Python for Computational Linguistics

Damir Ćavar
dcavar@indiana.edu
dcavar@unizd.hr

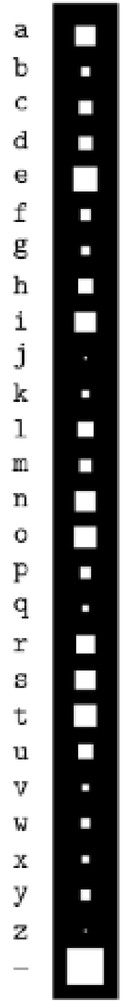
DGfS Herbstschule in Bochum

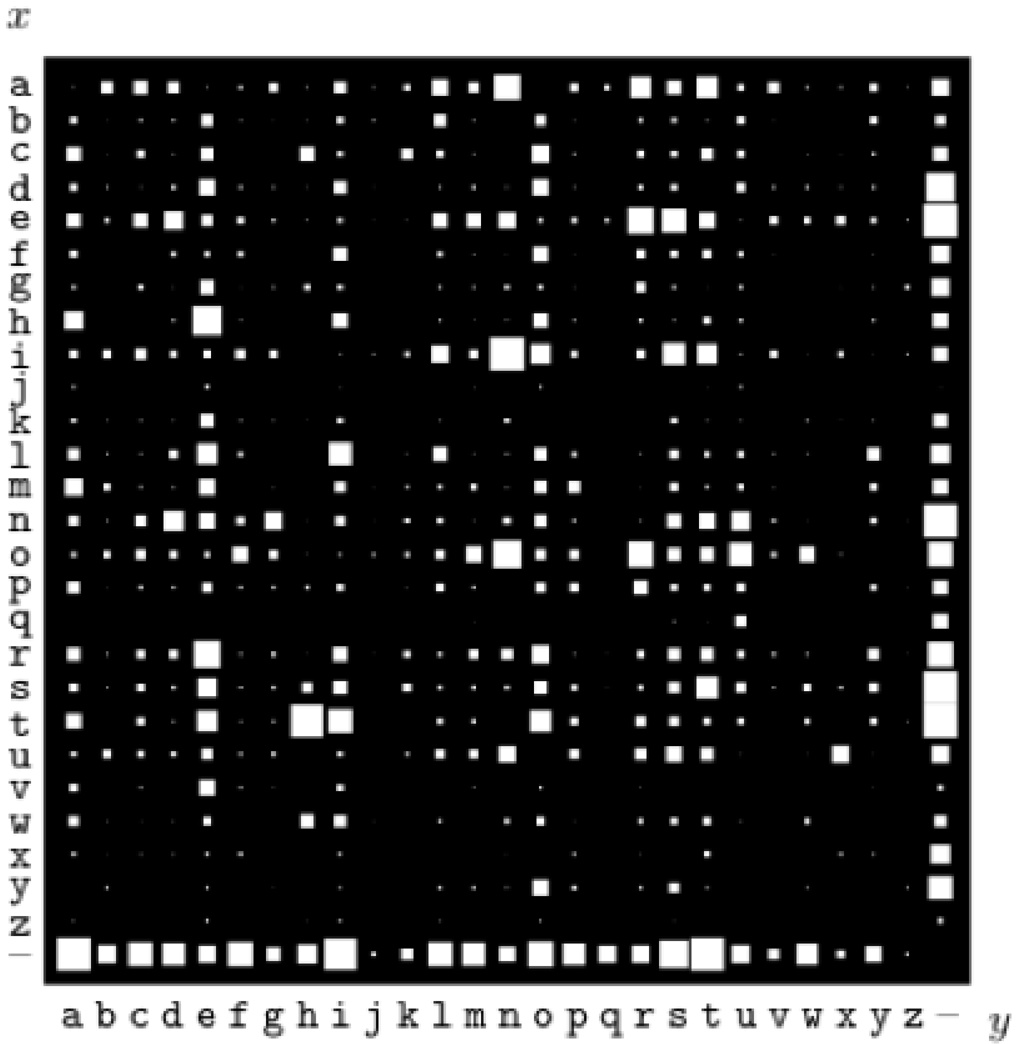
September 2005

Frequency Profiles

- *Uni-gram* frequencies: freq.py, . . .
- *Bi-gram* frequencies:
- General *n*-gram models
- Examples: from [MacKay(2003)]

i	a_i	p_i
1	a	0.0575
2	b	0.0128
3	c	0.0263
4	d	0.0285
5	e	0.0913
6	f	0.0173
7	g	0.0133
8	h	0.0313
9	i	0.0599
10	j	0.0006
11	k	0.0084
12	l	0.0335
13	m	0.0235
14	n	0.0596
15	o	0.0689
16	p	0.0192
17	q	0.0008
18	r	0.0508
19	s	0.0567
20	t	0.0706
21	u	0.0334
22	v	0.0069
23	w	0.0119
24	x	0.0073
25	y	0.0164
26	z	0.0007
27	–	0.1928




x
a b c d e f g h i j k l m n o p q r s t u v w x y z _
a b c d e f g h i j k l m n o p q r s t u v w x y z _
y

Frequency Profiles

- What can we do with *n*-gram frequency profiles?
  - Compression, modeling expectations, study of quantitative language properties, ...
- What value for *n* is best for what purpose?

Language Identification

- *N*-gram models for Language Identification
- Files: `lid.py`, `lidtrainer.py`, `*.dat`
- Calculations:
  - Mean of frequencies
  - Deviation

Numerical Statistics

- Measures of central tendencies of data
  - Mean
  - Median
  - Mode
- Measures of variation/variability
  - Spread in data

Numerical Statistics

- Arithmetic Mean
  - Data set:

File	Count words
Flo031201.txt	10346
Flo031202a.txt	5031
Flo031202b.txt	11876
Flo031203.txt	12175
Flo031204.txt	10943

Numerical Statistics

- Arithmetic Mean

$$arithmetic\ mean = \frac{sum\ of\ measures}{number\ of\ measures}$$

  – example:

$$\frac{10346 + 5031 + 11876 + 12175 + 10943}{5} = 10074.2$$

Numerical Statistics

- Median
  - Middle value of ordered measure values

File	Count words
Flo031202a.txt	5031
Flo031201.txt	10346
Flo031204.txt	10943
Flo031202b.txt	11876
Flo031203.txt	12175

Numerical Statistics

- Median
  - Decrease relevance of outliers:

File	Count words
Flo031202a.txt	5031
Flo031201.txt	10346
Flo031204.txt	10943
Flo031202b.txt	11876
Flo031203.txt	12175

Numerical Statistics

- Median
  - with even number of elements:

File	Count words
Flo031202a.txt	5031
Flo031201.txt	10346
Flo031204.txt	10943
Flo031202b.txt	11876

  - Arithmetic mean of the two middle values:

$$\frac{10346 + 10943}{2} = 10644.5$$

Numerical Statistics

- Mean: 10074.2
- Median: 10943
- Mean is reduced on the basis of the outlier:
  - Flo031202a.txt 5031
- Median may be a better indicator of central tendency if outliers/extreme values are present.

Numerical Statistics

- Mode
  - The measure value that occurs most often:

File	Count words
Flo031202a.txt	5031
Flo031201.txt	10943
Flo031204.txt	10943
Flo031202b.txt	6329
Flo031203.txt	12175

  - Mode = 10943

Numerical Statistics

- Approximation of
  - Mode
    - $mean - 3\ (mean - median)$
  - Median
    - $(2\ mean + mode)\ /\ 3$
  - Mean
    - $(3\ median - mode)\ /\ 2$

Numerical Statistics

- Notation
  - Mean (x bar): $\bar{x}$
  - Mean of a population: μ
  - Sum of values: $\sum$

Numerical Statistics

- Notation example:
  - Arithmetic mean:

$$\bar{x} = \frac{\sum x}{n} = \frac{x_1 + x_2 + \ldots + x_n}{n}$$

Numerical Statistics

- Arithmetic mean for grouped data:

Files	Count words
35%	0-4999
30%	5000-9999
25%	10000-14999
10%	15000-19999

  – With 100 sample documents what is the arithmetic mean?

Numerical Statistics

- Arithmetic mean for grouped data:

$$\bar{x} = \frac{\sum fx}{n}$$

  - $f =$ frequency
  - x = midpoint

Numerical Statistics

- Arithmetic mean for grouped data:

Files	Midpoint	fx	Count words
35	2500	87500	0-4999
30	7500	225000	5000-9999
25	12500	312500	10000-14999
10	17500	175000	15000-19999

$$\bar{x} = \frac{\sum fx}{n} = \frac{87500 + 225000 + 312500 + 175000}{100} = \frac{800000}{100} = 8000$$

Numerical Statistics

- Median for grouped data:

$$median = L + \frac{w}{f_{med}}\left(.5n - \sum f_b\right)$$

  - L = lower class limit that contains the interval
  - n = total number of measurements
  - w = class width
  - f_{med} = frequency of the class containing the median
  - $\bullet f_b$ = sum of the frequencies for all classes before the median class

Numerical Statistics

- Median for grouped data:

Files	Count words
35	0-4999
30	5000-9999
25	10000-14999
10	15000-19999

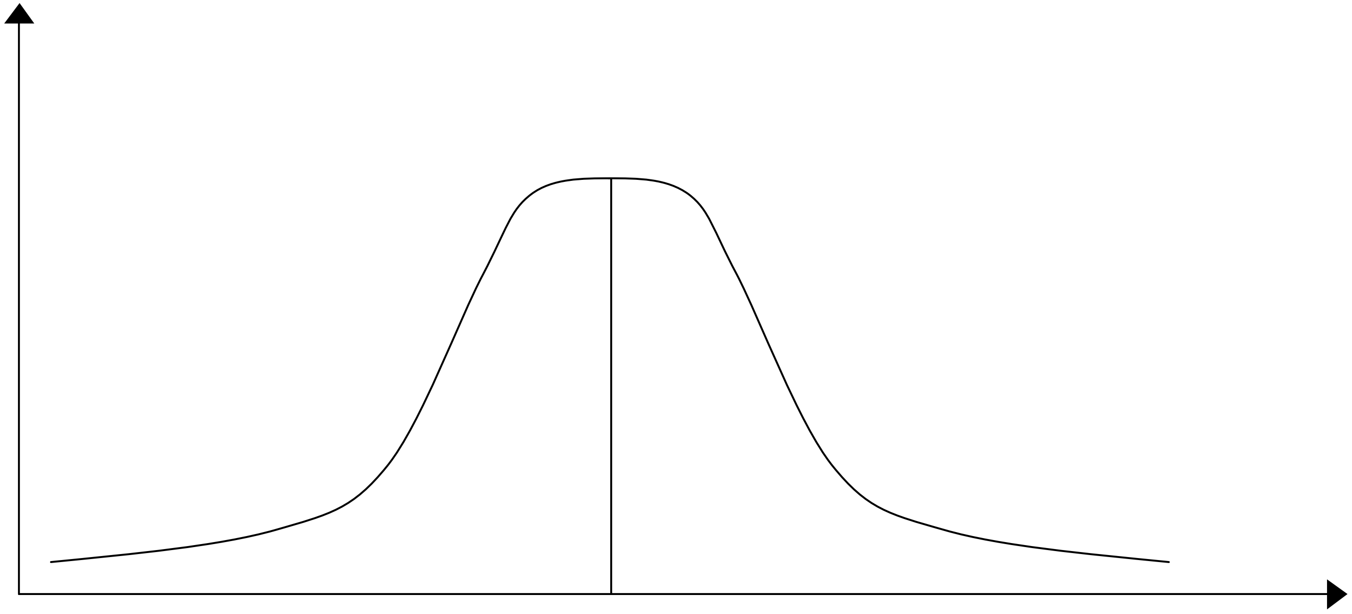
$$median = 5000 + \frac{4999}{30}(50 - 35) = 7499.5$$

Numerical Statistics

- Distribution
  - Symmetric distribution
  - Skewed curves
    - negatively skewed curves
    - positively skewed curves

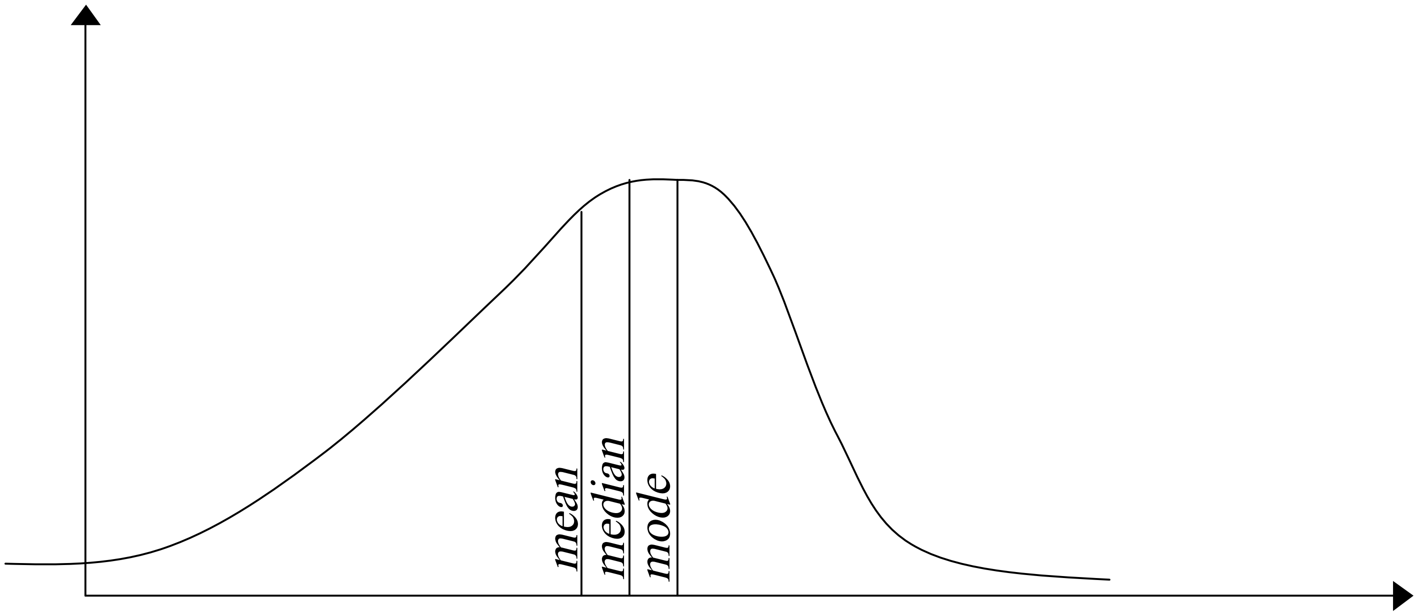
Numerical Statistics

- Symmetric distribution
  - Mean, median and mode are equal.



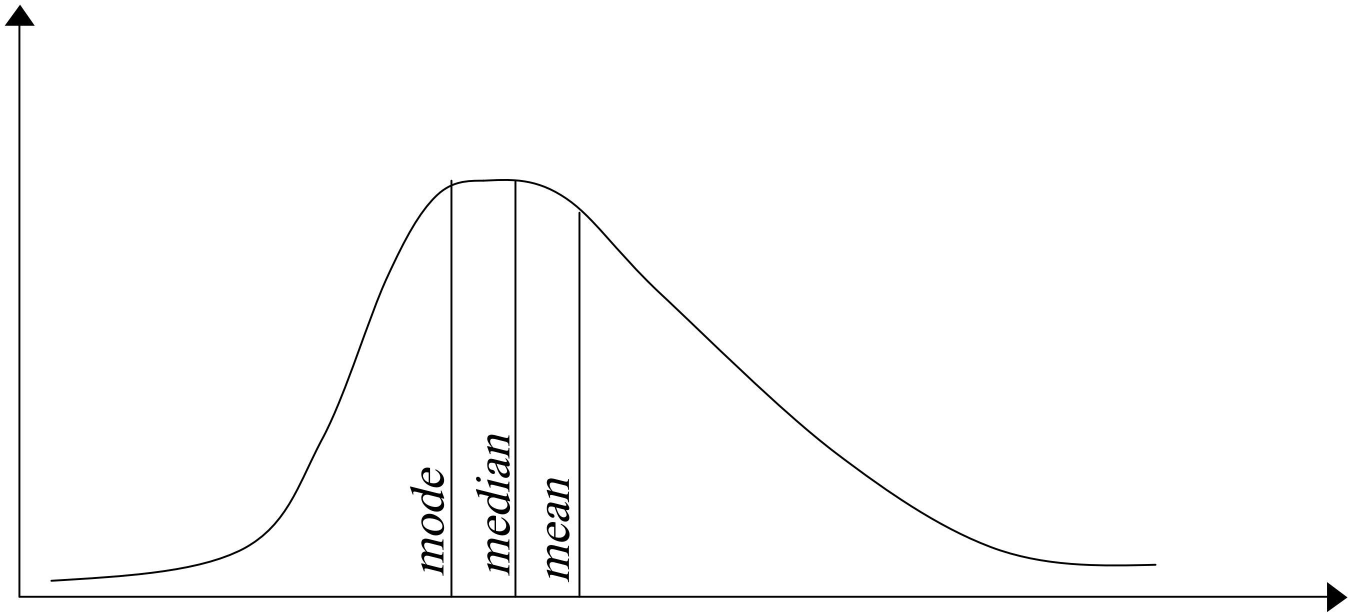
Numerical Statistics

- Skewed curves
  - Negatively skewed distribution: mean < median < mode



Numerical Statistics

- Skewed curves
  - Positively skewed distribution: mode < median < mean



Numerical Statistics

- Variability

Experiment 1	Experiment 2
195	10
210	0
199	400
200	20
205	380
190	200
200	390
201	200

Numerical Statistics

- Variability
  - For both experiments:
    - mean: 200
    - mode: 200
    - median: 200
  - Experiment 2 has greater variation.
- Measure of variation:
  - Range
  - Deviation
  - Variance

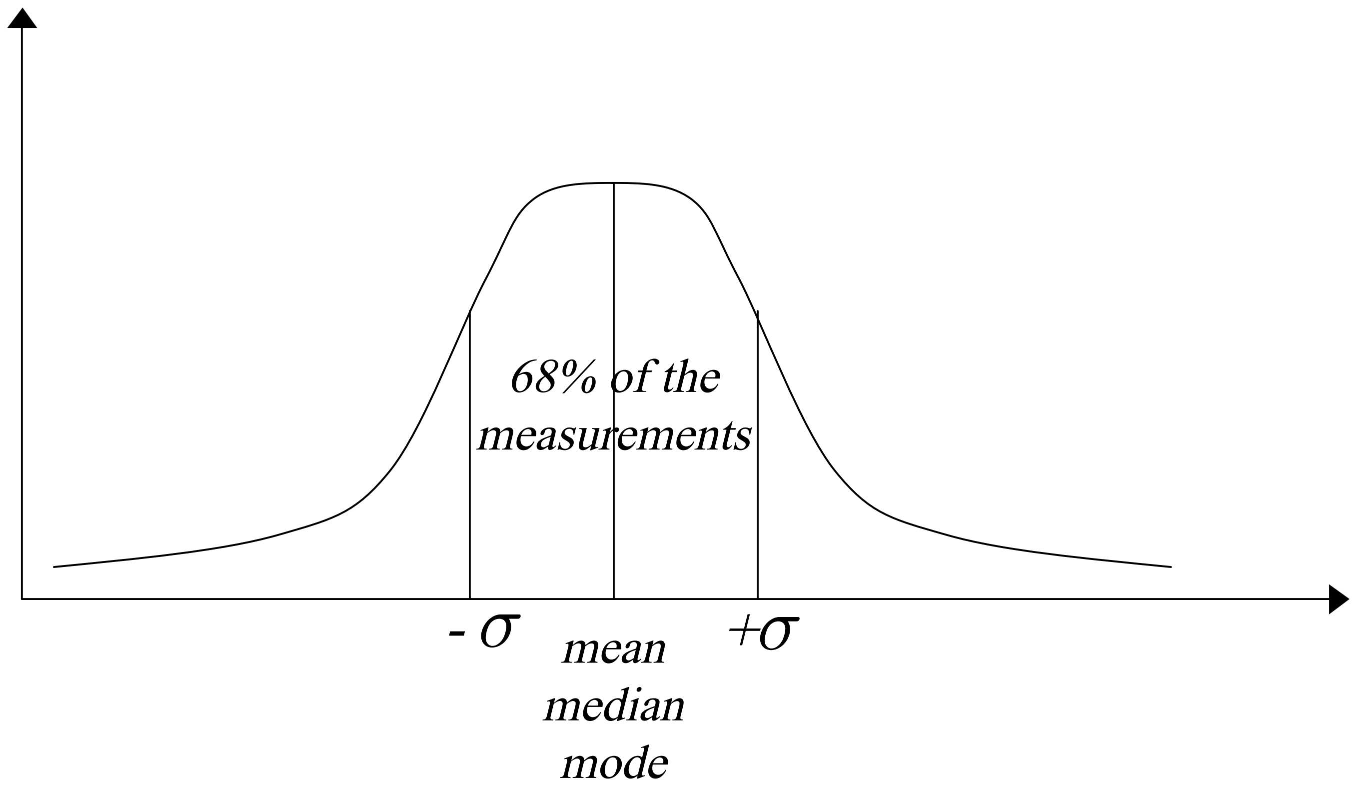
Numerical Statistics

- Range
  - Difference between largest and smallest value:
    - Experiment 1: 210 – 190 = 20
    - Experiment 2: 400 – 0 = 400
- Deviation
  - Distance of the measurements away from the mean:
    - Experiment 1: less
    - Experiment 2: more

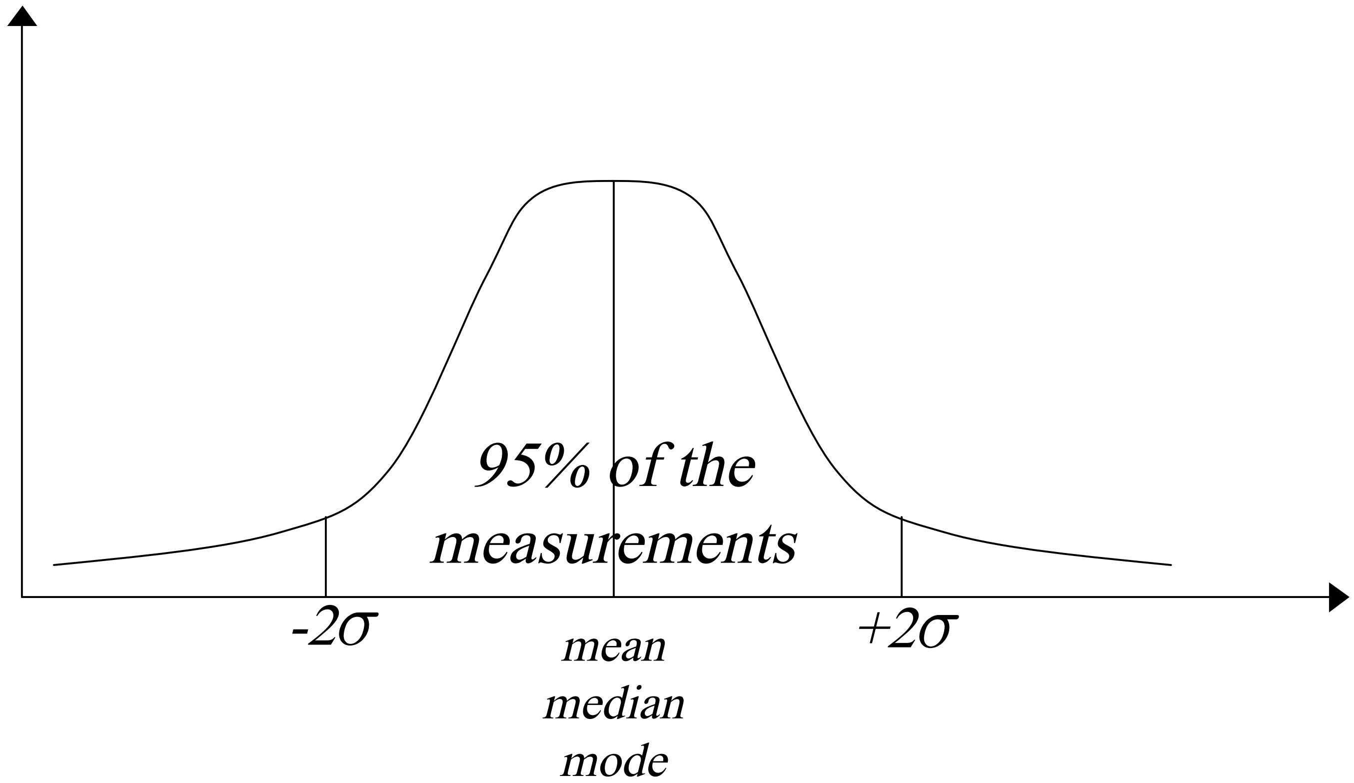
Numerical Statistics

- Notation
  - s^2 = variance of a sample
  - σ^2 = variance of a population
  - s = standard deviation of a sample
  - σ = standard devition of a population

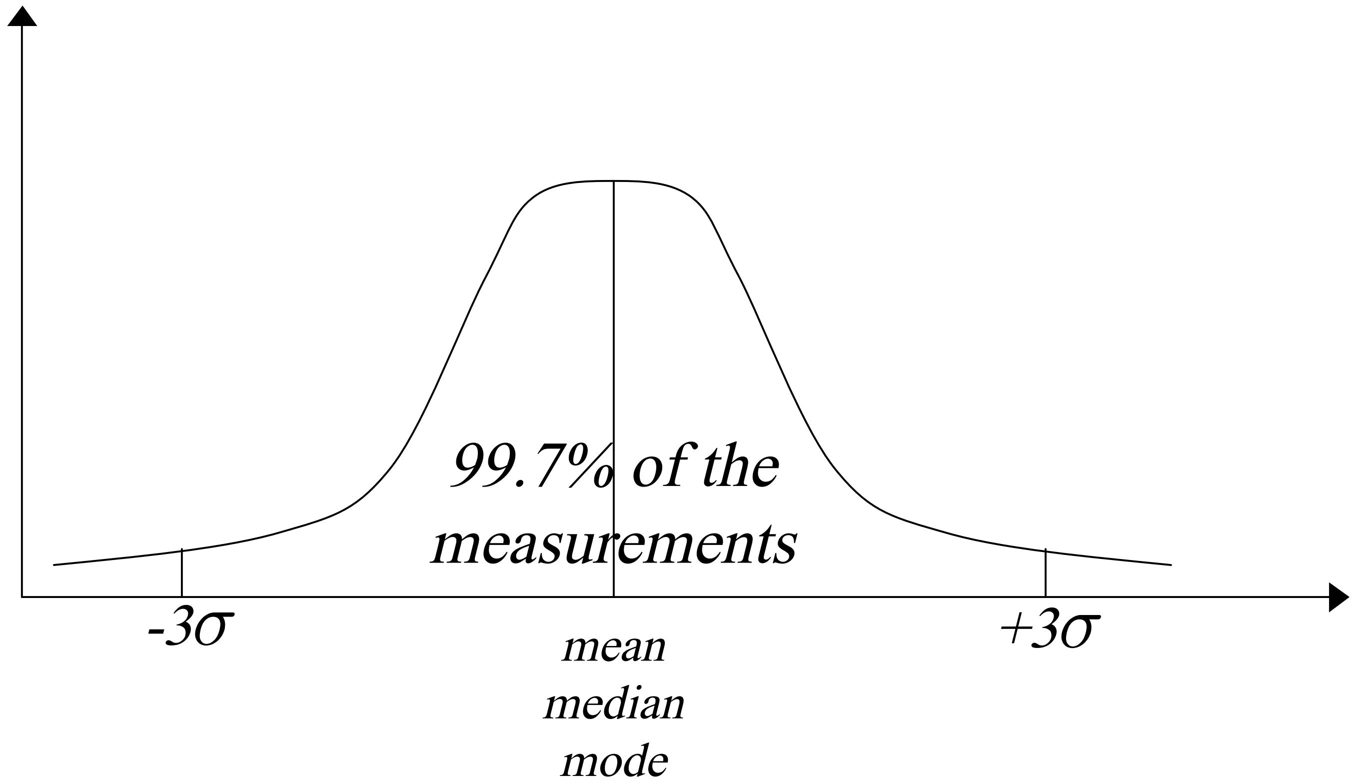
Numerical Statistics



Numerical Statistics



Numerical Statistics



Language Identification

- General *n*-gram class
- Files: `ngram.py`
- Task:
  - Develop a simple *n*-gram script on the basis of the *n*-gram class for uni-gram, bi-gram, and tri-gram models
  - Read in the data from the Brown corpus: a. *n*-gram model of the tokens and b. *n*-gram model of the types

Collocations

- Words in context
  - distribution
  - fixed expressions
  - collocations
    - statistical properties
    - function words

Tests for collocations

- Statistics
- Significance tests

Significance

- Notations:
  - Type I error rate of .05
  - Alpha level of .05 or $\alpha = .05$
  - Finding is significant at the .05 level
  - Confidence level is 95%
  - 95% certainty that a result is not due to chance
  - A 1 in 20 chance of obtaining the result

Testing

- Statistics as testing of scientific hypotheses
- Strategies:
  - Formulating a Research Hypothesis or Alternative Hypothesis (Ha)
    - Statement of the expectation to be tested

Testing

- Strategies:
  - Derivation of a statement that is the opposite of the research hypothesis: Null Hypothesis (H0)
    - Testing the null hypothesis

Testing

- Statistics as testing of scientific hypotheses
- Strategies:
  - If the null hypothesis can be rejected, this is evidence in favor of the research hypothesis.

Testing

- Strategies:
  - Usually:
    - No prove for research hypothesis, just support for it.

Testing

- Research Hypothesis:
  - At IU linguistics students perform differently in statistics than computer science students.
    - H_a: $\mu_1 \neq \mu_2$
    - H_a: $\mu_1 - \mu_2 \neq 0$

Testing

- Null Hypothesis:
  - At IU linguistics students perform the same in statistics as computer science students.
    - $H_0: \mu_1 = \mu_2$
    - $H_0: \mu_1 - \mu_2 = 0$

Testing

- More specific: Research Hypothesis:
  - At IU linguistics students perform better in statistics than computer science students.
    - H_a: $\mu_1 > \mu_2$
    - H_a: $\mu_1 - \mu_2 > 0$

Testing

- More specific: Null Hypothesis
  - At IU linguistics students perform worse in statistics, or equal to computer science students.
    - H_0: $\mu_1 \leq \mu_2$
    - H_0: $\mu_1 - \mu_2 \leq 0$

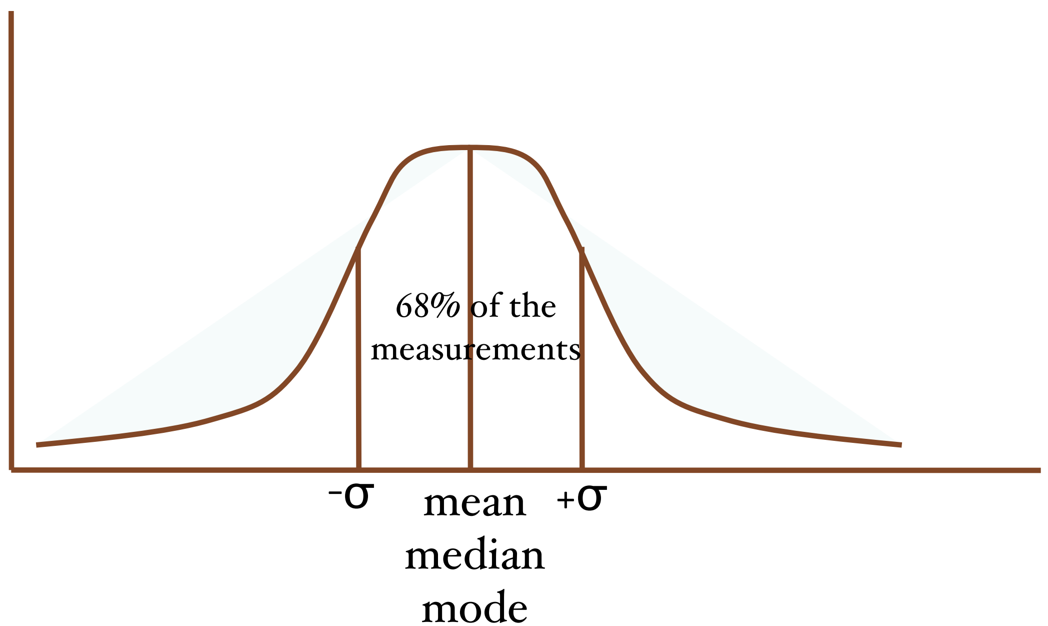
Testing

- Given the distribution of a known area
  - e.g. normal distribution
- estimate the probability of obtaining a certain value as a result of chance.
- If the probability is low, the likelihood for a mere coincidence is low, i.e. a certain theory is correct.

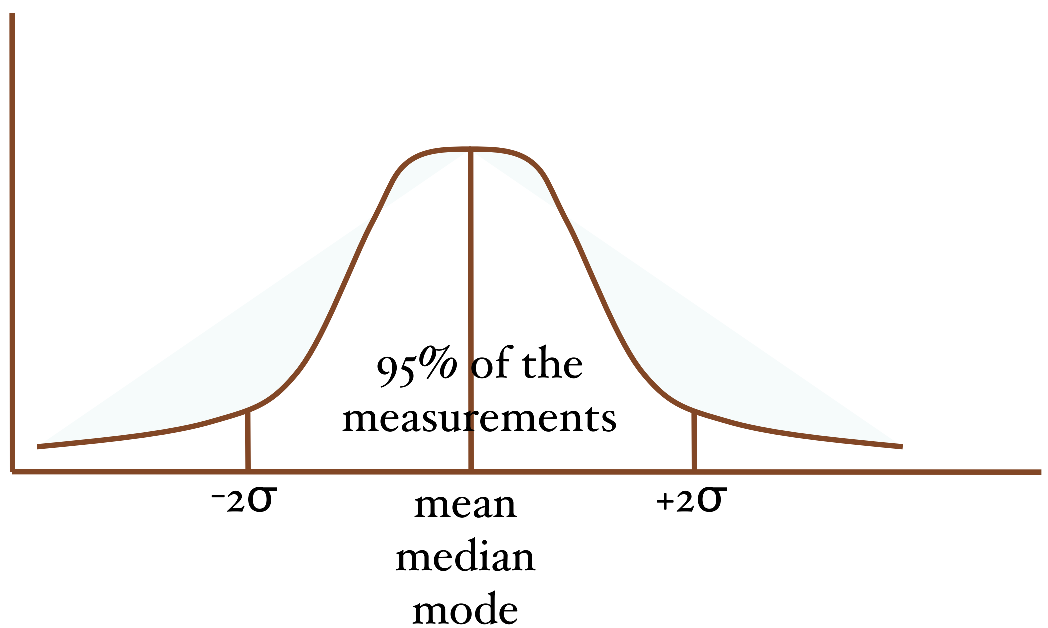
Testing

- Two possible outcomes of test:
  - Rejection of null hypothesis
  - Acceptance of null hypothesis

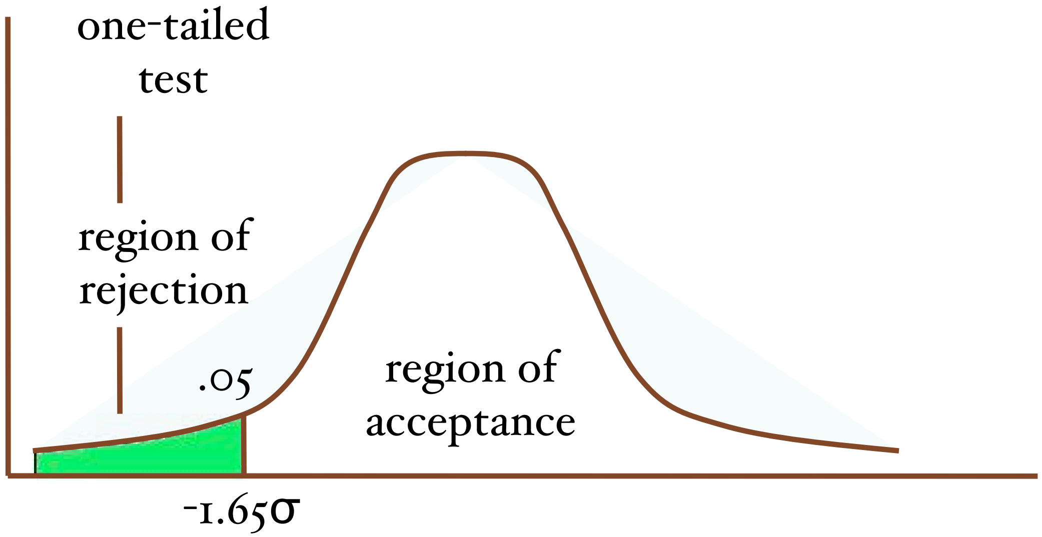
Numerical Statistics



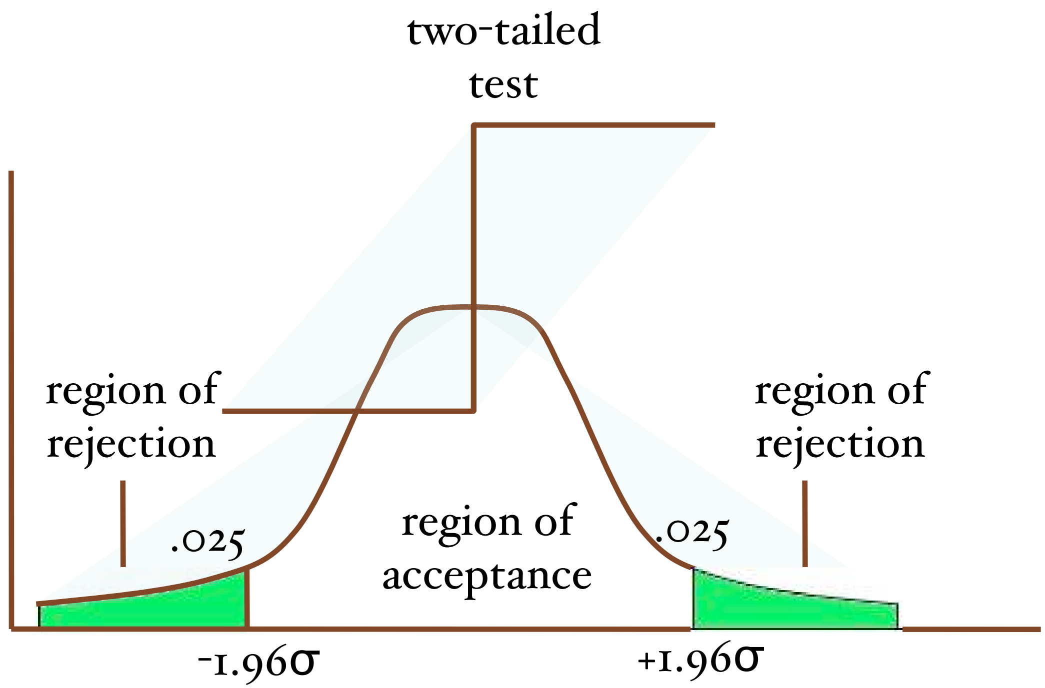
Numerical Statistics



Numerical Statistics



Numerical Statistics



Significance Table

	P	**0.99**	**0.95**	0.10	0.05	0.01	0.005	0.001
d.f.	1	0.00016	0.0039	2.71	3.84	6.63	7.88	10.83
	2	0.020	0.10	4.60	5.99	9.21	10.60	13.82
	3	0.115	0.35	6.25	7.81	11.34	12.84	16.27
	4	0.297	0.71	7.78	9.49	13.28	14.86	18.47
	100	70.06	77.93	118.5	124.3	135.8	140.2	149.4

Testing

- Probability as significance level
- Example: Collocations
  - Null Hypothesis: independence of two words
  - $P(w_1 w_2) = P(w_1)\ P(w_2)$

chi-square (X2) test

- Preferred activities over a population sample of 125 people:

	bowling	dancing	computer	total
male	30	29	16	75
female	12	33	5	50
total	42	62	21	125

chi-square (χ2) test

- Is the choice of activities related to the gender?
  - If the two variables are independent, we can use these probabilities to predict how many people should be in each cell.
  - If the actual number is different from the expectation for independence, the two variables must be related.

chi-square (χ2) test

- Research Hypothesis:
  - The variables are dependent.
- Null Hypothesis:
  - The variables are independent.

chi-square (χ2) test

- Overall probability of a person in the sample being:
  - male: 75/125 = .6
  - female: 50/125 = .4

chi-square (χ2) test

- Overall probability of each preference:
  - bowling: 42/125 = .336
  - dancing: 62/125 = .496
  - computer games: 21/125 = .168

chi-square (χ2) test

- Independent events: multiplication rule
  - The probability of two events occurring is the product of their two probabilities.

chi-square (χ2) test

- Probability of a person in the sample being male and preferring bowling:
  - P(male & bowling): .6 x .336 = .202
  - Expectation: .202 x 125 = 25.2

chi-square (X2) test

- Multiplication of row total with column total and division by total number in sample:
- (75 x 42) / 125 = 25.2

	bowling	dancing	computer	total
male	30 (25.2)	29 (37.2)	16 (12.6)	75
female	12 (16.8)	33 (24.8)	5 (8.4)	50
total	42	62	21	125

chi-square (χ^2) test

- Formula: $$\chi^2 = \sum \frac{(observed - expected)^2}{expected}$$

$$\chi^2 = \frac{(30-25.2)^2}{25.2} + \frac{(29-37.2)^2}{37.2} + \frac{(16-12.6)^2}{12.6} + \frac{(12-16.8)^2}{16.8} + \frac{(33-24.8)^2}{24.8} + \frac{(5-8.4)^2}{8.4} = 9.097$$

chi-square (χ^2) test

- The larger χ^2, the more likely the variables are related.
- Square effect of cells with large differences.

chi-square (χ^2) test

- Probability distribution of χ^2:
  - Critical values in table
  - Degree-of-freedom:
    - df = (number-of-rows - 1) x (number-of-columns – 1)
    - Example: (2 – 1) x (3 – 1) = 2
  - Example: 9.097 (< .025; > .01)

chi-square (χ^2) test

- Example: 9.097 (< .025; > .01)
  - Significance (at levels: .05, .01)!
  - Rejection of Null Hypotheses (independence of variables)

chi-square (χ^2) test

- Collocations
  - new, companies

	w1=new	w1¬new	total
w2=companies	8	4667	4675
w2¬companies	15820	14287181	14303001
total	15828	14291848	14307676

Statistics

- Maximum Likelihood Estimate
  - Frequency oriented
  - No inclusion of prior belief: here e. g. assuming a normal distribution, i. e.
  - No inclusion of a prior probability distribution

Statistics

- Example: Coin tossing
  - Observation: $8 \times head$ & $2 \times tail$
  - Prior probability distribution: $5 \times head$ & $5 \times tail$
  - Probability mass: $\frac{1}{2}$
  - How does observation: $X = tail$ change our expectation?

Statistics

- Example: Coin tossing
  - Bayesian answer: update prior belief (= prior probability distribution) in face of evidence and generate posterior probability estimate

Bayesian Statistics

- Example:
  - Data is added incrementally/sequentially
  - Given an a-priori probability distribution
    * Update our beliefs whith every new datum
    * Calculate Maximum A Posteriori (MAP) distribution
  - MAP probability becomes the new prior probability for the next datum

Information Theory

- Surprise effect:
  - Coin tossing and observing the results
  - What is our prior believe or expectation about an outcome?
  - How surprised are we to see a certain outcome?
- Data compression:
  - Knowing about the distributional properties of some data
  - What is the best compression we can get by mapping it to bit-representations?
  - Is there a formal way to calculate the optimal representation for data transmission?

Information Theory

- Entropy:
  - Entropy as uncertainty
    - Tossing a coin = not knowing what the outcome will be.
    - Probability distribution:
      - Fair coin
      - Biased coin, unlimited probability distributions

Information Theory

- Entropy:
  - Entropy as uncertainty
    - Is there a way to calculate the uncertainty and formulate a function on the basis of a probability distribution?
    - Let us design such a function:
      - $H[X]$ is the measure for X, with X a probability distribution
      - H takes X, with $X = \{P(1), P(2), \ldots P(N)\}$ as an argument
      - and returns a real number, the value of uncertainty

Information Theory

- Designing a function for Entropy:

  1. Maximum uncertainty in uniform distribution: every possible outcome is equally likely

  $\rightarrow$ This is the maximum H can return

  2. H is a continuous function over the probabilities

  $\rightarrow$ changing the probabilities slightly leads to slight changes of H

Information Theory

- Grouping Probabilities:
  - $X = \{P(1) = .5, P(2) = .2, P(3) = (.3)\}$:
  - is equivalent to:
    - $X = \{P(1) = .5, P(Y) = .5\}$
    - $Y = \{P(2) = .4, P(3) = .6\}$

3. Uncertainty H cannot depend on the grouping of events for a random variable.

Information Theory

- Entropy: Formal reformulation of (1–3)
  - $H(p)$ is a real valued function of $P(1), P(2), \ldots P(N)$, with N the number of values for the random variable or length of *domain*, then

1. $H(P(1), P(2), \ldots P(N))$ reaches a maximum if the distribution is uniform: $P(i) = 1/N, N = len(i), \forall\ i$.
2. $H(P(1), P(2), \ldots P(N))$ is a continuous function of all $P(i)$'s.

Information Theory

- Entropy: Formal reformulation of (1–3)

  3. Independence of subsets of probability groups: for N probabilities grouped into k subsets, w_k:

$$w_1 = \sum_{i=1}^{n_1} p_i; w_2 = \sum_{i=n_1+1}^{n_2} p_i; \ldots$$

Information Theory

- Entropy: Formal reformulation of (1–3)

  3. Independence of subsets of probability groups: assumption

$$H[p] = H[w] + \sum_{j=1}^{k} w_j H[\{p_i/w_j\}_j]$$

  - $\{p_i/w_j\}$ is: sum extends over p_i's that make up a particular w_j

Information Theory

- Entropy: Summary
  - Given the three requirements it follows that:

$$H[X] = k \sum_{x \in X} Pr(x) log Pr(x)$$

  - with k and arbitrary constant $[8, 40, 44]$. For $k = -1$ and log_2 the units are bit.

Information Theory

- Average Shannon Entropy: measured in bits

$$H[X] = -1 \sum_{x \in X} Pr(x) lg Pr(x)$$

$$H[X] = \sum_{x \in X} Pr(x) lg \frac{1}{Pr(x)}$$

Information Theory

- Average Shannon Entropy of one outcome: measured in bits

$$h[x] = Pr(x) lg \frac{1}{Pr(x)}$$

Joint Entropy

- For a pair of random variables: $X, Y \sim p(x, y)$

$$H(X,Y) = -\sum_{x \in X} \sum_{y \in Y} p(x,y) lg p(x,y)$$

- $X = \{A = .4, B = .6\}$
- $Y = \{C = .2, D = .8\}$

Joint Entropy

- $X \wedge Y = \{AC = .4 \times .2, AD = .4 \times .8, BC = .6 \times .2, BD = .6 \times .8\}$
- $X \wedge Y = \{AC = .08, AD = .32, BC = .12, BD = .48\}$
- $Z = \{AC = .08, AD = .32, BC = .12, BD = .48\}$

Mutual Information

- Reduction of uncertainty of one random variable due to knowing about another.
- Amount of information one random variable contains about another.
- Symmetric, Non-negative
- $MI = 0$, if two random variables are independent
- MI is high, if two random variables are dependent, depending on their entropy.

Mutual Information

- MI over random variables!

→ Pointwise Mutual Information

  – Pointwise MI over selected values of random variables!

$$I(X;Y) = P(XY) lg \frac{P(XY)}{P(X)P(Y)}$$

- How many bits can we spare by storing two elements, rather than each single element alone?

Relative Entropy – KL Divergence

- Average number of bits that are wasted by encoding events from random variable X with a code based on random variable Y. How close are two pmf's?

$$D(y||x) = p(y) lg \frac{p(y)}{p(y|x)}$$

$$D(y||x) = p(y) lg \frac{p(y)}{\frac{p(xy)}{p(x)}} = p(y) lg \frac{p(y)p(x)}{p(xy)}$$

- How many bits more would we use by storing $< xy >$, rather than each single element alone?

Vector Space

- Representing elements in a vector space:
  - $x = [2.0, 4.9, 12.4, \ldots]$
  - Matrix:
    - row = elements
    - column = features
  - Representation in an n-dimensional space
  - Linear Algebra for analysis of vector similarity
  - Vector similarity for clustering, grouping, association

Vector Space

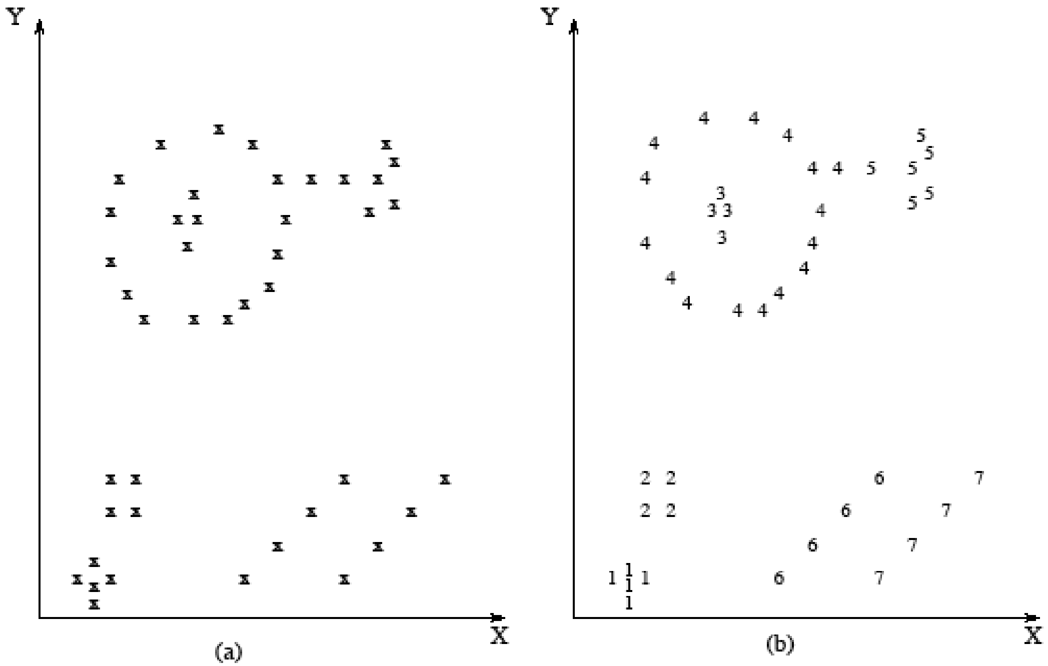
$$\mathscr{X} = \begin{bmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{1,2} & \cdots & \mathbf{x}_{1,d} \\ \mathbf{x}_{2,1} & \mathbf{x}_{2,2} & \cdots & \mathbf{x}_{2,d} \\ \vdots & & & \\ \mathbf{x}_{k,1} & \mathbf{x}_{k,2} & \cdots & \mathbf{x}_{k,d} \end{bmatrix}$$

Vector Space

- What is mapped on the vector space?
  - Number of individual words in a document: each row one document, each column a word
  - Number of individual words in the context of one word (left and right)
  - Features of words, documents, etc.

Vector Space

- Each vector is a point in `n`-dimensional space: Example for `n`



Optimization Clustering

- Given a clustering criterion
  - How to find a partition into n groups that optimizes the criterion?
- Find all possible partitions and calculate their value of the given criterion.
- Choose the partition with the optimal value.

Optimization Clustering

- Complexity:
  - Number of possible partitions given n objects into g groups (Liu, 1968):

$$N(n,g) = \frac{1}{g!}\sum_{m=1}^{g}(-1)^{g-m}\binom{g}{m}m^n \tag{1}$$

Optimization Clustering

- Complexity example:

$$N(50, 4) = 5.3 \times 10^{28} \tag{2}$$

$$N(100, 5) = 6.6 \times 10^{67} \tag{3}$$

Optimization Clustering

- Complexity solution
  - Programming strategies
    - Dynamic programming
    - Branch and bound algorithms
- Hill-climbing algorithms
  - Iterative search for optimum value of clustering criteria via rearrangement of existing partitions

Optimization Clustering

- K-means generates
  - k number of disjoint clusters (non-hierarchical)
  - globular clusters (spherical, elliptical, convex)
- properties:
  - numerical
  - unsupervised (limited!)
  - iterative

Optimization Clustering

- K-means
  - k clusters
  - At least one element per cluster
  - No overlapping clusters
  - Non-hierarchical
  - Every member of a cluster is closer to its cluster than to any other cluster
  - Procedure

Optimization Clustering

- K-means
  - Initial partitioning of data set into k clusters
  - For each data point: calculate distance to each cluster
  - If one data point is closer to another cluster, relocate it
  - Repeat until no further relocations possible

Optimization Clustering

- K-means advantages
  - For large number of variables it is faster than hierarchical algorithms (for small k's)
  - Tighter clusters than hierarchical clustering, if cluster are globular
- K-means disadvantages
  - Initial set of k clusters can affect the result
  - Does not work well with non-globular clusters

Optimization Clustering

- K-means example

Individual	Variable 1	Variable 2
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

Optimization Clustering

- Initial 2 clusters on the basis of the most distant individuals:

	Individual	Mean Vector
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

Optimization Clustering

- Initial clustering of all remaining individuals:
  - For every other individual:
    - Calculate Euclidean distance to the centroid of every cluster
    - Assign individual to cluster
    - Recalculate centroid for every cluster

Optimization Clustering

- Mean vector or centroid (with coordinates x_1 to x_n) with equal weight coordinates:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{4}$$

Optimization Clustering

- Mean vector or centroid example for $x = \{(3,5),(7,9)\}$, i. e. $n = |x| = 2$:

$$\bar{x} = \frac{\sum_{i=1}^{2} x_i}{2} = \frac{(3,5)+(7,9)}{2} =$$

$$\frac{(3+7,5+9)}{2} = (\frac{10}{2}, \frac{14}{2}) = (5,7)$$

Optimization Clustering

- Initial clustering of all remaining individuals:

	Group 1		Group 2	
	Individual	Mean Vector	Individual	Mean Vector
Step 1	1	(1.0, 1.0)	4	(5.0, 7.0)
Step 2	1, 2	(1.3, 1.5)	4	(5.0, 7.0)
Step 3	1, 2, 3	(1.8, 2.3)	4	(5.0, 7.0)
Step 4	1, 2, 3	(1.8, 2.3)	4, 5	(4.3, 6.0)
Step 5	1, 2, 3	(1.8, 2.3)	4, 5, 6	(4.3, 5.7)
Step 6	1, 2, 3	(1.8, 2.3)	4, 5, 6, 7	(4.1, 5.4)

Optimization Clustering

- Initial partitions and clustering criterion:

	Individual	Mean Vector	Sum of SQR error
Group 1	1, 2, 3	(1.8, 2.3)	6.84
Group 2	4, 5, 6, 7	(4.1, 5.4)	5.38
total			12.22

Optimization Clustering

- Error = for every point distance to centroid
  - Criterion: the smaller the sum of square errors, the better the cluster
- Two dimensional Euclidean distance:

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{5}$$

Optimization Clustering

- Error = for every point distance to centroid
- N-dimensional Euclidean distance, with p_i and q_i the coordinates for p and q in dimension i:

$$\sqrt{\sum_{i=1}^{N}(p_1 - q_1)^2} \tag{6}$$

Optimization Clustering

- Optimization Iteration:
  - Compare each individual's distance to its own mean with distance to the opposite group mean.
  - If distance to the mean in opposite group is smaller, relocate the individual.
  - Calculate the sum of square errors, if smaller than before, this is an improvement.

Optimization Clustering

- Distance to means:

Individual	distance to mean 1	distance to mean 2
1	1.5	5.4
2	0.4	4.3
3	2.1	1.8
4	5.7	1.8
5	3.2	0.7
6	3.8	0.8
7	2.8	1.1

Optimization Clustering

- Subsequent partitions and new clustering criterion:

	Individual	Mean Vector	Sum of SQR error
Group 1	1, 2	(1.3, 1.5)	0.63
Group 2	3, 4, 5, 6, 7	(3.9, 5.1)	7.9
total			8.53

- Decrease of clustering criterion (from 12.22 to 8.53).

Optimization Clustering

- Remember:
  - *k-means* or *k-nearest neighbors* is a fast and efficient algorithm.
  - You have to know how many clusters you are looking for.
  - Specific cluster shapes will not be discovered.

Optimization Clustering

- Expectation Maximization
  - Assume different Gaussian distribution for each cluster
  - Calculate the Expectation of belonging to each Gaussian for each data point
  - Assign each data point to the Gaussian with the highest expectation
  - Recalculate Gaussians given the new data points
  - Repeat until no significant improvement of expectation

$$f(X) = \frac{1}{\sqrt{2\pi\ deviation}} e^{-\frac{(value - mean)^2}{2\ deviation^2}} \tag{1}$$

Keep in mind...

Schlage die Trommel und fürchte dich nicht,
Und küsse die Marketenderin!
Das ist die ganze Wissenschaft,
Das ist der Bücher tiefster Sinn.
(Heinrich Heine, *Doktrin*)

Thanks for your attendance and hope to see you again!

References

[MacKay(2003)] David J. C. MacKay. *Information theory, inference, and learning algorithms*. Cambridge University Press, Cambridge, UK; New York, 2003.