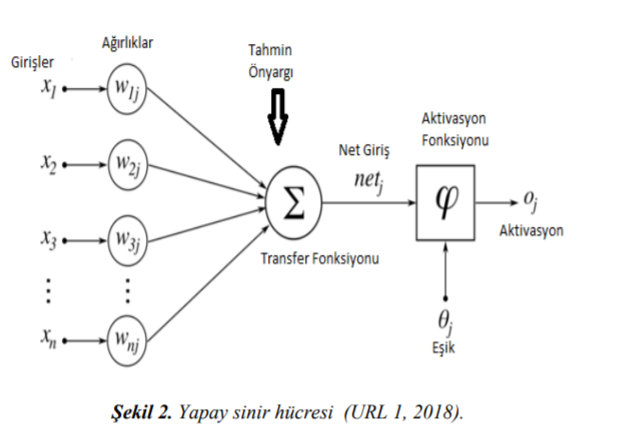
**1.Proje hakkında**

Proje insanların gelirlerini , eğitimleri ,meslekleri, ülkeleri gibi kriterlerin bulunduğu veri setini bir kişinin yıllık kazancının 50.000 üzerinde veya altında olup olmadığına kontrol eden bir sınıflandırma problemidir.

**2. Yapay Sinir Ağları**

Yapay sinir ağları (YSA), insan beyninin öğrenme yolunu taklit ederek beynin öğrenme, hatırlama, genelleme yapma yolu ile topladığı verilerden yeni veri üretebilme gibi temel işlevlerin gerçekleştirildiği bilgisayar yazılımlarıdır. Yapay sinir ağları; insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır. Özelikle insansı robotlarla birlikte yapay zekâ ve yapay sinir ağları çalışmaları hızlanmıştır. Burada öğrenme ve öğrendiği en doğru bilgiyi uygulayabilme prensibi esastır



**Girdi Katmanı (Input Layer)**: Bir ağa girdi olarak gelen öğrenilmesi istenen örneğin özniteliklerinin giriş olarak verildiği katmandır. Girdi katmanında öğretilecek örneklerin öznitelik sayısı kadar giriş nöronu bulunmalıdır.

**Çıktı Katmanı (Output Layer):**Yapay ağdaki öğrenilmesi istenen örneklerin sınıf bilgisinin veya etiket değerinin çıkış olarak hesaplandığı katmandır.

**Transfer Fonksiyonu**

Bir nörona gelen net girdiyi hesaplar. Bunun için değişik fonksiyonlar kullanılmaktadır. En yaygın olan ise ağırlıkların girdiler ile çarpımının toplamıdır.

**Gizli Katmanlar (Hidden Layers)**

Giriş katmanı ile çıkış katmanları arasıdaki katmalardır. Katman sayısı ve üzerindeki nöron sayısı problemden probleme değişebilmektedir. Bu katmalarda ileri yönlü hesaplamalar ve geri yönlü hata yayılımı yapılır. Katman sayısının çok olması hesaplama karmaşıklığına ve hesaplama süresinin artmasına sebep olur. Karmaşık problemlerde problem çözümü için genelde katman sayıları ve katmanlardaki nöron sayıları fazladır. Yapay sinir ağlarında, gizli katmanlar, yalnızca verilerin doğrusal olmayan bir şekilde ayrılması gerekiyorsa gereklidir

**Ağırlıklar (Weights)**

Girdilerin çıktılar üzerindeki etkisini ayarlayabilmek için kullanılan parametrelerdir.

Ağırlıklar girdi değerleri ile çarpılarak ileri doğru iletilir. Ağdaki **en kritik nokta,** verilen ağırlıkların eğitim kümesine göre hatayı yayarak olması gereken optimum ağırlık değerlerini hesaplamaktır. Bu ağırlıklar pozitif, negatif veya sıfır olabilir. Ağırlığı sıfır olan nöronların çıktı üzerinde etkisi olmamaktadır.

**Öğrenme Oranı**

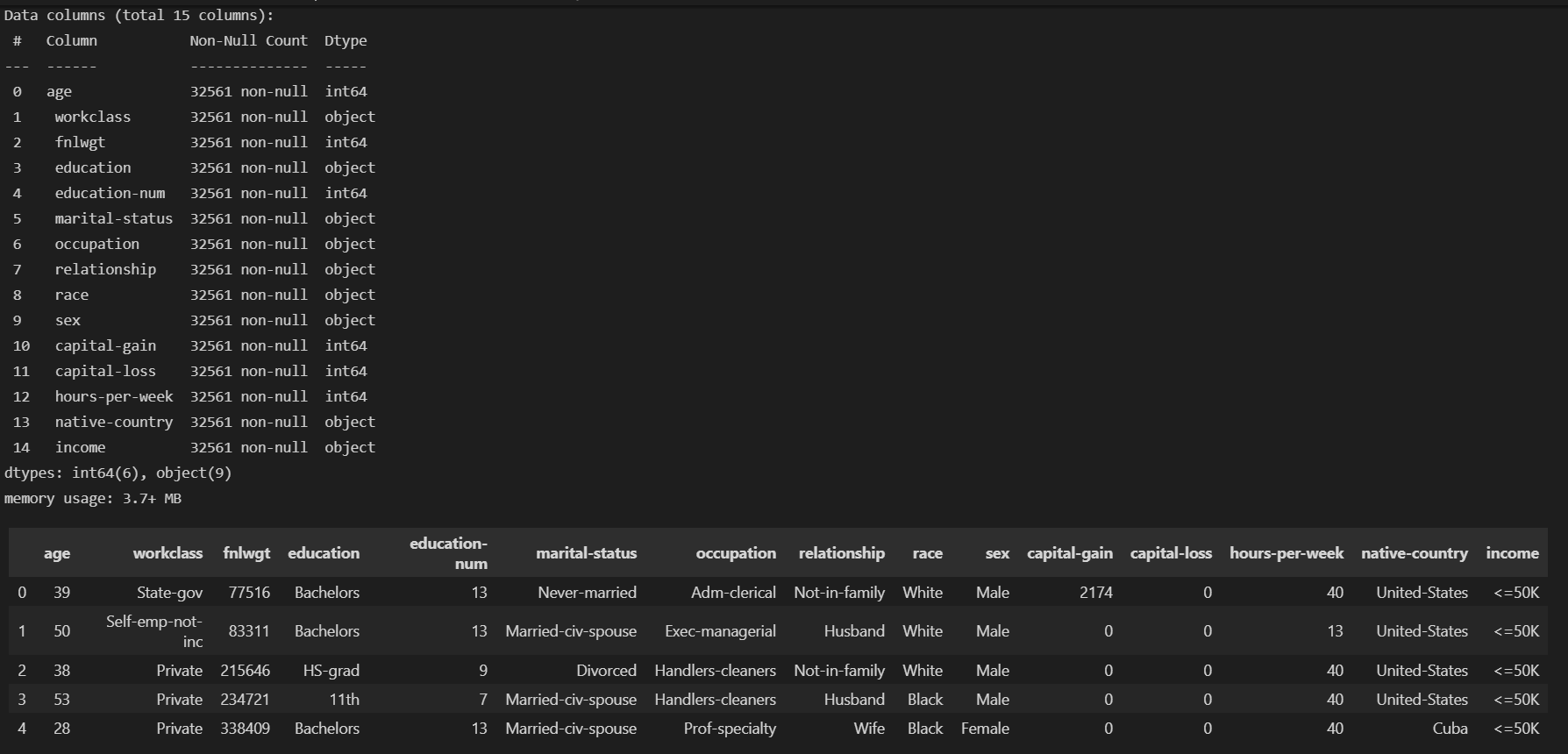
Öğrenme oranı, bir yapay sinir ağı modelinin her bir adımda oluşan hatalara göre modeldeki ağırlıkların ne kadar güncelleneceğini belirler. Yüksek öğrenme oranı, eğitim süresini kısaltır fakat optimum accuracy değerini bulmayı zorlaştırır. Aynı zamanda düşük öğrenme oranı eğitim süresini uzatır fakat optimum accuracy değerine ulaşabilme potansiyelini arttırır. Learning rate tecrübelere dayalı olarak genelde 0.4-0.6 arası seçilir. Ama bu değeri farklı problemlerde en optimum değerini bulmak için Goldstein, Wolfe condition, BFGS gibi optimizasyon algoritmaları kullanılır.

**Aktivasyon Fonksiyonu**

   Aktivasyon fonksiyonunda yapay sinir hücresi girdi verileri üzerinde işlem yaparak buna karşılık gelen net çıktı sonuçları elde eder. Bu fonksiyon genelde doğrusallık göstermez. Fonksiyonun doğru seçilmesi önemlidir. Çünkü sonuç performansı etkileyecektir. Fonksiyon tek ve çift iki kutup halinde olabilir. Bazı aktivasyon fonksiyonları şunlardır; Parçalı doğrusal fonksiyon, Hiperbolik tanjant Aktivasyon fonksiyon, Adımsal aktivasyon fonksiyonu gibi Sinir ağına verilen örnek sayısı optimum değerden fazla ve aşırı iterasyon ile eğitilmesi durumunda sinir ağı öğrenmemiş, ezberlemiş olacaktır.

Optimum değerden az eğitildi ise örnekleri öğrenememiş (**Underfitting**) olacaktır. Bu gibi durumlardan kaçınılmalıdır. Aksi durumda test kümesindeki ağın başarı oranı düşük çıkacaktır.

**3.Veri Seti**

****

Veri setimizAge, workclass, fnlwgt**,** education**,** education-num, marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country sütunlarından oluşmaktadır.32561 verimiz bulunmaktadır.Veri setimiz bize bir kişinin yıllık kazancının 50.000 üzeri olup olmadığını tahmin etmemizde yardımcı olacaktır.

**4. Veri Seti Ön İşleme**

Veri setimizdeki numerik olmayan yani string değerleri Sklearn kütüphanesinden LabelEncoder fonksiyonu ile numerik(sayısal)değerleri dönüştürdük.Bu sayede veri setimizdeki bütün değerler integer olarak tutuluyor**.**

**metin içeren bir resim

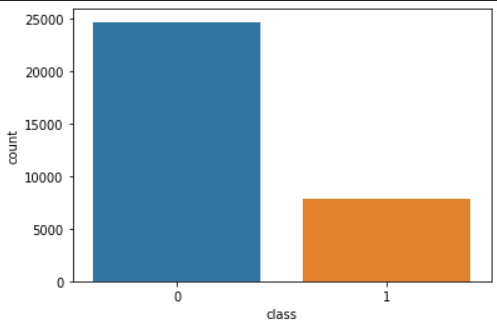
Açıklama otomatik olarak oluşturuldu**

Veri setimizdeki boş(null) olan değerleri kontrol ettim ve toplamını ekrana yazdırdım.

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Hedef değişkenlerinin verisinin görselleştirilmesi .Veri setindeki 50.000 üzeri maaş alanlar için ‘1‘ , alamayanlar için ‘0’ seklinde grafik çıktısı.

****

Veri setinin string değerleri ile işlemlerin ardından verinin daha okunabilir hale gelmesi için normalizyon ile özellik aralığı(feature\_range) (1,0) aralığına sınırlandırılmıştır.

**metin, ekran görüntüsü, ekran, birkaç içeren bir resim

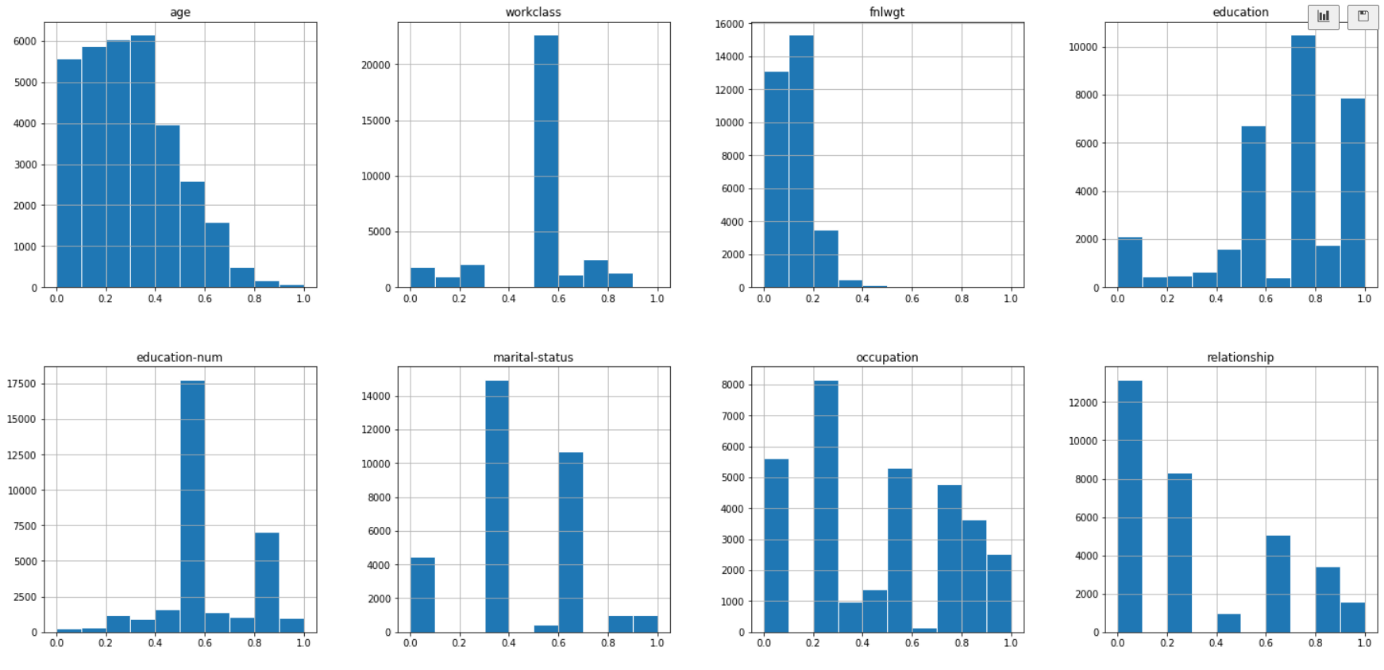
Açıklama otomatik olarak oluşturuldu**

Burada featurelarımızın ağırlıklarını görselleştirdik.

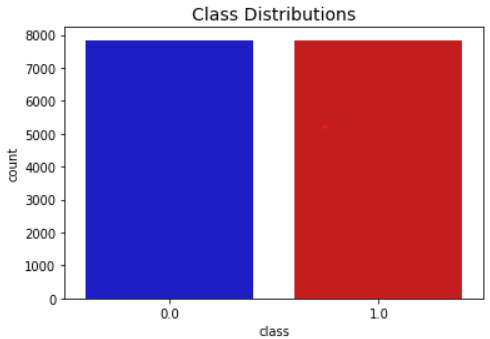
metin, saat içeren bir resim

Açıklama otomatik olarak oluşturuldu

Featurelarımızın ağırlıklarının histogram halinde gösterilmesi



Class(target) 0 ve 1 değeleri için kontrol edilmiştir. bu grafikte görüldüğü gibi iki class arasında büyük veri farkı bulunmaktadır. Böyle bir durumda model 0 classına yönelimli olacağından bu fark düşürülmektedir. Bundan dolayı verilerin eşit sayıya getirilmesi 24720 adet bulunan 0 classı 7841 adete rastgele olarak düşürülmüştür.(random under sampling)

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Train ve Test Verilerinin Hazırlanması**

   Model içerisinde kullanmak amacıyla verinin 2 ayrı katmana bölünmesi gerekmektedir.

**Train:** Veri setindeki bu katman modelin eğitilmesi amacıyla kullanılır.

**Test:** Oluşturulmuş olan modelin performansının tespit edilmesi için ayrılan alandır. Bu bölümde modele verilerek tahmin edilen değerlerin, olması gereken değerler ile karşılaştırılması yapılmaktadır.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu model içerisinde kullanmak amacıyla veri seti %30 test ve %70 train için bölünmüştür.

**Modeller ve sonuçlar**

Eğitim sırasında aşağıdaki parametler optimize edilmiştir.

- Batch size

- Learning rate

-Number of epochs

-Number of final fully connected layers

**Sonuçlar**

Modellerimizi aşağıdaki farklı verilere göre çıkan sonuçlardan (Accuracy) doğruluk oranı en yüksek 5 tane modeli alıp en iyi sonuçları almak epochs yükselterek tekrar denedik.

Aşağıda modelimizde kullandığımız recall,precision,f1 score değerlerinin fonksiyonları yer alıyor.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Öğrenme katsayısı (λ) ağırlıkların değişim miktarını belirlemektedir. Eğer büyük değerler seçilirse o zaman yerel çözümler arasında ağın dolaşması ve osilasyon yaşaması söz konusu olmaktadır. Küçük değerler seçilmesi ise öğrenme zamanını artırmaktadır. Tecrübeler genellikle 0.2-0.4 arasındaki değerlerin kullanıldığını göstermiştir. Bazı uygulamalar öğrenme katsayısının 0.6 değerini aldığı zaman en başarılı sonuçları verdiği göstermiştir. Bu durum tamamen probleme bağlıdır.

Aktivasyon fonksiyonları sigmoid ve tanh üzerinde farkı değerlerde learning rate ve momentum rate ile modelimizi bir çok kez denedik. En optimum sonuç aşağıdaki değerler sonucunda elde edilmiştir.

[Learning Rates = 0.01 ], [Momentum rates=0], [Hidden layers = [20,10,1] ]

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Modelimizin sonucu

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

***Referanslar:***

<https://www.kaggle.com/lodetomasi1995/income-classification>

<https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/>

<https://kadirguzel.medium.com/geri-yay%C4%B1l%C4%B1ml%C4%B1-%C3%A7ok-katmanl%C4%B1-yapay-sinir-a%C4%9Flar%C4%B1-1-47daa3856247>

<http://www.ibrahimcayiroglu.com/Dokumanlar/GoruntuIsleme/Goruntu_Isleme_Ders_Notlari-11.Hafta.pdf>

<https://dergipark.org.tr/tr/download/article-file/596690>

<https://www.veribilimiokulu.com/yapay-sinir-aglari/>

<https://keras.io/api/layers/activations/#layer-activation-functions>

<https://towardsdatascience.com/beginners-ask-how-many-hidden-layers-neurons-to-use-in-artificial-neural-networks-51466afa0d3e>

***Hazırlayan: Özgür Durak***

***Öğrenci Numarası: 170101012***

***Projenin Amacı: Gelir tahmini***