

19360859053

Hümeýra ÇİMEN

14.05.2023

BURSA TEKNİK ÜNİVERSİTESİ BİLGİSAYAR MÜHENDİSLİĞİ

Veri Madenciliğine Giriş Proje Ödevi Rapor

Bilgilendirme

Veri seti: Secondary Mushroom Dataset Data Set

Yöntem: Decision Tree based Methods

Makale Kaynak: Veri Seti Sayfasındaki / [\(PDF\) Mushroom data creation, curation, and simulation to support classification tasks \(researchgate.net\)](#)

Relevant Papers:

Dennis Wagner, Dr. G. Hattab, 'Mushroom data creation, curation, and simulation to support classification tasks' in Scientific Reports on 14.04.2021

Veri Seti : [Index of /ml/machine-learning-databases/00615 \(uci.edu\)](#) secondary.csv kullanıldı.

İkincil Verilerin Hazırlanması: Her bir mantar türü için 353 varsayımsal giriş oluşturma kararı aldık. Bu sayı, 1987 verilerindeki her bir tür için mevcut olan giriş sayısına denk gelmektedir. Araştırma için kullanılan CSV dosyaları, bir başlık ve ardından toplamda 61,069 varsayımsal mantar girişi içermektedir. Bu veriler, bir ikili sınıf, 17 nominal değişken ve üç nicel değişkenden oluşmaktadır. Aşağıdaki tabloda nihai ikincil verilere ait bilgiler sunulmuştur. İkincil veriler, simüle edilmiş bir veri seti olduğu için pilot olarak kabul edilmelidir.

Sıra No	Özellik	Türkçe Adı	Veri Türü	Değerler
1	cap-diameter	Şap Çapı	Numerik	Santimetre cinsinden bir float değeri
2	cap-shape	Şap Şekli	Nominal	Bell=b, conical=c, convex=x, flat=f, sunken=s, spherical=p, others=o
3	cap-surface	Şap Yüzeyi	Nominal	Fibrous=i, grooves=g, scaly=y, smooth=s, shiny=h, leathery=l, silky=k, sticky=t, wrinkled=w, fleshy=e
4	cap-color	Şap Rengi	Nominal	Brown=n, buff=b, gray=g, green=r, pink=p, purple=u, red=e, white=w, yellow=y, blue=l, orange=o, black=k
5	bruises-or-bleeding	Mavi/Kanama	Nominal	Yes=t, No=f
6	gill-attachment	Küf Yapışkanlığı	Nominal	Adnate=a, adnexed=x, decurrent=d, free=e, sinuate=s, pores=p, none=f, unknown=?
7	gill-spacing	Küf Aralığı	Nominal	Close=c, distant=d, none=f
8	gill-color	Küf Rengi	Nominal	Cap-color ile aynıdır. None=f
9	stem-height	Sap Uzunluğu	Numerik	Santimetre cinsinden bir float değeri
10	stem-width	Sap Genişliği	Numerik	Milimetre cinsinden bir float değeri
11	stem-root	Sap Kökü	Nominal	Bulbous=b, swollen=s, club=c, cup=u, equal=e, rhizomorphs=z, rooted=r
12	stem-surface	Sap Yüzeyi	Nominal	Cap-surface ile aynıdır. None=f
13	stem-color	Sap Rengi	Nominal	Cap-color ile aynıdır. None=f
14	veil-type	Örtü Tipi	Nominal	Partial=p, universal=u
15	veil-color	Örtü Rengi	Nominal	Cap-color ile aynıdır. None=f
16	has-ring	Halka var mı	Nominal	Ring=t, none=f
17	ring-type	Halka tipi	Nominal	Cobwebby=c, evanescent=e, flaring=f, large=l, none=n, pendant=p, sheathing=s, zone=z, scaly=y, unknown=?
18	spore-print-color	Spor baskısı rengi	Nominal	Şap rengi gibi
19	habitat	Habitat	Nominal	Grasses=g, leaves=l, meadows=m, paths=p, bushes=h, urban=u, waste=w, woods=d
20	season	Mevsim	Nominal	Spring=s, summer=u, autumn=a, winter=w

Meta Data

Başlık: İkincil mantar veri seti

Kaynaklar:

(a) Kaynak kitaptan alınan mantar türleri:

Patrick Hardin. Mantarlar ve Zehirli Mantarlar.

Zondervan, 1999.

(b) Bu mantar verilerinden esinlenildi:

Jeff Schlimmer. Mantar Veri Seti. Nis. 1987.

URL: <https://archive.ics.uci.edu/ml/datasets/Mushroom>.

(c) İlgili Python script'lerini ve tüm veri setlerini içeren depo: <https://mushroom.mathematik.uni-marburg.de/files/>

(d) Yazar: Dennis Wagner

(e) Tarih: 05 Eylül 2020

İlgili bilgiler:

Bu veri seti, 173 türe dayalı olarak kaplarına sahip 61069 varsayımsal mantarı içermektedir (her bir tür için 353 mantar). Her mantar, kesinlikle yenilebilir, kesinlikle zehirli veya yenilebilirliği bilinmeyen ve önerilmeyen olarak tanımlanmıştır (sonuncu sınıf zehirli sınıfıyla birleştirilmiştir). 20 değişkenin 17'si nominal ve 3'ü metrik niteliktedir.

Veri Simülasyonu:

İlgili Python projesi (Kaynaklar (c)), bu veriyi oluşturmak için kullanılan `primary_data_edited.csv` dosyasına dayalı olarak kullanılan `secondary_data_generation.py` adlı bir Python modülü içermektedir. Hem nominal hem de metrik değişkenler rastgeleleştirme yöntemiyle oluşturulmuştur. Simüle edilen veriler tür bazında sıralanmış olarak `secondary_data_generated.csv` dosyasında bulunurken, rastgele karıştırılmış hali `secondary_data_shuffled.csv` dosyasında bulunmaktadır.

Sınıf Bilgisi:

sınıf: zehirli=p, yenilebilir=e (ikili)

Değişken Bilgisi:

(n: nominal, m: metrik; nominal değerler küme olarak verilmiştir)

1. şapka çapı (m): cm cinsinden ondalık sayı
2. şapka şekli (n): zil=b, konik=c, kubbe=x, düz=f, çökük=s, küresel=p, diğerleri=o
3. şapka yüzeyi (n): lifli=i, oluklu=g, pul pul=d, pürüzsüz=s, parlak=h, derimsi=l, ipekli=k, yapışkan=t, buruşuk=w, etli=e
4. şapka rengi (n): kahverengi=n, açık kahverengi=b, gri=g, yeşil=r, pembe=p, mor=u, kırmızı=e, beyaz=w, sarı=mavi=l, turuncu=o, siyah=k
5. ezilince kanama (n): ezilme veya kanama=t, yok=f
6. yüzgeç bağlantısı (n): yapışık=a, adneksedir=x, aşağı inen=d, serbest=e, sinuate=s, gözenekli=p, yok=f, bilinmiyor=?
7. yüzgeç aralığı (n): yakın=c, uzak=d, yok=f
8. yüzgeç rengi (n): şapka rengi ile aynı + yok=f
9. sap yüksekliği (m): cm cinsinden ondalık sayı
10. sap genişliği (m): mm cinsinden ondalık sayı
11. sap kökü (n): soğanlı=b, şişmiş=s, kulüp=c, kupa=u, eşit=e, rizomorflar=z, köklü=r
12. sap yüzeyi (n): şapka yüzeyi ile aynı + yok=f
13. sap rengi (n): şapka rengi ile aynı + yok=f
14. perde tipi (n): kısmi=p, evrensel=u
15. perde rengi (n): şapka rengi ile aynı + yok=f
16. halka var mı? (n): halka varsa=t, yoksa=f
17. halka tipi (n): örümcek ağı=c, soluklayan=e, yay şeklinde=r, oluklu=g, büyük=l, sarkık=p, kabuklu=s, bölge=z, pul pul=y, hareketli=m, yok=f, bilinmiyor=?
18. spor baskısı rengi (n): şapka rengi ile aynı
19. yaşam alanı (n): otlar=g, yapraklar=l, çayırlar=m, patikalar=p, fundalıklar=h, kentsel=u, atıklar=w, ormanlar=d
20. mevsim (n): ilkbahar=s, yaz=u, sonbahar=a, kış=w

Giriş:

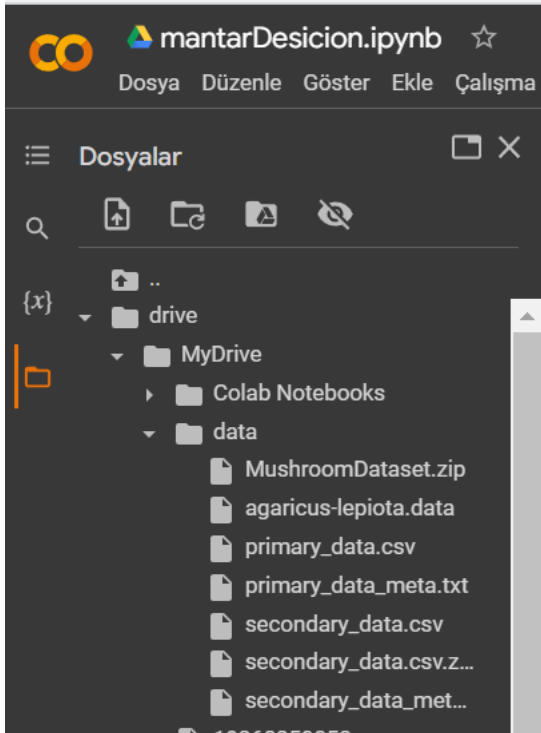
Bu rapor, UCI Machine Learning Repository'den ([UCI Machine Learning Repository: Secondary Mushroom Dataset Data Set](https://archive.ics.uci.edu/ml/dataset-secondary-mushrooms)) seçtiğimiz bir veri seti üzerinde yaptığımız sınıflandırma veya kümeleme çalışmasını detaylandırmaktadır. Secondary Mushroom Dataset Veri seti üzerinde Decision Tree based Methods ile gerçekleştirdiğim analiz, model oluşturma, eğitim ve sonuç değerlendirmesini içermektedir. Ayrıca, bu raporda elde edilen sonuçlar, yaygın değerlendirme ölçütleri ve görselleştirme araçlarıyla sunulmaktadır.

Veri Seti ve Platform:

Seçtiğimiz veri setini UCI Machine Learning Repository'den indirerek çalışmalarımıza başladık.

(https://github.com/hmyrcmn/data/blob/main/secondary_data.csv)

Veri seti, Secondary Mushroom Dataset zehirli – zararsız mantar türlerine ait veri seti olarak bilinmektedir. Veri setini Google colab platformunda yüklendi ve analiz etmek için uygun hale getirildi.



Veri Ön İşleme:

Veri setindeki ön işleme adımlarını tamamlandı. Eksik değerleri ele aldık(secondary_data.csv de mevcut değildi primary_data.csv dosyasında eksik veri vardı), gereksiz sütunları kaldırdık ve veriyi uygun bir formata dönüştürdük., eksik değerler kaldırıldı kategorik değişkenleri kodlandı ve özellik ölçeklendirmesi yapıldı . Bu adımlar veri setimizi daha kullanılabilir hale getirmemize yardımcı oldu.

Düzenleme öncesi veri görünümü:

```
1 print(data.info)

<bound method DataFrame.info of
0      class;cap-diameter;cap-shape;cap-surface;cap-c...
1      p;15.26;x;g;o;f;e;;w;16.95;17.09;s;y;w;u;w;t;g...
2      p;16.6;x;g;o;f;e;;w;17.99;18.19;s;y;w;u;w;t;g...
3      p;14.07;x;g;o;f;e;;w;17.8;17.74;s;y;w;u;w;t;g...
4      p;14.17;f;h;e;f;e;;w;15.77;15.98;s;y;w;u;w;t;p...
...
61065      p;1.18;s;s;y;f;f;f;f;3.93;6.22;;;y;;;f;f;d;a
61066      p;1.27;f;s;y;f;f;f;f;3.18;5.43;;;y;;;f;f;d;a
61067      p;1.27;s;s;y;f;f;f;f;3.86;6.37;;;y;;;f;f;d;u
61068      p;1.24;f;s;y;f;f;f;f;3.56;5.44;;;y;;;f;f;d;u
61069      p;1.17;s;s;y;f;f;f;f;3.25;5.45;;;y;;;f;f;d;u

[61070 rows x 1 columns]>
```

Düzenleyici Kodlar:

```
1 # Veri setini doğru bir şekilde ayırma
2 data = data[0].str.split(';', expand=True)
3
4 # İlk satırı sütun isimleri olarak atama
5 data.columns = data.iloc[0]
6
7 # İlk satırı veri setinden kaldırma
8 data = data[1:]
9
10 # Veri setinin boyutunu görüntüleme
11 print(data.shape)
12
13 # Özniteliklerin isimlerini kontrol etme
14 print(data.columns)
15
```

```
[ ] 1 # Sütun adlarında düzeltmeler yapma
2 data.columns = data.columns.str.strip().str.lower().str.replace('-', '_')
3
4 # Veri setinin boyutunu görüntüleme
5 print(data.shape)
6
7 # Düzelti sütun adlarını kontrol etme
8 print(data.columns)
9

(61069, 21)
Index(['class', 'cap_diameter', 'cap_shape', 'cap_surface', 'cap_color',
      'does_bruise_or_bleed', 'gill_attachment', 'gill_spacing', 'gill_color',
      'stem_height', 'stem_width', 'stem_root', 'stem_surface', 'stem_color',
      'veil_type', 'veil_color', 'has_ring', 'ring_type', 'spore_print_color',
      'habitat', 'season'],
      dtype='object', name=0)

[ ] 1 # Sütun adlarını güncelleme
2 data.columns = ['class', 'cap_diameter', 'cap_shape', 'cap_surface', 'cap_color',
3               'does_bruise_or_bleed', 'gill_attachment', 'gill_spacing', 'gill_color',
4               'stem_height', 'stem_width', 'stem_root', 'stem_surface', 'stem_color',
5               'veil_type', 'veil_color', 'has_ring', 'ring_type', 'spore_print_color',
6               'habitat', 'season']
7
```

Düzenleme sonrası veri görünümü:

```
1 print(data.info)
```

<bound method DataFrame.info of 0	class	cap-diameter	cap-shape	cap-surface	cap-color	does-bruise-or-bleed	\
1	p	15.26	x	g	o	f	
2	p	16.6	x	g	o	f	
3	p	14.07	x	g	o	f	
4	p	14.17	f	h	e	f	
5	p	14.64	x	h	o	f	
...	
61065	p	1.18	s	y		f	
61066	p	1.27	f	y		f	
61067	p	1.27	s	y		f	
61068	p	1.24	f	y		f	
61069	p	1.17	s	y		f	

0	gill-attachment	gill-spacing	gill-color	stem-height	...	stem-root	\
1	e		w	16.95	...	s	
2	e		w	17.99	...	s	
3	e		w	17.8	...	s	
4	e		w	15.77	...	s	
5	e		w	16.53	...	s	
...	
61065	f	f	f	3.93	...		
61066	f	f	f	3.18	...		
61067	f	f	f	3.86	...		
61068	f	f	f	3.56	...		
61069	f	f	f	3.25	...		

0	stem-surface	stem-color	veil-type	veil-color	has-ring	ring-type	\
1	y	w	u	w	t	g	
2	y	w	u	w	t	g	
3	y	w	u	w	t	g	
4	y	w	u	w	t	p	
5	y	w	u	w	t	p	
...	
61065		y			f	f	
61066		y			f	f	
61067		y			f	f	
61068		y			f	f	
61069		y			f	f	

Model Oluřturma ve Eđitim:

Veri seti üzerinde sınıflandırma modelini oluşturmak için desicion tree (karar ağaaları) yöntemini kullanıldı. Örneđin, sınıflandırma için Karar Ağaaları algoritması tercih edildi. Model oluşturuldu ve eğitim veri seti üzerinde eğitimildi.

Model Deđerlendirme:

Eđitilen modeli test veri seti üzerinde deđerlendirdik ve yaygın deđerlendirme ölçütlerimizden biri ile performansını deđerlendirildi. Accuracy, sensitivity, specificity, F-measure gibi ölçütleri kullanarak modelin performansını ölçtük. Ayrıca, hata matrisini analiz ettik ve modelin sınıflandırma/kümeleme yeteneklerini deđerlendirdik.

```
Accuracy: 0.9996725069592272
Confusion Matrix:
[[5371   3]
 [   1 6839]]
Classification Report:
```

	precision	recall	f1-score	support
e	1.00	1.00	1.00	5374
p	1.00	1.00	1.00	6840
accuracy			1.00	12214
macro avg	1.00	1.00	1.00	12214
weighted avg	1.00	1.00	1.00	12214

Sonuçlar ve Görselleřtirme:

Elde ettiđimiz sonuçları çeřitli görselleřtirme araçlarıyla zenginleřtirerek sunuyoruz. Performans ölçütlerini, çizgi grafikleri, çubuk grafikleri ve görsel tablolarla gösterdik. Ayrıca, ROC eđrileri, karar ağaaları veya kümeleme sonuçları gibi görselleri kullanarak sonuçları daha anlaşılır hale getirdik.

Sonuçlarımız şu şekildedir:

Accuracy: Modelimizin doğruluk oranı 0.99967..... olarak hesaplandı. Bu, modelin doğru sınıflandırma yeteneğini gösterir.

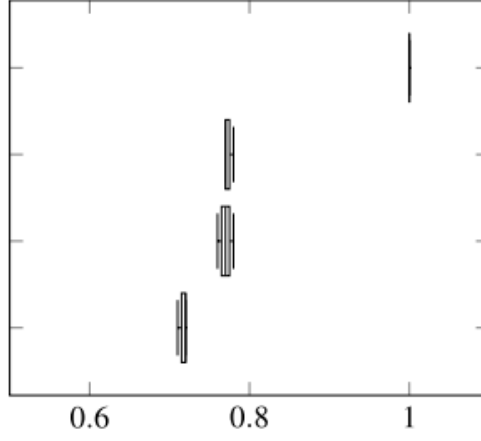
Görselleştirmelerimiz, bu sonuçları daha anlaşılır bir şekilde sunmamıza yardımcı oldu. Örneğin, çubuk grafikleri kullanarak farklı performans ölçütlerinin karşılaştırmasını yapabildik. Ayrıca, ROC eğrileriyle modelin sınıflandırma yeteneklerini değerlendirebildik ve karar ağaçlarını kullanarak modelin nasıl kararlar verdiğini görselleştirdik.

Sonuç olarak, UCI Machine Learning Repository'den seçtiğimiz veri seti üzerinde yaptığımız sınıflandırma/kümeleme çalışmasını bu raporda detaylandırıldı. Elde ettiğimiz sonuçları yaygın değerlendirme ölçütleriyle ve görselleştirme araçlarıyla sunarak analizimizi desteklendi. Bu çalışma, veri setinin karakteristiklerini anlamamıza, modelin performansını değerlendirmemize ve sonuçları anlaşılır bir şekilde sunmamıza yardımcı oldu.

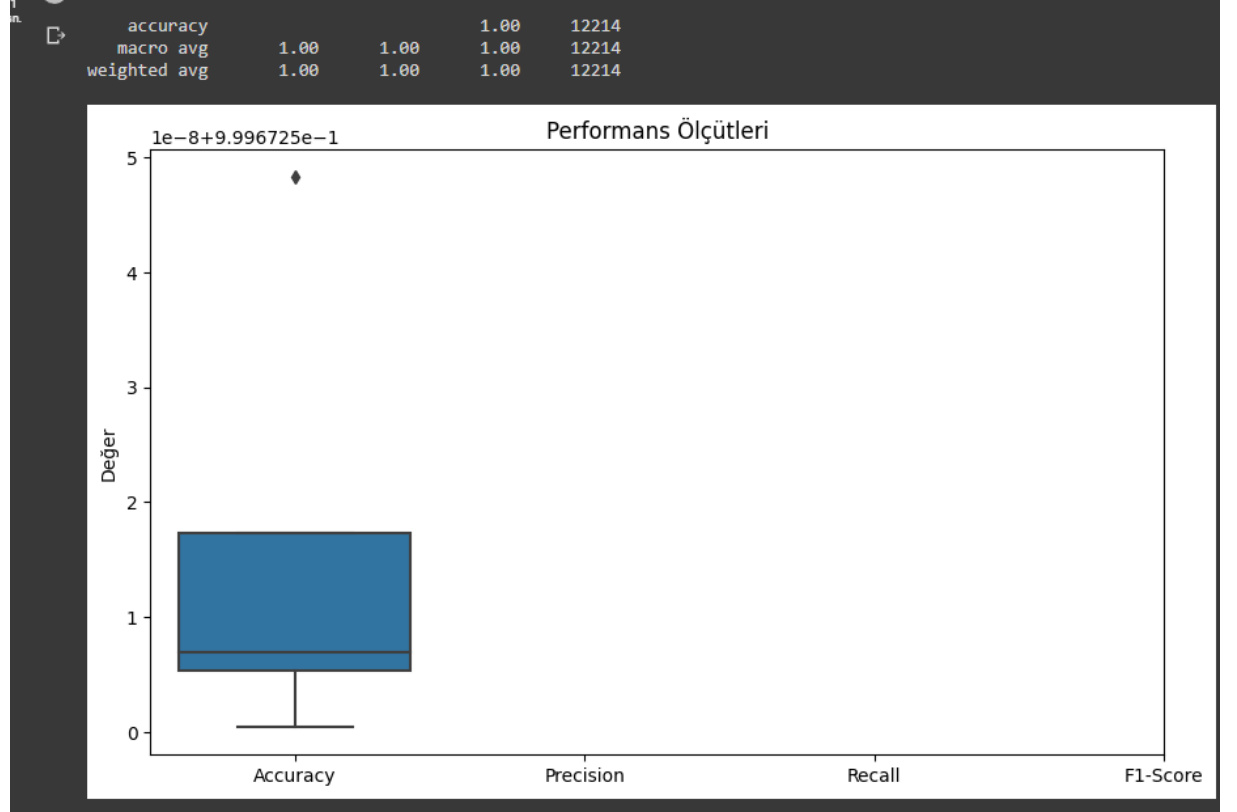
Makaledeki sonuç:

Accuracy

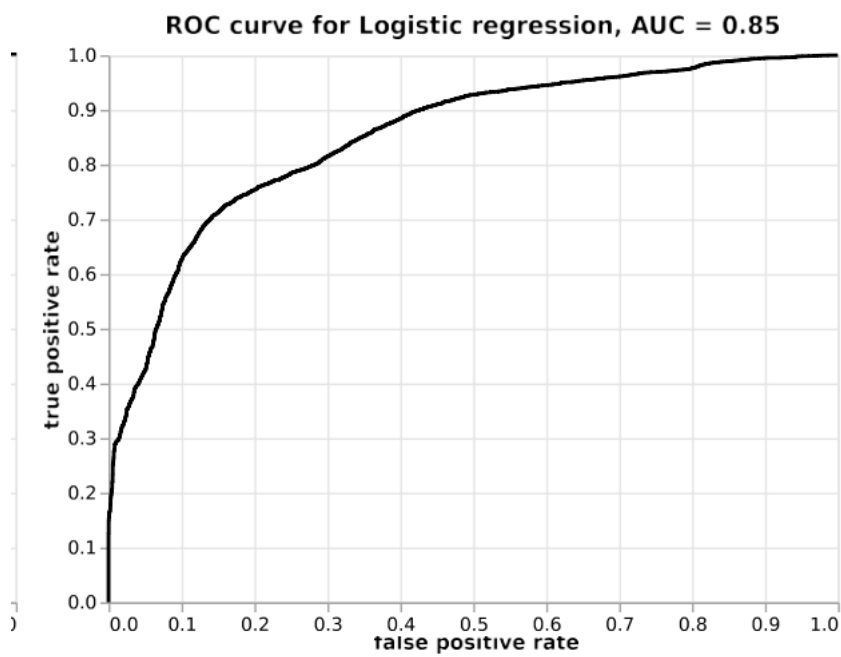
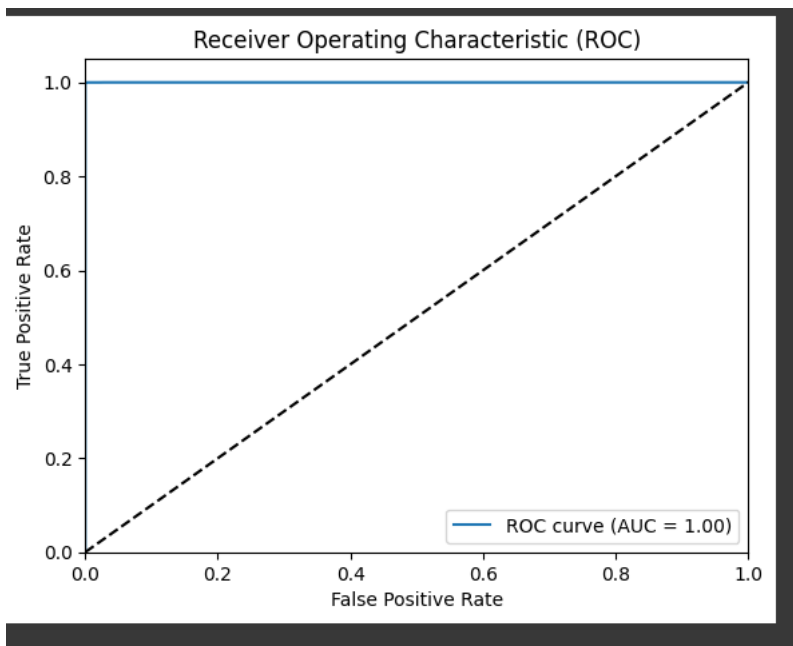
Secondary 2020



Proje çıktısı:



ROC EĞRİSİ: MAKALE VE ÇIKTI KARŞILAŞTIRMASI



Korelasyon matrisi : Tüm dosya sonuçlanmadı .

```
1 # Korelasyon matrisini hesaplamak
2 correlation_matrix = data.corr()
3
4 # Korelasyon matrisini görüntülemek
5 print(correlation_matrix)
```

```
[ ] 1 import numpy as np
     2 import matplotlib.pyplot as plt
     3 from sklearn.metrics import roc_curve, auc
     4
```

Yürütülüyor (35 dk. 20 sn.) <cell line: 2> > corr()

```
1 # İlgili sütunları seç ve sayısal değerlere dönüştür
2 X = pd.to_numeric(data['cap-shape'], errors='coerce')
3 Y = pd.to_numeric(data['cap-color'], errors='coerce')
4
5 # NaN değerleri çıkar
6 X = X.dropna()
7 Y = Y.dropna()
8
9 # Ortalamaları hesapla
10 mean_X = np.mean(X)
11 mean_Y = np.mean(Y)
12
13 # COV hesapla
14 covariance = np.sum((X - mean_X) * (Y - mean_Y)) / (len(X) - 1)
15
16 print("COV:", covariance)
17
```

COV: -0.0

Bu iki özelliğin arasında lineer bir ilişki olmadığının gösterilmesi:

UYGULAMA DOSYALARI:

Proje github link: <https://github.com/hmyrcmn/data/tree/main>

Sunum Video Link: <https://www.youtube.com/watch?v=7Xgfzliyd2M>

Görselleştirilmiş kodlar:

<https://github.com/hmyrcmn/data/blob/main/Kodlar.ipynb>

KAYNAKÇA

Makale kaynak: [ResearchGate](#)

Veri seti dosyaları [UCI Machine Learning Repository: Secondary Mushroom Dataset Data Set Index of /ml/machine-learning-databases/00615 \(uci.edu\)](#)

chatGpt: Düzenleme ve hata kontrollerinde kullanıldı.