

**T.C.**

**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Büyük Veriye Giriş Dersi Proje Ödevi**

Titanic - Machine Learning from Disaster

B191210004 – Mustafa Melih Tüfekcioğlu – 1.Öğretim A Grubu

B181210010 - Deniz Berfin Taştan – 1.Öğretim B Grubu

SAKARYA

Aralık 2022

Büyük Veriye Giriş Dersi

**1.Proje Özeti**

Proje Titanic veri seti kullanarak bir model eğitmek üzerine yapıldı. Eğitilen model gemi enkazından hangi yolcuların sağ çıktığını tahmin etmek için kullanılacak. Modeli eğitmek için izlenecek adımlar şu şekilde:

1. Kullanılacak olan veri setini analiz etme
2. Veri setindeki değişkenlerin birbirine gör bağımlılıklarını hesaplama
3. Tespit edilen bağımlılıklara göre kullanılacak olan değişkenleri tespit etme
4. Model eğitimi için kullanılacak veri setini eğitime uygun hale getirme (boş alan doldurma, değişken birleştirme vs.)
5. Final veri seti oluşturma
6. Model eğitme ve çıkan doğruluk değerlerine göre algoritma seçimi
7. Sonuçlandırma

**2. Projede Kullanılacak Veri Seti Hakkında**

**2.1 Eğitim için kullanılacak veri seti (train.csv)**

Modeli eğitmek için kullanılacak olan Train.csv dosyası 892 tane veriden oluşuyor. Her veri bir kişiyi temsil etmekte ve dosyada bir kişinin aşağıdaki bulunmaktadır.

-PassengerID (Her kişi için verilen bir ID numarası)

-Survived (Kişinin kaza esnasında hayatta kalıp kalmadığı hakkında bilgi)

-PClass (Kişinin hangi bilet sınıfına sahip olduğu bilgisi)

-Name (Kişinin adı ve soyadı)

-Sex (Kişinin cinsiyeti)

-Age (Kişinin yaşı)

-SibSp (Kişinin eş/kardeş sayısı)

-Parch (Kişinin ebeveyn/çocuk sayısı)

-Ticket (Kişinin bilet bilgisi)

-Fare (Kişinin ödediği ücret)

-Cabin (Kişinin kabin bilgisi)

-Embarked (Kişinin gemiye nereden bindiğinin bilgisi)

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu Veri seti içerisinde tekrarlanan bir veri bulunmuyor fakat bazı sütunlarda boş alanlar bulunuyor. Verileri matematiksel olarak sınıflamak istediğimizde de aşağıdaki tabloyu elde ediyoruz.

**2.2 Test için kullanılacak veri seti (test.csv)**

Test için kullanılacak veri seti 419 tane veriden oluşuyor. Test veri seti, eğitim veri seti içindeki tüm sütunlara sahip fakat test veri setinde Survived sütunu bulunmuyor. Veri seti içerisinde tekrarlanan bir veri bulunmuyor fakat bazı sütunlarda boş alanlar bulunuyor.

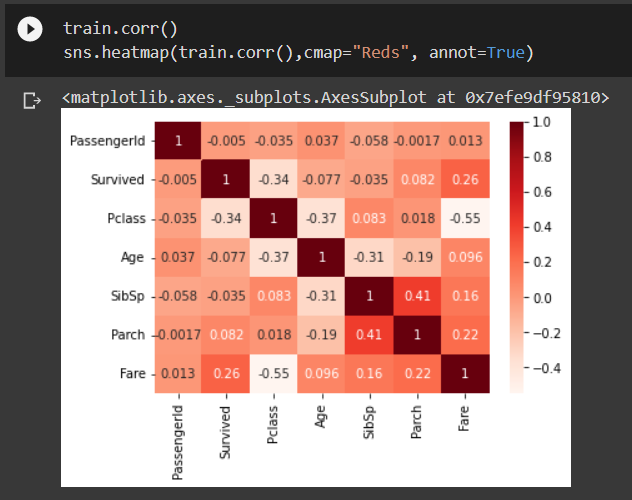
**3. Veri Setindeki Bilgilerin Birbirine Göre Bağımlılıkları**

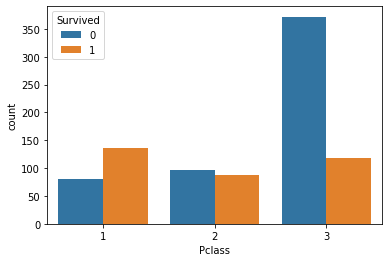
Veri setindeki değişkenlerin birbirlerine göre bağımlılıklarını ölçmek için korelasyon katsayısı hesaplaması yapılır. Bu katsayı -1 ile +1 arasında bir değer alır. Katsayı +1'e veya -1'e ne kadar yakınsa ilişkinin bağımlılığı o kadar güçlüdür. Korelasyon katsayısı aşağıdaki formül ile hesaplanır.

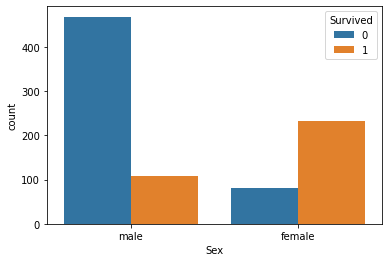
metin içeren bir resim

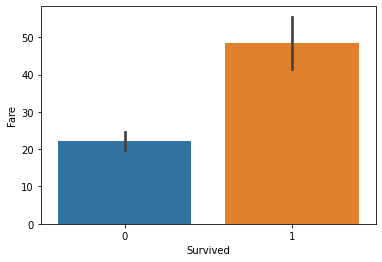
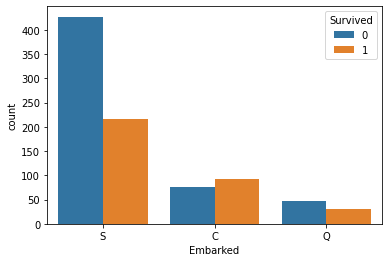
Açıklama otomatik olarak oluşturuldu

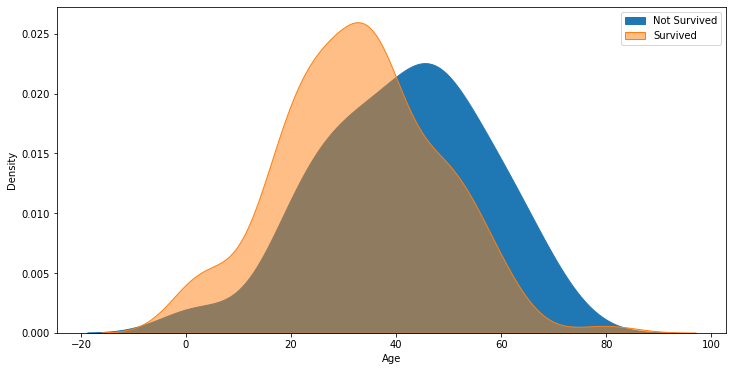
Python’da korelasyon katsayını hesaplamak için seaborn kütüphanesinin corr() fonksiyonu kullanılır. Veri setimiz için bu fonksiyonu kullandığımızda aşağıdaki tabloyu elde ediyoruz.



***Hayatta kalma bilgisinin diğer değişkenlerle olan grafikleri:***







**4.Veri Düzenleme**

Veri setlerini projemizde kullanmamız için önce düzenlememiz gerekiyor. Önce veri setleri içerisindeki boş alanları doldurma işlemi yaptık.

**Train.csv** içerisindeki boş alanları bulmak için **“.isnull().sum()”** fonksiyonunu kullandık. Çıkan sonuca göre *yaş(age), konum (embarked) ve kabin (cabin)* alanlarında boşluklar bulunuyor.

* *Yaş (age)* bilgisindeki boşlukları doldurmak için ilgili veri setindeki yaşların ortalamasını hesapladık ve çıkan sonucu boş alanlara eklendi.
* *Kişinin gemiye bindiği konum bilgisindeki (Embarked)* boş alanları doldurmak için ilgili sütundaki en çok tekrar eden veri olan “S” değeri boş alanlara eklendi.
* *Kabin (Cabin) bilgisinin* elde etmek istediğimiz hayatta kalma (survived) bilgisini tahmin etmede etkisi olmadığı için veri setinden silindi.

**Test.csv** içerisindeki boş alanları bulmak için tekrar aynı fonksiyonu kullandık. Çıkan sonuca göre yaş(age), ücret(fare) ve kabin(cabin) alanlarında boşluklar bulunuyor.

* *Yaş (age)* bilgisindeki boşlukları doldurmak için ilgili veri setindeki yaşların ortalamasını hesapladık ve çıkan sonucu boş alanlara eklendi.
* *Ücret (fare)* bilgisindeki boşlukları doldurmak için ilgili veri setindeki ücretlerin ortalamasını hesapladık ve çıkan sonucu boş alanlara eklendi.
* *Kabin (Cabin) bilgisinin* elde etmek istediğimiz hayatta kalma (survived) bilgisini tahmin etmede etkisi olmadığı için veri setinden silindi.

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

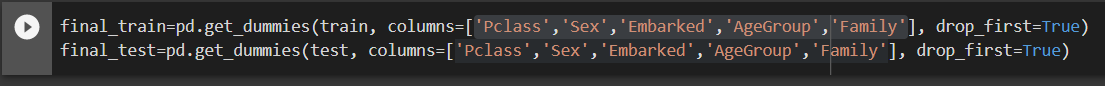
**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**Boş alanları doldurduktan sonra veri setlerindeki ihtiyaç olmayan verileri silmemiz gerekiyor. Örneğin *isim (name)* ve *bilet (ticket)* bilgilerinin *kişinin hayatta kalma (survived)* bilgisine bir etkisi olmadığı için ilgili sütunlar silindi. Daha sonra ilgili verileri birleştime işlemi yapıldı. *Kardeş/eş sayısı (Sibsp) ve ebeveyn/çocuk (Parch)* sayısının toplamı *temp\_family* geçici değişkenine atandı ve bu sütunlar veri setinden silindi.

metin, ekran görüntüsü, ekran, siyah içeren bir resim

Açıklama otomatik olarak oluşturulduBirleştirme işleminden sonra veri setlerine *Family* sütunu eklendi ve kişiler aile sayısına göre etiketlendi. Daha sonra veri setlerine *AgeGroup* sütunu eklendi ve kişiler yaş gruplarına göre etiketlendi.

**5. Final Veri Seti Oluşturma ve Model Eğitimi**

Veri düzenleme işlemlerinden sonra model eğitimi ve sonuç tahmin işlemleri için kullanılacak olan final veri seti oluşturuldu. Bu veri setine eğitimde kullanılacak olan *Pclass, Sex, Embarked, AgeGroup, Family* bilgileri eklendi.

metin içeren bir resim

Açıklama otomatik olarak oluşturulduTanımlanan **x\_test** değişkeni bağımsız değişkenleri temsil ettiği için tahmin edilmesi istenen ve bağımlı değişken olan *Survived* sütununu veri setinden siliyoruz. **Y\_test** ise bağımlı yani tahmin edilecek olan değişkeni ifade ediyor. Model eğitiminde kullanılacak olan verileri test ve train olarak bölmek için de **Sklearn** kütüphanesinin **train\_test\_split** fonksiyonunu kullanıyoruz. Bu fonksiyonun içerisine yazılan **test\_size** parametresini 0.2 vererek final veri setinin %80’inini eğitim için, %20’sini de test için kullanıyoruz.

**6. Sonuçlar**