**TÜRKİYE TEKNOLOJİ TAKIMI VAKFI YAPAY ZEKA EĞİTİMİ LİSE DERS PLANI 6**

**DERS ETİKETLERİ**

Sınıf: **Lise**

Zorluk Düzeyi: **Zor**

Süre: **200 Dakika**

**Konular**

**1.** Yapay Sinir Ağı Nedir?

**2.** Perceptron ve Çok Katmanlı Ağlar

**KAZANIM**

● Öğrenciler yapay sinir ağı hakkında bilgi sahibi olurlar.

● Öğrenciler, Yapay Sinir Ağının makine öğrenme tekniklerinden farkını keşfederler.

●Python üzerinde yapay zeka örnekleri yaparlar.

**BECERİLER**

● Eleştirel Düşünme

● Metabiliş Beceriler

● İşbirliği ile çalışma

**DERS GENEL HATLARI**

**Harekete geç:** Bilgi yarışması ile önceki haftaların konuları pekiştirilir.

**Keşfet:** Yapay Sinir Ağ ve Çok Katmanlı Yapay Sinir Ağ konusu detaylı anlatılır.

**Üret/İlerlet:** Yapay sinir ağı etkinlik ve uygulamaları yapılır.

**Değerlendir:**Öğrenciler örümcek ağı etkinliği ile günün sonunda neler öğrendikleri hakkında değerlendirme yaparlar.

**1.Adım Harekete Geç:**

Eğitmen burada bilgi yarışması etkinliği yaparak derse giriş yapacaktır.

**Bilgi Yarışması**

Sınıf iki gruba ayrılır, her grup diğer gruba sormak üzere 10 soru hazırlar. Bunun için öğrencilere belli süre verilir. Süre sonunda bilgi yarışması tadında sorular başlar. En çok bilen ekip kazanır. Sorular ilk 5 haftadaki Yapay Zeka konuları ile alakalı olmalı. Eğer öğrenciler soru bulmakta zorlanırsa eğitmen her iki gruba 3’er soru verebilir.

**2.Adım Keşfet:**

**Yapay Sinir Ağları Sunumu**

Haftalık ders işleyiş tablosunda yer alan ayrı ayrı yapılacak olarak gözüken maddelerin hepsi aşağıda linki verilen “Yapay Sinir Ağları” sunumu linkinde toplanmıştır. Sunumu açarak ve sırasına tabii kalarak “Neden Yapay Sinir Ağı?”, “Beyin Nasıl Öğrenir?”, “Geleneksel MÖ Algoritmaları ile YSA Farkı”, “Perceptron”, “Çok Katmanlı Yapay Sinir Ağları” konularını, Yapay Sinir Ağları ile ilgili ekstra kavramları, YSA etkinliklerini, YSA videosunu bulabilirsiniz. Python ile YSA uygulamasını ise bu belgenin sonunda bulabilirsiniz. Ayrıca aşağıda Neden Yapay Sinir Ağ? başlığı ile başlayan 8 sayfalık YSA bilgilendirmesi de verilmiştir, dersten önce bilgilerinizi ister bu dökümandan ister sunumdan tazeleyebilirsiniz.

Yapay Sinir Ağları Sunumu Link: <https://drive.google.com/file/d/1AaSQNDf1D6YasvAf8NacgZvIQUYIyzYy/view>

**Neden Yapay Sinir Ağ?**

Öncelikle bakacağımız şey “Yapay” ve “Zeka” ne demektir.

Yapay demek, doğa eliyle yapılmamış olan, insanların kendisinin yaptıklarıdır.

Zeka, insanın düşünme, akıl yürütme, nesnel gerçekleri algılama, kavrama, yargılama ve sonuç çıkarma yeteneklerinin tümüdür.

Bu kelimeleri bir araya getirince ortaya çıkan Yapay Zeka, “Bir bilgisayarın veya bilgisayar kontrolündeki bir robotun çeşitli faaliyetleri zeki canlılara benzer şekilde yerine getirme kabiliyeti.” olarak klasik şekilde tanımlanır ama bizim tabii ki bu sıkıcı ve klişe tanımlamayı ayrıntılı ele almamız gerek.

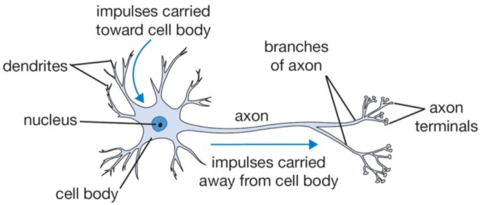
Yapay Zeka ise bilgisayarların insanlara ait düşünme, akıl yürütme, nesnel gerçekleri algılama, kavrama, yargılama ve sonuç çıkarma yeteneklerini kullanabilir hale gelmesi olarak tanımlanabilir ama aslında insanlar bildiğimiz en zeki canlılar ve yapay zeka da [insan zekasını](https://khosann.com/ne-kadar-hizli-dusunuyoruz-ozgur-irade-var-mi-diye-soran-profesor-dusunme-hizini-olctu/) taklit ediyor.

İstatistik ve matematiğe dayalı klasik yöntemler çözülmesi gereken tüm problemlere merhem olmuyor ve tabiri caizse demode kalıyordu. Ayrıca Yapay Zeka konusunun ulaşmak istediği nokta olan “İnsan ne yapıyorsa, makine de aynı insan gibi yapsın” yani “Makine mi İnsan mı olduğu belli olmayan bir makine” kavramları için bu yöntemler eksik kalıyordu.

Yapılması gereken insan nasıl öğreniyor ise makine de o şekilde öğrensin demekti. Bu yüzden insan beyninin çalışma yapısını bir algoritmaya uyarlamak gerekiyor. Bu da Yapay Sinir Ağları oldu.

Eğitilebilir, adaptif ve kendi kendine organize olup öğrenebilen ve değerlendirme yapabilen yapay sinir ağları ile insan beyninin öğrenme yapısı modellenmeye çalışılmaktadır. Aynı insanda olduğu gibi yapay sinir ağları vasıtasıyla makinelerin eğitilmesi, öğrenmesi ve karar vermesi amaçlanmaktadır.

İnsandaki bir sinir hücresinin (nöron) yapısı şu şekildedir:

Akson (Axon): Çıkış darbelerinin üretildiği elektriksel aktif gövdedir ve gövde üzerinde iletim tek yönlüdür. Sistem çıkışıdır.

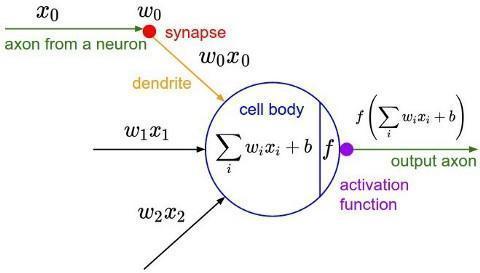
Dentritler (Dendrites): Diğer hücrelerden gelen işaretleri toplayan elektriksel anlamda pasif kollardır. Sistem girişidir.

Sinaps (Synapse): Hücrelerin aksonlarının diğer dentritlerle olan bağlantısını sağlar.

Miyelin Tabaka (Myelin Sheath): Yayılma hızına etki eden yalıtım malzemesidir.

Çekirdek (Nucleus): Akson boyunca işaretlerin periyodik olarak yeniden üretilmesini sağlar.

Aksonda taşınan işaret sinapslara kimyasal taşıyıcılar yardımıyla iletilmektedir. Stoplazma -85mV ile polarizedir. -40mV (Na+ içeri): uyarma (+) akıma yol açar. -90mV (K+ dışarı): bastırma (-) akıma yol açar. Yani belirli bir eşik gerilim değerinin üstünde iken hücre uyarılırken, diğer durumlarda hücre bastırılır. Σ Bu duruma göre çıkış işareti üretilmesine sinirsel hesaplama denir.

**Perceptron**

İnsandaki bir sinir hücresinin matematiksel modeli ise yandaki şekildeki gibi gösterilebilir:

Perseptron (Perceptron): Yapay sinir ağının en küçük parçası olarak bilinen perceptron, aşağıdaki gibi lineer bir fonksiyonla ifade edilmektedir.

y: x’in değerine bağlı olduğundan bağımlı değişkendir. Girdiye ait skoru verir.

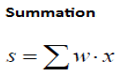
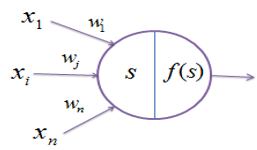
x: bağımsız değişken, girdi.

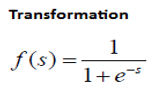
W: ağırlık parametresi

b: bias değeri

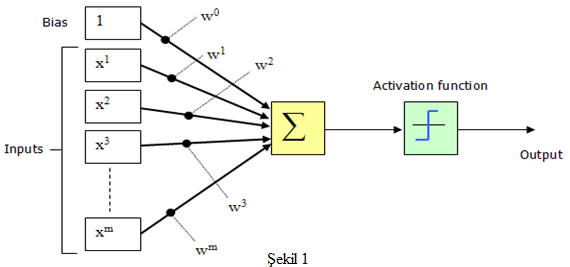
YSA (Yapay Sinir Ağları) ya da Derin Öğrenme modelinde yapılan temel işlem; modelin en iyi skoru vereceği W ve b parametrelerinin hesabını yapmaktır.

Perseptronu insan beynindeki tek bir hücre gibi düşünebiliriz.

Yapay Sinir Ağı, insan beynindeki sinir ağlarını modelleme üzerine kurulmuş olup 1960’lı yıllarda ilk olarak tasarlanmış ve kullanılmıştır. Bu ağ sadece tek bir nörondan (perseptron) oluşmaktadır ve bu nöron (perseptron) kendisine gelen veriyi belli işlemlere tabi tuttuktan sonra bir çıkış sonucu oluşturmaktadır.



**SINGLE PERCEPTRON MODEL**

Çok Katmanlı Perseptron’dan (Multi-layer Perceptron) önce Perseptron Model’den bahsetmek gerekiyor. Perseptron Model bir Yapay Sinir Ağları modelidir ve bugünkü Yapay Sinir Ağları için önemli bir temel oluşturmaktadır. Denetimli (supervised) bir öğrenme (training) algoritmasıdır. Yani ağa hem giriş hem de çıkış kümesi verilip ve öğrenme beklenir. Perseptron Modeli’nde en önemli faktör eşik değeridir. Bu değer kullanılarak güzel bir sınıflandırma yapılabilmektedir. Saptanacak olan eşik değeri probleme göre belirlenebilir. Modelde iterasyon sayısı arttırılarak öğrenme derecesi artırılabilir. Tek Katmanlı Algılayıcı’lar aşağıdaki şekilde modellenmiştir.

Modelde görülen x değerleri girişleri, w değerleri ise ağırlıkları ifade edilmektedir. BIAS değeri ise öğrenmeyi güçlendirmek için kullanılır. Ayrıca öğrenmeyi güçlendirirken yerel optimum değerlere takılmayı da önler. Bu modelin algoritma adımları şu şekildedir;

**Adım 1)** Tüm ağırlıklara başlangıç değerleri atanır. Öğrenme katsayısına küçük bir değer verilir. (Örn: Bias = 0)

**Adım 2)** Adım 3 ile 7 arasını belirli bir iterasyon sayısına ulaşılıncaya kadar tekrarla.

**Adım 3)** Her bir girdi için Adım 4 ile 6 arasını tekrarla.

**Adım 4)** Girdi değerlerini girdiden al.

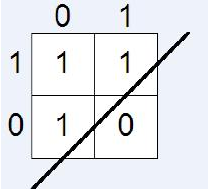
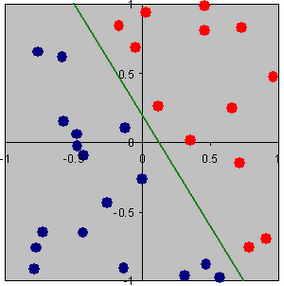
**Adım 5)** Perseptron’a gelen toplam sinyali hesapla. Aktivasyonu hesapla.

**Adım 6)** Eğer hesaplanan değer beklen değerden farklı ise, hesaplama yanlıştır. Ağırlıkları güncelle.

**Adım 7)** Döngü sonunu kontrol et.

Şekil-1’de görüldüğü üzere Perseptron Model tek katmanlıdır. Bu yüzden Single Perceptron Model de denmektedir. Sadece giriş ve çıkış katmanı bulunmaktadır. Net girdi hesaplanır. Girdi eşik değerin altındaysa 0, üstündeyse 1 olarak çıkış değeri belirlenir. Çıkış beklenen değerden farklı ise ağırlık güncellemesi yapılır. Şöyle ki; eğer çıkış 1 bekleniyorken, 0 olarak alınmışsa ağırlıklarda artırılmaya gidilir. Tersi durumda ise ağırlıklar azaltılır. Arttırım ve azaltım, belirlenen delta değeri ile yapılır. Tüm eğitim seti için doğru sonuçlar bulunana kadar algoritma devam ettirilir. Her biri için doğru sonuçlar bulunduğunda öğrenme tamamlanmış sayılır. Bu modelde elde edilen çıktı fonksiyonu doğrusaldır.

Perseptron Modeli ile ağa gösterilen örnekler iki sınıf arasında paylaştırılarak iki sınıfı birbirinden ayıran doğru bulunmaya çalışılır. Aşağıda örnekle gösterilmiştir.

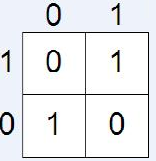


Şekil-2 Şekil-3

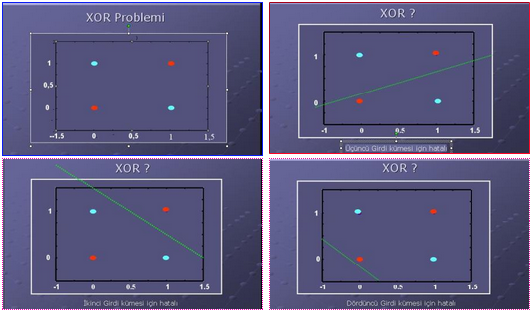
**SINGLE PERCEPTRON MODEL**

**DOĞRUSAL AYRILABİLİRLİK (Linear Seperability)**

Bir düzlemde iki sınıfa ait iki boyutlu örüntülerin bir kümesi doğrusal olarak ayrılabilmesine örnek olarak Şekil-2 gösterilebilir. Şekil-2’de görülen doğru düzlemi bir hatla ikiye ayırmıştır. Verilen düzlemdeki örneklerin hepsi iki gruptan birine dâhil olmuştur. Yani problem çözülebilir. Konuya örnek olarak VE(AND) ve VEYA(OR) fonksiyonları verilebilir. Bunları şekil üzerinden görelim. Yandaki şekilde veya fonksiyonu gösterilmiştir. Bilindiği üzere VEYA fonksiyonunun 1 çıkışı vermesi için giriş değerlerinden birinin 1 olması yeterli olur. Bu durumda VEYA fonksiyonunun çıktılarını düzlemde iki gruba ayırmak istersek Şekil-3’teki doğru bu işlemi gerçekleştirebilir. Yani VEYA fonksiyonu doğrusal ayrılabilirdir. VEYA fonksiyonunu yapay sinir ağları yöntemleriyle sisteme öğretmeye çalışırsak başarılı olabiliriz. Aynı durum VE fonksiyonu içinde geçerlidir. Tek farkı doğru farklı yerden geçip düzlemi ikiye ayıracaktır. Eğer bir problem doğrusal ayrılabilir ise o zaman Perseptron Öğrenimi ile örüntülerin bir kümesinden ağırlıklar elde edilebilir. Eğer problem doğrusal ayrılabilir değilse Single Perceptron Modeli ile çözüme ulaşamayız.

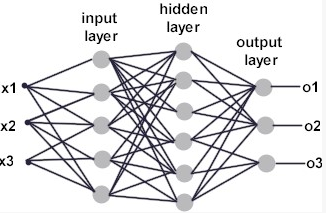
**XOR PROBLEMi**

XOR fonksiyonu doğrusal ayrılabilir değildir. Daha önce bahsettiğimiz VEYA ve VE fonksiyonunun çıktılarını düzlemde iki gruba ayırabiliyorduk. Fakat XOR fonksiyonunda bunu gerçekleştiremiyoruz. Düzlemdeki çıktıları tek bir hatla ikiye bölemiyoruz. (Şekil 5) En az iki doğru gerekiyor. XOR problemi Yapay Sinir Ağları’nın “Hello World”ü olarak bilinir. Perseptronlar XOR Problemi gibi doğrusal olarak sınıflandırılamayan problemleri çözümünde başarısızdır. XOR Problemi’ni çözmek için geriye yayılımlı çok katmanlı ağlardan faydalanılabilir.

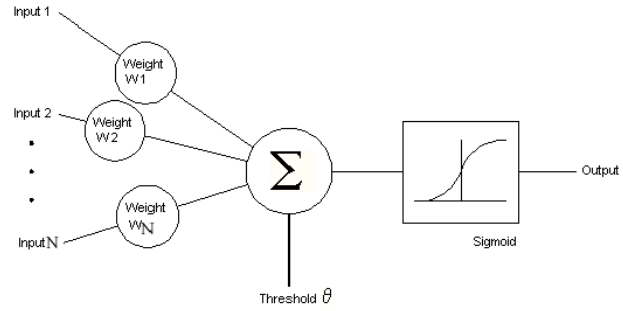
Şekil-4

Şekil-5

**ÇOK KATMANLI PERSEPTRON MODELİ (MULTI-LAYER PERCEPTRON MODEL - MLP)**

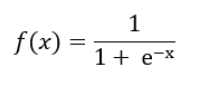
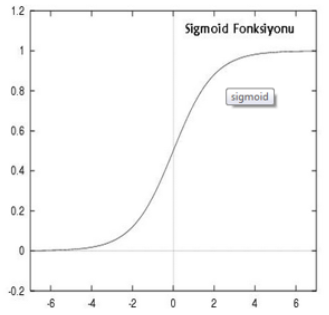
Çok Katmanlı Algılayıcılar (MLP) XOR Problemi’ni çözmek için yapılan çalışmalar sonucu ortaya çıkmıştır. Rumelhart ve arkadaşları tarafından geliştirilen bu modeli ‘Back Propogation Model’ ya da hatayı ağa yaydığı için ‘Hata Yayma Modeli’ de denmektedir. Delta Öğrenme Kuralı denilen bir öğrenme metodu kullanır. MLP özellikle sınıflandırma ve genelleme yapma durumlarında etkin çalışır. 

Birçok giriş için bir nöron yeterli olmayabilir. Paralel işlem yapan birden fazla nörona ihtiyaç duyulduğunda katman kavramı devreye girer. Görüldüğü üzere Single Perceptron Model’den farklı olarak arada gizli (hidden) katman bulunmaktadır. Giriş katmanı gelen verileri alarak ara katmana gönderir. Gelen bilgiler bir sonraki katmana aktarılırlar. Ara katman sayısı en az bir olmak üzere probleme göre değişir ve ihtiyaca göre ayarlanır. Her katmanın çıkışı bir sonraki katmanın girişi olmaktadır. Böylelikle çıkışa ulaşılmaktadır. Her işlem elemanı, yani nöron, bir sonraki katmanda bulunan bütün nöronlara bağlıdır. Ayrıca katmandaki nöron sayısı da probleme göre belirlenir. Çıkış katmanı önceki katmanlardan gelen verileri işleyerek ağın çıkışını belirler. Sistemin çıkış sayısı çıkış katmanında bulunan eleman sayısına eşittir. Single Perceptron Modeli incelerken bahsettiğimiz nöron yapısı burada aynen geçerlidir.

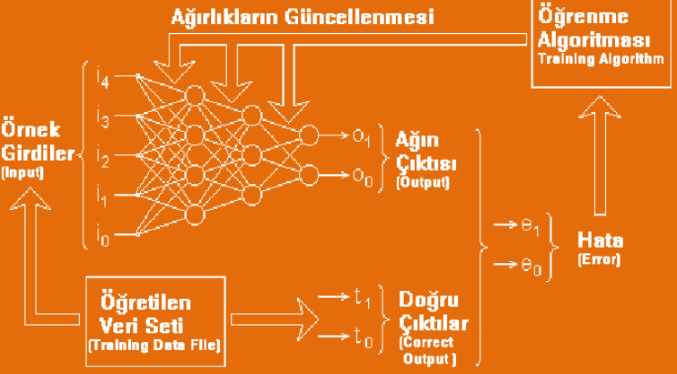
Modelde aktivasyon fonksiyonu olarak herhangi bir matematiksel fonksiyon kullanılabilir. Ancak Sigmoid, tang, lineer, threshold ve hard limiter fonksiyonları en çok kullanılan fonksiyonlardır.

Çok katmanlı ağlardaki hücreler yukarıdaki gibidir. Aktivasyon fonksiyonu olarak yandaki modelde sigmoid fonksiyonu seçilmiştir. Çok katmanlı ağda öğrenme Delta Öğrenme Kuralı tabanlıdır. Ağın öğrenebilmesi için örnek giriş ve çıkışlardan oluşan eğitim seti şarttır. Geri Yayılımlı Yapay Sinir Ağları’nda öğrenme işlemi bir anlamda örnek setindeki giriş değerleriyle, çıkış değerlerini eşleştiren fonksiyonu bulma işlemidir. Sistemin öğrenme metodu genel olarak iki aşamadan oluşur. Birinci kısım ileri doğru hesaplamadır. İkinci kısım ise geri doğru hesaplamadır (back propogation).

İleri doğru hesaplama aşamasında sisteme verilen girdi ara katmanlardan geçerek çıkışa ulaşır. Her işlem elemanına gelen girdiler toplanılarak net girdi hesaplanır. Bu net girdi aktivasyon fonksiyonundan geçirilerek mevcut işlem elemanının çıktısı bulunur. Ve bu çıktı değeri bir sonraki katmanda bulunan işlem elemanlarına gönderilir. Bu işlemler tekrar edilerek en son çıktı katmanından çıktılar elde edilir. En çok kullanılan aktivasyon fonksiyonu olan sigmoid fonksiyonu şekildedir.



Ağdan çıktı alınmasıyla öğrenmenin ilk aşaması bitirilmiş olur. İkinci aşama hatanın dağıtılması olacaktır. Beklenen çıktı değeri ile elde ettiğimiz birbirinden farklı ise hata vardır. Geriye doğru hesaplama aşamasında hata ağırlık değerlerine dağıtılarak her iterasyonda azaltılması beklenir. Sisteme başlangıçta random olarak verilen ağırlık değerleri, hataların ağırlıklara dağıtılmasıyla her iterasyonda güncellenmiş olur.

Genelleştirilmiş Delta Öğrenme Kuralı’nın yapısı genel olarak aşağıdaki gibidir.

Multi-layer Perceptron(MLP) yani Çok Katmanlı Algılayıcılar, Yapay Sinir Ağları’na olan ilgiyi hızlı bir şekilde artırmıştır. MLP ile birlikte YSA tarihinde yeni bir dönem başlamıştır. Geniş kullanım alanına sahiptir. Örnek verecek olursak; Otomotiv alanında yol izleme, rehberlik vs. gibi konularda kullanılmaktadır. Bankacılıkta kredi kartı suçu tespiti ve kredi uygulamalarında kullanılmaktadır. Uzay sanayinde uçuş simülasyonu ve otomatik pilot uygulamalarında kullanılır. Finans sektöründe ise döviz kuru tahminlerinde kullanılır.

**3.4. Adım:Üret/İlerlet(Dakika):**

**Python ile YSA Uygulaması**

Aşağıdaki linkten ulaşacağınız klasörde uygulama ile ilgili tüm detayları bulabilirsiniz.

[**https://drive.google.com/drive/u/0/folders/1avZfzFFXQc2sDjU9AVqjj-a-MGF-dvOG**](https://drive.google.com/drive/u/0/folders/1avZfzFFXQc2sDjU9AVqjj-a-MGF-dvOG)

**5.Adım Değerlendirme:**

Öğrenciler örümcek ağı etkinliği ile günün sonunda neler öğrendikleri hakkında düşünürler.

**ÖRÜMCEK AĞI ETKİNLİĞİ**



-Bu etkinlikte öğrenciler bir halka oluşturur.

-Bir öğrenci ip yumağını eline alır.İpin ucunu tutarak yumağı arkadaşlarından birine atar.

-Yumağı attığı arkadaşına ders içeriği ile ilgili bir soru sorar.

-Arkadaşı cevapladıktan sonra oda yumağı bir arkadaşına atarak ona soru sorar.Bu şekilde yumak tüm halkayı dolaşır.

-Herkes soru sorduktan sonra çıkan şekil üzerinden sınıf ruhu hakkında konuşulur.Bir bütünün parçası oldukları ve içlerinden herhangi birinin bırakması halinde tüm düzenin bozulacağı söylenir.

-Bu etkinlik ders içeriğine göre uygulanır.