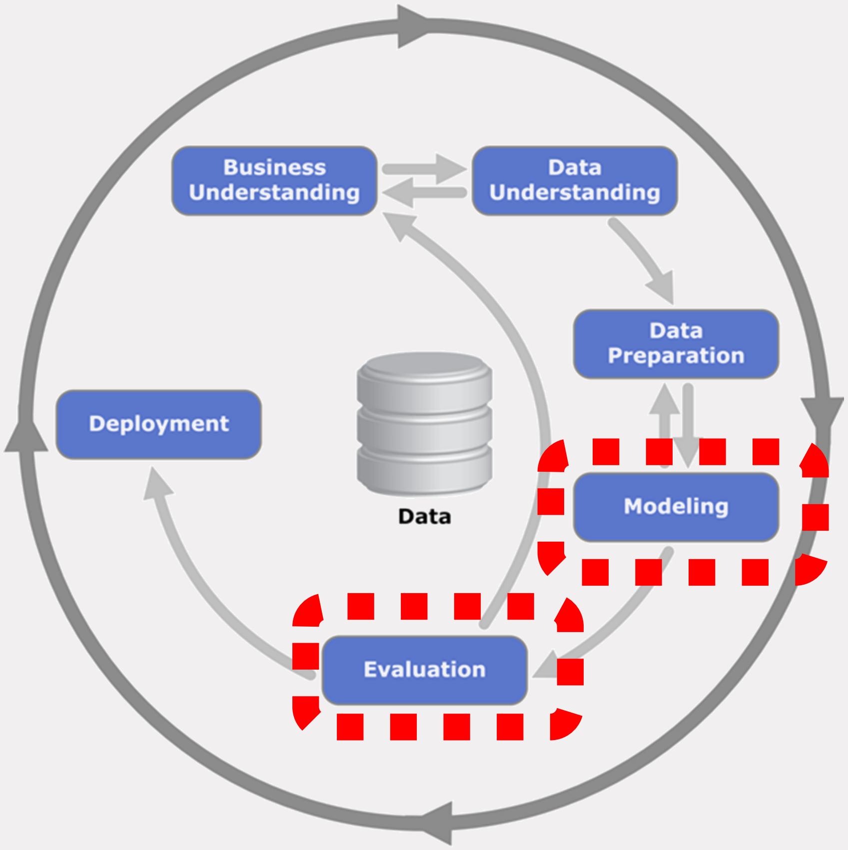
1BM110: Data driven AI
Temporal data mining

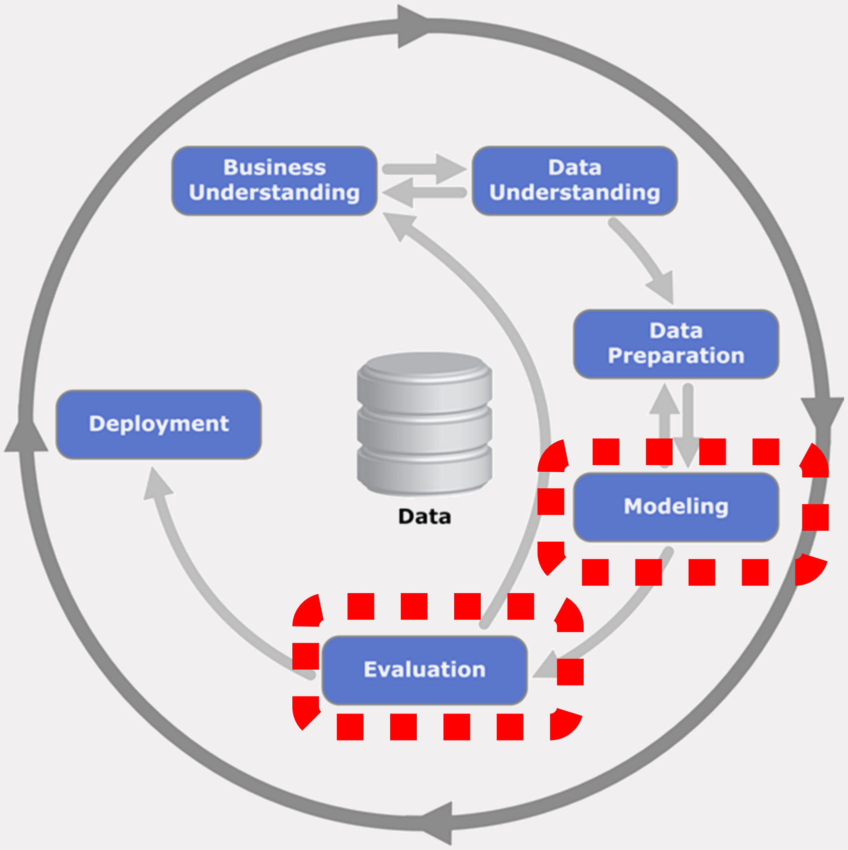
Dr. Yingqian Zhang

Previous lecture



- Several ML models
- Bias-Variance tradeoff
- How to choose models?

Previous lecture



Today:

Temporal data: need special attention

(Grossmann et al. Chapter 6)

Definitions

- A sequence of ordered time values, called observation times $t_1 \leq t_2, \leq \cdots \leq t_T$

- Attribute values at these times

$$x_1 \leq x_2 \leq \cdots \leq x_T$$

- Temporal data (time stamped data):

$$x = ((t_1, x_1), (t_2, x_2), \ldots (t_T, x_T))$$

Time Sequences/Series, State/Event Sequence

- A *time sequence:* a sequence of time-stamped data with measurements
  - Temperatures from sensor <(08:01, 20), (08:15, 20.1), (08:55, 19.8) …>
- A *time series:* a time sequence with measurements at a fixed temporal rate
  - Temperatures: <(day1, 20), (day2, 25), … (day100,22)…>
- A *state sequence:* a time sequence where the state variable contains a finite number of possible values.
  - Weather: <(day1, cloudy), (day2, rainy), (day3, sunny)…>, or simply <cloudy, rainy, sunny…> when time is not important

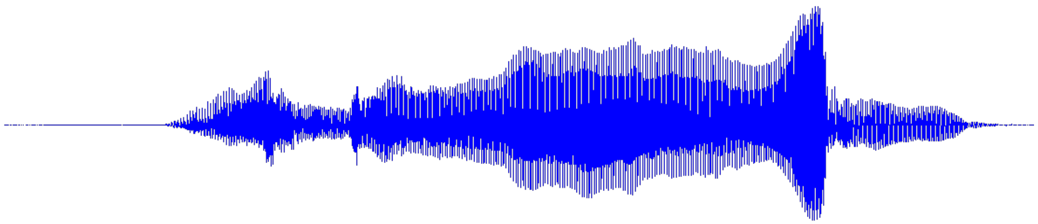
Temporal Data Mining

- Tasks with temporal data: classification, regression, or clustering
- *Sequential data is non i.i.d. (independent and identical distributed)!*
- Techniques to represent temporal data
  - *Time warping: aims* at finding a similarity measure for time sequences which can be used later on for standard DM/ML tasks.
  - *Response feature analysis:* extract characteristic features of the time sequence, then standard ML models can be used
  - *Model based (Markov chains):* targets at understanding the structural behavior of state sequences; the time sequence as direct input to the model for DM/ML tasks
    - *Recurrent Neural Networks (RNNs): a popular approach to sequence modeling problems (next week)*
  - *Association analysis:* derive behavioral rules from event sets.

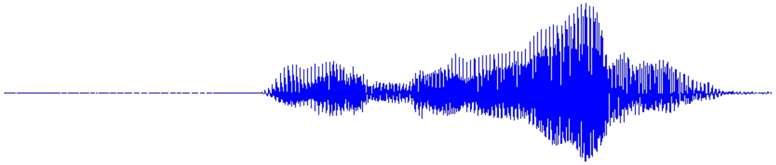
Dynamic time warping

Group the similar items

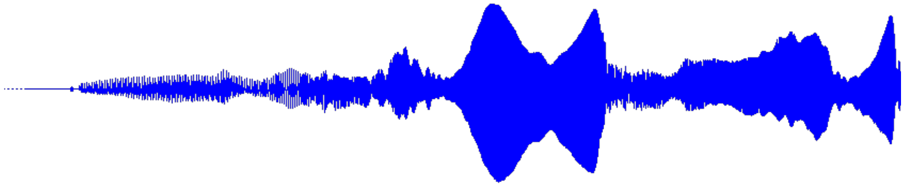
“Slow” Hello



“Fast” Hello



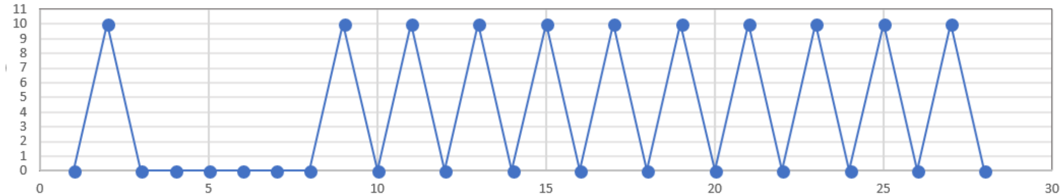
Good Morning



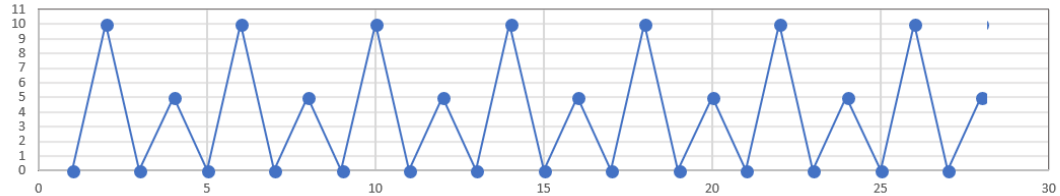
Euclidean distance measure is not a good choice!

Which would you group together?

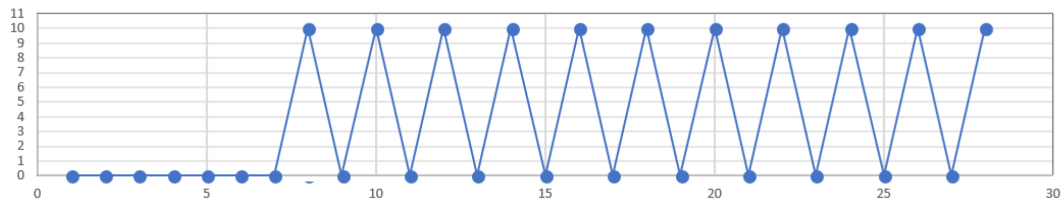
A:



B:



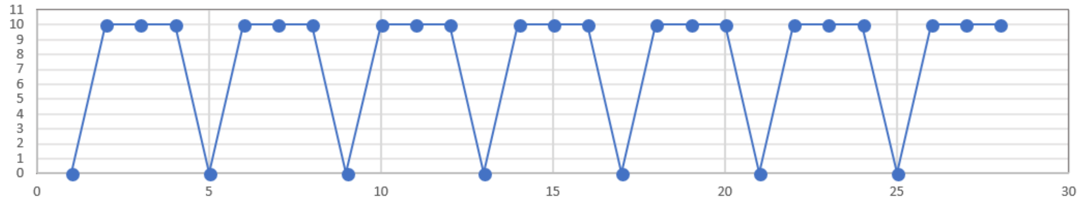
C:



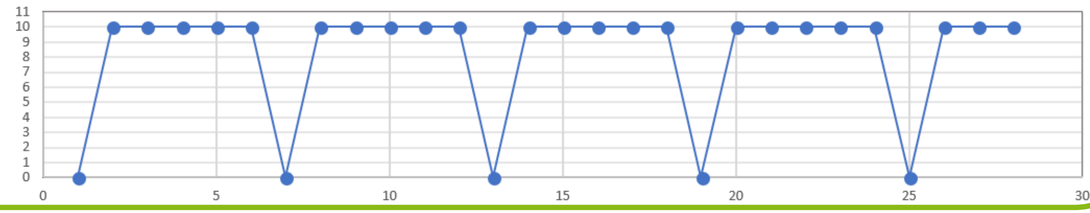
Good Property # 1: Small differences in behavior should not affect distance too much

Which would you group together?

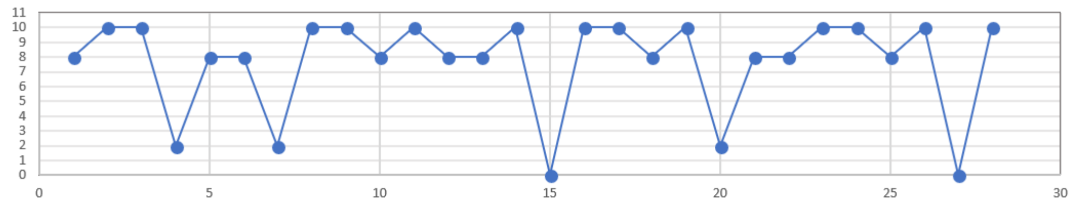
A:



B:



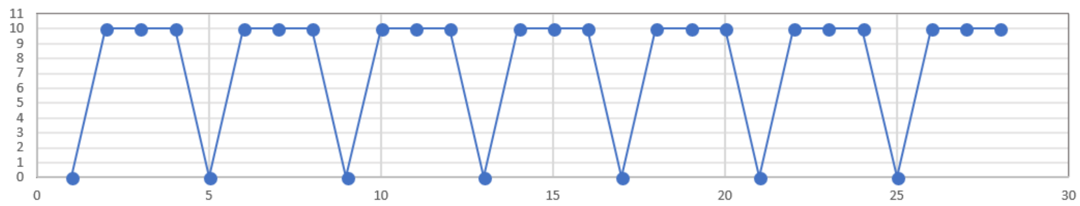
C:



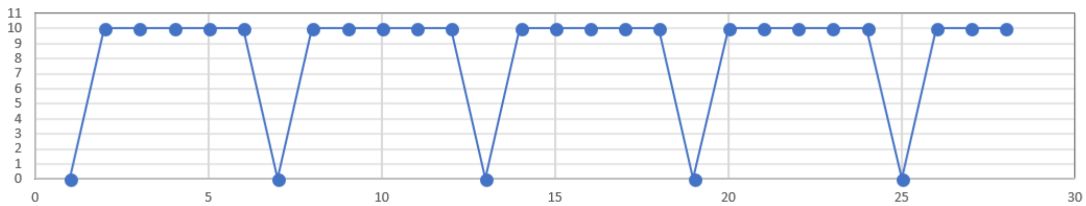
Good Property # 2: Out-of-phase or stretched sequences should not have large distances

Mapping of two sequences: method 1

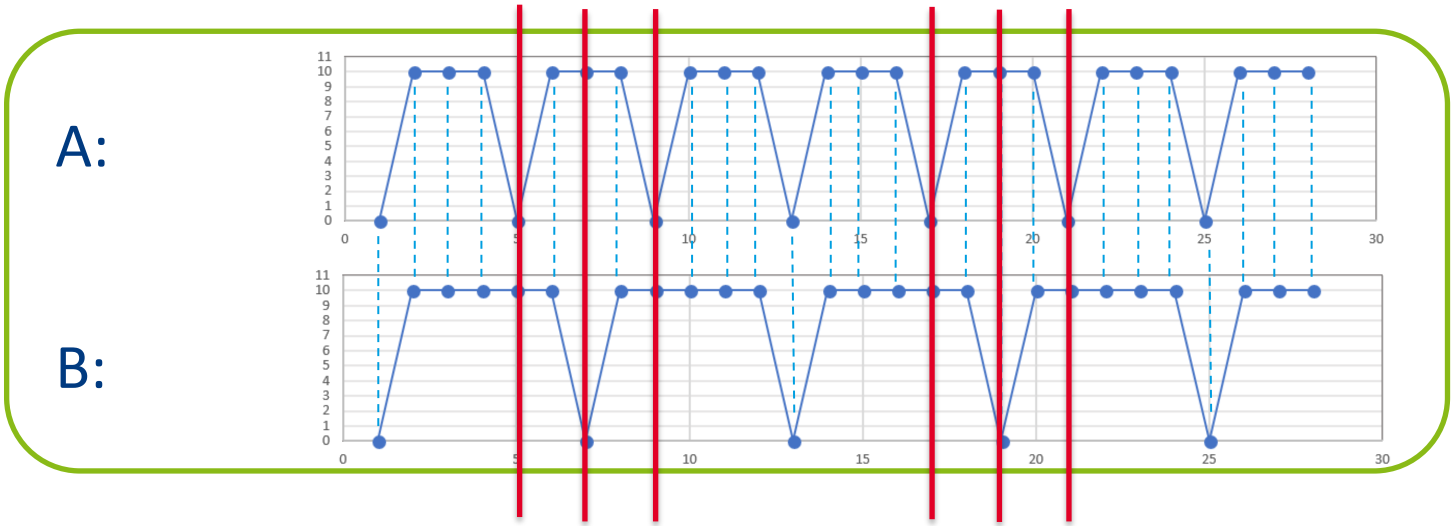
A:



B:



Mapping of two sequences: method 1

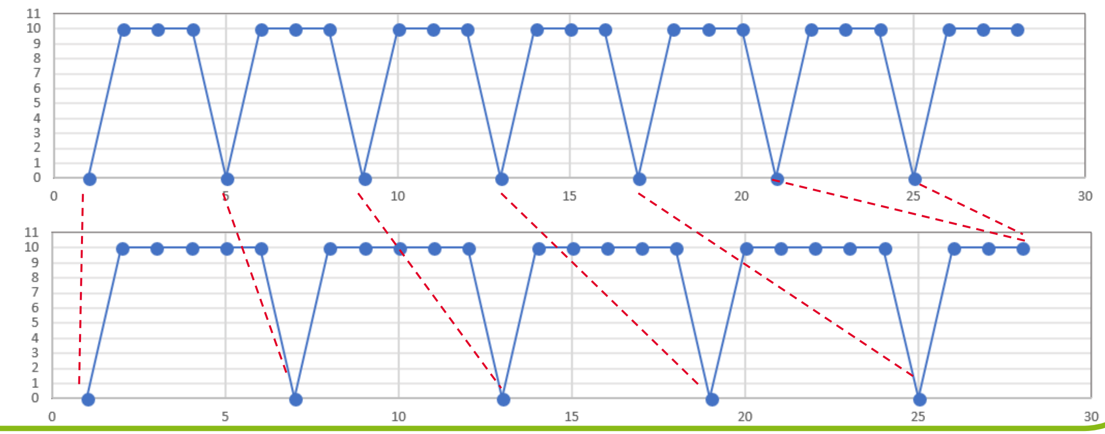


60

Mapping of two sequences: method 2

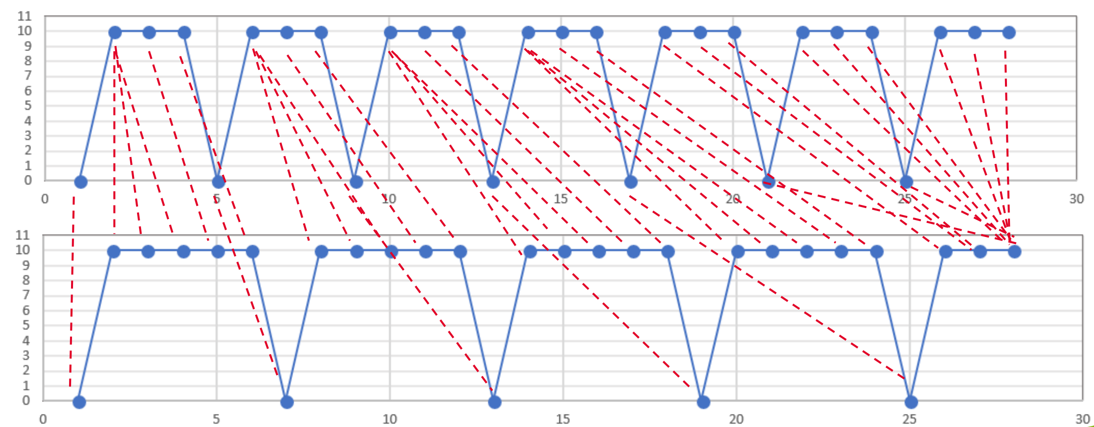
A:

B:



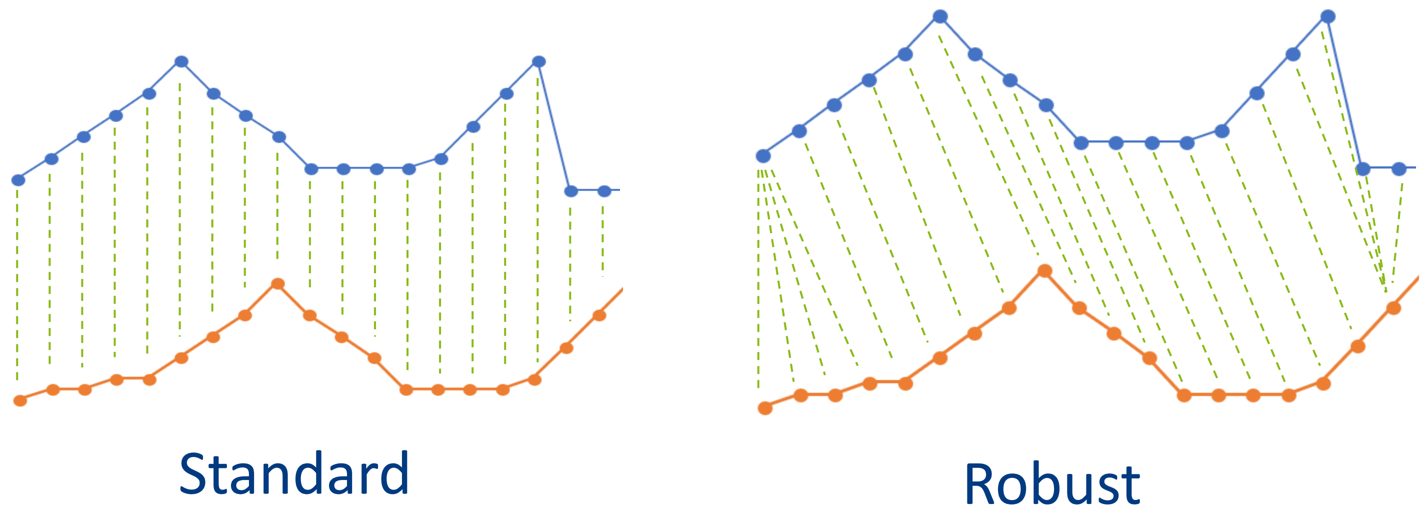
A:

B:



20

Dynamic Time Warping (DTW)



- (left) Not perfect: $Distance = \sum_{i=1}^{N} |blue_i - orange_i|$, or general Euclidean distance

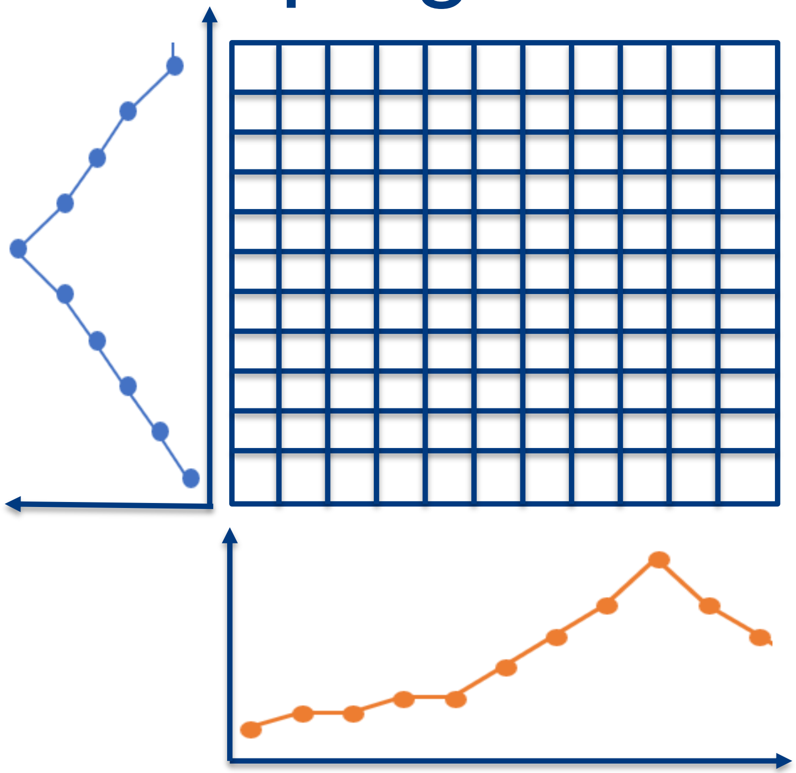
$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

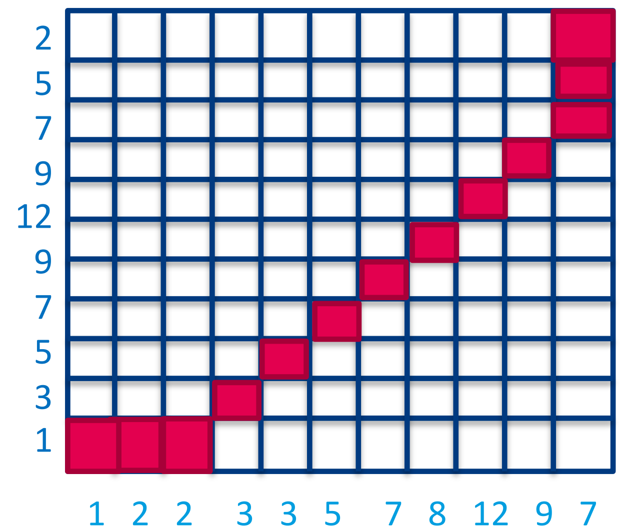
- (right) Better: Dynamic time warping allows the calculation of similarity in the presence of distortions (warps) in the time axis!

Dynamic Time Warping

- Developed in 1983
- Calculate distance between two curves in the presence of distortions (warps) in the time axis.
- Time Warp => distortions in time axis
- Main idea: Extract the best mapping that *minimizes the total distance*

Time-warping distance matrix





- Matrix (right): (1,2,2,3,3,5,7,8,12,9,7) are values for orange sequence;
  (1,3,5,7,9,12,9,7,5,2) are values for blue sequence
- The DTW algorithm finds a warping path for two sequences with minimal costs (distances); based on *dynamic programming*

Step1: fill out matrix

A

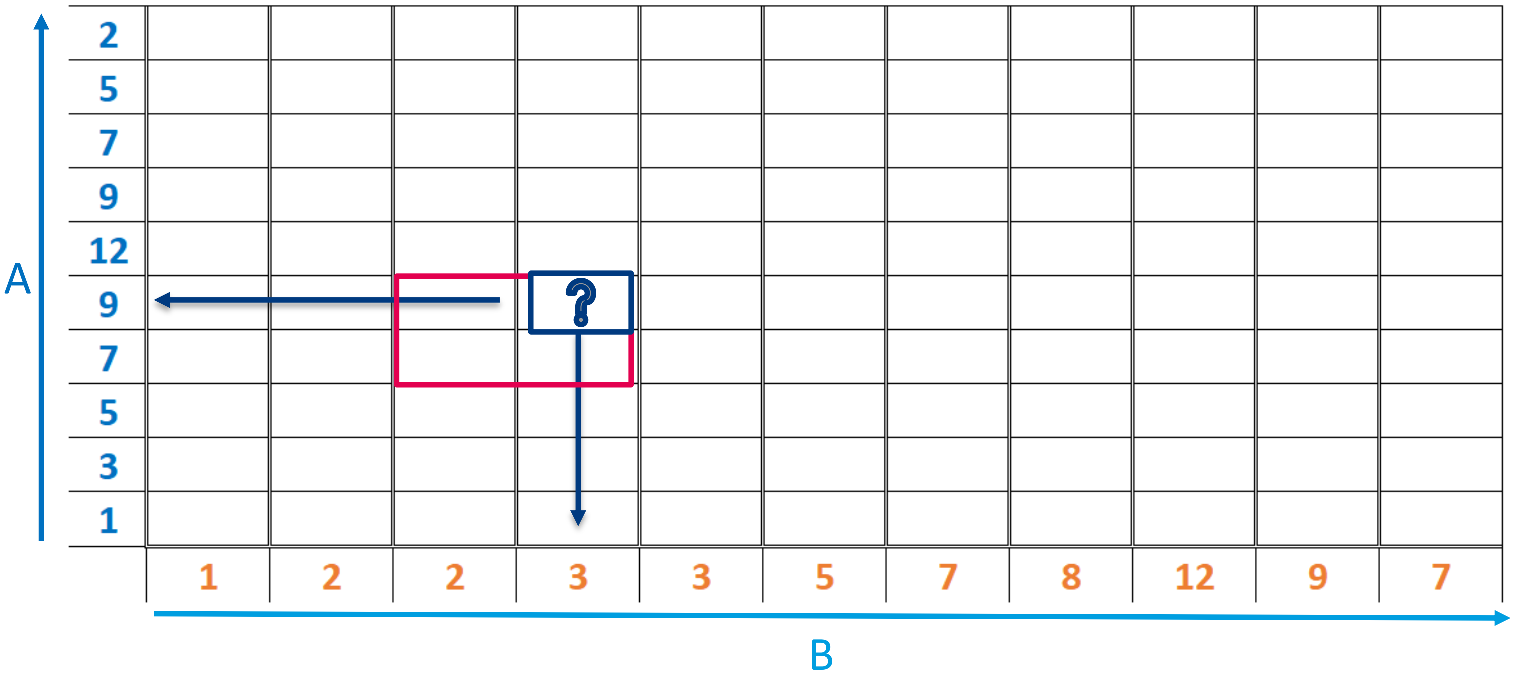
2											
5											
7											
9											
12											
9											
7											
5											
3											
1											
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = |A_i - B_j| + \min(D(i-1,j), D(i,j-1), D(i-1,j-1))$$

$D(i,j)$: distance between (A_1, A_2,…A_i) and (B_1, B_2, …, B_j)

Step1: fill out matrix



$$D(i,j) = \boxed{\|A_i - B_j\|} + \boxed{\min(D(i-1,j), D(i,j-1), D(i-1,j-1))}$$

Step1: fill out matrix

Calculate D(i,j) from bottom left to right top

A

2											
5											
7											
9											
12											
9											
7											
5											
3											
1	0										
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = \boxed{|A_i - B_j|} + \min(D(i-1,j), D(i,j-1), D(i-1,j-1))$$

$$D(1,1) = |A_1 - B_1| = 1 - 1 = 0$$

Step1: fill out matrix

A

2											
5											
7											
9											
12											
9											
7											
5											
3											
1	0										
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = \boxed{\left|A_i - B_j\right|} + \boxed{\min(D(i-1,j), D(i,j-1), D(i-1,j-1))}$$

$$D(1,2) = |A_1 - B_2| + \mathrm{D}(1,1) = |1-2| + 0 = 1$$

Step1: fill out matrix

A

2											
5											
7											
9											
12											
9											
7											
5											
3											
1	0	1	2	4	6	10	16	23	34		
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = \boxed{|A_i - B_j|} + \boxed{\min(D(i-1,j), D(i,j-1), D(i-1,j-1))}$$

$$D(1,10) = |A_1 - B_{10}| + \mathrm{D}(1,9) = |1-9| + 34 = 42$$

Step1: fill out matrix

A

2											
5											
7											
9											
12											
9											
7											
5											
3	2										
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = \left|A_i - B_j\right| + \min(D(i-1,j), D(i,j-1), D(i-1,j-1))$$

$$D(2,1) = |A_2 - B_1| + \mathrm{D}(1,1) = |3-1| + 0 = 2$$

Step1: fill out matrix

A											
2											
5											
7											
9											
12											
9											
7											
5											
3	2	1	1+1								
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = \boxed{|A_i - B_j|} + \boxed{\min(D(i-1,j), D(i,j-1), D(i-1,j-1))}$$

$$D(2,3) = |A_2 - B_3| + \min(\mathrm{D}(1,3), \mathrm{D}(2,2), \mathrm{D}(1,2)) = |3-2| + 1 = 2$$

Step1: fill out matrix

A

2	50	41	41	36	36	24	18	18	22	16	10
5	49	41	41	35	35	21	13	12	16	9	5
7	45	38	38	33	33	21	11	9	11	5	3
9	39	33	33	29	29	19	11	8	6	3	5
12	31	26	26	23	23	15	9	7	3	6	11
9	20	16	16	14	14	8	4	3	6	6	8
7	12	9	9	8	8	4	2	3	8	10	10
5	6	4	4	4	4	2	4	7	14	18	20
3	2	1	2	2	2	4	8	13	22	28	32
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = |A_i - B_j| + \min(D(i-1,j), D(i,j-1), D(i-1,j-1))$$

Step1: fill out matrix

A

2	50	41	41	36	36	24	18	18	22	16	10
5	49	41	41	35	35	21	13	12	16	9	5
7	45	38	38	33	33	21	11	9	11	5	3
9	39	33	33	29	29	19	11	8	6	3	5
12	31	26	26	23	23	15	9	7	3	6	11
9	20	16	16	14	14	8	4	3	6	6	8
7	12	9	9	8	8	4	2	3	8	10	10
5	6	4	4	4	4	2	4	7	14	18	20
3	2	1	2	2	2	4	8	13	22	28	32
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

$$D(i,j) = |A_i - B_j| + \min(D(i-1,j), D(i,j-1), D(i-1,j-1))$$

- The minimum distance between the complete sequences A and B is denoted in the right-top cell, which is 10.

Step 2: Warping path finding

A

2	50	41	41	36	36	24	18	18	22	16	10
5	49	41	41	35	35	21	13	12	16	9	5
7	45	38	38	33	33	21	11	9	11	5	3
9	39	33	33	29	29	19	11	8	6	3	5
12	31	26	26	23	23	15	9	7	3	6	11
9	20	16	16	14	14	8	4	3	6	6	8
7	12	9	9	8	8	4	2	3	8	10	10
5	6	4	4	4	4	2	4	7	14	18	20
3	2	1	2	2	2	4	8	13	22	28	32
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

Find a minimum cost path from right top to bottom left

Step 2: Warping path finding

A

2	50	41	41	36	36	24	18	18	22	16	10
5	49	41	41	35	35	21	13	12	16	9	5
7	45	38	38	33	33	21	11	9	11	5	3
9	39	33	33	29	29	19	11	8	6	3	5
12	31	26	26	23	23	15	9	7	3	6	11
9	20	16	16	14	14	8	4	3	6	6	8
7	12	9	9	8	8	4	2	3	8	10	10
5	6	4	4	4	4	2	4	7	14	18	20
3	2	1	2	2	2	4	8	13	22	28	32
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

Step 2: Warping path finding

A

2	50	41	41	36	36	24	18	18	22	16	**10**
5	49	41	41	35	35	21	13	12	16	9	**5**
7	45	38	38	33	33	21	11	9	11	5	**3**
9	39	33	33	29	29	19	11	8	6	3	5
12	31	26	26	23	23	15	9	7	3	6	11
9	20	16	16	14	14	8	4	3	6	6	8
7	12	9	9	8	8	4	2	3	8	10	10
5	6	4	4	4	4	2	4	7	14	18	20
3	2	1	2	2	2	4	8	13	22	28	32
1	0	1	2	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

Step 2: Warping path finding

A

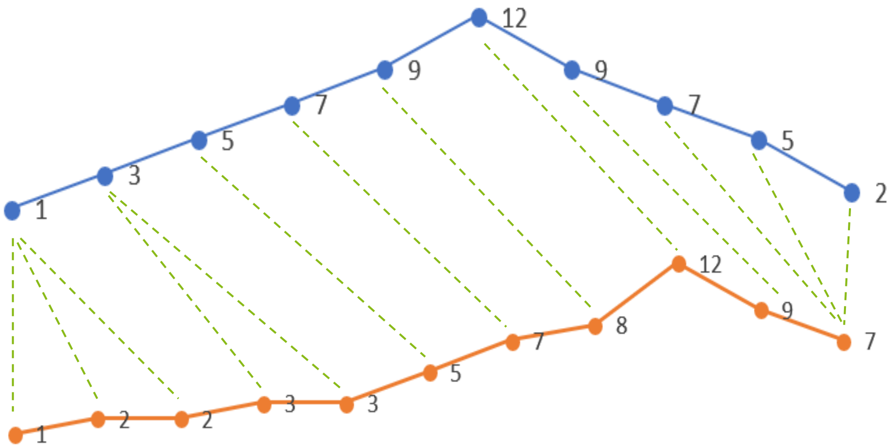
2	50	41	41	36	36	24	18	18	22	16	**10**
5	49	41	41	35	35	21	13	12	16	9	**5**
7	45	38	38	33	33	21	11	9	11	5	**3**
9	39	33	33	29	29	19	11	8	6	**3**	5
12	31	26	26	23	23	15	9	7	**3**	6	11
9	20	16	16	14	14	8	4	**3**	6	6	8
7	12	9	9	8	8	4	**2**	3	8	10	10
5	6	4	4	4	4	**2**	4	7	14	18	20
3	2	1	2	**2**	**2**	4	8	13	22	28	32
1	**0**	**1**	**2**	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

B

That’s it!

2	50	41	41	36	36	24	18	18	22	16	**10**
5	49	41	41	35	35	21	13	12	16	9	**5**
7	45	38	38	33	33	21	11	9	11	5	**3**
9	39	33	33	29	29	19	11	8	6	**3**	5
12	31	26	26	23	23	15	9	7	**3**	6	11
9	20	16	16	14	14	8	4	**3**	6	6	8
7	12	9	9	8	8	4	**2**	3	8	10	10
5	6	4	4	4	4	**2**	4	7	14	18	20
3	2	1	2	**2**	**2**	4	8	13	22	28	32
1	**0**	**1**	**2**	4	6	10	16	23	34	42	48
	1	2	2	3	3	5	7	8	12	9	7

Warping path
P=(p1, p2, …, p13) where,
p1=(1,1), p2=(2,1), p3=(3,1)
… … p13=(11,10)



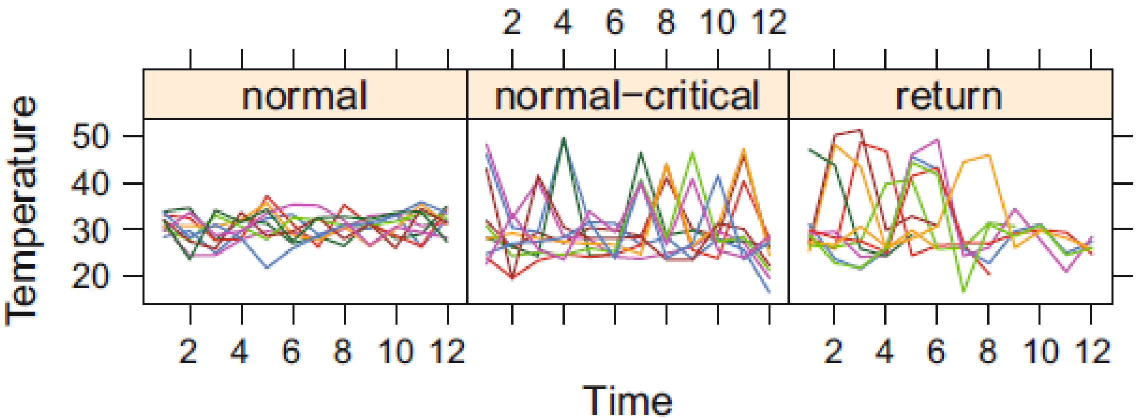
Classification and clustering using DTW

Example: Logistic Use Case

- Shipping pharmaceutical products:
  It describes the process of loading a vehicle at the origin and starting to move towards its destination. During the movement of the container, temperature is constantly monitored. If the temperature exceeds a certain threshold for some time, the vehicle has to move back to its origin. Otherwise, it continues to the destination where the containers are unloaded.
- Time warping starts with data representing time sequences for a number of process instances
- Analytical goals: segmentation (clustering) or classification of time sequences.

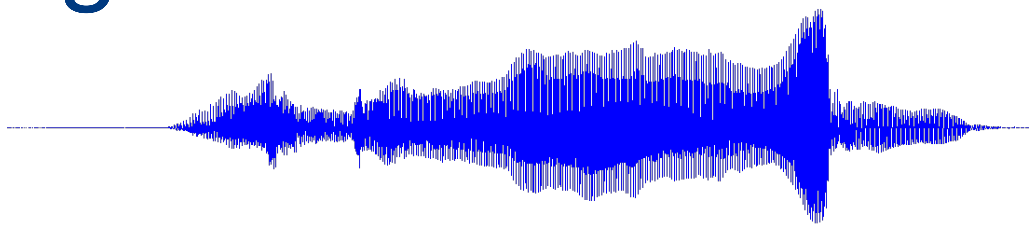
Clustering using DTW: example

- Use DTW to calculate the distance matrix between the 100 time sequences
- Apply standard clustering algorithms (k-means, hierarchical clustering) using DTW distances
- The clustering results show three different kinds of behavior:
  - normal temperature; critical temperature; return temperature

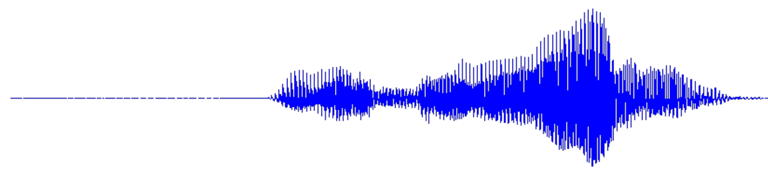


Other real world applications: Speech recognition

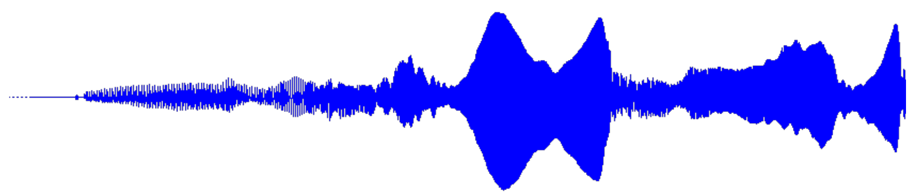
"Slow" Hello



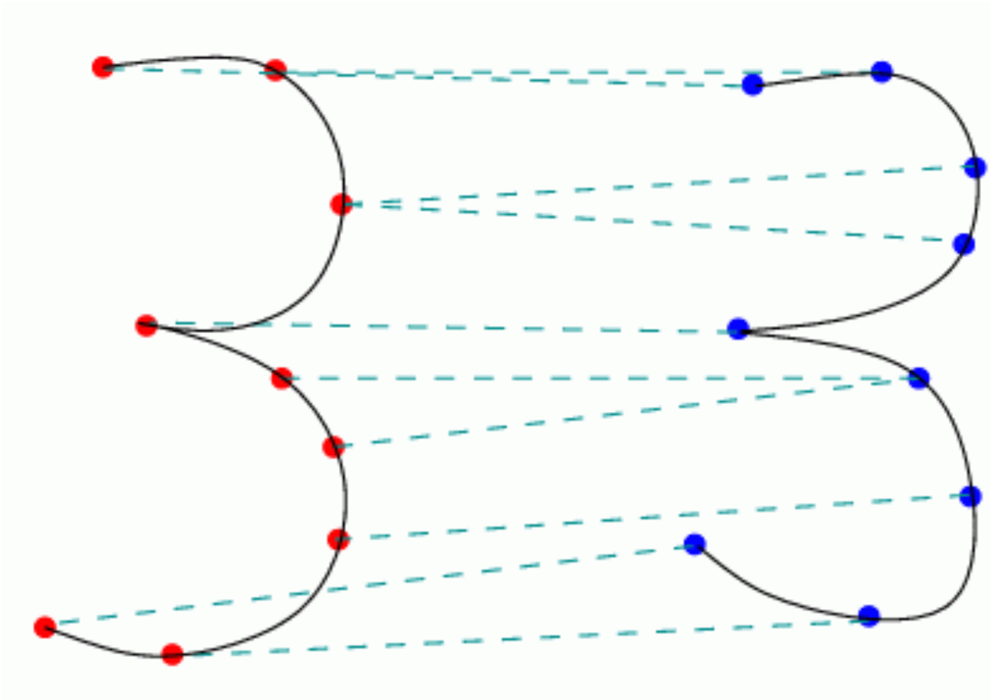
"Fast" Hello



Good Morning



Handwriting recognition



Gesture recognition

Temporal DM: Response Features

Response features

- Extract from the time sequence several time independent features
- Based on these extracted features, one can apply classification/regression/clustering methods

Response features

- Extract from the time sequence several time independent features
- Some examples of features:
  - Maximum and minimum of the time sequence
  - Temporal location of maximum and minimum
  - Breakpoints in the time sequence
  - Largest difference between two sequenced values
  - Length of the sequence
  - Area under the polygon defined by the sequence
  - …

Depending your analytics tasks and goals!

Response features

- Features Based on Frequency Distributions
  - mean, median, quantiles, or variances
- Features based on regression models
- Features based on change points

Response Feature Analysis

- Example: Pre-eclampsia

  Pre-eclampsia is a complication in pregnancy caused by multiple factors. In order to detect pre-eclampsia, weight, blood pressure and proteinuria (the condition of passing more than normal amounts of protein in the urine) of women are monitored during pregnancy.

- Binary classification task: pre-eclampsia (Hospital=true), and non-preeclampsia (Hospital=false)
- We extract features from the time sequences for weight, proteinuria, and blood pressure

Response Feature Analysis: Pre-eclampsia

Characteristic features may be based on properties of the frequency distribution of the time sequences, e.g. mean, median, quantiles, variances

- Figure shows boxplots for means and 0.95 quantiles of the proteinuria for persons in two Groups: Hospital = true; Hospital = false
- *It indicates that the 0.95 quantile is a promising response feature.*

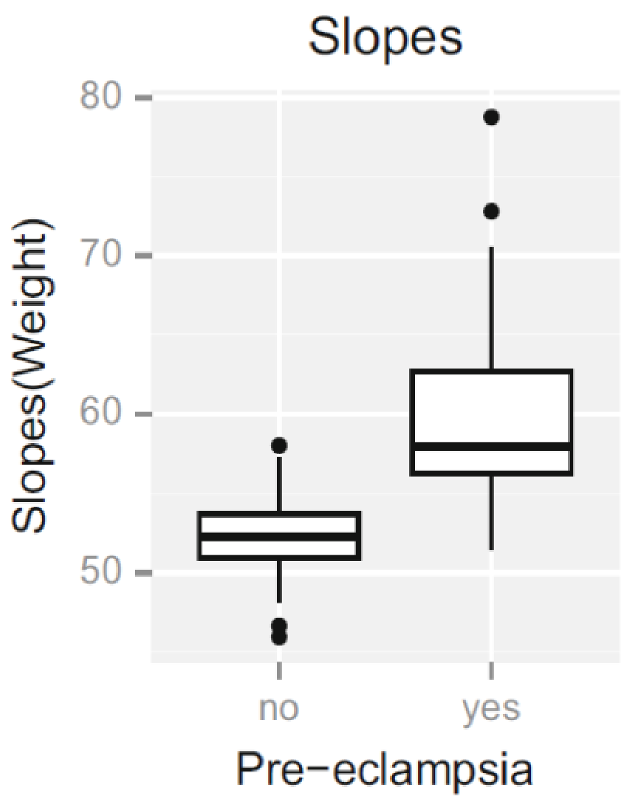
Response Feature Analysis: Pre-eclampsia

Feature extraction based on regression: applied for all variables.

- Example: a LR model for each woman's weight sequence

$$\mathtt{weight} = \beta_0 + \beta_1 \mathtt{day}$$

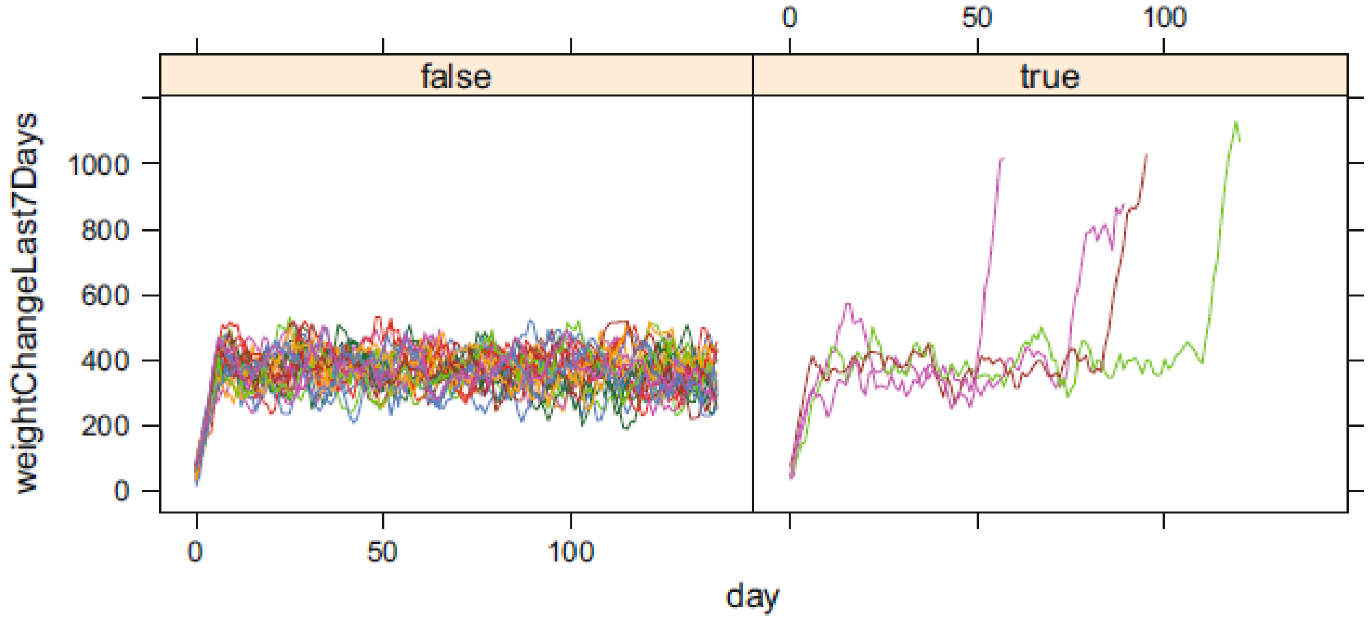
- The figure shows boxplot of slopes (β_1) in the two groups for the regression model applied for all time sequences
- It indicates that the slopes may be a reasonable response feature



Response Feature Analysis: Pre-eclampsia

Feature based on change points

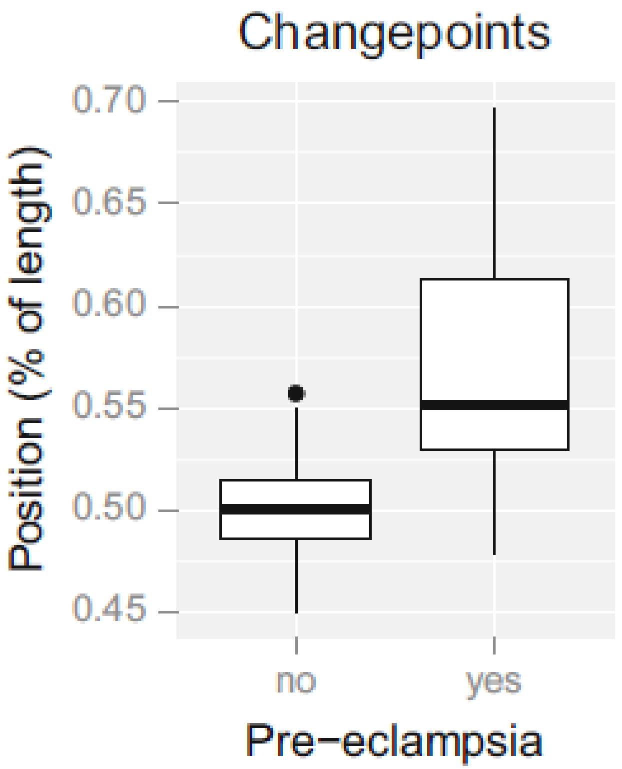
- Sometimes, one regression function for the entire time sequence is not an appropriate model
- The right figure leads to the conjecture that in the case of pre-eclampsia, there is a strong increase in weight change in the later phase of pregnancy. This change is not visible in the left figure



Response Feature Analysis: Pre-eclampsia

Feature based on change points

- Many methods for change points detection. One basic approach: learn a regression model for the time sequences, and calculate the residuals (error between predicted and actual values) of the regression model.
- Example: use the residuals of the sequences for weight.



After response features are constructed, standard classification algorithms can be applied

Summary: response features

- Extract from the time sequence several time independent features, based on
  – frequency distributions
  – regression models
  – change points

- After extracting features, one can apply classification/regression/clustering methods

Markov chains

Why Markov Chains?

- Many decisions need to consider uncertainty about a sequence of future events.
  - Uncertain demand for GM SUVs each month over the next year
  - Uncertain daily evolution of stock prices

Sequential Processes

- Suppose now we take a series of observations of that random variable, X0, X1, X2,…
- A stochastic/sequential process is an indexed collection of random variables {Xt}, where t is the index from a given set T.
  (The index t often denotes time.)

- Examples:
  Sales of an item, Xt : number of items sold on day t, t=1,2,…
  then the stochastic process { Xt } = {X0 , X1 ,X2 ,…..} provides a mathematical representation of how sales evolve starting today

- State evolution is typically random
- Used to model the probability of state sequences and to predict future states or events

Types of Stochastic Processes

- Several types of stochastic processes, based on how future values probabilistically depend on present and past values.
  - In general, future values may depend on the present value and all the past values (for example, stock prices may depend on past values)
  - On the other hand, future values may be completely independent of present and past values (fair coin tossing or fair die rolling).
  - In some cases, future values may be independent of past values and depend only on the present value (whether to eat?)

Example

- Let Xt be a random variable that takes value 0 if the weather is **dry** on day t and value 1 if the weather is **rainy** on day t.
- Then { Xt }={X0 , X1 ,X2 ,…..} provides a mathematical representation of how the weather evolves starting today (t=0), and the 'state' of the system is dry (0) or rainy (1).
- Suppose the probability that tomorrow is dry is 0.8 if today is dry, but is 0.6 if it rains today. We write:

  P(dry tomorrow | dry today) = 0.8 = $P(X_1=0 \mid X_0=0)$
  P(dry tomorrow | rainy today) = 0.6 = $P(X_1=0 \mid X_0=1)$

- Or for any day t: $P(X_{t+1}=0 \mid X_t=0) = 0.8$ and $P(X_{t+1}=0 \mid X_t=1) = 0.6$

Example

- Suppose we know that X0=0,X1=0,X2=1,X3=0 (dry, dry, rainy, dry).
- What is the probability that X4=0 (dry)?
  - Mathematically, what is $P(X_4=0 \mid X_3=0, X_2=1, X_1=0, X_0=0)$
  - We have $P(X_4=0 \mid X_3=0) = 0.8$

    We did not care about the values of X2, X1, X0

- Given today's weather and the weather in the past, the conditional probability of tomorrow's weather is independent of weather in the past and depends only on today's weather (*this is called the Markovian property*).

Markov Processes

- Markov chains assume the Markov property:

$$p(x_{t+1} \mid x_0, \ldots, x_t) = p(x_{t+1} \mid x_t)$$

"Conditioned on the present,
the future is independent from the past"

One-step Transition Probabilities

- The weather Markov chain

$p_{00} = P(X_{t+1} = 0 | X_t = 0) = 0.8$

$p_{10} = P(X_{t+1} = 0 | X_t = 1) = 0.6$

$p_{01} = P(X_{t+1} = 1 | X_t = 0) = 1 - P(X_{t+1} = 0 | X_t = 0) = 0.2$

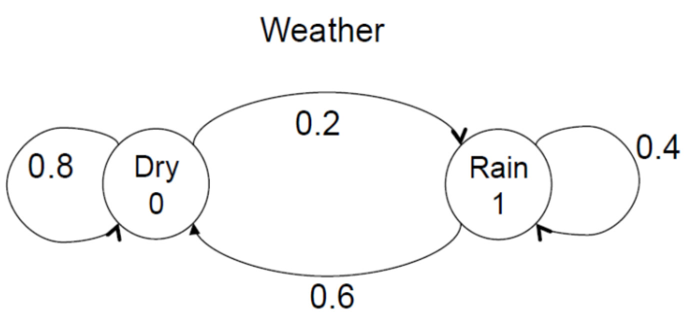
$p_{11} = P(X_{t+1} = 1 | X_t = 1) = 1 - P(X_{t+1} = 0 | X_t = 1) = 0.4$

- Arrange the four one-step transition probabilities in an one-step transition matrix whose rows and columns correspond to states and entries are $p_{ij} = P(X_{t+1} = j | X_t = i)$

State	0	1
0	$p_{00} = 0.8$	$p_{01} = 0.2$
1	$p_{10} = 0.6$	$p_{11} = 0.4$

State Transition Matrices

- A Markov chain with stationary transition probabilities can be illustrated using a state transition diagram



$$\begin{array}{cc} & \begin{array}{cc} 0 & 1 \end{array} \\ \begin{array}{cc} Dry & 0 \\ Rain & 1 \end{array} & \begin{bmatrix} 0.8 & 0.2 \\ 0.6 & 0.4 \end{bmatrix} \end{array}$$

- Constraints on valid transition matrices:
  - $q_{ij} \geq 0$
  - $\sum_{j=1}^{N} q_{ij} = 1$, for all i

Estimation of transition probabilities

state
1
1
2
3
3
1
2
3
3
3

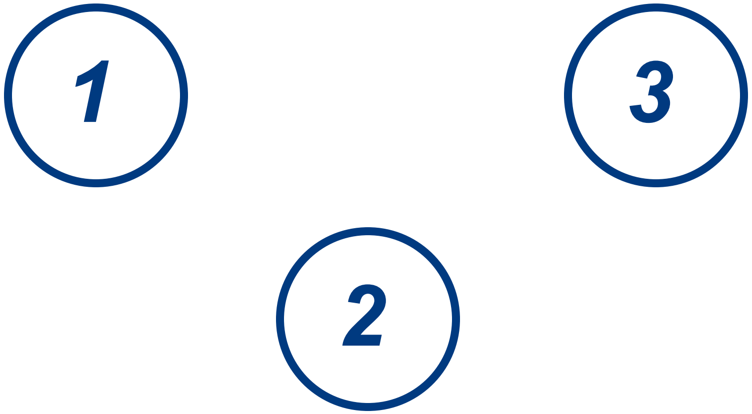
Maximum likelihood estimation:

$$\hat{p}_{ij} = \frac{n_{ij}}{n_i}$$

nij: observed number of one-step transitions from state i to j
ni: observed number of occurrences from state i

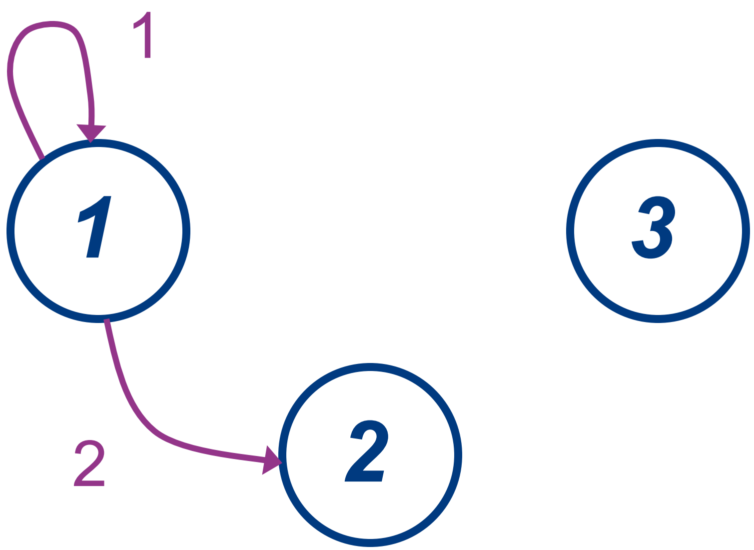
Estimation of transition probabilities

state
1
1
2
3
3
1
2
3
3
3



Estimation of transition probabilities

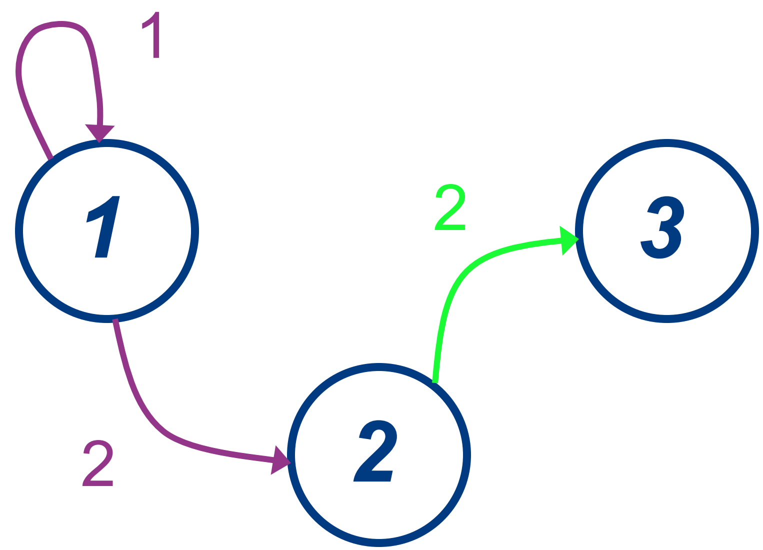
state
1
1
2
3
3
1
2
3
3
3



Count transition occurrences

Estimation of transition probabilities

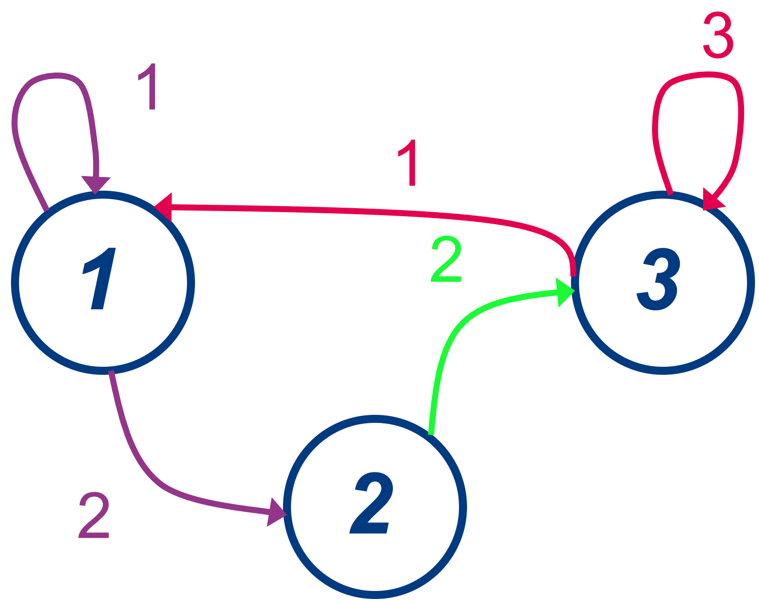
state
1
1
2
3
3
1
2
3
3
3



Count transition occurrences

Estimation of transition probabilities

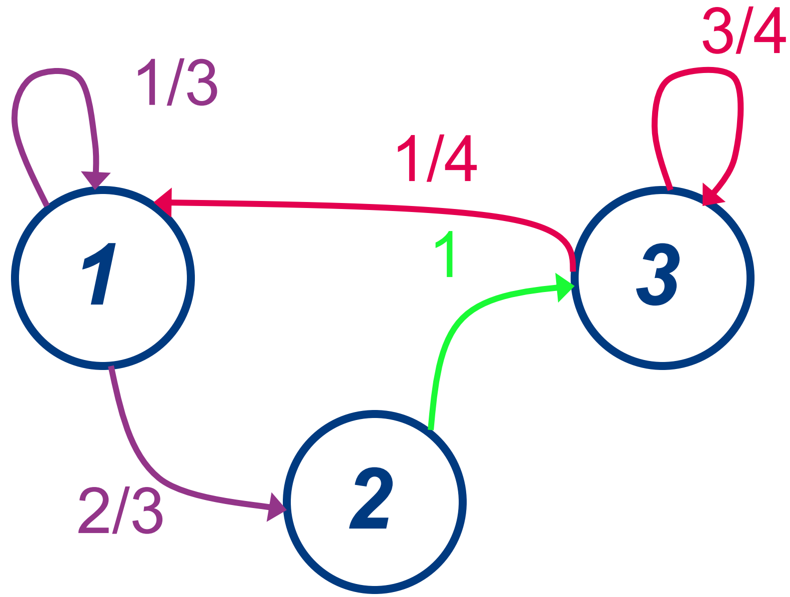
state
1
1
2
3
3
1
2
3
3
3



Count transition occurrences

Estimation of transition probabilities

state
1
1
2
3
3
1
2
3
3
3



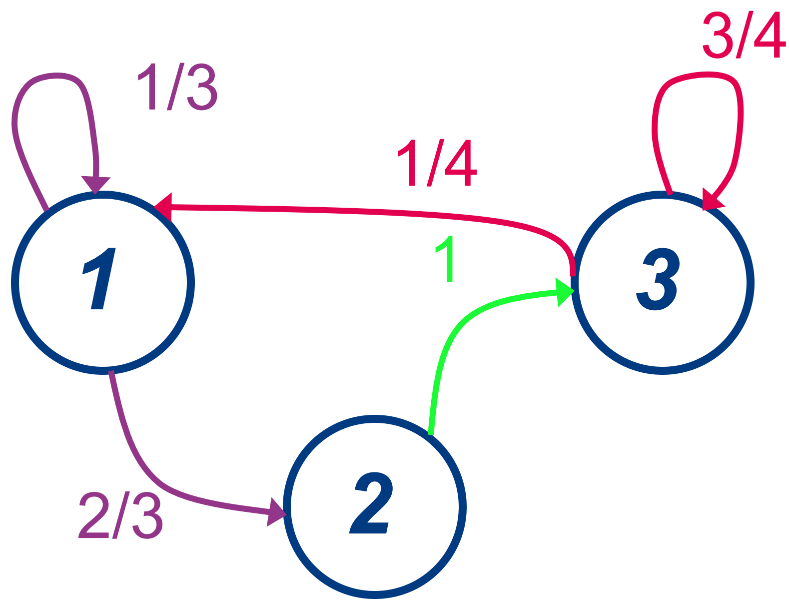
Normalize transition counts to sum up to 1.0

Predicting new state sequences

Prob(1123) =

P(1) * P(1→1) * P(1→2) * P(2 →3) =

?



$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1})$$

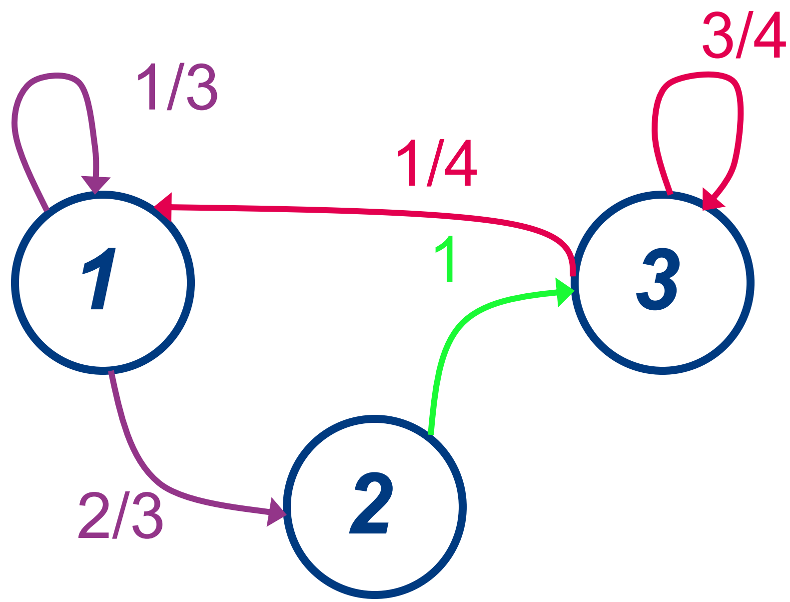
Predicting new state sequences

state
1
1
2
3
3
1
2
3
3
3

Prob(1123) =

P(1) * P(1→1) *
P(1→2) * P(2 →3) =

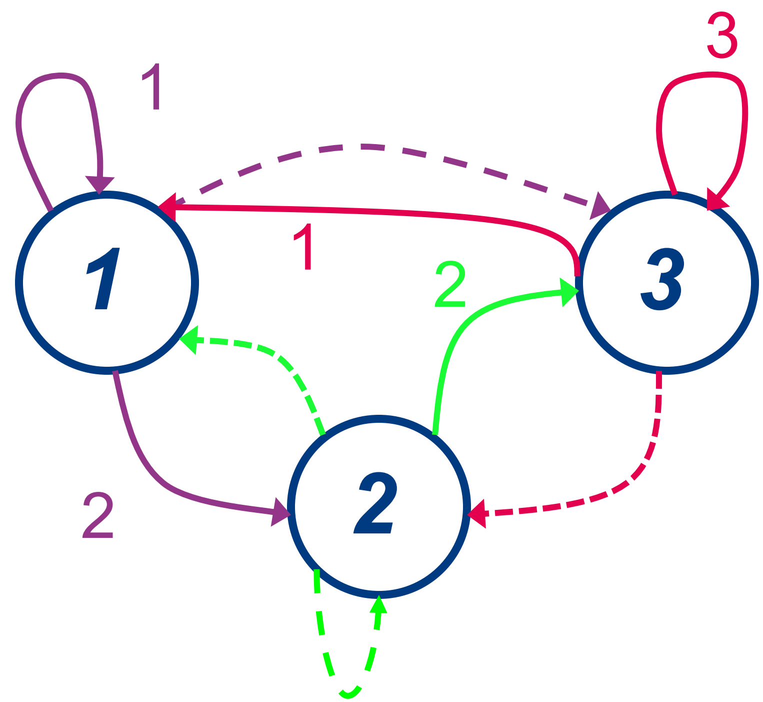
3/10 * 1/3 * 2/3 * 1
= 6/90



$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1})$$

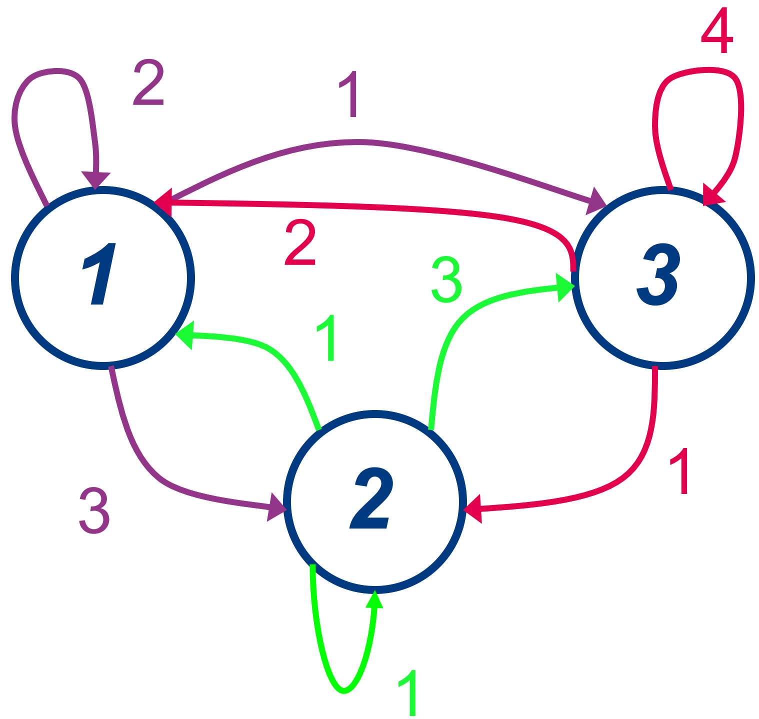
Laplace smoothing

- What if no occurrence for some transitions?
- Add a count of 1 to every possible state transition before normalization



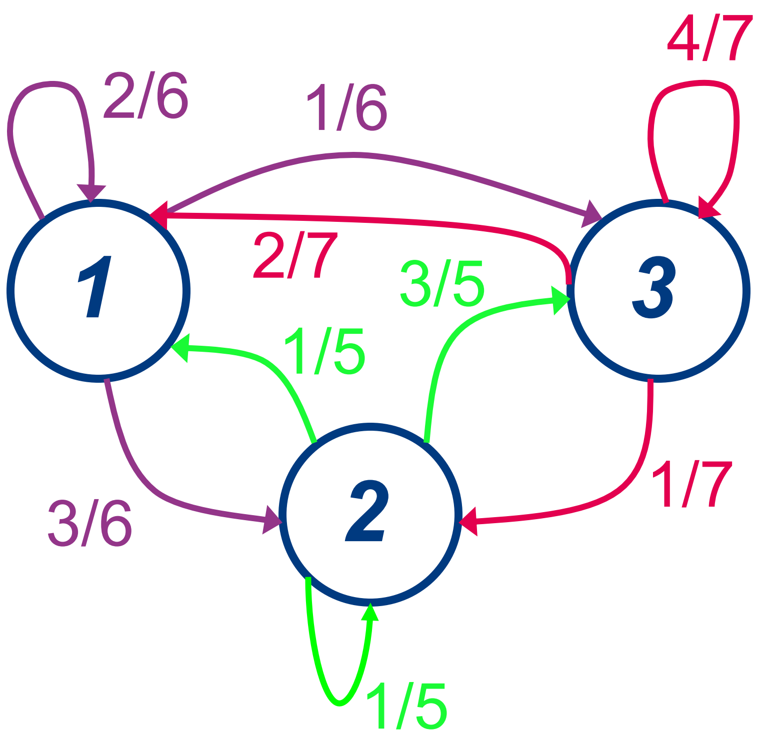
Laplace smoothing

- Add a count of 1 to every possible state transition before normalization



Laplace smoothing

- Add a count of 1 to every possible state transition before normalization

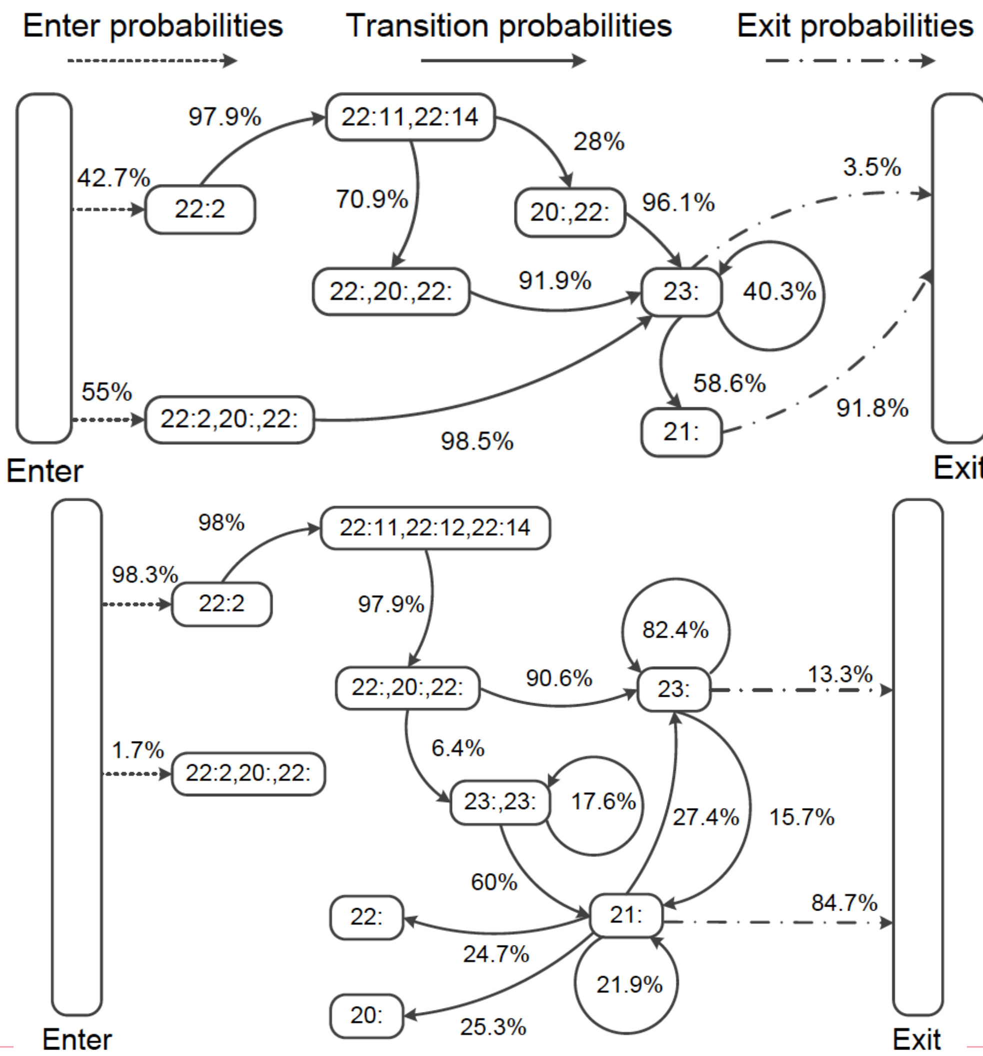


Markov chains

- Intuitive representation of sequence probabilities
  - Structure is informative
- Easy to estimate and use
  - Large bias, low variance
- Huge amount of applications in physics, statistics, economics, …

Example Markov chains

- For Twitter:
- For Dropbox:



Association analysis

Association analysis

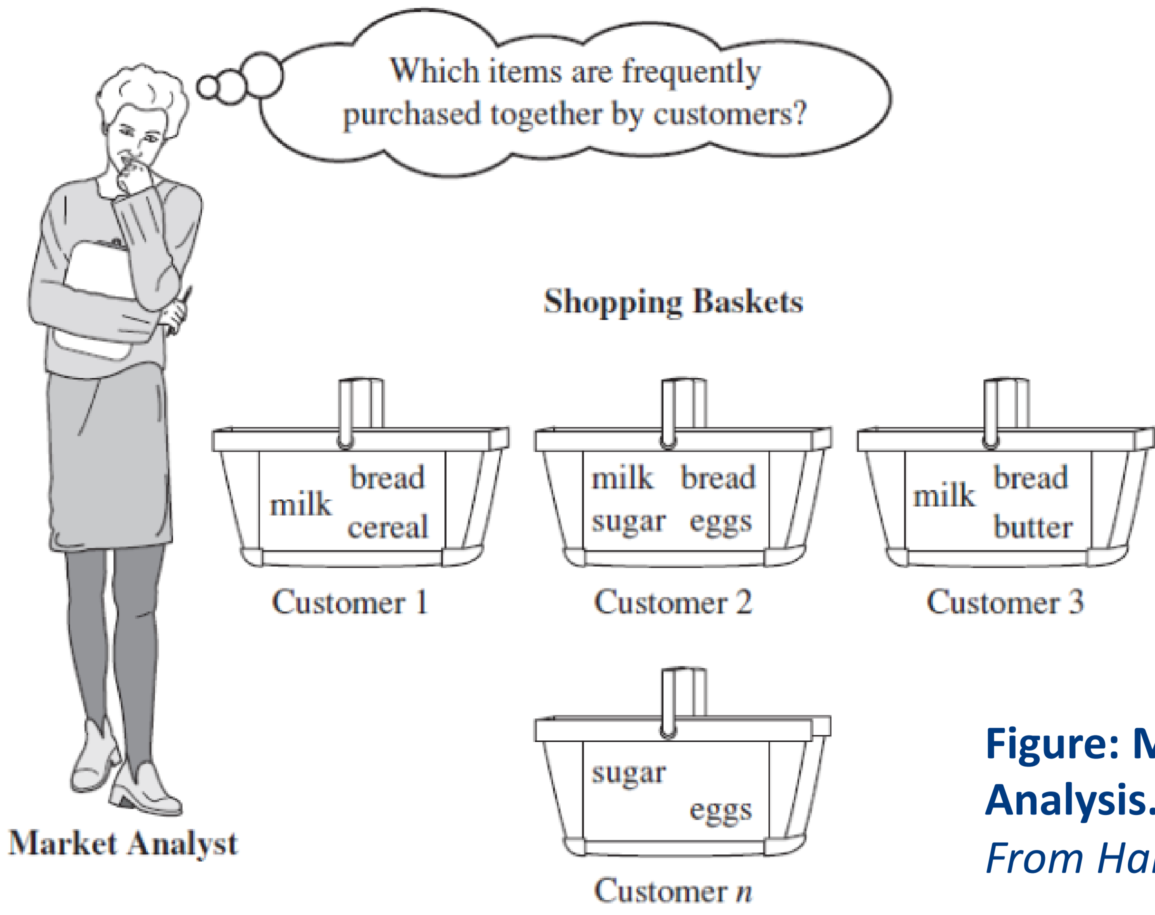
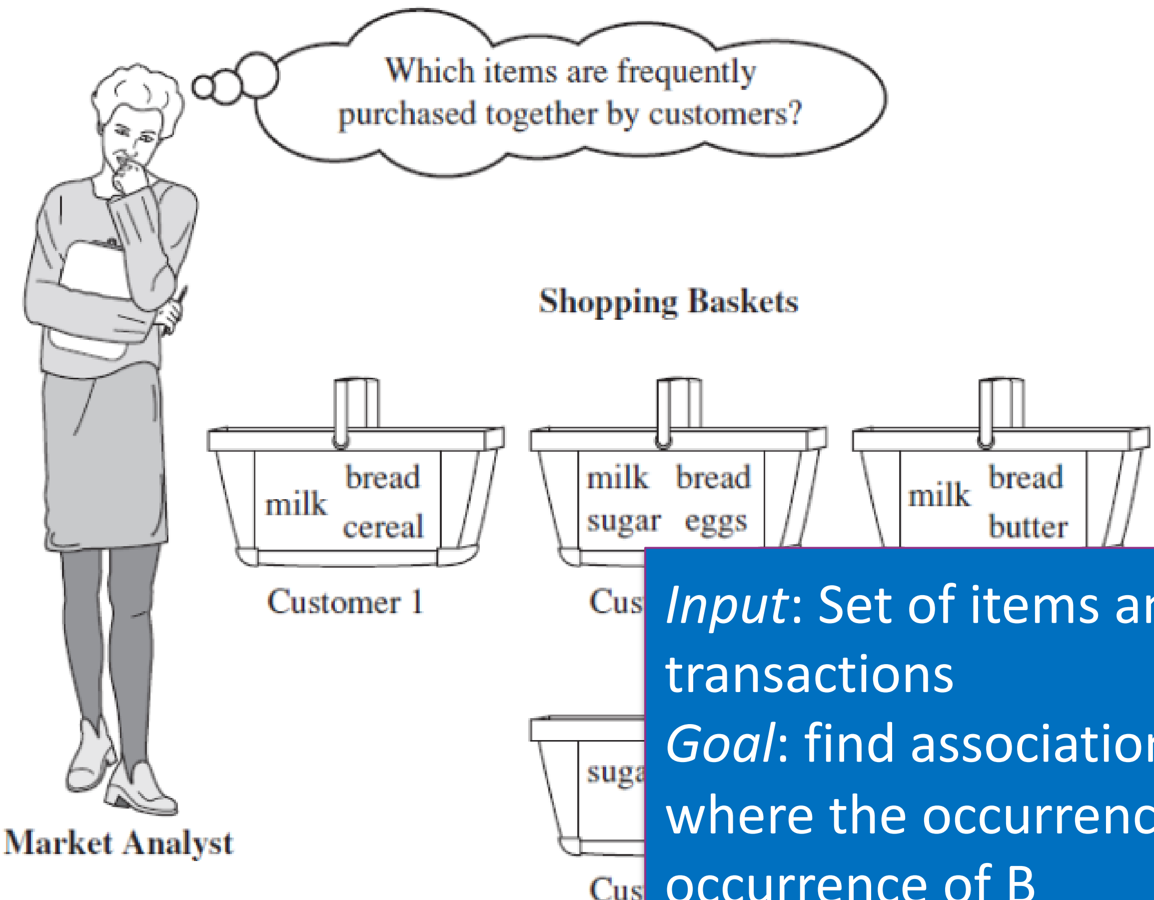


Figure: Market Basket Analysis.
From Han et al.

TU/e Technische Universiteit Eindhoven University of Technology

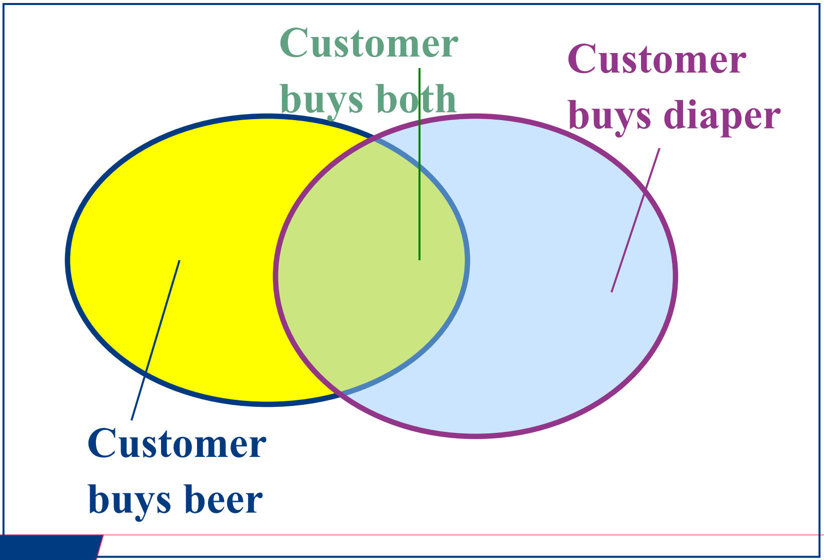
Association analysis



Input: Set of items and a set of transactions
Goal: find association rules $A\text{->}B$ where the occurrence of A implies the occurrence of B

Association analysis

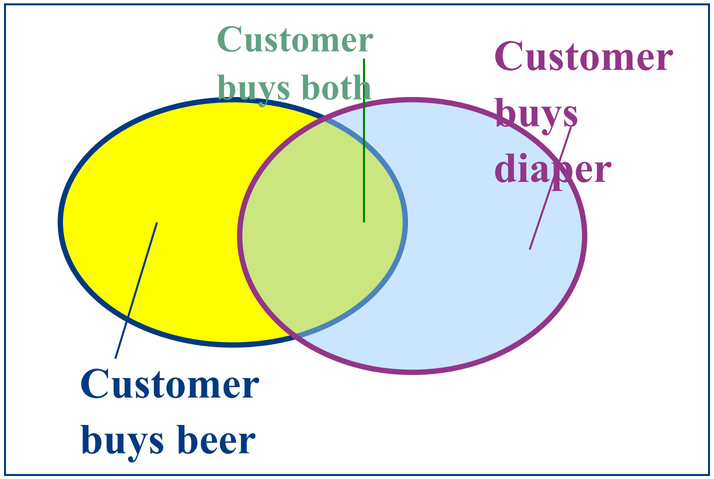
Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk



- Itemset X: A set of one or more items
- k-itemset $X = \{x_1, \ldots, x_k\}$
- *(absolute) support*, or, *support count* of X: Frequency or occurrence of an itemset X
- *(relative) support*, *s*, is the fraction of transactions that contains X (i.e., the probability that a transaction contains X)
- An itemset X is *frequent* if X's support is no less than a *minsup* threshold

Association analysis

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

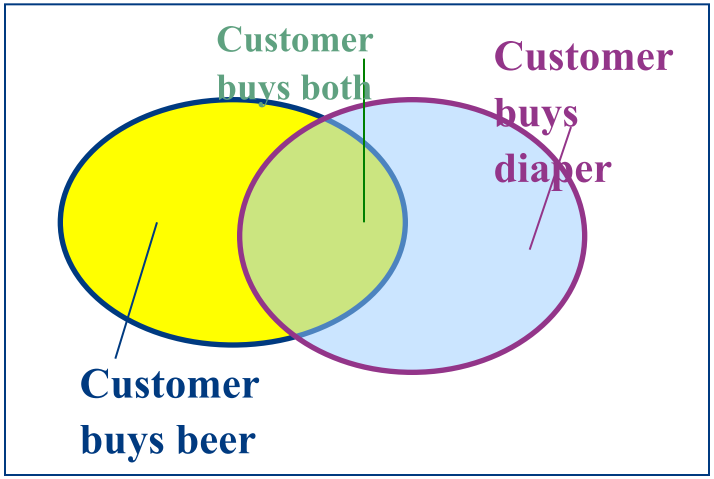


Find all the rules $X \rightarrow Y$ with minimum support and confidence

- support, $s(X \rightarrow Y)$, probability that a transaction contains $X \cup Y$
- confidence, $c(X \rightarrow Y)$, conditional probability that a transaction having X also contains Y

Association analysis

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk



Find all the rules *X* → *Y* with minimum support and confidence

- support *s*, probability that a transaction contains X ∪ Y (X AND Y)
- confidence, *c,* conditional probability that a transaction having X also contains *Y*

- s(Beer -> Diaper) = 3/5=0.6
- c(Beer -> Diaper) = 3/3=1

Association analysis

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

1. Find large or frequent item sets:

- Define minimal support
- Find all item sets, for which their support is >= the threshold

Example: *let minsup = 50%*

1. Large/frequent sets:

Beer:3, Nuts:3, Diaper:4, Eggs:3, {Beer, Diaper}:3

Association analysis

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

2. Discover rules within large item sets:

- Define minimal confidence
- Determine all possible rules in the large item sets that exceed confidence

Example: *let minsup = 50%, minconf = 50%*

2. Association rules:

Beer → *Diaper* (60%, 100%)

Diaper → *Beer* (60%, 75%)

and many more!

Summary

- A lot of data are temporal
- Finding series or patterns in such data can be very interesting for analysis and predictions

… but also challenging!

Recurrent Neural Networks (RNNs) *as an approach to sequence modeling problems -> next week!*