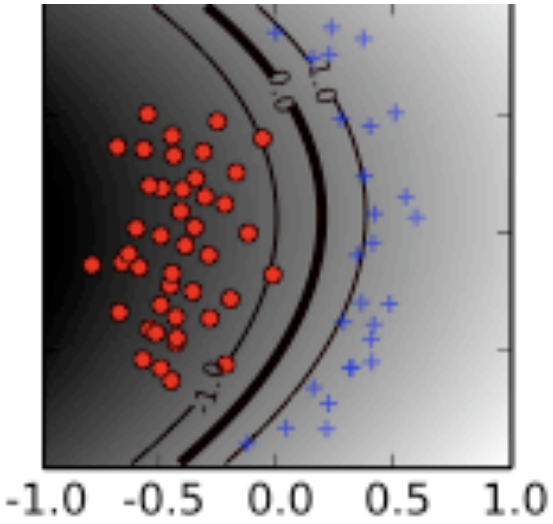
Learning from Data

The nearest neighbor algorithm (assignment 7)



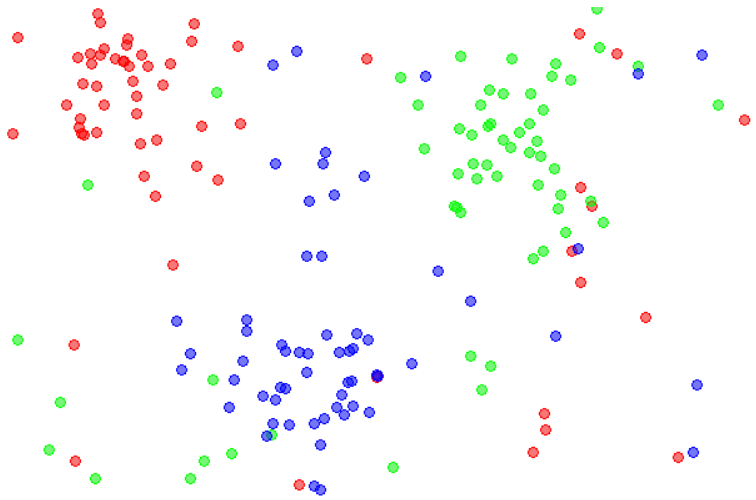
Pattern classification

- Classification: Given labeled examples of each digit, learn a classification rule

Examples of learning tasks

- OCR (Optical Character Recognition)
- Loan risk diagnosis
- Medical diagnosis
- Credit card fraud detection
- Speech recognition (e.g., in automatic call handling systems)
- Spam filtering
- Biometric identification (fingerprints, iris scan, face)
- Data mining, e.g. customer purchase behavior
- Bioinformatics: prediction of properties of genes and proteins.

Labeled data



Each example is a vector that has a label associated with it

- Representing the examples:
  - A two dimensional array

```
float[][] data = new
float[num_examples][dimensionality];
```

- The labels:
  - `int [] labels = new int[num_examples];`

The Nearest Neighbor Algorithm

NN(query):

1. Find the example in the training data which is closest to the query.
2. Return its label.

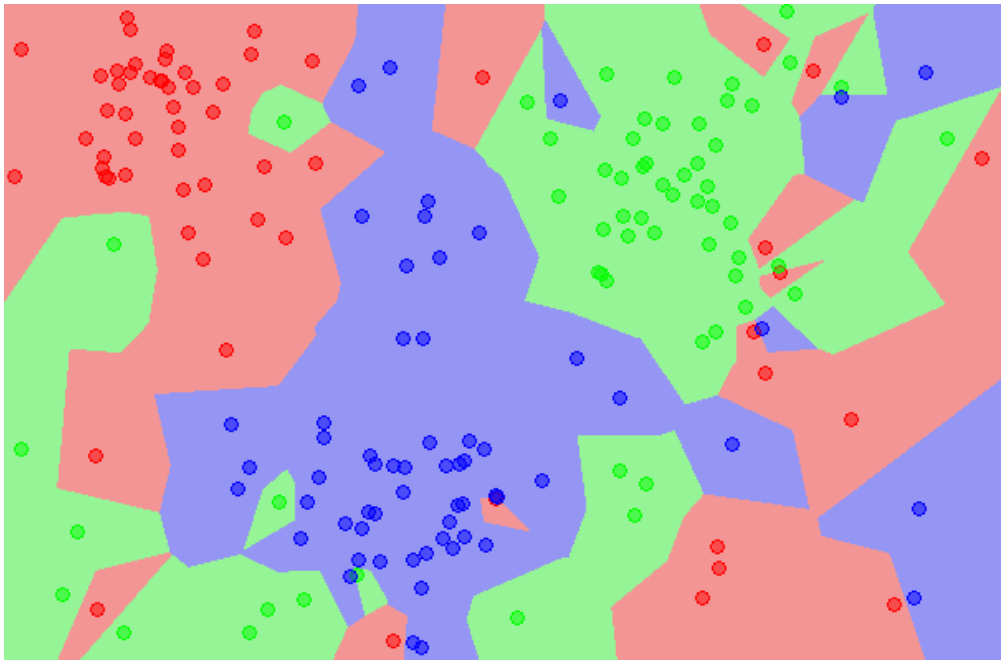
query

closest image

The Nearest Neighbor Algorithm

NN(query):

1. Find the example in the training data which is closest to the query.
2. Return its label.



Measuring distance

- How to measure closeness?

- Continuous data: Euclidean distance

  Let **x**=(x_1,…,x_d) and **y**=(y_1,…,y_d) be vectors of real numbers.

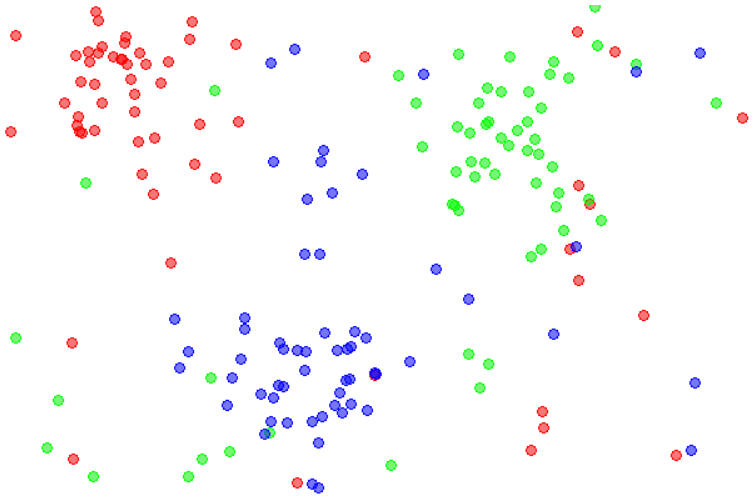
$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$$

k-NN

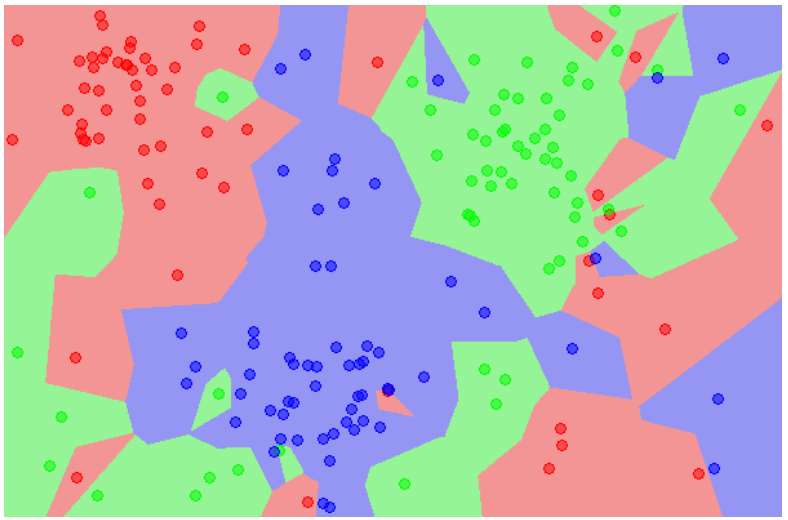
- Use the closest k neighbors to make a decision instead of a single nearest neighbor
- Choose the label that occurs among the majority of the k nearest neighbors

- Why do you expect this to work better?

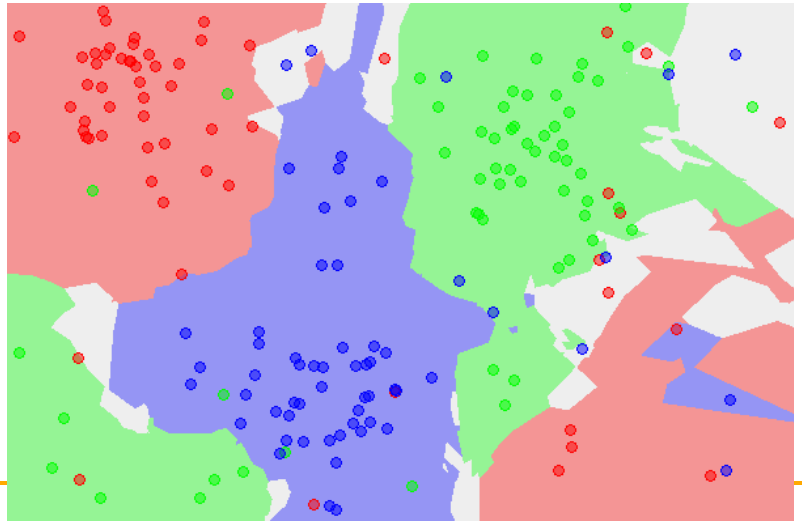
The data



Using
1 NN



Using
5 NN



Images from http://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

Measuring classifier performance

- To measure how well a classifier is working we compute its accuracy on data that is not part of the dataset used by the classifier.
- Accuracy: how many examples were classified with the correct label.

The assignment

- You will need to implement the following classes:
- **Dataset** – implements the labeled data interface
  - Constructor loads dataset from file
  - getExample(int i) – returns the ith example
  - getLabel(int i) – returns the label of example i
- **KNN** – implements the classifier interface
  - int classify(Dataset data, int i) – returns the label predicted for example i in the dataset
- **Evaluator** – runs a classifier on a given dataset and computes its accuracy