Discrete optimization:

We will focus on two motivating problems.

- Matchings
- MRF's

Matching:

- important cases can be solved with dynamic programming
  - eg. pictorial structure models
- Even these cases have problems
  - 2-leg issue with pictorial structs.
- many other cases
  - weighted bipartite matching.

Improved pictorial structure as ~~binary~~ 0-1 quadratic form

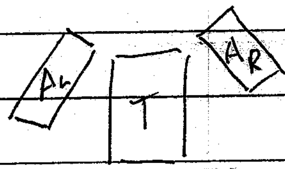


Image segments

model

- can represent a match with a constrained 0-1 vector

$$x^T = [\underbrace{\quad\quad\quad}_{A_1} \vdots \underbrace{\quad\quad\quad}_{A_2} \vdots \underbrace{\overbrace{\quad\quad\quad}^{\#\text{ image segments}}}_{T}]$$

$$x \in \{0,1\}^k$$

- now there is exactly one image seg per body seg, so:

$$[1\,1\,1\,1\,1 \cdots \vdots\; 0 \;\vdots\; 0] \cdot x = 1$$

$$[0 \;\vdots\; 1 \;\vdots\; 0] \cdot x = 1$$

$$[0 \;\vdots\; 0 \;\vdots\; 1] \cdot x = 1$$

Now, build a cost model

- 2 kinds of cost

$$Cost[M_i \to I_j] \text{ — unary}$$

$$Cost[M_i \to I_j, M_k \to I_l] \text{ - binary.}$$

- Pictorial structure:
  - binary costs form a forest
  - DP works

- More general.

$$x_{p(u,v)} = \begin{cases} 1 & \text{if } M_u \to I_v \\ 0 & \text{otherwise} \end{cases}$$

just an indexing scheme

so cost is

$$\sum_{u,v} x_{p(u,v)} \cdot Cost[M_u \to I_v] +$$

$$\sum_{u,v,r,s} x_{p(u,v)} \, x_{p(r,s)} \, C[M_u \to I_v, M_r \to I_s$$

Notice that we could incorporate

$$\text{Cost}\left[M_i \to I_j,\ M_k \to I_\ell\right] \quad \text{etc}$$

easily.

Abstract form:

$$\min \quad \frac{x^T A x}{2} + b^T x$$

$$\text{s.t.} \quad Lx = c$$

$$x \in \{0,1\}^K$$

We can get to this form in ~~other~~ ways, too.

Markov Random Field:

- We have a collection of discrete RVs F_i, and a neighbourhood structure

$P(F_i \mid \text{all others})$

$$= P(F_i \mid \text{Neighbours of } F_i)$$

- We have observations at each location O_i, and a model $P(O_i \mid F_i)$

$$P(O_i \mid \text{all } F) = P(O_i \mid F_i)$$

<u>Problem</u>: given O_i, models, find

$$F = \text{argmax } P(F, O)$$

<u>Thm</u>:

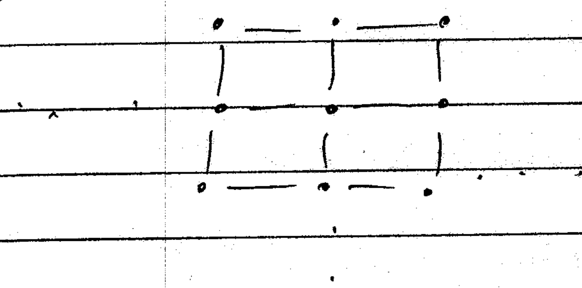
$$P(F) \propto \exp\Big(-\sum_{\text{cliques}} \Phi(F \text{ in clique})\Big)$$

(Hammersly - Clifford) clique potentials

We will confine attn to cases where Φ takes a <u>binary</u> form.

These are *very* common in vision

e.g. Neighbourhood structure:



i.e. we compare F_{ij} to its 4-neighbours because we expect it to be similar

$-\sum_{\text{cliques}} \phi$ becomes

$$\sum_i \left[\sum_{u \in N(i)} \phi(F_i, F_u) \right]$$

- typically, big if these two are different
  small if similar.

- e.g. image reconstruction:

  - we observe noisy pixel values.
  - we know neighbours are similar.

eg <u>Stereopsis</u>:

(Simplified)

- we want a depth map for I_L
- discretize
- given depth (disparity) at $I_{L(i,j)}$ we predict

  $I_{R\ \text{esp}(i,j)}$
- these should be similar — <u>Unary</u>
- neighbouring depths should be similar — <u>Binary</u>

<u>Abstract form:</u>

We have

$$F_i = \{1 \cdots K\}$$

$$\text{cost} = \sum_i \phi_u(F_i) + \sum_i \left[\sum_{u \in N(i)} \phi_b(F_i, F_u) \right]$$

We can turn this into a 0-1 quadratic form with linear constraints

$$x = \Big[\underbrace{\overset{1 \cdots K}{\longleftrightarrow}}_{F_1} \Big| \underbrace{\overset{1 \cdots K}{\longleftrightarrow}}_{F_2} \Big| \quad \cdots \quad \Big]$$

- each F_i can have only 1 value

$$Lx = 1$$

- cost write $x_{p(u,v)} = \begin{cases} 1 & \text{if } F_u = v \\ 0 & \text{otherwise} \end{cases}$

then

$$\sum_{u,v} x_{p(u,v)} \cdot \phi_{\text{unary}}\left[F_u = v\right] + \sum_{u,v} \sum_{\substack{r \in N_u \\ s}} \left[x_{p(u,v)} \cdot x_{p(r,s)} \, \phi_{\text{binary}}\left(F_u = v, F_r = s\right) \right]$$

So again, we have

$$\min \; \frac{x' A x}{2} + b' x$$

$$\text{st} \quad Lx = c$$

$$x \in \{0,1\}^K$$

Consider this class of problem

Max $\quad x^T \frac{A}{2} x + b^T x$

$x \in \{0,1\}$ (removed linear constraints)

can turn this into Linear Program:

write $\quad \delta_{u(i,j)} = x_i \cdot x_j$

then: $\quad \delta_{u(i,j)} \leq x_i$

$\delta_{u(i,j)} \leq x_j$

$\delta_{u(i,j)} \geq x_i + x_j - 1$

$\delta_{u(i,j)} \in \{0, 1\}$

max $\quad W_A^T \delta + b^T x$

st $\quad L \begin{bmatrix} \delta \\ x \end{bmatrix} = C$

$\delta_{u(i,j)} \in \{0, 1\} \qquad x_i \in \{0,1\} \quad x_j \in \{0,1\}$

There are some interesting features

1) extremum is at vertex of constraint polytope.

⇒ if we can guarantee this is ~~the~~ (0,1) then we could solve continuous LP, and soln would meet 0,1 constraints

Polytope:

$$Mx \geq c$$

vertex:

$$\underbrace{R(M)}_{n \text{ rows of } M} \overbrace{x}^{n \text{ dim}} = \underbrace{\hat{c}}_{n \text{ relevant entries of } c}$$

Sufficient conditions:

$$\begin{cases} \hat{c} \text{ is } \{0, 1\} \\ R(M)^{-1} \text{ has all entries } 0, -1, \text{ or } 1 \text{ for any } R(M). \end{cases}$$

$R(M)^{-1}$ — entries are ratio's of minors of M

Conditions: (Sufficient)

- any minor of M is $= 0, 1, -1$
- c is $0, 1, -1$

Back to original problem:

$$\max \quad w^T \delta + b^T x$$

$$\text{st} \quad L \begin{bmatrix} \delta \\ x \end{bmatrix} \geq c$$

this is

$\delta \in \{0,1\}$
$x \in \{0,1\}$

(1) Relax to

~~$\delta \geq 0$, $\delta \leq 1$~~

$0 \leq \delta \leq 1$

$0 \leq x \leq 1$

(2) constraints

$$\hat{L}\, x \geq \hat{c}$$

$$\hat{L} = \left[\begin{array}{l} \delta_{u(i,j)} \leq x_i \\ \\ \delta_{u(i,j)} \leq x_j \\ \\ \delta_u \leq 1 \\ \delta_u \geq 0 \\ x_i \leq 1 \\ x_i \geq 0 \\ \hline \delta_u \geq x_i + x_j - 1 \end{array}\right]$$

→ this block is Totally Unimodular (all minors 0, 1, -1)

→ this ~~[illegible]~~

SPECIAL CASE

$$A_{ij} = W_{u(i,j)} \geq 0$$

in this case, if $x_i = 1$, $x_j = 1$, then

$\delta_{u(i,j)} = 1$, because we are max

In SPECIAL CASE,

we can

- Drop the $\delta_u \geq x_i + x_j - 1$ block
- Solve the relaxed system
- Solution will be
  - $\{0,1\}$
  - maximal

Unfortunately, this is a special property (though it does occur, see papers)

<u>More general case:</u>

(not good) - Solve relaxed, then quantize

- <u>Other methods</u>

Unfortunately, not most efficient alg.