

T.C.
SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

BSM 498 BİTİRME ÇALIŞMASI

**MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE MEME
KANSERİ VERİLERİNİN DEĞERLENDİRİLMESİ**

G191210373 – Aysun ÇAĞ YILMAZKULAS

Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ
Tez Danışmanı : Doç. Dr. Nilüfer YURTAY

2020-2021 Bahar Dönemi

T.C.
SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

**MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE MEME
KANSERİ VERİLERİNİN DEĞERLENDİRİLMESİ**

BSM 498 - BİTİRME ÇALIŞMASI

Aysun CAĞ YILMAZKULAŞ

Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ

**Bu tez .. / .. / ... tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile
kabul edilmiştir.**

.....
Jüri Başkanı

.....
Üye

.....
Üye

ÖNSÖZ

Meme kanseri, sıklıkla kadınlar arasında görülen ve meme hücrelerinde başlayan bir kanser türündür. Erken teşhis ve doğru tedavi meme kanseri hastalarının hayatı kalma oranını artıtabilme açısından son derece önemlidir.

Makine öğrenimi, bilgisayarların algılayıcı verisi ya da veritabanları gibi veri türlerine dayalı öğrenimini olanaklı kıyan algoritmaların tasarım ve geliştirme süreçlerini konu edinen bir bilim dalıdır. Bu bilim dalı, verilere erişebilen ve öğrenciklerini kendileri için kullanabilen bilgisayar programlarının geliştirilmesine odaklanır. Temel amaç, makinelere, insan duyularına benzer duyularla veri toplamak için bilgisayarlı istihbarat araçları kullanarak toplanan verileri işleme yeteneği sağlamak ve ardından insanlarla aynı düzeyde tahminler yapmak ve kararlar almaktır. Doğal dil işleme, bilişsel hesaplama, bilgi sunumu, görüntü işleme, örüntü tanıma gibi birçok konuya odaklanmış ve sağlık, finans, pazarlama, kalite kontrol, bilgisayar ağları gibi birçok alanda uygulamaları bulunmaktadır.

Bu çalışmada, meme kanserin teşhisini için makine öğrenmesi algoritmalarından K-En Yakın Komşuluk algoritması kullanılmıştır. Çalışmanın amacı, ilgili veri setinin eğitilerek kanser hücresinin tanısı (İyi huylu – Kötü huylu) hakkında tahmin yürütülmesidir. Bu süreçte hangi makine öğrenmesi algoritmaları kullanılması gereği de dikkate alınmıştır. Sonuç olarak temel sınıflandırıcılarından KNN sınıflandırma yönteminin veri setine uygun bir algoritma olduğu belirlenerek bu yöntem uygulanmış, gerekli analizler yapılarak sonuç değerlendirilmiştir.

İÇİNDEKİLER

ÖNSÖZ	iii
İÇİNDEKİLER	iv
SİMGELER VE KISALTMALAR LİSTESİ	vi
ŞEKİLLER LİSTESİ	vii
TABLOLAR LİSTESİ.....	viii
ÖZET.....	ix
BÖLÜM 1. GİRİŞ	10
1.1. Meme Kanseri Nedir?.....	10
1.2. Meme Kanseri Teşhisi ve Tanı Yöntemleri.....	10
1.3. İnce İğne Biyopsisi ile Elde Edilen Veriler	11
BÖLÜM 2. MEME KANSERİ VERİ SETİNİN İNCELENMESİ.....	12
2.1. Veri Setindeki Özellikler	12
2.2. Veri Seti İncelenmesinde Kullanılan Kütüphaneler	14
2.3. Tanı Özelliğinin İncelenmesi ve Veri Seti Üzerine Genel Bilgi	15
2.4. Keşifsel Veri Analizi (EDA)	17
2.4.1. Korelasyon Grafiği	17
2.4.2. Tanı Özelliği ile Diğer Özelliklerin Korelasyon Grafiği	18
2.4.3. Box Plot	19
2.4.4. Pair Plot.....	20
2.4.5. Aykırı Değerlerin Tespiti.....	21
2.5. Veri Setinin Eğitim – Test Olarak Ayrılması	22
2.6. Standardizasyon.....	24
BÖLÜM 3. K-EN YAKIN KOMŞULUK ALGORİTMASI	26
3.1. K-En Yakın Komşuluk Algoritması Nedir?	26
3.2. KNN Algoritmasının Adımları.....	27
3.3. KNN Algoritmasının Modelde Uygulanması.....	28
3.4. KNN En Uygun Parametreleri Bulma	29

BÖLÜM 4. TEMEL BİLEŞEN ANALİZİ VE KOMŞULUK BİLEŞENLERİ ANALİZİ İLE SONUÇLARIN DEĞERLENDİRİLMESİ	31
4.1. Temel Bileşen Analizi (PCA).....	31
4.2. Komşuluk Bileşenleri Analizi (NCA)	32
BÖLÜM 5. SONUÇ VE ÖNERİLER	35
KAYNAKLAR	36
ÖZGEÇMIŞ	37
BSM 498 BİTİRME ÇALIŞMASI DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI	38

SİMGELER VE KISALTMALAR LİSTESİ

FNA	: İnce iğne aspiratı
M	: Kötü huylu kanser hücresi
B	: İyi huylu kanser hücresi
EDA	: Keşifsel veri analizi
LOF	: Yerel aykırı değer faktörü
μ	: Ortalama
σ	: Standart sapma
x	: Değişken değeri
KNN	: K-En yakın komşuluk
PCA	: Temel bileşen analizi
NCA	: Komşuluk bileşen analizi

ŞEKİLLER LİSTESİ

Şekil 2.1. Veri Seti Özelliklerinin Genel Bilgileri-1	13
Şekil 2.2. Veri Seti Özelliklerinin Genel Bilgileri-2	13
Şekil 2.3. Tanımlanan Kütüphaneleri Gösteren Python Kodu	14
Şekil 2.4. Tanı Özelliği Genel Dağılım Grafiği	15
Şekil 2.5. Veri Seti Korelasyon Grafiği	17
Şekil 2.6. Tanı Özelliğinin Diğer Özelliklerle İlişkisini Gösteren Grafik	18
Şekil 2.7. Box Plot Grafiği	19
Şekil 2.8. Pair Plot Grafiği	20
Şekil 2.9. Yarıçap Verilerindeki Aykırı Olabilecek Durumların Grafiği.....	21
Şekil 2.10. Yarıçap Verilerindeki Aykırı Değerlerin Grafiği	22
Şekil 2.11. Veri Seti Ayırma İşlemi Gösteren Python Kodu	23
Şekil 2.12. Standardizasyon İşlemi Ardından Box Plot Grafiği	25
Şekil 3.1. KNN Örnek Çizimi	26
Şekil 3.2. KNN Adımları	27
Şekil 4.1. PCA Analizi Grafiği	31
Şekil 4.2. PCA 2-Sınıf için Sınıflandırma Haritası	32
Şekil 4.3. NCA Analizi Grafiği.....	33
Şekil 4.4. NCA 2-Sınıf için Sınıflandırma Haritası	33
Şekil 4.5 Test Veri Setindeki Yanlış Sınıflandırılan Değeri Gösteren Grafik	34

TABLOLAR LİSTESİ

Tablo 2.1. Düzenlenmiş Veri Tablosu	16
Tablo 2.2. Describe Metodu ile Veri Özelliklerini Gösteren Tablo.....	16
Tablo 2.3. Eğitim - Test Veri Setleri Tabloları	23

ÖZET

Anahtar kelimeler: Makine Öğrenmesi, Algoritma, Sınıflandırma.

Makine öğrenmesi, sistemlere otomatik olarak ve deneyimlerden öğrenme yeteneği sağlayan yapay zekâının (YZ) bir uygulamasıdır. Verilere erişebilen ve kendileri için öğrenciklerini kullanabilen bilgisayar programlarının geliştirilmesine odaklanır. Ele alınan problemi, o probleme ait verilere göre modelleyen bilgisayar algoritmalarını içeren makine öğrenmesi, son zamanlarda yoğun çalışılan alanlardan biri olduğu için, konu ile ilgili önerilmiş birçok yaklaşım ve algoritma mevcuttur. Bu yaklaşımların bir kısmı tahmin ve kestirim bir kısmı da sınıflandırma yapabilme yeteneğine sahiptir. Temel amaç, makinelere insanı duyulara benzer duyularla veri toplama ve daha sonra tahminleri yürütmek ve insanlarla aynı seviyede kararlar almak için bilgisayarlı zekâ araçlarını kullanarak toplanan verileri işleme becerisi sunmaktadır. Doğal dil işleme, bilişsel hesaplama, bilgi sunumu, görüntü işleme, örüntü tanıma gibi birçok konuya odaklanmıştır ve sağlık hizmetleri, finans sektörü, pazarlama, kalite kontrol, bilgisayar ağları gibi birçok alanda uygulaması mevcuttur. Bu çalışmada Türkiye'de makine öğrenmesi konusu ile alakalı sınıflandırma algoritmaları incelenmiştir. Çalışmanın amacı, uygun sınıflandırma algoritmalarını uygulayarak ilgili veri setindeki özelliklere göre doğru tahminlerin yapılmasını sağlamaktır.

BÖLÜM 1. GİRİŞ

Kanser, en önemli toplumsal sağlık sorunlarından biridir. Vücut içinde hücrelerin değişerek kontrollsüz şekilde büyümeye neden olan hastalıklar grubudur. Pek çok kanser hücresi tipi, zamanla adına tümör denilen yumru veya kitle şeklinde ve tümörün olduğu vücut bölümünün ismini almaktadır [1].

Bu çalışmada, meme kanserinin teşhisini için makine öğrenmesi algoritmaları kullanılması ve bu algoritmaların birbirlerine göre sınıflandırma performanslarının karşılaştırılması hedeflenmektedir. Çalışmanın amacı, ilgili veri setinin eğitilerek kanser hücresinin tanısı (İyi huylu – Kötü huylu) hakkında tahmin yürütülmesidir.

1.1. Meme Kanseri Nedir?

Meme kanseri kadınlarla görülen kanser tipleri arasında ilk sırada yer almaktadır. Batı ülkelerinde her 8- 9 kadından birinin yaşamı boyunca meme kanserine yakalandığı görülmektedir. Bu oran %10'dan daha fazladır. Sık görülmesi, erken evrelerde tedavi edilebilir olması, günümüz koşullarında tanınmasının olanaklı olması, meme kanserinin önemini artırmaktadır.

Meme kanseri, meme dokusundaki hücrelerden gelişen bir kanserdir. Meme dokusunun herhangi bir yerinden kaynaklanabilir. En sık görülen tipi; meme kanallarından kaynaklanan “duktal” kanser denen kanserlerdir. Süt üreten bezlerden köken alan “lobüler” kanserler de sık görülür. Ayrıca diğer dokulardan kaynaklanan daha nadir medüller, tübüller, müsinöz gibi tipleri de vardır [2].

1.2. Meme Kanseri Teşhisini ve Tanı Yöntemleri

Meme kanseri teşhisinde birçok teşhis yöntemi bulunmaktadır. Bunların birbirlerine karşı farklı üstünlükleri vardır ancak birkaçının birlikte yapılması ile erken teşhis artmaktadır. Bu yöntemlerden bazıları; kendi kendine muayene, mamografi (meme röntgen filmi), meme ultrasonografisi, memenin manyetik rezonans tekniği ile incelemesi ve biyopsidir.

Bahsedilen teşhis yöntemlerinden biyopsi, kanser şüphesi oluşturan bir durum varsa saptanan oluşumdan hücre örnekleri alınarak mikroskop altında inceleme yapılarak kesin teşhisin konulmasıdır [3].

1.3. İnce İğne Biyopsisi ile Elde Edilen Veriler

Saptanan oluşum el ile hissedilebiliyorsa bir enjektör iğnesi hissedilen oluşuma birkaç defa saplanarak oluşumdan hücre örnekleri alınır ve bu hücreler mikroskop altında incelenir.

İnce iğne aspirasyon biyopsisi sadece birkaç dakika süren basit bir operasyondur. Bu işlemde ince bir iğne memeye sokularak sıvı ve hücre örneği alınır. Kist denilen, içi sıvı dolu lezyonlarda, sıvıyı boşaltmak ve örnek almak için kullanılır.

BÖLÜM 2. MEME KANSERİ VERİ SETİNİN İNCELENMESİ

Meme kanseri teşhisi bir meme kitlesinin ince iğne aspiratının (FNA) dijitalleştirilmiş bir görüntüsünden hesaplanır. Bu görüntülerde bulunan hücre çekirdeklerinin özelliklerini dikkate alınarak kanser hüresinin iyi huylu ya da kötü huylu olduğu hakkında bir sonuca varılabilir. Bu çalışmada incelenen veri setinde hücrelerin çekirdek özellikleri dikkate alınmıştır [4].

2.1. Veri Setindeki Özellikler

Veri setinde bulunan hücre çekirdeği öznitelik bilgileri aşağıda belirtilmiştir:

1. Kimlik Numarası
2. Tanı (M=Kötü Huylu B=İyi Huylu)

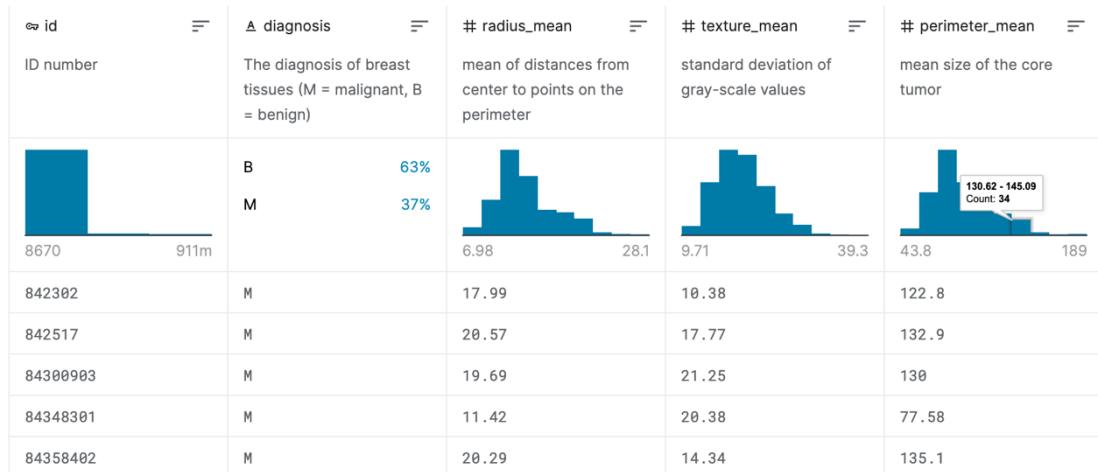
Her hücre çekirdeği için on gerçek değerli özellik hesaplanmıştır.

- a. Yarıçap (Merkezden çevre üzerindeki noktalara olan mesafelerin ortalaması)
- b. Doku (Gri ölçekli değerlerin standart sapması)
- c. Çevre (Çekirdek tümörün boyutu)
- d. Alan (Çekirdek tümörün alanı)
- e. Pürüzsüzlük (Yarıçap uzunluklarındaki yerel değişim)
- f. Yoğunluk ($\text{Çevre}^2 / \text{Alan} - 1.0$)
- g. İçbükeylik (Konturun içbükey kısımlarının şiddeti)
- h. İçbükey Noktalar (Konturun içbükey kısımlarının sayısı)
- i. Simetri
- j. Fraktal Boyut (Kıyı yaklaşımı – 1)

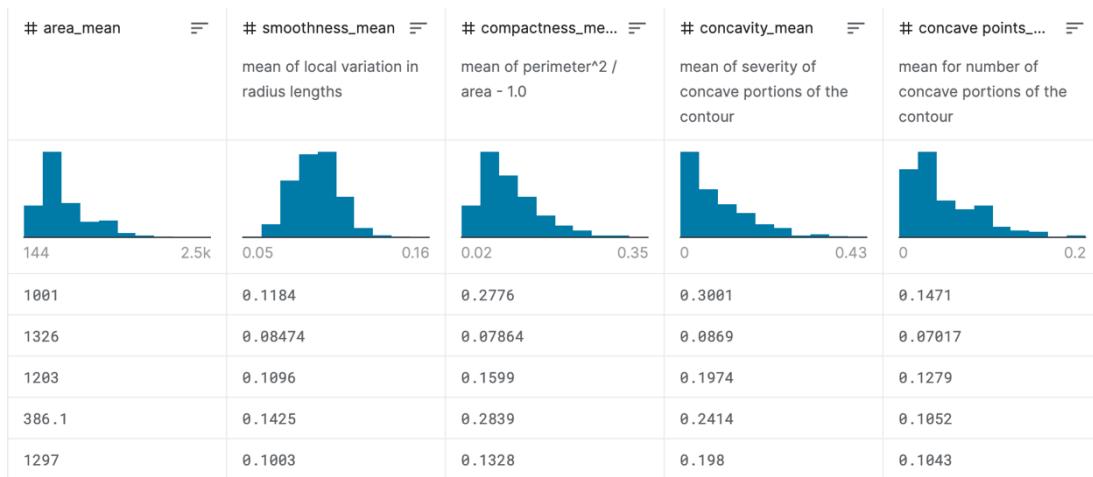
Her bir görüntü için Ortalama, Standart Hata ve en kötü veya en büyük değerler hesaplanarak öznitelik bilgileri ile toplamda 32 özellik belirlenmiştir. Tüm özellik değerleri dört anlamlı basamakla yeniden kodlanır.

Veri seti içerisinde eksik özellik (missing value) bulunmamaktadır.

Sınıf dağılımı 357 iyi huylu, 212 kötü huylu şeklindedir.



Şekil 2.1. Veri Seti Özelliklerinin Genel Bilgileri-1



Şekil 2.2. Veri Seti Özelliklerinin Genel Bilgileri-2

2.2. Veri Seti İncelenmesinde Kullanılan Kütüphaneler

Projede incelenen veri setinde istenilen çalışmalar Python programlama dili ile yapılmıştır. Bu programlama diline ait uygun kütüphaneler veri seti incelemesinde kullanılmıştır. İlgili kütüphaneler ve kullanım alanları aşağıda belirtilmiştir.

- Pandas kütüphanesi ve bu kütüphaneye ait özellikler veri setinin program tarafından okunmasında ve verinin analizinde kullanılmıştır.
- Numpy kütüphanesi veri seti üzerindeki matematiksel ve vektörel işlemlerin yapılabilmesi adına kullanılmıştır.
- Seaborn kütüphanesi veri seti üzerindeki işlemlerin görselleştirilebilmesi amacıyla kullanılmıştır.
- Matplotlib kütüphanesi görselleştirme işlemleri ve sonuçların görsel olarak değerlendirilebilmesi amacıyla kullanılmıştır.
- Sklearn kütüphanesi standardizasyon, veri setinin ayrıştırılması (test-train veri setleri), başarı belirleme ve nerede hatalar yapıldığı, K-en yakın komşuluk sınıflandırılmasında en iyi parametrelerin belirlenmesi ve PCA işlemlerinde kullanılmıştır.

```
"""
Aysun CAG YILMAZKULAS
G191210373

@author: aysuncag
"""

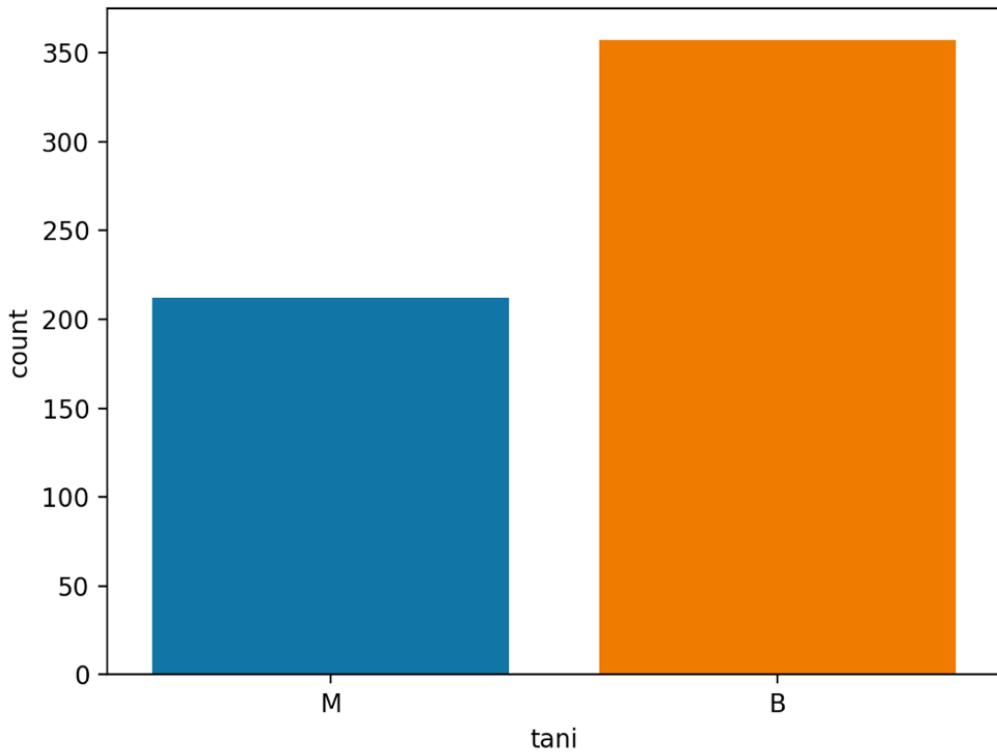
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib.colors import ListedColormap

from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.neighbors import KNeighborsClassifier, NeighborhoodComponentsAnalysis, LocalOutlierFactor
from sklearn.decomposition import PCA
```

Şekil 2.3. Tanımlanan Kütüphaneleri Gösteren Python Kodu

2.3. Tanı Özelliğinin İncelenmesi ve Veri Seti Üzerine Genel Bilgi

Veri seti üzerinde çalışırken öncelikle ihtiyaç duyulmayan özellikler veri setinden çıkarılmıştır. Ardından Diagnosis olarak tanımlanmış, hücrenin iyi-kötü huylu olduğunu belirten özellik Tanı olarak değiştirilmiştir. Burada tanı özelliğin dağılımı seaborn kütüphanesi kullanılarak görselleştirilmiştir.



Şekil 2.4. Tanı Özelliği Genel Dağılım Grafiği

Ayrıca bu özelliğin veri setinin görselleştirilmesi ve eğitimi işlemlerinde uygun bir şekilde kullanılabilmesi amacıyla M-B harfleri yerine 1-0 sayıları veri setine eklenmiştir. İlgili veri seti Tablo 2.1'de görülmektedir.

Tablo 2.1. Düzenlenmiş Veri Tablosu

Index	tani	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean
0	1	18.81	19.98	120.9	1102	0.08923	0.05884
1	1	20.58	22.14	134.7	1290	0.0909	0.1348
2	1	17.95	20.01	114.2	982	0.08402	0.06722
3	1	19	18.91	123.4	1138	0.08217	0.08028
4	1	21.71	17.25	140.9	1546	0.09384	0.08562
5	1	20.16	19.66	131.1	1274	0.0802	0.08564
6	1	19.19	15.94	126.3	1157	0.08694	0.1185
7	1	20.48	21.46	132.5	1306	0.08355	0.08348
8	0	12.54	18.07	79.42	491.9	0.07436	0.0265
9	1	17.01	20.26	109.7	904.3	0.08772	0.07304
10	0	13.01	22.22	82.01	526.4	0.06251	0.01938
11	0	17.85	13.23	114.6	992.1	0.07838	0.06217
12	0	14.61	15.69	92.68	664.9	0.07618	0.03515
13	0	14.97	19.76	95.5	690.2	0.08421	0.05352
14	0	16.84	19.46	108.4	880.2	0.07445	0.07223

Veri setindeki özelliklerin genel bilgilerini açıklamak amacıyla `describe=data.describe()` kodu kullanılmıştır. Bu kod ile Tablo 2.2'de görülen genel bilgiler elde edilmiştir.

Tablo 2.2. Describe Metodu ile Veri Özelliklerini Gösteren Tablo

Index	tani	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean
count	569	569	569	569	569	569	569
mean	0.372583	14.1273	19.2896	91.969	654.889	0.0963603	0.104341
std	0.483918	3.52405	4.30104	24.299	351.914	0.0140641	0.0528128
min	0	6.981	9.71	43.79	143.5	0.05263	0.01938
25%	0	11.7	16.17	75.17	420.3	0.08637	0.06492
50%	0	13.37	18.84	86.24	551.1	0.09587	0.09263
75%	1	15.78	21.8	104.1	782.7	0.1053	0.1304
max	1	28.11	39.28	188.5	2501	0.1634	0.3454

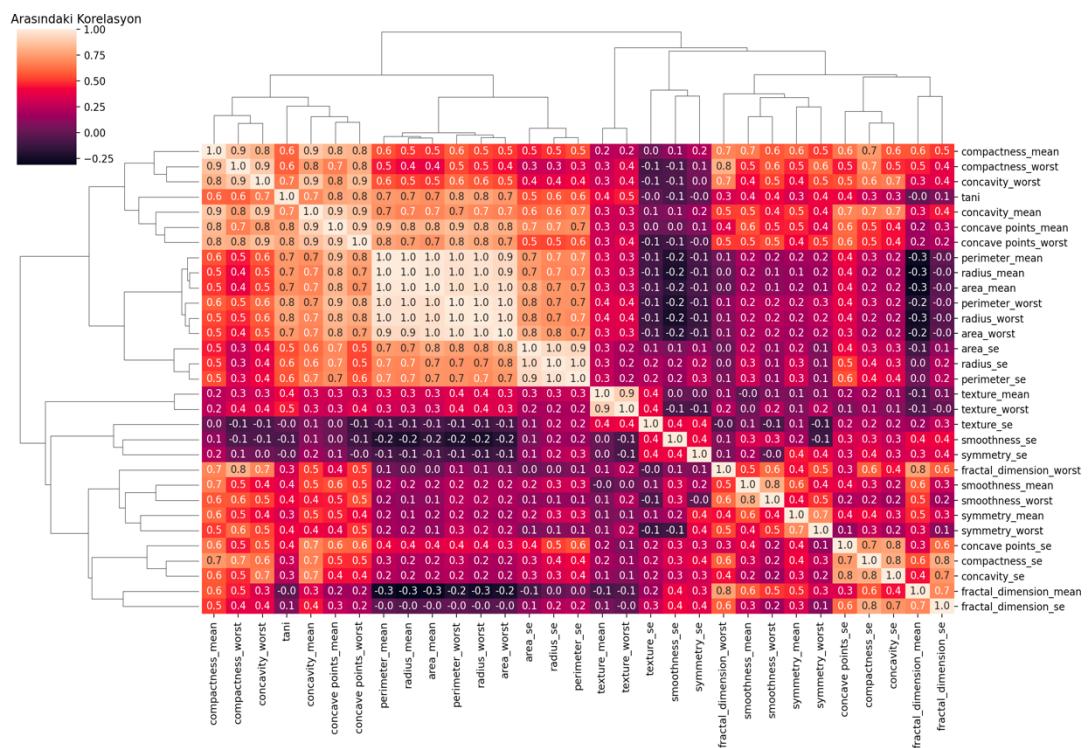
Tablo 2.2'deki bilgiler dikkate alındığında veriler arasında büyük farklar olduğu görülmüştür. Bu sebeple verilerin standardize edilmesi gerekliliği belirlenmiştir.

2.4. Keşifsel Veri Analizi (EDA)

İstatistikte Keşifsel Veri Analizi, genellikle istatistiksel grafikler ve diğer veri görselleştirme yöntemlerini kullanarak temel özelliklerini özetlemek için veri setlerini analiz etme yaklaşımıdır [4].

2.4.1. Korelasyon Grafiği

Veri setindeki özellikler arasındaki korelasyonu tanımlayabilmek ve analiz etmek amacıyla korelasyon grafiği oluşturulur.



Şekil 2.5. Veri Seti Korelasyon Grafiği

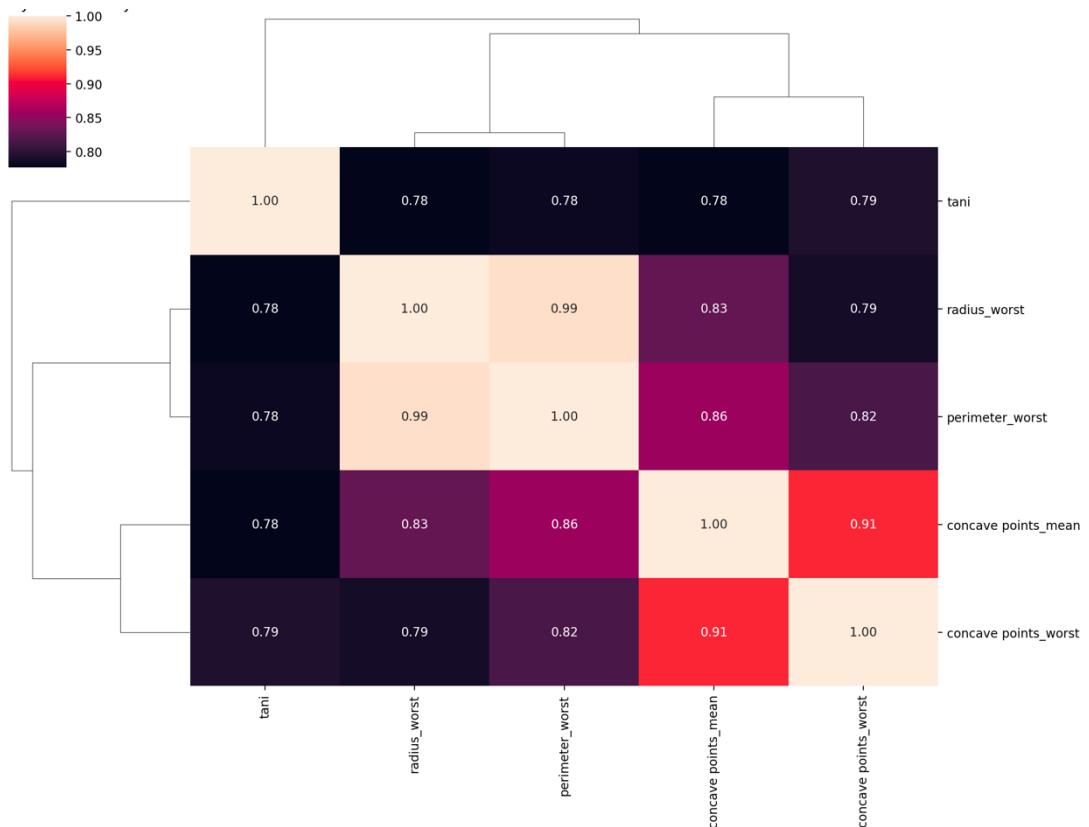
Korelasyon grafiğinde bulunan değerler -1 ile 1 arasında değişmektedir. Burada korelasyon değeri 1 olan özellikler birbirleri ile pozitif ilişki içerisinde olduğu anlamına gelir. Yani bir özellik artarken diğeri de artmakta, azalırken diğeri de azalmakta demektir. Korelasyon değeri -1 olan özellikler ise birbirleri ile negatif

ilişki içerisindedir. Bu ise bir özelliğin değeri artarken diğerinin değeri azalıyor anlamındadır. Korelasyon değeri 0 olan özelliklerin ise bir ilişkileri yoktur.

Şekil 2.5 incelendiğinde birbirine en yakın ilişkide olan özellikler grafiğin kenarlarındaki ağaç yapısı ile görülebilmektedir. Eğitilecek makine öğrenmesi modelinde bu özelliklerin hepsinin kullanılması gerekli değildir.

2.4.2. Tanı Özelliği ile Diğer Özelliklerin Korelasyon Grafiği

Yukarıdaki grafikteki değerlerin yorumlanmasıındaki karmaşıklık sebebi ile veri setimizdeki özelliklerin, Tanı özelliği üzerindeki etkilerinin gözlemlenebilmesi amacıyla yeni bir grafik oluşturulmuştur. Şekil 2.6'da korelasyon değerleri 0.75 değerinin üzerinde olan özellikler incelenmiştir.

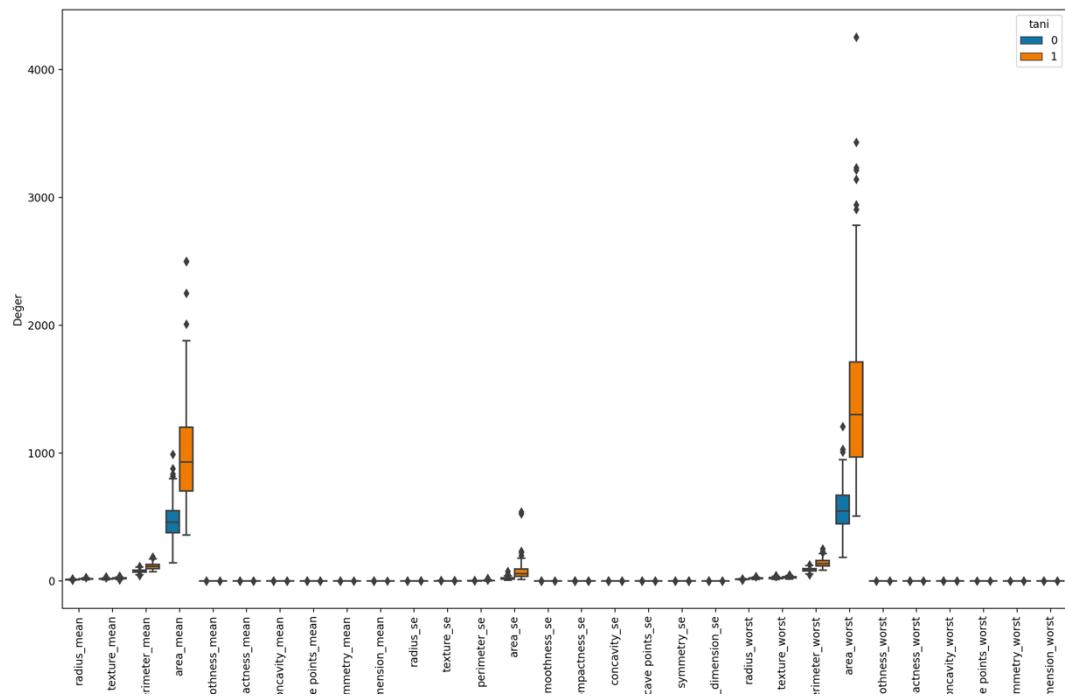


Şekil 2.6. Tanı Özelliğinin Diğer Özelliklerle İlişkisini Gösteren Grafik

Şekil 2.6 incelendiğinde Tanı özelliğinin grafikte belirlenen dört özellik ile yüksek ilişkide olduğu gözlemlenmektedir. Burada birbiri ile ilişkili özellikler varsa bunlar veri setinden çıkartılabilir.

2.4.3. Box Plot

Nümerik özellikler ile ilgili bilgiler elde edebilmek için görselleştirme tekniklerinden Box Plot tekniğini kullanabiliriz. Tanı özelliklerine göre ayrılarak özelliklerin dağılımını bu yöntem ile görebiliriz.

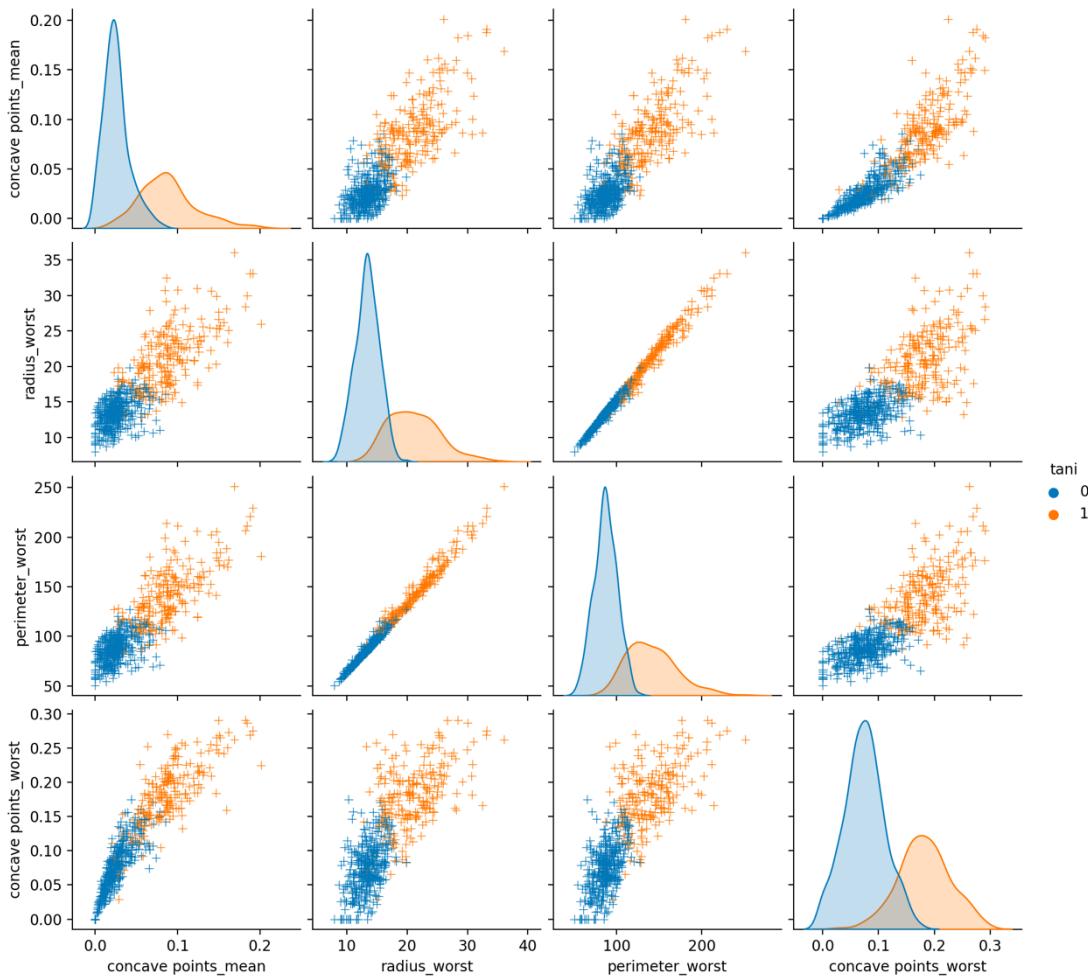


Şekil 2.7. Box Plot Grafiği

Şekil 2.7'deki Box Plot içerisindeki değerleri çok yüksek olduğu için grafikten anlam çıkarabilmek zordur. Bu sebeple standardizasyon işlemi yapıldıktan sonra tekrar görselleştirme işlemi yapılmalıdır.

2.4.4. Pair Plot

Pair Plot işlemi de Seaborn kütüphanesinin sunduğu bir görselleştirme tekniğidir. Bu işlem histogram kullanılarak ve Tanı özelliğine göre düzenlendiğinde Şekil 2.8 elde edilmektedir.



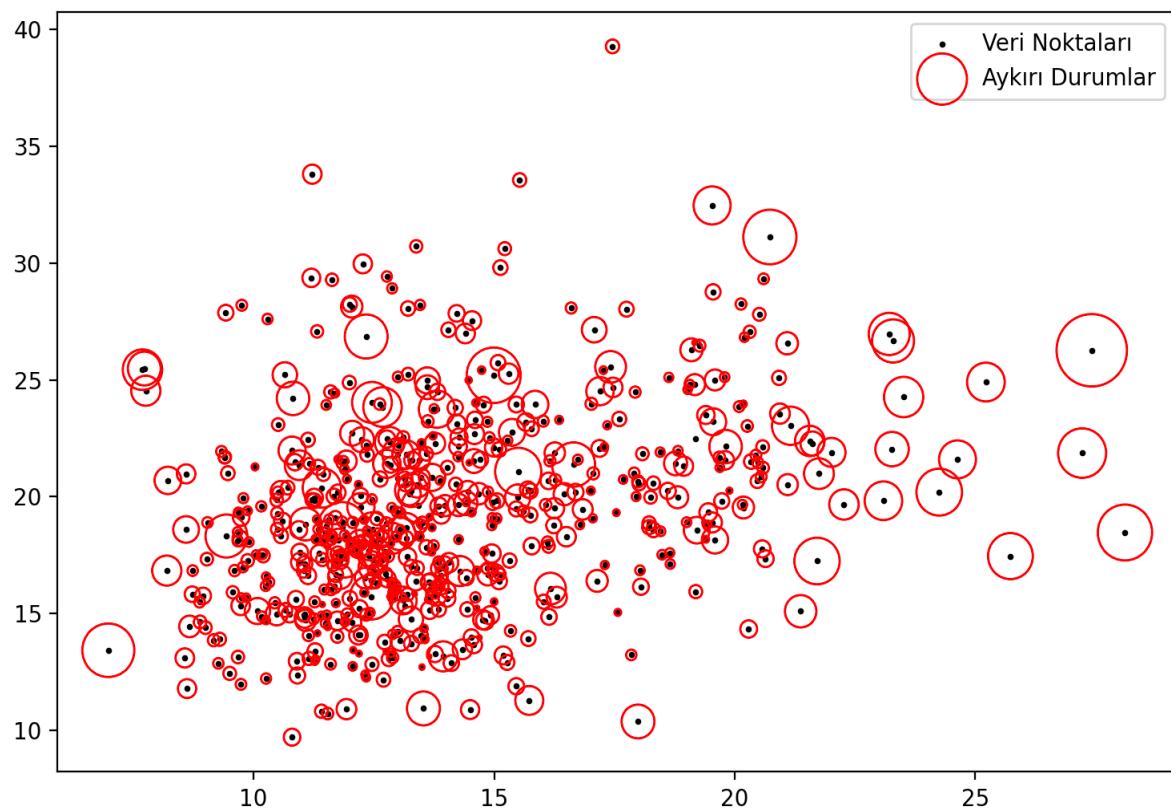
Şekil 2.8. Pair Plot Grafiği

Burada mavi ve turuncu ile gösterilen değerler Tanı özelliği içerisindeki iyi huylu ve kötü huylu değerleri göstermektedir. Görseller dikkate alınarak diğer özelliklere göre dağılım yorumlanabilmektedir. Özelliklerin dağılımında Gauss ya da pozitif çarpıklık olduğu görülmektedir. Pozitif çarpıklığın düzeltilebilmesi için aykırı değerlerin tespiti yapılmalıdır.

2.4.5. Aykırı Değerlerin Tespiti

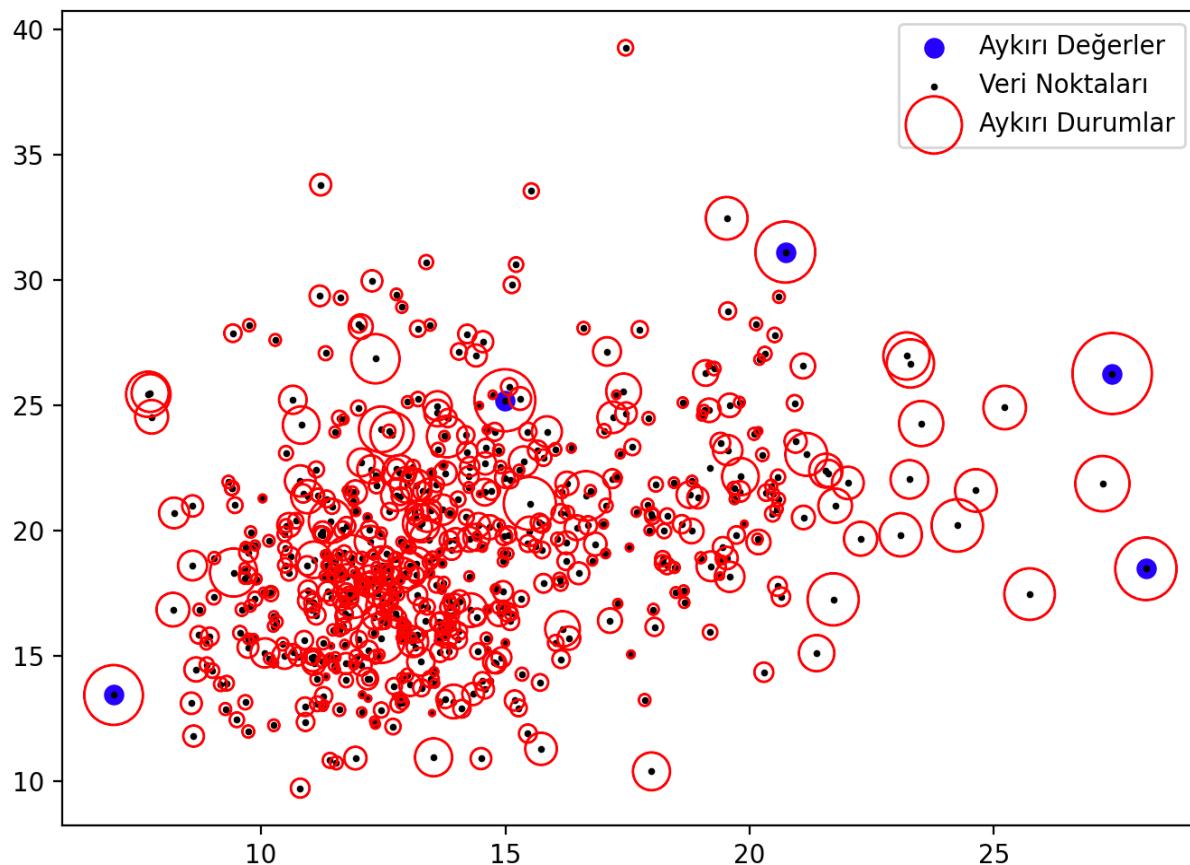
Aykırı değer, diğer gözlemlerden önemli ölçüde farklı bir veri noktasıdır. Ölçümlerdeki değişkenlige bağlı olabilir ya da deneysel hatayı gösterebilir. Gerektiğinde veri kümelerinden çıkarılır. Çünkü bir aykırı değer istatistiksel analizlerde ciddi sorunlara neden olabilir [5].

Veri setindeki aykırı değerlerin belirlenmesi için Yerel Aykırı Değer Faktörü (LOF) yöntemi kullanılacaktır. Bu yöntem, belirli bir veri noktasının komşularına göre yerel yoğunluk sapmasını hesaplayan bir anormallik algılama yöntemidir. Komşularından önemli ölçüde düşük yoğunluğa sahip örnekler aykırı değer olarak kabul edilebilir.



Şekil 2.9. Yarıçap Verilerindeki Aykırı Olabilecek Durumların Grafiği

Elde edilen grafikte eşik değeri belirleyerek aykırı değerler elde edilebilir. Eşik değeri 2 seçilerek grafiğekte yer alan yarıçap ortalamasını içeren verilerin aykırı değerleri Şekil 2.10'da gösterilmiştir.



Şekil 2.10. Yarıçap Verilerindeki Aykırı Değerlerin Grafiği

Şekil 2.10 incelendiğinde mavi ile gösterilen noktaların aykırı olduğu belirlenmektedir ve bu noktaların veri setinden çıkarılması gerekmektedir.

2.5. Veri Setinin Eğitim – Test Olarak Ayrılması

Makine öğrenmesinde model veri ile eğitilmektedir. Bu işlemde veriden model öğrenilir, başka veri setlerinde öğrenilen model kullanılır ve belirli sonuçlar sunulur. Bizler bu eğitimin ne kadar doğru ve sağlıklı olduğunu ise test verileriyle test ederiz ki modelin kullanılabilirliğine karar verelim.

```
# %% Train - Test Split
test_size=0.3
X_train, X_test, Y_train, Y_test=train_test_split(x,y,test_size=test_size,random_state=42)
```

Şekil 2.11. Veri Seti Ayırma İşlemi Gösteren Python Kodu

Veri seti ayırma işlemi için, *cross_validation* kütüphanesinin modülü olan *train_test_split()* modülüne iki parametre verilmiştir (Şekil 2.11). Bu parametreler (X ve y) veri kaynağı olarak neyin kullanılacağını göstermektedir. *test_size* parametresi ile test için ne kadar veri ayrılacağı belirtilmektedir. Modelimizde *test_size* 0.3 seçilmiştir ki bu, veri setinin %30'unun test veri seti olarak, %70'inin ise eğitim veri seti olarak ayrıldığını göstermektedir. Örneklem için de bir *random_state* değeri belirlenmiştir. Burada *random_state* her seferinde aynı ayrimın yapılmasını sağlayan parametredir ve sabitlenmiştir.

Tablo 2.3. Eğitim - Test Veri Setleri Tabloları

Index	radius mean	texture mean	perimeter mean	area mean	smoothness me	compactness me	concavity me	concave po
143	12.9	15.92	83.74	512.2	0.08677	0.09509	0.04894	0.0308
303	10.49	18.61	66.86	334.3	0.1068	0.06678	0.02297	0.0178
427	10.8	21.98	68.79	359.9	0.08801	0.05743	0.03614	0.0146
470	9.667	18.49	61.49	289.1	0.08946	0.06258	0.02948	0.0151
19	13.54	14.36	87.46	566.3	0.09779	0.08129	0.06664	0.0478
174	10.66	15.15	67.49	349.6	0.08792	0.04302	0	0
404	12.34	14.95	78.29	469.1	0.08682	0.04571	0.02109	0.0205
57	14.71	21.59	95.55	656.9	0.1137	0.1365	0.1293	0.0812
309	13.05	13.84	82.71	530.6	0.08352	0.03735	0.004559	0.0088
494	13.16	20.54	84.06	538.7	0.07335	0.05275	0.018	0.0125
403	12.94	16.17	83.18	507.6	0.09879	0.08836	0.03296	0.0239
386	12.21	14.09	78.78	462	0.08108	0.07823	0.06839	0.0253
530	11.75	17.56	75.89	422.9	0.1073	0.09713	0.05282	0.0444
514	15.05	19.07	97.26	701.9	0.09215	0.08597	0.07486	0.0433
367	12.21	18.02	78.31	458.4	0.09231	0.07175	0.04392	0.0202

Y_train - NumPy object array	
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	0



The screenshot shows two data structures in a Jupyter Notebook environment. At the top, there is a DataFrame titled "X_test - DataFrame". It has 17 rows and 10 columns. The columns are labeled: Index, radius mean, texture mean, perimeter mean, area mean, smoothness mean, compactness mean, concavity mean, concave points mean, and symmetry mean. The data values are mostly numerical, ranging from 11.57 to 98.73. Below the DataFrame is a NumPy object array titled "Y_test - NumPy object array". It has 9 rows and 1 column. The column is labeled "0" and contains binary values: 0, 0, 1, 0, 0, 0, 1, 0, 0. A vertical scroll bar is visible on the right side of the array.

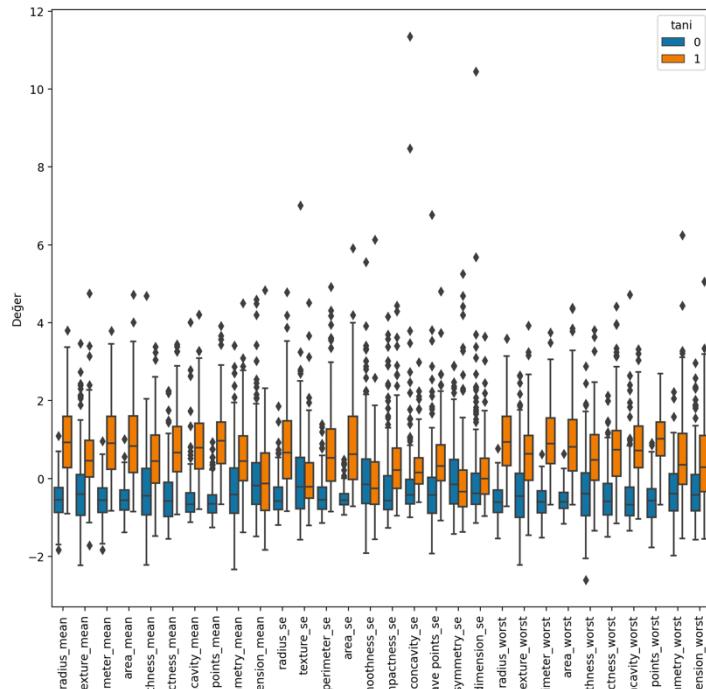
2.6. Standardizasyon

Değişkenin kendi içindeki bilgi ve varyans yapısını bozmadan, değerleri değiştirip belli bir formata sokup, bu formatta işlenmesine olanak sağlamak için veriyi standart hale getirme işlemi standardizasyon olarak adlandırılır. Bu işlem ile, değişken sütunlarının ortalama değeri 0 ve standart sapması 1 olacak şekilde standart normal dağılım oluşturulmaktadır.

$$z = \frac{x - \mu}{\sigma}$$

μ : ortalama, σ : standart sapma , x : değişken değeri

Yukarıda belirtilen formül kullanılarak z puanları belirlenir ve artık değişkenler yerine z puanları sütunlarda yer alır. Bu sayede olasılıklar üzerinden daha rahat analizler yapılabilmektedir.



Şekil 2.12. Standardizasyon İşlemi Ardından Box Plot Grafiği

Daha önce doğru bir şekilde yorumlayamadığımız Box Plot grafiğini standardizasyon işleminin ardından Şekil 2.12'de tekrar oluşturduğumuzda, daha iyi analiz edilebilir olduğu görülmektedir. Burada her bir özelliğe ait dağılım ile, aynı zamanda aykırı değerler de açıkça görülebilmektedir. Buna göre ileride model gerekli performansı sağlayamazsa bu aykırı değerlerin çıkarılabileceği yorumu yapılabilir.

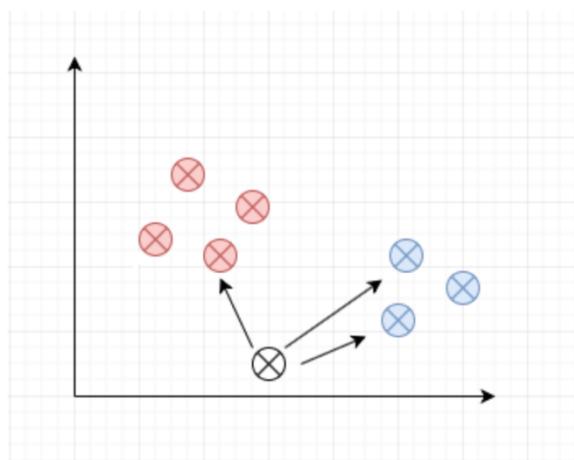
BÖLÜM 3. K-EN YAKIN KOMŞULUK ALGORİTMASI

3.1. K-En Yakın Komşuluk Algoritması Nedir?

K-en yakın komşuluk (KNN) algoritması, uygulaması kolay gözetimli öğrenme algoritmalarındandır. Hem sınıflandırma hem de regresyon problemlerinin çözümünde kullanılıyor olmakla birlikte, endüstride çoğunlukla sınıflandırma problemlerinin çözümünde kullanılmaktadır.

KNN algoritmaları, 1967 yılında T. M. Cover ve P. E. Hart tarafından önerilmiştir. Algoritma, sınıfları belli olan bir örnek kümesindeki verilerden yararlanılarak kullanılmaktadır. Örnek veri setine katılacak olan yeni verinin, mevcut verilere göre uzaklığı hesaplanıp, k sayıda yakın komşuluğuna bakılır. Uzaklık hesapları için genelde 3 tip uzaklık fonksiyonu kullanılmaktadır:

- “Euclidean” Uzaklık
- “Manhattan” Uzaklık
- “Minkowski” Uzaklığı’dır.



Şekil 3.1. KNN Örnek Çizimi

KNN; eski, basit ve gürültülü eğitim verilerine karşı dirençli olması sebebiyle en popüler makine öğrenme algoritmalarından biridir. Fakat bunun yanında dezavantajı da mevcuttur. Örneğin, uzaklık hesabı yaparken bütün durumları sakladığından, büyük veriler için kullanıldığında çok sayıda bellek alanına gereksinim duymaktadır.

3.2. KNN Algoritmasının Adımları

KNN algoritmasının adımları:

- İlk olarak k parametresi belirlenir. Bu parametre verilen bir noktaya en yakın komşuların sayısıdır. Örneğin: $k=2$ olsun. Bu durumda en yakın 2 komşuya göre sınıflandırma yapılacaktır.
- Örnek veri setine katılacak olan yeni verinin, mevcut verilere göre uzaklıği tek tek hesaplanır. İlgili uzaklık fonksiyonları yardımıyla.
- İlgili uzaklıklardan en yakın k komşu ele alınır. Öz nitelik değerlerine göre k komşu veya komşuların sınıfına atanır.
- Seçilen sınıf, tahmin edilmesi beklenen gözlem değerinin sınıfı olarak kabul edilir. Yani yeni veri etiketlenmiş (label) olur.



Şekil 3.2. KNN Adımları

3.3. KNN Algoritmasının Modelde Uygulanması

KNN algoritması, aykırı değerlere karşı duyarlı bir algoritma olması ve veri setimizde aykırı değerlerin önceki uygulamalarımızda çıkarılmış olması sebebiyle veri setimiz için uygundur.

KNN Algoritması aşağıdaki Python kodlarını kullanarak uygulanmıştır:

```
# %% KNN Method
knn=KNeighborsClassifier(n_neighbors=2)
knn.fit(X_train, Y_train)
y_pred=knn.predict(X_test)
cm=confusion_matrix(Y_test,y_pred)
acc=accuracy_score(Y_test,y_pred)
score=knn.score(X_test, Y_test)

print("Skor: ",score)
print("CM: ",cm)
print("KNN Accuracy: ",acc)
```

Şekil 0.1. KNN Algoritması için Uygulanan Python Kodu

Uygulanan kod sonucunda aşağıdaki skor ve doğruluk değerleri elde edilmiştir:

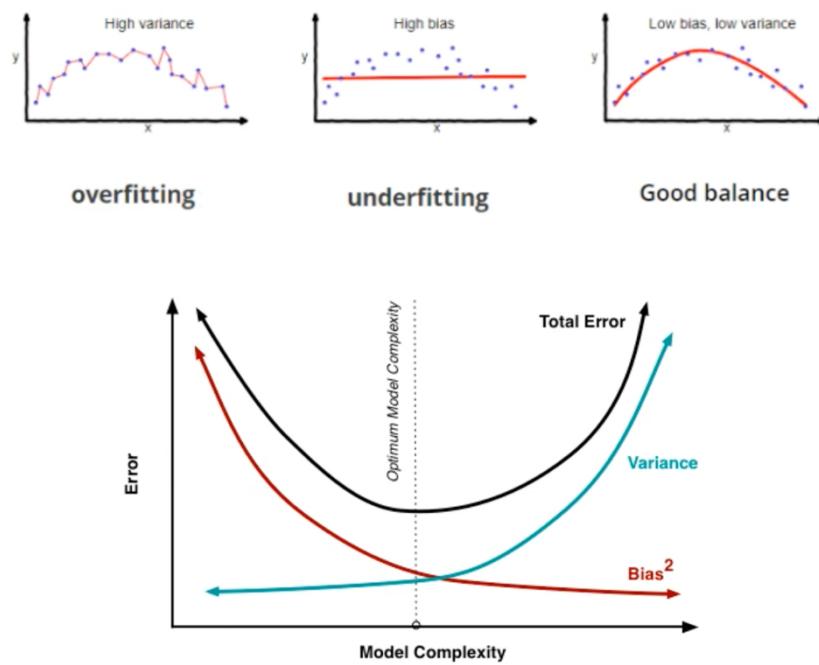
```
Skor:  0.9529411764705882
CM:  [[107    0]
       [ 8  55]]
KNN Accuracy:  0.9529411764705882
```

Yukarıda gösterilen sonuçları değerlendirdiğimizde, öğrenme işleminin %95 oranında başarılı olduğu görülmektedir. Aynı zamanda karışıklık matrisi incelediğinde, iyi huylu verilerden 107 tanesinin doğru tahmin edildiği ve herhangi bir yanlış tahminde bulunulmadığı yorumu yapılabilir. Kötü huylu verilerde ise toplamda 63 verinin 55 tanesinin doğru tahmin edildiği, 8 tanesinin yanlış tahmin edildiği görülmektedir. Doğruluk değerindeki %5 oranında başarısızlığın, bu yanlış tahmin dolayısıyla olduğu açıklar.

3.4. KNN En Uygun Parametreleri Bulma

Modelimizdeki ilk sonuçları test veri setine göre değerlendirdikten sonra eğitim veri setindeki başarının da incelenmesi gerekmektedir. Modelin karmaşıklığı çok yüksekse elindeki veriyi ezberlemiş olduğu anlamına gelmektedir ve bu aşırı öğrenmeye (overfitting) neden olmaktadır. Model ezberlediğinde ise yeni bir veri seti geldiğinde doğru tahminlerde bulunamaz. Model yetersiz olduğunda ise (underfitting) öğrenememiş olduğu anlaşılmaktadır. Dengeli öğrenmede ise model eğitildiği veri setini öğrenmiş ancak ezberlememiştir.

KNN algoritmasında k değeri değişikçe model karmaşıklığı (Model Complexity) değişmektedir. Bunu test edebilmek için ise parametrelerde değişiklik yapılması gerekmektedir.



Şekil 3.2. Model Karmaşıklığını Gösteren Grafik

KNN Algoritmasının modele uygulanması için aşağıdaki fonksiyon tanımlanmıştır:

```
# %% En iyi parametreleri seçme
def KNN_Best_Params(x_train,x_test,y_train,y_test):
    #30 k değeri için en uygun k değerini bulmaya çalışacagiz
    k_range = list(range(1,31))
    weight_options=["uniform","distance"]
    param_grid=dict(n_neighbors=k_range,weights=weight_options)

    knn=KNeighborsClassifier()
    grid=GridSearchCV(knn,param_grid,cv=10,scoring="accuracy")
    grid.fit(x_train,y_train)

    print("{} parametreleri ile en iyi eğitim sonucu: {}".format(grid.best_params_,grid.best_score_))
    print()

    knn=KNeighborsClassifier(**grid.best_params_)
    knn.fit(x_train,y_train)

    y_pred_test=knn.predict(x_test)
    y_pred_train=knn.predict(x_train)

    cm_test=confusion_matrix(y_test,y_pred_test)
    cm_train=confusion_matrix(y_train,y_pred_train)

    acc_test=accuracy_score(y_test, y_pred_test)
    acc_train=accuracy_score(y_train, y_pred_train)

    print("Test Skor: {}, Train skor: {}".format(acc_test, acc_train))
    print()
    print("CM Test: ",cm_test)
    print("CM Train: ",cm_train)

    return grid
```

Modele uygun şekilde fonksiyon çalıştırıldığında aşağıdaki sonuçlar elde edilmiştir:

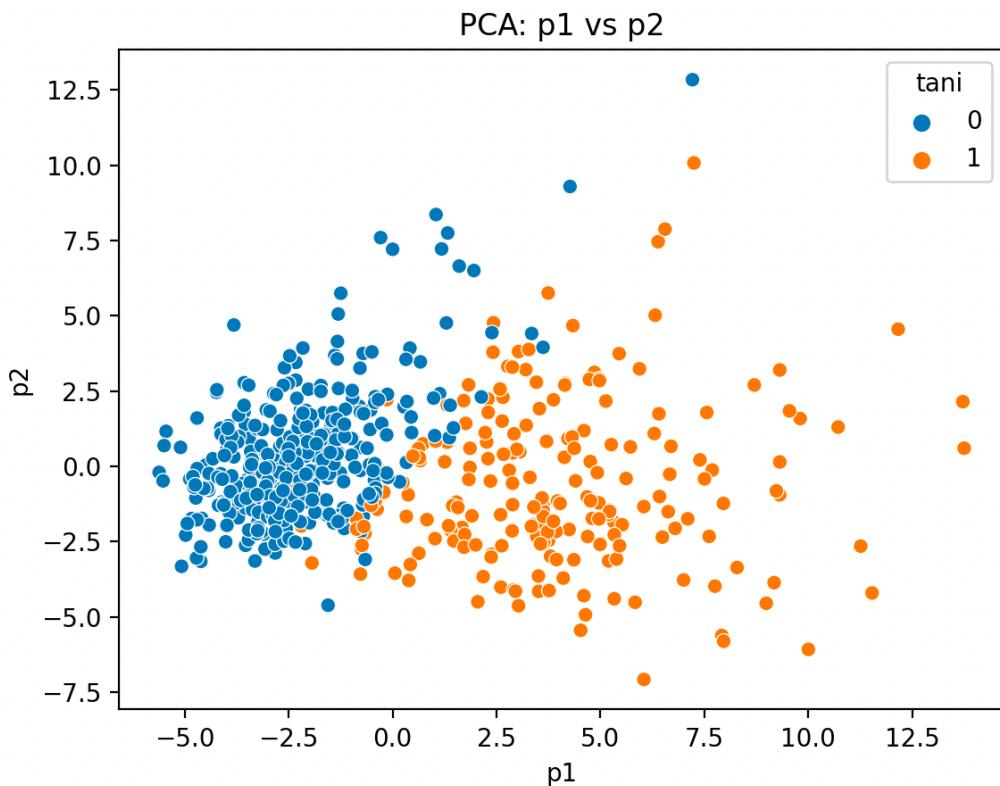
```
{'n_neighbors': 4, 'weights': 'distance'} parametreleri ile en iyi eğitim sonucu: 0.9692948717948718
Test Skor: 0.9470588235294117, Train skor: 1.0
CM Test:  [[104   3]
 [ 6  57]]
CM Train:  [[249   0]
 [ 0 145]]
```

Bu sonuçlara göre eğitim skoru test skorundan fazla çıktıgı için modelimizin aşırı öğrenme yaptığı görülmektedir. Bu sorunun çözümü için model karışıklığını düşürme işlemi uygulanabilir ve test veri seti kullanıldığında %97 doğruluk elde edilebilmektedir.

BÖLÜM 4. TEMEL BİLEŞEN ANALİZİ VE KOMŞULUK BİLEŞENLERİ ANALİZİ İLE SONUÇLARIN DEĞERLENDİRİLMESİ

4.1. Temel Bileşen Analizi (PCA)

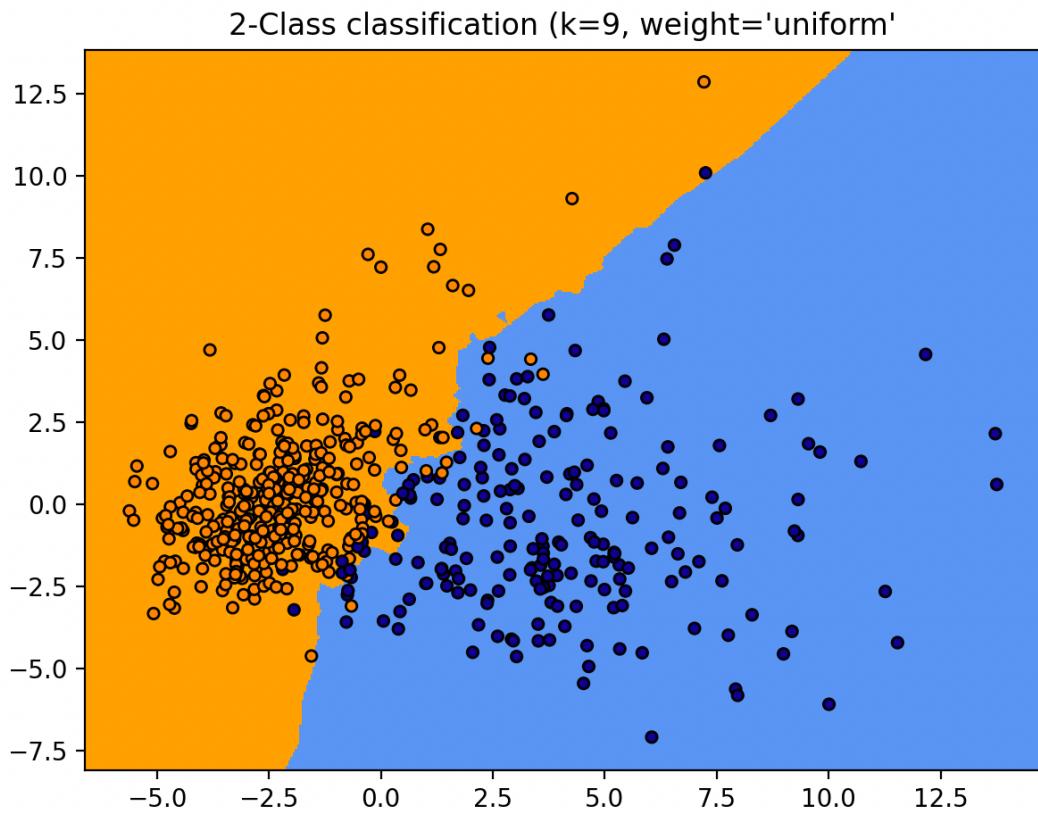
Temel Bileşen Analizi tanımı, sınıflandırma, görüntü sıkıştırma alanlarında kullanılan yararlı bir istatistiksel tekniktir. Temel amacı yüksek boyutlu verilerde en yüksek varyans ile veri setini tutmak ancak bunu yaparken boyut indirmeyi sağlamaktır.



Şekil 4.1. PCA Analizi Grafiği

PCA kullanılarak 30 boyutlu olan veri, 2 boyutlu hale getirilmiş ve görselleştirmesi Şekil 4.1.de gösterilmiştir. Grafikte bazı verilerde analiz işleminde yanlışlık yapılabileceği görülmektedir.

Bu veriler dikkate alınarak veri seti tekrar ayırmış ve %95 doğruluk elde edilmiştir.



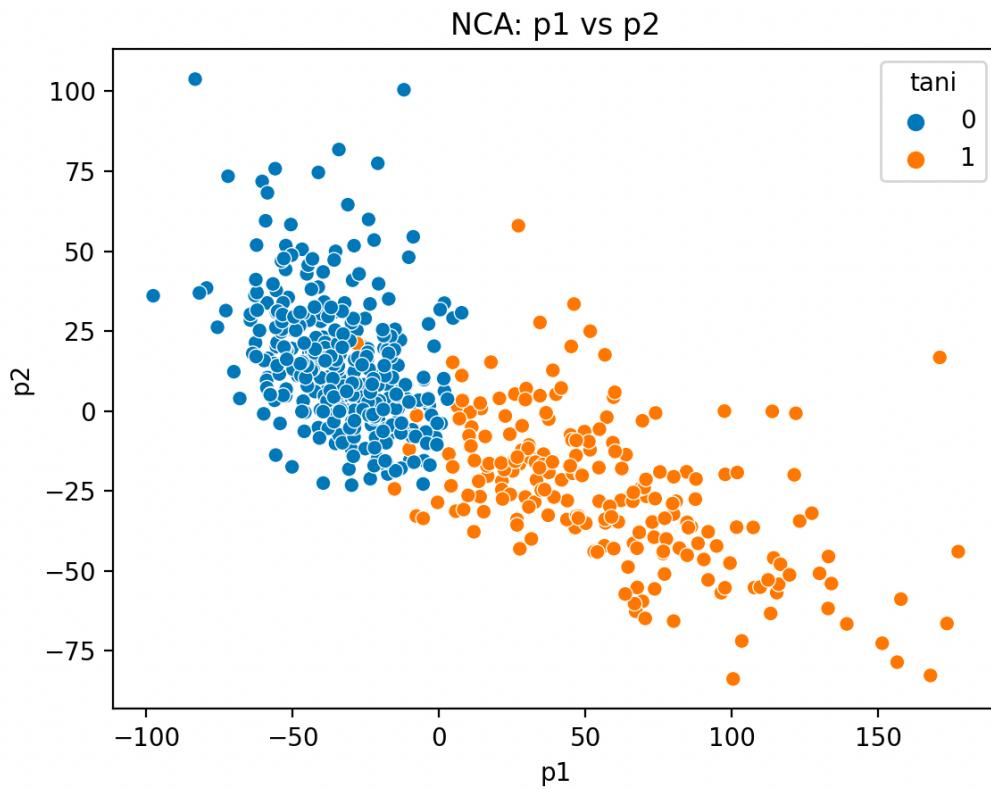
Şekil 4.2. PCA 2-Sınıf için Sınıflandırma Haritası

Şekil 4.2’de her nokta için tahmin yürütülmüş ve analizi kolaylaştırmak adına bir harita oluşturulmuştur. Böylece hangi noktalarda yanlış yapıldığı açıkça görülmektedir.

PCA analizi sonucunda test veri setinin doğruluk yüzdesi %92, eğitim veri setinin doğruluk yüzdesi ise %95 olarak elde edilmiştir.

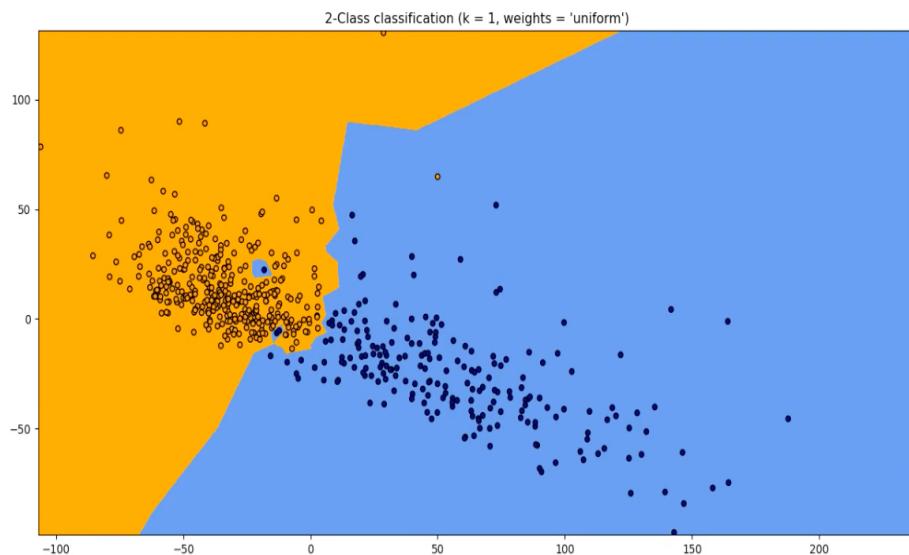
4.2. Komşuluk Bileşenleri Analizi (NCA)

Komşuluk Bileşen Analizi’nde aynı etiketi paylaşan noktaların komşularının farklı etiketlere sahip noktalardan daha dar olduğu bir alanı bulmak için en yakın komşulara benzer bir teknik kullanan bir algoritmadır. KBA’da özniteliklerin ağırlıkları, mesafe ölçümleri kullanılarak yapılmaktadır. KBA, parametrik olmayan, denetimli bir öznitelik seçim yöntemidir. Ayrıca Öklid mesafe ölçemeye alternatif mesafe ölçüm algoritmasıdır. KNN kullanılarak geliştirilmiş ve her öznitelik için pozitif ağırlıklar üretmektedir [6].



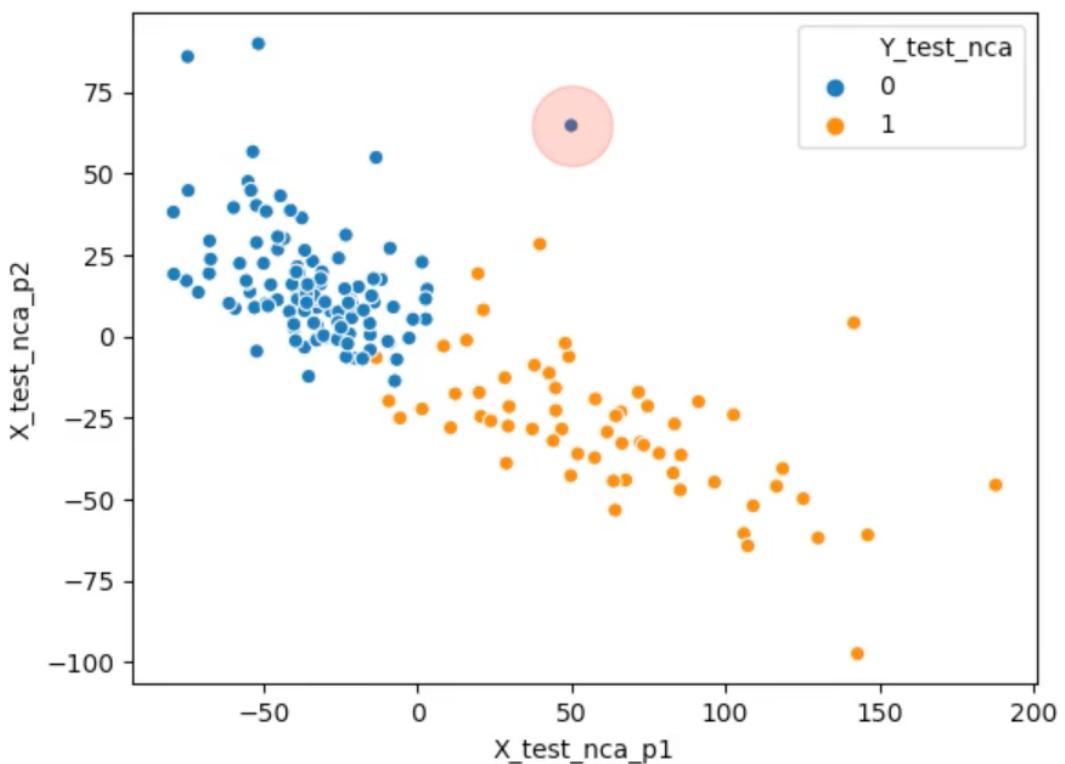
Şekil 4.3. NCA Analizi Grafiği

Şekil 4.3’de NCA analizi sonucunda oluşan grafik elde edilmiştir. Bu grafikte PCA analizine göre daha iyi bir ayırım yapıldığı görülmektedir.



Şekil 4.4. NCA 2-Sınıf için Sınıflandırma Haritası

NCA Algoritmasının uygulanmasının ardından test veri setinin doğruluk oranı %99, eğitim veri setinin doğruluk oranı ise %100 olarak belirlenmiştir. Buna göre Şekil 4.4 de incelendiğinde sarı bölgedeki iki noktanın farklı bir sınır belirlediği görülmektedir. Bu ise eğitim veri setinde belirli noktaların en yakınındaki değerleri seçtiği anlaşılmaktadır. Ancak aynı işlem test veri setinde sağlanamamış dolayısıyla mavi bölgede sarı değerler olduğu görülmektedir. Bu sebeple test veri setinin doğruluk oranı eğitim veri setine düşük çıkmıştır.



Şekil 4.5 Test Veri Setindeki Yanlış Sınıflandırılan Değeri Gösteren Grafik

Test veri setinde yanlış sınıflandırılarak doğruluk değerini düşüren veri Şekil 4.5'te gösterilmiştir.

BÖLÜM 5. SONUÇ VE ÖNERİLER

Bu çalışma, Meme Kanseri veri seti ele alınarak Makine Öğrenmesi ile kanser hücreleri ile ilgili tanı koyabilme amacı ile hazırlanmıştır. Çalışmada Python programlama dili kullanılarak, bilgisayarın veri setindeki tanı özelliğini doğru değerlendirebilmesi hedeflenmiştir.

Çalışmanın ilk bölümünde veri seti incelenmiş ve problem tanımlanmıştır. Ardından Keşifsel Veri Analizi (EDA) ile detaylı bir şekilde analiz gerçekleştirilmiştir. Aykırı değerler belirlenmiş ve bu değerler veri setinden çıkarılmıştır. Son olarak veri seti eğitim ve test veri seti olarak iki bölüme ayrılmış ve bu noktadan sonra veri seti eğitilebilir hale gelmiştir.

Bu işlemlerin ardından KNN algoritması eğitimi gerçekleştirilmiş ve bu işlem sonucunda % 95 oranında doğruluk elde edilmiştir. Bu değer beklenenden yüksektir çünkü aşırı öğrenme (overfit) ya da yetersiz öğrenme (underfit) gerçekleşmemiştir. Ardından en iyi parametrelerin bulunması işlemi yapılmış, burada aşırı öğrenme gerçekleşmiş ve bu sebeple doğruluk oranı %94'e inmiştir.

PCA analizi ile veri seti iki boyuta düşürülmüş, ancak doğruluk değeri % 92'ye kadar inmiştir.

NCA analizi sonrasında yine veri seti iki boyuta düşürülmüş ancak aşırı öğrenme ya da yetersiz öğrenme olmadan iyi bir model elde edilmiştir. Bu işlem sonucunda doğruluk değeri %99 olarak belirlenmiş, istenen şekilde veri setini makinenin öğrenmesi sağlanmıştır.

Çalışma ile ilgili kodlara ulaşabilmek adına aşağıdaki link verilmiştir.

Çalışma Github Linki: <https://github.com/AysunCagYilmazkulas/Breast-Cancer-Data-Set-KNN>

KAYNAKLAR

- [1] YILDIZ, O., BİLGE, H., Meme Kanseri Sınıflandırması İçin Gen Seçimi, Gazi Üniversitesi, Bilgisayar Mühendisliği Programı, 2012.
- [2] <http://www.memekanseri.org.tr/meme-sagligi/meme-kanseri-teshis-yontemleri>
- [3] <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>
- [4] https://en.wikipedia.org/wiki/Exploratory_data_analysis
- [5] <https://en.wikipedia.org/wiki/Outlier>
- [6] ÖZYURT, F. Uzaktan Algılama Görüntülerinin Evrişimsel Sinir Ağları ve Komşuluk Bileşen Analizi Tabanlı Özniteliklerinin Sınıflandırılması, Fırat Üniversitesi, Enformatik Bölümü, 2019.

ÖZGEÇMİŞ

Aysun Çağ Yılmazkulaş, 24.05.1988 tarihinde Artvin'de doğdu. İlk, orta ve lise eğitimini Karabük'te tamamladı. 2011 yılında Ondokuz Mayıs Üniversitesi, Matematik Bölümü'nden mezun oldu. İstanbul'da çeşitli eğitim kurumlarında Matematik Öğretmeni olarak çalıştı. 2019 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü'nü kazandı. 2020 yılında Argeset Yazılım ve Danışmanlık Şirketinde yazılım ve donanım stajını yapmıştır.

BSM 498 BİTİRME ÇALIŞMASI DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI

KONU : MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE MEME KANSERİ VERİLERİİN DEĞERLENDİRİLMESİ

ÖĞRENCİLER (Öğrenci No/AD/SOYAD):

G191210373/Aysun/Yılmazkulaş

Değerlendirme Konusu	İstenenler	Not Aralığı	Not
Yazılı Çalışma			
Çalışma klavuza uygun olarak hazırlanmış mı?	x	0-5	
Teknik Yönden			
Problemin tanımı yapılmış mı?	x	0-5	
Geliştirilecek yazılımın/donanımın mimarisini içeren blok şeması (yazılımlar için veri akış şeması (dfd) da olabilir) çizilerek açıklanmış mı?			
Blok şemadaki birimler arasındaki bilgi akışına ait model/gösterim var mı?			
Yazılımın gereksinim listesi oluşturulmuş mu?			
Kullanılan/kullanılması düşünülen araçlar/teknolojiler anlatılmış mı?			
Donanımların programlanması/konfigürasyonu için yazılım gereksinimleri belirtilmiş mi?			
UML ile modelleme yapılmış mı?			
Veritabanları kullanılmış ise kavramsal model çıkarılmış mı? (Varlık ilişki modeli, noSQL kavramsal modelleri v.b.)			
Projeye yönelik iş-zaman çizelgesi çıkarılarak maliyet analizi yapılmış mı?			
Donanım bileşenlerinin maliyet analizi (prototip-adetli seri üretim vb.) çıkarılmış mı?			
Donanım için gerekli enerji analizi (minimum-uyku-aktif-maksimum) yapılmış mı?			
Grup çalışmalarında grup üyelerinin görev tanımları verilmiş mi (iş-zaman çizelgesinde belirtilebilir)?			
Sürüm denetim sistemi (Version Control System; Git, Subversion v.s.) kullanılmış mı?			
Sistemin genel testi için uygulanan metodlar ve iyileştirme süreçlerinin dökümü verilmiş mi?			
Yazılımın sizme testi yapılmış mı?			
Performans testi yapılmış mı?			
Tasarımın uygulamasında ortaya çıkan uyumsuzluklar ve aksaklıklar belirtilerek çözüm yöntemleri tartışılmış mı?			
Yapılan işlerin zorluk derecesi?	x	0-25	
Sözlü Sınav			
Yapılan sunum başarılı mı?	x	0-5	
Soruları yanıtlama yetkinliği?	x	0-20	
Devam Durumu			
Öğrenci dönem içerisindeki raporlarını düzenli olarak hazırladı mı?	x	0-5	
Diğer Maddeler			
Toplam			

DANIŞMAN (JÜRI ADINA): DOÇ. DR. NILÜFER YURTAY

DANIŞMAN İMZASI: