Supervised learning → Classification

Unsupervised learning → Understanding & Generation

Reinforcement learning → Behavior generation development

Discriminant models

KNN

k-nearest neighbors

# Classification

- Can a computer learn to recognize objects?
- Shown 10,000 flowers, can a computer "understand" flowers? Can it say if the new photograph shown is a flower?

Iris Setosa

Iris Versicolor

Iris Virginica

# Let's try our brain's algorithm!




Iris Versicolor

Iris Setosa

Iris Virginica

???

# What is Similarity?

The quality or state of being similar; likeness; resemblance; as, a similarity of features.

**Webster's Dictionary**

For example, for someone who is writing a software for healthcare industry, They may have to deal with the questions of "how similar are two patients."

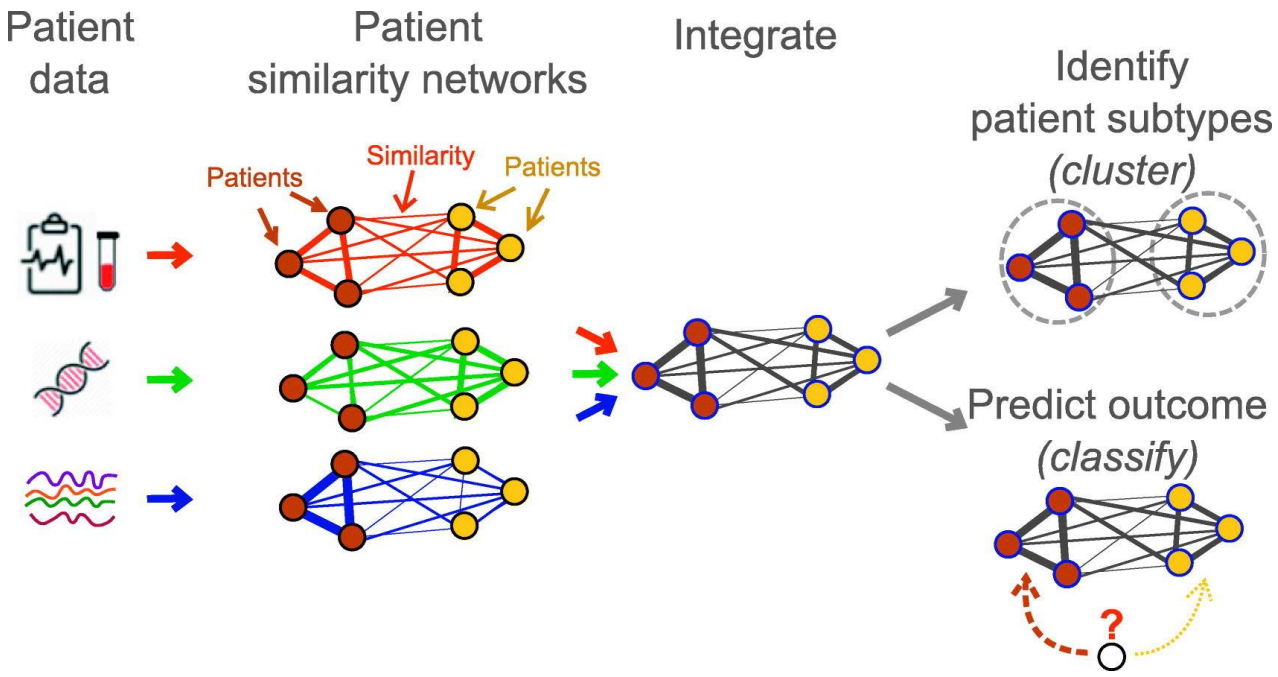
It depends on what you are comparing the two objects for.

Whole lot of research and Ph.D. thesis, just on the concept of similarity.

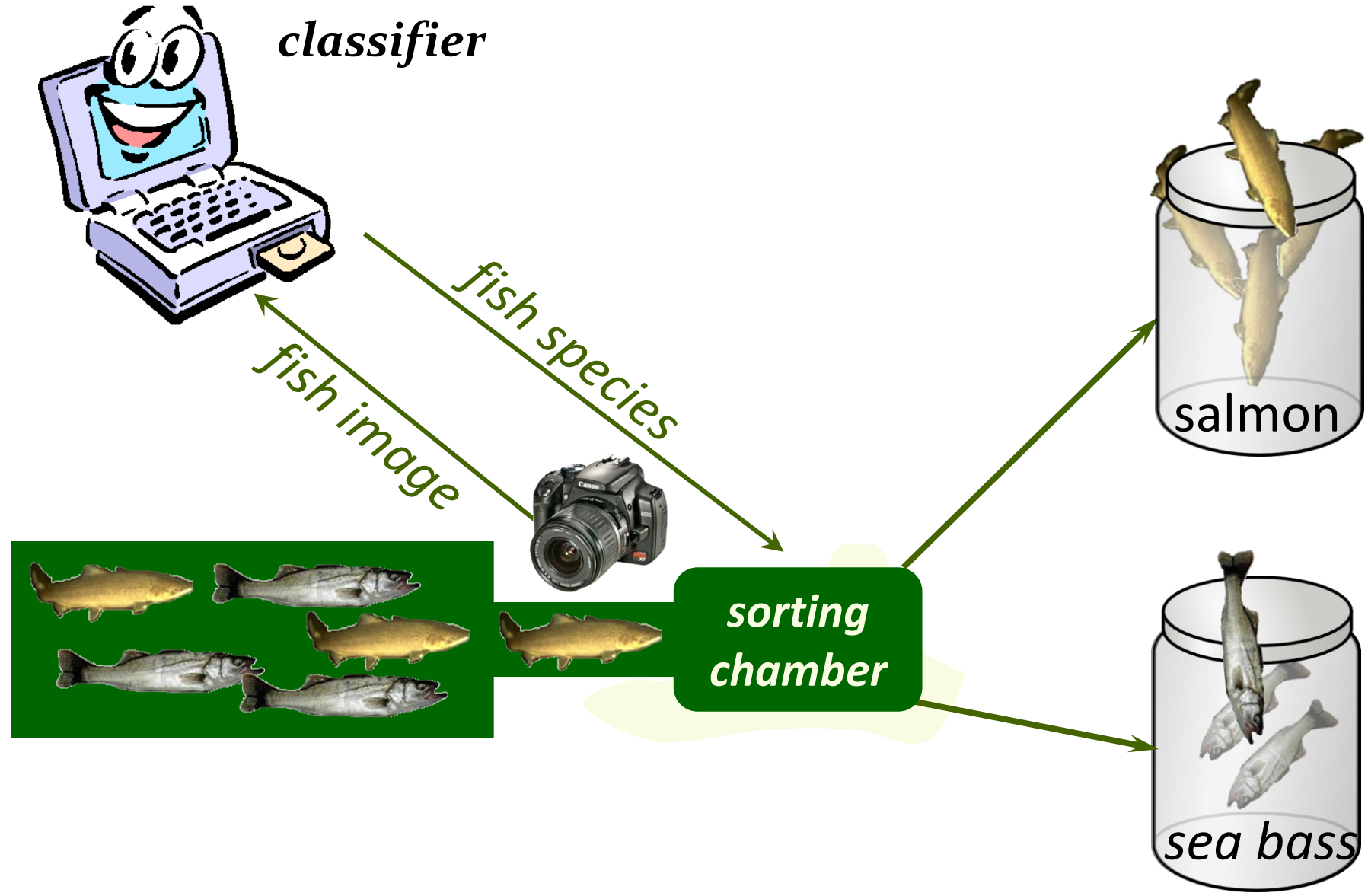
1. Patient Similarity Networks for Precision Medicine

2. Patient Similarity: Emerging Concepts in Systems and Precision Medicine

3. Machine learning of patient similarity: A case study on predicting survival in cancer patient after locoregional chemotherapy

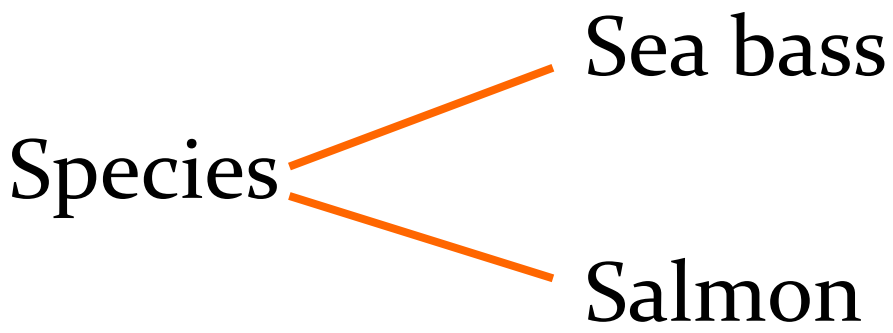


# Fish Sorting: For Packaging



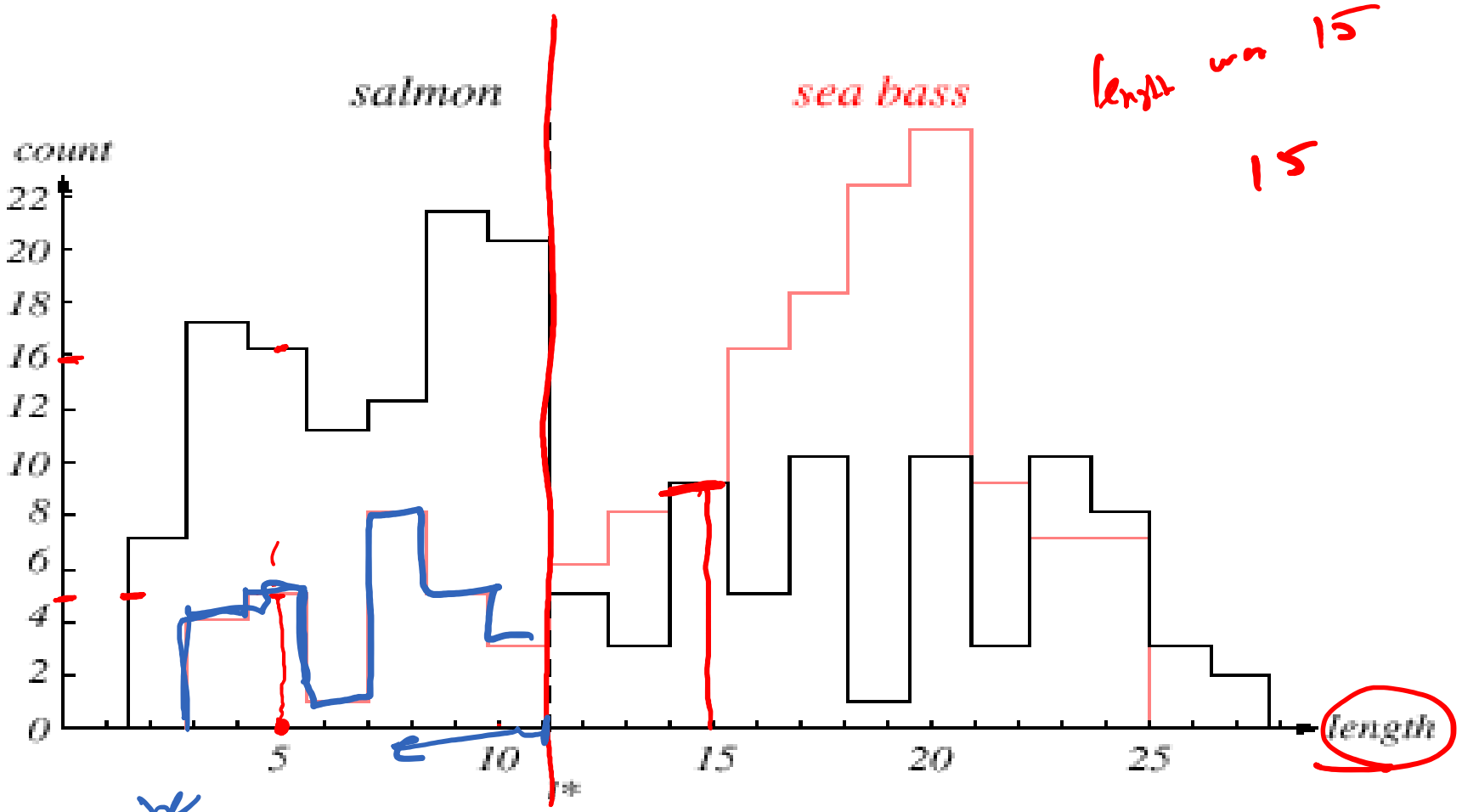
# An Example

- "Sorting incoming Fish on a conveyor according to species using optical sensing"



- Problem Analysis
  - Set up a camera and take some sample images to extract features
    - Length
    - Lightness
    - Width
    - Number and shape of fins
    - Position of the mouth, etc…
    - This is the set of all suggested features to explore for use in our classifier!

- Classification
  - Select the length of the fish as a possible feature for discrimination




length was 15

15

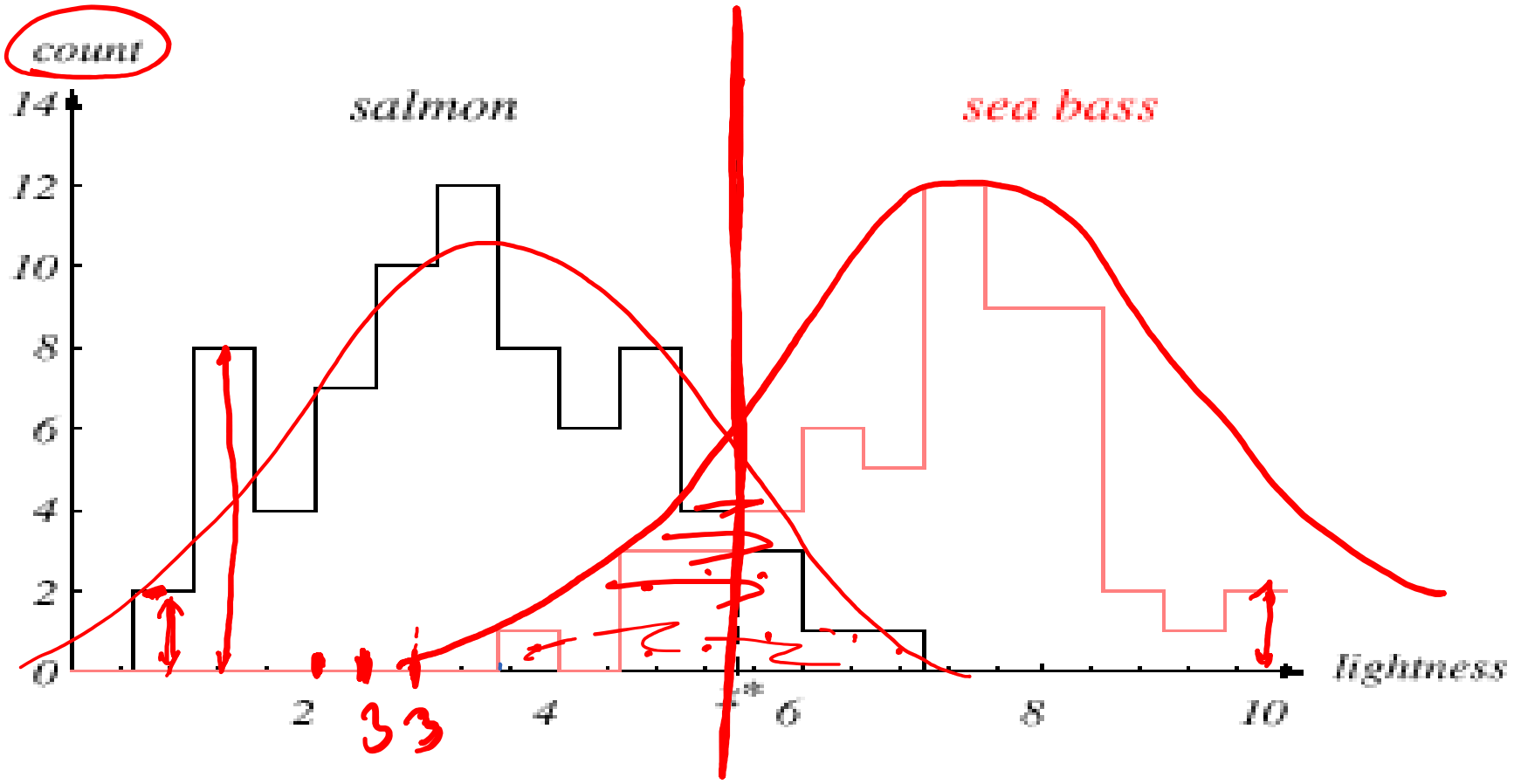
Maybe

5 sea bass were of length 5.

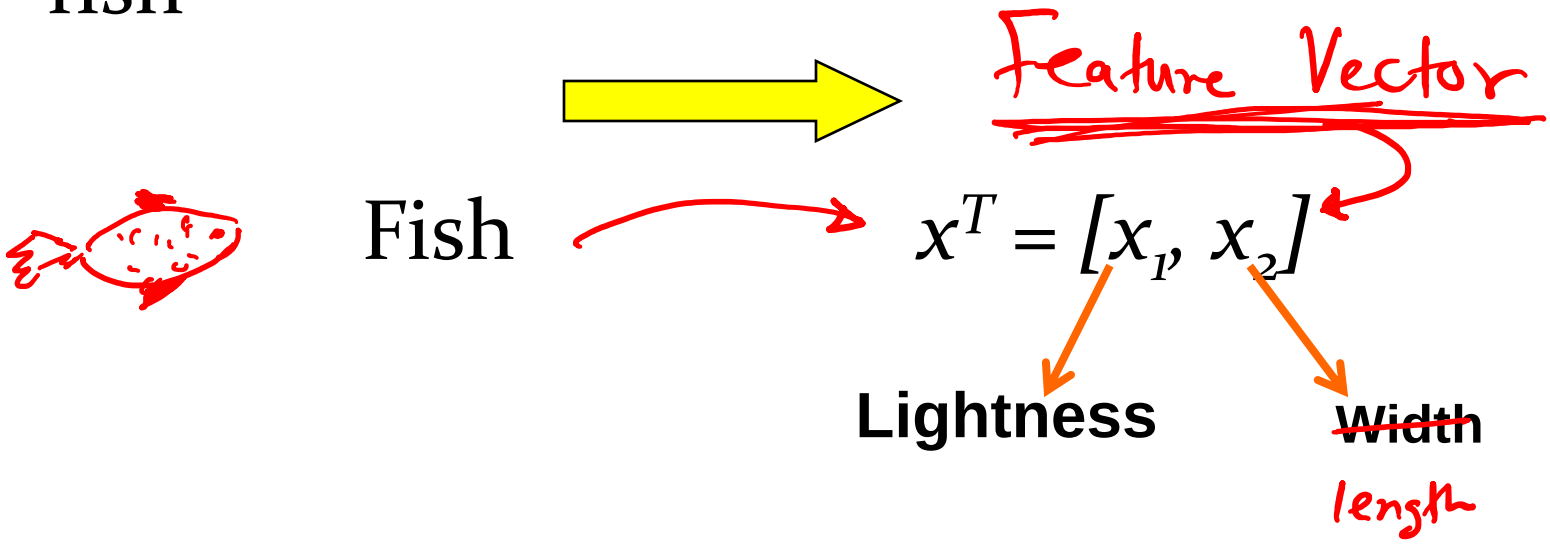
16 Salmon were of length 5

The length is a poor feature alone!

Select the lightness as a possible feature.

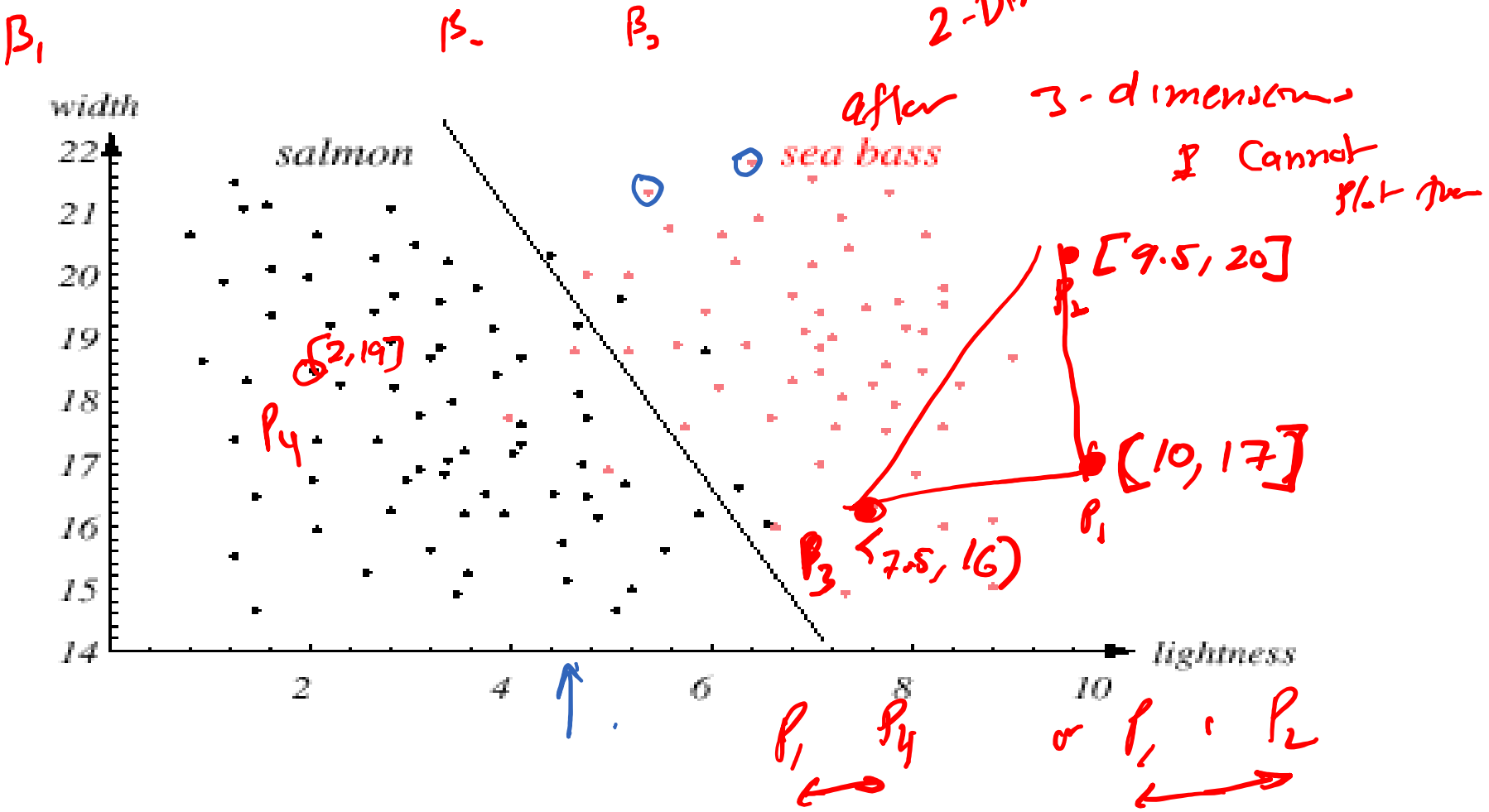

count
salmon
sea bass
14
12
10
8
6
4
2
0
2
4
6
8
10
lightness

- Adopt the lightness and add the width of the fish



2-dimensional Vector in 2-d space

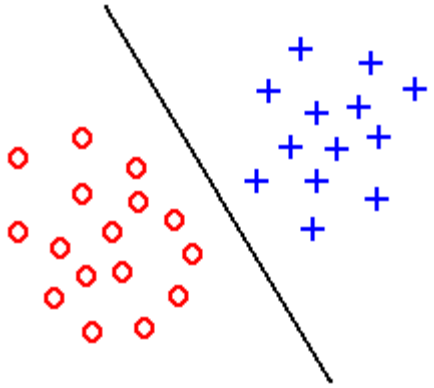
- Plotting Salmon and Seabass based on two-dimensional feature vector.



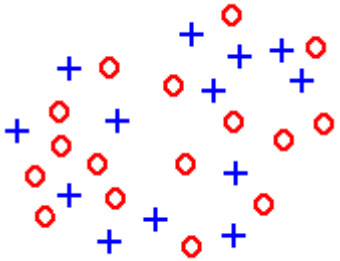
# Feature extraction

Task: to extract features which are good for classification.

Good features:
- Objects from the same class have similar feature values.
- Objects from different classes have different values.

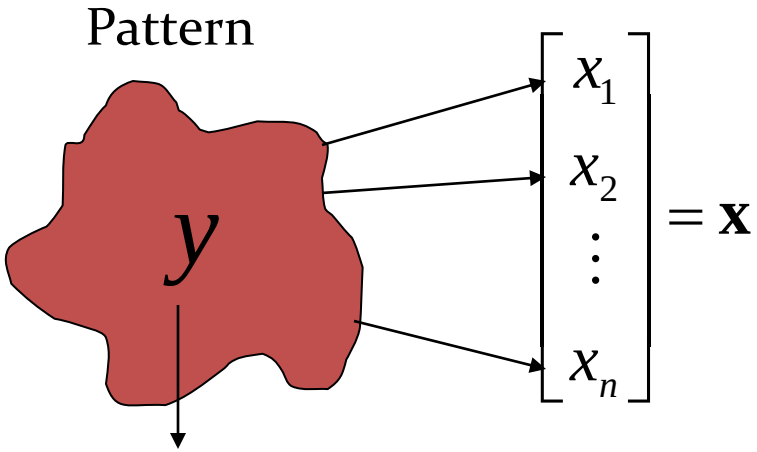


"Good" features



"Bad" features

# Basic concepts




Feature vector $\mathbf{x} \in X$

- A vector of observations (measurements).
- $\mathbf{x}$ is a point in feature space $X$.

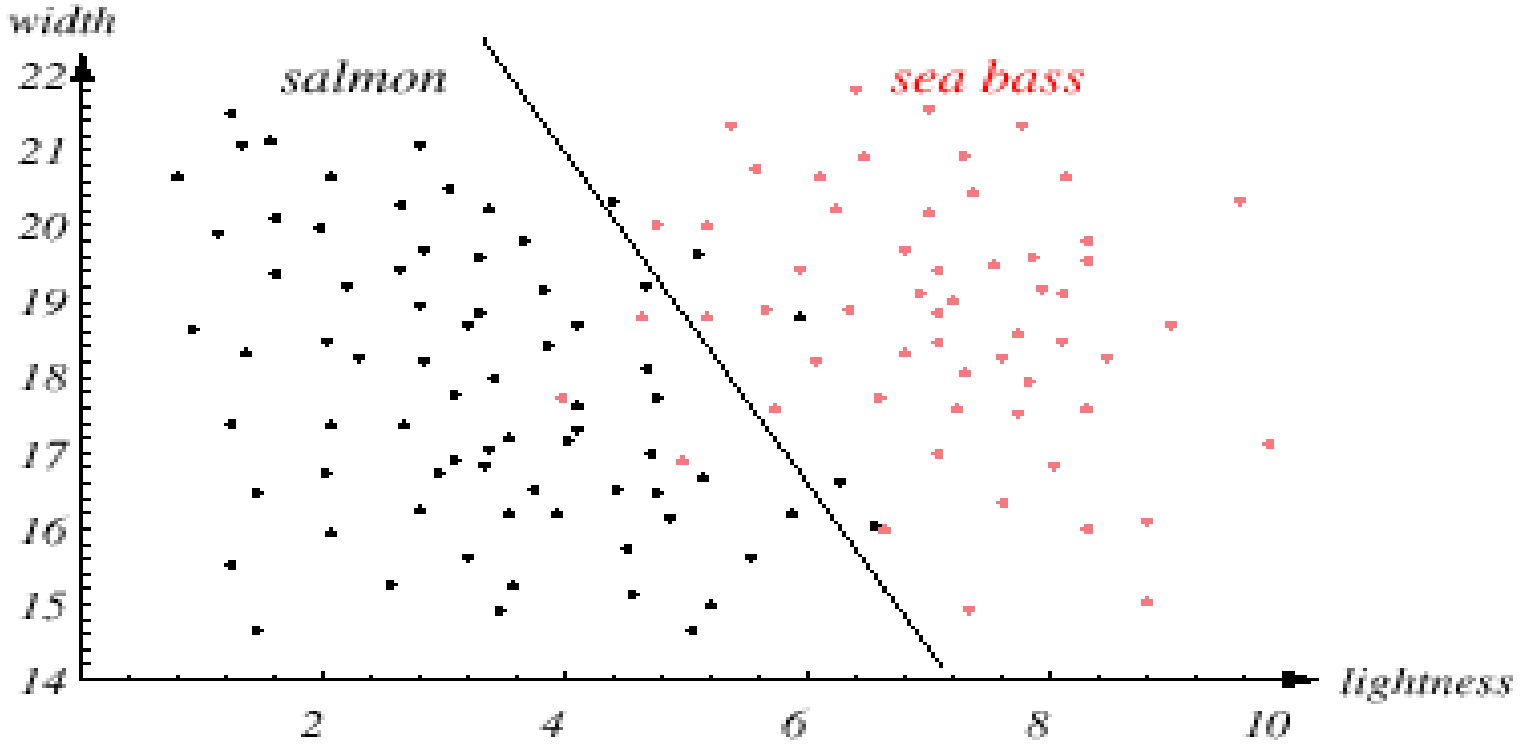
Hidden state $y \in Y$

- Cannot be directly measured.
- Patterns with equal hidden state belong to the same class.

Task

- To design a classifer (decision rule) $q : X \to Y$ which decides about a hidden state based on an onbservation.

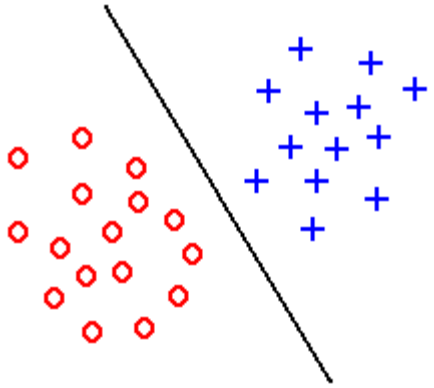
- Plotting Salmon and Seabass based on two-dimensional feature vector.



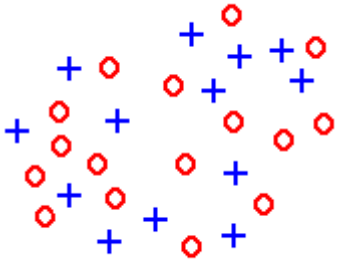
# Feature extraction

Task: to extract features which are good for classification.

Good features:
- Objects from the same class have similar feature values.
- Objects from different classes have different values.

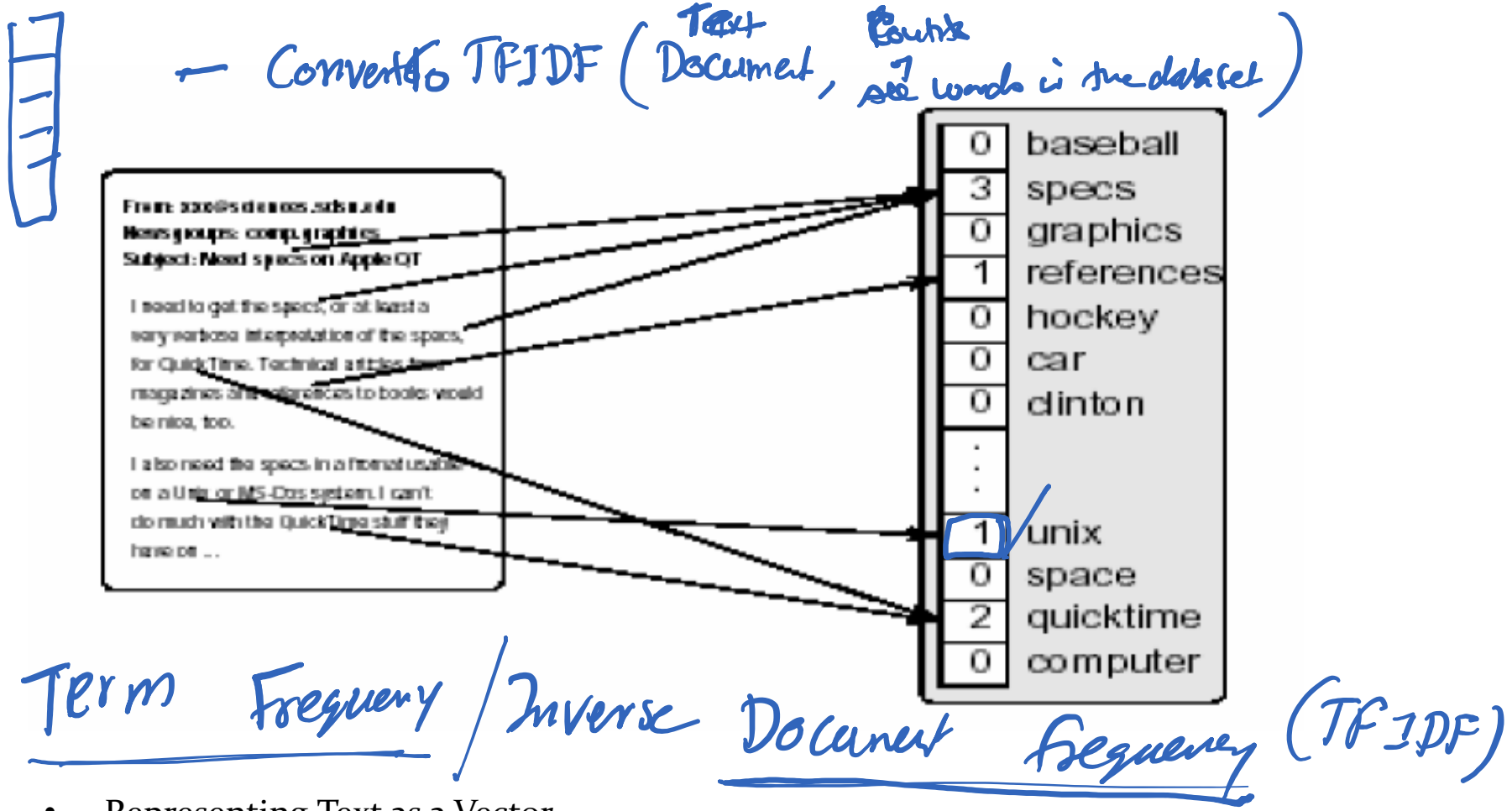


"Good" features



"Bad" features

# Text Classification




- Representing Text as a Vector.
- Stem words used, such that "computer, computes .." all get noted under "compute."
- The number in the vector is actually divided by the number of documents that number appears in. "Inverse Document Frequency"

Divide by # of document

Remove "stop words" e.g. is, the, a, an

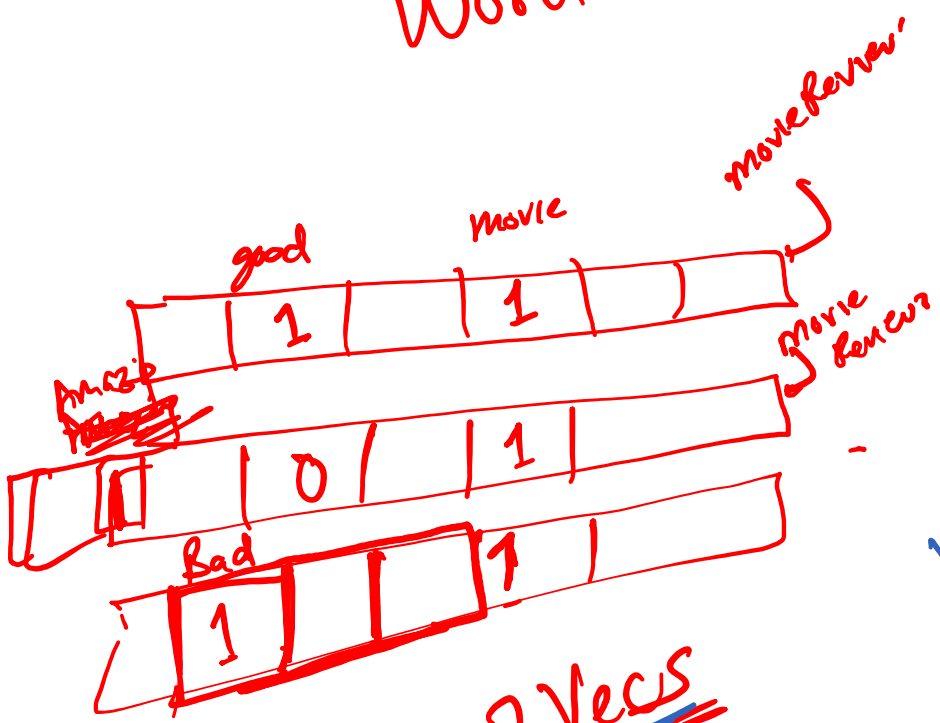
'Stemming' ← keep only the stem words

Compute, Computer, Computing.

!, ?

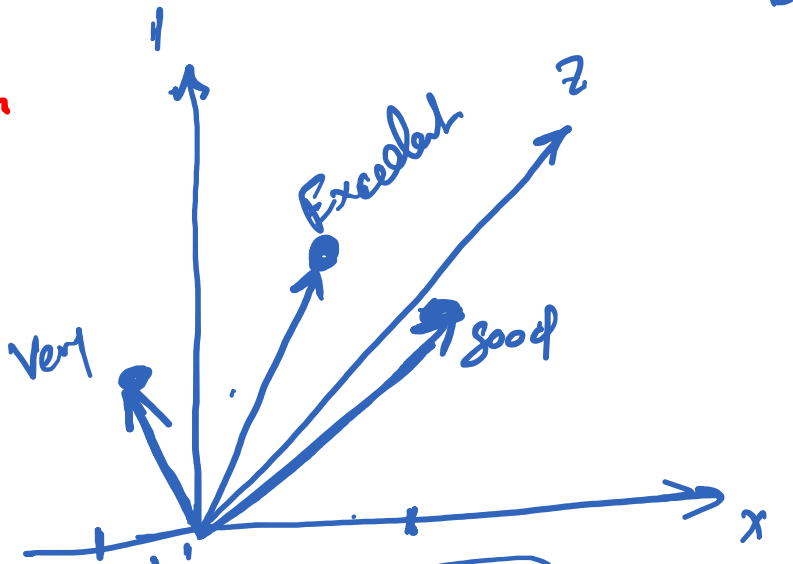
Sentiment detection for movie reviews

# Word 2 Vecs

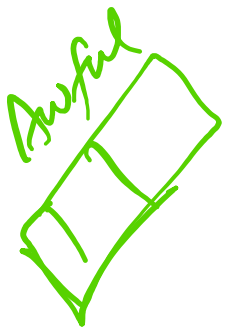


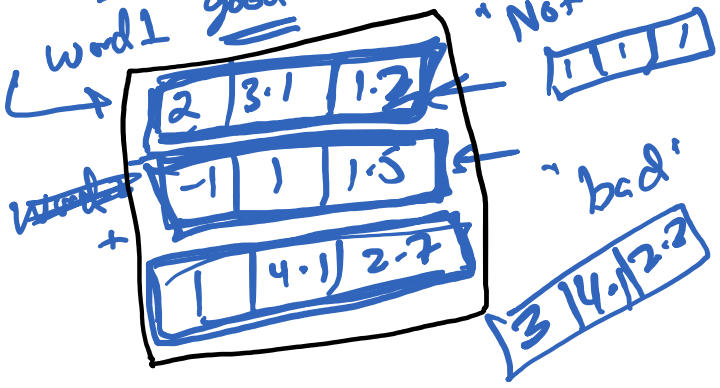

One document in TFIDf is one feature vector

What should be the vector for the word 'bad'

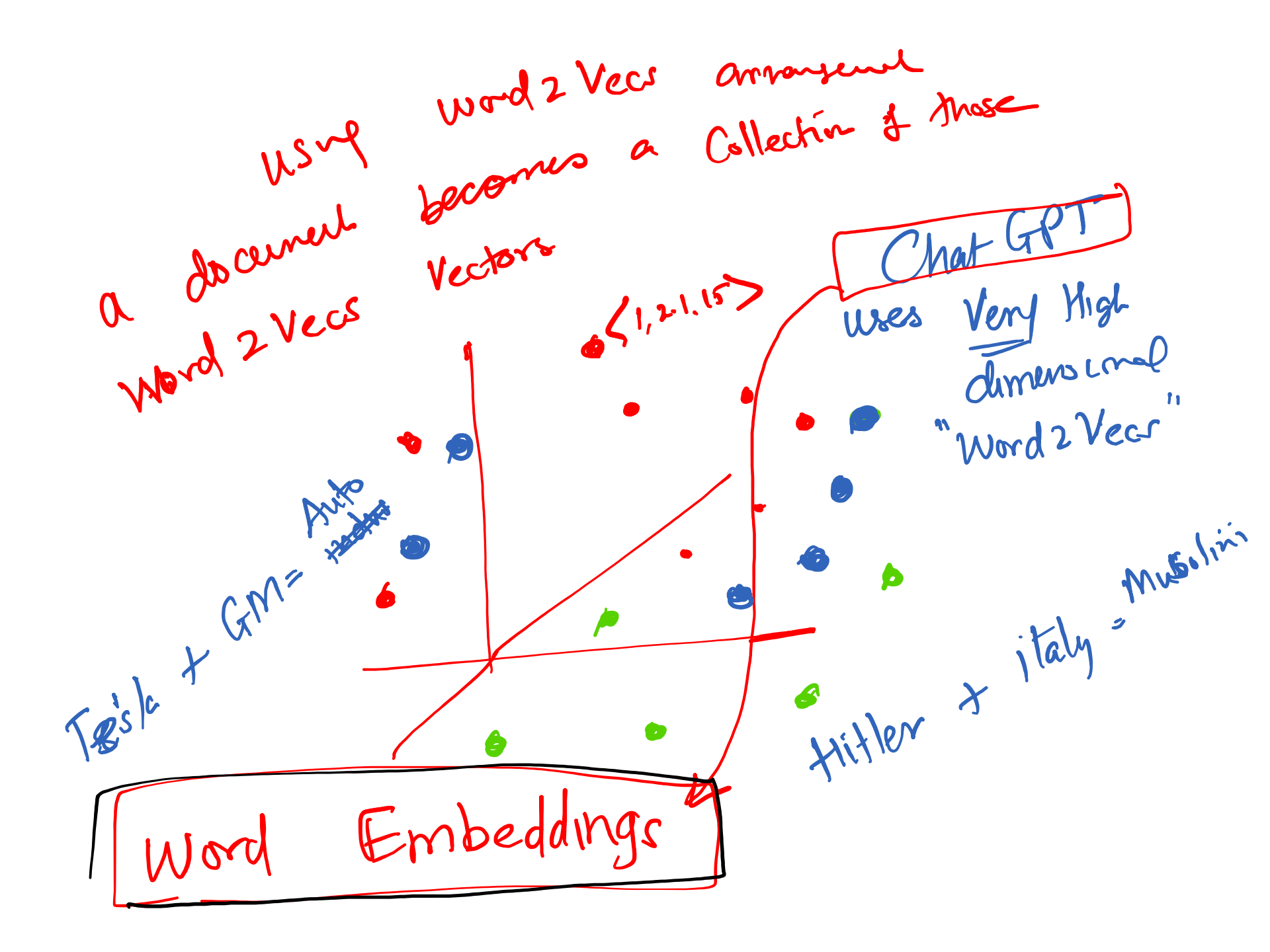


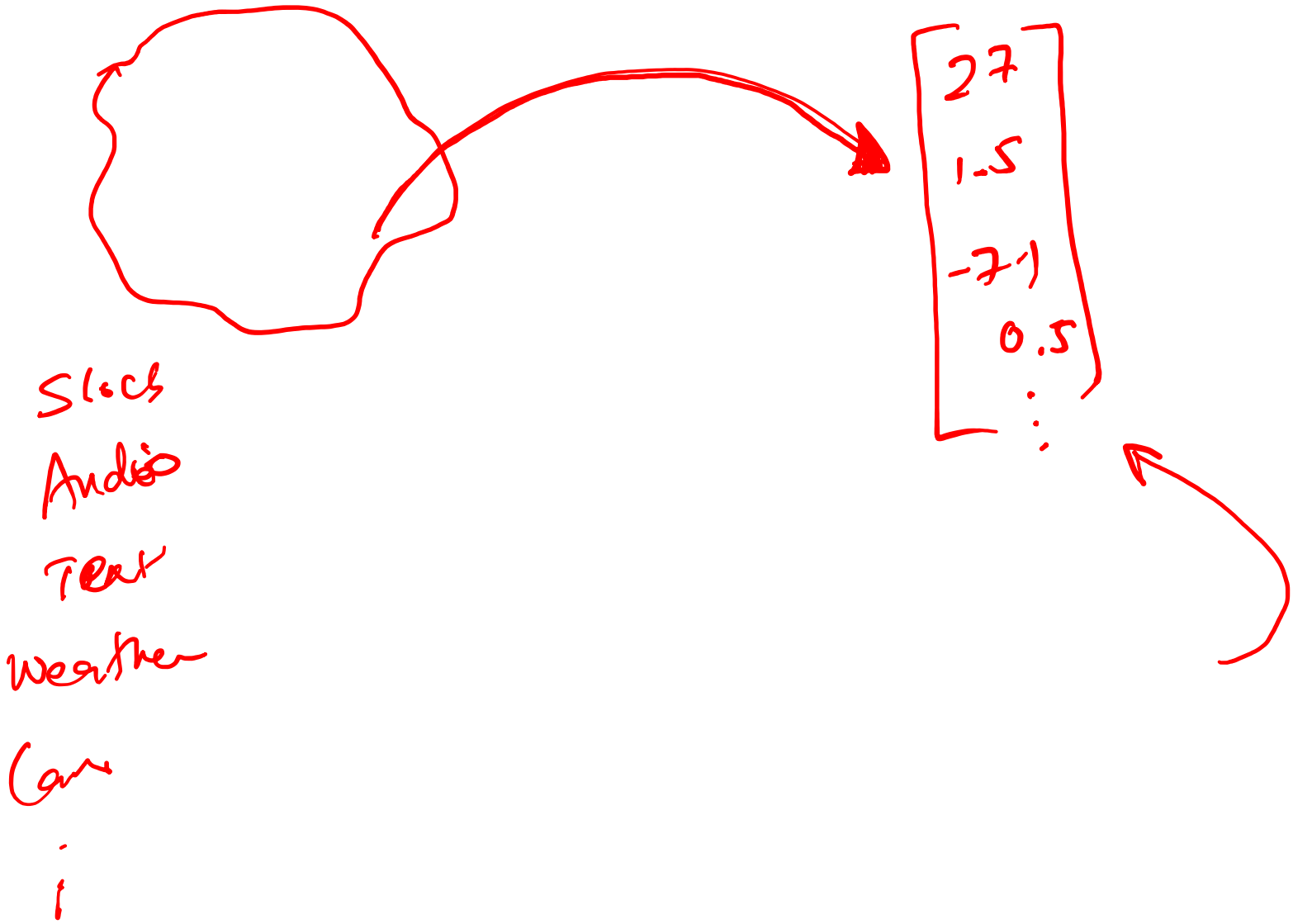

## Word 2 Vecs



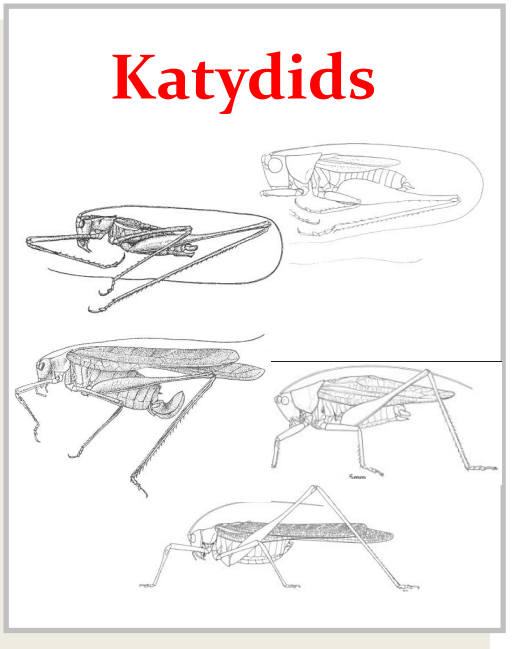

document will be a collection of all word Vec

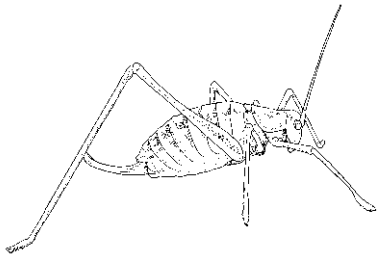

Using Word2Vecs arrangement
a document becomes a Collection of those
Word2Vecs Vectors
<1,2,1.15>
Chat GPT
uses Very High dimensional "Word2Vecs"
Tesla + GM = Auto
Hitler + italy = Mussolini
Word Embeddings


27
1.5
-7.1
0.5
Stock
Audio
Text
Weather
Car

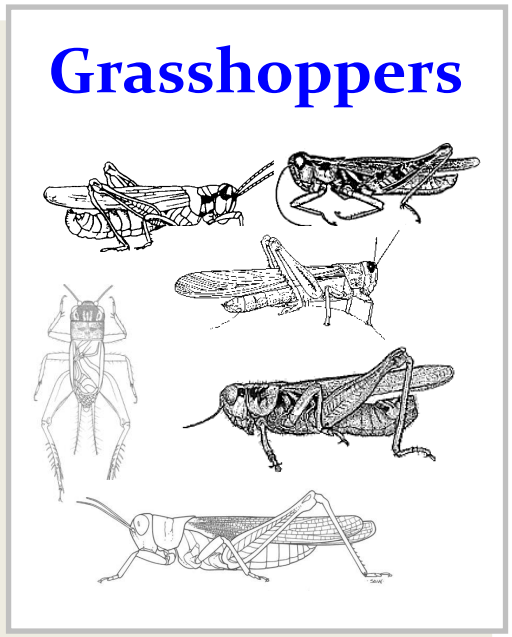
# Let's go back to agriculture!

Given a collection of annotated data. In this case 5 instances **Katydids** of and five of **Grasshoppers**, decide what type of insect the unlabeled example is.



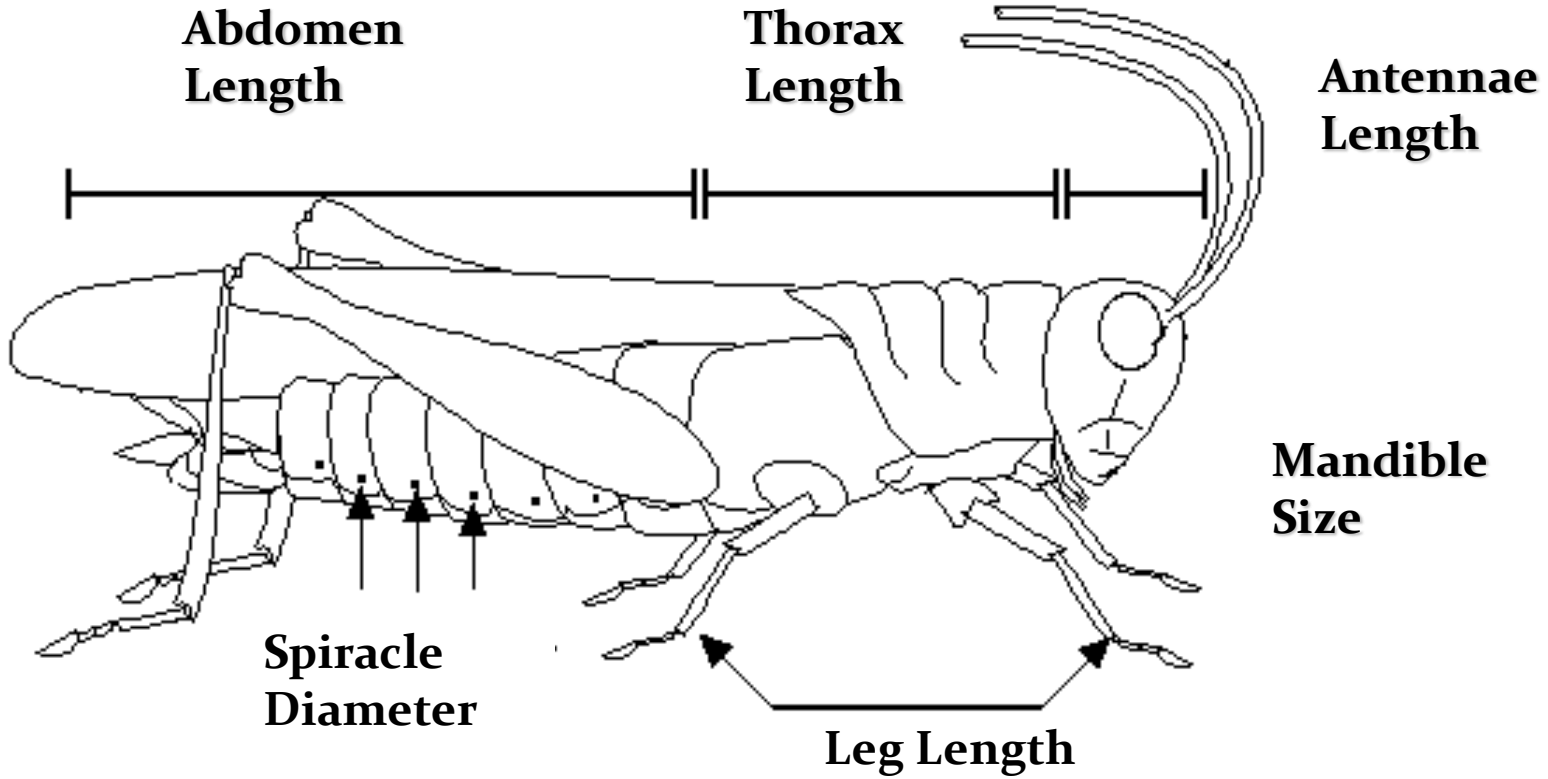
**Katydid** or **Grasshopper**?



# For any domain of interest, we can measure *features*

**Color {Green, Brown, Gray, Other}**

**Has Wings?**



We can store features in a database.

The classification problem can now be expressed as:

- Given a training database (My_Collection), predict the class label of a previously unseen instance

My_Collection

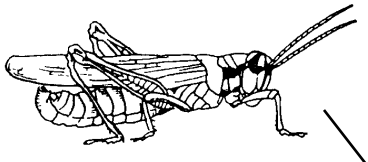
| Insect ID | Abdomen Length | Antennae Length | Insect Class |
|---|---|---|---|
| *1* | 2.7 | 5.5 | Grasshopper |
| *2* | 8.0 | 9.1 | Katydid |
| *3* | 0.9 | 4.7 | Grasshopper |
| *4* | 1.1 | 3.1 | Grasshopper |
| *5* | 5.4 | 8.5 | Katydid |
| *6* | 2.9 | 1.9 | Grasshopper |
| *7* | 6.1 | 6.6 | Katydid |
| *8* | 0.5 | 1.0 | Grasshopper |
| *9* | 8.3 | 6.6 | Katydid |
| *10* | 8.1 | 4.7 | Katydids |

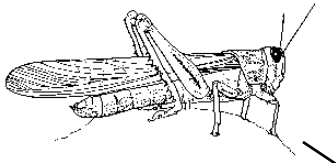
previously unseen instance =

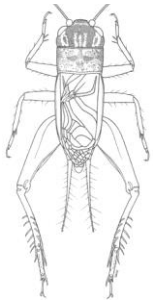
| | | | |
|---|---|---|---|
| *11* | 5.1 | 7.0 | ??????? |

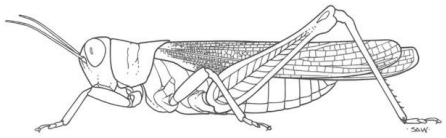
# Grasshoppers

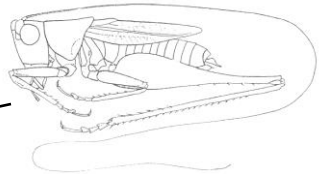
# Katydids

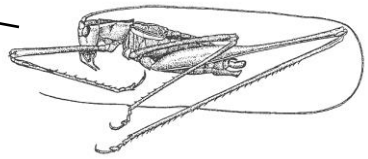


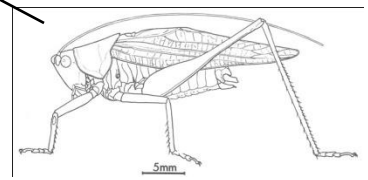


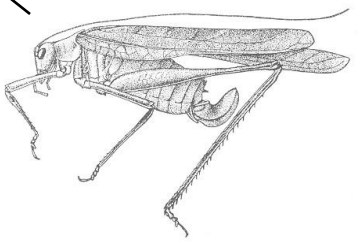


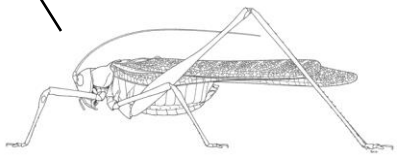


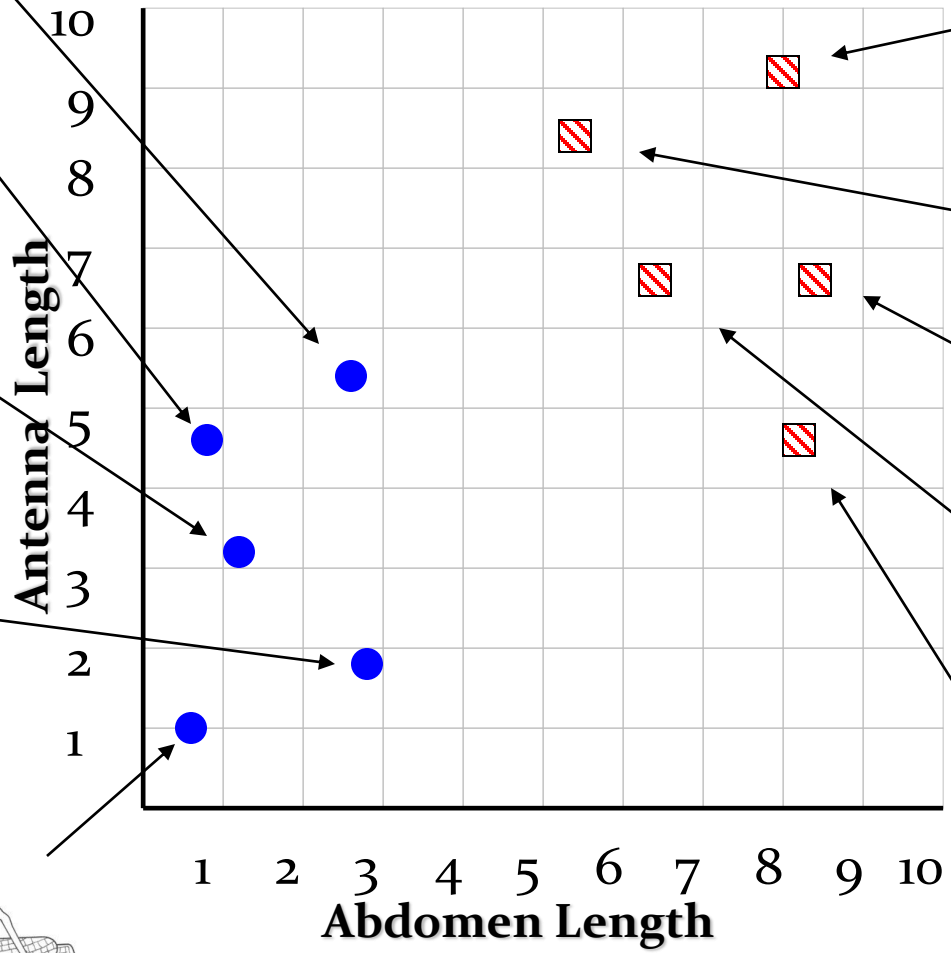


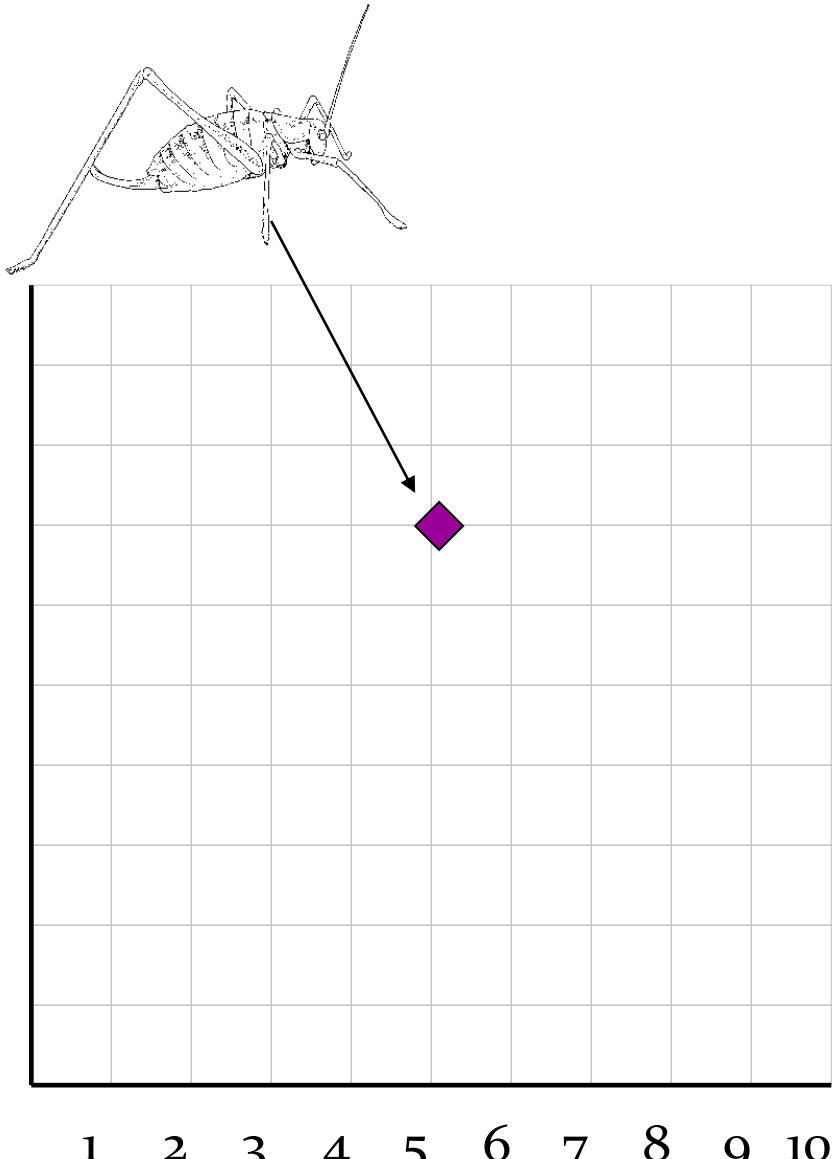


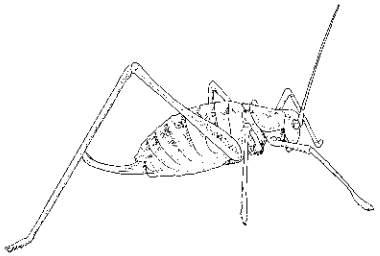




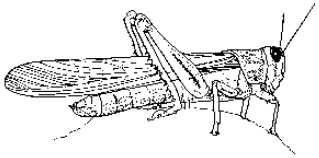




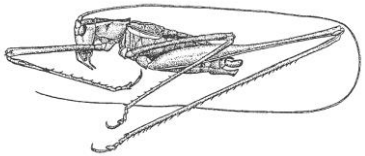
**Katydid** or **Grasshopper**?

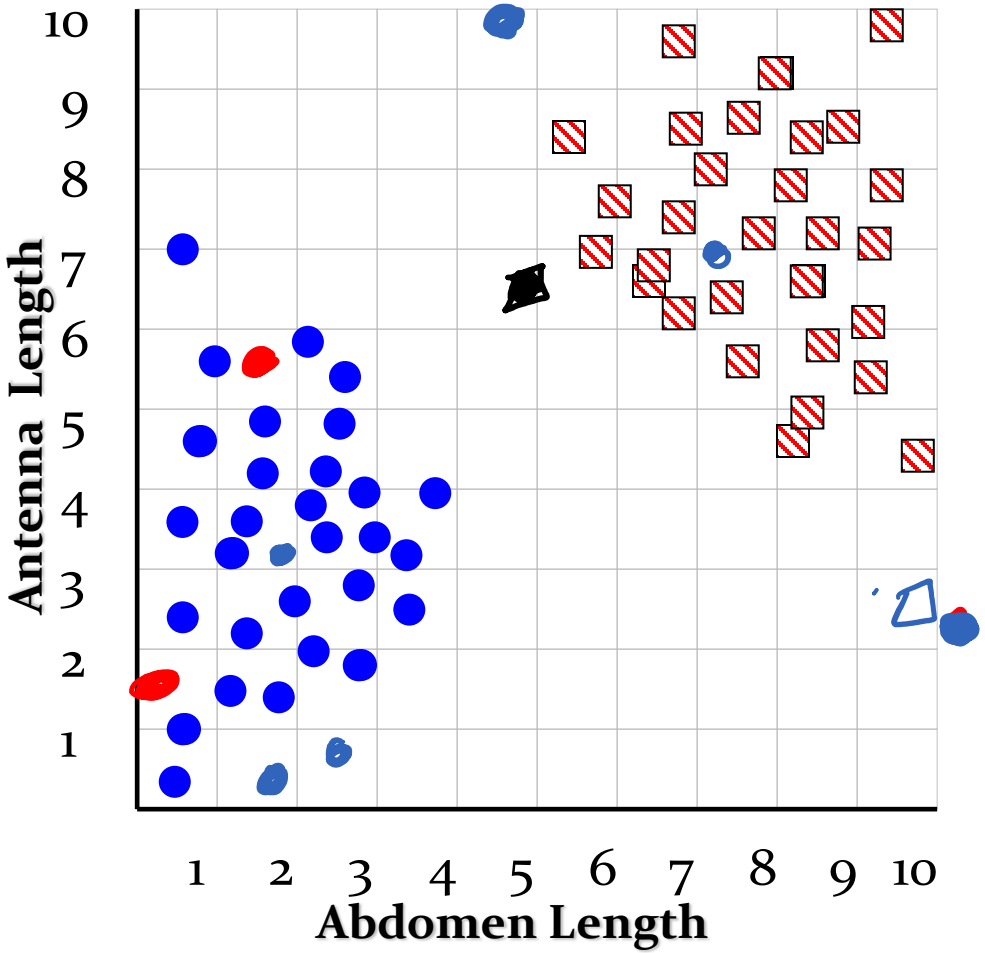
# Grasshoppers



We will also use this lager dataset as a motivating example…

# Katydids





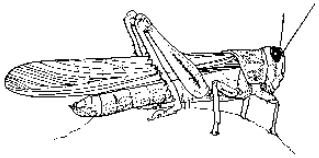

Each of these data objects are called…

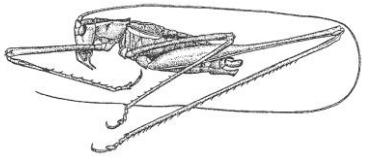
- exemplars
- (training) examples
- instances
- tuples

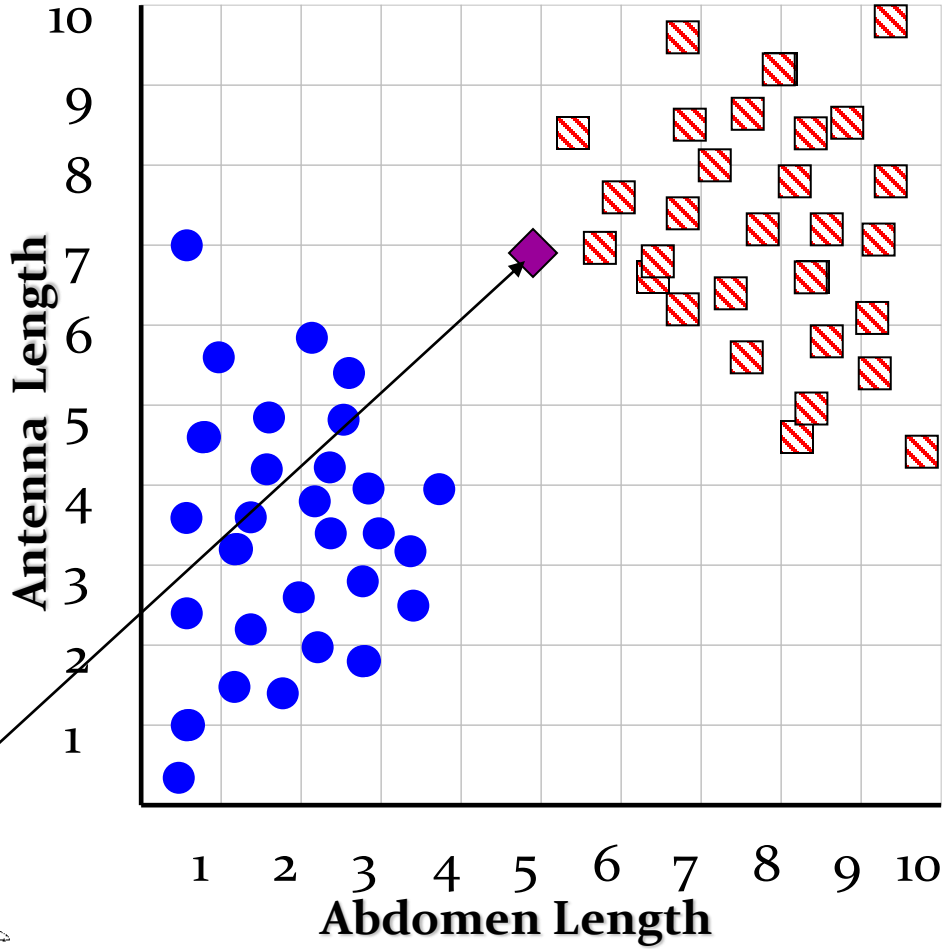
# Grasshoppers

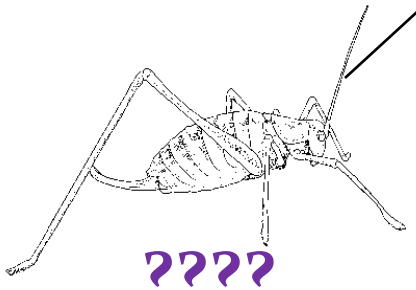
# Katydids

We will also use this lager dataset as a motivating example…







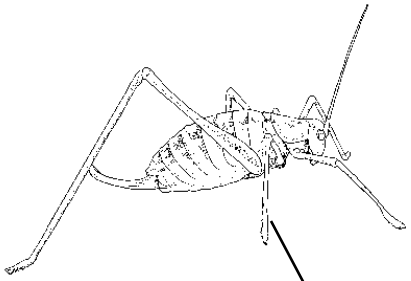



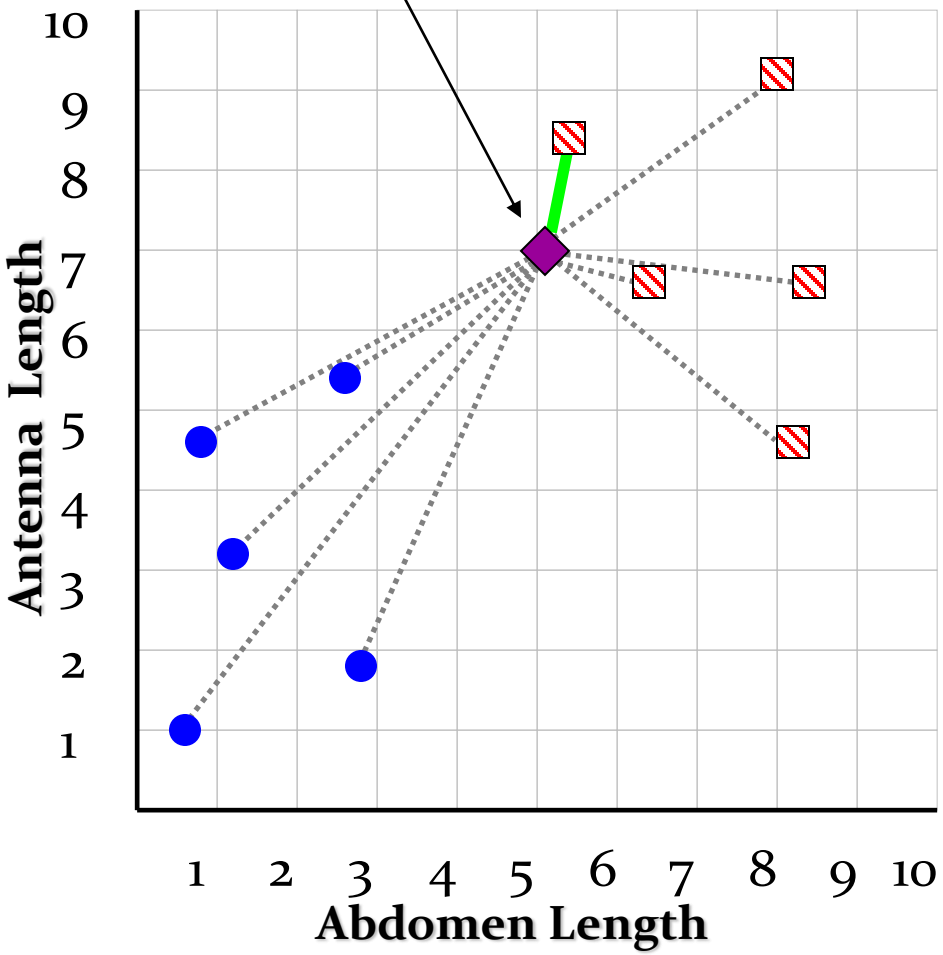
????

Each of these data objects are called…

- exemplars
- (training) examples
- instances
- tuples

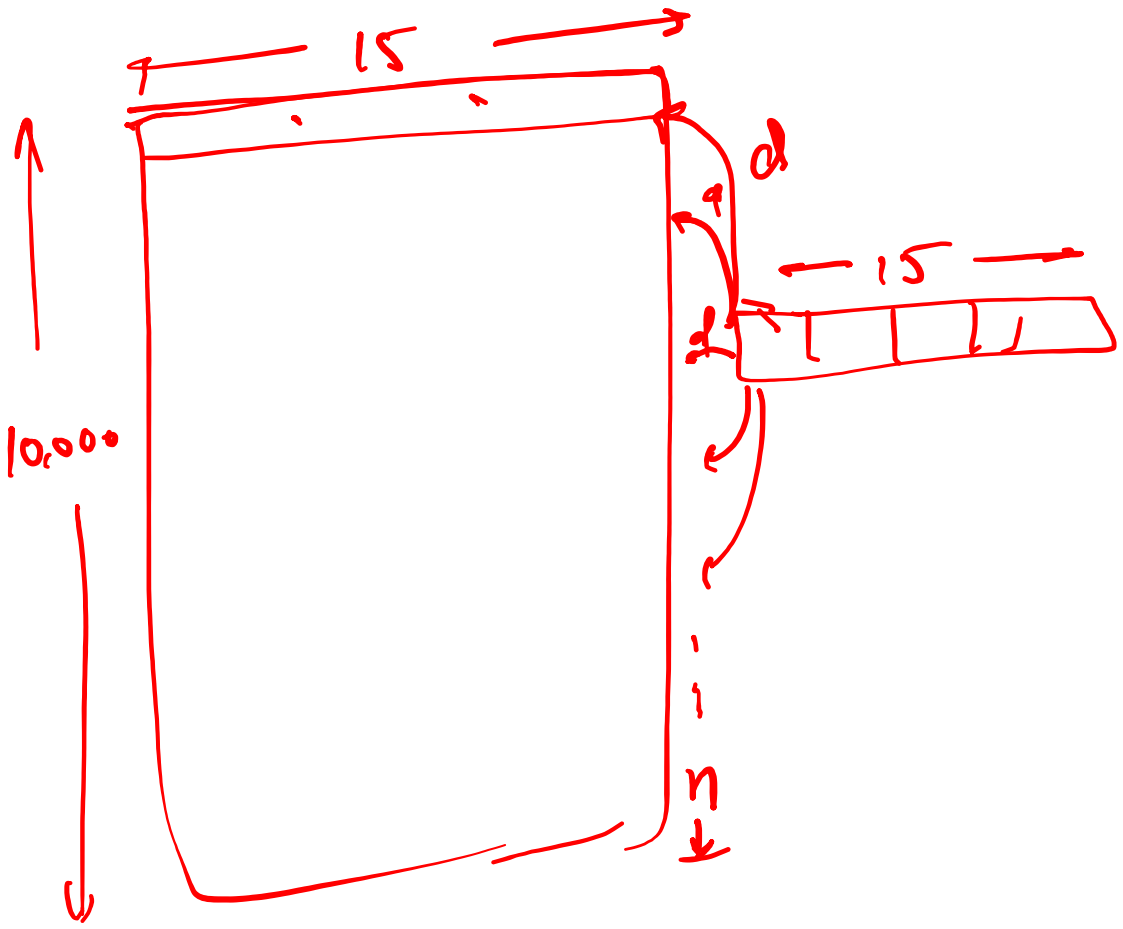
# Nearest Neighbor Classifier






**If** the **nearest** instance to the previously unseen instance **is a Katydid**
    class is **Katydid**
**else**
    class is **Grasshopper**

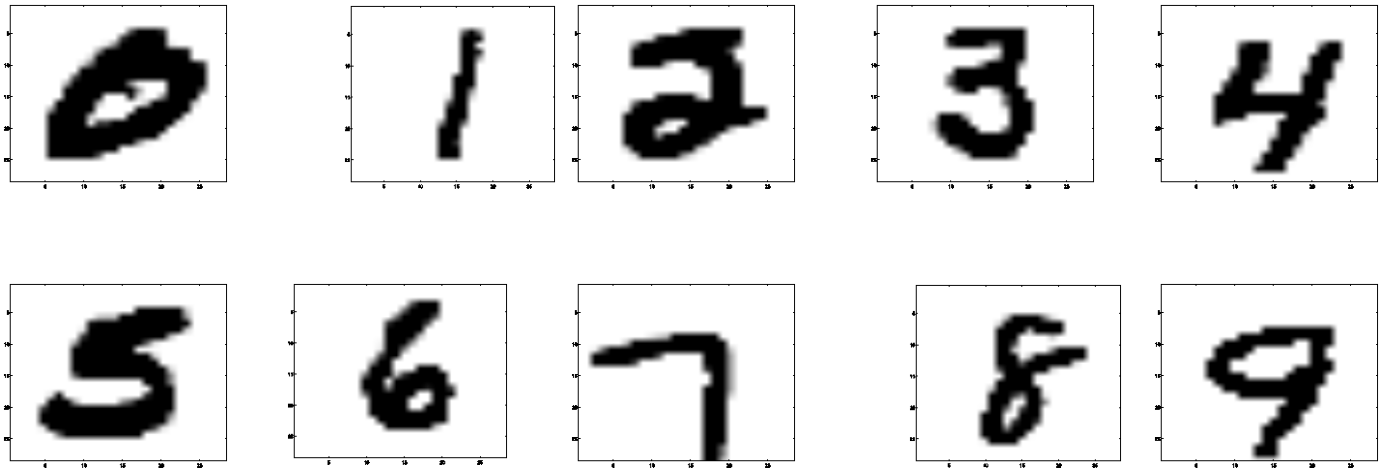
**Katydids**
**Grasshoppers**




Model optimization f.

# Hand Written Digits example

Database of 20,000 images of handwritten digits, each labeled by a human (Supervised Learning)



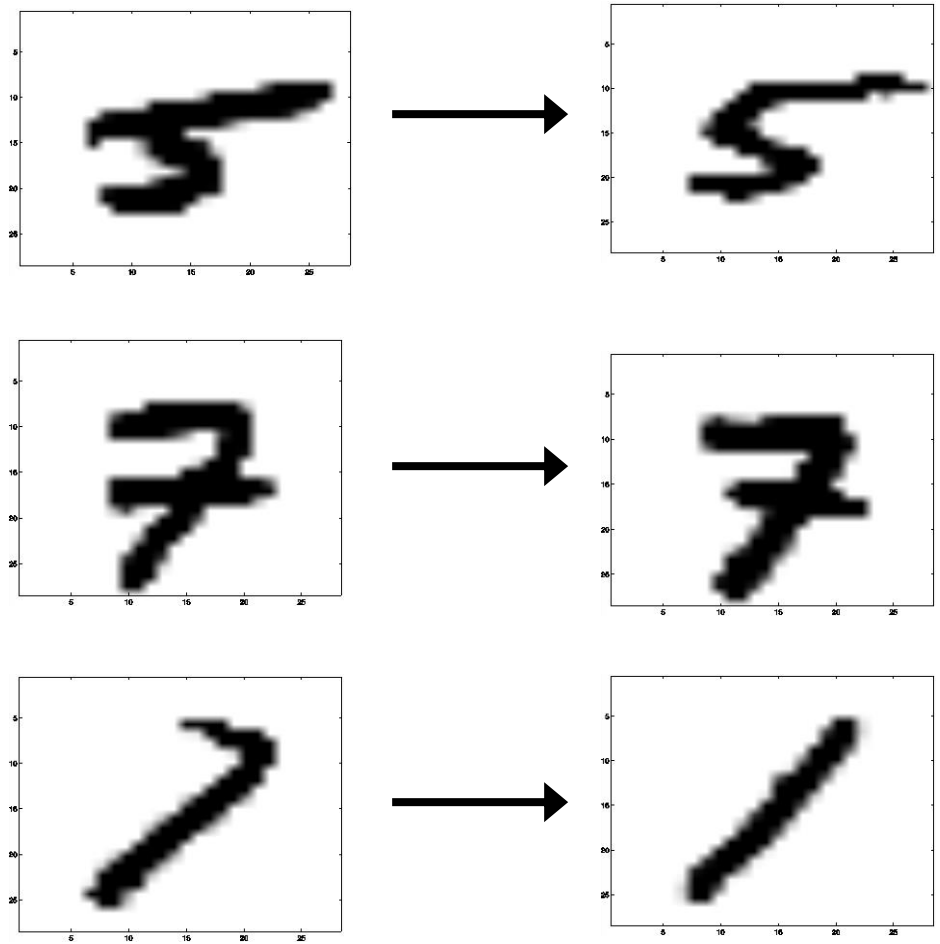
**[28 x 28 greyscale; pixel values 0-255; labels 0-9]**

Use these to learn a classifier which will label digit-images automatically…

# Nearest neighbor

Image to label

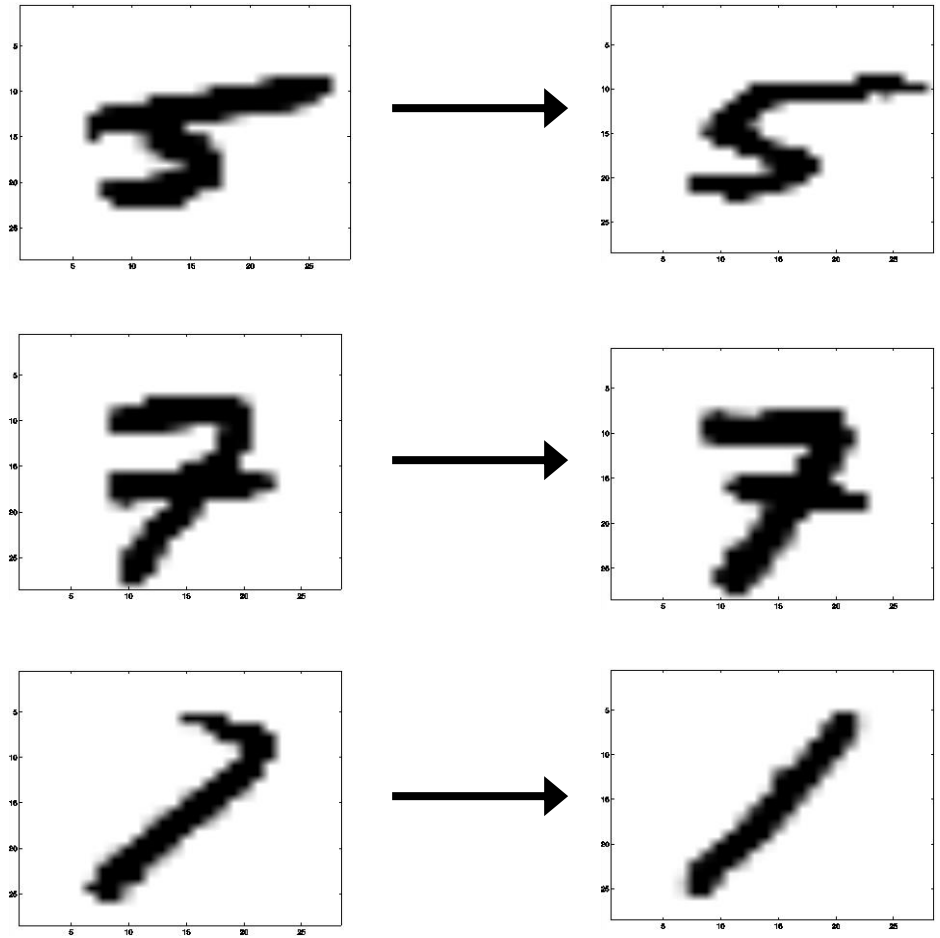
Nearest neighbor



# Nearest neighbor

Image to label

Nearest neighbor



Overall:

error rate = 6%

(on test set)