Case Study 4: Collaborative Filtering

GraphLab Review,

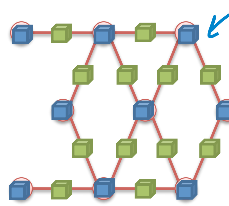
ALS, SGD, Gibbs Sampling for MF in GraphLab

Machine Learning for Big Data
CSE547/STAT548, University of Washington
Emily Fox
February 27th, 2014

The GraphLab Framework

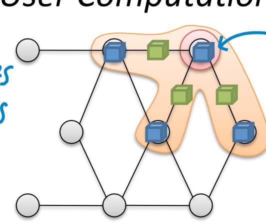
provided by user

1. Graph Based
Data Representation



"data" on nodes/vertices
and on edges
"data": obs, current est. of params, features, ⋮

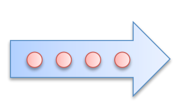
2. Update Functions
User Computation



update of this vertex given its "scope"
set of nbr vertices + edges

choice from a list

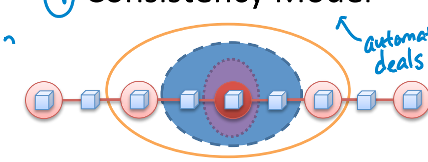
3. Scheduler

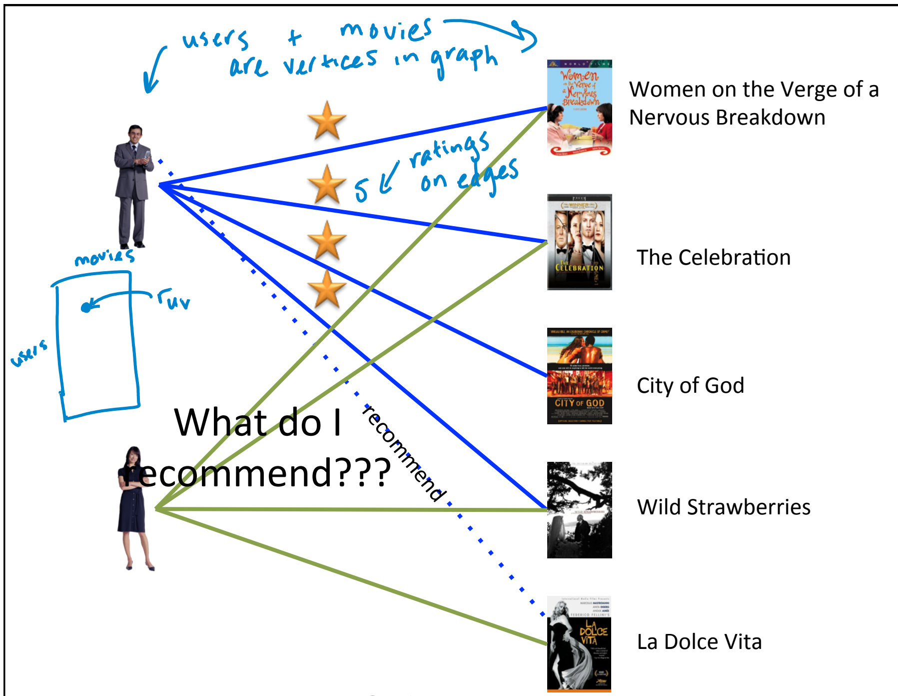


- round robin
- FIFO
⋮

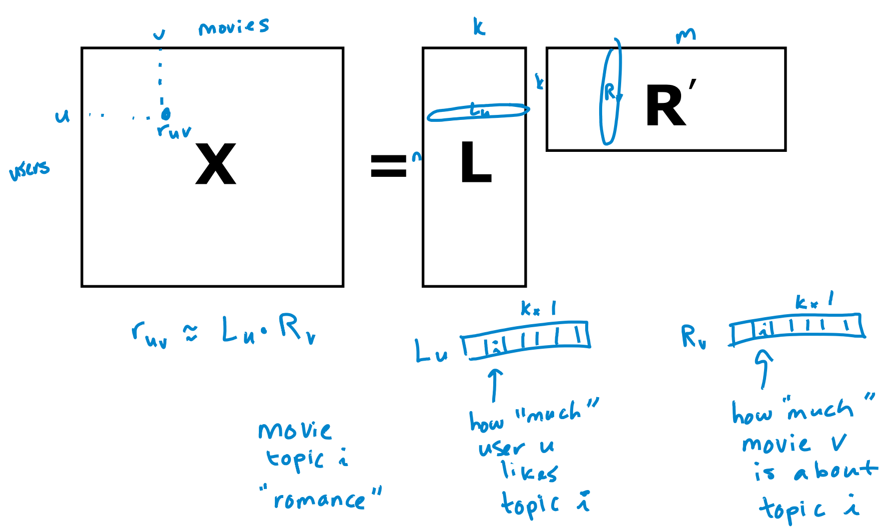
4. Consistency Model

automatically deals w/ race conds.



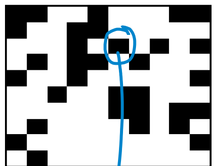


Interpreting Low-Rank Matrix Completion (aka Matrix Factorization)



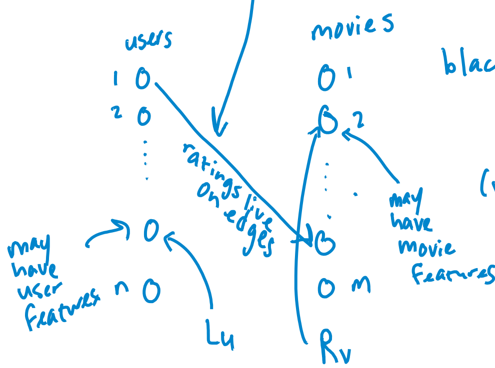
Matrix Completion as a Graph

X =



X_{ij} known for black cells
X_{ij} unknown for white cells

Rows index users
Columns index movies



black cells in data matrix = edges in graph

(value in cell = weight on edge)

Coordinate Descent for Matrix Factorization: Alternating Least-Squares

$$\min_{L,R} \sum_{(u,v):r_{uv}\neq ?} (L_u \cdot R_v - r_{uv})^2 + \lambda_u \|L\| + \lambda_v \|R\|$$

model w/o user/movie features

- Fix movie factors, optimize for user factors $\min_{L_u} \sum_{v \in V_u} (L_u \cdot R_v - r_{uv})^2 + \lambda_u \|L\|$
  - ☆ Independent least-squares over users

  nhbrs of user u (movies rated by user u)

- Fix user factors, optimize for movie factors $\min_{R_v} \sum_{u \in U_v} (L_u \cdot R_v - r_{uv})^2 + \lambda_v \|R\|$
  - ☆ Independent least-squares over movies

  nhbrs of movie v

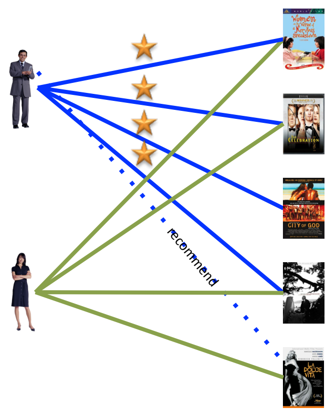
- System may be underdetermined: use regularization

- Converges to local optima

Alternating Least Squares Update Function

$$\min_{L_u} \sum_{v \in V_u} (L_u \cdot R_v - r_{uv})^2 + \lambda_u \|L\| \qquad \min_{R_v} \sum_{u \in U_v} (L_u \cdot R_v - r_{uv})^2 + \lambda_v \|R\|$$

e.g. user i



update(i, scope) {

read current factors for nbrs

build e.g. $X = \begin{bmatrix} \text{---} \\ - R_v - \\ \text{---} \\ \vdots \end{bmatrix}$ } all movies rated by user i

read all ratings on edges

build $y = \begin{bmatrix} \vdots \\ r_{iv} \\ \vdots \end{bmatrix} \leftarrow v \in V_i$

Solve local regression problem

e.g. for L_2 reg,

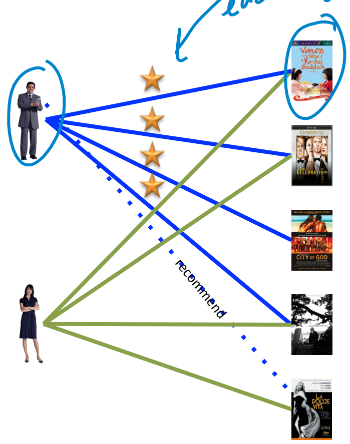
$L_u = (X^T X + \lambda_u I)^{-1} X^T y$

}

SGD for Matrix Factorization in GraphLab

$$\epsilon_t = L_u^{(t)} \cdot R_v^{(t)} - r_{uv} \qquad \begin{bmatrix} L_u^{(t+1)} \\ R_v^{(t+1)} \end{bmatrix} \leftarrow \begin{bmatrix} (1 - \eta_t \lambda_u) L_u^{(t)} - \eta_t \epsilon_t R_v^{(t)} \\ (1 - \eta_t \lambda_v) R_v^{(t)} - \eta_t \epsilon_t L_u^{(t)} \end{bmatrix}$$

update for each edge



GraphLab operates on vertices

Update(i, scope){

Perform SGD update

for each neighbor of i

(for every edge connected to i)

}

Bayesian PMF Example

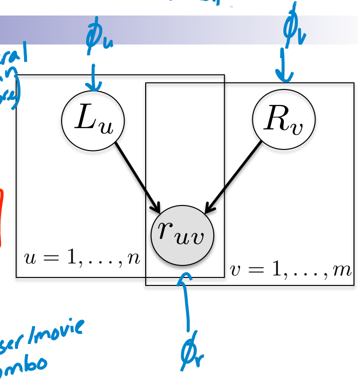
Full Bayesian approach: place priors on ϕ as well!

- Latent user and movie factors: (more general than before)

$$\boxed{\begin{aligned} L_u &\sim N(\mu_u, \Sigma_u) \\ R_v &\sim N(\mu_v, \Sigma_v) \end{aligned}} \quad \begin{aligned} u &= 1, \dots, n \\ v &= 1, \dots, m \end{aligned}$$

- Observations $\boxed{r_{uv} \sim N(L_u' R_v, \sigma_r^2)}$
- Hyperparameters:

$$\phi = \{\underbrace{\mu_u, \Sigma_u}_{\phi_u}, \underbrace{\mu_v, \Sigma_v}_{\phi_v}, \underbrace{\sigma_r^2}_{\phi_r}\}$$



- Want to predict new movie rating:

new user/movie combo

posterior given obs. so far

$$p(r_{uv}^* \mid X, \phi) = \int p(r_{uv}^* \mid L_u, R_v)\, p(L, R \mid X, \phi)\, dL\, dR$$

r_{uv}^*: new rating; X: obs. ratings

Bayesian PMF Gibbs Sampler

- Outline of Bayesian PMF sampler

1. Init $L^{(1)}, R^{(1)}$
2. For $k = 1, \dots, N_{iter}$
   (i) Sample hyperparams $\phi^{(k)} = \{\phi_u^{(k)}, \phi_v^{(k)}, \phi_r^{(k)}\}$
   (ii) For each user $u = 1, \dots, n$ sample in parallel
   $$\boxed{L_u^{(k+1)} \sim p(L_u \mid X, R^{(k)}, \phi^{(k)})}$$ form on next slide
   (iii) For each movie $v = 1, \dots, m$ sample in parallel
   $$\boxed{R_v^{(k+1)} \sim p(R_v \mid X, L^{(k+1)}, \phi^{(k)})}$$

Very similar to ideas of ALS (systematically)

Bayesian PMF Example

- For user u:

prior   likelihood for user u

$$p(L_u \mid X, R, \phi_u) \propto p(L_u \mid \phi_u) \prod_{v \in V_u} p(r_{uv} \mid L_u, R_v, \phi_r)$$

$$\propto N(L_u \mid \mu_u, \Sigma_u) \prod_{v \in V_u} N(r_{uv} \mid L_u \cdot R_v, \sigma_r^2)$$

$$= \boxed{N(L_u \mid \tilde{\mu}_u, \tilde{\Sigma}_u)}$$

via conjugacy
posterior is in the same family as prior

where $\tilde{\Sigma}_u^{-1} = \Sigma_u^{-1} + \sigma_r^{-2} \sum_{v \in V_u} R_v R_v^T$ sum over all nbhrs of user u

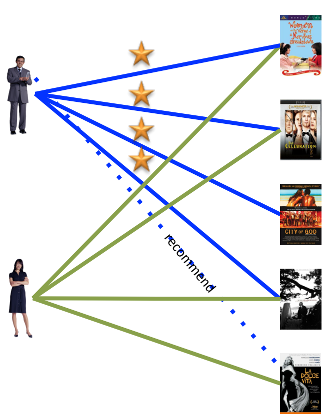
$\tilde{\mu}_u = \tilde{\Sigma}_u \left(\sigma_r^{-2} \sum_{v \in V_u} r_{uv} R_v + \Sigma_u^{-1} \mu_u \right)$ vertex weight } in scope of user u

edge weights

- Symmetrically for R_v conditioned on L (breaks down over movies)
- Luckily, we can use this to get our desired posterior samples

PMF Gibbs Sampling in GraphLab

$p(L_u \mid X, R, \phi_u) = N(\tilde{\mu}_u, \tilde{\Sigma}_u)$ $\tilde{\Sigma}_u^{-1} = \Sigma_u^{-1} + \sigma_r^{-2} \sum_{v \in V_u} R_v R_v^T$ $\tilde{\mu}_u = \tilde{\Sigma}_u \left(\sigma_r^{-2} \sum_{v \in V_u} r_{uv} R_v + \Sigma_u^{-1} \mu_u \right)$



update(i, scope){   e.g. user i (update to L_i)

read current factors for nbhrs R_j

read ratings on edges r_{ij}

Set $\tilde{\Sigma}_i^{-1} = \Sigma_i^{-1} + \sigma_r^{-2} \sum_{j \in nbhrs(i)} R_j R_j^T$

fixed at vertex i

Set $\tilde{\mu}_i = \tilde{\Sigma}_i \left(\sigma_r^{-2} \sum_{j \in nbhrs(i)} r_{ij} R_j + \Sigma_i^{-1} \mu_i \right)$

Sample $L_i \sim N(\tilde{\mu}_i, \tilde{\Sigma}_i)$

}

GraphLab

Release 2.2 available now

http://graphlab.org

Documentation… Code… Tutorials… (more on the way)

GraphChi 0.1 available now

http://graphchi.org

What you need to know…

- Data-parallel versus graph-parallel computation
- Bulk synchronous processing versus asynchronous processing
- GraphLab system for graph-parallel computation
  - Data representation
  - Update functions
  - Scheduling
  - Consistency model

collaborative filtering

- ALS, SGD and Gibbs for matrix factorization/PMF in GraphLab

Reading

- Papers under "Case Study IV: **Parallel Learning with GraphLab**"

- Optional:
  - Parallel Splash BP
    http://www.ml.cmu.edu/research/dap-papers/dap-gonzalez.pdf

Acknowledgements

- Slides based on Carlos Guestrin's GraphLab talk

Case Study 5: Mixed Membership Modeling

Clustering Documents Revisited, Latent Dirichlet Allocation

Machine Learning for Big Data
CSE547/STAT548, University of Washington
Emily Fox
February 27th, 2014

Document Retrieval

- **Goal:** Retrieve documents of interest
- **Challenges:**
  - Tons of articles out there
  - How should we measure similarity?

Task 1: Find Similar Documents

- **First considered:**
  - **Input:** Query article x
  - **Output:** Set of k similar articles $x^{NN_1}, \ldots, x^{NN_k}$

x

Task 2: Cluster Documents

- **Then examined:**
  - Cluster documents based on topic

sports

world news

Document Representation

- Bag of words model

previously: $x^d = [\ \ ]$ ← vector fcn of word counts (e.g. tf-idf)

in NN search or some clustering alg.

performed operations on this vector

document *d*

cat ← word 1	
dog	2
hat	3
car	⋮
⋮	

$w_i^d = 3$ ⇒ ith word in doc d is 'hat'

representation of doc d

now: $x^d = \{w_1^d, \dots, w_{N_d}^d\}$ ← "bag of words"

unordered set of N_d word indices with $w_i^d \in V$ vocab.

A Generative Model

- Documents: $x^1, \dots, x^D$ with $x^d = \{w_1^d, \dots, w_{N_d}^d\}$
- Associated topics: $z^1, \dots, z^D$ with $z^d \in \{1, \dots, K\}$ ← # topics
- Parameters: $\theta = \{\pi, \beta\}$

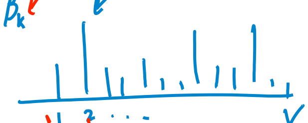
topic indicator for doc 1

as before $\begin{cases} \pi = [\pi_1, \dots, \pi_K] \\ Pr(z^d = k) = \pi_k \end{cases}$ topic probabilities ← $pr(z^d = k \mid \pi) = \pi_k$

$$\beta = \begin{array}{c|c} & 1 \quad 2 \quad \cdots \quad V \\ \hline 1 & \begin{bmatrix} \text{—} \ \beta_1 \ \text{—} \\ \vdots \\ \text{—} \ \beta_K \ \text{—} \end{bmatrix} \\ 2 & \\ \vdots & \\ K & \end{array}$$

β_k ← in topic k

prob. of word 2



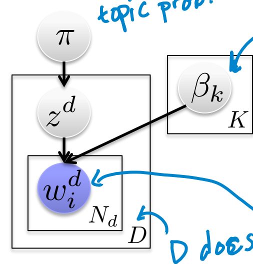
word 1, word 2, . . . , V

words

size of vocab

A Generative Model

- Documents: $x^1, \ldots, x^D$
- Associated topics: $z^1, \ldots, z^D$
- Parameters: $\theta = \{\pi, \beta\}$
- Generative model:

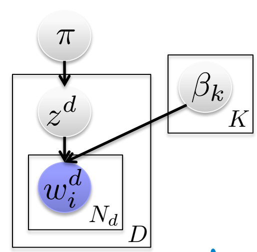


$z^d \sim \pi$ generate topic

$w_i^d \mid z^d \sim \beta_{z^d}$ $i=1,\ldots,N_d$

Given topic $z^d = k$ for doc d, draw each word from β_k ← word prob. for topic k

Model In Pictures



- Mixture weights (on topics)

π

1 2 . . . K topics

health computing . . . world news

$z^d = 2$

$z^d = K$

- Topic distributions (on words)

health β_1 apple run doctor laptop V

⋮

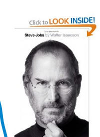
world news β_K drama Israel . . V

β_2 computer distributed . . .

w_1^d w_2^d w_3^d w_4^d

β_K drama Israel

w w w w



- For each document,

$$z^d \sim \pi$$

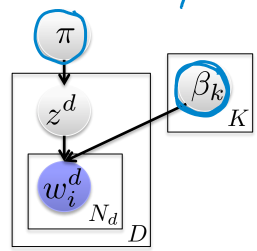
$$w_i^d \mid z^d \sim \beta_{z^d}$$

$z^d = 5$

sports

Bayesian Document Model

- Model parameters $\pi, \{\beta_k\}$ unknown ← can use EM as in case study 2



- Bayesian approach

  place priors on parameters

- Need distribution on pmf's

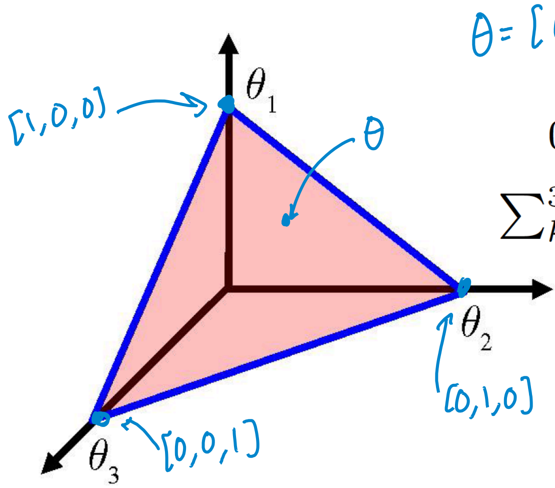
$\sum_{k=1}^{K} \pi_k = 1$ $\quad$ $\sum_{v=1}^{V} \beta_{kv} = 1$ $\quad$ ← π, β_k live on the simplex

(1) What is the simplex?
(2) What is a distribution on the simplex?

The Simplex in 3D

- The simplex defines the hyperplane of vectors that sum to 1

$\theta = [\theta_1, \dots, \theta_K]$ ← e.g. $K=3$ here



$$0 \leq \theta_k \leq 1$$

$$\sum_{k=1}^{3} \theta_k = 1$$

Dirichlet Distributions

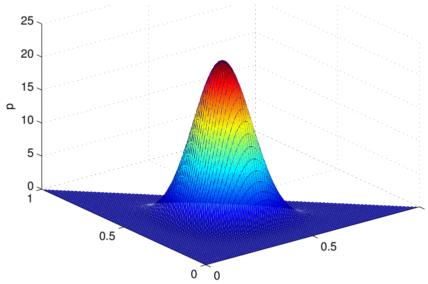
- The Dirichlet distribution is defined on the simplex

$\alpha_k = 10 \ \forall k$

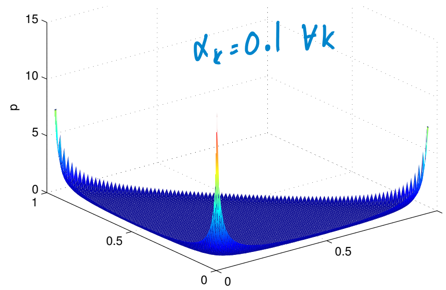
$\pi \sim \mathrm{Dir}(\alpha_1, \dots, \alpha_K)$

$\Rightarrow \sum \pi_k = 1$ and $\pi_k \geq 0 \ \forall k$

$$p(\pi \mid \alpha) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k=1}^{K} \pi_k^{\alpha_k - 1}$$



$\alpha_k = 0.1 \ \forall k$



Moments: $\mathbb{E}_\alpha[\pi_k] = \frac{\alpha_k}{\alpha_0}$

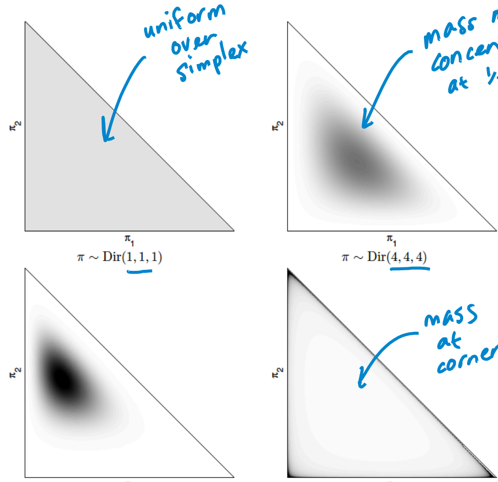
$$\mathrm{Var}_\alpha[\pi_k] = \frac{K-1}{K^2(\alpha_0 + 1)}$$

Dirichlet Probability Densities

uniform over simplex

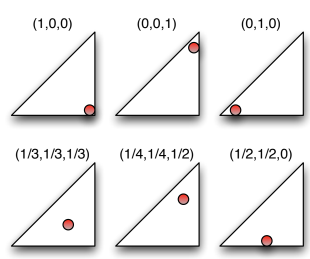
mass more concentrated at 1/3, 1/3, 1/3

mass at corners



$\pi \sim \mathrm{Dir}(1,1,1)$ $\pi \sim \mathrm{Dir}(4,4,4)$ $\pi \sim \mathrm{Dir}(4,9,7)$ $\pi \sim \mathrm{Dir}(0.2,0.2,0.2)$

$\pi \sim \mathrm{Dir}(\alpha_1, \dots, \alpha_K)$



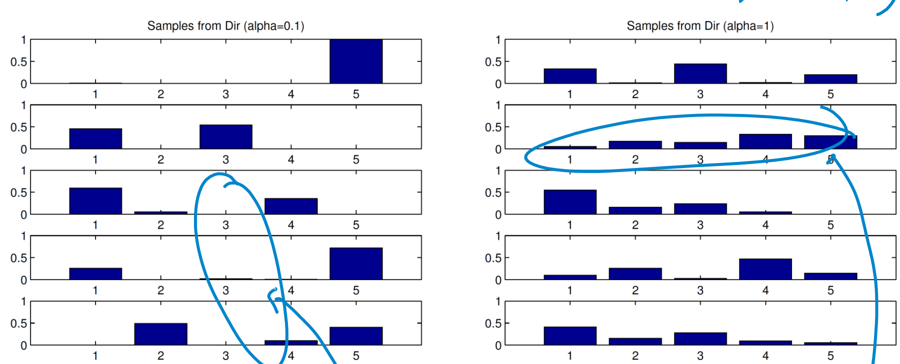
draws from Dir

Dirichlet Samples

$$\mathbb{E}_\alpha[\pi_k] = \frac{\alpha_k}{\alpha_0}$$

- Samples are ***sparse*** for small values of α_i

(5D example)



$\mathrm{Dir}(\pi \mid 0.1, 0.1, 0.1, 0.1, 0.1)$ $\mathrm{Dir}(\pi \mid 1.0, 1.0, 1.0, 1.0, 1.0)$

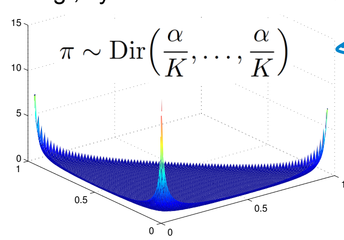
puts mass at corners

many small weights

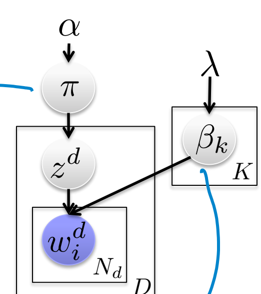
much more uniform

Model Summary

- Prior on model parameters
  - E.g., symmetric Dirichlet for π



$$\pi \sim \mathrm{Dir}\left(\frac{\alpha}{K}, \dots, \frac{\alpha}{K}\right)$$



  - Dirichlet prior for topic parameters $\beta_k \sim \mathrm{Dir}(\lambda_1, \dots, \lambda_V)$ $k = 1, \dots, K$
- Sample observations as

$$z^d \sim \pi \quad d = 1, \dots, D$$
$$w_i^d \mid z^d \sim \beta_{z^d} \quad i = 1, \dots, N_d$$

Posterior Inference via Sampling

- Iterate between sampling ← actual obs.

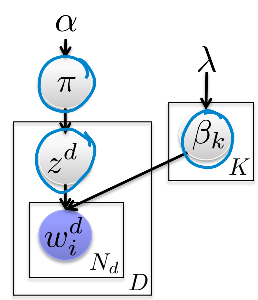
$\pi \sim p(\pi \mid \{z^d\}, \{\beta_k\}, \{w_i^d\})$

For $k=1,\dots,K$

$\beta_k \sim p(\beta_k \mid \pi, \{z^d\}, \{\beta_j, j\neq k\}, \{w_i^d\})$

For $d=1,\dots,D$

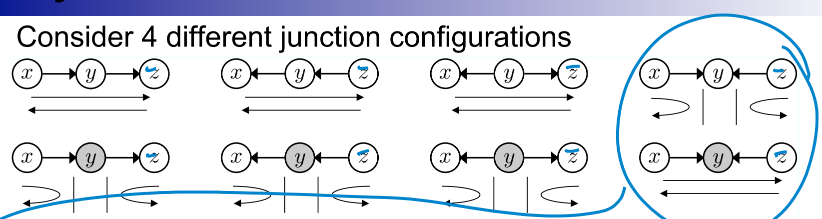
$z^d \sim p(z^d \mid \pi, \{z^i, i\neq d\}, \{\beta_k\}, \{w_i^d\})$



- What form do these complete conditionals take?
  - First a look at statements of conditional independence in ~~directed~~ graphical models

Conditional Independence in Bayes Nets

- Consider 4 different junction configurations



(a) (b) (c) (d)

- Conditional versus unconditional independence:

$p(x,y,z) = p(x)p(z)p(y|x,z) \xRightarrow{\text{int. over } y} p(x,z) = p(x)p(z) \Rightarrow x \perp\!\!\!\perp z$

$p(x,z|y) \propto p(x,y,z) = p(x)p(z)p(y|x,z)$

$\neq p(x|y)\,p(z|y)$ ← $x \not\perp\!\!\!\perp z \mid y$

"explaining away": x= earthquake z = burglar , y= car alarm

ind. apriori

If alarm (y=1) , an increase in earthquake p(x|y) , means p(z|y) lower