

Parçacık Sürüsü Optimizasyonun Yapay Sinir Ağının Eğitiminde Uygulanması ve Ağırlık Entropi Değişiminin İncelenmesi

Özlem İMİK ŞİMŞEK¹, Barış Baykant ALAGÖZ², Ali KARCI³

¹İnönü Üniversitesi, Bilgisayar Mühendisliği Bölümü, Malatya, Türkiye (imikozlem@gmail.com)

¹İnönü Üniversitesi, Bilgisayar Mühendisliği Bölümü, Malatya, Türkiye (baykant.alagoz@inonu.edu.tr)

¹İnönü Üniversitesi, Bilgisayar Mühendisliği Bölümü, Malatya, Türkiye (ali.karci@inonu.edu.tr)

Received Date : Apr. 06, 2020.

Acceptance Date : May 27, 2020.

Published Date : Dec. 1, 2020

Özetçe: İleri beslemeli Yapay Sinir Ağlarının (YSA) eğitiminde yaygın olarak geriye yayılım algoritması gradyan iniş yöntemi (GİY) ile uygulanmaktadır. Diğer taraftan rastgele süreçleri içeren metasezgisel yöntemlerin yapay sinir ağlarının eğitimlerinde kullanımına dönük çalışmalar yapılmaktadır. Metasezgisel optimizasyon yöntemleri karmaşık optimizasyon problemlerinin çözümünde, özellikle de çok sayıda ve karmaşık kısıtların olduğu durumda kolaylık sağlayabilmektedir. Bu çalışmada ileri beslemeli bir YSA'nın eğitiminde Parçacık Sürüsü Optimizasyonu (PSO) yönteminin bir uygulaması gerçekleştirilmiştir. Çalışmada vücut yağ oranı verilerinin ileri beslemeli bir yapay sinir ağı ile modellenmesi gerçekleştirilmiştir. Yapay sinir ağı modelinin eğitimi için GİY ve PSO algoritmaları kullanılmıştır. Eğitim süresince eğitim hatasının ve ağırlık matrisi entropilerinin değişimleri incelenmektedir.

Anahtar kelimeler: Parçacık Sürüsü Optimizasyonu, Yapay Sinir Ağı, Entropi, Gradyan İniş Yöntemi, Matlab

Abstract: In training of feed neural networks (ANN), backward propagation algorithm was applied by gradient descent method (GDM). However, studies related to employment of metaheuristic methods, which includes random processes, are conducted for neural network training task. Metaheuristic optimization methods can facilitate the solution of complex optimization problems, particularly for the cases that there are too many and sophisticated constraints. In this study, an application of Particle Swarm Optimization (PSO) method has been performed in the training of a feed-forward ANN. In the study, modeling of body fat rate data was carried out by using a feed-forward neural network model. GDM and the PSO methods are used for training of neural network model. During the training process, changes of data training errors and weight matrix entropies are examined.

Keywords: Particle Swarm Optimization, Artificial Neural Network, Entropy, Gradient Descent Method, Matlab

1. GİRİŞ

Yapay Sinir Ağları (YSA), yapay zekâ ve makine öğrenmesi alanlarında çalışılan temel bir konu haline dönüşmüştür. Bu konuda öncül çalışmalar, 1943 yılında fizyolojist Waracen McCulloch ve matematikçi Walter Pitts tarafından biyolojik beynin çalışmasını taklit etmek amacı ile önerilen ilk basit yapay sinir hücre modeli ile başlamıştır [1]. 1961 yılında Rosen Blatt çok kamanlı ağların eğitimi için (backpropagation) geriye yayılım algoritmasını önermiştir [2]. Böylece, katmanlı yapay sinir ağı modelleri kolaylıkla eğitilebilir duruma gelmiş ve yapay sinir ağlarının pratik uygulamaları yaygınlaşmıştır. Zaman içinde, veri bilimi dalında tanıma (identification), sınıflama (classification), modelleme (modelling) ve tahmin (prediction) gibi temel işlevlerde kullanılan bir araca dönüşmüştür [3]. Uygulamada, YSA modelleme [4], kontrol [5], işaret işleme [6], görüntü işleme [7] ... vb gibi bir alanlarda kullanım bulmuştur.

Yapay sinir ağlarının pratik bir probleme uygulanma süreci uygun bir yapay sinir ağı mimarisinin (Giriş katmanı, gizli katman ve çıkış katmanı) belirlenmesi ile başlar. Eğitim için uygun verileri içeren bir eğitim kümesinin oluşturulması ile süreç devam eder. Problemin çözümünü karakterize edebilen eğitim kümesi oluşturulduktan sonra YSA'nın eğitimi gerçekleştirilir. YSA'nın eğitimde, oluşturulmuş eğitim kümesindeki giriş verilerine karşılık doğru çıkış verilerini en az hata ile üretmesi beklenir. Bunun için YSA'nın ağırlık değerleri ve bias (eşik) değerleri hatayı minimize etmek üzere optimize edilir [8,9,10]. Dolayısı ile YSA eğitimi temelde eğitim kümesi üzerinde tanımlanan bir eğitim hatası fonksiyonunu minimize eden ağırlık ve katsayıların belirlenmesi problemine indirgenir. Bu konuda geriye yayılım algoritması en yaygın kullanım bulan YSA eğitim yöntemidir ve GİY ile optimizasyon işlemi gerçekleştirilir. Gradyan iniş optimizasyonu yapay sinir ağının lineer olmayan ağırlıklandırılmış toplam yapısına uyum sağlamış ve kolay hesaplanabilir iteratif parametre güncelleme çözümleri sunmuştur [2]. Ancak, YSA eğitiminde ağırlık ve parametre optimizasyon çözümleri GİY ile sınırlı kalmamış, ve bu optimizasyon problemi diğer optimizasyon teknikleri yardımı ile de çözülmüştür. Konu ile doğrudan ilgili olması nedeni ile özellikle metasezgisel optimizasyon yöntemlerinin YSA eğitiminde uygulamalarını konu alan bazı çalışmaları inceleyelim:

YSA'nın yapısı ve parametrelerinin optimizasyonu genetik algoritma ile sağlanmıştır [12]. Bu çalışmada genetik algoritmaya dayalı YSA eğitiminin güneş lekeli tahmini uygulaması için başarısı gösterilmiştir. Diğer bir çalışmada ileri beslemeli bir YSA'nın genetik algoritma ile eğitimi gösterilmiş ve çalışmada genetik algoritmanın, geriye yayılım algoritmasına göre avantajları tartışılmıştır [13]. PSO algoritmasının da YSA'da uygulamaları yapılmıştır: PSO algoritması ile yapay sinir ağının eğitimi sağlanmıştır [3,8,14]. Çok katmalı yapay sinir ağlarının eğitimine dönük çalışmalar PSO ile gerçekleştirilmiş ve elde seçilen üç sınıflama ve iki regresyon problemin de uygulamaları yapılarak performans değerlendirmeleri sunulmuştur [15].

Bu çalışmada vücut-yağ oranı verilerinin YSA algoritması ile modellenmesi problemi üzerinde uygulama yapılmıştır. Ağın eğitimi için geriye yayılım algoritmasını uygulayan GİY ve PSO algoritması kullanılmış ve eğitim performansı değerlendirilmiştir. Eğitim hatasının eğitim boyunca değişimi yanında ağırlık matrisi ve bias (eşik) vektörü değerlerinin entropi değişimleri incelenmiştir. Bu amaçla ile eğitim sürecinin her iterasyonunda ağırlık ve bias matrislerinin entropi değerleri hesaplanmış, ağırlık ve bias entropilerinin optimizasyon süresince değişimleri incelenmiştir. Ağırlık entropilerinin YSA eğitimdeki rolü araştırılmıştır.

Makalenin organizasyonu şöyle yapılmıştır: Gelecek bölümde kullanılan yöntemler ile ilgili temel bilgiler sunulmuştur. Çalışmada kullanılan YSA yapısı, PSO yöntemi ve bu eğitim sırasında sinir ağının anlık ağırlık ve bias (eşik) değerlerinin entropi değerlerinin hesaplanmasında kullanılan Shannon entropisi kavramı hakkında kısa bilgiler sunulmuştur. Bir sonraki bölümde PSO'nun YSA eğitiminde kullanım yöntemi açıklanmıştır. Son bölümde ise vücut-yağ oranı verilerinin YSA algoritması ile modellenmesi için MATLAB ortamında gerçekleştirilen çalışmanın sonuçları ve elde edilen bulgular değerlendirilmiştir.

I.

2. KONU İLE İLGİLİ TEMEL BİLGİLER

2.1. Çalışmada Kullanılan Yapay Sinir Ağı

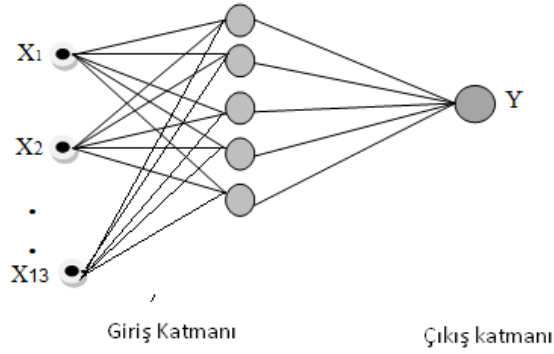
Şekil 1’de bu çalışmada eğitimi yapılan ileri beslemeli YSA’nın mimarisi gösterilmiştir. Vücut-yağ oranının YSA modellenmesi için eğitilen sinir ağı, 13 giriş (x_1, x_2, \dots, x_{13}) ve 1 çıkışa (y_1) sahiptir. Bu veri kümesinde 13 farklı fiziksel vücut parametresi için vücut yağ oranları ölçülmüştür. Bu parametreler yaş, ağırlık, boyun çevresi, göğüs çevresi, karın bölgesi çevresi, kalça çevresi, üst bacak çevresi, diz çevresi, ayak bileği çevresi, omuz çevresi, kolun çevresi, bilek çevresi verileridir ve santimetre biriminde ölçülmüştür. Veri seti 252 adet ölçüm verisinden oluşmaktadır. Bu eğitim setinin eğitimi için aktivasyon fonksiyonu olarak sigmoid fonksiyonu kullanılmıştır.

$$y_j = f(s_j) = \frac{1}{1 + e^{-s_j}} \quad (1)$$

Burada s_j toplam terimi, j . nöronun girişlerinin ağırlıklandırılmış toplamına bias eklenerek,

$$s_j = b_j + \sum_i w_{i,j} x_i \quad (2)$$

ile ifade edilebilir. Burada, $w_{i,j}$ ağırlığı, i . giriş ve j . nöron için ağırlık katsayısıdır ve b_j ise j . nöron için bias katsayısıdır.



Şekil 1. Çok katmanlı yapay sinir ağı modeli

Burada ağırlık vektörü $W = [w_{i,j}]$ ve bias vektörü $B = [b_j]$ eğitim süresince optimize edilerek sinir ağının verilen girişler için doğru çıkışları üretmesi sağlanır. Bu eğitim problemi için eğitim hatası

$$E = \frac{1}{2} \sum_{i=1}^{252} (d_i - y_i)^2 \quad (3)$$

ile ifade edilmiştir. Burada eğitim kümesinin i verisi için doğru vücut yağ oranı d_i değeridir. Bu veri için YSA'nın çıkışı y_i ile temsil edilmiştir.

2.2. Parçacık Sürüsü Optimizasyonu (PSO)

Parçacık sürüsü optimizasyonu canlıların sürü davranışlarından esinlenerek geliştirilmiştir. Çözümü temsil eden sürü bireylerinin arama uzayında gezinerek en iyi çözümü arama mekanizmasına dayanan metasezgisel bir optimizasyon algoritmasıdır [3]. PSO'nun en temel avantajlarından biri sürü zekası olarak adlandırılan kolektif zekayı kullanmasıdır [3, 9]. Probleme göre ayarlaması gereken parametre sayısının diğer klasik metasezgisel yöntemlere göre nispeten az olması kullanım kolaylığı sağlar [14].

PSO aday çözümü temsil eden parçacıkların arama uzayı içinde en iyi çözümlere doğru hareketini sosyal sürü davranışı ile modeller, parçacıkların optimal çözüme varmasını hedefler [3,9,11].

PSO algoritmasının başlangıcında, parçacıklar arama uzayına rastgele dağılırlar [3]. Her iterasyonda aday çözümleri temsil eden her bir parçacığın uygunluk değeri hesaplanır. Sürünün lokal ve global en iyi parçacıkları uygunluk değerine göre belirlenir. Bir sonraki iterasyonda, her bir parçacığın hareketi elde edilen lokal ve global en iyi parçacığın konumları dikkate alınarak güncellenir [3, 11]. Bu işlem, son iterasyon tamamlanana kadar devam eder. Her bir parçacık lokal ve global en iyi parçacıkların yönlendirmesi ile sürekli daha iyi çözümlere doğru hareket etme eğilimi gösterir [11].

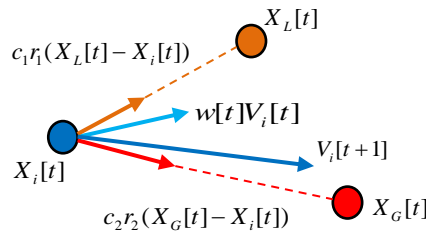
Çok boyutlu arama uzayında, i . parçacığın konumunu $x_i[t]$ ve hızını $v_i[t]$ ile gösterilsin. PSO algoritması parçacıkların hızlarını şu denkleme göre günceller.

$$V_i[t+1] = w[t].V_i[t] + c_1.r_1.(X_L[t] - X_i[t]) + c_2.r_2.(X_G[t] - X_i[t]) \quad (4)$$

Parçacığın bu yeni hızına karşılık parçacığın yeni konumu,

$$X_i[t+1] = X_i[t] + V_i[t+1] \quad (5)$$

ile güncellenir. Burada n boyutlu bir arama uzayı içinde her bir parçacığın konum ve hızı n boyutlu bir vektör ile temsil edilir. Şekil 2’de lokal en iyi ve global en iyi parçacığa göre bir parçacığın hareket yönünün belirlenmesi temsili olarak gösterilmiştir [11]. Denklem (4) 'de lokal en iyi çözüm $X_L[t]$ ve global en iyi çözüm $X_G[t]$ ile ifade edilir. c_1 katsayısı lokal öğrenme katsayısıdır ve lokal en iyiye yönelimin derecesini ayarlar ve c_2 katsayısı global öğrenme katsayısıdır ve global en iyiye yönelimin derecesini ayarlar [9]. Burada r_1 ve r_2 bir rastgele sayıdır ve parçacık hareketlerine bir rastgele serbestlik derecesi kazandırır [11]. Bu rastgele serbestlik derecesi algoritmaya her iterasyonda yeni çözümler bulabilme imkanı sağlar. İvme parametresi $w[t]$ parçacığın yavaşlamasını sağlar ve $w[t+1] = \xi w[t]$, $0 < \xi \leq 1$ ile her iterasyon adımında güncellenir. İterasyonlar ilerledikçe azalan ivme $w[t]$, parçacıkları buldukları iyi çözümlere yerleşmelerine imkan sağlar [9].



Şekil 2. PSO algoritmasında lokal ve global değerlerin parçacık hareketine etkileri [11]

2.3. Şanon Entropi ve Yapay Sinir Ağlarında Kullanımı

Entropi, bilgi kuramında ve veri analizinde bir olasılık dağılımının belirsizliğini ölçmek için kullanılır [16]. Bilimsel çalışmalarda kullanılan entropi kavramı, veri kümesi içerisinde bulunan öğelerin düzensizliğinin reel sayılarla bir ifadesi olarak kabul görmüştür [17]. Entropi pozitif ve sıfır reel sayı değeri alır ve entropi değerinin artması sistemdeki düzensizliğin artması durumunu ifade eder. Sıfır entropi veri kümesinin tek değer aldığı durumda söz konusudur. Bu durumda rastgele alınan bir

elemanın değeri kesin olarak bilinir ve dolayısı ile belirsizliği sıfır olarak kabul edilir. Aldığı değerler çeşitlendikçe entropi değeri artar [17]. Bunun nedeni, rastgele seçilen bir elemanın değerinin doğru tahmin edilebilme olasılığı düşer ve belirsizliğin artması olarak kabul edilir. Bilgi kuramında en yaygın kullanılan entropi tanımı Shannon entropisidir [17]. Ayırık bir olasılık dağılımı için

II.

$$\text{III. } S = - \sum_{i=1}^N p_i \log_2 p_i \quad (6)$$

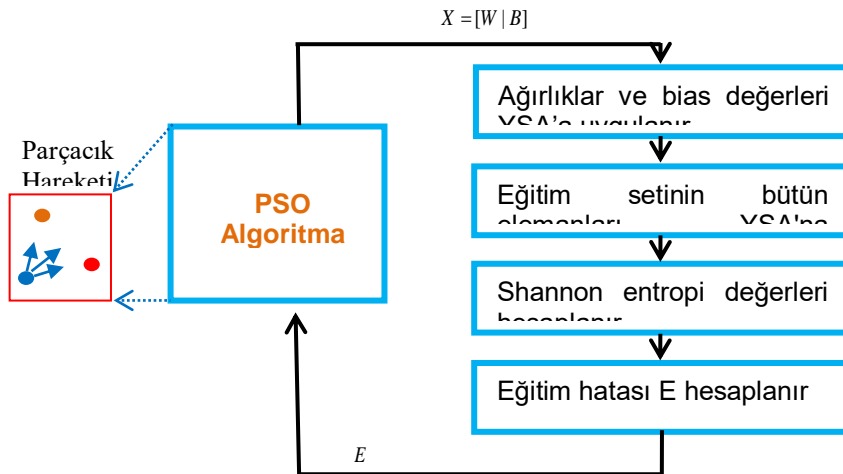
IV.

V. ile ifade edilir. Burada p_i veri kümesindeki i . veri sınıfının tüm sınıf içerisindeki bulunma olasılığını ifade etmektedir. [16, 17]

VI. Bu çalışmada YSA'nın farklı algoritmalar ile eğitimi boyunca her iterasyonda ağırlık matrisi ve bias vektörünün entropileri hesaplanmış ve bu entropi değerlerinin eğitim süresince değişimi incelenmiştir. Yüksek entropi ölçülmesi, ağırlık değerlerinin düzensiz olarak dağıldığını gösterir ve katsayılardaki düzensizliğin artışı ifade eder. Bu çalışmada entropi hesaplamasında ağırlık matrisinin ve bias vektörlerinin ayrı ayrı histogramları hesaplanmıştır. Bunun için Matlab'ın *hist()* fonksiyonu kullanılmıştır. Bu histogram değerleri normalize edilerek dağılım fonksiyonları kestirilmiştir. Kestirimi yapılan dağılım fonksiyonları doğrudan Shannon entropi hesaplamasında kullanılmıştır.

3. PSO ALGORİTMASININ İLERİ BESLEMELİ YAPAY SİNİR AĞLARININ EĞİTİMİNDE UYGULAMASI

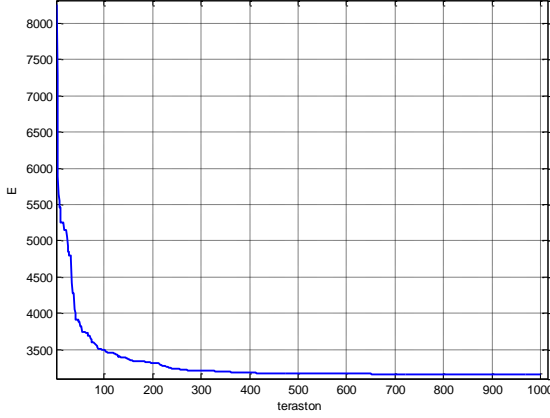
Şekil 3'de PSO algoritmasının YSA eğitiminde nasıl uygulandığını gösteren akış diyagramı gösterilmektedir. Optimizasyonun her iterasyon adımında YSA'nın ağırlık matrisi ve bias vektörünün hesaplaması gerçekleştirilmiştir. Burada, PSO algoritmasının parçacık konumu ağırlık ve bias vektörünün birleşimi ile $X=[W|B]$ ifade edilmiştir. PSO tarafından belirlenen parçacık konumları $X=[W|B]$ yapay sinir ağ modelinde uygulanmakta ve eğitim seti verileri için eğitim hatası E hesaplanmaktadır. Her iterasyonda ağırlık matrisi Shannon entropisi S_w ve bias vektörü Shannon entropisi S_b hesaplanmaktadır. Eğitim hatası $X=[W|B]$ aday çözüm (parçacık) konumunun uygunluk değeri olarak tekrar PSO algoritmasına dönülmektedir. Şekil 3'de görülen akış diyagramı MATLAB ortamında gerçekleştirilmiştir. Hata değişimleri ve entropi değişimlerini gösteren grafikler elde edilmiştir.



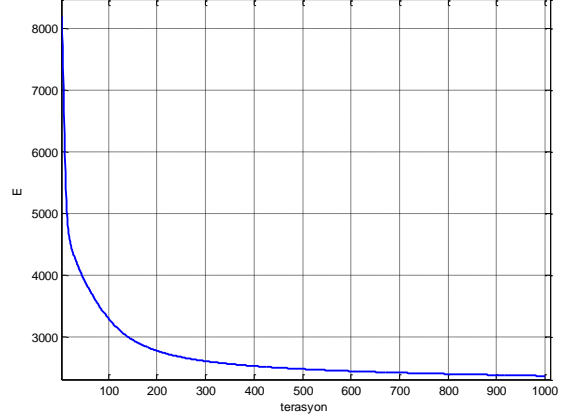
Şekil 3. PSO algoritmasının YSA eğitimine uygulandığını özetleyen akış diyagramı

3.1 Eğitim Sonucunda Elde Edilen Sonuçlar

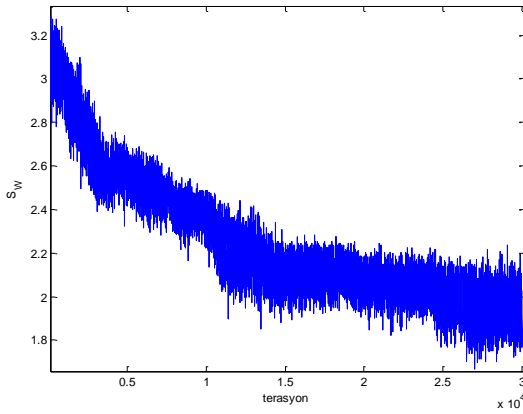
Bu çalışmada PSO parametreleri şöyle konfigüre edilmiştir: Optimizasyonda 30 adet parçacık kullanılmıştır. Maksimum iterasyon sayısı 1000 ile sınırlanmıştır. Bireysel öğrenme katsayıları $c_1 = 2.0$ ve $c_2 = 2.0$ olarak ayarlanmıştır. İvme katsayısı $w = 1$ ve $\xi = 0.99$ alınmıştır. Geliştirilen Matlab programı ile 13x252 'lik vücut yağ veri kümesi iki farklı algoritma {PSO, GİY} ile YSA eğitiminde kullanılmıştır. Eğitimin her iterasyonda Shannon entropi değerlerinin hesaplanması eğitim süresini çok artırmıştı. Eğitim süresince entropi değişimi izlenmiş ve elde edilen bulgular tartışılmıştır.



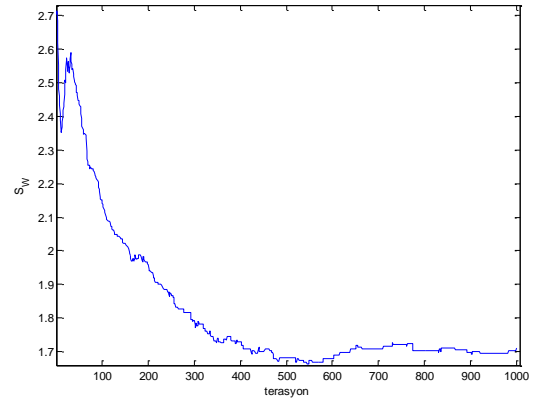
Şekil 4. PSO ile YSA Hata Değişimi



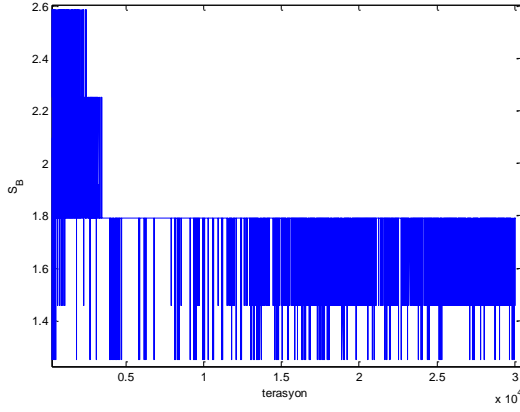
Şekil 5. GİY ile YSA Hata Değişimi



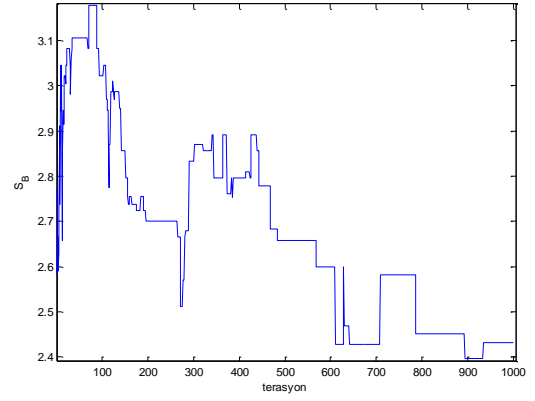
Şekil 6. PSO ile YSA Eğitiminde Ağırlıkların Entropi Değişimi



Şekil 7. GİY ile YSA Eğitiminde Ağırlıkların Entropi Değişimi



Şekil 8. PSO ile YSA Eğitimi Bias Değeri Entropi Değişimi



Şekil 9. GİY ile YSA Eğitimi Bias Değeri Entropi Değişimi

Şekil 4 ve 5 'de eğitim süresince her iki optimizasyon yöntemi için eğitim hatasının değişimi verilmektedir. Bu sonuçlar YSA'nın eğitimde, GİY 'nin çok daha hızlı ve başarılı bir şekilde eğitim hatasını düşürebildiği göstermiştir. Bunun temel nedeni hata fonksiyonun eğitim bilgisini kullanarak minimuma giden GİY yönteminin her iterasyon adımı daha verimli kullanabiliyor olmasıdır. Şekil 6 ve 7 'de bu eğitim boyunca ağırlık matrisi entropilerinin değişimleri verilmektedir. PSO ile gerçekleştirilen eğitimin entropi değerlerinin daha yüksek kaldığı görülmektedir. GİY ile eğitim boyunca ağırlık entropi değeri daha düşük kalmış ve daha az değişkenlik göstermiştir. Bu durum PSO ile optimizasyon da rastgele süreçler ve 30 farklı aday çözüm (parçacık sürüsü) ile hesaplama yapılması nedeni ile ağırlık matrisinde değerlerin düzensizliğinin (belirsizliğinin) daha fazla olmasından kaynaklanmıştır. Bekleneceği üzere GİY yöntemin ağırlık değişimleri tek aday çözümü ile ilerlemesi ve rastgele süreçler içermemesi nedeni ile ağırlık matrisi değerleri daha düzenli ve yavaş değişmektedir. Şekil 8 ve 9 'da eğitim süresince elde edilen bias vektörlerinin entropi değerleri sunulmuştur. Bias vektörü değerlerinin entropisi PSO 'da daha düşük elde edilmiştir. Bu PSO optimizasyonunda bias vektör değerlerinin daha düzenli kalabildiğine işaret etmiştir. Bu durumun arama işlemi için ihtiyaç duyulan düzensizliğin PSO 'da büyük oranda ağırlık matrisi ile karşılanmasının bir sonucu olduğu kanaatindeyiz. Her iki yöntemde ağırlık entropilerinin eğitim süresince nispeten yakın seviyelerde olduğu görülmektedir. Ancak, PSO ile optimizasyonda entropi değişkenliği çok daha yüksektir. Bu PSO'daki rastgele serbestlik derecesinin bir doğal sonucudur. Eğitimlerin ardından elde edilen sonuçlar ve istatistiki değerlendirmeleri Tablo 2 ve Tablo 3 de verilmiştir.

Tablo 2. Şekil 4-9'daki verilen sonuçlar için 1000 iterasyon tamamlandığı durumda hata, ağırlık entropi ve bias entropi değerleri

Eğitim Türü	İterasyon Sayısı	Hata Değeri	Ağırlık Entropi Değeri (S_w)	Bias Entropi Değeri (S_b)
YSA + PSO	1000	3150	1.87	1.45
YSA + GİY	1000	2377	1.71	2.43

Tablo 3. YSA+PSO ve YSA+GİY ile 10 farklı konfigürasyonda eğitim sonuçları ve istatistiki değerlendirmeler

Eğitim Türü	Son İterasyonda Hata Değeri	Entropi Değeri	Min Değeri	Max Değeri	Ortalama Değeri	Standart Sapma
YSA + PSO	3.4892e+003	1.4591	3.4892e-003	9.0512e+003	3.6013e+003	354.9544
	3.5144e+003	2.2516	3.5144e-003	8.3153e+003	3.6137e+003	378.1126
	3.1306e+003	1.4591	3.1306e-003	8.7093e+003	3.3393e+003	500.8730
	3.7577e+003	0.6500	3.7577e-003	8.6100e+003	3.8492e+003	356.5626
	3.3678e+003	1.4591	3.3678e-003	9.1241e+003	3.7650e+003	393.8632
	3.0619e+003	1.5850	3.0619e-003	9.2956e+003	3.2749e+003	386.2745
	3.2130e+003	1.7925	3.2130e-003	8.1490e+003	3.3503e+003	415.5156
	3.3685e+003	2.2516	3.3685e-003	8.6644e+003	3.5176e+003	552.2002
	3.4978e+003	1.9183	3.4978e-003	8.6140e+003	3.8492e+003	478.1243
	3.4892e+003	1.4591	3.4892e-003	9.0512e+003	3.6013e+003	354.9544
YSA + GIY	2.3713e+003	1.7127	2.3713e-003	8.3495e+003	2.7074e+003	666.3019
	2.3749e+003	1.6425	2.3749e-003	8.4027e+003	2.7147e+003	679.4003
	2.3713e+003	1.6652	2.3713e-003	8.3114e+003	2.7017e+003	647.0539
	2.3762e+003	1.6983	2.3762e-003	8.3632e+003	2.7110e+003	668.6444
	2.3722e+003	1.6832	2.3722e-003	8.3756e+003	2.7039e+003	667.0050
	2.3743e+003	1.8032	2.3743e-003	8.4132e+003	2.7184e+003	687.6888
	2.3763e+003	1.6685	2.3763e-003	8.3798e+003	2.7123e+003	669.4995
	2.3730e+003	1.7509	2.3730e-003	8.3404e+003	2.7045e+003	661.9861
	2.3704e+003	1.6849	2.3704e-003	8.3703e+003	2.7041e+003	667.2906
	2.3748e+003	1.7640	2.3748e-003	8.3513e+003	2.7114e+003	667.8367

Tablo 3 'de eğitim sürecinin istatistiki değerlendirme için YSA 'nın PSO ve GIY ile eğitimi 10 defa tekrarlanmıştır. Her test 1000 iterasyon çalıştırılarak sonuçların hata değeri ve entropi değeri kayıt edilmiştir. Hata değerlerinin optimizasyonu süresince maksimum değeri, minimum değeri, standart sapması ve ortalaması hesaplandı ve Tablo 3 'de sunuldu. Tablo incelendiğinde PSO ile eğitimde hatasının minimum değeri GIY yöntemine göre daha yüksek değişkenlik gösterdiği görülmüştür. GIY yönteminin eğitim hatası tutarlılığı PSO'ya göre çok daha iyidir. Bunun temel nedeni, PSO 'nun rastgele süreçler içermesinden dolayı parametre optimizasyonunun her çalışmasında farklılık gösterme potansiyelinin daha yüksek olmasından kaynaklanmıştır. Aynı zamanda PSO ile yapılan optimizasyonda ağırlık entropi değerlerinin GIY yöntemine göre daha düşük veya yüksek elde edilebilmesine imkân sağladığı görülmüştür. Bununla birlikte hata değerleri bir birine yakın olmasına karşın ağırlık entropi değerleri farklı eğitim çözümlerinin YSA eğitiminde mevcut olduğu görülmüştür. YSA eğitiminde farklı ağırlık entropi değerine sahip çözümlerin PSO ile elde edilebileceği görülmüştür. Ancak, vücut-yağ oranı verilerinin YSA ile modellenmesi uygulaması için ağırlık entropi değeri GIY 'nin PSO 'ya göre daha tutarlı ve daha düşük hatalı sonuçlar verebildiği görülmüştür.

Çalışmanın bir sonraki adımın PSO ile YSA eğitiminin GİY ile YSA eğitim hatası performanslarına yaklaşım yaklaşamayacağını araştırmak için PSO'nun eğitiminde kullanılan parametre değerleri üzerinde ayarlama yapıldı. Bu parametrelerden biri iterasyon sayısı iken diğeri parçacık sayısıdır. İterasyon sayısının artırılması PSO 'nun yakınsadığı lokal minimum civarında daha düşük hata değerlerine ulaşabilmesine imkan sağlamaktadır. İterasyon sayısının PSO eğitim performansına etkileri Tablo 4 'de sunulmuştur. Diğer bütün PSO parametrelerinin sabit tutulup iterasyon sayısının artırılmasının eğitimlerin ortalama mutlak hatasını PSO 'da artırabildiği görülmüştür.

Tablo 4. İterasyon Sayısı Değişikliği İle GİY-YSA ile PSO-YSA Ortalama Mutlak Hata Değeri

İterasyon Sayısı	GİY-YSA Ortalama Mutlak Hata	PSO-YSA Ortalama Mutlak Hata
200	3.805535	4.855843
400	3.805535	4.712247
600	3.805535	4.143270
800	3.805535	4.086126
1000	3.805535	4.065197

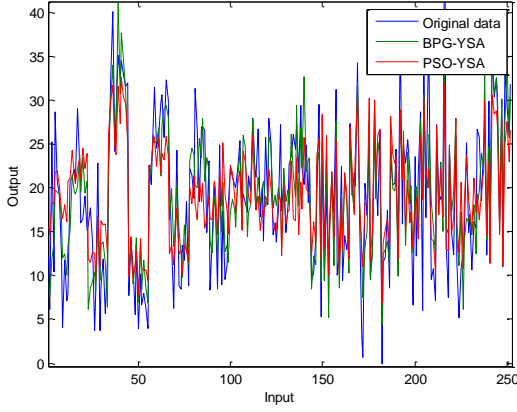
Tablo 4 'deki veriler incelendiğinde 5 farklı iterasyon sayısı durumunda GİY-YSA'nın ortalama mutlak hata değerinin çok değişmediği görülmüştür. Deterministik karaktere sahip bir yöntem olan GİY-YSA'nın 200. iterasyonda hata değerinin nerede ise sıfıra çok yakın değerlere ulaşması nedeni ile iterasyon sayısının daha fazla artırılmasının ortalama mutlak hataya önemli bir etkisinin olmadığı görülmüştür. Bu sonuçlara göre GİY-YSA 'nın iterasyon sayısının 200' den daha yüksek değerlere çıkarılmasının pratik olarak mutlak ortalama hata performansına etkisinin olmayacağı kabul edilebilir. PSO-YSA'da iterasyon sayısı arttıkça mutlak hatanın azaldığı görülmektedir.

Diğer taraftan, optimizasyonda parçacık sayısı PSO 'nun arama uzayı içerisinde rastgele dağılıma ve optimal çözüme ulaşma potansiyelini etkileyen bir diğer önemli parametredir. Tablo 5 'de iterasyon sayısı 500 ve diğer parametrelerinde sabit tutulduğu koşullarda parçacık sayısı 5 'den 100 'e kadar artırılarak YSA eğitimi yapılmıştır. Ortalama mutlak hata değerleri Tablo 5 'de sunulmuştur.

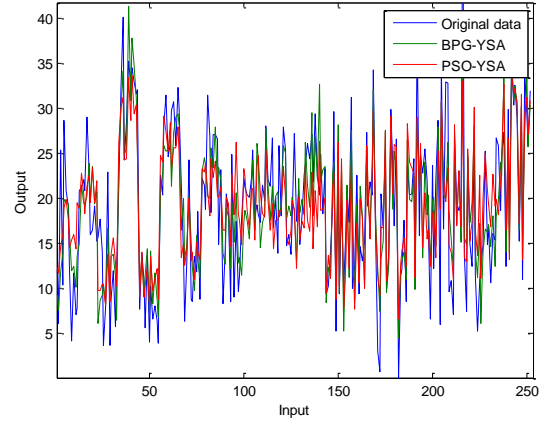
Tablo 5. Parçacık Sayısı Değişikliği İle GİY-YSA ile PSO-YSA Ortalama Mutlak Hata Değeri

Parçacık Sayısı	GİY-YSA Ortalama Mutlak Hata	PSO-YSA Ortalama Mutlak Hata
5	3.805535	4.837814
10	3.805535	4.900484
30	3.805535	4.430442
50	3.805535	4.408367
100	3.805535	4.269381

Tablo 5 'deki veriler incelendiğinde GİY ile YSA eğitiminde mutlak hata değerlerinin değişmeyeceği kabul edilebilir çünkü GİY bir sürü optimizasyon tekniği değildir. PSO-YSA da parçacık sayısının artışının mutlak hata değerlerinin azalmasına imkan verebildiği gösterilmiştir. Tablo 4 ve Tablo 5'de elde edilen sonuçlara göre iterasyon sayısının ve parçacık sayısının artırılmasının PSO-YSA eğitime katkı sağlayabildiği gösterilmiştir. Şekil 10 ve 11'de parçacık sayısı 100 durumu ve iterasyon sayısı 1000 durumları için PSO-YSA ve GİY_YSA eğitim sonuçları orijinal veri seti ile karşılaştırılmıştır.



Şekil 10. PSO-YSA ile GİY-YSA 100 parçacık sayısı için eğitim sonuçları



Şekil 11. PSO-YSA ile GİY-YSA 1000 iterasyon sayısı için eğitim sonuçları

Şekil-10 ve Şekil-11'de görülen eğitim sonuçları incelendiğinde GİY-YSA eğitiminde ezberleme (over-fitting) eğiliminin daha fazla olduğu, PSO-YSA eğitiminin daha iyi genelleme performansı sunabileceği görülmektedir. Burada PSO ile eğitimin düşük ağırlık entropisi sunabilmesi genelleme performansına katkı sağlayabilmiştir.

4. SONUÇLAR

Bu çalışmada, vücut-yağ oranı verilerinin YSA ile modellemesi uygulamasında GİY ve PSO tabanlı eğitim algoritmaları ile YSA eğitim performansları ve eğitim süresince katsayı entropi değerleri değişimleri incelenmiştir. Tekrarlı testler sonucunda istatistiksel olarak GİY 'in PSO 'ya göre daha tutarlı ve düşük hatalı eğitim sağlamasına rağmen, yüksek iterasyonlu eğitimlerde ezberleme eğiliminin ortaya çıkabileceği görülmüştür. PSO ile YSA eğitiminde ise daha yüksek hata değerleri sunmasına rağmen genelleme performansının GİY 'e göre daha iyi olabileceği görülmüştür. PSO algoritmasının rastgele serbestlik derecesi olması nedeni ile eğitim performansı eğitimden eğitime daha fazla değişkenlik gösterebilmekte ve bu durum farklı ağırlık ve bias entropi değerlerine sahip çözümlere ulaşabilmesine imkan sağlamaktadır. Düşük ağırlık entropisine sahip eğitimlerin genelleme performansının daha iyi eğiliminde olması dikkate değer bir bulgudur ve YSA'larda verinin ezberlenmesi (over fitting) sorunu için bir çözüm yolu olma potansiyeline sahiptir.

Literatürde entropi kavramı YSA 'da kayıp fonksiyonu (hata fonksiyonu) olarak uygulama bulabilmiştir. YSA eğitiminde, çapraz entropi kayıp fonksiyonu (Cross-entropy loss function) karesel hataya göre avantajlar sağlayabildiği gösterilmiştir [18]. Derin öğrenme çalışmalarında çapraz entropi fonksiyonun daha hızlı eğitim sağladığı gösterilmiştir [18]. Görüntü işlemede kullanılan bir yapay sinir ağı için özelliklerin iyileştirilmesi için maksimum entropi prensibi kullanılmıştır [19]. Ancak, YSA'nın ağırlık ve bias katsayılarının entropi değişimlerinin incelenmesi konusunda yeterli bir çalışma bulunamamıştır. Ağırlık entropisi, temelde eğitim sırasında ağırlık katsayılarının düzensizliği ve içerdiği bilgi miktarındaki belirsizlik hakkında bilgi sağlamaktadır. Ağırlıkların entropi değerlerinin yükselmesi, düzensizliğinin artması diğer bir ifade ile bilgi zenginliğinin artışı anlamına gelmektedir. Bunun eğitim hata değerinin azalması yönünde etkisi olduğu ve bunun sonunda eğitim verilerinin ezberlemesi durumuna yol açacağı görülmüştür. Bu nedenle düşük entropi değerlerine sahip eğitimlerin genelleme eğilimlerinin daha yüksek olduğu görülmüştür. Bu nedenle, gelecek çalışmalarda YSA'nın eğitim performansının ayarlanabilmesinde hata minimizasyonu yanında ağırlık entropi değerleri kısıtlarının kullanılmasının sağlayacağı katkılar detaylı incelenmelidir.

Kaynaklar

- [1] McCulloch W. S, Walter P. (1943) *A logical calculus of the ideas immanent in nervous activity*. The bulletin of mathematical biophysics 5.4: 115-133.

- [2] Widrow B, Lehr M. A. (1990) *30 years of adaptive neural networks: perceptron, madaline, and backpropagation*. Proceedings of the IEEE 78.9: 155-1442.
- [3] Çevik K.K, Koçer H.E. (2013) *Parçacık Sürü Optimizasyonu ile Yapay Sinir Ağları Eğitime Dayalı Bir Esnek Hesaplama Uygulaması*. Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 39-45.
- [4] Aminian J, Shahrokh S. (2008) *Evaluation of ANN modeling for prediction of crude oil fouling behavior*. Applied thermal engineering 28.7: 668-674.
- [5] Hasanien H. M. (2011) *FPGA implementation of adaptive ANN controller for speed regulation of permanent magnet stepper motor drives*. Energy Conversion and Management 52.2: 1252-1257.
- [6] Vijaya G, Kumar V. Verma H. K. (1998) *ANN-based QRS-complex analysis of ECG*. Journal of medical engineering & technology 22.4: 160-167.
- [7] Egmont-Petersen M, Ridder D, Handels H. (2002) *Image processing with neural networks—a review*. Pattern recognition 35.10: 2279-2301.
- [8] Çavuşlu M. A, Karakuzu C, Şahi S. (2010) *Parçacık Sürü Optimizasyonu Algoritması ile Yapay Sinir Ağı Eğitiminin FPGA Üzerinde Donanımsal Gerçeklenmesi*, Politeknik Dergisi, Vol :13, Pages. 83-92
- [9] Imik O, Alagoz B. B. (2017) *Discretization of fractional order transfer functions by weighted multi-objective particle swarm optimization method*. In 2017 International Artificial Intelligence and Data Processing Symposium (IDAP) pp.1-5.
- [10] Carvalho M, Ludermit T.B. (2015) *Particle Swarm Optimization of Neural Network Architectures and Weights*. Seventh International Conference on Hybrid Intelligent Systems.
- [11] Imik Simsek O. (2018) *Parçacık Sürüsü Optimizasyon Yöntemi İle Kesir Dereceli Filtre Gerçeklenmesi*.
- [12] Lam H. K, Ling S. H, Leung F. H, Tam P.K.S. (2001). *Tuning of the structure and parameters of neural network using an improved genetic algorithm*. In IECON'01. 27th Annual Conference of the IEEE Industrial Electronics Society (Cat. No. 37243) (Vol. 1, pp. 25-30). IEEE.
- [13] Montana D. J, Davis L. (1995) *Training Feedforward Neural Networks Using Genetic Algorithms*
- [14] Eberhart R, Kennedy J. "A new optimizer using particle swarm theory." MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. IEEE.
- [15] Mendes R, Cortez P, Rocha M, Neves J. (2002). *Particle swarms for feedforward neural network training*. In Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290) (Vol. 2, pp. 1895-1899). IEEE.
- [16] Wang Q. A. (2008). *Probability distribution and entropy as a measure of uncertainty*. Journal of Physics A: Mathematical and Theoretical, 41(6), 065004.
- [17] Bulut F. (2017). *Bilgi Kuramındaki Entropi Kavramıyla İlgili Farklı Matematiksel Modeller*. Bilge International Journal of Science and Technology Research, 1(2), 167-174.
- [18] Kline D. M, and Victor L. B. (2005) *Revisiting squared-error and cross-entropy functions for training neural network classifiers*. Neural Computing & Applications 14.4: 310-318.
- [19] Khabou M. A, Paul D. G. (2000) *Automatic target detection using entropy optimized shared-weight neural networks*. IEEE transactions on neural networks 11.1: 186-193.

Ekler:

```
% Matlab ile matrislerin Shannon entropi hesaplaması
% W matrisi.
[a b]=hist(W);
% Histogram dağılımına göre olasılık kestirimi yapılır.
p=a/sum(a);
% Olasılık kestirimleri yardımı ile Shannon entropisi hesaplanır
top=0;
for i=1:length(p)
    if ne(p(i),0)
        top=top+p(i)*log2(1/p(i));
    end
end
Entropy=top;
```