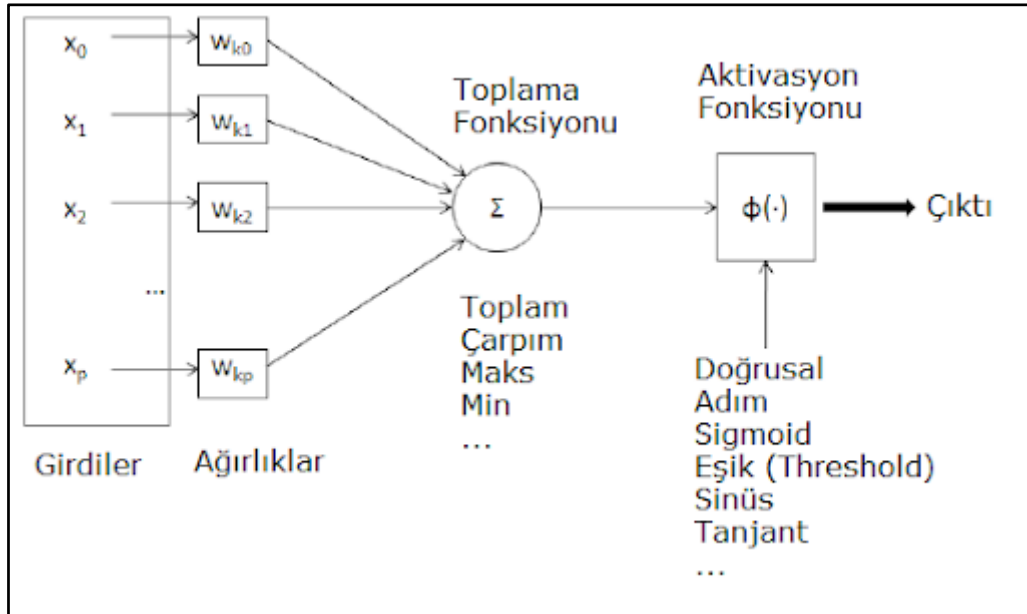


#### Uygulama 4. Yapay Sinir Ağlarının Geliştirilmesi ve Kullanılması

Canlıların davranışlarını inceleyip, matematiksel olarak modelleyip, benzer yapay modellerin üretilmesine **sibernetik** denir. 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. Şekil 1’de biyolojik sinir sistemi ile yapay sinir sistemi karşılaştırılmıştır.

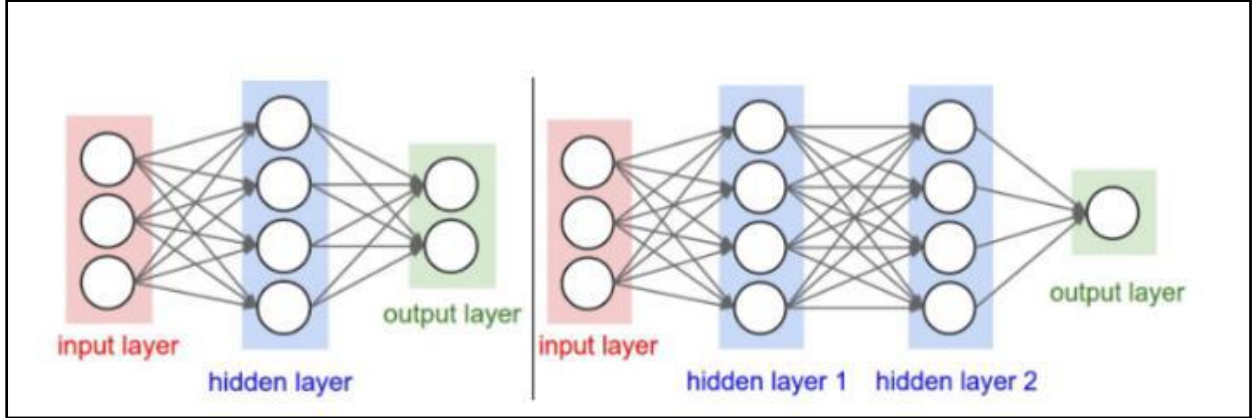
Biyolojik Sinir Sistemi	Yapay Sinir Sistemi
Nöron	İşlemci eleman
Dentrit	Toplama fonksiyonu
Hücre gövdesi	Transfer fonksiyonu
Aksonlar	Yapay nöron çıkışı
Sinapslar	Ağırlıklar

Şekil 1. Sinir sisteminin biyoloji ve yapay ağlardaki gösterimi.



Şekil 2: Yapay sinir ağının genel yapısı

Yapay sinir ağı tek katmanlı ve çok katmanlı olmak üzere ayrı ayrı ifade edilebilirler. Şekil 3'te tek ve çok katmanlı yapay sinir ağına örnekler verilmiştir.



**Şekil 3.** Tek ve çok katmanlı yapay sinir ağı.

**Girdi katmanı:** Bu katman, girdileri almaktan ve diğer katmanlara iletmekten başka hiçbir şey yapmayan nöronlardan oluşur. Giriş katmanındaki katman sayısı, veri kümesindeki niteliklere veya özelliklere eşit olmalıdır.

**Gizli Katman:** Giriş ve çıkış katmanı arasında, modelin türüne göre gizli katmanlar olacaktır. Gizli katmanlar çok sayıda nöron içerir. Gizli katmandaki nöronlar, girdilere ve bunları geçmeden önce dönüşümleri uygular. Ağ eğitildiği için ağırlıklar güncellenir, daha öngörücüdür.

**Toplama Fonksiyonu (Birleştirme Fonksiyonu):** Toplama fonksiyonu bir yapay sinir hücresine ağırlıklarla çarpılarak gelen girdileri toplayarak o hücrenin net girdisini hesaplayan bir fonksiyondur. Tablo 1'de kullanılan bazı toplama fonksiyonları verilmiştir.

**Tablo 1.** Bazı toplama fonksiyonları

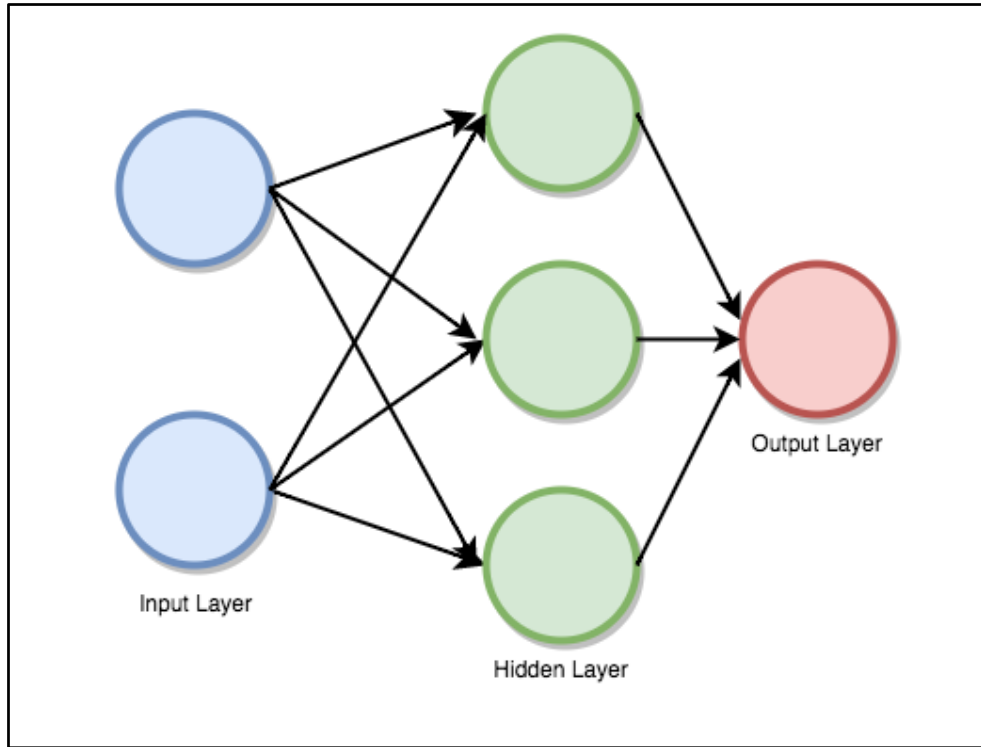
Toplam $Net = \sum_{i=1}^N X_i * W_i$	Ağırlık değerleri girdiler ile çarpılır ve bulunan değerler birbirleriyle toplanarak Net girdi hesaplanır.
Çarpım $Net = \prod_{i=1}^N X_i * W_i$	Ağırlık değerleri girdiler ile çarpılır ve daha sonra bulunan değerler birbirleriyle çarpılarak Net Girdi Hesaplanır.
Maksimum $Net = \text{Max}(X_i * W_i)$	n adet girdi içinden ağırlıklar girdilerle çarpıldıktan sonra içlerinden en büyüğü Net girdi olarak kabul edilir.
Minimum $Net = \text{Min}(X_i * W_i)$	n adet girdi içinden ağırlıklar girdilerle çarpıldıktan sonra içlerinden en küçüğü Net girdi olarak kabul edilir.
Çoğunluk $Net = \sum_{i=1}^N \text{Sgn}(X_i * W_i)$	n adet girdi içinden girdilerle ağırlıklar çarpıldıktan sonra pozitif ile negatif olanların sayısı bulunur. Büyük olan sayı hücrenin net girdisi olarak kabul edilir.
Kümülatif Toplam $Net = \text{Net}(\text{eski}) + \sum_{i=1}^N X_i * W_i$	Hücreye gelen bilgiler ağırlıklı olarak toplanır. Daha önce hücreye gelen bilgilere yeni hesaplanan girdi değerleri eklenerek hücrenin net girdisi hesaplanır.

**Aktivasyon Fonksiyonu:** Aktivasyon fonksiyonu, toplanan ağırlıklı girdinin nöronun çıkışına eşlenmesidir. Bir aktivasyon/transfer fonksiyonu olarak adlandırılır, çünkü nöronun aktive edildiği ve çıkış sinyalinin gücünün başlangıcını yönetir. Matematiksel olarak, en çok kullanılanları ReLu, tanh, Softmax gibi birçok aktivasyon fonksiyonumuz vardır.

**Çıktı Katmanı:** Çıktı katmanı tahmin edilen özelliktir, temel olarak oluşturduğunuz modelin türüne bağlıdır.

### İleri Beslemeli Derin Ağlar

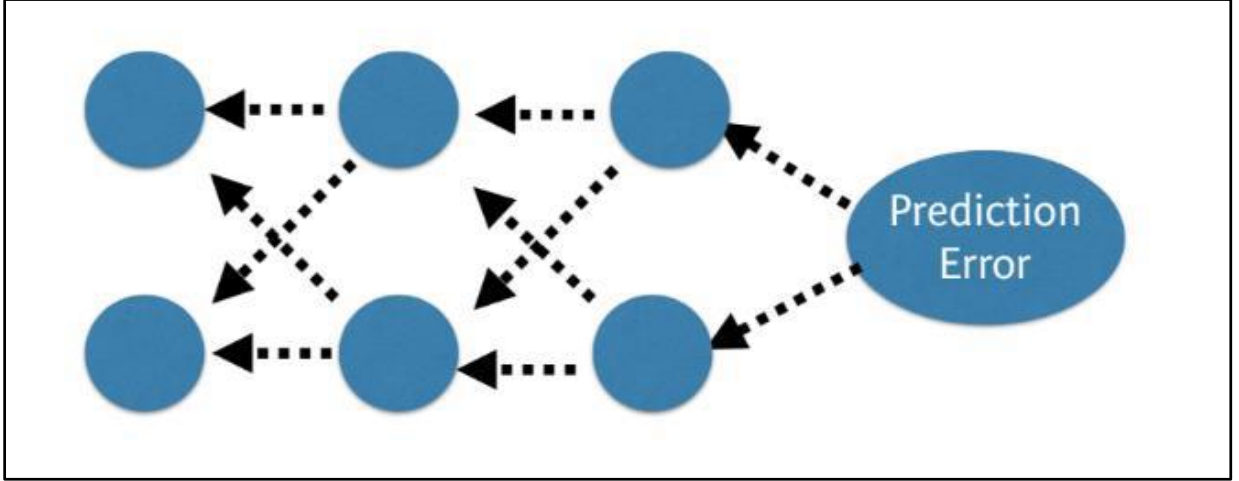
İleri beslemeli denetimli sinir ağları, ilk ve en başarılı öğrenme algoritmaları arasındaydı. Bunlar aynı zamanda derin ağlar, Çok Katmanlı Perceptron (MLP) veya basitçe sinir ağları olarak adlandırılır ve tek bir gizli katmana sahip vanilya mimarisi resmedilir. Her bir nöron, bazı ağırlıklarla diğer nöronla ilişkilidir. Ağ, bir çıkış değeri üretecek şekilde yukarı doğru harekete geçiren nöronları işler. Bu, ağ üzerinde bir ileri geçiş olarak adlandırılır. Şekil 4'te ileri beslemeli yapay sinir ağına örnek verilmiştir.



**Şekil 4.** İleri beslemeli yapay sinir ağı modeli

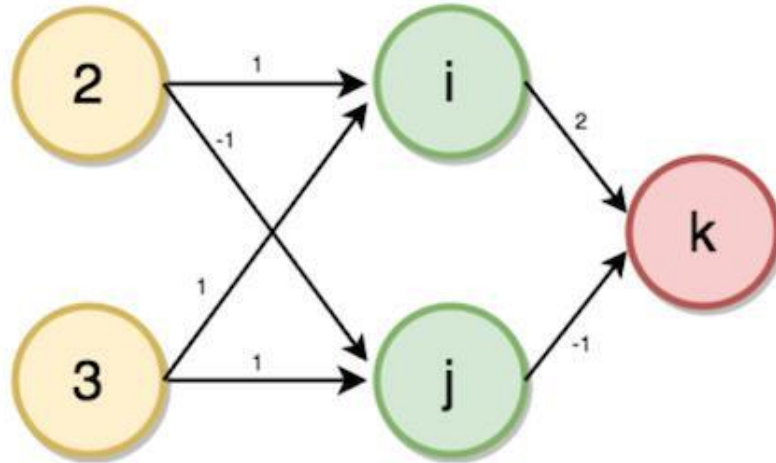
## Geri Yayılma Yapay Sinir Ağı

Ağın tahmini değeri beklenen çıktıyla karşılaştırılır ve bir fonksiyon kullanılarak bir hata hesaplanır. Bu hata daha sonra tüm ağda, bir kerede bir katman halinde geri yayılır ve ağırlıklar, hataya katkıda bulundukları değere göre güncellenir. Bu akıllı matematik bitine geri yayılım (back-propagation) algoritması denir. İşlem, eğitim verilerindeki tüm örnekler için tekrarlanır. Şekil 5'te örnek bir geriye yayılım YSA verilmiştir.



Şekil 5. Geriye yayılım yapay sinir ağı modeli.

Örnek uygulama



$$i = 2 * 1 + 3 * 1 = 5$$

$$j = 2 * -1 + 3 * 1 = 1$$

$$k = 5 * 2 + 1 * -1 = 9$$

#### Uygulama 4. Değerlendirme Soruları

- 1) Verilen kodu inceleyerek kendi modelinizi oluşturunuz **(10p)**.
- 2) Verilen kodu inceleyerek çapraz doğrulama işlemi yapınız ve buna göre başarıyı değerlendiriniz **(30p)**.
- 3) Verisetinde (telefon\_fiyat\_değişimi) toplam 20 adet özellik bulunmaktadır. Bu verisetinden "blue", "fc", "int\_memory", "ram" ve "wifi" değerlerini çıkarıp, sınıflandırma işlemi tekrar yapınız **(10p)**.
- 4) Diyabet verisetini kullanarak bir YSA modeli oluşturunuz. Bu YSA modeline, eğitim ve doğrulama işlemlerinin başarımları ve kayıplarını belirleyiniz. Bu değerleri bir grafik üzerinde gösteriniz **(20p)**.
- 5) Soru 4'te oluşturduğunuz modelin ROC eğrisini çiziniz **(30p)**.