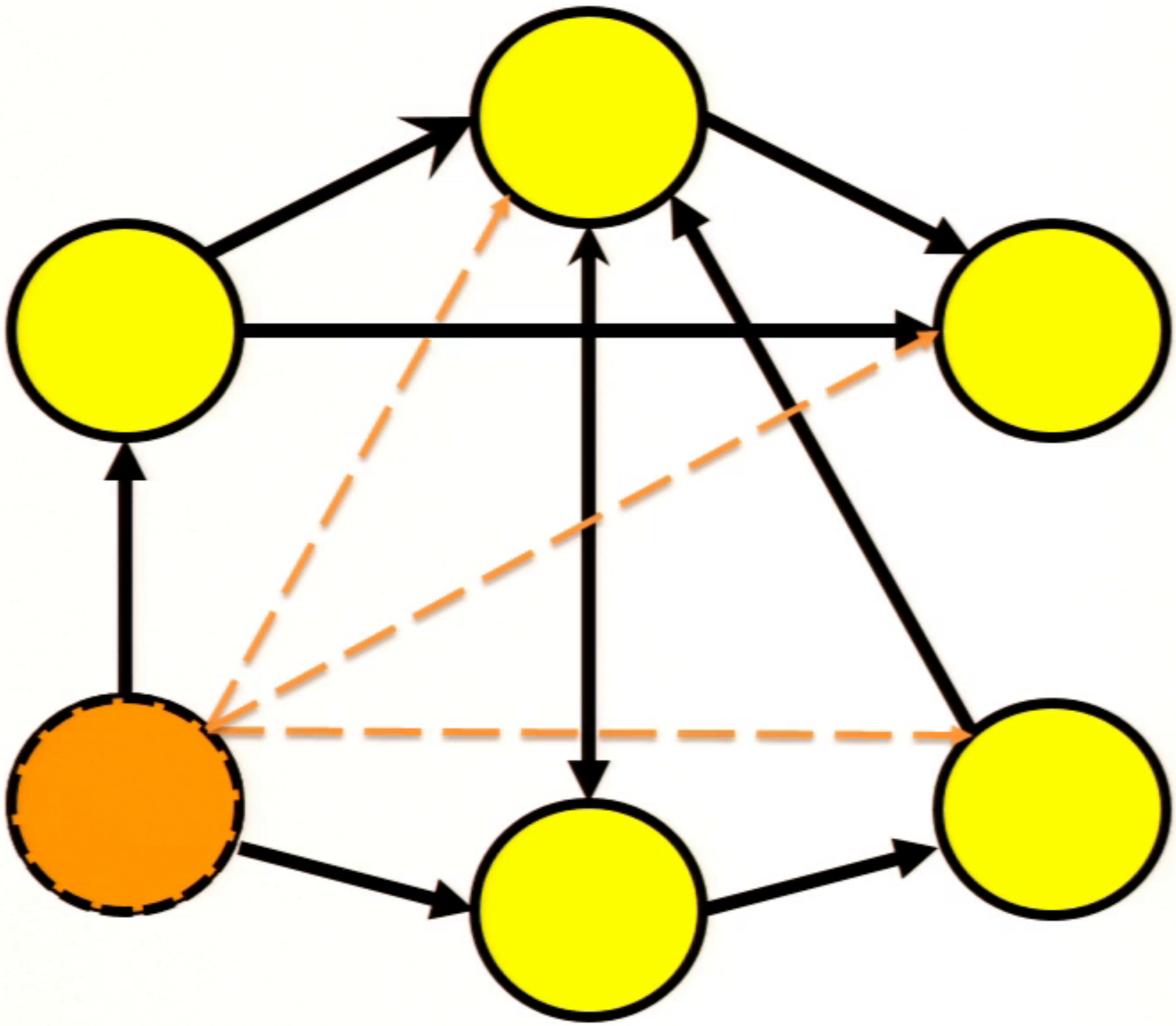
PLUMS: Predicting Links Using Multiple Sources

Karthik Subbian

Joint work with Arindam Banerjee (University of Minnesota) and Sugato Basu (Google Research)

Link Prediction



Multiple Sources

Single Sign-on

Multiple Sources

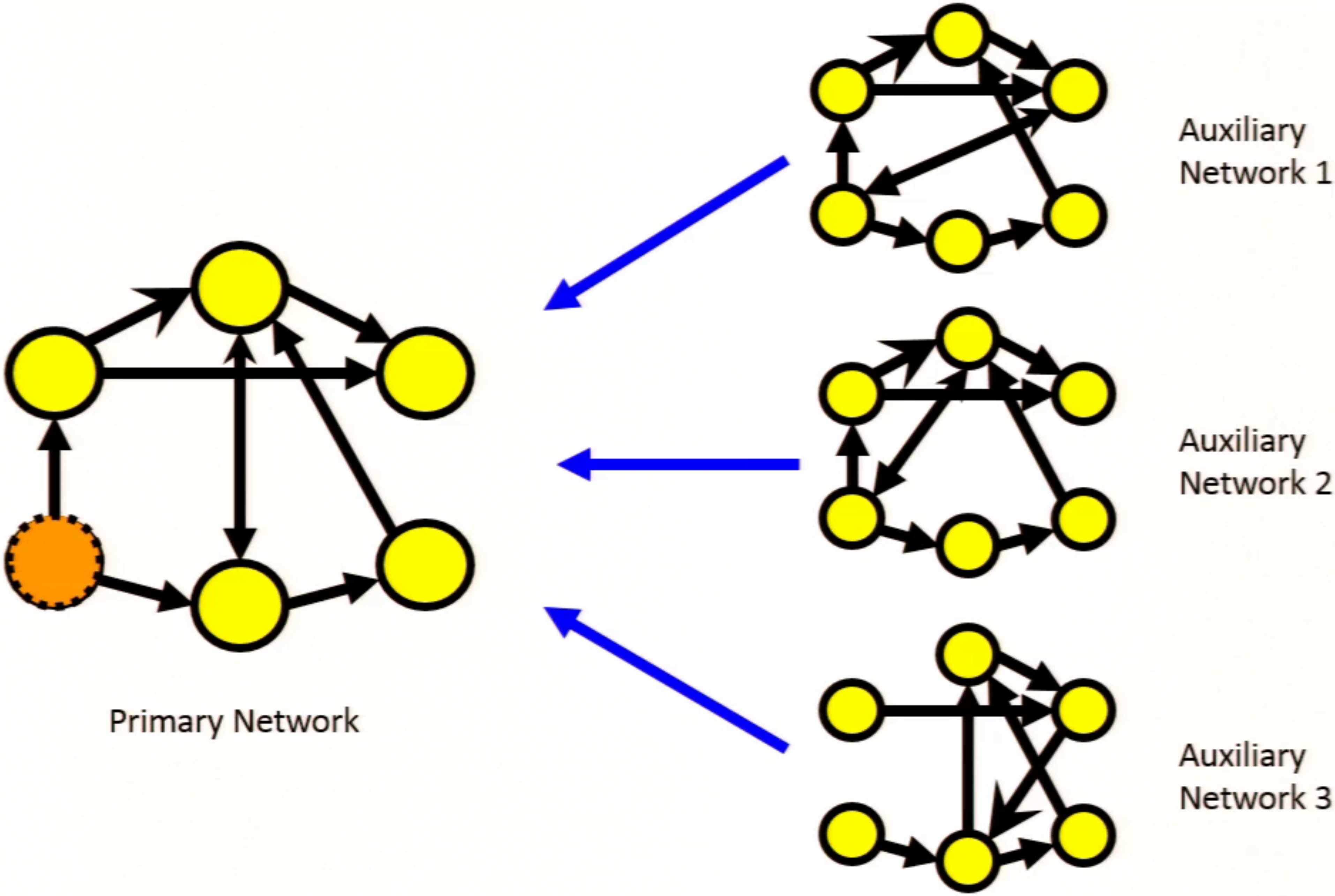
Consumer Networks

Enterprise Networks

academia.edu

Academic Networks

Primary Vs Auxiliary Networks



Challenges

- Network sparsity
  - Hard to make confident predictions
- Feature based link prediction models [Lu et al. 2010]
  - Features are extremely domain and network dependent
- Incorporating user feedback
  - Some users like friends from other network while other don't

Challenges

Multiple Sources

- Feature based link prediction models [Lu et al. 2010]
  - Features are extremely domain and network dependent
- Incorporating user feedback
  - Some users like friends from other network while other don't

Challenges

Multiple Sources

No Explicit Feature Engineering

- Incorporating user feedback
  - Some users like friends from other network while other don't

Challenges

Multiple Sources

No Explicit Feature Engineering

Supervise Using Past Accepts/Rejects

Our Approach

WANT SOME EXERCISE?

GO ON A RANDOM WALK!

Random Walk

Standard Random Walk Model

$$\boldsymbol{\pi} = (1-\alpha)\mathbf{P}_k^T\boldsymbol{\pi} + \alpha\boldsymbol{\gamma}$$

Moving to a neighbor **Teleportation**

For Multiple Networks

$$\boldsymbol{\pi}_i = (1-\alpha)\sum_{k=1}^{n} x_k \mathbf{P}_k^T \boldsymbol{\pi}_i + \alpha\mathbf{e_i}$$

Convex Combination

Incorporating Supervision

$$\min_{\mathbf{x},\boldsymbol{\Pi}} \quad -\sum_{i=1} \mathbf{y}_i^T \boldsymbol{\pi}_i$$

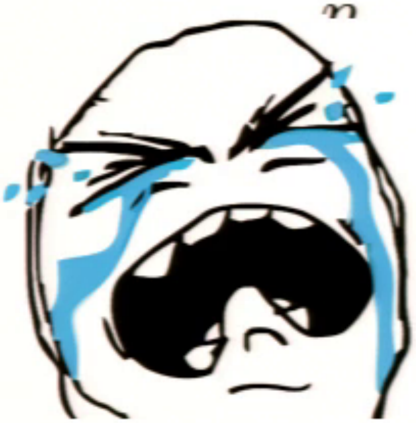
← Maximize number of accepts (consistent with user feedback)

subject to

$$\boldsymbol{\pi}_i = (1-\alpha)\left(\sum_{k=1}^{n} x_k \mathbf{P}_k^T\right)\boldsymbol{\pi}_i + \alpha \mathbf{e}_i,\ \forall i$$

↑ Do Random Walk

$$= 1\,,$$



Bi-linear constraint !!

Approximate Version

$$\min_{\mathbf{x},\boldsymbol{\Pi}} \quad -\sum_{i=1} \mathbf{y}_i^T \boldsymbol{\pi}_i$$

subject to

$$\boldsymbol{\pi}_i = (1-\alpha)\left(\sum_{k=1}^{n} x_k \mathbf{P}_k^T\right)\boldsymbol{\pi}_i + \alpha \mathbf{e}_i,\ \forall i$$

$$\sum_{i=1}^{m} \left\| \boldsymbol{\pi}_i - (1-\alpha)\left(\sum_{k=1}^{n} x_k \mathbf{P}_k^T\right)\boldsymbol{\pi}_i - \alpha \mathbf{e}_i \right\|^2 \leq \epsilon$$

$$x_k \geq 0\ ,\ k = 1, \ldots, n.$$

Approximate Version

$$\min_{\mathbf{x},\mathbf{\Pi}} \quad -\sum_{i=1} \mathbf{y}_i^T \boldsymbol{\pi}_i$$

subject to

$$\sum_{i=1}^{m} \left\| \boldsymbol{\pi}_i - (1-\alpha) \left(\sum_{k=1}^{n} x_k \mathbf{P}_k^T \right) \boldsymbol{\pi}_i - \alpha \mathbf{e}_i \right\|^2 \leq \epsilon$$

$$\sum_{k=1}^{n} x_k = 1 \, ,$$

$$x_k \geq 0 \, , \; k = 1, \ldots, n.$$

Problem Formulation

$$\min_{\mathbf{x},\mathbf{\Pi}} L_\lambda(\mathbf{x}, \mathbf{\Pi}) = -\sum_{i=1}^{m} \mathbf{y}_i^T \boldsymbol{\pi}_i$$

$$+ \frac{\lambda}{2} \sum_{i=1}^{m} \left\| \left(\boldsymbol{\pi}_i - (1-\alpha) \left(\sum_{k=1}^{n} x_k \mathbf{P}_k^T \right) \boldsymbol{\pi}_i - \alpha \mathbf{e}_i \right) \right\|_2^2$$

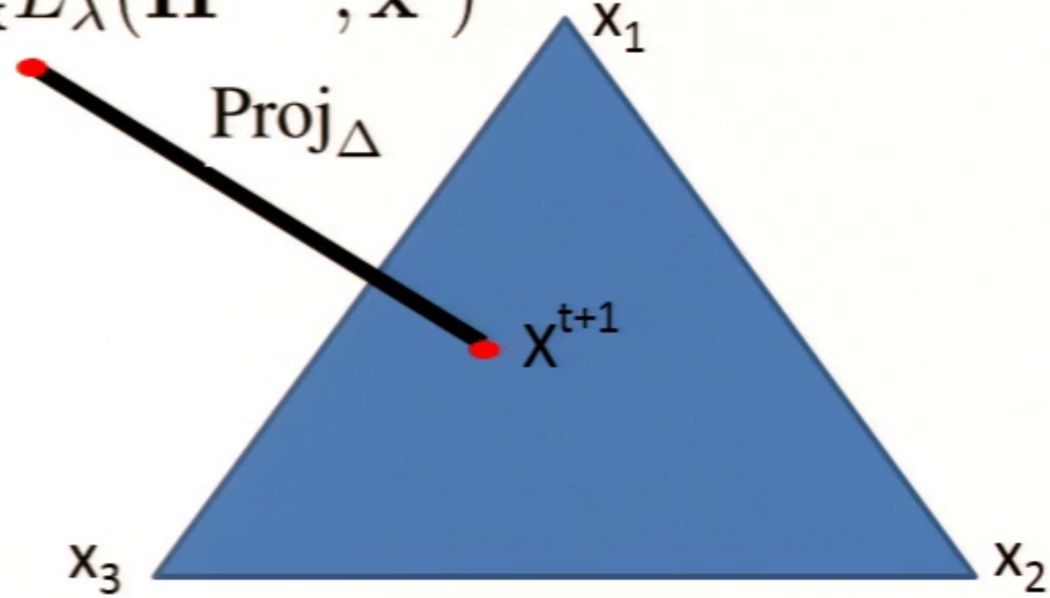
- Problem is not jointly convex on both X and π
- Convex when one variable is fixed
- Alternating minimization algorithm

Projected Gradient Descent

$$\mathbf{\Pi}^{t+1} = \mathbf{\Pi}^{t} - \eta \nabla_{\mathbf{\Pi}} L_{\lambda}(\mathbf{\Pi}^{t}, \mathbf{x}^{t})$$

$$\mathbf{x}^{t+1} = \mathrm{Proj}_{\Delta}(\mathbf{x}^{t} - \eta \nabla_{\mathbf{x}} L_{\lambda}(\mathbf{\Pi}^{t+1}, \mathbf{x}^{t}))$$

$$\mathbf{x}^{t} - \eta \nabla_{\mathbf{x}} L_{\lambda}(\mathbf{\Pi}^{t+1}, \mathbf{x}^{t})$$



PLUMS Updates

$$\mathbf{\Pi}^{t+1} = \mathbf{\Pi}^t - \eta \nabla_{\mathbf{\Pi}} L_\lambda(\mathbf{\Pi}^t, \mathbf{x}^t)$$

$$\mathbf{x}^{t+1} = \mathrm{Proj}_\Delta(\mathbf{x}^t - \eta \nabla_{\mathbf{x}} L_\lambda(\mathbf{\Pi}^{t+1}, \mathbf{x}^t))$$

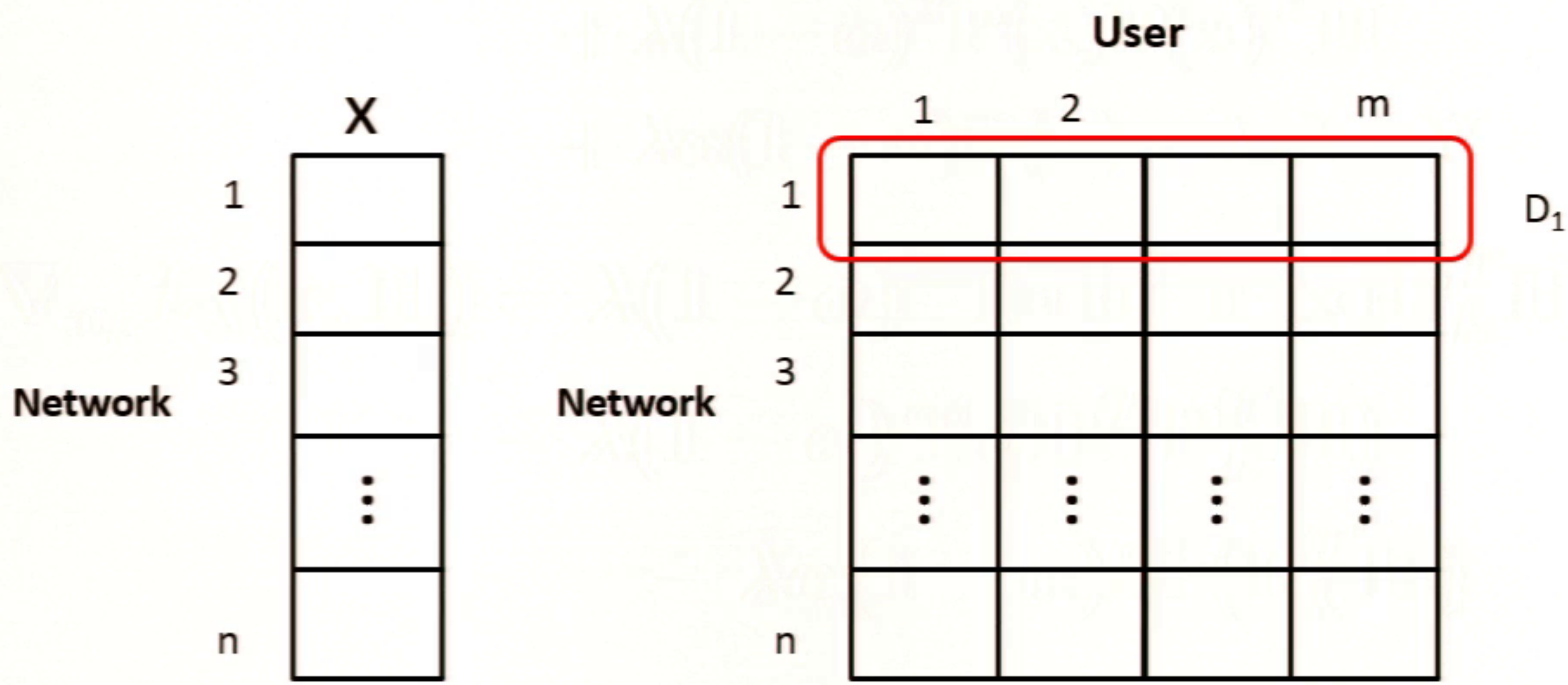
$$\begin{aligned}\nabla_{\mathbf{\Pi}} L_\lambda(\mathbf{x}, \mathbf{\Pi}) = \lambda \Pi - \lambda(1-\alpha)(\mathbf{P}(\mathbf{x})^T + \mathbf{P}(\mathbf{x}))\mathbf{\Pi} \\ + \lambda(1-\alpha)^2 \mathbf{P}(\mathbf{x})\mathbf{P}(\mathbf{x})^T \mathbf{\Pi} \\ + \lambda\alpha(1-\alpha)\mathbf{P}(\mathbf{x}) - \lambda\alpha\mathbb{I} - \mathbf{Y}\end{aligned}$$

$$\begin{aligned}\nabla_{x_k} L_\lambda(\mathbf{x}, \mathbf{\Pi}) = \lambda(1-\alpha)^2 \operatorname{Tr}(\mathbf{\Pi}^T \mathbf{P}(\mathbf{x}) \mathbf{P}_k^T \mathbf{\Pi}) \\ - \lambda(1-\alpha) \operatorname{Tr}(\mathbf{\Pi}^T \mathbf{P}_k^T \mathbf{\Pi}) \\ - \lambda\alpha(1-\alpha) \operatorname{Tr}(\mathbf{P}_k^T \mathbf{\Pi})\end{aligned}$$

Personalized PLUMS (pPLUMS)

$$\min_{\mathbf{D}_k,\mathbf{\Pi}} L_\lambda(\mathbf{D},\mathbf{\Pi};\lambda) = -\sum_i \mathbf{y}_i^T \pi_i$$

$$+\frac{\lambda}{2}\sum_i \left\| \left(\pi_i - (1-\alpha)\left(\sum_k D_k \mathbf{P}_k^T\right)\pi_i - \alpha \mathbf{e_i} \right) \right\|_2^2$$



pPLUMS Updates

$$
\begin{aligned}
&\nabla_{\mathbf{\Pi}} L_\lambda(\mathbf{D}, \mathbf{\Pi}) \\
&= \lambda \Pi - \lambda(1-\alpha)(\mathbf{P}(\mathbf{D})^T + \mathbf{P}(\mathbf{D}))\mathbf{\Pi} \\
&\qquad + \lambda(1-\alpha)^2 \mathbf{P}(\mathbf{D})\mathbf{P}(\mathbf{D})^T \mathbf{\Pi} \\
&\qquad + \lambda\alpha(1-\alpha)\mathbf{P}(\mathbf{D}) - \lambda\alpha\mathbb{I} - \mathbf{Y}
\end{aligned}
$$

$$
\begin{aligned}
\nabla_{D_k} L_\lambda(\mathbf{D}, \mathbf{\Pi}) = {} & \lambda(1-\alpha)^2 \operatorname{diag}\left(\mathbf{\Pi}\mathbf{\Pi}^T \mathbf{P}(\mathbf{D}) \mathbf{P}_{\mathbf{k}}^{\mathbf{T}}\right) \\
& - \lambda\alpha(1-\alpha) \operatorname{diag}\left(\mathbf{\Pi}\mathbf{P}_k^T\right) \\
& - \lambda(1-\alpha) \operatorname{diag}\left(\mathbf{\Pi}\mathbf{\Pi}^T \mathbf{P}_k^T\right)
\end{aligned}
$$

Data Sets

Synthetic

- BTER
- 2.5, 3 Power law exponent
- Edges binned t_1, t_2 and t_3

DBLP

- 1996-2011
- IR, DM, ML, OS
- 3 time periods
- Different auxiliary net.

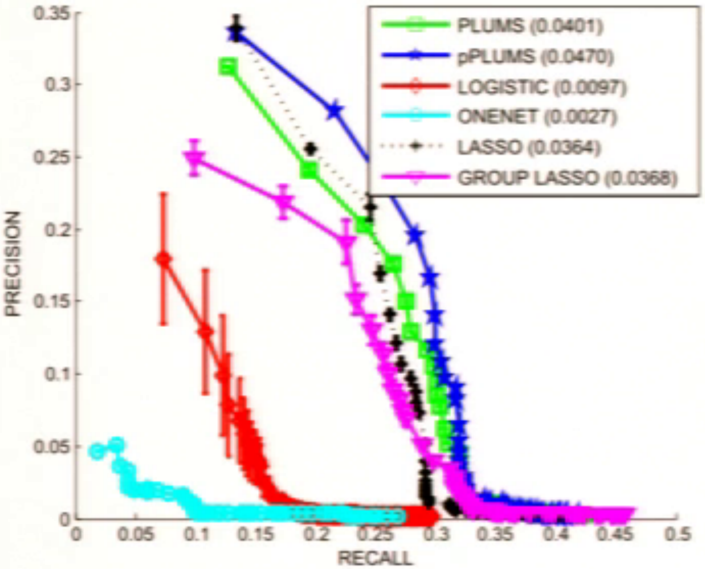
Protein-Protein

- thebiogrid.org
- PA*, DI, CO
- Edges binned 4:3:3 ratio

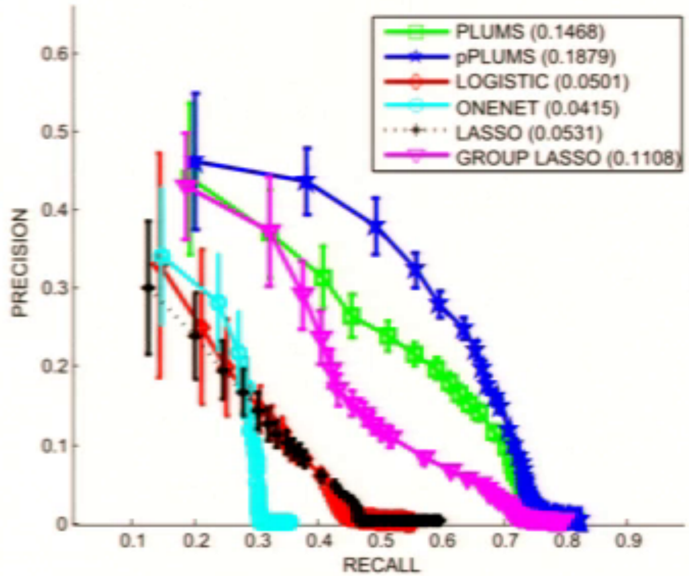
Baselines

- Logistic Regression
  - Edge training data point
  - Independent and combined network features
  - Degree, common neighbor, personalized page rank
- Lasso [Lu et. Al 2010]
  - One, two, three path features for various network permutations
  - L1 norm
- Group Lasso [Lu et. Al 2010]
  - Composite norm with hierarchical sparsity
- Onenet
  - Personalized page rank for single network

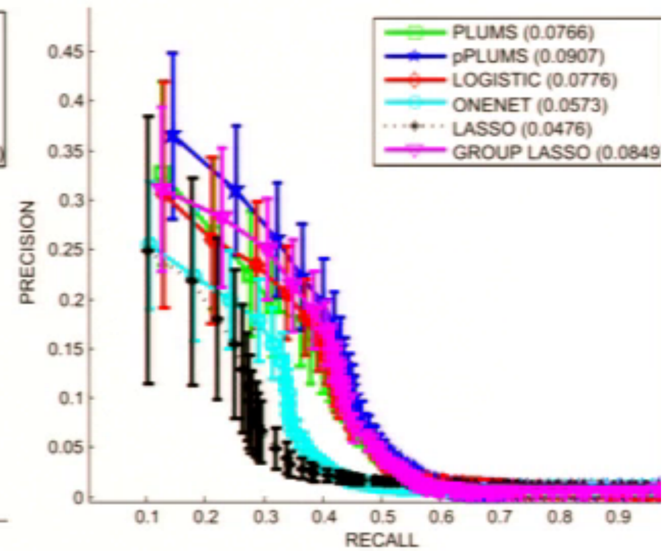
Effectiveness Analysis



DBLP



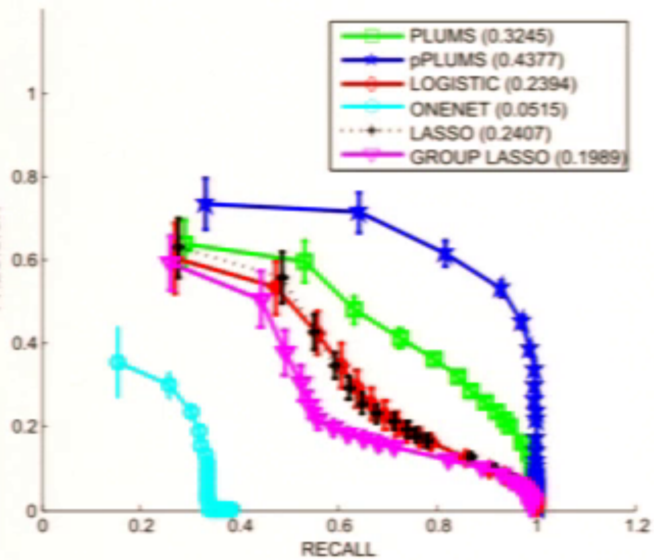
Synthetic



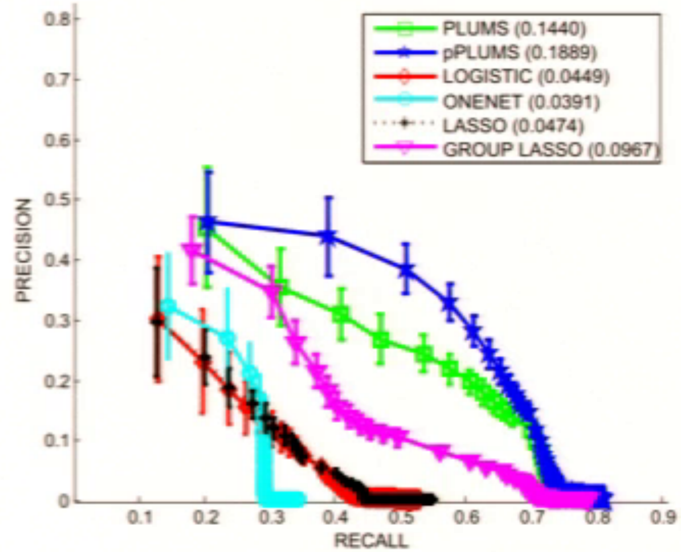
BioGrid

- pPLUMS consistently performs better
- oneNet performs poorly – lack of auxiliary information
- pPLUMS is 26% better than best performing baseline
- Group-Lasso is the consistently best performing baseline

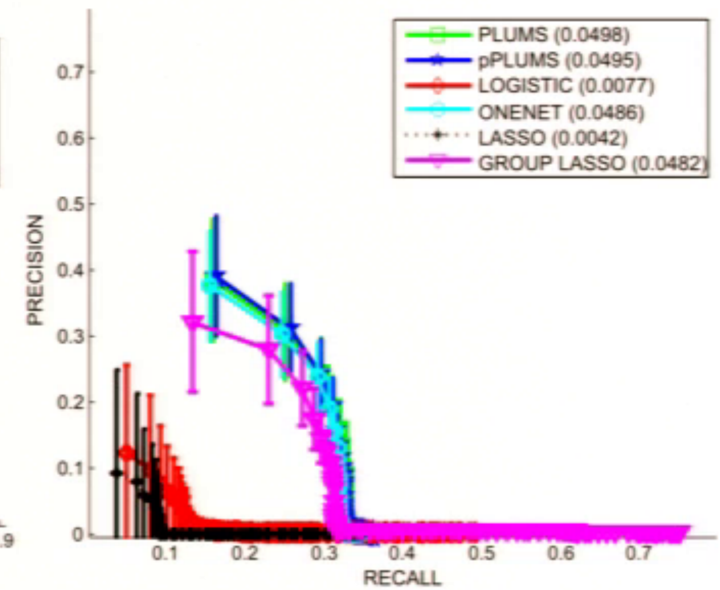
Sparsity Analysis



Low Sparsity



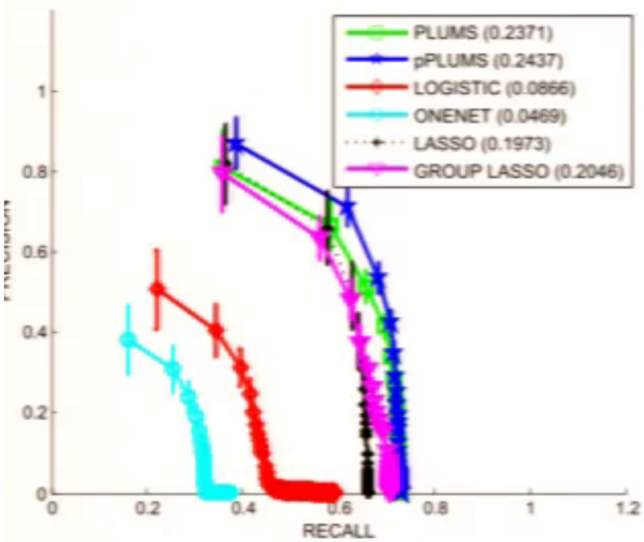
Medium Sparsity



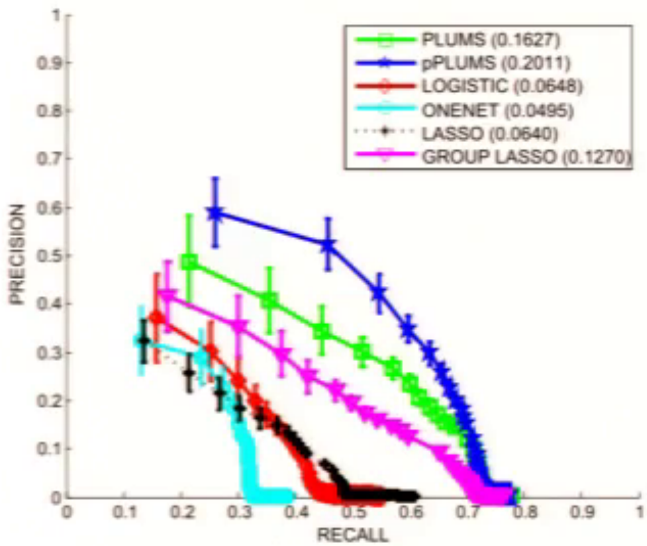
High Sparsity

- Increase sparsity by reducing the edges in auxiliary networks
- pPLUMS and plums makes best use of auxiliary information
- In high sparsity only Group Lasso performs well due to its composite norm structure

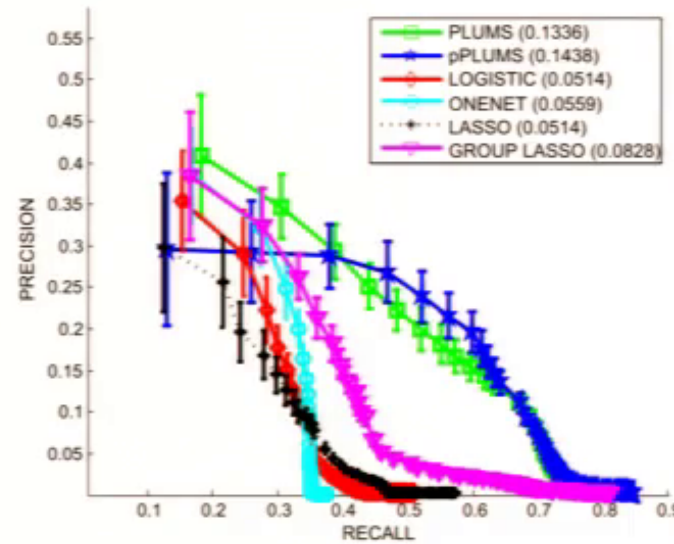
Robustness Analysis



Low Noise



Medium Noise



High Noise

- All approaches perform reasonably at low noise conditions
- Group-Lasso again is a robust baselines that performs consistently well
- Our approach pPLUMS performs well in all noise conditions

Conclusions

- Problem of link prediction is fundamental to social and collaboration networks
- Our goal was to incorporate
  - auxiliary information
  - supervision
  - no explicit features
- We developed both general and personalized models
- Our approach is generalization of Katz measure
- PLUMS is robust under sparse and noisy conditions
- Auxiliary information is mutually un-informative use PLUMS
- Auxiliary network has no extra information use Group-Lasso

Thank you

We acknowledge the research grants from NSF (IIS-1447566, IIS-1422557, CCF-1451986, CNS-1314560, IIS-0953274, IIS-1029711) and NASA grant NNX12AQ39A, DARPA grant W911NF-12-C-0028 and IBM PhD Fellowship.