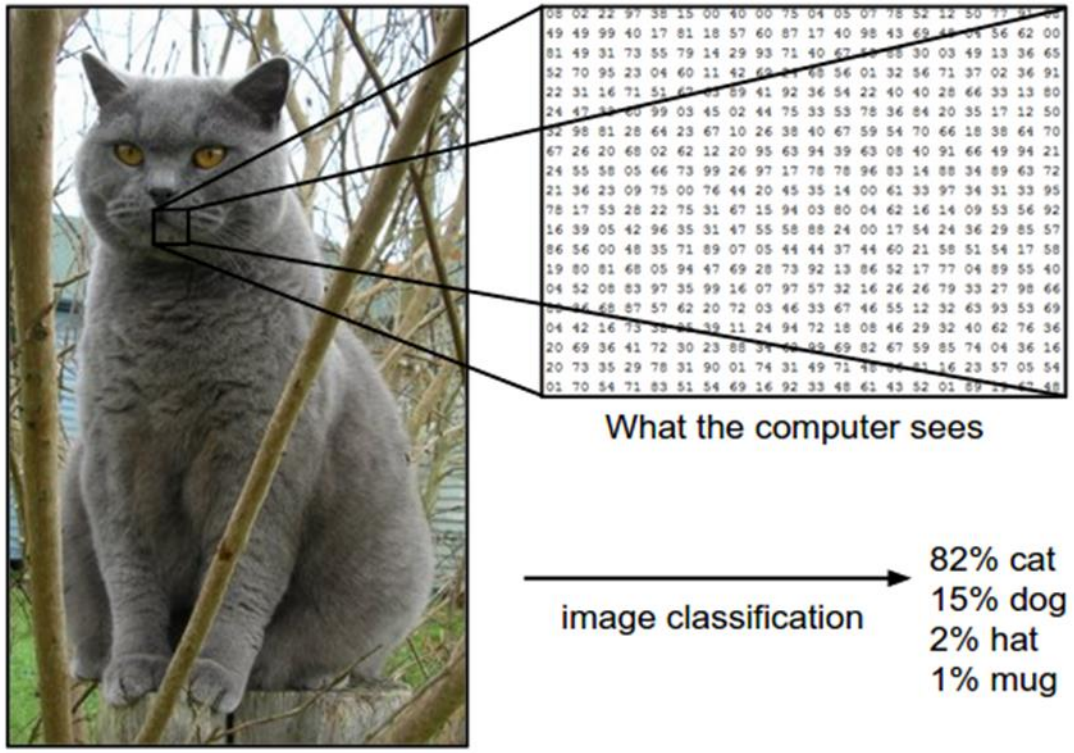
Machine Learning: Introduction to Classification

by Elena Battini Sönmez
İstanbul Bilgi University

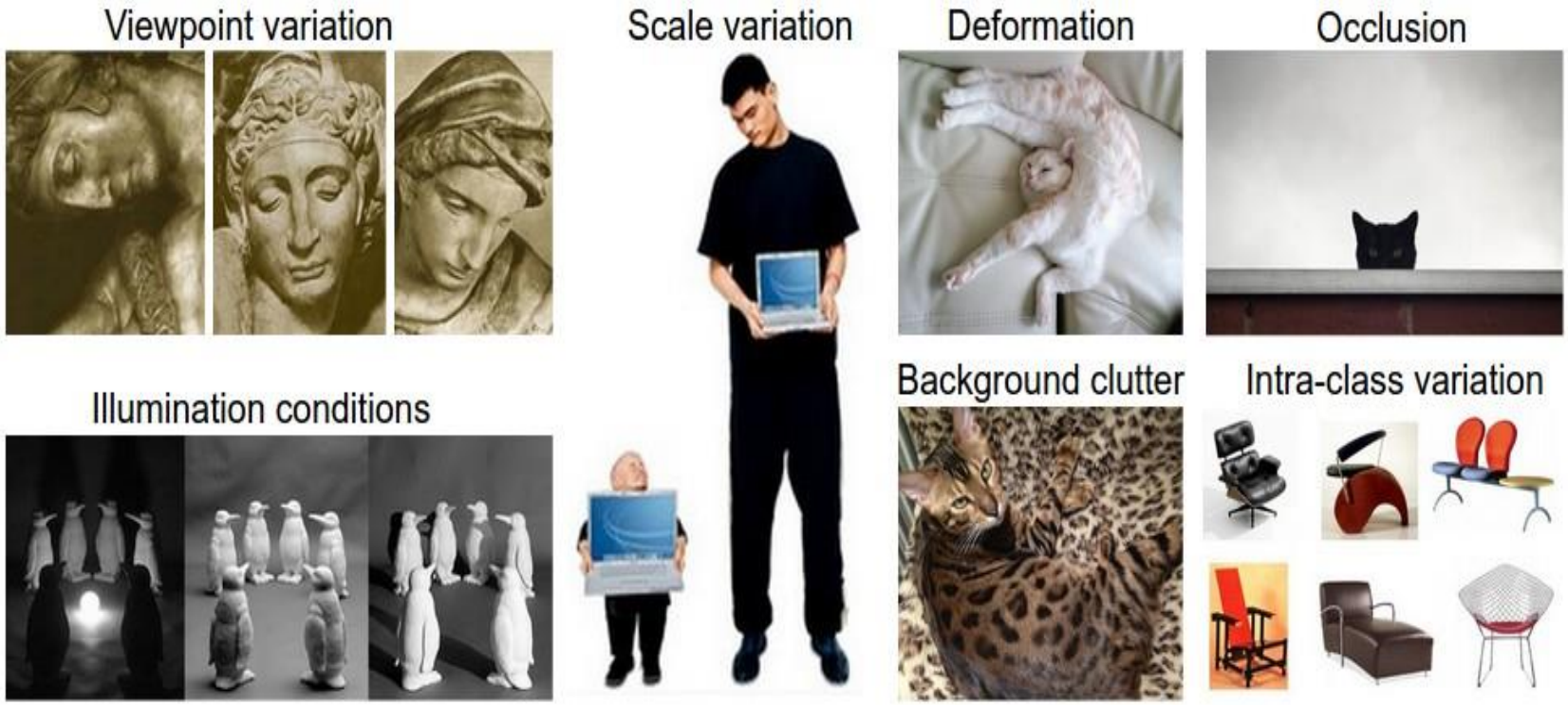
based on 'Stanford's notes'

Classification:

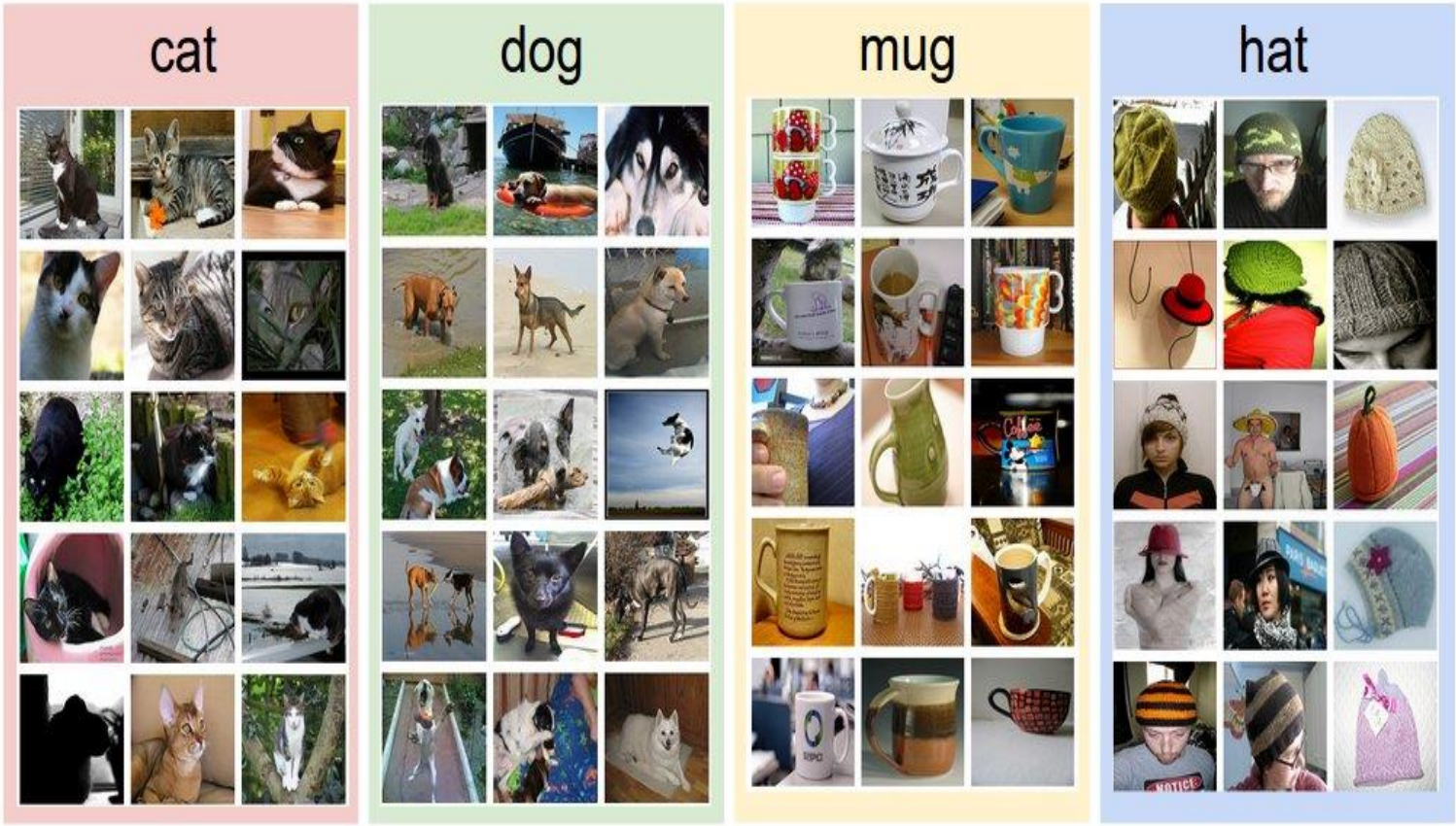
The task of image classification is to predict a single label for a given image.



Challenges:


Viewpoint variation
Scale variation
Deformation
Occlusion
Illumination conditions
Background clutter
Intra-class variation

Data Driven Approach:



Classification steps:

- **Input:** Our input consists of a set of *N* images, each labeled with one of *K* different classes. We refer to this data as the ***training set***.
- **Learning:** Our task is to use the training set to learn what every one of the classes looks like. We refer to this step as ***training a classifier***, or ***learning a model***.
- **Evaluation:** In the end, we evaluate the quality of the classifier by asking it to predict labels for a new set of images that it has never seen before. We will then compare the true labels of these images to the ones predicted by the classifier. Intuitively, we're hoping that a lot of the predictions match up with the true answers (which we call the ***ground truth***).

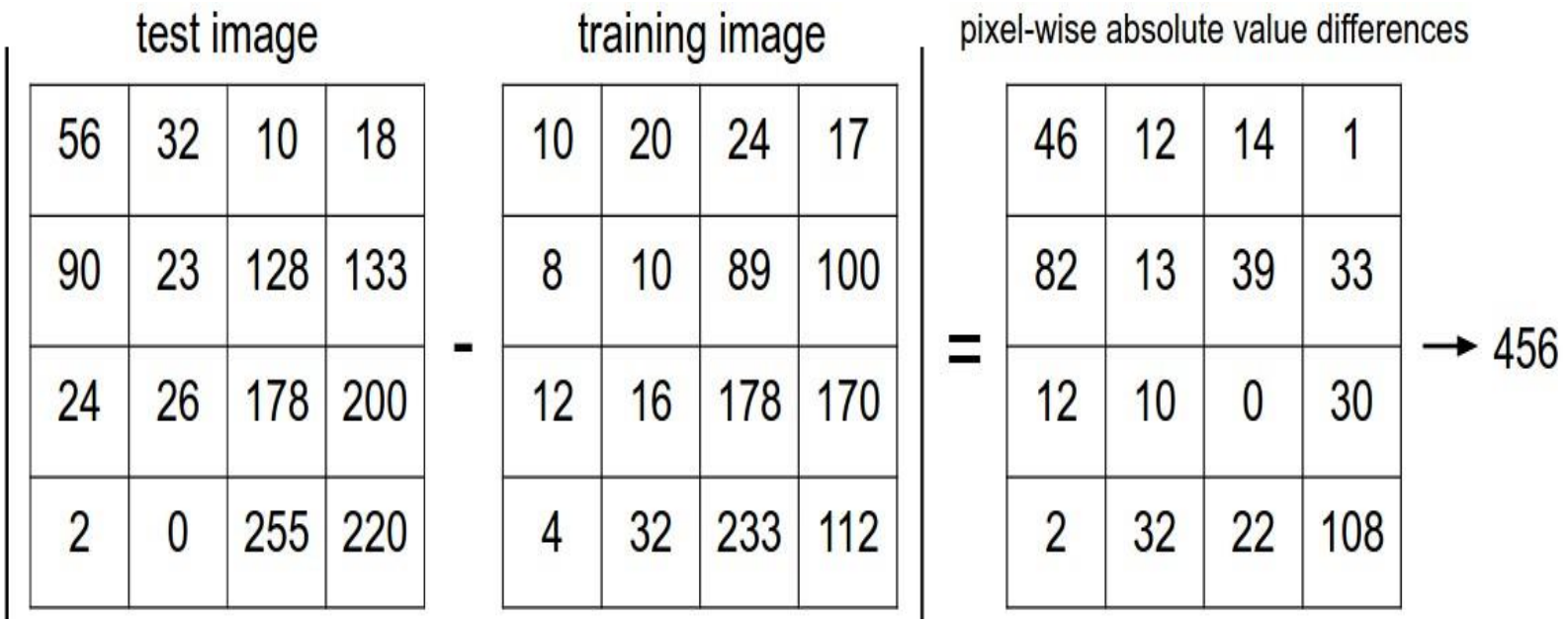
Nearest Neighbor (NN) Classifier:

- The nearest neighbor classifier takes a test image, compares it to every single training image, and assigns to the test sample the label of the closest training image

- Having two 1D images, I1 and I2, the L1 distance metric:

$$d1(I1,I2)=\sum p\,|I1-I2|$$

Example:


test image
training image
pixel-wise absolute value differences
56 32 10 18
90 23 128 133
24 26 178 200
2 0 255 220
-
10 20 24 17
8 10 89 100
12 16 178 170
4 32 233 112
=
46 12 14 1
82 13 39 33
12 10 0 30
2 32 22 108
→ 456

K-Nearest Neighbor (K-NN) Classifier:

- The idea is very simple: instead of finding the single closest image in the training set, we will find the top **k** closest images, and have them vote on the label of the test image.
- In particular, when $k = 1$, we recover the Nearest Neighbor classifier.
- Intuitively, higher values of **k** have a smoothing effect that makes the classifier more resistant to outliers

Comparison NN vs K-NN: