Introduction to Data Science: Structured Learning on Temporal Networks

Zoran Obradovic

zoran.obradovic@temple.edu
www.dabi.temple.edu/~zoran

Data Analytics and Biomedical Informatics Center,
Computer and Information Sciences Department,
Statistical Science Department,
Temple University, Philadelphia, USA

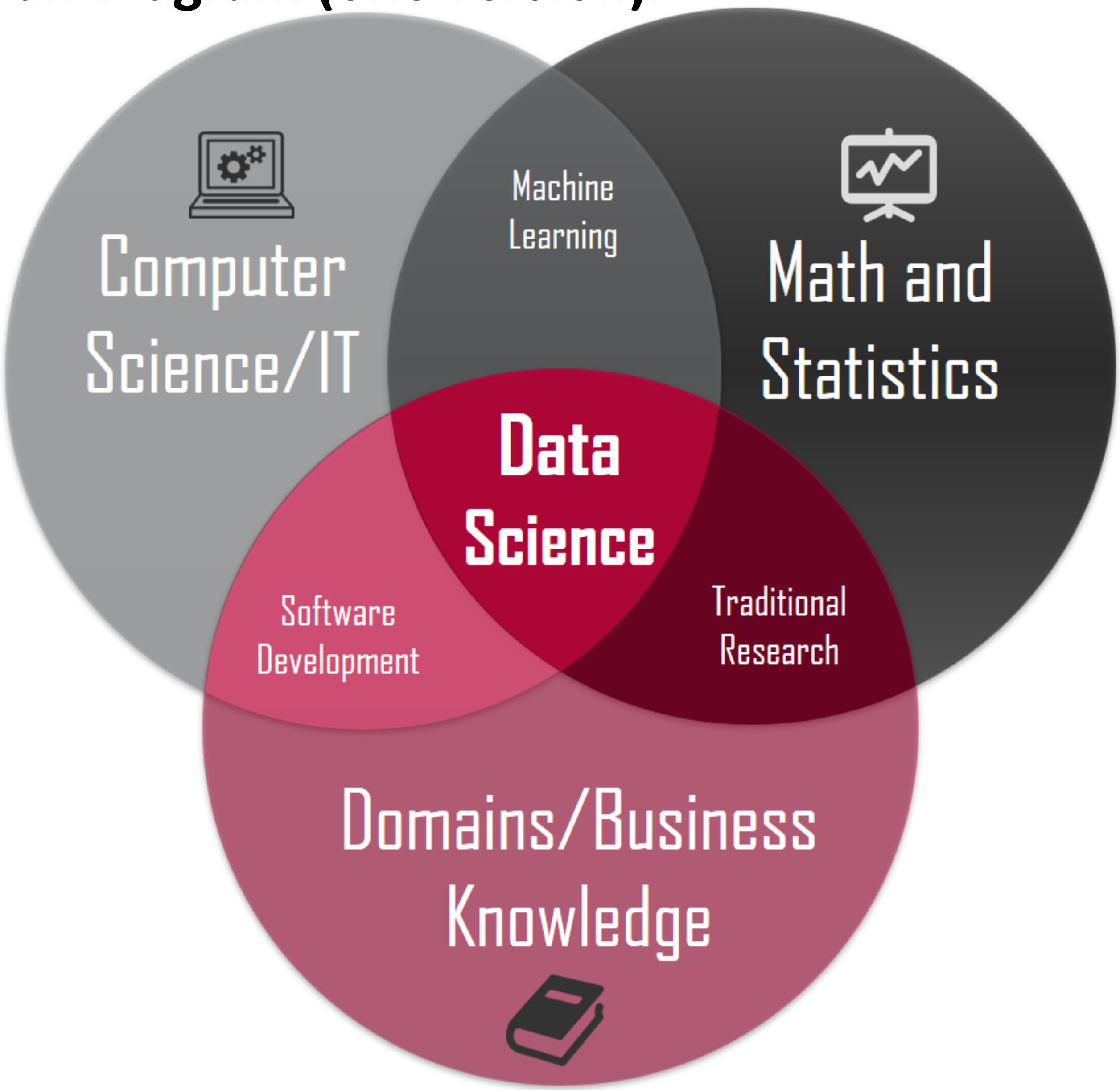
Talk of the town: Data Science

Harnessing the Data Revolution by exploiting Big Data

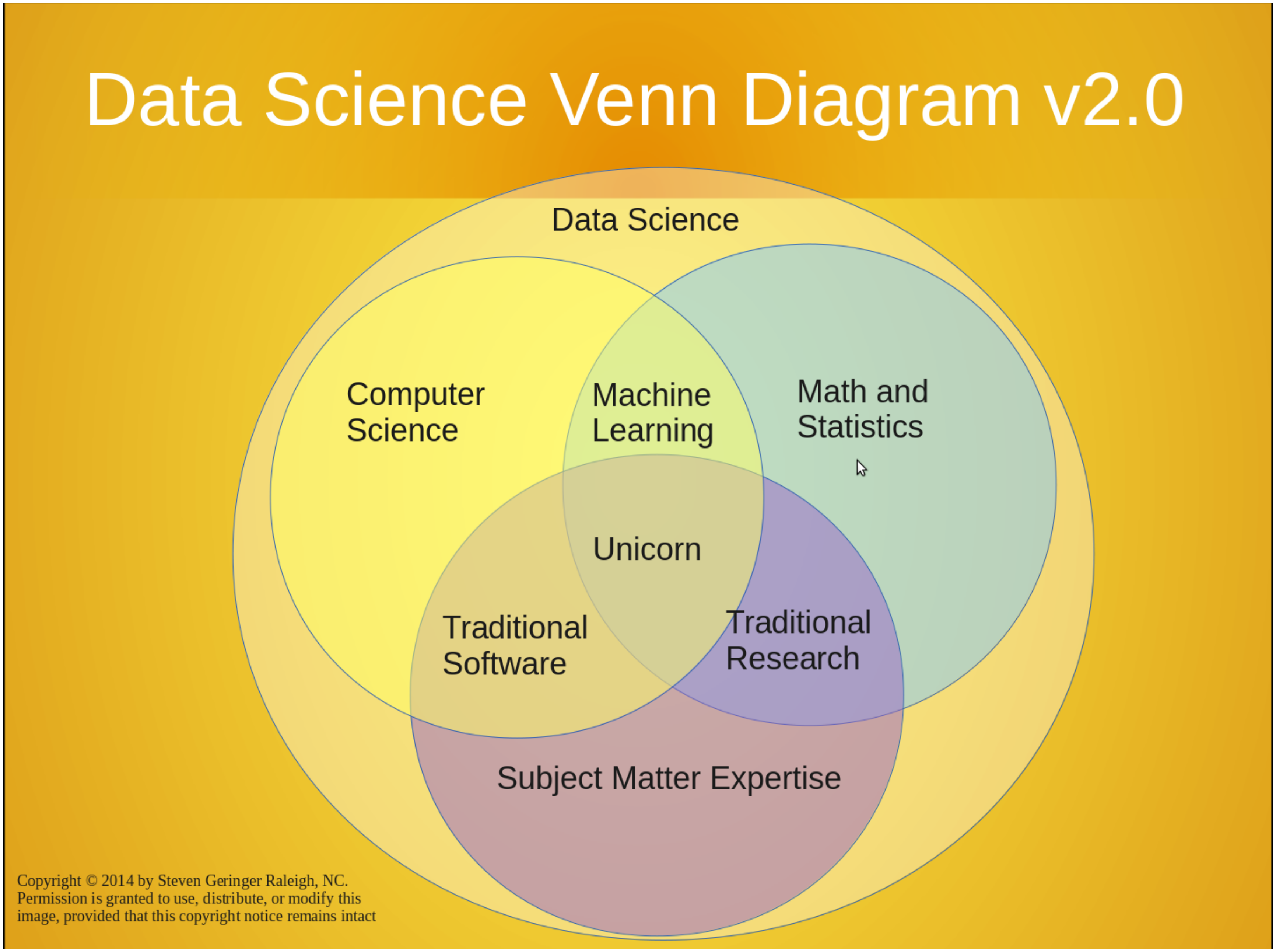
- Enable data driven discovery through machine learning
- From education to chemistry to biology to astronomy to physics to engineered systems like Internet of Things, and more
- Innovations grounded in an education-research-based framework
- Advanced cyberinfrastructure
- An example application: real-time sensing/computation of observational data from the atmosphere, land and water, enhancing our ability to:
  - Detect tornadoes/hazardous weather with pinpoint accuracy
  - Predict accurately storm tracks with real-time data assimilation
  - Warn and respond using data of human activity and context
  - Optimize weather-dependent logistics, transportation, etc.

What is Data Science?

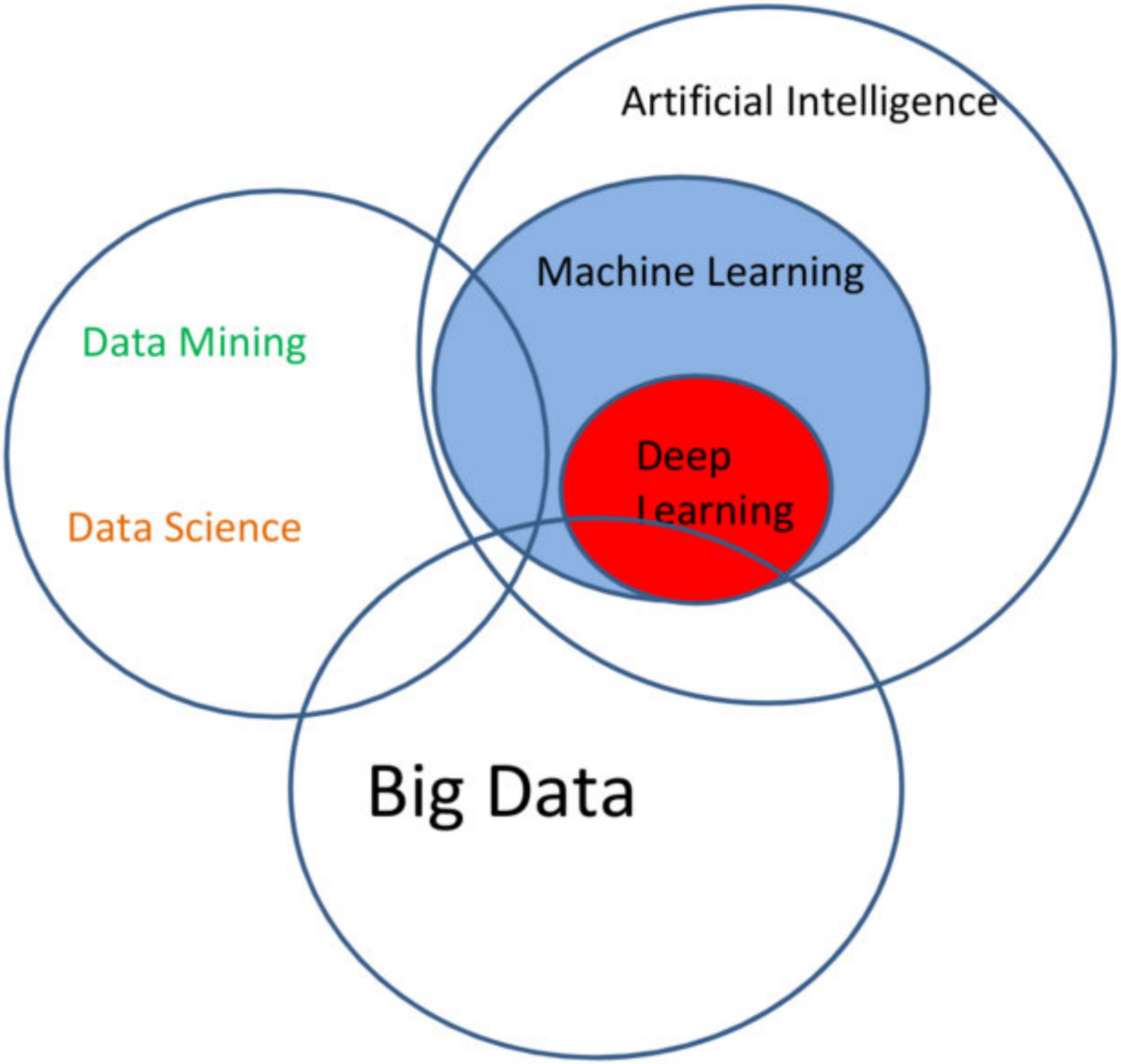
Van Diagram (one version):



Is Everything Data Science?
- Steven Geringer, 2014:



AI-Centric Perspective
- Gregory Piatetsky-Shapiro, 2016:



A Holistic View of Data Science

(David Blei and Padhraic Smyth, PNAS 2017):

Data science is <u>*more than a combination*</u> *of statistics and computer science* - it requires training in how to ***weave statistical and computational techniques into a <u>larger framework</u>***, problem by problem, and to address discipline-specific questions.

Requires:

- understanding the **context** of data
- appreciating the **responsibilities** involved in using private and public data
- clear communication on **what a dataset can and cannot tell**

Practitioner's Perspective on Data Science:

The practice of data science is not just a single step of analyzing a dataset.

Rather, it **<u>cycles</u>** between *data preprocessing, exploration, selection, transformation, analysis, interpretation, and communication*.

A comprehensive treatment (from qualitative to technical):

Cohen M, Guetta D., Jiao K, Provost F. "Data-Driven Investment Strategies for Peer-to-Peer Lending: A Case Study for Teaching Data Science," **Big Data, Sept. 2018**

Practitioner's definition:

Data Science is the <u>study of extracting value from data</u>

Jeannette Wing, Columbia Univ.

Outline

Focus: Structured Learning on Temporal Networks

- ❖ Complex systems perspective

Today: Three Predictive Analytics Topics

- ❖ Examples from Zoran’s lab

Challenges:

1. Large dynamic spatiotemporal networks

2. Network embeddings for outage occurrence prediction

3. Structure-aware intrinsic representation learning of temporal networks for wind power prediction

Data Science in Complex Systems - An Example: Learning to Predict Weather-Related Outages in Transmission

Feb 19, 2019, at 6:45 PM, TUalert Weather Advisory <9ab1be24-0005-3000-80c0-fceb55463ffe@notify2.mir3.com>:

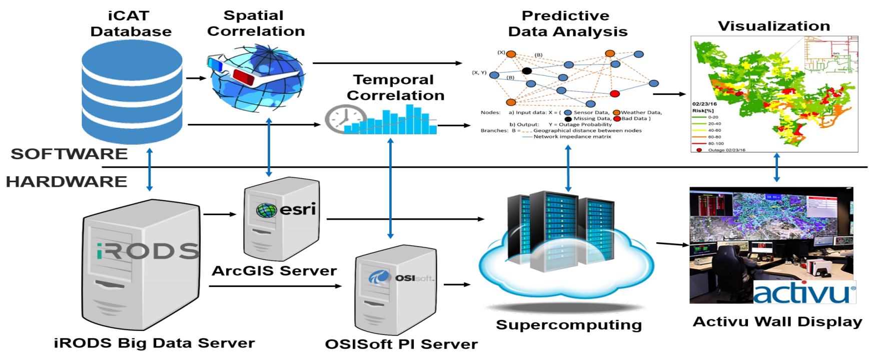
Zoran Obradovic,
Because of the likelihood of severe weather, Temple's U.S. campuses will be closed and classes are cancelled tomorrow, Wednesday, Feb. 20. Only essential employees should report as scheduled. Non-essential employees should not report. Medicine, Dentistry and Podiatry will issue information about clinic schedules. Details at temple.edu
TUalert Weather Advisory

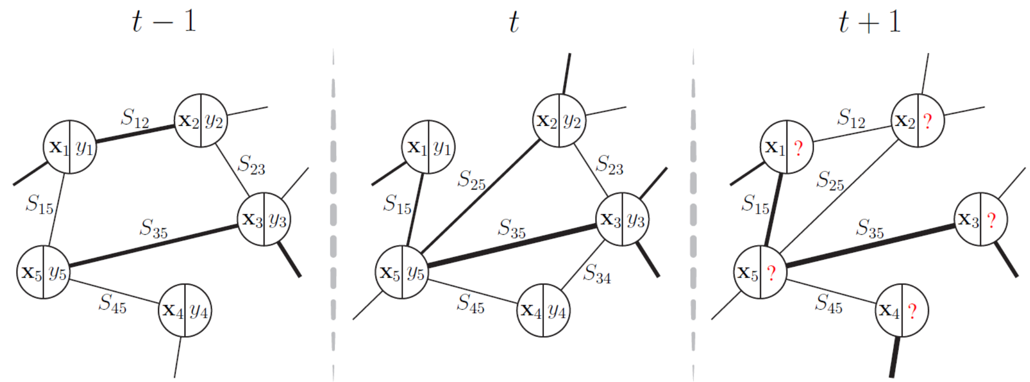
Known : 75% of power outages are weather related

Objective: A **pro-active** maintenance and operation of power system infrastructure upon evolving weather events based on **outage probability estimates**

Learning to Predict Weather-Related Outages in Transmission

Data: integration of Big Data sources related to weather impacts on electric transmission and distribution.





Approach: A *graphical model* is used to predict y at all nodes given x and dependencies temporally observed

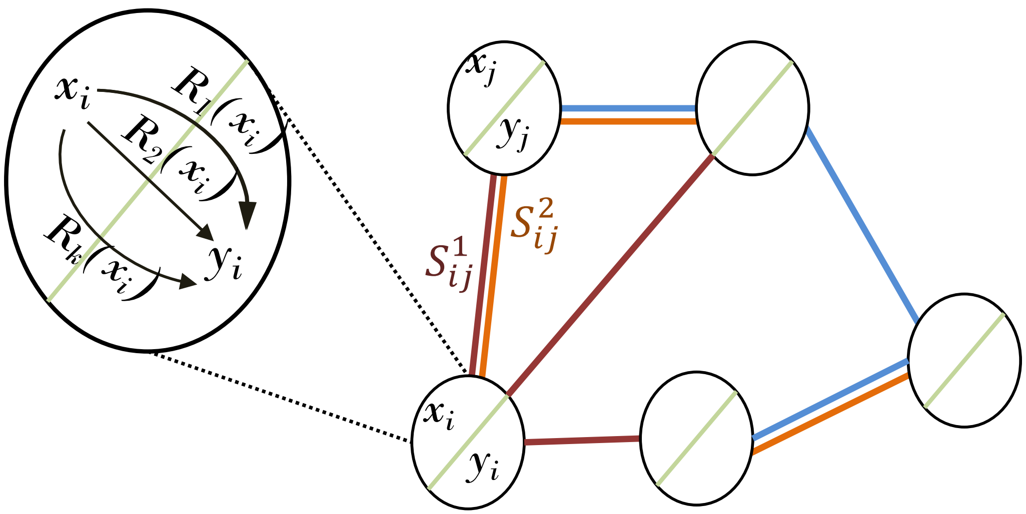
- exploits big data and physical network components together in <u>*time and space*</u>
- capable of <u>*predicting risk*</u> of a transmission line insulation breakdown in case of future lightning strikes (Dokic, et at HICS 2016)

Kezunovic, M., Obradovic, Z., Dokic, T., Roychoudhury, S. “Systematic Framework for Integration of Weather Data into Prediction Models for the Electric Grid Outage and Asset Management Applications,” *HICSS* 2018

CHALLENGE 1: Exploiting Structure

Goal: Prediction of a real valued N-dimensional response y = (y_1, …, y_N), given:

- explanatory variables x = (x_1, …, x_N)
- **dependencies** between the responses y, represented by a set of networks, each describing one of multiple types of connections among the nodes.



- The regression method should be able to take into consideration **structure represented as various linkage relations** among the nodes (weighted connections)
- The connections are of different nature, each offering partial information, so that the contributions should not be averaged and have valuable information lost

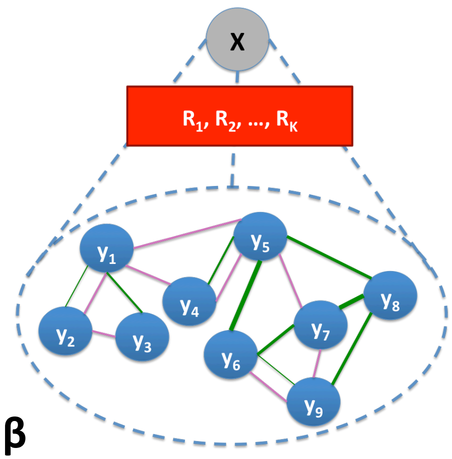
Structured Regression by Gaussian Conditional Random Fields (GCRF)

Given the weighted graph and unstructured predictors R_k **GCRF** learns:

- β: the importance of link weights E_{ij},
- α: the degree of belief towards unstructured predictors R_k

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\psi})} exp(\sum_{i=1}^{N} A(\boldsymbol{x}, \boldsymbol{\alpha}, y_i) + \sum_{j\sim i} I(\boldsymbol{\psi}, y_i, y_j, \boldsymbol{x}))$$

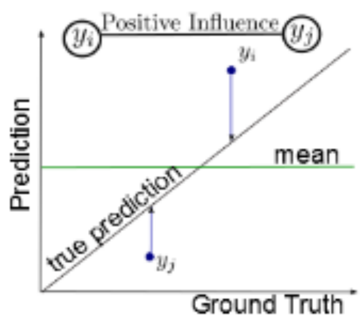
$$A(\alpha, y_i, \mathbf{x}) = -\sum_{i=1}^{N}\sum_{k=1}^{K} \alpha_k (y_i - R_k(\mathbf{x}))^2 \qquad I(\beta, y_i, y_j, \mathbf{x}) = -\sum_{i\sim j} \beta_l E_{ij} (y_i - y_j)^2$$



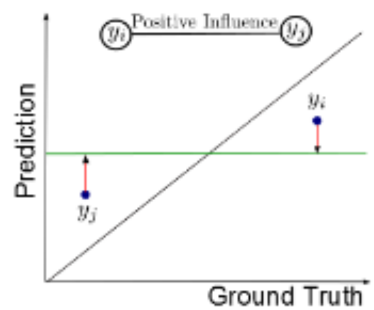
Learning: *Convex optimization* to find association and interaction parameters **α** and **β**

GCRF: Restricts both α and β to positive values to preserve convexity of the search space

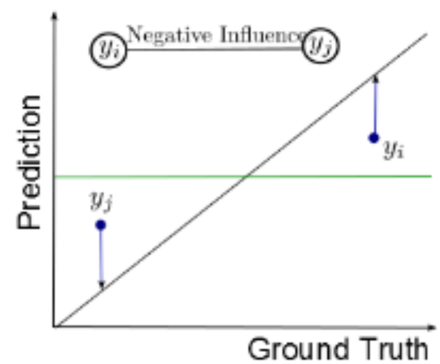
Good



Bad



Solution



PNI-GCRF: Extends GCRF parameter spaces (both α and β) while preserving convexity to allow modeling **positive and negative influences**

Glass, J., Ghalwash, M., Vukicevic, M., Obradovic, Z., "Extending the Modeling Capacity of GCRF while Learning Faster," *Proc. Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16),* Phoenix, AZ, February 2016.

PNI-GCRF Application: Predicting Number of Hospital Admissions by Diseases in California

Data: HCUP SID California EHR database

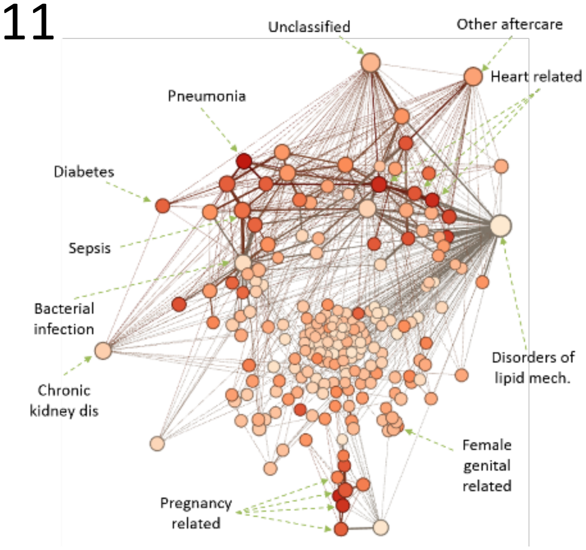
- **Size**: 35,844,800 inpatient discharge records for 19,319,350 distinct patients
- **Period**: 108 months from January 2003 to December 2011
- **Hospitals**: total 474 hospitals

Graphs: 108 monthly comorbidity graphs

- Nodes = 253 classes of diseases (CCS codes);
- Edges = correlations between number of admissions

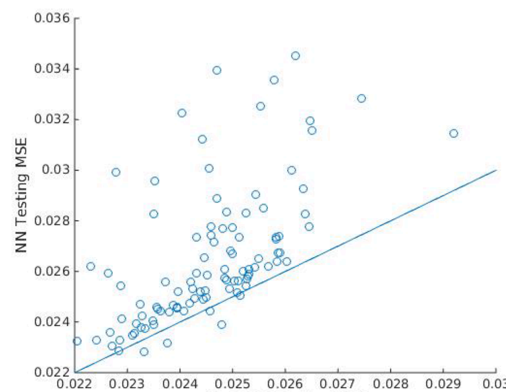
Training: 80 months (Jan. 2003 – Aug. 2009)

Test: 28 months (Sept. 2009 – Dec. 2011)



PNI-GCRF Results:

- PNI-GCRF was ***more accurate than deep learning model*** (see Figure)
- PNI-GCRF was also significantly ***better than GCRF*** (more accurate in 24 of 27 months)
- Using PNI-GCRF we ***found diseases that are negatively related to other diseases***



MSE for PNI-GCRF vs NN,

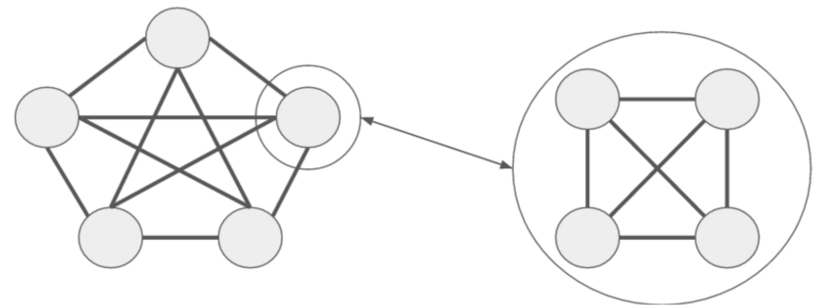
Structured Regression in Multi-Scale Networks (MSN-GCRF)

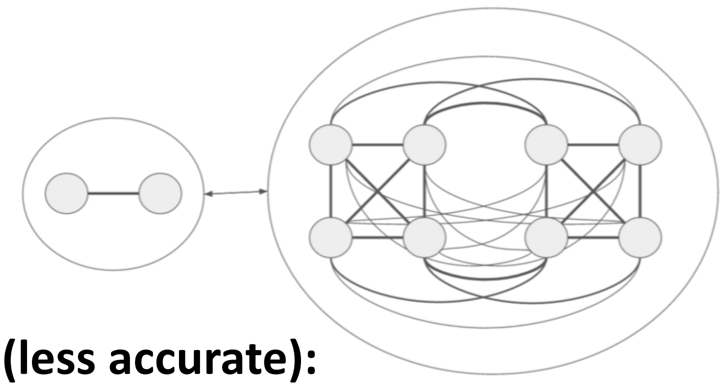
Application: *Predict monthly admissions for each disease* ***for each hospital*** in California

Data: About 36 million hospitalization records over 9 years

Nested network representation:

- Hospitals are nodes in a network
- Each node is a network of comorbidities at a single hospital





Problem:

This graph is huge (about **million nodes and trillion links**) while **GCRF** computational complexity **is O(N^3)**

SOLUTION: Convex optimization on a ***Kronecker product of matrices***

(we derived a theorem to compute Laplacian of a Kronecker product efficiently)

Result: ***logarithmic learning time and memory*** compared to naïve implementations.

Baselines (less accurate):

Neural Network learning needs 7 hours

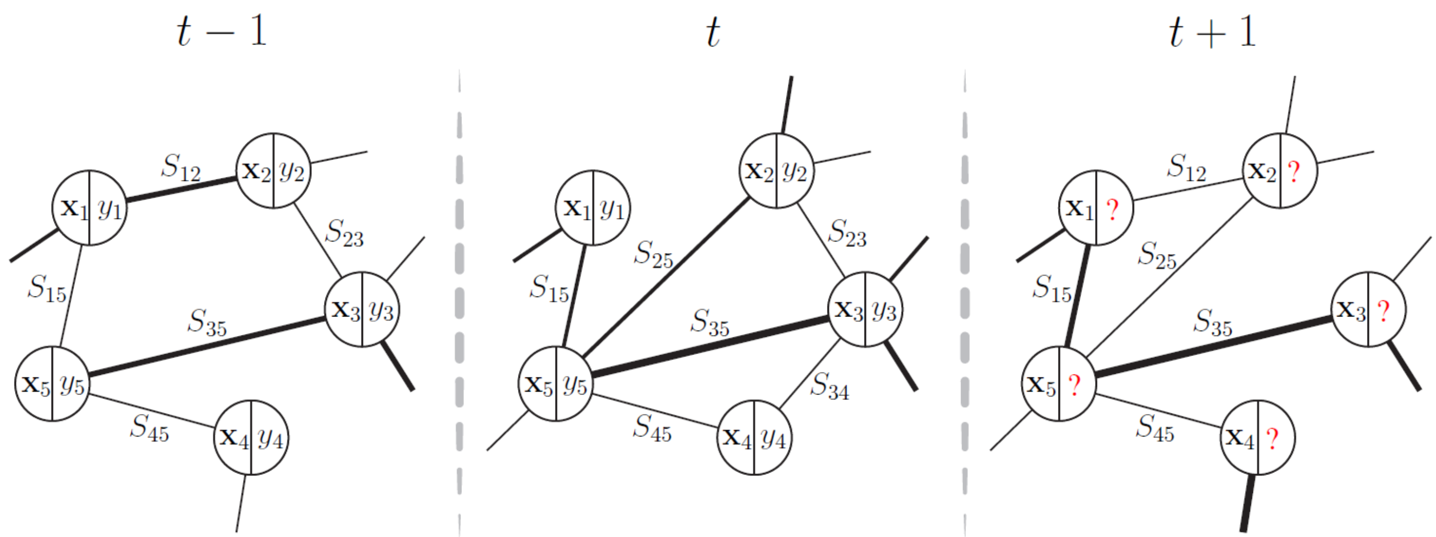
Vector Autoregression learning needs 6.9 days

Structured regression (more accurate):

MSN-GCRF	FE-GCRF	FF-GCRF	GCRF
10 minutes	1.1 weeks	1 week	2 months

Glass, J., Obradovic, Z. "Structured Regression on Multi-Scale Networks," *IEEE Intelligent Systems*, 2017

Structured Regression in Evolving Networks



- **Graphical Models:**
  - Commonly used to predict the response at each node in one or multiple upcoming time steps.
  - Retrained at each step

- **Challenges:**
  - Accumulating error in multi-step ahead prediction
  - Time for prediction is limited

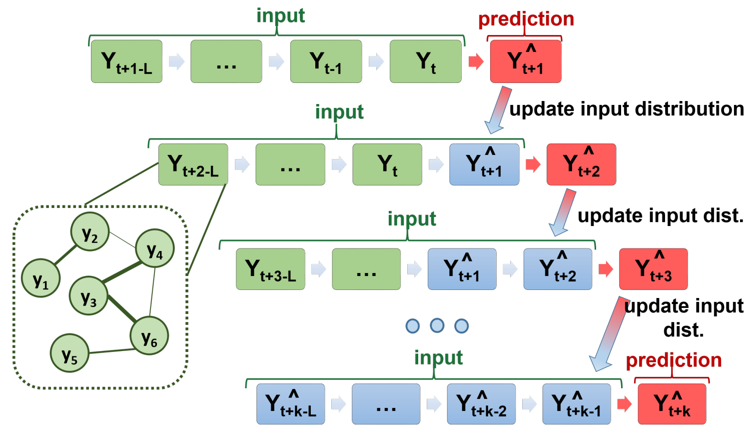
Uncertainty Propagation in Long-term Structured Regression on Evolving Networks

Motivation: *Long-term prediction* of the state of networks (structured temporal regression), with application to disease-disease networks

How: Incremental multi-step-ahead prediction relying on *previous predictions used as uncertain (noisy) inputs*

Challenge: Account for accumulating error

The idea: Take into account distribution of noisy input variable, x_*



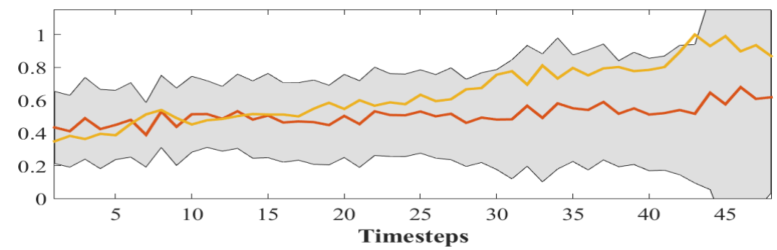
Objective: Long-term prediction of monthly **admission rate for Septicemia** in California hospitals

Graph: Comorbidity disease network in monthly scale:

Nodes: 260 primary diagnoses (CCS codes)

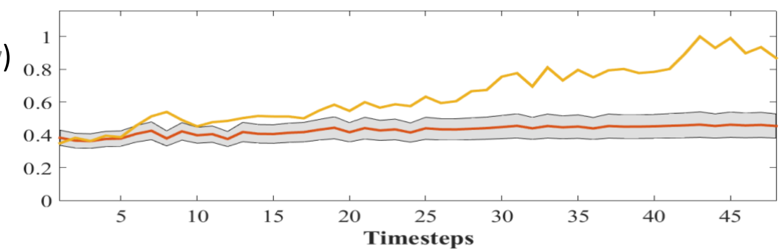
Edges: learned by GCRF as relationships of hospitalization rates for 2 CCS codes

Training: 60 months; **Test:** 48 remaining months



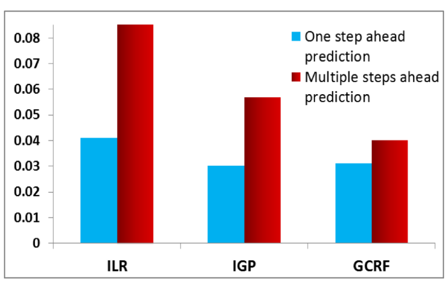
(a) GCRF

Predictions (red) and ***uncertainty estimates*** (gray) of GCRF and Iterative Gaussian Process (IGP) for **Septicemia admission rate** (orange)



(b) IGP

One month (blue) and 48 months (red) prediction of admission rate (MSE) **on all diseases**

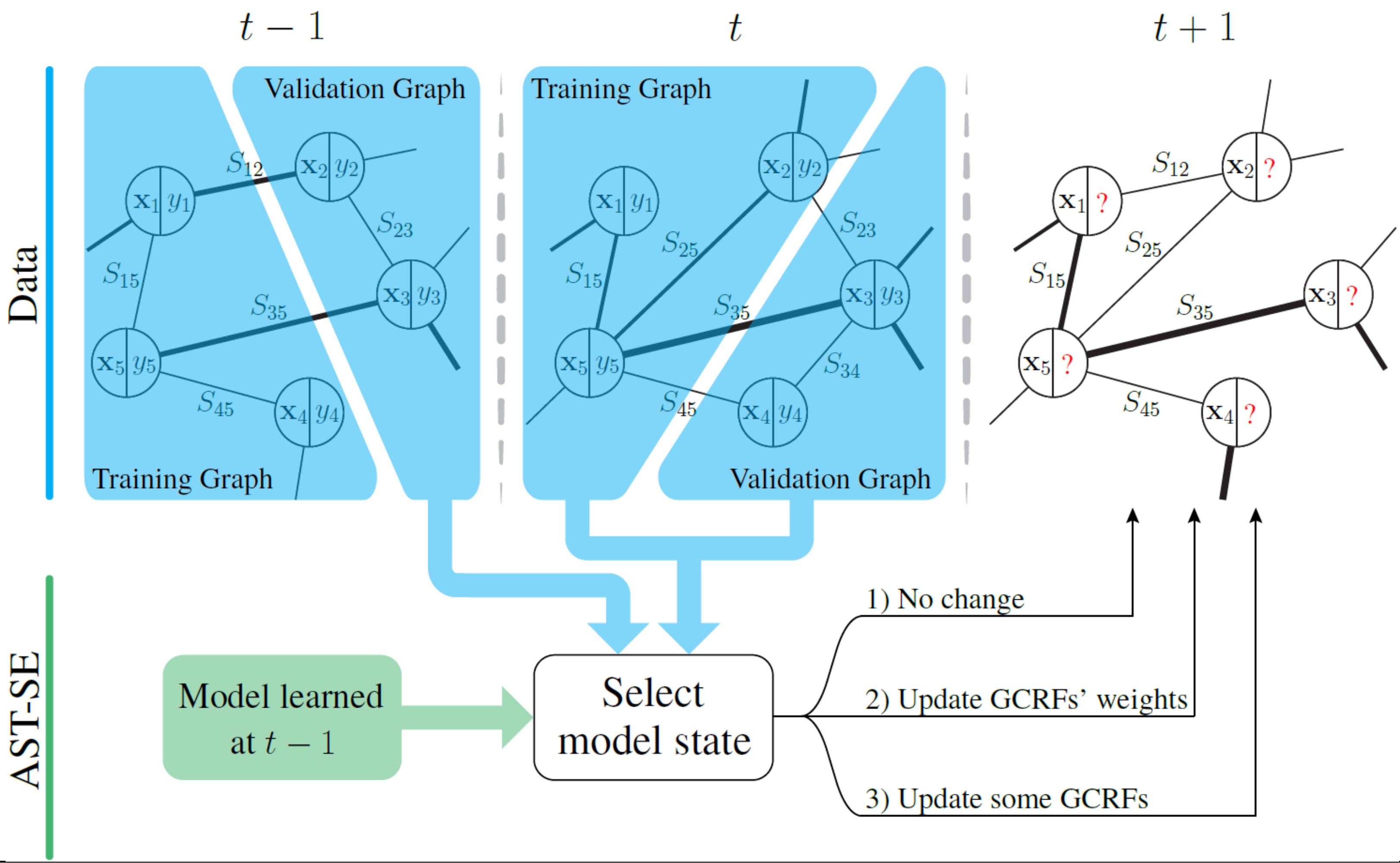


- **GCRF** is capable of properly propagating uncertainty when model is making mistakes
- **IGP** predictions make huge mistakes, however uncertainty is small which is wrong

Gligorijevic, Dj, Stojanovic, J., Obradovic, Z."Uncertainty Propagation in Long-term Structured Regression on Evolving Networks," *Proc. Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16),* Phoenix, AZ, February 2016.

Adaptive Skip-Train Structured Ensemble (AST-SE)

- *Avoid repetitive training* by: (1) employing **multiple graphical models** to learn different relationships and (2) **detecting changes** in a network once they occur and (3) **adapting accordingly**



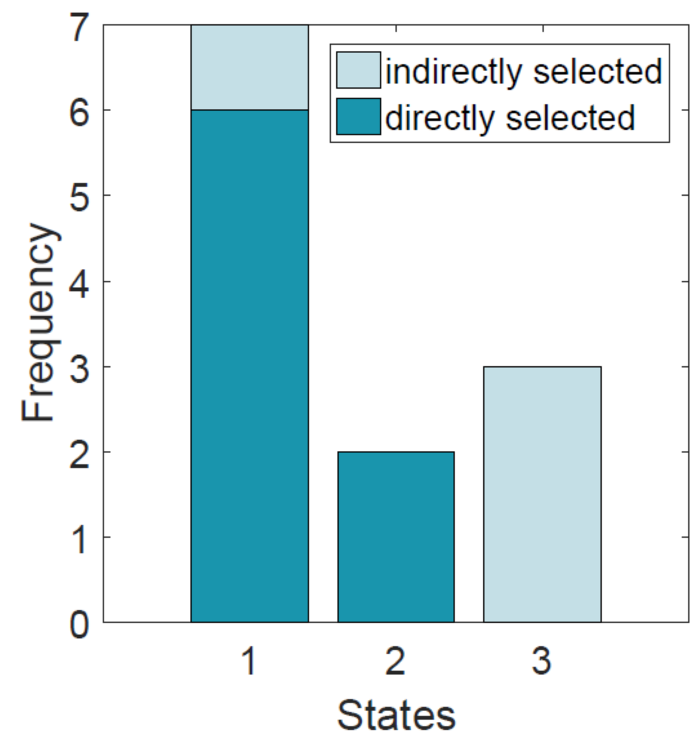
Pavlovski, M., Zhou, F., Stojkovic, I., Kocarev, Lj., Obradovic, Z. "Adaptive Skip-Train Structured Regression for Temporal Networks," *Proc. European Conf. Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, September 2017

AST-SE Application:
Influenza Virus Network Prediction

- **Data:** Infuenza A virus subtype H3N2 network observed over time (16 hours/steps)
- **Nodes:** 12,032 genes
  - Features: expression values from 3 previous time steps
  - Targets: expression values at the current time step
- **Structure:** similarities between gene expressions
- **Task:** *Predict texpression values at the next time step*

Model	MSE	Execution time
LR	0.38 ± 0.19	$\mathbf{0.10 \pm 0.03}$
GCRF	0.39 ± 0.21	9082.71 ± 1898.43
SE	0.39 ± 0.21	297.29 ± 19.42
WSE	0.35 ± 0.19	309.32 ± 19.44
AST-SE	$\mathbf{0.23 \pm 0.07}$	64.00 ± 45.73

- **34-41% more accurate than alternatives**
- **140 times faster than GCRF**



Selected AST-SE states

Learning Dependency Structure in a Network: Fast GMRF Learning of Sepsis Co-expression Network

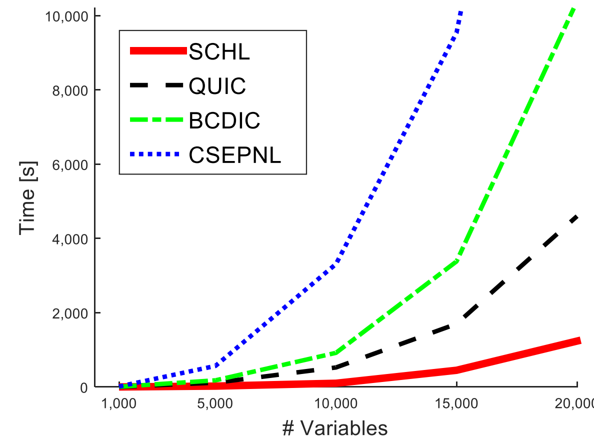
- **Fast SHCL method:** decompose Precision Matrix into a product of two Cholesky Factors and impose L1 penalty on the approximation

- **Gene expressions data:** 24,840 variables from 163 septic subjects

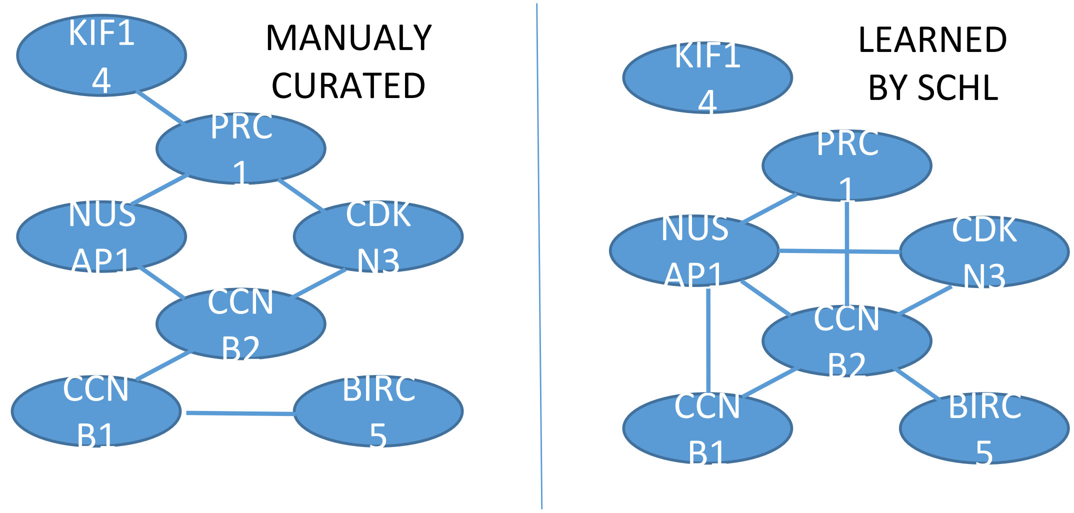
- **Graph learned by SCHL**: ~170,000 edges

- **Manually curated sepsis co-expression network**: 7 connections among 7 sepsis related genes

- **Sepsis co-expression network discovered by SCHL:** 8 connections where 4 overlap with manual curation



SCHL - **Much faster GMRF learning** of dependency structure



	SCHL	QUIC	BCDIC	CSEPNL
Time [s]	**5,038.6**	10,011.0	15,929.1	26,665.7

Stojkovic, I., Jelisavcic, V., Milutinovic, V., Obradovic, Z. "Fast Sparse Gaussian Markov Random Fields Learning Based on Cholesky Factorization," *Proc. 26th International Joint Conference on Artificial Intelligence (IJCAI)*, Aug. 2017

CHALLENGE 2: Graph Representation Learning by Node Embedding

Limitation of G=(V,E) representation for exploiting structure in large networks:

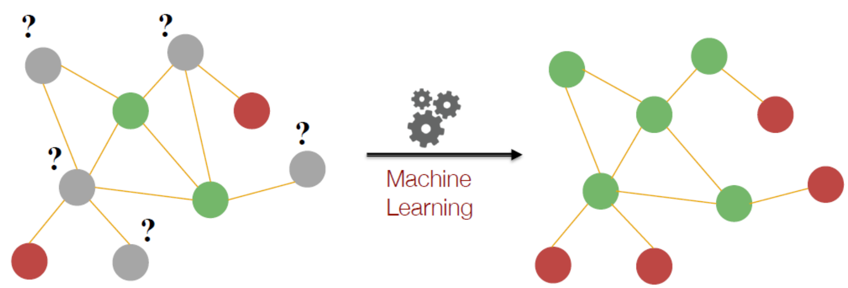
- **Relationships are represented <u>explicitly</u> using a set of edges**
- **Curse of dimensionality (large sparse matrix)**

- **Inapplicability of machine learning methods**

  - Nodes (examples) are **dependent** on each other

  - Off-the-shelf ML methods require examples to be represented by **independent vectors**

Node classification

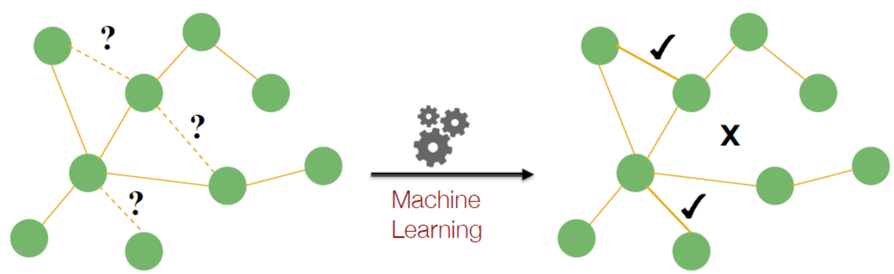


- **High computational complexity**

  - **Low parallelizability**

  - Nodes are coupled <u>explicitly</u> by E

Link prediction



Jure Leskovec, Stanford CS224W: Analysis of Networks, http://10/23/18 cs224w.stanford.edu

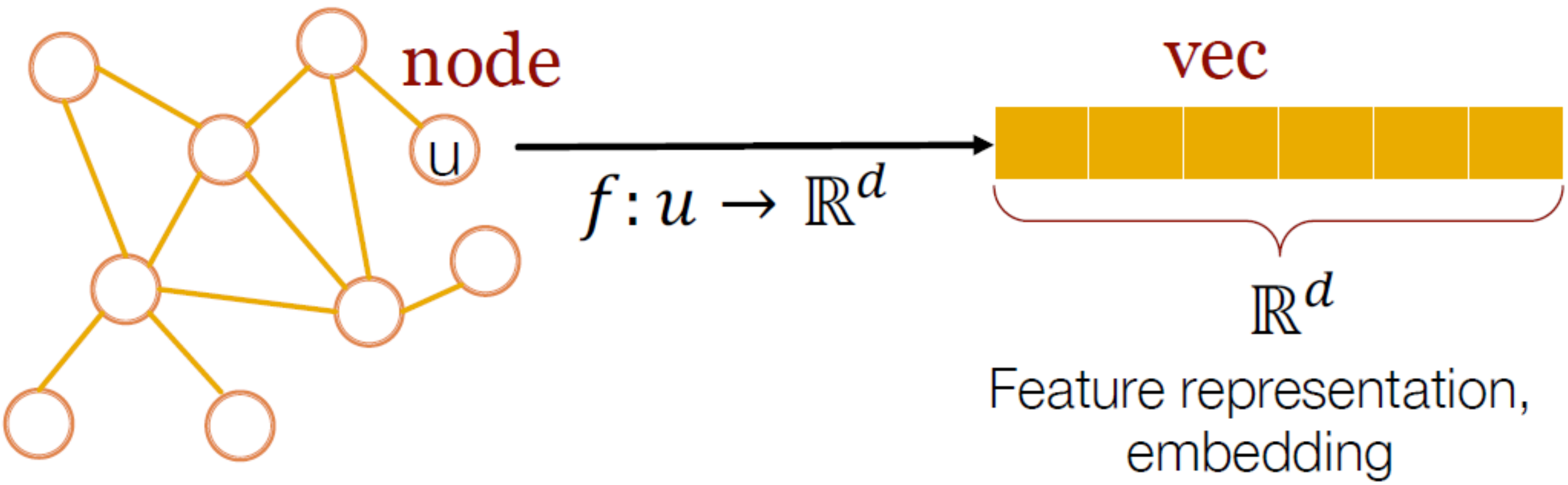
Solution: Network Embedding

- Assign nodes to **low dimensional** representations that effectively preserve the network structure

- **Relationships** among the nodes are captured by the **distances** between their vectors in the embedded space

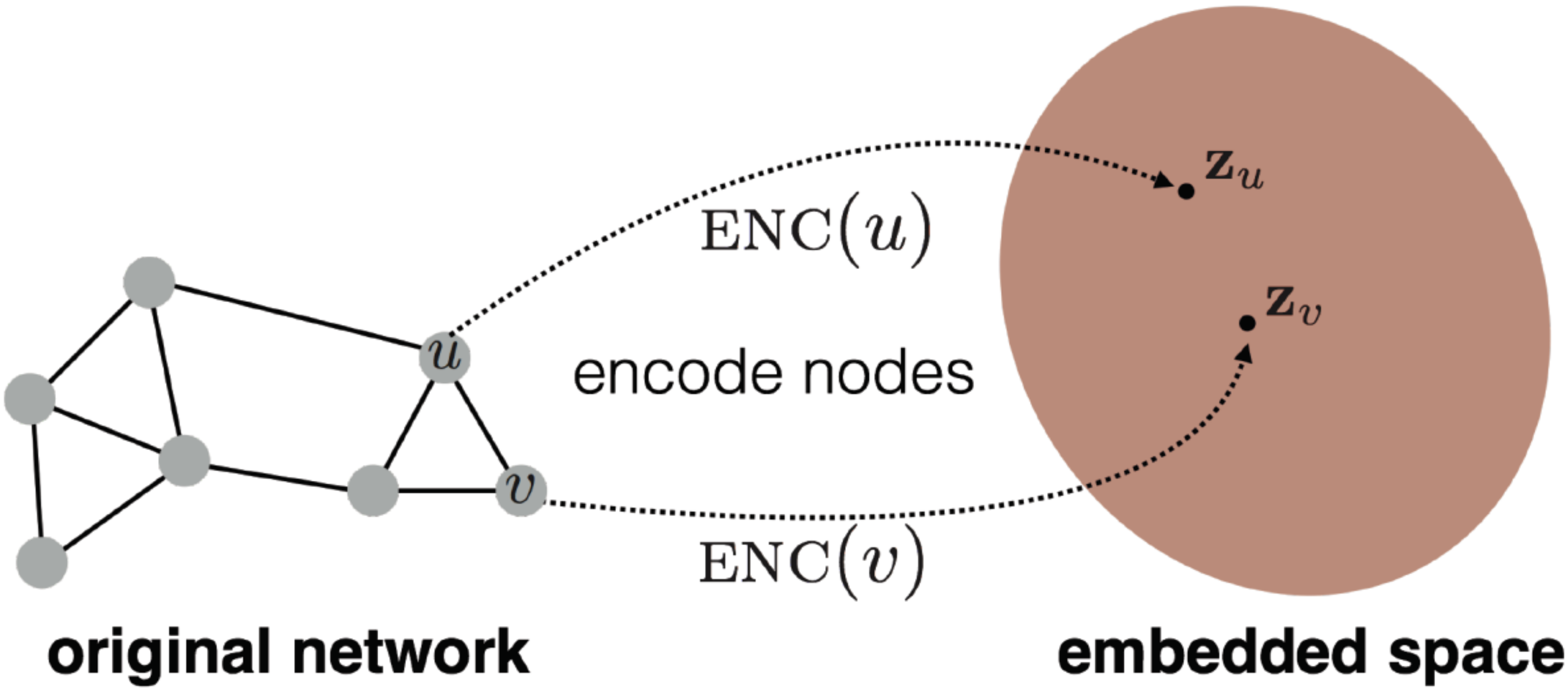
- **Embedded representations can be learned for:**
  - nodes, edges, even an entire network

Node Embedding



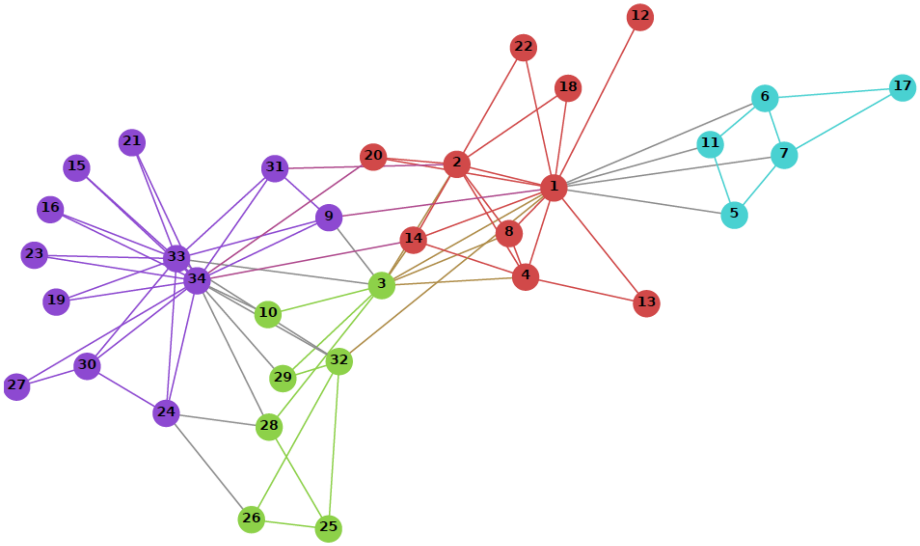
Jure Leskovec, Stanford CS224W: Analysis of Networks, http://10/23/18 cs224w.stanford.edu

Node Embedding - The Main Idea

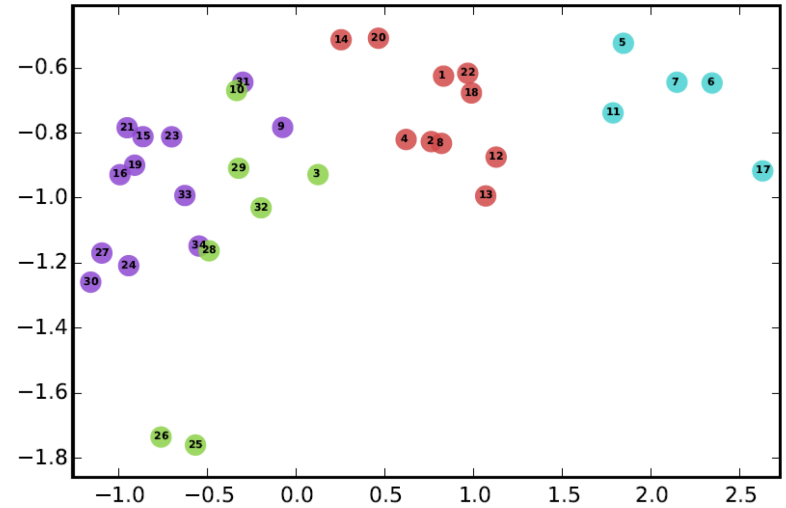


Jure Leskovec, Stanford CS224W: Analysis of Networks, http://10/23/18 cs224w.stanford.edu

Node Embedding - Example



(a) Input: Karate Graph



(b) Output: Representation

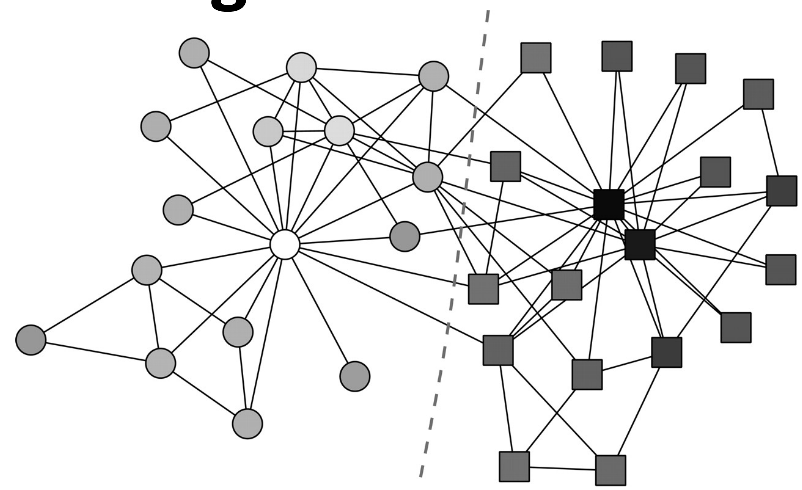
Node Embedding - Advantages

Dense, continuous, and low-dimensional representations of nodes

Therefore:

- **Noise** or **redundant information** can be **reduced**
- Intrinsic **structure information** can be preserved
- Nodes are **not coupled** anymore
- Main-stream **parallel** computing solutions for **large-scale** network analysis

Example: Modularity-based Embedding



- **Modularity matrix of a graph G:**

$$B_{ij} = A_{ij} - \frac{d_i d_j}{2m} \;,$$

where

A_{ij} – (i, j) entry in G's adjacency matrix,

d_i – degree of node i,

m – total number of links.

- The top K eigenvectors of $\boldsymbol{B}$ are used to embed the nodes in G

Newman, Mark EJ. "Modularity and community structure in networks." *PNAS 2006.*

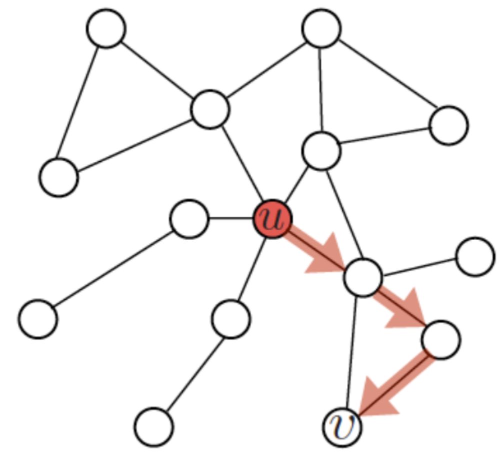
Another Example: DeepWalk-based Node Embedding

- **Generalizes** recent advancements in NLP and unsupervised feature learning (or deep learning) from sequences of **words to graphs**

- Uses local information obtained from truncated random walks
  - treats **walks as sentences**

- **Trivially parallelizable**

- Application: multi-label network classification for social networks

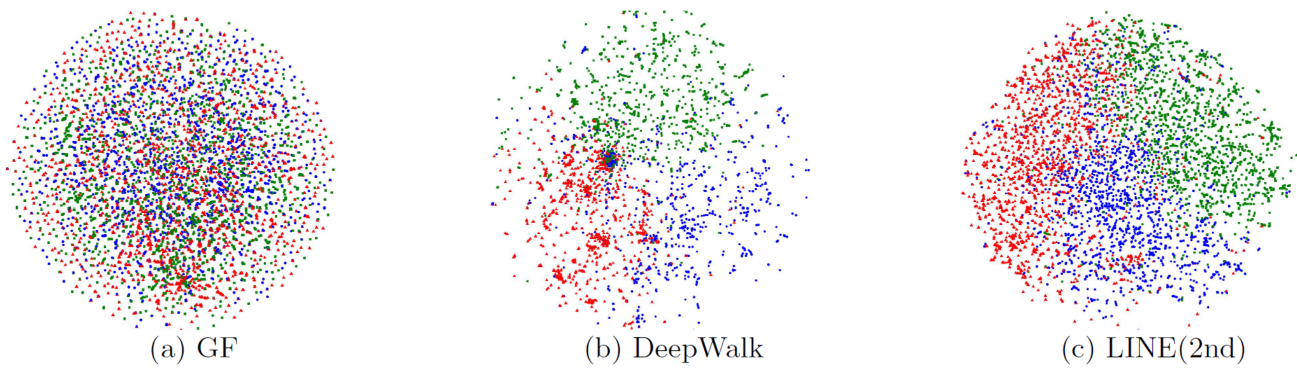


Perozzi, et al. “DeepWalk: Online learning of social representations.” *SIGKDD 2014.*

Another Example: LINE-based Node Embedding

- Suitable for **arbitrary** types of information networks
- Optimizes a carefully designed objective function that preserves both **local** and **global** network structures
- **Very efficient:** millions of vertices and billions of edges in a few hours

Visualization of a co-authorship network



Hubs are clustered tightly into the **center area**

LINE is not sensitive to hubs

Tang, et al. “LINE: Large-scale Information Network Embedding.” *WWW 2015.*

Node2vec – based Node Embedding

- **Scalable** Feature Learning for Networks

- Learns embeddings that maximize the likelihood of preserving **neighborhoods of nodes**

$$\max_f \sum_{u \in V} \log Pr(\underbrace{N_S(u)}_{\substack{\textbf{neighborhood} \\ \text{of node } u}} | \underbrace{f(u)}_{\substack{\textbf{embedding} \\ \text{of node } u}})$$

- Utilizes a biased **random walk procedure**, which efficiently explores diverse neighborhoods

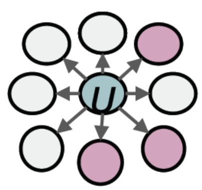
- **Flexible** notion of a node's neighborhood

- **Generalizes prior work!**

Grover, Aditya, and Leskovec "node2vec: Scalable feature learning for networks," *SIGKDD 2016.*

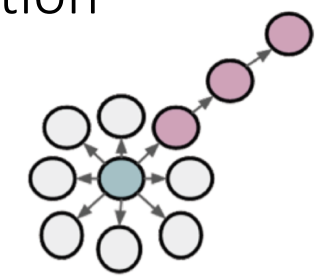
Node2vec Node Embedding

Neighbourhood definition



BFS:
Micro-view of neighbourhood

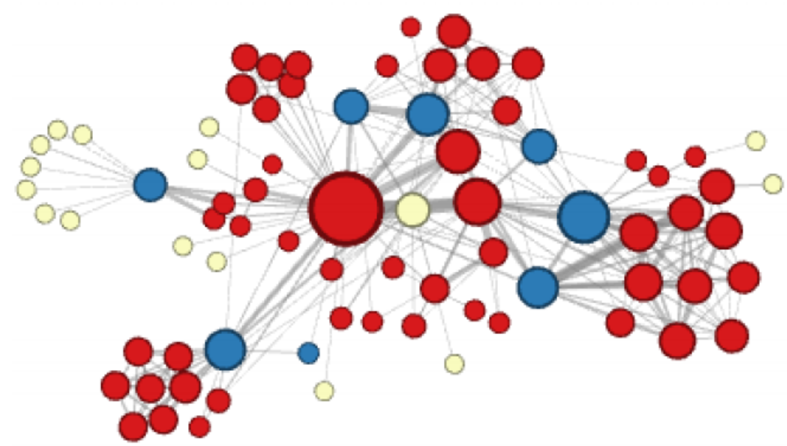
Captures structural equivalence



DFS:
Macro-view of neighbourhood

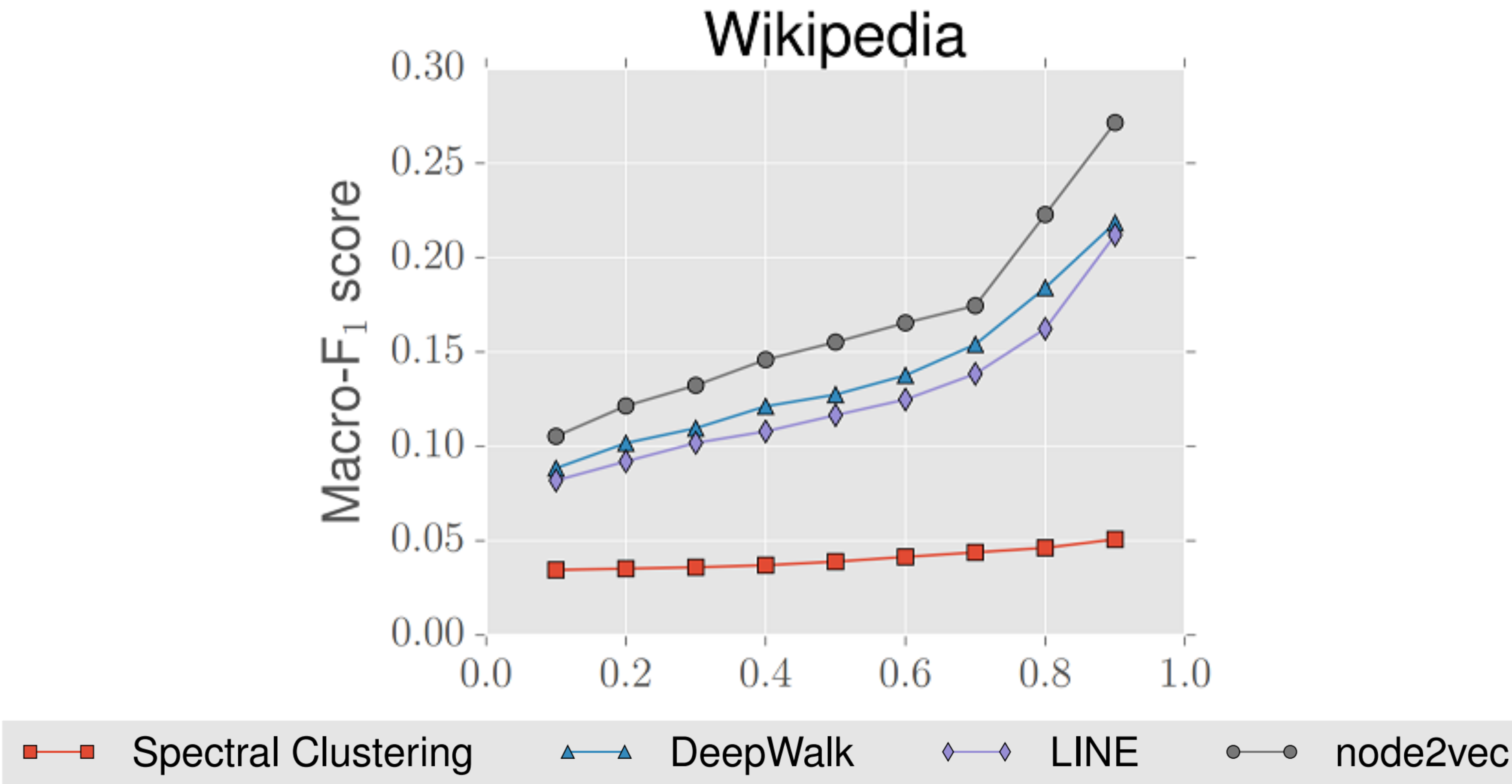
Captures homophily

DFS



BFS

Grover, Aditya, and Jure Leskovec. "node2vec: Scalable feature learning for networks." *SIGKDD 2016.*



- Node2vec **outperforms** alternative models for all **fractions of labeled data.**

Grover, Aditya, and Leskovec "node2vec: Scalable feature learning for networks," *SIGKDD 2016.*

What if a network evolves over time? *Dynamic Network Embedding*

- **Goal:** Learn ***time-preserving embeddings*** that maximize

$$\max_{f} \log \Pr\Big(\underbrace{W_T = \{v_{i-\omega}, \cdots, v_{i+\omega}\}}_{\text{temporal context window}} \setminus v_i \mid f(v_i) \Big)$$

- Utilize ***temporal random walks*** to explore nodes' neighborhoods:

> *Each node v in a* ***valid*** ***temporal walk*** *sequence must* ***temporally succeed*** *(i.e.* ***exist in time after****) every node that precedes it in the sequence.*

Nguyen, *et al*. "Continuous-time dynamic network embeddings," *WWW 2018.*

Valid Temporal Random Walks

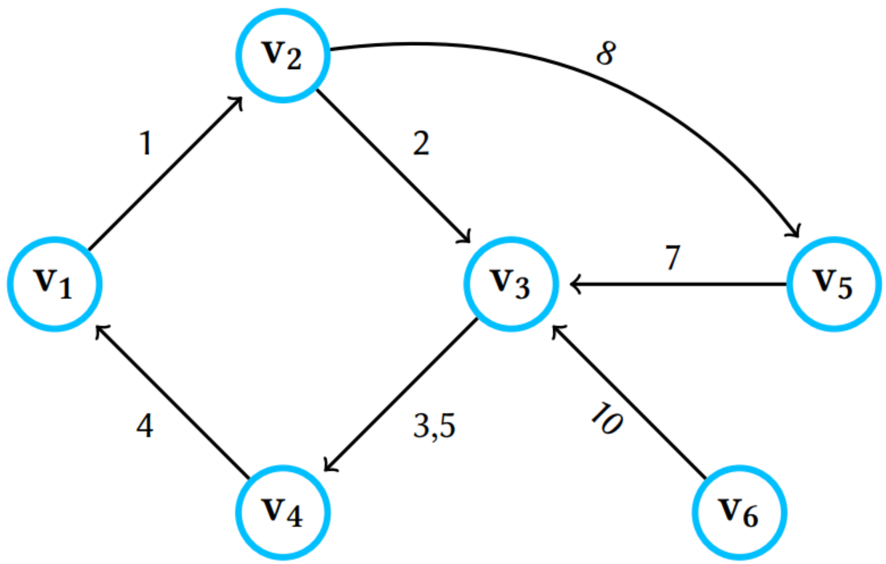


Figure 1: Dynamic network. Edges are labeled by time. Observe that v_4, v_1, v_2 is not a valid temporal walk since v_1, v_2 exists in the past with respect to v_4, v_1.

Nguyen, et al. "Continuous-time dynamic network embeddings." *WWW 2018.*

APPLICATION: Real-time Outage Prediction Mapping

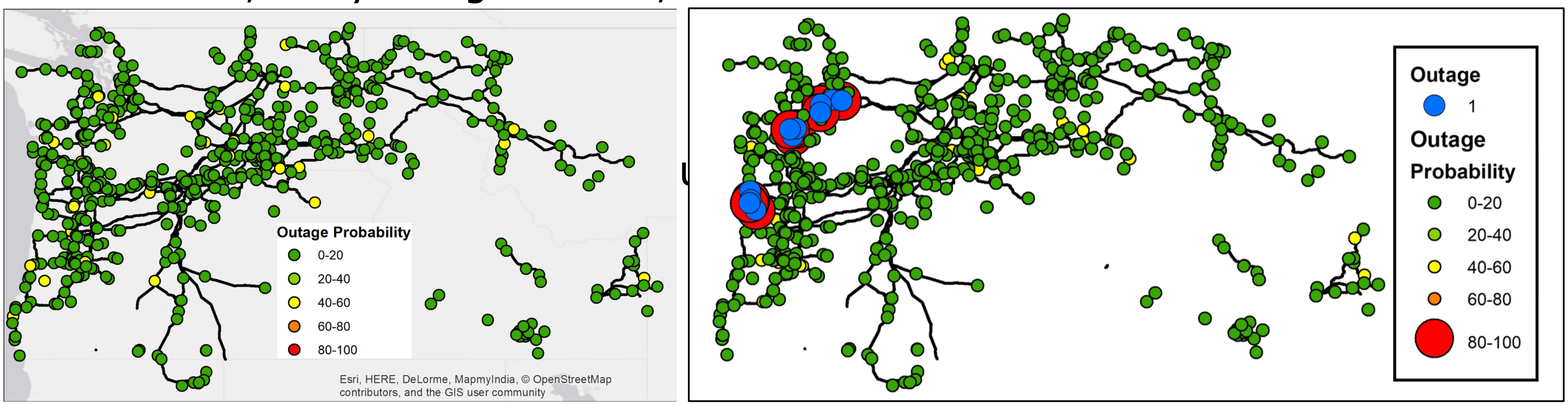
Collaborative Logistic Ensemble Classifier

+ utilized distance correlation to **balance underfit/overfit** [Pavlovski et al, *IJCAI 2018*]

+ accounted for generalization performance

+ learned from **spatial substructures**

+ data: GIS, utility outage records, weather measurements and forecast



Probabilities of outages estimated by CLEC when: no outages occurred (left), and outages were caused by lightning (right).

- **No outages occurred** ⇨ outage probabilities are **smaller than 60%** for all substations
- **Outages occurred** ⇨ the area around the outages has points with probability **over 80%**

Dokic T, Pavlovski M, Gligorijevic Dj, Kezunovic M, Obradovic Z, *HICSS 2019*

Outage Occurrence Prediction

Experimental Setup

- **Training**: data from 1999 to 2010
- **Prediction** horizon: 2010-2018
- Substations were **embedded into a 50-dimenstional space** based on their **spatial proximity**
- CLEC was run with $M = 30$ components
- $\eta = 30\%$ of the training data were sampled to construct the subset for each LR component

Model	Acc.	AUC	F1	Bias
LR	0.8467	0.9278	0.8097	0.6821
LR (spatial)	0.8624	0.9292	0.8242	0.7075
CLEC	**0.8919**	**0.9313**	**0.8532**	**0.7685**

Prediction performance w.r.t. different evaluation metrics.

Discussion

- LR (spatial) obtained greater classification performance compared to LR
  - ⇨ supports the hypothesis **that spatial information is truly relevant for this task**
- **CLEC outperforms its alternatives**, yielding higher values for accuracy, AUC and F1
- Large lift in Bias
  - ⇨ shows the **benefit of using a subsampling-based ensemble scheme**

Performance Variability Across Seasons

- **CLEC consistently outperformed LR and LR (spatial)**

  ~0.25-9.5% and **~0.33-6.2%** more accurate

- **Improvements** in AUC and F1 **in 3 out of 4 seasons**

- **CLEC ameliorates Bias across all seasons**

- **Largest improvements** were achieved for the **Winter** season, while the **smallest** ones were recorded for the **Summer** season

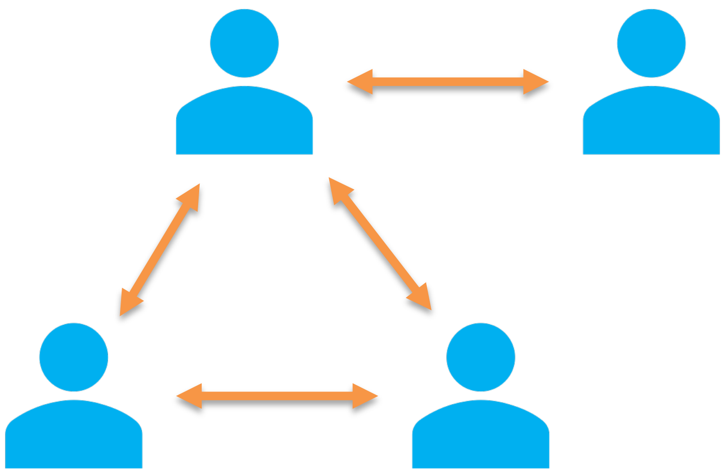
⇨ Reflects the **volatility of the climate conditions** in the **Pacific Northwest region**

Model	Acc.	AUC	F1	Bias
Winter				
LR	0.9089	0.8358	0.7340	0.5862
LR (spatial)	0.9176	0.8451	0.7533	0.6272
CLEC	**0.9305**	**0.8634**	**0.7803**	**0.7128**
Spring				
LR	0.8597	0.9361	0.8221	0.6687
LR (spatial)	0.8792	0.9325	0.8419	0.6932
CLEC	**0.9164**	**0.9363**	**0.8822**	**0.7463**
Summer				
LR	0.7849	0.8860	**0.8770**	0.8540
LR (spatial)	0.7841	0.8843	0.8753	0.8613
CLEC	**0.7874**	**0.8914**	0.8766	**0.8851**
Autumn				
LR	0.8132	0.8906	0.6855	0.5130
LR (spatial)	0.8462	**0.8967**	0.7211	0.5429
CLEC	**0.9080**	0.8874	**0.7961**	**0.6312**

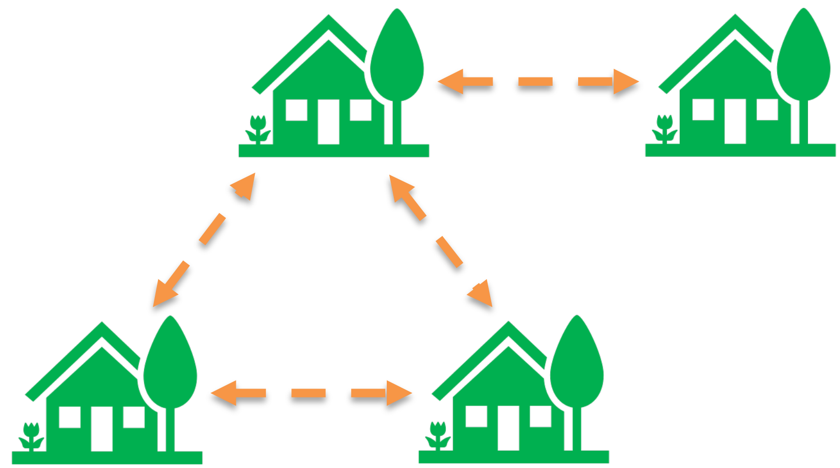
Prediction performance across different seasons.

CHALLENGE 3:
Implicit Attributed Temporal Graph Representation Learning

Deterministic Graph:
A social network



Implicit Graph:
A network of farms



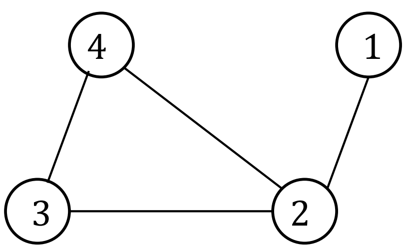
Edges are deterministic in social network. E.g., friendship connection

Edges are implicit in the network of farms. They are <u>decided by prior knowledge</u>. E.g., similarity of associated attributes.

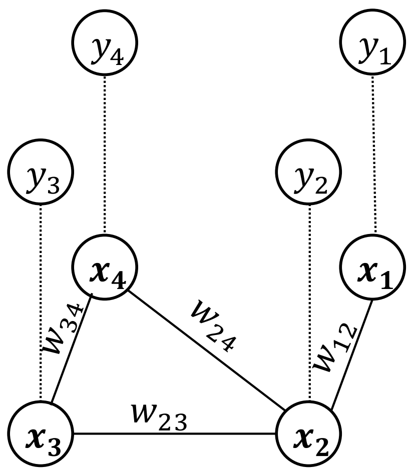
Implicit Attributed Graph

- $(y_i, \boldsymbol{x}_i)$ – node i is composed of a target variable and a vector of attributes
- w_{ij} - edge between node i and node j, determined by prior knowledge

Graph

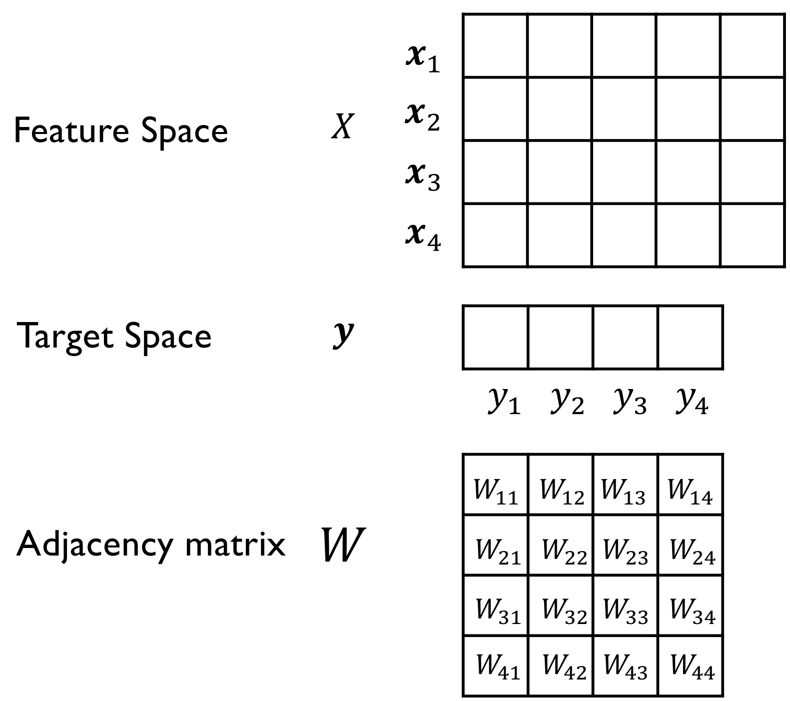


Attributed Graph



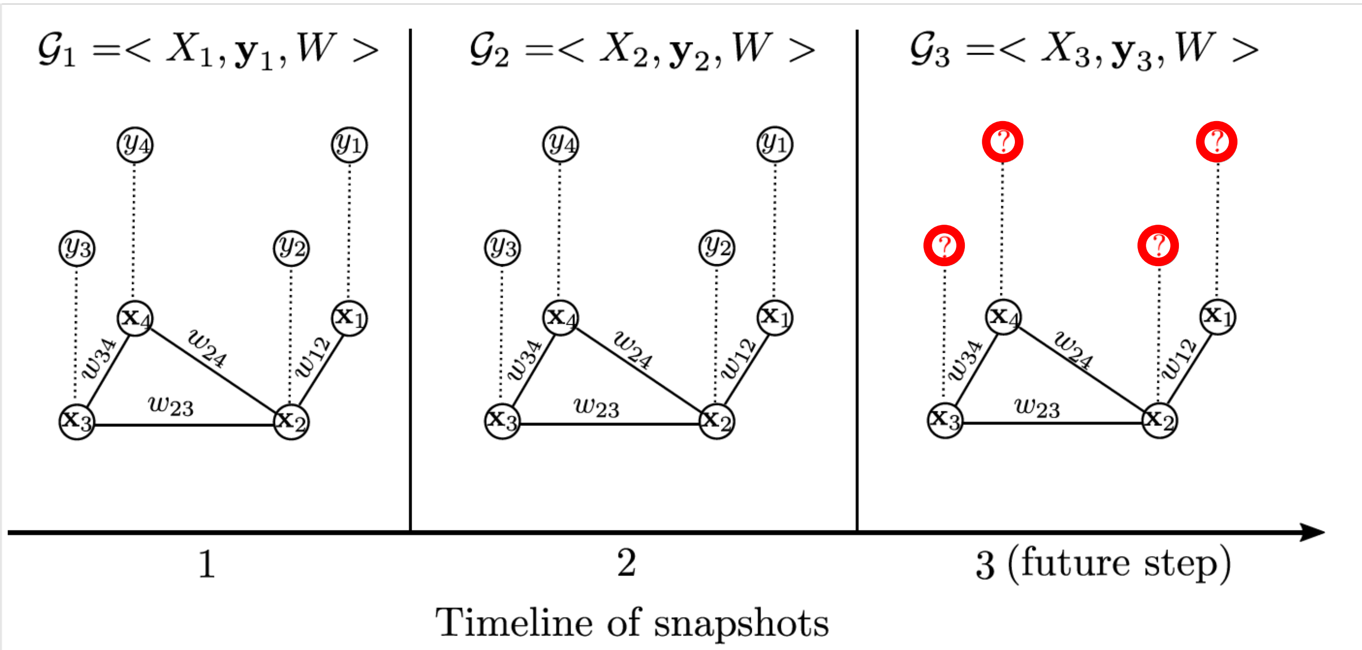
Matrix Representation

An example with 5 attributes in each node

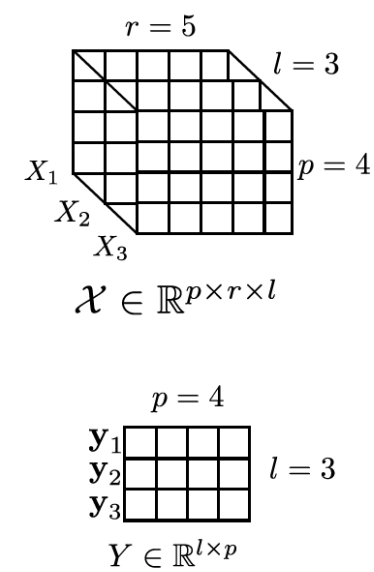


Temporal Graph Regression

Goal: Predict target variables y_i at future time step



Matrix view of the temporal graph



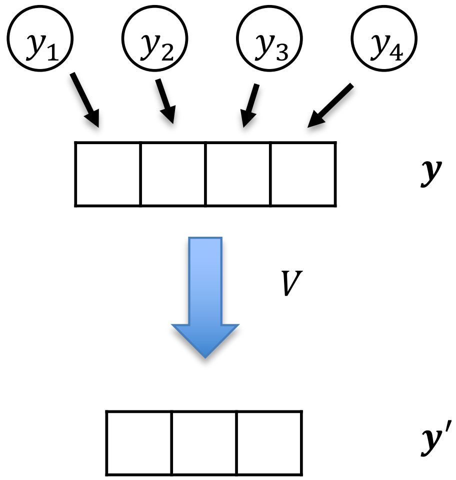
\# snapshots: l
\# nodes: p
\# features in node: r

Graph Representation Challenges

- **Number of variables in the target space: $\boldsymbol{O(l * p)}$**
  - Information contained in the target space is redundant
  - Model complexity is positively correlated to the number of variables

- **Existing Solutions**
  - Reducing the variables in the target space
  - Learning a more compact latent target space

PTST: Principal Target Space Transformation



The target space of a graph

Linear compression via a matrix V

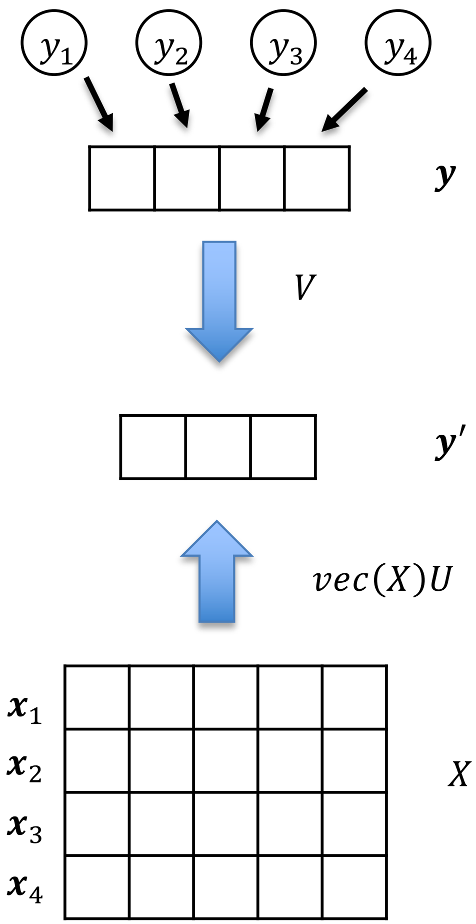
$\boldsymbol{y}' = \boldsymbol{y}V$ Latent target space

Intuition: find a linear transformation such that the original target space can be reconstructed from the latent target space

$$\min_{V} \|Y - YVV^T\|_F^2 \ s.t. V^T V = I$$

F. Tai and Lin H.-T. "Multi-Label classification with principal label space transformation," *Neural Computation*, 2012.

CPLST: Conditional Principle Target Space Transformation



The target space of a graph

Linear compression via a matrix V

$\boldsymbol{y}' = \boldsymbol{y}V$ Latent target space

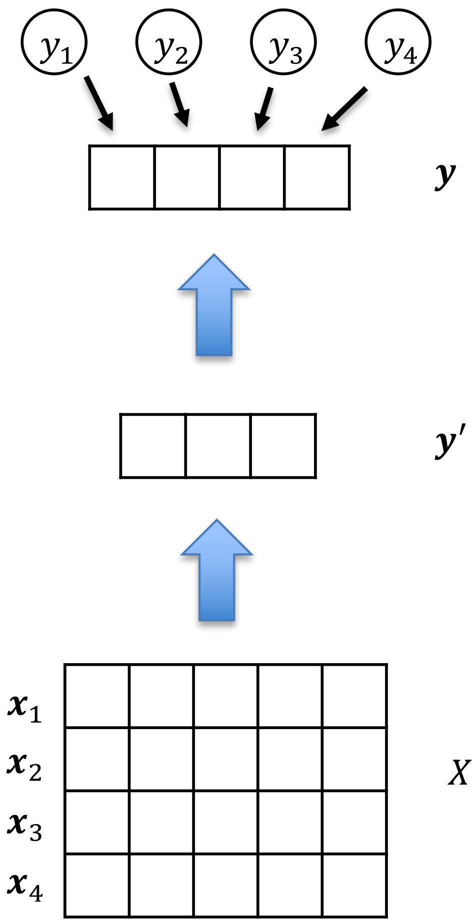
Latent target space is a predictive form of the feature space

The feature space of a graph

$$\min_{V} ||vec(X)U - YV|| + ||Y - YVV^T||_F^2 \; s.t. V^T V = I$$

Chen Y. and Lin H.-T. “Feature-aware Label Space Dimension Reduction for Multi-label Classification,” *NIPS,* 2012.

FaIE: Feature-aware Implicit target space Encoding



The target space of a graph

The original target space can be reconstructed from the latent target space

Learning the Latent target space

Latent target space is predicted directly from of the feature space

The feature space of a graph

Intuition: Learning a feature-aware latent target space directly

Lin Z. et al. “Multi-label Classification via Feature-aware Implicit Label Space Encoding,” *ICML*, 2014.

Our Recently Proposed Method:
Structure-aware Intrinsic Representation (SIR) Learning

- **Limitations of related works**
  - They do not model the representation of the feature space
  - They do not account for the structure of a temporal graph

- **SIR - Our Proposed Method**
  - Joint learning of feature space representation and target space representation

Han, C., Cao, X.H., Stanojevic, M., Ghalwash, M., Obradovic, Z. "Temporal Graph Regression vis Structure-Aware Intrinsic Representation Learning," *Proc. 19th SIAM Int'l Conf. on Data Mining, May* 2019.

SAGA: Structure-aware Graph Abstraction

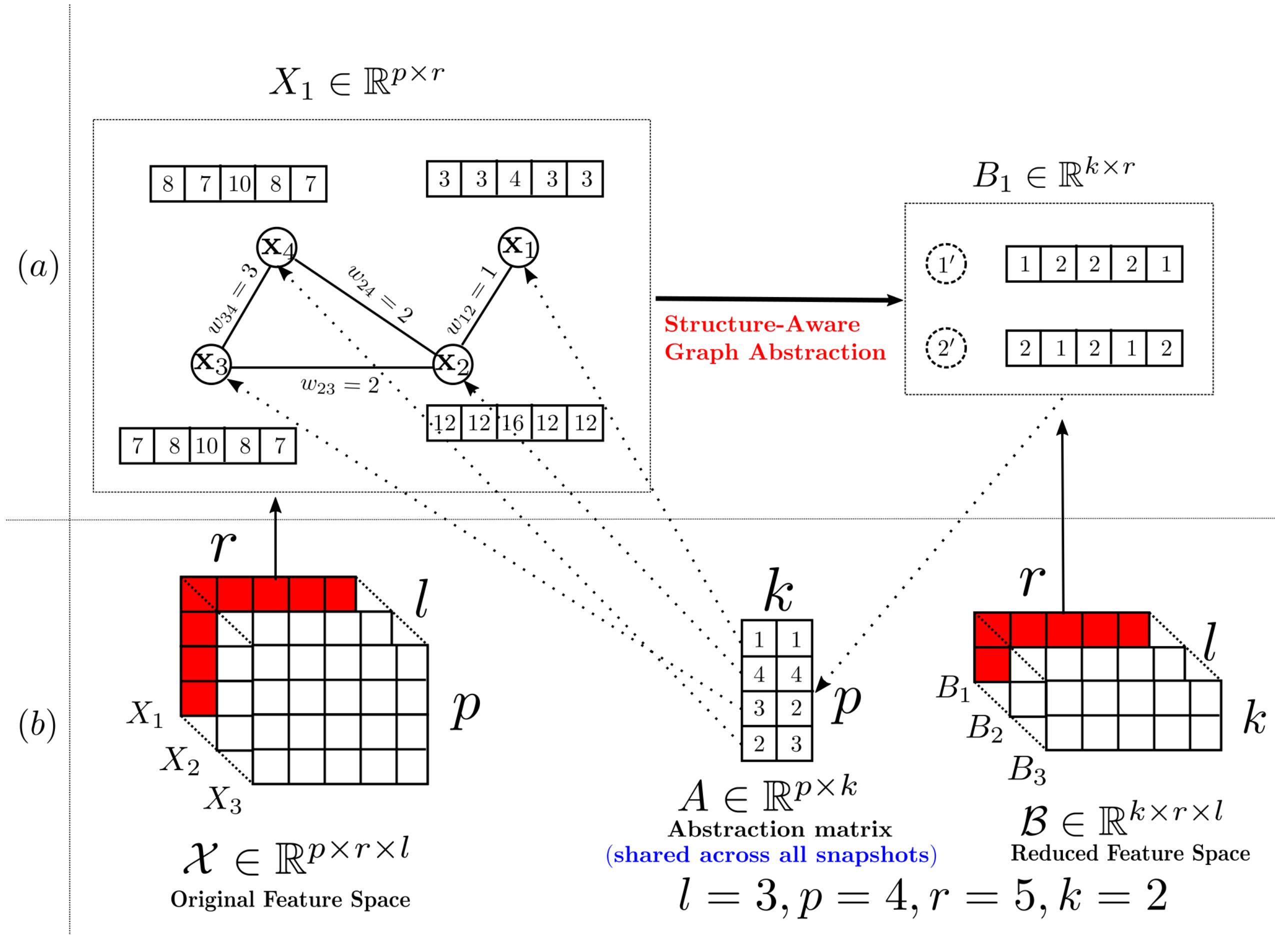
Module One (SAGA):

Graph Abstraction:

summarize p nodes into k nodes, k < p by minimizing reconstruction error

$$\min_{\mathcal{B},A} \|\mathcal{X} - \mathcal{B} \times_1 A\|_F^2$$

$\mathcal{X}$: feature space tensor
$\mathcal{B}$: latent feature space tensor
A: graph abstraction matrix



Structure-aware Graph Abstraction (Cont.)

Temporal Smoothness: neighboring graphs on timeline are similar

$$\min_{\mathrm{B}} \sum_{i=1}^{l-1} ||B_i - B_{i+1}||_2^2$$

Graph Structure Preservation: if two nodes are close then their abstractions should also be similar

$$\min_{\mathrm{A}} tr(A^T LA)$$

L **is the Laplacian matrix of the similarity matrix** W**.**

FAL: Integrating Feature-aware target space Learning

Module Two (FAL):

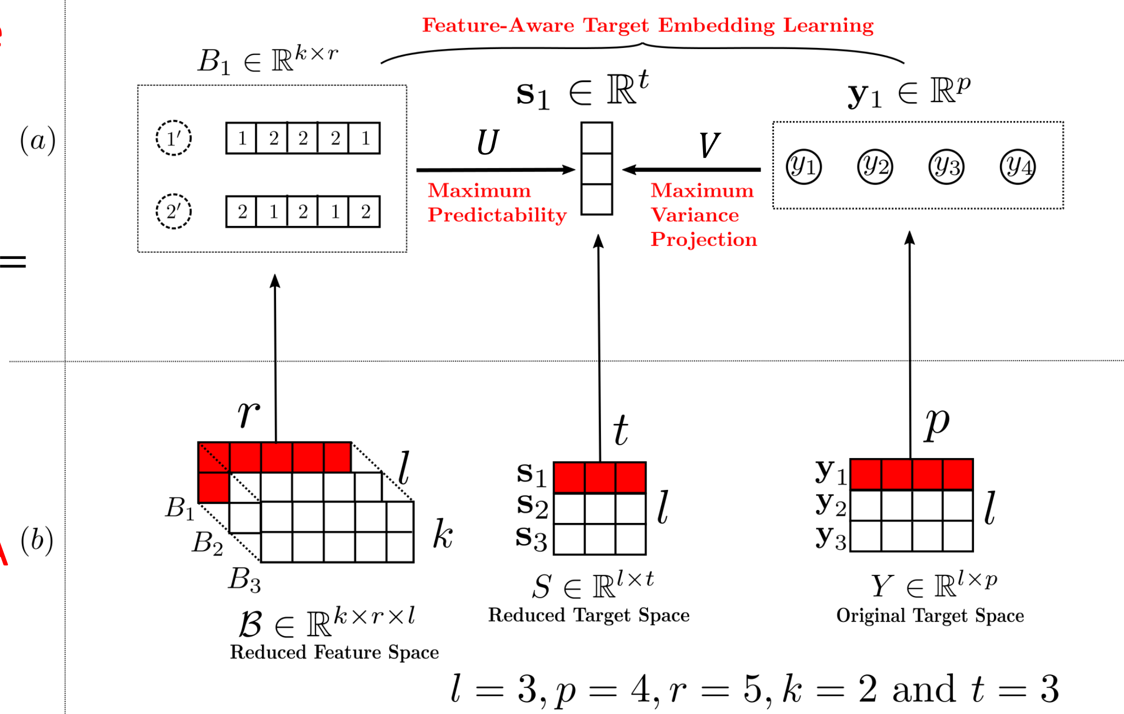
Maximum Predictability: maintain the predictability of the latent target space

$$\min_{U,V} ||YV - \mathcal{B}_{(3)}U||_F^2$$

$\mathcal{B}_{(3)}$ is the mode-3 unfolding of tensor $\mathcal{B}$ (vectorization of the frontal slices. i.e., $\mathcal{B}_{(3)} = [B_1(:), \cdots, B_l(:)]$)

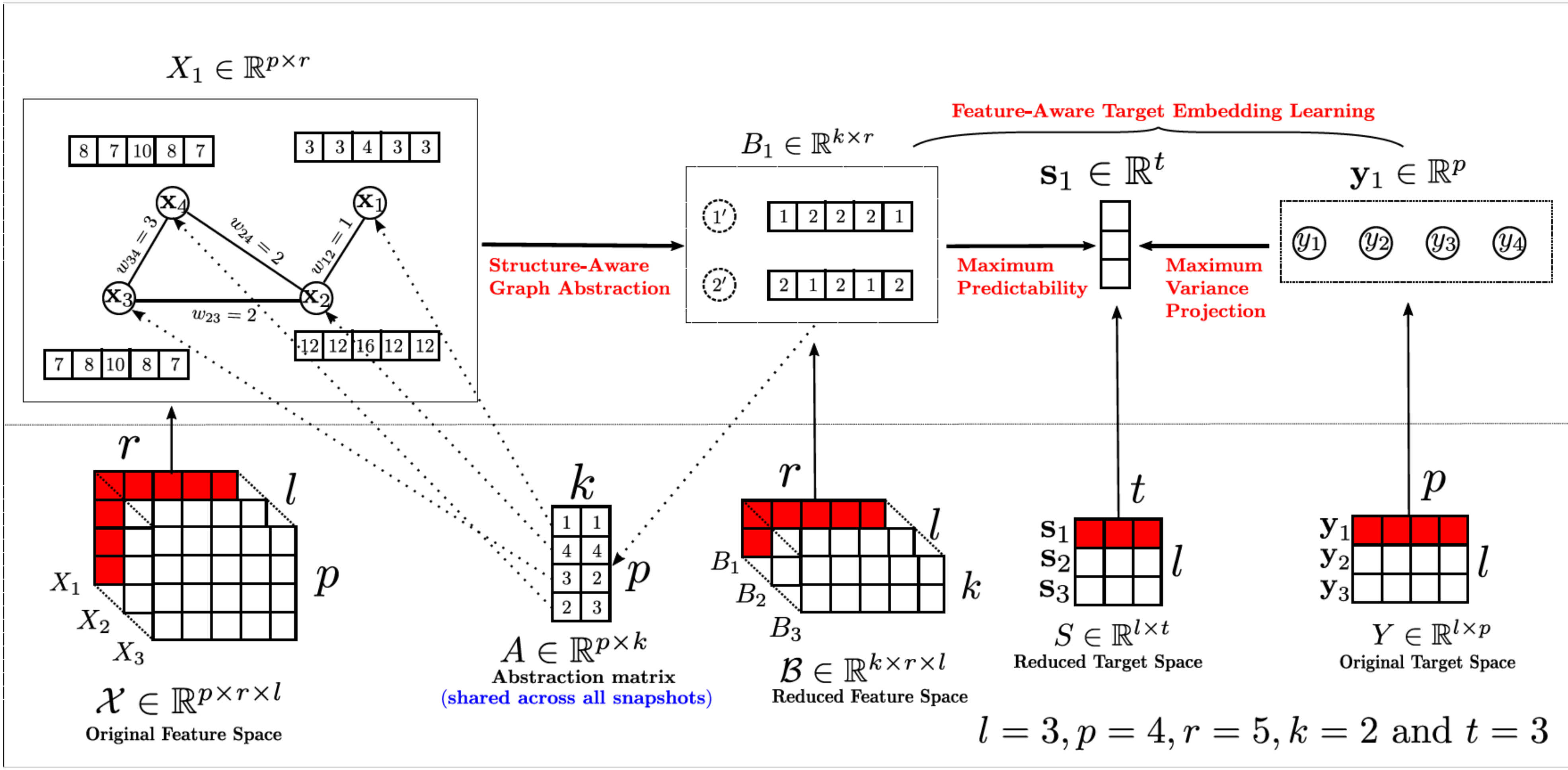
Maximum Variance Projection: find a projection such that the reconstruction error is minimized (PCA on target space)

$$max_{V^TV=I}\ tr(V^TYYV)$$



SIR: Joint Framework for Structure-aware Implicit Representation Learning

SIR = SAGA + FAL



SIR: Joint Learning Problem

$$f = \underbrace{||\mathcal{X} - \mathcal{B} \times_1 A||_F^2}_{\text{Shared Abstraction}} + \delta \sum_{i=1}^{l-1} \underbrace{||B_i - B_{i+1}||_2^2}_{\text{Temporal Smoothness}}$$
$$+ \underbrace{||\mathcal{B}_{(3)}U - YV||_F^2}_{\text{Maximum Predictability}} - \underbrace{tr(V^T Y^T Y V)}_{\text{Maximum Variance}} + \underbrace{\alpha tr(A^T L A)}_{\text{Structure Preservation}}$$

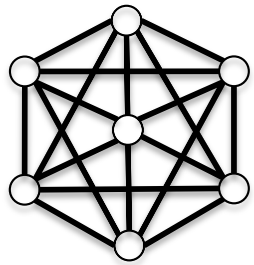
$$\{A^*, \mathcal{B}^*, U^*, V^*\} = argmin_{A,\mathcal{B},U,V^T V = I} f$$

- Derivative-free block coordinate descent algorithm is proposed to solve this optimization problem
- All sub-problems have closed-form solution.

SIR Application: Wind Energy Prediction

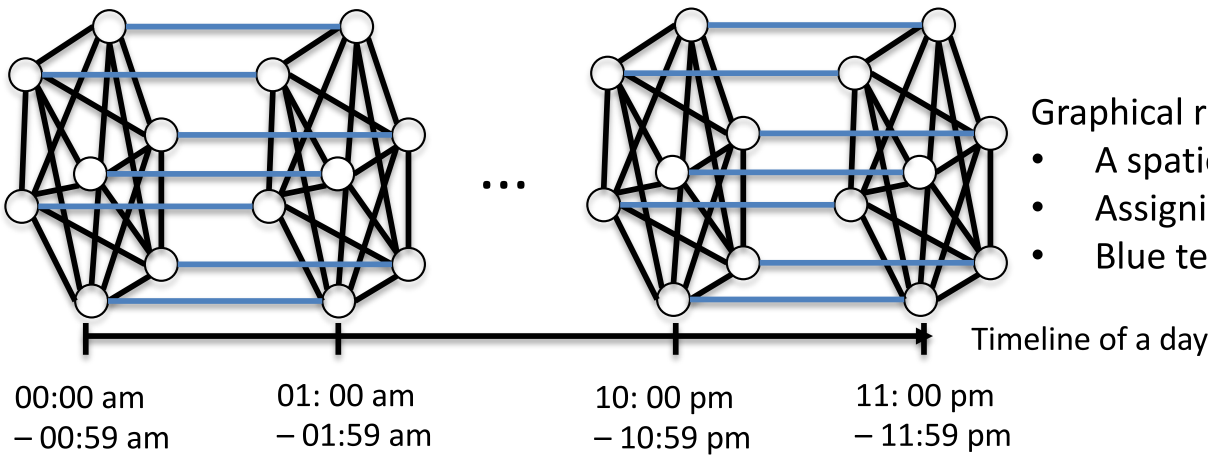
- **Objective: Providing hourly prediction of power generation at 7 wind farms in 24 hours**
  - Build an implicit graph with 24*7 = 168 nodes in each snapshot (p=168)
- **4 features are provided for each node**
  - zonal and meridional wind components, wind speed, and wind direction (r=4)
- **Data: Hourly wind data for 1,080 days from 2009/07/01 00:00 am to 2012/06/29 11:59 pm**

Illustration of the Graphs in Different Temporal Resolutions



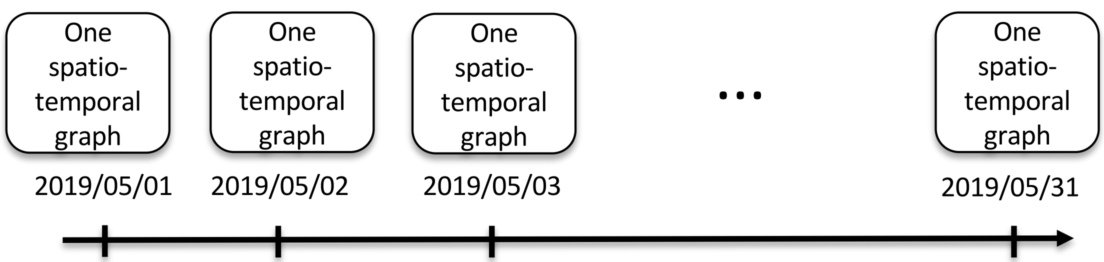
Graphical representation of the graph in one hour

- A full connected graph of 7 farms
- Assigning all edges with unit weights



Graphical representation of the graph in 24 hours

- A spatiotemporal graph of 168 nodes in each snapshot
- Assigning all edges with unit weights
- Blue temporal edges and black spatial edges



Graphical representation of the temporal graph

- Each square stands for a spatiotemporal graph presented above
- Graph representation evolves according the hypothesis temporal smoothness

Experimental Setting

- **Compare the embedding learned by our proposed method (SIR) to alternative embeddings (CPLST, FaIE and SAGA) and no embedding (Raw)**
  - Raw is a baseline without any embedding learning
  - CPLST and FaIE are previously introduced output representation learning methods
  - SAGA is the feature representation learning module of our proposed method

- **Evaluate the quality of embedding with two regressors (LASSO and SGCRF) for temporal graph regression using Mean Square Error (MSE).**
  - LASSO is an unstructured regressor, and SGCRF is a structured regressor [Wytock 2013]

- **Varied the training sizes from {20%, 40%, 60%, 80%, 100%} of training data and experimented on 8 windows for each training size.**
  - Size of training data is l, i.e., the #snapshots in the temporal graph. $l = 300$

Wytock M. et al. “Sparse Gaussian Conditional Random Fields,” *ICML*, 2013.

Results

Results using LASSO as regressor

Method	$20\% * l$	$40\% * l$	$60\% * l$	$80\% * l$	l
Raw	0.0398(0.013)	0.0362(0.006)	0.0482(0.027)	0.0363(0.005)	0.0338(0.007)
CPLST	0.0409(0.014)	0.0554(0.065)	0.0341(0.010)	0.0617(0.074)	0.0551(0.054)
FaIE	0.0507(0.021)	0.0754(0.103)	0.0442(0.017)	0.0498(0.027)	0.0510(0.024)
SAGA	0.0433(0.015)	0.0368(0.011)	**0.0342**(0.010)	0.0328(0.010)	0.0319(0.009)
SIR	**0.0388**(0.013)	**0.0357**(0.010)	0.0344(0.010)	**0.0327**(0.009)	**0.0317**(0.009)

Results using SGCRF as regressor

Method	$20\% * l$	$40\% * l$	$60\% * l$	$80\% * l$	l
Raw	1.0384(0.775)	1.1571(0.788)	0.3808(0.256)	0.0824(0.031)	0.0467(0.008)
CPLST	$> 10(> 10)$	5.7257(6.832)	2.3457(1.747)	1.5216(0.647)	1.0693(0.862)
FaIE	0.1790(0.083)	0.1022(0.015)	0.0809(0.027)	0.0703(0.023)	0.0582(0.015)
SAGA	**0.0469**(0.015)	0.0421(0.011)	0.0407(0.011)	0.0379(0.009)	0.0357(0.010)
SIR	0.0491(0.015)	**0.0413**(0.012)	**0.0391**(0.010)	**0.0371**(0.009)	**0.0356**(0.009)

- SIR-based embedding was always better than alternatives (lower MSE across all experimental settings)

Covered Today: Introduction to Data Science for Structured Learning on Temporal Networks

Several Methods were presented to facilitate Predictive Analytics in:

1. Large dynamic spatiotemporal networks
2. Network embeddings for outage occurrence prediction
3. Structure-aware intrinsic representation learning of temporal networks for wind power prediction

Questions?

Zoran Obradovic

zoran.obradovic@temple.edu
www.dabi.temple.edu/~zoran