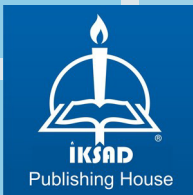
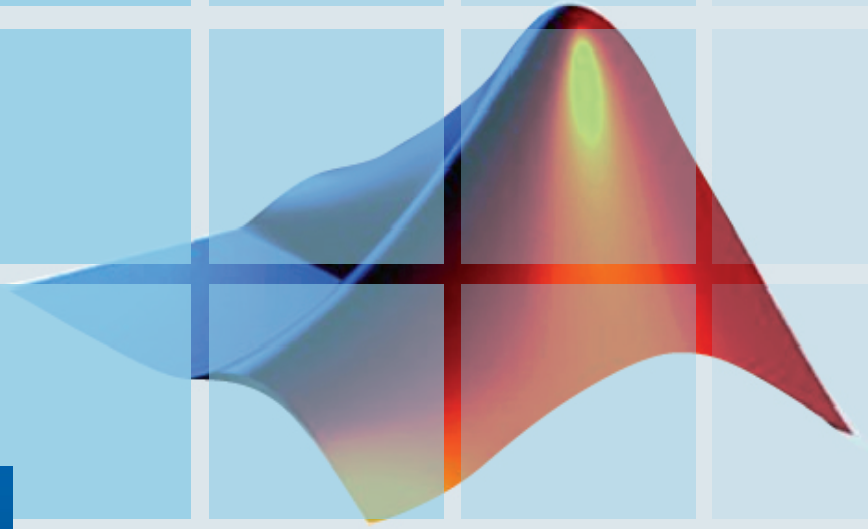


YAPAY SİNİR AĞLARI İLE KALİTE KONTROL: ÖRNEK OLAY ÇALIŞMASI

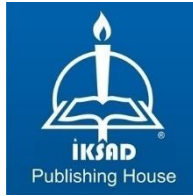
BİRCE YAVUZ
Dr. TÜLAY KORKUSUZ POLAT



YAPAY SİNİR AĞLARI İLE KALİTE KONTROL: ÖRNEK OLAY ÇALIŞMASI

BİRCE YAVUZ

Dr. TÜLAY KORKUSUZ POLAT



Copyright © 2020 by iksad publishing house
All rights reserved. No part of this publication may be reproduced,
distributed or transmitted in any form or by
any means, including photocopying, recording or other electronic or
mechanical methods, without the prior written permission of the publisher,
except in the case of
brief quotations embodied in critical reviews and certain other
noncommercial uses permitted by copyright law. Institution of Economic
Development and Social
Researches Publications®
(The Licence Number of Publicator: 2014/31220)
TURKEY TR: +90 342 606 06 75
USA: +1 631 685 0 853
E mail: iksadyayinevi@gmail.com
www.iksadyayinevi.com

It is responsibility of the author to abide by the publishing ethics rules.
Iksad Publications – 2020©

ISBN: 978-625-7687-73-7
Cover Design: Ibrahim KAYA
December / 2020
Ankara / Turkey
Size = 16 x 24 cm

İÇİNDEKİLER

BÖLÜM 1. GENEL GİRİŞ	5
BÖLÜM 2. MATLAB PROGRAMI	7
BÖLÜM 3. YAPAY SİNİR AĞLARI	16
BÖLÜM 4. ÖRNEK UYGULAMA	38
4.1. Mevcut Durumun İncelenmesi	39
4.2. Yapay Sinir Ağı Oluşturulması ve Eğitimi	41
4.3. Kullanıcı Arayüzü Oluşturma	58
BÖLÜM 5. İSTATİSTİKSEL ANALİZ	69
5.1. Hipotez Testi / Mann Whitney Testi	69
5.2. Lojistik Regresyon.....	77
BÖLÜM 6. SONUÇ	87
KAYNAKLAR	89
EKLER	98

TABLolar LİSTESİ

Tablo 4.1. Örnek veri toplama formu.....	40
Tablo 4.2. Girdi verileri (örnek).....	43
Tablo 4.3. Normalize edilen girdi verileri	44
Tablo 4.4. Hedef veri seti	45
Tablo 4.5. Nöron sayılarına göre YSA eğitim sonuçları.....	46
Tablo 4.6. Ağ parametreleri	47
Tablo 5.1. İstatistik sonuçları	75
Tablo 5.2. Tahmin edilen fark.....	75
Tablo 5.3. Mann-Whitney testi sonuçları	76
Tablo 5.4. Ürün durumları.....	80
Tablo 5.5. Lojistik regresyon model özeti.....	81
Tablo 5.6. Uyumluluk testi sonuçları	83
Tablo 5.7. Sonuç karşılaştırma tablosu	85

ŞEKİLLER LİSTESİ

Şekil 3.1. Geri Beslemeli Yapay Sinir Ağı'nın genel yapısı.....	33
Şekil 4.1. Matlab YSA aracı ağ oluşturma ekranı	48
Şekil 4.2. Matlab'da oluşturulan ağın yapısı	49
Şekil 4.3. Verilerin dağılımı	49
Şekil 4.4. YSA Eğitimi	50
Şekil 4.5. YSA Regresyon Grafikleri	51
Şekil 4.6. Sinir ağının simülasyonu.....	52
Şekil 4.7. Simülasyon çıktıları	53
Şekil 4.8. YSA çıktı değerlerinin hedef değerler ile karşılaştırılması	54
Şekil 4.9. YSA Simulink modeli	55
Şekil 4.10. YSA mimarisi	56
Şekil 4.11. Simulink modeli (1. katmanda yapılan işlemler)	56
Şekil 4.12. Simulink modeli (2. katmanda yapılan işlemler)	57
Şekil 4.13. Figure1 ekranı	59
Şekil 4.14. Figure1 edit text	60
Şekil 4.15. Matlab Grafikselle Kullanıcı Arayüzü.....	61
Şekil 4.16. Interface.m dosyası	62
Şekil 4.17. Edit text1 (thickness) callback	63
Şekil 4.18. Hata mesajı.....	64
Şekil 4.19. Kaydet butonu callback.....	65
Şekil 4.20. Matlab komut ekranı	67
Şekil 4.21. Excel'e kaydedilen veriler	68
Şekil 5.1. Aykırı değer testi - YSA Çıktı	70
Şekil 5.2. Aykırı değer testi - Hedef.....	71

Şekil 5.3. Normallik testi.....	72
Şekil 5.4. Mann-Whitney Testi.....	74
Şekil 5.5. Verilerin yüklenmesi.....	78
Şekil 5.6. Lojistik regresyon modelinin kurulması	79
Şekil 5.7. Lojistik regresyon	80
Şekil 5.8. Tahmin edilen sonuçların model ile uyumluluğu (fits).....	82
Şekil 5.9. Lojistik regresyonla tahmin	84

BÖLÜM 1.

GENEL GİRİŞ

Otomobil parçaları üreten firmada, bir projede üretilen çarpışma önleyici kirişlerin müşteri firma isteğiyle %100 ölçü ve görsel kontrolü zorunlu tutulmuştur. Pres işleminden önce büyük çelik rulosu halinde gelen hammadde, kesme, bükme, delik delme işlemleri ile form alır ve kontrollerden sonra sıradaki proses olan kaynak bölümüne gönderilir. Paslanmaz çelik plakalar tonluk kalıplar kullanılarak yüksek basınç sayesinde şekillendirilmektedir. Bu basınçların düzeyi, plakaların büyüklüğüne bağlı olacak şekilde değişebilmektedir. Pres aşamasından sonra ürünler büyük bir titizlikle kontrol edilmektedir. Ölçü kontrolü esnasında doğruluğu sağlamak için, üretilen her ürüne özel olan kontrol fişlerinden yararlanılmaktadır. Ölçüm sonuçlarının işlenmesinde de hiçbir hataya yer verilmemektedir.

Üretilen otomobil kirişlerin kontrol fişlerindeki ölçü kontrolleri sonucu parça kalınlığı, bükülme sonrası incelme miktarı, parçanın çizim dosyasında da verilen olması gereken ağırlığı, delik çapı ve uzunluğu değerlerinin hedeflenen tolerans aralıklarında olup olmadığı hızlı ve basit bir şekilde bilgisayar ortamında görülüp kaydedilecektir.

Yapılan çalışmada, ele alınan ürünün ölçüm değeri verileri ve sonucunda hedeflenen çıktının (kabul/red) Matlab tabanlı “Yapay Sinir Ağları” ile modellenmesi amaçlanmıştır. Girdi verileri kalınlık, incelme, ağırlık, delik çapı, uzunluk ve hedef çıktı 1 ‘kabul’ ve 0 ‘red’

olacak şekilde 1050 örnek için tanımlanmış, en uygun eğitim fonksiyonu ve nöron sayısı ile sinir ağı eğitilmiştir. MATLAB GUI yardımıyla kullanıcının ölçüm değerlerini girmesini, ürün durumunun belirlenmesi ve kaydedilmesini kolaylaştırmak amacıyla kullanıcı arayüzü oluşturulmuştur.

Eğitilen yapay sinir ağının güvenilirliği hipotez testi kurularak değerlendirilmiştir.

Lojistik regresyon modeli ile tahmin edilen sonuçlar ile yapay sinir ağı simülasyon sonuçları karşılaştırılmıştır.

BÖLÜM 2.

MATLAB PROGRAMI

MATLAB, MATrix LABoratory'un kısaltmasıdır. MATLAB, LINPACK (doğrusal sistem paketi) ve EISPACK (Eigen sistem paketi) projeleri tarafından geliştirilen matris yazılımına kolay erişim sağlamak için yazılmış, teknik bilgi işlem için yüksek performanslı orijinal bir dildir (Houcque, 2005).

Hesaplama, görselleştirme ve programlama ortamını bütünleştirir. Ayrıca MATLAB, modern bir programlama dili ortamıdır: karmaşık veri yapılarına sahiptir, yerleşik düzenleme ve hata ayıklama araçları içerir ve nesne yönelimli programlamayı destekler. Bu faktörler MATLAB'ı öğretim ve araştırma için mükemmel bir araç haline getirir. MATLAB, teknik sorunları çözmek için geleneksel bilgisayar dillerine (örneğin, C, FORTRAN) kıyasla birçok avantaja sahiptir (Houcque, 2005).

Sürekli geliştirilen MATLAB artık veri analizi, sinyal işleme, optimizasyon ve birkaç tür bilimsel hesaplama gerektiren sorunları çözmek için yerleşik işlevlere sahiptir. Ayrıca 2 boyutlu ve 3 boyutlu grafikler ve animasyon için işlevler içermektedir (Bellala, 2020).

MATLAB, temel veri ögesi boyutlandırma gerektirmeyen bir dizi olan etkileşimli bir sistemdir. Yazılım paketi 1984'ten beri ticari olarak

mevcuttur ve Őu anda dŐnya apında oęu Őniversite ve endŐstride standart bir ara olarak kabul edilmektedir (Houcque, 2005).

MATLAB programının nemli bir avantajı; “toolbox” adı verilen sembolik modelleri iermesidir (“control system toolbox”, “fuzzy logic toolbox” gibi) (Kubat, 2012).

MATLAB programını Windows ile baŐlatmak iin, Windows masaŐtŐnŐzdeki matlab simgesine ift tıklayın. Unix'ten baŐlatmak iin iŐletim sistemi komut istemine MATLAB yazın. Mac-os'dan baŐlamak iin X11'i aın (yani bir x terminal penceresi aın), ardından istemde MATLAB yazın.

MATLAB yazılımı baŐlatıldıęında, ana blŐmŐn komut penceresi olduęu bir pencere aılır. Komut penceresinde Őunu grmelisiniz:

>>

>>: bilgi istemi (prompt) olarak adlandırılır.

Bu komut istemi matlab'ın bir komut bekledięi anlamına gelir. AŐaęıdaki yntemlerden biriyle istedięiniz zaman ıkabilirsiniz:

- MATLAB masaŐtŐnŐn saę Őst kŐesindeki x (kutuyu kapat) tıklayın
- Komut penceresi isteminde quit veya exit yazın ve ardından "enter" tuŐuna basın.

MATLAB 'ın başlatılması, kullanıcının belgeler klasöründe otomatik olarak matlab adlı bir klasör oluşturur. Bu özellik oldukça kullanışlıdır çünkü varsayılan çalışma klasörüdür. Komut penceresinden kaydedilen her şey bu klasöre kaydedilir. Gerekirse, imleci komut penceresine yerleştirerek ve kenarlığı içinde herhangi bir yere fare düğmesine sol tıklayarak komut penceresini etkinleştirin (Hahn ve Valentine, 2019).

Komut penceresinde MATLAB etkileşimli olarak kullanılabilir. Prompt'a herhangi bir MATLAB komutu veya ifadesi girilebilir ve MATLAB sonuç kısmı ile hemen yanıt verir. Ayrıca, betik dosyalarında (script files) veya MATLAB kod dosyalarında bulunan programları MATLAB 'de yazmak da mümkündür (Attaway, 2018).

MATLAB'da satır sonunda çıktı verilmesini engellemek istiyorsak aşağıdaki örneklerde olduğu gibi komut satırını sonlandırmak için noktalı virgül kullanmak gerekmektedir. Noktalı virgül kullanılmazsa satırın çıktısı MATLAB tarafından gösterilecektir (Kattan, 2008).

```
>> y = 30;
```

```
>> z = 8;
```

```
>> x = 2; y-z;
```

```
>> w = 4*y + 3*z
```

```
w =
```

```
144
```

MATLAB programı büyük / küçük harfe duyarlıdır, yani küçük harfli değişkenler büyük harfli değişkenlerden farklıdır. X ve x değişkenlerini kullanarak aşağıdaki örnekler yapılabilir (Kattan, 2008):

```
>> x = 1
x =
     1
>> X = 2
X =
     2
>> x
x =
     1
>> X
X =
     2
```

Aşağıdaki örnekler, matrislerin MATLAB programında nasıl girileceğini ve bazı basit matris işlemlerinin nasıl gerçekleştirileceğini göstermektedir (Kattan, 2008):

```
>> x = [1 2 3 ; 4 5 6 ; 7 8 9]
x =
     1     2     3
     4     5     6
     7     8     9
>> y = [1 ; 0 ; -4]
```

```
y =  
    1  
    0  
   -4  
  
>> w = x*y  
  
w =  
   -11  
   -20  
   -29
```

MATLAB programının kullanımını ve uygulama alanlarını detaylı olarak anlatan pek çok kaynak bulmak mümkündür. Örneğin Gilat (2016) tarafından hazırlanan kitap, MATLAB 'ı ilk kez kullanan ve bilgisayar programlama konusunda çok az deneyimi olan veya hiç deneyimi olmayan öğrencilere yöneliktir. Birinci sınıf mühendislik derslerinde veya MATLAB 'ın öğretildiği atölyelerde ders kitabı olarak kullanılabilir olan kaynak, MATLAB'ın problem çözme aracı olarak kullanıldığı daha ileri bilim ve mühendislik derslerinde de referans görevi görebilir. Ayrıca öğrenciler ve uygulayıcı mühendisler tarafından MATLAB 'ın kendi kendine öğrenilmesi esnasında da kullanılabilir.

Lee (2019) kitabını, matlab ile programlamaya tamamen yeni başlayan lisans öğrencileri için tasarlamıştır. Örnek olay incelemeleri ve örnekler kitapta kapsamlı bir şekilde kullanılmaktadır. Lee (2019) matlab

öğrenmenin en iyi yolunun deneyimli programcılar tarafından yazılmış programları incelemek olduğuna ve bu örnek programların kalitesinin kitabın kalitesini belirlediğine inanmaktadır. Bu nedenle kitaptaki örnekler, kullanıcılara matlab programlamayı öğretmek ve kendi problem çözüme potansiyellerine ilham vermek için özenle tasarlanmıştır. Kitapta kullanılan örneklerin çoğu, tek ve belirli bir problemden ziyade bütün bir problem sınıfını çözmek için tasarlanmıştır.

Chapra (2017) “MATLAB ile uygulamalı sayısal metotlar” çalışmasını, sayısal problem çözmeyi öğrenmek isteyen mühendisler ve bilim adamları için yazmıştır. Geliştiricilerden çok sayısal yöntem kullanıcılarını hedefleyen kitap, okuyucuları motive etmek için matematikten çok problemleri kullanmaktadır. Kitapta her bir bölüm için bölüm hedefleri, çalışma örnekleri ve öğrenci dostu problemler de dahil olmak üzere öğrenci odaklı bir yaklaşım sunulmaktadır. MATLAB, kitaba tam olarak entegre edilmiştir ve birincil hesaplama ortamı olarak kullanılmaktadır. M-dosyası işlevlerinin nasıl programlanacağı da dahil olmak üzere MATLAB'a giriş sağlanmaktadır. Okuyucu dostu olan kitapta yerleşik MATLAB sayısal işlevleri açıklanmış ve gösterilmiştir.

Derin öğrenme için MATLAB'ın gücünden yararlanma konusunda bir çalışma yapan Paluszek ve Thornas (2020), çalışmaları ile derin öğrenmeye ve MATLAB'ın derin öğrenme araç kutularını kullanmaya giriş sağlamaktadırlar. Kitap, bu araç kutularının, derin öğrenmenin

tüm yönlerini uygulamak için gereken tüm işlevleri nasıl sağladığını anlatmaktadır. Kitapta borsa, doğal dil ve açılarla yörünge belirlemesi dahil olmak üzere karmaşık sistemlerin modellenmesi öğretilmektedir. MATLAB kullanarak dinamiklerin ve kontrolün ele alınmasını ve derin öğrenme algoritmalarının ve yaklaşımlarının entegre edilmesi anlatılmaktadır. Kitapta ayrıca görüntüleri kullanarak uçak navigasyonuna derin öğrenme uygulamasında ele alınmaktadır. Araştırmacılar kitapta özetle; MATLAB kullanarak derin öğrenmeyi keşfetmeyi, bunu algoritmalarla karşılaştırmayı, örnekleri ile birlikte MATLAB'da bir derin öğrenme işlevi yazmayı, Derin öğrenmeyle ilgili MATLAB araç kutularını kullanmayı vb. anlatmaktadırlar.

Robotik ve bilgisayarla görmenin birleşimi olan robotik görme, sensörlerden elde edilen verilere bilgisayar algoritmalarının uygulanmasını içermektedir. Corke (2017), biri robotik ve diğeri görme için olmak üzere iki açık kaynaklı MATLAB toolbox (Araç Kutusu) ile birçok önemli algoritmanın uygulamasını sağladığı kitabında kullanıcıların sadece temsili değil, gerçek problemlerle çalışmasına da fırsat sağlamaktadır.

MATLAB, fizik bilimi için pek çok soruna çözüm yolu bulmada eksiksiz bir araçlar topluluğu sağlamaktadır. Ayrıca pakette birçok öğretici ve kullanışlı arama özelliği de bulunmaktadır. Aslında, birçok üniversitenin MATLAB program lisansı bulunmaktadır. Böylece üniversitenin öğrencileri programı ücretsiz olarak kullanılabilir. Bu tür imkanlara sahip olmayanlar için ise MATLAB'ın öğrenci versiyonunu

bulmakta mevcuttur. MATLAB ayrıca python diliyle de uyumludur, böylece kullanıcılar isterlerse komut dosyalarını karıştırabilir. Farklı kullanım alanları ile çalışma yapan araştırmacılardan birisi de Green (2020)' dir. Green (2020) çalışmasında, uzay ile ilgili çalışmalarda keşif aracı olarak MATLAB - yazılım paketinin kullanılmasını anlatmaktadır. Başlangıçta yıldız oluşumu ve evrim anlayışındaki son gelişmeler, kara delikler, jeodezik hareketler, yerçekimi dalgalarının emisyonu gibi olayları anlattığı kitabında tüm bu konular, metin, rakamlar, formüller ve sayısal girdi ve çıktıyı birleştirmek için MATLAB Uygulamasını kullanarak bütünlük bir şekilde ele almıştır.

Ciaburro (2017) çalışmasında, makine öğreniminin temel kavramlarını tanıtmakta ve ardından farklı algoritma türlerini açıklamaktadır. Ek olarak, bazı giriş, arka plan bilgileri ve MATLAB ortamı ile ilgili temel bilgileri anlattığı kitabında, MATLAB 'ın makine öğrenimi için sunduğu temel araçları da detaylı şekilde anlatmaktadır. Ciaburro (2017), verilerin MATLAB'de nasıl içe aktarılacağını ve düzenleneceğini öğreterek daha sonra toplanan veriler için mevcut olan farklı formatları analiz eder ve verilerin MATLAB içine ve dışına nasıl taşınacağını gösterir. Veri analizinin sonraki aşaması için verilerin doğru biçimde nasıl organize edileceğini öğreten kitap, temel değişken türlerinin ve verilerin temizleme derecesinin analizinden başlayarak analiz ve modelleme için en uygun verilerin hazırlanması için mevcut teknikleri analiz eder, daha sonra verilerin anlaşılmasında kilit rol oynayan veri görselleştirmeyi de anlatmaktadır. Kitap, MATLAB ortamında doğru regresyon analizinin nasıl yapılacağını göstererek

tahmin ve grafik oluřturma dahil olmak üzere regresyon analizi için kullanılan matlab arayüzünü de anlatır. Sınıflandırma konusunun da anlatıldığı kitapta, *en yakın komřuları kullanarak bir nesnenin nasıl sınıflandırılacağı, *sınıflandırma için olasılık ilkelerinin nasıl kullanılacağı, *karar ağaçları ve kuralları kullanarak sınıflandırma tekniklerini gibi konularda yer almaktadır.

MATLAB konusunda daha detaylı bilgi sahibi olmak için Pratap (2016), Gonzales ve Woods (2020), Chapman (2019) ve Ling ve Langtangen (2016) kaynaklarından yararlanılabilir.

BÖLÜM 3. YAPAY SİNİR AĞLARI

Yapay sinir ağları, adlarından da anlaşılacağı gibi, biyolojik (insan veya hayvan) merkezi sinir sisteminin sinir hücresi (nöronları) ağlarındaki karar sürecini büyük bir şekilde simüle etmeye çalışan hesaplama ağlarıdır. Bu simülasyon, büyük bir hücre-hücre (nöron-nöron, eleman-eleman) simülasyonudur. Biyolojik nöronların nörofizyolojik bilgisinden ve bu tür biyolojik nöronların ağlarından ödünç alır. Bu nedenle, bilgisayar öğelerinin ve bunların ağlarının organizasyonuna bakılmaksızın insan beyni hesaplamasını değiştirmeye, geliştirmeye veya hızlandırmaya hizmet eden geleneksel (dijital veya analog) bilgisayar makinelerinden farklıdır. Yine de, sinir ağlarının sağladığı simülasyonun çok büyük olduğu tartışılmaz bir gerçektir (Graupe, 2019).

Yapay sinir ağları biyolojik olarak ilham alan akıllı hesaplamadır ve modern zamanlarda en kullanışlı ve güçlü yapay zeka yöntemlerinden biridir. Hali hazırda otomatik tanıma (sayılar / harfler / resimler vb.) sağlayan teknolojilerin çoğu, öğrenme ve kod çözme bazı sinir ağları tarafından yürütülmektedir. Bir vücuttaki nöronlar elektrik sinyalini bir uçtan diğerine, yani dendritlerden aksonlar boyunca uçlara iletirler. Çok karmaşık bir nöron ağı vardır ve nöronlardan gelen sinyaller sinir sistemi boyunca beyne iletilir. Bir insan beyni, dendritleri ve terminalleri yoluyla birbirine bağlanan ~ 100 milyar nörondan oluşur.

Yapay bir sinir ağı bu süreci taklit etmeye çalışır. Biyolojik sinir ağının yapay bir modele kopyası, her biri bir sonraki katmandaki diğer her nörona bağlanan nöron katmanlarından geçer. Nöronlar genellikle bir giriş sinyalini bir eşiği geçene ve bir çıkış tetiklenene kadar bastırır. Yapay sinir ağında, bu tür bir işleve etkinleştirme işlevi adı verilir (Thakur vd., 2021).

Yapay sinir ağları (YSA) ile teorik çerçeve, erişilebilir bilgilerdeki çok karmaşık eğilimleri karakterize edebilen veri yapılarından biridir. Yapay sinir ağları bir nöron veya düğüm katmanı grubundan oluşur. Sağlam bir paralel bağlantıya sahip bir yapıda organize edilmiştir. Sinir ağı modelini biriktirmek için, girdi ve çıktı verilerini değerlendirmek önemlidir. Girdi değişkenleri ya doğal veri analizinden ya da toplanan bilgilerden seçilir. Bu yüzden çıktı değişkenleri, her bir nöronun bilgiyi aldığı ve aktivasyon özelliği ile diğer nörona dağıttığı giriş verilerinin birbirine bağlanmasından kaynaklanmaktadır. Bu şekilde, veriler büyük çaplı çapraz iletişimlerle işlenir (Effendy vd., 2020).

Yapay Sinir Ağı (YSA), karmaşık örüntü tanıma ile ilgili insan beyni işlevinin matematiksel eşdeğeridir. İnsan verilerinin işlenmesine benzer şekilde, YSA'lar, hedef çıktı sağlayarak veya sağlamadan girdi olarak sunulan verilerle eğitilmektedir. Ağ, bir girdi katmanı, bir çıktı katmanı ve bir veya daha fazla gizli katmandan oluşur. Her katman bir dizi nöron veya düğümden oluşur. Nöronların sayısı, veri modelinin karmaşıklığına ve diğer gereksinimlere göre değişebilir. Katmanlar,

veri tahminine göre bir bağlantının göreceli önemini belirlemekten sorumlu olan ağ "ağırlıkları" ile birbirine bağlıdır (Kumar vd., 2020).

YSA'lar doğrusal olmayan problemler için metamodelleme yöntemleridir. Düğümlerin, katmanların ve eğitim yöntemlerinin sayısı bir YSA modelinin doğruluğunu etkileyebilir. Bu nedenle, bir sinir ağının topolojisinin nasıl belirleneceği, YSA yaklaşımı için önemli konulardan biridir. Öte yandan, düğümler ve bağlantılar içeren bir ağ grafiği, yani bir nedensel grafik, genellikle değişkenler ve hedefler arasındaki neden-sonuç ilişkilerini temsil etmek için kullanılır (Wu ve Wang, 2021).

Sinir ağları, paralel olarak çalışan basit işletim öğeleridir. Bu elementler biyolojik sinir sisteminden esinlenmiştir. Doğada, sinir ağlarının performansı, bileşenlerin nasıl bağlandığına göre belirlenir. YSA (Yapay Sinir Ağı), her bir bağlantının miktarı ağırlık olarak ayarlanarak bileşenler arasındaki bağlantıların ayarlandığı doğal bir sinir ağının yapay bir modelidir. Yapay sinir ağının unsurları arasındaki her bağlantının ağırlığını ayarlamaya eğitim denir. Genel olarak, ağırlıkları uygun şekilde ayarlamak için (aynı sinir ağı eğitimi ile) çok sayıda giriş ve çıkış çifti gerekir. Sinir ağları, karmaşık işlevlerin uygulanması, kanser teşhisi, MRI görüntü işleme, hastalık derecelendirmesinin sınıflandırılması ve EEG ve EKG analizi gibi birçok alanda kullanılmaktadır. Sinir ağlarının farklı türleri vardır ve veri türüne bağlı olarak bir veya daha fazla sinir ağı kullanılabilir. En yaygın sinir ağları, pozitron, Hopfield, çok katmanlı algılayıcı (MLP), radyal temel işlevi

(RBF) ve kendi kendini organize eden özellik haritasıdır. Her sinir ağının, o ağın gücünü belirleyen özel bir özelliği vardır (Kashani vd., 2020).

Yapay Sinir Ağlarının Uygulamalarından Örnekler

Yapay Sinir Ağları literatürde pek çok alanda uygulanmıştır. Effendy vd. (2020) çalışmalarında, bir giriş katmanı, bir veya iki gizli katman ve bir çıkış katmanı uygulayan çok katmanlı algılama ileri beslemeli sinir ağları kullanmışlardır. Giriş katmanındaki düğüm, herhangi bir pratik müdahale olmaksızın olası ilgili değişkenleri belirtmektedir. Burada sunulan çalışma, üretim parametrelerine dayalı olarak uzun vadeli bir etki enerjisi emilimi tahmini potansiyelini değerlendiriyor. Daha spesifik olarak, bu makalenin başlıca katkıları şu şekilde özetlenmektedir: (i) YSA girdisi olarak bir analitik model veri tabanının kullanılması; (ii) konsolidasyon parametreleri, termal bozunma kinetiği ve matris özelliklerinin bir fonksiyonu olarak darbe enerjisi absorpsiyonunun doğru tahminlerini elde etmek; ve (iii) geliştirilen YSA'nın doğruluğunu göstermek.

Gopal vd. (2021) çalışmalarında, yapay sinir ağı ve yanıt yüzeyi metodolojisi ve ön işlem kullanılarak çiçek atığının biyogaza dönüştüğü istatistiksel optimizasyona odaklanmaktadır. Biyogaz üretimi yanıt değeri deneysel değere çok yakındır. YSA tahmininin, deneysel tahmin ile karşılaştırıldığında oldukça etkili olduğu görülmüştür.

Gajewski ve Valis (2021) çalışmasında, kesintili modda çalışan sistemlerde yağ değişimi olmadan motor çalışma süresinin ne kadar uzatılabileceğini değerlendirmeyi amaçlamaktadırlar. Bu, özellikle kritik altyapı unsurları veya ara sıra iş yapan araçlar için geçerlidir. Uzunlamasına petrol numunesi verilerinin analizi, yağdaki partikül ve madde seviyelerindeki uzun vadeli değişikliklerin incelenmesini sağlamaktadır. Bu yeni veritabanı, ekibin eski araştırmasının doğrulanmasını destekleyecek ve gözden geçirilmiş sonuçlardan ortaya çıkan pratik sonuçları belirleyecektir. Veriler, güçlendirilmiş karar ağaçlarının ve yapay sinir ağlarının hesaplama güçlerini birleştiren sinerjiye dayalı bir yöntemle araştırılmıştır. On yıldan uzun süren araç testlerinden elde edilen yağ numunelerinin laboratuvar testlerinden karmaşık sinyal seçimi, optimum sistem durumu parametrelerinin oluşturulmasını kolaylaştıran faktördür. Önerilen güçlendirilmiş öğrenme ağacı (RLT) tabanlı model, özellikle tahmini veri madenciliği için yararlı olan bir yöntem olan stokastik gradyan artırılmış karar ağaçlarının bir uygulamasıdır. Araştırmacılar, RLT'yi yapay sinir ağı modellemesiyle birleştirerek, araştırılan teknik sistemin çalışma süresini tahmin etmek için etkili bir araç üretilmiş olmaktadır.

Pauletto vd (2020) çalışmalarında ilk kez kobalt (Co^{2+}), nikel (Ni^{2+}) ve metilen mavisinin (MB) tekli, ikili ve üçlü sistemlerden UM-kitine adsorpsiyonunda meydana gelen sinerjik ve antagonistik etkilerini YSA yaklaşımı ile incelemiştirler. Girdi verileri olarak her adsorbata başlangıç molar konsantrasyonu ve çözelti sıcaklığı dikkate alınarak,

tüm sistemlerdeki her adsorbatın adsorpsiyon kapasitesini eşzamanlı olarak tahmin etmek için optimal bir yapay sinir ağı geliştirmişlerdir.

Delgado vd. (2020) nin çalışmalarının amacı, kuruma-ıslatma döngüleri koşullarında beton numunelerinin klorür derinliğine nüfuz etmesini ve klorür difüzyonunu sağlamaktır. Deneysel sonuçlara dayanarak, analiz edilen değişkenler ile iyon penetrasyon derinliği arasındaki ilişkiyi haritalamak için Yapay Sinir Ağı (YSA) modellenmesi kullanılmıştır. Elde edilen sonuçlar, YSA modellemesinin betondaki klorür penetrasyon derinliğini ve klorür difüzyon katsayılarını tahmin etmede etkili olduğunu ve klorür penetrasyon derinliğini en çok etkileyen parametrelerin çimento türü, ekleme türü ve kürlenme süresi olduğunu göstermiştir.

Lee vd. (2021) çalışmalarında, kalıp döküm endüstrisinde akıllı fabrika platformu gerçekleştirmek için üç temel teknoloji geliştirmişlerdir: 1) bireysel ürün bilgileri sağlayan yüksek kaliteli proses verileri elde etmek için yeni bir uygun maliyetli ürün izleme teknolojisi, 2) gelişmiş proses verileri için işlem hatasını dikkate alan edinim sistemi ve 3) yapay sinir ağına dayalı bir hata algılama modülü. Kalıp döküm süreci için yeni geliştirilen sistemlerini, 1500 test üretimi kullanılarak doğrulamışlardır. Pilot üretim verilerine dayanarak, zaman serisi sıcaklık ve basınç ölçüm verilerinin ön işlenmesi ile bir arıza tespit modülü geliştirmişlerdir. Geliştirilen hata tespit modülü, eğitimsiz veriler için % 96,9 doğruluk göstermiştir. Araştırmacıların geliştirdikleri teknolojilerin, kalıp döküm endüstrisinde kusur oranını

ve üretim maliyetini düşürmek için gelecek vaat eden bir akıllı fabrika platformu olması beklenmektedir.

Pled vd. (2021) çalışmalarında önerdikleri sinir ağı tabanlı tanımlama yöntemi, rasgele uyumluluk alanının önceki bir stokastik modelinin hiperparametreleri ile çok ölçekli hesaplamanın bazı ilgili miktarları arasındaki doğrusal olmayan ilişkiyi öğrenmek için yapay bir sinir ağının eğitilebileceği bir veri tabanının oluşturulmasını gerektirmektedir. Girdi ve hedef verilerden oluşan birincil veri tabanı ilk olarak hesaplama modelinden üretilir ve buradan işlenmiş bir veri tabanı, parametrik olmayan istatistikler kullanılarak hedef verilere göre girdi verilerinin koşullandırılmasıyla çıkarılır. İki ve üç katmanlı ileri beslemeli yapay sinir ağları, daha sonra hiperparametreler (ağ çıktıları) ve ilgili miktarlar (ağ girdileri) arasındaki doğrusal olmayan eşlemenin cebirsel bir temsilini oluşturmak için ilk ve işlenmiş veritabanlarının her birinde eğitilmektedir.

Arslankaya (2020) çalışmasında, farklı bileşimlerle oluşturulmuş iki farklı alüminyum alaşımı (AA) 6061 çeşidine T6 ısıtma işlemi uygulandıktan sonra mekanik özellikleri ve mikroyapıları gözlemlenmiş ve daha sonra yapay sinir ağları (YSA) kullanılarak mekanik özelliklerini tahmin etmiştir. YSA'lar ile uzun ve maliyetli sürecin sonuçlarına daha kısa sürede ve daha düşük maliyetle ulaşılabilir. Bu nedenle, malzemenin akma dayanımı, çekme dayanımı, sertliği ve uzaması farklı YSA modelleri ile tahmin edilmiştir. Çalışmada farklı YSA modelleri arasında en düşük hataya sahip olan

model seçilmiş ve kullanılmıştır. Deneylerle elde edilen AA 6061'in mekanik özellikleri YSA eğitiminde kullanılmış ve YSA tahmin sonuçları deneysel sonuçlarla karşılaştırılmıştır. Geliştirilen YSA modeli, AA 6061'in mekanik özelliklerini % 90 oranında tahmin etmiştir.

Kompozit yapılara uygulanan yapay zekanın kullanımına ilişkin birçok çalışma bildirilmiş olsa da, çok azı YSA'nın termoplastik kompozitlerde tahminde bulunmak için bir teknik olarak kullanımına odaklanmıştır. Di Benedetto vd. (2020), yaptıkları çalışma, üretim parametrelerine dayalı olarak uzun vadeli bir etki enerjisi emilimi tahmini potansiyelini değerlendirmektedir. Daha spesifik olarak, araştırmacıların yaptıkları çalışmanın başlıca katkıları şu şekilde özetlenmektedir: (i) YSA girdisi olarak bir analitik model veri tabanının kullanılması; (ii) konsolidasyon parametreleri, termal bozunma kinetiği ve matris özelliklerinin bir fonksiyonu olarak darbe enerjisi absorpsiyonunun doğru tahminlerini elde etmek; ve (iii) geliştirilen YSA'nın doğruluğunu göstermek.

Gıda sektöründe bir YSA uygulaması da Baiocco vd. (2020) tarafından yapılmıştır. Çalışma, kirazların çekirdeklenme süreci için tahribatsız bir kalite kontrol önermektedir. Bir video kamera ve bir ışık kaynağından oluşan bir sistem, arkadan aydınlatmalı kirazların resimlerini kaydeder. MATLAB ortamında işlenen görüntüler, çukurun varlığını belirtmek için analiz edilen resimlerin sinamik histogramlarını sağlamaktadır. İleri beslemeli yapay sinir ağı uygulanan çalışmada elde edilen

histogramlarla eğitildi. Geliştirilen ağ, insan denetimi ile görülemeyen taş parçalarının hızlı bir şekilde tespit edilmesine ve uygun şekilde üretilmiş ürünlerin kazara reddedilmesinin azaltılmasına olanak tanımaktadır.

Kasık fitiği onarımı, en sık uygulanan cerrahi işlemlerden biridir. Gao vd. (2020), cerrahi sonuçların tahmini ve kasık fitiği onarımında risk faktörlerinin analizi için bir yapay sinir ağı (YSA) modeli geliştirmişlerdir. Amerikan Cerrahlar Koleji Ulusal Cerrahi Kalite İyileştirme Programı kapsamında hazırlanan çalışma kasık fitiği onarımı yapılan hastaları bulmak için kullanılmaktadır. Lojistik regresyon ve YSA modellerini kullanarak, alıcının çalışma karakteristik eğrilerinin altındaki alanı, gerçek-pozitif oranı, gerçek-negatif oranı, yanlış-pozitif oranı ve yanlış-negatif oranları kullanarak morbidite, geri kabul ve mortaliteyi değerlendirmişlerdir. YSA kullanılarak belirlenen morbidite, yeniden kabul ve mortalite sonuçları için belirlenen risk faktörleri, lojistik regresyon analizi ile tutarlı olmaktadır. YSA'lar kasık fitiği onarımı sonrası sonuçların tahmininde lojistik regresyon modelleriyle karşılaştırılabilir şekilde performans göstermişlerdir. Çalışmanın sonucunda YSA'ların, fitik cerrahisinin risk faktörü analizinde ve klinik uygulamalarda yararlı bir araç olabileceği belirtilmiştir (Gao vd., 2020).

Karmaşık sistemlerin çalışması, çevresel koşullar, ekipman aşınması veya belirli kısıtlamalar nedeniyle ürünlerin üretim aşamasında ilk tasarım koşullarından uzaklaşabilir. Buhar jeneratörleri karmaşık

ekipmanlardır ve düzgün çalışması ilgili parametrelerinin tanımlanmasına dayanmaktadır. Vieira vd. (2021) çalışmasında gerçek bir 360 MW enerji santralının kritik 6 adet buhar jeneratörünün operasyonel parametrelerini sıralamak için bir yaklaşım sunulmaktadır. Yapay Sinir Ağı - YSA, buhar jeneratörü verimliliğini, elektrik enerjisi üretimini ve baca gazı çıkış sıcaklığını tahmin etmek için 7 giriş parametrelili bir model sunmaktadırlar. YSA, 0.2015 ve 0.2741 eğitim hataları (ortalama mutlak ve kare hata) ve % 0.32 ve 2.350 (ortalama yüzde ve kare hata) doğrulama hataları ile iki yıllık bir veritabanı ile eğitilmiştir. Bu YSA modeli, Design of Experiment - DoE yaklaşımı tarafından önerilen durumların bir kombinasyonu aracılığıyla araştırılmıştır. 7 kontrollü parametrenin tümü, hem buhar jeneratörü verimliliğini hem de elektrik enerjisi üretimini ifade etmek için uygun olduğunun gösterildiği çalışmada, dinamik sınıflandırıcının birincil hava akış hızı, istatistiksel olarak anlamlı olmadığından baca gazı sıcaklığını hesaplamaya dahil edilmemiştir. DoE ayrıca baca gazı çıkış sıcaklığının hesaplanması için buhar jeneratörü verimliliği, elektrik enerjisi üretimi ve kömür kütle akış hızı açısından birincil hava basıncının önemini göstermektedir. YSA ve DoE'nin birleşik metodolojisi, karmaşık sistem verimliliğini artırmada anlamlı bir entegrasyon olmuştur (Vieira vd., 2021).

Üretim artışı sırasındaki proses kalitesi derecesi, bir ürünün yaşam döngüsünün başarısı için kritiktir. Gerekli proses kalitesine ulaşamazsa, yatırımlar yeniden çalışılarak düzenlenmelidir. Bu, daha fazla maliyete katkıda bulunur ve ürünün kar marjını azaltır. Ngo vd.

(2020) yaptıkları çalışma, üretim artışında süreç kalitesini sağlamada kalite ile ilgili süreç parametrelerinin konfigürasyonu için öneriler veren yapay zeka tabanlı bir karar modeli sağlamaktadır. Bir üretim simülasyonu yapılarak karar modelinin doğrulandığı çalışmada üretim artışına eşlik edecek karar modeli uygulanarak gerekli süreç kalitesine ulaşıldığı gösterilmiştir.

Gan vd. (2021), verimlilik tahmini için yapay sinir ağlarını kullandıkları bir çalışma yapmışlardır. Çalışmada, geri yayılım sinir ağı (BP), geri yayılıma dayalı genetik algoritma (GA-BP) ve aşırı öğrenme makinesi (extreme learning machine -ELM) yöntemleri; gümüş filmin parlaklığını ve bu nano-gümüş filmin elektrodepozisyonu için mevcut verimliliğini tahmin etmek için kullanılmaktadır. Yapay sinir ağları, sonuçları tahmin etmek ve deneysel sonuçlarla tezat oluşturmak için oluşturulmuştur. Tahmin sonuçları, geri yayılıma dayalı genetik algoritmanın basit geri yayılım algoritmasından nispeten daha yüksek bir doğruluk sergilediğini göstermektedir. ELM yöntemi ise verimli bir hız göstermiştir ve sonuçları üretmek için gereken süre geri yayılım algoritmasından ve geri yayılıma dayalı genetik algoritmadan çok daha azdır. Ve örnek verilerinin yeterli olduğu durumlarda yüksek doğruluğa sahiptir. Geri yayılıma dayalı genetik algoritma, yüksek doğruluk elde etmek için temel olarak yalnızca birkaç örnek veriye ihtiyaç duymaktadır. Araştırmacılar yaptıkları çalışma neticesinde, nano-gümüş filmin elektrodepozisyonu için öğrenme sürecinin doğruluk sırasını $ELM > GA-BP > BP$, analogik yetenek sırasını $GA-BP \approx BP > ELM$ ve gerekli zaman sırasını ise $GA-BP > BP > ELM$ olarak bulmuşlardır.

Mortalite ve hava kirliliđi arasındaki gnlk iliŐki endiŐe vericidir ve hava kirliliđinin mortalite ve solunum morbiditesi zerinde kısa vadeli etkileri olduđuna dair tutarlı kanıtlar bulunmaktadır. Hava kirliliđinin bu sađlık etkilerine iliŐkin dođru tahminler, ekonomik performanslarla ilgili eŐitli sektrlerin verimli planlanmasının yanı sıra insan sađlığını ve hava kalitesini iyileŐtirmeye ynelik stratejik ynetim iin de gereklidir. Khojasteh vd (2021) araŐtırmalarının temel amacı, Dickey-Fuller testi ile hava kirliliđinin solunum morbiditesi ve mortalitesi zerindeki uzun vadeli etkilerini belirlemek ve aynı zamanda dođrusal olmayan otoregresif ve yapay sinir ađı modelleri ile solunum morbiditesi ve mortalitesinin dođru tahminini geliŐtirmektir. Bu alıŐmayı gerekleŐtirmek iin araŐtırmacılar 9 yıllık sre boyunca Ahvaz Blgesinde hava kirleticilerine atfedilen solunum lmleri ve morbiditelerindeki gnlk deđiŐiklikleri incelemiŐlerdir. Sonular, nitrojen monoksit ve karbon monoksitin toplam solunum lmleri zerinde nemli bir etkiye sahip olduđunu gstermiŐtir. Duyarlılık analizi, diđer kirleticilerin (NO₂, SO₂, O₃, PM₁₀) toplam solunum morbiditesi ve mortalite oranı zerinde anlamlı bir etkisinin olmadıđını gstermiŐtir. Dođrusal olmayan otoregresif model iin, nitrojen oksit ve karbon oksidi ieren iki giriŐli topoloji (girdi katmanında 2 proses elemanı, 10 ara katman ve 1 proses elemanlı ıktı katmanı olan bir ađ topolojisine sahip (2-10-1)) iin, toplam lm oranını tahmin etmek iin en iyi 10 gizli katman (MSE = 0.1 ve R = 0.82) Őeklinindedir. Yapay sinir ađı ve dođrusal olmayan otoregresif modeller, en az  girdi ile solunum lmleri ve hareketliliđinin dođru tahmini iin ok gl yntemlerdir. Bu bulgular, politika yapıcıların evredeki karbon

monoksit ve nitrojen monoksit konsantrasyonlarını azaltmak için hedefler belirleme ihtiyaçlarını güçlü bir şekilde destekleyecek sonuçlar içermektedir.

Sayısal simülasyon teknolojisine dayalı olarak çalışan bir başka çalışmada Zhang vd. (2019) tarafından yapılmıştır. Yapay sinir ağı (YSA), genetik algoritma (GA) ve bu iki yöntemin entegrasyonu, yerel iç mekan hava kalitesini iki tipik havalandırma modeli ile kontrol etmek için ayrı ayrı kullanılmıştır. YSA yöntemi, kontrol değişkeni ile amaç arasında bariz bir korelasyon ilişkisi olduğunda daha yüksek verimlilik göstermiştir. Bariz korelasyon ilişkisi olmadığında, GA ve YSA'nın entegrasyonu tek başına GA'dan daha verimli ve tek başına YSA'dan daha doğru olduğunu göstermiştir.

Ghoniem vd. (2020)'nin çalışması: pasif normal süspansiyonu, düşük maliyetli yeni bir yarı aktif süspansiyonla değiştirerek araçlar için yeni bir süspansiyon sistemi önermektedir. Yeni önerilen sistem, iki giriş portu arasına piston silindirinin dışına monte edilmiş orantılı gaz kelebeği valfli bir hidrolik pistondan oluşur. Gaz kelebeği, silindirin yağdaki hidrolik enerjiyi dağıtmak için kullanılacaktır. Gaz kelebeği valfi, gerekli eşdeğer sönümlenme katsayısını elde etmek ve valf açıklığını düzenlemek için yapay sinir ağı (YSA) kullanacaktır. Önerilen sistem ilk olarak bir denetleyici tarafından kontrol edilecek ve yapay sinir ağını eğitmek için gereken verileri elde etmek için farklı durumlar simüle edilecektir. Önerilen sistem, piyasada bulunan manyeto-reolojik (MR) sönümlenme sistemlerine kıyasla, damperlerin

maliyetinin dörtte birine kadar düşük bir maliyetle kolayca inşa edilir. MATLAB / SIMSCAPE programları önerilen süspansiyon sisteminin fiziksel modelini oluşturmak için kullanılmaktadır. Aracın önerilen düşük maliyetli yarı aktif süspansiyon sistemi ile tepkisi, önerilen düşük maliyetli yarı aktif süspansiyon sisteminin etkinliğini değerlendirmek için geleneksel süspansiyon sistemi ve MR süspansiyon sistemi ile karşılaştırılacaktır. Sonuçlar, MR yarı aktif sistemler ve geleneksel sistemlere kıyasla titreşim azaltma ve yanıt süresi açısından iyi performans göstermektedir (Ghoniem vd., 2020).

KorkusuzPolat ve Arslankaya (2010), maliyet bileşenlerini girdi olarak kullanarak üretim miktarını tahmin eden bir YSA modeli oluşturmuşlardır.

Bir istatistik hesaplama programı olan SPSS, kullanıcının, mevcut iki işlevden biri olan Çok Katmanlı algılayıcı (MLP) veya Radyal Taban İşlevini (RBF) kullanarak istenen çıktıyı tahmin etmesini sağlayan bir yazılım olarak da kullanılabilir. Bu işlevler, tahminin doğruluğunu sağlamak için modelin tahmin edilen sonuçlarını hedef değişkenlerin bilinen değerleri ile karşılaştırmaktadır. Sharma ve Kushvaha (2020) çalışmalarında , çok katmanlı algılayıcı işlevi ileri beslemeli mimari ile bağlantılı olarak kullanılmışlardır. Bu mimari üç katman kullanılmaktadır (girdileri içeren giriş katmanı, gözlemlenemeyen düğümlerden oluşan gizli katman ve son yanıtları veren çıktı katmanı). Her katmanda bulunan ve nöron olarak bilinen bazı işlem birimleri vardır. YSA, karmaşık doğrusal olmayan verileri en yüksek doğrulukla

haritalamak için bu nöronlar arasındaki bir dizi ağırlıktan yararlanmaktadır. Bir katmanın ağırlıkları, aktivasyon fonksiyonları aracılığıyla bir sonraki katmana bağlanır. SPSS, ağırlık ayarlamasıyla ilgili hatayı hesaplamak ve en aza indirmek için uygun eğitim türünün seçilmesine izin vermektedir.

Gupta ve arkadaşları (2019), atölyedeki üretim süresini en azlamak için YSA'nın farklı algoritmalarını kullandıkları bir çalışma yapmışlardır.

Yapay sinir ağları konusunda yapılmış farklı uygulama örnekleri için Martin vd. (2007), Shin vd. (2020) incelenebilir.

Yapay Sinir Ağlarının Çalışma Prensipleri

YSA'nın ana bileşenleri girdi, gizli ve çıktı katmanlarıdır. Geri yayılma eğitim algoritması, ara bağlantı ağırlıkları ve nöron eşikleri ile ilgili aşağıdaki maliyet fonksiyonunu yinelemeli olarak en aza indirmek için yaygın olarak kullanılır (Fayed vd., 2019).

Yapay sinir ağlarında, daha iyi yakınsama ve sayısal kararlılık elde etmek için giriş ve çıkış değerleri normalleştirilir. Her bir fiziksel değer x_i , sırasıyla minimum ve maksimum değerler olarak x_{ni} ve x_{imin} ve x_{imax} ile birleşik değerlere x_{ni} yol açan normalleştirilmiş bir x_{ni} değerine dönüştürülür. Aşağıda normalizasyon formülü gösterilmektedir (Stoffel vd., 2020).

$$x_i = 0,1 + 0,8 \cdot \left(\frac{x_i - x_{imin}}{x_{imax} - x_{imin}} \right)$$

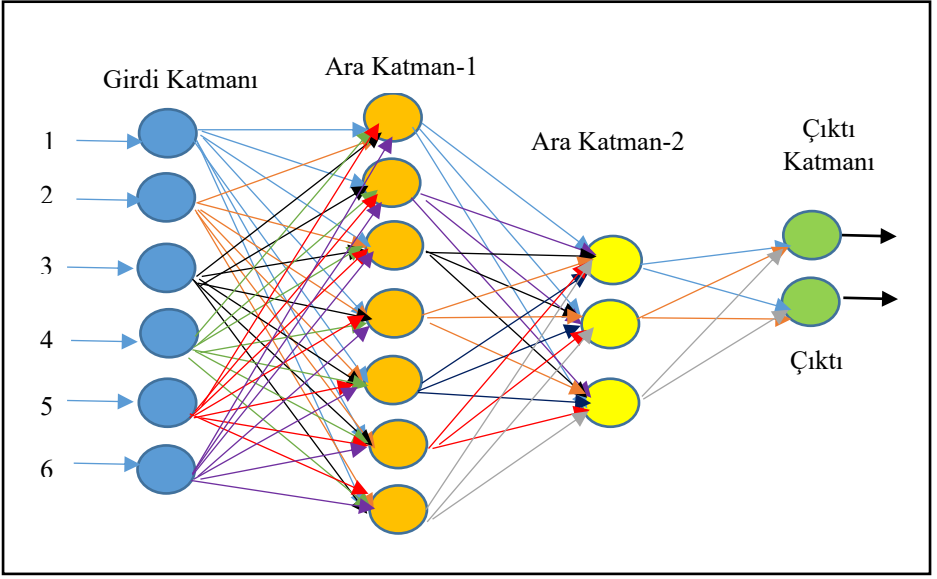
(Formülde i indeksi i. giriş nöronunu ifade eder)

Sinir ağının mimarisi, giriş (X değişkeni) ve çıkış katmanı (Y değişkeni) içeren bir gizli katmanı içermektedir. YSA mimarisinde düğüm, giriş verilerini alır ve verilere aktivasyon fonksiyonu uygulayarak basit işlemler gerçekleştirir. Her bir düğüm, ağırlıklı bağlantı değeri ile birbirine bağlanmaktadır. Gizli katman, ağırlıkla genelleştirilmesini sağlayan araç olarak düşünülebilir. Düğümleri olan bir gizli katman, sürekli işlevin yaklaştırılması için kullanışlıdır. Optimum sayıda gizli düğüm seçimi sonuç üzerinde daha etkili olmaktadır. Gizli düğümler, orijinal girişlerin doğrusal olmayan işlevleridir. YSA da bir de çıkış düğümü kullanılmaktadır. Gizli katman düğümlerine sahip bağlantıların sayısı YSA'nın karmaşıklığını gösterir, çünkü YSA'daki bağlantıların sayısını artırırlar. Gizli katmanların düğümlerinde uygulanan işlevlere etkinleştirme işlevleri denir. Aktivasyon fonksiyonları, belirleyici bir düğümün çıktısını sağlayan matematiksel formüllerdir. Doğrusal olmayan fonksiyonlarda, hiperbolik tanjant (TanH) uygulanmaktadır. Sürekli farklılaştığı için yaygın olarak kullanılan bir sigmoid işlevidir ve ağ öğrenimi için istenen bir özelliktir. Aktivasyon fonksiyonu, YSA'nın verimliliğini etkiler, çünkü giriş ve çıkış parametreleri arasındaki olguyu simüle etmenin tek yolu budur. (Banga vd., 2020).

Girdi verilerinin işlenmesi matematiksel transfer fonksiyonları aracılığıyla yapılır. Üç temel transfer işlevi türü genellikle ileri beslemeli ağlarda kullanılır: (i) hiperbolik tanjant veya TANSIG; (ii) lojistik sigmoidal veya LOGSIG ve; (iii) doğrusal fonksiyon veya PURE-LIN. TANSIG artı eksi sonsuz aralığında girdi alır ve 1 ile + 1 aralığında bir çıktı sağlar. TANSIG, gıda endüstrilerinde hat içi işleme gibi daha hızlı ağların istendiği yerlerde tercih edilir. Buna karşılık, LOGSIG, 0 ila 1 aralığında çıktı üretir ve varyasyonun oldukça büyük olduğu durumlarda tercih edilir (Kumar vd., 2020).

YSA'nın yapısını belirlemek için nedensel grafiği kullandıktan sonra, YSA'nın yaklaşıklık doğruluğunu iyileştirmek için ara değişkenlerin değerleri kullanılır. Bu nedenle, tasarımcının tasarım problemini sadece üst düzey neden-sonuç ilişkilerini türetebilecek kadar bildiği varsayılır. Bu öncülle, özellikle yüksek boyutlu mühendislik tasarım problemleri için metamodellerin doğruluğunu artırma konusunda ara değişkenlerin neden-sonuç bilgisini ve değerlerini içeren bir nedensel YSA tanımlanır ve geliştirilir (Wu ve Wang, 2021).

Yapay Sinir Ağının pek çok çeşidi bulunmaktadır. Geri beslemeli yapay sinir ağları da bunlardan biridir. 1 girdi katmanlı, 2 ara katmanlı ve 1 çıktı katmanlı bir geri beslemeli yapay sinir ağının şematik yapısı şekil 3.1'de gösterilmektedir.



Şekil 3.1. Geri Beslemeli Yapay Sinir Ağı'nın genel yapısı

Şekil 3.1'de gösterilen geri beslemeli yapay sinir ağında girdi katmanında 6 adet proses elemanı, birinci ara katmanda yedi adet proses elemanı, ikinci ara katmanda üç adet proses elemanı ve çıktı katmanında ise iki adet proses elemanı bulunmaktadır.

Geri Beslemeli Yapay Sinir Ağları çalışma prensibi hakkında detaylı bilgi için Stoffel vd., 2020; Chen vd., 2020; Wu ve Wang, 2021; Lella ve Vessio, 2020 kaynaklarından yararlanılabilir

Yapay sinir ağları genellikle bir çıktı katmanından, bazı gizli katmanlardan ve bir girdi katmanından oluşmaktadır. Bilgi ileri ve geri yönde akar ve bu da yapay sinir ağına dinamik bir hafıza sağlar. Nöronlar (proses elemanları), verilen girdi katmanına göre çıktı katmanını oluşturacaktır. İlk katman girdi katmanı, ikincisi gizli olan

katman (birden fazla sayıda olabilir) ve üçüncü katman çıktı katmanıdır (Gopal vd., 2020).

Bir giriş katmanı, bir veya daha fazla gizli katman ve bir çıktı katmanı içeren tipik bir YSA mimarisi Şekil 1'de gösterilmektedir. Genellikle, keşif değişkeni giriş katmanı verileri olarak ve hedef değişken de çıktı katmanı verileri olarak kullanılır. Girdi ve çıktı katmanı arasındaki bilinmeyen ve karmaşık ilişki, gizli katmanın aktivasyon fonksiyonları ve sayısal ağırlıkları tarafından oluşturulur (You vd., 2020).

BP-YSA (Geri Beslemeli Yapay Sinir Ağı), önemli ileri beslemeli YSAlardan biri olarak, bir tür girdi, gizli ve çıktı katmanlarını içeren çok katmanlı ağ. Hata geri yayılım algoritması BP-YSA'nın anahtarıdır, BP algoritması YSA eğitim verilerini öğrenene kadar bu hataları azaltır. Genel olarak, bir girdi katmanı, bir gizli katman ve bir çıktı katmanından oluşan tek bir BP-YSA birçok sorunu çözebilir (You vd., 2020).

Bir nöron, birkaç girdinin tek bir çıktıya doğrusal olmayan bir eşlemesidir. Belirli bir katmanın her nöronu, önceki katmanın her nöronuna bağlanır. Bir nöron, aktivasyon işlevi ile karakterizedir. Katman $n + 1$ 'de her nöron, katman n 'deki tüm nöron çıktılarından x_i girdileri alır. Bu girdiler ağırlıklandırılır ve toplanır. Formülasyon aşağıda verilmektedir (Iseli ve Schiffmann, 2021):

$$v = \sum_{i=1}^m w_i x_i + b$$

formülasyonda m , nöron giriş değerlerinin sayısıdır, w_i her bir girişin ağırlıklarıdır ve b , "önyargı" olarak da adlandırılan ek bir sabit terimdir. v değeri, nöronun potansiyeli olarak adlandırılır. Nöronun çıktısı, doğrusal olmayan bir aktivasyon fonksiyonu g ve potansiyel v ile hesaplanır. Örneğin hiperbolik tanjant sigmoid aktivasyon fonksiyonu kullanılacaksa g değeri aşağıdaki gibi hesaplanır (Iseli ve Schiffmann, 2021):

$$g(v) = \frac{2}{1 + e^{-2v}} - 1$$

Bu etkinleştirme işlevi tüm ara (gizli) katmanlar için kullanılır. Son tahmin edilen değer y_p , doğrusal bir aktivasyon işlevine sahip olan çıkış nöronu tarafından hesaplanır (Iseli ve Schiffmann, 2021):

$$y_p = cv$$

burada c bir sabittir ve v yukarıda verilen denklem ile hesaplanır (Iseli ve Schiffmann, 2021).

Geri beslemeli ağı eğitmek için, w_i ve bias ağırlıklarının, çıktının yanıt değişkenini tahmin edeceği şekilde ayarlanması gerekir. Model tahmin hatası, eğitim sırasında en aza indirilmesi gereken Ortalama Karekök Hatası (RMSE) ile ölçülebilir. Hangi ağı en iyi performansa sahip

olduđuna karar vermek için, test senaryoları için kesin çözümlere kıyasla kesit momentinin tahmin edilen kısa vadeli uç değerlerinin Ortalama Karekök Hatası (RMSE) kullanılır (Xu vd., 2020).

Toplama seviyelerinde, Ortalama Karekök Hatası (RMSE) bir doğruluk ölçüsü olarak hesaplanır. Regresyon çalışmasının katsayısı, çok yıllık ölçümün (R^2) tahmin doğruluğunun bir göstergesi olarak dahil edilmiştir.

$$RMSE = \sqrt{(e - r)^2}$$

Burada e: beklenen değerleri, r: gözlenen değerleri (bilinen sonuçların) göstermektedir (Li, 2020).

RMSE ayrıca şu şekilde de tanımlanır (Iseli ve Schiffmann, 2021).

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_p^{(i)} - y^{(i)})^2}$$

Formülasyonda N örnek sayısıdır, y_p ve y tahmin edilen ve örnek çıktı değerleridir. RMSE, normalleştirilmiş bir özellik olmadığında farklı çıktı parametrelerini karşılaştırmak için ideal değildir. Bu nedenle, normalleştirilmiş kök ortalama kare hatası NRMSE kullanılır ve bu şu şekilde tanımlanır (Iseli ve Schiffmann, 2021):

$$NRMSE = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N (y_p^{(i)} - y^{(i)})^2}}{y_{max} - y_{min}}$$

Formülasyonda y_{max} ve y_{min} , tüm doğrulama veri setindeki maksimum ve minimum değerlerdir. Yukarıdaki eşitlik: örnek aralığı ile normalize edilen kök ortalama kare hatasını temsil eder.

BÖLÜM 4. ÖRNEK UYGULAMA

Uygulama otomobil parçaları üreten bir yan sanayi firmasında yapılmıştır. Üretilen ürünlerden birisi “çarpışma önleyici tampon” dur. Bu ürünün müşterisi, üretim esnasında preste şekil alma prosesinden sonra %100 ölçü kontrolü yapılmasını istemektedir.

Tandem pres hattının başında işlenmemiş halde bulunan çelik plaka rulosu sırasıyla; istenilen ölçüde kesme, bükülme, delik delme işlemlerinden geçer ve kasalara doldurulur. Kalite personeli üretilen ürüne ait ölçü kontrol fikstürü ile her parçanın gerekli ölçümlerini yapar, sonuçları ‘Sample Data Sheet’ adını verdikleri, ürünü satın alacak firma tarafından hazırlanan forma kaydeder, her ölçüm için spesifikasyon sınırlarını kontrol eder ve formda ürünün geçerlilik durumunu belirtir.

Ancak işletmede kağıtların kaybolması veya hasar görmesi durumunda verilere ulaşımın zorlaşması, kağıt israfı ve oluşturulan klasörleri saklama alanının yetersizliği gibi sorunlar baş göstermiştir. Üstelik kalite operatörü için her değer kağıt forma yazılması ve kabul veya red durumuna karar verilmesi gereksiz işyükü oluşturmaktadır. Benzer şekilde formdaki verilerin bilgisayar ortamına aktarılması da ofis çalışanları için gereksiz bir çalışma yükü oluşturmaktadır.

Bu çalışmada, işletmedeki sorunları ortadan kaldırmak hedeflenerek, verilerin operatörler tarafından bilgisayara kaydedildikten sonra kabul veya red kararının verilmesinde yardımcı olacak bir sistem geliştirilmesi amaçlanmaktadır.

Bunun içinde öncelikle mevcut durum incelenecektir. Mevcut durum analizinden sonra MATLAB programında Levenberg-Marquardt's Algoritması kullanılarak yapay sinir ağı oluşturulacaktır. Sonraki aşamada ise MATLAB için GUI kullanıcı arayüzü oluşturulacaktır. Elde edilen sonuçlar istatistiksel olarak test edilerek lojistik regresyon ile değerlendirilecek ve yorumlanacaktır.

4.1. Mevcut Durumun İncelenmesi

Firmada “çarpışma önleyici çelik tampon” ürünü için yıl boyunca seri üretim yapılmaktadır. Çarpışma önleyici çelik tampon için pres makinesinde şekil verildikten sonra kalınlık, incelme miktarı, ağırlık, delik çapı ve uzunluk ölçümleri yapılmaktadır. Ölçüm değerleri alıcı firma tarafından gönderilen ve her ürün için kaydedilmesi istenen ‘Sample Data Sheet’ adını verdikleri formlara işlenmektedir. Ölçüm değerleri istenilen sınırlar içindeyse ürün için kabul, değilse red kararı verilir.

“Çarpışma önleyici çelik tampon” ürününün alıcı firması tarafından talep edilen ve verilerin kaydedildiği form (örnek veri toplama formu) örneği 8 örnek seti için tablo 4.1’de gösterilmektedir.

Tablo 4.1. Örnek veri toplama formu

Tedarikçi Adı:		ÖRNEK VERİ TOPLAMA FORMU							
Tedarikçi Kodu:									
Model:									
Parça Adı:									
Parça Numarası:									
Kontrol Ekibi:									
Tarih:									
ÖLÇÜ KONTROL		Ör-1	Ör-2	Ör-3	Ör-4	Ör-5	Ör-6	Ör-7	Ör-8
	Kalınlık (mm) 4,5± 0,5	4,8	3,9	5,0	4,5	4,65	5,22	5,22	3,8
	İncelme (mm) 0,2± 0,02	0,2	0,1 8	0,21	0,17	0,19	0,24	0,2	0,16
	Ağırlık (gr) 50±4	50, 7	47, 6	49,4	55,1	49,3	52	46	53,3
	Delik Çapı (cm) 2,7± 0,3	2,4 8	2,7	2,83	3	2,22	2,37	2,42	3,2
	Uzunluk (cm) 21± 0,5	21, 3	20, 6	21	20,4 3	20,9	21,7	20,4 4	21,6 6
	KABUL (K) / RED (R)	K	R	K	R	K	R	R	R

Üretim devam ederken, bir yandan ürün kontrolünün yapılması bir yandan da ölçüm sonuçlarının kağıtlara işlenmesi üretim sahasında karmaşaya, çalışan üzerindeki iş yükünün artmasına, verilerin güvenilir olarak elde edilememesine sebep olmaktadır.

Bu nedenle tahminleme çalışması için MATLAB programında yapay sinir ağı modeli oluşturularak problemin çözümü için alınan girdi ve hedef verileri ile makine eğitilecek ve test edilecektir. Sonrasında kalite personelinin kullanması için bir arayüz tasarımı yapılacaktır.

4.2. Yapay Sinir Ağı Oluşturulması ve Eğitimi

Yapay sinir ağı oluşturulması, eğitilmesi, belirlenen hedef değerleri ile elde edilen çıktı değerlerinin karşılaştırılması ve sonuçların değerlendirilmesi süreci öncelikle ölçüm değerlerinin alınması faaliyeti ile başlayacaktır. Ölçüm değerleri firmada kullanılan örnek veri toplam formu ile alınan değerler kullanılarak belirlenecektir. Daha sonra girdi ve hedef veri seti oluşturulacaktır. Girdi veri seti normalize edilecek ve YSA ile eğitilecektir. Oluşturulan YSA test edilecek ve sonuçlar değerlendirilecektir.

Örnek veri toplam formu ile takip edilen “Çarpışma önleyici çelik tampon” ürününe ait 1050 adet üretim için ‘kalınlık, incelme, ağırlık, delik çapı, uzunluk’ değerleriyle hazırlanan 5x1050 boyutlarında girdi ile veri seti oluşturulmuştur. Verilerin makineye tanıtılmasında kabul

edilen deęerler 0 ile 1 arasında olduęundan girdi verileri normalize edilmiřtir.

Ürün ölçüm deęerleri istenilen spesifikasyon deęerleri aralıęında ise ürün kabul edilir ve hazırlanan hedef veri setinde 1 deęerine karşılık gelir. Aksi durumda ürün reddedilecekse hedef deęeri 0' dır.

Hedef veri seti 0 ve 1 deęerleriyle 1x1050 boyutlarında oluşturulmuřtur.

Oluřturulan 1050 adet örnek veri setlerinin bir kısmı tablo 4.2'de gösterilmektedir.

Tablo 4.2. Girdi verileri (1050 adetten örnek olarak 15 adet)

Örnek	Kalınlık	İncelme	Ağırlık	Delik Çapı	Uzunluk
1	4,00	0,21	50,70	2,75	21,24
2	5,00	0,18	48,83	2,91	20,64
3	4,58	0,21	46,02	2,85	21,46
4	4,01	0,21	52,68	2,60	21,00
5	4,70	0,19	51,16	2,86	20,72
6	3,99	0,22	53,05	2,64	20,84
7	3,98	0,18	53,28	2,40	21,14
8	3,95	0,20	53,09	2,59	21,22
9	3,96	0,20	49,69	2,68	20,64
10	3,92	0,22	51,13	2,86	21,13
11	3,89	0,19	50,99	2,46	20,86
12	3,85	0,18	53,25	2,81	21,30
13	3,86	0,19	49,96	2,99	20,80
14	3,81	0,20	50,39	2,94	21,23
15	3,80	0,19	46,23	2,55	21,33

Tablo 4.2’de gösterilen verilerin normalize edilmiş halleri tablo 4.3’de verilmektedir.

Tablo 4.3. Normalize edilen girdi verileri

Normalize Değerler					
Örnek	Kalınlık	İncelme	Ağırlık	Delik Çapı	Uzunluk
1	0,52	0,42	0,72	0,40	0,05
2	0,69	0,32	0,65	0,48	0,03
3	0,62	0,40	0,55	0,45	0,06
4	0,52	0,41	0,79	0,33	0,04
5	0,64	0,35	0,74	0,46	0,03
6	0,51	0,43	0,80	0,35	0,04
7	0,51	0,33	0,81	0,24	0,05
8	0,51	0,38	0,80	0,33	0,05
9	0,51	0,39	0,68	0,37	0,03
10	0,50	0,44	0,73	0,46	0,05
11	0,50	0,34	0,73	0,26	0,04
12	0,49	0,32	0,81	0,44	0,05
13	0,49	0,35	0,69	0,52	0,04
14	0,48	0,36	0,71	0,49	0,05
15	0,48	0,35	0,56	0,31	0,05

Tablo 4.3. de gösterilen örnek setinin normalize edilmiş değerleri için hedef veri seti tablo 4.4’de gösterilmektedir (Kabul edilen örnekler için: 1, red edilen örnekler için: 0 olacak şekilde).

Tablo 4.4. Hedef veri seti

Örnek	Hedef
1	1
2	1
3	1
4	1
5	1
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0

Matlab programında oluşturulacak olan ađ; girdi katmanında her biri kalınlık-incelme-ađırlık-delik apı-uzunluk deđerlerini gösterecek şekilde beş proses elemanı (nöron), ıktı katmanında ise ürünün kabul veya red durumunu gösteren bir proses elemanı olan bir modeldir.

Matlab workspace alanına girdi, hedef ve test verileri komut penceresine eklendikten sonra “nntool” komutu ile açılan pencereden input data, target data olarak kaydedilir ve ağ oluşturulmaya başlanır.

Beş girdi ve bir çıktıya sahip bir ağ modeli oluşturmak için; nöron sayısı 6 ve 12 arasında olan ara katmanlı farklı YSA modelleri oluşturulmuştur. Bu modeller tablo 4.5’de gösterilmektedir (MSE değeri ortalama hatanın karesini vermektedir).

Tablo 4.5. Nöron sayılarına göre YSA eğitim sonuçları

Deneme	Eğitim Algoritması - Nöron Sayısı	R- Değeri	MSE
1	Levenberg-Marquardt - 7	7,57E+04	1,06E+04
2	Levenberg-Marquardt - 8	7,39E+04	1,14E+04
3	Levenberg-Marquardt - 9	7,44E+04	1,11E+04
4	Levenberg-Marquardt - 10	7,83E+03	9,64E+03
5	Levenberg-Marquardt - 11	8,02E+03	8,01E+04

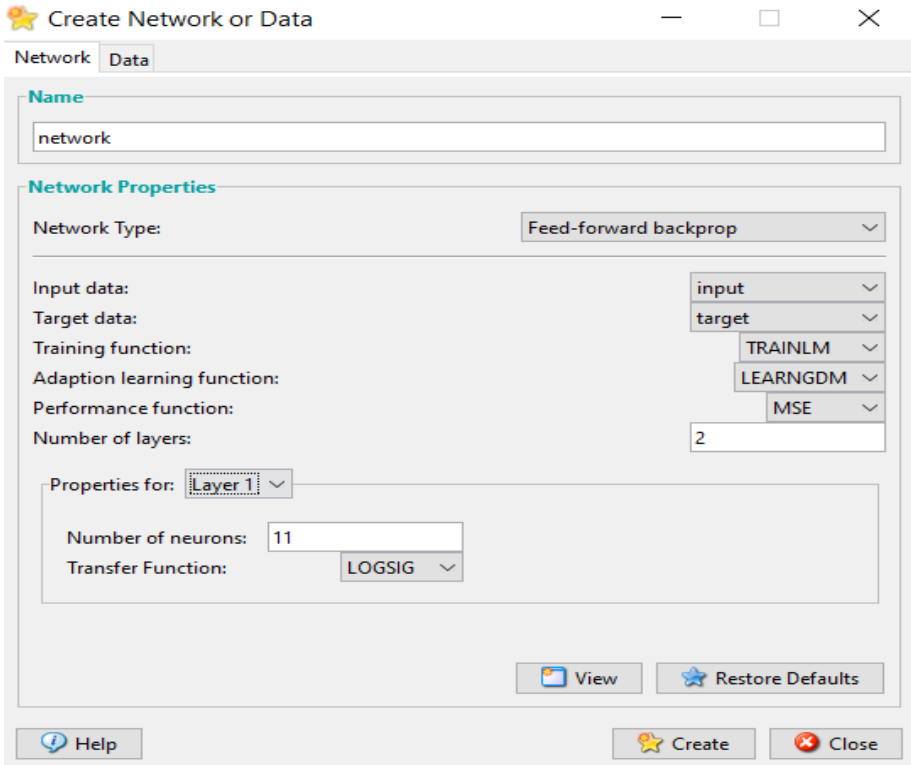
Tablo 4.5’de gösterilen farklı yapıdaki YSA modellerinden, R değeri en büyük ve MSE (Ortalama Hata Kare) değeri en küçük olan model, Levenberg-Marquardt eğitim algoritmasıyla eğitilmiş ve ara katmanda 11 nörona sahip olan ağ modelidir.

MATLAB YSA araçlarıyla oluşturulan ve eğitilecek sinir ağının parametreleri tablo 4.6’da gösterildiği gibidir.

Tablo 4.6. Ağ parametreleri

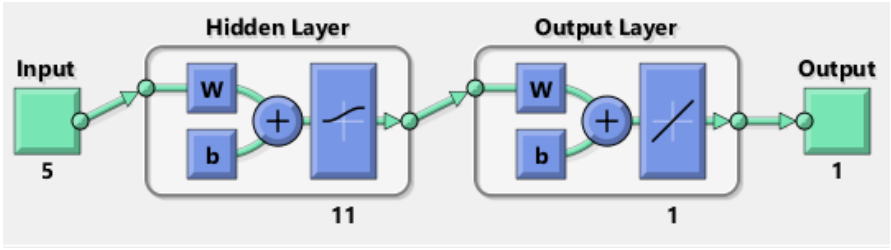
Ağ yapısı	Çok Katmanlı İleri Beslemeli Ağ
Eğitim Fonksiyonu	Trainlm
Öğrenme Fonksiyonu	Learngdm
Öğrenme Kuralı	Geri Yayılımlı Öğrenme
Performans fonksiyonu	MSE
Gizli Katman Sayısı	11
Gizli Katmanlar için Transfer Fonksiyonu	Logsig
Çıktı Katman için Transfer Fonksiyonu	Logsig

Belirlenen ağ parametrelerine göre girdi ve hedef verileri Matlab YSA aracına yüklenerek, şekil 4.1' deki gibi sinir ağı oluşturulur.



Şekil 4.1. Matlab YSA aracı ağ oluşturma ekranı

Oluşturulan ağın yapısı (girdi katmanında beş proses elemanı, bir ara katman ve ara katmanda 11 proses elemanı, çıktı katmanında ise bir proses elemanı olacak şekilde) şekil 4.2’de gösterilmektedir.



Şekil 4.2. Matlab’da oluşturulan ağıın yapısı

Firmanın üretimi esnasında toplanan 1050 adet üretime ait verinin %70’i eğitim, %15’i geçerlilik ve %15’i ağı test etmek için kullanılmak üzere rastgele bölünmüştür. Verilerin dağılımı şekil 4.3’de gösterilmektedir.

Validation and Test Data
Set aside some samples for validation and testing.

Select Percentages

Randomly divide up the 1050 samples:

Training:	70%	734 samples
Validation:	15%	158 samples
Testing:	15%	158 samples

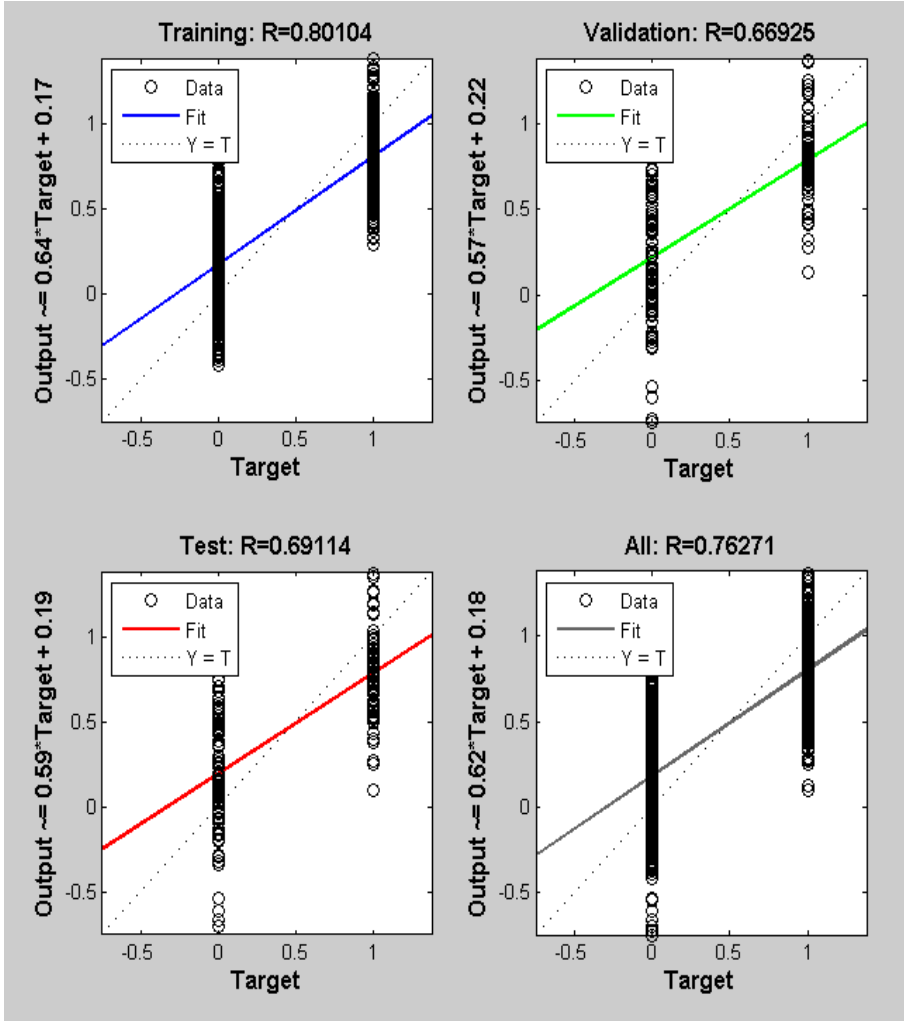
Şekil 4.3. Verilerin dağılımı

Oluşturulan YSA modelinin Matlab programındaki eğitim değerleri şekil 4.4’de gösterilmektedir.



Şekil 4.4. YSA Eğitimi

Matlab programında yapılan YSA eğitime ait regresyon sonuçlarının grafikleri ise şekil 4.5’de gösterilmektedir.

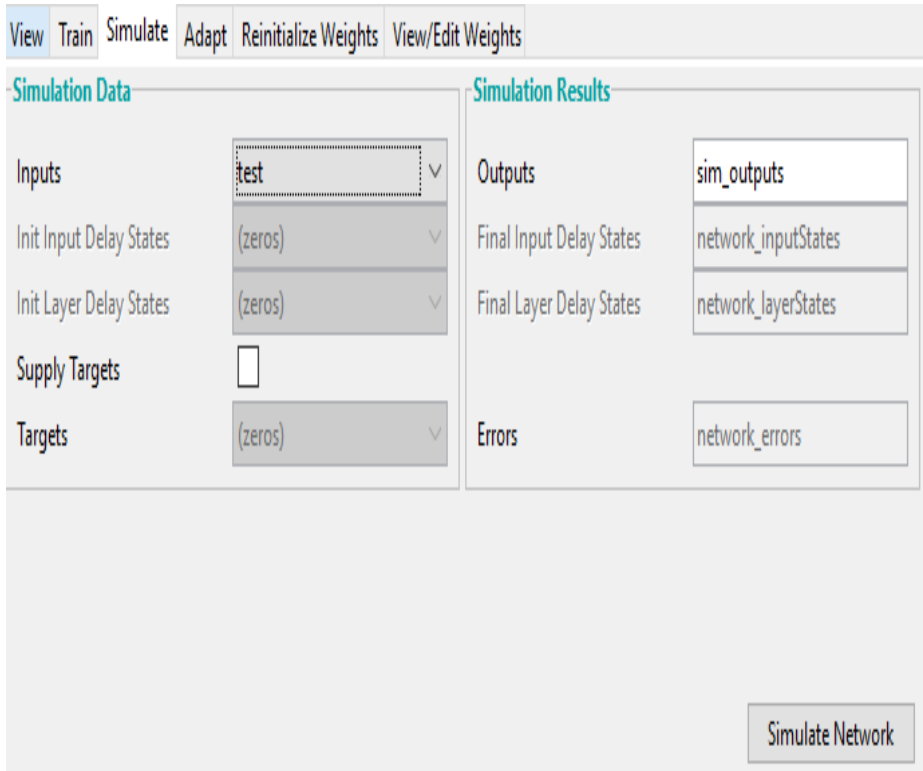


Şekil 4.5. YSA Regresyon Grafikleri

Regresyon değerleri ağ performansını doğrulamak için kullanılmaktadır. Şekil 4.5’deki regresyon grafikleri ağ çıktılarını;

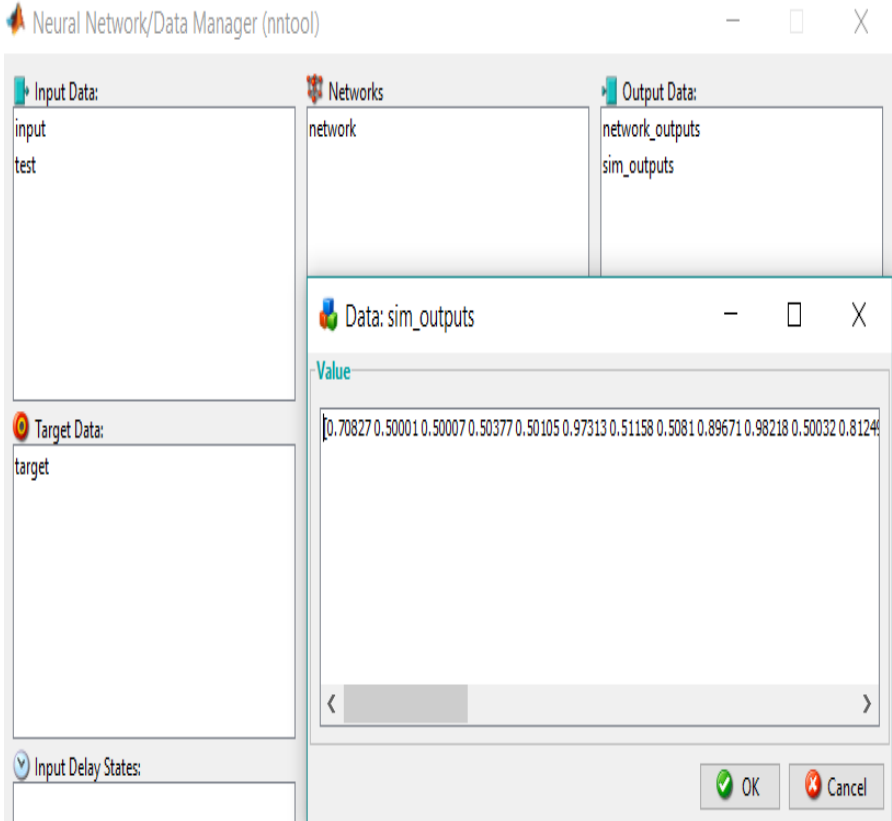
eđitim-geçerlilik-test grupları için hedef deđerlerine göre ayrı ayrı deđerlendirmektedir. Ađ performansı arttırılmak istenirse, ađ yeniden eđitilebilir.

Yapay sinir ađı eđitildikten sonra, eđitim verilerinden farklı hazırlanan örneklerle test edilir. Bunun için yeni girdi verileri yüklenir ve çıktıların “sim_outputs” adıyla yüklenmesi istenir. Matlab programı için ekran görüntüsü şekil 4.6’da verilmektedir.



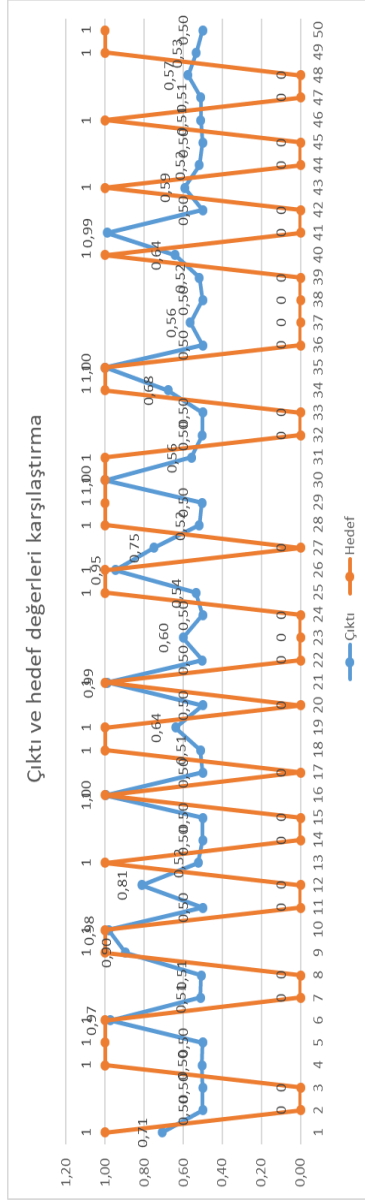
Şekil 4.6. Sinir ađının simülasyonu

5x50 boyutlarında hazırlanan test veri setiyle simüle edilen ağın çıktıları şekil 4.7’de verilmiştir.



Şekil 4.7. Simülasyon çıktıları

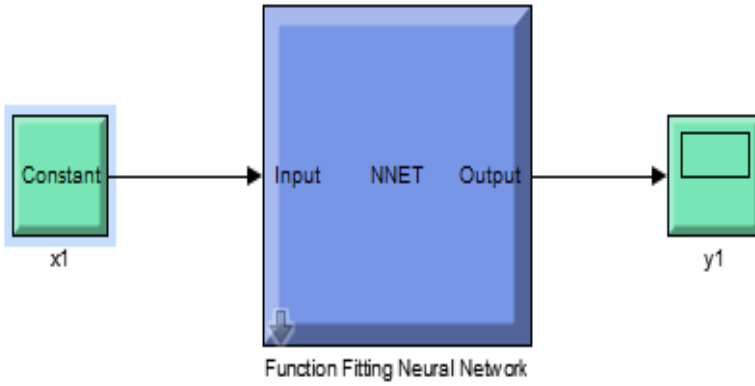
Örneklerden beklenen hedef değerlerin ve YSA’dan elde edilen çıktı değerlerinin karşılaştırması şekil 4.8’de gösterilmektedir.



Şekil 4.8. YSA çıktı değerlerinin hedef değerler ile karşılaştırılması

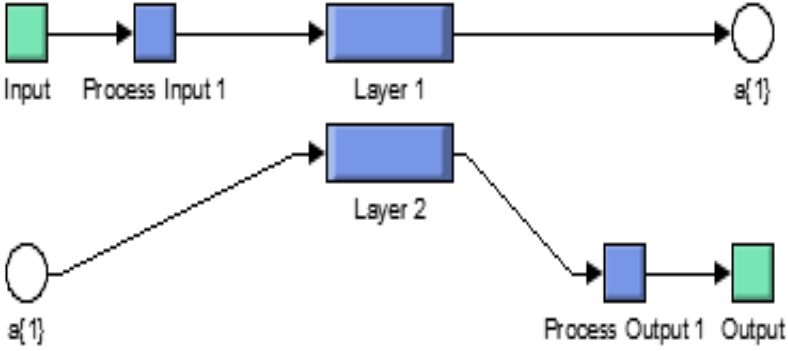
Logsig fonksiyonu ile 0 ile 1 deęerleri arasında elde edilen yapay sinir aęı simülasyon sonuçları ile 0 ve 1 deęerlerinden oluşan hedef verileri Őekil 8’de gösterilmiřtir. Őekil 8’de gösterilen karşılařtırma grafięinde; mavi renk ile gösterilen deęerler ıktı deęerlerini, turuncu ile gösterilen deęerler hedef deęerleri göstermektedir.

Őekil 4.9’da YSA Simulink modelinin yapısı gösterilmektedir.



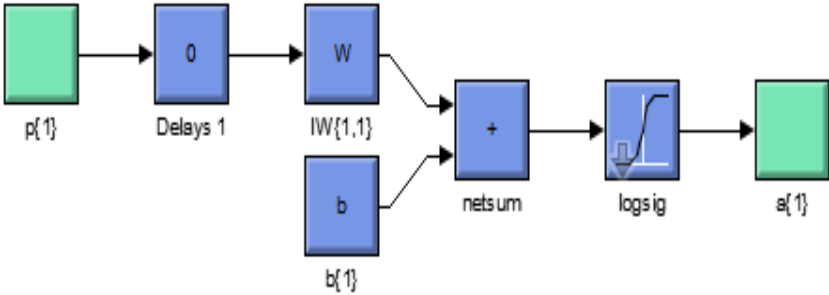
Őekil 4.9. YSA Simulink modeli

Őekil 4.10’da ise yapay sinir aęı eęitim modelinin alt sistemlerinin gösterildięi YSA mimarisi yer almaktadır.



Şekil 4.10. YSA mimarisi

Şekil 4.11’de, oluşturulan simulink modelinin birinci katmanında yapılan işlemler gösterilmektedir.

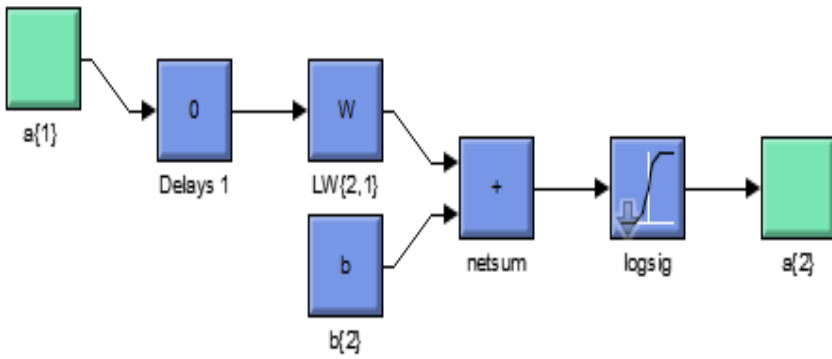


Şekil 4.11. Simulink modeli (1. katmanda yapılan işlemler)

Şekil 4.11’de gösterilen P değeri dış ortamdan alınan bilgilerdir. 1. katmandaki nöronlardan her biri, kendisine bağlı olan ağırlık değeriyle (w) çarpılır. Bir sonraki katmanda yer alan nöronların her birine denk

gelen eşik değerleri ($b\{1\}$) netsum fonksiyonuyla toplanır. Aktivasyon fonksiyonu logsig olarak belirlenmiştir. Logsig fonksiyonunda hesaplamalar yapılmakta ve nöronlarda hesaplanan değerler, $a\{1\}$ çıkış vektörüyle sonraki katmana aktarılmaktadır.

Şekil 4.12’de, oluşturulan simulink modelinin ikinci katmanında yapılan işlemler gösterilmektedir.



Şekil 4.12. Simulink modeli (2. katmanda yapılan işlemler)

1. katmanda nöronların logsig fonksiyonunda hesaplanan değeri ($a\{1\}$), sonraki katman ile arasında bulunan ağırlıklarla çarpılır ve 2. katmanda her bir nörona denk gelen eşik değerleri ile ($b\{2\}$) toplanır ve o nöronun girdi değeri oluşur. Logsig transfer fonksiyonuyla sonuç değerleri ($a\{2\}$) elde edilir.

4.3. Kullanıcı Arayüzü Oluşturma

Girilen ölçüm değerleri oluşturulan arayüz ile Excel dosyasına kaydedilecektir. Arayüz, kalite operatörlerinin kullanabileceği basit bir tasarımla oluşturulmuştur.

Grafiksel kullanıcı arayüzü oluşturmak için MATLAB çalışma ekranına GUIDE komutu yazılır ve açılan “Figure1” etiketli pencereden gerekli arayüz bileşenleri seçilir.

İlk aşamada arayüzün temel görünümünü oluşturmak için ihtiyaç duyulan edit text, static text, push button bileşenleri “Figure1” sayfasına taşınmıştır. İkinci aşamada modelin son görünümünü elde etmek için bileşenlerin ekran üzerindeki yerleri düzenlenir ve renk, boyut gibi özellikleri belirlenmiştir.

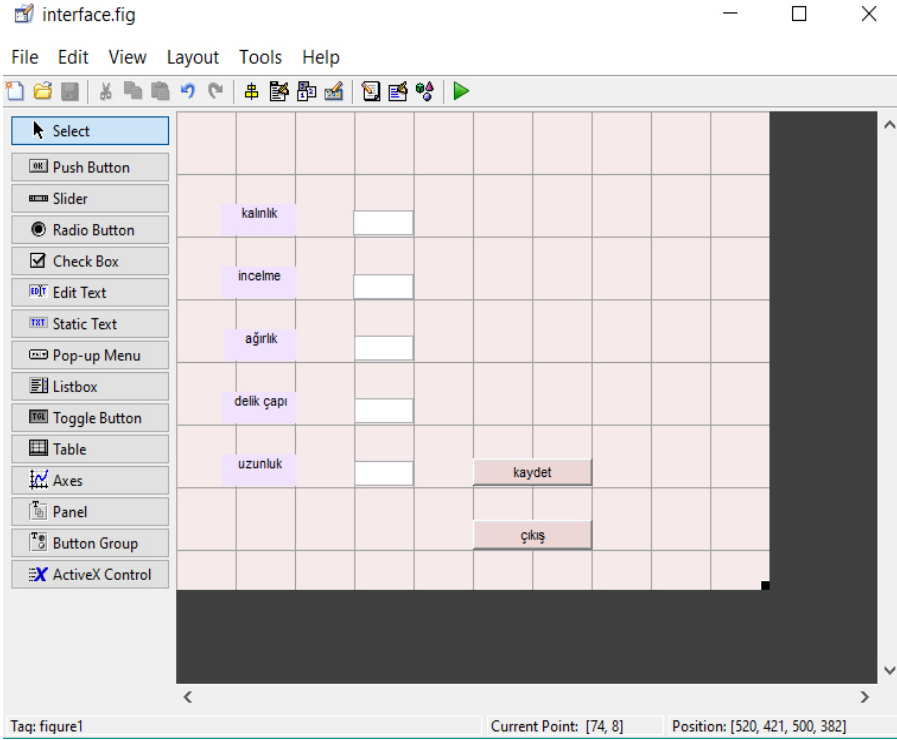
Yapılan kontroller sonucu ölçülen kalınlık, incelme miktarı, ağırlık, delik çapı ve uzunluk değerleri için “edit text” ler kullanılmıştır.

Her bir “edit text” e girilecek değer için başlığı “static text” bileşeni kullanılarak yazılmıştır.

Kaydet butonu, yeni kullanıcı girişlerinin bilgisayarda tanımlı Excel dosyasına kaydedilmesini sağlar.

Çıkış butonu ise veri girişi tamamlandıktan sonra ekranı kapatmak için kullanılır.

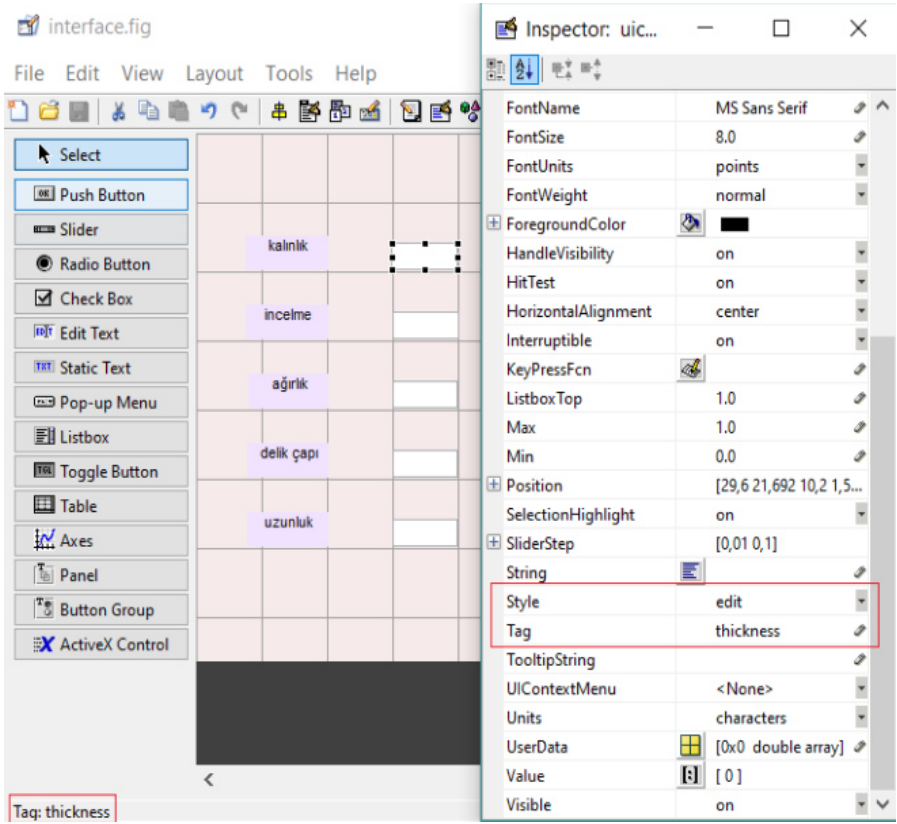
Şekil 4.13’de “figure 1” ekran görüntüsü yer almaktadır.



Şekil 4.13. Figure1 ekranı

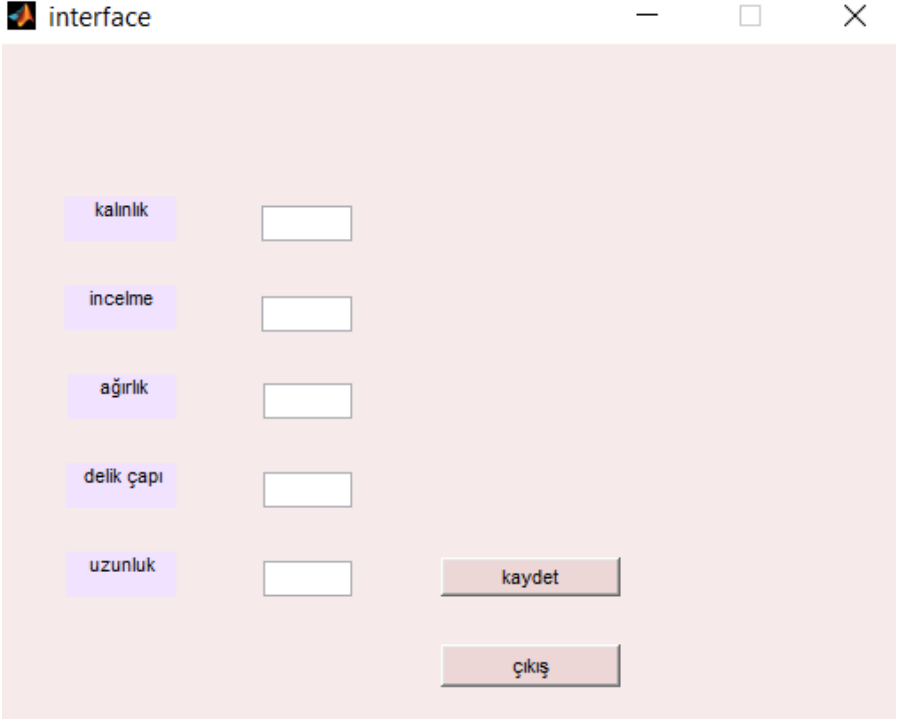
Eklenen bileşenlerin her birine bir görev tanımlanmıştır ve tanımlanan işlemleri yapılabilmesi için “callback” (geri çağırma) komutları ile kodlanması gerekmektedir.

Herhangi bir işlem gerçekleştirmesi beklenen birimler için; bileşenin üzerine çift tıkladığında açılan, özelliklerin belirlendiği pencereden özelliği ve etiketi belirlenir. Örneğin “kalınlık” değerinin girileceği “edit text” in kodlama aşamasında anlaşılır olması için “edittext1” olan etiketi kalınlık kelimesinin İngilizcesi olan “thickness” olarak değiştirilmiştir. Şekil 4.14’de “figure 1” için “edit text” ekranı gösterilmektedir.



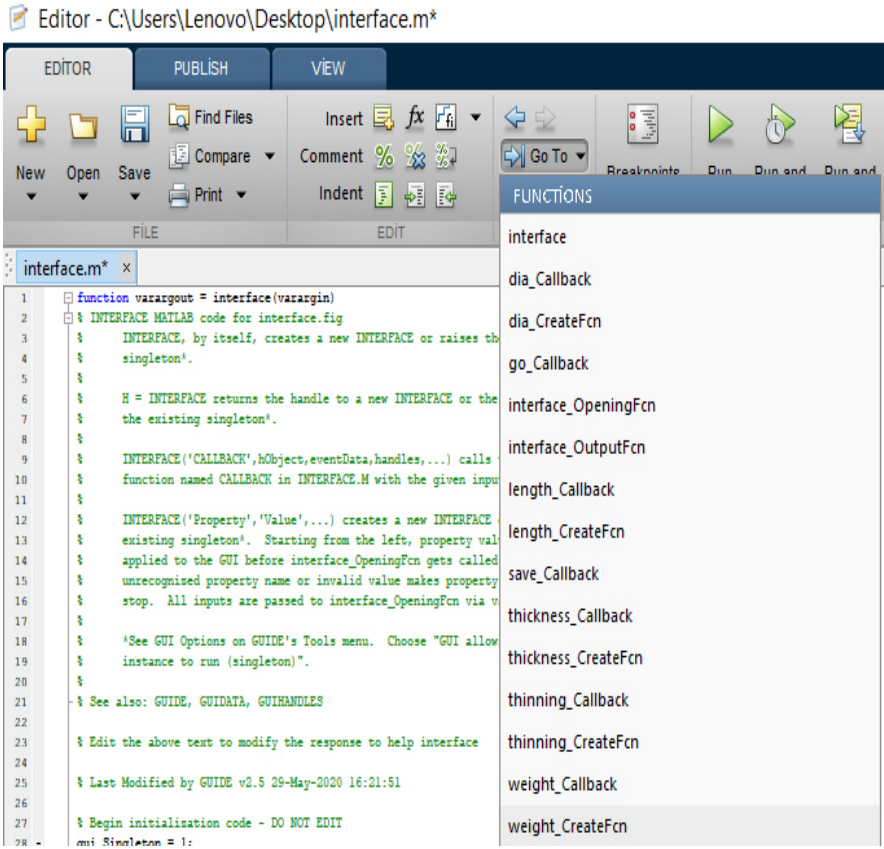
Şekil 4.14. Figure1 edit text

Arayüz tasarlandıktan sonra dosya kaydedilir. Oluşturulan arayüzün yer aldığı .fig uzantılı şekil dosyası ve Matlab kodlarının yazılacağı .m uzantılı iki dosya oluşur. Şekil 4.15’de Matlab Grafikselle Kullanıcı Arayüzü ekranı gösterilmektedir.



Şekil 4.15. Matlab Grafikselle Kullanıcı Arayüzü

Şekil 4.16’da ise, .m uzantılı interface (arayüz) dosyası gösterilmektedir.



Şekil 4.16. Interface.m dosyası

Şekil 4.16'daki dosyada her bileşen için tanımlanan etiketler ile “callback” komutları programlanır.

Çalışmanın amaçlarından birisi, beş adet ölçüm değerini (kalınlık, incelme, ağırlık, delik çapı, uzunluk) gösteren “edit text”lere girilen verilerin, kaydet butonuna basıldığında Excel dosyasına otomatik olarak kaydedilmesidir.

Bu sebeple “edit text” bileşelerinin veriyi “user_entry” adıyla sayısal değer olarak alması ve saklaması gerekmektedir.

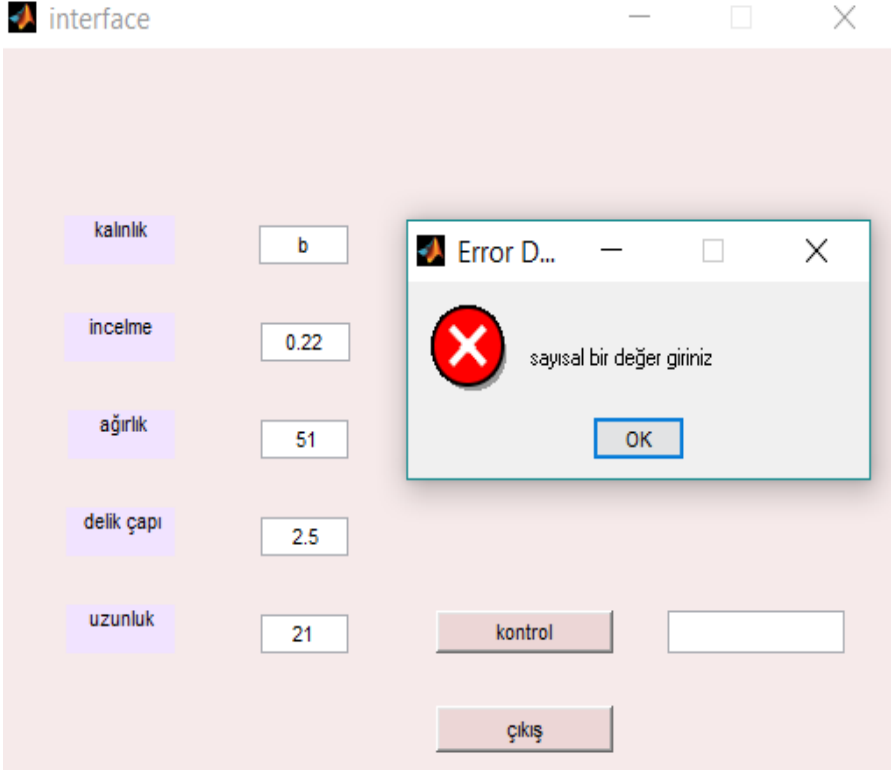
Şekil 4.17’de kalınlık (thickness) ölçüm değeri için callback yapılmasını sağlayan “edit text1” gösterilmektedir.

```
function thickness_Callback(hObject, eventdata, handles)
% hObject      handle to thickness (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of thickness as text
%        str2double(get(hObject,'String')) returns contents of thickness as a double
user_entry1=str2double(get(hObject,'string'));
if isnan(user_entry1)
    errordlg('sayısal bir değer giriniz')
    return
end
set(hObject,'value',user_entry1);
```

Şekil 4.17. Edit text1 (thickness) callback

Oluşturulan arayüze firma çalışanı tarafından, sayısal değerler dışında veri girildiğinde programın çalışmasına engel olunacağı için, sistem “errordlg” komutu kullanarak hata mesajıyla çalışanı uyarır. Şekil 4.18’de sayısal bir veri girilmediğinde sistem tarafından verilecek hata uyarısı için bir örnek gösterilmektedir.



Şekil 4.18. Hata mesajı

Şekil 4.19’da, oluşturulan arayüzde kayıt yapılmasını sağlayan “kaydet” butonu için yapılan “callback” gösterilmektedir.

```

% --- Executes on button press in save.
function save_Callback(hObject, eventdata, handles)
% hObject    handle to save (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

user_entry1=get(handles.thickness,'value');
user_entry2=get(handles.thinning,'value');
user_entry3=get(handles.weight,'value');
user_entry4=get(handles.dia,'value');
user_entry5= get(handles.length,'value');

a=[user_entry1 user_entry2 user_entry3 user_entry4 user_entry5];
disp(a);

fileName = 'sonuc.xls';

if exist(fileName,'file')
fileContentOLD = xlsread(fileName);
fileContentNEW = [fileContentOLD;a];
xlswrite(fileName,fileContentNEW);
else
xlswrite(fileName,a);
end

```

Şekil 4.19. Kaydet butonu callback

Şekil 4.19’da “callback” i gösterilen “Kaydet” butonu çalıştırıldığında “user_entry”lerde saklanan kullanıcı girişleri yeni veriler olarak kabul edilir.

“Get()” komutu ile kullanıcı girişleri her bir “edit text”ten alınmaktadır. Modelin yeni girdi verileri 1x5 boyutlu a matrisi haline getirilir. Oluşturulan a matrisi “disp(a)” komutu ile Matlab çalışma ekranında da görünür hale getirilir.

Her yeni kullanıcı girişini tanımlı Excel dosyasında yeni satıra yazdırmak için “xlswrite” komutu kullanılmaktadır. Ancak “xlswrite” komutu her kullanıldığında Excel'i yeniden başlatmayı ve kapatmayı içermektedir. Bu nedenle yeni verileri önceki satırın üzerine yazar. Veri aktarma işlemi “Sonuç” isimli Excel dosyası bilgisayarda mevcutsa başlar ve if şart döngüsüyle sağlanır. Veri eklemeyen önce “xlsread” komutu ile ilk satırdaki veriler okunur ve eski içerik olarak “fileContentOLD” şeklinde etiketlenir. Yeni girdiler (a matrisi) ve eski veriler “fileContentNEW” olarak etiketlenir. Sonuç dosyasına yazdırılır.

Matlab çalışma sayfasında “interface” ismiyle kaydedilen arayüz çağırılır ve ölçüm değerleri girilir ve kaydet butonuna basıldığında veriler Excel dosyasında kaydedilir. Şekil 4.20’de matlab komut ekranı gösterilmektedir.

```
Command Window
>> interface
    5.2000    22.0000    54.0000    3.0000    21.0000
    4.5000     0.2300    50.0000    2.7000    21.0000
    5.2000     0.1900    43.0000    2.4400    21.8000
    3.2400     0.1700    52.1000    2.3000    20.6000
    4.8000     0.2400    48.7000    3.1000    21.4000
    4.5000     0.2200    50.2000    2.7500    20.8900
```

Şekil 4.20. Matlab komut ekranı

Şekil 4.21’de ise excel dosyasına kaydedilen veriler gösterilmektedir.

	A	B	C	D	E	F	G
1	4,3	0,2	52	2,8	21		kabul
2	5,12	0,2	50	2,9	22		red
3	5	0,2	52,3	2,4	21,6		red
4	4,19	0,23	54	2,9	22		red
5	4,8	0,22	51,6	2,9	21		kabul
6	4,5	0,19	50,5	2,8	20,6		kabul
7	3,9	0,17	51,6	2,99	21,6		red
8	3,8	0,17	51,5	2,99	21,6		red
9	5,2	22	54	3	21		red
10	4,5	0,23	50	2,7	21		red
11	5,2	0,19	43	2,44	21,8		red
12	3,24	0,17	52,1	2,3	20,6		red
13	4,8	0,24	48,7	3,1	21,4		red
14	4,5	0,22	50,2	2,75	20,89		kabul

Şekil 4.20. Excel'e kaydedilen veriler

Ürünlerin, ölçüm değerlerine göre kabul veya reddedileceğine aşağıda gösterilen formülasyon ile karar verilmektedir.

=EĞER(VE(4<=A1;A1<=5; 0,18<=B1;B1<=0,22; 46<=C1;C1<=54; 2,4<=D1;D1<=3; 20,5<=E1; E1<=21,5);"kabul";"red")

BÖLÜM 5.

İSTATİSTİKSEL ANALİZ

Bu bölümde, eğitilen yapay sinir ağı modelinin çıktı değerlerini tahmin etmedeki güvenilirliğini hesaplamak üzere hipotez testi kurulacak ve analiz sonuçlarına göre modelin kullanılabilirliğine karar verilecektir. YSA çıktı verileri, lojistik regresyonla tahmin edilen çıktı değerleri ile karşılaştırılacaktır.

5.1. Hipotez Testi / Mann Whitney Testi

Yapay sinir ağı sonuçlarının güvenilirliğini test etmek için YSA test çıktıları ve istenen hedef değerleri arasında hipotez testi kurulur. Hedeflenen durum (yokluk hipotezi) iki sonuç arasında fark olmamasıdır. Yokluk hipotezine karşı gösterilen durum (alternatif hipotez) sonuçlar arasında fark olduğu, modelin güvenilir olmadığıdır.

H_0 = YSA sonuçları ile hedef değerleri arasında fark yoktur.

H_1 = YSA sonuçları ile hedef değerleri arasında fark vardır.

Yapay sinir ağı sonuçları ve hedef değerleri için aykırı değer ve normallik testi yapılarak kullanılacak istatistik testine karar verilecektir.

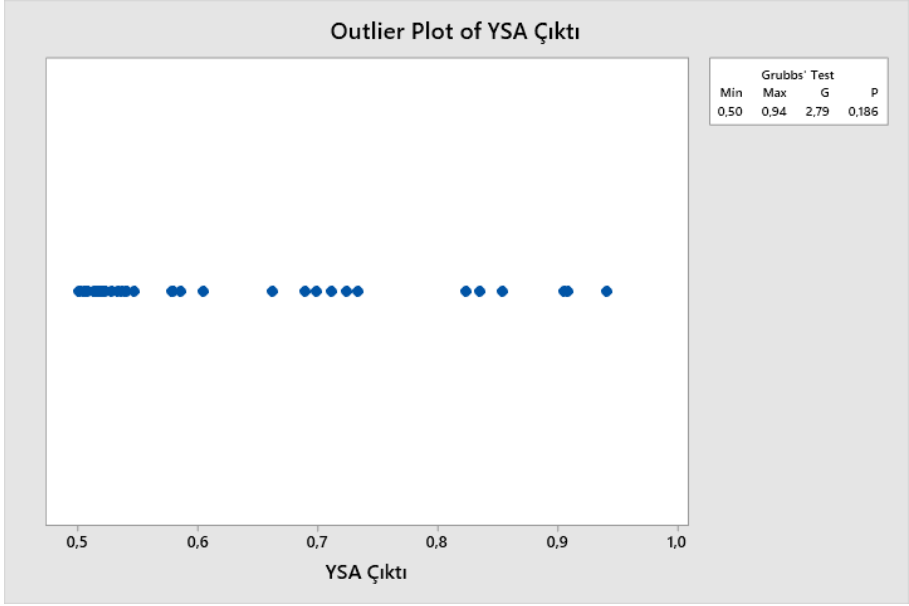
Aykırı değer testi için;

Yokluk hipotezi: Bütün veriler normal dağılmıştır.

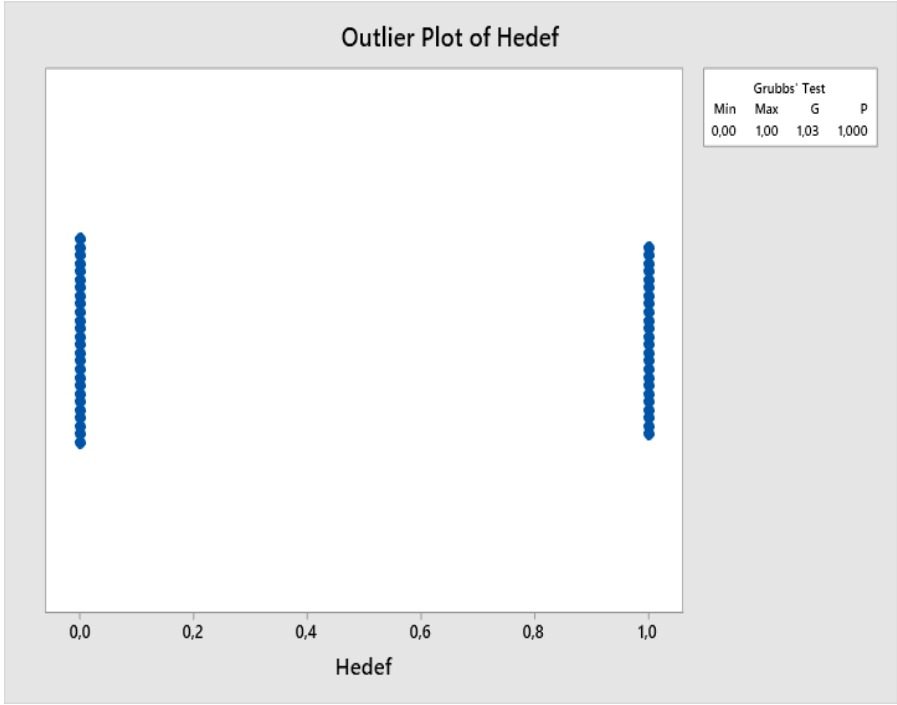
Alternatif hipotez: Aykırı değerler vardır.

Anlamlılık düzeyi: $\alpha = 0,05$

Şekil 5.1 ve şekil 5.2'deki grafiklerde görüldüğü gibi verilerde aykırı değer bulunmamaktadır.

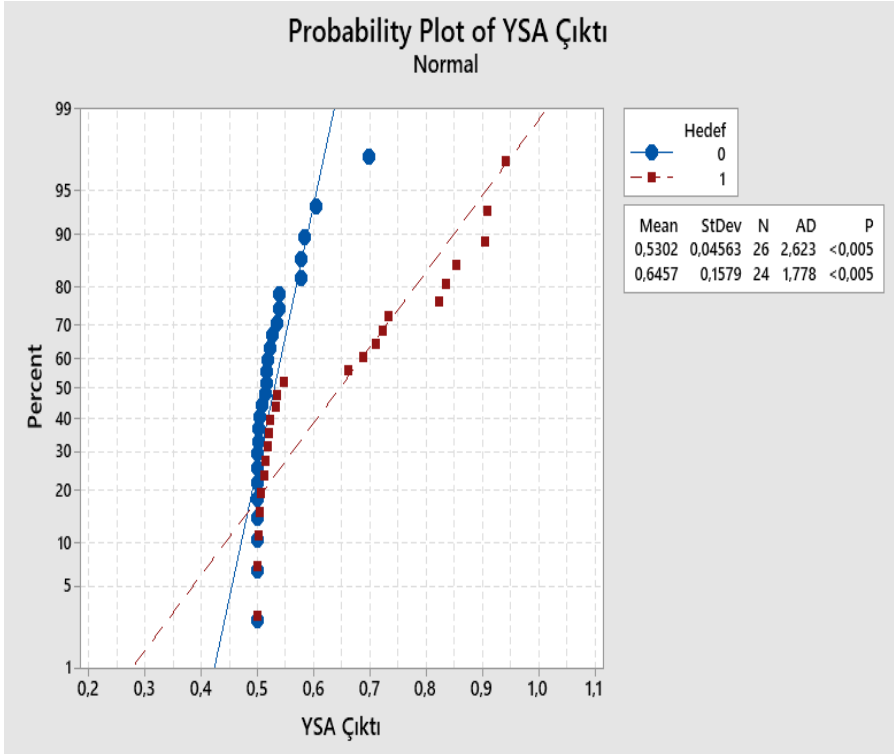


Şekil 5.1. Aykırı değer testi - YSA Çıktı



Şekil 5.2. Aykırı değer testi - Hedef

Şekil 5.3'de normallik testi sonuçları gösterilmektedir.



Şekil 5.3. Normallik testi

Şekil 5.3’de de görüldüğü üzere, yapay sinir ağı çıktılarının hedef değerlerine göre dağılımının normallik testi %95 güven aralığında yapılmıştır. P değerleri 0,05’ten küçük olduğundan veriler normal dağılmamıştır.

Korelasyon katsayıları ise şu şekildedir:

Correlations

YSA

Çıktı

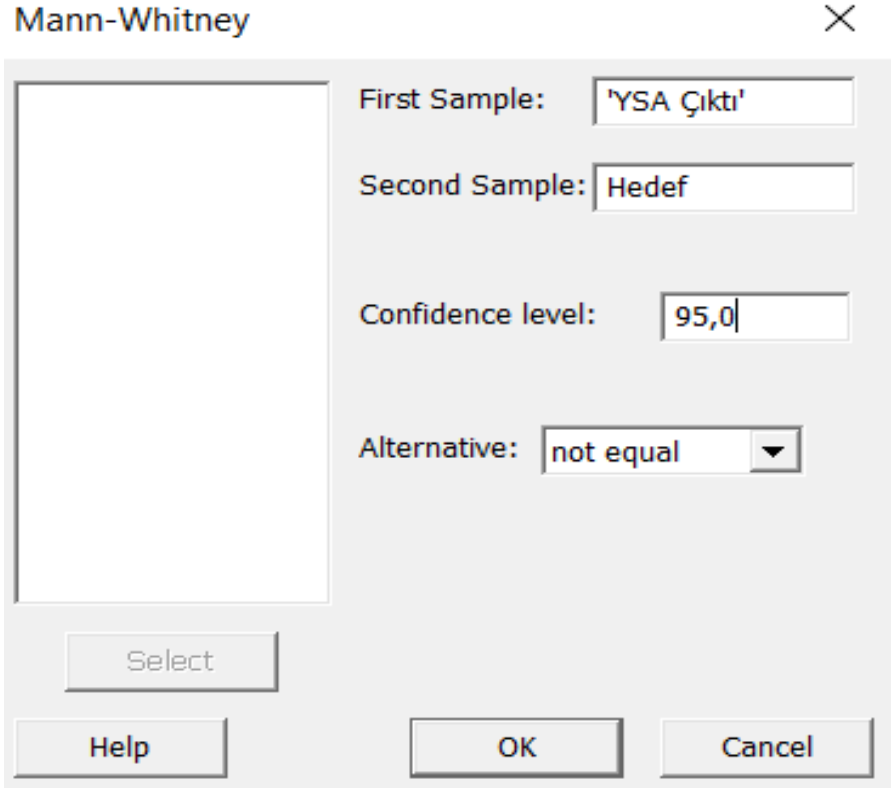
Hedef 0,405

İki sonuç arasındaki ilişki katsayısı 0,459'dur. Değişkenler arasında orta kuvvette korelasyon vardır. Verilerin normal dağılmadığı durumda kullanılan "Spearman Rank" korelasyon katsayısı tercih edilmiştir.

H0 ve H1 hipotezi, veriler normal dağılmadığından bağımsız iki örnek grubu için parametrik olmayan Mann-Whitney testi kullanılarak test edilmiştir. Medyan değerleri arasındaki fark değerlendirilecektir.

Minitab programında Stat >> Nonparametrics >> Mann-Whitney Test adımları takip edilir.

Şekil 5.4.'de Mann-Whitney testi gösterilmektedir.



Şekil 5.4. Mann-Whitney Testi

Metod;

μ_1 : median of YSA Çıktı

μ_2 : median of YSA Çıktı

Difference: $\eta_1 - \eta_2$

Yapılan istatistik analizlerin sonuçları tablo 5.1’de gösterilmektedir.

Tablo 5.1. İstatistik sonuçları

Descriptive Statistics		
Sample	N	Median
YSA Çıktı	50	0,521273
Hedef	50	0,000000

Hesaplanan 0,500009 fark değeri, tahminin doğruluğunun güven aralığı ile açıklandığı kısımda ise H0 hipotezi kabul edilir.

Tahmin edilen fark değerleri tablo 5.2’de gösterilmektedir.

Tablo 5.2. Tahmin edilen fark

Estimation for Difference		
Difference	CI for Difference	Achieved Confidence
0,500009	(-0,302275; 0,501238)	95,02%

Mann-Whitney testinin sonuçları tablo 5.3’de gösterilmektedir.

Tablo 5.3. Mann-Whitney testi sonuçları

Test		
Null hypothesis	Ho: $\eta_1 - \eta_2 = 0$	
Alternative hypothesis	H1: $\eta_1 - \eta_2 \neq 0$	
Method	W-Value	P-Value
Not adjusted for ties	2575,00	0,733
Adjusted for ties	2575,00	0,729

P değeri 0,05'ten büyük olduğundan ve hesaplanan fark değeri yokluk hipotezinin kabul edildiği alanda olduğu için H0 hipotezi kabul edilir. Yani yapay sinir ağı çıktıları ve istenilen hedef değerleri arasında farklılık yoktur. Ölçüm sonuçlarının tahmini eğitilen yapay sinir ağıyla gerçekleştirilebilir.

5.2. Lojistik Regresyon

Lojistik regresyon ile tahminlenen deęerlerin, hedef deęerlerine olan uyumunu hesaplamak için YSA eęitiminde de kullanılan 1050 adet ölçüm deęeri ve hedef verileri Şekil 5.5'deki gibi Minitab çalışma sayfasına yüklenmiştir.

Kurulacak modelde;

Baęımlı deęişken: Kabul/Red durumu

Baęımsız deęişkenler: Ölçüm deęerleri (kalınlık, incelme, aęırlık, delik çapı, uzunluk)

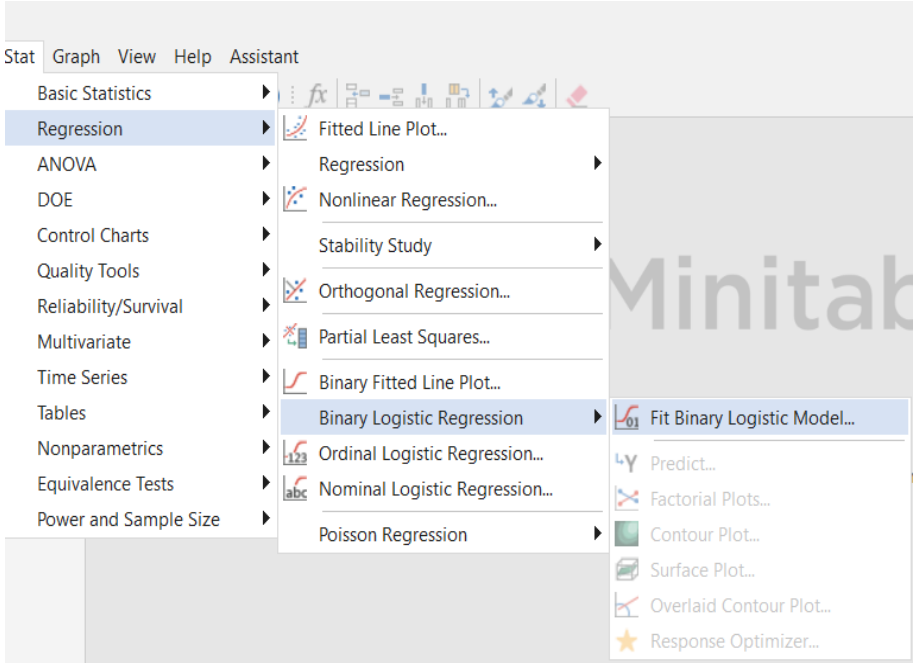
↓	C1	C2	C3	C4	C5	C6
	kalınlık	incelme	ağırlık	delik çapı	uzunluk	hedef
1	4,320	0,21	51,70	2,75	21,24	1
2	5,000	0,18	48,83	2,91	20,64	1
3	4,582	0,21	46,02	2,85	21,46	1
4	4,014	0,21	52,68	2,60	21,00	1
5	4,700	0,19	51,16	2,86	20,72	1
6	3,990	0,22	53,05	2,64	20,84	0
7	3,980	0,18	53,28	2,40	21,14	0
8	3,950	0,20	53,09	2,59	21,22	0
9	3,960	0,20	49,69	2,68	20,64	0
10	3,920	0,22	51,13	2,86	21,13	0
11	3,890	0,19	50,99	2,46	20,86	0
12	3,850	0,18	53,25	2,81	21,30	0
13	3,860	0,19	49,96	2,99	20,80	0
14	3,810	0,20	50,39	2,94	21,23	0
15	3,800	0,19	46,23	2,55	21,33	0
16	3,790	0,20	52,90	3,00	20,81	0
17	3,740	0,19	49,72	2,63	20,73	0
18	3,720	0,21	46,49	2,84	21,49	0
19	3,700	0,19	49,30	2,71	21,45	0
20	3,690	0,18	49,18	2,45	20,84	0

Şekil 5.5. Verilerin yüklenmesi

Lojistik regresyon modelini kurmak için;

Stat >> Regression >> Binary Logistic Regression >> Fit Binary Logistic Model adımları izlenir.

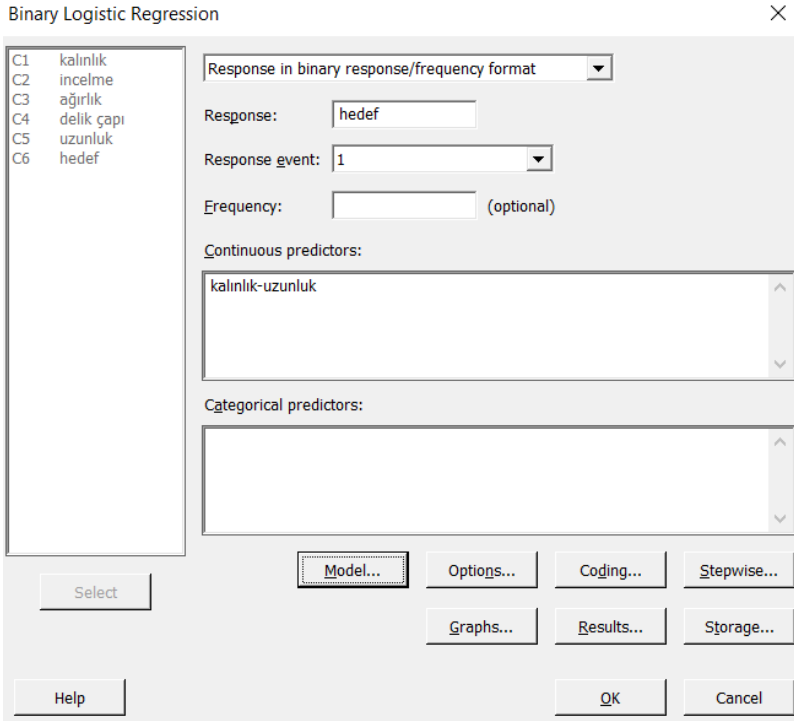
Lojistik Regresyon için Minitab programında modelin kurulması şekil 5.6'da gösterilmektedir.



Şekil 5.6. Lojistik regresyon modelinin kurulması

Lojistik regresyon modeli kurulurken referans değer 1 (kabul) olarak alınmıştır. Logit fonksiyonuyla % 95 güven aralığında bağımlı ve bağımsız değerler girilerek model oluşturulmuştur.

Minitab programındaki Lojistik Regresyon şekil 5.7’de gösterilmektedir.



Şekil 5.7. Lojistik regresyon

Tablo 5.4’de görüldüğü gibi ölçüm değerleri girilen 1050 ürünün 491’i kabul, 559’ u reddedilmiştir.

Tablo 5.4. Ürün durumları

Response Information			
Variable	Value	Count	
hedef	1	491	(Event)
	0	559	
	Total	1050	

Tablo 5.5’de lojistik regresyon modelinin özeti gösterilmektedir.

Tablo 5.5. Lojistik regresyon model özeti

Model Summary				
Deviance R-Sq	Deviance R-Sq(adj)	AIC	AICc	BIC
4,97%	4,63%	1391,05	1391,13	1420,79

Tablo 5.5’den de görüleceği üzere, Lojistik regresyon analizinde % 95 güven aralığında modelin, hedef değerlerini tahmin edebilmedeki başarısı %4,97’ dir.

Regresyon Eşitliği:

$Y' = 15,55 + 0,653 \text{ kalınlık} + 6,76 \text{ incelme} + 0,0118 \text{ ağırlık} - 1,071 \text{ delik çapı} - 0,835 \text{ uzunluk}$

Tahmin edilen sonuçların model ile uyumluluğu şekil 5.8’de gösterilmektedir.

↓	C1	C2	C3	C4	C5	C6 <input checked="" type="checkbox"/>	C7
	kalınlık	incelme	ağırlık	delik çapı	uzunluk	hedef	FITS
1	4,320	0,21	51,70	2,75	21,24	1	0,437297
2	5,000	0,18	48,83	2,91	20,64	1	0,567551
3	4,582	0,21	46,02	2,85	21,46	1	0,381806
4	4,014	0,21	52,68	2,60	21,00	1	0,472894
5	4,700	0,19	51,16	2,86	20,72	1	0,538103
6	3,990	0,22	53,05	2,64	20,84	0	0,504285
7	3,980	0,18	53,28	2,40	21,14	0	0,448340
8	3,950	0,20	53,09	2,59	21,22	0	0,403283
9	3,960	0,20	49,69	2,68	20,64	0	0,497434
10	3,920	0,22	51,13	2,86	21,13	0	0,373924
11	3,890	0,19	50,99	2,46	20,86	0	0,478347
12	3,850	0,18	53,25	2,81	21,30	0	0,291887
13	3,860	0,19	49,96	2,99	20,80	0	0,348684
14	3,810	0,20	50,39	2,94	21,23	0	0,282641
15	3,800	0,19	46,23	2,55	21,33	0	0,337665
16	3,790	0,20	52,90	3,00	20,81	0	0,356264
17	3,740	0,19	49,72	2,63	20,73	0	0,434830
18	3,720	0,21	46,49	2,84	21,49	0	0,265287
19	3,700	0,19	49,30	2,71	21,45	0	0,268174
20	3,690	0,18	49,18	2,45	20,84	0	0,431043

Şekil 5.8. Tahmin edilen sonuçların model ile uyumluluğu (fits)

Uyumluluk testinin sonuçları tablo 5.6'da gösterilmektedir.

Tablo 5.6. Uyumluluk testi sonuçları

Goodness-of-Fit Tests			
Test	DF	Chi-Square	P-Value
Deviance	1044	1379,05	0,000
Pearson	1044	1042,89	0,504
Hosmer-Lemeshow	8	105,66	0,000

Tablo 5.6’da gösterilen uyum iyiliği sonuçlarında Pearson testine göre p değeri 0,05 anlamlılık seviyesinden daha yüksektir, bu da modelin verilere uymadığı sonucuna varmak için yeterli kanıt olmadığını gösterir.

Oluşturulan regresyon eşitliği üzerinden tahmin yapmak için yapay sinir ağını test etmek için kullanılan veriler kullanılır. Daha sonra sinir ağı sonuçları, lojistik regresyon tahmin değerleri ve hedef değerleri karşılaştırılır.

Şekil 5.9’da lojistik regresyonla tahmin değerleri gösterilmektedir.

Predict

Response: hedef

Enter individual values

kalınlık	incelme	ağırlık	'delik çapı'	uzunluk	
4,38	0,21	48,8	3	21,16	^
4,07	0,18	45,55	2,63	20,44	
4,11	0,2	45,6	2,5	21,41	
4,73	0,19	46,01	2,59	21,04	
4,98	0,18	49,38	2,78	21,17	
4,95	0,21	47,61	2,42	21,16	v

Select Options... Results... Storage... View Model...

Help OK Cancel

Şekil 5.9. Lojistik regresyonla tahmin

Tablo 5.7’de lojistik regresyon ve YSA sonuçları karşılaştırılması gösterilmektedir.

Tablo 5.7. Sonuç karşılaştırma tablosu

Lojistik regresyon ile tahmin	Ysa sonuçları	Hedef
0,382	0,521	1
0,518	0,500	0
0,392	0,500	0
0,529	0,532	1
0,485	0,518	1
0,622	0,854	1
0,406	0,500	0
0,338	0,500	0
0,490	0,835	1
0,436	0,905	1
0,460	0,905	0
0,679	0,501	0
0,510	0,584	1
0,606	0,501	0
0,536	0,507	0
0,547	0,535	1
0,257	0,823	0
0,562	0,500	1
0,391	0,688	1
0,434	0,546	0
0,593	0,501	0
0,452	0,723	1
0,426	0,500	0

0,666	0,500	0
0,510	0,519	0
0,577	0,513	1
0,647	0,711	1
0,621	0,577	0
0,667	0,505	1
0,683	0,515	1

Hedef deęerlerini tahmin etmedeki yakınlıęa ve deęerlerin tutarlılıęına bakıldıęında yapay sinir aęı sonuęlarının daha gvenilir sonuęlar verdięi grlmektedir.

Eşik deęeri 0,5 olarak kabul edildięinde, yani 0,5 ve altı sonuęlar rnn reddedileceęini, 0,5'ten byk deęerler kabul edileceęini gsterdięinde yapay sinir aęı modelinin doęru tahmin etme oranı lojistik regresyonla edilen tahminlere gre az farkla daha yksektir. Kurulan modellerde kullanılan zm yntemleri, lm deęerleri deęiřtięinde tahmin performansları da deęiřecektir.

BÖLÜM 6.

SONUÇ

Matlab yazılımı ücretli bir yazılım olduğundan ölçüm kontrol verilerinin girileceği tüm bilgisarlarda bulunması işletme için maliyetli olabilir. Ancak oluşturulan arayüz programı internet ortamına aktarılarak kullanılabilir.

Çalışma bir yazılım programı üzerinden yapıldığından farklı sürümlerin çalışmalarında farklılık olacağından yapay sinir ağı modelinin her bilgisayara entegre edilmesi, hatasız kullanılması mümkün değildir.

Uygulama öncesinde işletmede veriler üretim alanında operatörler tarafından kağıtlara yazılıyordu. Bu durum üretim esnasında işlerin aksamasına, verilerin işlenmesi sırasında karmaşıklığa, zaman zaman verilerin kaybolmasına sebep oluyordu. Amaç, ölçüm değerleri girdi verisi kabul edilip sisteme işlendiğinde sonucu tahmin eden sistemin test edilmesiydi. Yapay sinir ağları yönteminin uygulanabilirliği göz önünde bulundurularak tahmin değerlerinin güvenilirliği test edildi ve modelde başarılı sonuçlar elde edildi. Eğitilen yapay sinir ağı geliştirilerek, belirlenen eşik değerine göre ürünün kabul veya red durumuna karar vermek üzere yeni girdi verileriyle kullanılacaktır.

İşletmede kullanılan Matlab GUI yazılımı yardımıyla arayüz oluşturuldu. Arayüz üzerinden girilen ölçüm değerlerinin Excel

dosyasına otomatik olarak kaydedilmesi amaçlandı. Kullanıcı arayüzü oluşturma çalışmalarının daha da geliştirilerek ürün özelliklerini eğitilen yapay sinir ağına göre tahmin edip kabul/red durumuna program tarafından karar verilecek hale getirilmesi mümkündür.

KAYNAKLAR

1. Houcque D., 2005, Introduction To MATLAB For Engineering Students,
<https://www.mccormick.northwestern.edu/documents/students/undergraduate/introduction-to-matlab.pdf> , (Elde edilme tarihi, Kasım 2020)
2. Bellala G., 2020, A Quick tutorial on MATLAB,
<https://web.eecs.umich.edu/~aey/eecs451/matlab.pdf> , (Elde Edilme tarihi, Kasım 2020)
3. Kubat C., 2012, MATLAB 7.11 – Yapay Zeka ve Mühendislik Uygulamaları, Beşiz Yayınları, Sakarya, ISBN: 978-605-86639-2-3
4. Hahn B., Valentine D., 2019, Essential MATLAB for Engineers and Scientists, Academic Press, 7th edition, ISBN-10 : 0081029977 ISBN-13 : 978-0081029978
5. Attaway S., 2018, MATLAB: A Practical Introduction to Programming and Problem Solving, Butterworth-Heinemann, 5th edition, ISBN-10 : 0128154799 ISBN-13 : 978-0128154793
6. Kattan P., 2008, MATLAB For Beginners: A Gentle Approach, CreateSpace Independent Publishing Platform, ISBN-10 : 1438203098 ISBN-13 : 978-1438203096
7. Gilat A., 2016, MATLAB: An Introduction with Applications, Wiley, 6th edition, ISBN-10 : 111938513X ISBN-13 : 978-1119385134

8. Lee H-H., 2019, Programming and Engineering Computing with MATLAB 2019, SDC Publications, ISBN-10: 1630572977 ISBN-13: 978-1630572976
9. Chapra S., 2017, Applied Numerical Methods with MATLAB for Engineers and Scientists, McGraw-Hill Education, 4th edition, ISBN-10 : 0073397962 ISBN-13 : 978-0073397962
10. Paluszek M., Thornas S., 2020, Practical MATLAB Deep Learning: A Project-Based Approach, Apress, 1st edition, ISBN-10 : 1484251237 ISBN-13 : 978-1484251232
11. Corke P., 2017, Robotics, Vision and Control: Fundamental Algorithms in MATLAB, Springer, 2nd edition, ISBN-10: 3319544128 ISBN-13: 978-3319544120
12. Green D., 2020, Stars and Space with Matlab apps (With Companion Media Pack), WSPC, ISBN-10: 9811216355 ISBN-13: 978-9811216350
13. Ciaburro G., 2017, MATLAB for Machine Learning: Practical examples of regression, clustering and neural networks, Packt Publishing, ISBN-10: 1788398432 ISBN-13: 978-1788398435
14. Pratap R., 2016, Getting Started with MATLAB: A Quick Introduction for Scientists and Engineers, Oxford University Press, 7th edition, ISBN-10: 0190602066 ISBN-13: 978-0190602062
15. Gonzales R.C., Woods R.E., 2020, Digital Image Processing Using MATLAB, Gatesmark, 3rd edition, ISBN-10: 0982085419 ISBN-13: 978-0982085417

16. Chapman S.J., 2019, MATLAB Programming for Engineers, Cengage Learning, 6th edition, ISBN-10: 0357030397 ISBN-13: 978-0357030394
17. Ling S., Langtangen H.P., 2016, Programming for Computations – MATLAB/Octave: A Gentle Introduction to Numerical Simulation with MATLAB/Octave, Springer, 1st edition, ASIN : B078YHJHL2
18. Graupe D., 2019, Principles of Artificial Neural Networks: Basic Designs to Deep Learning (Advanced Circuits and Systems), WSPC World Scientific, 4th Edition, ISBN-10 : 9811201226, ISBN-13 : 978-9811201226
19. Thakur A., Sarkar D., Bharti V., Kannan U., 2021, Development of in-core fuel management tool for AHWR using artificial neural networks, Annals of Nuclear Energy, 150, 107869
20. Effendy N., Ab Aziz S.H., Kamari H.M., Mohd Zaid M.H., Shabdin M.K., Khiri M.Z.A., Wahab S.A.A., 2020, Artificial neural network prediction on ultrasonic performance of bismuth-tellurite glass composition, Journal of Material Research and Technology, 9(6), page 14082-14092
21. Kumar Y., Singh L., Sharanagat V.S., Tarafdar A., 2020, Artificial Neural Network (ANNs) and mathematical modelling of hydration of green chickpea, Information Processing in Agriculture, Article in Press, <https://doi.org/10.1016/j.inpa.2020.04.001>

22. Wu D., Wang G.G., 2021, Causal artificial neural network and its applications in engineering design, *Engineering Applications of Artificial Intelligence*, 97
23. Kashani H.R.K., Azhari S., Nayebaghayee H., Salimi S., Mohammadi H.R., 2020, Prediction value of preoperative findings on meningioma grading using artificial neural network, *Clinical Neurology and Neurosurgery*, 196, 105947
24. Gopal L.C., Govindarajan M., Kavipriya M.R., Mahboob S., Al-Ghanim K.A., Virik P., Ahmed Z., Al-Mulahim N., Senthilkumaran V., Shankar V., Optimization strategies for improved biogas production by recycling of waste through response surface methodology and artificial neural network: Sustainable Energy perspective research, *Journal of King Saud University – Science*, Article in Press, doi: <https://doi.org/10.1016/j.jksus.2020.101241>
25. Gajewski J., Valis D., 2021, Verification of the technical equipment degradation method using a hybrid reinforcement learning trees-artificial neural network system, *Tribology International*, 153, 106618
26. Pauletto P.S., Dotto G.L., Salau N.P.G., 2020, Optimal artificial neural network design for simultaneous modeling of multicomponent adsorption, *Journal of Molecular Liquids*, 320, 114418
27. Delgado J.M.P.Q., Silva F.A.N., Azevedo A.C., Silva D.F., Campello R.L.B., Santos R.L., 2020, Artificial neural networks to assess the useful life of reinforced concrete elements

- deteriorated by accelerated chloride tests, *Journal of Building Engineering*, 31, 101445
28. Lee J., Lee Y.C., Kim J.T., 2021, Migration from the traditional to the smart factory in the die-casting industry: Novel process data acquisition and fault detection based on artificial neural network, *Journal of Materials Processing Tech.*, 290, 116972
 29. Pled F., Desceliers C., Zhang T., 2021, A robust solution of a statistical inverse problem in multiscale computational mechanics using an artificial neural network, *Computer Methods in Applied Mechanics and Engineering*, 373, 113540
 30. Arslankaya S., 2020, Estimating the effects of heat treatment on aluminum alloy with artificial neural networks, *Emerging Materials Research* 9(2): 540–549,
<https://doi.org/10.1680/jemmr.20.00059>
 31. Di Benedetto R.M., Botelho E.C., Janotti A., Ancelotti Junior A.C., Gomes G.F., 2020, Development of an artificial neural network for predicting energy absorption capability of thermoplastic commingled composites, *Composite Structure*, 113131, Article in press
 32. Baiocco G., Almonti D., Guarino S., Tagliaferri F., Tagliaferri V., Ucciardello N., 2020, Image-based system and artificial neural network to automate a quality control system for cherries pitting process, *Procedia CIRP*, 88, page 527-532 (13th CIRP Conference on Intelligent Computation in Manufacturing Engineering, CIRP ICME'19)

33. Gao J., Zagadailov P., Merchant A.M., 2020, The use of artificial neural network to predict surgical outcomes after inguinal hernia repair, *Journal of Surgical Research*, Article in Press
34. Vieira L.W., Marques A.D., Schneider P.S., Neto A.J.S., Viana F.A.C., Abdel-jawad M., Hunt J.D., Siluk J.C.M., 2021, Methodology for ranking controllable parameters to enhance operation of a steam generator with a combined artificial neural network and design of experiments approach, *Energy and AI*, 3, 100040
35. Ngo Q.H., Schmitt S., Ellerich M., Schmitt R.H., 2020, Artificial intelligence based decision model for a quality oriented production ramp-up, *Procedia CIRP*, 88, page 521-526 (13th CIRP Conference on Intelligent Computation in Manufacturing Engineering, CIRP ICME'19)
36. Gan H., Liu G., Shi C., Tang R., Xiong Y., Liu Y., Liu H., 2021, Comparison of three artificial neural networks for predict the electrodeposition of nano-silver film, *Materials Today Communications*, 26, 101950
37. Khojasteh D.N., Goudarzi G., Taghizadeh-Mehrjardi R., Asumadu-Sakyi A.B., Fehrest-Sani M., 2021, Long-term effects of outdoor air pollution on mortality and morbidity-prediction using nonlinear autoregressive and artificial neural networks models, *Atmospheric Pollution Research*, Article in Press

38. Zhang T., Li X., Zhao Q., Rao Y., 2019, Control of a novel synthetical index for the local indoor air quality by the artificial neural network and genetic algorithm, *Sustainable Cities and Society*, 51, 101714
39. Ghoniem M., Awad T., Mokhiamar O., 2020, Control of a new low-cost semi-active vehicle suspension system using artificial neural networks, *Alexandria Engineering Journal*, 59, page 4013-4025
40. KorkusuzPolat T., Arslankaya S., 2010, The cost forecasting application in an enterprise with artificial neural networks, *Proceedings of the World Congress on Engineering*, Vol III, WCE 2010, June 30-July 2, London, UK
41. Sharma A., Kushvaha V., 2020, Predictive modelling of fracture behaviour in silica-filled polymer composite subjected to impact with varying loading rates using artificial neural network”, *Engineering Fracture Mechanics*, 239, 107328
42. Gupta J.N.D., Majumder A., Laha D., 2019, Flowshop scheduling with artificial neural networks, *Journal of the Operational Research Society*,
DOI: 10.1080/01605682.2019.1621220
43. Martin O., Lopez M., Martin F., 2007, Artificial neural networks for quality control by ultrasonic testing in resistance spot welding, *Journal of Materials Processing Technology*, 183, page 226-233
44. Shin Y., Smith R., Hwang S., 2020, Development of model predictive control system using an artificial neural network: A

- case study with a distillation column, *Journal of Cleaner Production*, 277, 124124
45. Fayed M., Elhadary M., Abderrahmane H.A., Zakher B.N., 2019, The ability of forecasting flapping frequency of flexible filament by artificial neural network, *Alexandria Engineering Journal*, 58, page 1367-1374
46. Stoffel M., Gulakala R., Bamer F., Markert B., 2020, Artificial neural networks in structural Dynamics: A new modular radial basis function approach vs. convolutional and feedforward topologies, *Computer Methods in Applied Mechanics and Engineering*, 364, 112989
47. Banga K.S., Kotwaliwale N., Mohapatra D., Giri S.K., Bargale P.C., 2020, Assessment of bruchids density through bioacoustic detection and artificial neural network (ANN) in bulk stored chickpea and green gram, *Journal of Stored Products Research*, 88, 101667
48. Kumar Y., Singh L., SinghSharanagat V., Tarafdar A., 2020, Artificial Neural Network (ANNs) and mathematical modelling of hydration of green chichpea, *Information Processing in Agriculture*, Article in Press
49. Wu D., Wang G.G., 2021, Causal artificial neural network and its applications in engineering design, *Engineering Applications of Artificial Intelligence*, 97, 104089
50. Chen H., Lu F., He B., 2020, Topographic property of backpropagation artificial neural network: From human

- functional connectivity network to artificial neural network, *Neurocomputing*, 418, page 200-210
51. Lella E., Vessio G., Ensembling complex network ‘perspectives’ for mild cognitive impairment detection with artificial neural networks, *Pattern Recognition Letters*, 136, page 168-174
 52. You G., Zhao H., Gao D., Wang M., Ren X., Wang Y., 2020, Predictive models of tensile strength and disintegration time for simulated Chinese herbal medicine extracts compound tablets based on artificial neural network, *Journal of Drug Delivery Science and Technology*, 60, 102025
 53. Iseli E., Schiffmann J., 2021, Prediction of the reaction forces of spiral-groove gas journal bearings by artificial neural network regression models, *Journal of Computational Intelligence*, 48, 101256,
<https://doi.org/10.1016/j.jocs.2020.101256>
 54. Xu Y., Fenerci A., Oiseth O., Moan T., 2020, Efficient prediction of wind and wave induced long-term extreme load effects of floating suspension bridges using artificial neural networks and support vector machines, *Ocean Engineering*, 217, 107888
 55. Li C., 2020, Biodiversity assessment based on artificial intelligence and neural network algorithm, *Microprocessors and Microsystems*, 79, 103321

EKLER

EK 1: Yapay Sinir Ağı Oluşturma Kodları

```
% Solve an Input-Output Fitting problem with a Neural Network
```

```
% This script assumes these variables are defined:
```

```
%
```

```
% input - input data.
```

```
% target - target data.
```

```
inputs = input;
```

```
targets = target;
```

```
% Create a Fitting Network
```

```
hiddenLayerSize = 11;
```

```
net = fitnet(hiddenLayerSize);
```

```
% Choose Input and Output Pre/Post-Processing Functions
```

```
% For a list of all processing functions type: help nnprocess
```

```
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
```

```
net.outputs{2}.processFcns = {'removeconstantrows','mapminmax'};
```

```
% Setup Division of Data for Training, Validation, Testing
```

```
% For a list of all data division functions type: help nndivide
```

```

net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'sample'; % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% For help on training function 'trainlm' type: help trainlm
% For a list of all training functions type: help nntrain
net.trainFcn = 'trainlm'; % Levenberg-Marquardt

% Choose a Performance Function
% For a list of all performance functions type: help nnperformance
net.performFcn = 'mse'; % Mean squared error

% Choose Plot Functions
% For a list of all plot functions type: help nnplot
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
    'plotregression', 'plotfit'};

% Train the Network
[net,tr] = train(net,inputs,targets);

% Test the Network
outputs = net(inputs);
errors = gsubtract(targets,outputs);

```

```
performance = perform(net,targets,outputs)
```

```
% Recalculate Training, Validation and Test Performance
```

```
trainTargets = targets .* tr.trainMask{1};
```

```
valTargets = targets .* tr.valMask{1};
```

```
testTargets = targets .* tr.testMask{1};
```

```
trainPerformance = perform(net,trainTargets,outputs)
```

```
valPerformance = perform(net,valTargets,outputs)
```

```
testPerformance = perform(net,testTargets,outputs)
```

```
% View the Network
```

```
view(net)
```

EK 2: MATLAB GUI Oluşturma Callback Kodları

```
function varargout = interface(varargin)
% INTERFACE MATLAB code for interface.fig
%   INTERFACE, by itself, creates a new INTERFACE or raises the
existing
%   singleton*.

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @interface_OpeningFcn, ...
                  'gui_OutputFcn', @interface_OutputFcn, ...
                  'gui_LayoutFcn', [], ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

```

% --- Executes just before interface is made visible.
function interface_OpeningFcn(hObject, eventdata, handles, varargin)

% Choose default command line output for interface
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes interface wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = interface_OutputFcn(hObject, eventdata, handles)

% Get default command line output from handles structure
varargout{1} = handles.output;

function thickness_Callback(hObject, eventdata, handles)

user_entry1=str2double(get(hObject,'string'));
if isnan(user_entry1)
    errordlg('sayısal bir değer giriniz')
return

```

end

```
set(hObject,'value',user_entry1);
```

% --- Executes during object creation, after setting all properties.

```
function thickness_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
    set(hObject,'BackgroundColor','white');
```

```
end
```

```
function thinning_Callback(hObject, eventdata, handles)
```

```
user_entry2=str2double(get(hObject,'string'));
```

```
if isnan(user_entry2)
```

```
    errordlg('sayısal bir değer giriniz')
```

```
    return
```

```
end
```

```
set(hObject,'value',user_entry2);
```

% --- Executes during object creation, after setting all properties.

```
function thinning_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function weight_Callback(hObject, eventdata, handles)
```

```
user_entry3=str2double(get(hObject,'string'));
```

```
if isnan(user_entry3)  
    errordlg('sayısal bir değer giriniz')  
    return  
end
```

```
set(hObject,'value',user_entry3);
```

```
% --- Executes during object creation, after setting all properties.
```

```
function weight_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function dia_Callback(hObject, eventdata, handles)
```

```
user_entry4=str2double(get(hObject,'string'));
```

```
if isnan(user_entry4)
```

```
    errordlg('sayısal bir değer giriniz')
```

```
    return
```

```
end
```

```
set(hObject,'value',user_entry4);
```

```
% --- Executes during object creation, after setting all properties.
```

```
function dia_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
    set(hObject,'BackgroundColor','white');
```

```
end
```

```
function length_Callback(hObject, eventdata, handles)
```

```
user_entry5=str2double(get(hObject,'string'));
```

```
if isnan(user_entry5)
```

```
    errordlg('sayısal bir değer giriniz')
```

```
    return
```

```
end
```

```
set(hObject,'value',user_entry5);
```

```
% --- Executes during object creation, after setting all properties.
```

```
function length_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
    set(hObject,'BackgroundColor','white');
```

```
end
```

```
% --- Executes on button press in save.
```

```
function save_Callback(hObject, eventdata, handles)
```

```
user_entry1=get(handles.thickness,'value');
```

```
user_entry2=get(handles.thinning,'value');
```

```
user_entry3=get(handles.weight,'value');
```

```
user_entry4=get(handles.dia,'value');
```

```
user_entry5= get(handles.length,'value');
```

```
a=[user_entry1 user_entry2 user_entry3 user_entry4 user_entry5];
```

```
disp(a);
```

```
fileName = 'sonuc.xls';
```

```
if exist(fileName,'file')
```

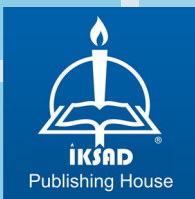
```
fileContentOLD = xlsread(fileName);  
fileContentNEW = [fileContentOLD;a];  
xlswrite(fileName,fileContentNEW);  
else  
xlswrite(fileName,a);  
end
```

% --- Executes on button press in go.

```
function go_Callback(hObject, eventdata, handles)
```

```
display teşekkürler
```

```
closereq();
```

ISBN: 978-625-7687-73-7