99.63	63.34	99.79	99.90
Poz.Öngörü, Ppr (%)	96.44	68.83	98.88	28.20	74.19	09.34	75.34	98.03

KNN, K-en yakın komşu sınıflayıcısının VK veri kümesi üzerinde çalıştırılmasıyla elde edilen sınıflama ve performans matrisleri Çizelge 5.21 ve 5.22’de görülmektedir.

Bu değerlere göre KNN sınıflayıcısının sınıflama başarımının %96.73 olduğu görülmektedir. Bu sınıflama işlemi 5.72 saniye içinde gerçekleştirilmiştir.

Çizelge 5.21 : KNN sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6637	0	0	54	12	189	4	0
	j	0	110	0	0	0	0	0	0
	V	0	0	376	2	1	0	0	0
	F	3	0	1	427	2	0	1	2
	f	0	2	0	0	105	0	1	0
	A	6	1	0	0	0	332	0	0
	a	1	0	0	2	1	1	63	0
	R	0	0	0	0	0	0	0	399

Çizelge 5.22: KNN sınıflayıcısının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	96.24	100.0	99.21	97.94	97.22	97.94	92.65	100.0
Özgüllük, Sp (%)	99.46	99.97	99.99	99.30	99.81	97.74	99.94	99.98
Poz.Öngörü, Ppr (%)	99.85	97.35	99.73	88.04	86.78	63.60	91.30	99.50

Bölüm 3.3.2.3'te açıklandığı şekilde, Kohonen SOM algoritması tabanlı sınıflayıcı iki ana kontrol parametresine sahiptir. Bunlar ağ boyutu ve yineleme sayısıdır. Kontrol parametrelerinin sonuç üzerindeki etkisini görmek amacıyla, kontrol parametrelerine farklı değerler verilerek, sınıflayıcı birden fazla sayıda çalıştırılmıştır. Aynı zamanda, arama sürecindeki rasgele arama özelliğinden ötürü, her çalıştırıldığında farklı sonuç vermektedir. Bu nedenle de her kontrol parametreleri çifti için üçer defa çalıştırılmış ve elde edilen sonuçlar Çizelge 5.23'de verilmiştir.

Bu denemelerin içinden en yüksek sınıflama başarımına sahip olan denemeye ait sınıflama ve performans matrisleri Çizelge 5.24 ve 5.25'de görülmektedir.

Bu değerlere göre Kohonen SOM algoritması tabanlı sınıflayıcının sınıflama başarımının %94.88 olduğu görülmektedir. Bu sınıflama işlemi 10.32 saniye içinde gerçekleştirilmiştir.

Çizelge 5.23 : Kohonen SOM algoritması tabanlı sınıflayıcının farklı kontrol parametreleri ile VK test kümesi üzerinde çalıştırılmasından elde edilen sınıflama başarımları sonuçları.

Ağ boyutu	Yineleme	Deneme1 (%)	Deneme2 (%)	Deneme3 (%)	Ortalama (%)	Ortalama(Ort) %
10x10	10K	85.62	75.00	81.38	80.67	
10x10	20K	76.88	94.29	73.18	81.45	79.72
10x10	30K	63.76	86.86	80.54	77.05	
14x14	10K	80.53	80.91	93.87	85.10	
14x14	20K	91.75	79.42	88.01	86.39	86.75
14x14	30K	87.08	86.47	92.74	88.76	
20x20	10K	94.02	89.58	94.42	92.67	
20x20	20K	90.67	94.13	90.44	91.75	92.17
20x20	30K	92.18	90.90	93.20	92.09	
30x30	10K	94.47	94.13	94.41	94.34	
30x30	20K	93.98	94.33	93.04	93.78	94.01
30x30	30K	92.32	94.49	94.88	93.90	

Çizelge 5.24 : Kohonen SOM algoritması tabanlı sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6612	2	1	22	41	215	2	1
	j	7	103	0	0	0	0	0	0
	V	15	0	360	4	0	0	0	0
	F	38	0	0	394	3	1	0	0
	f	15	2	0	3	86	1	0	1
	A	38	1	0	1	0	299	0	0
	a	11	0	0	3	0	1	52	1
	R	17	0	0	0	0	0	0	382

Çizelge 5.25: Kohonen SOM algoritması tabanlı sınıflayıcının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	95.88	93.64	94.99	90.37	79.63	88.20	76.47	95.74
Özgüllük, Sp (%)	92.33	99.94	99.99	99.60	99.49	97.40	99.98	99.96
Poz.Öngörü, Ppr (%)	97.91	95.37	99.72	92.27	66.15	57.83	96.30	99.22

ACO, karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının VK veri kümesi üzerinde çalıştırılmasıyla elde edilen sınıflama ve performans matrisleri Çizelge 5.26 ve 5.27’de görülmektedir.

Çizelge 5.26 : ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcı ile VK test kümesi üzerinde elde edilen sınıflama matrisi.

		Etiketlenen Vuru Tipi							
		N	j	V	F	f	A	a	R
Gerçek Vuru Tipi	N	6674	0	0	55	6	161	0	0
	j	0	110	0	0	0	0	0	0
	V	0	0	373	5	1	0	0	0
	F	2	1	1	423	4	1	2	2
	f	1	1	0	1	103	1	1	0
	A	23	0	0	0	0	316	0	0
	a	2	0	0	2	2	1	61	0
	R	0	0	0	0	0	0	0	399

Bu değerlere göre ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının sınıflama başarımının %96.84 olduğu görülmektedir. Bu sınıflama işlemi 6.53 saniye içinde gerçekleştirilmiştir.

Çizelge 5.27: ACO karınca kolonisi optimizasyon algoritması tabanlı sınıflayıcının, VK test kümesi üzerindeki sınıflama işlemine ait performans matrisi.

	N	j	V	F	f	A	a	R
Duyarlık, Se (%)	96.78	100.0	98.42	97.02	95.37	93.22	89.71	100.0
Özgüllük, Sp (%)	98.48	99.98	99.99	99.24	99.85	98.05	99.97	99.98
Poz.Öngörü, Ppr (%)	99.58	98.21	99.73	87.04	88.79	65.83	95.31	99.50

5.5 K-Katlı Çapraz Doğrulama ile Sınıflama Sonuçları

MABC algoritmasının sınıflama performansının, eğitim kümesinde yer alan örnek vektörlerinden bağımsız olduğunu görmek amacıyla Bölüm 4.3'te açıklanmış olan K-katlı çapraz doğrulama yöntemi kullanılmıştır. Çapraz doğrulama yöntemini kullanmak üzere öncelikle VK_KFold adında bir veri kümesi oluşturulmuştur. Bu veri kümesinde, aritmetik vurulara ait tüm örnekler kullanılmış, ancak incelenen VK veri kümesinde normal vurular çok sayıda bulunduğu için bu tip vurulardan rasgele seçilen 700 tanesi kullanılmıştır. Bu yöntemin gereği olarak, öncelikle Çizelge 5.28'de detayları verilmiş olan VK-KFold veri kümesi rasgele seçilen örneklerden oluşan beş eşit örnek sayısına sahip parçaya bölünmüştür. Bu beş parçanın dörtlü kombinasyonları bir araya getirilip eğitim kümeleri (Ex), beşincileri ise test kümesi (Tx) olarak kullanılmıştır.

Çizelge 5.28 : VK-KFold kümesinde bulunan örnek vektörlerinin vuru tiplerine göre dağılımları.

Toplam	N	j	V	F	f	A	a	R
4373	700	219	757	872	215	677	135	798

VK-KFold veri kümesinden hareketle oluşturulan Ex-Tx alt kümelerinde yer alan, farklı vuru tiplerine ait örnek vektörlerinin sayısı Çizelge 5.29’da verilmiştir.

Çizelge 5.29 : VK-KFold, Ex-Tx kümelerinde yer alan, farklı vuru tiplerine ait örnek vektörlerinin sayıları.

	Toplam	N	j	V	F	f	A	a	R
E1	3498	569	176	600	708	167	548	108	622
T1	875	131	43	157	164	48	129	27	176
E2	3498	559	180	607	683	180	550	104	635
T2	875	141	39	150	189	35	127	31	163
E3	3499	557	180	612	693	169	541	110	637
T3	874	143	39	145	179	46	136	25	161
E4	3499	550	172	614	699	179	529	110	646
T4	874	150	47	143	173	36	148	25	152
E5	3498	565	168	595	705	165	540	108	652
T5	875	135	51	162	167	50	137	27	146

Bu Ex-Tx kümelerinin üzerinde MABC sınıflama programı çalıştırılarak elde edilen sınıflama başarımları Çizelge 5.30’da görülmekte olduğu gibi, birbirine yakındır. Beş farklı Ex-Tx çiftinin sınıflama başarımlarının ortalaması %92.84’dür. Her bir çift için ortalamadan sapmalar da Çizelge 5.30’da verilmiştir. Bu sonuçlara göre, MABC algoritmasının performansının eğitim kümesinde kullanılan örnek vektörlerinden bağımsız olduğu görülmektedir.

Çizelge 5.30 : VK-KFold, Ex-Tx kümeleri kullanılarak elde edilen sınıflama başarımları.

	Sınıflama Başarımları (%)	Ortalamadan sapma (%)
E1-T1	92.69	- 0.15
E2-T2	92.91	+ 0.07
E3-T3	91.30	- 1.54
E4-T4	93.82	+ 0.98
E5-T5	93.49	+ 0.65

5.6 Sınıflayıcı Performanslarının Karşılaştırılması

Detayları Bölüm 5.2’de verilmiş olan VK_zaman veri kümesi üzerinde farklı sınıflama yöntemleri ile elde edilen sınıflama başarımları ve kalp vuru tipi bazında duyarlık değerleri Çizelge 5.31’de verilmiştir.

Çizelge 5.31 : VK_zaman veri kümesi üzerinde farklı sınıflama yöntemleri ile elde edilen sınıflama başarımları.

Sınıflama Yöntemi	Sınıflama Başarımı (%)	Duyarlık (%)							
		N	j	V	F	f	A	a	R
MABC	95.83	98.06	75.45	98.15	74.54	50.00	94.99	76.47	100.0
NMC	90.22	95.53	58.18	78.89	57.11	36.11	66.67	25.00	100.0
KNN	95.28	94.94	90.00	100.0	94.27	87.96	98.23	88.24	100.0
SOM	95.17	96.66	78.18	97.10	81.42	63.89	96.76	69.12	98.75
ACO	96.03	96.56	82.73	98.94	90.14	77.78	97.05	85.29	100.0

Sadece zaman domeni öznitelikleri kullanılarak yapılan sınıflama işlemlerinde, sınıflama başarımlarının genel olarak %90 ve üzerinde olduğu görülmektedir. Ancak, kaçış vurularında (j tipi vurular), füzyon tipi vurularda (F ve f tipi vurular) ve anormal erken kulakçık vurularında (a tipi vurular) duyarlık değerlerinde iyileştirme gerektiği görülmüştür. Bu amaçla, zaman domeni özniteliklerine ek olarak, frekans domeni öznitelikleri de kullanılarak veri kümesi genişletilmiştir. Örnek vektörleri sayısı aynı olmakla birlikte, örnek vektörlerinin boyutu 7’den 15’e arttırılmıştır.

VK adı verilen bu yeni veri kümesi üzerinde sınıflama işlemleri gerçekleştirilmiş, duyarlık değerlerinde anlamlı artışlar olduğu görülmüştür. VK veri kümesi üzerinde farklı sınıflama yöntemleri ile elde edilen sınıflama başarımları ve kalp vuru tipi bazında duyarlık değerleri Çizelge 5.32’de verilmiştir.

Çizelge 5.32 : VK veri kümesi üzerinde farklı sınıflama yöntemleri ile elde edilen sınıflama başarımları.

Sınıflama Yöntemi	Sınıflama Başarımı (%)	Duyarlık (%)							
		N	j	V	F	f	A	a	R
MABC	97.18	98.71	95.45	93.40	79.36	87.04	95.58	88.24	100.0
NMC	52.25	42.01	96.36	93.40	78.90	85.19	93.51	80.88	100.00
KNN	96.73	96.24	100.0	99.21	97.94	97.22	97.94	92.65	100.0
SOM	94.88	95.88	93.64	94.99	90.37	79.63	88.20	76.47	95.74
ACO	96.84	96.78	100.0	98.42	97.02	95.37	93.22	89.71	100.0

VK veri kümesi üzerinde farklı sınıflama yöntemlerinin sınıflama süreleri Çizelge 5.33’de verilmiştir.

Çizelge 5.33 : VK veri kümesi üzerinde farklı sınıflama yöntemlerinin sınıflama süreleri.

Sınıflama Yöntemi	Sınıflama Süresi (s)
MABC	0.31
NMC	0.31
KNN	5.72
SOM	10.32
ACO	6.53

Sınıflayıcı performanslarının karşılaştırılması aşamasında beş ölçüt dikkate alınır. Bunlar, sınıflama başarımları, duyarlık, özgüllük ve pozitif öngörü ile test kümesi sınıflama süresi değerleridir.

VK veri kümesi üzerinde MABC algoritması ile elde edilen sınıflama başarımları en yüksek değere sahip olup, %97.18’dir. NMC sınıflayıcısı ile elde edilen sınıflama başarımları %52.25, KNN sınıflayıcısı ile %96.73, Kohonen SOM algoritması tabanlı sınıflayıcının sınıflama başarımları %94.88 ve ACO algoritması tabanlı sınıflayıcının sınıflama başarımları ise %96.84’dür.

En yüksek sınıflama başarımları MABC ile elde edilmiştir. ACO ve KNN ona yakın başarıma sahip iken, Kohonen SOM algoritması tabanlı sınıflayıcının sınıflama başarımları daha düşüktür. NMC sınıflayıcısı ise, diğerlerine göre daha hızlı çalışmakla birlikte en düşük sınıflama başarımlarına sahiptir.

Duyarlık, özgüllük ve pozitif öngörü açısından, farklı sınıflayıcıların farklı vuru tipleri için değişen değerlere sahip oldukları görülmektedir. MABC, duyarlık ölçütünde normal tip vurularda, özgüllük ve pozitif öngörü ölçütünde A tipi vurularda en yüksek değerlere sahiptir.

NMC sınıflayıcısının, f ve a tipi vurularda çok düşük duyarlık değerlerine sahip olduğu görülmektedir. Aynı zamanda j, F ve A tipi vurular için de duyarlık değerleri düşüktür.

Kohonen SOM algoritması tabanlı sınıflayıcı ise, pozitif öngöründe f tipi vurularda %66'ya, A tipi vurularda %58'e düşmektedir.

K-NN, eğitim aşamasında herhangi bir öbekleme işlemi yapmamaktadır. Test aşamasında, sıradaki test örnek vektörünün bütün eğitim örnek vektörleri ile olan mesafeleri kontrol edilir. En yakın "k" adet örnek vektörünün sınıf etiketlerine göre test vektörü sınıflanmaktadır.

ACO tabanlı sınıflayıcıda ise, eğitim aşamasında ACO algoritması ile öbekleme işlemi yapılır. Test aşamasında, test örnekleri üzerinde sınıflama yapılırken ise KNN sınıflayıcı kullanılmaktadır. Bir test örneğine en yakın eğitim örneği belirlenir. Bu eğitim örneğinin ait olduğu öbeğin sınıf etiketi ile test örneği sınıflandırılır.

Sonuç olarak, bu veri kümesi için ACO ile KNN yöntemlerinin sınıflama başarımları birbirine benzer olmaktadır. Ancak, yukarıda açıklanan şekilde yöntemlerdeki farklılıktan dolayı, KNN ile sınıflama yapıldığında ACO'ya göre biraz daha iyi duyarlıklar elde edilirken, ACO algoritması tabanlı sınıflayıcıda ise, normal vuru tipleri KNN'e göre biraz daha iyi duyarlılıkta sınıflanabilmektedir.

MABC ve KNN sınıflayıcıların sınıflama matrisleri incelendiğinde (Çizelge 5.17 ve Çizelge 5.21 incelendiğinde), en çok hatalı sınıflamaların N, A ve F tipi vurular arasında gerçekleştiği görülmektedir. A tipi vurular, kulakçık erken vuruları olup, şekil özellikleri açısından normal vurulara çok benzemekte, sadece kendisinden önce gelen vuru ile arasında bulunan süre açısından normal vurulardan ayrılmaktadır. F tipi vurular, normal vurular ile karıncık vurularının füzyonudur, bu nedenle önemli ölçüde normal vurulara benzer özellikler göstermektedir. MABC sınıflayıcısı, normal

vuruları çok daha az oranda hatalı pozitif A olarak sınıflarken, KNN sınıflayıcısı çok sayıda normal vuruyu, hatalı pozitif A olarak sınıflamaktadır. Bu nedenle, MABC’de normal vuruların duyarlılığı daha yüksektir. Buna karşılık MABC sınıflayıcısının, F tipi vuruları daha yüksek oranda hatalı pozitif N olarak sınıfladığı görülmektedir. KNN sınıflayıcısında F tipi vuruların duyarlılığı anlamlı miktarda daha iyi durumdadır.

Çizelge 5.17 ve Çizelge 5.21’de, gerçekte normal vuru olduğu halde, hatalı pozitif A tipi vuru olarak etiketlenmiş vuruların MABC’de 49, KNN’de 189 olduğu görülmektedir. MABC algoritmasının yapısında, sınıf örneklerini temsil eden sınıf merkez koordinatının hesaplanmasında, tüm örneklerin konumları dikkate alınmaktadır. KNN sınıflayıcısında ise, bir test örnek vektörü, eğitim kümesindeki en yakın komşuların sınıf etiketine göre sınıflandırılmaktadır. Bunun sonucunda, test kümesindeki çok daha fazla sayıda normal vuru KNN tarafından hatalı pozitif A olarak etiketlenilmektedir. Buna karşılık MABC algoritmasında normal vuru tipi açısından daha düşük bir hata oranı mevcuttur.

Benzer durum, normal vuru olduğu halde, hatalı pozitif F veya f tipi vuru olarak etiketlenen vurular için de geçerlidir. MABC’de sadece 17 ve 19 vuru, KNN’de ise 54 ve 12 vuru gerçekte normal vuru olduğu halde, hatalı pozitif F ve f tipi olarak sınıflandırılmıştır.

Sonuç olarak, MABC algoritması, incelenen EKG verileri üzerinde, karşılaştırılan tüm diğer sınıflayıcılara göre en yüksek sınıflama başarımına sahip olan yöntem olmuştur. Ayrıca, test kümesinde bulunan toplam 8735 vuru, 0.31 saniye içinde sınıflanmıştır. Hesaplama süresinin de sınıflama başarımı oldukça düşük olan NMC haricindeki tüm diğer yöntemlerden daha kısa olduğu görülmektedir. Bu sayede, MABC algoritması, gerçek zamandaki EKG işaretlerinin analizinde de kullanılmaya uygun bir yöntem olmaktadır.

Elde edilen başarılı sonuçlar dikkate alınarak, geliştirilmiş olan MABC tabanlı sınıflayıcının değişik kullanıcılar tarafından çeşitli veri kümeleri üzerinde kolaylıkla kullanılabilmesini sağlamak amacıyla, tez çalışması kapsamında bir grafik kullanıcı ara yüzü (Graphical User Interface – GUI) hazırlanmıştır. Bu kullanıcı arayüzü hakkında bilgiler Ek-A’da verilmiştir.

6. SONUÇLAR

Bu çalışmada elektrokardiyogram verilerinin yapay arı kolonisi (ABC) algoritması ile analizi gerçekleştirilmiştir. Bu süreçte, ABC algoritması tabanlı sınıflayıcının performansını arttırmak için çeşitli iyileştirmeler yapılmış ve elde edilen bu yeni yönteme “İyileştirilmiş ABC (MABC)” adı verilmiştir. Geliştirilen yöntem, az sayıda kontrol parametresine sahiptir ve yüksek sınıflama başarımını hızlı bir şekilde gerçekleştirmektedir. Test vektörlerinin sınıflandırılması aşamasında, hafızada sadece sınıfları temsil eden öbek merkezlerinin bulunması yeterli olmaktadır. Bu da, gerçek zamanlı bir sistem bünyesinde kullanılmasını kolaylaştırmaktadır.

Bu sınıflama yöntemi MABC ile, MIT-BIH veri tabanından alınan elektrokardiyogram verilerindeki kalp vuruları sınıflandırılmıştır. Önce detaylı öznitelik çıkartma işlemleri gerçekleştirilmiş, en yüksek ayırt ediciliğe sahip özniteliklerin seçilmesi için diverjans analizi yöntemi kullanılmıştır. Zaman domeni öznitelikleri kullanılarak sınıflama yapıldığında, bazı vuru tipleri için duyarlık değerlerinin artırılmasına ihtiyaç olduğu görülmüştür. Bu nedenle zaman domeni özniteliklerine ek olarak, frekans domeni öznitelikleri de birlikte kullanılmıştır. Elde edilen sonuçlar literatürde yaygın olarak yer alan klasik sınıflama yöntemlerinden NMC, KNN sınıflayıcıları, Kohonen SOM algoritması tabanlı sınıflayıcı ve doğadan esinlenen yöntemlerden olan ACO algoritması tabanlı sınıflayıcı sonuçları ile karşılaştırılmıştır.

MABC algoritması ile elde edilen sınıflama başarımı en yüksek değere sahiptir. Karşılaştırılan yöntemlerden ACO ve KNN sınıflayıcıları, MABC’ye yakın sınıflama başarımına sahiptir. Ancak, bu sınıflayıcılarda sınıflama süresi daha uzundur ve test aşamasında hafızada tüm eğitim örneklerini tutmak zorunda oldukları için, gerçek zamanlı uygulamalara çok uygun değildirler. NMC sınıflayıcı ise, çok hızlı bir şekilde sınıflama yapabilmekte, fakat en düşük sınıflama başarımı vermektedir. SOM tabanlı sınıflayıcı ise, MABC’ye göre daha uzun bir sürede işlem yapmasının yanı sıra, sınıflama başarımında da biraz daha düşük değerlerde sonuç vermektedir.

Duyarlık, özgüllük ve pozitif öngörü açısından, farklı sınıflayıcıların farklı vuru tipleri için değişen değerlere sahip oldukları görülmektedir.

MABC algoritması, tez çalışması kapsamında sadece EKG verilerinin değil, IRIS, PIMA, BUPA vb başka veri kümelerinin sınıflanmasında da kullanılmıştır (Dilmaç ve Korürek, 2014). Bu kapsamda ilerideki çalışmalarda, EEG ve EMG gibi diğer biyomedikal işaretlerin sınıflanması için de özellikle önerilir. Ayrıca, aynı sınıfta olmakla birlikte farklı öbeklerde yer alan veri tiplerinin daha yüksek başarımla sınıflanabilmesi için, bir sınıfın örneklerinin birden fazla öbekte yer alabileceği bir yapının geliştirilmesi gelecek çalışma olarak düşünülmektedir.

Literatürde yer alan çok sayıda çalışma MIT-BIH veri tabanını kullanmasına karşın, her çalışmada farklı kayıtlardaki farklı örnek vuruların kullanıldığı görülmektedir. Bu durum ise, sınıflayıcı performanslarının sağlıklı bir şekilde karşılaştırılmasını engellemektedir. Yapılan karşılaştırmaların anlamlı olabilmesi için, sınıflayıcı algoritmaların standart veri kümeleri üzerinde test edilmeleri açık bir zorunluluktur. Bu amaçla, değişik sınıflama zorluğu seviyelerinde vuru örneklerinden oluşan standart veri kümelerinin oluşturularak, uluslararası araştırmacıların hizmetine sunulması düşünülmektedir. İncelenen EKG veri kümesindeki farklı vurulara ait sınıf sayısının artırılıp sınıflama başarımının incelenmesi, ayrıca dalgacık dönüşümü vb yöntemlerle yeni özniteliklerin araştırılması düşünülmektedir.

Hastaların durumu hakkında gerçek bilgiye sahip olmak için sadece vuru tiplerini incelemek yeterli değildir. Hastaların gerçek durumlarını anlayabilmek için tıbbi görüntüleme sonuçları, kan değerleri, kilosu, yaşı, cinsiyeti, geçmiş yaşadığı kalp rahatsızlıkları v.b. gibi birçok parametrenin birlikte incelenmesi gerekir. Bu bağlamda hastaların durumlarını anlayabilmek için hasta veri tabanı üzerinde analiz yapan bir yapay zekâ algoritmasına daha çok ihtiyaç duyulacaktır. Doktora tez çalışmasının konusu sadece vuru tiplerinin belirlenmesi üzerinedir; çalışmada gerçekleştirilen tanıma algoritmaları, bir üst yapay zekâ algoritması için yardımcı program olacaktır.

Tez çalışması kapsamında geliştirilmiş olan MABC algoritması tabanlı sınıflayıcı, hızlı bir şekilde ve yüksek doğrulukla vuru tiplerini belirleyebilmektedir. Günümüzdeki mevcut teknolojiler ile EKG cihazında gömülü PC kullanılarak elektrokardiyogram işaretinin ön işleme, normalizasyon, öznitelik çıkarma ve sınıflama sürecinin gerçek zamanlı olarak yapılabilmesinin mümkün olduğu gözükmemektedir. 24 saatlik veri kaydını incelemek uzun zaman almaktadır. Gerçek zamanda vuruların etiketlenerek saklanması, veri kaydı tamamlandıktan sonra doktorların problem yaşanan zaman aralıklarına hızla bakabilmelerine imkân verecektir.

KAYNAKLAR

- Abawajy, J.H., Kelarev, A.V. & Chowdhury, M.** (2013). Multistage approach for clustering and classification of ECG data. *Computer Methods and Programs in Biomedicine*, 112 (3), 720–730.
- Abdullah, A., Ku-Mahamud, K.R. & Sediyo, A.** (2013). A new feature set partitioning method for nearest mean classifier ensembles. *Proceedings of the 4th International Conference on Computing and Informatics, ICOCI 2013* (pp.39-44). Sarawak, Malaysia: August 28-30.
- Afkhami, R.G., Azarnia, G. & Tinati, M.A.** (2016). Cardiac arrhythmia classification using statistical and mixture modeling features of ECG signals. *Pattern Recognition Letters*, 70 (C), 45-51.
- Ahirwal, M.K., Kumar, A. & Singh, G.K.** (2014). Adaptive filtering of EEG/ERP through bounded range Artificial Bee Colony (BR-ABC) algorithm. *Digital Signal Processing*, 25, 164-172.
- Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Perez, M. & Perona, I.** (2013). An extensive comparative study of cluster validity indices. *Pattern Recognition*, 46 (1), 243-256.
- Babu, M.S.P. & Rao, N.T.** (2010). Implementation of artificial bee colony (ABC) algorithm on garlic expert advisory system. *International Journal of Computer Science and Research*, 1 (1), 69-74.
- Ballabio, D. & Vasighi, M.** (2012). A MATLAB toolbox for Self Organizing Maps and supervised neural network learning strategies, *Chemometrics and Intelligent Laboratory Systems* 118, 24-32.
- Basturk, B. & Karaboga, D.** (2006). An artificial bee colony (ABC) algorithm for numeric function optimization. *IEEE Swarm Optimization Symposium*, (pp. 687-697) Indiana, USA: May 12.
- Bolaji, A.L., Khader, A.T., Albetar, M.A. & Awadallah, M.A.** (2013). Artificial bee colony algorithm, its variants and applications: a survey. *Journal of Theoretical and Applied Information Technology*, 47 (2), 434-459.
- Bolshakova, N. & Azuaje, F.** (2003). Cluster validation techniques for genome expression data. *Signal Processing*, 83 (4), 825-833.
- Chen, G., Jaradat, S. A., Banerjee, N., Tanaka, T. S., Ko, M.S.H. & Zhang, M. Q.** (2002). Evaluation and Comparison of Clustering Algorithms in Analyzing ES Cell Gene Expression Data. *Statistica Sinica*, 12, 241-262.
- Chen, S. Hua, W., Li, Z., Li, J. & Gao, X.** (2017). Heartbeat classification using projected and dynamic features of ECG signal, *Biomedical Signal Processing and Control*, 31, 165–173.
- Chung, E.** (1989). *Principles of Cardiac Arrhythmias*. 4th Ed, Baltimore, Maryland, USA: Williams and Wilkins.

- Cohen, A.** (1986). *Biomedical signal processing*, volume II, CRC Press, Boca Raton, Florida, USA, 75-79.
- Cuevas, E., Zaldivar, D., Perez-Cisnerosa, M., Sossa, H. & Osuna, V.** (2013). Block matching algorithm for motion estimation based on Artificial Bee Colony (ABC), *Applied Soft Computing*, 13 (6), 3047-3059.
- De, A. & Guo, C.** (2015). An adaptive vector quantization approach for image segmentation based on SOM network, *Neurocomputing*, 149 (PA), 48-58.
- Dilmac, S. & Korurek, M.** (2015). ECG heart beat classification method based on modified ABC algorithm, *Applied Soft Computing*, 36 (C), 641–655.
- Dilmaç, S. & Korürek, M.** (2014). ABC algoritmasının sınıflamada kullanımı ve kontrol parametrelerinin etkilerinin incelenmesi, *BIYOMUT 2014*, 18. *Biyomedikal Mühendisliği Ulusal Toplantısı*. İstanbul, Türkiye: Ekim 16-17.
- Dilmac, S., Nizam, A. & Korurek, M.** (2014). Evaluation of a new heart beat classification method based on ABC algorithm, comparison with GA, PSO and ACO classifiers. *International Journal of Reasoning-based Intelligent Systems*, 3/4 (6), 98–108.
- Dilmac, S. & Korurek, M.** (2013a). A new ECG arrhythmia classification method by using modified ABC (MABC) algorithm. *Proceedings of the Fourth International Conference on Mathematical and Computational Applications*, (pp.390-398). Manisa, Turkey: June 11-13.
- Dilmac, S., Nizam, A. & Korurek, M.** (2013b). Modified artificial bee colony algorithm based ECG beat type classification method, comparison with ACO based classifier. *Proceedings of the First International Symposium on Innovative Technologies in Engineering and Science*, (pp.1157-1166). Sakarya, Turkey: June 7-9.
- Dilmac, S. & Korurek, M.** (2013c). A New ECG arrhythmia clustering method based on modified artificial bee colony algorithm, comparison with GA and PSO classifiers. *International Symposium on Innovations in Intelligent Systems and Applications (INISTA), IEEE International Symposium*, (pp. 1-5). Albena, Bulgaria: June 19-21.
- Dilmac, S. & Olmez, T.** (2017). Nature inspired algorithm MABC for clustering and classification of ECG heart beats, using time and frequency domain features. *Proceedings of the 10th International Conference on Electrical and Electronics Engineering*. Bursa, Turkey: November 30 – December 2.
- Dogan, B. & Korurek, M.** (2012). A new ECG beat classification method based on kernelized fuzzy c-means and hybrid ant colony optimization for continuous domains. *Applied Soft Computing*, 12 (11), 3442–3451.
- Dokur, Z., Olmez, T., Yazgan, E. & Ersoy, O.K.** (1997). Detection of ECG waveforms by neural networks, *Medical Engineering & Physics*, 19 (8), 738–741.

- Dokur, Z.** (2000). *Yapay sinir ağları ve genetik algoritmalar kullanılarak EKG vurularının sınıflandırılması*. (Doktora tezi). İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Dokur, Z. & Olmez, T.** (2001). ECG beat classification by a novel hybrid neural network. *Computer Methods and Programs in Biomedicine*, 66 (2), 167-181.
- Dokur, Z.** (2006). *Pattern Recognition by Neural Networks*, Ders Notları. İstanbul: İstanbul Teknik Üniversitesi.
- Dong, X., Wang, C. & Wenjie, S.** (2017). ECG beat classification via deterministic learning. *Neurocomputing*, 240, 1-12.
- Dorigo, M., Maniezzo, V. & Colorni, A.** (1991). *Ant System: An Autocatalytic Optimizing Process*. (Technical Report, 91-016). Milano, Italy: Politecnico di Milano, 1–21.
- Dorigo, M., Maniezzo, V. & Colorni, A.** (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions On Systems Man And Cybernetics*, 26 (1), 29–41.
- Dorigo, M. & Stützle, T.** (2004). *Ant Colony Optimization*. Cambridge MA, USA: MIT Press. ISBN 0-262-04219-3.
- Dudoit, S. & Fridlyand, J.** (2002). A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3 (7), 1-21.
- Elhaj, F.A., Salim, N., Harris, A.R., Swee, T.T. & Ahmed, T.** (2016). Arrhythmia recognition and classification using linear and nonlinear features of ECG signals. *Computer Methods and Programs in Biomedicine*, 127, 52-63.
- Fielding, A.H.** (2007). *Cluster and Classification Techniques for the Biosciences*. New York, USA: Cambridge University Press. ISBN-13 978-0-521-61800-7.
- Fisher, R.A.** (1936). The use of multiple measurements in taxonomic problems, *Annals of Eugenics*, 7 (2), 179–188.
- Forgy, E.W.** (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classification, *Biometrics*, 21 (3), 768–769.
- Fukunaga, K.** (1990). *Introduction to statistical pattern recognition* (2nd ed.). San Diego, CA, USA: Academic Press Professional Inc.
- Gacek, A. & Pedrycz, W.** (2012). *ECG Signal Processing, Classification and Interpretation*. London, UK: Springer-Verlag
- Garcia, T.B. & Miller, G.T.** (2004). *Arrhythmia Recognition, The Art of Interpretation*. Sudbury, Massachusetts, USA: Jones and Bartlett Publishers.
- Garro, B.A., Rodriguez, K. & Vazquez, R.A.** (2016). Classification of DNA microarrays using artificial neural networks and ABC algorithm. *Applied Soft Computing*, 38 (C), 548-560.
- Goldberg, D.E.** (1989). *Genetic Algorithms in Search, in: Optimization and Machine Learning*. Boston, MA, USA: Addison-Wesley Pub. Co.

- Guler, I. & Ubeyli, D.** (2005). ECG beat classifier designed by combined neural network model. *Pattern Recognition*, 38 (2), 199-208.
- Guyton, A.C. & Hall, J.E.** (2006). *Textbook of Medical Physiology 11th. Edition*, Philadelphia, PA, USA: Elsevier Saunders.
- Halkidi, M., Batistakis, Y. & Vazirgiannis, M.** (2001). On Clustering Validation Techniques. *Intelligent Information Systems Journal*, 17 (2-3), 107-145.
- Hall, P., Park, B.U. & Samworth, R.J.** (2008). Choice of neighbor order in nearest-neighbor classification. *Annals of Statistics*, 36 (5), 2135–2152.
- Homaeinezhad, M.R., Atyabi, S.A., Tavakkoli, E., Toosi, H.N., Gaffari, A. & Ebrahimpour, R.** (2012). ECG arrhythmia recognition via a neuro-SVM-KNN hybrid classifier with virtual QRS image-based geometrical features. *Expert Systems with Applications*, 39 (2), 2047–2058.
- Jain, A.K.** (2010). Data clustering: 50 years beyond K-means, *Pattern Recognition Letters*, 31 (8), 651-666.
- Javadi, M., Arani, S.A.A., Sajedin, A. & Ebrahimpour, R.** (2013). Classification of ECG arrhythmia by a modular neural network based on Mixture of Experts and Negatively Correlated Learning, *Biomedical Signal Processing and Control*, 8 (3), 289-296.
- Kang, F., Li, J. & Xu, Q.** (2009). Structural inverse analysis by hybrid simplex artificial bee colony algorithm, *Computers and Structures*, 87 (13), 861–870.
- Karaboga, D.** (2004). *Yapay zeka optimizasyon algoritmaları*. Istanbul: Atlas yayın dağıtım.
- Karaboga, D.** (2005). An idea based on honey bee swarm for numerical optimization (Technical report: TR06), Erciyes University, Engineering Faculty, Computer Engineering Department.
- Karaboga, D. & Akay, B.** (2007). Artificial bee colony (ABC) algorithm on training artificial neural networks. In: *SIU 2007 IEEE 15th signal processing and communications applications*, (pp. 818–821). Eskisehir, Turkey: Anadolu University, June 11-13.
- Karaboga, D., Okdem, S. & Ozturk, C.** (2010). Cluster based wireless sensor network routings using artificial bee colony algorithm. In: *2010 international conference on autonomous and intelligent systems (AIS)*, (pp.1-5). Porto, Portugal: June, 21-23.
- Karaboga, D. & Ozturk, C.** (2011). A novel clustering approach: artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 11 (1), 652–657.
- Karaboga, D., Gorkemli, B., Ozturk, C. & Karaboga, N.** (2014). A comprehensive survey: artificial bee colony (ABC) algorithm and applications, *Artificial Intelligence Review*, 42 (1), 21-57.
- Kasturi, J., Acharya R. & Ramanathan, M.** (2003). An information theoretic approach for analyzing temporal patterns of gene expression. *Bioinformatics*, 19 (4), 449-458.

- Kennedy, J. & Eberhart, R.C.** (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, (pp. 1942-1948). Perth, Western Australia: The University of Western Australia, November, 27 – December, 1.
- Khan, M. G.** (2008). *Rapid ECG Interpretation*. Totowa, NJ, USA: Humana Press.
- Kiranyaz, S., Ince, T., Pulkkinen, J. & Gabbouj. M.** (2011). Personalized long-term ECG classification: A systematic approach. *Expert Systems with Applications*, 38 (4), 3220-3226.
- Kohonen, T.** (1990). The self-organizing maps. *Proceedings of the IEEE*, 78 (9), 1464–1480.
- Kundu, M., Nasipuri, M. & Basu, D.K.** (2000). Knowledge-based ECG interpretation: a critical review. *Pattern Recognition*, 33 (3), 351-373.
- Korurek, M.** (1996). *Tip elektroniğinde tasarım ilkeleri*. İstanbul: İstanbul Teknik Üniversitesi, ISBN 975-561-081-2.
- Korurek, M. & Dogan, B.** (2010). ECG beat classification using particle swarm optimization and radial basis function neural network. *Expert Systems with Applications*, 37 (12), 7563–7569.
- Korurek, M. & Nizam, A.** (2008). A new arrhythmia clustering technique based on ant colony optimization. *Journal of Biomedical Informatics*. 41 (6), 874–881.
- Korurek, M. & Nizam, A.** (2010). Clustering MIT-BIH arrhythmias with ant colony optimization using time domain and PCA compressed wavelet coefficient. *Digital Signal Processing*, 20 (4), 1050–1060.
- Korurek, M., Yuksel, A., Dokur, Z. & Olmez, T.** (2010). Dimension reduction by a novel unified scheme using divergence analysis and genetic search. *Digital Signal Processing*, 20 (6), 1535–1546.
- Kovacs, F., Legany C. & Babos A.** (2005). Cluster validity measurement techniques. *6th International Symposium of Hungarian Researchers on Computational Intelligence*. Budapest, Hungary: November 18-19.
- Kuo, R. J., Wang, H. S., Hu, T.L. & Chou, S. H.** (2005). Application of ant K-means on clustering analysis in data mining. *International Journal of Computers and Mathematics with Applications*, 50 (10-12), 1709-1724.
- Li, Q., Rajagopalan, C. & Clifford, G.D.** (2014), A machine learning approach to multi-level ECG signal quality classification. *Computer Methods and Programs in Biomedicine*, 117 (3), 435-447.
- Lucic, P. & Teodorovic, D.** (2001). Bee system: modeling combinatorial optimization transportation engineering problems by swarm intelligence. In: *Preprints of the TRISTAN IV triennial symposium on transportation analysis*, (pp. 441–445). Sao Miguel, Azores Islands, Portugal: June, 13-19.
- Luz, E., Nunes, T., Albuquerque, V., Papa, J. & Menotti, D.** (2013). ECG arrhythmia classification based on optimum-path forest. *Expert Systems with Applications*, 40 (9), 3561-3573.
- Ma, M., Liang, J., Guo, M., Fan, Y. & Yin, Y.** (2011). SAR image segmentation based on Artificial Bee Colony algorithm. *Applied Soft Computing*, 11 (8), 5205-5214.

- Martis, R.J., Acharya, U.R., Mandana, K.M., Ray, A.K. & Chakraborty, C.** (2012). Application of principal component analysis to ECG signals for automated diagnosis of cardiac health. *Expert Systems with Applications*, 39 (14), 11792-11800.
- Mateo, J., Torres, A.M., Aparicio, A. & Santos, J.L.** (2016). An efficient method for ECG beat classification and correction of ectopic beats. *Computers and Electrical Engineering*, 53 (C), 219-229.
- McPhee, S. J. & Hammer, G. D.** (2010). *Pathophysiology of disease: An introduction to clinical medicine*. New York: McGraw-Hill Medical.
- Mishra, A.K. & Raghav, S.** (2010). Local fractal dimension based ECG arrhythmia classification. *Biomedical Signal Processing and Control*, 5 (2), 114–123.
- Nizam, A.** (2008). *Karınca koloni optimizasyonuna dayalı yeni bir aritmi sınıflama tekniği* (Doktora Tezi). İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Nozohour-leilabady, B. & Fazelabdolabadi, B.** (2016). On the application of artificial bee colony (ABC) algorithm for optimization of well placements in fractured reservoirs; efficiency comparison with the particle swarm optimization (PSO) methodology. *Petroleum*, 2 (1), 79-89.
- Omkar, S.N., Senthilnath, J., Khandelwal, R., Naik, G. N. & Gopalakrishnan, S.** (2011). Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures. *Applied Soft Computing*, 11 (1), 489-499.
- Osowski, S., Markiewicz, T., & Tran Hoai, L.** (2008). Recognition and classification system of arrhythmia using ensemble of neural networks. *Measurement*, 41 (6), 610-617.
- Osowski, S., Siwek, K. & Siroic, R.** (2011). Neural system for heartbeats recognition using genetically integrated ensemble of classifiers. *Computers in Biology and Medicine*, 41 (3), 173–180.
- Padmavathi, K. & Ramakrishna, K.S.** (2015). Classification of ECG signal during atrial fibrillation using autoregressive modelling. *Procedia Computer Science*, 46 (1), 53-59.
- Pan, J. & Tompkins, W.J.** (1985). A Real-Time QRS Detection Algorithm. *IEEE Transactions on Biomedical Engineering*, 32 (3), 230-236.
- Pasolli, E. & Melgani, F.** (2015). Genetic algorithm-based method for mitigating label noise issue in ECG signal classification. *Biomedical Signal Processing and Control*, 19 (5), 130-136.
- Rousi, E., Anagnostopoulou, C., Tolika, K., & Maheras, P.** (2015). Representing teleconnection patterns over Europe: A comparison of SOM and PCA methods, *Atmospheric Research*, 152 (1), 123-137.
- Samworth R. J.** (2012). Optimal weighted nearest neighbour classifiers. *Annals of Statistics*, 40 (5), 2733–2763.

- Sandoe, E. & Sigurd, B.** (1991). Arrhythmia – A guide to clinical electrocardiology. Mishawaka, IN, USA: Publishing Partners.
- Shadmand, S. & Mashoufi, B.** (2016). A new personalized ECG signal classification algorithm using block-based neural network and particle swarm optimization. *Biomedical Signal Processing and Control*, 25, 12-23.
- Sharan, R., Maron-Katz, A. & Shamir, R.** (2003). CLICK and EXPANDER: A System for Clustering and Visualizing Gene Expression Data. *Bioinformatics*, 19 (14), 1787-1799.
- Shin, D. & Kim, S.** (2009). Nearest mean classifier via-one class SVM. *Proceedings of the International Joint Conference on Computational Sciences and Optimization*, (pp. 593-596). Sanya, Hainan, China: April, 24-26.
- Shu, G., Zeng, B., Chen, Y.P. & Smith, O.H.** (2003). Performance assessment of kernel density clustering for gene expression profile data. *Comparative and Functional Genomics*, 4 (3), 287-299.
- Silipo, R., Bortolan, G. & Marchesi, C.** (1999). Design of hybrid architectures based on neural classifier and RBF pre-processing for ECG analysis. *Approximate Reasoning*, 21 (2), 177-196.
- Sternickel, K.** (2002). Automatic pattern recognition in ECG time series. *Computer Methods and Programs in Biomedicine*, 68 (2), 109–115.
- Szeto, W.Y., Wu, Y. & Ho, S.C.** (2012). An Artificial Bee Colony algorithm for the capacitated vehicle routing problem. *European Journal of Operational Research*, 215 (1), 126-135.
- Thomas, M., Das, M.K. & Ari, S.** (2015). Automatic ECG arrhythmia classification using dual tree complex wavelet based features. *International Journal of Electronics and Communications (AEU)*, 69 (4), 715-721.
- Tsai, C.F., Tsai, C.W., Wu, H.C. & Yang, T.** (2004). ACODF: A novel data clustering approach for data mining in large databases. *Journal of Systems and Software*, 73 (1), 133-145.
- Url-1** <<http://www.physionet.org/physiobank/database/html/mitdbdir/mitdbdir.htm>>, erişim tarihi: 25.02.2017.
- Url-2** <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.330.1578&rep=rep1&typ=pdf>>, erişim tarihi 13.08.2016.
- Url-3** <http://www.lohninger.com/helpsuite/kohonen_network_background_information.htm>, erişim tarihi 15.08.2016.
- Url-4** <https://en.wikipedia.org/wiki/Iris_flower_data_set>, erişim tarihi 17.08.2016.
- Url-5** <<http://archive.ics.uci.edu/ml/datasets/Iris?ref=datanews.io>>, erişim tarihi 17.08.2016.
- Url-6** <<http://physionet.org/physiobank/database/mitdb/>>, erişim tarihi 16.08.2016.
- Url-7** <<http://www.physionet.org/physiobank/database/html/mitdbdir/intro.htm>>, erişim tarihi 05.09.2015.
- Url-8** <<http://www.physionet.org>>, erişim tarihi 16.08.2016.

- Vafaie, M.H., Ataei, M. & Koofgar, H.R.** (2014). Heart diseases prediction based on ECG signals' classification using a genetic-fuzzy system and dynamical model of ECG signals. *Biomedical Signal Processing and Control*, 14, 291-296.
- Veenman, C.J. ve Reinders, M.J.T.** (2005). The nearest sub-class classifier: a compromise between the nearest mean and nearest neighbour classifier. *IEEE Transactions on PAMI*, 27 (9), 1417-1429.
- Wedde, H.F., Farooq, M. & Zhang, Y.** (2004). Beehive: An efficient fault-tolerant routing algorithm inspired by honey bee behaviour. In: Dorigo, M., Birattari, M., Blum, C., Gambardella, L.M., Mondada, F., Stützle, T. (eds.), *ANTS workshop. Lecture notes in computer science*, (pp. 83–94). Heidelberg, Berlin: Springer.
- Yan, X., Zhu, Y., Zou, W., & Wang, L.** (2012). A new approach for data clustering using hybrid artificial bee colony algorithm. *Neurocomputing*, 97, 241–250.
- Yang, X.S.** (2005). Engineering optimizations via nature-inspired virtual bee algorithms. In: Mira, J. & Alvarez J.R. (eds.), *First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC, Artificial intelligence and knowledge engineering applications: a bioinspired approach*, (pp. 317–323). Las Palmas, Canary Islands, Spain: June 15-18.
- Yao, B., Yang, C., Hu, J. & Yu, B.** (2010). The optimization of urban subway routes based on artificial bee colony algorithm. In: Chen F, Gao L, Bai Y (eds), *Key technologies of railway engineering - high speed railway, heavy haul railway and urban rail transit*, (pp. 747–751). Beijing, China: Beijing Jiaotong University, August, 31 – September, 2.
- Yazgan, E. ve Korürek, M.** (1996). *Tıp Elektroniği*. İstanbul: İstanbul Teknik Üniversitesi.
- Yeh, Y.C., Wang, W.J. & Chiou, C.W.** (2010). A novel fuzzy c-means method for classifying heartbeat cases from ECG signals. *Measurement*, 43 (10), 1542–1555.
- Yıldız, A.** (2011). *EEG ve EKG işaretlerinden örüntü tanıma uygulamaları ve karşılaştırılması*. (Doktora tezi). Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.
- Zhang, C., Ning, J. & Ouyang, D.** (2010). An artificial bee colony approach for clustering. *Expert Systems with Applications*, 37 (7), 4761–4767.
- Zhao, Y. & Karypis. G.** (2005). Data clustering in life sciences. *Molecular Biotechnology*, 31 (1), 55-80.
- Zidelmal, Z., Amirou, A., Ould-Abdeslam, D. & Merckle, J.** (2013). ECG beat classification using a cost sensitive classifier. *Computer Methods and Programs in Biomedicine*, 111 (3), 570-577.

EKLER

EK-A : MABC sınıflayıcı için kullanıcı arayüzü

EK-B : Sözlük

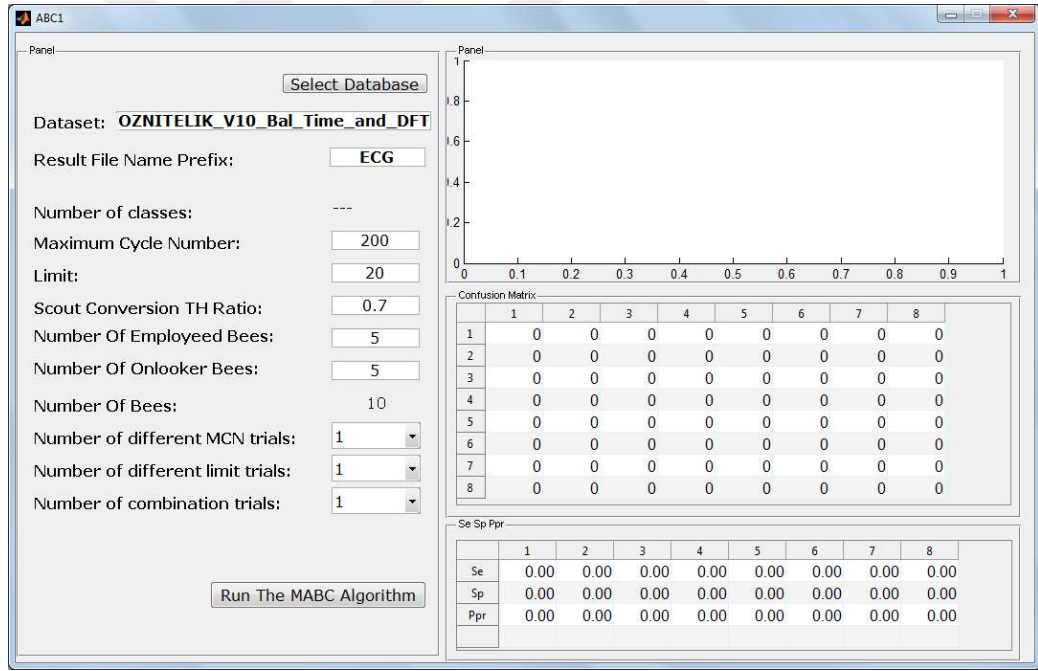




EK-A: MABC sınıflayıcı için kullanıcı arayüzü

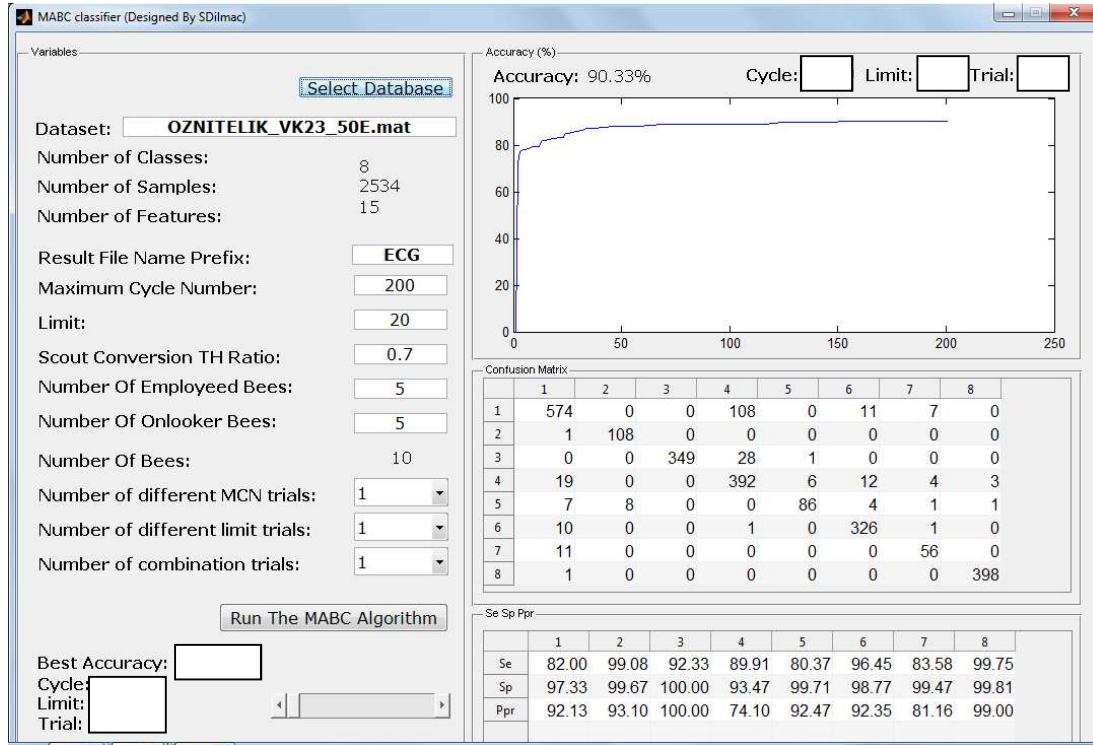
Bu çalışmada geliştirilmiş olan MABC tabanlı sınıflayıcının, değişik kullanıcılar tarafından çeşitli veri kümeleri üzerinde kolaylıkla kullanılabilmesini sağlamak amacıyla, tez çalışması kapsamında bir grafik kullanıcı ara yüzü (Graphical User Interface – GUI) hazırlanmıştır. MABC tabanlı sınıflayıcı için hazırlanan kullanıcı arayüzünün başlangıç görüntüsü Şekil A.1’de görülmektedir.

Bu kullanıcı arayüzünde, üzerinde sınıflama yapılacak veri kümesinin bulunduğu dosya bilgisayar hafızasından seçilebilir. Kullanılacak olan örneklere ait öznitelik değerleri ve sınıf bilgileri bir matris şeklinde hazırlanmış olmalıdır.



Şekil A.1 : MABC tabanlı sınıflayıcı için hazırlanan kullanıcı arayüzünün başlangıç görüntüsü.

Algoritmanın kontrol parametreleri olan maksimum yineleme sayısı (MCN), limit, izci arı dönüşüm eşiği (SCTR), izci arı ve gözcü arıların sayısı kullanıcı tarafından seçilebilir. Belirlenen kontrol parametreleri ile MABC algoritmasının bir deneme veri kümesi üzerinde bir defa çalıştırılması halinde elde edilen sonuçlar Şekil A.2’de görülmektedir.



Şekil A.2 : Belirlenen kontrol parametreleri ile MABC algoritmasının bir defa çalıştırılması halinde elde edilen sonuçlar.

Programın çalışması devam ederken her bir yineleme adımında elde edilen sınıflama başarımları grafik olarak çizdirilmektedir. Algoritmanın arama süreci tamamlandığında ise, o veri kümesi için elde edilen sınıflama matrisi (confusion matrix) ve her bir sınıf bazında sınıflama performans değerleri matrisi (Se Sp Ppr) ekranda görüntülenir. Şekil A.2’de 8 farklı sınıfa ait verilerin üzerinde çalışan programın elde etmiş olduğu sonuçlar görülmektedir.

MABC algoritmasında rasgele arama süreci mevcuttur. Bu nedenle çok büyük sapmalar olmasa da her çalıştırmada farklı sonuçlar vermektedir. Algoritmanın birden fazla çalıştırılıp, sonuçların istatistiksel olarak değerlendirilebilmesi için kullanıcı arayüzüne deneme sayısı (trial number) opsiyonu eklenmiştir. “Number of combination trials” opsiyonu, seçilmiş olan MCN ve Limit değerlerinin kombinasyonu ile kaç defa denemenin gerçekleştirileceğinin seçilmesini sağlar.

Buna ilaveten, kontrol parametreleri MCN ve Limit için, kaçar farklı değer olarak algoritmanın çalıştırılacağı da seçilebilmektedir. Örneğin, “MCN=200” ve “Number of different MCN trials=5” seçilmesi halinde, 200’ün katları olarak MCN=200, 400, 600, 800 ve 1000 değerleri için algoritma çalışacaktır. Benzer şekilde, “Limit=20” ve

“Number of different limit trials=3” seçilmesi halinde, 20’nin katları olarak Limit=20, 40 ve 60 değerleri için algoritma çalışacaktır.

Bir örnek üzerinde göstermek üzere,

MCN=200

Limit=20

Number of different MCN trials=5

Number of different limit trials=3

Number of combination trials=4

şeklinde bir seçim yapılması halinde,

MCN=200, Limit=20 için 4 deneme

MCN=200, Limit=40 için 4 deneme

MCN=200, Limit=60 için 4 deneme

MCN=400, Limit=20 için 4 deneme

MCN=400, Limit=40 için 4 deneme

...

MCN=1000, Limit=40 için 4 deneme

MCN=1000, Limit=60 için 4 deneme

olmak üzere algoritma toplam 60 defa çalışacaktır. Deneme sayıları için, her üç parametre de 5 olarak seçilirse, toplam 125 deneme yapılmış olur. Elde edilen sonuçlar, kullanıcının ön-ekini (“Result File Name Prefix”) belirleyebildiği bir isme sahip olan sonuç dosyasına yazılmaktadır. Bu dosya bir Microsoft Excel dosyası olup, her sayfasında Limit parametresinin almış olduğu değere karşı yapılan tüm denemelerde elde edilen sonuçların detayları bulunmaktadır. Sonuç dosyasında, çalışma sayfasında yer alan sonuç verilerinin yapısı Şekil A.3’te görülmektedir.

Şekil A.4 : Sonuç dosyasında “BasariYuzdeleri” sayfası.

Geliştirilen bu kullanıcı arayüzü ile, MABC algoritmasının çeşitli uygulama alanlarına ait veri kümeleri üzerinde kolaylıkla çalıştırılması mümkün olmaktadır.



EK-B: SÖZLÜK

Bölümleme	: Partitioning
Çözünürlük	: Resolution
Denetimli	: Supervised
Denetimsiz	: Unsupervised
Doğadan esinlenen	: Nature inspired
Duyarlık	: Sensitivity
İşçi arı	: Employed bee
İyileştirilmiş ABC	: Modified ABC
Gözcü arı	: Onlooker bee
Karınca Kolonisi Optimizasyonu	: Ant Colony Optimization (ACO)
Kâşif arı	: Scout bee
K-Katlı çapraz doğrulama	: K-Fold cross validation
Öbekleme	: Clustering
Örtüşme	: Overlap
Özgüllük	: Specificity
Parçacık Sürü Optimizasyonu	: Particle Swarm Optimization (PSO)
Pozitif öngörü	: Positive predictivity
Sınıflama	: Classification
Sınıflama başarımı	: Classification accuracy
Sürü zekâsı	: Swarm Intelligence (SI)
Yineleme	: Iteration



ÖZGEÇMİŞ



Ad-Soyad : Selim Dilmaç
Doğum Tarihi ve Yeri : 05 / 10 / 1967, Kütahya
E-posta : sdilmac@arroweurope.com

ÖĞRENİM DURUMU:

- **Lisans** : 1988, İstanbul Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü
- **Yüksek lisans** : 1992, İstanbul Teknik Üniversitesi, Elektronik ve Haberleşme Anabilim Dalı, Elektronik Mühendisliği Programı

MESLEKİ DENEYİM VE ÖDÜLLER:

- 1988-1993 yılları arasında AES Elektronik San. ve Tic. Firmasında elektronik tasarım mühendisi olarak çalıştı. Otuzdan fazla çeşitli otomasyon, güvenlik ve bilgilendirme cihazlarının tasarımını gerçekleştirdi.
- 1994 yılında Ekinciler Holding, EKDATA A.Ş.'de tasarım mühendisi olarak çalıştı.
- 1994-1999 yılları arasında İBB, BELBİM A.Ş.'de İstanbul Büyük Şehir Belediyesi bünyesinde hizmet veren toplu taşıma araçlarında kullanılan elektronik bilet sistemi Akbil'in ilk tasarım mühendisi ve proje yöneticisi olarak görev yaptı. Otobüs, metro, deniz otobüsü gibi tüm toplu taşıma araçları genelinde kullanılan ödeme sisteminin alt yapısını kurma konusundaki başarıları nedeniyle ödüllendirildi.
- 1999-2000 yıllarında askerlik görevini yedek subay asteğmen olarak gerçekleştirdi.
- 2001-2004 yılları arasında ELEKTRO A.Ş.'de saha uygulama mühendisi olarak görev yaptı.
- 2005 Şubat ayından itibaren, ARROW Electronics firmasında İş Geliştirme Müdürü olarak görev yapmaya devam ediyor. Bu süre zarfında iki defa NXP Semiconductor, birer defa Analog Devices, Texas Instruments, Telit ve Recom firmaları tarafından Doğu Avrupa bölgesinde en başarılı iş geliştirme uzmanı olarak ödüllendirilmiştir. Ayrıca 2016 yılında NXP Semiconductor tarafından Avrupada RFID konusunda en başarılı iş geliştirme uzmanı olarak ödüllendirilmiştir.

DOKTORA TEZİNDEN TÜRETİLEN YAYINLAR, SUNUMLAR VE PATENTLER:

- **Dilmac, S. & Olmez, T. (2017).** Nature inspired algorithm MABC for clustering and classification of ECG heart beats, using time and frequency domain features. *Proceedings of the 10th International Conference on Electrical and Electronics Engineering*. Bursa, Turkey: November 30 – December 2.
- **Dilmac, S. & Korurek, M. (2015).** ECG heart beat classification method based on modified ABC algorithm, *Applied Soft Computing*, 36 (C), 641–655. <http://dx.doi.org/10.1016/j.asoc.2015.07.010>
- **Dilmaç, S. & Korürek, M. (2014).** ABC algoritmasının sınıflamada kullanımı ve kontrol parametrelerinin etkilerinin incelenmesi, *BIYOMUT 2014*, 18. *Biyomedikal Mühendisliği Ulusal Toplantısı*, İstanbul, Türkiye: Ekim 16-17.
- **Dilmac, S., Nizam, A. & Korurek, M. (2014).** Evaluation of a new heart beat classification method based on ABC algorithm, comparison with GA, PSO and ACO classifiers. *International Journal of Reasoning-based Intelligent Systems*, 3/4 (6), 98–108.
- **Dilmac, S. & Korurek, M. (2013a).** A new ECG arrhythmia classification method by using modified ABC (MABC) algorithm. *Proceedings of the Fourth International Conference on Mathematical and Computational Applications*, (pp.390-398). Manisa, Turkey: June 11-13.
- **Dilmac, S., Nizam, A. & Korurek, M. (2013b).** Modified artificial bee colony algorithm based ECG beat type classification method, comparison with ACO based classifier. *Proceedings of the First International Symposium on Innovative Technologies in Engineering and Science*, (pp.1157-1166). Sakarya, Turkey: June 7-9.
- **Dilmac, S. & Korurek, M. (2013c).** A new ECG arrhythmia clustering method based on modified artificial bee colony algorithm, comparison with GA and PSO classifiers. *International Symposium on Innovations in Intelligent Systems and Applications (INISTA), IEEE International Symposium* (pp. 1-5). Albena, Bulgaria: June 19-21. doi: 10.1109/INISTA.2013.6577616

DOKTORA TEZİNDEN TÜRETİLEN YAYINLARA YAPILAN ATIFLAR

“2” nolu yayına yazarların haricinde 17 Temmuz 2017 itibariyle 7 adet atıf yapılmıştır.

“4” nolu yayına yazarların haricinde 17 Temmuz 2017 itibariyle 1 adet atıf yapılmıştır.

“7” nolu yayına yazarların haricinde 17 Temmuz 2017 itibariyle 3 adet atıf yapılmıştır.