Foundations of Bayesian NLP

MSc Artificial Intelligence

Lecturer: Wilker Aziz
Institute for Logic, Language, and Computation

2018

The problem with MLE

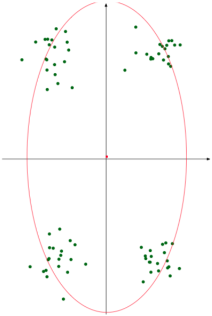
Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM

The problem with MLE

Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM



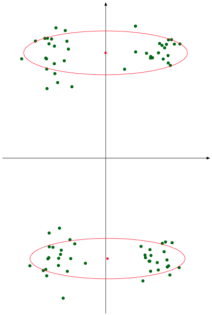
K	training log-likelihood	test log-likelihood
1	-364	

Example from Liang and Klein (2007): ACL tutorial on Structured Bayesian Nonparametric Models

The problem with MLE

Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM



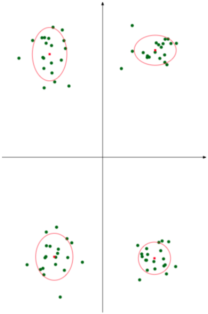
K	training log-likelihood	test log-likelihood
1	-364	
2	-204	

Example from Liang and Klein (2007): ACL tutorial on Structured Bayesian Nonparametric Models

The problem with MLE

Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM



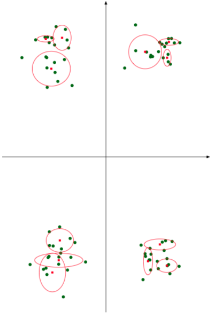
K	training log-likelihood	test log-likelihood
1	-364	
2	-204	
4	-82	

Example from Liang and Klein (2007): ACL tutorial on Structured Bayesian Nonparametric Models

The problem with MLE

Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM



K	training log-likelihood	test log-likelihood
1	-364	
2	-204	
4	-82	
12	21	

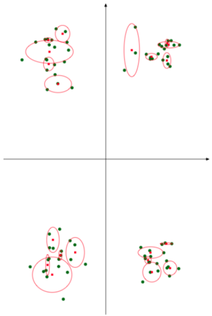
- as the capacity of the model increases (more clusters), training likelihood strictly improves

Example from Liang and Klein (2007): ACL tutorial on Structured Bayesian Nonparametric Models

The problem with MLE

Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM



K	training log-likelihood	test log-likelihood
1	-364	
2	-204	
4	-82	
12	21	
20	86	

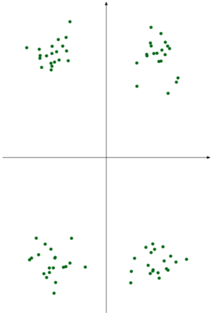
- as the capacity of the model increases (more clusters), training likelihood strictly improves
- but what happens with test likelihood?

Example from Liang and Klein (2007): ACL tutorial on Structured Bayesian Nonparametric Models

The problem with MLE

Motivating example from Liang and Klein (2007)

- mixture of Gaussians trained via EM



K	training log-likelihood	test log-likelihood
1	-364	-368
2	-204	-206
4	-82	→ -128
12	21	-147
20	86	-173

- as the capacity of the model increases (more clusters), training likelihood strictly improves
- but what happens with test likelihood?

Example from Liang and Klein (2007): ACL tutorial on Structured Bayesian Nonparametric Models

The problem with MLE

That's why you were told to always do model selection

- on heldout set
- preferably via cross-validation

The problem with MLE

That's why you were told to always do model selection

- on heldout set
- preferably via cross-validation

Can you see limitations of this approach?

The problem with MLE

That's why you were told to always do model selection

- on heldout set
- preferably via cross-validation

Can you see limitations of this approach?

- availability of data
- representativeness of heldout set
- discrete optimisation: combinatorial search over models

NLP1

Preliminaries

Bayesian modelling

Applications

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$
- all but the ith observation $\mathbf{x}_{-i}$

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$
- all but the ith observation $\mathbf{x}_{-i}$
- N cluster indicators
  $\mathbf{z} = \langle z_1, \ldots, z_N \rangle$

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$
- all but the ith observation $\mathbf{x}_{-i}$
- N cluster indicators
  $\mathbf{z} = \langle z_1, \ldots, z_N \rangle$
- ith cluster indicator $z_i \in \{1, \ldots, C\}$

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$
- all but the ith observation $\mathbf{x}_{-i}$
- N cluster indicators
  $\mathbf{z} = \langle z_1, \ldots, z_N \rangle$
- ith cluster indicator $z_i \in \{1, \ldots, C\}$
- all but the ith cluster assignment $\mathbf{z}_{-i}$

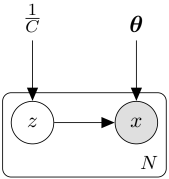
Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$
- all but the ith observation $\mathbf{x}_{-i}$
- N cluster indicators
  $\mathbf{z} = \langle z_1, \ldots, z_N \rangle$
- ith cluster indicator $z_i \in \{1, \ldots, C\}$
- all but the ith cluster assignment $\mathbf{z}_{-i}$
- Parameter vector
  $\theta = \langle \theta_1, \ldots, \theta_K \rangle$

Conventions

- N observations
  $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$
- ith observation $x_i \in \{1, \ldots, K\}$
- all but the ith observation $\mathbf{x}_{-i}$
- N cluster indicators
  $\mathbf{z} = \langle z_1, \ldots, z_N \rangle$
- ith cluster indicator $z_i \in \{1, \ldots, C\}$
- all but the ith cluster assignment $\mathbf{z}_{-i}$
- Parameter vector
  $\theta = \langle \theta_1, \ldots, \theta_K \rangle$
- Collection of parameter vectors
  $\boldsymbol{\theta} = \langle \theta^{(1)}, \ldots, \theta^{(C)} \rangle$

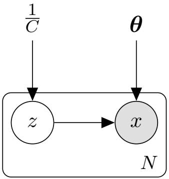
Mixture model



Let's assume x to be 1 of K, and z to be 1 of C

- categorical likelihood

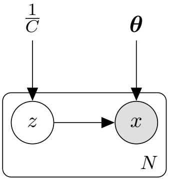
Mixture model



Let's assume x to be 1 of K, and z to be 1 of C

- categorical likelihood
- uniform prior over mixture components, i.e. mixing weights are fixed and uniform

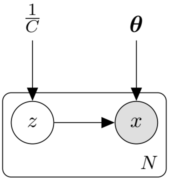
Mixture model



Let's assume x to be 1 of K, and z to be 1 of C

- categorical likelihood
- uniform prior over mixture components, i.e. mixing weights are fixed and uniform
- $\theta^{(c)} \in \Delta_{K-1}$

Mixture model



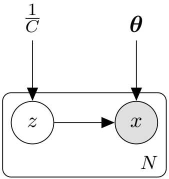
For $i = 1, \ldots, N$

Let's assume x to be 1 of K, and z to be 1 of C

- categorical likelihood
- uniform prior over mixture components, i.e. mixing weights are fixed and uniform
- $\theta^{(c)} \in \Delta_{K-1}$

$$Z_i \sim \mathcal{U}(C)$$

Mixture model



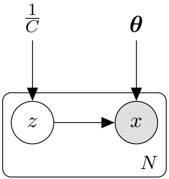
For $i = 1, \ldots, N$

Let's assume x to be 1 of K, and z to be 1 of C

- categorical likelihood
- uniform prior over mixture components, i.e. mixing weights are fixed and uniform
- $\theta^{(c)} \in \Delta_{K-1}$

$$\begin{aligned} Z_i &\sim \mathcal{U}(C) \\ X_i|\boldsymbol{\theta}, \mathbf{z}_{-i}, z_i = c &\sim \mathrm{Cat}(\theta^{(c)}) \end{aligned} \tag{1}$$

Mixture model



Let's assume x to be 1 of K, and z to be 1 of C

- categorical likelihood
- uniform prior over mixture components, i.e. mixing weights are fixed and uniform
- $\theta^{(c)} \in \Delta_{K-1}$

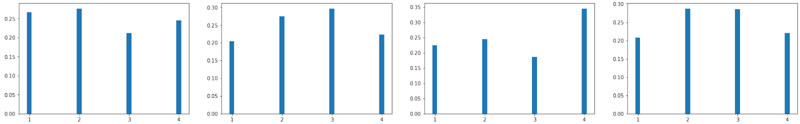
For $i = 1, \ldots, N$

$$\begin{aligned} Z_i &\sim \mathcal{U}(C) \\ X_i|\boldsymbol{\theta}, \mathbf{z}_{-i}, z_i = c &\sim \mathrm{Cat}(\theta^{(c)}) \end{aligned} \tag{1}$$

What is a sensible conditional distribution $X|\theta^{(c)} \sim \mathrm{Cat}(\theta^{(c)})$?

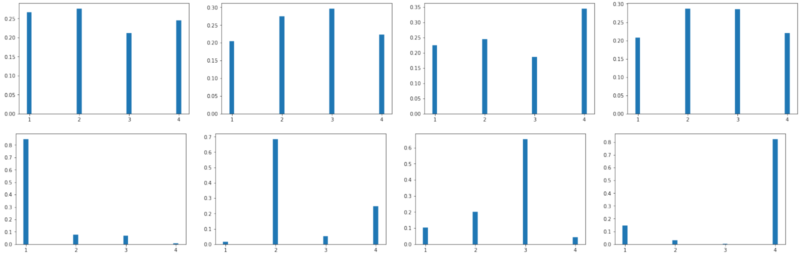
What makes a good conditional?

$c = 1$ (the blue cluster), $K = 4$



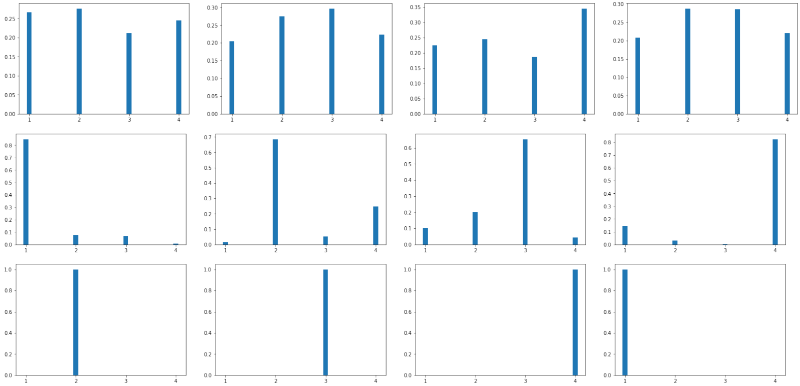
What makes a good conditional?

$c = 1$ (the blue cluster), $K = 4$



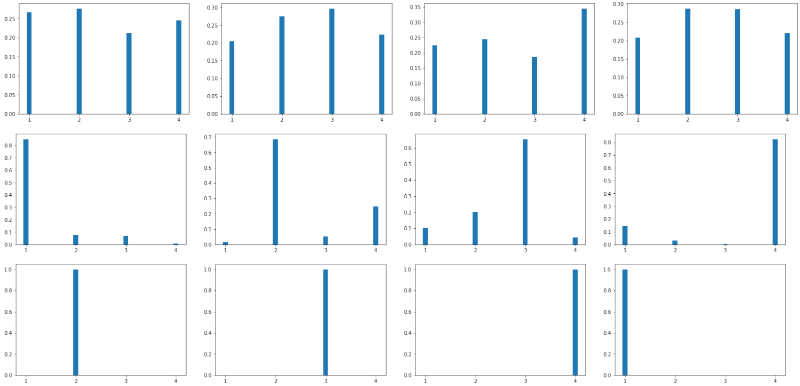
What makes a good conditional?

$c = 1$ (the blue cluster), $K = 4$



What makes a good conditional?

$c = 1$ (the blue cluster), $K = 4$



Can you make any assumptions before observing data?

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} =$$

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} = \frac{\overbrace{P(d|h)}^{\text{likelihood}} \overbrace{P(h)}^{\text{prior}}}{\underbrace{P(d)}_{\text{evidence}}}$$

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} = \frac{\overbrace{P(d|h)}^{\text{likelihood}} \overbrace{P(h)}^{\text{prior}}}{\underbrace{P(d)}_{\text{evidence}}} \propto P(d|h)P(h) \quad (2)$$

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} = \frac{\overbrace{P(d|h)}^{\text{likelihood}} \overbrace{P(h)}^{\text{prior}}}{\underbrace{P(d)}_{\text{evidence}}} \propto P(d|h)P(h) \tag{2}$$

- the likelihood tells you how well a hypothesis h explains the observed data d;

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} = \frac{\overbrace{P(d|h)}^{\text{likelihood}} \overbrace{P(h)}^{\text{prior}}}{\underbrace{P(d)}_{\text{evidence}}} \propto P(d|h)P(h) \tag{2}$$

- the likelihood tells you how well a hypothesis h explains the observed data d;
- the prior tells you how much h conforms to expectations about what a good hypothesis looks like **regardless** of observed data;

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} = \frac{\overbrace{P(d|h)}^{\text{likelihood}}\overbrace{P(h)}^{\text{prior}}}{\underbrace{P(d)}_{\text{evidence}}} \propto P(d|h)P(h) \tag{2}$$

- the likelihood tells you how well a hypothesis h explains the observed data d;
- the prior tells you how much h conforms to expectations about what a good hypothesis looks like **regardless** of observed data;
- the evidence tells you how well your model $\mathcal{M}$ explains the data, i.e. $P(d)$ is actually $P(d|\mathcal{M})$

Bayes rule

What does Bayes rule tell you?

$$\underbrace{P(h|d)}_{\text{posterior}} = \frac{\overbrace{P(d|h)}^{\text{likelihood}}\ \overbrace{P(h)}^{\text{prior}}}{\underbrace{P(d)}_{\text{evidence}}} \propto P(d|h)P(h) \tag{2}$$

- the likelihood tells you how well a hypothesis h explains the observed data d;
- the prior tells you how much h conforms to expectations about what a good hypothesis looks like **regardless** of observed data;
- the evidence tells you how well your model $\mathcal{M}$ explains the data, i.e. $P(d)$ is actually $P(d|\mathcal{M})$
- the posterior updates our beliefs about hypotheses **in light of** observed data.

Maximum likelihood estimation

An optimisation problem based on the *(log-)likelihood function*

$$h^\star = \arg\max_h \; P(d|h)$$

Maximum likelihood estimation

An optimisation problem based on the *(log-)likelihood function*

$$h^\star = \arg\max_h \; P(d|h) = \arg\max_h \; \underbrace{\log P(d|h)}_{\mathcal{L}(h)} \tag{3}$$

Maximum likelihood estimation

An optimisation problem based on the *(log-)likelihood function*

$$h^\star = \arg\max_h \; P(d|h) = \arg\max_h \; \underbrace{\log P(d|h)}_{\mathcal{L}(h)} \tag{3}$$

- all hypotheses are **equally likely a priori**;

Maximum likelihood estimation

An optimisation problem based on the *(log-)likelihood function*

$$h^\star = \arg\max_h \; P(d|h) = \arg\max_h \; \underbrace{\log P(d|h)}_{\mathcal{L}(h)} \tag{3}$$

- all hypotheses are **equally likely a priori**;
- can be approached by coordinate ascent methods;

Maximum likelihood estimation

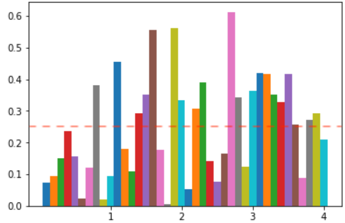
An optimisation problem based on the *(log-)likelihood function*

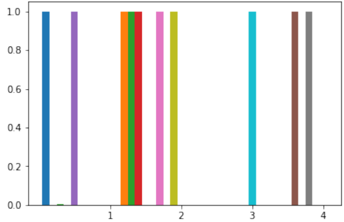
$$h^\star = \arg\max_h \ P(d|h) = \arg\max_h \ \underbrace{\log P(d|h)}_{\mathcal{L}(h)} \tag{3}$$

- all hypotheses are **equally likely a priori**;
- can be approached by coordinate ascent methods;
- local optimality guarantees;

All the same a priori

Before data, MLE is equally happy with the hypotheses on the left





Constraining MLE

Maximum a posteriori

$$\begin{aligned} h^\star &= \arg\max_h \; P(d|h)P(h) \\ &= \arg\max_h \; \log P(d|h) + \log P(h) \end{aligned} \tag{4}$$

- perhaps fine if $P(h)$ has a single narrow peak

Constraining MLE

Maximum a posteriori

$$
\begin{aligned}
h^\star &= \arg\max_h \; P(d|h)P(h) \\
&= \arg\max_h \; \log P(d|h) + \log P(h)
\end{aligned} \tag{4}
$$

- perhaps fine if $P(h)$ has a single narrow peak
- priors often indicate preference for a subset of hypotheses over another, multiple peaks make optimisation considerably harder

Constraining MLE

Maximum a posteriori

$$h^\star = \arg\max_h \; P(d|h)P(h) \\ = \arg\max_h \; \log P(d|h) + \log P(h) \tag{4}$$

- perhaps fine if $P(h)$ has a single narrow peak
- priors often indicate preference for a subset of hypotheses over another, multiple peaks make optimisation considerably harder
- still a point estimate, teaches us very little about the overall model (set of assumptions)

Constraining MLE

Maximum a posteriori

$$\begin{aligned} h^\star &= \arg\max_h \; P(d|h)P(h) \\ &= \arg\max_h \; \log P(d|h) + \log P(h) \end{aligned} \tag{4}$$

- perhaps fine if $P(h)$ has a single narrow peak
- priors often indicate preference for a subset of hypotheses over another, multiple peaks make optimisation considerably harder
- still a point estimate, teaches us very little about the overall model (set of assumptions)

"I read before that Bayesian priors are just like regularisers,

Constraining MLE

Maximum a posteriori

$$h^\star = \arg\max_h \; P(d|h)P(h) \\ = \arg\max_h \; \log P(d|h) + \log P(h) \tag{4}$$

- perhaps fine if $P(h)$ has a single narrow peak
- priors often indicate preference for a subset of hypotheses over another, multiple peaks make optimisation considerably harder
- still a point estimate, teaches us very little about the overall model (set of assumptions)

"I read before that Bayesian priors are just like regularisers, I even know that a Gaussian prior is just L_2 regularisation"

Constraining MLE

Maximum a posteriori

$$\begin{aligned} h^\star &= \arg\max_h \; P(d|h)P(h) \\ &= \arg\max_h \; \log P(d|h) + \log P(h) \end{aligned} \quad (4)$$

- perhaps fine if $P(h)$ has a single narrow peak
- priors often indicate preference for a subset of hypotheses over another, multiple peaks make optimisation considerably harder
- still a point estimate, teaches us very little about the overall model (set of assumptions)

"I read before that Bayesian priors are just like regularisers, I even know that a Gaussian prior is just L_2 regularisation"

- that only covers the specification of a prior

Constraining MLE

Maximum a posteriori

$$
\begin{aligned}
h^\star &= \arg\max_h \; P(d|h)P(h) \\
&= \arg\max_h \; \log P(d|h) + \log P(h)
\end{aligned}
\tag{4}
$$

- perhaps fine if $P(h)$ has a single narrow peak
- priors often indicate preference for a subset of hypotheses over another, multiple peaks make optimisation considerably harder
- still a point estimate, teaches us very little about the overall model (set of assumptions)

"I read before that Bayesian priors are just like regularisers, I even know that a Gaussian prior is just L_2 regularisation"

- that only covers the specification of a prior
- Bayesian modelling does not end at prior specification you need the crucial part: posterior inference

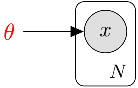
A Bayesian model

In a Bayesian model, parameters are no different from data

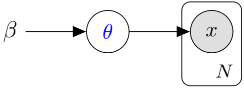
- they are random variables much like data

A Bayesian model

Frequentist



Bayesian



In a Bayesian model, parameters are no different from data

- they are random variables much like data
- only they are not observed

A Bayesian model

In a Bayesian model, parameters are no different from data

- they are random variables much like data
- only they are not observed

Bayesians do condition on deterministic quantities

- β here are called *hyperparameters*

A Bayesian model

In a Bayesian model, parameters are no different from data

- they are random variables much like data
- only they are not observed

Bayesians do condition on deterministic quantities

- β here are called *hyperparameters*
- but most Bayesians leave those fixed (no search!)

A Bayesian model

In a Bayesian model, parameters are no different from data

- they are random variables much like data
- only they are not observed

Bayesians do condition on deterministic quantities

- β here are called *hyperparameters*
- but most Bayesians leave those fixed (no search!)

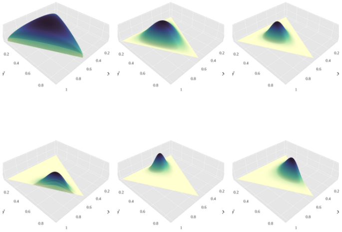
We will study an example that illustrates important concepts
Dirichlet-Multinomial model

Dirichlet distribution

A distribution over the open simplex of K-dimensional vectors

we denote the simplex by

$$\Delta_{K-1} = \left\{ \theta \in \mathbb{R}^K_{>0} : \sum_{k=1}^{K} \theta_k = 1 \right\} \subseteq \mathbb{R}^K_{>0} \tag{5}$$



Use this notebook and this wikipage to learn more

Count vector

For observations $\mathbf{x}$, where x_i is 1 of K
define $n^{(\mathbf{x})}$ as the K-dimensional vector such that

$$n_k = \sum_{i=1}^{N} [x_i = k] \tag{6}$$

Count vector

For observations $\mathbf{x}$, where x_i is 1 of K
define $n^{(\mathbf{x})}$ as the K-dimensional vector such that

$$n_k = \sum_{i=1}^{N} [x_i = k] \qquad (6)$$

Example: for $K = 3$ and $N = 6$

$$\mathbf{x} = \langle x_1 = 2, x_2 = 3, x_3 = 1, x_4 = 2, x_5 = 2, x_6 = 3 \rangle$$

$$n^{(\mathbf{x})} =$$

Count vector

For observations $\mathbf{x}$, where x_i is 1 of K
define $n^{(\mathbf{x})}$ as the K-dimensional vector such that

$$n_k = \sum_{i=1}^{N} [x_i = k] \tag{6}$$

Example: for $K = 3$ and $N = 6$

$$\mathbf{x} = \langle x_1 = 2, x_2 = 3, x_3 = 1, x_4 = 2, x_5 = 2, x_6 = 3 \rangle$$
$$n^{(\mathbf{x})} = \langle n_1 = 1, n_2 = 3, n_3 = 2 \rangle$$

Gamma function

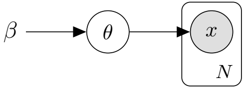
A generalisation of the factorial function to $\mathbb{R}$

$$\Gamma(z) = \int_0^\infty \epsilon^{z-1} \exp(-\epsilon) \mathrm{d}\epsilon \tag{7}$$

Properties

- $\Gamma(n) = (n-1)!$ for positive integer n
- $\Gamma(z) = (z-1)\Gamma(z-1)$

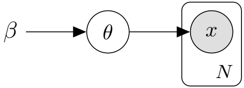
Dirchlet-Multinomial



Model

$$
\begin{aligned}
\theta|\beta &\sim \mathrm{Dir}(\beta) \\
X_i|\theta &\sim \mathrm{Cat}(\theta) \quad \text{for } i = 1, \ldots, N
\end{aligned}
\tag{8}
$$

Dirchlet-Multinomial



Model

$$\begin{aligned} \theta|\beta &\sim \mathrm{Dir}(\beta) \\ X_i|\theta &\sim \mathrm{Cat}(\theta) \quad \text{for } i = 1, \ldots, N \end{aligned} \tag{8}$$

Joint distribution

$$\begin{aligned} P(\mathbf{x}, \theta|\beta) &= P(\theta)P(\mathbf{x}|\theta) \\ &= \mathrm{Dir}(\theta|\beta)\,\mathrm{Mult}(n^{(\mathbf{x})}|\theta, N) \end{aligned} \tag{9}$$

Multinomial likelihood

For $\theta \in \Delta_{K-1}$

$$P(\mathbf{x}|\theta) = \mathrm{Mult}(n^{(\mathbf{x})}|\theta, N)$$

Multinomial likelihood

For $\theta \in \Delta_{K-1}$

$$\begin{aligned} P(\mathbf{x}|\theta) &= \mathrm{Mult}(n^{(\mathbf{x})}|\theta, N) \\ &= \frac{N!}{\prod_{k=1}^{K} n_k!} \prod_{k=1}^{K} \theta_k^{n_k} \end{aligned}$$

Multinomial likelihood

For $\theta \in \Delta_{K-1}$

$$\begin{aligned} P(\mathbf{x}|\theta) &= \mathrm{Mult}(n^{(\mathbf{x})}|\theta, N) \\ &= \frac{N!}{\prod_{k=1}^{K} n_k!} \prod_{k=1}^{K} \theta_k^{n_k} \\ &= \frac{\Gamma(\sum_{k=1}^{K} n_k + 1)}{\prod_{k=1}^{K} \Gamma(n_k + 1)} \prod_{k=1}^{K} \theta_k^{n_k} \end{aligned}$$

Multinomial likelihood

For $\theta \in \Delta_{K-1}$

$$\begin{aligned} P(\mathbf{x}|\theta) &= \mathrm{Mult}(n^{(\mathbf{x})}|\theta, N) \\ &= \frac{N!}{\prod_{k=1}^{K} n_k!} \prod_{k=1}^{K} \theta_k^{n_k} \\ &= \frac{\Gamma(\sum_{k=1}^{K} n_k + 1)}{\prod_{k=1}^{K} \Gamma(n_k + 1)} \prod_{k=1}^{K} \theta_k^{n_k} \end{aligned} \tag{10}$$

Example: for $K = 3$ and $N = 6$

$$\begin{aligned} \theta &= \langle \theta_1 = 0.2, \theta_2 = 0.3, \theta_3 = 0.5 \rangle \\ \mathbf{x} &= \langle x_1 = 2, x_2 = 3, x_3 = 1, x_4 = 2, x_5 = 2, x_6 = 3 \rangle \\ n^{(\mathbf{x})} &= \langle n_1 = 1, n_2 = 3, n_3 = 2 \rangle \end{aligned}$$

$$P(\mathbf{x}|\theta) = \frac{\Gamma(\ldots)}{\prod \ldots} \theta_1^1 \times \theta_2^3 \times \theta_3^2$$

Dirichlet prior

For $\beta \in \mathbb{R}^K_{>0}$

$$\mathrm{Dir}(\theta|\beta) = \frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}$$

Dirichlet prior

For $\beta \in \mathbb{R}^K_{>0}$

$$\begin{aligned} \mathrm{Dir}(\theta|\beta) &= \frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^{K} \theta_k^{\beta_k - 1} \\ &\propto \prod_{k=1}^{K} \theta_k^{\beta_k - 1} \end{aligned} \tag{11}$$

Dirichlet prior

For $\beta \in \mathbb{R}^K_{>0}$

$$\begin{aligned}\mathrm{Dir}(\theta|\beta) &= \frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1} \\ &\propto \prod_{k=1}^K \theta_k^{\beta_k - 1}\end{aligned} \tag{11}$$

We call

$$\int_{\Delta_{K-1}} \prod_{k=1}^K \theta_k^{\beta_k - 1} = \frac{\prod_{k=1}^K \Gamma(\beta_k)}{\Gamma(\sum_{k=1}^K \beta_k)}$$

the *Dirichlet normaliser*

Posterior

$$P(\theta|\mathbf{x}, \beta) \propto$$

Posterior

$$P(\theta|\mathbf{x}, \beta) \propto P(\mathbf{x}|\theta)P(\theta|\beta)$$

Posterior

$$
\begin{aligned}
P(\theta|\mathbf{x}, \beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^{K} n_k + 1)}{\prod_{k=1}^{K} \Gamma(n_k + 1)} \prod_{k=1}^{K} \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times
\end{aligned}
$$

Posterior

$$\begin{aligned} P(\theta|\mathbf{x},\beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\ &\propto \underbrace{\frac{\Gamma(\sum_{k=1}^K n_k + 1)}{\prod_{k=1}^K \Gamma(n_k + 1)} \prod_{k=1}^K \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}}_{\text{Dir}(\theta|\beta)} \end{aligned}$$

Posterior

$$
\begin{aligned}
P(\theta|\mathbf{x}, \beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^{K} n_k + 1)}{\prod_{k=1}^{K} \Gamma(n_k + 1)} \prod_{k=1}^{K} \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(\beta_k)} \prod_{k=1}^{K} \theta_k^{\beta_k - 1}}_{\text{Dir}(\theta|\beta)} \\
&\propto \prod_{k=1}^{K} \theta_k^{n_k} \times
\end{aligned}
$$

Posterior

$$
\begin{aligned}
P(\theta|\mathbf{x},\beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^K n_k + 1)}{\prod_{k=1}^K \Gamma(n_k + 1)} \prod_{k=1}^K \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}}_{\text{Dir}(\theta|\beta)} \\
&\propto \prod_{k=1}^K \theta_k^{n_k} \times \prod_{k=1}^K \theta_k^{\beta_k - 1}
\end{aligned}
$$

Posterior

$$
\begin{aligned}
P(\theta|\mathbf{x}, \beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^K n_k + 1)}{\prod_{k=1}^K \Gamma(n_k + 1)} \prod_{k=1}^K \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}}_{\text{Dir}(\theta|\beta)} \\
&\propto \prod_{k=1}^K \theta_k^{n_k} \times \prod_{k=1}^K \theta_k^{\beta_k - 1} \\
&= \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1}
\end{aligned}
$$

Posterior

$$
\begin{aligned}
P(\theta|\mathbf{x}, \beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^K n_k + 1)}{\prod_{k=1}^K \Gamma(n_k + 1)} \prod_{k=1}^K \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}}_{\text{Dir}(\theta|\beta)} \\
&\propto \prod_{k=1}^K \theta_k^{n_k} \times \prod_{k=1}^K \theta_k^{\beta_k - 1} \\
&= \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \propto \text{Dir}(\theta|n^{(\mathbf{x})} + \beta)
\end{aligned}
$$

Posterior

$$
\begin{aligned}
P(\theta|\mathbf{x},\beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^K n_k + 1)}{\prod_{k=1}^K \Gamma(n_k+1)} \prod_{k=1}^K \theta_k^{n_k}}_{\text{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}}_{\text{Dir}(\theta|\beta)} \\
&\propto \prod_{k=1}^K \theta_k^{n_k} \times \prod_{k=1}^K \theta_k^{\beta_k - 1} \\
&= \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \propto \text{Dir}(\theta|n^{(\mathbf{x})} + \beta)
\end{aligned}
$$

Thus

$$
P(\theta|\mathbf{x},\beta) = \underbrace{\qquad\qquad\qquad}_{\frac{1}{\text{normaliser}} \text{ of } \text{Dir}(n^{(\mathbf{x})}+\beta)} \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \tag{12}
$$

Posterior

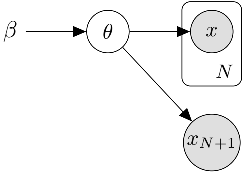
$$
\begin{aligned}
P(\theta|\mathbf{x}, \beta) &\propto P(\mathbf{x}|\theta)P(\theta|\beta) \\
&\propto \underbrace{\frac{\Gamma(\sum_{k=1}^K n_k + 1)}{\prod_{k=1}^K \Gamma(n_k + 1)} \prod_{k=1}^K \theta_k^{n_k}}_{\mathrm{Mult}(n^{(\mathbf{x})}|\theta)} \times \underbrace{\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \prod_{k=1}^K \theta_k^{\beta_k - 1}}_{\mathrm{Dir}(\theta|\beta)} \\
&\propto \prod_{k=1}^K \theta_k^{n_k} \times \prod_{k=1}^K \theta_k^{\beta_k - 1} \\
&= \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \propto \mathrm{Dir}(\theta|n^{(\mathbf{x})} + \beta)
\end{aligned}
$$

Thus

$$
P(\theta|\mathbf{x}, \beta) = \underbrace{\frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)}}_{\frac{1}{\text{normaliser}} \text{ of } \mathrm{Dir}(n^{(\mathbf{x})} + \beta)} \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \tag{12}
$$

Posterior predictive distribution

Suppose a new data point $x_{N+1} = j$ is available

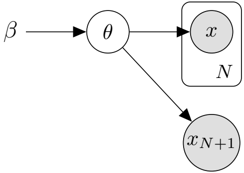


$$P(x_{N+1} = j | \mathbf{x}, \beta) = \int_{\Delta_{K-1}} P(\theta, x_{N+1} | \mathbf{x}, \beta) \mathrm{d}\theta$$

x_{N+1} is independent of $\mathbf{x}$ given θ

Posterior predictive distribution

Suppose a new data point $x_{N+1} = j$ is available



$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} P(\theta, x_{N+1}|\mathbf{x}, \beta) \mathrm{d}\theta$$

$$= \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

x_{N+1} is independent of $\mathbf{x}$ given θ

Posterior predictive distribution (cont.)

Suppose a new data point $x_{N+1} = j$ is available

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

Posterior predictive distribution (cont.)

Suppose a new data point $x_{N+1} = j$ is available

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \int_{\Delta_{K-1}} \theta_j \times \qquad \mathrm{d}\theta$$

Posterior predictive distribution (cont.)

Suppose a new data point $x_{N+1} = j$ is available

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \int_{\Delta_{K-1}} \theta_j \times \underbrace{\frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta$$

Posterior predictive distribution (cont.)

Suppose a new data point $x_{N+1} = j$ is available

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \int_{\Delta_{K-1}} \theta_j \times \underbrace{\frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta$$

$$= \underbrace{\frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \int_{\Delta_{K-1}} \theta_j \times \qquad \mathrm{d}\theta$$

Posterior predictive distribution (cont.)

Suppose a new data point $x_{N+1} = j$ is available

$$P(x_{N+1} = j | \mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j | \theta)}_{\text{likelihood}} \underbrace{P(\theta | \mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \int_{\Delta_{K-1}} \theta_j \times \underbrace{\frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \prod_{k=1}^{K} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta$$

$$= \underbrace{\frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \int_{\Delta_{K-1}} \theta_j \times \underbrace{\theta_j^{n_j + \beta_j - 1} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1}}_{\prod_{k=1}^{K} \theta_k^{n_k + \beta_k - 1}} \mathrm{d}\theta$$

Posterior predictive distribution (cont.)

Suppose a new data point $x_{N+1} = j$ is available

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \int_{\Delta_{K-1}} \theta_j \times \underbrace{\frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta$$

$$= \underbrace{\frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)}}_{\text{constant wrt } \theta} \int_{\Delta_{K-1}} \theta_j \times \underbrace{\theta_j^{n_j + \beta_j - 1} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1}}_{\prod_{k=1}^K \theta_k^{n_k + \beta_k - 1}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)} \int_{\Delta_{K-1}} \theta_j^{n_j + \beta_j} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j + \beta_j} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j+\beta_j} \prod_{k \neq j} \theta_k^{n_k+\beta_k-1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j + \beta_j} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \rule{8cm}{0.4pt}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j + \beta_j} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \frac{}{\Gamma(N + \sum_{k=1}^{K} \beta_k + 1)}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j + \beta_j} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \frac{\Gamma(n_j + \beta_j + 1) \prod_{k \neq j} \Gamma(n_k + \beta_k)}{\Gamma(N + \sum_{k=1}^{K} \beta_k + 1)}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j + \beta_j} \prod_{k \neq j} \theta_k^{n_k + \beta_k - 1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \frac{\Gamma(n_j + \beta_j + 1) \prod_{k \neq j} \Gamma(n_k + \beta_k)}{\Gamma(N + \sum_{k=1}^{K} \beta_k + 1)}$$

$$= \frac{\Gamma(N + \sum_{k=1}^{K} \beta_k)}{\prod_{k=1}^{K} \Gamma(n_k + \beta_k)} \frac{(n_j + \beta_j)\Gamma(n_j + \beta_j) \prod_{k \neq j} \Gamma(n_k + \beta_k)}{(N + \sum_{k=1}^{K} \beta_k)\Gamma(N + \sum_{k=1}^{K} \beta_k)}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1}=j|\mathbf{x},\beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1}=j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x},\beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N+\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k+\beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j+\beta_j} \prod_{k\neq j} \theta_k^{n_k+\beta_k-1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

$$= \frac{\Gamma(N+\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k+\beta_k)} \frac{\Gamma(n_j+\beta_j+1)\prod_{k\neq j}\Gamma(n_k+\beta_k)}{\Gamma(N+\sum_{k=1}^K \beta_k+1)}$$

$$= \frac{\Gamma(N+\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k+\beta_k)} \frac{(n_j+\beta_j)\Gamma(n_j+\beta_j)\prod_{k\neq j}\Gamma(n_k+\beta_k)}{(N+\sum_{k=1}^K \beta_k)\Gamma(N+\sum_{k=1}^K \beta_k)}$$

$$= \frac{\Gamma(N+\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k+\beta_k)} \frac{(n_j+\beta_j)\Gamma(n_j+\beta_j)\prod_{k\neq j}\Gamma(n_k+\beta_k)}{(N+\sum_{k=1}^K \beta_k)\Gamma(N+\sum_{k=1}^K \beta_k)}$$

Posterior predictive distribution (cont.)

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \int_{\Delta_{K-1}} \underbrace{P(x_{N+1} = j|\theta)}_{\text{likelihood}} \underbrace{P(\theta|\mathbf{x}, \beta)}_{\text{posterior}} \mathrm{d}\theta$$

$$= \frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)} \underbrace{\int_{\Delta_{K-1}} \theta_j^{n_j+\beta_j} \prod_{k \neq j} \theta_k^{n_k+\beta_k-1} \mathrm{d}\theta}_{\text{Dir normaliser}}$$

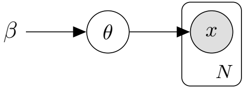
$$= \frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)} \frac{\Gamma(n_j + \beta_j + 1) \prod_{k \neq j} \Gamma(n_k + \beta_k)}{\Gamma(N + \sum_{k=1}^K \beta_k + 1)}$$

$$= \frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)} \frac{(n_j + \beta_j)\Gamma(n_j + \beta_j) \prod_{k \neq j} \Gamma(n_k + \beta_k)}{(N + \sum_{k=1}^K \beta_k)\Gamma(N + \sum_{k=1}^K \beta_k)}$$

$$= \frac{\Gamma(N + \sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(n_k + \beta_k)} \frac{(n_j + \beta_j)\Gamma(n_j + \beta_j) \prod_{k \neq j} \Gamma(n_k + \beta_k)}{(N + \sum_{k=1}^K \beta_k)\Gamma(N + \sum_{k=1}^K \beta_k)}$$

$$= \frac{n_j + \beta_j}{N + \sum_{k=1}^K \beta_k}$$

Dirchlet-Multinomial (overview)



Joint distribution

$$\begin{aligned} P(\mathbf{x}, \theta|\beta) &= P(\theta)P(\mathbf{x}|\theta) \\ &= \mathrm{Dir}(\theta|\beta)\,\mathrm{Mult}(n^{(\mathbf{x})}|\theta, N) \end{aligned} \tag{13}$$

Posterior

$$P(\theta|\mathbf{x}, \beta) = \mathrm{Dir}(\theta|n^{(\mathbf{x})} + \beta) \tag{14}$$

Predictive posterior

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \frac{n_j + \beta_j}{N + \sum_{k=1}^{K} \beta_k} \tag{15}$$

Exchangeability

Random variables are called exchangeable under a model when all permutations of the set of outcomes have the same probability

Exchangeability

Random variables are called exchangeable under a model when all permutations of the set of outcomes have the same probability

- in our Dirichlet-Multinomial model any re-ordering of the observations is equally likely to occur

Exchangeability

Random variables are called exchangeable under a model when all permutations of the set of outcomes have the same probability

- in our Dirichlet-Multinomial model any re-ordering of the observations is equally likely to occur

Combine that fact with the predictive posterior result

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \frac{n_j + \beta_j}{N + \sum_{k=1}^{K} \beta_k} \tag{16}$$

Exchangeability

Random variables are called exchangeable under a model when all permutations of the set of outcomes have the same probability

- in our Dirichlet-Multinomial model any re-ordering of the observations is equally likely to occur

Combine that fact with the predictive posterior result

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \frac{n_j + \beta_j}{N + \sum_{k=1}^{K} \beta_k} \tag{16}$$

and we can single out any observation, e.g. $\mathbf{x}_i$

$$P(\mathbf{x}_i = j|\mathbf{x}_{-i}, \beta) = \underline{\qquad\qquad\qquad} \tag{17}$$

Exchangeability

Random variables are called exchangeable under a model when all permutations of the set of outcomes have the same probability

- in our Dirichlet-Multinomial model any re-ordering of the observations is equally likely to occur

Combine that fact with the predictive posterior result

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \frac{n_j + \beta_j}{N + \sum_{k=1}^{K} \beta_k} \tag{16}$$

and we can single out any observation, e.g. $\mathbf{x}_i$

$$P(\mathbf{x}_i = j|\mathbf{x}_{-i}, \beta) = \frac{}{N - 1 + \sum_{k=1}^{K} \beta_k} \tag{17}$$

Exchangeability

Random variables are called exchangeable under a model when all permutations of the set of outcomes have the same probability

- in our Dirichlet-Multinomial model any re-ordering of the observations is equally likely to occur

Combine that fact with the predictive posterior result

$$P(x_{N+1} = j|\mathbf{x}, \beta) = \frac{n_j + \beta_j}{N + \sum_{k=1}^{K} \beta_k} \tag{16}$$

and we can single out any observation, e.g. $\mathbf{x}_i$

$$P(\mathbf{x}_i = j|\mathbf{x}_{-i}, \beta) = \frac{n_j^{(\mathbf{x}_{-i})} + \beta_j}{N - 1 + \sum_{k=1}^{K} \beta_k} \tag{17}$$

Summary

Friends do not let friends optimise

Summary

Friends do not let friends optimise

- no point estimates, we use all possible model parameters

Summary

Friends do not let friends optimise

- no point estimates, we use all possible model parameters
- this is called *Bayesian inference*, or simply, inference

Summary

Friends do not let friends optimise

- no point estimates, we use all possible model parameters
- this is called *Bayesian inference*, or simply, inference
- Bayesian models have memory: the posterior summarises what we learnt from data

Summary

Friends do not let friends optimise

- no point estimates, we use all possible model parameters
- this is called *Bayesian inference*, or simply, inference
- Bayesian models have memory: the posterior summarises what we learnt from data
- If we collect more data $\mathbf{x}'$, we can update the posterior, $P(\theta|\mathbf{x}, \mathbf{x}', \beta) = \mathrm{Dir}(\theta|n^{(\mathbf{x})} + n^{(\mathbf{x}')} + \beta)$

Summary

Friends do not let friends optimise

- no point estimates, we use all possible model parameters
- this is called *Bayesian inference*, or simply, inference
- Bayesian models have memory: the posterior summarises what we learnt from data
- If we collect more data $\mathbf{x}'$, we can update the posterior, $P(\theta|\mathbf{x}, \mathbf{x}', \beta) = \mathrm{Dir}(\theta|n^{(\mathbf{x})} + n^{(\mathbf{x}')} + \beta)$
- MLE is memoryless: there is one fixed θ, no matter how much more data you see, θ will never change

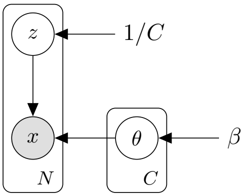
NLP1

Preliminaries

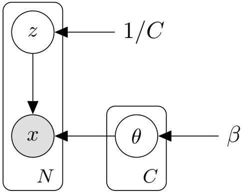
Bayesian modelling

Applications

Bayesian mixture model with categorical observations



Bayesian mixture model with categorical observations

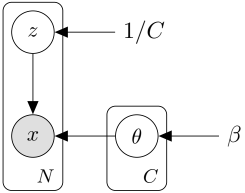


Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$n_{c,k} = \sum_{j \neq i} [z_j = c][x_j = k]$$

$$n_c = \sum_{k=1}^{K} n_{c,k}$$

Bayesian mixture model with categorical observations



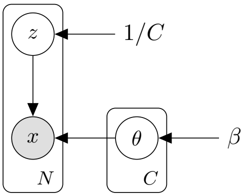
Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$n_{c,k} = \sum_{j \neq i} [z_j = c][x_j = k]$$

$$n_c = \sum_{k=1}^{K} n_{c,k}$$

$$P(z_i = c|\mathbf{x}, \mathbf{z}_{-i}, \beta) \propto P(x_i = k, z_i = c|\mathbf{x}_{-i}, \mathbf{z}_{-i}, \alpha, \beta)$$

Bayesian mixture model with categorical observations



Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$n_{c,k} = \sum_{j \neq i} [z_j = c][x_j = k]$$

$$n_c = \sum_{k=1}^{K} n_{c,k}$$

$$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \beta) \propto P(x_i = k, z_i = c | \mathbf{x}_{-i}, \mathbf{z}_{-i}, \alpha, \beta)$$

$$\propto \frac{1}{C} \times \frac{n_{c,k} + \beta}{n_c + K\beta}$$

Mixing weights

What does it mean to have uniform prior over components?

Mixing weights

What does it mean to have uniform prior over components?

- unlike it may seem, it does not mean to promote diversity!

Let's see whether the posterior is *peaked*

$$P(z|x) = \frac{\frac{1}{C} \times P(x|z)}{P(x)} \propto P(x|z)$$

Mixing weights

What does it mean to have uniform prior over components?

- unlike it may seem, it does not mean to promote diversity!

Let's see whether the posterior is *peaked*

$$P(z|x) = \frac{\frac{1}{C} \times P(x|z)}{P(x)} \propto P(x|z)$$

- uniform prior leaves it up to the likelihood to control sparsity

Mixing weights

What does it mean to have uniform prior over components?

- unlike it may seem, it does not mean to promote diversity!

Let's see whether the posterior is *peaked*

$$P(z|x) = \frac{\frac{1}{C} \times P(x|z)}{P(x)} \propto P(x|z)$$

- uniform prior leaves it up to the likelihood to control sparsity
- luckily we are promoting sparse likelihoods $X|z$
  because $\theta^{(z)} \sim \mathrm{Dir}(\beta)$

Mixing weights

What does it mean to have uniform prior over components?

- unlike it may seem, it does not mean to promote diversity!

Let's see whether the posterior is *peaked*

$$P(z|x) = \frac{\frac{1}{C} \times P(x|z)}{P(x)} \propto P(x|z)$$

- uniform prior leaves it up to the likelihood to control sparsity
- luckily we are promoting sparse likelihoods $X|z$ because $\theta^{(z)} \sim \mathrm{Dir}(\beta)$
- but $P(z)$ has nothing to do with it!

Mixing weights

What does it mean to have uniform prior over components?

- unlike it may seem, it does not mean to promote diversity!

Let's see whether the posterior is *peaked*

$$P(z|x) = \frac{\frac{1}{C} \times P(x|z)}{P(x)} \propto P(x|z)$$

- uniform prior leaves it up to the likelihood to control sparsity
- luckily we are promoting sparse likelihoods $X|z$ because $\theta^{(z)} \sim \mathrm{Dir}(\beta)$
- but $P(z)$ has nothing to do with it!

Is there really no preference we can express about $P(z)$?

Mixing weights

What does it mean to have uniform prior over components?

- unlike it may seem, it does not mean to promote diversity!

Let's see whether the posterior is *peaked*

$$P(z|x) = \frac{\frac{1}{C} \times P(x|z)}{P(x)} \propto P(x|z)$$

- uniform prior leaves it up to the likelihood to control sparsity
- luckily we are promoting sparse likelihoods $X|z$ because $\theta^{(z)} \sim \mathrm{Dir}(\beta)$
- but $P(z)$ has nothing to do with it!

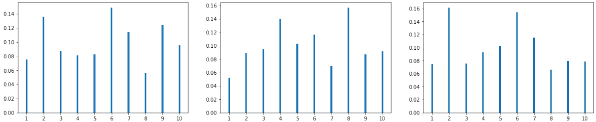
Is there really no preference we can express about $P(z)$?

- what about preferring to use fewer components?

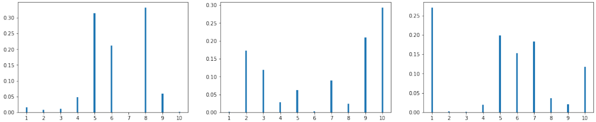
Sparse prior over mixing weights

Say we have 10 components, how do you want to use them?

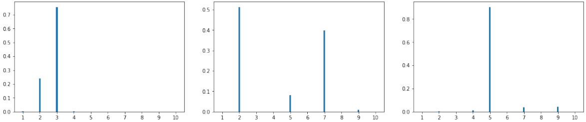
I couldn't care less



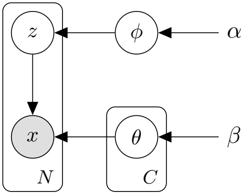
Sparingly



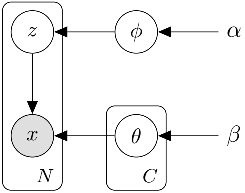
Like I pass students



Bayesian mixture model - Sparse prior over mixing weights



Bayesian mixture model - Sparse prior over mixing weights

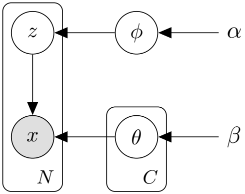


Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$n_{c,k} = \sum_{j \neq i} [z_j = c][x_j = k]$$

$$n_c = \sum_{k=1}^{K} n_{c,k}$$

Bayesian mixture model - Sparse prior over mixing weights



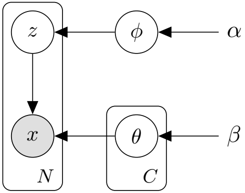
Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$n_{c,k} = \sum_{j \neq i} [z_j = c][x_j = k]$$

$$n_c = \sum_{k=1}^{K} n_{c,k}$$

$$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \alpha, \beta) \propto P(x_i = k, z_i = c | \mathbf{x}_{-i}, \mathbf{z}_{-i}, \alpha, \beta)$$

Bayesian mixture model - Sparse prior over mixing weights



Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

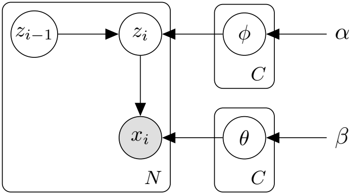
$$n_{c,k} = \sum_{j \neq i} [z_j = c][x_j = k]$$

$$n_c = \sum_{k=1}^{K} n_{c,k}$$

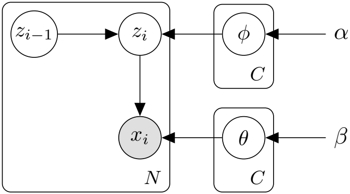
$$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \alpha, \beta) \propto P(x_i = k, z_i = c | \mathbf{x}_{-i}, \mathbf{z}_{-i}, \alpha, \beta)$$

$$\propto \frac{n_c + \alpha}{N - 1 + C\alpha} \times \frac{n_{c,k} + \beta}{n_c + K\beta}$$

Bayesian HMM



Bayesian HMM

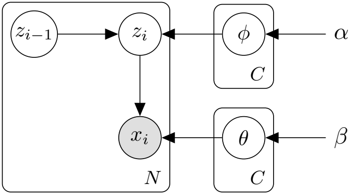


Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$m_{b,c} = \sum_{j \neq i} [z_{j-1} = b][z_j = c]$$

$$m_b = \sum_{c=1}^{C} m_{b,c}$$

Bayesian HMM



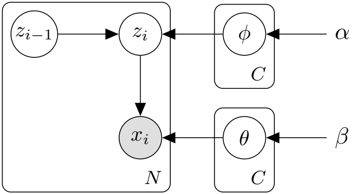
$$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \alpha, \beta)$$

Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$m_{b,c} = \sum_{j \neq i} [z_{j-1} = b][z_j = c]$$

$$m_b = \sum_{c=1}^{C} m_{b,c}$$

Bayesian HMM



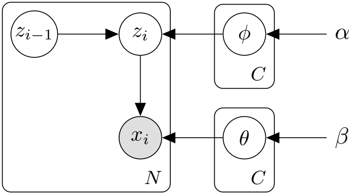
Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$m_{b,c} = \sum_{j \neq i} [z_{j-1} = b][z_j = c]$$

$$m_b = \sum_{c=1}^{C} m_{b,c}$$

$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \alpha, \beta)$ note that $\begin{cases} z_{i-1} = b \\ z_{i+1} = d \end{cases}$ is in $\mathbf{z}_{-i}$

Bayesian HMM



Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

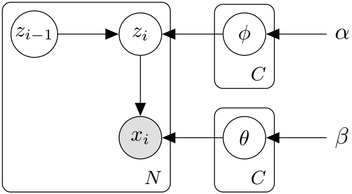
$$m_{b,c} = \sum_{j \neq i} [z_{j-1} = b][z_j = c]$$

$$m_b = \sum_{c=1}^{C} m_{b,c}$$

$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \alpha, \beta)$ note that $\begin{cases} z_{i-1} = b \\ z_{i+1} = d \end{cases}$ is in $\mathbf{z}_{-i}$

$$\propto P(z_{i-1} = b, z_i = c, z_{i+1} = d, x_i = k | \mathbf{x}_{-i}, \mathbf{z}_{-i}, \alpha, \beta)$$

Bayesian HMM



Define counts based on joint assignments to $\mathbf{x}_{-i}, \mathbf{z}_{-i}$

$$m_{b,c} = \sum_{j \neq i} [z_{j-1} = b][z_j = c]$$

$$m_b = \sum_{c=1}^{C} m_{b,c}$$

$P(z_i = c | \mathbf{x}, \mathbf{z}_{-i}, \alpha, \beta)$ note that $\begin{cases} z_{i-1} = b \\ z_{i+1} = d \end{cases}$ is in $\mathbf{z}_{-i}$

$$\propto P(z_{i-1} = b, z_i = c, z_{i+1} = d, x_i = k | \mathbf{x}_{-i}, \mathbf{z}_{-i}, \alpha, \beta)$$

$$\propto \frac{m_{b,c} + \alpha}{m_b + C\alpha} \times \frac{n_{c,k} + \beta}{n_c + K\beta} \times \frac{m_{c,d} + \alpha}{m_c + C\alpha}$$

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$
- under certain conditions the chain converges to a stationary distribution π such that $\mathbf{P}\pi = \pi$

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$
- under certain conditions the chain converges to a stationary distribution π such that $\mathbf{P}\pi = \pi$
- possible states are assignments to the variables in the model

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$
- under certain conditions the chain converges to a stationary distribution π such that $\mathbf{P}\pi = \pi$
- possible states are assignments to the variables in the model
- by defining $\mathbf{P}$ properly we guarantee that π is the true posterior

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$
- under certain conditions the chain converges to a stationary distribution π such that $\mathbf{P}\pi = \pi$
- possible states are assignments to the variables in the model
- by defining $\mathbf{P}$ properly we guarantee that π is the true posterior
- once the chain has converged each y_t will be a sample from the posterior

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$
- under certain conditions the chain converges to a stationary distribution π such that $\mathbf{P}\pi = \pi$
- possible states are assignments to the variables in the model
- by defining $\mathbf{P}$ properly we guarantee that π is the true posterior
- once the chain has converged each y_t will be a sample from the posterior
- we can design $\mathbf{P}$ by decomposing it $P_1 \cdots P_M$
  where each component satisfies $P_k(y, y')\pi(y) = P_k(y', y)\pi(y')$

Markov Chain Monte Carlo

We draw from the posterior $P(\mathbf{z}|\mathbf{x})$ via a Markov chain of random states $Y_1, \ldots, Y_T$ where $P(y_t|y_{<t}) = P(y_t|y_{t-1})$

- the transition probability from y to y' is coded in a matrix $\mathbf{P}$
  P_{ij} corresponds to $P(Y = i, Y = j)$
- under certain conditions the chain converges to a stationary distribution π such that $\mathbf{P}\pi = \pi$
- possible states are assignments to the variables in the model
- by defining $\mathbf{P}$ properly we guarantee that π is the true posterior
- once the chain has converged each y_t will be a sample from the posterior
- we can design $\mathbf{P}$ by decomposing it $P_1 \cdots P_M$
  where each component satisfies $P_k(y, y')\pi(y) = P_k(y', y)\pi(y')$
- applying each of P_k in turn or choosing P_k at random produces a $\mathbf{P}$ that satisfies the necessary conditions

Gibbs sampler

We want to sample from $P(\mathbf{z}|\mathbf{x})$ with a Markov chain
a state $y_t = \mathbf{z}^{(t)}$ is the t-th assignment to $\mathbf{z}$

Gibbs sampler

We want to sample from $P(\mathbf{z}|\mathbf{x})$ with a Markov chain
a state $y_t = \mathbf{z}^{(t)}$ is the t-th assignment to $\mathbf{z}$

To obtain a new state we

1. start a draft state $\mathbf{z} = \mathbf{z}^{(t-1)}$

Gibbs sampler

We want to sample from $P(\mathbf{z}|\mathbf{x})$ with a Markov chain
a state $y_t = \mathbf{z}^{(t)}$ is the t-th assignment to $\mathbf{z}$

To obtain a new state we

1. start a draft state $\mathbf{z} = \mathbf{z}^{(t-1)}$
2. repeat for $i = 1, \ldots, N$

Gibbs sampler

We want to sample from $P(\mathbf{z}|\mathbf{x})$ with a Markov chain
a state $y_t = \mathbf{z}^{(t)}$ is the t-th assignment to $\mathbf{z}$

To obtain a new state we

1. start a draft state $\mathbf{z} = \mathbf{z}^{(t-1)}$
2. repeat for $i = 1, \ldots, N$
   - resample $Z_i \sim P(z_i|\mathbf{x}_{-i}, \mathbf{z}_{-i})$
     only variables in the Markov blanket of z_i play a role
     that's why this is feasible

Gibbs sampler

We want to sample from $P(\mathbf{z}|\mathbf{x})$ with a Markov chain
a state $y_t = \mathbf{z}^{(t)}$ is the t-th assignment to $\mathbf{z}$

To obtain a new state we

1. start a draft state $\mathbf{z} = \mathbf{z}^{(t-1)}$
2. repeat for $i = 1, \ldots, N$
   - resample $Z_i \sim P(z_i|\mathbf{x}_{-i}, \mathbf{z}_{-i})$
     only variables in the Markov blanket of z_i play a role
     that's why this is feasible
3. after complete pass over the data we have a new state $\mathbf{z}^{(t)}$

Gibbs sampler

We want to sample from $P(\mathbf{z}|\mathbf{x})$ with a Markov chain
a state $y_t = \mathbf{z}^{(t)}$ is the t-th assignment to $\mathbf{z}$

To obtain a new state we

1. start a draft state $\mathbf{z} = \mathbf{z}^{(t-1)}$
2. repeat for $i = 1, \ldots, N$
   - resample $Z_i \sim P(z_i|\mathbf{x}_{-i}, \mathbf{z}_{-i})$
     only variables in the Markov blanket of z_i play a role
     that's why this is feasible
3. after complete pass over the data we have a new state $\mathbf{z}^{(t)}$

When we have collected a large number T of samples

- we can summarise the distribution and/or make decisions

Summary

- Friends don't let friends optimise

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification
- Bayesians compare models (a set of assumptions)
  not point estimates

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification
- Bayesians compare models (a set of assumptions)
  not point estimates
- Comparing Bayesian models is easier

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification
- Bayesians compare models (a set of assumptions)
  not point estimates
- Comparing Bayesian models is easier
- Bayesian modelling requires some maths ;)

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification
- Bayesians compare models (a set of assumptions) not point estimates
- Comparing Bayesian models is easier
- Bayesian modelling requires some maths ;)
- Some families enjoy analytically available posteriors

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification
- Bayesians compare models (a set of assumptions)
  not point estimates
- Comparing Bayesian models is easier
- Bayesian modelling requires some maths ;)
- Some families enjoy analytically available posteriors
- Inference can be done by simulation (MCMC)

Summary

- Friends don't let friends optimise
- Bayesian modelling is not only about prior specification
- Bayesian modelling is about uncertainty quantification
- Bayesians compare models (a set of assumptions) not point estimates
- Comparing Bayesian models is easier
- Bayesian modelling requires some maths ;)
- Some families enjoy analytically available posteriors
- Inference can be done by simulation (MCMC)

Beyond

For more on latent variable modelling, especially with structured data

- take NLP2
- though most of it will be *frequentist* (for very good reasons!)

For more on Bayesian modelling, approximate inference, and probabilistic modelling with neural networks

- take ML4NLP
- though MCMC will not be the method of choice, instead we will look into *variational inference*
- and we will need to count on optimisation =O
- though with a nice twist ;)

References I