

## İSTANBUL GELİŞİM MESLEK YÜKSEKOKULU BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ BİLGİSAYAR PROGRAMCILIĞI PROGRAMI

## PYTHON DİLİ İLE CİLT HASTALIKLARI

## VERİ SETİNİN İNCELENMESİ

**Hazırlayan:**

**220111699 - Hatice AYVA**

**210111425 - Emirhan GÜL**

**220111703 - Beyzanur MURAT**

**Ödev Danışmanı: Öğr.Gör.Tuğba**

**SARAY ÇETİNKAYA**

# ÖDEV TANITIM FORMU

YAZAR ADI SOYADI : HaticeAYVA, Beyzanur MURAT, Emirhan GÜL ,

ÖDEVİN DİLİ : Türkçe

ÖDEVİN ADI : Python Dili İle Cilt Hastalıkları Veri Setinin İncelenmesi

BÖLÜM : Bilgisayar Teknolojileri PROGRAM : Bilgisayar Programcılığı

ÖDEVİN TÜRÜ : Proje

ÖDEVİN TES. TARİHİ : 02.06.2023

SAYFA SAYISI : 20

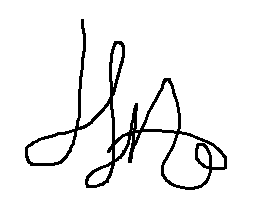
ÖDEV DANIŞMANI : Öğr. Gör. Tuğba SARAY ÇETİNKAYA

# BEYANNAME

Bu proje ödevini grup arkadaşlarım Beyzanur MURAT ve Emirhan GÜL ile birlikte hazırlamış olup. Yapılan bu projeyi baştan sona danışmanım Tuğba SARAY ÇETİNKAYA’nın sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi çalışma sürecinde etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.

09.05.2023

Hatice AYVA



# KABUL VE ONAY SAYFASI

220111699 numaralı Hatice AYVA’nın Python Dili İle Cilt Hastalıkları Veri Setinin İncelenmesi adlı çalışması,benim tarafımdan proje ödevi olarak kabul edilmiştir.

Tuğba SARAY ÇETİNKAYA Öğretim Görevlisi

# ÖZET

Bu çalışma, HAM10000\_metadata veri setini kullanarak sivilce, akne ya da deri üzerinde

oluşan değişim için bir derin öğrenme modeli geliştirmeyi amaçlamaktadır.Amacımız insanların

deri üzerinde oluşan değişim hakkında anında bilgi alabilmesini sağlamaktır. Veri setimiz 10.013

veriden oluşmaktadır. Veri setimizin ayrıntılı incelemesi aşağıda yapılmıştır. Çalışmamızda aynı veya

aynı konuda benzer veri setlerinin kullanıldığı dört adet çalışmanın makalesini de inceledik ve bizim

çalışmamız ile karşılaştırmalar yapıp yorumladık. Bu çalışma sürecinde Makine Öğrenmesi, Uygulama Geliştirme aşamaları uygulanmıştır. Bu aşamalar aşağıda yer alan çalışmamızda daha ayrıntılı

açıklanmıştır.

Bu çalışmamızda yaşa göre sivilce, akne ya da deri üzerinde oluşan değişimin hangi bölgede görüldüğünü ve cinsiyete göre sivilce, akne ya da deri üzerinde oluşan değişimin hangi bölgede

görüldüğünü grafik üzerinden inceleyip yorumladık.

Çalışmamızda regresyon modelleri uygulamayı denedik ama veri setimizde sadece yaş sütunu

int türünde olduğu için bu uygulamaları yapamadık. Bunların yerine veri setimizde bağımlı bağımsız

değişkenler belirledik. Veri setimizi test ve eğitim olarak ikiye ayırdık. Bunların yanında veri setimizin

istatiksel özetini oluşturduk ve bunu yorumladık. Yaptığımız bütün çalışmaların kodları ve açıklamaları

aşağıda yer almaktadır.

İçindekiler

[VERİ SETİNİN İNCELENMESİ 1](#_Toc16273)

[ÖDEV TANITIM FORMU 2](#_Toc21459)

[BEYANNAME 3](#_Toc14300)

[KABUL VE ONAY SAYFASI 4](#_Toc25606)

[ÖZET 1](#_Toc26286)

[ÖN SÖZ 3](#_Toc22899)

[1. Makine Öğrenmesi Nedir? 4](#_Toc8284)

[I. Denetimli Öğrenme: 4](#_Toc32679)

[a. Sınıflandırma: 4](#_Toc20644)

[b. Regresyon: 5](#_Toc12356)

[II. Denetimsiz Öğrenme: 5](#_Toc31762)

[Denetimsiz Öğrenme makine öğrenmesi modellerinden bir diğeridir. Gözetimsiz Öğrenme ve 5](#_Toc29004)

[a. Kümeleme: 5](#_Toc32158)

[b. Boyut Azaltma: 6](#_Toc29395)

[III. Yarı Danışmanlı Öğrenme: 6](#_Toc27173)

[IV. Takviyeli Öğrenme: 6](#_Toc20239)

[2. Uygulama Geliştirme Süreci 7](#_Toc4259)

[3. Veri Setlerinin Karşılaştırılması: 8](#_Toc18467)

[4. Veri Setinin Ayrıntılı İncelenmesi: 9](#_Toc1376)

[5. Tahmin(Prediction) Çeşitleri 10](#_Toc22814)

[6. Veri Seti Hakkında yapılan Literatür Taramaları 13](#_Toc22386)

[7. Python Dili ile Veri Setimizde Uyguladığımız İşlemler: 17](#_Toc23549)

[a. Pandas: 17](#_Toc19254)

[b. Numpy: 18](#_Toc21683)

[c. Matplotlib: 19](#_Toc2481)

[d. Seaborn: 19](#_Toc12679)

[KAYNAKÇA 34](#_Toc29699)

# ÖN SÖZ

Bu projenin geliştirilme aşamasında bana yardımcı olan Beyzanur MURAT ve Emirhan GÜL’e teşekkür ederim.

Hatice AYVA

# Makine Öğrenmesi Nedir?

Makine öğrenmesi yapay zekanın (AI) bir alt kümesidir. Bilgisayarları verilerden öğrenmeyi

öğretmeye ve bunu yapmak için açıkça programlanmak yerine deneyimle geliştirmeye odaklanır.

Makine öğrenmesinde algoritmalar büyük veri kümelerinde desenleri ve korelasyonları bulmak ve

bu analize dayalı en iyi kararları ve tahminleri yapmak için eğitilir. Makine Öğrenmesi evlerimizde,

alışveriş sepetlerinde, eğlence ortamlarında ve sağlık alanında kısacası hayatın hemen her alında

yanımızdadır. Örneğin bir makine öğrenmesi uygulaması olan Siri, kullanıcının sesli komutlarını

anlamak ve yanıtlamak için tasarlanmıştır. Başka bir örnek olarak, Netflix’in öneri motoru,

kullanıcının izleme geçmişine ve beğenilerine dayanarak kişiselleştirilmiş film ve dizi önerileri sunar.

Makine Öğrenmesinin dört farklı çeşidi vardır:

## Denetimli Öğrenme:

Denetimli Öğrenme makine öğrenmesi modellerinden biridir. Gözetimli Öğrenme veya

Danışmalı Öğrenme olarak da geçmektedir. Sadece girdilerin ve çıktıların önemli olduğu olduğu

öğrenme algoritmasıdır. Verilerin etiketlenmesi ile çalışır. Etiketlediği verilerden “girdi” ve “çıktı”

veri çiftleri oluşturur. Denetimli öğrenme modelleri, günümüzde birçok uygulamada kullanılmaktadır.

Örneğin, spam, filtreleme, yüz tanıma ve araba sınıflandırma gibi uygulamalar denetimli öğrenme

modelleri kullanılarak gerçekleştirilir. Örnek vermek gerekirse bir resim tanıma modelinde resimler

etiketlenir ve makine bu etiketleri kullanarak resimlerdeki nesneleri tanımaya çalışır. Eğer nesne

ile etiket eşleşmesi doğru sonuçlanırsa tanıma modeli çalışmış olur. Sınıflandırma ve Regresyon

olarak ikiye ayrılır:

1. **Sınıflandırma:**

Sınıflandırma, denetimli bir makine öğrenme algoritmasıdır. Sınıflandırma makine

öğreniminde öğeleri önceden kategorize edilmiş bir eğitim veri kümesine göre kategorilere ayırma sürecidir. Verileri belirli bir kategoriye veya sınıfa atama işlemidir. Sınıflandırma algoritmaları,

özelliklerin ve etiketlerin verilerde nasıl dağıldığını analiz ederek yeni örneklerin hangi sınıfa ait

olduğunu tahmin eder. Kullanım alanlarına resim sınıflandırma, kimlik dolandırıcılığı tespiti, teşhis

koyma örnekleri verilebilir.

1. Regresyon:

Regresyon, bir bağımlı değişkenin bir veya daha fazla bağımsız değişken tarafından açıklanması

için kullanılan bir istatistiksel modelleme tekniğidir. Regresyon analizi, veriler arasındaki ilişkiyi

anlamak için kullanılır ve gelecekteki olayların tahmin edilmesine yardımcı olabilir. Bu teknik özellikle ekonometri ve finansal analizde yaygın olarak kullanılır. Bu model, bağımlı değişkenin değerini tahmin

etmek için kullanılabilir. Regresyona örnek olarak hava durumu tahmini, popülasyon büyümesi

tahmini, zamana bağlı altın ve dolar tahminlemesi verilebilir.

## Denetimsiz Öğrenme:

## Denetimsiz Öğrenme makine öğrenmesi modellerinden bir diğeridir. Gözetimsiz Öğrenme ve

Denetimsiz Öğrenme olarak da isimlendirilir.Tahmin etmek istediğimiz veri geçmişteki veri setlerinde

yoktur. Ayrıca sınıflara ayrılmamış veriler bulunmaktadır. Örneğin, bir makine öğrenmesi modeli, birçok

farklı müşterinin satın alma alışkanlıklarını analiz edebilir ve benzer satın alma alışkanlıklarına sahip

müşterileri bir araya getirebilir. Bu tür bir model, müşterilerin satın alma alışkanlıklarını daha iyi

Anlamak ve daha iyi hedeflenmiş pazarlama kampanyaları oluşturmak için kullanılabilir.

Kümeleme ve Boyut Azaltma yöntemlerini içinde barındırır:

1. Kümeleme:

Kümeleme, makine öğrenmesinde verileri benzer özelliklere sahip gruplara ayırmak için

kullanılan bir yöntemdir. Bu yöntem verileri benzer özelliklere sahip gruplara ayırarak verilerin daha

iyi anlaşılmasını sağlar. Kümeleme algoritmaları, verilerin benzerliklerine göre gruplandırır ve bu

gruplar arasındaki farklılıkları belirler. Geçmişte bir veri yoktur bu yüzden benzer durumlar üzerinden sınıflandırma yapılır. Bu teknik birçok alanda kullanılır, örnek olarak müşteri segmentasyonu,

sosyal ağ analizi, takviye sistemleri, hedefli pazarlama verilebilir.

1. **Boyut Azaltma:**

Veri ön işlem aşamasında kullanılır. Veri setindeki değişken sayısını azaltarak veri setinin

boyutunu küçültür. Boyut azaltma, veri kümesindeki özellik sayısını azaltmak için kullanılan

bir yöntemdir. Bu yöntemde veri kümesindeki özelliklerin bir kısmı seçilerek yeni bir veri kümesi

oluşturulur. Bu yeni veri kümesindeki özellikler, orijinal veri kümesindeki özelliklerin bir kombinasyonu

olarak ifade edilir. Boyut azaltma yöntemleri arasında PCA (Principal Component Analysis),

LDA (Linear Discriminant Analysis) ve t-SNE (t-Distributed Stochastic Neighbor Embedding)

gibi yöntemler bulunmaktadır.

## Yarı Danışmanlı Öğrenme:

Hibrit Öğrenme olarak da isimlendirilir. Bu aşamada hem denetimli öğrenme hem denetimsiz

öğrenme uygulanır. Denetimsiz öğrenmedeki modelin karmaşıklığı araştırılıp denetimsiz öğrenmenin

devreye girmesi ile sonucun daha başarılı olması amaçlanır. Yani denetimli öğrenme yaparken veri

ön işlem aşamasında denetimsiz öğrenme yapılır. Bu öğrenme kuralı denetimli öğrenmeye yakın bir

metottur.

## Takviyeli Öğrenme:

Sistemde beklenen çıktı ile hesaplanan çıktı karşılaştırılır, doğru ise ödül sistemi yanlış ise

ceza sistemi uygulanır. Kısacası bu öğrenme türünde, bir makine, bir görevi yerine getirmek için

bir dizi adım atar ve her adımda bir ödül veya ceza alır. Makine, bu ödülleri ve cezaları kullanarak

görevi yerine getirmek için en iyi stratejiyi öğrenir. Bu tür öğrenme genellikle oyunlar ve robotik

gibi uygulamalarda kullanılır.

# Uygulama Geliştirme Süreci

**Ham Verinin Okunması:** Kaggle üzerinden veri setimizi ayrıntılı bir şekilde inceledik.

Satır sütün kontrolu yaptık. Veri setimizde yaş, cinsiyet, lokasyon(deri üzerindeki yaranın hangi

bölgede görüldüğü) sütunları bulunmaktadır.

**Veri Ön İşleme:**  Bu aşamada kaggle’dan indirdiğimiz veri setimizde eksik bir veri var mı,

kayıp bir veri var mı diye veri setimizi kontrol ettik. Mod, medyan ve ortalama hesaplamaları

yaptık. Çalışmamızda kullanılacak uygun veri setini oluşturduk. Sonrasında veri ön işleme

aşamasına geçtik bu aşamada veri temizleme yani başarı oranının yüksek olması için kayıp veri analizi

yaptık ama veri setimizde herhangi bir kayıp veri olmadığı için bu veri ön işleme aşamasında bir faaliyet gerçekleştiremedik. Veri temizleme dışında sonucun gerçeğe yakın olması için veri düzenleme yani

yeterli veri var mı diye kontrol de yaptık ve bu kontrol sonucunda veri setimizde yeterli veri olduğu

çıkarımına vardık.

Ayrıca bu aşamada df.fillna(999, inplace=True) kodu ile Null alanlara 999 yazılmasını

sağladık ki ileride yapacağımız işlemler sırasında Null alanlar sorun çıkarmasın.

**Modelin Öğrenmesi:** Modelin öğrenme süreci, derin öğrenme algoritmaları kullanılarak bir

Modelin eğitildiği süreci ifade eder. Bu süreçte, model, veri seti üzerindeki ilişkileri öğrenerek ve

optimize edilerek belirli bir görevi gerçekleştirmek için "öğrenme" yapar. Derin öğrenme algoritmaları,

genellikle büyük veri setleri üzerinde karmaşık yapıları algılamak ve sınıflandırmak için kullanılır.

Bu aşamada yapılan çalışmaları raporumuzun devamında daha ayrıntılı inceleyebilirsiniz.

**Modelin Değerlendirilmesi:** Skin Cancer isimli veri setimiz sınıflandırma modellerinin eğitimi

ve değerlendirilmesi için kullanılabilecek zengin ve çeşitli bir kaynaktır. Biz de modellerin

değerlendirilmesi aşamasında güzel çalışmalar yaptık. Çalışmalarımızı raporun devamında görebilir

ve inceleyebilirsiniz ama burada özel olarak değinmek istediğimiz bir nokta bulunmaktadır.

Sınıflandırma modelinin doğruluk skorunu hesapladık ve sonucu yazdırdığımızda gördük ki sonucumuz

0.1208 gibi düşük bir değer çıktı. Bu sonuç bizi oldukça şaşırttı çünkü doğruluk skoru, modelin test

verileri üzerindeki performansının düşük olduğunu gösterir. Bu, modelin veriler arasındaki ilişkiyi

yeterince iyi yakalayamadığı veya modelin yanlış sınıflandırmalar yaptığı anlamına da gelebilir.

Bu yüzden veri setimizde tekrar ön işlem aşamasında döndük ama herhangi bir yanlış göremedik.

**Performansın Değerlendirilmesi:** Yazmış olduğumuz kod bloklari ile veri seti üzerinde

yapmış olduğumuz işlemleri yorumladık ve işlemler arasında karşılaştırma yaptık.

!!! Skin Lesion Segmentation and Classification Using Deep Learning (Derin Öğrenme

ile Deri Lezyonu Segmentasyonu ve Sınıflandırması), çalışmasının uygulama geliştirme süreci ile

kendi projemizin uygulama geliştirme sürecini karşılaştırıp açıkladık. Uygulama geliştirme süreci

ayrıntıları için bu bölüme bakabilirsiniz. (sayfa 11)

# Veri Setlerinin Karşılaştırılması:

Veri setimizi kaggle platformundan indirip aldık. Başlangıçta Stanford isimli hastanenin veri

setini ([Cilt Hastalıkları Veri Kümesi | Veri Bilimi ve Makine Öğrenimi | Arjantin (kaggle.com)](https://www.kaggle.com/discussions/general/58249)) indirip uygulamaya koyulduk ama özel bir hastanenin veri seti olduğu için erişim kısıtlamaları vardı. Mesela

kullanım koşulları görüntülerin indirilmesini yasaklıyordu. Bunu biraz geç fark etsek de zaman

kaybetmeden yeni bir veri seti araştırması ile veri setimizi bulduk. Veri setimizi incelemeye geçmeden

önce elediğimiz veri setlerini eleme sebeblerimize kısaca değinmek istiyorum.

Skin Diseases ([Skin diseases image dataset | Kaggle](https://www.kaggle.com/datasets/ismailpromus/skin-diseases-image-dataset) ) isimli veri setimiz veri açısından çok

büyük bir veri seti idi. 27.000’ den fazla veri içeriği vardı aa isminden de anlayabileceğiniz gibi bir

img yani resim veri setiydi. İçerisinde yaş, cinsiyet gibi sütunları olan bir veri bulunmamaktaydı bu

yüzden çok iyi bulsak da bu veri setini listemizden eledik.

İnceleme yaptığımız diğer bir veri seti olan Lumpy Skin Disease isimli veri setini

([Lumpy Skin Disease Dataset | Kaggle](https://www.kaggle.com/datasets/saurabhshahane/lumpy-skin-disease-dataset)) tercih etmememizin sebebi ise bizim değerlendirmeye

alacağımız yaş ve cinsiyet verilerinin bu veri setinde bulunmamasıydı. Veri setinde veriler ülkelere

göre kategorize edilmişti. Ayrıca sayıca az bir veri bulunduruyordu. Bildiğiniz gibi veri seti ne kadar

Büyük ise algoritma sonucumuz gerçeğe o kadar yakın olur. Bu sebeplerden dolayı bu veri setini de

kullanmaktan da vazgeçtik.

Kullanmaya karar verdiğimiz veri seti olan Skin Cancer isimli veri setini

([Skin Cancer MNIST: HAM10000 | Kaggle](https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000)) açıklamak gerekirse veri setimizde yaş, cinsiyet

gibi veri sütunlarının bulunması dışında bu veri setimizde ciltte görülen rahatsızlığın hangi

Bölgede olduğuna dair bilgi veren bir location isimli veri sütunu bulunmaktaydı. Bu sütunun

çalışmalarımıza katkısının daha çok olacağını düşündük. Ayrıca veri setimizde 10.014 verinin

bulunması çalışma sonuçlarımızın gerçeğe yakınlığını arttıracağı için bu veri setini kullanmaya

karar verdik. Kullanmış olduğumuz HAM10000 veri seti genellikle 2018-2020 yılları arasında deri

kanseri sınıflandırması üzerine yapılan çalışmalarda kullanılan bir veri seti olarak popülerlik kazanmıştır.

# Veri Setinin Ayrıntılı İncelenmesi:

Çalışmamızda Skin Cancer ([Skin Cancer MNIST: HAM10000 | Kaggle](https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000?select=hmnist_8_8_RGB.csv)) isimli veri setini

kullandık. Veri Setimiz 10.015 farklı deri lezyonu görüntüsünden oluşur ve toplamda 7 farklı sınıfa

ayrılmıştır. Profesyonel dermataloglar tarafından deri lezyonlarının klinik tanısıyla etiketlenmiş

görüntüler içermektedir. Her bir görüntü, dermatoskop kullanılarak çekilmiş ve 600x450 piksel

çözünürlüğe sahiptir. Sınıflandırma modellerinin eğitimi ve değerlendirilmesi için kullanılabilecek

zengin ve çeşitli bir kaynaktır. Kaggle platformunda, bu veri setini kullanarak deri kanseri

sınıflandırma modelleri geliştirmek, doğrulamak veya karşılaştırmak için birçok araştırma ve

yarışma düzenlenmiştir.

Veri setimiz kimlik numarası, klinik tanısı, yaş, cinsiyet, lokalizasyon sütunlarından

oluşmaktadır. Yaş sütununda minumum değerimiz 0 maksimum değerimiz 85’dir. Yani veri

setimizdeki en yaşlı kişi 85 yaşındadır yorumunu yapabiliriz. Cinsiyet sütunumuz kız, erkek ve

bilinmeyen olarak üç seçeneğe ayrılmaktadır. Lokalizasyon sütunu ciltte görülen sivilce, akne

veya deri üstü değişiminin hangi bölgede görüldüğü bilgisini içeren bir sütundur. Lokalizasyon

sütunu kulak, yüz, ayak, el, genital bölge... olmak üzere 15 bölümden oluşmaktadır.

# Tahmin(Prediction) Çeşitleri

Tahmin (Prediction), verilerin analizi ve modeleme süreci sonucunda elde edilen bir modelin kullanılmasıyla gelecekteki olayların veya değerlerin tahmin edilmesidir. Sınıflandırma Tahmini,

Regresyon Tahmini, Zaman Serisi Tahmini, Gruplama Tahmini, Durum Tahmini gibi birçok tahmin

çeşiti bulunmaktadır. Biz ise bu yazımızda Regresyon Tahmin çeşitlerini daha ayrıntılı ele alacağız.

1. Doğrusal Regresyon(Lineer Regresyon):

Bazı kaynaklarda Doğrusal Korelasyon veya Basit Doğrusal Regresyon olarak da

isimlendirilmektedir. Doğrusal regresyon, bağımlı bir değişkenin (hedef değişken) bir veya daha

fazla bağımsız değişkenle ilişkisini modellemek için kullanılan istatistiksel bir yöntemdir. Bu

yöntemde, bağımlı değişkenin değeri, bağımsız değişkenlerin değerleri ve bir dizi regresyon

katsayısı aracılığıyla tahmin edilmeye çalışılır. en basit haliyle, bir doğru denklemi kullanarak

değişkenler arasındaki ilişkiyi ifade eder. Doğrusal regresyon, veriler arasındaki ilişkiyi anlamak,

tahmin yapmak ve değişkenler arasındaki etkileşimi analiz etmek için yaygın olarak kullanılır.

Genellikle en küçük kareler yöntemi kullanılarak regresyon katsayıları tahmin edilir. Doğrusal

regresyon, istatistiksel analiz, ekonomi, finans, sosyal bilimler, pazarlama ve mühendislik gibi birçok

alanda yaygın olarak kullanılan temel bir yöntemdir.

Doğrusal regresyonun matematiksel formülü şu şekildedir:

**Y = b0 + b1X1 + b2X2 + ... + bnXn**

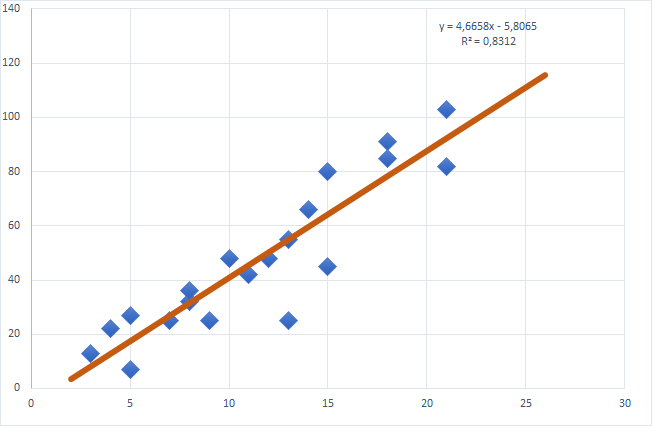
*(y = bağımlı değişken)*

*(x = bağımsız değişken)*

*(b0 = sabit)*

*(b1 = eğim)*

*Doğrusal Regresyon Örneği:*

**

1. Çoklu Lineer Regreson(Multiple Lineer Regresyon):

Çoklu Lineer Regresyon bağımlı bir değişkenin birden fazla bağımsız değişkenle

ilişkisini modellemek için kullanılan istatistiksel bir yöntemdir. Bu yöntemde, bağımlı değişkenin

değeri, birden fazla bağımsız değişkenin değeri ve bir dizi regresyon katsayısı aracılığıyla tahmin

edilmeye çalışılır. Çoklu lineer regresyon, doğrusal regresyonun genişletilmiş bir versiyonudur.

Tek değişken yerine birden fazla bağımsız değişkenin kullanıldığı bir regresyon modelidir. Bu

model, bağımlı değişkenin tahmin edilen değerini en iyi şekilde temsil etmeye çalışırken, birden

fazla bağımsız değişken arasındaki ilişkiyi de yakalamayı hedefler.

Çoklu lineer regresyon modelleri, regresyon katsayılarını tahmin etmek ve modelin

doğruluğunu değerlendirmek için istatistiksel yöntemler kullanır. Örneğin, en küçük kareler yöntemi

kullanarak regresyon katsayıları tahmin edilebilir ve R-kare (determinasyon katsayısı), standart hata

gibi istatistiksel metrikler kullanılarak modelin performansı değerlendirilebilir.

Çoklu lineer regresyonun matematiksel formülü şu şekildedir:

Y = b0 + b1X1 + b2X2 + ... + bnXn

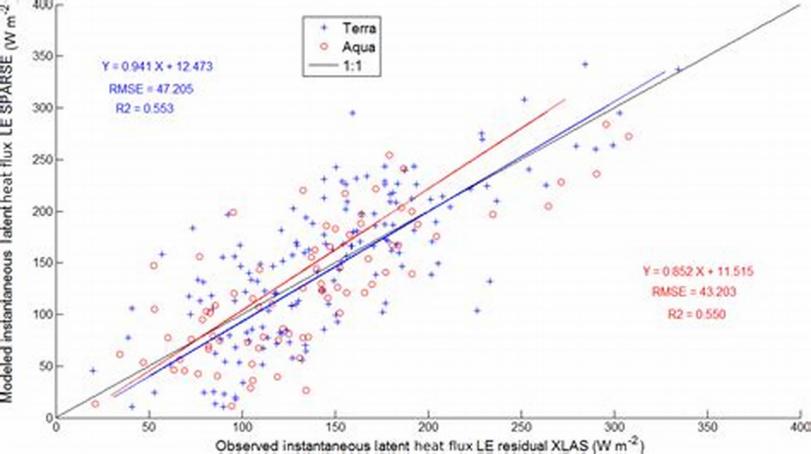
*(y = bağımlı değişken)*

*(X1, X2, ..., Xn = bağımsız değişkenlerin değerleri)*

*(b0 = sabit)*

*(b1, b2, ..., bn = regresyon katsayıları)*

*Çoklu Lineer Regresyon Örneği:*



1. Polinom Regresyon (Polynominal Regression):

Polinom regresyon (Polynomial Regression), bağımlı değişkenin bir veya daha fazla

bağımsız değişkene polinomik bir denklemle modellemek için kullanılan bir regresyon yöntemidir.

Bu yöntem, doğrusal olmayan ilişkileri yakalamak için kullanılır. Polinom regresyon, doğrusal

regresyonun genişletilmiş bir versiyonudur. Doğrusal regresyonda, bağımlı değişken ile bağımsız

değişken arasındaki ilişkiyi doğrusal bir doğruyla temsil etmeye çalışırken, polinom regresyonu,

ilişkiyi polinom fonksiyonlarıyla ifade etmek için kullanılır. Bu, veri setindeki eğilimleri daha esnek

bir şekilde yakalamayı sağlar. Polinom regresyonu, çeşitli alanlarda kullanılır, özellikle matematiksel

modelleme, ekonomi, fizik, biyoloji ve mühendislik gibi bilimsel disiplinlerde yaygın olarak kullanılır.

Polinom regresyonu kullanılarak elde edilen modeller, verilerdeki eğilimleri ve ilişkileri anlamak,

gelecekteki tahminler yapmak veya yeni verileri tahmin etmek için kullanılabilir.

Polinom regresyon modelinin genel formülü şu şekildedir:

Y = β0 + β1X + β2X^2 + ... + βnX^n + ε

*(Y, bağımlı değişkeni temsil eder.)*

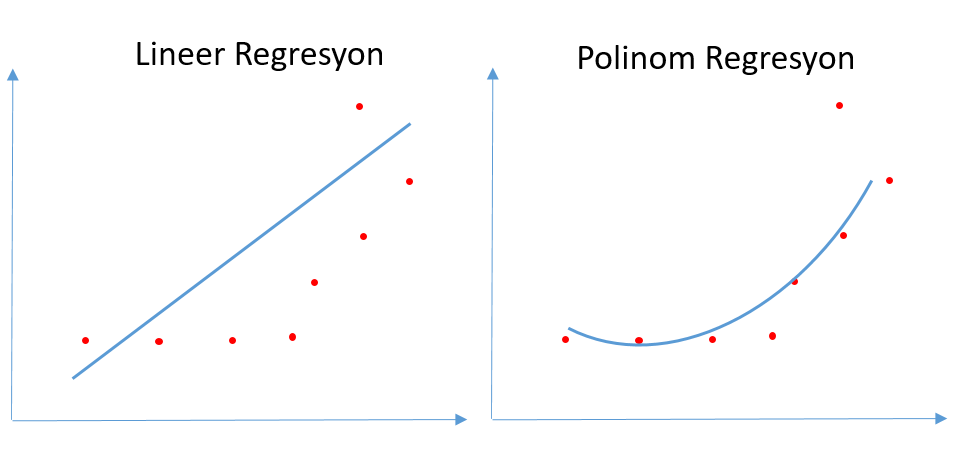
*(X, bağımsız değişkeni temsil eder.)*

*(β0, β1, β2,..., βn, polinom katsayılarını temsil eder.)*

*(X^2, X'in karesini temsil eder.)*

*(ε, hata terimidir.)*

*Polinom Regresyon Örneği:*

**

# Kullanmış Olduğumuz Veri Seti veya Benzer Veri Setleri Kullanarak Yapılmış Çalışmaları İnceleme, Literatür Taraması

Araştırmalarımız sonucunda bu veri seti üzerinden yapılmış dört adet doğrudan veri

seti ile ilgili çalışmaları veya veri seti ile aynı konuda gerçekleştirilmiş çalışmalar inceledik.

Şimdi bu çalışmaları inceleyelim:

**Skin Cancer Classification Using Convolutional Neural Networks (Derin Sinir**

**Ağları Kullanarak Deri Kanseri Sınıflandırması):** Bu çalışma2018 ve 2020 yılları arasında

yapılmıştır. Bu çalışmada derin sinir ağları (CNN) kullanarak deri kanseri sınıflandırması üzerine

yapılan çalışmalar yer almaktadır. Deri lezyonlarının görüntülerini analiz etmek ve farklı

sınıfları doğru sınıflandırmak için CNN algoritmalarının kullanılmasını içerir. CNN algoritmasının

ne olduğundan kısaca bahsedelim. CNN genellikle görüntü işlemede kullanılan ve girdi olarak

görselleri alan bir derin öğrenme algoritmasıdır. Özellikle görüntü işleme ve sınıflandırma

problemlerinde başarılı sonuçlar veren derin öğrenme modelleridir.

Çalışma sonucunda yayımlanan makaleye göre CNN'ler son teknoloji deri lezyonu

sınıflandırıcıları olarak yüksek performans gösterirler. Ne yazık ki, farklı sınıflandırma

yöntemlerini karşılaştırmak zordur çünkü bazı yaklaşımlar eğitim ve test için halka açık olmayan

veri kümelerini kullanır ve böylece tekrarlanabilirliği zorlaştırır.

**Bu çalışmada uygulanan CNN çalışması bizim veri setimizde de uygulanabilir mi?**

CNN'ler, büyük boyutlu görüntülerle çalışmak için etkilidir. Bizim veri setimizde 6806 adet görüntü

vardır. Bu yüzden veri setimizde CNN'leri kullanmak yerine daha basit modeller veya algoritmalar

tercih edilebilir. Örneğin daha önce yukarıda açıklanan sebeplerden kaynaklı vazgeçtiğimiz Skin

Diseases ([Skin diseases image dataset | Kaggle](https://www.kaggle.com/datasets/ismailpromus/skin-diseases-image-dataset)) isimli veri setinde 27.000’den fazla img(görsel)

vardır. Bu veri seti farklı açılardan çekilmiş veya farklı ışık açılarıyla çekilmiş görselleri içinde

barındırmaktadır. CNN de farklı açılardan çekilmiş veya farklı ışık koşullarında görüntülenmiş

verilere karşı dayanıklıdır. Bu yüzden bizim veri setimizin yerine Skin Diseases veri seti bu makalede

açıklanan CNN çalışmasını yapmaya daha uygundur.

**Skin Lesion Segmentation and Classification Using Deep Learning (Derin Öğrenme**

**ile Deri Lezyonu Segmentasyonu ve Sınıflandırması):** Bu çalışma deri lezyonlarının

segmentasyonunu (deri lezyonlarının otomatik olarak belirlenmesini) ve sınıflandırmasını

hedeflemektedir. Bu çalışmanın uygulama geliştirme sürecini inceleyelim ve bizim çalışmamızdaki

uygulama geliştirme süreciyle karşılaştıralım:

**Veri Hazırlığı:** İlk adımda, deri kanseri veri seti ( HAM10000) kullanılarak çalışma için

uygun bir veri seti oluşturulur. Veri seti genellikle etiketli görüntülerden oluşur yani her görüntüye

doğru sınıf etiketi atanmıştır.

Peki biz projemizin veri hazırlığı aşamasında ne yaptık? Biz de bu aşamada tıpkı bu

çalışmadaki gibi ham veriyi okuma eylemini gerçekleştirdik. Çalışmamızda kullanılacak uygun veri

setini oluşturduk. Sonrasında veri ön işleme aşamasına geçtik bu aşamada veri temizleme yani başarı

oranının yüksek olması için kayıp veri analizi yaptık ama veri setimizde herhangi bir kayıp veri

olmadığı için bu veri ön işleme aşamasında bir faaliyet gerçekleştiremedik. Veri temizleme dışında

sonucun gerçeğe yakın olması için veri düzenleme yani yeterli veri var mı diye kontrol de yaptık ve

bu kontrol sonucunda veri setimizde yeterli veri olduğu çıkarımına vardık. Bu çalışmada da kayıp

veri olmadığı için ve veriler yeterli olduğu için veri ön işleme aşaması yer almamaktadır. Bu çıkarımlar sonucunda veri setinde herhangi bir ön işlem aşaması yapılmamıştır şeklinde bir yorum yapabiliriz.

**Görüntü Segmentasyonu:** Deri lezyonlarının otomatik olarak belirlenmesi için görüntü

segmentasyonu yöntemleri kullanılır. Bu adımda, derin öğrenme modelleri (CNN gibi) kullanılarak görüntülerdeki lezyon bölgeleri belirlenir. Bu, lezyonların sınırlarını çizmek veya piksel düzeyinde

lezyon bölgelerini belirlemek gibi farklı yöntemlerle gerçekleştirilebilir.

Biz projemizin bu aşamasında yani model geliştirme aşamasında ne yaptık? Biz de bu

aşamada modelimizi geliştirerek derin öğrenme modelleri kullandık. Aşağıda projemizin ayrıntıları

daha kapsamlı anlatılmıştır.

**Özellik Çıkarma:** Lezyon bölgeleri belirlendikten sonra, bu bölgelerden anlamlı özelliklerin

çıkarılması gerekmektedir. Derin öğrenme modelleri, genellikle özellik çıkarma için kullanılan önceden

eğitilmiş ağlar üzerine transfer öğrenme yöntemleriyle bu görevi gerçekleştirebilir. Bu sayede lezyon

bölgeleri için temsilci özellikler elde edilir.

**Sınıflandırma:** Elde edilen özellikler kullanılarak, deri lezyonlarının sınıflandırması yapılır.

Bu aşamada, derin öğrenme sınıflandırma modelleri (örneğin, CNN'ler) kullanılarak lezyonların

türlerine (iyi huylu veya kötü huylu) göre sınıflandırma gerçekleştirilir. Bu aşamada, veri setinde

bulunan özelliklerin kullanılarak farklı sınıflara ait veri noktalarının sınıflandırılması gerçekleştirilir. Sınıflandırma aşaması genellikle makine öğrenimi veya derin öğrenme algoritmaları kullanılarak

gerçekleştirilir.

**Transfer Learning for Skin Cancer Classification" (Deri Kanseri Sınıflandırması için**

**Aktarım Öğrenmesi):**

Bu çalışmada, HAM10000 veri seti üzerinde transfer öğrenme yöntemleri kullanılarak deri

kanseri sınıflandırması gerçekleştirilmiştir. Önceden eğitilmiş bir derin sinir ağı modeli kullanılarak,

modelin sınıflandırma yetenekleri HAM10000 veri setine uyarlanmış ve değerlendirilmiştir.

Genel olarak transfer öğrenme, bir görevden (kaynak görev) öğrenilen bilgiyi başka bir

Görevde (hedef görev) kullanma fikrine dayanır. Deri kanseri sınıflandırması için aktarım öğrenmesi

bir önceden eğitilmiş bir derin öğrenme modelinin deri kanseri sınıflandırması görevinde kullanılması

anlamına gelir. Genel olarak, transfer öğrenimi, sınırlı miktarda etiketli veri olsa bile, cilt kanseri

sınıflandırma görevlerinin performansını ve verimliliğini artırmak için önceden eğitilmiş modellerden

öğrenilen bilgi ve temsillerden yararlanmaya izin verir.

Peki bizim projemizde bu çalışmada uygulanan Transfer Lerarning yapılabilir mi? Öncelikli

olarak Transfer Learning çalışmasını daha basit ve anlaşılır olarak ifade etmek gerekirse bir problemi

çözerken elde edilen bilgiyi saklamak ve daha sonra farlı bir probleme saklanan bilgiyi uygulamaktır.

Mesela basit bir örnekle otomobili öğrenirkenki bilgiyi saklayıp kamyonu öğrenirken kullanmak gibi.

Genellikle çok görevli öğrenme (multi-tasking-learning) ve kavram sapması gibi problemlerle ilgilidir.

Transfer öğrenimi için etkili sonuçlar elde etmek için kaynak bir veri setine ihtiyaç vardır. Bu veri seti,

genellikle büyük ve çeşitli örnekler içermelidir. Yani Transfer Leranin bizim veri setimizde

uygulanabilecek bir öğrenme çeşitidir. Lakin biz bu ver seti üzerindeki proje çalışmamızda Transfer

Learnig uygulaması yapmadık.

**Feature Extraction and Selection for Skin Cancer Classification (Deri Kanseri**

**Sınıflandırması için Özellik Çıkarma ve Seçimi):**

Bu çalışma, HAM10000 veri setindeki lezyonların özelliklerini çıkarma ve seçme üzerine

odaklanmıştır. Farklı özellik çıkarma yöntemleri (örneğin, HOG, LBP gibi) kullanılarak lezyonların

karakteristik özellikleri çıkarılmış ve ardından bu özellikler kullanılarak deri kanseri sınıflandırması gerçekleştirilmiştir. Bu çalışma, deri kanseri sınıflandırması için özellik çıkarma ve seçimi konusunda

bilgi sağlamayı ve deri kanseri teşhisinde daha etkili ve doğru sonuçlar elde etmeyi amaçlamaktadır.

Bu çalışmanın sonuçları, deri kanseri teşhisinde kullanılan makine öğrenmesi veya derin öğrenme

tabanlı sınıflandırma sistemlerinin geliştirilmesine katkıda bulunabilir. Deri kanseri sınıflandırması

için özellik çıkarma, görüntülerdeki renk, kenarlar, şekiller vb. gibi karakteristikleri yakalamayı

amaçlar. Elde edilen özellik vektörleri genellikle yüksek boyutlu olabilir ve gereksiz veya gürültülü

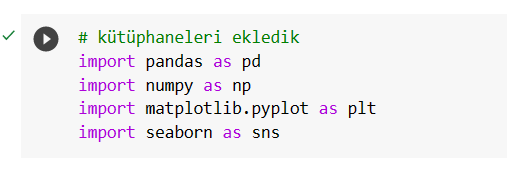
bilgiler içerebilir. Özellik seçimi, bu özellik vektörlerinden en önemli ve en ayrımcı olanları seçmeyi

amaçlar. Bu adım, sınıflandırma performansını artırmak ve hesaplama maliyetini azaltmak için

önemlidir.

# Python Dili ile Veri Setimizde Uyguladığımız İşlemler:

Python dili kullanarak yaptığımız kod çalışmalarının tamamını colob üzerinden yaptık.



Yukarıda görmüş olduğunuz kod bloğunda kütüphane ekleme işlemini gerçekleştirdik.

Kütüphanelerimizi daha ayrıntılı ele almak gerekirse:

1. Pandas:

Python programlama dili için güçlü ve esnek veri analizi ve manipülasyonu için kullanılan

popüler bir kütüphanedir. Pandas, tablo şeklindeki verilerin işlenmesi ve analiz edilmesi için bir

dizi veri yapıları ve fonksiyonlar sağlar. Pandas'ın temel veri yapılarından biri DataFrame'dir,

bu yapı verileri tablo şeklinde temsil etmek için kullanılır.

* Pandas, verilerin seçilmesi, filtrelenmesi, sıralanması, gruplanması ve dönüştürülmesi

gibi çeşitli manipülasyon işlemlerini kolaylıkla yapmanızı sağlar.

* Pandas, farklı veri kaynaklarından (CSV dosyaları, Excel dosyaları, veritabanları, web

kaynakları vb.) veri okuma ve bu verileri çeşitli formatlarda kaydetme yetenekleri sunar.

* Pandas, veri üzerinde matematiksel işlemler, istatistiksel hesaplamalar ve zaman serisi

analizleri gibi çeşitli hesaplama işlemlerini gerçekleştirmek için işlevler sunar.

* Pandas, eksik veya bozuk verilerin ele alınması ve temizlenmesi için işlevler sağlar. Boş

hücrelerin doldurulması, veri noktalarının atılması veya eksik veri noktalarının tahmin edilmesi

gibi işlemleri kolaylıkla yapabilirsiniz.

* Pandas, verileri grafikler, çizimler ve tablolar şeklinde görselleştirmek için entegre edilmiş

araçlara sahiptir. Bu sayede veri analizini daha anlaşılır ve etkileyici hale getirebilirsiniz.

1. Numpy:

Python programlama dilinde kullanılan bir kütüphanedir. Bilimsel hesaplamalar için

güçlü bir çok boyutlu dizi nesnesi ve matematiksel işlemler için fonksiyonlar sağlar. NumPy,

Python'daki veri analizi, veri manipülasyonu ve bilimsel hesaplama gibi birçok alanda temel

bir araçtır.

* NumPy, homojen veri dizilerini temsil etmek için çok boyutlu dizi (ndarray) nesnesini

kullanır. Bu diziler, matrisler veya vektörler gibi verileri hızlı ve etkili bir şekilde depolamanızı

ve işlemenizi sağlar.

* NumPy, matematiksel işlemler için optimize edilmiş fonksiyonlara sahiptir. Bu fonksiyonlar

sayesinde, NumPy ile vektörel ve matris işlemlerini hızlı bir şekilde gerçekleştirebilirsiniz. Örneğin

toplama, çıkarma, çarpma, transpoz alma, matris çarpımı ve lineer cebir işlemleri gibi işlemleri

kolayca yapabilirsiniz.

* NumPy, farklı şekillerdeki diziler arasında otomatik olarak yayınlama yapabilme yeteneğine

sahiptir. Bu, farklı şekil ve boyutlara sahip dizileri birbirleriyle uyumlu hale getirerek matematiksel

işlemleri gerçekleştirmenizi sağlar.

* NumPy, veri dosyalarıyla (CSV, TXT, HDF5 vb.) kolayca entegre olabilen fonksiyonlara

sahiptir. Bu sayede, NumPy ile veri okuma, yazma ve manipülasyonu yapabilirsiniz.

* Bilimsel hesaplama ve veri analizi alanında diğer birçok Python kütüphanesiyle uyumlu çalışır.

Örneğin, Pandas, Matplotlib, SciPy ve Scikit-learn gibi kütüphaneler, NumPy üzerine inşa edilmiştir

ve NumPy ile uyumlu veri yapılarını ve işlevleri kullanır.

1. Matplotlib:

Python programlama dili için bir veri görselleştirme kütüphanesidir. Grafikler, çizimler,

tablolar ve diğer görsel öğeleri oluşturmak için kullanılır. Matplotlib, basit çizimlerden karmaşık görselizasyonlara kadar geniş bir yelpazede görsel sunumlar oluşturmanıza olanak tanır.

* Matplotlib, çizgi grafikleri, saçılım grafikleri, çubuk grafikleri, alan grafikleri, histogramlar,

kutu grafikleri, yüzey grafikleri, 3D grafikler ve daha fazlası gibi çeşitli çizim türlerini destekler.

Bu çizim türleri, veri analizinde sıklıkla kullanılan farklı veri yapılarını ve dağılımlarını

görselleştirmek için kullanılabilir.

* Matplotlib, grafiklerinizi kişiselleştirmenize olanak tanır. Renkler, çizgi stilleri, etiketler,

eksenlerin ölçekleri ve biçimlendirme gibi birçok özellik üzerinde tam kontrol sağlar. Bu

grafiklerinizi kendi gereksinimlerinize ve tercihlerinize göre özelleştirebilmenizi sağlar.

* Matplotlib, birden çok grafik ve eksenin aynı görüntü üzerinde düzenlenmesini destekler.

Çoklu grafikler ve eksenler, farklı veri setlerini veya farklı görüntüleme stillerini aynı anda

karşılaştırmak veya göstermek için kullanılabilir.

* Matplotlib, grafiklerinizde eksen etiketleri, başlık, açıklama ve diğer metinleri eklemek için

işlevlere sahiptir. Bu, grafiklerinizi daha açıklayıcı hale getirir ve okuyucuların anlamalarını

kolaylaştırır.

* Matplotlib, IPython ve Jupyter Notebook gibi ortamlarda interaktif grafikler oluşturmanıza

olanak sağlar. Bu, grafiklerinizi etkileşimli olarak keşfetme ve analiz etme imkanı sunar.

1. Seaborn:

Python programlama dili için bir veri görselleştirme kütüphanesidir. Matplotlib'e dayanan

ve onun üzerine inşa edilen Seaborn, daha yüksek seviye bir arayüz sağlayarak kullanımı daha kolay

ve çekici görselleştirmeler yapmanızı sağlar. Seaborn, özellikle istatistiksel veri analizi ve keşifsel

veri analizi için kullanılan grafiklerin oluşturulmasını kolaylaştırır.

* Seaborn, grafiklerinizi daha çekici ve profesyonel bir görünüm kazandırmak için önceden

tanımlanmış renk paletleri, stilleme seçenekleri ve tema ayarları sağlar. Bu, görsellerinizin görsel

açıdan daha etkileyici olmasını sağlar.

* Seaborn, istatistiksel veri analizi için özel olarak tasarlanmış grafiklerin oluşturulmasını

destekler. Örneğin, çekirdek yoğunluk tahminleri, regresyon çizgileri, dağılım grafikleri, kutu

grafikleri, violon grafikleri, nokta grafikleri ve ısı haritaları gibi istatistiksel görselleştirmeleri

kolayca yapabilirsiniz.

* Seaborn, kategorik verileri etkili bir şekilde görselleştirmenize olanak tanır. Örneğin, sütun

grafikleri, çizgi grafikleri, barplot'lar, stripplot'lar, swarmplot'lar gibi çeşitli grafiklerle kategorik

verilerinizi görselleştirebilirsiniz.

* Seaborn, veri setindeki değişkenler arasındaki ilişkileri ve dağılımları analiz etmek için

kullanışlı araçlar sunar. Çiftler plotları, nokta plotları, korelasyon matrisleri gibi grafiklerle veri

setinizdeki bağımlılıkları görselleştirebilirsiniz.

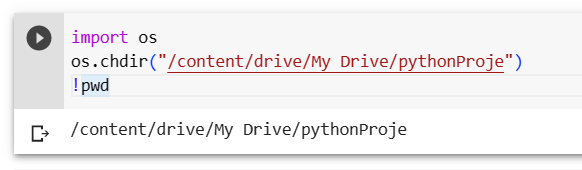
* Seaborn, birden çok grafik veya alt grafik oluşturmanızı sağlar. Böylece farklı veri setlerini

veya farklı görüntüleme biçimlerini aynı anda karşılaştırabilirsiniz.



Yukarıdaki kod ile colab’a drive’ı bağladık. Bu işlemi yapma sebebimiz veri setimizin drive

alanında olması. Yani veri setimizi drive’dan çekebilmek için yaptık.



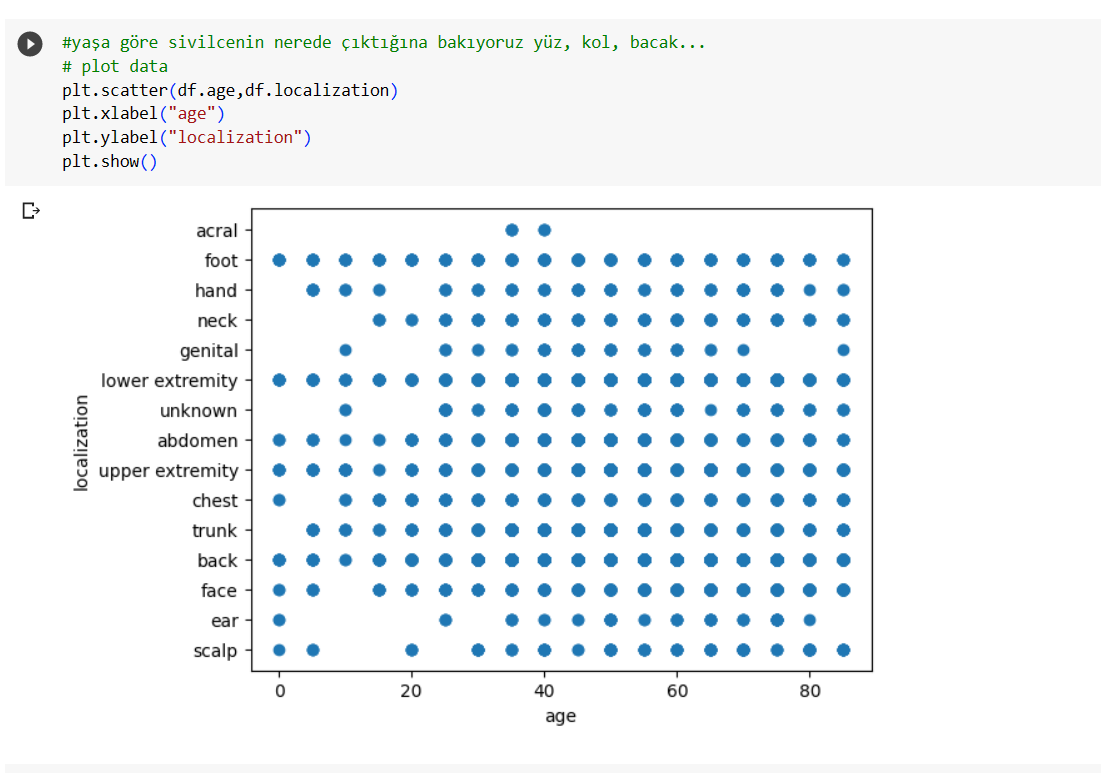
Yukarıdaki kod ile drive alanında bulunan verisetimizin hangi dosya içinde olduğunu colob’a

tarif ettik. Veri setimiz pythonProje isimli dosyanın içinde yer almaktadır.



Yukarıdaki kod ile ismi “HAM10000\_metadata.csv” olan veri setimizi çağırdık ve veri

setimizi görüntüledik.



Yukarıdaki kodda x etiketini yaş olarak, y etiketini ise locasyon olarak belirledik. Çıktı

sonucunda da yaşa göre sivilce, akne ya da deri üzerinde oluşan değişimin hangi bölgede olduğuna

dair grafiğimiz geldi.

Çıktı sonucundaki grafiği yorumlarsak 20 yaşındaki bir bireyde sivilce, akne ya da deri

üzerinde oluşan değişim kulak, bilinmeyen bölge, genital bölge, el, akral(tıbbi bir terim)

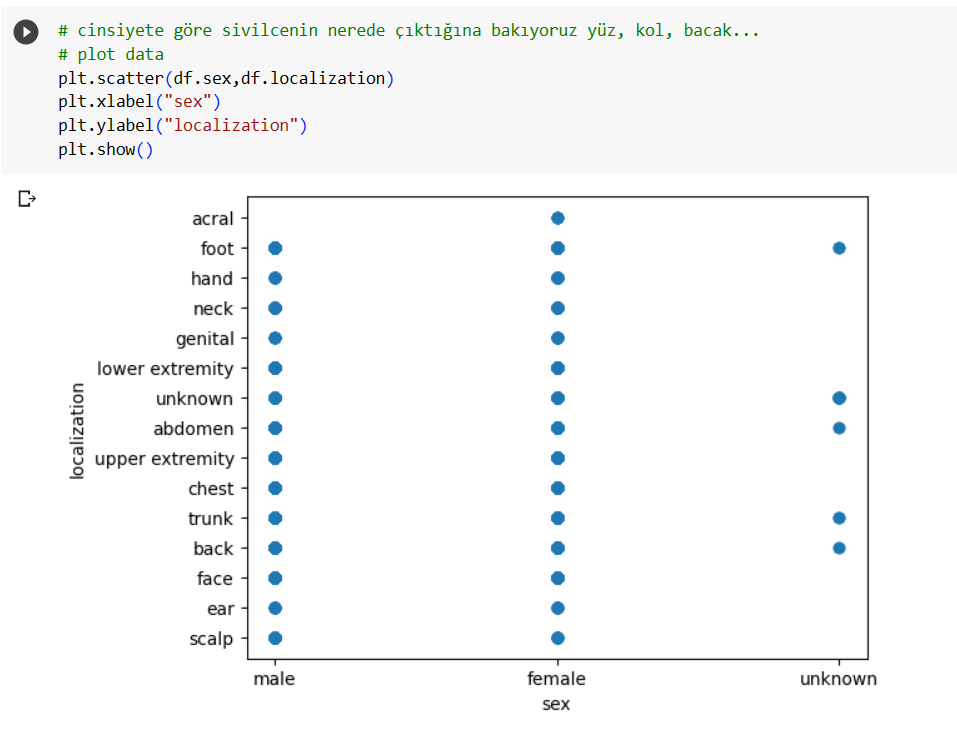
bölgeleri dışında diğer bütün bölgelerde görüldüğü yorumu yapılabilir. Aynı şekilde grafiği

gözlemlemeye devam ettiğimizde 40 yaşındaki bireylerin vücutlarının bütün bölgelerinde sivilce,

akne ya da deri üzerinde oluşan değişim görülmektedir. 60 yaş aralığındaki bir bireye bakacak

olursak da acral dışında vücudunun bütün bölümlerinde sivilce, akne ya da deri üzerinde oluşan

değişim görülmektedir. Yaş aralıkları grafiğe göre benzer şekillerde daha ayrıntılı yorumlanabilir.



Yukarıdaki kod bloğunda önceki kod bloğundan farklı olarak x etiketimizi cinsiyet(sex),

y etiketimizi ise bölge(localization) olarak belirledik. Çıktı sonucunda ise cinsiyete göre sivilce,

akne ya da deri üzerinde oluşan değişimin hangi bölgede görüldüğü grafikte gösterilmektedir.

Grafiğimizi yorumlayacak olursak erkeklerde acral(tıbbi bir terim) bölgesi dışında bütün

bölgelerde sivilce, akne ya da deri üzerinde oluşan değişim görülmektedir. Kadınlarda ise bütün

bölgelerde sivilce, akne ya da deri üzerinde oluşan değişim görülmektedir. Buradan çıkan sonuca

göre kadınlarda acral bölgesinde görülme var iken erkeklerde bu bölgede görülmemektedir şeklinde

bir karşılaştırma yapılabilir.

Sonraki kod bloğuna geçmeden önce yukarıdaki kodlarda kullanılan Plot Data’ya biraz

bakalım:

1. Plot Data:

"Plot data" terimi, verilerin grafiksel bir şekilde gösterilmesi anlamına gelir. Verilerin

doğru anlaşılması, analizi ve karşılaştırılması için grafiklerin kullanılması yaygın bir yöntemdir.

Veri görselleştirmesi, karmaşık veri setlerinin daha anlaşılır ve erişilebilir hale getirilmesine

yardımcı olur.Bir veri setinin çeşitli özelliklerini ve ilişkilerini göstermek için farklı türde grafikler

kullanılabilir:

* Çizgi Grafiği: Verilerin zamana veya başka bir bağımsız değişkene bağlı olarak nasıl değiştiğini

gösterir. Genellikle sürekli bir değişkenin eğilimini analiz etmek için kullanılır.

* Sütun Grafiği: Kategorik verileri göstermek için kullanılır. Her kategori bir sütunla temsil edilir

ve sütunların yükseklikleri, ilgili kategoriye ait değerleri temsil eder.

* Pasta Grafiği: Verilerin yüzdelik dağılımını göstermek için kullanılır. Bir daire şeklinde temsil

edilir ve farklı kategorilerin oransal paylarını gösteren dilimlere ayrılır.

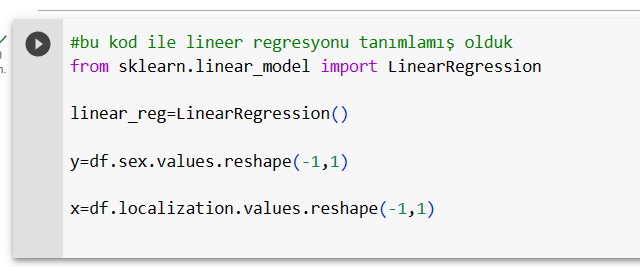
* Dağılım Grafiği: Veri setinin dağılımını gösterir. Genellikle noktaların bir eksen boyunca nasıl

dağıldığını veya bir histogram şeklinde verinin frekansını gösterir.

* Kutu Grafiği: Veri setinin istatistiksel özetini gösterir. Genellikle minimum, maksimum, medyan,

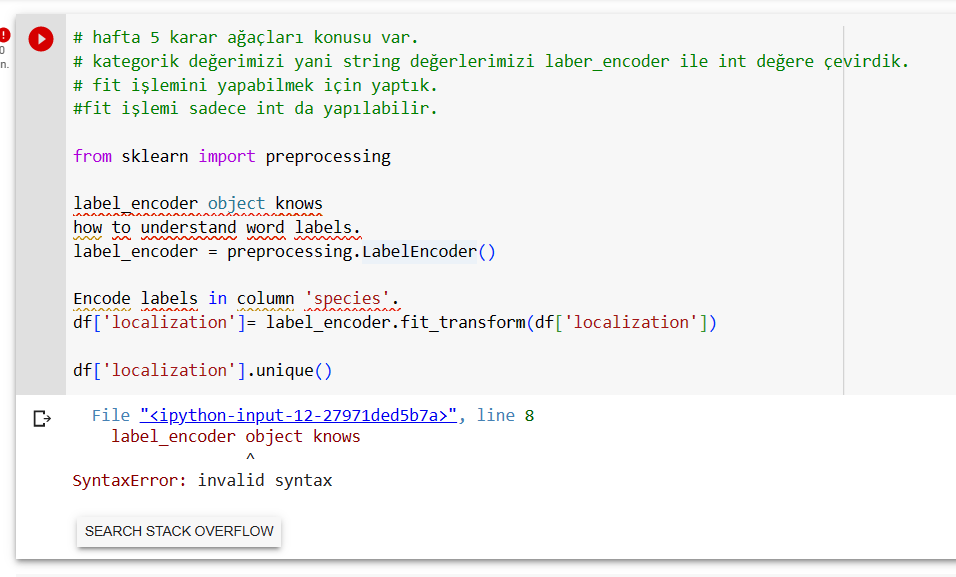
çeyreklikler ve aykırı değerler gibi istatistiksel ölçümleri gösteren bir kutu ve bıyıklar şeklinde temsil

edilir.



Yukarıdaki kod Python'da scikit-learn kütüphanesinin LinearRegressionsınıfını kullanarak

doğrusal regresyon modeli oluşturmayı göstermektedir.

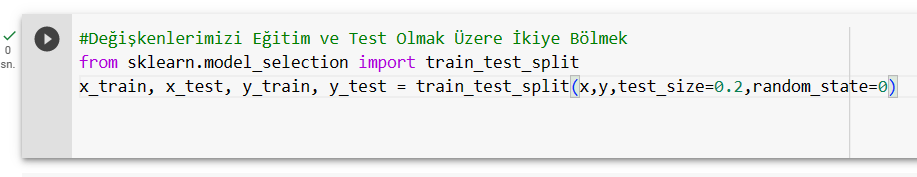


Yukarıdaki kod bloğunda fit ile yapılan öğrenme işlemini yapmayı denedik ama fit işlemi

Sadece int yani sayısal değerdeki ifadelerde kullanıldığı için kodumuz hata verdi. Çünkü bizim

veri setimizde sayısal türden değer sadece yaş sütununda bulunmaktadır. Diğer sütunlarda bulunan

verilerin türleri sayısal değildir.

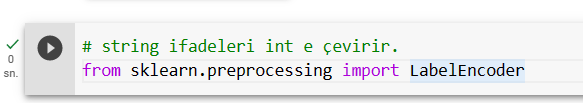


Yukarıdaki kod ile değişkenlerimizi eğitim ve test olmak üzere ikiye ayırdık.

Veri setlerini yaklaşık olarak %75 , %80 eğitim verisi ve yaklaşık %25, %20 test verisi

olarak ayırırız. Eğitim verisi ile modelin öğrenmesi sağlarız. Test verisi ile modelin değerlendirilmesi

sağlarız.



Yukarıdaki kodu kategorik veya metinsel verileri sayısal değerlere dönüştürmek için kullandık.

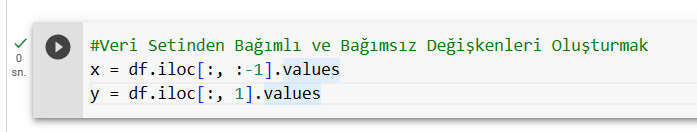
transform() metodu öğrenilen dönüşümü kullanarak etiketleri sayısal değerlere dönüştürür.

Buradaki dönüştürme işlemini yapmamızın sebebi veri setimizde sadece ‘age’ isimli sütunun

sayısal veri türü olup diğer bütün sütunların karakter veri tipinde olmasıdır. Veri türlerimiz sayısal

olmadığı için regresyon çeşitlerini yapamamıştık bu dönüştürme ile artık regresyon çeşitlerini

uygulayabiliriz.



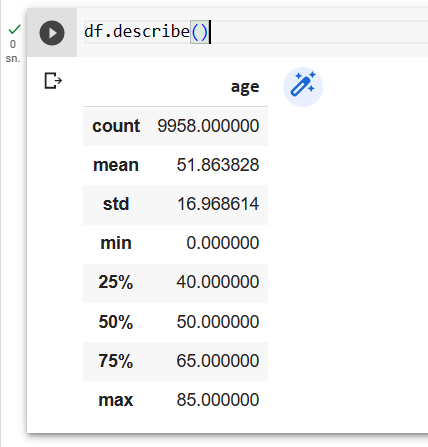
Yukarıdaki kod bloğu ile bağımlı ve bağımsız değişkenlerimizi oluşturduk. x bağımsız

değişkendir, y bağımlı değişkendir. Bu kod, veri çerçevesindeki sütunları bağımsız ve bağımlı

değişkenlere ayırır. **x** değişkeni, tüm sütunları içeren bir NumPy dizisidir, ancak son sütunu

içermez. **y** değişkeni ise ikinci sütunu içeren bir NumPy dizisidir ve bağımlı değişkeni temsil eder.

Bu şekilde, makine öğrenimi modeli için giriş ve çıkış değişkenlerini hazırlanmış olduk.



Yukarıdaki kodda yer alan **describe()** kodu veri çerçevesinin temel istatistiksel özetini

sağlayan bir metoddur. Bu metot, veri çerçevesindeki sayısal sütunların dağılımı, merkezi eğilimleri

ve dağılım ölçütleri gibi istatistiksel bilgileri görüntüler.

Kod çıktısını yorumlamak gerekirse:

**Count :** Veri çerçevesindeki değerlerin sayısıdır.

**Mean :**  Ortalama değer. Yani veri setimizdeki yaş ortalaması yaklaşık olarak 51.86’dır.

**Std :** Standart sapma, bir veri kümesinin dağılımının ölçüsüdür. Bir veri kümesindeki

değerlerin ne kadar yayıldığını ve ortalamadan ne kadar uzaklaştığını gösterir. Bizim standart

sapmamız 16.96 çıkmıştır. Tek başına standart sapmaya bakılarak verilerin durumu hakkında

kesin bilgiye ulaşılamaz ama standart sapmamız normalin biraz daha üzerinde olduğu için

verilerimizin ortalamadan daha fazla yayıldığı sonucuna ulaşılabilir.

**Min :** Minumum değer. İstatiksel özete göre minumum değerimiz sıfır çıkmıştır.

**%25 :** 1. çeyrek değerimiz 40’dır. Veri setinin küçüklük sırasına göre sıralandığında,

ilk çeyrek olan yüzde 25'lik dilimi ifade eder. Yani, veri setinin en küçük değerinden başlayarak,

küçükten büyüğe doğru sıralandığında, ilk çeyrek değerine karşılık gelir.

**%50:** 2. çeyrek değerimiz 50’dir. Veri setinin küçüklük sırasına göre sıralandığında,

ortadaki yüzde 50'lik dilimi ifade eder. Yani, veri setinin en küçük değerinden başlayarak,

küçükten büyüğe doğru sıralandığında, ortadaki değere karşılık gelir. Medyan, veri setini ikiye

böler ve yarısının üzerinde, yarısının altında olduğu değerdir.

**%75:** 3. çeyrek değerimiz 65’dir. Veri setinin küçüklük sırasına göre sıralandığında,

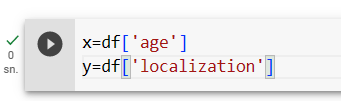
üçüncü çeyrek olan yüzde 75'lik dilimi ifade eder. Yani, veri setinin en küçük değerinden başlayarak,

küçükten büyüğe doğru sıralandığında, üçüncü çeyrek değerine karşılık gelir. Bu değer, veri setinin

üst yüzde 25'ini temsil eder.

**Max:** Maksimum değeri ifade eder. Maksimum değerimiz 85 çıkmıştır. Yani bu demektir ki

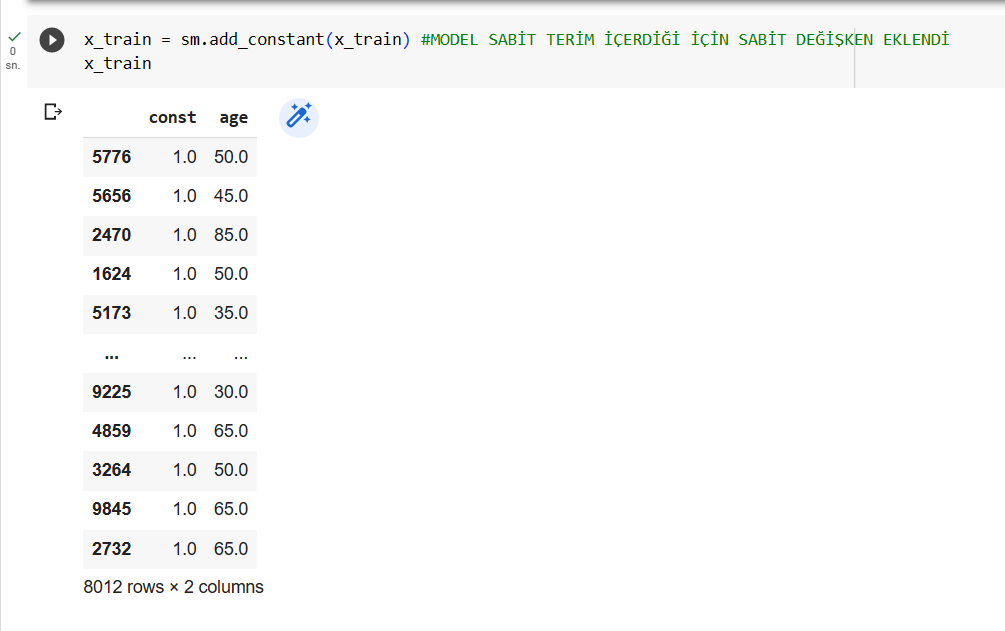
veri setimizdeki en büyük yani maksimum yaş 85’dir.



Yukarıdaki kodda x değişkenini ‘age’ (yaş) y değişkenini ‘localization’ (lokasyon) olarak

belirledik. x bağımsız değişken, y ise bağımlı değişkendir. Yani burada bağımsız değişkenimiz

yaş olarak belirlenmiş olup bağımlı değişkenimiz ise lokasyon olarak belirlenmiştir.



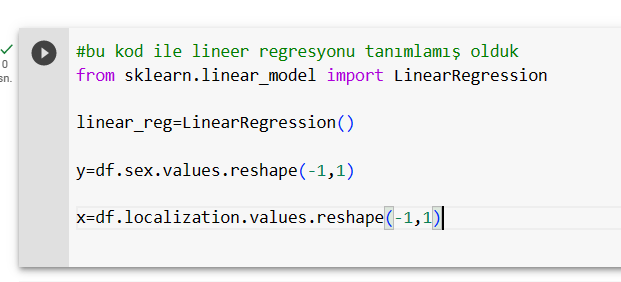
Yukarıdaki kodda sabit terim ekleme işlemi gerçekleştirdik. sm.add\_constant fonksiyonu

veri setine sabit bir terim ekler. Bu genellikle lineer regresyon modellerinde kullanılır çünkü lineer

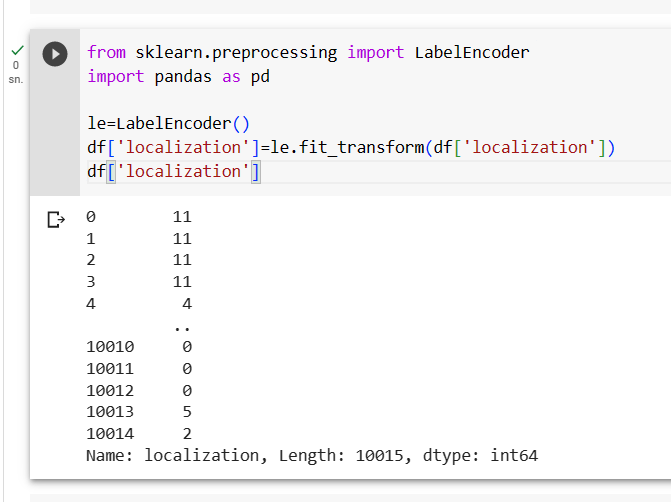
regresyon modeli, sabit bir terim içerir. Sabit terim doğru regresyon katsayılarının tahmin edilmesine

yardımcı olur. Modelin doğru şekilde tahmin yapabilmesi ve yorumlama yapılabilmesi için sabit

terim eklenmesi önemlidir.



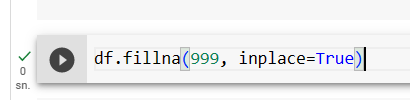
Yukarıdaki kod ile lineer regresyonumuzu tanımlamış olduk.



Yukarıdaki kod ile kategorik verilerin sayısal olarak kodlanmasını gerçekleşmiş oldu. Bu

dönüşüm, kategorik değerlerin orijinal sıralamasına göre sayısal bir temsilini oluşturur. Sonuç olarak, **df['localization']** sütunu artık sayısal olarak kodlanmış kategorik verileri içerecektir. Bu kodlama,

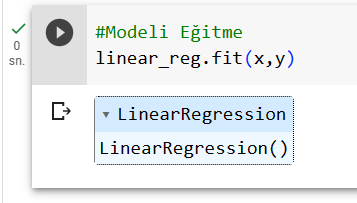
kategorik verilerin makine öğrenimi modellerine veya istatistiksel analizlere girdi olarak kullanılmasını kolaylaştırır.



Bu kod satırı, pandas DataFrame üzerinde eksik değerleri belirli bir değerle doldurmayı sağlar.

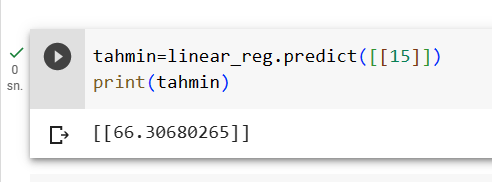
999 eksik değerleri doldurmak için kullanılan değeri temsil eder. Bu durumda, Null değerlerinin

yerine 999 değeri atanacaktır. DataFrame üzerinde bu değişiklikler kalıcı olarak uygulanır.



Bu kod lineer regresyon modelini eğitmek için kullanılır. Model, veri kümesi üzerindeki

ilişkiyi öğrenir ve bağımlı değişkeni (**y**) bağımsız değişkenlere (**x**) bağlamaya çalışır.



Bu kod parçası, eğitilen lineer regresyon modeli linear\_reg kullanılarak bir tahmin

yapmayı ve sonucu yazdırmayı gösterir. 15 değeri için bir tahmin yapılır ve bu tahmin değeri

**tahmin** değişkenine atanır. Ardından bu tahmin değeri yazdırılır. Tahmin değerimiz 66.30

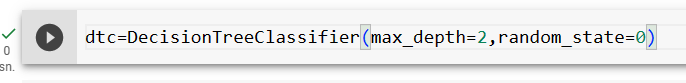
çıkmıştır.



DecisionTreeClassifier sınıfı, ağaç yapısını oluşturmak ve sınıflandırma işlemini

gerçekleştirmek için çeşitli parametreler ve yöntemler sunar. Bu sınıf kullanılarak veri setinin

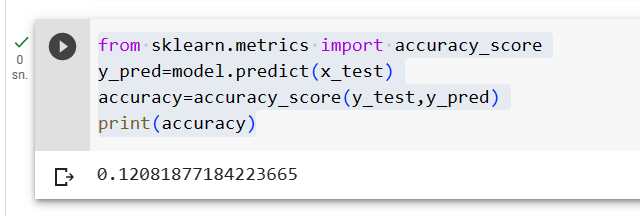
özelliklerine dayalı olarak bir karar ağacı sınıflandırma modeli oluşturulabilir ve eğitilebilir.



Bu kod satırı, DecisionTreeClassifier sınıfından bir nesne oluşturarak bir karar ağacı

sınıflandırma modelini yapılandırmayı gösterir. Model, veri setindeki özellikleri kullanarak

sınıflandırma yapmak için eğitilebilir.



Bu kod sınıflandırma modelinin doğruluk skorunu hesaplamayı ve sonucu yazdırmayı

göstermektedir. Modelin test verileri üzerindeki doğruluk skoru hesaplanır ve ekrana yazdırılır.

Doğruluk skoru, doğru tahmin edilen etiketlerin toplam veri noktalarına oranını ifade eder. 0.1208

gibi düşük bir doğruluk skoru, modelin test verileri üzerindeki performansının düşük olduğunu gösterir.

Bu, modelin veriler arasındaki ilişkiyi yeterince iyi yakalayamadığı veya modelin yanlış sınıflandırmalar

yaptığı anlamına gelebilir.

Düşük doğruluk skoru, modelin eğitim verilerine fazla uyum sağlamış olabileceğini veya veri

setindeki dengesizliklerden kaynaklanabileceğini düşündürebilir. Ayrıca modelin seçilen

parametrelerinin veya kullanılan algoritmanın uygun olmadığı durumlar da düşük doğruluk skoruna

neden olabilir. Bu durumda, modelin performansını artırmak için farklı bir model seçilebilir, modelin parametreleri ayarlanabilir veya veri ön işleme teknikleri kullanılabilir. Ayrıca, daha fazla veri toplama

veya mevcut veri setinin dengesizliklerini düzeltme çabaları da doğruluk skorunu iyileştirebilir.

# KAYNAKÇA

YARARLANILAN MAKALELER:

http://acikerisim.harran.edu.tr:8080/jspui/bitstream/11513/2715/1/682539.pdf

https://acikerisim.fsm.edu.tr/xmlui/bitstream/handle/11352/4381/Sar%20.pdf?sequence=1

https://dergipark.org.tr/en/download/article-file/2268361

https://dergipark.org.tr/en/download/article-file/1300890

http://www.eleco.org.tr/ELECO2022/eleco2022\_papers/159.pdf

https://www.researchgate.net/profile/Beyza-Akilotu/publication/334112417\_Classification\_of\_Dermoscopic\_Images\_Using\_Deep\_Features\_and\_Color/links/5d175c92458515c11c00ee42/Classification-of-Dermoscopic-Images-Using-Deep-Features-and-Color.pdf

SAYFALAR:

https://medium.com/@yasinguzel/yapay-zeka-ders-notlar%C4%B1-04-yapay-sinir-a%C4%9Flar%C4%B1n%C4%B1n-yap%C4%B1s%C4%B1-88f60c5d2e1d

https://www.researchgate.net/publication/337687860\_Danismanli\_ve\_yari\_danismanli\_ogrenme\_kullanarak\_dokuman\_vektorleri\_tabanli\_tweetlerin\_duygu\_analizi

https://www.datasciencearth.com/boyut-azaltma-mantigi-ve-ornek-bir-temel-bilesenler-analizi-uygulamasi/

https://www.veribilimiokulu.com/polinom-regresyon-python-uygulama-1/

VİDEOLAR:

https://www.youtube.com/watch?v=k0OhRWiCD8Y

https://www.youtube.com/watch?v=0Yo2j9nNmjI