CS340: MACHINE LEARNING

REVIEW FOR FINAL

KEVIN MURPHY

OUTLINE

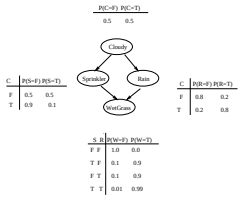
- Directed graphical models
- Undirected graphical models
- State estimation
- MCMC
- Gaussians
- Mixture models

CHAIN RULE

See Alpaydin sec 3.7

$$X_i \perp X_{non-desc(i)} | X_{\pi_i} \Rightarrow X_i \perp X_{anc(i)} | X_{\pi_i} \quad (1)$$

$$\Rightarrow p(X_{1:d}) = \prod_i p(x_i | x_{\pi_i}) \quad (2)$$



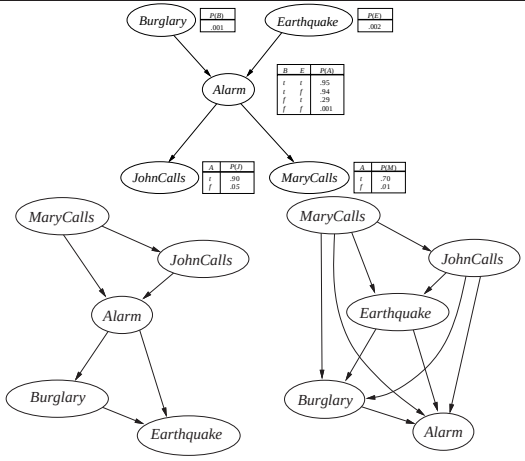
$$P(C,S,R,W) = P(C)P(S|C)P(R|S,C)P(W|S,R,C) \text{ chain rule} \quad (3)$$

$$= P(C)P(S|C)P(R|\not{S},C)P(W|S,R,C) \text{ since } S \perp R|C \quad (4)$$

$$= P(C)P(S|C)P(R|\not{S},C)P(W|S,R,\not{C}) \text{ since } W \perp C|S,R \quad (5)$$

$$= P(C)P(S|C)P(R|C)P(W|S,R) \quad (6)$$

ORDER MATTERS



(a) (b)

CPDs

- Consider $p(X_i|X_{\pi_i})$. Let $U = X_{\pi_i}$.
- Tabular

$$p(X_i = k|U = j) = \theta_{ijk}$$

- Sigmoid / logistic

$$p(X_i = 1|U = u) = \sigma(w_i^T u), \quad \sigma(z) = 1/(1 + e^{-z})$$

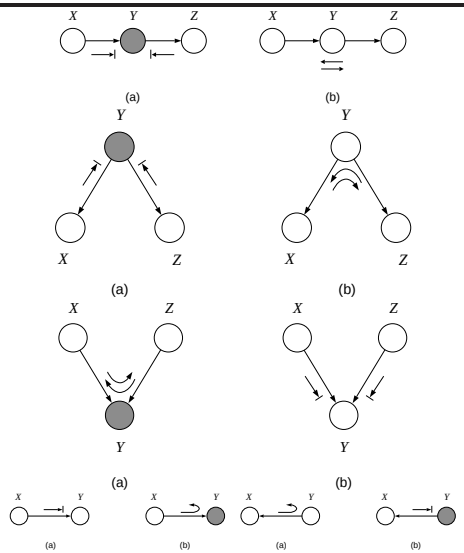
- Noisy-or

$$p(X_i = 1|U = u) = \prod_{j:u_j=1} q_{ij}$$

- Linear Gaussian

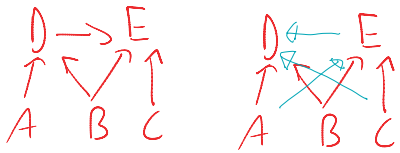
$$p(X_i = x|U = u) = \mathcal{N}(x|w_i^T u, \sigma_i^2)$$

Bayes ball/ d-separation

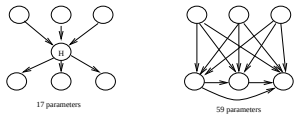


Graph manipulation

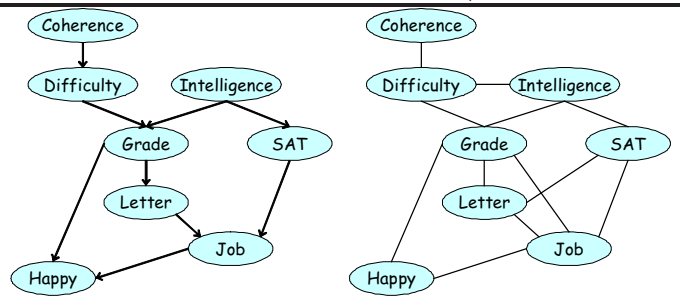
arc reversal



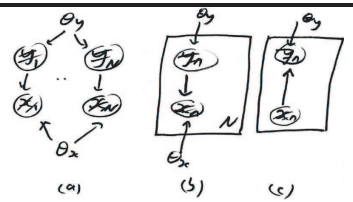
node elimination



Converting a DGM to a UGM (moralization)



PLATES



$$p(D|\theta) = \prod_{n=1}^{N} p(y_n|\theta_y)p(x_n|y_n,\theta_x) \tag{7}$$

PARAMETER ESIMTATION FOR COMPLETE DATA

Factored prior + factored likelihood + complete data $\Rightarrow$ factored posterior $\Rightarrow$ problem decomposes into d separate problems
eg for MLE

$$\hat{\theta}_i = \arg\max_\theta \sum_n \log p(X_i^n|X_{\pi_i}^n,\theta) \tag{8}$$

eg for MAP

$$\hat{\theta}_i = \arg\max_\theta \log p(\theta) + [\sum_n \log p(X_i^n|X_{\pi_i}^n,\theta)] \tag{9}$$

eg for Bayes

$$p(\theta_i|\mathcal{D}) = p(\theta_i)\prod_n p(X_i^n|X_{\pi_i}^n,\theta_i) \tag{10}$$

** EM FOR DGMS

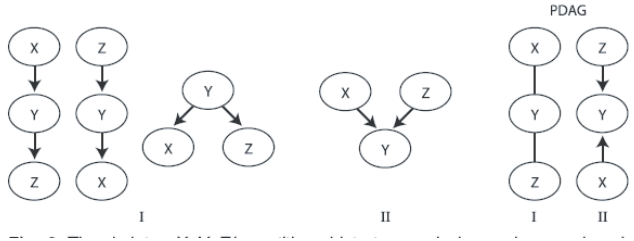
- E step:

$$p(X_i^n, X_{\pi_i}^n|D_n,\theta^{old}) \tag{11}$$

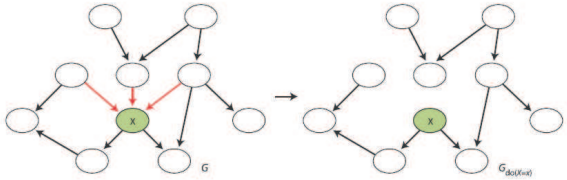
- M step:

$$\hat{\theta}_i = \arg\max_\theta \sum_n < \log p(X_i^n|X_{\pi_i}^n,\theta) > \tag{12}$$

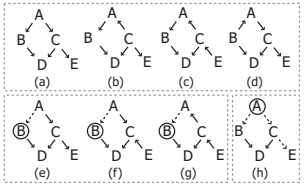
MARKOV EQUIVALENCE/ PDAGS



PERFECT INTERVENTIONS



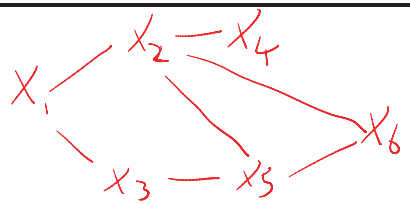
** INTERVENTION EQUIVALENCE



OUTLINE

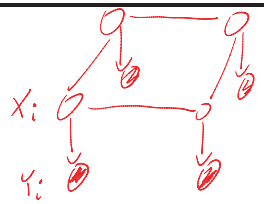
- Directed graphical models $\surd$
- Undirected graphical models
- State estimation
- MCMC
- Gaussians
- Mixture models

HAMMERSLEY CLIFFORD THEOREM



$$p(x_{1:6}) = \frac{1}{Z}\psi_{12}(x_1, x_2)\psi_{13}(x_1, x_3)\psi_{24}(x_2, x_4)\psi_{35}(x_3, x_5)\psi_{256}(x_2, x_5, x_6) \quad (13)$$

ISING MODEL



$$p(x,y) = p(x)p(y|x) \tag{14}$$

$$= \left[\frac{1}{Z}\prod_{<ij>}\psi_{ij}(x_i,x_j)\right]\left[\prod_i p(y_i|x_i)\right] \tag{15}$$

$$\psi_{ij}(x_i,x_j) = \exp[J_{ij}x_ix_j] = \begin{pmatrix} e^{J_{ij}} & e^{-J_{ij}} \\ e^{-J_{ij}} & e^{J_{ij}} \end{pmatrix} \tag{16}$$

CONDITIONAL INDEPENDENCE

$A \perp B|C$ iff all nodes in A are separated from all nodes in B, after we remove all nodes in C

** PARAMETER ESTIMATION

Finding parameter estimates (eg MLEs) is hard because the likelihood does not decompose into separate problems, because of the global normalizing constant Z

$$p(x|\theta) = \frac{1}{Z(\theta)}\prod_{c\in C}\psi_c(x_c|\theta_c) \tag{17}$$

OUTLINE

- Directed graphical models $\surd$
- Undirected graphical models $\surd$
- State estimation
- MCMC
- Gaussians
- Mixture models

KINDS OF INFERENCE

1. **State estimation**: inferring $p(X|y, \theta, G)$.
2. **Parameter estimation (learning)**: inferring $p(\theta|y, G)$.
3. **Model selection (structure learning)**: inferring $p(G|y)$.

KINDS OF STATE ESTIMATION

- sum-product

$$p(x_i) = \frac{1}{Z} \sum_{x_{-i}} \prod_c \psi_c(x_c) \tag{18}$$

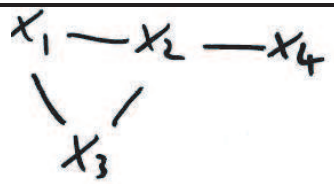
- max-product

$$x^{MAP} = \arg\max_x \prod_c \psi_c(x_c) \tag{19}$$

- max-sum-product

$$x_i^{MMAP} = \arg\max_{x_i} \sum_{x_{-i}} \prod_c \psi_c(x_c) \tag{20}$$

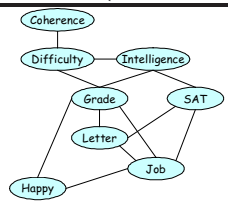
BRUTE FORCE



$$p(X_{1:4}) = \frac{1}{Z} \psi_{123}(X_1, X_2, X_3) \psi_{24}(X_2, X_4) \tag{21}$$

Build table and find marginals by enumeration.

VARIABLE ELIMINATION (DYNAMIC PROGRAMMING)



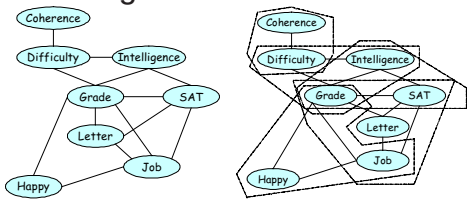
$$P(J) = \sum_L \sum_S \psi_J(J,L,S) \sum_G \psi_L(L,G) \sum_H \psi_H(H,G,J) \sum_I \psi_S(S,I) \psi_I(I) \sum_D \psi_(G,I,D) \underbrace{\sum_C \psi_C(C) \psi_D(D,C)}_{\tau_1(D)} \tag{22}$$

$$= \sum_L \sum_S \psi_J(J,L,S) \sum_G \psi_L(L,G) \sum_H \psi_H(H,G,J) \sum_I \psi_S(S,I) \psi_I(I) \underbrace{\sum_D \psi_(G,I,D) \tau_1(D)}_{\tau_2(G,I)} \tag{23}$$

$$\ldots \tag{24}$$

MAXIMAL CLIQUES IN INDUCED GRAPH

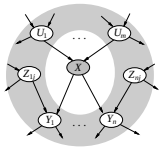
Largest maxclique is G, L, S, J so treewidth is 4-1=3. Other orders may produce larger treewidth.



GIBBS SAMPLING FOR DGMS

Full conditional

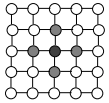
$$p(X_i = \ell | x_{-i}) \propto p(X_i = \ell | Pa(x_i)) \prod_{y_j \in ch(X_i)} p(y_j | Pa(y_j)) \quad (25)$$



GIBBS SAMPLING FOR PAIRWISE UGMS

Full conditional

$$p(X_i = \ell | x_{-i}) \propto \prod_{j \in N_i} \psi_{ij}(X_i = \ell, x_j) \quad (26)$$



OUTLINE

- Directed graphical models $\surd$
- Undirected graphical models $\surd$
- State estimation $\surd$
- MCMC
- Gaussians
- Mixture models

$$I = \int h(x)p(x)dx \approx \frac{1}{S}\sum_{s=1}^{S} h(x^{(s)}) \tag{27}$$

Build a Markov chain whose stationary distribution is proportional to the target, $\pi(x) \propto p(x)$. Then samples from this chain can be used for MC integration.

1. Initialize X_0 arbitrarily.
2. For $s = 0, 2, \ldots$
   (a) Generate a proposed state $x' \sim q(x'|x_s)$
   (b) Evaluate the acceptance propability

$$\alpha = \frac{\pi(x')q(x|x')}{\pi(x)q(x'|x)} = \frac{\pi(x')/q(x'|x)}{\pi(x)/q(x|x')} \tag{28}$$

$$r(x'|x) = \min\{1, \alpha\} \tag{29}$$

   (c) Set

$$X_{s+1} = \begin{cases} x' & \text{with probability } r \\ x_s & \text{with probability } 1-r \end{cases} \tag{30}$$

Proposal $q(x'|x)$ is symmetric, so

$$\alpha = \frac{\pi(x')}{\pi(x)} \tag{31}$$

$$r(x'|x) = \min\{1, \alpha\} \tag{32}$$

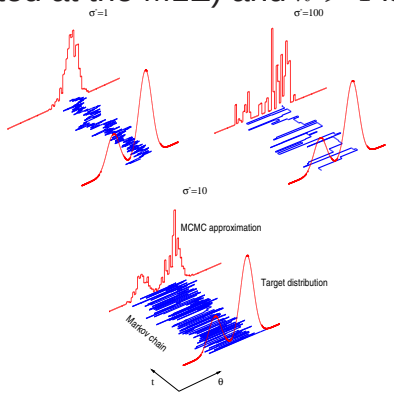
If $\pi(x') > \pi(x)$, we always accept, otherwise we may accept.

MH: WHY SO USEFUL?

- Don't need to be able to compute Z, i.e. only need $p'(x) = p(x)/Z$.
- Statistical efficiency is (in principle!) independent of the dimension of x (does not suffer from the curse of dimensionality)
- Can use any mixture of heuristics as proposals (so long as it is possible to reach all states), so good way to combine different techniques into coherent framework.

PROPOSAL MATTERS

If using Gaussian proposal, $q(x'|x) = \mathcal{N}(x'|x, \Sigma)$, must pick Σ carefully. Can use $\Sigma = kH$, where H is the Hessian of the log likelihood (computed at the MLE) and $k > 1$ is a scale factor.



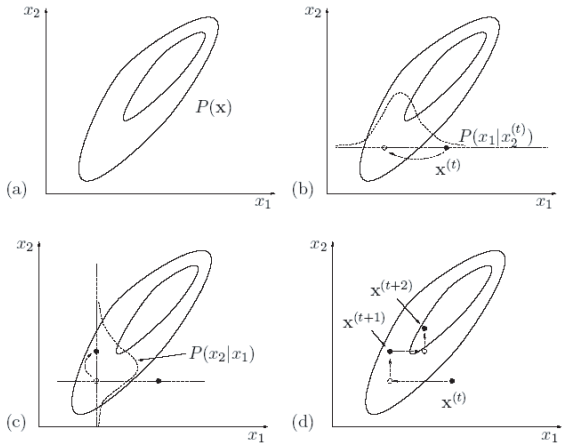
GIBBS SAMPLING

Special case of MH which is useful when the full conditionals are easy to sample from (eg in many graphical models)

1. $x_1^{s+1} \sim p(x_1|x_2^s, \ldots, x_D^s)$
2. $x_2^{s+1} \sim p(x_2|x_1^{s+1}, x_3^s, \ldots, x_D^s)$
3. $x_i^{s+1} \sim p(x_i|x_{1:i-1}^{s+1}, x_{i+1:D}^s)$
4. $x_D^{s+1} \sim p(x_D|x_1^{s+1}, \ldots, x_{D-1}^{s+1})$

GIBBS FOR 2D GAUSSIANS

Alternately sample from $p(x_1|x_2)$ and $p(x_2|x_1)$



GIBBS: GOOD AND BAD

Good

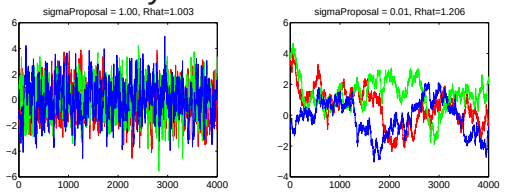
- No need to design proposal
- Acceptance rate $\alpha = 1$

Bad

- Can be slow since only updates one variable at a time (eg for Gaussians, axis parallel moves)

CONVERGENCE

Can only use the samples x^s once the chain has converged to its stationary distribution. How detect this? Use trace plots.



SIMULATED ANNEALING

Similar to Metropolis except we gradually change the target distribution from smooth to peaky.
Let $\pi_s(x) = \pi(x)^{1/T_s}$ be the target at step s, where T_s is the temperature. Let $\pi(x) = exp(-E(x))$ be the target defined in terms of energy. Then

$$\alpha = \frac{\pi(x')^{1/T_s}}{\pi(x)^{1/T_s}} \tag{33}$$

$$= \exp\left((E(x') - E(x))/T_s\right) \tag{34}$$

We can maximize the probability or minimize the energy by cooling T_s.
If $E_s(x') < E_s(x)$ then we always accept, otherwise we accept with a probability that depends on $E_s(x') - E_s(x)$: at large temperatures we are more willing to go up in energy, at small temperatures we will not go uphill.

OUTLINE

- Directed graphical models $\surd$
- Undirected graphical models $\surd$
- State estimation $\surd$
- MCMC $\surd$
- Gaussians
- Mixture models

1D

$$\mathcal{N}(x|\mu,\sigma) \stackrel{\text{def}}{=} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} \quad (35)$$

MLE

$$\mu_{ML} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad (36)$$

$$\hat{\sigma^2}_{ML} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_{ML})^2 \quad (37)$$

$$= \frac{1}{N}\sum_{i}(x_i^2) - (\frac{1}{N}\sum_{i} x_i)^2 \quad (38)$$

MULTID

Read Alpaydin sec 5.1-5.4

$$\mathcal{N}(\vec{x}|\vec{\mu},\Sigma) \stackrel{\text{def}}{=} \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp[-\tfrac{1}{2}(\vec{x}-\vec{\mu})^T\Sigma^{-1}(\vec{x}-\vec{\mu})] \quad (39)$$

Σ (and hence Σ^{-1}) is a symmetric positive definite matrix i.e. for all vectors u

$$\Delta = u^T\Sigma^{-1}u \geq 0$$

If $u = x - \mu$, this is the Mahalanobis distance between x and μ.

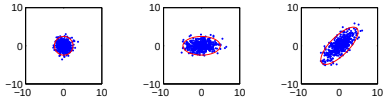
2D GAUSSIANS

$$\Sigma = \begin{pmatrix} \sigma_X & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y \end{pmatrix} \quad (40)$$

$$\rho = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}} \quad (41)$$

SPHERICAL, DIAGONAL, FULL COVARIANCE MATRICES

$1, d, O(d^2)$ parameters



MLE

$$\mu_{ML} = \frac{1}{N}\sum_i \vec{x}_i \tag{42}$$

$$\Sigma_{ML} = \frac{1}{N}\sum_{i=1}^{N}(\vec{x}_i - \mu_{ML})(\vec{x}_i - \mu_{ML})^T \tag{43}$$

N, $\sum_i \vec{x}_i$ and $\sum_i \vec{x}_i \vec{x}_i^T$ are called **(minimal) sufficient statistics**.

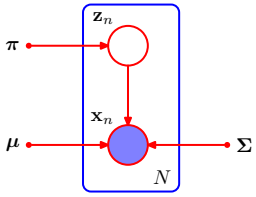
OUTLINE

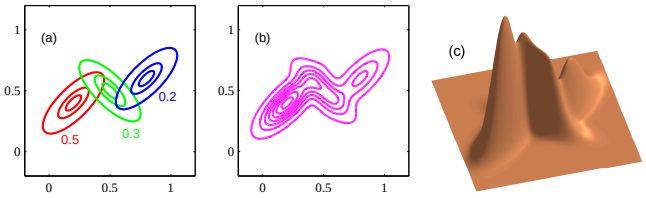
- Directed graphical models $\surd$
- Undirected graphical models $\surd$
- State estimation $\surd$
- MCMC $\surd$
- Gaussians $\surd$
- Mixture models

GAUSSIAN MIXTURE MODELS

Read Alpaydin ch 7!

$$p(x|\theta) = \sum_{k=1}^{K} p(z=k)p(x|z=k) = \sum_k \pi(k)\mathcal{N}(x|\mu_k, \Sigma_k) \tag{44}$$





MIXTURE OF BERNOULLIS

Just change the "class conditional density" $p(x|z=k)$

$$p(x|z=k,\theta) = \prod_{i=1}^{K} Be(x_i|\theta_{ki}) = \prod_{i=1}^{K} x_i^{\theta_{ki}}(1-x_i)^{1-\theta_{ki}} \tag{45}$$

Useful for clustering binary data

Log likelihood

$$\ell(\theta) = \sum_n \log \sum_{z_n=1}^{K} \pi(z_n)\mathcal{N}(x_n|z_n, \mu(z_n), \Sigma(z_n)) \quad (46)$$

Can use Newton's method or conjugate gradient descent, etc. Or can use EM. Both will get stuck in local maxima.

- E step:

$$p(z_n = k|x_n, \theta^{old}) = r_{nk} = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)} \quad (47)$$

The value r_{nk} is called the **responsibility** of cluster k for data point n.

- M step:

$$\theta^{new} = \arg\max_\theta Q(\theta, \theta^{old}) = \arg\max_\theta E \sum_n \log p(x_n, z_n|\theta) \quad (48)$$

$$\pi_k = \frac{1}{N}\sum_n r_{nk} \quad (49)$$

$$\mu_k^{new} = \frac{\sum_n r_{nk} x_n}{\sum_n r_{nk}} \quad (50)$$

$$\Sigma_k = \frac{\sum_n r_{nk}(x_n - \mu_k^{new})(x_n - \mu_k^{new})^T}{\sum_n r_{nk}} \quad (51)$$

- We assume $\Sigma_k = I$ and $\pi_k = 1/K$ are fixed and just learn the centers μ_k (prototypes).
- E step: hard assign each point to closest prototype

$$z_n^* = \arg\max_k p(z_n = k|x_n, \theta^{old}) \quad (52)$$

$$= \arg\max_k \exp(-\tfrac{1}{2}(x_n - \mu_k)^2) \quad (53)$$

$$= \arg\min_k ||x_n - \mu_k||^2 \quad (54)$$

- M step: just take average of all the points assigned to you

$$\mu_k^{new} = \frac{\sum_{n:z_n^*=k} x_n}{\sum_n z_n^* = k} \quad (55)$$

- Cross validation
- MDL: pick K to minimize cost, which is number of bits required to encode model and data given model. By Shannon, we have

$$cost(K) \approx -\log p(D|, \hat{\theta}, K) - \log p(\hat{\theta}|K) \quad (56)$$

This is the number of bits required to specify the parameters $\hat{\theta}$, and the number of bits required to specify the residual errors.

GMM is a flat clustering. We can do hierarchical clustering by greedily merging clusters that are most similar.
Single link clustering:

$$D_{SL}(G_i, G_j) = \min_{x^r \in G_i, x^s \in G_j} D(x^r, x^s) \tag{57}$$

where $D(x^r, x^s)$ is a distance measure between two feature vectors. Same as building a minimum spanning tree of the data. Order of merges produces a dendogram.