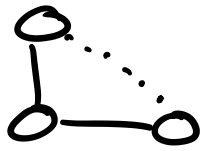
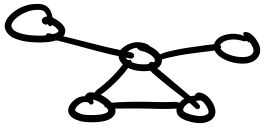
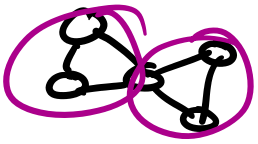
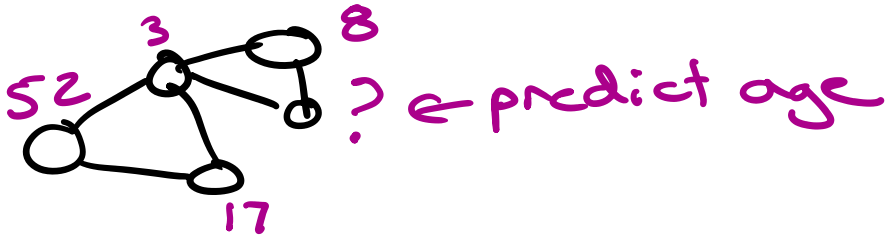
Graph mining classic topics:

1. Link prediction 
2. Centrality  ← which vertex is important
3. clustering / CD 
4. Vertex classification 

← predict age

Vertex Labeling problem

Given graph $G = \{V, E, W, Y\}$

V = vertex set

E = edge set

W = edge weights

Y = vertex labels

Ex: social networks

V = people

V = people

E = friendships

W = strength of friendships

Y = demographic information

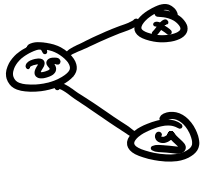
The Problem:

Given $G = \{V, E, W, Y_e\}$

(Y_e: existing labels)

Predict Y_u ← unlabeled data

Approach: Iterative classification



Features

Age of v

Gender of v

Politics of v

Avg. age of $N(v)$

Proportions of M/F/NB/etc. of $N(v)$

Generally: metadata of v, $N(v)$

Basic classification problem:

Basic classification problem:

We have Y_l, pieces of data

We can construct features

Train classifier using Y_l, features

Predict Y_u, unlabeled data

Our general:

Construct Φ_l, Φ_u from G
(features)

Train f on Φ_l $\left(\min_{v_i \in V} \|f(\Phi_l) - Y_l\|\right)$
(classifier)

For some #iterations:

Predict $Y_u = f(\Phi_u)$

update Φ_u

Return final Y_u

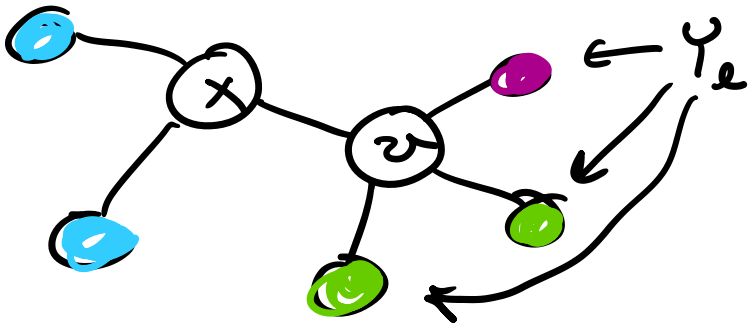
Classifier = Label Propagation

Y_l = ground truth labels

Φ_v = labels in $N(v)$

ϕ_v = labels in $N(v)$

f = return label with max count in $N(v) \mid \phi$



$\phi_v = \{\bullet = 2, \bullet = 1\}$

$f(\phi_v) = \{\bullet\}$

Naive Bayes Classifier

$\bar{x}$ = features

$\bar{x}_v$ = features for vertex v

$\bar{x}_v = \{x_{v_1}, x_{v_2}, \ldots x_{v_n}\}$

To classify some v as class C_i

$$\max_{i \in C} P(C_i \mid \bar{x}_v)$$

↑ highest probability over all C_i given features $\bar{x}_v$ of v

Two things first:

Bayes' Theorem → $P(A \mid B) = \dfrac{P(A)P(B \mid A)}{P(B)}$

Bayes' Theorem → $P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)}$

Chain rule → $P(A \cap B) = P(A) P(B|A)$

for 3+ events

$$P(A_1 \cap A_2 \cap \dots \cap A_k) = P(A_1 | A_2 \cap \dots \cap A_k) * P(A_2 \cap \dots \cap A_k)$$

$$P(C_i | \bar{x}_v) = \frac{P(C_i) P(\bar{x}_v | C_i)}{\cancel{P(\bar{x}_v)}}$$

Note: constant

$$P(C_i) P(\bar{x}_v | C_i) = P(C_i \cap \bar{x}_v)$$

$$P(C_i \cap \bar{x}_v) = P(C_i \cap x_{v_1} \cap x_{v_2} \dots x_{v_k})$$

$$\hookrightarrow P(x_{v_1} \cap x_{v_2} \cap \dots x_{v_k} \cap C_i)$$

chain rule again

$$= P(x_{v_1} | x_{v_2} \dots x_{v_k} \cap C_i) P(x_{v_2} \dots x_{v_k} \cap C_i)$$

Chain rule recursively

$$= P(x_{v_1} | x_{v_2} \dots C_i) P(x_{v_2} | x_{v_k} \dots C_i) \dots P(C_i)$$

Naive Bayes assumption:

→ all x_j are indepent

$$P(x_{v_1} | C_i) P(x_{v_2} | C_i) \dots P(C_i)$$

$$= P(C_i) \prod_{j=1}^{k} P(x_{v_j} | C_i)$$

$$Y_u(\bar{x}_v) = \max_{i \in C} P(C_i) \prod_{j=1}^{k} P(x_{v_j} | C_i)$$

$P(C_i)$ = ratio of C_i labels

$$P(C_i) = \frac{|C_i|}{|V_e|}$$

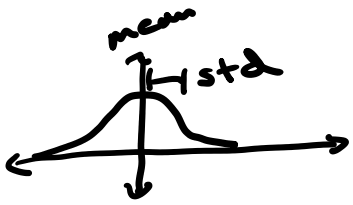
$P(x_{v_j} | C_i)$ = over all C_i labels, how often feature x_{v_j} shows up

→ if discrete → easy

if continuous

if continuous

↳ we need to assume a given distribution of feature values



We can assume e.g. a normal distribution for a given feature

$P(x_{vj} | C_i)$ = the probability that the feature value x_{vj} is sampled from some distribution (mean, variance) calculated from known labeled x_{uj} features $u \in V_l$

Random Walks

Idea: the probability of some $v \in V_u$ assuming some label C_i is the probability of a random walk from v ends on some $u \in V_l$ with label C_i

c_i

$P = D^{-1}W$

P = transition prob. matrix

D = diagonal degree matrix

W = weighted adjacency matrix

$P_{ij} \in P \rightarrow$ prob. of random walk from $i \rightarrow j$

$P = D^{-1}W$ only gives one step of a random walk

$\lim_{t \to \infty} P^t$ = gives us steady-state walk probability distribution

However: what if we want to stop our walk if we land on a labeled vertex?

We can modify our P

$P_i = e_i$ ← identity if $i \in V_l$

$P_j = (D^{-1}W)_i$ if $j \in V_u$

We can order vertices from V_l first and then V_u second

we stop here forever ← we don't do this

we stay here forever ← we don't do this

$$P = \begin{pmatrix} P_{ll} & P_{lu} \\ P_{ul} & P_{uu} \end{pmatrix} = \begin{pmatrix} I & 0 \\ P_{ul} & P_{uu} \end{pmatrix}$$

we still $\lim_{t \to \infty} P^t$

$$P^\infty = \begin{pmatrix} I & 0 \\ (I - P_{uu})^{-1} P_{ul} & P_{uu}^\infty \end{pmatrix}$$

↖ goes to zero as we don't stop a V_u

$$P^\infty = \begin{pmatrix} I & 0 \\ (I - P_{uu})^{-1} P_{ul} & 0 \end{pmatrix}$$

Y_l = probability distribution over same class label

vertex in labeled set ← ith column

$v_i \rightarrow Y_{l_i} = [0 \ldots 1 \ldots 0]$

unlabed

$v_j \rightarrow Y_{u_j} = [0.1 \ldots 0.2 \ldots 0.05]$

Prediction matrix

$$Y_u = (I - P_{uu})^{-1} P_{ul} Y_l$$

we want to find a max. prob. for our prediction

$$\underset{C_i \in C}{\operatorname{argmax}} \; Y_{ui}$$

Can also do this iteratively

$$Y_u^{t+1} = P_{ul} Y_l + P_{uu} Y_u^t$$