

EGE UNIVERSITY
FACULTY of ENGINEERING
COMPUTER ENGINEERING DEPARTMENT
204 DATA STRUCTURES (3+1)
2022-2023 FALL SEMESTER

PROJECT 1 : (Arrays, Matrices, Methods, Classes, Random Numbers)

*Bu proje tek kişi olarak hazırlanabileceği gibi, iki veya en fazla üç kişilik gruplar oluşturularak da yapılabilir.

I. YOLCULARI KOLTUKLARA ATAMA

Bir tur otobüsüne 40 yolcunun yerleştirilmesi planlanmaktadır. Aşağıdaki maddelerde belirtilen işlemleri yapan kodu yazınız:

10 p

a) **Uzaklık Matrisi Oluşturma:** 40 adet yolcunun birbirleri ile yakınlık derecelerinin bilindiği varsayılmaktadır. DM yani Uzaklık matrisi (40x40'lık) içerisine yolcu ikilileri arasındaki 0 ile 10 arasında rastgele double sayılardan oluşan yakınlık derecelerini içeren değerleri üretip döndüren metodu yazınız (Tablo 1). 40 elemanlı bir diziye de yolcu isimlerini (ad soyad) atayınız.

10 p

b) **Yolcuları Koltuklara Yerleştirme** (Yolcuları koltuklara yerleştiren metodu yazınız)
i) Rastgele (random) bir sayı üreterek 1 numaralı koltuğa dizinin o indisindeki yolcuyu atayınız. 2 numaralı koltuğa, uzaklık matrisinden yararlanarak 1 numaralı koltuğa yerleştirilen yolcuya en az uzaklıktaki yolcuyu atayınız. 3 ve 4 numaralı koltuklara da, kalan yolcular içinde kendi sollarındaki koltuğa yerleşen yolcuya en az uzaklıktaki yolcuları atayınız. İkinci sıraya geçince, 5. koltuk için hemen üstündeki (önündeki) ve üst çaprazındaki koltuklara (1 ve 2) yerleşen yolculara uzaklıkları toplamı en az olan yolcuyu atayınız. 6. koltuk için 5, 1, 2 ve 3.; 7. koltuk için 6, 2,3 ve 4.; 8. koltuk için 7, 3 ve 4. koltuklardaki yolculara uzaklıkları toplamı en az olan yolcuları atayınız. Atamalar, yerleşenler haricindeki yolcular dikkate alınarak yapılacaktır. Bu işlemi, tüm yolcular koltuklara yerleşene kadar tekrarlayınız.

10 p

c) Yolcuları ekrana Tablo 2'deki koltuk yerleşimi düzeninde 4x10 elemanlık bir matris şeklinde **listeleyniz**. Koltuklarda Yolcu numaraları ve Yolcu isimleri olacak şekilde.

5 p

d) b maddesindeki hesaba göre koltuklara yerleşen yolcuların toplam uzaklıklarını (sol yan + üst (ön) + üst sol çapraz + üst sağ çapraz) yine koltuk düzeninde koltukların üzerinde olacak şekilde **listeleyniz**. İlgili yönde koltuk yoksa o yön için 0 değeri alınacaktır.

5 p

e) b maddesine göre yerleşen yolcular arasındaki uzaklıklar toplamını yazdırınız (**tüm yolcuların uzaklıklar toplamı**). d maddesindeki değerlerin toplamı olan **tek değer** istenmektedir.

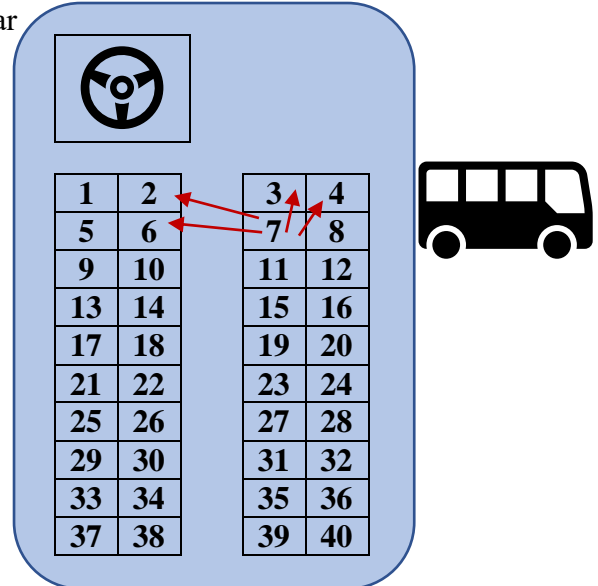
Tablo 2: Otobüs koltuk düzeni ve numaraları

Tablo 1: Yolcu Çiftleri arasındaki temsili uzaklıklar

Uzaklık Matrisi (DM)						
	0	1	2	3	...	39
0	0	1,2	0,5	4,7	...	4,9
1	1,2	0	3,1	2	...	4
2	0,5	3,1	0	6,1	...	1,9
3	4,7	2	6,1	0	...	3,5
...	0	...
39	4,9	4	1,9	3,5	...	0

$$DM[i,j]=DM[j,i]$$

(i'den j'ye uzaklık ile j'den i'ye uzaklık aynıdır).

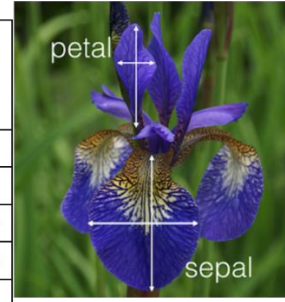


II. FLOWER CLASSIFICATION USING A SIMPLE NEURAL NETWORK YAPAY SİNİR AĞI ile ÇİÇEK SINIFLANDIRMA

Doğada gördüğümüz bir çiçeğin, Zambak (veya süsen) bitkisi olduğu biliniyor. Üç farklı türünden hangisine ait olduğunu bulduran bir algoritmanın yazılması istenmektedir. Elimizde her bir çiçek türünden 150 örnek üzerinden ölçülerek alınan veriler bulunmaktadır. Her bir örnek için 4'er adet özellik (çanak yaprak uzunluğu, çanak yaprak genişliği, taç yaprak uzunluğu, taç yaprak genişliği) ve hangi sınıfta (tür) olduğu bilgisi hazır olarak verilmektedir. Tablo 3'te 6 tanesine yer verilmiştir. Veriseti: <https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data> bağlantısında yer almakta olup projede kullanılacaktır.

Tablo 3: Çiçek veri setinden alınmış 6 adet bitki örneğine ilişkin bilgiler

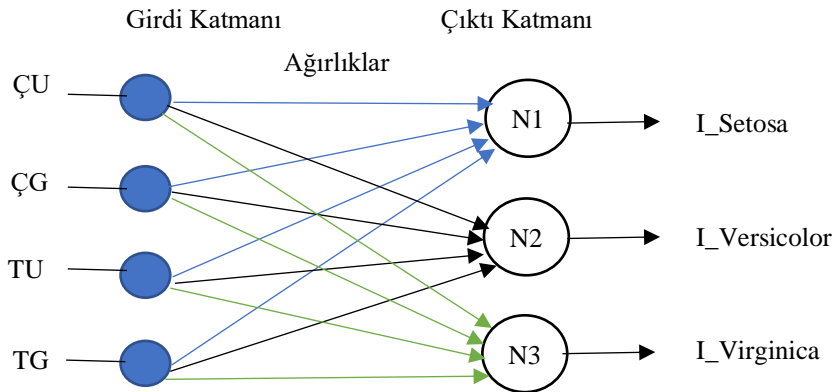
Örnek No	Çanak Yaprak Uzunluğu	Çanak Yaprak Genişliği	Taç Yaprak Uzunluğu	Taç Yaprak Genişliği	Tür
0	5.1	3.5	1.4	0.2	<i>I. setosa</i>
1	4.9	3.0	1.4	0.2	<i>I. setosa</i>
2	7.0	3.2	4.7	1.4	<i>I. versicolor</i>
3	6.4	3.2	4.5	1.5	<i>I. versicolor</i>
4	6.3	3.3	6.0	2.5	<i>I. virginica</i>
5	5.8	2.7	5.1	1.9	<i>I. virginica</i>
...



Şekil 1: Zambak çiçeğinde çanak (sepal) ve taç (petal) yapraklar.

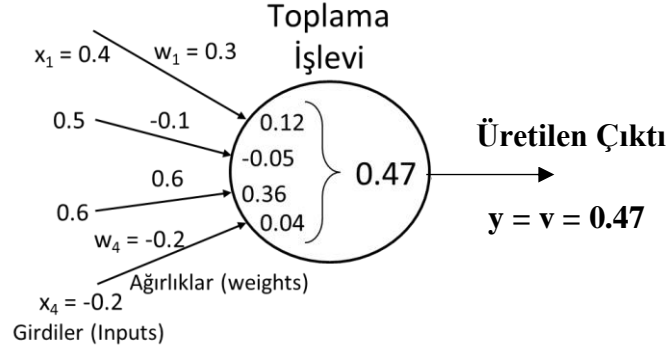
Aşağıdaki gibi bir yapay sinir ağı üreterek (sınıfı oluşturarak), verilen çiçek sınıflandırma probleminin çözümünde kullanınız. Java veya C# dillerinden herhangi birisini tercih edebilirsiniz. Hazır Makine öğrenmesi ve Veri Madenciliği kütüphanesinden yararlanmayınız.

ÇU : Çanak yaprak uzunluğu, ÇG: Çanak yaprak genişliği,
TU : Taç yaprak uzunluğu, TG : Taç yaprak genişliği,
N1 : Çıktı nöronu 1 N2 : Çıktı nöronu 2 N3 : Çıktı nöronu 3 olmak üzere,



Makine Öğrenmesi yöntemi olan ve derin öğrenme alanının temelini de oluşturan Yapay Sinir Ağları (Artificial Neural Networks - ANN) konusundaki en temel yapılar Yapay Sinir Hücreleridir (Artificial Neuron). ANN'ler sınıflandırma, kümeleme ve tahminleme gibi birçok problemin çözümünde kullanılırlar.

Yapay sinir hücresinin yapısı ve örnek bir hesaplama işlemi Şekil 1'de gösterilmektedir. Şekildeki nöronun 4 adet girdisi (x) ve 1 adet çıktısı (y) bulunmaktadır.



Şekil 1: Sinir Hücresi (Nöron) Modeli ve İşleyişi

Toplama İşlevi, girdilerle ağırlıkların çarpımları toplamının alınması şeklinde gerçekleştirilir:

$$v_k = \sum_{i=1}^n w_i x_i = w_1 x_1 + w_2 x_2 + \dots + w_n x_n = [w_1 \ w_2 \ \dots \ w_n] \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}$$

Gözetimli Öğrenmede (Supervised Learning), girdilerle beraber, olması gereken çıktı değerleri (target) verilir / sistem tarafından sağlanır. Ağ eğitilirken yukarıda bağlantısı verilen iris.data veri setinin kullanılması gerekmektedir. Eğitim setinde her biri

5.1, 3.5, 1.4, 0.2, Iris-setosa

formatında 150 adet veri (bitki örneği) bulunmaktadır. Satırdaki ilk 4 değer, ilgili çiçek örneğinin öznelikleri olup ağa girdi olarak verilecek, sondaki değer ise ilgili zambak bitkisinin türü olup çıktıda yer alacak, target yani beklenen değer olarak kullanılacaktır.

10 p a) **Bir Neuron (Sinir Hücresi) sınıfı** oluşturunuz. Girdiler ve ağırlıkları tutmak için **uygun veri yapılarını** tercih ediniz. Tüm ağırlıkları en başta [0, 1] arasında rastgele (random) pozitif değerlerden oluşturunuz. **Hesaplamaları ve gerekli işlemleri yapan metodu** veya metotları **yazınız** (nöron çıktısını hesaplayan).

10 p b) **Neural Network (Yapay Sinir Ağı) Sınıfı ve 3 nöron içerecek nesneyi oluşturunuz.** Beklenen değerler ile nöron çıktılarına bakarak **eğitimi yapan metodu yazınız: Eğitimden önce tüm girdi verilerini 10'a bölerek [0, 1] aralığına ölçeklendiriniz** (Dileyenler bölme işlemini yapmak yerine gerçek normalizasyon işlemi yapabilirler). Eğitim, basitleştirilmiş bir öğrenme kuralına göre yapılacaktır: Çıktısı I Setosa olan verilerde N1'in, Çıktısı I Versicolor olan verilerde N2'nin, Çıktısı I Virginica olan verilerde N3'ün beklenen değeri 1; Diğer çıktıların değeri 0 alınacaktır. İlgili veri için ağın çıktı değerleri (N1, N2 ve N3) hesaplanacak, **beklenen çıktı değeri ile ağın ürettiği çıktılardan en büyük değere sahip nöron aynı ise işlem yapılmayacak**; farklı ise: **Ağın ürettiği çıktıların en büyüğüne bağlı ağırlıkların (w) değerleri, λ öğrenme katsayısı ve x ise ilgili ağırlığa bağlı girdinin değeri olmak üzere,**

- $w = w - (\lambda * x)$ formülü ile azaltılacak

ve beklenen çıktıya bağlı ağırlıkların (w) değerleri ise

- $w = w + (\lambda * x)$ formülü ile artırılabilecektir.

10 p c) **Ağı λ (öğrenme katsayısı) = 0.01 olacak şekilde 50 epok boyunca eğitiniz.** Bir epok, tüm eğitim verilerinin (burada 150 adet) sisteme bir kere sıra ile verilerek ağırlıkların değiştirilmesi işlemidir. İşlem bittikten sonra girdi verilerini sadece sonuç elde edecek şekilde (ağırlıkları değiştirmeden) ağa verip çıktı değerlerini hesaplayınız. Bu aşamada ağın ürettiği çıktılardan en büyük olanı ile beklenen değeri 1 olan nöron aynı ise doğru bilinenlerin sayısını bir artırınız. **Doğruluk** değerini (doğru olarak sınıflandırılan **örnek (veri) sayısı / toplam örnek sayısıdır**) hesaplayıp yazdırınız. Elinizdeki 150 verinin 120 tanesi doğru olarak sınıflandırıldıysa doğruluk değeri (accuracy) = 120/150 = %80.

- d) Bu işlemi **20 ve 100 epok** için de baştan tekrarlayınız. $\lambda = 0.005$ ve 0.025 için (20, 50 ve 100 epok) için deneyleri tekrarlayınız. Epok sayıları (20,50,100) satırlarda, λ değerleri (0.005, 0.01 ve 0.025) sütunlarda olmak üzere doğruluk değerlerini 3x3'lük bir tabloya kaydederek raporda sununuz. c ve d maddelerindeki deneyleri 2 kere daha tekrarlayarak sonuçları 2 yeni tablo daha oluşturunuz. Başarı değerlerinin değişip değişmediğini gözlemleyiniz. Nedenini düşününüz.

PROJE TESLİMİNE İLİŞKİN BİLGİLER

- 1) Data Structures dersinin bu projesi için (ortak çalışma imkânınız bulunan kişilerle) **2 veya 3 kişilik çalışma grupları** oluşturunuz önerilir. Dileyen öğrenciler projeyi **tek** kişi olarak da hazırlayabilir. İkinci ve daha sonraki projeler ise sadece tek kişi olarak hazırlanacaktır, grup olarak teslim edilmeyeceklerdir.
- 2) **Rapor (20 p) hazırlanırken Egeders'te yer alan “DS_22_P1 Report Template.docx” şablonunu kullanınız. Sonda yer alan özdeğerlendirme tablosunu (10 p) doldurmayı ve rapor yazımı sonrası içindekiler bölümündeki sayfa numaralarını güncellemeyi unutmayınız.**
- 3) Projenizin işleyişini anlatan maksimum **5 dk'lık bir demo** videosu hazırlayınız. Video hazırlama için:
 1. Teams içerisinde **takvim** (calendar) sayfasına gidip “**şimdi toplantı yapın** (meet now)” seçeneği ile bir toplantı başlatın.
 2. Eğer grup çalışması yapıyorsanız diğer grup üyelerini de bu toplantıya dahil edin.
 3. Toplantı içerisinde kaydı başlatın.
 4. Ekran paylaşımı ile kodunuzu ve nasıl çalıştırıldığını, projede istenenleri dikkate alarak anlatın. (Grup çalışmasında her bir grup üyesi anlatımda görev almalıdır).
 5. Kaydı durdurun.
 - 6 **Toplantıdan ayrılmadan, sohbet** (chat) bölümünü açın. Videonuz burada oluşturulacaktır. Videonuza tıklayarak **indirin ve saklayın**.
 - Mikrofon gibi ekipman sıkıntısı yaşayanlar alternatif olarak demoyu ekran görüntüsü destekli yazılı doküman olarak da hazırlayabilirler. Demo dokümanı rapordan farklı olarak adım adım ekran görüntüleri ve ayrıntılı açıklama ile proje kodlarının nasıl çalıştırılacağını anlatmalıdır (raporda sadece ilgili maddenin konsol çıktısı istenmekteydi). Bir başka deyişle video demosunun yazılı bir versiyonudur.
- 4) **Rapor, demo ve her iki programın açıklama satırları destekli kaynak kodları (.cs uzantılı), çalışma grubundan bir öğrenci tarafından (dersin duyurularında belirtilen formata uygun olarak) son teslim tarihine kadar ilgili seçenekten sisteme **tek bir zip dosyası** halinde yüklenmelidir.** Oluşturulan “zip” veya “rar” dosyasının adı 8 haneli öğrenci numarası ve öğrenci ad ve soyadının alt çizgi (_) ile ayrılmasıyla oluşturulacaktır. Birden fazla öğrenci olması durumunda, proje üyeleri birbirinden yine alt çizgi kullanılarak ayrılacaktır.
Örneğin; “05146924_AliYılmaz_05130084_AyşeÖztürk.zip”
- 5) **MOSS** üzerinden belli ölçüde kod benzerlikleri görülen gruplara proje notu olarak 0 atanacaktır.