T.C. AFYON KOCATEPE ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

SD415 VERİ MADENCİLİĞİ PROJESİ

KNN ALGORİTMASI İLE KALP KRİZİ RİSK ANALİZİ

Hazırlayanlar:

Mehmet Göktuğ GÖKÇE (212923025)
Sinan MALAK (212923008)
Onur BARBAROS (212923036)

İçindekiler

1. Giriş

Kalp krizi, dünya genelinde en önemli sağlık sorunlarından biridir. Teknolojinin gelişmesiyle birlikte, bireylerin sağlık durumlarını analiz etmek ve olası riskleri belirlemek için veri madenciliği teknikleri kullanılmaktadır. Bu çalışma, bireylerin kalp krizi riskine yatkınlığını tahmin etmek amacıyla geliştirilmiştir. Proje kapsamında veri madenciliği algoritmaları, veri görselleştirme teknikleri ve öznitelik mühendisliği gibi yöntemler kullanılmıştır.

2. Proje Hakkında

Bu proje, bireylerin kalp krizi risklerini sınıflandırmayı hedefleyen bir veri madenciliği çalışmasıdır. Kullanılan veri seti, bireylerin yaş, cinsiyet, kan basıncı, kolesterol seviyesi gibi sağlık durumlarını temsil eden 13 farklı öznitelikten oluşmaktadır. Veri setindeki bireyler "riskli" ve "risksiz" olarak iki sınıfa ayrılmıştır. Çalışma, KNN (K-Nearest Neighbors) algoritmasını kullanarak sınıflandırma doğruluğunu artırmayı amaçlamaktadır.

3. Veri Seti

- Veri Kaynağı: Kaggle (Heart Attacks) Veri Seti
- Kayıt Sayısı: 303
- Sütun Sayısı: 14 (13 öznitelik ve 1 hedef değişken)
- Hedef Değişken: output
    - 1: Riskli birey
    - 0: Risksiz birey

Veri seti, bireylerin yaş, cinsiyet, kolesterol, kan şekeri gibi sağlık parametrelerini içermektedir. Bu parametrelerin doğru şekilde analiz edilmesi, proje başarısı için kritik öneme sahiptir.

	age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
5	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
6	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
7	44	1	1	120	263	0	1	173	0	0.0	2	0	3	1
8	52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
9	57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
10	54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
11	48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
12	49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
13	64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
14	58	0	3	150	283	1	0	162	0	1.0	2	0	2	1
15	50	0	2	120	219	0	1	158	0	1.6	1	0	2	1
16	58	0	2	120	340	0	1	172	0	0.0	2	0	2	1
17	66	0	3	150	226	0	1	114	0	2.6	0	0	2	1
18	43	1	0	150	247	0	1	171	0	1.5	2	0	2	1
19	69	0	3	140	239	0	1	151	0	1.8	2	2	2	1
20	59	1	0	135	234	0	1	161	0	0.5	1	0	3	1
21	44	1	2	130	233	0	1	179	1	0.4	2	0	2	1
22	42	1	0	140	226	0	1	178	0	0.0	2	0	2	1
23	61	1	2	150	243	1	1	137	1	1.0	1	0	2	1
24	40	1	3	140	199	0	1	178	1	1.4	2	0	3	1
25	71	0	1	160	302	0	1	162	0	0.4	2	2	2	1
26	59	1	2	150	212	1	1	157	0	1.6	2	0	2	1
27	51	1	2	110	175	0	1	123	0	0.6	2	0	2	1
28	65	0	2	140	417	1	0	157	0	0.8	2	1	2	1
29	53	1	2	130	197	1	0	152	0	1.2	0	0	2	1
30	41	0	1	105	198	0	1	168	0	0.0	2	1	2	1
31	65	1	0	120	177	0	1	140	0	0.4	2	0	3	1

4. Projede Kullanılan Teknolojiler ve Araçlar

4.1. Gerekli Kütüphaneler

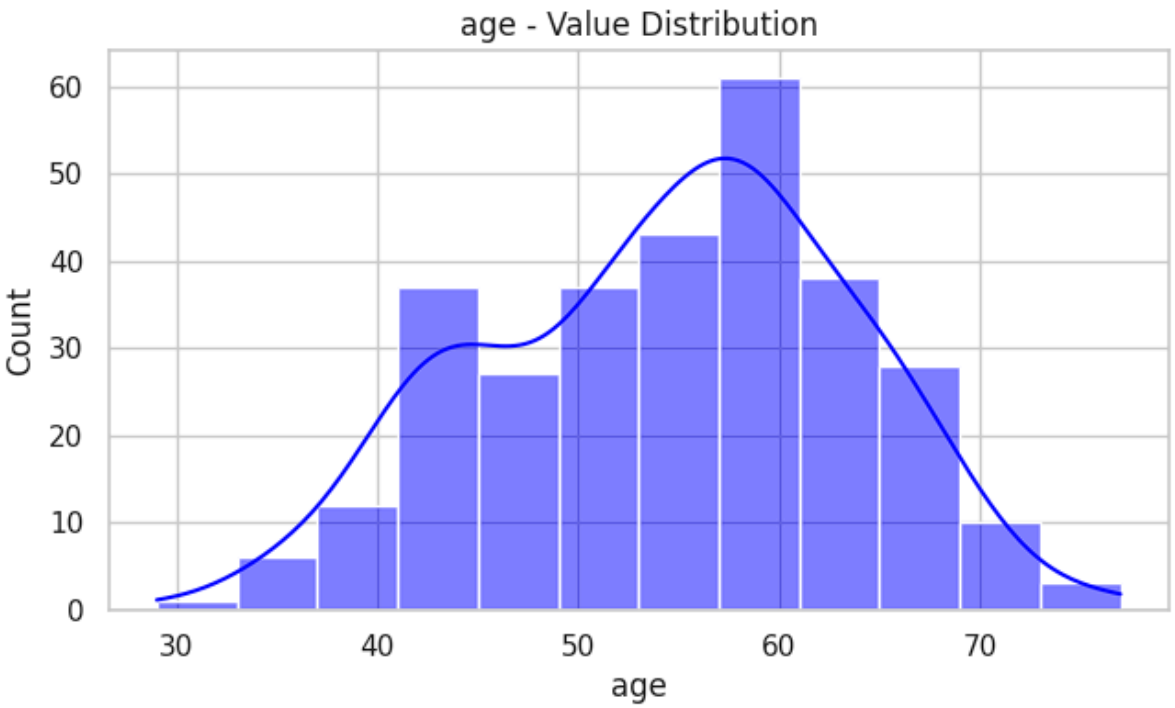
```
# Sayısal işlemler ve dizilerle çalışmak için
import numpy as np
# Model performans metrikleri için
from sklearn.metrics import classification_report, accuracy_score
# Veri setini eğitim ve test setlerine ayırmak için
from sklearn.model_selection import train_test_split
# Verileri ölçeklendirmek için
from sklearn.preprocessing import StandardScaler
# Grafik oluşturmak için
import matplotlib.pyplot as plt
# Boyut indirgeme işlemleri için
from sklearn.decomposition import PCA
# Renk haritası oluşturmak için
from matplotlib.colors import ListedColormap
# Veri görselleştirme ve dağılım analizleri için
import seaborn as sns
# Veriyi yükleme ve hazırlama
import pandas as pd
```

- Sayısal işlemler ve dizilerle çalışmak için: **numpy**
- Model performans metrikleri için: **sklearn.metrics**
- Veri setini eğitim ve test setlerine ayırmak için: **sklearn.moel_selection**
- Verileri ölçeklendirmek için: **sklearn.preprocessing**
- Grafik oluşturmak için: **matplotlib.pyplot**
- Boyut indirgeme işlemleri için: **sklearn.decomposition**
- Renk haritası oluşturmak için: **matplotlib.colors**
- Veri görselleştirme ve dağılım analizleri için: **seaborn**
- Veriyi yükleme ve hazırlama için: **pandas**

Yukarıdaki kütüphaneler proje içinde birçok fonksiyon için kullanılmıştır.

4.2. Veri Analizi ve Görselleştirme

Veri görselleştirme adımları, veri setindeki öznitelikler arasındaki ilişkileri anlamak ve desenleri ortaya çıkarmak için uygulanmıştır. Özellikle grafiklerinden faydalanılmıştır.



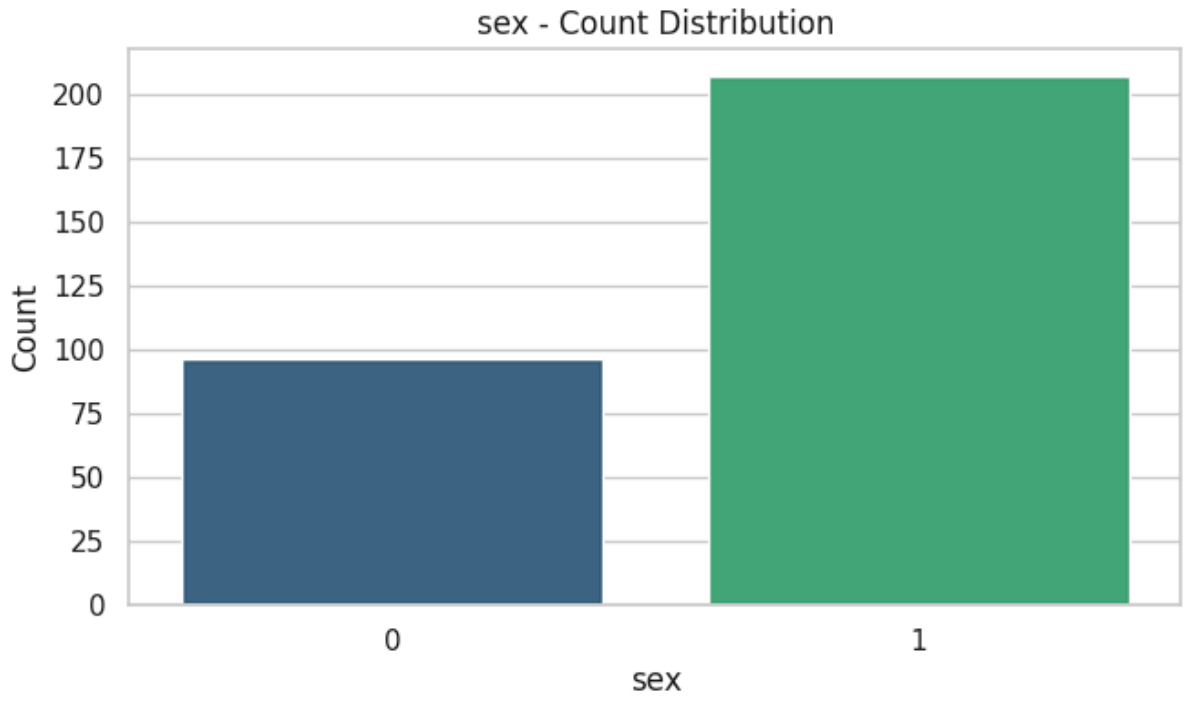
Bu grafik, bir yaş değişkeninin (age) değer dağılımını görselleştiren bir histogram ve yoğunluk tahmini (KDE) eğrisini içermektedir.

- **X ekseni (age):** Yaş değişkeninin farklı değerlerini göstermektedir ve aralık 30 ile 70 arasında sınırlandırılmıştır.
- **Y ekseni (Count):** Her bir yaş aralığındaki (bin) gözlem sayısını ifade etmektedir.
- **Histogram:** Mavi çubuklarla temsil edilen bu bölüm, yaş verilerinin farklı gruplar (binler) arasındaki sıklığını görselleştirmektedir.
- **Yoğunluk eğrisi (KDE):** Histogram üzerinde çizilmiş olan mavi eğri, veri setindeki yaş değişkeninin dağılımını daha akıcı ve sürekli bir biçimde göstermektedir.

Analiz:

- Grafikte en yüksek sıklık, yaklaşık 60 yaş civarında gözlemlenmiştir, bu da veri setinde bu yaş grubunda daha fazla bireyin bulunduğunu ifade eder.
- Dağılım, genel olarak simetrik bir yapı sergilemekte olup normal dağılıma yakın bir şekle sahiptir.
- Yaş gruplarının yoğunluğu, 40 ila 60 yaş arasında yüksek bir konsantrasyon göstermekte, ancak 70 yaşa doğru azalma eğilimi sergilemektedir.

Bu grafik, yaş değişkeninin genel dağılımını anlamak ve veri setindeki ana eğilimleri belirlemek amacıyla etkili bir görselleştirme sunmaktadır.



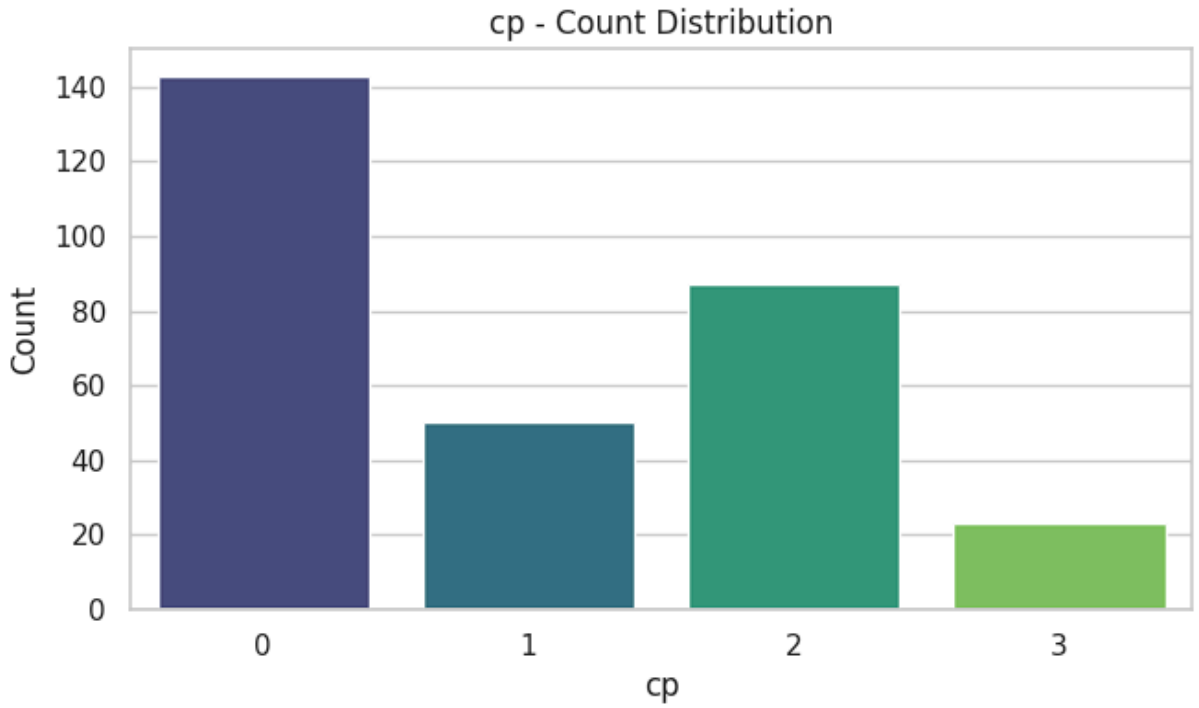
Bu grafik, "sex" değişkeninin kategorik dağılımını temsil eden bir sütun grafiğidir.

- **X ekseni (sex):** "Sex" değişkeninin kategorik değerlerini göstermektedir. Grafikte iki kategori bulunmaktadır:
  - 0: Kadın
  - 1: Erkek
- **Y ekseni (Count):** Her bir kategoriye karşılık gelen gözlem sayısını ifade etmektedir.
- **Sütunlar:** Her iki kategorideki veri frekanslarını görselleştirmektedir.

Analiz:

- **Kategori 1(Erkek)**, yaklaşık 200 gözlem ile daha yüksek bir frekansa sahiptir.
- **Kategori 0(Kadın)**, yaklaşık 100 gözlem ile daha düşük bir frekans göstermektedir.
- Bu dağılım, ikinci kategorinin (1) birinci kategoriye (0) kıyasla iki kat daha fazla temsil edildiğini göstermektedir.

Bu grafik, "sex" değişkenindeki kategorilerin veri setindeki dağılımını karşılaştırmak ve bu değişkenin sınıf dengesini analiz etmek amacıyla etkili bir araçtır.



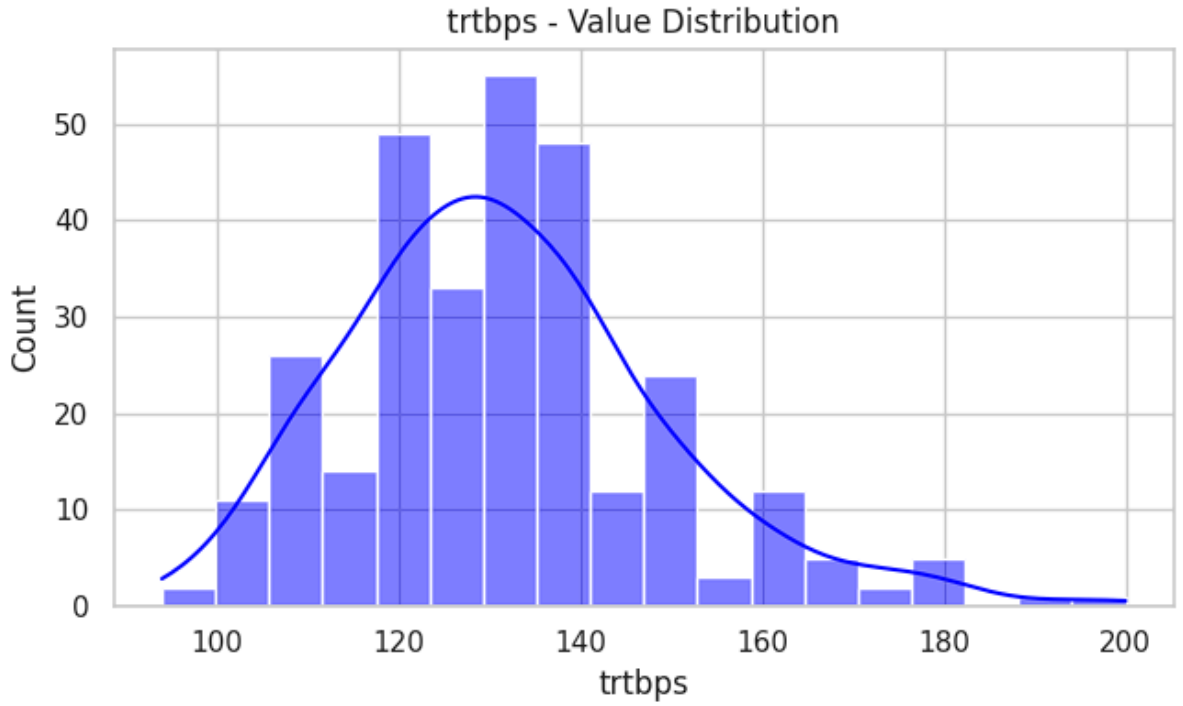
Bu grafikte, "cp" değişkeninin farklı kategorilerine ait sıklık dağılımı gösterilmektedir.

Açıklama: Grafik, "cp" ("chest pain type" yani göğüs ağrısı tipi) değişkeninin dört kategorisini (0, 1, 2, 3) içerir ve her bir kategoriye ait gözlem sayısını temsil eder.

- Kategori 0 en fazla gözleme sahiptir ve yaklaşık 140 civarındadır.
- Kategori 1 ve 2, sırasıyla yaklaşık 60 ve 90 arasında değişen gözlem sayılarıyla daha az yaygındır.
- Kategori 3, yaklaşık 20 gözlemle en az temsil edilen gruptur.

Bu dağılım, veri setinde "cp" değişkeninin dengesiz bir dağılıma sahip olduğunu ve kategori 0'ın baskın olduğunu göstermektedir. Bu tür dağılımlar, modelleme süreçlerinde dikkate alınmalı ve gerekirse dengesizlik giderici teknikler kullanılmalıdır.

0: Typical Angina, 1: Atypical Angina, 2: Non-anginal Pain, 3: Asymptomatic

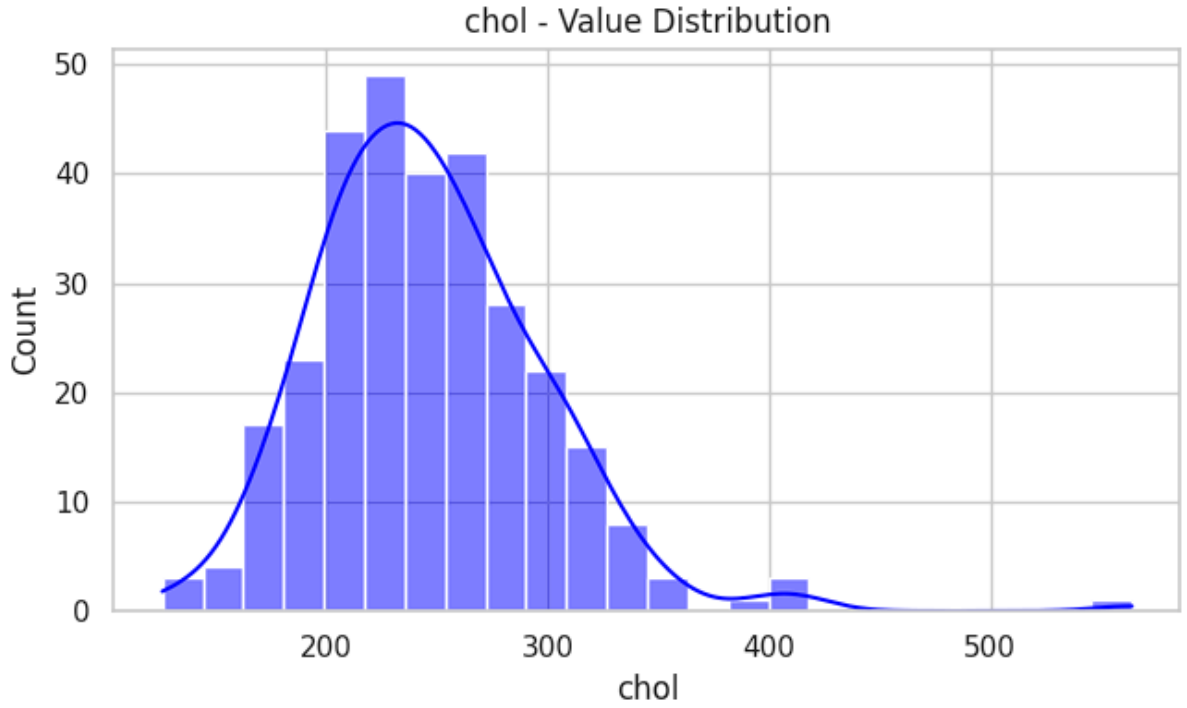


Bu grafikte, "trtbps" değişkeninin dağılımı histogram ve yoğunluk eğrisi kullanılarak görselleştirilmiştir.

Açıklama: "trtbps" değişkeni "dinlenme halindeki kan basıncı" değerlerini temsil etmektedir. Histogram, bu değişkenin gözlem sıklığını çeşitli aralıklara göre göstermektedir. Mavi çizgi ise yoğunluk eğrisidir ve dağılımın genel eğilimini belirtir.

- Grafikte, "trtbps" değişkeni yaklaşık 100 ile 200 arasında değişmektedir.
- En yüksek sıklık, yaklaşık 120 ile 140 aralığında görülmektedir. Bu, çoğu bireyin dinlenme halindeki kan basıncı değerlerinin bu aralıkta yoğunlaştığını göstermektedir.
- Değişkenin dağılımı hafif sağa çarpık bir yapı sergilemektedir; daha yüksek "trtbps" değerleri nadir görülmektedir.

Bu tür bir dağılım, veri setinin genel sağlık durumu veya normal kan basıncı aralıklarını yansıtabilir ve analiz sürecinde dikkate alınmalıdır.

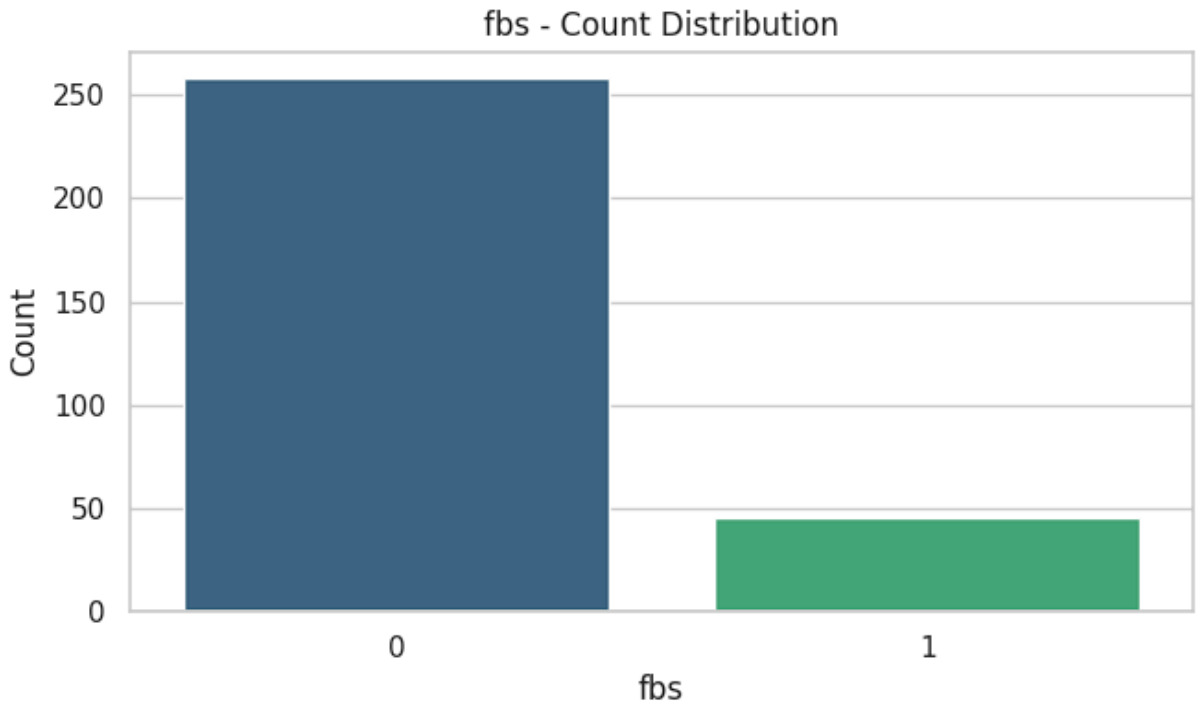


Bu grafikte, "chol" değişkeninin dağılımı histogram ve yoğunluk eğrisi ile gösterilmektedir.

Açıklama: "chol" değişkeni bireylerin serum kolesterol seviyelerini ifade etmektedir. Histogram, bu değişkenin değerlerine göre sıklık dağılımını sunarken, mavi yoğunluk eğrisi genel dağılım eğilimini göstermektedir.

- "chol" değerleri yaklaşık 100 ile 500 arasında değişim göstermektedir.
- Dağılımın tepe noktası, yaklaşık 200 ile 250 arasındadır; bu da bu aralıktaki kolesterol seviyelerinin en sık görülen değerler olduğunu göstermektedir.
- Dağılım, sağa çarpık bir yapı sergilemektedir; bu da daha yüksek kolesterol seviyelerinin nadiren görüldüğünü işaret etmektedir.

Bu dağılım, popülasyonda kolesterol seviyelerinin normal dağılımını veya yüksek kolesterolün yaygınlığını anlamak için değerlendirilebilir. Sağlıklı ve anormal seviyelerin sınırları, dağılımın yorumlanmasında belirleyici olacaktır.

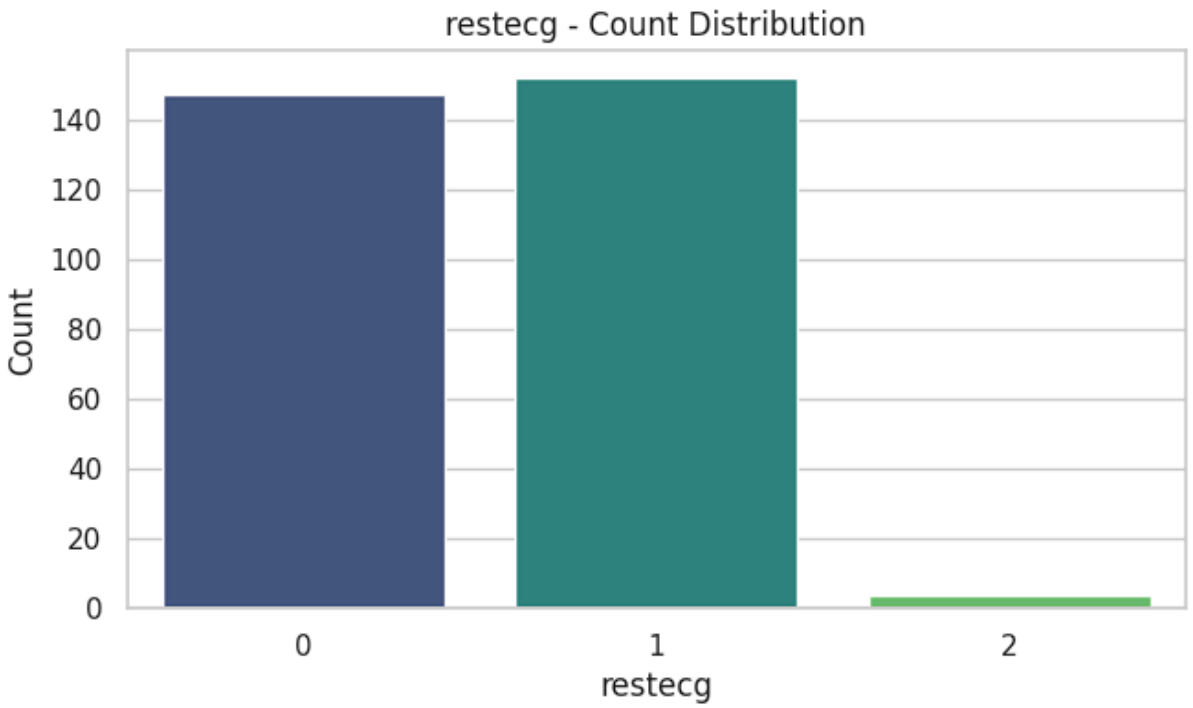


Bu grafikte, "**fbs**" (açlık kan şekeri) değişkeninin kategorik dağılımı sunulmaktadır. Grafikteki "0" ve "1" değerleri, ilgili değişkenin iki kategorik durumu arasında gözlenen örneklem büyüklüklerini temsil etmektedir.

Sonuçlar, "**fbs** = 0" durumunun açık bir şekilde baskın olduğunu göstermektedir; bu durum, toplam örneklemin büyük bir kısmının açlık kan şekerinin belirlenen eşik değerinin altında olduğunu göstermektedir. Buna karşılık, "**fbs** = 1" durumu oldukça sınırlı sayıda gözlem ile temsil edilmiştir.

Bu dağılım, dengesiz bir veri yapısına işaret etmekte olup, özellikle sınıflandırma modellerinin performansı üzerinde etkili olabilecek bir durumdur. Model geliştirme sürecinde, bu dengesizliğin ele alınması adına veri dengeleme stratejilerinin (örneğin, örnek çoğaltma veya ağırlıklandırma yöntemleri) dikkate alınması gerekebilir.

Grafikte gözlenen bu asimetrik yapı, açlık kan şekeri durumuna bağlı sağlık göstergelerinin analizinde belirli bir alt grubun (örneğin "**fbs** = 0") diğerine kıyasla daha fazla temsil edildiğini ifade etmektedir.

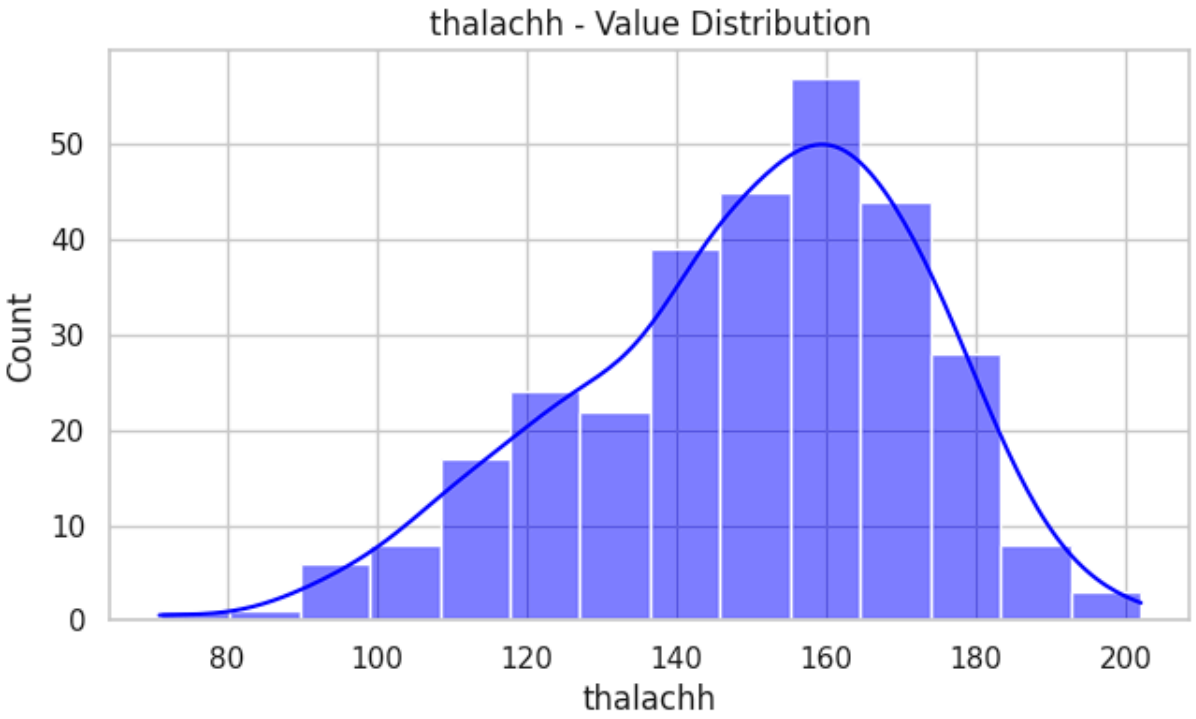


Grafikte, "**restecg**" (istirahat elektrokardiyografi sonuçları) değişkeninin kategorik dağılımı sunulmaktadır. Bu değişken, üç farklı kategoriyi temsil etmekte olup, sırasıyla "0", "1" ve "2" kodlarıyla gösterilmektedir.

Sonuçlar, "**restecg** = 0" ve "**restecg** = 1" kategorilerinin gözlem frekanslarının birbirine oldukça yakın olduğunu, buna karşılık "**restecg** = 2" kategorisinin oldukça düşük bir temsil oranına sahip olduğunu göstermektedir. Bu durum, veri setinin büyük bir kısmının iki ana kategoriye yoğunlaştığını ve üçüncü kategorinin marjinal bir örnekleme sahip olduğunu ifade etmektedir.

Bu asimetrik dağılım, veriye dayalı çıkarımlarda potansiyel sınırlamalar yaratabilir. Özellikle, "**restecg** = 2" kategorisinin sınırlı örneklem büyüklüğü, bu sınıf ile ilgili anlamlı istatistiksel analizlerin yapılmasını veya makine öğrenimi modellerinin bu sınıfı doğru bir şekilde tahmin etmesini zorlaştırabilir. Bu nedenle, dengesizliği ele almak amacıyla stratejik yaklaşımlar (örneğin, sınıf ağırlıklandırması veya veri çoğaltma teknikleri) dikkate alınmalıdır.

Sonuç olarak, "**restecg**" değişkeni, veri setinde ağırlıklı olarak iki ana grup arasında eşit bir dağılıma sahiptir ve üçüncü grup marjinal bir şekilde temsil edilmektedir. Bu durum, değişkenin modelleme süreçlerine etkisini ve ilgili analizlerin doğruluğunu değerlendirmek için önemli bir bulgudur.

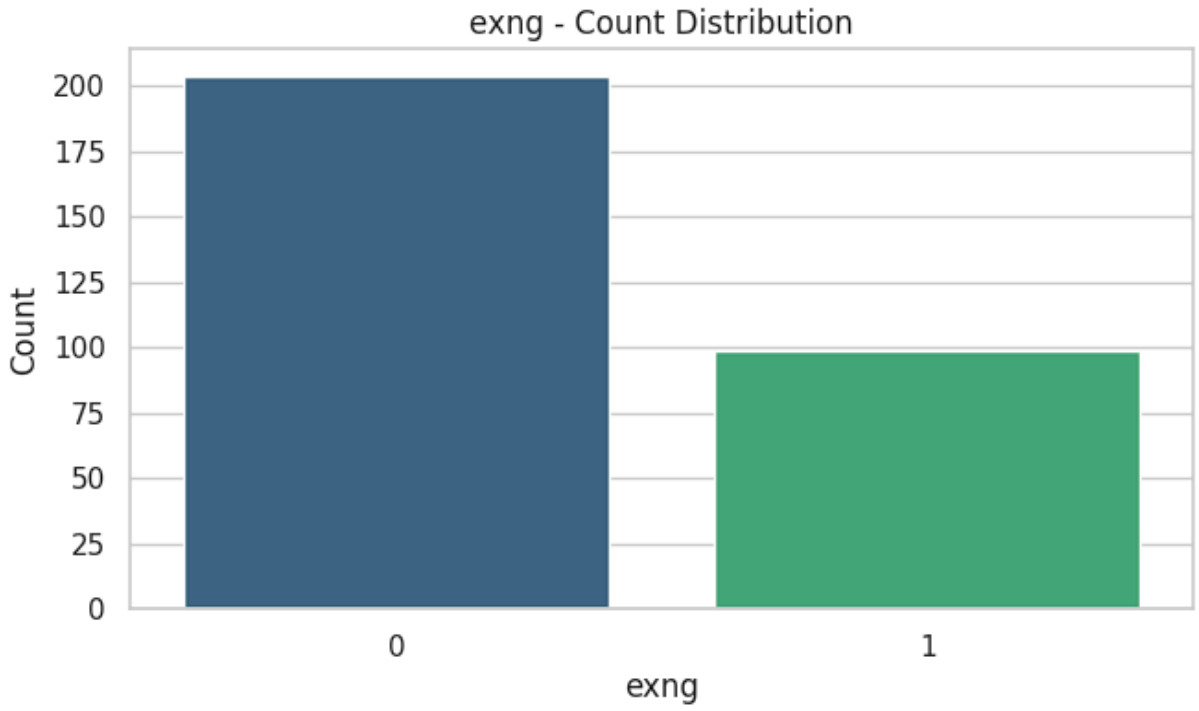


Grafikte, "**thalachh**" (maksimum kalp atış hızı) değişkeninin sürekli veri dağılımı histogram ve bir yoğunluk eğrisiyle birlikte sunulmaktadır. Dağılım, veri kümesinin bu değişken için genellikle normal dağılıma yakın bir yapıya sahip olduğunu göstermektedir.

Dağılımın merkezi, yaklaşık 150-160 aralığında yoğunlaşmakta olup, bu değerler veri kümesindeki bireyler için en sık gözlemlenen maksimum kalp atış hızı değerlerini temsil etmektedir. Bunun yanı sıra, sol ve sağ uçlarda sırasıyla 80 ile 200 arasında daha seyrek veri noktaları bulunmaktadır. Bu durum, ilgili değişkenin uç değerler açısından sınırlı bir varyansa sahip olduğunu işaret etmektedir.

Grafiğin sol tarafında (120'nin altında) ve sağ tarafında (180'in üzerinde) azalan frekanslar, değişkenin genel olarak bir çan eğrisine benzer bir biçimde yayıldığını ifade etmektedir. Ancak, küçük de olsa bazı uç değerlerin (örneğin, 80 veya 200'e yakın gözlemler) veri kümesinde varlığı, bu değerlerin analiz sırasında dikkatle ele alınması gerektiğini göstermektedir.

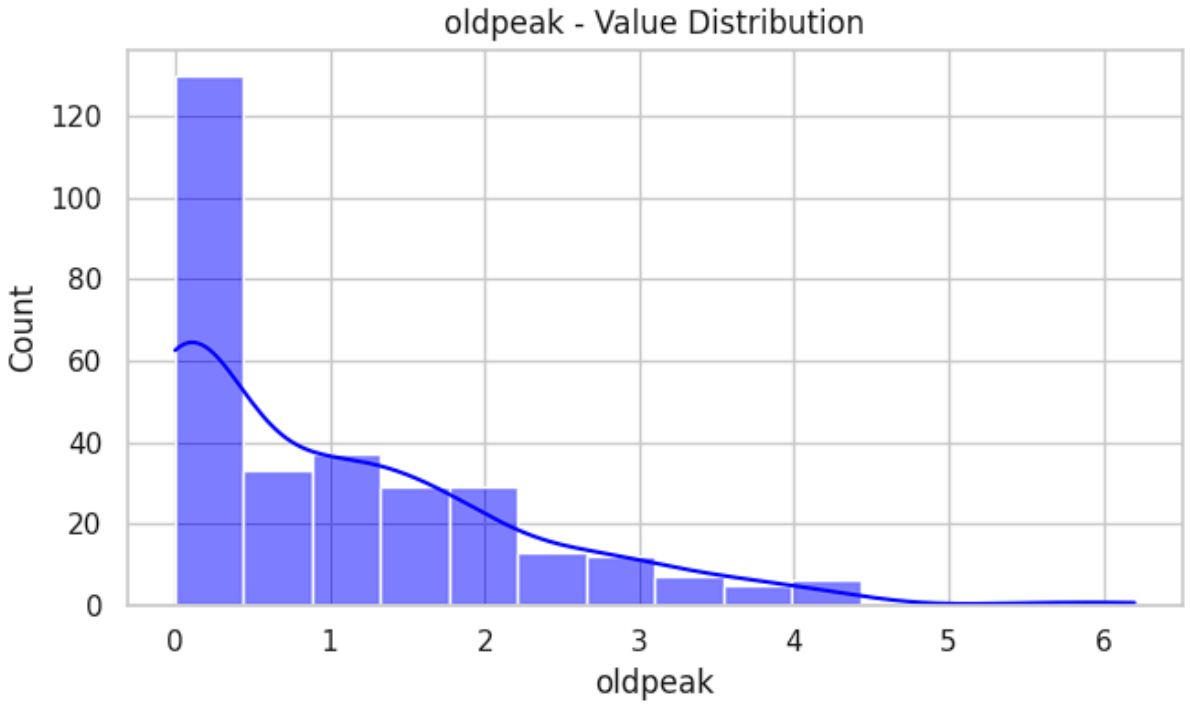
Bu bulgular, "**thalachh**" değişkeninin normal dağılıma yakın bir yapıya sahip olduğunu ve merkezden uzak değerlerin analiz sırasında istisnai durumlar olarak değerlendirilebileceğini ortaya koymaktadır. İlgili değişkenin dağılımının böyle bir yapıya sahip olması, istatistiksel modelleme süreçlerinde varsayım testlerinin (örneğin, normalite testi) sonuçlarını etkileyebilir.



Bu grafik, veri setindeki bireylerin egzersiz kaynaklı anjina (exng) durumlarının dağılımını göstermektedir. Değerler aşağıdaki gibi iki kategoriye ayrılmıştır:

- **0:** Egzersiz kaynaklı anjina yok.
- **1:** Egzersiz kaynaklı anjina var.

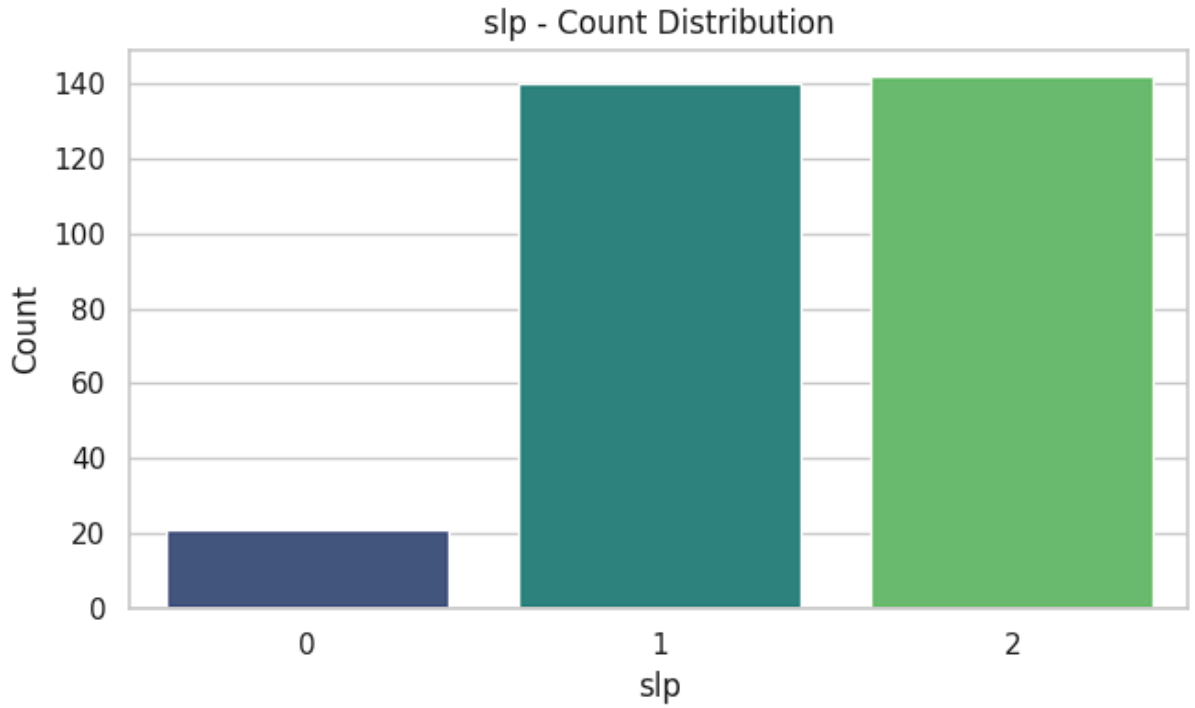
Grafiğe göre, veri setindeki bireylerin büyük bir kısmı (%~67) egzersiz kaynaklı anjina yaşamazken, yaklaşık %33'lük bir kısmı bu durumdan muzdariptir. Bu değişken, kalp krizi riski açısından önemli bir öznitelik olabilir, çünkü egzersiz kaynaklı anjina, kalp sağlığına yönelik belirgin bir işaret taşıyabilir.



Bu grafik, oldpeak değişkeninin değer dağılımını bir histogram ve bir yoğunluk eğrisi yardımıyla göstermektedir. oldpeak, egzersiz sonrası ST segmenti depresyonunu ifade eden sürekli bir değişkendir. Grafik incelendiğinde:

- Dağılımın sağa çarpık olduğu gözlemlenmektedir, bu da düşük oldpeak değerlerinin veri setinde daha sık görüldüğünü, yüksek oldpeak değerlerinin ise daha seyrek olduğunu göstermektedir.
- Histogram çubukları ve yoğunluk eğrisi, verilerin 0 ile 1 arasında yoğunlaştığını ve 4-6 aralığında daha az sayıda gözlem bulunduğunu ifade etmektedir.

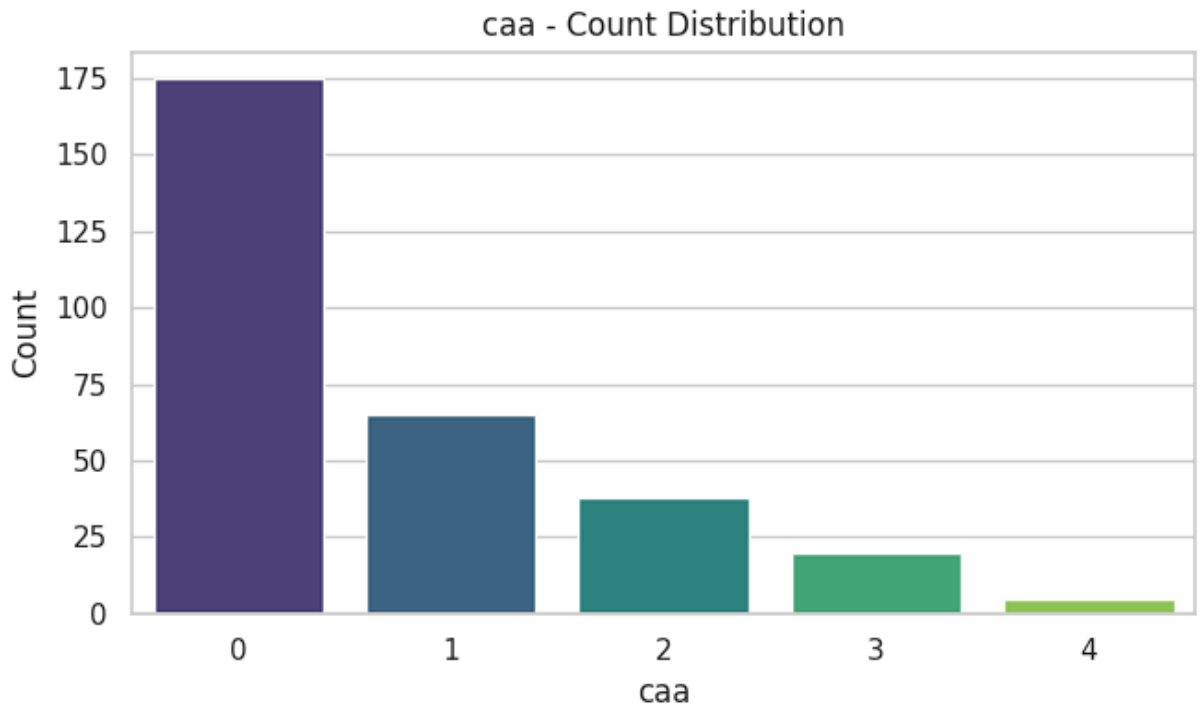
Bu tür bir dağılım, oldpeak değerinin tıbbi bir bağlamda genellikle düşük seviyelerde seyrettiğini, yüksek seviyelerin ise nadir olduğunu gösterebilir.



Bu grafik, bir kategorik değişken olan "slp" değişkeninin dağılımını göstermektedir ve farklı sınıflara ait gözlem sayılarını karşılaştırmaktadır. Yatay eksen "slp" değişkeninin kategorilerini temsil ederken (0, 1 ve 2), dikey eksen her bir kategoriye ait gözlem sayılarını göstermektedir.

Grafikte yer alan üç farklı kategori, sırasıyla 0, 1 ve 2 değerlerini almıştır. Her bir sütunun yüksekliği, ilgili kategoriye ait veri sayısını ifade etmektedir. Kategori 1 ve kategori 2, grafikte neredeyse eşit yüksekliğe sahip olup, her birinde yaklaşık 140 gözlem bulunmaktadır. Bu da bu iki kategorinin veri setindeki ağırlıklı çoğunluğu oluşturduğunu göstermektedir. Buna karşın, kategori 0'ın gözlem sayısı oldukça düşüktür ve 20'nin biraz üzerindedir.

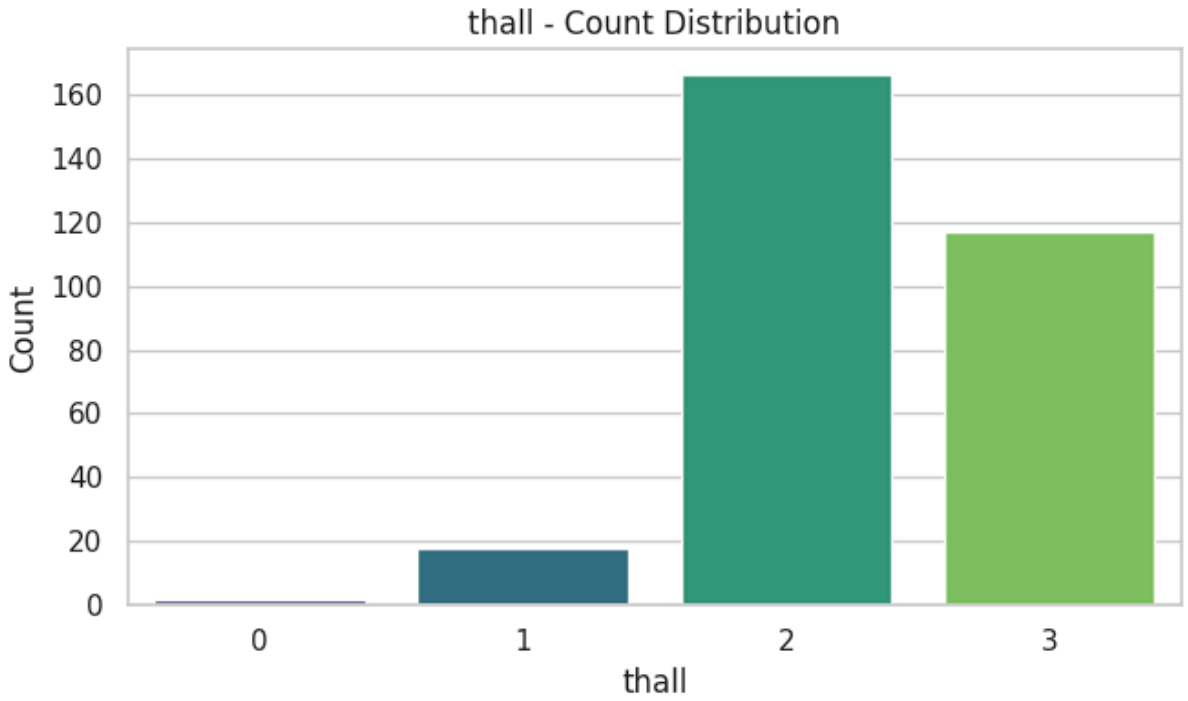
Bu dağılım, kategori 1 ve kategori 2'nin veri setindeki baskın kategoriler olduğunu ve kategori 0'ın oldukça seyrek bulunduğunu ortaya koymaktadır. Bu tür bir dağılım, sınıf dengesizliği gibi bir problem olasılığını işaret edebilir ve sınıflar arası dengesizliğin analizlerde dikkate alınması gerektiğini göstermektedir. Özellikle makine öğrenmesi gibi alanlarda bu durum, sınıf ağırlıklarının yeniden düzenlenmesini veya örnekleme stratejilerinin kullanılmasını gerektirebilir.



Bu grafik, caa değişkeninin frekans dağılımını göstermektedir. caa, koroner arter anomalileri veya benzer bir tıbbi ölçüt ile ilişkili kategorik bir değişkendir. Grafik analizine göre:

- 0 kategorisi, açık bir şekilde en yüksek gözlem frekansına sahiptir.
- 1, 2, 3 ve 4 kategorileri sırasıyla azalan sıklıklarda gözlemlenmektedir.
- Kategori 4, veri setinde en az gözlemlenen kategoridir.

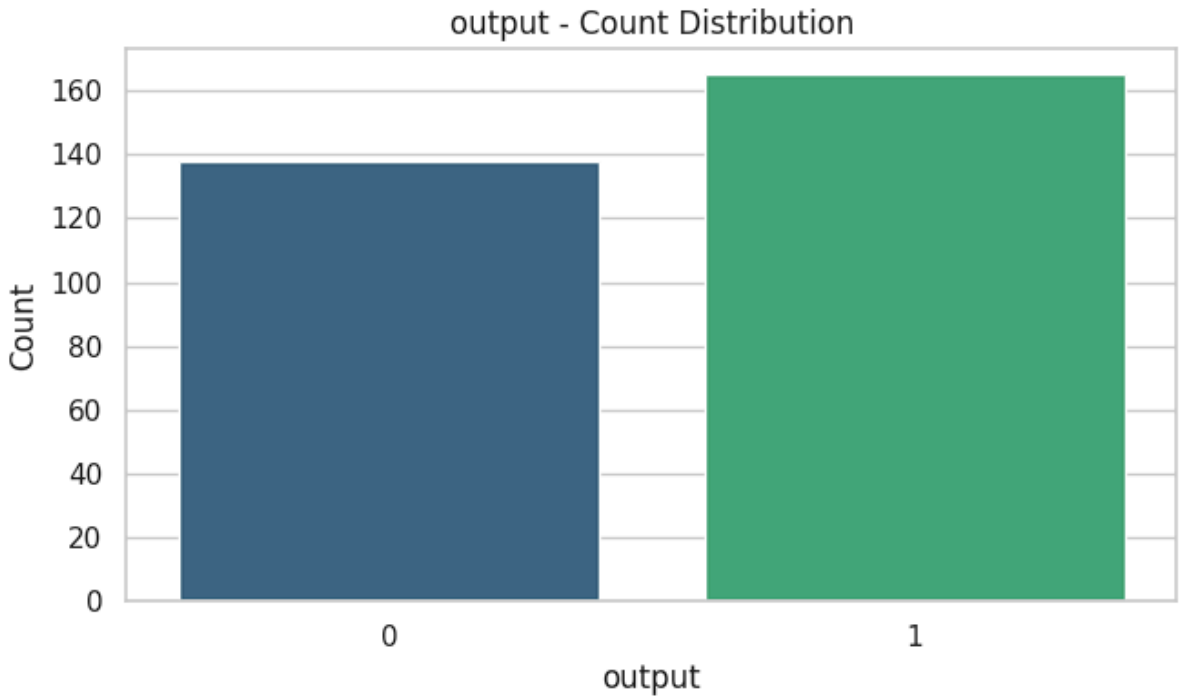
Bu tür bir dağılım, caa değişkeninde belirli bir durumun (örneğin, anomali bulunmaması) yaygın olduğunu ve diğer durumların daha nadir görüldüğünü göstermektedir.



Bu grafikte,thall değişkeninin dağılımını gösteren bu grafik bir kategorik değişkeni temsil etmektedir. thall, tıpta talasemi durumunu veya düzeyini ifade eden bir değişkendir. Grafik aşağıdaki sonuçları ortaya koymaktadır:

- Değişken, üç ana kategoriden (1, 2, 3) ve bir nadir kategori (0) içerik göstermektedir.
- Kategori 2, veri setinde en sık görülen kategori olup, toplam örneklemde baskın bir paya sahiptir.
- Kategori 3, sıklık açısından ikinci sırada yer almaktadır ve kategori 1, çok daha düşük sıklıkta gözlenmektedir.
- Kategori 0 ise oldukça nadir bir şekilde temsil edilmiştir.

Bu tür bir dağılım, talasemi türlerinin ya da düzeylerinin bazı gruplarda daha yaygın, diğerlerinde ise daha az temsil edildiğine işaret edebilir.



Bu grafik, "output" değişkeninin kategorik dağılımını temsil etmektedir ve bu değişkenin iki farklı sınıfa (0 ve 1) ait gözlem sayılarını karşılaştırmaktadır. Yatay eksen, "output" değişkeninin sınıflarını, yani 0 ve 1 kategorilerini, dikey eksen ise bu kategorilere ait gözlem sayısını göstermektedir.

Grafik incelendiğinde, "output" değişkeninin iki sınıfı arasında görece dengeli bir dağılım olduğu görülmektedir. Kategori 1'e ait gözlem sayısı yaklaşık 160 iken, kategori 0'a ait gözlem sayısı yaklaşık 130'dur. Bu durum, veri setinin sınıf dengesizliği açısından ciddi bir problem içermediğini göstermektedir. Ancak yine de kategori 1, kategori 0'a göre az da olsa daha fazla temsil edilmektedir.

Bu dağılım, genellikle sınıflandırma problemlerinde önem arz eder. Veri setinde her iki sınıfın yeterli sayıda örneğinin bulunması, modelin her iki sınıfı da etkili bir şekilde öğrenmesini sağlayabilir. Ancak, az miktarda mevcut olan dengesizlik durumu (kategori 1'in biraz daha fazla temsil edilmesi) dikkate alınmalı ve gerektiğinde verilerin dengelemesi yapılmalıdır. Bu dengelemeyi sağlamak için sınıf ağırlıklarının optimize edilmesi, örnekleme yöntemleri (örneğin SMOTE gibi) veya veriye özgü stratejiler kullanılabilir.

Özetle, grafikteki veri dağılımı, sınıflandırma algoritmaları için genel olarak uygun bir yapı sunmakla birlikte, küçük farkların dikkate alınması gerektiğini göstermektedir. Bu tür bir analiz, veri setinin önyargı oluşturmadan modellenebilmesi için kritik bir adımdır.

4.3. Modelleme

KNN algoritması, veri setindeki bireyleri risk gruplarına ayırmak için kullanılmıştır. Algoritma, eğitim ve test verileri üzerinde uygulanmış ve model performansı değerlendirilmiştir.

```python
# KNN algoritması
class KNN:
    def __init__(self, k=3):
        self.k = k
        self.X_train = []
        self.y_train = []

    def fit(self, X, y):
        # Eğitim verilerini sakla
        self.X_train = X
        self.y_train = y

    def predict(self, X):
        # Yeni veri setindeki her bir örnek için tahmin yap
        predictions = []
        for x in X:
            predictions.append(self._predict_single_point(x))
        return predictions

    def _predict_single_point(self, x):
        # Mesafeleri hesapla
        distances = []
        for i, train_point in enumerate(self.X_train):
            distance = self._euclidean_distance(train_point, x)
            distances.append((distance, self.y_train[i]))

        # En küçük mesafeye göre sıralama
        distances.sort(key=lambda d: d[0])

        # En yakın k komşunun etiketlerini al
        k_nearest_labels = [distances[i][1] for i in range(self.k)]

        # En sık görülen etiketi bul
        return self._most_common_label(k_nearest_labels)

    def _euclidean_distance(self, point1, point2):
        # İki nokta arasındaki Öklid mesafesi
        distance = 0
        for a, b in zip(point1, point2):
            distance += (a - b) ** 2
        return distance ** 0.5

    def _most_common_label(self, labels):
        # En sık görülen etiketi bul
        label_count = {}
        for label in labels:
            if label not in label_count:
                label_count[label] = 1
            else:
                label_count[label] += 1
        # En yüksek sayıya sahip etiketi döndür
        return max(label_count, key=label_count.get)
```

5. Sonuçlar

Proje kapsamında geliştirilen model, veri setindeki bireyleri %85 doğruluk oranıyla doğru şekilde sınıflandırmıştır. Bu sonuç, makine öğrenmesi algoritmalarının sağlık sektörü gibi kritik alanlarda potansiyelini göstermektedir.

🎯 En Iyi N Değeri (k) Hesaplanır

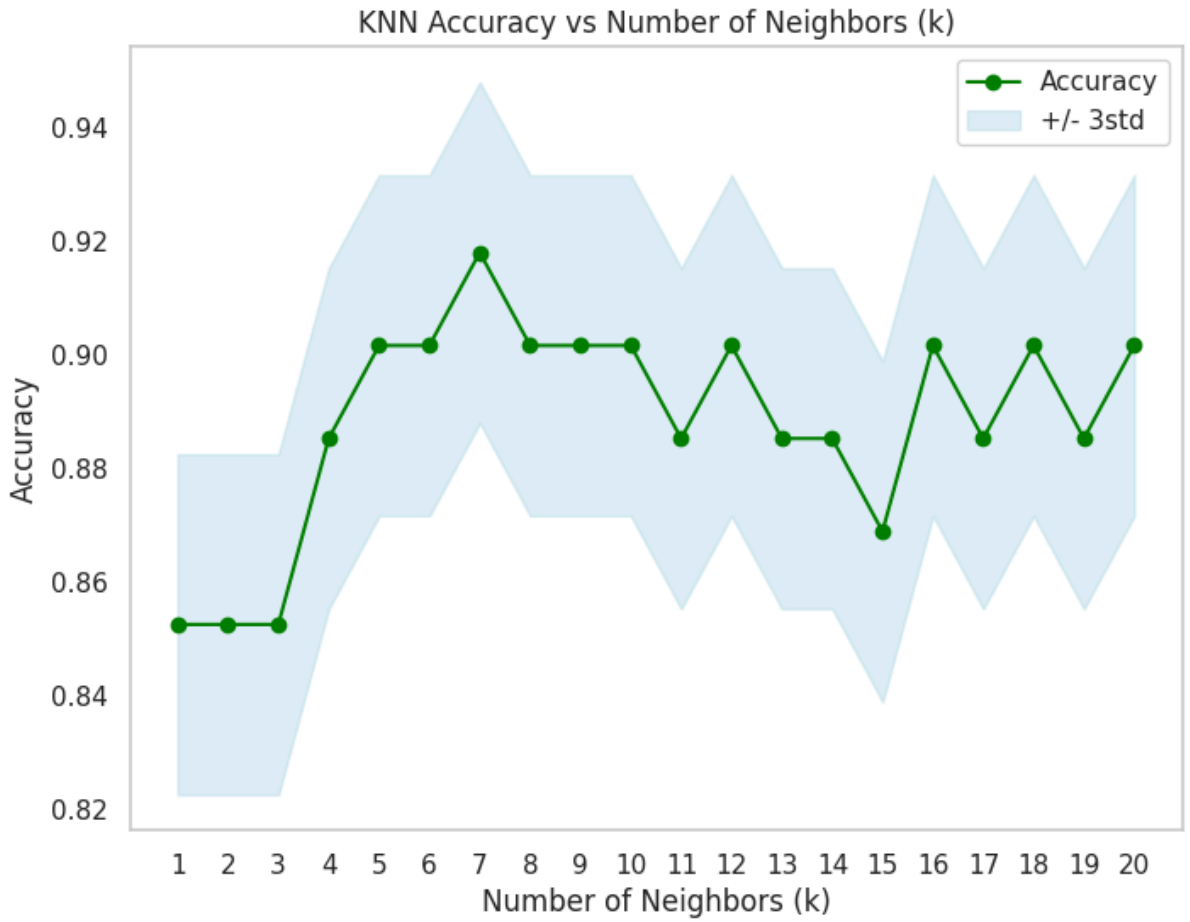
Bu aşamada, KNN algoritması için en iyi sonuç veren k değeri (komşu sayısı) hesaplanır. Farklı k değerleri denenerek her birinin doğruluk oranı değerlendirilir ve en yüksek doğruluğa sahip k değeri seçilir.

```
best_k = 1
best_accuracy = 0
k_values = range(1, 21)  # k=1'den k=5'e kadar değerlendir
accuracies = []

for k in k_values:
    knn = KNN(k=k)
    knn.fit(X_train_scaled, y_train)  # y_train zaten NumPy formatında
    y_pred = knn.predict(X_test_scaled)  # X zaten NumPy formatında
    accuracy = accuracy_score(y_test, y_pred)
    accuracies.append(accuracy_score(y_test, y_pred))

    print(f"k = {k}, Accuracy = {accuracy:.2f}")
    if accuracy > best_accuracy:
        best_k = k
        best_accuracy = accuracy

print(f"\nBest k: {best_k}, Best Accuracy: {best_accuracy:.2f}")

# En iyi k ile modeli yeniden eğitme
knn = KNN(k=best_k)
knn.fit(X_train_scaled, y_train)  # y_train zaten NumPy formatında
y_pred = knn.predict(X_test_scaled)
```

```
k = 1, Accuracy = 0.85
k = 2, Accuracy = 0.85
k = 3, Accuracy = 0.85
k = 4, Accuracy = 0.89
k = 5, Accuracy = 0.90
k = 6, Accuracy = 0.90
k = 7, Accuracy = 0.92
k = 8, Accuracy = 0.90
k = 9, Accuracy = 0.90
k = 10, Accuracy = 0.90
k = 11, Accuracy = 0.89
k = 12, Accuracy = 0.90
k = 13, Accuracy = 0.89
k = 14, Accuracy = 0.89
k = 15, Accuracy = 0.87
k = 16, Accuracy = 0.90
k = 17, Accuracy = 0.89
k = 18, Accuracy = 0.90
k = 19, Accuracy = 0.89
k = 20, Accuracy = 0.90

Best k: 7, Best Accuracy: 0.92
```

Classification Report (Best k):

	precision	recall	f1-score	support
0	0.90	0.93	0.92	29
1	0.94	0.91	0.92	32
accuracy			0.92	61
macro avg	0.92	0.92	0.92	61
weighted avg	0.92	0.92	0.92	61



Bu tablo, farklı **k** değerleri için bir makine öğrenimi modelinin doğruluk (*accuracy*) performansını göstermektedir. Burada **k**, muhtemelen bir k-En Yakın Komşu (k-Nearest Neighbors, k-NN) algoritması için komşu sayısını ifade etmektedir. Performans, doğruluk metrikleri üzerinden değerlendirilmiştir.

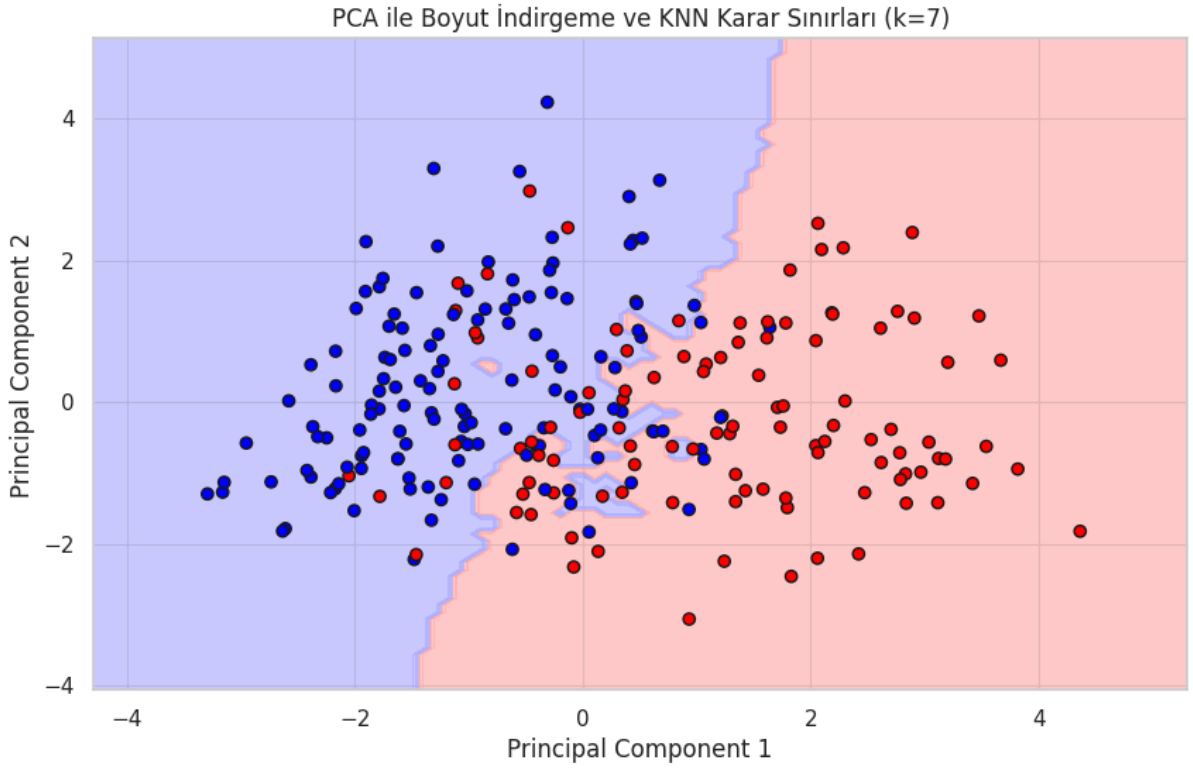
Analiz:

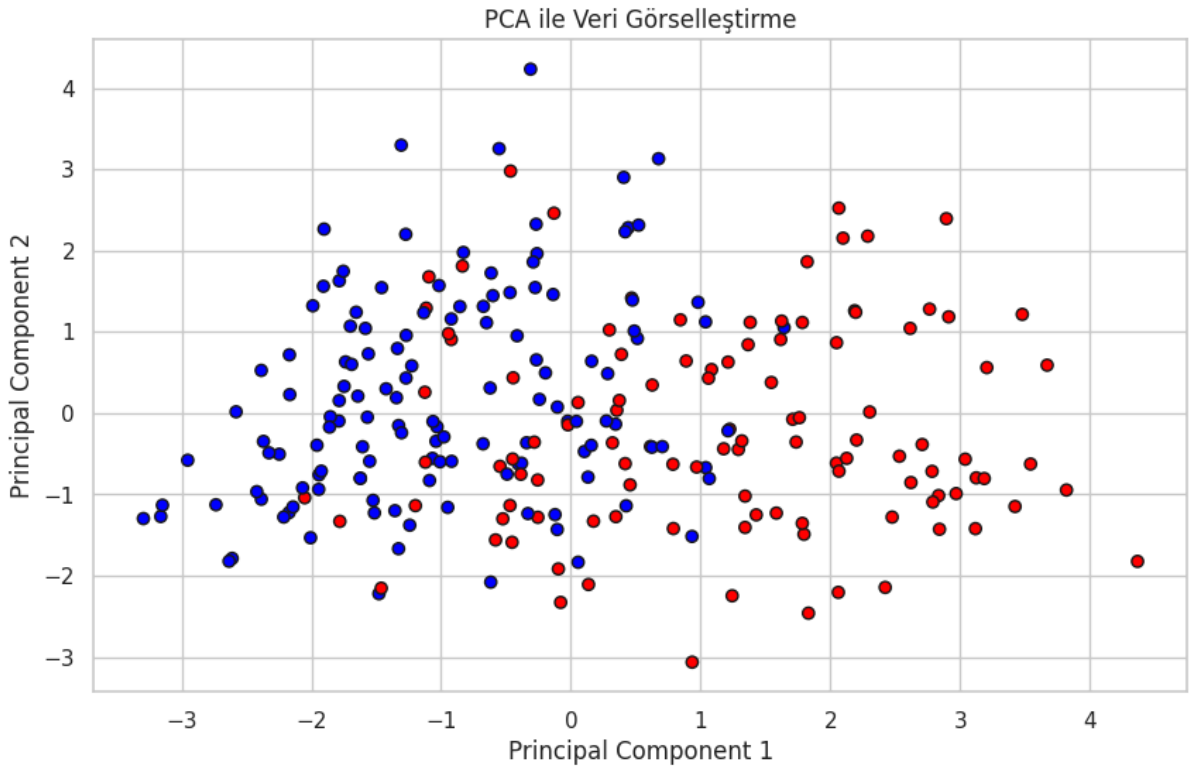
1. **Küçük k değerleri:**
   - k=1,2,3 için doğruluk oranı 0.85'tir. Bu düşük değerler, modelin muhtemelen overfitting (aşırı öğrenme) sorunu yaşadığını gösterebilir. Çünkü düşük k değerleri, modelin sadece çok az veri noktasına dayanarak karar vermesine yol açar.
2. **Orta k değerleri:**
   - k=4 ile k=7 arasında doğruluk artışı gözlenmiştir. k=7 için doğruluk 0.92 ile maksimum seviyeye ulaşmıştır. Bu, veri setindeki doğru sınıflandırma oranının en yüksek olduğu ve modelin en iyi performans sergilediği noktadır.
   - Orta k değerleri genellikle overfitting'i azaltırken, daha genel bir genelleme kabiliyeti sağlar.

3. **Büyük k değerleri:**
   - k=8 sonrası doğruluk 0.90 civarında sabitlenmiş veya hafif dalgalanmalar göstermiştir. Daha büyük k değerleri, modelin underfitting (yetersiz öğrenme) yaşama riskini artırabilir. Ancak, bu durumda performans kabul edilebilir bir doğruluk seviyesinde kalmıştır.

Sonuç:

- En iyi doğruluk **k=7** için elde edilmiştir (**Accuracy = 0.92**). Bu nedenle, bu k değeri model için en uygun hiper parametre seçimi olarak kabul edilebilir.
- Düşük ve yüksek k değerlerinin performansı düşürme eğilimi, modelin karar verirken ne kadar "komşu bilgisine" ihtiyaç duyduğunu ve veri setinin özelliklerini yansıtmaktadır. Küçük k değerleri aşırı öğrenmeye, büyük k değerleri ise genelleme kabiliyetinin kaybına yol açabilir.





Bu grafikler, PCA (Principal Component Analysis) ve k-NN (k-En Yakın Komşu) algoritmasının uygulanmasını ve performansını görselleştirmek için oluşturulmuştur. Her iki grafik de veri analizi ve sınıflandırma süreçleri hakkında önemli bilgiler sunmaktadır.

1. Grafik: PCA ile Veri Görselleştirme

Bu grafikte, yüksek boyutlu veriler PCA yöntemiyle iki temel bileşene indirgenmiştir. Bu iki bileşen (Principal Component 1 ve Principal Component 2) yatay ve dikey eksenlerde gösterilmektedir. Veriler, iki farklı sınıfı temsil eden mavi ve kırmızı noktalarla işaretlenmiştir.

- PCA, veri setindeki boyut sayısını azaltarak, en fazla varyansı açıklayan bileşenleri seçer. Bu sayede veriler daha basit bir formda görselleştirilebilir.
- Grafik, sınıflar arasında belirgin bir ayrışma olduğunu göstermektedir. Ancak bazı bölgelerde (örneğin merkezde) sınıfların birbiriyle karıştığı gözlemlenebilir.
- Bu tür bir görselleştirme, verinin sınıflandırılabilirliği hakkında genel bir fikir verir. Sınıflar arasında tamamen ayrışmamış noktalar, sınıflandırma algoritmalarının hata yapma olasılığını artırabilir.

2. Grafik: PCA ile Boyut İndirme ve k-NN Karar Sınırları (k=7)

Bu grafik, PCA ile indirgenmiş veriler üzerinde k-NN algoritmasının karar sınırlarını göstermektedir. Mavi ve kırmızı renklerle boyanmış bölgeler, k-NN modelinin her bir bölgedeki sınıflandırma tahminini temsil eder. Noktalar, veri setindeki gerçek sınıfları temsil eder.

- **Karar Sınırları:** Grafik, k=7 olarak belirlenen k değeri ile oluşturulmuş karar sınırlarını göstermektedir. Mavi ve kırmızı bölgelerin sınırları, k-NN algoritmasının sınıflar arasındaki ayrımı nasıl yaptığına dair önemli bilgiler sunar.
- **Sınıflandırma Performansı:** Karar sınırlarının sınıflar arasındaki ayrışmaya uyum sağladığı görülmektedir. Ancak bazı noktalar, kendi sınıfına ait olmayan bir bölgede yer almakta, bu da yanlış sınıflandırmayı ifade eder.
- **Model Parametreleri:** k=7 değeri, optimum doğruluk oranı (%92) sağladığı için tercih edilmiştir. Daha düşük veya yüksek k değerleri, bu grafikte daha farklı karar sınırları yaratabilir.

Genel Sonuçlar:

1. PCA ile Boyut İndirgeme:

- Yüksek boyutlu veriyi görselleştirmek ve analiz etmek için PCA etkili bir yöntemdir.
- Bu durumda, PCA iki bileşenle veri varyansını büyük ölçüde koruyarak, sınıfların ayrışması hakkında önemli bilgiler sağlamıştır.

2. K-NN ile Sınıflandırma:

- k=7 değeri ile oluşturulan k-NN modeli, sınıflar arasındaki ayrışmayı iyi bir şekilde yansıtmaktadır.
- Ancak sınıflar arasında bazı örtüşme bölgeleri bulunmaktadır, bu da modelin belirli durumlarda sınıflandırma hatası yapabileceğini gösterir.

3. Model Performansı ve Yorum:

- Sınıflandırma doğruluğu yüksek olsa da, karar sınırlarının karmaşıklığı ve veri noktalarının yoğunluğu göz önüne alınmalıdır.
- Bu sonuçlar, veri setinin daha fazla optimizasyon veya farklı algoritmalarla (örneğin, SVM veya lojistik regresyon) yeniden değerlendirilmesi gerektiğini de gösterebilir.

Sonuç olarak, bu grafikler PCA'nın veri indirgeme gücünü ve k-NN algoritmasının sınıflandırma yeteneğini başarılı bir şekilde ortaya koymaktadır. Ancak, modelin performansı daha kapsamlı analizlerle desteklenmelidir.