Features
K-Nearest Neighbors

Research Skills: Machine Learning

Grzegorz Chrupała
g.chrupala @ uvt.nl

What is a learning example?

- We saw some types of example targets/outputs:
  - (Real) number
  - Class label
  - Sequence of labels

How about inputs?

- Textual
  - emails
  - tweets
- Visual
  - photos
  - video
  - scanned handwriting
- Audio
  - voice recordings
  - music tracks
- Physical
  - people / animals
  - plants
  - other objects

Generic algorithms require common representations

Decompose objects into FEATURES

Features

- We decompose inputs into features
- A feature is a measurable aspect of an object

- Features are often extracted before learning
- Some learning algorithms can extract features from some types of input (e.g. images or text)

We want to distinguish between three species of the iris plant

Iris setosa

Iris versicolor

Iris virginica

How do we extract features?

- If inputs are physical samples of flowers
- If inputs are photographs of flowers

- If inputs are physical samples of flowers
  - Manual or automatic measurements
    - size of petals, leaves, color, weight, ...
- If inputs are photographs of flowers
  - Image processing: edges, color, gradients, ...
  - Automatic learning of features from pixels

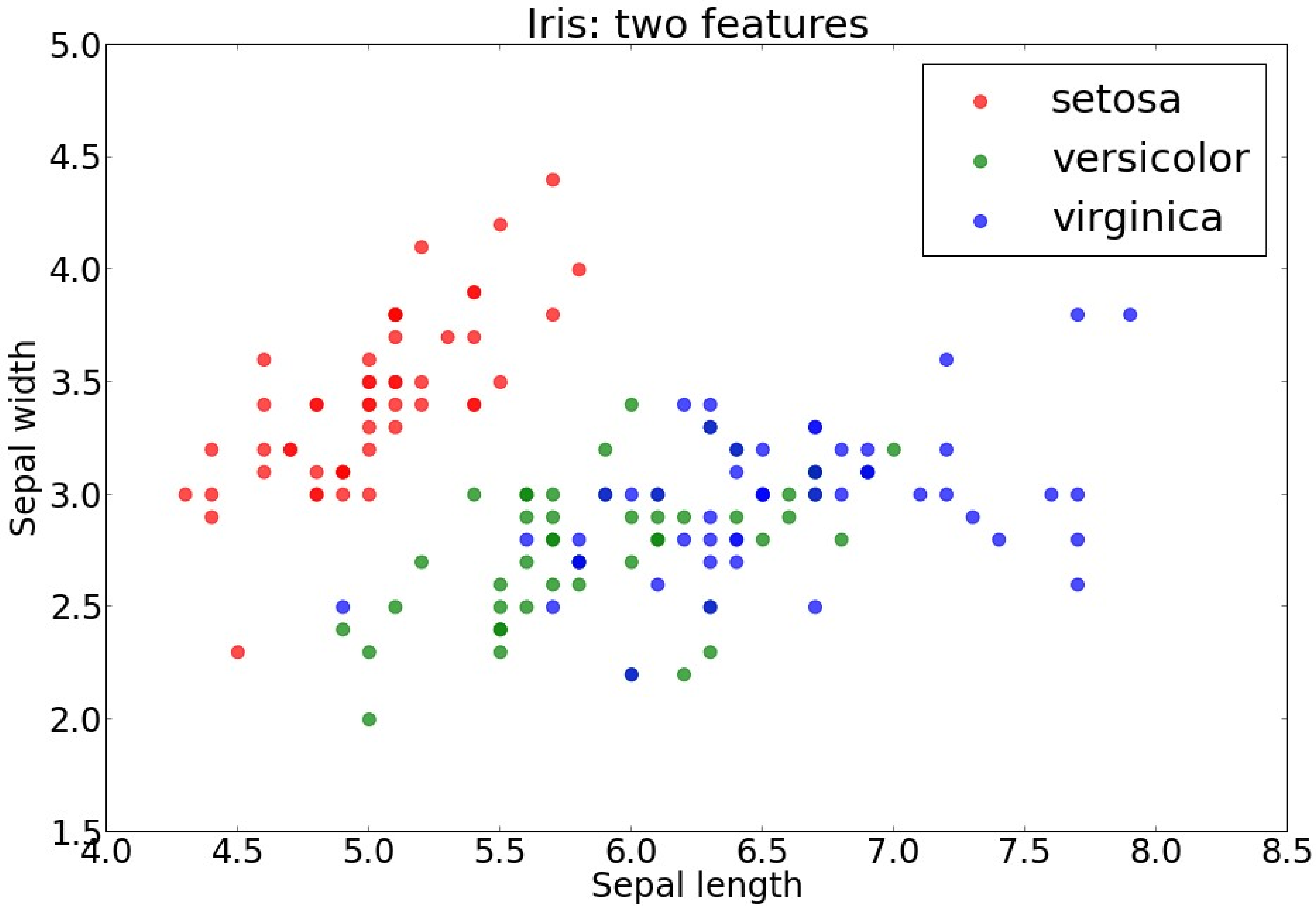
Later in the course

The iris dataset

```
INPUT               OUTPUT
6.9 3.2 5.7 2.3 virginica
5.4 3.4 1.5 0.4 setosa
7.2 3.0 5.8 1.6 virginica
6.3 3.3 4.7 1.6 versicolor
5.8 2.7 3.9 1.2 versicolor
7.2 3.6 6.1 2.5 virginica
5.4 3.9 1.7 0.4 setosa
```

Features:

Sepal_Length, Sepal_Width, Petal_Length, Petal_Width



Iris: two features
setosa
versicolor
virginica
Sepal width
Sepal length
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5

Census income

INPUT					**TARGET**
age	**edu**	**occupation**	**race**	**sex**	**income**
39	13	Adm-clerical	White	Male	<=50K
50	13	Exec-managerial	White	Male	<=50K
38	9	Handlers-cleaners	White	Male	<=50K
53	7	Handlers-cleaners	Black	Male	<=50K
28	13	Prof-specialty	Black	Female	<=50K
37	14	Exec-managerial	White	Female	<=50K
49	5	Other-service	Black	Female	<=50K
52	9	Exec-managerial	White	Male	>50K
31	14	Prof-specialty	White	Female	>50K
42	13	Exec-managerial	White	Male	>50K
37	10	Exec-managerial	Black	Male	>50K
30	13	Prof-specialty	Asian	Male	>50K
23	13	Adm-clerical	White	Female	<=50K
32	12	Sales	Black	Male	<=50K

Categorical features

- Some algorithms can easily use categorical features such as `occupation` or `race` or `sex`
- In many cases we'll convert them to numerical features

Categorical → Numerical

race	sex
White	Male
Black	Male
Black	Female
White	Female
White	Male
Asian	Male

White	Black	Asian	Male	Female
1	0	0	1	0
0	1	0	1	0
0	1	0	0	1
1	0	0	0	1
1	0	0	1	0
0	0	1	1	0

Such new features are known as

- Dummy variables
- Indicator features
- Binarized features

Use features to predict targets

Simple idea: Similarity

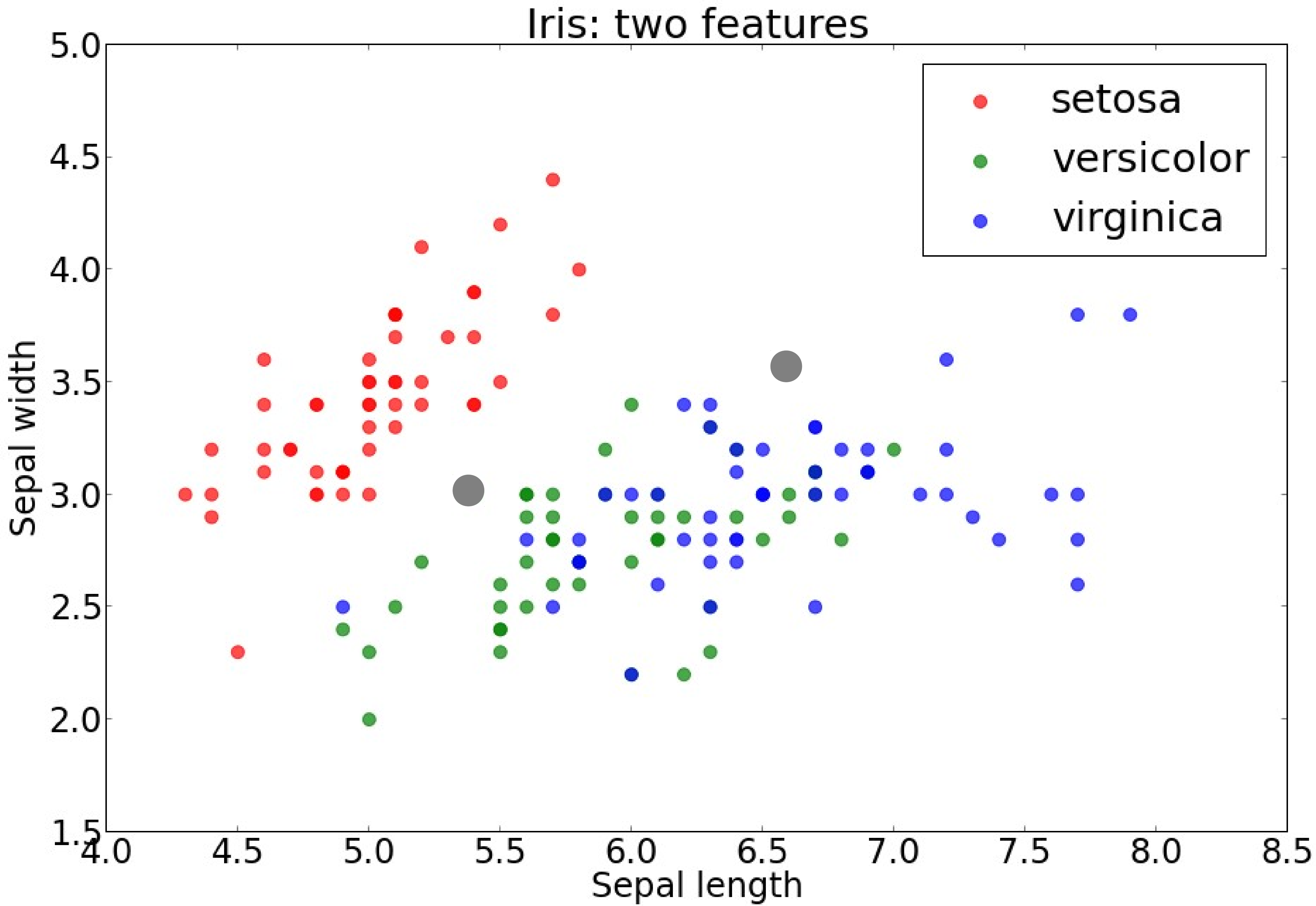
- Given a new example x_j
- We look for the most similar example in training set
- Predict the same target for x_j

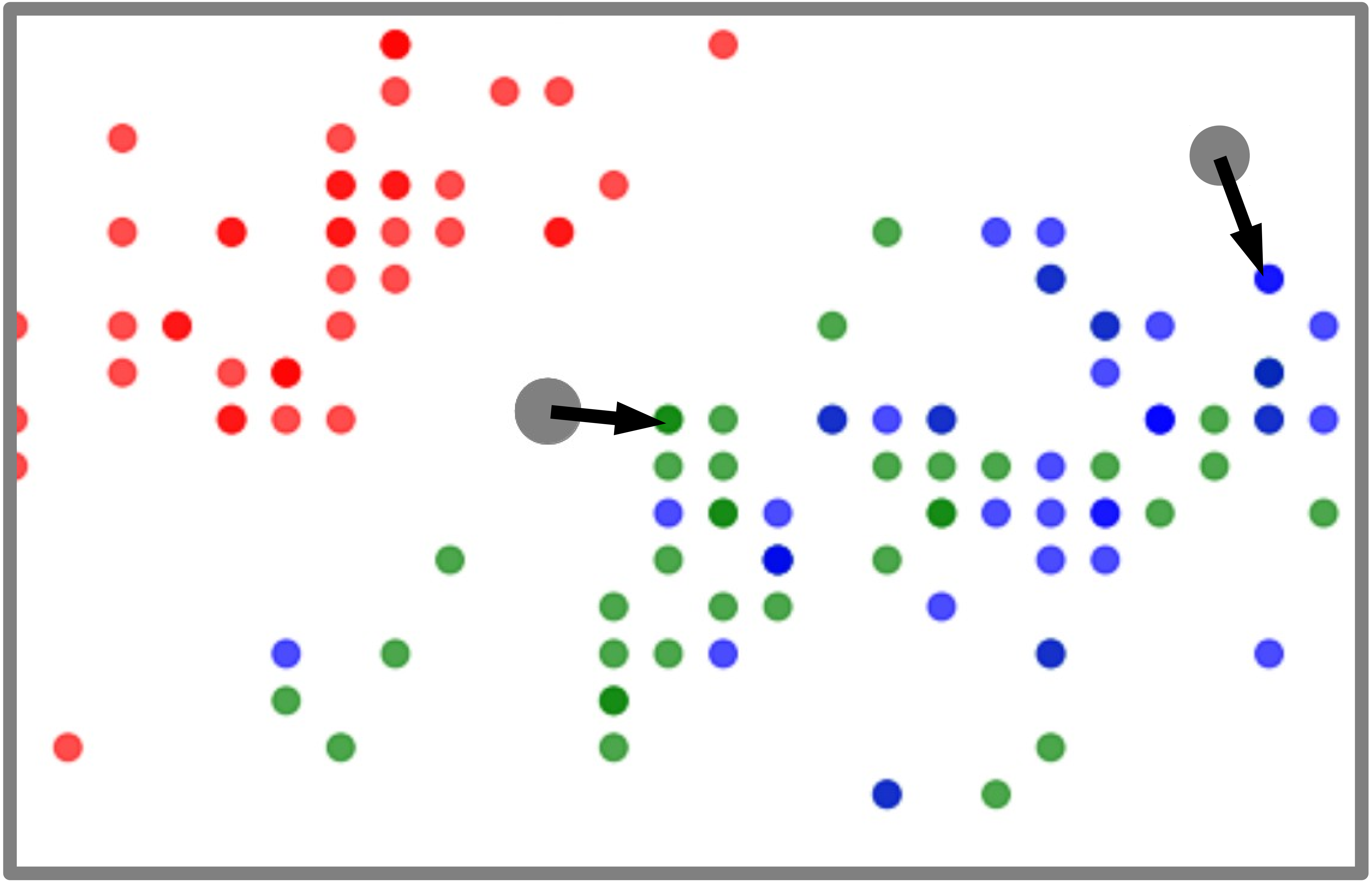
How do we measure similarity

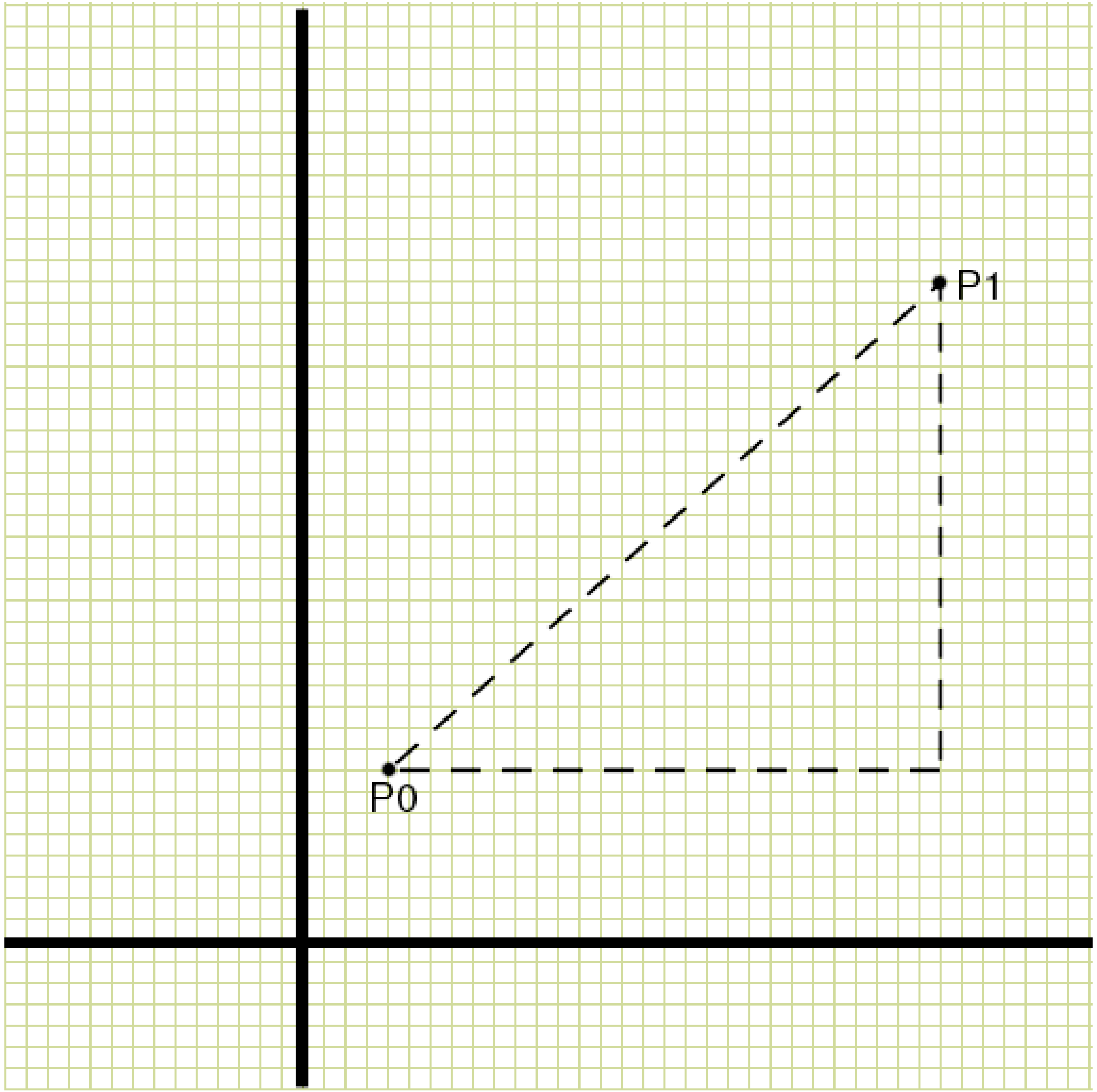
- Distance – opposite of similarity
- Find the nearest training example

How do we measure similarity

- Distance – opposite of similarity
- Find the nearest training example


Iris: two features
setosa
versicolor
virginica
Sepal width
Sepal length
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5





Source: http://resumbrae.com/ub/dms423/05/triangleOnCartesian.png

Distance

- Euclidean distance – like in physical space
  - In 2 dimensions

$$D(\mathbf{u},\mathbf{v}) = \sqrt{((u_1 - v_1)^2 + (u_2 - v_2)^2)}$$

  - In N dimensions

$$D(\mathbf{u},\mathbf{v}) = \sqrt{\left(\sum_{i=1}^{N}(u_i - v_i)^2\right)}$$

Finding the nearest neighbor

- Check the distances to all the training point, and pick the point with the smallest distance
- We need to remember the target of this point

Argmin

x_i^n – the i^{th} feature of the n^{th} example

y^n – the target of the n^{th} example

x^{new} – the new example

$y_{\text{pred}}^{\text{new}}$ – the prediction for the new example

$$NN = \arg\min_{n=1}^{N} D(\mathbf{x}^{\text{new}}, \mathbf{x}^{n})$$

$$y_{\text{pred}}^{\text{new}} = y^{NN}$$

Learning?

- In what sense can K-NN be said be learning?

- Illustrates most basic forms learning: **memorization**
- No **abstraction**

Assignment 1

For assignment 1 we will implement several functions needed for experiments with the Nearest Neighbor algorithm.

Exercise 1

- Define function `argmin` in Python
- This function should accept a list of values, and return the index of the minimum value

```
x = [10,11,4,5]
i = argmin(x)
print x[i]
```

→ 4

K-Nearest Neighbors

- Instead of only the closest example, we can look at several
- Predict the target which is most common among these

What to replace `argmin` with?

Exercise 2

- Define function `argsort` in Python
- It should accept a list of values, and return the list of indices sorted according to the values

```
x = [10,11,4,5]
print argsort(x)
→ [2, 3, 0, 1]
```

Summary

- In order to learn from examples we decompose complex objects into features
- Often we need to convert categorical features into indicator features
- K-NN exemplified learning-as-memorization