## Review of Clustering Concepts

## Data Representation

- Most common is an N dimensional “feature” vector.
- Most common distance is Euclidian distance.
- Be careful with scaling and units!
- Probabilistic modals finesse multiple modalities
- Problems with correlated variables can be mitigated using transformations and data reduction methods such as PCA, ICA.

## Insert Lecture 20 Here

- In 2008 Ranjini finshed clustering as lecture 20, which should be inserted here for proper sequence.

## Fitting

- Work with a parametric representation for “objects”
  - (e.g “line”, “ellipse”).
- Most interesting case is when criterion is not local
  - can’t tell whether a set of points lies on a line by looking only at each point and the next
- Three main questions:
  - what object represents a given set of tokens best?
  - which of several objects gets which token? (**correspondence**!)
  - how many objects are there?

## Example: Hough Transform for lines

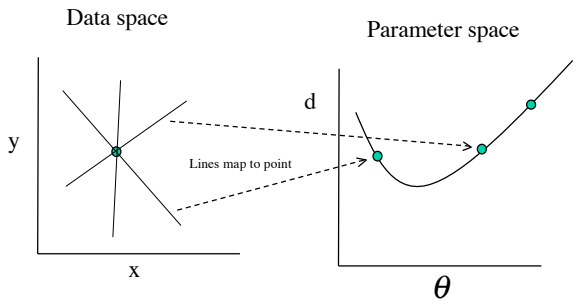
- A line is the set of points (x, y) such that

$$(\sin\theta)x + (\cos\theta)y + d = 0$$

- Different choices of θ, d>0 give different lines
- For any (x, y) there is a family of lines through this point, given by

$$(\sin\theta)x + (\cos\theta)y + d = 0$$

- The choice of θ fixes d. The family of lines has **one** parameter.




$$(\sin\theta)x + (\cos\theta)y + d = 0$$

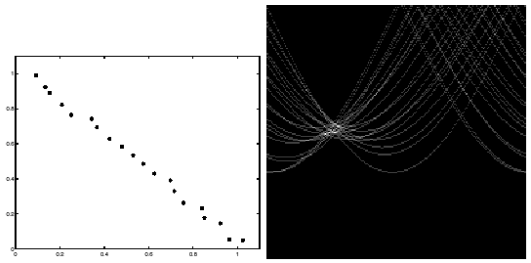
(Note: Curve is **not** accurate)

## Example: Hough Transform for lines

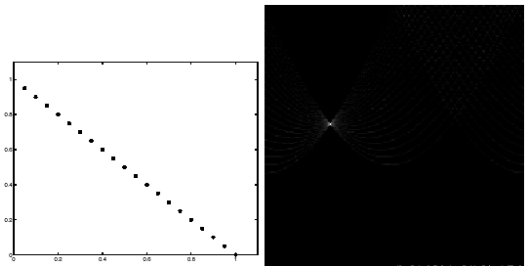
- Main idea: Each observed (x,y) votes for all (θ, d) satisfying

$$(\sin\theta)x + (\cos\theta)y + d = 0$$

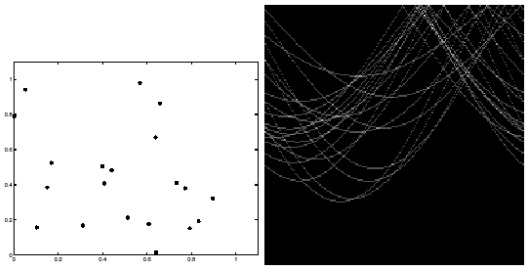
- Discretize the parameter space (θ, d) by an array
- Now each (x,y) leads to a bunch of votes (counts) in a (θ, d) grid (along the curve in the preceding slide).
- To find lines, let all edge points (x,y) vote, and look for (θ, d) cells with lots of votes.



tokens votes



tokens votes

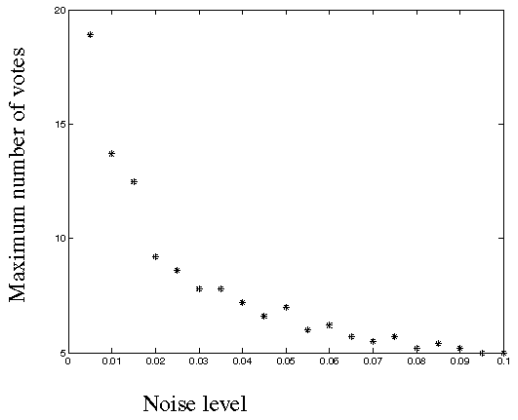


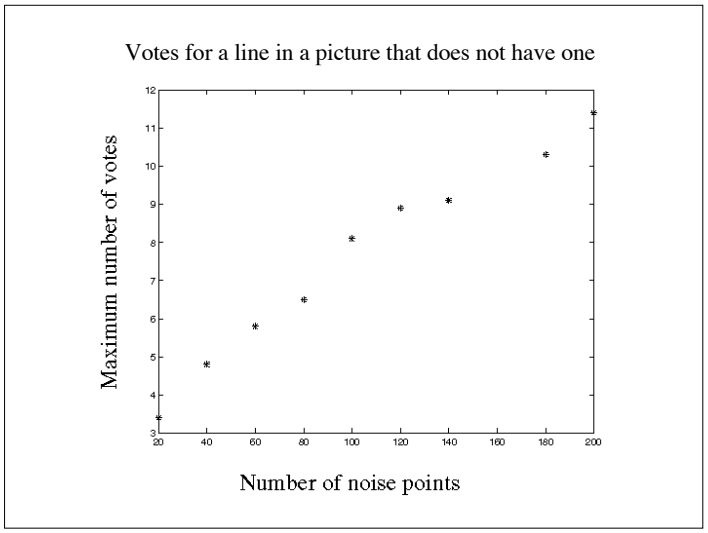
tokens votes

## Difficulties with the Hough transform

- How big should the cells be? (too big, and we cannot distinguish between quite different lines; too small, and noise causes lines to be missed)
- How many lines?
  - count the peaks in the Hough array
- Who belongs to which line?
  - tag the votes
- Hough transform is a useful idea, but it is not often satisfactory in practice, because problems with noise and cell size defeat it

Votes for a real line of 20 points versus noise




Votes for a line in a picture that does not have one
Maximum number of votes
Number of noise points
12
11
10
9
8
7
6
5
4
3
20
40
60
80
100
120
140
160
180
200