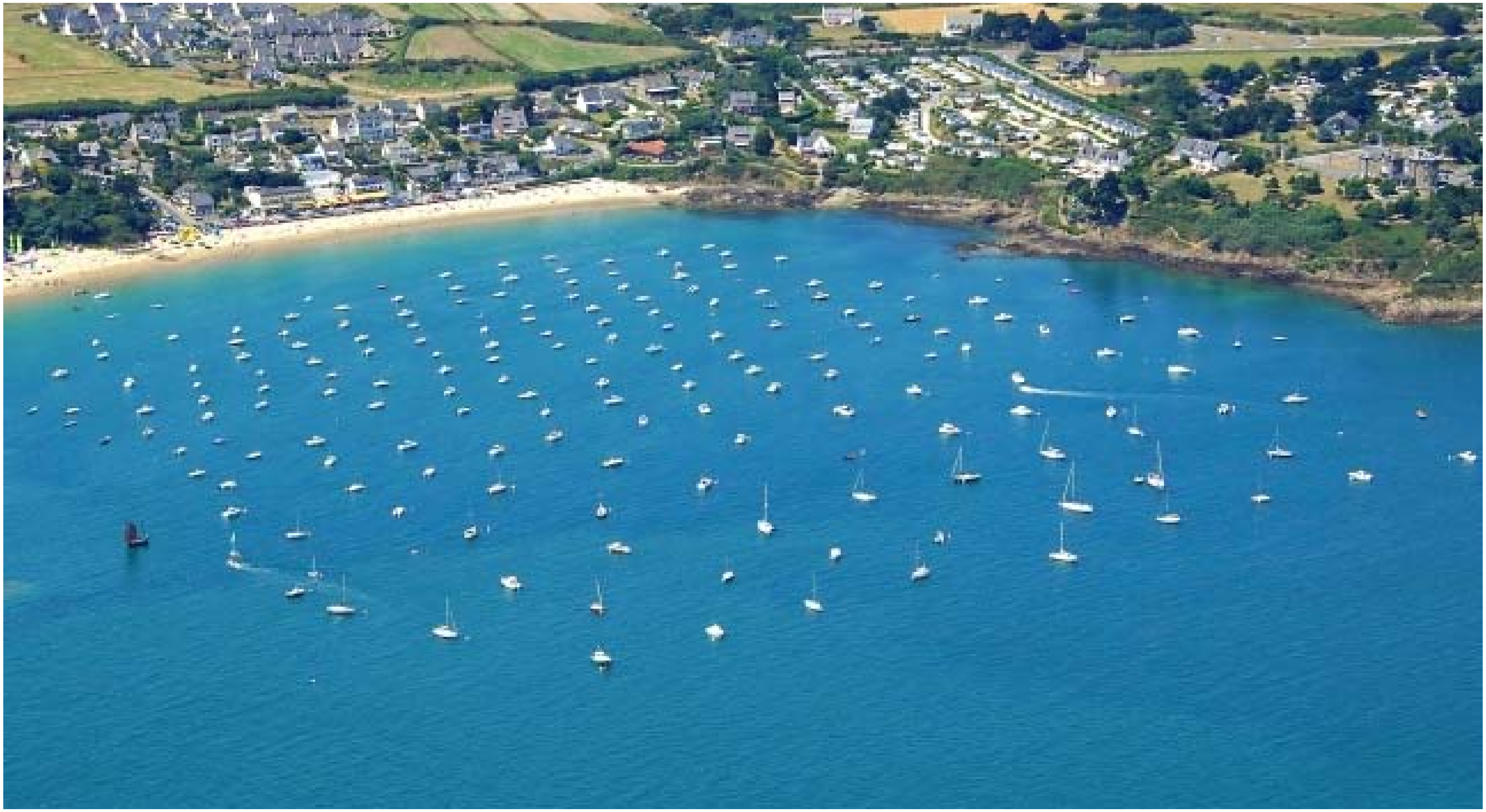
Support Vector Machines

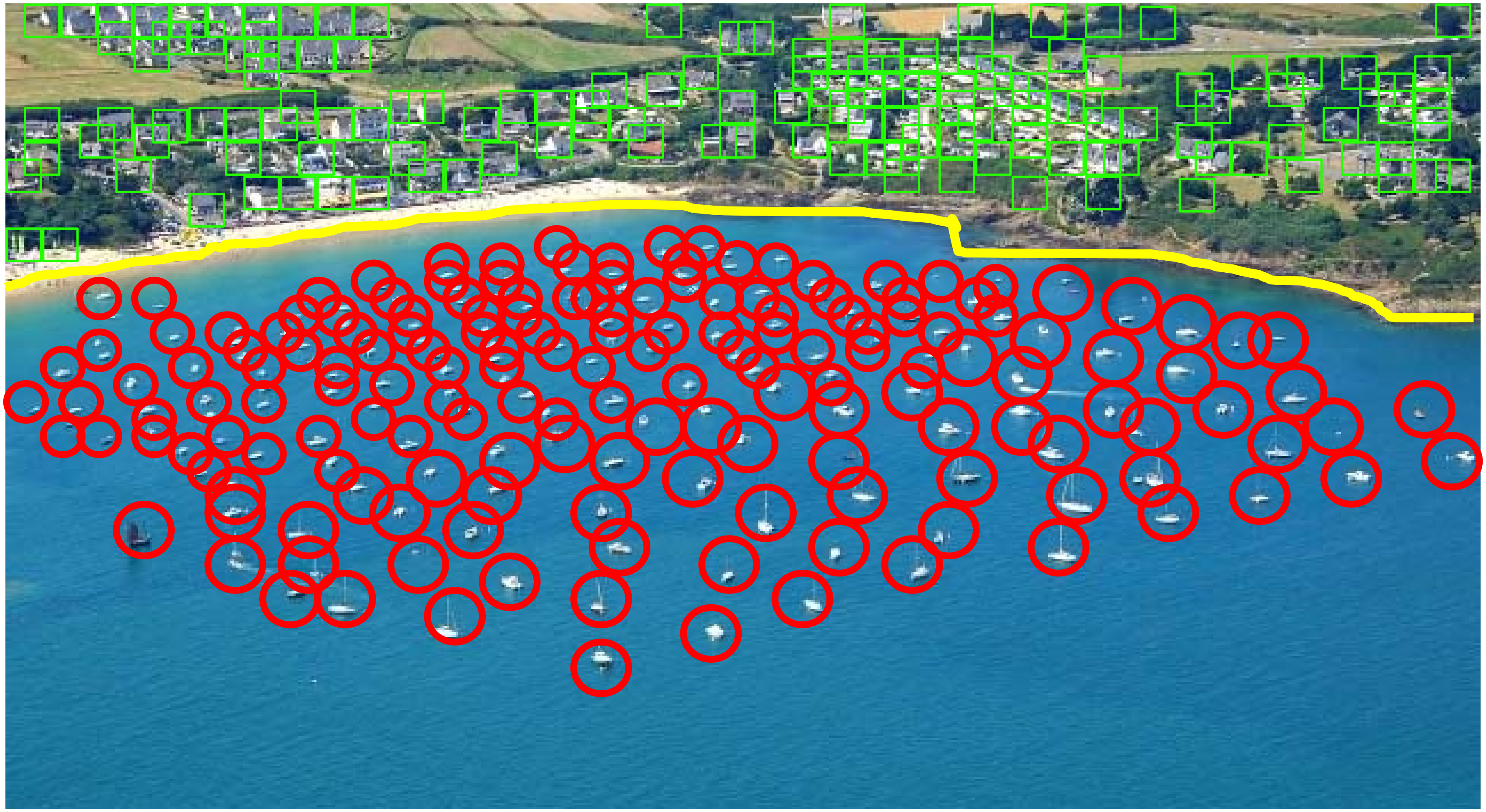
Harvey Alférez, Ph.D.

Basic principles of classification



- Want to classify objects as boats and houses.

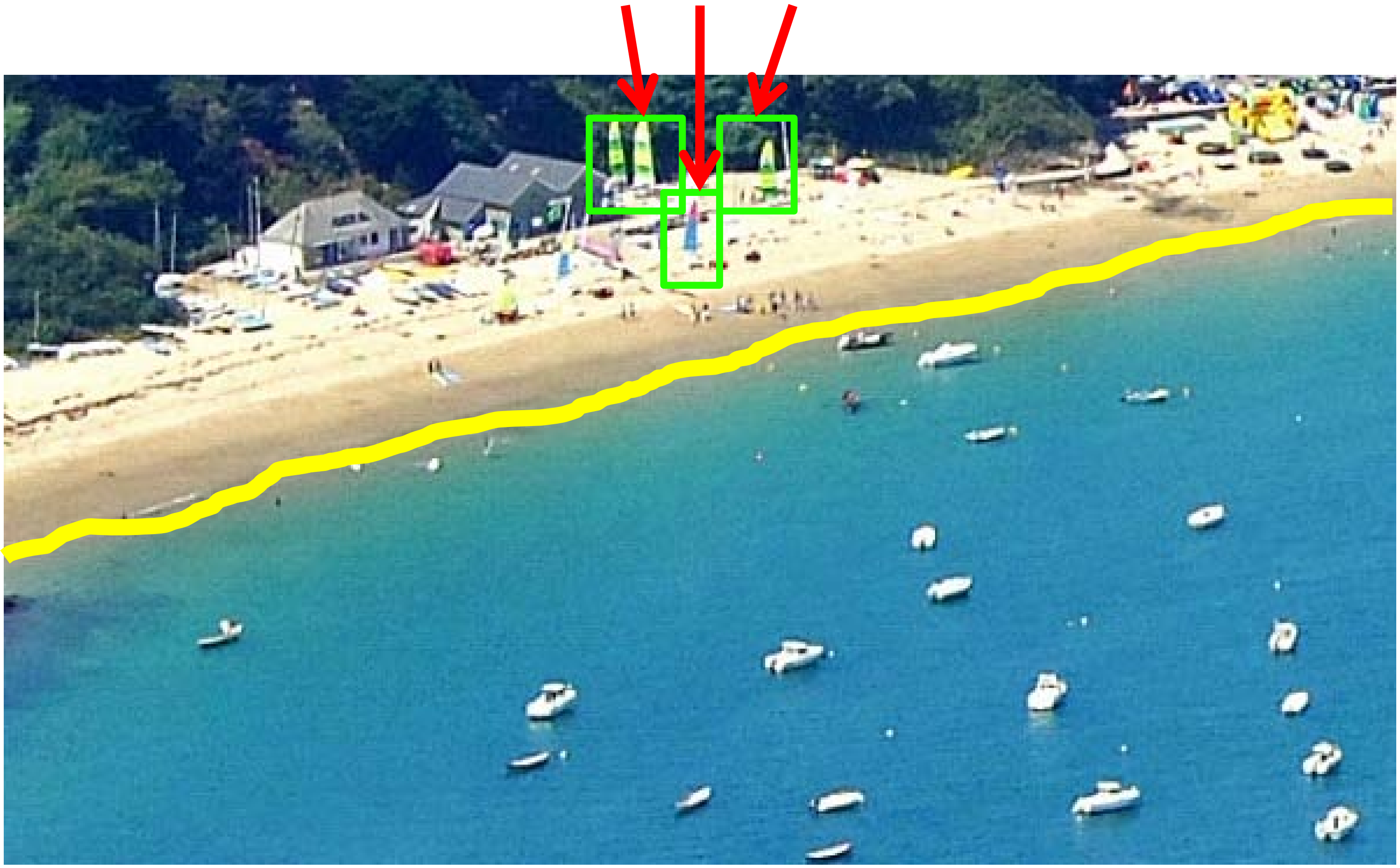
Basic principles of classification



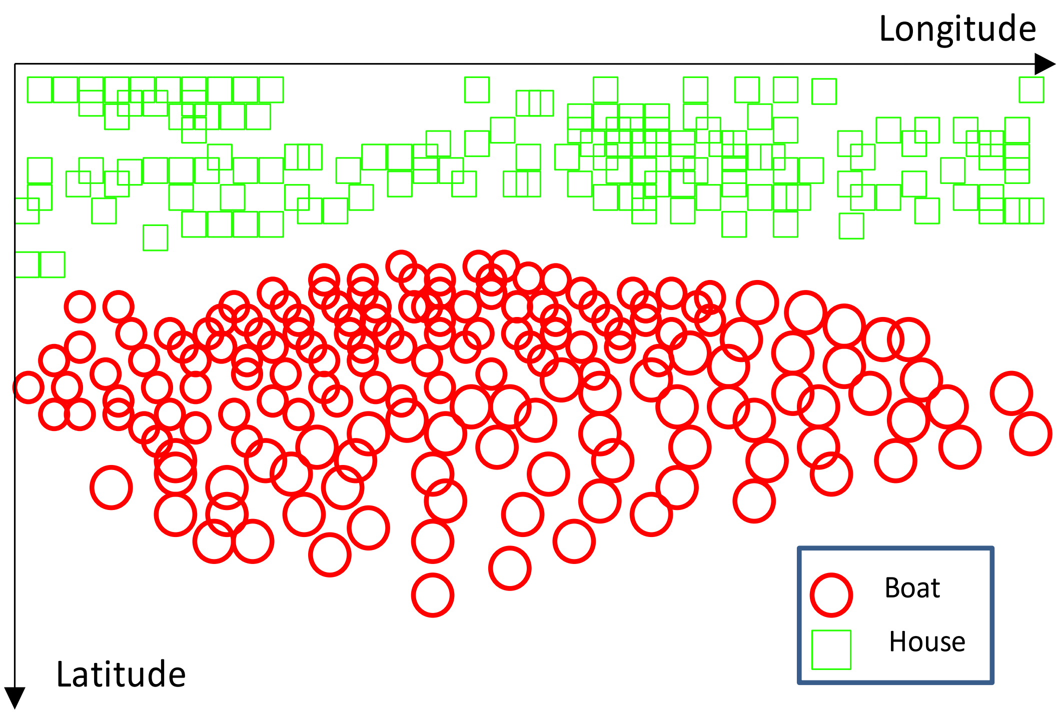
- All objects before the coast line are boats and all objects after the coast line are houses.
- Coast line serves as a *decision surface* that separates two classes.

Basic principles of classification

These boats will be misclassified as houses

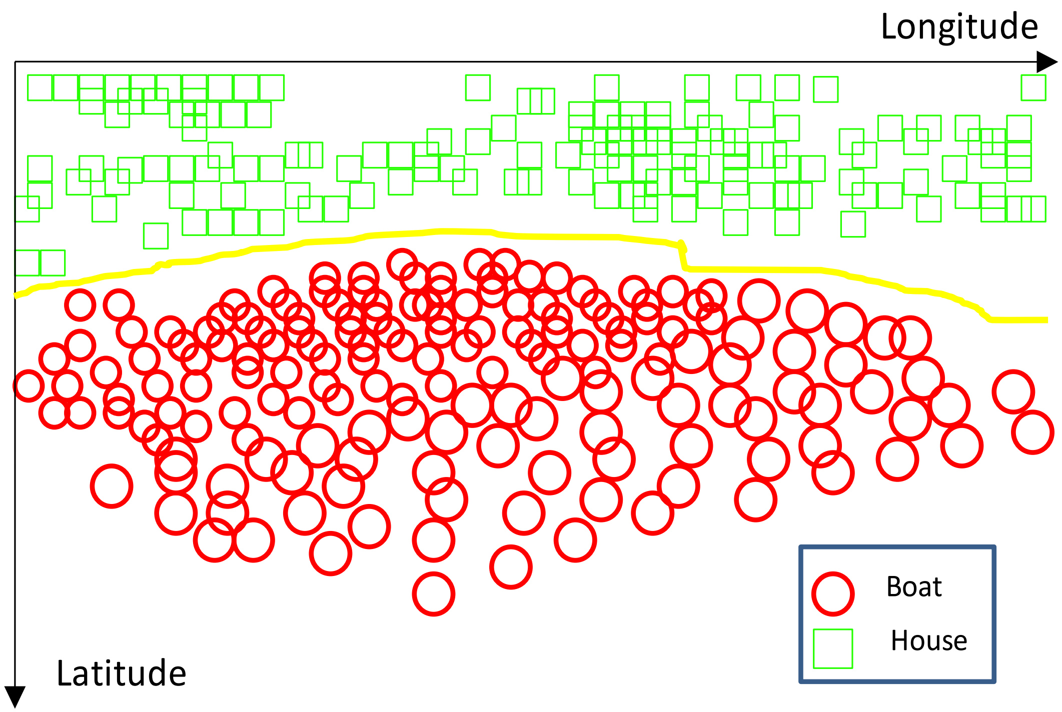


Basic principles of classification



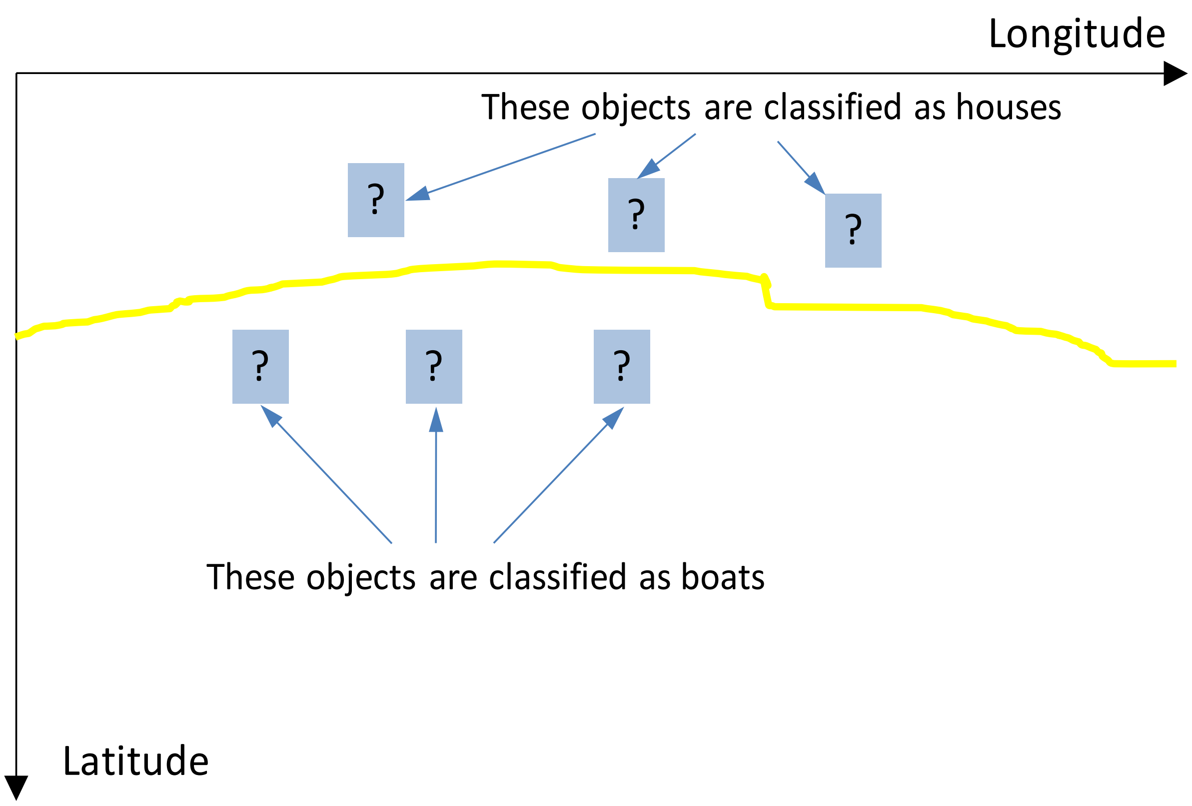
- The methods that build classification models (i.e., "*classification algorithms*") operate very similarly to the previous example.
- First all objects are represented geometrically.

Basic principles of classification



Then the algorithm seeks to find a decision surface that separates classes of objects

Basic principles of classification

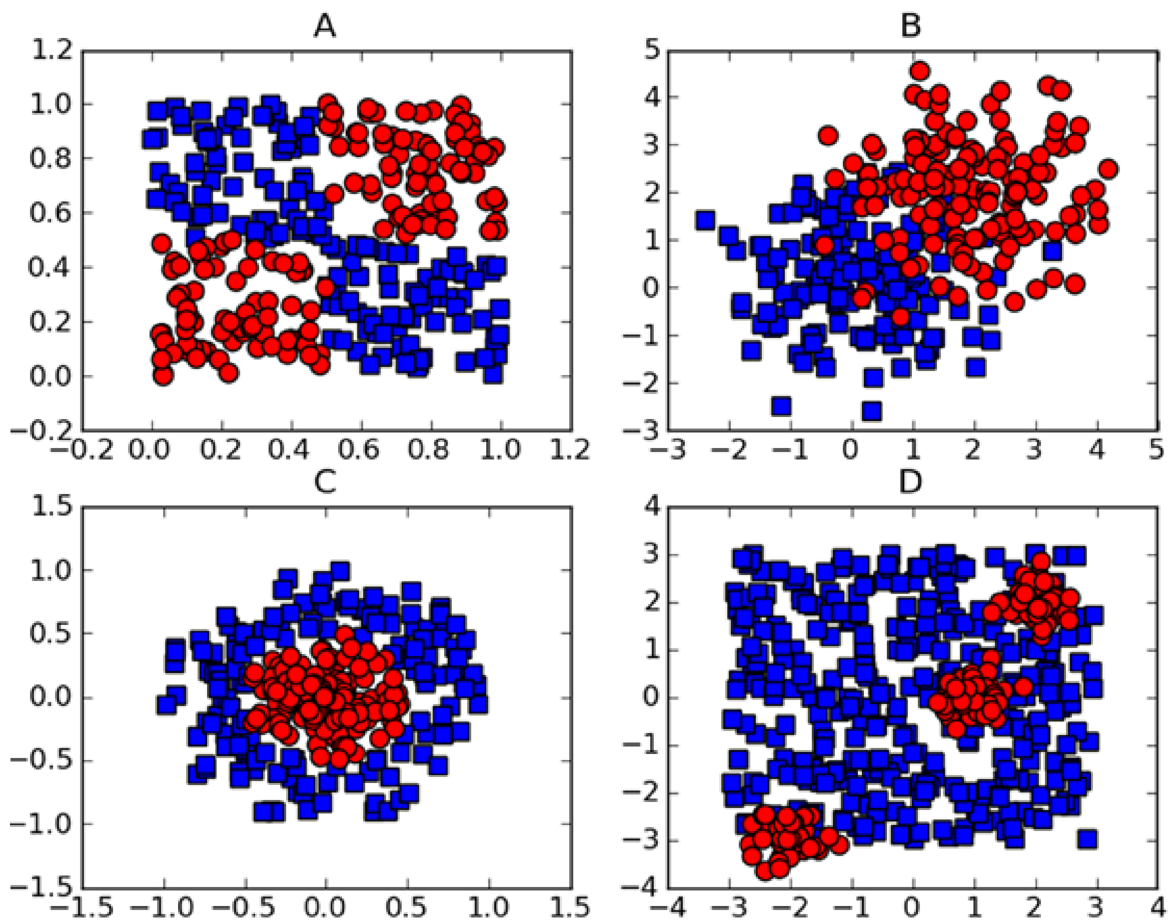


Unseen (new) objects are classified as "boats" if they fall below the decision surface and as "houses" if the fall above it

SVM

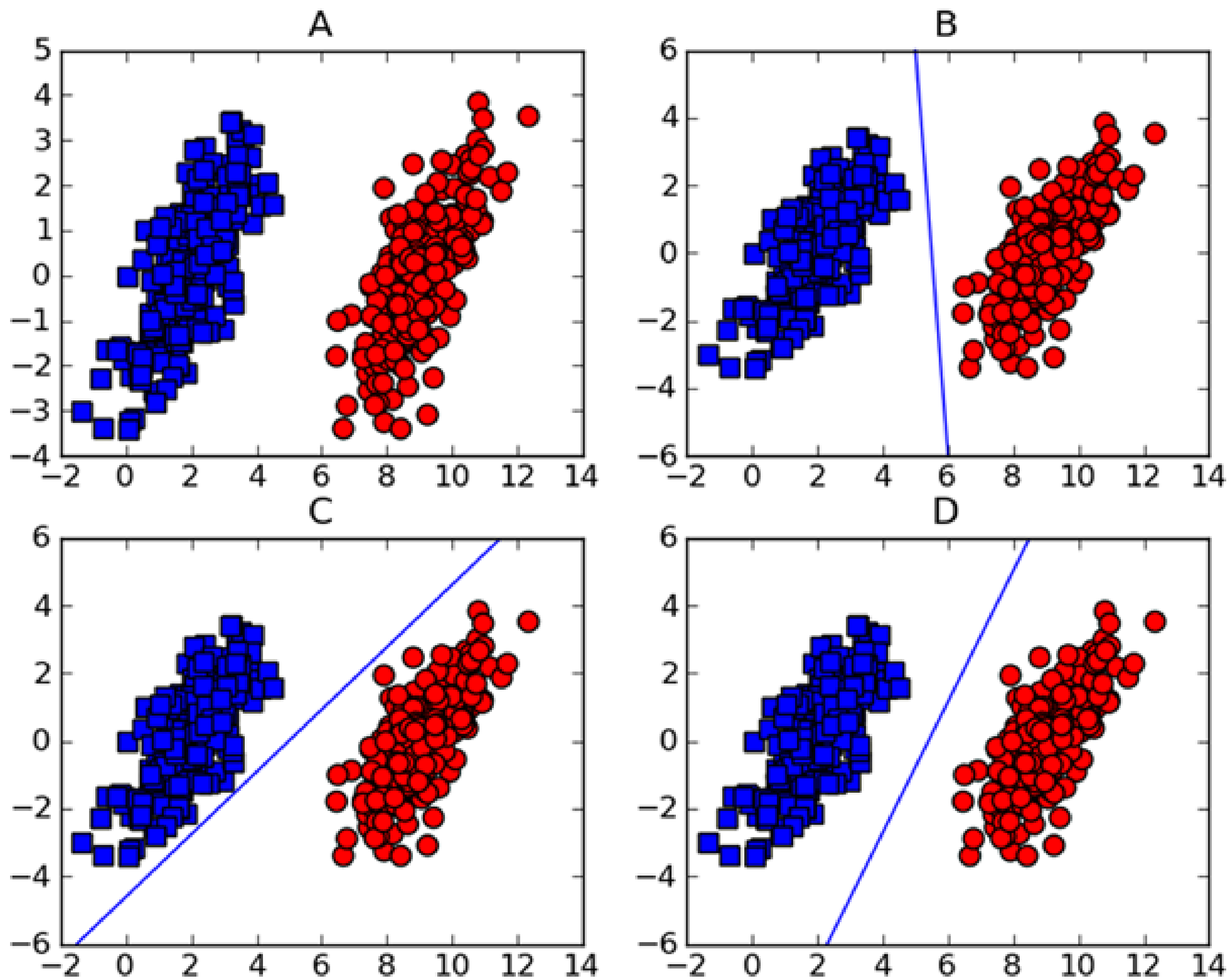
- **Support vector machines** are considered by some people to be the **best stock classifier**.
- By **stock**, I mean **not modified**.
    - This means you can take the classifier in its basic form and run it on the data, and the results will have low error rates.
- Support vector machines make **good decisions** for **data points** that are **outside** the **training set**.

Separating Data with the Maximum Margin



Could you draw **a straight line** to put all of the circles on one side and all of the squares on another side?

Separating Data with the Maximum Margin (Cont.)



Separating Data with the Maximum Margin (Cont.)

- The **line** used to separate the dataset is called a **separating hyperplane**.
  - In our simple **2D** plots, it’s just a **line**.
  - The **hyperplane** is our **decision boundary**.
    - Everything on one side belongs to one class, and everything on the other side belongs to a different class.

Separating Data with the Maximum Margin (Cont.)

- We'd like to make our classifier in such a way that the farther a data point is from the decision boundary, the more confident we are about the prediction we've made.

- Consider the plots in figure 2, frames B–D. They all separate the data, but which one does it best?

Separating Data with the Maximum Margin (Cont.)

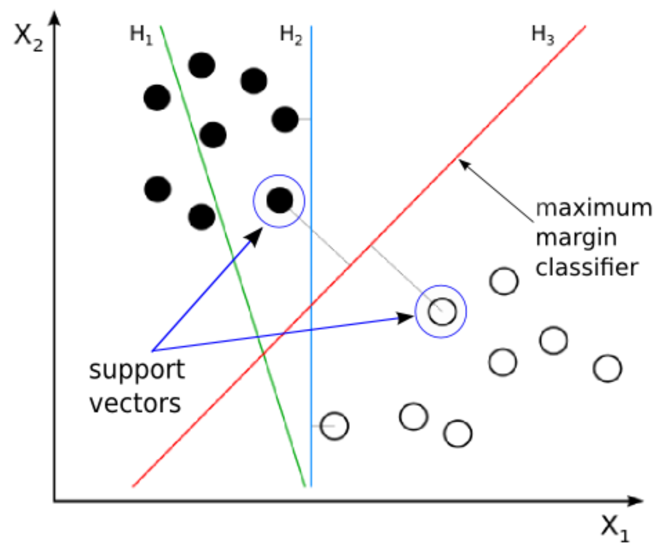
- We'd like to find the **point closest to the separating hyperplane** and make sure this is as **far away** from the **separating line** as possible.
  - This is known as **margin**.
    - We want to have the greatest possible margin!

Separating Data with the Maximum Margin (Cont.)

- The **points closest to the separating hyperplane** are known as **support vectors**.

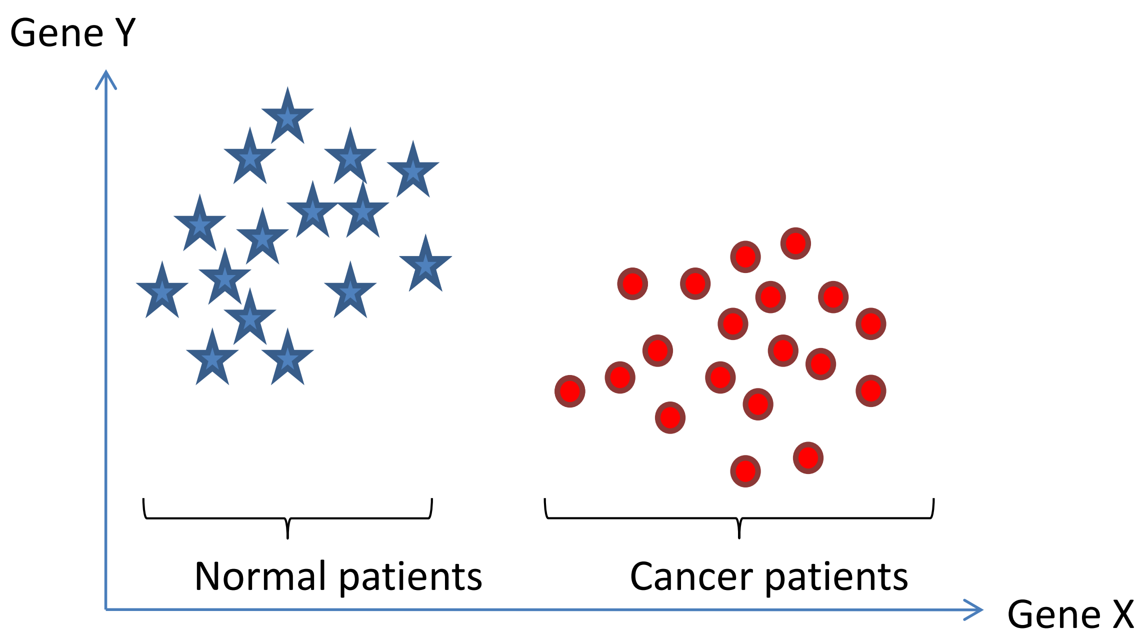
The simplest form of SVM is one that takes a set of *linearly separable* input data and predicts, for each given input, which of two possible classes forms the output. In other words: non-probabilistic binary linear SVM.

Consider the simple binary example below, where we're trying to find a hyperplane that can achieve optimal separation between two datasets.



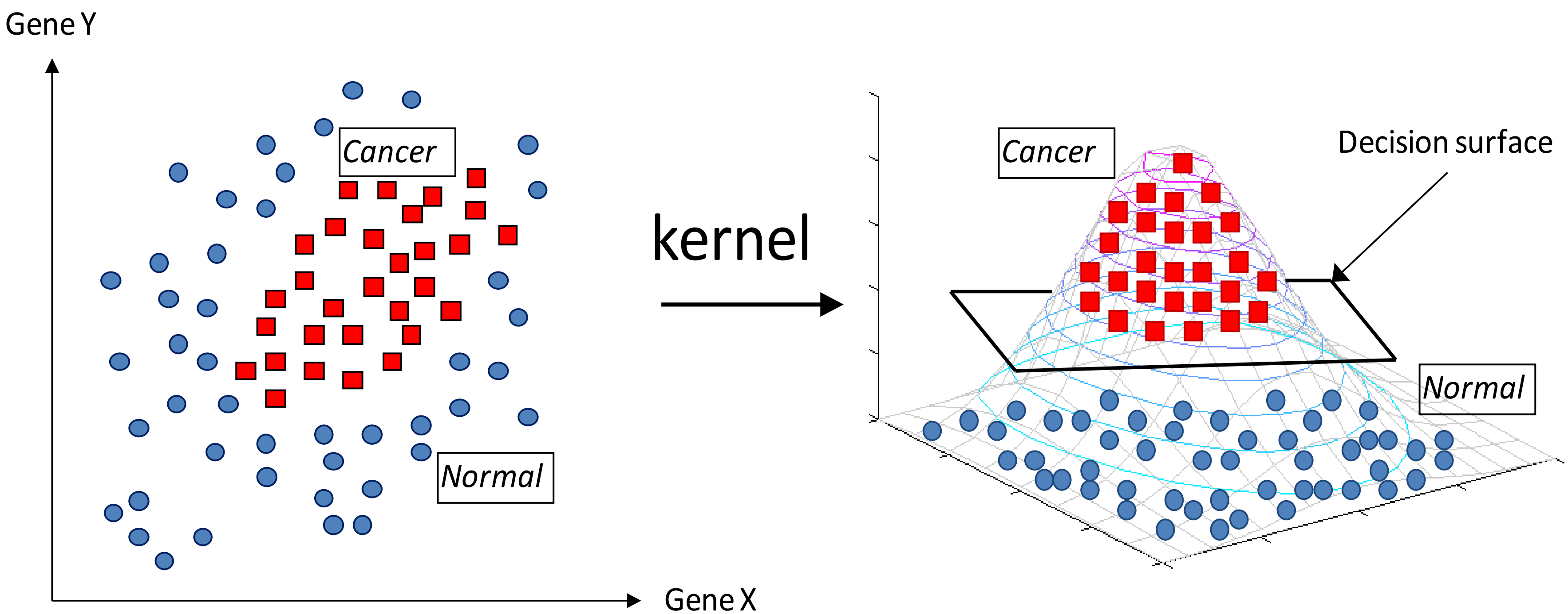
- H1 would be a pretty terrible hyperplane, as it groups some of the black points together with the white ones.
- H2 successfully separates the two datasets, but there is a very small margin between the black points and the separator.
- H3 can be seen as the optimal separator, since maximum margin was used.

Main ideas of SVMs



- Consider example dataset described by 2 genes, gene X and gene Y
- Represent patients geometrically (by "vectors")

Main ideas of SVMs

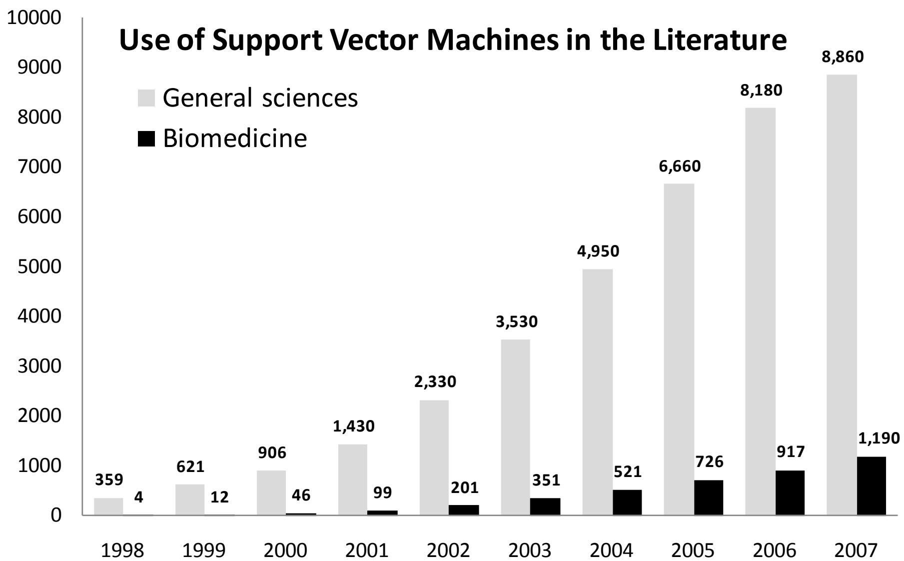


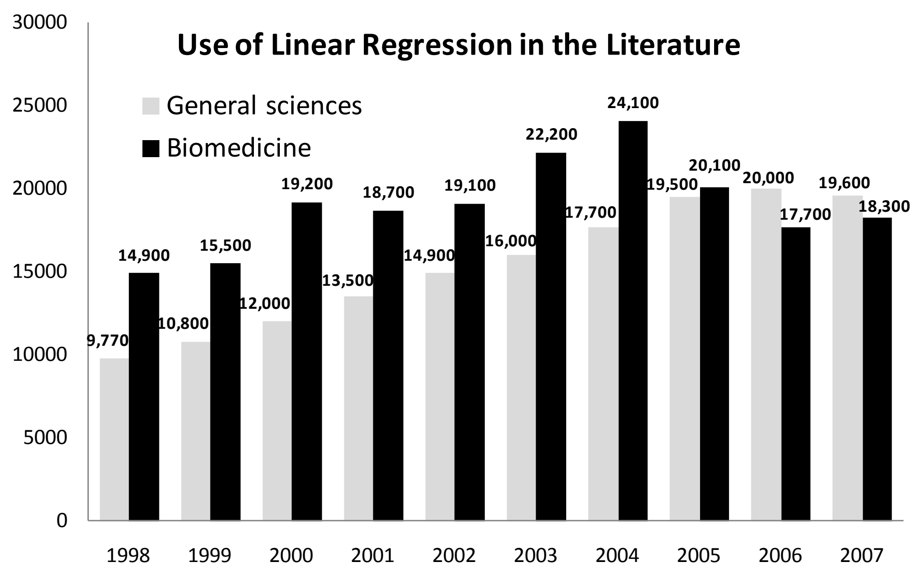
- If such linear decision surface does not exist, the data is mapped into a much higher dimensional space ("feature space") where the separating decision surface is found;
- The feature space is constructed via very clever mathematical projection ("kernel trick").

https://www.youtube.com/watch?v=3liCbRZPrZA

History of SVMs and usage in the literature

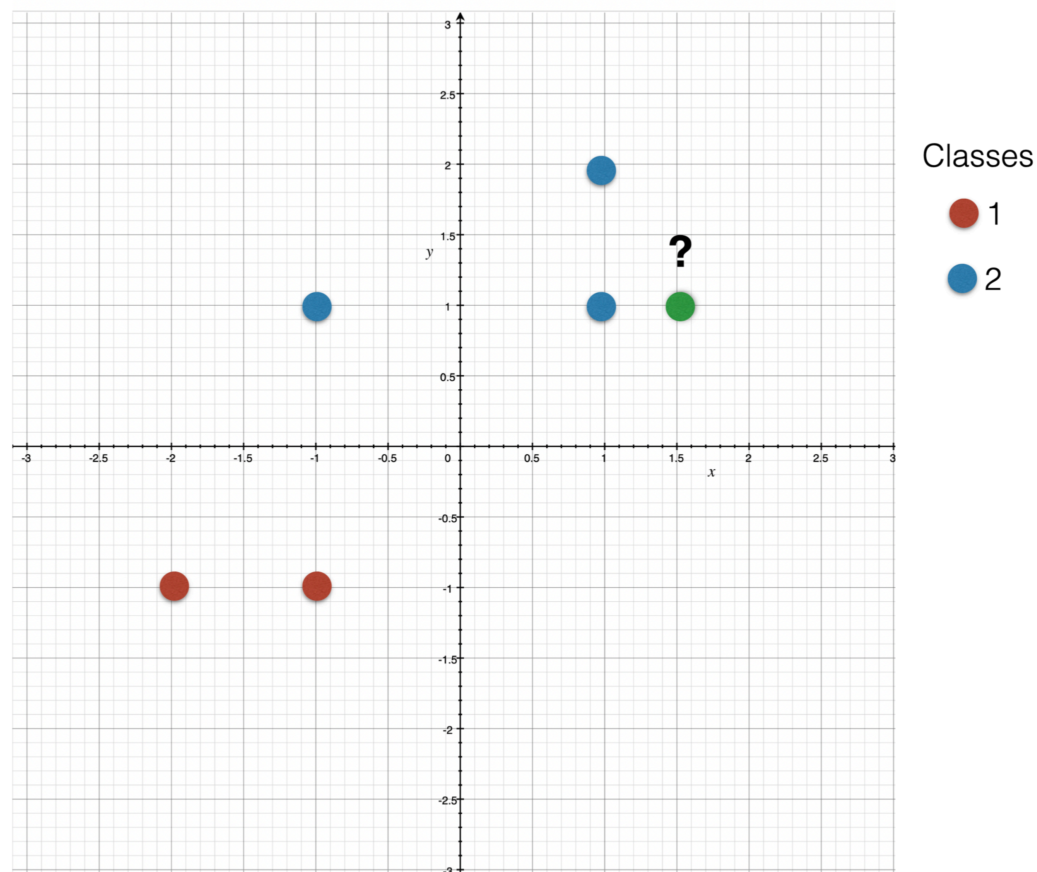
- Support vector machine classifiers have a long history of development starting from the 1960's.
- The most important milestone for development of modern SVMs is the 1992 paper by Boser, Guyon, and Vapnik ("*A training algorithm for optimal margin classifiers*")





```
import numpy as np
features = np.array([[-1, -1], [-2, -1], [1, 2], [-1, 1], [1, 1]])
labels = np.array([1, 1, 2, 2, 2])
from sklearn.svm import SVC
clf = SVC()
clf.fit(features, labels)
print(clf.predict([[1.5, 1]]))
```

```
[2]
```



Example 2

- http://scikit-learn.org/stable/auto_examples/svm/plot_iris.html