Lecture 3B
Deep Learning for Computer Vision – The Basics

15.S04: Hands-on Deep Learning
Spring 2024

Farias, Ramakrishnan

Representing Images Digitally

How Grayscale Images are Represented

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]	[,14]	[,15]	[,16]	[,17]	[,18]	[,19]	[,20]	[,21]	[,22]	[,23]	[,24]	[,25]	[,26]	[,27]	[,28]
[1,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[2,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[3,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[4,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[5,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[6,]	0	0	0	0	0	0	0	0	0	0	0	0	3	18	18	18	126	136	175	26	166	255	247	127	0	0	0	0
[7,]	0	0	0	0	0	0	0	0	30	36	94	154	170	253	253	253	253	253	225	172	253	242	195	64	0	0	0	0
[8,]	0	0	0	0	0	0	0	49	238	253	253	253	253	253	253	253	253	251	93	82	82	56	39	0	0	0	0	0
[9,]	0	0	0	0	0	0	0	18	219	253	253	253	253	253	198	182	247	241	0	0	0	0	0	0	0	0	0	0
[10,]	0	0	0	0	0	0	0	0	80	156	107	253	253	205	11	0	43	154	0	0	0	0	0	0	0	0	0	0
[11,]	0	0	0	0	0	0	0	0	0	14	1	154	253	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[12,]	0	0	0	0	0	0	0	0	0	0	0	139	253	190	2	0	0	0	0	0	0	0	0	0	0	0	0	0
[13,]	0	0	0	0	0	0	0	0	0	0	0	11	190	253	70	0	0	0	0	0	0	0	0	0	0	0	0	0
[14,]	0	0	0	0	0	0	0	0	0	0	0	0	35	241	225	160	108	1	0	0	0	0	0	0	0	0	0	0
[15,]	0	0	0	0	0	0	0	0	0	0	0	0	0	81	240	253	253	119	25	0	0	0	0	0	0	0	0	0
[16,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45	186	253	253	150	27	0	0	0	0	0	0	0	0
[17,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	93	252	253	187	0	0	0	0	0	0	0	0
[18,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	249	253	249	64	0	0	0	0	0	0	0
[19,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46	130	183	253	253	207	2	0	0	0	0	0	0	0
[20,]	0	0	0	0	0	0	0	0	0	0	0	0	39	148	229	253	253	253	250	182	0	0	0	0	0	0	0	0
[21,]	0	0	0	0	0	0	0	0	0	0	24	114	221	253	253	253	253	201	78	0	0	0	0	0	0	0	0	0
[22,]	0	0	0	0	0	0	0	0	23	66	213	253	253	253	253	198	81	2	0	0	0	0	0	0	0	0	0	0
[23,]	0	0	0	0	0	0	18	171	219	253	253	253	253	195	80	9	0	0	0	0	0	0	0	0	0	0	0	0
[24,]	0	0	0	0	55	172	226	253	253	253	253	244	133	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[25,]	0	0	0	0	136	253	253	253	212	135	132	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[26,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[27,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[28,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Image credit: fast.ai

How Grayscale Images are Represented

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]	[,14]	[,15]	[,16]	[,17]	[,18]	[,19]	[,20]	[,21]	[,22]	[,23]	[,24]	[,25]	[,26]	[,27]	[,28]
[1,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[2,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[3,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[4,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[5,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[6,]	0	0	0	0	0	0	0	0	0	0	0	0	3	18	18	18	126	136	175	26	166	255	247	127	0	0	0	0
[7,]	0	0	0	0	0	0	0	0	30	36	94	154	170	253	253	253	253	253	225	172	253	242	195	64	0	0	0	0
[8,]	0	0	0	0	0	0	0	49	238	253	253	253	253	253	253	253	253	251	93	82	82	56	39	0	0	0	0	0
[9,]	0	0	0	0	0	0	0	18	219	253	253	253	253	253	198	182	247	241	0	0	0	0	0	0	0	0	0	0
[10,]	0	0	0	0	0	0	0	0	80	156	107	253	253	205	11	0	43	154	0	0	0	0	0	0	0	0	0	0
[11,]	0	0	0	0	0	0	0	0	0	14	1	154	253	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[12,]	0	0	0	0	0	0	0	0	0	0	0	139	253	190	2	0	0	0	0	0	0	0	0	0	0	0	0	0
[13,]	0	0	0	0	0	0	0	0	0	0	0	11	190	253	70	0	0	0	0	0	0	0	0	0	0	0	0	0
[14,]	0	0	0	0	0	0	0	0	0	0	0	0	35	241	225	160	108	1	0	0	0	0	0	0	0	0	0	0
[15,]	0	0	0	0	0	0	0	0	0	0	0	0	0	81	240	253	253	119	25	0	0	0	0	0	0	0	0	0
[16,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45	186	253	253	150	27	0	0	0	0	0	0	0	0
[17,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	93	252	253	187	0	0	0	0	0	0	0	0
[18,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	249	253	249	64	0	0	0	0	0	0	0
[19,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46	130	183	253	253	207	2	0	0	0	0	0	0	0
[20,]	0	0	0	0	0	0	0	0	0	0	0	0	39	148	229	253	253	253	250	182	0	0	0	0	0	0	0	0
[21,]	0	0	0	0	0	0	0	0	0	0	24	114	221	253	253	253	253	201	78	0	0	0	0	0	0	0	0	0
[22,]	0	0	0	0	0	0	0	0	23	66	213	253	253	253	253	198	81	2	0	0	0	0	0	0	0	0	0	0
[23,]	0	0	0	0	0	0	18	171	219	253	253	253	253	195	80	9	0	0	0	0	0	0	0	0	0	0	0	0
[24,]	0	0	0	0	55	172	226	253	253	253	253	244	133	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[25,]	0	0	0	0	136	253	253	253	212	135	132	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[26,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[27,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[28,]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

- A grayscale image is a rectangular array of pixels
- The light intensity of each pixel is a number between 0 and 255. As the number increases from 0 to 255, the pixel goes from black through gray to white
- Each cell of the matrix shows the light intensity of the pixel at that location

Image credit: fast.ai

How Color Images are Represented

- Each pixel of a color image is represented by <u>three</u> intensities (not one), corresponding to the pixel's "redness", "blueness" and "greenness" (RGB)
- Each light intensity is still a number between 0 and 255
- Thus color images are represented as 3 matrices of numbers, corresponding to the Red, Green and Blue "channels" respectively.

How Color Images are Represented

killian.jpeg
JPEG image - 6 KB

Tags Add Tags...
Created Today, 4:30 PM
Modified Today, 4:30 PM
Content created Friday, April 3, 2020 at 4:30 PM
Dimensions 200×200
Color space RGB

Red

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	147	131	138	144	131	134	144	135	133	145
[2,]	140	131	141	149	138	138	143	132	136	146
[3,]										
[4,]										
[5,]										
[6,]										
[7,]										
[8,]										
[9,]										
[10,]										

Green

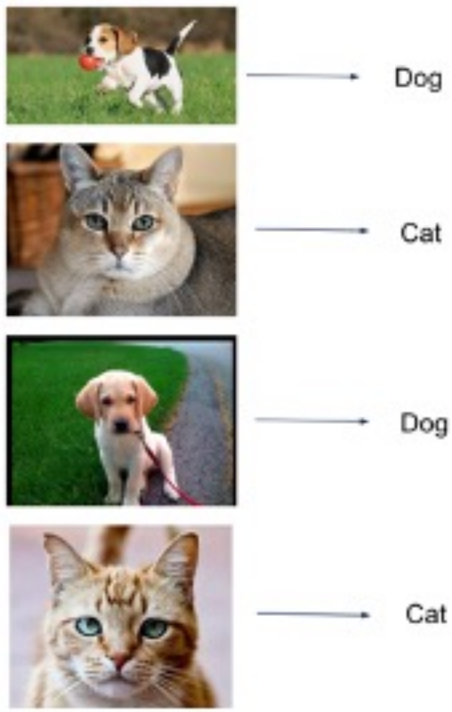
	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	186	171	179	185	171	172	180	171	168	180
[2,]	177	169	180	188	176	175	178	167	169	180
[3,]										
[4,]										
[5,]										
[6,]										
[7,]										
[8,]										
[9,]										
[10,]										

Blue

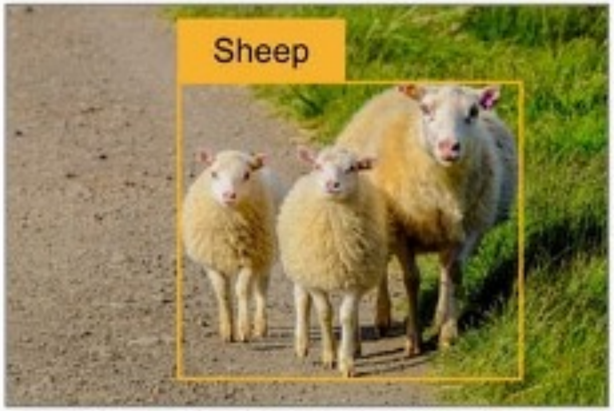
	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	251	232	233	237	230	243	255	255	250	246
[2,]	248	234	239	245	238	246	255	251	246	243
[3,]	255	241	238	236	229	241	253	249	238	234
[4,]	255	252	243	233	228	237	242	234	218	205
[5,]	255	255	249	231	228	231	224	215	204	166
[6,]	255	255	230	192	189	202	205	205	204	147
[7,]	231	231	188	140	138	152	156	159	177	136
[8,]	155	172	149	114	113	111	93	82	119	115
[9,]	107	130	108	93	113	100	67	66	81	95
[10,]	84	104	90	69	69	61	52	63	59	46

Key Tasks in Computer Vision

Image Classification

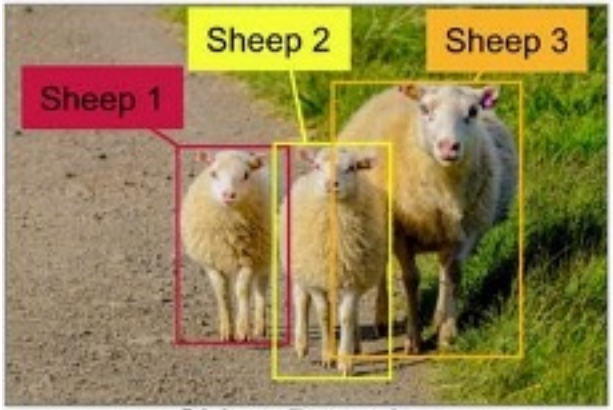


Classification and Localization



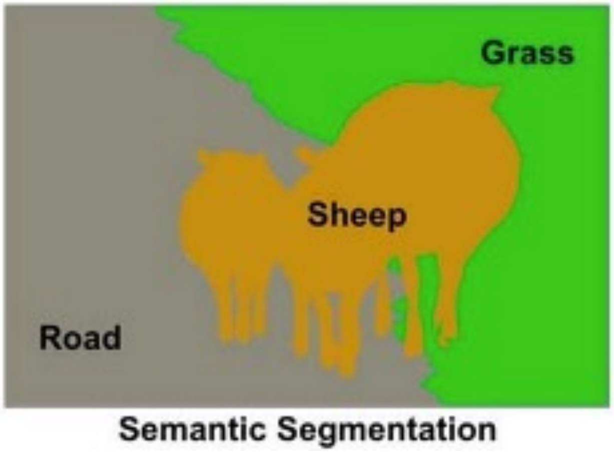
Classification + Localization

Object Detection



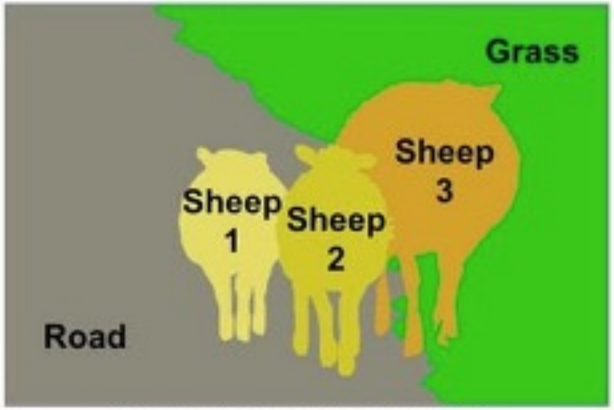
Object Detection

Semantic Segmentation



Every pixel needs to be classified into one of N categories

Instance Segmentation



Instance Segmentation

Every pixel needs to be classified into one of N categories and

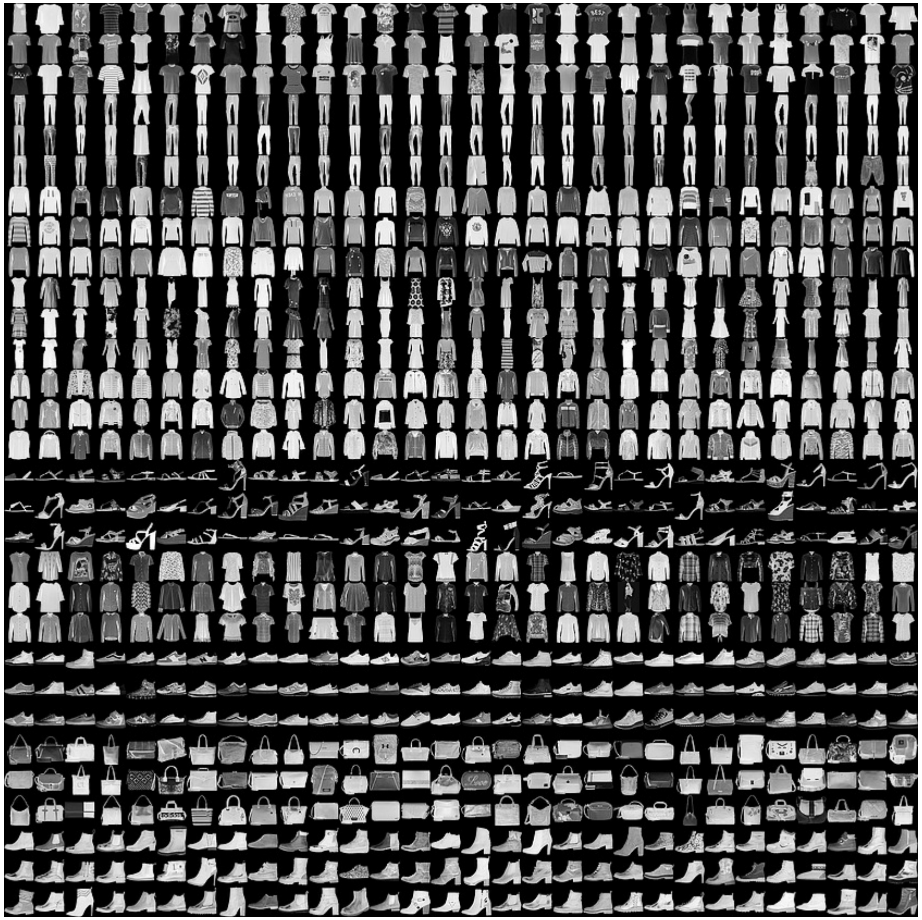
Different instances (e.g., Sheep 1, Sheep 2, Sheep 3) of the same category (e.g., Sheep) need to be identified

Image Classification

Motivating application: Fashion MNIST

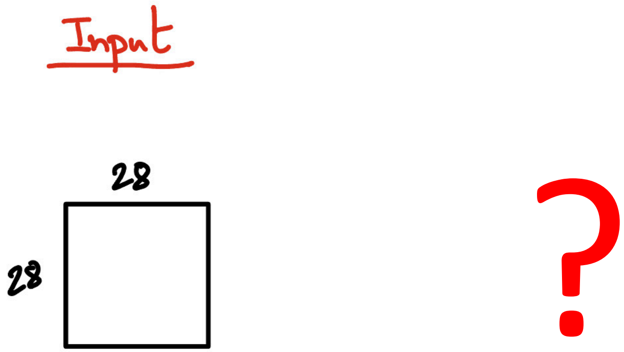
The fashion-mnist dataset consists of 70,000 images of clothing items across 10 categories.

We will build a deep learning network from scratch to classify clothing into these 10 categories with over 90% accuracy!

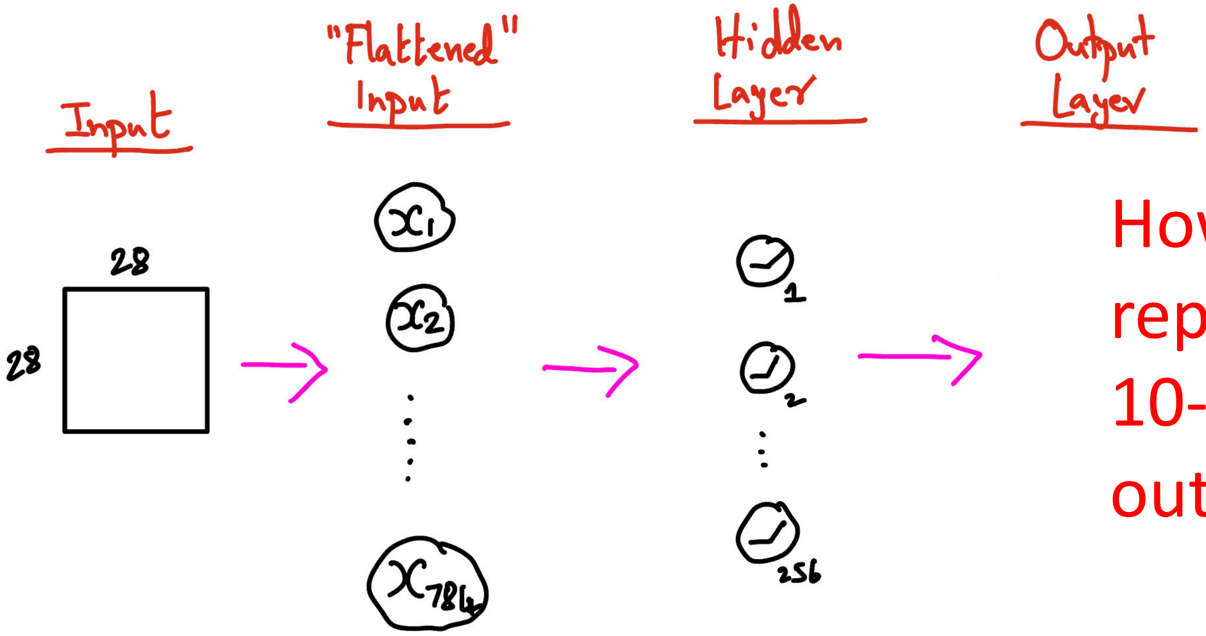


Source: https://www.kaggle.com/datasets/zalando-research/fashionmnist

A simple NN to classify grayscale clothing images



A simple NN to classify grayscale clothing images

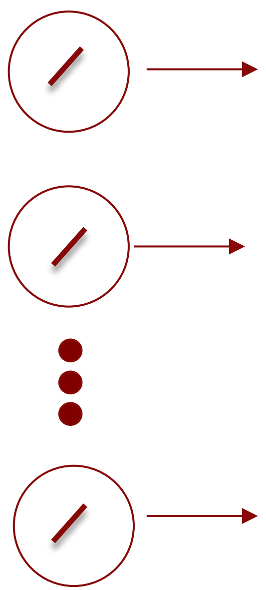


How do we represent a 10-way output?

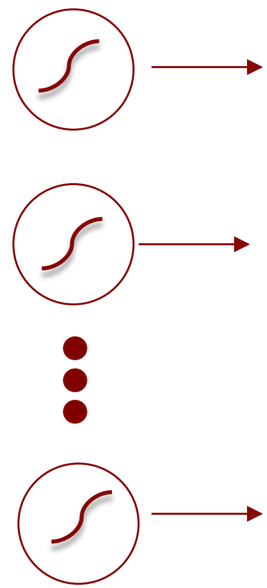
Multi-Class Classification

Suppose the output variable is categorical with 10 levels

We know how to
output 10 numbers



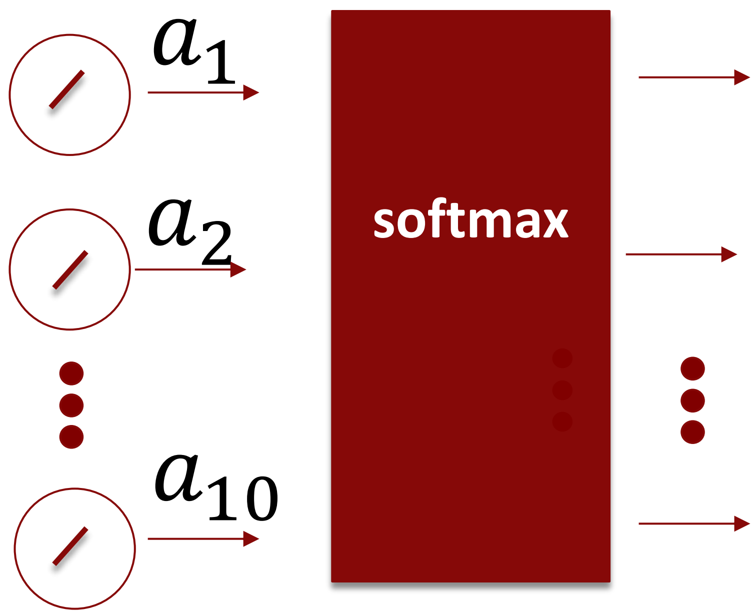
We know how to
output 10 probabilities



How do we output 10 probabilities that sum to 1.0?

The Softmax Layer

softmax takes in *n* arbitrary numbers and converts them to *n* probabilities

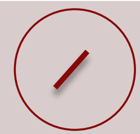


$$\frac{e^{a_1}}{e^{a_1} + e^{a_2} + \ldots + e^{a_{10}}}$$

$$\frac{e^{a_2}}{e^{a_1} + e^{a_2} + \ldots + e^{a_{10}}}$$

$$\frac{e^{a_{10}}}{e^{a_1} + e^{a_2} + \ldots + e^{a_{10}}}$$

Summary: Output Layers for Regression and Classification

Output Variable	Output Layer
Single number (regression with a single output)	
Single probability (binary classification)	
Vector of *n* numbers (regression with multiple outputs)	Stack of 
Vector of n probabilities that add up to 1 (multi-class classification)	Softmax

Refresher: How binary and categorical variables are encoded

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

Refresher: How binary and categorical variables are encoded

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

MULTI-CLASS CLASSIFICATION EXAMPLE

RAW DATA	SPARSE ENCODED VERSION
T-shirt/top	0
Trouser	1
Pullover	2
Dress	3
Coat	4
Sandal	5
Shirt	6
Sneaker	7
Bag	8
Ankle boot	9

Refresher: How binary and categorical variables are encoded

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

MULTI-CLASS CLASSIFICATION EXAMPLE

RAW DATA	SPARSE ENCODED VERSION	ONE-HOT ENCODED VERSION									
T-shirt/top	0	1	0	0	0	0	0	0	0	0	0
Trouser	1	0	1	0	0	0	0	0	0	0	0
Pullover	2	0	0	1	0	0	0	0	0	0	0
Dress	3	0	0	0	1	0	0	0	0	0	0
Coat	4	0	0	0	0	1	0	0	0	0	0
Sandal	5	0	0	0	0	0	1	0	0	0	0
Shirt	6	0	0	0	0	0	0	1	0	0	0
Sneaker	7	0	0	0	0	0	0	0	1	0	0
Bag	8	0	0	0	0	0	0	0	0	1	0
Ankle boot	9	0	0	0	0	0	0	0	0	0	1

Important: Pick the Keras crossentropy loss function that matches the encoding

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

→ binary_crossentropy

MULTI-CLASS CLASSIFICATION EXAMPLE

RAW DATA	SPARSE ENCODED VERSION	ONE-HOT ENCODED VERSION									
T-shirt/top	0	1	0	0	0	0	0	0	0	0	0
Trouser	1	0	1	0	0	0	0	0	0	0	0
Pullover	2	0	0	1	0	0	0	0	0	0	0
Dress	3	0	0	0	1	0	0	0	0	0	0
Coat	4	0	0	0	0	1	0	0	0	0	0
Sandal	5	0	0	0	0	0	1	0	0	0	0
Shirt	6	0	0	0	0	0	0	1	0	0	0
Sneaker	7	0	0	0	0	0	0	0	1	0	0
Bag	8	0	0	0	0	0	0	0	0	1	0
Ankle boot	9	0	0	0	0	0	0	0	0	0	1

<u>Important</u>: Pick the Keras crossentropy loss function that matches the encoding

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

binary_crossentropy

MULTI-CLASS CLASSIFICATION EXAMPLE

RAW DATA	SPARSE ENCODED VERSION	ONE-HOT ENCODED VERSION									
T-shirt/top	0	1	0	0	0	0	0	0	0	0	0
Trouser	1	0	1	0	0	0	0	0	0	0	0
Pullover	2	0	0	1	0	0	0	0	0	0	0
Dress	3	0	0	0	1	0	0	0	0	0	0
Coat	4	0	0	0	0	1	0	0	0	0	0
Sandal	5	0	0	0	0	0	1	0	0	0	0
Shirt	6	0	0	0	0	0	0	1	0	0	0
Sneaker	7	0	0	0	0	0	0	0	1	0	0
Bag	8	0	0	0	0	0	0	0	0	1	0
Ankle boot	9	0	0	0	0	0	0	0	0	0	1

sparse_categorical_crossentropy

Important: Pick the Keras crossentropy loss function that matches the encoding

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

→ binary_crossentropy

MULTI-CLASS CLASSIFICATION EXAMPLE

RAW DATA	SPARSE ENCODED VERSION	ONE-HOT ENCODED VERSION									
T-shirt/top	0	1	0	0	0	0	0	0	0	0	0
Trouser	1	0	1	0	0	0	0	0	0	0	0
Pullover	2	0	0	1	0	0	0	0	0	0	0
Dress	3	0	0	0	1	0	0	0	0	0	0
Coat	4	0	0	0	0	1	0	0	0	0	0
Sandal	5	0	0	0	0	0	1	0	0	0	0
Shirt	6	0	0	0	0	0	0	1	0	0	0
Sneaker	7	0	0	0	0	0	0	0	1	0	0
Bag	8	0	0	0	0	0	0	0	0	1	0
Ankle boot	9	0	0	0	0	0	0	0	0	0	1

sparse_categorical_crossentropy

categorical_crossentropy

Important: Pick the Keras crossentropy loss function that matches the encoding

BINARY CLASSIFICATION EXAMPLE

RAW DATA	ONE-HOT ENCODED VERSION
Yes	1
No	0

→ binary_crossentropy

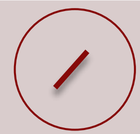
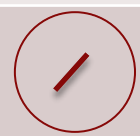
MULTI-CLASS CLASSIFICATION EXAMPLE

RAW DATA	SPARSE ENCODED VERSION	ONE-HOT ENCODED VERSION									
T-shirt/top	0	1	0	0	0	0	0	0	0	0	0
Trouser	1	0	1	0	0	0	0	0	0	0	0
Pullover	2	0	0	1	0	0	0	0	0	0	0
Dress	3	0	0	0	1	0	0	0	0	0	0
Coat	4	0	0	0	0	1	0	0	0	0	0
Sandal	5	0	0	0	0	0	1	0	0	0	0
Shirt	6	0	0	0	0	0	0	1	0	0	0
Sneaker	7	0	0	0	0	0	0	0	1	0	0
Bag	8	0	0	0	0	0	0	0	0	1	0
Ankle boot	9	0	0	0	0	0	0	0	0	0	1

↓ sparse_categorical_crossentropy

↓ categorical_crossentropy

Summary: Loss functions for different output layers

Output Variable	Output Layer	Loss Function
Single number (regression with a single output)		Mean squared error
Single probability (binary classification)		Binary cross-entropy
Vector of *n* numbers (regression with multiple outputs)	Stack of 	Mean squared error
Vector of n probabilities that add up to 1 (multi-class classification)	Softmax	Categorical cross-entropy

Let’s translate this NN to Keras and train it!

Colab