



T.C.

SELÇUK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

YAPAY ZEKA TABANLI YÖNTEMLER KULLANILARAK FUTBOL MÜSABAKALARININ SONUÇLARININ KESTİRİLMESİ VE HİBRİT MODEL ÖNERİLERİ

İsmail Hakkı KINALIOĞLU

DOKTORA TEZİ

İstatistik Anabilim Dalı

Aralık-2019
KONYA
Her Hakkı Saklıdır

ÖZET

DOKTORA TEZİ

YAPAY ZEKA TABANLI YÖNTEMLER KULLANILARAK FUTBOL MÜSABAKALARININ SONUÇLARININ KESTİRİLMESİ VE HİBRİT MODEL ÖNERİLERİ

İsmail Hakkı KINALIOĞLU

Selçuk Üniversitesi Fen Bilimleri Enstitüsü
İstatistik Anabilim Dalı

Danışman: Prof. Dr. Coşkun KUŞ

2019, 189 Sayfa

Jüri
Prof. Dr. Hamza EROL
Prof. Dr. Coşkun KUŞ
Doç. Dr. İsmail KINACI
Dr. Öğr. Üyesi Ahmet PEKGÖR
Dr. Öğr. Üyesi Yunus AKDOĞAN

Bu tez çalışmasında, yapay zeka tabanlı sınıflandırma ve kümeleme yöntemleri kullanılarak futbol müsabakalarının sonuçları kestirilmiştir. Çalışma verisini, Türkiye, İngiltere, İspanya, Almanya, İtalya ve Fransa'nın en üst düzey futbol liglerinde, 2014-15, 2015-16 ve 2016-17 sezonlarında oynanmış olan toplam 6396 müsabakaya ilişkin nitel ve nicel bilgiler oluşturmaktadır. Bu bilgilerin dağınık olarak yer aldıkları web platformlarından toplanabilmeleri için çeşitli veri toplama araçları geliştirilmiştir. Sonrasında ise geliştirilen bir yazılım aracılığıyla, bu ham veriler üzerinde çeşitli hesaplamalar ve filtrelemeler yapılarak eğitim ve test çalışmalarında kullanılmak üzere alternatif veri setleri oluşturulmuştur. Sınıflandırma için çok terimli lojistik regresyon, destek vektör makinesi, *k* en yakın komşu, rasgele orman, sade bayes ve yapay sinir ağları yöntemlerinden yararlanılmış ve bu yöntemler ile elde edilen bulgular bazı performans ölçütlerine göre karşılaştırılmıştır. Sınıflandırma yöntemlerine ek olarak, *k* ortalamalar ve bulanık *c* ortalamalar kümeleme yöntemlerinin de kullanılmasıyla birlikte hibrit modeller geliştirilmiştir. Geliştirilen hibrit modellerin kestirim başarısına olan etkileri araştırılarak elde edilen bulgular paylaşılmıştır. Son olarak ise çalışma sürecinde elde edilen deneyimler, çıkarılan sonuçlar ve öneriler paylaşılarak sonraki süreçte yapılacak çalışmalar hakkında bilgiler verilmiştir.

Anahtar Kelimeler: Çoklu Sınıflandırma, Futbol İstatistikleri, Hibrit Sınıflandırıcı, Kümeleme, Maç Sonucu Kestirimi, Makine Öğrenmesi, Sınıflandırma, Yapay Zeka

ABSTRACT

Ph.D THESIS

PREDICTION OF FOOTBALL MATCH RESULTS BY USING ARTIFICIAL INTELLIGENCE BASED METHODS AND PROPOSALS ON HYBRID MODEL

İsmail Hakkı KINALIOĞLU

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF SELÇUK UNIVERSITY
THE DOCTOR OF PHILOSOPHY
IN STATISTICS

Advisor: Prof. Dr. Coşkun KUŞ

2019, 189 Pages

Jury
Prof. Dr. Hamza EROL
Prof. Dr. Coşkun KUS
Assoc. Prof. Dr. İsmail KINACI
Assist. Prof. Dr. Ahmet PEKGÖR
Assist. Prof. Dr. Yunus AKDOĞAN

In this thesis, the results of football matchs are predicted by using classification and clustering methods based on artificial intelligence. The data of the study consists of qualitative and quantitative information of 6396 matches played in 2014-15, 15-16 and 16-17 seasons in Turkey, England, Spain, Germany, Italy and France top football leagues. Various data scraping tools are developed in order to collect this information from scattered web platforms. Afterwards, various calculations and filtering are performed on these raw data by means of developed software and alternative data sets are created to be used in the training and testing process. Multinomial logistic regression, support vector machine, k nearest neighbor, random forest, naive bayes and artificial neural network methods are used for classification and the results obtained by these methods are compared according to some performance measures. Then, in addition to classification methods, hybrid models are developed by using k means and fuzzy c means clustering methods. The effects of these hybrid models on accurate prediction rate are investigated and the findings are shared. Finally, the experiences, conclusions and suggestions obtained from during the study process are shared and information is given about the studies to be carried out in the next process.

Keywords: Multiple Classification, Football Statistics, Hybrid Classifier, Clustering, Match Prediction, Machine Learning, Classification, Artificial Intelligence

Batın Çınar ve Elvin Güneş'e…

ÖNSÖZ

Futbol ve istatistik bilimi arasında kökleri oldukça eskiye dayanan güçlü bir ilişki vardır. Uzun yıllardan beri saha içi ve saha dışından gerek takımlar gerekse de oyuncular hakkında bilgiler veren çok sayıda veri toplanmaktadır. Teknolojik yeniliklerin de etkisiyle birlikte toplanan bu verilerin miktarı her geçen gün katlanarak artmaktadır. Veri miktarında meydana gelen artış bir yandan bu veriler ile gerçekleştirilen çalışmaların sayısının artmasına neden olurken, bir yandan da futbol verileri üzerinde yıllardır kullanılan geleneksel istatistiksel yöntemlere ek olarak yeni alternatiflerin ortaya çıkmasına neden olmuştur. Futbolun yapay zeka ile tanışması da tam olarak bu noktada gerçekleşmiştir. Geleneksel istatistiksel yöntemler ile gerçekleştirilemeyen veya uzun süren analiz ve modellemeler gelişen bilişim teknolojilerinin de katkısıyla çok daha işlevsel yöntemler ile rahatlıkla gerçekleştirilmeye başlanmıştır. Bu çalışmada ise yapay zekaya dayalı yöntemler, futbol müsabakalarının sonuçlarının kestirilmesinde kullanılarak, futbol ve istatistik arasındaki ilişkiye farklı bir pencereden bakılmıştır. Ulaşılan bulgular yapay zeka tabanlı yöntemlerin futbol verileri üzerinde gerçekleştirilen çalışmalarda başarılı birer alternatif olabileceklerini göstermiştir.

Fikir aşamasından tamamlanmasına kadar yaklaşık olarak üç yıl süren bu çalışma boyunca birçok kişinin desteğini hep yanımda hissettim. Her ne kadar onlara olan minnettarlığımı birkaç satırla ifade edemeyecek olsam da yazmış olduğum bu önsözü isimleri ile onurlandırmaktan gurur duyuyorum.

Öncelikle, lisansüstü eğitimime başladığım ilk günden doktora çalışmamın tamamlanmasına kadar geçen süreçte ihtiyaç duyduğum her anda tüm enerjisiyle yanımda olan, gerek akademik gerekse de sosyal anlamda çok değerli katkılar sunan ve bu tez çalışmasının şekillenmesine rehberlik eden sevgili danışmanım Prof. Dr. Coşkun KUŞ'a bana gösterdiği sabır, verdiği emek ve ayırdığı zaman için sonsuz teşekkür ederim.

Yine bu süreçte son derece destekleyici eleştiriler ile ufkumu genişleten ve tez çalışmamın her aşamasında çok değerli katkıları bulunan sevgili hocalarım Doç. Dr. İsmail KINACI, Dr. Öğr. Üyesi Ahmet PEKGÖR, Dr. Öğr. Üyesi M. Kemal KARAMAN ve Dr. Adem KARATAŞ'a, tez çalışmam boyunca her türlü yardım talebime olumlu dönütler vererek iş yükümü azaltan değerli çalışma arkadaşlarıma ve eğitim hayatım boyunca bana kattıkları her şey için sevgili ilkokul öğretmenim Necati MOLLAVEİSOĞLU nezdinde tüm hocalarıma sonsuz şükranlarımı sunarım.

Ayrıca bu zorlu sürecin her aşamasında bana zaman yaratmak adına birçok sorumluluğumu üstlenen ve doktora eğitimime odaklanabilmemi sağlayan sevgili eşim İlknur'a, her ihtiyaç duyduğumda yanımda olup bana güç veren annem, babam ve ablama son olarak da hayatıma girdikleri ilk günden itibaren gülüşlerinde huzur bulduğum dünyalar tatlısı çocuklarım Batın Çınar ve Elvin Güneş'e teşekkür ederim.

İsmail Hakkı KINALIOĞLU
KONYA-2019

İÇİNDEKİLER

ÖZET iv

ABSTRACT v

ÖNSÖZ vii

İÇİNDEKİLER viii

SİMGELER VE KISALTMALAR xi

ÇİZELGELER DİZİNİ xiii

ŞEKİLLER DİZİNİ xvi

RESİMLER DİZİNİ xviii

1. GİRİŞ 1

1.1. Tezin Amacı 5
1.2. Tezin Önemi 6
1.3. Tezin Kapsamı 7
1.4. Tezin Metodolojisi ve Organizasyonu 7

2. KAYNAK ARAŞTIRMASI 10

2.1. Çalışmanın Amacına Yönelik Kaynak Araştırması 10
2.2. Kullanılan Yöntemlere Yönelik Kaynak Araştırması 18

3. YAPAY ZEKA VE MAKİNE ÖĞRENMESİ 23

3.1. Yapay Zeka 23
3.2. Makine Öğrenmesi 26
3.3. Makine Öğrenmesinin Aşamaları 26
3.3.1. Problemin Tanımlanması 27
3.3.2. Veri Toplama 27
3.3.3. Veri Ön işleme 27
3.3.4. Boyut İndirgeme 30
3.3.5. Modelleme 32
3.3.6. Test ve Değerlendirme 32
3.4. Öğrenme Stratejileri 38
3.4.1. Gözetimli Öğrenme 39
3.4.2. Gözetimsiz Öğrenme 40
3.4.3. Yarı Gözetimli Öğrenme 41
3.4.4. Pekiştirmeli Öğrenme 41

4. KULLANILAN SINIFLANDIRMA VE KÜMELEME YÖNTEMLERİ 43

4.1. Sınıflandırma 43
4.1.1. Yapay Sinir Ağları 44
4.1.2. Çokterimli Lojistik Regresyon 49

4.1.3. Destek Vektör Makineleri 51
4.1.4. K En Yakın Komşu Algoritması 54
4.1.5. Sade Bayes 56
4.1.6. Rasgele Orman 59
4.2. Kümeleme 62
4.2.1. *K*-Ortalamalar Kümeleme 63
4.2.2. Bulanık *C* Ortalamalar Kümeleme 64

5. VERİ SETİNİN OLUŞTURULMASI 66

5.1. Veri Setinin Kapsamı 66
5.2. Veri Toplama Süreci 68
5.2.1. Maç Günü Verilerinin Toplanması 68
5.2.2. Saha İçi İstatistiklerin Toplanması 75
5.2.3. Eksik ve Hatalı Verilerin Tamamlanması 81
5.3. Veri Hesaplamaları ve Veri Setinin Hazırlanması 81
5.3.1. Veri Hesaplama Uygulaması 82
5.3.2. Veri Setinin Yapısı 83
5.3.3. Takımların Saha İçi Performanslarının Hesaplanması 85
5.3.4. Yeni Değişkenlerin Türetilmesi 91
5.3.5. Veri Setinin Hazırlanması 93
5.3.6. Veri Normalleştirme 96
5.3.7. Özellik Seçimi ve Özellik Çıkarımı 96

6. HİBRİT YÖNTEMLERİN YAPISI 98

7. TEST VE DEĞERLENDİRME SÜRECİ 101

7.1. Uygulama Sürecinin Planlanması 101
7.2. Genelleştirilmiş Performans Ölçütleri 102

8. BULGULAR 105

8.1. Sınıflandırma Yöntemlerine İlişkin Bulgular 105
8.1.1. MS-1 (G,B,M) Durumunda Ulaşılan Bulgular 107
8.1.2. MS-2 (GB,M) Durumunda Ulaşılan Bulgular 115
8.1.3. MS-3 (G,BM) Durumunda Ulaşılan Bulgular 121
8.2. KO Kümeleme ile Oluşturulan Hibrit Modellerin Bulguları 127
8.2.1. MS-1 (G,B,M) İçin KO Hibrit Yöntemler 127
8.2.2. MS-2 (GB,M) İçin KO Hibrit Yöntemler 129
8.2.3. MS-3 (G,BM) İçin KO Hibrit Yöntemler 131
8.3. BCO Kümeleme İle Oluşturulan Hibrit Modellerin Bulguları 132
8.3.1. MS-1 (G,B,M) için BCO Hibrit Yöntemler 133
8.3.2. MS-2 (GB,M) İçin BCO Hibrit Yöntemler 135
8.3.3. MS-3 (G,BM) İçin BCO Hibrit Yöntemler 136
8.4. Ortalama Başarılara ilişkin Bulgular 138
8.5. Veri Setinin İçeriğine İlişkin Bulgular 139

9. SONUÇLAR VE ÖNERİLER 141

9.1. Sonuçlar 141

9.2. Öneriler ve Sonraki Çalışmalar 145

KAYNAKLAR 148

EKLER 172

ÖZGEÇMİŞ 187

SİMGELER VE KISALTMALAR

Simgeler

Y : Bağımlı Değişken
X : Bağımsız Değişken
G : Galibiyet
B : Beraberlik
M : Mağlubiyet

Kısaltmalar

ACC : Doğruluk (Observed Accuracy)
BCO : Bulanık C Ortalamalar (Fuzzy-C Means)
ÇLR : Çok Terimli Lojistik Regresyon (Multinomial Logistic Regression)
DVM : Destek Vektör Makinesi (Support Vector Machine)
EACC : Beklenen Doğruluk (Expected Accuracy)
FNR : FN Rate
FPR : FP Rate (1-Belirleyicilik):
KEYK : K En Yakın Komşu (K Nearest Neighbor)
KO : K Ortalamalar (K-Means)
KS : Kappa İstatistiği (Kappa Statistics)
MS-1 : Maç Sonucu 1 – (Galibiyet, Beraberlik, Mağlubiyet)
MS-2 : Maç Sonucu 2 – (Galibiyet-Beraberlik, Mağlubiyet)
MS-3 : Maç Sonucu 3 – (Galibiyet, Beraberlik-Mağlubiyet)
NPV : Negatif Kestirim Değeri
PPV : Kesinlik (Precision)
RO : Rasgele Orman (Random Forest)
SB : Sade Bayes (Naive Bayes)
TNR : Belirleyicilik (Specificity)
TPR : Duyarlılık (Sensitivity)
YSA : Yapay Sinir Ağları (Artificial Neural Network)
YZ : Yapay Zeka (Artificial Intelligence)
AUC **:** Eğri Altında Kalan Alan **(**Area Under Curve)
KO-YSA : K-Ortalamalar ve Yapay Sinir Ağları Hibrit Sınıflandırıcı
KO-RO : K-Ortalamalar ve Rasgele Orman Hibrit Sınıflandırıcı
KO-DVM : K-Ortalamalar ve Destek Vektör Makinesi Hibrit Sınıflandırıcı

KO-KEYK : K-Ortalamalar ve K-En Yakın Komşu Hibrit Sınıflandırıcı

KO-LR : K-Ortalamalar ve Lojistik Regresyon Hibrit Sınıflandırıcı

KO-SB : K-Ortalamalar ve Sade Bayes Hibrit Sınıflandırıcı

BCO-YSA : Bulanık C Ortalamalar ve Yapay Sinir Ağları Hibrit Sınıflandırıcı

BCO-RO : Bulanık C Ortalamalar ve Rasgele Orman Hibrit Sınıflandırıcı

BCC-DVM : Bulanık C Ortalamalar ve Destek Vektör Makinesi Hibrit Sınıflandırıcı

BCO-KEYK : Bulanık C Ortalamalar ve K-En Yakın Komşu Hibrit Sınıflandırıcı

BCO-LR : Bulanık C Ortalamalar ve Lojistik Regresyon Hibrit Sınıflandırıcı

BCO-SB : Bulanık C Ortalamalar ve Sade Bayes Hibrit Sınıflandırıcı

ÇİZELGELER DİZİNİ

Çizelge 3.1. Karışıklık matrisi gösterimi 35
Çizelge 4.1. KEYK de kullanılacak örnek veri 55
Çizelge 4.2. Adım – 1'de elde edilen uzaklık değerleri 56
Çizelge 5.1. Liglere ve sezonlara göre ele alınan maç sayıları 67
Çizelge 5.2. Verilere eklenen bahis seçenekleri 72
Çizelge 5.3. Maç günü verilerinin içeriği 73
Çizelge 5.4. Taraftar görüşleri verisinin içeriği 75
Çizelge 5.5. Saha içi genel performans verileri 78
Çizelge 5.6. Saha içi şut performansı verileri 79
Çizelge 5.7. Saha içi pas performansı verileri 79
Çizelge 5.8. Türetilen değişkenler 91
Çizelge 5.9. Boyutu azaltılmış yeni veri seti 97
Çizelge 7.1. Çalışmanın genel karışıklık matrisi yapısı 102
Çizelge 7.2. Beraberliğe göre genelleştirilmiş karışıklık matrisi yapısı 102
Çizelge 7.3. Galibiyete göre genelleştirilmiş karışıklık matrisi yapısı 103
Çizelge 7.4. Mağlubiyete göre genelleştirilmiş karışıklık matrisi yapısı 103
Çizelge 8.1. MS-1 Model parametreleri 107
Çizelge 8.2. MS-1 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri .. 108
Çizelge 8.3. MS-1 için karışıklık matrislerinin özeti 108
Çizelge 8.4. MS-1 için performans ölçütleri 109
Çizelge 8.5. MS-1 için holdout veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları 111
Çizelge 8.6. MS-1 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları 111
Çizelge 8.7. MS-1 için Boyut azaltma işleminin kestirim başarısına olan etkisi 115
Çizelge 8.8. MS-2 Model parametreleri 115
Çizelge 8.9. MS-2 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri .. 116
Çizelge 8.10. MS-2 için karışıklık matrislerinin özeti 116
Çizelge 8.11. MS-2 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri 117
Çizelge 8.12. MS-2 için holdout veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları 118
Çizelge 8.13. MS-2 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları 118

Çizelge 8.14. MS-2 için boyut azaltma işleminin kestirim başarısına olan etkisi 120
Çizelge 8.15. MS-3 Model parametreleri 121
Çizelge 8.16. MS-3 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri 122
Çizelge 8.17. MS-3 için karışıklık matrislerinin özeti 122
Çizelge 8.18. MS-3 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri 123
Çizelge 8.19. MS-3 için holdout veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları 124
Çizelge 8.20. MS-3 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları 124
Çizelge 8.21. MS-3 için boyut azaltma işleminin kestirim başarısına olan etkisi 126
Çizelge 8.22. KO kümeleme ile oluşturulan hibrit modellerde kullanılan veri setleri . 127
Çizelge 8.23. MS-1 için KO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri 127
Çizelge 8.24. MS-1 için KO kümeleme ile elde edilen karışıklık matrislerinin özeti .. 128
Çizelge 8.25. MS-1 için KO hibrit yöntemlerle ulaşılan performans ölçütleri..... 128
Çizelge 8.26. MS-1 için KO hibrit yöntemlerle kestirim başarısında yaşanan değişim128
Çizelge 8.27. MS-2 için KO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri 129
Çizelge 8.28. MS-2 için KO kümeleme ile elde edilen karışıklık matrislerinin özeti .. 129
Çizelge 8.29. MS-2 için KO hibrit yöntemlerle ulaşılan performans ölçütleri..... 130
Çizelge 8.30. MS-2 için KO hibrit yöntemlerle kestirim başarısında yaşanan değişim130
Çizelge 8.31. MS-3 için KO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri 131
Çizelge 8.32. MS-3 için KO kümeleme ile elde edilen karışıklık matrislerinin özeti .. 131
Çizelge 8.33. MS-3 için KO hibrit yöntemlerle ulaşılan performans ölçütleri..... 131
Çizelge 8.34. MS-3 için KO hibrit yöntemlerle kestirim başarısında yaşanan değişim132
Çizelge 8.35. BCO kümeleme ile oluşturulan hibrit modellerde kullanılan veri setleri 133
Çizelge 8.36. MS-1 için BCO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri 133
Çizelge 8.37. MS-1 için BCO kümeleme ile elde edilen karışıklık matrislerinin özeti 133
Çizelge 8.38. MS-1 için BCO hibrit yöntemlerle ulaşılan performans ölçütleri 134
Çizelge 8.39. MS-1 için BCO hibrit yöntemlerle kestirim başarısında yaşanan değişim 134

Çizelge 8.40. MS-2 için BCO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri 135
Çizelge 8.41. MS-2 için BCO kümeleme ile elde edilen karışıklık matrislerinin özeti 135
Çizelge 8.42. MS-2 için BCO hibrit yöntemlerle ulaşılan performans ölçütleri 135
Çizelge 8.43. MS-2 için BCO hibrit yöntemlerle kestirim başarısında yaşanan değişim 136
Çizelge 8.44. MS-3 için BCO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri 136
Çizelge 8.45. MS-3 için BCO kümeleme ile elde edilen karışıklık matrislerinin özeti 137
Çizelge 8.46 MS-3 için BCO hibrit yöntemlerle ulaşılan performans ölçütleri 137
Çizelge 8.47. MS-3 için BCO hibrit yöntemlerle kestirim başarısında yaşanan değişim 137
Çizelge 8.48. Sınıflandırma yöntemleri ile ulaşılan ortalama kestirim başarıları 138
Çizelge 8.49. KO kümeleme yöntemi ile hibritlenmiş sınıflandırma yöntemlerinin ortalama kestirim başarıları 138
Çizelge 8.50. BCO kümeleme yöntemi ile hibritlenmiş sınıflandırma yöntemlerinin ortalama kestirim başarıları 139

ŞEKİLLER DİZİNİ

Şekil 1.1. Çalışmanın metodolojisi. 8
Şekil 3.1. Yapay Zeka'nın alt alanları 25
Şekil 3.2. Makine öğrenmesinin aşamaları 26
Şekil 3.3. Veriler arasındaki aykırı gözlemler 28
Şekil 3.4. Makine öğrenmesinde modelleme yapısı 32
Şekil 3.5. Holdout yönteminde eğitim ve test küme örnekleri 33
Şekil 3.6. Çapraz geçerlilik yönteminde eğitim ve test verisinin değişimi 33
Şekil 3.7. Tabakalı örnekleme yönteminde eğitim ve test verisi 34
Şekil 3.8. Üçlü ayırma yönteminde eğitim, test ve doğrulama verisi 34
Şekil 3.9. Bootstrap yönteminde eğitim ve test verisi oluşturma 35
Şekil 3.10. ROC eğrisi çizimi 38
Şekil 3.11. Makine öğrenmesinde öğrenme türleri 39
Şekil 3.12. Gözetimli öğrenme çalışma mantığı 40
Şekil 3.13. Gözetimsiz öğrenme örneği 40
Şekil 3.14. Yarı gözetimli öğrenme çalışma mantığı 41
Şekil 3.15. Pekiştirmeli öğrenmenin yapısı 42
Şekil 4.1. Kullanılacak makine öğrenmesi yöntemleri 43
Şekil 4.2 DVM'de doğrusal ayrılabilme durumu 53
Şekil 4.3. Bayes ağı modeli 57
Şekil 4.4. Kümeleme Yöntemleri 62
Şekil 5.1. Takımların geçmiş performanslarını hesaplanma süreci 89
Şekil 7.1. Kestirim başarısını etkileyen faktörler ve ilişkisel yapılar 101
Şekil 8.1. MS-1 için paylaşılan bulguların başlıkları 106
Şekil 8.2. Sınıflandırma yöntemlerinin eğitim süreleri 106
Şekil 8.3. MS-1 için holdout veri seti ayırma yönteminde ulaşılan kestirim başarıları 110
Şekil 8.4. MS-1 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan kestirim başarıları 110
Şekil 8.5. MS-1 için holdout veri seti ayırma yönteminde elde edilen ROC eğrileri ... 112
Şekil 8.6. MS-1 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen ROC eğrileri 113
Şekil 8.7. MS-2 için holdout veri seti ayırma yönteminde ulaşılan kestirim başarıları 117
Şekil 8.8. MS-2 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan kestirim başarıları 118

Şekil 8.9. MS-2 için holdout veri seti ayırma yönteminde ulaşılan ROC eğrileri 119
Şekil 8.10. MS-2 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan ROC eğrileri 119
Şekil 8.11. MS-3 için holdout veri seti ayırma yönteminde ulaşılan kestirim başarıları 123
Şekil 8.12. MS-3 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan kestirim başarıları 124
Şekil 8.13. MS-3 için holdout veri seti ayırma yönteminde ulaşılan ROC eğrileri 125
Şekil 8.14. MS-3 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan ROC eğrileri 125
Şekil 8.15. MS-1 için çıkarılan veri seti parçalarının kestirim başarısına etkisi 139
Şekil 8.16. MS-1 için veri setinin her bir parçası ile elde edilen kestirim başarısı..... 139
Şekil 8.17. MS-2 için çıkarılan veri seti parçalarının kestirim başarısına etkisi 140
Şekil 8.18. MS-2 için veri setinin her bir parçası ile elde edilen kestirim başarısı..... 140
Şekil 8.19. MS-3 için çıkarılan veri seti parçalarının kestirim başarısına etkisi 140
Şekil 8.20. MS-3 için veri setinin her bir parçası ile elde edilen kestirim başarısı..... 140

RESİMLER DİZİNİ

Resim 4.1. Biyolojik ve yapay sinir hücrelerinin yapısı 45
Resim 4.2. Geri yayılım algoritmasının yapısı 47
Resim 4.3. Rasgele orman sınıflandırıcısının çalışma mantığı 60
Resim 5.1. Veri toplama süreci 68
Resim 5.2. İlk 11 kadro bilgisi 69
Resim 5.3. İlk 11 piyasa değerleri 69
Resim 5.4. İddaa oranları 70
Resim 5.5. Taraftar anket sonuçları-1 70
Resim 5.6. Taraftar anket sonuçları-2 70
Resim 5.7. Sakatlık ve ceza bilgileri 71
Resim 5.8. Maç günü verilerinin içeriği 71
Resim 5.9. Saha içi performans istatistikleri-1 76
Resim 5.10. Saha içi performans istatistikleri-2 76
Resim 5.11. Saha içi performans istatistikleri-3 77
Resim 5.12. Saha içi performans istatistikleri-4 77
Resim 5.13. Saha içi performans verisi çekme uygulaması 80
Resim 5.14. Maçlar tablosunun içeriği 81
Resim 5.15. Veri seti hazırlama uygulaması 82
Resim 5.16. Çalışma veri setinin yapısı 84
Resim 5.17. Resim Önceki Maçların Veri aralığı 85
Resim 5.18. Takım verisi karşılaştırma 90
Resim 5.19. Veri seti içeriği 94
Resim 5.20. Eğitim dosyası çıktı seçenekleri 95
Resim 5.21. Lig ve sezon filtre seçenekleri 95
Resim 5.22. Resim Eğitim dosyası kayıt seçenekleri 96
Resim 6.1. K-ortalamalar kümeleme ile elde edilen hibrit yöntemlerin yapısı 98
Resim 6.2. Bulanık C ortalamalar kümeleme ile oluşturulan hibrit yöntemlerin yapısı 99

1. GİRİŞ

Sözlükte "bedeni ve zihni geliştirmek amacıyla bireysel ya da toplu olarak gerçekleştirilen aktiviteler" şeklinde tanımlanan (TDK, 2019) ve dilsel kökenini Fransızcadan alan *"spor"* kavramı yüzyıllar boyunca, ırk, dil, din fark etmeksizin tüm uygarlıkların ortak değerlerinden birisi olmuştur. Tarihsel araştırmalar, spor kavramını, insanoğlunun çeşitli amaçlar doğrultusunda gerçekleştirdiği aktivitelere aynı zamanda oyun ve eğlence misyonunun da yüklenmesi olarak açıklarlar. Bu oyunların kimisi günlük aktivitelerden yola çıkarak türetilirken kimisi ise ait oldukları dönemin eğlence anlayışına göre şekillenmiştir. Örneğin eski çağlarda yaşamı sürdürebilmenin bir gerekliliği olan, avlanma, ok-mızrak atma, at sürme gibi aktiviteler günlük hayatın bir parçası olmalarının yanı sıra sportif faaliyetler şeklinde de gerçekleştirilmekteydiler. Yüzyıllar içerisinde pek çok sportif aktivite belli coğrafyalardan doğarak sporun evrenselliğine paralel olarak uygarlıklar arasında yayılmıştır. Bu aktiviteler zamanla her biri kendi kurallarına sahip olan oyunlar olarak evrensel değerler haline gelmişlerdir.

İnsanlık gelişip, büyük ve modern medeniyetler inşa edildikçe yaşamın tüm alanlarında meydana gelen paradigma değişimlerinden spor da nasibini almıştır. Önceleri, genel olarak bireysel bazda gerçekleştirilen küçük çaplı oyunlardan ibaret olan sportif faaliyetler, zamanla büyük kitleler tarafından takip edilen ve yüksek boyutlu ekonomik yatırımların yapıldığı profesyonel spor dalları halini almışlardır.

Bireysel olarak ya da takımlar halinde oynanabilen pek çok spor dalından bahsedebiliriz. Her birinin kendine has kuralları olan bu spor dallarından bazıları popülerlik açısından diğerlerinin çok önündedir. Bu popülerliği belirleyen pek çok unsur vardır. Bazı spor dalları sadece doğdukları coğrafyalarda kayda değer bir popülerliğe sahipken bazı spor dallarının popülerlik sınırları dünyanın geneline yayılmıştır. Örneğin ülkemizde güreş, Avustralya'da ragbi son derece popüler spor dalları iken futbol, basketbol, atletizm gibi spor dalları ise dünya genelinde bir popülariteye sahiptirler.

Sahip oldukları popülarite sayesinde adeta evrensel birer değer haline gelmiş olan bu spor dalları içerisinde de futbol diğerlerine oranla en popüler olanıdır (Caraffa ve ark., 1996; Heidt ve ark., 2000; Houston ve Wilson, 2002; Junge ve Dvorak, 2004; Decrop ve Derbaix, 2010). Futbolun diğer spor dallarına oranla bu denli popüler olmasının sebebi basitliğidir (Bromberger, 1998). Futbol sokak arasında, parkta, bahçede, bir evin koridorlarında yani kısacası her yerde, oyuncu sayısı ya da sınırı olmaksızın, ulaşılması zor ekipmanlar gerektirmeden rahatlıkla oynanabilir (Boniface, 2007).

Ekonomik ve kültürel olarak her geçen gün daha da büyüyen futbolu tanımlamak adına pek çok ifade kullanılmıştır. Bu ifadelerin en net olanı "Halkın Oyunu" dur (Morrow, 2003). Bir spor dalını "halkın oyunu" şeklinde tanımlamak, koşulsuz şartsız toplumun her bireyinin ona ulaşabildiği anlamına gelmektedir. Halk ile özdeşleşen bir spor dalının da bu denli popüler olması ve milyarlarca insanı ardından sürüklemesi ise son derece normaldir. Futbol üzerine yazılmış pek çok romantik hikayede, ona duyulan aşk, belli toplumlar için ifade ettiği değer ve kimileri için bir yaşam şekline nasıl dönüştüğü görülebilir.

Futbol endüstrisindeki büyümenin bir sonucu olarak 1990'ların başından itibaren pek çok ülkede profesyonel futbolda ekonomik anlamda önemli dönüşümler yaşanmaya başladı (Dobson ve ark., 2001). Bu dönüşümler ile birlikte futbol ekonomisinin büyüklüğü yıldan yıla katlanarak bugünkü devasa boyutuna ulaşmıştır. Her geçen gün futbol ve para kavramlarının daha da iç içe geçmesi izleyiciler için olmasa da oynayanlar, yönetenler ve bir şekilde futboldan gelir elde edenler için futbola olan bakış açısının değişmesine neden olmuştur.

Günümüzde futbolcu satışları için ödenen bonservis bedelleri, yıllık maaş miktarları, menajerlik ücretleri, takımların gelir-giderleri ve futbol temelli diğer tüm maliyet ve kazançlar bundan birkaç yıl öncesi ile kıyaslandığında öngörülmesi zor boyutlara ulaşmışlardır. Yakın geçmişte gerçekleştirilen bir araştırmaya göre, 2017-18 sezonunda Avrupa futbolunun market büyüklüğü 28.4 milyar avroya ulaşmıştır (Jones ve Green, 2019).

Futbolun ulaşmış olduğu kültürel ve ekonomik büyüklüğün korunması ve daha da artırılması amacıyla birçok çalışma gerçekleştirilmektedir. Bu çalışmalara örnek olarak, futbolu daha geniş kitlelere yayma, oyuncuların performans yönetimini planlama, takımların gelirlerini artırma, futbolun marka değerini yükseltme vb. amaçlarla gerçekleştirilen çeşitli faaliyetler verilebilir. Günümüzde, büyük Avrupa kulüplerinin Uzak Doğu ülkelerinde gerçekleştirdiği kamplar, 2022 dünya kupasının Katar'da düzenlenecek olması, yüzbinlerce Euro değerindeki performans ölçüm cihazlarının ve antrenman araçlarının temin edilmesi, kulüplerin tesisleşme ve altyapı giderlerinin artması gibi birçok uygulama, futbol ve onun alt ürünlerinden elde edilecek olan gelirin artırılmasına yönelik faaliyetlere somut birer örnektir. Bu faaliyetlerin gerçekleştirilebilmesi için antrenman bilimi, ekonomi, hukuk, yönetim bilimi, psikoloji, istatistik, mühendislik ve daha birçok bilim dalı ile futbol arasında kuvvetli ilişkiler

kurulmuştur. Bazı alt alanları itibariyle futbol bu bilim dalları ile iç içe geçmiş durumdadır.

İstatistik bilimi de futbolun gelişimine sağladığı katkı bakımından bu bilim dalları arasında en önemlilerinden bir tanesi olarak kabul edilmektedir. Birçok bilim insanı tarafından çeşitli şekillerde tanımlanan (Cooke ve Clarke, 1989; Hahn ve Doganaksoy, 2012; Akdeniz, 2015) istatistik biliminin, futboldaki işlevi eldeki verilerden anlamlı sonuçlar çıkarmaktır. Futbol ile istatistik arasındaki etkileşimin artmasının en önemli nedenlerinden birisi de teknolojik yenilikler ile birlikte verilerin elde ediliş sürecinde yaşanan gelişmelerdir.

Bu gelişmeler neticesinde ortaya çıkan pek çok donanımsal ve yazılımsal veri toplama aracı ile gerek bireysel olarak oyuncuları gerekse de takımların genelini açıklayan yüksek boyutlu verilere rahatlıkla ulaşılabilmektedir. Elde edilen bu verilerin etkin bir şekilde analiz edilmesi önemli bir ihtiyaçtır (Han ve ark., 2011). Bu analizler gerçekleştirilirken doğru yöntem ve tekniklerin kullanılması elde edilecek çıkarımların başarılı olması açısından oldukça önemlidir.

Gerek profesyonel futbolda gerekse diğer birçok spor dalında bu amaca hizmet etmek için Opta, Instat, Stats, Sentio gibi şirketler kurulmuştur. Bu şirketler genel olarak çok büyük miktardaki verilerin toplanması, saklanması ve analiz edilmesi ile ilgilenirler. Çeşitli yazılımlar, veri bankaları, danışmanlık hizmetleri gibi başlıklar altında da profesyonel futbol kulüpleri ile çözüm ortaklıkları gerçekleştirerek futbol endüstrisinde önemli bir ekonomik pay elde ederler.

Günümüzde futbol kulüplerinin bünyelerinde gerek bu şirketlerden alınan verileri gerekse de kendi imkanları ile elde ettikleri verileri analiz ederek anlamlandırabilecek istatistik uzmanları istihdam edilmektedir. Genellikle teknik direktörlerin yetki ve sorumlulukları arasında gerçekleşen bu veri toplama, işleme ve analiz çalışmalarını gerçekleştirmek amacıyla birçok kulüpte çok sayıda uzmandan oluşan birimler kurulmuştur. Bu birimlerin görevleri arasında, kadroda bulunan futbolcuların bireysel performans ölçümlerini analiz etmek, antrenman verilerinden performans değişimleri hakkında bilgiler çıkarmak, takım verilerini analiz ederek mevcut durumu ortaya koymak, rakiplerin istatistiki değerlerini inceleyerek teknik ekibe rakibin artı ve eksi özellikleri hakkında faydalı bilgiler sunmak vb. faaliyetler yer almaktadır.

Tüm bu çalışmaların sağlıklı ve amacına uygun bir şekilde gerçekleştirilebilmesi için eldeki verilerin doğru yöntemlerle analiz edilerek anlamlandırılması gerekir. Bu işlem için ise çeşitli matematiksel ve istatistiksel yöntemler kullanılabilir fakat veri

setlerinin çok büyük boyutlara ulaşması beraberinde çeşitli sorunları da getirmektedir (Özkan, 2016). Veri miktarındaki bu artış, verilerin işlenerek faydalı bilgiye dönüştürülmesi için kullanılacak yeni yöntem ve araçlara ihtiyaç duyulmasına neden olmuştur (Chen ve ark., 1996). Bu nedenle veri miktarının çok yüksek boyutlara ulaştığı futbolda da analiz sürecinde kullanılabilecek istatistiksel yöntemlerin titizlikle belirlenmesi ve değişkenler arası ilişkilerin ustalıkla kurulması gerekmektedir.

Yüksek boyutlardaki veriler üzerinde çalışmak geleneksel araç ve yöntemler ile çözülemeyecek iki zorluğu da beraberinde getirmektedir. Bunlardan ilki, bu verilerin saklanarak ihtiyaç halinde hızlıca geri çağırılabileceği veri tabanı mimarilerinin inşa edilmesi ve donanımsal altyapının oluşturulmasıdır. İkinci zorluk ise yüksek boyutlu veriler üzerinde matematiksel ve istatistiksel hesaplamaların hızlı bir şekilde gerçekleştirilmesine imkan tanıyan hesaplama gücünün sağlanmasıdır. Son yıllarda bilişim teknolojilerinde yaşanan hızlı gelişmeler bu problemlerin üstesinden gelinmesini sağlamıştır. Önceleri eldeki kısıtlı hesaplama gücü nedeniyle gerçekleştirilemeyen analizler, artan hesaplama gücü sayesinde, teorikte aynı matematiksel ve istatistiksel yöntemler kullanılarak rahatlıkla gerçekleştirilebilmektedir. Yapay zeka (YZ) temelli makine öğrenmesi yöntemleri de bu sürecin en önemli kazançlarından birisi olarak karşımıza çıkmaktadır.

Son dönemlerde oldukça popüler bir kavram haline gelen YZ; temelde öğrenme, karar verme, problem çözme, ilişki kurma vb. insani kabiliyetlerin elektronik sistemler tarafından taklit edilmesi olarak tanımlanabilir. Makine öğrenmesi ise YZ'nin bir alt çalışma alanı olup bilgisayar teknolojisinin örnek veriler veya geçmiş deneyimler aracılığı ile başarılı çıkarımlar yapacak şekilde programlanması sürecini ifade etmektedir (Alpaydın, 2017). Makine öğrenmesi daha genel bir biçimde ele alındığında ise psikoloji ve davranış bilimini temel alarak, insanın öğrenme mekanizmasının anlamlandırılması, öğrenme sürecinin sayısal olarak modellenmesi ve yeni öğrenme yöntemlerinin geliştirilmesi çalışmalarının bütünü olarak tanımlanabilir (Yuntian, 2012).

Makine öğrenmesi yöntemlerinin temel amacı gözlenen bir örneklemden gereken bilgi çıkarımını yapmaktır. Bu çıkarım sürecinde, öğrenme algoritmalarının yüksek boyutlu veriler üzerinde uygulanabilmesi için istatistik kuramından, elde edilen modellerin bellekte uygun şekilde gösterilebilmesi ve gerektiğinde kullanılabilmesi için de bilgisayar bilimlerinden yararlanırlar (Alpaydın, 2017).

Birçok alanda tahmin, sınıflandırma, ilişkilendirme, kümeleme, boyut azaltma, özellik çıkarımı vb. amaçlar için kullanılan makine öğrenmesi yöntemleri, son

dönemlerde başta futbol olmak üzere birçok spor dalında da benzer amaçlarla gerçekleştirilmiş çalışmalarda kullanılmaktadır. Futbol üzerine yapılan çalışmalar incelendiğinde, makine öğrenmesine dayalı yöntemlerin, farklı boyut ve yapıdaki veri setlerinden yararlanılarak, skor-sonuç kestirimi, gol sayısı tahmini, bahis oranı belirleme gibi çalışmalarda kullanıldığı gözlemlenmektedir.

Bir futbol müsabakasının sonucu üç farklı şekilde neticelenebilmektedir. Bunlar; "ev sahibi lehine", "beraberlik", "deplasman lehine" şeklinde ifade edilebilirler. Çalışma genelinde bu durumlardan "ev sahibi lehine" olanı 1, "beraberlik" 0, "deplasman lehine" olanı ise 2 olarak değerlendirilmiştir. Böylelikle ortaya üç farklı sınıf etiketi değeri alabilen bir bağımlı değişken çıkmıştır.

$$y = \{1, 0, 2\}$$

Veri setine, sonraki bölümlerde nasıl ve nereden elde edildikleri detaylıca açıklanacak olan bağımız değişkenlerin de eklenmesiyle birlikte aşağıdaki gibi bir veri seti elde edilmiştir.

$$A = (x_1, x_2, ..., x_n, y), \quad n = \text{bağımsız değişken sayısı}, \quad y = \{1, 0, 2\}$$

Veri setinin hazır olmasıyla birlikte, çalışmanın amacı doğrultusunda bağımlı değişken olan "maç sonucu" kestirilmeye çalışılmıştır. Kestirim işlemi bir sınıflandırma problemi şeklinde ele alınmıştır.

Sınıflandırma problemleri veri madenciliği ve makine öğrenmesi alanlarında en sık çalışılan konu başlıklarından birisidir ve geçtiğimiz birkaç on yıl boyunca çok farklı disiplinden araştırmacılar tarafından ele alınmıştır (Aggarwal, 2014). Sınıflandırma kısaca veri setinde yer alan satırların özelliklerine göre sınıflarda gruplandırılmasıdır (Mennis ve Guo, 2009). Literatürde verilerin sınıflandırılması amacıyla kullanılan pek çok yöntem mevcuttur (Weiss ve ark., 1990; Kiang, 2003; Wu ve ark., 2003; Aly, 2005).

1.1. Tezin Amacı

Bu tez çalışmasının öncelikli amacı; bir futbol müsabakasının sonucunun makine öğrenmesi yöntemleri ile kestirilmesidir.

Bu temel amaç doğrultusunda çalışmanın alt amaçları ise:

- Kestirim işlemini bir sınıflandırma problemi olarak ele alıp farklı makine öğrenmesi yöntemlerini kullanarak futbol verileri üzerinde hangi yöntemin daha başarılı olduğunu saptamak,

- Takımların temel performans değerlerini yansıtan değişkenlere (pas sayısı, şut sayısı vs.) ek olarak farklı platformlardan elde edilen yeni değişkenler ile veri setini genişletip kestirim başarısında artış sağlamak,
- Müsabaka öncesi elde edilen veriler üzeride çeşitli hesaplamalar gerçekleştirerek kestirim başarısın yukarıya çekecek yeni değişkenler türetmek,
- Sınıflandırma ve kümeleme amacıyla kullanılan çeşitli makine öğrenmesi yöntemlerinin bir araya getirildiği hibrit modeller oluşturularak kestirim başarısını artırmak,
- Bağımlı değişken olan maç sonucunu, standart yapısı olan üç sınıflı $y = \{G, B, M\}$ halinin dışında, iki sınıflı $y = \{G, BM\}$ ve $y = \{GB, M\}$ şeklindeki yapılarda da ele alarak oluşturulan kestirim modellerinin bu durumlardaki performansını ölçmektir.

1.2. Tezin Önemi

Bu çalışma genel ve özel amaçları itibariyle ülkemizde bir ilk olmasının yanı sıra kullanılan yöntem çeşitliliği ve veri setinin genişliği bakımından da uluslararası çalışmalarla farklılık göstermektedir.

Uluslararası literatürde, benzer amaçlar doğrultusunda gerçekleştirilen çalışmalar incelendiğinde bir sınıflandırma problemi olarak ele alınan müsabaka sonuçlarının kestirilmesinde genellikle tek bir yöntemden faydalanıldığı, kıyaslama amaçlı gerçekleştirilen bazı çalışmalarda ise birden çok yöntemin kestirim başarısına bakıldığı gözlemlenmiştir. Çalışmamızda ise altı farklı sınıflandırma yönteminin performansı karşılaştırılmış sonrasında ise iki farklı kümeleme algoritmasının da kullanılmasıyla birlikte kestirim başarısını artıracak hibrit modeller geliştirilmiştir.

Çalışmamızı benzerlerinden ayıran en önemli özelliklerinden birisi de ulaşılan veri çeşitliliğidir. Benzer çalışmalarda genel olarak takımlar hakkında belli başlı performans istatistikleri kullanılırken, çalışmamızda geçmiş saha içi performans değerleri, sosyal medya verileri, bahis oranları ve taraftar anketleri ve çeşitli hesaplamalar sonucu elde edilmiş birtakım veriler kullanılmıştır.

Kullanılan yöntemler ve ulaşılan veri miktarının dışında çalışmanın futbol eksenindeki önemini değerlendirecek olursak; birçok spor dalında olduğu gibi futbolda da oynanmamış bir müsabakanın sonucunu yüksek bir başarı yüzdesi ile kestirebilen ya da takımların kazanma olasılıkları üzerine tahmini veriler sunabilen bir modelin varlığı oldukça önemlidir. Böyle bir model, maç sonucuna etki eden değişkenlerin saptanması,

takım ve oyuncu performanslarının değerlendirilmesi, verilerin birbirleri ile olan ilişkilerinin açıklanması, futbol temelli bahis faaliyetlerinin gerçekleştirilmesi ve sayısını artırabileceğimiz çok sayıda örnek ile futbolun tüm paydaşlarınca kullanılabilir.

1.3. Tezin Kapsamı

Çalışma kapsamında Türkiye, İngiltere, İspanya, Almanya, İtalya ve Fransa'nın en üst düzey liglerinde oynanan 6396 müsabakayla ilgili nitel ve nicel çok sayıda bilgiye ulaşılmıştır. Belirlenen müsabakalara ait bu bilgilerin dağınık olarak yer aldıkları web platformlarından toplanabilmesi için çalışma kapsamında geliştirilen çeşitli veri toplama araçları kullanılmıştır. Elde edilen tüm bilgiler oluşturulan veri tabanı tablolarında ilgili alanlara kaydedilmiştir. Sonrasında ise yine çalışma kapsamında geliştirilen farklı bir uygulama aracılığıyla da veri tabanı tablolarında yer alan bu bilgiler çalışmada kullanılacak veri setlerine dönüştürülmüştür. Bu dönüştürme sürecinde ise uygulama içerisinde yer alan hesaplama ve filtreleme seçeneklerinin kullanılmasıyla veri setinin farklı alternatifleri oluşturulmuştur.

Oluşturulan bu veri setleri, çalışmanın dördüncü bölümünde detaylı olarak açıklanacak yöntemler aracılığıyla maç sonucunun kestirilmesinde kullanılmıştır. Bu yöntemler; sınıflandırma için, çokterimli lojistik regresyon, destek vektör makinesi, k en yakın komşu, rasgele orman, sade bayes ve yapay sinir ağları, kümeleme için ise k-ortalamalar ve bulanık c ortalamalar yöntemleridir. İlk olarak altı farklı sınıflandırma yönteminin performansları karşılaştırılmış sonrasında bu sınıflandırma yöntemlerine ek olarak iki farklı kümeleme yönteminin kullanılmasıyla birlikte hibrit yöntemler oluşturularak performansları test edilmiştir.

Veri setinin içeriğinde yer alan bağımsız değişkenler temelde üç başlık altında toplanmıştır. Bunlar; "maç günü verileri", "saha içi istatistikleri" ve "türetilen veriler" dir. Geliştirilen hibrit yöntemde bağımlı değişken olan müsabaka sonucunu kestirilmesi için en azından "maç günü verileri" ve "saha içi istatistikleri" başlıklarında toplanan değişkenlerden bir kısmına ihtiyaç vardır. Bu değişkenler sağlandığı sürece geliştirilen model dünya üzerinde oynanan herhangi bir müsabakanın sonucunun kestirilmesinde kullanılabilir.

1.4. Tezin Metodolojisi ve Organizasyonu

Çalışma sürecini, başlangıcından bitişine kadar toplam sekiz aşamaya ayırabiliriz. Çalışmanın ilerleyen bölümlerinde çeşitli ana ve alt başlıklar altında raporlaştırılmış olan bu süreç Şekil 1.1'de infografik olarak paylaşılmıştır.



Şekil 1.1. Çalışmanın metodolojisi.

Bu metodoloji doğrultuda çalışma toplam dokuz ana bölümde raporlaştırılmıştır. Birinci bölümde amacı, önemi ve yapısı hakkında bilgiler verildikten sonra ilerleyen bölümler aşağıda belirtildiği gibi organize edilmiştir:

İkinci bölümde; ilgili araştırmalara değinilmiştir. İlgili araştırmalar bölümü iki başlık altında incelenmiş olup ilk başlıkta, tez çalışmasının amacı ile benzerlik gösteren yayınlar incelenmiştir. İkinci başlıkta ise çalışmamızla yöntem olarak benzerlik gösteren çeşitli çalışmalar incelenmiştir.

Üçüncü bölümde ilk olarak YZ ile ilgili genel bilgiler ve tanımlamalar verilmiştir. Bölümün devamında makine öğrenmesi hakkında açıklayıcı bilgiler ile birlikte öğrenme yöntemlerinden bahsedilmiştir.

Dördüncü bölümde kullanılacak sınıflandırma ve kümeleme yöntemleri hakkında teorik bilgiler verilmiş ve çalışma prensiplerinden bahsedilmiştir.

Beşinci bölümde tezin veri toplama ve hazırlama aşamasında gerçekleştirilen işlemler detaylı bir şekilde anlatılmıştır. Öncelikli olarak veri toplanırken yararlanılan kaynaklardan, toplanan veri türlerinden ve geliştirilen veri toplama araçlarından bahsedilmiştir. Bölümün devamında ise toplanan ham verilerin çalışma kapsamında kullanılacak olan veri setlerine dönüştürülme süreci hakkında detaylı bilgiler verilmiştir.

Altıncı bölümde çalışma kapsamında geliştirilecek olan hibrit yöntemlerin yapılarından bahsedilmiştir.

Yedinci bölümde, test ve değerlendirme sürecinin nasıl organize edildiği hakkında bilgi verilerek gerekli performans ölçütlerinin hesaplanmasında kullanılacak karışıklık matrislerinin çalışma özelindeki yapısından bahsedilmiştir.

Sekizinci bölümde, kullanılan sınıflandırma yöntemlerinin ve geliştirilen hibrit yöntemlerin performansları farklı varyasyonlara göre sınanarak elde edilen bulgular paylaşılmıştır.

Son bölümde ise araştırma sonuçları tartışılarak gerek mevcut problemin çözümüne gerekse de sonraki araştırmalara katkı sağlayacak öneriler üzerinde durulmuştur. Ardından sonraki süreçte çalışmanın devamı niteliğini taşıyacak araştırmalar hakkında bilgiler verilerek bölüm sonlandırılmıştır.

2. KAYNAK ARAŞTIRMASI

Kaynak araştırması bölümü çalışmanın amacına ve bu amaç doğrultusunda kullanılan yöntemlere göre iki bölümde ele alınmıştır. İlk bölümde çalışmanın genel amacı ile benzerlik gösteren yayınlar incelenmiştir. İkinci bölümde ise çalışmanın amacı doğrultusunda faydalanılacak olan yöntemlerin doğrudan veya hibrit yöntemler içerisinde kullanıldığı yayınlar ele alınmıştır.

2.1. Çalışmanın Amacına Yönelik Kaynak Araştırması

Crowder ve ark. (2002) çalışmalarında, Dixon ve Coles (1997) tarafından önerilen bağımsız poisson modelini iyileştirerek İngiltere Futbol Ligi'nde 1992-1997 yılları arasında yer alan 92 takımın oynadığı maçların sonuçlarını kestirmişlerdir. İyileştirdikleri modelde Dixon ve Coles (1997) tarafından önerilen modelin aksine, takımların hücüm ve defans güçlerinin sürekli sabit tutulmaktansa gözlemlenemeyen iki değişkenli bazı rasgele süreçlere göre zaman içerisinde değişiklik gösterebileceğini varsaymışlardır. Bu varsayım doğrultusundan takımların hücüm ve defans güçlerini daha hızlı şekilde hesaplayabilen bir yaklaşım önererek elde ettikleri sonuçları Dixon ve Coles (1997) tarafından elde edilen sonuçlar ile kıyaslamışlardır. Berabere biten hiçbir maçı doğru kestiremeyen modelleri, ev sahibinin kazanması durumunda Dixon ve Coles (1997) yaklaşımının altında (%48 - %49) deplasman takımının kazanması durumunda da üstünde (%39 - %36) bir kestirim başarısı göstermiştir.

Goddard ve Asimakopoulos (2004) çalışmalarında, bir sıralı probit regresyon modeli ile 10 yıllık karma bir veri kullanarak İngiltere futbol liglerindeki maçların sonuçlarını kestirmişlerdir. Veri setini oluşturuken geçmiş maç sonuçlarının yanında, sezon sonu sıralamaları, takımların kupalara katılıp katılmaması ve iki takımın bulunduğu şehirler arası coğrafi mesafe de hesaba katılmıştır. Geliştirdikleri modeli bir cadde bahiscisinden elde ettikleri sabit oranlı bahisler için oynanmış bahislerde test etmişler ve %8'lik fazladan bir getirinin gerçekleşeceğini hesaplamışlardır.

Goddard (2005) çalışmasında, maç çıktılarını modellemek için genel olarak 2 tür çalışmanın varlığından bahsederek atılan ve yenilen gollerin sayısını kestirmenin amaçlandığı ilk tür ile maç sonucunun kestirilmesinin amaçlandığı ikinci tür çalışmalara örnek olacak iki yöntem belirleyerek bunların başarılarını karşılaştırmıştır. Atılan ve yenilen golleri kestirmek için iki değişkenli poisson regresyon, maç sonucunu kestirmek içinse sıralı poisson regresyon kullanmıştır. 25 yıllık İngiltere Premier Lig verilerini kullandığı çalışmasında en başarılı sonuca, gol temelli takım performans kovaryansları

ile maç sonucunun (ev sahibi - berabere - deplasman) kestirilmeye çalışıldığı hibrit bir model ile ulaşmıştır.

Rotshtein ve ark. (2005) çalışmalarında, her iki takımın önceki müsabakalarda elde ettiği sonuçlardan yola çıkarak futbol maçlarının sonuçlarını kestirebilecek hibrit bir model önermişlerdir. Önerdikleri model bulanık olmayan bilgi tabanlarından doğrusal olmayan bağıntılar elde etmeyi amaçlamaktadır. Bulanık terimlerin üyelik fonksiyonlarının parametrelerinin ve kural ağırlıklarının belirlenmesinde genetik ve sinirsel optimizasyon algoritmalarının bir kombinasyonunu kullanmışlardır. Çalışmanın eğitim verisini 1994-2001 yılları arasında Finlandiya liginde oynanan 1056 maç, test verisini ise 1991-1993 yılları arasındaki 350 maç oluşturmaktadır. Farklı mağlubiyet, mağlubiyet, beraberlik, galibiyet ve farklı galibiyet şeklinde 5 farklı karar için genetik ve sinirsel olarak ayrı ayrı hesaplamalar yapmışlar ve sonuçları paylaşmışlardır. Sonuçlara göre 5 karar için genetik hesaplamada %77 ile %87,5 aralığında sinirsel hesaplamada ise %84,1 ile %94,6 aralıklarında değişen kestirim başarısına ulaşmışlardır.

Joseph ve ark. (2006) çalışmalarında maç sonuçlarını kestirmek için uzman yapılandırılmış bir bayes ağı ile bazı diğer makine öğrenmesi yöntemlerini (MC4, karar ağacı, sade bayes ve K en yakın komşu) karşılaştırmışlardır. Tottenham Hotspur Futbol Kulübü'nün 1995 ve 1997 yılları arasında oynadığı maçlar belli periyotlara bölünerek eğitim ve test işlemi gerçekleştirilmiştir. Sonuç olarak uzman bayes ağı periyotların çoğunda makine öğrenmesi yöntemlerine göre daha başarılı sonuçlar vermiştir. Ortalama başarı oranı %59,21'dir.

Aslan ve Inceoglu (2007) çalışmalarında, futbol maçlarının sonuçlarını kestirmede sinir ağlarının modelleme kapasitelerinin etkililiğini ortaya koymayı ve giriş parametlerinin seçimi ile kestirimin doğası üzerine bazı tartışmalar başlatmayı amaçlamışlardır. İki farklı giriş vektör parametresi test etmişlerdir. Başarılı kestirim performansı hakkında bir fikir vermesi amacıyla bu iki yaklaşımı hem birbirleriyle hem de diğer yapay sinir ağı yaklaşımları ve istatistiksel metodlarla kıyaslamışlardır. Kıyasladıkları yaklaşımlar literatürde yer alan elo puanlama (Elo, 1978), gol oran kıyaslama modeli (Jackson, 1990), son altı maç karşılaştırma modeli (Mehrez ve Hu, 1995), Cheng'in hibrit yöntemi (Cheng ve ark., 2003) yaklaşımlarıdır. Önerdikleri yaklaşımların kestirim başarıları %51,29 ve %53,25'dir. Bu değerler ile kıyasladıkları diğer yaklaşımların üzerine çıkmışlardır.

Hvattum ve Arntzen (2010) çalışmalarında, maç sonucunu kestirebilmek için gerekli kovaryansları yaratmak amacıyla Elo (1978) tarafından satranç oyuncularının

gücünü değerlendirmek amacıyla geliştirilen “Elo Puanlama Sistemi” nin kullanımını incelemişlerdir. Elo Puanlama Sistemi’ni sıralı logit regresyon modellerinde kullanılan kovaryansları elde etmek için kullanmışlardır. Elo Puanlama Sistemi’ne dayanan 2 adet kestirim model geliştirerek bu modelleri, istatistiksel ve ekonomik ölçütler kullanılarak elde edilmiş 6 adet kestirim modeli ile kıyaslamışlardır. Sonuç olarak geliştirdikleri kestirim modelleri kıyasladıkları modeller içerisinden 2 tanesinden (AVG, MAX) daha başarısız olurken 4 tanesinden (UNI, FRQ, GOD_B, GOD_G) daha yüksek kestirim başarısına ulaşmışlardır.

Huang ve Chen (2011) çalışmalarında, yapay sinir ağlarından yararlanarak 2006 Dünya Kupası’nın kazananını kestirmişlerdir. Veri setlerini, takımların grup aşamasında oynadıkları maçlarda elde ettikleri değerler oluşturmaktadır. Bu veri setinde takımların gol sayısı, şut sayısı gibi toplamda 8 özellik bulunmaktadır. 2 sınıfa (galibiyet, mağlubiyet) ayırdıkları bağımlı değişkeni kestirmeye çalışan modellerini uzatmaya gitmeyen maçlar için test etmişler ve sonuç olarak ikinci turda %85 (6/7), çeyrek finalde %50 (1/2), yarı finalde %50 (1/2) kestirim başarısı yakalamışlardır. Bunlara ek olarak final maçının kazananını da doğru kestirmişlerdir.

Owen (2011) çalışmasında, takımları niteleyen yeteneklerinin sabit kalmadığını ve zaman içerisinde değiştiğini varsaymaktadır. Bu varsayımdan hareketle geliştirdiği dinamik genelleştirilmiş doğrusal bir model yardımıyla futbol maçlarının sonuçlarını olasılıksal olarak tahmin etmiştir. Çalışmanın veri setini İskoçya Premier Ligi’nde 2003-2004 sezonu ile 2005-2006 sezonu arasında oynanan maçlar oluşturmaktadır. Model sonuçlarını test ettiğinde geliştirdiği modelin dinamik olmayan formlarına göre daha başarılı sonuçlar verdiğini ortaya koymuştur.

Constantinou ve ark. (2012) çalışmalarında, futbol maçlarının sonuçlarının kestirilmesi için “pifootball” adını verdikleri yeni bir bayes ağı modeli sunmuşlardır. Ev sahibi takımın kazanması, beraberlik ve deplasman ekibinin kazanması şeklinde 3 sonuç vardır. Geliştrdikleri modeli eğitimek için İngiltere Premier Liginde 1993-94 ile 2009-10 sezonları arasında oynanan tüm maçları, test etmek için ise 2010-11 sezonunda oynanan toplam 380 maçı kullanmışlardır. Her takım için 4 faktör tanımlamışlardır, bunlar; güç, form, psikoloji ve yorgunluktur. Her bir faktör altında değişen sayılarda özellikler bulunmaktadır. Amaçları nesnel bilgilere ek olarak uzmanlarca belirtilen öznel bilgileri de modele dahil ederek başarı oranını artırmakır. Çalışmanın çıktılarını değerlendirdiklerinde ise bu amaçlarına ulaştıklarını ve öznel bilgilerin modelin başarısını yukarıya çektiğini ortaya koymuşlardır.

Timmaraju ve ark. (2013) çalışmalarında, İngiltere Premier Lig'indeki maçların sonuçlarını kestirmeye çalışmışlardır. Multinomial lojistik regresyon ve destek vektör makinelerini kullandıkları kestirim işleminde en büyük zorluk olarak İngiltere Premier Ligi'nde berabere biten maçların fazla olmasını göstermişlerdir. Modeli eğitimek için 2012-13 sezonunda oynanan, 166'sını ev sahibi takımın, 106'sını deplasman takımının kazandığı ve 108'i berabere biten 380 maçı kullanmışlardır. Veri setlerine, takımları tanımak için gol, korner, kaley, bulan şut gibi performans ölçütlerinin yanında, takımlar arası rekabet, takımların son form durumu, ev sahibi avantajı, takımların birbirine göre performansı, örneğin: a takımının b takımından daha formda olması gibi özellikler ekleyerek problem olarak belirledikleri beraberlik sayısının fazlalığından kaynaklanan kestirim zorluluğunu çözmeye çalışmışlardır. Geliştirdikleri model test aşamasında 2011-2012 sezonu için %52,1 2012-2013 sezonu için ise %48,15'lik bir kestirim başarısına ulaşmıştır.

Ulmer ve ark. (2013) çalışmalarında, İngiltere Premier Ligi'ndeki maçların sonuçlarını kestirmek için YZ ve makine öğrenmesi algoritmalarını kullanmışlardır. Geçmiş verilerden takımların maç günü verilerini ve mevcut performanslarını içeren bir veri seti oluşturmuşlardır. Temelde beş farklı sınıflandırıcı kullanmışlardır, bunlar; stochastic gradient descent , sade bayes, gizli markov modeli, destek vektör makinesi ve rasgele ormandır. Bu sınıflandırıcılarında kendi içinde farklı formlarını da kullanarak toplamda sekiz farklı model ile çalışmışlardır. Normalde üç sınıftan oluşan maç sonucunu, destek vektör makinelerinin iki sınıfı ayırmada kullanılan doğrusal formu için beraberlik sınıfını saf dışı bırakarak "kazanma" ve "mağlubiyet" şeklinde iki sınıfta, diğer modeller için ise "kazanma", "berberlik" ve "mağlubiyet" şeklinde üç sınıfta ele almışlardır. 2002-2003 sezonu ile 2011-2012 sezonu arasındaki verileri eğitim için 2012-2013 ve 2013-2014 sezonlarına ait verileri de test için kullanmışlardır. Gerçekleştirdikleri test işlemleri sonucu en yüksek başarı oranına %67 ile rasgele orman yöntemi ile ulaşmışlardır.

Igiri ve Nwachukwu (2014) çalışmalarında, futbol maçlarının sonucuna etki edebilecek 9 özellik kullanarak maç sonuçlarını kestirebilecek bir model oluşturmuşlardır. Rapid Miner yazılımının ilgili araçları yardımıyla lojistik regresyon ve yapay sinir ağları tekniklerini kullanmışlardır. Çalışmalarının veri setini 2014-2015 sezonunda İngiltere Premier Ligi'nde oynanan maçlardan 110 tanesi oluşturmaktadır. Test işlemleri sonucunda kestirim başarısında yapay sinir ağları ile %85, lojistik regresyon ile %93 oranını yakalamışlardır.

Igiri (2015) çalışmasında, futbol maçlarının sonuçlarının kestirilmesinde destek vektör modellerinin performansının araştırmıştır. Her bir maç için 38 özelliğin ele alındığı bir veri seti oluşturmuştur. 16 veri setinin eğitim amaçlı kullanıldığı çalışmanın test aşamasında 15 maç sonucu vardır. Test işlemi sonucu ulaşılan başarı oranı (8/15) %53,3'tür.

Koopman ve Lit (2015) çalışmalarında, futbol maçlarının sonuçlarını kestirmek ve analiz etmek için zaman içerisinde rasgele olarak değişen yoğunluk katsayılarına sahip iki değişkenli Poison dağılımının kullanıldığı bir model geliştirmişlerdir. Yoğunluk katsayıları takımların hücum ve defans güçlerine dayandırarak ve zaman içerisinde değişiklik göstereceğini varsaymışlardır. 2003-04 sezonu ile 2011-12 sezonu arasında oynanan 9 sezon boyunca İngiltere Premier Ligi'nde mücadele etmiş toplam 36 takımın bu süreçteki maçları verisetine dahil edilmiştir. Her bir takım için haftalık olarak değişen defans ve hücum güçleri hesaplayarak 2010-11 ve 2011-12 sezonları için testler gerçekleştirmişlerdir. Elde ettikleri sonuçları iddaa oyuncularının tahminleri ile karşılaştırdıklarında kazancın yükseldiği sonucuna ulaşmışlardır.

Amadin ve Obi (2015) çalışmalarında, Uyarlamalı Sinirsel Bulanık Denetim Sistem (ANFIS) yaklaşımıyla maç sonuçlarını kestirmeye çalışmışlardır. Veri setlerinde, her iki takımın oynadıkları son 2 maçın sonucu, puan tablosundaki yerleri, takımların popülaritesi ve ev sahibi takım avantajı olmak üzere toplam 5 değişken vardır. Takımların oynadıkları son iki maçı niteleyen değişkeleri "WW, WD, WL, DW, DD, DL, LW, LD, LL" şeklinde yedi sınıfta geri kalan üç değişkeni için ise "highA, same, lowA" şeklinde üç sınıfta kategorik hale getirmişlerdir. Bu beş kategorik bağımsız değişkene, bağımlı değişken olan maç sonucunu da ekleyerek oluşturdukları veri seti ile modellerini eğitmişlerdir. Bu işlem için Matlab Fuzzy Logşc Toolbox kullanarak test ettikleri 7 maçın 5'ini doğru kestirmişler ve %71 başarı oranına ulaşmışlardır.

Robertson ve ark. (2016) çalışmalarında, Avustralya futbolunda, takım performans göstergelerinin maç sonucunu açıklayabilme becerisini ölçmüşlerdir. 2013-2014 yılları arasında 395 adet maçtan toplamda 17 adet performans göstergesine ait veriler toplamışlardır. Göstergelerin birbirleri ile olan ilişkisine bakmak için Spearmen'ın korelasyon katsayısını kullanmışlar ardından da maç sonucunu (galibiyet, mağlubiyet) bağımlı değişken, göstergeleri de bağımsız değişken olarak ele alıp tek yönlü varyans analizi yapmışlardır. Son olarak maç sonucu ile anlamlı ilişki gösteren göstergeleri, doğrusal olasılık modeli geliştirmek için lojistik regresyonda kullanmışlardır. Test aşamasında, birinde göstergeler de boyut azaltma işlemi yaptıkları toplam iki model

geliştirmişlerdir. Tüm göstergelerle yaptıkları testte %87,1, boyut azalttıkları modelde %85,8 kestirim başarısı sağlamışlardır.

Gevaria ve ark. (2015) çalışmalarında, futbol maçlarının sonuçlarını kestirebilmek için çeşitli makine öğrenmesi ve veri madenciliği yöntemlerini kullanmışlardır. Atak ve defans ile ilgili olarak belirledikleri birçok değişkenden bazılarını önemli olarak işaretleyerek bir veri seti oluşturmuşlardır. Bu veri seti üzerindeki, eğitim ve test işlemlerini de WEKA yazılımı aracılığıyla gerçekleştirmişlerdir. Kullandıkları veriyi hangi liglerden elde ettiklerini ve detaylı test sonuçlarını paylaşmadıkları çalışmalarında, lojistik regresyon ile %90'ın üzerinde, diğer yöntemler ile de %47-85 aralığında kestirim başarısına ulaştıklarını ifade etmişlerdir.

Liu ve ark. (2016) çalışmalarında, maç sonuçlarının, maç içi aksiyonlar ve durumsal değişkenler ile olan ilişkisini araştırmışlardır. Maç içi aksiyonlar olarak, şut sayısı, pas sayısı vb. gibi 16 adet değişken, durumsal değişken olarak ise oynanan maçın iç sahada mı, deplasmanda mı olduğu ele alınmıştır. İlişkiyi belirlemek için genelleştirilmiş karma doğrusal modelleri kullanmışlardır. Veri setlerini İspanya 1. Lig'inde 2012-2013 sezonunda oynanan 380 maç oluşturmaktadır. Bu veriler kullanılarak takımların karşılıklı olarak attıkları gol sayıları arasındaki farka göre IBM SPSS programında K-Ortalamalar yöntemi ile kümeleme analizi yapılmıştır. Gol farkının 2 ya da daha büyük olduğu durumu birinci küme, 2'den küçük olduğu durumu ikinci küme olarak değerlendirmişler, gol farkının aykırı değerlerde olduğu 60 maçı da veri setinin dışında tutmuşlardır. Çalışma sonunda topa sahip olmanın maç sonucuna küçük bir etki yarattığına buna karşın, şut sayısı, kaleyi bulan şut sayısı gibi değişkenlerin daha büyük etki yarattığı şeklinde sonuçlara ulaşmışlardır. Diğer değişkenler ile ilgili durumu da tablolaştırarak paylaşmışlardır.

Prasetio ve Dra. Harlili (2016) çalışmalarında maç sonuçlarını kestirebilmek için bir lojistik regresyon modeli oluşturmuşlar ve maçın kazanılması için hangi değişkenlerin önemli olduğunu belirlemişlerdir. Geliştirdikleri model, ev sahibinin kazanması, deplasman takımın kazanması ve beraberlik şeklinde 3 farklı sonuçta kestirim yapabilmektedir. Çalışmalarında diğer çalışmalardan farklı olarak değişkenlerin önemlerini kendileri belirlememiş alandaki benzer araştırmalardan yararlanmışlardır. Aynı zamanda kestirim başarısını yükseltebilmek amacıyla bazı konsol oyunlarından da veriler elde etmişlerdir. İngiltere Premier Ligi'nde 2010-11 ve 2015-16 sezonları arasında oynanan maçları eğitim verisi, 2015-16 sezonunda oynanan maçları ise test verisi olarak kullanmışlardır. Test verisini sabit tutarak, eğitim verisini tüm eğitim verisinin farklı alt

kümeleri şeklinde ele almış ve birden çok test işlemi gerçekleştirmişlerdir. Tüm bu testler sonucunda ulaştıkları en yüksek kestirim başarısı oranı %69,5 iken en düşük başarı oranı ise %68'dir.

Vaidya ve ark. (2016) çalışmalarında, makine öğrenmesi yöntemlerini kullanarak geçmiş verilerle İngiltere Premier Ligi'ndeki maçların sonuçlarını kestirmeye çalışmışlardır. 10 sezonluk bir aralıkta oynanan tüm karşılaşmaları dahil ettikleri çalışmalarında veri setlerini oluştururken, atılan gol sayısı, yenilen gol sayısı, başarılı şut oranı gibi değişkenlerden yararlanmışlardır. Lojistik regresyon, rasgele orman ve sade bayes sınıflandırıcıyı kullandıkları çalışmalarında %47-50 arası başarı oranına ulaşmışlardır.

Lata ve Gupta (2016) çalışmalarında, İngiltere Premier Ligi maçlarının sonuçlarının kestirmek amacıyla YZ ve makine öğrenmesi algoritmalarını kullanmışlardır. Oluşturdukları veri seti, geçmiş verilerden elde edilmiş olan oyuncuların maç günü performanslarını içermektedir. Oyuncu performanslarından 15'er dakikalık aralıklarla ortalama takım performansları elde etmişlerdir. Bu performans değerlerini iki takımı kıyaslamak için kullanmışlardır. Galibiyet, mağlubiyet ve beraberlik için bir kestirim yüzdesi belirlemişlerdir. Gerçekleştirdikleri test işlemi sonucunda ulaştıkları en yüksek kestirim başarısı %41,2'dir.

Martins ve ark. (2017) çalışmalarında, futbol maçlarının sonuçlarının kestirilmesinde yeni bir yaklaşım öne sürmüşlerdir. Kestirim başarısını artırabilmek adına sınıf değişkeni olan maç sonucunu galibiyet-beraberlik, galibiyet-mağlubiyet ve mağlubiyet-beraberlik şeklinde iki durum içeren üç grupta ele almışlardır. İngiltere, İspanya ve Brezilya liglerinde oynanan maçlardan oluşturdukları veri setlerinde bir maç için her iki takıma ait 9 değişken (gol, şut, korner vb.) yer almaktadır. Kestirim işlemi için çok terimli bir sınıflandırma yöntemi geliştirerek bu modeli bazı makine öğrenmesi algoritmaları ile kıyaslamışlar ve kestirim başarısında ulaştıkları %96'lık başarı ile kıyasladıkları yöntemlere göre önemli bir fark elde etmişlerdir.

Nazim ve ark. (2017) çalışmalarında, bayes ağlarını kullanarak İngiltere Premier Ligi'nde oynanan maçların sonuçlarını kestirmeye çalışmışlardır. 2010-11 ile 2012-13 sezonların arasındaki 3 sezonluk veriyi dahil ettikleri çalışmadan verilerin %90'ını eğitim %10'unu test amacıyla kullanmışlardır. Çalışmalarında gerçekleştirdikleri testler sonucunda ortalama %75,09 kestirim başarısına ulaşmışlardır.

Bunker ve Thabtah (2017) çalışmalarında yapay sinir ağlarının spor müsabakalarının sonuçlarını kestirinde kullanılmasına odaklanmışlar ve eleştirel bir

analiz sunmuşlardır. Veri madenciliğinde CRISP-DM olarak adlandırılan iş akış çerçevesine eklemeler yaparak SRP-CRISP-DM (sport result prediction-CRISP-DM) adında yeni bir metodoloji oluşturmuşlardır. CRISP-DM, hangi düzeyde olursa olsun veri madenciliği ile uğraşan herkesin, veri madenciliği projelerini kusursuzca yürütebilmek adına faydalanabileceği son derece kapsamlı bir iş akış çerçevesidir (Shearer, 2000). SRP-CRISP-DM ise bu çerçeveye spor müsabakalarının sonuçlarının kestirilmesine yönelik yapılan iyileştirmeler sonucu yazarların verdiği isimdir. Test aşamasında bu iki çerceveye göre gerçekleştirdikleri uygulamalar neticesinde SRP-CRISP-DM çerçevesinin özelikle spor alanından kaynaklanan kestirim sorunlarının üstesinden gelinebilmesi adına faydalı olacağını savunmuşlardır.

Liti ve ark. (2017) çalışmalarında, bir futbol maçının ilk yarısında toplanan istatistiksel verileri kullanarak, ilk yarı sonucunun "berabere" olduğu maçların sonuçlarını kestirmişlerdir. Aynı zamanda bir maçta elde edilen istatistiklerin hangilerinin maç sonucunu kestirmede daha yararlı olacağını gösterdikleri bu çalışmanın uygulaması İtalya Seria A'da oynanan maçlar üzerinde gerçekleştirilmiştir. Veri setlerini 2014-15 sezonun ikinci yarısı ile 2015-16 sezonun ilk yarısında oynanan maçlar oluşturmaktadır. Bu veri setine ilk olarak gol sayısı, şut sayısı vb. 50 özellik eklemişler ama daha sonra özellik çıkarımı ile bu sayısı 27'ye düşürmüşlerdir. Weka yazılımı aracılığıyla 4 farklı sınıflandırıcı kullanmışlardır, bunlar, sade bayes, 2 farklı tipte destek vektör makinesi ve RBF sınıflandırıcılarıdır. Bu sınıflandırıcılar ile veri setinin özellik seçimi öncesi ve sonrası halleri için ayrı ayrı testler gerçekleştirmişler ve bu örnek özelinde en başarılı sınıflandırıcıların sade bayes ve RBF sıınıflandırıcıları olduğunu belirtmişlerdir.

Prateek ve ark. (2017) çalışmalarında, Ulusal Futbol Ligi maç sonuçlarının kestirilmesinde normal veri diye adlandırdıkları takım ve oyuncuların geçmiş performans değerleri ile twitter verileri arasındaki ilişkiyi araştırmışlardır. Performans verilerini takımların ve oyuncuların önceki müsabakalarda elde ettikleri değerlerden, twitter verisini ise maç saatine göre belirlenen 24 saatlik bir aralıkta paylaşılan twitter gönderilerinden elde etmişlerdir. Önerdikleri modelde bir maçın sonucundan ziyade sezonun kazananını kestirmeyi amaçlamışladır. Toplanan twitleri, sade bayes ile gerçekleştirdikleri duygu analizi sayesinde olumlu, olumsuz ve nötr olarak işaretlemişlerdir. Bu sonuçları normal veri diye adlandırdıkları verilerle harmanlayarak, sadece istatistiksel verileri kullanarak maç sonuçlarını kestirmeye dayanan modellerden daha yüksek bir kestirim başarısı yakaladıklarını belitrmişlerdir.

Che Mohamad Firdaus Che Mohd ve ark. (2018) çalışmalarında, futbol kulüplerinin menajer ve koçlarının bakış açılarına göre futbol maçlarının sonuçlarını kestirmeye çalışmışlardır. Galibiyet, mağlubiyet ve beraberlik olarak üç sınıfta ele aldıkları maç sonucunu kestirebilmek için çeşitli veri madenciliği yöntemleri kullanmışlardır. Çalışmanın verisetini 2013-14 ile 2015-16 sezonları arasındaki toplam 1140 maç oluşturmaktadır. Karar ağaçları, yapay sinir ağları, bayes ağları, k en yakın komşu sınıflandırıcı yöntemlerini kullandıkları çalışmada eğitim ve test işlemleri aynı veri üzerinde gerçekleştirildiğinden ulaşılan kestirim başarısı oranları oldukça yüksektir, en yüksek başarı oranına %99,56 ile karar ağaçları ile ulaşılırken en düşük başarı %76,41 ile bayes ağları ile elde edilmiştir, diğer yöntemlerle ulaşışan kestirim başarıları da bu aralıkta değişmektedirler.

2.2. Kullanılan Yöntemlere Yönelik Kaynak Araştırması

Floyd ve ark. (1994) çalışmalarında, mamografik bulgulardan göğüs kanserini kestirmek için bir yapay sinir ağı modeli geliştirmişlerdir. Veri seti, biyopsi yapılması planlanan hastaların radyologlar tarafından yorumlanmış toplam sekiz adet mamografik bulgusundan oluşmaktadır. Bu bulgular ağın girdisini oluştururken tümörün iyi huylu ya da kötü huylu olduğuna dair bilgi de ağın çıktısı olarak değerlendirilmiştir. Biyopsi sonuçlarına göre 168'i iyi huylu, 92'si kötü huylu tümöre sahip olduğu bilinen toplam 260 hastanın mamografik bulguları üzerinde test ettikleri model kötü huylu tümerlerin tümünü doğru sınıflandırırken iyi huylu tömerlerin de 98'ini kötü huylu olarak sınıflandırmıştır. Aynı mamografik veriler ışğında radyologlar ise kötü huylu tümörlerin tamamını doğru sınıflandırırken iyi huylu tümörlerin 38'ini kötü huylu olarak sınıflandırmışlardır.

Pal (2005) çalışmasında, rasgele orman ve destek vektör makinesi yöntemlerinin sınıflandırma başarısı, eğitim zamanı ve kullanıcı tanımlı parametreler bakımından karşılaştırmıştır. Çalışmanın veri setini Littleport, Cambridgeshire, Birleşik Krallık yakınlarındaki bir tarım alanının Landsat-7 Geliştirilmiş Tematik Haritacısından elde edilen veriler oluşturmaktadır. Arazide, buğday, şeker, patates vb. 7 tür ekinden hangisinin ekili olduğunu kestirmişlerdir. Test işlemi sonucunda rasgele orman sınıflandırıcısı ile %88,37 destek vektör makinesi ile %87,9 sınıflandırma başarısına ulaşmışlardır.

Gislason ve ark. (2006) çalışmalarında, arazi örtüsü sınıflandırması için rasgele orman sınıflandırıcısının kullanımını araştırmışlardır. Farklı sınıflarda olan toplam 1008

örneklik veriyi eğitim, 1011 örneklik veriyi de test amacıyla kullanmışlardır. Rasgele orman sınflandırıcısını farklı tipteki sınıflandırıcılar ile kıyaslamışlardır. Bunlar karar tablosu (Kohavi, 1995a), IR sınıflandırıcısı (Witten ve Frank, 1993), j4.8 (Holte, 1993) ve CART'tır. Test sonuçlarına göre en yüksek sınıflandrıma başarısına ulaşan snıflandırıcı %85,3 ile IR sınflandırıcısı iken rasgele orman sınıflandırıcısı %82,8 sınıflandrıma başarısına ulaşmıştır.

Chen ve ark. (2008) çalışmalarında, destek vektör modellerinde seçilen çekirdek fonksiyona göre sınıflandırma hassasiyetini azaltmak ve destek vektör modellerinin farklı verilere de genellenebilme yeteneğini artırmak için çoklu destek vektör sınıflandırıcılarını birleştiren bulanık bir fizyon modeli önermişlerdir. Test sonuçlarında önerilen tip-2 DVM füzyon sistemi, bireysel DVM'lerden daha kararlı ve daha güçlü genelleme kabiliyeti göstermektedir. Beraber kullanıdıkları birkaç adet DVM sınıflandırıcısını tip-2 füzyon modelinde birleştirdiklerinde, birbirlerini iyi bir şekilde tamamladıklarını ve her bir sınıflandırıcının daha iyi performans gösterdiğini belirtmişlerdir.

Omid (2011) çaşılmasında, Matlab Simulink'te geliştirmiş olduğu, karar ağacı ve bulanık sınıflandırıcı kullanarak açık ve kapalı kabuklu şam fıstıklarını belirleyip sıralayabilecek bir sistem sunmuştur. Çalışmanın verisetini 300 adet açık ve kapalı kabuklu şamfıstığına ait veriler oluşturmaktadır. Verilerin 210 adeti eğitim 90 adeti test amacıyla kullanılmıştır. Özellikle seçimi uygulamasından sonra model eğitilerek test işlemi gerçekleştirilmiştir. Test sonucunda eğitim verileri %99,52 test verileri %95,56 başarı oranıyla sınıflandırılmıştır.

Khodadadzadeh ve ark. (2014) çalışmalarında piksel tabanlı hipersectral sınıflandırma için bir çok terimli lojistik regresyon modeli önermişlerdir. Uygulamayı hem gerçek verilere hem de simülasyonla elde ettikleri verilerle yapmışlardır. Çokterimli lojistik regresona dayalı iki farklı sınıflandırma yöntemi önermişlerdir. Gerçekleştirdikleri test işlemi sonucunda iki sınflandrıma yöntemiylede %80'in üzerinde bir sınıflandrıma başarısı elde etmişlerdir.

Parvinnia ve ark. (2014) çaışmalarında, "ağırlıklı uzaklık en yakın komşu (WDNN)" olarak adlandırılan genel bir uyarlanabilir metod ile EEG sinyallerini sınıflandırmaya çalışmışlardır. Kullandıkları sınıflandırma algoritmasında eğitim örneklerine birer ağırlık vererek sınflandırma işlemi gerçekleştirmişlerdir. Bu sayede verilen ağırlık değerlerinin test örneklerinin sınıflandırılmasındaki etkileri kontrol edilebilmişlerdir. Veri seti 13'ü şizeofreni tanısı konmuş olan toplam 31 kişi'nin EEG sinyallerinden elde edilmiştir. Önerdikleri yöntemin başarısnı kıyaslamak amacıyla beş

farklı yöntemle daha sınıflandırma işlemi gerçekleştirmişlerdir. Bu yöntemler, temel sinir ağları, destek vektör makinesi, sade bayes, BDLDA (Boostani ve ark., 2009) ve ADM (Wang ve ark., 2007) yöntemleridir. Farklı sürelerdeki EEG sinyallerinden elde edilen veriler ile gerçekleştirilen test işlemi sonucunda en yüksek başarı oranına Parvinnia ve ark. tarafından önerilen WDNN metodu ile %95,32 olarak ulaşılmıştır. Testlerde EEG sinyallerinin süresinden kaynaklanan %4,12 gibi bir standart sapma meydana gelmektedir. Diğer yöntemler ile ulaşılan başarı oranı WDNN yöntemi ile ulaşılan başarı oranını altında kalmıştır. Sınıflandırma başarısı ve standart sapma olarak; basit sinir ağları için (%91.08–8.43), destek vektör makinesi için (%85.02–16.18), sade bayes için (%88.19–16.18), BDLDA yöntemi için (%87.51–16.98) ve ADM yöntemi için (%92.75–8.14) değerlerine ulaşılmıştır.

Azar ve ark. (2014) çalışmalarında lenf hastalıklarını teşhis etmek için bir rasgele orman sınıflandırıcı yaklaşımı önermişlerdir. Öncelikle, oldukça karmaşık ve büyük boyutlu olan veri seti üzerinde çeşitli özellik seçimi algoritmalarını uygulayarak yeni verisetleri elde etmişlerdir. Daha sonra rasgele orman sınıflandırıcısı ile sınıflandırdıkları veri setlerinde en başarısı sınıflandırma yüzdesine özellik seçimi için genetik algoritmalardan yararlandıkları veri seti üzerinde %99,2 oranıyla ulaşmışlardır.

Dutta ve ark. (2015) çalışmalarında, Hint borsalarında aktif olarak işlem gören hisse senetlerinin borsa performansının kestirilmesinde lojistik regresyonu kullanmışlardır. Çalışmanın veri setini otuz büyük firmaya ait dört yıllık veriler oluşturmaktadır. Hisse senetlerine ait sekiz verinin bağımsız değişken olarak ele alındığı çalışmada bağımlı değişken yüksek ve düşük şeklinde 2 gruptan oluşmaktadır. Oluşturdukları model ile 118 örneği sınflandırmışlar, yüksek örneklerde %74 düşük örneklerde %75 ortalamada ise %74,6 gibi bir başarı oranı yakalamışlardır.

Samanthula ve ark. (2015) çalışmalarında şifrelenmiş veriler üzerinde sınflandırma problemlerine odaklanarak özellikle bulutta saklanan şifrelenmiş veriler üzerinde sınıflandırma yapabilecek güvenli bir k-en yakın komşu sınıflandırıcısı önermişlerdir. Önerdikleri modeli farklı 5 ile 25 arasındaki farklı k parametrelerine göre test ettiklerinde %66.29 ile %71.66 arasında değişen doğru sınıflandrıma yüzdelerine erişmişlerdir.

Amirkhah ve ark. (2015) çalışmalarında sade bayes sınıflandırıcı kullanarak kalın bağırsak kanserinde belirleyici bir unsur olan hedef mikro RNA türlerinin (miRNA) kestirimi için yeni bir hesaplama algoritması geliştirmişlerdir. CRCmiRTar olarak adlandırdıkları bu algoritmanın, sınıflandırma modelinin eğitilmesinde gerekli olan veriyi

kalın bağırsak ve diğer kanser vakalarındaki doğrulanmış hedef miRNA'lardan elde edilmişlerdir. Başlangıçta 70 bağımsız değişken bulunan verisetlerinde farklı türde özellik seçimi yöntemleri uygulayarak yeni veri setleri elde etmişlerdir. Bu veri setleri üzerinde destek vektör makinesi, yapay sinir ağları, ragele orman ve sade bayes yöntemlerini kullanarak gerçekleştirdikleri test çalışmalarında, en yüksek doğruluk oranına %95,4 ile sade bayes yöntei ile ulaşmışlardır. Bu nedenden dolayı da geliştirdikleri algoritmada sınıflandırıcı olarak sade bayesi kullanamya karar vermişlerdir.

Tsangaratos ve Ilia (2016) çalışmalarında, heyelan duyarlılık değerlendirmelerinde lojistik regresyon ile sade bayes sınıflandırıcıya karşılaştırmışlardır. Çalışmanın veri setini ciddi toprak kayması olaylarının yaşandığı Yunanistan'daki Epirus dağlarında yer alan 116 bölgeye ait veriler oluşturmaktadır. Bu alanlar heyelanlı ve heyelansız olarak iki gruba sınıflandırılmışlardır. 9 farklı model oluşturmuşlardır. İlk üç modelde 74, ikinci üç modelde 82 ve son üç modelde de 92 örnekten oluşan bir eğitim verisi kullanmışlardır. Her bir model için gerçekleştirilen test işlemi sonucunda sonucunda lojistik regresyon için en yüksek sınıflandırma başasırı sekizinci model de elde edilen %82,61 iken sade bayes için ulaşılan en yüksek sınıflandırma başarısı altıncı ve sekkizin modellerde ulaşılan %87,50'dir.

Adeniyi ve ark. (2016) çalışmalarında, yeni geliştirilen bir RSS okuyucu websitesi üzerinde, araka plamda otomatik olarak çalışacak olan web kullanımına dayalı bir veri madenciliği uygulaması geliştirmişlerdir. Bu uygulama arka planda KNN algoritmasını kullanarak, yeni gelen kullanıcıyı web sitesindeki davranışlarına göre otomatik olarak bir grupta sınıflandırmaktadır. Daha sonra da politika, iş, spor, vb. kategorilere ayrıştırılmış bu kullanıcılara çeşitli gezinme önerileri, sayfa bağlantıları vb. özelleştirilmiş içerikler sunulmaktadır.

Miranda ve ark. (2016) çalışmalarında, sade bayes sınıflandırıcı kullanarak kardiyovasküler hastalıkları tespit edip risk düzeyini belirleyebilecek bir model önermektedirler. Kandaki yağ seviyesi, şeker, böbrek fonksiyonu gibi ölçümler ile bir veri seti oluşturmuşlardır. Sınıf değeri ise bu değerlerin derecelerine göre seviye1, seviye2 ve seviye3 olarak 3 risk grubuna ayrılmışdır. Gerçekleştirdikleri test işlemi sonucunda, seviye1 için %85,9, seviye2 için %87,98, seviye3 için %85,9 sınıflandrıma başarısına ulaşmışlardır. Testin sonuçlarının değerlendirildiği uzman toplantısında ise kardiyolaglar ve dahiliyeciler de dahil olmak üzere toplantıya katılanların %80'i önerilen

modelin kardiyovasküler hastalıkların tıbbi sürecinde fayda sağlayacağı fikrinde birleşmişlerdir.

Dey ve ark. (2016) çalışmalarında, internet kullancılarının film incelemeleri ve otel değerlendirmeleri üzerinde bir duygu analizi çalışması yapmışlardır. Kullanıcı görüşlerindeki baskın duyguların tespit edilmesinde sade bayes ve k-en yakın komşu algoritmalarından yararlanmışlardır. Duygu analizi yapabilmek için gerekli olan negatif ve pozitif anahtar kelimelerin belirleyerek, kullanıcı görüşleri üzerinde sadeleştirmeler ve dilbilimsel işlemler gerçekleştirdikten sonra her iki sınıflandırma yöntemiyle de eldeki veri seti üzerinde test işlemi gerçekleştirmişlerdir. Önceden pozitif ve negatif olarak işaretlenmiş olan 4500 film eleştirisi ve 4500 otel yorumu üzerinde yapılan test sonucunda sade bayes, film eleştirilerinde %82,43 otel yorumlarında %55,09 oranlarında başarılı bir sınflandrıma yaparken, k-en yakın komşu, film eleştirilerinde %69,81 otel yorumlarında %52,14 oranlarında başarılı bir sınıflandırma gerçekleştirmiştir.

3. YAPAY ZEKA VE MAKİNE ÖĞRENMESİ

3.1. Yapay Zeka

Herhangi bir problemin çözümü için gerekli olan tüm adımların önceden planlandığı ve tüm parametrelerin belli olduğu durumlarda bilgisayarlar, insan zekasına oranla çok daha yüksek işlem kapasitelerine erişebilirler. Bunun tam tersi olarak, ortada bir belirsizliğin veya bilinmeyen bazı parametrelerin mevcut olduğu durumlarda ya da çözüme ulaşmak için geçmiş deneyim ve gözlemlerden yararlanılması gereken durumlarda ise insan zekası bilgisayarların çok ötesinde bir performans sergiler.

İnsan zekası, öğrenebilme kaabiliyeti, deneyimlerden yararlanabilmesi, duyu organlarından gelen sinyalleri çok kısa sürede işleyebilmesi, karmaşık ilişkiler kurabilmesi, karşılaşılan problemleri çözme becerisi ve sayısını artırabileceğimiz bir çok özelliği nedeniyle uzun yıllardır bilgisayar bilimleri için önemli bir araştırma konusudur. Bu alanda yapılan araştırmalar zaman içerisinde bilgisayar bilimlerinin bir alt alanı haline gelen yapay zekanın (YZ) doğuşuna neden olmuştur. Temelde insan zekasının sahip olduğu eşsiz yeteneklerin, bir takım algoritmalar ile modellenerek taklit edilmesini amaçlayan YZ, son dönemin popüler araştırma alanlarından birisi konumundadır.

YZ'yi doğru anlamanın yolu öncelikli olarak zeka kavramını doğru tanımlamaktan geçmektedir. Zeka geçmişten günümüze birçok bilimsel çalışmanın odak noktasını oluşturmaktadır. Böylesine uzun bir araştırma ve tartışma geçmişi olmasına rağmen zeka kavramının hala standart bir tanımı yoktur (Legg ve Hutter, 2007). İki düzine tanınmış kuramcıdan zeka kavramının tanımlamaları istense ortaya iki düzine farklı tanım çıkacaktır (Sternberg ve Detterman, 1986). Yıllar içerisinde farklı disiplinlerden birçok bilim insanı zeka kavramını çeşitli şekillerde tanımlamışlardır. Bu tanımlardan bazıları şu şekildedir:

- Binet zekayı, yargılama, sağduyu, pratik düşünme, insiyatif alma ve bulunduğu koşullara adapte olma kabiliyeti olarak tanımlar (Binet ve Simon, 1916).
- Zeka, soyut düşünceyi sürdürme becerisidir (Terman, 1921).
- Zeka biyolojik bir adaptasyondur, zekanın biyolojik adaptasyonun özel bir örneği olduğunu söylemek aslında onun bir organizasyon olduğunun varsaymaktır. Zekanın yaşamla olan ilişkisini anlamak için organizmanın çevresi ile olan ilişkisine bakmak gerekir. Organizmanın yakın çevresini yapılandırması gibi, zekanın fonksiyonu da evreni yapılandırmaktır (Piaget, 1952).

- Zeka, bireyin bilinçli davranması, gerçekçi düşünmesi ve çevresi ile etkili bir şekilde başa çıkabilmesi için sahip olduğu kapasitesidir (Wechsler, 1958).
- Bireyler karmaşık fikirleri anlayabilme, çevrelerine uyum sağlama, geçmiş tecrübelerinden öğrenme, akıl yürütme, engellerin üstesinden gelme gibi yeteneklerde farklılıklar gösterebilir, bu farklılıklar önemli olsa da farklı ölçütlerle değerlendirildiklerinden farklı sonuçlar verdikleri için tutarlı değildirler. Zeka kavramı tüm bu karmaşık fenomenler silsilesini organize ederek açıklığa kavuşturmaya yönelik girişimlerdir (Neisser ve ark., 1996).
- Zeka bir ya da daha fazla kültürel ortam için değerli olan bir ürün yaratma veya problemleri çözme becerisidir (Gardner, 2011)
- Zihinsel bir özellik olarak değerlendirilen zeka, henüz başlangıç aşamasında dürtülere odaklanabilme kapasitesidir ve bu nedenle de zeka, kısıtlayıcı bir süreç olan soyutlama kapasitesidir (Thurstone, 2013).
- Zeka aklı kullanma yeteneğidir (Erol, 2019).

Henüz zeka kavamı için bile üzerinde herkesçe uzlaşılmış bir tanım yapılamamışken YZ'yi herkesçe kabul görmüş ortak bir tanım ile nitelemek çok da kolay değildir. Çok sayıda bileşenden oluşması ve multidisipliner yapısı nedeniyle, YZ kavramı birçok uzman tarafından farklı şekillerde tanımlanmıştır.

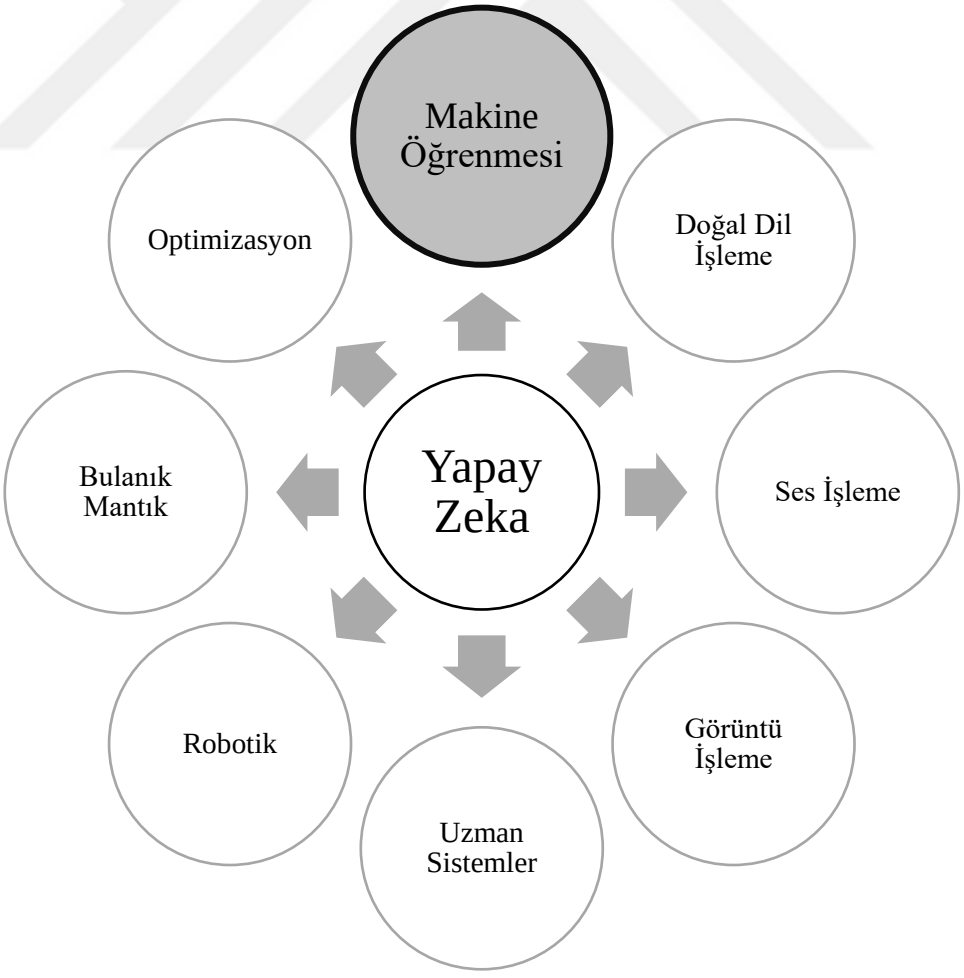
YZ kavramını ilk kez 1955 yılında Amerikalı bir bilgisayar bilimcisi olan John McCcarthy kullanmış (Nilsson, 2010) ve onu akıllıymış gibi davranan makinelerin geliştirilmesi olarak tanımlanmıştır (Ertel, 2018). McCarthy'nin "Yapay Zeka Yaz Araştırma Projesi" başlığı ile teklif ettiği projenin Rockefeller Vakfı tarafından desteklenmesiyle birlikte 1956 yılında Dartmouth College'de YZ'nin doğuşu olarak kabul edilen bir konferans düzenlenmiştir. Bu konferansa John MCCarthy ile birlikte katılan, Marvin Minsky, Allen Newell, Herbert Simon YZ'nin babaları olarak kabul edilmektedir (Nabiyev, 2010).

Zaman içerisinde tıpkı zeka kavramı gibi çeşitli şekillerde tanımlanmaya devam eden YZ kavramı için yapılan bazı tanımlar şu şekildedir:

- YZ, hesaplamalı modellerin kullanımı sayesinde zihinsel kabiliyetlerin araştırılmasıdır (Charniak, 1985).
- YZ, akıllı makineleri, özellikle de akıllı bilgisayar programlarını yapabilmenin bilimi ve mühendisliğidir (McCarthy, 1998).

- YZ, genel olarak insana özgü nitelikler olarak kabul edilen, akıl yürütme, anlam çıkarma, genelleme ve geçmiş tecrübelerden öğrenme gibi ileri zihinsel işlemler gerektiren görevlerin bilgisayar veya bilgisayar denetimli makineler tarafından yerine getirilmesidir (Nabiyev, 2010).
- YZ, akıllıca hareket eden hesaplama araçlarının analiz ve sentezine yönelik çalışmaların yapıldığı bir alandır (Poole ve Mackworth, 2010).
- YZ'nin her birisi tanımlayanın kendi görüşüne doğru eğimli olan pek çok tanımı vardır ancak bu tanımlar kabaca, insanlar tarafından sergilendiklerinde akıllı olarak değerlendirilen davranışları gerçekleştirebilecek bilgisayar programları veya makineler yaratabilme şeklindeki bir tanım ekseninde birleşmiştir (Kaplan, 2016).

YZ, üzerinde çalışılan problemin türüne ve kullanılan yöntemlere göre çeşitli alt alanlara ayrılır. Günümüzde her biri oldukça popüler olan ve farklı türdeki birçok araştırmaya temel oluşturan bu alt alanlar, literatürde en sık kullanılan isimleriyle birlikte Şekil 3.1'de gösterilmektedir. Çalışmamızın temelini bu alanlar arasından "Makine Öğrenmesi" oluşturmaktadır.



Şekil 3.1. Yapay Zeka'nın alt alanları

3.2. Makine Öğrenmesi

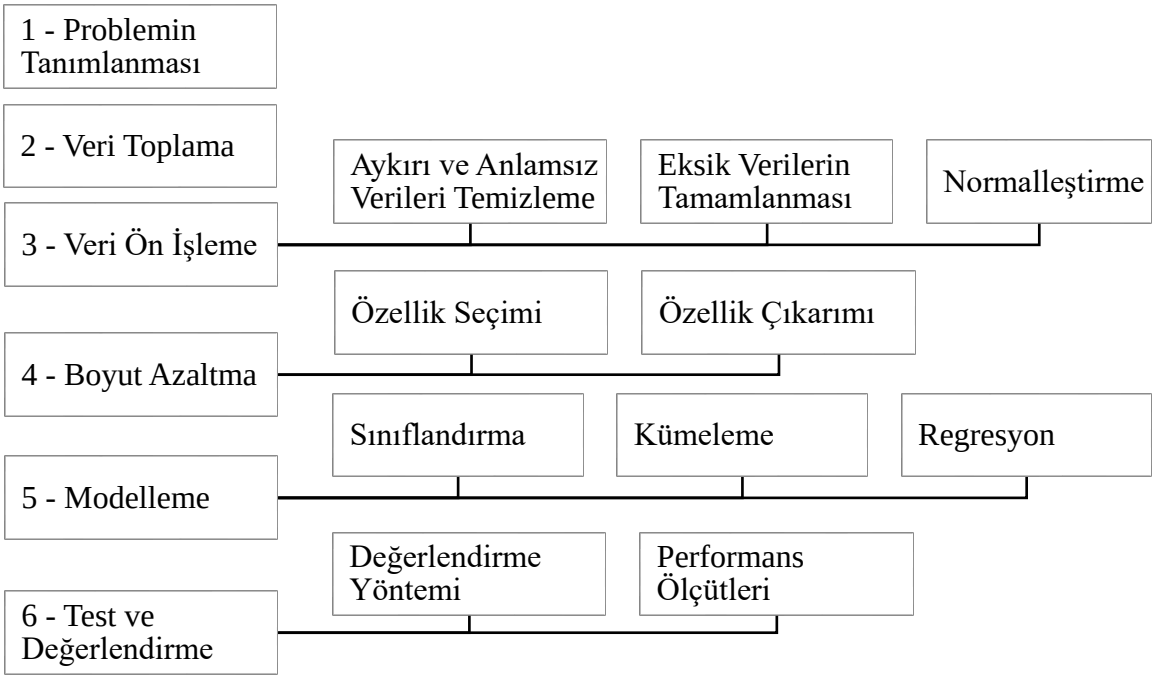
Öğrenme, benzer popülasyondan alınan benzer görev veya görevlerin bir sonraki seferde daha verimli ve daha etkili bir şekilde yapılabilmesi için sistemde meydana gelen değişiklikleri ifade eder (Simon, 1983). Buradan hareketle makine öğrenmesinde de daha iyi bir sonuca ulaşmak adına sürekli olarak belli döngülerin tekrarlandığını söyleyebiliriz. Kabaca tanımlamak gerekirse; makine öğrenmesi, insanoğlunun geçmiş tecrübelerinden veya yeni karşılaştığı bilgilerden yola çıkarak bir problemi çözebilme yeteneğinin, bilgisayar teknolojileri tarafından taklit edilmesidir. Genellikle bir dizi örnekten yola çıkarak öğrenme gerçekleştirebilen, mantıksal veya aritmetik işlemlere dayalı otomatik hesaplama prosedürlerine sahip yapılardır (Michie ve ark., 1994).

3.3. Makine Öğrenmesinin Aşamaları

Makine öğrenmesi yöntemlerinin uygulama süreci literatürde farklı şekillerde açıklanmış ve belli aşamalar ile gösterilmiştir (Mannila, 1996; Kononenko ve Kukar, 2007; Libbrecht ve Noble, 2015; Lehr ve Ohm, 2017).

Genelde problemin tanımlanması ve ön araştırmanın yapılması ile başlayan bu süreç, veri seti ile ilgili işlemleri ile devam eder. En uygun veri setinin oluşturulmasını takiben model oluşturma ile ilgili çalışmalar gerçekleştirilir. Son olarak da oluşturulan model test edilerek çıktılar değerlendirilir.

Bu süreç, çalışmamızda Şekil 3.2'de gösterildiği gibi planlanmıştır.



Şekil 3.2. Makine öğrenmesinin aşamaları

Toplam altı parçada ele alınan bu süreç aşağıda yer alan başlıklarda kısaca açıklanmıştır.

3.3.1. Problemin Tanımlanması

Birçok bilimsel yöntem ve teknikte olduğu gibi makine öğrenmesi yöntemlerinde de ilk yapılması gereken problemin tüm hatlarıyla ortaya koyulmasıdır. Problemin hangi varsayımlar altında oluşturulduğu ve sınırlılıkların ne olduğu tam anlamıyla belirlenmelidir. Problemin çözümü için ne tür bir veriye ihtiyaç duyulduğu, bu verinin hangi araçlar ile toplanacağı, toplanan verinin hangi yöntemler ile modelleneceği bu aşamada planlanmalıdır.

3.3.2. Veri Toplama

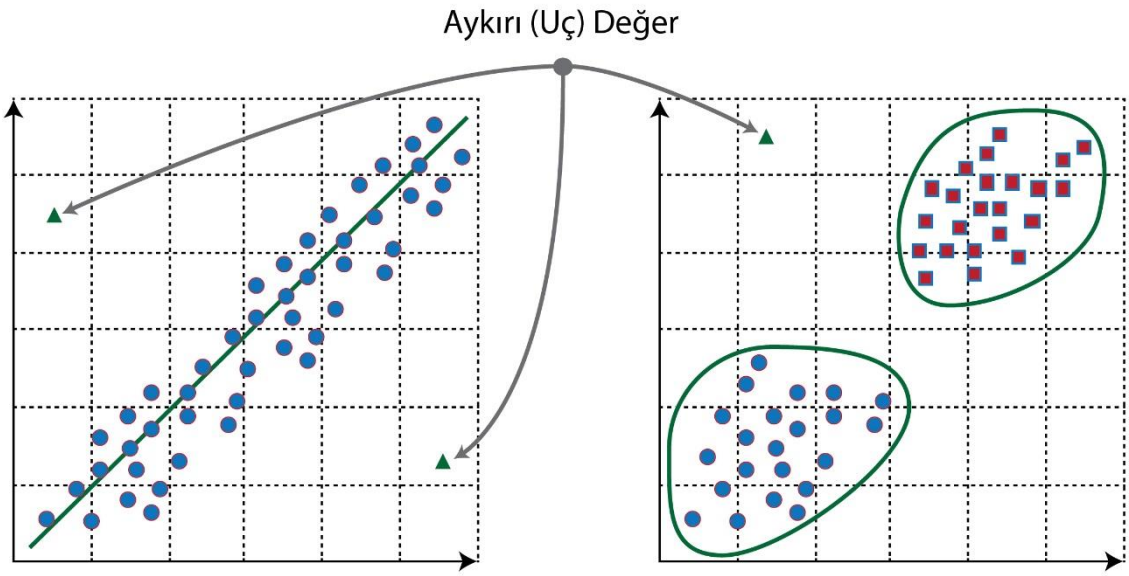
Problem tanımlama aşamasında tüm hatları ile planlanan makine öğrenmesi çalışmasının en önemli basamaklarından birisi veri toplama aşamasıdır. Önceden belirlenmiş veri kaynaklarına mevcut veri toplama araçları ile ulaşarak veri toplama sürecinin en az hatayla gerçekleştirilmesi sağlanır. Toplanan verinin türüne göre kayıt ortamları belirlenir. Veri tabanı tablolarına kaydedilecek veriler için bu tablolar oluşturulur ve gerekli durumlarda tablolar arasındaki ilişkiler kurulur.

3.3.3. Veri Ön işleme

Veri ön işlemeyi, çeşitli kaynaklardan elde edilmiş ham verilerin, en uygun yapıdaki veri setine dönüştürülmesi süreci olarak tanımlayabiliriz. Bu aşamada aykırı ve yanlış değerler içeren örnekler dışarıda bırakılır, eksik veriler için gerekli düzenlemeler gerçekleştirilir ve veri normalleştirme işlemi yapılır. Bu işlemler aşağıda kısaca açıklanmıştır.

Aykırı ve Hatalı Verilerin Temizlenmesi

Aykırı gözlemler, diğer gözlemlerin kümelendiği bölgeden şüphe uyandıracak şekilde bir sapma gösteren gözlemlerdir (Hawkins, 1980). Bir istatistiksel analiz öncesi ideal veri setinin oluşturulması sürecinde çözüm getirilmesi gereken en önemli konulardan birisi olan aykırı gözlemler için Şekil 3.3’de örnekler gösterilmektedir. Literatürde, aykırı gözlemlerin belirlenmesi adına gerçekleştirilmiş çok sayıda çalışma vardır (Hadi, 1994; Aggarwal ve Yu, 2001; Hawkins ve ark., 2002; Hodge ve Austin, 2004; Ben-Gal, 2005).



Şekil 3.3. Veriler arasındaki aykırı gözlemler

Toplanan verileri düzenlerken dikkat edilmesi gereken bir diğer konu da varsa hatalı verilerin düzenlenmesidir. Veri kaynağından, veri toplama aracından veya farklı nedenlerden dolayı bazı verilerde hatalı gözlemler oluşabilmektedir. Reel sayı biçiminde bir değere sahip olması gereken bir özelliğin metinsel bir değer alması veya bir tarih içermesi gereken özelliğin bir tam sayı ile değerlenmesi bu duruma verilebilecek örneklerdendir. Hatalı verilerin düzeltilmesi sürecinde çeşitli istatistiksel yazılımlardan veya algoritmalardan yardım alınabilir.

Eksik Verilerin Tamamlanması

Eksik ya da kayıp veri olarak adlandırılan bu durumu, bazı gözlemlerin bir ya da daha çok özelliğine dair değerlerin bulunmayışı olarak tanımlayabiliriz. Bu durum çeşitli nedenler ile gerçekleşebilir. Bir anket aracılığı ile veri toplanırken katılımcılardan birisinin bazı soruları cevaplamaması kayıp veriye neden olacaktır. Aynı şekilde bir ölçüm aracı ile belli zaman aralıklarında veriler elde edilirken de ölçüm aracında meydana gelecek olan bir arıza bazı zaman aralıklarında veri elde edilmesini engelleyecek ve yine bir kayıp veri durumu oluşacaktır. Bu durumda kayıp verilerin tamamlanabilmesi için çeşitli yöntemlerden faydalanabilir. Literatürde kayıp verilerin tamamlanabilmesi için çeşitli yöntemlerin sunulduğu çok sayıda çalışma mevcuttur (Grzymala-Busse ve Hu, 2000; Raghunathan ve ark., 2001; Royston, 2004; Acock, 2005; Moons ve ark., 2006). Bu yöntemler genel olarak, en çok tekrarlanan değeri, en az tekrarlanan değeri, sütun ortalamasını, ortanca değeri veya belli fonksiyonlar sonucu elde edilmiş değerleri kayıp verilerin tamamlanması için kullanırlar.

Veri Normalleştirme

Veri normalleştirme süreci, birçok istatistiksel analizin, makine öğrenmesi yönteminin veya veri madenciliği algoritmasının en önemli aşamalarından biridir. Temel amaç çok farklı ölçeklerdeki değişkenlerin tamamına ortak bir ölçekte ve belli bir aralıkta değerler vermektir. Bu işlem pek çok hesaplamanın daha hızlı gerçekleşmesini, eğitim ve test sürelerinin azalmasını sağlar. En önemli avantajı ise veri setinde yer alan değişkenlerin ölçeklerine karşı duyarlı olan yöntemlerin performanslarında artışa neden olmasıdır.

Veri normalleştirme amacıyla kullanılan pek çok yöntem vardır (Phan ve ark., 2006; Jayalakshmi ve Santhakumaran, 2011; Long ve ark., 2015; Bhanja ve Das, 2018). Literatürde çeşitli çalışmalarda veri normalleştirme tekniklerinin performanslar karşılaştırılmıştır (Naumann ve ark., 1992; Bolstad ve ark., 2003; Rao ve ark., 2008; Bullard ve ark., 2010). Elde edilen performanslar, üzerinde çalışılan veri setine veya kullanılan makine öğrenmesi yöntemine göre de değişkenlik gösterebilmektedir. En sık kullanılan yöntemler aşağıda açıklanmıştır.

Bu yöntemlerde;

$$X = \left(x_1, x_2, \ldots, x_p \right),$$

$$x_i = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad i = 1, 2, \ldots, p,$$

x_{ij} : i. sütunun j. değerini,

x'_{ij} : i. sütunun j. değerinin normalleştirilmiş halini,

$x_{i_{\min}}$: i. sütunun en küçük değerini,

$x_{i_{\max}}$: i. sütunun en büyük değerini,

μ_{x_i} : i. sütunun ortalamasını,

σ_{x_i} : i. sütunun standart sapmasını,

ifade etmektedir.

Min-Max Normalleştirme

Bu yöntemde veri setinin her bir sütunu kendi içinde ele alınır. Sütundaki en yüksek değer 1 en küçük değer 0 olarak kabul edilip geri kalan tüm değerler bu aralıkta dağıtılır. Bu değerler (3.1)'de gösterildiği şekilde hesaplanır.

$$x'_{ij} = \frac{x_{ij} - x_{i_{\min}}}{x_{i_{\max}} - x_{i_{\min}}}, \qquad i = 1,2,\dots p, \quad j = 1,2,\dots,N \tag{3.1}$$

Min-Max normalizasyon yönteminde (3.1)'de verilen denklem, (3.2)'deki gibi düzenlenerek verilerin $[0,1]$ aralığı yerine $[a,\ a+b]$ aralığında dağılması sağlanabilir.

$$x'_{ij} = a + (b-a) \cdot \frac{x_{ij} - x_{i_{\min}}}{x_{i_{\max}} - x_{i_{\min}}} \tag{3.2}$$

Z-Skor Normalleştirme

Bu yöntemde (3.3)'te gösterildiği gibi sütün içerisindeki her değerden önce sütun ortalaması çıkarılır ardından ise bulunan değer sütunun ortalamasına bölünür.

$$x'_{ij} = \frac{x_{ij} - \mu_{x_i}}{\sigma_{x_i}} \tag{3.3}$$

Medyan Normalleştirme

Bu yöntemde sütun içerisindeki her değer (3.4)'te gösterildiği gibi sütunun medyanına bölünerek normalleştirilmiş veri elde edilir.

$$x'_{ij} = \frac{x_{ij}}{median(x_i)} \tag{3.4}$$

Sigmoid Normalleştirme

Bu yöntemde sigmoid fonksiyonu kullanılarak veriler normalleştirilir. (3.5)'te verilen denklem aracılığıyla normalleştirilen veriler $[0,1]$ veya $[-1,1]$ aralığında dağılır.

$$x'_{ij} = \frac{e^{x_{ij}} - e^{-x_{ij}}}{e^{x_{ij}} + e^{-x_{ij}}} \tag{3.5}$$

3.3.4. Boyut İndirgeme

Verilerden anlam çıkarmaya yönelik çalışmalarda, mevcut veri seti çok fazla öznitelik içerebilir. Bu durum geliştirilen modelin yüksek performans gösterebilmesi açısından bir avantaj olarak görülse de bazı durumlarda öznitelikler arasındaki yüksek ilişkiler veya kimi öz niteliklerin modelin performansı üzerindeki negatif etkisi bu avantajın göz ardı edilemeyecek bir dezavantaja dönüşmesine neden olabilir. Bu gibi sorunlardan kaynaklanan performans kaybının önlenebilmesi açısından öznitelikler

üzerinde gerçekleştirilen çeşitli hesaplamalar ile mevcut veri setini temsil edecek daha küçük boyutlu bir veri setine ulaşmak amaçlanır. Bu işlem için literatürde en sık kullanılan çözümler özellik seçimi ve özellik çıkarımı başlıkları altında incelebilir.

Özelik Seçimi

Özellik seçimi, mevcut veri setinde yer alan öznitelikler içerisinden bazılarının seçilmesiyle, ana veri setini temsil edebilecek daha küçük boyutlu yeni bir veri setine ulaşmaya yönelik çalışmalardır. Özellik seçimi yapılabilecek çok sayıda yöntem mevcuttur (Kira ve Rendell, 1992; Liu ve Setiono, 1995; Liu ve Yu, 2005; Liu ve Motoda, 2007; Yu ve ark., 2008; Li ve ark., 2012). Bu yöntemlerin model performanslarına olan etkisinin karşılaştırılması amacıyla çok sayıda çalışma yapılmıştır (Sun ve ark., 2002; Garrett ve ark., 2003; Refaeilzadeh ve ark., 2007; SpolaôR ve ark., 2013).

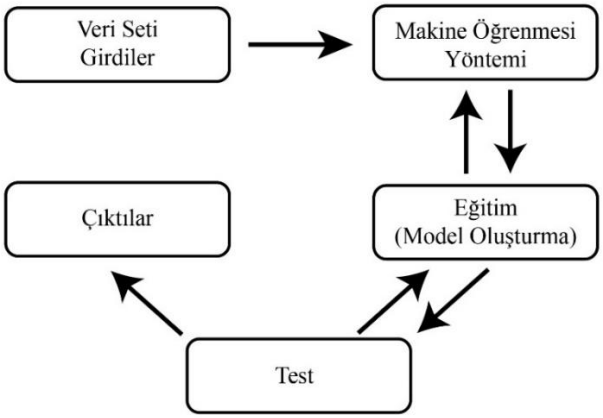
Özellik seçimi için kullanılan yöntemler üç ana yaklaşıma dayanırlar, bunlar filtre (filters), sarmal (wrappers) ve gömülü (embedded) yöntemlerdir (El Akadi ve ark., 2011). Filtrelemeye dayalı yöntemlerde eğitim işlemi gerçekleştirilmeden veri setinden elde edilen bilgilere (korelasyon, frekans, vs.) göre özellik seçimi gerçekleştirilir. Sarmal yöntemlerde veri setindeki özelliklerin her bir alt kümesi için denemeler yapılır ve en başarılı sonucu veren alt küme seçilir. Gömülü yöntemlerde ise veri seti üzerindeki özellik seçimine ek olarak kullanılan yöntemin parametrelerinde de değişiklik gerçekleştirilerek en optimum sonuç elde edilir.

Özellik Çıkarımı

Özellik çıkarımında veri setindeki öz nitelikler kullanılarak daha az sayıda yeni öz nitelik türetilir. Amaç olarak özellik seçimi ile benzerlik gösteren özellik çıkarımının temel farklılığı, mevcut veri setini temsil edecek daha küçük boyutlu yeni veri setinin oluşturulma mantığıdır. Özellik seçiminde eldeki öznitelikler içerisinden bazıları seçilerek yeni veri setine aktarılırken özellik çıkarımında eldeki öznitelikler kullanılarak daha az sayıda yeni öznitelik türetilir. Böylelikle, ilk veri setini temsil edecek ve ondan daha küçük boyuta sahip olan yeni bir veri seti elde edilir. Özellik çıkarımı yapılabilecek çok sayıda yöntem mevcuttur (Yang ve ark., 2009; Shrawankar ve Thakare, 2013; Kumar ve Bhatia, 2014). Literatürde çeşitli çalışmalarda bu yöntemlerin etkililikleri karşılaştırılmıştır (Gong ve ark., 1992; Pichler ve ark., 1996; Preece ve ark., 2008). En çok kullanılan özellik çıkarım yöntemleri ise temel bileşenler analizi (principal component analysis) doğrusal ayırıcı analiz (linear discriminant analysis) ve bunların doğrusal olmayan durumlar için çekirdek fonksiyonları ile genelleştirilmiş halleridir.

3.3.5. Modelleme

Bu aşamada sınıflandırma, kümeleme veya regresyon diye temel başlıklara ayırdığımız yöntemlerden biri ya da birkaçı belirlenir. Hangi yöntemlerin belirleneceği, araştırma problemine ve eldeki veri setine bağlı olarak değişkenlik gösterir. Bu yöntemlere ek olarak araştırmacı tarafından sıfırdan geliştirilen yeni yöntemler veya mevcut yöntemlerden uyarlanan hibrit yöntemler de kullanılabilir. Gerekli yöntemler seçildikten sonra eğitim işlemi gerçekleştirilerek modeller oluşturulur. Model oluşturma işlemi deneme yanılma ile birçok kez tekrarlanır. Oluşturulan çok sayıda model içerisinden araştırma problemine en yüksek performans ile çözüm getiren model nihai model olarak kabul görür. Bu durum Şekil 3.4'de gösterilmektedir.



Şekil 3.4. Makine öğrenmesinde modelleme yapısı

Bu çalışmada kullanılan mevcut makine öğrenmesi yöntemleri ve oluşturulan hibrit yöntemler hakkındaki detaylı bilgiler çalışmanın dördüncü ve altıncı bölümünde verilmiştir.

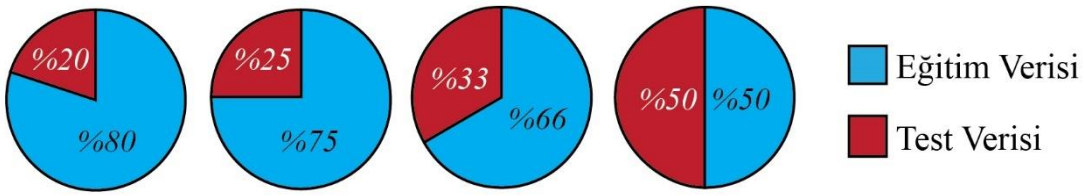
3.3.6. Test ve Değerlendirme

Test Yöntemleri

Makine öğrenmesi çalışmalarından oluşturulan modelin performansının test edilerek değerlendirilebilmesi için çeşitli yöntemler mevcuttur (Kohavi, 1995b; Isaksson ve ark., 2008; Kim, 2009; Borra ve Di Ciaccio, 2010). Bunların en bilinenleri "holdout", "çapraz geçerlilik (cross validation)", "tabakalı örnekleme (stratified sampling)", "üçlü ayırma (three-way split)" ve "yeniden örnekleme (bootstrap)" yöntemleridir.

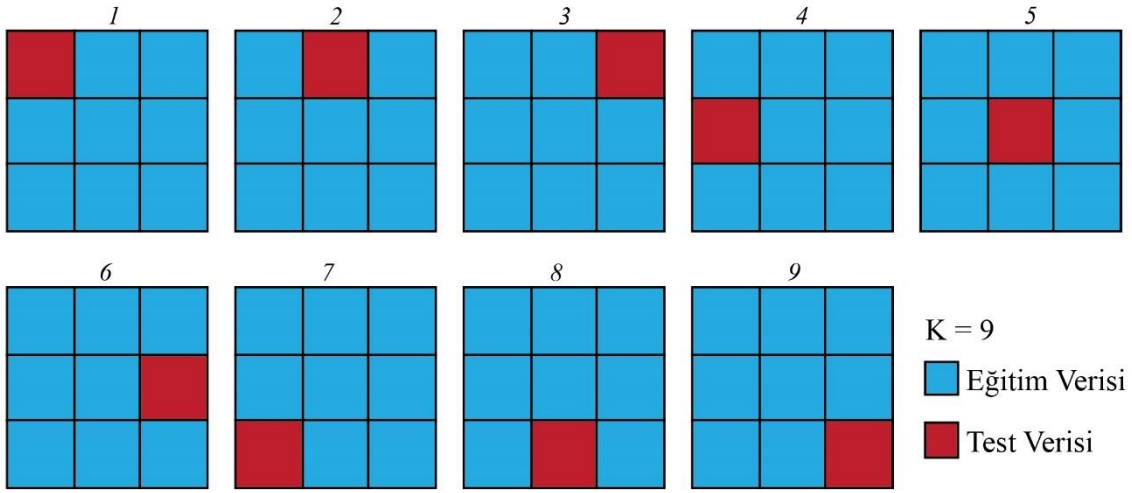
Holdout yönteminde veri seti belli bir oran belirlenerek eğitim ve test verisi şeklinde iki dosyaya ayrılır. Belirlenecek olan oran mevcut veri setindeki örnek sayısı ile

ilişkilidir. Birçok çalışmada %66'ya %33, %75'e %25, %80'e %20 gibi bir oranlara rastlanabilirken veri setindeki örnek sayısının çok fazla olduğu durumlarda test veri setinin oranı çok düşük seviyelerde de tutulabilir. Şekil 3.5'te eğitim ve test kümesinin nasıl ayrılacağı örnek oranlar ile gösterilmektedir.



Şekil 3.5. Holdout yönteminde eğitim ve test küme örnekleri

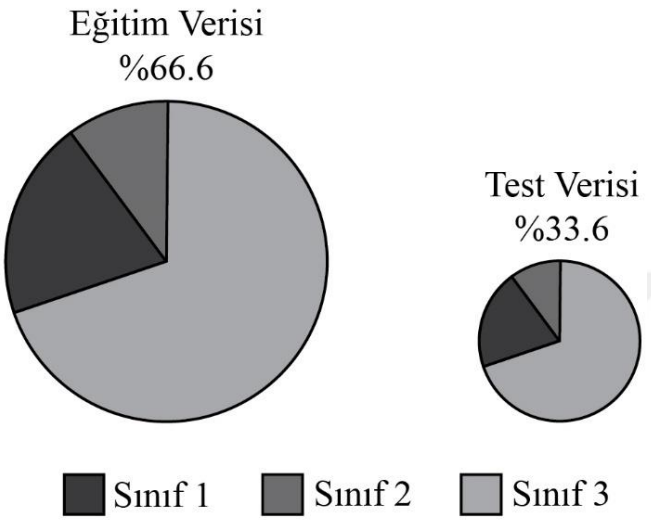
Çapraz geçerlilik yönteminde ise veri seti önceden belirlenen bir *k* sayısı kadar eşit parçaya ayrılır. Literatürde bu ayrılması işleminin nasıl gerçekleştirileceğine dair çeşitli çalışmalar mevcuttur (Kohavi, 1995b; Isaksson ve ark., 2008; Borra ve Di Ciaccio, 2010). Ayrılan parçalardan her biri sırası ile test verisi olarak ele alınırken geri kalan *k-1* tanesi de eğitim verisi olarak değerlendirilir. Böylelikle *k* sayısı kadar eğitim ve test gerçekleştirilir. Gerçekleştirilen bu testlerde elde edilen performans değerlerinin ortalaması da modelin nihai performansı olarak kabul edilir. Şekil 3.6'da ayırma sonrası eğitim ve test verisindeki değişimler gösterilmektedir.



Şekil 3.6. Çapraz geçerlilik yönteminde eğitim ve test verisinin değişimi

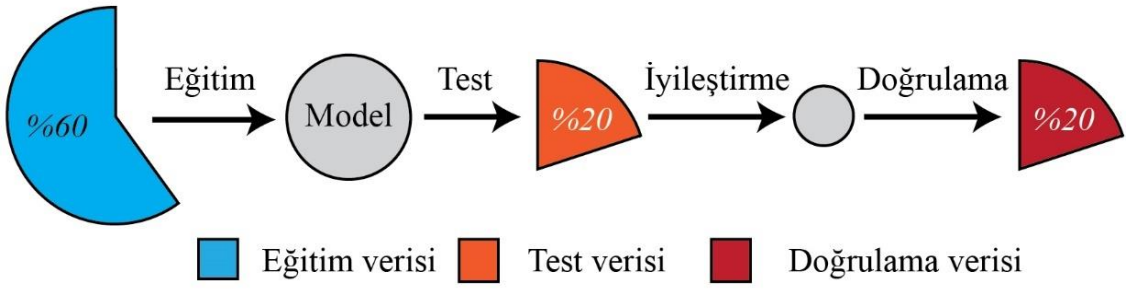
Tabakalı örnekleme yöntemi, veri setinde her sınıftan birbirine yakın sayıda örnek bulunmadığı durumlarda test verisinin belirlenmesi amacıyla kullanılır. Veri setindeki gözlemlerin üç farklı sınıf etiketine sahip olduğu ve %70'inin birinci sınıfta, %20'sinin ikinci sınıfta, %10'unun da üçüncü sınıfta etiketlendiği örnek bir durumda rasgele

seçilecek test verisinde sınıflar arasındaki oran sınıflandırıcının performansı açısından çok sağlıklı olmayabilir. Bu nedenle tabakalı örnekleme yönteminde test verisi oluşturulurken her bir sınıf etiketinden kaç gözlemin seçileceği o sınıf etiketinin toplam veri içerisindeki yoğunluğuna göre belirlenir. Böylelikle sınıf etiketlerinin toplam veri içerisindeki oranları test verisinde de korunmuş olur olur. Şekil 3.7'de tabakalı örnekleme yönteminin örnek bir gösterimi yer almaktadır.



Şekil 3.7. Tabakalı örnekleme yönteminde eğitim ve test verisi

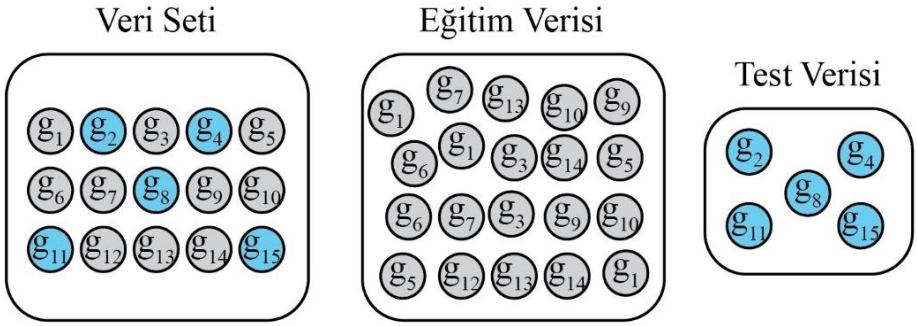
Üçlü ayırma yönteminde veri seti, eğitim, test ve doğrulama diye üç parçaya ayrılır. Eğitim verisi ile eğitilip, test verisi ile test edilen model üzerinde, test işlemi sonrasında ulaşılan sonuçlara göre birtakım iyileştirmeler yapılarak yeni bir model elde edilir. Sonrasında ise bu model doğrulama verisi ile tekrar test edilir. Bu süreç Şekil 3.8'de gösterilmektedir.



Şekil 3.8. Üçlü ayırma yönteminde eğitim, test ve doğrulama verisi

Yeniden örnekleme yönteminde ise veri setinden belli bir sayıda gözlem çekilir, her bir gözlem çekildikten sonra veri setinden çıkarılmaz ve iadeli bir seçim işlemi gerçekleştirilir. Bu işlem eğitim verisi için belirlenen sayı kadar tekrarlandıktan sonra

eğitim verisi oluşturulur. Eğitim verisinde yer almayan gözlemler ise test verisine atanır. Eğitim verisinde aynı gözlemler tekrar edebilirken, test verisinde ise tekrarlı gözlem yer almaz. Şekil 3.9'da bootstrap yönteminin nasıl gerçekleştiği gösterilmektedir.



Şekil 3.9. Bootstrap yönteminde eğitim ve test verisi oluşturma

Performans Ölçütleri

Makine öğrenmesine dayalı geliştirilen bir modelin etkililiğini ölçmek için çeşitli performans değerleri incelenir (Willis ve ark., 1993; Sing ve ark., 2005; Huang, 2006; Costa ve ark., 2007; Cherny ve Madan, 2008). Bu değerlere ulaşmak için literatürde karışıklık matrisi (confusion matrix) olarak adlandırılan bir tablodan yararlanılır. Karışıklık matrisleri bir takım test değerlerine göre sınıflandırıcının performansını özetlerler (Ting, 2017). Bu test değerleri, literatürde "true positive (TP), false positive (FP), true negative (TN) ve false negative (FN)" isimleri ile anılmakta ve temelde doğru ve yanlış sınıflandırma sayılarına dayandırılarak tanımlanmaktadırlar.

TP, FP, TN ve FN değerlerinden oluşan bir karışıklık matrisinin yapısı en genel hali ile Çizelge 3.1'de gösterildiği gibidir. Karışıklık matrislerinin bu yapısı, kestirilmek istenen değişkenin "Pozitif" ve "Negatif" şeklindeki iki sınıf etiketine sahip olduğu durumlar içindir.

Çizelge 3.1. Karışıklık matrisi gösterimi

		Sınıflandırma	
		1	**0**
Gerçek Sonuç	1	TP (Doğru Pozitif)	FN (Yanlış Negatif) (2. Tip Hata)
	0	FP (Yanlış Pozitif) (1. Tip Hata)	TN (Doğru Negatif)

Bu matriste;

TP: Gerçekte 1 iken sınıflandırma sonucunda da 1 olarak sınıflandırılan gözlemlerin sayısını,

TN: Gerçekte 0 iken sınıflandırma sonucunda da 0 olarak sınıflandırılan gözlemlerin sayısını,

FP: Gerçekte, 0 iken sınıflandırma sonucunda 1 olarak sınıflandırılan gözlemlerin sayısını,

FN: Gerçekte 1 iken sınıflandırma sonucunda 0 olarak sınıflandırılan gözlemlerin sayısını gösterir.

Karışıklık matrisinde yer alan bu değerler kullanılarak bir makine öğrenmesi yönteminin performans ölçütleri hesaplanır. Bu performans ölçütlerinin ne anlama geldiği ve nasıl hesaplandıkları aşağıda kısaca açıklanmıştır.

Gözlenen Doğruluk (Observed Accuracy – ACC): Modelin doğru sınıflandırdığı gözlemlerin oranıdır. Gözlenen doğruluk (Observed Accuracy) olarak da adlandırılır. (3.6)'da gösterildiği gibi hesaplanır.

$$\boldsymbol{ACC} = \frac{TP + TN}{TN + FN + FP + TP} \tag{3.6}$$

Beklenen Doğruluk (Expected Accuracy – EACC): (3.7)'de gösterildiği şekilde hesaplanan bu değer modelin beklenen doğruluk oranını ifade eder.

$$\boldsymbol{EACC} = \frac{(TP + FP)(TP + FN) + (FN + TN)(FP + TN)}{(TN + FN + FP + TP)^2} \tag{3.7}$$

Duyarlılık (Sensitivity – TPR): Sınıflandırıcının pozitif gözlemleri doğru sınıflandırmadaki başarısıdır. Literatürde TP Rate, recall gibi isimlerle de geçen duyarlılık (3.8)'deki gibi hesaplanır.

$$\boldsymbol{TPR} = \frac{TP}{TP + FN} \tag{3.8}$$

Belirleyicilik (Specificity – TNR): Sınıflandırıcının negatif gözlemleri doğru sınıflandırmadaki etkililiğidir. Bazı çalışmalarda, belirlilik, özgüllük, TN Rate, spesificity gibi isimlerle de kullanılan belirleyicilik (3.9)'da gösterildiği gibi hesaplanır.

$$\boldsymbol{TNR} = \frac{TN}{TN + FP} \tag{3.9}$$

FP Rate (1-Belirleyicilik, FPR): Yanlış sınıflandırılmış pozitif gözlemlerin oranını verir. (3.10)'da gösterildiği gibi hesaplanır.

$$\boldsymbol{FPR} = \frac{FP}{FP + TN} \tag{3.10}$$

FN Rate (FNR): Yanlış sınıflandırılmış negatif gözlemlerin oranını verir ve (3.11)'de gösterildiği gibi hesaplanır.

$$\boldsymbol{FNR} = \frac{FN}{TP + FN} \tag{3.11}$$

Kesinlik (Precision – PPV): Doğru şekilde sınıflandırılmış pozitif gözlemlerin, pozitif sınıflandırılmış tüm gözlemlere oranına kesinlik ya da pozitif kestirim değeri denir. (3.12)'deki gibi hesaplanır.

$$\boldsymbol{PPV} = \boldsymbol{Kesinlik} = \frac{TP}{TP + FP} \tag{3.12}$$

Negatif Kestirim Değeri (NPV): Doğru sınıflandırılmış negatif gözlemlerin tüm negatif gözlemlere oranını gösterir ve (3.13)'de gösterildiği gibi hesaplanır.

$$\boldsymbol{NPV} = \frac{TN}{TN + FN} \tag{3.13}$$

F-Ölçüsü (F-Measure – F-M): Duyarlılık ve kesinlik değerlerinin harmonik ortalaması olan F-ölçüsü (3.14)'te gösterildiği gibi hesaplanır.

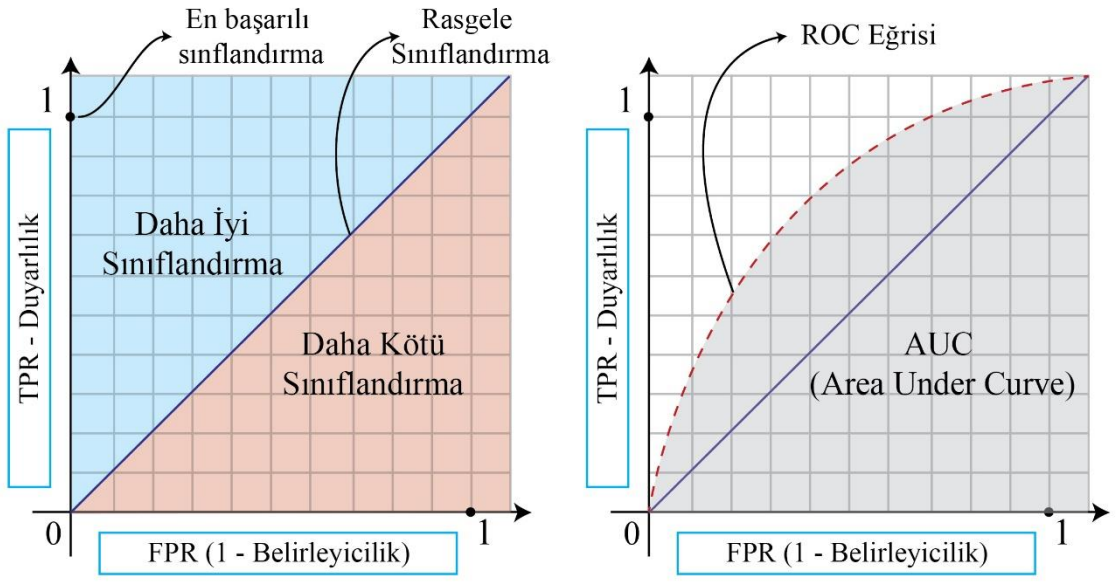
$$\boldsymbol{F\text{ - }M} = \frac{2}{\frac{1}{TPR} + \frac{1}{PPV}} = \boldsymbol{2} \times \frac{PPV \times TPR}{PPV + TPR} \tag{3.14}$$

Kappa İstatistiği (Kappa Statistics – KS): Kappa istatistiği, tesadüfen doğru sınıflandırılmış örnekleri kontrol etmek için beklenen ve gözlenen sınıflandırma sonuçları

arasındaki uyuma bakar. (3.15)'deki gibi hesaplanan KS 0 ile 1 arasında değer alır. 1'e yaklaşması sınıflandırıcının başarısının da yükselmesi anlamına gelirken 0 değerini alması ise sınıflandırıcının rasgele yapılmış bir tahminden pek de farkının olmadığını gösterir.

$$\boldsymbol{KS} = \frac{ACC - EACC}{1 - EACC} \tag{3.15}$$

ROC (Receiver Operating Characteristic – Alıcı İşlem Karakteristikleri) Eğrileri: ROC eğrileri test işlemi boyunca sınıflandırılan her değer sonrası TPR (duyarlılık) ve FPR (1-Belirleyicilik) değerlerinin güncellenmesi ile çizilir. Şekil 3.10'da gösterildiği üzere, TPR, X eksenine, FPR ise Y eksenine yerleştirilir.



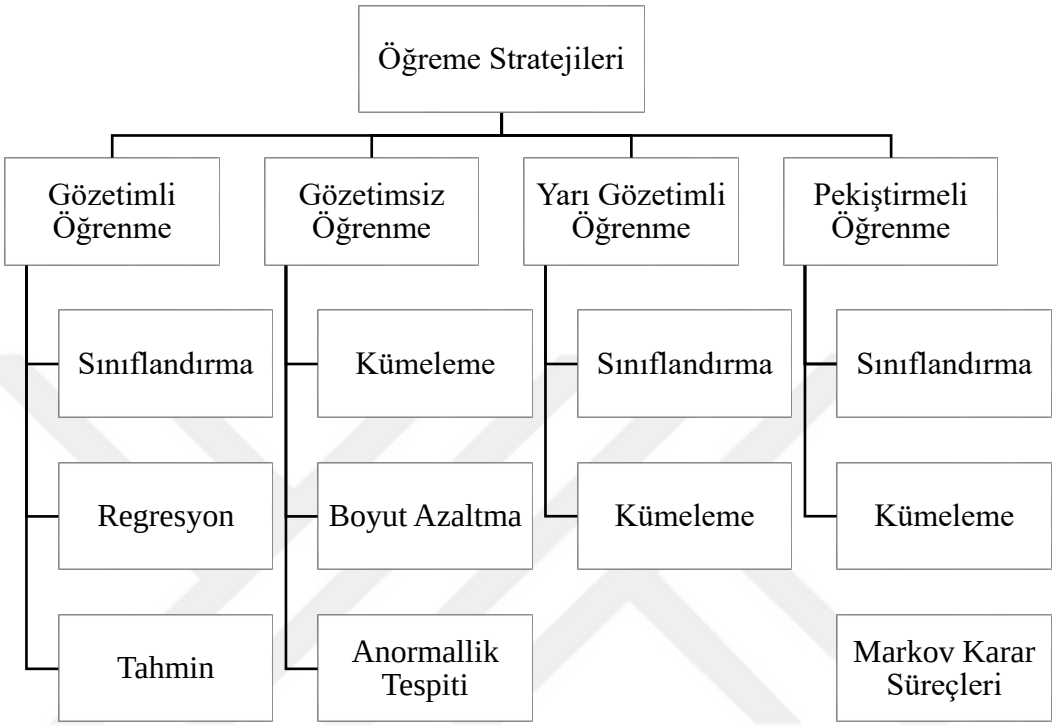
Şekil 3.10. ROC eğrisi çizimi

Roc eğrisinin altında kalan alanın 1'e yaklaşması sınıflandırıcının başarısının artması anlamına gelir. Literatürde AUC (Area Under Curve) şeklinde geçen bu değerin 1 olması sınıflandırıcının mükemmel bir sınıflandırma gerçekleştirdiğini gösterir.

3.4. Öğrenme Stratejileri

Makine öğrenmesinde, öğrenme işlemi genelde dört farklı şekilde gerçekleşir. Bunlar; gözetimli, yarı gözetimli, gözetimsiz ve pekiştirmeli öğrenmedir (Hoogendoorn ve Funk, 2017). Öğrenme türüne göre Şekil 3.11'de gösterildiği gibi dört kategoride

incelenebilen makine öğrenmesi çalışmaları, üzerinde çalışılan probleme göre de çeşitli kategorilere ayrılabilir. Sınıflandırma, tahmin, kümeleme, ilişki belirleme, karar verme, anormallik tespiti, veri indirgeme vb. türdeki çalışmalar bu kategorilere örnek olarak verilebilir.



Şekil 3.11. Makine öğrenmesinde öğrenme türleri

3.4.1. Gözetimli Öğrenme

Girdileri X, çıktıları Y olan ve N sayıda örnekten oluşan bir A veri setimizin olsun.

$$X = (x_1, x_2, ..., x_m),\ m = 1, 2, ..., M$$

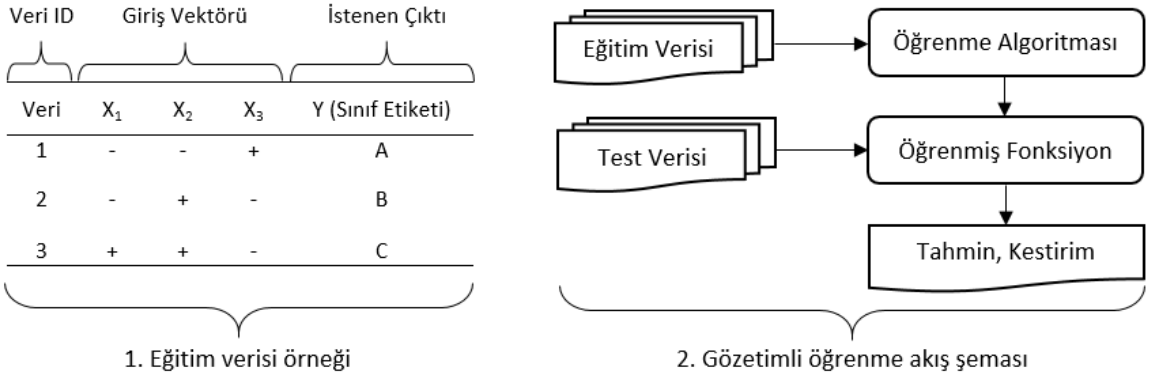
$$Y = (y_1, y_2, ..., y_c),\ c = 1, 2, ..., C$$

$$A = \{(x_1, y_1), (x_2, y_2), ..., (x_3, y_n)\},\ n = 1, 2, ..., N$$

M veri setindeki her bir örneğin bağımsız değişken sayısını, *C*, bağımlı değişkenin sınıf sayısını, *N* ise veri setindeki örnek sayısını gösterir. Bağımsız değişkenler veya girdiler örnekler hakkında bilgi içerdikleri için genellikle özellikler olarak adlandırılırken (Schuld ve Petruccione, 2018) bağımlı değişkenler de hedef olarak adlandırılırlar.

Gözetimli öğrenmede amaç her bir x_n girdisini y_n çıktısına taşıyacak olan $y = f(x)$ şeklindeki eşleştirme fonksiyonuna ulaşmaktır. Sonrasında ise gerçek hedef ile

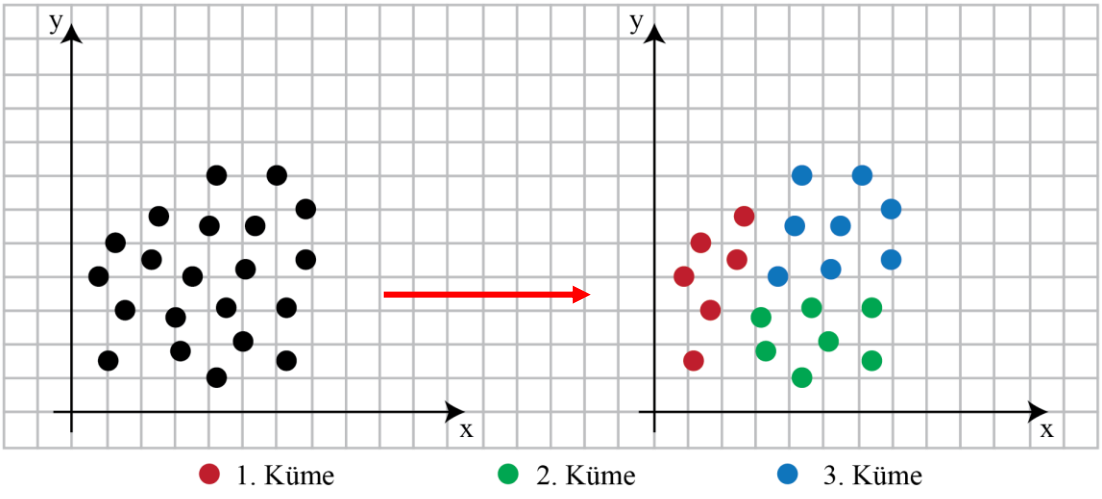
ulaşılan hedef arasındaki farkı hesaplayacak bir kayıp fonksiyonu seçilerek bu fark minimize edilmeye çalışılır (Narsky ve Porter, 2013). Şekil 3.12'de çalışma mantığı gösterilen gözetimli öğrenme sınıflandırma, regresyon ve sıralama problemlerinde kullanılan en yaygın öğrenme yöntemidir (Mohri ve ark., 2018).



Şekil 3.12. Gözetimli öğrenme çalışma mantığı

3.4.2. Gözetimsiz Öğrenme

Gözetimsiz öğrenme algoritmaları, belirgin öğretici bilgiler olmadan veri setinden anlamlı yapılar çıkarmak için tasarlanmışlardır (Buhmann, 1998). Gözetimli öğrenmede sınıf etiketleri önceden bilindiği için net bir başarı ölçümü yapılabilir ama gözetimsiz öğrenme de bu durum söz konusu değildir (Hastie ve ark., 2009). Şekil 3.13'te örneklendirilmiş olan gözetimsiz öğrenme genellikle sınıf etiketi bilinmeyen örneklerden oluşan veri setleri üzerinde kümeleme yapmak amacıyla kullanılır. Aynı zamanda büyük boyutlu veri setlerini, aynı veri setini temsil edecek daha küçük boyutlu bir yapıya dönüştürmek amacıyla da sıklıkla kullanılır.

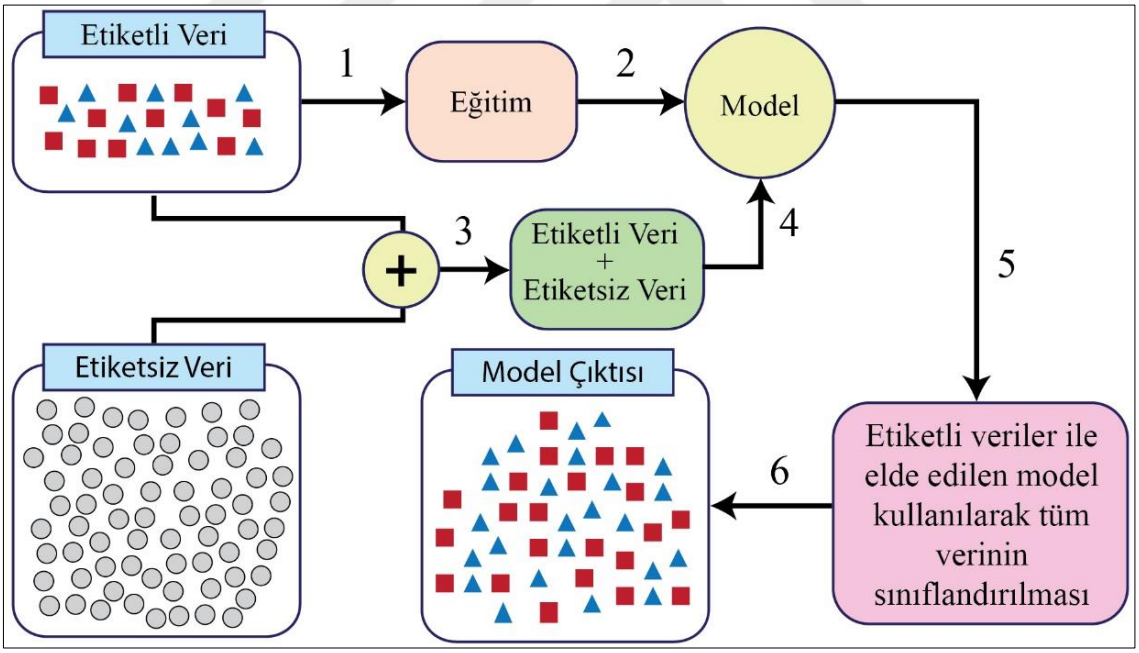


Şekil 3.13. Gözetimsiz öğrenme örneği

3.4.3. Yarı Gözetimli Öğrenme

Yarı gözetimli öğrenme, gözetimli öğrenme ve gözetimsiz öğrenmenin tam orta noktasıdır (Chapelle ve ark., 2010). Veri setinin, hem etiketli hem de etiketsiz veri içerdiği durumlarda öğrenme işleminin nasıl gerçekleşeceği ile ilgilenen öğrenme paradigmasıdır (Zhu ve Goldberg, 2009). Bazı alanlarda etiketli verilerin elde edilme süreci, uzman görüşüne ihtiyaç duyulması, uzun bir süreçte gerçekleşmesi, daha masraflı olması gibi nedenlerden dolayı oldukça zahmetlidir (Zhu ve ark., 2003). Bu alanlarda etiketsiz verinin toplanmasının daha az zahmet gerektirmesine ek olarak ulaşmış olduğu yüksek doğruluk oranlarıyla da yarı gözetimli öğrenme, teoride ve pratikte oldukça ilgi gören bir öğrenme yöntemi haline gelmiştir (Zhu, 2005).

Veri setindeki etiketlenmemiş veri miktarı etiketlenmiş veri miktarına oranla oldukça fazladır. Şekil 3.14'te çalışma mantığı gösterilen yarı gözetimli öğrenme ile az miktardaki etiketlenmiş veriden yola çıkarak etiketlenmemiş veriler hakkında bir yargıya varılır. Metin ve görüntü sınıflandırmada sıklıkla kullanılan öğrenme yöntemlerinden birisidir.

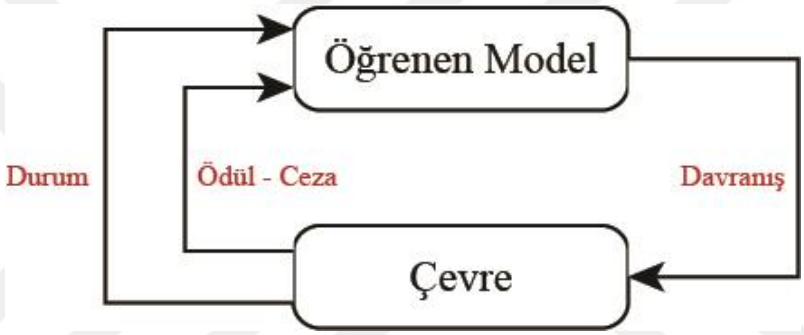


Şekil 3.14. Yarı gözetimli öğrenme çalışma mantığı

3.4.4. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme, hedefe ulaşmak için deneme yanılmaya dayalı bir öğrenme modelidir (Sutton, 1992; Barto, 1997; Sutton ve Barto, 1998; Barto ve Rosenstein, 2004; Cerrada ve Aguilar, 2008; Peters, 2008). Sistemin öğrenebilmesi için çevresinden bilgi

alması gerekir (Cerrada ve Aguilar, 2008). Gözetimli öğrenmede sistem her girdi değerine karşılık gelen gerçek çıktıları önceden bilir ve ek bir bilgiye ihtiyaç duymadan bu girdi-çıktılar sayesinde öğrenme işlemini gerçekleştirir. Pekiştirmeli öğremede ise öncesinde girdilerin eşleştiği gerçek çıktılar hakkında bilgisi olmayan sistem, eldeki girdilerden yol çıkarak bir takım çıktılar üretir ve bu çıktılara karşılık olarak da çevresinden dönütler alır. Burada çevreyi sistemin davranışlarını yorumlayan bir eleştirmen olarak düşünebiliriz (Alpaydın, 2017). Bu eleştirmenin gerçek görevi ise sisteme, doğru ürettiği çıktılar için ödül (artı puan), yanlış ürettiği çıktılar için ise ceza (eksi puan) vermektir. Ödülleri puan olarak değerlendirecek olursak sistemin amacı en yüksek puana ulaşmaktır. Oluşturulan modeller içerisinde en yüksek puana ulaşan model ise nihai model olarak kabul görür. Bu yapı Şekil 3.15'te gösterilmektedir.

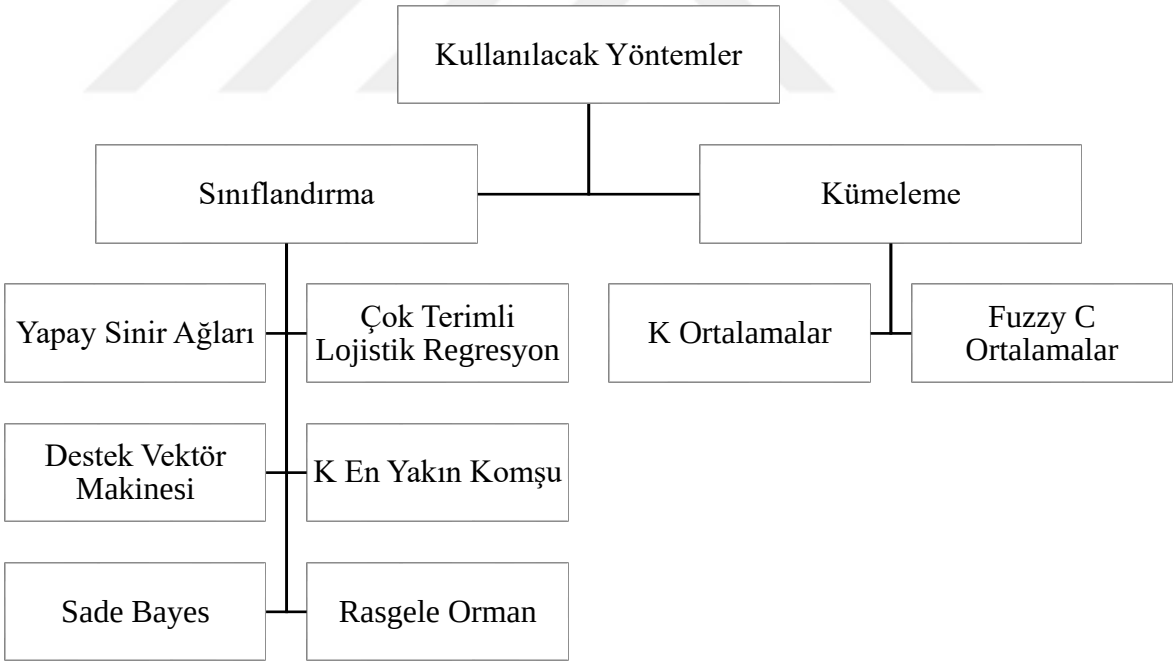


Şekil 3.15. Pekiştirmeli öğrenmenin yapısı

4. KULLANILAN SINIFLANDIRMA VE KÜMELEME YÖNTEMLERİ

Literatürde, çeşitli öğrenme algoritmalarından faydalanılarak geliştirilmiş çok sayıda sınıflandırma ve kümeleme yöntemi mevcuttur (Estivill-Castro, 2002; Abbasi ve Younis, 2007; Wei ve ark., 2008; Anyanwu ve Shiva, 2009; Lamkanfi ve ark., 2011; Tang ve ark., 2014; Zhang ve ark., 2017). Bu yöntemler kimi zaman, farklı türlerdeki problemler ve veri setleri üzerinde çeşitli performans ölçütlerine göre karşılaştırılırken kimi zaman da hibrit bir modelin oluşturulması adına bir arada kullanılmıştır.

Bu tez çalışmasında da genel ve özel amaçlar doğrultusunda Şekil 4.1'de gösterildiği üzere, çeşitli sınıflandırma yöntemlerine ek olarak kümeleme yöntemlerinden de faydalanılmıştır. Sınıflandırma için, YSA (Yapay Sinir Ağları - Artificial Neural Network), ÇLR (Çokterimli Lojistik Regresyon - Multinomial Logistic Regression), KEYK (K En Yakın Komşu Algoritması - K Nearest Neighbor), SB (Sade Bayes - Naive Bayes), RO (Rasgele Orman - Random Forest), DVM (Destek Vektör Makinesi - Support Vector Machine), kümeleme için ise KO (K-Ortalamalar Kümeleme - K-Means Clustering) ve BCO (Bulanık C Ortalamalar - Fuzzy C-Means Clustering) yöntemleri kullanılmıştır.



Şekil 4.1. Kullanılacak makine öğrenmesi yöntemleri

4.1. Sınıflandırma

Basit şekilde tanımlamak gerekirse; sınıflandırma, örneklemdeki her bir örneği bir sınıfa ya da kategoriye atama işlemidir (Mohri ve ark., 2018). Bu atama işlemini yapacak

olan modelin atama öncesinde belli sayıda örnek veri kullanılarak eğitilmesi gerekir. Eğitim işlemi sınıflandırmada kullanılacak olan yöntemin yapısına göre farklılık gösterebilir. Sınıflandırma için kullanılabilecek çok sayıda yöntem vardır (Harrington, 2012). Temelde, olasılıksal ve olasılıksal olmayan sınıflandırıcılar diye iki farklı şekilde ele alınabilecek olan bu yöntemler bir çok alanda başarılı şekilde kullanılmaktadır (Rogers ve Girolami, 2016).

Olasılıksal sınıflandırıcılar bir örneğin tüm sınıflara ait olabilme olasılıklarını hesaplar ve bunlar içerisinden en yüksek olanını bu örneğin sınıfı olarak atar. Sade bayes lojistik regresyon veya rasgele orman sınıflandırıcılarını olasılıksal sınıflandırıcılara örnek olarak verebiliriz.

Girdileri X , çıktıları Y olan bir sınflandırma problemini (4.1)'de gösterildiği gibi ifade edebiliriz.

$$\begin{aligned} X &= \{x_1, x_2, ..., x_n\} \\ Y &= \{y_1, y_2, ..., y_n\} \\ x_i &\in X, \quad y_i \in Y \\ g(x_i) = y_i &= \arg\max_{y_i} \Pr(Y = y \mid X) \end{aligned} \tag{4.1}$$

Olasılıksal olmayan sınıflandırıcılar literatürde ilişkisel, bilgi temelli gibi isimlerle de karşımıza çıkabilir. Genelde, sınıf etiketi bilinen örneklerden yola çıkarak bir takım ilişkilerin varlığını tespit etmeye çalışırlar ya da bir başka deyişle etiketleri bilinen örneklerden bir takım bilgiler elde ederler. Daha sonra ise bu ilişkiler veya bilgiler ışığında yeni gelen örnekleri bir sınıfa atarlar. Destek vektör makineleri, K en yakın koşu, yapay sinir ağları gibi yöntemler olasılıksal olmayan sınıflandırma yöntemlerine örnek olarak verilebilir.

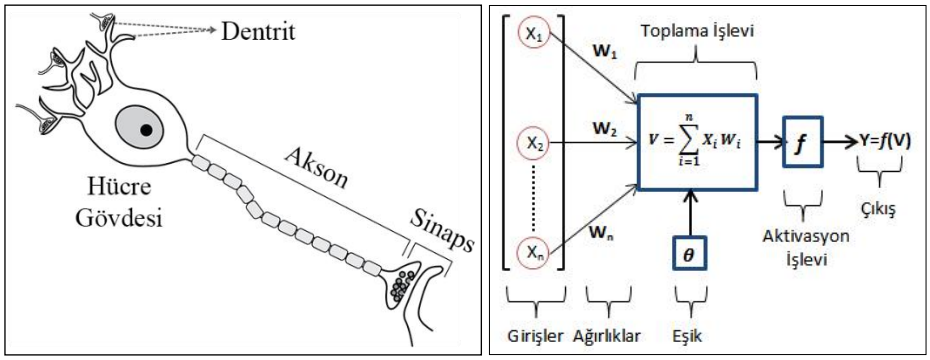
4.1.1. Yapay Sinir Ağları

YSA (Yapay Sinir Ağları) kavramının ortaya çıkışı, sinir bilimi, mantık ve sibernetik üzerine çalışmalar yapan Warren McCulloch ve Walter Pitts tarafından sinirsel aktivitelerin mantıksal olarak modellenmesi amacıyla 1943 yılında yapılmış olan bir çalışmaya dayanır (McCulloch ve Pitts, 1943). YSA, insan beynindeki biyolojik sinir ağlarının çalışma mantığını taklit etmek amacıyla geliştirilmiştir (Civco, 1993; Fish ve ark., 1995; Corchado ve Lees, 1998; Khan ve ark., 2001; Topcu ve Sarıdemir, 2008).

Çeşitli alanlardaki farklı türlerden problemlerin (Yao ve Liu, 1997; Cannady, 1998; Zhang ve ark., 1998; Hydrology, 2000; Kalogirou, 2000; Dawson ve Wilby, 2001) çözümünde kullanılabilen YSA için zaman içerisinde çok sayıda tanım yapılmıştır. Bu tanımlardan bazıları şunlardır.

- YSA, tecrübeler sonucu elde edilen bilginin depolanmasını ve gerektiği zaman kullanılmak üzere hazır hale getirilmesini sağlayan paralel olarak dağıtılmış çok büyük ölçekteki bir işlemcidir (Haykin, 1994).
- YSA, ağın yapısına, bağlantı ağırlıklarına ve hesaplama elemanlarında gerçekleşen işlemlere göre farklı fonksiyonlara sahip olabilen ve birbirine paralel olarak çalışan çok sayıdaki basit işlem biriminden oluşan sistemlerdir (Widrow, 1988).
- YSA, insan beyninin en önemli özelliği olan öğrenme işlevini çeşitli örneklerden yararlanarak gerçekleştirebilen bilgisayar sistemleridir (Öztemel, 2012).

İnsan beyninin reaksiyon süresini kusursuz kılan ve aynı zamanda anlık karar verebilme yeteneğinin de çok üst düzeyde olmasını sağlayan en temel bileşenlerden birisi biyolojik sinir ağlarıdır. Bu sinir ağları, hem duyu organlarından alınan sinyallerin beyine ulaştırılmasını hem de beynin bu sinyaller sonucu beynin oluşturduğu reaksiyonun gerekli organlara taşınmasını sağlarlar. Bu ağları oluşturan biyolojik sinir hücreleri yani nöronlar, birbirleri ile haberleşme sürecinde belli bir mantığa göre hareket ederler. YSA, yapıları ve çalışma biçimleri itibariyle biyolojik sinir ağlarını taklit ederek belli başlı problemlerin çözümünde insan beyninin kusursuzluğunu yakalamaya çalışırlar. Resim 4.1'de gösterilmekte olan biyolojik ve yapay sinir hücreleri incelendiğinde ortaya çok benzer bir yapı çıkmaktadır (Jain ve ark., 1996; Sharma ve ark., 2012).



Resim 4.1. Biyolojik ve yapay sinir hücrelerinin yapısı

Biyolojik sinir hücreleri genel olarak dört temel bileşenden oluşur, bunlar dendrit, soma (hücre gövdesi), akson ve sinapstır. Dendritler giriş verisi olarak adlandırılan bilgiyi

diğer hücrelerden alarak bu bilgileri somaya iletirler. Soma bilgiler üzerinde gerekli işlemleri gerçekleştirdikten sonra ulaştığı sonuçlar akson aracılığı ile sinapslara iletir. Sinapslar ise diğer hücrelerin dendritleri ile etkileşime geçerek bu bilginin iletimini sağlarlar. Burada aksonlar çıkış birimi, sinapslar ise sinir hücreleri arasındaki iletişimin sağlanması için gerekli olan elektriksel bağlantıların gerçekleştiği birimlerdir.

Yapay sinir hücrelerinde de bu iş akışı taklit edilmektedir, ağa giriş katmanı aracılığıyla gelen bilgiler işlenir ve çıkışa aktarılır. Bu süreçte beş temel bileşenden bahsedebiliriz. Bunlar; girişler, bağlantı ağırlıkları, toplama işlevi (fonksiyonu), aktivasyon işlevi (fonksiyonu) ve çıkış işlevidir. Bu birimlerin işlevleri şu şekildedir.

- Girişler, ağa dış dünyadan veya diğer hücrelerden veya bizzat kendisinden gelen bilgileri temsil eder. Bu bilgiler ağın öğrenebilmesini sağlayan örneklerdir.
- Bağlantı ağırlıkları, girişlerden gelen bilgilerin hücre üzerindeki etkisini belirleyen değerlerdir. Bağlantı ağırlıklarını biyolojik sinir hücrelerindeki sinapslara benzetebiliriz(Eluyode ve Akomolafe, 2013). Ağırlık değerinin büyüklüğü o girişin hücre ile olan bağlantısının gücünü gösterir. Bağlantı ağırlıkları pozitif veya negatif olabilirler. Bir bağlantı ağırlığının büyük olması o bilginin çok önemli olduğu anlamı taşımaz. Oluşturulan modellerde çoğu zaman başlangıçta rasgele değerler atanan bağlantı ağırlıkları ağın öğrenme sürecinde en ideal değerlere ulaşırlar.
- Toplama işlevi (fonksiyonu), hücreye gelen net girdinin hesaplandığı adımdır. Çeşitli toplama fonksiyonları mevcuttur ve bunlar içerisinde ağırlıklı toplam en çok tercih edilenidir ve (4.2)'de gösterildiği gibi hesaplanır (Öztemel, 2012). Hangi toplama fonksiyonun kullanılacağına ağın ulaştığı başarıya göre deneme yanılma yöntemi ile karar verilebilir.

$$NET = \sum_{i}^{n} G_i A_i \tag{4.2}$$

NET : Hücreye gelen net girdi,
G_i : Girdiler,
A_i : Bağlantı Ağırlıkları,

- Aktivasyon işlevi (fonksiyonu), toplama fonksiyonu ile elde edilen hücreye ait net girdinin işlenmesiyle hücrenin çıktı değerinin hesaplandığı adımdır. Tıpkı toplama fonksiyonunda olduğu gibi aktivasyon için de çeşitli fonksiyonlar kullanılabilir (Salas ve ark., 2000; Karlik ve Olgac, 2011). Çeşitli çalışmalarda aktivasyon fonksiyonlarının ağın başarısındaki etkililikleri karşılaştırılmıştır (Özkan ve Erbek, 2003; Zadeh ve ark., 2010).

(4.3)'te gösterilen sigmoid aktivasyon fonksiyonu çok katmanlı YSA'da en çok tercih edilen aktivasyon fonksiyonudur (Öztemel, 2012). Eldeki veri ile oluşturulacak ağ için hangi aktivasyon fonksiyonunun seçileceğine tıpkı toplam fonksiyonunda olduğu gibi ağın sonucunda ulaşılan başarıya göre deneme yanılma yapılarak karar verilebilir.

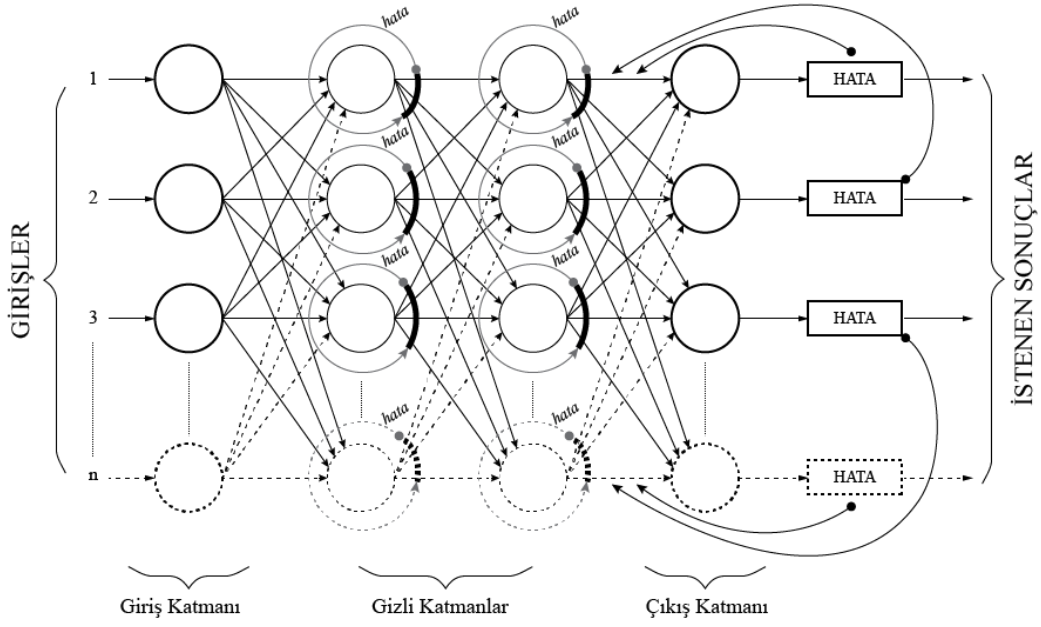
$$F(NET) = \frac{1}{1+e^{-NET}} \tag{4.3}$$

- Çıkış işlevi, aktivasyon fonksiyonu ile elde edilen çıktı değerleridir. Bu değer dış dünyaya gönderilebileceği gibi hücrenin kendisine veya başka bir hücreye giriş değeri olarak da işlem görebilir.

YSA'da öğrenme işlemi bağlantı ağırlıklarının ideal değerlere ulaştırılmasıdır. Bu amaçla çeşitli öğrenme algoritmaları kullanılmaktadır. Bunlar içerisinde geri yayılım algoritması en çok kullanılanlardan birisidir.

Geri Yayılım Algoritması

Geri yayılım algoritması, çok katmanlı YSA'da ağ başarısını en üst düzeye çıkaracak olan bağlantı ağırlıklarının elde edilmesi için kullanılır. Bu algoritmanın geliştirilmesiyle ileri yönelimli çok katmanlı ağlar en yaygın kullanılan ağ mimarisi haline gelmişlerdir (Yegnanarayana, 2005). Geriye yayılım algoritmasında genellikle toplam fonksiyonu olarak ağırlıklı toplam, aktivasyon fonksiyonu olarak ise sigmoid fonksiyonu kullanılır. Geri yayılım algoritmasının örnek bir gösterimi Resim 4.2'de verilmiştir.



Resim 4.2. Geri yayılım algoritmasının yapısı

Ağ ilk oluşturulduğunda belirlenmesi gereken parametreler, bağlantı ağırlıkları, gizli katman sayısı ve gizli katmanlardaki nöron sayılarıdır. Bağlantı ağırlıkları rastgele değerler olarak belirlenir. Gizli katmanların yapısına ise gerçekleştirilen testler sonucunda karar verilir. Gizli katmanlar YSA'nın zeka içeren kısımlarıdır. Gizli katmanlardaki nöron sayıları ağ ilk oluşturulduğunda en az sayıda tutulur (Nabiyev, 2010). Sonrasında ise yapılan deneme yanılmalar sonucunda ağı en yüksek başarıya ulaştıracak olan gizli katman ve nöron sayıları belirlenir (Feng ve ark., 2009; Panchal ve ark., 2011; Esfe ve ark., 2015). Gizli katmanların ve bu katmanlardaki nöron sayılarının fazla olması ağı ezberlemeye yöneltir. Bu durumu insan beynindeki nöronların bilgi tutmasına benzetebiliriz. Herhangi bir görüntü ya da metni tam olarak belleğinde tutan insan bu iş için çok fazla nöron kullanır, oysaki gerekli bilgileri kodlayarak hafızasına kaydettiğinde bu işlemi çok daha az nöron kullanarak ve daha kalıcı şekilde gerçekleştirebilir.

$q = 1, 2, \ldots, Q$: katman sayısı,

H_i^q : q'inci katmandaki i biriminin girdisi,

y_i^q : q'inci katmandaki i biriminin çıktısı,

w_{ij}^q : $q-1$'inci katmandaki i birimini, q'inci katmandaki j birimine bağlayan ağırlık,

δ_i^q : q'inci katmandaki i birimindeki hata,

v_i : istenilen çıktı,

n : öğrenme katsayısı,

olmak üzere K katmanlı bir ileri beslemeli yapay sinir ağı modeli için geri yayılım algoritmasının çalışma mantığı aşağıdaki gibidir.

- Başlangıç ağırlık değerleri rastgele atanır.
- q katmanındaki her bir j birimi için, (4.4)'teki gibi ileri yönde çıktı değerleri hesaplanır.

$$y_i^q = f\left(\sum_i y_i^{(q-1)} w_{ij}^q\right) \tag{4.4}$$

- Beklenen ve oluşan çıktı değerleri hesaba katılarak (4.5)'te gösterilen şekilde, çıkış birimleri için hata terimleri hesaplanır.

$$\delta_i^Q = \left(v_i^Q - y_i^Q \right) f'\left(H_i^Q \right) \tag{4.5}$$

- (4.6)'da gösterildiği gibi, $q=2,3,\ldots,Q$ katmanlarındaki tüm i birimleri için geriye yayılımla gizli katman birimlerinin hata terimleri hesaplanır.

$$\delta_i^{q-1} = f'\left(H_i^{q-1} \right) \sum_i \delta_i^q w_{ij}^q \tag{4.6}$$

- Bütün ağırlıklar w_{ij}'ler kullanılarak (4.7)'de gösterildiği gibi güncellenir (Elmas, 2010).

$$\begin{aligned} w_{ij}^{yeni} &= w_{ij}^{eski} + \Delta w_{ij}^q \\ \Delta w_{ij}^q &= n\delta_i^q y_i^{q-1} \end{aligned} \tag{4.7}$$

4.1.2. Çokterimli Lojistik Regresyon

Lojistik regresyonun kökleri 19. yy'ın başlarına kadar inmektedir (Cramer, 2002). Doğrusal regresyon modelinden farkı bağımlı değişkenin iki veya daha fazla sınıftan oluşan kategorik bir değişken olmasıdır (Hosmer ve Lemeshow, 2000). Bağımsız değişkenler ise tıpkı doğrusal regresyonda olduğu gibi kategorik veya sürekli olabilirler (Özdamar, 2004).

Bağımlı değişkenin iki gruptan oluştuğu hali ikili lojistik regresyon olarak adlandırılmasına karşın literatürde sıklıkla lojistik regresyon olarak kullanılır. ÇLR (Çok Terimli Lojistik Regresyon) ise ikili lojstik regresyonun basit bir uzantısıdır (Bayaga, 2010), (Kwak ve Clayton-Matthews, 2002), (Holmes ve Held, 2006). Literatürde polychotomous veya polytomous lojistik regresyon olarak da karşımıza çıkabilen ÇLR, ünlü Amerikalı ekonometrist Daniel McFadden'ın (1974) yılında yaptığı bir çalışmada lojistik regresyonun düzenlenmiş bir şekli olarak önerdiği "Ayrık Seçim Modeli (discrete choice model)" ismiyle de özellikle işletme ve ekonometri gibi alanlarda kullanılmaktadır (Hosmer ve ark., 2013).

ÇLR bağımlı değişkenin, bir başka deyişle çıktı değişkeninin ikiden fazla gruptan meydana geldiği durumlarda kullanılır. Örneğin bağımlı değişkenin bir hastalığın ilerleme derecesi olduğunu varsayalım, eğer bu ilerleme derecesi "başlangıç", "orta", "ileri" gibi ikiden fazla değer alıyorsa bu durumda ÇLR kullanılabilir. Bu durum Y=1,2,3

olarak kabul edildiğinde, kategorilerden birisi referans kategorisi olarak belirlenir ve geri kalan kategoriler ile ikili lojistik modeller kurulur.

Bağımlı değişken, bağımsız değişkenlerin bir fonksiyonu olarak ifade edilerek bağımlı değişkenin kategorilerine toplamları 1 eden birer olasılık değeri atanır. Burada amaç bu olasılık değerleri içerisinde gerçekleşen sonuç olan kategorinin olasılığının en yüksek olarak hesaplanmasıdır. Bu durumda tahmin, sınıflandırma ya da kümeleme problemi doğru neticelendirilmiş olacaktır.

J adet gruptan oluşan bağımlı bir değişkenimiz olduğunu düşünürsek herhangi bir kategori referans kategorisi olarak seçilip *J*-1 adet lojit model kurulur.

$i.$ birime ait bağımlı değişkenin *J* farklı değer aldığı varsayılsın:

$$Y_i = \begin{cases} 1 & p_{i1}\ olasılıkla \\ 2 & p_{i2}\ olasılıkla \\ \vdots & \vdots \\ J & p_{ij}\ olasılıkla \end{cases}$$

$x_{i1},\ldots,x_{iK}$, $i.$ birime ait bağımsız değişkenlerdir. Y_i iki değer aldığında ($J = 2$) için lojistik regresyon modeli (4.8)'de gösterildiği şekilde tanımlanır.

$$P\left[Y_i = 1 \middle| x_{i1}, x_{i2}, \ldots, x_{iK}\right] = \frac{e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_K x_{iK}}}{1+e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_K x_{iK}}} \tag{4.8}$$

Lojit model ise (4.9)'da gösterildiği şekilde yazılır.

$$\log\left[\frac{P\left[Y_i = 1 \middle| x_{i1},\ldots,x_{iK}\right]}{P\left[Y_i = 2 \middle| x_{i1},\ldots,x_{iK}\right]}\right] = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_K x_{iK} \tag{4.9}$$

$J > 2$ için çok terimli lojistik regresyon modeli,

$$\log\left[\frac{P\left[Y_i = 1 \middle| x_{i1},\ldots,x_{iK}\right]}{P\left[Y_i = J \middle| x_{i1},\ldots,x_{iK}\right]}\right] = \beta_{10} + \beta_{11} x_{i1} + \cdots + \beta_{1K} x_{iK} = \boldsymbol{\beta}_1' \mathbf{x}_i$$

$$\log\left[\frac{P\left[Y_i = 2 \middle| x_{i1},\ldots,x_{iK}\right]}{P\left[Y_i = J \middle| x_{i1},\ldots,x_{iK}\right]}\right] = \beta_{20} + \beta_{21} x_{i1} + \cdots + \beta_{2K} x_{iK} = \boldsymbol{\beta}_2' \mathbf{x}_i$$

$$\vdots$$

$$\log\left[\frac{P\left[Y_i = j \middle| x_{i1},\ldots,x_{iK}\right]}{P\left[Y_i = J \middle| x_{i1},\ldots,x_{iK}\right]}\right] = \beta_{j0} + \beta_{j1} x_{i1} + \cdots + \beta_{jK} x_{iK} = \boldsymbol{\beta}_j' \mathbf{x}_i$$

$$\vdots$$

$$\log\left[\frac{P\left[Y_i = J-1 \middle| x_{i1},\ldots,x_{iK}\right]}{P\left[Y_i = J \middle| x_{i1},\ldots,x_{iK}\right]}\right] = \beta_{J0} + \beta_{J1} x_{i1} + \cdots + \beta_{JK} x_{iK} = \boldsymbol{\beta}_J' \mathbf{x}_i$$

şeklinde tanımlanır. Burada $\boldsymbol{\beta}_j^{'} = \left(\beta_{j0}, \beta_{j1}, \ldots, \beta_{jK} x_{iK}\right)$, $j = 1, 2, \ldots, J$ 'dir ve j. lojit model de (4.10)'da gösterildiği biçimde ifade edilebilir.

$$\log\left[\frac{P\left[Y_i = j \middle| x_{i1}, \ldots, x_{iK}\right]}{P\left[Y_i = J \middle| x_{i1}, \ldots, x_{iK}\right]}\right] = \boldsymbol{\beta}_j' \mathbf{x}_i \tag{4.10}$$

Sınıflandırma olasılıkları,

$$p_{ij} = P\left[Y_i = j \middle| x_{i1}, \ldots, x_{iK}\right], \quad j = 1, 2, \ldots, J$$

olmak üzere $J = 2$ için

$$p_{i1} = \frac{\exp\left[\boldsymbol{\beta}' \mathbf{x}_i\right]}{1 + \exp\left[\boldsymbol{\beta}' \mathbf{x}_i\right]}$$

$$p_{i2} = \frac{1}{1 + \exp\left[\boldsymbol{\beta}' \mathbf{x}_i\right]}$$

olup $J > 2$ ve $j < J$ için

$$p_{ij} = \frac{\exp\left[\boldsymbol{\beta}_j' \mathbf{x}_i\right]}{1 + \sum_{j=1}^{J-1} \exp\left[\boldsymbol{\beta}_j' \mathbf{x}_i\right]}$$

ve $j = J$ için ise

$$p_{iJ} = \frac{1}{1 + \sum_{j=1}^{J-1} \exp\left[\boldsymbol{\beta}_j' \mathbf{x}_i\right]} \text{ 'dir.}$$

4.1.3. Destek Vektör Makineleri

DVM'nin (Destek Vektör Makinesi) temelleri Rus Matematikçi Viladimir Vapnik ve arkadaşlarının 1960'lardan 1990'ların ortalarına kadar gerçekleştirdikleri, Vapnik ve Chervonenkis tarafından ortaya atılan istatistiksel öğrenme teorisini (İÖT) temel alan çalışmalara dayanmaktadır (Trafalis ve Ince, 2000; Dibike ve ark., 2001; Scholkopf ve Smola, 2001; Jack ve Nandi, 2002; Samanta ve ark., 2003; Wang, 2005; Xing ve ark., 2005; Han ve ark., 2007; Nagi ve ark., 2008; Oskoei ve Hu, 2008).

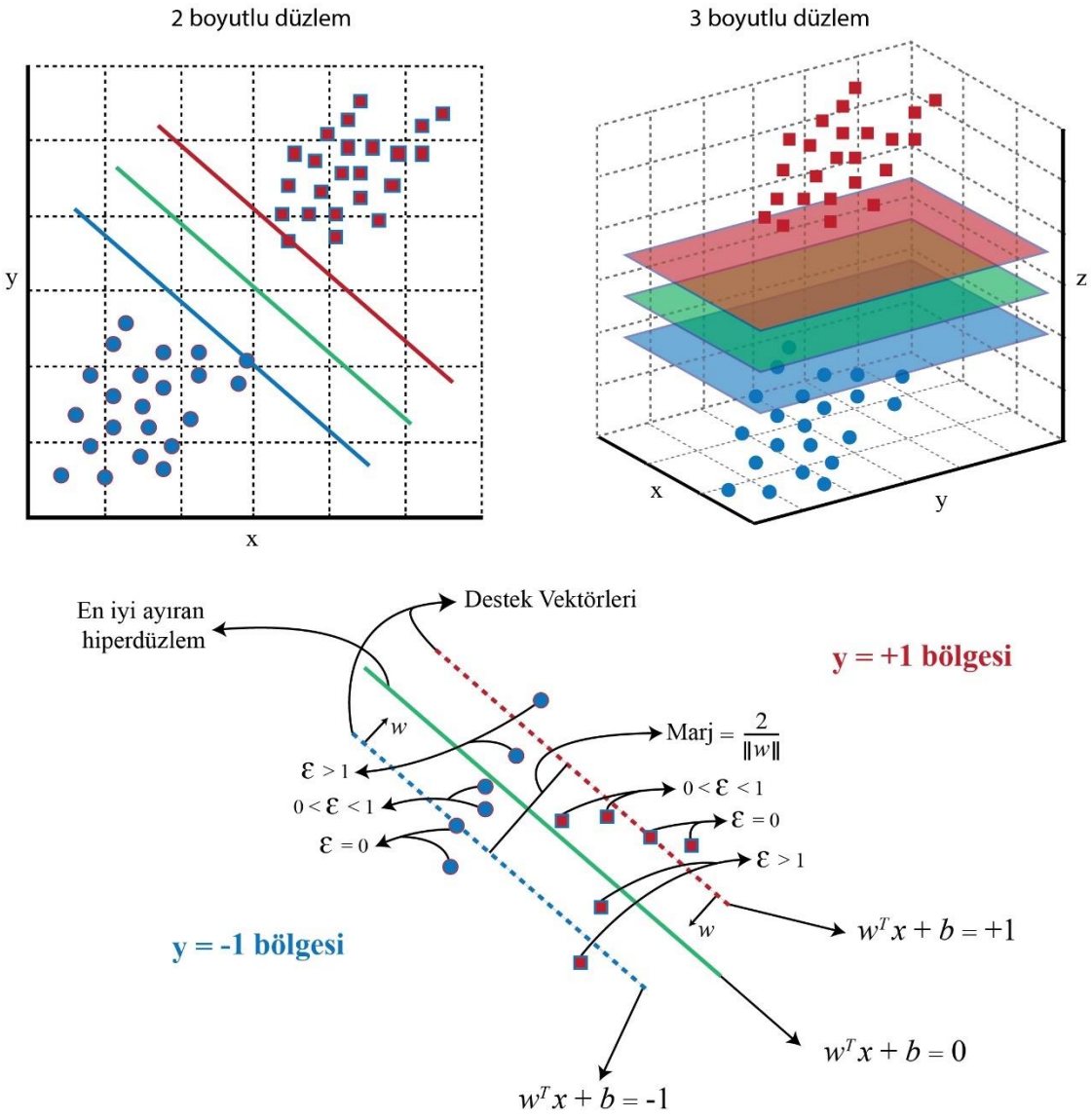
İstatistiksel öğrenme teorisi, günümüzde sıklıkla kullanılan makine öğrenmesi algoritmalarının birçoğunun teorik altyapısını oluşturur (Von Luxburg ve Schölkopf, 2011). Temel amacı eldeki veri setinden, kestirim yapma, karar verme, model kurma vb. bilgi çıkarımı işlemlerini yapabilecek bir çerçeve sunmaktır (Bousquet ve ark., 2003). İÖT, makine öğrenmesinin matematiksel temellerini oluşturan teorik bir dalıdır.

İÖT'nin yanısıra, İngiliz Matematikçi James Mercer (1909) tarafından ortaya atılan çekirdek (kernel) kavramı (Schölkopf ve ark., 2002), (Cristianini ve Shawe-Taylor,

2000) ile Alman Matematikçi David Hilbert tarafından ortaya atılan Hilbert Uzayı (Wahba, 1999) kavramı da DVM'nin teorik temellerine katkı sağlamıştır.

DVM ile ilgili ilk çalışmaların 1960'lara kadar inmesine karşın bu çalışmalarda genelde teoriden öteye gidilememesi, o dönemlerde makine öğrenmesi alanında çalışanların destek vektör makinelerini pratikte çok da uygulanabilir bir yöntem olarak değerlendirmemelerine neden oluyordu. Sonraları 1990'larda Vapnik ve arkadaşlarının destek vektör makinelerini çeşitli sınıflandırma ve örüntü tanıma problemlerinde kullanarak birçok yöntemden daha başarılı sonuçlar elde etmeleri, bu alanda çalışan araştırmacıların da DVM'lere olan bakış açılarının değişmesine neden olmuştur (Huang ve ark., 2006). DVM, bu dönemden sonra çeşitli sınıflandırma, örüntü tanıma, regresyon vb. birçok problemin çözümünde kullanılan güçlü bir yöntem halini almıştır (Osuna ve ark., 1997; Tong ve Koller, 2001; Campbell ve ark., 2006; Press ve ark., 2007).

DVM, temelde iki sınıfa ayrılmak istenen veriyi daha yüksek boyuttaki bir uzaya taşıyarak en uygun hiper düzlem aracılığıyla ayırmayı amaçlar (Lin ve Wang, 2002; Avidan, 2004). Buradaki en uygun sözcüğünün tam karşılığı ayırıcı hiper düzlemin, her iki sınıftan kendisine en yakın gözlem değerlerine maksimum uzaklıkta olmasıdır. Her iki sınıfın ayırıcı hiperdüzleme en yakın noktaları üzerinden destek vektörleri geçer. DVM'de bu destek vektörleri arasındaki mesafeyi maksimize edecek olan hiper düzlem elde edilmeye çalışılır (Dash ve ark., 2006; Karimi ve ark., 2006). DVM'lerin doğrusal olarak ayrılan veriler için çalışma mantığı Şekil 4.2'de gösterilmektedir.



Şekil 4.2 DVM'de doğrusal ayrılabilme durumu

S bir eğitim kümesi olsun,

$S = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$

$(x_i, y_i)_{i=1}^{n}$ için $x_i \in \mathbb{R}^d$ ve $y_i \in \{-1, +1\}$

$\mathbf{w}$: ağırlık vektörü,

b : yanlılık (bias), olmak üzere,

$y_i = 1$ için $\mathbf{w}^T x_i + b \geq 1$ ve $y_i = -1$ için $\mathbf{w}^T x_i + b \leq 1$, eşitlikleri sağlanıyor ise S kümesindeki örnekler doğrusal olarak ayrılabiliyor demektir.

Bu eşitlikler genellendiğinde (4.11) elde edilir.

$$y_i \left(w^T x_i + b \right) \geq 1, \quad i = 1, 2, ..., n \tag{4.11}$$

Doğrusal olarak ayrılabilen S eğitim kümesindeki örnekleri en iyi ayıran hiperdüzleme paralel olan ve her iki sınıfın düzleme en yakın noktalarından geçen destek vektörlerinin fonksiyonları,

$f(x) = \mathbf{w}^T x + b = +1$ ve $f(x) = \mathbf{w}^T x + b = +1$, şeklindedir.

S eğitim kümesini en iyi ayıran hiperdüzlemin fonksiyonu ise

$f(x) = \mathbf{w}^T x + b = 0$, şeklindedir.

Eğitim kümesinin doğrusal olarak ayrılamadığı durumlarda ise çekirdek hilesi denilen bir yöntem kullanılır. Burada yapılan işlem; çekirdek fonksiyonları denilen ve giriş verisinin kendinden daha yüksek boyutlu bir uzaya taşınarak sınıflandırılmasını sağlayan fonksiyonlar kullanılmasıdır.

4.1.4. K En Yakın Komşu Algoritması

KEYK (*K* En Yakın Komşu) sınıflandırma algoritması, olasılık yoğunluklarının güvenilir parametrik tahminlerinin yapılamadığı durumlarda da başarılı sınıflandırmalar gerçekleştirmek amacıyla ortaya çıkmıştır (Peterson, 2009). Parametrik olmayan bu yöntem Fix ve Hodges'in (1951) tarafından sınıflandırma amacıyla geliştirilmiş ve en yakın komşu kuralı olarak adlandırılmıştır. Daha sonra Cover ve Hart (1967) yöntemin biçimsel bir takım özelliklerini ve sınırlılıklarını belirlemişlerdir. KEYK günümüzdeki halini alıncaya kadar çok sayıda bilim insanı (Hellman, 1970), (Fukunaga ve Hostetler, 1975), (Dudani, 1978), (Bailey ve Jain, 1978), (Jóźwik, 1983), (Keller ve ark., 1985), (Bermejo ve Cabestany, 2000) tarafından geliştirilmiştir.

KEYK oldukça basit bir yapıda olmasına rağmen son derece etkili bir sınıflandırma yöntemidir (Guo ve ark., 2003), (Song ve ark., 2007). Bir gözlemin sınıflandırabilmesi için ilk olarak o gözlemin veri setindeki diğer tüm gözlemlere olan uzaklıkları ölçülerek bu uzaklıklar küçükten büyüğe sıralanır. Söz konusu bu uzaklıkların hesaplanabilmesi için en yaygın şekilde kullanılan yöntem oklid uzaklığı'dır (Hoi ve ark., 2006), (Hu ve ark., 2016), (Manjusha ve Harikumar, 2016).

Küçükten büyüğe sıralanan bu uzaklık ölçülerinden *K* tanesi seçilir. Bu *K* değerinin kaç olacağı konusunda kesin bir kural yoktur, mevcut veriye göre değişiklik gösterir ve genelde test sonuçlarına bakarak deneme yanılma ile değiştirilir (Song ve ark., 2007; Hassanat ve ark., 2014). Böylelikle *K* adet gözlem belirlenmiş olur, bu *K* adet gözlem içerisinde hangi sınıf çoğunlukta ise yeni gelen gözlem de o sınıfta etiketlenir.

A ve B noktaları arasındaki oklid uzaklığı (4.12)'te gösterilmektedir;

$$\mathbf{a} = (a_1, a_2, \ldots, a_n), \qquad \mathbf{b} = (b_1, b_2, \ldots, b_n)$$
$$u(A,B) = u(B,A) = \sqrt{(a_1 - b_1)^2 + (a_1 - b_1)^2 + \cdots + (a_n - b_n)^2} \tag{4.12}$$
$$= \sqrt{\sum_{i=1}^{n} (a_i - b_i)^2}$$

KEYK algoritmasını aşağıdaki gibi bir örnekle de açıklayabiliriz.

Çizelge 4.1. KEYK de kullanılacak örnek veri

Gözlem	x_1	x_2	x_3	y
1	0.71	0.42	0.38	2
2	0.73	0.57	0.65	1
3	0.68	0.41	0.23	2
4	0.04	0.98	0.80	2
5	0.85	0.55	0.49	2
6	0.39	0.74	0.51	1
7	1.00	0.50	0.26	2
8	0.49	0.62	0.57	1
9	0.90	0.60	0.76	2

Çizelge 4.1'de verilen örnek veriler ve sınıf etiketleri kullanılarak, $A = (0.23, 0.79, 0.60)$ noktası sınıflandırılmak istenirse;

Adım 1. Tüm noktalar ile olan uzaklık hesaplanır.

$$U_1 = \sqrt{(0.23 - 0.68)^2 + (0.79 - 0.42)^2 + (0.60 - 0.38)^2} = 0.6447$$

$$U_2 = \sqrt{(0.23 - 0.73)^2 + (0.79 - 0.57)^2 + (0.60 - 0.65)^2} = 0.5485$$

$$U_3 = \sqrt{(0.23 - 0.68)^2 + (0.79 - 0.41)^2 + (0.60 - 0.23)^2} = 0.6956$$

$$U_4 = \sqrt{(0.23 - 0.04)^2 + (0.79 - 0.98)^2 + (0.60 - 0.80)^2} = 0.3350$$

$$U_5 = \sqrt{(0.23 - 0.85)^2 + (0.79 - 0.55)^2 + (0.60 - 0.49)^2} = 0.6739$$

$$U_6 = \sqrt{(0.23 - 0.39)^2 + (0.79 - 0.74)^2 + (0.60 - 0.51)^2} = 0.1903$$

$$U_7 = \sqrt{(0.23 - 1.00)^2 + (0.79 - 0.50)^2 + (0.60 - 0.26)^2} = 0.8903$$

$$U_8 = \sqrt{(0.23 - 0.49)^2 + (0.79 - 0.62)^2 + (0.60 - 0.57)^2} = 0.3221$$

$$U_9 = \sqrt{(0.23-0.90)^2 + (0.79-0.60)^2 + (0.60-0.76)^2} = 0.7146$$

Adım 2. Bulunan uzaklıklar Çizelge 4.2'de gösterildiği gibi küçükten büyüğe sıralanır.

Çizelge 4.2. Adım – 1'de elde edilen uzaklık değerleri

Gözlem	Uzaklık	Etiket
6	0.1903	1
8	0.3121	1
4	0.3350	2
2	0.5485	1
1	0.6447	2
5	0.6739	2
3	0.6956	2
9	0.7146	2
7	0.8903	2

Adım 3. *K* değerinin belirlenmesi ve sınıflandırma.

K değeri 3 olduğunda; uzaklık ölçüsü olarak en yakın olan 3 noktanın sınıf etiketlerine bakılırsa yeni gözlem "1" olarak sınıflandırılır,

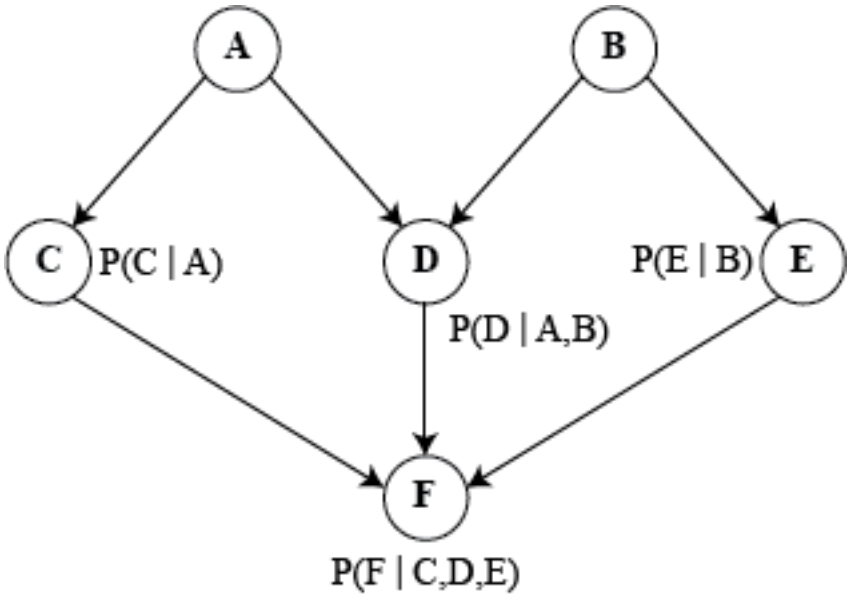
K değeri 5 olduğunda; "1" olarak,

K değeri 7 olduğunda ise "2" olarak sınıflandırılır.

Burada dikkat edilmesi gereken nokta, KEYK sınıflandırıcının performansını en yükseğe çekebilmek adına farklı *K* değerleri için de denemeler yapılması gerektiğidir. *K* değerinin yanı sıra, veri setinin boyutu, gözlemlerin dağılımı, aykırı değerler vb. faktörler de KEYK sınıflandırıcısının performansını doğrudan etkilerler.

4.1.5. Sade Bayes

Sınıflandırma için oldukça sık kullanılan SB (Sade Bayes) yönteminin temeli, bayes teoremi veya bayes kuralı olarak da bilinen basit bir olasılık teoremine dayanır (Lewis, 1998). SB, bayes ağlarının basit bir çeşididir (Domingos ve Pazzani, 1997), (Zhang, 2004), (Amor ve ark., 2004). Bayes Ağları, yönlü düz ağaçlar kullanılarak rasgele değişkenlerin, koşullu olasılıkları ile birlikte gösterilmesini sağlayan olasılıksal grafik modelleridir (Leu ve Chang, 2013). Şekil 4.3'te bu modele bir örnek gösterilmektedir.



Şekil 4.3. Bayes ağı modeli

Bayes ağlarında, olasılıksal akıl yürütme olarak da bilinen bir işlem sayesinde diğer tüm değişkenler, bilinmeyen bir değişkenin sonsal olasılığını tahmin etmede kullanılır (Torres-Toledano ve Sucar, 1998). Bayes ağlarında gerçekleştirilen bu hesaplamalar, temel olasılık kurallarından birisi olarak kabul edilen bayes teoremini de ortaya atmış olan Thomas Bayes'in koşullu olasılık ile ilgili çalışmalarını temel almaktadır.

Koşullu Olasılık ve Bayes Teoremi

A ve *B* iki olay olsun; *P(A)*, *A* olayının *P(B) ise B* olayının gerçekleşme olasılığıdır.

B olayı gerçekleştiğinde *A* olayının gerçekleşmesi olasılığı,

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)},$$

A olayı gerçekleştiğinde *B* olayının gerçekleşmesi olasılığı,

$$P(B \mid A) = \frac{P(B \cap A)}{P(A)}$$

şeklindedir.

Bu denklemler ile $P(A \cap B) = P(B \cap A)$ denklemi birleştirildiğinde;

$$P(A \mid B)P(B) = P(A \cap B),\ P(B \mid A)P(A) = P(B \cap A),$$

$$P(A \mid B)P(B) = P(B \mid A)P(A),$$

eşitliği elde edilir. Bu eşitlik yardımıyla da,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

eşitliği elde edilir.

Bu yapının kullanıldığı SB sınıflandırıcısının çalışma mantığı ise aşağıda açıklanmıştır.

$x_1, x_2, ..., x_n$ öznitelikler, y ise sınıf etiketi olsun.

$i = 1, \ldots, n,\ j = 1, \ldots, k$ iken,

$P(X_i = x_i \mid Y_j = y_j) = \frac{P(Y_j = y_j \mid X_i = x_i)P(X_i = x_i)}{P(Y_j = y_j)}$, sınıf etiketinin y_j olduğunun bilindiği durumda x_i özniteliğinin var olması olasılığı,

$P(Y_j = y_j \mid X_i = x_i) = \frac{P(X_i = x_i \mid Y_j = y_j)P(Y_j = y_j)}{P(X_i = x_i)}$, x_i özniteliğinin bilinmesi durumunda sınıf etiketinin y_j olması olasılığıdır.

Buradan hareketle *n* adet özniteliğin var olduğu bir örnekte aşağıdaki gibi bir bağıntı kurabiliriz.

$$P(Y_k = y_k \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = P(X_1 = x_1 \mid X_2 = x_2, X_3 = x_3, ..., X_n = x_n, Y_k = y_k)$$
$$\times P(X_2 = x_2 \mid X_3 = x_3, X_4 = x_4, ..., X_n = x_n, Y_k = y_k) \ldots$$
$$\times P(X_{n-1} = x_{n-1} \mid X_n = x_n, Y_k = y_k) P(X_n = x_n \mid Y_k = y_k) P(Y_k = y_k)$$

Tüm özniteliklerin *Y* sınıfında koşullu olarak birbirinden bağımsız oldukları varsayıldığında (4.13) elde edilir.

$$P(X_i = x_i \mid X_{i+1} = x_{i+1}, X_{i+2} = x_{i+2}, ..., X_n = x_n, Y_k = y_k) = P(X_i = x_i, Y_k = y_k) \qquad (4.13)$$

Sonrasında ise (4.14)'te gösterildiği gibi genelleştirilmiş bir denklem oluşturulabilir.

$$P(Y_k = y_k \mid X_1 = x_1, X_2 = x_2, ..., X_n = x_n) = \frac{P(Y_k = y_k)\prod_{i=1}^{n} P(X_i = x_i \mid Y_k = y_k)}{P(X_1 = x_1, X_2 = x_2, ..., X_n = x_n)} \qquad (4.14)$$

Bu denklem aracılığıyla tüm özniteliklerin bilindiği durumda sınıf etiketlerinin her biri için ayrı ayrı olasılık değerleri hesaplanabilir. Hesaplamalar sonucunda en yüksek olasılığa ulaşılan sınıf etiketi geçerli sınıf etiketi olarak atanır.

SB sınıflandırıcısı işlemsel olarak oldukça basit yapıda olmasına karşın son derece etkili bir sınıflandırma modelidir (Kantarcıoglu ve ark., 2003). Temelde özniteliklerin birbirinden bağımsız olduğu varsayımına dayanmasına karşın istatistiksel olarak bağımsız olmayan özniteliklerin var olduğu veri setlerinde de başarılı sonuçlar verebilmektedir. Metin sınıflandırma, medikal teşhis, örüntü tanıma, özellik seçimi ve çıkarımı gibi birçok türdeki problemlerin çözümünde etkin şekilde kullanılmaktadır (Rish, 2001; Metsis ve ark., 2006; Pattekari ve Parveen, 2012; Feng ve ark., 2018).

4.1.6. Rasgele Orman

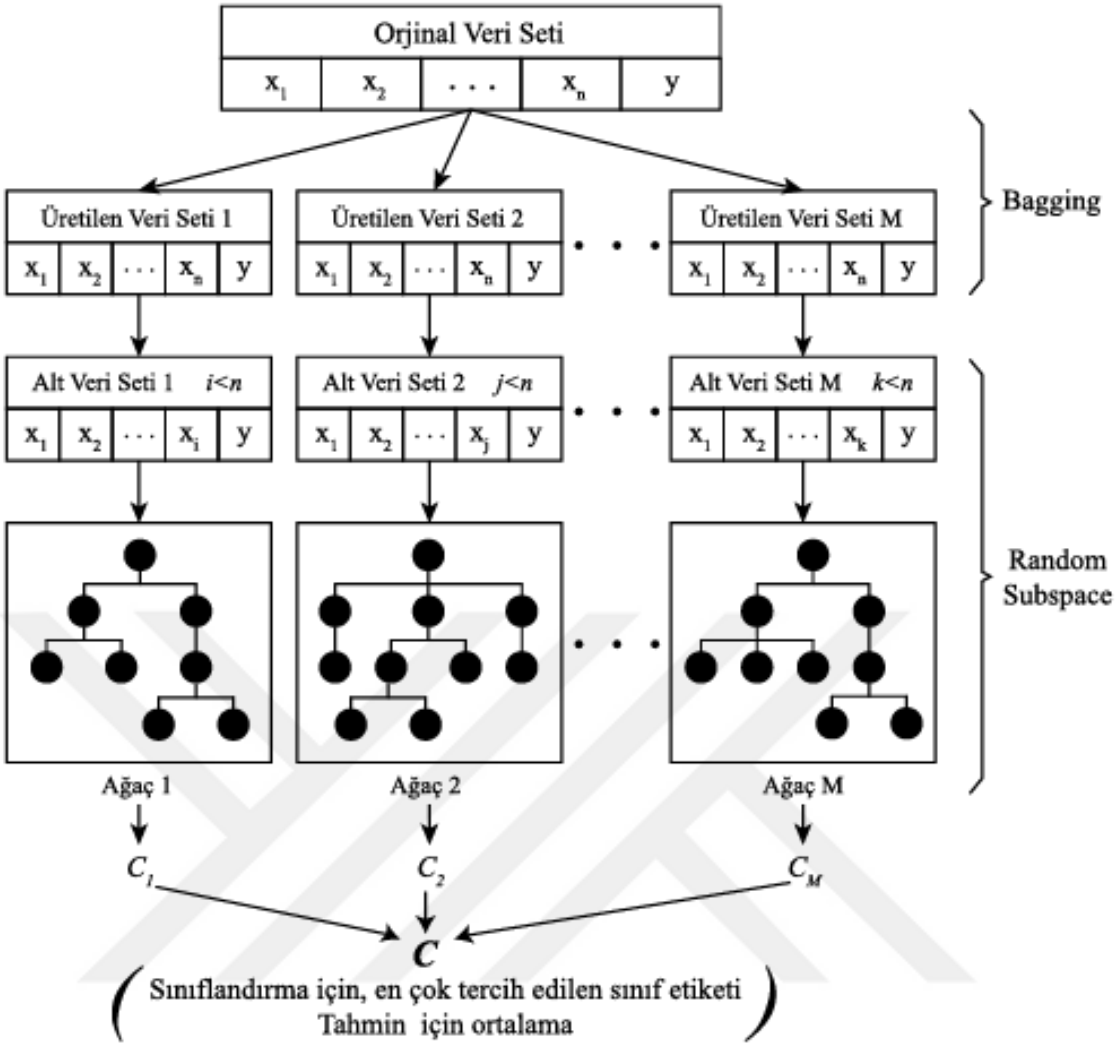
Rasgele orman (RO), mevcut verilerden bootstrap (yeniden örnekleme) ile elde edilmiş bağımsız ve aynı dağılımlı Θ_k vektörlerinin kullanılmasıyla oluşturulan karar ağaçlarının bir kombinasyonudur (Breiman, 2001).

$$\left\{h\left(x,\Theta_k\right),k=1,2,...\right\}$$

Orijinal veri setinden yeni veriler üretip bunları da farklı birer karar ağacı yapısında kullanma fikri Breiman (1996) tarafından ortaya atılmıştır. Breiman, bagging (bootstrap aggregating) ismini verdiği bu teknik ile farklı veri setleri üzerinde testler gerçekleştirmiş ve doğruluğun arttığını göstermiştir.

RO'nın şekillenmesine katkı sağlayan önemli kavramlardan birisi de Ho (1998) tarafından sunulan 'Random Subspace' tekniğidir. Random Subspace tekniği orijinal veri setinden rasgele şekilde seçilen belli sayıdaki öznitelik ile yeni karar ağaçları oluşturulması ve bu karar ağaçlarını en iyi dallara ayıracak olan özniteliklerin belirlenmesidir.

RO, Breiman tarafından önerilen 'bagging' ve Ho tarafından önerilen 'random subspace' tekniklerinin karması şeklinde oluşturulan bir yöntemdir. Orijinal veri setinden bagging yöntemi ile türetilen yeni veri setleri ile çok sayıda karar ağacı yapısı oluşturulur. Bu karar ağaçları oluşturulurken her bir karar ağacı için random subspace yöntemi ile rasgele seçilmiş farklı sayılardaki değişkenler kullanılır. Karar ağaçlarının her birinden elde edilen sonuçlara göre ortalama bir karar verilir ve bu karar RO'nun genel kararı olarak sunulur. Bu ortak karar, sınıflandırma amaçlı çalışmalarda fazla sayıda olan sınıf etiketine, regresyon amaçlı yapılan çalışmalarda ise karar ağaçlarından gelen tahminlerin ortalamasına göre hesaplanır. RO yöntemi ile gerçekleştirilen bir sınıflandırmanın yapısı Resim 4.1'de gösterilmektedir.



Resim 4.3. Rasgele orman sınıflandırıcısının çalışma mantığı

RO, sınıflandırmanın yanı sıra değişkenlerin önem derecesinin belirlenmesinde de kullanılır (Breiman, 2001; Millard ve Richardson, 2015; Cerrada ve ark., 2016). Genellikle çok fazla sayıda değişkenin kullanıldığı çalışmalarda değişkenler içerisinden önemli olanların seçilmesi modelin başarısı açısından oldukça önemlidir. RO bu özellikleri sayesinde veri setini indirgemek için kullanılan özellik seçimi ve özellik çıkarımı yöntemlerine de bir alternatif olabilir. RO'da değişkenlerin önem derecesinin belirlemek adına iki farklı yöntem kullanılır. Bunların ilki her bir değişken üzerinde gerçekleştirilen karıştırma işlemi sonucunda ağaçlarda elde edilen doğruluk değişimlerinin ortalamasının alınması ile gerçekleştirilir. İkincisi ise her değişken için ağaç yapıları üzerinde elde edilen Gini skorlarının değişimlerinin ortalaması ile hesaplanır. Bu yöntemler detaylıca açıklanacak olursa;

Yöntem 1: $x_1, x_2, ..., x_n, y$ test verisi olsun. $a_1, a_2, ..., a_n$ de ormandaki ağaç sayısını temsil etsin.

Adım 1. Test verisi ormandaki her bir ağaca ayrı ayrı yerleştirilerek doğru sınıflandırma sayısı hesaplanır.

Adım 2. Sonrasında her bir *x* değişkeni tek tek ele alınarak kendi içerisinde karıştırılır ve oluşan yeni test verileri ormandaki tüm ağaçlarda tekrar test edilir.

Adım 3. *x* değişkeninin karıştırılmamış hali ile her bir *a* ağacında elde edilen doğru sınıflandırma sayısından x değişkeninin karıştırılmış hali sonrası elde edilen doğru sınıflandırma sayıları çıkarılır.

Adım 4. Bu işlem tüm değişkenler için tekrarlanır ve her ağaçtan elde edilen doğru sınıflandırma değişimlerinin ortalaması hesaplanır.

Adım 5. Her değişken için ağaçlardan elde edilen değişim miktarlarının birbirinden bağımsız olduğu ve normal dağıldığı varsayılarak standart hata değeri hesaplanır.

Adım 6. Son olarak her değişken için değişim ortalaması standart hata değerine bölünerek önem derecesi bulunur.

Yöntem 2: $x_1, x_2, ..., x_n, y$ test verisi olsun. $a_1, a_2, ..., a_n$ de ormandaki ağaç sayısını temsil etsin.

Bu yöntemde her x değişkeni için her a ağacında dallanma öncesi ve sonrası (4.15)'da gösterildiği gibi *gini* skorları hesaplanır. Bir *x* değişkeninin *gini* önem derecesini hesaplamak için, ilgili değişkenin her ağaçta ayrı ayrı dallanma öncesi *gini* skoru ile dallanma sonrası *gini* skoru arasında fark alınır. Her ağaçtan elde edilen bu farkların toplamı ilgili x değişkeninin *gini* önem derecesini gösterir.

$$Gini_{sol} = 1 - \sum_{i=1}^{k} \left[\frac{L_i}{|T_{sol}|} \right]^2 \qquad Gini_{sağ} = 1 - \sum_{i=1}^{k} \left[\frac{R_i}{|T_{sağ}|} \right]^2$$
$$Gini = \frac{1}{n} \left(|T_{sol}| Gini_{sol} + |T_{sağ}| Gini_{sağ} \right) \tag{4.15}$$

k : sınıf etiketi sayısı

T : bir düğümdeki örnek sayısı

T_{sol} : sol daldaki örnek sayısı

$T_{sağ}$: sağ daldaki örnek sayısı

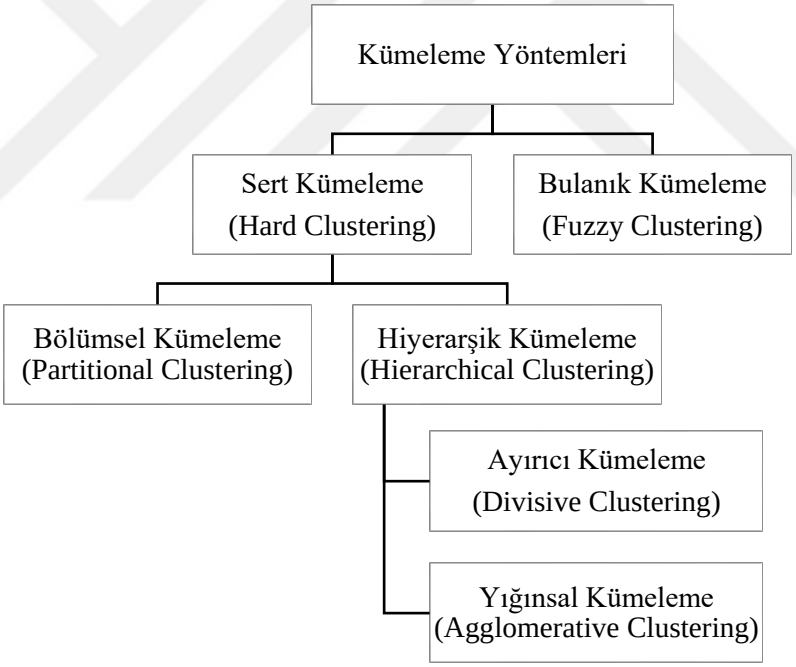
L_i : sol dalda i sınıf etiketine sahip örneklerin sayısı

R_i : sağ dalda i sınıf etiketine sahip örneklerin sayısı

RO yöntemi sınıflandırma, tahmin, özellik seçimi gibi başlıklarda pek çok alanda çeşitli problemlere cevap aramak amacıyla kullanılmıştır. Literatürde hem tek başına kullanıldığı hem de diğer yöntemler ile karşılaştırıldığı çok sayıda çalışmaya rastlanmaktadır (Wang ve ark., 2015; Masetic ve Subasi, 2016; Muchlinski ve ark., 2016; Chen ve ark., 2017; Zhu ve ark., 2018).

4.2. Kümeleme

Kümeleme eldeki örnekleri özel bir problem dahilinde ele alarak anlamlı bir şekilde alt kümelere ayırma işlemidir (Jain ve Dubes, 1988). Genellikle benzer özellikteki örnekler aynı küme altında toplanır (Abbas, 2008). Kümeleme yöntemlerini Şekil 4.4'te gösterildiği gibi temelde iki gruba ayrılabilir. Bu gruplardan ilki literatürde "Hard Clustering" şeklinde geçen net (sert) kümeleme, diğeri ise literatürde "Fuzzy Clustering" ismiyle rastladığımız bulanık (yumuşak) kümelemedir (Mirkin, 2005).



Şekil 4.4. Kümeleme Yöntemleri (Gan ve ark., 2007)

Sert kümelemede her bir örnek sadece bir kümenin elemanı olurken, bulanık kümelemede bir örnek farklı üyelik dereceleri ile birden çok kümenin elemanı olabilir (Bora ve ark., 2014).

Sert kümeleme algoritmalarında, örneklere ait oldukları kümeleri niteleyen bir l_i sınıf etiketi verilir. Bu etiketlerin kümesi $L = \{1, 2, ..., k\}$ ve $\forall l_i \in L$ 'dir. Buradaki k

örneklemin ayrıldığı küme sayısını gösterir. Bu sayı bir uzman tarafından sabitlenebilir ya da kullanılan kümeleme algoritması ile elde edilir. Kümeleme analizine dayalı çalışmalarda en önemli tartışma konularının başında k küme sayısının kaç olacağı gelir. Literatürde en optimum sonuca ulaşmak için gerekli olan k küme sayısının değerinin belirlenebileceği çeşitli yöntemler vardır (Kodinariya ve Makwana, 2013).

4.2.1. *K*-Ortalamalar Kümeleme

MacQuenn tarafından 1967 yılında önerilen KO (K ortalamalar) kümeleme algoritmasında amaç bir veri setini k sayıda alt kümeye ayırmaktır (MacQueen, 1967). Bu algoritma temelde iki adımdan oluşur. İlk adımda önceden belirlenmiş bir k sayısı kadar rasgele merkez noktası belirlemeye yönelik işlemler, ikinci adımda ise veri setindeki her bir örneği kendisine en yakın merkez noktasına atamaya yönelik işlemler gerçekleştirilir (Na ve ark., 2010).

Küme sayısını ifade eden *k* değerinin doğru belirlenebilmesi kümeleme algoritmasının performansını doğruda etkiler, bu nedenle sabit bir *k* değeri yerine bir *k* değerler kümesi kullanılabilir (Pham ve ark., 2005).

Veri setindeki tüm örnekleri, önceden belirlenmiş merkez noktalarının temsil ettiği kümeler arasından, kendilerine en yakın olanına atamak için bir uzaklık ölçümü yapılır. Buradaki uzaklık her bir örneğin küme merkezlerine olan mesafesini ifade etmektedir. Bu uzaklıkların doğru şekilde belirlenebilmesi *k* ortalamalar kümeleme algoritması için oldukça önemli bir aşamadır ve bu amaçla çeşitli uzaklık ölçüm fonksiyonları kullanılmaktadır (Loohach ve Garg, 2012). Bunlardan bazıları, Euclidean, Manhattan, Minkowski, Cosine uzaklık ölçüm fonksiyonlarıdır (Pandit ve Gupta, 2011; Patel ve Mehta, 2012; Gomathi ve ark., 2014).

KO kümeleme yönteminin aşamaları şu şekildedir;

Adım 1. *k* değeri belirlenir.

Adım 2. Veri setinden *k* değeri kadar merkez nokta seçilir.

Adım 3. Tüm örneklerin merkez noktalarına olan uzaklıkları ölçülür.

Adım 4. Her örnek kendisine en yakın mesafedeki merkez noktası ile temsil edilen kümeye aktarılır. (İkinci ve daha sonraki tekrarlarda oluşan kümeleme bir önceki kümeleme ile aynı ise döngü Adım 4'te sonlandırılır, eğer farklı bir kümeleme oluşuyorsa devam edilir.)

Adım 5. Her küme içerisine düşen noktaların ortalamaları alınarak kümelerin yeni merkezleri belirlenir.

Adım 6. Yeni merkezlere göre Adım 3'ten devam edilir.

4.2.2. Bulanık *C* Ortalamalar Kümeleme

Bulanık kümeleme algoritmalarının, her bir örneğin tek kümeye ait olabileceği varsayımına dayanan diğer kümeleme algoritmalarından temel farklılığı, örneklerin belli üyelik dereceleri ile birden çok kümenin elemanı olabilmeleridir. Literatürde birçok bulanık kümeleme algoritması vardır (Höppner ve ark., 1999; Ming ve ark., 2011; Shieh, 2014; Patil ve Dongre, 2015). BCO (Bulanık C ortalamalar) kümeleme algoritması bu algoritmalar içerisinde en yaygın kullanılanlardan bir tanesidir (Hung ve Yang, 2001; Bezdek, 2013). Tıp, coğrafi bilgi sistemleri, görüntü işleme ve birçok alanda sınıflandırma, kümeleme ve özellik seçimi işlemlerinde sıklıkla kullanılmaktadır (Chuang ve ark., 2006; Cai ve ark., 2007; Di Martino ve Sessa, 2009; Sopharak ve ark., 2009; Hung ve ark., 2010; Polat, 2012; Rustempasic ve Can, 2013).

İlk olarak Dunn (Dunn, 1973) tarafından sunulan algoritma daha sonra çeşitli çalışmalar (Bezdek, 1973; Gustafson ve Kessel, 1979; Bezdek ve ark., 1981) ile geliştirilmiştir. KO kümeleme algoritması ile karşılaştırıldığında temel farklılık her bir örneğin tüm kümelere 0 ile 1 arasında bir üyelik derecesi ile ait olmasıdır. Bir örneğin tüm sınıflara olan üyelik derecelerinin toplamı ise 1 olmalıdır.

C küme sayısı olsun, *A* ise *N* tane örnekten oluşan bir veri seti olsun.

$$A = \{x_1, x_2, ..., x_n\},$$

A kümesinin her bir elemanının *C* adet kümeden her birine aitliğini belirten üyelik dereceleri matrisi ise,

$U = \left[u_{ij}\right],\ \ i = (1,2,..N),\ \ j = (1,2,...,C),\ \ 0 \le u_{ij} \le 1$, şeklinde gösterilir.

c_j kümelerin merkezi, *m* ise bulanıklaştırma katsayısı olmak üzere, algoritmanın uygulama adımları şu şekildedir.

Adım 1. Başlangıç üyelik dereceleri rasgele belirlenir $U = \left[u_{ij}\right]$,

Adım 2. Üyelik derecelerine göre merkez noktaları (4.16)'deki gibi belirlenir

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} . x_i}{\sum_{i=1}^{N} u_{ij}^{m}}\ , \tag{4.16}$$

Adım 3. Tüm örneklerin merkez noktalarına olan uzaklıkları ölçülür,

Adım 4. Yeni üyelik dereceleri (4.17)'e göre hesaplanır

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}}, \tag{4.17}$$

Adım 5. Yeni üyelik derecelerine göre merkezler yeniden hesaplanır. Merkezler sabit kalıncaya kadar ya da merkezler arası fark kabul edilebilir hata oranı ε 'nın altına ininceye kadar adım 2'den başlanarak döngü tekrarlanır.

5. VERİ SETİNİN OLUŞTURULMASI

Verilerden anlam çıkarmaya yönelik tüm çalışmalarda eldeki veri setinin nitelikleri ulaşılacak sonucu doğrudan etkiler. Veri seti oluşturulurken bağımlı değişkeni (hedef) açıklayabilecek, çok sayıda öznitelik (bağımsız değişken) veri setine dahil edilir. Yeterli miktarda veri toplandıktan sonra ise ham veri üzerinde çeşitli ön işlemler ve boyut azaltma çalışmaları gerçekleştirilerek veri setine son şekli verilir.

5.1. Veri Setinin Kapsamı

Bir futbol müsabakasının sonucunu kesitirebilmek için iki takım hakkında çok sayıda bilgiye sahip olmak gerekir. Takımların önceden oynadığı müsabakalarda ortaya koydukları saha içi performans istatistikleri, birbirlerine karşı olan rekabet durumları, maç günü performans tabloları, sakat ve cezalı oyuncuların varlığı, iddaa oranları, taraftarlar görüşleri, anlık gelişmeler vb. ölçülebilen ya da ölçülemeyen çok sayıda etmen bu bilgilere örnek olarak verilebilir.

Bu çalışma kapsamında oluşturulan veri setinde ise toplam 191 adet öznitelik bulunmaktadır. Bu öznitelikleri büyük çoğunluğu çeşitli web platformlarından elde edilirken belli sayıda öznitelik birtakım hesaplamalar sonucu oluşturulmuştur.

Bu veri setinin oluşturulabilmesi amacıyla, aşağıda kısaca bilgileri verilmiş olan, Türkiye, İngiltere, İspanya, Almanya, İtalya ve Fransa'nın en üst düzey liglerinde oynanan maçlar incelenmiştir.

Türkiye Süper Ligi

1959'da Türkiye 1. Futbol ligi adı ile kurulmuştur (TFF, 2019). Türkiye'deki futbol liglerinin en üst basamağını oluşturan lig 2002-03 sezonundan itibaren Türkiye Süper Ligi adını almıştır. 18 takım mücadele etmektedir.

İngiltere Premier League

1888'de kurulan İngiltere Futbol Liglerinin 1992-93 sezonunda gerçekleştirilen değişiklik ile en üst kademesi İngiltere Premier League olarak adlandırılmaktadır (Footballhistory, 2019a). 20 takım mücadele etmektedir.

İspanya La Liga

1929'dan beri İspanya'nın en üst kademe futbol liginin adı La Liga olarak anılmaktadır. 1997-98 sezonundan itibaren 20 takım mücadele etmektedir (Footballhistory, 2019b).

Almanya Bundesliga

Bundesliga Almanya'daki tüm futbol liglerini kapsayan bir yapı olarak 1962'de kurulmuştur. Bünyesinde 5 lig bulunan bir yapının ismi olan Bundesliga genel olarak en üst düzey lig olan 1. Ligi tanımlamak için kullanılır (Wikimedia, 2019). Bu ligde 18 takım mücadele etmektedir.

İtalya Seria A

1898'de kurulmuştur ve İtalyan futbolunun en üst düzey ligini ifade eder. Zaman içerisinde 16 ile 21 arası değişik sayılarda takımların mücadele ettiği lig, 2004 yılından itibaren günümüzdeki 20 takımın mücadele ettiği halini almıştır (Footballhistory, 2019c).

Fransa Ligue 1

1932'de Fransa 1. Futbol Ligi ismiyle kurulmuş olup 2002'den beri Ligue 1 ismi ile anılmaktadır (Footballhistory, 2019d). Fransa'nın en üst seviye futbol ligi olan Ligue 1'de 20 takım mücadele etmektedir.

Bu liglerde, sırasıyla 2014-15, 2015-16 ve 2016-17 sezonlarında oynanan toplam 6396 maç ele alınarak, bu maçlara dair çok sayıda nitel ve nicel bilgiye erişim sağlanmıştır. Bu bilgilerin dağınık olarak yer aldıkları web platformlarından toplanabilmesi amacıyla çalışma kapsamında çeşitli veri toplama araçları geliştirilmiştir. Geliştirilen bu veri toplama araçları ile erişilen ham veriler, veri setinin oluşturulması sürecinde kullanılmak üzere MS Office Access veri tabanı yönetim sistemi aracılığıyla çeşitli veritabanı tablolarına kaydedilmiştir.

İlk adım olarak, dikkate alınacak olan maçların belirlenmesi amacıyla *"fikstür"* isimli bir veri tabanı tablosu oluşturulmuştur. Bu veri tabanı tablosunda hangi ligden ne kadar maçın yer aldığına dair bilgiler Çizelge 5.1'de gösterilmektedir.

Çizelge 5.1. Liglere ve sezonlara göre ele alınan maç sayıları

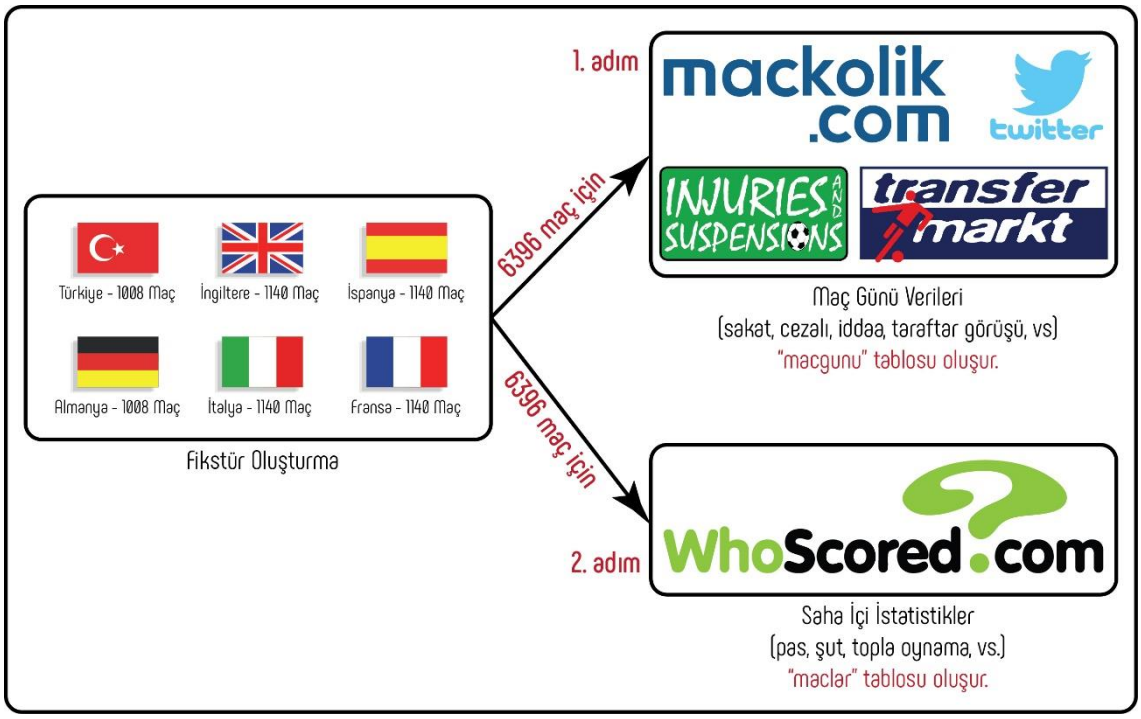
Lig Bilgisi		**Maç Sayısı**		
Ülke	**Lig**	**2014-15**	**2015-16**	**2016-17**
Türkiye	Süper Lig	306	306	306
İngiltere	Premier League	380	380	380
İspanya	La Liga	380	380	380
Almanya	Bundesliga	306	306	306
İtalya	Seria A	380	380	380
Fransa	Ligue 1	380	380	380
Toplam Maç Sayısı		2132	2132	2132

Oluşturulan "*fikstür*" tablosunda bu 6396 maçın tamamı için ayrı birer veri satırı şeklinde, lig bilgisi, sezon bilgisi, oynanma tarihi, takım isimleri ve sonuç bilgileri yer almaktadır.

Fikstür tablosundaki tüm maçlar sıralı bir şekilde ele alınarak veri toplama aşamaları her bir maç için ayrı ayrı gerçekleştirilmiştir. Bu sürecin tamamlanmasıyla birlikte de tüm maçlar için gerekli bilgilere ulaşılmıştır.

5.2. Veri Toplama Süreci

Resim 5.1'de yapısal olarak gösterilen veri toplama süreci fikstür tablosunun oluşturulmasını takiben iki adımda gerçekleşir. Bu adımlardan ilki maç günü verilerinin ikincisi ise saha içi istatistiklerinin elde edilmesidir.



Resim 5.1. Veri toplama süreci

Veri çekimi işlemi tamamlandığında iki yeni veri tabanı tablosu oluşacaktır. Bunlar maç günü verilerinin yer aldığı "maçgünü" tablosu ve saha içi istatistiklerinin yer aldığı "maçlar" tablosudur.

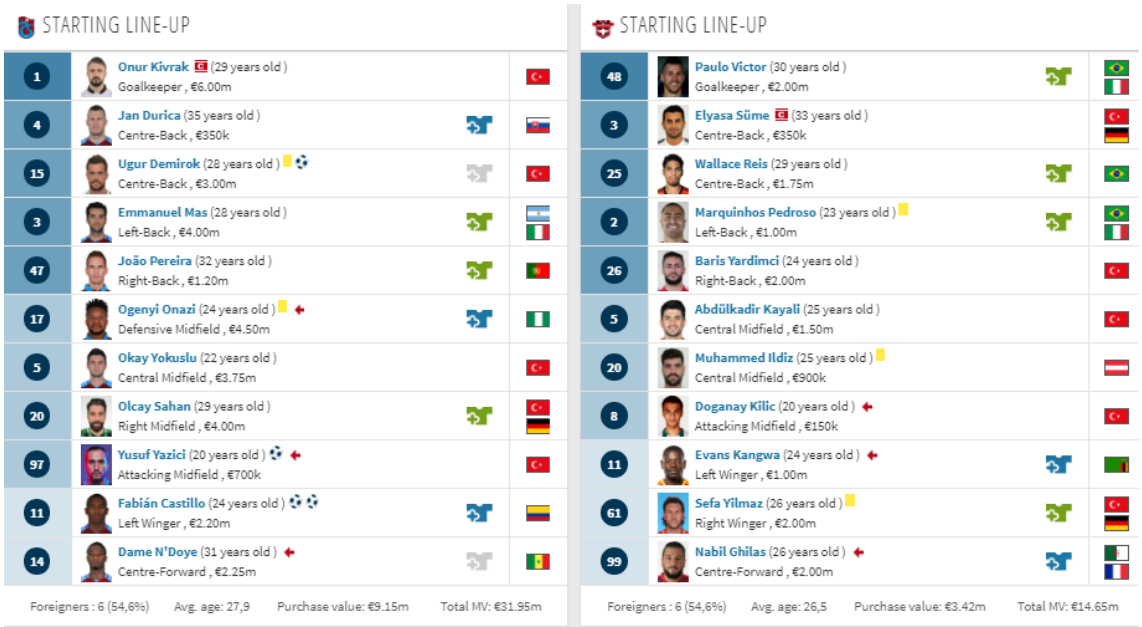
5.2.1. Maç Günü Verilerinin Toplanması

Maç günü verilerinin içerisinde, oynanacak olan müsabaka için belirlenen iddaa oranları, takımların o maça çıktıkları ilk 11 değerleri, sakat ve cezalı oyuncuların durumu, taraftar anketleri, sosyal medya paylaşımları vb. bilgiler yer almaktadır.

Bu bilgilerin bir kısmı, dağınık olarak yer aldıkları “mackolik.com” (2002), “transfermarkt.com” (2000), “injuriesandsuspensions.com” (2013) ve “twitter.com” (2006) gibi web sitelerinden, python, php, javascript, c ve pascal programlama dillerini kullanarak geliştirdiğimiz çeşitli veri toplama araçları ile elde edilirken bir kısmı ise çeşitli gözlemler sonucu el yordamı ile toplanmıştır. Aşağıdaki görsellerde bu bilgilerin yer aldığı web sayfalarının ekran görüntülerine örnekler verilmiştir.



Resim 5.2. İlk 11 kadro bilgisi



Resim 5.3. İlk 11 piyasa değerleri



Maç Sonucu			Handikap				
1	X	2	H	H1	HX	H2	H
1.30	3.70	6.25		2.05	3.40	2.35	+1h

İY 1,5 Gol		AÜ 1,5 Gol		AÜ 2,5 Gol		AÜ 3,5 Gol	
Alt	Üst	Alt	Üst	Alt	Üst	Alt	Üst
1.30	2.30	3.40	1.10	1.80	1.55	1.25	2.50

Karşılıklı Gol		İlk Yarı Sonucu		
Var	Yok	İY 1	İY 0	İY 2
1.70	1.65	1.70	2.30	5.50

Çifte Şans			Toplam Gol			
1-X	1-2	X-2	0-1	2-3	4-6	7+
-	1.08	2.32	3.40	1.75	2.90	18.00

İlk Yarı / Maç Sonucu		
1/1	1/X	1/2
1.80	19.00	45.00
X/1	X/X	X/2
3.75	6.00	15.00
2/1	2/X	2/2
22.00	14.00	13.00

Resim 5.4. İddaa oranları



Anket

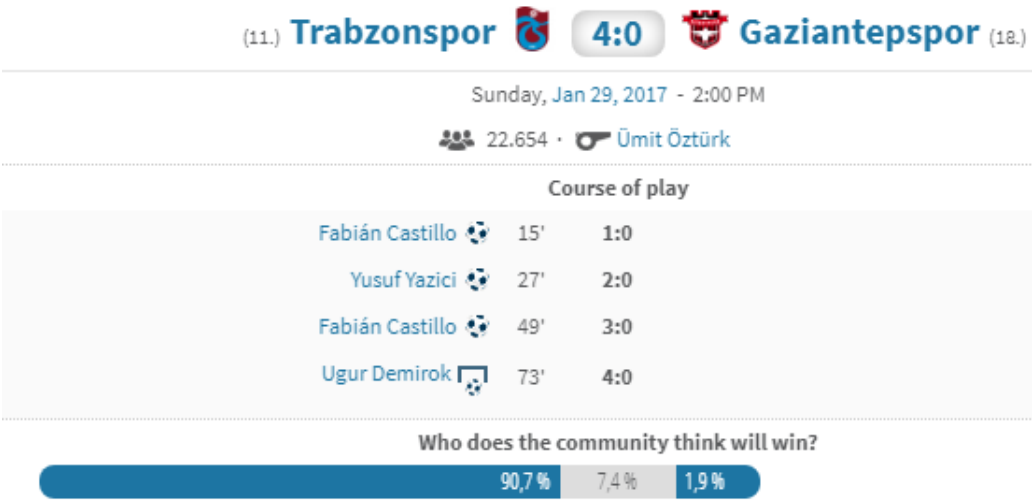
Trabzonspor vs Gaziantepspor maçını kim kazanır? isimli anket tamamlanmıştır. İlginiz için teşekkür ederiz.

Kim kazanır?

sırala	sırala	
Trabzonspor	%65	10175 Oy
Berabere	%22	3448 Oy
Gaziantepspor	%12	1917 Oy

Toplam Oy : 15540

Resim 5.5. Taraftar anket sonuçları-1



(11.) Trabzonspor 4:0 Gaziantepspor (18.)

Sunday, Jan 29, 2017 - 2:00 PM

22.654 · Ümit Öztürk

Course of play

Fabián Castillo	15'	1:0
Yusuf Yazici	27'	2:0
Fabián Castillo	49'	3:0
Ugur Demirok	73'	4:0

Who does the community think will win?

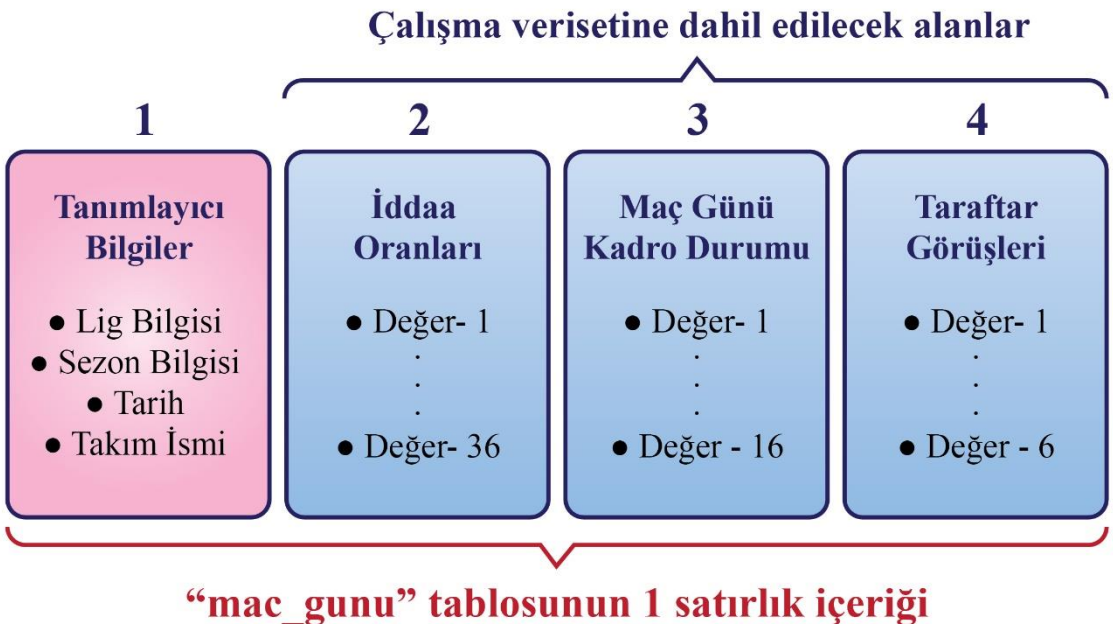
90,7 % 7,4 % 1,9 %

Resim 5.6. Taraftar anket sonuçları-2



Resim 5.7. Sakatlık ve ceza bilgileri

Aşağıda *"iddaa oranları"*, *"maç günü kadro durumu"* ve *"taraftar görüşleri"* şeklinde üç başlık altında incelenen bu bilgiler, *"fikstür"* tablosunda yer alan tüm maçlar için toplanarak Resim 5.8'de gösterildiği gibi veri tabanında yer alan *"maçgünü"* tablosuna kaydedilmektedir.



Resim 5.8. Maç günü verilerinin içeriği

İddaa Oranları

İddaa oranları, Spor Toto Teşkilatı tarafından belirlenen yasal oranlardan oluşmaktadır. Toplamda 36 farklı iddaa seçeneği bulunmaktadır. Bu seçeneklerin adları ve kısa açıklamalar Çizelge 5.2'de yer almaktadır.

Çizelge 5.2. Verilere eklenen bahis seçenekleri

MS 1 İddaa Oranı Maçı ev sahibi takımın kazanacağını varsayan iddaa oranı	***İY X İddaa Oranı*** Maçın ilk yarısının berabere tamamlanacağını varsayan iddaa oranı
MS X İddaa Oranı Maçın berabere tamamlanacağını varsayan iddaa oranı	***İY 2 İddaa Oranı*** Maçın ilk yarısının deplasman takımının önde tamamlayacağını varsayan iddaa oranı
MS 2 İddaa Oranı Maçı deplasman takımının kazanacağını varsayan iddaa oranı	***1-X Çifte Şans İddaa Oranı*** Maçın ev sahibi lehine veya berabere tamamlanacağını varsayan iddaa oranı
Handikaplı Takım Hangi takımın lehine handikap tanımlandığı bilgisi	***1-2 Çifte Şans İddaa Oranı*** Maçın ev sahibi veya deplasman takımı lehine sonuçlanacağını varsayan iddaa oranı
Handikaplı MS 1 İddaa Oranı Handikaplı durumda maçı ev sahibi takımın kazanacağını varsayan iddaa oranı	***X-2 Çifte Şans İddaa Oranı*** Maçın berabere veya deplasman takımı lehine sonuçlanacağını varsayan iddaa oranı
Handikaplı MS X İddaa Oranı Handikaplı durumda maçın berabere tamamlanacağını varsayan iddaa oranı	***0-1 Gol İdaa Oranı*** Maç boyunca en fazla 1 gol atılacağını varsayan iddaa oranı
Handikaplı MS 2 İddaa Oranı Handikaplı durumda maçı deplasman takımının kazanacağını varsayan iddaa oranı	***2-3 Gol İdaa Oranı*** Maç boyunca 2 veya 3 gol atılacağını varsayan iddaa oranı
İlk Yarı 1.5 Altı İddaa Oranı Maçın ilk yarısında 1.5 gol altı (0-1 gol) gol atılacağını varsayan iddaa oranı	***4-6 Gol İdaa Oranı*** Maç boyunca 4,5 veya 6 gol atılacağını varsayan iddaa oranı
İlk Yarı 1.5 Üstü İddaa Oranı Maçın ilk yarısında 1.5 gol üstü (2 ve üzeri) gol atılacağını varsayan iddaa oranı	***+7 Gol İdaa Oranı*** Maç boyunca 7 ve üzeri gol atılacağını varsayan iddaa oranı
1.5 Altı İddaa Oranı Maçın tamamında 1.5 gol altı (0-1 gol) gol atılacağını varsayan iddaa oranı	***İY/MS 1/1 İddaa Oranı*** İlk yarısı ev sahibi lehine tamamlanan maçın sonucunda da ev sahibinin kazanacağını varsayan iddaa oranı
1.5 Üstü İddaa Oranı Maçın tamamında 1.5 gol üstü (2 ve üzeri) gol atılacağını varsayan iddaa oranı	***İY/MS 1/X İddaa Oranı*** İlk yarısı ev sahibi lehine tamamlanan maçın berabere biteceğini varsayan iddaa oranı
2.5 Altı İddaa Oranı Maçın tamamında 2.5 gol altı (0-1-2 gol) gol atılacağını varsayan iddaa oranı	***İY/MS 1/2 İddaa Oranı*** İlk yarısı ev sahibi lehine tamamlanan maçı deplasman takımının kazanacağını varsayan iddaa oranı
2.5 Üstü İddaa Oranı Maçın tamamında 2.5 gol üstü (3 ve üzeri) gol atılacağını varsayan iddaa oranı	***İY/MS X/1 İddaa Oranı*** İlk yarısı berabere tamamlanan maçı ev sahibi takımın kazanacağını varsayan iddaa oranı
3.5 Altı İddaa Oranı Maçın tamamında 3.5 gol altı (0-1-2-3 gol) gol atılacağını varsayan iddaa oranı	***İY/MS X/X İddaa Oranı*** İlk yarısı berabere tamamlanan maçın sonucunun da berabere biteceğini varsayan iddaa oranı
3.5 Üstü İddaa Oranı Maçın tamamında 3.5 gol üstü (4 ve üzeri) gol atılacağını varsayan iddaa oranı	***İY/MS X/2 İddaa Oranı*** İlk yarısı berabere tamamlanan maçı deplasman takımının kazanacağını varsayan iddaa oranı
Karşılıklı Gol Var İddaa Oranı Maç içerisinde iki takımdan en fazla bir tanesinin gol atabileceğini varsayan iddaa oranı	***İY/MS 2/1 İddaa Oranı*** İlk yarısı deplasman takımı lehine tamamlanan maçı ev sahibinin kazanacağını varsayan iddaa oranı
Karşılıklı Gol Var İddaa Oranı Maç içerisinde iki takımında gol atacağını varsayan iddaa oranı	***İY/MS 2/X İddaa Oranı*** İlk yarısı deplasman takımı lehine tamamlanan maçın berabere biteceğini varsayan iddaa oranı
İY 1 İddaa Oranı Maçın ilk yarısını ev sahibi takımın önde tamamlayacağını varsayan iddaa oranı	***İY/MS 2/2 İddaa Oranı*** İlk yarısı deplasman takımı lehine tamamlanan maçı deplasman takımının kazanacağını varsayan iddaa oranı

Maç Günü Kadro Durumları

Maç günü kadro durumları, karşılaşan takımların ilk 11'deki oyuncuları ile sakat ve cezalı oyuncuları hakkında çeşitli istatistikler içermektedir. Toplamda 16 farklı değişkenin olduğu bu bölüm yer alan değişkenler Çizelge 5.3'te açıklanmıştır.

Çizelge 5.3. Maç günü verilerinin içeriği

Ev Sahibi İlk 11 Değeri Ev sahibi takımın maça çıktığı ilk 11'in o günkü güncel piyasa değerleri toplamı	***Ev Sahibi Sakat ve Cezalı Değeri*** Ev sahibi takımın o maçta oynayamayacak sakat ve cezalı oyuncularının o günkü güncel piyasa değerleri toplamı
Deplasman İlk 11 Değeri Deplasman takımının maça çıktığı ilk 11'in o günkü güncel piyasa değerleri toplamı	***Deplasman Sakat ve Cezalı Değeri*** Deplasman takımının o maçta oynayamayacak sakat ve cezalı oyuncularının o günkü güncel piyasa değerleri toplamı
Ev Sahibi İlk 11 Rating Oranı Ev sahibi takımın maça çıktığı ilk 11'in o maç öncesi elde ettikleri performans puanları ortalaması	***Ev Sahibi Sakat ve Cezalı Rating Oranı*** Ev sahibi takımın o maçta oynayamayacak sakat ve cezalı oyuncularının o maç öncesi elde ettikleri performans puanları ortalaması
Deplasman İlk 11 Rating Oranı Deplasman takımının maça çıktığı ilk 11'in o maç öncesi elde ettikleri performans puanları ortalaması	***Deplasman Sakat ve Cezalı Rating Oranı*** Deplasman takımının o maçta oynayamayacak sakat ve cezalı oyuncularının o maç öncesi elde ettikleri performans puanları ortalaması
Ev Sahibi İlk 11 Ortalama Süre Toplamı Ev sahibi takımın maça çıktığı ilk 11'deki oyuncuların her birinin o maça kadar oynadığı maç başı sürelerin toplamı*	***Ev Sahibi Sakat ve Cezalı Ortalama Süre Toplamı*** Ev sahibi takımın o maçta oynayamayacak sakat ve cezalı oyuncularının her birinin o maça kadar oynadığı maç başı sürelerin toplamı*
Deplasman İlk 11 Ortalama Süre Toplamı Deplasman takımının maça çıktığı ilk 11'deki oyuncuların her birinin o maça kadar oynadığı maç başı sürelerin toplamı*	***Deplasman Sakat ve Cezalı Ortalama Süre Toplamı*** Deplasman takımının o maçta oynayamayacak sakat ve cezalı oyuncularının her birinin o maça kadar oynadığı maç başı sürelerin toplamı*
Ev Sahibi İlk 11 Gole Katkı Toplam Ev sahibi takımın maça çıktığı ilk 11'deki oyuncuların her birinin o maça kadar oynadığı maçlardaki gol ve asist sayılarının toplamı	***Ev Sahibi Sakat ve Cezalı Gole Katkı Toplam*** Ev sahibi takımın o maçta oynayamayacak sakat ve cezalı oyuncularının her birinin o maça kadar oynadığı maçlardaki gol ve asist sayılarının toplamı
Deplasman İlk 11 Gole Katkı Toplam Deplasman takımının maça çıktığı ilk 11'deki oyuncuların her birinin o maça kadar oynadığı maçlardaki gol ve asist sayılarının toplamı	***Deplasman Sakat ve Cezalı Gole Katkı Toplam*** Deplasman takımının o maçta oynayamayacak sakat ve cezalı oyuncularının her birinin o maça kadar oynadığı maçlardaki gol ve asist sayılarının toplamı

* oyuncunun maç başı süresi = oynadığı toplam süre / takımın yaptığı maç sayısı

Taraftar Görüşleri

Taraftar görüşleri verisi temel olarak iki bölümden oluşmaktadır. Bunlardan ilki çeşitli web sitelerinde ve taraftar forumlarından maç öncesi gerçekleştirilen ve maçın sonucunun ne olacağına dair tahminlerin yer aldığı anketlerdir. İkincisi ise sosyal medya platformlarında maç öncesi yapılan paylaşımlar üzerinde gerçekleştirilen duygu analizi sonucu elde edilen verilerdir.

Taraftar anketleri, “mackolik.com” ve “whoscored.com” web sitelerinde yer alan maçlara özel sayfalarda maç saatine kadar aktif şekilde durmaktadır. Çok fazla kullanıcısı olan bu sitelerde bir maç için ortalama oy sayısı yirmi binin üzerindedir. İki siteden elde edilen toplam oylar birleştirilerek elde edilen veriler oransal olarak, “ev sahibi”, “beraberlik”, “deplasman” sütünlarına ayrılarak veri tabanına eklenmiştir.

Sosyal medya verileri, çeşitli sosyal medya platformları aracılığıyla maçların başlama saati öncesinde taraflı ve tarafsız kişilerce yapılan paylaşımlardan oluşmaktadır. Tez sürecinde faydalanmak üzere bu platformlar içerisinden, gerek kullanıcı sayısındaki fazlalık gerekse yapılan paylaşımların genel olarak metinsel olmasından dolayı sadece Twitter üzerindeki paylaşımlar dikkate alınmıştır.

Twitter verilerinin elde ediliş mantığı açıklanacak olursa;

- Öncelikle takımları niteleyen takım ismi ve var ise lakapları belirlenir. Örneğin; Trabzonspor, Trabzon, Beşiktaş, Kara Kartal, Galatasaray, Cimbom, Fenerbahçe, Sarı Kanarya, Bursaspor, Yeşil Timsah gibi.
- A ve B takımları arasında gerçekleşecek olan bir maç için “A” ve “B” anahtar kelimelerinin seçildiğini düşünelim.
- Maçın başlangıç saatinden 48 saat öncesinden başlanarak maç saatine kadar “A” ve “B” anahtar kelimelerinin geçtiği tüm twitler çekilir. Twitlerin çekiliş işlemi python kütüphaneleri kullanılarak geliştirilen bir uygulama aracılığıyla yapılmaktadır.
- Twit çekme işlemi sonrasında A ve B takımlarına ait paylaşımların yer aldığı iki ayrı veri dosyası elde edilir ve bu dosyalar üzerinde spam, reklam vb. temizlikler gerçekleştirilir.
- Bir sonraki adımda olabildiğince fazla sayıda twit okunarak el yordamı ile “olumlu” ve “olumuz” diye işaretlenir. Bu işaretleme yapılırken atılan twit eğer anahtar kelimedeki takımın lehine ise olumlu aleyhine ise olumsuz olarak değerlendirilir. Örneğin; “Ankara gücü rahat kazanır” twiti “olumlu” işaretlenirken, “Ankaragücü’nün işi zor” twiti olumsuz olarak işaretlenmektedir.
- Bu şekilde, 3000 olumlu, 3000 olumsuz toplamda 6000 twit işaretlenir. Ardından olumlu ve olumsuz twitlerde geçen sıklığı en yüksek kelimeler belirlenerek twitleri kümelememize yarayacak olan sözlükler elde edilir.
- Bir sonraki adımda A ve B takımları için atılan twitler üzerinde duygu analizi gerçekleştirilir. Bu işlemler k ortalamalar kümeleme yöntemi kullanılarak WEKA yazılımı ile yapılmıştır.

- Eldeki tüm twitler bu şekilde kümelenerek A ve B takımları için olumlu, olumsuz ve nötr twit sayısı bulunur.
- Ardından A takımı için atılan olumlu twit sayısına B takımı için atılan olumsuz twit sayısı, B takımı için atılan olumlu twit sayısına da A takımı için atılan olumsuz twit sayısı eklenir. Son olarak her iki takım için nötr olan twit sayıları da toplanır ve ortaya A takımı, B takımı ve beraberlik (nötr) için üç farklı değer çıkar. Bu üç değer, üç değişken olarak ele alınır ve veri setine eklenir.

Taraftar görüşleri bölümünde taraftar anketlerinden ve sosyal medya verilerinden elde edilen toplam altı adet değişken Çizelge 5.4'te gösterildiği gibidir.

Çizelge 5.4. Taraftar görüşleri verisinin içeriği

Ev Sahibi Oy Oranı	***(Ev Sahibi Pozitif Görüş Oranı + Deplasman Negatif Görüş Oranı)***
Taraftar anketlerinde ev sahibi takımın kazanacağına dair kullanılan oyların toplam oylara olan oranı	Ev sahibi için toplanan twitler içerisindeki pozitif duygu barındıran twitlerin sayısı ile deplasman takımı için toplanan twitlerde negatif duygu barındıran twitlerin sayısının toplamı
Beraberlik Oy Oranı	***(Ev Sahibi Nötr Görüş Oranı + Deplasman Nötr Görüş Oranı)***
Taraftar anketlerinde maçın berabere biteceğine dair kullanılan oyların toplam oylara olan oranı	Toplanan twitler içerisinde her iki takım için de nötr duygu barındıran twitlerin toplam sayısı.
Deplasman Oy Oranı	***(Ev Sahibi Negatif Görüş Oranı + Deplasman Pozitif Görüş Oranı)***
Taraftar anketlerinde deplasman takımının kazanacağına dair kullanılan oyların toplam oylara olan oranı	Ev sahibi için toplanan twitler içerisindeki negatif duygu barındıran twitlerin sayısı ile deplasman takımı için toplanan twitlerde pozitif duygu barındıran twitlerin sayısının toplamı

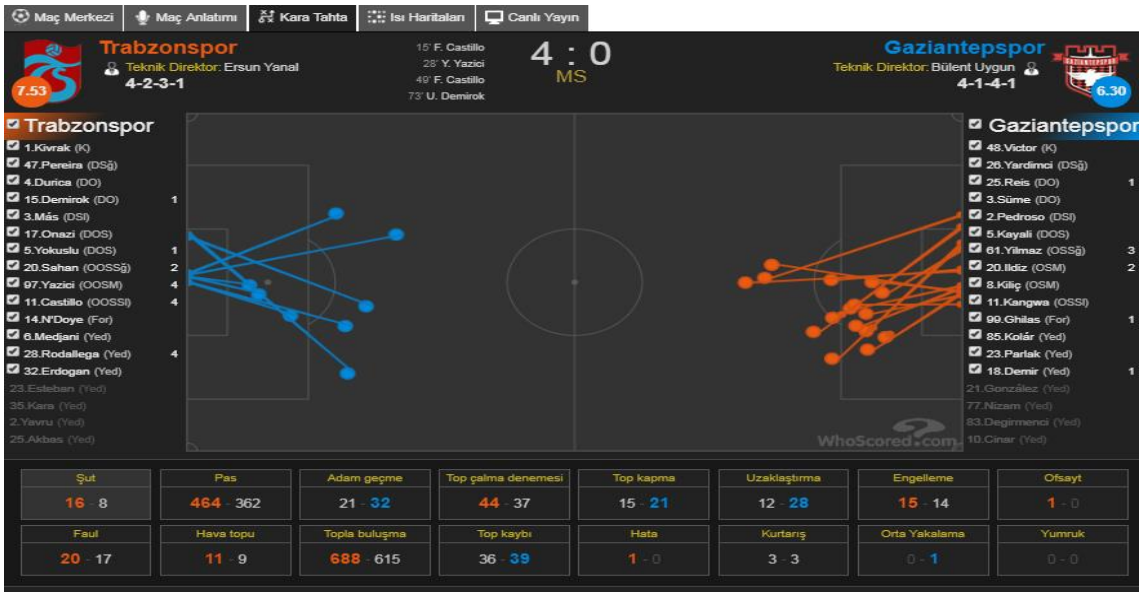
5.2.2. Saha İçi İstatistiklerin Toplanması

Herhangi bir haftada birbirlerine rakip olan iki takımın güçleri mukayese edilirken yararlanılabilecek bilgilerden birisi de bu takımların o maça gelmeden önce oynadıkları belli sayıdaki maçta ulaştıkları saha içi istatistiklerdir. Örneğin, "A takımı son 5 maçında kaç şut çekmiştir?", "B takımı son 7 maçında yüzde kaç topa sahip olmuştur?" şeklindeki sorular bu saha içi istatistiklerin bilinmesiyle cevaplanabilir.

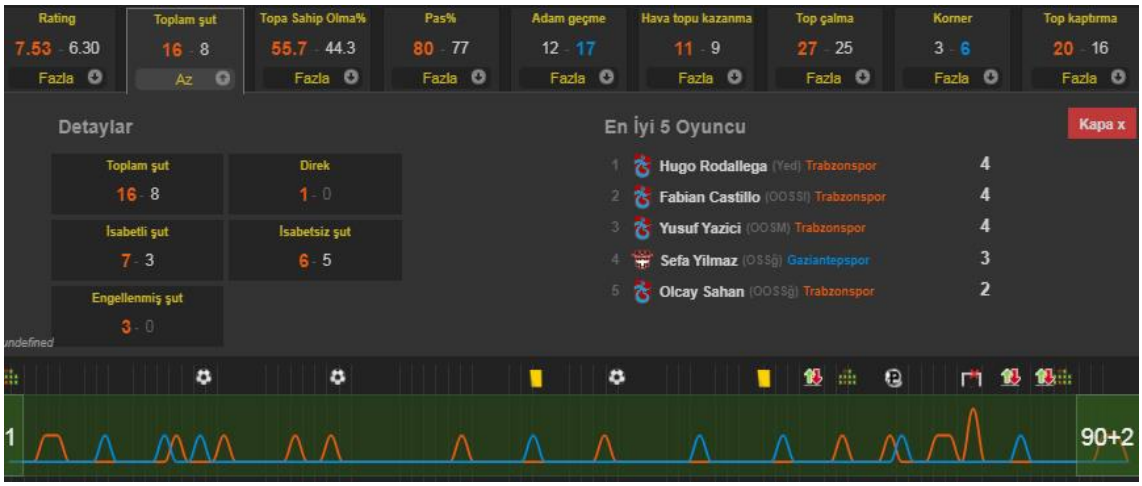
Çalışmada kullanılacak saha içi istatistikler, düzenli olarak dünyanın çeşitli liglerinde oynanan futbol müsabakalarına ait istatistikler paylaşan "whoscored.com" (2008) isimli web sitesinden elde edilmiştir. Bu web sitesinde her maç için ayrı ayrı özelleştirilmiş "maç raporu" sayfaları yer almakta ve bu sayfalarda ise o maçlara ait çok sayıda istatistiksel veri paylaşılmaktadır. Paylaşılan bu verilere ait ekran görüntüleri aşağıdaki görsellerde paylaşılmıştır.



Resim 5.9. Saha içi performans istatistikleri-1



Resim 5.10. Saha içi performans istatistikleri-2



Resim 5.11. Saha içi performans istatistikleri-3



Resim 5.12. Saha içi performans istatistikleri-4

Maç raporu sayfası incelenerek takımların maç içerisinde elde ettikleri bir çok istatistiksel veri içerisinden toplam 65 tanesi seçilmiştir. Seçilen bu istatistiksel veriler Çizelge 5.5, 5.6, ve 5.7'de belirtilmiştir. Tablolarda değişkenlerin veri tabanındaki isimleri ve açıklamaları yer almaktadır. Çizelge 5.5'te genel değişkenler, Çizelge 5.6 'da çeşitli şut istatistikleri, Çizelge 5.7'de ise pas istatistikleri açıklamaları ile yer almaktadır.

Çizelge 5.5. Saha içi genel performans verileri

Alt Kategori	Değişken İsmi	Açıklama
Performans	***Rating****	Takımların o maç için topladıkları performans puanı
	Topa_Sahip_Olma_yuzde	Topla oynama yüzdeleri
	Topla_bulusma	Akan oyunda ve duran toplarda topla buluşma sayısı
Adam Geçme	***Adam_gecme_denemesi***	Rakibini geçme girişimlerinin toplamı
	Basarili_Adam_gecme	Başarılı şekilde rakibi geçişlerin toplamı
	Adam_gecme_yuzde	Rakibi geçme girişimlerindeki başarı yüzdesi
Top Çalma	***Basarili_top_calma***	Rakipten başarılı şekilde top çalma
	Top_calma_denemesi	Top çalma için yapılan hamle sayısı
	Calim_yeme	Rakip oyunculardan çalım yeme sayısı
	Top_calma_yuzde	Başarılı top çalma girişimlerinin yüzdesi
	Uzaklastirma	Rakipteki veya rakibe giden topu uzaklaştırma
	Top_kapma	Sahipsiz topu kazanma sayısı
Hava Topu	***Hava_topu_kazanma***	Toplam kazanılan hava topu mücadelesi sayısı
	Hava_topu_kazanma_yuzde	Hava topu müdahalelerindeki başarı yüzdesi
	Ofansif_hava_topu	Hücumda kazanılan hava topu sayısı
	Defansif_hava_topu	Savunmada kazanılan hava topu sayısı
Korner	***Korner***	Toplan kullanılan korner sayısı
	Korner_isabet_yuzde	Kullanılan kornerlerin takım arkadaşına ulaşma yüzdesi
Kaleci	***Kurtaris***	Toplam kurtarış sayısı
	Orta_Yakalama	Gelen ortayı havada yakalama sayısı
	Yumruk	Kaleye gelen topu yumrukla uzaklaştırma sayısı
Diğer	***Ofsayt***	Toplam ofsayda düşme sayısı
	Faul	Toplam yapılan faul sayısı
	Top_kaybi	Her türlü sebeple kaybedilen toplasın sayısı
	Hata	Gole, penaltıya veya karta sebep ıolan hayati hata sayısı
	Engelleme	Gol, şut, orta engelleme sayısı

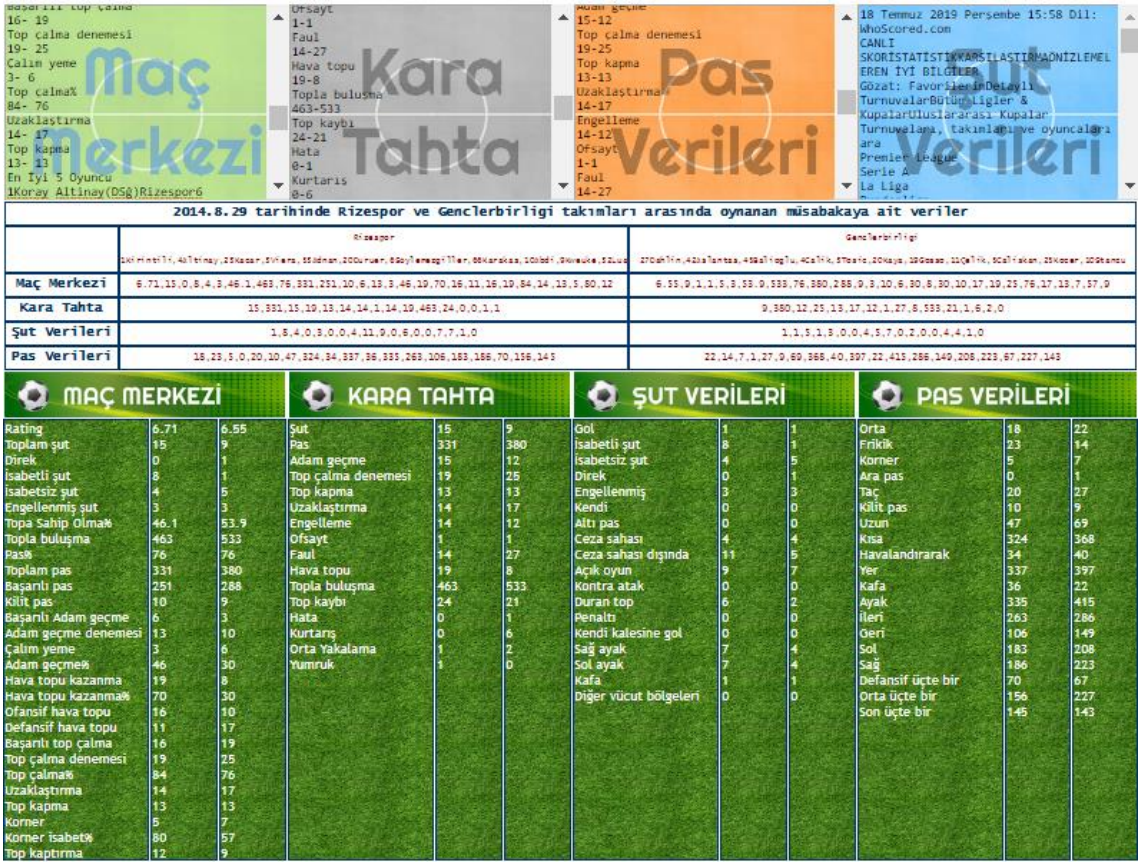
Çizelge 5.6. Saha içi şut performansı verileri

Alt Kategori	Değişken İsmi	Açıklama
Sayılar	***Toplam_sut***	Toplam şut sayısı
Sonuç	***Gol***	Golle sonuçlanan şut sayısı
	Direk	Direkten dönen şut sayısı
	Isabetli_sut	İsabetli şut sayısı
	Isabetsiz_sut	İsabetsiz şut sayısı
	Engellenmis_sut	Engellenmiş şut sayısı
	Kendi_kalesine_gol	Kendi kalesine atılan gol sayısı
Bölge	***Alti_pas***	Altı pastan çekilen şut sayısı
	Ceza_sahasi	Ceza sahası içinden çekilen şut sayısı
	Ceza_sahasi_disinda	Ceza sahası dışından çekilen şut sayısı
Durum	***Acik_oyun***	Akan oyunda çekilen şut sayısı
	Kontra_atak	Kontra atak sonucu çekilen şut sayısı
	Duran_top	Duran toptan çekilen şut sayısı
	Penalti	Kullanılan penaltı sayısı
Vücut Bölgesi	***Sag_ayak_şut***	Sağ ayakla çekilen şut sayısı
	Sol_ayak_şut	Sol ayakla çekilen şut sayısı
	Kafa_şut	Kafa vuruşuyla yapılan şut sayısı
	Diger_vucut_bolgeleri	Diğer vücut uzuvları ile çekilen şut sayısı

Çizelge 5.7. Saha içi pas performansı verileri

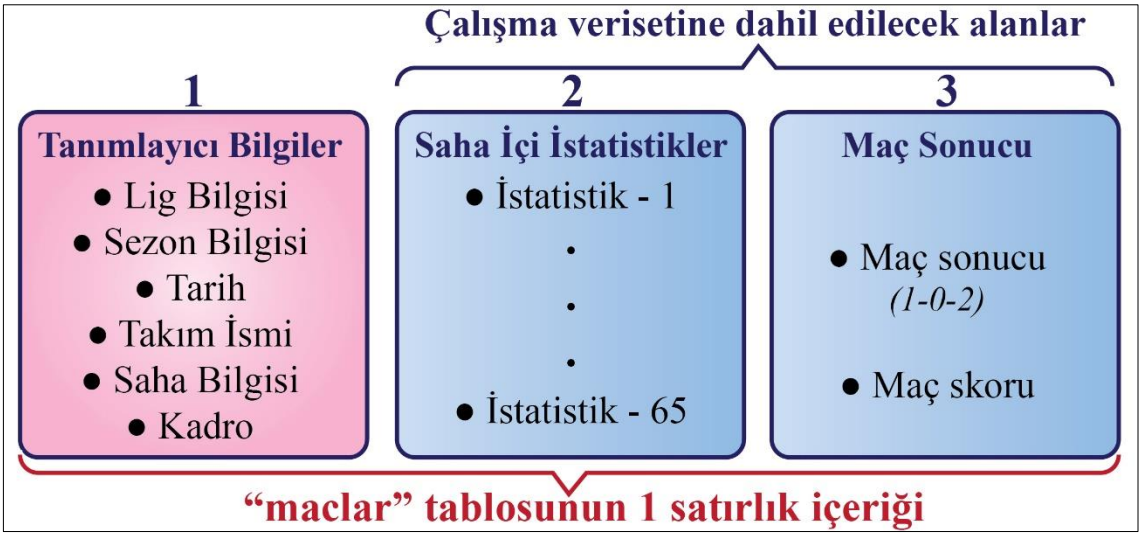
Alt Kategori	Değişken İsmi	Açıklama
Sayılar	***Pas_yuzde***	Başarılı pas yüzdesi
	Toplam_pas	Toplam pas girişimi sayısı
	Basarili_pas	Başarılı pas sayısı
Tip	***Kilit_pas***	Kilit pas sayısı
	Orta	Orta Sayısı
	Frikik	Frikikten atılan pas sayısı
	Ara_pas	Ara pas sayısı
	Tac	Taçtan atılan pas sayısı
Uzunluk	***Uzun_pas***	Uzun pas sayısı
	Kisa_pas	Kısa pas sayısı
Yükseklik	***Havalandirarak_pas***	Topu havalandırarak atılan pas sayısı
	Yerden_pas	Yerden kullanılan pas sayısı
Vücut Bölgesi	***Kafa_pas***	Kafa ile gönderilen pas sayısı
	Ayak_pas	Ayak ile gönderilen pas sayısı
Yön	***Ileri_pas***	İleri doğru atılan pas sayısı
	Geri_pas	Geri doğru atılan pas sayısı
	Sol_pas	Sol tarafa atılan pas sayısı
	Sag_pas	Sağ tarafa atılan pas sayısı
Bölge	***Defansif_ucte_bir_pas***	1. bölgede gerçekleştirilen pas sayısı
	Orta_ucte_bir_pas	2. bölgede gerçekleştirilen pas sayısı
	Son_ucte_bir_pas	3. Bölgede gerçekleştirilen pas sayısı

Daha sonra seçilen bu değerlerin elde edilebilmesi için Resim 5.13'te arayüzü gösterilen veri çekme uygulaması kullanılmıştır. Bu uygulama, belli anahtar kelimelere göre maç raporu sayfalarında çeşitli arama algoritmalarını çalıştırarak gerekli değerleri bulur.



Resim 5.13. Saha içi performans verisi çekme uygulaması

Bulunan bu değerler her iki takım için ayrı ayrı satırlarda yer alacak şekilde daha önceden oluşturulan "maçlar" isimli veri tabanı tablosuna 2 satır halinde eklenir. Bu satırın içeriği tek bir takım için Resim 5.14'de gösterilmektedir. Bu işlem "fikstür" tablosunda yer alan tüm maçlar için tekrarlanarak her maçın saha içi istatistikleri "maçlar" tablosuna kaydedilir.



Resim 5.14. Maçlar tablosunun içeriği

5.2.3. Eksik ve Hatalı Verilerin Tamamlanması

Veri toplama sürecinde elde edilen veriler üzerinde eksik, hatalı ve aykırı olduğu gözlenen veriler ile ilgili işlemler bu aşamada gerçekleştirilmiştir. Veri kaynaklarında bazı müsabakalar için veri bulunmamaktadır. Bu nedenle de eksik veri durumları ortaya çıkmıştır. Çalışmada eksik veri bulunan müsabakalar göz ardı edilerek veri seti oluşturma sürecine dahil edilmemişlerdir. Verilerinde bir aykırılık veya hata bulunan müsabakalar da çalışmanın sağlıklı olması açısından göz ardı edilmişlerdir.

Verilerdeki eksik, hatalı ve aykırı değerler için gerekli işlemlerin yapılmasının ardından bu adımda gerçekleştirilecek bir diğer işlem ise veri normalleştirmedir. Fakat çalışma gereği normalleştirme işlemi *"Veri Hesaplamaları ve Veri Setinin Hazırlanması"* başlığında belirtilen hesaplama işlemlerinin ardından gerçekleştirilmiştir.

5.3. Veri Hesaplamaları ve Veri Setinin Hazırlanması

Veri toplama sürecinin ardından veri tabanımızda önceden var olan "fikstür" tablosuna ek olarak iki yeni veri tabanı tablosu daha oluşmuştur. Bunlar her takımın *"fikstür"* tablosunda yer alan maçlarda elde ettiği saha içi istatistikleri içeren *"maçlar"* tablosu ve bu maçların maç günü bilgilerini veren *"maçgünü"* tablosudur. Veri hesaplama adımının amacı bu veri tabanı tablolarındaki verileri kullanarak çalışmada kullanılacak bir veri seti oluşturmaktır.

Cevaplamak istediğimiz soru herhangi bir zamanda karşı karşıya gelecek olan A ve B takımları arasındaki maçın sonucunun ne olacağıdır. Bu amaçla daha önceden oynanmış maçlar birer örnek durumu olarak ele alınarak mümkün olduğunca fazla sayıda

örnek veri setine eklenecektir. Çok sayıda örneğin yer aldığı nitelikli bir veri seti geliştirilecek olan modelin kestirim başarısının yüksek olmasında son derece etkilidir.

Geliştirilen bir uygulama aracılığı ile *"fisktür"* tablosundaki her maç için *"maçlar"* ve *"maçgünü"* tablolarında yer alan veriler üzerinde çeşitli hesaplamalar gerçekleştirilerek bir veri seti oluşturulmuştur.

5.3.1. Veri Hesaplama Uygulaması

Veri hesaplama uygulamasının temelde iki işlevi vardır. İlk işlevi veri tabanı tabloları üzerinde bir dizi hesaplamalar gerçekleştirmek ikincisi ise hesaplamalar sonucu elde edilen değerler ile bir veri seti oluşturmaktır. Resim 5.15'te ara yüzü verilmiş olan bu uygulama, her iki işlevini de gerçekleştirirken kullanıcı tarafından belirlenen parametreleri baz alır (Kınalıoğlu, 2019).



Resim 5.15. Veri seti hazırlama uygulaması

Bu parametreler hesaplama ve veri seti oluşturma aşamalarında geçerli olacak kullanıcı tercihleridir. İlerleyen başlıklarda detaylı şekilde anlatılacak olan bu parametreleri, uygulama ara yüzündeki konumlarına göre beş başlıkta kısaca açıklamak gerekirse;

Hesaplama aşaması için,

1. Önceki Maçların Aralığı: Karşı karşıya gelen takımların, saha içi istatistiklerinin incelenmesi amacıyla son oynadıkları kaç maçın inceleneceği bilgisinin girildiği alandır.

2. Takım Verisi Karşılaştırma: Karşı karşıya gelen takımların son maçlarda elde ettikleri ortalama saha içi istatistik değerlerinin nasıl kıyaslanacağı bilgisinin belirtildiği alandır.

Veri setinin hazırlanması aşaması için

3. Veri Seti İçeriği: Oluşturulacak olan veri setinde hangi alt başlıklardaki özniteliklerin yer alacağının belirlendiği alandır.

4. Eğitim Dosyası Çıktı Seçenekleri: Oluşturulan veri setinin eğitim ve test dosyalarına nasıl ayrıştırılacağına dair bilgilerin girildiği alandır.

5. Kayıt Seçenekleri: Oluşturulan veri setinin kayıt formatının ve kaydedilecek dosyanın yapısının belirlendiği alandır.

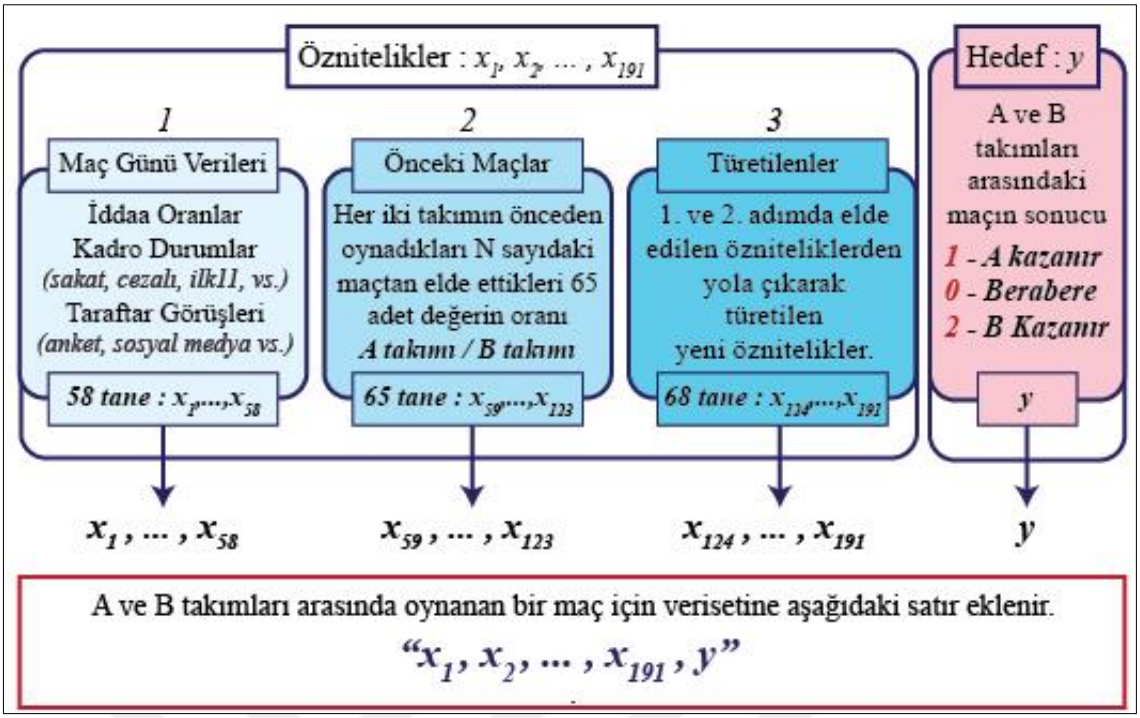
Hesaplama aşamasında iki tür hesaplama işlemi yapılır. Bunlardan ilki *"Takımların Saha İçi Performanslarının Hesaplanması"* başlığında detaylıca açıklanacak olan takım performanslarının karşılaştırılmasıdır. Bu kısımda kullanıcı tarafından belirlenen sayıdaki geçmiş maçın verisine ulaşılır ardından da takımların bu maçlarda elde ettikleri ortalama performans değerleri kıyaslanır.

Hesaplama aşamasının ikinci basamağı ise *"Yardımcı Değişkenlerin Türetilmesi"* başlığında detaylıca anlatılacak olan yeni değişken türetme işlemi yapılır. Bu işlem karşı karşıya gelen iki takım için de geçmişte oynadıkları maçlarda elde ettikleri istatistiki veriler baz alarak gerçekleştirilir. Burada türetilen değişkenler bahis uzmanlarından alınan önerilerle belirlenmiştir.

Veri seti oluşturma aşamasında ise kullanıcı tarafından özelleştirilebilir parametrelere göre veri seti oluşturulur. Oluşturulan veri seti uygulamalarda kullanılmak üzere *.xlsx ve *.csv formatlarında kaydedilebilir.

5.3.2. Veri Setinin Yapısı

Resim 5.16'da yapısı gösterilen veri seti üç bölümden oluşmaktadır. Bunlar; maç günü verileri, önceki maçlardan elde edilen saha içi performans değerleri ve türetilen yeni değişkenlerdir.



Resim 5.16. Çalışma veri setinin yapısı

"Maç günü verileri" bölümünde 58 öznitelik bulunmaktadır ve bunların tamamı internetten elde edilmiştir. Bu özniteliklerin elde edilişi hakkında detaylı bilgi "Maç Günü Verilerinin Toplanması" başlığı altında verilmektedir.

"Önceki Maçlar" bölümünde 65 öznitelik veri vardır. Bu öznitelikler A ve B takımlarının o maç öncesinde oynadıkları belli sayıdaki maçtan elde ettikleri ortalama saha içi istatistiklerin kıyaslanması ile oluşturulur. "Saha İçi İstatistiklerin Toplanması" başlığı altında nasıl elde edildiği açıklanan bu istatistiklerin kıyaslanması ve 65 özniteliğe dönüştürülmesi süreci de "Takım Performanslarının Hesaplanması" başlığı altında açıklanmıştır.

"Türetilenler" bölümünde ise toplam 68 öznitelik vardır. Bu öznitelikler "Maç Günü Verileri" ve "Saha İçi İstatistikleri" kullanılarak türetilmiştir. Bu işlemin detayları ise "Yardımcı Değişkenlerin Türetilmesi" bölümünde açıklanmıştır.

Bu üç bölümde elde edilen özniteliklerin bir araya getirilmesiyle toplam da 191 adet özniteliğin yer aldığı veri seti elde edilir. Bu veri setindeki bağımlı değişkeni ise A ve B takımları arasında oynanan maçın sonucudur. Maçın sonucu 1,0 ve 2 şeklindeki 3 sınıftan oluşmaktadır. 1, A takımının galibiyetini, 0, beraberliği, 2 ise B takımının galibiyetini temsil etmektedir.

5.3.3. Takımların Saha İçi Performanslarının Hesaplanması

Takımların saha içi performanslarının hesaplanması adımı *"fikstür"* tablosundaki tüm maçlar için sıra ile gerçekleştirilir. Karşı karşıya gelen iki takımın saha içi performansları, o maç öncesinde başka takımlar ile oynadıkları belli sayıdaki maçta elde ettikleri istatistiklerin ortalamasın gösterir. Bu değerlerin hesaplanması adımında ilk karar verilmesi gereken nokta her bir takımın önceki haftalarda oynadığı maçların hangilerinin dikkate alınacağıdır.

Kullanıcı, uygulamanın Resim 5.17'de gösterilen *"Önceki Maçların Aralığı"* başlıklı bölümden bu karara ilişkin tercihlerini belirtir.



Resim 5.17. Resim Önceki Maçların Veri aralığı

Bu bölümde 3 seçenek vardır. Bunlar *"En az son M maç", "Sadece son M maç"* ve *"Mevcut Sezondaki Tüm Maçlar"* seçenekleridir. Bu seçeneklerin ne anlama geldiğini ve seçildiklerinde hesaplama adımlarının nasıl gerçekleşeceğini birer örnek ile açıklamak gerekirse:

Fikstür tablosundaki ilk maç *A* ve *B* takımları arasında olsun.

"En az son M maç" seçeneği seçilip, ardından yan tarafta yer alan kutucuğa da ***"M"*** değeri girildiğinde sistem şunu yapmaktadır;

Adım 1: İlk olarak "maçlar" tablosu önce *A* takımına, sonrasında da *B* takımına göre filtrelenir. Böylelikle her iki takımında o sezon oynadığı maçların sayısı ve bilgilerine erişilir.

Adım 2: Ardından kullanıcının belirlediği *"M"* sayısına göre bir kontrol gerçekleştirir, eğer *A* ve *B* takımlarından herhangi birisi, oynayacakları maç öncesinde *"M"* sayısından daha az maç yapmış ise *A* ve *B* takımları arasında oynanan bu maç pas geçilir ve hiçbir hesaplama yapılmaz. Dolayısı ile bu maça ait bilgiler veri setine eklenmez. Eğer *A* ve *B* takımları *"M"* sayısına eşit ya da daha fazla sayıda maç yapmış

ise sistem *A* ve *B* takımlarının önceki *"M"* sayıdaki maçta elde ettikleri saha içi istatistiklerinin ortalamasını alır ve (5.1)'de gösterildiği gibi iki takım için de 1x65 boyutunda birer değer vektörü oluşturur.

$$\mathbf{X}_A = \frac{1}{M_A}\left(\sum_{j=1}^{M_A} X_A^{1j}, \sum_{j=1}^{M_A} X_A^{2j}, \ldots, \sum_{j=1}^{M_A} X_A^{65j}\right)$$

$$\mathbf{X}_B = \frac{1}{M_B}\left(\sum_{j=1}^{M_B} X_B^{1j}, \sum_{j=1}^{M_B} X_B^{2j}, \ldots, \sum_{j=1}^{M_B} X_B^{65j}\right) \quad (5.1)$$

Burada;

A, B : takım adını,

i : değişken numarasını (1,2,…,65),

j : hafta numarasını,

X_A^{ij} : A takımının j. haftadaki i. değişkeni,

X_B^{ij} : B takımının j. haftadaki i. değişkeni,

M : Kullanıcının belirlediği sınırı,

M_A : *A* takımı maç sayısını $|M_A \geq M$,

M_B : *B* takımı maç sayısını $|M_A \geq M$,

göstermektedir.

Herhangi bir ligi göz önüne alacak olursak ligin *(M+1)*'inci haftasına kadar hiçbir takım *"M"* maç yapmamış olacağından *(M+1)*'inci haftaya kadar oynana hiçbir maç için hesaplama yapılmaz ve bu maçların bilgileri veri setine eklenmez. Bu nedenle "*M"* değeri ne kadar düşük olursa veri setine o kadar çok sayıda maç eklenecektir.

Örneğin kullanıcı *"M"* değerini 10 maç olarak belirlerse *A* ve *B* takımları arasındaki bir maçta takımların saha içi performans değerlerinin hesaplanması için her iki takımında en azından o maç öncesinde 10'ar maç yapmış olması gerekecektir. Böylelikle veri setine ancak 10. Haftadan sonra oynanan maçların bilgileri eklenecektir.

Bu sınır aslında daha basit bir ifade ile kaçıncı haftadan sonra oynanan maçları eğitim kümesine eklemek istediğimizin karşılığıdır. Bu seçeneğin koyulma sebebi, özellikle sezonun ilk haftalarında takımların elde ettikleri saha içi istatistiklerinin takımların gerçek güçlerini çok fazla yansıtmıyor olmasıdır.

"Sadece son M maç" seçeneğinde ise *A* ve *B* takımları karşılaştıkları haftanın öncesinde kaç maç oynamış olursa olsunlar, sistem bu takımlar hakkındaki bilgileri sadece son *"M"* sayıdaki maçlarından alarak değerler vektörlerini (5.2.)'deki gibi

oluşturacaktır. Örneğin; bu *"M"* sınırının 5 olduğunu düşünelim, sistem 8. Haftada oynanan bir maç için A takımının ortalama şut sayısını hesaplarken sadece 7,6,5,4, ve 3. haftalarda elde ettiği şut sayılarının ortalamasını alacaktır.

$$\mathbf{X}_A = \frac{1}{M}\left(\sum_{j=(M_A-M+1)}^{M_A} X_A^{1j}, \sum_{j=(M_A-M+1)}^{M_A} X_A^{2j}, \ldots, \sum_{j=(M_A-M+1)}^{M_A} X_A^{65j} \right)$$

$$\mathbf{X}_B = \frac{1}{M}\left(\sum_{j=(M_B-M+1)}^{M_B} X_B^{1j}, \sum_{j=(M_B-M+1)}^{M_B} X_B^{2j}, \ldots, \sum_{j=(M_B-M+1)}^{M_B} X_B^{65j} \right) \quad (5.2)$$

Burada;

A, B : takım adını,

i : değişken numarasını (1,2,…,65),

j : hafta numarasını,

X_A^{ij} : A takımının j. haftadaki i. değişkeni,

X_B^{ij} : B takımının j. haftadaki i. değişkeni,

M_A : *A* takımı maç sayısını,

M_B : *B* takımı maç sayısını,

M : Kullanıcının belirlediği sınırı,

göstermektedir.

Bu seçenekte de *"M"* sınırı kaçıncı haftadan sonra oynanan maçlar için hesaplamalar gerçekleştirileceğini ve hangi maçların bilgilerinin veri setine ekleneceğini belirlemektedir. Diğer seçenekten farkı ise takımların saha içi performansları hesaplanırken oynadıkları maç sayısı *M'den* fazla olsa bile bu maçlar içerisinden sadece son *"M"* sayıdaki maçın dikkate alınmasıdır.

"Mevcut sezondaki tüm maçlar" seçeneği bu bölümdeki son seçenektir ve diğer iki seçeneğe oranla daha az kullanılır. Bu seçenek seçildiğinde *A* ve *B* takımlarının oynayacakları maç öncesinde sadece birer maç yapmış olmaları, (5.3)'teki hesaplamanın gerçekleşebilmesi için yeterli olur. Böylelikle 2. haftadan itibaren oynanan tüm maçların bilgisi veri setine eklenecektir. Takımların saha içi performans değerleri o hafta öncesi oynadıkları tüm maçlardan elde ettikleri değerlerin ortalaması ile elde edilir. Örneği 3. haftada oynanan bir maç için 1 ve 2. haftalarda oynanan maçların verisi kullanılırken, 15. haftada oynanan bir maç için ise 1,2,…,14. hafta aralığında oynana tüm maçların verisi kullanılır.

$$\mathbf{X}_A = \frac{1}{M_A}\left(\sum_{j=1}^{M_A} X_A^{1j}, \sum_{j=1}^{M_A} X_A^{2j}, \ldots, \sum_{j=1}^{M_A} X_A^{65j}\right)$$

$$\mathbf{X}_B = \frac{1}{M_B}\left(\sum_{j=1}^{M_B} X_B^{1j}, \sum_{j=1}^{M_B} X_B^{2j}, \ldots, \sum_{j=1}^{M_B} X_B^{65j}\right) \quad (5.3)$$

Burada;

A, B : takım adını,

i : değişken numarasını (1,2,…,65),

j : hafta numarasını,

X_A^{ij} : A takımının j. haftadaki i. değişkeni,

X_B^{ij} : B takımının j. haftadaki i. değişkeni,

M_A : *A* takımı maç sayısını,

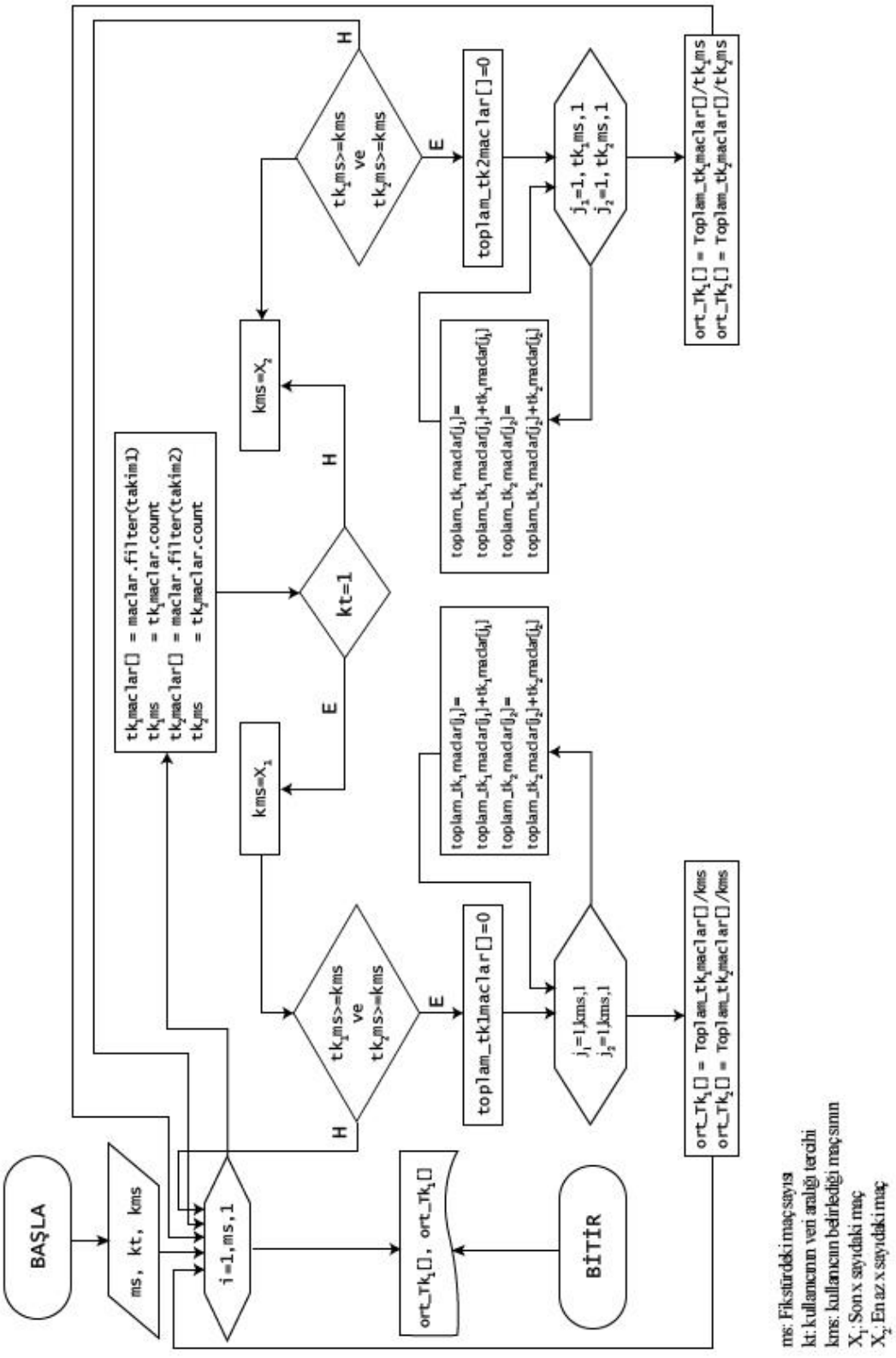
M_B : *B* takımı maç sayısını,

göstermektedir.

Yine bu bölümde bu üç seçenekle de kullanılabilecek *"**sezonun ikinci yarısında sayacı sıfırla**"* isimli ortak bir seçenek vardır. Bu seçeneğin amacını bir örnekle açıklayalım.

Örneğin; kullanıcı, tercihini *"Sadece son 5 maç"* olarak belirledi. Bu durumda 6. haftadan itibaren oynanan maçlar eğitim kümesine eklenmeye başlayacak. Sezonun ilk yarısının 17 hafta olduğu Türkiye Süper Ligi ile örneğimize devam edersek, 18. Hafta geldiğinde her takım 17 maç yapmış olacağından *"sezonun ikinci yarısında sayacı sıfırla"* seçeneği işaretli değilse, uygulama 18. haftadaki bir maçta takımların ortalama saha içi performans değerlerini hesaplayabilmek için geriye doğru, 17,16,15,14 ve 13. haftalara bakacaktır. Öte yandan takımların güçlerinde devre arasında genelde ciddi anlamda değişimler yaşanabilmektedir. Bu nedenle kullanıcı dilerse *"sezonun ikinci yarısında sayacı sıfırla"* seçeneğini işaretleyerek, sistemin 18. haftada oynanan bir maçı 1. hafta gibi değerlendirmesini sağlayabilir. Bu durumda *"Sadece son 5 maç"* seçeneği seçildiğinde, 18,19,20,21 ve 22. Haftalarda oynanan maçlar eğitim kümesine eklenmeyecektir. Çünkü takımlar 23. haftaya kadar 5 maç yapmış olmayacaklardır.

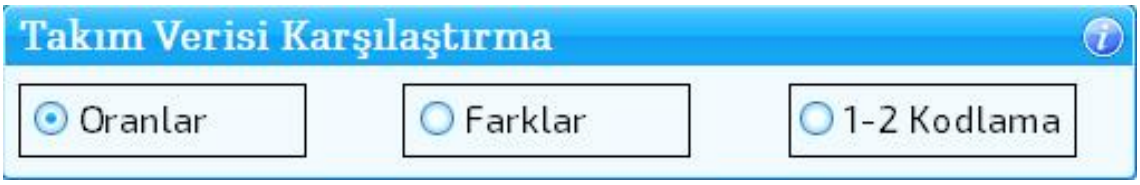
Takımların geçmiş maçlarından elde ettikleri saha içi performansları ifade eden $\mathbf{X}_A$ ve $\mathbf{X}_B$ vektörlerinin elde ediliş süreci Şekil 5.1'de gösterilen akış diyagramında anlatılmaktadır.



Şekil 5.1. Takımların geçmiş performanslarını hesaplanma süreci

Bu hesaplamalar sonucu her iki takım için de geçmiş maçlardan elde edilen ortalama saha içi performanslarını gösteren vektörlerin kıyaslanarak veri setine eklenebilmesi için Resim 5.18'de gösterilen *"Takım Verisi Karşılaştırma"* bölümü kullanılır. Bu bölümde "farklar", "oranlar" ve "1-2 kodlama" seçenekleri bulunmaktadır.

Aşağıda çalışma mantığı açıklanan bu seçeneklerle, A ve B takımlarının geçmiş maçlardan elde ettikleri ortalama saha içi performanslarının tek bir vektörde nasıl birleştirileceği belirlenir.



Resim 5.18. Takım verisi karşılaştırma

A, B takımlarının geçmiş maçlardan elde ettikleri ortalama saha içi performanslarını gösteren vektörler (5.4)'te verilmiştir.

$$\mathbf{X}_A = \left[\bar{X}_A^1, \bar{X}_A^1, \ldots, \bar{X}_A^{65}\right]_{1\times 65}$$
$$\mathbf{X}_B = \left[\bar{X}_B^1, \bar{X}_B^1, \ldots, \bar{X}_B^{65}\right]_{1\times 65} \tag{5.4}$$

"Farklar" seçeneği seçildiğinde $\mathbf{X}_{AB}$, (5.5)'te gösterildiği gibi A takımının verisinden B takımının verisi çıkarılarak hesaplanır.

$$\mathbf{X}_{AB} = \left(\mathbf{X}_A - \mathbf{X}_B\right) = \left[\left(\bar{X}_A^1 - \bar{X}_B^1\right), \left(\bar{X}_A^2 - \bar{X}_B^2\right), \ldots, \left(\bar{X}_A^{65} - \bar{X}_B^{65}\right)\right]_{1\times 65} \tag{5.5}$$

"Oranlar" seçeneği seçildiğinde A takımına ait vektör elemanları karşılıklı olarak B takımına ait vektör elemanlarına bölünerek (5.6)'de gösterildiği gibi $\mathbf{X}_{AB}$ hesaplanır.

$$\mathbf{X}_{AB} = \frac{\mathbf{X}_A}{\mathbf{X}_B} = \left[\frac{\bar{X}_A^1}{\bar{X}_B^1}, \frac{\bar{X}_A^2}{\bar{X}_B^2}, \ldots, \frac{\bar{X}_A^{65}}{\bar{X}_B^{65}}\right]_{1\times 65} \tag{5.6}$$

Oranlar seçeneği seçildiğinde hesaplama esnasında takımlara ait 65 öznitelik karşılıklı olarak oranlanır. Bu oranlama yapılırken öncelikli olarak *A* ve *B* takımlarının oranlanan öznitelikte elde ettikleri değerlerin 0'dan farklı olup olmadığı kontrol edilir. Eğer iki takımdan en azından bir tanesinin o öznitelik için sahip olduğu değer 0'a eşitse ilgili öznitelikteki karşılaştırma işlemi "oranlar" seçeneği yerine "farklar" seçeneği ile gerçekleştirilir.

"1-2 Kodlama seçeneği" seçildiğinde ise (5.7)'de gösterildiği gibi A takımının verisi B takımının verisinden büyükse 1, eşitse 0, küçükse 2 şeklinde kodlanarak 1x65'lik vektör oluşturulur.

$$\mathbf{X}_{AB} = \begin{cases} 1 & , \quad \bar{X}_A^i > \bar{X}_B^i \\ 0 & , \quad \bar{X}_A^i = \bar{X}_B^i \\ 2 & , \quad \bar{X}_A^i < \bar{X}_B^i \end{cases}, \; i = 1,2,\ldots,65 \tag{5.7}$$

5.3.4. Yeni Değişkenlerin Türetilmesi

Bu adımda ise takımların geçmiş maçlarından yola çıkarak oynayacakları maçın sonucuna etki edebileceği düşünülen yeni değişkenler türetilmiştir. Değişkenlerin belirlenmesinde bahis uzmanlarının görüşlerine başvurulmuştur. Türetilen bu değişkenlerin bir kısmı, takımların birbirinden bağımsız olarak düşünülmesiyle elde edilirken bir kısmı takımların birbirleri ile kıyaslanması sonucu gerçekleştirilen hesaplamalar ile elde edilmiştir. Elde edilen tüm değişkenler açıklamaları ile birlikte Çizelge 5.8'de gösterilmektedir.

Çizelge 5.8. Türetilen değişkenler

1	***Ev Sahibi Toplam Galibiyet*** Ev sahibi takımın o sezonki toplam galibiyet sayısı	***35***	***Deplasman Deplasmanda Galibiyet Sayısı*** Deplasman takımının kendi sahasında galip geldiği maçların sayısı
2	***Ev Sahibi Toplam Beraberlik*** Ev sahibi takımın o sezonki toplam beraberlik sayısı	***36***	***Deplasman Deplasmanda Galibiyet Yüzde*** Deplasman takımının kendi sahasında galip gelme yüzdesi
3	***Ev Sahibi Toplam Mağlubiyet*** Ev sahibi takımın o sezonki toplam mağlubiyet sayısı	***37***	***Deplasman Deplasmanda Beraberlik Sayısı*** Deplasman takımının kendi sahasında berabere kaldığı maçların sayısı
4	***Ev Sahibi Son X Maç Galibiyet*** Ev sahibi takımın kullanıcının belirlediği sınır olan son X maçtaki galibiyet sayısı	***38***	***Deplasman Deplasmanda Beraberlik Yüzde*** Deplasman takımının kendi sahasında berabere kalmaz yüzdesi
5	***Ev Sahibi Son X Maç Beraberlik*** Ev sahibi takımın kullanıcının belirlediği sınır olan son X maçtaki beraberlik sayısı	***39***	***Deplasman Deplasmanda Mağlubiyet Sayısı*** Deplasman takımının kendi sahasında mağlup olduğu maçların sayısı
6	***Ev Sahibi Son X Maç Mağlubiyet*** Ev sahibi takımın kullanıcının belirlediği sınır olan son X maçtaki mağlubiyet sayısı	***40***	***Deplasman Deplasmanda Mağlubiyet Yüzde*** Deplasman takımının kendi sahasında mağlup olma yüzdesi
7	***Ev Sahibi Toplam Puan*** Ev sahibi takımın sezon başından beri topladığı toplam puan	***41***	***Deplasman Galibiyete Uzaklık*** Deplasman takımının en son kaç hafta önce galip geldiğinin sayısı
8	***Ev Sahibi Son X Maç Puan*** Ev sahibi takımın son X maçta elde ettiği toplam puan	***42***	***Deplasman Beraberliğe Uzaklık*** Deplasman takımının en son ne zaman berabere kaldığının sayısı
9	***Ev Sahibi Galibiyet Yüzde*** Ev sahibi takımın galibiyet yüzdesi	***43***	***Deplasman Mağlubiyete Uzaklık*** Deplasman takımının en son kaç hafta önce mağlup olduğunun sayısı
10	***Ev Sahibi Beraberlik Yüzde*** Ev sahibi takımın beraberlik yüzdesi	***44***	***Deplasman En Uzun Galibiyet Serisi*** Deplasman takımının üst üste en fazla kaç maçta galip geldiğinin sayısı
11	***Ev Sahibi Mağlubiyet Yüzde*** Ev sahibi takımın mağlubiyet yüzdesi	***45***	***Deplasman En Uzun Mağlubiyet Serisi*** Deplasman takımının üst üste en fazla kaç maçta mağlup olduğunun sayısı
12	***Ev Sahibi Evde Galibiyet Sayısı***	***46***	***Deplasman En Uzun Beraberlik Serisi***

Ev sahibi takımın kendi sahasında galip geldiği maçların sayısı

13 ***Ev Sahibi Evde Galibiyet Yüzde***
Ev sahibi takımın kendi sahasında galip gelme yüzdesi

14 ***Ev Sahibi Evde Beraberlik Sayısı***
Ev sahibi takımın kendi sahasında berabere kaldığı maçların sayısı

15 ***Ev Sahibi Evde Beraberlik Yüzde***
Ev sahibi takımın kendi sahasında berabere kalmaz yüzdesi

16 ***Ev Sahibi Evde Mağlubiyet Sayısı***
Ev sahibi takımın kendi sahasında mağlup olduğu maçların sayısı

17 ***Ev Sahibi Evde Mağlubiyet Yüzde***
Ev sahibi takımın kendi sahasında mağlup olma yüzdesi

18 ***Ev Sahibi Galibiyete Uzaklık***
Ev sahibi takımın en son kaç hafta önce galip geldiğinin sayısı

19 ***Ev Sahibi Beraberliğe Uzaklık***
Ev sahibi takımın en son ne zaman berabere kaldığının sayısı

20 ***Ev Sahibi Mağlubiyete Uzaklık***
Ev sahibi takımın en son kaç hafta önce mağlup olduğunun sayısı

21 ***Ev Sahibi En Uzun Galibiyet Serisi***
Ev sahibi takımın üst üste en fazla kaç maçta galip geldiğinin sayısı

22 ***Ev Sahibi En Uzun Mağlubiyet Serisi***
Ev sahibi takımın üst üste en fazla kaç maçta mağlup olduğunun sayısı

23 ***Ev Sahibi En Uzun Beraberlik Serisi***
Ev sahibi takımın üst üste en fazla kaç maçta berabere kaldığının sayısı

24 ***Deplasman Toplam Galibiyet***
Deplasman takımının o sezonki toplam galibiyet sayısı

25 ***Deplasman Toplam Beraberlik***
Deplasman takımının o sezonki toplam beraberlik sayısı

26 ***Deplasman Toplam Mağlubiyet***
Deplasman takımının o sezonki toplam mağlubiyet sayısı

27 ***Deplasman Son X Maç Galibiyet***
Deplasman takımının kullanıcının belirlediği sınır olan son X maçtaki galibiyet sayısı

28 ***Deplasman Son X Maç Beraberlik***
Deplasman takımının kullanıcının belirlediği sınır olan son X maçtaki beraberlik sayısı

29 ***Deplasman Son X Maç Mağlubiyet***
Deplasman takımının kullanıcının belirlediği sınır olan son X maçtaki mağlubiyet sayısı

30 ***Deplasman Toplam Puan***
Deplasman takımının sezon başından beri topladığı toplam puan

31 ***Deplasman Son X Maç Puan***
Deplasman takımının son X maçta elde ettiği toplam puan

32 ***Deplasman Galibiyet Yüzde***
Ev sahibi takımın galibiyet yüzdesi

33 ***Deplasman Beraberlik Yüzde***
Ev sahibi takımın beraberlik yüzdesi

34 ***Deplasman Mağlubiyet Yüzde***
Ev sahibi takımın mağlubiyet yüzdesi

Deplasman takımının üst üste en fazla kaç maçta berabere kaldığının sayısı

47 ***Ev Sahibi Tek Farklı Galibiyet Sayısı***
Ev sahibi takımın tek farklı galibiyetlerinin sayısı

48 ***Ev Sahibi Handikaplı Galibiyet Sayısı***
Ev sahibi takımın en az 2 farkla kazandığı maçların sayısı

49 ***Ev Sahibi Tek Farklı Mağlubiyet Sayısı***
Ev sahibi takımın tek farklı mağlubiyetlerinin sayısı

50 ***Ev Sahibi Handikaplı Mağlubiyet Sayısı***
Ev sahibi takımın en az 2 farkla kaybettiği maçların sayısı

51 ***Deplasman Tek Farklı Galibiyet Sayısı***
Deplasman takımının tek farklı galibiyetlerinin sayısı

52 ***Deplasman Handikaplı Galibiyet Sayısı***
Deplasman takımının en az 2 farkla kazandığı maçların sayısı

53 ***Deplasman Tek Farklı Mağlubiyet Sayısı***
Deplasman takımının tek farklı mağlubiyetlerinin sayısı

54 ***Deplasman Handikaplı Mağlubiyet Sayısı***
Deplasman takımının en az 2 farkla kaybettiği maçların sayısı

55 ***Deplasman Son 1. Maç Sonucu***
Deplasman takımının 1 önceki maçının sonucu (galibiyet:2, beraberlik:0, mağlubiet:1)

56 ***Deplasman Son 2. Maç Sonucu***
Deplasman takımının 2 önceki maçının sonucu (galibiyet:2, beraberlik:0, mağlubiet:1)

57 ***Deplasman Son 3. Maç Sonucu***
Deplasman takımının 3 önceki maçının sonucu (galibiyet:2, beraberlik:0, mağlubiet:1)

58 ***Deplasman Son 4. Maç Sonucu***
Deplasman takımının 4 önceki maçının sonucu (galibiyet:2, beraberlik:0, mağlubiet:1)

59 ***Deplasman Son 5. Maç Sonucu***
Deplasman takımının 5 önceki maçının sonucu (galibiyet:2, beraberlik:0, mağlubiet:1)

60 ***Ev Sahibi Son 1. Maç Sonucu***
Ev sahibi takımın 1 önceki maçının sonucu (galibiyet:1, beraberlik:0, mağlubiet:2)

61 ***Ev Sahibi Son 2. Maç Sonucu***
Ev sahibi takımın 2 önceki maçının sonucu (galibiyet:1, beraberlik:0, mağlubiet:2)

62 ***Ev Sahibi Son 3. Maç Sonucu***
Ev sahibi takımın 3 önceki maçının sonucu (galibiyet:1, beraberlik:0, mağlubiet:2)

63 ***Ev Sahibi Son 4. Maç Sonucu***
Ev sahibi takımın 4 önceki maçının sonucu (galibiyet:1, beraberlik:0, mağlubiet:2)

64 ***Ev Sahibi Son 5. Maç Sonucu***
Ev sahibi takımın 5 önceki maçının sonucu (galibiyet:1, beraberlik:0, mağlubiet:2)

65 ***Toplam Puan Farkı***
Birinci takımın puanı ile ikinci takımın puanları arasındaki fark

66 ***Ev Sahibi Hedef Maçı mı?***
Ev sahibi takım kazanması için bir hedef var mı? (şampiyonluk, kupa, küme düşme vs.)

67 ***Deplasman Hedef Maçı mı?***
Deplasman takımının kazanması için bir hedef var mı? (şampiyonluk, kupa, küme düşme vs.)

68 ***Takım İlk 11 Kadro Değeri Oranı***
Birinci takımın ilk 11 değeri (M€) / İkinci takımın ilk 11 Değeri (M€)

5.3.5. Veri Setinin Hazırlanması

Veri Dosyası Oluşturma Adımları

Çalışmada kullanılacak veri seti sırasıyla aşağıdaki adımlar gerçekleştirilerek oluşturulur.

Adım 1. *"fikstür"* tablosundan sıra ile maç bilgileri alınır.

- İlk gelen maç A ve B takımları arasında T_1 tarihinde gerçekleştirilen bir maç olsun.

Adım 2. *A* ve *B* takımları, o sezon T_1 tarihinden önce *"Önceki Maçların Aralığı"* bölümünde kullanıcı tarafından belirlenen *"M"* sınırının altında maç yapmışsalar *Adım 1*'e geri dönülür ve yeni bir maç bilgisi gelir. Eğer en az *"M"* sayıda maç yapmışsalar *Adım 3*'ten devam edilir.

- A ve B takımları arasında gerçekleştirilen maçın bilgilerinin yer aldığı vektör $\mathbf{V}_{AB}$ olsun.

Adım 3. *"fikstür"* tablosundan gelen bu maçın bilgilerine göre *"maçgünü"* tablosu filtrelenir ve bu tablodan *A* ve *B* takımları arasında oynanacak olan maç için "maç günü verileri" elde edilir. Bu veriler "Maç Günü Verilerinin Toplanması" bölümünde de açıklandığı üzere toplam 58 adet öznitelikten oluşur. Bağımlı değişken olarak da maç sonucunun eklenmesiyle birlikte $\mathbf{V}_{AB}$ vektörünün ilk hali oluşur.

$$\mathbf{V}_{AB_{macgunu}} = \left[x_{AB_1}, x_{AB_2}, \ldots, x_{AB_{58}}, y_{AB} \right]_{1\times 59}$$

$$\mathbf{V}_{AB^1} = \mathbf{V}_{AB_{macgunu}} = \left[x_{AB_1}, x_{AB_2}, \ldots, x_{AB_{58}}, y_{AB} \right]_{1\times 59}$$

Adım 4. Bu adımda, veri setine *A* ve *B* takımlarının önceki haftalarda oynadıkları maçlarda elde ettikleri saha içi ortalama performans istatistiklerinin kıyaslanması ile elde edilen 65 yeni öznitelik eklenir. Kıyaslama sonucu elde edilen bu özniteliklerin nasıl oluşturulduğu *"Takımların Saha İçi Performanslarının Hesaplanması"* bölümünde detaylı şekilde açıklanmıştır. Böylelikle $\mathbf{V}_{AB}$ vektörü yeni halini almıştır.

$$\mathbf{V}_{AB_{saha_içi_performans}} = \left[x_{AB_{59}}, x_{AB_{60}}, \ldots, x_{AB_{123}}, y_{AB} \right]_{1\times 66}$$

$$\mathbf{V}_{AB^2} = \mathbf{V}_{AB_{macgunu}} + \mathbf{V}_{AB_{saha_ici_performans}} = \left[x_{AB_1}, x_{AB_2}, \ldots, x_{AB_{123}}, y_{AB} \right]_{1\times 124}$$

Adım 4. Bu adımda da *"Yardımcı Değişkenlerin Türetilmesi"* bölümünde nasıl açıklanan toplam 68 yeni özniteliğin eklenir ve $\mathbf{V}_{AB}$ vektörü son halini almış olur.

$$\mathbf{V}_{AB_{türetilenler}} = \left[x_{AB_{124}}, x_{AB_{125}}, \ldots, x_{AB_{191}}, y_{AB} \right]_{1\times 69}$$

$$\mathbf{V}_{AB^3} = \mathbf{V}_{AB_{macgunu}} + \mathbf{V}_{AB_{saha_içi_performans}} + \mathbf{V}_{AB_{türetilenler}} = \left[x_{AB_1}, x_{AB_2}, \ldots, x_{AB_{191}}, y_{AB} \right]_{1\times 192}$$

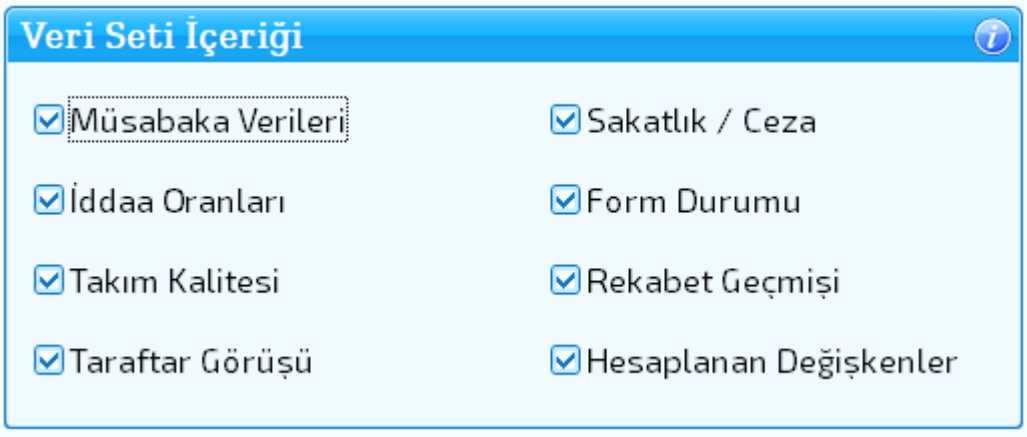
Bu dört adım *"fikstür"* tablosundaki tüm maçlar için tekrarlanır. *"fikstür"* tablosunda yer alan 6396 maç içerisinde her iki takımında en az *M* sayıda maç yaptığı *N* adet maç olduğunu varsayarsak ortaya çalışmada kullanılabilecek *N*x192 boyutlu bir veri seti çıkar. Bu veri setini aşağıdaki gibi ifade edilebilir.

$$\mathbf{E} = \begin{bmatrix} x_{1_1} & x_{1_2} & \cdots & x_{1_{191}} & y_1 \\ x_{2_1} & x_{2_2} & \cdots & x_{2_{191}} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{N_1} & x_{N_2} & \cdots & x_{N_{191}} & y_N \end{bmatrix}_{(N\times 192)}$$

Bu matris çalışmanın uygulama sürecinde kullanılmak üzere *"eğitim"* isimli bir veri tabanı tablosuna kaydedilmiştir. Oluşturulan bu veri tabanı tablosu, veri setinin belli filtrelere göre yeniden düzenlenerek veri dosyalarına dönüştürülmesi aşamasında kullanılmıştır.

Dosya Oluşturma Seçenekleri

Uygulamanın bu aşamasında daha önceden *"eğitim"* isimli bir veri tabanı tablosuna kaydedilen veri setinin veri dosyalarına dönüştürülme süreci planlanır. Resim 5.19'da gösterilen "Veri Seti İçeriği" bölümünde veri dosyasının içeriğinde yer alması istenilen kategoriler belirlenir.



Resim 5.19. Veri seti içeriği

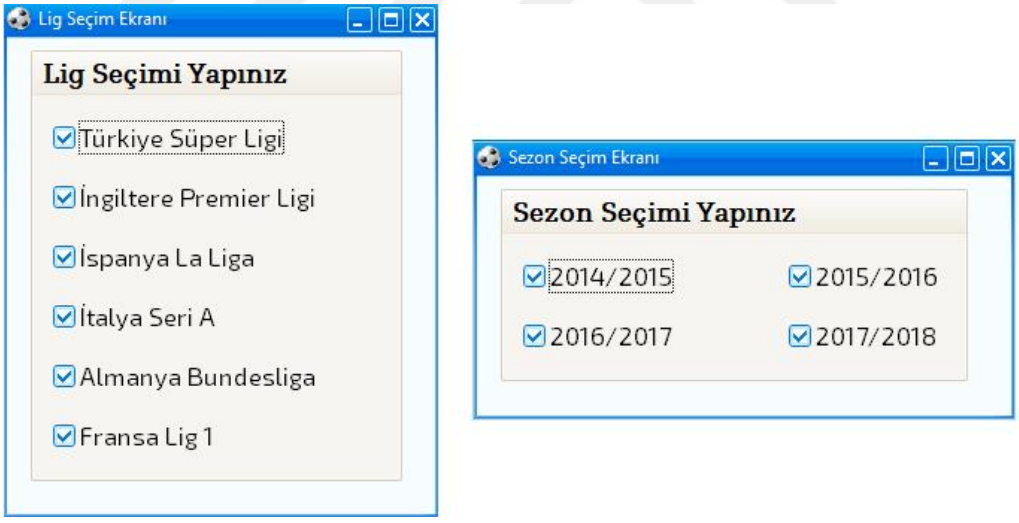
Veri dosyasının içeriğine karar verdikten sonra oluşturulacak olan bu dosyanın yapısını belirlemek adına Resim 5.20'da gösterilen *"Eğitim Dosyası Çıktı Seçenekleri"* düzenlenir. Burada kullanıcı veri dosyasını dilerse tek parça halinde dilerse de belli bir yüzdeye göre eğitim ve test için belirlenmiş iki dosya halinde dışarıya çıkartabilir. Bu

bölümde aynı zamanda Resim 5.21'de gösterilen lig ve sezon filtreleri de kullanılarak veri dosyaları oluşturulabilir.

Bu bölümdeki son özelleştirilebilir alan *"Takımların verilerini ayrı ayrı ver"* seçeneğidir. Bu seçenek işaretlendiğinde *"Takımların Saha İçi Performanslarının Hesaplanması"* bölümünde 1x65'lik tek bir vektörde birleştirilen *A* ve *B* takımlarının saha içi performans değerleri ayrı ayrı 1x65'lik birer vektör olarak veri setine eklenir. Böyle bir durumda veri setindeki öznitelik sayısı 65 artış gösterecektir. Bu seçenek uygulamanın başka amaçlarda oluşturulabilecek veri setlerine de imkan tanıması açısından eklenmiştir ve bu çalışmanın veri seti oluşturulurken kullanılmayacaktır.

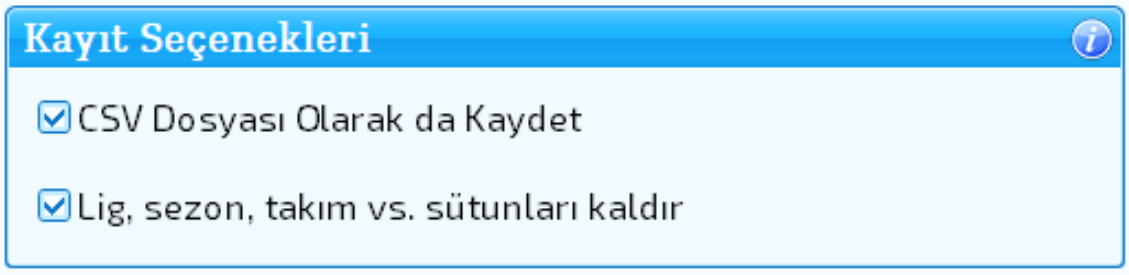


Resim 5.20. Eğitim dosyası çıktı seçenekleri



Resim 5.21. Lig ve sezon filtre seçenekleri

Eğitim ve test veri dosyalarını yapısı ayarlandıktan sonra son olarak da oluşacak olan dosyaların kayıt türleri ve sütun başlıkları ile ilgili özelleştirmeler Resim 5.22'de gösterilen *"Kayıt Seçenekleri"* bölümünden yapılır. Bu bölümde, standart olarak *"xlsx"* formatında kaydedilecek dosyanın aynı zamanda *"csv"* olarak da kaydedilip edilmeyeceğine ve analizde kullanılmayacak bir takım nitel ifadeler içeren sütunların gizlenip gizlenmeyeceğine ilişkin seçenekler vardır.



Resim 5.22. Resim Eğitim dosyası kayıt seçenekleri

5.3.6. Veri Normalleştirme

Veri seti oluşturma işleminin ardından ulaşılan veri seti, üçüncü bölümdeki *"Veri Ön İşleme"* başlığı altından anlatılan normalleştirme yöntemleri ile normalleştirilmiştir. Gerçekleştirilen genel bir sınıflandırma sonucunda veri normalleştirme yöntemlerin modelin başarısına etkilerinin anlamlı olmadığı tespit edilmiştir. Bu nedenle literatürde sıklıkla kullanılan ve işlem kolaylığı açısından son derece basit bir normalleştirme yöntemi olan (Jain ve ark., 2005) *"min-max normalleştirme yöntemi"* çalışmanın geri kalanında tek normalleştirme alternatifi olarak değerlendirilmiştir.

5.3.7. Özellik Seçimi ve Özellik Çıkarımı

Bu çalışmada özellik seçimi gerçekleştirmek için, literatürde bir çok çalışmada kullanılan bilgi kazanç oranı (information gain ratio) ve korelasyon temelli yöntemlerden faydalanılmıştır (Hall, 1999; Karegowda ve ark., 2010). Özellik çıkarımı işlemi için de literatürde farklı formları yer alan temel bileşenler analizi kullanılmıştır (Wold ve ark., 1987; Tipping ve Bishop, 1999; Zou ve ark., 2006; Candès ve ark., 2011).

Özellik seçimi ve özellik çıkarımı sonucu elde edilen veri setleri ile ön testler gerçekleştirildiğinde, özellik seçimi sonrası oluşan veri setlerinin model performansına katkısının özellik çıkarımı sonucu oluşan veri setlerinden daha yüksek olduğu tespit edilmiştir. Bu nedenle çalışmanın uygulamaları gerçekleştirilirken veri setinin indirgenmesi amacıyla özellik seçiminden faydalanılmıştır.

Özellik seçimi sonrası elde edilen daha düşük boyutlu veri setinde bulunan değişkenler Çizelge 5.9'da gösterilmektedir.

Çizelge 5.9. Boyutu azaltılmış yeni veri seti

x_1	*Ev Sahibi İlk 11 Değer*	x_{26}	*Engellenmiş Şut Sayısı*
x_2	*Deplasman İlk 11 Değer*	x_{27}	*Altı Pastan Çekilen Şut Sayısı*
x_3	*MS:1 Bahis Oranı*	x_{28}	*Açık Oyunda Çekilen Şut Sayısı*
x_4	*MS:2 Bahis Oranı*	x_{29}	*Pas Yüzdesi*
x_5	*İlk Yarı:1 Bahis Oranı*	x_{30}	*Kilit Pas Sayısı*
x_6	*İlk Yarı:2 Bahis Oranı*	x_{31}	*Ara Pas Sayısı*
x_7	*İY/MS:1/1 Bahis Oranı*	x_{32}	*Ev Sahibi Galibiyet Yüzde*
x_8	*İY/MS 1/2 Bahis Oranı*	x_{33}	*Ev Sahibi Beraberlik Yüzde*
x_9	*İY/MS X/1 Bahis Oranı*	x_{34}	*Deplasman Galibiyet Yüzde*
x_{10}	*İY/MS 2/2 Bahis Oranı*	x_{35}	*Ev Sahibi Evde Galibiyet Yüzde*
x_{11}	*Ev Sahibi Oy Oranı*	x_{36}	*Toplam Puan Farkı*
x_{12}	*Ev Sahibi Olumlu Twit Sayısı*	x_{37}	*Ev Sahibi İçin Hedef Maçı mı?*
x_{13}	*Deplasman Olumlu Twit Sayısı*	x_{38}	*Deplasman Mağlubiyete Uzaklık*
x_{14}	*Ev Sahibi Sakat ve Cezalı Değeri*	x_{39}	*Takım Değeri Oranı*
x_{15}	*Deplasman Sakat ve Cezalı Değeri*	x_{40}	*Ev Sahibi Son X Maç Galibiyet*
x_{16}	*Ev Sahibi Sakat ve Cezalı Rating Oranı*	x_{41}	*Ev Sahibi Evde Galibiyet Sayısı*
x_{17}	*Deplasman Sakat ve Cezalı Rating Oranı*	x_{42}	*Deplasman Handikaplı Galibiyet Sayısı*
x_{18}	*Ev Sahibi İlk 11 Gole Katkı Toplam*	x_{43}	*Ev Sahibi Beraberliğe Uzaklık*
x_{19}	*Rating*	x_{44}	*Deplasman Handikaplı Mağlubiyet Sayısı*
x_{20}	*Topa Sahip Olma Yüzdesi*	x_{45}	*Ev Sahibi Mağlubiyete Uzaklık*
x_{21}	*Başarılı Adam Geçme Sayısı*	x_{46}	*Deplasman Toplam Beraberlik*
x_{22}	*Hava Topu Kazanma Yüzdesi*	x_{47}	*Deplasman En Uzun Beraberlik Serisi*
x_{23}	*Korner Sayısı*	x_{48}	*Ev Sahibi Tek Farklı Mağlubiyet Sayısı*
x_{24}	*Kurtartış Sayısı*	x_{49}	*Ev Sahibi Galibiyete Uzaklık*
x_{25}	*İsabetli Şut Sayısı*	x_{50}	*Ev Sahibi En Uzun Mağlubiyet Serisi*

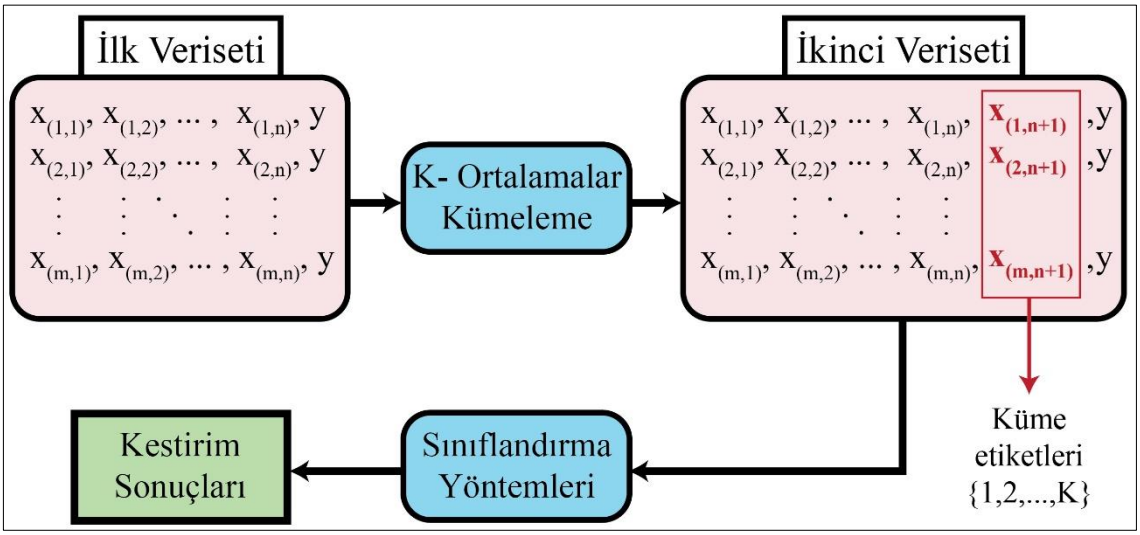
6. HİBRİT YÖNTEMLERİN YAPISI

Çalışma kapsamında sınıflandırma yöntemleri ile ulaşılabilecek olan kestirim başarısını daha üst seviyelere çıkarmak amacıyla hibrit yöntemler geliştirilmiştir. Bu modellerde kümeleme yöntemleri ile sınıflandırma yöntemleri bir arada kullanılmaktadır.

Yapılan işlemi kabaca açıklamak gerekirse; mevcut veri seti üzerinde gerçekleştirilen kümeleme analizi sonucunda ulaşılan değerlerin, yeni sütunlar halinde veri setine eklenmesi ve ardından da oluşturulan bu yeni veri seti üzerinde sınıflandırma analizi gerçekleştirilmesi şeklinde bir açıklama yapılabilir. Bu kümeleme analizi daha önceden herhangi bir boyut azaltma (özellik seçimi, özellik çıkarımı) işleminin uygulanmadığı veri seti üzerinde gerçekleştirilmemiştir.

İlk olarak kullanılacak kümeleme yöntemleri belirlenmiştir. Bu yöntemler *"Kullanılacak Yöntemler"* başlığında detaylıca açıklanan *"K-Ortalamalar Kümeleme"* ve *"Bulanık C Ortalamalar Kümeleme"* yöntemleridir.

"K-Ortalamalar Kümeleme" yöntemi ile elde edilen hibrit yöntemde küme sayısını belirten *K* sayısı deneme yanılma ile zamanla artırılmıştır. Her deneme sonrasında veri setindeki örneklerin hangi kümede etiketlendikleri kontrol edilerek, veri setine eklenecek olan yeni sütun bu küme etiketleri ile doldurulmuştur. Bu denemelerde her defasında, oluşan veri seti ile sınıflandırma analizi gerçekleştirilmiş ve performans değerleri kontrol edilmiştir. En yüksek performans değerlerine ulaşılan *K* değeri nihai değer olarak alınmıştır. Bu modelin yapısı Resim 6.1'de gösterilmektedir.



Resim 6.1. K-ortalamalar kümeleme ile elde edilen hibrit yöntemlerin yapısı

K-ortalamalar kümeleme yönteminin kullanılmasıyla elde edilen hibrit yöntemler;

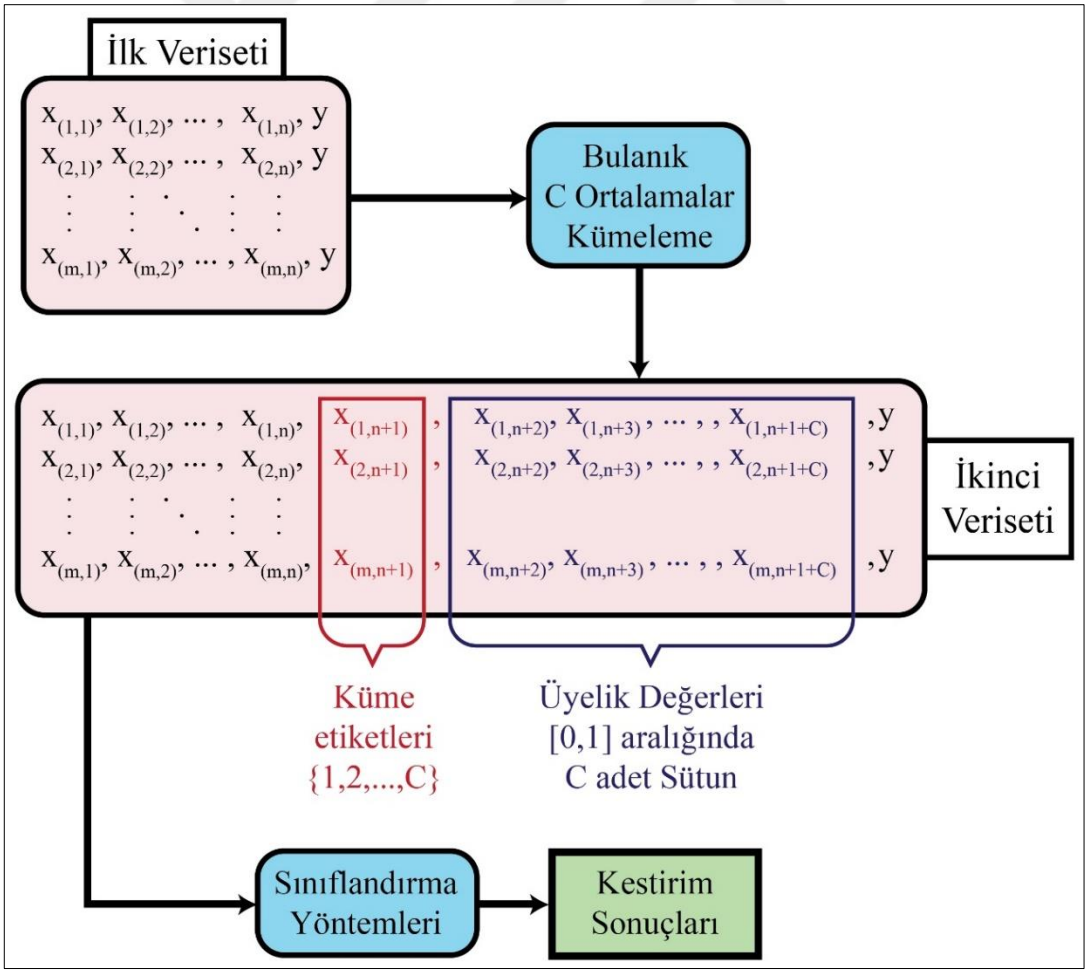
- K-Ortalamalar ve Destek Vektör Makinesi Hibrit Sınıflandırıcı (KO-DVM),

- K-Ortalamalar ve K-En Yakın Komşu Hibrit Sınıflandırıcı (KO-KEYK),
- K-Ortalamalar ve Lojistik Regresyon Hibrit Sınıflandırıcı (KO-LR),
- K-Ortalamalar ve Sade Bayes Hibrit Sınıflandırıcı (KO-SB),
- K-Ortalamalar ve Rasgele Orman Hibrit Sınıflandırıcı (KO-RO).
- K-Ortalamalar ve Yapay Sinir Ağları Hibrit Sınıflandırıcı (KO-YSA),

şeklinde isimlendirilmişlerdir.

Bulanık C ortalamalar kümeleme yöntemi ile oluşturulan hibrit yöntemlerde ise sınıf etiketlerine ek olarak üyelik dereceleri de küme sayısı kadar yeni sütun yaratılarak veri setine eklenmiştir.

Küme sayısını belirten *C* değeri tıpkı *K* ortalamalardaki *K* değeri gibi deneme yanılma ile artırılmıştır. Her deneme sonrası (*C*+1) sayıda yeni sütun veri setine eklenmiştir. Bu sütunlardan *C* tanesi örneklerin kümelere olan üyelik derecelerini, bir tanesi de küme etiketlerinin değerlerini göstermektedir. Bu modelin çalışma mantığı Resim 6.2'de gösterildiği şekildedir.



Resim 6.2. Bulanık C ortalamalar kümeleme ile oluşturulan hibrit yöntemlerin yapısı

Bulanık C-ortalamalar kümeleme yönteminin kullanılmasıyla elde edilen hibrit yöntemler ise;

- Bulanık C Ortalamalar ve Destek Vektör Makinesi Hibrit Sınıflandırıcı (BCC-DVM),
- Bulanık C Ortalamalar ve K-En Yakın Komşu Hibrit Sınıflandırıcı (BCO-KEYK),
- Bulanık C Ortalamalar ve Lojistik Regresyon Hibrit Sınıflandırıcı (BCO-LR),
- Bulanık C Ortalamalar ve Rasgele Orman Hibrit Sınıflandırıcı (BCO-RO),
- Bulanık C Ortalamalar ve Sade Bayes Hibrit Sınıflandırıcı (BCO-SB),
- Bulanık C Ortalamalar ve Yapay Sinir Ağları Hibrit Sınıflandırıcı (BCO-YSA),

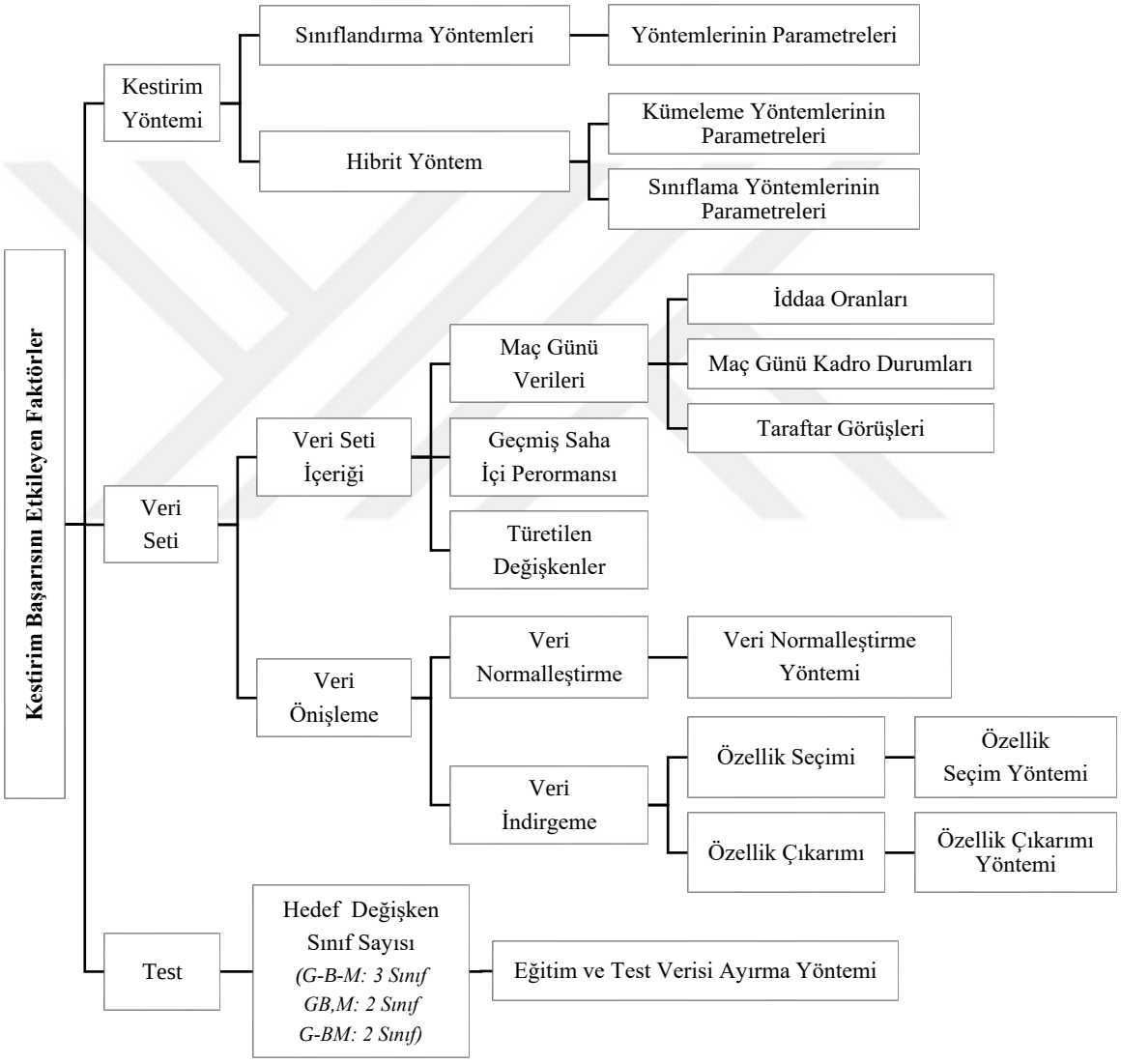
şeklinde isimlendirilmişlerdir.

7. TEST VE DEĞERLENDİRME SÜRECİ

7.1. Uygulama Sürecinin Planlanması

Bir kestirim çalışması sonucunda erişilecek olan performans değerlerini etkileyen çok sayıda faktör mevcuttur. Kullanılan kestirim yöntemi, veri setinin yapısı, veri önişleme araçları, test seçenekleri bu faktörlere örnek olarak verilebilir.

Bu çalışmada da Şekil 7.1'de üç başlık altında gösterilmekte olan faktörler kestirim başarısının sonucuna etki etmektedir.



Şekil 7.1. Kestirim başarısını etkileyen faktörler ve ilişkisel yapılar

Çalışmanın uygulama süreci, söz konusu bu faktörlerin, performans ölçütlerine olan etkilerine göre planlanmıştır. Bu süreçte çeşitli R ve Python kütüphanelerinin yanı sıra Matlab, Weka, SPSS ve MedCalc yazılımlarından da yararlanılmıştır.

7.2. Genelleştirilmiş Performans Ölçütleri

Çalışmanın üçüncü bölümünde test ve değerlendirme başlığı altında performans ölçütleri açıklanırken 2 sınıf etiketine sahip bir veri olduğu varsayılmıştı. Bu çalışmada ise uygulamaların bir kısmında maç sonucu değişkeni 3 sınıf olarak (G-M-B) ele alınmıştır. Bu durumda performans ölçütlerinin hesaplanabilmesi için her bir sınıf etiketi için karışıklık matrisinin genelleştirilmiş halleri oluşturulmuştur. Çizelge 7.1'de maç sonucunun üç sınıftan oluştuğu durumda elde edilen karışıklık matrisi gösterilmektedir. 3X3 boyutundaki bu karışıklık matrisinde sırası ile 0,1 ve 2 etiketleri için gerekli hesaplamalar yapılarak 2X2 boyutlu üç adet karışıklık matrisi oluşturulmuştur.

Çizelge 7.2, Çizelge 7.3 ve Çizelge 7.4'te yapıları hakkında bilgi verilen bu matrislerdeki TP, FP, TN ve FN değerlerinin ne anlama geldiği de aşağıda açıklanmıştır.

Çizelge 7.1. Çalışmanın genel karışıklık matrisi yapısı

			Sınıflandırma Sonucu		
			Beraberlik 0	*Galibiyet* 1	*Mağlubiyet* 2
Gerçek Sonuç	*Beraberlik*	0	K_{11}	K_{12}	K_{13}
	Galibiyet	1	K_{21}	K_{22}	K_{23}
	Mağlubiyet	2	K_{31}	K_{32}	K_{33}

Çizelge 7.2. Beraberliğe göre genelleştirilmiş karışıklık matrisi yapısı

		Sınıflandırma Sonucu	
		Beraberlik (0)	*Galibiyet (1) + Mağlubiyet (2)*
Gerçek Sonuç	*Beraberlik (0)*	TP = K_{11}	FN = $K_{12} + K_{13}$
	Galibiyet (1) + Mağlubiyet (2)	FP = $K_{21} + K_{31}$	TN = $K_{22} + K_{23} + K_{32} + K_{33}$

Çizelge 7.2'ye göre;

TP, gerçekte berabere bitmiş aynı zamanda sınıflandırma sonucunda da berabere şeklinde etiketlenmiş örnek sayısını,

TN, gerçekte berabere bitmemiş aynı zamanda sınıflandırma sonucunda da berabere şeklinde etiketlenmemiş örnek sayısını,

FP, gerçekte berabere bitmemiş ama sınıflandırma sonucunda berabere etiketlenmiş örnek sayısını,

FN, gerçekte berabere bitmiş ama sınıflandırma sonucunda berabere etiketlenmemiş örnek sayısını ifade etmektedir.

Çizelge 7.3. Galibiyete göre genelleştirilmiş karışıklık matrisi yapısı

		Sınıflandırma Sonucu	
		Galibiyet (1)	*Beraberlik (0)+ Mağlubiyet (2)*
Gerçek Sonuç	*Galibiyet (1)*	TP = K_{22}	FN = $K_{21} + K_{23}$
	Beraberlik (0)+ Mağlubiyet (2)	FP = $K_{12} + K_{32}$	TN = $K_{11} + K_{13} + K_{31} + K_{33}$

Çizelge 7.3'e göre;

TP, gerçekte galibiyetle bitmiş aynı zamanda sınıflandırma sonucunda da galibiyet şeklinde etiketlenmiş örnek sayısını,

TN, gerçekte galibiyetle bitmemiş aynı zamanda sınıflandırma sonucunda da galibiyet şeklinde etiketlenmemiş örnek sayısını,

FP, gerçekte galibiyetle bitmemiş ama sınıflandırma sonucunda galibiyetle etiketlenmiş örnek sayısı,

FN, gerçekte galibiyetle bitmiş ama sınıflandırma sonucunda galibiyetle etiketlenmemiş örnek sayısını ifade etmektedir.

Çizelge 7.4. Mağlubiyete göre genelleştirilmiş karışıklık matrisi yapısı

		Sınıflandırma Sonucu	
		Mağlubiyet (2)	*Beraberlik (0) + Galibiyet (1)*
Gerçek Sonuç	*Mağlubiyet (2)*	TP = K_{33}	FN = $K_{31} + K_{32}$
	Beraberlik (0) + Galibiyet (1)	FP = $K_{13} + K_{23}$	TN = $K_{11} + K_{12} + K_{21} + K_{22}$

Çizelge 7.4'e göre;

TP, gerçekte mağlubiyetle bitmiş aynı zamanda sınıflandırma sonucunda da mağlubiyet şeklinde etiketlenmiş örnek sayısını,

TN, gerçekte mağlubiyetle bitmemiş aynı zamanda sınıflandırma sonucunda da mağlubiyet şeklinde etiketlenmemiş örnek sayısını,

FP, gerçekte mağlubiyetle bitmemiş ama sınıflandırma sonucunda mağlubiyetle etiketlenmiş örnek sayısını,

FN, gerçekte mağlubiyetle bitmiş ama sınıflandırma sonucunda mağlubiyetle etiketlenmemiş örnek sayısını ifade etmektedir.

Maç sonucunun üç sınıf olarak ele alındığı durumlarda modele ait performans ölçütlerinin hesaplanmasında gerekli TP, TN, FP, FN değerleri, Çizelge 7.2, Çizelge 7.3 ve Çizelge 7.4'te gösterildiği gibi hesaplanır.

8. BULGULAR

Çalışmanın bulguları beş başlık altında paylaşılmıştır. Bu başlıkların her birinde ise maç sonucu değişkeninin üç farklı durumu için gerçekleştirilen uygulamaların çıktıları yer almaktadır.

Maç sonucu değişkeninin söz konusu bu üç durumunu kısaca açıklamak gerekirse;

MS-1 (G,B,M): İlk durumda, maç sonucu değişkeni, ilk veri seti üzerinde herhangi bir değişiklik yapılmadan *"Galibiyet"*, *"Beraberlik"* ve *"Mağlubiyet"* şeklinde üç sınıftan oluşan yapısıyla kullanılmıştır. *"Galibiyet" in* sınıf etiketi 1, *"Beraberlik"* in 0 ve *"Mağlubiyet"* in ise 2'dir.

MS-2 (GB,M): İkinci durumda, bağımlı değişken olan *"maç sonucu"* üzerinde yeniden kodlama yapılarak, *"Galibiyet"* ve *"Beraberlik"* sınıfları birleştirilmiştir. Böylelikle, *"Galibiyet veya Beraberlik"* ile *"Mağlubiyet"* şeklinde iki sınıflı bir bağımlı değişkene sahip olan yeni bir veri seti elde edilmiştir. *"Galibiyet veya Beraberlik"* durumunun sınıf etiketi 1, *"Mağlubiyet"* durumunun ise 0'dır.

MS-3 (G,BM): Son durumda ise *"Beraberlik"* ve *"Mağlubiyet"* sınıfları birleştirilerek, bağımlı değişkeni, *"Galibiyet veya Beraberlik"* ile *"Mağlubiyet"* şeklindeki iki sınıftan oluşan yeni bir veri seti oluşturulmuştur. *"Galibiyet"* durumunun sınıf etiketi 1, *"Beraberlik veya Mağlubiyet"* durumunun ise 0'dır.

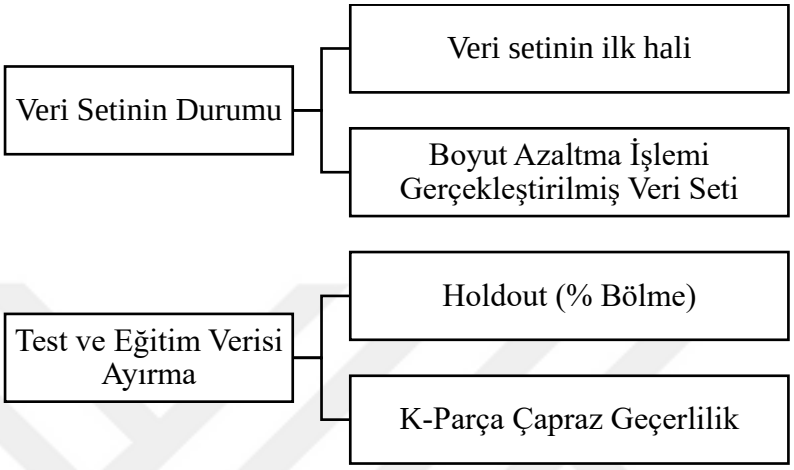
Bulgular bölümünün:

- İlk başlığında, kullanılan altı farklı sınıflandırma yönteminin, maç sonucu değişkeninin her üç durumu için ulaştığı performans değerleri tablolar ve grafiklerle paylaşılmıştır.
- İkinci başlığında KO kümeleme yönteminin sınıflandırma yöntemleri ile bir arada kullanılması sonucu oluşturulan hibrit yöntemlere ilişkin bulgular,
- Üçüncü başlığında BCO kümeleme yönteminin sınıflandırma yöntemleri ile birlikte kullanılmasıyla elde edilen hibrit yöntemlerin bulguları,
- Dördüncü başlıkta ise ilk üç başlıkta elde edilen bulgulara ilişkin ortalama değerler yer almaktadır.
- Son başlıkta da veri setinin içeriğinde gerçekleştirilen değişiklikler sonucu elde edilen bulgular paylaşılmıştır.

8.1. Sınıflandırma Yöntemlerine İlişkin Bulgular

Bu başlıkta paylaşılan tüm bulgular; Şekil 8.1'de gösterilen faktörlere göre çeşitlendirilmiştir. Ulaşılan performans değerleri, veri setine boyut azaltma uygulanıp

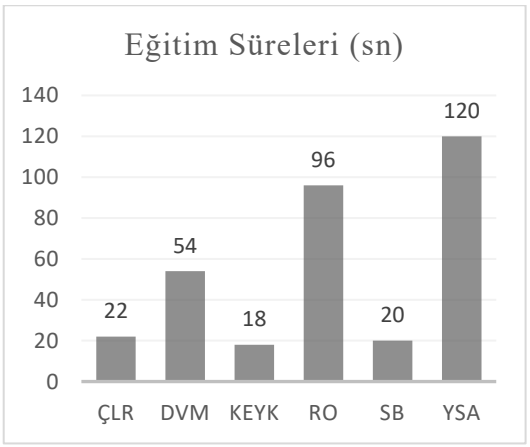
uygulanmadığı ve eğitim-test verisi ayırma işleminin nasıl gerçekleştiğine göre değişiklik göstermektedir. Boyutu azaltılmış veri seti olarak çalışmanın "Özellik Seçimi ve Çıkarımı" başlığı altında paylaşılan veri seti kullanılmıştır. Mevcut veri setinin eğitim ve test verisi diye ikiye ayrılması işlemi ise "Test ve Değerlendirme" başlığı altında açıklanan ayırma yöntemlerinden "holdout" ve "çapraz geçerlilik" kullanılarak yapılmıştır.



Şekil 8.1. MS-1 için paylaşılan bulguların başlıkları

Sınıflandırma yöntemlerinin R ve Weka aracılığyla, MS-1, MS-2 ve MS-3 durumları için gerçekleştirilen uygulamalarda elde ettiği ortalama eğitim sürelerine ilişkin değerler Şekil 8.2'de gösterilmektedir.

İşlemci : Intel Core i7-2600QM 2.2GHz
Ekran Kartı : NVIDIA GeForce GT 555M / 2048 MB / 3140MHz
HDD : 240GB SSD
RAM : 8192MB



Şekil 8.2. Sınıflandırma yöntemlerinin eğitim süreleri

8.1.1. MS-1 (G,B,M) Durumunda Ulaşılan Bulgular

Maç sonucunun 3 sınıf olarak ele alındığı bu bölümde, eğitim ve test uygulamaları gerçekleştirilirken kestirim performansını en üst düzeye çıkarabilmek adına, kullanılan yöntemlerin parametreleri deneme yanılma yoluyla birçok kez değiştirilmiştir. Bu deneme yanılma işlemleri neticesinde, her bir yöntemin kestirim başarısını en üst düzeye çıkaran değerler belirlenerek Çizelge 8.1'de paylaşılmıştır.

Çizelge 8.1. MS-1 Model parametreleri

Yöntem	Parametre	İlk Veri Seti	Boyutu Azaltılmış Veri Seti
ÇLR	*Conjugate gradient descent kullanımı*	var	var
	İterasyon Sayısı	150	150
DVM	*c*	3	3
	Çekirdek Fonksiyon	polly kernel	polly kernel
	Kalibratör	rasgele orman	rasgele orman
KEYK	*K*	91	34
	Komşu Arama Algoritması	doğrusal arama	doğrusal arama
	Uzaklık Fonksiyonu	manhatten uzaklığı	manhatten uzaklığı
	Uzaklık Ağırlıklandırma Yöntemi	1/uzaklık	1/uzaklık
RO	*İterasyon Sayısı*	500	250
	Kök Sayısı	1	1
SB	*Çekirdek Tahmin Edici Kullanımı*	var	var
YSA	*Gizli katman (GK) sayısı*	3	3
	GK Nöron Sayıları	3-7-11	3-5-9
	Momentum	0.2	0.2
	Öğrenme oranı	0.3	0.5

Çizelge 8.1'de paylaşılmış olan bu parametreler kullanılarak altı farklı sınıflandırma yöntemi ile maç sonucunun kestirilmesine yönelik eğitim ve test uygulamaları gerçekleştirilmiştir. Bu uygulamalar sonucu elde edilen, karışıklık matrisleri, Çizelge 8.2'de, karışıklık matrislerinin özeti, Çizelge 8.3'te, performans ölçütleri Çizelge 8.4'de, kestirim başarılarının grafiksel gösterimleri Şekil 8.3 ve Şekil 8.4'de, AUC değerlerine ilişkin bilgiler Çizelge 8.5 ve Çizelge 8.6'da, ROC eğrileri ise Şekil 8.5 ve Şekil 8.6'da paylaşılmıştır.

Çizelge 8.2. MS-1 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri

			ÇLR			DVM			KEYK			RO			SB			YSA		
			0	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**
İlk Veri Seti	**Holdout E=%80-T=%20**	**0**	45	54	19	40	60	18	27	67	24	29	65	24	**61**	40	17	49	53	16
		1	19	154	12	13	163	9	14	154	17	3	**167**	15	39	136	10	24	153	8
		2	34	30	76	24	37	79	14	40	86	13	35	**92**	44	20	76	27	34	79
	Çapraz Geç. K=10	**0**	190	234	158	167	258	157	89	309	184	131	270	181	**232**	183	167	64	321	197
		1	131	694	92	99	756	64	41	779	99	41	**795**	83	235	593	91	43	790	86
		2	122	158	436	107	165	444	40	198	478	55	167	**494**	155	101	460	33	215	468
Boyutu Azaltılmış Veri Seti	**Holdout E=%80-T=%20**	**0**	24	66	28	6	88	24	24	64	30	31	64	23	**55**	44	19	25	66	27
		1	12	160	13	0	**174**	11	13	161	11	8	165	12	31	145	9	8	162	15
		2	16	32	92	15	42	83	12	33	**95**	17	34	89	38	22	80	21	28	91
	Çapraz Geç. K=10	**0**	106	271	195	65	348	169	110	273	199	145	262	175	**221**	197	164	99	275	208
		1	76	753	90	21	**823**	75	56	468	95	58	780	81	207	641	71	72	751	96
		2	78	156	482	63	206	447	72	157	487	75	165	476	145	111	460	53	175	**488**

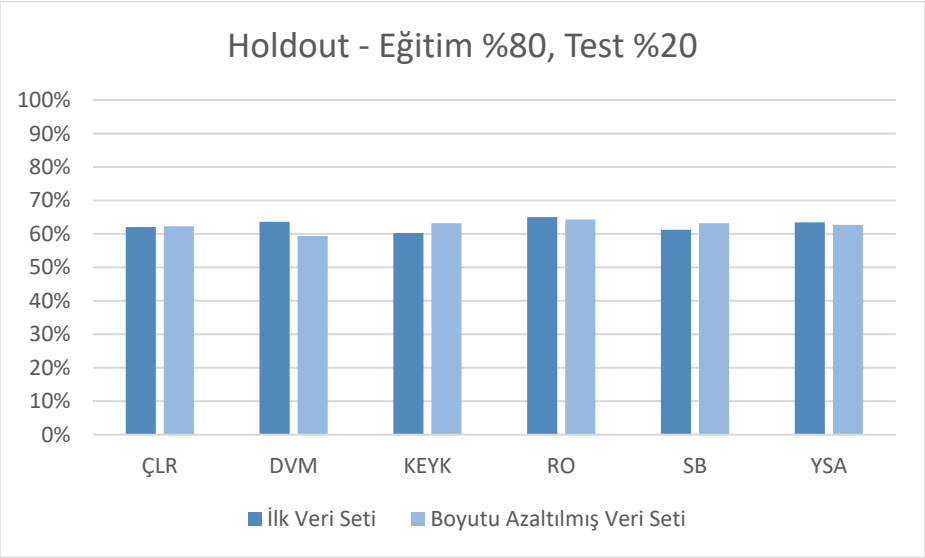
Çizelge 8.2'de paylaşılan karışıklık matrisleri incelendiğinde, gerçekleştirilen dört farklı test sonucunda maç sonucu değişkeninin G,B ve M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.3'te gösterilmektedir.

Çizelge 8.3. MS-1 için karışıklık matrislerinin özeti

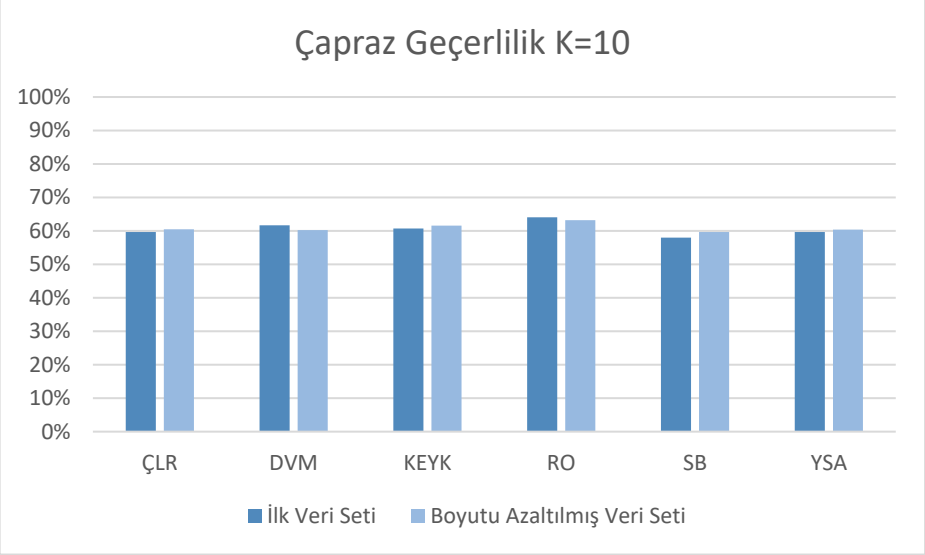
		0 (B - Beraberlik)		**1 (G - Galibiyet)**		**2 (M - Mağlubiyet)**	
		Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
İlk Veri seti	Holdout	SB	61/118	RO	167/185	RO	92/140
	Çap. Geç. K=10	SB	232/588	RO	795/919	RO	494/716
Boyutu Azaltılmış Veri Seti	Holdout	SB	55/118	DVM	174/185	KEYK	95/140
	Çap. Geç. K=10	SB	221/588	DVM	823/919	YSA	488/716

Çizelge 8.4. MS-1 için performans ölçütleri

			İlk Veri Seti						Boyutu Azaltılmış Veri Seti					
	Yöntem	*Sınıf*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*
Holdout Eğitim %80 - Test %20	ÇLR	0	0.381	0.837	0.459	0.417			0.203	0.914	0.462	0.282		
		1	0.832	0.674	0.647	0.726	0.41	62.08%	0.865	0.620	0.620	0.722	0.40	62.30%
		2	0.543	0.898	0.710	0.615			0.657	0.865	0.692	0.674		
	DVM	0	0.339	0.886	0.519	0.410			0.051	0.954	0.286	0.086		
		1	0.881	0.624	0.627	0.733	0.43	63.65%	0.941	0.494	0.572	0.712	0.34	59.36%
		2	0.564	0.911	0.745	0.642			0.593	0.884	0.703	0.643		
	KEYK	0	0.229	0.914	0.491	0.312			0.203	0.923	0.490	0.287		
		1	0.832	0.585	0.590	0.691	0.37	60.27%	0.870	0.624	0.624	0.727	0.42	63.20%
		2	0.614	0.865	0.677	0.644			0.670	0.865	0.699	0.688		
	RO	0	**0.246**	**0.951**	**0.644**	**0.356**			**0.263**	**0.923**	**0.554**	**0.356**		
		1	**0.903**	**0.612**	**0.625**	**0.739**	**0.44**	**65.01%**	**0.892**	**0.620**	**0.627**	**0.737**	**0.43**	**64.33%**
		2	**0.657**	**0.871**	**0.702**	**0.679**			**0.636**	**0.884**	**0.718**	**0.674**		
	SB	0	0.517	0.745	0.424	0.466			0.466	0.788	0.444	0.455		
		1	0.735	0.767	0.694	0.714	0.41	61.25%	0.784	0.744	0.687	0.732	0.43	63.20%
		2	0.543	0.911	0.738	0.626			0.571	0.908	0.741	0.645		
	YSA	0	0.415	0.843	0.490	0.450			0.212	0.911	0.463	0.291		
		1	0.827	0.663	0.638	0.720	0.43	63.43%	0.876	0.636	0.633	0.735	0.41	62.75%
		2	0.564	0.921	0.767	0.650			0.650	0.861	0.684	0.667		
Çapraz Geçerlilik K=10	ÇLR	0	0.326	0.845	0.429	0.371			0.182	0.906	0.408	0.252		
		1	0.757	0.698	0.640	0.694	0.37	59.63%	0.819	0.663	0.633	0.714	0.38	60.49%
		2	0.609	0.833	0.636	0.622			0.673	0.81	0.628	0.650		
	DVM	0	0.287	0.874	0.448	0.350			0.112	0.949	0.436	0.178		
		1	0.823	0.674	0.641	0.721	0.40	61.66%	0.896	0.573	0.598	0.717	0.36	60.22%
		2	0.620	0.853	0.668	0.643			0.624	0.837	0.647	0.635		
	KEYK	0	0.153	0.950	0.524	0.237			0.189	0.922	0.462	0.268		
		1	0.848	0.609	0.606	0.707	0.38	60.71%	0.836	0.669	0.641	0.726	0.39	61.57%
		2	0.668	0.813	0.628	0.647			0.680	0.804	0.624	0.651		
	RO	0	**0.225**	**0.941**	**0.577**	**0.324**			**0.249**	**0.919**	**0.522**	**0.337**		
		1	**0.865**	**0.663**	**0.645**	**0.739**	**0.43**	**64.05%**	**0.849**	**0.671**	**0.646**	**0.734**	**0.42**	**63.19%**
		2	**0.690**	**0.824**	**0.652**	**0.670**			**0.665**	**0.829**	**0.650**	**0.657**		
	SB	0	0.399	0.761	0.373	0.385			0.380	0.788	0.386	0.383		
		1	0.645	0.781	0.676	0.660	0.36	57.96%	0.697	0.763	0.675	0.686	0.38	59.63%
		2	0.642	0.828	0.641	0.642			0.642	0.843	0.662	0.652		
	YSA	0	0.110	0.954	0.457	0.177			0.170	0.924	0.442	0.246		
		1	0.680	0.587	0.596	0.704	0.36	59.63%	0.817	0.653	0.625	0.708	0.37	60.35%
		2	0.654	0.813	0.623	0.638			0.682	0.797	0.616	0.647		



Şekil 8.3. MS-1 için holdout veri seti ayırma yönteminde ulaşılan kestirim başarıları



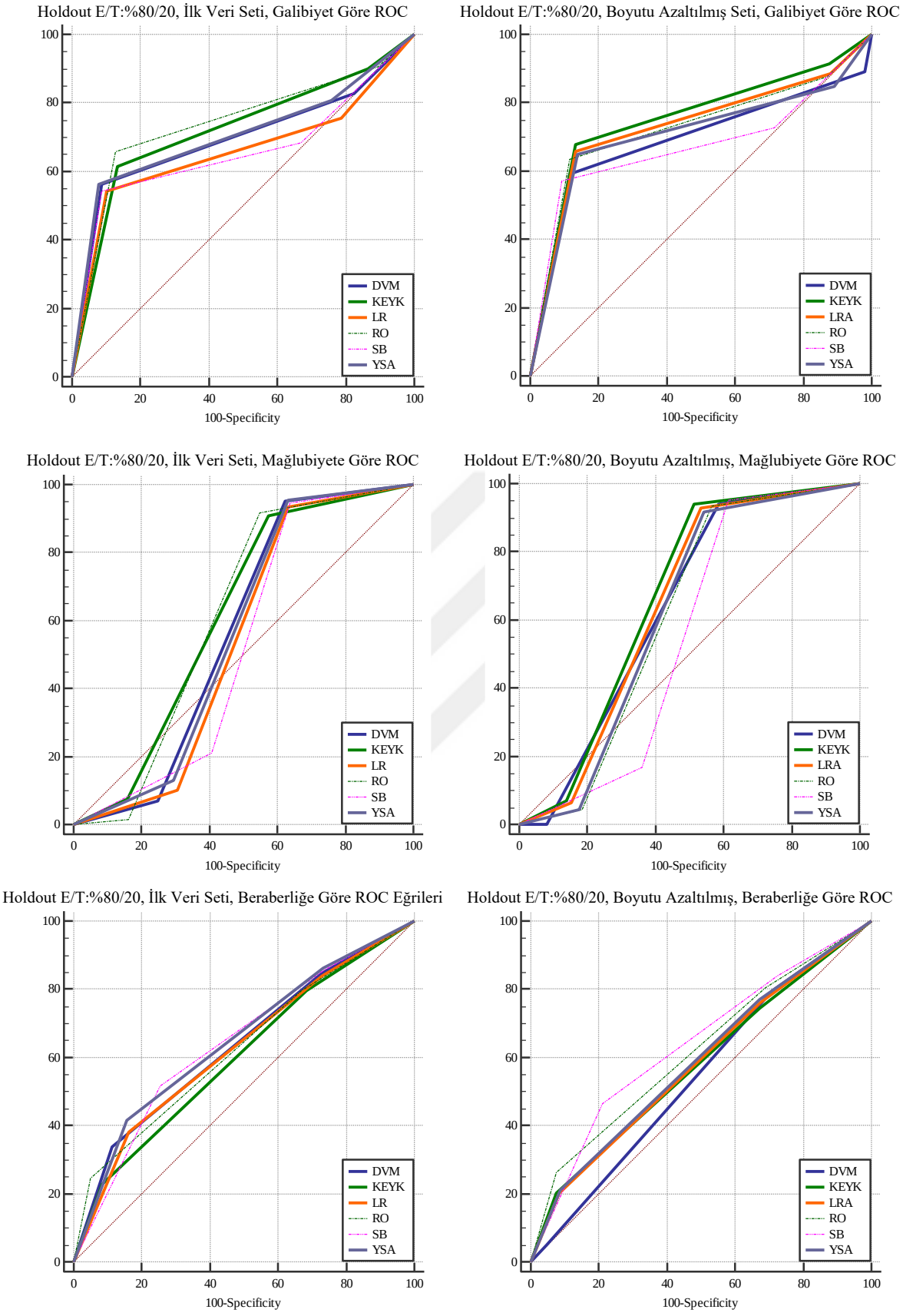
Şekil 8.4. MS-1 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan kestirim başarıları

Çizelge 8.5. MS-1 için holdout veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları

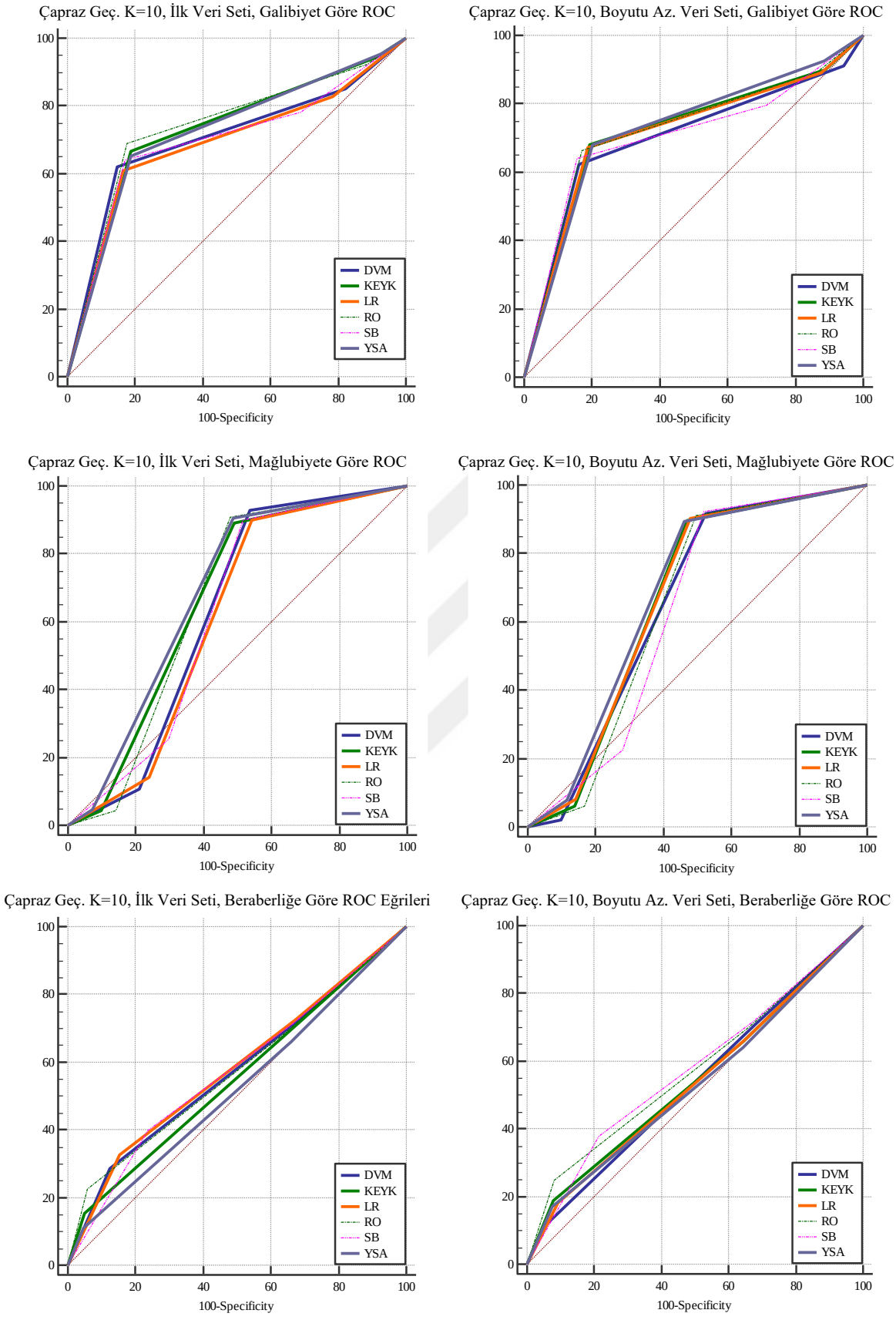
		Galibiyete Göre			Mağlubiyete Göre			Beraberliğe Göre		
	Yöntem	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**
İlk Veri Seti	ÇLR	0.660	0.0311	0.599 - 0.721	0.541	0.0254	0.491 - 0.591	0.626	0.0283	0.571 - 0.681
	DVM	0.698	0.0292	0.641 - 0.755	0.568	0.0244	0.520 - 0.616	0.634	0.0276	0.580 - 0.688
	KEYK	0.722	0.0266	0.670 - 0.774	**0.617**	**0.0233**	**0.571 - 0.663**	0.601	0.0274	0.547 - 0.655
	RO	**0.742**	**0.0265**	**0.690 - 0.794**	0.614	0.0235	0.568 - 0.660	0.626	0.0276	0.572 - 0.680
	SB	0.659	0.0313	0.598 - 0.720	0.528	0.0263	0.476 - 0.580	**0.641**	**0.0282**	**0.586 - 0.696**
	YSA	0.706	0.0291	0.649 - 0.763	0.563	0.0251	0.514 - 0.612	0.65	0.0277	0.596 - 0.704
Boyutu Azaltılmış Veri Seti	LR	0.732	0.0274	0.678 - 0.786	0.643	0.0232	0.598 - 0.688	0.579	0.0279	0.524 - 0.634
	DVM	0.695	0.0287	0.639 - 0.751	0.639	0.0206	0.599 - 0.679	0.543	0.024	0.496 - 0.590
	KEYK	**0.754**	**0.0258**	**0.703 - 0.805**	**0.665**	**0.0227**	**0.621 - 0.709**	0.576	0.0283	0.521 - 0.631
	RO	0.73	0.0277	0.676 - 0.784	0.61	0.0238	0.563 - 0.657	0.617	0.0277	0.563 - 0.671
	SB	0.677	0.031	0.616 - 0.738	0.548	0.0261	0.497 - 0.599	**0.637**	**0.0286**	**0.581 - 0.693**
	YSA	0.71	0.0292	0.653 - 0.767	0.618	0.0239	0.571 - 0.665	0.586	0.0278	0.532 - 0.640

Çizelge 8.6. MS-1 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları

		Galibiyete Göre			Mağlubiyete Göre			Beraberliğe Göre		
	Yöntem	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**
İlk Veri Seti	ÇLR	0.692	0.0125	0.668 - 0.717	0.609	0.0111	0.587 - 0.631	**0.580**	**0.0135**	**0.554 - 0.606**
	DVM	0.706	0.0125	0.682 - 0.731	0.627	0.0108	0.606 - 0.648	0.573	0.0134	0.547 - 0.599
	KEYK	0.731	0.0110	0.709 - 0.753	0.668	0.0100	0.648 - 0.688	0.551	0.0128	0.526 - 0.576
	RO	**0.743**	**0.0113**	**0.721 - 0.765**	0.660	0.0104	0.64 - 0.68	0.574	0.0133	0.548 - 0.6
	SB	0.701	0.0126	0.676 - 0.726	0.624	0.0113	0.602 - 0.646	0.572	0.0135	0.546 - 0.598
	YSA	0.726	0.0110	0.704 - 0.748	**0.687**	**0.0096**	**0.668 - 0.706**	0.521	0.0126	0.496 - 0.546
Boyutu Azaltılmış Veri Seti	ÇLR	0.717	0.0119	0.694 - 0.74	0.668	0.0104	0.648 - 0.688	0.535	0.0131	0.509 - 0.561
	DVM	0.705	0.0120	0.681 - 0.729	0.657	0.0098	0.638 - 0.676	0.534	0.0123	0.51 - 0.558
	KEYK	0.719	0.0118	0.696 - 0.742	0.664	0.0104	0.644 - 0.684	0.542	0.0132	0.516 - 0.568
	RO	0.726	0.0118	0.703 - 0.749	0.645	0.0106	0.624 - 0.666	**0.574**	**0.0134**	**0.548 - 0.6**
	SB	0.709	0.0126	0.684 - 0.734	0.630	0.0112	0.608 - 0.652	0.572	0.0136	0.545 - 0.599
	YSA	**0.728**	**0.0113**	**0.706 - 0.75**	**0.682**	**0.0101**	**0.662 - 0.702**	0.530	0.0132	0.504 - 0.556



Şekil 8.5. MS-1 için holdout veri seti ayırma yönteminde elde edilen ROC eğrileri



Şekil 8.6. MS-1 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen ROC eğrileri

Bu çizelge ve şekiller incelendiğinde, uygulanan sınıflandırma yöntemlerinin performanslarına dair ulaşılan bulgular aşağıda açıklandığı gibidir.

Boyut azaltma işlemi uygulanmamış veri seti üzerinde holdout veri seti ayırma yöntemi kullanılarak yapılan uygulamalarda, galibiyet, beraberlik ve mağlubiyet için ortalama olarak en yüksek kestirim başarısına RO yöntemi ile ulaşıldığı gözlemlenmektedir *(ACC:%65.01)*. RO'yu sırasyla, DVM *(ACC:%63.65)*, YSA *(ACC:%63.43)*, ÇLR *(ACC:%62.08)*, SB *(ACC:%61.25)* ve KEYK *(ACC:%60.27)* takip etmektedir.

Boyut azaltma işlemi uygulanmamış veri seti üzerinde çapraz geçerlilik veri seti ayırma yöntemi kullanılarak gerçekleştirilen uygulamaların performans ölçütleri incelendiğinde ise, RO yönteminin yine en yüksek kestirim başarısına ulaştığı görülmektedir *(ACC:%64.05)*. RO'yu bu kez de sırasıyla DVM *(ACC:%61.66)*, KEYK *(ACC:%60.71)*, ÇLR *(ACC:%59.63)*, YSA *(ACC:%59.63)* ve SB *(ACC:%57.96)* takip etmektedir.

Boyut azaltma işlemi sonrası elde edilen veri seti üzerinde holdout veri seti ayırma yöntemi kullanılarak gerçekleştirilen uygulamalarda en yüksek kestirim başarısına ulaşan yöntem RO'dur *(ACC:%64.33)*. RO'yu sırasıyla, KEYK *(ACC:%63.20)*, SB *(ACC:63.20)*, YSA *(ACC:62.75)*, ÇLR *(ACC:62.30)* ve DVM *(ACC:59.36)* takip etmektedir.

Boyut azaltma işlemi uygulanmış veri setinin çapraz geçerlilik yöntemi ile eğitim ve test verisine ayrılması sonrası gerçekleştirilen uygulamalarda da RO en yüksek kestirim başarısına ulaşan sınıflandırma yöntemi olmuştur *(ACC:%63.19)*. RO'yu bu kez de sırasıyla KEYK *(ACC:%61.57)*, ÇLR *(ACC:%60.49)*, YSA *(ACC:%60.35)*, DVM *(ACC:%60.22)* ve SB *(ACC:59.63)* yöntemleri takip etmektedir.

Kappa istatistikleri incelendiğinde ise en düşük kestirim başarısına ulaşan yöntem kullanılarak gerçekleştirilen sınıflandırmanın dahi rasgele yapılan bir sınıflandırmadan %32 oranla daha iyi olduğu görülmektedir. En iyi kestirim başarısının yakalandığı RO yönteminde ise bu durum %44'e kadar yükselmiştir.

Galibiyet, beraberlik ve mağlubiyet için elde edilen ROC eğrileri ve bu eğrilerin altında kalan alanlara bakıldığında ise galibiyet ve mağlubiyetin kestirilmesinde çok yüksek başarı oranları yakalanırken beraberlikle biten maçların doğru kestiriminde bu oranın aşağılara indiği gözlemlenmiştir.

Boyut azaltma işleminin, kestirim başarısına olan etkisine bakıldığında ise bu etkinin bazı sınıflandırma yöntemlerin pozitif bazılarında ise negatif olduğu gözlemlenmektedir. Bu değişimler ise Çizelge 8.7'de verilmiştir.

Çizelge 8.7. MS-1 için Boyut azaltma işleminin kestirim başarısına olan etkisi

	Holdout			Çapraz Geçerlilik			Ortalama		
	BA-*	**BA+***	**Değişim**	**BA-***	**BA+***	**Değişim**	**BA-***	**BA+***	**Değişim**
ÇLR	62.08%	**62.30%**	0.22%	59.63%	**60.49%**	0.86%	60.86%	**61.40%**	-0.54%
DVM	**63.65%**	59.36%	-4.29%	**61.66%**	60.22%	-1.44%	**62.66%**	59.79%	2.87%
KEYK	60.27%	**63.20%**	2.93%	60.71%	**61.57%**	0.86%	60.49%	**62.39%**	-1.90%
RO	**65.01%**	64.33%	-0.68%	**64.05%**	63.19%	-0.86%	**64.53%**	63.76%	0.77%
SB	61.25%	**63.20%**	1.95%	57.96%	**59.63%**	1.67%	59.61%	**61.42%**	-1.81%
YSA	**63.43%**	62.75%	-0.68%	59.63%	**60.35%**	0.72%	61.53%	**61.55%**	-0.02%

BA-: Boyut azaltılmamış veri seti, BA+: Boyut azaltılmış veri seti

8.1.2. MS-2 (GB,M) Durumunda Ulaşılan Bulgular

Maç sonucunun GB (ev sahibi yenilmez) / M (yenilir) şeklindeki iki sınıf olarak ele alındığı bu durum için gerçekleştirilen uygulamalarda kullanılan yöntemlerin performanslarını en üst düzeye çıkaran parametreler Çizelge 8.8'de paylaşılmıştır.

Çizelge 8.8. MS-2 Model parametreleri

Yöntem	Parametre	İlk Veri Seti	Boyutu Azaltılmış Veri Seti
ÇLR	*Conjugate gradient descent kullanımı*	var	var
	İterasyon Sayısı	150	150
DVM	*c*	1	1
	Çekirdek Fonksiyon	polly kernel	polly kernel
	Kalibratör	Sade bayes	rasgele orman
KEYK	*K*	91	35
	Komşu Arama Algoritması	doğrusal arama	doğrusal arama
	Uzaklık Fonksiyonu	manhatten uzaklığı	manhatten uzaklığı
	Uzaklık Ağırlıklandırma Yöntemi	1/uzaklık	1/uzaklık
RO	*İterasyon Sayısı*	500	200
	Kök Sayısı	1	1
SB	*Çekirdek Tahmin Edici Kullanımı*	var	var
YSA	*Gizli katman (GK) sayısı*	3	3
	GK Nöron Sayıları	3-5-9	6-6-12
	Momentum	0.2	0.2
	Öğrenme oranı	0.2	0.3

Çizelge 8.8'de verilen bu parametreler kullanılarak MS-2 durumu için çeşitli eğitim ve test uygulamaları gerçekleştirilmiştir. Gerçekleştirilen bu uygulamalar sonucunda elde edilen bulgulardan, karışıklık matrisleri, Çizelge 8.9'da, karışıklık matrislerinin özeti, Çizelge 8.10'da, performans ölçütleri Çizelge 8.11'de, kestirim başarılarının grafiksel gösterimleri Şekil 8.7 ve Şekil 8.8'de, AUC değerlerine ilişkin bilgiler Çizelge 8.12 ve Çizelge 8.13'te, ROC eğrileri ise Şekil 8.9 ve Şekil 8.10'da paylaşılmıştır.

Çizelge 8.9. MS-2 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri

			ÇLR		DVM		KEYK		RO		SB		YSA	
			0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
İlk Veri Seti	**Holdout**	**0**	75	65	84	56	58	82	77	63	95	45	**90**	50
		1	23	280	26	277	15	**288**	20	283	45	258	44	259
	Çapraz Geç. K=10	**0**	393	323	439	277	356	360	428	288	**493**	223	392	324
		1	196	1305	212	1289	148	**1353**	172	1329	331	1170	176	1325
Boyutu Azaltılmış Veri Seti	**Holdout**	**0**	67	73	65	75	74	66	75	65	**95**	45	73	67
		1	23	280	21	282	22	281	23	280	39	264	19	**284**
	Çapraz Geç. K=10	**0**	362	354	372	344	406	310	426	290	**507**	209	426	290
		1	159	**1342**	168	1333	173	1328	175	1326	303	1198	207	1294

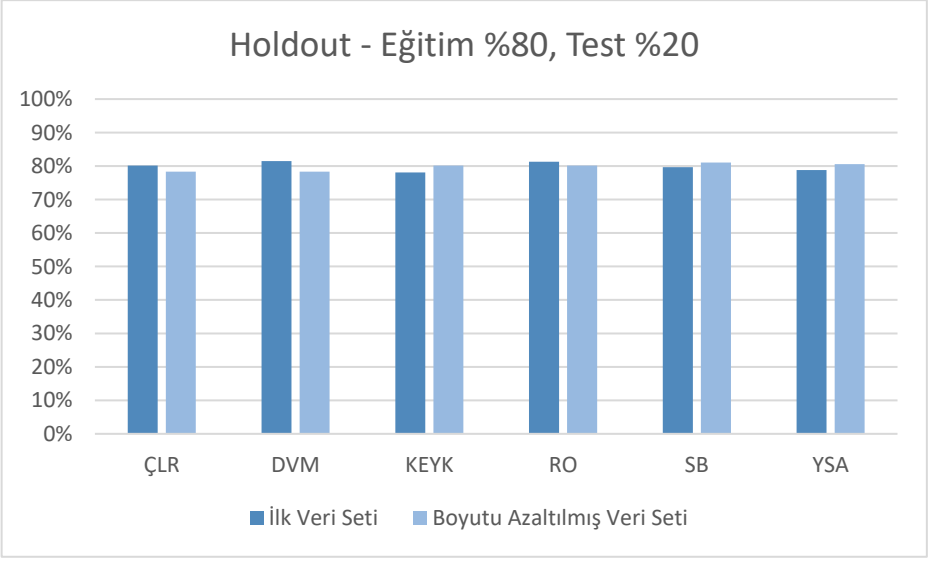
Çizelge 8.9'da paylaşılan karışıklık matrisleri incelendiğinde gerçekleştirilen dört farklı test sonucunda maç sonucu değişkeninin G-B ve M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.10'da gösterilmektedir.

Çizelge 8.10. MS-2 için karışıklık matrislerinin özeti

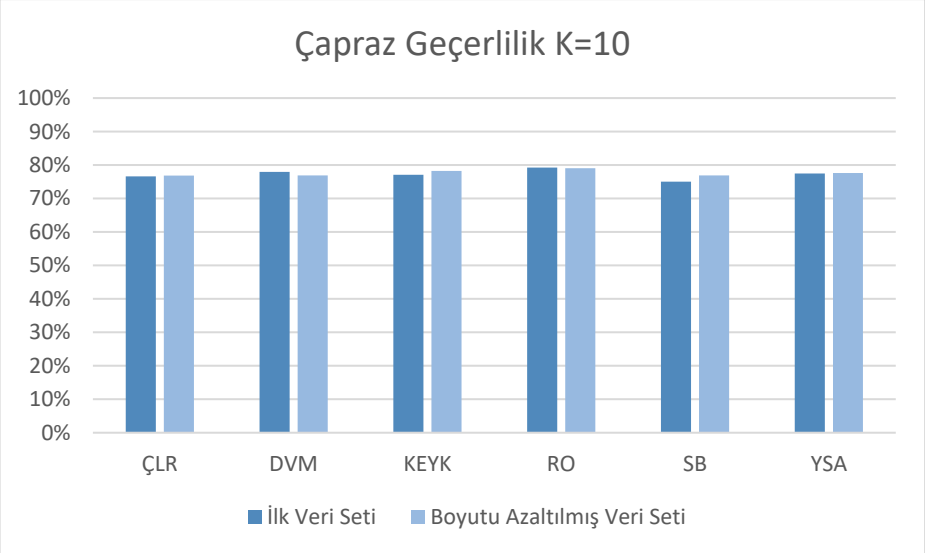
		0 (M - Mağlubiyet)		**1 (G-B) Galibiyet ve Beraberlik**	
		Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
İlk Veri seti	Holdout	YSA	90/140	KEYK	288/303
	Çap. Geç. K=10	SB	493/716	KEYK	1353/1501
Boyutu Azaltılmış Veri Seti	Holdout	SB	95/140	YSA	284/303
	Çap. Geç. K=10	SB	507/716	ÇLR	1342/1501

Çizelge 8.11. MS-2 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri

			İlk Veri Seti						Boyutu Azaltılmış Veri Seti					
	Yöntem	*Sınıf*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*
Eğitim %80 Test %20	ÇLR	0	0.536	0.924	0.765	0.630	0.50	80.14%	0.479	0.924	0.744	0.583	0.45	78.33%
		1	0.924	0.536	0.812	0.864			0.924	0.479	0.793	0.854		
	DVM	0	**0.600**	**0.914**	**0.764**	**0.672**	**0.55**	**81.49%**	0.464	0.931	0.756	0.575	0.44	78.33%
		1	**0.914**	**0.600**	**0.832**	**0.871**			0.931	0.464	0.790	0.855		
	KEYK	0	0.414	0.950	0.795	0.545	0.42	78.10%	0.529	0.927	0.771	0.627	0.50	80.14%
		1	0.950	0.414	0.778	0.856			0.927	0.529	0.810	0.865		
	RO	0	0.550	0.934	0.794	0.650	0.53	81.26%	0.536	0.924	0.765	0.630	0.50	80.14%
		1	0.934	0.550	0.818	0.872			0.924	0.536	0.812	0.864		
	SB	0	0.679	0.851	0.679	0.659	0.53	79.68%	**0.679**	**0.871**	**0.709**	**0.693**	**0.56**	**81.04%**
		1	0.851	0.679	0.851	0.851			**0.871**	**0.679**	**0.854**	**0.863**		
	YSA	0	0.643	0.855	0.672	0.657	0.50	78.78%	0.521	0.937	0.793	0.629	0.51	80.59%
		1	0.855	0.643	0.838	0.846			0.937	0.521	0.809	0.869		
K=10 Çapraz Geçerlilik	ÇLR	0	0.549	0.869	0.667	0.602	0.44	76.59%	0.506	0.894	0.695	0.585	0.43	76.86%
		1	0.869	0.549	0.802	0.834			0.894	0.506	0.791	0.840		
	DVM	0	0.613	0.859	0.674	0.642	0.48	77.94%	0.520	0.888	0.689	0.592	0.44	76.91%
		1	0.859	0.613	0.823	0.841			0.888	0.520	0.795	0.839		
	KEYK	0	0.497	0.901	0.706	0.584	0.43	77.08%	0.567	0.885	0.701	0.627	0.48	78.21%
		1	0.901	0.497	0.790	0.842			0.885	0.567	0.811	0.846		
	RO	**0**	**0.598**	**0.885**	**0.713**	**0.650**	**0.50**	**79.25%**	**0.595**	**0.883**	**0.709**	**0.647**	**0.50**	**79.03%**
		1	**0.885**	**0.598**	**0.822**	**0.852**			**0.883**	**0.595**	**0.821**	**0.851**		
	SB	0	0.689	0.779	0.598	0.640	0.45	75.01%	0.708	0.798	0.626	0.664	0.49	76.91%
		1	0.779	0.689	0.840	0.809			0.798	0.708	0.851	0.824		
	YSA	0	0.548	0.883	0.690	0.611	0.45	77.45%	0.595	0.862	0.673	0.632	0.67	77.58%
		1	0.883	0.547	0.804	0.841			0.862	0.595	0.817	0.839		



Şekil 8.7. MS-2 için holdout veri seti ayırma yönteminde ulaşılan kestirim başarıları



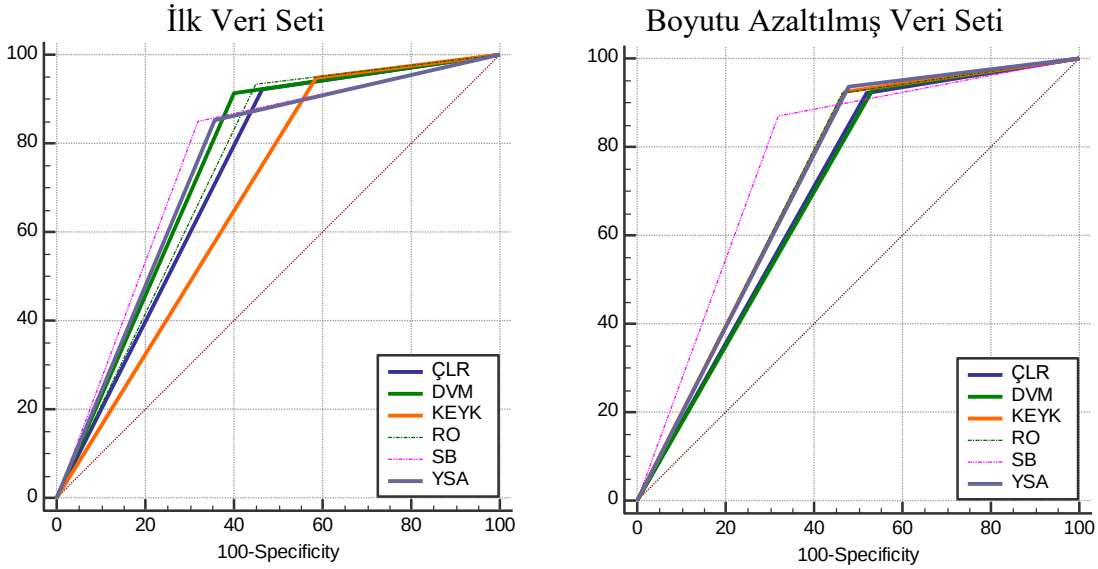
Şekil 8.8. MS-2 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan kestirim başarıları

Çizelge 8.12. MS-2 için holdout veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları

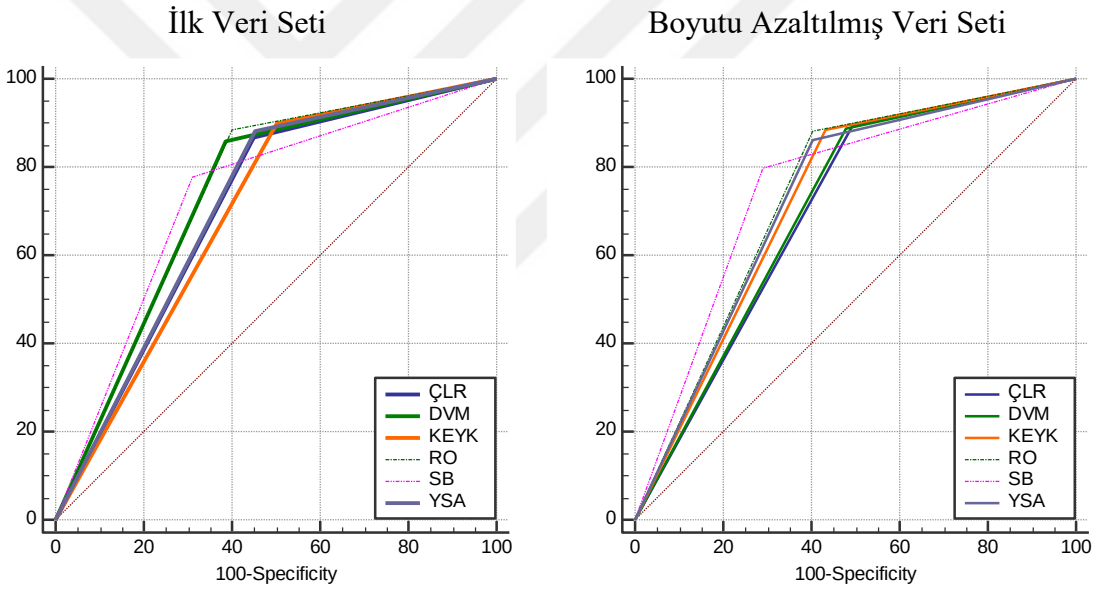
	İlk Veri Seti			Boyutu Azaltılmış Veri Seti		
Yöntem	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**
ÇLR	0.730	0.0225	0.686 - 0.774	0.701	0.0225	0.657 - 0.745
DVM	0.757	0.0223	0.713 - 0.801	0.697	0.0224	0.653 - 0.741
KEYK	0.682	0.0218	0.639 - 0.725	0.728	0.0224	0.684 - 0.772
RO	0.742	0.0223	0.698 - 0.786	0.73	0.0225	0.686 - 0.774
SB	**0.765**	**0.0223**	**0.721 - 0.809**	**0.775**	**0.022**	**0.732 - 0.818**
YSA	0.749	0.0227	0.705 - 0.793	0.729	0.0223	0.685 - 0.773

Çizelge 8.13. MS-2 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları

	İlk Veri Seti			Boyutu Azaltılmış Veri Seti		
Yöntem	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**
ÇLR	0.709	0.0103	0.689 - 0.729	0.700	0.0102	0.68 - 0.72
DVM	0.736	0.0102	0.716 - 0.756	0.704	0.0102	0.684 - 0.724
KEYK	0.699	0.0101	0.679 - 0.719	0.726	0.0101	0.706 - 0.746
RO	**0.742**	**0.0100**	**0.722 - 0.762**	0.739	0.0101	0.719 - 0.759
SB	0.734	0.0102	0.714 - 0.754	**0.753**	**0.0099**	**0.734 - 0.772**
YSA	0.715	0.0102	0.695 - 0.735	0.729	0.0102	0.709 - 0.749



Şekil 8.9. MS-2 için holdout veri seti ayırma yönteminde ulaşılan ROC eğrileri



Şekil 8.10. MS-2 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan ROC eğrileri

Bağımlı değişkenin ev sahibinin yenilip yenilmemesine göre oluşturulduğu bu durumda gerçekleştirilen test çalışmalarına ait bulgular şu şekildedir.

Boyut azaltma işlemi uygulanmayan veri setinin, holdout veri seti ayırma yöntemi ile eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda en yüksek performans değerlerine DVM *(ACC:%81.49)* ulaşmıştır. DVM'yi sırasıyla RO *(ACC:%81.26)*, ÇLR *(ACC:%80.14)*, SB *(ACC:%79.68)*, YSA *(ACC:%78.78)* ve KEYK *(ACC:%78.10)* takip etmektedir.

Boyut azaltma işlemi uygulanmayan veri setinin, çapraz geçerlilik veri seti ayırma yöntemi ile eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda ise en yüksek performans değerlerine RO *(ACC:%79.25)* ulaşmıştır. RO'yu sırasıyla DVM *(ACC:%77.94)*, YSA *(ACC:%77.45)*, KEYK *(ACC:%77.08)*, ÇLR *(ACC:%76.59)* ve SB *(ACC:%75.45)* takip etmektedir.

Boyut azaltma işlemi sonucu elde edilen indirgenmiş veri setinin holdout veri seti ayırma yöntemi ile eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda en yüksek kestirim başarısını SB *(ACC:%81.04)* yakalamıştır. SB'yi sırasıyla YSA *(ACC:%80.59)*, RO *(ACC:%80.14)*, KEYK *(ACC:%80.14)*, DVM *(ACC:%78.33)* ve ÇLR *(ACC:%78.33)* takip etmektedir.

Boyut azaltma işlemi sonucu elde edilen indirgenmiş veri setinin çapraz geçerlilikle eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda ise en yüksek kestirim başarısına RO *(ACC:%79.03)* ulaşmıştır. RO'yu sırasıyla, KEYK (ACC:%78.21), YSA *(ACC:%77.58)*, DVM *(ACC:%79.91)*, SB *(ACC:%76.91)* ve ÇLR *(ACC:%76.68)* takip etmektedir.

MS-2 durumunda gerçekleştirilen uygulamalarda modeller %80 üzerinde kestirim başarıları yakalamışlardır. Bağımlı değişkenin bu hali için çok yüksek performans değerlerine ulaşılan uygulamalarda elde edilen kappa istatistikleri incelendiğinde ise 0.42 ile 0.67 arasında değişen değerler gözlemlenmektedir. Bu değerler kestirim modellerinin rasgele gerçekleştirilecek bir kestirimden çok ileride performans sergilediklerini göstermektedir. Aynı zamanda ROC eğrileri altında kalan alanlarda MS-1'e oranla oldukça yüksektir.

Boyut azaltma işleminin, kestirim başarısına olan etkisi incelendiğinde ise bu etkinin bazı sınıflandırma yöntemlerin pozitif bazılarında ise negatif olduğu gözlemlenmiştir. Bu değişimler Çizelge 8.14'te verilmiştir.

Çizelge 8.14. MS-2 için boyut azaltma işleminin kestirim başarısına olan etkisi

	Holdout			**Çapraz Geçerlilik**			**Ortalama**		
	BA-*	**BA+***	**Değişim**	**BA-***	**BA+***	**Değişim**	**BA-***	**BA+***	**Değişim**
ÇLR	**80.14%**	78.33%	-1.81%	76.59%	**76.86%**	0.27%	78.37%	**77.60%**	0.77%
DVM	**81.49%**	78.33%	-3.16%	**77.94%**	76.91%	-1.03%	**79.72%**	77.62%	2.10%
KEYK	78.10%	**80.14%**	2.04%	77.08%	**78.21%**	1.13%	77.59%	**79.18%**	-1.58%
RO	**81.26%**	80.14%	-1.12%	**79.25%**	79.03%	-0.22%	**80.26%**	79.59%	0.67%
SB	79.68%	**81.04%**	1.36%	75.01%	**76.91%**	1.90%	77.35%	**78.98%**	-1.63%
YSA	78.78%	**80.59%**	1.81%	77.45%	**77.58%**	0.13%	78.12%	**79.09%**	-0.97%

BA-: Boyut azaltılmamış veri seti, BA+: Boyut azaltılmış veri seti

8.1.3. MS-3 (G,BM) Durumunda Ulaşılan Bulgular

Maç sonucunun G (ev sahibi kesin kazanır) / BM (deplasman yenilmez) şeklindeki iki sınıf olarak ele alındığı bu durum için gerçekleştirilen uygulamalarda kullanılan yöntemlerin performanslarını en üst düzeye çıkaran parametreler Çizelge 8.15'te paylaşılmıştır.

Çizelge 8.15. MS-3 Model parametreleri

Yöntem	Parametre	İlk Veri Seti	Boyutu Azaltılmış Veri Seti
ÇLR	*Conjugate gradient descent kullanımı*	var	var
	İterasyon Sayısı	150	150
DVM	*c*	1	1
	Çekirdek Fonksiyon	polly kernel	polly kernel
	Kalibratör	Sade bayes	rasgele orman
KEYK	*K*	95	17
	Komşu Arama Algoritması	doğrusal arama	doğrusal arama
	Uzaklık Fonksiyonu	manhatten uzaklığı	manhatten uzaklığı
	Uzaklık Ağırlıklandırma Yöntemi	1/uzaklık	1/uzaklık
RO	*İterasyon Sayısı*	250	100
	Kök Sayısı	1	1
SB	*Çekirdek Tahmin Edici Kullanımı*	var	var
YSA	*Gizli katman (GK) sayısı*	3	3
	GK Nöron Sayıları	9-8-11	9-8-11
	Momentum	0.2	0.2
	Öğrenme oranı	0.3	0.4

Çizelge 8.15'te verilen bu parametrelerin kullanılmasıyla MS-3 durumu için çeşitli eğitim ve test uygulamaları gerçekleştirilmiştir. Gerçekleştirilen bu eğitim ve test uygulamaları sonucu elde edilen bulgulardan, karışıklık matrisleri, Çizelge 8.16'da, karışıklık matrislerinin özeti, Çizelge 8.17'de, performans ölçütleri Çizelge 8.18'de, kestirim başarılarının grafiksel gösterimleri Şekil 8.11 ve Şekil 8.12'de, AUC değerlerine ilişkin bilgiler Çizelge 8.19 ve Çizelge 8.20'de, ROC eğrileri ise Şekil 8.13 ve Şekil 8.14'te paylaşılmıştır.

Çizelge 8.16. MS-3 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri

			ÇLR		DVM		KEYK		RO		SB		YSA	
			0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
İlk Veri Seti	**Holdout**	**0**	195	63	188	70	**199**	59	183	75	191	67	195	63
		1	51	134	43	142	59	126	26	**159**	41	144	51	134
	Çapraz Geç. K=10	**0**	995	303	1003	295	**1019**	279	961	337	981	317	676	322
		1	303	616	290	629	339	580	206	**713**	278	641	268	651
Boyutu Azaltılmış Veri Seti	**Holdout**	**0**	192	66	**215**	43	192	66	182	76	185	73	166	92
		1	49	136	76	109	36	149	26	159	34	151	25	**160**
	Çapraz Geç. K=10	**0**	1025	273	**1127**	171	1015	283	952	346	953	345	997	301
		1	331	588	471	448	295	624	213	**706**	234	685	318	601

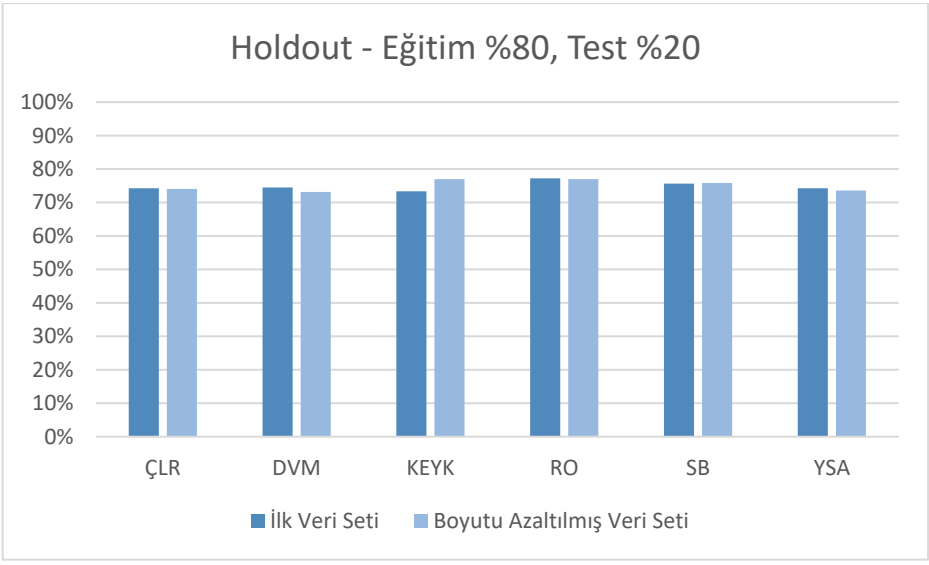
Çizelge 8.16'da paylaşılan karışıklık matrisleri incelendiğinde gerçekleştirilen dört farklı test sonucunda maç sonucu değişkeninin G ve B-M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.17'de gösterilmektedir.

Çizelge 8.17. MS-3 için karışıklık matrislerinin özeti

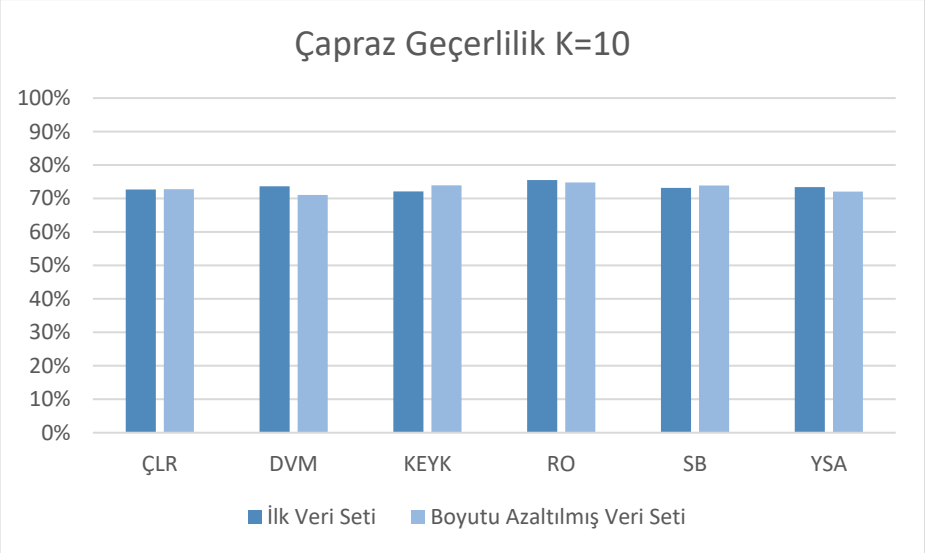
		0 (B-M) Beraberlik ve Mağlubiyet		**1 (G - Galibiyet)**	
		Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
İlk Veri seti	Holdout	KEYK	199/258	RO	159/185
	Çap. Geç. K=10	KEYK	1019/1298	RO	713/919
Boyutu Azaltılmış Veri Seti	Holdout	DVM	215/258	YSA	160/185
	Çap. Geç. K=10	DVM	1127/1298	RO	706/919

Çizelge 8.18. MS-3 için sınıflandırma yöntemleri ile elde edilen karışıklık matrisleri

			İlk Veri Seti						Boyutu Azaltılmış Veri Seti					
	Yöntem	*Sınıf*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*
Eğitim %80 Test %20	ÇLR	0	0.576	0.724	0.793	0.774	0.48	74.27%	0.744	0.735	0.797	0.770	0.47	74.04%
		1	0.724	0.756	0.680	0.702			0.735	0.744	0.673	0.703		
	DVM	0	0.729	0.768	0.814	0.769	0.49	74.49%	0.833	0.589	0.739	0.783	0.43	73.13%
		1	0.768	0.729	0.670	0.715			0.589	0.833	0.717	0.647		
	KEYK	0	0.771	0.681	0.771	0.771	0.45	73.36%	**0.744**	**0.805**	**0.842**	**0.790**	**0.54**	**76.98%**
		1	0.681	0.771	0.681	0.681			**0.805**	**0.744**	**0.693**	**0.745**		
	RO	0	**0.709**	**0.859**	**0.876**	**0.709**	**0.55**	**77.20%**	**0.705**	**0.859**	**0.876**	**0.781**	**0.54**	**76.98%**
		1	**0.859**	**0.709**	**0.679**	**0.759**			**0.859**	**0.705**	**0.677**	**0.757**		
	SB	0	0.740	0.778	0.823	0.780	0.51	75.62%	0.717	0.816	0.845	0.776	0.52	75.85%
		1	0.778	0.740	0.682	0.727			0.816	0.717	0.674	0.738		
	YSA	0	0.756	0.724	0.793	0.774	0.48	74.27%	0.643	0.865	0.869	0.739	0.48	73.59%
		1	0.724	0.756	0.680	0.702			0.865	0.643	0.635	0.732		
K=10 Çapraz Geçerlilik	ÇLR	0	0.767	0.670	0.767	0.767	0.44	72.67%	0.790	0.640	0.756	0.772	0.43	72.76%
		1	0.670	0.767	0.670	0.670			0.640	0.790	0.683	0.661		
	DVM	0	0.773	0.684	0.776	0.774	0.46	73.61%	0.868	0.487	0.705	0.778	0.37	71.04%
		1	0.684	0.773	0.681	0.683			0.487	0.868	0.724	0.487		
	KEYK	0	0.785	0.631	0.750	0.767	0.42	72.12%	0.782	0.679	0.775	0.778	0.46	73.93%
		1	0.631	0.785	0.675	0.652			0.679	0.782	0.688	0.683		
	RO	0	**0.740**	**0.776**	**0.823**	**0.780**	**0.51**	**75.50%**	**0.733**	**0.768**	**0.817**	**0.773**	**0.49**	**74.79%**
		1	**0.776**	**0.740**	**0.679**	**0.724**			**0.768**	**0.733**	**0.671**	**0.716**		
	SB	0	0.756	0.697	0.779	0.756	0.45	73.16%	0.734	0.745	0.803	0.734	0.47	73.88%
		1	0.697	0.756	0.669	0.697			0.745	0.734	0.665	0.703		
	YSA	0	0.752	0.708	0.785	0.768	0.46	73.39%	0.768	0.654	0.758	0.763	0.42	72.08%
		1	0.708	0.752	0.669	0.688			0.654	0.768	0.666	0.654		



Şekil 8.11. MS-3 için holdout veri seti ayırma yönteminde ulaşılan kestirim başarıları



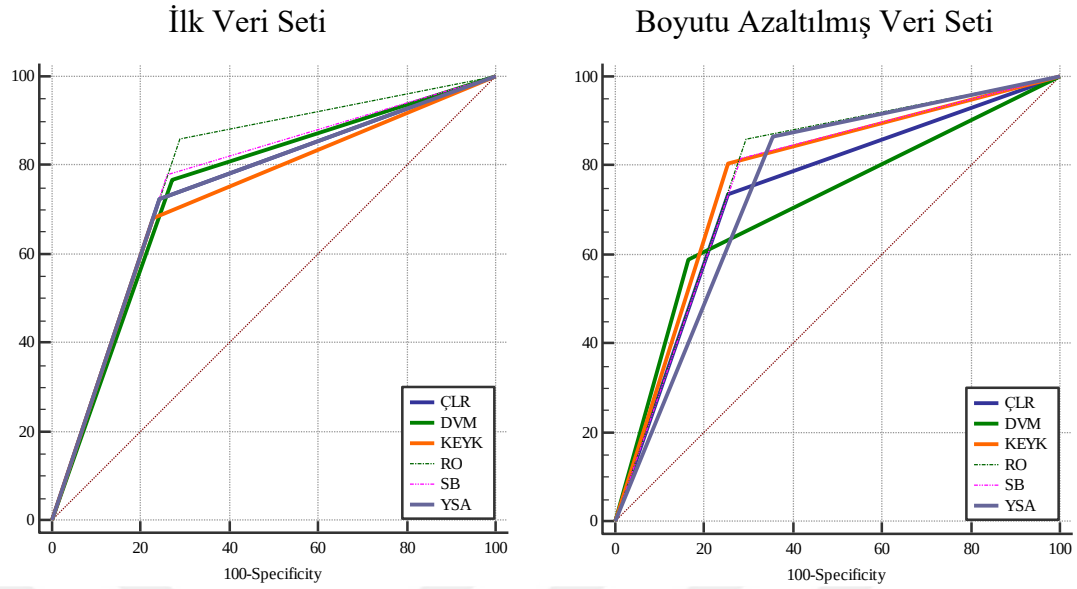
Şekil 8.12. MS-3 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan kestirim başarıları

Çizelge 8.19. MS-3 için holdout veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları

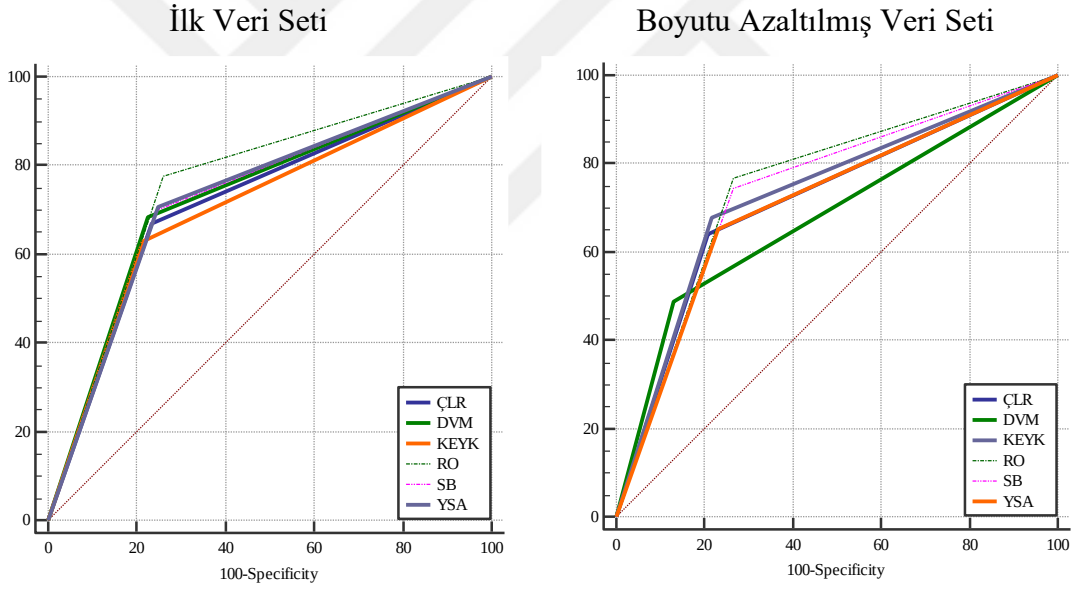
	İlk Veri Seti			Boyutu Azaltılmış Veri Seti		
Yöntem	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**
ÇLR	0.74	0.0212	0.698 - 0.782	0.74	0.0212	0.698 - 0.782
DVM	0.748	0.0208	0.707 - 0.789	0.711	0.0215	0.669 - 0.753
KEYK	0.726	0.0216	0.684 - 0.768	0.775	0.02	0.736 - 0.814
RO	**0.784**	**0.0191**	**0.747 - 0.821**	**0.782**	**0.0191**	**0.745 - 0.819**
SB	0.759	0.0205	0.719 - 0.799	0.767	0.02	0.728 - 0.806
YSA	0.74	0.0212	0.698 - 0.782	0.754	0.0195	0.716 - 0.792

Çizelge 8.20. MS-3 için çapraz geçerlilik veri seti ayırma yönteminde elde edilen AUC ve güven aralıkları

	İlk Veri Seti			Boyutu Azaltılmış Veri Seti		
Yöntem	**AUC**	**Std. Hata**	**95% Güven Aralığı**	**AUC**	**Std. Hata**	**95% Güven Aralığı**
ÇLR	0.718	0.00973	0.699 - 0.737	0.715	0.00974	0.696 - 0.734
DVM	0.729	0.00963	0.71 - 0.748	0.678	0.00949	0.659 - 0.697
KEYK	0.708	0.00979	0.689 - 0.727	0.73	0.0096	0.711 - 0.749
RO	**0.758**	**0.00919**	**0.740 - 0.776**	**0.751**	**0.00928**	**0.733 - 0.769**
SB	0.727	0.00965	0.708 - 0.746	0.74	0.00945	0.721 - 0.759
YSA	0.73	0.0096	0.711 - 0.749	0.711	0.0098	0.692 - 0.73



Şekil 8.13. MS-3 için holdout veri seti ayırma yönteminde ulaşılan ROC eğrileri



Şekil 8.14. MS-3 için çapraz geçerlilik veri seti ayırma yönteminde ulaşılan ROC eğrileri

Bağımlı değişkenin deplasman takımının yenilip yenilmemesine göre oluşturulduğu bu durumda gerçekleştirilen test çalışmalarına ait bulgular şu şekildedir.

Boyut azaltma işlemi uygulanmayan veri setinin holdout veri seti ayırma yöntemi ile eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda en yüksek performans değerlerine RO *(ACC:%77.20)* ulaşmıştır. RO'yu sırasıyla SB *(ACC:%75.62)*, DVM *(ACC:%74.49)*, ÇLR *(ACC:%74.27)*, YSA *(ACC:%74.27)* ve KEYK *(ACC:%73.36)* takip etmektedir.

Boyut azaltma işlemi uygulanmayan veri setinin çapraz geçerlilik veri seti ayırma yöntemi ile eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda ise en yüksek performans değerlerine RO *(ACC:%75.50)* ulaşmıştır. RO'yu sırasıyla DVM *(ACC:%73.61)*, YSA *(ACC:%73.39)*, SB *(ACC:%73.16)*, ÇLR *(ACC:%72.67)* ve KEYK *(ACC:%72.12)* takip etmektedir.

Boyut azaltma işlemi sonucu elde edilmiş indirgenmiş veri setinin holdout veri seti ayırma yöntemi ile eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda en yüksek kestirim başarısına KEYK ve RO ile *(ACC:%76.98)* ulaşılmıştır. Bu yöntemleri sırasıyla SB *(ACC:%75.85)*, ÇLR *(ACC:%74.04)*, YSA *(ACC:%73.59)* ve DVM *(ACC:%73.13)* takip etmektedir.

Boyut azaltma işlemi sonucu elde edimiş indirgenmiş veri setinin, çapraz geçerlilikle eğitim ve test verisi olarak ayrılması sonucu gerçekleştirilen uygulamalarda ise en yüksek kestirim başarısına RO *(ACC:%74.79)* ulaşmıştır. RO'yu sırasıyla KEYK *(ACC:%73.93)*, SB *(ACC:%73.88)*, ÇLR *(ACC:%72.76)*, YSA *(ACC:%72.08)* ve DVM *(ACC:%71.04)* takip etmektedir.

MS-3 durumunda gerçekleştirilen uygulamalarda modeller %71.04 ile %77.20 arasında değişen kestirim başarıları yakalamışlardır. Bu kestirim başarıları MS-1 durumundan çok iyi durumda MS-2 durumunun ise biraz aşağısındadır. Elde edilen kappa istatistikleri incelendiğinde ise 0.42 ile 0.54 arasında değişen değerler gözlemlenmektedir. Bu durum kestirim modellerinin rasgele gerçekleştirilen bir kestirimden ortalama %50 daha iyi kestirim yapabildiğini göstermektedir. ROC eğrileri altında kalan alanlar incelendiğinde ise 0.7 etrafında değişen değerler gözlemlenmektedir.

Boyut azaltma işleminin kestirim başarısına olan etkisi incelendiğinde ise bu etkinin bazı sınıflandırma yöntemlerin pozitif bazılarında ise negatif olduğu gözlemlenmiştir. Bu değişimler Çizelge 8.21'de verilmiştir.

Çizelge 8.21. MS-3 için boyut azaltma işleminin kestirim başarısına olan etkisi

	Holdout			**Çapraz Geçerlilik**			**Ortalama**		
	BA-*	**BA+***	**Değişim**	**BA-***	**BA+***	**Değişim**	**BA-***	**BA+***	**Değişim**
ÇLR	**74.27%**	74.04%	-0.23%	72.67%	**72.76%**	0.09%	73.47%	**73.40%**	0.07%
DVM	**74.49%**	73.13%	-1.36%	**73.61%**	71.04%	-2.57%	**74.05%**	72.09%	1.96%
KEYK	73.36%	**76.98%**	3.62%	72.12%	**73.93%**	1.81%	72.74%	**75.46%**	-2.72%
RO	**77.20%**	76.98%	-0.22%	**75.50%**	74.79%	-0.71%	**76.35%**	75.89%	0.47%
SB	75.62%	**75.85%**	0.23%	73.16%	**73.88%**	0.72%	74.39%	**74.87%**	-0.48%
YSA	**74.27%**	73.59%	-0.68%	**73.39%**	72.08%	-1.31%	73.83%	**72.84%**	0.99%

BA-: Boyut azaltılmamış veri seti, BA+: Boyut azaltılmış veri seti

8.2. KO Kümeleme ile Oluşturulan Hibrit Modellerin Bulguları

K sayısı 2-15 aralığında teker teker artırılarak gözlemlerin hangi kümelerin üyesi olduğunu belirten 14 adet veri sütunu oluşturulmuştur. Elde edilen bu sütunlar bazı yöntemler için veri setinin ilk haline bazı yöntemler için ise boyutu azaltılmış haline eklenmiştir. Ekleme işleminin hangi veri seti üzerine yapılacağına, kullanılan sınıflandırma yöntemlerinin bir önceki başlıkta paylaşılmış olan performanslarına göre karar verilmiştir. Boyutu azaltılmamış veri setinde daha iyi performans gösteren yöntemler için ilk veri seti, boyutu azaltılmış veri setinde daha iyi performans gösteren yöntemler için boyut azaltma işlemi uygulanan veri seti kullanılmıştır. Bu bilgiler Çizelge 8.22'de yer almaktadır. Veri setinin eğitim ve test verisi olarak ayrılması sürecinde ise sadece çapraz geçerlilik yönteminden yararlanılmıştır.

Çizelge 8.22. KO kümeleme ile oluşturulan hibrit modellerde kullanılan veri setleri

	MS-1	MS-2	MS-3
KO-ÇLR	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış
KO-DVM	İlk Veri Seti	İlk Veri Seti	İlk Veri Seti
KO-KEYK	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış
KO-RO	İlk Veri Seti	İlk Veri Seti	İlk Veri Seti
KO-SB	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış
KO-YSA	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış

8.2.1. MS-1 (G,B,M) İçin KO Hibrit Yöntemler

Öncelikli olarak her yöntem için ideal *K* değerinin belirlenebilmesi için deneme yanılma yolu ile testler gerçekleştirilmiştir. Sonrasında ise her yöntem için belirlenen ideal *K* değeri ile ulaşılan bulgular paylaşılmıştır. Elde edilen karışıklık matrisleri Çizelge 8.23'te, bu matrislerin özeti Çizelge 8.24'te, performans değerleri Çizelge 8.25'de ve MS-1 için yapılan ilk uygulamayla oluşan farklar ise Çizelge 8.26'da yer almaktadır.

Çizelge 8.23. MS-1 için KO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri

	KO-ÇLR			**KO-DVM**			**KO-KEYK**			**KO-RO**			**KO-SB**			**KO-YSA**		
	0	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**
0	239	269	80	214	284	90	154	309	125	140	328	120	**274**	219	95	259	204	125
1	84	775	60	70	805	45	45	795	79	34	**830**	55	159	715	45	124	725	70
2	174	143	399	133	184	399	66	169	481	66	164	**486**	184	113	419	138	128	450

Çizelge 8.23'te paylaşılan karışıklık matrisleri incelenerek, maç sonucu değişkeninin G,B ve M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.24'te gösterilmiştir.

Çizelge 8.24. MS-1 için KO kümeleme ile elde edilen karışıklık matrislerinin özeti

0 (B - Beraberlik)		1 (G - Galibiyet)		2 (M - Mağlubiyet)	
Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
KO-SB	274/588	KO-RO	830/919	KO-RO	486/716

Çizelge 8.25. MS-1 için KO hibrit yöntemlerle ulaşılan performans ölçütleri

Yöntem	*K*	*Sınıf*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*
KO-ÇLR	5	0	0.407	0.843	0.485	0.442	0.43	63.66%
		1	0.843	0.682	0.655	0.738		
		2	0.557	0.908	0.736	0.634		
KO-DVM	5	0	0.364	0.877	0.518	0.428	0.43	63.89%
		1	0.876	0.640	0.635	0.736		
		2	0.557	0.911	0.743	0.637		
KO-KEYK	5	0	0.263	0.932	0.585	0.363	0.44	64.33%
		1	0.865	0.632	0.627	0.727		
		2	0.671	0.865	0.696	0.684		
KO-RO	**12**	**0**	**0.237**	**0.938**	**0.583**	**0.337**	**0.45**	**65.46%**
		1	**0.903**	**0.620**	**0.630**	**0.742**		
		2	**0.679**	**0.884**	**0.731**	**0.704**		
KO-SB	12	0	0.466	0.791	0.447	0.456	0.44	63.43%
		1	0.778	0.744	0.686	0.729		
		2	0.586	0.908	0.745	0.656		
KO-YSA	12	0	0.441	0.840	0.500	0.468	0.45	64.56%
		1	0.789	0.744	0.689	0.736		
		2	0.629	0.871	0.693	0.659		

Çizelge 8.26. MS-1 için KO hibrit yöntemlerle kestirim başarısında yaşanan değişim

Yöntem	*Önceki ACC*	*Hibrit ACC*	*Değişim*
KO-ÇLR	60.49%	63.66%	3.19%
KO-DVM	61.66%	63.89%	2.29%
KO-KEYK	61.57%	64.33%	2.76%
KO-RO	**64.05%**	**65.46%**	1.41%
KO-SB	59.63%	63.43%	**3.80%**
KO-YSA	63.35%	64.56%	1.21%

MS-1 için KO kümeleme yöntemlerinden yararlanılarak oluşturulan hibrit modellere ilişkin bulgular incelendiğinde en yüksek başarı oranının %65.46 ile KO-RO yöntemi ile elde edildiği gözlemlenmektedir. Öte yandan MS-1 için sınıflandırma yöntemleri kullanılarak gerçekleştirilen uygulamaların sonuçları ile kıyaslandığında ise en yüksek artışın KO-SB yönteminde yaşandığı görülmektedir. SB yöntemi MS-1 için uygulandığında eriştiği kestirim başarısı %59.63 iken, aynı yöntem KO kümeleme ile hibritlenerek uygulandığında bu oran %3.80'lik artışla %63.43'e yükselmiştir.

8.2.2. MS-2 (GB,M) İçin KO Hibrit Yöntemler

Her yöntem için ideal *K* değerine göre gerçekleştirilen uygulamalar sonucu elde edilen bulgulardan karışıklık matrisleri Çizelge 8.27'de, bu matrislerinin özeti Çizelge 8.28'de, performans değerleri Çizelge 8.29'da ve MS-2 için yapılan ilk uygulamayla oluşan farklar Çizelge 8.30'da paylaşılmıştır.

Çizelge 8.27. MS-2 için KO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri

	KO-ÇLR		**KO-DVM**		**KO-KEYK**		**KO-RO**		**KO-SB**		**KO-YSA**	
	0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
0	389	327	430	286	327	389	384	332	**491**	225	389	327
1	114	1387	129	1372	84	**1417**	94	1407	218	1283	139	1362

Çizelge 8.27'de paylaşılan karışıklık matrisleri incelendiğinde maç sonucu değişkeninin GB ve M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.28'de gösterilmektedir.

Çizelge 8.28. MS-2 için KO kümeleme ile elde edilen karışıklık matrislerinin özeti

0 (M -Mağlubiyet)		**1 (GB – Galibiyet Beraberlik)**	
Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
KO-SB	491/716	KO-KEYK	1471/1501

Çizelge 8.29. MS-2 için KO hibrit yöntemlerle ulaşılan performans ölçütleri

Yöntem	*K*	*Sınıf*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*
KO-ÇLR	3	0	0.543	0.924	0.768	0.636	0.51	80.36%
		1	0.924	0.543	0.814	0.866		
KO-DVM	**3**	**0**	**0.600**	**0.914**	**0.764**	**0.672**	**0.55**	**81.49%**
		1	**0.914**	**0.600**	**0.832**	**0.871**		
KO-KEYK	9	0	0.457	0.944	0.790	0.579	0.45	79.00%
		1	0.944	0.457	0.790	0.860		
KO-RO	3	0	0.536	0.937	0.798	0.641	0.52	81.04%
		1	0.937	0.536	0.814	0.871		
KO-SB	12	0	0.686	0.855	0.686	0.686	0.54	80.14%
		1	0.855	0.686	0.855	0.855		
KO-YSA	5	0	0.543	0.908	0.731	0.623	0.48	79.23%
		1	0.908	0.543	0.811	0.857		

Çizelge 8.30. MS-2 için KO hibrit yöntemlerle kestirim başarısında yaşanan değişim

Yöntem	*Önceki ACC*	*Hibrit ACC*	*Değişim*
KO-ÇLR	76.86%	80.36%	3.50%
KO-DVM	77.94%	**81.49%**	**3.55%**
KO-KEYK	78.21%	79.00%	0.79%
KO-RO	**79.25%**	81.04%	1.79%
KO-SB	76.91%	80.14%	3.23%
KO-YSA	77.58%	79.23%	1.65%

MS-2 için KO kümeleme yöntemleri kullanılarak oluşturulan hibrit modellere ilişkin bulgular incelendiğinde en yüksek başarı oranının %81.49 ile KO-DVM yöntemi ile elde edildiği gözlemlenmektedir. Ayrıca hibrit yöntemler ile elde edilen bu sonuçlar, MS-2 için sınıflandırma yöntemleri kullanılarak elde edilen sonuçlarla karşılaştırıldığında ise en yüksek artışın da yine KO-DVM yönteminde yaşandığı görülmektedir. DVM yöntemi MS-2 için uygulandığında eriştiği kestirim başarısı %77.94 iken, aynı yöntem KO kümeleme ile hibritlenerek uygulandığında bu oran %3.55'lik artışla %81.49'a yükselmiştir.

8.2.3. MS-3 (G,BM) İçin KO Hibrit Yöntemler

Her yöntem için ideal *K* değerine göre gerçekleştirilen uygulamalar sonucu elde edilen bulgulardan karışıklık matrisleri Çizelge 8.31'de, bu matrislerinin özeti Çizelge 8.32'de, performans değerleri Çizelge 8.33'de ve MS-3 için yapılan ilk uygulamayla oluşan farklar Çizelge 8.34'de paylaşılmıştır.

Çizelge 8.31. MS-3 için KO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri

	KO-ÇLR		**KO-DVM**		**KO-KEYK**		**KO-RO**		**KO-SB**		**KO-YSA**	
	0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
0	**986**	312	946	352	971	327	906	392	971	327	976	322
1	253	666	189	730	194	725	129	**790**	199	720	238	681

Çizelge 8.31'de paylaşılan karışıklık matrisleri incelendiğinde maç sonucu değişkeninin G ve BM sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.32'de gösterilmektedir.

Çizelge 8.32. MS-3 için KO kümeleme ile elde edilen karışıklık matrislerinin özeti

0 (MB – Mağlubiyet - Beraberlik)		**1 (G -Galibiyet)**	
Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
KO-ÇLR	989/1298	KO-RO	790/919

Çizelge 8.33. MS-3 için KO hibrit yöntemlerle ulaşılan performans ölçütleri

Yöntem	***K***	***Sınıf***	***TPR***	***TNR***	***PPV***	***F-M***	***KS***	***ACC***
KO-ÇLR	9	0	0.760	0.724	0.794	0.776	0.48	74.50%
		1	0.724	0.760	0.684	0.703		
KO-DVM	9	0	0.729	0.795	0.832	0.777	0.51	75.62%
		1	0.795	0.729	0.677	0.795		
KO-KEYK	**3**	**0**	**0.748**	**0.789**	**0.832**	**0.788**	**0.53**	**76.52%**
		1	**0.789**	**0.748**	**0.692**	**0.737**		
KO-RO	**5**	**0**	**0.698**	**0.859**	**0.874**	**0.776**	**0.54**	**76.52%**
		1	**0.859**	**0.698**	**0.671**	**0.859**		
KO-SB	12	0	0.748	0.784	0.828	0.786	0.52	76.30%
		1	0.784	0.748	0.690	0.734		
KO-YSA	12	0	0.752	0.741	0.802	0.776	0.49	74.72%
		1	0.741	0.752	0.682	0.710		

Çizelge 8.34. MS-3 için KO hibrit yöntemlerle kestirim başarısında yaşanan değişim

Yöntem	*Önceki ACC*	*Hibrit ACC*	*Değişim*
KO-ÇLR	72.76%	74.50%	1.74%
KO-DVM	73.61%	75.62%	2.01%
KO-KEYK	73.93%	**76.52%**	**2.59%**
KO-RO	75.50%	**76.52%**	1.02%
KO-SB	73.88%	76.30%	2.42%
KO-YSA	73.39%	74.72%	1.33%

MS-3 için KO kümeleme yöntemleri ile oluşturulmuş hibrit modellerin bulguları incelendiğinde en yüksek başarı oranının %76.52 olarak KO-KEYK ve KO-RO yöntemleri ile elde edildiği gözlemlenmektedir. Hibrit yöntemler ile elde edilen bulguları, MS-3 için sadece sınıflandırma yöntemleri kullanılarak elde edilmiş bulgularla kıyasladığımızda ise en yüksek artışın KO-KEYK yönteminde gerçekleştiği görülmektedir. KEYK yöntemi MS-3 için uygulandığında eriştiği kestirim başarısı %73.93 iken, aynı yöntem KO kümeleme ile hibritlenerek uygulandığında bu oran %2.59'luk artışla %76.52'ye yükselmiştir.

8.3. BCO Kümeleme ile Oluşturulan Hibrit Modellerin Bulguları

BCO kümeleme yöntemi ile hibrit modeller oluşturulurken öncelikle *C* sayısı 2-15 aralığında teker teker artırılarak 14 farklı kümeleme analizi yapılmıştır. Her kümeleme analizi sonucunda bir tane üyelik bilgilerinin içeren, *C* tane de üyelik derecelerinin yer aldığı toplam *C*+1 adet yeni sütun oluşmuştur. Her defasında kümeleme analizi sonrası oluşan bu yeni sütunlar hem veri setinin ilk haline hem de boyutu azaltılmış haline eklenmiştir. Ardından kullanılan sınıflandırma yöntemleri ile uygulamalar gerçekleştirirken her yöntem için daha iyi performans sergilediği veri seti baz alınmıştır. Hangi yöntem için hangi veri setinin kullanıldığına dair bilgiler Çizelge 3.1'de yer almaktadır. Veri setinin eğitim ve test verisi olarak ayrılması sürecinde ise sadece çapraz geçerlilik yönteminden yararlanılmıştır.

Çizelge 8.35. BCO kümeleme ile oluşturulan hibrit modellerde kullanılan veri setleri

	MS-1	MS-2	MS-3
BCO-ÇLR	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış
BCO-DVM	İlk Veri Seti	İlk Veri Seti	İlk Veri Seti
BCO-KEYK	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış
BCO-RO	İlk Veri Seti	İlk Veri Seti	İlk Veri Seti
BCO-SB	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış
BCO-YSA	Boyut Azaltılmış	Boyut Azaltılmış	Boyut Azaltılmış

8.3.1. MS-1 (G,B,M) İçin BCO Hibrit Yöntemler

Öncelikli olarak her yöntem için ideal *C* değerinin belirlenebilmesi için deneme yanılma yolu ile testler gerçekleştirilmiştir. Sonrasında ise her yöntem için belirlenen ideal *C* değeri ile ulaşılan bulgular paylaşılmıştır. Elde edilen karışıklık matrisleri Çizelge 8.36'da, bu matrislerin özeti Çizelge 8.37'de, performans değerleri Çizelge 8.38'de ve MS-1 için yapılan ilk uygulamayla oluşan farklar ise Çizelge 8.39'da yer almaktadır.

Çizelge 8.36. MS-1 için BCO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri

	BCO-ÇLR			**BCO-DVM**			**BCO-KEYK**			**BCO-RO**			**BCO-SB**			**BCO-YSA**		
	0	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**	**0**	**1**	**2**
0	219	274	95	194	309	85	60	394	135	159	329	100	**279**	214	95	130	299	159
1	94	760	65	50	820	50	20	**830**	70	30	820	70	174	695	50	40	785	94
2	148	153	414	128	199	389	56	189	471	77	164	476	205	113	399	61	148	**506**

Çizelge 8.36'da paylaşılan karışıklık matrisleri incelenerek maç sonucu değişkeninin G,B ve M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.37'de gösterilmiştir.

Çizelge 8.37. MS-1 için BCO kümeleme ile elde edilen karışıklık matrislerinin özeti

0 (B - Beraberlik)		**1 (G - Galibiyet)**		**2 (M - Mağlubiyet)**	
Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
BCO-SB	279/588	BCO-KEYK	830/919	BCO-YSA	506/716

Çizelge 8.38. MS-1 için BCO hibrit yöntemlerle ulaşılan performans ölçütleri

Yöntem	*K*	*Sınıf*	*TPR*	*TNR*	*PPV*	*F-M*	*KS*	*ACC*
BCO-ÇLR	3	0	0.373	0.852	0.478	0.419	0.42	62.76%
		1	0.827	0.671	0.643	0.723		
		2	0.579	0.894	0.717	0.640		
BCO-DVM	5	0	0.331	0.892	0.527	0.406	0.42	63.21%
		1	0.892	0.609	0.620	0.392		
		2	0.543	0.911	0.738	0.626		
BCO-KEYK	5	0	0.144	0.957	0.548	0.228	0.40	62.76%
		1	0.903	0.574	0.603	0.903		
		2	0.671	0.865	0.696	0.684		
BCO-RO	**12**	**0**	**0.271**	**0.935**	**0.604**	**0.374**	**0.45**	**65.46%**
		1	**0.892**	**0.620**	**0.627**	**0.737**		
		2	**0.664**	**0.888**	**0.732**	**0.697**		
BCO-SB	12	0	0.475	0.769	0.427	0.450	0.42	61.86%
		1	0.757	0.748	0.683	0.718		
		2	0.557	0.904	0.729	0.632		
BCO-YSA	12	0	0.220	0.938	0.565	0.317	0.43	63.88%
		1	0.854	0.655	0.640	0.731		
		2	0.707	0.832	0.660	0.683		

Çizelge 8.39. MS-1 için BCO hibrit yöntemlerle kestirim başarısında yaşanan değişim

Yöntem	*Önceki ACC*	*Hibrit ACC*	*Değişim*
BCO-ÇLR	60.49%	62.76%	**2.27%**
BCO-DVM	61.66%	63.21%	1.55%
BCO-KEYK	61.57%	62.76%	1.19%
BCO-RO	**64.05%**	**65.46%**	1.41%
BCO-SB	59.63%	61.86%	2.23%
BCO-YSA	63.35%	63.88%	0.53%

MS-1 için BCO kümeleme yöntemlerinden yararlanılarak oluşturulan hibrit modellere ilişkin bulgular incelendiğinde en yüksek başarı oranının %65.46 ile BCO-RO yöntemi ile elde edildiği gözlemlenmektedir. Öte yandan MS-1 için sınıflandırma yöntemleri kullanılarak gerçekleştirilen uygulamaların sonuçları ile kıyaslandığında ise en yüksek artışın BCO-ÇLR yönteminde yaşandığı görülmektedir. ÇLR yöntemi MS-1 için uygulandığında eriştiği kestirim başarısı %60.49 iken, aynı yöntem BCO kümeleme ile hibritlenerek uygulandığında bu oran %2.27'lik artışla %62.76'ya yükselmiştir.

8.3.2. MS-2 (GB,M) İçin BCO Hibrit Yöntemler

Her yöntem için ideal *C* değerine göre gerçekleştirilen uygulamalar sonucu elde edilen karışıklık matrisleri Çizelge 8.40'da, bu matrislerin özeti Çizelge 8.41'de, performans değerleri Çizelge 8.42'de ve MS-1 için yapılan ilk uygulamayla oluşan farklar ise Çizelge 8.43'te yer almaktadır.

Çizelge 8.40. MS-2 için BCO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri

	BCO-ÇLR		**BCO-DVM**		**BCO-KEYK**		**BCO-RO**		**BCO-SB**		**BCO-YSA**	
	0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
0	368	348	424	292	317	399	399	317	**471**	245	465	251
1	104	1397	119	1382	79	**1422**	104	1397	208	1293	193	1308

Çizelge 8.40'da paylaşılan karışıklık matrisleri incelendiğinde maç sonucu değişkeninin GB ve M sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.41'de gösterilmektedir.

Çizelge 8.41. MS-2 için BCO kümeleme ile elde edilen karışıklık matrislerinin özeti

0 (M -Mağlubiyet)		**1 (GB – Galibiyet Beraberlik)**	
Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
KO-SB	471/716	KO-KEYK	1422/1501

Çizelge 8.42. MS-2 için BCO hibrit yöntemlerle ulaşılan performans ölçütleri

Yöntem	***C***	***Sınıf***	***TPR***	***TNR***	***PPV***	***F-M***	***KS***	***ACC***
BCO-ÇLR	3	0	0.514	0.931	0.774	0.618	0.49	79.91%
		1	0.931	0.514	0.806	0.684		
BCO-DVM	**3**	**0**	**0.593**	**0.921**	**0.776**	**0.672**	**0.55**	**81.76%**
		1	**0.921**	**0.593**	**0.830**	**0.873**		
BCO-KEYK	12	0	0.443	0.947	0.795	0.569	0.44	78.78%
		1	0.947	0.443	0.786	0.859		
BCO-RO	3	0	0.557	0.931	0.788	0.653	0.53	81.26%
		1	0.931	0.557	0.820	0.872		
BCO-SB	3	0	0.657	0.861	0.687	0.682	0.52	79.69%
		1	0.861	0.657	0.845	0.853		
BCO-YSA	3	0	0.650	0.871	0.700	0.674	0.53	80.14%
		1	0.871	0.650	0.843	0.857		

Çizelge 8.43. MS-2 için BCO hibrit yöntemlerle kestirim başarısında yaşanan değişim

Yöntem	*Önceki ACC*	*Hibrit ACC*	*Değişim*
BCO-ÇLR	76.86%	79.91%	3.05%
BCO-DVM	77.94%	**81.76%**	**3.82%**
BCO-KEYK	78.21%	78.78%	0.57%
BCO-RO	**79.25%**	81.26%	2.01%
BCO-SB	76.91%	79.69%	2.78%
BCO-YSA	77.58%	80.14%	2.56%

MS-2 için BCO kümeleme yöntemleri kullanılarak oluşturulan hibrit modellere ilişkin bulgular incelendiğinde en yüksek başarı oranının %81.49 ile BCO-DVM yöntemi ile elde edildiği gözlemlenmektedir. Ayrıca hibrit yöntemler ile elde edilen bu sonuçlar, MS-2 için sınıflandırma yöntemleri kullanılarak elde edilen sonuçlarla karşılaştırıldığında ise en yüksek artışın da yine BCO-DVM yönteminde yaşandığı görülmektedir. DVM yöntemi MS-2 için uygulandığında eriştiği kestirim başarısı %77.94 iken aynı yöntem BCO kümeleme ile hibritlenerek uygulandığında bu oran %3.82'lik artışla %81.76'ya yükselmiştir.

8.3.3. MS-3 (G,BM) İçin BCO Hibrit Yöntemler

Her yöntem için ideal *C* değerine göre gerçekleştirilen uygulamalar sonucu elde edilen karışıklık matrisleri Çizelge 8.44'ee, bu matrislerin özeti Çizelge 8.45'te, performans değerleri Çizelge 8.46'da ve MS-1 için yapılan ilk uygulamayla oluşan farklar ise Çizelge 8.47'de yer almaktadır.

Çizelge 8.44. MS-3 için BCO kümeleme ile oluşturulan hibrit modellere ait karışıklık matrisleri

	BCO-ÇLR		**BCO-DVM**		**BCO-KEYK**		**BCO-RO**		**BCO-SB**		**BCO-YSA**	
	0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
0	986	312	946	352	**1051**	247	931	367	971	327	1001	297
1	258	661	199	720	298	621	124	**795**	204	715	278	641

Çizelge 8.44'de paylaşılan karışıklık matrisleri incelendiğinde maç sonucu değişkeninin G ve BM sınıfları için en yüksek sayıda gözlemi doğru sınıflandıran yöntemler Çizelge 8.45'te gösterilmektedir.

Çizelge 8.45. MS-3 için BCO kümeleme ile elde edilen karışıklık matrislerinin özeti

0 (MB – Mağlubiyet - Beraberlik)		**1 (G -Galibiyet)**	
Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı	Yöntem	Doğru Sınıflandırılan Maç Sonucu Oranı
KO-KEYK	1051/1298	KO-RO	795/919

Çizelge 8.46 MS-3 için BCO hibrit yöntemlerle ulaşılan performans ölçütleri

Yöntem	***C***	***Sınıf***	***TPR***	***TNR***	***PPV***	***F-M***	***KS***	***ACC***
BCO-ÇLR	3	0	0.760	0.719	0.790	0.775	0.48	74.27%
		1	0.719	0.760	0.682	0.700		
BCO-DVM	3	0	0.729	0.784	0.825	0.774	0.5	75.17%
		1	0.784	0.729	0.674	0.725		
BCO-KEYK	3	0	0.810	0.676	0.777	0.793	0.49	75.40%
		1	0.676	0.810	0.718	0.693		
BCO-RO	**3**	**0**	**0.717**	**0.865**	**0.881**	**0.791**	**0.56**	**77.88%**
		1	**0.865**	**0.717**	**0.687**	**0.766**		
BCO-SB	9	0	0.748	0.778	0.825	0.785	0.52	76.07%
		1	0.778	0.748	0.689	0.778		
BCO-YSA	9	0	0.771	0.697	0.780	0.776	0.47	74.04%
		1	0.697	0.771	0.686	0.692		

Çizelge 8.47. MS-3 için BCO hibrit yöntemlerle kestirim başarısında yaşanan değişim

Yöntem	***Önceki ACC***	***Hibrit ACC***	***Değişim***
BCO-ÇLR	72.76%	74.27%	1.51%
BCO-DVM	73.61%	75.17%	1.56%
BCO-KEYK	73.93%	75.40%	1.47%
BCO-RO	**75.50%**	**77.88%**	**2.38%**
BCO-SB	73.88%	76.07%	2.19%
BCO-YSA	73.39%	74.04%	0.65%

MS-3 için BCO kümeleme yöntemleri ile oluşturulmuş hibrit modellerin bulguları incelendiğinde en yüksek başarı oranının %77.88 ile BCO-RO yöntemi ile elde edildiği gözlemlenmektedir. Hibrit yöntemler ile elde edilen bulguları, MS-3 için sadece sınıflandırma yöntemleri kullanılarak elde edilmiş bulgularla kıyasladığımızda ise en yüksek artışın yine BCO-RO yönteminde yaşandığı görülmektedir. RO yöntemi MS-3 için uygulandığında eriştiği kestirim başarısı %75.50 iken, aynı yöntem BCO kümeleme ile hibritlenerek uygulandığında bu oran %2.38'lik artışla %77.88'ye yükselmiştir.

8.4. Ortalama Başarılara ilişkin Bulgular

MS-1, MS-2 ve MS-3 durumları için veri seti içeriğine (boyut indirgeme var-yok) ve eğitim-test verisi ayırma yöntemine (holdout, çapraz geçerlilik) göre farklı uygulamalar gerçekleştirilmiştir. Gerçekleştirilen bu uygulamalarda elde edilen kestirim başarılarının ortalamasının alınmasıyla her yöntemin MS-1, MS-2 ve MS-3 durumlarında elde ettikleri ortalama başarılar hesaplanmıştır. Sonrasında ise bu değerler tablolaştırılarak her bir yöntemin tüm durumları içerir ortalama kestirim başarısına ulaşılmıştır. Sınıflandırma yöntemlerinin direkt olarak kullanılması sonucu ulaşılan sonuçlar Çizelge 8.48'de, K ortalamalar kümeleme algoritması ile birlikte kullanılması sonucu ulaşın sonuçlar Çizelge 8.49'da BCO kümeleme algoritması ile birlikte kullanılması sonucu ulaşılan sonuçlar ise Çizelge 8.50'de paylaşılmıştır.

Çizelge 8.48. Sınıflandırma yöntemleri ile ulaşılan ortalama kestirim başarıları

	MS-1	**MS-2**	**MS-3**	**ORTALAMA**
RO	64.15%	79.92%	76.12%	73.39%
KEYK	61.44%	78.38%	74.10%	71.31%
YSA	61.54%	78.60%	73.33%	71.16%
SB	60.51%	78.16%	74.63%	71.10%
DVM	61.22%	78.67%	73.07%	70.99%
ÇLR	61.13%	77.98%	73.44%	70.85%

Çizelge 8.49. KO kümeleme yöntemi ile hibritlenmiş sınıflandırma yöntemlerinin ortalama kestirim başarıları

	MS-1	**MS-2**	**MS-3**	**ORTALAMA**
KO-RO	65.46%	81.04%	76.52%	74.34%
KO-DVM	63.89%	81.49%	75.62%	73.67%
KO-SB	63.43%	80.14%	76.30%	73.29%
KO-KEYK	64.33%	79.00%	76.52%	73.28%
KO-ÇLR	63.66%	80.36%	74.50%	72.84%
KO-YSA	64.56%	79.23%	74.72%	72.84%

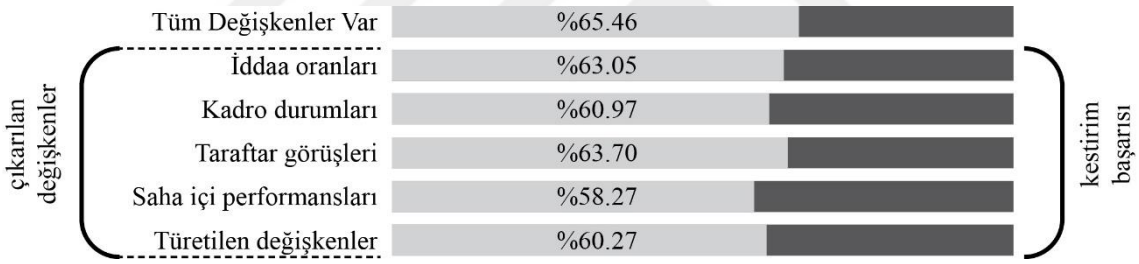
Çizelge 8.50. BCO kümeleme yöntemi ile hibritlenmiş sınıflandırma yöntemlerinin ortalama kestirim başarıları

	MS-1	MS-2	MS-3	ORTALAMA
BCO-RO	65.46%	81.26%	77.88%	74.87%
BCO-DVM	63.21%	81.76%	75.17%	73.38%
BCO-YSA	63.88%	80.14%	74.04%	72.69%
BCO-SB	61.86%	79.69%	76.07%	72.54%
BCO-ÇLR	62.76%	79.91%	74.27%	72.31%
BCO-KEYK	62.76%	78.78%	75.40%	72.31%

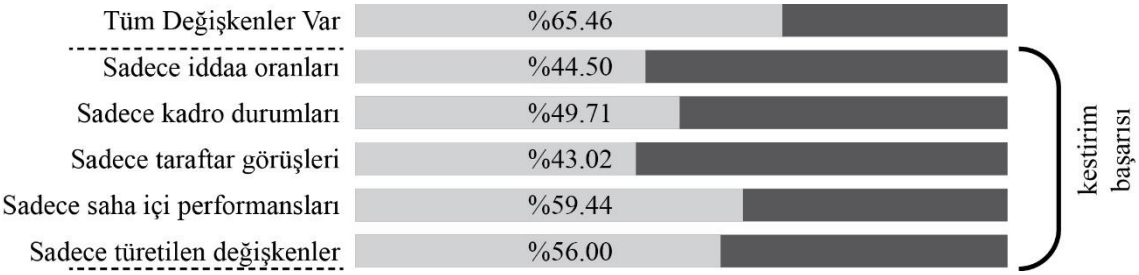
8.5. Veri Setinin İçeriğine İlişkin Bulgular

Adı belirtilen veri seti parçasının modele dahil edilmediği düşünüldüğünde kestirim başarısında yaşanan değişimler ve sadece adı belirtilen veri seti parçası ile oluşturulan modellerin kestirim başarıları;

- MS-1 (G,B,M) için, Şekil 8.15 ve Şekil 8.16'de,

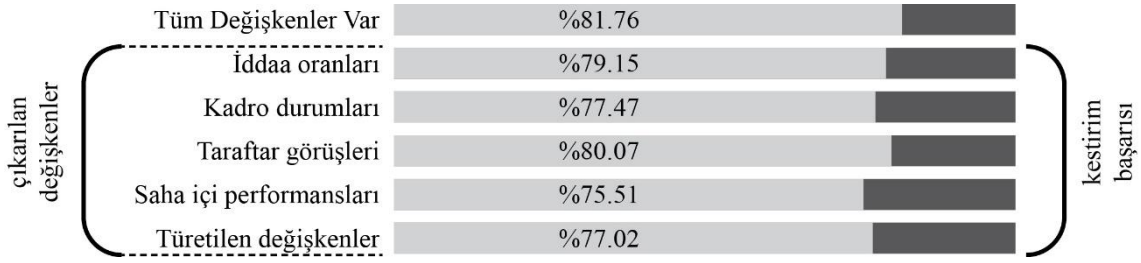


Şekil 8.15. MS-1 için çıkarılan veri seti parçalarının kestirim başarısına etkisi

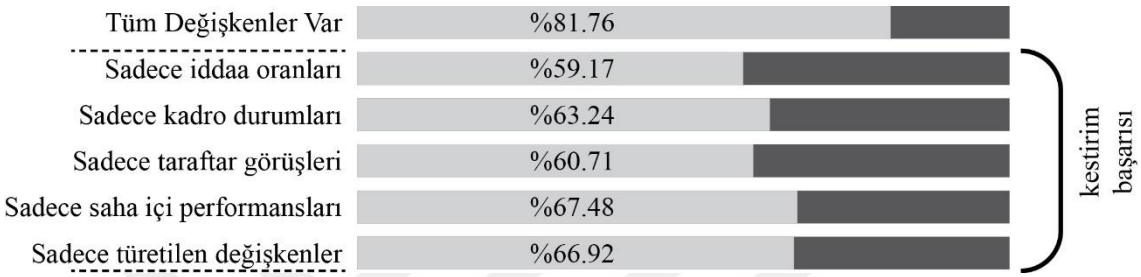


Şekil 8.16. MS-1 için veri setinin her bir parçası ile elde edilen kestirim başarısı

- MS-2 (GB,M) için, Şekil 8.17 ve Şekil 8.18'te,

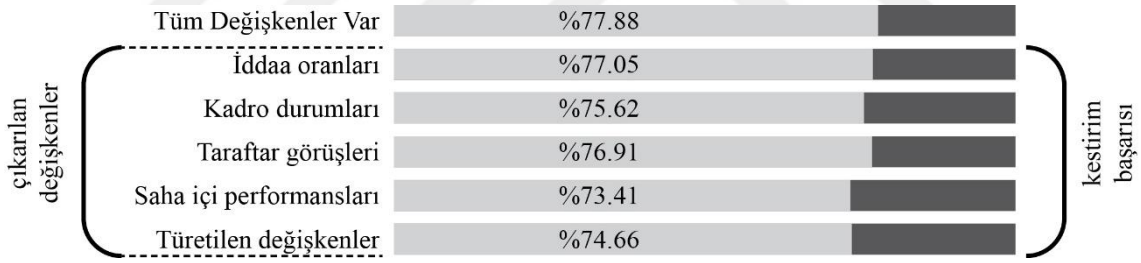


Şekil 8.17. MS-2 için çıkarılan veri seti parçalarının kestirim başarısına etkisi

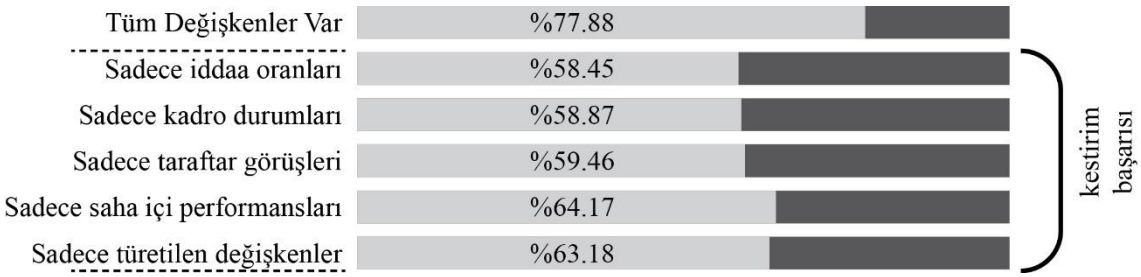


Şekil 8.18. MS-2 için veri setinin her bir parçası ile elde edilen kestirim başarısı

- MS-3 (G,BM) için ise Şekil 8.19 ve Şekil 8.20'de paylaşılmıştır.



Şekil 8.19. MS-3 için çıkarılan veri seti parçalarının kestirim başarısına etkisi



Şekil 8.20. MS-3 için veri setinin her bir parçası ile elde edilen kestirim başarısı

9. SONUÇLAR VE ÖNERİLER

9.1. Sonuçlar

Bu tez çalışması kapsamında futbol müsabakalarının sonuçlarının kestirilmesine yönelik uygulamalar gerçekleştirilmiştir. Bu bağlamda yapay zeka temelli altı sınıflandırma ve iki kümeleme yöntemi kullanılmıştır. Sınıflandırma için çok terimli lojistik regresyon, destek vektör makinesi, k en yakın komşu, rasgele orman, sade bayes ve yapay sinir ağları yöntemleri kullanılırken kümeleme için ise k-ortalamalar ve bulanık c-ortalamalar yöntemlerinden yararlanılmıştır. Sınıflandırma yöntemlerinin performansları karşılaştırılmış sonrasında ise kümeleme yöntemleri de sürece dahil edilerek kestirim başarısını artıran hibrit modeller geliştirilmiştir.

Çalışmanın verilerini, Türkiye, İngiltere, İspanya, İtalya, Almanya ve Fransa'nın en üst düzey liglerinde oynanan toplam 6396 maça dair bilgiler oluşturmaktadır. Bu bilgiler, çalışma kapsamında geliştirilen veri toplama araçları ile dağınık olarak yer aldıkları web tabanlı kaynaklardan toplanarak, oluşturulan veri tabanı tablolarına kaydedilmişlerdir.

Veri toplama ve kayıt sürecinin ardından yöntem performanslarını ölçmek ve veri seti içeriğine göre kestirim performansındaki değişimi izlemek amacıyla alternatif veri setleri oluşturulmuştur. Oluşturulan veri setleri ile öncelikli olarak kullanılan altı farklı sınıflandırma yönteminin başarıları karşılaştırılmış ardından da geliştirilen hibrit yöntemlerin kestirim başarısına olan etkileri araştırılmıştır.

Çalışma boyunca elde edilen sonuçlar, sınıflandırma yöntemlerine, hibrit yöntemlere ve veri seti içeriğine göre farklı başlıklar altında ela alınmış son başlık olarak da çalışmanın genel sonuçları paylaşılmıştır.

Sınıflandırma Yöntemlerine İlişkin Sonuçlar

Yapay zeka tabanlı sınıflandırma yöntemleri futbol müsabakalarının sonuçlarının kestirilmesinde son derece başarılı olmuşlardır. Elde edilen kestirim başarıları, futbol ve bahis istatistikleri alanında çalışmalar gerçekleştiren uzmanlarca oldukça yüksek olarak değerlendirilmiştir.

Sınıflandırma yöntemleri ile elde edilen bulgular bağımlı değişkenin farklı şekillerde ele alındığı MS1, MS2 ve MS3 durumları için aşağıda özetlenmiştir. Paylaşılan performans değerleri veri setinin çapraz geçerlilikle 10 parçaya ayrılması sonucu elde edilen eğitim ve test kümeleri ile gerçekleştirilen uygulamalarda elde edilen değerlerdir.

- MS-1 (G,B,M) için,
    - Ortalama olarak en başarılı yöntem RO'dur. *(ACC: %64.05)*
    - Galibiyet en iyi kestiren yöntem DVM'dir. *(ACC: %89.60)*
    - Beraberliği en iyi kestiren yöntem SB'dir. *(ACC: %39.90)*
    - Mağlubiyeti en iyi kestiren yöntem RO'dur. *(ACC: %69.00)*
- MS-2 (GB,M) için,
    - Ortalama olarak en başarılı yöntem RO'dir. *(ACC: %79.25)*
    - GB'yi iyi kestiren yöntem KEYK'dir. *(ACC: %90.13)*
    - M'yi en iyi kestiren yöntem SB'dir. *(ACC: %70.81)*
- MS-3 (G,BM) için,
    - Ortalama olarak en başarılı yöntem RO'dur. *(ACC: %75.05)*
    - G'yi iyi kestiren yöntem RO'dur. *(ACC: %77.58)*
    - BM'yi en iyi kestiren yöntem DVM'dir. *(ACC: %86.80)*
- DVM ve RO yöntemleri, boyut indirgeme çalışmaları yapılmamış veri seti ile daha yüksek performans göstermişlerdir.
- ÇLR, KEYK, SB, YSA yöntemleri, boyutu indirgenmiş veri setinde daha iyi performans göstermişlerdir.

Geliştirilen Hibrit Yöntemlere İlişkin Sonuçlar

- MS-1 (G,B,M) için,
    - KO hibrit yöntemleri içerisinde en iyi kestirim oranını yakalayan yöntem KO-RO'dur. *(ACC: %65.46)*
    - KO hibrit yöntemleri içerisinde hibrit olmayan yöntemlere oranla en yüksek artış KO-SB yönteminde yaşanmıştır. *Değişim: %3.80 (%59.63 – %63.43)*
    - BCO hibrit yöntemleri içerisinde en iyi kestirim oranını yakalayan yöntem BCO-RO'dur. *(ACC: %65.46)*
    - BCO hibrit yöntemleri içerisinde hibrit olmayan yöntemlere oranla en yüksek artış BCO-ÇLR yönteminde yaşanmıştır. *Değişim: %2.27 (%60.49 – %62.76)*
- MS-2 (GB,M) için,
    - KO hibrit yöntemleri içerisinde en iyi kestirim oranını yakalayan yöntem KO-RO'dur. *ACC: %81.04*

  - KO hibrit yöntemleri içerisinde hibrit olmayan yöntemlere oranla en yüksek artış KO-DVM yönteminde yaşanmıştır. *Değişim: %3.55 (%77.94 – %81.49)*
  - BCO hibrit yöntemleri içerisinde en iyi kestirim oranını yakalayan yöntem BCO-DVM'dir. *(ACC: %81.76)*
  - BCO Hibrit yöntemleri içerisinde hibrit olmayan yöntemlere oranla en yüksek artış BCO-DVM yönteminde yaşanmıştır. *Değişim: %3.82 (%77.94 – %81.76)*
- MS-3 (G,BM) için,
  - KO hibrit yöntemleri içerisinde en iyi kestirim oranını yakalayan yöntemler KO-KEYK ve KO-RO'dur. *(ACC: %76.52)*
  - KO hibrit yöntemleri içerisinde hibrit olmayan yöntemlere oranla en yüksek artış KO-KEYK yönteminde yaşanmıştır. *Değişim: %2.59 (%73.93 – %76.52)*
  - BCO hibrit yöntemleri içerisinde en iyi kestirim oranını yakalayan yöntem BCO-RO'dur. *(ACC: %77.88)*
  - BCO hibrit yöntemleri içerisinde hibrit olmayan yöntemlere oranla en yüksek artış BCO-RO yönteminde yaşanmıştır. *Değişim: %2.38 (%75.50 – %77.88)*

Veri Seti İçeriğine ve Veri Setinin Oluşturulma Sürecine İlişkin Sonuçlar

- Veri seti parçaları içerisinde kestirim başarısına en önemli katkıyı sunan veriler takımların geçmiş maçlarından elde ettikleri ortalama saha içi performans değerleridir.
- Takımların geçmiş maçlardan elde ettikleri ortalama saha içi performans değerleri hesaplanırken belli sayıda maç dikkate alınmaktadır. Dikkate alınan maç sayısının kestirim başarısına etkisini araştırmak adına çeşitli denemeler yapılmış ve bu denemeler sonucunda en yüksek kestirim başarısına son 6 maçın dikkate alınması sonucu oluşturulan verilerle ulaşıldığı gözlemlenmiştir.
- Her bir takım için ayrı ayrı hesaplanan ortalama saha içi performans değerlerinin içeren [1x65] boyutlu iki vektörün kıyaslanarak [1x65] boyutlu yeni bir vektöre dönüştürülmesi sürecinde "oranlar", "farklar" ve "1-2 kodlama" seçenekleri kullanılmıştır. Gerçekleştirilen denemelerde,

kestirim başarısının en yüksek düzeye "oranlar" seçeneği ile yapılan kıyaslamalar sonucunda çıktığı gözlemlenmiştir.

- Veri seti içerisinde kestirim başarısına en düşük katkıyı sağlayan parçalar taraftar görüşleri ve iddaa oranlarıdır.

Genel Sonuçlar

- En yüksek kestirim başarısına galibiyetle biten maçlarda ulaşılmaktadır.
- En düşük kestirim başarısı beraberlikle biten maçlarda ortaya çıkmaktadır.
- Tüm durumlar için sınıflandırma algoritmaları arasında en başarılı olanı RO'dur. (*ACC: 73.39%)*
- Tüm durumlar için KO kümeleme yöntemi ile hibritlenmiş sınıflandırıcılar içerisinde en başarılı olanı KO-RO'dur. *(ACC: 74.34%)*
- Tüm durumlar için BCO kümeleme yöntemi ile hibritlenmiş sınıflandırıcılar içerisinde en başarılı olanı BCO-RO'dur. *(ACC: 74.87%)*
- MS-1 için en başarılı yöntem KO-RO ve BCO-RO'dur. *(ACC: %65.46)*
- MS-2 için en başarılı yöntem BCO-DVM'dir. *(ACC: %81.76)*
- MS-3 için en başarılı yöntem BCO-RO'dur. *(ACC: %77.88)*
- Geliştirilen hibrit yöntemler kestirim başarısını artırmışlardır.
- BCO kümeleme yöntemi ile oluşturulan hibrit modeller KO kümeleme yöntemi ile oluşturulan modellere oranla daha yüksek kestirim başarısına erişmişlerdir.
- Geliştirilen hibrit modellerde kestirim başarısında meydana gelen artış farklı hibrit çözümler aranabileceğini göstermiştir.
- Kullanılan sınıflandırma yöntemlerinin modeli eğitmek için harcadıkları sürelere bakıldığından en hızlı yöntem KEYK, en yavaş yöntem YSA'dır.
- Futbol müsabakalarına ilişkin istatistiksel verilere çok sayıda internet sitesinden erişilebilmekte fakat bu kaynakların akredite olmaları gereken bir üst organizasyon olmadığı için özellikle küçük çaplı sitelerde sağlıksız veriler yer almaktadır.
- Futbol maçlarından çıkarılan istatistiklerin belli bir kısmı maçı gözlemleyen kişilerce raporlaştırıldığı için farklı platformlarda paylaşılan istatistiksel bulgularda tutarsızlık olabilmektedir.

- Futbol istatistiklerini paylaşan sitelerden elektronik tablolar halinde veri elde edilememektedir ve bu sitelerin bazıları çeşitli güvenlik uygulamaları ile web scrapig araçlarını engellemektedirler. Bu nedenle bazı verileri elde etme süreci çok fazla zaman alabilmektedir.
- Futbola dair en çok sayıda istatistiksel paylaşımının yapıldığı platformlar online iddaa siteleridir.
- Futbolda saha içi istatistiklerle ilgilenen akademik çalışmaların büyük çoğunluğu iddaa temelli çalışmalardır. Literatürde iddaa oranlarının belirlenmesi, karlılığın artırılması vb. amaçlarla gerçekleştirilmiş çok sayıda çalışma mevcuttur.
- FIFA, UEFA ve TFF gibi organizasyonların bilgi bankalarında futbol maçlarına dair detaylı istatistikler yer almamaktadır.
- Futbol istatistikleri alanında çalışmalar gerçekleştiren ve internet üzerindeki açık erişimli verilerden çok daha fazlasını ticari olarak pazarlayan organizasyonlar mevcuttur.
- Futbol istatistikleri ile ilgili en fazla sayıdaki akademik çalışma İngiltere Premier Ligi baz alınarak gerçekleştirilmiştir.

9.2. Öneriler ve Sonraki Çalışmalar

Çalışma sürecinde edinilen deneyimler doğrultusunda yapılacak olan önerileri şu şekilde sıralayabiliriz.

Veri seti içeriği genişletilerek performans değerlerinin artırılmasına yönelik öneriler:

- İnternette erişime açık olarak yer alan verilerin dışında ticari olarak belli organizasyonlarca pazarlanan ve daha detaylı bilgiler içeren verilere ulaşılarak, bu verilerden elde edilecek belli sayıdaki değişken veri setine eklenebilir.
- Takım kadrolarında yer alan oyuncuların o dönemki bireysel performansları hakkında detaylı bilgiler verebilecek değişkenler veri setine eklenebilir.
- Futbolcuların o maç öncesindeki belli bir aralık için antrenman verilerine ulaşılıp bu verilerden elde edilecek bilgiler çalışma veri setine eklenebilir.
- Belli kriterlere göre futbolcuların güçleri hesaplanıp bu değerler ile ortalama takım güçleri elde edilerek veri setine eklenebilir.

- Rakip takımdan bağımsız şekilde her bir takımın eldeki mevcut verilerinden yola çıkarak derin öğrenme yöntemleri ile gelecek performansları öngörülebilir ve ulaşılan değerler çalışmanın veri setine eklenebilir.
- Oynanacak olan müsabakanın sonucuna etkisi olabilecek ama istatistiksel olarak ölçülemeyen çeşitli değişkenler hesaba katılabilir. Örneğin maç öncesi yaşanmış bir teknik direktör değişikliği, müsabaka primi miktarı, futbolcu veya teknik adamların duygu durumları vb. değişkenler sayısallaştırılarak veri setine eklenebilir.
- Maçın oynanacağı yer, tarih, saat, coğrafi koşullar vb. bilgiler veri setine eklenebilir.
- Berabere biten maçların verileri incelenerek yapılacak çıkarımlar sonucu veri setine modellerin berabere biten maçları daha iyi kestirebilmelerini sağlayacak yeni değişkenler eklenebilir.

Kullanılan yöntemlere ilişkin öneriler:

- Farklı sınıflandırma ve kümeleme algoritmalarının kestirim başarısı üzerindeki etkisi araştırılabilir.
- Kestirim başarısını yukarı çekecek yeni sınıflandırma yöntemlerinin geliştirilmesi yönelik araştırmalar gerçekleştirilebilir.
- Literatürdeki mevcut sınıflandırma ve kümeleme yöntemleri ile yeni hibrit çözümler oluşturmaya yönelik çalışmalar yapılabilir.
- Futbolcu ve takımların güçlerinin hesaplanması durumunda bu işlemin gerçekleştirilebilmesi adına çok kriterli karar verme tekniklerinden yararlanılabilir.
- Beraberliği en iyi kestiren yöntem olan sade bayes sınıflandırıcının ekseninde hibrit bir model geliştirilebilir.

Bu çalışma boyunca edinilen deneyimlerin de katkısıyla bundan sonraki süreçte çalışma konusu olarak belli başlıklar belirlenmiştir. Bunlar;

- Futbolcuların piyasa değerlerinin belirlenmesi amacıyla çalışmalar gerçekleştirilecektir.
- Futbolcu performanslarını değerlendirecek yapay zeka tabanlı bir uygulama ile en iyi kadronun kurulmasına yönelik çalışmalar gerçekleştirilecektir.
- Maç sonucunun yanı sıra takımların attıkları gol sayısına odaklanarak maç skorunun öngörülmesine yönelik çalışmalar yapılacaktır.

- Futbolcuların antrenman verileri ve performans değerleri elde edilerek bu verilerden istatistiksel açıdan anlamlı sonuçlar çıkarılmasına yönelik çalışmalar gerçekleştirilecektir.
- Çalışma kapsamında kullanılan sınıflandırma ve kümeleme yöntemlerinin performanslarının iyileştirilmesine yönelik modifikasyonlar ve hibrit çözümler araştırılacaktır.
- Geliştirilen hibrit yöntemlerinin performansları farklı problem durumları ve farklı veri setlerinde ele alınacaktır.
- Yeni sınıflandırma ve kümeleme yöntemlerinin geliştirilmesi üzerine çalışmalar gerçekleştirilecektir.

KAYNAKLAR

Abbas, O. A., 2008, Comparisons between data clustering algorithms, *International Arab Journal of Information Technology (IAJIT)*, 5 (3), 320-325.

Abbasi, A. A. ve Younis, M., 2007, A survey on clustering algorithms for wireless sensor networks, *Computer communications*, 30 (14-15), 2826-2841.

Acconci, C., 2008, Whoscored.Com, https://tr.whoscored.com/AboutUs:

Acock, A. C., 2005, Working with missing values, *Journal of Marriage and family*, 67 (4), 1012-1028.

Adeniyi, D. A., Wei, Z. ve Yongquan, Y., 2016, Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method, *Applied Computing and Informatics*, 12 (1), 90-108.

Aggarwal, C. C. ve Yu, P. S., 2001, Outlier detection for high dimensional data, *ACM Sigmod Record*, 30 (2), 37-46.

Aggarwal, C. C., 2014, Data classification: algorithms and applications, CRC Press.

Akdeniz, F., 2015, İstatistikte Yeni Eğilimler ve Gelişmeler, *Sosyal Bilimler Araştırma Dergisi*, 4 (4), 1-11.

Alpaydın, E., 2017, Yapay öğrenme, *İstanbul*, Boğaziçi Üniversitesi Yayınevi.

Aly, M., 2005, Survey on multiclass classification methods, *Neural Netw*, 19, 1-9.

Amadin, F. ve Obi, J. C., 2015, English Premier League (EPL) Soccer Matches Prediction using An Adaptive Neuro-Fuzzy Inference System (ANFIS), *Transactions on Machine Learning and Artificial Intelligence*, 3 (2), 34.

Amirkhah, R., Farazmand, A., Gupta, S. K., Ahmadi, H., Wolkenhauer, O. ve Schmitz, U., 2015, Naive Bayes classifier predicts functional microRNA target interactions in colorectal cancer, *Molecular bioSystems*, 11 (8), 2126-2134.

Amor, N. B., Benferhat, S. ve Elouedi, Z., 2004, Naive bayes vs decision trees in intrusion detection systems, *Proceedings of the 2004 ACM symposium on Applied computing*, 420-424.

Anyanwu, M. N. ve Shiva, S. G., 2009, Comparative analysis of serial decision tree classification algorithms, *International Journal of Computer Science and Security*, 3 (3), 230-240.

Aslan, B. G. ve Inceoglu, M. M., 2007, A comparative study on neural network based soccer result prediction, *Intelligent Systems Design and Applications, 2007. ISDA 2007. Seventh International Conference on*, 545-550.

Avidan, S., 2004, Support vector tracking, *IEEE Transactions on pattern analysis and machine intelligence*, 26 (8), 1064-1072.

Azar, A. T., Elshazly, H. I., Hassanien, A. E. ve Elkorany, A. M., 2014, A random forest classifier for lymph diseases, *Computer Methods and Programs in Biomedicine*, 113 (2), 465-473.

Bailey, T. ve Jain, A., 1978, A note on distance-weighted $ k $-nearest neighbor rules, *IEEE Transactions on Systems, Man, and Cybernetics* (4), 311-313.

Barto, A. G., 1997, Reinforcement learning, In: Neural systems for control, Eds: Elsevier.

Barto, M. ve Rosenstein, M. T., 2004, J. 4 supervised actor-critic reinforcement learning, *Handbook of learning and approximate dynamic programming*, 2, 359-360.

Bayaga, A., 2010, Multinomial logistic regression: usage and application in risk analysis, *Journal of applied quantitative methods*, 5 (2).

Ben-Gal, I., 2005, Outlier detection, In: Data mining and knowledge discovery handbook, Eds: Springer.

Bermejo, S. ve Cabestany, J., 2000, Adaptive soft k-nearest-neighbour classifiers, *Pattern Recognition*, 33 (12), 1999-2005.

Bezdek, J. C., 1973, Fuzzy mathematics in pattern classification, *Ph. D. Dissertation, Applied Mathematics, Cornell University*.

Bezdek, J. C., Coray, C., Gunderson, R. ve Watson, J., 1981, Detection and characterization of cluster substructure i. linear structure: Fuzzy c-lines, *SIAM Journal on Applied Mathematics*, 40 (2), 339-357.

Bezdek, J. C., 2013, Pattern recognition with fuzzy objective function algorithms, Springer Science & Business Media.

Bhanja, S. ve Das, A., 2018, Impact of Data Normalization on Deep Neural Network for Time Series Forecasting, *arXiv preprint arXiv:1812.05519*.

Binet, A. ve Simon, T., 1916, The development of intelligence in children: The Binet-Simon Scale, 11, Williams & Wilkins Company.

Bolstad, B. M., Irizarry, R. A., Åstrand, M. ve Speed, T. P., 2003, A comparison of normalization methods for high density oligonucleotide array data based on variance and bias, *Bioinformatics*, 19 (2), 185-193.

Boniface, P., 2007, Futbol ve küreselleşme, NTV Yayınları.

Boostani, R., Sadatnezhad, K. ve Sabeti, M., 2009, An efficient classifier to diagnose of schizophrenia based on the EEG signals, *Expert Systems with Applications*, 36 (3), 6492-6499.

Bora, D. J., Gupta, D. ve Kumar, A., 2014, A comparative study between fuzzy clustering algorithm and hard clustering algorithm, *arXiv preprint arXiv:1404.6059*.

Borra, S. ve Di Ciaccio, A., 2010, Measuring the prediction error. A comparison of cross-validation, bootstrap and covariance penalty methods, *Computational Statistics & Data Analysis*, 54 (12), 2976-2989.

Bousquet, O., Boucheron, S. ve Lugosi, G., 2003, Introduction to statistical learning theory, *Summer School on Machine Learning*, 169-207.

Breiman, L., 1996, Bagging predictors, *Machine Learning*, 24 (2), 123-140.

Breiman, L., 2001, Random Forests, *Machine Learning*, 45 (1), 5-32.

Bromberger, C., 1998, Football: la bagatelle la plus sérieuse du monde, Bayard París.

Buhmann, J., 1998, Empirical risk approximation: An induction principle for unsupervised learning, Citeseer.

Bullard, J. H., Purdom, E., Hansen, K. D. ve Dudoit, S., 2010, Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments, *Bmc Bioinformatics*, 11 (1), 94.

Bunker, R. P. ve Thabtah, F., 2017, A machine learning framework for sport result prediction, *Applied Computing and Informatics*, 15 (1), 27-33.

Cai, W., Chen, S. ve Zhang, D., 2007, Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation, *Pattern Recognition*, 40 (3), 825-838.

Campbell, W. M., Sturim, D. E. ve Reynolds, D. A., 2006, Support vector machines using GMM supervectors for speaker verification, *IEEE signal processing letters*, 13 (5), 308-311.

Candès, E. J., Li, X., Ma, Y. ve Wright, J., 2011, Robust principal component analysis?, *Journal of the ACM (JACM)*, 58 (3), 11.

Cannady, J., 1998, Artificial neural networks for misuse detection, *21st National Information Systems Security Conference*, Arlington, Virginia, United States.

Caraffa, A., Cerulli, G., Projetti, M., Aisa, G. ve Rizzo, A., 1996, Prevention of anterior cruciate ligament injuries in soccer, *Knee Surgery, Sports Traumatology, Arthroscopy*, 4 (1), 19-21.

Cerrada, M. ve Aguilar, J., 2008, Reinforcement learning in system identification, In: Reinforcement Learning, Eds: IntechOpen.

Cerrada, M., Zurita, G., Cabrera, D., Sánchez, R.-V., Artés, M. ve Li, C., 2016, Fault diagnosis in spur gears based on genetic algorithm and random forest, *Mechanical Systems and Signal Processing*, 70, 87-103.

Chapelle, O., Schölkopf, B. ve Zien, A., 2010, Semi-supervised Learning, MIT Press.

Charniak, E., 1985, Introduction to artificial intelligence, Pearson Education India.

Che Mohamad Firdaus Che Mohd, R., Mohd Zainuri, S., Nazim, R. ve Aida, M., 2018, A Comparative Study of Data Mining Techniques on Football Match Prediction, *Journal of Physics: Conference Series*, 1020 (1).

Chen, M.-S., Han, J. ve Yu, P. S., 1996, Data mining: an overview from a database perspective, *Ieee Transactions on Knowledge and Data Engineering*, 8 (6), 866-883.

Chen, W., Xie, X., Wang, J., Pradhan, B., Hong, H., Bui, D. T., Duan, Z. ve Ma, J., 2017, A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility, *Catena*, 151, 147-160.

Chen, X., Li, Y., Harrison, R. ve Zhang, Y.-Q., 2008, Type-2 fuzzy logic-based classifier fusion for support vector machines, *Applied Soft Computing*, 8 (3), 1222-1231.

Cheng, T., Cui, D., Fan, Z., Zhou, J. ve Lu, S., 2003, A new model to forecast the results of matches based on hybrid neural networks in the soccer rating system, *Computational Intelligence and Multimedia Applications, 2003. ICCIMA 2003. Proceedings. Fifth International Conference on*, 308-313.

Cherny, A. ve Madan, D., 2008, New measures for performance evaluation, *The Review of Financial Studies*, 22 (7), 2571-2606.

Chuang, K.-S., Tzeng, H.-L., Chen, S., Wu, J. ve Chen, T.-J., 2006, Fuzzy c-means clustering with spatial information for image segmentation, *computerized medical imaging and graphics*, 30 (1), 9-15.

Civco, D. L., 1993, Artificial neural networks for land-cover classification and mapping, *International journal of geographical information science*, 7 (2), 173-186.

Constantinou, A. C., Fenton, N. E. ve Neil, M., 2012, pi-football: A Bayesian network model for forecasting Association Football match outcomes, *Knowledge-Based Systems*, 36, 322-339.

Cooke, D. ve Clarke, G., 1989, A basic course in statistics, Arnold.

Corchado, J. ve Lees, B., 1998, Artificial neural networks in pattern recognition: multicollinearity and heterocedasticity, *Colloquium on Knowledge Discovery*, London / UK.

Costa, E., Lorena, A., Carvalho, A. ve Freitas, A., 2007, A review of performance evaluation measures for hierarchical classifiers, *Evaluation Methods for Machine Learning II: papers from the AAAI-2007 Workshop*, 1-6.

Cover, T. M. ve Hart, P. E., 1967, Nearest neighbor pattern classification, *IEEE transactions on information theory*, 13 (1), 21-27.

Cramer, J. S., 2002, The origins of logistic regression. Tinbergen Institute Discussion Paper, No. 02-119/4. Amsterdam and Rotterdam, Tinbergen Institute.

Cristianini, N. ve Shawe-Taylor, J., 2000, An introduction to support vector machines and other kernel-based learning methods, Cambridge university press.

Crowder, M., Dixon, M., Ledford, A. ve Robinson, M., 2002, Dynamic modelling and prediction of English Football League matches for betting, *Journal of the Royal Statistical Society: Series D (The Statistician)*, 51 (2), 157-168.

Dash, P., Samantaray, S. ve Panda, G., 2006, Fault classification and section identification of an advanced series-compensated transmission line using support vector machine, *IEEE transactions on power delivery*, 22 (1), 67-73.

Dawson, C. ve Wilby, R., 2001, Hydrological modelling using artificial neural networks, *Progress in physical Geography*, 25 (1), 80-108.

Decrop, A. ve Derbaix, C., 2010, Pride in contemporary sport consumption: a marketing perspective, *Journal of the Academy of Marketing Science*, 38 (5), 586-603.

Dey, L., Chakraborty, S., Biswas, A., Bose, B. ve Tiwari, S., 2016, Sentiment Analysis of Review Datasets Using Naive Bayes and K-NN Classifier, *arXiv preprint arXiv:1610.09982*.

Di Martino, F. ve Sessa, S., 2009, Implementation of the extended fuzzy C-means algorithm in geographic information systems, *Journal of Uncertain Systems*, 3 (4), 298-306.

Dibike, Y. B., Velickov, S., Solomatine, D. ve Abbott, M. B., 2001, Model induction with support vector machines: introduction and applications, *Journal of Computing in Civil Engineering*, 15 (3), 208-216.

Dixon, M. J. ve Coles, S. G., 1997, Modelling association football scores and inefficiencies in the football betting market, *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 46 (2), 265-280.

Dobson, S., Goddard, J. A. ve Dobson, S., 2001, The economics of football, Cambridge University Press Cambridge.

Domingos, P. ve Pazzani, M., 1997, On the optimality of the simple Bayesian classifier under zero-one loss, *Machine Learning*, 29 (2-3), 103-130.

Dorsey, J., Glass, N., Williams, E. ve Stone, B., 2006, Twitter, https://about.twitter.com/tr.html:

Dudani, S., 1978, The distance-weighted k-nearest neighbor rule, *IEEE trans. on systems, man and cybernetics*, 8 (4), 311-313.

Dunn, J. C., 1973, A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, *Journal of Cybernetics*, 3 (3), 32-57.

Dutta, A., Bandopadhyay, G. ve Sengupta, S., 2015, Prediction of stock performance in indian stock market using logistic regression, *International Journal of Business and Information*, 7 (1), 105-136.

El Akadi, A., Amine, A., El Ouardighi, A. ve Aboutajdine, D., 2011, A two-stage gene selection scheme utilizing MRMR filter and GA wrapper, *Knowledge and Information Systems*, 26 (3), 487-500.

Elmas, Ç., 2010, Yapay zeka uygulamaları:(yapay sinir ağı, bulanık mantık, genetik algoritma), Seçkin Yayıncılık.

Elo, A. E., 1978, The rating of chessplayers, past and present, Arco Pub.

Eluyode, O. ve Akomolafe, D. T., 2013, Comparative study of biological and artificial neural networks, *European Journal of Applied Engineering and Scientific Research*, 2 (1), 36-46.

Erol, H., 2019, (kişisel iletişim tarih:18.12.2019).

Ertel, W., 2018, Introduction to artificial intelligence, Springer.

Esfe, M. H., Wongwises, S., Naderi, A., Asadi, A., Safaei, M. R., Rostamian, H., Dahari, M. ve Karimipour, A., 2015, Thermal conductivity of Cu/TiO2–water/EG hybrid nanofluid: Experimental data and modeling using artificial neural network and correlation, *International communications in heat and mass transfer*, 66, 100-104.

Estivill-Castro, V., 2002, Why so many clustering algorithms: a position paper, *SIGKDD explorations*, 4 (1), 65-75.

Feng, G., Huang, G.-B., Lin, Q. ve Gay, R., 2009, Error minimized extreme learning machine with growth of hidden nodes and incremental learning, *IEEE Transactions on Neural Networks*, 20 (8), 1352-1357.

Feng, X., Li, S., Yuan, C., Zeng, P. ve Sun, Y., 2018, Prediction of slope stability using naive Bayes classifier, *KSCE Journal of Civil Engineering*, 22 (3), 941-950.

Fish, K. E., Barnes, J. H. ve AikenAssistant, M. W., 1995, Artificial neural networks: a new methodology for industrial market segmentation, *Industrial Marketing Management*, 24 (5), 431-438.

Fix, E. ve Hodges Jr, J. L., 1951, Discriminatory analysis-nonparametric discrimination: consistency properties, *California Univ Berkeley*.

Floyd, C. E., Lo, J. Y., Yun, A. J., Sullivan, D. C. ve Kornguth, P. J., 1994, Prediction of breast cancer malignancy using an artificial neural network, *Cancer*, 74 (11), 2944-2948.

Footballhistory, 2019a, Premier League, https://www.footballhistory.org/league/premier-league.html: [Ziyaret Tarihi: 3.10.2019].

Footballhistory, 2019b, Spanish La Liga, https://www.footballhistory.org/league/la-liga.html: [Ziyaret Tarihi: 3.10.2019].

Footballhistory, 2019c, Italian Serie A, https://www.footballhistory.org/league/serie-a.html: [Ziyaret Tarihi: 3.10.2019].

Footballhistory, 2019d, French Ligue 1, https://www.footballhistory.org/league/ligue-1.html: [Ziyaret Tarihi: 3.10.2019].

Fukunaga, K. ve Hostetler, L., 1975, K-nearest-neighbor Bayes-risk estimation, *IEEE transactions on information theory*, 21 (3), 285-293.

Gan, G., Ma, C. ve Wu, J., 2007, Data clustering: theory, algorithms, and applications, Siam.

Gardner, H., 2011, Frames of mind: The theory of multiple intelligences, Hachette UK.

Garrett, D., Peterson, D. A., Anderson, C. W. ve Thaut, M. H., 2003, Comparison of linear, nonlinear, and feature selection methods for EEG signal classification, *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, 11 (2), 141-144.

Gevaria, K., Sanghavi, H., Vaidya, S. ve Deulkar, K., 2015, Football Match Winner Prediction.

Gislason, P. O., Benediktsson, J. A. ve Sveinsson, J. R., 2006, Random Forests for land cover classification, *Pattern Recognition Letters*, 27 (4), 294-300.

Goddard, J. ve Asimakopoulos, I., 2004, Forecasting football results and the efficiency of fixed-odds betting, *Journal of Forecasting*, 23 (1), 51-66.

Goddard, J., 2005, Regression models for forecasting goals and match results in association football, *International Journal of Forecasting*, 21 (2), 331-340.

Gomathi, V., Karthikeyan, S. ve Sohar, O., 2014, Performance analysis of distance measures for computer tomography image segmentation, *International Journal of Computer Technology and Applications*, 5 (2), 400-405.

Gong, P., Marceau, D. J. ve Howarth, P. J., 1992, A comparison of spatial feature extraction algorithms for land-use classification with SPOT HRV data, *Remote Sensing of Environment*, 40 (2), 137-151.

Grzymala-Busse, J. W. ve Hu, M., 2000, A comparison of several approaches to missing attribute values in data mining, *International Conference on Rough Sets and Current Trends in Computing*, 378-385.

Guo, G., Wang, H., Bell, D., Bi, Y. ve Greer, K., 2003, KNN model-based approach in classification, *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 986-996.

Gustafson, D. E. ve Kessel, W. C., 1979, Fuzzy clustering with a fuzzy covariance matrix, *1978 IEEE conference on decision and control including the 17th symposium on adaptive processes*, 761-766.

Hadi, A. S., 1994, A modification of a method for the detection of outliers in multivariate samples, *Journal of the Royal Statistical Society: Series B (Methodological)*, 56 (2), 393-396.

Hahn, G. J. ve Doganaksoy, N., 2012, A career in statistics: Beyond the numbers, John Wiley & Sons.

Hall, M. A., 1999, Correlation-based feature selection for machine learning.

Han, D., Chan, L. ve Zhu, N., 2007, Flood forecasting using support vector machines, *Journal of hydroinformatics*, 9 (4), 267-276.

Han, J., Pei, J. ve Kamber, M., 2011, Data mining: concepts and techniques, Elsevier.

Harrington, P., 2012, Machine learning in action, Manning Publications Co.

Hassanat, A. B., Abbadi, M. A., Altarawneh, G. A. ve Alhasanat, A. A., 2014, Solving the problem of the K parameter in the KNN classifier using an ensemble learning approach, *arXiv preprint arXiv:1409.0919*.

Hastie, T., Tibshirani, R. ve Friedman, J., 2009, Unsupervised learning, In: The elements of statistical learning, Eds: Springer.

Hawkins, D. M., 1980, Identification of outliers, Springer.

Hawkins, S., He, H., Williams, G. ve Baxter, R., 2002, Outlier detection using replicator neural networks, *International Conference on Data Warehousing and Knowledge Discovery*, 170-180.

Haykin, S., 1994, Neural networks: a comprehensive foundation, Prentice Hall PTR.

Heidt, R. S., Sweeterman, L. M., Carlonas, R. L., Traub, J. A. ve Tekulve, F. X., 2000, Avoidance of Soccer Injuries with Preseason Conditioning, *The American Journal of Sports Medicine*, 28 (5), 659-662.

Hellman, M. E., 1970, The nearest neighbor classification rule with a reject option, *IEEE Transactions on Systems Science and Cybernetics*, 6 (3), 179-185.

Ho, T. K., 1998, The random subspace method for constructing decision forests, *IEEE Transactions on pattern analysis and machine intelligence*, 20 (8), 832-844.

Hodge, V. ve Austin, J., 2004, A survey of outlier detection methodologies, *Artificial intelligence review*, 22 (2), 85-126.

Hoi, S. C., Liu, W., Lyu, M. R. ve Ma, W.-Y., 2006, Learning distance metrics with contextual constraints for image retrieval, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2072-2078.

Holmes, C. C. ve Held, L., 2006, Bayesian auxiliary variable models for binary and multinomial regression, *Bayesian analysis*, 1 (1), 145-168.

Holte, R. C., 1993, Very simple classification rules perform well on most commonly used datasets, *Machine Learning*, 11 (1), 63-90.

Hoogendoorn, M. ve Funk, B., 2017, Machine Learning for the Quantified Self: On the Art of Learning from Sensory Data, Springer International Publishing.

Hosmer, D. W. ve Lemeshow, S., 2000, Applied Logistic Regression. John Wiley & Sons, *New York*.

Hosmer, D. W., Lemeshow, S. ve Sturdivant, R. X., 2013, Applied logistic regression, John Wiley & Sons.

Houston, R. G. ve Wilson, D. P., 2002, Income, leisure and proficiency: an economic study of football performance, *Applied Economics Letters*, 9 (14), 939-943.

Höppner, F., Klawonn, F., Kruse, R. ve Runkler, T., 1999, Fuzzy cluster analysis: methods for classification, data analysis and image recognition, John Wiley & Sons.

Hu, L.-Y., Huang, M.-W., Ke, S.-W. ve Tsai, C.-F., 2016, The distance function effect on k-nearest neighbor classification for medical datasets, *SpringerPlus*, 5 (1), 1304.

Huang, J., 2006, Performance measures of machine learning, *University of Western Ontario*, 100.

Huang, K.-Y. ve Chen, K.-J., 2011, Multilayer perceptron for prediction of 2006 world cup football game, *Advances in Artificial Neural Systems*, 2011, 11.

Huang, T.-M., Kecman, V. ve Kopriva, I., 2006, Kernel based algorithms for mining huge data sets, Springer.

Hung, C.-C., Kulkarni, S. ve Kuo, B.-C., 2010, A new weighted fuzzy c-means clustering algorithm for remotely sensed image classification, *IEEE Journal of Selected Topics in Signal Processing*, 5 (3), 543-553.

Hung, M.-C. ve Yang, D.-L., 2001, An efficient fuzzy c-means clustering algorithm, *Proceedings 2001 IEEE International Conference on Data Mining*, 225-232.

Hvattum, L. M. ve Arntzen, H., 2010, Using ELO ratings for match result prediction in association football, *International Journal of Forecasting*, 26 (3), 460-470.

Hydrology, A. T. C. o. A. o. A. N. N. i., 2000, Artificial neural networks in hydrology. I: Preliminary concepts, *Journal of Hydrologic Engineering*, 5 (2), 115-123.

Igiri, C. P. ve Nwachukwu, E. O., 2014, An improved prediction system for football a match result, *IOSR Journal of Engineering (IOSRJEN)*, 4, 12-20.

Igiri, C. P., 2015, Support Vector Machine—Based Prediction System for a Football Match Result, *IOSR Journal of Computer Engineering (IOSR-JCE)*, 17 (3), 21-26.

Injuriesandsuspensions, 2013, Injuries & Suspensions, https://injuriesandsuspensions.com/about-us/:

Isaksson, A., Wallman, M., Göransson, H. ve Gustafsson, M. G., 2008, Cross-validation and bootstrapping are unreliable in small sample classification, *Pattern Recognition Letters*, 29 (14), 1960-1965.

Jack, L. ve Nandi, A., 2002, Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms, *Mechanical Systems and Signal Processing*, 16 (2-3), 373-390.

Jackson, D., 1990, The parameter gaming in sports, *Proceedings of the International Gaming Conference*, New York, USA.

Jain, A., Nandakumar, K. ve Ross, A., 2005, Score normalization in multimodal biometric systems, *Pattern Recognition*, 38 (12), 2270-2285.

Jain, A. K. ve Dubes, R. C., 1988, Algorithms for clustering data. Englewood Cliffs / New Jersey, Prentice Hall

Jain, A. K., Mao, J. ve Mohiuddin, K. M., 1996, Artificial neural networks: A tutorial, *Computer*, 29 (3), 31-44.

Jayalakshmi, T. ve Santhakumaran, A., 2011, Statistical normalization and back propagation for classification, *International Journal of Computer Theory and Engineering*, 3 (1), 1793-8201.

Jones, D. ve Green, M., 2019, Deloitte Annual Review of Football Finance 2019, Deloitte.

Joseph, A., Fenton, N. E. ve Neil, M., 2006, Predicting football results using Bayesian nets and other machine learning techniques, *Knowledge-Based Systems*, 19 (7), 544-553.

Jóźwik, A., 1983, A learning scheme for a fuzzy k-NN rule, *Pattern Recognition Letters*, 1 (5-6), 287-289.

Junge, A. ve Dvorak, J., 2004, Soccer Injuries, *Sports Medicine*, 34 (13), 929-938.

Kalogirou, S. A., 2000, Applications of artificial neural-networks for energy systems, *Applied energy*, 67 (1-2), 17-35.

Kantarcıoglu, M., Vaidya, J. ve Clifton, C., 2003, Privacy preserving naive bayes classifier for horizontally partitioned data, *IEEE ICDM workshop on privacy preserving data mining*, 3-9.

Kaplan, J., 2016, Artificial Intelligence: What everyone needs to know, Oxford University Press.

Karegowda, A. G., Manjunath, A. ve Jayaram, M., 2010, Comparative study of attribute selection using gain ratio and correlation based feature selection, *International Journal of Information Technology and Knowledge Management*, 2 (2), 271-277.

Karimi, Y., Prasher, S., Patel, R. ve Kim, S., 2006, Application of support vector machine technology for weed and nitrogen stress detection in corn, *Computers and Electronics in Agriculture*, 51 (1-2), 99-109.

Karlik, B. ve Olgac, A. V., 2011, Performance analysis of various activation functions in generalized MLP architectures of neural networks, *International Journal of Artificial Intelligence and Expert Systems*, 1 (4), 111-122.

Keller, J. M., Gray, M. R. ve Givens, J. A., 1985, A fuzzy k-nearest neighbor algorithm, *IEEE Transactions on Systems, Man, and Cybernetics* (4), 580-585.

Khan, J., Wei, J. S., Ringner, M., Saal, L. H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C. R. ve Peterson, C., 2001, Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, *Nature Medicine*, 7 (6), 673.

Khodadadzadeh, M., Li, J., Plaza, A. ve Bioucas-Dias, J. M., 2014, A Subspace-Based Multinomial Logistic Regression for Hyperspectral Image Classification, *Ieee Geoscience and Remote Sensing Letters*, 11 (12), 2105-2109.

Kiang, M. Y., 2003, A comparative assessment of classification methods, *Decision support systems*, 35 (4), 441-454.

Kim, J.-H., 2009, Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap, *Computational Statistics & Data Analysis*, 53 (11), 3735-3745.

Kınalıoğlu, İ. H., 2019, Veri Seti Oluşturma Programı.

Kira, K. ve Rendell, L. A., 1992, A practical approach to feature selection, In: Machine Learning Proceedings 1992, Eds: Elsevier.

Kodinariya, T. M. ve Makwana, P. R., 2013, Review on determining number of Cluster in K-Means Clustering, *International Journal*, 1 (6), 90-95.

Kohavi, R., 1995a, The power of decision tables, *European conference on machine learning*, 174-189.

Kohavi, R., 1995b, A study of cross-validation and bootstrap for accuracy estimation and model selection, *Ijcai*, 1137-1145.

Kononenko, I. ve Kukar, M., 2007, Machine learning and data mining, Horwood Publishing.

Koopman, S. J. ve Lit, R., 2015, A dynamic bivariate Poisson model for analysing and forecasting match results in the English Premier League, *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 178 (1), 167-186.

Kumar, G. ve Bhatia, P. K., 2014, A detailed review of feature extraction in image processing systems, *2014 Fourth international conference on advanced computing & communication technologies*, 5-12.

Kwak, C. ve Clayton-Matthews, A., 2002, Multinomial logistic regression, *Nursing research*, 51 (6), 404-410.

Lamkanfi, A., Demeyer, S., Soetens, Q. D. ve Verdonck, T., 2011, Comparing mining algorithms for predicting the severity of a reported bug, *2011 15th European Conference on Software Maintenance and Reengineering*, 249-258.

Lata, K. ve Gupta, P., 2016, Forecasting English Premier League Match Results, *International Journal of Modern Trends in Engineering and Research*, 03 (02), 261-268.

Legg, S. ve Hutter, M., 2007, A collection of definitions of intelligence, *Frontiers in Artificial Intelligence and applications*, 157, 17.

Lehr, D. ve Ohm, P., 2017, Playing with the data: what legal scholars should learn about machine learning, *UCDL Rev.*, 51, 653.

Leu, S.-S. ve Chang, C.-M., 2013, Bayesian-network-based safety risk assessment for steel construction projects, *Accident Analysis & Prevention*, 54, 122-133.

Lewis, D. D., 1998, Naive (Bayes) at forty: The independence assumption in information retrieval, *European conference on machine learning*, 4-15.

Li, Z., Yang, Y., Liu, J., Zhou, X. ve Lu, H., 2012, Unsupervised feature selection using nonnegative spectral analysis, *Twenty-Sixth AAAI Conference on Artificial Intelligence*.

Libbrecht, M. W. ve Noble, W. S., 2015, Machine learning applications in genetics and genomics, *Nature Reviews Genetics*, 16 (6), 321-332.

Lin, C.-F. ve Wang, S.-D., 2002, Fuzzy support vector machines, *IEEE Transactions on Neural Networks*, 13 (2), 464-471.

Liti, C., Piccialli, V. ve Sciandrone, M., 2017, Predicting soccer match outcome using machine learning algorithms, *Proceedings of MathSport International 2017 Conference*, 229.

Liu, H. ve Setiono, R., 1995, Chi2: Feature selection and discretization of numeric attributes, *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*, 388-391.

Liu, H. ve Yu, L., 2005, Toward integrating feature selection algorithms for classification and clustering, *IEEE Transactions on Knowledge & Data Engineering* (4), 491-502.

Liu, H. ve Motoda, H., 2007, Computational methods of feature selection, CRC Press.

Liu, H., Hopkins, W. G. ve Gómez, M.-A., 2016, Modelling relationships between match events and match outcome in elite football, *European journal of sport science*, 16 (5), 516-525.

Long, N. C., Meesad, P. ve Unger, H., 2015, A highly accurate firefly based algorithm for heart disease prediction, *Expert Systems with Applications*, 42 (21), 8221-8231.

Loohach, R. ve Garg, K., 2012, Effect of distance functions on k-means clustering algorithm, *International Journal of Computer Applications*, 50.

MacQueen, J., 1967, Some methods for classification and analysis of multivariate observations, *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, 281-297.

Manjusha, M. ve Harikumar, R., 2016, Performance analysis of KNN classifier and K-means clustering for robust classification of epilepsy from EEG signals, *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, 2412-2416.

Mannila, H., 1996, Data mining: machine learning, statistics, and databases, *Proceedings of 8th International Conference on Scientific and Statistical Data Base Management*, 2-9.

Martins, R. G., Martins, A. S., Neves, L. A., Lima, L. V., Flores, E. L. ve do Nascimento, M. Z., 2017, Exploring polynomial classifier to predict match results in football championships, *Expert Systems with Applications*, 83, 79-93.

Masetic, Z. ve Subasi, A., 2016, Congestive heart failure detection using random forest classifier, *Computer Methods and Programs in Biomedicine*, 130, 54-64.

McCarthy, J., 1998, What is artificial intelligence?, http://www-formal.stanford.edu/jmc/whatisai.pdf:

McCulloch, W. S. ve Pitts, W., 1943, A logical calculus of the ideas immanent in nervous activity, *The bulletin of mathematical biophysics*, 5 (4), 115-133.

McFadden, D., 1974, Conditional logit analysis of qualitative choice behavior, In, Eds, *California*: Institute of Urban and Regional Development, University of California.

Mehrez, A. ve Hu, M. Y., 1995, Predictors of the outcome of a soccer game—a normative analysis illustrated for the Israeli Soccer League, *Zeitschrift für Operations Research*, 42 (3), 361-372.

Mennis, J. ve Guo, D., 2009, Spatial data mining and geographic knowledge discovery—An introduction, *Computers, Environment and Urban Systems*, 33 (6), 403-408.

Mercer, J., 1909, Xvi. functions of positive and negative type, and their connection the theory of integral equations, *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, 209 (441-458), 415-446.

Metsis, V., Androutsopoulos, I. ve Paliouras, G., 2006, Spam filtering with naive bayes-which naive bayes?, *CEAS*, 28-69.

Michie, D., Spiegelhalter, D. J. ve Taylor, C. C., 1994, Machine Learning, Neural and Statistical Classification, Ellis Horwood.

Millard, K. ve Richardson, M., 2015, On the importance of training data sample selection in random forest image classification: A case study in peatland ecosystem mapping, *Remote sensing*, 7 (7), 8489-8515.

Ming, L. K., Kiong, L. C. ve Soong, L. W., 2011, Autonomous and deterministic supervised fuzzy clustering with data imputation capabilities, *Applied Soft Computing*, 11 (1), 1117-1125.

Miranda, E., Irwansyah, E., Amelga, A. Y., Maribondang, M. M. ve Salim, M., 2016, Detection of Cardiovascular Disease Risk's Level for Adults Using Naive Bayes Classifier, *Healthc Inform Res*, 22 (3), 196-205.

Mirkin, B., 2005, Clustering for Data Mining: A Data Recovery Approach, Chapman and Hall/CRC.

Mohri, M., Rostamizadeh, A. ve Talwalkar, A., 2018, Foundations of machine learning, MIT press.

Moons, K. G., Donders, R. A., Stijnen, T. ve Harrell Jr, F. E., 2006, Using the outcome for imputation of missing predictor values was preferred, *Journal of clinical epidemiology*, 59 (10), 1092-1101.

Morrow, S., 2003, The people's game: football, finance and society, *New York*, Palgrave Macmillan.

Muchlinski, D., Siroky, D., He, J. ve Kocher, M., 2016, Comparing random forest with logistic regression for predicting class-imbalanced civil war onset data, *Political Analysis*, 24 (1), 87-103.

Na, S., Xumin, L. ve Yong, G., 2010, Research on k-means clustering algorithm: An improved k-means clustering algorithm, *2010 Third International Symposium on intelligent information technology and security informatics*, 63-67.

Nabiyev, V. V., 2010, Yapay zeka: insan bilgisayar etkileşimi, Seçkin Yayıncılık.

Nagi, J., Yap, K. S., Tiong, S. K., Ahmed, S. K. ve Mohammad, A., 2008, Detection of abnormalities and electricity theft using genetic support vector machines, *TENCON 2008-2008 IEEE Region 10 Conference*, 1-6.

Narsky, I. ve Porter, F. C., 2013, Statistical analysis techniques in particle physics, Wiley Online Library.

Naumann, E., Huber, C., Maier, S., Plihal, W., Wustmans, A., Diedrich, O. ve Bartussek, D., 1992, The scalp topography of P300 in the visual and auditory modalities: a comparison of three normalization methods and the control of statistical type II error, *Electroencephalography and clinical Neurophysiology*, 83 (4), 254-264.

Nazim, R., Aida, M., Faiz Ahmad, Y. ve Ruhaya Ab, A., 2017, Predicting football matches results using bayesian networks for english premier League (EPL), *International Conference On Advances In Computing And Intelligent System (ICACIS2017)*, Melaka, Malaysia.

Neisser, U., Boodoo, G., Bouchard Jr, T. J., Boykin, A. W., Brody, N., Ceci, S. J., Halpern, D. F., Loehlin, J. C., Perloff, R. ve Sternberg, R. J., 1996, Intelligence: Knowns and unknowns, *American psychologist*, 51 (2), 77.

Nilsson, N. J., 2010, Yapay Zeka Geçmişi ve Geleceği, Boğaziçi Üniversitesi Yayınevi.

Omid, M., 2011, Design of an expert system for sorting pistachio nuts through decision tree and fuzzy logic classifier, *Expert Systems with Applications*, 38 (4), 4339-4347.

Oskoei, M. A. ve Hu, H., 2008, Support vector machine-based classification scheme for myoelectric control applied to upper limb, *Ieee Transactions on Biomedical Engineering*, 55 (8), 1956-1965.

Osuna, E., Freund, R. ve Girosi, F., 1997, An improved training algorithm for support vector machines, *Neural networks for signal processing VII. Proceedings of the 1997 IEEE signal processing society workshop*, 276-285.

Owen, A., 2011, Dynamic bayesian forecasting models of football match outcomes with estimation of the evolution variance parameter, *IMA Journal of Management Mathematics*, 22 (2), 99-113.

Özdamar, K., 2004, Paket programlar ile istatistiksel veri analizi (çok değişkenli analizler), *Kaan Kitabevi, Eskişehir*.

Özkan, C. ve Erbek, F. S., 2003, The comparison of activation functions for multispectral Landsat TM image classification, *Photogrammetric Engineering & Remote Sensing*, 69 (11), 1225-1234.

Özkan, Y., 2016, Veri madenciliği yöntemleri, Papatya Yayıncılık Eğitim.

Öztemel, E., 2012, Yapay Sinir Ağlari, *PapatyaYayincilik, Istanbul*.

Pal, M., 2005, Random forest classifier for remote sensing classification, *International Journal of Remote Sensing*, 26 (1), 217-222.

Panchal, G., Ganatra, A., Kosta, Y. ve Panchal, D., 2011, Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers, *International Journal of Computer Theory and Engineering*, 3 (2), 332-337.

Pandit, S. ve Gupta, S., 2011, A comparative study on distance measuring approaches for clustering, *International Journal of Research in Computer Science*, 2 (1), 29-31.

Parvinnia, E., Sabeti, M., Zolghadri Jahromi, M. ve Boostani, R., 2014, Classification of EEG Signals using adaptive weighted distance nearest neighbor algorithm, *Journal of King Saud University - Computer and Information Sciences*, 26 (1), 1-6.

Patel, V. R. ve Mehta, R. G., 2012, Data Clustering: Integrating Different Distance Measures with Modified k-Means Algorithm, New Delhi, 691-700.

Patil, D. B. ve Dongre, Y. V., 2015, A fuzzy approach for text mining, *IJ Mathematical Sciences and Computing*, 4, 34-43.

Pattekari, S. A. ve Parveen, A., 2012, Prediction system for heart disease using Naïve Bayes, *International Journal of Advanced Computer and Mathematical Sciences*, 3 (3), 290-294.

Peters, J., 2008, Machine learning for motor skills in robotics, *KI-Künstliche Intelligenz*, 2008 (4), 41-43.

Peterson, L. E., 2009, K-nearest neighbor, *Scholarpedia*, 4 (2), 1883.

Pham, D. T., Dimov, S. S. ve Nguyen, C. D., 2005, Selection of K in K-means clustering, *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 219 (1), 103-119.

Phan, J. H., Quo, C.-F. ve Wang, M. D., 2006, Functional genomics and proteomics in the clinical neurosciences: data mining and bioinformatics, *Progress in brain research*, 158, 83-108.

Piaget, J., 1952, The origins of intelligence in children, International Universities Press New York.

Pichler, O., Teuner, A. ve Hosticka, B. J., 1996, A comparison of texture feature extraction using adaptive Gabor filtering, pyramidal and tree structured wavelet transforms, *Pattern Recognition*, 29 (5), 733-742.

Polat, K., 2012, Classification of Parkinson's disease using feature weighting method on the basis of fuzzy C-means clustering, *International Journal of Systems Science*, 43 (4), 597-609.

Poole, D. L. ve Mackworth, A. K., 2010, Artificial Intelligence: foundations of computational agents, Cambridge University Press.

Prasetio, D. ve Dra. Harlili, M. S., 2016, Predicting football match results with logistic regression, *Advanced Informatics: Concepts, Theory And Application (ICAICTA), 2016 International Conference On*, 1-5.

Prateek, P., Ankit Kumar, G., Samarth, S., Neelu, K. ve Sweta, B., 2017, Football league win prediction based on online and league table data, *IOP Conference Series: Materials Science and Engineering*, 263 (4), 60-68.

Preece, S. J., Goulermas, J. Y., Kenney, L. P. ve Howard, D., 2008, A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data, *Ieee Transactions on Biomedical Engineering*, 56 (3), 871-879.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. ve Flannery, B. P., 2007, Numerical recipes 3rd edition: The art of scientific computing, Cambridge university press.

Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J. ve Solenberger, P., 2001, A multivariate technique for multiply imputing missing values using a sequence of regression models, *Survey methodology*, 27 (1), 85-96.

Rao, Y., Lee, Y., Jarjoura, D., Ruppert, A. S., Liu, C.-g., Hsu, J. C. ve Hagan, J. P., 2008, A comparison of normalization techniques for microRNA microarray data, *Statistical applications in genetics and molecular biology*, 7 (1).

Refaeilzadeh, P., Tang, L. ve Liu, H., 2007, On comparison of feature selection algorithms, *Proceedings of AAAI workshop on evaluation methods for machine learning II*, 5.

Rish, I., 2001, An empirical study of the naive bayes classifier, *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, 41-46.

Robertson, S., Back, N. ve Bartlett, J. D., 2016, Explaining match outcome in elite Australian Rules football using team performance indicators, *Journal of sports sciences*, 34 (7), 637-644.

Rogers, S. ve Girolami, M., 2016, A first course in machine learning, Chapman and Hall/CRC.

Rotshtein, A. P., Posner, M. ve Rakityanskaya, A., 2005, Football predictions based on a fuzzy model with genetic and neural tuning, *Cybernetics and Systems Analysis*, 41 (4), 619-630.

Royston, P., 2004, Multiple imputation of missing values, *The Stata Journal*, 4 (3), 227-241.

Rustempasic, I. ve Can, M., 2013, Diagnosis of parkinson's disease using fuzzy c-means clustering and pattern recognition, *Southeast Europe Journal of Soft Computing*, 2 (1).

Salas, J., Markus, M. ve Tokar, A., 2000, Streamflow forecasting based on artificial neural networks, In: Artificial neural networks in hydrology, Eds: Springer.

Samanta, B., Al-Balushi, K. ve Al-Araimi, S., 2003, Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection, *Engineering applications of artificial intelligence*, 16 (7-8), 657-665.

Samanthula, B. K., Elmehdwi, Y. ve Jiang, W., 2015, k-Nearest Neighbor Classification over Semantically Secure Encrypted Relational Data, *Ieee Transactions on Knowledge and Data Engineering*, 27 (5), 1261-1273.

Scholkopf, B. ve Smola, A. J., 2001, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press.

Schölkopf, B., Smola, A. J. ve Bach, F., 2002, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press.

Schuld, M. ve Petruccione, F., 2018, Supervised Learning with Quantum Computers, Springer.

Seidel, M., 2000, Transfermarkt.Com, https://www.transfermarkt.com/intern/impressum:

Sharma, V., Rai, S. ve Dev, A., 2012, A comprehensive study of artificial neural networks, *International Journal of Advanced research in computer science and software engineering*, 2 (10).

Shearer, C., 2000, The CRISP-DM model: the new blueprint for data mining, *Journal of data warehousing*, 5 (4), 13-22.

Shieh, H.-L., 2014, A Hybrid Fuzzy Clustering Method with a Robust Validity Index, *International Journal of Fuzzy Systems*, 16 (1).

Shrawankar, U. ve Thakare, V. M., 2013, Techniques for feature extraction in speech recognition system: A comparative study, *arXiv preprint arXiv:1305.1145*.

Simon, H. A., 1983, Why should machines learn?, In: Machine Learning, Volume I, Eds: Elsevier.

Sing, T., Sander, O., Beerenwinkel, N. ve Lengauer, T., 2005, ROCR: visualizing classifier performance in R, *Bioinformatics*, 21 (20), 3940-3941.

Song, Y., Huang, J., Zhou, D., Zha, H. ve Giles, C. L., 2007, Iknn: Informative k-nearest neighbor pattern classification, *European Conference on Principles of Data Mining and Knowledge Discovery*, 248-264.

Sopharak, A., Uyyanonvara, B. ve Barman, S., 2009, Automatic exudate detection from non-dilated diabetic retinopathy retinal images using fuzzy c-means clustering, *sensors*, 9 (3), 2148-2161.

SpolaôR, N., Cherman, E. A., Monard, M. C. ve Lee, H. D., 2013, A comparison of multi-label feature selection methods using the problem transformation approach, *Electronic Notes in Theoretical Computer Science*, 292, 135-151.

Sternberg, R. J. ve Detterman, D. K., 1986, What is intelligence?: Contemporary viewpoints on its nature and definition, Praeger Pub Text.

Sun, Z., Bebis, G., Yuan, X. ve Louis, S. J., 2002, Genetic feature subset selection for gender classification: A comparison study, *Sixth IEEE Workshop on Applications of Computer Vision, 2002.(WACV 2002). Proceedings.*, 165-170.

Sutton, R. S., 1992, Reinforcement learning architectures, *Proceedings ISKIT*, 92.

Sutton, R. S. ve Barto, A. G., 1998, Introduction to reinforcement learning, MIT press Cambridge.

Tang, J., Alelyani, S. ve Liu, H., 2014, Feature selection for classification: A review, *Data classification: Algorithms and applications*, 37.

TDK, 2019, Türk Dil Kurumu Sözlükleri, http://sozluk.gov.tr/: [24.07.2019].

Terman, L. M., 1921, Intelligence and its measurement: A symposium--II, *Journal of Educational Psychology*, 12 (3), 127.

TFF, 2019, Süper Lig Tarihçesi, https://www.tff.org/default.aspx?pageID=545: [Ziyaret Tarihi: 3.10.2019].

Thurstone, L. L., 2013, The nature of intelligence, Routledge.

Timmaraju, A. S., Palnitkar, A. ve Khanna, V., 2013, Game ON! Predicting English Premier League Match Outcomes.

Ting, K. M., 2017, Confusion Matrix, In: Encyclopedia of Machine Learning and Data Mining, Eds: Sammut, C. ve Webb, G. I., *Boston, MA*: Springer US.

Tipping, M. E. ve Bishop, C. M., 1999, Probabilistic principal component analysis, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61 (3), 611-622.

Tong, S. ve Koller, D., 2001, Support vector machine active learning with applications to text classification, *Journal of machine learning research*, 2 (Nov), 45-66.

Topcu, I. B. ve Sarıdemir, M., 2008, Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic, *Computational Materials Science*, 41 (3), 305-311.

Torres-Toledano, J. G. ve Sucar, L. E., 1998, Bayesian Networks for Reliability Analysis of Complex Systems, Berlin, Heidelberg, 195-206.

Trafalis, T. B. ve Ince, H., 2000, Support vector machine for regression and applications to financial forecasting, *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, 348-353.

Tsangaratos, P. ve Ilia, I., 2016, Comparison of a logistic regression and Naïve Bayes classifier in landslide susceptibility assessments: The influence of models complexity and training dataset size, *Catena*, 145, 164-179.

Ulmer, B., Fernandez, M. ve Peterson, M., 2013, Predicting Soccer Match Results in the English Premier League, *Doctoral dissertation, Ph. D. dissertation, Stanford*.

Vaidya, S., Sanghavi, H. ve Gevaria, K., 2016, Football Match Winner Prediction, *International Journal of Computer Applications*, 154 (3).

Von Luxburg, U. ve Schölkopf, B., 2011, Statistical learning theory: Models, concepts, and results, In: Handbook of the History of Logic, Eds: Elsevier.

Wahba, G., 1999, Support vector machines, reproducing kernel Hilbert spaces and the randomized GACV, *Advances in Kernel Methods-Support Vector Learning*, 6, 69-87.

Wang, J., Neskovic, P. ve Cooper, L. N., 2007, Improving nearest neighbor rule with a simple adaptive distance measure, *Pattern Recognition Letters*, 28 (2), 207-213.

Wang, L., 2005, Support vector machines: theory and applications, Springer Science & Business Media.

Wang, Z., Lai, C., Chen, X., Yang, B., Zhao, S. ve Bai, X., 2015, Flood hazard risk assessment model based on random forest, *Journal of Hydrology*, 527, 1130-1141.

Wechsler, D., 1958, The measurement and appraisal of adult intelligence.

Wei, D., Kaplan, S. ve Chan, H. A., 2008, Energy efficient clustering algorithms for wireless sensor networks, *ICC Workshops-2008 IEEE International Conference on Communications Workshops*, 236-240.

Weiss, S. M., Kapouleas, I. ve Shavlik, J., 1990, An empirical comparison of pattern recognition, neural nets and machine learning classification methods, *Readings in machine learning*, 177-183.

Widrow, B., 1988, DARPA Neural Network Study, *Armed Forces Communication and Electronics Association, Fairfax, VA1988*.

Wikimedia, 2019, Bundesliga, https://tr.wikipedia.org/wiki/Bundesliga: [Ziyaret Tarihi: 3.10.2019].

Willis, T. H., Huston, C. R. ve Pohlkamp, F., 1993, Evaluation measures of just-in-time supplier performance, *Production and Inventory Management Journal*, 34 (2), 1.

Witten, I. ve Frank, E., 1993, Data Mining: Practical machine learning tools with Java implementations, ed, *M. Kaufmann, San Francisco*.

Wold, S., Esbensen, K. ve Geladi, P., 1987, Principal component analysis, *Chemometrics and Intelligent Laboratory Systems*, 2 (1-3), 37-52.

Wu, B., Abbott, T., Fishman, D., McMurray, W., Mor, G., Stone, K., Ward, D., Williams, K. ve Zhao, H., 2003, Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data, *Bioinformatics*, 19 (13), 1636-1643.

Xing, F., Guo, P. ve Lyu, M. R., 2005, A novel method for early software quality prediction based on support vector machine, *16th IEEE International Symposium on Software Reliability Engineering (ISSRE'05)*, 10 pp.-222.

Yang, W., Wang, J., Ren, M., Yang, J., Zhang, L. ve Liu, G., 2009, Feature extraction based on Laplacian bidirectional maximum margin criterion, *Pattern Recognition*, 42 (11), 2327-2334.

Yao, X. ve Liu, Y., 1997, A new evolutionary system for evolving artificial neural networks, *IEEE Transactions on Neural Networks*, 8 (3), 694-713.

Yegnanarayana, B., 2005, Artificial neural networks, PHI Learning Pvt. Ltd.

Yu, L., Ding, C. ve Loscalzo, S., 2008, Stable feature selection via dense feature groups, *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 803-811.

Yuntian, W., 2012, Based on machine learning of data mining to further explore, *Computer Science and Information Processing (CSIP), 2012 International Conference on*, 1235-1238.

Yurdanur, E., 2002, Maçkolik İnternet Hizmetleri, https://www.mackolik.com/kunye:

Zadeh, M. R., Amin, S., Khalili, D. ve Singh, V. P., 2010, Daily outflow prediction by multi layer perceptron with logistic sigmoid and tangent sigmoid activation functions, *Water resources management*, 24 (11), 2673-2688.

Zhang, C., Liu, C., Zhang, X. ve Almpanidis, G., 2017, An up-to-date comparison of state-of-the-art classification algorithms, *Expert Systems with Applications*, 82, 128-150.

Zhang, G., Patuwo, B. E. ve Hu, M. Y., 1998, Forecasting with artificial neural networks:: The state of the art, *International Journal of Forecasting*, 14 (1), 35-62.

Zhang, H., 2004, The optimality of naive Bayes, *AA*, 1 (2), 3.

Zhu, X., Ghahramani, Z. ve Lafferty, J. D., 2003, Semi-supervised learning using gaussian fields and harmonic functions, *Proceedings of the 20th International conference on Machine learning (ICML-03)*, 912-919.

Zhu, X. ve Goldberg, A. B., 2009, Introduction to semi-supervised learning, *Synthesis lectures on artificial intelligence and machine learning*, 3 (1), 1-130.

Zhu, X., Du, X., Kerich, M., Lohoff, F. W. ve Momenan, R., 2018, Random forest based classification of alcohol dependence patients and healthy controls using resting state MRI, *Neuroscience letters*, 676, 27-33.

Zhu, X. J., 2005, Semi-supervised learning literature survey, *University of Wisconsin-Madison Department of Computer Sciences*.

Zou, H., Hastie, T. ve Tibshirani, R., 2006, Sparse principal component analysis, *Journal of computational and graphical statistics*, 15 (2), 265-286.

EKLER

EK-1: Kullanılan sınıflandırma ve kümeleme yöntemlerinin R dilindeki kodları

```
library(caret)
library(e1071)
library(ROCR)
library(EleMStatLearn)
library(dplyr)
library(ggplot2)
library(scales)
library(randomForest)
library(class)
library(MASS)
library(e1071)
library(neuralnet)
library(rminer)
library(nnet)
library(aod)
library(factoextra)
library(cluster)
library(NbClust)
library(reshape)
library(neuralnet)
```

```
#####################################################################
####################    VERİ SEÇİMİ BAŞLANGIÇ    ####################
#  vesecim değişkeni iki değer alır
#  vsecim=1 olduğunda 'yol' değişkeni belirtilen dosyayı seçer,
#  vsecim=0 olduğunda gözatma penceresi açılır ve kullanıcı veri
dosyasını seçer.
vsecim<-0;
yol<-"veri.csv";
if (vsecim==0){veriler <- read.csv(file.choose(),header=TRUE,
sep=",")}
if (vsecim==1){veriler <- read.csv(file=yol, header=TRUE, sep=",")}
####################      VERİ SEÇİMİ BİTİŞ    ###################
#####################################################################
```

```
#####################################################################
################    VERİ NORMALLEŞTİRME BAŞLANGIÇ    ###############
# nrow:veri setindeki satır sayısı
satirsay<-nrow(veriler)
```

```
# ncol:veri setindeki sütun sayısı
sutunsay<-ncol(veriler)
#normalleştirme fonksiyonu yaratılıyor
normalize <- function(x) {return ((x - min(x)) / (max(x) -
min(x)))}
#veriler dosyasında bağıMSız değişkenlerin tamamını normalleştir.
normalveriler <- as.data.frame(lapply(veriler[1:sutunsay-1],
normalize))
#normalleştirilen verilere MS isimli bir sütun ekler ekler
normalveriler["MS"]<- NA
#eklenen MS isimli sütunun değerlerini 'veriler' isimli veri
setinden çeker.
normalveriler$MS <- veriler$MS
normalveriler$MS <- as.factor(normalveriler$MS)
################      VERİ NORMALLEŞTİRME BİTİŞ     ###############
#####################################################################



#####################################################################
##########   HOLDOUT (%) EĞİTİM_TEST AYIRMA BAŞLANGIÇ  ############
testorani=20
ysa_egitim_oran=0.8
testsay<-round(satirsay/(100/testorani))
egitimsay<-satirsay-testsay
secim=1;
test_baslangic=satirsay-testsay;
#secim değişkeni iki değer alır,
# '1' oluğunda test verisi rasgele seçilir
# '0' olduğunda ise belirlenen aralıktan seçilir.
if (secim==1){
  test.rows <- sample(1:nrow(normalveriler), testsay)
  test <- normalveriler[test.rows, ]
  egitim <- normalveriler[-test.rows, ]}
if (secim==0){
  egitim <- normalveriler[1:test_baslangic,1:sutunsay]
  test <- normalveriler[(test_baslangic+1):satirsay,1:sutunsay]
  egitimsay=nrow(egitim);
  testsay=nrow(test);}
##########     HOLDOUT (%) EĞİTİM_TEST AYIRMA BİTİŞ    ############
#####################################################################
```

```
#####################################################################
#############    SINIFLANDIRMA YÖNTEMLERİ  BAŞLANGIÇ  #############
#####################################################################

#####################################################################
############   1 Çok Terimli Lojistik Reg. BAŞLANGIÇ   #############
Tum_Degisken_isimleri<-colnames(egitim[1:sutunsay-1])
tahmin_degiskenleri<-
Tum_Degisken_isimleri[!Tum_Degisken_isimleri%in%""]
tahmin_degiskenleri<-paste(tahmin_degiskenleri,collapse="+")
formul=as.formula(paste("MS~",tahmin_degiskenleri,collapse="+"))
model_CLR<- multinom(MS ~ ., data=egitim, family =
binomial(link="logit"))
kestirim_skorlari <- predict (model_CLR, test, "probs")
kestirilen_siniflar <- predict (model_CLR, test)
konfisyonmatrisi_CLR<-table(kestirilen_siniflar, test$MS)
model_CLR_basari<-
sum(diag(konfisyonmatrisi_CLR))/sum(konfisyonmatrisi_CLR)
###########      1 Çok Terimli Lojistik Reg.  BİTİŞ     ###########
#####################################################################

#####################################################################
##############    2 Destek Vektör Mak. BAŞLANGIÇ    ##############
modeldvm<-svm(MS~.,data=egitim, kernel="linear")
kestirim_dvm<-predict(modeldvm,test)
tablo_dvm<-table(kestirim_dvm,test$MS)
konfisyonmatrisi_dvm<-table(kestirim_dvm,test$MS)
dvm_basari<-
sum(diag(konfisyonmatrisi_dvm))/sum(konfisyonmatrisi_dvm)
##############       2 Destek Vektör Mak. BİTİŞ      ##############
#####################################################################

#####################################################################
##############     3 K-En Yakın Komşu BAŞLANGIÇ     ##############
egitim.target_keyk<- egitim[1:egitimsay,sutunsay]
test.target_keyk<- test[1:testsay,sutunsay]
egitim_keyk<-egitim[1:sutunsay-1]
test_keyk<-test[1:sutunsay-1]
modelknn.2 <-  knn(train=egitim_keyk, test=test_keyk,
cl=egitim.target_keyk, k=151)
table(modelknn.2 ,test.target_keyk)
konfisyonmatrisi_keyk<-table(modelknn.2, test.target_keyk)
keyk_basari<-
sum(diag(konfisyonmatrisi_keyk))/sum(konfisyonmatrisi_keyk)
##############        3 K-En Yakın Komşu BİTİŞ       ##############
#####################################################################
```

```
#####################################################################
#################    4 Rasgele Orman BAŞLANGIÇ     ################
set.seed(222)
model_ro <- randomForest(MS~., data=egitim, ntree = 500, mtry =
8,importance = TRUE, proximity = TRUE)
kestirim_ro <- predict(model_ro, test)
tablo_ro<-table(kestirim_ro,test$MS)
konfisyonmatrisi_ro<-table(kestirim_ro,test$MS)
ro_basari<-sum(diag(konfisyonmatrisi_ro))/sum(konfisyonmatrisi_ro)
#################      4 Rasgele Orman BAİTİŞ      ################
#####################################################################

#####################################################################
#################     5 Sade Bayes  BAŞLANGIÇ    ################
modelsb<-naiveBayes(as.factor(MS)~.,data=egitim)
kestirim_sb<-predict(modelsb,test)
tablo_sb<-table(kestirim_sb,test$MS)
konfisyonmatrisi_sb<-table(kestirim_sb,test$MS)
sb_basari<-sum(diag(konfisyonmatrisi_sb))/sum(konfisyonmatrisi_sb)
#################      5 Sade Bayes BİTİŞ      ################
#####################################################################

#####################################################################
################   6 Yapay Sinir Ağlar BAŞLANGIÇ   ################
veriler_ann<-veriler
veriler_ann$sifir <- veriler_ann$MS=="0"
veriler_ann$bir <- veriler_ann$MS == "1"
veriler_ann$iki <- veriler_ann$MS == "2"
veriler_ann.train.idx <- sample(x = satirsay, size =
satirsay*ysa_egitim_oran)
veriler_ann.train <- veriler_ann[veriler_ann.train.idx,]
veriler_ann.valid <- veriler_ann[-veriler_ann.train.idx,]
Tum_Degisken_isimleri_ann<-colnames(veriler[1:ncol(veriler)-1])
tahmin_degiskenleri_ann<-
Tum_Degisken_isimleri_ann[!Tum_Degisken_isimleri_ann%in%""]
tahmin_degiskenleri_ann<-
paste(tahmin_degiskenleri_ann,collapse="+")
formul_ann=as.formula(paste("sifir+bir+iki~",tahmin_degiskenleri_an
n,collapse="+"))
modelann <- neuralnet(formula=formul_ann,
                      data=veriler_ann.train, hidden=c(0), rep = 5,
err.fct = "ce", linear.output = F, lifesign = "minimal", stepmax =
1000000, threshold = 0.001)
veriler.prediction <- compute(modelann, veriler_ann.valid[-
(sutunsay-1):-(sutunsay+1)])
```

```
idx <- apply(veriler_ann.prediction$net.result, 1, which.max)
kestirim_ysa <- c('0', '1', '2')[idx]
konfisyonmatrisi_ysa<-table(kestirim_ysa, veriler_ann.valid$MS)
YSA_basari<-
sum(diag(konfisyonmatrisi_ysa))/sum(konfisyonmatrisi_ysa)
################     6 Yapay Sinir Ağlar BİTİŞ     ################
#####################################################################

#####################################################################
################    SINIFLANDIRMA YÖNTEMLERİ  SON  ################
#####################################################################



#####################################################################
################    KÜMELEME YÖNTEMLERİ BAŞLANGIÇ  ################
#####################################################################

#####################################################################
#################### 1 - K ORTALAMALAR BAŞLANGIÇ ####################
ko_kume_say<-3
ko_x<-normalveriler[1:satirsay,1:(sutunsay-1)]
ko_kumeleme <- kmeans(x= ko_x  , centers = ko_kume_say, nstart =
25)
ko_kume_bilgisi<-ko_kumeleme[["cluster"]]
KO_yeni_veriler<-cbind(ko_x,ko_kume_bilgisi,normalveriler$MS)
#oluşturulan 'KO_yeni_veriler' isimli yeni veri seti yukarıdaki
# en üstteki 'veriler' isimli dosya ile değiştirilerek
# script tekrar çalıştırılır ve KOO uzantılı sonuçlar elde edilir.
####################  1 - K ORTALAMALAR son  ####################
#####################################################################

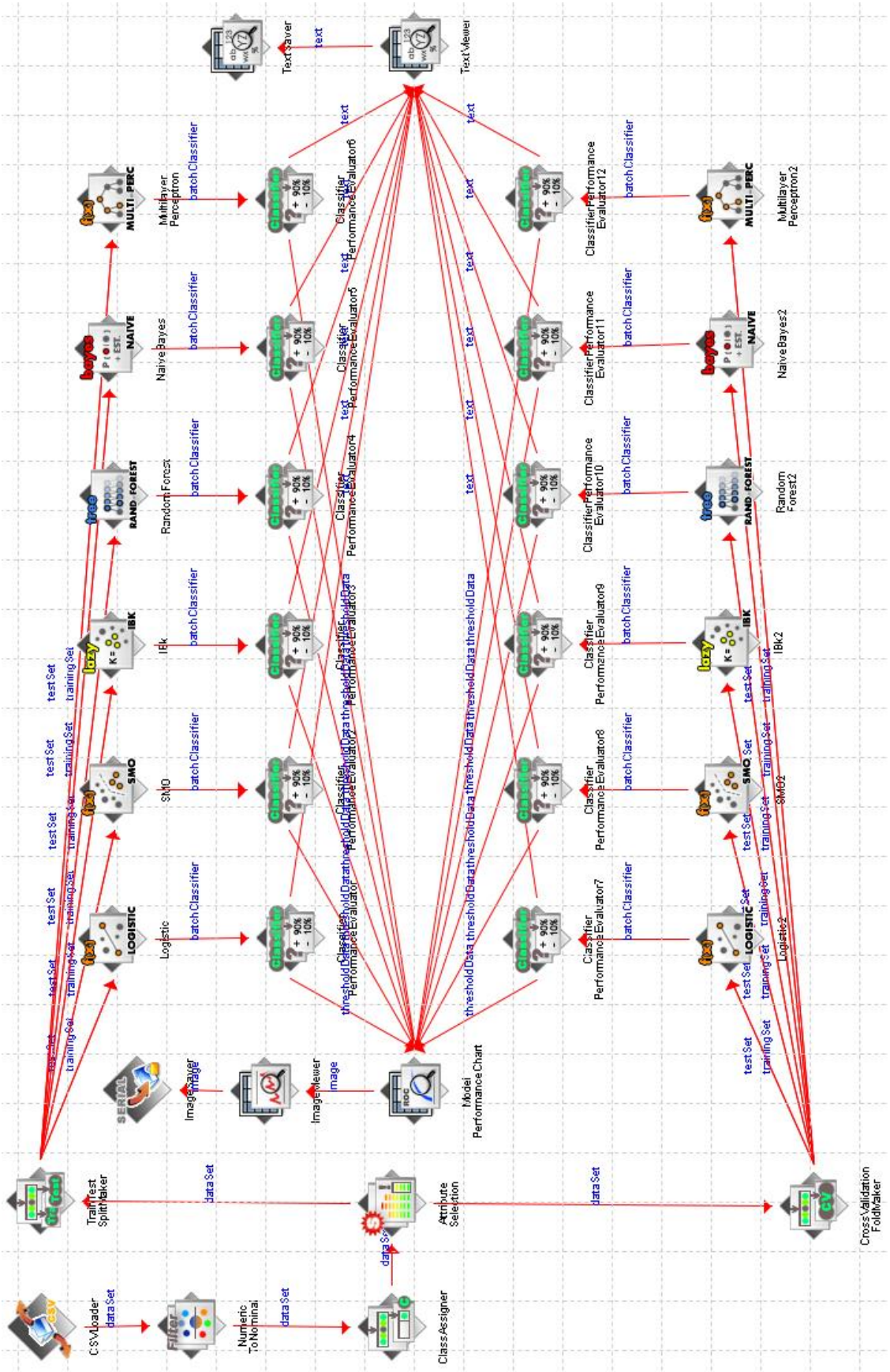
#####################################################################
############### 2 - BULANIK C ORTALAMALAR BAŞLANGIÇ ###############
bco_kume_say<-3
bco_x<-normalveriler[1:satirsay,1:(sutunsay-1)]
veriler.bco <- fcm(bco_x, centers=bco_kume_say)
as.data.frame(veriler.bco$u)[1:sutunsay,]
veriler.bco$v0
veriler.bco$v
summary(veriler.bco)
uyelik_dereceleri<-veriler.bco[["u"]]
bco_kume_bilgisi<-veriler.bco[["cluster"]]
BCO_yeni_veriler<-
cbind(bco_x,uyelik_dereceleri,bco_kume_bilgisi,normalveriler$MS)
#oluşturulan 'BCO_yeni_veriler' isimli yeni veri seti yukarıdaki
# en üstteki 'veriler' isimli dosya ile değiştirilerek
# script tekrar çalıştırılır ve BCO uzantılı sonuçlar elde edilir.
```

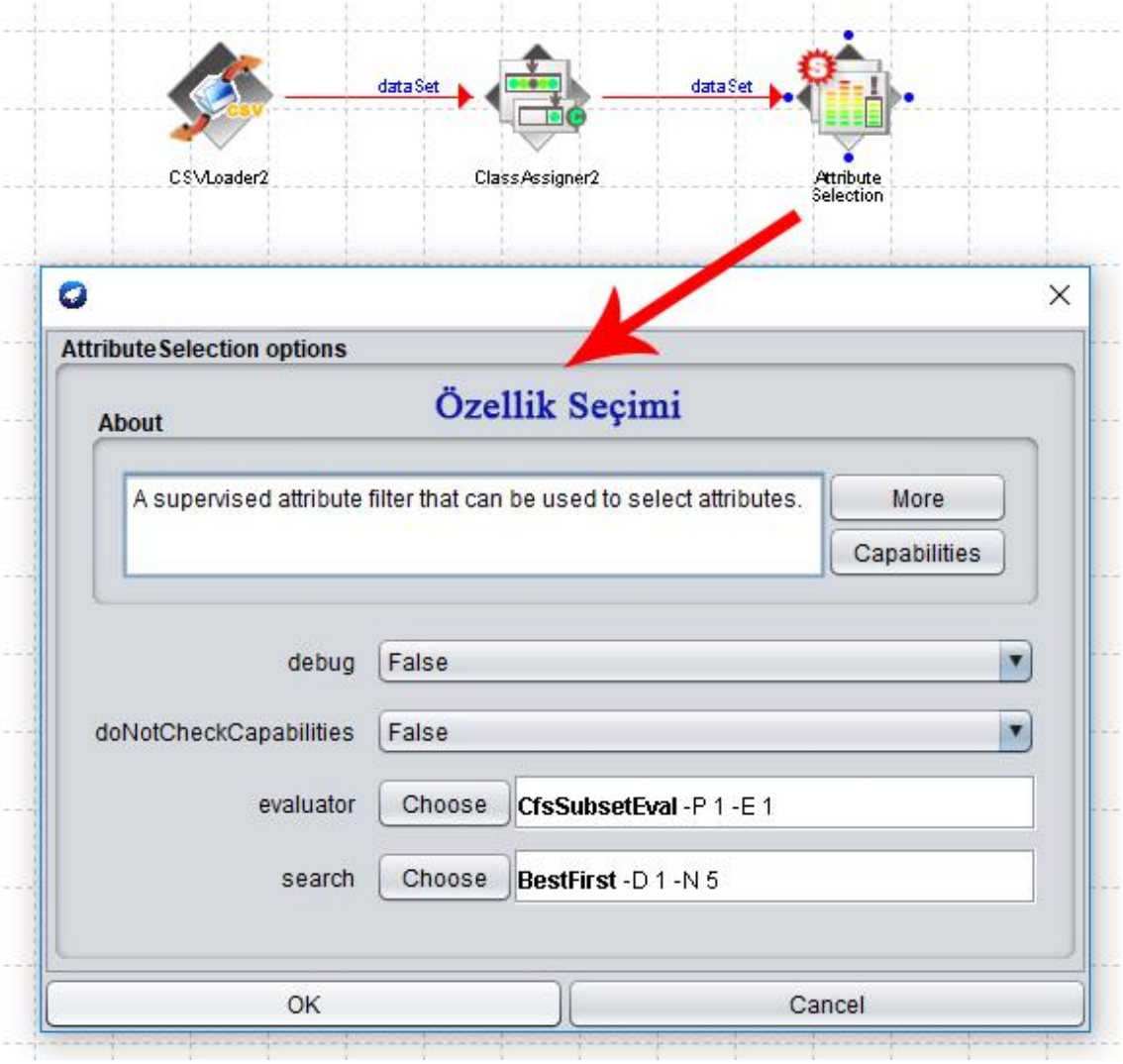
```
################ 2 -  BULANIK C ORTALAMALAR SON   #################
#####################################################################

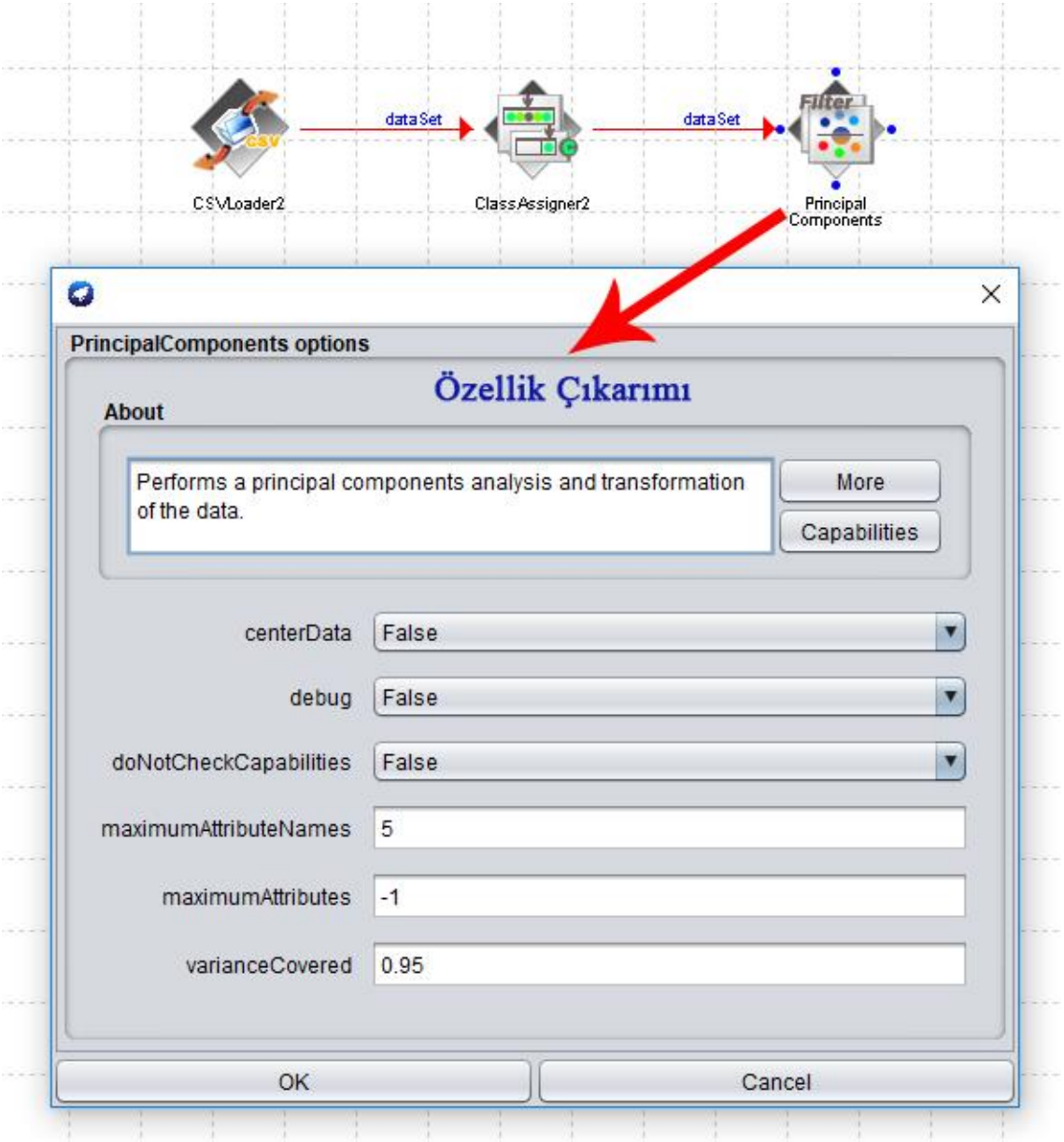
#####################################################################
###################    KÜMELEME YÖNTEMLERİ SON  ###################
#####################################################################

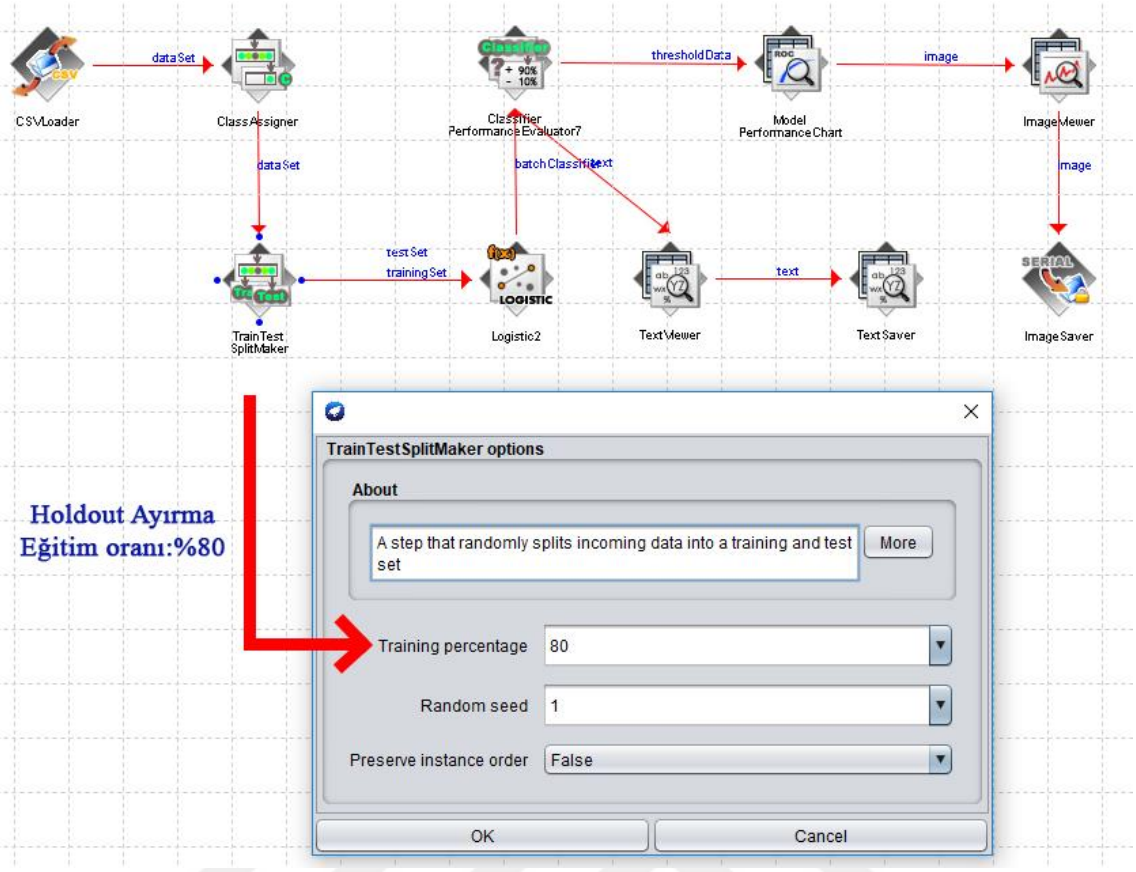

#####################################################################
#####################    SONUÇLAR BAŞLANGIÇ  ########################
#####################################################################
kmsb<-t(konfisyonmatrisi_sb)
kmclr<-t(konfisyonmatrisi_CLR)
kmkeyk<-t(konfisyonmatrisi_keyk)
kmro<-t(konfisyonmatrisi_ro)
kdvm<-t(konfisyonmatrisi_dvm)
kmysa<-t(konfisyonmatrisi_ysa)
print(kmclr)
cat("Çoklu Lojistik Regresyon Kestirim Başası:%",
model_CLR_basari*100)
print(kdvm)
cat("Destek Vektör Makinesi Kestirim Başası:%", dvm_basari*100)
print(kmkeyk)
cat("K En Yakın Komşu Kestirim Başası:%", keyk_basari*100)
print(kmro)
cat("Rasgele Orman Kestirim Başası:%", ro_basari*100)
print(kmsb)
cat("Sade Bayes Kestirim Başası:%", sb_basari*100)
print(kmysa)
cat("Yapay Sinir Ağları Kestirim Başası:%", YSA_basari*100)
#####################################################################
#######################   SONUÇLAR BİTİŞ    #########################
#####################################################################
```

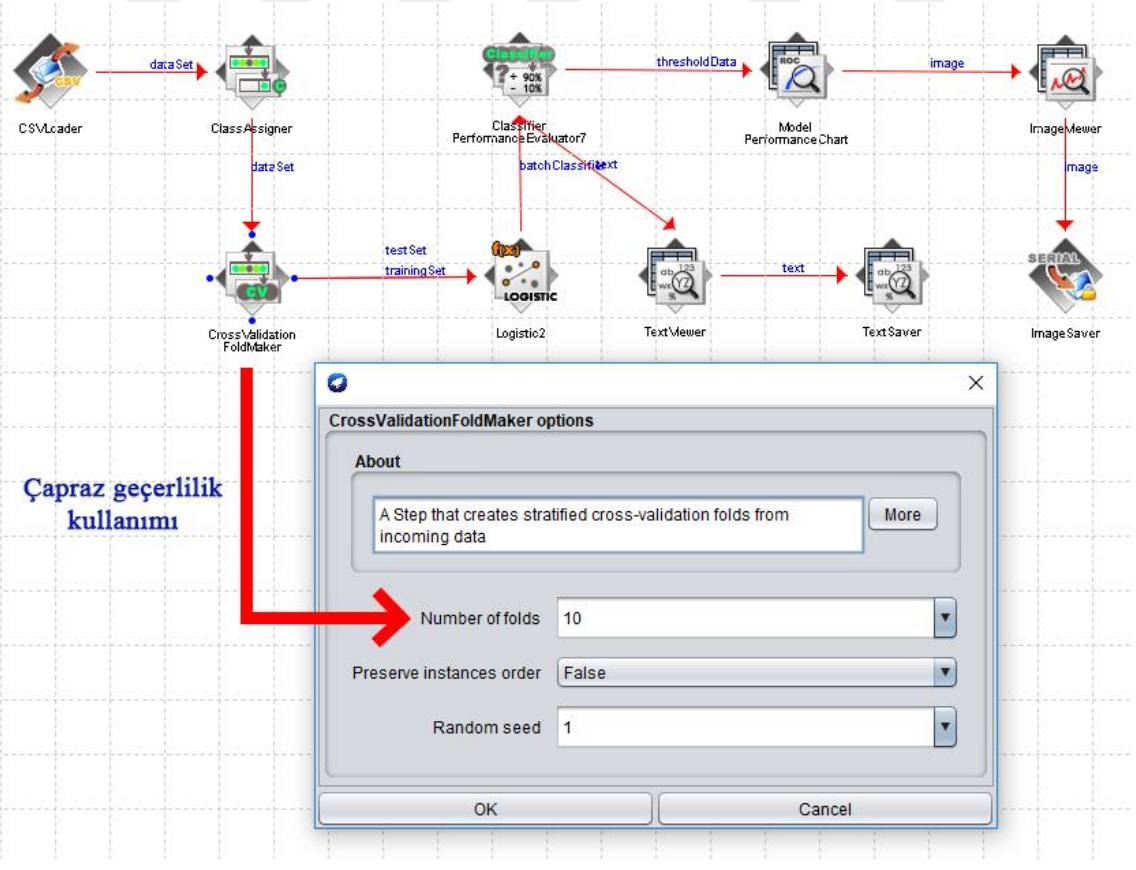
EK-2: Kullanılan sınıflandırma yöntemlerinin WEKA yazılımının “Knowledge Flow” aracı ile oluşturulmuş çözümler için örnek akış diyagramları.

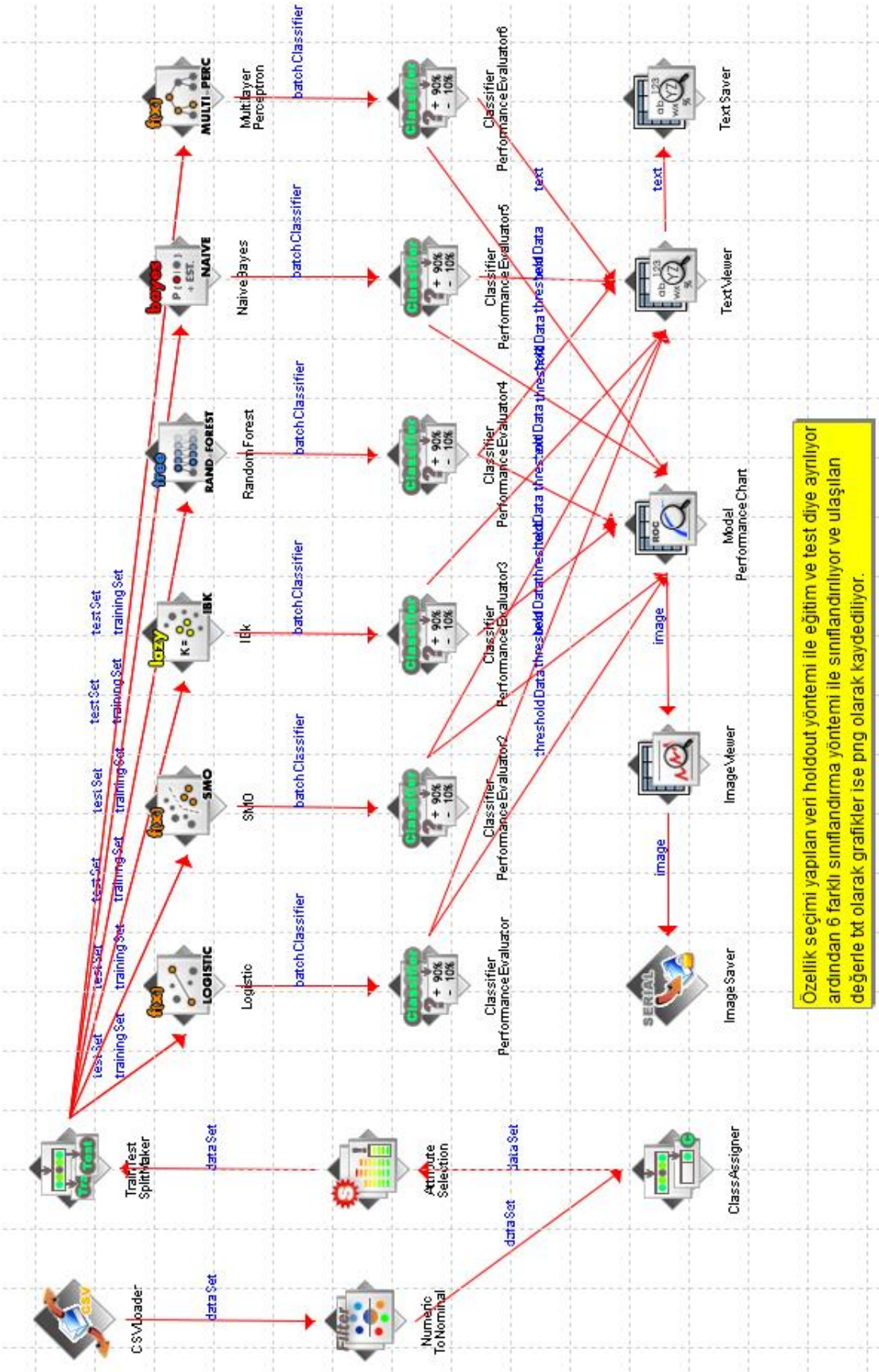


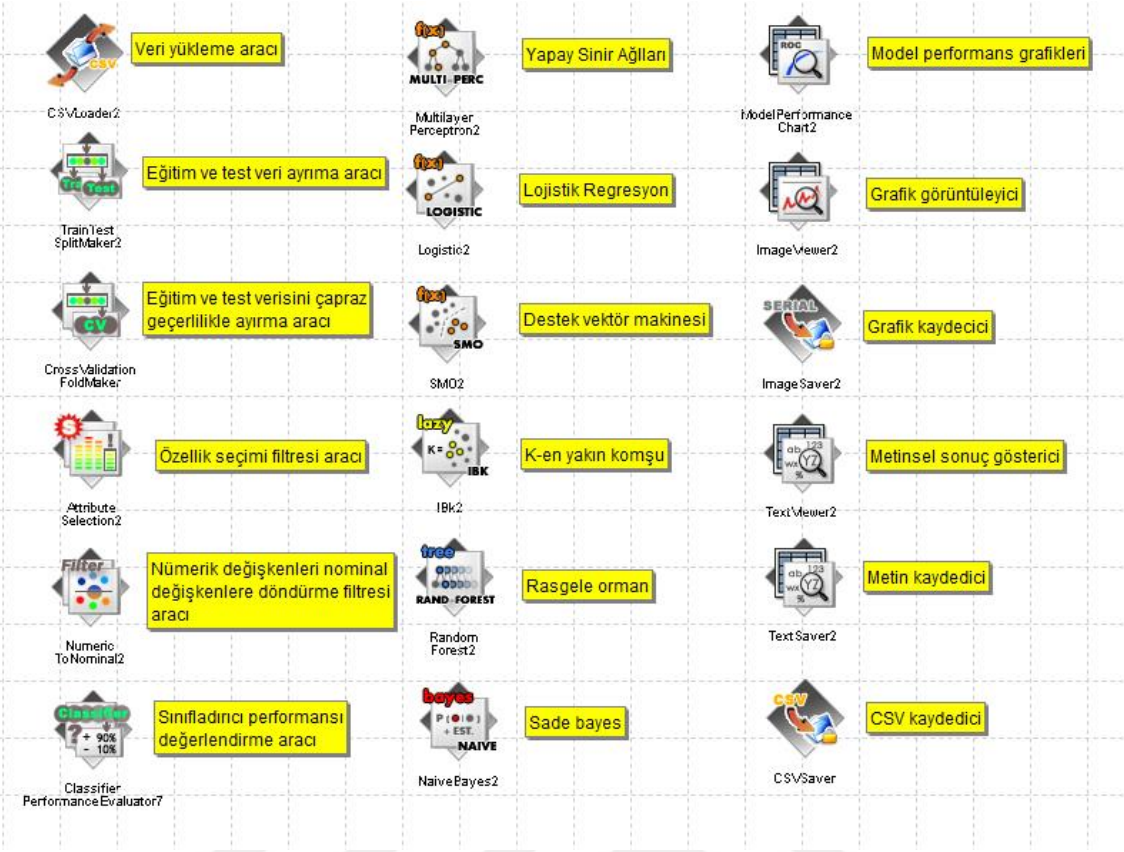

dataSet
dataSet
CSVLoader2
ClassAssigner2
Attribute Selection
AttributeSelection options
Özellik Seçimi
About
A supervised attribute filter that can be used to select attributes.
More
Capabilities
debug
False
doNotCheckCapabilities
False
evaluator
Choose
CfsSubsetEval -P 1 -E 1
search
Choose
BestFirst -D 1 -N 5
OK
Cancel


dataSet
dataSet
CSVLoader2
ClassAssigner2
Principal Components
PrincipalComponents options
Özellik Çıkarımı
About
Performs a principal components analysis and transformation of the data.
More
Capabilities
centerData
False
debug
False
doNotCheckCapabilities
False
maximumAttributeNames
5
maximumAttributes
-1
varianceCovered
0.95
OK
Cancel

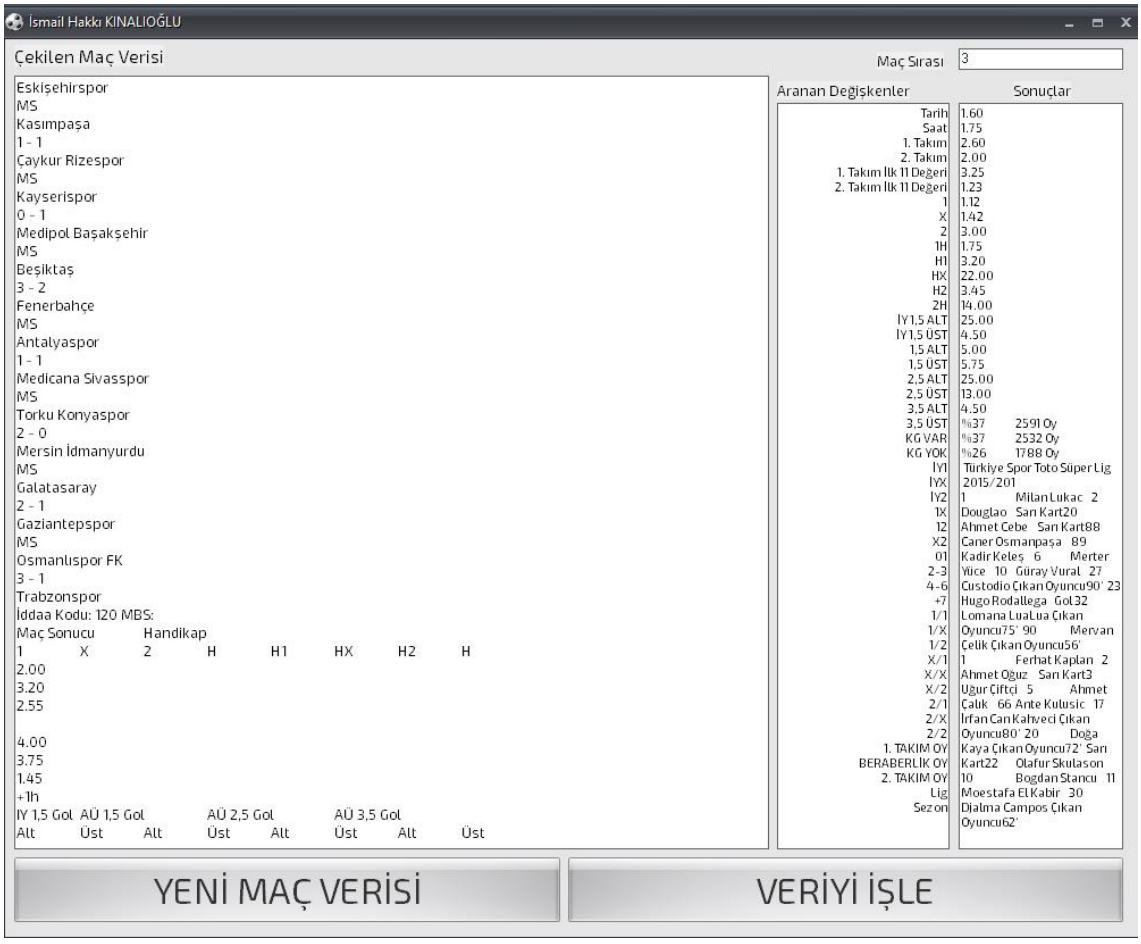

CSVLoader
data Set
ClassAssigner
ClassifierPerformanceEvaluator7
threshold Data
Model Performance Chart
image
ImageViewer
data Set
batch Classified text
image
test Set
training Set
TrainTestSplitMaker
Logistic2
TextViewer
text
TextSaver
ImageSaver
Holdout Ayırma
Eğitim oranı:%80
TrainTestSplitMaker options
About
A step that randomly splits incoming data into a training and test set
More
Training percentage
80
Random seed
1
Preserve instance order
False
OK
Cancel


CSVLoader
data Set
ClassAssigner
ClassifierPerformanceEvaluator7
threshold Data
Model Performance Chart
image
ImageViewer
data Set
batch Classified text
image
test Set
training Set
CrossValidationFoldMaker
Logistic2
TextViewer
text
TextSaver
ImageSaver
Çapraz geçerlilik
kullanımı
CrossValidationFoldMaker options
About
A Step that creates stratified cross-validation folds from incoming data
More
Number of folds
10
Preserve instances order
False
Random seed
1
OK
Cancel


CSVLoader
Train/Test SplitMaker
Attribute Selection
Numeric To Nominal
ClassAssigner
Logistic
SMO
IBk
RandomForest
NaiveBayes
MultilayerPerceptron
ClassifierPerformanceEvaluator
ClassifierPerformanceEvaluator2
ClassifierPerformanceEvaluator3
ClassifierPerformanceEvaluator4
ClassifierPerformanceEvaluator5
ClassifierPerformanceEvaluator6
ImageSaver
ImageViewer
ModelPerformanceChart
TextViewer
TextSaver
Özellik seçimi yapılan veri holdout yöntemi ile eğitim ve test diye ayrılıyor ardından 6 farklı sınıflandırma yöntemi ile sınıflandırılıyor ve ulaşılan değerle txt olarak grafikler ise png olarak kaydediliyor.

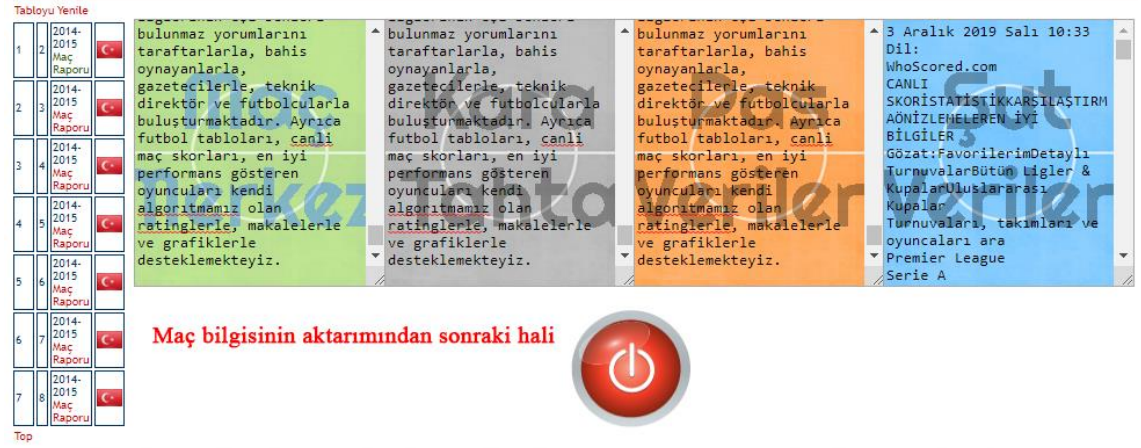

Veri yükleme aracı
CSVLoader2
Eğitim ve test veri ayırma aracı
TrainTest SplitMaker2
Eğitim ve test verisini çapraz geçerlilikle ayırma aracı
CrossValidation FoldMaker
Özellik seçimi filtresi aracı
Attribute Selection2
Nümerik değişkenleri nominal değişkenlere döndürme filtresi aracı
Numeric ToNominal2
Sınıfladırıcı performansı değerlendirme aracı
Classifier PerformanceEvaluator7
Yapay Sinir Ağları
Multilayer Perceptron2
Lojistik Regresyon
Logistic2
Destek vektör makinesi
SMO2
K-en yakın komşu
IBk2
Rasgele orman
Random Forest2
Sade bayes
NaiveBayes2
Model performans grafikleri
ModelPerformance Chart2
Grafik görüntüleyici
ImageViewer2
Grafik kaydedici
ImageSaver2
Metinsel sonuç gösterici
TextViewer2
Metin kaydedici
TextSaver2
CSV kaydedici
CSVSaver

EK-3: Maçlara dair verilerin çekilmesi için geliştirilen araçlar - 1



EK-4: Maçlara dair verilerin çekilmesi için geliştirilen araçlar - 2





2014.8.29 tarihinde Balikesirspor ve Akhisarspor takımları arasında oynanan müsabakaya ait veriler		
	Balikesirspor 1ayvaz,77Akdemir,20Kulusic,4Copalho,39Akyuz,13Santos,8Cavikar,25Alanzinho,19Yildirim,11Dapa,9Unal	Akhisarspor 44Yuncal,20Dapa,40Dougl8o,15Demirok,22Vural,10Bzar,5Ktsa,8Yuca,8Bojoun,28Ktbazonkiza,11Gekas
Maç Merkezi	6.49,10.0,1.5,4.41.6,507.66.310.206.7.9.20.14.45.12.46.17.9.19.33.58.21.23.3.33.7	6.82.11.1.4.5.2.58.4.634.79.435.345.6.14.22.9.64.14.54.9.17.18.27.67.25.8.2.50.11
Kara Tahta	10.310.21.33.23.21.11.5.19.12.507.16.1.2.2.0	11.435.26.27.8.25.13.3.27.14.634.24.0.0.1.1
Şut Verileri	1.1.5.0.4.0.0.3.7.5.0.5.0.0.6.3.1.0	2.4.5.1.2.0.1.5.5.8.0.3.0.0.5.5.1.0
Pas Verileri	11.24.3.0.15.7.67.287.24.320.34.310.248.94.177.165.44.190.110	23.17.2.1.20.6.88.394.73.409.29.453.335.147.253.227.113.204.165



Butona basıldıktan sonra aşağıdaki görüntü oluşur ve veriler veri tabanına kaydedilir

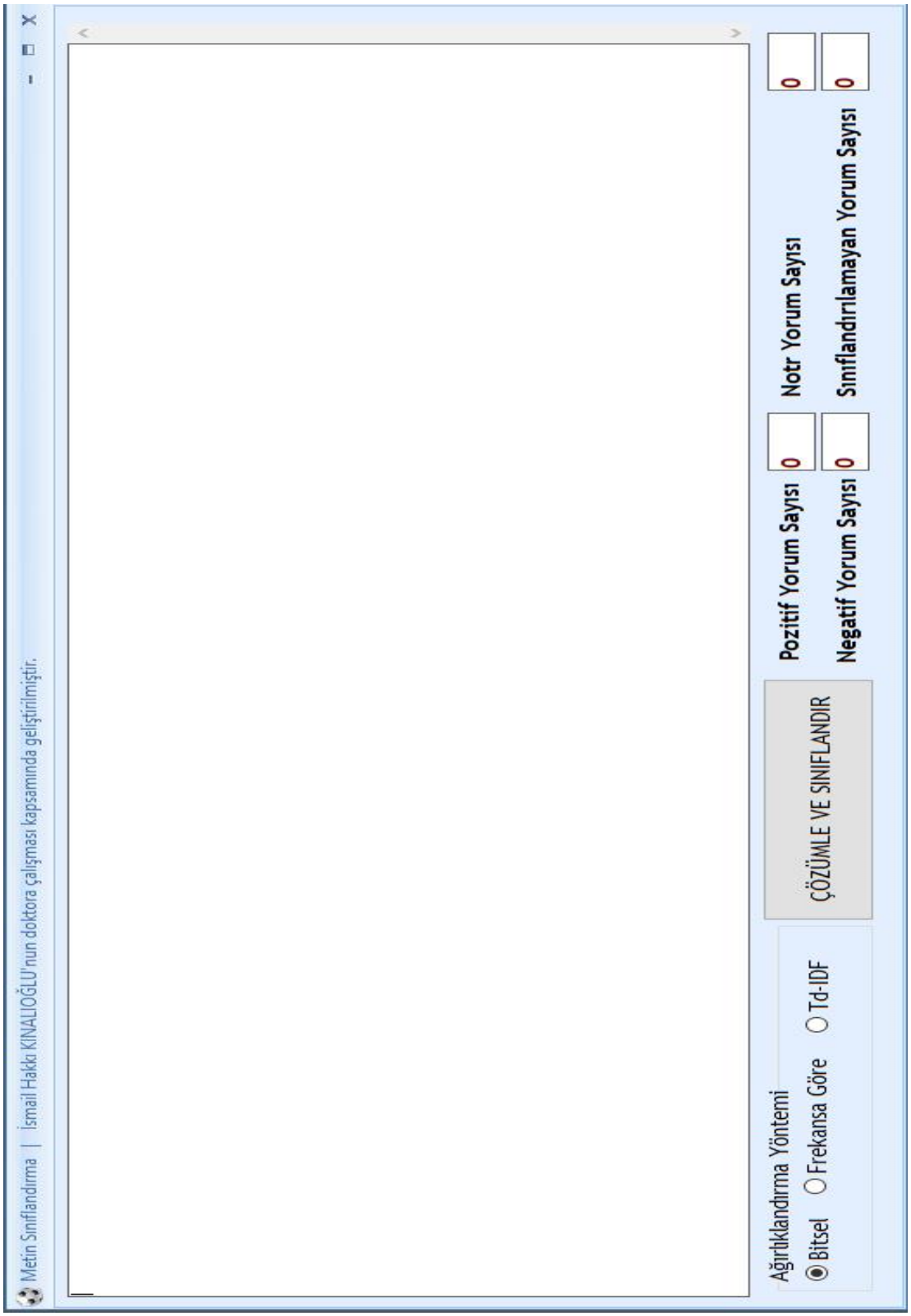
MAÇ MERKEZİ		
Rating	6.49	6.82
Toplam şut	10	11
Direk	0	1
İsabetli şut	1	4
İsabetsiz şut	5	5
Engellenmiş şut	4	2
Topa Sahip Olma%	41.6	58.4
Topla buluşma	507	634
Pas%	66	79
Toplam pas	310	435
Başarılı pas	206	345
Kilit pas	7	6
Başarılı Adam geçme	9	14
Adam geçme denemesi	20	22
Çalım yeme	14	9
Adam geçme%	45	64
Hava topu kazanma	12	14
Hava topu kazanma%	46	54
Ofansif hava topu	17	9
Defansif hava topu	9	17
Başarılı top çalma	19	18
Top çalma denemesi	33	27
Top çalma%	58	67
Uzaklaştırma	21	25
Top kapma	23	8
Korner	3	2
Korner isabet%	33	50
Top kaptırma	7	11

KARA TAHTA		
Şut	10	11
Pas	310	435
Adam geçme	21	26
Top çalma denemesi	33	27
Top kapma	23	8
Uzaklaştırma	21	25
Engelleme	11	13
Ofsayt	5	3
Faul	19	27
Hava topu	12	14
Topla buluşma	507	634
Top kaybı	16	24
Hata	1	0
Kurtarış	2	0
Orta Yakalama	2	1
Yumruk	0	1

ŞUT VERİLERİ		
Gol	1	2
İsabetli şut	1	4
İsabetsiz şut	5	5
Direk	0	1
Engellenmiş	4	2
Kendi	0	0
Altı pas	0	1
Ceza sahası	3	5
Ceza sahası dışında	7	5
Açık oyun	5	8
Kontra atak	0	0
Duran top	5	3
Penaltı	0	0
Kendi kalesine gol	0	0
Sağ ayak	6	5
Sol ayak	3	5
Kafa	1	1
Diğer vücut bölgeleri	0	0

PAS VERİLERİ		
Orta	11	23
Frikik	24	17
Korner	3	2
Ara pas	0	1
Taç	15	20
Kilit pas	7	6
Uzun	57	88
Kısa	287	394
Havalandırarak	24	73
Yer	320	409
Kafa	34	29
Ayak	310	453
İleri	248	335
Geri	94	147
Sol	177	253
Sağ	165	227
Defansif üçte bir	44	113
Orta üçte bir	190	204
Son üçte bir	110	165

EK-5: Sosyal medyadan çekilen taraftar görüşleri üzerinde duygu analizi gerçekleştirmek adına geliştirilen uygulamanın ara yüzü



ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı	**:**	İsmail Hakkı KINALIOĞLU
Uyruğu	**:**	T.C.
Doğum Yeri ve Tarihi	**:**	Trabzon/Araklı – 15.03.1986
Telefon	**:**	+905069947961
Faks	**:**	-
E-Posta	**:**	ismailhakkikinali@gmail.com
Web Sitesi	**:**	www.netderslerim.com

EĞİTİM

Derece		**Adı**	**İl**	**Bitirme Yılı**
Lise	:	Trbzon Lisesi	Trabzon	2003
Üniversite	:	Karadeniz Teknik Üniversitesi	Trabzon	2007
Yüksek Lisans	:	Selçuk Üniversitesi	Konya	2012

İŞ DENEYİMLERİ

Yıl	**Kurum**	**Görevi**
2008-Çalışıyor	Uşak Üniversitesi	Öğretim Görevlisi

UZMANLIK ALANI

Yapay Zeka, Yapay Öğrenme, Veri Madenciliği, İstatistiksel Analiz, Veri Görselleştirme, İstatistiksel Yazılımlar, Web ve Masaüstü Programlama, Web Tasarım, Bitmap ve Vektör Tabanlı Tasarım, Bilgisayar Destekli Animasyon, 3D Çizim ve Modelleme, Ses ve Görüntü Kurgu, Bilgisayar Bilimleri, Bilgisayar Öğretimi, Bilişim Teknolojileri, Öğretim Teknolojileri, E-Öğrenme, Uzaktan Eğitim

YABANCI DİLLER

İngilizce

BELİRTMEK İSTEĞİNİZ DİĞER ÖZELLİKLER

YAYINLAR

Kınalıoğlu İ.H, Karataş A., 2018, Sentiment analysis on user comments in video sharing networks, 4th International Researchers, Statisticians and Young Statisticians Congress, İzmir/Turkey, Özet Bildiri

Kınalıoğlu İ.H, Kuş C., Kınacı İ., 2018, A Sentiment Analysis On The Social Media Posts Of Turkish Football Fans, 4th International Researchers, Statisticians And Young Statisticians Congress, İzmir/Turkey, Özet Bildiri

Kınalıoğlu İ.H, Kuş C., Kınacı İ., 2018, A Hybrid Classification Method on the Prediction of Football Matches Result, 4th International Researchers, Statisticians And Young Statisticians Congress, İzmir/Turkey, Özet Bildiri

Karataş A., Kınalıoğlu İ.H., 2018, Köy Okullarında Çalışan Sınıf Öğretmenlerinin Sorunları, Usak University Journal of Social Sciences, Uluslararası Hakemli Makale

Gündüz C., Erbey A., Kınalıoğlu İ.H., 2018, Developing A Pipeline for Cleaning Twitter Data from Spam, Uluslararası, International Conference on Data Science and Applications, Yalova/Turkey, Tam Metin Bildiri

Karataş A. Kınalıoğlu İ.H., 2018, Mesleklerinin İlk Yılında Köy Okullarında Çalışan Sınıf Öğretmenlerinin Sorunları, Uluslararası, 17. Uluslararası Sınıf Öğretmenliği Sempozyumu, Ankara/Turkey, Özet Bildiri

Kınalıoğlu İ.H, Kuş C., Kınacı İ., 2017, Comparison of Artificial Neural Network and Data Mining Techniques for Predictıon of 2017 Uefa Champions League, Third International Researchers, Statisticians, and Young Statisticians Congress, Konya/Turkey, Özet Bildiri

Kınalıoğlu İ.H, Kuş C., Kınacı İ., 2016, A Comparison Study on Classification Performance of Multinomial Logistic Regression Analysis And Artificial Neural Network, Uluslararası, Second International Researchers, Statisticians, and Young Statisticians Congress, Ankara/Turkey, Özet Bildiri

Kınalıoğlu İ.H., 2011, Uzaktan Eğitim Sisteminde Öğrenci Başarısının Ölçülmesinde Karşılaşılan Güçlükler ve Çözüm Önerileri, Akademik Bilişim 2011, Malatya/Turkey, Tam Metin Bildiri

Karaman M.K., Kınalıoğlu İ.H., Akgün E., 2009, İlköğretim Bilişim Teknolojileri I II II Basamak Ders Kitaplarındaki Geleneksel Toplumsal Cinsiyet Rolleri ve Cinsiyet Ayrımcılığı, Uluslararası, 3. Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu, Trabzon/Turkey, Tam Metin Bildiri

PROJELER

Görme Engelli Bireylerin Uzaktan Eğitim Sistemi ile Mesleki Eğitimi (Vocational Education of Visually Impaired People via Distance Education), Hayatboyu Öğrenme Programı, Leonardo da Vinci Mesleki Eğitim Programı Yenilik Transferi Projesi, Araştırmacı (Başlangıç – Bitiş Tarihi: 2010 – 2012),

Bilgisayarla Görünürlük Kazanma, Avrupa Toplulukları Türkiye Katılım Öncesi Mali Yardım Aracı (IPA) Genç İstihdamının Desteklenmesi Hibe Programı, Eğitmen (Başlangıç – Bitiş Tarihi: 2010 – 2011),

Uşak Üniversitesi Eğitim Fakültesi Öğrencilerine Verilecek Olan Ortak Derslerin Harmanlanmış Öğrenme Yaklaşımı İle Verilmesinde Hangi Harmanlama Düzeyinin Öğrenci Başarışı Açısından Daha Uygun Olduğunun Belirlenmesi, Uşak Üniversitesi Bilimsel Araştırma Projesi, Araştırmacı (Başlangıç – Bitiş Tarihi: 2010 – 2011)