Electronic Health Records and Temporal Abstractions

Stockholms universitet

Panagiotis Papapetrou, PhD
Professor, Stockholm University
Adjunct Professor, Aalto University

Who are we?

DSV @ Stockholm University

- DSV: Data- och SystemVetenskap (Computer and Systems Sciences)
- # of students: approx. 5400
- # of staff members: 176 (60 profs. /associate profs. / lecturers)

Stockholms universitet

Research at DSV – data science group

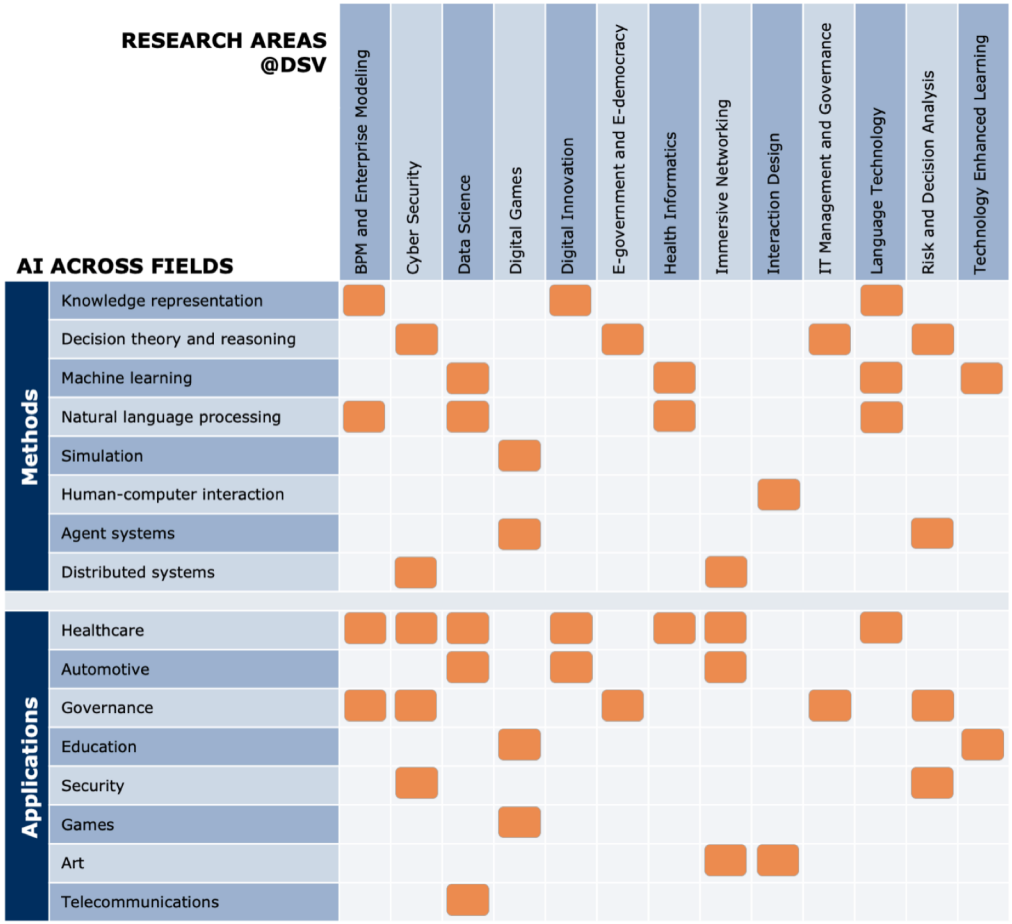
Main research areas:

- Sequential and temporal data mining
- Interpretability and explainability of machine learning methods
- Random forests and ensemble learning
- Machine learning for healthcare applications
- Clinical text mining and natural language processing

Current projects:

- **TempoMiner (2017-2020):** temporal mining for detecting ADEs in healthcare
- **CorIL (2017-2019):** discovering patient trajectories for heart failure treatment
- **EXTREME (2019-2020):** ethical AI for healthcare
- **Tropical (2019-2020):** network traffic prediction in 5G networks

Our AI research arena @ DSV



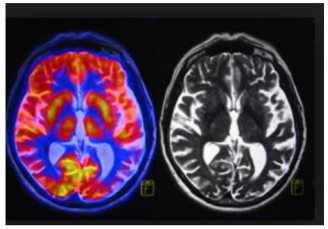
Part II - Outline

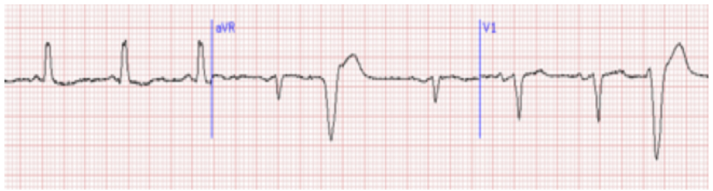
- **Definition** and **examples** of EHRs and EHR systems
- **Overview** of the usage of EHRs globally and in Sweden
- **Temporal abstractions** of EHR variables
- **Predictive models** on EHR data
- Dealing with **sparsity** in EHR data

Electronic Health Records: content

Longitudinal collection of electronic health information about individual patients and populations

- **Diagnoses** **I25.110**
- **Drug prescriptions** **A01AD05**
- **Clinical tests**
- **More complex structures**
  - clinical notes
  - medical images
  - MRIs
  - ECGs
  - …





Electronic Health Records

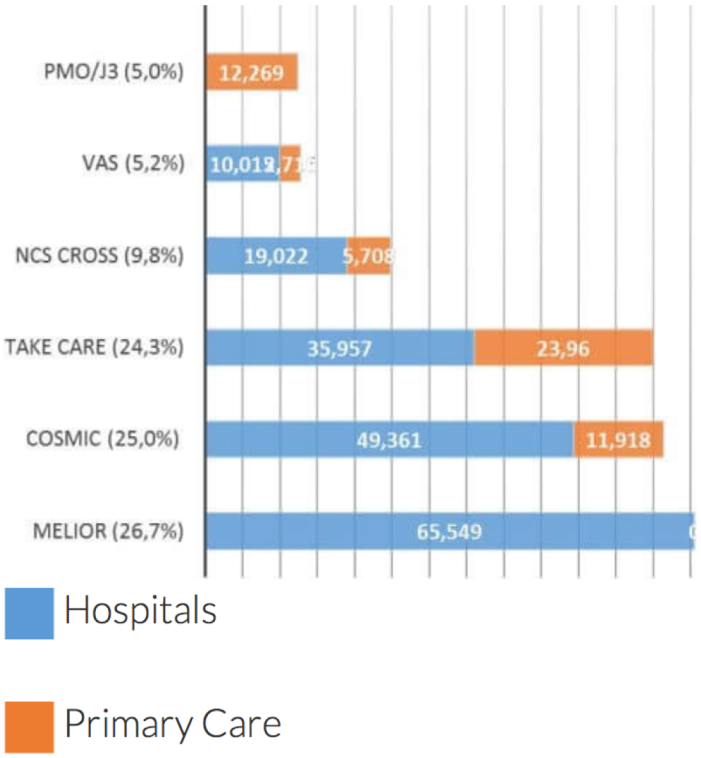
- **Examples of EHRs**
  - Australia: PCEHR initiative
  - Austria: EHR-Act
  - Canada: Interoperability
  - Estonia: first Country to implement a nation-wide EHR launched in 2008
  - Sweden: the Stockholm ERP corpus, the VAL databases, the TakeCare database
  - USA: various systems exist; less than 10% is used

Five players

- Five major vendors for the whole Swedish territory (accounting 96% of the market share) among which: Melior (Siemens, USA), Cosmic (Cambio, Sweden), TakeCare (CompGroup, Germany)

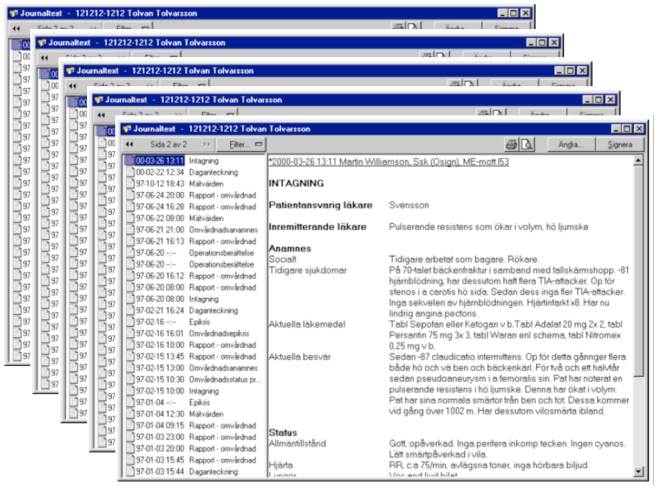
- Regional systems are not interconnected!

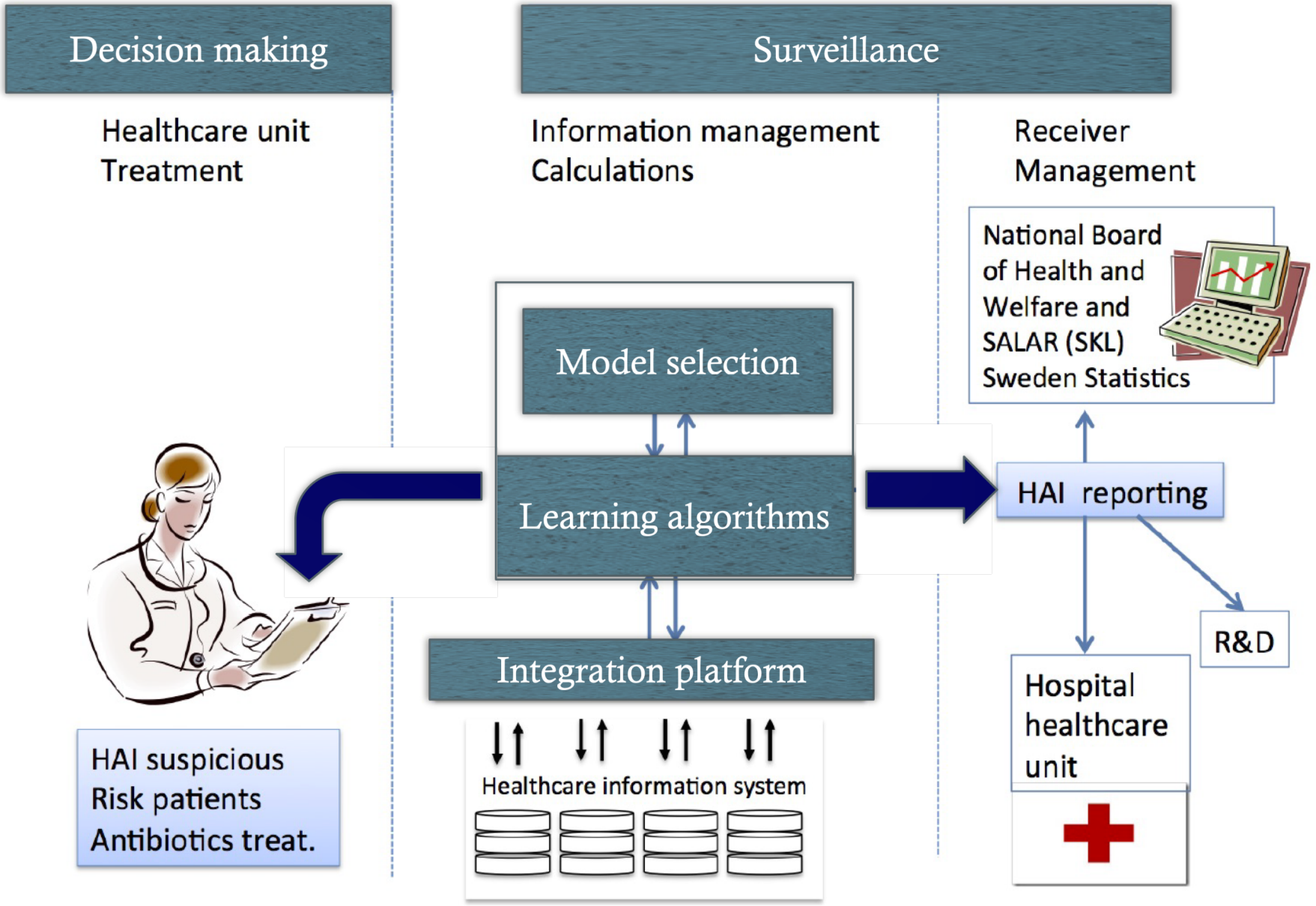
- E.g., the Cosmic EHR system which is adopted in 8 different counties, lacks interconnectivity since each system for each county has its own configuration making it 8 different systems



HealthBank @ Stockholm University

- More than 1.4 million in-patients
- Years: 2007-2014
- From Karolinska University Hospital
- De-identified but still sensitive
- 500 clinics/units
- Incl. oncology, gynecology, emergency, etc
- Approx. **160 million hospital events** between 2007-2014




Decision making
Surveillance
Healthcare unit
Treatment
Information management
Calculations
Receiver
Management
National Board of Health and Welfare and SALAR (SKL) Sweden Statistics
Model selection
Learning algorithms
HAI reporting
R&D
Integration platform
Hospital healthcare unit
Healthcare information system
HAI suspicious
Risk patients
Antibiotics treat.

ICD10* codes

- 10th revision of the International Classification of Diseases and Related Health Problems
- a classification system that is used to record medical activity
- the system enables classification and quantification of diseases and other health-related issues

3 -7 characters long

* http://www.ahima.org/icd10

ICD10 codes: examples

Chapter	Code Range	Estimated # of Codes	Description
1	A00-B99	1,056	Certain infectious and parasitic diseases
2	C00-D49	1,620	Neoplasms
3	D50-D89	238	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism
4	E00-E89	675	Endocrine, nutritional and metabolic diseases
5	F01-F99	724	Mental, Behavioral and Neurodevelopmental disorders
6	G00-G99	591	Diseases of the nervous system
7	H00-H59	2,452	Diseases of the eye and adnexa
8	H60-H95	642	Diseases of the ear and mastoid process
9	I00-I99	1,254	Diseases of the circulatory system
10	J00-J99	336	Diseases of the respiratory system
11	K00-K95	706	Diseases of the digestive system

ICD10 codes: examples

Code	Description
Combination Codes	
I25.110	Atherosclerotic heart disease of native coronary artery with unstable angina pectoris
Increased Specificity	
S72.044G	Non-displaced fracture of base of neck of right femur, subsequent encounter for closed fracture with delayed healing
Laterality	
C50.511	Malignant neoplasm of lower-outer quadrant of right female breast
C50.512	Malignant neoplasm of lower-outer quadrant of left female breast
"X" Placeholder	
H40.11X2	Primary open-angle glaucoma, moderate stage

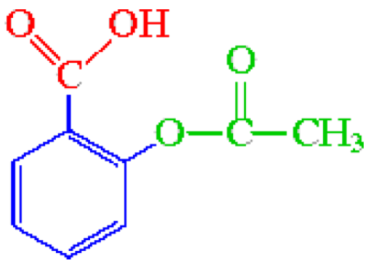
ATC* codes

- Anatomical Therapeutic Chemical codes, first published in 1976
- Used for classification of active ingredients of drugs
- Based on the organ/system on which they act
  - therapeutic
  - pharmacological and chemical properties
- Controlled by the World Health Organization Collaborating Centre (WHOCC) for drug statistics methodology

* http://www.whocc.no/atc_ddd_index/

ATC codes: a hierarchical structure

- Divides drugs into different groups according to the organ or system on which they act and/or their therapeutic and chemical characteristics
- Each bottom-level ATC code stands for a pharmaceutically used substance, or a combination of substances, in a single indication (or use)
  - ✓ one drug can have more than one codes
  - ✓ different brands share the same code, if they have the same active substance and indications



acetylsalicylic acid (aspirin)

- **A01AD05** as a drug for local oral treatment
- **N02BA01** as an analgesic and antipyretic
- **B01AC06** as a platelet inhibitor

ATC codes – example: A10BA02

ATC codes classify drugs into 5 different levels

Level	Content	Type	Example
I	anatomical main group	1 letter	A: alimentary tract and metabolism
II	therapeutic subgroup	2 digits	A10: diabetes drugs
III	pharmacological subgroup	1 letter	A10B: blood glucose lowering drugs, excl. insulins
IV	chemical subgroup	1 letter	A10BA: biguanides
V	chemical substance	2 digits	A10BA02: metformin

Adverse Drug Events (ADEs)

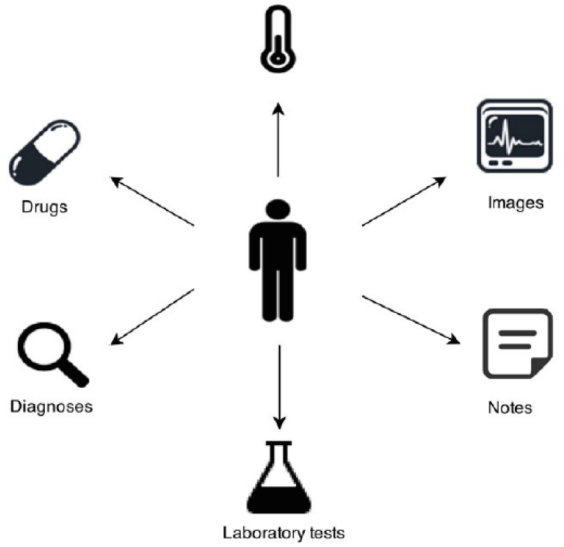
- Many ADEs are not being identified as such, due to limited knowledge about the effects of medical treatments, e.g., drugs being tested only in limited clinical trials under controlled conditions

Alternative: resort to machine learning methods and explore different feature abstractions: static or temporal

Learning classification models: extremely useful for patient monitoring, outcome prediction, and decision support

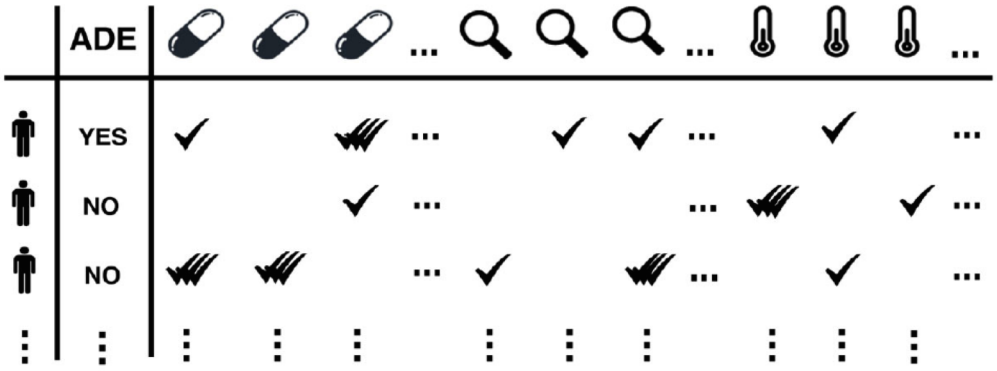
Extracting features from EHRs

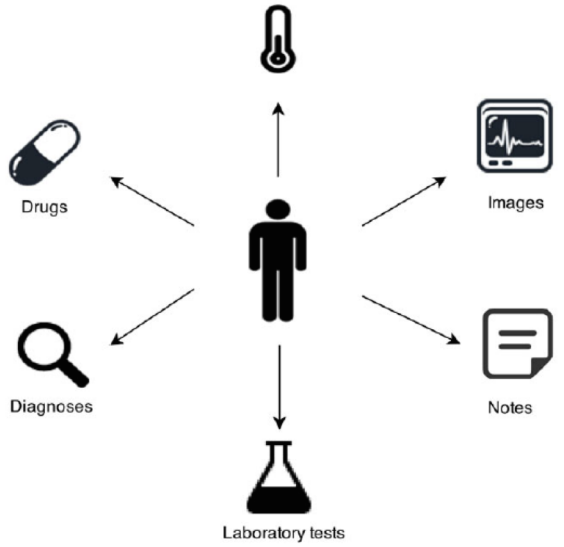
- **Mainly two lines of approaches:**
  - static features
  - temporal features



Static features

- **Mainly two lines of approaches:**
  - static features
  - temporal features



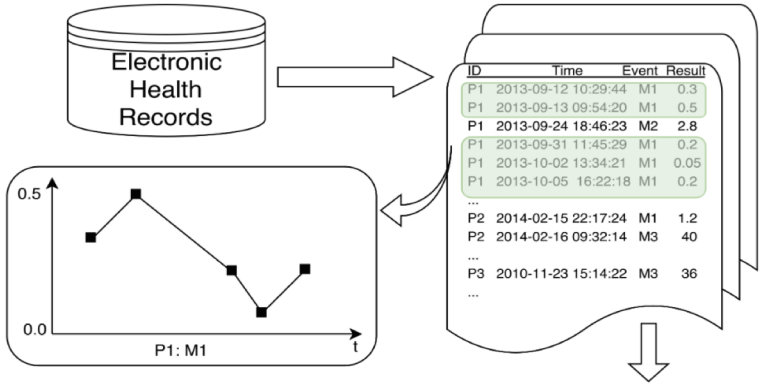


The class labels assigned depending on task at hand, e.g., ADE detection

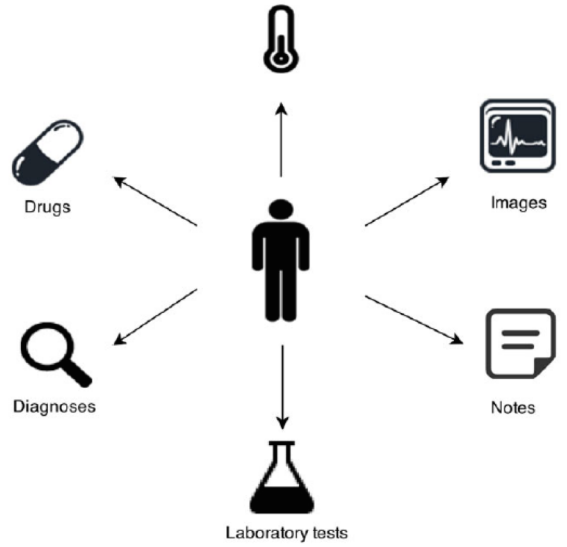
- **Existing out-of-the-box classifiers are used**
  - **Decision trees, random forests, SVMs, deep learning architectures** [Chazard2011, Zhao2013, Karlsson2013, Shickel2018]

Temporal features

- **Mainly two lines of approaches:**
  - static features
  - temporal features



ID	C	M1	M2	M3	...
P1	1			NA	...
P2	0				...
P3	1		NA		...
...	...	...	...	...	...



Clinical measurements:

- different units
- times of measurement
- sparsity

Time plays a major role in Medical Information Systems

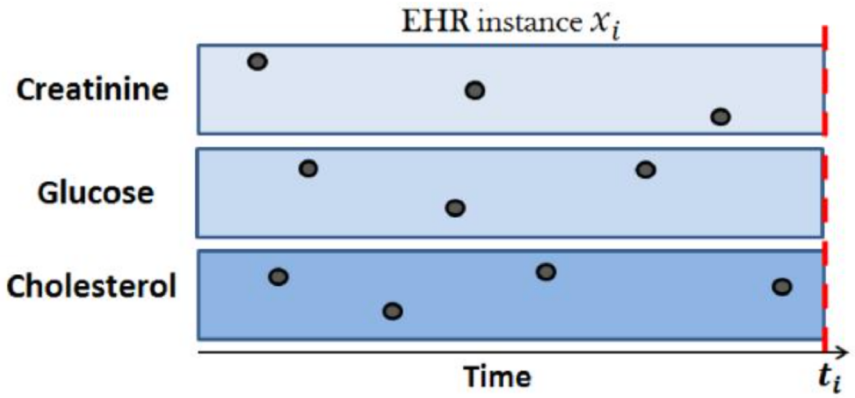
- Events occur at some time point(s)
- Certain facts hold during a time period
- Temporal relationships exist between facts and/or events

Abstracting time away means that dynamic situations are converted to static (snap-shot) situations, where neither the evolution of disorders, nor patient states can be modeled

Such abstractions should be carefully parametrized!

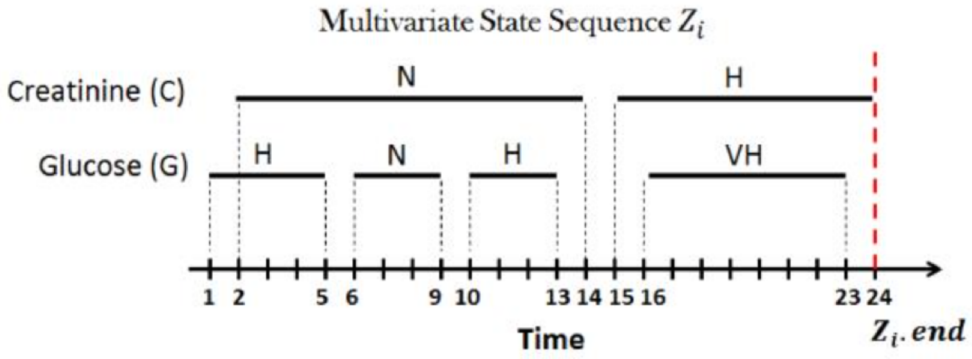
Temporal abstractions of EHRs

- Database: **EHRs of Patients**
- **Each EHR:**
  - Multiple temporal variables registered and evolving concurrently
  - Each variable with multiple readings until a critical time point t_i, e.g., glucose, creatinine, cholesterol
  - **Class label:** Disease/symptom detected at time t_i (event of interest)



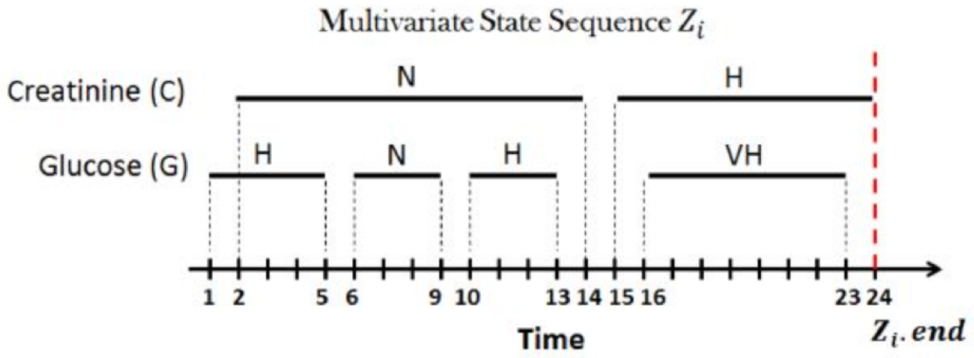
Temporal abstractions of EHRs

- Database: **EHRs of Patients**
- **Each EHR:**
  - Multiple temporal variables registered and evolving concurrently
  - Each variable with multiple readings until a critical time point t_i, e.g., glucose, creatinine, cholesterol
  - **Class label:** Disease/symptom detected at time t_i (event of interest)



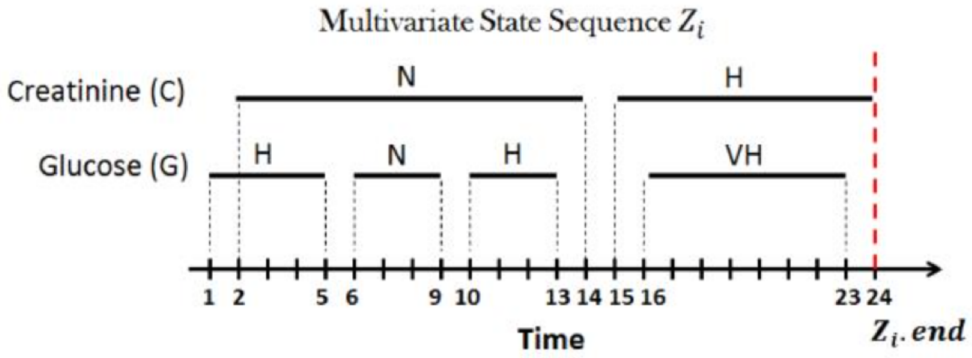
Two types of temporal abstractions

- **Trend abstraction:**
  - e.g., *decreasing*, *steady*, *increasing*
  - time series segmentation + identify slopes [Keogh2003]
- **Value abstraction**
  - e.g., *very low*, *low*, *normal*, *high*, *very high*
  - use 10^{th}, 25^{th}, 75^{th}, and 90^{th} percentiles on the lab values to define [Batal2012]



Temporal abstractions of EHRs

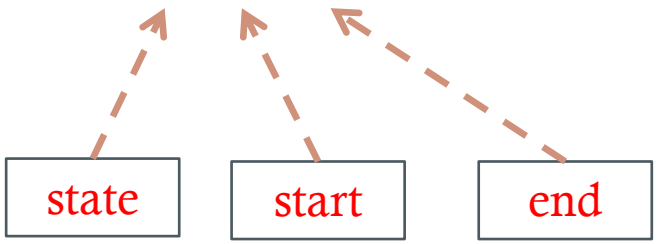
- **Supervised temporal prediction [Batal2012, Rebane2019]**
  - Given a labeled dataset of temporal instances up to time t_i
  - Find frequently occurring "temporal patterns" for each label
  - *Given a sample instance → predict its label*



Temporal abstractions: definition

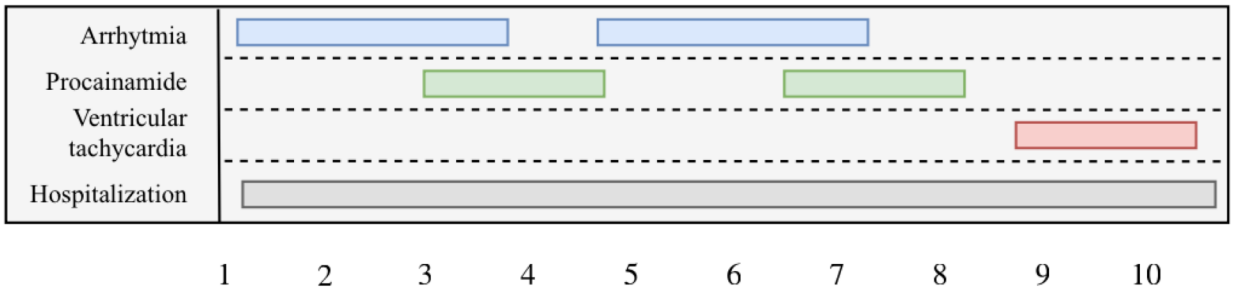
- **Defining a temporal abstraction**
  - Numeric/trend values → finite ***abstraction alphabets*** → define ***states***
  - All contiguous values with the same abstraction form an ***interval***
  - **Time series variables in EHRs can be represented as**

$$\{ \langle v_1, s_1, e_1 \rangle, \quad \langle v_2, s_2, e_2 \rangle, \quad \ldots \}$$



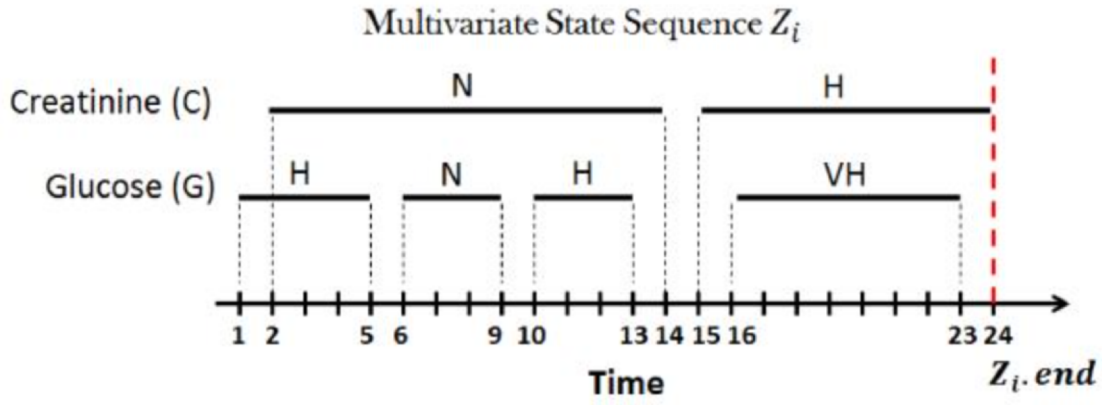
Temporal abstractions: definition

- **Three concepts:**
  - **F:** temporal variable
  - **V:** state
  - **E (F, V, s, e):** state interval
- **Hence:**
  - each single time series variable in an EHR is a ordered set of *state intervals*



Multivariate state sequences

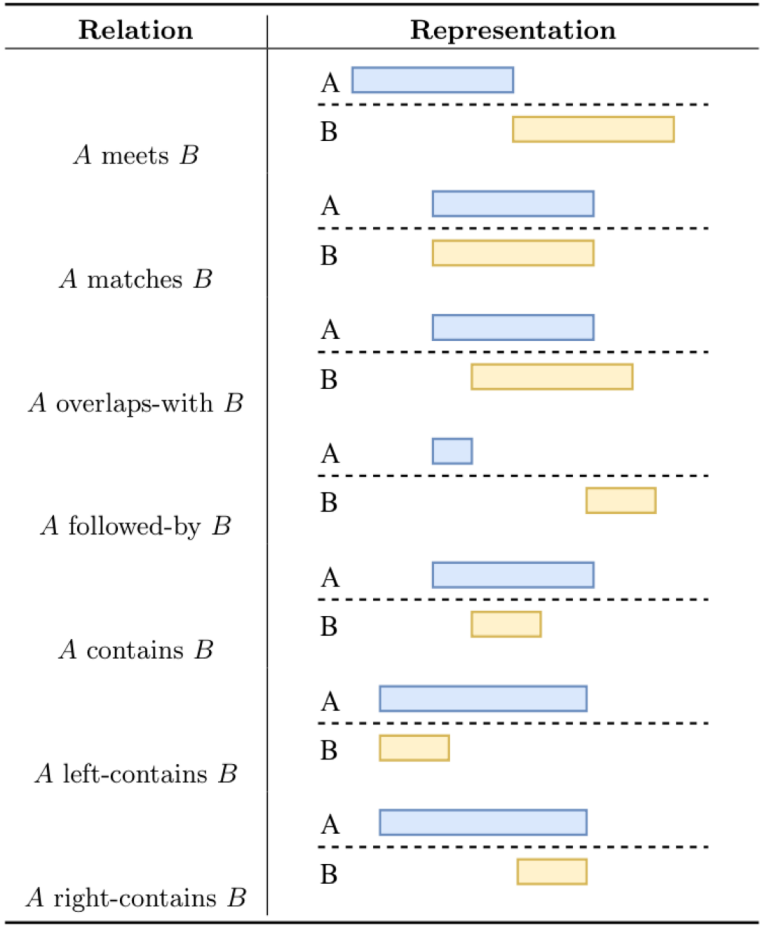
- **Multivariate state sequence Z_i – (basically a patient record):** an ordered combination of state intervals for all variables
  - *ordered* by start time
  - if *start times collide*, sort by *end time*
  - if *both collide*, sort by *lexical ordering*



Temporal abstraction patterns

- **What is a temporal pattern?**
  - a sequence of ***"temporal relations"*** between state intervals A and B
- **What kind of "temporal relations"?**

Based on Allen's temporal logic



Temporal abstraction patterns

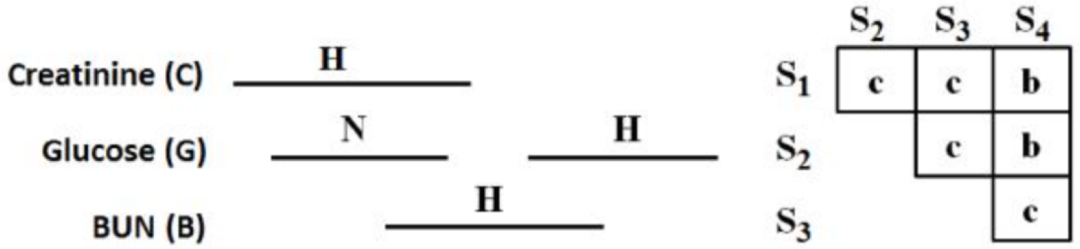
Temporal Pattern

$$\mathbf{P} = (\langle \mathbf{S_1}, \mathbf{S_2}, \ldots, \mathbf{S_k} \rangle, \mathbf{R})$$

- $\mathbf{S_i}$: state i
- **R**: the relation matrix defining the pair-wise relations between the states
  - ***R is a upper triangular matrix***
  - ***P is called k-pattern where*** $k = |\langle S_i, \ldots, S_k \rangle|$

Temporal abstraction patterns

A temporal pattern:



simplification
b: before
c: co-occurs

with states:

$$\langle (C, H), (G, N), (B, H), (G, H) \rangle$$

and relations:

$$R_{1,2} = c,\ R_{1,3} = c,\ R_{1,4} = b,\ R_{2,3} = c,\ R_{2,4} = b \text{ and } R_{3,4} = c$$

Pattern containment

Given:

- **a pattern** P = (<$\mathbf{S_1}$, $\mathbf{S_2}$, …, $\mathbf{S_k}$>, **R**)
- **an MSS** Z = <$\mathbf{E_1}$, …, $\mathbf{E_m}$>

Z contains P iff

- all $\mathbf{S_i}$ are in **Z**
- for *i= 1, …, k* and *j = i, …, k-1*, temporal relation $\mathbf{R_{i,j}}$ holds

Goal: identify containment of a pattern in a temporal sequence

Recent state intervals and patterns

- Given a point of interest (e.g., a diagnosis timestamp)
  - **recent state intervals:** occurring within ***at most g time units before*** the point of interest
  - **recent patterns:** with all consecutive events being ***at most g' time points apart***

Goal: frequent recent patterns

Frequent Recent Pattern

Given a database *D* of MSS, gap parameter *g*, g' and a support threshold σ

A pattern **P** is called "recent-frequent" if the number of times it occurs "recent-frequently" in *D* is greater than or equal to σ, i.e.,

$$RTP\text{-}sup_g(P, D) \geq \sigma$$

MSS Mining: the goal

- **Goal:** For a given database, for all given labels

  → *find all Frequent Recent Patterns associated with each given label*

- In other words…
  - for each *class y*, given the *database* D_y, output a *set of patterns* that satisfy:

$$\{P \in TP : RTP - sup_g(P, D_y) \geq \sigma_y\}$$

$$conf\left(P \Rightarrow y\right) = \frac{sup(P, D_y)}{sup(P, D)}$$

MSS Mining Algorithm

Simple approach

- Build patterns of incremental size
- Start with patterns of size 1 and build on top of that
- For $(k+1)^{th}$ stage, i.e., to find (k+1)-RTPs given K-RTPs, the algorithm follows two steps:
  - **candidate generation** using frequent patterns with common suffix sub-patterns
  - **removing candidates** that do not qualify

Generating coherent candidates

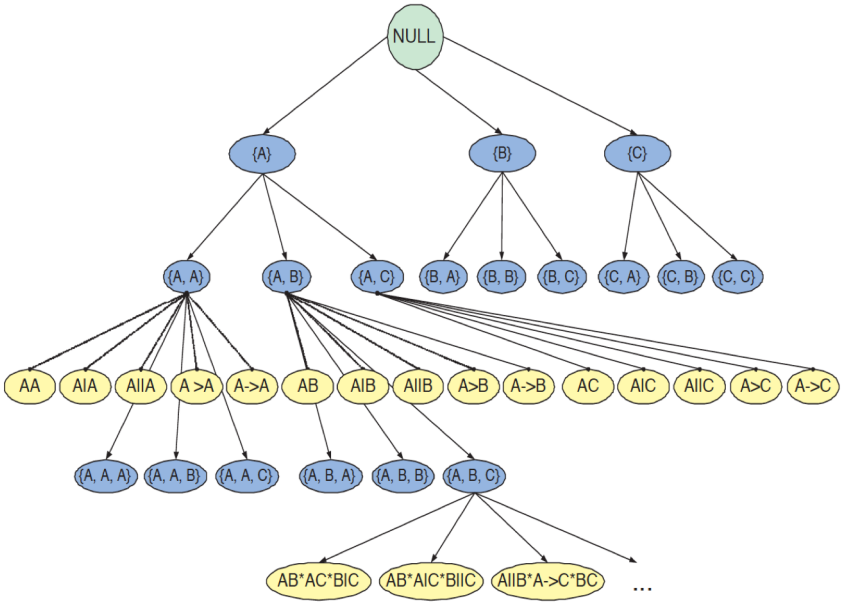
Prefix-based approaches

Input: Frequent *k-patterns* (F_k)
Output: Candidate *(k+1)-patterns* ($Cand$) with their *pid-lists*

```
1  foreach P ∈ F_k do
2      foreach I ∈ F_1 do
3          C = generate_coherent_candidates(P, I);
4          for q = 1 to | C | do
5              S = generate_k_subpatterns(C[q]);
6              if ( S[i] ∈ F_k : ∀i ∈ {1, ..., k} ) then
7                  C[q].pid-list = F_{k_S[1]}.id-list ∩ ... ∩ F_{k_S[k]}.id-list;
8                  C[q].mcs = max{F_{k_S[1]}.mc, ..., F_{k_S[k]}.mc};
9                  if ( | C[q].pid-list | ≥ σ_y ) then
10                     Cand = Cand ∪ C[q];
11                 end
12             end
13         end
14     end
15 end
16 return Cand
```

[Batal2012]

Tree-based approaches



[Papapetrou2009, Moskovitch 2014]

Improving efficiency: incoherent patterns

Removing "**incoherent**" patterns:

- some candidate patterns may yield incoherent temporal relations
- "incoherence" depends on the application domain at hand
- some works [Batal2012] avoid co-occurrences of the same event label
- others [Papapetrou2009, Kostakis2015] allow the same event label to repeat and correlate to itself

Support counting

Naïve counting

- for each variable y
  - for each candidate P and each MSS Z in database D_y
    - ✓ verify if P is a RTP in Z and increment Count for P for variable y

Other alternatives

- *Suffix lists*
- *Bitmap representations*
- *Hashing*

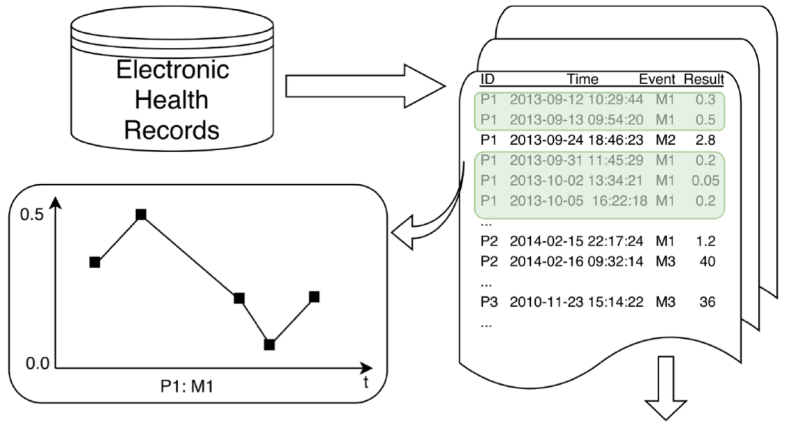
Learning a predictive model

- For each instance in D, **get the temporal abstraction (MSS) Z_i**
- **Mine frequent RTPs** for each label
- **Combine** all the RTPs into a set Ω
- **Create** a feature vector f of size |Ω|
  - for each MSS Z_i, create a binary feature vector: set it to 1 if that pattern is in Z_i, and 0 otherwise
- Use any classifier (ANN, SVN, etc.) for learning

MMS: Some remarks

- "Recent Temporal Patterns" are of special interest, especially in medical domain, but should have similar behavior in other domains
- Time series abstractions provide a simple approximation as well as compression of data
- The gap parameter in detecting pattern is critical for scaling up the mining process (but is domain dependent)
- RTPs provide efficient mining and higher prediction accuracy as compared to detecting patterns over the entire series (validated in the medical domain)
- How can we leverage/extend this?
  - Towards defining multi-level abstractions for time series
  - Extend from "independent" multivariate to interdependent multivariate models, where different vertices form variables and the edges define the dependencies

Temporal abstractions of sparse EHRs



ID	C	M1	M2	M3	...
P1	1			NA	...
P2	0				...
P3	1		NA		...
...	...	...	...	...	...

Hielscher et al. **Mining Longitudinal Epidemiological Data to Understand a Reversible Disorder**, Intelligent Data Analysis, 2014

Zhao et al. **Learning from Heterogeneous Temporal Data in Electronic Health Records**, Journal of Biomedical Informatics, 2017

Bagattini et al. **A classification framework for exploiting sparse multi-variate temporal features with application to adverse drug event detection in medical records**, BMC Medical Informatics and Decision Making, 2019

Framework overview

ID	C	M1	M2	M3	...
P1	1			NA	...
P2	0				...
P3	1		NA		...
...	...	...	...	...	...

Multi-variate feature representation

$\mathcal{L}$

Normalization

Summarization

Symbolic representation

Alphabet calibration

$\hat{\mathcal{L}}$

Subsequence generation

Subsequence evaluation

Subsequence selection

$\mathcal{Z}^* = s^*(\hat{\mathcal{A}}_1), \ldots, s^*(\hat{\mathcal{A}}_m)$

Data transformation

$\mathcal{D} = \tau^*(\mathcal{O})$

Phase A

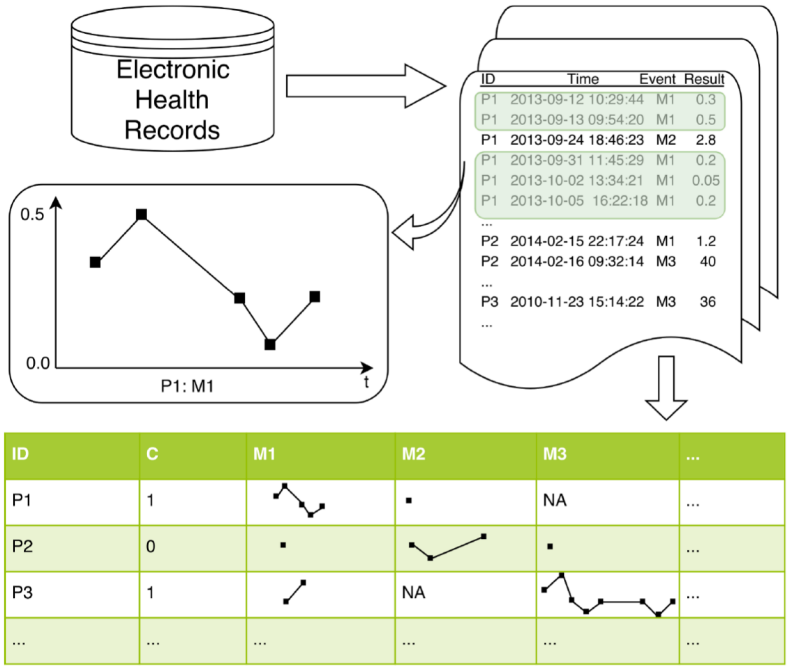
Phase B

Phase C

Phase A: normalization

- Z-normalization
  - Each multi-variate feature S is z-normalized:

$$S := \frac{\sum_{i=1}^{|S|} \{s_i - \mu(S)\}}{\sigma(S)}$$



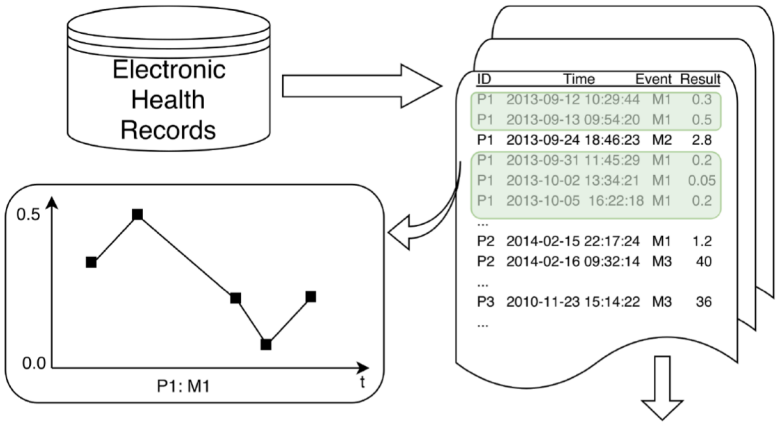
Phase A: summarization

- Z-normalization
  - Each multi-variate feature S is z-normalized:

$$S := \frac{\sum_{i=1}^{|S|} \{s_i - \mu(S)\}}{\sigma(S)}$$

- Summarization
  - Piecewise Aggregate Approximation (PAA)
  - Dimensionality reduction from d to w

$$\overline{S} = \{\overline{s}_1, \ldots, \overline{s}_w\}$$

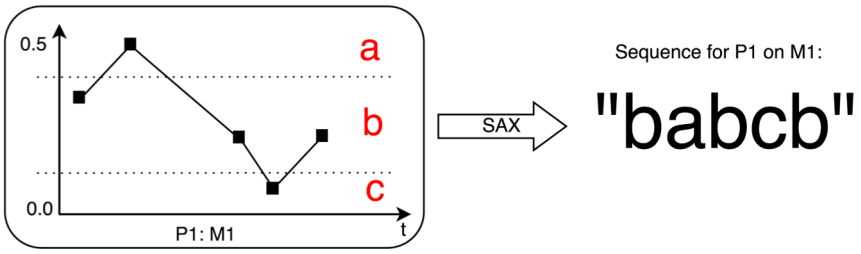


ID	C	M1	M2	M3	...
P1	1			NA	...
P2	0				...
P3	1		NA		...
...	...	...	...	...	...

$$\overline{s}_i = \frac{w}{d} \sum_{j=\frac{d}{w}(j-1)+1}^{\frac{d}{w}i} s_j$$

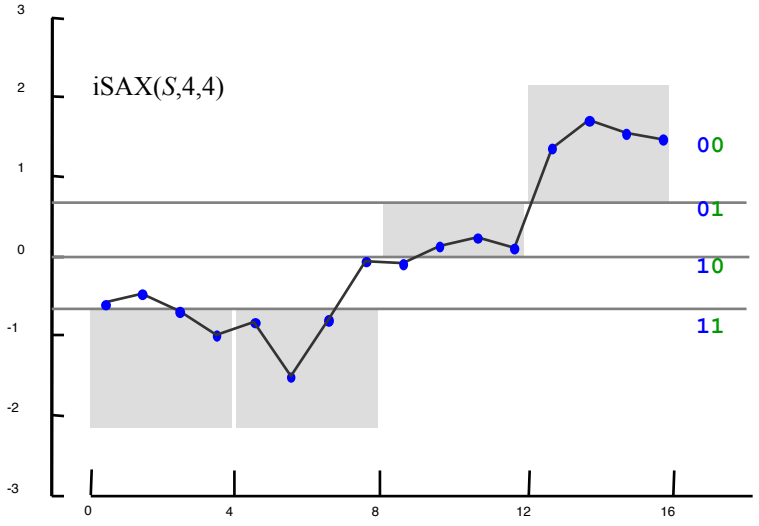
Phase A: symbolic representation

- SAX mapping
  - each record is mapped to a string using SAX
  - **length:** number of measurements
  - **alphabet:** 2 – 5, or set using domain knowledge

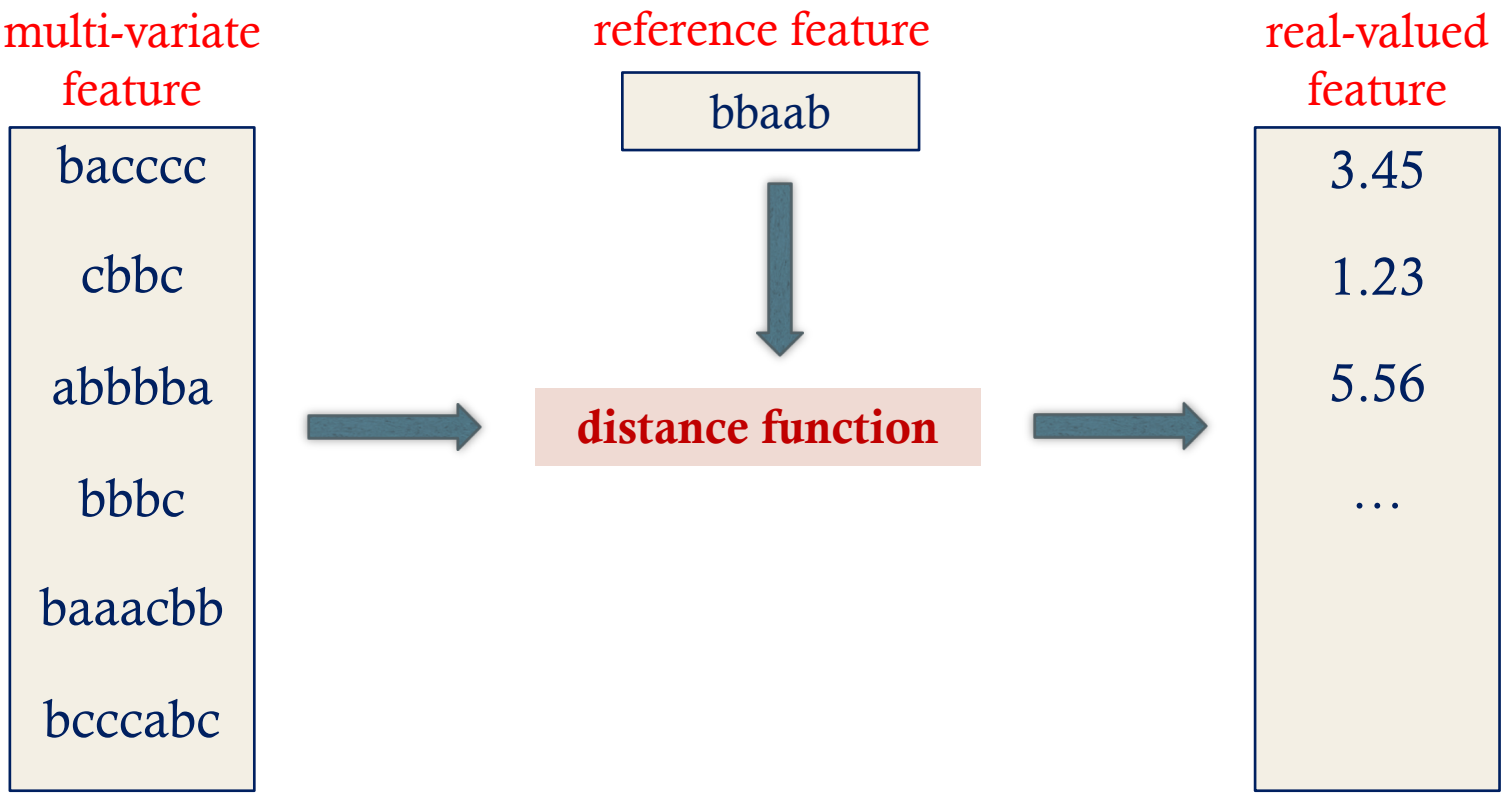


Discretize into a vector of symbols

- breakpoints map to small alphabet ***a*** of symbols



Phase B: mapping to real features



Phase B: subsequence enumeration

- *s-shapelet* generation:
  - random subsequences ***s*** of length t ∈ [1, l_{max}]

l_{max}: max length of a feature sequence

- *s-shapelet* evaluation:
  - each *s* is converted to a real value based on its distance to each multi-variate feature sequence

$$Dist\left(s, \widehat{S}\right) := \min_{s' \subseteq \widehat{S}, |s'|=|s|} \left\{D\left(s, s'\right)\right\}$$

Phase B: subsequence selection

- For each mutli-variate feature:
  - select the s-shapelet s^* with the max utility:

$$s^* := \arg\max_{s \in \widehat{\mathcal{L}}} Gain\left(s, \delta_{osp}(s), \widehat{\mathcal{L}}\right) \qquad s^*_\alpha := \arg\max_{s \in S_\alpha} Gain\left(s, \delta_{osp}(s), \widehat{\mathcal{L}}\right)$$

  - select the alphabet size with the max utility

$$\alpha^* := \arg\max_{\alpha \in \mathcal{I}} Gain\left(s^*_\alpha, \delta_{osp}(s^*_\alpha), \widehat{\mathcal{L}}\right)$$

- Final set of s-shapelets:

$$\mathcal{Z}^* = \left\{ s^*\left(\widehat{\mathcal{A}}_1\right), \ldots, s^*\left(\widehat{\mathcal{A}}_m\right) \right\}$$

Phase C: transformation

- A function τ* is learned:
  - transform any data object of the original multi-variate space to a set of real-valued features

$$\tau^* : \mathcal{A} \rightarrow \mathbb{R}^m$$

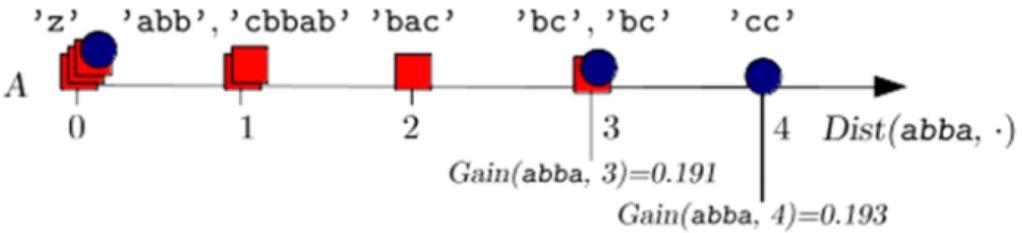
- Each data example is transformed using τ*:

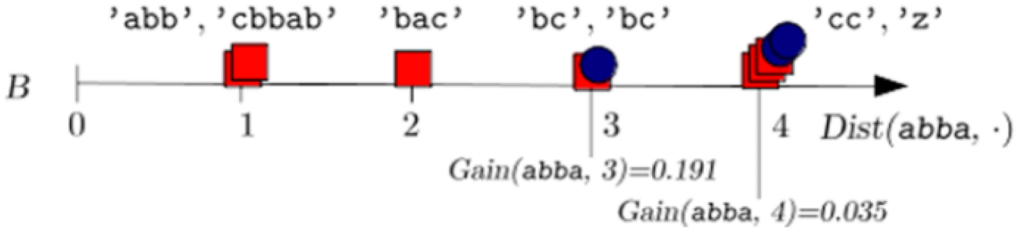
$$\tilde{O} = \tau^*(O)$$

Sparsity encoding

- Three ways of handling sparse features:
  - **plain:** no encoding [Zhao et al. 2017]
  - **mc:** most common encoding
  - **lr:** left-right encoding [Bagattini et al. 2019]

lr encoding





Assign all empty records either to ***0*** or ***max_dist***

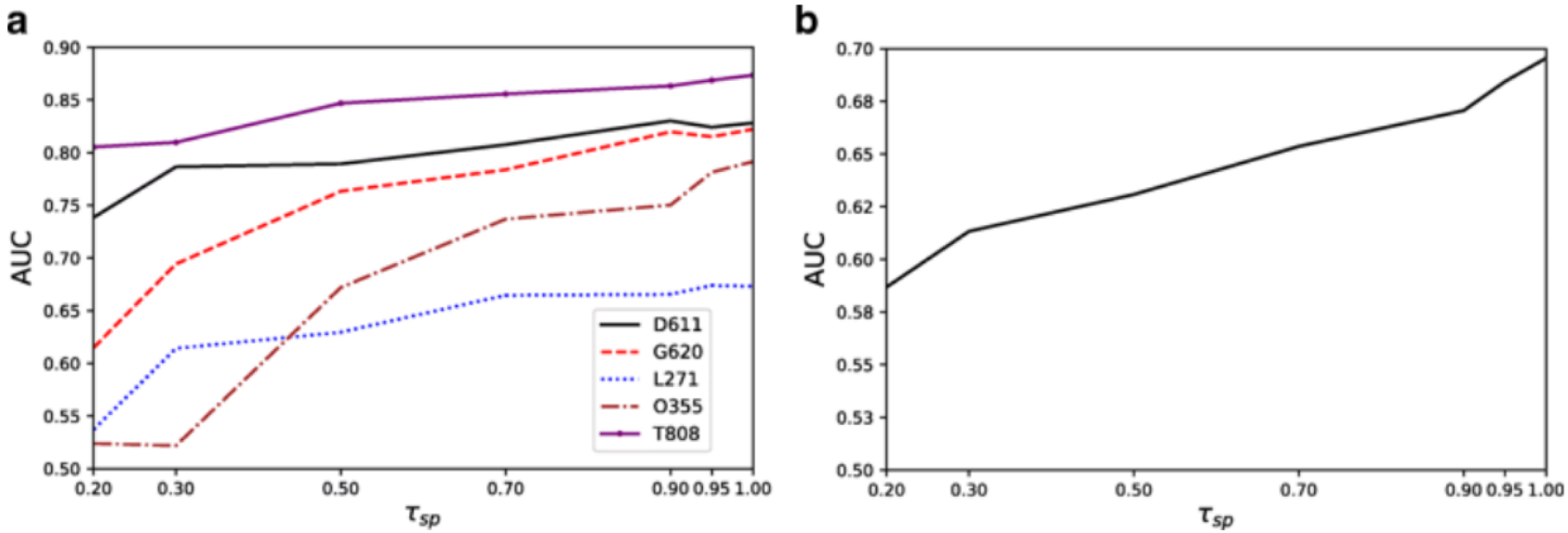
$$Dist(s, \emptyset) := \begin{cases} 0 & \emptyset \to \hat{\mathcal{L}}_1 \\ \max\limits_{\hat{S} \in \hat{\mathcal{L}}} Dist\left(s, \hat{S}\right) & \emptyset \to \hat{\mathcal{L}}_2 \end{cases}$$

Dataset	Class label description	Pos.	Neg.	Feat.
D611	Drug-induced aplastic anaemia	593	105	285
D642	Drug-induced secondary sideroblastic anaemia	217	9673	513
D695	Secondary thrombocytopenia	1246	2148	450
E273	Drug-induced adrenocortical insufficiency	70	259	229
G620	Drug-induced polyneuropathy	96	783	258
I952	Drug-induced hypotension	115	1287	324
L270	Drug-induced generalized skin eruption	182	468	314
L271	Drug-induced localized skin eruption	151	498	311
M804	Drug-induced osteoporosis with pathological fracture	52	1170	282
M814	Drug-induced osteoporosis	57	5097	434
O355	Maternal care for (suspected) damage to fetus by drugs	146	260	148
R502	Drug-induced fever	80	6434	498
T782	Adverse effects: anaphylactic shock	131	856	293
T783	Adverse effects: angioneurotic oedema	283	720	293
T784	Adverse effects: allergy	574	415	294
T801	Vascular complications following infusion, transfusion and therapeutic injection	66	609	229
T808	Other complications following infusion, transfusion and therapeutic injection	538	138	229
T886	Drug-induced anaphylactic shock	89	1506	363
T887	Unspecified adverse effect of drug or medicament	1047	550	363

Dataset	Class label description	Pos.	Neg.	Feat.
D611	Drug-induced aplastic anaemia	593	105	285
D642	Drug-induced secondary sideroblastic anaemia	217	9673	513
D695	Secondary thrombocytopenia	1246	2148	450
E273	Drug-induced adrenocortical insufficiency	70	259	229
G620	Drug-induced polyneuropathy	96	783	258
I952	Drug-induced hypotension	115	1287	324
L270	Drug-induced generalized skin eruption	182	468	314
T808	Other complications following infusion, transfusion and therapeutic injection	538	138	229
T886	Drug-induced anaphylactic shock	89	1506	363
T887	Unspecified adverse effect of drug or medicament	1047	550	363

- ADEs are injuries that occur from the use of a drug, such as overdoses or dose reductions, or drug interactions
- They account for 3.7% of hospital admissions around the world
- ADEs have been estimated to come at a cost of $3.5 billion in the U.S alone, despite ADEs being preventable

lr: AUC vs feature sparsity



- As the % of **feature sparsity** increases, **AUC** also increases!
- Shorter s-shapelets (i.e., 2-8) are preferable to longer ones (> 20)

Ongoing improvements

- **Variable window length** over the patient history
- Exploiting all **variable types**
- Looking into **textual features**
- Deep learning models with **attention mechanisms**
- Model **interpretability** and **explainability**

Do not miss!

- **Workshop on Applied Data Science for Healthcare (DSHealth)**
- 1-5pm tomorrow!
- POSTER #14:

  ”Aggregate-Eliminate-Predict: Detecting Adverse Drug Events from Heterogeneous Electronic Health Records”

  by Maria Bampa and Panagiotis Papapetrou