

**T.C.  
SÜLEYMAN DEMİREL ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**MOTORLU ARAÇLARIN PERFORMANSLARININ MAKİNE  
ÖĞRENMESİ TEKNİKLERİYLE ANALİZİ VE TAHMİNİ**

**İbrahim Arda ÇANKAYA**

**Danışman  
Dr. Öğr. Üyesi Arif KOYUN**

**DOKTORA TEZİ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
ISPARTA- 2021**



© 2021 [İbrahim Arda ÇANKAYA]

## İÇİNDEKİLER

|                                                                        | Sayfa |
|------------------------------------------------------------------------|-------|
| İÇİNDEKİLER.....                                                       | i     |
| ÖZET .....                                                             | iii   |
| ABSTRACT .....                                                         | iv    |
| TEŞEKKÜR.....                                                          | v     |
| ŞEKİLLER DİZİNİ .....                                                  | vi    |
| ÇİZELGELER DİZİNİ .....                                                | viii  |
| SİMGELER VE KISALTMALAR DİZİNİ .....                                   | ix    |
| 1. GİRİŞ.....                                                          | 1     |
| 2. KAYNAK ÖZETLERİ .....                                               | 5     |
| 3. MATERYAL VE YÖNTEM .....                                            | 13    |
| 3.1. Dinamometre .....                                                 | 13    |
| 3.1.1. Motor Dinamometresi .....                                       | 14    |
| 3.1.2. Şasi Dinamometresi.....                                         | 15    |
| 3.2. Makine Öğrenmesi.....                                             | 17    |
| 3.2.1. Pekiştirmeli öğrenme.....                                       | 19    |
| 3.2.2. Denetimsiz öğrenme .....                                        | 20    |
| 3.2.3. Yarı denetimli öğrenme .....                                    | 22    |
| 3.2.4. Denetimli öğrenme .....                                         | 23    |
| 3.2.4.1. Destek vektör regresyonu.....                                 | 26    |
| 3.2.4.2. Çoklu doğrusal regresyon .....                                | 29    |
| 3.2.4.3. Rastgele orman regresyonu .....                               | 30    |
| 3.2.5. Yapay sinir ağları.....                                         | 32    |
| 3.2.5.1. Katmanlar.....                                                | 36    |
| 3.2.5.2. Aktivasyon fonksiyonları .....                                | 37    |
| 3.2.5.3. Yapay sinir ağları sınıflandırmaları .....                    | 38    |
| 3.2.5.4. Eğitim algoritmaları .....                                    | 41    |
| 3.2.6. Adaptif ağ tabanlı bulanık mantık çıkarım sistemi .....         | 41    |
| 4. ARAŞTIRMA BULGULARI.....                                            | 47    |
| 4.1. Dinamometre Denetleme Sisteminin Yapılması .....                  | 48    |
| 4.2. Dinamometre Yazılım Sisteminin Yapılması.....                     | 51    |
| 4.3. Kaporta Hasar Yazılımının Yapılması.....                          | 57    |
| 4.4. Verilerin Hazırlanması.....                                       | 60    |
| 4.5. Destek Vektör Regresyonu Bulguları.....                           | 63    |
| 4.6. Çoklu Doğrusal Regresyon Bulguları .....                          | 64    |
| 4.7. Rastgele Orman Regresyonu Bulguları .....                         | 65    |
| 4.8. Yapay Sinir Ağları Bulguları .....                                | 67    |
| 4.9. Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi Bulguları ..... | 69    |
| 5. TARTIŞMA VE SONUÇLAR.....                                           | 77    |
| KAYNAKLAR .....                                                        | 80    |
| ÖZGEÇMİŞ.....                                                          | 86    |

## ÖZET

### Doktora Tezi

## MOTORLU ARAÇLARIN PERFORMANSLARININ MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE ANALİZİ VE TAHMİNİ

İbrahim Arda ÇANKAYA

Süleyman Demirel Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Arif KOYUN

Motorlu araçların kullanımı giderek artmaktadır. Bu araçlar sınıflandırılırken performansları, güçleri baz alınmaktadır. Araçların performans, güç ölçümleri yapılırken birden fazla seçenek sunulmaktadır. Fakat en doğru sonucu veren test şekli şasi dinamometresi cihazı ile ölçümün gerçekleştirilmesidir. Şasi dinamometresi testleri kayıp gücü de hesapladığı için en yüksek doğruluk oranına sahip olmaktadır. Şasi dinamometreleri araçların çekiş gücünün olduğu tekerlerin silindirler üzerindeki yüksek hızda hareketiyle performans ölçümleri yapmaktadır. Fakat bu testler kısa sürede yüksek hızda gerçekleştirildiği için bazı araçlarda motor hasarlarına sebebiyet vermektedir. Bu tez çalışmasında ülkemizde kullanılmakta olan şasi dinamometresi cihazlarının ölçümlerini ve haberleşmelerini sağlayabilecek bir sistem geliştirilecektir. Geliştirilen bu sistem ile birlikte araçların hasar durumları da göz önüne alınarak makine öğrenmesi teknikleri kullanılarak araç performans tahminleri yapılmıştır. Çalışmada adaptif ağ tabanlı bulanık mantık çıkarım sistemi, çoklu doğrusal regresyon, rastgele orman regresyonu, destek vektör regresyonu ve yapay sinir ağları algoritmaları kullanılarak ortaya çıkan tahminler değerlendirilmiştir. Elde edilen sonuçlara göre performansı en başarılı makine öğrenmesi tekniği %96,6 tahmin oranıyla adaptif ağ tabanlı bulanık mantık çıkarım sistemi olmuştur. Çalışmada makine öğrenmesi tekniklerinin başarı oranlarına göre şasi dinamometresi cihazı kullanılmadan, sadece araç ve kullanıcı ile ilgili bilgileri analiz ederek aracın performans tahminin yapılması amaçlanmıştır.

**Anahtar Kelimeler:** Şasi dinamometresi, Makine öğrenmesi, Yazılım geliştirme.

**2021, 87 sayfa**

## **ABSTRACT**

### **Ph.D. Thesis**

## **ANALYSIS AND ESTIMATION OF MOTOR VEHICLES' PERFORMANCES USING MACHINE LEARNING TECHNIQUES**

**İbrahim Arda ÇANKAYA**

**Süleyman Demirel University  
Graduate School of Natural and Applied Sciences  
Department of Computer Engineering**

**Supervisor: Asst. Prof. Dr. Arif KOYUN**

Usages of motor vehicles is increased day by day. These vehicles are classified based on their performance and power. There are several options when these vehicles performing performance and power measures. But the most accurate result is that the test is performed with the chassis dynamometer. Chassis dynamometer tests have the highest accuracy because of calculating loss power also. Chassis dynamometers measure the performance of the wheels where the traction power is on the cylinders at high speed. But because these tests are carried out at high speeds and in a short time, it causes engine damage in some vehicles. Chassis dynamometers are also widely used in the measurement of the performance of used vehicles market. In this thesis, design measurement and communication of chassis dynamometer are developed which is using our country. Along with the developed chassis dynamometer communication and measurement system, the pattern information given by the users will be used learning purposes. By taking all the information of these tests into consideration, a system will be made to predict the performance of the vehicles using machine learning techniques. Adaptive neuro-fuzzy inference system, multiple linear regression, random forest regression, support vector regression and artificial neural network algorithms were modeled and used for prediction. The most successful machine learning algorithm in the modeling of estimation of motor vehicles' performance was determined as Adaptive neuro-fuzzy interference system with 96,6 % accurate prediction. It is aimed to predict the performance of the vehicle by analyzing only vehicle and user information without using the chassis dynamometer according to the success rates of the machine learning techniques.

**Keywords:** Chasses dynamometer, Machine learning, Software development.

**2021, 87 pages**

## TEŞEKKÜR

Bu araştırma için beni yönlendiren, karşılaştığım zorlukları bilgi ve tecrübesi ile aşmamda yardımcı olan değerli danışman hocam Dr. Öğr. Üyesi Arif KOYUN'a ve yine tezimde her türlü bilgi ve tecrübesini benden eksik etmeyen Dr. Öğr. Üyesi Asım Sinan YÜKSEL'e ve Öğr. Gör. Muzaffer TATLI'ya teşekkürlerimi sunarım.

Araştırmanın yürütülmesinde maddi ve manevi yardımlarını gördüğüm ve tez izleme komitesinde bulunan Prof. Dr. Tuncay YİĞİT ve Prof. Dr. İbrahim ÜÇGÜL'e teşekkür ederim.

Tezimin her aşamasında beni yalnız bırakmayan aileme sonsuz sevgi ve saygılarımı sunarım.

İbrahim Arda ÇANKAYA  
İSPARTA, 2021

## ŞEKİLLER DİZİNİ

|                                                                               | Sayfa |
|-------------------------------------------------------------------------------|-------|
| Şekil 1.1. Türkiye 2020 yılı araç cinslerine göre sayıları .....              | 1     |
| Şekil 3.1. Motor dinamometresi şeması.....                                    | 15    |
| Şekil 3.2. Motor dinamometresi.....                                           | 16    |
| Şekil 3.3. Şasi dinamometresi .....                                           | 17    |
| Şekil 3.4. Makine öğrenmesi şeması.....                                       | 19    |
| Şekil 3.5. Makine öğrenmesi yöntemleri .....                                  | 20    |
| Şekil 3.6. Pekiştirmeli öğrenme yönteminin genel yapısı.....                  | 21    |
| Şekil 3.7. Denetimsiz öğrenme yönteminin genel yapısı.....                    | 23    |
| Şekil 3.8. Yarı denetimli öğrenme yöntemiyle ilgili açıklayıcı bir örnek..... | 24    |
| Şekil 3.9. Denetimli öğrenme aşamaları .....                                  | 25    |
| Şekil 3.10. Destek vektör regresyonu vektörleri.....                          | 28    |
| Şekil 3.11. Rastgele orman yönteminin genel yapısı .....                      | 32    |
| Şekil 3.12. Sinir ağlarının genel yapısı.....                                 | 34    |
| Şekil 3.13. Yapay sinir ağlarının temel yapısı.....                           | 35    |
| Şekil 3.14. Yapay sinir ağlarının katmanları .....                            | 37    |
| Şekil 3.15. İleri beslemeli örnek.....                                        | 39    |
| Şekil 3.16. Geri beslemeli örnek.....                                         | 40    |
| Şekil 3.17. Adaptif ağ tabanlı bulanık mantık çıkarım sistemi.....            | 43    |
| Şekil 4.1. Geliştirilen sistemin mimarisi .....                               | 47    |
| Şekil 4.2. Kare dalga sinyali üreten denetleyici sistem.....                  | 48    |
| Şekil 4.3. Basit hareketli ortalama algoritması .....                         | 49    |
| Şekil 4.4. AD9833 sinyal jeneratörü modülü .....                              | 50    |
| Şekil 4.5. Kare dalga üretim algoritması.....                                 | 50    |
| Şekil 4.6. Dinamometre cihazından sürekli gelen bilgiler .....                | 51    |
| Şekil 4.7. Dinamometre yazılımının ana sayfası .....                          | 52    |
| Şekil 4.8. Dinamometre yazılımı kullanım süreci akış diyagramı.....           | 53    |
| Şekil 4.9. Dinamometre yazılımı kayıt formu .....                             | 54    |
| Şekil 4.10. Dişli oranı eşitleme formu.....                                   | 55    |
| Şekil 4.11. Dinamometre yazılımı ayarlar formu .....                          | 56    |
| Şekil 4.12. Boya kalınlığı ölçüm cihazı.....                                  | 57    |
| Şekil 4.13. Hasar uygulaması ekran görüntüsü.....                             | 58    |
| Şekil 4.14. Hasar uygulaması ekran görüntüsü 2 .....                          | 59    |
| Şekil 4.15. Hasar uygulaması sonuç raporu .....                               | 60    |
| Şekil 4.16. SVR doğruluk oranı .....                                          | 63    |
| Şekil 4.17. SVR tahmin-gerçek veri karşılaştırılması.....                     | 64    |
| Şekil 4.18. MLR doğruluk oranı.....                                           | 64    |
| Şekil 4.19. MLR tahmin-gerçek veri karşılaştırması.....                       | 65    |
| Şekil 4.20. RF doğruluk oranı.....                                            | 66    |
| Şekil 4.21. RF tahmin-gerçek veri karşılaştırılması .....                     | 67    |
| Şekil 4.22. YSA doğruluk oranı .....                                          | 68    |
| Şekil 4.23. YSA tahmin-gerçek veri karşılaştırması .....                      | 69    |
| Şekil 4.24. ANFIS GUI.....                                                    | 70    |
| Şekil 4.25. Verilerin ANFIS'e aktarılması.....                                | 71    |
| Şekil 4.26. Gauss2mf üyelik fonksiyonu grafiği .....                          | 72    |
| Şekil 4.27. Modelin öğrenme eğrisi.....                                       | 72    |
| Şekil 4.28. ANFIS ağ yapısı .....                                             | 73    |

|                                                                        |    |
|------------------------------------------------------------------------|----|
| Şekil 4.29. ANFIS test verisi doğruluk oranı.....                      | 74 |
| Şekil 4.30. ANFIS eğitim verisi doğruluk oranı .....                   | 74 |
| Şekil 4.31. ANFIS gerçek-test verisi tahminleri karşılaştırması .....  | 75 |
| Şekil 4.32. ANFIS gerçek-eğitim verisi tahminleri karşılaştırması..... | 76 |





## ÇİZELGELER DİZİNİ

|                                                                         | Sayfa |
|-------------------------------------------------------------------------|-------|
| Çizelge 3.1. Yapay sinir ağları toplama fonksiyonları.....              | 36    |
| Çizelge 3.2. Yaygın kullanılan aktivasyon fonksiyonları .....           | 38    |
| Çizelge 4.1. Elde edilen veri türleri ve çeşitliliği.....               | 61    |
| Çizelge 4.2. Tahmin modellerinde kullanılacak verilerin bir kısmı ..... | 61    |
| Çizelge 4.3. Tahmin algoritmaları $R^2$ sonuçları .....                 | 76    |



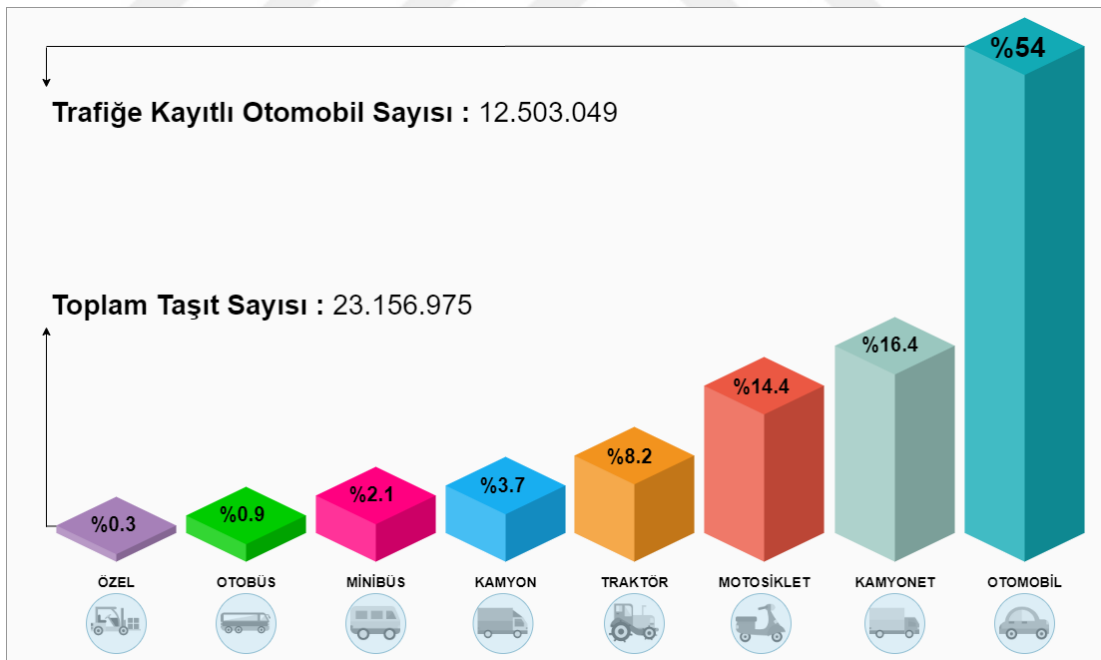
## SİMGELER VE KISALTMALAR DİZİNİ

|       |                                                   |
|-------|---------------------------------------------------|
| ANFIS | Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi |
| ANN   | Adaptive Neural Network                           |
| ÇLR   | Çoklu Lineer Regresyon                            |
| DVR   | Destek Vektör Regresyonu                          |
| HP    | Horse Power                                       |
| MLR   | Multi Linear Regression                           |
| PS    | Pferd Starke                                      |
| RF    | Random Forest                                     |
| RO    | Rasgele Orman                                     |
| RPM   | Revolution Per Minute                             |
| SVR   | Support Vector Regression                         |
| YSA   | Yapay Sinir Ağları                                |



## 1. GİRİŞ

Motorlu araçlar, bütün dünya üzerinde insan ve yük taşıma işlemlerinde kullanılmak için imal edilen, hayatı kolaylaştıran, kullanımı hiçbir zaman bitmeyecek bir eşya bir hizmet bir ürün olarak değerlendirilir. Yapılan araştırmalara göre dünya üzerindeki motorlu araçların sayısı 1,3 milyarı geçmiştir. TÜİK 2020 verilerine göre ülkemizde ise 23 milyondan fazla araç bulunmaktadır. Bu araçların cinslerine göre sayıları aşağıdaki grafikte gösterilmektedir. Dünya genelinde ortalama 1000 kişiye düşen araçların sayısı 145 tir. Türkiye’de ise sayı 150’nin üstündedir. Avrupa birliği ülkelerinde ise bu sayı 520’nin üstündedir (TÜİK, 2020). Motorlu araçlar hem fiziksel hem de tekniksel açılarından birbirinden ayrılmaktadır. Bu farklılıkların oluşmasında önemli bir paya sahip olan şeyler ise motor gücü ve verimdir. Şekil 1.1’de Ağustos 2020 yılı itibariyle Türkiye’deki araçların cinslerine göre sayıları gösterilmektedir.



Şekil 1.1. 2020 Ağustos ayı itibariyle araç cinslerine göre sayıları (TÜİK, 2020)

Motorlu taşıtlar laboratuvar ortamlarında test edilerek performans ölçümlerine ulaşabilmektedir. Motorlu taşıtlara yapılmış olan bu performans testlerinin güvenilir olabilmesi gerekli tüm parametrelerin dikkatlice ölçülmesi ve verilerin

eksiksiz bir şekilde kayıt altına alınmasıyla gerçekleştirilebilir. Güvenilirliđi sağlayabilmek adına birçok donanım içeren motor test sistemleri gerekmektedir. Geliştirilen bu sistemlerle elde edilmek istenen performans değeri hede flenen değeri göre doğruluđunu ölçmektir. Gücün birimi genelde beygir gücü olarak kullanılmaktadır. Fakat araç kataloglarında HP (Horse Power), PS (pferd starke) ve kW olarak da gösterilmektedir. HP ve PS hesaplamasında ufak farklılıklar olsa da birbirlerine çok yakındır. Bu tip farkları önlemek adına, elektriksel güç birimi olan kW olarak da gösterilmektedir. Bu üç değeri birbirlerine dönüşümü Denklem 1.1 ve Denklem 1.2’de gösterilmektedir.

$$PS = KW * 0.736 \quad (1.1)$$

$$HP = KW * 0.746 \quad (1.2)$$

Araçların motorları iki farklı güç içermektedir. Birincisi indike güçtür. İndike güç; silindirlere giren yakıtın yanması sonucu ortaya çıkan ısının enerjisinin mekanik enerjiye dönüşümü ile elde edilmektedir. En düşük yakıt tüketimi sağlayabilmek için yanma verimi aracın bütün kullanım şartlarında en yüksek tutulması gerekmektedir (Çelik ve Balcı, 2002). İkinci güç ise faydalı güç olarak adlandırılmaktadır. Fren güç olarak da adlandırılabilen bu güç motora ait volandan ya da kasnaktan ölçülen değeri ve motora ait gerçek güçtür. Ölçüm yerinden dolayı ölçülen güçte motordaki sürtünme ve diğeri kayıp güçleri yoktur, bundan dolayı bu hesaplanan güç gerçek güçtür.

Fren ya da faydalı güç dinamometre cihazı ile ölçülmektedir (Batmaz, 1998). Dinamometreler güç ölçüm cihazlarına verilen genel isimdir. Dönen herhangi bir makinenin gücünü ölçmek için kullanılmaktadır. Dinamometreler çoğunlukla otomobil motorlarının beygir gücü hesaplamalarında kullanılmaktadır.

Bilgisayarların ilk üretim anından beri en çok merak edilen şeylerden biri de insanların öğrenme becerisinin bilgisayarlar tarafında da yapıp yapılamayacağı konusu olmuştur. Makine öğrenmesi bilgisayarların öğrenme konusunu incelemektedir. Makine öğrenmesinden bahsedebilmek için bilgisayardaki

herhangi bir sistemin geçmiş çalışmalarına göre başarısını artırabilmesi gerekmektedir (Kahveci, 2011).

Yapay zekâ bilgisayarların akıllı, daha başarılı olması için uğraşan bir alandır. Öğrenme ihtiyacı zeki olan bir davranışın sonucudur. Dolayısıyla günümüzde çoğu araştırmacı öğrenme işlemi olmadan zekâ kavramından bahsetmenin yanlış olduğunu düşünmektedir. Bunlardan yola çıkarak Makine öğrenmesinin yapay zekâ uygulamaları kıyaslandığında hem önemli hem de hızlı gelişen alt bir disiplini olduğunu söyleyebiliriz. (Shavlik ve Dietterich, 1990; Michie vd. 1997) Makine öğrenmesindeki algoritmalar uzun süredir otomotiv sektöründe kullanılmaktadır. Sektör içerisinde verilerin toplanması ve analizleri belirli algoritmalar ile yapılarak yıllar içerisinde otomotiv içeren bütün sektörlerde birçok alana etki edebilecek şekilde değişimler ve gelişimler sağlamaktadır. Araç satış işlemlerinden, araç içerisinde arızalanan parça tespitlerine, müşterilerin yaşam standartlarına bakarak ihtiyacı olacak araçların belirlenmesinde, yakıt tüketim analizlerinin yapılarak öneri sistemlerine, sürüş dinamiklerinin incelenerek araç yıpranma süreçlerinin belirlenmesine kadar birçok farklı düzeyde bilgiler makine öğrenmesi algoritmaları kullanılarak analiz edilmektedir.

Dünya genelinde makine öğrenmesi alanında yapılan çalışmalar genel olarak incelendiğinde, makine öğrenmesi algoritmaları üzerinde belirgin bir sonuç ortaya çıkmıştır; Öğrenme algoritmalarının hiçbirinin bir diğerinden üstünlüğü yoktur, öğrenme algoritmalarının deneysel bir şekilde karşılaştırılması sonucunda en uygun algoritma bulunmaktadır (Wolpert ve Macready, 2005; Hoste, 2005).

Motorlu araçlar için performans testleri yapılırken ortaya bazı problemler çıkabilmektedir. Bu problemlerin en önemlilerinden birisi de aracın test sırasında zorlanmasından dolayı motorun hasar görme ya da kullanılamayacak hale gelme ihtimalinin olmasıdır. Bu tezin amacı piyasadaki güçlü, belirli bir yere gelmiş dinamometre cihazı üreten firmaların cihazların altyapısına eşdeğer kalitede bir şasi dinamometre cihazının haberleşme ve iletişim tasarımı ve

prototipin geliştirilmesiyle birlikte gerekli detaylı testleri yapmak ve bu test sonuçlarına ek olarak aracın genel bilgileri ve hasar durumları da göz önüne alınarak aracın dinamometre testi yapmadan beygir gücünün tahminini makine öğrenmesi teknikleri kullanarak hesaplamaktır. Çalışmada hem şase dinamometresi donanımıyla haberleşecek yazılımlar hem de ortaya çıkan sonuçlar destek vektör regresyonu, çoklu doğrusal regresyon, rastgele orman regresyonu, yapay sinir ağları ve adaptif ağ tabanlı bulanık mantık çıkarım sistemi gibi farklı makine öğrenmesi algoritmalarıyla birlikte analiz edilerek en başarılı sonuç veren algoritma bulunacaktır.

Tezin birinci bölümünde dinamometre ve makine Öğrenmesi kavramlarından kısaca bahsedilmiştir. İkinci bölümde şase dinamometreleri ve makine öğrenmesiyle regresyon analizleri üzerine yapılan çalışmalardan bahsedilmiştir. Üçüncü bölümde materyal ve yöntem anlatılmıştır. Dördüncü bölümde çalışma esnasında karşılaşılan araştırma bulgularına değinilmiştir. Beşinci bölümde sonuçlar ve öneriler sunulmuştur.

## 2. KAYNAK ÖZETLERİ

Batmaz (1998), yaptığı çalışmada şasi dinamometresi, yakıt metre ve egzoz analiz cihazlarının bilgisayar kullanılarak otomasyona çevrilmesini amaçlamıştır. Çalışmasında bu cihazlar üzerinden bilgisayar ortamına elektronik sinyaller ve sinyaller üzerinden anlamlandırılan veriler aktarılmıştır. Batmaz aktarılan verilerin bilgisayar ortamında değerlendirilerek mantıklı sonuçlar üretmeyi de sağlamıştır. Çalışma tamamlandığında bu tip motorlu araç testlerinde kullanılması gerekli olan yardımcı cihaz ihtiyacı da azalmıştır. Cihazlar üzerinden aktarılan elektronik sinyaller ve anlamlandırılmış veriler sayesinde verilerin doğrudan aktarılması sonuçlarda hata yapma ihtimalini de azaltmıştır. Kullanılan bilgisayarlardaki donanımların da etkisiyle cihazlardan gelen bilgilerin analiz edilmesi kolaylaşmıştır.

Hocaoğlu ve Kurban (2005), yaptıkları çalışmada güneşlenme sürelerinin tahmin edilebileceği bir çalışma yapmışlardır. Yapılan çalışmada güneşlenme süresi bilgileri Eskişehir bölgesinden alınmıştır. Alınan bilgiler 1995-2002 yılları arasındaki bütün güneşlenme süresi verileridir. Çalışmada hedeflenen bu 7 yıllık verilerin kullanılmasıyla 2003 yılı için güneşlenme sürelerinin hesaplanmasıdır. Bu çalışmada eğri uydurma, doğru uydurma ve ANFIS kullanılarak karşılaştırmalı olarak en iyi tahmin sonuçlarına ulaşılmaya çalışılmıştır. Çalışmada kullanılan yöntemlerden biri olan ANFIS ile birlikte; ANFIS'in bütün yıllara ait olan verileri daha iyi bir şekilde karakterize etmesi, tahmin sonuçlarını diğer test edilen iki yönteme göre daha yakın, daha düşük hatalarla bulabildiğini göstermiştir.

Potter ve Negnevitsky (2006), yaptıkları çalışmada ANFIS kullanarak rüzgârın tahmini için bir sistem geliştirmişlerdir. Yaptıkları çalışmada kullandıkları veriler Avustralya'nın Tazmanya bölgesine aittir. Çalışmada rüzgâr gücünün dünya üzerinde en hızlı büyüyen sektörü olduğunu ve rüzgâr tahminlerinin en yüksek şekilde kazanç sağlayabilmesi için rüzgarla ilgili birçok parametrenin bilinmesi ve kısa süreli olarak rüzgarların tahminlerinin oldukça başarılı şekilde yapılmasının gerekli olduğunu söylemektedirler. Yapılan tahminlerin hem rüzgârın hızı hem de rüzgârın yönü ile ilgili olması gerektiğini böylelikle de

rüzgâr türbinlerinin rüzgârdan en yüksek verim alacak yöne çevrilmesinin gerektiğini vurgulamışlardır. Çalışmada ANFIS ile diğer bir yöntemi karşılaştırmışlar ve diğer yöntem %30'luk bir hata oranına göre tahmin üretirken, ANFIS %4'ten daha az hata ile rüzgâr ile ilgili tahminler yapmıştır.

Çelik vd. (2007), yaptıkları çalışmada dinamometre tasarımı ve üretimini yapmışlardır. 2007'de yaptıkları bu çalışmada tasarımını ve üretimini gerçekleştirdikleri cihaz için içten yanmalı olan bir motorun bilgisayar kullanarak test edebilmesini sağlamışlardır. Aynı zamanda yaptıkları geliştirme ile birlikte çalışma esnasında cihazdan aldıkları deneysel veriler aynı anda bilgisayar ortamındaki sistemlerine aktarılabilmiştir. Çelik ve arkadaşlarının yaptıkları bu sistem ile birlikte motorun harekete ilk başladığı an ve sürtünmenin ortaya çıkardığı güç değerlerinin de ölçülmesi sağlanmıştır.

Altuntaş (2007), yaptığı yüksek lisans çalışmasında YSA'da LM öğrenmeli ileri beslemeli geri yayılım algoritması kullanarak yakıt tüketimi tahmininde bulunmuştur. Kullanmış olduğu üç katmanlı ağda Newton metodunu kullanılarak Uçuş Veri Kaydedicisinden elde etmiş olduğu hava hızı, uçak ağırlığı, irtifa, sıcaklık ve mach sayısını her uçuş fazında farklı türden giriş verilerini kullanarak yakıt akış miktarı tahmin etmiştir. Elde ettiği tahminler analiz edildiğinde makul bir değere sahip olduğunu ifade etmiştir.

Chen ve Yu (2007), yaptıkları çalışmada destek vektör regresyonu kullanarak sel tahmin yöntemi geliştirmişlerdir. Çalışmada sel baskınlarının sel tahmin yöntemleriyle önlemenin önemini vurgulamışlardır. Yapay sinir ağları ve destek vektör makineleri yöntemlerinin tahminler üzerindeki sonuçlarını karşılaştırmalarla göstermişlerdir. Yapay sinir ağlarında gizli nöron sayısı gibi belirlenmesi gereken yöntemlerin deneme ve yanılma yoluyla belirlenmesinin süreci uzattığından söz etmişlerdir. Destek vektör makinelerinin çalışma yapısındaki analitik işlemler ve vektörlerin otomatik bir şekilde belirlenmesi, bu algoritmanın başarılı yönü olarak gösterilmiştir. Çalışmalarında geliştirdikleri yöntemi Tayvan'da nehir üzerinde denemişlerdir. Nehirdeki parametreler



kullanılarak destek vektör makineleri ile birlikte sel tahmini yapmaya çalışmışlardır. Yöntem oldukça başarılı sonuçlar vermiştir.

Guo ve Wang (2008), yaptıkları çalışmada bilgisayar tabanlı bir dinamometre kontrolü yapmışlardır. Yaptıkları çalışmada dinamometre ile ilgili olan kontrollerden, donanım üzerinde olanları ve bazı yeni özellikleri de kapsayacak şekilde bütün kontrol yapılarını bilgisayar üzerinden gerçekleştirmeye çalışmışlardır. Buna ek olarak görsel ve arayüzleri ile daha anlaşılır biçimde olan test standına ait bir simülasyonu da geliştirmişlerdir. Çalışmalarında dinamometrenin bütün parametrelerini kontrol edebilmek için Toplu Araç Sistemi yani *Multi-Agent System* kullanmışlardır. Çalışmada yapılan bu geliştirmeler ve değişimler sayesinde kullanılan işçi gücü sıfıra azaltılmış, harcanan zamanın artması gibi bir durumlar karşılaşılmamış ve karşılaşılan hataların oranı da azaltılmıştır.

Zhang vd. (2008), yaptıkları çalışmada, dinamometrelerin çalışma esnasında değerlerin doğruyu ifade etmesi adına, ortam koşullarının iyileştirilmesi ve gerçekçi bir yapıda olmasını amaçlamışlardır. Çalışmayı başarılı bir şekilde gerçekleştirebilmek için bulanık-PID kontrolörü kullanmışlardır. Çalışmada hedefledikleri sonuçlara hem kısa zamanda hem de hatasız bir şekilde ulaşmaya çalışmışlardır. Çalışmada gerçekleştirilen sistem sayesinde hem optimum hızda hem de en yüksek doğrulukta araç değerlerine ulaşabilmişlerdir.

Wang ve vd. (2009), yaptıkları çalışmada hidrolojik tahmin için birçok yapay zekâ yöntemini deneyerek onların performanslarını, başarı ölçütlerini kıyaslayan kapsamlı bir çalışma yapmışlardır. Çalışmada otoregresif hareketli ortalama, yapay sinir ağları, adaptif ağ tabanlı bulanık mantık çıkarım sistemi, destek vektör makineleri, genetik programlama gibi yöntemler ile birlikte en iyi tahminsel sonuçlara ulaşmaya çalışmışlardır. Çalışmada geliştirme esnasındaki eğitim ve test sürecindeki kriterlerin farklılıklarına göre Genetik programlama, ANFIS ve destek vektör makinelerinin en iyi tahmin sonuçları gösterdiğini gözlemlemişlerdir.

Özgan vd. (2009), yaptıkları çalışmada ANFIS kullanarak asfalt numunelerinde kullanılan Marshall stabilite değerlerinin tahmin edilmesi noktasında uygulanabilirlik durumlarını araştırmışlardır. Bunu test edebilme amacıyla deneysel yöntemlerle elde edilmiş birbirinden farklı 37 deneyin bilgilerini kullanmışlardır. Bu verilerin 26 tanesini eğitime, 11 tanesini de teste ayırarak iki grup oluşturmuşlardır. Veriler üzerinde eğitim işlemi gerçekleştirildiğinde ortalama karesel hata değerini oldukça düşük olarak elde etmişlerdir. Çalışmada eğitim sırasında kullanılmayan 11 adet veri de test aşamasındaki model üzerinde çalışırken kullanılmıştır. Oluşturulan ANFIS modelinin tahminsel ürettiği sonuçlar ile test verileri üzerindeki değerlerin birbirleri ile oldukça yakın oldukları gözlemlenmiştir. Çalışma sonucunda Özgan ve arkadaşları, adaptif ağ tabanlı bulanık mantık çıkarım sisteminin, buna benzer ve tahmin içeren başka çalışmalar için de kullanılabileceği sonucuna ulaşmışlardır.

Demirkoparan vd. (2010), yaptıkları çalışmada petrol fiyatlarındaki değişebilen yapı üzerinde hem orta vade hem de uzun vadede tahmin etmek istemişlerdir. Çalışmada ham petrolün fiyatlarını sisteme girdi olarak tanımlamışlardır. Çalışmada bulanık mantıktan faydalanmışlar ve yapay sinir ağlarının birleştirilmesiyle adaptif ağ tabanlı bulanık mantık çıkarım sistemini kullanmışlardır. Oluşturdukları modelin başarısını test edebilmek adına ortalama mutlak yüzde hata değerini kullanmışlardır.

Ok (2010), yaptığı yüksek lisans tezi çalışmasında, elektrik enerjisinin talebi üzerinde tahminde bulunabilecek bir sistem geliştirmiştir. Yaptığı çalışmayla birlikte Türkiye'nin orta dönemli brüt elektrik enerjisinin talebinin ne kadar olacağını tahmin etmiştir. Tahmin yöntemi olarak da bulanık mantık ve yapay sinir ağlarının birlikte kullanılmasıyla ortaya çıkan adaptif ağ tabanlı bulanık mantık çıkarım sistemini kullanılmıştır. Problemin çözümünden tahminin başarısını karşılaştırabilmek adına bir de regresyon analizi yapmıştır. ANFIS te tahmin için sonuçları mutlak yüzde hata kriterinin değerini kullanarak incelemiştir. Yaptığı çalışmada, Ok regresyon analizi ile ANFIS'i kıyaslamış ve ANFIS'in daha yüksek performansa sahip başarılı sonuçlar üreten sonuçlara sahip olduğunu göstermiştir.

Subaşı vd. (2010), yaptıkları çalışmada adaptif ağ tabanlı bulanık mantık çıkarım sistemi kullanarak basınç dayanım tahmini için bir model geliştirmişlerdir. Geliştirdikleri bu modeli uçucu kül ikameli olan betonların basınç dayanımlarının tahmini için kullanmışlardır. ANFIS'te tahmin için veri girdisi olarak betonun yaş bilgisi, pull-off ve ultra ses geçişinin hız bilgileri verilmiş ve model çıktısı olarak da basınç dayanımı verisi tanımlanmıştır. Yapılan bu çalışma ile birlikte geliştirilen ANFIS modeli sayesinde tahmin ile ulaşılan beton dayanım sonuçları, ellerinde bulunan gerçek beton dayanım verileri ile oldukça düşük hata oranıyla birlikte benzerlik gösterdiğini, ANFIS'in başarılı bir şekilde tahmin ettiğini gözlemlemişlerdir.

Erilli vd. (2010), yaptıkları çalışmada enflasyon üzerinde öngörü yapabilecek bir tahmin sistemi geliştirmeye çalışmışlardır. Denedikleri yöntem ile birlikte oldukça değişken olan bu enflasyon değerlerini tahmin etmeye çalışmışlardır. Çalışmada yapay sinir ağlarını kullanan melez bir yöntem denenmiştir. Çalışmada ileri ve geri beslemeli yapay sinir ağları yöntemi kullanılarak Tüketici Fiyat Endeksi (TÜFE) tahmin edilmeye çalışılmıştır. Tahminlerin yüksek başarıya sahip olamamasının enflasyonun sadece rakamsal tahminlerden ziyade ülkenin karşılaştığı yeni durumlardan da etkilenmesinden kaynaklandığını gözlemlemişlerdir.

Kandananond (2011), yaptığı çalışmada elektrik tüketim değerleriyle elektrik tüketim talep tahminini yapmıştır. Çalışmasında 1986-2000 yılları arasındaki Tayland elektrik tüketimi değerlerini kullanmış ve farklı tahmin modelleri ile çalışmasını gerçekleştirmiştir. Tahminler için ARIMA, Yapay sinir ağları ve çoklu lineer regresyonunu kullanıp, bunlar arasında performans karşılaştırması yapmıştır. Yaptığı çalışmadan elde ettiği sonuçlara göre, en yüksek başarı yapay sinir ağı modelinde ortalama hata yüzdesi %0.996 olmuştur. Ortalama hata yüzdesi ARIMA modelinde %2.80981, çoklu lineer regresyonunda ise %3.2604527 çıkmıştır. Hata oranları arasında en düşük hata oranına sahip yapay sinir ağları en yüksek performansı göstermiştir.

Türkmen ve Korkmaz (2011), yaptıkları çalışmada uçağa ait hava veri bilgisayarına alternatif olarak Bulanık Mantık Sistemine Dayalı Uyarlanır Ağ kullanarak basınç irtifası tahmin eden bir hesaplama yöntemi geliştirmiştir. Beş katmanlı yapay sinir ağları yapısında geriye yayılım algoritması kullanarak en küçük kareler yöntemi ile DE+LM öğrenme algoritmasıyla basınç irtifası tahmin etmiştir. Giriş parametreleri olarak statik hava basıncı ve sıcaklık değerlerini sisteme birinci tip Sugeno üyelik fonksiyonlarında eğer/öyleyse kural setiyle tanımlamıştır. Neticede çok küçük hata payı ile gerçek değerlere yakın basınç irtifası tahmininde bulunmuştur.

Zhao ve Li (2012), yaptıkları çalışmada şasi dinamometresi üzerine bir çalışma yapmışlardır. Çalışmada hibrit araçların kullanmasına yönelik alan kontrol ağ yapısına dayanan bir şasi dinamometresi geliştirmişlerdir. Çalışmayı yapabilmek için öncelikle bu tip hibrit araçlarda ölçümlerin nasıl metotlar kullanılarak yapıldığını ve yükleme aygıtlarını inceleyip, analiz etmişlerdir. Çalışmada kullanmak üzere hibrit teknolojisine sahip elektrikli araçlar için dirençlerin gösterileceği bir yol direnç konusu üzerinde çalışmışlar ve simülasyon geliştirmişlerdir. Yapılanlar sayesinde hibrit elektrikli araçlar için bütün bağlantı ve montaj seviyelerinin geliştirilmesi sağlanmıştır. Çalışma süresince ortaya çıkarılan test metotları ve belirlenen standartlar sayesinde bu tip hibrit elektrikli araçlar için geliştirme sürecinde faydalanılabilecek ve geliştirmeye açık olan bir çalışma alanı sağlamışlardır.

Turhan vd. (2013), yaptıkları çalışmada binaların enerji tüketimlerini tahmin etmeye çalışmışlardır. Çalışmalarında bu tahminler için bulanık mantık ve yapay sinir ağları yöntemlerini kullanmışlardır. Bu tahminleri yapabilecek modelleri kullanmak için gerekli olan örnek verileri İzmir'de bulunan 3 farklı ilçeden 148 bina seçerek sağlamışlardır. Binalara ait mimari proje bilgileri ve ısı hesaplarının olduğu raporları kullanarak elde edilen bilgiler ile modeller için giriş bilgileri sağlanmıştır. Çalışmada bu veriler kullanılarak bir tane bulanık mantık modeli ve 4 tane de yapay sinir ağları için model oluşturulmuştur. Modellerin testlerine göre yapay sinir ağlarında ortalama mutlak yüzde hata bilgisi %4,1 çıkarken ortalama mutlak sapma değeri ise 6,57 olarak bulunmuştur. Aynı veriler ile

birlikte bulanık mantık modeli ile testler yapıldığında %4.86 ortalama mutlak yüzde hata değeri ve 7.59 ortalama mutlak sapma değeri hesaplanmıştır. Çalışmada testleri yapılan birbirinden farklı modeller içerisinde en iyi tahminlerin hesaplandığı yöntem yapay sinir ağları olmuştur.

Patel ve Parekh (2014), yaptıkları çalışmada muson yağışlarının tahmini için kullanılacak başarılı bir model araştırması yapmışlardır. Çalışmada adaptif ağ tabanlı bulanık çıkarım sistemi kullanmışlardır. ANFIS kullanarak Gandhinagar istasyonunun muson yağışlarını tahmin etmişlerdir. Bu çalışmada, yedi üyelikli melez bir model ile birlikte sıcaklık, bağıl nem ve rüzgâr hızını içeren üç girdisi olan bir sistem kullanmışlardır. Hazırladıkları bu sistem yağışların tahmini üzerine oldukça başarılı sonuçlar vermiştir.

Liu vd. (2014), yaptıkları çalışmada makro varlıklar için fiyat endeksi tahmin eden bir sistem geliştirmişlerdir. Geliştirdikleri bu modelde destek vektör makinelerini kullanmışlardır. Makroekonomiyi temsil edecek ekonomik ve istatistiksel 12 değişken bu model için seçmişlerdir. Destek vektör regresyonu yöntemi ile Çin enflasyonunu geleneksel ekonometrik yöntemlerden farklı olarak tahmin etmede kullanmışlardır. Araştırmacılar tarafından destek vektör makinelerinin kullanım amacı, değişkenlerle enflasyon arasında ilişkinin açıkça anlaşılmasını sağlamak olmuştur. Çalışmada ağırlık vektörleri ve birbirinden farklı kernel fonksiyonları kullanılmış ve en başarılı tahminin bulunması hedeflenmiştir. Çalışma sonunda geleneksel yöntemlerle destek vektör makineleri sonuçları kıyaslanmıştır. Karşılaştırma sonucunda destek vektör makinelerinin bulunduğu tahminlerin daha iyi olduğu gözlemlenmiştir. Model özellikle kısa-orta vadeli kararlar için faydalı olmuştur.

Es ve Hamzaçebi (2014), yaptıkları çalışmada enerji taleplerinin tahminini yapmışlardır. Çalışmada kullanılan veriler Türkiye'nin 1970-2015 yılları arasındaki gayri safi hasıla, binaların yüz ölçümü, taşıt sayıları, nüfus bilgisi, ithalat ve ihracatın değerleri şeklindedir. Bu veriler kullanılarak 1970 ile 2010 yılları arasındaki model girdileri olarak kullanılıp, 2011 ile 2015 yılları arasındaki Türkiye'nin enerji talepleri tahmin edilmeye çalışılmıştır. Çalışmada

tahmin yapmak için çoklu regresyon ile yapay sinir ağıları modelleri kullanılmıştır. Tahmin performansları karşılaştırıldığında yapay sinir ağlarının daha iyi sonuçlar verdiği gözlemlenmiştir.

Korkmaz (2016), yaptığı yüksek lisans tezi çalışmasında ilerleyen 10 dakikalık süreç için rüzgârın hızını tahmin etmeye çalışmıştır. Verilerin ölçümü için İstanbul'da farklı yüksekliklerde 10'ar dakika olacak şekilde aralıklar belirleyerek gerçekleştirmiştir. Rüzgâr hızı tahmininde yöntem olarak çoklu doğrusal regresyon, YSA ve ANFIS kullanmıştır. Çalışmasında topladığı verileri %75'e %25 olacak şekilde eğitim ve test için bölmüştür. Tahmin yöntemlerinden çoklu doğrusal regresyon ile %8,6, YSA ile %8,0 MAPE hata değeri ile başarılı sonuçlar almıştır.

Makas ve Karaatlı (2016), yaptıkları çalışmada, Türkiye'nin Isparta ilinin Aksu ilçesine ait Yenice deresinden beslenen bir hidroelektrik santralin 12 aylık elektrik üretimi tahmin edilmeye çalışılmıştır. Aylık ortalama nisbi nem, aylık ortalama debi ve aylık ortalama yağış verilerini girdi verileri olarak kullanıp ileri beslemeli geri yayımlı sinir ağı modeli ile 12 aylık elektrik üretimi tahmin çalışması yapılmıştır. Kurulan modelin aylık elektrik tüketimini başarı ile temsil ettiği belirtilmiştir.

Aksoy (2018), yaptığı yüksek lisans çalışmasında Bursa ilinde bulunan 16 türbine sahip olan santralden 2 yıl süresince 10'ar dakikalık periyotlarla ölçülmüş olan rüzgâr hızı ve sıcaklık verilerini kullanmıştır. Rüzgâr hızı tahmininde ise çoklu doğrusal regresyon, destek vektör regresyonu, K-En yakın komşu regresyonu ve YSA kullanmıştır. Tahminlerden en başarılı olanlar destek vektör regresyonu ve K-En yakın komşu regresyonu olmuştur. Verilerin %80'i eğitim, %20'si testte kullanılmıştır. Devamında diğer başarılı yöntem ise YSA olurken, en kötü sonuçlar ise çoklu doğrusal regresyon ile alınmıştır. Hata Karelerinin Ortalaması 0.014 ile 0.168 arasında bulunmuştur. Sonuçlarının başarılı olmasının sebebi, veri olarak gerçekleşen rüzgâr hızını kullanmasından kaynaklanmıştır.

### 3. MATERYAL VE YÖNTEM

Motorlu araçların performans testleri dinamometre cihazları kullanılarak yapılmaktadır. Dinamometreler kullanım yeri ve şekline göre çeşitlilik göstermektedir. Kullanım alanlarına bakılarak, kullanımı en kolay hesaplanması en hızlı olan dinamometreler, Şasi dinamometreleridir. Bu çalışmada öncelikle dinamometrelerin genel yapısı ve çeşitlerinden bahsedilecektir. Motorlu araçların performans testlerinin tahminleri için öncelikle hazırlanan dinamometre cihazı ile ölçümler ve test yapılacak araçların dış parçalarının hasar durumları incelenecektir. Sonrasında ise makine öğrenmesinde DLR, MLR, RF, YSA ve ANFIS kullanılarak regresyon analizleri yapılacaktır. Çalışmada hedeflenen dinamometre test sonuçlarıyla elde edilen araçların beygir güçlerinin, aracın genel ve hasar bilgilerini kullanarak belirlenen makine öğrenmesi yöntemleriyle en yakın tahminleri elde edebilmektir. Çalışmada, dinamometrelerin denetlenmesi kısmında arduino kart, kodlamasında ise C programlama dilinden yararlanılmıştır. Şasi dinamometresi ile haberleşmenin ve araç performanslarının ölçümü için C# dilinde Visual Studio 2017 editöründe yazılım geliştirilmiştir. Alınan bütün veriler sonrası araç performans tahmininde bulunmak için *scikit* kütüphanesinin altındaki regresyon algoritmaları Python dilini kullanılarak hesaplanmıştır.

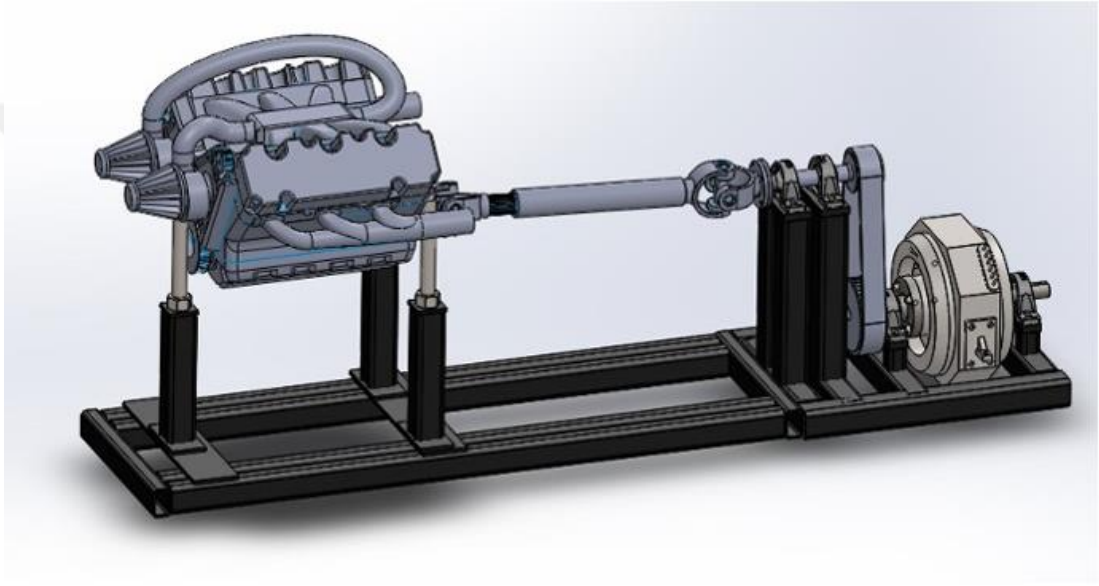
#### 3.1. Dinamometre

Ölçüm yapılan türe göre dinamometreler birçok kategoriye ayrılabilirler. Şasi dinamometreleri, motor dinamometreleri, endüstriyel dinamometreler, motosiklet dinamometreleri bunlardan bazılarıdır. Fakat yapısal alanda Motor dinamometreleri ve şasi dinamometreleri olarak iki kategoride incelenebilir.

##### 3.1.1. Motor Dinamometresi

Motor Dinamometreleri en çok fabrikalarda kullanılmaktadır. Bu dinamometrenin kullanılabilmesi için öncelikle araç üzerinden motor sökülme gerekmektedir. Sonrasında motorun volan çıkışı bağlantıyı gerçekleştirecek

parçalar kullanılarak dinamometreye sabitlenmektedir. Ölçüm işlemi için elektriksel bağlantılar sağlandıktan sonra motor başlatılır. Ölçüm yapılan şey motorun volan çıkışındaki güçtür. Ölçüm sonrası elde edilen güç yalın olarak motora ait güçtür. Bu sistemlerdeki en büyük zorluk ölçüm işlemleri için motorun ve motordaki elektriksel bağlantıların, üzerinde bağlı olduğu araçtan sökülmesinin gerekmesidir. Şekil 3.1’de örnek bir motor dinamometresi tasarımı gösterilmektedir. Şekildeki dinamometreyi diğerlerinden ayıran en önemli fark yüksek ölçüm hassasiyetinin olmasıdır (Çelik vd., 2007).



Şekil 3.1. Motor dinamometresi şeması (Focus Applied Technologies, 2019)

Motor dinamometresi ile çalışmak, ölçümler sonucunda performans etkileşimleri sonucunda motor üzerinde değiştirme ve geliştirmenin rahat bir şekilde yapılabilmesini sağlamaktadır. Araç yarışı firmaları ve fabrikalar bu cihazları sağladıkları kolaylık ve yapılabilen geliştirmeleri hızlı gözlemleyebilmek adına yaygın bir şekilde kullanılmaktadırlar. Dünya üzerinde satılan bütün araçların fabrika verilerinde yer alan beygir gücü bilgileri de bu cihazlardan okuma yapılarak sağlanmaktadır. Şekil 3.2’de kullanımda olan bir motor dinamometresi gösterilmektedir.





Şekil 3.2. Motor dinamometresi (Martino Race Engines, 2019)

### 3.1.2. Şasi Dinamometresi

Şasi dinamometreleri araç tekerleri üzerinden ölçümlerini gerçekleştiren, günümüzde birçok servis, oto ekspertiz ve ölçüm merkezlerinde kullanılmakta olan bir sistemdir. Şasi dinamometresi konum olarak yer seviyesi ya da silindirlerin içerisine konulacak yerden maksimum 40 cm yükseklikte yeni bir kat yapılan bir yerde konumlandırılmaktadır. Test yapılacak araç, çekiş gücünün olduğu tekerleri, dinamometrenin silindirleri arasına gelecek şekilde yerleştirilir. Araç cihaz üzerinde sabitlendikten sonra, araçtan en yüksek performans alınacak şekilde gaza basılarak ölçüm yapılır. Şasi dinamometreleri, araçlar için bir yol simülasyonu olarak da kullanılabilir. Şekil 3.3'te örnek bir şasi dinamometresi gösterilmektedir.



Şekil 3.3. Şasi Dinamometresi

Şasi dinamometrelerinin tasarımlarında farklılıklar olsa da sistemin temel prensipleri aynıdır. Ölçüm işleminde en kolay yöntem çekiş gücünün olduğu tekerlerden etrafında olan silindirlere iletilmekte olan gücü hesaplamaktır. Bu sistemler araç üzerinden herhangi bir parça sökmeye gerek kalmadan yapılabildiği için de kullanım oranı en yüksektir (French ve Stark, 2000). Hafif tip olarak tabir edilebilecek araçların performans testleri şasi dinamometresi ile yapılmaktadır (Pelkmans ve Debal, 2006). Şasi dinamometreleri Atalet ve Frenli olmak üzere iki kategoride incelenebilir;

**Atalet tipi dinamometreler:** Atalet tipi (inertial type) dinamometreler hesaplı olmaları ve kullanım kolaylığı dolayısıyla günümüzde en çok tercih edilen dinamometrelerdir. 2 çeker ve 4 çeker araçlar için çeşitli modelleri vardır. Bu dinamometrelerde ölçüm prensibi cihazın merdanelerinin atalet kuvvetine dayanmaktadır. Dolayısıyla düzgün, tekrarlanabilen ölçümler yapmak için merdane et kalınlıkları yüksek (ortalama 4 cm et kalınlığı, ortalama 320 mm merdane çapı) imal edilmektedir.

**Frenli tip dinamometreler:** Bu tip dinamometrelerde frenleme etkisi yaratan, araçlardaki retarder sistemine benzer bir düzenek kullanılmaktadır. Bu düzenek dönen silindirin miline bağlıdır ve yükü tekerleklere aktarır. Aktarılan yükü

ölçebilmek için ise bu düzeneğe bağlı bir kol ve yük hücresi kullanılmaktadır. Bu tip dinamometrelerde büyük ve ağır silindirlere ihtiyaç yoktur çünkü gereken yük frenleme sistemi vasıtasıyla sağlanır. Bu yük yazılım ile kontrol edilebildiği için bu tip dinamometreler yol simülasyonu, sabit devirde test, sabit yükte test gibi birçok işleve sahiptir.

Bazı 4 çeker dinamometrelerde ön ve arka merdanelerin aynı hızda dönebilmesi için ön ve arka aks merdaneleri birbirine bağlantılıdır. Bazı 4x4 araç modellerinde ön ve arka tekerlekler aynı hızda dönmediğinde problemler ortaya çıkabilmektedir.

Bu çalışmada kullanılacak olan dinamometre gerçek yol şartlarını da sağlayan atalet tipi şasi dinamometresidir. Şasi dinamometresiyle ölçülen güç tekerler gücüdür (Martyr ve Plint, 2007; Ribbens, 1998). Çalışmada performans değerlerinin tahminsel işlemleri için öncelikle şasi dinamometresi ile haberleşen kart ve kullanılan bu kartın haberleşip hesaplamalar yaparak aracın beygir gücünü hesaplayan yazılım sistemleri geliştirilmiştir.

### **3.2. Makine Öğrenmesi**

Öğrenme tecrübe, çalışma ve eğitim yoluyla davranışlarda meydana gelen ölçümlenebilir ve kalıcı olabilen değişimlerdir. Öğrenilen bilgilerin artması insanların herhangi bir problem sonucuna ulaşmasında kendisinin yetersiz olduğunu göstermektedir. Verilerin büyümesiyle analizlerin yapılabilmesi ve öğrenilmiş olarak insanmış gibi cevap verilebilmesi için 1950'li yıllarda makine öğrenmesi terimi kullanılmaya başlanmıştır. (Turing, 1950). Makine öğrenmesi, bilgisayarın örnek verilerden veya geçmiş verilerden sağlanan deneyimlerin kullanılarak, geleceğe yönelik durumları tahmin edebilmesi ve bu tahminlerle model ortaya çıkarabilmesine yarayan yapay zekanın bir alanıdır. (Alpaydın, 2004).

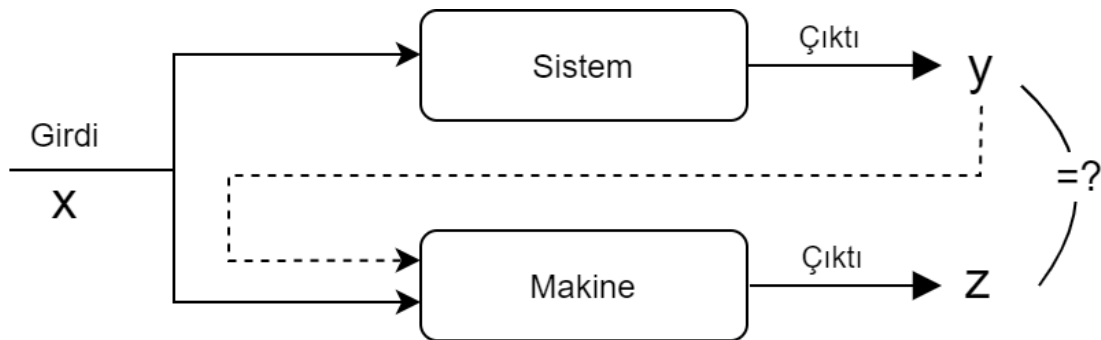
Makine öğrenmesi, D deneyimi ile birlikte buna uygun olan G görev sınıflarının, P performansı ile birlikte öğrenmesinin sağlandığı bir bilgisayar sistemidir. Bu

sistemde G görevlerinin işlem sırasındaki performansı P kullanılarak ölçümlenebiliyorsa, D deneyimleriyle birlikte geliştirilmeye uygunluk sağlar (Mitchell, 1997). Bahsedilen bu yapı için aşağıdaki örnek gösterilmiştir.

- Görev G: Tavla Oynamak
- Performans P: Rakipler karşısındaki oyunların kazanılma yüzdeliği.
- Deneyim D: Sistemin kendi kendine alıştırma ve öğrenme için oyun oynaması.

Geliştirilen bilgisayar programlarının, öğrenme başarısı önceki olayların sağladığı tecrübelerin örneklenmesiyle sağlanır. Bilgisayar programları kullanılan örneklerin sayısı arttıkça tecrübe kazanır ve artık öğrenilen bilgilere göre sonuçlar üretmeye başlar. Örnek sayısının artması üretilen sonuçların doğruluğunun artmasıyla doğrudan ilişkilidir. Dolayısıyla Makine öğrenmesi yöntemlerine tecrübe tabanlı öğrenme de denilebilir (Öztemel, 2003).

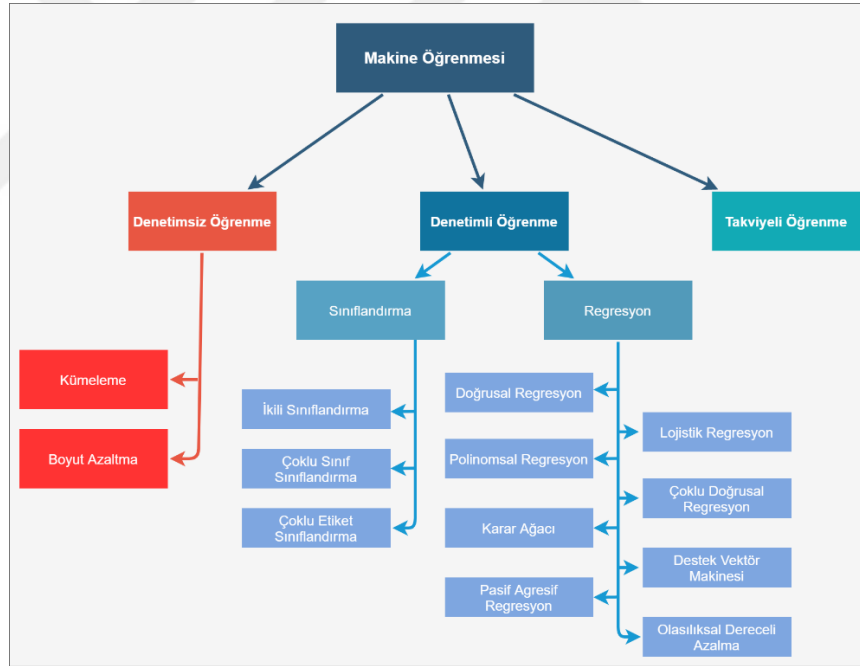
Makine öğrenmesi birçok alanda sıklıkla kullanılmaktadır. Bu alanlar finanstan eğitime, sağlıktan otomotiv sektörüne oldukça büyük bir yelpazede uygulanabilmektedir. Bu alanlarda makine öğrenmesinin kullanımı ile birlikte gelecek hakkında oluşma ihtimali olan olayları göstermesidir. Bu gösterimler veriler üzerinden sınıflandırma ve tahminlerdir. Belirtilen alanlardaki ileriye dönük yapılabilecek ufak hatalar geri dönüşü olmayacak şekilde hem zamansal hem maddi kayıplar oluşturabileceğinden makine öğrenmesi öngörülleri oldukça önem kazanmaktadır. Makine öğrenmesi çalışma şekli Şekil 3.4'te gösterilmektedir.



Şekil 3.4. Makine öğrenmesi şeması (Forsyth, 1989).

Yukarıda belirtilen diyagramda x veri olarak bir girdi vektörü y ise girdiye göre uygun olan bir çıktıdır. Bu sistemde,  $y = f(x)$  fonksiyonu, tanımlaması önceden belirli durumlara göre hazırlanmış bir modeldir. Sisteme tanımlanmış x değerlerine göre y çıktıları ile makine öğrenmesi en doğru çıkarımları yapabilmek için model oluşturur. Oluşturulan modelde tanımlanan kritere göre, eğitim kümesinden gelen veriler eğitilir ve test kümesi oluşturularak z değerleri incelenir (Uzun, 2005).

Makine öğrenmesi yöntemi, uygulama süresince problemin durumuna ve eldeki veri setine bakılarak karar verilir. Şekil 3.5'te makine öğrenmesi yöntemleri gösterilmektedir. Bu yöntemlerin genel başlıkları Denetimli Öğrenme (Supervised Learning), Yani Denetimli Öğrenme, Denetimsiz öğrenme ve pekiştirmeli öğrenme olarak 4 ana başlıkta gösterilmiştir.

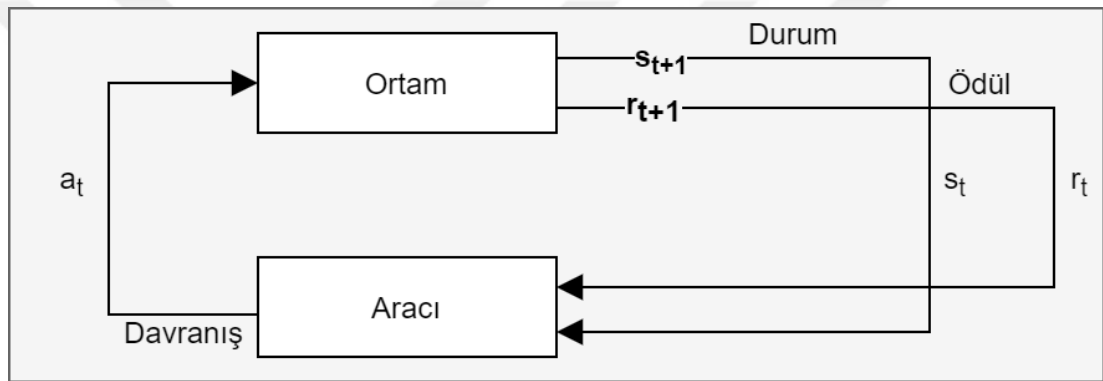


Şekil 3.5. Makine öğrenmesi yöntemleri

### 3.2.1. Pekiştirmeli öğrenme

Pekiştirmeli Öğrenme (Reinforcement Learning) yöntemiyle birlikte sistemler, karşısına çıkan durumlar ile ilgili en iyi değerlendirme sonuçlarını ortaya koyabilmek için kendilerini adapte ederler (Sutton ve Barto, 1999). Test ve eğitim

adımları birlikte gerçekleştirilir. Dolayısıyla sistem kontrolcü ile sürekli devam eden bir bağlantıya sahiptir. Amaç hedefli bir öğrenme türüdür. Pekiştirmeli öğrenmede yapılan işlem sistemin üretmiş olduğu sonuçları sürekli olarak girdi şeklinde verilen değerlere göre doğru yanlış olarak değerlendirir. Bu değerlendirme işlemi ödül olarak adlandırılmaktadır. Bir de bu sistemin her değerlendirme işleminden sonra ödül sonucunun yanı sıra durum bilgisi olarak değerlendirme durumuna göre sistemde değişiklikler yapılarak daha doğru çalışması sağlanmaktadır. Sistemin hedefi yapılan değerlendirmelere göre kendisini geliştirmek ve en fazla ödülü almaya çalışmaktır. Şekil 3.6'da pekiştirmeli öğrenme yönteminin genel yapısı gösterilmektedir.



Şekil 3.6. Pekiştirmeli öğrenme yönteminin genel yapısı

Satranç ve tavlâ oyunları pekiştirmeli öğrenme yöntemine verilebilecek birer örnektir. Bu oyunların kuralları basit ve değişmez olmasına rağmen oynayan oyuncuya göre çok fazla hamle çeşitliliği bulunmaktadır. Bundan dolayı herhangi bir oyuncunun hamlelerinin tahmin edilmesi zordur. Pekiştirmeli öğrenme yöntemi sürekli olarak karşı oyuncunun yapacağı hamlelere göre öğrenme işlemi yaparak tahminlerini geliştirir (Alpaydın, 2004).

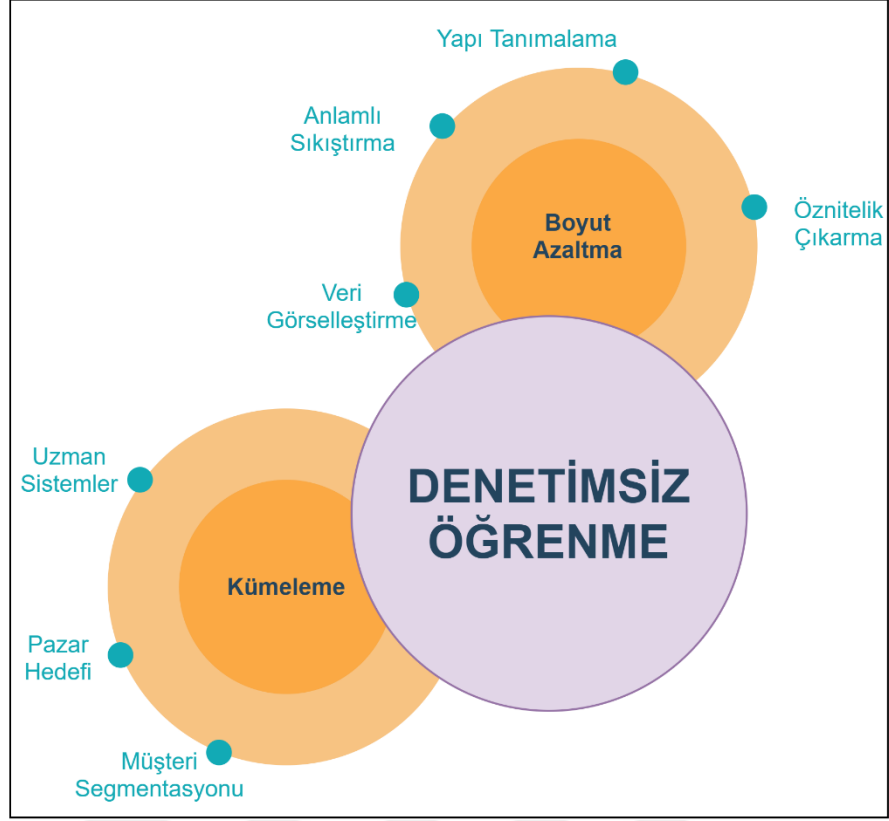
### 3.2.2. Denetimsiz öğrenme

Denetimsiz Öğrenme (Unsupervised Learning) sadece gözlem yapılarak öğrenme işlemi yapan bir yöntemdir. Bu yöntemde herhangi bir şekilde etiketleme işlemi yoktur. Bütün veriler hiçbir bilgi ya da kural verilmeden sisteme yüklenir. Bu yöntemde sisteme öğretim yapacak denetçi yoktur. Denetimsiz öğrenme

yöntemi, girdi bilgilerinin kendi arasındaki durumlarını ilişkilerini inceleyerek sınıflandırma ve kümeleme gibi sonuçlar üretmektedir. Bu yöntemde üretilen sonuçların çıktı bilgileri ya da sınıflandırma yöntemleri yoktur. Dolayısıyla sonuçları ölçümlemek zordur (Mohri vd., 2012).

Denetimsiz öğrenme yönteminde kullanılan algoritmalar sayesinde hedeflenen verilerin sahip olduğu gizli öznitelikleri ortaya çıkarmaktır. Denetimsiz öğrenme yöntemi ile birlikte ulaşılan çözümleri doğru veya yanlış, eksik ya da fazla gibi anlamlandırmak doğru değildir, ama yöntemde işlenecek verilerin ortaya çıkardığı özniteliklerin incelenerek bir bütünlük elde etmeye çalışılır. Örneğin bir firmanın müşteri hizmetleriyle konuşup sonrasındaki ankete katılan müşterilerin verdikleri cevaplar denetimsiz öğrenme yönteminin sahip olduğu algoritmalar ile kullanılarak müşterilerin verdiği cevaplara göre müşterileri gruplayabilir, bu gruplama sonrasında da müşterilere özel indirimler ya da geri dönüşler sağlanabilir, ya da resimlerin sıkıştırılması için denetimsiz öğrenme yöntemleri kullanılabilir, böylelikle resim üzerindeki renklerden çok yakın olanlar tek bir renk olarak kümelenecek resmin aynısı olmasa da çok benzer bir halini daha küçük bir boyutta elde etmek mümkündür (Alpaydın , 2004).

Sahip olunan verilerden benzer olanlarını kümelemek ya da boyut azaltma işlemleri denetimsiz öğrenme yönteminin temel amaçlarındandır. Kümeleme, Hiyerarşik Kümeleme, Temel Bileşen Analizi, Tekil Değer Ayrıştırması, K-ortalama Kümeleme, Boyut İndirgeme denetimsiz öğrenme yöntemlerinden bazılarıdır. Şekil 3.7’de denetimsiz öğrenme kapsamı gösterilmektedir.



Şekil 3.7. Denetimsiz öğrenme kapsamı

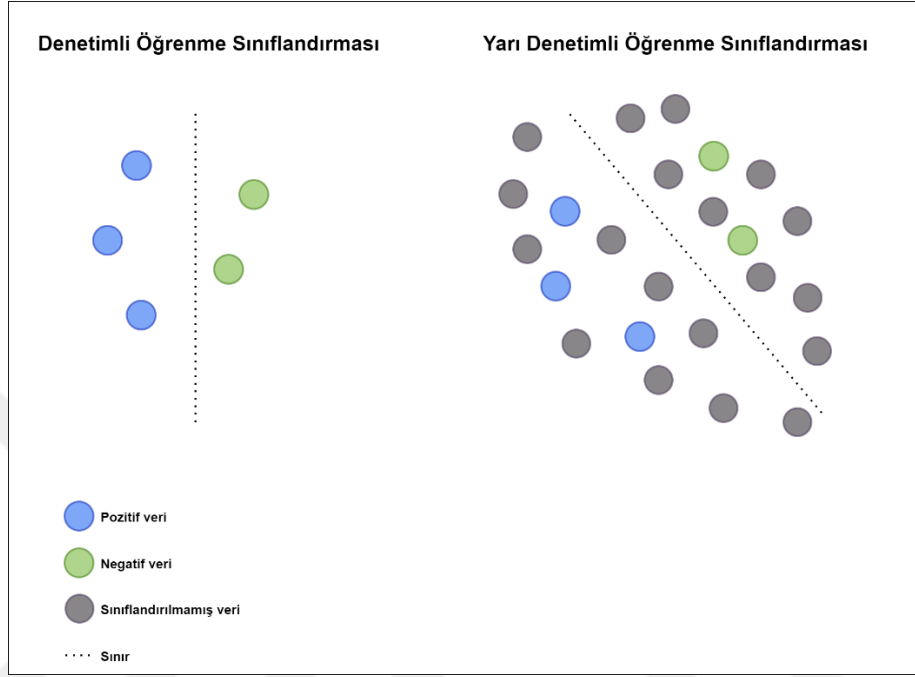
### 3.2.3. Yarı Denetimli öğrenme

Yarı denetimli öğrenme yöntemi, denetimli öğrenme ve denetimsiz öğrenme yöntemlerinin özelliklerine kısmen sahip olarak oluşturulmuş bir yöntemdir. Bu yöntemde eğitim verilerinin çoğu denetimsiz öğrenme yönteminde olduğu gibi etiketlenmemiş şekildedir. Fakat eğitim verilerinin içerisinde etiketlenmiş veriler de bulunmaktadır. Sistem en iyi çıkarımı yapabilmek için etiketli ve etiketsiz verilerin hepsini kullanarak tahminler yaparak sistemin yapısını anlamaya çalışır (Huang vd., 2006).

Verilerin etiketleme işleminin maliyetli ve uğraştırıcı olduğu zamanlarda bu yöntem kullanılmaktadır. Etiketlenmemiş olarak kullanılan verilerin öznitelikleri bilinmese de sonuç değerlerinde önemli bir etkisi olduğu bilinmektedir. Yarı denetimli öğrenme yöntemi de hem etiketlenmemiş hem de etiketlenmiş verileri kullanarak maliyeti düşük ve hızlı bir çözüm sunmaya çalışmaktadır. Yarı denetimli öğrenme yöntemi sınıflandırma ve regresyon gibi problemleri



çözmekte kullanılabilmektedir. Şekil 3.8’de yarı denetimli öğrenme yöntemiyle ilgili açıklayıcı bir örnek gösterilmektedir. Mavi, yeşil ve siyah daireler pozitif, negatif ve etiketlenmemiş verileri ifade etmektedir. Kesikli çizgi sınıflandırıcının karar sınırını belirlemektedir.



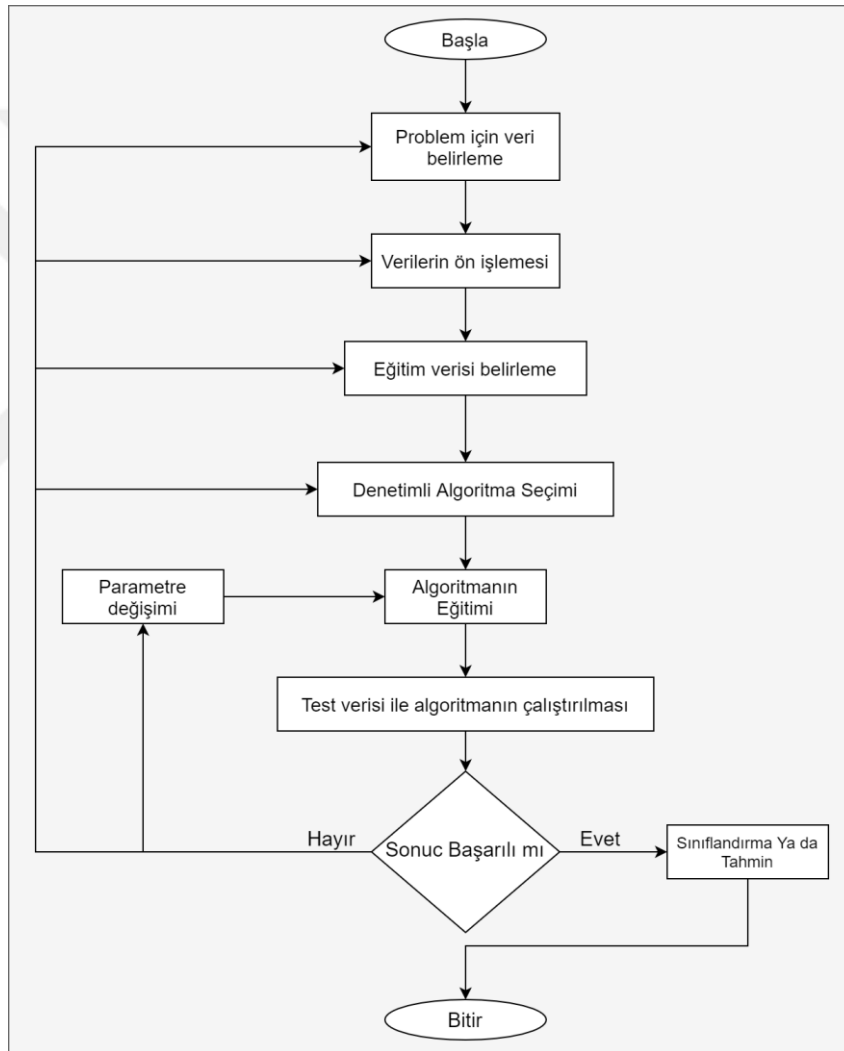
Şekil 3.8. Yarı denetimli öğrenme yöntemiyle ilgili açıklayıcı bir örnek

#### 3.2.4. Denetimli öğrenme

Denetimli Öğrenme yöntemi (Supervised Learning) eğitim sürecinin yer aldığı bir öğrenmedir. Giriş ve çıkış bilgilerine sahip eğitim verisi bulunmaktadır. Bu eğitim verisi uygulayan kişi tarafından sisteme tanıtılıp anlamlı sonuçların elde edilmesi hedeflenmektedir. Öğrenme işlemi süresince giriş ve çıkış bilgilerine sahip bilinen veriler girdi olarak sisteme tanıtılır. Makineye verilen bu girdiler ve sonuçlar ile öğretim işlemi tamamlandıktan sonra, öğrenilen bu bilgiler kullanarak öncesinde hiç gösterilmemiş veriler için tahminler yapılmaktadır (Huang vd., 2006; Ayodele, 2010).

Bir problemin değerlendirilmesinde eğitim ve test olarak adlandırılan iki kısma ayrılmış veriler bulunmaktadır. Denetimli öğrenme yöntemi kullanılırken problem çözümü için eldeki verilerden rastgele olarak seçim yapılır ve seçime

göre bir kısım eğitim verisi diğer kısım test verisi olarak seçilir. Eğitim verisi olarak seçilen kısım için denetimli öğretim yöntemlerinden en uygun çözümleri üreten bir algoritma seçilip uygulanarak öğrenme modeli ortaya çıkartılır. Ortaya çıkartılan bu model sadece eğitim verilerine göre ayarlanmıştır. Daha sonraki adımda ise diğer kısımdaki test verileri, elde edilen model ile kullanılarak sistemin ne kadar doğru tahmin ettiği ölçülmektedir. Yapılan tahminlerin doğruluğuna göre sistem yeni test verilerine de uygulanmaktadır (Özkan, 2013). Denetimli öğrenmenin temel akış diyagramı Şekil 3.9’da gösterilmektedir.



Şekil 3.9. Denetimli öğrenme aşamaları (Kotsiantis, 2007)

Denetimli Öğrenme Yönteminde yer alan modeller regresyon modelleri ve sınıflandırma modelleri olacak şekilde temelde iki kategoriye ayrılmışlardır (Theodoridis, 2015; Marsland, 2015). Ancak üzerinde çalışılan problemin

çözümüne göre hiyerarşik ve katmanlı gibi farklı modeller de literatürde yer almaktadır (Hastie vd., 2001; Suthaharan, 2016).

Denetimli öğrenme modelleri literatürde, genellikle bağımlı değişkenin sürekli ya da kategorik olmasına bağlı olarak regresyon ve sınıflandırma modelleri şeklinde ikiye ayrılmıştır (Barber, 2012; Theodoridis, 2015; Marsland, 2015). Ancak bazı kaynaklar denetimli öğrenme modellerini; regresyon (tahmin), sınıflandırma (matematiksel), hiyerarşik, katmanlı vb. farklı modellere de ayırmaktadır (Hastie vd, 2001; Suthaharan, 2016).

### ***Regresyon yöntemi***

Regresyon iki ya da daha çok bilinmeyen değişkenlerin sahip olduğu ilişkileri sayısal biçimde ifade ederek tahminler ya da kestirimler yapmak için kullanılır. İstatistiksel bir analiz yöntemidir. Regresyon bağımlı değişken ya da değişkenlerin bağımsız değişkenler tarafından sonuca ulaşılmaya çalışıldığı bir yöntemdir (Weisberg, 2005; Xu, 2003).

Regresyon yöntemleri, değişkenler arasında ortaya çıkan matematiksel fonksiyonlara göre sabit olan değişimlerde doğrusal model, sabit olmayan değişimlerde doğrusal olmayan model olarak ikiye ayrılmaktadır (Orhunbilge, 2002).

### ***Sınıflandırma Yöntemi***

Sınıflandırma yönteminde kullanılan algoritmalar, problem için sahip olunan verilerin öznitelikleri doğrultusunda sınıflandırılmasını sağlamaktır. Bu yöntemde sahip olunan veriler ile, kendi içerisinde tür, koşul, sınıf gibi bilgilere göre ayrılarak eğitildiğinde yeni gelen verilerin hangi sınıfa ait olacağı bilgileri tahmin edilebilmesi amaçlanmaktadır (Özekes, 2003). Kredi onayı, endüstriyel kalite kontrolleri, hastalık teşhisleri, resim tanıma gibi birçok problemde sınıflandırma yöntemi kullanılmaktadır (Dunham, 2003).

### 3.2.4.1. Destek vektör regresyonu

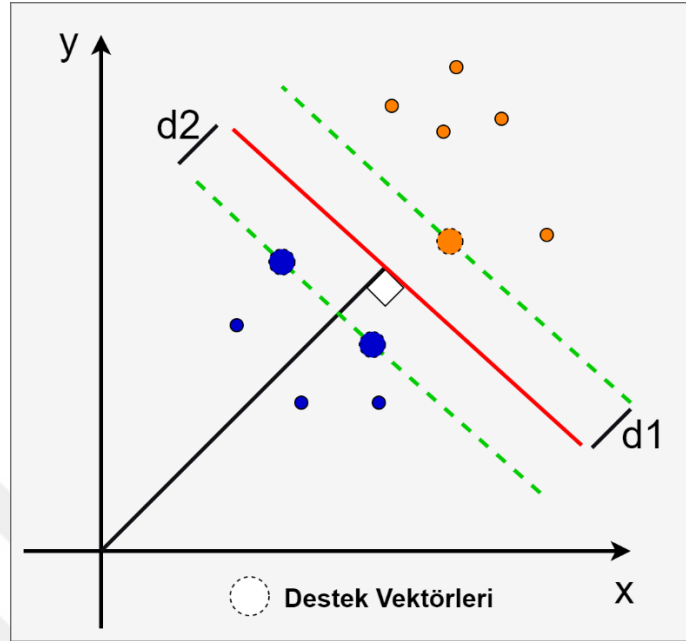
Vapnik ile arkadaşlarının Rusya’da geliştirmiş olduğu destek vektör makineleri algoritması lineer olmayan bir genelleştirme algoritmasıdır. Bu ekip 30 yılı aşkın süredir destek vektör makineleri algoritması modelini geliştirmekle uğraşmışlardır. İstatistiksel öğrenme teorisinin üzerine kurulmuş bir algoritmadır. Bu algoritma, lineer olmayan gerçek ortamından lineer fonksiyonları nonlinear fonksiyonlar ile değiştirme işlemi yapmaktadır (Wang vd., 2012). Bunun sonucu olarak destek vektör makineleri başarılı bir genelleştirme yapabilmekte ve bundan dolayı çoğu alanda kullanılabilir.

En iyi sonuçlara ulaşabilme kabiliyeti, karmaşık olmayan yöntemine rağmen gerçekçi tahminleri ve genelleştirme yeteneği ile destek vektör makineleri oldukça kullanılan bir tahmin ve öğrenme aracıdır. Buna ek olarak bu algoritma diğer algoritmalarla kıyasla oldukça başarılı bir sınıflandırma yöntemine sahiptir. Borsa tahmini, ses tanıma, piyasa tahminleri, hava durumu tahminleri ve başka birçok alanda kullanılmaktadır (Smola ve Scholkopf, 2004).

Destek vektör makinelerinin sayısal değerlerdeki tahminsel hesaplamaları için geliştirilmiş olan yöntemi destek vektör regresyonudur (DVR). Destek vektör regresyonu incelemesi öncesinde, destek vektör makinelerine, bu algoritmanın sınıflandırma yöntemine bakmak, regresyonun daha iyi anlaşılmasını sağlamaktadır.

Destek vektör makineleri “n” boyutlu olarak belirtilen uzay içerisinde, sahip olduğu verileri iki grup vektör gibi kümeleyen bir yöntemdir. Sınıflandırmanın en başarılı şekilde olabilmesini sağlayabilmek, sınıflandırma yapılacak olan sınıfların ortasından geçecek hiperdüzlem sınırının optimize edilmesiyle mümkündür. Sınıflandırması iyi yapılmış bir sistemdeki hiperdüzlem, kümelerin sahip oldukları veri setlerinin birbirlerine değer olarak en yakın olanlarının üzerinden geçirilen iki paralel doğrunun tam ortasındaki noktada olmaktadır. Çizilen paralel doğrular, geçirilen hiperdüzlemin sınırlarını oluşturur. Destek

vektörleri ise en yakında bulunan üyelerin üzerlerinden geçmektedir. Şekil 3.10'da bu vektörler gösterilmektedir (Cornell, 2020).



Şekil 3.10. Destek vektör regresyonu vektörleri (Kotsiantis, 2007)

Makine öğrenmesi algoritmalarında olduğu gibi destek vektör makinelerinde de herhangi bir problemin fonksiyonunun tahmin edilmesi, regresyon problemi olarak adlandırılmaktadır (Drucker vd., 1997).

$$f(x, \lambda) = X(\delta^d) \rightarrow \delta \quad (3.1)$$

$X \delta^d$  gibi bir veri uzayını temsil etmektedir.  $\lambda \in A$ ,  $A$  bağımsız özdeş dağılımlı  $N$  adet örneği olan parametreler kümesi iken

$$(x_1, y_1), \dots, (x_b, y_b), \quad x_b \in X(\delta^d) \text{ ve } y_b \in \delta \quad (3.2)$$

Sonuçları bilinmeyen  $p(x, y)$  dağılımında olmaktadır. Bu kısımda istenen şey beklenen riskin en küçük sonuçlara sahip olmasını sağlamak için gerekli bir  $f(x, \lambda)$  fonksiyonunu bulabilmektir (GUNN, 1998).

$$R[\lambda] = \int l(y, f(x, \lambda)) P(x, y) dx dy \quad (3.3)$$

Denklem 3.3'deki  $l$  olarak belirtilen sembol kayıp fonksiyonu ifade etmektedir. Çoğu zaman  $p(x,y)$  tahmini dağılımı bilinmemektedir. Dolayısıyla  $R[\lambda]$  beklenen risk hesaplaması ve azaltma işlemi mümkün değildir. Ama denklem 3.2'deki bilgiler ışığında  $p(x,y)$  ile ilgili tahmin yürütülebilmektedir. Bu sayede denklem 3.4'teki denklemde gösterilen deneysel risk bilgisi beklenen riskin yaklaşımı olarak hesaplanabilmektedir (Vural, 2007).

$$Remp[\lambda] = 1N \sum l(y_i, f(x_i, \lambda)) \quad (3.4)$$

Yapılan hesaplamalarla yalnızca deneysel risk üzerinde optimize işlemi yapmak küçük örneklemelerden dolayı tahmin etmede yanlış sonuçlara kullanıcı götürebilir, kullanıcılar için problemler oluşturabilir. Bu tip problemler de Vapnik ve Chervonenkis teorisi veya istatistiksel teori kullanılarak çözülebilir. VC teorisi 3.5'teki denklem sayesinde deneysel risk ile beklenen riskin kapsayacağı alanı  $1 - \eta$  olasılık sınırlarında belirlemektedir (Vapnik ve Chervonenkis, 1964).

$$R[\lambda] \leq Remp[\lambda] + \sqrt{h(\ln 2Nh + 1) - \ln \eta} 4N, \forall \lambda \in A \quad (3.5)$$

Denklem 3.5'teki  $h$   $f(x, \lambda)$  fonksiyonuna ait VC boyutudur. Beklenen risk değerinin en az olabilmesi için VC boyutunun en az değerlerde olmalıdır. Sayısı belirli bir düzeyde olan örnekler için en uygun VC boyutları vardır. Fakat sayı olarak az bir veriye sahip örneklerin  $h$  seçinin doğru yapılması oldukça önemlidir. Bundan dolayı Vapnik, yapısal risk minimizasyonunu geliştirmiştir. Regresyon yapan destek vektör makinesi, destek vektör regresyonu olarak tanımlanmıştır. Destek vektör regresyonunda hedeflenen şey, denklem 3.5'te gösteriler regresyon riskini azaltarak  $f(x, w, b)$  fonksiyonu belirlemektedir (Smola ve Scholkopf, 2004).

$$Rreg(f) = 12(w, w) + C \sum l(f(x_i), y_i) \quad (3.6)$$

Denklem 3.6'daki ilk kısımda gösterilen terim destek vektör makinelerindeki sınırı ifade etmektedir.  $C$  ile belirtilen terim de hata maliyetidir. Bu denklemdeki

$(w, w)$  kısmını küçültmek fonksiyonu en optimize şekilde düzleştirecektir. Bu fonksiyon denklem 3.7’de tanımlanmaktadır.  $\Phi(x):x \rightarrow \Omega$  olmak üzere,

$$f(x, w, b) = (w, \Phi(x)) + b \quad (3.7)$$

Denklemi  $x \in X(\delta^d)$  i belirtmektedir. Burada  $\Omega$  yüksek boyutlu uzay gösterimidir. Ayrıca  $b \in \delta$  olmaktadır.

Geliştirilen bu sistemde, destek vektör regresyonunun birbiriyle alakasız verilere karşı esnekliğinin olmaması ve veri aralığının çok olmaması gerekleriyle, bütün veriler standart bir yapıya dönüştürülmüştür. Verilerin standartlaştırılması işlemi, verilerin birbirileri ile olan ilişkilerinin aralarındaki mesafelerin oransal olarak bozmadan indirgenmesini ifade etmektedir. Bu standartlaştırma işlemi için *scikit* kütüphanesinin *standarScaler* sınıfı kullanılmıştır. Bu sayede bütün veriler herhangi bir bozulmaya uğramadan denklem 3.8’de gösterilen fonksiyon yardımıyla standartlaştırılmıştır.

Bu denklemde  $Ox$  diye belirtilen terim standartlaştırılacak veri,  $M$  ise o andaki bulunulan kategorideki bütün verilerin ortalamasını ifade etmektedir.  $n$  değeri ise  $Ox$  in de içerisinde yer aldığı kategorideki bütün verilerin standart sapmasını ifade etmektedir (Scikit-Learn, 2013).

$$r = (Ox - M)/n \quad (3.8)$$

Destek vektör regresyonu ile problem çözümünde *scikit* kütüphanesinin *support vector machine* modülünün *svr (support vector regression)* sınıfı kullanılmıştır.

#### 3.2.4.2. Çoklu doğrusal regresyon

Karmaşık hesaplama ve ölçüm işlemlerine dayalı verilerin tahminlerini yapan Çoklu doğrusal regresyon İngilizce olarak *Multiple Lineer Regression* (MLR) olarak adlandırılmaktadır. Çoklu doğrusal regresyon, bağımlı olan değişkenin tahmini olarak hesaplanması için bağımsız olan değişkenlerin kullanılması,

bağımsız değişkenlerin bağımlı değişken üzerindeki etkilerini hesaplayarak sonuca ulaşmasını sağlayan bir regresyon çeşididir (Xu, 2003).

$$D = y_0 + y_1M_1 + y_2M_2 + \dots + y_nM_n + e \quad (3.9)$$

Yukarıdaki denklem 3.9'da  $y_0$  olarak geçen terim, doğrunun x eksenindeki kesim noktası,  $y_0, y_1, \dots, y_n$  regresyon katsayıları,  $M_0, M_1, \dots, M_n$  ise bağımsız değişkenlerdir. e terimi ise hata değeri olarak adlandırılmaktadır.

Çoklu doğrusal regresyon ile analiz yapılırken önemli olan bir nokta da bağımsız değişkenlerin seçimidir. Bağımsız değişken seçimi yapılırken korelasyon değeri yüksek olan değişkenlerin seçilmesi gerekmektedir, aksi halde tahminlerin güven oranı düşük olmaktadır.

Geliştirilen bu sistemde, birbiriyle alakasız verilere karşı esneklik olmama ihtimaline karşı ve veri aralığının çok olmaması gerekleriyle, bütün veriler standart bir yapıya dönüştürülmüştür. Verilerin standartlaştırılması işlemi, verilerin birbirileri ile olan ilişkilerinin aralarındaki mesafelerin oransal olarak bozmadan indirgenmesini ifade etmektedir. Bu standartlaştırma işlemi için *scikit* kütüphanesinin *standarScaler* sınıfı kullanılmıştır. Bu sayede bütün veriler herhangi bir bozulmaya uğramadan denklem 3.8'de gösterilen fonksiyon yardımıyla standartlaştırılmıştır. Destek vektör regresyonu ile aynı tip veriler elde edilerek tahminler yapılmıştır.

Çoklu doğrusal regresyon işlemleri *scikit* kütüphanesinin *linearRegression* sınıfı kullanılmıştır.

#### 3.2.4.3. Rastgele orman regresyonu

Leo Breiman, 2001 senesinde *Random Forest* yani Rastgele Orman (RF) algoritmasını geliştirmiştir (Breiman vd., 2001). Rastgele orman yöntemi kısaca akıllı ve esnek olan bir makine öğrenme yöntemidir. Rastgele orman yöntemi ile birlikte problemlere hem sınıflandırma hem de regresyon analizleri için

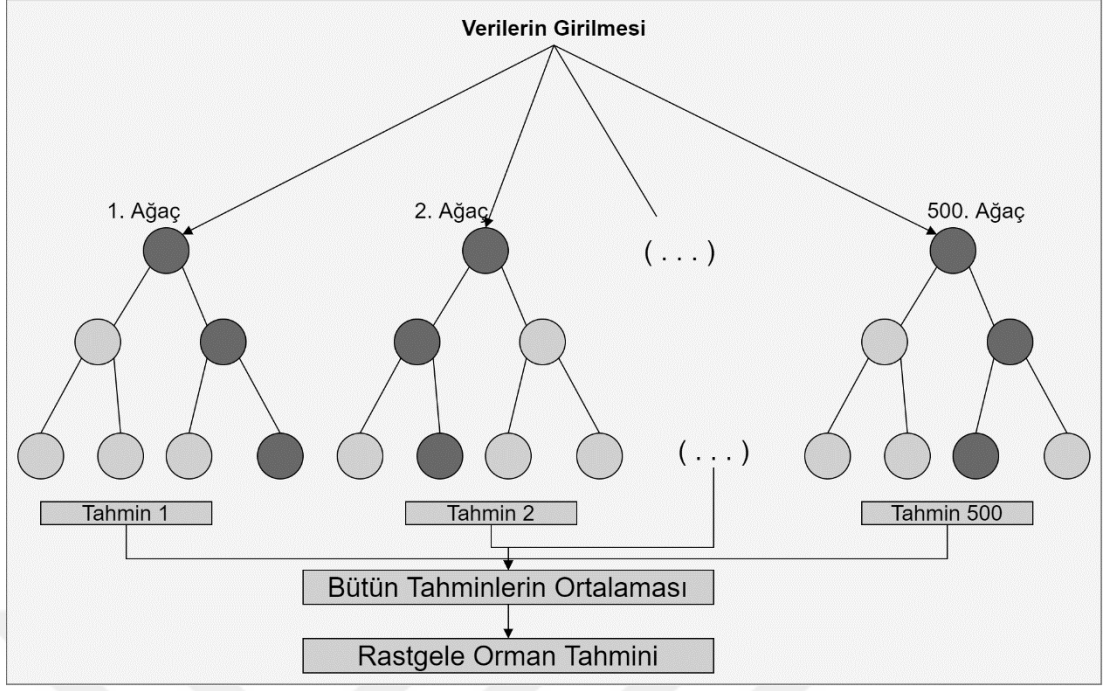


yaklaşımında bulunulabilmektedir. Bu yöntemde birden fazla modelin bulunduğu bir grup başarılı sonuçlar üretebilecek bir çözüm oluşturabilmek için birlikte çalışmaktadır (Sullivan, 2017).

Rastgele orman yöntemiyle regresyon karar ağaçları ve sınıflama (CART) ile birlikte karar ormanı denen topluluk oluşturulur. Rastgele orman ile birlikte ağaçlar oluşturulurken iki yapı kullanılır; Bunlar *bootstrap* örneklemeleri ve düğümlerde kullanılan, rassal olarak seçilen, en iyi bölünmeyi hedefleyen  $m$  adet tahminci terimi ile olmaktadır.  $m$  olarak belirtilen tahmincinin, toplam tahminci sayısına göre daha az olması gerekmektedir. Bunların sonunda oluşturulmuş ağaçlar bu haliyle bırakılır ve regresyon tahmin işlemleri için en az düğüm olacak şekilde bölünmeler ağaçlarda devam eder (Cutler vd., 2012).

Rastgele orman yönteminde birden fazla karar ağacı vardır ve Regresyon, kümeleme ve sınıflama için ağaçlar kurulabilmektedir. Bu yöntem kullanılırken kullanıcının elindeki probleme ait verilerdeki sınıf değişkeni süreklilik isteniyorsa regresyon kategorilik isteniyorsa ağaçları oluşturulabilmektedir (Akman, 2010).

Rastgele orman yöntemi doğru genellemeleri sunmada ve tahminlerde yakın sonuçlar vermede oldukça başarılıdır. Bunun sebebi topluluk yöntemlerinin optimize edilmiş özelliklerini ve rastgele örneklemeyi içermesindendir. Rastgele orman yönteminin genel yapısı Şekil 3.11’de gösterilmektedir.



Şekil 3.11. Rastgele orman yönetiminin genel yapısı

Rastgele Orman Regresyonu için işlemler *scikit* kütüphanesinin *RandomForestRegressor* sınıfı kullanılmıştır.

### 3.2.5. Yapay sinir ağları

Yapay Sinir Ağları (YSA), insanların gerçek hayattaki karşılaştıkları durumları örnek olarak alan, aldığı bu örnekleri çoğaltarak karşılaşmadığı durumları oluşturabilen, oluşturduğu durumlarla kendisine yeni yetenekler sağlayan bir bilgisayar sistemidir (Haykin, 1994). YSA hiç karşılaşmadığı durumları ya da eksik olarak öğrendiği durumları kendisi de yorumlayarak ortaya sonuç çıkarabilen bir makine öğrenmesi yöntemidir (Öztemel, 2003).

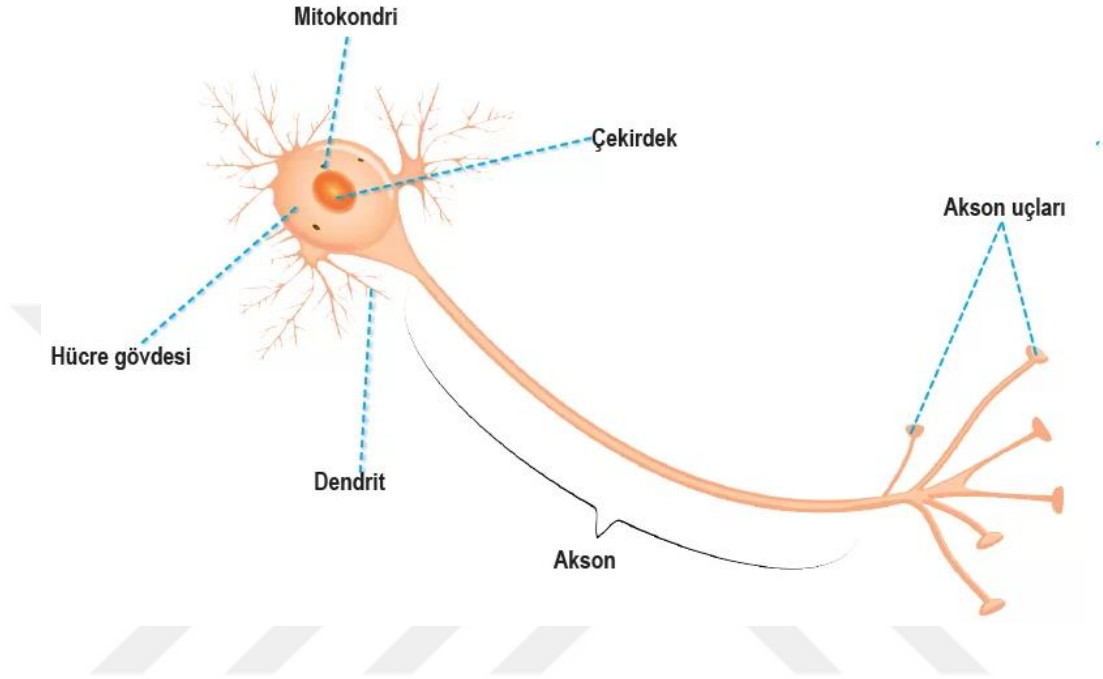
YSA kavramı bilim adamları tarafından insan beyninin ve tek hücreli olarak kategorize edilmiş olan canlıların sahip oldukları sinirsel sistemi araştırmaları sonucunda ilk defa ortaya çıkmıştır. YSA'nın tarihsel gelişimi şu şekildedir;

- 1940'lı yıllarda ilk defa temelleri atılmıştır.

- Hebb ve arkadaşları beyindeki nöronları modelleyip matematiksel olarak göstermişler ve ilk yapay sinir hücresini oluşturmuşlardır.
- 1949 Yılında Hebbian öğrenmesi denilen ve şu anda kullanılan öğrenme kurallarının birçoğunu oluşturan kurallar Donald Hebb tarafından geliştirilmiştir.
- 1954 yılında Adaptif tepki üretme terimi ortaya atılmıştır.
- 1958 yılında Adaptif Tepki Üretme terimi geliştirilmesi Roseblatt tarafından sağlanmıştır.
- Roseblatt'ın geliştirmiş olduğu algılayıcı model olarak geçen perseptron bu süreç için çok önemli bir nokta olarak görülmüş ve çok katmanlı algılayıcı temelleri oluşturulmuştur.
- 1962 yılında Roseblatt'ın geliştirmiş olduğu perseptron yapısına benzeyen, ama daha başarılı bir öğrenme yapısında olan ADALINE modelini Wifrof ve Hoff bilime kazandırmıştır.
- 1969 yılında algılayıcıların XOR problemini çözmemesi kanıt olarak gösterilerek, karmaşık olan mantıksal problemler için algılayıcıların işe yaramayacağı bilgisi Misnky ve Pappert tarafından öne sürülmüştür.
- 1972 yılında Çağrışimli Bellek konusunda temel olarak sayılabilecek çalışmalar Kohonen ve Anderson tarafından yapılmıştır.
- 1982 yılında tek katmanlı olan algılayıcıların çözümde başarısız olduğu XOR probleminde başarıya ulaşılan MLP hakkında çalışmayı Rumelhart ve ekip arkadaşları paylaşmıştır.
- 1990 ve 1991'li yıllarda, radyal tabanlı fonksiyonlardan gelen genel regresyon ağları, probabilistik yani tahmin edilemeyen ağlar Specht tarafından geliştirilmiştir (Tsoukalas ve Uhring, 1997).

Bu çalışmaların ışığında halen yayımlanan makaleler ve yapılan konferanslarda YSA geliştirilmeye devam etmektedir. Günümüzde teknolojinin de gelişmesiyle birlikte çalışmaların yapılabileceği bilgisayarların da donanımsal olarak daha iyi yerlere gelmesi, YSA üzerinde sürekli olarak yapılan değişim ve gelişimleri test etme ve gerçek hayata uyarlama imkânı da sağlamaktadır.

YSA kavramı insan beyninin sahip olduđu özelliklerden üretilmiş bir kavramdır, bu sebeple nöronların yapısının ve çalışma mantığının incelenmesi YSA'nın daha iyi anlaşılmasını sağlamaktadır. Şekil 3.12'de gösterildiği gibi sinir hücreleri dentrit, hücrenin gövdesi ve akson denilen 3 ana bölümden oluşmaktadır.



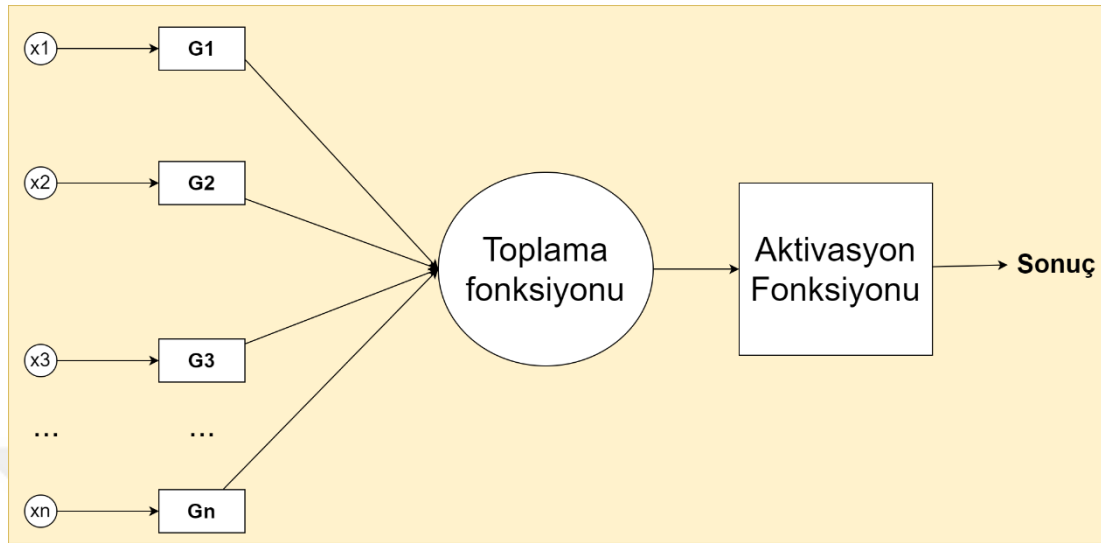
Şekil 3.12. Sinir ağlarının genel yapısı (Fikir, 2019)

Şekil 3.12'de görülen sinir hücresi, dentritler ile sinyalleri alıp hücre gövdesinde işleyen, işlenen bu sinyalleri aksonlar üzerinden başka sinir hücrelerine ileten bir yapıdadır.

Bir sinir hücresinin YSA ile benzerliği oldukça yüksektir. Biyolojik sinir hücresindeki dentritler, YSA'daki transfer fonksiyonlarına benzemektedir. Hücre gövdeleri hareket fonksiyonlarıyla, akson uçları çıkış katmanıyla sinapslar ise ağırlıklarla örtüşmektedir.

Biyolojik sistemler için öğrenme kavramı sinir hücreleri arasındaki bağlantılar sayesinde olmaktadır. İnsanlar dışarıdan gelen bilgileri kullanarak bu bağlantıları güncellemektedir. Öğrenme denilen kavram ise, gerçekleşmiş olayların sonuçlarla bağlantı kurulması sayesinde olmaktadır. Eğitim denilen kavram ise gerçekleşen olaylar üzerinden bağlantı ağırlıklarının belirlenmesiyle

en optimize sonuçlara ulaşılmasına kadar süren işlemlerdir (Aydın, 2016). YSA'nın temel yapısı Şekil 3.13'te gösterilmektedir.



Şekil 3.13. Yapay sinir ağlarının temel yapısı

Toplama Fonksiyonu, hücreye gelen net bilginin hesaplandığı kısımdır. Toplama fonksiyonu olarak Ağırlıklı Toplam, Maksimum, Minimum, Çoğunluk, Çarpım ya da Kümülatif gibi fonksiyonlar kullanılabilir. (Öztemel, 2003).

YSA'nın kullanacağı giriş parametreleri G1, G2, G3 ve Gm olarak gösterilmiştir. Bu giriş parametreleri sayesinde oluşturulacak sistemde eğitim örnekleri oluşacaktır.  $x_1$ ,  $x_2$ ,  $x_n$  olarak belirtilen parametreler ise ağırlıklardır. Oluşturulacak bu sistemde gelen bilgilerin etkisini belirlemektedir. Etkiler pozitif, negatif ya da sıfır değerlerinde olabilir, etkilerin pozitif ya da negatif durumu etkinin önem sırasını değiştirmemektedir. Hesaplama işlemleri için en çok tercih edilen ise Ağırlıklı Toplam olarak geçen fonksiyondur (Öztemel, 2003). Denklem 10'da gösterildiği gibi, m bu problemde ağırlıklı girdilerin sayısını, A her bir girdinin ağırlığını G ise girdiler ile bilgileri göstermektedir.

$$NET = \sum_{i=1}^m A_i \cdot G_i \quad (3.10)$$

Diğer toplama fonksiyonları Çizelge 3.1'de gösterilmektedir. Toplama fonksiyonu kullanılacağında her zaman girdi değeri önemli olmayabilir, bazen de girdi sayısı

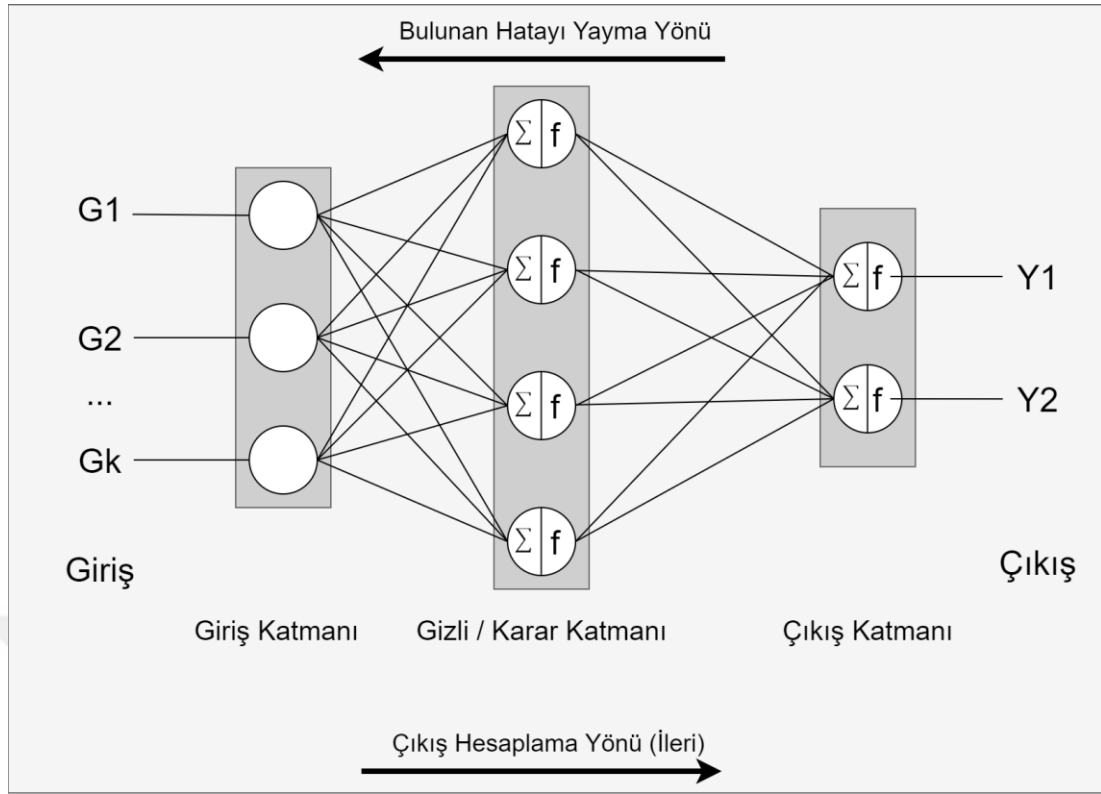
önemli olabilmektedir. Bu fonksiyonlardan hangisinin seçileceği geliştiriciye bağlıdır.

Çizelge 3.1. Yapay sinir ağları toplama fonksiyonları

| Fonksiyon Türü                                                  | Açıklama                                                                                                   |
|-----------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| Çarpım Fonksiyonu<br>$NET = \prod_{i=1}^m Ai. Gi$               | Bu fonksiyonda değerler ile ağırlık girdileri çarpılır. Bulunan bütün değerler de birbirleri ile çarpılır. |
| Maksimum Fonksiyonu<br>$NET = Max(Ai. Gi)$                      | Girdiler ile ağırlıklarının çarpımlarından en büyük olanı Net girdi olarak seçilir                         |
| Minimum Fonksiyonu<br>$NET = Min(Ai. Gi)$                       | Girdiler ile ağırlıklarının çarpımlarından en küçük olanı Net girdi olarak seçilir                         |
| Kümülatif Fonksiyon<br>$NET = NET_{ESKİ} + \sum_{i=1}^m Ai. Gi$ | Fonksiyon hesaplanırken ağırlık ve girdilerin çarpımlarının toplamı eski gelen bilgilere eklenir           |

### 3.2.5.1. Katmanlar

YSA makine öğrenme algoritması temelde 3 katmana sahiptir. Bunlar giriş katmanı, ara katman ve çıkış katmanıdır. Giriş katmanı geliştirilen YSA'ya dışarıdan gelen bilgileri alıp ara katmana (Gizli katman) gönderir. Ara katmanda ise giriş katmanından gelen ve işlemlere uğrayan bilgiler çıkış katmanına iletilir. Çıkış katmanı ise gelen bilgilere göre uygun sonuçları kullanıcıya verir. Çoklu çıkışlar da mevcuttur. Ara katman kullanıcının kararlarına göre birden fazla olabilmektedir. Şekil 3.14' te YSA'nın genel olarak şematik görüntüsü verilmiştir.



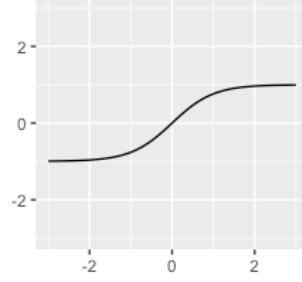
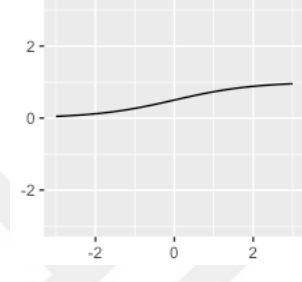
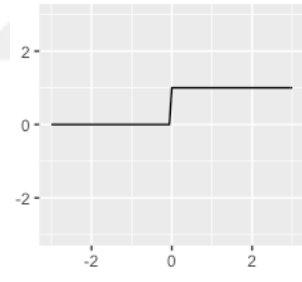
Şekil 3.14. Yapay sinir ağlarının katmanları

### 3.2.5.2. Aktivasyon fonksiyonları

Aktivasyon Fonksiyonu hücreye gelen net girdilerin işlenmesi sonucu hücrenin çıktı değerinin hesaplanması işlemidir. Aktivasyon fonksiyonu olarak birçok fonksiyon bulunmaktadır. Bunlardan Sigmoid, Lineer, Step, Sinüs ve Eşik değer fonksiyonu en yaygın kullanılan Aktivasyon fonksiyonlarıdır (Efe, 2004). YSA'da en sıklıkla kullanılan aktivasyon fonksiyonu ise Sigmoid fonksiyonudur.

Aktivasyon Fonksiyonları Sinir ağına gelen bilgilerin işleme sokulması ile çıkış biriminden ortaya çıkacak değerin hesaplanmasında kullanılır. Bu fonksiyonların birçok çeşidi bulunmaktadır. Bilinirliği ve türev alımı en kolay olan fonksiyon Sigmoid Fonksiyonudur. Çizelge 3.2'de en çok kullanılan aktivasyon fonksiyonları hakkında bilgi verilmektedir. İlk sütunda fonksiyonun adı, ikinci sütunda aktivasyon fonksiyonunun grafiği, üçüncü sütunda bu aktivasyon fonksiyonlarına ait formülleri son sütunda ise açıklamalar gösterilmektedir.

Çizelge 3.2. Yaygın olarak kullanılan aktivasyon fonksiyonları

|                                      |                                                                                     |                                                                                                                                                |                                                                                                                                                                        |
|--------------------------------------|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Tanjant Hiperbolik Fonksiyonu</b> |    | $F(NE\text{T}) = \frac{e^{NE\text{T}} - e^{-NE\text{T}}}{e^{NE\text{T}} + e^{-NE\text{T}}}$                                                    | Sigmoid fonksiyonuna benzemektedir. Tanjant Hiperbolik Fonksiyonu -1 ile 1 arasında değer üretir                                                                       |
| <b>Sigmoid Fonksiyonu</b>            |    | $F(NE\text{T}) = \frac{1}{1 + e^{-NE\text{T}}}$                                                                                                | Doğrusal Olmayan bir fonksiyon değildir. Sürekli ve Türevi alınabilir. En çok kullanılan Aktivasyon Fonksiyonu Çeşididir. Bu fonksiyon 0 ile 1 Aralığında değer üretir |
| <b>Step Fonksiyonu</b>               |  | $F(NE\text{T}) = \begin{cases} 1 & \text{Eğer } NE\text{T} > \text{Eşik Değer} \\ 0 & \text{Eğer } NE\text{T} < \text{Eşik Değer} \end{cases}$ | Gelen değer Belirlenen eşik değer altında ya da üstünde olmasına göre 0 ya da 1 üreten bir fonksiyondur.                                                               |

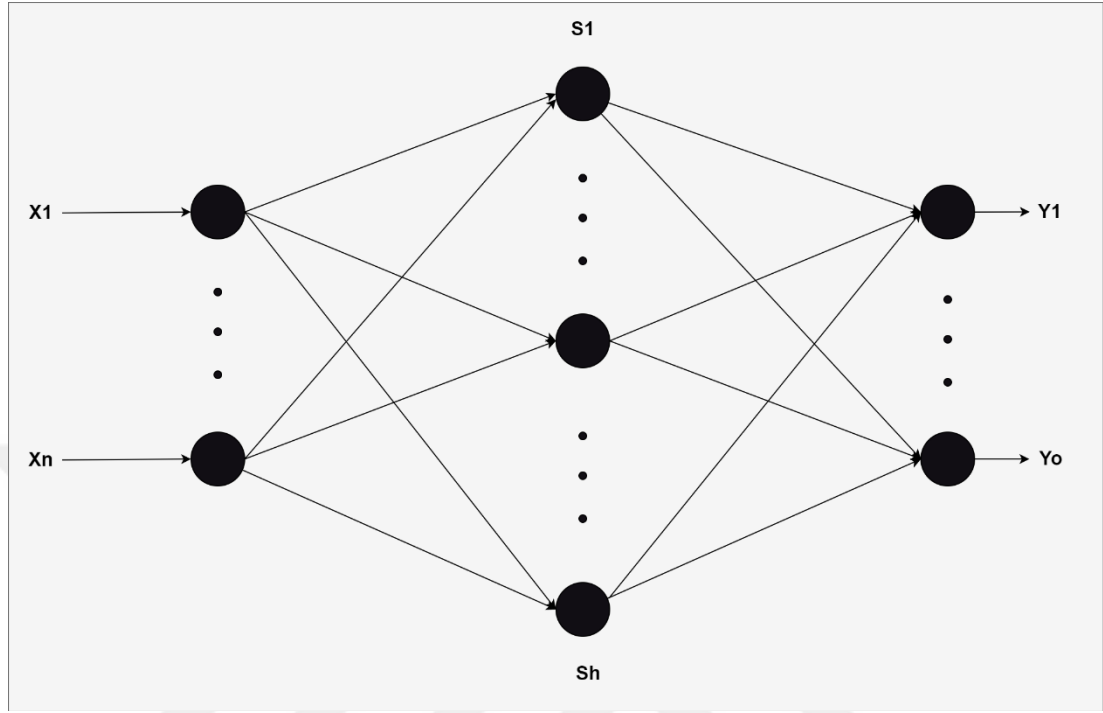
### 3.2.5.3. Yapay sinir ağları sınıflandırmaları

YSA yapılarına ve öğrenme algoritmalarına göre iki sınıfta incelenmektedir. Yapılarına göre yapay sinir ağları ileri beslemeli ve geri beslemeli olarak iki grupta, öğrenme algoritmalarına göre de denetimli, denetimsiz ve destekli öğrenme olarak üç grupta incelenmektedir.

**İleri Beslemeli Yapay Sinir Ağları:** Yapay sinir ağlarından en basiti olarak bilinmektedir. Tasarlanmış ilk yapay sinir ağıdır. Bu yapıda girdiler orta katmanlara da uğrayarak doğrudan çıkışa gidip sonuç verirler. Tek yönlü bir hareket söz konusu olmaktadır. İleri beslemeli yapay sinir ağları karmaşık ve

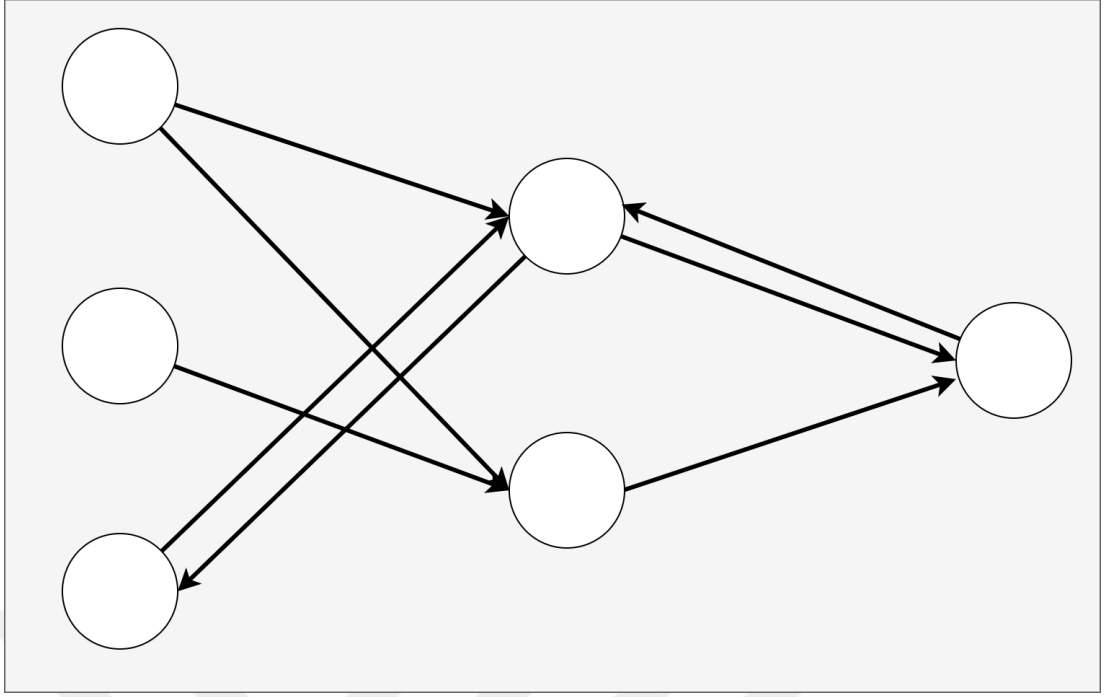


gerçek hayat problemlerini çözmekte yeterli olmamaktadır (Yadav vd., 2015). Şekil 3.15'te ileri beslemeli bir yapay sinir ağı gösterilmektedir.



Şekil 3.15. İleri beslemeli örnek (Zhang vd., 2007)

**Geri Beslemeli Yapay Sinir Ağları:** İleri Beslemeli Yapay Sinir Ağları modeline geri dönüşleri sağlayan kısımların eklenmesiyle oluşmaktadır. Şekil 3.16'da örnek bir geri beslemeli yapay sinir ağı gösterilmektedir. Ağırlık hesaplarına göre değişimlerin olabildiği bu yapılar karmaşık olan, doğrusal olmayan ve dinamik olan yapılar için daha uygun görülmektedir (Baddari vd., 2010; Aksu ve Çoban, 2014).



Şekil 3.16. Geri beslemeli örnek

**Denetimli Öğrenme:** Bu ağlar öğretmen denilen bir eğitici tarafından öğretilme işlemi ile birlikte eğitilmektedir. Öğrenme sürecinde giriş ve çıkışta oluşacak veriler ağı öğretilir. Bu yöntem Naive Bayes ve Destek Vektör sistemlerinde kullanılmaktadır (Dey, 2016).

**Denetimsiz Öğrenme:** Bu ağlarda öğretilme işlemi yoktur. Dolayısıyla herhangi bir veri ile giriş ve çıkışlara herhangi bir eğitim yapılmaz. Bu ağların bir avantajı da eğitim olmadığı için çok farklı sonuçlar üretebilmesi olarak sağlanmasıdır. Fakat sistem başarısı çok yüksek olmamaktadır. Temel analiz ve kümeleme işlemlerinde kullanılmaktadır (Dey, 2016).

**Destekli Öğrenme:** Bu ağlarda öğretici vardır fakat sonuca etki edecek bir eğitim yoktur. Öğretici sadece çıkışın istenildiği gibi sonuçlarda olabilmesi için hangi eylemler gerçekleştirilmeli gibi yönlendirmeleri yapar (Dey, 2016).

#### 3.2.5.4. Eğitim algoritmaları

**Levenberg-Marquardt Algoritması (LM):** LM ismi verilen bu algoritmayı geliştiren kişi Levenberg-Marquardt'tır. Ağların orta ölçekli olanları için yüksek hızda çalışabilir. Maksimum komşuluk fikrine dayanan en küçük kareler yöntemidir. Yinelemeli bir tekniktir (Lourakis ve Argyros, 2005).

**Bayesian Regularization Algoritması (BR):** BR algoritması türetilmiş bir algoritmadır. Bu türetileme işlemi LM eğitim algoritmasından yapılmıştır. En optimize ağ yapılarının bulunmasında fayda sağlamaktadır. Ağırlıklarla hata kareleri toplamalarının birlikte oldukları durumları en aza indirmeye çalışır. Ağ ezberleme (over fitting) olasılığının azaltılmasını sağlar. (Burden ve Winkler, 2008; Shahsavari vd., 2013).

**Geri yayılım (GY) algoritması:** Diğer algoritmalarla göre fark edilecek şekilde avantajlara sahip olan bu algoritma çok yaygın şekilde kullanılmaktadır. Anlaşılması ve matematiksel ispatlanabilirliğinin olmasından dolayı çok tercih edilen bir öğretim algoritmasıdır. Bu algoritmanın çalışma mantığı hataların çıkıştan giriş katmanına doğru azaltılmaya çalışılmasıdır. Çok katmanlı perseptronlar fazlaca kullanılmaktadır. Çalışma mantığı ise şu şekildedir: Öncelikle rastgele olacak şekilde başlangıç ağırlıkları seçilir, öğrenme başladıktan sonra yapılan işlemler sonucu çıkış değerleri hesaplanır ve değerlerdeki hata oranları yüksek ise ağırlıklar tekrar düzenlenip giriş katmanında tekrar tanıtılır. (Ganatra vd., 2011; Adewole vd., 2011).

#### 3.2.6. Adaptif ağ tabanlı bulanık mantık çıkarım sistemi

Yapay sinir ağlarının tercih sebebi olması, onun iki güçlü özelliğine dayanır. Bunlardan birisi haritalama yapabilmesi diğeri de eş zamanlı çalışma yürütebilmesidir. Fakat güçlü yanlarının yanında yapay sinir ağlarının zayıf kaldığı noktalar da bulunmaktadır. Çok katmanlı bir ağ yapısında sisteme ait bilgilerin ağırlıklarının özelliklerini açıklamak, ağdaki önceden bulunan bilgilerle birleştirmek oldukça zor, hatta imkansızdır. Bulanık mantıkta ise, sisteme ait

bilgiler insanların anlayacağı, kendisinin belirleyebileceği sözel yapıdaki terimler ile yapılmaktadır. Dolayısıyla sistem ve kullanıcı arasında başarılı bir etkileşim sağlanabilmektedir. Fakat bulanık mantıktaki en büyük eksiklik te öğrenme işlevinin olmamasıdır.

Bulanık mantıkta öğrenme becerisi yapay sinir ağları kullanarak mümkün olabilir. Yapay sinir ağlarındaki büyük eksikliklerden olan bilgilerin anlaşılmasındaki zorluk durumları bulanık mantıktaki başarılı olarak görülen kullanıcı sistem arasındaki sonuçların anlaşılmasında kullanılan sözel terimler ile birlikte kullanılarak rahat, anlaşılması kolay çıktılara dönüşebilir. Zaten *Sinirsel-Bulanık* sistem olarak tanımlanan yapıların asıl amacı iki farklı sistemin sahip olduğu faydalı noktaları bir araya getirerek daha iyi bir sistem ortaya çıkarmasıdır. Öğrenme becerisi, esneklik, sisteme ait çıktılarının kolay ifade edilebilmesi ve paralel şekilde çalışabilmek sinirsel – bulanık sistemlerin avantajlarıdır.

*Adaptive neuro fuzzy inference system (ANFIS)*, *Fuzzy Adaptive Learning Control Network (FALCON)*, *Neuro Fuzzy Classification (NEFCLASS)*, *Neuro Fuzzy Function Approximation (NEFPROX)* ve *Fuzzy Inference and Neural Network in Fuzzy Inference Software (FINEST)* geliştirilmiş ve son yıllarda oldukça popülerleşen sinirsel bulanık sistemlerden bazılarıdır. Adaptif ağ tabanlı bulanık mantık çıkarım sistemi *sugeno* tip sinirsel bulanık sistemi kullanır ve geri yayımlı öğrenme yöntemine sahiptir. Diğer bahsedilen *sinirsel-bulanık* sistemler ise *Mamdani* tip bulanık sistem ile kullanılmaktadır.

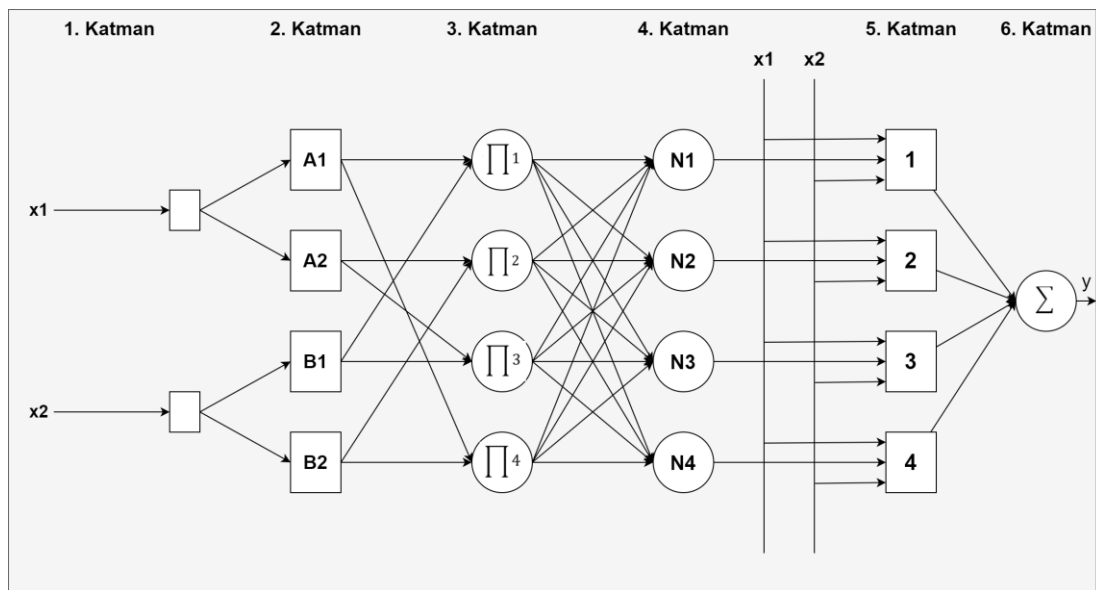
Sugeno tip sistemlerin, Mamdani tip sistemlere göre ortalama hatanın karekökü (*RMSE*) kriteri göz önüne alındığında daha yüksek başarıda sonuçlara ulaşıldığı görülmüş, fakat hesaplama ve uygulama sürecinde Sugeno tip sistemlerde daha fazla karmaşıklık gerektirmiştir (Jang, 1993). Bahsedilen her bir sinirsel-bulanık sistemin bir diğerine karşı performans, anlamlılık, zaman gibi kriterlerde avantaj ve dezavantajları bulunmaktadır.

Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi 1993 yılında Jang tarafından geliştirilmiştir. Şu ana kadar kabul görmüş en iyi sinirsel-bulanık sistemdir. Takagi – Sugeno Bulanık modelini kullanmaktadır. Jang'ın geliştirmiş olduğu ANFIS sistemi, öğrenme becerisini yapay sinir ağlarından, uygulama tarafını melez öğrenme ve geri yayımlı öğrenme algoritmasından alırken, Karar verebilme ve uzman görüşü becerisini de Sugeno tipi bulanık mantık çıkarım sisteminden almaktadır. Diğer bir tanım da öğrenme becerisini yapay sinir ağlarından, karar verme kısmını bulanık mantıktan alan sisteme Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi denmektedir.

ANFIS sistemine ait değişkenleri ayarlamak ve kuralları belirlemek için Yapay sinir ağları kullanılmaktadır. Yapay sinir ağlarında ise yapısal ve değişken ayarlama aşamaları mevcuttur.

Yapısal ayarlama aşaması kuralların ve giriş çıkış değişkenlerinin oluşturulmasını içermektedir.

Değişken ayarlama ile birlikte ise üyelik fonksiyon ağırlıkları ve kural fonksiyon hesaplamalarını kapsamaktadır. İki girişe ve tek çıkışa sahip olan, 4 kurallı Takagi – Sugeno tipi ANFIS yapısı Şekil 3.17'de gösterilmektedir.



Şekil 3.17. Adaptif ağ tabanlı bulanık mantık çıkarım sistemi (Jang, 1993)

İlk katmandaki düğümlere gelen bilgiler 2. katmana aktarılırlar. İkinci katmanda,  $x_1$  ve  $x_2$  terimleri üyelik fonksiyonu yardımıyla, üyelik derece bilgisini alıp diğer katmana geçerler. 3. katman kural katmanıdır. Girdi olarak gelen sinyaller çarpılır ve düğümden çıkar. 4. katman yani normalizasyon katmanına gelen sinyaller giriş değişkenleri olarak alınır ve normalleştirilmiş ateşleme seviyesi hesaplanır. 5. katmana gelindiğinde arındırma işlemleri yapılmaktadır. Her kurala ait ağırlık dereceleri hesaplanır. Son katmanda ise sonuç hesaplanarak çıktı verilir. Aşağıda katmanlara ait detaylar gösterilmektedir.

### 1. Katman

Bu katman girdi katmanıdır. Katmandaki her bir giriş sinyali diğer katmana aktarılmaktadır. Çıkış bilgileri  $i$  düğümleri için aşağıda gösterilmektedir.

$$O_{1,i} = \mu A_i(x), \quad i = 1,2 \quad (3.11)$$

$$O_{1,i} = \mu B_{i-2}(x), \quad i = 3,4 \quad (3.12)$$

### 2. Katman

Bu katman bulanıklaştırma katmanıdır. Katmandaki her bir düğüm bulanık kümelerle ayrılmaktadır. Bu katmandaki düğümlerin çıkışı, giriş değerlerine ve aynı zamanda kullanılmakta olan üyelik fonksiyonlarına bağlı olarak üyelik derecelerini almışlardır. Bu katmandaki üyelik dereceleri  $\mu A_i$  ve  $\mu B_i$  olarak gösterilmektedir. Üyelik fonksiyonları çan eğrisi şeklinde kullanılırsa  $\mu A_i$  aşağıdaki eşitlikteki gibi gösterilir.

$$\mu A_i = \frac{1}{1 + \left[ \left( \frac{x - c_i}{a_i} \right)^2 \right]^{b_i}} \quad (3.13)$$

Bu denklemdeki  $a_i$ ,  $b_i$ ,  $c_i$  terimleri üyelik fonksiyonuna ait sigma, eğim ve merkez bilgileridir. Farklı üyelik fonksiyonları bu katmanda kullanılabilmektedir (Yücel ve Güneri, 2010).

### 3. Katman

Bu katman kural katmanı olarak bilinmektedir. Katmanda bulunan düğümler, Sugeno bulanık mantık çıkarımı sistemine göre oluşturulmuş olan kural ve sayı bilgilerini göstermektedir. Bu katman üzerindeki hücrelerin sayısı ile kuralların sayısı aynıdır. Kuralların düğümlerinin çıktısı 2. katmandan gelen üyelik derecelerini çarpımı ile hesaplanmaktadır.  $O_{3i}$  Çıktıları,  $\mu_i$  ise ağırlık derecelerini göstermektedir.

$$O_{3i} = \mu_i = \mu A_i(x) \cdot \mu B_i(y) \quad (3.14)$$

### 4. Katman

Bu katman normalizasyon katmanı olarak bilinmektedir. Katmanda bulunan düğümler, kural katmanı yani 3. Katmandan gelen düğümleri giriş değerleri olarak kabul eder ve bütün kuralların normalleştirilmiş katsayı bilgisini hesaplar. Önceki katmandaki gibi, buradaki hücrelerde sabittir ve ağırlık dereceleri önceki katmandan alınan derecelerdir. Normalleştirme işlemi Denklem 3.15'teki gibi yapılmaktadır.

$$O_{4i} = \frac{\mu_i}{\sum_{i=1}^n \mu_i} = \bar{\mu}_i, \quad (i = 1, n) \quad (3.15)$$

### 5. Katman

Bu katman arındırma katmanı olarak adlandırılmaktadır. Bu katmandaki normalizasyon katmanından gelen bütün düğümlerin çıkış verileriyle arındırma katmanındaki giriş değerlerinin arasından ilişkiler vardır. Arındırma katmanında düğümlerin kurallarının ağırlıklı sonuçlarının hesaplanması yapılmaktadır. Bu katmandaki düğümlerin çıktı değerlerinin hesaplanması denklem 3.16'daki gibi yapılmaktadır.

$$O_{5i} = \bar{\mu}_i [p_i x_1 + q_i x_2 + r_i], \quad (i = 1, n) \quad (3.16)$$

Denklem 3.16'daki  $(p_i, q_i, r_i)$  olarak tanımlanan değişkenler, kuralların sonuç parametrelerini temsil etmektedir.

## 6. Katman

Bu katman toplama katmanı olarak adlandırılmaktadır. Bu katmanda düğüm noktası bir tanedir.

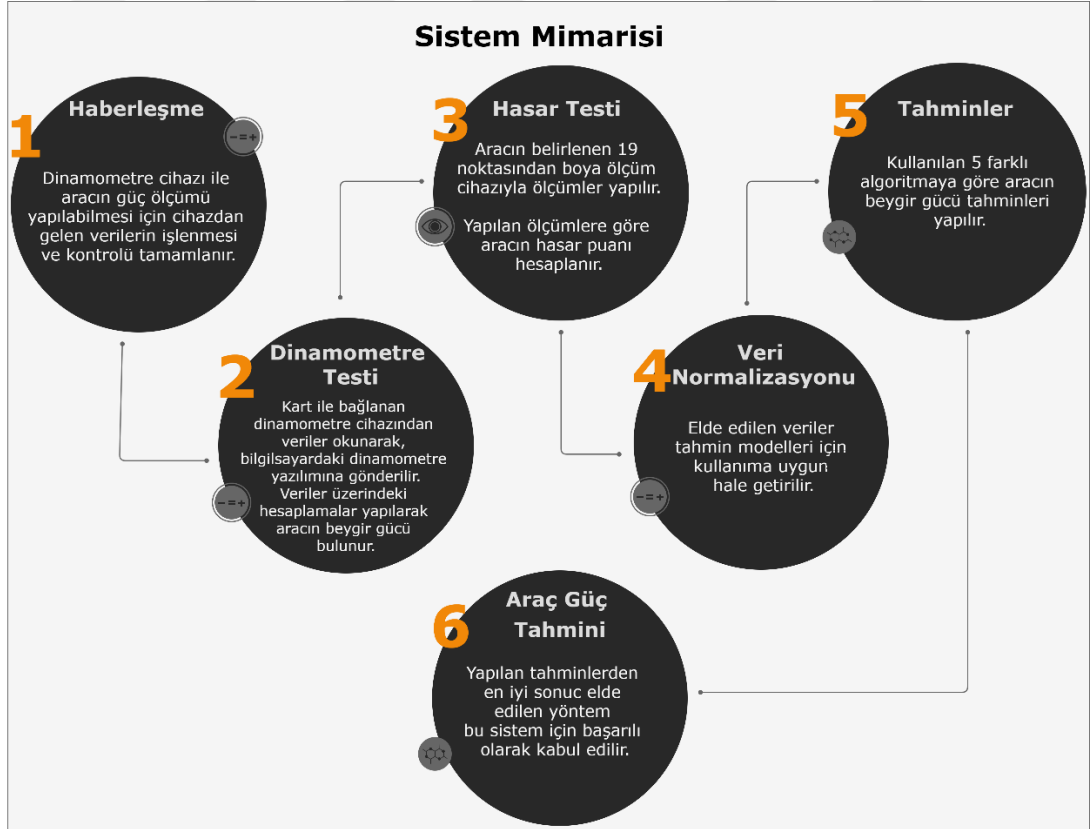
$\Sigma$  sembolü kullanılarak bu düğüm noktası gösterilmektedir. Toplama katmanında, arındırma katmanından aktarılan bütün düğüm noktalarının çıkışa ait değerleri toplanılır ve ANFIS'e ait sistemin gerçek değeri bulunmuş olur. Sisteme ait sonuç yani çıkış değerini gösteren  $y$  değişkeninin hesaplaması Denklem 3.17'de gösterilmektedir (Jang, 1993).

$$O_{6i} = \sum_{i=1}^n \bar{\mu}_i [p_i x_1 + q_i x_2 + r_i] \quad (3.17)$$



#### 4. ARAŞTIRMA BULGULARI

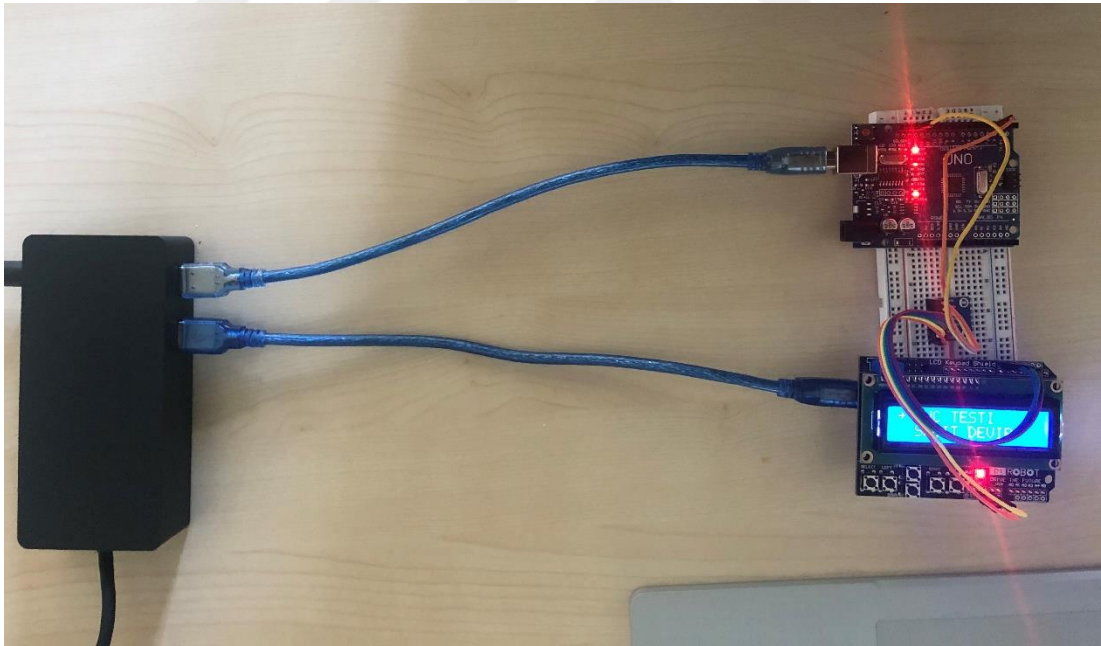
Bu tez çalışmasında uygulanmış olan yöntemler adım adım aşağıdaki başlıklar halinde anlatılmıştır. Şekil 4.1’de geliştirilmiş olan bütün sistemin mimarisi gösterilmektedir. Öncelikle şasi dinamometresi cihazıyla haberleşme ve cihazların doğru çalıştığının testlerinin yapılması işlemleri gerçekleştirilecektir. Geliştirilen bu haberleşme yapısı kullanılarak, şasi dinamometresi cihazından okunan bilgiler yardımıyla dinamometre ölçüm yazılımı yapılacaktır. Çalışmada tahminlerin daha başarılı olması için test yapılan araçlar ile ilgili daha fazla bilgi de toplayabilmek, aynı zamanda araç ile ilgili daha fazla bilgiye sahip olabilmek için hasar testleri yapılacaktır. Geliştirilen yazılımlar bittikten sonraki adımda toplanan verilerin normalizasyonu yapılacaktır. Veriler tahmin modellerinde kullanılacak hale geldikten sonra, belirlenen tahmin modelleri kullanılarak araçların beygir gücü bilgileri tahmin edilecektir. Son olarak da tahmin için kullanılan modeller karşılaştırılarak en başarılı sonucun hangi algoritma ile sağlanmış olduğu bulunacaktır.



Şekil 4.1. Geliştirilen sistemin mimarisi.

#### 4.1. Dinamometre Denetleme Sisteminin Yapılması

Çalışmada kullanılacak olan atalet tipli şasi dinamometresinin tamburlarındaki tırnakların dönüşünden sensörlerle bilgi alınmaktadır. Kullanılan sensör *Hall Effect* sensörüdür. Sensör her tırnağa denk geldiğinde okuma yapar ve dolayısıyla bir kare dalga oluşur. Bu tırnakların sayısı hesaplamalarda kullanılacak olan darbe sayısıdır. Darbe sayısı fazla olursa okuma da bir o kadar netleşmektedir. Bu tırnakların kendi kalınlıkları ve aralarındaki mesafe çok hassas bir şekilde birbirine eşit olmalıdır. Cihaz merdanenin devir değerini ölçmekte, dişli oranını hesaplayarak ekranda motor devrini gösterebilmektedir. Devir değeri diskin her turunu ne kadar zamanda tamamladığı ölçülerek bulunmaktadır. Birimi devir/dakikadır. Bu bilgiler geliştirilen Arduino kart üzerine C programlama dili ile geliştirilmiş olan sistem ile sağlanmaktadır. Şekil 4.2’de şasi dinamometresinin bağlı olduğu geliştirilen denetleyici sistem gösterilmektedir.



Şekil 4.2. Veri okuma ve veri denetleme kartları.

Hazırlanan bu kart devir değerlerini ölçer ve değerlendirir. Fren sistemi, hava valfi, soğutma fanı gibi çevre elemanlarının da kontrolünü sağlar. Ayrıca analog sensörlerden (ortam sıcaklığı, ortam hava basıncı, egzoz gaz sıcaklığı, yağ sıcaklığı, manifold basıncı, lambda sensörü) gelen veriyi işler. Dinamometre

ölçümlerini gösterecek masaüstü bilgisayar yazılımı ile haberleşerek verileri aktarır, kontrol komutlarını işler. Denetleyici kart üzerinden gönderilecek değerler verilerin filtrelenmesi ve düzensiz verilerin yumuşatılması adına basit hareketli ortalama yöntemiyle işlenmiştir. Basit hareketli ortalama yönteminin hesaplanmasına ait kod parçacığı Şekil 4.3'te gösterilmektedir.

```
1 Loop(firstcontroller=0;j<datasize-period+1;firstcontroller++)
2 {
3     Loop(i=secondcontroller;i<=secondcontroller+period-1;i++)
4     {
5         totalValue = totalValue + array1[i];
6     }
7     array2[j] = totalValue/period;
8     printf("period result : "array2[firstcontroller]);
9     totalValue = 0;
10    secondcontroller = secondcontroller+1;
11 }
```

Şekil 4.3. Basit hareketli ortalama algoritması.

Çalışmada elektronik kontrol kartının hazırlanmasından sonra, ölçülen değerlerin doğru olup olmadığını kontrol edebilmek için dinamometre cihazının taklidini yapabilen, tam değerler gönderip, dinamometre yazılımının doğru değerleri okuyup, hatasız filtrelemeler uyguladığını kanıtlamak için dalga üretimi yapabilecek AD9833 sinyal jeneratör modülü kullanılmıştır. Bu sinyal jeneratör modülü ile gönderilecek kare dalganın frekansını hesaplamak için, kullanılan dinamometredeki iki parametresinin bilinmesi gerekir. Bunlar darbe sayısı ve dişli oranı bilgisidir. Darbe sayısı merdanenin her bir dönüşündeki görmesi gereken sayıdır. Kullanılan merdanede darbe sayısı 8'dir. Ölçüm esnasında ölçülen devir bilgisi merdane devridir. Fakat ölçümler sonucu elde edilen tork ve beygir gücü bilgileri motor devir ile hesaplanmaktadır. Bu hesabı yapabilmek için test başlamadan önce merdane üzerinde önceden belirlenen devir değerine araç getirilir ve eşitleme yapılarak gelen değerler orantılı bir şekilde değiştirilir. Değişimler sonrası bulunan devir değerine göre kare dalga frekansı üretilir ve doğrulamalar yapılır. Doğrulama işleminde sabit devir ve güç testi fonksiyonları yapılmıştır. Sabit devir fonksiyonunda istenilen bir devir değeri için hesaplanmış frekans değeri sabit olarak dinamometreye gönderilmektedir. Dinamometreden masaüstü yazılımına frekanstan devir değerine dönüşmüş olan değerler

karşılaştırılarak cihazların verileri doğru okuyup okumadığı, dişli oranı hesaplarında yanlışlık olup olmadığı gözlemlenmiştir.

Güç testi fonksiyonunda ise dinamometre cihazı üzerinde yapılan bir testin simule edilmesi sağlanmıştır. Fonksiyon çalıştırıldığında lineer ya da logaritmik bir şekilde önceden tork ve beygir gücü bilgileri bilinen bir testin, dinamometre cihazından da aynı sonuçları göstermesi hedeflenmiştir. Karşılaştırma sayesinde hesaplama ve filtreleme algoritmalarının doğru çalıştığı gözlemlenmiştir. Şekil 4.4'te kalibrasyon işlemleri için hazırlanan devrenin ekran görüntüleri gösterilmektedir.



Şekil 4.4. Kalibrasyon işlemlerindeki devrenin ekran görüntüleri.

Değerlerin doğruluğunun ölçümlerinin yapıldığı mikrodenetleyici sisteminin frekans üretimine ait kod parçacığı Şekil 4.5'teki algorithmada gösterilmiştir.

```
double fixedHz = startRPM * (double)nPulse / (gearRatio * 60);
gen.Begin();
gen.ApplySignal(SQUARE_WAVE,REG0, fixedHz);
gen.EnableOutput(true);
lcd.home();
lcd.print(gen.GetActualProgrammedFrequency (REG0)*gearRatio*60/(double)nPulse, 0);
lcd.print(" d/d");
```

Şekil 4.5. Kare dalga üretim algoritması.

## 4.2. Dinamometre Yazılım Sisteminin Yapılması

Dinamometre yazılımı masaüstü uygulama olarak geliştirilmiştir. Geliştirilen bu uygulama C# dilinde Visual Studio 2017 editöründe hazırlanmıştır. Editörün ve programlama dilinin sağlamış olduğu eklentiler ve gelişmiş özellikler sayesinde kullanıcı dostu arayüze sahip raporlama yapabilen bir yazılım geliştirilmiştir.

Şasi dinamometresinden denetleme kartı yardımıyla gelen veriler USB bağlantısı ile birlikte hazırlanan dinamometre masaüstü yazılımına aktarılmıştır. Aktarılan veriler dinamometre testi süresince saniyede 20 adet veri gelecek şekilde ayarlanmıştır. Gelen veriler tamburların dönme bilgisinin yanı sıra çevresel bilgileri de içermektedir. Tek seferde gelen bilgiler Şekil 4.6'da gösterilmektedir.

```
gelenVeri = donusRPM+donuskmH+ortamSicakligi+ortamBasinci+nemOrani+duzeltmeKriteri+duzeltmeKatsayisi;
```

Şekil 4.6. Dinamometre cihazından sürekli gelen bilgiler.

Dinamometre cihazından gelen bilgiler sürekli işlenerek, aracın anlık RPM bilgisi ve aynı zamanda sensörlerden gelen bilgiler ekrandan gösterilmektedir. Test başlamadan önce araç ile yazılımın doğru senkrona çalışabilmesi için eşitlemeler yapılmaktadır. Eşitlemeler sonrasında ise test başlatılmaktadır. Şekil 4.7'de dinamometre yazılımının Ana Sayfası gösterilmektedir.

Dosya Hakkımızda

**Başlangıç Devri**  
**1601**

**ORAN AYARI**

**BAŞLAT**

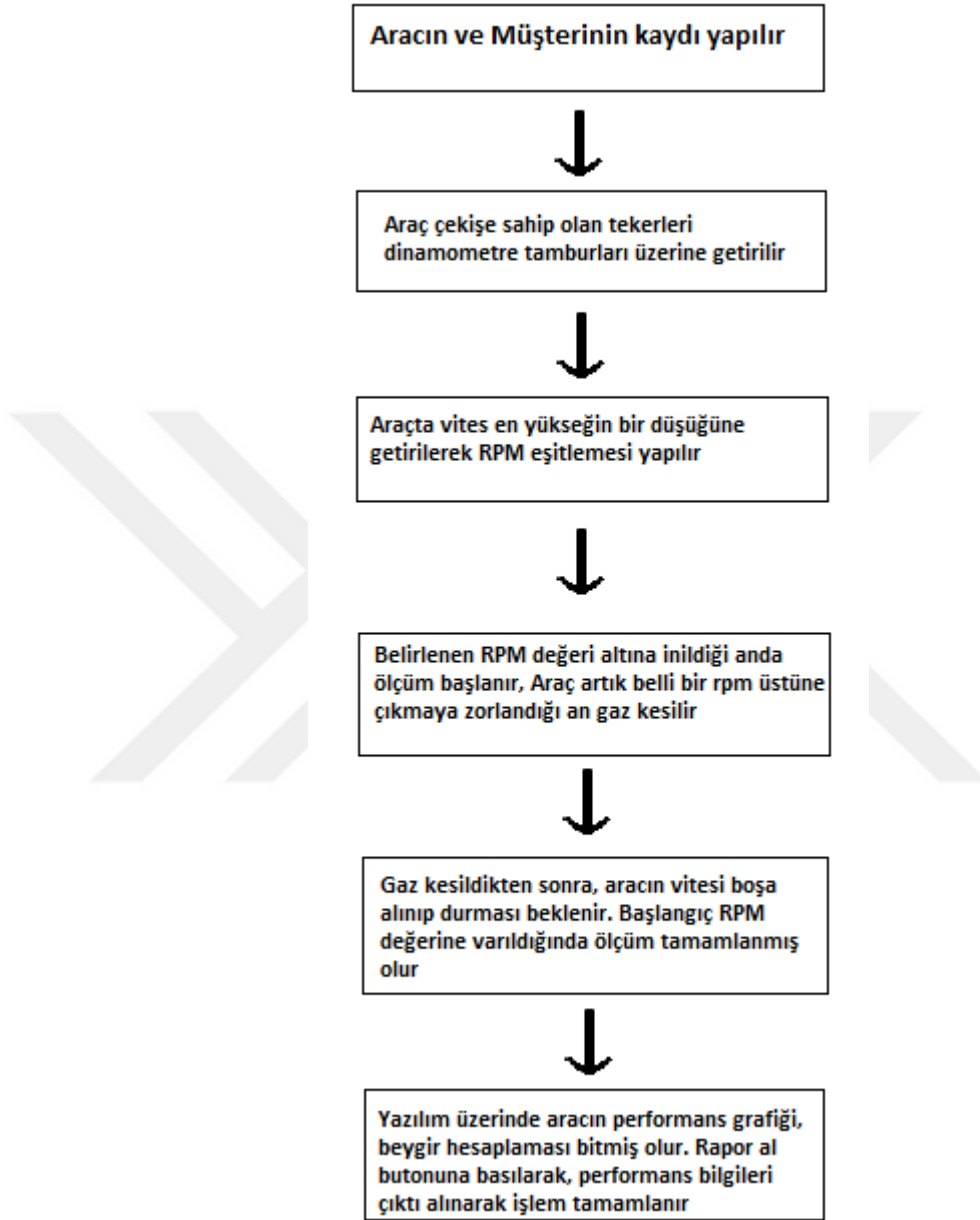
**ÇIKTI AL**

| DEVİR               | HIZLANMA            |                     |
|---------------------|---------------------|---------------------|
| 0                   | 0                   |                     |
| TORK                | BEYGİR GÜCÜ         |                     |
| 0                   | 0                   |                     |
| En Yüksek Teker Güç | En Yüksek Motor Güç | En Yüksek Kayıp Güç |
| HP                  | HP                  | HP                  |
| -                   | -                   | -                   |

Şekil 4.7. Dinamometre yazılımının ana sayfası.

Ana Sayfa formunun üst menülerinde, testlerle ilgili ön ayarlama yapılabilmektedir. Formun üst kısmında ise teste başlamadan önce, aracın test ölçümlerinin hesaplamalarının ne zaman hangi devirde başlayacağı, hangi devirde dişlilerin oranlarıyla yazılımdaki değerlerin eşitleneceği bilgileri gösterilmektedir. Bu değerler istenirse kullanıcı tarafından girilip ayarlamaları kontrol edebilmektedir. Dişli oranı ayarla butonuna basılarak açılan oran hesaplama formundan ise tamburların dönüş hızıyla testteki aracın ekrandaki görünen RPM değerleri arasındaki farklılıklar giderilmektedir. Sağ kısımda bulunan sarı panellerde ise test süresince gelen değerlerin ve sensör verilerinin görüntüleme kısmı bulunmaktadır. Bu bilgiler test süresince anlık olarak değişmektedir. Çalıştıktan sonra ortaya gelen göstergelerde ise, kullanıcının gerçek araç ekranında gördüğü değerler aynı anda burada da gösterilmiştir. Bu aynı zamanda kullanıcının ölçümü daha rahat takip edebilmesini sağlamıştır.

Yazılımın ilk açıldığı andan itibaren beygir güç ölçüm sonucunun alınmasına kadar geçen sürecin akış diyagramı Şekil 4.8’de gösterilmektedir.



Şekil 4.8. Dinamometre yazılımı kullanım süreci akış diyagramı.

Program çalıştırıldığında, öncelikle test yapılmak istenen aracın ve aracı getiren müşterinin bilgileri kayıt edilmektedir. Kayıt işlemi, sonuçların analiz işlemleri ve raporlama sürecinde kullanılmaktadır. Araç kayıt formu Şekil 4.9’da gösterilmektedir. Kayıt sayfasında, aracın diğer araçlardan da ayırt edilmesi için eşsiz değerler olarak şasi numarası ve plaka bilgisi alınmaktadır. Bu bilgilere ek olarak aracın marka, model, yıl, renk, km, kullanıldığı şehir bilgileri de



alınmaktadır. Sonrasında, analizler için gerekli olacağı için; aracı getiren kişinin cinsiyet, ad, soyad, telefon bilgileri de kaydedilmektedir. Şu an için kullanılmamış olsa da sonrasındaki geliştirmeler için ihtiyaç duyulabilir diye bu testleri yapan firmanın genel bilgileri de kayıt altına alınmaktadır. Son olarak da test yapılacak araçta kullanılacak RPM başlangıç devir bilgisi ve Dişli oranı RPM hesap devir bilgileri alınmaktadır. Bu işlem genelde çoğu otomobil sınıfındaki araç için aynı uygulanmaktadır. Çalışma süresince test yapılan bütün araçlar otomobil kategorisinde olduğu için bu parametreler hep aynı olarak kullanılmıştır. Kayıt formunun en altında ise, geçmişte yapılmış olan bütün araçların bilgileri tabloda listelenmiş şekildedir. Herhangi bir kontrol etme durumunda, kullanıcı tablodan bir aracı tıklayarak, aracın geçmiş bilgilerine ulaşabilir.

Şasi Numarası :

Plaka :

Marka :

Model :

Araç Yılı :

Güç (BG) :

Yakıt Cinsi :

Başlangıç Devri

Dişli Oranı Hesap Devri

Firma Adı :

Telefon :

Firma Şehir :

**Geçmiş Testler**

Temizle Kaydet

Şekil 4.9. Dinamometre yazılımı kayıt formu.

Aracın kaydı tamamlandıktan sonra, araç dinamometre cihazı üzerindeki tamburların üzerine getirilerek durdurulur. Sonrasında araç alt kısımlardan şase ile bağlanarak, test esnasında kaymaların olması engellenir. Araç bağlandıktan sonra motor tekrar çalıştırılarak test başlamasına hazır hale getirilir. Araç en üst vitesin bir alt vitesine getirilerek başlangıç RPM değerine getirtilerek dişli oranı ayarla butonuna basılır ve dinamometreden giden RPM bilgisi ile aracın gerçek RPM bilgisinin gösterildi form açılır. Bu formda aracın gerçek RPM bilgisi önceden belirlenmiş RPM değerine geldiğinde birbirlerine senkron edilir.



Böylelikle ölçümlerin doğruluğu sağlanır. Eşitlik sağlandığında butona basılarak işleme devam edilir. Dişli oranı eşitleme formu Şekil 4.10'da gösterilmektedir.



|                                        |                                                                                                   |
|----------------------------------------|---------------------------------------------------------------------------------------------------|
| <b>Belirlenen Değer</b><br><b>2000</b> |  <b>EŞİTLE</b> |
| <b>Anlık Değer</b><br><b>-</b>         |  <b>İPTAL</b>  |

Şekil 4.10. Dişli oranı eşitleme formu.

Dinamometre yazılımında test ile ilgili ayarlamaların yapılabilmesi için Şekil 4.11'de gösterilen form yapılmıştır. Bu form ile test ile ilgili bütün ayarlamalar detaylı olarak yapılabilmekte ve kullanılan dinamometrenin özelliklerine göre ölçümlerin doğruluğu sağlanabilmektedir.

|                                                                                     |                                |
|-------------------------------------------------------------------------------------|--------------------------------|
| <b>Gear Ratio :</b>                                                                 | <input type="text"/>           |
| <b>Smoothing Factor :</b>                                                           | <input type="text"/>           |
| <b>Drum Inertia :</b>                                                               | <input type="text"/>           |
| <b>Pulses Per Dyno Rev :</b>                                                        | <input type="text"/>           |
| <b>Roller Diameter</b>                                                              | <input type="text"/>           |
|                                                                                     |                                |
| <b>Sensor Aux 1</b>                                                                 | <input type="text"/>           |
| <b>Sensor Aux 2</b>                                                                 | <input type="text"/>           |
| <b>Sensor Aux 3</b>                                                                 | <input type="text"/>           |
|                                                                                     |                                |
| <b>Sensor Com 1</b>                                                                 | <input type="text"/>           |
| <b>Sensor Com 2</b>                                                                 | <input type="text"/>           |
| <b>Sensor Com 3</b>                                                                 | <input type="text"/>           |
|                                                                                     |                                |
| <b>Başlangıç Devri</b>                                                              | <input type="text"/>           |
| <b>Dişli Oran Hesap Devri</b>                                                       | <input type="text"/>           |
| <b>Correction</b>                                                                   | <input type="text" value="v"/> |
|                                                                                     |                                |
|  | <b>Kaydet</b>                  |

Şekil 4.11. Dinamometre yazılımı ayarlar formu.

Yazılım tamamlanıp atalet tipli şasi dinamometresine bağlanarak başarılı bir şekilde denenmiştir. Bütün testler doğru sonuçları üretmiş ve dinamometre haberleşme ve raporlama yazılımı tamamlanmıştır.

#### 4.3. Kaporta Hasar Yazılımının Yapılması

Çalışmada araç performans testlerinin analizinin yapılması için araç ve kullanıcı bilgilerinin dışında, aracın hasar durumu da parametre olarak kullanılmıştır. Aracın hasarlı ya da kazalı olması beygir gücü performansında etkili olacağı düşünülmektedir. Bunun için öncelikle CEM marka hem kablolu hem de kablosuz haberleşme yapabilen, demir ve demir olmayan ürünlerin üzerindeki boya miktarını ölçebilen kalınlık ölçüm cihazı kullanılmıştır. Şekil 4.12’de kullanılan cihaz gösterilmektedir.



Şekil 4.12. Boya kalınlığı ölçüm cihazı.

Çalışmada cihaz ile yapılan ölçümlerin girdi olarak girilip, raporlama ve veritabanına kayıt işlemlerinin yapılması için yine masaüstü bir hasar uygulaması geliştirilmiştir. Geliştirilen bu uygulamada araçlar sabit olarak 19 parçadan bilgi alınmıştır. Bu parçalar; Sol ön çamurluk, sol ön kapı, sol arka kapı, sol arka çamurluk, kaput, tavan, bagaj, sağ ön çamurluk, sağ ön kapı, sağ arka kapı, sağ arka çamurluktur, sol ön direk, sol orta direk, sol arka direk, sağ ön direk, sağ

orta direk, sağ arka direk, sol marşpiyel ve sağ marşpiyeldir. Belirlenen bu parçalar hasarlı ya da değil olarak işaretlenmiştir. İşaretlemeler sonucunda her araç, 0 ile 19 arasında hasar durumuna göre puanlanmıştır. Puanlamada hasarsız araçlar 0, maksimum hasarlı araçlar ise 19 puan almıştır. Hasarlar belirlenen bir sistem ile araç üzerinde boya ölçüm cihazı kullanılarak bulunmuştur. Her bir parça birbiriyle kare oluşturacak şekilde 4 farklı noktadan ölçümle yapılmıştır. Ölçümler sonucunda aracın herhangi bir parçasında ortaya çıkan ölçüm sonucu diğerlerinden farklı ise, aracın ölçülen parçasının hasarlı olarak işaretlenmesi yapılmıştır. Boya kalınlığı ölçüm cihazları aracın üzerindeki boyanın mikron cinsinden kalınlığını vermektedir. Normalde hasarsız araçlarda boya kalınlıkları yıldan yıla azalacak şekilde 100 gibi değerlerden 80 gibi değerlere kadar inebilmektedir. Fakat herhangi bir nokta 130-140 mikron değerlerinin üzerindeyse parça üzerinde işlem ya da değişim yapıldığını göstermektedir. Çalışmada bu değerler göz önünde bulundurularak işlemler yapılmıştır. Ölçüm yapılan aracın bütün hasar bilgileri, geliştirilen hasar yazılımına girilip kaydedilerek işlem tamamlanmıştır. Şekil 4.13 ve Şekil 4.14'te hasar yazılımı formu gösterilmektedir.

Şekil 4.13. Hasar uygulaması ekran görüntüsü.

Çalışmada hasar uygulamasına girilen değerler sonrasında kaydedildiğinde hem veritabanına bilgiler kaydedilmekte hem de aracın hasar formu oluşturulmaktadır. Bu işlem de başarıyla tamamlanmış, dinamometre testi

yapılan araçların hasar testleri de bitirilerek, veriler toplanmıştır. Bu adım sonrasında, toplanan veriler ile belirlenen makine öğrenmesi algoritmaları uygulanacaktır.

Şekil 4.14. Hasar uygulaması ekran görüntüsü 2.

Hasar uygulaması ile kontrolleri tamamlanıp verileri girilmiş olan formda kayıt tamamlandığında aracı getiren kişiye verilebilecek şekilde hasar raporu hazırlanmıştır. Bu rapor Şekil 4.15'te gösterilmektedir.



Çizelge 4.1. Elde edilen veri türleri ve çeşitliliği.

| Bilgiler               | Çeşitlilik |
|------------------------|------------|
| Şasi numarası          | 832        |
| Plaka                  | 832        |
| Marka                  | 14         |
| Model                  | 21         |
| Tür                    | 5          |
| Yıl                    | 17         |
| KM                     | 832        |
| Beygir Gücü (Ruhsat)   | 832        |
| Kaporta Bilgisi (0-19) | 19         |
| Beygir Gücü (Ölçülen)  | 832        |
| Araç Sahibi            | 832        |
| Cinsiyet               | 2          |
| Şehir                  | 2          |

Verilerin toplanması süresince aracı test yaptırmak için getiren kişilerin çoğunlukla aracı sürekli kullanan kişilerin olmaması, araçlar için yapılan testlerin sadece 1 şehirde yapılması elde edilen verilerden bazılarının kullanılmasına zorunluluk oluşturmamıştır. Kaporta raporlarından elde edilen hasar sonuçlarının da her parça için ayrı ayrı özellik olarak gösterilmesi toplanan verilerin azlığından kaynaklı farklılıklar göstermemiştir. Araçların marka model bilgileri de istenilen çeşitlilik ve sayılarda olmadığından dolayı sonuçları kötü etkileyeceği öngörülmüştür. Dolayısıyla makine öğrenmesi algoritmaları kullanılmadan önce algoritmaların sonuçlarına etki eden öznitelikler kullanılmıştır. Bunların küçük bir kısmı Çizelge 4.2’de gösterilmektedir.

Çizelge 4.2. Tahmin modellerinde kullanılacak verilerin bir kısmı.

| Araç Türü | Yıl  | Kaporta Puan | Araç KM | Güç (Beygir) | Hesaplanan Güç (Yüzdelik) |
|-----------|------|--------------|---------|--------------|---------------------------|
| 1         | 2016 | 0            | 13681   | 95           | 0.91                      |
| 1         | 2016 | 18           | 31005   | 108          | 0.88                      |
| 1         | 2009 | 2            | 105411  | 95           | 0.79                      |
| 1         | 2001 | 7            | 180477  | 118          | 0.69                      |
| 2         | 2007 | 9            | 172987  | 100          | 0.72                      |
| 2         | 2008 | 11           | 89718   | 125          | 0.80                      |
| 3         | 2011 | 1            | 239885  | 115          | 0.83                      |

Makine öğrenmesi kullanılırken veriler öncelikle belirli bir kurala göre normalize edilmektedir. Bu çalışmada da kullanılacak veriler normalize işlemlerine girmiştir. Geçmiş çalışmalar incelendiğinde birden fazla normalize etme yöntemi vardır. Bu yöntemlerden bazıları aşağıdaki denklemlerde gösterilmektedir (Zhang vd., 2008).

$$x_n = \frac{x_0 - x_{\min}}{x_{\max} - x_{\min}} \quad (4.1)$$

$$x_n = \frac{(x_0 - x_{\min})(b - a)}{(x_{\max} - x_{\min}) + a} \quad (4.2)$$

Denklem 4.1’de verilerin hepsini 0 ile 1 değerleri arasına normalize etme işlemi uygulanır. Bu işleme aynı zamanda lineer transformasyon da denir. Bu denklemde x herhangi bir noktadaki normalize edilecek değeri, max ve min ise veriler içerisindeki en küçük ve en büyük değerleri ifade etmektedir. Denklem 4.2’de gösterilen işlem ise verilerin belirlenen iki sayı arasında sonuçları normalize etmek için kullanılmaktadır. Buradaki denklemde a ve b değişkenleri başlangıç ve bitiş değerlerini ifade etmektedir. Bu çalışmada Denklem 4.1’deki normalizasyon işlemi uygulanmış ve bütün veriler 0 ile 1 aralığına normalize edilmiştir.

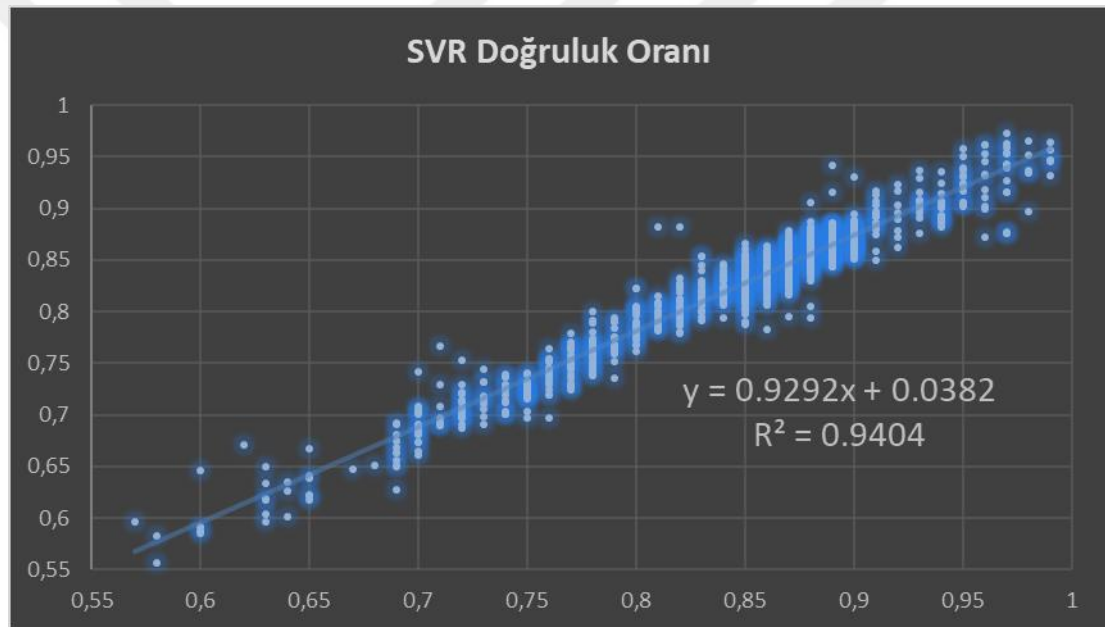
Verilerin algoritmalara girdikten sonra da daha bağımsız ve geçerliliği yüksek sonuçlar üretebilmesi adına çapraz geçerlilik yöntemi uygulanmıştır. Çapraz geçerlilik yöntemi tahminleme yapacak modellerin doğruluklarının belirlenmesinde sıklıkla kullanılmaktadır. Veri setinde bulunan bilgiler kullanılıp tahmin ve gerçek değerler arasındaki bağlantıyı incelemektedir. Çapraz geçerliliğin kullanımı şu şekilde yapılmaktadır. Veri setindeki bir grup veri çıkarılır ve geri kalan bütün veriler ile tahmin işlemi yapılır, sonrasında farklı bir grup veri çıkartılır ve aynı tahmin işlemi tekrar gerçekleştirilir. Bu işlemler bütün veri setine uygulanana kadar devam eder.



Bu çalışmada veri seti büyüklüğü de göz önüne alınarak, veriler 5 kümeye ayrılmış ve sırasıyla 5 küme 5 farklı kombinasyonda tahmin edilmiş ve bu tahminlerin ortalaması alınmıştır.

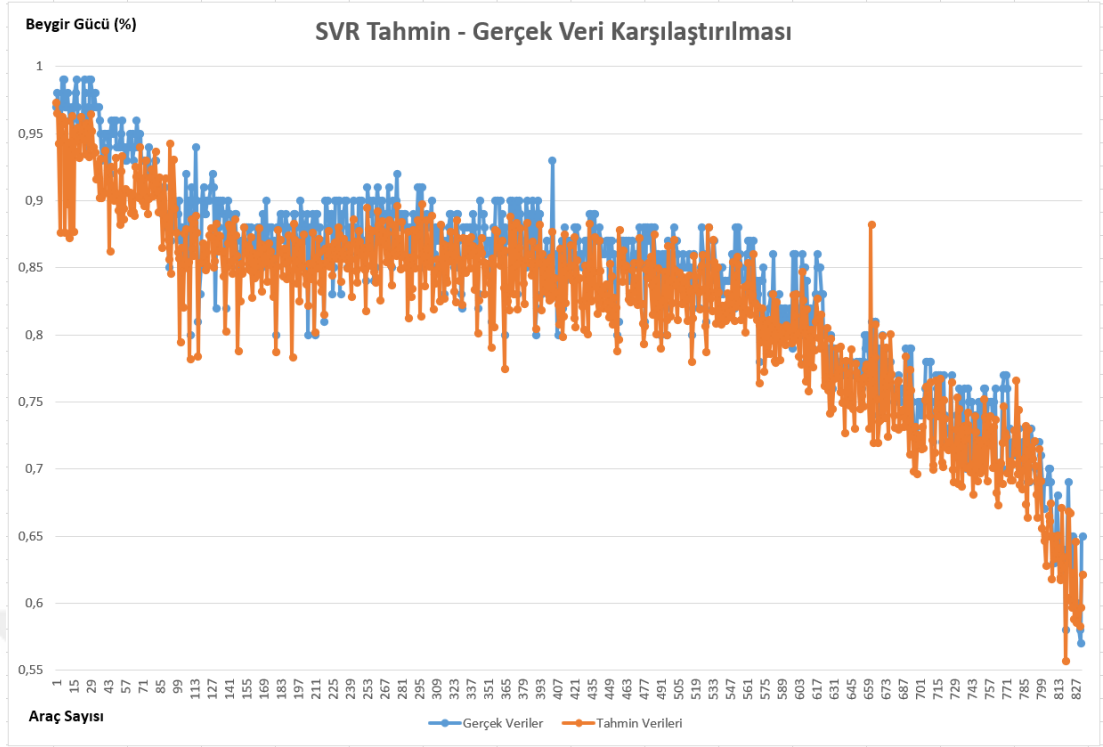
#### 4.5. Destek Vektör Regresyonu Bulguları

Destek vektör regresyonu ile geliştirilen modelde bağımlı ve bağımsız değişkenlerin olduğu veri seti ile modelin  $R^2$  değeri 0.9404 olarak bulunmuştur. Geliştirilen model için polinom, sigmoid ve doğrusal çekirdek fonksiyonları denenmiş ve en iyi sonuç doğrusal fonksiyon olmuştur.



Şekil 4.16. SVR doğruluk oranı.

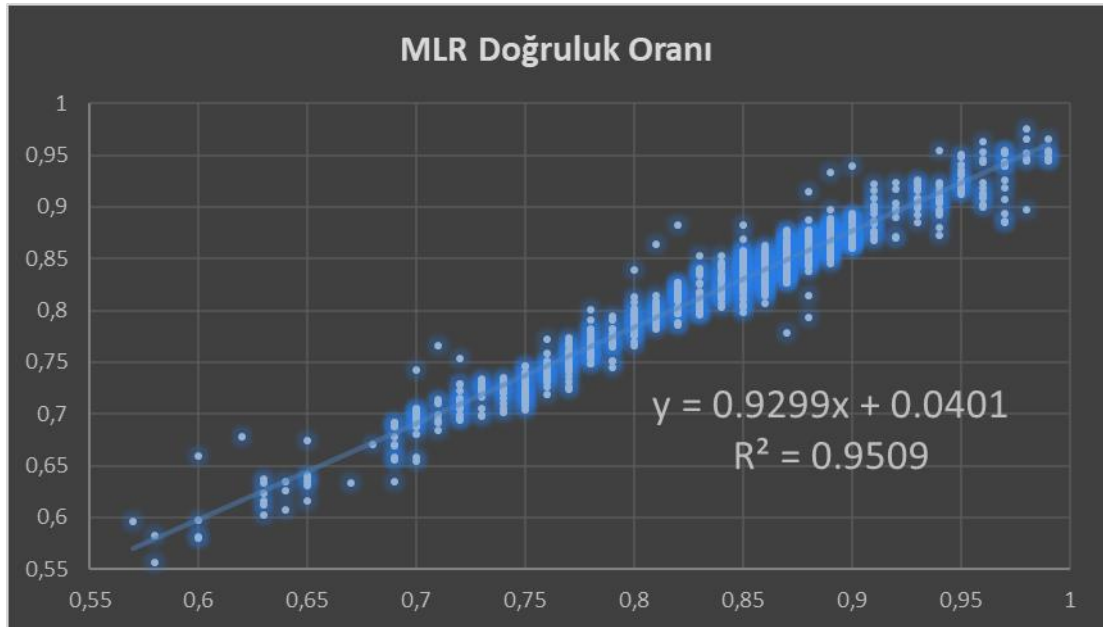
Çalışmada araçlardan elde edilen 832 adet verinin 665 tanesi eğitim verisi olarak kullanılmış 167 tanesi de test verisi olarak kullanılmıştır. Algoritma  $R^2$  doğruluğu hesaplamasında çapraz geçerlilik yöntemi kullanılmıştır. Veriler 5 kümeye bölünmüş ve 5 farklı veri kümesine göre regresyon yapılmıştır. Yapılan farklı verilere göre regresyonların  $R^2$  ortalaması Şekil 4.16'da gösterilmektedir. Şekil 4.17'de gerçek veriler ve tahmin verilerin bir arada olduğu karşılaştırma grafiği gösterilmiştir.



Şekil 4.17. SVR tahmin – gerçek veri karşılaştırılması.

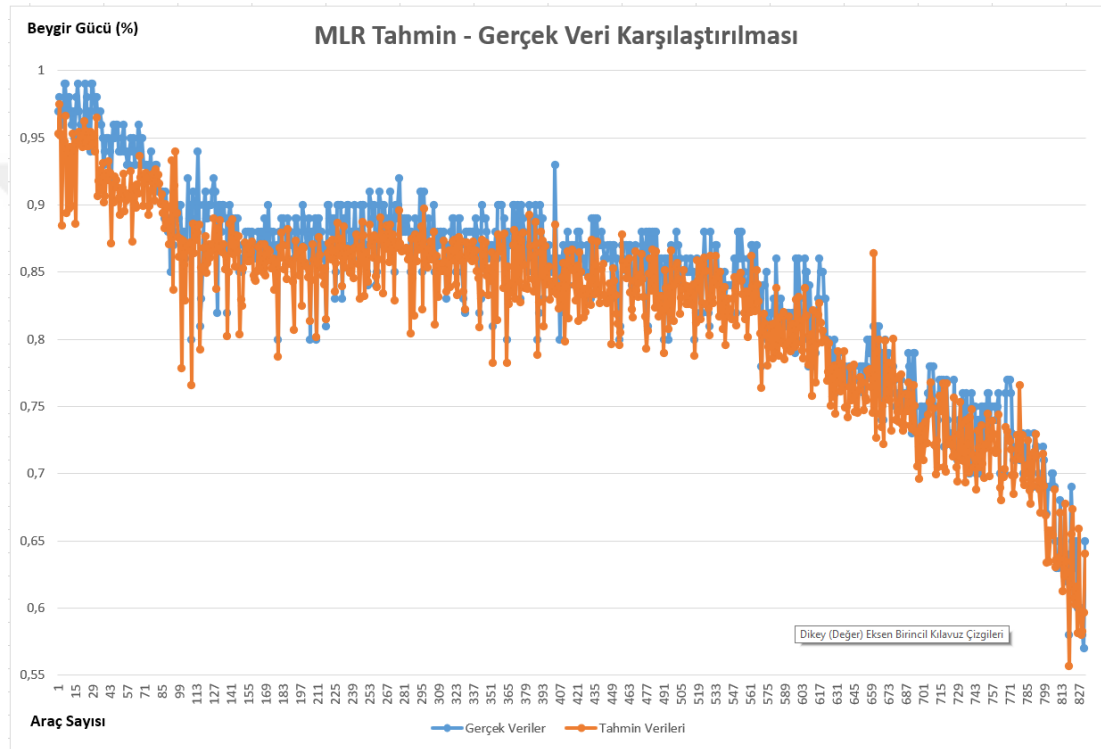
#### 4.6. Çoklu Doğrusal Regresyonu Bulguları

Çoklu doğrusal regresyon kullanılarak geliştirilen modelde bağımlı ve bağımsız değişkenlerin olduğu veri seti ile modelin  $R^2$  değeri 0.9509 olarak bulunmuştur.



Şekil 4.18. MLR doğruluk oranı.

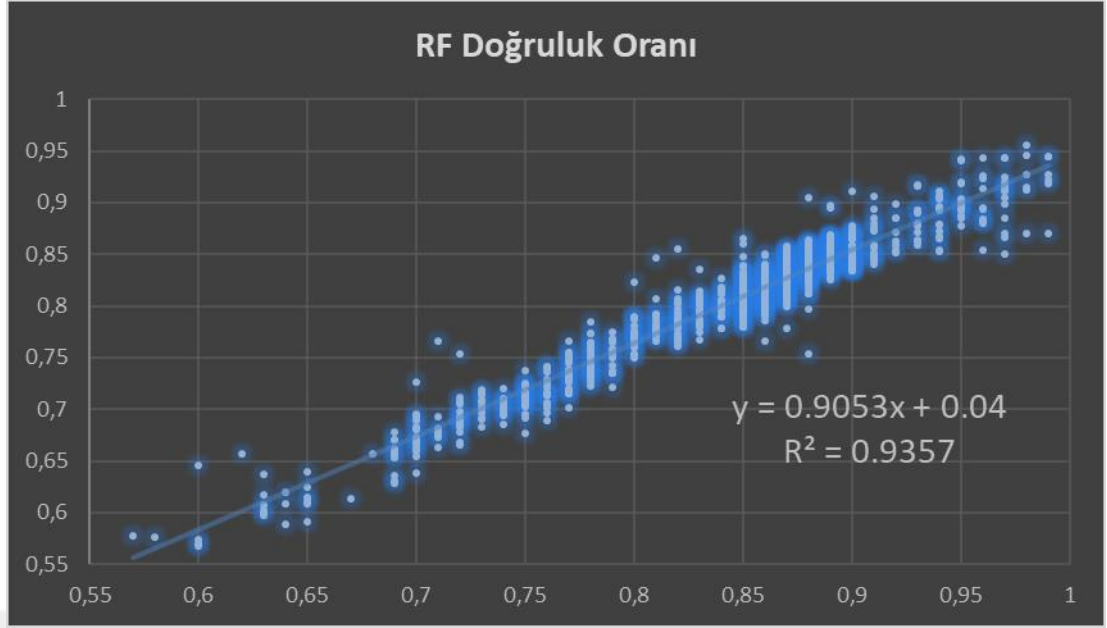
Çalışmada araçlardan elde edilen 832 adet verinin 665 tanesi eğitim verisi olarak kullanılmış 167 tanesi de test verisi olarak kullanılmıştır. Algoritma  $R^2$  doğruluğu hesaplamasında çapraz geçerlilik yöntemi kullanılmıştır. Veriler 5 kümeye bölünmüş ve 5 farklı veri kümesine göre regresyon yapılmıştır. Yapılan farklı verilere göre çoklu doğrusal regresyonu  $R^2$  ortalaması Şekil 4.18’de gösterilmektedir. Şekil 4.19’da gerçek veriler ve tahmin verilerin bir arada olduğu karşılaştırma grafiği gösterilmiştir.



Şekil 4.19. MLR tahmin – gerçek veri karşılaştırması.

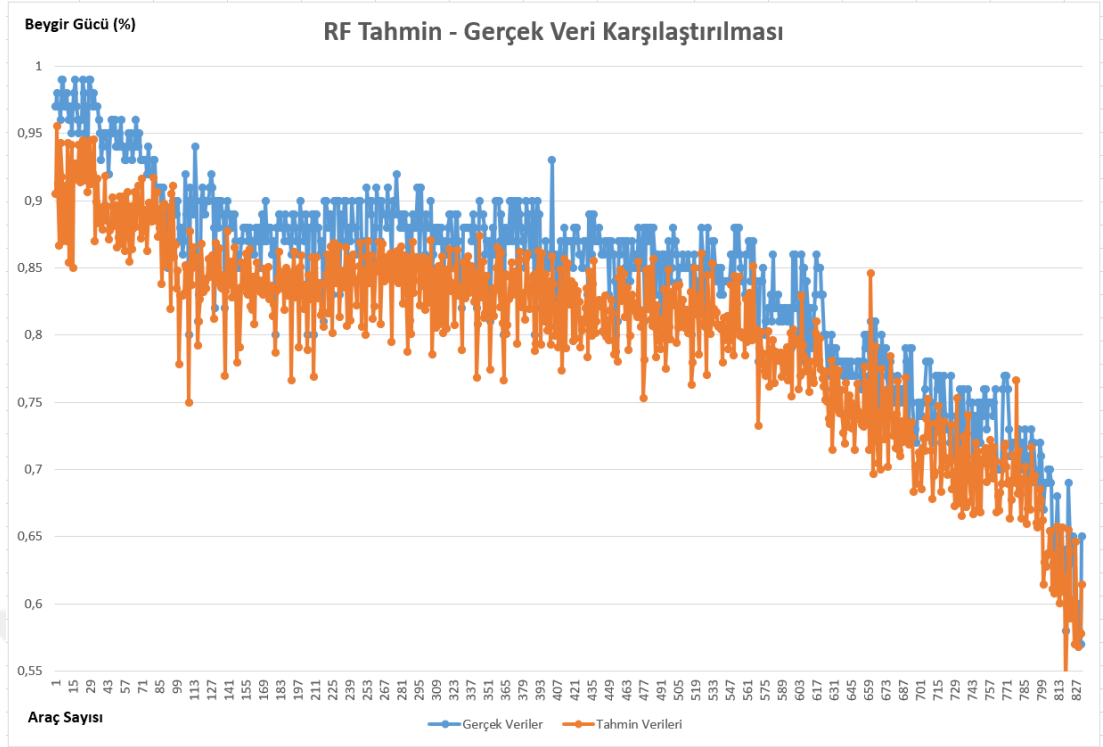
#### 4.7. Rasgele Orman Regresyonu Bulguları

Rasgele orman regresyonu ile geliştirilen modelde bağımlı ve bağımsız değişkenlerin olduğu veri seti ile modelin  $R^2$  değeri 0.9357 olarak bulunmuştur. Geliştirilen modelde 10 adet ağaç kullanılmıştır.



Şekil 4.20. RF doğruluk oranı.

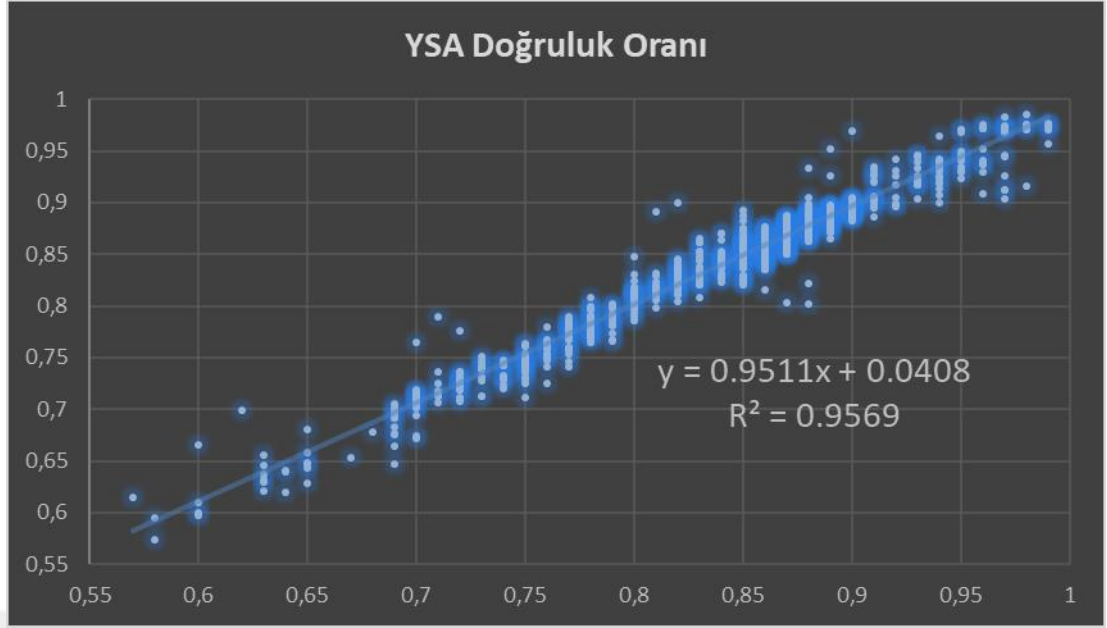
Çalışmada araçlardan elde edilen 832 adet verinin 665 tanesi eğitim verisi olarak kullanılmış 167 tanesi de test verisi olarak kullanılmıştır. Algoritma  $R^2$  doğruluğu hesaplamasında çapraz geçerlilik yöntemi kullanılmıştır. Veriler 5 kümeye bölünmüş ve 5 farklı veri kümesine göre regresyon yapılmıştır. Yapılan farklı verilere göre rastgele orman regresyonunu  $R^2$  ortalaması Şekil 4.20’de gösterilmektedir. Şekil 4.21’de gerçek veriler ve tahmin verilerin bir arada olduğu karşılaştırma grafiği gösterilmiştir.



Şekil 4.21. RF tahmin – gerçek veri karşılaştırılması.

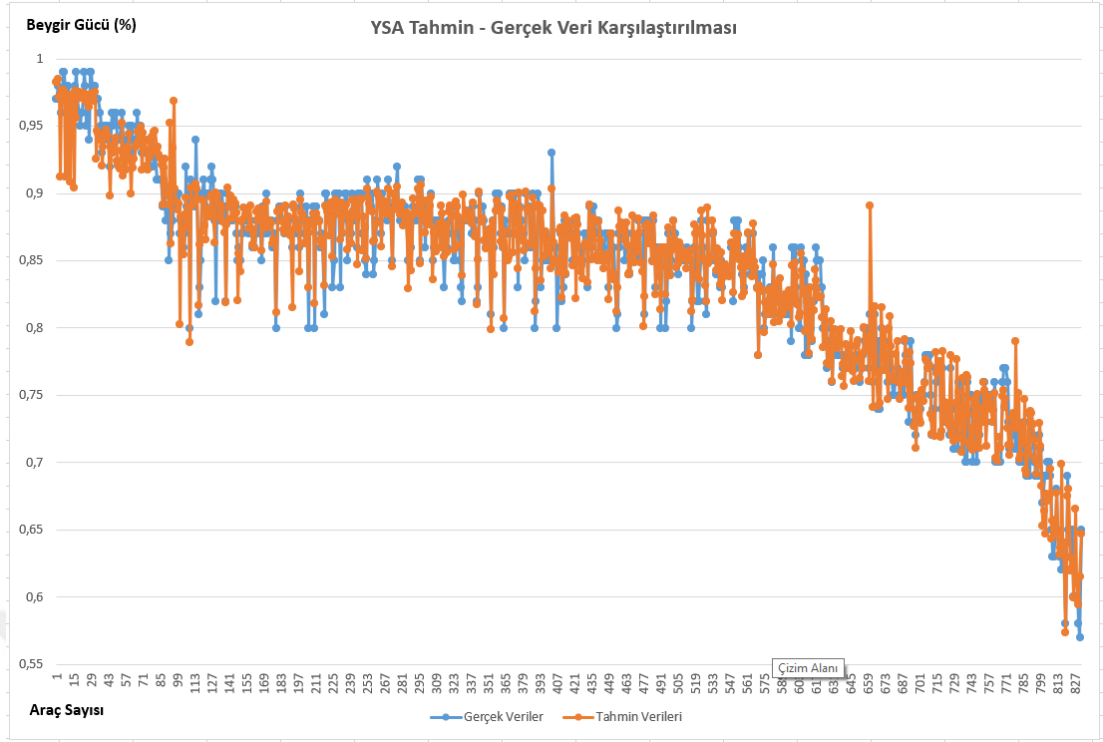
#### 4.8. Yapay Sinir Ağının Bulguları

Yapay Sinir Ağı algoritması ile geliştirilen modelde bağımlı ve bağımsız değişkenlerin olduğu veri seti ile modelin  $R^2$  değeri 0.9569 olarak bulunmuştur. Geliştirilen modelde katman sayıları ve seçimler geçmiş çalışmalarda en başarılı olan seçimler referans olarak kararlaştırılmıştır. Yenileme sayısı 640 olarak seçildiğinde en doğru sonuç alınmıştır. Algoritmada 8 nöron üretilmiştir. 1 gizli katman oluşturulmuştur. Eğitim algoritması olarak Levenberg-Marquardt kullanılmıştır. Maksimum komşuluk fikrine dayanan yinelemeli bir tekniktir. Aktivasyon fonksiyonu olarak sigmoid fonksiyonu kullanılmıştır.



Şekil 4.22. YSA doğruluk oranı.

Çalışmada araçlardan elde edilen 832 adet verinin 665 tanesi eğitim verisi olarak kullanılmış 167 tanesi de test verisi olarak kullanılmıştır. Şekil 4.22’de Yapay sinir ağları  $R^2$  doğruluk grafiği ve Şekil 4.23’te gerçek veriler ve tahmin verilerin bir arada olduğu karşılaştırma grafiği gösterilmiştir. Çapraz geçerlilik yöntemi yapay sinir ağlarında kullanılmamıştır.

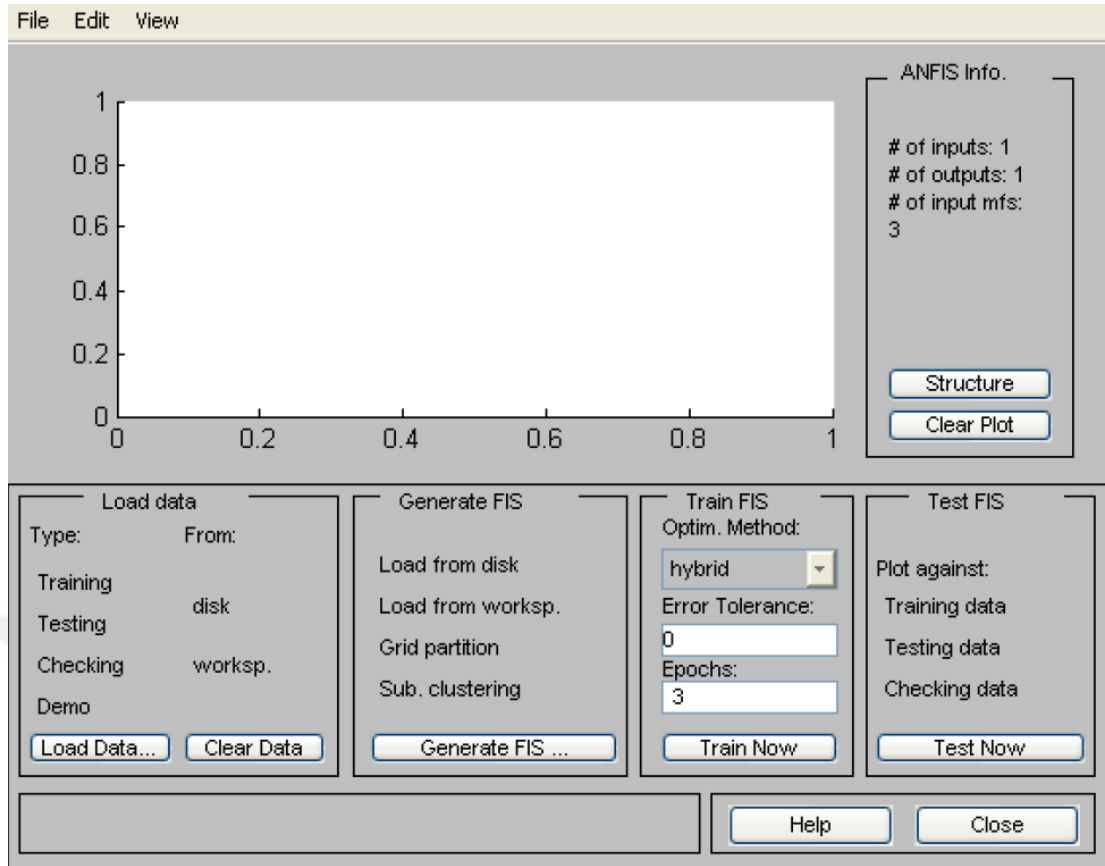


Şekil 4.23. YSA tahmin-gerçek veri karşılaştırması.

#### 4.9. Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi Bulguları

Çalışmada araç performanslarının tahmin edilebilmesi için adaptif ağ tabanlı bulanık mantık çıkarım sistemi de kullanılmıştır. Makine öğrenmesindeki algoritmalar kullanılacağına öncelikle verilerin hazırlanması gerekir.

Çalışmada ANFIS modelini kullanmak için MATLAB 7.2 programında bulunan *Fuzzy Logic Toolbox*'ta yer alan *ANFIS GUI* arayüzü kullanılmıştır. Şekil 4.24'te *ANFIS GUI* formu gösterilmektedir.

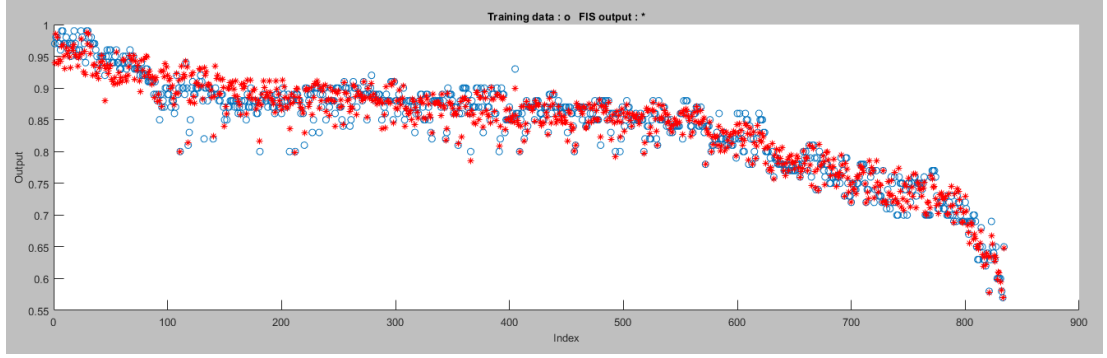


Şekil 4.24. ANFIS GUI.

MATLAB üzerinde çalışmada kullanılacak model oluşturulurken, elde edilen veriler 3'e bölünmüştür. Bunlar sırasıyla; Eğitim seti ve test setidir. ANFIS modelini eğitirken verilerin hangi oranlarda bölüneceği durumu literatürde de başarılı sonuçlar vermiş şekilde belirlenmiştir. Bütün verilerin %80'lik kısmı eğitim için, %20'lik kısmı test için ayrılmıştır.

Şekil 4.24'teki veri yükleme butonuna basarak veriler belirlenen kategorilere göre yüklenmiştir. Şekil 4.25'te gösterilen grafikte mavi nokta işaretleri eğitim, kırmızı artı işaretleri ise test verilerini göstermektedir.





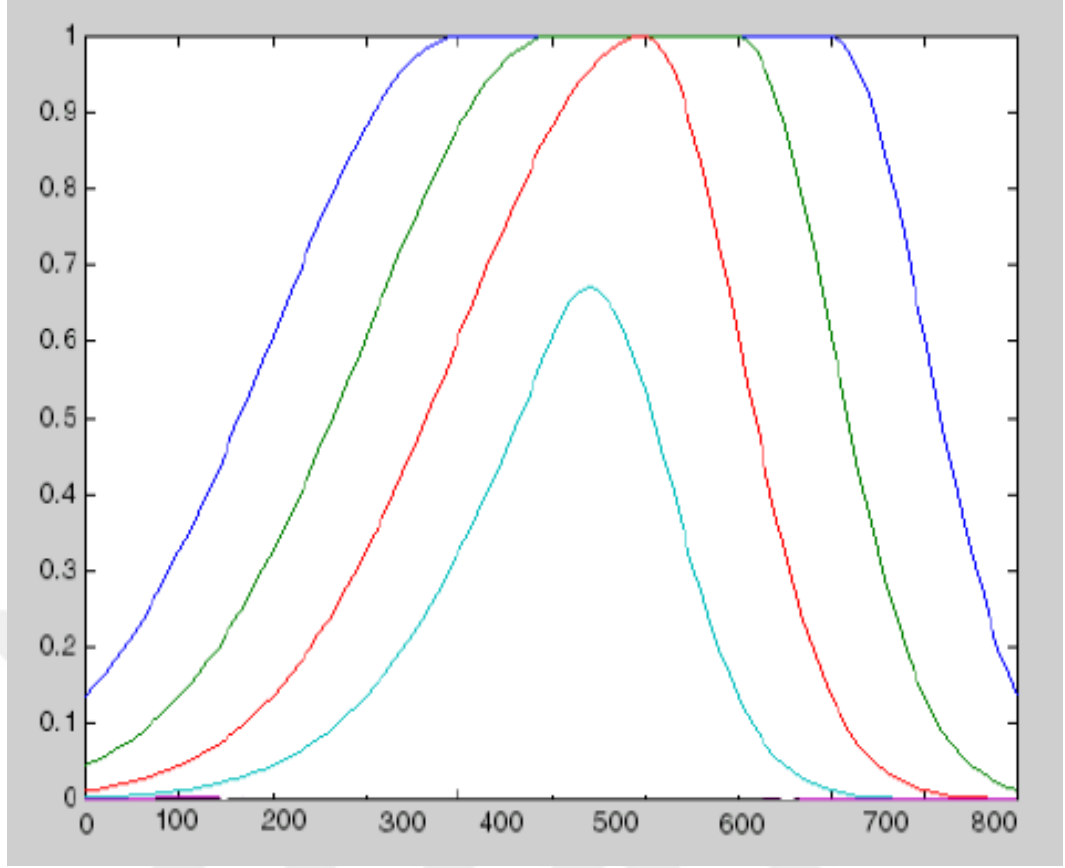
Şekil 4.25. Verilerin ANFIS'e aktarılması.

Veri seti ve test-kontrol-eğitim veri sayıları belirlendikten sonra ANFIS ile oluşturulacak model ile ilgili belirli seçimlerin yapılması gerekmektedir. Belirlenecek olan faktörler sayesinde model en başarılı şekilde kullanılabilir. Bu faktörler aşağıda sıralanmıştır;

- Dilsel değişken sayısı
- Üyelik fonksiyonu tipi
- Optimizasyon çeşidi
- Hata toleransı
- Eğitim çevrim sayısı

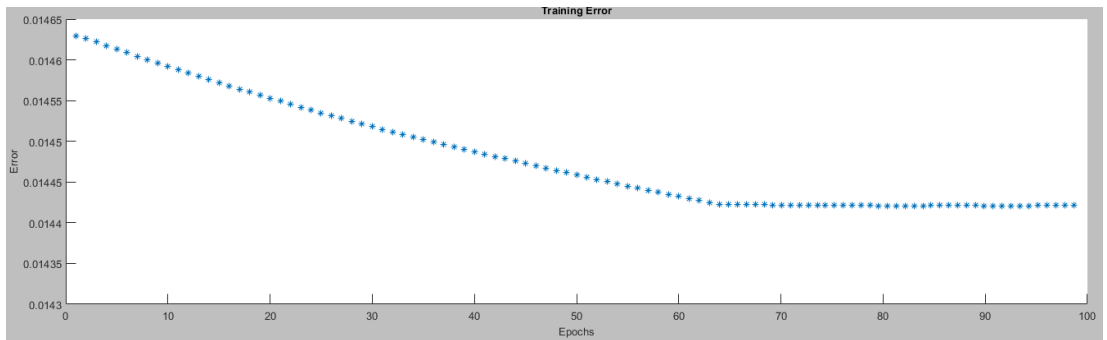
Bu faktörlerden bazıları için seçim işlemi modele bırakılır. Fakat üyelik fonksiyonu seçimi için, mevcutta olan farklı üyelik fonksiyonları test edilerek en iyi sonucu veren üyelik tipi fonksiyonu seçilebilir.

Çalışmada üyelik fonksiyonları ile değerlendirme sonucu hata oranı en düşük olarak 2. tip gauss üyelik fonksiyonu (gauss2mf) seçilmiştir. Gauss2mf üyelik fonksiyonu grafiği Şekil 4.26'da gösterilmektedir.



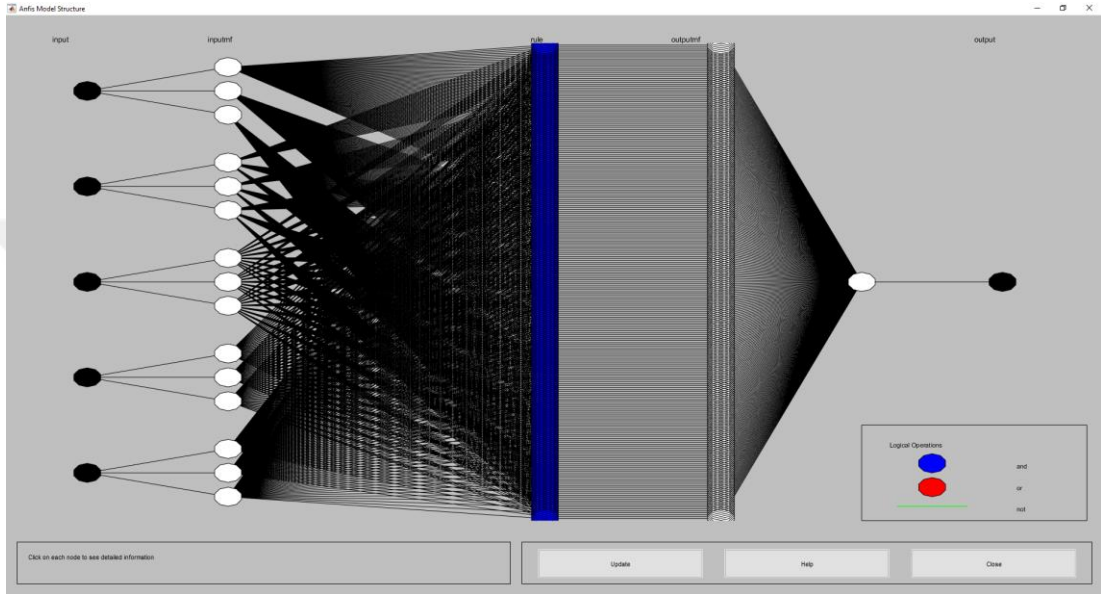
Şekil 4.26. Gauss2mf üyelik fonksiyonu grafiği.

120 çevrim ve 2. tip gauss üyelik fonksiyonu kullanarak yapılan değerlendirmenin öğrenme eğrisi Şekil 4.27’de gösterilmektedir. Şekilde hata tolerans değeri de işaretlenmiştir. Buna ek olarak şekil incelendiğinde 63. çevrim değerinden sonra ağıın öğrenme işlemini bitirdiği gözlemlenmektedir. Öğrenme grafiğinde değerlerin eğimi ortadan kalktıktan sonra öğrenme işleminin devam etmediği anlaşılmaktadır.



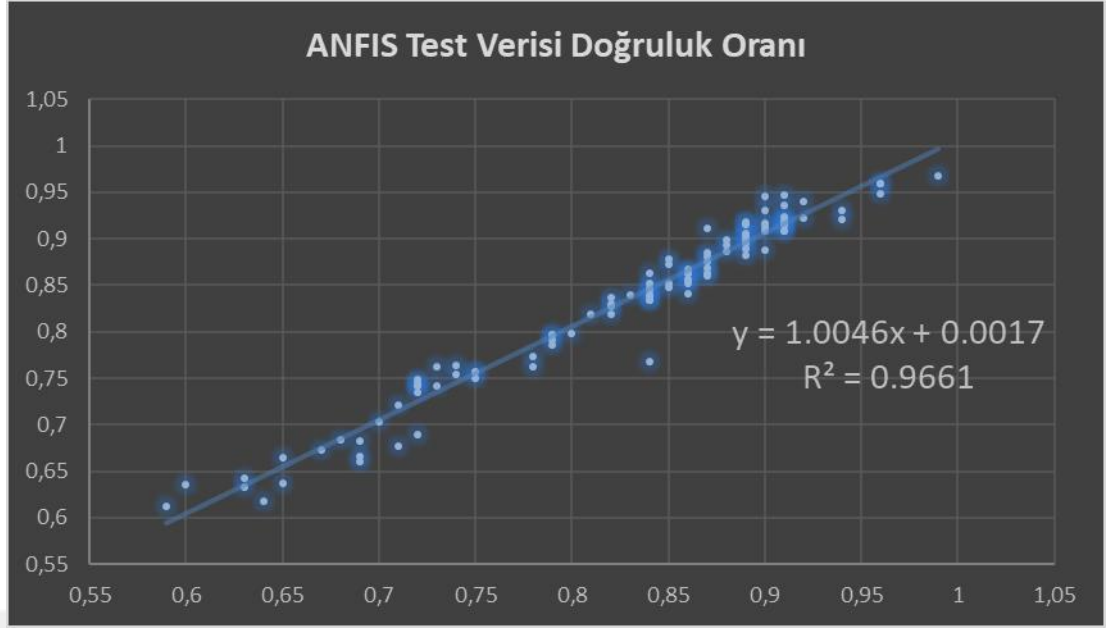
Şekil 4.27. Modelin öğrenme eğrisi.

Şekil 4.28’de ANFIS modelinin ağ yapısı gösterilmektedir. Başlangıçta girdiler bağımsız 5 değişkendir. Bir sonraki adımda girdiler 3 üyelik fonksiyonuna belirli üyelik derecesinde sistem tarafından eklenmişlerdir. Devamında ise üyelik fonksiyonları kullanılarak 243 farklı kural oluşturulmuştur. ANFIS en optimize öğrenme işlemi için 243 farklı kuraldan değerler üretmiş ve tek bir çıktı değerine dönüştürmüştür.



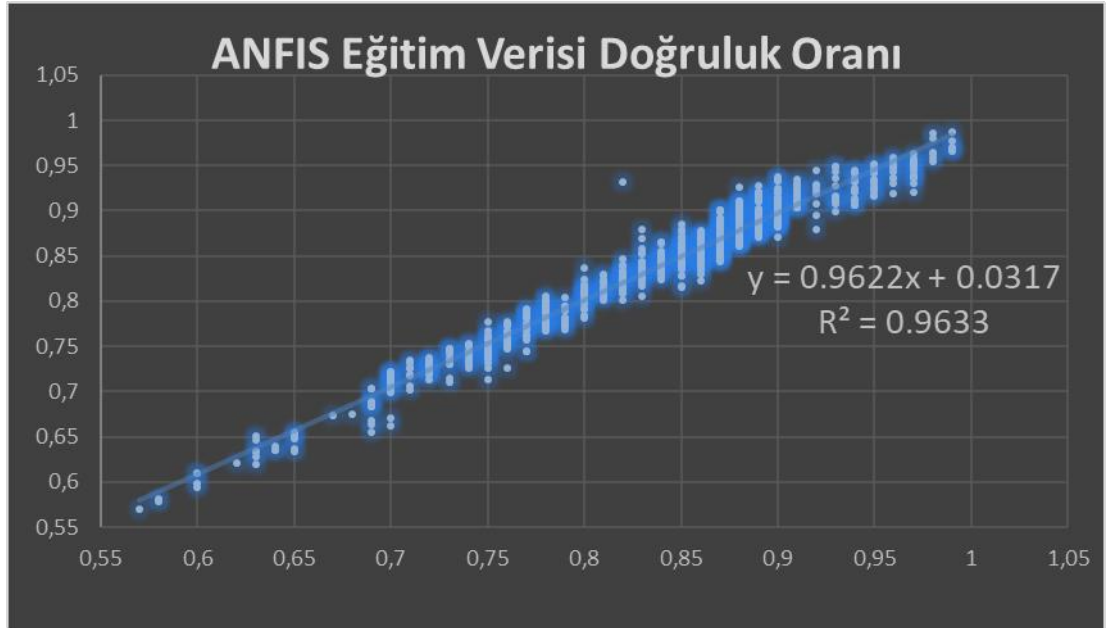
Şekil 4.28. ANFIS ağ yapısı.

ANFIS ile oluşturulan tahminlerin diğer verilerle kıyaslayabilmek için verileri dışarıya aktarabilmek mümkündür. MATLAB programı üzerinden *evalsif* komutu yardımıyla bütün veriler listelenebilir. Bu yöntem kullanılarak elde edilen bütün veriler dinamometre cihazının ölçtüğü gerçek verilerle kıyaslanmıştır. Şekil 4.29’daki grafikte test verilerinin doğruluk oranı grafiği gösterilmiştir.



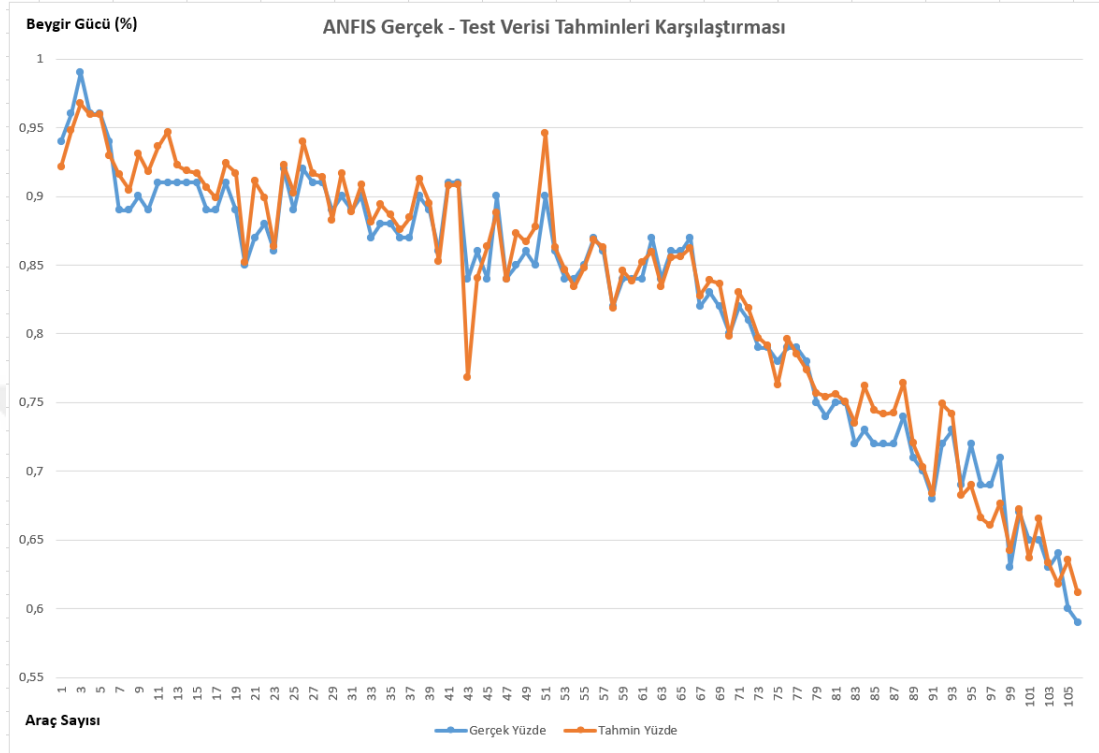
Şekil 4.29. ANFIS test verisi doğruluk oranı.

Adaptif ağ tabanlı bulanık mantık çıkarım sistemi kullanılarak geliştirilen modelde bağımlı ve bağımsız değişkenlerin olduğu test veri seti ile modelin  $R^2$  değeri 0.9661 olarak bulunmuştur. Bu modelin eğitim veri seti ile  $R^2$  değeri 0.9633 olarak bulunmuştur. Şekil 4.30'da eğitim için kullanılan verilerin doğruluk oranı gösterilmiştir.

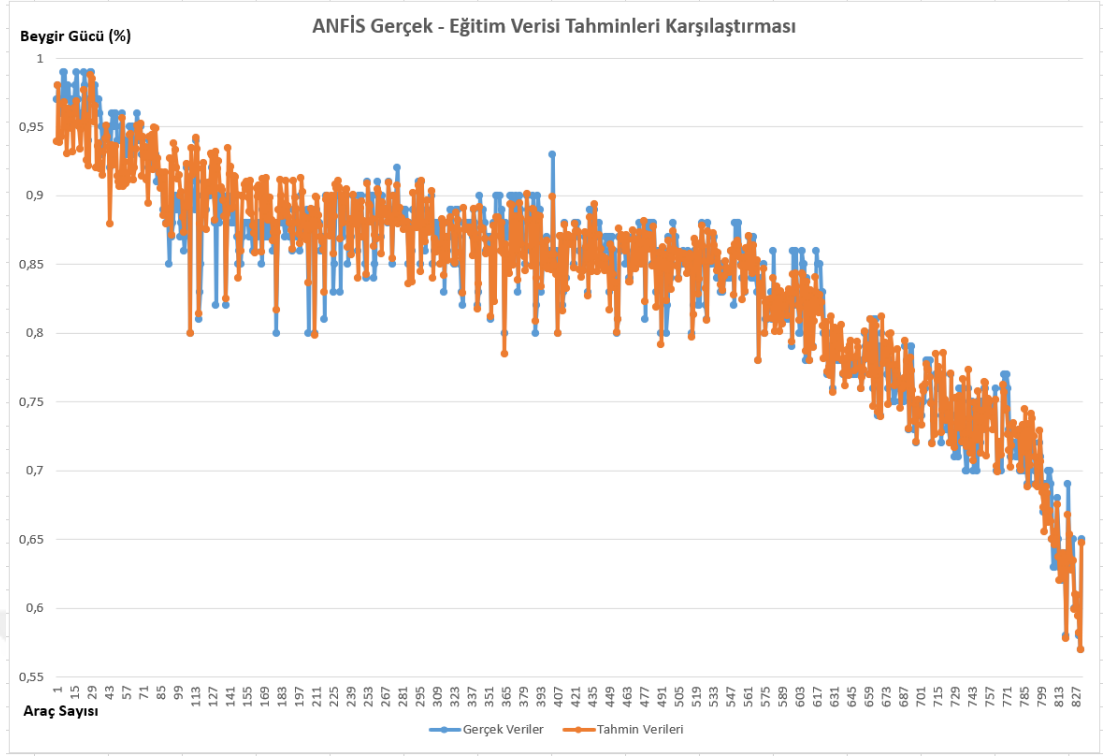


Şekil 4.30. ANFIS eğitim verisi doğruluk oranı.

Şekil 4.31 ve Şekil 4.32’de adaptif ağ tabanlı bulanık mantık çıkarım sisteminde kullanılan eğitim ve test verilerinin ayrı ayrı tahminlerinin karşılaştırmalı grafikleri gösterilmiştir.



Şekil 4.31. ANFIS gerçek-test verisi tahminleri karşılaştırması.



Şekil 4.32. ANFIS gerçek-eğitim verisi tahminleri karşılaştırması.

Çapraz geçerlilik yöntemi adaptif ağ tabanlı bulanık mantık çıkarım sisteminde kullanılmamıştır.

Oluşturulan verisetine göre bütün algoritmalar modellenmiştir. Kullanılan tahmin algoritmalarının sonuçları  $R^2$  değerleri üzerinden değerlendirilmiştir. Bütün algoritmaların  $R^2$  sonuçları Çizelge 4.3'te gösterilmektedir.

Çizelge 4.3. Tahmin algoritmaları  $R^2$  sonuçları.

| Tahmin Algoritması                                | $R^2$ Değeri |
|---------------------------------------------------|--------------|
| Çoklu Doğrusal Regresyon                          | 0.9509       |
| Destek Vektör Regresyonu                          | 0.9404       |
| Rastgele Orman Regresyonu                         | 0.9357       |
| Yapay Sinir Ağları                                | 0.9569       |
| Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi | 0.9661       |

## 5. TARTIŞMA VE SONUÇLAR

Bu tez çalışmasında motorlu araçların performanslarının ölçümünü yapacak atalet tipli şasi dinamometresi cihazları için denetleme kartı ve dinamometre cihazı ile haberleşerek araçların ölçümlerini yapacak yazılımlar geliştirilmiştir. Geliştirilen bu yazılımlardan birisi de araç kaporta ölçüm cihazları kullanılarak ölçümlerin gerçekleştirilip kaydedebilen detaylı bir hasar analiz sistemidir. Ayrıca araçların beygir gücü ölçümleri ve hasar durumlarıyla birlikte araçların ruhsatlarında yazılı olarak bulunan beygir güçleri de kullanılarak, makine öğrenmesi yöntemleriyle performans tahminleri yapılmıştır.

Çalışmada atalet tipli şasi dinamometresini denetleyerek, oluşan dalgaların sayısal olarak sonuçlanmasını sağlayarak dinamometre yazılımına gönderecek, aynı zamanda testler esnasında karesel dalgalar üreterek yazılımın doğru çalışıp çalışmadığını denetleyecek mikrodenetleyici tabanlı bir sistem hazırlanmıştır. Hazırlanan bu sistem ile hem dinamometre cihazı ile başarılı bir şekilde haberleşme sağlanmış hem de cihazdan veri göndermen, denetleyici tarafından gönderilen kare dalgalar sayesinde geliştirilen dinamometre yazılımının verileri doğru hesaplayabildiği, aracın beygir ve tork hesaplarını hatasız olarak hesaplayabildiği gözlemlenmiştir.

Çalışmada denetleyici kart üzerinden gelen verileri işleyerek araç ile ilgili bilgilerin gösterildiği, hesaplamaların yapıldığı, kullanıcıya raporlar üretebilen, kullanıcı dostu arayüze sahip olan bir dinamometre yazılımı geliştirilmiştir. Geliştirilen bu yazılım ile birlikte, araç dinamometre cihazının üzerine geldiği andan itibaren bütün ölçümler hem grafiklerle birlikte anlaşılır bir şekilde gösterilebilmiş hem de yazılan filtreleme algoritmaları sayesinde veri kaybı olmadan, gerçek grafiklere yakın olacak şekilde beygir gücü ve tork hesaplamalarını yapabilmektedir. Yapılan tüm hesaplamalar ve bilgiler anlaşılması kolay bir raporlama yöntemiyle kullanıcıya sunulmuştur.

Çalışmada asıl hedeflenen konu, araçların şasi dinamometresi ile performans ölçümlerinin azaltılarak, test esnasında ortaya çıkabilecek olası motor hasarları

ve test yapan kişinin başına gelebilecek olası iş kazalarının ortadan kaldırılabilmesi adına araç beygir gücü sonucunun makine öğrenmesi teknikleri kullanılarak tahmin edilebilmesidir. Hedeflenen bu sonuç için donanımsal olarak hazırlanan dinamometre ölçümleri, boya ölçüm cihazı kullanılarak tespit edilen araç hasar durumları ve araç ile araç kullanıcısının bilgileri kullanılarak belirlenen makine öğrenmesi teknikleriyle regresyon analizleri yapılmıştır.

Çalışmada Performans tahminleri için Destek vektör regresyonu, Çoklu doğrusal regresyon, Rastgele orman regresyonu, Yapay sinir ağları ve Adaptif ağ tabanlı bulanık mantık çıkarım sistemi algoritmaları kullanılmıştır. Kullanılan bu algoritmalar sonucunda bulunan beygir gücü bilgisi ile dinamometre cihazı kullanılarak yapılan test sonucunda bulunan beygir gücü bilgisi arasındaki farklılıklar incelenmiş ve en başarılı olan algoritma bulunmuştur. En başarılı algoritma  $0.9661 R^2$  değeriyle Adaptif ağ tabanlı bulanık mantık çıkarım sistemi algoritmaları kullanılarak elde edilmiştir. Algoritmaların sonuçları incelendiğinde beygir gücü tahminlerinde en yüksek iki kriterin aracın kilometre ve yaş bilgisi olduğu gözlemlenmiştir. Sonrasında aracın hasar durumunun etkili olduğu anlaşılmıştır.

Yapılan bu çalışma aracın performans hesaplarına farklı bir bakış açısıyla yaklaşan tek çalışma özelliğine sahiptir. Ayrıca regresyon analizleri yapılırken kullanılan verilerin de geliştirilen denetleme kartı ve yazılımlar ile birlikte çalışma süresince elde edilmesi veri bakımından tezin özgünlüğünü göstermektedir.

Çalışmada, günümüzde oldukça popüler olan ve tahminsel sonuçlarıyla oldukça başarılı algoritmalara sahip olan derin öğrenme algoritmaları kullanılmamıştır. Çünkü çalışma süresince 832 farklı araçtan beygir gücü bilgisi toplanmış ve bu sayı derin öğrenme yöntemleri için oldukça az olduğu ve yanlış sonuçlar üreteceği öngörülmüştür. Gelecekteki çalışmalarda, performans testi yapılarak beygir gücü hesaplanan araçların artmasıyla derin öğrenme yöntemleri denenebilir ve daha da başarılı sonuçlar elde edilebilir. Ayrıca çalışmada çeşitlilik olarak az olan araç marka model ve araç sahiplerinin bilgilerinin artmasıyla



oldukça etkili daha güçlü sonuçlar alınması sağlanabilir. Beygir gücü verilerinin artmasıyla birlikte geliştirilebilecek otomatik bir tahmin sistemi sayesinde, özellikle ikinci el araçların satışlarında zorunlu olan ekspertiz testlerinin yapılması esnasında, dinamometre testlerinin yapılmaması, araç performans tahmini yapan sistemin kullanılması, dinamometre testi sırasında oluşabilecek motor arızası problemlerini azaltılabilir, çalışan kişilerin iş güvenlikleri de artırılabilir. Bu çalışmadaki analizlerin, gelecekteki yapılacak çalışmalara ışık tutması beklenmektedir.



## KAYNAKLAR

- Adewole, A. P., Akinwale, A. T., Akintomide, A. B., 2011. Artificial neural network model for forecasting foreign exchange rate. *World of Computer Science and Information Technology Journal (WCSIT)*, 1(3), 110-118.
- Akman, M., 2010. Veri Madenciliğine Genel Bir Bakış ve Random Forests Yönteminin İncelenmesi: Sağlık Alanında Bir Uygulama. Ankara Üniversitesi, Sağlık Bilimleri Enstitüsü. Yüksek Lisans Tezi, Ankara.
- Aksoy, F., 2018. Makine öğrenmesi ve derin öğrenme yöntemiyle kısa dönem rüzgâr gücü üretim tahmini, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 109s, İstanbul.
- Aksu, İ. Ö., Çoban, R., 2014. Çok Katmanlı-Geri Beslemeli Yapay Sinir Ağı (MFLNN) ve Parçacık Sürü Optimizasyon Algoritması (PSO) Kullanarak Bir DC Motor Tanılaması, Eleco 2014 Elektrik Elektronik Bilgisayar ve Biyomedikal Mühendisliği Sempozyumu, 27-29 Kasım, Bursa, 758-761.
- Alpaydın, E., 2004. *Introduction to Machine Learning*. MIT Press, 640p, Cambridge.
- Altuntaş, Y., 2007. Uçak Kara Kutusundan Alınan Kayıtların Yapay Sinir Ağları İle İncelenmesi. Haliç Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 94s, Ankara.
- Aydın, D. B., 2016, Yoğun Bakım Ünitesinde İdrar Yolu Enfeksiyonunu Tanımlama için Yapay Sinir Ağı Tabanlı bir Karar Destek Sisteminin Geliştirilmesi, Bozok Üniversitesi, Fen Bilimleri Enstitüsü. Yüksek Lisans Tezi, 67s, Yozgat.
- Ayodele, T. O., 2010. Types of machine learning algorithms. *New advances in machine learning*, 3, 19-48.
- Baddari, K., Djarfour, N., Aïfa, T., Ferahtia, J., 2010. Acoustic impedance inversion by feedback artificial neural network. *Journal of Petroleum Science and Engineering*, 71(3-4), 106-111.
- Batmaz, İ., 1998. Bilgisayar ve veri toplama sisteminin taşıt performans ölçümlerinde kullanılması, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 13(2), 171-179.
- Breiman, L., 2001. Random forests. *Machine learning*, 45(1), 5-32.
- Burden, F., Winkler, D., 2008. Bayesian regularization of neural networks. *In Artificial neural networks* 1, 23-42.
- Chen, S.T., Yu, P.S., 2007. Support vector networks pruning on flood forecasting and data mining. *Journal of Hydrology*, 347, 67-78.

- Cornell, 2020. Support Vector Machines. Erişim Tarihi: 05.02.2020. <http://pi.math.cornell.edu/~numb3rs/kostyuk/num219.htm>
- Cutler, A., Cutler, D. R., Stevens, J.R., 2012. Esemble Machine Learning, Springer, 329p, New York.
- Çelik, M. B., Bayır, R., Özdalyan, B., 2007. Bilgisayar destekli motor test standının tasarımı ve imalatı. Teknoloji, 10(2), 131-141.
- Demirkoparan, F., Zontul, M., Kaynar, O., 2010. Ham petrol fiyatlarının ANFIS ile tahmini. Anadolu Bil Meslek Yüksekokulu Dergisi, 17, 3-14.
- Dey, A., 2016. Machine learning algorithms: a review. International Journal of Computer Science and Information Technologies, 7(3), 1174-1179.
- Drucker, H., Burges, C., Kaufman, L., Smola, A., Vapnik, V., 1997. Support vector regression machines, advances in neural information processing systems 9, 1-4 February, CO, 155-161.
- Dunham, M. H., 2003. Data Mining Introductory and Advanced Topics. Prentice Hall/Pearson Education, 315p, New Jersey.
- Efe, M. Ö., 2004. Yapay Sinir Ağları ve Uygulamaları. Boğaziçi Üniversitesi, 141s, İstanbul.
- Erilli, N., Eğrioğlu, E., Yolcu, U., Aladağ, Ç., Uslu, V., 2010. Türkiye’de enflasyonun ileri ve geri beslemeli yapay sinir ağlarının melez yaklaşımı ile öngörüsü. Doğu Üniversitesi Dergisi, 11 (1), 42-55.
- Es, H. A., Hamzaçebi, C., 2014. Yapay Sinir Ağları İle Türkiye Net Enerji Talep Tahmini. Journal of the Faculty of Engineering and Architecture of Gazi University, 29 (3), 495-504.
- Fikir, 2019. Sinir Sisteminin Yapısı, Görevi ve İşleyişi – Nöron Nedir? Nöron Kaça Ayrılır. Erişim Tarihi: 01.02.2020. <https://www.fikir.gen.tr/sinir-sisteminin-yapisi-gorevi-ve-isleyisi-noron-nedir-noron-kaca-ayrilir/>
- Focus Applied Technologies, 2019. Automotive Engine Dynamometer. Erişim Tarihi: 20.12.2019. <http://www.focusappliedtechnologies.com/aed.html>
- Forsyth, R., 1989. Machine learning: Principles and techniques. Chapman & Hall Ltd, 252 p, United Kingdom.
- Ganatra, A., Kosta, Y. P., Panchal, G., Gajjar, C., 2011. Initial classification through back propagation in a neural network following optimization through GA to evaluate the fitness of an algorithm. International Journal of Computer Science and Information Technology, 3(1), 98-116.

- Guo, L., Wang, H., 2008. Information management system of automobile chassis dynamometer based on multi agent Web serve. IEEE International Conference on Computational Intelligence and Security, 13-17 December, Suzhou, 66-70.
- Gunn, S. R., 1998. Support vector machines for classification and regression. ISIS technical report, 14(1), 5-16.
- Hastie, T., Tibshirani, R. and Friedman, J., 2001. The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer Series in Statistics, 747p, California.
- Haykin, S., 1994. Neural Networks: A Comprehensive Foundation. Prentice Hall PTR, 768p, New York.
- Hocaoğlu, F., Kurban, M., 2005. Adaptif Ağ Tabanlı Bulanık Mantık Çıkarım Sistemi İle Eskişehir Bölgesi İçin Güneşlenme Süreleri Tahmini. Elektrik-Elektronik-Bilgisayar Mühendisliği, 11, 22-25.
- Huang, G., Zhu, Q., Siew, C., 2006. Extreme Learning Machine: Theory and Applications. Neurocomputing, 70(1-3), 489-501.
- Jang, J. S., 1993. ANFIS: adaptive-network-based fuzzy inference system. IEEE transactions on systems, man, and cybernetics, 23(3), 665-685.
- Kandananond, K., 2011. Forecasting electricity demand in Thailand with an artificial neural network approach. Energies, 4(8), 1246-1257.
- Korkmaz, E., 2016. Farklı Yüksekliklerdeki Rüzgar Hızının Yapay Zeka Teknikleriyle Tahmini Kullanılarak Elektriksel Güç Hesabı, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 125 s, İstanbul.
- Liu, P., Sun, J., Han, L., Wang, B., 2014. Research on the construction of macro assets price index based on support vector machine. Procedia Computer Science, 29, 1801-1815.
- Lourakis, M. L. A., Argyros, A. A., 2005. Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment?. In Tenth IEEE International Conference on Computer Vision (ICCV'05), 17-21 October, Beijing, 1526-1531.
- Makas, Y. ve Karaatlı, M., 2016. Yapay Sinir Ağlarıyla Hidroelektrik Enerji Üretiminin Çok Dönemli Tahmini. Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 21 (3), 757-772.
- Marsland, S., 2015. Machine Learning An Algorithmic Perspective. CRC Press Taylor & Francis Group, 457p, New York.

- Martino Race Engines, 2019. Engine Dyno Service. Eriřim Tarihi: 20.12.2019. <https://martinoraceengines.com/engine-dyno-service/>
- Mitchell, M.T., 1997. Machine Learning. McGraw-Hill Education, 432p, New York.
- Ok Y., 2010. Adaptif Sinirsel Bulanık Mantık Çıkarım Sistemi (ANFIS) ile Türkiye'de Orta Dönemli Elektrik Enerjisi Talep Tahmini, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 118s, Ankara.
- Orhunbilge N., 2017. Uygulamalı Regresyon ve Korelasyon Analizi, Nobel Akademik Yayıncılık, 386, İstanbul
- Özekes, S., 2003. Veri Madenciliği Modelleri ve Uygulama Alanları. İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 2(3), 65-82.
- Özgan, E., Kap, T., Beycioğlu, A., Emiroğlu, M., 2009. Asfalt betonunda Marshall stabilitesinin uyarlamalı sinirsel bulanık mantık yaklaşımı ile tahmini. 5. Uluslararası İleri Teknolojiler Sempozyumu, 13-15 Mayıs, Karabük, 1-4.
- Özkan, Y., 2013. Veri Madenciliği Yöntemleri. Papatya Yayıncılık, 304s, İstanbul.
- Öztemel E., 2003. Yapay Sinir Ağları. Papatya Yayıncılık, 232s, İstanbul
- Patel J., Parekh F., 2014. Forecasting Rainfall Using Adaptive Neuro-Fuzzy Inference System (ANFIS). International Journal of Application or Innovation in Engineering & Management, 3(6), 262-269.
- Potter, C. W., Negnevitsky, M., 2006. Very short-term wind forecasting for Tasmanian power generation. IEEE Transactions on power systems, 21(2), 965-972.
- Scikit-Learn CO, 2013. Standart Scaler. Eriřim Tarihi: 04.03.2020. <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>
- Shahsavari, S., Bagheri, G., Mahjub, R., Bagheri, R., Radmehr, M., Rafiee-Tehrani, M., Dorkoosh, F. A., 2014. Application of artificial neural networks for optimization of preparation of insulin nanoparticles composed of quaternized aromatic derivatives of chitosan. Drug research, 64(03), 151-158.
- Smola, A. J., Schölkopf, B., 2004. A tutorial on support vector regression. Statistics and computing, 14(3), 199-222.
- Subaşı, S., Şahin, İ., Çomak, B., 2010. Tahribatsız test sonuçları kullanılarak uçucu kül ikameli betonlarda basınç dayanımının ANFIS ile tahmini. Uluslararası Teknolojik Bilimler Dergisi, 2(3), 9-16.
- Sullivan, W., 2017. Machine Learning Beginners Guide Algorithms. CreateSpace Independent Publishing Platform, 266p, North Charleston.

- Suthaharan, S. 2016. Machine Learning Models and Algorithms for Big Data Classification. Springer International Publishing, 378p, New York.
- Sutton, R. S., Barto, A. G., 1999. Reinforcement learning. Journal of Cognitive Neuroscience, 11(1), 126-134
- Theodoridis, S., 2015. Machine Learning A Bayesian and Optimization Perspective. Academic Press of Elsevier, 1160p, London
- Tsoukalas, L.H., Uhring, R.E., 1997. Fuzzy and Neural Approaches in Engineering. John Wiley & Sons, 606p, New York.
- TÜİK, 2020. Motorlu Kara Taşıtları. Erişim Tarihi: 10.10.2020. <https://data.tuik.gov.tr/Bulten/Index?p=Motorlu-Kara-Tasitlari-Eylul-2020-33657>
- Turhan, C., Gökçen, G., Kazanasmaz, Z.T., 2013. Yapay Sinir Ağları ile İzmir'deki Çok Katlı Binaların Toplam Enerji Tüketimlerinin Tahmin Edilmesi. 11. Ulusal Tesisat Mühendisliği Kongresi ve Teskon+Sodex Fuarı Bildirileri, 17- 20 Nisan, İzmir, 61-68.
- Turing, A. M., 1950. Computing machinery and intelligence. Mind, 59, 433-460.
- Türkmen, İ., Korkmaz, Y., 2011. Computation of the pressure altitude using adaptive neuro fuzzy inference system for air data computer in aircrafts. IEEE 19th Signal Processing and Communications Applications Conference (SIU), 20-22 April, Antalya, 1141-1144.
- Vapnik, V. N., Chervonenkis A., 1964. A note on one class of perceptrons. Automation and Remote Control, 25(1), 838-845.
- Vural, B. B., 2007. Yapay sinir ağları ile finansal tahmin, Finans ve Ekonomi Alanında Yapay Sinir Ağları Uygulamaları, Ankara Üniversitesi, Sosyal Bilimleri Enstitüsü. Yüksek Lisans Tezi, 161s, Ankara
- Wang, W. C., Chau, K. W., Cheng, C. T., Qiu, L., 2009. A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. Journal of hydrology, 374(3-4), 294-306.
- Wang, Y., Wang, B., Zhang, X., 2012. A new application of the support vector regression on the construction of financial conditions index to CPI prediction. Procedia Computer Science, 9,1263 – 1272.
- Weisberg, S., 2005. Applied Linear Regression. John Wiley & Sons, Hoboken, 317p, New Jersey.

- Xu, Y., 2003. Using Data Mining in Educational Research: A Comparison of Bayesian Network with Multiple Regression in Prediction, The University of Arizona, Ph.D. Thesis, 253p, Arizona.
- Yadav, N., Yadav, A., Kumar, M., 2015. An introduction to neural network methods for differential equations. Springer, 114p, Netherlands
- Zhang, T., Zhang, D., 2007. Agent-based simulation of consumer purchase decision-making and the decoy effect. Journal of business research, 60(8), 912-922.
- Zhang, W., Qu, F., Wang, X., 2008. Development of the MCG-200 simulated chassis dynamometer based on fuzzy-PID. IEEE International Conference on Automation and Logistics, 1-3 September, Qingdao, 1794-1798.
- Zhao, S., Li, N., 2012. Chassis dynamometer for hybrid electric vehicle based on controller area network. IEEE Third International Conference on Intelligent Control and Information Processing, 10-15 July, Dalian, 352-354.

SDÜ Mühendislik Fakültesi Bilgisayar Mühendisliği – Araştırma Görevlisi (2012-Halen).

### **Yayınlar**

Koyun, A., Cankaya, I. A., 2018. Implementation of a Beacon-Enabled Mobile Indoor Navigation System Using Augmented Reality. *Tehnički vjesnik*, 25(4), 979-985.

Yuksel, A. S., Senel, F. A., Cankaya, I. A., 2019. Classification of soft keyboard typing behaviors using Mobile device sensors with machine learning. *Arabian Journal for Science and Engineering*, 44(4), 3929-3942.

Cankaya, I. A., Yuksel, A. S., Koyun, A., Yigit, T., 2015. A Dynamic Content Generation Tool for OOP Course. *International Journal of Computer Theory and Engineering*, 7(1), 66-71.

Yigit, T., Koyun, A., Yuksel, A. S., Cankaya, I. A., 2014. Evaluation of blended learning approach in computer engineering education. *Procedia-Social and Behavioral Sciences*, 141, 807-812.

Cankaya, I. A., Koyun, A., Yigit, T., Yuksel, A. S., 2015. Mobile indoor navigation system in iOS platform using augmented reality. In 2015 9th International Conference on Application of Information and Communication Technologies, Moskow, 10-12 October, 281-284.



Çankaya, İ. A., Yüksel, A. S., Koyun, A., 2014. IOS platformunda artırılmış gerçeklik ile yön belirleme. Akademik Bilişim, Mersin, 12-15 Eylül, 5-7.

