

# CONTENTS

Preface	xv		
List of notation	xx		
BRMLTOOLBOX	xxi		
<b>I Inference in probabilistic models</b>			
<b>1 Probabilistic reasoning</b>	<b>3</b>		
1.1 Probability refresher			
1.1.1 Interpreting conditional probability			
1.1.2 Probability tables			
1.2 Probabilistic reasoning			
1.3 Prior, likelihood and posterior			
1.3.1 Two dice: what were the individual scores?			
1.4 Summary			
1.5 Code			
1.6 Exercises			
<b>2 Basic graph concepts</b>	<b>22</b>		
2.1 Graphs			
2.2 Numerically encoding graphs			
2.2.1 Edge list			
2.2.2 Adjacency matrix			
2.2.3 Clique matrix			
2.3 Summary			
2.4 Code			
2.5 Exercises			
<b>3 Belief networks</b>	<b>29</b>		
3.1 The benefits of structure			
3.1.1 Modelling independencies			
3.1.2 Reducing the burden of specification			
3.2 Uncertain and unreliable evidence			
3.2.1 Uncertain evidence			
3.2.2 Unreliable evidence			
3.3 Belief networks			
3.3.1 Conditional independence			
3.3.2 The impact of collisions			
3.3.3 Graphical path manipulations for independence			
3.3.4 d-separation			
3.3.5 Graphical and distributional in/dependence			
3.3.6 Markov equivalence in belief networks			
3.3.7 Belief networks have limited expressibility			
3.4 Causality			
3.4.1 Simpson's paradox			
3.4.2 The do-calculus			
3.4.3 Influence diagrams and the do-calculus			
3.5 Summary			
3.6 Code			
3.7 Exercises			
<b>4 Graphical models</b>	<b>58</b>		
4.1 Graphical models			
4.2 Markov networks			
4.2.1 Markov properties			
4.2.2 Markov random fields			
4.2.3 Hammersley–Clifford theorem			
4.2.4 Conditional independence using Markov networks			
4.2.5 Lattice models			
4.3 Chain graphical models			
4.4 Factor graphs			
4.4.1 Conditional independence in factor graphs			
4.5 Expressiveness of graphical models			
4.6 Summary			
4.7 Code			
4.8 Exercises			

<b>5</b>	<b>Efficient inference in trees</b>	<b>77</b>		
5.1	Marginal inference			
5.1.1	Variable elimination in a Markov chain and message passing			
5.1.2	The sum-product algorithm on factor graphs			
5.1.3	Dealing with evidence			
5.1.4	Computing the marginal likelihood			
5.1.5	The problem with loops			
5.2	Other forms of inference			
5.2.1	Max-product			
5.2.2	Finding the $N$ most probable states			
5.2.3	Most probable path and shortest path			
5.2.4	Mixed inference			
5.3	Inference in multiply connected graphs			
5.3.1	Bucket elimination			
5.3.2	Loop-cut conditioning			
5.4	Message passing for continuous distributions			
5.5	Summary			
5.6	Code			
5.7	Exercises			
<b>6</b>	<b>The junction tree algorithm</b>	<b>102</b>		
6.1	Clustering variables			
6.1.1	Reparameterisation			
6.2	Clique graphs			
6.2.1	Absorption			
6.2.2	Absorption schedule on clique trees			
6.3	Junction trees			
6.3.1	The running intersection property			
6.4	Constructing a junction tree for singly connected distributions			
6.4.1	Moralisation			
6.4.2	Forming the clique graph			
6.4.3	Forming a junction tree from a clique graph			
6.4.4	Assigning potentials to cliques			
6.5	Junction trees for multiply connected distributions			
6.5.1	Triangulation algorithms			
6.6	The junction tree algorithm			
6.6.1	Remarks on the JTA			
6.6.2	Computing the normalisation constant of a distribution			
6.6.3	The marginal likelihood			
6.6.4	Some small JTA examples			
6.6.5	Shafer–Shenoy propagation			
6.7	Finding the most likely state			
6.8	Reabsorption: converting a junction tree to a directed network			
6.9	The need for approximations			
6.9.1	Bounded width junction trees			
6.10	Summary			
6.11	Code			
6.12	Exercises			
<b>7</b>	<b>Making decisions</b>	<b>127</b>		
7.1	Expected utility			
7.1.1	Utility of money			
7.2	Decision trees			
7.3	Extending Bayesian networks for decisions			
7.3.1	Syntax of influence diagrams			
7.4	Solving influence diagrams			
7.4.1	Messages on an ID			
7.4.2	Using a junction tree			
7.5	Markov decision processes			
7.5.1	Maximising expected utility by message passing			
7.5.2	Bellman’s equation			
7.6	Temporally unbounded MDPs			
7.6.1	Value iteration			
7.6.2	Policy iteration			
7.6.3	A curse of dimensionality			
7.7	Variational inference and planning			
7.8	Financial matters			
7.8.1	Options pricing and expected utility			
7.8.2	Binomial options pricing model			
7.8.3	Optimal investment			
7.9	Further topics			
7.9.1	Partially observable MDPs			
7.9.2	Reinforcement learning			
7.10	Summary			
7.11	Code			
7.12	Exercises			

## II Learning in probabilistic models

### 8 Statistics for machine learning 165

- 8.1 Representing data
  - 8.1.1 Categorical
  - 8.1.2 Ordinal
  - 8.1.3 Numerical
- 8.2 Distributions
  - 8.2.1 The Kullback–Leibler divergence  $KL(q|p)$
  - 8.2.2 Entropy and information
- 8.3 Classical distributions
- 8.4 Multivariate Gaussian
  - 8.4.1 Completing the square
  - 8.4.2 Conditioning as system reversal
  - 8.4.3 Whitening and centring
- 8.5 Exponential family
  - 8.5.1 Conjugate priors
- 8.6 Learning distributions
- 8.7 Properties of maximum likelihood
  - 8.7.1 Training assuming the correct model class
  - 8.7.2 Training when the assumed model is incorrect
  - 8.7.3 Maximum likelihood and the empirical distribution
- 8.8 Learning a Gaussian
  - 8.8.1 Maximum likelihood training
  - 8.8.2 Bayesian inference of the mean and variance
  - 8.8.3 Gauss-gamma distribution
- 8.9 Summary
- 8.10 Code
- 8.11 Exercises

### 9 Learning as inference 199

- 9.1 Learning as inference
  - 9.1.1 Learning the bias of a coin
  - 9.1.2 Making decisions
  - 9.1.3 A continuum of parameters
  - 9.1.4 Decisions based on continuous intervals
- 9.2 Bayesian methods and ML-II
- 9.3 Maximum likelihood training of belief networks
- 9.4 Bayesian belief network training
  - 9.4.1 Global and local parameter independence

- 9.4.2 Learning binary variable tables using a Beta prior
- 9.4.3 Learning multivariate discrete tables using a Dirichlet prior
- 9.5 Structure learning
  - 9.5.1 PC algorithm
  - 9.5.2 Empirical independence
  - 9.5.3 Network scoring
  - 9.5.4 Chow–Liu trees
- 9.6 Maximum likelihood for undirected models
  - 9.6.1 The likelihood gradient
  - 9.6.2 General tabular clique potentials
  - 9.6.3 Decomposable Markov networks
  - 9.6.4 Exponential form potentials
  - 9.6.5 Conditional random fields
  - 9.6.6 Pseudo likelihood
  - 9.6.7 Learning the structure
- 9.7 Summary
- 9.8 Code
- 9.9 Exercises

### 10 Naive Bayes 243

- 10.1 Naive Bayes and conditional independence
- 10.2 Estimation using maximum likelihood
  - 10.2.1 Binary attributes
  - 10.2.2 Multi-state variables
  - 10.2.3 Text classification
- 10.3 Bayesian naive Bayes
- 10.4 Tree augmented naive Bayes
  - 10.4.1 Learning tree augmented naive Bayes networks
- 10.5 Summary
- 10.6 Code
- 10.7 Exercises

### 11 Learning with hidden variables 256

- 11.1 Hidden variables and missing data
  - 11.1.1 Why hidden/missing variables can complicate proceedings
  - 11.1.2 The missing at random assumption



Contents	ix
15.3 High-dimensional data <ul style="list-style-type: none"> <li>15.3.1 Eigen-decomposition for <math>N &lt; D</math></li> <li>15.3.2 PCA via singular value decomposition</li> </ul> 15.4 Latent semantic analysis <ul style="list-style-type: none"> <li>15.4.1 Information retrieval</li> </ul> 15.5 PCA with missing data <ul style="list-style-type: none"> <li>15.5.1 Finding the principal directions</li> <li>15.5.2 Collaborative filtering using PCA with missing data</li> </ul> 15.6 Matrix decomposition methods <ul style="list-style-type: none"> <li>15.6.1 Probabilistic latent semantic analysis</li> <li>15.6.2 Extensions and variations</li> <li>15.6.3 Applications of PLSA/NMF</li> </ul> 15.7 Kernel PCA           15.8 Canonical correlation analysis <ul style="list-style-type: none"> <li>15.8.1 SVD formulation</li> </ul> 15.9 Summary           15.10 Code           15.11 Exercises	17.4.1 Logistic regression           17.4.2 Beyond first-order gradient ascent           17.4.3 Avoiding overconfident classification           17.4.4 Multiple classes           17.4.5 The kernel trick for classification           17.5 Support vector machines <ul style="list-style-type: none"> <li>17.5.1 Maximum margin linear classifier</li> <li>17.5.2 Using kernels</li> <li>17.5.3 Performing the optimisation</li> <li>17.5.4 Probabilistic interpretation</li> </ul> 17.6 Soft zero-one loss for outlier robustness           17.7 Summary           17.8 Code           17.9 Exercises
<b>16 Supervised linear dimension reduction</b>	<b>359</b>
16.1 Supervised linear projections           16.2 Fisher's linear discriminant           16.3 Canonical variates <ul style="list-style-type: none"> <li>16.3.1 Dealing with the nullspace</li> </ul> 16.4 Summary           16.5 Code           16.6 Exercises	<b>367</b>
<b>17 Linear models</b>	<b>367</b>
17.1 Introduction: fitting a straight line           17.2 Linear parameter models for regression <ul style="list-style-type: none"> <li>17.2.1 Vector outputs</li> <li>17.2.2 Regularisation</li> <li>17.2.3 Radial basis functions</li> </ul> 17.3 The dual representation and kernels <ul style="list-style-type: none"> <li>17.3.1 Regression in the dual space</li> </ul> 17.4 Linear parameter models for classification	<b>392</b>
<b>18 Bayesian linear models</b>	<b>392</b>
18.1 Regression with additive Gaussian noise <ul style="list-style-type: none"> <li>18.1.1 Bayesian linear parameter models</li> <li>18.1.2 Determining hyperparameters: ML-II</li> <li>18.1.3 Learning the hyperparameters using EM</li> <li>18.1.4 Hyperparameter optimisation: using the gradient</li> <li>18.1.5 Validation likelihood</li> <li>18.1.6 Prediction and model averaging</li> <li>18.1.7 Sparse linear models</li> </ul> 18.2 Classification <ul style="list-style-type: none"> <li>18.2.1 Hyperparameter optimisation</li> <li>18.2.2 Laplace approximation</li> <li>18.2.3 Variational Gaussian approximation</li> <li>18.2.4 Local variational approximation</li> <li>18.2.5 Relevance vector machine for classification</li> <li>18.2.6 Multi-class case</li> </ul> 18.3 Summary           18.4 Code           18.5 Exercises	

<p><b>19 Gaussian processes</b></p> <p>19.1 Non-parametric prediction</p> <p>19.1.1 From parametric to non-parametric</p> <p>19.1.2 From Bayesian linear models to Gaussian processes</p> <p>19.1.3 A prior on functions</p> <p>19.2 Gaussian process prediction</p> <p>19.2.1 Regression with noisy training outputs</p> <p>19.3 Covariance functions</p> <p>19.3.1 Making new covariance functions from old</p> <p>19.3.2 Stationary covariance functions</p> <p>19.3.3 Non-stationary covariance functions</p> <p>19.4 Analysis of covariance functions</p> <p>19.4.1 Smoothness of the functions</p> <p>19.4.2 Mercer kernels</p> <p>19.4.3 Fourier analysis for stationary kernels</p> <p>19.5 Gaussian processes for classification</p> <p>19.5.1 Binary classification</p> <p>19.5.2 Laplace's approximation</p> <p>19.5.3 Hyperparameter optimisation</p> <p>19.5.4 Multiple classes</p> <p>19.6 Summary</p> <p>19.7 Code</p> <p>19.8 Exercises</p> <p><b>20 Mixture models</b></p> <p>20.1 Density estimation using mixtures</p> <p>20.2 Expectation maximisation for mixture models</p> <p>20.2.1 Unconstrained discrete tables</p> <p>20.2.2 Mixture of product of Bernoulli distributions</p> <p>20.3 The Gaussian mixture model</p> <p>20.3.1 EM algorithm</p> <p>20.3.2 Practical issues</p> <p>20.3.3 Classification using Gaussian mixture models</p> <p>20.3.4 The Parzen estimator</p> <p>20.3.5 K-means</p>	<p><b>412</b></p> <p><b>432</b></p>	<p>20.3.6 Bayesian mixture models</p> <p>20.3.7 Semi-supervised learning</p> <p>20.4 Mixture of experts</p> <p>20.5 Indicator models</p> <p>20.5.1 Joint indicator approach: factorised prior</p> <p>20.5.2 Poly prior</p> <p>20.6 Mixed membership models</p> <p>20.6.1 Latent Dirichlet allocation</p> <p>20.6.2 Graph-based representations of data</p> <p>20.6.3 Dyadic data</p> <p>20.6.4 Monadic data</p> <p>20.6.5 Cliques and adjacency matrices for monadic binary data</p> <p>20.7 Summary</p> <p>20.8 Code</p> <p>20.9 Exercises</p> <p><b>21 Latent linear models</b></p> <p>21.1 Factor analysis</p> <p>21.1.1 Finding the optimal bias</p> <p>21.2 Factor analysis: maximum likelihood</p> <p>21.2.1 Eigen-approach likelihood optimisation</p> <p>21.2.2 Expectation maximisation</p> <p>21.3 Interlude: modelling faces</p> <p>21.4 Probabilistic principal components analysis</p> <p>21.5 Canonical correlation analysis and factor analysis</p> <p>21.6 Independent components analysis</p> <p>21.7 Summary</p> <p>21.8 Code</p> <p>21.9 Exercises</p> <p><b>22 Latent ability models</b></p> <p>22.1 The Rasch model</p> <p>22.1.1 Maximum likelihood training</p> <p>22.1.2 Bayesian Rasch models</p> <p>22.2 Competition models</p> <p>22.2.1 Bradley–Terry–Luce model</p> <p>22.2.2 Elo ranking model</p> <p>22.2.3 Glicko and TrueSkill</p>	<p><b>462</b></p> <p><b>479</b></p>
--	-------------------------------------	---	-------------------------------------

Contents	xi
22.3 Summary	
22.4 Code	
22.5 Exercises	
<b>IV Dynamical models</b>	
<b>23 Discrete-state Markov models</b>	<b>489</b>
23.1 Markov models	
23.1.1 Equilibrium and stationary distribution of a Markov chain	
23.1.2 Fitting Markov models	
23.1.3 Mixture of Markov models	
23.2 Hidden Markov models	
23.2.1 The classical inference problems	
23.2.2 Filtering $p(h_t v_{1:t})$	
23.2.3 Parallel smoothing $p(h_t v_{1:T})$	
23.2.4 Correction smoothing	
23.2.5 Sampling from $p(h_{1:T} v_{1:T})$	
23.2.6 Most likely joint state	
23.2.7 Prediction	
23.2.8 Self-localisation and kidnapped robots	
23.2.9 Natural language models	
23.3 Learning HMMs	
23.3.1 EM algorithm	
23.3.2 Mixture emission	
23.3.3 The HMM-GMM	
23.3.4 Discriminative training	
23.4 Related models	
23.4.1 Explicit duration model	
23.4.2 Input–output HMM	
23.4.3 Linear chain CRFs	
23.4.4 Dynamic Bayesian networks	
23.5 Applications	
23.5.1 Object tracking	
23.5.2 Automatic speech recognition	
23.5.3 Bioinformatics	
23.5.4 Part-of-speech tagging	
23.6 Summary	
23.7 Code	
23.8 Exercises	
<b>24 Continuous-state Markov models</b>	<b>520</b>
24.1 Observed linear dynamical systems	
24.1.1 Stationary distribution with noise	
24.2 Auto-regressive models	
24.2.1 Training an AR model	
24.2.2 AR model as an OLDS	
24.2.3 Time-varying AR model	
24.2.4 Time-varying variance AR models	
24.3 Latent linear dynamical systems	
24.4 Inference	
24.4.1 Filtering	
24.4.2 Smoothing: Rauch–Tung–Striebel correction method	
24.4.3 The likelihood	
24.4.4 Most likely state	
24.4.5 Time independence and Riccati equations	
24.5 Learning linear dynamical systems	
24.5.1 Identifiability issues	
24.5.2 EM algorithm	
24.5.3 Subspace methods	
24.5.4 Structured LDSs	
24.5.5 Bayesian LDSs	
24.6 Switching auto-regressive models	
24.6.1 Inference	
24.6.2 Maximum likelihood learning using EM	
24.7 Summary	
24.8 Code	
24.9 Exercises	
<b>25 Switching linear dynamical systems</b>	<b>547</b>
25.1 Introduction	
25.2 The switching LDS	
25.2.1 Exact inference is computationally intractable	
25.3 Gaussian sum filtering	
25.3.1 Continuous filtering	
25.3.2 Discrete filtering	
25.3.3 The likelihood $p(\mathbf{v}_{1:T})$	
25.3.4 Collapsing Gaussians	
25.3.5 Relation to other methods	
25.4 Gaussian sum smoothing	
25.4.1 Continuous smoothing	
25.4.2 Discrete smoothing	
25.4.3 Collapsing the mixture	
25.4.4 Using mixtures in smoothing	
25.4.5 Relation to other methods	

25.5	Reset models		27.4.1	Markov chains	
25.5.1	A Poisson reset model		27.4.2	Metropolis–Hastings sampling	
25.5.2	Reset-HMM-LDS		27.5	Auxiliary variable methods	
25.6	Summary		27.5.1	Hybrid Monte Carlo (HMC)	
25.7	Code		27.5.2	Swendson–Wang (SW)	
25.8	Exercises		27.5.3	Slice sampling	
<b>26</b>	<b>Distributed computation</b>	<b>568</b>	27.6	Importance sampling	
26.1	Introduction		27.6.1	Sequential importance sampling	
26.2	Stochastic Hopfield networks		27.6.2	Particle filtering as an approximate forward pass	
26.3	Learning sequences		27.7	Summary	
26.3.1	A single sequence		27.8	Code	
26.3.2	Multiple sequences		27.9	Exercises	
26.3.3	Boolean networks		<b>28</b>	<b>Deterministic approximate inference</b>	<b>617</b>
26.3.4	Sequence disambiguation		28.1	Introduction	
26.4	Tractable continuous latent variable models		28.2	The Laplace approximation	
26.4.1	Deterministic latent variables		28.3	Properties of Kullback–Leibler variational inference	
26.4.2	An augmented Hopfield network		28.3.1	Bounding the normalisation constant	
26.5	Neural models		28.3.2	Bounding the marginal likelihood	
26.5.1	Stochastically spiking neurons		28.3.3	Bounding marginal quantities	
26.5.2	Hopfield membrane potential		28.3.4	Gaussian approximations using KL divergence	
26.5.3	Dynamic synapses		28.3.5	Marginal and moment matching properties of minimising $KL(p q)$	
26.5.4	Leaky integrate and fire models		28.4	Variational bounding using $KL(q p)$	
26.6	Summary		28.4.1	Pairwise Markov random field	
26.7	Code		28.4.2	General mean-field equations	
26.8	Exercises		28.4.3	Asynchronous updating guarantees approximation improvement	
<b>V</b>	<b>Approximate inference</b>		28.4.4	Structured variational approximation	
<b>27</b>	<b>Sampling</b>	<b>587</b>	28.5	Local and KL variational approximations	
27.1	Introduction		28.5.1	Local approximation	
27.1.1	Univariate sampling		28.5.2	KL variational approximation	
27.1.2	Rejection sampling		28.6	Mutual information maximisation: a KL variational approach	
27.1.3	Multivariate sampling				
27.2	Ancestral sampling				
27.2.1	Dealing with evidence				
27.2.2	Perfect sampling for a Markov network				
27.3	Gibbs sampling				
27.3.1	Gibbs sampling as a Markov chain				
27.3.2	Structured Gibbs sampling				
27.3.3	Remarks				
27.4	Markov chain Monte Carlo (MCMC)				



## Contents

xiii

28.6.1	The information maximisation algorithm	28.12	Code	
28.6.2	Linear Gaussian decoder	28.13	Exercises	
28.7	Loopy belief propagation	<b>Appendix A: Background mathematics</b>		<b>655</b>
28.7.1	Classical BP on an undirected graph	A.1	Linear algebra	
28.7.2	Loopy BP as a variational procedure	A.2	Multivariate calculus	
28.8	Expectation propagation	A.3	Inequalities	
28.9	MAP for Markov networks	A.4	Optimisation	
28.9.1	Pairwise Markov networks	A.5	Multivariate optimisation	
28.9.2	Attractive binary Markov networks	A.6	Constrained optimisation using Lagrange multipliers	
28.9.3	Potts model	References		675
28.10	Further reading	Index		689
28.11	Summary			
			<i>Colour plate section between pp. 360 and 361</i>	