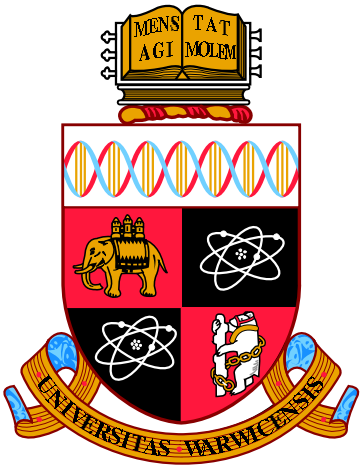


Bayesian Complementary Clustering, MCMC and Anglo-Saxon Placenames

by

Giacomo Zanella

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

Department of Statistics

September 2015

THE UNIVERSITY OF WARWICK

Contents

List of Tables

List of Figures

Acknowledgments

I would like to thank my supervisor, Wilfrid Kendall, for his generous availability and constant support, for having put my professional growth ahead of his academic interests and for his kindness and esteem towards me.

I am also grateful to many members of the Department of Statistics at the University of Warwick: the countless number of colleagues who have been available to discuss the research of this thesis, the various friends, especially my fellow PhD students, that contributed to make my time in the department enjoyable, and finally the administrative staff for their prompt support with organizational and bureaucratic issues.

I acknowledge John Blair for collaboration and arranging supply of data and the Engineering and Physical Sciences Research Council for funding through the CRiSM grant EP/D002060/1.

Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself and has not been submitted in any previous application for any degree.

The work presented was carried out by the author except where explicitly indicated otherwise by references. Large part of the work in this thesis (excluding Chapter 5) has formed the following paper:

> Zanella, G. (2015) Random Partition Models and complementary clustering of Anglo-Saxon place-names. *The Annals of Applied Statistics*, 9(4), 1792-1822.

Abstract

Common cluster models for multi-type point processes model the aggregation of points of the same type. In complete contrast, in the study of Anglo-Saxon settlements it is hypothesized that administrative clusters involving complementary names tend to appear. We investigate the evidence for such a hypothesis by developing a Bayesian Random Partition Model based on clusters formed by points of different types (complementary clustering).

As a result we obtain an intractable posterior distribution on the space of matchings contained in a k-partite hypergraph. We use the Metropolis-Hastings (MH) algorithm to sample from such a distribution. We consider the problem of what is the optimal, informed MH proposal distribution given a fixed set of allowed moves. To answer such a question we define the notion of balanced proposals and we prove that, under some assumptions, such proposals are maximal in the Peskun sense. Using such ideas we obtain substantial mixing improvements compared to other choices found in the literature. Simulated Tempering techniques can be used to overcome multimodality and a multiple proposal scheme is developed to allow for parallel programming. Finally, we discuss results arising from the careful use of convergence diagnostic techniques.

This allows us to study a dataset including locations and placenames of 1316 Anglo-Saxon settlements dated around 750-850 AD. Without strong prior knowledge, the model allows for explicit estimation of the number of clusters, the average intra-cluster dispersion and the level of interaction among placenames. The results support the hypothesis of organization of settlements into administrative clusters based on complementary names.

Chapter 1

Introduction

The starting point of this work is a dataset supplied by John Blair of Queen's College Oxford, Professor of History at the University of Oxford. The dataset consists of the locations and placenames of 1316 Anglo-Saxon settlements dated approximately around 750-850 AD. Professor Blair's hypothesis is that those settlements were organized into administrative districts involving settlements with complementary roles and that such roles were indicated by the different placenames (see Section 2.1).

If present, such a phenomenon would induce clusters of closely located settlements with *different* placenames (complementary clustering). On the contrary, typical cluster models for marked point processes deal with the aggregation of elements of the *same* type (or with similar features). In the latter case, one can treat the marks/types as an additional dimension and perform clustering in a higher dimensional space (e.g. two spatial dimensions plus one marks dimension), while in our scenario the spatial dimensions and the marks/types dimension have different roles. This has both modeling and computational implications.

We develop a Bayesian Random Partition Model (RPM) to study complementary clustering phenomena. RPMs provide a general, flexible framework and allow to make explicit inferences on the unobserved partition. Because of the complementarity requirement, the posterior sample space uses the space of matchings contained in a k-partite hypergraph. We devote much attention to computational aspects, both from the theoretical and practical point of view.

The model and algorithm we develop allow the study of the Anglo-Saxon settlements dataset. Without strong prior knowledge, the model allows for explicit estimation of the number of clusters, the average intra-cluster dispersion and the level of interaction among placenames. The results support Professor Blair's hypothesis

of organization of settlements into administrative clusters based on complementary names and provide additional information and insight into such a phenomenon.

Thesis contributions

This work arises out of a specific applied problem presented to us by distinguished historians. In trying to provide satisfactory answers to their questions, we develop novel theory and methodology having wider implications, both from the modeling and especially from the computational point of view.

First we develop a Bayesian Random Partition Model tailored to study complementary clustering scenarios (see Chapter 3). Such a model may have applications to other scenarios (such as ecology, see e.g. Chapter 8) where clusters of distinct elements occur, rather than clusters of similar objects. In particular two prior distributions (see Sections 3.5.1 and 3.7.1) are proposed to model clustering scenarios with many small clusters with a bounded number of points each. We describe these models in terms of matchings contained in hypergraphs (Chapter 4), thus relating computational tasks to the Complexity Theory literature.

Secondly we study Markov chain Monte Carlo (MCMC) algorithms in discrete spaces such as matchings contained in hypergraphs, in particular considering the problem of designing informed Metropolis-Hastings (MH) proposals in such spaces. Under some assumptions (satisfied by our model) we derive the class of asymptotically optimal proposal distributions (Section 5.4). To prove such optimality we use the so-called Peskun ordering (Peskun, 1973), extending it to cases involving a constant in the off-diagonal comparison (see Theorem 4). These results motivate the introduction of the apparently new notion of *balanced proposals* (see Chapter 5). Such a notion provides a general and coherent way to incorporate local information into MH proposal distributions and has the potential to extend the benefits of gradient-based MCMC algorithms to discrete settings (see Section 8.3).

Finally we describe practical implementations of the ideas in Chapter 5 for the applied problem under consideration. We discuss various practical issues (tempering, parallel computing, convergence diagnostic) and we study the Anglo-Saxon settlements dataset, managing to provide informative answers to the historians' questions (Chapter 7).

Organization of the thesis

Chapter 2 is divided in two parts. The first part describes the historical problem, the questions of historical interest and the Anglo-Saxon settlements dataset. The

second part contains a preliminary analysis of the dataset with traditional spatial statistics tools such as K-cross functions. This exploratory analysis suggests that there is some attractive interaction between settlements with different placenames and motivates further analysis.

In Chapter 3 we define a Bayesian RPM for complementary clustering scenarios. Some attention is devoted to the choice of prior distribution for the partition. This is necessary because the commonly used prior distributions, such as the Dirichlet Process prior, are not appropriate for this context, for example because they model situations with few big clusters, while our scenario leads to many small clusters.

The posterior distribution induced by the model of Chapter 3 is intractable. To address this more precisely, in Chapter 4 we study the computational complexity of tasks associated with such a posterior distribution by appealing to known results from the Complexity Theory literature. To do so we exploit the fact that the posterior distribution is proportional to the weight of a corresponding matching contained in a weighted k-partite hypergraph. We thus link the problems of sampling from the posterior and finding the posterior mode (or the maximum likelihood estimator) to the more classical problems of Data Association and Optimal Assignment.

Given the results discussed in Chapter 4 we need to use approximate methods to perform inferences on the posterior distribution of interest. Motivated by the results recalled in Section 4.2.4, we use Monte Carlo methods and in particular MCMC algorithms. In Chapter 5 we introduce MCMC methods, in particular the Metropolis-Hastings (MH) algorithm and some theoretical notions related to measuring MCMC efficiency. Then we consider the problem of choosing the optimal proposal distribution given a fixed set of allowed moves (in a fairly general framework). To solve such a problem we introduce the notions of balanced proposals and we show that under some assumptions such a family of proposals is optimal in terms of (asymptotic) Peskun ordering (see Theorems 4, 5 and 6).

In Chapter 6 we return to the applied problem and we describe the actual MCMC algorithm we use to obtain approximate samples from the posterior of interest. To do so we deal with sampling matchings contained in hypergraphs (a problem often encountered, for example, in Data Association problems) and we use ideas from Chapter 5 to speed up such sampling process. We consider convergence diagnostic issues, we explore the use of Simulated Tempering to overcome multimodality and we develop a multiple proposal scheme to allow for parallel computation.

In Chapter 7 we analyze the Anglo-Saxon settlements dataset, fitting the Bayesian RPM of Chapter 3 with the algorithm of Chapter 6. The results sup-

port the hypothesis of settlements being organized into administrative clusters and provide explicit inferences of various quantities of historical interest.

Finally in Chapter 8 we summarize the results and discuss possible directions of future research. Supplementary material, available at `https://sites.google.com/site/gzanellawebpage/compclust_supp_f.zip`, includes the Anglo-Saxon settlements dataset and the R codes used to perform the data analysis.

Chapter 2

Historical problem and preliminary analysis

2.1 The historical question under consideration

As already mentioned in Chapter 1, the starting point of this work is the Anglo-Saxon settlements dataset provided by Professor John Blair, which contains the locations and placenames of more than a thousand settlements. In the dataset there are 20 different kinds of placenames in total. Placenames form an important source of information regarding the Anglo-Saxon civilization and are intensively studied by the historical community (see for example Gelling & Cole, 2000 and Jones & Semple, 2012).

In particular, the placenames included in this dataset are often described as *functional* placenames, as they were probably used to indicate specific functions or features of their corresponding settlements. For example *Burton* is thought to label fortified settlements having a military role, *Charlton* the settlements of the peasants and *Drayton* the settlements dedicated to portage.

Moreover, historians expect the settlements in this dataset (especially those having one of the placenames underlined in Table 2.1) to have been formed approximately at the same time and in the same context (specifically, royal administration in the period c.750-850). This suggests that there could be some coherence in the distribution of such placenames. In particular Professor Blair's hypothesis is that those settlements were not independent units but rather that they were organized into administrative clusters (or districts) where placenames were used to indicate the role of each settlement within the district. According to this hypothesis such clusters would tend to involve a variety of complementary placenames in each of

them. For example Figure 2.1 indicates a plausible administrative cluster made of four settlements, with, for example, a settlement dedicated to military functions (*Burton*) and one dedicated to agriculture (*Carlton*).

Figure 2.1: A cluster of four Anglo-Saxon settlements (circled and highlighted in green) in the region of Great Glen (written in short as G^t.Glen).

The objective of our statistical approach to the study of settlements names and geographical locations is to address the following questions: is there statistical support for Blair's hypothesis? What is the typical distance between settlements in the same cluster? How many settlements are clustered together and how many are singletons? Which placenames tend to cluster together? Can we provide a list of those clusters that are more strongly supported by the analysis?

Our intention is to provide a useful contribution to historical research on this topic based on a quantitative approach, bearing in mind the scarcity of textual evidence regarding the Anglo-Saxon period. Since there is much uncertainty and controversy regarding the meaning of placenames, even the apparently obvious ones, we should try to be fairly neutral from the historical point of view, avoiding strong assumptions on the functions of placenames and relationships among them. This will help our statistical analysis to be a genuine contribution to the ongoing historical debate on this topic.

We note that there has already been statistical work related to Anglo-Saxon placenames. In particular see the work of Keith Briggs on this topic (see `http://keithbriggs.info/place-names.html` for a full list). Nevertheless both the historical questions considered and the statistical methodologies used are substantially different from ours.

2.2 The AngloSaxon settlements dataset

We now describe the Anglo-Saxon settlements dataset supplied by Prof. John Blair and the data cleaning operations that we carried out. The dataset (fully available at `https://sites.google.com/site/gzanellawebpage/compclust_supp_f.zip`)

is made of 20 different groups, each of which contains the list of settlements having one of the 20 placenames (see Table 2.1). The historians involved in the project

Placenames	total number	# of settlements with less precise location	# of couples (as classified by historians)	# of couples (as classified by proximity)
Aston/Easton	90	0	1	8
Bolton	17	1	1	0
Burh-Stall	29	2	1	0
Burton	108	2	1	7
Centres	46	0	0	0
Charlton/Charlcot	98	3	7	1
Chesterton	9	0	0	0
Claeg	84	13	0	5
Draycot/Drayton	55	1	0	2
Eaton	33	1	1	5
Kingston	71	1	1	1
Knighton	26	1	0	0
Newbold	34	3	1	0
Newton	191	5	4	5
Norton	74	1	8	1
Stratton	37	0	5	0
Sutton	101	2	4	5
Tot	77	17	1	1
Walton/Walcot	51	4	1	0
Weston	85	3	3	2
Total	1316	60	40	43

Table 2.1: Number of settlements in the Anglo-Saxon placenames location dataset supplied by Prof. Blair. The historians expect the clustering behavior mainly to involve 13 of those placenames (underlined and emboldened in this table). Settlements with less precise locations (third column) are settlements whose location is given with 1 km accuracy, rather than 100 m, or having a more uncertain location (see Section 2.2). The term "couples" (last two columns) refers to multiple records of the same settlements (see Section 2.2.1 for discussion). The "total number" column refers to the count after merging the couples classifieds by historians.

expect the clustering behavior to involve 13 of those placenames in particular, indicated in Table 2.1. We refer to the settlements relative to those 13 placenames as the *reduced dataset*, and to all the settlements recorded as the *full dataset*. We will perform statistical analyses on both datasets.

For each settlement the following variables are given: County, place, Parish or Township, grid ref and date of first evidence (see Table 2.2).

The locations are expressed through the Ordnance Survey (OS) National Grid reference system. A set of OS National Grid coordinates, like $SU230870$,

COUNTY	PLACE	PARISH OR TOWNSHIP	GRID REF	DATE OF FIRST EVIDENCE
BRK	Bourton	Bourton	SU 230870	c. 1200
BUC	Bierton	Bierton with Broughton	SP 836152	DB
BUC	Bourton	Buckingham	SP 710333	DB
CHE	Burton	Burton (T)	SJ 509639	DB
CHE	Burton	Burton (T)	SJ 317743	1152
CHE	Buerton	Buerton (T)	SJ 682433	DB

Table 2.2: Exemplary data available for the first 6 settlement with the name *Burton*. The acronym DB stands for Domesday Book, compiled in 1086.

identify a 100m $\times$ 100m square on a grid covering Great Britain. Some locations have just 2 letters and 4 digits (e.g. $SU2387$) and they identify a 1km $\times$ 1km square, and some have a letter c in front of them (e.g. $c.SU2387$) to indicate that the location is less accurate (see Table 2.1 for amounts of these).

2.2.1 Data cleaning and data assumptions

Our analysis is concerned with placenames (variable "place") and geographical locations (variable "Grid reference"). By considering placenames as marks attached to points, we model our data as the realization of a k-type point process (also called k-variate point process), where k is the number of different placenames available (see Baddeley, 2010 or Section 2.3 for more details on point processes). We convert the data to a k-type point process form as described below. This data cleaning process entails historical assumptions on the dataset and thus we have been guided by the judgment of the subject-specific historians involved in this project in doing so. Figure 2.2 shows the resulting k-type point pattern for the full dataset, while Figure 2.3 shows that only for the reduced dataset.

Placenames: we express the variable "place" as a categorical variable with k possible values (i.e. k types). For the full dataset k equals 20, while for the reduced dataset k equals 13. By describing the variable "place" as a categorical variable we ignore minor variations in placenames. For example we consider all the settlements of Table 2.2 as having placename *Burton*: their actual recorded placenames vary amongst *Burton, Bourton, Bierton, Buerton.*

Four groups (out of 20) are made up of two subgroups each with similar placenames: $Aston/Easton$, $Charlton/Charlcot$, $Drayton/Draycot$ and $Walton/Walcot$. We consider placenames within such subgroups to be the same, for example $Charlton$ and $Charlcot$ are treated as the same placename.

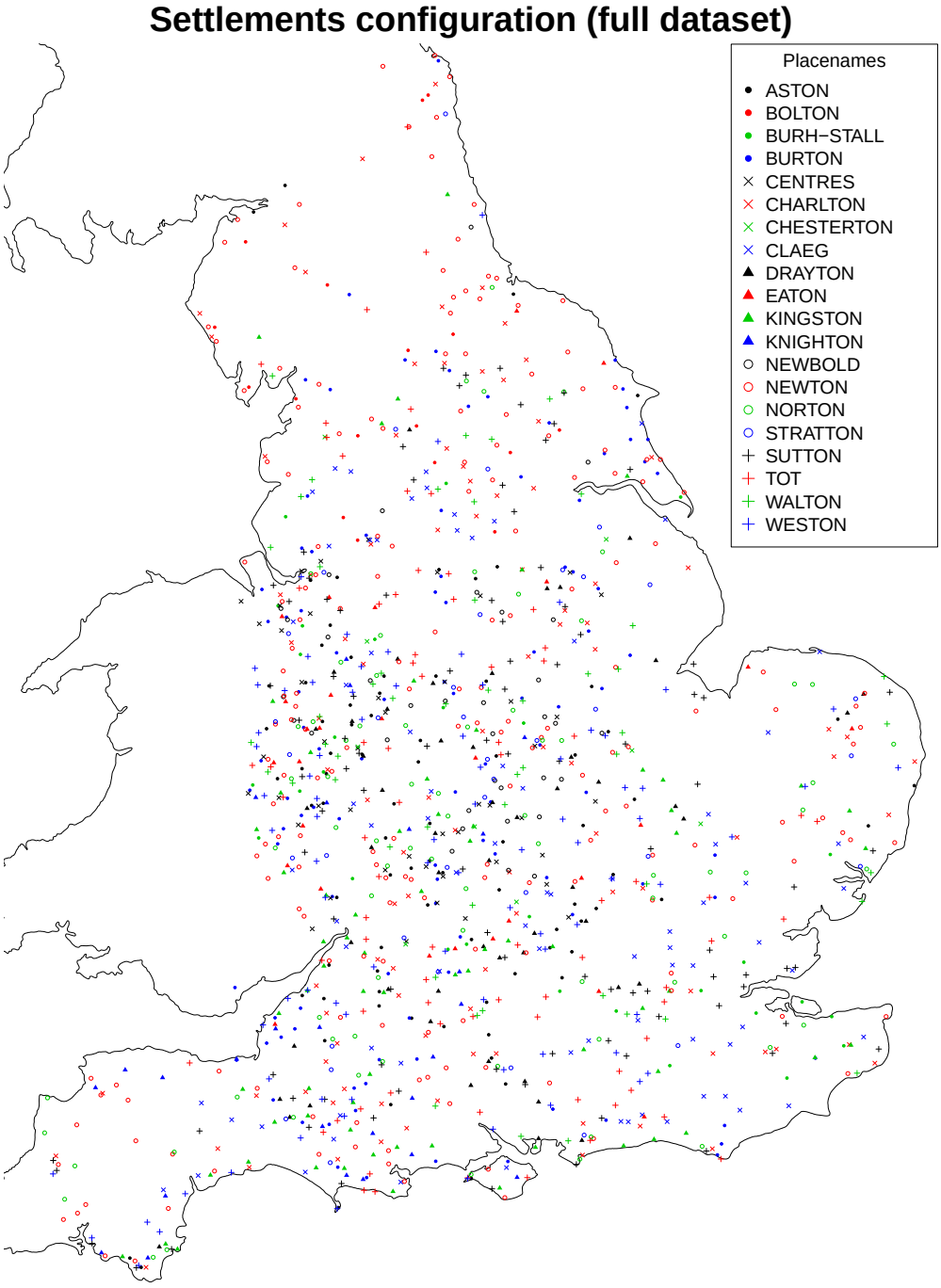


Figure 2.2: Plot of the full dataset together with UK coastline. Each point represents a settlement. Different symbols represent different placenames. The "couples" (see Table 2.1 and the paragraph "Multiple records" above) have already been merged and the resulting number of settlements is 1273.

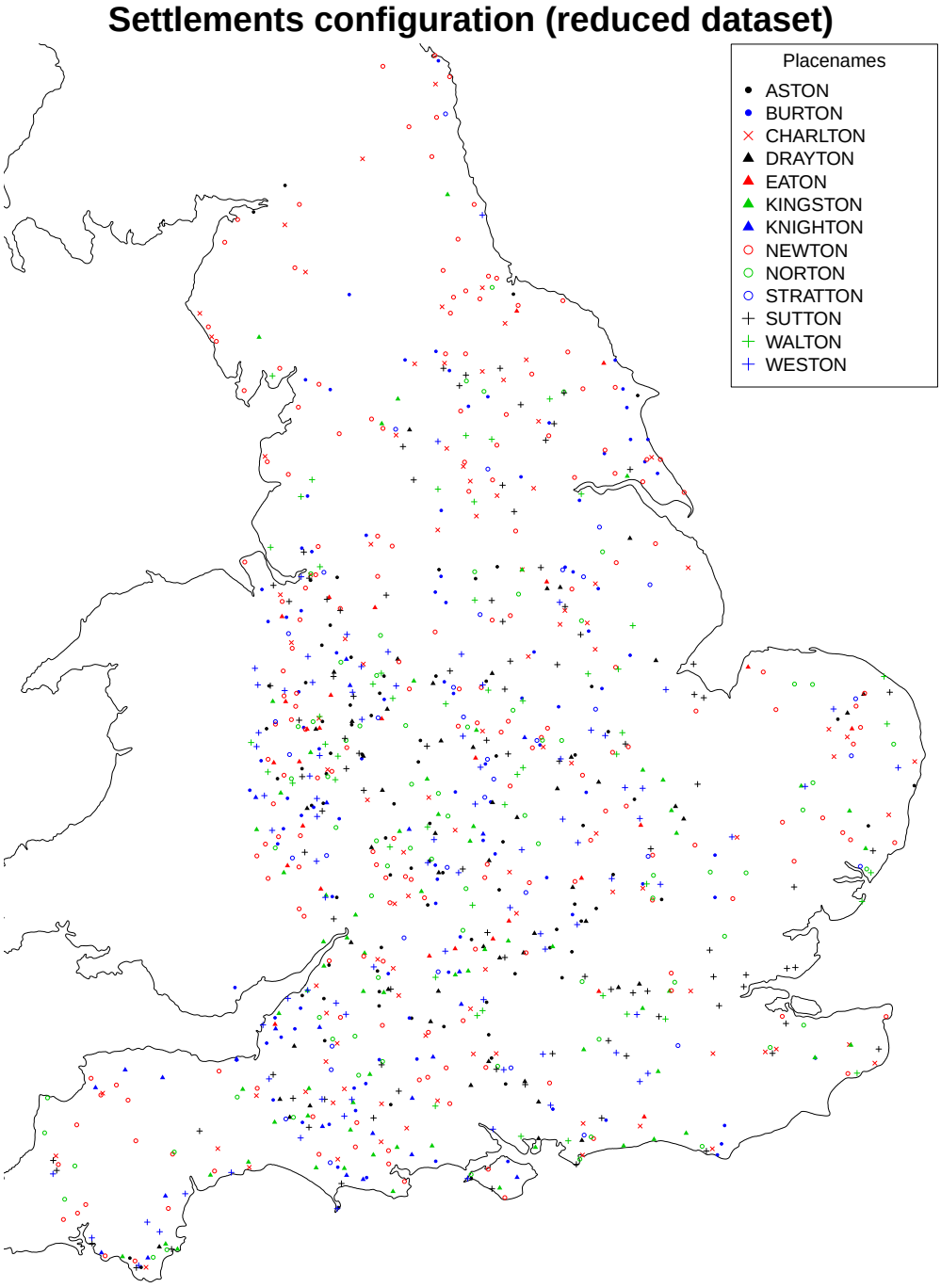


Figure 2.3: Plot of the reduced dataset with UK coastline. Each point represents a settlement. Different symbols represent different placenames. The reduced dataset corresponds to the 13 placenames that historians expect to be more involved in the clustering behavior. The "couples" (see Table 2.1) have already been merged.

Figure 2.4 provides a box-plot representation of the observed x and y coordinates divided by placenames. Such a plot provides a crude representation of the variability in the distribution of settlements across different placenames. Note that, especially for the reduced dataset (represented with black solid lines in Figure 2.4), there is no clear suggestion of grouping. In fact, for most couples of placenames, say A and B, the area with a high density of settlements of type A has a consistent overlap with that of settlements of type B. Therefore, it does not seem appropriate to divide placenames into groups and to analyze the corresponding settlements separately.

We will not model the heterogeneity in placenames distribution explicitly in the cluster analysis (see Section 8.2 for discussion of possible improvements). Note, however, that we do model such heterogeneity in the null-hypothesis testing of Section 2.3.2 and the sanity check of Section 7.1.

Locations: we convert OS National Grid coordinates to two-dimensional Euclidean coordinates and each settlement is assumed to be located at the center of the corresponding OS National Grid square.

Multiple records: it is sometimes indicated in the original dataset that some couples (or triples) of settlements, with the same placename and very close locations, have to be considered as multiple records of the same settlement. We replaced such couples (or triples) of settlements with a single settlement located at their midpoint. Moreover there are some other pairs of records having very close locations and the same placename (see Table 2.1 for amounts). It is primarily a matter of historical interpretation whether these couples have to be considered as single settlements. We performed the analysis under both hypotheses (keeping them separated or merged) without seeing significant changes in the results. The analysis presented here is made with those settlements merged together (3 km is the threshold distance below which we identify two records of settlements with the same placename).

Observation region W: a point process realization consists of point locations *and* of the region W where the points have been observed. Indeed both the K-cross function analysis of Section 2.3.1 and the Bayesian model of Chapter 3 will use information about W. In our case we define W as Great Britain (coastline obtained from the *mapdata* *R* package Becker *et al.* , 2013) cropping the region where the point process intensity g falls below a certain threshold, approximately at the borders between England-Scotland and England-Wales. We also added a small buffer zone of 3 km around the region to include the few points that were falling outside the region (e.g. because the coastline has moved or because the location was inaccurate). See Figure 3.1 for a plot of the region.

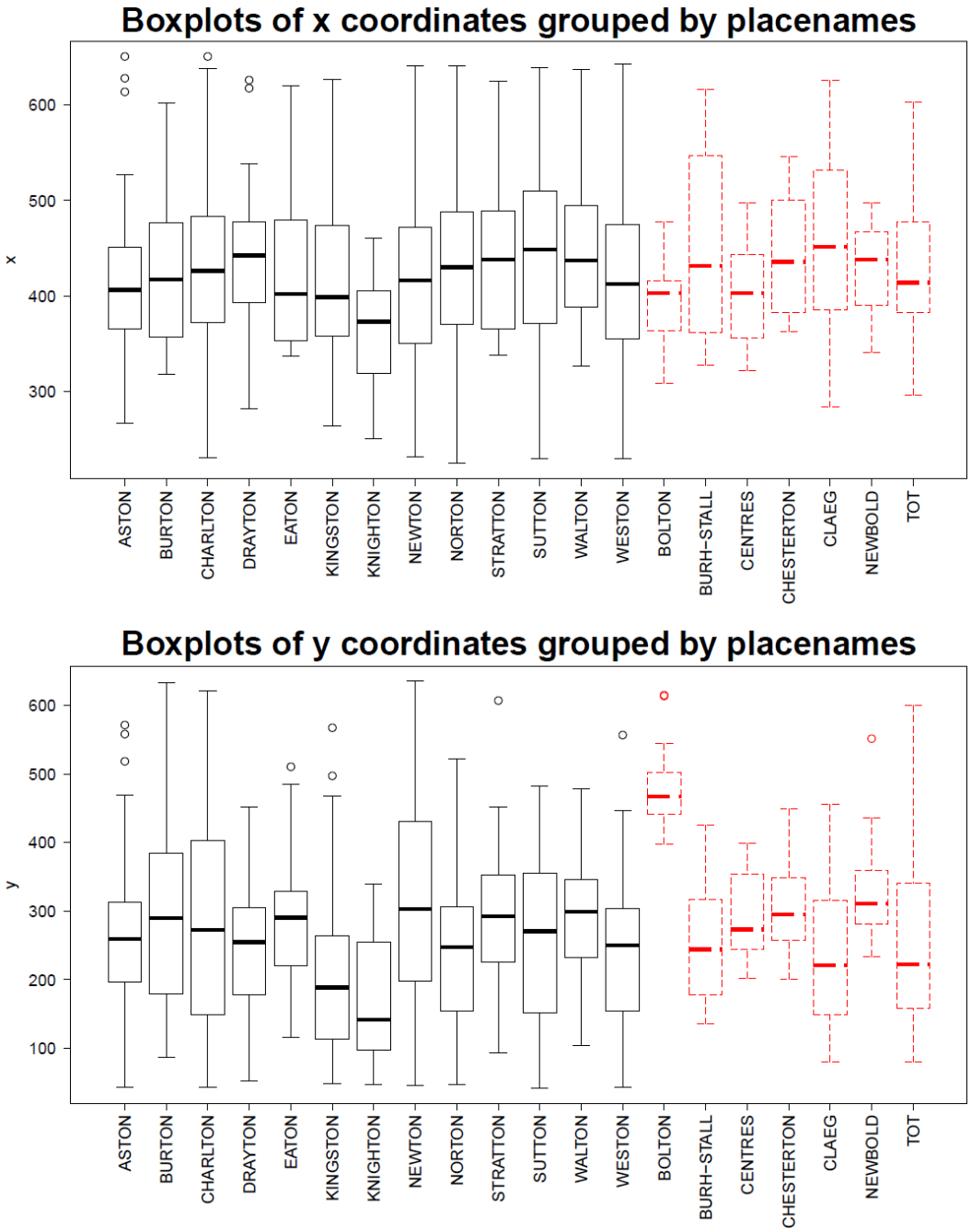


Figure 2.4: Boxplots of x and y coordinates (expressed in kilometers with reference axes given by the national OS grid system) divided by placenames. These plots provide an indication of the heterogeneity among the locations of settlements with different placenames. The reduced dataset is on the left (black solid line) and the rest is on the right (red dashed line).

2.3 Preliminary analysis of the Anglo-Saxon settlements dataset

We now perform preliminary analysis on the resulting point pattern using classic Spatial Statistic tools. As mentioned before, we can model the data as the realization of a *spatial point process*. Spatial point processes are random configuration of points in the plane, or more precisely stochastic processes whose realizations consist of a finite (or countably infinite) set of points contained in some window $W \subseteq \mathbb{R}^2$. For simplicity we assume the number of points to be finite and all the points to have distinct locations (i.e. no two points are allowed to be one on top of the other). We denote a realization of a spatial point process by $\mathbf{x} = \{x_1, \ldots, x_{n(\mathbf{x})}\}$. Note that the set of points is unordered and that the number of points is not fixed.

The most important example of point process is the *Poisson point process*. A point process $\mathbf{x}$ is a Poisson point process driven by an intensity measure $\Lambda(\cdot)$, with $\Lambda(\cdot)$ being a measure on $W \subseteq \mathbb{R}^2$, if for any measurable $A \subseteq W$ the number of points of $\mathbf{x}$ belonging to A is a random variable which follows a Poisson distribution with mean $\Lambda(A)$. The importance of the Poisson point process is due to the fact that it is the only possible model where points are completely independent of each other, meaning that, given A_1 and A_2 disjoint subsets of W, the number of points in A_1 and A_2 are independent random variables. See Daley & Vere-Jones (2002) and Daley & Vere-Jones (2008) for a rigorous introduction to point processes in terms of random discrete measures.

Another important class of models is the class of *cluster point processes*. Such point processes are built in two stages: first a set of "parents" (or "centers") is generated and then, for each center, a "daughter" point process (or "cluster") is generated. The cluster point process is then defined as the superposition of all the daughter point processes. Cluster point processes are typically used to model scenarios where the clusters consist of the aggregation of closely located points. See Chiu *et al.* (2013) and Isham (2010) for more details on models for spatial point processes.

2.3.1 K-cross function analysis

Second moment functions are a useful tool to investigate inter-point interaction (e.g. Chiu *et al.* , 2013). In particular, given a multi-type point pattern, bivariate (or cross-type) K-functions provide good summary functions of the interaction across points of different types. The bivariate K-function $K_{ij}(r)$ is the expected number of points of type j closer than r to a typical point of type i, divided by the intensity λ_j

of the type j sub-pattern of points $\mathbf{x}_j$ (e.g. Baddeley, 2010, Sec. 6). For testing and displaying purposes we define a single summary function, a multi-type K-function $K_{cross}(r)$, as the weighted average of $K_{ij}(r)$ for $i \neq j$, where the weights are the product of the intensities $\lambda_i \lambda_j$.

Classical K-functions, however, rely strongly on the assumption that the point pattern is stationary, which is not appropriate for our dataset. Therefore we use the inhomogeneous version of the K-functions, where the contribution coming from each couple of points is reweighted to account for spatial inhomogeneity (Baddeley *et al.* , 2000). Standard estimates of the inhomogeneous bivariate K-functions $\hat{K}_{ij}$ are obtained using the *spatstat* *R* package (Baddeley & Turner, 2005).

2.3.2 Null hypothesis testing

In order to test whether the interaction shown by K-functions is significant or not we need to define a null hypothesis (representing no-interaction among placenames). Section 8 of Baddeley (2010) describes three classical null hypotheses for multivariate point processes: random labeling (given the locations the point types are i.i.d.), Complete Spatial Randomness and Independence (CSRI, the locations arise from a uniform Poisson point process and the point types are i.i.d.) and independence of components (points of different types are independent). The random labeling and the CSRI hypotheses are unrealistic assumptions for our dataset because our point pattern is clearly not stationary and the distribution of placenames is not spatially homogeneous (some placenames are more concentrated in the South, some in the North and so on). The independence of components hypothesis is realistic but, in order to test it, stationarity of the points pattern is usually assumed. Instead we define the following no-interaction null hypothesis: each sub-pattern of points $\mathbf{x}_j$ is an inhomogeneous Poisson point process (with intensity function $\lambda_j(\cdot)$ potentially varying over j). Note that a more realistic null hypothesis would also include repulsion among points of the same type. In Section 2.3.3 we implement such a null hypothesis using Strauss point-processes. The results are very similar to the ones presented here.

Given the null hypothesis we perform the following approximate Monte Carlo test. First we estimate the intensities $\lambda_j(\cdot)$ with $\hat{\lambda}_j(\cdot)$ (see Figures 2.5) obtained through standard Gaussian kernel smoothing with bandwidth chosen according to the cross-validation method (e.g. Diggle, 2003, p.115-118), and edge correction performed according to Diggle (1985). The cross-validation method considers, for each point x_i, the density function estimated using all points apart from x_i evaluated at x_i, and then maximizes the product of such values over the bandwidth. Secondly,

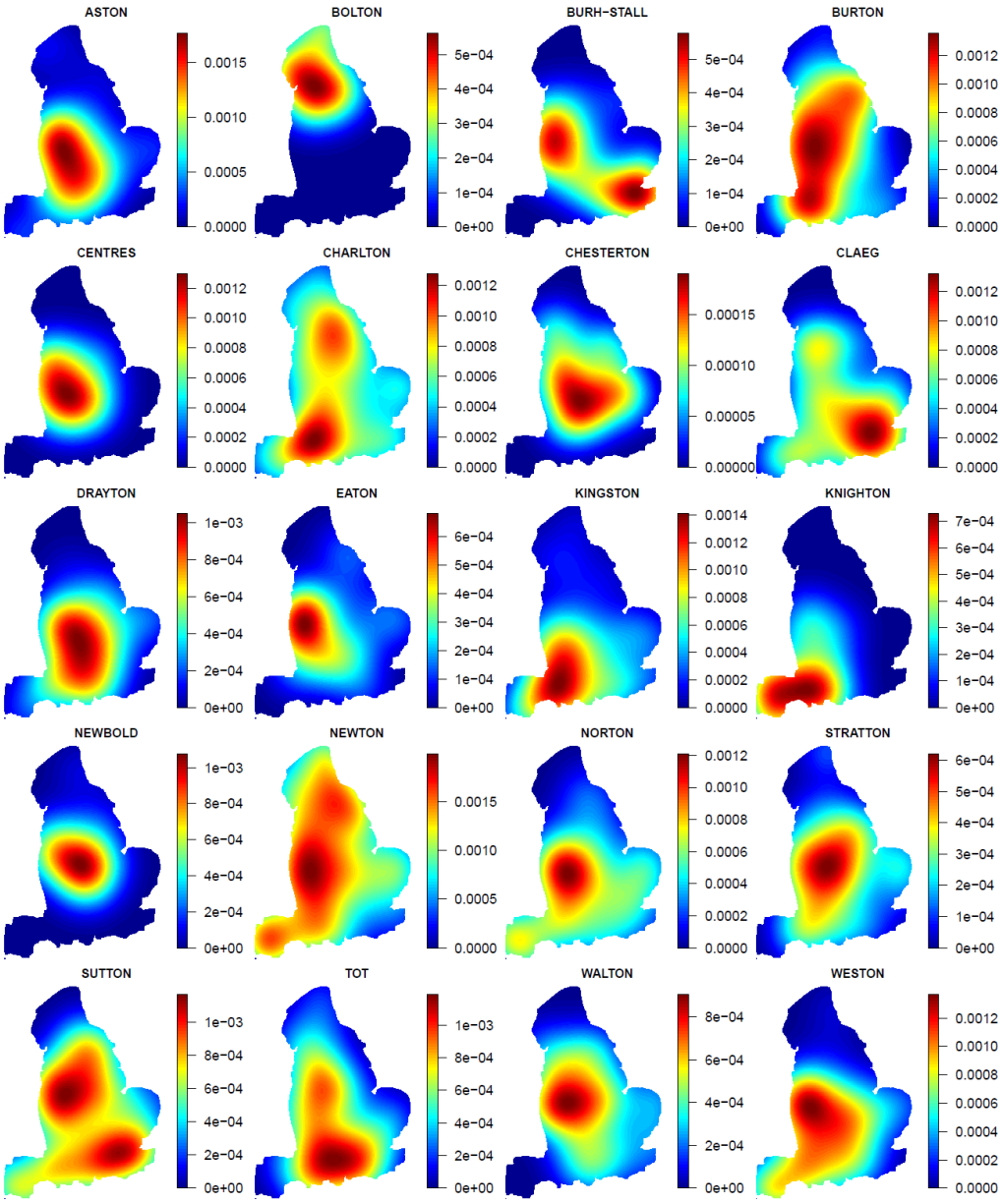


Figure 2.5: Estimates of the intensity function for each placename obtained through Gaussian kernel smoothing (and truncation) with bandwidth chosen using the function *bw.relrisk* from the Spatstat R package (Baddeley & Turner, 2005), which is based on the cross-validation method (see for example Diggle, 2003, p.115-118). Edge correction is performed according to Diggle (1985). The color scales are different for each placename, but they all start from 0. These estimates are used in Section 2.3.1 to simulate synthetic samples from the no-interaction null hypothesis.

given the intensity estimates, we sample 99 independent multivariate inhomogeneous Poisson point patterns according to $\left\{\hat{\lambda}_j(\cdot)\right\}_{j=1}^{k}$. Finally we use those samples to plot

simulation envelopes and to perform a deviation test with significance $\alpha = 0.05$ using as a summary function a centered version of the L-function $\hat{L}_{cross}(r) = \sqrt{\frac{\hat{K}_{cross}(r)}{\pi}}$ for $r \in (0, r_{max})$, with $r_{max} = 15$km. The deviation test (Grabarnik *et al.* , 2011) summarizes the summary function with a single value $D = \max_{r \in (0, r_{max})} \hat{L}_{cross}(r) - \mathbb{E}[\hat{L}_{cross}(r)]$ and compares it to the ones obtained from the 99 simulated samples.

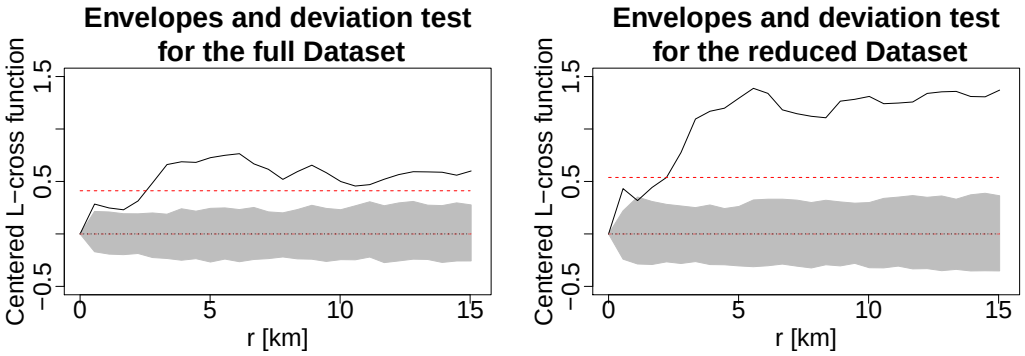


Figure 2.6: The centred L-function $\hat{L}_{cross}(r) - \mathbb{E}[\hat{L}_{cross}(r)]$ for the observed pattern is represented by (black) solid lines, the 95% envelopes (gray areas) are obtained using 99 simulated patterns and the (red) dashed lines indicate the upper deviations. Deviation test: if the (black) solid line rises above the (red) dashed line then the interaction can be considered significant at significance level $\alpha = 0.05$. The values of $\mathbb{E}[\hat{L}_{cross}(r)]$ are estimated using independently simulated point patterns generated according to the null hypothesis.

The null hypothesis is rejected for both the full and the reduced dataset (see Figure 2.6). For the reduced dataset this provides evidence of a stronger clustering effect. The R code used to perform this test and produce Figure 2.6 is given at `https://sites.google.com/site/gzanellawebpage/compclust_supp_f.zip`. Application of the same deviation test on the bivariate L-functions $\hat{L}_{ij}(r)$ provides an indication of which couples of placenames exhibit significant interaction (see Figure 2.7). Such a plot allows historians to compare the interaction reported by the K-function analysis with historical hypotheses and contextual information (see Section 7.1 for more details).

2.3.3 Null-hypothesis using Strauss point processes

In Section 2.3.2 we defined the following null hypothesis for the distribution of the marked point process $\mathbf{x}$ under consideration: each point pattern $\mathbf{x}^{(j)}$ is an inhomogeneous Poisson point process with intensity function $\lambda_j(\cdot)$. Here $\mathbf{x}^{(j)}$ denotes the type j sub-pattern of points. In order to make such a null hypothesis more realistic we could introduce some repulsion among points of the same type. In fact

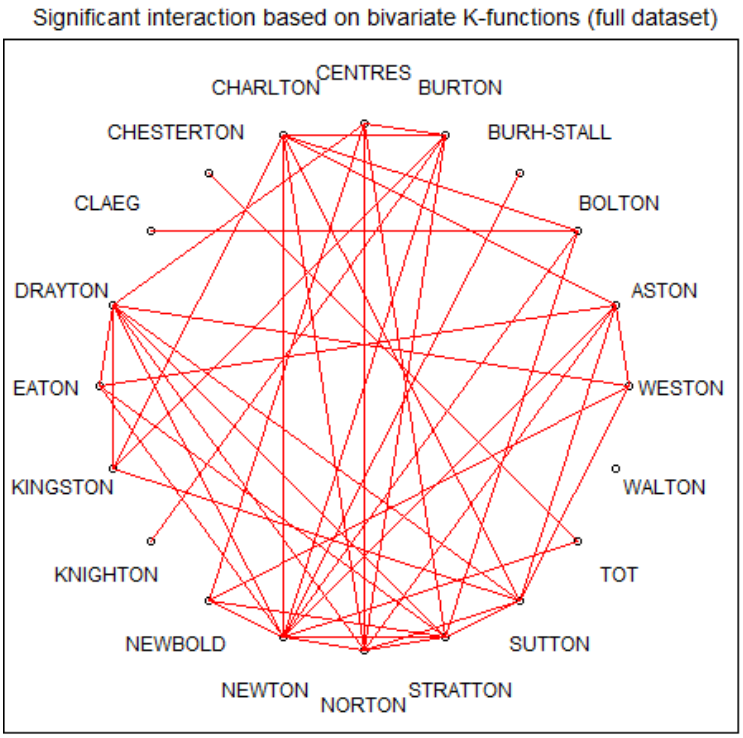


Figure 2.7: Graphical representation of significant pairwise interaction among placenames based on K-cross functions. A line connecting two placenames indicates that the deviation test described in Section 2.3.2 reports significant interaction when applied to the bivariate L-function corresponding to such two placenames (see Section 2.3.2 for more details).

it is reasonable to expect settlements with the same placename not to be too close to each other. This could be modeled by assuming that each point pattern $\mathbf{x}_j$ is distributed according to an inhomogeneous Strauss process, and $\mathbf{x}^{(j)}$ is independent from $\mathbf{x}^{(i)}$ for i different from j. A Strauss point process $\mathbf{x}^{(j)} = \{x_1^{(j)}, \ldots, x_{n(\mathbf{x}_j)}^{(j)}\}$ has probability density function

$$f(\mathbf{x}^{(j)}) = \alpha\, \gamma^{s(\mathbf{x}^{(j)})} \prod_{i=1}^{n(\mathbf{x}^{(j)})} \lambda_j(x_i^{(j)})\,,$$

with respect to the distribution of a unitary homogeneous Poisson point process (in a Radon-Nikodym derivative sense). Here α is a normalizing constant, γ is a inhibition parameter between 0 and 1, $s(\mathbf{x}^{(j)})$ is the number of (unordered) couples of points in $\mathbf{x}^{(j)}$ closer than some distance $R > 0$ apart, and $\lambda_j(\cdot)$ is the intensity function. See Chiu *et al.* (2013) for more rigorous definitions of the Strauss process and other Gibbs-type point processes.

We then perform the same approximate Monte Carlo test of Section 2.3.2, replacing the inhomogeneous Poisson point process model with the Strauss one (the estimated intensities $\hat{\lambda}_j(\cdot)$ are obtained as in Section 2.3.2). In order to perform such a test we need to choose the values of the inhibition parameter γ and the maximal inhibition distance R determining the distribution of the Strauss process. We considered γ equal to 0.1, 0.5 and 0.9 (corresponding to strong, medium and mild interaction). Given the historical context we considered values of the inhibition

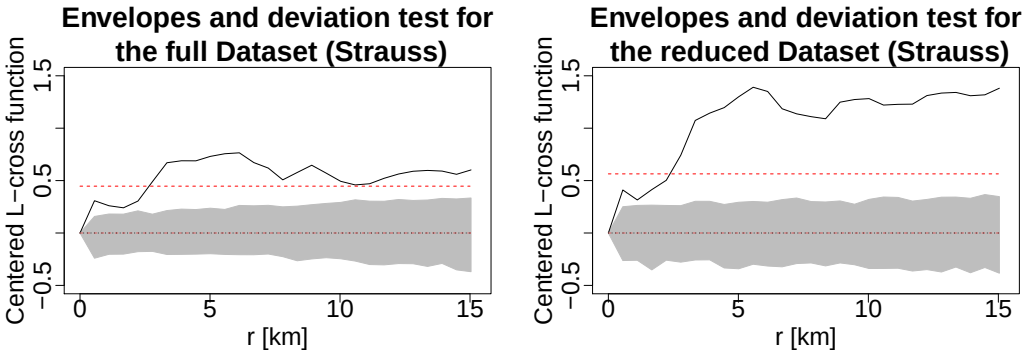


Figure 2.8: Testing the null hypothesis of Section 2.3.3, based on Strauss point processes, with the procedure described in Section 2.3.2. The centred L-function $\hat{L}_{cross}(r) - \mathbb{E}[\hat{L}_{cross}(r)]$ for the observed pattern is represented by (black) solid lines. The 95% envelopes (gray areas) are obtained using 99 simulated patterns and the (red) dashed lines indicate the upper deviations. Deviation test: if the (black) solid line rises above the (red) dashed line then the interaction can be considered significant at significance level $\alpha = 0.05$.

distance R equal to 5, 10 and 20 km. We tried all the 9 resulting combinations of γ and R. The results did not change significantly from the ones obtained in Section 2.3.2 using the inhomogeneous Poisson point process model. Figure 2.8 shows the result obtained using $\gamma = 0.1$ and $R = 20$ (the strongest interaction among the ones we considered). It can be seen that the 95% envelopes with such a null hypothesis are very similar to the ones obtained in Section 2.3.2. Note that one could try to estimate R and γ from the data. We did not consider such estimation process to be necessary at this stage, as the effect of introducing repulsion among same-type points did not impact the results of the null-hypothesis test.

2.3.4 Conclusion from the preliminary analysis

Our preliminary analysis indicates a clustering interaction between points of different types. Nevertheless K-functions do not provide explicit estimates and quantification of uncertainty for the parameters of interest (including the cluster partition itself).

In the next Chapter we develop a Random Partition Model in order to provide more informative answers to the questions of historical interest. We regard K-functions as a useful exploratory tool and the fact that they indicate interaction is a motivation to pursue further statistical analysis.

We understand that Dr. Stuart Brookes from UCL has already used second moment functions to do some preliminary analysis on the Anglo-Saxon settlements dataset presented here (personal communication by Prof. John Blair).

Chapter 3

A Bayesian complementary clustering model

3.1 Overview of possible modeling approaches

We can view our problem as a clustering problem based on aggregations of points of different types. In fact we seek a *complementary clustering*: each cluster may contain at most one settlement for each placename. This simplifying requirement is motivated by the assumption that each placename represents a different administrative function (role) within the cluster (see Section 2.1). Note that the use of a clustering model is motivated by the historians' belief that settlements were organized into administrative units that can be represented by clusters. See Section 8.2 for a discussion of some alternative modeling approaches.

Our intention is to perform explicit inferences on the partition of settlements into clusters. As with hierarchical models, it would be desirable to analyze the dataset all at once, so at not to lose statistical power, and also to provide inferences at the single cluster level to facilitate visualization and historical interpretation of the results of the analysis.

We employ Random Partition Models (RPMs), often used in the Bayesian Nonparametric literature (e.g. Lau & Green, 2007), as they permit natural inferences on the cluster partition and they have enough flexibility to allow specification of a useful model for complementary clustering.

Standard approaches for point process cluster modeling, like the Log-Gaussian Cox Processes (see Lawson & Denison, 2010, Ch.3) or the Neyman-Scott model (e.g. Loizeaux & McKeague, 2001), are not appropriate here, as such models usually provide inferences on the cluster centers or on the point process intensity, while we

seek explicit inferences on the cluster partition. Moreover standard cluster methods for marked point process consider the marks as an additional dimension and search for aggregations of points with similar marks. In complete contrast, we seek aggregations of points of different types.

Diggle *et al.* (2006) seek evidence for repulsion among points of different types in a bivariate spatial distribution of amacrine cells. They use a pairwise interaction model, which has theoretical limitations preventing its use for clustering. While this approach could be extended to our case by using area-interaction point processes, which can model clustering (Baddeley & Van Lieshout, 1995), it would not provide us with explicit estimates of the cluster partition and it would not easily allow complementary clustering specification (at most one point of each type in each cluster).

Multi-target tracking involves the Data Association problem, that is to group together measurements recorded at different time intervals to create objects tracks (e.g Oh *et al.* , 2009). This problem is similar to the problem of performing complementary clustering of a k-type point process. In Data Association problems, however, the interest is to find the best association, while we are interested in assessing the strength of clustering and the level of interaction between different placenames, and in quantifying the uncertainty of our estimates. In fact the modeling aspects we have to be careful about are different from the ones of Data Association problems, though the computational challenges are similar (see Chapters 4 and 6).

3.2 Random Partition Models

We present Random Partition Models (RPMs) in the specific context of planar k-type point processes. For more general and detailed discussions see Lau & Green (2007) and Müller & Quintana (2010). Let ρ be a partition of an ordered set of marked points $\mathbf{x} = \big((x_1, m_1), \ldots, (x_{n(\mathbf{x})}, m_{n(\mathbf{x})})\big)$, with each (x_i, m_i) belonging to $\mathbb{R}^2 \times \{1, \ldots, k\}$. Thus ρ can be represented as an *unordered* collection $\{C_1, \ldots, C_{N(\rho)}\}$ of disjoint non-trivial subsets of the indices $\{1, \ldots, n(\mathbf{x})\}$ whose union is the whole set $\{1, \ldots, n(\mathbf{x})\}$. RPMs are used to draw inferences on the partition ρ given the observed points $\mathbf{x}$. Given $C_j = \left\{i_1^{(j)}, \ldots, i_{s_j}^{(j)}\right\}$ we define

$$\mathbf{x}_{C_j} = \left(\left(x_{i_1^{(j)}}, m_{i_1^{(j)}}\right), \ \ldots \ , \left(x_{i_{s_j}^{(j)}}, m_{i_{s_j}^{(j)}}\right) \right)$$

for j running from 1 to $N(\rho)$. We call $\mathbf{x}_{C_j}$ cluster and s_j the size of the cluster. Given the partition ρ, we suppose that locations in each cluster $\mathbf{x}_{C_j}$ are generated

independently of locations in other clusters, according to a probability density function $h_{(s_j,\sigma)}(\cdot)$ depending on s_j and on a global intra-cluster dispersion parameter σ. Thus the probability density function of $\mathbf{x}$ conditional on ρ and σ is

$$\pi(\mathbf{x} \,|\, \rho, \sigma) \quad \propto \quad \prod_{j=1}^{N(\rho)} h_{(s_j,\sigma)}(\mathbf{x}_{C_j}) \,.$$

We assign independent prior distributions to ρ and σ. With a slight abuse of notation, we denote them by $\pi(\rho)$ and $\pi(\sigma)$ respectively. We require $\pi(\rho)$ to be exchangeable with respect to the point indices $\{1, \ldots, n(\mathbf{x})\}$ to reflect the fact that point labels are purely arbitrary and have no specific meaning. This is a common requirement in the RPMs literature. We obtain the following expression for the posterior density function

$$\pi(\rho, \sigma|\mathbf{x}) \quad \propto \quad \pi(\rho)\, \pi(\sigma) \prod_{j=1}^{N(\rho)} h_{(s_j,\sigma)}(\mathbf{x}_{C_j}) \,.$$

3.3 Likelihood function

Given ρ and σ, each cluster $\mathbf{x}_{C_j}$ is constructed as follows. First an unobserved center point z_j is sampled from the observation region $W \subseteq \mathbb{R}^2$ with probability density function $g(\cdot)$. Then the observed points $x_{i_1^{(j)}}, \ldots, x_{i_{s_j}^{(j)}}$ are given by

$$x_{i_l^{(j)}} = z_j + y_{i_l^{(j)}} \,, \qquad l = 1, \ldots, s_j \tag{3.1}$$

where $y_{i_l^{(j)}}$ is defined as

$$y_{i_l^{(j)}} \quad = \quad w_{i_l^{(j)}} - \frac{1}{s_j} \sum_{l=1}^{s_j} w_{i_l^{(j)}}$$

with $w_{i_1^{(j)}}, \ldots, w_{i_{s_j}^{(j)}}$ being independent bivariate $\mathrm{N}(0, \frac{\sigma^2}{\pi}\mathbb{I}_2)$ random vectors, where $\mathbb{I}_2$ is the 2×2 identity matrix. The variance parametrization $\frac{\sigma^2}{\pi}$ is chosen so that σ equals the expected distance between two points in the same cluster, independently of the value of s_j. In fact if x_1 and x_2 belong to the same cluster it holds

$$\mathbb{E}\left[\sqrt{(x_1 - x_2)^\top (x_1 - x_2)}\right] = \mathbb{E}\left[\sqrt{(w_1 - w_2)^\top (w_1 - w_2)}\right] = \sqrt{\frac{\pi}{2}}\sqrt{\frac{2\sigma^2}{\pi}} = \sigma \,,$$

where $a^\top a = \sum_{i=1}^2 a_i^2$ for a in $\mathbb{R}^2$, and we used the fact that the euclidean norm of a two dimensional $N(0, \eta^2 \mathbb{I}_2)$ random vector follows the Rayleigh distribution and its mean equals $\sqrt{\frac{\pi}{2}}\eta$ for $\eta \geq 0$.

Finally the marks $m_{i_1^{(j)}}, \ldots, m_{i_{s_j}^{(j)}}$ are sampled uniformly from the set

$$\mathcal{M}_{s_j} \quad = \quad \left\{(m_1, \ldots, m_{s_j}) \,|\, m_l \in \{1, \ldots, k\} \text{ and } m_{l_1} \neq m_{l_2} \text{ for } l_1 \neq l_2\right\} .$$

The resulting likelihood function is

$$h_{(s_j,\sigma)}(\mathbf{x}_{C_j}) \;=\; \frac{g\left(\overline{x}_{C_j}\right) \prod_{l_1,l_2 \in C_j; l_1 \neq l_2} \mathbb{1}(m_{l_1} \neq m_{l_2})}{\frac{k!}{(k-s_j)!} s_j \, (2\sigma^2)^{s_j - 1}} \exp\left(-\frac{\pi \delta_{C_j}^2}{2\sigma^2}\right), \tag{3.2}$$

where $\overline{x}_{C_j}$ is the Euclidean barycenter of $\mathbf{x}_{C_j}$ and $\delta_{C_j}^2 = \sum_{i \in C_j} \left(x_i - \overline{x}_{C_j}\right)^\top \left(x_i - \overline{x}_{C_j}\right)$. Section 3.3.1 below provides detailed calculations to obtain (3.2).

Note that in this section we are treating $g(\cdot)$ as a known function. For the purposes of data analysis we will replace g with an estimate $\hat{g}$. Note that this replacement commits us to the use of a data-driven prior. The estimate $\hat{g}$ (see Figure 3.1) is obtained using Gaussian kernel smoothing with bandwidth chosen according to the cross-validation method (Diggle, 2003, p.115-118) and edge correction performed according to Diggle (1985). See Section 2.3.1 for more details on the cross-validation method.

Remark 1. *Given the heterogeneity in the number of settlements across different placenames, the assumption of the marks being sampled uniformly seems not to be very realistic. In Section 3.7.2 we propose an empirical Bayes approach to include non-uniformity of marks in the model while maintaining computational feasibility and we present inferences under that assumption. Here we retain the uniform marks assumption for simplicity and because the two approaches produce similar inferences. Moreover the inferences with the uniform marks assumption are more conservative (see Section 7.2) and therefore preferable in this context.*

Remark 2. *This model does not constrain $x_{i_l^{(j)}} = z_j + y_{i_l^{(j)}}$ to lie in the observation region W. To make the model more realistic one could condition the distribution of $y_{i_l^{(j)}}$ in* (3.1) *on $z_j + y_{i_l^{(j)}} \in W$ (which would be an additional form of edge-correction). Nevertheless in our application the density function g is not concentrated on the borders (apart from the England-Wales border) and the values of σ are small (below 10 kilometers) compared to the size of W. Therefore most correction terms would be negligible. Moreover computing a correction term for each center point z_j would result in a consistent additional computational burden for each step of the Markov*

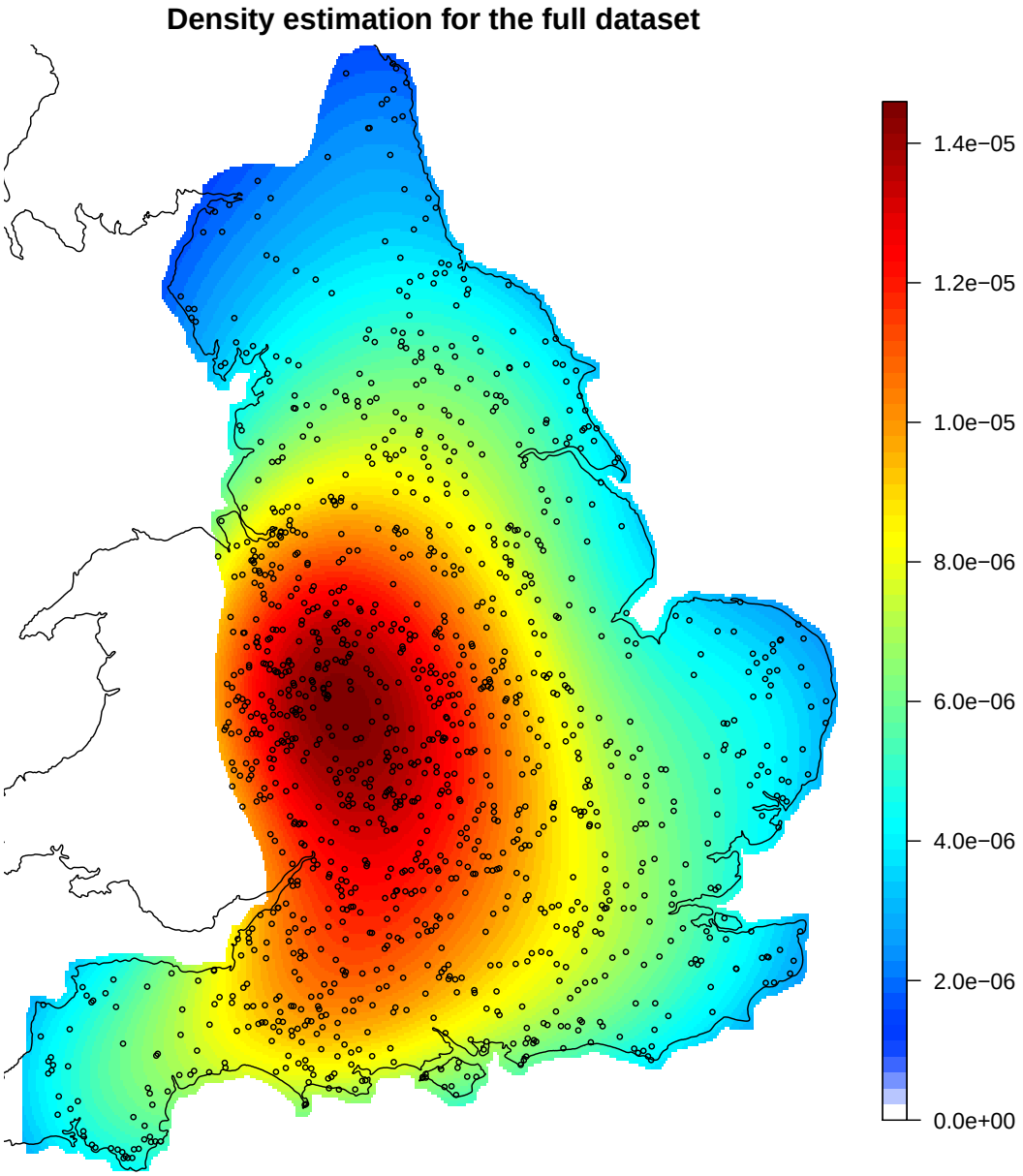


Figure 3.1: Estimate of the density function for the full dataset obtained through Gaussian kernel smoothing (and truncation) with bandwidth chosen using the function *bw.relrisk* from the Spatstat R package (Baddeley & Turner, 2005), which is based on the cross-validation method (see e.g. Diggle, 2003, p.115-118). Edge correction has been performed according to Diggle (1985). This function is used as an estimate of the probability density $g(\cdot)$ of the center process $\mathbf{z}$ (Section 3.3).

chain Monte Carlo (MCMC) algorithm in Chapter 6. Therefore we avoid such correction terms here. Note that, since such correction terms would increase the

probability of points being clustered, this approximation has a conservative effect.

3.3.1 Derivation of likelihood function

We now provide detailed calculations to derive the likelihood expression in (3.2). Suppose that $x_1, ..., x_s$ are random vectors in $\mathbb{R}^2$ given by

$$x_l = z + y_l \ , \qquad l = 1, ..., s, \tag{3.3}$$

where z has probability density function $g(\cdot)$ on $\mathbb{R}^2$ and

$$y_l = w_l - \frac{1}{s}\sum_{i=1}^{s} w_i \ , \qquad l = 1, ..., s \ , \tag{3.4}$$

with $w_1, ..., w_s$ being s independent bivariate $N(0, \frac{\sigma^2}{\pi}\mathbb{I}_2)$ random vectors, where $\mathbb{I}_2$ is the 2×2 identity matrix. We need to prove that the probability density function (pdf) of $\mathbf{x} = (x_1, ..., x_s)$ on $\mathbb{R}^{2s}$ is

$$f_{(s,\sigma)}(x_1, ..., x_s) \quad = \quad \frac{g\left(\overline{x}\right)}{s(2\sigma^2)^{s-1}} \exp\left(-\frac{\pi\delta_{\mathbf{x}}^2}{2\sigma^2}\right), \tag{3.5}$$

where $\overline{x}$ is the Euclidean barycenter of $\mathbf{x}$ and $\delta_{\mathbf{x}}^2 = \sum_{i=1}^{s}\left(x_i - \overline{x}\right)^2$. Expression (3.2) can be obtained from multiplying (3.5) by

$$\frac{1}{\frac{k!}{(k-s_j)!}} \prod_{i,j=1,\, i\neq j}^{k} \mathbb{1}(m_i \neq m_j)\,. \tag{3.6}$$

The expression in (3.6) comes from the probability of obtaining a sequence of marks $(m_1, \ldots, m_s)$, that is $\prod_{i\neq j}\mathbb{1}(m_i \neq m_j) \cdot \frac{(k-s)!}{k!}$.

Let $y_i = \left(y_i^{(1)}, y_i^{(2)}\right)$ for i running from 1 to s. Note that the random vectors $\left(y_1^{(1)}, \ldots, y_s^{(1)}\right)$ and $\left(y_1^{(2)}, \ldots, y_s^{(2)}\right)$ are independent and identically distributed. Thus it suffices to consider $\left(y_1^{(1)}, \ldots, y_s^{(1)}\right)$.

If we define $\mathbf{y} = \left(y_1^{(1)}, ..., y_s^{(1)}\right)^T$ and $\mathbf{w} = \left(w_1^{(1)}, ..., w_s^{(1)}\right)^T$ then (3.4) can be expressed as

$$\mathbf{y} = \mathbf{w} - \frac{1}{s}\mathbb{H}_s\mathbf{w},$$

where $\mathbb{H}_s$ is the $s \times s$ matrix with 1 in every position. Since the random vector $\mathbf{y}$

has zero mean then its covariance matrix Σ is

$$\Sigma \;=\; \mathbb{E}[\mathbf{y}^T\mathbf{y}] \;=\; \mathbb{E}[\mathbf{w}^T\mathbf{w}] - \frac{1}{s}\mathbb{E}[\mathbf{w}^T\mathbb{H}_s\mathbf{w}] - \frac{1}{s}\mathbb{E}[\mathbf{w}^T\mathbb{H}_s^T\mathbf{w}] + \frac{1}{s^2}\mathbb{E}[\mathbf{w}^T\mathbb{H}_s^T\mathbb{H}_s\mathbf{w}].$$

Then using the fact that $\mathbb{H}_s^T\mathbb{H}_s = s\mathbb{H}_s$ and $\mathbb{H}_s^T = \mathbb{H}_s$ we obtain

$$\Sigma \;=\; \frac{\sigma^2}{\pi}\mathbb{I}_s - \frac{2}{s}\mathbb{E}[\mathbf{w}^T\mathbb{H}_s\mathbf{w}] + \frac{1}{s}\mathbb{E}[\mathbf{w}^T\mathbb{H}_s\mathbf{w}] \;= \\ \frac{\sigma^2}{\pi}\mathbb{I}_s - \frac{1}{s}\mathbb{E}[\mathbf{w}^T\mathbb{H}_s\mathbf{w}] \;=\; \frac{\sigma^2}{\pi}\left(\mathbb{I}_s - \frac{\mathbb{H}_s}{s}\right).$$

Note that $y_s^{(1)}$ equals $-\sum_{i=1}^{s-1} y_i^{(1)}$ because

$$\sum_{i=1}^{s} y_i^{(1)} = \sum_{i=1}^{s}\left(w_i^{(1)} - \frac{1}{s}\sum_{j=1}^{s} w_j^{(1)}\right) = \sum_{i=1}^{s} w_i^{(1)} - \sum_{j=1}^{s} w_j^{(1)} = 0.$$

Therefore we can focus on the distribution of $y_1^{(1)}, ..., y_{s-1}^{(1)}$ only. Such random variables form a Gaussian random vector $\mathbf{y}_{s-1} = \left(y_1^{(1)}, ..., y_{s-1}^{(1)}\right)^T$ with zero mean and covariance matrix Σ_{s-1} which is the restriction of Σ to the first $s-1$ coordinates

$$\Sigma_{s-1} = \frac{\sigma^2}{\pi}\left(\mathbb{I}_{s-1} - \frac{\mathbb{H}_{s-1}}{s}\right). \tag{3.7}$$

Therefore the joint pdf of $\mathbf{y}_{s-1}$ in $\mathbb{R}^{s-1}$ is

$$(2\pi)^{-\frac{s-1}{2}}|\Sigma_{s-1}|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\mathbf{y}_{s-1})^T\Sigma_{s-1}^{-1}\mathbf{y}_{s-1}\right),$$

where $|\Sigma_{s-1}|$ denotes the determinant of Σ_{s-1}. Using the fact that $\mathbb{H}_{s-1}^2$ equals $(s-1)\mathbb{H}_{s-1}$ we can show that the inverse of Σ_{s-1} is $\frac{\pi}{\sigma^2}(\mathbb{I}_{s-1} + \mathbb{H}_{s-1})$. In fact

$$\frac{\sigma^2}{\pi}\left(\mathbb{I}_{s-1} - \frac{\mathbb{H}_{s-1}}{s}\right)\frac{\pi}{\sigma^2}\left(\mathbb{I}_{s-1} + \mathbb{H}_{s-1}\right) \;=\; \mathbb{I}_{s-1} + \mathbb{H}_{s-1} - \frac{\mathbb{H}_{s-1}}{s} - \frac{\mathbb{H}_{s-1}^2}{s} \;= \\ \mathbb{I}_{s-1} + \frac{s-1}{n}\mathbb{H}_{s-1} - \frac{s-1}{n}\mathbb{H}_{s-1} \;=\; \mathbb{I}_{s-1}.$$

The determinant of Σ_{s-1} is $\frac{1}{s}\left(\sigma^2/\pi\right)^{s-1}$. This can be derived by the fact that the $s-1$ eigenvalues of $\frac{\pi}{\sigma^2}\Sigma_{s-1} = \mathbb{I}_{s-1} - \frac{\mathbb{H}_{s-1}}{s}$ are $\frac{1}{s}, 1, ..., 1$. An orthonormal basis of corresponding eigenvectors is given by the rows $\mathbf{r}_1, ..., \mathbf{r}_{s-1}$ of an $(s-1)\times(s-1)$

Helmert matrix:

$$\begin{aligned}
\mathbf{r}_1 &= (s-1)^{-1/2}\big(1,...,1\big), \\
\mathbf{r}_k &= \big(k(k-1)\big)^{-1/2}\big(1,...,1,1-k,0,...,0\big) \qquad k=2,...,s-2, \\
\mathbf{r}_{s-1} &= \big((s-1)(s-2)\big)^{-1/2}\big(1,...,1,1-(s-1)\big).
\end{aligned}$$

From $\mathbb{H}_{s-1}\mathbf{r}_1^T = (s-1)\mathbf{r}_1^T$ and $\mathbb{H}_{s-1}\mathbf{r}_k^T = 0$ for k in $2,...,s-1$, it follows

$$\left(\mathbb{I}_{s-1} - \frac{\mathbb{H}_{s-1}}{s}\right)\mathbf{r}_1^T = \mathbf{r}_1^T - \frac{(s-1)}{s}\mathbf{r}_1^T = \frac{1}{s}\mathbf{r}_1^T$$

and

$$\left(\mathbb{I}_{s-1} - \frac{\mathbb{H}_{s-1}}{s}\right)\mathbf{r}_k^T = \mathbf{r}_k^T - 0 = \mathbf{r}_k^T \qquad k=2,...,s-1.$$

Therefore the joint pdf of $\mathbf{y}_{s-1}$ in $\mathbb{R}^{s-1}$ is

$$2^{-\frac{s-1}{2}}\frac{\sqrt{s}}{\sigma^{s-1}}\exp\left(-\frac{\pi}{2\sigma^2}(\mathbf{y}_{s-1})^T\big(\mathbb{I}_{s-1}+\mathbb{H}_{s-1}\big)\mathbf{y}_{s-1}\right). \tag{3.8}$$

Focusing on the exponent we have

$$-\frac{\pi}{2\sigma^2}(\mathbf{y}_{s-1})^T\big(\mathbb{I}_{s-1}+\mathbb{H}_{s-1}\big)\mathbf{y}_{s-1} = -\frac{\pi}{2\sigma^2}\left(\sum_{i=1}^{s-1}\left(y_i^{(1)}\right)^2 + \sum_{i=1}^{s-1}\sum_{j=1}^{s-1}y_i^{(1)}y_j^{(1)}\right),$$

which equals

$$-\frac{\pi}{2\sigma^2}\left(\sum_{i=1}^{s-1}\left(y_i^{(1)}\right)^2 + \left(\sum_{i=1}^{s-1}y_i^{(1)}\right)^2\right).$$

If we multiply together the joint pdfs of $\left(y_1^{(1)},\ldots,y_{s-1}^{(1)}\right)$ and $\left(y_1^{(2)},\ldots,y_{s-1}^{(2)}\right)$ we obtain the following expression for the pdf of the Gaussian family $y_1,...,y_{s-1}$ in $(\mathbb{R}^2)^{s-1}$, where $y_i = \left(y_i^{(1)}, y_i^{(2)}\right)$

$$\frac{s}{(2\sigma^2)^{s-1}}\exp\left(-\frac{\pi}{2\sigma^2}\left(\sum_{i=1}^{s-1}|y_i|^2 + \left|\sum_{i=1}^{s-1}y_i\right|^2\right)\right). \tag{3.9}$$

The pdf of $\mathbf{x}$ given in (3.5) can be obtained by linear transformation from the pdf

of z and $y_1, ..., y_{s-1}$. Equations (3.3) and (3.4) can be expressed as

$$x_i = z + y_i \qquad i = 1, ..., s-1,$$
$$x_s = z - \sum_{i=1}^{s-1} y_i \,,$$

or equivalently as

$$\mathbf{x}^{(j)} = z^{(j)} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} + \left(\begin{array}{ccc} & \mathbb{I}_{s-1} & \\ \hline -1 & \cdots & -1 \end{array}\right) \begin{pmatrix} y_1^{(j)} \\ y_2^{(j)} \\ \vdots \\ y_{s-1}^{(j)} \end{pmatrix} \qquad j = 1, 2,$$

where the j-th superscript denotes the j-th coordinate and $\mathbb{I}_{s-1}$ denotes the $(s-1) \times (s-1)$ identity matrix. Thus $\mathbf{x}^{(j)}$ is a linear transformation of the random vector $(z^{(j)}, y_1^{(j)}, ..., y_{s-1}^{(j)})^T$ through the matrix

$$J_s = \left(\begin{array}{c|ccc} 1 & & & \\ \vdots & & \mathbb{I}_{s-1} & \\ 1 & & & \\ \hline 1 & -1 & \cdots & -1 \end{array}\right). \tag{3.10}$$

Therefore the pdf of $\mathbf{x}$ in $\mathbb{R}^{2s}$ is equal to the pdf of $(z, y_1, ..., y_{s-1})^T$ divided by the squared determinant $|J_s|^2$. Using Laplace's formula on the last row

$$|J_s| \;=\; (-1)^{s+1}|M_{s,1}| + \sum_{j=2}^{s}(-1)^{s+j}(-1)|M_{s,j}| \;= \\ (-1)^{s+1}\Big(|M_{s,1}| + \sum_{j=2}^{s}(-1)^j|M_{s,j}|\Big)\,, \tag{3.11}$$

where $M_{i,j}$ is the matrix obtained from J_s by removing the i-th row and the j-th column. Note that $M_{s,1}$ is the identity matrix so its determinant is 1. Moreover

$$M_{s,2} = \left(\begin{array}{c|ccc} 1 & 0 & \cdots & 0 \\ \hline 1 & & & \\ \vdots & & \mathbb{I}_{s-2} & \\ 1 & & & \end{array}\right),$$

and so its determinant is 1 too. For $j = 3, ..., s$, note that $M_{s,j}$ can be obtained from $M_{s,j-1}$ by switching the $(j-2)$-th row and the $(j-1)$-th one. Therefore

$$|M_{s,j}| \;=\; (-1)|M_{s,j-1}| \;=\; (-1)^{j-2}|M_{s,2}| \;=\; (-1)^{j-2}\,.$$

Plugging this results in (3.11) it follows

$$|J_s| \;=\; (-1)^{s+1}\Big(1+\sum_{j=2}^{s}(-1)^j(-1)^{j-2}\Big) \;=\; (-1)^{s+1}s\,, \tag{3.12}$$

and therefore $|J_s|^2 = s^2$. Multiplying together the pdf of z and the pdf of $y_1, ..., y_{s-1}$ obtained in (3.9), and dividing by the Jacobian term we obtain (3.5) and therefore (3.2).

3.4 Prior distribution on σ

History and context suggest some considerations regarding the expected intra-cluster dispersion (in particular σ between 3 and 10km). For example, a basic consideration is that settlements of the same cluster needed to be at no more than a few hours walking distance, in order for the inhabitants of the settlements to interact administratively and politically. Nevertheless we prefer not to impose strong prior information on σ, as this gives us the opportunity to see whether our study of geographical location is in accordance with available contextual information. We use a flat uniform prior for σ, as recommended for example by Gelman (2006, Sec. 7.1)

$$\sigma \;\sim\; \text{Unif}(0, \sigma_{max})\,.$$

We set $\sigma_{max} = 50$km. Given the historical context, such an upper bound for σ constitutes a safe and conservative assumption. We tested other values of σ_{max}, namely 20 and 100 km, and the inferences presented in Chapter 7 were not sensitive to such changes, which is in accordance, for example, with Gelman (2006, Sec.2.2).

3.5 Prior distribution on ρ

We need to model a partition made up of many small clusters. In fact each cluster can contain at most k points (one for each color), and the historians expect most of the original clusters to have had fewer than 6 settlements. Common RPMs usually result in clusters with many data points each and therefore do not seem

to be appropriate to our case (see for example Remark 3). We now define a prior distribution $\pi(\rho)$ designed for situations where each cluster can have at most k points, with k being small compared to the number of points n.

3.5.1 Poisson Model for $\pi(\rho)$

In this model the number of clusters $N(\rho)$ follows a Poisson distribution with mean λ and each cluster size s_j is sampled from $\{1, \dots k\}$ according to a probability distribution

$$\mathbf{p}^{(c)} = (p_1^{(c)}, \dots, p_k^{(c)}) \,,$$

with $p_i^{(c)} \geq 0$ and $\sum_{i=1}^{k} p_i^{(c)} = 1$. Note that, as a consequence, the (unobserved) point process of centers $\{z_1, \dots, z_{N(\rho)}\}$ is a Poisson point process with intensity measure $\lambda\, g(\cdot)$ and the number of observed points need not equal n. Conditioning on observing n points, the induced prior distribution on ρ is

$$\pi(\rho|\lambda, \mathbf{p}^{(c)}) \;\propto\; \prod_{j=1}^{N(\rho)} \lambda\, |s_j|!\, p_{s_j}^{(c)} \,. \tag{3.13}$$

We assign the following conjugate priors to λ and $\mathbf{p}^{(c)}$

$$\begin{aligned} \lambda &\sim \text{Gamma}(k_\lambda, \theta_\lambda) \,, \\ \mathbf{p}^{(c)} = (p_1^{(c)}, \dots, p_k^{(c)}) &\sim \text{Dir}(\alpha_1^{(c)}, \dots, \alpha_k^{(c)}) \,. \end{aligned}$$

Combinations of the following choices of hyperparameters did not change the posterior significantly: $k_\lambda = 0.5, 1, 3$; $\theta_\lambda = 100, 300, 600$ and $(\alpha_1^{(c)}, \dots, \alpha_k^{(c)}) = (1/k, \dots, 1/k)$, $(1, \dots, 1)$ and $(1, 1/(k-1) \dots, 1/(k-1))$. In the data analysis of Chapter 7 we set

$$\begin{aligned} k_\lambda &= 1 \,, \\ \theta_\lambda &= 300 \,, \\ (\alpha_1^{(c)}, \dots, \alpha_k^{(c)}) &= (1/k, \dots, 1/k) \,. \end{aligned}$$

Remark 3. *In the RPMs literature it is common to assign a Dirichlet Process (DP) prior to ρ, which is $\pi(\rho \,|\, \theta) \propto \prod_{j=1}^{N(\rho)} \theta(s_j - 1)!$. The parameter θ is often called the concentration parameter and can either be fixed or random. A DP prior conditioning on having no cluster with more than k points would be equivalent to the Poisson model for a fixed $\boldsymbol{p}^{(c)}$, namely $p_s^{(c)} = \left(\sum_{l=1}^{k} \frac{1}{l}\right)^{-1} \frac{1}{s}$, for $s = 1, \dots, k$. Our experiments suggest that, in this complementary clustering context, the posterior distribution of ρ is highly sensitive to the value of the probability vector $\boldsymbol{p}^{(c)}$.*

Therefore we preferred to provide a prior distribution to $\boldsymbol{p}^{(c)}$ and estimate it, rather than fixing its value a priori (e.g. by fixing $p_s^{(c)} \propto \frac{1}{s}$ or $p_s^{(c)} = \frac{1}{k}$).

Remark 4. *Note that, for general $\boldsymbol{p}^{(c)}$ and k, the prior distribution $\pi(\rho|\lambda, \boldsymbol{p}^{(c)})$ in* (3.13) *is finitely exchangeable but not infinitely exchangeable. Although infinite exchangeability is a desirable property for random partition models (Lau & Green, 2007; Müller & Quintana, 2010) it appears to be too restrictive for this framework. For example, in our model the size of the clusters of ρ should be upper bounded by k and this cannot be obtained with an infinitely exchangeable model.*

Remark 5. *In Section 3.7.1 we define an alternative model for the prior distribution $\pi(\rho)$, based on the Dirichlet-Multinomial distribution rather than the Poisson one.*

3.6 Model parameters and Posterior Distribution

The model presented above results in the following unknown elements

$$(\rho, \sigma, \mathbf{p}^{(c)}, \lambda) \ \in \ \mathcal{P}_n \times \mathbb{R}_+ \times [0,1]^k \times \mathbb{R}_+,$$

where $\mathcal{P}_n$ is the set of all partitions of $\{1, \ldots, n\}$. Figure 3.2 provides a graphical representation of the underlying conditional independence structure. Given the

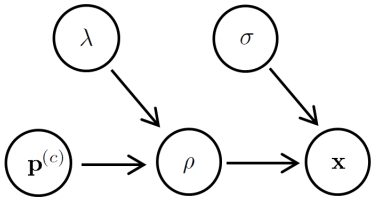


Figure 3.2: Conditional independence structure of the random elements involved in the Poisson Model.

prior and likelihood distributions described in Sections 3.3, 3.4 and 3.5, we obtain the following conditional posterior distributions:

$$\pi(\rho \mid \mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda) \propto \prod_{j=1}^{N(\rho)} \left(\frac{g\left(\overline{x}_{C_j}\right) \lambda \, p_{s_j}^{(c)}}{c_{s_j} \sigma^{2(s_j-1)}} \ \exp\left(-\frac{\pi \delta_{C_j}^2}{2\sigma^2}\right) \prod_{i,l \in C_j,\, i \neq l} \mathbb{1}(m_i \neq m_l) \right), \quad (3.14)$$

$$\pi(\sigma \mid \mathbf{x}, \rho, \mathbf{p}^{(c)}, \lambda) \quad \propto \quad \frac{\mathbb{1}_{(0,\sigma_{max})}(\sigma)}{\sigma^{2(n-N(\rho))}} \exp\left(\frac{\pi \sum_{j=1}^{N(\rho)} \delta_{C_j}^2}{2\sigma^2}\right), \tag{3.15}$$

$$\mathbf{p}^{(c)} \mid \mathbf{x}, \rho, \sigma, \lambda \quad \sim \quad \mathrm{Dir}\left(\alpha_1^{(c)} + N_1(\rho), \ldots, \alpha_k^{(c)} + N_k(\rho)\right), \tag{3.16}$$

$$\lambda \mid \mathbf{x}, \rho, \sigma, \mathbf{p}^{(c)} \quad \sim \quad \mathrm{Gamma}\left(k_\lambda + N(\rho)\ ,\ \theta_\lambda/(\theta_\lambda + 1)\right). \tag{3.17}$$

Here, $c_s = \binom{k}{s_j} s_j\, 2^{s_j - 1}$ and $\mathbb{1}_{(0,\sigma_{max})}(\cdot)$ is the indicator function of $(0, \sigma_{max})$.

The posterior distribution $\pi(\rho|\mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$ in (3.14) is intractable, meaning that we cannot obtain exact inferences from it and even performing approximate inferences is challenging. In fact the posterior sample space $\mathcal{P}_n$ is too large (of order between $n!$ and n^n) to perform brute force optimization or integration, and the complementary clustering condition makes it not easy to move in the state space. To make these statements more precise in Chapter 4 we describe $\pi(\rho|\mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$ in terms of hypergraphs and then we consider complexity theory results regarding its intractability.

Note that, although we have little hope of solving the problem in its general form (see Chapter 4), various methods, such as Monte Carlo ones, can still give satisfactory results in specific applications. In Chapter 6 we develop Markov chain Monte Carlo techniques to perform approximate inferences and use careful diagnostic techniques to monitor its convergence.

Remark 6. *One could consider the marginal distribution $\pi(\rho|\boldsymbol{x})$ obtained by integrating out σ, $\boldsymbol{p}^{(c)}$ and λ rather than the conditional one $\pi(\rho|\boldsymbol{x}, \sigma, \boldsymbol{p}^{(c)}, \lambda)$. However such a distribution is even more difficult to deal with because it does not factorizes over clusters as $\pi(\rho|\boldsymbol{x}, \sigma, \boldsymbol{p}^{(c)}, \lambda)$ does, see* (3.14)*. In fact the product form of* (3.14) *makes computation significantly easier (see Chapters 4 and 5).*

3.7 Model extensions

We now describe the model extensions mentioned in Remarks 1 and 5.

3.7.1 Alternative model for the prior distribution of ρ

In Section 3.5.1 we define a model for the prior distribution of the partition ρ based on the Poisson distribution. As mentioned in Remark 5, we consider an alternative model for $\pi(\rho)$ based on the Dirichlet-Multinomial distribution. In this section we define such a model an we refer to it as Dirichlet-Multinomial model.

Remark 7. *In Section 7.2 we compare the results of the analysis of the Anglo-Saxon*

settlements dataset obtained using the two different models. The results are almost equivalent. However the Poisson model is preferable because its posterior distribution factorizes over clusters, which drastically simplifies the computations needed at each MCMC step. We include the Dirichlet-Multinomial model as a form of additional sensitivity analysis with respect to the prior distribution on partitions.

Dirichlet-Multinomial Model for $\pi(\rho)$

For l running from 1 to k, we define $N_l(\rho)$ as the number of clusters of ρ having size l and $Y_l(\rho) = l \cdot N_l(\rho)$, so that $Y_l(\rho)$ is the total number of points in all the clusters of size l. Note that $\sum_{l=1}^{k} Y_l(\rho) = n(\mathbf{x})$, where $n(\mathbf{x})$ is the number of points in the k-type point pattern $\mathbf{x}$. In this model the random vector $\mathbf{Y}(\rho) = (Y_1(\rho), \ldots, Y_k(\rho))$ follows a Dirichlet-Multinomial distribution conditioned on Y_l being a multiple of l (for l running from 1 to k)

$$\Pr(Y_1 = y_1, \ldots, Y_k = y_k) \propto \begin{cases} \frac{n!}{y_1! \cdots y_k!} p_1^{y_1} \cdots p_k^{y_k} & \text{if } \sum_{l=1}^{k} y_l = n \text{ and} \\ & \quad y_l \text{ is a multiple of } l, \\ 0 & \text{otherwise.} \end{cases} \tag{3.18}$$

We assume that the parameter vector $\mathbf{p} = (p_1, \ldots, p_k)$ is unknown with prior distribution

$$\mathbf{p} = (p_1, \ldots, p_k) \ \sim \ \mathrm{Dir}(\alpha_1, \ldots, \alpha_k)\,.$$

The resulting prior distribution of ρ given $\mathbf{p}$, recalling that we want such distribution to be exchangeable, is

$$\pi(\rho \mid \mathbf{p}) \quad \propto \quad \frac{1}{\eta(\rho)} \, \frac{n(\mathbf{x})!}{Y_1(\rho)! \cdots Y_k(\rho)!} p_1^{Y_1(\rho)} \cdots p_k^{Y_k(\rho)}, \tag{3.19}$$

where $\eta(\rho) = \#\{\tilde{\rho} \mid \mathbf{Y}(\rho) = \mathbf{Y}(\tilde{\rho})\} = n! \left(\prod_{l=1}^{k} (l!)^{\frac{Y_l}{l}} \, (Y_l/l)! \right)^{-1}$. Equivalently

$$\pi(\rho \mid \mathbf{p}) \quad \propto \quad \prod_{l=1}^{k} \left(\frac{N_l(\rho)!}{(l N_l(\rho))!} \right) \prod_{j=1}^{N(\rho)} \left(s_j! \, p_{s_j}^{s_j} \right) . \tag{3.20}$$

Note from (3.20) that in the Dirichlet-Multinomial model $\pi(\rho \mid \mathbf{p})$ does not factorize over clusters because of the terms depending on $N_1(\rho)$, …, $N_k(\rho)$.

Remark 8. *This model can be seen as a Dirichlet-Multinomial mixture of k classes having Y_1, Y_2, up to Y_k points corresponding to singletons, couples, up to k-tuples. We are therefore converting the problem of finding an unknown number (between $\frac{n}{k}$*

and n) of small clusters into the problem of finding k big clusters, with k fixed and relatively small (20 in our case).

Remark 9. *Note that p_l represents the probability of a point being in a cluster of size l. Since we conditioned Y_l on being a multiple of l, though, this is just an approximation. Nevertheless for large $n(\boldsymbol{x})$ (such as $n(\boldsymbol{x}) \geq 10$) the approximation error is negligible.*

Model parameters and Posterior Distribution

The Dirichlet-Multinomial model results in the following unknown elements

$$(\rho, \sigma, \mathbf{p}) \ \in \ \mathcal{P}_n \times \mathbb{R}_+ \times [0,1]^k,$$

where $\mathcal{P}_n$ is the set of all partitions of $\{1, \ldots, n\}$. Figure 3.3 provides a graphical representation of the underlying conditional independence structure. Given the prior

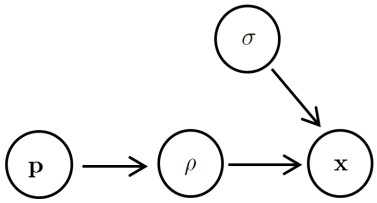


Figure 3.3: Conditional independence structure of the random elements involved in the Dirichlet-Multinomial model.

distribution described above and the likelihood distribution described in Section 3.3 we obtain the following conditional posterior distributions for the Dirichlet-Multinomial model:

$$\pi(\rho \mid \mathbf{x}, \sigma, \mathbf{p}) \ \propto \ \prod_{l=1}^{k} \frac{N_l!}{(lN_l)!} \cdot \\ \cdot \prod_{j=1}^{N(\rho)} \left(\frac{g\left(\overline{x}_{C_j}\right) (p_{s_j})^{s_j}}{c_{s_j} \sigma^{2(s_j-1)}} \ \exp\left(-\frac{\pi \delta_{C_j}^2}{2\sigma^2}\right) \ \prod_{i,l \in C_j,\, i \neq l} \mathbb{1}(m_i \neq m_l) \right) , \quad (3.21)$$

$$\mathbf{p} \mid \mathbf{x}, \rho, \sigma \quad \sim \quad \mathrm{Dir}\left(\alpha_1 + Y_1(\rho), \ldots, \alpha_k + Y_k(\rho)\right) . \quad (3.22)$$

where $c_s = \binom{k}{s_j} s_j \, 2^{s_j - 1}$. Similarly to the Poisson model, the full conditional posterior distribution of σ, $\pi(\sigma|\mathbf{x}, \rho, \mathbf{p})$, depends only on σ, $\mathbf{x}$ and ρ and is given by (3.15).

3.7.2 Dropping the uniform marks assumption

When defining the likelihood function in Section 3.3 we assumed that, given the number of points s in a cluster $\mathbf{x}_C$, the marks $m_1, \ldots, m_s$ of such points are sampled uniformly from the set

$$\mathcal{M}_s \quad = \quad \left\{ \{m_1, \ldots, m_s\} \subseteq \{1, \ldots, k\} \mid m_{l_1} \neq m_{l_2} \text{ for } l_1 \neq l_2 \right\}. \tag{3.23}$$

Since the cardinality of $\mathcal{M}_s$ is $\binom{k}{s}$, this leads to the term

$$\frac{1}{\binom{k}{s}} \prod_{i,l=1,\, i \neq l}^{k} \mathbb{1}(m_i \neq m_l) \tag{3.24}$$

in the likelihood function $h_{(s,\sigma)}(\mathbf{x}_C)$ given in (3.2).

Nevertheless, as already mentioned in Remark 1, the assumption of the marks being sampled uniformly does not seem very realistic because of the heterogeneity in the number of settlements across different placenames (see Table 2.1). In this section we develop a model where the marks within each cluster are sampled non-uniformly.

Suppose we have a probability vector on the set of possible marks $\{1, \ldots, k\}$,

$$\mathbf{p}^{(m)} = \left(p_1^{(m)}, \ldots, p_k^{(m)} \right), \tag{3.25}$$

with $p_i^{(m)} > 0$ for any i and $\sum_{i=1}^{k} p_i^{(m)} = 1$. Then, given the number of points s in a cluster $\mathbf{x}_C$, the marks $m_1, \ldots, m_s$ are independently sampled from $\{1, \ldots, k\}$ according to $\mathbf{p}^{(m)}$, conditioning on all the marks being different among themselves. Therefore the probability of a certain configuration $m_1, \ldots, m_s$ is

$$\frac{p_{m_1}^{(m)} \cdots p_{m_s}^{(m)}}{Z_s} \prod_{i,j=1,\, i \neq j}^{k} \mathbb{1}(m_i \neq m_j), \tag{3.26}$$

where Z_s is a normalizing constant defined as

$$Z_s \;=\; Z_s(\mathbf{p}^{(m)}) \;=\; \sum_{\{a_1, \ldots, a_s\} \in \mathcal{M}_s} p_{a_1}^{(m)} \cdots p_{a_s}^{(m)}. \tag{3.27}$$

Note that if the probability vector $\mathbf{p}^{(m)}$ is uniform then (3.26) equals (3.24). Replacing (3.24) with (3.26) in the likelihood function (3.2) we obtain the new likelihood

function

$$h_{(s,\sigma)}(\mathbf{x}_C) \;=\; \frac{p_{m_1}^{(m)} \cdots p_{m_s}^{(m)}\, g\left(\overline{x}_C\right) \prod_{i,j=1,\, i\neq j}^{k} \mathbb{1}(m_i \neq m_j)}{Z_s\, s\, (2\sigma^2)^{s-1}} \exp\left(-\frac{\pi \delta_C^2}{2\sigma^2}\right), \qquad (3.28)$$

where, as in Section 3.3, $\overline{x}_C$ is the Euclidean barycenter of $\mathbf{x}_C$ and $\delta_C^2 = \sum_{i\in C} \left(x_i - \overline{x}_C\right)^\top \left(x_i - \overline{x}_C\right)$. Since the terms $\{Z_s\}_{s=1}^k$ make the computation hard, in Section 7.2 we consider an Empirical Bayes approach to keep computation feasible.

Note that an even more realistic assumption would be to assume the probability vector in (3.25) to be spatially dependent. This would be more in accordance with the heterogeneity in the observed spatial patterns of Figure 2.5. However the approach of Section 7.2 would not be applicable to this case and therefore it is not obvious how to perform computationally efficient inferences with a spatially dependent probability vector in (3.25). See Section 8.2 for this and other research directions to improve the current modeling approach.

Chapter 4

Computational complexity of the posterior

As we already mentioned in Section 3.6, the posterior distribution $\pi(\rho \,|\, \mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$ in (3.14) is intractable. To make such a statement more precise we describe the distribution $\pi(\rho \,|\, \mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$ in terms of hypergraphs and then we consider complexity theory results from the literature related to its intractability. For simplicity we will denote $\pi(\rho \,|\, \mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$ by $\hat{\pi}(\rho)$.

4.1 Formulation of the model in terms of hypergraphs

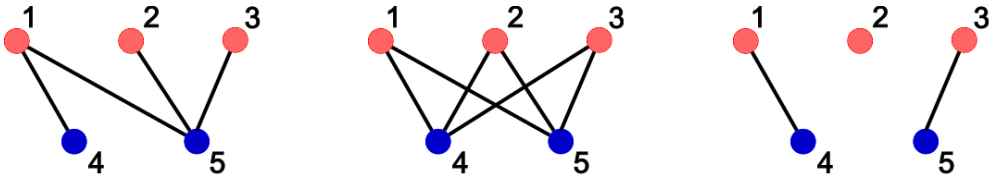
4.1.1 Two-color case

First we consider the model of Chapter 3 in the two-type or two-color case (i.e. k equal to 2). We describe the model in terms of weighted graphs and the state space in terms of matchings. We now introduce the basic notions needed.

Bipartite graphs and matchings

In the following we will consider only graphs $G = (V, E)$ which are *simple*. This means graphs that are finite, with undirected edges and no loops (i.e. no edges connecting a vertex to itself). Moreover we consider *weighted* graphs and we denote the weight function $w : E \to \mathbb{R}$. The value $w(e)$ is the weight of the edge e.

A graph $G = (V, E)$ is *bipartite* if there exist $V_1, V_2 \subseteq V$ such that $V = V_1 \cup V_2$ and any edge $e \in E$ contains one vertex in V_1 and one in V_2. A natural way of representing bipartite graphs is by coloring the vertices of V_1 and V_2 of two different colors, say red and blue. Figure 4.1(a) provides a representation of a bipartite graph.



(a) Bipartite graph with 5 vertices and 4 edges.

(b) Complete bipartite graph with 5 vertices.

(c) Matching contained in the graph of Figure 4.1a.

Figure 4.1: examples of bipartite graphs.

A bipartite graph $G = (V, E)$ is *complete* if for any red vertex v_1 and any blue vertex v_2 the edge $e = \{v_1, v_2\}$ is contained in E. Figure 4.1(b) represents a complete bipartite graph with 3 red vertices and 2 blue ones. Suppose $G = (V, E)$ is a complete bipartite graph with n_1 red vertices and n_2 blue vertices. Then we denote the edge connecting the i-th red vertex and the j-th blue vertex by the *ordered* couple (i, j). Thus there is a bijection between E and $\{1, \dots, n_1\} \times \{1, \dots, n_2\}$.

Given a graph $G = (V, E)$, a *matching* of G is a set $M \subseteq E$ such that no two edges of M share a common vertex. If the edges of a matching M contain all the vertices of the graph, then we say that M is a *complete* matching or a perfect matching. Otherwise we say that M is a *partial* matching. Figure 4.1(c) represents a (partial) matching contained in the graph of 4.1(a). We denote the set of matchings contained in a graph $G = (V, E)$ by

$$\mathcal{M}_G \quad = \quad \{M \subseteq E \,|\, \forall e_1, e_2 \in M \text{ with } e_1 \neq e_2 \text{ it holds } e_1 \cap e_2 = \emptyset\} \,. \tag{4.1}$$

If the graph G is weighted then we define the weight of a matching M as the product of the weights of its edges

$$w(M) \quad = \quad \prod_{e \in M} w(e) \,,$$

and the total weight of G as the sum of the weights of its matchings

$$w(G) \quad = \quad \sum_{M \in \mathcal{M}_G} \prod_{e \in M} w(e) \,.$$

Connection to the two-color complementary cluster model

In Chapter 3 we defined a Random Partition Model (RPM) to study complementary clustering of k-type point configurations of the form $\mathbf{x} = \{(x_1, m_1), \dots, (x_n, m_n)\} \subseteq \mathbb{R}^2 \times \{1, \dots, k\}$. When k is equal to 2 there are only two types of points. A two-type

point configuration $\mathbf{x}$ can be associated to a complete bipartite graph $G_{\mathbf{x}}$ where each point corresponds to a vertex and the two point types induce a partition into red and blue vertices.

We are interested in studying the posterior distribution $\pi(\rho \,|\, \mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$ defined in (3.14), which we denote by $\hat{\pi}(\rho)$ for simplicity. The state space of $\hat{\pi}(\rho)$ is the space of partitions of $\{1, \dots, n\}$. However $\hat{\pi}(\rho)$ is non-zero only for partitions ρ such that no cluster of ρ contains two points of the same type (i.e. i_1 and i_2 belonging to the same cluster implies $m_{i_1} \neq m_{i_2}$). Since we have only two point types, it follows that a (non-empty) set of points is an admissible cluster if and only if it is a singleton or a red-blue couple. If we associate each cluster consisting of a red-blue couple with the corresponding edge of the complete bipartite graph $G_{\mathbf{x}}$, then we obtain a bijection between admissible partitions (i.e. partitions ρ such that $\hat{\pi}(\rho) > 0$) and the set of matchings $\mathcal{M}_{G_{\mathbf{x}}}$ contained in $G_{\mathbf{x}}$ (see (4.1) for definition). For example the matching in Figure 4.1(c) corresponds to the partition $\rho = \{\{1,4\},\{3,5\},\{2\}\}$. Note that each unlinked point is a cluster by itself. From now on we will treat ρ indifferently as a partition or as a matching, as the two formulations are equivalent.

For each edge $e = \{x_1^{(e)}, x_2^{(e)}\}$ of $G_{\mathbf{x}}$ we define the weight

$$w(e) = \frac{(c_1)^2 \, \lambda \, p_2^{(c)} g\left(\overline{x}_e\right) \sigma^{-2}}{c_2 \left(\lambda \, p_1^{(c)}\right)^2 g\left(x_1^{(e)}\right) \, g\left(x_2^{(e)}\right)} \exp\left(-\frac{\pi \, \delta_e^2}{2\,\sigma^2}\right) , \tag{4.2}$$

where

$$\overline{x}_e = \frac{x_1^{(e)} + x_2^{(e)}}{2} \qquad \text{and} \qquad \delta_e^2 = \sum_{l=1}^{2} \left(x_l^{(e)} - \overline{x}_e\right)^T \left(x_l^{(e)} - \overline{x}_e\right) .$$

Given the weights in (4.2), the measure $\hat{\pi}(\rho)$ is proportional to the weight of the matching ρ. In fact multiplying $\prod_{e\in\rho} w(e)$ by the term $\prod_{i=1}^{n} \frac{\lambda \, p_1^{(c)} g(x_i)}{c_1}$, which does not depend on ρ, we obtain

$$\prod_{e\in\rho} w(e) \quad \propto \quad \prod_{i\,:\,x_i \notin e \,\forall e\in\rho} \left(\frac{\lambda \, p_1^{(c)} g\left(x_i\right)}{c_1}\right) \prod_{e\in\rho} \left(\frac{\lambda \, p_2^{(c)} g\left(\overline{x}_e\right)}{c_2 \, \sigma^2} \exp\left(-\frac{\pi \, \delta_e^2}{2\,\sigma^2}\right)\right) . \tag{4.3}$$

Since the right-hand side of (4.3) is the two-color version of $\hat{\pi}(\rho)$ in (3.14), it follows that

$$\hat{\pi}(\rho) \;\propto \prod_{e\in\rho} w(e) \qquad\qquad \rho \in \mathcal{M}_{G_{\mathbf{x}}} . \tag{4.4}$$

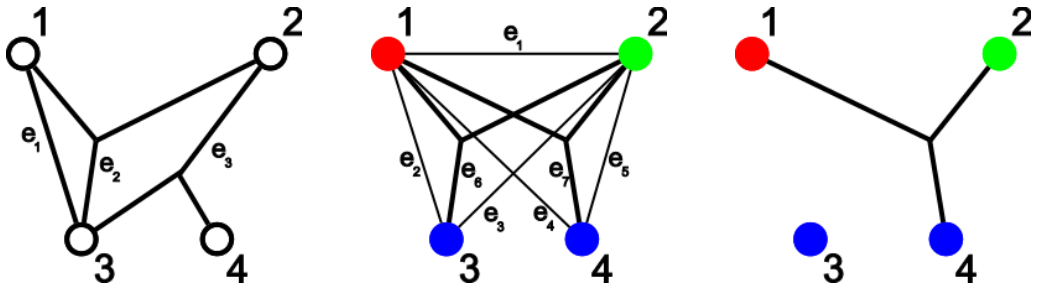
Equation (4.4) shows that the state space of $\hat{\pi}(\rho)$ can be interpreted as $\mathcal{M}_{G_{\mathbf{x}}}$, the space of matchings contained in $G_{\mathbf{x}}$, and that the probability of each matching ρ is proportional to its weight. This will be useful in understanding the computational complexity of tasks associated with the distribution $\hat{\pi}(\rho)$.

4.1.2 The general k-color case

In Section 4.1.1 we described $\hat{\pi}(\rho)$ for $k = 2$ in terms of matchings of a weighted bipartite graphs. To have an analogous characterization for $k \geq 3$ we first need to recall the notion of a hypergraph.

Complete k-partite hypergraphs

Hypergraphs are the generalization of graphs where each hyperedge can contain more than two vertices (Berge & Minieka, 1973). More precisely an *hypergraph* $G = (V, E)$ consists of two finite sets V and E. The set V is the set of vertices and the set E is the set of hyperedges. Each *hyperedge* $e \in E$ is a collection of at least two vertices, such as $e = \{v_1, \ldots, v_{|e|}\}$ with $v_1, \ldots, v_{|e|} \in V$, $|e| \geq 2$ and $v_i \neq v_j$ for $i \neq j$. We can visually represent hypergraphs in an analogous way to graphs. For example Figure 4.2(a) represents an hypergraph with $V = \{1, 2, 3, 4\}$ and $E = \big\{\{1, 3\}, \{1, 2, 3\}, \{2, 3, 4\}\big\}$. Note that, as for graphs, we consider only



(a) Hypergraph with vertices $V = \{1, 2, 3, 4\}$ and edges $E = \{e_1, e_2, e_3\}$.

(b) Complete 3-partite hypergraph with $V_1 = \{1\}$, $V_2 = \{2\}$ and $V_3 = \{3, 4\}$.

(c) Matching corresponding to the partition ρ equal to $\big\{\{1\}, \{2, 3, 4\}\big\}$.

Figure 4.2: examples of hypergraphs.

simple hypergraphs, meaning that we do not allow for loops, directed hyperedges or infinite hypergraphs.

Given $k \geq 2$, we say that a hypergraph $G = (V, E)$ is *k-partite* if there exist $V_1, \ldots, V_k \subseteq V$, with $V = V_1 \cup \cdots \cup V_k$ and $V_i \cap V_j = \emptyset$ for $i \neq j$, such that

$$|e \cap V_i| \leq 1 \qquad \forall\, e \in E,\, i \in \{1, \ldots, k\}\,.$$

In particular the *complete k-partite hypergraph* induced by $V_1, \ldots, V_k$ is defined as $G = (V, E)$ where $V = V_1 \cup \cdots \cup V_k$ and

$$E = \{e \subseteq V \,:\, |e \cap V_l| \leq 1 \; \forall \, l \,,\, |e| \geq 2\} \,.$$

Figure 4.2(b) shows a complete 3-partite hypergraph induced by $V_1 = \{1\}$, $V_2 = \{2\}$ and $V_3 = \{3, 4\}$.

As for graphs, a *matching* contained in an hypergraph $G = (V, E)$ is a set $M \subseteq E$ such that no two hyperedges of M share a common vertex. We still denote the set of matchings contained in G by $\mathcal{M}_G$, see (4.1). Figure 4.2(c) shows an example of matching contained in the complete 3-partite hypergraph of Figure 4.2(b).

Connection to the k-color complementary cluster model

We now return to the distribution of interest $\hat{\pi}(\rho)$, in the general case $k \geq 2$. A k-type point configuration $\mathbf{x} = \{(x_1, m_1), \ldots, (x_n, m_n)\} \subseteq \mathbb{R}^2 \times \{1, \ldots, k\}$ can be associated to a complete k-partite hypergraph $G_{\mathbf{x}} = (V, E)$ where each point corresponds to a vertex and the k point types induce a partition into colors. More precisely $G_{\mathbf{x}}$ is the complete k-partite hypergraph with $V = \{1, \ldots, n\}$ and $V_s = \{i \,:\, m_i = s\}$ for s from 1 to k.

As before, a partition $\rho \in \mathcal{P}_n$ of n points into clusters has non-zero probability, $\hat{\pi}(\rho) > 0$, if and only if no cluster of ρ contains two points of the same type (i.e. i_1 and i_2 belonging to the same cluster implies $m_{i_1} \neq m_{i_2}$). Therefore a set of points is an admissible cluster if and only if the hyperedge connecting such points belongs to the complete k-partite hypergraph $G_{\mathbf{x}}$. Every admissible partition ρ can then be interpreted as a partial matching contained in $G_{\mathbf{x}}$ as follows: each cluster with at least two points corresponds to a hyperedge and each unlinked point is a cluster by itself. For example Figure 4.2(c) shows the matching corresponding to the partition $\rho = \big\{\{1\}, \{2, 3, 4\}\big\}$. As for the two-color case, the state space space of $\hat{\pi}(\rho)$ can be seen as $\mathcal{M}_{G_{\mathbf{x}}}$, the space of matchings contained in $G_{\mathbf{x}}$, and we will treat ρ indifferently as a partition or as a matching.

We define the weight $w(e)$ for each hyperedge $e = \{x_1^{(e)}, \ldots, x_{|e|}^{(e)}\}$ of the hypergraph $G_{\mathbf{x}}$ as

$$w(e) = \frac{(c_1)^{|e|} \lambda \, p_{|e|}^{(c)} g\left(\overline{x}\right) \sigma^{-2(|e|-1)}}{c_{|e|} \left(\lambda \, p_1^{(c)}\right)^{|e|} g(x_1) \cdots g(x_{|e|})} \exp\left(-\frac{\pi \, \delta_e^2}{2 \, \sigma^2}\right), \tag{4.5}$$

where

$$\overline{x}_e = \frac{x_1^{(e)} + \cdots + x_{|e|}^{(e)}}{|e|} \qquad \text{and} \qquad \delta_e^2 = \sum_{l=1}^{s} \left(x_l^{(e)} - \overline{x}_e\right)^T \left(x_l^{(e)} - \overline{x}_e\right) .$$

If we multiply $\prod_{e\in\rho} w(e)$ by $\prod_{i=1}^{n} \frac{\lambda\, p_1^{(c)} g(x_i)}{c_1}$, which does not depend on ρ, we obtain

$$\prod_{e\in\rho} w(e) \quad \propto \quad \prod_{i\,:\,i\notin e\;\forall e\in\rho} \left(\frac{\lambda\, p_1^{(c)} g\left(x_i\right)}{c_1}\right) \prod_{e\in\rho} \left(\frac{\lambda\, p_{|e|}^{(c)} g\left(\overline{x}_e\right)}{c_{|e|}\;\sigma^{2(|e|-1)}} \exp\left(-\frac{\pi\,\delta_e^2}{2\,\sigma^2}\right)\right) . \quad (4.6)$$

It follows from (4.6) and (3.14) that

$$\hat{\pi}(\rho) \;\propto\; \prod_{e\in\rho} w(e)\,, \qquad\qquad \rho \in \mathcal{M}_{G_{\mathbf{x}}}\,. \quad (4.7)$$

4.2 Some complexity theory results from the literature

Given the formulation of $\hat{\pi}(\rho)$ in terms of matching, see (4.4) and (4.7), we can exploit known complexity theory results to obtain rigorous statements on the intractability of $\hat{\pi}(\rho)$. In particular we consider the complexity of the following tasks:

1. finding the normalizing constant of $\hat{\pi}(\rho)$

$$Z_{\hat{\pi}} \;=\; \sum_{\rho\in\mathcal{M}_{G_{\mathbf{x}}}} \prod_{e\in\rho} w(e)\,,$$

2. finding the mode of $\hat{\pi}(\rho)$

$$\rho_{max} = \arg\max_{\rho\in\mathcal{M}_{G_{\mathbf{x}}}} \hat{\pi}(\rho)\,,$$

3. sampling from $\hat{\pi}(\rho)$.

We will distinguish between the two-color case ($k = 2$) and the multi-color case ($k \geq 3$) because they present substantially different complexity issues. Guided by the corresponding literature in the theory of algorithms we often refer to those as two-dimensional case and k-dimensional case respectively, even though in both cases our points lie on a plane.

4.2.1 Recalling basic complexity theory notions

First we introduce some basic notions from Computational Complexity Theory. In particular we recall the basic definitions related to the notion of NP-complete decision problems. Our aim is just to provide some context for understanding the implications of the results presented later. For more rigorous and detailed definitions we refer to Garey & Johnson (1979) and Jerrum (2003).

The class P is the class of decision problems which can be solved in polynomial time (in the size of the input) by a Deterministic Turing Machine (DTM). A DTM is a basic model for CPU computation. Here we can think at a DTM as a machine which reads and modifies symbols on a tape according to a set of rules and is allowed to perform only one operation at a time. See Garey & Johnson (1979, Sec2.2) for a proper definition. Note that some care should be taken to properly define the size of the input and to show that the results do not depend on the way the input is encoded (see Garey & Johnson (1979, Sec2.1) for more details on encoding schemes).

The class NP is the class of decision problems that can be solved in polynomial time by a NonDeterministic Turing Machine (NDTM). A NDTM is a machine that first guesses a possible solution to the problem and then verifies (in polynomial time) with a DTM whether that solution was correct. We say that a NDTM solves a problem if, when the answer to the decision problem is "yes", there exists at least one guess such that the NDTM returns "yes" and if the answer is "no" then there is no guess such that the NDTM returns "yes". For this reason the class NP is often referred to as polynomially verifiable problems, meaning that we can think of NP as the class of problems for which we can verify in polynomial time with a DTM whether a proposed solution is correct or not. See Garey & Johnson (1979, Sec2.3) for a proper definition.

A decision problem is NP-hard if it is at least as hard as any problem in NP (and NP-complete if it is NP-hard and belongs to NP). By this we mean that if we can solve such a problem then we can solve any other problem in NP at roughly the same cost, because we can "transform" any instance of any problem in NP into an instance of that specific problem (see Garey & Johnson (1979, Sec2.5) for the notion of polynomial transformation). NP-complete problems are widely believed to be intractable, meaning that no algorithm is expected to exist that can solve them in polynomial time. The reason is that, if such an algorithm existed, then it would also exist for any problem in NP and therefore it would hold $P = NP$.

Note that the notion of NP-completeness is defined for decision problems, while we will mainly deal with other kind of problems, such as optimization or

counting problems. However the basic ideas related to the complexity of decision problems also extend to the main types of problems investigated here and the basic intuition is preserved.

For example any optimization problem (e.g. "find the minimum of a real valued function f") can be related to a corresponding decision problem ("is the minimum of f smaller than b?" for some real value b). The corresponding decision problem is no harder than the original optimization problem and therefore if the decision problem is NP-complete then, unless P=NP, the optimization problem cannot be solved in polynomial time either (see Garey & Johnson (1979) for more details on the relationship between decision and optimization problems).

The #P-complete complexity class for counting problems is analogous to the NP-complete complexity class for decision problems. See Valiant (1979) and Jerrum (2003, Ch.2) for more details on the complexity theory of counting problems. Given the basic intuition on the notion of NP-completeness, we now recall some results from the Computer Science literature which are related to our problem.

4.2.2 Finding the normalizing constant

The normalizing constant Z_G is the sum of the weights of all the matchings ρ contained in G, that is the total weight of G. The easiest non-trivial version is the case where $k = 2$ and the edge weights can only be 0 or 1. In such a case computing Z_G is equivalent to counting the number of partial matchings in a bipartite graph. Even in this easiest version, this problem is an $\#P$-complete counting problem (Valiant, 1979). Therefore the more general problem of computing the total weight of a weighted k-partite hypergraph is also $\#P$-hard.

4.2.3 Finding the posterior mode

Finding the mode of $\hat{\pi}(\rho)$ is related to the k-dimensional optimal assignment problem, which can be formulated as follows.

Problem 1. *(k-dimensional optimal assignment problem)*
Instance: k sets $I_1,\ldots,I_k$ of size n and a cost function $C : I_1 \times \cdots \times I_k \to \mathbb{R}$.
Problem: find an assignment A, i.e. a subset $A \subseteq I_1 \times \cdots \times I_k$ containing each point of $I_1,\ldots,I_k$ exactly once, that minimizes $\sum_{(i_1,\ldots,i_k)\in A} C(i_1,\ldots,i_k)$.

We show how to reduce the problem of finding ρ_{max} to a k-dimensional optimal assignment problem. First note that by taking a log-transform

$$\rho_{max} \quad = \quad \arg\max_{\rho\in\mathcal{M}_{G_{\mathbf{x}}}} \sum_{e\in\rho} \log\left(w(e)\right) \quad = \quad \arg\min_{\rho\in\mathcal{M}_{G_{\mathbf{x}}}} -\sum_{e\in\rho} \log\left(w(e)\right) .$$

We then need to express a matching ρ as an assignment for some choice of $I_1, \dots I_k$. Suppose that V is made of $n_1, \dots, n_k$ vertices of colors $1, \dots, k$ respectively, inducing a partition $V_1, \dots, V_k$. Construct a graph $\tilde{G}$ containing G as follows. First define $\tilde{V}_1, \dots, \tilde{V}_k$ by adding $n - n_i$ auxiliary points to V_i for i from 1 to k. Define $\tilde{G} = (\tilde{V}, \tilde{E})$ as the complete k-uniform k-partite hypergraph induced by $\tilde{V}_1, \dots, \tilde{V}_k$. The graph $\tilde{G}$ inherits the weight function from G in such a way that the auxiliary points do not contribute to any weight. More precisely the weight of $\tilde{e} \in \tilde{E}$ equals the weight of $e = \tilde{e} \cap V$ if $e \in E$ (meaning that $\tilde{e}$ has two or more vertices in V) and equals 1 if $e \notin E$ (meaning that $\tilde{e}$ has zero or one vertex in V). Each partial matching ρ of G can then be seen as a complete matching $\tilde{\rho}$ in $\tilde{G}$ by adding auxiliary points to the edges (and to the singletons) of ρ in such a way that each resulting hyperedge has exactly k vertices. Note that the resulting $\tilde{\rho}$ (which is not unique) has the same weight of ρ. Finally, noting that a complete matching in $\tilde{G}$ can be interpreted as assignment $A \subseteq I_1 \times \cdots \times I_k$ with $I_i = \tilde{V}_i$ for i from 1 to k, we obtain Problem 1.

When $k = 2$ Problem 1 becomes the classical (2-dimensional) optimal assignment problem. Such a problem is efficiently solvable, for example in $O(n^3)$ steps using the Hungarian Algorithm (Kuhn, 1955), which is based on concepts from Optimal Transportation Theory (Villani, 2009).

In contrast for $k \geq 3$ Problem 1 is an NP-hard optimization problem. Even more, unless P=NP, there is no deterministic polynomial-time approximation algorithm for a general cost function, meaning that the problem is not in the class APX (see Ausiello *et al.* , 2012, Def.3.9 for a definition of APX). The same holds even if the cost function C is decomposable as $C(x_1, \dots, x_k) = \sum_{i \neq j} d(x_i, x_j)$. Some polynomial time approximation algorithms exist if d satisfies the triangle inequality (see, for example, Crama & Spieksma (1992) and Bandelt *et al.* , 1994) but this is not our case. Balas & Saltzman (1991) propose an heuristic algorithm for a general cost function C, but no constant of approximation is provided and only the case $k = 3$ is considered.

Finally De la Vega *et al.* (2003) propose a polynomial time approximation scheme to partition n points of $\mathbb{R}^d$ in m clusters that minimize the sum of the intra-clusters squared Euclidean distances. This problem is similar to ours but unfortunately the running time of their algorithm is polynomial in n but exponential in m and in our context it seems reasonable to suppose m to be roughly of the same order of n.

In conclusion the literature does not appear to provide a generic bounded-complexity method to obtain (or approximate) ρ_{max}. Heuristic methods could still work well in our particular case.

4.2.4 Approximate sampling

We now consider the problem of drawing approximate samples from $\hat{\pi}(\rho)$. By approximate samples we mean samples drawn from some distribution close to $\hat{\pi}(\rho)$, for example in total variation distance (see (5.2) for definition). Such a task is often accomplished using Monte Carlo methods, for example Markov Chain Monte Carlo (MCMC) algorithms, which we describe more in details in Chapter 5.

Being able to draw approximate samples from $\hat{\pi}(\rho)$ is an important task as it allows us to approximate expectations with respect to $\hat{\pi}(\rho)$. See Chapter 5 for more details.

Two-color case

When k equals 2, sampling from $\hat{\pi}(\rho)$ can be related to monomer-dimer systems. In Statistical Physics a monomer-dimer system is a collection of n sites covered by molecules occupying one site (monomers) or two sites (dimers), which can be described with the following model.

Model 1. *(monomer-dimer system)*
Instance: A simple graph $G = (V, E)$ with non negative edge weights $w : E \to [0, \infty)$ such that $w(e) > 0$ for at least one $e \in E$.
State space: the set $\mathcal{M}_G$ of matchings contained in G.
Probability distribution: $\hat{\pi}(\rho) \propto \prod_{e \in \rho} w(e)$.

Although monomer-dimer systems are usually considered in lattice frameworks, the two-dimensional version of our model is indeed the monomer-dimer system corresponding to the weighted complete bipartite graph $G_{\mathbf{x}}$, see (4.4). Jerrum & Sinclair (1996) propose a Metropolis-Hastings (MH) random walk algorithm (see Chapter 5) to obtain approximate samples from monomer-dimer systems distributions in polynomial time. Using a canonical paths argument they prove that for any starting state ρ_0 the mixing time of their Markov Chain satisfy

$$\tau_{\rho_0}(\epsilon) \leq 4(\#E)(\#V)w'^2 \left(\log(\#E)\#E + \log\left(\epsilon^{-1}\right)\right), \tag{4.8}$$

where $w' = \max\{1, \max_{e \in E} w(e)\}$. Here the mixing time $\tau_{\rho_0}(\epsilon)$ can be thought as the number of steps needed by the algorithm to draw a sample from a distribution close to $\hat{\pi}(\rho)$ (namely whose distance is smaller than ϵ in total variation distance). See Section 5.1.1 for more details. Huber & Law (2012) consider the same Markov Chain starting from the mode ρ_{max} (which can be found in $O(\#V)^3$ by the Hun-

garian algorithm) and slightly improve the bound (4.8) to

$$\tau_{\rho_{max}}(\epsilon) \leq 4(\#E)(\#V)w'^2 \left(\log(2)\#E + \log\left(\epsilon^{-1}\right)\right). \tag{4.9}$$

Remark 10. *The bounds* (4.8) *and* (4.9) *seem to be very conservative in practice. In fact these bounds are often not tight enough to be used in applications. For example in the framework of Section 6.1.3 the bound in* (4.9) *is of order* 10^9 *(depending weakly on* ϵ*). Convergence diagnostic methods, though, suggest that order* 10^5 *steps are enough to approximate* $\hat{\pi}(\rho)$*. The apparent slackness of* (4.8) *and* (4.9) *is coherent with the fact that bounds obtained using canonical path techniques are typically over-conservative (see e.g. Jerrum & Sinclair, 1996).*

k-color case

Can we approximately sample from $\hat{\pi}(\rho)$ in polynomial time for $k \geq 3$ too? This is related to approximate count matchings in hypergraphs in polynomial time (see Chapter 3 of Jerrum (2003) for the relationship between approximate sampling and approximate counting). Unfortunately, as far as we are aware, there are not many results in this field. Karpinski *et al.* (2012) try to extend the methods of Jerrum & Sinclair (1996) to a hypergraph setting but they managed to do it only for a specific class of sparse hypergraphs that do not include our case. They also prove a negative result: unless NP=RP (RP is the analogous of P but for randomized decision algorithms), there cannot be any FPRAS (Fully Polynomial Random Approximation Scheme, see for example Jerrum, 2003, Sec. 3.1) to obtain approximate samples from the k-dimensional version of the monomer-dimer system for $k \geq 6$ (see Proposition 3 of Karpinski *et al.* , 2012). Strictly speaking, this still does not imply that such a scheme cannot exist for our problem even if NP≠RP, because our problem is constrained by additional conditions that they do not assume in their negative result (e.g. our hypergraph is k-partite).

4.2.5 Summary of intractability situation

Assuming P≠NP, we should not expect to perform exact posterior inferences in polynomial time. In fact finding the normalizing constant of $\hat{\pi}(\rho)$ is NP-hard even for $k = 2$ and the posterior mode ρ_{max} can be efficiently computed for $k = 2$, but it is an NP-hard problem for $k \geq 3$ (although heuristic algorithms exist).

Polynomial-time algorithms to sample from $\hat{\pi}(\rho)$ exist for $k = 2$ (although they are not practically feasible), while some results suggest that, unless P=NP, they cannot exist for $k \geq 6$ (see Section 4.2.4). Theoretical results of this kind do

not rule out the possibility of obtaining approximate samples in specific situations, but do exclude the possibility of finding a scheme that does so (in polynomial time) for arbitrary instances of a certain class of distributions. Since the problem we consider is by no means arbitrary it is feasible that special methods may produce good approximate samples.

In Chapter 6 we propose an MCMC algorithm for the two-color case and one for the k-color case. As a consequence of the results presented in this Section it is clear that additional care must be taken when empirically studying MCMC mixing properties.

Chapter 5

Theory and methodology from MCMC

In Chapter 3 we defined a Bayesian Random Partition Model to study complementary clustering of Anglo-Saxon settlements (see Chapter 2). The resulting posterior distribution (see Section 3.6) is intractable. The difficult part is the full conditional distribution of the partition ρ, namely $\pi(\rho \,|\, \mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$, which we denote by $\hat{\pi}(\rho)$. The theoretical results presented in Chapter 4 suggest that we have little hope of being able to perform exact inferences regarding $\hat{\pi}(\rho)$, such as evaluating posterior probabilities or expectations. However we might still obtained reliable approximate inferences. In particular, Monte Carlo methods constitute a broad and powerful class of tools to perform such approximate inferences.

In this chapter we first provide the relevant background on Monte Carlo methods and in particular Markov chain Monte Carlo (MCMC) algorithms (Section 5.1) and the Metropolis-Hastings (MH) algorithm (Section 5.1.1). For a more general and detailed discussion of Monte Carlo methods we refer to Robert & Casella (2005). Then we consider the question of what is an appropriate way to design informed MH proposal distributions in discrete spaces, such as the state space of $\hat{\pi}(\rho)$. To provide an answer, we introduce a class of informed proposals, which we call balanced proposals, and we show that under some assumptions they are optimal in an asymptotic Peskun sense (see Theorems 4, 5 and 6). Such a class of proposals will be directly relevant to our applied problem as they will significantly speed up of the algorithm we will use to analyze our dataset (see Section 6.1.3).

5.1 Markov chain Monte Carlo algorithms

The basic idea of Monte Carlo methods is to approximate a deterministic quantity by expressing it as the expectation of some random variable X and then estimate it with the empirical average of samples of X.

More precisely suppose we can express the quantity of interest as

$$\mathbb{E}_\pi [h] = \int_\Omega h(x)\pi(dx) < \infty$$

for some probability measure π on a sample space Ω and some function $h : \Omega \to \mathbb{R}$. The basic Monte Carlo approach consist in estimating $\mathbb{E}_\pi [h]$ with

$$\hat{h}_N \;=\; \frac{\sum_{i=1}^N h(X_i)}{N} \quad \xrightarrow{a.s.} \quad \mathbb{E}_\pi [h] \tag{5.1}$$

where $X_1, \ldots, X_N$ are independent and identically distributed (i.i.d) samples from π. The almost sure convergence as $N \to \infty$ in (5.1) follows from the Strong Law of Large Numbers. If in addition $\mathbb{E}_\pi \left[h^2\right]$ is finite, the Central Limit Theorem implies that the Monte Carlo error $|\hat{h}_N - \mathbb{E}_\pi [h]|$ goes to 0 at rate $O(\frac{1}{\sqrt{N}})$.

In practice, however, we often seek to estimate $\mathbb{E}_\pi [h]$ in cases where we cannot sample directly from π, such as our distribution of interest $\hat{\pi}(\rho)$. In such cases there are various possible approaches. For example Importance Sampling methods draw samples from a distribution ν which is not too far from π and then correct for the difference between ν and π. The approach we will use for our applied problem is to draw approximate samples from π using Markov chain Monte Carlo (MCMC) algorithms, which are powerful and general tools to draw approximate samples form an arbitrary distribution π. The main idea is to use the states visited by an ergodic Markov chain which admits π as a stationary distribution as approximate samples from π.

Theoretical background

In this section we present very briefly the main theoretical results underlying the MCMC methodology. This will be useful in the rest of the chapter. For simplicity we suppose that the state space of π is a finite set Ω. This assumption includes the case of our distribution of interest $\hat{\pi}(\rho)$ and simplifies the discussion from the technical point of view, without affecting the aim of this chapter. For a discussion of theoretical results for MCMC algorithms in general state spaces see, for example, Roberts & Rosenthal (2004).

The two main results underlying the MCMC methodology are the so-called Convergence Theorem and Ergodic Theorem for Markov chains (Theorems 1 and 2 respectively). To state such theorems we need some definitions. Let $P = \{P(x, \cdot)\}_{x \in \Omega}$ be a Markov transition kernel on Ω and $(X_t)_{t \geq 1}$ be the corresponding discrete time Markov chain. A Markov transition kernel P defined on Ω is

- *irreducible* if for any x and y in Ω there exist a $t \geq 1$ such that $P^t(x, y) > 0$;
- *aperiodic* if for any x in Ω the greatest common divisor of the set $\{t \geq 1 : P^t(x, x) > 0\}$ equals 1;
- π-*stationary* if it holds $\sum_{y \in \Omega} \pi(y) P(y, x) = \pi(x)$ for any x in Ω.

Finally we define the total variation distance between two probability measures μ and ν on Ω as

$$\|\mu - \nu\|_{TV} \quad = \quad \max_{A \subset \Omega} |\mu(A) - \nu(A)| \, . \tag{5.2}$$

Theorem 1. *(Convergence Theorem) Let P be an irreducible, aperiodic and π-stationary Markov transition kernel on Ω. Then there exist $C > 0$ and $\alpha \in (0, 1)$ such that*

$$\max_{x \in \Omega} \|P^t(x, \cdot) - \pi(\cdot)\|_{TV} \; \leq \; C\alpha^t \, .$$

Proof. See Theorem 4.9 of Levin *et al.* (2009). □

Motivated by the Convergence Theorem, a naive MCMC algorithm simulates N independent copies of $(X_t)_{t \geq 1}$ starting from some state x and collects the states of the N chains at some large time T. Such states are i.i.d. samples from $P^T(x, \cdot)$ and can be used to estimate $\mathbb{E}_{P^T(x,\cdot)}[h]$ for any function $h : \Omega \to \mathbb{R}$ as in (5.1). The Convergence Theorem guarantees that $P^T(x, \cdot)$ converges to π in total variation norm and therefore $\mathbb{E}_{P^T(x,\cdot)}[h] \to \mathbb{E}_\pi[h]$ for $T \to \infty$. Note that the starting state x could be replaced with any starting distribution on Ω. However, this algorithm is not efficient because it uses only the last state of the Markov chains that have been simulated. The Ergodic Theorem allows one to overcome this problem.

Theorem 2. *(Ergodic Theorem) Let $(X_t)_{t \geq 1}$ be an irreducible and π-stationary Markov chain on Ω (with arbitrary starting distribution). Then for any function $h : \Omega \to \mathbb{R}$ it holds*

$$\hat{h}_T \; = \; \frac{\sum_{t=1}^{T} h(X_t)}{T} \quad \xrightarrow{a.s.} \quad \mathbb{E}_\pi[h] \, . \tag{5.3}$$

Proof. See Theorem 4.16 of Levin *et al.* (2009). □

Equation (5.3) suggests a straightforward way to estimate $\mathbb{E}_\pi[h]$, which is to simulate an irreducible and π-stationary Markov chain $(X_t)_{t\geq 1}$ and then use $\frac{\sum_{t=1}^T h(X_t)}{T}$ to approximate $\mathbb{E}_\pi[h]$. This is indeed the general scheme underlying most MCMC algorithms.

5.1.1 The Metropolis-Hastings algorithm

In order to use the MCMC methodology, one needs to be able to simulate a π-stationary Markov chain (and then check that it is irreducible). The Metropolis-Hastings (MH) algorithm provides a way to turn a generic Markov transition kernel Q into a π-stationary kernel P. More precisely the MH algorithm produces a π-reversible kernel P, which means a kernel P such that

$$\pi(x)P(x,y) \;=\; \pi(y)P(y,x) \qquad \forall x,y \in \Omega\,, \tag{5.4}$$

which implies that P is π-stationary. The kernel P is obtained from Q by introducing the following accept/reject step. Given the current state x, the MH algorithm samples a proposed state y according to $Q(x,\cdot)$ and then accepts it with probability $a(x,y) = 1 \wedge \frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)}$ and rejects it otherwise (meaning that the chain stays at x). The resulting π-reversible transition kernel is

$$P(x,y) = \begin{cases} Q(x,y)a(x,y) & \text{if } y \neq x, \\ 1 - \sum_{z\neq x} Q(x,z)a(x,z) & \text{if } y = x\,. \end{cases}$$

The acceptance function $a(x,y)$

We now provide more details on the choice of the acceptance function $a(x,y) = 1 \wedge \frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)}$. This will be useful for the rest of the chapter. The acceptance function $a(x,y)$ needs to be chosen in such a way that P is π-reversible. This means $\pi(x)Q(x,y)a(x,y) = \pi(y)Q(y,x)a(y,x)$ for any x and y in Ω or, equivalently,

$$a(x,y) = a(y,x)\,t(x,y) \qquad \forall x,y:\, Q(x,y) > 0\,, \tag{5.5}$$

where $t(x,y)$ denotes the ratio $\frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)}$.

Since $a(x,y)$ and $a(y,x)$ are probabilities they must be no greater than 1. It follows that $a(x,y) \leq 1$ and $a(y,x) = \frac{a(x,y)}{t(x,y)} \leq 1$ (which implies $a(x,y) \leq t(x,y)$) and thus

$$a(x,y) \leq 1 \wedge t(x,y) \qquad \forall x,y:\, Q(x,y) > 0\,, \tag{5.6}$$

where $s \wedge t$ denotes $\min\{s,t\}$. Any choice of $\{a(x,y)\}_{x,y\in\Omega}$ satisfying (5.5) and (5.6)

would lead to a valid algorithm generating a π-reversible kernel P. Nevertheless it is desirable for $a(x, y)$ to be as big as possible, because rejecting moves reduces the efficiency of the algorithm (see Section 5.1.3). Therefore the optimal choice of $a(x, y)$ is the *Metropolis-Hastings acceptance function*

$$a_{MH}(x, y) \quad = \quad 1 \wedge t(x, y) \quad = \quad 1 \wedge \frac{\pi(y)Q(y, x)}{\pi(x)Q(x, y)} \,, \tag{5.7}$$

which satisfies (5.5) and achieves the upper bound in (5.6). Sometimes also the *Barker acceptance function* is used, which is defined as

$$a_B(x, y) \quad = \quad \frac{t(x, y)}{1 + t(x, y)} \quad = \quad \frac{\pi(y)Q(y, x)}{\pi(x)Q(x, y) + \pi(y)Q(y, x)} \,. \tag{5.8}$$

Remark 11. *Note that $a_{MH}(x, y) = g_{MH}(t(x, y))$ and $a_B(x, y) = g_B(t(x, y))$ with $g_{MH}(t) = 1 \wedge t$ and $g_B(t) = \frac{t}{1+t}$. More generally any acceptance function of the form $a(x, y) = g(t(x, y))$ is a valid acceptance function if and only if $g : \mathbb{R}_+ \to \mathbb{R}_+$ satisfies $g(t) \leq 1$ and*

$$g(t) \quad = \quad t\, g\,(1/t) \qquad t > 0\,.$$

5.1.2 Measuring the efficiency of MCMC algorithms

MCMC algorithms produce samples X_1, X_2, ... that are neither independent nor identically distributed. This can reduce the efficiency of the algorithm by slowing down the convergence of the estimator $\hat{h}_T$ to $\mathbb{E}_\pi[h]$ given by (5.3). In particular the performances of MH algorithms depend crucially on the proposal distribution Q. In this section we review some measures of efficiency for MCMC algorithms in order to provide guidelines to choose a good proposal Q. Such results will be helpful in the rest of the chapter.

Asymptotic variance

The notion of *asymptotic variance* describes how the correlation among MCMC samples affects the variance of the estimator $\hat{h}_T$. Given a π-stationary transition kernel P and a function $h : \Omega \to \mathbb{R}$, the asymptotic variance $\mathrm{var}_\pi(h, P)$ is defined as

$$\mathrm{var}_\pi(h, P) \;=\; \lim_{T\to\infty} T \,\mathrm{var}\left(\hat{h}_T\right) \;=\; \lim_{T\to\infty} T^{-1} \,\mathrm{var}\left(\sum_{t=1}^{T} h(X_t)\right) \,, \tag{5.9}$$

where $X_1,\ X_2,\ \ldots$ is a Markov chain in stationarity (i.e. with $X_1 \sim \pi$) evolving according to P. The smaller $\mathrm{var}_\pi(h, P)$ is, the more efficient the corresponding MCMC algorithm is in estimating $\mathbb{E}_\pi[h]$. The asymptotic variance $var_\pi(h, P)$ can also be expressed as

$$\mathrm{var}_\pi(h, P) \;=\; \mathrm{var}_\pi(h) \left(1 + 2\sum_{i=2}^{\infty} \mathrm{corr}(h(X_1), h(X_i))\right) , \tag{5.10}$$

where $var_\pi(h) = \mathrm{var}(h(X_1)) = \mathbb{E}_\pi[(h - \mathbb{E}_\pi[h])^2]$ and $\mathrm{corr}(h(X_1), h(X_i))$ denotes the correlation between $h(X_1)$ and $h(X_i)$. See for example Kypraios (2007, Sec.1.10) for calculations leading to (5.10). Motivated by (5.10) the *Integrated Autocorrelation Time* (IAT) is defined as

$$\mathrm{IAT}_\pi(h, P) \;=\; 1 + 2\sum_{i=2}^{\infty} \mathrm{corr}(h(X_1), h(X_i)) \,. \tag{5.11}$$

The value of $\mathrm{IAT}_\pi(h, P)$ represents how many MCMC samples are equivalent to one i.i.d. sample in terms of estimating $\mathbb{E}_\pi[h]$. For example if $\hat{h}_T$ was produced with i.i.d. samples then $\mathrm{var}\big(\hat{h}_T\big) = \mathrm{var}_\pi(h)/T$ and therefore $\mathrm{var}_\pi(h, P) = \mathrm{var}_\pi(h)$ and $\mathrm{IAT}_\pi(h, P) = 1$. Sometimes it can be useful the define the Effective Sample Size of an MCMC estimator $\hat{h}_T$, denoted by $\mathrm{ESS}_\pi(h, P)$, as T divided by $\mathrm{IAT}_\pi(h, P)$. In this way, for large T, it holds the intuitive equation

$$\mathrm{var}\big(\hat{h}_T\big) \approx \frac{\mathrm{var}_\pi(h)}{\mathrm{ESS}_\pi(h, P)} \,.$$

These three measures of efficiency ($\mathrm{var}_\pi(h, P)$, $\mathrm{IAT}_\pi(h, P)$ and $\mathrm{ESS}_\pi(h, P)$) are closely related and all focused on the MCMC autocorrelation. Summarizing, the smaller $\mathrm{var}_\pi(h, P)$ and $\mathrm{IAT}_\pi(h, P)$ are (or equivalently the bigger $\mathrm{ESS}_\pi(h, P)$ is) the more efficient the Markov chain P is in estimating $\mathbb{E}_\pi[h]$.

Speed of convergence to stationarity

In practice, however, the starting distribution of an MCMC algorithm is not π itself but some other distribution, often a fixed state x in Ω. Therefore one should choose a kernel P such that $P^T(x, \cdot)$ converges as quickly as possible to π in order to reduce the bias introduced by the starting state x and thus speed-up the convergence of $\hat{h}_T$ to $\mathbb{E}_\pi[h]$ given by (5.3). The Convergence Theorem (see Theorem 1) states that $d(t) = \max_{x\in\Omega} \|P^t(x, \cdot) - \pi(\cdot)\|_{TV}$ satisfies $d(t) \leq C\alpha^t$ for some $C > 0$ and $\alpha \in (0, 1)$. Indeed the rate at which $d(t)$ goes to 0 is controlled by the second largest eigenvalue

of P in absolute value, say λ_*, meaning that

$$\lim_{t\to\infty} d(t)^{1/t} = \lambda_*$$

(see Levin *et al.* , 2009, Thm.12.6). Therefore we would like λ_* to be small or, equivalently, the absolute spectral gap of P

$$\text{AbsGap}(P) = 1 - \lambda_*$$

to be large. However, rather than $\text{AbsGap}(P)$, one often uses the spectral gap

$$\text{Gap}(P) = 1 - \lambda_2\,,$$

where λ_2 is the second largest eigenvalue of P, without taking the absolute value. Since MCMC samples are typically positively correlated the two notions are usually equivalent in practice. In particular if one considers the lazy version of P, obtained by moving according to P with probability a half and staying at the current location otherwise, then all the eigenvalues of P are positive and thus $\text{AbsGap}(P) = \text{Gap}(P)$.

While the (absolute) spectral gap of P controls only the rate α in the expression $d(t) \leq C\alpha^t$, the constant C could still be arbitrarily large. A more stringent notion of convergence is given by the notion of *mixing time*, which is defined as

$$t_{mix}(\varepsilon) \quad = \quad \inf\left\{t \geq 1 \,:\, d(t) \leq \varepsilon\right\}\,, \qquad \varepsilon \in (0,1)\,. \tag{5.12}$$

In summary, in order for a Markov chain to converge quickly to stationarity, it should have a large (absolute) spectral gap and a small mixing time.

Remark 12. *Although asymptotic variance and speed of convergence lead to different theoretical notions of efficiency for MCMC algorithms, the two are closely connected (see Mira (2001) for a detailed discussion) and in practice it is often the case that by improving the one also the other improves and vice versa.*

5.1.3 Peskun ordering

Peskun (1973) introduced the following partial ordering among Markov kernels that controls both the asymptotic variance and the spectral gap.

Theorem 3. *Let P_1 and P_2 be two π-reversible Markov transition kernels on a finite*

space Ω *such that* $P_1(x,y) \geq P_2(x,y)$ *for any* $x \neq y$*. Then it holds*

$$\begin{aligned} &(a) \quad var_\pi(h, P_1) \leq var_\pi(h, P_2) \qquad \forall h : \Omega \to \mathbb{R}\,, \\ &(b) \qquad Gap(P_1) \geq Gap(P_2)\,. \end{aligned}$$

Proof. Part (a): Peskun (1973, Thm.2.1.1) and Tierney (1998, Thm.4). (b) follows from the variational characterization of Gap (Levin *et al.* , 2009, Lemma 13.12). □

Theorem 3 implies that if $P_1(x,y) \geq P_2(x,y)$ for any $x \neq y$, then P_1 leads to a more efficient MCMC algorithm than P_2. We extend such an ordering to cases involving a constant in the inequality $P_1(x,y) \geq P_2(x,y)$, as we will need such an extension in Section 5.4.

Theorem 4. *Let* P_1 *and* P_2 *be two* π*-reversible and irreducible Markov transition kernels on a finite space* Ω *and* $c > 0$ *such that* $P_1(x,y) \geq c\, P_2(x,y)$ *for any* $x \neq y$*. Then it holds*

$$\begin{aligned} &(a) \quad var_\pi(h, P_1) \leq \frac{var_\pi(h, P_2)}{c} + \frac{1-c}{c} var_\pi(h) \qquad \forall h : \Omega \to \mathbb{R}\,, \\ &(b) \qquad Gap(P_1) \geq c \cdot Gap(P_2)\,. \end{aligned}$$

To prove part (a) of Theorem 4 we need the following Lemma.

Lemma 1. *Let* P *be a* π*-reversible and irreducible Markov transition kernels on* Ω*, with* $|\Omega| = n$*. Let* $\tilde{P} = c\, P_1 + (1-c)\mathbb{I}_n$*, where* $\mathbb{I}_n$ *is the* $n \times n$ *identity matrix and* $c \in (0,1]$*. Then it holds*

$$var_\pi(h, \tilde{P}) = \frac{var_\pi(h, P)}{c} + \frac{1-c}{c}\, var_\pi(h) \qquad \forall h : \Omega \to \mathbb{R}\,.$$

Proof of Lemma 1. Suppose $\mathbb{E}_\pi[h] = 0$ (otherwise consider $h - \mathbb{E}_\pi[h]$). Let $\{(\lambda_i, f_i)\}_{i=1}^n$ and $\{(\tilde{\lambda}_i, \tilde{f}_i)\}_{i=1}^n$ be the eigenvalues and eigenfunctions of P and $\tilde{P}$ respectively. Then it holds $\lambda_1 = \tilde{\lambda}_1 = 1$, $f_1 = \tilde{f}_1 = (1, \ldots, 1)^T$ and $-1 \leq \lambda_i, \tilde{\lambda}_i < 1$ for $i \geq 2$ (Levin *et al.* , 2009, Lemmas 12.1,12.2). The asymptotic variances can be written as

$$\mathrm{var}_\pi(h, P) = \sum_{i=2}^n \frac{1+\lambda_i}{1-\lambda_i} \mathbb{E}_\pi[h\, f_i]^2 \quad \text{and} \quad \mathrm{var}_\pi(h, \tilde{P}) = \sum_{i=2}^n \frac{1+\tilde{\lambda}_i}{1-\tilde{\lambda}_i} \mathbb{E}_\pi[h\, \tilde{f}_i]^2\,. \quad (5.13)$$

See for example the proofs of Mira (2001, Theorem 1) and Levin *et al.* , 2009, Lemmas 12.20 for (5.13). From the definition of $\tilde{P}$ it follows that $\tilde{f}_i = f_i$ and $\tilde{\lambda}_i = c\,\lambda_i + (1-c)$. Rearranging the latter equality we obtain $\frac{1+\tilde{\lambda}_i}{1-\tilde{\lambda}_i} = \frac{1}{c}\frac{1+\lambda_i}{1-\lambda_i} + \frac{1-c}{c}$

for $i \geq 2$. Thus

$$\mathrm{var}_\pi(h, \tilde{P}) = \sum_{i=2}^{n} \frac{1+\tilde{\lambda}_i}{1-\tilde{\lambda}_i} \mathbb{E}_\pi[h\, f_i]^2 = \frac{1}{c} \sum_{i=2}^{n} \frac{1+\lambda_i}{1-\lambda_i} \mathbb{E}_\pi[h\, f_i]^2 + \frac{1-c}{c} \sum_{i=2}^{n} \mathbb{E}_\pi[h\, f_i]^2 \,. \tag{5.14}$$

Since $\{f_i\}_{i=1}^n$ form an orthonormal basis of $L^2(\mathbb{R}^\Omega, \pi)$ and $\mathbb{E}_\pi[h\, f_1] = \mathbb{E}_\pi[h] = 0$, then $\sum_{i=2}^n \mathbb{E}_\pi[h\, f_i]^2 = \mathbb{E}_\pi[h^2] = \mathrm{var}_\pi(h)$. Therefore (5.14) becomes $\mathrm{var}_\pi(h, \tilde{P}_1) = \frac{1}{c} \cdot \mathrm{var}_\pi(h, P_1) + \frac{1-c}{c} \mathrm{var}_\pi(h)$. □

Proof of Theorem 4. Part (b) follows for example from Levin *et al.* (2009, Lemma 13.22). Part (a), case $c > 1$: define $\tilde{P}_1 = \frac{1}{c} P_1 + (1 - \frac{1}{c}) \mathbb{I}_n$. From Lemma 1 it follows $\mathrm{var}_\pi(h, \tilde{P}_1) = c \cdot \mathrm{var}_\pi(h, P_1) + (c-1)\, \mathrm{var}_\pi(h)$ or, equivalently, $\mathrm{var}_\pi(h, P_1) = \frac{1}{c} \mathrm{var}_\pi(h, \tilde{P}_1) + \frac{1-c}{c}\, \mathrm{var}_\pi(h)$. Since $\tilde{P}_1(x, y) \geq P_2(x, y)$ for $x \neq y$, by Theorem 3 it holds $\mathrm{var}_\pi(h, \tilde{P}_1) \leq \mathrm{var}_\pi(h, P_2)$. Therefore

$$\mathrm{var}_\pi(h, P_1) \;=\; \frac{\mathrm{var}_\pi(h, \tilde{P}_1)}{c} + \frac{1-c}{c}\, \mathrm{var}_\pi(h) \;\leq\; \frac{\mathrm{var}_\pi(h, P_2)}{c} + \frac{1-c}{c}\, \mathrm{var}_\pi(h)\,,$$

as desired. Part (a), case $c \leq 1$: define $\tilde{P}_2 = c\, P_2 + (1-c) \mathbb{I}_n$. From Lemma 1 it follows $\mathrm{var}_\pi(h, \tilde{P}_2) = \frac{1}{c} \mathrm{var}_\pi(h, P_2) + \frac{1-c}{c} \mathrm{var}_\pi(h)$. The latter equality and $\mathrm{var}_\pi(h, P_1) \leq \mathrm{var}_\pi(h, \tilde{P}_2)$, which follows from $P_1(x, y) \geq \tilde{P}_2(x, y)$ for $x \neq y$ and Theorem 3, provide us with part (a). □

Remark 13. *Note that the constant c introduced in the Peskun ordering translates directly in the spectral gap comparison: $Gap(P_1) \geq c \cdot Gap(P_2)$, Thm.4 part (b).*

On the other hand there is an additional term, $\frac{1-c}{c}\, var_\pi(h)$, appearing in the asymptotic variance comparison (Thm.4, part (a)). If $c > 1$ then $\frac{1-c}{c}\, var_\pi(h) \leq 0$ and thus $var_\pi(h, P_2) \leq \frac{var_\pi(h, P_2)}{c}$. If $c < 1$ then $var_\pi(h, P_2) \leq \frac{var_\pi(h, P_2)}{c}(1 + \frac{1-c}{ESS_\pi(h,P)})$. Note that typically $(1 + \frac{1-c}{ESS_\pi(h,P)}) \approx 1$ because $ESS_\pi(h, P)$ is much larger than 1 in practical applications. Although the additional term $\frac{1-c}{c}\, var_\pi(h)$ cannot be dropped, the leading term is usually $\frac{var_\pi(h, P_2)}{c}$.

5.2 Building informed proposal distributions

When designing a MH proposal distribution it would be desirable to propose global moves (i.e. $Q(x, y) > 0$ for any $(x, y) \in \Omega \times \Omega$), ideally sampling from the target π itself, $Q(x, y) = \pi(y)$. However, most of the times this is not computationally feasible and the typical choice is to implement a MH algorithm performing local moves. In most discrete frameworks, for example, the MH chain is only allowed to jump to states belonging to some neighborhood of the current state, meaning that

$Q(x,y) > 0$ for $y \in N(x)$ and $Q(x,y) = 0$ for $y \notin N(x)$ with $N(x)$ small compared to Ω. In such a framework the proposed state is often chosen uniformly at random from the neighbors

$$Q(x,y) \;=\; Q^U(x,y) \;=\; \begin{cases} \frac{1}{|N(x)|} & \text{if } y \in N(x), \\ 0 & \text{if } y \notin N(x)\,. \end{cases} \tag{5.15}$$

See for example Jerrum & Sinclair (1996) or the MC^3 algorithm of Madigan *et al.* (1995).

The uniform proposal in (5.15) is an *uninformed* proposal, meaning that no information from π is incorporated in the proposal. This is a practical choice because often it is easy to sample from Q^U. On the other hand, it is reasonable to expect that an *informed* proposal Q that incorporates local information from π would improve the mixing of the resulting Markov chain compared to the "blind" proposal Q^U.

Informed proposals in continuous spaces

In continuous frameworks, such as $\Omega = \mathbb{R}^n$ and $\pi(dx) = f(x)dx$, various *informed* MH schemes have been designed to improve upon the uninformed proposal $y \sim N(x, \sigma^2 \mathbb{I}_n)$. For example the Metropolis Adjusted Langevin Algorithm (MALA) uses the gradient of $\log f$ to move towards high probability regions by proposing a state according to $y \sim N(x + \frac{\sigma^2}{2}\nabla(\log f)(x), \sigma^2 \mathbb{I}_n)$. Such an algorithm is derived by discretizing the π-reversible Langevin diffusion X_t satisfying $dX_t = \frac{\sigma^2}{2}\nabla(\log f)(x)dt + \sigma dB_t$. MALA typically has better mixing properties compared to the Markov chain obtained from the random walk proposal $y \sim N(x, \sigma^2 \mathbb{I}_n)$ (see e.g. Roberts & Rosenthal, 1998), although MALA can sometimes lead to unstable behaviors.

Note that, in order to design an informed proposal, one should think carefully about *how* to appropriately incorporate the information in Q. For example, for the MALA proposal it is not enough to move towards high probability regions (by following the gradient of the target) but it is crucial that, given a normal proposal with variance σ^2, the mean is shifted by exactly $\frac{\sigma^2}{2}\nabla(\log f)(x)$. The intuitive reason is that, because this is the discretization of a π-reversible diffusion, the MALA proposal is "almost" π-reversible even before applying the MH correction and thus fewer moves are rejected and longer moves are allowed. Sections 5.3 and 5.4 are related to such remarks.

Various other informed proposal schemes for continuous frameworks have

been proposed in the literature (e.g. Neal (2011), Girolami & Calderhead (2011), Welling & Teh (2011), Durmus *et al.* (2015)) resulting in a substantial improvement of MCMC performances in practical applications, though few or no analogs can be found in discrete frameworks.

Towards informed proposals in discrete spaces

Informed proposal schemes for continuous frameworks (e.g. MALA) are typically derived as discretization of continuous time diffusion processes, and are based on derivatives and Gaussian distributions. Therefore it is not clear how to extend such methods to frameworks where Ω is a discrete space. In this chapter we aim at providing some theoretically justified indications on how to incorporate local information about π in MH proposal distributions in discrete frameworks. To do so we will make assumptions on the target measures which, although being theoretically restrictive, are satisfied in most applied scenarios (see Section 5.4).

As mentioned before, we consider MH chains that are only allowed to perform *local* moves: $Q(x, y) > 0$ for $y \in N(x)$ and $Q(x, y) = 0$ for $y \notin N(x)$ with $|N(x)| \ll |\Omega|$. Think, for example, of adding or removing one variable (or a few variables) in a variable selection context, or moving an edge (or a few edges) in a graphical model context. We assume that the neighboring structure $\{N(x)\}_{x\in\Omega}$ is fixed and we try to optimize over Q. Usually the specific problem under consideration suggests a natural neighboring structure and often, given the current state x, the probability $\pi(y)$ $\left(\text{or the ratio } \frac{\pi(y)}{\pi(x)}\right)$ can be efficiently evaluated for $y \in N(x)$. The question we consider here is how to incorporate the knowledge about $\{\pi(y)\}_{y\in N(x)}$ in the proposal $Q(x, \cdot)$. Somewhat surprisingly, proposing from the target itself restricted to $N(x)$, i.e. $Q(x, y) \propto \pi(y)\mathbb{1}_{N(x)}(y)$, is not the optimal choice (even if one could sample from it) and indeed not even close to the optimal one. Instead, our results suggest that informed proposals should resemble a transformation of the target measure, obtained through a balancing function g. The choice of g is not unique and we characterize the family of optimal choices (in terms of Peskun ordering) which we call *balanced proposals*.

These results may indicate on how to use local information about the target distribution to design more efficient MH proposal distributions. This work may also be a first step towards the extension of methods for continuous frameworks to discrete frameworks: in Section 5.6 we give some suggestions on how the two approaches may be related.

Organization of the rest of the Chapter

In Section 5.3 we introduce the idea of balanced proposals, motivating it with some heuristic calculations and demonstrating it on the two-color version of our model. In Section 5.4 we define the class of target measures and of proposals that we are going to consider and we prove that, in the asymptotic regime, balanced proposals are maximal elements in terms of Peskun ordering. Then we discuss the close connection between balancing functions and acceptance probability functions. In Section 5.5 we consider the natural consequent question: is there a best choice among various balanced proposals? We answer this question in a specific case (namely the hypercube with product-like measure) where we derive the explicit expression of the balanced proposal leading to the smallest mixing time. In Section 5.6 we explore the connection between balanced proposals and continuous space MCMC schemes using local informations (e.g. MALA) and discuss possible future works.

Remark 14. *To our knowledge the notion of balanced proposals is new to the literature. Note that there are many results on mixing times of Markov chains in discrete spaces (Levin* et al. *, 2009). While such literature usually considers uniform target distributions and focuses on proving explicit mixing times results, we are focusing on the choice of proposal given a* non-uniform π*.*

The problem we consider is also different from the optimal scaling problem for MH algorithms (Roberts et al. *, 1997), which deals with optimizing the scale of the proposal distribution (e.g. by tuning the variance). In fact we are fixing the scale of the proposal (by fixing* $\{N(x)\}_{x\in\Omega}$*) and then optimizing over the choice of* Q*.*

5.3 Heuristic calculations and illustrative example

We illustrate, through some heuristic calculations, the intuition as to why setting Q proportional to π is not appropriate when performing local moves and why it is necessary to introduce a balancing function g.

The difference between global and local moves

When performing global moves (i.e. $N(x) = \Omega$ for any x) the "ideal" proposal would be $Q(x, y) = \pi(y)$. The Metropolis-Hastings ratio would then be equal to one

$$a(x, y) \;=\; 1 \wedge \frac{\pi(y)Q(y, x)}{\pi(x)Q(x, y)} \;=\; 1 \wedge \frac{\pi(y)\pi(x)}{\pi(x)\pi(y)} = 1,$$

and the MCMC algorithm would reduce to i.i.d. Monte Carlo sampling.

Consider now the local move case where $N(x)$ is small compared to Ω. For simplicity suppose $\Omega = \mathbb{R}$ with $\pi(dx) = f(x)dx$, $Q(x, dy) = q(x, y)dy$ and $N(x) = B_R(x) = \{y \in \mathbb{R} : |x - y| < R\}$ for some small $R > 0$. Moreover suppose for illustrative purposes that $\log f$ is linear in a region around the current state x, meaning that

$$f(y) \;=\; d\, e^{c\, y} \qquad y \in (x - 2R, x + 2R)\,,$$

for some $c, d > 0$. Motivated by the global case we could propose according to the target distribution restricted to $N(x)$,

$$q(x, y) = f(y)\frac{\mathbb{1}_{N(x)}(y)}{\int_{N(x)} f(z)dz} = \frac{f(y)}{f(x)}\frac{\mathbb{1}_{N(x)}(y)}{\int_{N(x)} \frac{f(z)}{f(x)}dz} = \frac{f(y)}{f(x)}\frac{\mathbb{1}_{N(x)}(y)}{Z_R}, \tag{5.16}$$

where $Z_R = \int_{-R}^{R} e^{c\, z}dz$ is a constant non depending on x. Similarly for $q(y, x)$. The Metropolis-Hastings ratio for $y \in N(x)$ would be

$$a(x, y) \;=\; 1 \wedge \frac{f(y)q(y, x)}{f(x)q(x, y)} \;=\; 1 \wedge \frac{f(y)\frac{f(x)}{f(y)}\frac{1}{Z_R}}{f(x)\frac{f(y)}{f(x)}\frac{1}{Z_R}} \;=\; 1 \wedge \frac{f(x)}{f(y)} \neq 1. \tag{5.17}$$

We modify the proposal (5.16) according to a balancing function $g : \mathbb{R}_+ \to \mathbb{R}_+$ as follows

$$q_g(x, y) = g\left(\frac{f(y)}{f(x)}\right)\frac{\mathbb{1}_{N(x)}(y)}{\int_{N(x)} g\left(\frac{f(z)}{f(x)}\right)dz} = g\left(\frac{f(y)}{f(x)}\right)\frac{\mathbb{1}_{N(x)}(y)}{Z_R^{(g)}}, \tag{5.18}$$

where $Z_R^{(g)} = \int_{-R}^{R} g\left(e^{c\, z}\right)dz$ is a constant non depending on x. Then the Metropolis-Hastings ratio for $y \in N(x)$ becomes

$$a_g(x, y) \;=\; 1 \wedge \frac{f(y)q_g(y, x)}{f(x)q_g(x, y)} \;=\; 1 \wedge \frac{f(y)g\left(\frac{f(x)}{f(y)}\right)\frac{1}{Z_R^{(g)}}}{f(x)g\left(\frac{f(y)}{f(x)}\right)\frac{1}{Z_R^{(g)}}} \;=\; 1 \wedge \frac{\frac{f(y)}{f(x)}g\left(\frac{f(x)}{f(y)}\right)}{g\left(\frac{f(y)}{f(x)}\right)}. \tag{5.19}$$

If g satisfies

$$g(t) = t\, g(1/t), \qquad t > 0\,, \tag{5.20}$$

then the MH ratio in (5.19) satisfies

$$a_g(x, y) \;=\; 1\,. \tag{5.21}$$

Equation (5.21) means that Q is already π-reversible, $\pi(x)q(x, y) = \pi(y)q(y, x)$,

before applying the MH correction. This would allow for more moves to be accepted and longer moves to be performed. Note that $a_g(x, y)$ equals 1 because we assumed $\log f$ to be linear in a neighborhood of x. If this was not the case, then $a_g(x, y)$ would not be equal to 1. Nevertheless, if $\log f$ is smooth enough then it can be approximated with its first order Taylor expansion in a neighborhood of x and $a_g(x, y)$ would be still close to 1. Therefore Q would be "approximately" π-reversible, in the same way as a MALA proposal, being the discretization of a π-reversible diffusion, is approximately π-reversible (Section 5.2).

We will refer to functions g satisfying (5.20) as *balancing functions*. These illustrative calculations provide some intuition motivating the use of balancing function g. In Section 5.4, in order to obtain rigorous results, we will consider the asymptotic regime and introduce some smoothness assumptions on the target measure.

5.3.1 Example: sampling perfect matchings from bipartite graphs

Before proving theoretical results, we demonstrate the improvements given by balancing functions in sampling from the distribution $\hat{\pi}(\rho)$ considered in Chapter 4. For simplicity we consider the two-color version of $\hat{\pi}(\rho)$. This is equivalent to sampling matchings from a weighted bipartite graph (see Section 4.1.1). For simplicity, and motivated by Section 4.2.3, we restrict our attention to perfect matchings.

More formally, let $G = (V, E)$ be a weighted bipartite graph (Section 4.1.1) with n red vertices, n blue vertices and strictly positive weights. We denote the edge connecting the i-th red vertex and the j-th blue vertex by (i, j) and its weight by w_{ij}. The state space Ω is the space of perfect matchings contained in G. There is an natural bijection between such Ω and the space of permutations of n elements $\mathcal{S}_n$: a permutation $\rho \in \mathcal{S}_n$ represents the perfect matching with edges $\{(i, \rho(i))\}_{i=1}^n$. The target measure under consideration is

$$\pi(\rho) = \frac{\prod_{i=1}^n w_{i\rho(i)}}{Z} \qquad \rho \in \mathcal{S}_n \,,$$

where Z is the normalizing constant $\sum_{\rho \in \mathcal{S}_n} \prod_{i=1}^n w_{i\rho(i)}$. We drop the hat over $\hat{\pi}(\rho)$ and use simply $\pi(\rho)$ to be consistent with the notation of this chapter.

We want to construct a Metropolis-Hastings algorithm targeting $\pi(\rho)$. As is common, we consider local moves that pick two indices and switch them. The induced neighboring structure is given by

$$N(\rho) \;=\; \left\{\rho' \in \mathcal{S}_n \,:\, \rho' = \rho \circ (i, j) \text{ for some } i, j \in \{1, \ldots, n\} \text{ with } i \neq j\right\} \,, \quad (5.22)$$

where $\rho' = \rho \circ (i,j)$ is defined by $\rho'(i) = \rho(j)$, $\rho'(j) = \rho(i)$ and $\rho'(l) = \rho(l)$ for l different from i and j. Note that, if $\rho' \in N(\rho)$, the ratio $\frac{\pi(\rho')}{\pi(\rho)}$ is easy to evaluate because $\frac{\pi(\rho\circ(i,j))}{\pi(\rho)} = \frac{w_{i\rho(j)}w_{j\rho(i)}}{w_{i\rho(i)}w_{j\rho(j)}}$.

We compare three proposal distributions, which we denote by Q^U, Q^T and Q^B. All such proposals use the same neighboring structure $\{N(\rho)\}_{\rho\in\mathcal{S}_n}$, defined in (5.22), and can be written as $Q_g(\rho,\rho') \propto g\left(\frac{\pi(\rho')}{\pi(\rho)}\right)\mathbb{1}_{N(\rho)}(\rho')$ for some $g:\mathbb{R}_+\to\mathbb{R}_+$.

The proposal Q^U is an uninformed, uniform proposal

$$Q^U(\rho,\rho') \;=\; \begin{cases} \frac{1}{n(n-1)} & \text{if } \rho' = \rho\circ(i,j)\,, \\ 0 & \text{otherwise}\,, \end{cases} \tag{5.23}$$

and corresponds to Q_g with $g(t)\equiv 1$. The second proposal, Q^T, is an informed proposal but the information about $\{\pi(\rho')\}_{\rho'\in N(\rho)}$ is used in a naive way, meaning that the proposal coincides with the target itself restricted to the neighborhood

$$Q^T(\rho,\rho') \;\propto\; \pi(\rho')\mathbb{1}_{N(\rho)}(\rho') \;\propto\; \begin{cases} \frac{w_{i\rho(j)}w_{j\rho(i)}}{w_{i\rho(i)}w_{j\rho(j)}} & \text{if } \rho' = \rho\circ(i,j)\,, \\ 0 & \text{if } \rho'\notin N(\rho)\,, \end{cases} \tag{5.24}$$

and corresponds to Q_g with $g(t)\equiv t$. Finally we consider an informed proposal where the information is incorporated using a balancing function, namely $g(t)=\sqrt{t}$

$$Q^B(\rho,\rho') \;\propto\; \sqrt{\pi(\rho')}\mathbb{1}_{N(\rho)}(\rho') \;\propto\; \begin{cases} \sqrt{\frac{w_{i\rho(j)}w_{j\rho(i)}}{w_{i\rho(i)}w_{j\rho(j)}}} & \text{if } \rho' = \rho\circ(i,j), \\ 0 & \text{if } \rho'\notin N(\rho)\,. \end{cases} \tag{5.25}$$

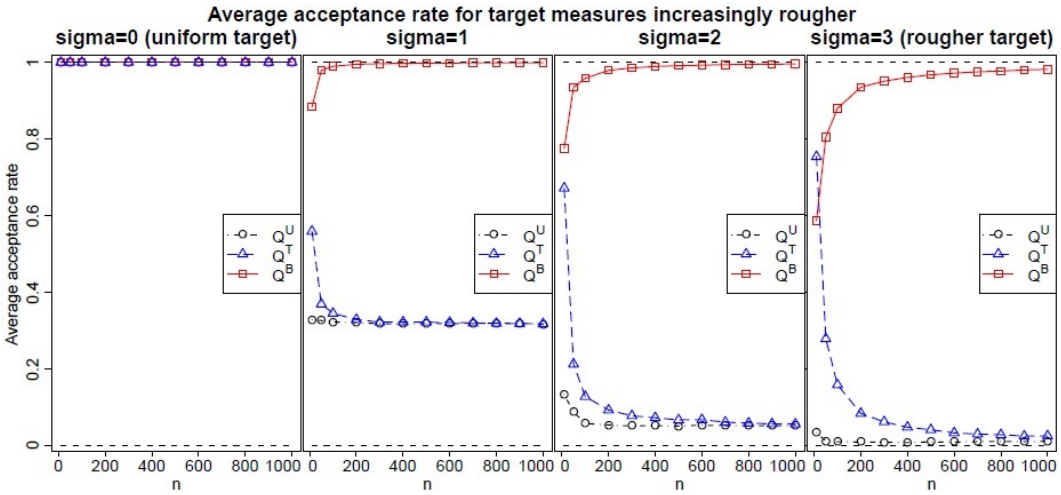


Figure 5.1: Average acceptance rates of the MH algorithm with proposals Q^U, Q^T and Q^B for different values of n and σ.

In order to compare these proposals we consider the following set up. The weights $\{w_{ij}\}_{i,j=1}^{n}$ are i.i.d. with $\log(w_{ij}) \sim N(0, \sigma^2)$. We consider different values of n and σ to vary the dimension of the sample space and the smoothness of the target distribution, respectively. In fact when σ equals 0 the target measure π is uniform and as σ increases π becomes more rough. Figure 5.1 shows the MH average acceptance rate for different values of σ and n. Such a figure suggests that in the asymptotic regime (i.e. for $n \to \infty$) the MH average acceptance rate for Q^B converges to 1, while those of Q^U and Q^T converge to a value depending on σ that goes to 0 as σ increases. Figure 5.2 shows some traceplots and convergence

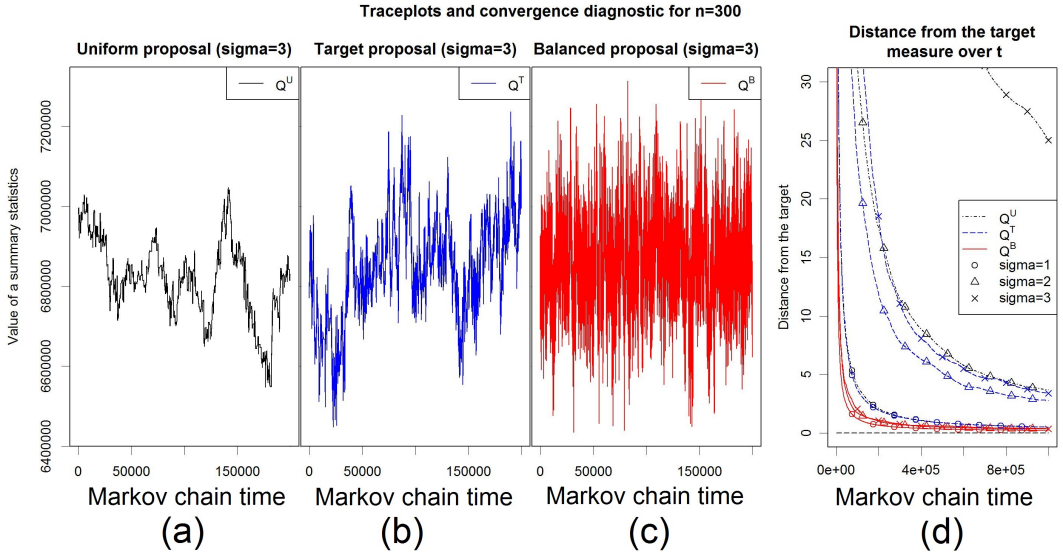


Figure 5.2: (a)-(c) Traceplots of a summary statistic defined as $S(\rho) = \sum_{i=1}^{n} i\rho(i)$. (d) Distance from the target measure defined as $D(T) = \sum_{i,j=1}^{n} (\hat{p}_{ij}^{(T)} - p_{ij})^2$, where p_{ij} is the probability of $\{(i,j) \in \rho\}$ under $\pi(\rho)$ (estimated with a long MCMC run independently from the rest) and $\hat{p}_{ij}^{(T)} = \sum_{t=1}^{T} \mathbb{1}((i,j) \in \rho_t)/T$, with $(\rho_t)_{t=1}^{T}$ being the samples from the MCMC algorithm under consideration.

diagnostics. Figure 5.2 suggests that Q^B is mixing faster than Q^U and Q^T. Moreover Figure 5.2(d) indicates that the difference between the two proposals increases as σ increases. Interestingly, Q^B seems to be robust to an increase in roughness of the target (i.e. σ increasing), see again Figure 5.2(d).

5.4 Peskun Ordering result

As before, we consider proposal distributions which can be written as

$$Q_g(x,y) \quad = \quad \frac{1}{Z_g(x)} \begin{cases} g\left(\frac{\pi(y)}{\pi(x)}\right) & \text{if } y \in N(x), \\ 0 & \text{otherwise,} \end{cases} \tag{5.26}$$

for some $g : \mathbb{R}^+ \to \mathbb{R}^+$, where $Z_g(x) = \sum_{z \in N(x)} g\left(\frac{\pi(z)}{\pi(x)}\right)$ is a normalizing constant. Section 5.3 suggests that the optimal choice of g should belong to the following class.

Definition 1 (Balanced proposals)**.** *A function $g : \mathbb{R}_+ \to \mathbb{R}_+$ satisfying*

$$g(t) \;=\; t\, g(1/t) \qquad\qquad t > 0\,, \tag{5.27}$$

is called a balancing function *and the corresponding Q_g is called a* balanced proposal*.*

We now show that, under some regularity assumptions on π, balanced proposals are the asymptotically maximal elements, in terms of Peskun ordering, among the proposal of type (5.26). First we define

$$c_g \quad = \quad \sup_{x \in \Omega,\, y \in N(x)} \frac{Z_g(y)}{Z_g(x)} \;\geq\; 1\,. \tag{5.28}$$

The constant c_g satisfies $c_g \geq 1$ and represents the roughness of π. In particular c_g is related to how non-linear $\log \pi$ is with respect to the neighboring structure $\{N(x)\}_{x \in \Omega}$. For example, if Ω equals $\mathbb{Z}^n$ or $\mathbb{R}^n$, then $\frac{Z_g(x)}{Z_g(y)}$ would be equal to 1 in regions where $\log \pi$ is linear, see e.g. (5.16), and thus c_g would be 1. The relevance of c_g for our framework comes from the following theorem.

Theorem 5. *Let $g : \mathbb{R}_+ \to \mathbb{R}_+$. Define $\tilde{g}(t) = \min\{g(t), t\, g(1/t)\}$ and let P_g and $P_{\tilde{g}}$ be the Metropolis-Hastings transition kernels obtained from the proposals Q_g and $Q_{\tilde{g}}$ respectively (see* (5.26) *for definition). Then, given $c = c_g c_{\tilde{g}}$, it holds*

$$\begin{aligned} &(a) \quad P_{\tilde{g}}(x,y) \quad \geq \quad \frac{P_g(x,y)}{c} \qquad \forall x, y \in \Omega\,, \\ &(b) \quad var_\pi(h, P_{\tilde{g}}) \;\leq\; c\, var_\pi(h, P_g) + (c-1)\, var_\pi(h) \qquad \forall h : \Omega \to \mathbb{R}\,, \\ &(c) \qquad Gap(P_{\tilde{g}}) \;\geq\; \frac{Gap(P_g)}{c}\,. \end{aligned} \tag{5.29}$$

Proof. If $y \notin N(x)$ then (5.29) holds trivially. Suppose $y \in N(x)$ and denote for

brevity $t = \frac{\pi(y)}{\pi(x)}$. Then

$$P_g(x,y) \;=\; \frac{g(t)}{Z_g(x)}\,\min\left\{1\,,\,t\,\frac{g(1/t)}{Z_g(y)}\frac{Z_g(x)}{g(t)}\right\} \;=\; \min\left\{\frac{g(t)}{Z_g(x)}\,,\,\frac{tg(1/t)}{Z_g(y)}\right\} \;\leq$$
$$c_g\frac{\min\{g(t),t\,g(1/t)\}}{Z_g(x)} \;=\; c_g\frac{\tilde{g}(t)}{Z_g(x)} \;\leq\; c_g\frac{\tilde{g}(t)}{Z_{\tilde{g}}(x)}\,, \quad (5.30)$$

where we used $Z_g(x) \geq Z_{\tilde{g}}(x)$, which comes from $g(t) \geq \tilde{g}(t)$. We also have

$$P_{\tilde{g}}(x,y) \;=\; \frac{\tilde{g}(t)}{Z_{\tilde{g}}(x)}\,\min\left\{1\,,\,t\,\frac{\tilde{g}(1/t)}{Z_{\tilde{g}}(y)}\frac{Z_{\tilde{g}}(x)}{\tilde{g}(t)}\right\} \;=\; \min\left\{\frac{\tilde{g}(t)}{Z_{\tilde{g}}(x)}\,,\,\frac{t\tilde{g}(1/t)}{Z_{\tilde{g}}(y)}\right\} \;\geq$$
$$\frac{\min\left\{\tilde{g}(t),t\tilde{g}(1/t)\right\}}{c_{\tilde{g}}\,Z_{\tilde{g}}(x)} \;=\; \frac{\tilde{g}(t)}{c_{\tilde{g}}\,Z_{\tilde{g}}(x)}\,. \quad (5.31)$$

Part (a) follows from (5.30) and (5.31), while (b) and (c) follow from (a) and Theorem 4. □

The function $\tilde{g}(t) = \min\{g(t), t\,g(1/t)\}$ satisfies $\tilde{g}(t) = t\,\tilde{g}(1/t)$ by definition. Therefore Theorem 5 implies that for any $g : \mathbb{R}_+ \to \mathbb{R}_+$ there is a corresponding balancing function $\tilde{g}$ which leads to a more efficient MH algorithm modulo $c_g c_{\tilde{g}}$. This result is relevant only if $c_g c_{\tilde{g}}$ is not too large. We now show that for our distribution of interest, $\pi(\rho)$ of Section 5.3.1, it holds that $c_g \to 1$ as $|\Omega| \to \infty$.

Theorem 6. *Let $\{w_{ij}\}_{i,j=1}^{\infty}$ be positive weights with $\inf_{i,j\in\mathbb{N}} w_{ij} > 0$ and $\sup_{i,j\in\mathbb{N}} w_{ij} < \infty$. Let $\pi^{(n)}(\rho) \propto \prod_{i=1}^{n} w_{i\rho(i)}$ for $\rho \in \Omega^{(n)} = \mathcal{S}_n$ and let the neighboring structure $\{N(\rho)\}_{\rho\in\Omega^{(n)}}$ be as in (5.22). For any $g : \mathbb{R}_+ \to \mathbb{R}_+$ with g and $1/g$ locally bounded, $c_g^{(n)}$ given by (5.28) satisfies*

$$c_g^{(n)} \;\to\; 1 \qquad \text{as } n \to \infty\,.$$

Proof. Fix $\rho \in \mathcal{S}_n$ and $\rho' = \rho \circ (i_0, j_0)$ for some $i_0, j_0 \in \{1,\ldots,n\}$, with $i_0 < j_0$. Denoting $g\left(\frac{\pi^{(n)}(\rho\circ(i,j))}{\pi^{(n)}(\rho)}\right) = g\left(\frac{w_{i\rho(j)}w_{j\rho(i)}}{w_{i\rho(i)}w_{j\rho(j)}}\right)$ by g_{ij}^{ρ} it holds

$$Z_g^{(n)}(\rho) \;=\; \sum_{i,j=1,\,i<j}^{n} g_{ij}^{\rho} \;=\; \sum_{\substack{i,j=1,\,i<j\\ \{i,j\}\cap\{i_0,j_0\}=\emptyset}}^{n} g_{ij}^{\rho} + \sum_{\substack{i,j=1,\,i<j\\ \{i,j\}\cap\{i_0,j_0\}\neq\emptyset}}^{n} g_{ij}^{\rho}\,.$$

Given $I = \left[\frac{\inf_{i,j} w_{ij}^2}{\sup_{i,j} w_{ij}^2}, \frac{\sup_{i,j} w_{ij}^2}{\inf_{i,j} w_{ij}^2}\right]$, $\underline{g} = \inf_{t\in I} g(t)$ and $\overline{g} = \sup_{t\in I} g(t)$ it holds $\underline{g} \leq g_{ij}^{\rho} \leq \overline{g}$. Note that $\underline{g} > 0$ and $\overline{g} < \infty$ because g and $1/g$ are locally bounded and I is

compact. Therefore

$$\sum_{\substack{i,j=1,\, i<j \\ \{i,j\}\cap\{i_0,j_0\}=\emptyset}}^{n} g_{ij}^{\rho} \;\geq\; \sum_{\substack{i,j=1,\, i<j \\ \{i,j\}\cap\{i_0,j_0\}=\emptyset}}^{n} \underline{g} \;=\; \left(\frac{n(n-1)}{2} - (2n-3)\right)\underline{g} = O(n^2)$$

and

$$\sum_{\substack{i,j=1,\, i<j \\ \{i,j\}\cap\{i_0,j_0\}\neq\emptyset}}^{n} g_{ij}^{\rho} \;\leq\; \sum_{\substack{i,j=1,\, i<j \\ \{i,j\}\cap\{i_0,j_0\}\neq\emptyset}}^{n} \overline{g} \;=\; (2n-3)\overline{g} = O(n)\,.$$

If follows that

$$\lim_{n\to\infty} \frac{Z_g^{(n)}(\rho')}{Z_g^{(n)}(\rho)} \;=\; \lim_{n\to\infty} \frac{\sum_{\{i,j\}\cap\{i_0,j_0\}=\emptyset} g_{ij}^{\rho}}{\sum_{\{i,j\}\cap\{i_0,j_0\}=\emptyset} g_{ij}^{\rho'}} \;=\; \lim_{n\to\infty} \frac{\sum_{\{i,j\}\cap\{i_0,j_0\}=\emptyset} g_{ij}^{\rho}}{\sum_{\{i,j\}\cap\{i_0,j_0\}=\emptyset} g_{ij}^{\rho}} \;=\; 1\,,$$

where $g_{ij}^{\rho} = g_{ij}^{\rho'}$ for $\{i,j\}\cap\{i_0,j_0\} = \emptyset$ because $\rho'(i) = \rho(i)$ for $i \in \{1,\ldots,n\}\backslash\{i_0,j_0\}$. □

Theorem 6 implies that for the model we are considering balanced proposals (Definition 1) are asymptotically more efficient than non-balanced proposals. Moreover we conjecture that such a property ($c_g \to 1$ as $|\Omega| \to \infty$) is not specific to the model we are considering but rather it is a smoothness condition that holds for most distributions encountered in practical MCMC application.

Note that both the uniform proposal $Q^U(x,y) \propto \mathbb{1}_{N(x)}(y)$ of (5.23) and the "naive" informed proposal $Q^T(x,y) \propto \pi(y)\mathbb{1}_{N(x)}(y)$ of (5.24) can be expressed as Q_g of (5.26) for $g(t) = 1$ and $g(t) = t$ respectively. Thus Theorems 5 and 6 imply that such proposals are asymptotically Peskun dominated by $Q_{\tilde{g}}$ for $\tilde{g}(t) = \min\{1,t\}$.

5.4.1 Connection between balancing functions and acceptance probability functions.

Balancing functions (BFs) are closely connected to acceptance probability functions (APFs), which are the functions leading to valid MCMC algorithms when used in the accept/reject mechanism (Section 5.1.1). In fact a function $a(x,y) = g(t(x,y))$, with $t(x,y) = \frac{\pi(y)Q(x,y)}{\pi(x)Q(x,y)}$, is a valid APF if and only if $g(t) = tg(1/t)$ and $g(t) \leq 1$ (see Remark 11). Therefore APFs and BFs need to satisfy the same equation: $g(t) = tg(1/t)$. It is somehow intuitive that accept/reject functions inducing detailed balance (i.e. APFs) are related to functions that generate proposals in approximate detailed balance (i.e. BFs).

At the same time APFs and BFs are different in at least two aspects. First,

BFs do not need to be bounded by 1 and thus the class of APFs is smaller than the class of BFs. The latter includes elements such as $g(t) = \sqrt{t}$ or $g(t) = \max\{1, t\}$. Moreover, the optimality in the two classes appears to behave differently. It is well known that, in the context of APF, the Metropolis-Hastings function $g_{MH}(t) = \min\{1, t\}$ is the optimal function in terms of Peskun ordering (Tierney, 1998). Instead, when used as a BF to produce a proposal Q_g, no choice of g seems to Peskun-dominate the others. In other words, given two BFs g_1 and g_2 (e.g. $g_1(t) = \min\{1, t\}$ and $g_2(t) = \sqrt{t}$), neither P_{g_1} nor P_{g_2} will (asymptotically) Peskun-dominate the other in general.

In Section 5.5 we show that, at least in one specific case, the choice of g that minimizes the mixing time is the Barker function, $g_B(t) = \frac{t}{1+t}$. We should also note that, among the class of BFs, $g(t) = \sqrt{t}$ is a special choice, for example because of its linearity in the log scale and because of its connection to informed proposals in continuous frameworks (see Section 5.6).

5.5 The hypercube case

In Section 5.4 we showed that balanced proposals are the most efficient among the class of proposals Q_g of (5.26). In this section we consider the following natural question: is there an optimal balanced proposal? If yes, which one is it?

We answer this question in the specific framework of independent components. For simplicity, in this section we consider the case of binary random variables, although we conjecture that this argument could be extended to more general cases (e.g. variables taking values in $\{1, \dots, k\}$ rather than $\{0, 1\}$). We show that in this specific case the Barker balancing function $g_B(t) = \frac{t}{1+t}$ leads to the smallest mixing time. Nevertheless we note that, in practical implementations, all the balanced proposals we tested performed similarly.

The optimal proposal for independent binary variables

For any positive integer n we define $\Omega^{(n)} = \{0, 1\}^n$. Given $\mathbf{x}_{1:n} = (x_1, \dots, x_n)$ in $\Omega^{(n)}$ let

$$\pi^{(n)}(\mathbf{x}_{1:n}) \;=\; \prod_{i=1}^{n} p_i^{1-x_i}(1-p_i)^{x_i}\,.$$

For simplicity, we assume that $\inf_{i\in\mathbb{N}} p_i > 0$ and $\sup_{i\in\mathbb{N}} p_i < 1$. The neighborhood of $\mathbf{x}_{1:n}$ is defined as

$$N\big(\mathbf{x}_{1:n}\big) \;=\; \Big\{\mathbf{y}_{1:n} = (y_1, \ldots, y_n) \;:\; \sum_{i=1}^{n} |x_i - y_i| = 1\Big\}\,.$$

Given such a neighboring structure it can be shown that, for any $g : \mathbb{R}_+ \to \mathbb{R}_+$ with g and $1/g$ locally bounded, $c_g^{(n)} \to 1$ as $n \to \infty$ (see (5.28) for definition of $c_g^{(n)}$ and Theorem 6 for a similar result and proof). In this section we assume that g and $1/g$ are locally bounded. Any proposal Q_g defined in (5.26) can be written as

$$Q^{(n)}(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) \propto \begin{cases} a_i & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}_{1:n}^{(i)}\,, \\ b_i & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}_{1:n}^{(i)}\,, \\ 0 & \text{if } \mathbf{y}_{1:n} \notin N\big(\mathbf{x}_{1:n}\big)\,, \end{cases} \tag{5.32}$$

where a_i and b_i are positive real numbers bounded away from 0 and infinity and $\mathbf{e}_{1:n}^{(i)}$ is a vector having the i-th coordinate equal to 1 and the others equal to 0, meaning that $\mathbf{e}_{1:n}^{(i)} = (e_1^{(i)}, \ldots, e_n^{(i)})$ with $e_j^{(i)}$ being 1 if i equals j and 0 otherwise. Although not strictly necessary, we assume that if $p_i = p_j$ then also $a_i = a_j$ and $b_i = b_j$, so that the proposals in (5.32) can be expressed as Q_g in (5.26) for some g. We want to study the behavior of MCMC algorithms induced by the proposals in (5.32) as n increases, to obtain some indication on what is the optimal proposal in this framework. We reparametrize these distributions as

$$Q^{(n)}(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) = \frac{1}{Z^{(n)}(\mathbf{x}_{1:n})} \begin{cases} v_i\, c_i\, (1 - p_i) & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}_{1:n}^{(i)}\,, \\ v_i\, (1 - c_i)\, p_i & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}_{1:n}^{(i)}\,, \\ 0 & \text{if } \mathbf{y}_{1:n} \notin N\big(\mathbf{x}_{1:n}\big)\,, \end{cases} \tag{5.33}$$

where $c_i \in (0, 1)$, $v_i > 0$ with $\sup_{i\in\mathbb{N}} v_i < \infty$ and $\inf_{i\in\mathbb{N}} v_i > 0$, and

$$Z^{(n)}(\mathbf{x}_{1:n}) \;=\; \sum_{i=1}^{n} v_i \left(c_i(1 - p_i)(1 - x_i) + (1 - c_i)p_i x_i\right)$$

is the normalizing constant of $Q(\mathbf{x}_{1:n}, \cdot)$. Note that, for $\mathbf{X}_{1:n} \sim \pi^{(n)}$, it holds

$$\operatorname{var}\left[\frac{Z^{(n)}(\mathbf{X}_{1:n})}{n}\right] < \frac{\sup_{i\in\mathbb{N}} v_i^2}{n} \to 0\,, \tag{5.34}$$

and

$$\mathbb{E}\left[\frac{Z^{(n)}(\mathbf{X}_{1:n})}{n}\right] \to \bar{Z} = \bar{Z}(\mathbf{v}) = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} v_i\, p_i(1-p_i)}{n}\,, \tag{5.35}$$

where $\mathbf{v}$ denotes the sequence $(v_1, v_2, \dots)$ and we assumed that $\lim_{n\to\infty} \frac{\sum_{i=1}^{n} v_i\, p_i(1-p_i)}{n}$ exists and has a finite non-zero value. It follows that

$$\frac{Z^{(n)}(\mathbf{X}_{1:n})}{n} \quad \xrightarrow{a.s.} \quad \bar{Z} \qquad \text{for } \mathbf{X}_{1:n} \sim \pi^{(n)} \text{ and } n \to \infty\,.$$

Let $\mathbf{X}^{(n)}(t)$ be the Metropolis-Hastings Markov chain on $\Omega^{(n)}$ with target measure $\pi^{(n)}$ and proposal $Q^{(n)}$ in (5.33). For any real time t and positive integer $k \le n$, we define

$$S^{(n)}_{1:k}(t) = \left(X^{(n)}_1(\lfloor nt \rfloor), \dots, X^{(n)}_k(\lfloor nt \rfloor)\right)\,,$$

with $\lfloor nt \rfloor$ being the largest integer smaller than nt. Note that $S^{(n)}_{1:k} = (S^{(n)}_{1:k}(t))_{t\ge 0}$ is a continuous-time (non-Markov) stochastic process on $\{0,1\}^k$ describing the first k components of $(\mathbf{X}^{(n)}(t))_{t\ge 0}$.

Theorem 7. *Let $\boldsymbol{X}^{(n)}(0) \sim \pi^{(n)}$ for every n. For any positive integer k, it holds*

$$S^{(n)}_{1:k} \overset{n\to\infty}{\Longrightarrow} S_{1:k},$$

where $\Rightarrow$ denotes weak convergence and $S_{1:k}$ is a continuous-time Markov chain on $\{0,1\}^k$ with jumping rates given by

$$A\left(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}\right) = \begin{cases} e_i(\mathbf{v}, c_i)\cdot(1-p_i) & \text{if } \mathbf{y}_{1:k} = \mathbf{x}_{1:k} + \mathbf{e}^{(i)}_{1:k}\,, \\ e_i(\mathbf{v}, c_i)\cdot p_i & \text{if } \mathbf{y}_{1:k} = \mathbf{x}_{1:k} - \mathbf{e}^{(i)}_{1:k}\,, \\ 0 & \text{if } \mathbf{y}_{1:k} \notin N\big(\mathbf{x}_{1:k}\big) \text{ and } \mathbf{y}_{1:k} \neq \mathbf{x}_{1:k}\,, \end{cases} \tag{5.36}$$

where

$$e_i(\boldsymbol{v}, c_i) = \frac{1}{\bar{Z}(\boldsymbol{v})}\, v_i\, \left((1-c_i) \wedge c_i\right)\,. \tag{5.37}$$

Proof. Let k be fixed and let $A^{(n)}$ be the $k \times k$ matrix describing the jumping rates of $S^{(n)}_{1:k}$. For any $i \le k$ it holds the following. If $x_i = 0$

$$\begin{aligned} &A^{(n)}(\mathbf{x}_{1:k}, \mathbf{x}_{1:k} + \mathbf{e}^{(i)}_{1:k}) \;= \\ &\quad n\, Q^{(n)}(\mathbf{x}_{1:n}, \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}) \left(1 \wedge \frac{\pi^{(n)}(\mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}) Q^{(n)}(\mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}, \mathbf{x}_{1:n})}{\pi^{(n)}(\mathbf{x}_{1:n}) Q^{(n)}(\mathbf{x}_{1:n}, \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n})}\right) \;= \\ &\qquad\qquad \frac{v_i(1-p_i)}{\frac{Z^{(n)}(\mathbf{x}_{1:n})}{n}} \left(c_i \wedge \left((1-c_i)\frac{Z^{(n)}(\mathbf{x}_{1:n})}{Z^{(n)}(\mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n})}\right)\right)\,, \end{aligned}$$

and, similarly, if $x_i = 1$

$$A^{(n)}(\mathbf{x}_{1:k}, \mathbf{x}_{1:k} - \mathbf{e}^{(i)}_{1:k}) \;=\; \frac{v_i p_i}{\frac{Z^{(n)}(\mathbf{x}_{1:n})}{n}} \left((1 - c_i) \wedge \left(c_i \frac{Z^{(n)}(\mathbf{x}_{1:n})}{Z^{(n)}(\mathbf{x}_{1:n} - \mathbf{e}^{(i)}_{1:n})} \right) \right) .$$

On the other hand, for any $\mathbf{x}_{1:k}$ and $\mathbf{y}_{1:k}$ in $\{0,1\}^k$ such that $\mathbf{y}_{1:k} \neq \mathbf{x}_{1:k}$ and $\mathbf{y}_{1:k} \notin N\big(\mathbf{x}_{1:k}\big)$, it holds $A^{(n)}(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}) = 0$. Note that $S^{(n)}_{1:k}$ is not a Markov process because the jumping rates $A^{(n)}(\mathbf{x}_{1:k}, \mathbf{x}_{1:k} + \mathbf{e}^{(i)}_{1:k})$ and $A^{(n)}(\mathbf{x}_{1:k}, \mathbf{x}_{1:k} - \mathbf{e}^{(i)}_{1:k})$ depend also on the last $(n-k)$ components $(x_{k+1}, \ldots, x_n)$. We define the following sequence of sets:

$$R_n = \Big\{ (x_{k+1}, \ldots, x_n) \in \{0,1\}^{n-k} \;: \\ \left| \frac{Z^{(n)}(\mathbf{x}_{1:n})}{n} - \mathbb{E}\left[\frac{Z^{(n)}(\mathbf{X}_{1:n})}{n} \right] \right| \leq \frac{1}{n^{1/4}} \;\; \forall\, (x_1, \ldots, x_k) \in \Omega^{(k)} \Big\} . \quad (5.38)$$

Given $\alpha_n = \frac{1}{n^{1/4}} + \left| \mathbb{E}\left[\frac{Z^{(n)}(\mathbf{X}_{1:n})}{n} \right] - \bar{Z} \right|$, it holds

$$\sup_{\mathbf{x}_{1:n} \in \Omega^{(k)} \times R_n} \left| \frac{Z^{(n)}(\mathbf{x}_{1:n})}{n} - \bar{Z} \right| \;\leq\; \alpha_n \;\to\; 0 \,, \quad (5.39)$$

where the inequality follows by (5.38) and the convergence follows from (5.35). Moreover, from (5.34) and (5.38) it follows that $\lim_{n\to\infty} \pi^{(n)}(\Omega^{(k)} \times R_n) = 1$. For any $i \leq k$ and $\mathbf{x}_{1:n}$ in $\Omega^{(k)} \times R_n$ with $x_i = 0$ it holds

$$|A^{(n)}(\mathbf{x}_{1:k}, \mathbf{x}_{1:k} + \mathbf{e}^{(i)}_{1:k}) - A(\mathbf{x}_{1:k}, \mathbf{x}_{1:k} + \mathbf{e}^{(i)}_{1:k})| \;= \\ v_i(1-p_i) \left| \frac{1}{\frac{Z^{(n)}(\mathbf{x}_{1:n})}{n}} \left(c_i \wedge \left((1-c_i) \frac{Z^{(n)}(\mathbf{x}_{1:n})}{Z^{(n)}(\mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n})} \right) \right) - \frac{c_i \wedge (1-c_i)}{\bar{Z}} \right| \leq \\ v_i \Bigg(\frac{1}{\frac{Z^{(n)}(\mathbf{x}_{1:n})}{n}} \left| \left(c_i \wedge \left((1-c_i) \frac{Z^{(n)}(\mathbf{x}_{1:n})}{Z^{(n)}(\mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n})} \right) \right) - (c_i \wedge (1-c_i)) \right| + \\ \left| \frac{(c_i \wedge (1-c_i))}{\frac{Z^{(n)}(\mathbf{x}_{1:n})}{n}} - \frac{(c_i \wedge (1-c_i))}{\bar{Z}} \right| \Bigg) \leq \\ v_i \left(\frac{1}{\bar{Z} - \alpha_n} \left| \frac{2\alpha_n}{\bar{Z} - \alpha_n} \right| + \left| \frac{\alpha_n}{\bar{Z}\,(\bar{Z} - \alpha_n|)} \right| \right) \quad \overset{n\to\infty}{\longrightarrow} \quad 0 \,,$$

where we used (5.39) and the fact that p_i, $(1-p_i)$, c_i and $(1-c_i)$ belong to $(0,1)$.

The case $x_i = 1$ is analogous. It follows that

$$\sup_{\mathbf{x}_{1:n} \in \Omega^{(k)} \times R_n} |A^{(n)}(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}) - A(\mathbf{x}_{1:k}, \mathbf{y}_{1:k})| \quad \stackrel{n\to\infty}{\longrightarrow} \quad 0\,.$$

From the latter convergence and $\lim_{n\to\infty} \pi^{(n)}(\Omega^{(k)} \times R_n) = 1$ it follows, using Ethier & Kurtz (1986, Chapter 4, Corollary 8.7), that $S^{(n)}_{1:k} \stackrel{n\to\infty}{\Longrightarrow} S_{1:k}$. □

Theorem 7 tells us that, in the limiting process $S_{1:k}$, each bit is flipping independently of the others, with flipping rate of the i-th bit being proportional to $e_i(\mathbf{v}, c_i)$. Note that Theorem 7 considers the stationarity regime: $\mathbf{X}^{(n)}(0) \sim \pi^{(n)}$ for every n.

From (5.37) we see that, in the limiting process, the parameter c_i influences only the behaviour of the i-th component. It follows that, for any i, the asymptotically optimal choice of c_i can be derived by maximizing the limiting speed of the i-th component, $e_i(\mathbf{v}, c_i)$, which is done by setting $c_i = \frac{1}{2}$. On the other hand, from (5.37) we can see that each v_i is proportional to the rate at which the i-th component is flipping in the limiting process $S_{1:k}$, but at the same time affects the other components through the normalizing constant $\bar{Z}(\mathbf{v})$. Intuitively, the parameter v_i represents how much effort we put into updating the i-th component, where increasing v_i reduces the effort put into updating other components.

Remark 15. *Choosing $c_i = \frac{1}{2}$ for each i corresponds to choosing a balanced proposal (see Definition 1). The optimality of $c_i = \frac{1}{2}$ reflects the fact that a balanced proposal (i.e. $c_i = \frac{1}{2}$) produces better mixing than a non-balanced one (i.e. $c_i \neq \frac{1}{2}$), as suggested by Theorem 5. On the other hand it is not so obvious how to optimally choose $(v_1, v_2, \dots)$. This reflects the fact that it is not straightforward to compare balanced proposals among themselves.*

In order to discriminate among various choices of $(v_1, v_2, ...)$ we choose the one that minimizes the mixing time of $\{S_{1:k}\}_{k=1}^{\infty}$ for k going to infinity. Although this is not the only possible criterion to use, it is a reasonable and natural one. Barrera *et al.* (2006, Prop. 7) and Bon & Păltănea (2001, Cor. 4.3) give us an expression for the mixing time of $\{S_{1:k}\}_{k=1}^{\infty}$. Such results tell us that, in the case of a sequence of independent binary Markov processes like $\{S_{1:k}\}_{k=1}^{\infty}$, the asymptotic mixing time depends on the flipping rates of the worst components (provided they are a non-negligible quantity), and in particular in our case the mixing time is minimized by maximizing the quantity $\bar{Z}(\mathbf{v})^{-1} \liminf_{i\to\infty} v_i$ (see (5.35) for the definition of $\bar{Z}(\mathbf{v})$ and Barrera *et al.* (2006) and Bon & Păltănea (2001) for the precise assumptions on the flipping rates). It can be seen that the latter quantity is maximized by choosing

v_i to be constant over i, meaning $v_i = \bar{v}$ for any $i \in \mathbb{N}$ for some $\bar{v} > 0$. The value of $\bar{v}$ is irrelevant because we defined the proposal $Q^{(n)}$ up to proportionality. Therefore we can simply set $v_i = 1$ for any $i \in \mathbb{N}$. Thus the proposal that minimizes the asymptotic mixing time of $\{S_{1:k}\}_{k=1}^{\infty}$ is

$$Q^{(n)}(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) \propto \begin{cases} (1-p_i) & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}\,, \\ p_i & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}^{(i)}_{1:n}\,, \\ 0 & \text{if } \mathbf{y}_{1:n} \notin N\big(\mathbf{x}_{1:n}\big)\,. \end{cases} \tag{5.40}$$

Note that this proposal corresponds to the balanced proposal with balancing function $g_B(t) = \frac{t}{1+t}$

$$Q^{(n)}(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) \propto \begin{cases} \frac{\pi^{(n)}(\mathbf{y}_{1:n})}{\pi^{(n)}(\mathbf{x}_{1:n}) + \pi^{(n)}(\mathbf{y}_{1:n})} & \text{if } \mathbf{y}_{1:n} \in N(\mathbf{x}_{1:n})\,, \\ 0 & \text{otherwise}\,. \end{cases} \tag{5.41}$$

Remark 16. *In this section we have proceeded in two steps. First we proved the convergence of the finite dimensional projections $S^{(n)}_{1:k}$ to $S_{1:k}$ for fixed k and n going to infinity (Theorem 7). Secondly we studied the mixing time of $S_{1:k}$ for k going to infinity. However, it would be more elegant and neat to prove directly the convergence of $S^{(n)}_{1:n}$ to an infinite dimensional stochastic process S, thus avoid the additional intermediate step of finite dimensional projections and the consequent double limiting operation (first $n \to \infty$, then $k \to \infty$). The need to work with finite dimensional projections arises from the probabilistic technique used to prove Theorem 7, which is the typical technique used to prove MCMC scaling results (e.g. Roberts* et al. *, 1997 and Roberts, 1998). To overcome such shortcomings, we have been working on a novel approach to prove scaling results, based on Dirichlet Forms and Mosco convergence (Mosco, 1994), that naturally allows to work on infinite dimensional spaces, leading to more general and robust MCMC scaling results. We are currently writing up the results in Zanella* et al. *(2015).*

5.6 Possible extensions and future works

The results of this chapter (e.g. Theorem 5) provide valuable theoretical guidance to help design efficient proposals to sample from the distribution $\hat{\pi}(\rho)$ of Chapter 4. We will use such theoretical guidelines in particular in Section 6.1. Nonetheless there are various open research questions related to the notion of balanced proposals. In this section we list some of those. Chapter 8 will also provide more details.

Connections to MALA and other continuous-state algorithms

It is natural to wonder if and how the results of Section 5.4 relate to continuous state frameworks. In this section we present possible approaches to extend the framework of balanced proposals to continuous frameworks. Interestingly, we observe that, for example, the MALA algorithm (see Section 5.2) can be interpreted as an example of a balanced proposal for a certain choice of g.

At the current stage, for simplicity and concreteness, we restrict to the case of state space $\Omega = \mathbb{R}^n$, target measure $\pi(dx) = f(x)dx$ and proposal distribution $Q(x, dy) = q(x, y)dy$.

Informed proposal distributions considered

In a directly analogous way to the discrete case (5.26) we could suppose that, given a current state $x \in \Omega$, the set of allowed moves is defined by a neighborhood of x in $\mathbb{R}^n$ that we denote by $N(x) \subseteq \mathbb{R}^n$. A simple example is $N(x) = B_x(R) = \{y \in \mathbb{R}^n : |x - y| < R\}$. Following the framework we used for discrete spaces, it would be natural to consider proposals depending only on the probability of the proposed state. This would lead us to suppose that there exists a function $g : \mathbb{R}^+ \to \mathbb{R}^+$ such that

$$q(x, y) = q_g(x, y) \propto \begin{cases} g\left(\frac{f(y)}{f(x)}\right) & \text{if } y \in N(x), \\ 0 & \text{otherwise.} \end{cases} \tag{5.42}$$

Nevertheless, when $\Omega = \mathbb{R}^n$, it is not natural to consider proposal distributions based on a neighboring structure as in (5.42). It is more common to consider local moves based on some smoothing function as follows. Suppose that for any x we have a function $h_x : \mathbb{R}^n \to \mathbb{R}^+$ going to 0 for $|x|$ going to infinity and then consider $q(x, y) \propto g\left(\frac{f(y)}{f(x)}\right) h_x(y)$. For simplicity here we limit ourselves to the Gaussian case $h_x(y) = \exp(-\frac{|y-x|^2}{2\sigma^2})$ with $\sigma > 0$. Therefore we could consider proposals of the following form:

$$q(x, y) \propto g\left(\frac{f(y)}{f(x)}\right) \cdot \exp\left(-\frac{|y - x|^2}{2\sigma^2}\right). \tag{5.43}$$

In general we have no guarantee that (5.43) defines a proper density function, but we ignore integrability issues for the moment.

Derivative-based proposal distributions

Neither (5.42) nor (5.43) are feasible choices in practice. In fact, most of the time, sampling from (5.42) or (5.43) is as difficult as sampling from the target itself. In practice, one can approximate the log density with its Taylor expansion about

x. The first order proposal distribution with balancing function g and Gaussian smoothing would then be

$$q(x,y) \;\propto\; g\left(\frac{\exp\left(\log f(x) + \nabla(\log f)(x)\cdot(y-x)\right)}{f(x)}\right)\cdot\exp\left(-\frac{|y-x|^2}{2\sigma^2}\right) \;\propto$$
$$g\left(\exp\left(\nabla(\log f)(x)\cdot(y-x)\right)\right)\cdot\exp\left(-\frac{|y-x|^2}{2\sigma^2}\right). \quad (5.44)$$

Similarly the k-th order proposal distribution with balancing function g and Gaussian smoothing would be

$$q(x,y) \propto g\left(\frac{\exp\left(\sum_{|\alpha|\le k} D^\alpha(\log f)(x)(y-x)^\alpha\right)}{f(x)}\right)\cdot\exp\left(-\frac{|y-x|^2}{2\sigma^2}\right), \quad (5.45)$$

where $\alpha = (\alpha_1,\ldots,\alpha_n)$ is a multi-index, $D^\alpha = \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1}\cdots\partial x_n^{\alpha_n}}$ and $x^\alpha = x_1^{\alpha_1}\cdots x_n^{\alpha_n}$. Note that, when one considers higher order derivatives, some care should be taken to ensure $q(x,y)$ is integrable.

If one chooses the balancing function $g(t) = \sqrt{t}$, then the resulting first order balanced proposal, see (5.44), has proposal density

$$q(x,y) \propto \exp\left(\nabla(\log f)(x)\cdot(y-x)/2 - \frac{|y-x|^2}{2\sigma^2}\right), \quad (5.46)$$

which is exactly the MALA proposal (see Section 5.2) where different values of σ corresponds to different values of the step size.

This simple connection with continuous state proposals suggests various research directions. Section 8.3 describes some of these.

Practical implementations

Theorems 5 and 6 guarantee that, for large $|\Omega|$, an informed proposal will produce a MCMC algorithm that mixes faster than the one obtained from the uninformed, uniform proposal. Nevertheless, in order to use balanced proposals one has to evaluate the target measure on the neighbors of the current state, $\left\{\frac{\pi(y)}{\pi(x)}\right\}_{y\in N(x)}$, before proposing a new state. Clearly, this is an additional computational burden (similar to the need to evaluate the first derivative of the target log-density for the MALA algorithm). Whether the improvement in mixing is worth this additional burden probably varies from case to case.

Note that once an informed proposal producing good mixing has been iden-

tified, then one can build an approximation to it which is easy to sample from. This is what we do in Section 6.1.2. In general, it would be interesting to study performances of balanced proposals on common discrete models arising, for example, from Model Selection problems or Bayesian Non-Parametric models and to devise cheap approximate versions in such contexts (similarly to what is done in Section 6.1.2). In Section 8.3 we discuss this point further.

Chapter 6

MCMC algorithm for matching spaces

In Chapter 5 we described MCMC methods and we discussed how to design informed proposals in discrete spaces, providing some theoretical results. In this chapter we describe the actual MCMC algorithm we implemented to sample from the posterior distribution of the Bayesian Random Partition Model of Chapter 2 and we discuss various related issues (convergence diagnostic, tempering and parallel computation).

The distribution of interest is $\pi(\rho, \sigma, \mathbf{p}^{(c)}, \lambda|\mathbf{x})$ given in (3.14)-(3.17). In order to obtain approximate samples from this distribution we use the following Metropolis-within-Gibbs algorithm:

1. Initialize $(\rho, \sigma, \mathbf{p}^{(c)}, \lambda)$ with some $(\rho_0, \sigma_0, \mathbf{p}_0^{(c)}, \lambda_0) \in \mathcal{M}_{G_\mathbf{x}} \times \mathbb{R}_+ \times [0,1]^k \times \mathbb{R}_+$,
2. For t running from 1 to $(T_{burn} + T_{sample})$ do the following operations
   (a) Sample $\mathbf{p}_t^{(c)} \sim \pi(\mathbf{p}^{(c)}|\rho_{t-1}, \sigma_{t-1}, \lambda_{t-1}, \mathbf{x})$,
   (b) Sample $\lambda_t \sim \pi(\mathbf{p}^{(c)}|\rho_{t-1}, \sigma_{t-1}, \mathbf{p}_t^{(c)}, \mathbf{x})$,
   (c) Sample $\sigma_t \sim P_{\pi(\sigma|\rho_{t-1}, \lambda_t, \mathbf{p}_t^{(c)}, \mathbf{x})}(\sigma_{t-1}, \cdot)$, where $P_{\pi(\sigma|\rho_{t-1}, \lambda_t, \mathbf{p}_t^{(c)}, \mathbf{x})}$ is an ergodic, $\pi(\sigma|\rho_{t-1}, \lambda_t, \mathbf{p}_t^{(c)}, \mathbf{x})$-stationary Markov transition kernel,
   (d) Sample $\rho_t \sim P_{\pi(\rho|\sigma_t, \lambda_t, \mathbf{p}_t^{(c)}, \mathbf{x})}(\rho_{t-1}, \cdot)$, where $P_{\pi(\rho|\sigma_t, \lambda_t, \mathbf{p}_t^{(c)}, \mathbf{x})}$ is an ergodic, $\pi(\rho|\sigma_t, \lambda_t, \mathbf{p}_t^{(c)}, \mathbf{x})$-stationary Markov transition kernel,
3. Collect the samples $\{(\rho_t, \sigma_t, \mathbf{p}_t^{(c)}, \lambda_t)\}_{t=T_{burn}+1}^{T_{sample}}$ as approximate samples from the joint distribution $\pi(\rho, \sigma, \mathbf{p}^{(c)}, \lambda|\mathbf{x})$.

Note that direct sampling from $\pi(\mathbf{p}^{(c)}|\rho, \sigma, \lambda, \mathbf{x})$ and $\pi(\lambda|\rho, \sigma, \mathbf{p}^{(c)}, \mathbf{x})$ is straightforward. Moreover, given $(\rho, \mathbf{p}^{(c)}, \lambda, \mathbf{x})$, few Markov chain steps (for example using

basic MH algorithms) are sufficient for the distribution of σ to be close to its stationary distribution $\pi(\sigma|\rho, \mathbf{p}^{(c)}, \lambda, \mathbf{x})$. Therefore Steps $2(a) - 2(c)$ of the algorithm are easy to implement. In contrast sampling from $\pi(\rho|\mathbf{x}, \sigma, \mathbf{p}^{(c)}, \lambda)$, which for simplicity we denote by $\hat{\pi}(\rho)$, is challenging (see Chapter 4). Therefore we need to be careful in designing an appropriate Metropolis-Hastings (MH) algorithm targeting $\hat{\pi}(\rho)$ in Step 2(d) of the Metropolis-within-Gibbs algorithm. In this chapter we consider ways of improving the efficiency and assessing the convergence of MH algorithms in this framework.

Section 6.1 focuses on the 2-color version of $\hat{\pi}(\rho)$, while Section 6.2 deals with the k-colors case for general k (i.e. $k \geq 3$). We commence by considering the two-color case because there is more known theory than in the general case and because the combinatorial structure of the sample space is simpler (and thus can help to provide intuition). Secondly, since the two-color version of the algorithm will constitute the building block of the general version, it should be designed carefully (see Section 6.2). Finally, an algorithm to sample from the two-color version of the posterior could allow one to study pairwise interaction between placenames, and thus could also be relevant for other applications (see e.g. Dellaert *et al.* , 2003).

6.1 The two-color case

We view ρ as a matching in a bipartite graph with n_1 red points and n_2 blue points (see Section 4.1.1). We denote the edge connecting the i-th red point and the j-th blue point by the ordered couple $(i, j) \in \{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$.

The proposal $Q^{2D}(\rho_{old}, \rho_{new})$ for ρ is defined in two steps. First we select an edge (i, j) according to some probability distribution $q_{\rho_{old}}(i, j)$ on $\{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$. Then, having defined i' as the index such that $(i', j) \in \rho_{old}$, if such an i' exists, and similarly j' as the index such that $(i, j') \in \rho_{old}$, if such a j' exists, we propose a new state $\rho_{new} = \rho_{old} \circ (i, j)$ defined as

$$\begin{cases} \rho_{old} + (i,j), & \text{if neither } i' \text{ nor } j' \text{ exists,} & \text{(Addition)} \\ \rho_{old} - (i,j), & \text{if } (i,j) \in \rho_{old}, & \text{(Deletion)} \\ \rho_{old} - (i,j') + (i,j), & \text{if } j' \text{ exists and } i' \text{ does not exist,} & \text{(Switch)} \\ \rho_{old} - (i',j) + (i,j), & \text{if } i' \text{ exists and } j' \text{ does not exist,} & \text{(Switch)} \\ \rho_{old} - (i',j) - (i,j') & & \\ \quad + (i,j) + (i',j'), & \text{if } i' \text{ and } j' \text{ exist and } (i,j) \notin \rho_{old}, & \text{(Double-Switch)} \end{cases} \tag{6.1}$$

where $\rho - (i, j)$ and $\rho + (i, j)$ denote the matchings obtained from ρ by respec-

tively removing or adding the edge (i, j). Display (6.1) defines the set of allowed moves starting from ρ_{old} and it induces a neighboring structure on the space of matchings as follows: ρ_{new} is a neighbor of ρ_{old} if $\rho_{new} = \rho_{old} \circ (i, j)$ for some (i, j). Figure 6.1provides an example. Note that different values of (i, j) can lead

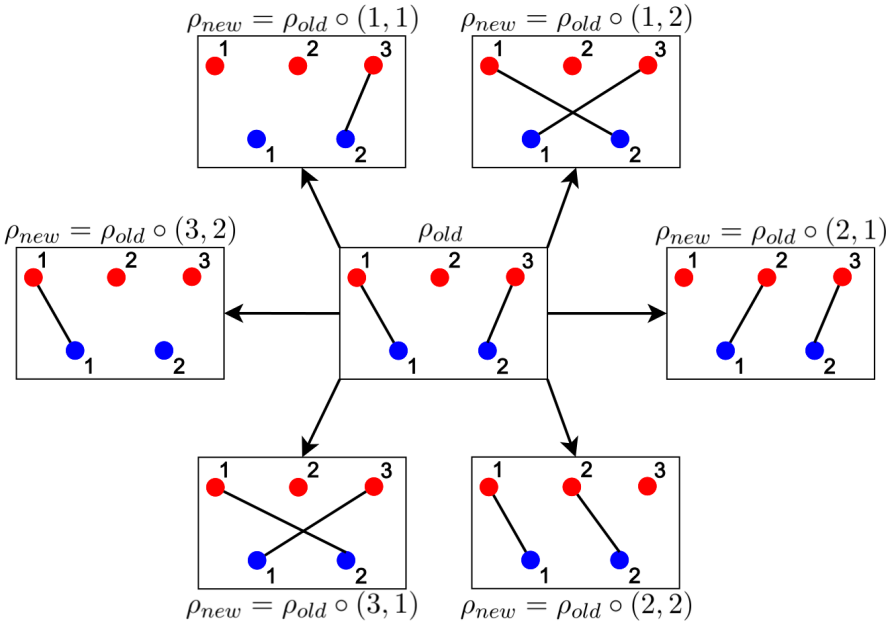


Figure 6.1: Example of allowed moves induced by (6.1). In this case $n_1 = 3$ and $n_2 = 2$, where n_1 and n_2 are the number of red and blue points respectively. Note that the index i in $\rho_{new} = \rho_{old} \circ (i, j)$ refers to a red point, while the index j refers to a blue point.

to the same proposed matching (this is not a problem in the Metropolis-Hastings framework as long as the balancing takes it into account). Moreover note that all the proposed moves are reversible, meaning that, given the current state ρ, for any $(i, j) \in \{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$ there exist $(s, t) \in \{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$ such that $\rho = (\rho \circ (i, j)) \circ (s, t)$. Jerrum & Sinclair (1996) and Oh *et al.* (2009) consider similar but slightly smaller sets of allowed moves, given by: (1) addition and deletion moves and (2) addition, deletion and switch moves. It is plausible that increasing the set of allowed moves improves the mixing of the MH Markov chain.

6.1.1 Different proposals

Display (6.1) does not uniquely identify the proposal $Q^{2D}(\rho_{old}, \rho_{new})$ because we still need to choose $q_{\rho_{old}}(\cdot, \cdot)$. Different choices of $q_{\rho_{old}}(\cdot, \cdot)$ will affect the mixing properties of the MH algorithm. Note that we are in the scenario considered in

Section 5.2, where we need to design a MH proposal distribution given a fixed set of allowed moves for a discrete (actually finite) state space. Previous works (e.g. Jerrum & Sinclair (1996) and Oh *et al.* , 2009) chose $q_{\rho_{old}}(i,j)$ to be a uniform measure over the edges $(i,j) \in E$. A naive implementation of such choice leads to poor mixing because most proposed matchings ρ_{new} are improbable and therefore are typically rejected (in our experiments usually less than 1% of the proposed moves were accepted). Some authors overcome this problem using a truncation approximation of the posterior: they force edge weights below a certain threshold δ to be zero, and then choose

$$q_{\rho_{old}}(i,j) \;\propto\; \mathbb{1}_{\{w_{ij}>\delta\}} \,, \tag{P1}$$

where w_{ij} is the weight of the edge (i,j) defined in (4.5) and $\mathbb{1}$ denotes the indicator function. See for example the measurement validation step in Oh *et al.* (2009).

In the following we use the results of Chapter 5 to propose a choice of $q_{\rho_{old}}$ that achieves a better mixing than (P1) and does so without requiring to target an approximation of the posterior.

Firstly note that, since $\hat{\pi}(\rho)$ factorizes in terms of edge weights, it is straightforward to evaluate $\hat{\pi}$ up to a multiplicative constant on the set of neighbors of ρ_{old} defined in (6.1), for example, for the addition move, $\frac{\hat{\pi}(\rho_{old}\circ(i,j))}{\hat{\pi}(\rho_{old})} = w_{ij}$. Thus, one may be tempted to propose proportionally to $\hat{\pi}$ restricted on the set of allowed moves as follows

$$q_{\rho_{old}}(i,j) \;\propto\; \hat{\pi}(\rho_{new}) \qquad \text{where } \rho_{new} = \rho_{old} \circ (i,j)\,. \tag{P2}$$

Such a choice, however, does not take into account the fact that the normalizing constants of $q_{\rho_{old}}(\cdot,\cdot)$ and $q_{\rho_{new}}(\cdot,\cdot)$ differ for $\rho_{old} \neq \rho_{new}$ (Section 5.3). As a consequence, for example, detailed balance conditions, $\frac{Q^{2D}(\rho_{old},\rho_{new})}{Q^{2D}(\rho_{new},\rho_{old})} = \frac{\hat{\pi}(\rho_{new})}{\hat{\pi}(\rho_{old})}$, are not satisfied, not even approximately (see Chapter 5 for more details). A better choice for $q_{\rho_{old}}(\cdot,\cdot)$ is given by a balanced proposal (see Definition 1 in Section 5.4) such as

$$q_{\rho_{old}}(i,j) \propto \frac{\hat{\pi}\left(\rho_{new}\right)}{\hat{\pi}\left(\rho_{old}\right) + \hat{\pi}\left(\rho_{new}\right)}, \qquad \text{where } \rho_{new} = \rho_{old} \circ (i,j)\,. \tag{P3}$$

Our experiments show that the latter choice leads to a significant improvement in the mixing of the MH Markov chain compared to (P1) and (P2) (see Section 6.1.3). The main reason is that the MH algorithm induced by such proposal has a very high acceptance rate (usually above 99%) without changing the set of allowed moves. These empirical evidences are in accordance with the theoretical results of

Chapter 5.

There is a trade-off between the complexity of the proposal and the mixing obtained: a complex proposal increases the cost of each step, while a poor proposal increases the number of MCMC steps needed. We seek a compromise with good mixing properties, like (P3), while still requiring little computation at each MCMC step, like (P1). In Section 6.1.2 we derive the following proposal distribution to try to obtain such a goal:

$$q_{\rho_{old}}(i,j) \propto \begin{cases} q^{(add)}(i,j) & \text{if}(i,j) \notin \rho_{old}, \\ q^{(rem)}(i,j) & \text{if}(i,j) \in \rho_{old}. \end{cases} \tag{P4}$$

Here, $q^{(rem)}(i,j) = w_{ij}^{-1/2}$ and

$$q^{(add)}(i,j) \;=\; \sqrt{w_{ij}} \left(1 - \sum_{j' \neq j} \frac{w_{ij'} - \sqrt{w_{ij'}}}{1 + \sum_{s \neq i} w_{sj'} + \sum_l w_{il}}\right) \left(1 - \sum_{i' \neq i} \frac{w_{i'j} - \sqrt{w_{i'j}}}{1 + \sum_{s \neq j} w_{i's} + \sum_l w_{lj}}\right).$$

Note that $q^{(rem)}(i,j)$ and $q^{(add)}(i,j)$ do not depend on ρ and can be precomputed at the beginning of the MCMC run. See Section 6.1.3 for discussion of performance.

6.1.2 Derivation of (P4)

Note that in order to evaluate $\frac{\hat{\pi}(\rho \circ (i,j))}{\hat{\pi}(\rho)}$, and thus $q_\rho(i,j)$ defined in (P2) and (P3), it is not enough to know whether $(i,j) \in \rho$ or $(i,j) \notin \rho$. For example, if $\rho \circ (i,j)$ is a switch move then $\frac{\hat{\pi}(\rho \circ (i,j))}{\hat{\pi}(\rho)}$ equals $\frac{w_{ij}}{w_{i'j}}$ or $\frac{w_{ij}}{w_{ij'}}$ and so one needs to know about i' or j' respectively, where i' or j' are defined in (6.1). This increases the amount of computation needed at each MH step when using (P3) because order n values of $\frac{\hat{\pi}(\rho \circ (i,j))}{\hat{\pi}(\rho)}$ need to be updated at each step.

We want to define a modification of $q_\rho(i,j)$, say $\tilde{q}_\rho(i,j)$, that depends on ρ only through whether $(i,j) \in \rho$ or $(i,j) \notin \rho$, meaning that it can be written in the following form

$$\tilde{q}_\rho(i,j) \propto \begin{cases} q^{(add)}(i,j) & \text{if}(i,j) \notin \rho \\ q^{(rem)}(i,j) & \text{if}(i,j) \in \rho \end{cases} \tag{6.2}$$

for some $q^{(add)}(i,j)$ and $q^{(rem)}(i,j)$. This way, one can evaluate $q^{(add)}(i,j)$ and $q^{(rem)}(i,j)$ for each edge (i,j) before running the MH algorithm and then, at each MH step, one would only need to update the value of $\tilde{q}_\rho(i,j)$ for the links that have

been added or removed (at most 4) by switching from $q^{(add)}(i,j)$ to $q^{(rem)}(i,j)$ or the other way around. At the same time we want $\tilde{q}_\rho(i,j)$ to be similar to $q_\rho(i,j)$ in order to inherit some of its desirable properties in terms of acceptance rates and mixing. In order to do so we do not start from $q_\rho(i,j)$ as defined in (P3) but instead from $q_\rho(i,j) \propto \sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}$. This choice is still a balanced proposal (see Chapter 5) and has similar mixing properties to (P3) (for example it satisfies detailed balance conditions in the asymptotic regime), while it allows some simplifications in the calculations below that would be less easy with (P3).

Given $q_\rho(i,j) \propto \sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}$ and (6.2) a natural choice for $q^{(add)}$ and $q^{(rem)}$ is $\mathbb{E}\left[\sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}\middle|(i,j)\notin\rho\right]$ and $\mathbb{E}\left[\sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}\middle|(i,j)\in\rho\right]$ respectively, where the expectations are taken over $\rho\sim\hat{\pi}$. If $(i,j)\in\rho$ then $\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}$ equals $\frac{1}{w_{ij}}$ regardless of ρ and therefore we have

$$q^{(rem)}(i,j) \;=\; \mathbb{E}\left[\sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}\middle|(i,j)\in\rho\right] \;=\; w_{ij}^{-1/2}\,. \tag{6.3}$$

Note that if $(i,j)\in\rho$ then $w_{ij}>0$ almost surely and so $q^{(rem)}(i,j)$ is well-defined.

On the other hand if $(i,j)\notin\rho$ then $\sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}$ can have different values depending on ρ and we cannot compute $\mathbb{E}\left[\sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}\middle|(i,j)\notin\rho\right]$ in closed form. Thus approximations are needed. First we fix $(i,j)\in\{1,\dots,n_1\}\times\{1,\dots,n_2\}$ and we define the following probabilities:

$$\begin{aligned}
p_i^{(r)} &= \Pr\left((i,j')\notin\rho\;\forall j'=1,\dots,n_2\middle|(i,j)\notin\rho\right)\,,\\
p_j^{(r)} &= \Pr\left((i',j)\notin\rho\;\forall i'=1,\dots,n_1\middle|(i,j)\notin\rho\right)\,,\\
p_{i'j'} &= \Pr\left((i',j')\in\rho\middle|(i,j)\notin\rho\right)\,.
\end{aligned}$$

Then we use the following approximation:

$$\begin{aligned}
\mathbb{E}\left[\sqrt{\frac{\hat{\pi}(\rho\circ(i,j))}{\hat{\pi}(\rho)}}\middle|(i,j)\notin\rho\right] \;\approx\; & p_i^{(r)}p_j^{(b)}\sqrt{w_{ij}} + \\
& p_j^{(b)}\sum_{j'\neq j}\sqrt{\frac{w_{ij}}{w_{ij'}}}p_{ij'} \;+\; p_i^{(r)}\sum_{i'\neq i}\sqrt{\frac{w_{ij}}{w_{i'j}}}p_{i'j} \;+\; \sum_{i'\neq i}\sum_{j'\neq j}\sqrt{\frac{w_{ij}w_{i'j'}}{w_{i'j}w_{ij'}}}p_{ij'}p_{i'j}\,.
\end{aligned} \tag{6.4}$$

Equation (6.4) is an approximation because it factorizes probabilities of non-independent events, like $(i,j')\in\rho$ and $(i',j)\in\rho$.

Then we introduce a further approximation by dropping the terms $w_{i'j'}$ on

the right-hand side of (6.4), which becomes

$$\sqrt{w_{ij}}\left(p_i^{(r)}+\sum_{j'\neq j}\frac{p_{ij'}}{\sqrt{w_{ij'}}}\right)\left(p_j^{(b)}+\sum_{i'\neq i}\frac{p_{i'j}}{\sqrt{w_{i'j}}}\right) =$$
$$\sqrt{w_{ij}}\left(1-\sum_{j'\neq j}p_{ij'}+\sum_{j'\neq j}\frac{p_{ij'}}{\sqrt{w_{ij'}}}\right)\left(1-\sum_{i'\neq i}p_{i'j}+\sum_{i'\neq i}\frac{p_{i'j}}{\sqrt{w_{i'j}}}\right) =$$
$$\sqrt{w_{ij}}\left(1-\sum_{j'\neq j}p_{ij'}\left(\frac{\sqrt{w_{ij'}}-1}{\sqrt{w_{ij'}}}\right)\right)\left(1-\sum_{i'\neq i}p_{i'j}\left(\frac{\sqrt{w_{i'j}}-1}{\sqrt{w_{i'j}}}\right)\right). \quad (6.5)$$

Finally by approximating $p_{ij'}$ with the quantity $\frac{w_{ij'}}{1+\sum_{s\neq i}w_{sj'}+\sum_l w_{il}}$ and similarly $p_{i'j}$ with $\frac{w_{i'j}}{1+\sum_{s\neq j}w_{i's}+\sum_l w_{lj}}$ we obtain

$$q^{(add)}(i,j) = \sqrt{w_{ij}}\left(1-\sum_{j'\neq j}\frac{w_{ij'}-\sqrt{w_{ij'}}}{1+\sum_{s\neq i}w_{sj'}+\sum_l w_{il}}\right)$$
$$\left(1-\sum_{i'\neq i}\frac{w_{i'j}-\sqrt{w_{i'j}}}{1+\sum_{s\neq j}w_{i's}+\sum_l w_{lj}}\right),$$

which is the expression used in (P4).

6.1.3 Convergence Diagnostics

We used various convergence diagnostic techniques in order to assess the reliability of our algorithm, to indicate the number of iterations needed, and to compare the efficiency of the four proposals (P1)-(P4) of Section 6.1.1. We demonstrate these techniques on the posterior $\pi(\rho|\sigma,\mathbf{p}^{(c)},\lambda,\mathbf{x})$ with $k=2$, $\sigma=0.3$, $p_1^{(c)}=p_2^{(c)}=0.5$, $\lambda=50$ and the center intensity $g(\cdot)$ being the uniform measure over $W=[0,10]\times[0,10]$. Here $\mathbf{x}$ is a synthetic sample of 44 red and 47 blue points generated according to the model just defined, see Figure 6.3 (a). We set the threshold δ of (P1) to 0.001. The R code used to produce the results presented in this Section is available at `https://sites.google.com/site/gzanellawebpage/compclust_supp_f.zip`.

We first performed some qualitative output analysis by looking at summary plots of the MCMC samples of the partition, as the one in Figure 6.3(a). Such plots can be helpful to spot when mixing has not yet occurred (see Section 6.1.4).

Secondly we considered different real valued summary statistics of the chain state (typically the number of different edges from some fixed reference matching).

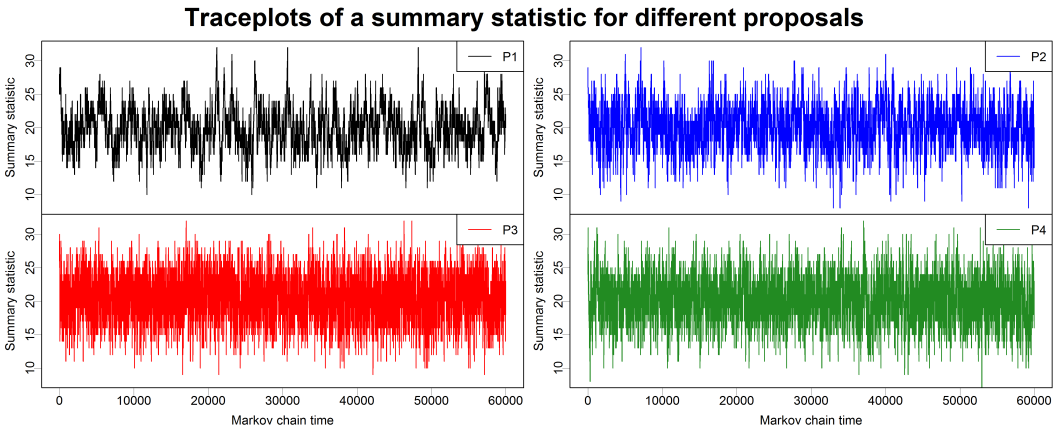


Figure 6.2: Traceplots of the number of differences from a reference matching.

We plotted time series (see Figure 6.2) and empirical distributions of such real valued functions for different runs of the MCMC starting from different configurations. We estimated the autocorrelation functions (Figure 6.3(b)), the Integrated Autocorrelation Time (IAT) and the Effective Sample Size (ESS) of such real-valued time series using the *R* package *coda* (see Plummer *et al.* , 2005) in order to compare different versions of the algorithm (see Table 6.1). See Section 5.1.2 for definitions of IAT and ESS.

Thirdly we used some standard convergence diagnostic techniques (see Brooks & Roberts (1998) and Cowles & Carlin (1996) for an overview of the techniques available). In particular we used the multivariate version of Gelman and Rubin's diagnostic (see Gelman & Rubin (1992) and Brooks & Gelman, 1998). Figure 6.3(d) shows the results obtained by using a 10-dimensional summary statistic of ρ. Note that in this context univariate summary statistics are not sufficiently informative and therefore misleading results can be obtained if these are used as the sole basis for convergence diagnostics.

Finally we compared two independent runs of the algorithm (with different starting states) by looking at estimates of the association probabilities $p_{ij} = \Pr\big((i,j) \in \rho\big)$ with $\rho \sim \hat{\pi}$. We consider the measure of proximity

$$D = \sup_{(i,j)\in E} |\hat{p}_{ij}^{(1)} - \hat{p}_{ij}^{(2)}|\,, \tag{6.6}$$

where $\hat{p}_{ij}^{(1)}$ and $\hat{p}_{ij}^{(2)}$ denote the proportion of time that (i,j) was present in the two MCMC runs. As starting states we considered the empty matching (each point is a cluster by itself), the posterior mode (obtained with the Hungarian algorithm) and matchings obtained as the output of the MCMC itself. Since equation (6.6) considers

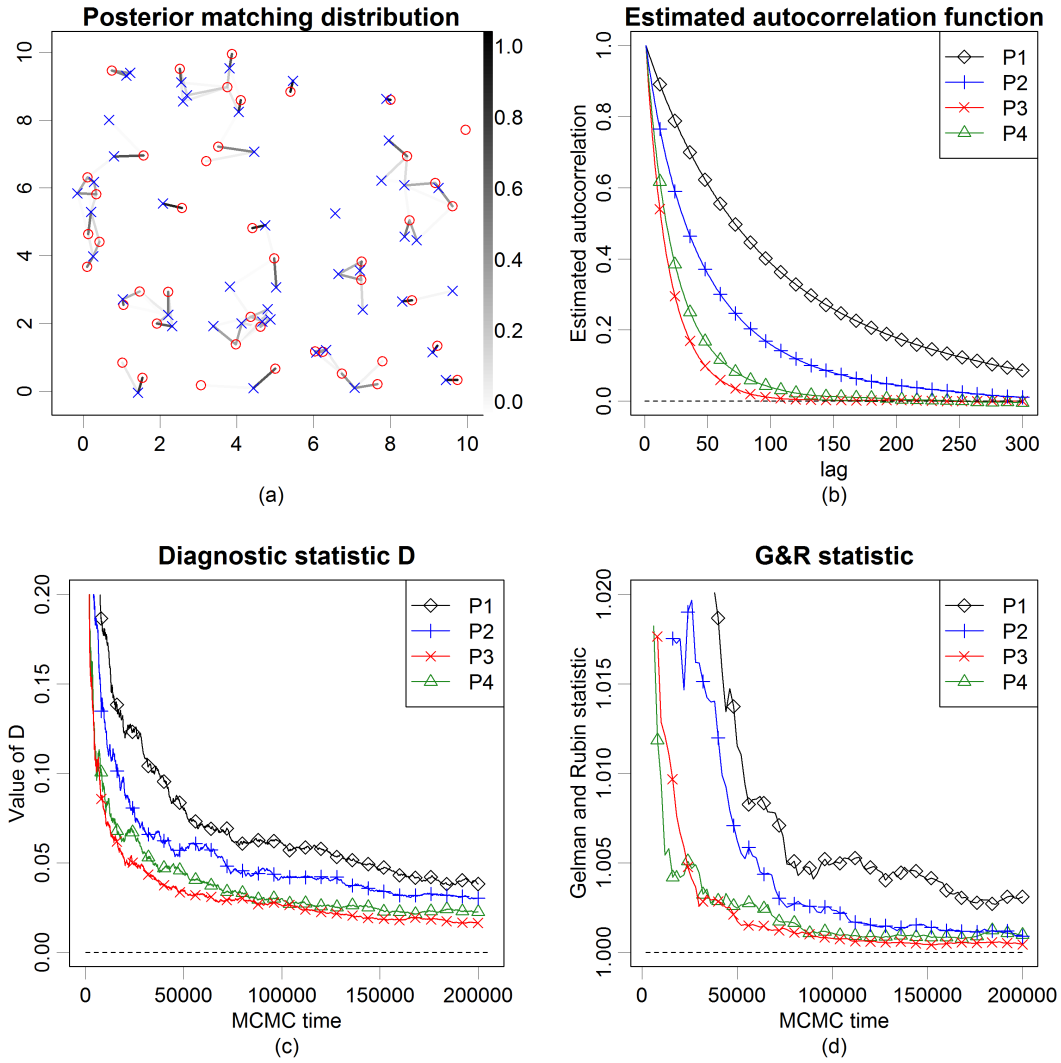


Figure 6.3: The results of the diagnostic performed with four convergence diagnostic techniques described in Section 6.1.3.

each link individually, we expect the resulting convergence diagnostic indicator D to be more severe than the ones obtained from one or few summary statistics. Results are shown in Figure 6.3(d).

None of the convergence diagnostics presented indicate convergence issues except in the complete matching case (when the parameter $p_1^{(c)}$ is equal or very close to 0), which is considered in the next section.

All convergence diagnostic techniques agree in indicating that proposal (P3) gives the best mixing; however in terms of real computation time the most efficient proposal is (P4). Note that such performances depend on the measure being targeted and, when running time is considered, on the computer implementation of such

	mean acc.rate	Estimated IAT	ESS for 10^4 steps [for 1 sec]	steps [sec] to $D < .05$	steps [sec] to $GR < .005$
P1	17%	206	262 [270]	1.4e05 [7.3]	7.6e04 [13.5]
P2	41%	108	544 [40]	7.1e04 [84.6]	6.2e04 [97]
P3	97%	40	1358 [99]	2.0e04 [32.7]	2.4e04 [27.3]
P4	68%	55	1038 [747]	3.4e04 [2.2]	1.6e04 [4.8]

Table 6.1: Performances of the four proposals of Section 6.1.1 tested on the configuration in Figure 6.3(a) and averaged over 5 independent runs for each proposal. GR denotes the multivariate Gelman and Rubin statistic (potential scale reduction factor). See Section 6.1.3 for more details. The running time indicated in brackets is evaluated using R software on a desktop computer with Intel $i7$-2600 processor, 3.40GHz CPU and 16GB of RAM.

proposals. For the case considered in this section, proposal (P4) gives a 3-4 times speed-up over the commonly used choice (P1). Depending on the configuration such speed-up may vary. According to our experiments, for "flatter" distributions (e.g. increasing σ to 1 and $p_1^{(c)}$ to 0.9, while keeping the other parameters unchanged) the speed-up almost disappears, while for "rougher" distributions (e.g. decreasing both σ and $p_1^{(c)}$ to 0.1, while keeping the other parameters unchanged) the speed-up increases and (P4) can be to 10 times faster than (P1). Moreover note that (P1) introduces an approximation in $\hat{\pi}(\rho)$ by using the truncation procedure, while (P4) does not.

6.1.4 Multimodality and Simulated Tempering

In the complete matching case the posterior distribution of ρ presents a strongly multimodal behavior. Cycle-like configurations like the one in Figure 6.4(a) are local maxima for $\hat{\pi}(\rho)$. In fact in order to reach a higher probability configuration (i.e. shorter links) from such a "cycle" configuration, with the set of allowed moves defined by (6.1), the chain needs to pass through lower probability configurations (i.e. longer links). If we consider extreme cycle-like configurations such as the one in Figure 6.4(b), then the MCMC run will typically to get stuck in such local maxima. In order to overcome this potential multimodality problem we implemented a simulated tempered version of our MCMC algorithm. The basic version of simulated tempering methods consist of running an MCMC in an extended space containing additional "tempered" copies of the target measure, which work as "bridges" between local modes. See for example Geyer & Thompson (1995) or Marinari & Parisi (1992) for more details.

This technique manages to overcome local maxima for the complete match-

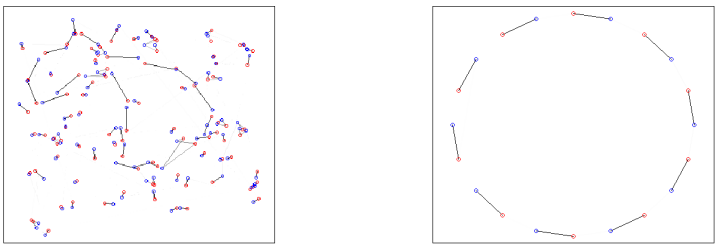


Figure 6.4: Configurations corresponding to local maxima of $\hat{\pi}(\rho)$ for (a) a synthetic sample and (b) an artificially designed configuration.

ing case even when extreme cycle-like configurations are present as in Figure 6.4(b). Nevertheless our specific application do not present a complete matching case and therefore we have a milder multimodality and the MCMC algorithm exhibits sufficient mixing without the use of simulated tempering. In this case the "noise" (i.e. unlinked points) present in the data has a smoothing effect on the posterior distribution, thus helping the algorithm to traverse the space. Therefore Simulated Tempering is not used for the real data analysis, as convergence diagnostic tools do not show suspicious behavior.

We note that Dellaert *et al.* (2003) deal with multimodality in a similar sample space (made of perfect matchings in a bipartite graph) arising from the Structure from Motion Problem. In order to allow the MH algorithm to overcome local maxima like the one in Figure 6.4(b) they allow the MH proposal to include "long" moves which they call "chain flipping".

6.1.5 Scaling the proposal and parallel computation

When using the MH algorithm on continuous sample spaces one can usually tune the variance of the proposal distribution to improve the efficiency of the algorithm (see for example Roberts *et al.* , 1997). Given the very high acceptance rate obtained by proposing according to (P3) it is natural to consider the possibility of scaling such a proposal in order to obtain longer-scale moves. The scaling problem for MH algorithms in discrete contexts has been considered, for example, in Roberts (1998). In that case the sample space under consideration was $\{0,1\}^N$, the vertices of the N-dimensional hypercube, and the scaling parameter, say l, was a positive integer representing the number of randomly-chosen bits to be flipped at any given proposal.

Unfortunately, because of the nature of our sample space, it is not so straightforward to scale the proposal distribution $Q^{2D}(\rho_{old}, \rho_{new})$. One possibility is to scale by choosing l edges, $\{(i_h, j_h)\}_{h=1}^{l}$, and performing l moves defined in (6.1), proposing

$\rho_{new} = \rho_{old} \circ (i_1, j_1) \circ \cdots \circ (i_l, j_l)$. However the l moves corresponding to $\{(i_h, j_h)\}_{h=1}^{l}$ cannot be performed independently: consider, for example, the case where i_1 equals i_2. Therefore we would then have to perform l moves sequentially, at a computational cost being roughly l times the one of a single move. Scaling the proposal in such a way does not seem to be effective.

Instead, if the l moves could be performed independently, it would be possible to implement a multiple proposal scheme using parallel computation, thus leading to a significant computational gain. This can be obtained by considering an approximation of our model, where points at a distance greater than or equal to some r_{max} have probability 0 of being in the same cluster. The latter procedure is equivalent to the truncation procedure mentioned in Section 6.1.1 and can be viewed as coming from the use of truncated Gaussian distributions to model point distributions within clusters, see (3.1). Using this truncated model and dividing the observed region into a grid, we defined a multiple proposal scheme where the l moves are proposed and accepted/rejected simultaneously and independently. Therefore, at each MH step, such l moves can be performed in an embarrassingly parallel fashion, meaning that they can be performed without the need for any communication between them. In Section 6.1.6 we give more details on the implementation and we show that in practice the mixing of the resulting MH algorithm improves by a factor roughly equal to l itself (note that the maximum value of l is bounded above, in a way that depends on r_{max} and the size of the observation region W). A parallel-computing implementation of this algorithm would offer significant speed-ups (we anticipate speed-ups by a factor of around 8 for our dataset, see Section 6.1.6). Such speed-ups would increase with the size of the dataset and window, making this proposal scheme especially relevant for applications to very large datasets. In Section 6.1.6 this scheme is presented and tested for fixed σ. In case σ is varying, one can either require an upper bound on σ, or generate different square grids for different values of σ.

6.1.6 Multiple proposal scheme implementation

In this section we describe in more detail the multiple proposal scheme described in Section 6.1.5. First we define the square grid and the transition kernel we use, then we show that the corresponding Markov chain is targeting the correct measure and finally we test the performances on synthetic samples.

Defining the square grid

Suppose that we observe a bivariate point pattern $\mathbf{x}$ in a square window $W = [0, a] \times [0, a]$, with $a > 0$ (otherwise consider a square containing the observed window). In the spirit of Besag's coding method (Besag, 1974), we first divide the window according to a grid of squares of side of length (at least) $2r_{max}$ like in Figure 6.5 (left). Then, we divide the squares into 4 groups, in order to have no adjacent (nor corner adjacent) squares in the same group. Say for simplicity that we have l squares for each group from 1 to 4. We denote the squares as $\left\{S_s^g\right\}_{s=1,\dots,l}^{g=1,2,3,4}$, where the superscript denotes the group and the subscript the square in the group (see Figure 6.5, left).

Defining the transition step

Each step of the multiple proposal scheme works as follows:

1. Choose an index g uniformly at random from $\{1, 2, 3, 4\}$;
2. For s running from 1 to l:
   (a) Define R_s^g as the set of all the red points inside S_s^g;
   (b) Define B_s^g as the set of blue points inside

$$S_s^g \oplus r_{max} = \{x \in W \,:\, |x - y| \leq r_{max} \text{ for some } y \in S_s^g\},$$

   that are *not* linked to any red point in a square different from S_s^g (see the right-hand side of Figure 6.5);
   (c) Choose a red-blue couple (i, j) uniformly at random from $R_s^g \times B_s^g$;
   (d) Propose to move to $\rho_{new} = \rho_{old} \circ (i, j)$ and accept the move with probability $1 \wedge \frac{\hat{\pi}(\rho_{new})}{\hat{\pi}(\rho_{old})}$.

Note that, since we are using truncation, only points closer than r_{max} can be linked.

For simplicity, in step 2(c) we considered (i, j) to be chosen uniformly at random from $R_s^g \times B_s^g$. The extension to a general proposal $q_{\rho_{old}}(i, j)$ like in Section 6.1.1 is straightforward: one simply needs to take into account for the proposal in the MH acceptance probability in step 2(d).

Note that, since the target measure $\hat{\pi}$ factorizes, see equation (3.14), the l different MH steps in step 2 of such multiple proposal scheme can be implemented in an embarrassingly parallel fashion, meaning that they can be performed without the

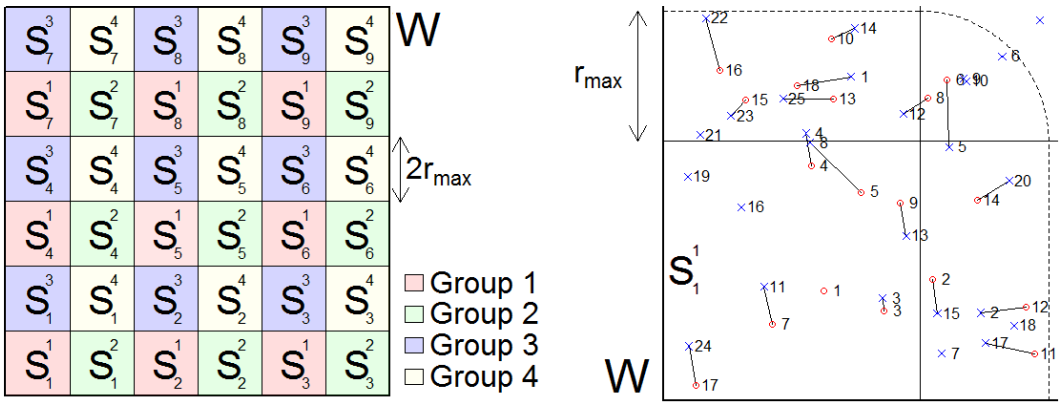


Figure 6.5: Left: the observed window W divided into squares. Right: in this case $R_1^1 = \{1, 3, 4, 5, 7, 9, 17\}$ and $B_1^1 = \{3, 4, 6, 7, 8, 9, 10, 11, 13, 16, 18, 19, 21, 24\}$.

need of any communication between them. In fact it is easy to see that such l moves involve separate subgraphs of the original bipartite graph and, since $\hat{\pi}$ factorizes, the l acceptance-rejection steps are independent.

Showing the correctness of the induced MCMC

Before testing such scheme on a synthetic sample we need to show that the induced Markov chain is ergodic with stationary distribution $\hat{\pi}$. This is not obvious and indeed a careful choice of the sets R_s^g and B_s^g, like the one in step $2(a)$-(b), is necessary for all the moves to be reversible and for the proposal distribution to be symmetric. Note that the definitions of R_s^g and B_s^g allow for links across squares to be modified.

Lemma 2. *The Markov Chain induced by the multiple proposal scheme is an ergodic Markov chain with stationary distribution $\hat{\pi}$.*

Proof. The transition kernel P induced by the multiple proposal scheme can be seen as a mixture of 4 cycles of l transition kernels: $P = \frac{1}{4}\big(P_1^1 \cdots P_l^1 + P_1^2 \cdots P_l^2 + P_1^3 \cdots P_l^3 + P_1^4 \cdots P_l^4\big)$, where P_s^g is the MH transition kernel with proposal Q_s^g given by steps $2(a) - (d)$ for fixed $s \in \{1, \ldots, l\}$ and $g \in \{1, 2, 3, 4\}$. If each P_s^g satisfies detailed balance conditions with respect to $\hat{\pi}$, then it follows that P satisfies them too. We need to show that, for any s and g, $Q_s^g(\rho_{old}, \rho_{new}) = Q_s^g(\rho_{new}, \rho_{old})$ for any couple of matchings (ρ_{old}, ρ_{new}), from which it follows that $1 \wedge \frac{\hat{\pi}(\rho_{new})}{\hat{\pi}(\rho_{old})}$ is the correct MH acceptance probability.

The probability of choosing a certain couple $(i, j) \in R_s^g \times B_s^g$ in step $2(c)$ is $\frac{1}{|R_s^g||B_s^g|}$. Note that the set R_s^g does not depend on ρ_{old}. On the other hand

the set B_s^g does depend on ρ_{old}, but it does not change for $\rho_{new} = \rho_{old} \circ (i,j)$ with $(i,j) \in R_s^g \times B_s^g$. Therefore, when the current matching becomes ρ_{new} the probability of choosing (i,j) remains $\frac{1}{|R_s^g||B_s^g|}$ with the same R_s^g and B_s^g.
Let $\rho_{new} = \rho_{old} \circ (i,j)$, with $(i,j) \in R_s^g \times B_s^g$ (otherwise $Q_s^g(\rho_{old}, \rho_{new})$ is clearly 0). Let us first consider the case where $\rho_{old} \circ (i,j)$ is an addition or deletion move, see equation (6.1). In this case the only way to propose moving from ρ_{old} to ρ_{new} (and back from ρ_{new} to ρ_{old}) with Q_s^g is by choosing the red-blue couple (i,j) in step 2(c). Therefore $Q_s^g(\rho_{old}, \rho_{new}) = Q_s^g(\rho_{new}, \rho_{old}) = \frac{1}{|R_s^g||B_s^g|}$. If $\rho_{old} \circ (i,j)$ is a switch move then the only way to propose to move from ρ_{old} to ρ_{new} is by choosing the red-blue couple (i,j), while the only way to propose to move back from ρ_{new} to ρ_{old} is by choosing either the couple (i',j) or the couple (i,j'), depending on whether $\rho_{old} \circ (i,j)$ equals $\rho_{old} - (i',j) + (i,j)$ or $\rho_{old} - (i,j') + (i,j)$, respectively. In the first case $(i',j) \in \rho_{old}$ and $j \in B_s^g$ and thus, by definition of B_s^g we have $i \in R_s^g$. In the second case, since $(i,j') \in \rho_{old}$, the j'-th blue point has a distance smaller than r_{max} from S_s^g. Therefore, since $(i,j') \in \rho_{old}$, we have $j' \in B_s^g$. Therefore, since $(i',j) \in R_s^g \times B_s^g$, or $(i,j') \in R_s^g \times B_s^g$ respectively, $Q_s^g(\rho_{old}, \rho_{new}) = Q_s^g(\rho_{new}, \rho_{old}) = \frac{1}{|R_s^g||B_s^g|}$. Finally, if $\rho_{old} \circ (i,j)$ is a double-switch move then there are respectively two ways to propose to move from ρ_{old} to ρ_{new} (choosing (i,j) and (i',j') in step 2(c)) and two ways to propose to move back from ρ_{new} to ρ_{old} (choosing (i,j') and (i',j)). Similarly to the switch move one can show that $i' \in R_s^g$ and $j' \in B_s^g$. Therefore $Q_s^g(\rho_{old}, \rho_{new}) = Q_s^g(\rho_{new}, \rho_{old}) = \frac{2}{|R_s^g||B_s^g|}$.

The desired ergodicity follows from the fact that P is an aperiodic and irreducible Markov transition kernel on a finite state space, satisfying detailed balance conditions with respect to $\hat{\pi}$ (see Theorem 2 of Chapter 5). □

Demonstration of performance on a synthetic sample

We test the multiple proposal scheme on the posterior $\pi(\rho|\sigma, \mathbf{p}^{(c)}, \lambda, \mathbf{x})$ given by the Random Partition Model with $k = 2$, $\sigma = 0.3$, $p_1^{(c)} = p_2^{(c)} = 0.5$, $\lambda = 400$ and the center intensity $g(\cdot)$ being uniform over a window $W = [0,20] \times [0,20]$. Here, $\mathbf{x}$ is a synthetic sample generated according to the model we just specified. The sample $\mathbf{x}$ is made of 310 red and 316 blue points (see Figure 6.6). We consider three cases, $l = 1, 4, 9$, in order to show that the mixing of the MH Markov chain improves roughly at rate equal to l. We use the convergence diagnostic techniques presented in Section 6.1.3. The results are shown in Figure 6.6 and Table 6.2.

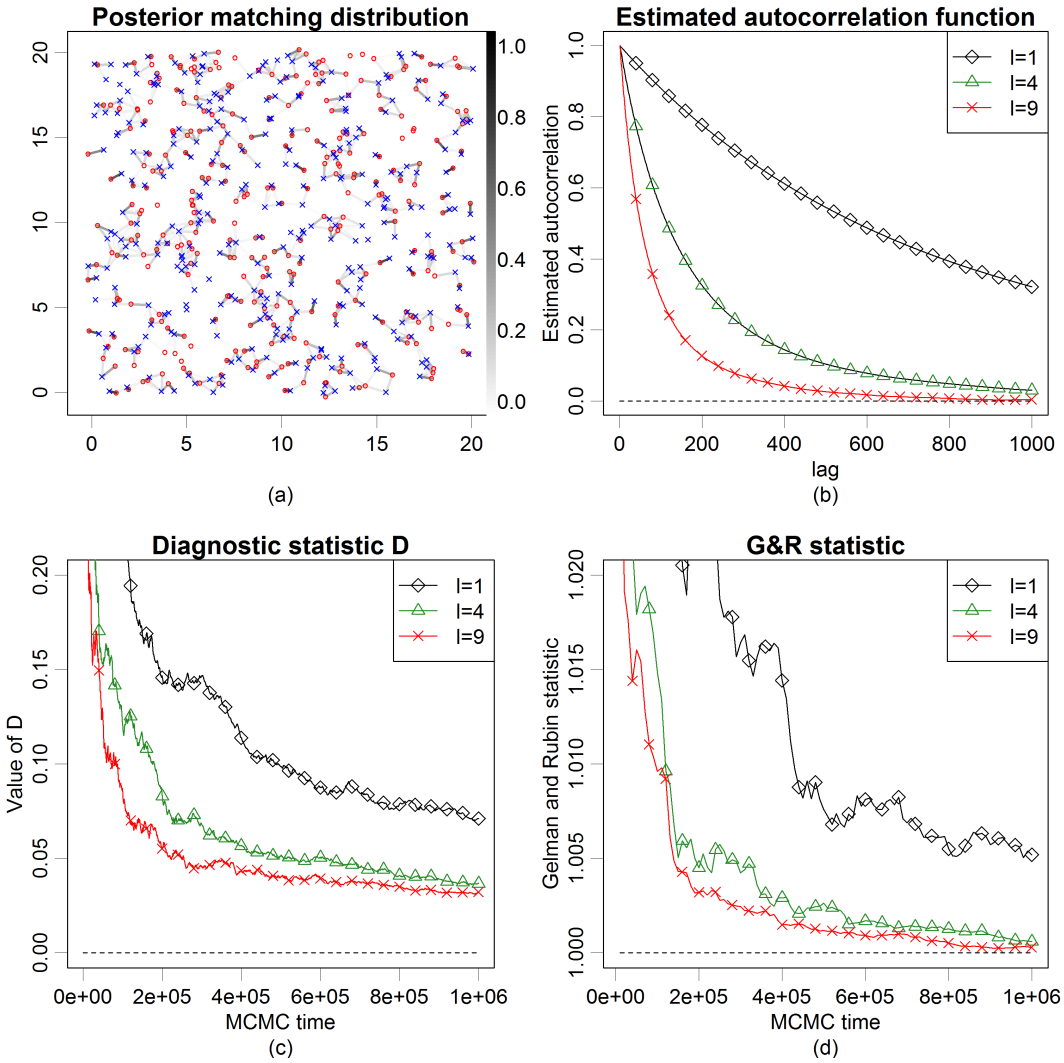


Figure 6.6: Comparison of the multiple proposal scheme for $l = 1, 4, 9$ using the convergence diagnostic techniques of Section 6.1.3.

An approximate prediction of the speed-up for the real data

Given the historical context (groups of settlements interacting from the administrative and political point of view), two settlements in the same administrative cluster should be close enough to allow inhabitants to walk between them, spend time conducting business, and then return in a single day (for example, 3 hours outbound and 3 hours inbound). In fact, the historians involved in the project consider it to be implausible for two settlements in the same cluster to be separated by a distance greater than 15 km. When analyzing the settlements dataset, a reasonable choice of

	Estimated IAT	ESS for 10^4 steps	steps to reach $D < .1$	steps to reach reach $GR < .01$
$l = 1$	2030	32	4.9e05	4.3e05
$l = 4$	428	161	1.7e05	1.2e05
$l = 9$	193	262	7e04	1e05

Table 6.2: Performance of the multiple proposal scheme for $l = 1, 4, 9$ on the configuration in Figure 6.6(a) averaged over 5 independent runs for each value of l. GR denotes the multivariate Gelman and Rubin statistic (potential scale reduction factor).

r_{max} could be 20 km (increasing on the upper bound given by the historians in order to have additional confidence of not imposing conditions that are too restrictive).

The area over which we observe settlements is roughly 53 000 km. Therefore if we were to divide that area in squares of side $2r_{max} = 40$ km we would obtain approximately $\frac{53\,000}{40^2} \approx 33$ different squares. Therefore, we would have approximately 8 squares for each group, i.e. $l = 8$. Hence, given the results above, it is reasonable to expect a parallel implementation of such a scheme to yield approximately an 8-fold speed-up of the MH Markov Chain.

6.2 The k-color case

We now define an MCMC algorithm that targets $\hat{\pi}(\rho)$ in the general case, $k \geq 3$. This case is harder than the two-dimensional one because it involves clusters with different dimensions and not just pairwise interaction.

Description of proposed Gibbs projection MCMC algorithm

We define the transition kernel P of our MCMC algorithm as a mixture of $\binom{k}{\lfloor k/2 \rfloor}$ MH transition kernels, each corresponding to a group A of $\lfloor k/2 \rfloor$ colors:

$$P(\rho_{old}, \rho_{new}) = \binom{k}{\lfloor k/2 \rfloor}^{-1} \sum_{A \subset \{1,\dots,k\}, |A| = \lfloor k/2 \rfloor} P^{(A)}(\rho_{old}, \rho_{new})\,. \tag{6.7}$$

Here, $\lfloor k/2 \rfloor$ denotes the integer part of $k/2$, and $\binom{k}{\lfloor k/2 \rfloor}$ denotes the binomial coefficient. The kernel $P(\cdot, \cdot)$ of (6.7) selects a set of colors A, "projects" the k-color configuration to a 2-color configuration where the new two colors correspond to A and $A^c = \{1, \dots, k\} \backslash A$, and then acts on the two-color configuration. More precisely, the action of $P^{(A)}$ is the following (see Figure 6.7):

1. Reduce the k-color configuration $(\mathbf{x}, \rho_{old})$ to a two-color configuration $(\mathbf{x}^{2D}, \rho^{2D}_{old})$ by replacing the points having colors in A and A^c respectively with their cluster centroids. Denote by d_i the number of points merged together into the i-th point x^{2D}_i.

2. Obtain ρ^{2D}_{new} from $(\mathbf{x}^{2D}, \rho^{2D}_{old})$ with one or more MH moves using the proposal Q^{2D} of Section 6.1 with the target measure being the two-dimensional version of $\hat{\pi}$, $\hat{\pi}^{2D}$ (modified to take account of the multiplicity of the points d_i, see Section 6.2.1).

3. Obtain the k-color configuration $(\mathbf{x}, \rho_{new})$ from $(\mathbf{x}^{2D}, \rho^{2D}_{new})$ by the inverse operation of Step 1 (note that here one needs to know what A is).

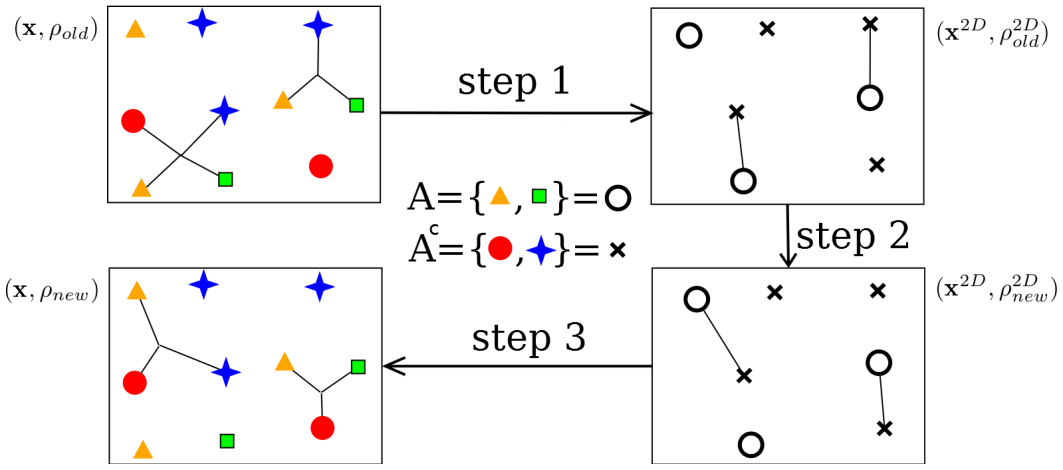


Figure 6.7: The action of a transition kernel $P^{(A)}$ for a given A.

We denote the two-color configuration and the corresponding partition constructed according to Step 1 of P^A by

$$\mathbf{x}^{2D} = \left((x^{2D}_1, m^{2D}_1, d_1), ..., (x^{2D}_{n^{2D}}, m^{2D}_{n^{2D}}, d_{n^{2D}})\right) \quad \text{and} \quad \rho^{2D} = \left(C^{2D}_1, ..., C^{2D}_{N(\rho^{2D})}\right)$$

respectively. The measure $\hat{\pi}^{2D}$ targeted in Step 2 is defined as

$$\hat{\pi}^{2D}(\rho^{2D}) = \pi^{2D}(\rho^{2D} \mid \mathbf{x}^{2D}, \sigma, \lambda, \mathbf{p}^{(c)}) \propto \prod_{j=1}^{N(\rho^{2D})} \left(\frac{g\left(\overline{x}_{C^{2D}_j}\right) \lambda\, p^{(c)}_{s^{2D}_j}}{c_{s^{2D}_j}} \exp\left(-\frac{\pi \delta^2_{C^{2D}_j}}{2\sigma^2}\right) \prod_{i,l \in C^{2D}_j,\, i \neq l} \mathbb{1}(m^{2D}_i \neq m^{2D}_l) \right), \quad (6.8)$$

where the modified multiplicities, barycenters and intra-cluster square distances are defined as $s_j^{2D} = \sum_{i \in C_j^{2D}} d_i$, $\overline{x}_{C_j^{2D}} = \frac{\sum_{i \in C_j^{2D}} d_i x_i^{2D}}{\sum_{i \in C_j^{2D}} d_i}$ and $\delta^2_{C_j^{2D}} = \sum_{i \in C_j^{2D}} d_i (x_i^{2D} - \overline{x}_{C_j^{2D}})^2$, respectively. In order for this algorithm to be correct, $\hat{\pi}^{2D}$ must be proportional to $\hat{\pi}$ on the collection of possible moves of $P^{(A)}$, so that $P^{(A)}$ satisfy detailed balance conditions with respect to $\hat{\pi}$. This follows from basic properties of the Gaussian density function and is proven in Section 6.2.1. Therefore no additional accept/reject mechanism is needed at Step 3 of P^A.

Remark 17. *The distribution $\hat{\pi}^{2D}(\rho^{2D})$ given in* (6.8) *can be expressed as a probability distribution on the space of matchings contained in a weighted bipartite graph where the probability of each matching ρ^{2D} is proportional to its total weight. Namely, $\hat{\pi}^{2D}(\rho^{2D}) \propto \prod_{(i,j) \in \rho^{2D}} w_{ij}^{2D}$ for some suitably defined weights w_{ij}^{2D} given by* (6.8) *and depending only on $\boldsymbol{x}^{2D}$. Therefore, $\hat{\pi}^{2D}(\rho^{2D})$ is of the same form as the two-color version of $\hat{\pi}(\rho)$ (see e.g. Section 4.1.1). This is a useful property of the projection scheme (see Section 6.2.2).*

Note that, when k is even, $P^{(A)}$ is the same transition kernel as $P^{(A^c)}$. This is not an issue and it is indeed equivalent to never using $P^{(A^c)}$ and using $P^{(A)}$ twice as often.

6.2.1 Correctness of the k-dimensional algorithm

We need to prove that $\hat{\pi}^{2D}$ is proportional to $\hat{\pi}$ on the collection of possible moves of $P^{(A)}$. First we need the following Lemma.

Lemma 3. *For any $x_1, ..., x_s, z \in \mathbb{R}^n$, let $\overline{x} = s^{-1} \sum_{i=1}^s x_i$. It holds that*

$$\sum_{i=1}^{s} \big(x_i - z\big)^2 = \sum_{i=1}^{s} \big(x_i - \overline{x}\big)^2 + s(\overline{x} - z)^2.$$

Proof. Given $x_1, \ldots, x_s, z$ and $\overline{x}$ as above it holds

$$\begin{aligned}
\sum_{i=1}^{s}(x_i - z)^2 &= \sum_{i=1}^{s}(x_i - \overline{x} + \overline{x} - z)^2 = \\
&= \sum_{i=1}^{s}\Big((x_i - \overline{x})^2 + (\overline{x} - z)^2 + (x_i - \overline{x})(\overline{x} - z)\Big) = \\
&= \sum_{i=1}^{s}(x_i - \overline{x})^2 + s(\overline{x} - z)^2 + 2(\overline{x} - z)\sum_{i=1}^{s}(x_i - \overline{x}) = \sum_{i=1}^{s}(x_i - \overline{x})^2 + s(\overline{x} - z)^2.
\end{aligned}$$

□

Let C_j be a cluster of $(\mathbf{x}, \rho)$ and C_j^{2D} be the corresponding cluster in the projected two-color configuration $(\mathbf{x}^{2D}, \rho^{2D})$. We define $C_j^A = \{i \in C_j : m_i \in A\}$, $s_j^A = \#\{i \in C_j : m_i \in A\}$ and $\overline{x}_{C_j^A} = \frac{\sum_{i \in C_j^A} x_i}{s_j^A}$. Furthermore, $C_j^{A^c}$, $s_j^{A^c}$ and $\overline{x}_{C_j^{A^c}}$ are defined analogously for A^c. Then, it follows from Lemma 3 that

$$\begin{aligned}\delta_{C_j}^2 = \sum_{i \in C_j} (x_i - \overline{x}_{C_j})^2 = \sum_{i \in C_j^A} (x_i - \overline{x}_{C_j})^2 + \sum_{i \in C_j^{A^c}} (x_i - \overline{x}_{C_j})^2 = \\ \sum_{i \in C_j^A} \left(x_i - \overline{x}_{C_j^A}\right)^2 + s_j^A \left(\overline{x}_{C_j^A} - \overline{x}_{C_j}\right)^2 + \\ + \sum_{i \in C_j^{A^c}} \left(x_i - \overline{x}_{C_j^{A^c}}\right)^2 + s_j^{A^c} \left(\overline{x}_{C_j^{A^c}} - \overline{x}_{C_j}\right)^2 = \\ \sum_{i \in C_j^A} \left(x_i - \overline{x}_{C_j^A}\right)^2 + \sum_{i \in C_j^{A^c}} \left(x_i - \overline{x}_{C_j^{A^c}}\right)^2 + \delta_{C_j^{2D}}^2. \quad (6.9)\end{aligned}$$

From (6.8), (6.9) and equation (3.14) it follows that

$$\hat{\pi}(\rho) \;=\; \hat{\pi}^{2D}(\rho^{2D}) \cdot \exp\left\{ \sum_{j=1}^{N(\rho)} \left(\sum_{i \in C_j^A} \left(x_i - \overline{x}_{C_j^A}\right)^2 + \sum_{i \in C_j^{A^c}} \left(x_i - \overline{x}_{C_j^{A^c}}\right)^2 \right) \right\}. \quad (6.10)$$

The second factor on the right-hand side of (6.10) is constant with respect to the action of $P^{(A)}$. It follows that $\hat{\pi^{2D}}$ is proportional to $\hat{\pi}$ on the set of allowed moves of $P^{(A)}$, as desired.

6.2.2 Discussion of the projection scheme

By merging colors together, the projection scheme of Section 6.2 allows proposals that move several points at the same time from one cluster to another. Therefore, the induced set of allowed moves is broader than, for example, that of a scheme that moves one point at a time. Oh *et al.* (2009) consider, for example, "birth" moves proposing to create a cluster from three or more single points in one step. Such moves are likely to be useful to speed up mixing in applications where clusters with many points appear.

One advantage of the mixture proposal in (6.7) is that, after projecting, the posterior $\hat{\pi}^{2D}(\rho^{2D}) \propto \prod_{(i,j) \in \rho^{2D}} w_{ij}^{2D}$ involves only *pairwise interaction* among the points $\mathbf{x}^{2D}$ (see Remark 17). This allows us to re-use the two-color algorithm of Section 6.1 and in particular the approximation given in (P4). Therefore, given

$(\mathbf{x}^{2D}, \rho_{old}^{2D})$, it is possible to perform informed MH moves in the two-color matching space in a computationally efficient way (see Table 6.1 for the performance with two colors).

It would be desirable to design informed proposals like (P3) or (P4) directly in the k-color space, without the need of projecting on two-color subspaces. However it would not be easy to do so in a computationally efficient way. In fact, given the high-dimensionality of the space of matchings contained in a complete k-partite hypergraph, the set of neighboring states ρ_{new} of the current state ρ_{old} would be extremely large. Therefore, it would be very expensive to use a scheme like (P3) in this context. Moreover, since $\hat{\pi}(\rho)$ involves interactions between three or more points, it would not be easy to design an approximation similar to (P4) that could be evaluated efficiently.

Note that the mixture proposal in (6.7) first chooses a lower-dimensional subspace uniformly at random and then performs informed proposals in this space. Therefore, this scheme is a compromise between a "fully uninformed" proposal (which would choose some neighbour of ρ_{old} uniformly at random and thus mix poorly), and a "fully informed" proposal (which would be computationally expensive if it were to make informed proposals in the k-color space).

Since the k-color sample space is more complicated than the two-color one, additional care and longer MCMC runs are needed. We implemented convergence diagnostic techniques similar to those in Section 6.1.3. As might be expected, the number of MCMC steps needed to reach stationarity and to obtain mixing is much higher than in the two-color case (see the end of Chapter 7, for example). Nevertheless, our experiments suggest that, as in the two-color case, the MCMC manages to mix properly unless we are in a case close to complete matching (see Section 6.1.4). In Section 8.3 we briefly mention some possible future research directions concerning upper bounds on the number of projections needed for this MCMC scheme to reach stationarity.

Chapter 7

Analysis of the Anglo-Saxon settlements dataset

In this chapter we present the main results obtained by analyzing the Anglo-Saxon settlements dataset with the Bayesian Random Partition Model (RPM) described in Chapter 3. The computations have been performed using the MCMC algorithm described in Chapter 6. The analysis supports the historians' hypothesis that settlements are clustered according to complementary functional placenames, and allows for statistical inference on the ranges of relevant parameters, thus providing additional insight into the historical phenomenon.

7.1 Main results of the analysis

The no-clustering null hypothesis corresponds to $p_1^{(c)} = 1$ in the RPM of Chapter 3 . As shown in Figure 7.1(a), such a hypothesis clearly lies outside the region where the posterior distribution is concentrated. As a sanity check we also fitted our model to synthetic samples generated according to the no-clustering null hypothesis of Section 2.3.2 (both with and without inhibition among points of the same type). As one would expect, in this case $p_1^{(c)} = 1$ is typically included in the posterior support (see Figure 7.1(a) for an example).

Figure 7.2(a) shows the estimated posterior distribution of σ for the reduced dataset, which is clearly peaked around 4 - 5 km. The 95% Highest Posterior Density interval is $(3.3, 5.9)$ km and the posterior mean is 4.6 km. Therefore, according to the fit given by our model, the clustering behavior consists of clusters with settlements separated by an average distance of 5 km. It is encouraging to note that, although no strong prior information on σ has been exploited, this value is in accordance

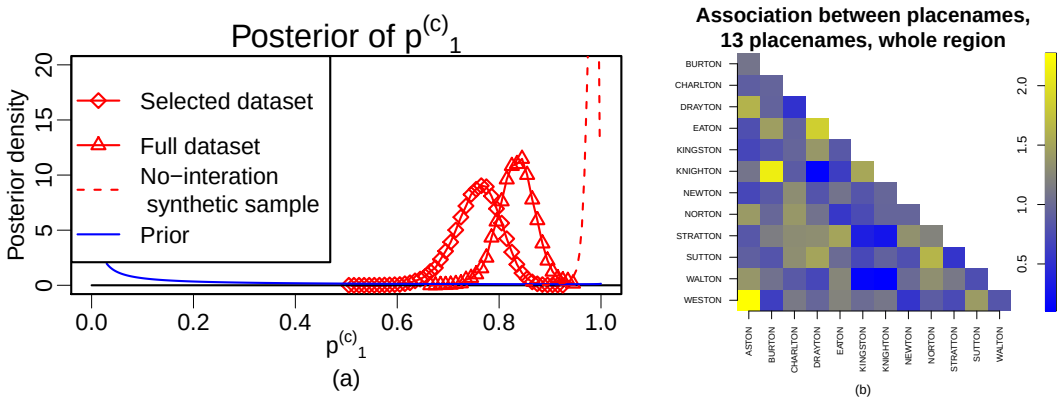


Figure 7.1: (a) Estimated posterior distribution of $p_1^{(c)}$ (see Chapter 3) for the reduced and full dataset (13 and 20 placenames respectively). The hypothesis of no clustering ($p_1^{(c)} = 1$) lies outside the support of the posterior for the real dataset. (b) Measure of association between placenames (see end of Chapter 7).

with the range of plausible values suggested by historians and coherent with the historical interpretation (see Section 3.4).

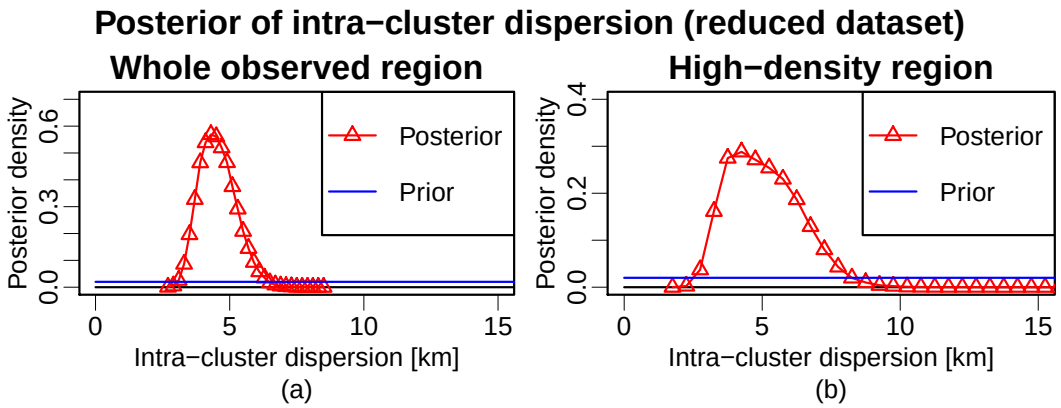


Figure 7.2: (a) $\pi(\sigma|\mathbf{x})$ for the reduced dataset. (b) $\pi(\sigma|\mathbf{x})$ considering only a high-density region (see Chapter 8).

Figure 7.3(a) shows a box plot representation of the posterior distribution of $(Y_1, \ldots, Y_k)$, where Y_l is the number of settlements in clusters of size l (i.e. clusters with l settlements). Note that, on average, more than half of the settlements are not clustered (i.e belong to clusters of size 1). Moreover most of the clustered settlements belong to clusters of size 2. Historians were expecting to see more clusters involving three or four settlements than what was reported by our model. This could be due to a lack of flexibility of our model. In particular, inspection shows that model-fitting, and the requirement to fit clusters in the low-density region

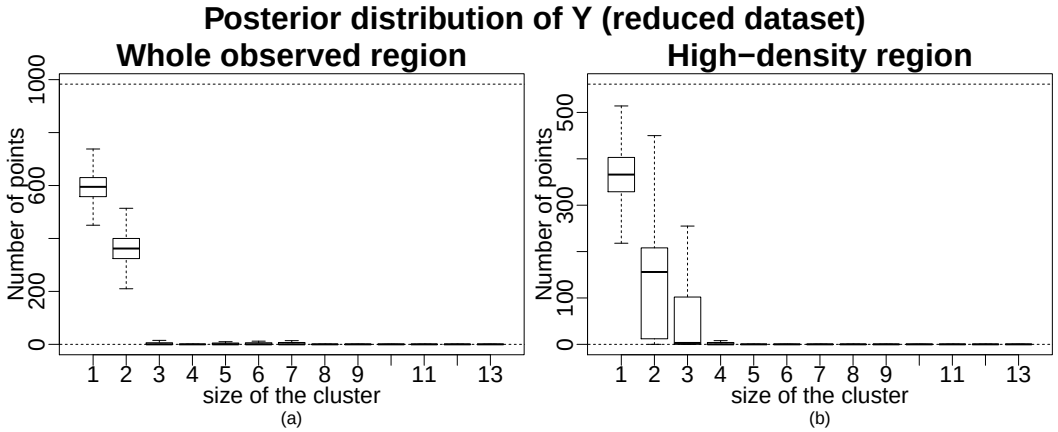


Figure 7.3: (a) Posterior distribution of $\mathbf{Y} = (Y_1, \ldots, Y_k)$ for the reduced dataset. (b) Same as (a) but considering only the settlements in a high density region (see Section 7.1).

(which mostly contain couples with a high posterior probability), pushes clusters in the high-density region to be couples too. In fact when the high-density region is analyzed separately (approximately 600 settlements) more triples appear and the posterior of σ includes also slightly larger values, see Figures 7.2(b) and 7.3(b). This suggests that there might be a heterogeneity in the clustering behaviour between high and low-density regions which is not captured in the model applied to the whole region. This indicates a possible direction for future work (see Chapter 8).

Figure 7.4 shows a graphical representation of the posterior distribution of the partition ρ for the reduced dataset. This representation is of considerable use since it provides a visual representation of how the model is fitting the data and enables comparison with contextual information.

We performed sensitivity analysis on the values of the hyperparameters of σ, λ and $\mathbf{p}^{(c)}$ (see Chapter 3 for details on tested values) and the posterior distribution did not seem to be particularly sensitive to their specification. In Section 7.2 we compare the results obtained with this model with that of the alternative model for the prior distribution of the partition ρ of Section 3.7.1. Such comparison provides further sensitivity analysis to the specification of the prior for ρ.

Figure 7.1(b) graphically depicts a measure of association between placenames. Given two placenames, say a and b, the measure is defined as

$$\frac{\Pr[A|B]}{\Pr[A]} \quad = \quad \frac{\Pr[A \cap B]}{\Pr[A] \cdot \Pr[B]} \quad = \quad \frac{\Pr[B|A]}{\Pr[B]}, \tag{7.1}$$

where A and B are the events of observing placename a and b respectively in a

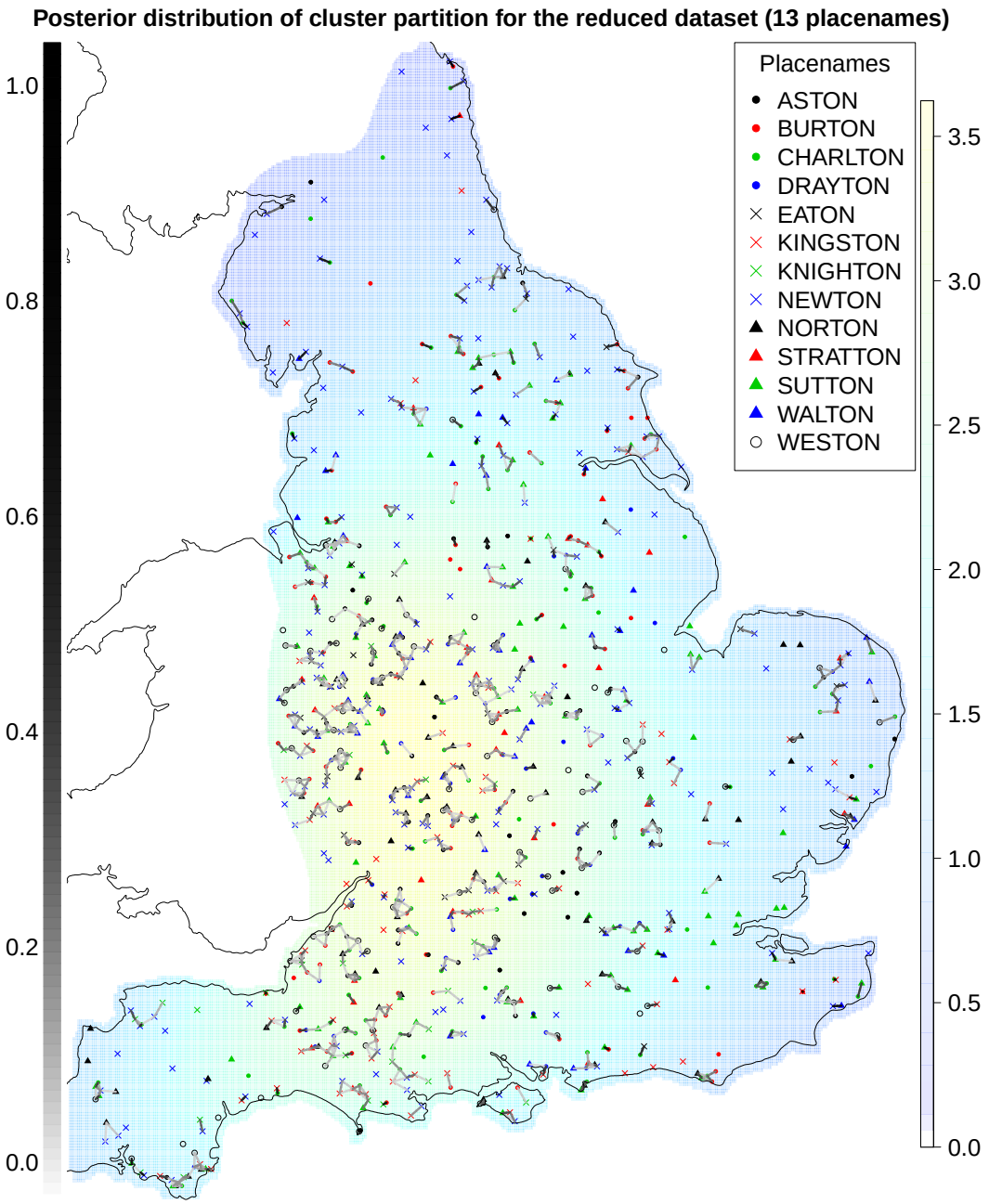


Figure 7.4: Graphical representation of $\pi(\rho|\mathbf{x})$, where $\mathbf{x}$ is the reduced dataset (13 placenames) in the whole observed region. The intensity of gray corresponds to the estimated posterior probability of the cluster. The truncated kernel density estimation of g is plotted in the background, with values express in relative terms with respect to the uniform measure.

cluster chosen uniformly at random from the clusters of $\rho \sim \pi(\rho|\mathbf{x})$. In Figure

7.1(b) we plot MCMC estimates of the values of the measure in equation (7.1) for the placenames in the reduced dataset. Such values are plotted in relative terms with respect to that of the following null hypothesis regime: first choose a cluster from ρ as before; then, denoting the number of settlements in the cluster by s, we sample s placenames independently of each other with placename probabilities proportional to their cardinality in the dataset, conditioning on having pairwise different placenames. The expected values of interest under the null distribution have been estimated using standard Monte Carlo methods. High values in Figure 7.1(b) suggest positive interaction between placenames, while low values suggest negative interaction. Most of the positive associations suggested by Figure 7.1(b), such as $Knighton$-$Burton$, $Weston$-$Aston$ or $Eaton$-$Drayton$, are coherent with the current historians hypotheses. Since such hypotheses have been developed independently of this analysis, the correspondence between the two is an encouraging sign that the association indicated by our analysis is historically meaningful.

We note that, for a fixed ρ, the measure in (7.1) reduces to the *coefficient of association* used by ecologists to measure association between species (Dice, 1945). Many different measures of association have been proposed in the ecological literature (see e.g. Janson & Vegelius, 1981) We chose (7.1) because it is symmetric, clearly interpretable and our experiments suggest that (7.1) is not much influenced by the cardinality of placenames a or b, unlike most measures proposed in Janson & Vegelius (1981).

In order to obtain the results presented in this section, the MCMC algorithm of Section 6.2 was run for 10^6 steps, where at each step 200 moves of the two-color configuration $(\mathbf{x}^{2D}, \rho^{2D})$ were proposed. We assessed convergence using the diagnostic methods described in Section 6.1.3 (e.g. the value of D in (6.6) was approximately 0.02). It took approximately 40 hours to obtain these runs using a basic R implementation (available at `https://sites.google.com/site/gzanellawebpage/compclust_supp_f.zip`) on a desktop computer with Intel $i7$-2600 processor, 3.40GHz CPU and 16GB of RAM was approximately 40 hours.

7.2 Additional results and sensitivity analysis

In Section 3.7 we defined two model variants. The first one employs a different model for the prior distribution of the partition ρ, namely $\pi(\rho)$, based on the Dirichlet-Multinomial distribution rather than the Poisson one. This model can be considered as some form or sensitivity analysis with respect to the choice of $\pi(\rho)$. The second model extension relaxes the assumption that the point marks are sampled uniformly.

In this section we present the results obtained when analyzing the Anglo-Saxon settlements dataset under the model variants just described. In both cases the results obtained are concordant with the ones obtained with the previous model (Section 7.1). This increases the reliability of the results obtained in Section 7.1.

Note that both these model variations incur an additional computational cost, which is one of the reasons we preferred the Poisson model of Chapter 3.

Comparing the Poisson and the Dirichlet-Multinomial models

We used the MCMC algorithm described in Chapter 6 to target the posterior distribution arising from the Multinomial-Dirichlet model (Section 3.7.1) when applied to the Anglo-Saxon settlements dataset. Figure 7.5 compares some results obtained with the Dirichlet-Multinomial model to the ones obtained with the Poisson model (Section 3.5.1), displayed in Section 7.1. The posterior distributions obtained with the two models are very similar.

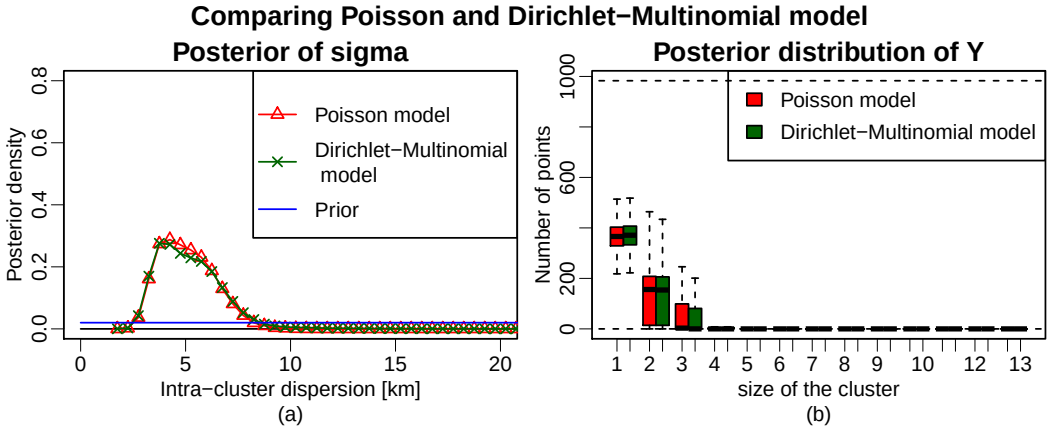


Figure 7.5: Comparison of the Dirichlet-Multinomial model (see Section 3.7.1) and the Poisson one (see Section 3.5.1) on the reduced dataset in the high-density region. (a) Posterior distribution of σ and (b) posterior distribution of $\mathbf{Y} = (Y_1, \ldots, Y_k)$ (see Section 3.7.1 for a definition of $\mathbf{Y}$).

Comparing the uniform and non-uniform marks assumptions

In Section 3.7.2 we defined a model where the marks are sampled according to a non-uniform probability vector $\mathbf{p}^{(m)} = (p_1^{(m)}, \ldots, p_k^{(m)})$. Since $\mathbf{p}^{(m)}$ is unknown, the standard Bayesian approach would be to define a prior distribution on $\mathbf{p}^{(m)}$ and to consider the joint posterior distribution of $\mathbf{p}^{(m)}$ and the other unknown quantities (namely σ, $\mathbf{p}^{(c)}$ and ρ). In order to explore such a posterior distribution, one should add a Metropolis-Hastings step updating $\mathbf{p}^{(m)}$ to the MCMC algorithm of Chapter

6. This step would require the evaluation of the normalizing constants $Z_1(\mathbf{p}^{(m)})$ up to $Z_k(\mathbf{p}^{(m)})$ defined in (3.27) for the proposed value of $\mathbf{p}^{(m)}$. Note that the evaluation of $Z_s(\mathbf{p}^{(m)})$ is costly because its definition involves a summation over all the elements of $\mathcal{M}_s$ (see (3.23) for its definition). By expressing $Z_s(\mathbf{p}^{(m)})$ as the permanent of an appropriate $k \times k$ matrix, we could use Ryser's algorithm (Ryser, 1963), whose complexity is of order $O(2^k k)$. This allows us to evaluate $Z_1(\mathbf{p}^{(m)})$ up to $Z_k(\mathbf{p}^{(m)})$ but the cost is too high to perform such evaluation at each MCMC step (the step updating $\mathbf{p}^{(m)}$ would dominate the others in terms of computational cost, making the algorithm too expensive).

In order to circumvent this problem we replace $\mathbf{p}^{(m)}$ with a plug-in estimator, in an empirical Bayes fashion, requiring the evaluation of $Z_1(\mathbf{p}^{(m)})$ up to $Z_k(\mathbf{p}^{(m)})$ only once. However, in such a setup the posterior distribution will not account for the uncertainty over $\mathbf{p}^{(m)}$. Nevertheless this will allow us to understand what the impact of using a non-uniform $\mathbf{p}^{(m)}$ is over the estimates of the quantities of interest (e.g. σ and $\mathbf{p}^{(c)}$) in a computationally feasible way. A natural estimator for the probability of the i-th mark, $p_i^{(m)}$, is the number of points with this mark divided by the total number of points, $\frac{n_i(\mathbf{x})}{n(\mathbf{x})}$. Although such an estimator is biased for our model (because of the complementarity requirement), it serves the aim of this section.

We performed posterior inference setting $p_i^{(m)} = \frac{n_i(\mathbf{x})}{n(\mathbf{x})}$ for i from 1 to k and replacing the likelihood (3.2) with the non-uniform version in (3.28). The results are

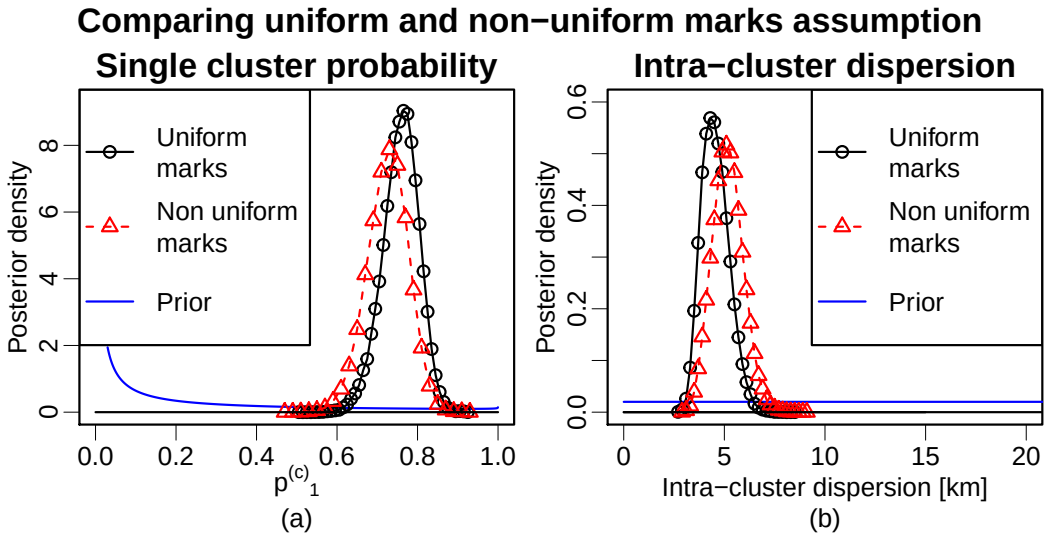


Figure 7.6: Comparison of the posterior distributions of (a) $p_1^{(c)}$ and (b) σ, obtained with and without the assumption of the marks being sampled uniformly (see Section 3.7.2 for details).

in accordance with each other, although there are some differences (see Figure 7.6). In particular the results obtained with the uniform marks assumptions are more conservative, meaning that they produce less clustering. In the main data analysis part (Section 7.1) we used the uniform marks assumption for simplicity and because it produces more conservative results.

We tested the sensitivity of the results to the choice of the plug-in estimator $p_i^{(m)} = \frac{n_i(\mathbf{x})}{n(\mathbf{x})}$. In particular we sampled perturbed values $(\tilde{n}_1, \ldots, \tilde{n}_k)$ according to a multinomial distribution $\text{Mult}(n(\mathbf{x}), \mathbf{p}^{(m)})$, with $p_i^{(m)} = \frac{n_i(\mathbf{x})}{n(\mathbf{x})}$, and used the perturbed values $\tilde{p}_i^{(m)} = \frac{\tilde{n}_i}{n(\mathbf{x})}$ as plug-in estimator. The results with and without the perturbation were extremely similar.

7.3 Summary of results of the analysis

The analysis of the Anglo-Saxon settlements dataset using the Random Partition Model (RPM) of Chapter 3 supports the complementary clustering hypothesis of Prof. Blair (see Chapter 2 for some more historical details and Figure7.2(a) for the posterior distribution of $p_1^{(c)}$). This is in accordance with the K-function preliminary analysis of Chapter 2 (see Figure 2.6).

In addition, the RPM analysis provides valuable additional information at a detailed level. For example, it suggests that settlements in the same cluster have an average distance of 5 km (Figure 7.2a) and that most clusters have 2 or 3 settlements (Figure 7.3). It also provides information regarding the level of interaction among placenames (Figure 7.2a). Such information is important for historians to validate or discredit current hypotheses on the roles and interaction of placenames. Graphical representations of the posterior distribution of the partition ρ, such as Figure 7.4, allow for a visual appreciation of the clustering behavior and for comparison of the results of the analysis with contextual historical information.

Finally Section 7.2 suggests that the results are not too sensitive to changes in modeling assumptions such as relaxing the uniformity of marks assumption. Although modifying such assumptions can have an effect on the resulting posterior distribution, the main features remain unchanged (Figure 7.6). In general we preferred choices leading to more conservative results.

The results in this chapter have been presented to Prof. Blair and collaborators and have been appreciated for their capacity to give information at a global as well as at a detailed level. For example the interaction plot of Figure 7.2(a) has been valuable both for the new information it provides and because, being in accordance with independent historical information, it increased the historians' confidence in

the results of the statistical analysis.

Prof. Blair and collaborators are currently working on the historical interpretation of the results and the comparison with current historical knowledge.

Chapter 8

Discussion and future works

8.1 Summary

We have considered a question posed by Prof. John Blair regarding complementary clustering of Anglo-Saxon settlements (Chapter 2). We designed a Random Partition Model (RPM) that is able to capture the clustering behaviour expected by the historians involved in the project. With no strong prior information, the model produces estimates that are meaningful in the historical context. See, for example, the posterior distribution of σ and the association between placenames in Figure 7.1(b). We also defined a flexible prior distribution for cluster partitions that is designed for a "small clusters" framework (where each cluster has at most k points with k small). In doing so we developed an RPM to perform complementary clustering which is applicable more generally to contexts where one needs to find aggregations of elements of different types. For example Professor Susan Holmes from Stanford University suggests that, in biological contexts, species living in the same geographical area assemble by dissimilarity as they fill different ecological niches, resulting in clusters of complementary species.

We carefully analyzed the computational aspects of this problem. After considering related problems in the complexity theory literature (Chapter 4) we employed Markov chain Monte Carlo (MCMC) methods. We studied the problem of designing informed Metropolis-Hastings (MH) proposal distributions in discrete sample space contexts (Chapter 5). We proposed a choice of MH proposal distributions that, compared to the usual choices found in the literature, achieves a significantly better mixing by approximating detailed balance conditions (see Section 6.1.1). We developed a multiple proposal scheme to allow for parallel computation that could be relevant for applications to bigger datasets (see Section 6.1.5). Re-

garding convergence diagnostics we noted that, when monitoring the convergence of the MCMC in the partition space, univariate summary statistics did not appear to be sufficiently informative. Diagnostics based on multivariate summary statistics or on the matrix of the estimated association probabilities seemed to give more robust results (see Section 6.1.3).

8.2 Modeling aspects

Although the proposed model manages to capture the pattern we were looking for and to provide additional information about it, there is much room for improvement. Here we mention some possible directions for future work.

The RPM model of Chapter 3 could be extended in order to capture the heterogeneity in the clustering behaviour between high and low-density regions (see Section 7.1). One could try to do this by allowing the parameters $\mathbf{p}^{(c)}$ and σ to vary over different regions, perhaps as a function of the points density, while taking care not to over-parametrize the model (the amount of data is limited). An alternative approach would be to modify the metric we use to evaluate distances between settlements. For example one could use a non-Euclidean distances, perhaps based on the inverse square root of the settlement density, in order to allow for larger clusters (meaning with points further apart) in less dense regions. One could also try to model the dispersion of settlements in the same cluster with a non-Gaussian distribution having heavier tails.

Another extension that could result in a better fit is to introduce spatial dependence of placename probabilities. In fact in our model, both under the assumption of uniform and non-uniform marks (see Remark 1 of Chapter 3 and Section 3.7.2), the probability of choosing a certain placename does not depend on the location, while the data suggest that different placenames have different probabilities of being chosen in different regions. As we pointed out in Section 3.7.2, some care should be taken to allow for such spatial dependence without incurring in a consistent increase of the computational burden required by the inference procedure (see also Section 7.2).

The context suggests that we are observing a thinned version of the original settlements distribution. Nevertheless it is not obvious how to incorporate missing data in this model without making further assumptions that do not seem realistic and are not supported by the historical informations available (e.g. that in each cluster there is a settlement for each type).

An interesting direction for future work is to try to incorporate other sources

of data in the model. For example topographical information seems to be related to settlement clustering. In particular historians think that settlements named Burton are related to good vantage points. It would be interesting to find an efficient way to incorporate information obtained from viewshed analysis (now available in most geographic information systems) in the model.

In this work we assumed strict complementarity between placenames in the same cluster, meaning that no two settlements in the same cluster can have the same placename. Although this assumption is well motivated by the historical context (see Sections 2.1 and 3.1), it would be interesting to explore the sensitivity of the results to such a requirement. For example, this could be done by allowing for up to two settlements with the same placename per cluster, or alternatively one could design a more elaborate generative model where the placenames within the same cluster are more likely to be distinct among themselves, but are not strictly mutually exclusive.

Another direction that could be explored is to use a soft clustering approach, allowing settlements to belong to different clusters according to various membership levels. It would be interesting to explore the impact of using such an approach to the computational complexity of the problem (Chapter 4). However, we are not aware of strong historical evidences to support such a modeling approach in this context, and therefore we decided not to employ it in this work.

8.3 Computational aspects

A significant part of the thesis has been focused on computational aspects, in particular on studying MCMC algorithms on matching spaces (Chapter 6) and more generally on discrete spaces (Chapter 5). We mention some possible directions for future work.

Balanced proposals in continuous frameworks

As we showed in Section 5.6, the framework of Chapter 5 (and thus the idea of balanced proposals) can be extended to continuous frameworks. For example, if $\Omega = \mathbb{R}^n$ and $\pi(dx) = f(x)dx$, one can consider a proposal $Q(x, dy) = q(x, y)dy$ with probability density function given by

$$q(x,y) \;\propto\; g\left(\frac{f(y)}{f(x)}\right) \cdot \exp\left(-\frac{|y-x|^2}{2\sigma^2}\right), \tag{8.1}$$

where g is a balancing function satisfying $g(t) = t\, g(1/t)$. Note that the Gaussian kernel $\exp\left(-\frac{|y-x|^2}{2\sigma^2}\right)$ could be replaced, for example, with other symmetric functions of $|y-x|$ decaying at infinity (see Section 5.6 for more details).

It would be interesting to extend the ordering results in Theorem 5 to continuous state spaces and proposals such as in (8.1), using the extension of Peskun ordering for general state spaces (Tierney, 1998). Moreover, in order to make a result like Theorem 5 relevant, one should also extend Theorem 6 to continuous frameworks, paying attention to necessary smoothness assumptions on the density f.

Generalized Langevin MCMC

In Section 5.6 we showed that the Metropolis-Adjusted Langevin Algorithm (MALA) can be seen as a special case of the balanced proposal in (8.1), where the balancing function chosen is $g(t) = \sqrt{t}$ and the target is replaced with a first order Taylor expansion as follows:

$$q(x,y) \;\propto\; \sqrt{\exp\left(\nabla(\log f)(x)\cdot(y-x)\right)}\,\exp\left(-\frac{|y-x|^2}{2\sigma^2}\right). \tag{8.2}$$

Such an observation suggests various ways of extending MALA, which could have both theoretical and practical relevance.

For example by varying the balancing and the smoothing function in (8.2), currently $g(t) = \sqrt{t}$ and Gaussian smoothing respectively, one would obtain a whole family of proposal distributions using first derivative information in a theoretically justified way, like MALA. This flexibility could allow to overcome some shortcomings of MALA, like explosive and unstable behavior for light tails, while retaining its desirable mixing properties. More precisely, we conjecture that appropriate choices of balancing and smoothing functions in (8.2) lead to Markov chains that are geometrically ergodic even for targets with light tails (when MALA is not, see Roberts & Tweedie, 1996) and still scale as $O(d^{1/3})$ as the number of dimensions d goes to infinity, like MALA (see Roberts & Rosenthal, 1998).

Moreover (8.1) suggests a way to incorporate information from second and higher derivatives in the proposal distribution. Many questions arise: do the resulting schemes improve over MALA? Are they related to the previously proposed schemes incorporating second and third derivative, such as MMALA of Girolami & Calderhead (2011) and fMALA of Durmus *et al.* (2015)?

Gradient-free informed proposals

The balanced proposal framework provides a simple and principled way of incorporating local information about the target in the proposal distribution. We could exploit such a framework to design gradient-free informed proposals (both in continuous and discrete settings). In fact derivatives can be expensive to evaluate and may not be the most informative tools to produce local approximations of the target. For example, a promising approach would be to use full conditional distributions to produce local approximations of the target measure. Full conditional distributions contain more information than the simple gradient and can be easy to sample from. Incorporating such information in a proposal which moves all coordinates at once (unlike Gibbs sampling) would result in a "parallel local Gibbs sampling" that could be potentially very efficient and scale well to high-dimensional distributions. Such a scheme would have interesting connections to pseudo-likelihood (or composite-likelihood) methods for the way the posterior is approximated.

Informed proposals for Bayesian variable selection

As mentioned in Section 5.6, it would be interesting to design efficient informed proposals for commonly used discrete models based on the balanced proposal framework of Chapter 5. In particular, in order to achieve full benefits, one should think at efficient ways of using balanced proposals to propose longer moves. In fact, the very high acceptance rate of balanced proposals in Section 6.1 suggest that there is much space to increase the length of the proposed steps and that this could result in a significant improvement in performance.

In doing so, one interesting framework to consider is variable selection (and more generally model selection). Variable selection is a central problem in applied statistics. In this context, the Bayesian approach is particularly attractive, as it allows a principled and robust way to account for model uncertainty and average over it. However, in order to perform inference with this approach one needs to integrate (or sample) over a space made of a continuous part (parameter space) and a discrete part (model space). When the number of variables is large, sampling from the model space is notoriously challenging. Various MCMC schemes have been proposed to accomplish such a task. In particular adaptive MCMC schemes (such as the Individual Adaptation Algorithm of Griffin *et al.* , 2014) have been proposed to incorporate information about the target distribution and have been applied successfully to problems with tens of thousands of variables. However those schemes, mostly motivated by heuristic arguments, still need to be better understood

and therefore improved. A combination of the theoretical results of Chapter 5 (which indicates how to incorporate information in the proposals) with the adaptive MCMC methodology (which allows to collect information from the target efficiently) could provide significant improvements to tackle the computational challenges posed by variable selection and model averaging scenarios.

Trigonometry of Gibbs sampling and the Projection Scheme

The projection scheme of Section 6.2 is related to Gibbs sampling schemes (more precisely to Metropolis-within-Gibbs schemes). It is known that the convergence of a two-stage Gibbs sampler (i.e. a Gibbs sampler on state spaces with two variables) can be characterized in terms of angles between the two functional subspaces, one for each variable (Amit, 1991). In some cases such results can be extended to Metropolis-within-Gibbs algorithms. When $k = 3$ the projection scheme targeting $\hat{\pi}(\rho)$ can be described as a two-stage Gibbs sampler (actually a two-stage Metropolis-within-Gibbs). It would be interesting to explore the geometric ideas of Amit (1991) to see if these can provide upper bounds on the number of projections needed by the projection scheme of Section 6.2 to reach stationarity when $k = 3$.

Bibliography

Amit, Y. 1991. On rates of convergence of stochastic relaxation for Gaussian and non-Gaussian distributions. *Journal of Multivariate Analysis*, **38**(1), 82–99.

Ausiello, G, Crescenzi, P, Gambosi, G, Kann, V, Marchetti-Spaccamela, A, & Protasi, M. 2012. *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. Springer Science & Business Media.

Baddeley, AJ. 2010. Multivariate and marked point processes. *Handbook of spatial statistics*, 371–402.

Baddeley, AJ, & Turner, R. 2005. Spatstat: an R package for analyzing spatial point patterns. *Journal of Statistical Software*, **12**(6), 1–42. ISSN 1548-7660.

Baddeley, AJ, & Van Lieshout, MNM. 1995. Area-interaction point processes. *Annals of the Institute of Statistical Mathematics*, **47**(4), 601–619.

Baddeley, AJ, Moller, J, & Waagepetersen, R. 2000. Non- and semi-parametric estimation of interaction in inhomogeneous point patterns. *Statistica Neerlandica*, **54**(3), 329–350.

Balas, E, & Saltzman, MJ. 1991. An algorithm for the three-index assignment problem. *Operations Research*, **39**(1), 150–161.

Bandelt, HJ, Crama, Y, & Spieksma, FCR. 1994. Approximation algorithms for multi-dimensional assignment problems with decomposable costs. *Discrete Applied Mathematics*, **49**(1), 25–50.

Barrera, J, Lachaud, B, & Ycart, B. 2006. Cut-off for n-tuples of exponentially converging processes. *Stochastic processes and their applications*, **116**(10), 1433–1446.

Becker, RA, Wilks, AR, & Brownrigg, R. 2013. *Mapdata: Extra Map Databases*. R package version 2.2-2.

Berge, C, & Minieka, E. 1973. *Graphs and hypergraphs.* Amsterdam: North-Holland publishing company.

Besag, J. 1974. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 192–236.

Bon, JL, & Păltănea, E. 2001. Convergence of the number of failed components in a Markov system with nonidentical components. *Journal of applied probability*, **38**(4), 882–897.

Brooks, SP, & Gelman, A. 1998. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, **7**(4), 434–455.

Brooks, SP, & Roberts, GO. 1998. Assessing convergence of Markov chain Monte Carlo algorithms. *Statistics and Computing*, **8**(4), 319–335.

Chiu, SN, Stoyan, D, Kendall, WS, & Mecke, J. 2013. *Stochastic geometry and its applications.* John Wiley & Sons.

Cowles, MK, & Carlin, BP. 1996. Markov chain Monte Carlo convergence diagnostics: a comparative review. *Journal of the American Statistical Association*, **91**(434), 883–904.

Crama, Y, & Spieksma, FCR. 1992. Approximation algorithms for three-dimensional assignment problems with triangle inequalities. *European Journal of Operational Research*, **60**, 273–279.

Daley, DJ, & Vere-Jones, D. 2002. *An Introduction to the Theory of Point Processes. Volume I: Elementary Theory and Methods.* 2nd ed. Springer, New York.

Daley, DJ, & Vere-Jones, D. 2008. *An Introduction to the Theory of Point Processes. Volume II: General Theory and Structure.* 2nd ed. Springer, New York.

De la Vega, WF, Karpinski, M, Kenyon, C, & Rabani, Y. 2003. Approximation Schemes for Clustering Problems in Finite Metrics and High Dimensional Spaces. *Proceedings on the thirty-fifth annual ACM symposium on Theory of Computing*, 50–58.

Dellaert, F, Seitz, SM, Thorpe, CE, & Thrun, S. 2003. EM, MCMC, and chain flipping for structure from motion with unknown correspondence. *Machine Learning*, **50**(1-2), 45–71.

Dice, LR. 1945. Measures of the amount of ecologic association between species. *Ecology*, **26**(3), 297–302.

Diggle, PJ. 1985. A kernel method for smoothing point process data. *Applied statistics*, 138–147.

Diggle, PJ. 2003. *Statistical analysis of spatial point patterns*. Edward Arnold.

Diggle, PJ, Eglen, SJ, & Troy, JB. 2006. Modelling the bivariate spatial distribution of amacrine cells. *Pages 215–233 of: Case Studies in Spatial Point Process Modeling*. Springer.

Durmus, A, Roberts, GO, Vilmart, G, & Zygalakis, KC. 2015. Fast Langevin based algorithm for MCMC in high dimensions. *arXiv preprint arXiv:1507.02166*.

Ethier, SN, & Kurtz, TG. 1986. Markov Processes: Characterization and Convergence.

Garey, MR, & Johnson, DS. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co.

Gelling, M, & Cole, A. 2000. *The landscape of place-names*. Shaun Tyas.

Gelman, A. 2006. Prior distributions for variance parameters in hierarchical models. *Bayesian analysis*, **1**(3), 515–534.

Gelman, A, & Rubin, D. 1992. Inference from Iterative Simulation using Multiple Sequences. *Statistical Science*, **4**(4), 457–511.

Geyer, CJ, & Thompson, EA. 1995. Annealing Markov chain Monte Carlo with applications to ancestral inference. *Journal of the American Statistical Association*, **90**(431), 909–920.

Girolami, M, & Calderhead, B. 2011. Riemann manifold Langevin and Hamiltonian monte carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **73**(2), 123–214.

Grabarnik, P, Myllymäki, M, & Stoyan, D. 2011. Correct testing of mark independence for marked point patterns. *Ecological Modelling*, **222**(23), 3888–3894.

Griffin, J, Latuszynski, K, & Steel, M. 2014. Individual adaptation: an adaptive MCMC scheme for variable selection problems. *arXiv preprint arXiv:1412.6760*.

Huber, ML, & Law, J. 2012. Simulation reduction of the Ising model to general matchings. *Electronic Journal of Probability*, **17**(33), 1–15.

Isham, V. 2010. Spatial point process models. *In:* Gelfand, AE, Diggle, PJ, Guttorp, P, & Fuentes, M (eds), *Handbook of spatial statistics*. CRC press.

Janson, S, & Vegelius, J. 1981. Measures of ecological association. *Oecologia*, **49**(3), 371–376.

Jerrum, M. 2003. *Counting, Sampling and Integrating: Algorithms and Complexity*. Lectures in Mathematics ETH Zürich. Basel: Birkhäuser Verlag.

Jerrum, M, & Sinclair, A. 1996. The Markov chain Monte Carlo method: an approach to approximate counting and integration. *Approximation algorithms for NP-hard problems*, 482–520.

Jones, R, & Semple, S. 2012. *Sense of Place in Anglo-Saxon England*. Shaun Tyas.

Karpinski, M, Rucinski, A, & Szymanska, E. 2012. Approximate Counting of Matchings in Sparse Uniform Hypergraphs. *arXiv preprint arXiv:1204.5335*, 1–13.

Kuhn, HW. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, **2**(1-2), 83–97.

Kypraios, T. 2007. *Efficient Bayesian Inference for Partially Observed Stochastic Epidemics and A New Class of Semi - Parametric Time Series Models*. Ph.D. thesis, Lancaster University.

Lau, JW, & Green, PJ. 2007. Bayesian model-based clustering procedures. *Journal of Computational and Graphical Statistics*, **16**(3), 526–558.

Lawson, AB, & Denison, DGT. 2010. *Spatial cluster modelling*. CRC press.

Levin, DA, Peres, Y, & Wilmer, EL. 2009. *Markov chains and mixing times*. American Mathematical Soc.

Loizeaux, MA, & McKeague, IW. 2001. Perfect sampling for posterior landmark distributions with an application to the detection of disease clusters Bayesian cluster models. *Selected Proceedings of the Symposium on Inference for Stochastic Processes*, **37**(1999), 321–332.

Madigan, D, York, J, & Allard, D. 1995. Bayesian graphical models for discrete data. *International Statistical Review/Revue Internationale de Statistique*, 215–232.

Marinari, E, & Parisi, G. 1992. Simulated tempering: a new Monte Carlo scheme. *EPL (Europhysics Letters)*, 1–12.

Mira, A. 2001. Ordering and improving the performance of Monte Carlo Markov chains. *Statistical Science*, 340–350.

Mosco, U. 1994. Composite media and asymptotic Dirichlet forms. *Journal of Functional Analysis*, **123**(2), 368–421.

Müller, P, & Quintana, F. 2010. Random partition models with regression on covariates. *Journal of Statistical Planning and Inference*, **140**(10), 2801–2808.

Neal, RM. 2011. MCMC using Hamiltonian dynamics. *In:* Brooks, S, Gelman, A, Jones, G, & Meng, XL (eds), *Handbook of Markov Chain Monte Carlo*. CRC press.

Oh, S, Russell, S, & Sastry, S. 2009. Markov chain Monte Carlo data association for multi-target tracking. *Automatic Control, IEEE Transactions on*, **54**(3), 481–497.

Peskun, PH. 1973. Optimum monte-carlo sampling using Markov chains. *Biometrika*, **60**(3), 607–612.

Plummer, M, Best, N, Cowles, K, & Vines, K. 2005. Output analysis and diagnostics for MCMC. *R package version 0.10-3, URL http://cran. rproject. org*.

Robert, CP, & Casella, G. 2005. Monte Carlo Statistical Methods.

Roberts, GO. 1998. Optimal Metropolis algorithms for product measures on the vertices of a hypercube. *Stochastics and Stochastic Reports*, **62**, 275–283.

Roberts, GO, & Rosenthal, JS. 1998. Optimal scaling of discrete approximations to Langevin diffusions. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, 255–268.

Roberts, GO, & Rosenthal, JS. 2004. General state space Markov chains and MCMC algorithms. *Probability Surveys*, **1**, 20–71.

Roberts, GO, & Tweedie, RL. 1996. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 341–363.

Roberts, GO, Gelman, A, & Gilks, WR. 1997. Weak convergence and optimal scaling of random walk Metropolis algorithms. *The Annals of Applied Probability*, **7**(1), 110–120.

Ryser, HJ. 1963. *Combinatorial mathematics*. Washington: MAA.

Tierney, L. 1998. A note on Metropolis-Hastings kernels for general state spaces. *Annals of Applied Probability*, 1–9.

Valiant, LG. 1979. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, **8**(3), 410–421.

Villani, C. 2009. *Optimal transport: old and new*. Springer New York.

Welling, M, & Teh, YW. 2011. Bayesian learning via stochastic gradient Langevin dynamics. *Pages 681–688 of: Proceedings of the 28th International Conference on Machine Learning (ICML-11)*.

Zanella, G, Kendall, WS, & Bedard, M. 2015. A Dirichlet form approach to MCMC optimal scaling. *In preparation*.