Lecture 5-a

Feb 13, 2024

Empirical Data $G^* = (U^*, E^*)$ is often incomplete.

~~Let x_i be the attribute of node i.~~

We have four types of predictions.

- $*_1$ missing node attributes
- — missing link attributes
- $*_2$ missing links
- — missing nodes

} interpolation (imputation)

} extrapolation

↓ increasing difficulty.

$*_1$ Lecture 5 (associative mixing)

$*_2$ Lecture 6 (next week)

Start from ① & ②

Then go here!

Vershynin (2018)

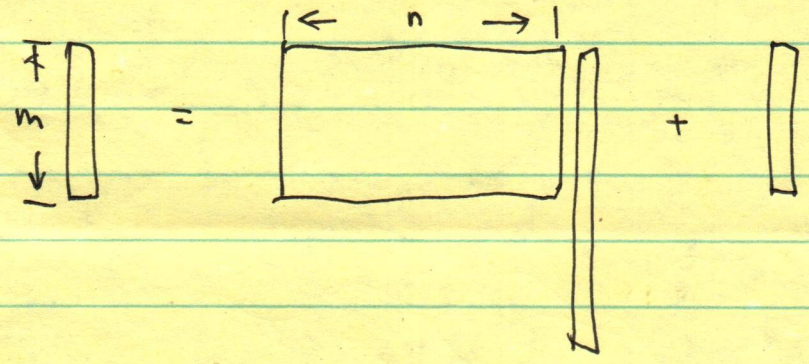
General intuition from "high-dimensional signal recovery problems".

unknown signal $x \in \mathbb{R}^n$

m = random, linear, noisy measurements of x, $y \in \mathbb{R}^m$

$$y = Ax + w$$

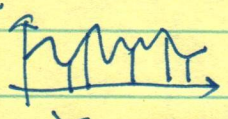
A: unknown measurement matrix
w: unknown noise vector.



y A x w

⇒ Goal: Recover x from A and y as accurately as possible.

example:
- Audio sampling
- Linear regression



$Y = X\beta + w$

$X \in \mathbb{R}^{m \times n}$ sample of predictor variables.

X_{ij} : expression of gene j in patient i

Y_i : whether (or to what extent) patient has disease.

$m = |Y| \sim 100$ # patients while $n = |\beta| \sim 10000$ # genes.

Problem:

① If $m \ll n$, even if $w = 0$, problem is ill-posed!

② sol'n lives in a space of dim $\geq n - m$

(common assumption)

$x \in T \subset \mathbb{R}^n$

↑

but T is informed by some prior information about the signal x.

Something we know, believe, or enforce / want to.

Benefit:

small $T \Rightarrow$ signal recovery is possible when $m \ll n$.

$\}_1$ Predict missing node attributes

$\begin{cases} G^* = (V^*, E^*) & \text{actual data (ground truth)} \\ X = X^\circ \cup X^* & \text{node metadata} \end{cases}$

$X^\circ = \{\phi\}$ $\quad k_i \in X = \begin{cases} \text{categorical} \\ \text{scalar} \quad \mathbb{R}, \mathbb{Z} \end{cases}$

Goal: Recover X° from X^* and G^* as accurately as possible.
(Predict)

$\}_2$ Predict missing link attributes

$\begin{cases} G^* = (V^*, E^*) \\ W = W^\circ \cup W^*, \quad W^\circ = \{\phi\} \end{cases}$

Goal: Recover W° from W^* and G^* as accurately as possible.
(Predict)

$\}_3$ Predict missing links

$\begin{cases} G^* = (V^*, E^*) \\ E = E^\circ \cup E^*, \quad E^- = V^* \times V^* - E^* \end{cases}$

so $|E^\circ|$ is big!

Goal: Predict E° from G^* as accurately as possible.
(i.e., V^* and E^*)

④

§4 Predict missing nodes (extrapolation)

new nodes!

$G^* = (V^*, E^*)$ $\quad V^\circ = \{n+1, \dots, n+k\}$

$V = V^\circ \cup V^*$, $E = E^\circ \cup E^*$, where $E^\circ \in V \times V - E^*$

Goal: Predict V° (and E°) from V^* and E^* as accurately as possible.

— less explored !

— may resort to growth models. (week 13)

network

network sampling / growth.

In ensuing two weeks we focus on §1 and §3.

(*) → "baseline algorithm"

→ refined ones that either use local / global information.

(Δ) → measuring performance. (confusion matrix C, AUC)

area under ROC curve.

⑤ W5-a (*): up to assortative mixing / local smoothing

-b (Δ) + advanced methods (semi-supervised ML)

W6-a (*) on "link prediction"

-b Depth dive — Ghasemian et al. (PNAS, 2018)

and Peixoto & Kirkley (PRE, 2023)

§1 missing node attributes.

(global)

— baseline algorithm

~~take~~ x_i° from Uniform$\left(\mathbb{X}^* - \phi\right)$

take
(impute)

side: why is this "baseline"?
ans: ignores G.

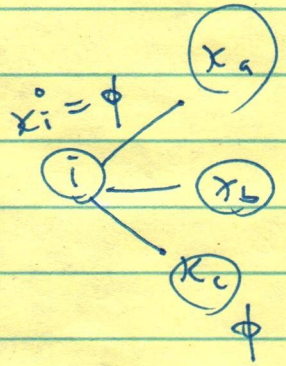
⇒ note: this works whether $\mathbb{X}$ categorical or in $\mathbb{Z}$ or $\mathbb{R}$.

— local smoothing algorithm

take x_i° from $\begin{cases} \text{mean}\left(\mathbb{X}_{nb(i)} - \phi\right) & \text{if } x \in \mathbb{R} \\ \text{mode}\left(\mathbb{X}_{nb(i)} - \phi\right) & \text{if } x \in \{1, \dots, c\} \end{cases}$

The predictor.

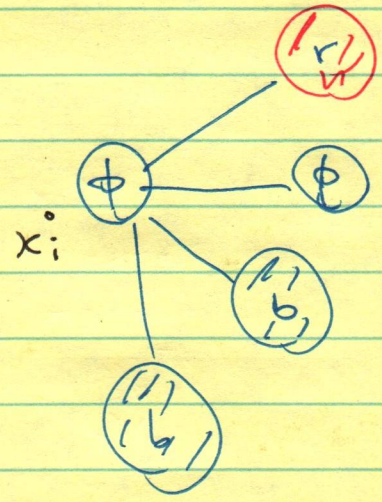
$\mathbb{X}_{nb(i)}$ ~~set of neighbor attributes~~.
(attrib. of neighbors of i)
meta data
(used G's info)

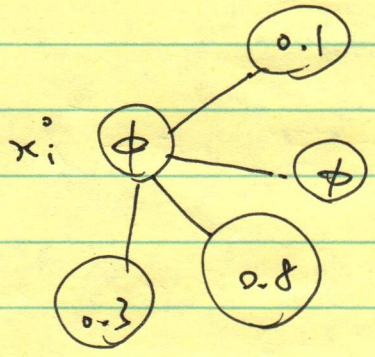


— synchronous updating (don't eat your predictors).
— what if $\{\mathbb{X}_{nb}\} = \{\phi\}$
1) make nb(i) bigger → nb's of nb's.
2) use baseline predictor.

(6)

ex:





impute x_i with b

impute x_i with $\frac{0.1 + 0.8 + 0.3}{3} = 0.4$.

⇓

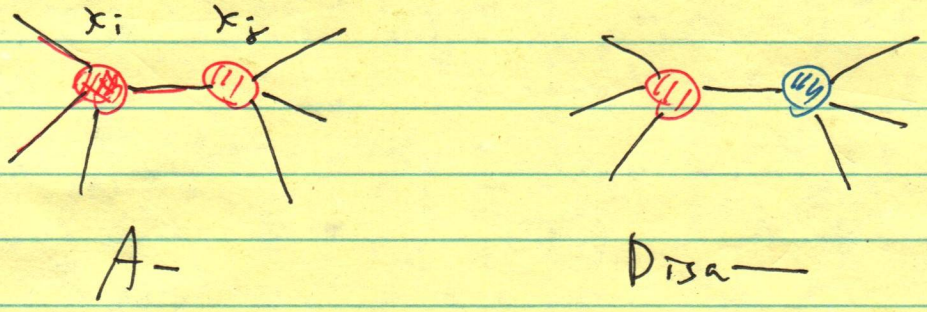
Version of "label propagation"
(whole class of algorithms!)

⑦

Mixing patterns

Phenomenon : Assortative mixing (homophily)

Disassortative —



A— Disa—

economic / ecological 网络.

producer / consumers

x_i : belief, age, social status.

Given $(i,j) \in E$, how similar are x_i, x_j ?

null models !

EXAMPLES :

— Add health 网络 : assortative mixing by age and race, but not by gender !

Newman & Clauset (Nature Communication 2016)

— Weddell Sea food web : disassortative mixing by
- by feed type ; and
- by metabolic category

same paper as above.

— Peel et al. (PNAS 2018)

mixing pattern can be locally heterogeneous.

Caution : Even after showing the slides, we still ran 10 minutes early!

for future classes.

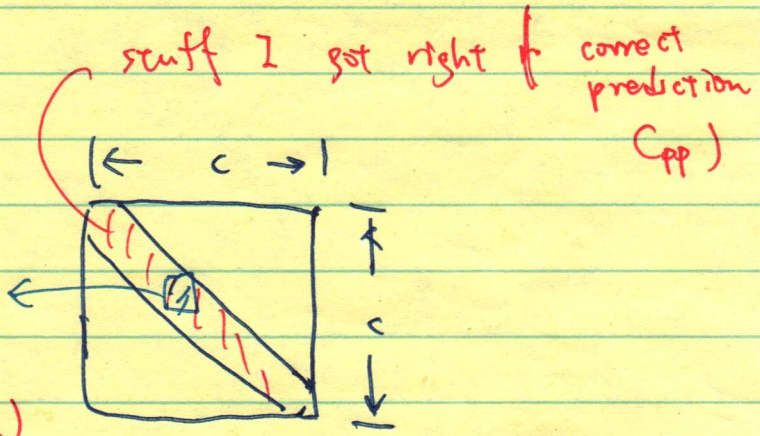
Wk - 5b Feb 15

— only dive to neural #'s this semester.
— How do we measure the performance of what we came up?

measuring performance

confusion matrix

C_{pt} = # of inputs w/ (predicted label p) and (actual label t)



complete summary of the performance. (H) (c^2) in size

$C_{p \neq t}$ = mistakes = things I thought should be p but actually t.

$$\text{accuracy (Acc)} = \frac{\#\text{ current}}{\#\text{ predictions}} = \frac{1}{N}\sum_p C_{pp}.$$

"~~Imbalance~~"

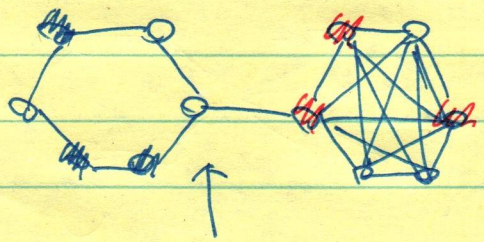
(what does $C_{pt}\big|_{p \neq t}$ count ?)

4:03 F22

§ Imbalance

(2)

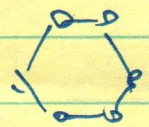
ex:



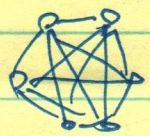
this one go last! (when teaching)

— take red b/c "summarizing all possible label prop. paths" (which we usually don't do)

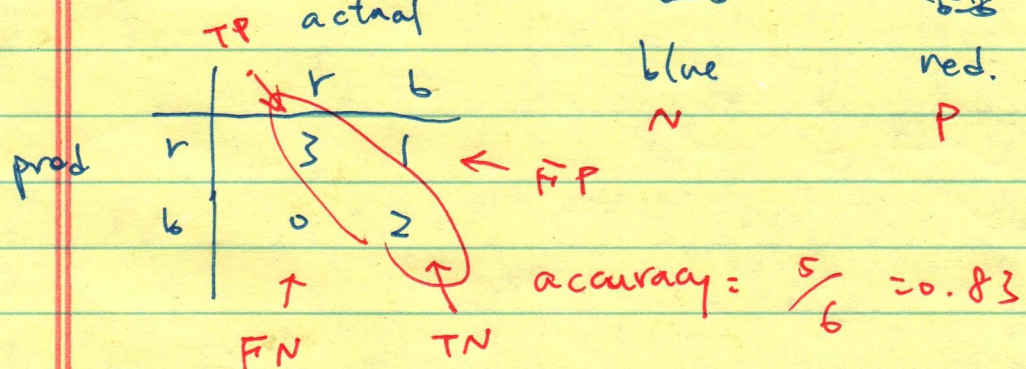
If assume actual:



blue
N



red.
P



accuracy: $\frac{5}{6} = 0.83$

If "binary classification" (and only if)

(3)

§ Penguin example. (see Notes)

— 2020 H. Sayama & Michelle Girvan / Fang. Lift / (I know penguins in this!

— label propagation fails

— lesson: at least look at the data once.

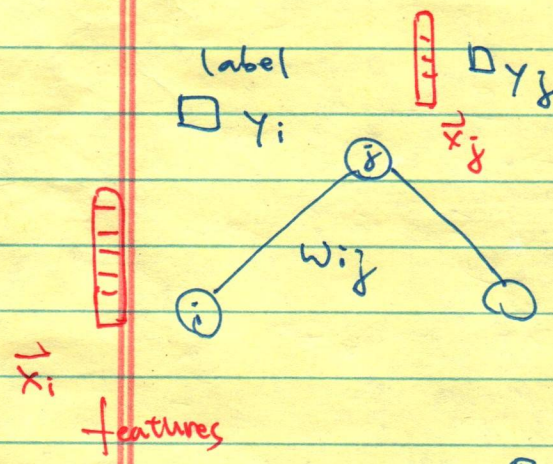
(Exploratory data analysis)

plot / compute assortativity coefficient, etc.

§ Modern techniques

— predicting missing node attr. { node classification (2018)
(Deep) { collective inference

Moore & Neville (2017)



Goal: predict missing y_i

assume: obs. $G=(V,E)$

and $\vec{x}$ node features.

$n \times (p+1)$

$\mathbb{R}$

node covariates. (multi-dim generalization of the things we saw on Wk-5a)

→ ① $\mathrm{cov}(\{\vec{x}_i\})$ within features of a node

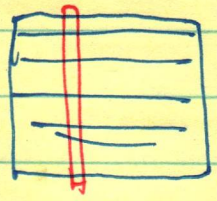
→ ② $\mathrm{cov}(\{\vec{x}_i, \vec{x}_j\})$, conditioned on $(i,j) \in E$

(4)

- At least 15 yr history.

§ Deep learning came for graph (2014) "GNN" 2009 Complicated. not scalable.

"Deepwalk" graphs are hard, vectors are easy. (spaces)

Obs = 

predicted variables

$f($  $|\{\theta\}) =$ 

↑ not graphs. / not adj matrix.

"point cloud machine learning"

- sum,
- random forest,
- regression
- lable prop.

Deepwalk $(G, \{\theta\})$ $\xrightarrow{\text{output}}$ 

"embedding" (penguin ### ⇒ also an embedding by 4r - sex)

Solve optimation prob. (low-dim structure in high-dim space)

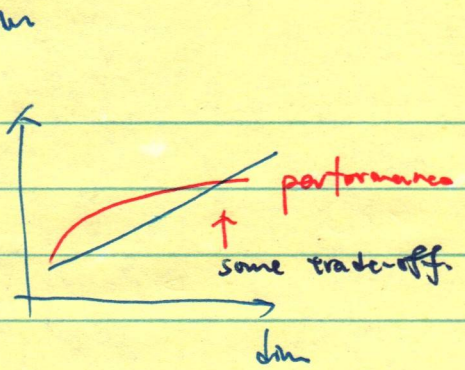
find $\mathbb{R}^{|V| \times d}$ matrix,

s.t. $f($ embedding $)$ is minimized !

(how good is the embedding) → variety of methods !

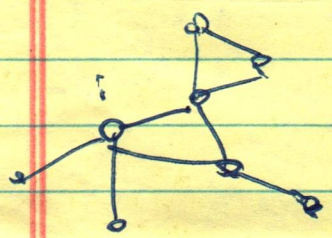
also.

— node2vec(G, θ)



— How do we generate  ?

"Random Walk"



deepwalk uses

Word2vec(•)

(random walk for NN embeddings, but works on sentences)

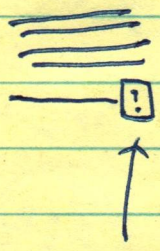
— NLP

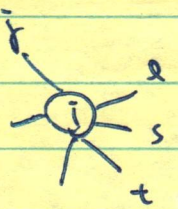
a quick brown fox jump ...

Words : nodes

Sentence : seq. of nodes. from a RW. of length k

corpus : a walk of sentences





select uniformly at random from nb(i)

INPUT

height × width

(n × r) × lk

(node) (sentences for every node) } each of length k

OUTPUT embeddy

deepwalk is just a wrapper for word2vec

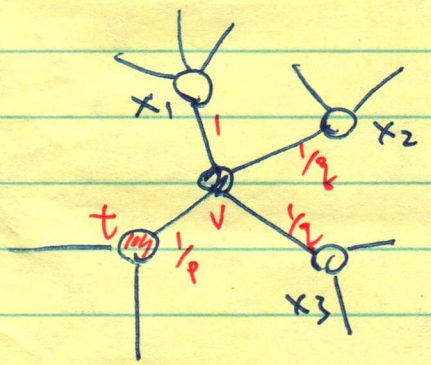
seq encodes the structure of graph.

graph → seq → vectors.

perhaps graph → vectors.

§ node2vec (generalization of DeepWalk)

"parametrizes the walk"



p return parameter

q in-out

Deepwalk : 1^{st}-order RW.

node2vec : 2^{nd}-order RW.

... t v ?

Assign weights: according to

$$\alpha_{pq}(t,x) = \begin{cases} \frac{1}{p} & \text{if } d_{tx}=0 \\ 1 & \text{if } d_{tx}=1 \\ \frac{1}{q} & \text{if } d_{tx}=2 \end{cases}$$

v to decide which vertex to visit next, given that it came from t.

— feed to word2vec to move on!

- blow your mind!
- DeepWalk (2014) 0517 citations as of (2017)
- node2vec (2016) 113.6

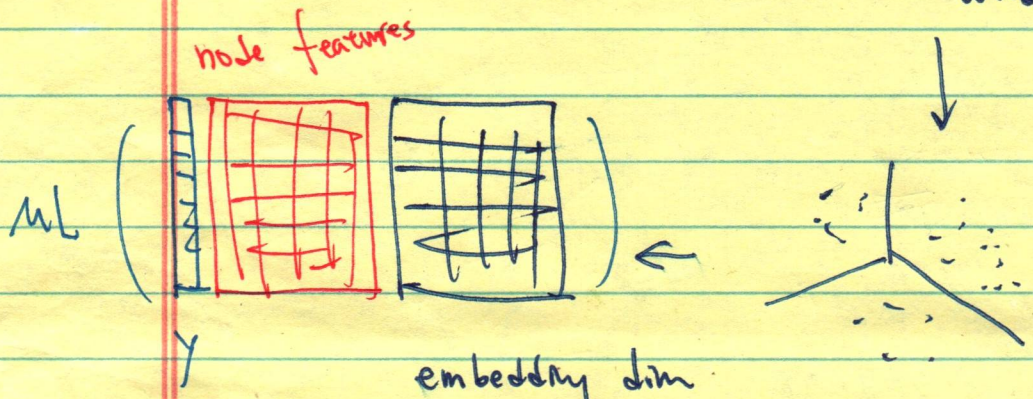
(7)

— One argument of DW/NV ⇒ they are scalable !

— hierarchical softmax ... ?? but there are libraries !

skip gram

... tV [?] → ≡≡≡ } seq's → word2vec(·)



not many theory about why it works ... until ...

§ Third paper ! Jia-Benson (2022)

Great paper, see figure 1. (and Table 4) (that LR better than LP)

parametrized by smoothness and noise levels by cross validation

MRF → NN

see Fig 3 of node2vec for different embedding → different structure.

graph ___ label prop. / linear Regression

metadata ___ NN or MRF

maybe existance of other embedding to explore structure.

trade-off general, ok performance method v.s. specialized method

(if you want to explore in projects, that's good.
ex: they focus on prediction, but not how the parameters scale !)

8

Cool take-home messages.

— ML papers brags that their method is the best.
(but, truth is, some are good in one set of networks,
some good at another.
>> "no-free lunch" theorem.

— cross-validation in graphs.

— math V.S. @ NN-representation (slow/fast v. Bayesian / assumptions)
structure

— A lot of variations for RW methods
see Huang-Silva-Singh (2021)