

**T.C.
GİRESUN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**Pİ-SİGMA YAPAY SİNİR AĞLARI İLE TURİST
SAYISI TAHMİNİ**

YÜKSEK LİSANS TEZİ

Öğrencinin Adı SOYADI : Muzhda MAMMADOVA
ORCID : 0000-0002-6696-5767
Tezin Enstitüye Verildiği Tarih : 16/11/2020

Enstitü Anabilim Dalı : İstatistik

Tez Danışmanı : Erol EĞRİOĞLU
ORCID : 0000-0003-4301-4149

**Aralık 2020
GİRESUN**

**T.C.
GİRESUN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**Pİ-SİGMA YAPAY SİNİR AĞLARI İLE TURİST
SAYISI TAHMİNİ**

YÜKSEK LİSANS TEZİ

Muzhda MAMMADOVA

Enstitü Anabilim Dalı

:

İstatistik

Bu tez 08/12/2020 tarihinde aşağıdaki jüri tarafından oybirliği ile kabul edilmiştir.

**Doç. Dr.
Cem KOÇAK
Jüri Başkanı**

**Doç. Dr.
Eren BAŞ
Üye**

**Prof. Dr.
Erol EĞRİOĞLU
Üye**

**Doç. Dr.
Bahadır KOZ
Enstitü Müdürü**

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Muzhda MAMMADOVA

08/12/2020

TEŞEKKÜR

Tez çalışmamın başından sonuna kadar her alanında engin bilgisi, birikim ve tecrübeleri ile bana destek olan çok kıymetli hocam Prof. Dr. Erol EĞRİOĞLU ' na sonsuz teşekkür ve içten saygılarımı sunarım.

Çalışmalarım sırasında sürekli yardım eden başta kıymetli Doç. Dr. Eren BAŞ hocama ve bir o kadar da gerek lisans gerekse yüksek lisansta vermiş oldukları bilgilerden dolayı Giresun Üniversitesi İstatistik Bölümü akademisyenlerine de saygı ile çok teşekkür ederim.

Çalışmalarım sırasında sürekli her türlü desteği esirgemeyen arkadaşım Sinan AKMAN'a da teşekkürlerimi bir borç bilirim.

Bu zorlu süreçte sürekli yanımda olan ve hiçbir zaman desteklerini esirgemeyen aileme de teşekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR.....	I
İÇİNDEKİLER	II
SİMGELER VE KISALTMALAR LİSTESİ	IV
ŞEKİLLER LİSTESİ	V
TABLOLAR LİSTESİ	VI
ÖZET	VII
SUMMARY	VIII
BÖLÜM 1. GİRİŞ.....	1
 BÖLÜM 2. KAYNAK ARAŞTIRMASI	 2
2.1. Tezin Amacı ve Önemi.....	2
2.2. Literatür Taraması.....	3
 BÖLÜM 3. MATERYAL VE YÖNTEM	 4
3.1. Yapay Sinir Ağları.....	4
3.1.1. Yapay Sinir Ağları Avantaj ve Dezavantajları.....	5
3.1.2. Yapay Sinir Ağları Bileşenleri.....	6
3.1.3 Yapay Sinir Ağları Mimari Yapısı	6
3.1.4. İleri Beslemeli Yapay Sinir Ağları.....	7
3.1.5. Geri Beslemeli Yapay Sinir Ağları.....	8
3.1.6. Aktivasyon Fonksiyonu.....	9
3.1.7. Yapay Nöron Modelleri.....	10
3.1.8. Tek Katmanlı Yapay Sinir Ağları	12
3.1.9. Çok Katmanlı Yapay Sinir Ağları	13
3.1.10. Eğitim Algoritmaları.....	14
3.1.11. Geri Yayılım Öğrenme Algoritması.....	15
3.2. Paracık Sürü Optimizasyonu.....	17
3.2.1. Paracık Sürü Optimizasyonu Aşamaları.....	18
3.3. Pi-Sigma Yapay Sinir Ağları.....	19
3.3.1. Pi-Sigma YSA'nın Paracık Sürü Optimizasyonu İle Eğitim Algoritması	22

BÖLÜM 4. ARAŞTIRMA BULGULARI.....	24
4.1. Türkiye’ye Gelen Turist Sayısının Tahmin Edilmesi.....	24
 BÖLÜM 5. TARTIŞMA VE SONUÇ	29
 KAYNAKLAR	30
EKLER	32
ÖZGEÇMİŞ	



SİMGELER VE KISALTMALAR LİSTESİ

ÇNM-YSA	: Çarpımsal Nöron Model Yapay Sinir Ağı
DNM-YSA	:Dentrit Nöron Model Yapay Sinir Ağı
MAPE	: Hatanın Mutlak Yüzdelik Ortalaması
MLP-YSA	: Çok Katmanlı Algılayıcı Yapay Sinir Ağı
PSO	: Parçacık Sürü Optimizasyonu
PS-YSA	: Ps-Sigma Yapay Sinir Ağı
RMSE	: Hata Kareler Ortalaması Karekökü
TÜİK	: Türkiye İstatistik Kurumu
YSA	: Yapay Sinir Ağları

ŞEKİLLER LİSTESİ

Şekil 2.1. Yapay Sinir Ağ Yapısı	7
Şekil 2.2. İleri Beslemeli Yapay Sinir Ağı	8
Şekil 2.3. Geri Beslemeli Yapay Sinir Ağı	9
Şekil 2.4. Yapay Nöron Yapısı	11
Şekil 2.5. İki Girdi Ve Bir Çıktıdan Oluşan Yapay Sinir Ağı Modeli	13
Şekil 2.6. Çok Katmanlı Yapay Sinir Ağ Modeli.....	14
Şekil 2.7. Geri Yayılımlı Yapay Sinir Ağı	16
Şekil 2.8. Çok Katmanlı Ysa'nın Geri Beslemeli Öğrenme Modeli.....	17
Şekil 2.9. Parçacık Sürü Optimizasyonu Akış Şeması	19
Şekil 2.10. Pi-Sigma Yapay Sinir Ağı Mimari Yapısı	21
Şekil 4.1. 1998-2018 Arası Türkiye'ye Gelen Yabancı Ziyaretçi Sayısı	24
Şekil 4.2. Yöntemlerin Öngörülleri ve Gerçek Değerler Grafiği	28
Şekil 4.3. En iyi test performanslarının karşılaştırılması için Nemenyi Testi	28

TABLÖLAR LİSTESİ

Tablo 2.1. Aktivasyon Fonksiyonları.....	10
Tablo 2.2. Bazı Toplama Fonksiyonları.....	12
Tablo 3.1. PS-YSA'nın PSO İle Eğitiminde Bir Parçacığın Pozisyonları.....	22
Tablo 4.1. Holt Ve Rastgele Yürüyüş Süreci Sonuçları.....	25
Tablo 4.2. YSA yöntemlerinden elde edilen sonuçlar	26
Tablo 4.3 Uygulanan tüm yöntemlerin öngöröleri ve en iyi test kümesi performansları	27



PI-SİGMA YAPAY SİNİR AĞLARI İLE TURİST SAYISI TAHMİNİ

ÖZET

Son yıllarda zaman serilerinin öngörü problemi için yapay sinir ağları güçlü bir seçenek oluşturmaktadır. Yüksek dereceli yapay sinir ağlarının klasik yapay sinir ağlarına göre daha başarılı sonuçlar ürettiği iddia edilmektedir. Bu çalışmada Pi-Sigma yapay sinir ağının Türkiye'ye gelen yabancı turist sayısı zaman serisini öngörmedeki performansı araştırılmıştır. Pi-Sigma yapay sinir ağının performansı, klasik öngörü yöntemleri ve alternatif yapay sinir yöntemleri ile çeşitli istatistik ve Nemenyi testi kullanılarak karşılaştırılmıştır. Elde edilen sonuçlar tablo ve şekiller yardımıyla yorumlanarak sunulmuştur.

Anahtar kelimeler: Pi-Sigma Yapay Sinir Ağları, Zaman Serisi, Turist Sayısı

FORECASTING NUMBER OF TOURISTS WITH PI-SIGMA ARTIFICIAL NEURAL NETWORKS

SUMMARY

Artificial neural networks are powerful alternatives to well-known forecasting methods. It is asserted that high order neural networks produce better forecasting results than classical artificial neural networks. In this study, the forecasting performance of pi-sigma artificial neural networks on number of foreign arrivals to Turkey time series are investigated. The performance of pi-sigma artificial neural networks is compared with classical forecasting methods and alternative artificial neural networks by using various statistics and Nemenyi test. The obtained results presented by tables and figures.

Keywords: Pi-Sigma Artificial Neural Networks, Time Series, Number of Tourists

BÖLÜM 1. GİRİŞ

Turizm; bir ülkeye gezmek, eğlenmek, yeni yerler görmek, tatil yapmak gibi faaliyetler için yapılan seyahatlerdir. Kişiler yıl içerisinde istedikleri zaman diliminde seyahat edebilmektedirler. Turistler hem bu sayede yeni kültürler hem de farklı yaşayışları görmektedirler. Turizm kişiler için yeni şeyler öğrenme bakımından bir kazanç olduğu gibi gelen turistler sayesinde ülkeye giren döviz sayesinde ülkede kazanç sağlamış olmaktadır.

Turizm, hem ülke ekonomisine hem de ülke içerisindeki istihdama büyük oranda katkı sağlamakta olan ve her geçen gün gelişmekte olan bir faktördür. Ülkeler ekonomik dengelerinin sağlanmasında turizmin ülkeye katkısından faydalanmaktadır. Turizm ülkenin gelirlerinde artış, bölgesel ve mali açıdan ülkenin kalkınmasına etki eden önemli bir etken olmuştur.

Türkiye'nin jeopolitik konumu, geçmişten günümüze aktarılan kültürü ve tarihi bakımdan birçok turistik mekâna ev sahibi yapmaktadır. Turistlerin her geçen gün artarak seyahat etmeleri ve Türk kültür ve tarihi mekânlarının reklamının yapılması ülkenin daha çok turist çekmesini sağlamıştır. Türkiye her geçen gün turist sayısının artırılması için gelen turistlere sağlanan imkânları çoğaltmıştır.

Turizm işletmeleri için yıl içerisinde gelecek olan turist sayısının önceden öngörülmesi çok önemlidir. Çünkü işletmeler gelebilecek olan turist sayısına göre ekonomik açıdan imkânlarını ayarlamaktadırlar. Bir işletme gelebilecek olan turist sayısına göre önceden yer ayarlamak, yiyecek ve içecek imkânlarını düzenlemek, yapılacak tur sayısını belirlemek ve çeşitli farklı organizasyonları için düzenleme yapabilmektedir. İşletme yöneticileri için önceden tahmin yapılması hem ekonomik bakımdan doğru karar alınmasına hem de kaynak planlamasında önem arz etmektedir. Turizmdeki gelişmelerin önceden tahmin edilmesi, yöneticilerin doğru karar almalarını kolaylaştırıp (İçöz ve Kozak, 2002), kaynak planlaması konusunda önemli yararlar sağlayacağından (Cho, 2003), yöneticilerin turizm faaliyetlerini planlama süreci daha verimli olabilecektir (Soysal ve Ömür Gönülşen, 2010).

BÖLÜM 2. KAYNAK ARAŞTIRMASI

2.1. Tezin Amacı ve Önemi

Turizm sektörü ülkemizin, yoğun faaliyet gösterdiği ve yüksek beklenti içinde olduğu başlıca sektördür. Ülkemizde her yıl turizm yatırımları artarak devam etmektedir. Sektör için turist sayılarının tahmin edilmesi oldukça önemli olup, gelecek yatırım planlarına ve beklentilerin şekillenmesine yardımcı olmaktadır. Bu tez çalışmasının amacı turist sayılarının öngörülmesinde alternatif öngörü yöntemlerinin ve pi-sigma yapay sinir ağının performansının araştırılmasıdır.

Pi-Sigma yapay sinir ağları (PS-YSA) yüksek dereceden yapay sinir ağları (YSA) modeli olup zaman serisi öngörü problemlerinde başarılı sonuçlar üretebilmektedir. PS-YSA literatürde ilk kez Shin and Ghosh (1991)'de önerilmiştir. PS-YSA 'da girdilerin farklı lineer kombinasyonları ve bunların çarpımı ile elde edilen değerler kritik öneme sahiptir. Mimarisinde hem toplamsal hem de çarpımsal nöron modeli kullanan PS-YSA'nın gizli tabakasındaki ağırlıklar sabit ve "1" değerini almaktadır. Sadece girdi ve gizli tabaka arasındaki ağırlıklarının optimize edilmesine ihtiyaç duyan PS-YSA, çok katmanlı algılayıcıya göre daha az parametre ile çalışmaktadır. Literatürde PS-YSA farklı amaçlarla kullanılmıştır. Nayak ve ark. (2014a, 2014b, 2015 and 2016) PS-YSA'yı sınıflama için, Kanungo ve ark. (2018) PS-YSA ve parçacık sürü optimizasyonunu (PSO) doğrusal olmayan sınıflama problemi için kullanmışlardır. Tavfik ve Liatsis (1997) doğrusal olmayan zaman serisi öngörüsü için PS-YSA'yı kullanmışlardır. Ghazali ve ark. (2006, 2008) PS-YSA'yı kaotik zaman serisi öngörüsü için kullanmışlardır. Husaini ve ark. (2011) PS-YSA ile Batu, Pahat da sıcaklık tahmini elde etmiştir. Nayak (2017) döviz kuru tahmininde PS-YSA'yı kullanmıştır. Akdeniz ve ark. (2018) geri beslemeli PS-YSA'yı zaman serisi öngörüsü için önermiştir. Bas et al. (2018) ise PS-YSA 'yı bulanık zaman serilerinde kullanmışlardır. Literatür taraması sonucunda PS-YSA'nın turist sayısı öngörü problemi için henüz kullanılmadığı görülmektedir. Bu nedenle bu çalışmanın literatüre katkı sağlaması beklenmektedir.

2.2. Literatür Taraması

Yapılan literatür taramasında PS-YSA ile Türkiye turist sayısı tahmini ile ilgili daha önce herhangi bir çalışmanın yapılmadığı görülmüştür. Daha önceki çalışmalardan YSA ile ilgili yapılmış olan çalışmalar şu şekildedir.

ABD, İngiltere, Almanya, Fransa ve Avusturya'dan ülkemize gelen turist sayıları Baldemir ve Bahar (2003) çalışmasında YSA ile tahmin edilmiştir. Güngör ve Çuhadar (2005), turizm verisi için çoklu regresyon modeli ve YSA'yı karşılaştırmışlardır. Çuhadar ve Kayacan (2005) konaklama doluluk oranlarını YSA ile öngörmüşlerdir. Çuhadar vd. (2009) turizm verisi klasik zaman serisi yöntemleri ve YSA ile modellenmiştir. YSA'nın başarılı öngörü performansı ön plana çıkmıştır. Önder ve Hasgöl (2009), Türkiye'ye yurtdışından gelen turist sayılarını YSA ve geleneksel yöntemler ile tahmin etmiştir. Çuhadar (2013) YSA modelleri ile Türkiye turist sayısı zaman serisini tahmin etmiştir.

Law (1998), Uysal ve El Roubi (1999), Law (2000), Burger vd. (2001), Cho (2003), Fernandes vd.(2008), Chang-Jui vd., (2011), Kuan-Yu, Chen (2011), Chun vd. (2012), Chang ve Tian (2013) ve Yang vd., (2013) çalışmaları da yurtdışı ülkelerdeki turist sayısı zaman serilerinin öngörülmesinde YSA modellerini kullanan çalışmalardır.

BÖLÜM 3. MATERYAL VE YÖNTEM

3.1. Yapay Sinir Ağları (YSA)

Bilimin her geçen yıl gelişen teknolojiye ayak uydurması için birçok yeni yöntemler geliştirilmiştir. Yeni bir yöntem bulmak ya da bulunan yöntemi geliştirmek için farklı şekilde çalışmalar yapılmış. Bilim insanları gerek doğadan gerekse de insan yapısından yola çıkarak yeni yöntemler geliştirmişlerdir. Bu yöntemlerden biriside insan beynindeki sinir hücresinin çalışma prensibinden esinlenerek oluşturulan ve matematiksel olan bilgisayar algoritmasıdır. Bu algoritmaya biyolojik sinir hücresinden esinlendiği için yapay sinir ağı (YSA) denmektedir.

Yapay sinir ağı (YSA), matematiksel kaynaklı bilgisayar algoritması olup beyin deki sinir hücresinin işlem yapabilme yeteneğinin kopyasıdır. Beynin öğrenme, algılama, duylara karşı tepki verme, yorum ve karar yapabilmesi için sinir hücrelerine ihtiyacı vardır. Bu sinir hücrelerinin birbiri ile bağıntı ya da bağıntılar kurması sonucu sinir ağları karşılaşılan probleme göre cevap üretmektedir. YSA'lar da beyin bu özelliklerinden yola çıkarak karşılaşılan problemlere hem daha kısa sürede hem de en az maliyetle ulaşabilen algoritmalarıdır.

YSA'lar ilk olarak 1943 yılında bir nörofizyolog olan Warren Sturgis McCulloch ve mantıkçı olan Walter Harry Pitts tarafından yazılan 'sinirsel aktiviteler için fikirlerin mantıklı bir hesabı' adlı makalede ortaya atılmıştır. (URL 1, 2016). Bu makale beyin teorisinin gelişimi hakkında büyük önem taşımaktadır. YSA teorisinin başlangıcı olarak kabul edilmektedir. Günümüze kadar olan çalışmaların genelinde de bu çalışma kaynak olarak gösterilmiştir. Bu makale de temel alınan işlem ise beyin içindeki bir nöronun işleyiş şeklinin, dış dünya hakkındaki bir duruma karşı nasıl tepki verdiği idi. YSA biyolojik sinir ağlarını taklit eden sentetik yapılardır (Eğrioğlu, 2009). Bu yüzden beyinde ki nöronların dış dünyadan almış olduğu bilgileri bir takım işlemler sonucunda probleme karşı çıktı üretmesini taklit etmişlerdir. YSA'lar dış dünyadan gelen olaylar karşısında nasıl bir tepki vermesi gerektiğini belirleyen belli karakteristik özelliklere sahiptir. Bunlar;

- Doğrusal olmama
- Öğrenme
- Genelleme
- Uyarlanabilirlik
- Hata toleransı
- Donanım ve hız
- Analiz ve tasarım kolaylığı

3.1.1. Yapay Sinir Ağları Avantaj ve Dezavantajları

Birçok farklı alanda ve birçok farklı şekilde kullanılmakta olan YSA'ların avantajlı ve dezavantajlı taraflarını şu şekilde sıralayabiliriz. Genel olarak belli başlı avantajları şunlardır.

- YSA'lar algoritmik yapıda olduğu için makine öğrenmesi gerçekleştirebilirler.
- Bilgi işleme ve veri saklama yöntemleri geleneksel algoritmalardan farklı şekildedir.
- Bilgiler ağın bütün kısımlarında saklanır. Tek bir kısımda toplanmaz.
- Daha önce ki benzer örnekler sayesinde yeni örneklerle çözüm üretirler.
- Görmediği örnekler için benzer örneklerden faydalanıp çözüm bulurlar.
- Örüntü tamamlama yapabilirler.
- Örüntü (pattern) ilişkilendirme ve sınıflandırma yapabilirler.
- Eksik bilgi ile çalışabilirler.
- Kendi kendine öğrenebilme ve organize etme yeteneğine sahiptirler.
- Algılamaya yönelik olaylarda kullanılabilirler.

Birçok avantajlı tarafı olmasına rağmen bazı konularda tam olarak açıklanamadığı için dezavantajlı yanları da mevcuttur. Bunlar;

- Ağın davranışlarının açıklanamaması
- Uygun ağ yapısının belirlenmesinde belli bir kuralın olmayışı.

- Parametre değerlerinin belirlenmesinde belli bir kuralın olmayışı.
- Öğrenilecek problemin önceden ağa gösterilmesi.
- Ağın eğitiminin ne zaman bitirilmesi gerektiğine ilişkin belli bir yöntemin olmayışı.
- Donanımın bağımlı olması.
- Paralel işlem gücüne sahip işlemcilere ihtiyaç duymasıdır.

3.1.2. YSA Bileşenleri

YSA'ların avantajlı taraflarına rağmen karşılaştığı problemin çözümü için uygun ağ mimarisinin belirlenmesi bu yapıdaki gizli katman sayısının ve gizli katman ile çıktı arasında kullanılacak olan aktivasyon (transfer) fonksiyonunun seçilmesi ve hangi modelin kullanılması gerektiği ağın performansı açısından büyük önem taşımaktadır.

3.1.3. YSA Mimari Yapısı

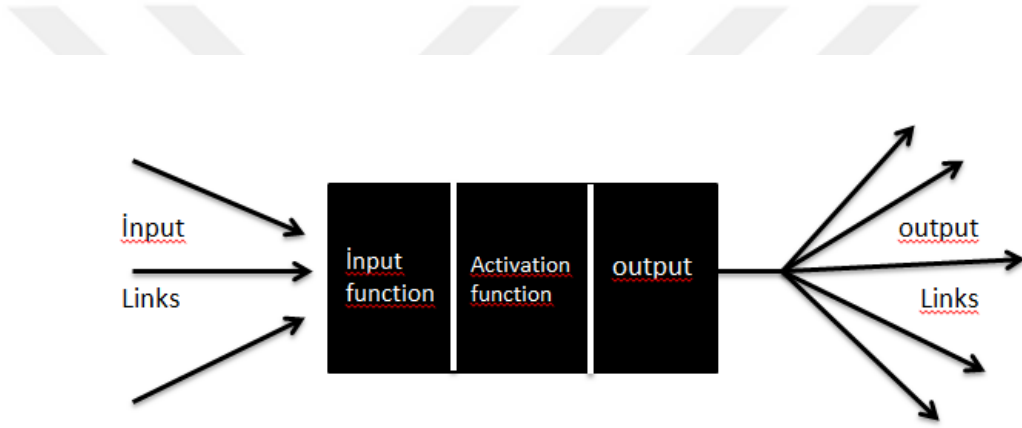
YSA'ların bilgiyi öğrenme ve türetilen bilginin ağ yapısından çıktı olarak çıkması iki şekilde meydana gelmektedir. Bunlar ileri beslemeli ve geri beslemeli ağlardır. Bütün YSA'lar nöronların bağlantı ve birbiri ile bilgiyi iletmesi sayesinde çıktı üretirler. Öncelikle genel olarak YSA'ların mimari yapısını inceleyelim.

Nöronların birbiri ile bağlantı kurması sonucu oluşan YSA'lar üç temel katmandan oluşmaktadır. Bunlar girdi katmanı, çıktı katmanı ve bu iki katman arasındaki temel işlemlerin yapıldığı ve ağ modeline göre sayısının değişiklik göstermiş olduğu gizli (ara)katmandır.

Nöronların bir araya gelmesi ile oluşan YSA'ların katmanlarını şu şekilde açıklayabiliriz.

Girdi katmanı; YSA'ya dış dünyadan bilgilerin geldiği katmandır. Bu katmanda dış dünyadan gelecek giriş sayısı kadar hücrenin bulunmasına rağmen genelde girdiler herhangi bir işleme uğramadan alt katmanlara iletilmektedir.

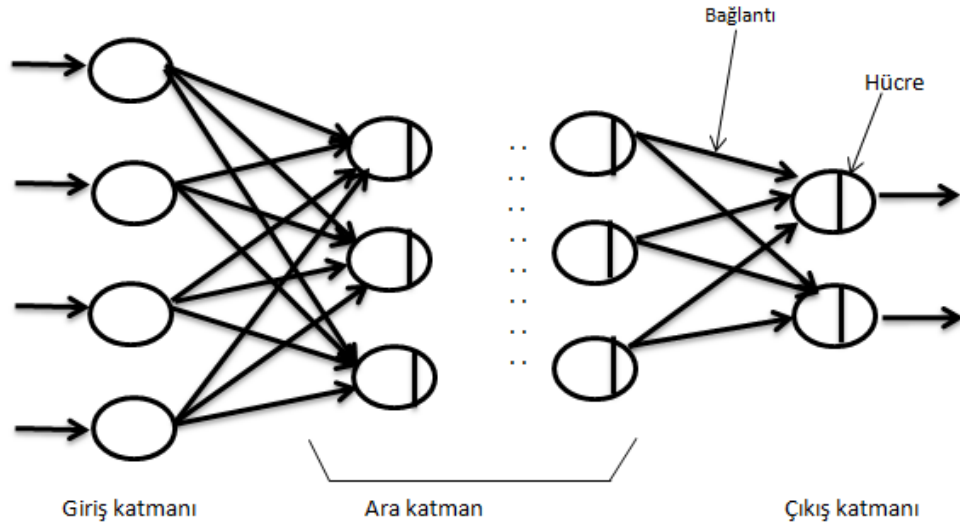
Gizli (ara) katmanı; ağa gelen bilgilerin girdi katmanından sonra ki işlemlerin yapılması gelmiş olduğu katmandır. Bir ağda ki gizli katman sayısı problemin performansını etkileyen kısımdır. Ağda kullanılan gizli katman sayısı sabit değil probleme göre değişmektedir. Gizli katmanların ve bu katmanlardaki nöron sayısının artması problemin hesaplanması karmaşıklığına ve süresinin artmasına rağmen yapay sinir ağının daha karmaşık problemlerin çözümünde de kullanılabilmesini sağlar.



Şekil 2.1. Yapay sinir ağ yapısı

3.1.4. İleri Beslemeli Yapay Sinir Ağları

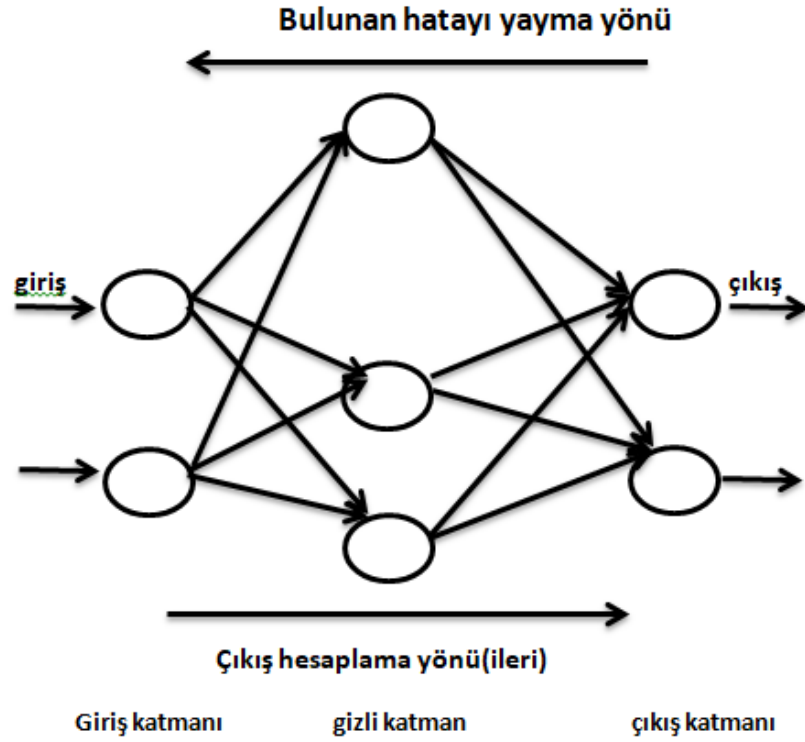
İleri beslemeli yapay sinir ağlarında bilgi girdi katmanından çıktı katmanına doğru düzenli bir şekilde ve tek taraflı olarak iletir. Nöronlar birbirleri ile bağlantı kurarken bilgiyi kendi içlerine aktaramazlar bu da ağın hızlı çıktı oluşturmalarını sağlamaktadır.



Şekil 2.2. İleri Beslemeli Yapay sinir ağı (Öztemel, 2006)

3.1.5. Geri Beslemeli Yapay Sinir Ağları

Geri beslemeli sinir ağlarında en az bir hücrenin çıktısı kendi içerisine veya kendinden önceki nörona girdi olarak verilmektedir. Geri besleme, bir katmandaki hücreler arasında olduğu gibi katmanlar arasındaki hücreler arasında da olabilir. Bu yapısı ile geri beslemeli YSA, doğrusal olmayan dinamik bir davranış gösterir. Dolayısıyla, geri beslemenin yapılış şekline göre farklı yapıda ve davranışta geri beslemeli YSA yapıları elde edilebilir. Katmanlar arasında bilginin aktarımı düzenli olmadığı için geri beslemeli sinir ağları karmaşık bir yapıya sahiptir. Ancak dinamik hafıza sebebiyle tahmin ve sınıflandırma problemlerinde başarılı sonuçlar elde etmişlerdir.



Şekil 2.3. Geri Beslemeli Yapay sinir ağı (Kabalcı, 2015)

3.1.6. Aktivasyon Fonksiyonu

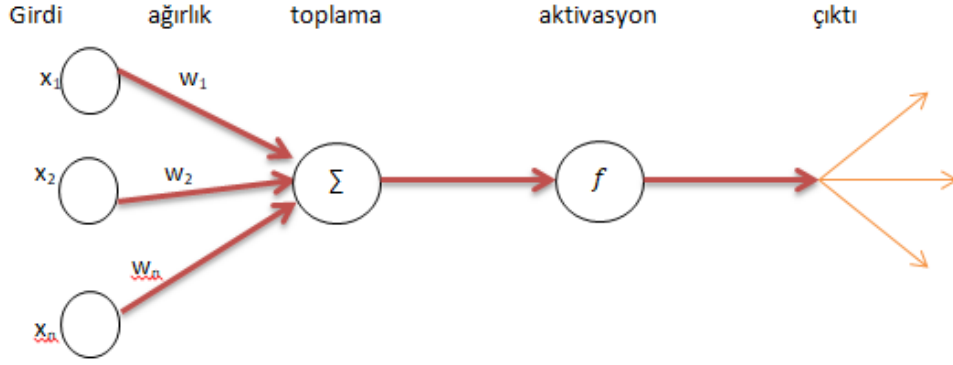
YSA'ların doğrusallık göstermeme özelliğinden dolayı aktivasyon fonksiyonları da genellikle doğrusallık göstermez. Aktivasyon fonksiyonunun işlevi ağı performansında önemli bir etkidir. Aktivasyon fonksiyonu seçimi sırasında dikkate alınması göz önünde bulundurulan bir diğer konu ise fonksiyonun türevinin kolay hesap edilebilir olmasıdır. Aktivasyon fonksiyonunun ne kadar doğru ve türevinin de ne kadar kolay hesaplanabilir olması hücre çıktısına yansıyacak ve bu da hücrenin performansını etkileyecektir. Birden fazla aktivasyon fonksiyonu mevcuttur. Bazı aktivasyon fonksiyonları aşağıda tablo olarak verilmiştir.

Tablo 2.1. Aktivasyon fonksiyonları (Çayıroğlu,2015)

Doğrusal (Lineer)Aktivasyon Fonksiyonu	$F(Net)=A*Net$ (A sabit bir sayı)
Adım (Step) Aktivasyon Fonksiyonu	$F(Net) = \begin{cases} 1 & \text{if } Net > EşikDeğer \\ 0 & \text{if } Net \leq Eşik Değer \end{cases}$
Sigmoid Aktivasyon Fonksiyonu	$F(Net) = \frac{1}{1 + e^{-Net}}$
Tanjant Hiperbolik Aktivasyon Fonksiyonu	$F(Net) = \frac{e^{Net} + e^{-Net}}{e^{Net} - e^{-Net}}$
Eşik Değer Fonksiyonu	$F(Net) = \begin{cases} 0 & \text{if } Net \leq 0 \\ Net & \text{if } 0 < Net < 1 \\ 1 & \text{if } Net \geq 1 \end{cases}$
Sinüs Aktivasyon Fonksiyonu	$F(Net) = Sin(Net)$

3.1.7. Yapay Nöron Modelleri

YSA'lar birbiri ile bağlantı şeklinde olan nöronlardan oluşmaktadır. Bu bağlantılar da ki katman sayısına göre tek katmanlı ve çok katmanlı olmak üzere ikiye ayrılırlar. Bu nöronlarının işleyişi biyolojik sinir hücresinde olduğu gibi bilgiyi dış dünyadan veya bir önceki nörondan alıp ya çıktı üretmektir ya da çıktı üretilmesi için diğer nörona iletilmesini sağlamaktır. Bir nöron bilgiyi diğer nörona aktarması ya da çıktı üretmesi için beş kısımdan oluşan bir sisteme sahiptir. Bu kısımlar girdi, ağırlık, toplama fonksiyonu, aktivasyon (transfer) fonksiyonu ve çıktıdır.



Şekil 2.4. Yapay nöron yapısı

Girdi; nörona dış dünyadan ya da bir önceki nörondan gelen bilgilerin tümüdür. Ağırlıklar; nörona gelen bilgilerin sinirdeki önemini gösteren ve sinir üzerindeki etkisini tayin eden katsayılarıdır. Nörona gelen her bir bilginin bir ağırlığı mevcuttur. Ancak bu ağırlık değerin büyük ya da küçük olması önemli ya da önemsiz olması anlamına gelmemektedir. Birleştirme fonksiyonu; nörona gelen bilgiler almış oldukları ağırlıklar ile önce çarpılıp sonra toplanması ile hücrenin net girdi değerini hesaplayan fonksiyondur. Bazı durumlarda gelen verilerin değerleri dikkate alınırken bazı durumlarda ise gelen verilerin sayısı önemli olmaktadır. Genel olarak toplama fonksiyonu deneme yanılma yolu ile belirlenir. Hangi toplama fonksiyonunu kullanmaya karar vermek kişiye bağlıdır. Bazı toplama fonksiyonları şunlardır.

Aktivasyon (transfer) fonksiyonu; toplama fonksiyonundan gelen net bilgiyi işleyerek hücrenin bu bilgiler doğrultusunda oluşacak çıktısını belirleyen kısımdır. Aktivasyon fonksiyonu genellikle doğrusallık göstermez.

Çıktı; aktivasyon (transfer) fonksiyonu oluşan değerdir. Bu değer ya dış dünya aktarılır ya da diğer hücreye aktarılır.

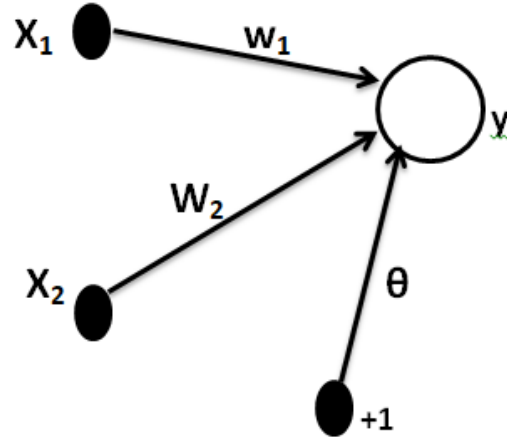
Tablo 2.2. Bazı toplama fonksiyonları (Çayıroğlu, 2015)

Toplam $Net = \sum_{i=1}^N X_i * W_i$	Ağırlık değerleri girdiler ile çarpılır ve bulunan değerler birbirleriyle toplanarak Net girdi hesaplanır.
Çarpım $Net = \prod_{i=1}^N X_i * W_i$	Ağırlık değerleri girdiler ile çarpılır ve daha sonra bulunan değerler birbirleriyle çarpılarak Net girdi hesaplanır.
Maksimum $Net = \text{Max}(X_i * W_i)$	n adet girdi içinden ağırlıklar girdilerle çarpıldıktan sonra içlerinden en büyüğü Net girdi olarak kabul edilir.
Minimum $Net = \text{Min}(X_i * W_i)$	n adet girdi içinden ağırlıklar girdilerle çarpıldıktan sonra içlerinden en küçüğü Net girdi olarak kabul edilir.
Çoğunluk $Net = \sum_{i=1}^N \text{Sgn}(X_i * W_i)$	n adet girdi içinden girdilerle ağırlıklar çarpıldıktan sonra pozitif ile negatif olanların sayısı bulunur. Büyük olan sayı hücrenin net girdisi olarak kabul edilir.
Kümülatif Toplam $Net = Net(eski) + \sum_{i=1}^N X_i * W_i$	Hücreye gelen bilgiler ağırlıklı olarak toplanır. Daha önce hücreye gelen bilgilere yeni hesaplanan girdi değerleri eklenerek hücrenin net girdisi hesaplanır.

A

3.1.8. Tek Katmanlı Yapay Sinir Ağları

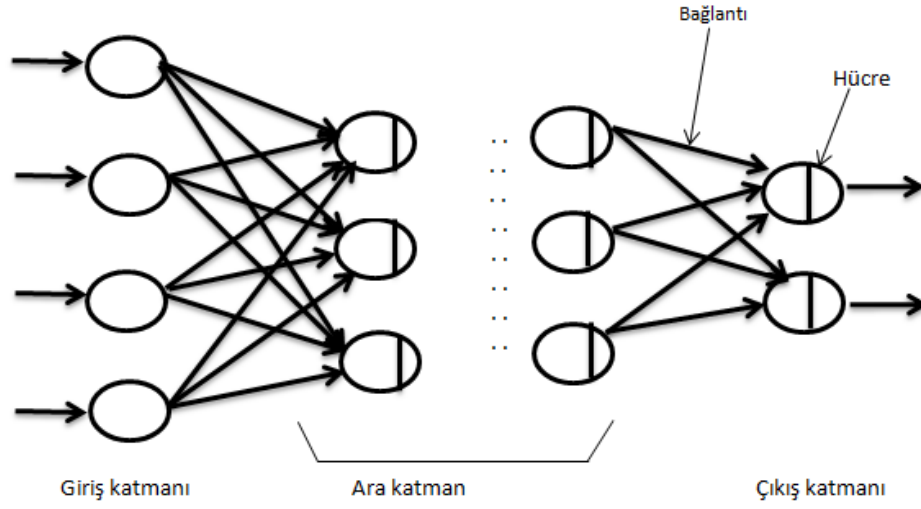
Tek katmanlı yapay sinir ağları gizli katmanı olmayan ve sadece girdi ve çıktı katmanından oluşan bir yapıya sahiptir. En basit şekilde ise iki girdi ve tek bir çıktıdan oluşmaktadır. Bu modelde girdi katmanı bilgiye direk olarak çıktı katmanına iletir.



Şekil 2.5.İki Girdi ve Bir Çıktıdan Oluşan Tek Katmanlı YSA Modeli (Öztemel,2006)

3.1.9. Çok Katmanlı Yapay Sinir Ağları

Çok katmanlı sinir ağları bir ya da daha fazla gizli katmanı olan ileri beslemeli sinir ağlarıdır. Genelde bu ağ bir girdi katmanı, en az bir gizli katman ve bir de çıkış katmanından oluşur. Gizli katmanlara ihtiyaç duyulmasının sebebi girdi katmanlarından gelen genellikle işlenmemiş sinyallerin özelliklerini belirlemek, ağırlıklandırmak ve sonuçları çıktı katmanına yönlendirmektir.



Şekil 2.6. Çok Katmanlı Yapay Sinir Ağ Modeli (Öztemel,2006)

3.1.10. Eğitim Algoritmaları

YSA'nın bilgiyi öğrenmesi ağın problemi çözmesinde kolaylık sağlayacaktır. Biyolojik sinir hücresinde bilgiler Sinapsis denen uç noktalar vasıtasıyla dış dünyaya veya diğer nörona aktarılmaktadır. YSA da ise bilgi ağırlıklar vasıtasıyla aktarılır. YSA'nın eğitilmesinin amacı hedeflenen çıktı değeri ile ağdan çıkan çıktı değerini arasındaki hata payını minimum seviyeye indirebilmektir. YSA'lar eğitimi için şu adımlar izlenilir.

- Örneklerin Toplanması
- Ağın Topolojik Yapısının Belirlenmesi
- Öğrenme Parametrelerinin Tespit Edilmesi
- Ağırlıkların Başlangıç Değerlerinin Atanması
- Öğrenme Setinden Örneklerin Seçilmesi Ve Ağa Gösterilmesi
- Öğrenme Sırasında İleri Hesaplamaların Yapılması
- Gerçekleşen Çıktının Beklenen Çıktı İle Karşılaştırılması
- Ağırlıkların Değiştirilmesi
- Öğrenmenin Tamamlanması

En sık kullanılan eğitim algoritması geri yayılım öğrenme algoritması ve parçacık sürü optimizasyonudur (PSO). Bunların yanı sıra şu algoritmalarda literatürde mevcuttur.

Esnek Yayılım Algoritması, Delta Bar Delta Algoritması, Geliştirilmiş Delta Bar Delta Algoritması, Hızlı Yayılım Algoritması, Levenberg-MarquardtMetodu Eşleştirmeli Eğitim Algoritmaları ve Genetik Algoritmadır.

3.1.11. Geri Yayılım Öğrenme Algoritması

Rumelhart, Hinton ve Williams tarafından geliştirilmiş geri yayılım algoritması, Literatürde en yaygın olarak kullanılan öğrenme algoritmasıdır. Geri yayımlı öğrenme algoritması iki aşamadan oluşmaktadır. İlk aşamada ağın öğrenme yapabilmesi için eğitim girdileri ağı sunulur. Daha sonra ağ bu girdilerden oluşan çıktıyı üretir. Eğer çıktı katmanından çıkan sonuç ile hedeflenen sonuç farklı ise ya da beklenen değeri oluşmamış ise hata hesaplanır ve ağ bu hatayı çıktı katmanından girdi katmanına geriye doğru aktarılır. Bu aşamada bilgilerin ağırlıkları hata oranına göre güncellenmektedir.

Geri yayılım öğrenme algoritması delta kuralını kullanarak ağırlıkların farkını hesaplar. YSA'nın eğitim ve test verisi adımları geri yayılım öğrenme algoritmasının aşağıdaki temel formülleri ile hesaplanır.

$$\Delta\omega_{ji}(t) = T\delta_j x_i + \alpha\Delta\omega_{ji}(t - 1) \quad (2.1)$$

Burada T öğrenme katsayısı, α momentum katsayısı ve δ_j ara veya çıktı katmanındaki herhangi bir j nöronuna ait bir faktördür. Çıktı katmanındaki bu faktör şu şekilde verilir.

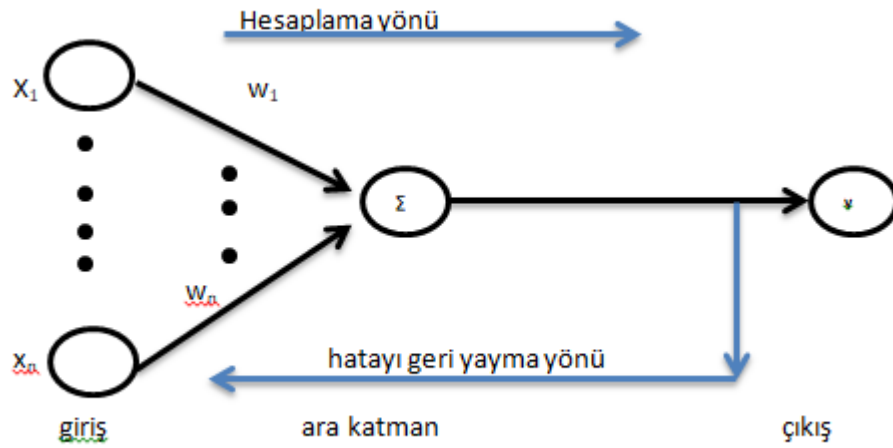
$$\delta_j = \frac{\partial f}{\partial net_j} (y_j^{(t)} - y_j) \quad (2.2)$$

Burada;

$net_j = \sum x_j * \omega_{ji}$ ve $y_j^{(t)}$ ise nöronun hedef çıktı değeridir. Gizli katmandaki nöronlar için ise bu faktör şu şekildedir.

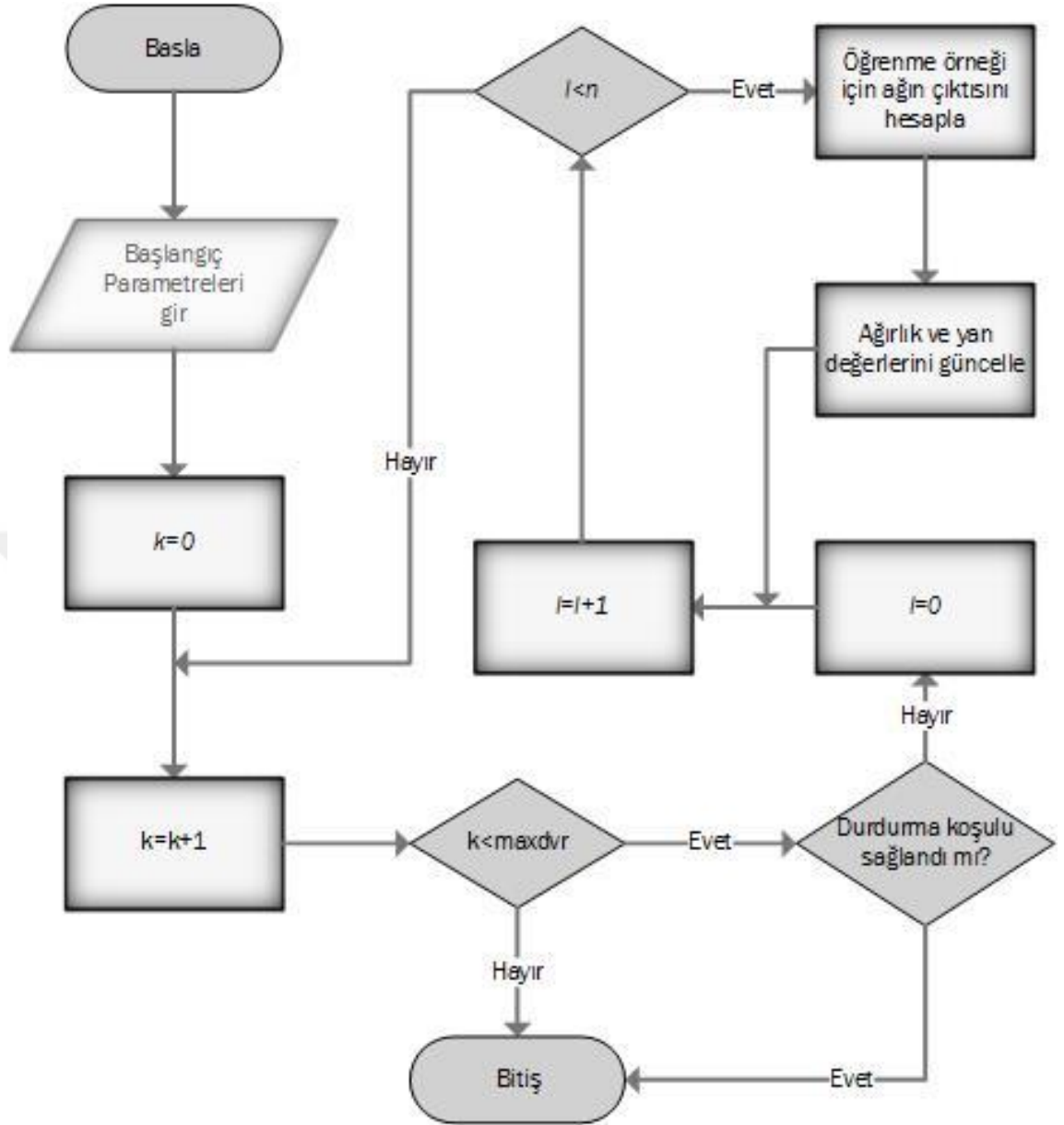
$$\delta_j = \left(\frac{\partial f}{\partial net_j} \right) \sum \omega_{qi} \delta_q \quad (2.3)$$

Gizli katmandaki nöronlar için çıkış olmadığından denklem (2) yerine (3) kullanılır. Bu denkleme bağlı olarak çıktı katmanından başlayarak δ_j faktörü bütün katmanlardaki nöronlar için hesaplanır. Sonra denklem (1)' deki formül ile bütün bağlantılar için ağırlıklar güncelleştirilir.



Şekil 2.7.Geri Beslemeli Yapay Sinir Ağı (Kabalıcı, 2015)

Aşağıda geri beslemeli yapay sinir ağının eğitim algoritması akış şeması verilmiştir. Akış şeması ile ağın eğitilmesi hataların geriye doğru dağıtılması adım adım gösterilmektedir.



Şekil 2.8. Çok Katmanlı Yapay Sinir Ağı'nın Geri Beslemeli Öğrenme Modeli Akış Şeması (Egrioglu vd., 2019)

3.2. Parçacık Sürü Optimizasyonu (PSO)

Doğada yaşayan bazı canlılar yiyecek bulmak ve yaşamak için sürüler halinde hareket etmektedir. Her ne kadar bu canlıların yaşayışı rastgele olarak gözüксе de her birinin sürü içindeki görevi farklıdır. Bu sürüler halinde hareket eden canlılardan birisi de kuşlardır. Kuşların sürü halindeki hareketlerinin inceleyen Kennedy ve Eberhart 1995

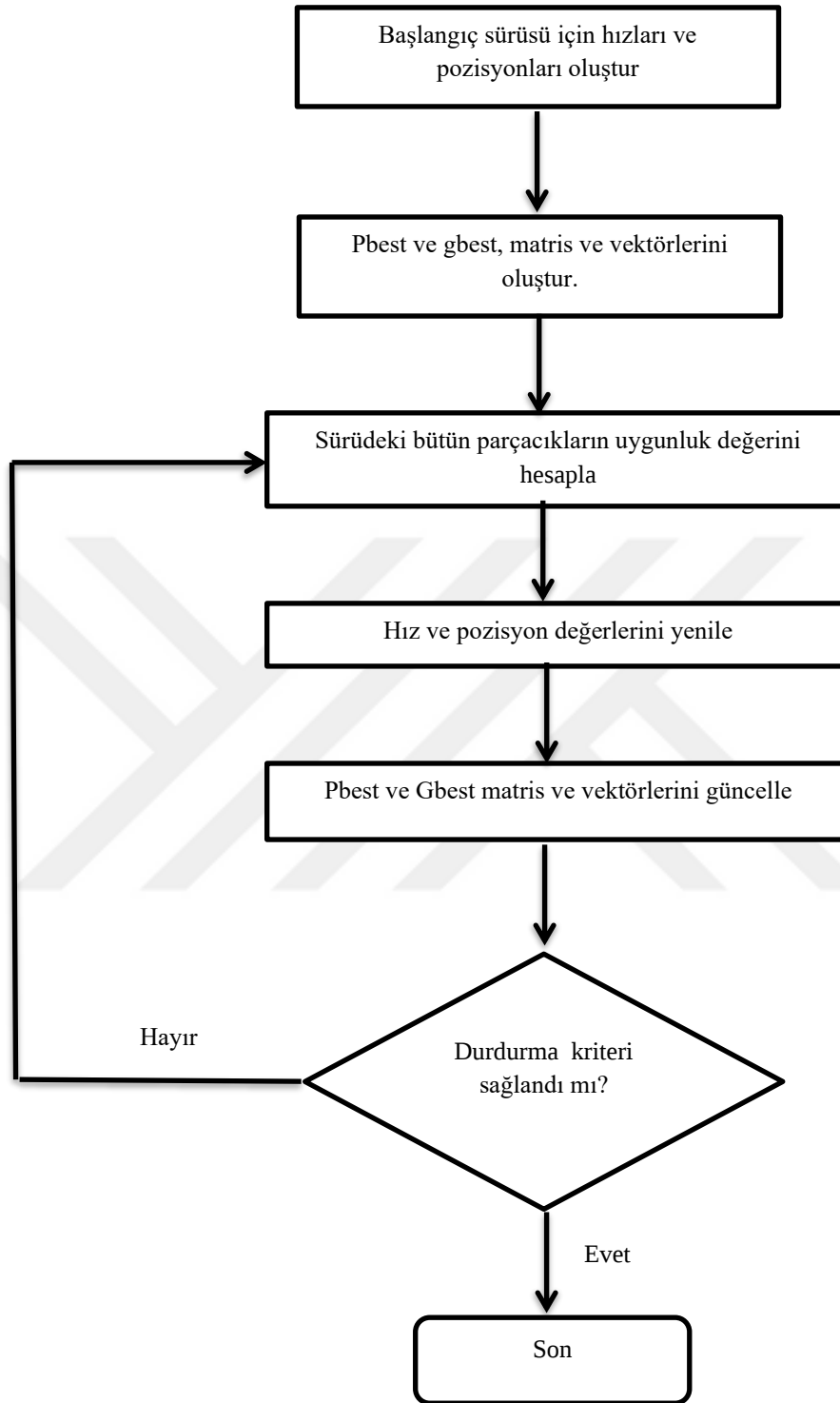
yılında parçacık sürü optimizasyonu (PSO) olarak sezgisel bir algoritma geliştirmişlerdir. Burada sürüdeki her kuş birer parçacığı oluşturmaktadır ve bütün parçacıkların oluşturduğu sürüye popülasyon denmektedir. PSO algoritmasında parametre sayısı oldukça az olup yönteminin kullanımını daha basit hale getirmektedir (Zhao 2005). Bu nedenle literatürde nümerik optimizasyon için sık kullanılan bir yöntemdir.

3.2.1. Parçacık Sürü Optimizasyonu (PSO) Aşamaları

PSO’da sürü içindeki her bir parçacık ilk başta rastgele olarak değer almaktadır. Daha sonra her bir iterasyon sonucu bu almış oldukları değerler güncellenmekte ve en iyi değere sahip olan parçacığı bulmaktır. ilk başta pbest’ini yani parçacıklar içindeki en iyi değere sahip olan parçacık belirlenir. Daha sonra ise sürünün içindeki en iyi değere sahip parçacık belirlenir bu da gbest olarak adlandırılmaktadır. En iyi değere sahip olan parçacık bulunduğu yeri diğer parçacıklarla paylaşır. d adet parametreden oluşan ve n adet parçacık için popülasyon matrisi aşağıdaki gibi oluşmaktadır.

$$\begin{bmatrix} x_{11} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nd} \end{bmatrix} \quad (2.4)$$

Burada i . parçacık $x_i = [x_{i1}, x_{i2}, \dots, x_{id}]$ olarak gösterilir. Her iterasyon da her bir parçacık iki en iyi değere göre güncellenir. Bunlardan ilki bir parçacığın o ana kadar bulmuş olduğu en iyi uygunluk değeridir ve bu değer daha sonra kullanılmak üzere hafızada tutulmaktadır. Bu değere *pbest* (parçacığın en iyi uygunluk değeri) denmektedir. *pbest* değerinin i . parçacığın pozisyonları $pbest_i = [p_{i1}, p_{i2}, \dots, p_{id}]$ olarak gösterilir. Diğer en iyi değer ise popülasyondaki en iyi parçacık tarafından elde edilmiş en iyi uygunluk değeridir. Bu değer popülasyon için en iyi global değerdir ve *gbest* olarak ifade edilir ve $gbest = [p_1, p_2, \dots, p_d]$ olarak gösterilir. Her bir parçacığın hızı $v_i = [v_{i1}, v_{i2}, \dots, v_{id}]$ olarak ifade edilir. En iyi uygunluk değerleri ve popülasyonun hızı bulunduktan sonra parçacık hızı ve konumu eşitlik (2.5) ve (2.6) ile güncellenir. v_i = Parçacığın Değişim Hızı, c_1, c_2 = Sabit Değerler,



Şekil 2.9. Parçacık sürü optimizasyonu (PSO) akış diyagramı

$x_i=i$. Parçacığın Konumu $rand_1, rand_2$ =Rastgele Üretilen Değerler, $pbest$ =Yerel En İyi Değer, $gbest$ = Global En İyi Değer, k =İterasyon Sayısı

$$v_i^{k+1} = v_i^k + c_1 rand_1^k (pbest_i^k - x_i^k) + c_2 rand_2^k (gbest_i^k - x_i^k) \quad (2.5)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (2.6)$$

3.3. Pi-Sigma Yapay Sinir Ağları (PS-YSA)

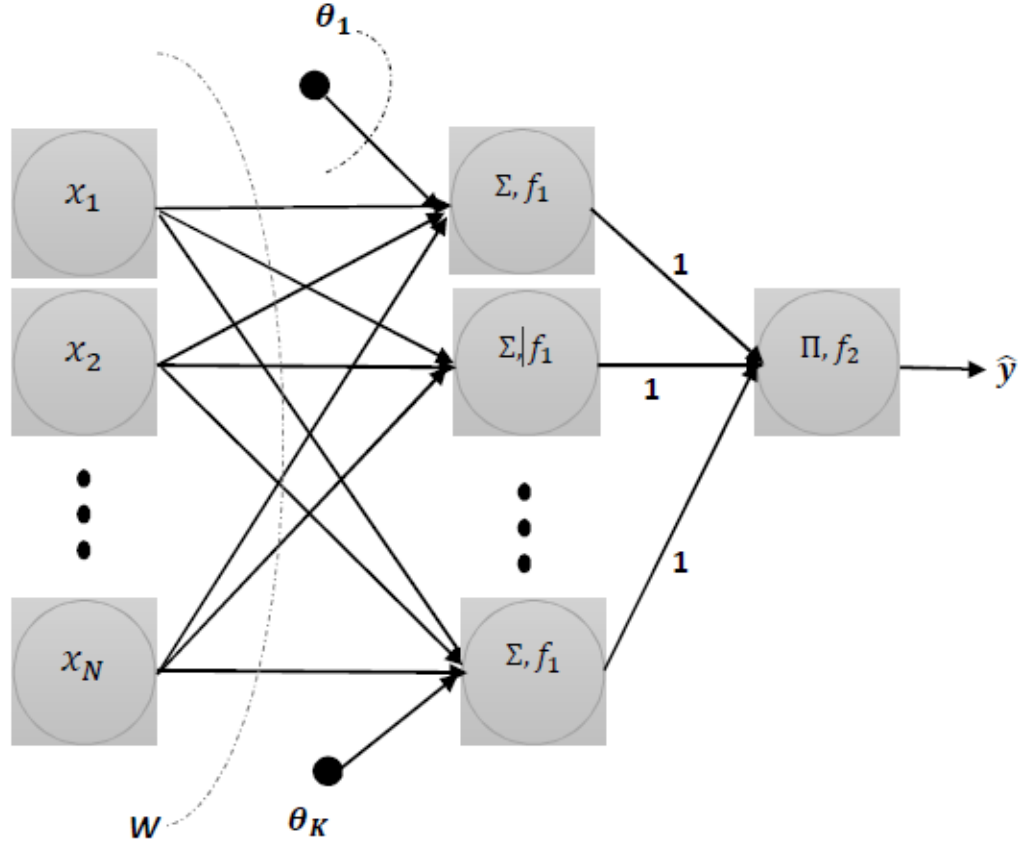
Pi-Sigma yapay sinir ağları (PS-YSA) ilk olarak 1991 yılında Ghosh and Shin tarafından önerilmiştir. PS-YSA da girdilerin farklı doğrusal kombinasyonlarının çarpımı ağırlık çıkırtısını oluşturmaktadır. Lineer kombinasyonların sayısı PS-YSA'nın derecesidir. PS-YSA'nın derecesinin artırılması girdi ile çıktı arasındaki ilişkiyi tanımlayan fonksiyonun daha fazla parametreye bağlı ve daha karmaşık bir yapıda olmasına neden olur. Bu durum daha iyi tahmin sonuçları elde edilmesini sağlar. Ancak fazla parametre ile çalışılması aşırı tahmin (overfitting) problemine ve eğitim algoritması için daha fazla hesaplama zamanına ihtiyaç duyulmasına sebep olur. Bu nedenle PS-YSA için çapraz geçerlilik yöntemleri kullanılarak, ağırlık ezberlemeden öğrenmesine dikkat edilmelidir. K dereceden N girdili bir PS-YSA'nın mimari yapısı Şekil 10'da verilmiştir.

Girdi birimlerinin doğrusal kombinasyonları $\omega_{ij} (i = 1, 2, \dots, N, j = 1, 2, \dots, K)$ ağırlıkları ve $\theta_j (j = 1, 2, \dots, K)$ yanları ile elde edilmektedir. ω_{ij} i.girdiden j.tabaka birimine giden ağırlığı, θ_j ise j.gizli tabaka birimi için yan değerini göstermektedir. Gizli tabaka birim sayısı kadar doğrusal kombinasyon, doğrusal aktivasyon fonksiyonundan geçerek gizli tabakaların çıktıları oluşturur. h_j j. gizli tabaka biriminin çıktısını gösterir ve aşağıdaki formül ile hesaplanmaktadır.

$$h_j = f_1 \left(\sum_{i=1}^N \omega_{ij} x_i + \theta_j \right) , j= 1, 2, \dots, K \quad (3.1)$$

Burada $f_1(x) = x$ şeklinde doğrusal aktivasyon fonksiyonunu göstermektedir. Ağırlık çıkırtısı ise $f_2(x) = \frac{1}{1+\exp(-x)}$ lojistik aktivasyon fonksiyonu olmak üzere aşağıdaki formül ile hesaplanır.

$$\hat{y} = f_2(\prod_{j=1}^K h_j) = \frac{1}{1+\exp(-\prod_{j=1}^K h_j)} \quad (3.2)$$



Şekil 2.10. Pi-Sigma Yapay Sinir Ağı (PS-YSA) Mimari Yapısı

3.3.1. PS-YSA'nın PSO ile Eğitim Algoritması

Pi- Sigma Yapay Sinir Ağı'nın eğitimi için kullanılan algoritmalarından biri de parçacık sürü optimizasyon (PSO) algoritmasıdır. Aşağıda eğitim algoritmasının adımları verilmiştir.

Adım 1. Her bir parçacığın pozisyonu rastgele olarak üretilir ve aşağıda gösterilen P_k vektöründe saklanır.

$$P_k = \{p_{k,1}, p_{k,2}, \dots, p_{k,d}\}, \quad k = 1, 2, \dots, pn \quad (3.3)$$

$p_{k,i} (i = 1, 2, \dots, d) = k$. Parçacığın i 'nci Pozisyonunu, $pn =$ Parçacık Sayısını,
 $d =$ Parçacıkların Pozisyon Sayısı

Parçacıkların her biri çarpımsal nöron model için bir ağırlık ve yan değerleri seti olmaktadır. Bir parçacığın pozisyonları PS-YSA'nın ağırlık ve yan değerlerinde oluşur ve aşağıdaki tabloda gösterilmiştir. Parçacığın pozisyon sayısı PS-YSA'nın derecesine ve girdi sayısına bağlı olarak $N \times K + K$ olmaktadır.

Tablo 3.1.PS-YSA'nın PSO ile eğitiminde bir parçacığın pozisyonları

Pozisyonlar	1	...	$N \times K$	$N \times K + 1$	...	$N \times K + K$
Ağırlık ve Yanlar	W_{11}	...	W_{NK}	θ_1	...	θ_K

Adım 2. Parçacıkların her bir pozisyonu için bir hız değeri rastgele üretilir ve V_k vektöründe saklanır.

$$V_k = \{v_{k,1}, \dots, v_{k,d}\}, \quad k = 1, 2, \dots, pn \quad (3.4)$$

$V_k = k$. Parçacığın Hızı

Adım 3. Her bir parçacık için amaç fonksiyonu değeri hesaplanarak bu değerlere dayalı olarak $Pbest$ ve $Gbest$ belirlenir.

$$Pbest_k = \{pb_{k,1}, \dots, pb_{k,d}\}, \quad k = 1, 2, \dots, pn \quad (3.5)$$

$Pbest =$ Yerel En İyi Değer

$$Gbest = \{P_{g,1}, \dots, P_{g,d}\} \quad (3.6)$$

$Gbest =$ Global En İyi Değer

Burada $Pbest_k$ 'nci parçacığın geçerli iterasyona kadar bulunduğu en iyi pozisyonların vektörü ve $Gbest$ ise sürünün geçerli iterasyona kadar bulunduğu en iyi

pozisyonun vektörüdür. Bu adımda amaç fonksiyonu olarak hedef değerler ve ağırlık çıktıları arasında hesaplanan bir hata kareler ortalaması kullanılabilir.

Adım 4. c_1 ve c_2 bilişsel ve sosyal katsayılar, w ise eylemsizlik ağırlığı parametresi olsun. $(c_{1i}, c_{1f}), (c_{2i}, c_{2f})$, ve (w_1, w_2) değerleri parametreler için başlangıç ve bitiş değerleri olsun. Her bir iterasyonda bu parametreler aşağıdaki formüller ile hesaplanır.

$$c_1(t) = (c_{1f} - c_{1i}) \frac{t}{maxt} + c_{1i} \quad (3.7)$$

$$c_2(t) = (c_{2f} - c_{2i}) \frac{t}{maxt} + c_{2i} \quad (3.8)$$

$$w(t) = (w_2 - w_1) \frac{maxt-t}{maxt} + w_1 \quad (3.9)$$

Adım 5. Hız ve pozisyon değerleri aşağıdaki formüller ile güncellenir.

$$v_{i,j}^{t+1} = [w \times v_{i,j}^t + c_1 \times rand_1 \times (pb_{i,j} - p_{i,j}) + c_2 \times rand_2 \times (p_{g,j} - p_{i,j})] \quad (3.10)$$

$V_{i,j}=i,j$. Parçacık hızı, t =İterasyon Sayısı, w =Eylemsizlik parametresi, c_1, c_2 =sabit değer, $rand_1, rand_2$ = rastgele değer, Pb =Yerel En İyi Değer,

$$p_{i,j}^{t+1} = p_{i,j} + v_{i,j}^{t+1} \quad (3.11)$$

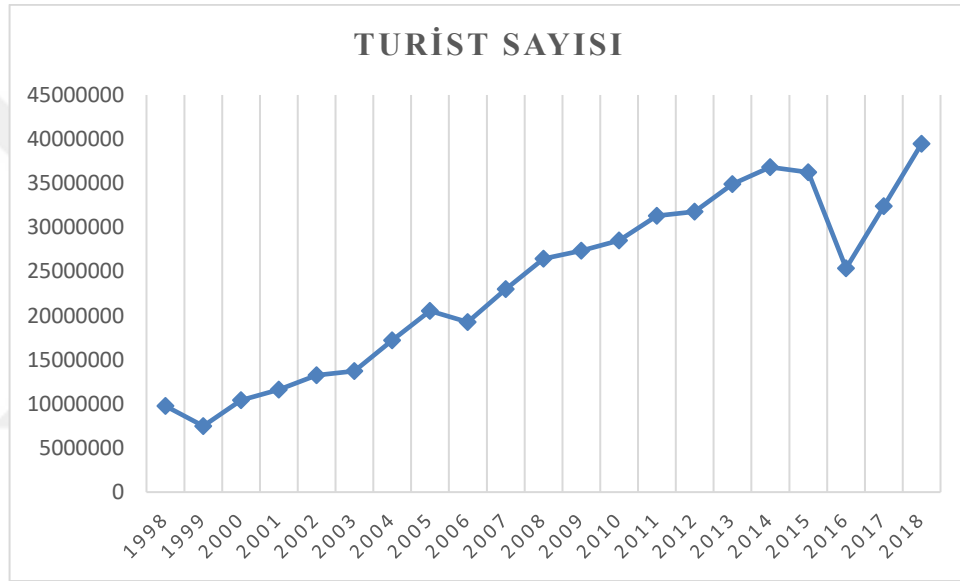
$P_{i,j}=i$. Parçacığın j . Pozisyonu, t =İterasyon Sayısı, $v_{i,j}=i$. Parçacığın j . hızı,

Adım 6. Adım 3 ve 5 arası önceden belirlenen maksimum iterasyon sayısı kadar devam ettirilir.

BÖLÜM 4. ARAŞTIRMA BULGULARI

4.1. Türkiye’ye Gelen Turist Sayısının Tahmin Edilmesi

Tez çalışmasının uygulanmasında T.C. Kültür ve Turizm Bakanlığı ve TÜİK tarafında ortak derlenen Türkiye’ye gelen yabancı ziyaretçi sayısı zaman serisi kullanılmıştır. Zaman serisi kısaca “Turist Sayısı” olarak isimlendirilmiştir. Zaman serisi 1998-2018 yılları arasında yıllık olarak gözlemlenmiş olup, grafiği Şekil 4.1 de verilmiştir.



Şekil 4.1. 1998-2018 arası Türkiye’ye gelen yabancı ziyaretçi sayısı

Zaman serisinin 1998-2015 yılları arasındaki gözlemler eğitim verisi, 2016-2018 yılları arasındaki gözlemleri ise test verisi olarak kullanılmıştır. Uygulamanın amacı Pi-Sigma yapay sinir ağının “Turist sayısı” zaman serisinin çözümündeki performansının araştırılmasıdır.

Pi-Sigma yapay sinir ağının (PS-YSA) performansı klasik yöntemlerden rastgele yürüyüş süreci (naive yöntem), Holt üstel düzleştirme yöntemi ve yapay sinir ağlarından çok katmanlı algılayıcı yapay sinir ağı (MLP-YSA), çarpımsal nöron model yapay sinir ağı (ÇNM-YSA), dentrit nöron model yapay sinir ağı (DNM-YSA) ile karşılaştırılmıştır.

Uygulanan yöntemlerin test kümesi performansları hata kareler ortalaması karekökü (RMSE) ve hatanın mutlak yüzdelik ortalaması (MAPE) ölçütleri kullanılarak karşılaştırılmıştır. RMSE ve MAPE ölçütlerinin formülleri aşağıda verilmiştir.

$$RMSE = \sqrt{\frac{1}{n_{test}} \sum_{t=1}^{n_{test}} (x_t - \hat{x}_t)^2} \quad (4.1)$$

$$MAPE = \frac{1}{n_{test}} \sum_{t=1}^{n_{test}} \left| \frac{x_t - \hat{x}_t}{x_t} \right| \quad (4.2)$$

İlk olarak rastgele yürüyüş süreci ve Holt yönteminin sonuçlarını inceleyelim. Tablo 4.1’de görüldüğü gibi rastgele yürüyüş sürecinin tahmin performansı Holt yöntemine göre daha iyidir. MAPE ölçütüne göre rastgele yürüyüş sürecinin %27’lik bir hata performansı vardır.

Tablo 4.1. Holt ve Rastgele Yürüyüş Süreci Sonuçları

	Rastgele Yürüyüş Süreci	Holt Yöntemi
RMSE	8.535.434.149	9.926.815
MAPE	0.2755	0.3102

Öngörüler incelendiğinde, gerçek değerler yıllara göre artış gösterir iken rastgele yürüyüş süreci öngörülerinin azalış gösterdiği, Holt yönteminin ise gerçek değerlere benzer hareket sergilediği görülmektedir. Ancak öngörülerin artış eğilimini yakalamasına rağmen Holt yönteminin öngörülerinin rastgele yürüyüş sürecine göre daha yüksek hataya sahip olduğu görülmektedir. Holt yönteminin tüm öngörülerinin gerçek değerlere göre daha yüksek olduğu ve öngörü hatalarının negatif olduğu görülmektedir. Turist sayısı zaman serisine yapay sinir ağı yöntemlerinin uygulanmasında farklı bir strateji izlenmiştir. Yapay sinir ağlarının uygulanmasında girdilerin sayısı 1 ile 5 arasında yani ilk 5 adım gecikmeli değişkenler olarak ve gizli tabaka birim sayısı ise 1 ile 5 arasında denenerek en iyi test kümesi performansını veren mimari tespit edilmiştir. Daha sonra belirlenen mimaride yapay sinir ağı 30

kez farklı rastgele başlangıç ağırlıklarına göre çalıştırılarak test kümesi için elde edilen ölçüt değerlerinin istatistikleri hesaplanmıştır. Hesaplanan istatistikler Tablo 4.2 de verilmiştir.

Tablo 4.2. YSA yöntemlerinden elde edilen sonuçlar

Method	Hata Ölçütü	Ortalama	Medyan	Standart Sapma	Çeyrek Sapma	Minimum	Maksimum
ÇKA-YSA	RMSE	6.346.260	6.366.423	194.736	213.298	5.982.129	6.674.615
	MAPE	0.1897	0.1921	0.0117	0.0089	0.1679	0.2046
Method	Hata Ölçütü	Ortalama	Medyan	Standart Sapma	Çeyrek Sapma	Minimum	Maksimum
ÇNM-YSA	RMSE	7.311.408	7.179.344	833.945	1.448.148	6.372.167	8.630.189
	MAPE	0.2255	0.2310	0.0226	0.0397	0,1946	0.2583
Method	Hata Ölçütü	Ortalama	Medyan	Standart Sapma	Çeyrek Sapma	Minimum	Maksimum
DNM-YSA	RMSE	7.513.886	7.552.195	461.531	793.029	6.915.005	8.186.525
	MAPE	0.1824	0.1834	0.0036	0.0025	0.1737	0.1864
Method	Hata Ölçütü	Ortalama	Medyan	Standart Sapma	Çeyrek Sapma	Minimum	Maksimum
PS-YSA	RMSE	6.459.503	6.490.473	220.698	276.170	5.998.527	6.708.127
	MAPE	0.1956	0.1993	0.0116	0.0167	0.1751	0.2078

İlk olarak zaman serisine çok katmanlı algılayıcı uygulanmış ve en iyi mimari girdi sayısının 4 ve gizli tabaka birim sayısının 4 olduğu durum olarak belirlenmiştir. Farklı rastgele başlangıç ağırlıklarına göre ortalama RMSE istatistiği 6.346.260 olup hem Holt hem de rastgele yürüyüş sonuçlarına göre daha iyidir. Bunun yanında ÇKA'nın en kötü test performansı bile 6.674.615 olup klasik yöntemlerden daha iyi olduğu görülmektedir. Benzer durum MAPE istatistikleri içinde geçerlidir. ÇKA-YSA'nın öngörülleri klasik yöntemlere göre özellikle 2017 yılında oldukça düşük hataya sahiptir. Ancak öngörüler gerçek değerlerdeki trend hareketini yakalayamamıştır. İkinci olarak zaman serisine çarpımsal nöron model yapay sinir ağı uygulanmıştır. En iyi mimari girdi sayısının 3 olduğu mimari olarak seçilmiştir. ÇNM-YSA'nın performansının ÇKA-YSA'ya göre daha kötü olduğu ancak klasik yöntemlere göre daha başarılı öngörü performansı elde edildiği görülmektedir. Üçüncü olarak zaman serisine dentrit nöron model yapay sinir ağı uygulanmıştır. En iyi mimari girdi sayısının 5, gizli tabaka birim sayısının 2 olduğu mimari olarak seçilmiştir. DNM-YSA'nın en kötü test performansı sonuçları bile klasik yöntemlere göre daha

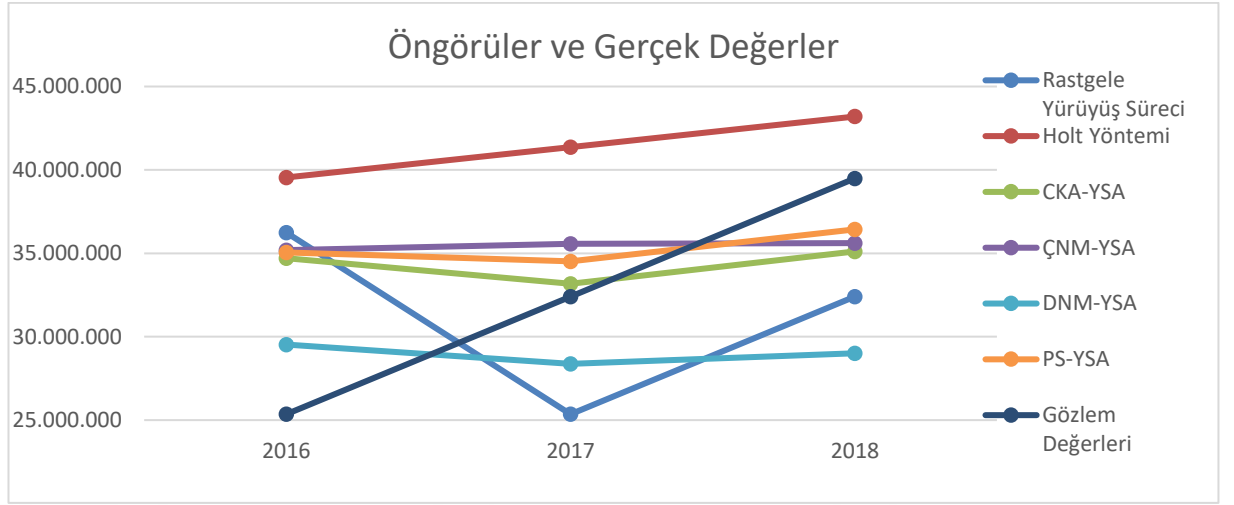
başarılıdır. Ancak YSA sonuçları ile karşılaştırıldığında ÇKA-YSA'ya göre daha başarısız öngörü sonuçları ortaya çıkmaktadır. ÇNM-YSA ile elde edilen sonuçları karşılaştırsak daha düşük maksimum istatistiği verdiği için en kötü performansın DNM-YSA için daha başarılı olduğu görülmektedir. Zaman serisine son olarak tezde performansı araştırılmak istenen PS-YSA uygulanmıştır. En iyi mimari girdi sayısının 4, gizli tabaka birim sayısının 3 olduğu durumda elde edilmiştir. Elde edilen sonuçlar incelenirse PS-YSA'nın klasik yöntemlere göre daha başarılı öngörü sonuçları elde edildiği görülmektedir. DNM-YSA sonuçları ile karşılaştırma yapıldığında PS-YSA'nın daha düşük ortalama RMSE, daha yüksek ortalama MAPE istatistiğine sahip olduğu görülmektedir. PS-YSA'nın daha düşük minimum ve maksimum RMSE istatistiği ve daha yüksek minimum ve maksimum MAPE istatistiğine sahip olduğu görülmektedir. Özetlemek gerekirse RMSE istatistiğine göre PS-YSA, MAPE istatistiğine göre de DNM-YSA daha başarılı olmaktadır. PS-YSA'yı ÇNM-YSA ile karşılaştırsak, RMSE ve MAPE istatistiklerine göre PS-YSA'nın daha başarılı sonuçlar ürettiği görülmektedir. PS-YSA'yı ÇKA-YSA ile karşılaştırsak PS-YSA'ya göre daha başarılı öngörü sonuçlarına sahip olduğu görülmektedir. Öngörüler incelendiğinde özellikle 2017 ve 2018 yılı için tahmin performansının oldukça iyi olduğu görülmektedir. Uygulanan tüm yöntemlerin öngörülerini ve en iyi test kümesi performansları Tablo 4.3 de verilmiştir.

Tablo 4.3. Uygulanan tüm yöntemlerin öngörülerini ve en iyi test kümesi performansları

Yıllar	Yöntemler					
	Rastgele Yürüyüş Süreci	Holt Yöntemi	CKA-YSA	ÇNM-YSA	DNM-YSA	PS-YSA
2016	36.244.632	39.547.697	34.711.858	35.189.865	29.528.833	35.055.813
2017	25.352.213	41.373.460	33.173.785	35.572.574	28.372.736	34.519.365
2018	32.410.034	43.199.222	35.109.892	35.611.226	29.014.244	36.432.613
RMSE	8.535.434	9.926.815	5.982.129	6.372.167	6.915.005	6.708.127
MAPE	0.2756	0.3102	0.1679	0.1946	0.1737	0.2078

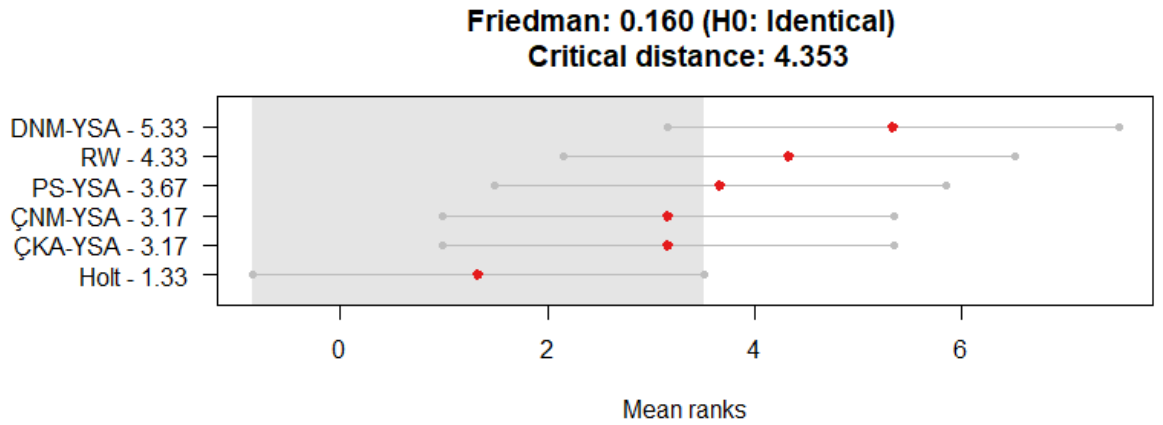
Ayrıca tüm yöntemlerden elde edilen öngörülerin, gerçek değerler ile bir arada grafiği Şekil 4.2 de verilmiştir.

Şekil 4.2. Yöntemlerin Öngörülleri ve Gerçek Değerler Grafiği



Son olarak elde edilen tüm öngörü yöntemlerinin sonuçları Nemenyi testi ile karşılaştırılmıştır. Testin uygulaması R programında gerçekleştirilmiştir. Nemenyi testi sonucuna göre öngörü yöntemlerinin performansları arasında anlamlı bir fark bulunamamıştır ($p=0.16$). Bu sonuçların elde edilen en iyi test performansı sonuçlarının bir karşılaştırması olduğu göz ardı edilmemelidir.

Şekil 4.3. En iyi test performanslarının karşılaştırılması için Nemenyi Testi



BÖLÜM 5. TARTIŞMA VE SONUÇ

Bu tez çalışmasında PS-YSA'nın Türkiye'ye gelen yabancı ziyaretçi sayısı zaman serisinin tahminindeki performansı araştırılmıştır. PS-YSA'nın performansı seçilmiş klasik öngörü yöntemleri ve yapay sinir ağları ile karşılaştırılmıştır. Yüksek dereceden bir YSA olan PS-YSA'nın klasik yöntemlere göre daha başarılı öngörü sonuçları üretebildiği görülmektedir. Diğer YSA ile PS-YSA'nın performansı karşılaştırıldığında ÇKA-YSA'ya göre daha başarısız sonuçlar elde edildiği ancak ÇNM-YSA ve DNM-YSA'ya göre daha başarılı sonuçlar elde edildiği görülmektedir. Test kümesi için elde edilen en iyi öngörü sonuçlarının hataları Nemenyi testi ile karşılaştırıldığında sonuçlar arasında anlamlı bir farklılık olmadığı görülmektedir. PS-YSA'nın Türkiye'ye gelen yabancı ziyaretçi sayısı zaman serisi için tercih edilebilir bir yöntem olduğu görülmektedir. Gelecek çalışmalarda PS-YSA'nın tahmin performansının derin YSA'lar ile karşılaştırılması planlanmaktadır.

KAYNAKLAR

- Akdeniz, E., Eğrioğlu, E., Bas, E., Yolcu, U. 2018. Recurrent Pi-Sigma Neural Network , Journal Of Artificial Intelligence And Soft Computing Research, 8(2),121-132.
- Akram, U., Ghazali, R., Mushtag, M.F. 2017. A Comprehensive Servey On Pi-Sigma Neural Network For Time Series Prediction, Faculty Of Computer Science And Information Technology, Technical Report.
- Ataseven, B. 2013. Yapay Sinir Ağları İle Öngörü Modellemesi, 10(39),101-115.
- Bas, E., Grosan, C., Eğrioğlu, E., Yolcu, U. High Order Fuzzy Time Series Method Based On Pi-Sigma Neural Network Engineering Applications Of Artificial Intelligence, 72, 350-356.
- Çuhadar, M. 2013. Modelling And Forecasting İnbound Tourism Demand To Turkey By MLP, RBF And TDNN Artificial Neural Networks: A Comparative Analysis, Journal Of Yasar University, 8(31), 5274-5295.
- Çuhadar, M., Güngör, İ., Göksu, A. 2009. Forecasting Tourism Demand By Artificial Neural Networks And Time Series Methods: A Comparative Analysis İn İnbound Tourism Demand To Antalya, 14(1),99-114.
- Çuhadar, M., Kayacan, C., 2005. “Yapay Sinir Ağları Kullanılarak Konaklama İşletmelerinde Doluluk Oranı Tahmini: Türkiye’deki Konaklama İşletmeleri Üzerine Bir Deneme”, Anatolia: Turizm Araştırmaları Dergisi, 16 (1),24-30.
- Delice, Y. 2008. Parçacık Sürü Optimizasyonu İle Yapay Sinir Ağlarından Sınıflandırma Kuralı Çıkarımı, (Yüksek Lisans Tezi) Erciyes Üniversitesi Sosyal Bilimler Enstitüsü (Kayseri).
- Ghazali, R., Husaini, N.A., Elderedy, W. 2006. Application Of Ridge Polynomial Neural Networks To Financial Time Series Prediction. İn: 2006 International Joint Conference On Neural Networks., 913-920.
- Ghazali, R., Hussain, A.J., Liatsis, P. And Tawfik, H. 2008. The Application Of Ridge Polynomial Neural Network To Multistep Ahead Financial Time Series Prediction, Neural Computing & Applications, 17(3), 311-323.
- Güngör, İ., Çuhadar, M. 2005. Antalya İline Yönelik Alman Turist Talebinin Yapay Sinir Ağı Yöntemi İle Tahmini, Gazi Üniversitesi Ticaret Ve Turizm Eğitim Fakültesi Dergisi, (2),84-98.
- Husaini, N.A., Ghazali, R., Nawı, N.M., İsmail, L.H. 2011. Pi-Sigma Neural Network For Temperature Forecasting İn Batu Pahat Communications İn Computer And Information Science , 180 CCIS (PART2), 530-541.
- İçöz, O., Kozak, M. 2002. Turizm Ekonomisi, Turhan Kitabevi, Ankara
- Kanungo, D.P., Nayak, J., Naik B., Behera H.S. 2016. Non-Linear Classification Using Higher Order Pi-Sigma Neural Network And Improved Particle Swarm Optimization : An Experimental Analysis Advances İn Intelligent Systems And Computing, 411, 507-518.

- Karahan, M. 2011. İstatistiksel Tahmin Yöntemleri: Yapay Sinir Ağları Metodu İle Ürün Talep Tahmini, Doktora Tezi, Selçuk Üniversitesi Sosyal Bilimler Enstitüsü, Konya
- Karahan, M. 2015. A Case Study On Forecasting Of Tourism Demand With Artificial Neural Network Method, Süleyman Demirel University The Journal Of Faculty Of Economics And Administrative Sciences, 20(2), 195-209
- Mehrotra, K., Mohan, C.K., Ranka, S. 1996. Elements Of Artificial Neural Networks
- Mijwel, M.M. 2018. Artificial Neural Networks Advantages And Disadvantages.
- Nayak, J., Kanungo, D.P., Naik, B., Behera, H.S. 2014a. A Higher Order Evolutionary Jordan Pi-Sigma Neural Network With Gradient Descent Learning For Classification, 2014a International Conference On High Performance Computing And Applications.
- Nayak, J., Naik, B., Behera, H.S. 2014b. A Hybrid Based Pi-Sigma Neural Network (PSNN) With Standard Back Propagation Gradient Descent Learning For Classification. International Conference On Control, Instrumentation, Communication And Computational Technologies, Iccicct 2014, 878-885.
- Nayak, J., Naik, B., Behera, H.S. 2016. Optimizing A Higher Order Neural Network Through Teaching Learning Based Optimization Algorithm Advances In Intelligent Systems And Computing, 410, 57-71.
- Nayak, J., Naik, B., Behera, H.S., Abraham, A. 2015. Particle Swarm Optimization Based Higher Order Neural Network For Classifications, Smart Innovation Systems And Technologies, 31, 401-414.
- Nayak, S.C. 2017. Development And Performance Evaluation Of Adaptive Hybrid Higher Order Neural Networks For Exchange Rate Prediction International Journal Of Intelligent Systems And Applications, 9(8), 71-85.
- Öztemel, E. 2012. Yapay Sinir Ağları, Papatya Yayıncılık.
- Öztürk, K., Şahin, M.E. 2018. A General View Of Artificial Neural Networks And Artificial Intelligence, 6(2), 25-36.
- Shin And Gosh 1991. The Pi-Sigma Network: An Efficient Higher Order Neural Network For Pattern Classification and Function Approximation. In Proceedings Of The International Joint Conference On Neural Networks,
- Soysal, M., Ömürgönülşen, M. 2010. An Application On Demand Forecasting In The Turkish Tourism Industry, Turizm Araştırmaları Dergisi, 21(1), 128-136.
- Tawfik, H., Liatsis, P. 1997. Prediction Of Nonlinear Time Series Using Higher Order Neural Networks, Proceeding IWSSIP'97 Conference, Ponzan, Poland
- Yılmaz, O. 2019. Pi-Sigma Yapay Sinir Ağlarının Diferansiyel Gelişim Algoritması İle Eğitimi, (Yüksek Lisans Tezi) Giresun Üniversitesi Fen Bilimleri Enstitüsü (Giresun).
- Zorlutuna, Ş., Bircan, H. 2019. Türkiye'ye Gelen Turist Sayısı Tahmininde Zaman Serileri Analizi Ve Yapay Sinir Ağları Yöntemlerinin Karşılaştırılması, S.C.Ü. İktisadi Ve İdari Bilimler Dergisi, 20(2).

EKLER

Türkiye'ye gelen turist sayısı verilerinin Pi-Sigma yapay sinir ağı ile çözümü için kullanılan MATLAB kodu aşağıda verilmiştir.

```
function
[rmsetest,mapetest,msebest,pgbest,rmse,yhatg,yhattestdg,rn,mape,itn]=
pismgpsotest(xt,m,drc,ps,wmaps,w1,w2,c1i,c2i,c1f,c2f,itn,ntest)
msebestEski=10^6;kk=0;
x0=xt;
n0=length(xt);
xtest=x0((n0-ntest+1):n0);
xt=x0(1:(n0-ntest));
x=(xt-min(xt))/(max(xt)-min(xt));
rn=rand('seed');
p=m*drc+drc;
n=length(x);
A=unifrnd(0,1,ps,p);
V=unifrnd(-vmaps,vmaps,ps,p);
w=zeros(1,itn);
c1=zeros(1,itn);
c2=zeros(1,itn);
for k=1:ps
    W(1,:)=A(k,1:drc);
    for j=2:m
        W(j,:)=A(k,drc*(j-1)+1:drc*j);
    end
    b=A(k,drc*m+1:drc*m+drc);
    yhat=multifpismg(x,W,b,m,drc);
    nh=length(yhat);
    for i=1:nh
        hata(i)=(yhat(i)-x(n-nh+i))^2;
    end
    mse(k)=mean(hata);
end
MSEegt=min(mse);
for i=1:ps
    if MSEegt==mse(i)
        dd=i;
        break
    end
end
pgbest=A(dd,:);
msebest=mse(dd);
pid=A;
msepid=mse;
i22=0;
for i1=1:itn
    i22=i22+1;
    w(i1)=(w1-w2)*((itn-i1)/itn)+w2;
    c1(i1)=(c1f-c1i)*(i1/itn)+c1i;
    c2(i1)=(c2f-c2i)*(i1/itn)+c2i;
    for i2=1:ps
        for i3=1:p
            V(i2,i3)=V(i2,i3)*w(i1)+c1(i1)*unifrnd(0,1)*(pid(i2,i3)-
A(i2,i3))+c2(i1)*unifrnd(0,1)*(pgbest(i3)-A(i2,i3));
            V(i2,i3)=min(vmaps,max(-vmaps,V(i2,i3)));
            A(i2,i3)=A(i2,i3)+V(i2,i3);
```

```

        end
    end
    if i22>=100
        A=unifrnd(0,1,ps,p);
        V=unifrnd(-vmaps,vmaps,ps,p);
        i22=0;
    end
    for k=1:ps
        W(1,:)=A(k,1:drc);
        for j=2:m
            W(j,:)=A(k,drc*(j-1)+1:drc*j);
        end
        b=A(k,drc*m+1:drc*m+drc);
        yhat=multifpisgm(x,W,b,m,drc);
        nh=length(yhat);
        for i=1:nh
            hata(i)=(yhat(i)-x(n-nh+i))^2;
        end
        mse(k)=mean(hata);
    end
    MSEegt=min(mse);
    for i=1:ps
        if MSEegt==mse(i)
            dd=i;
            break
        end
    end
    if MSEegt<msebest
        pgbest=A(dd,:);
        msebest=mse(dd);
        [i1,msebest]
    end
    for j1=1:ps
        if mse(j1)<=msepid(j1)
            pid(j1,:)=A(j1,:);
            msepid(j1)=mse(j1);
        end
    end
    if abs((msebestEski-msebest)/msebest)<10^-3
        kk=kk+1;
    end
    if kk>50
        break
    end
    msebestEski=msebest;
end
itn=i1;
W(1,:)=pgbest(1:drc);
for j=2:m
    W(j,:)=pgbest(drc*(j-1)+1:drc*j);
end
b=pgbest(drc*m+1:drc*m+drc);
yhat=multifpisgm(x,W,b,m,drc);
nh=length(yhat);
yhatg=(yhat)*(max(xt)-min(xt))+min(xt);
for i=1:nh
    hata(i)=(yhatg(i)-xt(n-nh+1))^2;
    hata2(i)=abs((yhatg(i)-xt(n,nh+i))/xt(n-nh+i));
end

```

```

rmse=mean(hata)^0.5;
mape=mean(hata2);

x=(x0-min(x0))/(max(x0)-min(x0));
yhattum=multifpism(x,W,b,m,drc);
nt=length(yhattum);
yhatte= yhattum( (nt-ntest+1):nt);
yhatte= (yhatte) * (max(xt)-min(xt))+min(xy);
for i=1:ntest
    hata3(i)=(xtest(i)-yhatte(i))^2;
    hata4(i)=abs((xtest(i)-yhatte(i))/xtest(i));
end
rmsetest=(mean(hata3))^0.5;
mapetest=mean(hata4);
end

```

Pi-Sigma yapay sinir ağının kod çıktısı için gerekli kodlar şu şekildedir.

```

function [y]=multifpism(x,W,b,mi,drc)
M=lagmatrix(x,(1:mi));
n=length(x);
M=M(mi+1:n,:);
nc=size(M',2);
nr=size(M',1);
y=zeros(nr,1);
for i=1:nr
    for j=1:drc
        a(j)=sum(M(:,i).*W(:,j))+b(j);
    end
    y(i)=1/(1+exp(-prod(a)))
end

```

ÖZGEÇMİŞ

1996'da Azərbaycan Naxçıvan'da doğdu. İlk, ortaöğretim ve lise öğrenimini Naxçıvan şəhər C.Naxçıvanski adına 11 nömrəli tam ortaokulunda tamamladı. 2014'de Giresun Üniversitesi Fen-Edebiyat Fakültesi İstatistik bölümünü kazandı. 2018'de Giresun Üniversitesinden mezun oldu. 2019 Ocak ayında Giresun Üniversitesi Fen-Edebiyat Fakültesi İstatistik ABD Yüksek Lisans Programına başladı. Aktiv olaraq Azərbaycan Naxçıvan'da Gemikaya Decoria şirkətində satın alma pozisiyonda çalışmaktadır.

