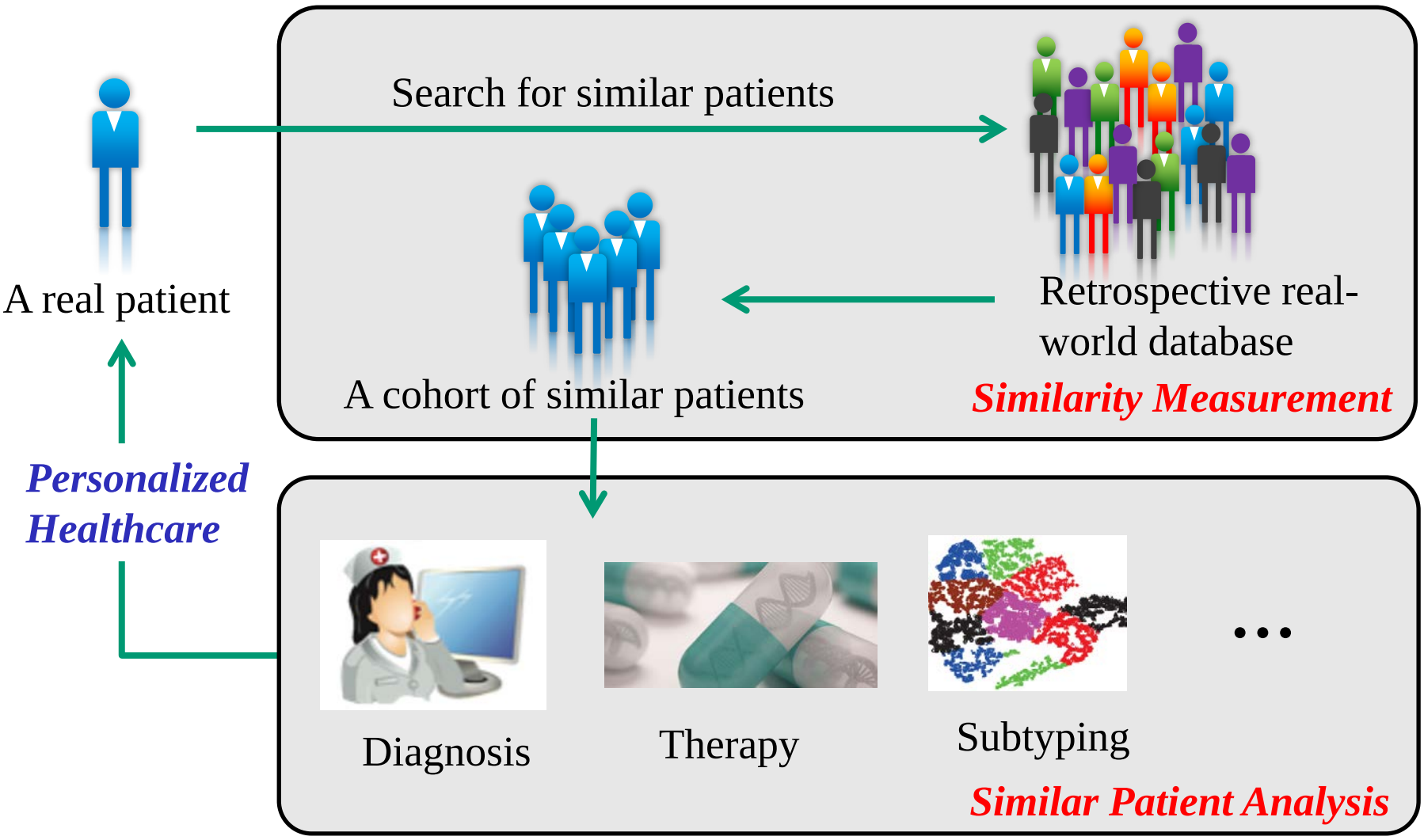
Patient Similarity Learning for Personalized Medicine

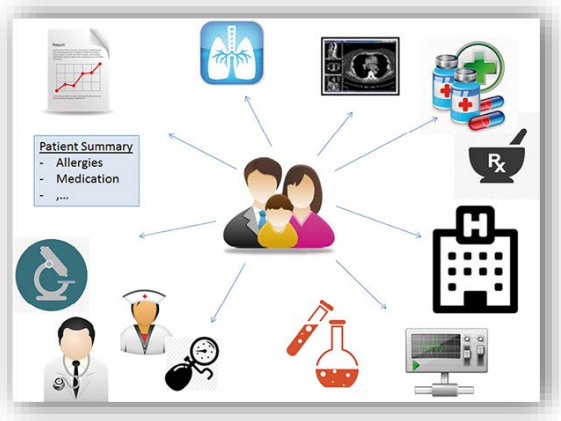
Aidong Zhang
Program Director at NSF
and
SUNY Distinguished Professor
State University of New York (SUNY) at Buffalo

Personalized Modeling



Patient Similarity

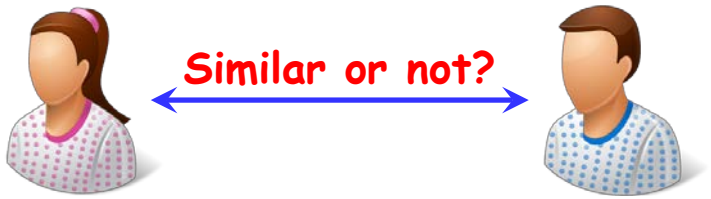
- **Tremendous clinical information of patients are available due to the prevalence of Electronic Health Records (EHRs)**



- **These wealth clinical information make it possible to perform patient similarity analysis**
  - A fundamental problem in healthcare informatics
  - The goal is to measure the similarity between a pair of patients
  - Could help to retrieve similar reference cases for predicting the clinical outcome of interest

Patient Similarity

- The key part of **patient similarity learning** is to learn a clinically meaningful and precise **similarity metric**



- Learn a similarity function

Needs to be learned

$$s(x_i, x_j) = x_i^T M x_j ,$$

which can measure the similarity between any two patient samples x_i and x_j

- Compared with some simple metrics, such as Euclidean distances, the metric generated by patient similarity learning can capture **more statistical regularities** in the patient data

Metric Learning

Metric Learning

- Metric learning is to learn **a distance metric**, which pulls the same class samples closer together and push different class samples further apart.

Fig. Illustration of metric learning applied to a toy dataset, which contains two kinds of samples (i.e., red dots and blue triangles).

Distance metric learning has been studied by many works which address different metric learning problems

- Sparse feature selection
- Multi-class metric learning
- Deep metric learning
- ….

Metrics Learned Used in Many Applications

Needs for similarity measures

- Classifications: k-nearest neighbors, support vector machines …
- Clustering: k-means and its variants.
- Data visualization in high dimensions.
- Zero-short learning
- Person re-identification
- …

The performance of these algorithms heavily depends on the choices of similarity between objects.

Challenges

- **The patient data are usually high dimensional, complex and noisy**
  - The features used for similarity learning may contain much irrelevant and redundant information
  - These information can hide the relationship between the learning task and the most relevant features
- **It is essential to remove irrelevant and redundant information when conducting patient similarity learning**
- **To address the complex nature of patient data, some sparse feature selection methods have been proposed**
  - Select relevant features that are highly correlated with the learning task
  - Ignore the correlations among the selected features

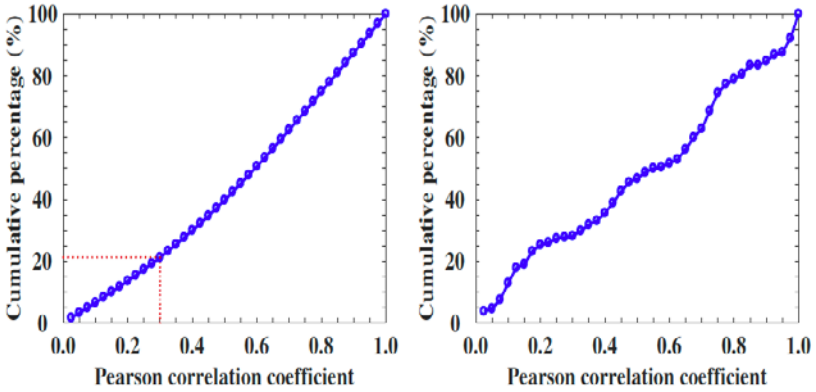
Uncorrelated Patient Similarity Learning (SDM 2018)

Metric learning for patient similarity measure

- The goal of patient similarity learning is to derive a clinically meaningful distance metric to measure the similarity between a pair of patients according to their clinical information.

The challenges

- Existing works mainly focus on sparse feature selection, and they ignore the correlations among the selected features.



The correlated features may share similar properties and reveal overlapped information

Figure : Feature correlations on Colon Cancer dataset and Parkinson's disease dataset.

- Patient data may be distributed across different sites.

Uncorrelated Patient Similarity Learning (UnPSL)

- **Step 1:** Formulate the similarity learning problem as a *maximum likelihood estimation* problem.
- **Step 2:** Introduce *two regularization terms*, controlling *sparse feature selection* and *uncorrelated feature selection.*

Distributed patient similarity learning

- Decompose UnPSL such that the metric can be learned *without directly accessing the raw data at each site.*

Table 1 : The average feature correlation over Parkinson disease dataset under three different situations: (1) the original dataset, (2) only considering sparsity; (3) considering both sparsity and correlation

Original dataset	Sparsity	Sparsity +correlation
0.502	0.412	**0.337**

Table 2 : Performance comparison on Parkinson disease dataset.

	UnPSL	Cosine	Euc	GMML	ITML	LMNN	Low -Rank
Accuracy	**0.822**	0.643	0.775	0.744	0.793	0.815	0.797
F2-score	**0.880**	0.759	0.798	0.856	0.843	0.866	0.719

Distributed Patient Similarity Learning

Input:

- A set of parties (i.e., sites) $\mathcal{P} = \{1,2,3,\dots,P\}$
- Each party p has two sets of sample pairs:

$$I_s^p = \{(x_{ip}, x_{jp}) : y_{ij}^p = 1\} \text{ and } I_d^p = \{(x_{ip}, x_{jp}) : y_{ij}^p = -1\},$$

Two patient samples of party p

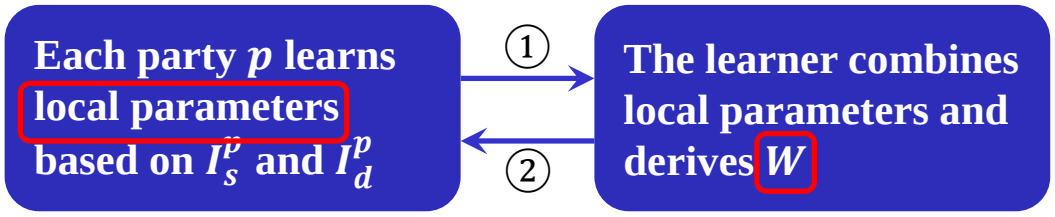
Denotes x_{ip} and x_{jp} are similar

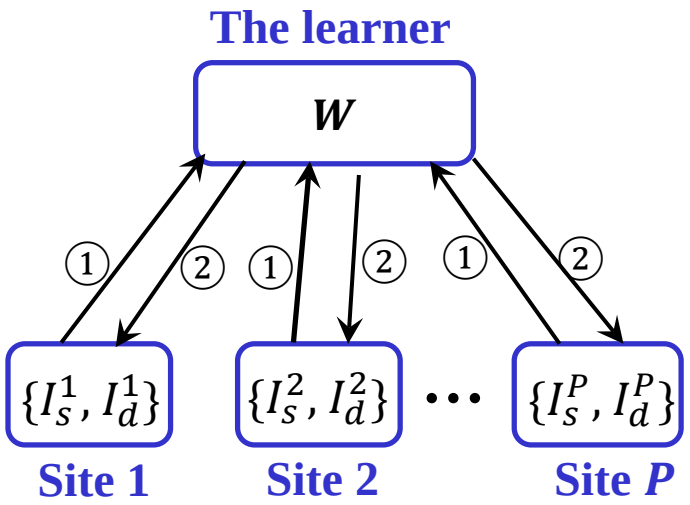
Denotes x_{ip} and x_{jp} are not similar

Output:

- The similarity metric W

Main idea:





Metric Learning from Probabilistic Labels (KDD 2018)

Probabilistic Labels

- An implicit assumption in traditional metric learning setting is that the associated labels of the data instances are deterministic.
- In many real-world applications, the associated labels come naturally with probabilities instead of deterministic values.
  - Instance-wise probabilistic label (Figure (b)).
  - Group-wise probabilistic label (Figure (c)).

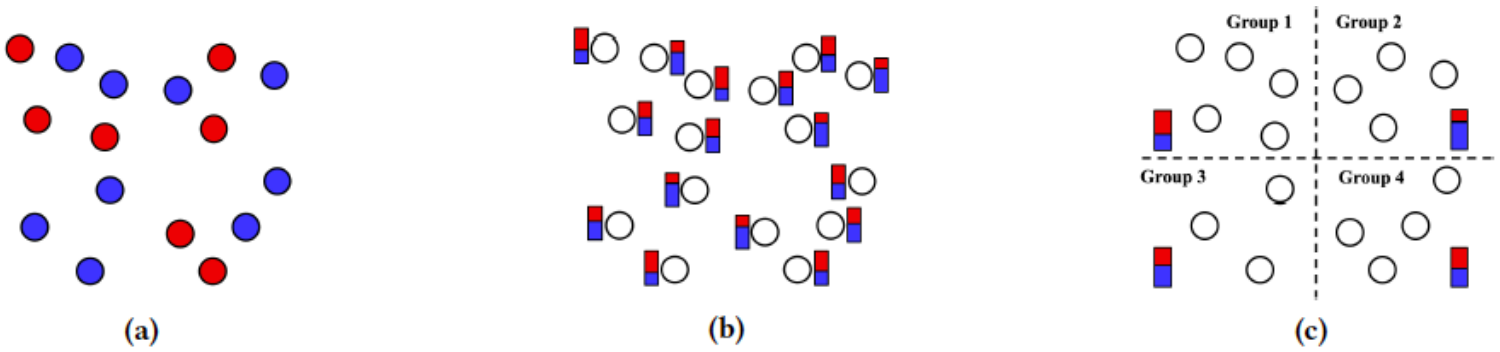


Figure: The datasets with different label information. (a) Deterministic labels. (b) Instance-wise probabilistic labels. (c) Group-wise probabilistic labels.

Metric Learning from Probabilistic Labels

Instance-level metric learning method (InML)

- Step 1: Construct the relative constraints via ranking the instance-wise probabilistic labels;
- Step 2: Design an optimization function to enforce the constructed relative constraints.

Table: The accuracy of InML on Breast cancer dataset when the training dataset sizes are 50 and 100

	InML	Cosine	Euc	GMML	ITML	LMNN	Low-Rank	R2ML
50	**0.653**	0.500	0.530	0.551	0.329	0.647	0.534	0.563
100	**0.676**	0.551	0.568	0.571	0.354	0.653	0.566	0.571

Group-level metric learning method (GrML)

- Step 1: model the unknown pairwise similarity labels as latent variables (whether two instances are similar or not);
- Step 2: propose a maximum likelihood framework which jointly optimizes over the unknown similarity labels and the distance metric.

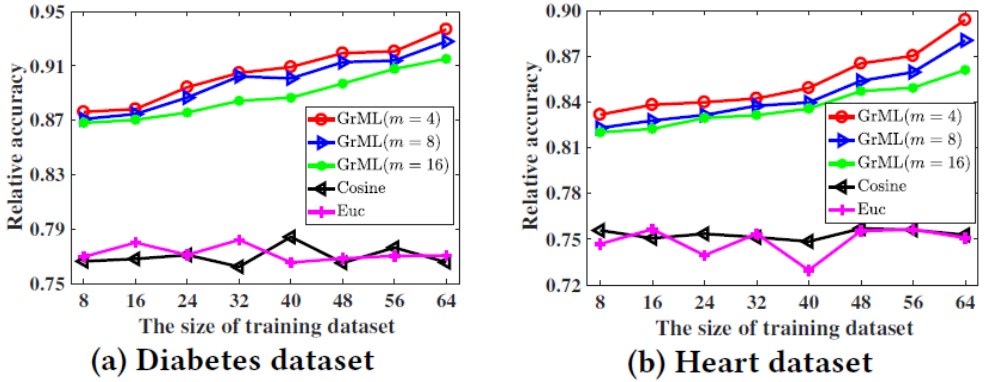


Figure: Relative accuracy of GrML w.r.t. the size of training dataset. The *relative accuracy* is defined as the accuracy of GrML relative to the accuracy that can be achieved by the traditional methods which have full access to the deterministic labels, e.g. LMNN.

Multi-task Patient Similarity Learning

How to capture the temporal progression of patient health condition

Challenges

- Patient health condition is changing slowly over time. Current methods ignore temporal relatedness among time points.
- Disease labels are ordinal: different severity levels.

Method

- Multi-task metric learning: each task is learning at one time point. Make use of the temporal **relatedness** among tasks to improve generalization performance

$$d_{\boldsymbol{t}} = (x_i - x_j)^T (W_0 + W_t)(x_i - x_j)$$

W_0: shared metric, W_t: task-specific metric

General Framework

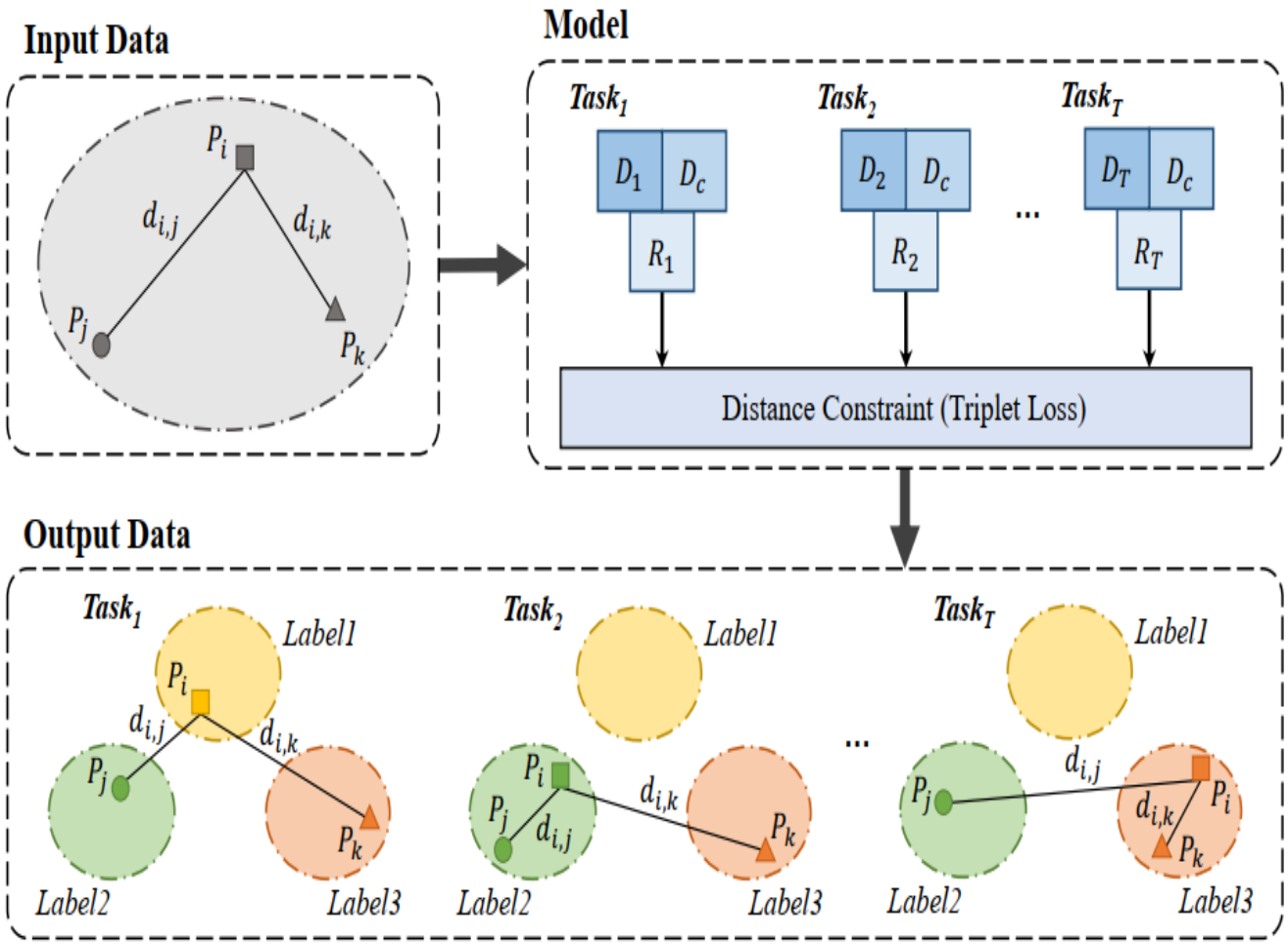


Fig. 1: The training process of the proposed method on measuring patient similarity progression. We use the combination of a common distance D_c and task-specific ones $D_1, D_2, ..., D_T$ to represent the desired distances, and then formulate the constraint function based on the distances. $R_1, R_2, ..., R_T$ are the sparse regularization terms.

Multi-task Metric Learning

- **Conventional framework**
  - Distance function

$$d_t^2(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{M}_0 + \mathbf{M}_t)(\mathbf{x}_i - \mathbf{x}_j),$$

  - Objective function

$$\min_{\mathbf{M}_0,\ldots,\mathbf{M}_T} \mathcal{L}(\mathbf{M}_0, \mathbf{M}_t) + \gamma_0 \mathrm{Reg}(\mathbf{M}_0) + \sum_{t=1}^{T} \gamma_t \mathrm{Reg}(\mathbf{M}_t)$$

- **Not suitable for high dimensional noisy data**
  - cannot learn a low-rank metric
  - not able to remove the noisy and irrelevant information of the input features
  - not able to constraint on the similarity degree information

Proposed Model

➢ Distance Constraint Construction

- consider a triplet of samples (x_i, x_j, x_k), and corresponding labels (c_i, c_j, c_k)
- $c_i = c_j \neq c_k$: x_i has the same label as x_j, and different with x_k.
- $c_i > c_j > c_k$ or $c_i < c_j < c_k$: the severity label of x_i is closer to x_j than to x_k
- $d^2(x_i, x_j) \leq d^2(x_i, x_k) - g$

➢ Learning framework

triplet constraint feature selection

$$\min_{\mathbf{L}_0,\ldots,\mathbf{L}_T} \mathcal{F} = \sum_{t=1}^{T} \left[\mathcal{L}_t(\mathbf{L}_0, \mathbf{L}_t) + \gamma_1 \|\mathbf{L}_t\|_{2,1} \right] + \gamma_0 \|\mathbf{L}_0\|_F^2$$

where,

$$\mathcal{L}_t(\cdot) = \frac{1}{|\mathcal{R}_t|} \sum_{(i,j,k)\in\mathcal{R}_t} [d_t^2(\mathbf{x}_i, \mathbf{x}_j) - d_t^2(\mathbf{x}_i, \mathbf{x}_k) + g]_+$$

Experiments: Setup

- Datasets
  - Alzheimer's Disease Neuroimaging Initiative (ADNI)
  - The study of osteoporotic fracture (SOF)

TABLE I: Statistics of the ADNI and SOF datasets.

Dataset	# of samples					# of features
	Task1	Task2	Task3	Task4	Task5	
ADNI	732	693	615	425	105	364
SOF	539	544	542	230	540	200

- Baselines
  - Single-task (ST) learning methods
  - Multi-task (MT) learning methods

Experiments: Results

Mean square error of KNN classification results on two datasets

TABLE II: Performance comparison on the ADNI dataset in terms of MSE.

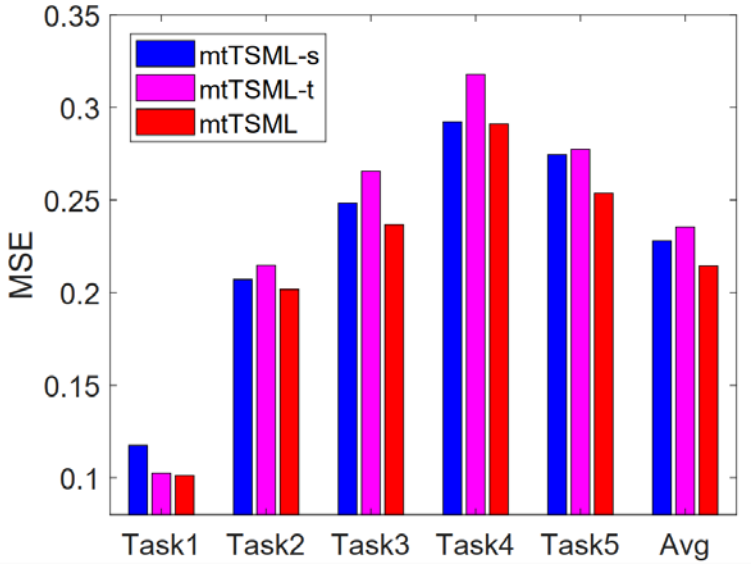
		ADNI (20% as training data)						ADNI (40% as training data)					
		Task1	Task2	Task3	Task4	Task5	Avg	Task1	Task2	Task3	Task4	Task5	Avg
ST	Euclidean	0.3795	0.4938	0.6369	0.6912	0.8583	0.6119	0.3617	0.4828	0.5818	0.6367	0.6470	0.5420
	Cosine	0.3654	0.5022	0.6508	0.6632	0.8052	0.5974	0.3443	0.4611	0.6017	0.6436	0.6134	0.5328
	GMML	0.2392	0.3616	0.5858	0.4376	0.4121	0.4072	0.2503	0.4221	0.5112	0.5450	0.3998	0.4257
	SCML	0.2096	0.3894	0.4481	0.5681	0.5147	0.4260	0.1995	0.3615	0.3864	0.4126	0.5217	0.3763
	LowRank	0.1699	0.2636	0.3104	0.4003	0.7528	0.3794	0.2027	0.3302	0.3646	0.3895	0.2568	0.3088
	ITML	0.1271	0.2227	0.3108	0.3481	0.3509	0.2719	0.1180	0.2290	0.2929	0.3615	0.3294	0.2662
	LMNN	0.1818	0.3257	0.3673	0.4471	0.5246	0.3693	0.1333	0.3128	0.3797	0.4516	0.4037	0.3362
	TSML*	**0.0957**	0.2057	0.2687	0.3274	0.4469	0.2689	0.1098	0.1892	0.2300	0.2830	0.3513	0.2327
MT	mtSCML	0.2301	0.3053	0.3798	0.4981	0.5147	0.3850	0.1913	0.2799	0.3117	0.3592	0.3478	0.2980
	mtLMNN	0.2137	0.2681	0.3270	0.3972	0.3425	0.3097	0.1470	0.2358	0.2935	0.3247	0.2560	0.2514
	CP-mtML	0.1317	0.2075	0.3823	0.3637	0.3336	0.2838	0.1098	0.2092	0.2965	0.3656	0.3234	0.2609
	mtMLCS	0.1777	0.2500	0.2800	0.3917	0.3734	0.2946	0.1355	0.2325	0.2379	0.3410	0.3237	0.2541
	mtTSML*	0.1011	**0.2018**	**0.2367**	**0.2976**	**0.2536**	**0.2182**	**0.0962**	**0.1766**	**0.2170**	**0.2652**	**0.2093**	**0.1929**

TABLE III: Performance comparison on the SOF dataset in terms of MSE.

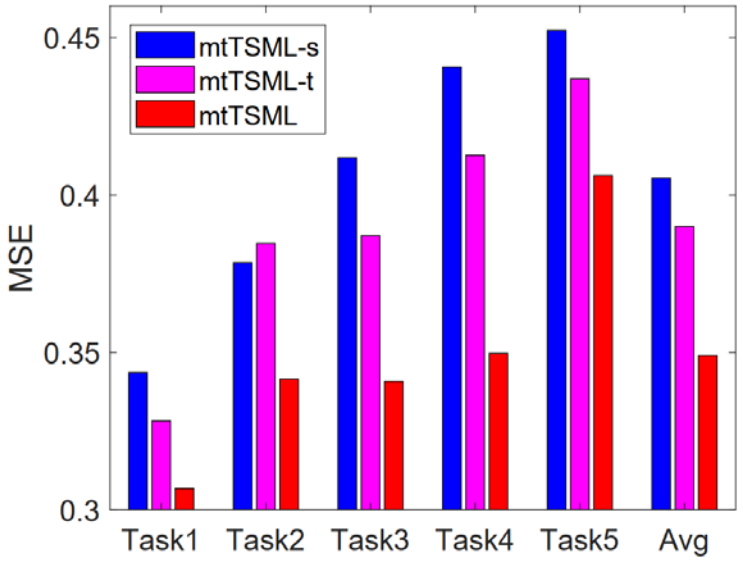
		SOF (20% as training data)						SOF (40% as training data)					
		Task1	Task2	Task3	Task4	Task5	Avg	Task1	Task2	Task3	Task4	Task5	Avg
ST	Euclidean	0.5951	0.6154	0.6409	0.8671	0.8031	0.7043	0.5417	0.5576	0.5207	0.6067	0.7256	0.5905
	Cosine	0.5552	0.5262	0.5666	0.8392	0.7600	0.6494	0.5691	0.5664	0.5786	0.5926	0.7392	0.6092
	GMML	0.4417	0.4369	0.4737	0.4685	0.5138	0.4669	0.4398	0.4424	0.4332	0.5730	0.5023	0.4782
	SCML	0.3558	0.4185	0.3560	0.5315	0.4677	0.4259	0.3333	0.3410	0.3628	0.6279	0.5000	0.4330
	LowRank	0.4847	0.4954	0.4892	0.5455	0.5846	0.5199	0.4352	0.4286	0.4240	0.6629	0.5116	0.4925
	ITML	0.4417	0.4185	0.5232	0.4825	0.5446	0.4821	0.4306	0.4147	0.4747	0.5281	0.6279	0.4952
	LMNN	0.3834	0.4369	0.4582	0.6503	0.7108	0.5279	0.3796	0.3318	0.3963	0.5506	0.4884	0.4293
	TSML	0.3282	0.3631	0.4180	0.4615	0.4462	0.4034	0.3287	0.3180	**0.3410**	**0.3708**	0.4558	0.3629
MT	mtSCML	0.3037	0.3846	0.3467	0.5245	0.4185	0.3944	0.2824	0.3226	0.3721	0.5814	0.4393	0.3995
	mtLMNN	0.3804	0.3938	0.4334	0.5245	0.5108	0.4486	0.3426	0.3410	0.4055	0.4270	0.4698	0.3972
	CP-mtML	0.3589	0.3538	0.3994	0.4895	0.4554	0.4114	0.3056	0.3456	0.3548	0.4270	0.4465	0.3759
	mtMLCS	0.3776	0.4055	0.4022	0.4814	0.5196	0.4372	0.3102	0.3088	0.3502	0.3933	0.4651	0.3655
	mtTSML	**0.3067**	**0.3415**	**0.3407**	**0.3497**	**0.4062**	**0.3490**	**0.2824**	**0.2811**	**0.3410**	0.3820	**0.4279**	**0.3429**

Experiments: analysis

- Comparison of reduced models
  - mtTSML-s: w/o feature selection regularizer
  - mtTSML-t: w/o $c_i > c_j > c_k$ and $c_i < c_j < c_k$ constraints



ANDI dataset



SOF dataset

- Smaller MSE values, better results
  - **Sparse feature selection** and **label similarity constraints** are both beneficial to model performance

Deep Metric Learning

Deep metric learning

- Deep metric learning aims to learn a *distance function* $D\left(f(x_i;\theta), f(x_j;\theta)\right)$ through training a deep neural network such that the similarity between any two instances x_i and x_j can be effectively computed.

Advantages

Methods	Size of dataset	Projection
Traditional Metric Learning	Small	Linear
Deep Metric Learning	**Large**	**Nonlinear**

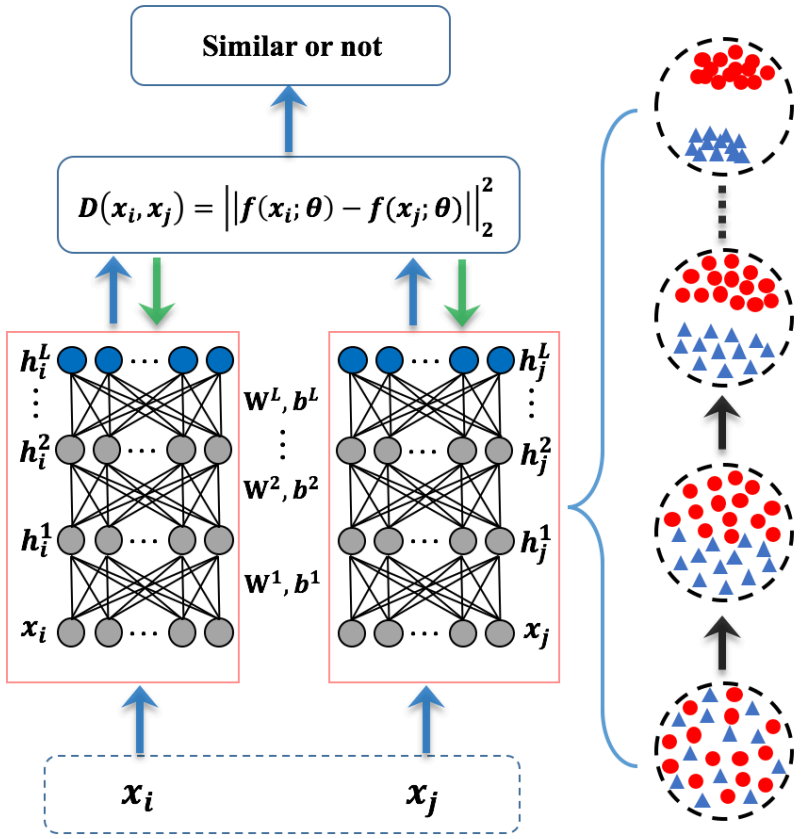
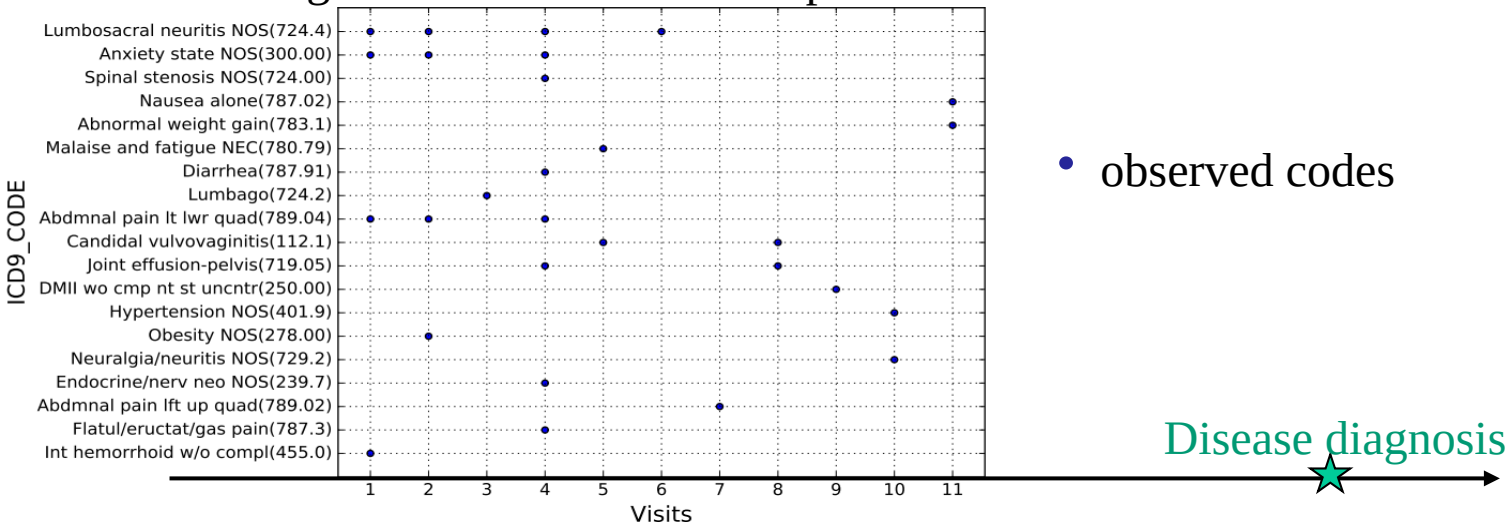


Figure: Deep metric learning model

Longitudinal EHR Data

EHR data: longitudinal, sparse and high dimensional

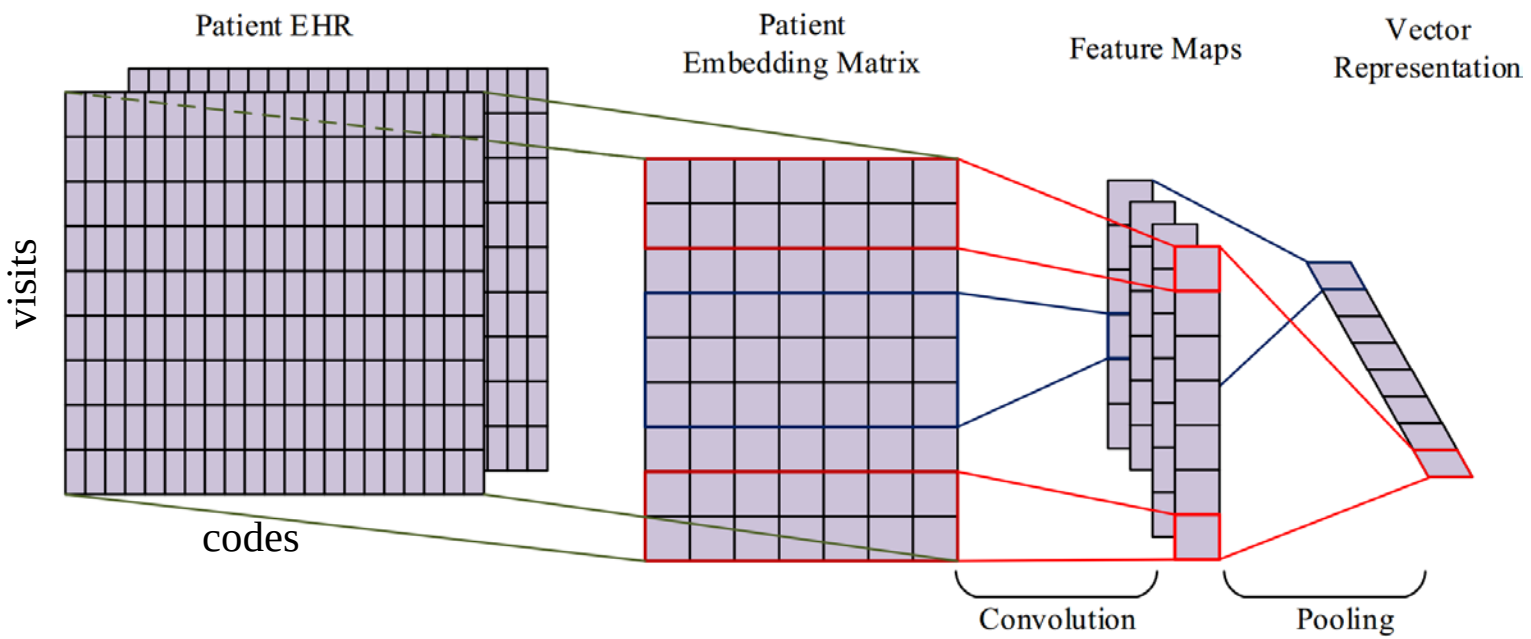
Fig. The data format of one patient



How to measure the similarity among patients?

- **Representation learning**: reduce feature dimension; capture sequential visit information; identify informative features
- **Similarity learning**: ensure patients from the same cohort have smaller distances than patients from different cohorts

Learning Patient Representations: CNN



- Embedding Layer
  - Reduce feature dimensions and learn code relationships.
- One-side Convolution
  - **Capture the sequential relations over adjacent visits.**
  - **Extract effective patterns.**
- Max-pooling
  - Capture the most important information for each feature map.

Deep Metric Learning for EHR

- How to measure patient similarity on **longitudinal** EHR?
- An end-to-end similarity learning framework

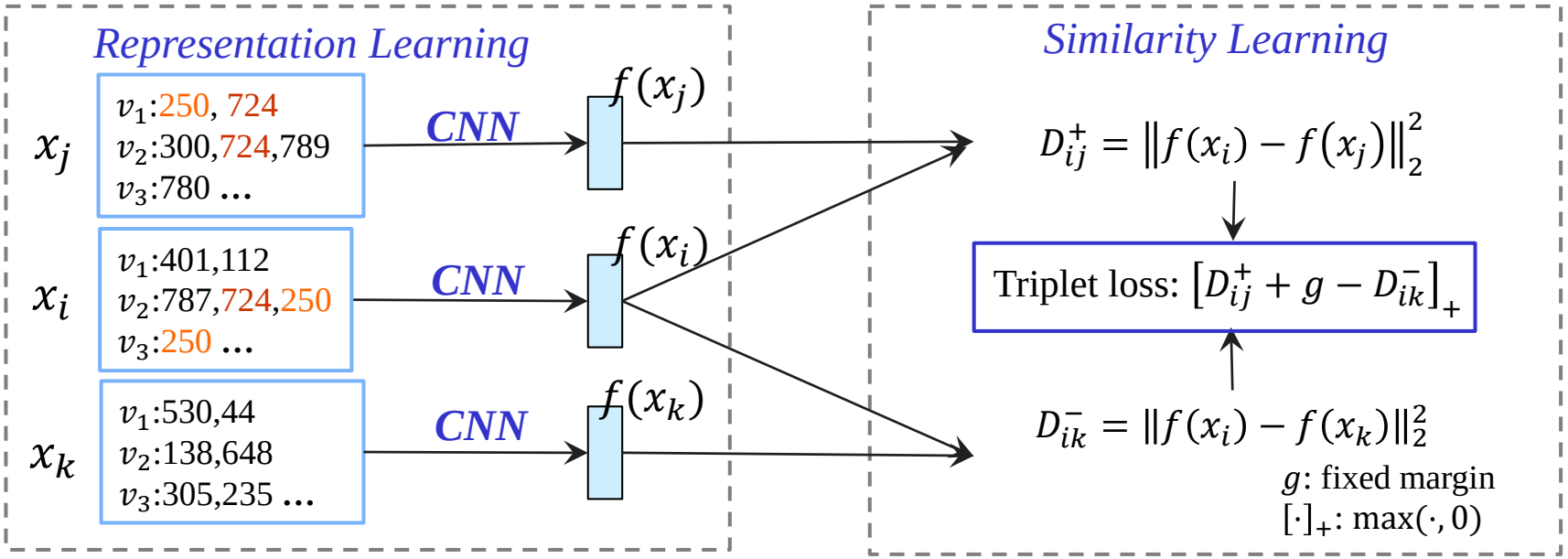


Fig: Macro-averaging of KNN classified results for diabetes, obesity and COPD.

Method	Accuracy	Recall	Precision	F1 score
Euclidean	0.5660	0.5490	0.6199	0.5347
Cosine	0.5981	0.5920	0.6032	0.5914
GMML	0.5877	0.5801	0.6062	0.5812
ITML	0.5751	0.5591	0.6202	0.5476
LMNN	0.6848	0.6789	0.6925	0.6808
CNN_triplet	**0.7736**	**0.7731**	**0.7740**	**0.7730**

- CNN makes use of sequential structure and learns local important information.
- Triplet loss ensures margin between positive pair and negative pair。
- Baselines: not able to deal with large data; cannot extract important features

Toward Smart Health

- **Paradigm Shift: From Reactive and hospital-centered to preventive, proactive, evidence-based, person-centered and focused on well-being rather than disease.**
- **From prediction to prescription: personalized prescription, recommendations, and control for medications, behavior, and other interventions.**
- **Ultimate goal: Toward precision medicine and optimize health.**
- **Metric learning is a basic but important step toward success of personalized medicine.**
- **Future work will focus on more sophisticated non-linear models on large data sets.**

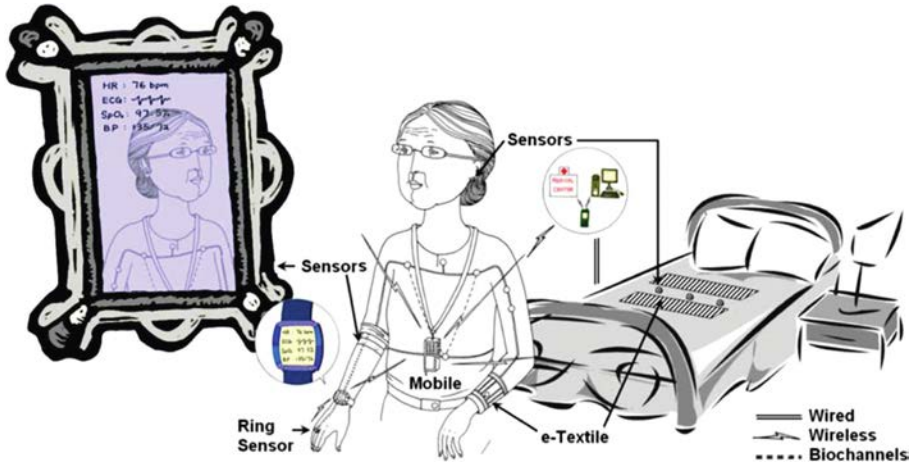


Illustration of unobtrusive physiological measurements for *p*-health in a home environment. As shown in "Health Informatics: Unobtrusive Physiological Measurement Technologies," by Zhang *et al.*, p. 000.