Data Preparation and Distance Measures

Data Preparation

- Feature extraction: derive meaningful features from data.
- Portability: cast data to unify the structure, and to fit the Algorithm.
- Data cleaning.
- Data Reduction.
- Feature Selection.
- Transformation.

Feature Extraction

- Raw data is not always useful when used as is. Instead, representative features need to be computed, resulting in a feature vector per data record
- The composition of the feature vector is domain specific, and problem specific

Feature Selection vs. Feature Extraction

Feature extraction is different from feature selection.

Feature selection involves choosing a subset of the original features,
while feature extraction creates new features based on the original ones.

Both techniques aim to simplify the data and improve the performance of machine learning models.

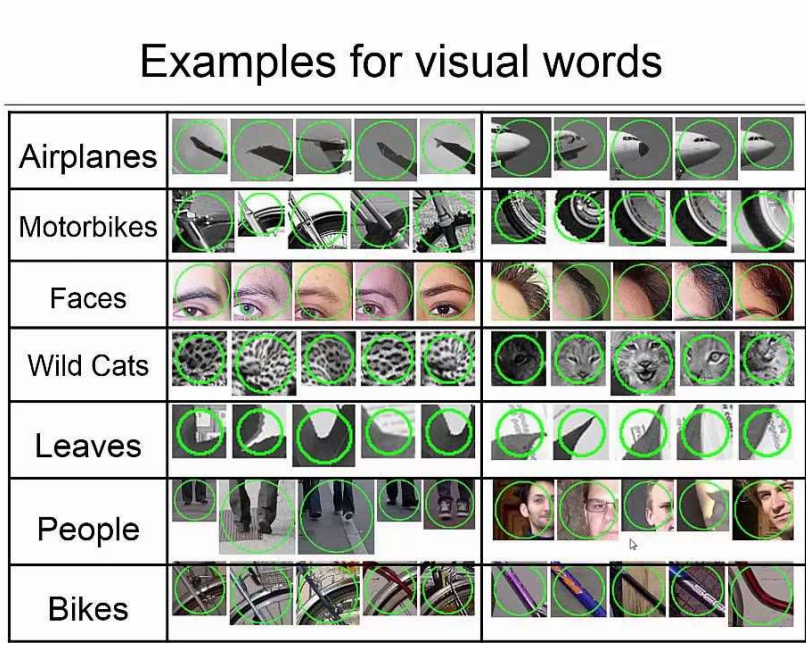
Image Feature Extraction

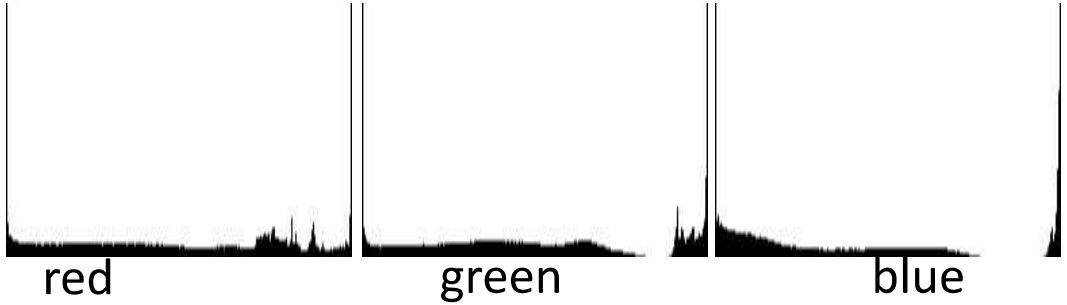
- Color histogram.
- Visual words.

Image Feature Extraction:

In image processing, feature extraction involves capturing relevant information from images.

Techniques like edge detection, texture analysis, and histogram of oriented gradients (HOG) can be used to represent images in a more compact and meaningful way.





ULB

Document Feature Extraction

- Named-entity recognition: persons, organizations, locations, actions, cities
- Stop word removal
- Word count histograms

Extract features from document data
Common techniques used for document feature extraction

1) Bag of Words
The Bag of Words model represents each document as an unordered set of words, ignoring grammar and word order but keeping track of word frequency
Problem: we loose word order

2) Extracting n gram
N-grams represent contiguous sequences of N items (usually words) in a document.
Unigrams (single words), bigrams (pairs of consecutive words), and trigrams (triplets of consecutive words) are common choices.
N-grams capture local patterns and dependencies between words.

1 gram is used for token
2 grams for token pair

Stop word removal is a preprocessing step in NLP that involves removing common, non-meaningful words like “the” and “and” from text data

Type Portability

- Text to numeric
- Time series to discrete sequence
- Time series to numeric
- Discrete sequence to numeric
- Spatial to numeric

1) Number to categorical data
Number is divided into several categories by dividing them based on range
Equi width - divided into range of a-b in such a way that b-a is always same
Equi depth - divided into range of a-b in such as way that number of records is always same in that range
Equi log ranges - Range [min,max] is divided into n ranges [ai, bi]. For all i, log(bi)- log(ai) is the same

2) Categorical to Numerical data
It is possible to convert categorical attributes to binary form and then use numeric algorithms on binarized data
Suppose, we have categorical attribute with n possible values,
then n different binary attributes are created
Each binary attribute corresponds to one possible value of categorical attribute
For a record, one of the n binary attributes take value 1 and remaining take value 0

Type Portability

Text Documents

- Are all words important ?
- Are different words equally important ?
- Entity extraction.
- Bag of words.

Type Portability - Numeric to categorical (discretization):

- Divide a continuous range into n ranges.
- Age: [0,5], [6,12], [13,18], [18,30], [30,60], 60+
- Lossy process. Age 5 and 6 are equally different as Age 1, 11.
- How to discretized ?

Type Portability - Numeric to categorical (discretization):

- Equi-width ranges. Range [min,max] is divided into n ranges $[a_i, b_i]$.
  - For all i, $b_i - a_i$ is the same.
  - Good for data that is uniformly distributed.
- Equi-log ranges. Range [min,max] is divided into n ranges $[a_i, b_i]$.
  - For all i, $\log(b_i) - \log(a_i)$ is the same.
  - Good for data that is shows an exponential distribution.
- Equi-depth ranges
  - Each range has the same number of records.
  - Sort then select the division points.

Data Cleaning?

In real world, Data can be:
1) Incomplete - Null values present in some field
2) Noisy - Error in data or outliers
3) Inconsistent - Discrepancy in data collected from two different sources, Age and DOB different
4) Duplicate

Data in the real world is dirty

- incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
  - e.g., occupation=“ ”
- noisy: containing errors or outliers
  - e.g., Salary=“-10”
- inconsistent: containing discrepancies in codes or names
  - e.g., Age=“42” Birthday=“03/07/1997”

Why Is Data Dirty?

- Incomplete data may come from
  - "Not applicable" data value when collected.
  - Different considerations between the time when the data was collected and when it is analyzed.
  - Human/hardware/software problems.
- Noisy data (incorrect values) may come from
  - Faulty data collection instruments.
  - Human or computer error at data entry.
  - Errors in data transmission.
- Inconsistent data may come from
  - Different data sources.
  - Functional dependency violation (e.g., modify some linked data).
- Duplicate records also need data cleaning.

How to Investigate ?

- Descriptive Analytics:
  - Null count.
  - Repeated data.
  - Numeric data: max, min, mean, median, variance.
  - Images: image dimensions, resolution, average color, color histogram.
  - GPS: sampling rate (max, min, mean), segment speed (max, min, mean), bounding box.

Measuring the Central Tendency (1 number summary)

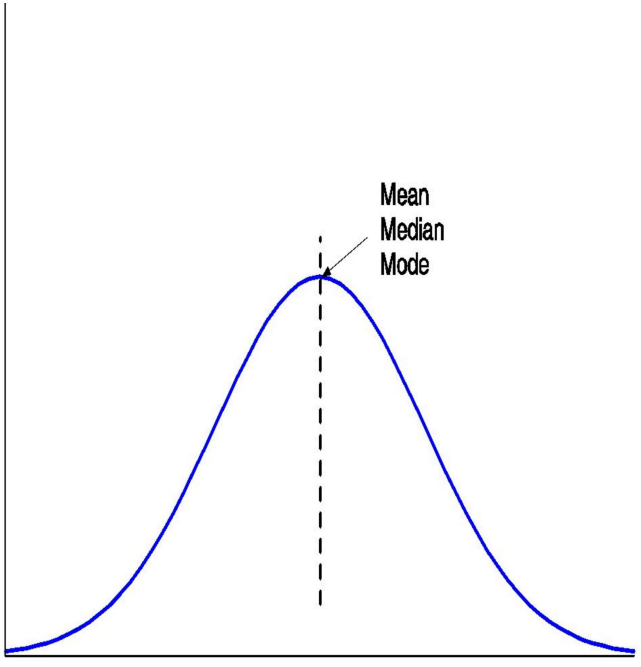
- Mean (algebraic measure) (sample vs. population):
  - Weighted arithmetic mean:
  - Trimmed mean: chopping extreme values

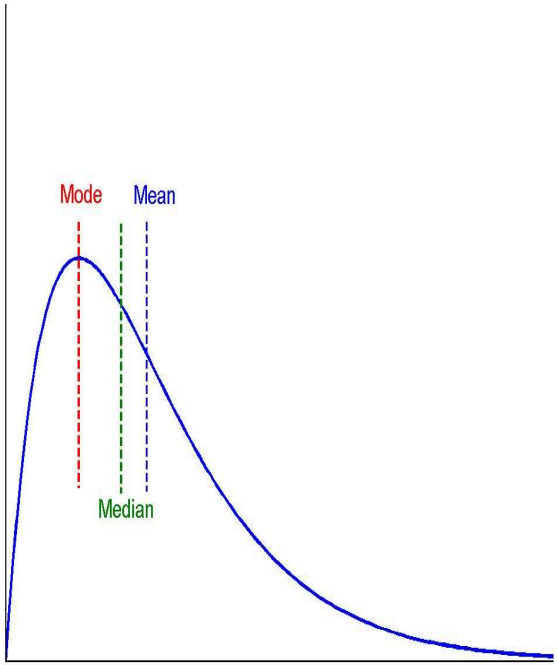
$$\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

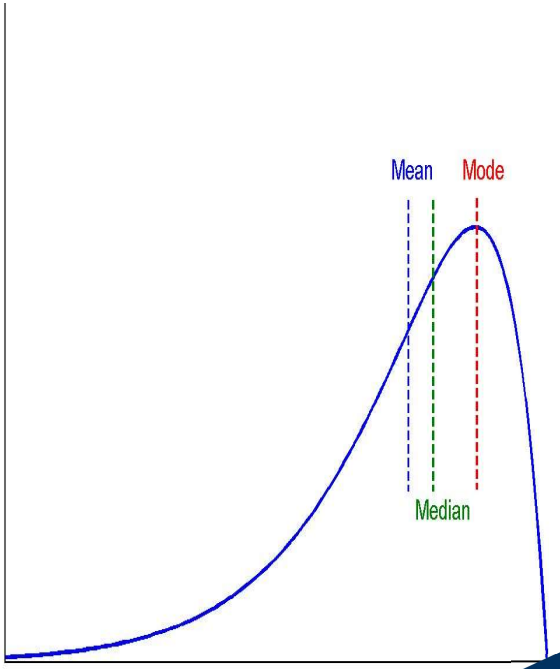
- Median: A holistic measure
  - Middle value if odd number of values, or average of the middle two values otherwise
- Mode
  - Value that occurs most frequently in the data
- 13, 13, 13, 13, 14, 14, 16, 18, 21
  - Mean= sum(13, 13, 13, 13, 14, 14, 16, 18, 21)/9
  - Median= 14, Mode= 13, Range= 8

Symmetric vs. Skewed Data (3 number summary)

- Median, mean and mode of symmetric, positively and negatively skewed data







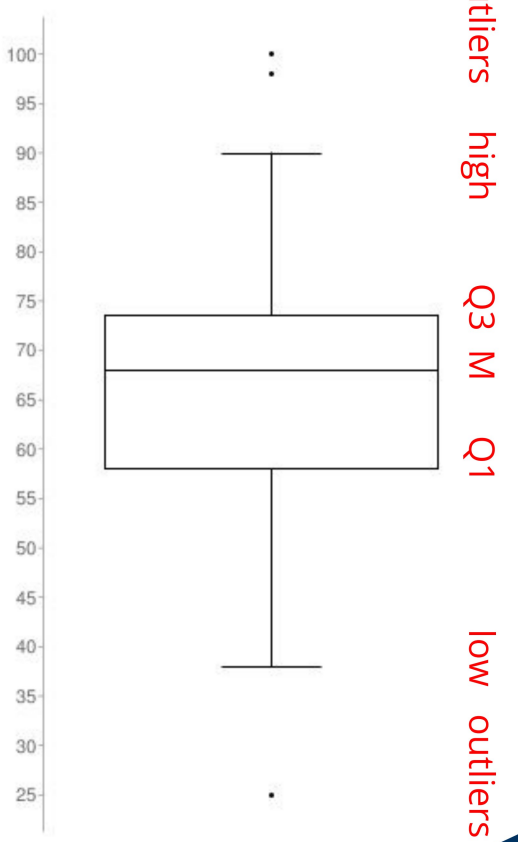
negative skewed
left skewed

Measuring the Dispersion of Data (5 number summary)

- Quartiles: Q1 (25th percentile), Q3 (75th percentile)
- Inter-quartile range: IQR = Q3 – Q1
- Five number summary: low, Q1, median, Q3, high
- Boxplot: ends of the box are the quartiles, median is marked, whiskers, and plot outlier individually
- Outlier: *usually*, a value higher/lower than 1.5 x IQR

Data:
[25,38,52,57,57,58,58,58,58
,58,58,63,66,66,67,67,68,68
,68,68,68,68,69,70,70,70,70
,72,73,75,75,76,76,78,79,90
,98,100]

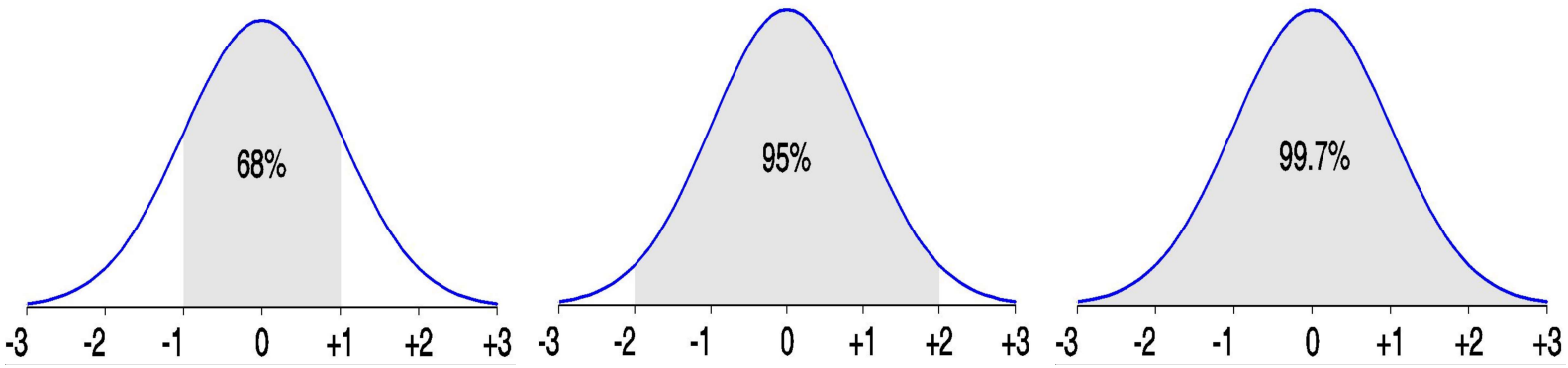
Population size: 38
Median: 68
Minimum: 25
Maximum: 100
First quartile: 58
Third quartile: 73.5
Interquartile Range: 15.5
Outliers: 25 100 98



https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51

Comparing with Normal Distribution

- From μ–σ to μ+σ: contains about 68% of the measurements (μ: mean, σ: standard deviation)
- From μ–2σ to μ+2σ: contains about 95% of it
- From μ–3σ to μ+3σ: contains about 99.7% of it



ULB

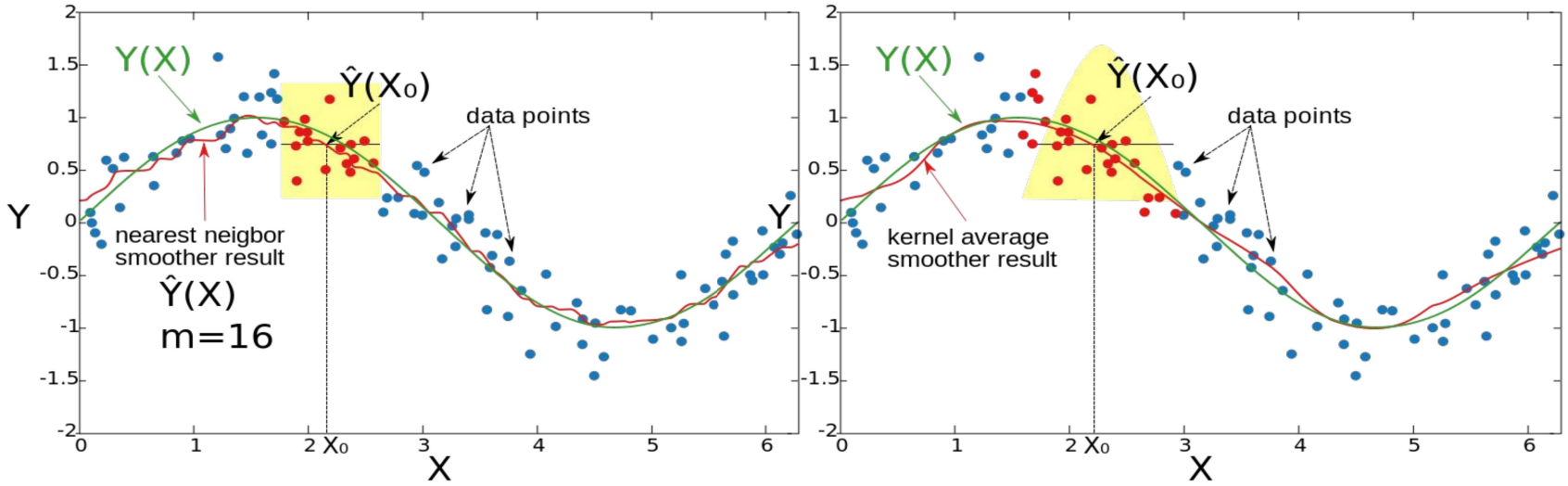
How to Clean Missing and Inconsistent Entries ?

- Delete the entire record: safe, but not practical in most cases.
- Impute/estimate the missing values: can lead to data bias.
- Global constant, mean, substitute with a value of a similar record, interpolation, extrapolation, regression, cluster representative.
- Change the mining Algorithm to tolerate missing values.

How to Clean Noisy Entries ?

Kernel smoothing:

- kNN smoother replaces a value with the average of itself and the k nearest values.
- kernel average smoother replaces a value with the weighted average of itself and its neighbors in a fixed size window



Source: https://en.wikipedia.org/wiki/Kernel_smoother

ULB

How to Clean Noisy Entries ?

- Binning:
  - sort data and partition into equally sized bins, then smooth by bin mean, median, boundary.
- Regression:
  - smooth by fitting the data into a regression function.
- Change the mining Algorithm to tolerate noise.

Binning Example

Sorted data for price: 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

- Partition into equal-frequency (equi-depth) bins:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34
- Smoothing by bin means:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29

Similarity and Distance

- Given two objects a, b, determine their similarity sim(a,b) or distance dist(a,b).
- Most of the DM Algorithms require the computation of similarity or distance.
- dist(a,b)= 1.43 is not interesting, while if we also know that dist(a,c)=0.5 it becomes interesting as we know that a is more similar to c than b.
- Useful DM analysis depends on expressive distance function, which depends on good selection of features and good normalization.

L_p-norm

$$Dist(\overline{X}, \overline{Y}) = \left(\sum_{i=1}^{d} |x_i - y_i|^p \right)^{1/p}.$$

- Euclidean (p=2), Manhattan (p=1).
- Compute the L_p-norm between (1, 2), (3, 4) for p = 1, 2, ∞.

Euclidean
distance between two points

Manhattan
Distance between two points measured along axes at right angle
it is distance measure that is calculated by taking sum of distances between x and y co-ordinates

Generalized Minkowski

Generalized Lp distance
Generalized form of Euclidean and Manhattan distance

$$Dist(\overline{X}, \overline{Y}) = \left(\sum_{i=1}^{d} a_i \cdot |x_i - y_i|^p \right)^{1/p}.$$

- According to the domain, some features may be more important than others.
- Generally p is set to the number of dimensions.
- X, Y are represented as two points in the p-dimensional space. Their distance represents the length of the line connecting them.

Scaling and Normalization

- Multidimensional data has different scales for the different dimensions, resulting in features dominating others in distance computations.

Example:

P1: ULB, 1/1/2018, 10:00

P2: C. d'Ixelles, 1/1/2018, 11:00

P3: Beijing, 1/1/2018, 10:30

d(P1, P3)= (8000 km, 30 minutes).

The Euclidean distance= 8000

d(P1, P2)= (1 km, 60 minutes)=1800

The Euclidean distance= 60

Which pair is more close ?

In case of multidimensional data, different dimensions have different scales resulting in one feature to dominate other in distance computations.
To solve this, use scaling and normalization

Scaling and Normalization

Normalization changes data in such a way that it has a mean of 0 and a standard deviation of 1
For this each value i of attribute j is replaced with:

- Normalization replaces each value with:

$$z_i^j = \frac{x_i^j - \mu_j}{\sigma_j}$$

- Assuming a normal distribution of the attribute values, most of the values will be normalized to the range [-3, 3].
- Scaling maps the values to the range [0, 1]

$$y_i^j = \frac{x_i^j - min_j}{max_j - min_j}$$

Scaling involves transforming the values of the features in your dataset so that they all have a similar scale.

It is common that we obtain data that is not in same units or range. This can cause issue when we are trying to compare of combine data, or when we are trying to build model out of this

Scaling is used when we want to compare data in different units or when we want to ensure that outliers do not have much influence on our results.

Normalization ensures that data points are centered around 0 and that they are all within 1 standard deviation of the mean. It is used when we want to compare data in different units or when we want to ensure that outliers do not have much influence on our results.

Edit Distance

There are operations like
r: replace one character by another
i: insert one character
d: delete one character

Edit distance between two strings is the minimum number of string operations needed to transform string 1 to string 2

- How similar is Mahmoud to Mohammed ?

M a h * m o u d

M o h a m m e d

✓ r ✓ i ✓ r r
✓

- Edit distance= 4
- If replace costs 2 (Levenshtein), then distance is 7

Finding Minimum Edit Distance

For two sequences a and b with lengths i and j respectively, the Levenshtein distance is defined as:

$$
\mathrm{ld}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \mathrm{ld}(i-1,j) + 1 \\ \mathrm{ld}(i,j-1) + 1 \\ \mathrm{ld}(i-1,j-1) + k \end{cases} & \text{otherwise.} \end{cases}
$$

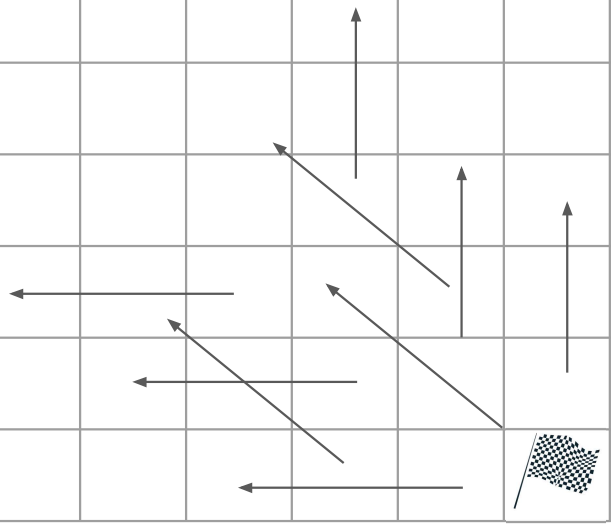
where $k = 0$ if $(a_i = b_j)$, 1 otherwise.

Recursive Computation of Edit Distance

$$\min \begin{cases} ld(i-1,j) + 1 \\ ld(i,j-1) + 1 \\ ld(i-1,j-1) + k \end{cases}$$

What is the complexity ?

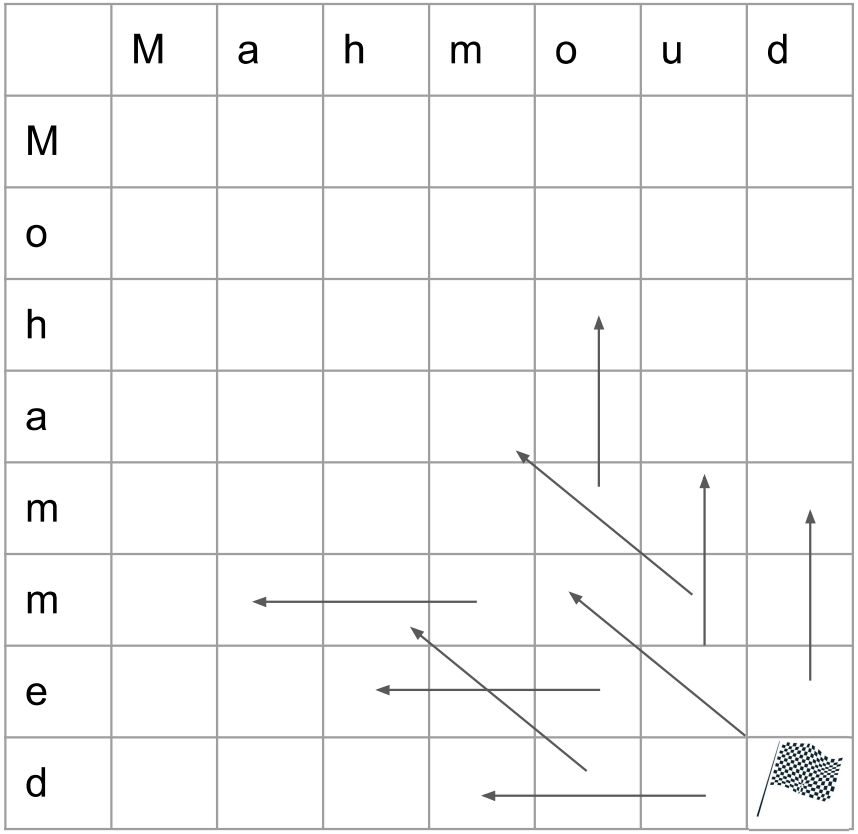
	M	a	h	m	o	u	d
M							
o							
h							
a							
m							
m							
e							
d							



Recursive Computation of Edit Distance

$$\min \begin{cases} ld(i-1,j) + 1 \\ ld(i,j-1) + 1 \\ ld(i-1,j-1) + k \end{cases}$$

- Time: O(i.j)
- Space: O(i.j)
- Backtrace: O(i+j)



ULB

Dynamic Programming

Initialization

	*	M	a	h	m	o	u	d
*	0	1	2	3	4	5	6	7
M	1							
o	2							
h	3							
a	4							
m	5							
m	6							
e	7							
d	8							

Dynamic Programming

row by row, or column by column

```
for i ← 1 to |s1|
do for j ← 1 to |s2|
   do m[i, j] = min{m[i − 1, j − 1] + if (s1[i] = s2[j]) then 0 else 1fi,
                    m[i − 1, j] + 1,
                    m[i, j − 1] + 1}
return m[|s1|, |s2|]
```

What is the complexity ?

	*	M	a	h	m	o	u	d
*	0	1	2	3	4	5	6	7
M	1	0	1	2	3	4	5	6
o	2	1	1	2	3	3	4	5
h	3	2	2	1	2	3	4	5
a	4	3	2	2	2	3	4	5
m	5	4	3	3	2	3	4	5
m	6	5	4	4	3	3	4	5
e	7	6	5	5	4	4	4	5
d	8	7	6	6	5	5	5	

Dynamic Programming

row by row, or column by column

```
for i ← 1 to |s1|
do for j ← 1 to |s2|
   do m[i,j] = min{m[i−1,j−1] + if (s1[i] = s2[j]) then 0 else 1fi,
                   m[i−1,j] + 1,
                   m[i,j−1] + 1}
return m[|s1|, |s2|]
```

- Time: O(i.j)
- Space: O(i.j)
- Backtrace: 0

Can we do better ?

	*	M	a	h	m	o	u	d
*	0	1	2	3	4	5	6	7
M	1	0	1	2	3	4	5	6
o	2	1	1	2	3	3	4	5
h	3	2	2	1	2	3	4	5
a	4	3	2	2	2	3	4	5
m	5	4	3	3	2	3	4	5
m	6	5	4	4	3	3	4	5
e	7	6	5	5	4	4	4	5
d	8	7	6	6	5	5	5	

Dynamic Programming

row by row, or column by column

```
for i ← 1 to |s1|
do for j ← 1 to |s2|
    do m[i,j] = min{m[i−1,j−1] + if (s1[i] = s2[j]) then 0 else 1fi,
                    m[i−1,j] + 1,
                    m[i,j−1] + 1}
return m[|s1|, |s2|]
```

We only need two rows/columns

- Time O(m.n)
- Space: O(min(m.n))

	*	M	a	h	m	o	u	d
*								
M								
o								
h								
a	4	3	2	2	2	3	4	5
m	5							
m								
e								
d								

Other Distance Functions

- Frechet distance for spatial and spatiotemporal sequences
- Histogram distance (Earth Mover)
- Longest common sub-sequence.
- Cosine similarity
- Graph edit distance
- Supervised similarity functions

Readings

- The topics discussed in this lecture are explained in Chapter 2,3 of:
  - Charu C.Aggarwal. Data Mining The Textbook, Springer.