**Lineer Regresyon Analizi**

Lineer regresyon, 2 nicel veri arasındaki ilişkiyi özetleyen istatiksel bir metottur. X ekseninde gösterilen 1.değişken tahmin edici, bağımsız değişkendir. Y ekseninde gösterilen 2.değişken ise tahmin edilen çıktı ise bağımlı değişkendir.

Lineer regresyon ile bulunan bu ilişki, istatistiksel bir ilişkidir.

Sonuç olarak karşımıza istatistiksel bir grafik çıkacak. Örn:

A close up of a map

Description automatically generated

* Matris işlemlerini yapmak için “numpy” kütüphanesi eklendi.
* X ve Y değerleri DataFrame formatında olacağı için “pandas” kütüphanesi eklendi.
* Görsel sonuç elde etmek için grafik çizmeye yarayan “pyplot” kütüphanesi eklendi. (Kütüphaneleri eklemek için üzerinde gelip Örn: Install Package Pandas)
* Dizi değişkeni b0 ve b1’i tutacak
* Deger değişkeni sabit değer verilip verilmeyeceği kontrol edilecek.
* Denk değişkeni daha önce Denklem metodu kullanılıp kullanılmadığı kontrol edilecek.
* Def ConverToNumpy
* Pandas kütüphanesinin eldeki DataFrame’i Numpy kütüphanesindeki Array’e çevirecek.
* Def Transpoze
* Eğer 1 satırlık veri dizisi gelirse devriğini alıp 1 sütunluk veri dizisine çevirecek.
* .ndim🡪 Numpy’ye ait array dizisinin boyutunu verir.
* .reshape🡪Kaç satır ve sütun olacağını belirtir.(Satır sayısı belli olmadığı için -1 değeri verildi.)
* Def DegerEkle
* .shape🡪Kaç satır ve sütun olduğunu verir.
* .ones🡪İstenen satır ve sütunda matris oluşturur.(Sadece 1 değerini içerir.)
* .hstack🡪İki matrisi yan yana ekler.
* Def Denklem
* Matrisin eşit olacağı denklemi kurup son değeri aldık
* [b1]  
  [b0]  
   =((((X^T).X)^-1)\*((X^T).Y)  
   1-Değerlerin(X) transpozunu al.(X^T)  
   2-Kendisiyle çarp.((X^T).X)  
   3-Sonucun tersini al.(((X^T).X)^-1)  
   4-Değerlerin transpozunu al.(X^T)  
   5-Sonuç verilerinin olduğu matrisle çarp.(X^T).Y  
   6-3 numara ve 5 numarada elde ettiğin sonuçları birbirleriyle çarp.((((X^T).X)^-1)\*((X^T).Y)
* .inv🡪Matrisin tersini alır.
* .dot🡪İki matrisi birbiriyle çarpar.
* .T🡪Transpozunu alır.
* Def TahminEt
* Önce Denklem metodundaki işlemlerin yapılıp yapılmadığı kontrol edildi. Yapılmadıysa hata gösterildi.
* Eldeki veri (X) Array’e çevirildi.
* Sabit değer istenip istenmediği kontrol edildi. Eğer istendiyse sabit değer olarak 1 verildi.
* Verilerle (X) Dizi değişkeninin içindeki b0 ve b1 çarpıldı.
* Def ToplamDeger
* .mean🡪Aritmetik ortalama alır.
* X ve Y Array’e çevirildi.
* Veriler TahminEt metodundan geçti ve tahmin değişkenine aktarıldı.
* Tahmin edilen değer (tahmin)’den gerçek değer (Y) çıkarılıp karesi alındı.
* DataFrame formatında csv dosyasını programa ekleyip (kaggle.com) read\_csv komutuyla verileri okuttuk.
* .series komutuyla weight ve height adı altındaki 2 seriyi oluşturup çağırdık.
* 2 seriyi birleştirmek için concat fonksiyonunu kullandık.
* Axis=1 kolonlar ile işlem yapmak için kullanılır. Satırlar için axis=0 olarak ayarlanır.
* İslemDframe kısmında %70’lik kısmı işleme, testDframe kısmında %30’luk kısmı teste ayırdık.
* X eksenine height, Y eksenine weight yazdırdık.
* X eksenine height verilerini, Y eksenine weight verilerini çağırdık ve Y ekseni için TahminEt metodunu döndürdük.
* Pyplot kütüphanesini kullanarak X ve Y değişkenlerini çağırıp sapma çizgisinin rengini kırmızı olarak ayarladık.

