

CIFAR 10 ÖZELLİKLERİ

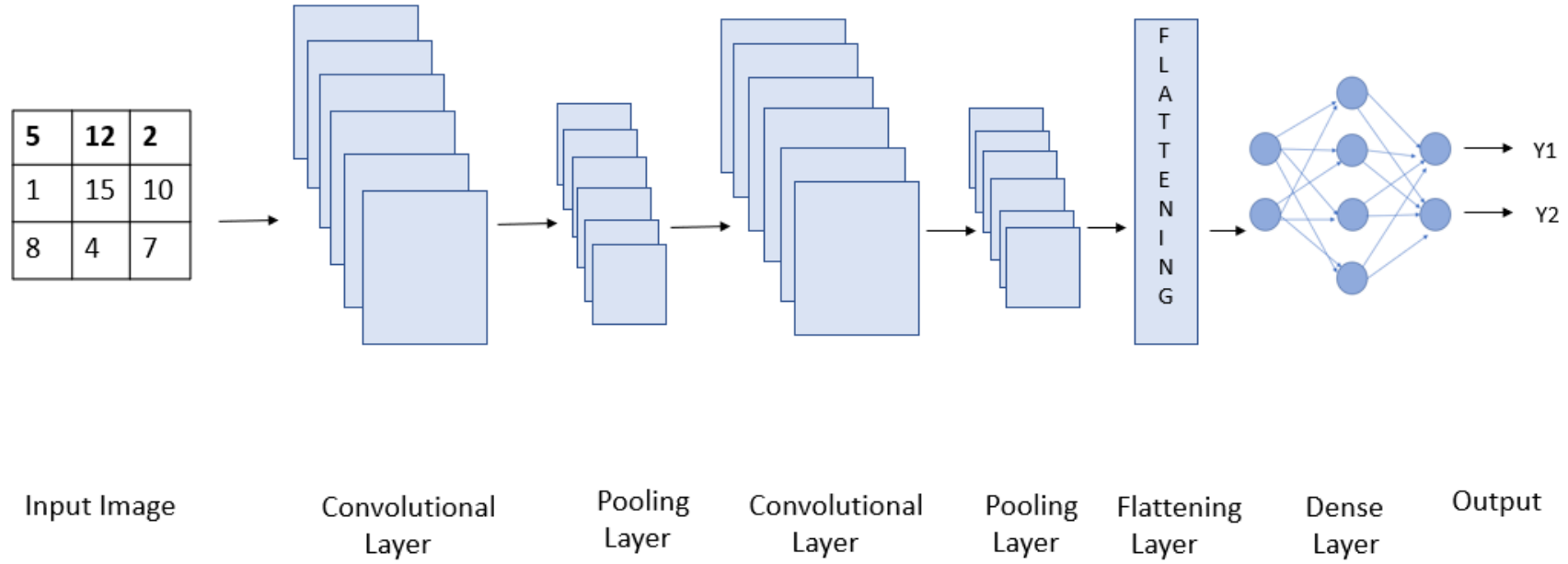
- CIFAR-10 veri seti, sınıf başına 6000 görüntü olmak üzere 10 sınıfta 60000 32x32 renkli görüntüden oluşur. 50000 eğitim görseli ve 10000 test görseli bulunmaktadır.



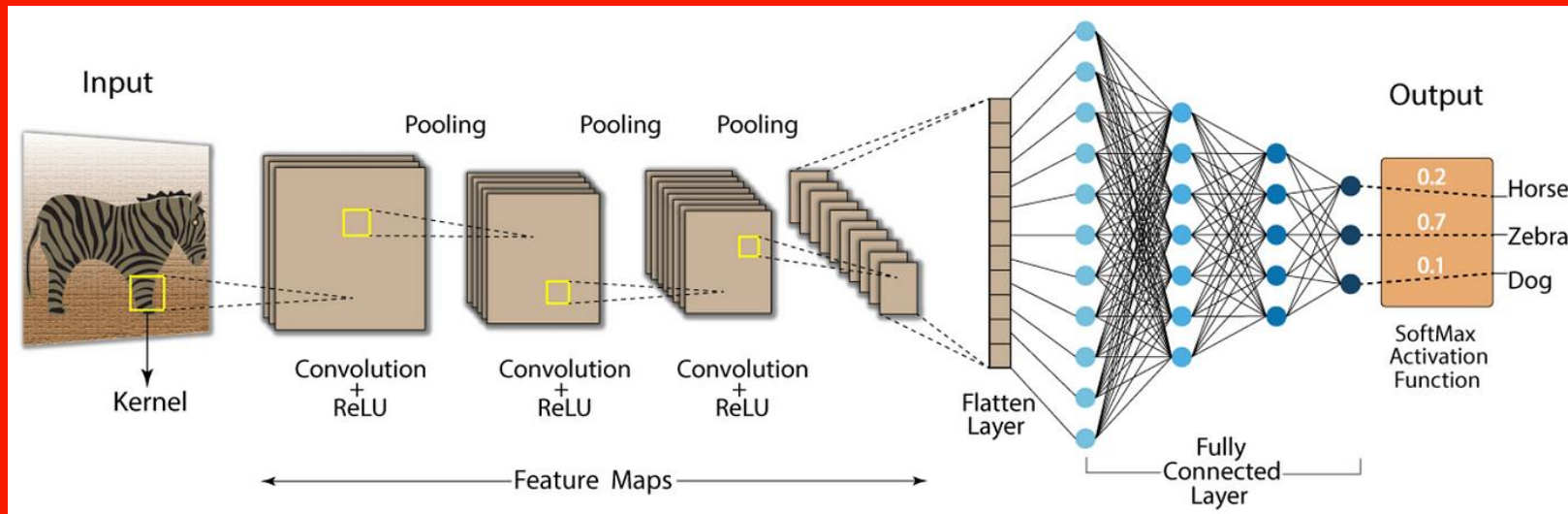
CIFAR10 VERİ SETİNDEN SINIFLANDIRILMA ÖRNEKLERİ



KULLANILAN YÖNTEM VE ARAÇLAR



EVRIŞİMLİ SİNİR AĞI



Kullanım Alanları:

- **Optimizasyon Problemleri:** Karmaşık fonksiyonların veya sistemlerin optimizasyonu.
- **Makine Öğrenimi:** Eğitim süreçlerinin iyileştirilmesi ve hiperparametrelerin otomatik ayarlanması.
- **Oyun ve Robotik:** Dinamik ve öngörülemeyen ortamlarda adaptasyon ve öğrenme.

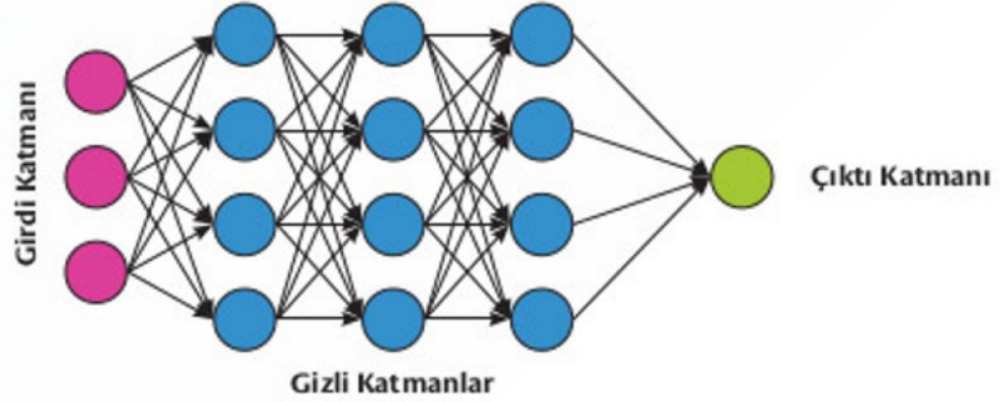
Avantajları:

- **Adaptif Öğrenme:** Karmaşık ve değişken çevrelerde adaptasyon yeteneği.
- **Küresel Optimizasyon:** Yerel minimumlara takılma olasılığını azaltır, daha geniş arama alanını tarar.
- **Otomasyon:** Manuel ayar ve ince ayar gereksinimini azaltır.

Dezavantajları:

- **Hesaplama Maliyeti:** Evrimsel algoritmalar, yoğun hesaplama gerektirir.
- **Zaman:** Yüksek performanslı bir çözüm elde etmek uzun sürebilir.
- Evrimsel sinir ağları, klasik sinir ağlarının sınırlarını aşmak ve daha karmaşık problem alanlarında daha iyi çözümler üretmek için etkili bir yöntemdir. Bu teknik, özellikle geniş arama alanlarına ve karmaşık optimizasyon problemlerine sahip uygulamalarda faydalı olabilir.

YAPAY SİNİR AĞI (ANN)



Derin sinir ağlarının
bir örneği

1.Nöronlar (Düğümleler): Beyindeki nöronlara benzer şekilde, bilgiyi işleyen ve ileten temel birimlerdir. Yapay sinir ağlarında bu nöronlar, giriş bilgilerini alır, işler ve çıktı üretir.

2.Katmanlar: Sinir ağları genellikle üç tür katmandan oluşur:

- 1. Giriş Katmanı (Input Layer):** Ağa gelen ham veriyi alır.
- 2. Gizli Katmanlar (Hidden Layers):** Giriş katmanından gelen bilgiyi işler ve çıktı katmanına iletir. Bu katmanlar birden fazla olabilir ve ağın derinliğini belirler.
- 3. Çıkış Katmanı (Output Layer):** İşlenmiş bilgiyi sonuca dönüştürür ve kullanıcıya veya diğer sistemlere iletir.

3.Ağırlıklar (Weights): Nöronlar arasındaki bağlantıların kuvvetini temsil eder. Ağırlıklar, ağın öğrenme sürecinde ayarlanarak modelin performansını optimize eder.

4.Aktivasyon Fonksiyonları: Nöronun çıktısını hesaplamak için kullanılan matematiksel fonksiyonlardır. Sigmoid, ReLU (Rectified Linear Unit), ve tanh gibi çeşitli aktivasyon fonksiyonları vardır.

5.Eğitim ve Öğrenme: Sinir ağları, eğitim verileri kullanılarak öğrenir. Öğrenme süreci, hata fonksiyonunu minimize etmek için ağırlıkların ayarlanmasını içerir. Yaygın kullanılan bir öğrenme algoritması olan geri yayılım (backpropagation), hatayı hesaplar ve ağırlıkları günceller.

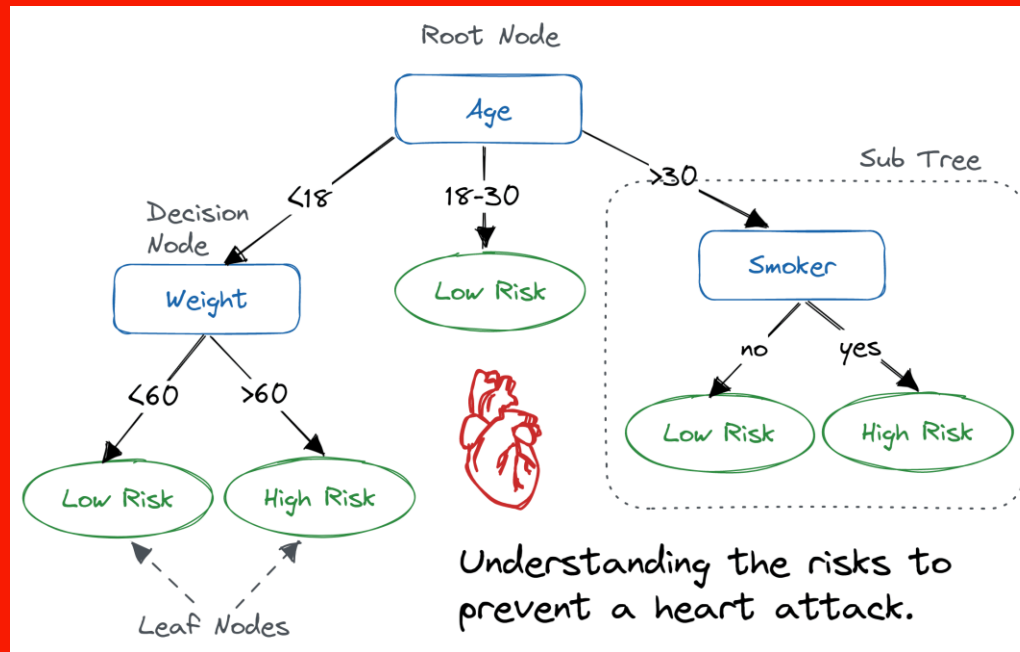
Yapay Sinir Ağlarının Çeşitleri:

- **İleri Beslemeli Sinir Ağları (Feedforward Neural Networks):** Bilgi, giriş katmanından çıkış katmanına doğru tek yönlü akar.
- **Geri Beslemeli Sinir Ağları (Recurrent Neural Networks, RNN):** Önceki çıktıları hatırlayarak ve kullanarak, sıralı veri ve zaman serileri üzerinde çalışır.
- **Evrimsel Sinir Ağları:** Evrimsel algoritmalar kullanarak sinir ağlarını optimize eder.
- **Konvolüsyonel Sinir Ağları (Convolutional Neural Networks, CNN):** Özellikle görüntü işleme ve tanıma görevlerinde kullanılır, yerel özellikleri yakalamak için konvolüsyonel katmanlar kullanır.

Kullanım Alanları:

- **Görüntü ve Ses Tanıma:** Görüntü sınıflandırma, yüz tanıma, sesli komut tanıma.
- **Doğal Dil İşleme (NLP):** Makine çevirisi, duygu analizi, metin oluşturma.
- **Tahmin ve Sınıflandırma:** Finansal tahminler, hastalık teşhisi, spam tespiti.
- **Oyun ve Robotik:** Otonom araçlar, robotik kontroller, oyun stratejileri.

KARAR AĞACI SINIFLANDIRMASI



1.Düğüm (Nodes):

1. **Kök Düğüm (Root Node):** Ağacın en üstünde bulunan ve veri kümesinin tamamını temsil eden düğüm. Bu düğüm, ilk bölünme kriterini belirler.
2. **Karar Düğümleri (Decision Nodes):** Verinin belirli özelliklerine göre bölündüğü düğümler.
3. **Yaprak Düğümleri (Leaf Nodes):** Nihai sınıfları temsil eden düğümler. Burada daha fazla bölünme yapılmaz ve sınıflandırma tamamlanır.

2.Dallar (Branches): Karar düğümleri arasındaki bağlantıları temsil eder ve verinin belirli özellik değerlerine göre nasıl bölündüğünü gösterir.

Çalışma Prensipleri:

1.Bölme (Splitting): Veri kümesi, belirli bir özellik ve bu özelliğin değerine göre alt kümelere bölünür. Bu işlem, her düğümde tekrarlanır.

2.Kriterler: Bölünme kriterleri genellikle bilgi kazancı (information gain), Gini indeks veya varyans azaltma gibi ölçütler kullanılarak belirlenir.

3.Büyüme: Ağaç, tüm veriler sınıflandırılana veya belirli bir durdurma kriterine ulaşılan kadar büyür.

4.Budama (Pruning): Aşırı öğrenmeyi (overfitting) önlemek için gereksiz veya az bilgi sağlayan dallar kaldırılabilir.

Avantajları:

- Kolay Anlaşılabilirlik:** Ağaç yapısı sezgisel ve görselleştirilmesi kolaydır, bu da sonuçların yorumlanmasını basitleştirir.
- Az Veri Ön İşleme:** Genellikle veri normalleştirme veya ölçeklendirme gibi ön işlemler gerektirmez.
- Hızlı ve Verimli:** Büyük veri kümeleri üzerinde hızlıca çalışabilir ve hesaplama açısından verimlidir.

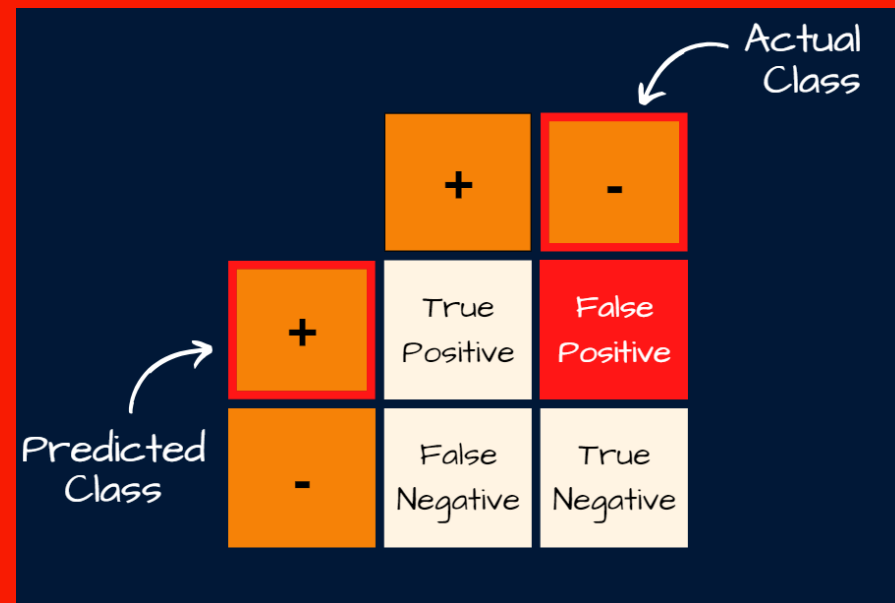
Dezavantajları:

- Aşırı Öğrenme (Overfitting):** Çok derin ağaçlar, eğitim verisine fazla uyum sağlayarak genel performansı düşürebilir.
- Dengesiz Veriler:** Dengesiz veri kümelerinde dengesiz sınıflandırmalar yapabilir.

Kullanım Alanları:

- Tıp:** Hastalık teşhisi ve tedavi önerileri.
- Finans:** Kredi risk değerlendirmesi, müşteri segmentasyonu.
- Pazarlama:** Müşteri davranış analizi ve hedefleme.
- İmalat:** Kalite kontrol ve hata tespiti.

confusion matrix



A confusion matrix diagram. The vertical axis is labeled 'Predicted Class' with an arrow pointing to the left column. The horizontal axis is labeled 'Actual Class' with an arrow pointing to the top row. The matrix is a 2x2 grid of colored squares. The top-left square is orange with a '+' sign. The top-right square is orange with a '-' sign. The bottom-left square is orange with a '+' sign. The bottom-right square is orange with a '-' sign. The cells are labeled as follows: True Positive (top-left), False Positive (top-right), False Negative (bottom-left), and True Negative (bottom-right).

	Actual Class	+	-
Predicted Class	+	True Positive	False Positive
	-	False Negative	True Negative

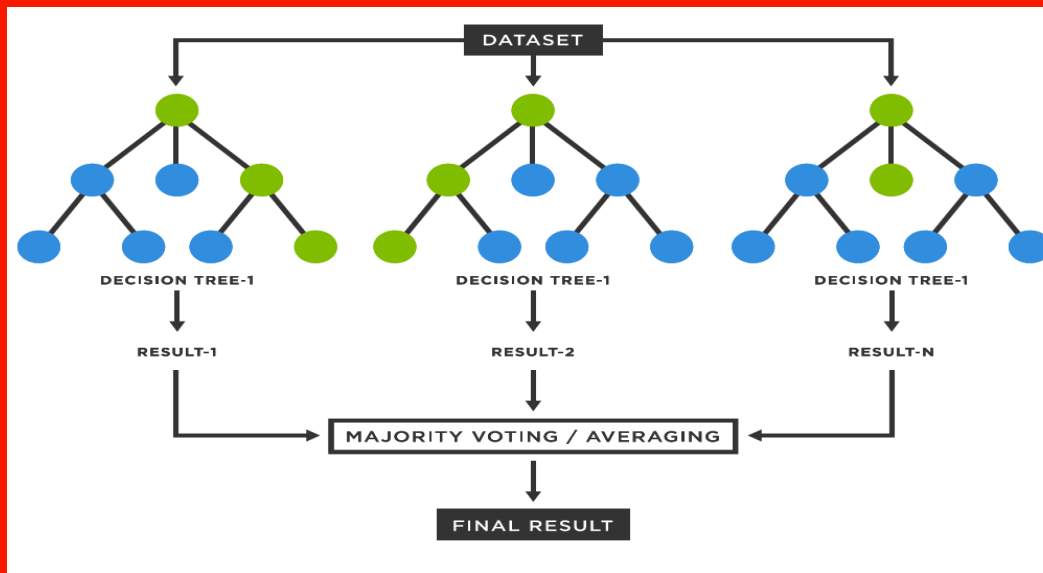
1.True Positive (TP): Modelin doğru şekilde pozitif olarak tahmin ettiği örnekler.

2.True Negative (TN): Modelin doğru şekilde negatif olarak tahmin ettiği örnekler.

3.False Positive (FP): Modelin pozitif olarak tahmin ettiği ancak gerçekte negatif olan örnekler (Type I error).

4.False Negative (FN): Modelin negatif olarak tahmin ettiği ancak gerçekte pozitif olan örnekler (Type II error).

RANDOM FOREST



1.Topluluk Öğrenme (Ensemble Learning): Random Forest, birçok karar ağacının birlikte çalışması prensibine dayanır. Her bir ağaç, bağımsız olarak eğitilir ve sınıflandırma veya regresyon problemi için oy kullanır.

2.Çeşitlendirme: Farklı ağaçlar, eğitim verisinin farklı alt kümeleri ve rastgele seçilen öznitelikler kullanılarak oluşturulur. Bu çeşitlilik, modelin aşırı öğrenmesini (overfitting) önler ve genelleme kabiliyetini artırır.

Çalışma Prensibi:

1.Bootstrap Örnekleme: Eğitim verisi, rastgele örnekleme (bootstrap) yöntemiyle birçok alt kümeye bölünür. Bu alt kümeler, her bir karar ağacının eğitilmesi için kullanılır.

2.Rastgele Öznitelik Seçimi: Her ağaç için, her düğümde rastgele bir öznitelik alt kümesi seçilir ve en iyi bölünme bu alt küme üzerinden yapılır.

3.Ağaçların Eğitimi: Her ağaç, kendi bootstrap örneği ve öznitelik alt kümesi ile bağımsız olarak eğitilir.

4.Tahmin: Sınıflandırma problemlerinde, her ağacın tahmin ettiği sınıf için oy kullanılır ve en çok oyu alan sınıf nihai tahmin olarak seçilir. Regresyon problemlerinde ise, ağaçların tahminlerinin ortalaması alınır.

Avantajları:

- Yüksek Doğruluk:** Topluluk öğrenme sayesinde, tek bir karar ağacından daha doğru ve genelleştirilebilir sonuçlar elde edilir.
- Aşırı Öğrenmeye Dayanıklılık:** Rastgele öznitelik seçimi ve bootstrap örnekleme, modelin aşırı öğrenmesini önler.
- Öznitelik Önem Derecesi:** Random Forest, özniteliklerin önem derecesini belirleyebilir, bu da model açıklanabilirliğini artırır ve önemli özniteliklerin seçilmesine yardımcı olur.
- Esneklik:** Hem sınıflandırma hem de regresyon problemlerinde etkili bir şekilde kullanılabilir.

Dezavantajları:

- Hesaplama Maliyeti:** Çok sayıda ağaç oluşturma ve tahmin süreci, hesaplama açısından maliyetli olabilir.
- Yorumlama Zorluğu:** Tek bir karar ağacına kıyasla, modelin açıklanabilirliği daha düşüktür.

Kullanım Alanları:

- Finans:** Kredi risk değerlendirmesi, dolandırıcılık tespiti.
- Sağlık:** Hastalık teşhisi, hasta sonuç tahminleri.
- Pazarlama:** Müşteri segmentasyonu, müşteri davranış tahminleri.
- İmalat:** Kalite kontrol, arıza tahminleri.

Random Forest, güçlü, esnek ve yüksek doğruluk sağlayan bir makine öğrenimi algoritmasıdır. Hem sınıflandırma hem de regresyon problemlerinde etkili olup, özellikle büyük ve karmaşık veri kümelerinde üstün performans sergiler.