# Part IV: Monte Carlo and nonparametric Bayes

## Outline

#### Monte Carlo methods

#### Nonparametric Bayesian models

## Outline

#### Monte Carlo methods

#### Nonparametric Bayesian models

## The Monte Carlo principle

• The expectation of *f* with respect to *P* can be approximated by

$$E_{P(x)}[f(x)] \approx \frac{1}{n} \sum_{i=1}^{n} f(x_i)$$

where the  $x_i$  are sampled from P(x)

• Example: the average # of spots on a die roll





## Two uses of Monte Carlo methods

- 1. For solving problems of probabilistic inference involved in developing computational models
- 2. As a source of hypotheses about how the mind might solve problems of probabilistic inference

# Making Bayesian inference easier

$$P(h \mid d) = \frac{P(d \mid h)P(h)}{\sum_{h' \in H} P(d \mid h')P(h')}$$

Evaluating the posterior probability of a hypothesis requires considering all hypotheses

Modern Monte Carlo methods let us avoid this

1 /	113 161	1737 1 1	11/13/1	6 11 111
1	1171111080	84860686	11134 15	172144
1 11/11/	0917148002	50233070	81482131	5//1/9
1 1 1 4 1 5 3	1211122408	941400000	60011/99/	
35/11/51	1825042206	87.013.0240	06405471	111921114148
1571 26	7110223464	021051261	008896/1	1 1/1/77412
1 111111	2912302004	03854220	582028031	1/412 1115(1
1 (1//	1/60263060	42668220:	186088281	1111 111/ 157
2 1 1 1	1105080020	500834031	00862221	A//// 1// 791
	1152087984	830430750	080202304	12078//11/18
5 1 11	1532224104	641044991	00039205	1 4 9 4 1 1 7 1 4 4 1 3 1
15 117	120786836	448793081	30395351	1163/127/2/11
1 1	1104711750	423487288	379008406	6/1176/7/311
411519	5664634701	1.1.4468800	698060200	2017520765651
/ 9 / 1 / / / /	2/04318126	7275/0480	86306023	124604124259
////////	1159574387	9688546dl	383604573	9206144419189
1 1111	4066132036	086243030	28002006	8860066673334
1 7557	1114228380	09202085	322260263	119577651950
1 1/11/2	51/005280	810346401	335206321	0169590484444
7 4/91	991060676	884600183	830205000	569643676706
1/1/648092	6811479825	844287583	083060588	3283661458989
26/10348020	0858011961	810000000	62050523	12865898762400
9148892480	48058219 1	810285921	072000004	688208068280
76616565851	4406803979	9983689/1	069654001	1000406028283
4745022056	0046008880	003354481	006046830	680566008608
13128080500	0838559903	19654276	578350030	030856008670
8566858726	5068586568	45235763	48308802	10438808856662
42406098950	0800080868	48830005	60324850	+890008962087
4640930486	0462837284	407068601	3202624060	295868520987
8286908800	0800000389	60220076	2220838999	1620546000598
05868898800	6242763384	973180920	355 95 7 931	2460386980939
4680026003	8659306054	06805420	000607931	1005555402880
9066802826	2664654054	10222408	260020365	3206636364326
9084400336.	2009057089	91+12223:	156048337:	2005230064502
5078602066	6830715551	38797855	083003340	0065686638002
16030208330	00031 5181	719106000	12000303	0574507082030
1011525250	2231866391	046164716	86572308	1520629303452
0028660000	8264082147	197460491	98560359	2483628034256
3095030380.	8858068441	43340045	85885230.	078704633853
08925066200	0630606441	851870584	1081850850	3682898046544
0440756609	2003898627	E 1850480	08655836	000000000000000000000000000000000000000
2036002602	5620253505	089895965	358100305	336260800862
3080090899	2000098042	830760710	244855009	5323050588390
8622330336	5406606814	703646500	09 (502630	036688600003
0026523066	3880809514	200680000	887560321	8288305335458
0250060308	1000045888	42426185	08 02022	200000000000000000000000000000000000000
0340005834	6598742885	04802354	596407006	080600030883
0275825820	0342575460	24871337	532300621	068082838246
2228030003	9938024210	00620200	221208054	033095028040
0300028000	2709803585	00708062.	5053883	608700009046
2238638349	6283484930	20230420	88550008	06600000000000000

## Modern Monte Carlo methods

- Sampling schemes for distributions with large state spaces known up to a multiplicative constant
- Two approaches:
  - importance sampling (and particle filters)
  - Markov chain Monte Carlo

## Importance sampling

Basic idea: generate from the wrong distribution, assign weights to samples to correct for this

$$E_{p(x)}[f(x)] = \int f(x)p(x)dx$$
  
=  $\int f(x)\frac{p(x)}{q(x)}q(x)dx$   
 $\approx \frac{1}{n}\sum_{i=1}^{n}f(x_i)\frac{p(x_i)}{q(x_i)}$  for  $x_i \sim q(x)$ 

## Importance sampling



works when sampling from proposal is easy, target is hard

## An alternative scheme...

$$\begin{split} E_{p(x)}[f(x)] &\approx \frac{1}{n} \sum_{i=1}^{n} f(x_i) \frac{p(x_i)}{q(x_i)} & \text{for } x_i \sim q(x) \\ E_{p(x)}[f(x)] &\approx \frac{\sum_{i=1}^{n} f(x_i) \frac{p(x_i)}{q(x_i)}}{\sum_{i=1}^{n} \frac{p(x_i)}{q(x_i)}} & \text{for } x_i \sim q(x) \end{split}$$

works when p(x) is known up to a multiplicative constant

# Likelihood weighting

- A particularly simple form of importance sampling for posterior distributions
- Use the prior as the proposal distribution
- Weights:

 $\frac{p(h \mid d)}{p(h)} = \frac{p(d \mid h)p(h)}{p(d)p(h)} = \frac{p(d \mid h)}{p(d)} \propto p(d \mid h)$ 

# Likelihood weighting

- Generate samples of all variables except observed variables
- Assign weights proportional to probability of observed data given values in sample



# Importance sampling

- A general scheme for sampling from complex distributions that have simpler relatives
- Simple methods for sampling from posterior distributions in some cases (easy to sample from prior, prior and posterior are close)
- Can be more efficient than simple Monte Carlo

   particularly for, e.g., tail probabilities
- Also provides a solution to the question of how people can update beliefs as data come in...

## Particle filtering



We want to generate samples from  $P(s_4|d_1, ..., d_4)$   $P(s_4 | d_1, ..., d_4) \propto P(d_4 | s_4) P(s_4 | d_1, ..., d_3)$   $= P(d_4 | s_4) \sum_{s_3} P(s_4 | s_3) P(s_3 | d_1, ..., d_3)$ We can use likelihood weighting if we can sample from  $P(s_4|s_3)$  and  $P(s_3|d_1, ..., d_3)$ 



# The promise of particle filters

- People need to be able to update probability distributions over large hypothesis spaces as more data become available
- Particle filters provide a way to do this with limited computing resources...
  - maintain a fixed finite number of samples
- Not just for dynamic models
  - can work with a fixed set of hypotheses, although this requires some further tricks for maintaining diversity

## Markov chain Monte Carlo

- Basic idea: construct a *Markov chain* that will converge to the target distribution, and draw samples from that chain
- Just uses something proportional to the target distribution (good for Bayesian inference!)
- Can work in state spaces of arbitrary (including unbounded) size (good for nonparametric Bayes)

## Markov chains



Variables  $\mathbf{x}^{(t+1)}$  independent of all previous variables given immediate predecessor  $\mathbf{x}^{(t)}$ 

# An example: card shuffling

- Each state **x**<sup>(*t*)</sup> is a permutation of a deck of cards (there are 52! permutations)
- Transition matrix **T** indicates how likely one permutation will become another
- The transition probabilities are determined by the shuffling procedure
  - riffle shuffle
  - overhand
  - one card

# Convergence of Markov chains

- Why do we shuffle cards?
- Convergence to a uniform distribution takes only 7 riffle shuffles...
- Other Markov chains will also converge to a *stationary distribution*, if certain simple conditions are satisfied (called "ergodicity")
  - e.g. every state can be reached in some number of steps from every other state

## Markov chain Monte Carlo



- States of chain are variables of interest
- Transition matrix chosen to give target distribution as stationary distribution

- Transitions have two parts:
  - proposal distribution:  $Q(\mathbf{x}^{(t+1)}|\mathbf{x}^{(t)})$
  - acceptance: take proposals with probability

$$A(\mathbf{x}^{(t)}, \mathbf{x}^{(t+1)}) = \min(1, \frac{P(\mathbf{x}^{(t+1)}) Q(\mathbf{x}^{(t)} | \mathbf{x}^{(t+1)})}{P(\mathbf{x}^{(t)}) Q(\mathbf{x}^{(t+1)} | \mathbf{x}^{(t)})})$$













## Metropolis-Hastings in a slide



## Gibbs sampling

#### Particular choice of proposal distribution

For variables 
$$\mathbf{x} = x_1, x_2, ..., x_n$$
  
Draw  $x_i^{(t+1)}$  from  $P(x_i | \mathbf{x}_{-i})$   
 $\mathbf{x}_{-i} = x_1^{(t+1)}, x_2^{(t+1)}, ..., x_{i-1}^{(t+1)}, x_{i+1}^{(t)}, ..., x_n^{(t)}$ 

(this is called the *full conditional* distribution)

## In a graphical model...



Sample each variable conditioned on its Markov blanket

#### Gibbs sampling $x_2$ $x_2$ $X_1$ $\Lambda\gamma$ $P(\mathbf{x})$ $\underline{P(x_1|x_2^{(t)})}$ $\mathbf{x}^{(t)}$ (a) (b) $\overline{x_1}$ $x_1$ $x_2$ $x_2$ $\mathbf{x}^{(t+2)}$ $X_2$ $\mathbf{x}^{(t+1)}$ $P(x_2|x_1)$ $\mathbf{x}^{(t)}$ (c) (d) $x_1$ $x_1$

(MacKay, 2002)

# The magic of MCMC

- Since we only ever need to evaluate the relative probabilities of two states, we can have huge state spaces (much of which we rarely reach)
- In fact, our state spaces can be *infinite* common with nonparametric Bayesian models
- But... the guarantees it provides are asymptotic making algorithms that converge in practical
  - amounts of time is a significant challenge

# MCMC and cognitive science

- The main use of MCMC is for probabilistic inference in complex models
- The Metropolis-Hastings algorithm seems like a good metaphor for aspects of development...
- A form of cultural evolution can be shown to be equivalent to Gibbs sampling (Griffiths & Kalish, 2007)
- We can also use MCMC algorithms as the basis for experiments with people...

### Samples from Subject 3 (projected onto plane from LDA)



# Three Two-uses of Monte Carlo methods

- 1. For solving problems of probabilistic inference involved in developing computational models
- 2. As a source of hypotheses about how the mind might solve problems of probabilistic inference
- 3. As a way to explore people's subjective probability distributions

## Outline

Monte Carlo methods

Nonparametric Bayesian models

# Nonparametric Bayes

- Nonparametric models...
  - can capture distributions outside parametric families
  - have infinitely many parameters
  - grow in complexity with the data
- Provide a way to automatically determine how much structure can be inferred from data
  - how many clusters?
  - how many dimensions?

## How many clusters?



#### Nonparametric approach: Dirichlet process mixture models

## Mixture models

- Each observation is assumed to come from a single (possibly previously unseen) cluster
- The probability that the *i*th sample belongs to the *k*th cluster is

$$p(z_i = k \mid x_i) \propto p(x_i \mid z_i = k) p(z_i = k)$$

• Where  $p(x_i|z_i=k)$  reflects the structure of cluster k (e.g. Gaussian) and  $p(z_i=k)$  is its prior probability

# Dirichlet process mixture models

- Use a prior that allows infinitely many clusters (but finitely many for finite observations)
- The *i*th sample is drawn from the *k*th cluster with probability

$$P(k) = \begin{cases} \frac{n_k}{i+\alpha}, & n_k > 0 \text{ (i.e., } k \text{ is old)} \\ \frac{\alpha}{i+\alpha}, & n_k = 0 \text{ (i.e., } k \text{ is new)} \end{cases}$$

where  $\alpha$  is a parameter of the model (known as the "Chinese restaurant process")

# Nonparametric Bayes and cognition

- Nonparametic Bayesian models are useful for answering questions about how much structure people should infer from data
- Many cognitive science questions take this form
  - how should we represent categories?
  - what features should we identify for objects?

# Nonparametric Bayes and cognition

- Nonparametic Bayesian models are useful for answering questions about how much structure people should infer from data
- Many cognitive science questions take this form
  - how should we represent categories?
  - what features should we identify for objects?

# The Rational Model of Categorization (RMC; Anderson 1990; 1991)

- Computational problem: predicting a feature based on observed data
  - assume that category labels are just features
- Predictions are made on the assumption that objects form clusters with similar properties
  - each object belongs to a single cluster
  - feature values likely to be the same within clusters
  - the number of clusters is unbounded

## Representation in the RMC

# Flexible representation can interpolate between prototype and exemplar models



## The "optimal solution"

The probability of the missing feature (i.e., the category label) taking a certain value is

$$P(j|F_n) = \sum_{x_n} P(j|x_n, F_n) \frac{P(x_n|F_n)}{\text{posterior over partitions}}$$

where *j* is a feature value,  $F_n$  are the observed features of a set of *n* objects, and  $x_n$  is a partition of objects into clusters

## The prior over partitions

- An object is assumed to have a constant probability of joining same cluster as another object, known as the *coupling probability*
- This allows some probability that a stimulus forms a new cluster, so the probability that the *i*th object is assigned to the *k*th cluster is

$$P(k) = \begin{cases} \frac{cn_k}{(1-c)+ci} & n_k > 0 \text{ (i.e., } k \text{ is old)} \\ \frac{(1-c)}{(1-c)+ci} & n_k = 0 \text{ (i.e., } k \text{ is new)} \end{cases}$$

## Equivalence

Neal (1998) showed that the prior for the RMC and the DPMM are the same, with

$$\alpha = (1 - c)/c$$

RMC prior: 
$$P(k) = \begin{cases} \frac{cn_k}{(1-c)+ci} & n_k > 0 \text{ (i.e., } k \text{ is old)} \\ \frac{(1-c)}{(1-c)+ci} & n_k = 0 \text{ (i.e., } k \text{ is new)} \end{cases}$$

DPMM prior: 
$$P(k) = \begin{cases} \frac{n_k}{i+\alpha}, & n_k > 0 \text{ (i.e., } k \text{ is old)} \\ \frac{\alpha}{i+\alpha}, & n_k = 0 \text{ (i.e., } k \text{ is new)} \end{cases}$$

## The computational challenge

The probability of the missing feature (i.e., the category label) taking a certain value is

$$P(j|F_n) = \sum_{x_n} P(j|x_n, F_n) P(x_n|F_n)$$

where *j* is a feature value,  $F_n$  are the observed features of a set of *n* objects, and  $x_n$  is a partition of objects into groups

 $|x_n|$  1 2 5 15 52 203 877 4140 21147 115975

# Anderson's approximation



- Data observed sequentially
- Each object is deterministically assigned to the cluster with the highest posterior probability
- Call this the "Local MAP"
  - choosing the cluster with the maximum *a posteriori* probability

## Two uses of Monte Carlo methods

- 1. For solving problems of probabilistic inference involved in developing computational models
- 2. As a source of hypotheses about how the mind might solve problems of probabilistic inference

# Alternative approximation schemes

- There are several methods for making approximations to the posterior in DPMMs
  - Gibbs sampling
  - Particle filtering
- These methods provide asymptotic performance guarantees (in contrast to Anderson's procedure)

(Sanborn, Griffiths, & Navarro, 2006)

# Gibbs sampling for the DPMM



- All the data are required at once (a batch procedure)
- Each stimulus is sequentially assigned to a cluster based on the assignments of all of the remaining stimuli
- Assignments are made probabilistically, using the full conditional distribution

# Particle filter for the DPMM



- Data are observed sequentially
- The posterior distribution at each point is approximated by a set of "particles"
- Particles are updated, and a fixed number of are carried over from trial to trial

# Approximating the posterior



For a single order, the Local MAP will produce a single partition

The Gibbs sampler and particle filter will approximate the exact DPMM distribution

# Order effects in human data

• The probabilistic model underlying the DPMM does not produce any order effects

– follows from exchangeability

- But... human data shows order effects (e.g., Medin & Bettger, 1994)
- Anderson and Matessa tested local MAP predictions about order effects in an unsupervised clustering experiment

(Anderson, 1990)

## Anderson and Matessa's Experiment

ront-Anchored Order	End-Anchored Order		
scadsporm	snadstirb		
scadstirm	snekstirb		
sneksporb	scadsporm		
snekstirb	sceksporm		
sneksporm	sneksporm		
snekstirm	snadsporm		
scadsporb	scedstirb		
scadstirb	scadstirb		

- Subjects were shown all sixteen stimuli that had four binary features
- Front-anchored ordered stimuli emphasized the first two features in the first eight trials; endanchored ordered emphasized the last two

## Anderson and Matessa's Experiment

Proportion that are Divided Along a Front-Anchored Feature

	Experimental Data	Local MAP	Particle Filter (1)	Particle Filter (100)	Gibbs Sampler
Front-Anchored Order	0.55	1.00	0.59	0.50	0.48
End-Anchored Order	0.30	0.00	0.38	0.50	0.49

# A "rational process model"

- A rational model clarifies a problem and serves as a benchmark for performance
- Using a psychologically plausible approximation can change a rational model into a "rational process model"
- Research in machine learning and statistics has produced useful approximations to statistical models which can be tested as general-purpose psychological heuristics

# Nonparametric Bayes and cognition

- Nonparametric Bayesian models are useful for answering questions about how much structure people should infer from data
- Many cognitive science questions take this form
  - how should we represent categories?
  - what features should we identify for objects?

# Learning the features of objects

- Most models of human cognition assume objects are represented in terms of abstract features
- What are the features of this object?



• What determines what features we identify?

(Austerweil & Griffiths, 2009)



![](_page_64_Picture_0.jpeg)

## Binary matrix factorization

$$P(x_{i,t} = 1 | \mathbf{Z}, \mathbf{Y}) = 1 - (1 - \lambda)^{\langle \mathbf{Z}_{i,:}, \mathbf{Y}_{:,t} \rangle} (1 - \epsilon)$$

![](_page_65_Figure_2.jpeg)

## Binary matrix factorization

$$P(x_{i,t} = 1 | \mathbf{Z}, \mathbf{Y}) = 1 - (1 - \lambda)^{\langle \mathbf{Z}_{i,:}, \mathbf{Y}_{:,t} \rangle} (1 - \epsilon)$$

![](_page_66_Figure_2.jpeg)

How should we infer the number of features?

## The nonparametric approach

Assume that the total number of features is unbounded, but only a finite number will be expressed in any finite dataset

![](_page_67_Figure_2.jpeg)

Use the Indian buffet process as a prior on Z (Griffiths & Ghahramani, 2006)

![](_page_68_Picture_0.jpeg)

![](_page_68_Figure_1.jpeg)

(Austerweil & Griffiths, 2009)

## Summary

- Sophisticated tools from Bayesian statistics can be valuable in developing probabilistic models of cognition...
- Monte Carlo methods provide a way to perform inference in probabilistic models, and a source of ideas for process models and experiments
- Nonparametric models help us tackle questions about how much structure to infer, with unbounded hypothesis spaces
- We look forward to seeing what you do with them!