• Goals for this lab are:
  
  - NLTK documentation
  
  - Try some NLTK clustering algorithms
  
  - Examine the results
• Root is at

• Package we are interested in today is nltk.cluster

• Pretty standard documentation structure
Using k-means clusterer

- Initialize a clusterer.
  - Must give it # of clusters and distance metric
    - Euclidean_distance, cosine_distance
  - Some defaults
    - repeats=1,
    - conv_test=1,e-0,6,
    - initial_means=None

- Create clusters using clusterer we initialized
  - Must give it vectors
  - Defaults
    - assign_clusters=False
    - trace=False
Vectors for clustering

• Array of vectors with $n$ elements
  • Each document is one vector
  • $n$ is the number of features
• Actually a numpy array
• Demo methods in the clusterers.
Another Example

Four documents: “Computer science is a STEM technology.”
“Biology is a STEM technology.” “Philosophy is a liberal art.”
“History is a liberal art.”

Assign positions to terms: 1:Computer 2:science 3:is 4:a
5:STEM 6:technology 7:Biology 8 philosophy 9:liberal 10:art
11:history

Computer science is a STEM technology: [1, 1, 1, 1, 1, 1, 0, 0,
0, 0, 0]

Biology is a STEM technology: [0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0]

Philosophy is a liberal art: [0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0]

History is a liberal art: [0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1]
Okay, Try Some Stuff

- Create some simple examples and try the various parameters of the various clusterers, using the API to see what’s what.

- Get your own corpus in, do simple word counts, and cluster your documents.
More Useful NLTK stuff

• We have mostly been working with the nltk.text module. It has useful methods like
  
  • count: takes a word, gives # of times it occurs
  
  • vocab: produces a frequency distribution of terms in a text object
  
  • it has a class nltk.text.TextCollection, which has a tf*idf method
And more

• For GAAC clusterer:
  • NO required parameters, typical to give it # of clusters.

• For EM clusterer
  • The only required parameter is initial_means: the means of the gaussian cluster centers.
  • Plus a bunch of things which have reasonable defaults.