

Derin Öğrenme Deep Learning

Hazırlayan: M. Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Bu dersin sunumları, "Simon Haykin, Neural Networks and Learning Machines, Prentice Hall, 2016." kitabı kullanılarak hazırlanmıştır.

İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ Hyperparameters

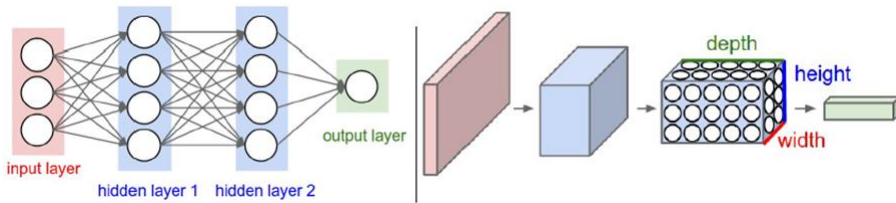
Convolutional neural networks

- ▶ **Convolutional neural network (CNN)**, yapay sinir ağlarının özel bir türüdür.
- ▶ CNN'ler özellikle görüntü problemlerinde yaygın kullanılan derin öğrenme mimarisidir.
- ▶ Bir **CNN**, klasik neural networks yapısına benzer şekilde **neuron'lardan oluşur** ve **öğrenmek için ağırlıklar** ile **bias değerine sahiptir**.
- ▶ Her neuron **girişleri alır, birleştirir** ve genellikle **non-lineer bir fonksiyon ile çıkış üretir**.
- ▶ CNN uygulamaları girişleri görüntü olarak varsayar ve özellikleri mimariye kodlamamızı sağlar.

3

Convolutional neural networks

- ▶ CNN'lerde **neuron'lar üç boyutlu** olarak düzenlenir.



- ▶ CNN'lerde **her katman 3D giriş alabilir** ve **3D çıkış üretebilir**.
- ▶ **Giriş katmanı** görüntüyü alır.
- ▶ Giriş katmanının **genişlik** ve **yüksekliği**, görüntünün **genişlik** ve **yüksekliği** kadardır.
- ▶ Giriş katmanının **derinliği 3 olabilir** (red, green, blue).

4

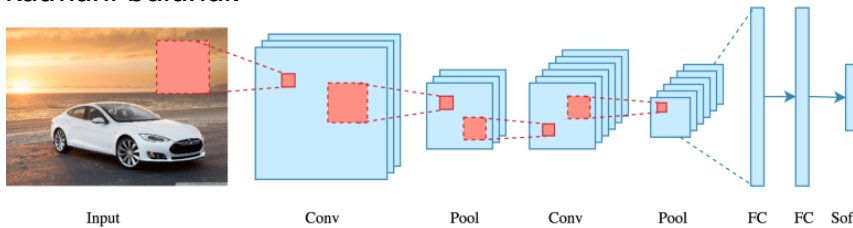
İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ Hyperparameters

5

CNN'lerin yapısı

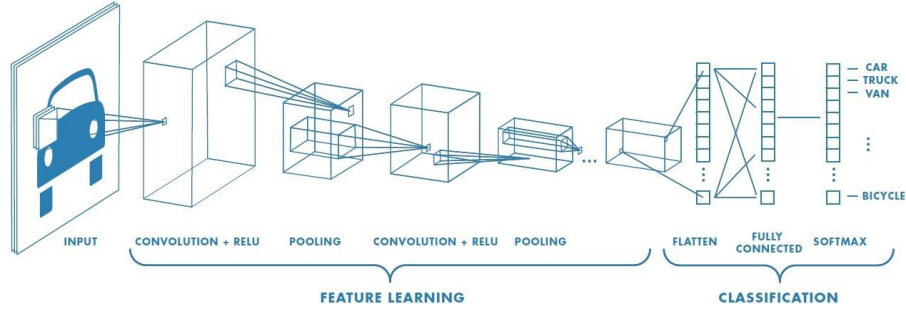
- ▶ CNN, **convolution** ve **pooling** operatörlerini kullanır.
- ▶ Bir CNN **üç temel katman türüne sahiptir**:
 - ▶ Convolutional layer
 - ▶ Pooling layer
 - ▶ Fully-connected layer
- ▶ Art arda çok sayıda **convolution+pooling** yapılabilir.
- ▶ Daha sonra birkaç tane **fully connected katmanı** bulunur.
- ▶ Çok etiketli sınıflandırma problemlerinde, en sonda **softmax** katmanı bulunur.



6

CNN'lerin yapısı

- ▶ **Fully-connected katmanı** üç boyutlu girişi tek boyuta indirgeyerek alır ve bir sınıf etiketi elde eder.
- ▶ **Softmax katmanı** çıkış sınıflarının **olasılık dağılımını hesaplar**.

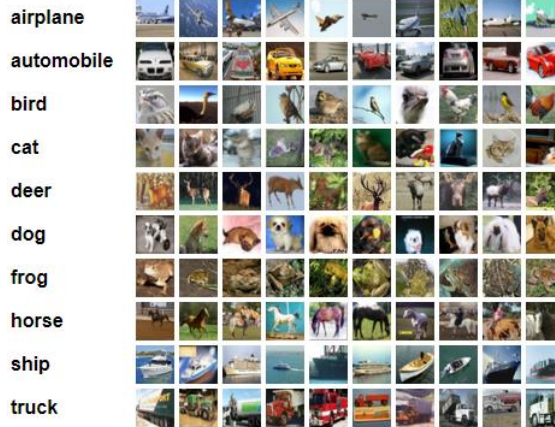


7

CNN'lerin yapısı

Örnek

- ▶ **CIFAR-10*** veriseti, **10 sınıfa** ait **60.000 tane 32x32 renkli görüntüye sahiptir** (her sınıf için 6.000 görüntü).
- ▶ **50.000 train** ve **10.000 test** için ayrılabilir.



*CIFAR-100 (Canadian Institute For Advanced Research) 100 sınıfa ve 600.000 adet 32x32 görüntüye sahiptir.

8

CNN'lerin yapısı

Örnek

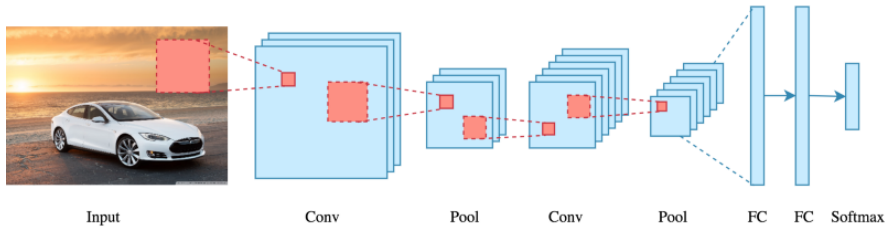
- ▶ CIFAR-10 veriseti için [**Input-Conv-ReLU-Pool-FC**] katmanları kullanılabilir.
- ▶ **Giriş katmanı, 32x32x3 (kırmızı, yeşil, mavi)** görüntü piksellerini alır.
- ▶ **Convolution katmanı, seçilen filtreyi kullanarak** girişin lokal bölgelerinden aldığı değerler üzerinde **hesaplama yapar**.
- ▶ Eğer **12 farklı filtre kullanılırsa**, convolution katmanının **çıkışı 32x32x12 olur** (RGB birleştirilir).
- ▶ **ReLU (Rectifier Linear Units) katmanı, $\max(0, x)$** aktivasyon fonksiyonu sonucunu hesaplar ve 32x32x12 çıkış üretir.

9

CNN'lerin yapısı

Örnek

- ▶ **Pool katmanı**, bir alt örnekleme işlemi gerçekleştirir ve çıkış boyutu örneğin 16x16x12 olabilir.
- ▶ **Fully Connected katmanı**, 1x1x10 ile **çıkış sınıfının skorunu hesaplar**.
- ▶ Problem türüne göre farklı sayıda CONV+RELU+POOL katmanları art arda kullanılarak daha başarılı sonuç alınabilir.

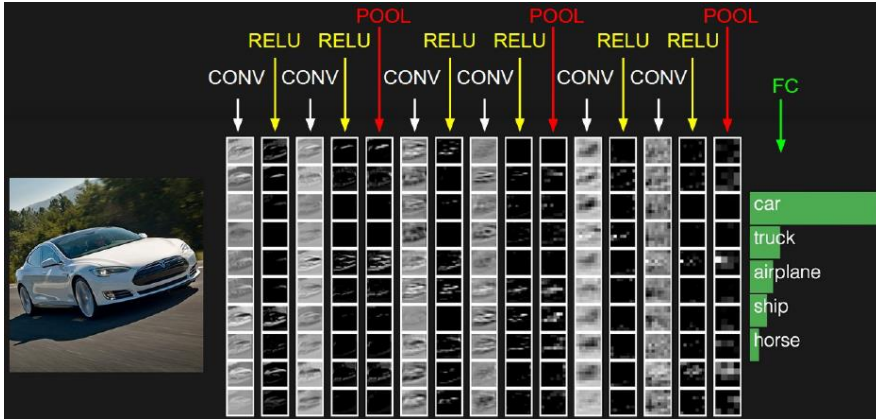


10

CNN'lerin yapısı

Örnek

- ▶ CIFAR-10 veriseti için örnek uygulama <http://cs231n.stanford.edu/>



11

İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ **Convolution**
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ Hyperparameters

12

Convolution

- ▶ CNN'de **en temel blok convolution katmanıdır.**
- ▶ **Convolution, iki kümenin birleştirilmesini** sağlayan matematiksel işlemidir.
- ▶ **Convolution filter (kernel)**, girişe uygulanarak **özellik haritası (feature map)** oluşturulur.

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Input

1	0	1
0	1	0
1	0	1

Filter / Kernel

13

Convolution

- ▶ Örnekte, giriş **5x5** ve filtre **3x3** boyutundadır.
- ▶ Giriş matrisinin üzerinde **filtre kaydırılarak convolution işlemi yapılır.**
- ▶ Karşılıklı elemanlar ile yapılan **matris çarpımının sonucu** özellik harita matrisinin **bir elemanını oluşturur.**
- ▶ Şekilde, 3x3 filtre ile 2D üzerinde convolution yapılıyor.

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

Input x Filter

4		

Feature Map

14

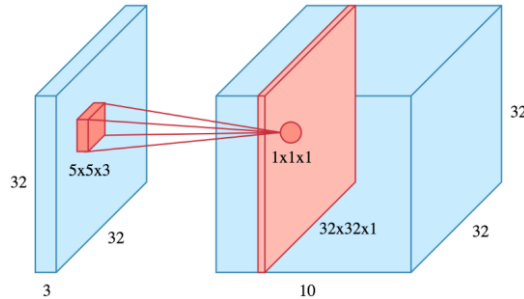
Convolution

- ▶ **Gerçek uygulamalarda** görüntü **3D** gösterilir (yükseklik, genişlik ve derinlik).
- ▶ **Derinlik**, görüntüdeki renk kanallarını gösterir.
- ▶ **RGB** için derinlik **3 olarak alınır**.
- ▶ Bir giriş üzerinde **farklı filtrelere sahip farklı convolution işlemleri yapılabilir**.
- ▶ Her filtrenin özellik haritası farklıdır.
- ▶ **Tüm özellik haritaları birleştirilerek** bir özellik haritası elde edilir.

15

Convolution

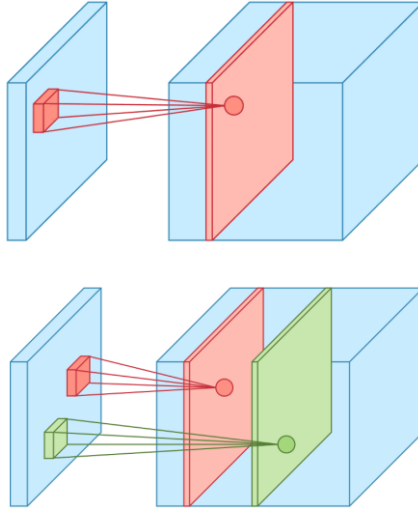
- ▶ Şekilde, **32x32x3** bir görüntü, **5x5x3** filtre kullanılmıştır.
- ▶ **Üç tane 5x5x1 matris toplanarak 1x1x1 değer elde edilir**.
- ▶ Elde edilen **özellik haritası 32x32x1 boyutundadır**.
- ▶ Eğer **10 farklı filtre kullanılırsa**, convolution katmanı **32x32x10 boyutunda oluşur**.



16

Convolution

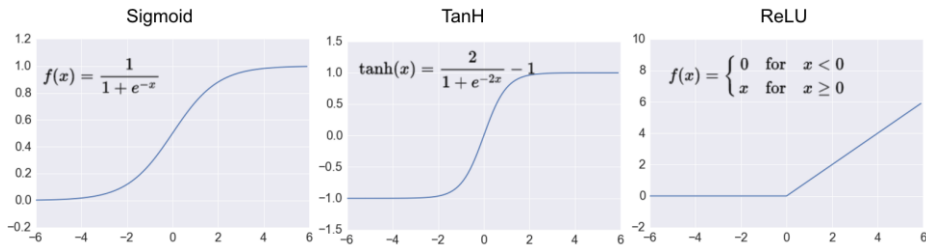
- **Filtre** tüm girişte **kaydırılarak özellik haritası elde edilir.**



17

Convolution

- CNN'de **convolution operatörünün** sonucunu **aktivasyon fonksiyonuna gönderir.**
- **Aktivasyon fonksiyonu probleme özgü olarak seçilir.**



18

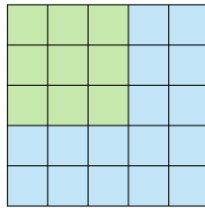
İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ **Stride ve padding**
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ Hyperparameters

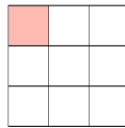
19

Stride ve padding

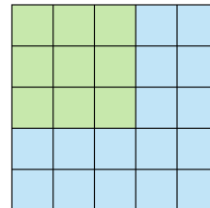
- ▶ **Stride**, her adımda convolution filtresinin **hareket adım boyutunu belirler** (default = 1).
- ▶ **Hareket adım boyutu arttıkça**, elde edilecek **özellik haritasının boyutu küçülür**.



Stride 1



Feature Map



Stride 2

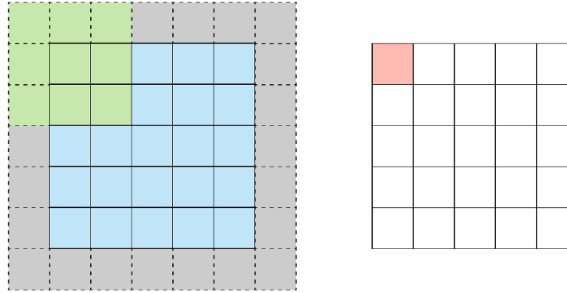


Feature Map

20

Stride ve padding

- ▶ **Padding**, giriş ile aynı boyutta özellik haritası oluşturmak için kullanılır.
- ▶ Giriş matrisinin etrafına **0 değerine sahip hücreler padding olarak eklenir.**



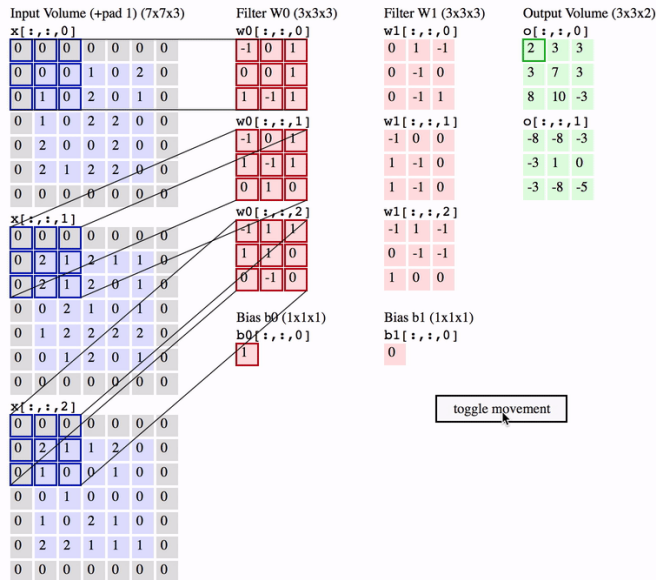
Stride 1 with Padding

Feature Map

21

Stride ve padding

- ▶ **Örnek: Giriş= 5x5x3, Padding= 1, Stride= 2**



22

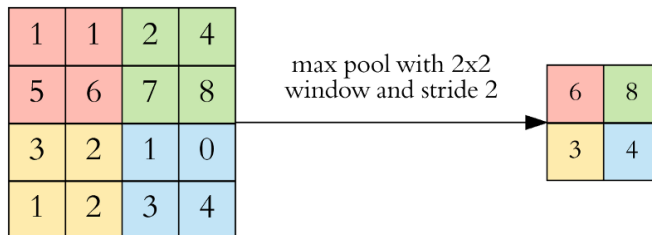
İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ Hyperparameters

23

Pooling

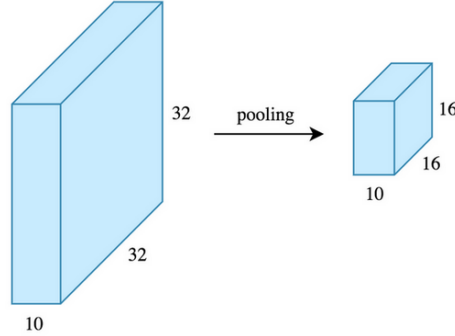
- ▶ **Pooling**, convolution işleminden sonra uygulanır ve **boyut indirgeme yapar**.
- ▶ Pooling katmanı, özellik haritasının yüksekliğini ve genişliğini düşürerek örnekler (derinlik aynı kalır).
- ▶ **En yaygın** kullanılan **max pooling** yöntemidir.
- ▶ **Window size** ve **stride** değerleri belirlenir.



24

Pooling

- ▶ Genellikle, **window size** ve **stride** değerleri girişteki **özellik haritasının yarısı** elde edilecek şekilde seçilir.
- ▶ Pooling işleminden sonra **özellik haritasının boyutu yarıya indirgenmiş olur.**



25

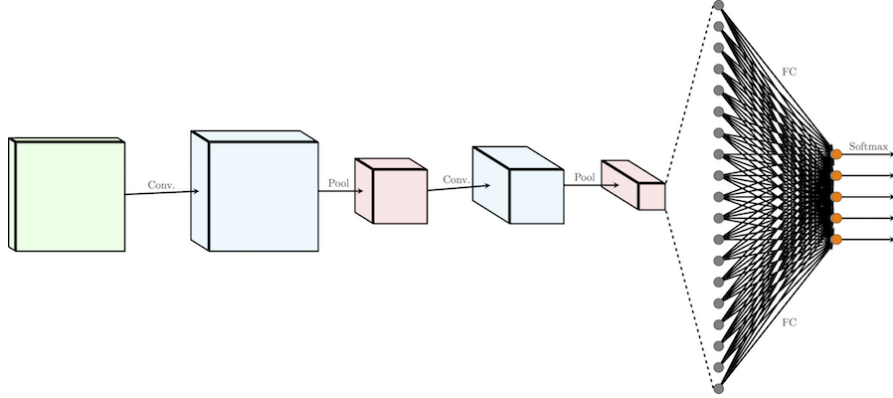
İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ Hyperparameters

26

Fully connected layer

- ▶ Pooling katmanından sonra, **fully connected ANN** eklenir.
- ▶ Pooling katmanı çıkışı **3D alınır** ve fully connected ANN ile **1D vektör çıkışı elde edilir.**



27

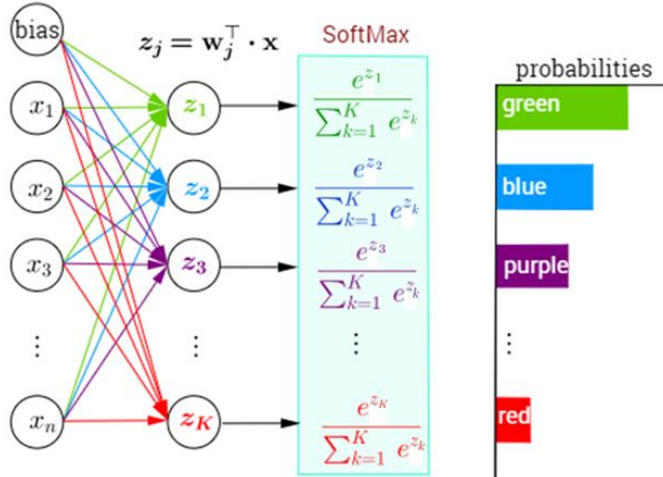
İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ **Softmax**
- ▶ Hyperparameters

28

Softmax

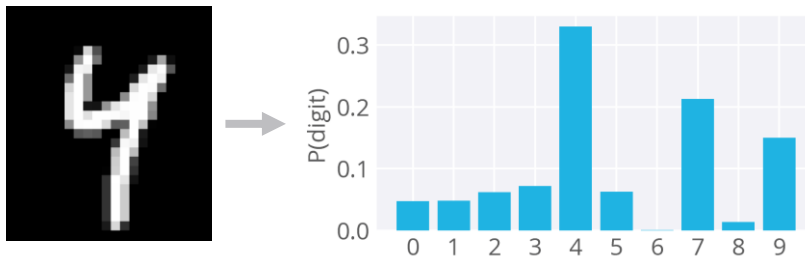
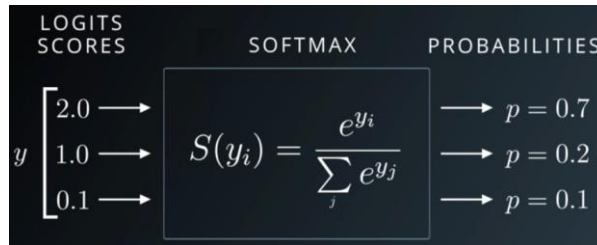
- ▶ Softmax fonksiyonu sınıflandırma problemlerinde kullanılır.
- ▶ **Softmax katmanı** çıkış sınıflarının **olasılık dağılımını hesaplar.**



29

Softmax

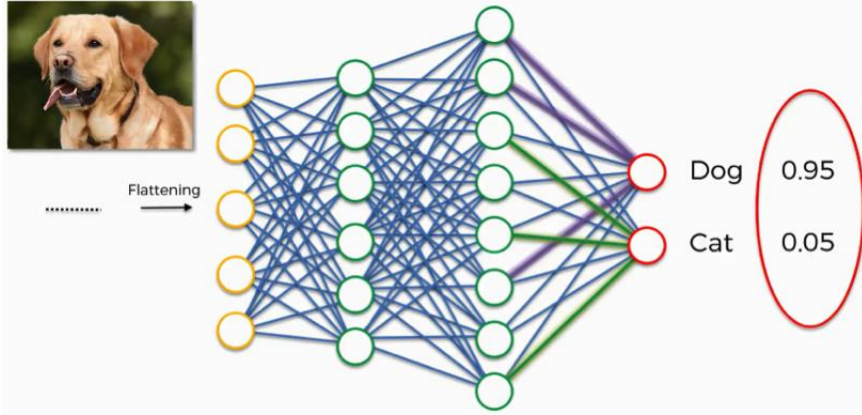
- ▶ Softmax, çıkışın sınıflara ait olma olasılığının dağılımını verir.



30

Softmax

- ▶ Genellikle çıkış neuron sayısı sınıf etiketi sayısı kadar alınır.
- ▶ **Yüksek olasılığa sahip sınıf etiketi verilen girişler için atanır.**



31

İçerik

- ▶ Convolutional neural networks
- ▶ CNN'lerin yapısı
- ▶ Convolution
- ▶ Stride ve padding
- ▶ Pooling
- ▶ Fully connected layer
- ▶ Softmax
- ▶ **Hyperparameters**

32

Hyperparameters

- ▶ **Hiper parametreler**, doğrudan öğrenilmezler, ancak **modelin özelliklerini belirlerler**.
- ▶ CNN'de, aşağıdaki hiper parametreler kullanılır:
 - ▶ **Filtre boyutu**: Genellikle 3x3 kullanılır, ancak probleme göre daha büyük boyutta olabilir.
 - ▶ **Filtre sayısı**: Daha çok filtre kullanıldıkça daha güçlü model elde edilir. Ancak, çok sayıdaki parametre **overfitting** riskini artırır.
 - ▶ **Stride**: Genellikle 1 seçilir, ancak probleme göre farklı değer seçilebilir.
 - ▶ **Padding**: Genellikle padding 1 olarak alınır, ancak probleme göre kullanılmayabilir.