

Search Engines

Information Retrieval in Practice

Evaluation

- Which method is better ?
- Why?

Method 1

1 R

2 NR

3 R

4 NR

5 NR

Method 2

1 NR

2 R

3 R

4 R

5 NR

Evaluation

- Evaluation is key to building *effective* and *efficient* search engines
 - measurement usually carried out in controlled laboratory experiments
 - *online* testing can also be done
- Efficiency measures similar to database systems
 - e.g., indexing time, query throughput, index size
- Our focus is on *effectiveness* metrics
 - Comparing systems
 - Parameter tuning

Evaluation Corpus

- *Test collections* consisting of documents, queries, and relevance judgments, e.g.,
 - CACM: Titles and abstracts from the Communications of the ACM from 1958-1979. Queries and relevance judgments generated by computer scientists.
 - AP: Associated Press newswire documents from 1988-1990 (from TREC disks 1-3). Queries are the title fields from TREC topics 51-150. Topics and relevance judgments generated by government information analysts.
 - GOV2: Web pages crawled from websites in the .gov domain during early 2004. Queries are the title fields from TREC topics 701-850. Topics and relevance judgments generated by government analysts.

Test Collections

Collection	Number of documents	Size	Average number of words/doc.
CACM	3,204	2.2 Mb	64
AP	242,918	0.7 Gb	474
GOV2	25,205,179	426 Gb	1073

Collection	Number of queries	Average number of words/query	Average number of relevant docs/query
CACM	64	13.0	16
AP	100	4.3	220
GOV2	150	3.1	180

ClueWeb

- ClueWeb09
 - <http://lemurproject.org/>
 - 1B web pages, 5T B compressed
 - Queries and relevance judgments from various TREC tracks
 - Lots of spam
- ClueWeb12
 - 733M web pages
 - Higher quality crawl

TREC Topic Example

<top>

<num> Number: 794

<title> pet therapy

<desc> Description:

How are pets or animals used in therapy for humans and what are the benefits?

<narr> Narrative:

Relevant documents must include details of how pet- or animal-assisted therapy is or has been used. Relevant details include information about pet therapy programs, descriptions of the circumstances in which pet therapy is used, the benefits of this type of therapy, the degree of success of this therapy, and any laws or regulations governing it.

</top>

Important decisions based on “track”:

- Number of queries
- Query types

Relevance Judgments

- Obtaining relevance judgments is an expensive, time-consuming process
 - who does it?
 - what are the instructions?
 - what is the level of agreement?
- TREC judgments
 - depend on task being evaluated
 - generally binary
 - agreement good because of “narrative”

Pooling

- Exhaustive judgments for all documents in a collection is not practical
- Pooling technique is used in TREC
 - top *k results* (for TREC, *k varied between 50 and 200*) from the rankings obtained by different search engines (or retrieval algorithms) are merged into a pool
 - duplicates are removed
 - documents are presented in some random order to the relevance judges
- Produces a large number of relevance judgments for each query, although still incomplete
- More queries are generally better than more judgments per query

Query Logs

- Used for both tuning and evaluating search engines
 - also for various techniques such as query suggestion
- Typical contents
 - User identifier or user session identifier
 - Query terms - stored exactly as user entered
 - List of URLs of results, their ranks on the result list, and whether they were clicked on
 - Timestamp(s) - records the time of user events such as query submission, clicks

Query Logs

- Clicks are not relevance judgments
 - although they are correlated
 - biased by a number of factors such as rank on result list
- Can also use clickthrough data to predict *preferences* between pairs of documents
 - appropriate for tasks with multiple levels of relevance, focused on user relevance
 - various “policies” used to generate preferences

Example Click Policy

- *Skip Above and Skip Next*

- click data

- d_1

- d_2

- d_3 (clicked)

- d_4

- generated preferences

- $d_3 > d_2$

- $d_3 > d_1$

- $d_3 > d_4$

Effectiveness Measures

A is set of relevant documents,
 B is set of retrieved documents

	Relevant	Non-Relevant
Retrieved	$A \cap B$	$\overline{A} \cap B$
Not Retrieved	$A \cap \overline{B}$	$\overline{A} \cap \overline{B}$

$$Recall = \frac{|A \cap B|}{|A|}$$

$$Precision = \frac{|A \cap B|}{|B|}$$

Precision and recall

	Relevant	Nonrelevant
Retrieved	true positives (TP)	false positives (FP)
Not retrieved	false negatives (FN)	true negatives (TN)

$$P = TP / (TP + FP)$$

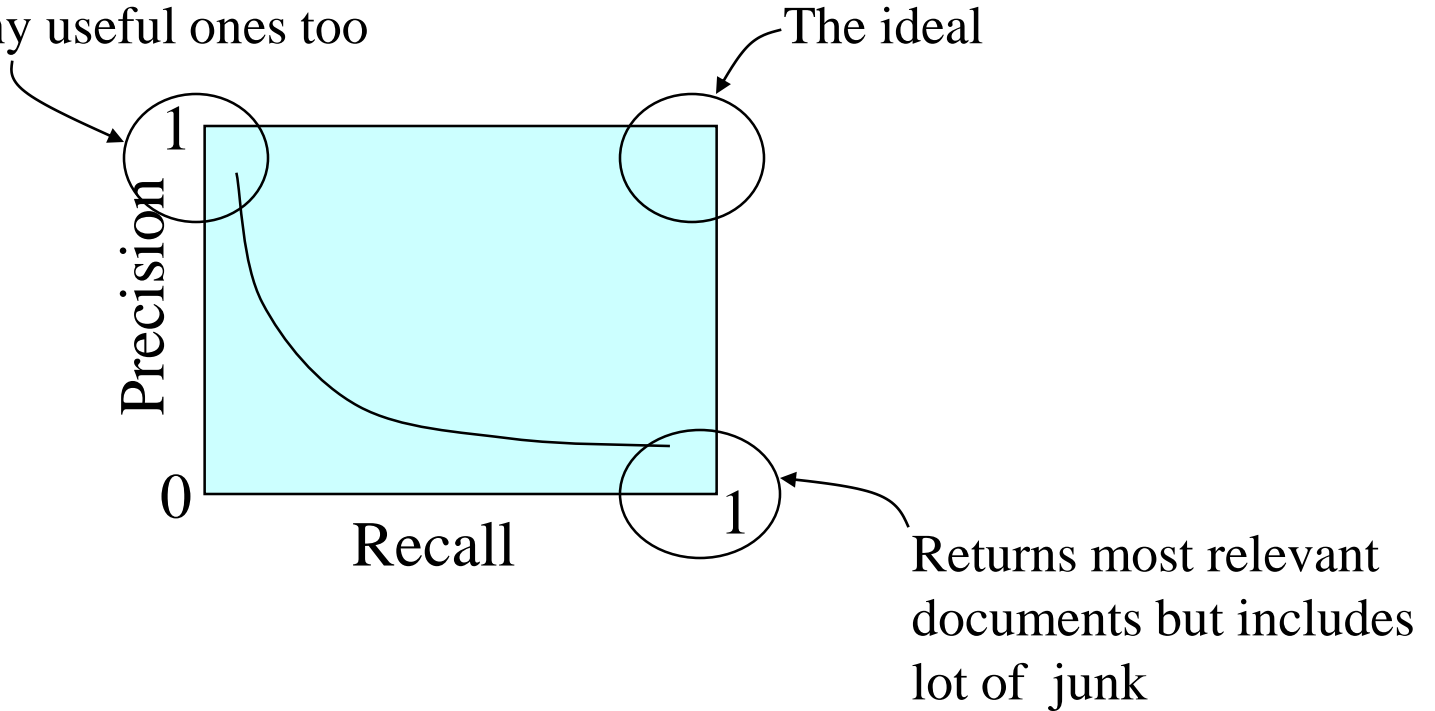
$$R = TP / (TP + FN)$$

Classification Errors

- *False Positive* (Type I error)
 - a non-relevant document is retrieved
- $$Fallout = \frac{|\overline{A} \cap B|}{|\overline{A}|}$$
- *False Negative* (Type II error)
 - a relevant document is not retrieved
 - 1- *Recall*
- *Precision* is used when probability that a positive result is correct is important
 - *Information retrieval, medical tests*

Trade-offs

Returns relevant documents but
misses many useful ones too



Precision/Recall

- You can get high recall (but low precision) by retrieving all docs for all queries!
- Recall is a non-decreasing function of the number of docs retrieved
- In a good system, precision decreases as either the number of docs retrieved or recall increases
 - This is not a theorem, but a result with strong empirical confirmation

Example

rank	docid	binary relevance
1	43	1
2	531	1
3	183	1
4	195	1
5	2	0
6	109	1
7	176	0
8	1612	0
9	16	1
10	13	1

- Assume there is 21 relevant documents.

- What is Precision?
- What is Recall?

F Measure

- *Harmonic mean* of recall and precision

$$F = \frac{1}{\frac{1}{2}(\frac{1}{R} + \frac{1}{P})} = \frac{2RP}{(R+P)}$$

- harmonic mean emphasizes the importance of small values, whereas the arithmetic mean is affected more by outliers that are unusually large
- More general form

$$F_{\beta} = (\beta^2 + 1)RP/(R + \beta^2 P)$$

- β is a parameter that determines relative importance of recall and precision
- People usually use balanced F_1 measure

F: Example

	relevant	not relevant	
retrieved	20	40	60
not retrieved	60	1,000,000	1,000,060
	80	1,000,040	1,000,120

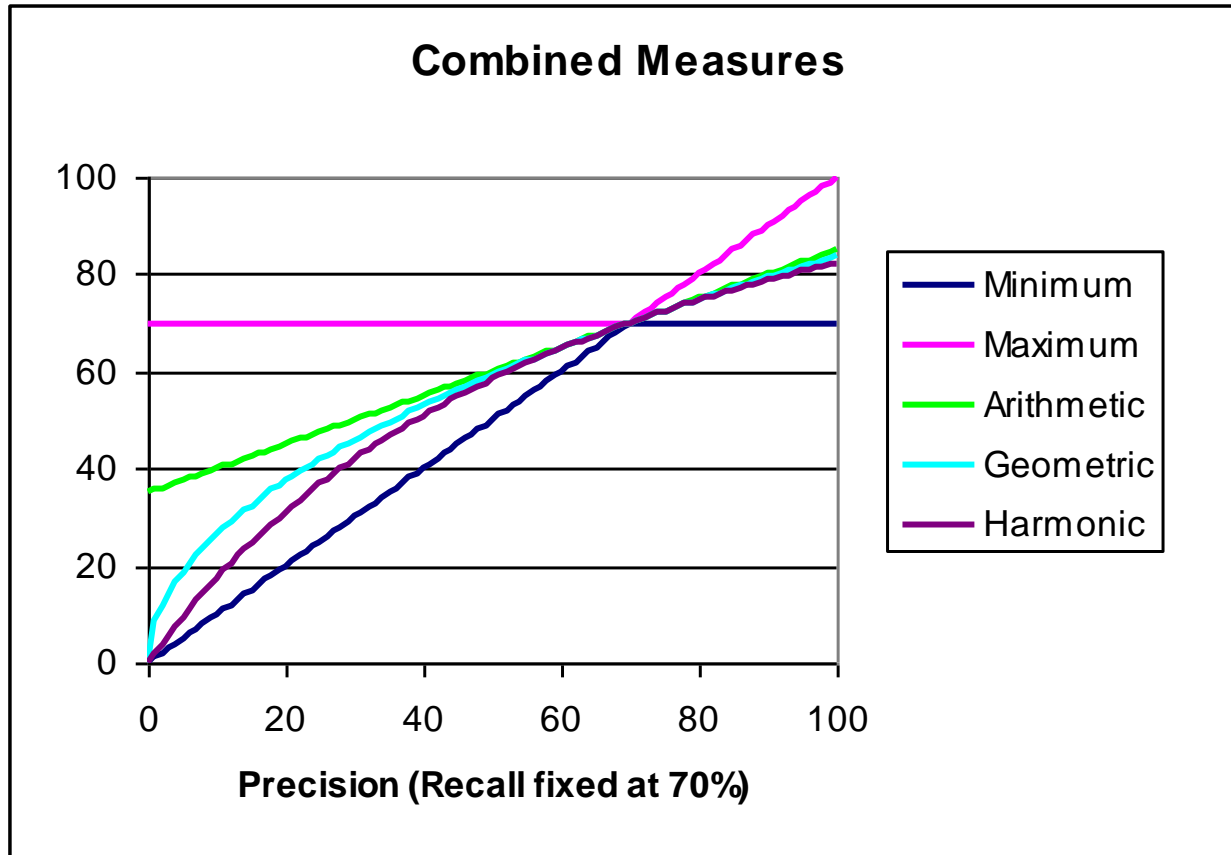
- $P = 20/(20 + 40) = 1/3$

- $R = 20/(20 + 60) = 1/4$

-

$$F_1 = 2 \frac{1}{\frac{1}{3} + \frac{1}{4}} = 2/7$$

F_1 and other averages













- We can view the harmonic mean as a kind of soft minimum








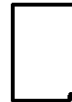


Ranking Effectiveness

 = the relevant documents

Ranking #1

										
Recall	0.17	0.17	0.33	0.5	0.67	0.83	0.83	0.83	0.83	1.0
Precision	1.0	0.5	0.67	0.75	0.8	0.83	0.71	0.63	0.56	0.6

Ranking #2

										
Recall	0.0	0.17	0.17	0.17	0.33	0.5	0.67	0.67	0.83	1.0
Precision	0.0	0.5	0.33	0.25	0.4	0.5	0.57	0.5	0.56	0.6

Summarizing a Ranking

- Calculating recall and precision at fixed rank positions
 - $P@k$, $R@k$
- Calculating precision at standard recall levels, from 0.0 to 1.0
 - requires *interpolation*
- Averaging the precision values from the rank positions where a relevant document was retrieved
 - AP

Example

rank	docid	binary relevance
1	43	1
2	531	1
3	183	1
4	195	1
5	2	0
6	109	1
7	176	0
8	1612	0
9	16	1
10	13	1











- Assume there is only 7 relevant documents.

- What is $P@3$? $P@5$? $P@10$?
- What is $R@3$? $R@5$?
- What ranking maximizes $P@10$?
- What is AP?











Average Precision

 = the relevant documents

Ranking #1

										
Recall	0.17	0.17	0.33	0.5	0.67	0.83	0.83	0.83	0.83	1.0
Precision	1.0	0.5	0.67	0.75	0.8	0.83	0.71	0.63	0.56	0.6


Ranking #2

										
Recall	0.0	0.17	0.17	0.17	0.33	0.5	0.67	0.67	0.83	1.0
Precision	0.0	0.5	0.33	0.25	0.4	0.5	0.57	0.5	0.56	0.6











Ranking #1: $(1.0 + 0.67 + 0.75 + 0.8 + 0.83 + 0.6)/6 = 0.78$

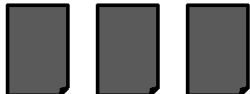
Ranking #2: $(0.5 + 0.4 + 0.5 + 0.57 + 0.56 + 0.6)/6 = 0.52$

Averaging Across Queries











 = relevant documents for query 1

Ranking #1

										
Recall	0.2	0.2	0.4	0.4	0.4	0.6	0.6	0.6	0.8	1.0
Precision	1.0	0.5	0.67	0.5	0.4	0.5	0.43	0.38	0.44	0.5

 = relevant documents for query 2


Ranking #2

										
Recall	0.0	0.33	0.33	0.33	0.67	0.67	1.0	1.0	1.0	1.0
Precision	0.0	0.5	0.33	0.25	0.4	0.33	0.43	0.38	0.33	0.3











Averaging


- *Mean Average Precision* (MAP)
 - summarize rankings from multiple queries by averaging average precision
 - most commonly used measure in research papers
 - assumes user is interested in finding many relevant documents for each query
 - requires many relevance judgments in text collection
- Recall-precision graphs are also useful summaries

MAP











 = relevant documents for query 1

Ranking #1

										
Recall	0.2	0.2	0.4	0.4	0.4	0.6	0.6	0.6	0.8	1.0
Precision	1.0	0.5	0.67	0.5	0.4	0.5	0.43	0.38	0.44	0.5

 = relevant documents for query 2

Ranking #2

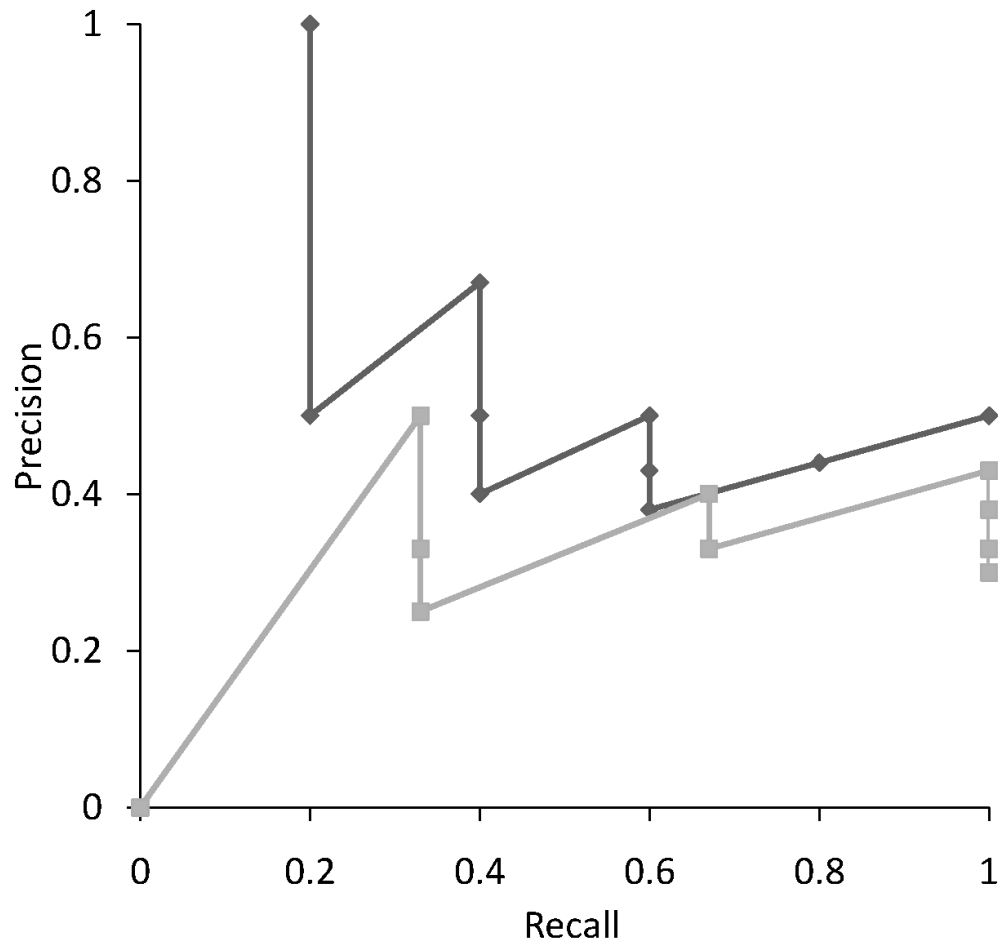
										
Recall	0.0	0.33	0.33	0.33	0.67	0.67	1.0	1.0	1.0	1.0
Precision	0.0	0.5	0.33	0.25	0.4	0.33	0.43	0.38	0.33	0.3

$$\text{average precision query 1} = (1.0 + 0.67 + 0.5 + 0.44 + 0.5)/5 = 0.62$$

$$\text{average precision query 2} = (0.5 + 0.4 + 0.43)/3 = 0.44$$

$$\text{mean average precision} = (0.62 + 0.44)/2 = 0.53$$

Recall-Precision Graph



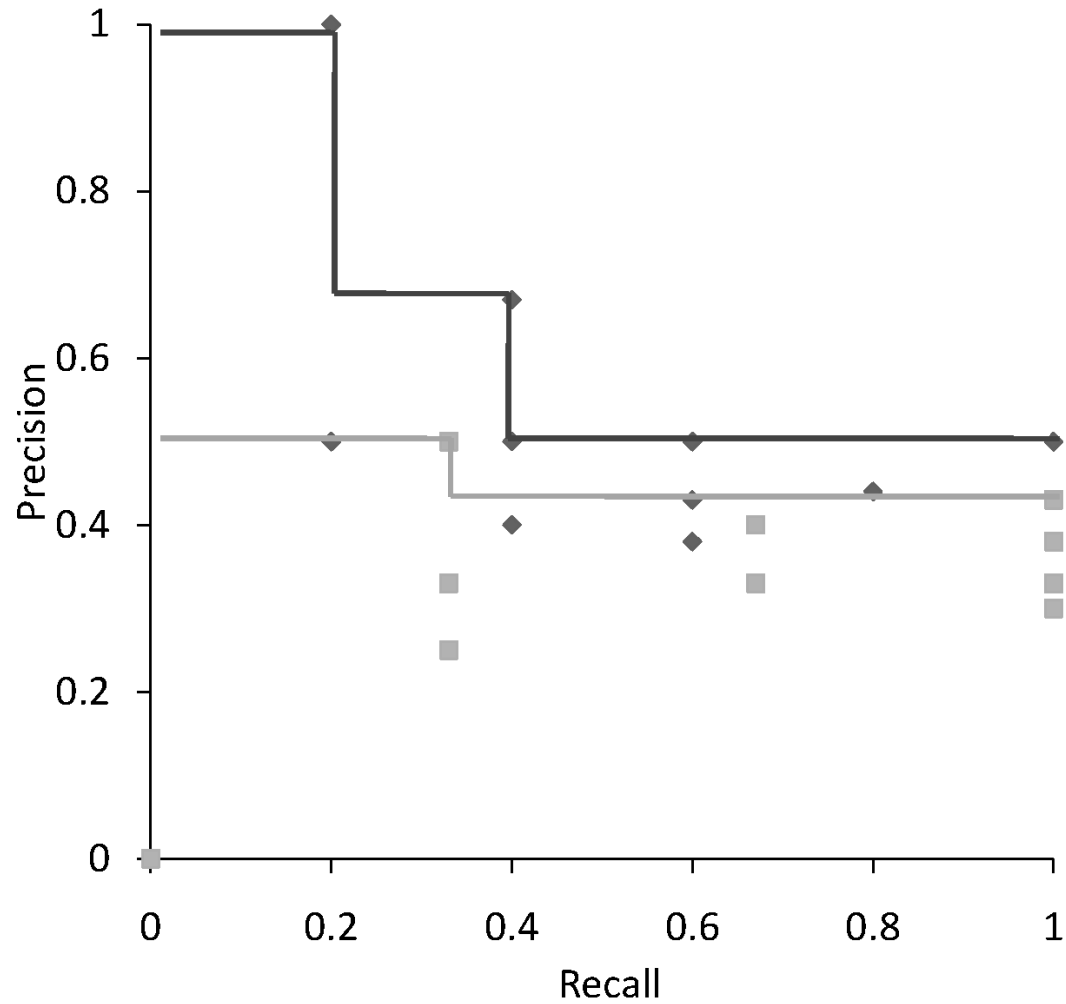
Interpolation

- To average graphs, calculate precision at standard recall levels:

$$P(R) = \max\{P' : R' \geq R \wedge (R', P') \in S\}$$

- where S is the set of observed (R, P) points
- Defines precision at any recall level as the *maximum* precision observed in any recall-precision point at a higher recall level
 - produces a step function
 - defines precision at recall 0.0

Interpolation

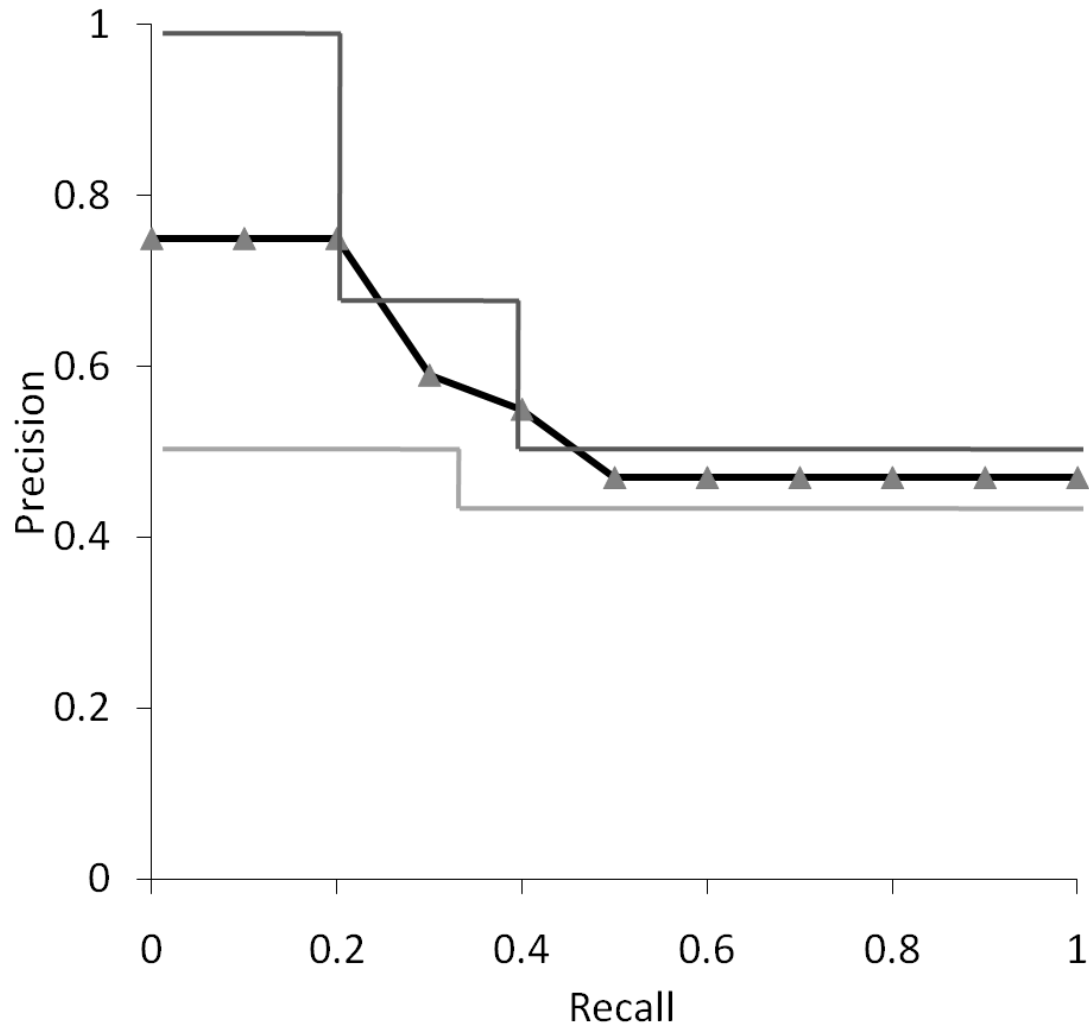


Average Precision at Standard Recall Levels

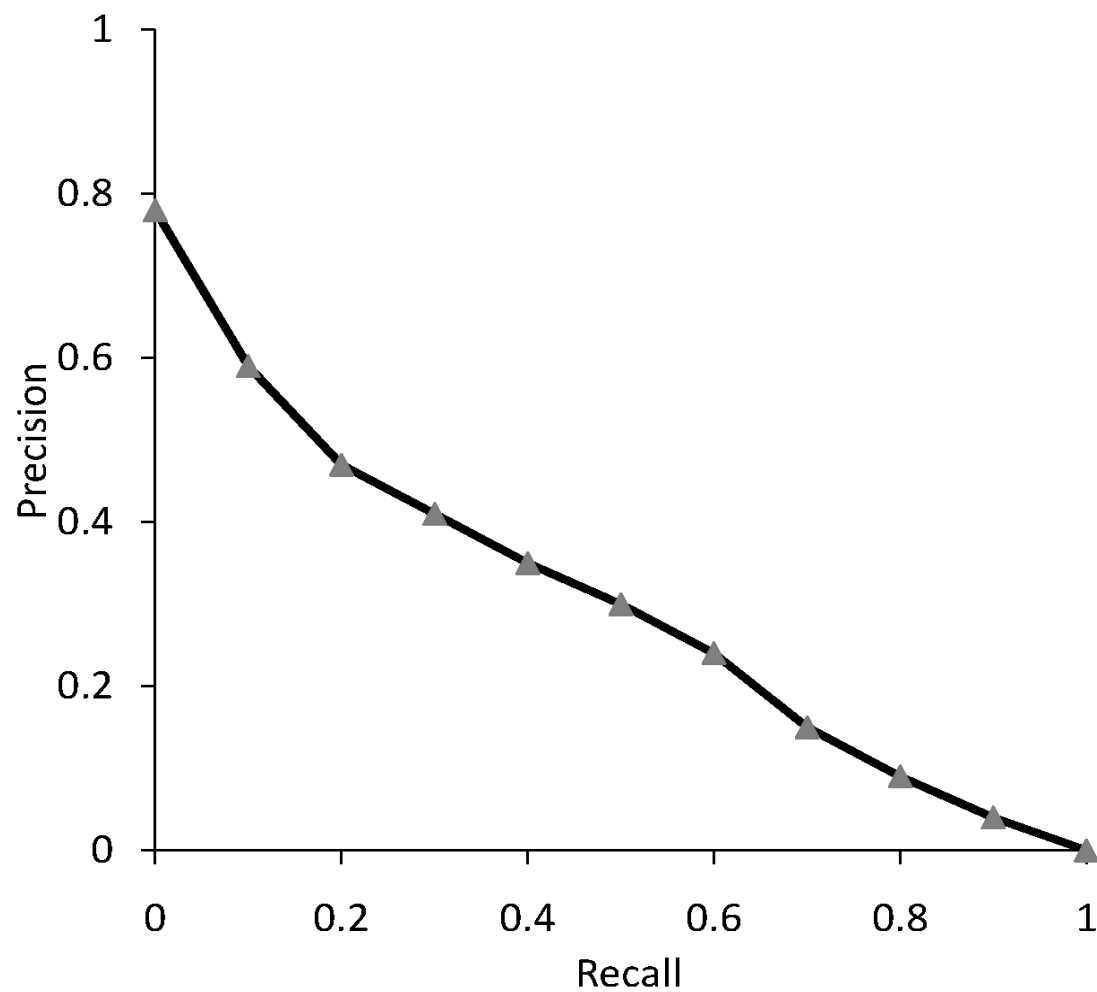
Recall	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Ranking 1	1.0	1.0	1.0	0.67	0.67	0.5	0.5	0.5	0.5	0.5	0.5
Ranking 2	0.5	0.5	0.5	0.5	0.43	0.43	0.43	0.43	0.43	0.43	0.43
Average	0.75	0.75	0.75	0.59	0.55	0.47	0.47	0.47	0.47	0.47	0.47

- Recall-precision graph plotted by simply joining the average precision points at the standard recall levels

Average Recall-Precision Graph



Graph for 50 Queries



Focusing on Top Documents

- Users tend to look at only the top part of the ranked result list to find relevant documents
- Some search tasks have only one relevant document
 - e.g., navigational search, question answering
- Recall not appropriate
 - instead need to measure how well the search engine does at retrieving relevant documents at very high ranks

Focusing on Top Documents

- Precision at Rank R
 - R is the number of relevant documents
 - easy to compute, average, understand
 - not sensitive to rank positions less than R
- Reciprocal Rank
 - reciprocal of the rank at which the first relevant document is retrieved
 - *Mean Reciprocal Rank (MRR)* is the average of the reciprocal ranks over a set of queries
 - very sensitive to rank position

Example

rank	docid	binary relevance
1	43	1
2	531	1
3	183	1
4	195	1
5	2	0
6	109	1
7	176	0
8	1612	0
9	16	1
10	13	1

- Assume there is only 7 relevant documents.

- What is P@R?
- What is RR?

Discounted Cumulative Gain

- Popular measure for evaluating web search and related tasks
- Two assumptions:
 - Highly relevant documents are more useful than marginally relevant document
 - the lower the ranked position of a relevant document, the less useful it is for the user, since it is less likely to be examined

Discounted Cumulative Gain

- Uses *graded relevance* as a measure of the usefulness, or *gain*, from examining a document
- Gain is accumulated starting at the top of the ranking and may be reduced, or *discounted*, at lower ranks
- Typical discount is $1/\log(\text{rank})$
 - With base 2, the discount at rank 4 is $1/2$, and at rank 8 it is $1/3$

Discounted Cumulative Gain

- *DCG* is the total gain accumulated at a particular rank p :

$$DCG_p = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2 i}$$

- Alternative formulation:

$$DCG_p = \sum_{i=1}^p \frac{2^{rel_i} - 1}{\log(1+i)}$$

– used by some web search companies

DCG Example

- 10 ranked documents judged on 0-3 relevance scale:
3, 2, 3, 0, 0, 1, 2, 2, 3, 0
- discounted gain:
 $3, 2/1, 3/1.59, 0, 0, 1/2.59, 2/2.81, 2/3, 3/3.17, 0$
 $= 3, 2, 1.89, 0, 0, 0.39, 0.71, 0.67, 0.95, 0$
- DCG:
3, 5, 6.89, 6.89, 6.89, 7.28, 7.99, 8.66, 9.61, 9.61

Normalized DCG

- DCG numbers are averaged across a set of queries at specific rank values
 - e.g., DCG at rank 5 is 6.89 and at rank 10 is 9.61
- DCG values are often *normalized* by comparing the DCG at each rank with the DCG value for the *perfect ranking*
 - makes averaging easier for queries with different numbers of relevant documents

NDCG Example

- Perfect ranking:
3, 3, 3, 2, 2, 2, 1, 0, 0, 0
- ideal DCG values:
3, 6, 7.89, 8.89, 9.75, 10.52, 10.88, 10.88, 10.88, 10
- NDCG values (divide actual by ideal):
1, 0.83, 0.87, 0.76, 0.71, 0.69, 0.73, 0.8, 0.88, 0.88
 - $\text{NDCG} \leq 1$ at any rank position

Example

rank	docid	graded relevance	binary relevance
1	43	1	1
2	531	4	1
3	183	2	1
4	195	4	1
5	2	0	0
6	109	1	1
7	176	0	0
8	1612	0	0
9	16	4	1
10	13	1	1

$$DCG_p = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2 i}$$

- What is DCG at 5 using binary relevance?
- What is DCG at 5 using graded relevance?
- What is NDCG@5 using binary relevance?
- What is NDCG@5 using graded relevance?

Variance

- For a test collection, it is usual that a system does poorly on some information needs (e.g., MAP = 0.1) and excellently on others (e.g., MAP = 0.7)
- Indeed, it is usually the case that the variance in performance of the same system across queries is much greater than the variance of different systems on the same query.
- That is, there are easy information needs and hard ones!

Can we avoid human judgments?

- No
- Makes experimental work hard
 - Especially on a large scale
- In some specific settings, we can use approximations or implicit judgments
- But once we have test collections, we can reuse them (as long as we don't over-train too badly)

Validity of relevance assessments

- Relevance assessments are only usable if they are consistent.
- If they are not consistent, then there is no “truth” and experiments are not repeatable.
- How can we measure this consistency or agreement among judges?
- → Kappa measure

Kappa measure

- Kappa is measure of how much judges agree or disagree.
- Designed for categorical judgments
- Corrects for chance agreement
- $P(A)$ = proportion of time judges agree
- $P(E)$ = what agreement would we get by chance

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

- κ =? for (i) chance agreement (ii) total agreement

Kappa measure (2)

- Values of k in the interval $[2/3, 1.0]$ are seen as acceptable.
- With smaller values: need to redesign relevance assessment methodology used etc.

Calculating the kappa statistic

		Judge 2 Relevance		
		Yes	No	Total
Judge 1 Relevance	Yes	300	20	320
	No	10	70	80
	Total	310	90	400

Observed proportion of
the times the judges agreed

$$P(A) = (300 + 70)/400 = 370/400 = 0.925$$

Pooled marginals

$$P(\text{nonrelevant}) = (80 + 90)/(400 + 400) = 170/800 = 0.2125$$

$$P(\text{relevant}) = (320 + 310)/(400 + 400) = 630/800 = 0.7878$$

Probability that the two judges agreed by chance $P(E) =$

$$P(\text{nonrelevant})^2 + P(\text{relevant})^2 = 0.2125^2 + 0.7878^2 = 0.665$$

Kappa statistic $\kappa = (P(A) - P(E))/(1 - P(E)) =$

$$(0.925 - 0.665)/(1 - 0.665) = 0.776 \text{ (still in acceptable range)}$$

Evaluation of large search engines

- Search engines have test collections of queries and hand-ranked results
- Recall is difficult to measure on the web
- Search engines often use precision at top k , e.g., $k = 10$
- . . . or measures that reward you more for getting rank 1 right than for getting rank 10 right.
 - NDCG (Normalized Cumulative Discounted Gain)
- Search engines also use non-relevance-based measures.
 - Clickthrough on first result
 - Not very reliable if you look at a single clickthrough ... but pretty reliable in the aggregate.
 - Studies of user behavior in the lab
 - A/B testing

A/B testing

- Purpose: Test a single innovation
- Prerequisite: You have a large search engine up and running.
- Have most users use old system
- Divert a small proportion of traffic (e.g., 1%) to the new system that includes the innovation
- Evaluate with an “automatic” measure like clickthrough on first result
- Now we can directly see if the innovation does improve user happiness.
- Probably the evaluation methodology that large search engines trust most

Significance Tests

- Given the results from a number of queries, how can we conclude that ranking algorithm A is better than algorithm B?
- A significance test enables us to reject the *null hypothesis* (no difference) in favor of the *alternative hypothesis* (B is better than A)
 - the *power* of a test is the probability that the test will reject the null hypothesis correctly
 - increasing the number of queries in the experiment also increases power of test

Reasoning for Significance Tests

- Suppose that the claim is true
- Find your statistic and look at the sampling distribution
- Find the probability of getting a result more extreme
- Is the evidence convincing?
 - An outcome that is very unlikely if a claim is true is good evidence that the claim is not true

Null Hypothesis (H_0)

A statistical tests begins by supposing that the effect we want is not present.

- Then we try to find evidence against this claim
- The claim that we are trying to find evidence against is called the *null hypothesis*
- This statement being tested:
 - No effect
 - No difference
- We want to assess the strength of the evidence against the null hypothesis

Alternative Hypothesis (H_a)

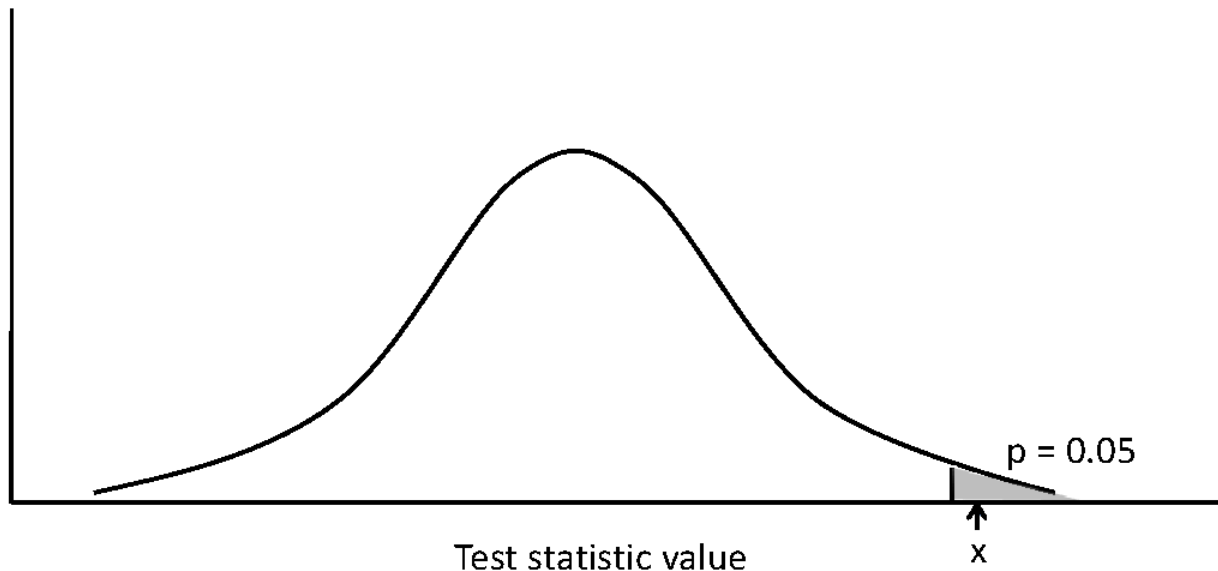
- The statement we hope or suspect is true is called the *alternative hypothesis*
- What we are trying to prove or the effect we are hoping to see
 - In IR: the difference between two systems

Significance Tests

1. Compute the effectiveness measure for every query for both rankings.
2. Compute a *test statistic* based on a comparison of the effectiveness measures for each query. The test statistic depends on the significance test, and is simply a quantity calculated from the sample data that is used to decide whether or not the null hypothesis should be rejected.
3. The test statistic is used to compute a *P-value*, which is the probability that a test statistic value at least that extreme could be observed if the null hypothesis were true. Small P-values suggest that the null hypothesis may be false.
4. The null hypothesis (no difference) is rejected in favor of the alternate hypothesis (i.e., *B* is more effective than *A*) if the P-value is $\leq \alpha$, the *significance level*. Values for α are small, typically .05 and .1, to reduce the chance of a Type I error.

One-Sided Test

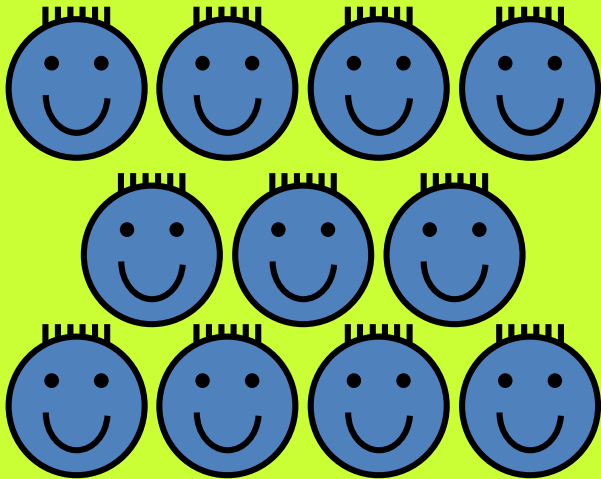
- Distribution for the possible values of a test statistic assuming the null hypothesis



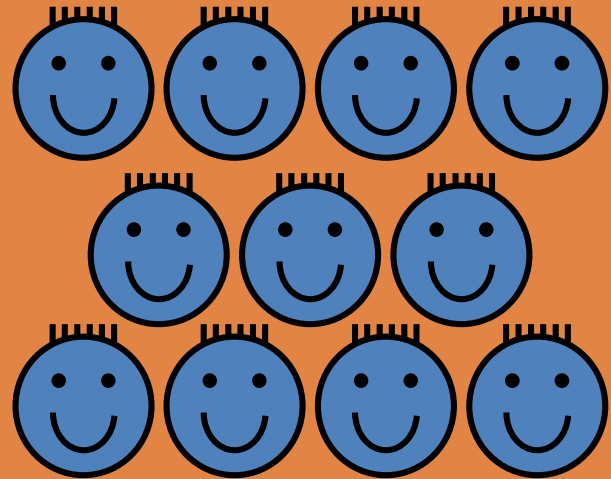
- shaded area is *region of rejection*

Imagine we chose two children at random from two class rooms...

D8

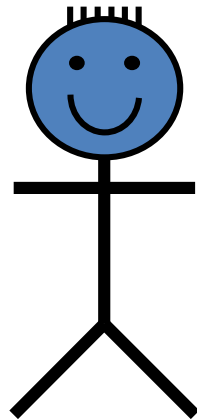
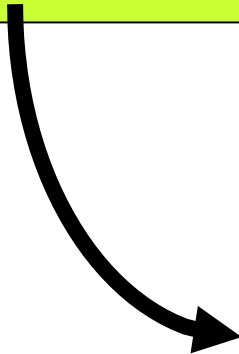
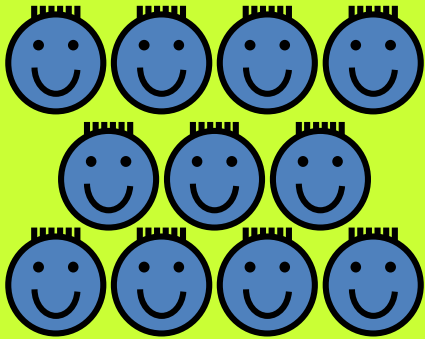


C1



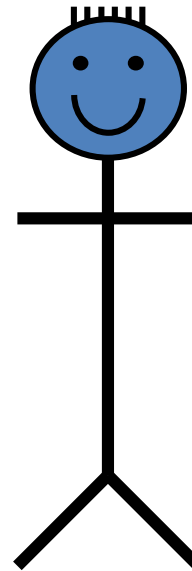
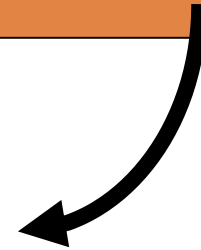
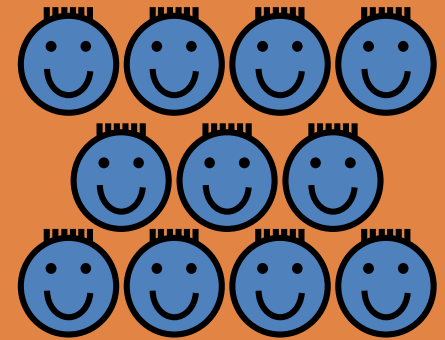
... and compare their height ...

D8



... we find that
one pupil is taller
than the other

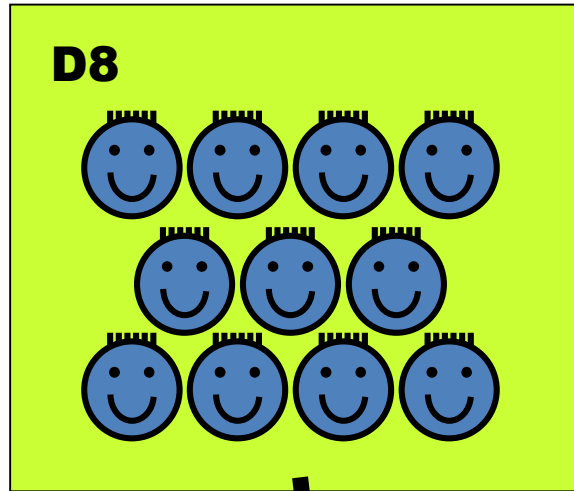
C1



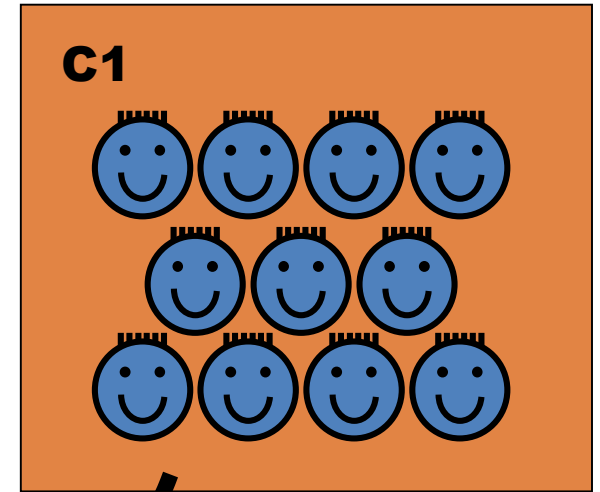
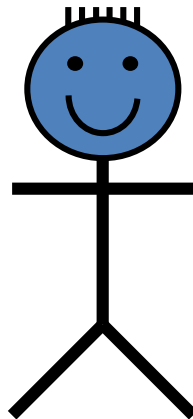
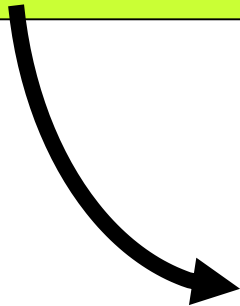
WHY?

REASON 1:

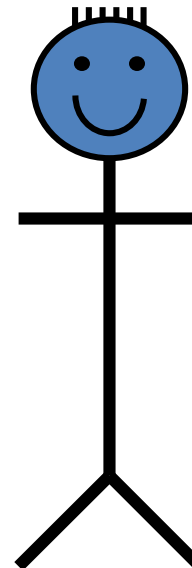
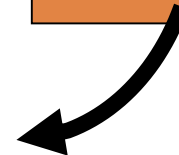
There is a significant difference between the two groups, so pupils in C1 are taller than pupils in D8



YEAR 7

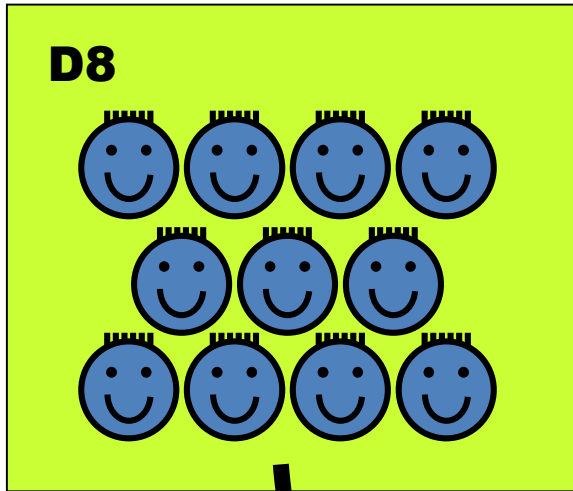


YEAR 11

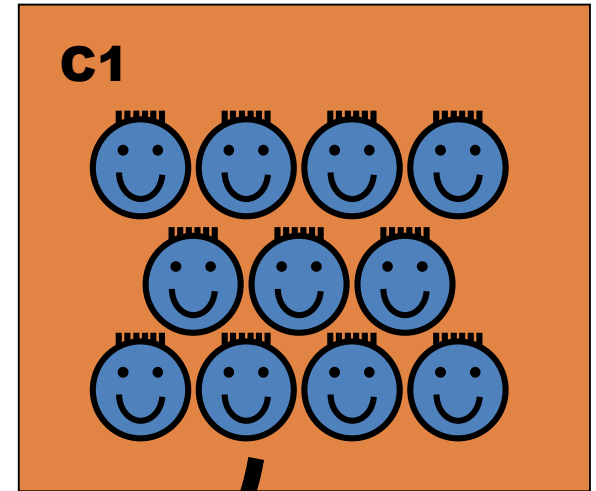
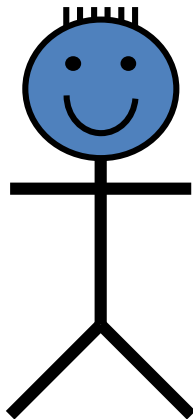


REASON 2:

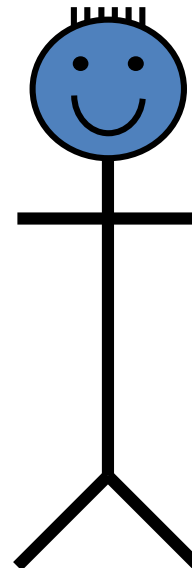
By chance, we picked a short pupil from D8 and
a tall one from C1



TITCH
(Year 9)



HAGRID
(Year 9)

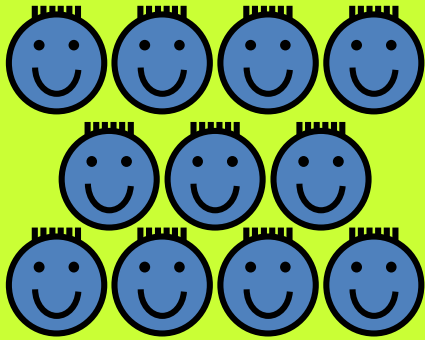


How do we decide which reason is
most likely?

MEASURE MORE STUDENTS!!!

If there is a significant difference between the two groups...

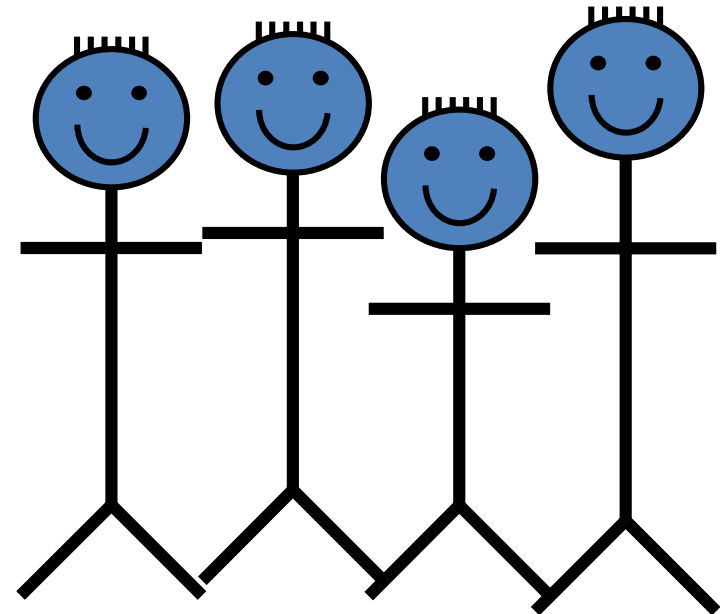
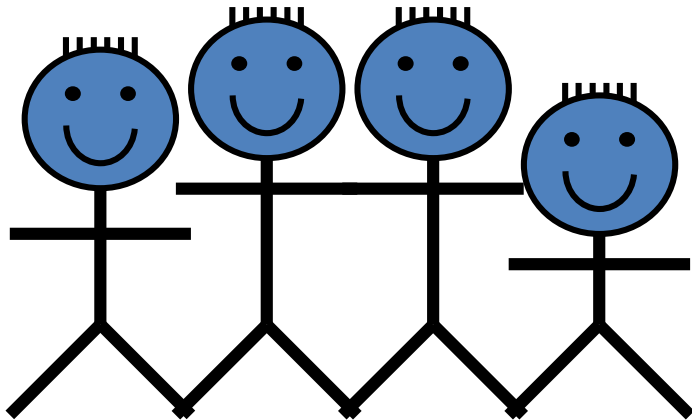
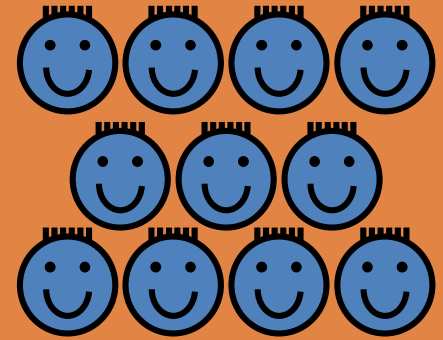
D8



... the average or mean
height of the two
groups should be
very...

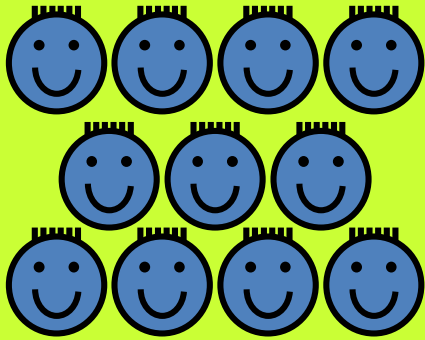
... DIFFERENT

C1



If there is no significant difference between the two groups...

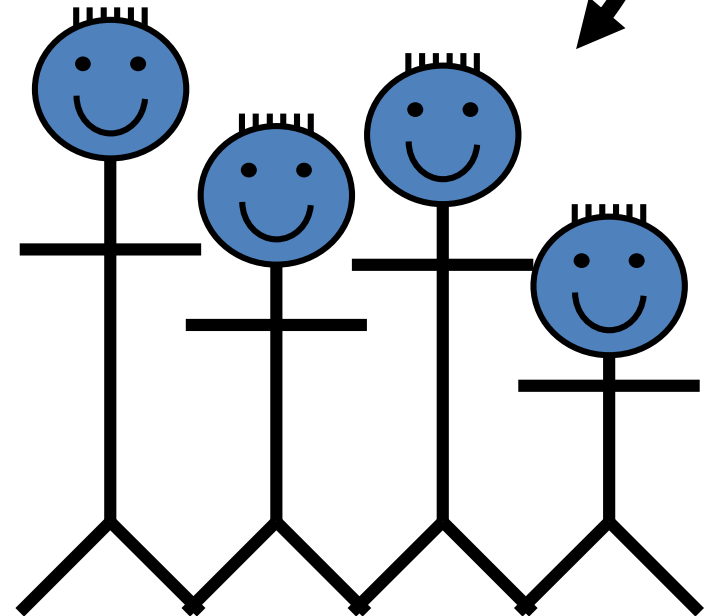
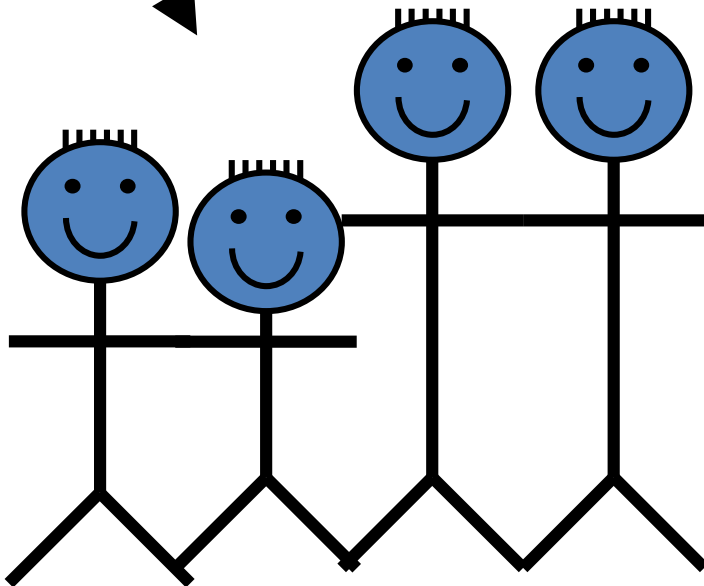
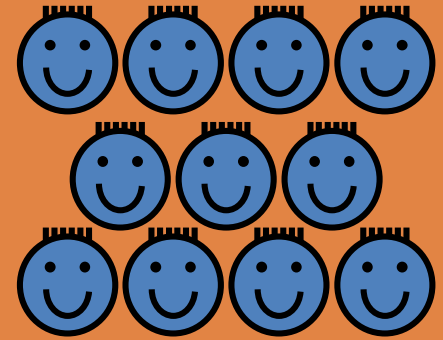
D8



... the average or mean
height of the two
groups should be
very...

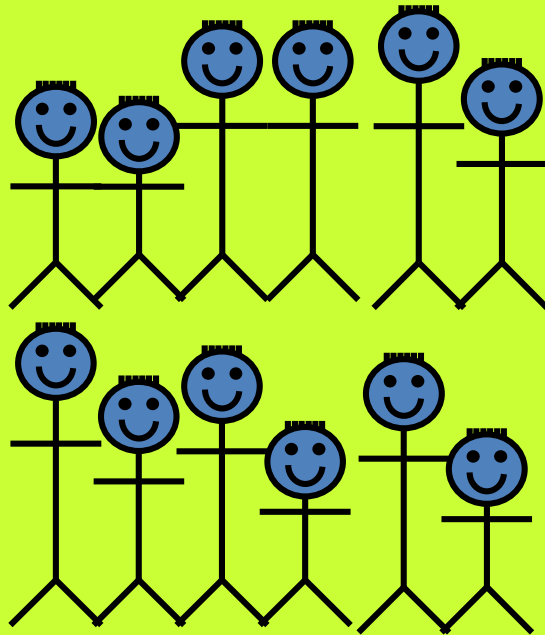
... SIMILAR

C1



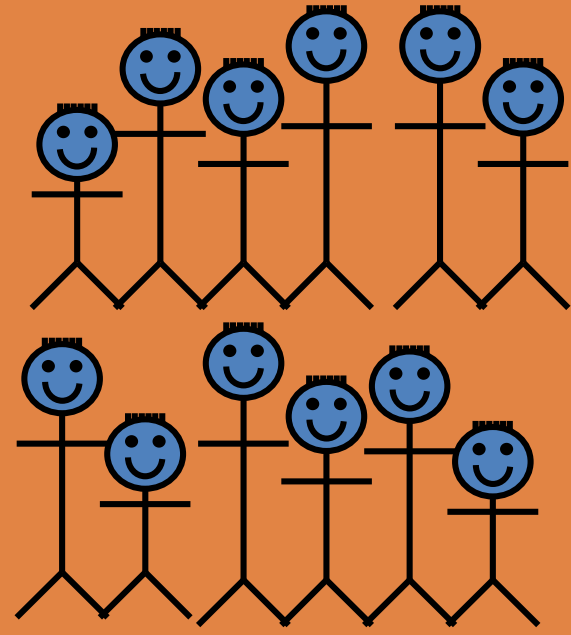
It is *VERY* unlikely that the mean height of our two samples will be exactly the same

C1 Sample



Average height = 162 cm

D8 Sample



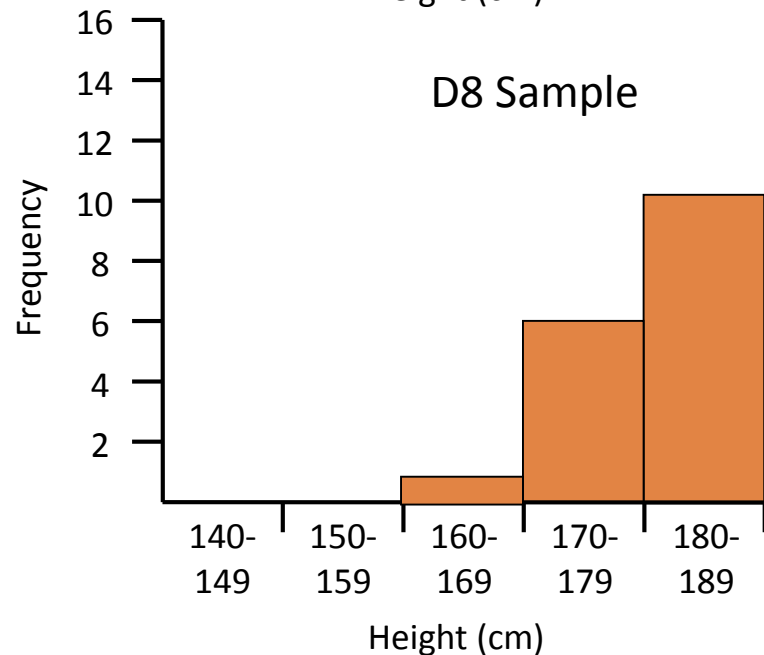
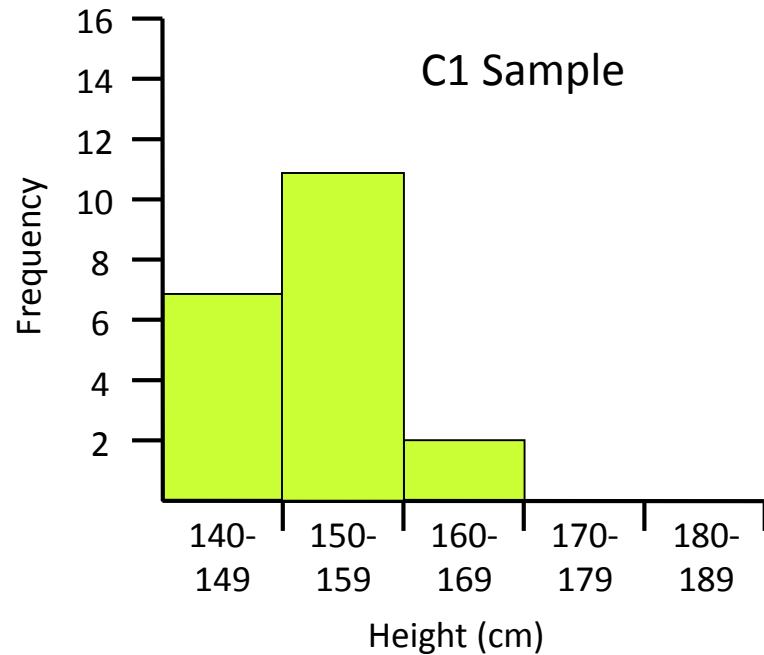
Average height = 168 cm

Is the difference in average height of the samples large enough to be significant?

We can analyse the spread of the heights of the students in the samples by drawing *histograms*

Here, the ranges of the two samples have a small overlap, so...

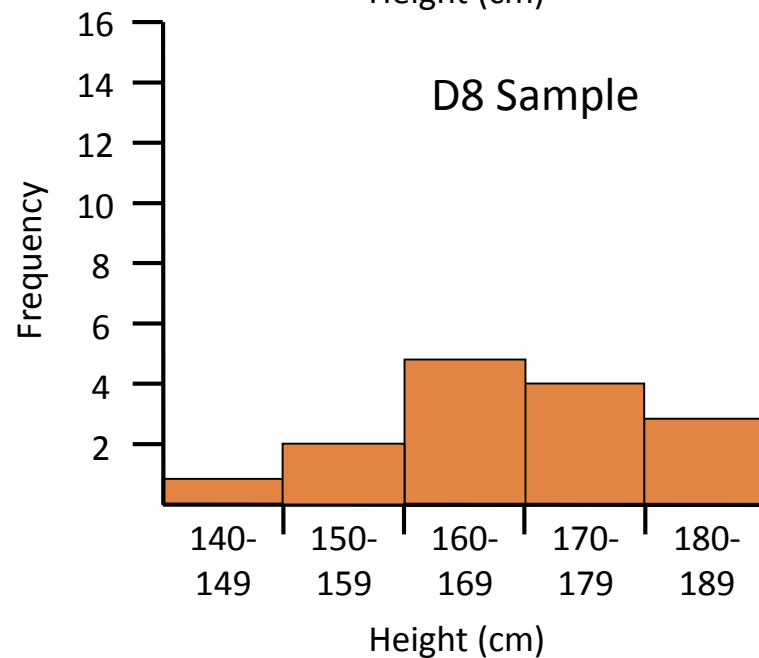
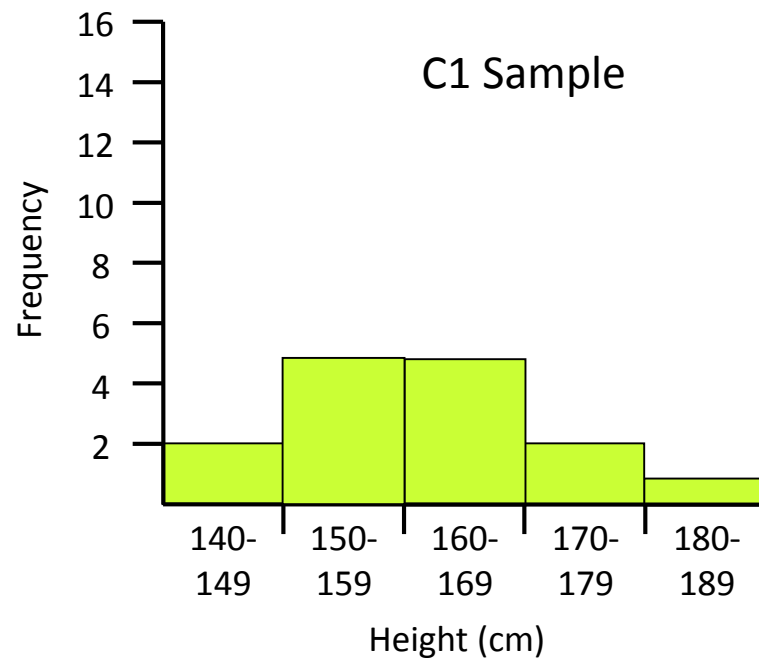
... the difference between the means of the two samples *IS* probably significant.



Here, the ranges of the two samples have a large overlap, so...

... the difference between the two samples may *NOT be* significant.

The difference in means is possibly due to *random sampling error*



To decide if there is a significant difference between two samples we must compare the *mean height* for each sample...

... and the *spread* of heights in each sample.

Statisticians calculate the *standard deviation* of a sample as a measure of the spread of a sample

You *can* calculate standard deviation using the formula:

$$S_x = \sqrt{\frac{\sum x^2 - \frac{(\sum x)^2}{n}}{n - 1}}$$

Where:

S_x is the standard deviation of sample

Σ stands for 'sum of'

x stands for the individual measurements in the sample

n is the number of individuals in the sample

Student's t -test

The Student's t -test compares the averages and standard deviations of two samples to see if there is a significant difference between them.

We start by calculating a number, t

t can be calculated using the equation:

$$t = \frac{(\bar{x}_1 - \bar{x}_2)}{\sqrt{\frac{(s_1)^2}{n_1} + \frac{(s_2)^2}{n_2}}}$$

Where:

\bar{x}_1 is the mean of sample 1

s_1 is the standard deviation of sample 1

n_1 is the number of individuals in sample 1

\bar{x}_2 is the mean of sample 2

s_2 is the standard deviation of sample 2

n_2 is the number of individuals in sample 2

Example Experimental Results

Query	A	B	B-A
1	25	35	10
2	43	84	41
3	39	15	-24
4	75	75	0
5	43	68	25
6	15	85	70
7	20	80	60
8	52	50	-2
9	49	58	9
10	50	75	25

t-Test

- Assumption is that the difference between the effectiveness values is a sample from a normal distribution
- Null hypothesis is that the mean of the distribution of differences is zero
- Test statistic

$$t = \frac{\overline{B-A}}{\sigma_{B-A}} \cdot \sqrt{N}$$

– for the example,

$$\overline{B-A} = 21.4, \sigma_{B-A} = 29.1, t = 2.33, \text{p-value} = .02$$

Setting Parameter Values

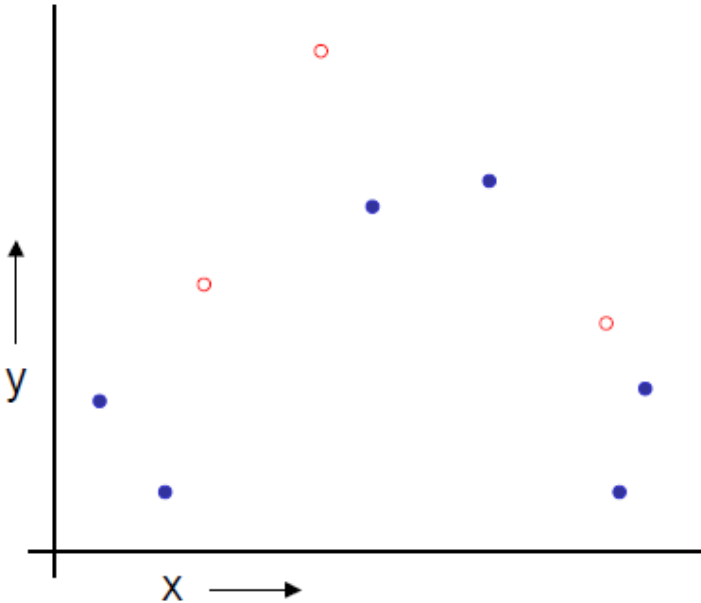
- Retrieval models often contain parameters that must be tuned to get best performance for specific types of data and queries
- For experiments:
 - Use *training* and *test* data sets
 - If less data available, use *cross-validation* by partitioning the data into K subsets
 - Using training and test data avoids *overfitting* – when parameter values do not generalize well to other data

Finding Parameter Values

- Many techniques used to find optimal parameter values given training data
 - standard problem in machine learning
- In IR, often explore the space of possible parameter values by *brute force*
 - requires large number of retrieval runs with small variations in parameter values (*parameter sweep*)
- *Learning to rank* techniques are efficient procedures for finding good parameter values with large numbers of parameters

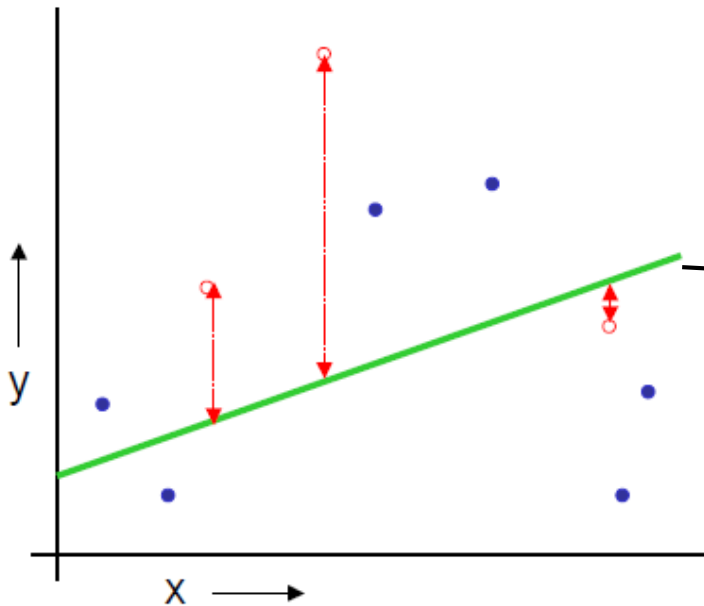
Test set method

- Randomly split some portion of your data
Leave it aside as the **test set**
- The remaining data is the **training data**



Test set method

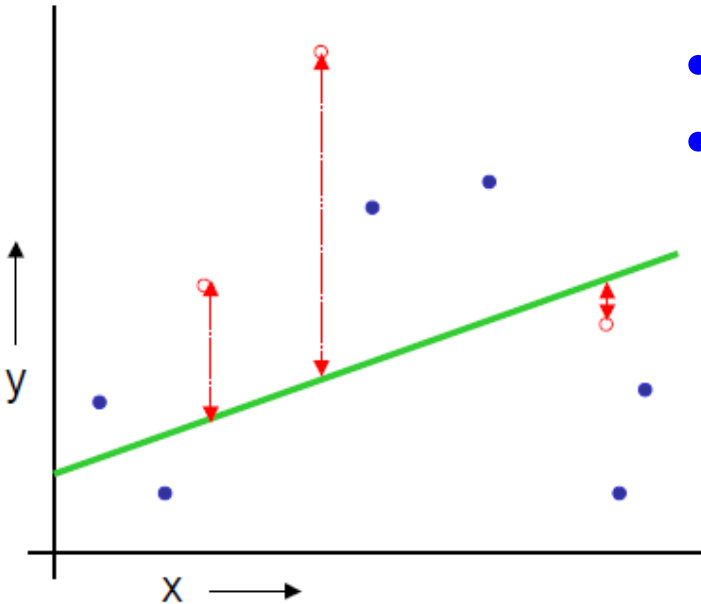
- Randomly split some portion of your data
Leave it aside as the **test set**
The remaining data is the **training data**
Learn a **model** from the training set



This the model you learned.

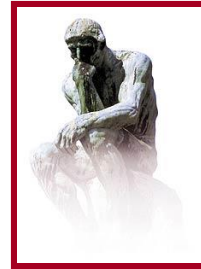
How good is the prediction?

- Randomly split some portion of your data
Leave it aside as the **test set**
- The remaining data is the **training data**
- Learn a **model** from the training set
- Estimate your future performance with the test data



Train test set split

- It is simple
- What is the down side ?



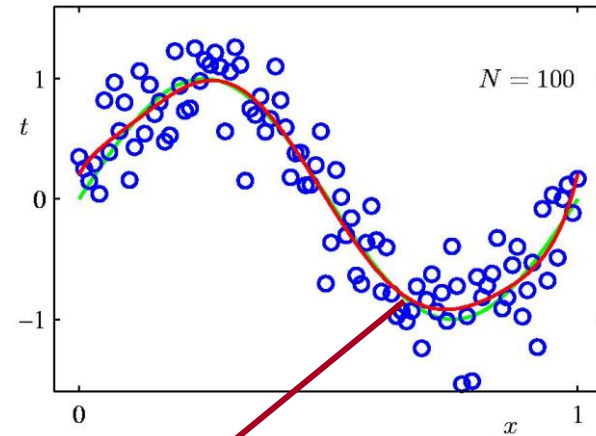
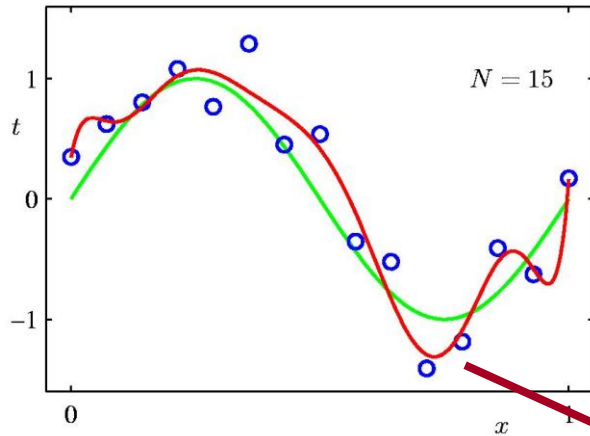
More data is better

With more data you can learn better

Blue: Observed data

Red: Predicted curve

True: Green true distribution



Compare the predicted curves

Train test set split

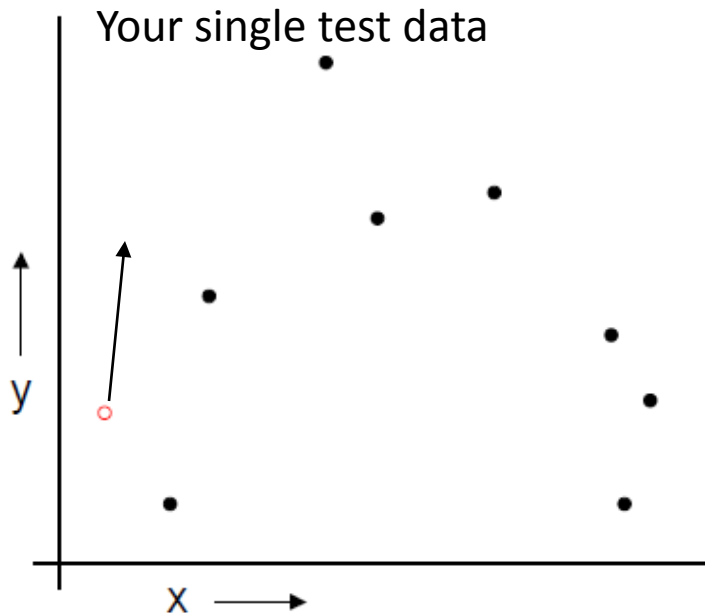
- It is simple
 - What is the down side ?
-
1. You waste some portion of your data.
 2. Luck has more effect on the model

Cross Validation

Recycle the data!



LOOCV (Leave-one-out Cross Validation)



Let say we have N data points
 k be the index for data points
 $k=1..N$

Let (x_k, y_k) be the k^{th} record

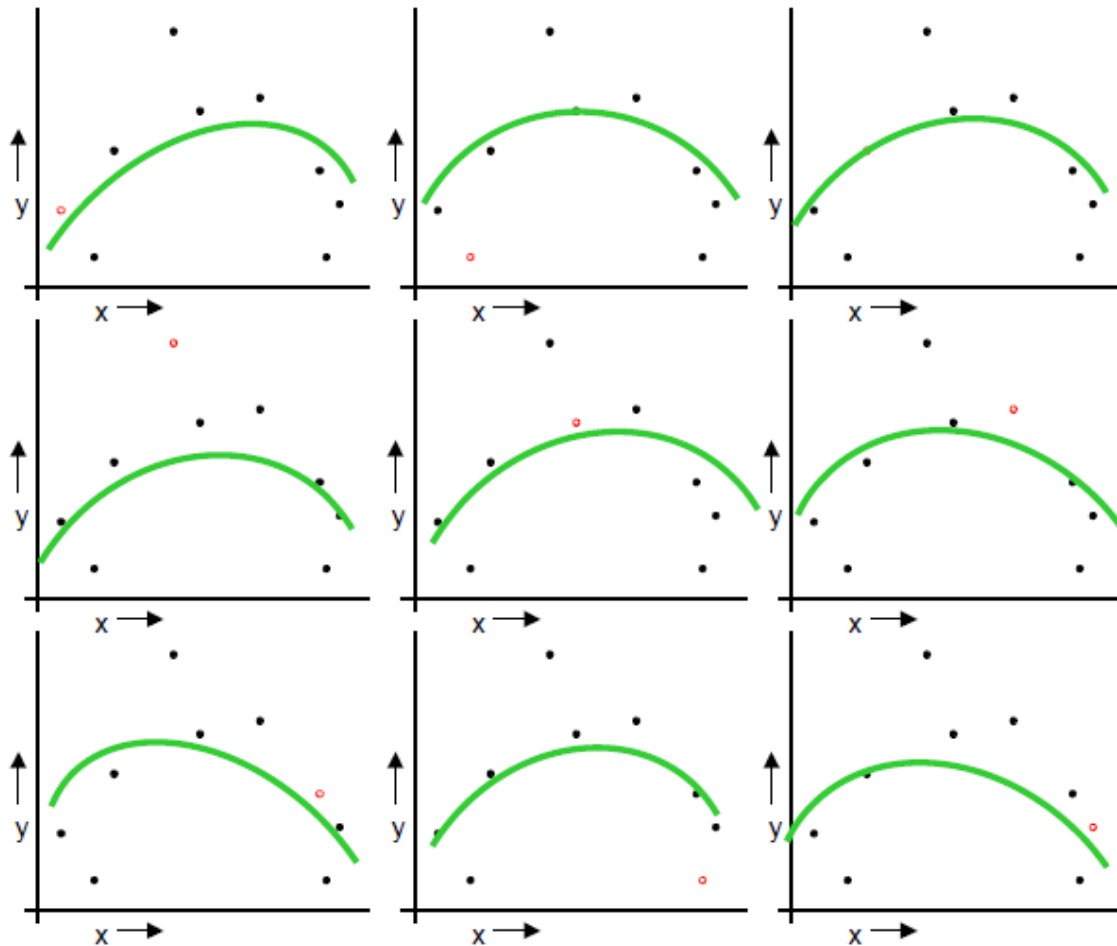
Temporarily remove (x_k, y_k)
from the dataset

Train on the remaining $N-1$
Datapoints

Test your error on (x_k, y_k)

Do this for each $k=1..N$ and report the mean
error.

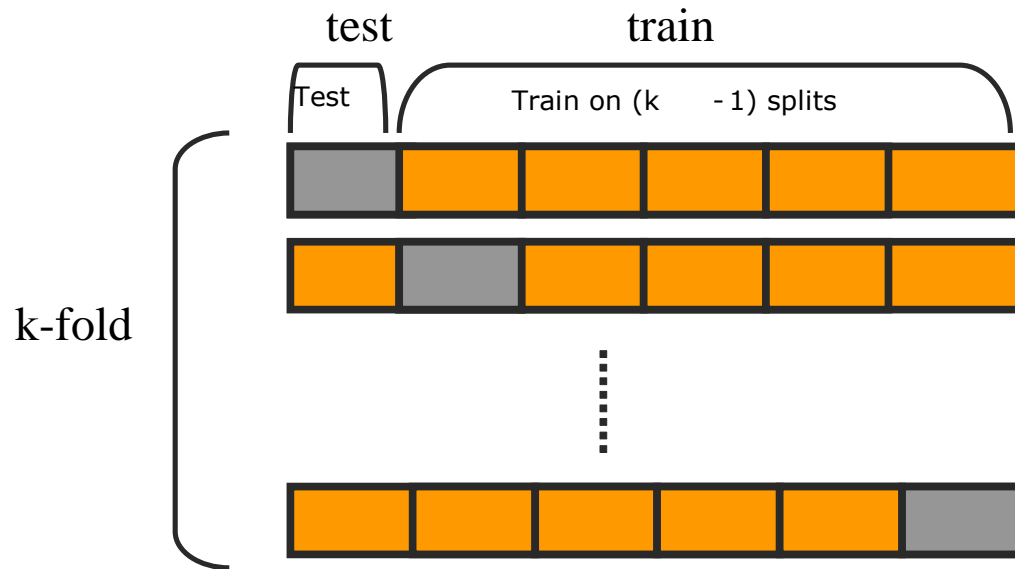
LOOCV (Leave-one-out Cross Validation)



There are N data points..
Do this N times. Notice the
test data is changing each time

$MSE=3.33$

K-fold cross validation



In 3 fold cross validation, there are 3 runs.

In 5 fold cross validation, there are 5 runs.

In 10 fold cross validation, there are 10 runs.

the error is averaged over all runs

Summary

- No single measure is the correct one for any application
 - choose measures appropriate for task
 - use a combination
 - shows different aspects of the system effectiveness
- Use significance tests (t-test)
- Analyze performance of individual queries

Query Summary

