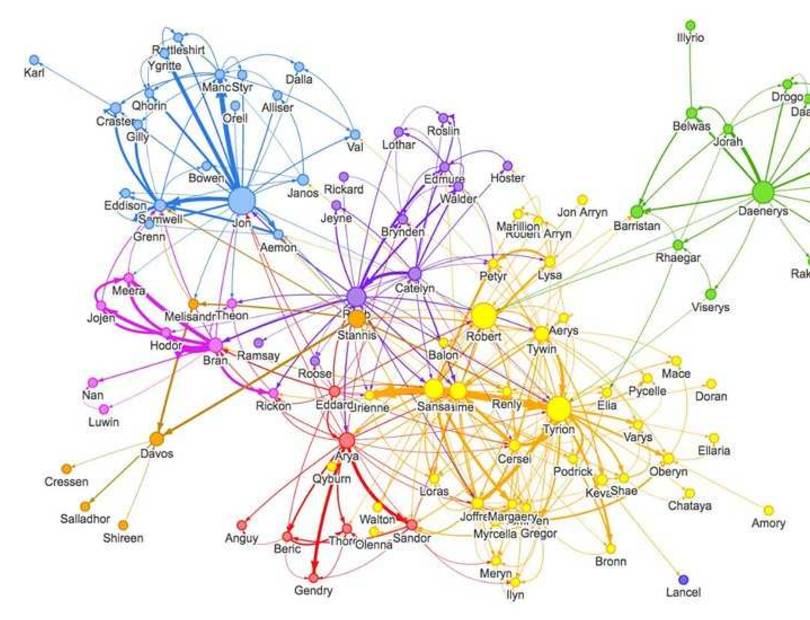
Graph-assisted Bayesian node classifiers

Hakim HAFIDI, Philippe CIBLAT, Mounir GHOGHO

Telecom Paris, Institut Polytechnique de Paris, France
Université Internationale de Rabat, Morocco
Université Mohammed VI Polytechnique, Morocco
https://perso.telecom-paris.fr/ciblat/

Problem statment



Predict class of each unlabeled node in the graph by relying
- on nodes' features and
- on nodes' graph connections

by applyig **homophily principle**

Examples:
- in social networks, people are more likely to connect with those who share the same areas of interest
- in research articles' database, more likely to have connections/citations between articles dealing with the same research topic

	Cora	*Citeseer*
Intra-class connectivity (p)	23×10^{-3}	12×10^{-3}
Inter-class connectivity (q)	5.5×10^{-3}	4.3×10^{-3}
Degree of Impurity (q/p)	0.23	0.36
Logistic Regression (LR)	56.0%	57.2%
Two-layer GNN	81.5%	70.3%

Techniques:
- Label Propagation (LP)
  - Distributed voting
- Feature propagation (FP)
  - Gossiping, sometimes followed by a nonlinear function
  - Graph Neural Networks (GNN). Training done with labeled nodes

Our contributions: Graph Node classification
- No Graph Neural Network (GNN)
- Interpretable algorithm
- Less complex algorithm (with less hyperparameters)

System Model

- $\mathcal{V}_u$: set of nodes involved in the classification of node u.
- $\mathcal{X}_u = \{\mathbf{x}_u\} \cup \{\mathbf{x}_v, v \in \mathcal{V}_u\}$: set of features of u and its "helping" nodes
- y_u: class of node u (what we are looking for!)
- D_k: probability density function of features belonging to class k.

$$D_k(\mathbf{x}_u) = p(\mathbf{x}_u | y_u = k).$$

Graph-Assisted Bayesian (GAB) Classifier

$$\hat{k}_u = \arg\max_k P_u(k)$$

with $P_u(k) = \Pr(y_u = k | \mathcal{X}_u, \mathcal{I}_\mathcal{G})$.
We show that

$$P_u(k) = \pi_k D_k(\mathbf{x}_u) \prod_{d=1}^{\Delta_u} \prod_{v \in \mathcal{N}_u(d)} \left(\sum_{k'=1}^{K} r_{u,v}(k,k') D_{k'}(\mathbf{x}_v) \right)$$

with Δ_u the diameter of the set $\mathcal{V}_u$, π_k a priori classes' probability, and $r_{u,v}(k,k') = \Pr(y_v = k' | y_u = k, \mathcal{I}_\mathcal{G})$ the probability to be on class k' for node v given the fact that we are in class k for node u.

Main Results

- 2 equilikely classes
  - $p(k)$ probability that two nodes from class k are connected
  - $\overline{p}_{\text{arithmetic}}$ arithmetic average of $\{p(k)\}_k$
  - q probability that two nodes from different classes are connected.
- Information on graph is 1-hop

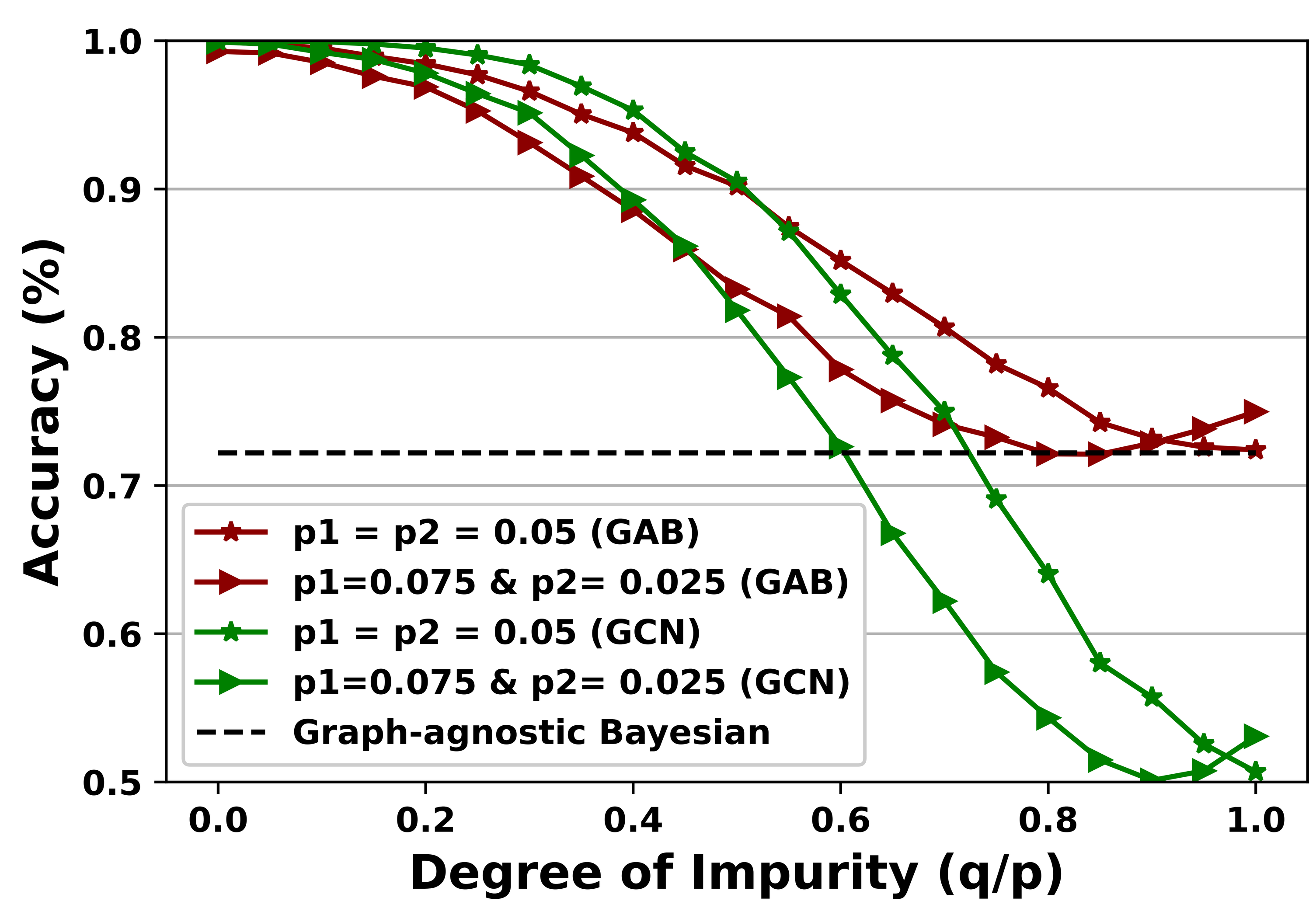
We get

$$\begin{array}{c|c} r(1,2) = \frac{q}{p(1)+q} & r(2,2) = \frac{p(2)}{q+p(2)} \\ r(1,1) = \frac{p(1)}{p(1)+q} & r(2,1) = \frac{q}{q+p(2)} \end{array}$$

Graph-agnostic iff
- $r(1,2) = r(2,2)$ and $r(1,1) = r(2,1)$, or
- $q = \sqrt{p(1)p(2)} = \overline{p}_{\text{geometric}}$, or
- Degree of Impurity $= \frac{q}{\overline{p}_{\text{arithmetic}}} = \frac{\overline{p}_{\text{geometric}}}{\overline{p}_{\text{arithmetic}}} \leq 1$

Numerical Results

Synthetic data:
- 2 classes
- Gaussian distributions with different means and covariance matrices
- Number of nodes $N = 5,000$ and number of features $F = 500$
- 500 (already-labeled) nodes



Real data:

	MLP	**GCN**	**SAGE**	**GAT**	**GMN**	**DGCN**	**GBPN**	**GAB**
Cora	72.1	87.1	86.9	87.1	86.4	**87.2**	86.4	86.9
CiteSeer	71.2	73.5	73.5	73.1	72.9	73.9	74.8	**75.2**
PubMed	86.5	87.1	87.8	88.1	86.7	84.7	**88.5**	86.4
CS	94.2	93.2	93.7	94.0	93.3	94.9	**95.5**	94.5
Physics	95.8	96.1	96.3	96.3	96.1	96.7	**96.9**	96.4

Complexity analysis:

	Parameters to estimate in GAB	Weights to learn in GNN
Cora	10,087	369,066
PubMed	1,512	129,286

Conclusion

- GAB close to GBPN and GAT, the best ones in the literature
- But interpretability
- But low-complexity