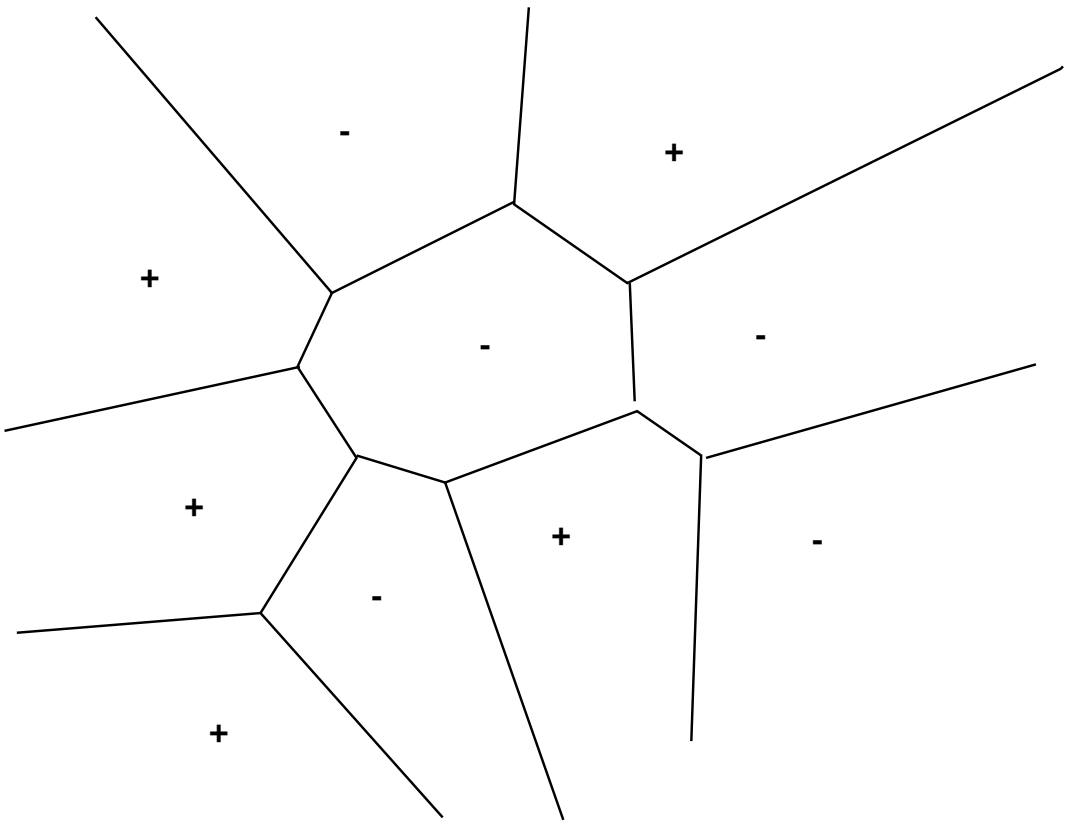
Nearest neighbor learning

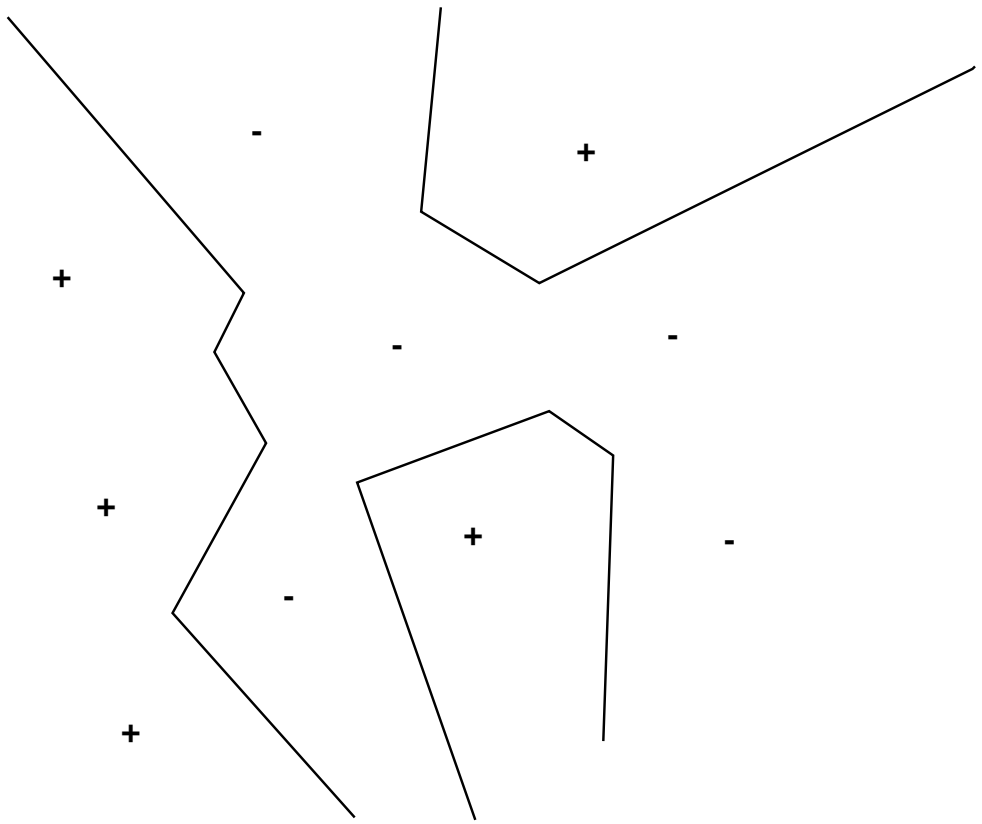
Nearest neighbor / "memory-based learning"

- Define some notion of "distance" between points (a metric).
- Remember your training data in a table or other data structure.
- Given a test example, find the closest training example (according to your metric) and output its classification.

WYSIWYG learning.



(picture is "Voronoi diagram")



- Similar to growing decision tree all the way.

- Also k-nearest neighbor: take majority vote over k nearest.

General aspects

Advantages:

- Training is fast.
- Performance on training set.
- Don't lose information

Problems:

- Lookup can be expensive computation. But, good ways to store in special cases:
  - 2-dim (low dim): decision trees where points near border (Voronoi cells have points on both sides) go both ways.
  - Exact or near exact matches: use hashing.
  - Easily parallelizable.
- What metric to use?

  Representing image as 960-dimensional point is not so good.
- Gets easily fooled by irrelevant info (bad metrics).

Behavior "in the limit"

Say target is a piecewise continuous function: $\texttt{instance space} \rightarrow [0, 1]$.

$f(x) =$ probability x gets classified as positive.

Nearest neighbor:

- As number of points $\rightarrow \infty$, nearest neighbor approaches "Gibbs Algorithm".

  Gibbs: with probability $f(x)$ predict 1, and with prob $1 - f(x)$ predict 0.

k-nearest neighbor:

- As number of points $\rightarrow \infty$ and k gets large, k-nn approaches Bayes-optimal algorithm.

  Bayes optimal: if $f(x) > 0.5$ predict positive and if $f(x) < 0.5$, predict negative.

Gibbs at most twice as bad as Bayes optimal. E.g., if $f(x) = 0.9$, then prob of error of Bayes is 0.1. Prob of error of Gibbs is $(0.1 \times 0.9) + (0.9 \times 0.1) = 0.18$.

MBRtalk: nearest-neighbor based pronunciation

[Stanfill and Waltz '86]

Problem: Given database of English words and their pronunciation, learn to pronounce new words.

Plan: For each word, for each position (field), store a record with 7 letters (3-1-3) and correct pronunciation. [Also use (4-1-10)].

```
...goal  g
..goal.  O
.goal..  -
goal...  l
```

Given new word to predict, make record for each letter and predict according to closest match in database.

Question: What metric to use?

Metrics for MBRtalk

- "Overlap metric": number of positions with different values.

  problem - all weight positions weighted equally, even target. good/hood

- "Weighted feature metric": how much variation is there in desired pronunciation given that position p has letter l? Use to weight.

- Others....

Performance

- Whole words on test set: 47% correct, 21% marginally wrong, 32% badly wrong. Phonemes: 86%.

  (marginally wrong look pretty good)

- Comparable to NETtalk.

- Bad cases: barroom (sounds like barren), item (it'm), libel (libbel), surveillance (serve lance), who (w).

Case-based reasoning

Idea: use nearest neighbor for planning.

- Store previously seen cases.
- Given new situation, pull up "most relevant" case in database.

  Issue: what types of features to use for indexing.

- Use the case to guide new plan.

Uses of CBR

- MEDIATOR [Kolodner,Simpson,Sycara '85] Dispute resolution. Focuses on 10 specific tasks.

- CHEF [Hammond '90] Plans Szechwan meals. Remembers successes and failures (soggy vegetables). Abstract features.

- HOLS [Craig '92] Holiday planner. Combining several agents to plan for a group.

- Analogy in general problem solving in PRODIGY [Veloso '92]. (combination of case-cased and explanation-based).

- ...