COMPUTER VISION

LECTURE 12 11.10.2019

Laura Ruotsalainen, Associate Professor
Department of Computer Science

TODAY'S LECTURE

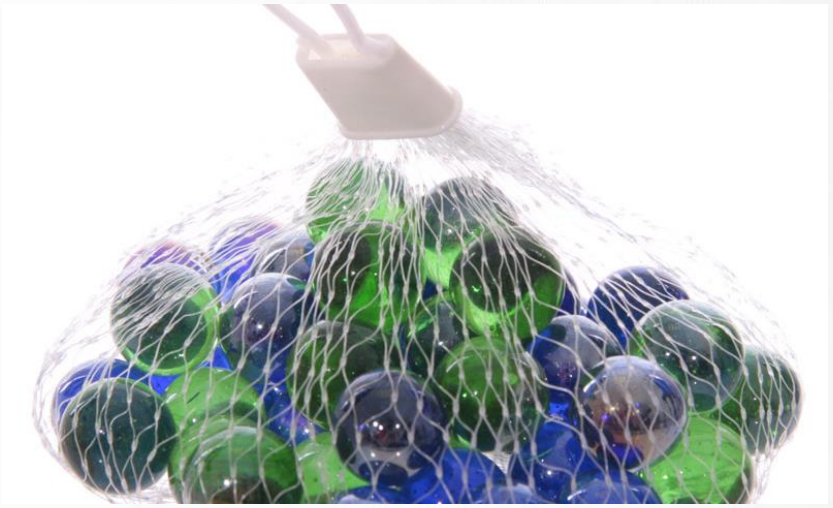
- Object detection and recognition
- Basic approaches with more traditional computer vision techniques
- Lecture 13: Deep learning in object detection and recognition
  - also discussion about the topical issues, e.g. how to fool the system
- Szeliski chapt. 14 has some material, but is quite out-dated

OBJECT DETECTION AND RECOGNITION

- **Object detection:** where is *this* object in the image?
  - *input:* a clear image of an object, or some kind of model of an object (e.g. duck) and an image (possibly) containing the object of interest
  - output: position, or a bounding box of the input object if it exists in the image
- **Object Recognition:** which object is depicted in the image?
  - *input*: an image containing unknown object(s)
  - *output*: position(s) and label(s) (names) of the objects in the image
  - The positions of objects are either acquired form the input, or determined based on the input image.
  - When labeling objects, there is usually a set of categories/labels which the system "knows" and between which the system can differentiate

OBJECT DETECTION NOW

- Object detection is one of the most challenging computer vision tasks
- Aims at identifying the presence of various individual objects in an image
- Good results have been obtained when dealing with images with relatively simple image scenes and clear foreground objects
- The problem is not adequately addressed when dealing with the images and videos containing objects placed in arbitrary poses, with various shapes, and appearing in a cluttered and occluded environment



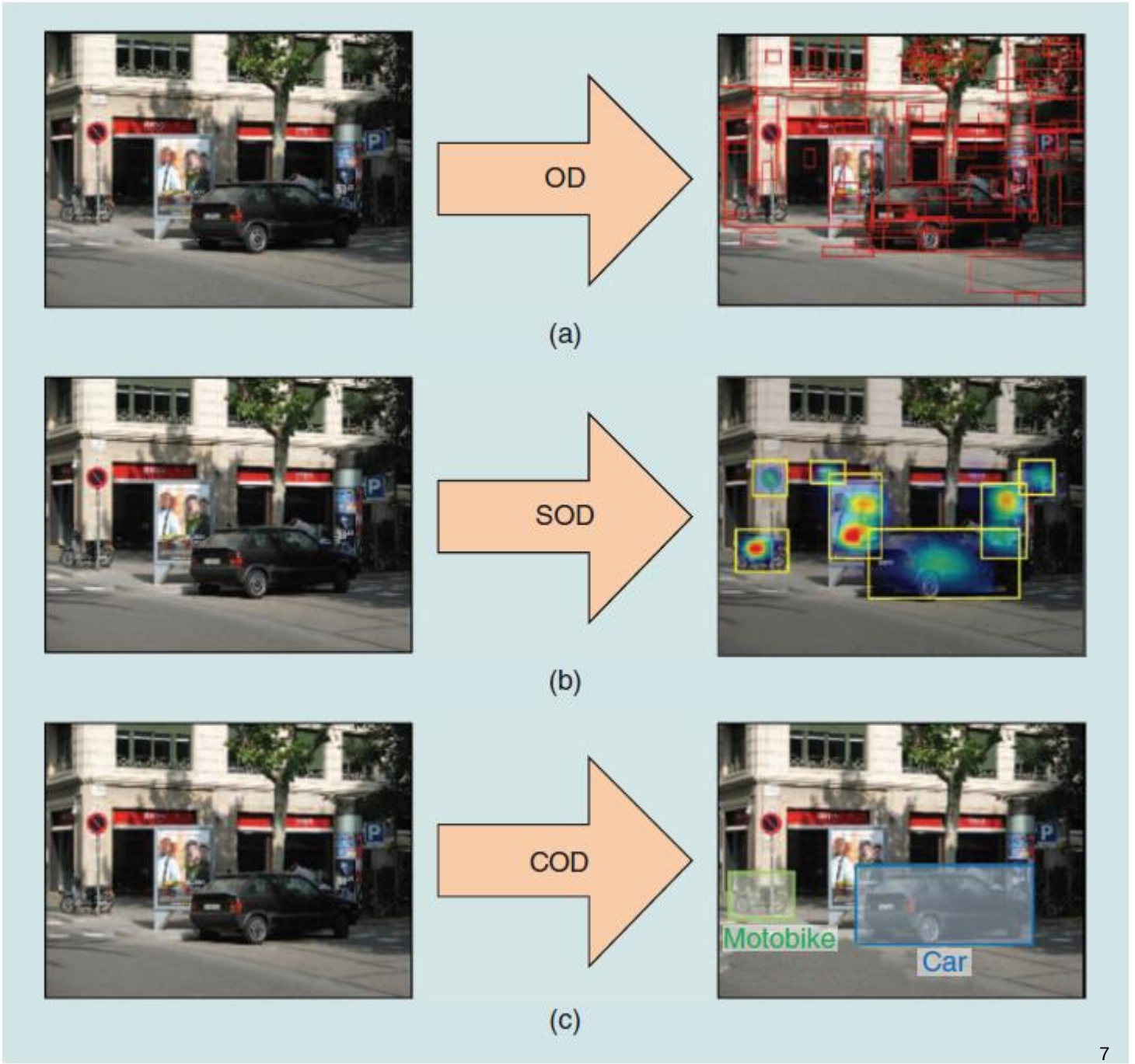
Han et al. (2018). Deep Learning for Visual Understanding: Part 2: Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection. IEEE Signal Processing Magazine.

OBJECT DETECTION CATEGORIES

- Object detection can be divided into three directions: Objectness Detection (OD), Salient Object Detection (SOD), Category-specific Object Detection (COD)
- OD aims at detecting all possible objects appearing in each given image, does not mind about the object category
  - Quantifies how likely it is for an image window to contain an object of any class, as opposed to backgrounds
  - Challenge: different objects have large appearance variation
  - Outputs thousands of object proposals
- SOD highlights objects that draw attention
  - Mimics human's visual system
  - Challenge: how to associate desired visual stimulus with corresponding image regions

OBJECT DETECTION CATEGORIES

- COD detects predefined object categories
  - Challenge: in addition to recognizing the image areas where objects are and must recognize the object category
  - Solves a computational problem without trying to understand the visual attention
  - Multiclass classification problem
- SIFT and HOG (Histograms of Oriented Gradients) the most used features
- Object detection and recognition is always a machine learning task
  - the type and amount of training images is different for each
- Convolutional Neural Networks have been used since 2004, breakthrough 2014



Han et al. (2018). Deep Learning for Visual Understanding: Part 2: Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection. IEEE Signal Processing Magazine.

What do we mean by “object recognition”?

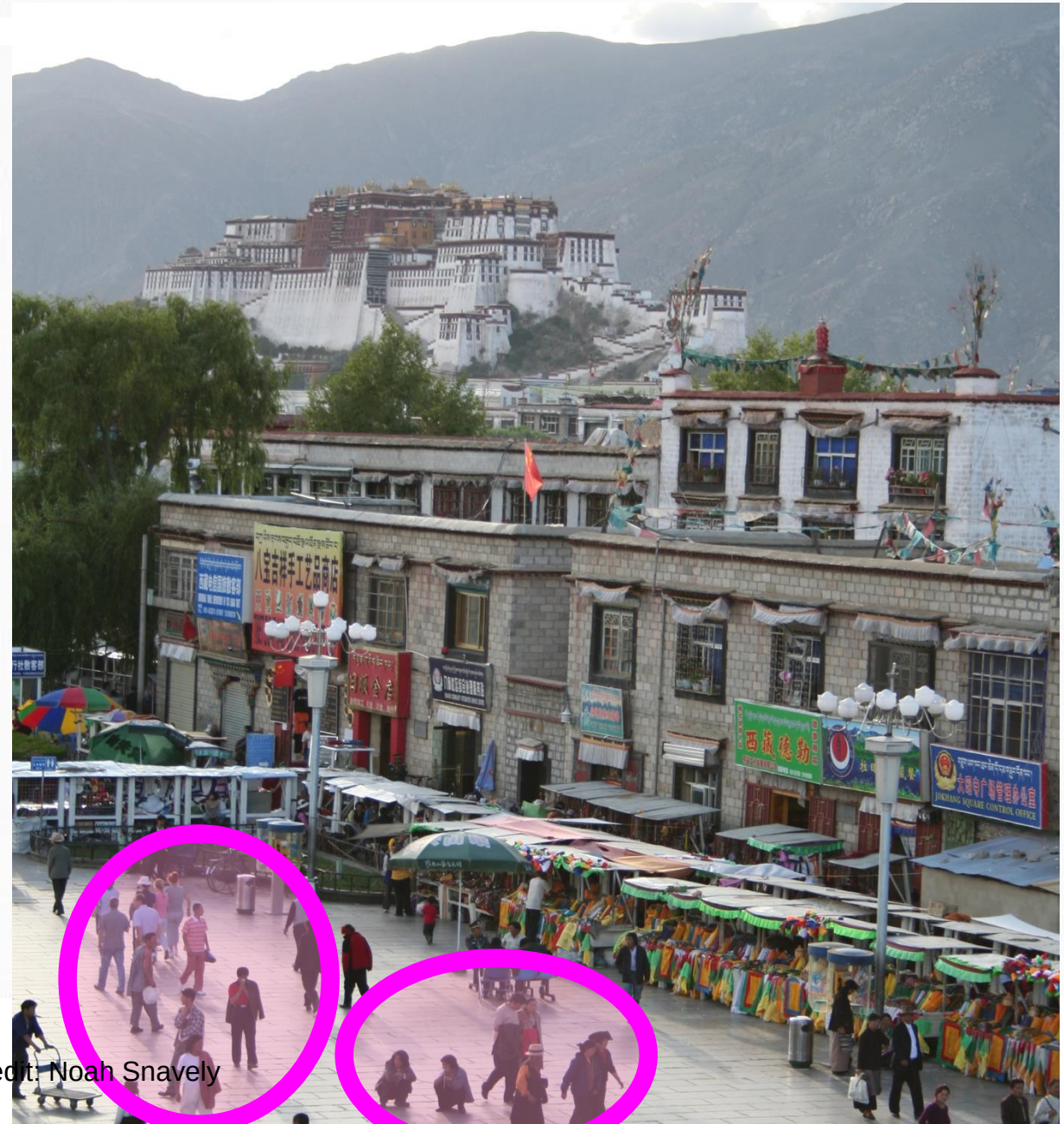
Next slides adapted from
Li, Fergus, & Torralba’s excellent
short course on category and
object recognition

Slide credit: Noah Snavely

Verification: is that a lamp?

Slide credit: Noah Snavely

Detection: where are the people?



Slide credit: Noah Snavely

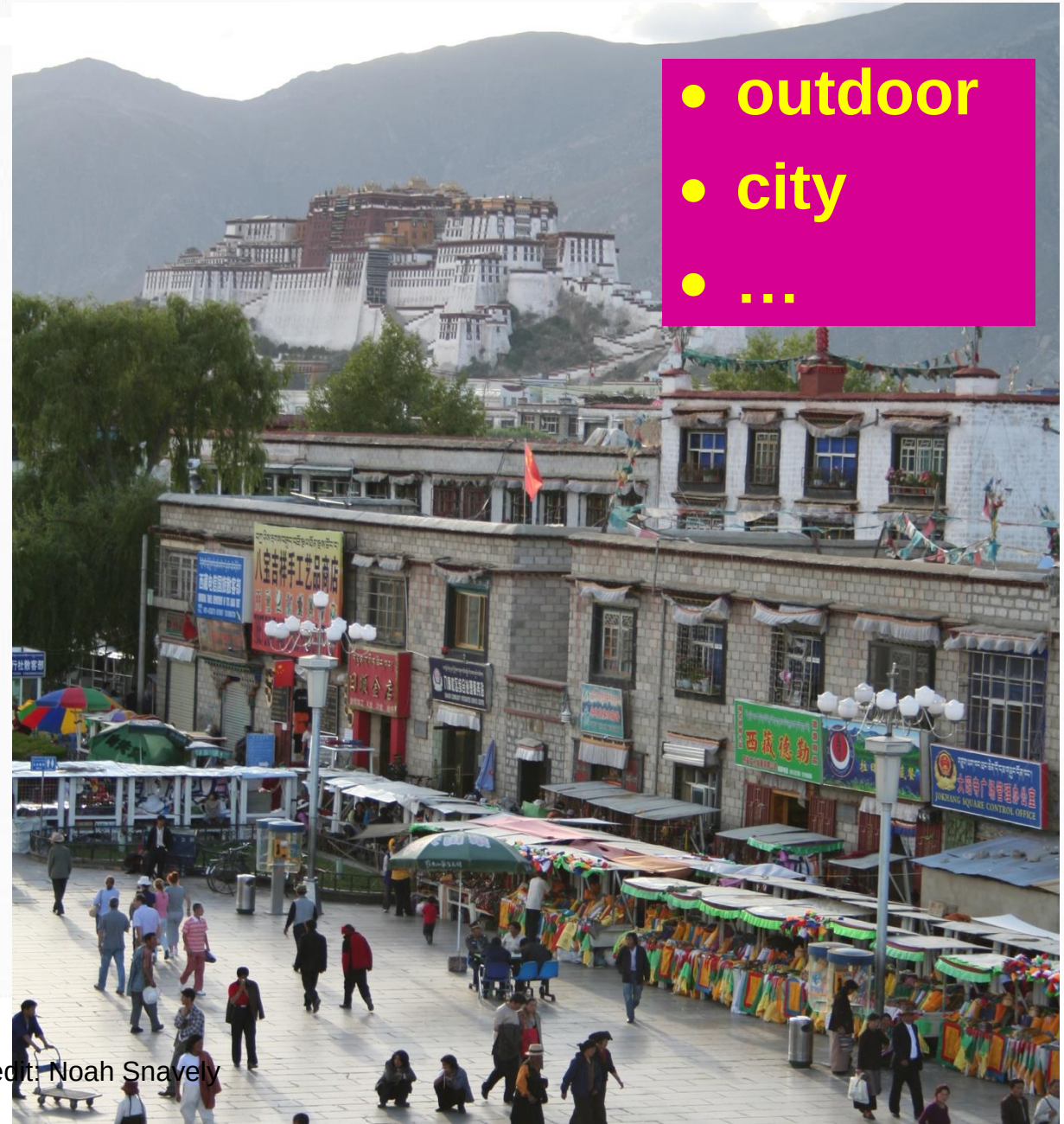
Identification: is that Potala Palace?

Slide credit: Noah Snavely

Object categorization

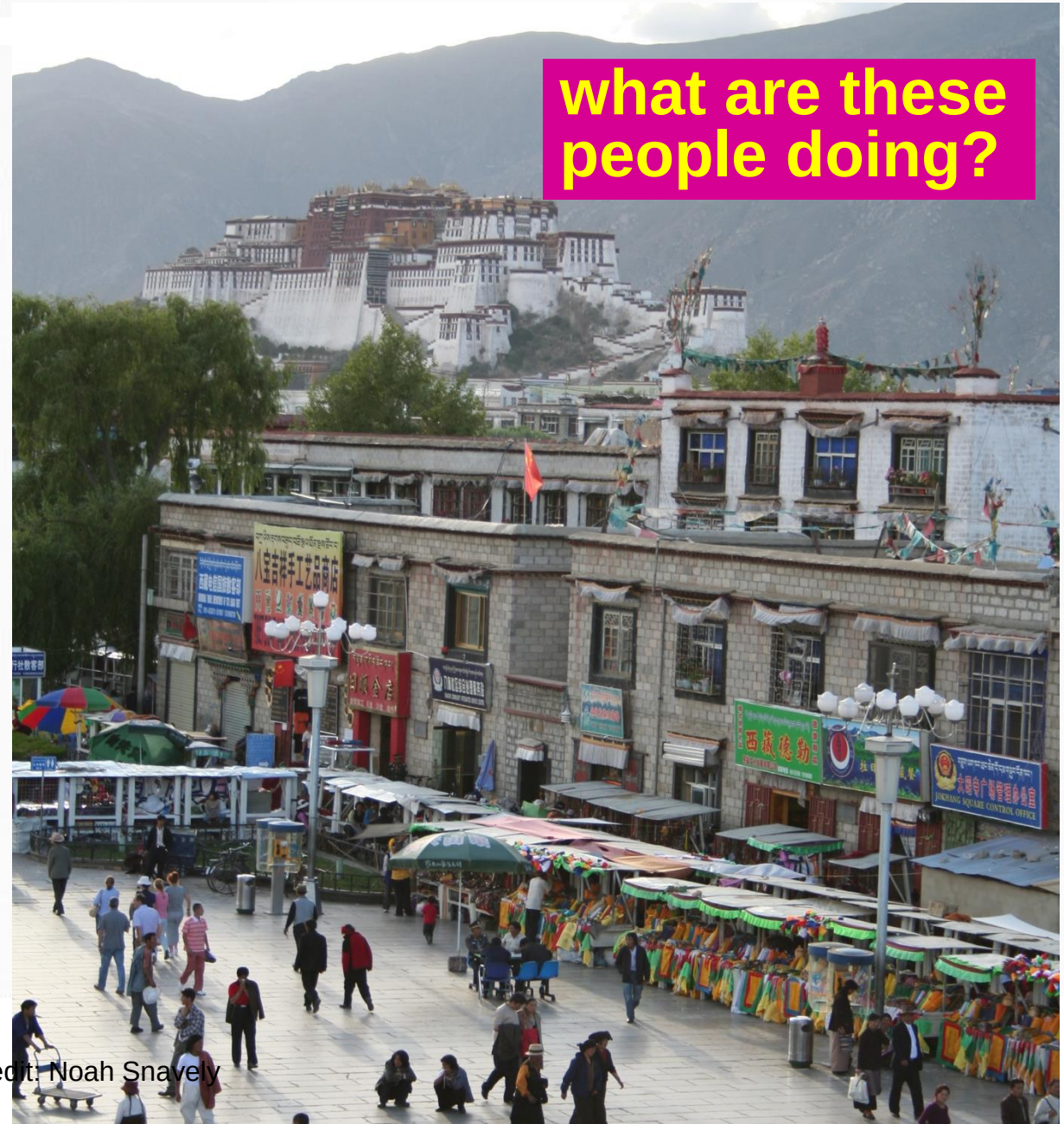
Slide credit: Noah Snavely

Scene and context categorization



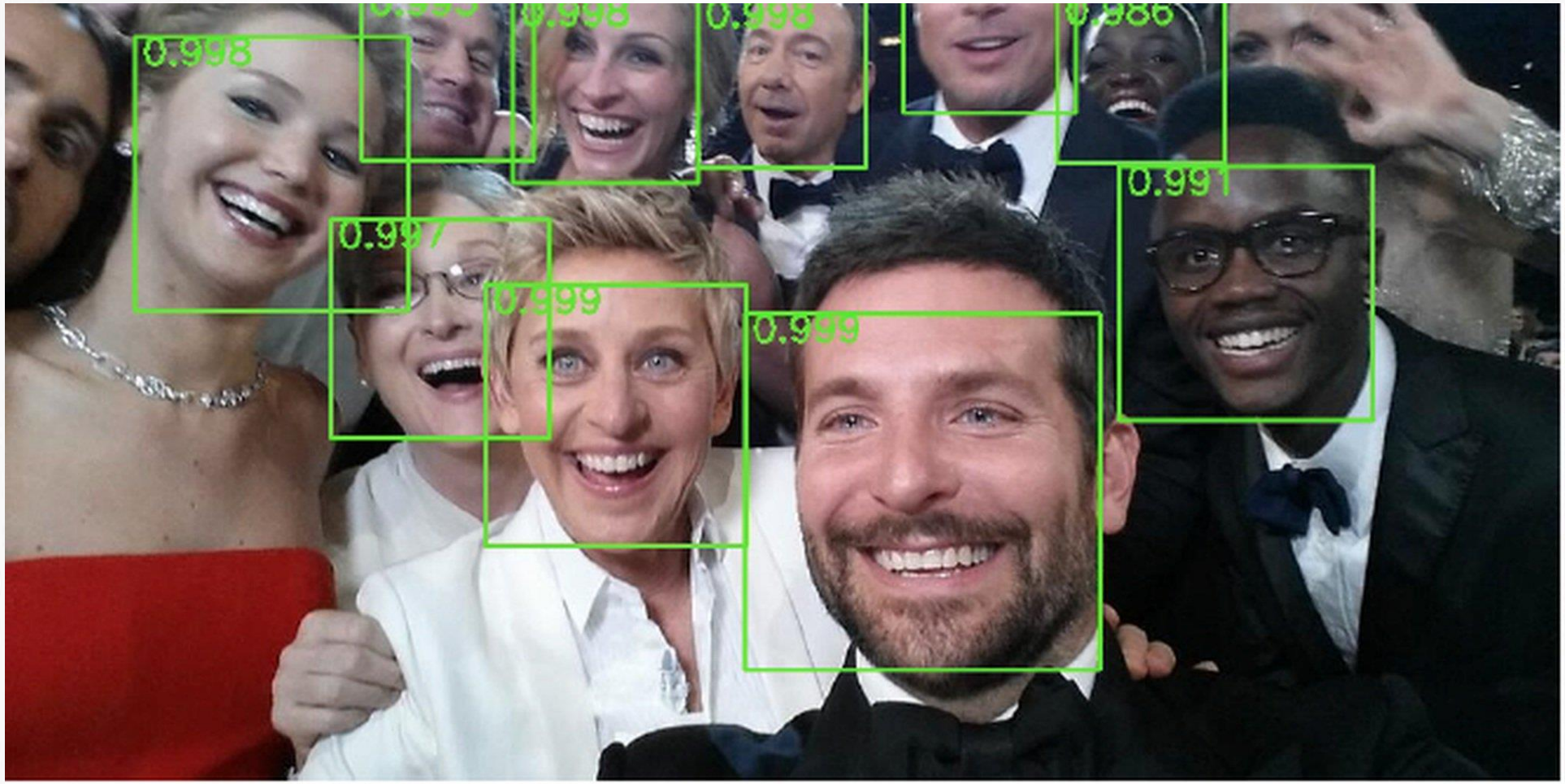
Slide credit: Noah Snavely

Activity / Event Recognition



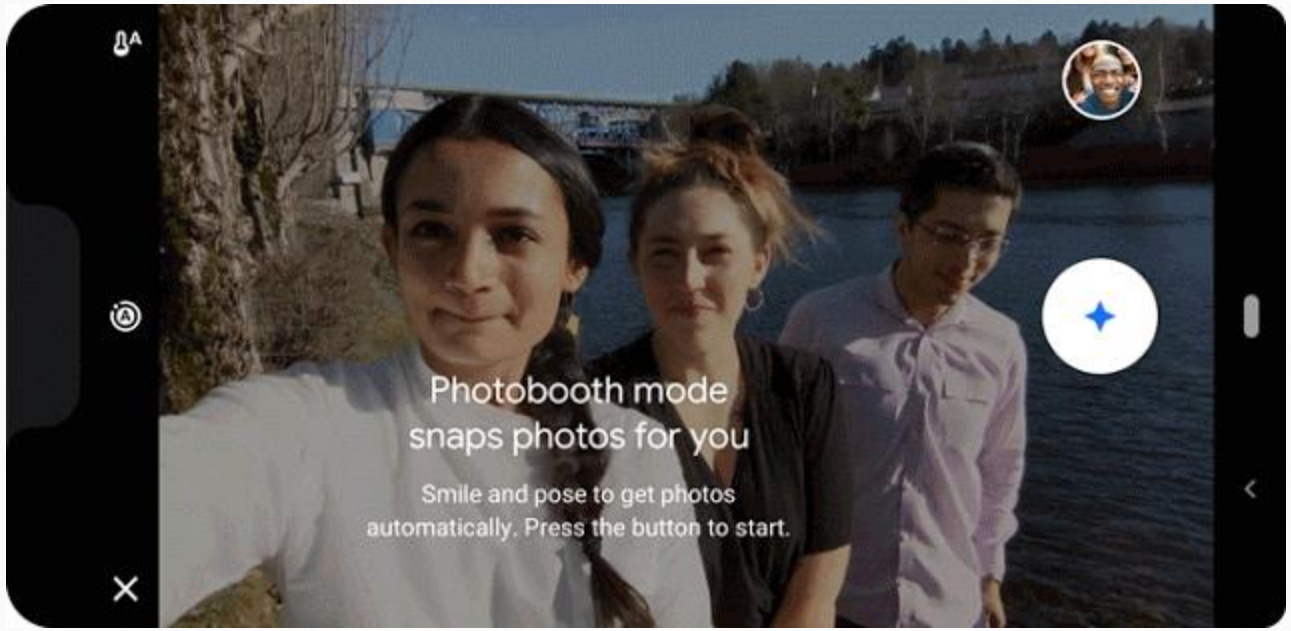
Slide credit: Noah Snavely

Applications: Photography



Slide credit: Noah Snavely

Applications: Shutter-free Photography



Take Your Best Selfie Automatically, with Photobooth on Pixel 3

https://ai.googleblog.com/2019/04/take-your-best-selfie-automatically.html

(Also features “kiss detection”)

Slide credit: Noah Snavely

Applications:
Assisted / autonomous driving

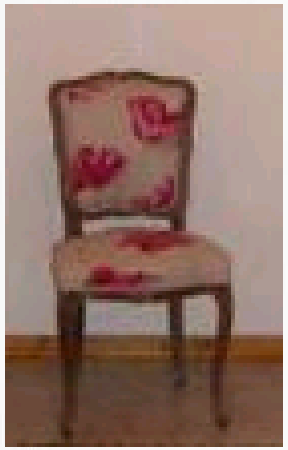
Slide credit: Noah Snavely

OLD SCHOOL OBJECT DETECTION

- Image segmentation
  - Segmentation and measuring the properties
  - Not for complicated images
- Template analysis
  - Take a small template image
  - Go through the image and look for a match by cross-correlation
  - Not robust for rotation, scaling, change in lighting

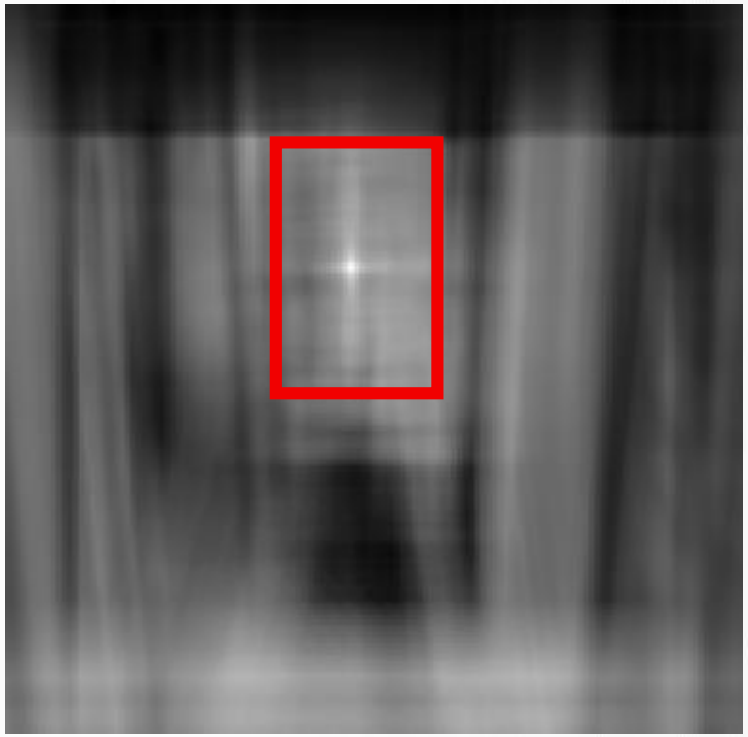
Object recognition
Is it really so hard?

This is a chair

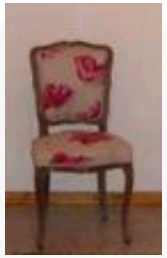


Find the chair in this image

Output of normalized correlation

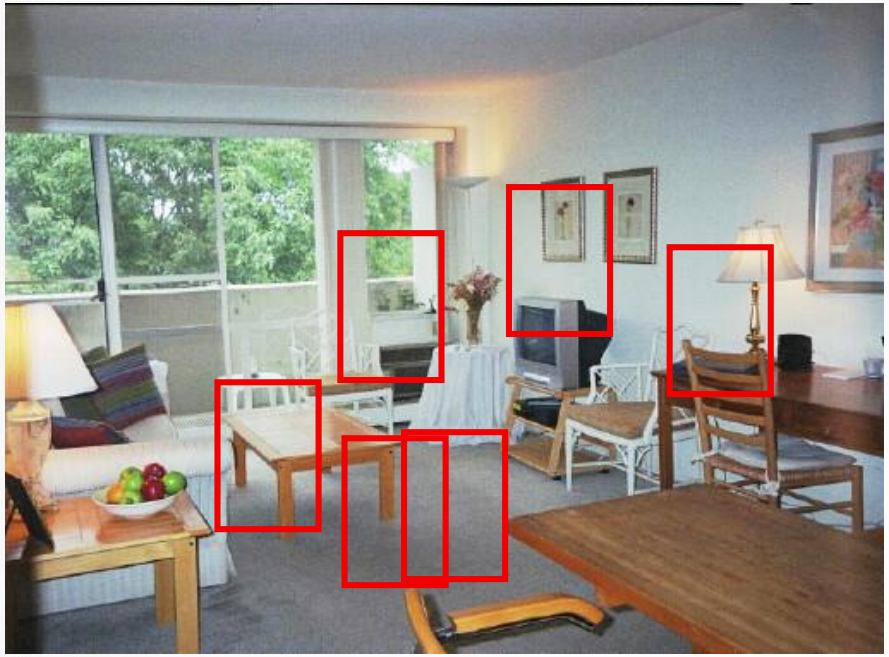


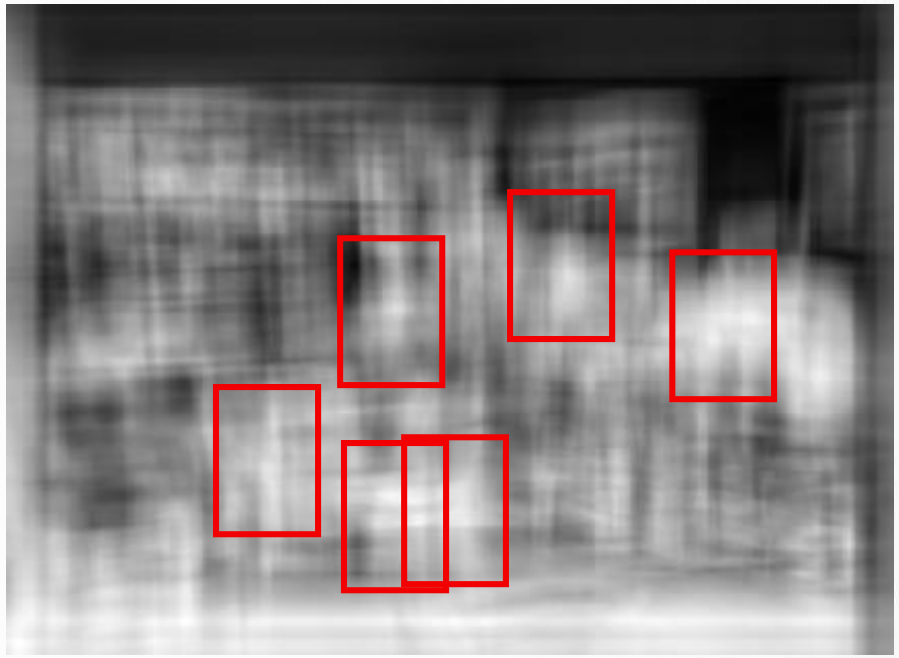
Slide credit: Noah Snavely



Object recognition
Is it really so hard?

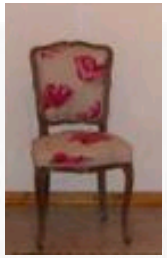
Find the chair in this image





Pretty much garbage
Simple template matching is not going to do the trick

Slide credit: Noah Snavely



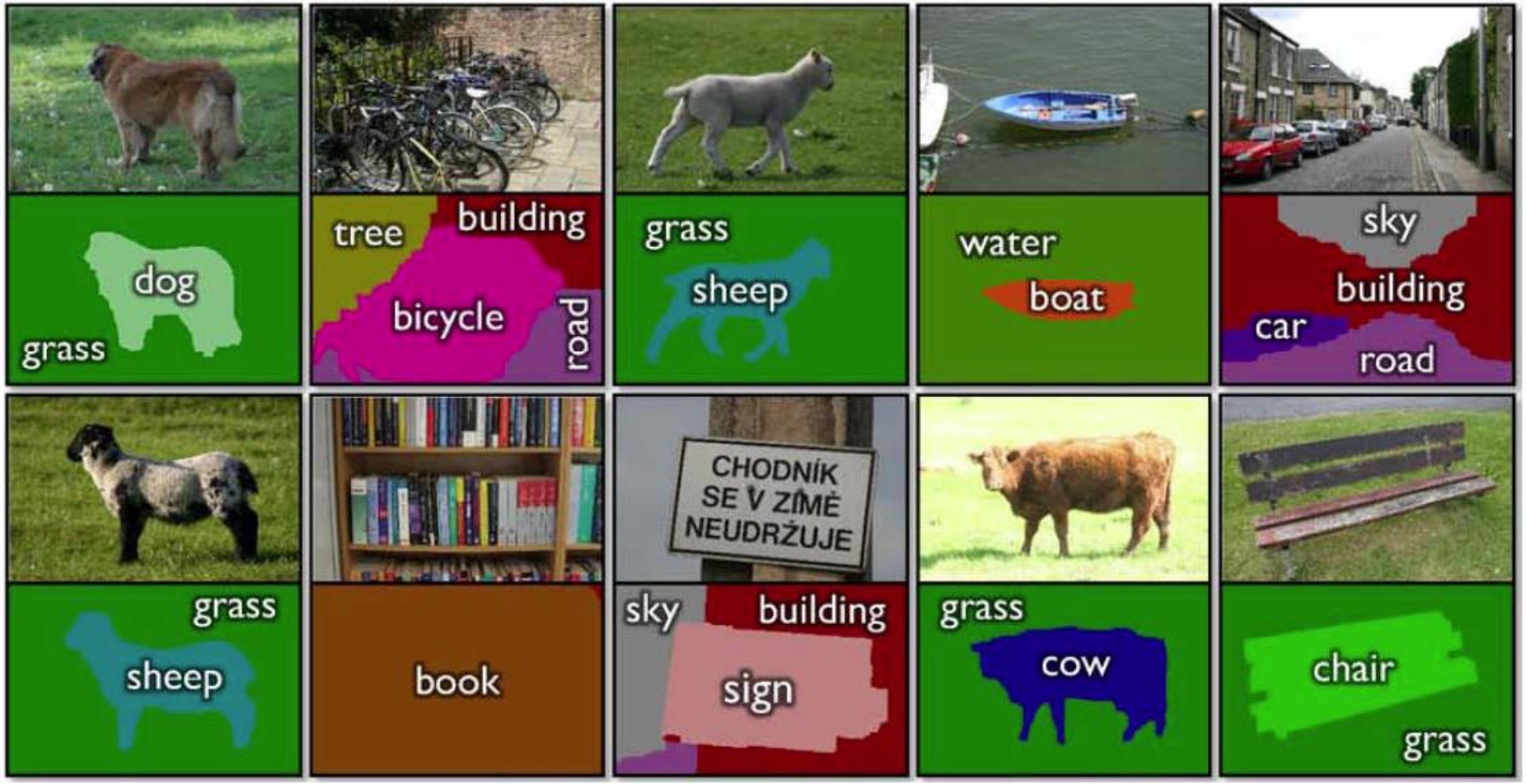
Object recognition
Is it really so hard?

Find the chair in this image

A “popular method is that of template matching, by point to point correlation of a model pattern with the image pattern. These techniques are inadequate for three-dimensional scene analysis for many reasons, such as occlusion, changes in viewing angle, and articulation of parts.” Nivatia & Binford, 1977.

Slide credit: Noah Snavely

Simultaneous recognition, detection, and segmentation

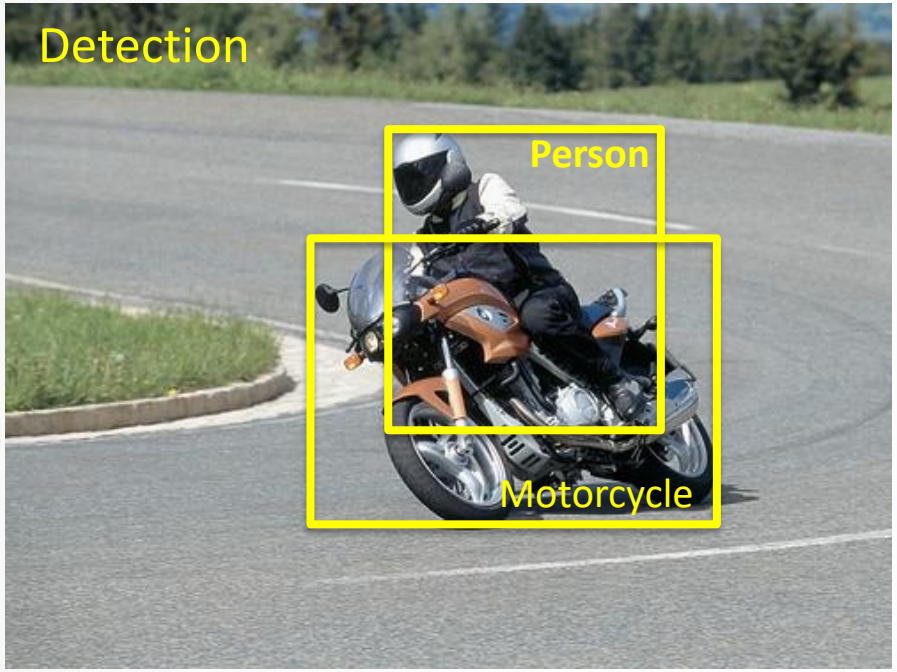


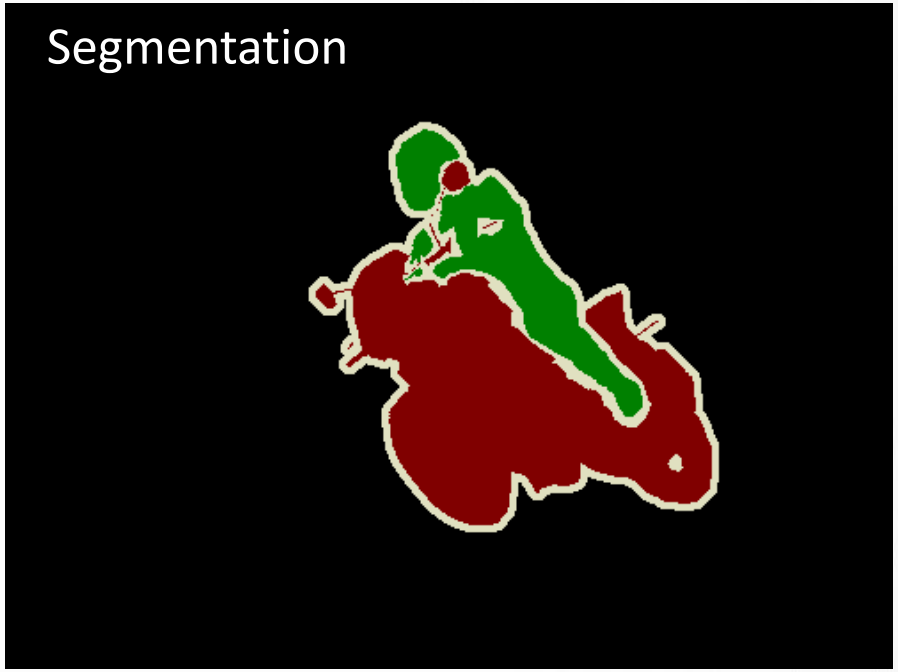
Slide credit: Noah Snavely

PASCAL VOC 2005-2012

20 object classes **22,591 images**

Classification: person, motorcycle





Action: riding bicycle

Slide credit: Noah Snavely

Everingham, Van Gool, Williams, Winn and Zisserman.
The PASCAL Visual Object Classes (VOC) Challenge. IJCV 2010.

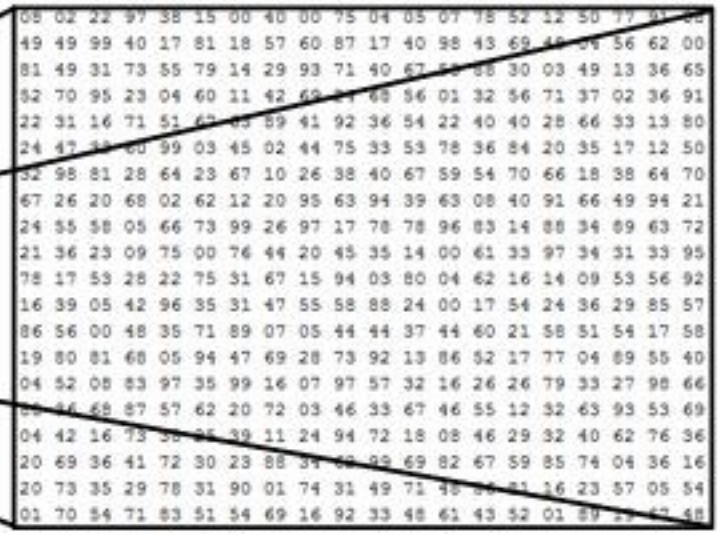
IMAGE CLASSIFICATION (IMAGE RECOGNITION)

(assume given set of discrete labels)
{dog, cat, truck, plane, ...}

cat

Slide credit: Noah Snavely

IMAGE CLASSIFICATION: PROBLEM



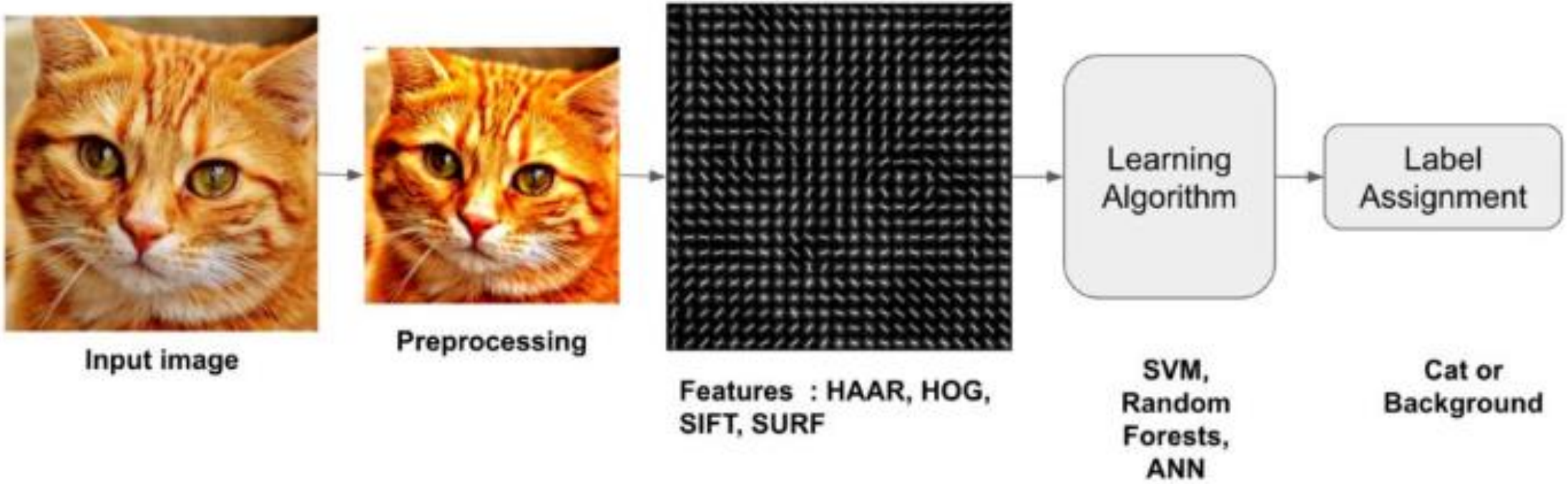
What the computer sees

image classification

82% cat
15% dog
2% hat
1% mug

Slide credit: Noah Snavely

IMAGE CLASSIFIER



https://www.learnopencv.com/image-recognition-and-object-detection-part1/

PREPROCESSING THE IMAGE

- Image is pre-processed to normalize contrast and brightness
- The suitable filetring method depends on the situation
- Image cropped and resized to a fixed size
- Feature detection
  - edge detection is used sometimes, but e.g. SIFT (and its variants), give much better performance
  - HOG is a regional descriptor (not local as SIFT) and most used in object detection

OpenCV gamma correction

Why not use SIFT matching for everything?

- Works well for object *instances* (or distinctive images such as logos)

- Not great for generic object

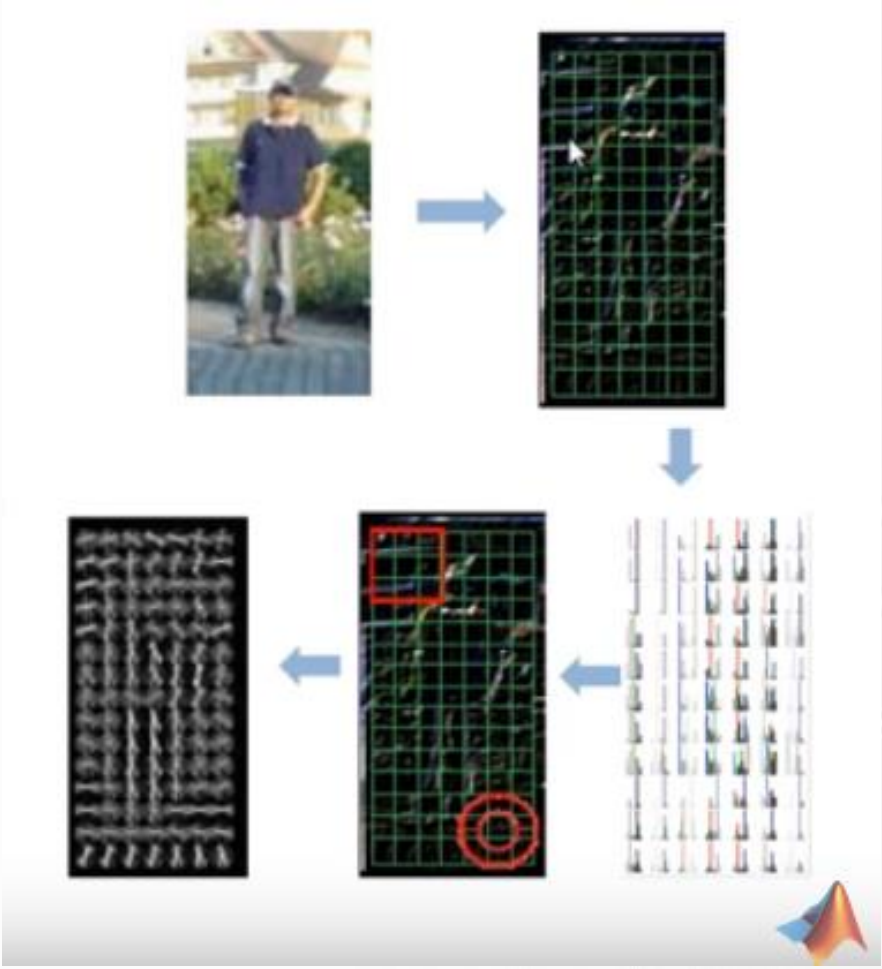
Slide credit: Noah Snavely

And it can get a lot harder

Brady, M. J., & Kersten, D. (2003). Bootstrapped learning of novel objects. J Vis, 3(6), 413-422

HISTOGRAM OF ORIENTED GRADIENTS (HOG)

- Compute gradients of the image, e.g. using [-1, 0, 1] mask with no smoothing
- Divide into cells
- Compute a histogram of gradient orientations on each cell
- Group cells into overlapping blocks, normalize vector of histogram values
- Compute that for e.g. 1000 images and use those for training a classifier



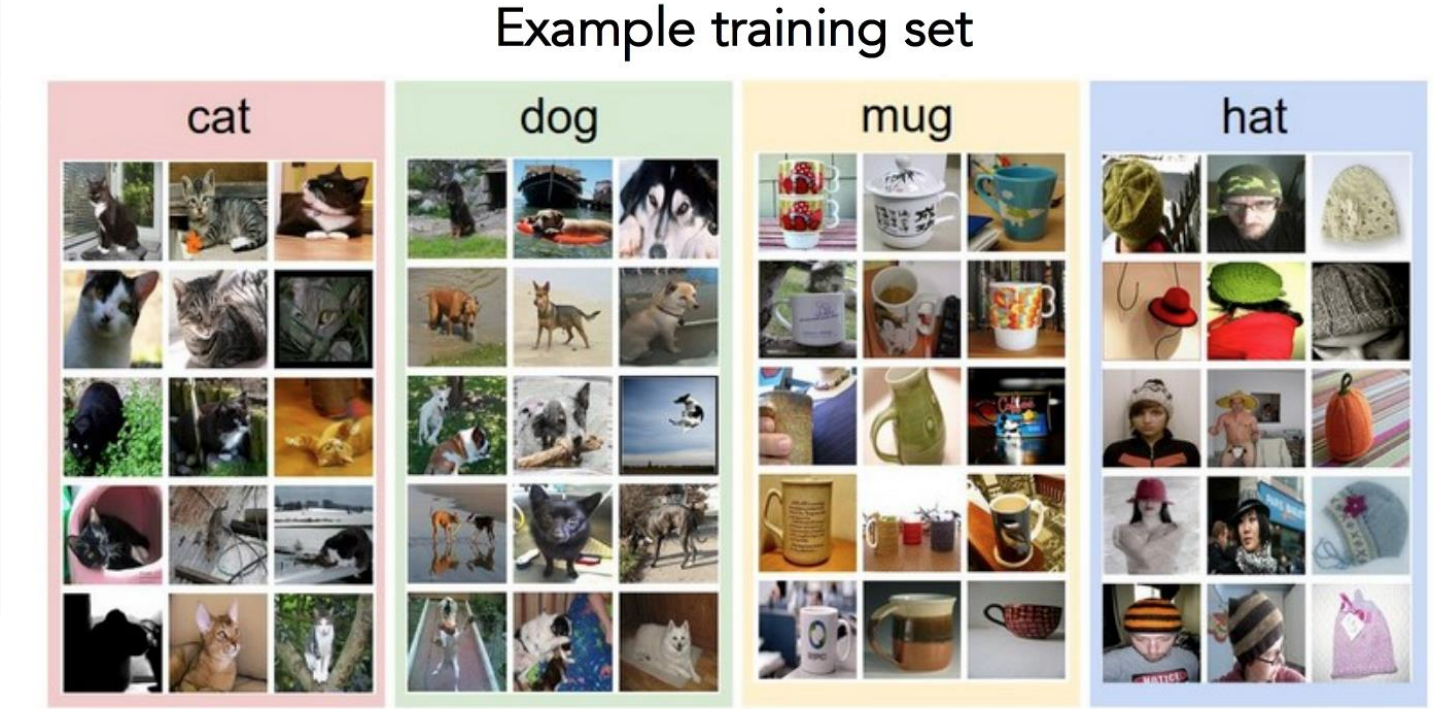
https://www.youtube.com/watch?v=9_6utqvsCtA

DATA-DRIVEN APPROACH

Collect a database of images with labels

Use ML to train an image classifier

Evaluate the classifier on test images



DATA-DRIVEN APPROACH

- Compute Bag-of-Visual-Words
- Support Vector Machine (SVM) was mainly used in traditional detection methods

https://www.youtube.com/watch?v=nnGijZ9vok4

CalTech6 dataset

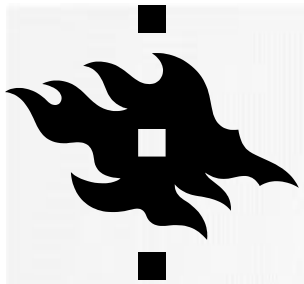





class	bag of features	bag of features	Parts-and-shape model
	Zhang et al. (2005)	Willamowski et al. (2004)	Fergus et al. (2003)
airplanes	**98.8**	97.1	90.2
cars (rear)	98.3	**98.6**	90.3
cars (side)	**95.0**	87.3	88.5
faces	**100**	99.3	96.4
motorbikes	**98.5**	98.0	92.5
spotted cats	**97.0**	—	90.0

Works pretty well for image-level classification

Csurka et al. (2004), Willamowski et al. (2005), Grauman & Darrell (2005), Sivic et al. (2003, 2005)



Dictionary Learning:
Learn Visual Words using clustering

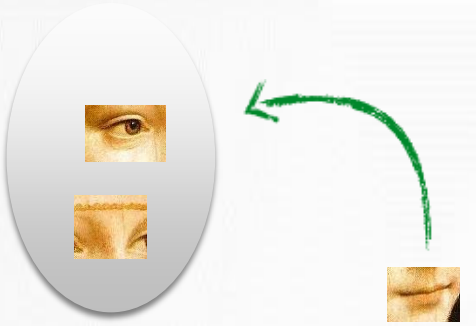
Encode:
build Bags-of-Words (BOW) vectors
for each image

Classify:
Train and test data using BOWs

EXAMPLE DICTIONARY

Appearance codebook

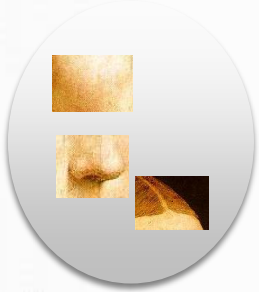
Source: B. Leibe



1. Quantization: image features gets associated to a visual word (nearest cluster center)

Encode:

build Bags-of-Words (BOW) vectors for each image

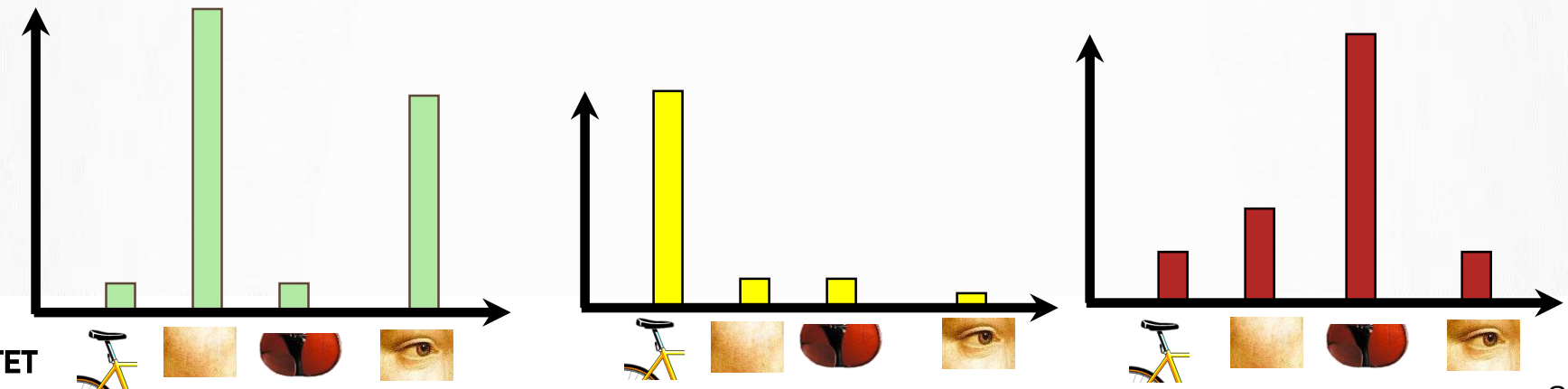


Slide credit: Noah Snavely

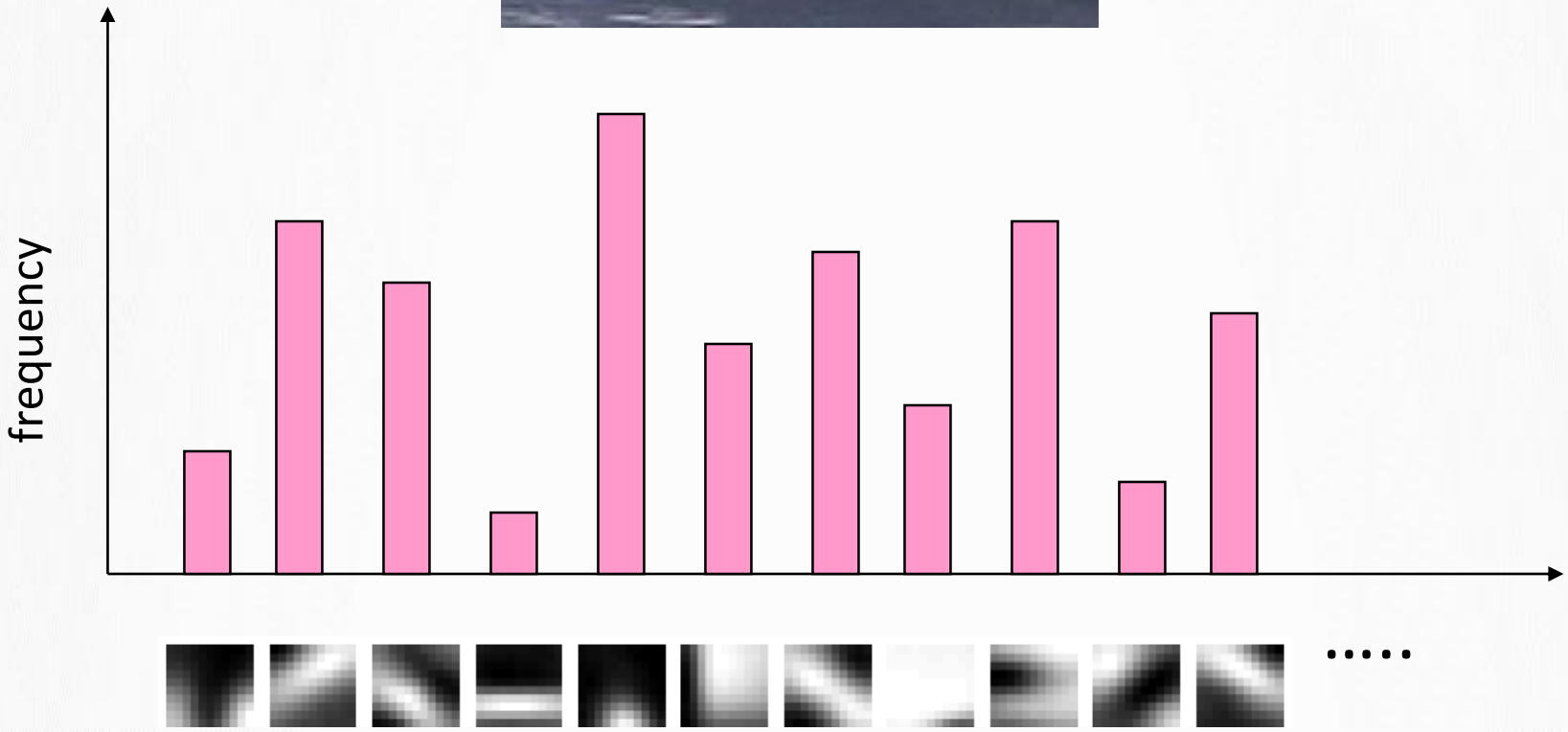
Encode:

build Bags-of-Words (BOW) vectors for each image

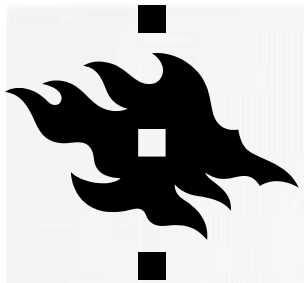
2. Histogram: count the number of visual word occurrences



Slide credit: Noah Snavely



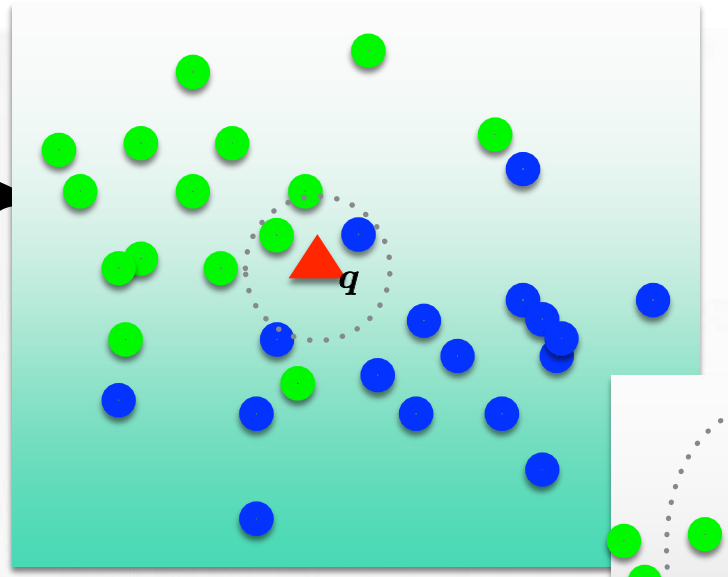
Slide credit: Noah Snavely



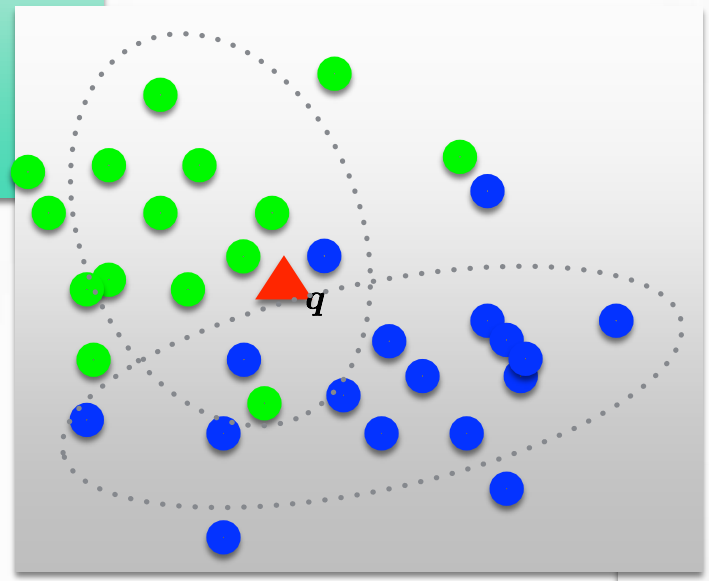
Dictionary Learning:
Learn Visual Words using clustering

Encode:
build Bags-of-Words (BOW) vectors
for each image

Classify:
Train and test data using BOWs

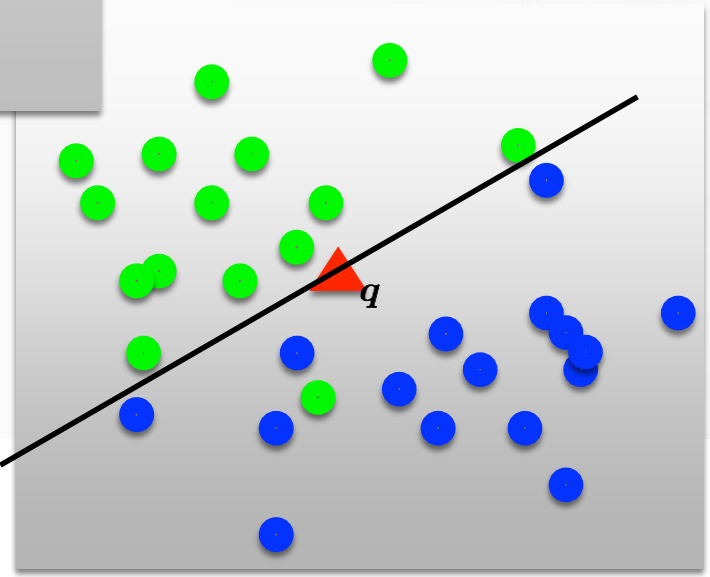


K nearest neighbors



Naïve Bayes

Support Vector Machine

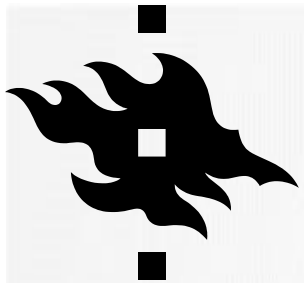


Slide credit: Noah Snavely

CIFAR-10 AND NN RESULTS

Example dataset: **CIFAR-10**
10 labels
50,000 training images
10,000 test images.

For every test image (first column), examples of nearest neighbors in rows



SUPPORT VECTOR MACHINE

IMAGE CLASSIFICATION

(assume given set of discrete labels)
{dog, cat, truck, plane, ...}

→ cat

Slide credit: Noah Snavely

SCORE FUNCTION

class scores

Slide credit: Noah Snavely

LINEAR CLASSIFIER

define a **score function**

data (histogram)

$$f(x_i, W, b) = Wx_i + b$$

class scores

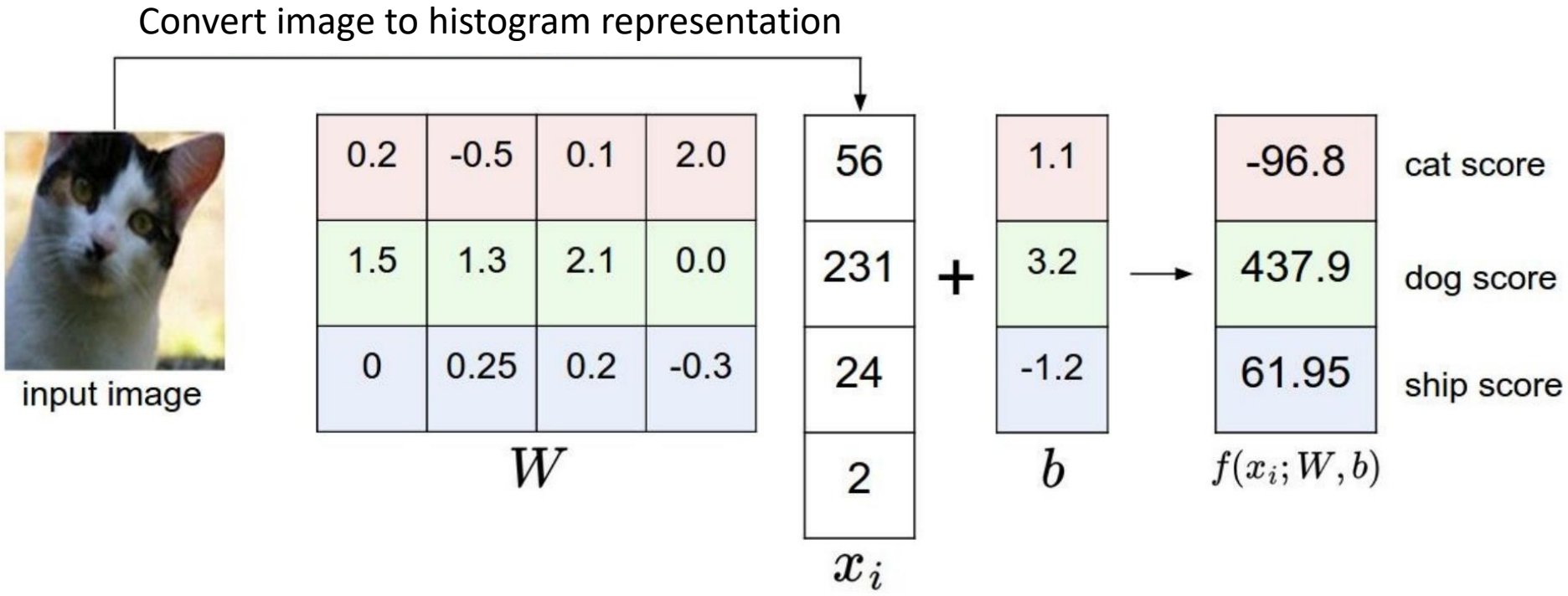
"weights"

"bias vector"

"parameters"

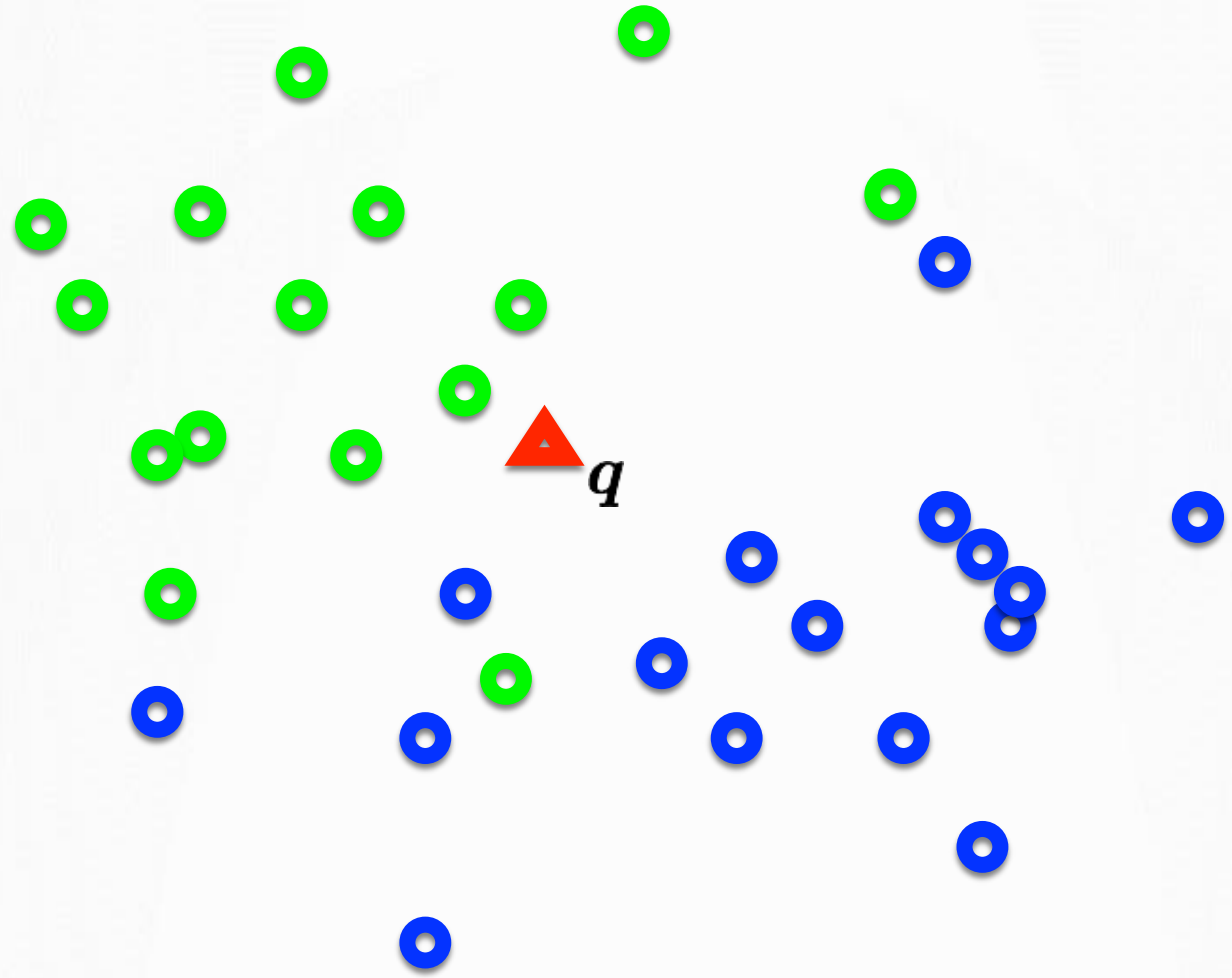
Slide credit: Noah Snavely

Example with an image with 4 pixels, and 3 classes (cat/dog/ship)



Slide credit: Noah Snavely

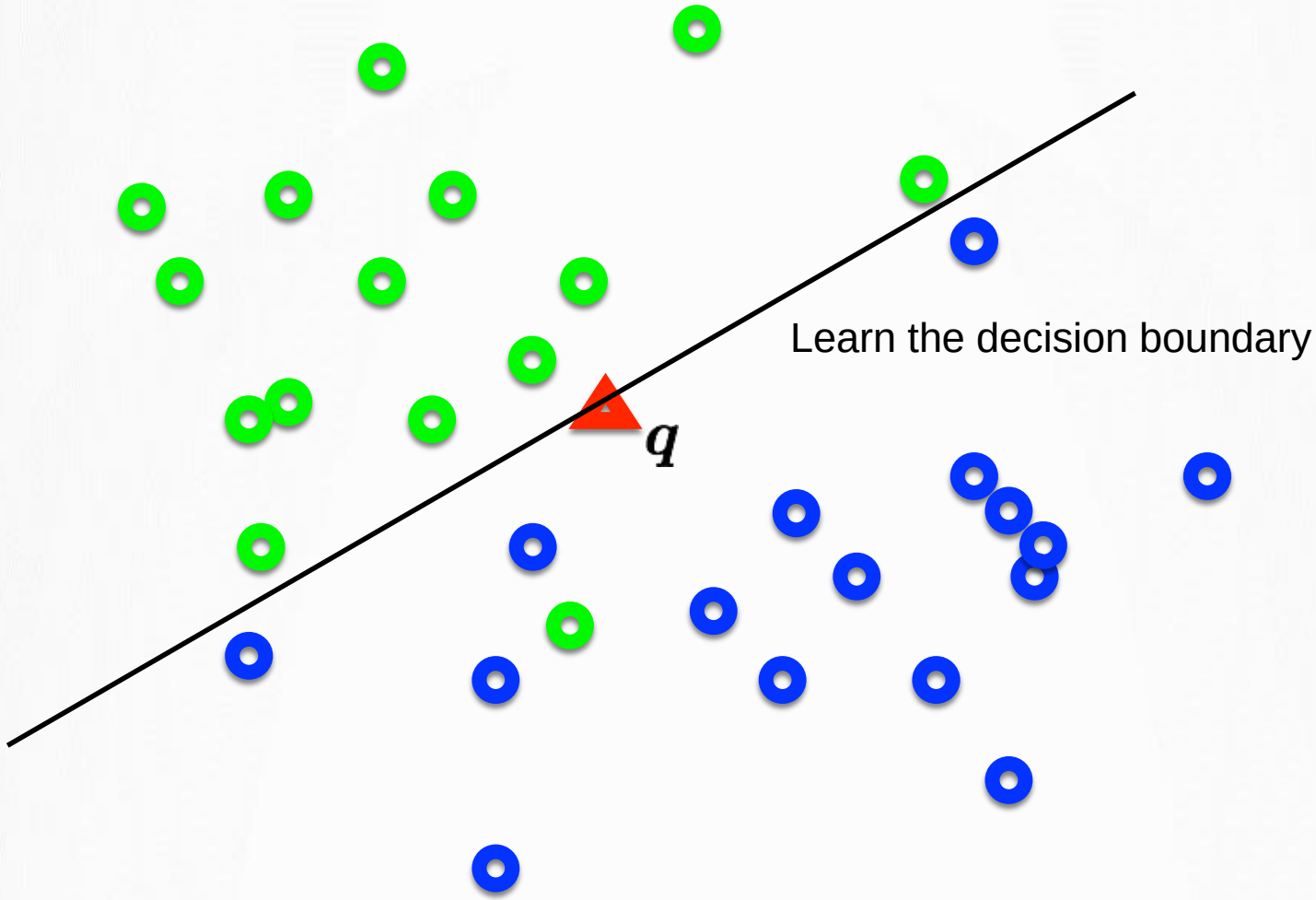
Distribution of data from two classes



Which class does q belong too?

Slide credit: Noah Snavely

Distribution of data from two classes

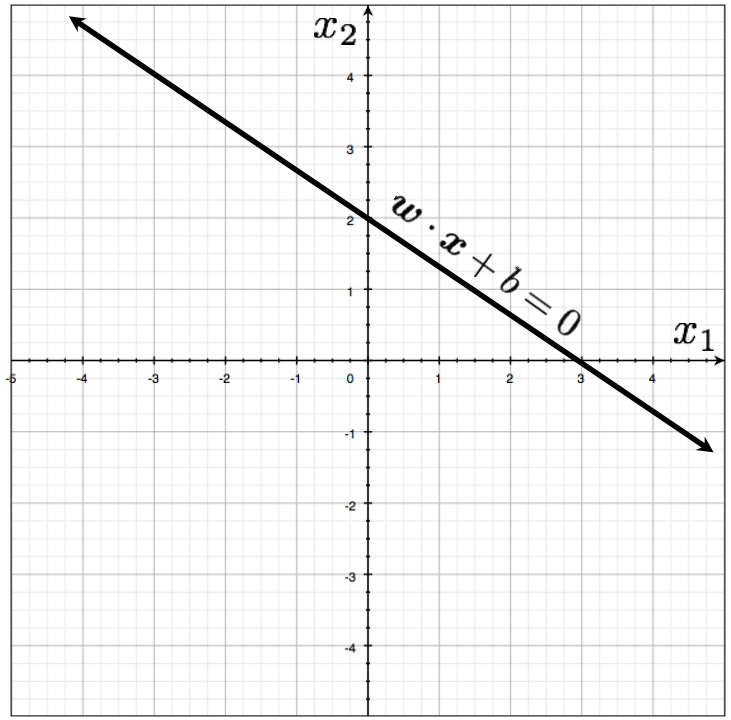


Slide credit: Noah Snavely

Hyperplanes (lines) in 2D

$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$ (offset/bias outside) $\boldsymbol{w} \cdot \boldsymbol{x} = 0$ (offset/bias inside)

$$w_1 x_1 + w_2 x_2 + b = 0$$



Important property:

Free to choose any normalization of w

The line

$$w_1 x_1 + w_2 x_2 + b = 0$$

and the line

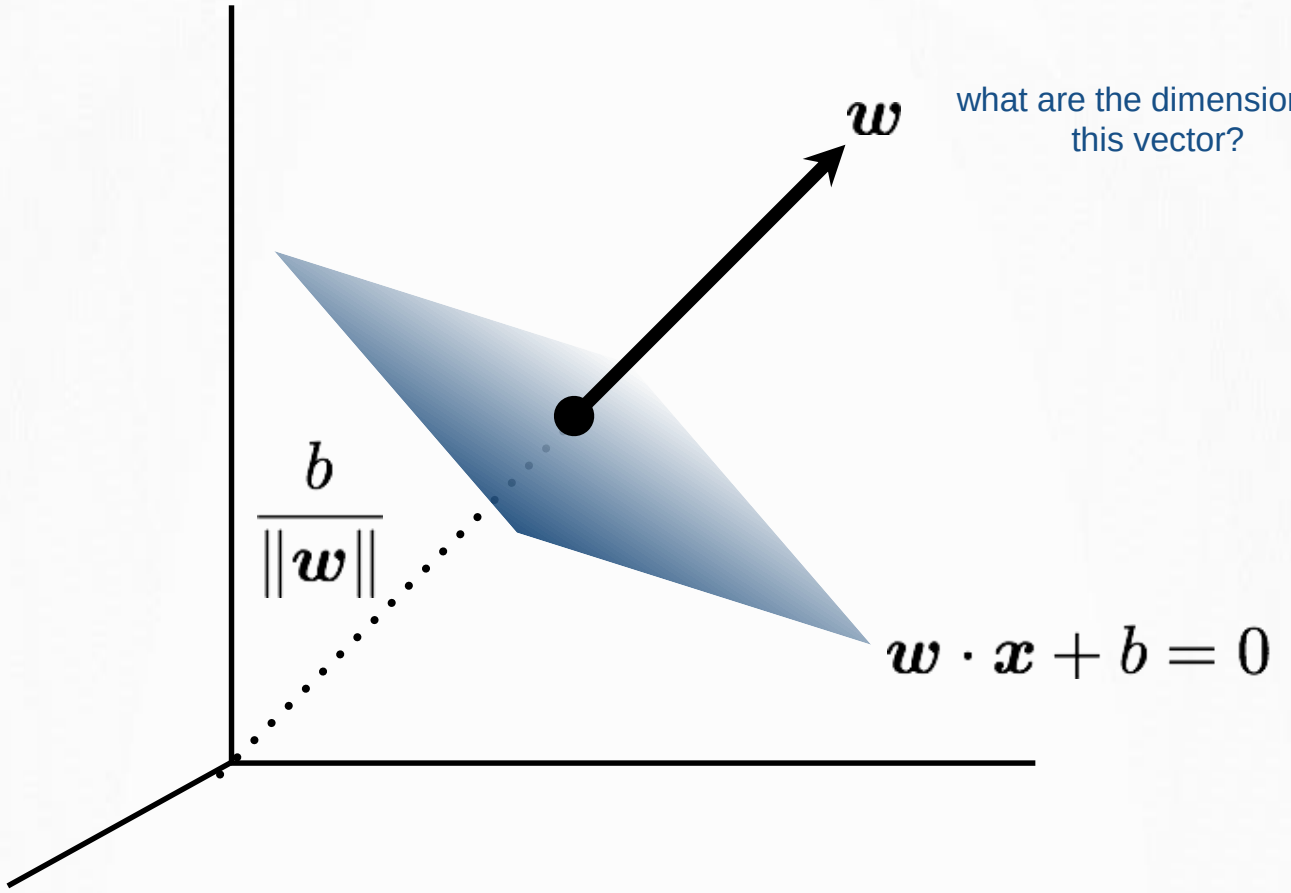
$$\lambda(w_1 x_1 + w_2 x_2 + b) = 0$$

define the same line

Slide credit: Noah Snavely

Now we can go to 3D …

Hyperplanes (planes) in 3D

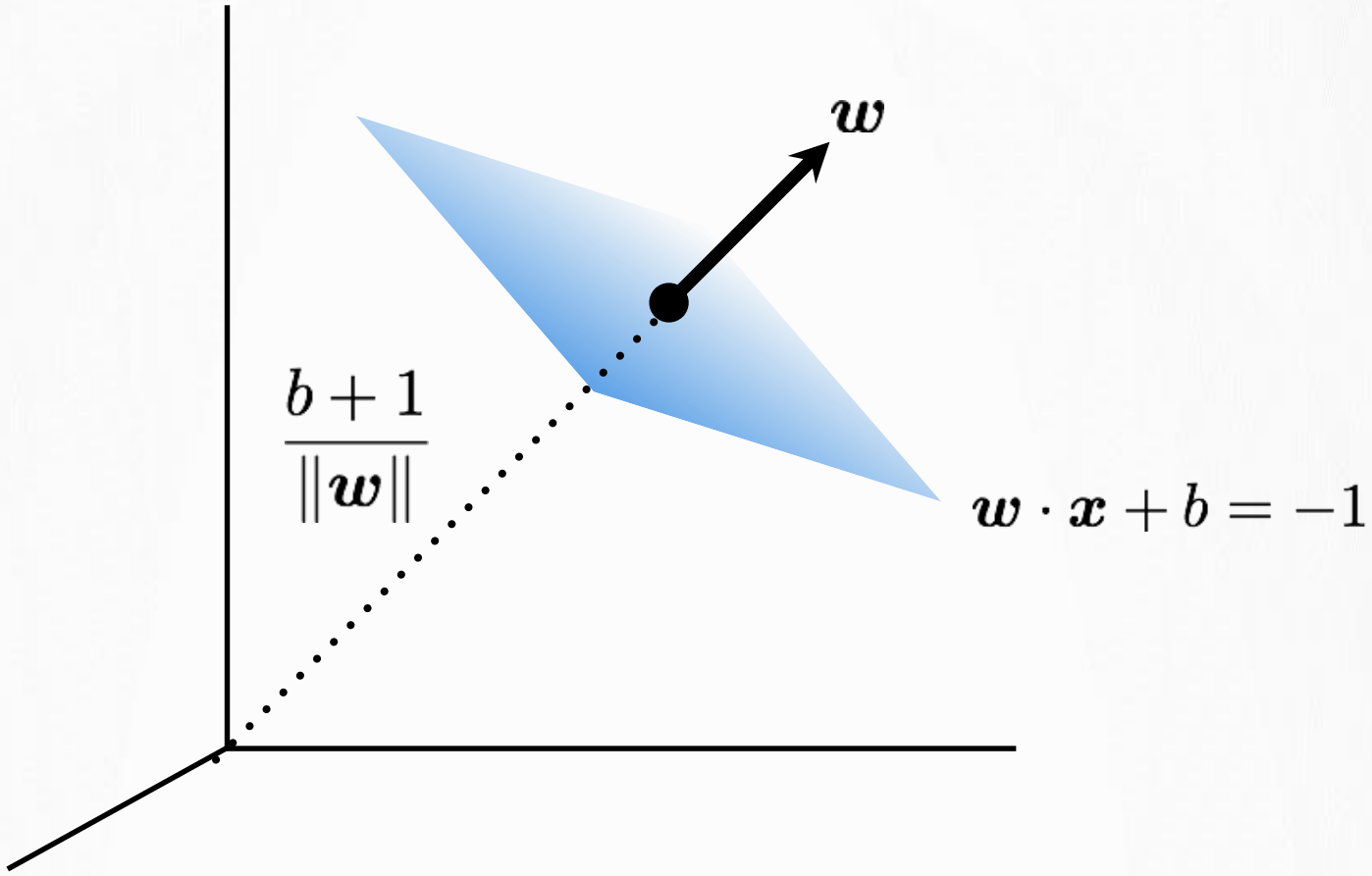


what are the dimensions of this vector?

*What happens if you change **b**?*

Slide credit: Noah Snavely

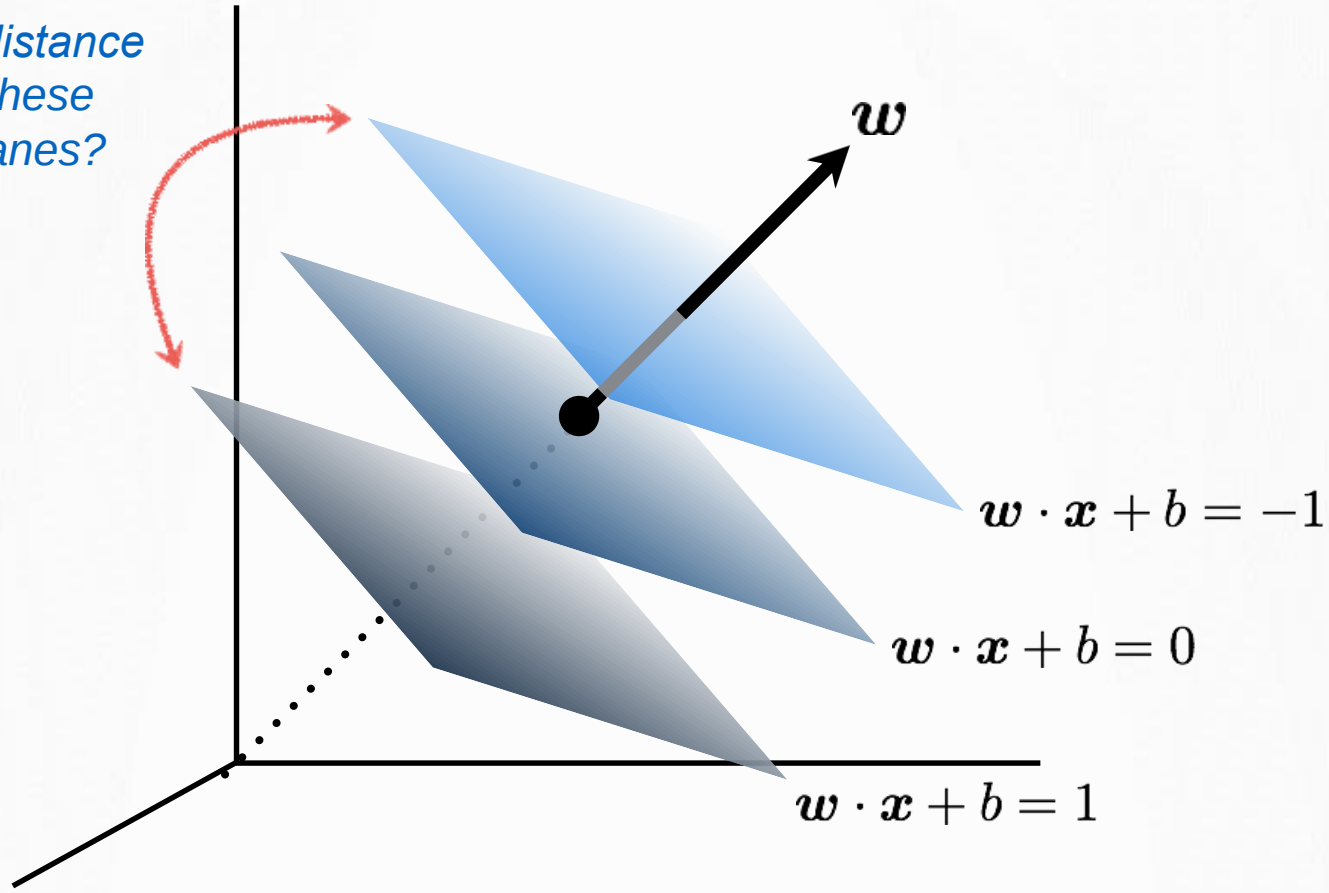
Hyperplanes (planes) in 3D



Slide credit: Noah Snavely

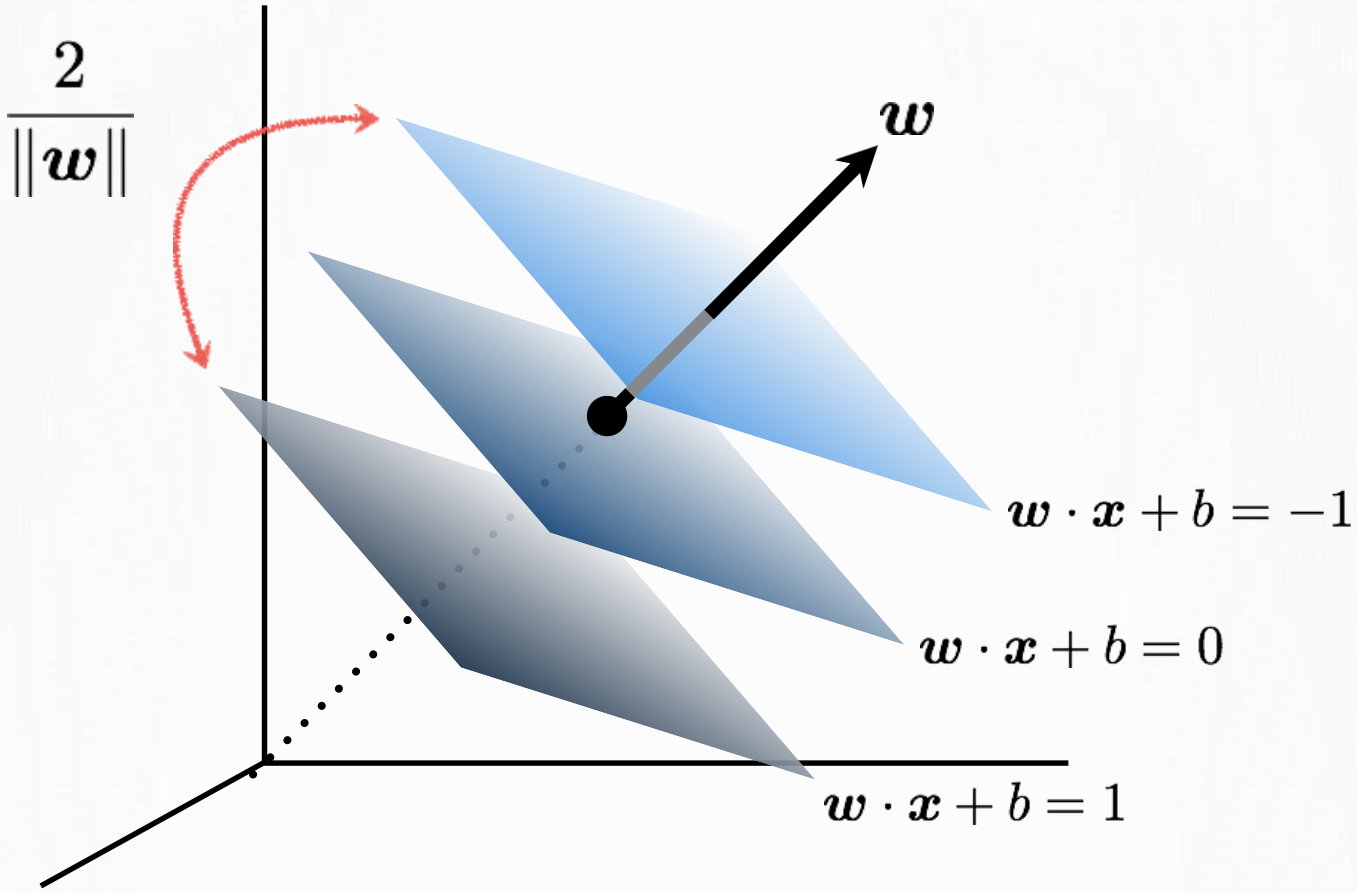
Hyperplanes (planes) in 3D

What's the distance between these parallel planes?



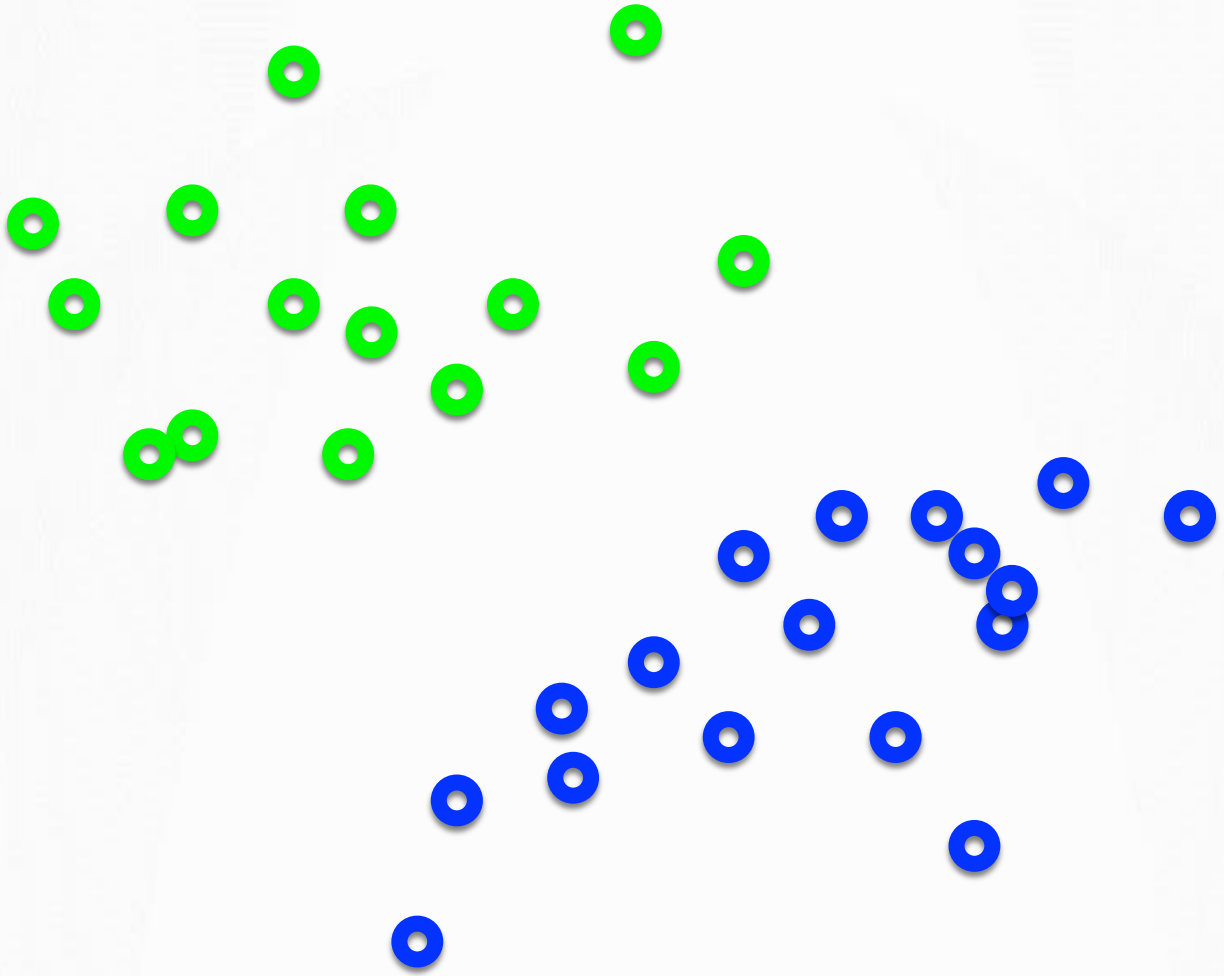
Slide credit: Noah Snavely

Hyperplanes (planes) in 3D



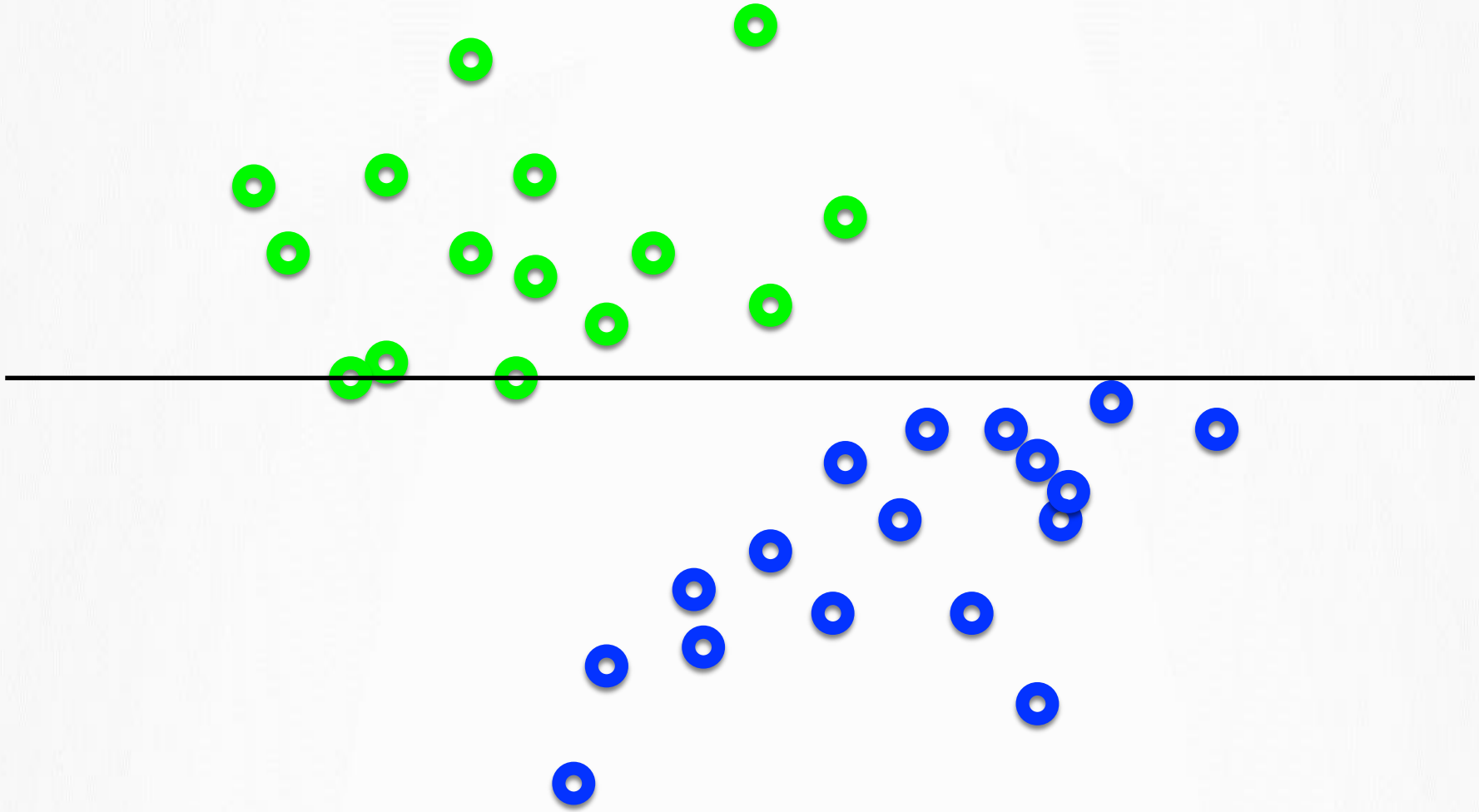
Slide credit: Noah Snavely

What's the best **w**?



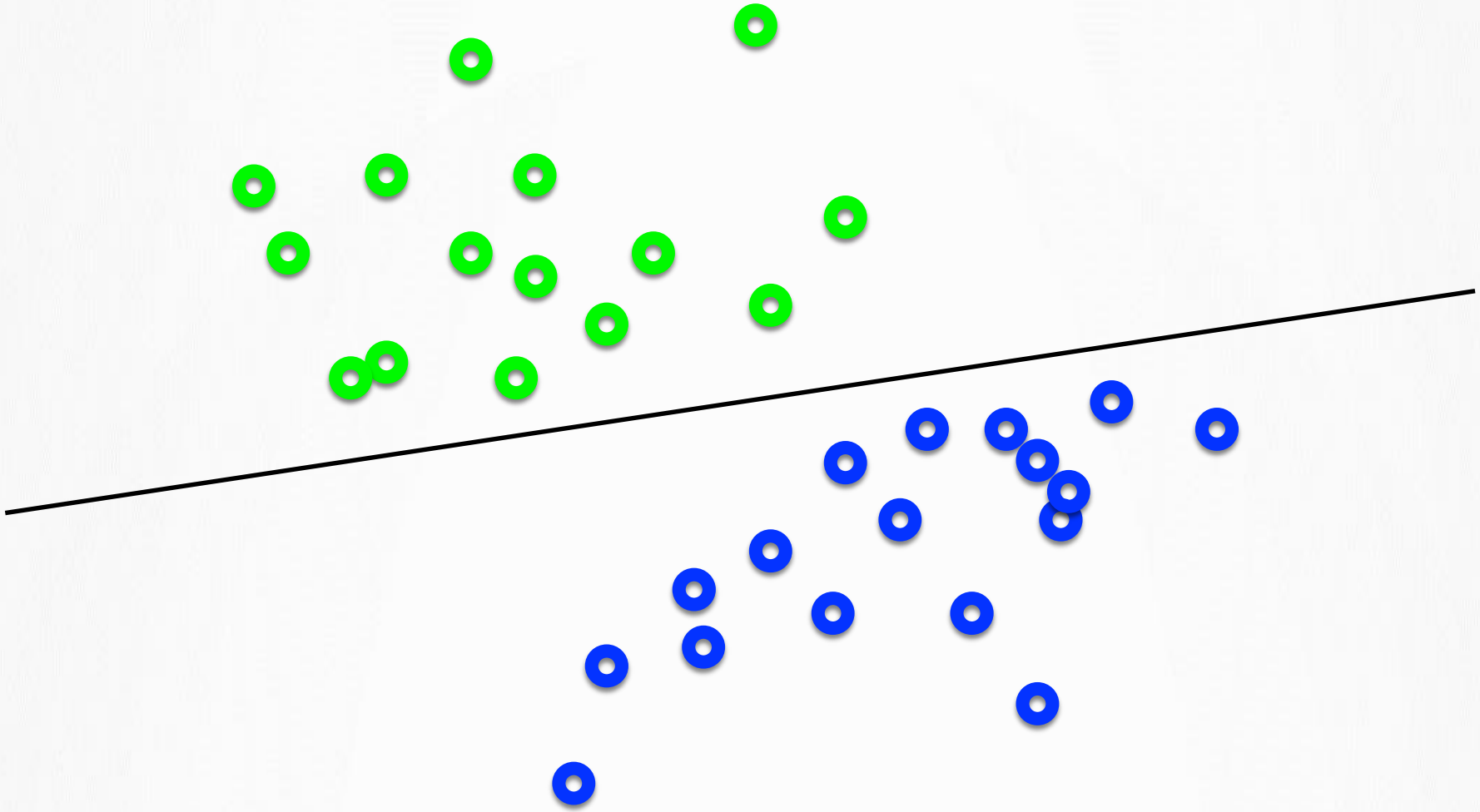
Slide credit: Noah Snavely

What's the best **w**?



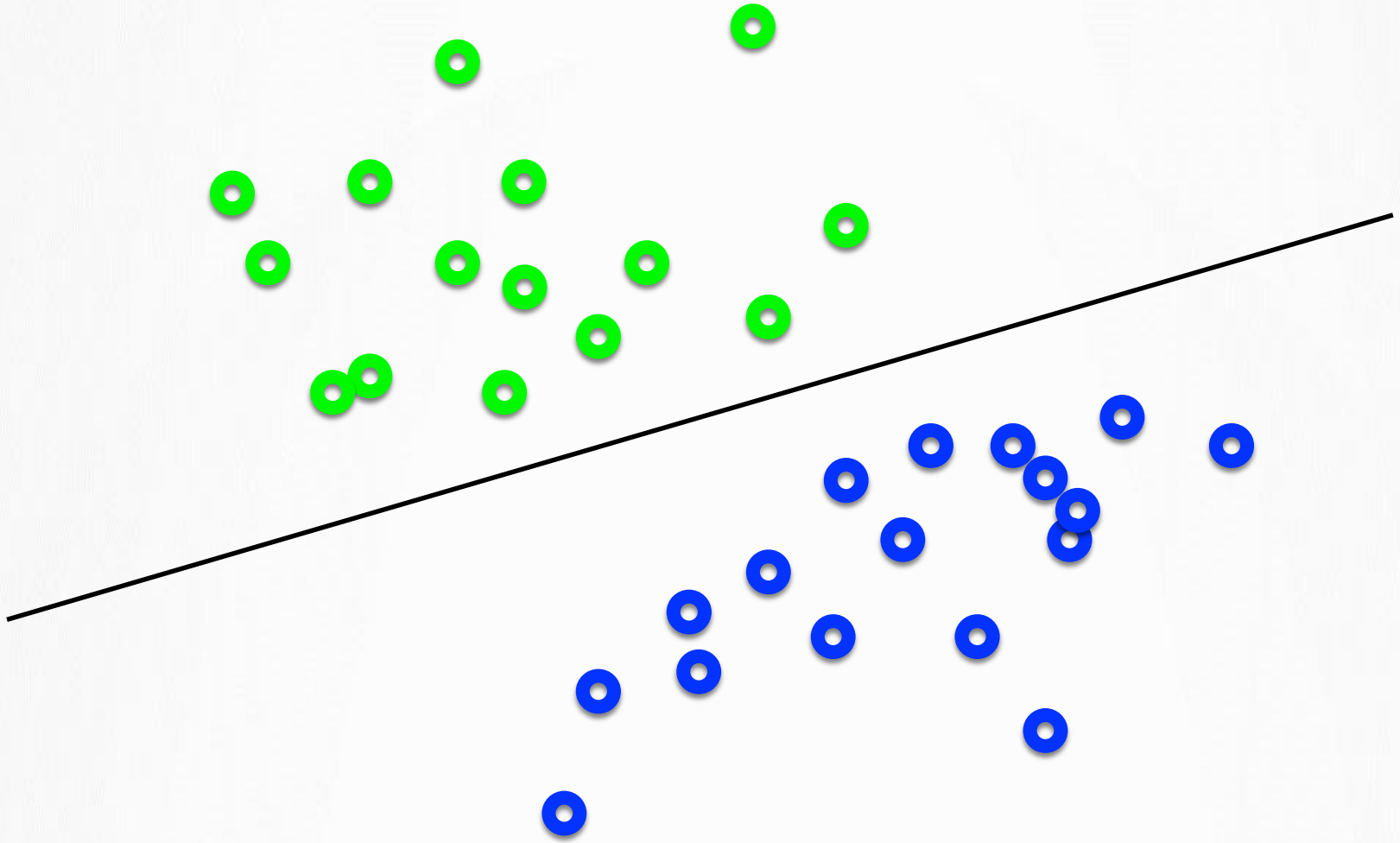
Slide credit: Noah Snavely

What's the best **w**?



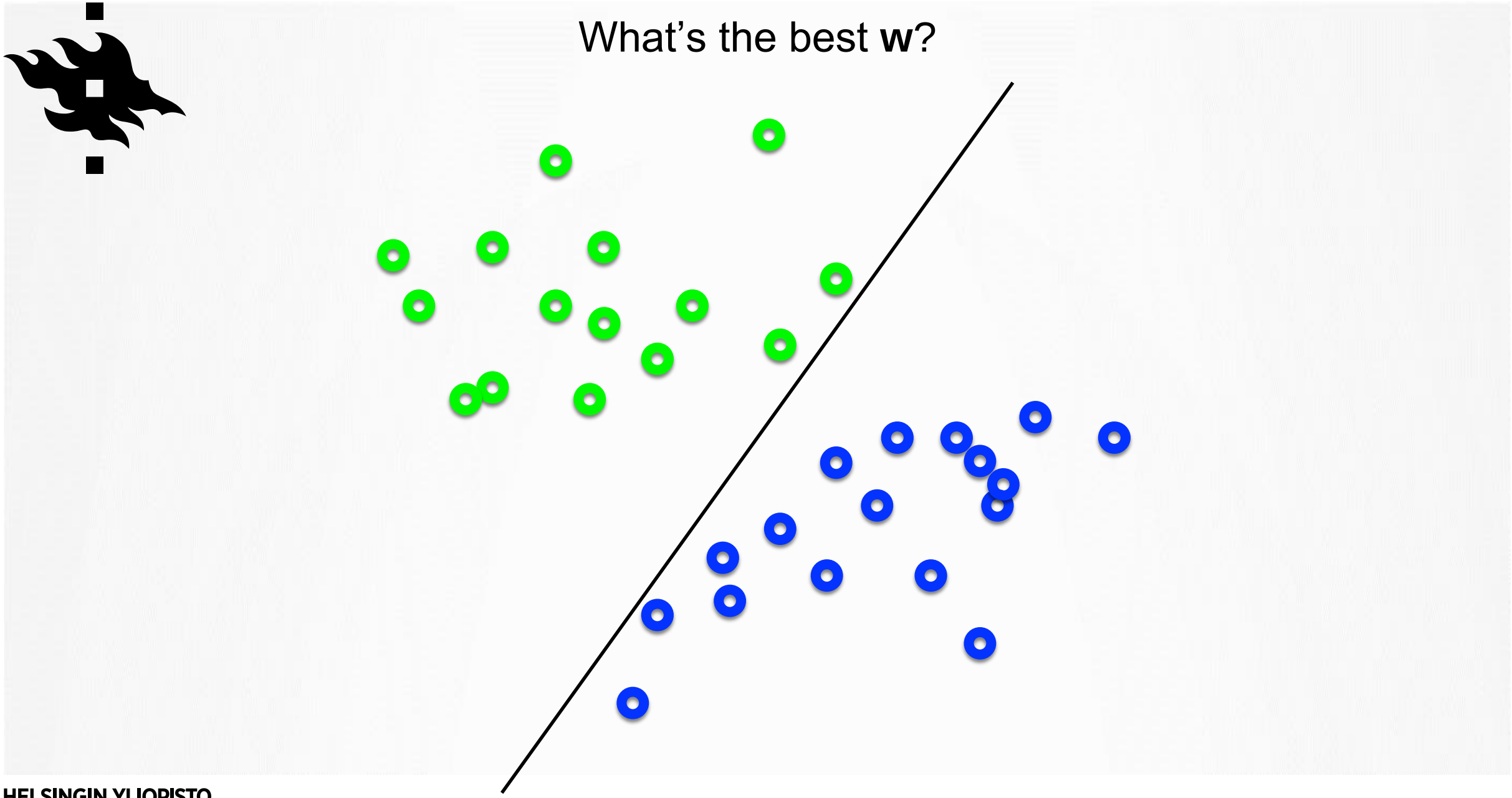
Slide credit: Noah Snavely

What's the best **w**?

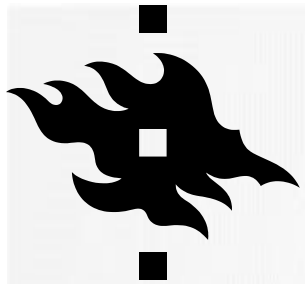


Slide credit: Noah Snavely

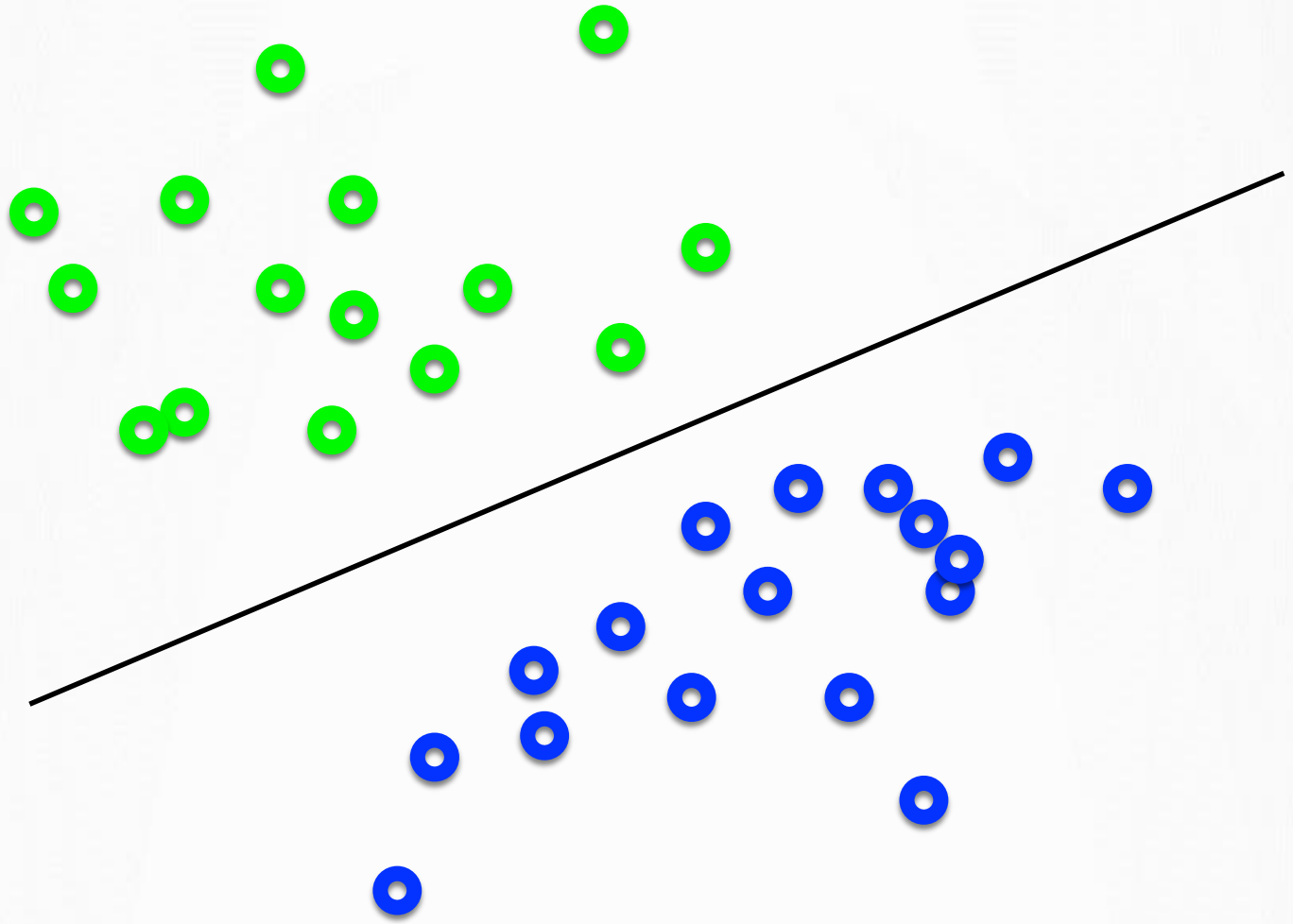
What's the best **w**?



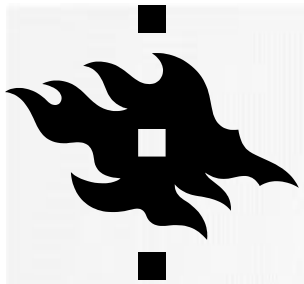
Slide credit: Noah Snavely



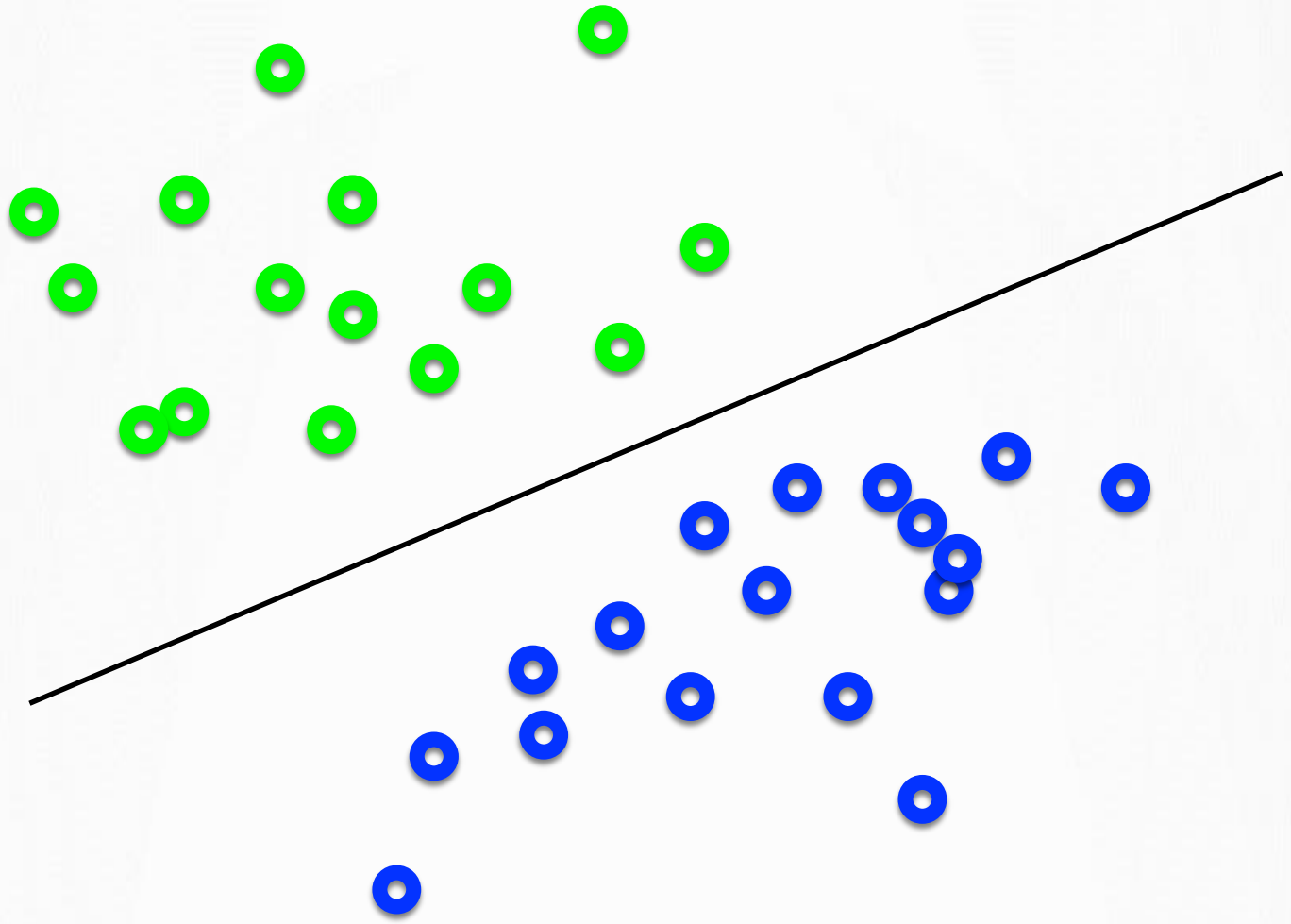
What's the best **w**?



Intuitively, the line that is the farthest from all interior points

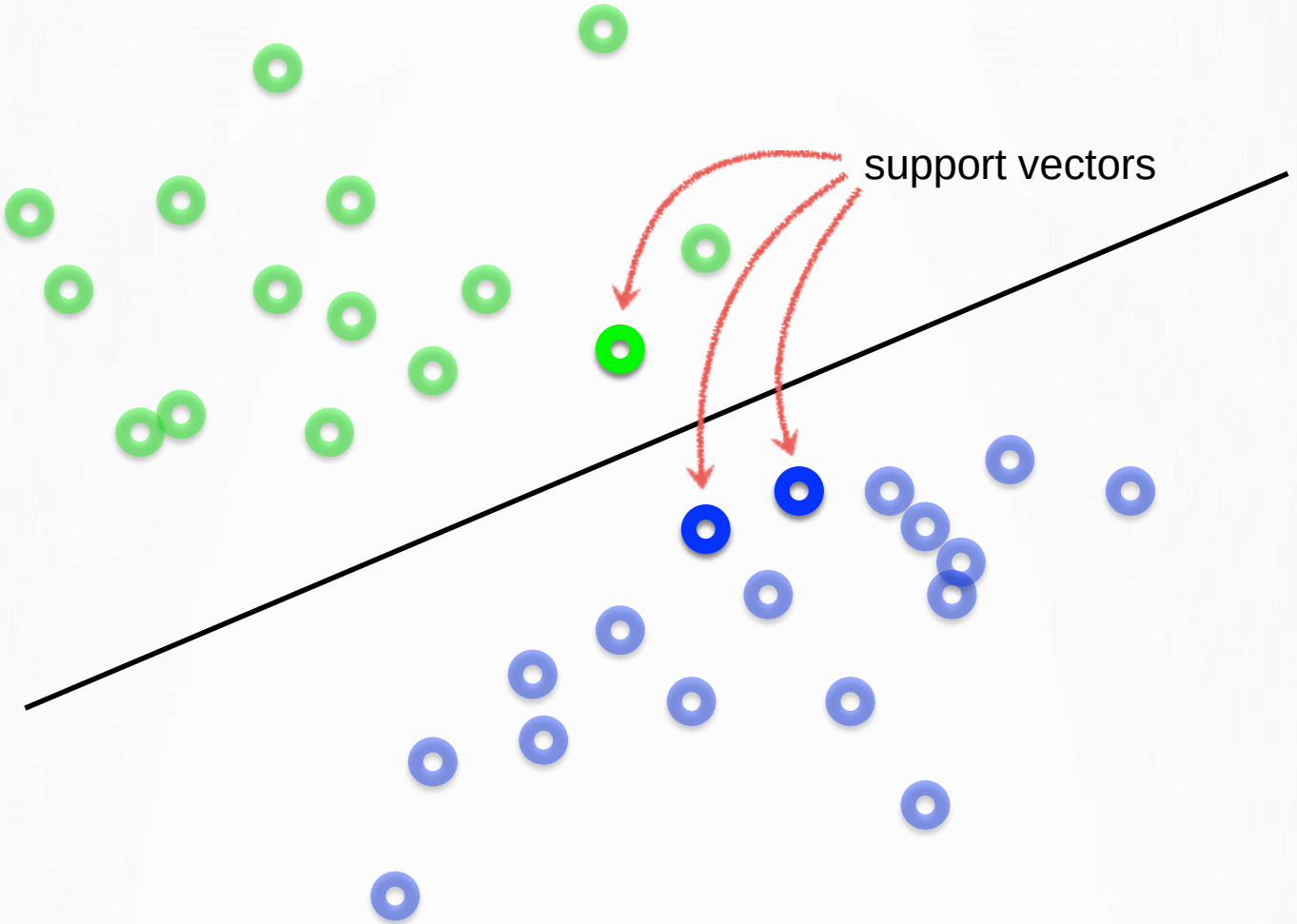


What's the best **w**?



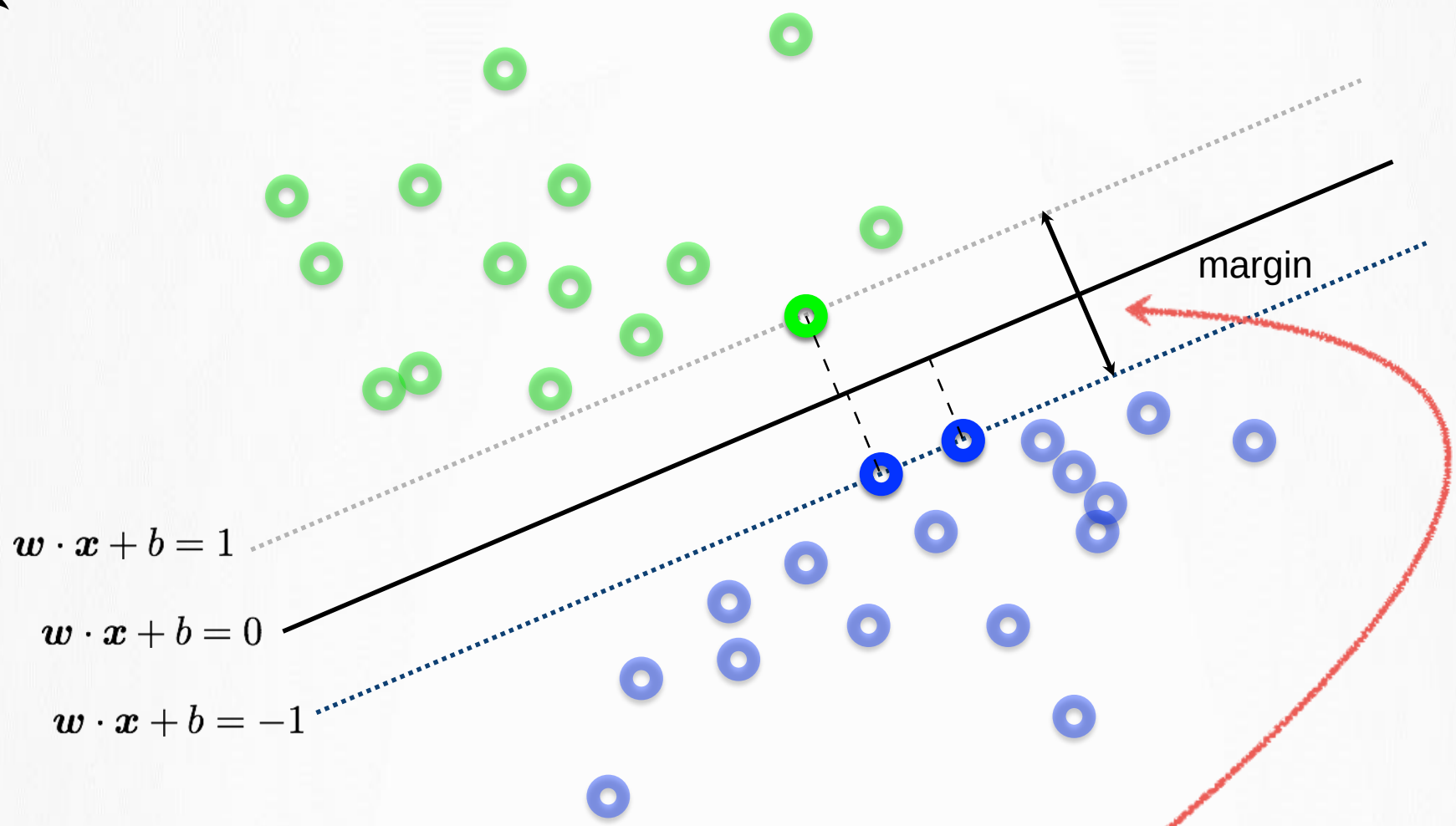
Maximum Margin solution:
most stable to perturbations of data

What's the best **w**?



Want a hyperplane that is far away from 'inner points'

Find hyperplane **w** such that …



the gap between parallel hyperplanes $\frac{2}{\|\boldsymbol{w}\|}$ is maximized

Can be formulated as a maximization problem

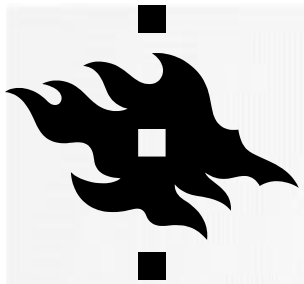
$$\max_{\boldsymbol{w}} \frac{2}{\|\boldsymbol{w}\|}$$

$$\text{subject to } \boldsymbol{w} \cdot \boldsymbol{x}_i + b \begin{array}{l} \geq +1 \text{ if } y_i = +1 \\ \leq -1 \text{ if } y_i = -1 \end{array} \text{ for } i = 1, \ldots, N$$

What does this constraint mean?

label of the data point

Slide credit: Noah Snavely



Can be formulated as a maximization problem

$$\max_{\boldsymbol{w}} \frac{2}{\|\boldsymbol{w}\|}$$

$$\text{subject to } \boldsymbol{w} \cdot \boldsymbol{x}_i + b \begin{array}{l} \geq +1 \text{ if } y_i = +1 \\ \leq -1 \text{ if } y_i = -1 \end{array} \text{ for } i = 1, \ldots, N$$

Equivalently,

$$\min_{\boldsymbol{w}} \|\boldsymbol{w}\|$$

$$\text{subject to } y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 \text{ for } i = 1, \ldots, N$$

Slide credit: Noah Snavely

‘Primal formulation’ of a linear SVM

$$\min_{\boldsymbol{w}} \|\boldsymbol{w}\|$$

Objective Function

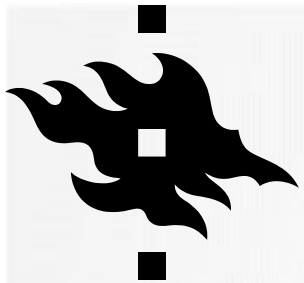
$$\text{subject to } y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1 \text{ for } i = 1, \ldots, N$$

Constraints

This is a convex quadratic programming (QP) problem

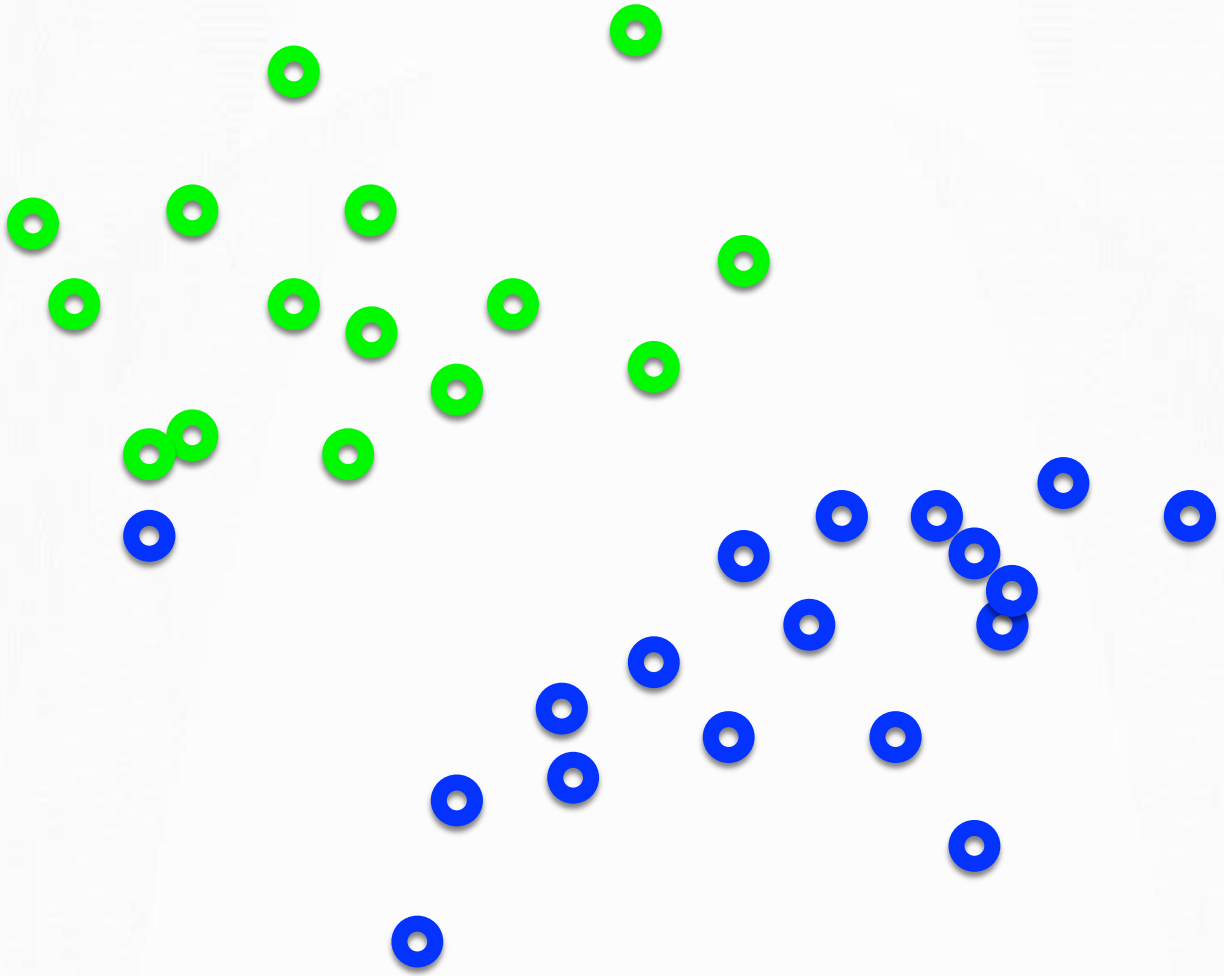
(a unique solution exists)

Slide credit: Noah Snavely



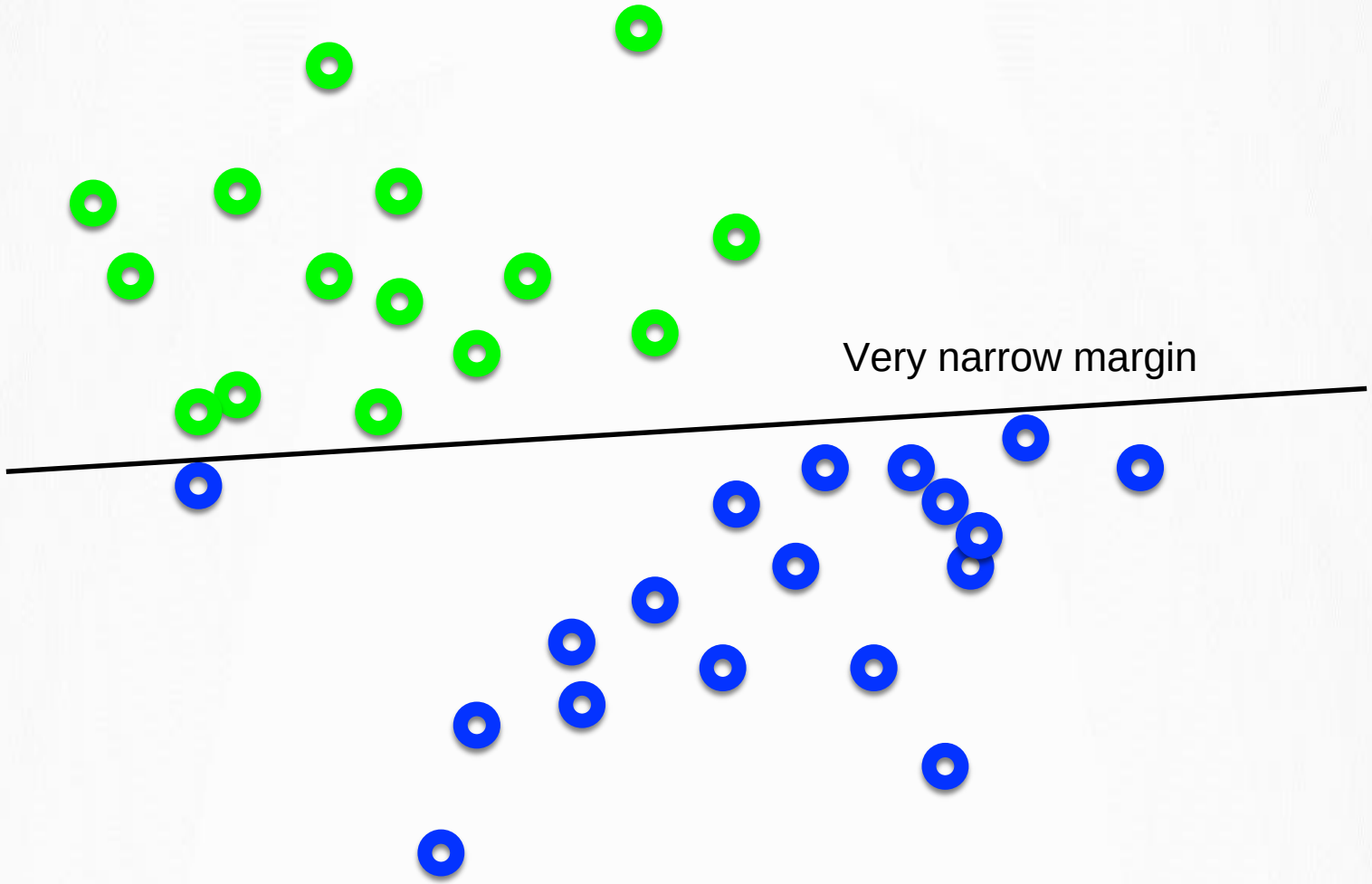
‘SOFT’ MARGIN

What's the best **w**?



Slide credit: Noah Snavely

What's the best **w**?



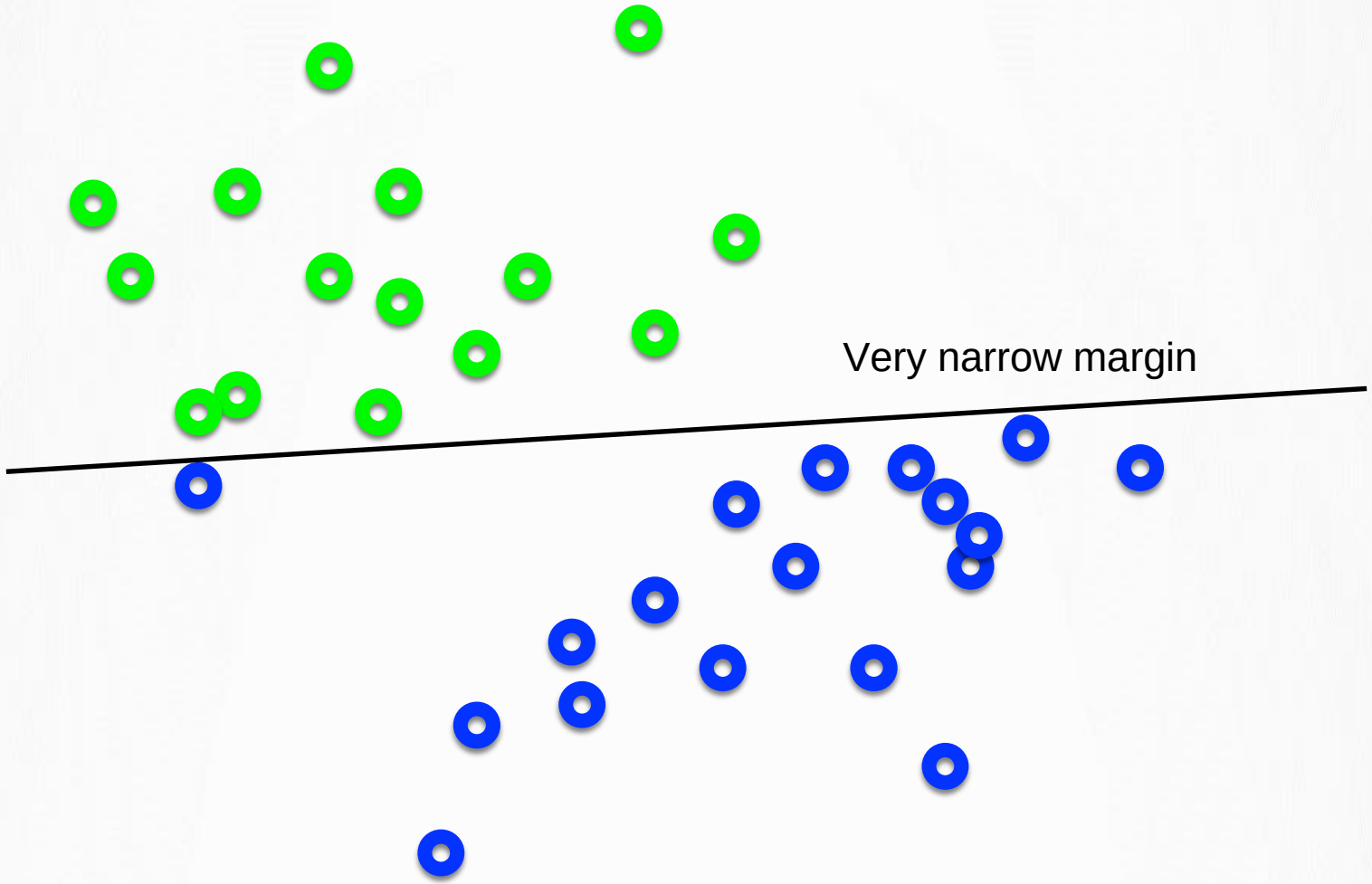
Slide credit: Noah Snavely

Separating cats and dogs

Very narrow margin

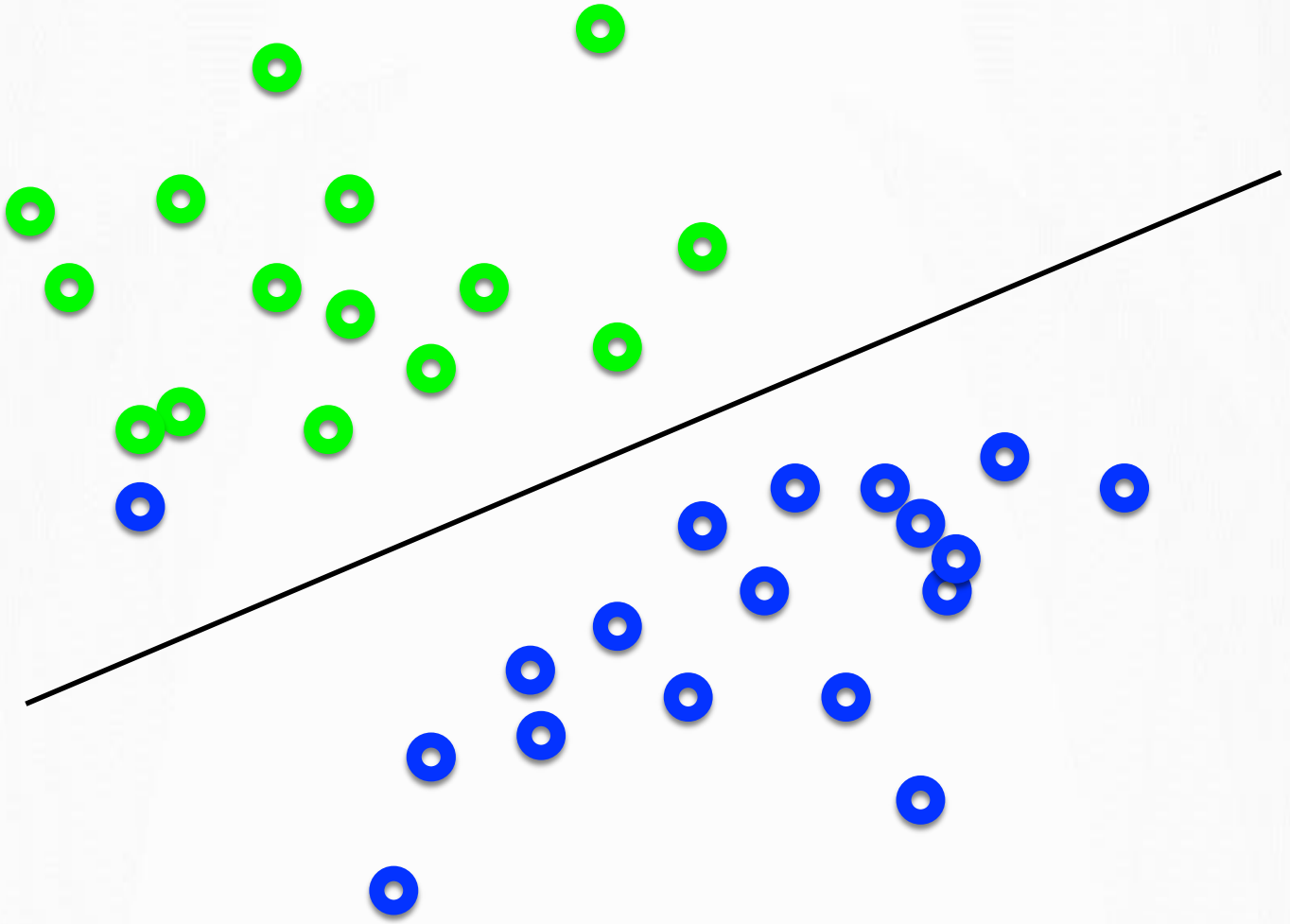
Slide credit: Noah Snavely

What's the best **w**?



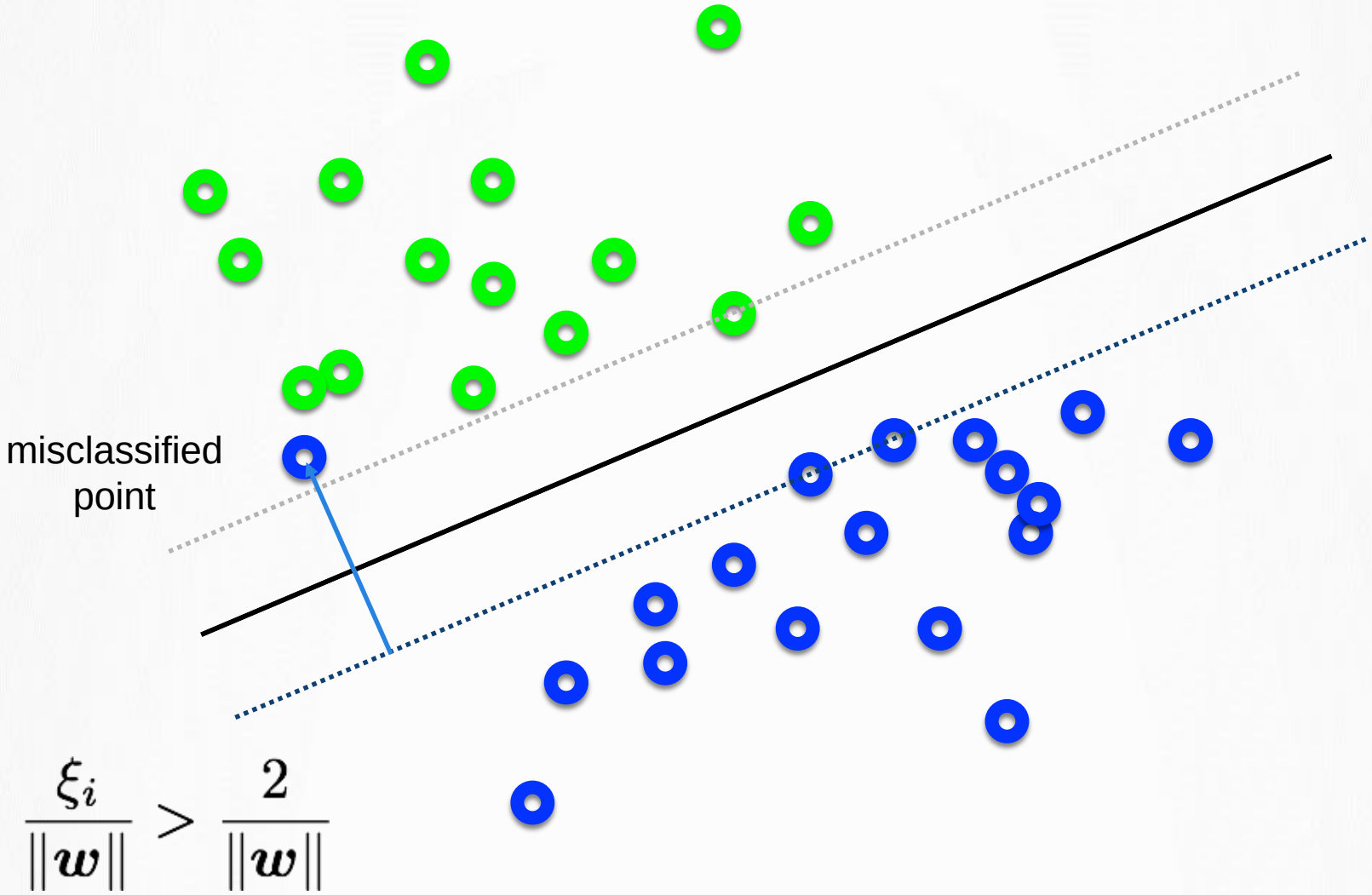
Intuitively, we should allow for some misclassification if we can get more robust classification

What's the best **w**?



Trade-off between the MARGIN and the MISTAKES
(might be a better solution)

Adding slack variables $\xi_i \geq 0$

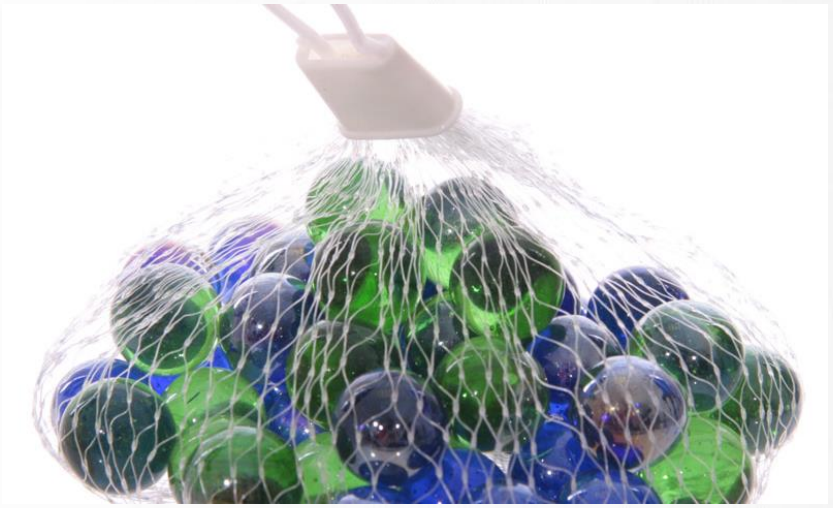


$$\frac{\xi_i}{\|\boldsymbol{w}\|} > \frac{2}{\|\boldsymbol{w}\|}$$

Slide credit: Noah Snavely

OBJECT DETECTION NOW

- Object detection is one of the most challenging computer vision tasks
- Aims at identifying the presence of various individual objects in an image
- Good results have been obtained when dealing with images with relatively simple image scenes and clear foreground objects
- The problem is not adequately addressed when dealing with the images and videos containing objects placed in arbitrary poses, with various shapes, and appearing in a cluttered and occluded environment



Han et al. (2018). Deep Learning for Visual Understanding: Part 2: Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection. IEEE Signal Processing Magazine.


60˚ 10 1.2 N, 24˚ 57 18 E