



Parçacık Sürü Optimizasyonu ile Yapay Sinir Ağları Eğitimine Dayalı Bir Esnek Hesaplama Uygulaması

Kerim Kürşat ÇEVİK^{1*}, Hasan Erdinç KOÇER²

¹Niğde Üniversitesi, Bor Meslek Yüksek Okulu, Bilgisayar Teknolojileri Bölümü, 51700, Niğde

²Selçuk Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümü, 42003, Konya

(Alınış Tarihi: 14.06.2013, Kabul Tarihi: 21.08.2013)

Anahtar Kelimeler

Esnek hesaplama
Parçacık sürü optimizasyonu
Yapay sinir ağları
Meme kanseri

Özet: Bu çalışmada, meme kanseri hastalığının esnek hesaplama tabanlı teşhisini gerçekleştiren yazılım sunulmuştur. Yazılım, Parçacık Sürüsü Optimizasyonu (PSO) ile Yapay Sinir Ağları (YSA) tekniklerinin bir arada uygulamaktadır. Sınıflandırma amacıyla kullanılan YSA'nın öğrenme algoritmasında PSO kullanılmıştır. Meme kanseri veri kümesi UCI (University of California, Irvine) veritabanından alınmıştır. Çalışmamızda aldığımız deneysel sonuçlara göre PSO ile öğrenme gerçekleştiren YSA tekniği hız açısından daha iyi performans göstermektedir.

A Soft Computing Application Based On Artificial Neural Networks Training by Particle Swarm Optimization

Keywords

Soft computing
Particle swarm optimization
Artificial neural Networks
Breast cancer

Abstract: A software conducting the soft computing based diagnosis of breast cancer was presented in this study. This software applies Particle Swarm Optimization (PSO) and Artificial Neural Networks (ANN) techniques together. PSO was used in the learning algorithm of ANN which was used for classification. Breast cancer data set was taken from UCI (University of California, Irvine)'s database. According to the experimental results of our study, the ANN technique which realizes learning through PSO shows a better performance.

1. Giriş

Son yıllarda yapılan çalışmalar göstermektedir ki, çağımızın vebası olarak tanımlanan kanserin, erken teşhis edildiğinde tedavi edilme olasılığı yüksektir. Özellikle 35-65 yaşları arasında olan kadınların birçoğunda ölüme sebep olan meme kanserinin teşhisinde tıp uzmanına yardımcı olacak bilgisayar destekli karar verme yöntemleri sıkça kullanılmaktadır. Bu çalışmada esnek hesaplama tekniği kullanarak meme kanseri ile ilgili tanı koyabilen bir yazılım uygulaması gerçekleştirilmiştir.

Esnek hesaplama, özellikle klasik hesaplama teknikleri ile çözümü mümkün olmayan karmaşık yapıdaki çok parametrelili problemlerin analitik modellenmesini gerçekleştirebilen bir tekniktir. Aslında esnek hesaplamada standart bir hesaplama yapılmaz. İki veya daha fazla hesaplama tekniğini bir arada kullanan esnek hesaplama, karmaşık sorunlara karmaşık olmayan çözümler üretebilmektedir. Bu çalışmada yapay zekânın tanıma-karar verme konusunda güçlü bir yapıya sahip olan Yapay Sinir Ağları (YSA) tekniği kullanılarak meme kanseri

verileri üzerinde tanı konulmuştur. YSA'nın sınıflandırmada önemli bir noktası olan öğrenme aşamasında sezgisel bir algoritma olan Parçacık Sürü Optimizasyonu (PSO) kullanılarak sınıflandırmanın başarısı daha da yukarı çekilmiştir.

Çalışmada ilk olarak YSA ve PSO yöntemleri hakkında bilgiler verilmiştir. Daha sonraki kısımda PSO'nun Yapay Sinir Ağının eğitiminde nasıl kullanıldığı anlatılmıştır. Üçüncü bölümde hastalardan alınan meme hücre verileri üzerinde belirtilen yöntemlerle yapılan deney sonuçları raporlanmış ve son bölümde de elde edilen bulgular değerlendirilmiştir.

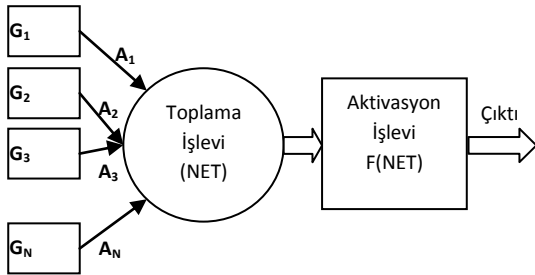
2. Materyal Metot

2.1. Yapay Sinir Ağları (YSA)

Yapay zekâ (YZ), bir bilgisayarın ya da bilgisayar denetimli bir makinenin, genellikle insana özgü nitelikler olduğu varsayılan akıl yürütme, anlam çıkartma, genelleme ve geçmiş deneyimlerden öğrenme gibi yüksek zihinsel süreçlere ilişkin

görevleri yerine getirme yeteneği olarak tanımlanmaktadır (Nabiyev, 2005).

Sınıflandırma problemlerinde YZ tekniklerinden genel olarak YSA kullanılır. Yapay Sinir Ağları, insan beyninin işleyişini taklit ederek yeni sistem oluşturmaya çalışan yaklaşımlardır (Öz vd., 2002). Beynimizdeki biyolojik sinir hücrelerinin yapısı temel alınarak oluşturulan YSA'da aynen beynimizde olduğu gibi öğrenme ve öğrenilen bilgilere göre karar verme mekanizmaları bulunur (Allahverdi, 2002; Öztemel, 2003). YSA, birbirine hiyerarşik olarak bağlı, Şekil 2'de gösterildiği gibi yapay sinir hücrelerden (nöron) meydana gelmektedir. Yapay sinir hücreleri proses elemanı olarak adlandırılmaktadır (Öztemel, 2003) ve her proses elemanının Şekil 1' deki gibi 5 farklı birimi vardır.



Şekil 1. Bir Yapay Sinir Hücresi örneği

$G_1, G_2, G_3, \dots, G_N$ ile gösterilen Girdiler bir YSA' nın girişleri olarak bilinir. $A_1, A_2, A_3, \dots, A_N$ Ağırlıklar olarak tanımlanır ve yapay sinir hücresine gelen bilginin etkisini gösterir. Toplama İşlevi Fonksiyonu (NET) bir sinir hücresine gelen net bilgiyi hesaplar. Bu net değeri bulmak için ağ yapısına göre maksimum, minimum ya da çarpım fonksiyonu gibi değişik fonksiyonlardan yararlanılmakla birlikte en çok kullanılan Denklem (1)' de gösterilen toplam ağırlığı bulan ifadedir. Burada G_i i. giriş değerini, A_i ise bu giriş değerinin ağırlığını ve NET ise fonksiyonun toplam değerini gösterir.

$$NET = \sum_{i=1}^n G_i A_i \quad (1)$$

Yapay sinir hücresindeki Aktivasyon İşlevi Fonksiyonu (FNET) hücreye gelen net girdileri hesaplayarak üretilecek olan çıktı değerini belirler. Aktivasyon fonksiyonlarının lineer fonksiyon, step fonksiyonu, sinüs fonksiyonu, eşik değer fonksiyonu, sigmoid fonksiyonu ve hiperbolik tanjant fonksiyonu olmak üzere değişik gösterimleri mevcuttur. Uygulamaların çoğu Çok Katmanlı Algılayıcı biçiminde tasarlandığı için çalışmalarda genel olarak Sigmoid Fonksiyonu tercih edilmektedir (Öztemel, 2003). Bu fonksiyon aşağıdaki formülle gösterilir. Bu çalışmada geliştirilen yazılım da, problemlerin çözümünde Sigmoid fonksiyonundan yararlanacak biçimde tasarlanmıştır. Sigmoid fonksiyonu Denklem (2) ile gösterilmiştir.

$$F(NET) = \frac{1}{1+e^{-NET}} \quad (2)$$

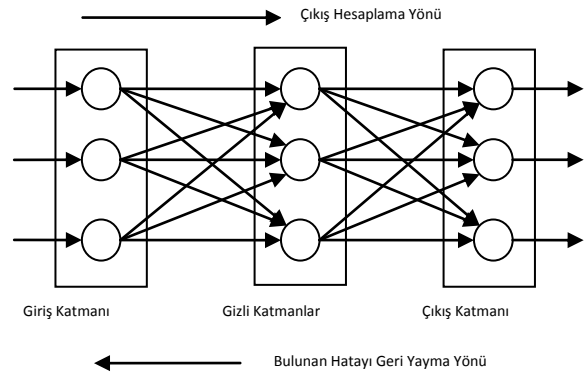
Proses elemanındaki Çıktı değeri de aktivasyon fonksiyonu tarafından belirlenen çıktı değeridir. Üretilen çıktı ya başka hücreye ya da kendisine tekrar gönderilerek değerlendirilir. Temel olarak bir YSA' nın görevi, kendisine verilen giriş setine karşılık bir çıkış seti belirlemektir. Bunu gerçekleştirebilmek için ağ, ilgili problemin örnekleri ile eğitilerek (öğrenme), o problemle ilgili istenenleri çözebilme yeteneğine kavuşturulur (Kaastra ve Boyd, 1995).

Bir YSA' nın genel ağ yapısı Şekil 2.' de gösterilmiştir. Yapay sinir ağı bir dizi sinir hücresinin ileri sürümlü ve geri beslemeli bağlantı şekilleri ile birbirine bağlanması ile oluşur (Uğur ve Kınacı, 2006; Haykin, 1999). YSA' da sinir hücrelerinin aynı doğrultuda bir araya gelmeleriyle YSA' nın katmanları meydana gelmektedir. Genel olarak yapay sinir ağları 3 katman halindedir ve bu katmanlar sırasıyla şöyledir:

Giriş Katmanı: Bu katmandaki hücreler giriş bilgilerini gizli katmana ulaştırmakla görevlidir.

Gizli Katman: Giriş katmanından gelen bilgiler işlenerek çıkış katmanına gönderilir. Bir ağda problemin durumuna göre, birden fazla gizli katman olabilmektedir.

Çıkış Katmanı: Bu katmandaki hücreler gizli katmandan gelen bilgiyi işleyerek problemin çözümüne yönelik çıktıları oluştururlar (Uğur ve Kınacı, 2006).



Şekil 2. YSA' nın genel ağ yapısı

YSA' da hücre eleman bağlantılarının ağırlık değerlerinin belirlenmesi işleme ağı eğitilmesi denir. Başlangıçta bu değerler rastgele belirlenir. Ancak daha sonra çıktı değerlerine göre bu ağırlık değerleri tekrar tekrar değiştirilerek gerçek durumuna erişir (Haykin, 2009). Eğitim verisinin tamamlanmasından sonra eğitilmiş olan ağ, ağırlık değerlerinin son durumuna göre, verilen herhangi bir veri setinin sonucunu tahmin edebilmektedir. (Uğur ve Kınacı, 2006; Haykin, 1999). Buna da ağı eğitilmesi denir. Ağı eğitilmesine referans olarak gösterilen farklı öğrenme modelleri vardır (Haykin, 2009). Bu çalışmada ağı eğitilmesi işleminde PSO algoritması kullanılmıştır.

2.2. Parçacık Sürüsü Optimizasyonu

1995 yılında J.Kennedy ve R.C. Eberhart tarafından; kuş sürülerinin davranışlarından esinlenilerek ortaya çıkarılmış popülasyon tabanlı optimizasyon tekniğidir (Kennedy ve Eberhart, 1995). Doğrusal olmayan problemlerin çözümü için tasarlanmıştır. Çok parametrelili ve çok değişkenli optimizasyon problemlerine çözüm bulmak için kullanılmaktadır (Xiaohui, 2006).

PSO, genetik algoritmalar gibi evrimsel hesaplama teknikleriyle birçok benzerlik gösterir. Sistem rastgele çözümler içeren bir popülasyonla başlatılır ve nesilleri güncelleyerek en optimum çözümü araştırır. PSO da parçacık olarak adlandırılan olası muhtemel çözümler, o andaki optimum parçacığı izleyerek problem uzayında dolaşırlar. PSO' nun klasik optimizasyon tekniklerinden en önemli farklılığı türev bilgisine ihtiyaç duymamasıdır. Bu özellik birçok problemin çözümü için gerekli olan karmaşık işlem yükünün hafifletilmesini sağlamaktadır. Ayrıca PSO' nu uygulamak, algoritmasında ayarlanması gereken parametre sayısının az olması sebebiyle oldukça basittir. PSO; fonksiyon optimizasyonu, bulanık sistem kontrolü, yapay sinir ağı eğitimi gibi birçok alanda başarıyla uygulanabilmektedir (Zhao vd., 2005; Juang ve Lu, 2006; Ghoshal, 2004; Awad, 2006).

PSO kuş sürülerinin davranışlarının bir benzetimidir. Kuşların uzayda, yerini bilmedikleri yiyeceği aramaları, bir probleme çözüm aramaya benzetilir. Kuşlar yiyecek ararken yiyeceğe en yakın olan kuşu takip ederler. Parçacık olarak adlandırılan her tekil çözüm, arama uzayındaki bir kuştur. Parçacık hareket ettiğinde, kendi koordinatlarını bir fonksiyona gönderir ve böylece parçacığın uygunluk değeri ölçülmüş olur. (Yani yiyeceğe ne kadar uzaklıkta olduğu ölçülmüş olur.) Bir parçacık, koordinatlarını, hızını (çözüm uzayındaki her boyutta ne kadar hızla ilerlediği), şimdiye kadar elde ettiği en iyi uygunluk değerini ve bu değeri elde ettiği koordinatları hatırlamalıdır. Parçacığın çözüm uzayının her boyutundaki hızının ve yönünün her seferinde nasıl değişeceği, komşularının en iyi koordinatları ve kendi kişisel en iyi koordinatlarının birleşimi ile doğru orantılıdır. Bireyler kendi tecrübelerine ve komşularının tecrübelerine göre hareket edecektir. Yani bireyler arasında bilgi paylaşımı sağlanacaktır.

PSO' da Problemin çözüm uzayı, değişken veya bilinmeyen sayısına bağlı olarak çok boyutta olabilir. Örneğin; $5x^2 + 2y^3 - (z/w)^2 + 4$ Fonksiyonunun çözüm uzayı x, y, z ve w bilinmeyenlerinden dolayı 4 boyutludur. Bu problemin çözüm uzayında tanımlanan bir parçacığın pozisyonu 4 koordinat ile $P=[x, y, z, w]$ şeklinde belirtilmektedir.

Örneğin yukarıdaki fonksiyonu sıfıra eşitleyerek bir çözüm bulunabilir.

$P=[-1, 0, 3, 1]$ şeklindeki parçacık için fonksiyon; $5x^2 + 2y^3 - (z/w)^2 + 4 = 0$ Veya $P=[3, 3, 8, 1]$ şeklindeki bir parçacık; $x=3, y=3, z=8$ ve $w=1$ koordinatları için bir uygunluk fonksiyonunu temsil eder. Görsel olarak insanların resmedemediği 4 veya daha fazla boyutlu karmaşık problemlerde çalışmanın PSO için herhangi bir zorluğu bulunmamaktadır.

PSO, bir grup rastgele çözümlerle (parçacık sürüsü) başlatılır ve güncellemelerle optimum çözüm bulunmaya çalışılır. Her tekrarlamada (iterasyonda), parçacık konumları, iki en iyi değere göre güncellenir. İlki; o ana kadar parçacığın elde ettiği en iyi çözümü sağlayan koordinatlarıdır. Bu değer "pbest" olarak adlandırılır ve hafızada saklanmalıdır. Diğer en iyi değer ise, popülasyonda o ana kadar tüm parçacıklar tarafından elde edilen en iyi çözümü sağlayan koordinatlarıdır. Bu değer global en iyidir ve "gbest" ile gösterilir. Örneğin D adet parametreden oluşan n adet parçacık olduğunu varsayalım. Bu durumda popülasyon parçacık matrisi eşitlik Denklem (3)' deki gibidir.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & \dots & x_{1D} \\ x_{21} & x_{22} & \dots & \dots & x_{2D} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & \dots & x_{nD} \end{bmatrix} \quad (3)$$

Parçacık sürü matrisindeki i' ninci parçacık aşağıdaki gibi ifade edilir.

$$x_i = [x_{i1}, x_{i2}, \dots, x_{iD}] \quad (4)$$

Önceki en iyi uygunluk değerini veren i' ninci parçacığın pozisyonu (pbest_i);

$$pbest_i = [p_{i1}, p_{i2}, \dots, p_{iD}] \quad (5)$$

şeklinde ifade edilir. gbest ise her iterasyonda tüm parçacıklar için tektir ve

$$gbest_i = [p_1, p_2, \dots, p_D] \quad (6)$$

şeklinde gösterilir. i' ninci parçacığın hızı (her boyuttaki konumunun değişim miktarı)

$$v_i = [v_{i1}, v_{i2}, \dots, v_{iD}] \quad (7)$$

olarak ifade edilir. İki en iyi değer bulunmasından sonra parçacık hızları ve konumları aşağıda verilen denklemlere göre güncellenir.

$$v_i^{k+1} = v_i^k + c_1 \cdot rand_1^k \cdot (pbest_i^k - x_i^k) + c_2 \cdot rand_2^k \cdot (gbest^k - x_i^k) \quad (8)$$

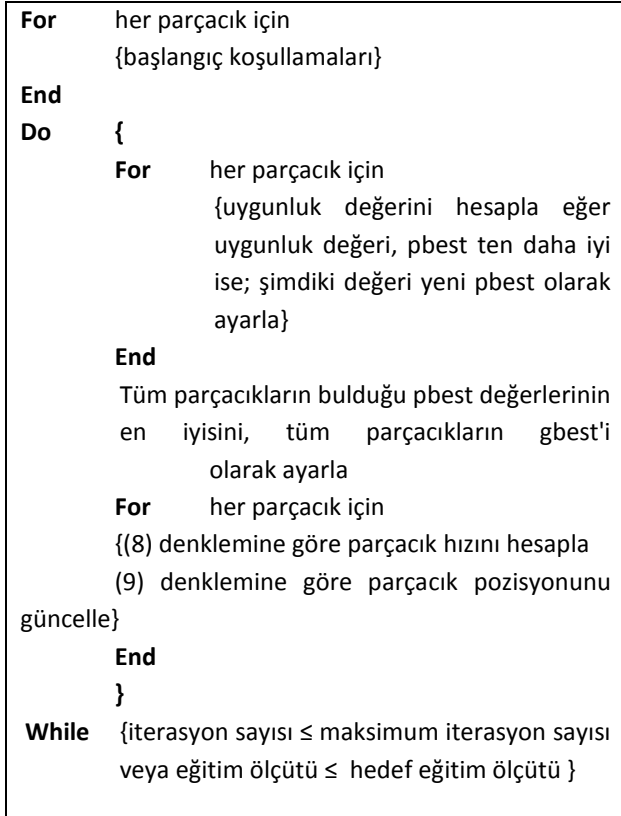
$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (9)$$

Denklem (8)'deki c_1 ve c_2 öğrenme faktörleridir. c_1 ve c_2 , her parçacığı pbest ve gbest pozisyonlarına doğru çeken, hızlanma terimlerini ifade eden sabitlerdir. c_1 ,

parçacığın kendi tecrübelerine göre hareket etmesini, c_2 ise sürüdeki diğer parçacıkların tecrübelerine göre hareket etmesini sağlar. Denklemdaki $rand_1$ ve $rand_2$, $[0,1]$ arasında düzgün dağılımlı rastgele sayılardır. k iterasyon sayısını, i ise parçacık indisini belirtmektedir.

c_1 ve c_2 için düşük değerler seçilmesi parçacıkların hedef bölgeye doğru çekilmeden önce, bu bölgeden uzak yerlerde dolaşmalarına imkân verir. Ancak hedefe ulaşma süresi uzayabilir. Diğer yandan, bunlar için yüksek değerler seçilmesi, hedefe ulaşmayı hızlandırırken, beklenmedik hareketlerin oluşmasına ve hedef bölgenin es geçilmesine sebep olabilir. Bu algoritma üzerinde araştırmacıların yaptığı denemelerde $c_1=c_2= 2$ olarak alınan iyi sonuçlar verdiği belirtilmiştir (Shi ve Eberhart, 1998; Li-ping vd., 2005; Xiaohui, 2006).

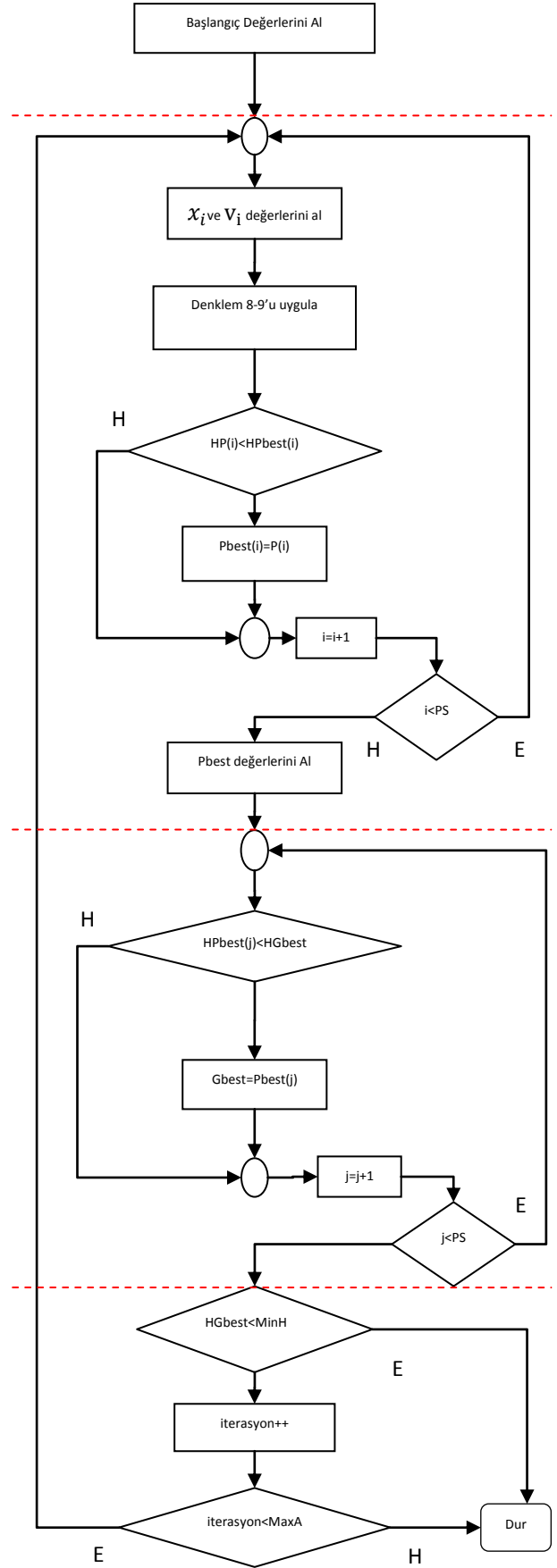
PSO algoritması için gerekli olan prosedür Şekil 3' de özetlenmiştir (Çavuşlu vd., 2010).



Şekil 3. PSO algoritmasının kod yapısı

2.3. PSO İle Yapay Sinir Ağı Eğitimi

Bu çalışmada YSA' nın eğitiminde ağ parametrelerinin belirlenmesi işlemi PSO algoritması ile gerçekleştirilmiştir. Bu algoritma Şekil 4'de görüldüğü gibi dört bölümden oluşmaktadır.



Şekil 4. PSO algoritmasının kod yapısı

Algoritmanın ilk aşamasında PSO ve YSA parametrelerinin ilk değerleri atanmakta ve algoritmanın çalışması için gerekli değişkenler oluşturulmaktadır.

İkinci aşamada algoritmaya belirtilen parçacık sayısı kadar bir döngü açılmakta ve bu parçacıklar için Denklem (8) ve Denklem (9) değerleri hesaplanmaktadır. Ayrıca bu aşamada her bir parçacığın en iyi değeri olan Pbest değerleri, eğer bulunan değerden daha yüksek ise; bulunan P değeri ile değiştirilmektedir.

Algoritma üzerinde görülen HP, HPbest ve HGbest değerleri algoritmanın geçerli adımındaki yerel P, Pbest ve Gbest değerleridir. PS değeri algoritmadaki parçacık sayısıdır. MaxA değeri algoritmanın çalışabileceği maximum adımı belirtmektedir.

Algoritmanın üçüncü aşamasında sistemin genel en iyi çözümü olan Gbest değeri elde bulunan HPbest değerleri ile karşılaştırılmakta ve eğer Gbest' den daha iyi bir Pbest değeri var ise Gbest değiştirilmektedir.

Algoritmanın son bölümünde ise; algoritmaya verilen minimum hata seviyesine ulaşıp ulaşılmadığına bakılmakta eğer istenilen hata seviyesine ulaşılmış ise program sonlandırılmaktadır. Daha sonra programdaki adım sayısı artırılmakta ve algoritmaya verilen maximum adım sayısına ulaşılmış ise program sonlandırılmaktadır. Eğer bu iki şart da sağlanmamış ise program hesaplamasına ikinci aşamadan itibaren devam etmektedir (Çavuşlu vd., 2010). Bu aşamalarda hata miktarı hesabı için YSA'nın ileri hesaplama yöntemi kullanılmıştır.

3. Deneysel Bulgular

Bu bölümde hastalardan alınan meme hücre verileri üzerinde esnek hesaplama tekniği kullanılarak yapılan deneysel çalışmalar anlatılmıştır. Çalışmamızda kullandığımız meme kanseri veri kümesi UCI (University of California, Irvine) veri tabanından alınmıştır (Wolberg, 1992).

Meme kanserinin iyi huylu veya kötü huylu olduğunun belirlenmesi için hasta insandan alınan hücrelerde bulunan dokuz tane özelliğe bakılmaktadır. YSA'nın girdi verileri için kullanılacak olan bu dokuz tane hücre özelliği; Küme Kalınlığı, Hücre Uniformity Boyutu, Hücre Uniformity Şekli, Marjinal Yapışma, Tek Epital Hücre Boyutu, Yalın Çekirdek, Bland Kromatin, Normal Nükleol ve Mitoz'dur. Bu dokuz tip girdi verisine göre elde edilebilecek sınıflandırma çıkışı "2" veya "4" sayısı olarak seçilmiştir. Eğer sonuç "2" ise iyi huylu kanser hücresi, sonuç "4" ise kötü huylu kanser hücresi olmaktadır. Bu özelliklere ait değerler ve sınıflandırma bilgileri Tablo 1'de gösterilmiştir (Wolberg, 1992).

Tablo 1. Meme kanseri veri seti

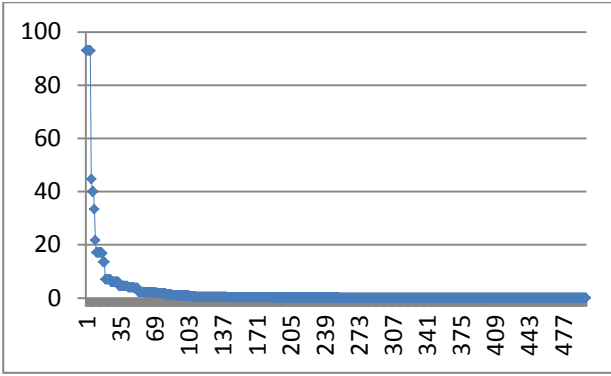
Sınıf (2 İyi Huylu, 4 Kötü Huylu)	Mitoz	Normal Nükleol	Bland Kromatin	Yalın Çekirdek	Tek Epital Hücre Boyutu	Marjinal Yapışma	Hücre Uniformity	Hücre Uniformity	Küme Kalınlığı
5	1	1	1	2	1	3	1	1	2
5	3	3	3	2	3	4	4	1	4
3	1	1	1	2	2	3	1	1	2
7	4	6	4	6	1	4	3	1	4
4	1	1	3	2	1	3	1	1	2
8	10	10	8	7	10	9	7	1	4
1	1	1	1	2	10	3	1	1	2
2	1	2	1	2	1	3	1	1	2
10	7	7	6	4	10	4	1	2	4
4	2	1	1	2	1	2	1	1	2

Tablo 1'de belirtilen, meme hücrelerine ait veri kümesi YSA'ya girdi olarak verilmiştir. Kullanılan YSA çok katmanlı mimariye sahip olup her denemede 500 iterasyon çalıştırılmıştır. YSA eğitimi yapılırken PSO ile ağırlıkların güncellenmesi işlemleri gerçekleştirilmiştir. PSO'da kullanılan parçacık sayısı ve hız aralığı yazılımda parametrik yapılarak istenildiği anda değiştirilmesine imkân sağlanmıştır. Benzer şekilde YSA'da da gizli katman nöron sayısı ve hata oranları yine parametrikdir. Eğitimin durdurulması kriteri olarak maksimum iterasyon sayısı veya hata oranı kullanılmıştır. Tüm girişler için bulunan çıkışlara göre ortalama karesel hata değerine bakılmıştır.

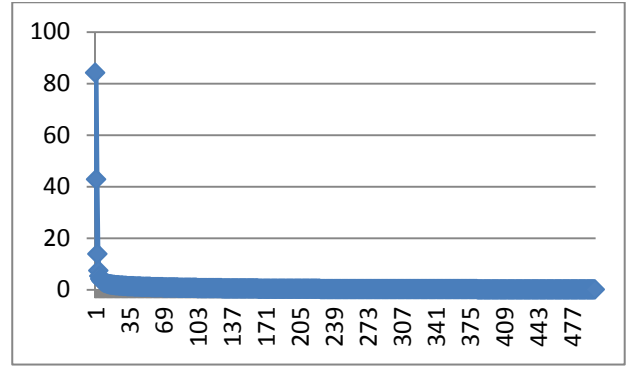
Elde edilen veriler ile YSA ve PSO parametreleri; Nöron Sayısı: 30, Parçacık Sayısı: 10, Hızlanma Aralığı: 0.1, Maksimum Adım Sayısı: 1000, İstenilen Hata Miktarı: 0.00001 ve elde bulunan 679 örnekten 350 tanesi eğitim aşamasında YSA'ya gösterilmiştir.

Program 350 adet eğitim verisi, iki farklı eğitim algoritması ile çalıştırılmıştır. İlkinde YSA'nın eğitimi backpropagation (BPG) (Geriye yayılım) algoritması ile gerçekleştirilmiştir (Çevik ve Dandil, 2012). İkinci aşamada ise YSA'nın eğitimi PSO ile yapılmış ve deneysel sonuçlar kaydedilmiştir.

BPG ile YSA eğitiminde 500 iterasyon sonunda genel hata miktarı 0,0726 seviyesine düşmüştür. PSO ile YSA eğitiminde ise 500 iterasyon sonunda genel hata miktarı 0,0003 seviyesine düşmüştür. Bu eğitimlerde alınan hata oranı-iterasyon grafiği Şekil 5 ve Şekil 6'da verilmiştir.



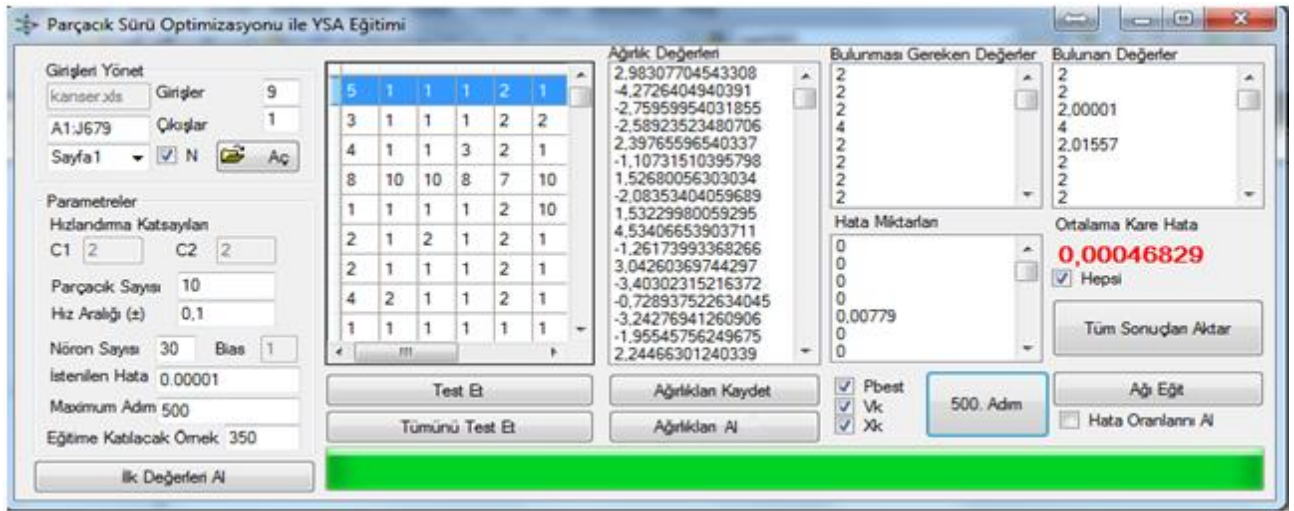
Şekil 5. PSO ile YSA Eğitimi Hata Oranı Değişimi



Şekil 6. BPG ile YSA Eğitimi Hata Oranı Değişimi

Tasarlanan program ile eğitim sonucunda bulunan ağırlık değerleri bir dosyaya kaydedilerek istendiğinde daha sonra tekrar yüklenmesi sağlanmıştır. Bulunan tüm deneysel sonuçlar bir

Excel dosyasına da aktarılabilmektedir. Programın eğitimi tamamladıktan sonraki genel görüntüsü Şekil 7'de gösterilmektedir.



Şekil 7. Programın eğitimi bitirdikten sonraki ekran görüntüsü

Eğitimlerin tamamlanmasının ardından test aşamasına geçilmiştir. Toplam 679 örneğin eğitimden sonra kalan 329 örnek test aşamasında kullanılmıştır. Elde edilen sonuçlar Tablo 2'de gösterildiği gibidir.

Tablo 2. Meme kanseri tespiti için elde edilen sonuçlar

	Adım	Süre	Hata	Eğitim (%)	Test (%)
YSA+BPG	500	07.45	0,0726	100	98,51
YSA+PSO	500	01.27	0,0003	100	99,40

Çalışmanın bir sonraki aşamasında PSO' da yer alan parçacık sayısı parametresi Tablo 3'teki gibi değiştirilmiştir.

Tablo 3. Parçacık sayısının değişimine göre elde edilen sonuçlar

Parçacık Sayısı	Hızlanma	Hata Miktarı	İterasyon	Süre
3	0,1	0,0017	500	01:35
6	0,1	0,0013	431	01:22
10	0,1	0,0011	253	01.10
20	0,1	0,0012	184	01.20

30	0,1	0,0011	139	01.18
40	0,1	0,0010	127	01.26
50	0,1	0,0011	105	01.30

4. Sonuç ve Öneriler

Bu çalışmada meme kanseri teşhisi için esnek hesaplama yaklaşımli yazılım sunulmuştur. Meme hücre verileri yapay sinir ağları ve parçacık sürü optimizasyonu teknikleri kullanılarak sınıflandırılmıştır. YSA'nın standart eğitim algoritması geriye yayılım (BPG) yerine PSO kullanılarak performans değerlendirilmesi yapılmıştır. Alınan deneysel bulgulara göre YSA'nın eğitim algoritması olarak PSO (YSA+PSO) kullanıldığında sınıflandırma başarısı arttığı gözlemlenmiştir. Ayrıca yine PSO kullanımı ile eğitim süresinin oldukça kısaldığı görülmüştür. Bununla birlikte PSO parametrelerinden parçacık sayısının eğitim süresi ve hata miktarları üzerinde kayda değer bir fark oluşturmadığı sonucuna varılmıştır.

Kaynaklar

Allahverdi, N., 2002. Uzman Sistemler Bir Yapay Zekâ Uygulaması. Atlas Yayın Dağıtım, İstanbul.

Awad, H., A., 2006. "A Novel Particle Swarm-Based Fuzzy Control Scheme". IEEE International Conference on Fuzzy Systems, July 16-21, Canada.

Çavuşlu, M., A., Karakuzu, C., Şahin, S. 2010. "Parçacık Sürü Optimizasyonu Algoritması ile Yapay Sinir Ağı Eğitiminin FPGA Üzerinde Donanımsal Gerçeklenmesi". Journal of Polytechnic, Vol: 13 No: 2 pp. 83-92.

Çevik, K., K., Dandıl, E. 2012. "Yapay Sinir Ağları İçin .Net Platformunda Görsel Bir Eğitim Yazılımının Geliştirilmesi". Gazi Üniversitesi Bilişim Teknolojileri Dergisi, Vol:5., No:1., pp. 19-28.

Ghoshal, S., P., 2004. "Optimizations of PID gains by particle swarm optimizations in fuzzy based automatic generation control", Electric Power Systems Research, Volume 72, Issue 3, pp: 203-212.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation. Pearson Prentice Hall Inc., New Jersey.

Haykin, S., 2009. Neural Networks and Learning Machines (Third Edition). Pearson Prentice Hall Inc., New Jersey.

Juang, C., F., Lu, C., F. 2006. "Load-frequency control by hybrid evolutionary fuzzy PI controller". IEE Proc.-Gener. Transm. Distrib, Vol. 153, No:2.

Kaastra, I., Boyd, M. 1995. "Designing a Neural Network for Forecasting Financial and Economic Time Series". Neurocomputing, 10(3), 215-236.

Kennedy, J., Eberhart, R., C., 1995. "Particle swarm optimization", Proc. IEEE int'l conf. on neural Networks, Vol: IV, p: 1942-1948. IEEE service center, Piscataway, NJ.

Li-ping, Z., Huan-jun Y., and Shang-xu, H. 2005. "Optimal choice of parameters for particle swarm optimization". Journal of Zhejiang University Science, Volume 6, Number 6 , pp 528-534.

Nabiyev, V., 2005. Yapay Zekâ Problemler, Yöntemler, Algoritmalar. Seçkin Yayıncılık, Ankara.

Öz, C., Köker R., Çakar, S., 2002. "Yapay Sinir Ağları ile Karakter Tabanlı Plaka Tanıma". Elektrik Elektronik ve Bilgisayar Mühendisliği Sempozyumu (ELECO'2002), Bursa, 1-6.

Öztemel, E., 2003. Yapay Sinir Ağları. Papatya Yayınevi, İstanbul.

Shi, Y., Eberhart, R., C., 1998. "Parameter selection in particle swarm optimization". Evolutionary Programming VII: Proc. EP 98 s: 591-600. Springer-Verlag, New York.

Uğur, A., Kınacı, A., C., 2006. "Yapay Zeka Teknikleri ve Yapay Sinir Ağları Kullanılarak Web Sayfalarının Sınıflandırılması". XI. Türkiye' de İnternet Konferansı (inet-tr'06), Ankara, 1-4.

Wolberg W., H. 1992. Breast Cancer Wisconsin. <http://archive.ics.uci.edu/ml/datasets/Breast+Cancer> (Erişim Tarihi: 15.08.2012)

Xiaohui, H. 2006. PSO Tutorial. <http://www.swarmintelligence.org/tutorials.php> (Erişim Tarihi: 15.08.2012).

Zhao, F., Ren, Z., Yu, D., Yang, Y. 2005. "Application of An Improved Particle Swarm Optimization Algorithm for Neural Network Training". IEEE 0-7803-9422.