Lecture 4: predicting missing attributes & missing links

- Most network data sets are incomplete

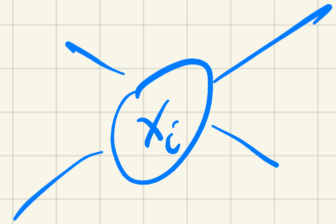
  why? $\Theta(n^2)$ possible interactions

- but, if what is missing correlates with what we observe, we can use one to predict the other

4 types of missing data prediction tasks for networks

* 1) predict missing node attributes

$G=(V,E)$ $\vec{x}$



mol. weight
functional behav.
eco sys func.
metab. func.

* 2) predict missing links

Some edges are missing

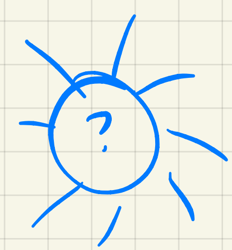
$G_0=(V,E_0)$ $\quad i,j \in V \times V - E_0$

3) predict missing link attributes

predict edge weights
or directions
or link attributes

4) predict missing nodes

predicting a bundle
of missing links



Predicting missing node attributes

let $G=(V,E)$ be a fully observed set of nodes and links

and $\vec{x}$ is node metadata $\quad x_i : \begin{cases} \text{categorical} & \text{red, blue, green} \\ \text{scalar} & \text{int., real,} \end{cases}$

$x_i = \emptyset$

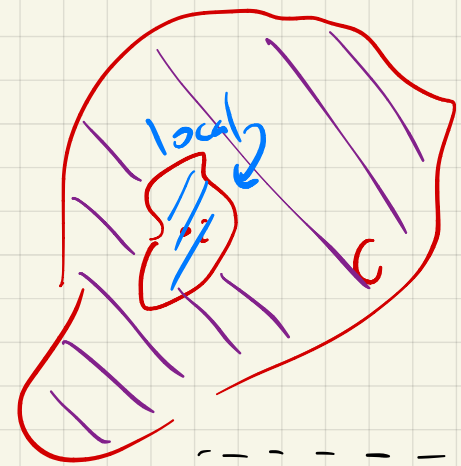
imagine full labeling $\vec{x}_*$

missingness function $f(\vec{x}_*) = \vec{x}$

$\uparrow$ observed, incomplete labeling

the baseline algorithm: we know nothing about f.

(global predictor)



$$\text{if } \underbrace{x_i = \emptyset}_{x_i \text{ is missing}}, \qquad \overbrace{x_i = \text{Uniform}(\vec{x} - \emptyset)}^{\text{predict this}}$$

why is this a "baseline"?

Assortative mixing and local smoothing

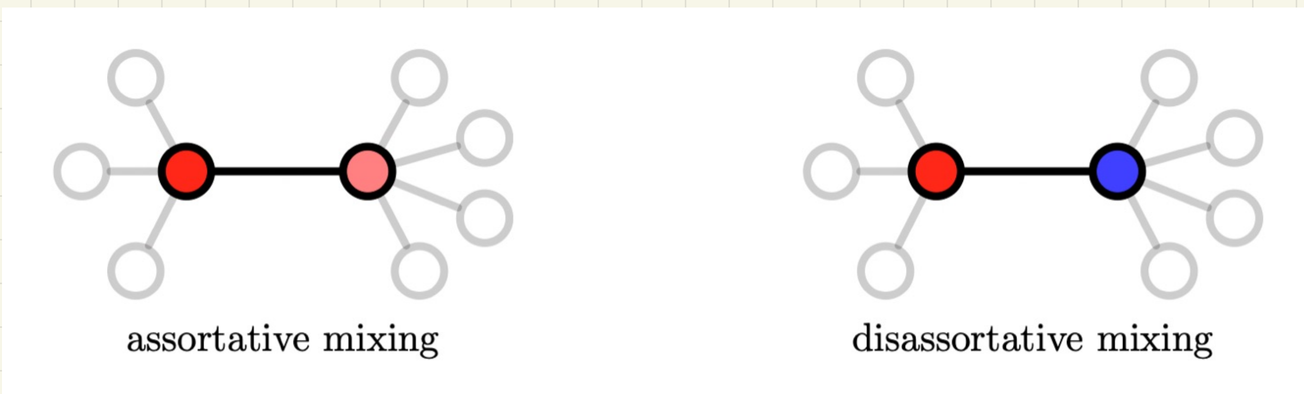
observation: in many networks, node attributes are **assortative**.

$i - j$ $\quad x_i \quad x_j \quad$ $i \quad j$

In some networks, node attributes are **disassortative**.

like links w/ like

like links w/ dislike



The idea

$$\boxed{\text{predict a missing attribute to be the average (scalar) or most common (categorical) value of its neighbors' non-missing attribute}}$$

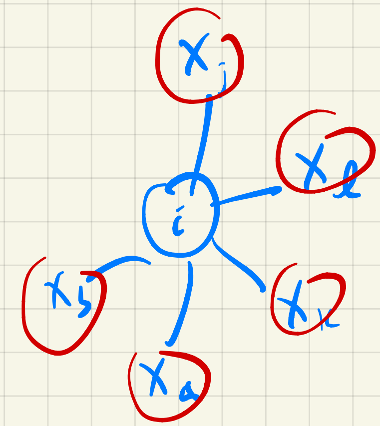
local smoothing algorithm

> Why is this a "local smoothing algorithm"?

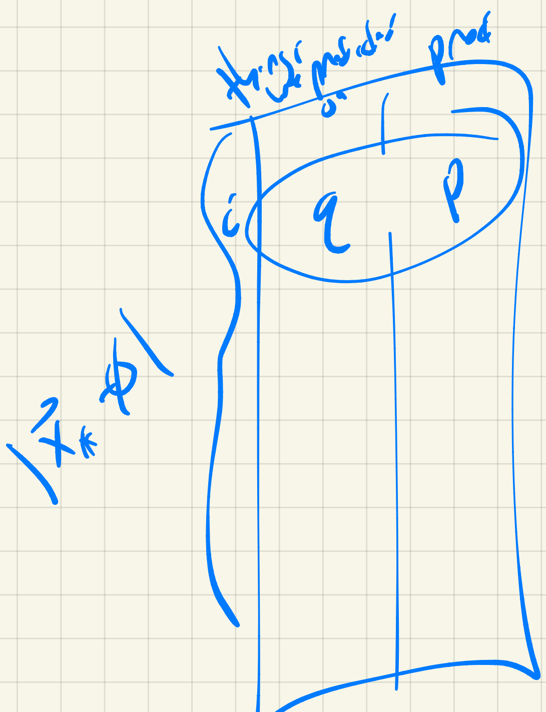
define a "neighborhood" function $\nu(i)$

$Z_0 \; \{x_{\nu(i)}\}$: the node attributes in i's neighborhood

$g : \{x_{\nu(i)}\} \to x_i$ our guess of i's metadata



if $x_i = \emptyset$, $\quad x_i = \begin{cases} \mathrm{mean}(\{x_{\nu(i)}\} - \emptyset) & \text{if } x \text{ is scalar} \\ \mathrm{mode}(\{x_{\nu(i)}\} - \emptyset) & \text{if } x \text{ is categorical} \end{cases}$.



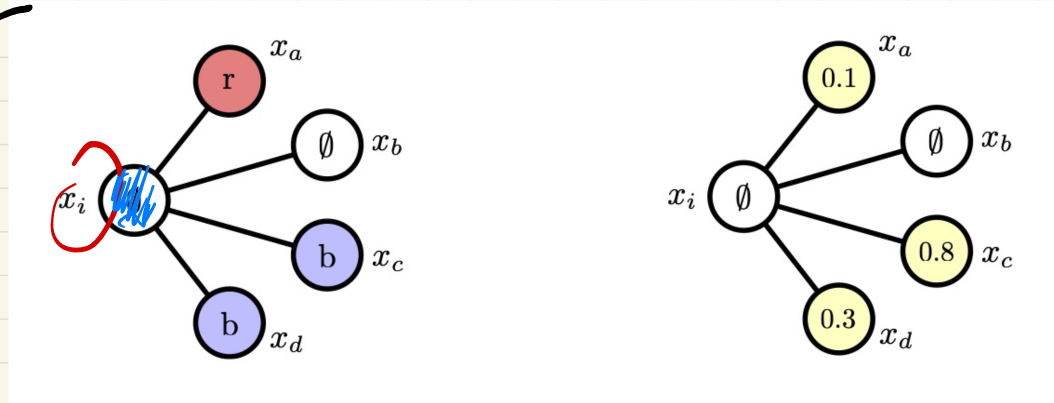
→ if $\{x_{\nu(i)} - \emptyset\} = \emptyset$ then revert to baseline

$\emptyset$

→ all predictions are synchronous

for example

$v(i) = \{a, b, c, d\}$ $\quad \{x_{v(i)}\} = \{r, \phi, b, b\}$



$\text{mode}(\{r, b, b\})$

$\rightarrow b$

$\{x_{v(i)}\} = \{0.1, \phi, 0.8, 0.3\}$

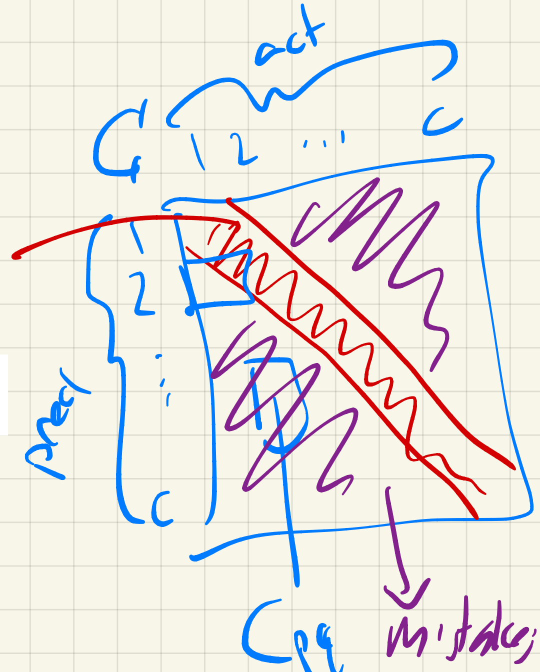
$\text{mean}(0.1, 0.8, 0.3)$

$\rightarrow 0.4$

Measuring performance: the confusion matrix C

let $x_i \in \{1, 2, \ldots, c\}$ ← # of unique labels

C_{pq} = the number of inputs with (*predicted* label p) and (*actual* label q)



what does C_{pp} count? or $\sum_p C_{pp}$?

what does $C_{p \neq q}$ count? or $\sum_p \sum_{q \neq p} C_{pq}$?

$$\text{accuracy (ACC)} = \frac{\overbrace{\text{number of correct predictions}}}{\text{number of inputs}} = \frac{\sum_p C_{pp}}{\sum_{p,q} C_{pq}} = \frac{1}{N}\sum_p C_{pp}$$

\# of correct things

\# of things

warning: what happens when one label is much more common than all others?

$c = 2$ (binary classifier)

$n_1 = 90\%$ of total

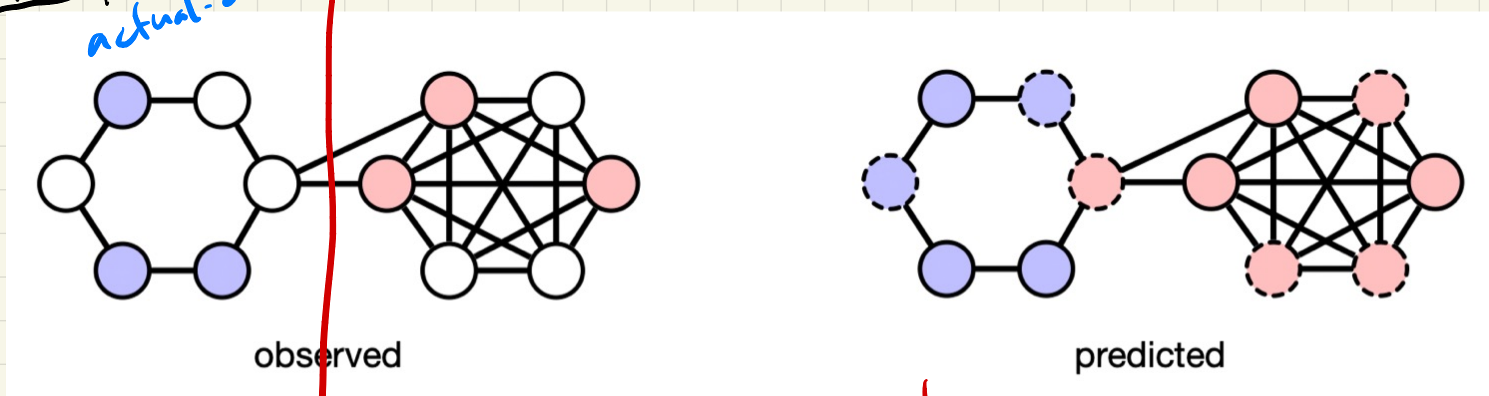
$n_2 = 10\%$

		act 1	act 2
pred	1	90	10
	2	0	0

for example

actual: b

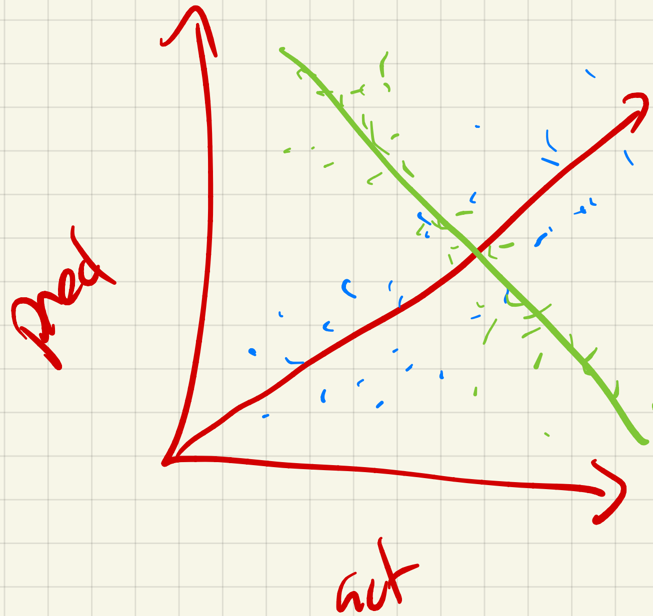
actual: a



C	act a	act b
pred a	3	1
pred b	0	2

$$ACC = \frac{5}{6} = 0.83$$

Measuring performance: scalars



$y = x$ perfect prediction

ACC: r^2 of $y=x$ line

a predictor that does the opposite of what it should

Predicting missing links

actual $G = (V, E)$

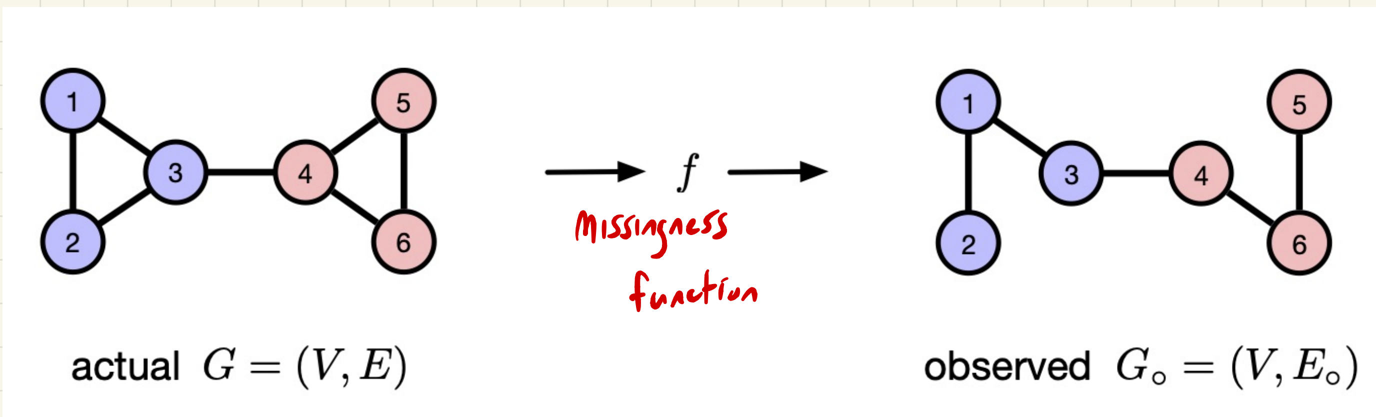
observed $G_o = (V, E_o)$ observed edge set

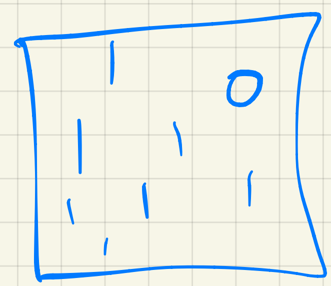
$G_o \subset E$

unconnected pairs $X = V \times V - E_o$

missing links $Y = E - E_o$

$f(E) = E_o$





$\cancel{X_i = \emptyset}$

non-edge

$\Theta(n^2)$ $A_{ij} = 0$

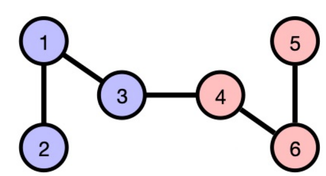
$O\left(\frac{1}{n}\right)$

→ $O(n)$ needles in a $\Theta(n^2)$ haystack

→ all predictors : $\text{score}(i,j) : i,j \in X \to \mathbb{R}$

a baseline algorithm

if $i, j \in X \quad \text{score}(i,j) = \underbrace{\text{Uniform}(0,1)}_{r}$



observed $G_\circ = (V, E_\circ)$

score table B

pairs in X

i	j	score(i,j)
1	5	r
1	4	r
1	6	r
2	3	r
2	6	r
2	4	r
2	5	r
3	5	r
3	6	r
4	5	r

scores

topological predictors & local smoothing

↳ simple functions of G_o

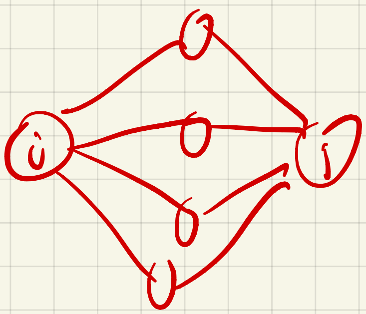
the idea

predict a missing link to occur where it would cluster with other (observed) edges

two ways (of many) to operationalize this idea:

1) Jaccard coefficient z_o

common neighbors $J(i,j) = \dfrac{|\nu(i) \cap \nu(j)|}{|\nu(i) \cup \nu(j)|}$

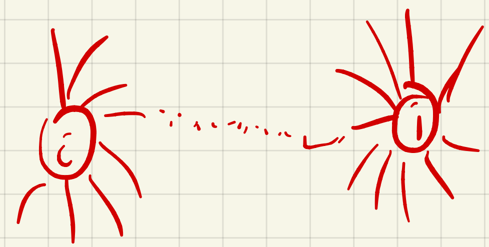


$\epsilon < \dfrac{1}{2k_{max}}$

$$\text{score}(i,j) = \text{Jaccard}(i,j) + \text{Uniform}(0, \epsilon)$$

2) degree product z_o

$D(i,j) = k_i k_j$



$\epsilon < 1$

$$\text{score}(i,j) = k_i k_j + \text{Uniform}(0, \epsilon)$$

an example

$$J(i,j) = \frac{3}{5+1} = \frac{1}{2}$$

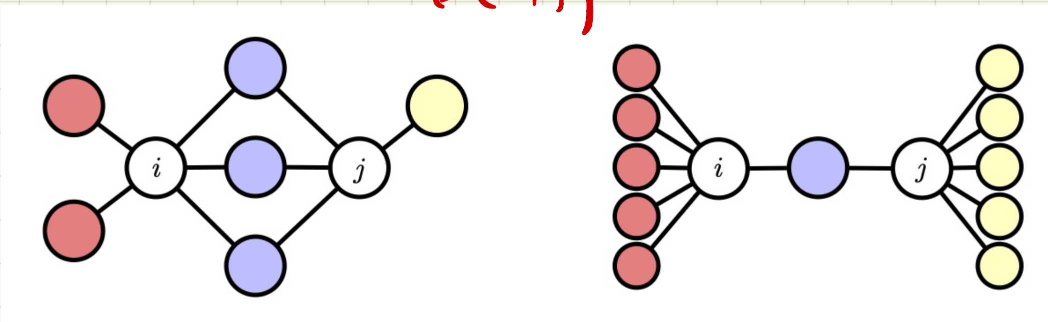
$$D(i,j) = 5 \times 4 = 20$$

$$J(i,j) = \frac{1}{6+5} = \frac{1}{11}$$

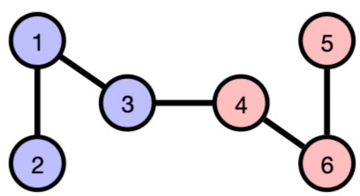
$$D(i,j) = 36 = 6 \times 6$$

$$\cancel{D(i,j) = 21 + \varepsilon}$$

$$\cancel{D(k,l) = 20 + \varepsilon}$$



another example



observed $G_\circ = (V, E_\circ)$

	i	j	Jaccard score(i,j)
Y	**4**	**5**	$1/2 + r$
Y	**2**	**3**	$1/2 + r$
X	3	6	$1/3 + r$
X	1	4	$1/3 + r$
X	1	5	r
X	1	6	r
X	2	6	r
X	2	4	r
X	2	5	r
X	3	5	r

	i	j	degree product score(i,j)
	1	4	$4 + r$
	1	6	$4 + r$
	3	6	$4 + r$
	1	5	$2 + r$
Y	**2**	**3**	$2 + r$
	2	6	$2 + r$
	2	4	$2 + r$
	3	5	$2 + r$
Y	**4**	**5**	$2 + r$
	2	5	$1 + r$

Measuring performance: the AUC

predicting missing links is a binary classifier

We can use Area Under the Curve (AUC)

nice properties

1) Scale invariant ⇝ absolute scores don't matter, only rankings

2) threshold invariant ⇝ general measure

Mathematically,

$\Theta(n^2)$ true negatives

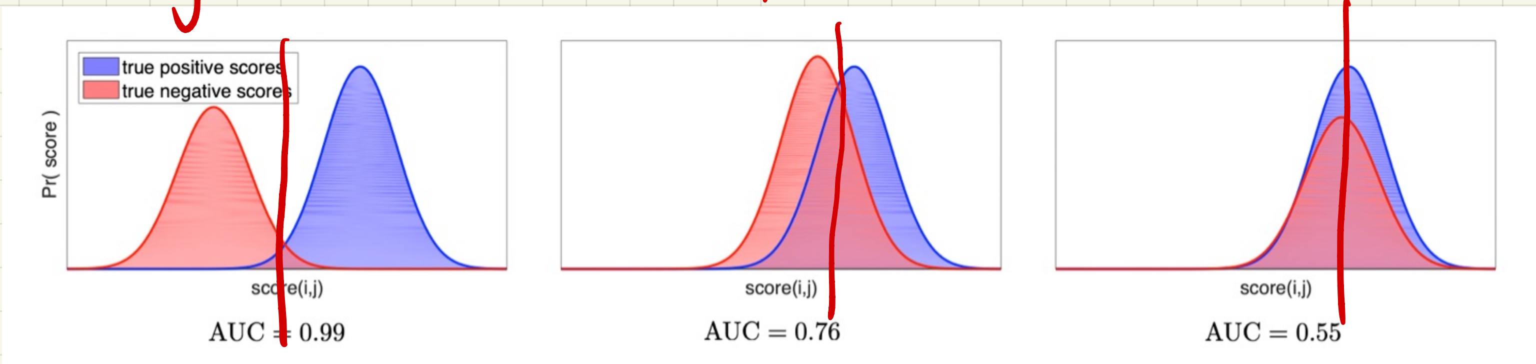
AUC = Pr[score(true positive) > score(true negative)]

$O(n)$ true positives

AUC ranges from $\frac{1}{2} \to 1$

good

okay-ish

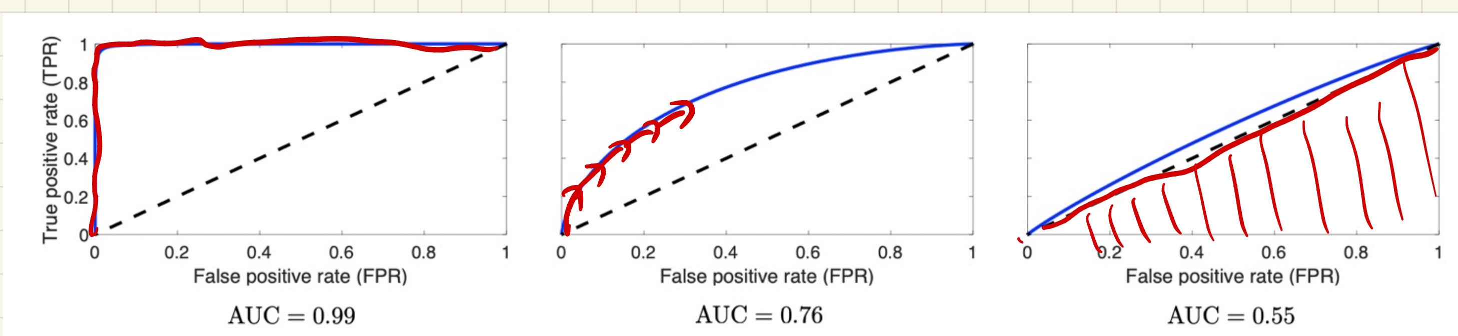


near perfect distinction

only slightly better than random

from radar!

the AUC is the integration of the $\underbrace{\text{ROC curve}}_{\text{TPR vs FPR}}$

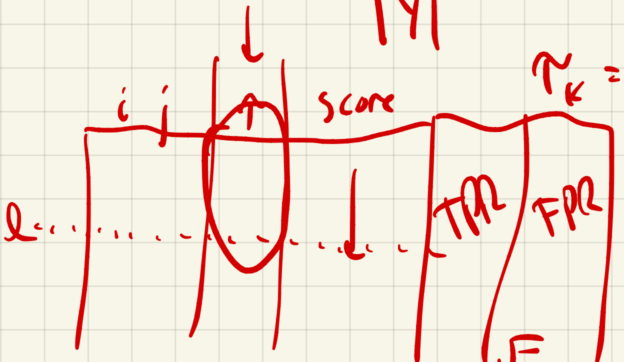


y=x line is AUC=1/2

$$TPR(\ell) = \frac{1}{\mathcal{I}} \sum_{k=1}^{\ell} \tau_k \qquad \mathcal{I} \leftarrow |Y|$$

$$FPR(\ell) = \frac{1}{\mathcal{J}} \sum_{k=1}^{\ell} 1 - \tau_k \qquad \mathcal{J} \leftarrow |X-Y|$$

$$\tau_k = \begin{cases} 0 & \text{if it's a TN} \\ 1 & \text{if TP} \end{cases}$$

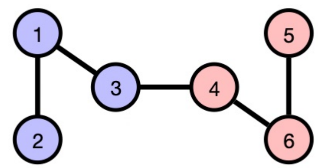


$$AUC = \sum_{\ell=1}^{\mathcal{J}} \underbrace{TPR(\ell)}_{Y} \times \underbrace{\left[FPR(\ell) - FPR(\ell-1)\right]}_{\Delta x}$$

* define $FPR(0) = 0$

(box-rule for numerical integration!)

an example

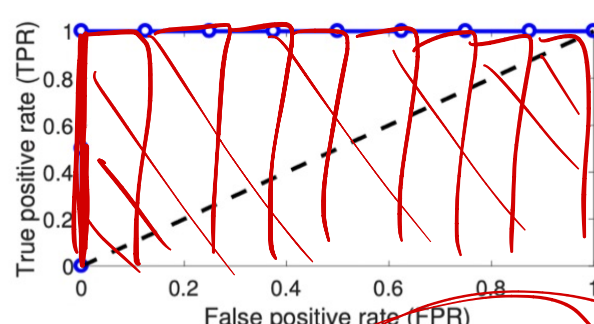


observed $G_\circ = (V, E_\circ)$

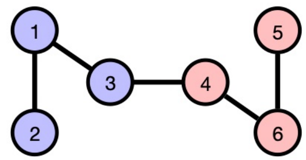
$$\text{Jaccard}(i,j) = \frac{|\nu(i) \cap \nu(j)|}{|\nu(i) \cup \nu(j)|}$$

$$\text{score}(i,j) = \text{Jaccard}(i,j) + \text{Uniform}(0,\epsilon)$$

i	j	τ_k	TPR_ℓ	FPR_ℓ	Jaccard score(i,j)
4	5	1	0.5	0.0	$1/2 + r$
2	3	1	1.0	0.0	$1/2 + r$
3	6	0	1.0	0.125	$1/3 + r$
1	4	0	1.0	0.250	$1/3 + r$
1	5	0	1.0	0.375	r
1	6	0	1.0	0.500	r
2	6	0	1.0	0.625	r
2	4	0	1.0	0.750	r
2	5	0	1.0	0.875	r
3	5	0	1.0	1.00	r



Jaccard coefficient, AUC = 1.00

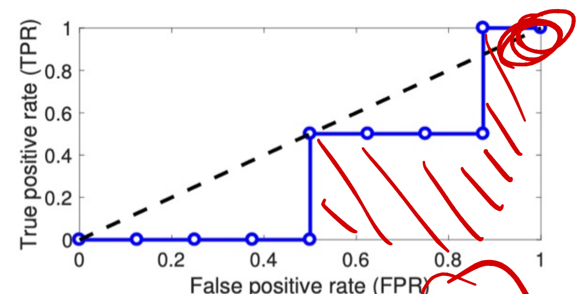


observed $G_\circ = (V, E_\circ)$

degree product $= k_i k_j$

$$\text{score}(i,j) = k_i k_j + \text{Uniform}(0,\epsilon)$$

i	j	τ_k	TPR_ℓ	FPR_ℓ	degree product score(i,j)
1	4	0	0.0	0.125	$4+r$
1	6	0	0.0	0.250	$4+r$
3	6	0	0.0	0.375	$4+r$
1	5	0	0.0	0.500	$2+r$
2	3	1	0.5	0.500	$2+r$
2	6	0	0.5	0.625	$2+r$
2	4	0	0.5	0.750	$2+r$
3	5	0	0.5	0.875	$2+r$
4	5	1	1.0	0.875	$2+r$
2	5	0	1.0	1.000	$1+r$

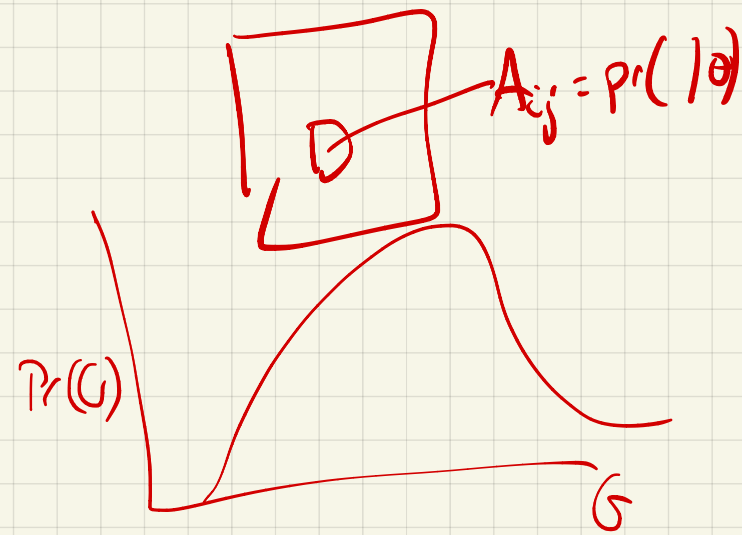


degree product, AUC = 0.31

the wild world of missing link predictors

3 classes:

1) topological : based on local structural patterns around i, j
[often local, sometimes global]



2) model-based: learn some $\Pr(G \mid \theta)$
[usually global]

3) embeddings : embed G_0 into $\mathbb{R}^d$, predict $i \to j$ if $d(i,j)$ is small
[global]