Introdução À Análise De Dados Espacialmente Referenciados

Elias Teixeira Krainski

Outubro, 2012

Sumário

1 Introdução **1**

2 Dados Públicos **3**
2.1 Dados Públicos e Publicação de Dados 3
2.2 Algumas Fontes de Dados 4
2.3 Obtenção dos Dados 5
2.3.1 Dados do IBGE 5
2.3.2 Dados do Datasus 7
2.4 Importação em R 8
2.4.1 Arquivos CSV 8
2.4.2 Arquivos DBC (TabWin) 8
2.5 Exercícios 9

3 Mapas **13**
3.1 Fonte de Mapas Territoriais no Brasil 13
3.2 Importando um *shapefiles* em R 14
3.3 Georeferenciamento Espacialmente Discreto 14
3.3.1 Exemplo com dados de população 16
3.3.2 Salário médio 17
3.3.3 Dados de mortalidade Infantil 18
3.4 Mapa temático 20
3.5 Exercícios 22

4 Georeferenciamento **25**
4.1 Georeferenciamento de Logradouro 25
4.2 Localização de Endereço num Setor Censitário 27
4.2.1 Mapa de Setores Censitários 27
4.2.2 Endereço Pontual em Setor Censitários 28
4.3 Georeferenciamento de CEP 30
4.4 Visualizando no GogleMaps 31
4.5 Imagens do GoogleMaps 31
4.5.1 Obtendo Imagens do GoogleMaps 31
4.5.2 Georeferenciamento de imagens 32

5 Dependência Espacial em Dados de Áreas **37**
5.1 Matriz de Vizinhança 37
5.2 Índice de Moran 40
5.3 Índice de Moran Bayesiano Empírico 42

6 Introdução aos Modelos Espaciais para Dados de Áreas **45**
6.1 Os modelos SAR e CAR 45
6.2 Um Exemplo de Aplicação 46

7 Introdução aos modelos bayesianos espaciais **51**
7.1 Modelo espacial básico 52
7.1.1 Introdução 52
7.1.2 Distribuição à posteriori 53
7.1.3 Inferência 55
7.2 Regressão com priori CAR 56
7.2.1 Introdução 56
7.2.2 Distribuição à posteriori 57
7.2.3 Inferência 59
7.3 Regressão dinâmica via INLA 60
7.4 Modelo dinâmico espaço temporal via INLA 62
7.4.1 Um modelo para mortalidade infantil 62

Capítulo 1

Introdução

Neste material nós apresentamos alguns recursos disponíveis em ambiente R [22] para a visualização de dados georeferenciados. Vamos abordar os diferentes tipos de dados disponíveis, as principais fontes de dados, as formas de importar dados em R, o georeferenciamento e a visualização. A abordagem será de forma prática, ilustrada com diversos exemplos.

Muitos orgãos públicos e/ou pesquisa disponibilizam dados de forma sistemática. Muitos desses dados possuem algum tipo de referência geográfica. Esses dados podem ser as informações demográficas, índices socio-econômicos, dados de saúde, dados de clima, entre outros. Eles podem representar dados de unidades administrativas políticas (países, estados, municípios, etc.), de uma área definida arbitrariamente, de endereços postais ou de um ponto geográfico.

Um dado com referência espacial (ou geográfica) nada mais é um dado que contém a informação do local a que se refere, ou que foi coletado. Assim como um dado temporal contém a data de quando se refere ou foi coletado. Por exemplo, se tivermos duas colunas numa tabela, uma indicando cada estado brasileiro e outra com a taxa de analfabetismo de cada estado, teremos então essa taxa referenciada por estado. A referência geográfica fica completa ao se obter as coordenadas dos polígnos que representam cada estado. Quando temos por exemplo a ocorrência de um crime, geralmente temos também o endereço (logradouro) onde a ocorrencia se passou. Neste caso precisamos obter a coordenada geográfica desse endereço, isto é, fazer o georeferenciamento do dado. Outro exemplo é quando temos dados de estações metereológicas (temperatura, precipitação, etc.) e temos a coordenada geográfica de cada estação.

Inicialmente nós vamos apresentar algumas fontes de dados, o formato digital dos dados e como importá-los no ambiente R. Muitos dos dados apresentados aqui serão de municípios brasileiros ou setores censitários. Neste caso vamos precisar também de mapas digitais da malha de municípios ou setores censitários. Neste caso vamos considerar a malha digital disponibilizada pelo Instituto Brasileiro de Geografia e Estatística - IBGE.

Alguns dados de saúde estão referenciados por Código de Endereçamento Postal - CEP. Neste caso, nós vamos usar uma base de dados de CEPs, disponível na Internet, para obter o logradouro. Com o logradouro, podemos usar uma API (*Apli-*

cation Program Interface) de georeferenciamento do GoogleMaps. Esta API permite a obtenção das coordenadas geográficas (Latitude e Longitude) a partir de um endereço dado por logradouro.

Nós geralmente denominamos dados à informações. Quando esses dados estão georeferenciados, nós os chamamos de dados espaciais. O processo de georeferenciar e visualizar dados espaciais é também conhecido como geoprocessamento. O geoprocessamento comumente é feito num Sistema de Informações geográficas (SIG).

Um SIG é um *software* especialista em operações espaciais, tais como projeções, cálculo de distâncias, de estruturas de vizinhança, consultas espaciais, etc. Um SIG geralmente implementa métodos estatísticos simples e alguns deles implementam métodos simples de Estatística Espacial. Mas neste último quesito, nenhum deles é completo.

O Instituto Nacional de Pesquisas Espaciais tem um SIG que é gratuito, o SPRING [8], e um que é gratuito e de código aberto (*open source*), o TerraView [27], que é baseado na, também *open source*, TerraLib [9].

O R é uma linguagem e ambiente voltado para análises estatísticas. Existem vários pacotes adicionais em R que tornam possíveis todas as operações disponíveis em um SIG. Assim, podemos fazer georefernciamento, visualização, operações espaciais e, principalmente, análise estatística de dados espaciais usando o R. O georeferenciamento, porém, é feito usando uma API de georeferenciamento do GoogleMaps. Nós apenas fazemos consulta a essa APIa partir do ambiente R. Outra possibilidade é a busca de imagens (de ruas, de satélite, etc.) na API de visualização do GoogleMaps, nos possibilitando sobrepor uma imagem a um mapa e vice-versa.

A análise estatística espacial é uma área bastante ampla, sendo subdividida, segundo [11], em três áreas: análise de padrões pontuais, análise de dados pontualmente referenciados (geoestatística) e análise de dados de áreas. A modelagem de dados considerando a referência espacial, assim como a análise de séries temporais, é importante tanto para estudar a dependência espacial existênte nos dados quanto para produzir resultados mais fidedignos em análises via modelos de regressão.

Este material está organizado em capítulos. No Capítulo 2, vamos mostrar algumas das fontes de dados públicos disponíveis. Os mapas disponibilizados pelo IBGE são abordados no Capítuloٌefch:mapas, bem como algumas funcionalidades básicas para a análise de dados espaciais disponíveis em R. No Capítulo 4, abordamos o georeferenciamento pontual. No Capítulo 6 introduzimos algumas técnicas de detecção de dependência espacial em dados de áreas. No Capítulo **??** introduzimos algumas técnicas de modelagem espacial. Esperamos que este seja um material de ajuda aos iniciantes na Estatística Espacial.

Capítulo 2

Dados Públicos

Não pretendemos listar exaustivamente as fontes de dados públicos. Provavelmente, o número de fontes não listadas neste material excede o de fontes listadas. Dentre as fontes de dados, citamos fontes de dados demográficos e socio-econômicos, de saúde, de educação, de clima, etc.

Inicialmente, vamos classificar os dados em três grupos: os microdados, dados tabulados a partir de microdados e dados observacionais. Microdados são os dados digitalizados de cada questionário de uma pesquisa. Se fazemos uma entrevista com várias pessoas, os microdados são as respostas de cada pessoa. Dados tabulados a partir de microdados são uma agregação dos dados a um nível mais alto. Por exemplo, a média, por município, da idade das pessoas entrevistadas na últimu Censo. Os dados observacionais, por sua vez, são dados coletados de observações de um fenômeno. Por exemplo, a precipitação acumulada por dia num determinado local.

2.1 Dados Públicos e Publicação de Dados

Diversos orgãos públicos brasileiros tem tido uma política de abertura no sentido de disponibilizar dados. Talvez isso tenha ocorrido devido à força do argumento de que dados coletados com recursos públicos também deveriam ser de domínio público, ou seja, dados públicos deveriam ser publicados. Mas, muito dessa abertura é devido ao avanço dos recursos computacionais e tecnológicos. A importância de se ter dados públicos é que qualquer pessoa/orgão pode utilizá-los em pesquisas e para subsídio políticas públicas. Infelizmente há muitos dados que seriam úteis em diversas pesquisas que não são públicos, tais como telefonia, saúde suplementar etc.

Há quem argumente que disponibilizar microdados é anti-ético, pela sempre presente possibilidade de idenficiação da fonte primária dos dados. Há também argumentos no sentido de que o mau uso de dados pode ser perigoso. Por exemplo, imagine que numa pesquisa do IBGE seja possível idenficar algumas residências de um setor censitário, apenas observando as respostas tais como número de pessoas e característica (etária, etc.) de cada uma.

Mas o que pensar quando artigos que discutem saúde pública, num nível que seria difícil, ou até impossível, sem a publicação desses dados? Por exemplo, considere

que está disponível informações de cada recém nascido. Isto é, são disponibilizados dados demográficos, socio-econômicos da mãe do nascido vivo, dados de assistência à saúde durante a gestação, dados do parto e das condições do nascido vivo. Ou seja, qualquer pessoa pode, por exemplo, estudar a dificuldade respiratória ao nascer em função das demais informações.

Exercício Você concorda com a publicação de dados? Porque? Não responda apenas com argumentação filosófica, cite pelo menos um exemplo real.

2.2 Algumas Fontes de Dados

O Instituto Brasileiro de Geografia e Estatística - IBGE realiza várias pesquisas periódicas e algumas não periódicas. Dentre as periódicas, temos a Pesquisa Nacional por Amostragem de Domicílios - PNAD, com periodicidade anual. Esta pesquisa obtem basicamente dados demográficos e sócio-econômicos. Em alguns anos foram incluidas questões de saúde, de segurança, entre outras. A política de disponibilização dos dados de PNADS tem sido a de disponibilizar microdados, ocultando algumas variáveis que permitam a identificação de pessas, e disponibilizar estimativas.

O censo populacional feito pelo IBGE tem periodicidade decendial, apenas em 1990 não foi realizado e foi realizado um censo populacional em 1991. Neste caso, são disponibilizados muitos dados com referência geográfica, em nível estadual, municipal, bairros (para alguns municípios) ou setor sensitário. Ou seja, o nível territorial é bastante detalhado. Além do censo populacional, há outros, tais como o censo agropecuário anual, do qual são disponibilizados dados de produção agropecuária a nível de município.

Os dados do IBGE podem ser obtidos a partir do seu *site* `http://www.ibge.gov.br`. Ainda, no Paraná, o Instituto Paranaense de Desenvolvimento Econômico e Social - IPARDES divulga dados de pesquisas feitas em conjunto com o IBGE entre outras. Por exemplo, no seu *site*, `http://www.ipardes.gov.br` podemos obter os microdados da Pesquisa Mensal de Emprego - PME.

O Ministério da Saúde (MS) também tem disponibilizado sistematicamente uma variedade enorme de informações de saúde com referência geográfica, tanto microdados quanto dados tabulados. São disponibilizados microdados do registro de nascimento e morte, microdados de atendimento ambulatorial, internações, etc. Nos dados de nascimento e morte há referência de município de residência e de ocorrência. Nos dados de atendimento ambulatorial e de internações há também o CEP da residência. O MS disponibiliza seus dados através do site do su departamento de informática *site* `http://www.datasus.gov.br` (DATASUS).

No DATAUS também são disponibilizados dados tabulados, tais como número de nascidos vivos por município em cada ano. Ou seja, para visualizar um mapa de um estado dividido por municípios, colorindo cada município em função da mortalidade infantil, podemos vamos usar os dados do DATASUS tabulados por município.

A maioria dos dados são disponibilizados após algum período. Por exemplo,

os dados de PNADs do IBGE e de nascimento e morte do DATASUS demoram aproximadamente dois anos para serem disponibilizados. Há dados que demoram menos tempo, tais como os dados de internações, que demoram aproximadamente dois meses para serem disponibilizados.

O Ministério da Educação disponibiliza dados de censos escolares, provão, Exame Nacional do Ensino Médio (ENEM), etc. Por exemplo, os microdados do ENEM, isto é, os dados do questionário sócio-economico e as respostas a cada questão das provas podem ser obtidos. Isto, através do *site* do Instituto Nacional de Estudos e Pesquisas Educacionais A nísio Teixeira (INEP) em `http://www.inep.gov.br`.

O Instituto Nacional de Pesquisas Espaciais (INPE) também tem uma política de disponibilizar dados com referência espacial. É possível, por exemplo, obter dados diários de estações metereológicas automáticas. Os dados das estações automáticas, demoram poucas horas para serem disponibilizados. Ainda, a Agência Nacional de Águas - ANA disponibiliza dados de precipitação, com alguns meses de defasagem, em milhares de estações metereológicas.

A *National Aeronautics and Space Administration* - NASA disponibiliza dados obtidos de imagens de satélites e de radares embarcados em satélites. Esses dados são globais e podem ser obtidos para uma região do globo. Podemos por exemplo, obter a estimativa da precipitação acumulada a cada 90 minutos em cada quadrado de aproximadamente 2km de lado. Esses dados demoram poucas horas para serem disponibilizados.

Exercício Você sabe de alguma outra fonte de dados? Qual? Como obter esses dados?

2.3 Obtenção dos Dados

A maioria dos dados são obtidos a partir da página de internet do orgão que disponibiliza os dados. Geralmente há uma página específica para *download* dos dados. Nessa página geralmente há alguma forma de consulta de forma a se escolher a variável, período de tempo, abrangência ou nível geográfico, etc. A maioria deles está num formato simples de serem importados em *softwares* de análise estatística, tais como o formato texto.

2.3.1 Dados do IBGE

Suponha que queremos obter a população residente por município. Esses dados são dados tabulados (há também projeções) e podem ser obtidos a partir do SIDRA (Sistema IBGE de Recuperação Automática). Na página principal do IBGE, clicar em 'Banco de dados', depois em <SIDRA>. Nosso objetivo é fazer visualização espacial dos dados, neste caso vamos obter dados do Censo, cujos dados tabulados estão disponíveis até ao nível de setor censitário. Assim, vamos escolher a seção <Demográfico e Contagem>. Vamos escolher a tabela 1552, na seção 'características

da população e dos municípios', que é uma tabela de "População residente, por situação do domicílio e sexo, segundo a forma de declaração da idade e a idade".

A tabela 1552 é uma matriz multidimensional (2x3x3x3x132x2x31784) com 453.112.704 valores, cujo último nível possui 31784 dimensões, e são as localizações geográficas. Vamos selecionar: em Variável: "População Residente"; em Situação de Domicílio: "Total"; em Sexo: "Total", em Forma de declaração de idade: "Total"; em Idade: "Total"; em Ano: "2000"e "2010"; e em Unidade Territorial: "Município". Vamos montar a tabela de forma que Ano fique na coluna e Município na linha.

Um detalhe importante é marcar, no cabeçalho da dimensão Unidade Territorial a opção <Exibir código>. Isto para que a coluna de código de município seja incluida. Esta coluna é necessária para que possamos, mais adiante, associar esses dados a um mapa e fazer a visualização espacial.

Ainda nesta última dimensão, vamos considerar a opção de "Município 2009 em ordem de UF e código (5565)". Além disto, vamos selecionar apenas os municípios do Paraná. Para isto, vamos clicar em seleção avançada, escolher Grupar, escolher unidade da federação e escolher Paraná.

Após isso, vamos escolher a opção <Gravar> e escolher um nome para o arquivo. O arquivo salvo tem a extensão .csv. Agora, basta ler esse arquivo em R e fazer análises. Nós escolhemos o nome "pop20002010munPR" e o arquivo salvo foi "pop20002010munPR.csv". Este arquivo fica por 60 dias gravado num local fixado no site do SIDRA. O exemplo considerado pode ser importado diretamente (até o dia 12 de dezembro de 2012) para o R com

```
pop <- read.csv2("http://www.sidra.ibge.gov.br/download/
pop20002010munPR.csv", skip=8, nrows=399)
```

Onde, `skip=8` indica que vamos iniciar a leitura a partir da linha 9 do arquivo. Isto porque as primeiras 8 linhas contém a descrição/especificação dos dados. Usamos `nrows=399` porque nas últimas linas contém várias linhas com notas explicativas sobre os dados, que não queremos importar.

O IBGE também disponibiliza tabela prontas de dados tabulados no *link* `ftp://ftp.ibge.gov.br/` Por exemplo, variáveis do censo 2010 tabuladas por setores censitários, no *link* `ftp://ftp.ibge.gov.br/Censos/Censo_Demografico_2010/Resultados_do_Universo/Agregados_por_Setores_Censitarios/` Temos um arquivo para cada UF dos dados.

Para o Censo 2000, por exemplo, segundo o site do IBGE "Os agregados por setores censitários foram gerados a partir dos microdados do universo do Censo Demográfico 2000 e são formados por 21 planilhas de dados para cada Unidade da Federação, contendo mais de 3.200 variáveis.

Setor Censitário é unidade territorial de coleta das operações censitárias, definido pelo IBGE, com limites físicos identificados, em áreas contínuas e respeitando a divisão político-administrativa do Brasil.

O Território Nacional foi dividido em 215811 setores para a realização do Censo Demográfico de 2000."

Uma inovação feita pelo IBGE no CENSO 2010 é o Cadastro Nacional de Endereços Para Fins Estatísticos - CNEFE. Trata-se de uma lista de 78 milhões de en-

dereços urbanos e rurais (http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=2028&id_pagina=1). Esta lista está disponível em um ou mais arquivo(s) para cada município no *link* ftp://ftp.ibge.gov.br/Censos/Censo_Demografico_2010/Cadastro_Nacional_de_Enderecos_Fins_Estatisticos/

2.3.2 Dados do Datasus

A partir da página principal do Datasus, podemos obter facilmente dados tabulados ao nível de município. Também podemos acessar microdados de Nascimentos, de Mortalidade e microdados de alguns serviços prestados pelo Sistema Único de Saúde, tais como atendimento ambulatorial e internações. Os dados tabulados podem ser obtidos no formato de arquivos texto separados por ";". Os microdados são disponibilizados num formato específico do DATASUS.

Por exemplo, para conseguir dados de tabulados de número de nascidos vivos por municípios, podemos clicar em <Informações de Saúde>, depois em <Estatísticas Vitais>, depois selecionar, por exemplo, <Nascidos vivos - 1994 a 2010>. Após isso, ou selecionamos um estado no mapa que aparece ou selecionamos na caixa de seleção logo acima do mapa.

A partir da página aberta, podemos construir uma tabela de dados, selecionando a informação na linha, a informação na coluna e qual a variável, neste caso Nascimentos por residência da mãe ou Nascimentos por local de ocorrência. Como exemplo, vamos colocar Município na linha, Ano de nascimento na Coluna e vamos considerar Nascimentos por residência da mãe. Podemos selecionar mais de um ano, por exemplo, selecionar os anos 2000 e 2010. Vamos selecionar todos os anos disponíveis: 1994-2010.

A seguir, no final da página, escolhemos um modo de exibição. Podemos escolher <Tabelas com bordas>, e será exibida uma tabela com a identificação do município (código e nome) e o número de nascidos vivos em cada um dos anos selecionados. Ao final dessa tabela, temos a opção de "Copia com .CSV". Neste caso será salvo um arquivo do tipo texo separados por ";". Lendo no R:

```
nv <- read.csv2("http://tabnet.datasus.gov.br/csv/A110846200_17_213_49.csv", skip=3, n
```

Agora vamos buscar microdados de nascimento. Procurando o termo 'sinasc' no buscador, localizamos a página do Sistema de Informações de Nascidos Vivos. ftp://ftp.datasus.gov.br/catalogo/sinasc.htm. Nesta página, clique em <Arquivos>, depois em e <Arquivos de 1996 em diante> e depois em <Arquivos de declarações de nascidos vivos>. Neste ponto, é mostrada uma tabela com cada linha sendo uma UF e cada coluna sendo um ano.

No interior dessa tabela estão os links para os arquivos de declarações de nascimento. Os números representam o tamanho do arquivo em Kb. Posicionando o cursor na linha Paraná e na coluna 2010, observamos que o link é ftp://ftp.datasus.gov.br/dissemin/publicos/SINASC/NOV/dnres/DNPR2010.DBC. O diretório ftp ftp://ftp.datasus.gov.br/dissemin/publicos/SINASC/NOV/dnres/ lista todos os arquivos disponíveis. É importante conhecer essa estrutura para facilitar download automático desses dados.

Exercício Entre num site que disponibiliza dados e encontre um dado (ou microdado). Você gostou?

2.4 Importação em R

Vamos considerar os formatos de texto separados por vírgulas (ou ponto e vírgula) e o formato DBC.

2.4.1 Arquivos CSV

O IBGE, por exemplo, disponibiliza dados em formato de texto separado por vírgulas - CSV)*comma separated ...*). Vale aqui uma nota sobre o formato CSV. O separador decimal inglês é "." (ponto) enquanto que o nosso é "," (vírgula). Por esse motivo há dois formatos CSV, um onde os campos são separados por "," (vírgula), quando o separador decimal é ponto; e outro onde os campos são separados por ";" (ponto-e-vírgula), quando o separador decimal é "," (vírgula).

Temos em R algumas funções que trabalham com arquivos do tipo texto delimitados: `read.table()`, `read.csv()`, `read.csv2()`, `read.delim()` e `read.delim2()`. Observando o help dessas funções, notamos vários argumentos. Estes argumentos dependem da estrutura do arquivo texto que desejamos importar.

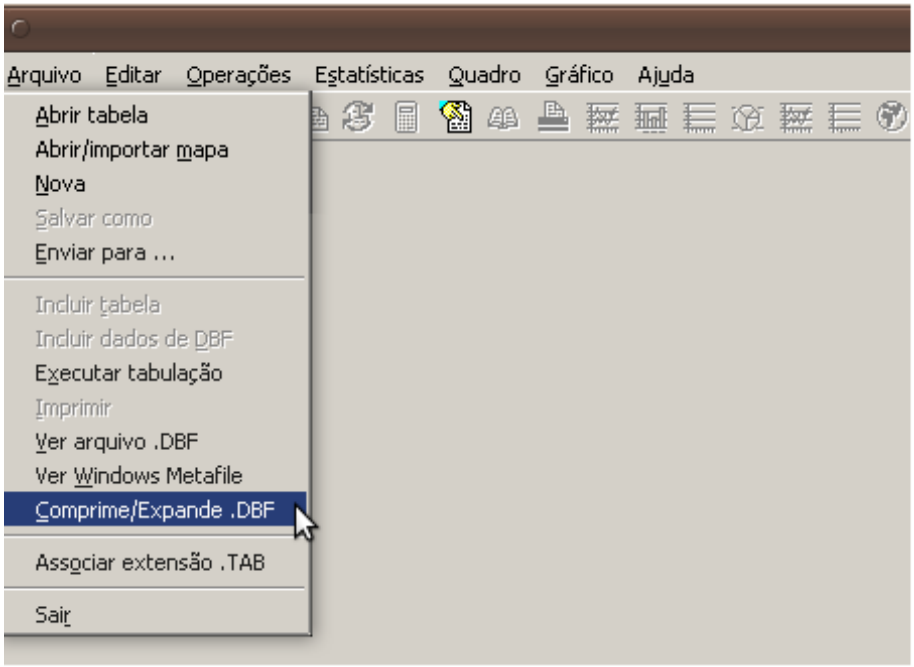
O exemplo dos dados de população do IBGE, considerado na seção anterior, pode ser importado usando a função `read.csv2()`, pois trata-se de um texto com ";" como sendo o separador de campos e "," como sendo o separador decimal. Lembramos que esse arquivo pode ser lido diretamente (até o dia 12 de dezembro de 2012) para o R com

```
pop <- read.csv2("http://www.sidra.ibge.gov.br/
download/pop20002010munPR.csv", skip=8, nrows=399)
```

2.4.2 Arquivos DBC (TabWin)

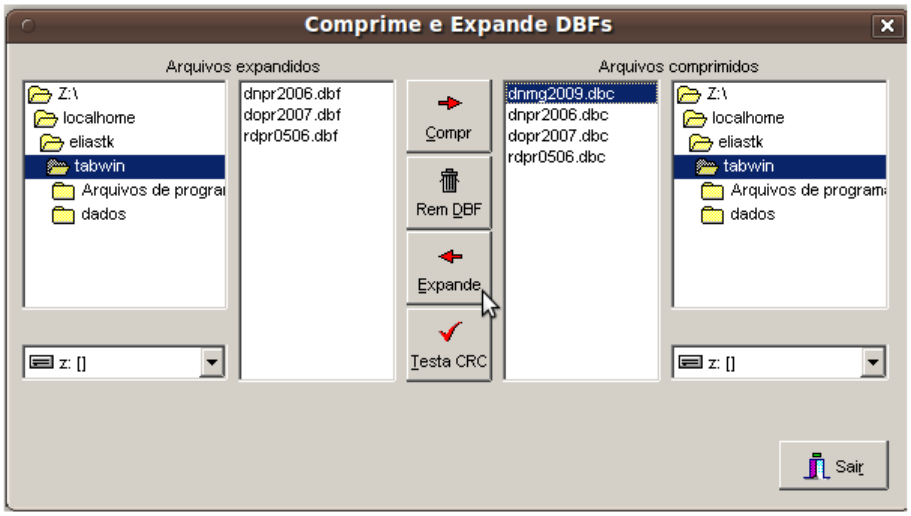
O DATASUS disponibiliza microdados em formato específico de arquivo, o formato DBC. Um arquivo DBC é um arquivo DBF comprimido. O DATASUS fornece um *software*, o TabWin, [12], que permite a expansão de arquivos DBC para o formato DBF e algumas análises exploratórias, incluindo tabulação e visualização espacial. As tabulações feitas pelo TabWin são implementadas em R. Portanto, para trabalhar com microdados do DATASUS é necessário usar o TabWin para expandir os arquivos DBC.

Após iniciar o TabWin, vamos no menu <Arquivo> e selecionamos a opção <Comprime/Expande .DBF>, como mostrado na figura a seguir



Na tela aberta, é necessário localizar o diretório onde está(ão), ou onde será(ão) salvo(s), o(os) arquivo(s) em formato .dbc. Também é necessário localizar o diretório onde estão, ou onde será(ão) salvo(s), o(s) arquivos .dbf.

Localizando os diretórios, basta selecionar o(s) arquivo(s) .dbc que serão expandidos para o formato .dbf e clicar em <Expande>, comforme a figura a seguir



A importação em R de arquivos .dbf pode ser feita usando a função `read.dbf()` do pacote **foreign**, [21].

2.5 Exercícios

1. Entre no site do IBGE e clique em "SIDRA". Na nova tela, clique em "Acervo". Selecione "Variáveis". Selecione "Salário". Selecione "Salários (Cadastro Central de Empresas)". Selecione "Tabela 1735 - Dados gerais (...) e Municípios".

Obtenha o ”Pessoal ocupado total”, ”Pessoal ocupado assalariado”e ”Salários - decimais:3/0”para os anos de 1996 a 2006 para o município de Maringá.

(a) Importe esses dados em R.

(b) Faça um resumo descritivo e gráfico desses dados.

(c) Calcule o salário por pessoa assalariada. Faça um resumo descritivo e gráfico desse dado.

2. Entre no site do IBGE, clique em ”SIDRA”. Na nova tela, clique em ”Acervo”. Selecione ”Níveis Territoriais”. Selecione ”Bairro”e em seguida ”População”. Selecione ”Valor do rendimento nominal médio mensal das pessoas de 10 anos ou mais de idade, com rendimento’ Selecione ’Tabela 3170 - Pessoas de 10 anos ou mais de idade, com rendimento, Valor do rendimento nominal médio mensal e Valor do rendimento nominal mediano mensal das pessoas de 10 anos ou mais de idade, com rendimento, por sexo, situação do domicílio e grupos de idade’. Obtenha o Valor do rendimento nominal médio mensal das pessoas de 10 anos ou mais de idade, com rendimento (Reais) para os bairros dos Municípios de Londrina, Maringá e Curitiba.

(a) Importe esses dados em R.

(b) Faça uma análise descritiva desses dados

(c) Teste se o rendiménto médio mensal das pessoas de 10 anos ou mais de idade (com rendimento) difere nos municípios

3. Entre no site do DATASUS. Clique em ”Informações de Saúde”. Na lista de opções aberta, clique em ”Estatísticas Vitais”. Selecione ”Nascidos vivos - 1994 a 2010”. No mapa mostrado ao lado, selecione o estado do Paraná. Obtenha o ”número de nascidos vivos por município de residência da mãe”por ”Duração da Gestação”no ano de 2010 nos municípios de Curitiba e Maringá.

(a) Importe essa tabela em R.

(b) Existe associação entre ”Duração da Gestação”e município? (Desconsidere os ignorados se houver.)

4. Entre no site do DATASUS. Clique em ”Informações de Saúde”. Na lista de opções aberta, clique em ”Estatísticas Vitais”. Selecione ”Nascidos vivos - 1994 a 2010”. No mapa mostrado ao lado, selecione o estado do Paraná. Obtenha o número de nascidos vivos por local de residência da mãe por ”Peso ao Nascer”no ano de 2010 em todos os municípios de Paraná.

(a) Importe esses dados em R.

(b) Faça uma análise descritiva

(c) Identifique a categoria de ”Peso ao Nascer”que possui o maior número de nascidos vivos no estado

(d) Calcule a proporção dos nascidos vivos nessa categoria para cada município e faça uma análise descritiva (Desconsidere os ignorados se houver).

5. Entre no site de sistema de infomações de mortalidade `http://www.datasus.gov.br/catalogo/sim.htm`.

   NOTA : Considere o arquivo de documentação disponível nessa página.

   (a) Faça o download e um arquivo de declarações de mortalidade de PR em 2010

   (b) Expanda o arquivo para .dbf usando o TabWin

   (c) Importe esses dados para o R

   (d) Quantos homens e quantas mulheres foram a óbito?

   (e) Calcule a idade média para homens e mulheres separadamente

Capítulo 3

Mapas

Um mapa de áreas (municípios, estados) é, basicamente, um conjunto de polígonos. Porém, geralmente temos atributos (variáveis) associados aos polígonos, tais como PIB, População Residente, IDH, etc. Um formato padrão de mapa em formato digital é o *shapefile*, que é um conjunto de pelo menos três arquivos: um arquivo com os polígonos (com extensão .shp), um arquivo com os atributos (com extensão .dbf), e um arquivo com índices (com extensão .shx). Outro arquivo que pode estar incluso num *shapefile* é o arquivo indicando o sistema de projeção das coordenadas dos polígonos, com extensão .prj.

Há outros formatos usados para mapas, que também podem importados em **R**. Um mais recente é o formato kml, usado pelo *google earth*.

3.1 Fonte de Mapas Territoriais no Brasil

O IBGE disponibiliza um conjunto de milhares de mapas. Há mapas do Brasil dividido por municípios e mapas de cada município dividido por setores censitários. Os polígonos de cada área (município/setor censitário), são representados por um conjunto de pontos. Quanto mais pontos utilizados na representação de um polígono, mais apropriado será o mapa para a produção de mapas em alta resolução gráfica. O IBGE disponibiliza mapas para três diferentes resoluções.

Um polígono pode estar representados em diferentes projeções ou cartográficas, [19]. O IBGE disponibiliza mapas em duas projeções: Projeção Geográfica e Projeção Policonica. Além disso, os mapas são disponibilizados em dois formatos de arquivo diferentes: *shapefiles* e *Mge_Dgn*. O mapa de municípios está disponível em arquivo para o Brasil todo, por Região e por Unidade da Federação. Além disso, para cada município está disponível um *shapefiles* da divisão do município em setores censitários.

Vamos, por exemplo, obter o mapa do Paraná subdividido em municípios. Na página principal do IBGE `http://www.ibge.gov.br`, clicamos em <Geociências>, depois em <Mapeamento das unidades territoriais>, depois em <Produtos> e depois em <Malha municipal digital 2007>. Na nova janela, escolhemos escala2500mil (há opções E500, E100 e E2500 em versões anteriores), que é a menor resolução, pois não estamos interessados num produto de maior resolução. Escolhemos a projeção

geográfica <Proj_Geográfica_sad69>, escolhemos o nível territorial Unidade da Federação <UF> e escolhemos o estado do Paraná <PR>.

Assim, chegamos no seguinte diretório `ftp://geoftp.ibge.gov.br/malhas_digitais/municipio_2007/escala_2500mil/proj_geografica_sad69/uf/pr/` e fazemos o *download* do arquivo '41mu2500gsd.zip' que contém os arquivos que compoem o *shapefiles*.

3.2 Importando um *shapefiles* em R

Há mais de um pacote para leitura de mapas em **R** [23]. Para ler *shapefiles* podemos usar o pacote **rgdal** [15]. Este pacote usa as classes de dados espaciais definidas no pacote **sp** [20].

Vamos considerar o mapa obtido na seção anterior. Inicialmente carregamos o pacote **rgdal**.

```
> require(rgdal)
```

Vamos considerar que o conjunto de arquivos do *shapefiles* está do diretório "mapas" do diretório corrente. Vamos usar a função `readOGR()` para ler o mapa.

```
> pr <- readOGR("../mapas", "41mu2500gsd", input_field_name_encoding='latin1')
```

```
OGR data source with driver: ESRI Shapefile
Source: "../mapas", layer: "41mu2500gsd"
with 399 features and 9 fields
Feature type: wkbPolygon with 2 dimensions
```

O objeto `pr` é contém o conjunto de polígonos dos 399 municípios do Paraná e uma tabela com alguns atributos desses municípios, isto é, um `data.frame` com 399 linhas. Podemos visualizar esse mapa simplesmente fazendo

onde `par(mar=c(0,0,0,0))` foi usado para tirar as margens da janela gráfica, aumentando sua área útil.

Na tabela de atributos do mapa temos uma coluna denominada GEOCODIG_M, que é o código de cada município do mapa.

```
> names(pr)
```

```
[1] "GEOCODIG_M" "UF"         "Sigla"
[4] "Nome_Munic" "Região"     "Mesorregiã"
[7] "Nome_Meso"  "Microrregi" "Nome_Micro"
```

3.3 Georeferenciamento Espacialmente Discreto

Algumas vezes temos uma tabela com informações/atributos de áreas geográficas (municípios por exemplo) e gostaríamos de plotar essas informações num mapa. O georeferenciamento espacialmente discreto nada mais é que associar essa tabela de atributos ao mapa.

```
> par(mar=c(0,0,0,0))
> plot(pr)
```

Figura 3.1: Mapa do Paraná dividido em municípios

3.3.1 Exemplo com dados de população

Vamos considerar os dados de população do capítulo anterior e o mapa da seção anterior.

```
> pop <- read.csv2("../dados/pop20002010munPR.csv", skip=8, nrow=399)
```

O código de município no mapa é formado por sete dígigos

```
> head(pr@data, 2)
```

```
  GEOCODIG_M UF Sigla      Nome_Munic Região
0    4100103 41    PR       Abati\xe1    Sul
1    4100202 41    PR Adrian\xf3polis    Sul
  Mesorregiã                  Nome_Meso Microrregi
0       4104 Norte Pioneiro Paranaense      41015
1       4110 Metropolitana de Curitiba      41035
               Nome_Micro
0 Corn\xe9lio Proc\xf3pio
1              Cerro Azul
```

Na tabela dos dados de população temos o código de município como parte da primeira coluna, isto é, os sete primeiros dígitos dessa coluna,

```
> head(pop,2)
```

```
                    X1...Brasil X169799170
1       4100103 - Abati\xe1 - PR       8259
2 4100202 - Adrian\xf3polis - PR       7007
  X190755799
1       7764
2       6376
```

```
> names(pop) <- c("codMun", "pop2000", "pop2010")
```

O que precisamos é, simplesmente, colocar os dados na mesma ordem do mapa. Isso será feito considerando o código do município. Inicialmente, vamos criar uma variável que é o código de município obtido da tabela dos dados.

```
> pop$GEOCODIG_M <- as.integer(substr(as.character(pop[,1]), 1, 7))
```

Vamos criar uma nova tabela contendo o código, o nome, a mesoregião, a micro região, a população em 2000 e a população em 2010.

```
> opop <- merge(pr@data, pop, sort=FALSE)
> head(opop, 3)
```

```
  GEOCODIG_M UF Sigla      Nome_Munic Região
1    4100103 41    PR       Abati\xe1    Sul
2    4100202 41    PR Adrian\xf3polis    Sul
3    4100301 41    PR   Agudos do Sul    Sul
  Mesorregiã                 Nome_Meso Microrregi
1       4104 Norte Pioneiro Paranaense      41015
2       4110 Metropolitana de Curitiba      41035
3       4110 Metropolitana de Curitiba      41039
               Nome_Micro
1 Corn\xe9lio Proc\xf3pio
2              Cerro Azul
3               Rio Negro
                          codMun pop2000 pop2010
1       4100103 - Abati\xe1 - PR    8259    7764
2 4100202 - Adrian\xf3polis - PR    7007    6376
3   4100301 - Agudos do Sul - PR    7221    8270
```

e vamos conferir se esta tabela esta na mesma ordem do mapa.

```
> all.equal(pr@data$GEOCODIG_M, opop$GEOCODIG_M)
```

```
[1] TRUE
```

e atribuir esta tabela ao `SpatialPolygonsDataFrame`.

```
> pr@data <- opop
```

3.3.2 Salário médio

Os dados de salário médio foram obtidos do SIDRA. Foram obtidos dados de salário médio mensal por município para os anos de 2006 a 2010.

```
> smpr <- read.csv2("../dados/salarioMedioMunPR20062010.csv", skip=3, nrow=399)
> dim(smpr)
```

```
[1] 399   6
```

```
> head(smpr,3)
```

```
                              X X2006 X2007
1       4100103 - Abati\xe1 - PR   2.1   2.0
2 4100202 - Adrian\xf3polis - PR   2.2   2.2
3   4100301 - Agudos do Sul - PR   1.7   1.9
  X2008 X2009 X2010
1   2.0   2.0   2.0
2   2.7   2.2   2.4
3   1.8   1.8   1.7
```

```
> tail(smpr,3)
```

```
                          X X2006 X2007 X2008
397   4128658 - Virmond - PR   1.7   1.9   2.0
398  4128708 - Vitorino - PR   2.4   2.4   2.6
399 4128807 - Xambr\xea - PR   1.6   1.6   1.5
    X2009 X2010
397   1.8   1.7
398   2.3   2.4
399   1.5   1.5
```

renomeando as colunas para melhor identificá-las

```
> names(smpr) <- c("codMun", paste("s", 2006:2010, sep=""))
```

Vamos juntar esses dados ao mapa do Paraná dividido em municípios.

```
> codsm <- substr(as.character(smpr[,1]), 1, 7)
> pr@data <- merge(pr@data, data.frame('GEOCODIG_M'=codsm,
+     smpr[,-1]), sort=FALSE)
```

3.3.3 Dados de mortalidade Infantil

Os dados de saúde foram obtidos do site do DATASUS . Temos o número de nascidos vivos por município

```
> nv <- read.csv2("../dados/nascVivosMunPR19942010.csv",
+                 skip=3, nrow=399, na.string='-', fileEncoding='latin1')
> names(nv)
```

```
 [1] "Município" "X1994"     "X1995"
 [4] "X1996"     "X1997"     "X1998"
 [7] "X1999"     "X2000"     "X2001"
[10] "X2002"     "X2003"     "X2004"
[13] "X2005"     "X2006"     "X2007"
[16] "X2008"     "X2009"     "X2010"
```

```
> nv[1:3, 1:5]
```

```
            Município X1994 X1995 X1996 X1997
1        410010 Abatiá   189   197   176   166
2  410020 Adrianópolis   179   192   142   106
3 410030 Agudos do Sul   103    66   132   158
```

e o número de óbitos de crianças com menos de um ano por município

```
> obt <- read.csv2("../dados/obInfMunPR19962010.csv",
+                  skip=3, nrow=399, na.string='-', fileEncoding='latin1')
> names(obt)
```

```
 [1] "Município" "X1996"     "X1997"
 [4] "X1998"     "X1999"     "X2000"
 [7] "X2001"     "X2002"     "X2003"
[10] "X2004"     "X2005"     "X2006"
[13] "X2007"     "X2008"     "X2009"
[16] "X2010"
```

```
> obt[1:3, 1:5]
```

```
             Município X1996 X1997 X1998 X1999
1        410010 Abatiá     4    NA     5     2
2  410020 Adrianópolis     6     3     4     2
3 410030 Agudos do Sul     5     6     8    NA
```

Vamos renomear as colunas dos dados de nascidos vivos e óbitos infantis para que ao juntar os dados tenhamos facilidade de identificá-las

```
> names(nv) <- c(names(nv)[1], paste('n', 1994:2010, sep=''))
> names(obt) <- c(names(obt)[1], paste('o', 1996:2010, sep=''))
```

juntando ambos os dados num único `data.frame`.

```
> nv.obi <- merge(nv, obt)
> nv.obi[1,]
```

```
      Município n1994 n1995 n1996 n1997 n1998
1 410010 Abatiá   189   197   176   166   162
  n1999 n2000 n2001 n2002 n2003 n2004 n2005 n2006
1   119   115   122   114    94   107   126    81
  n2007 n2008 n2009 n2010 o1996 o1997 o1998 o1999
1    99    90    83    80     4    NA     5     2
  o2000 o2001 o2002 o2003 o2004 o2005 o2006 o2007
1     5     2     6     2     2     1     2    NA
  o2008 o2009 o2010
1     1     1    NA
```

Precisamos colocar esses dados na mesma ordem do mapa. Para isso criamos uma coluna de código de município. No caso dos dados do DATASUS, o código de município tem seis dígitos, em vez de sete (IBGE). Porém, usando os seis primeiros é suficiente. Assim, vamos criar uma coluna na tabela de atributos do mapa de código com seis digitos

```
> pr@data$cod6 <- substr(as.character(pr$GEOCODIG_M), 1, 6)
```

Para facilitar, fazemos isso atribuindo o mesmo nome da respectiva coluna no mapa.

```
> nv.obi$cod6 <- as.integer(substr(as.character(nv.obi[,1]), 1, 6))
```

Juntando todos os dados no mapa

```
> pr@data <- merge(pr@data, nv.obi, sort=FALSE)
> pr@data[1,]
```

```
    cod6 GEOCODIG_M UF Sigla Nome_Munic Região
1 410010    4100103 41    PR  Abati\xe1    Sul
  Mesorregiã                 Nome_Meso Microrregi
1       4104 Norte Pioneiro Paranaense      41015
               Nome_Micro
1 Corn\xe9lio Proc\xf3pio
                    codMun pop2000 pop2010 s2006
1 4100103 - Abati\xe1 - PR    8259    7764   2.1
  s2007 s2008 s2009 s2010     Município n1994
1     2     2     2     2 410010 Abatiá   189
  n1995 n1996 n1997 n1998 n1999 n2000 n2001 n2002
1   197   176   166   162   119   115   122   114
  n2003 n2004 n2005 n2006 n2007 n2008 n2009 n2010
1    94   107   126    81    99    90    83    80
  o1996 o1997 o1998 o1999 o2000 o2001 o2002 o2003
1     4    NA     5     2     5     2     6     2
  o2004 o2005 o2006 o2007 o2008 o2009 o2010
1     2     1     2    NA     1     1    NA
```

Salvando em arquivo para uso posterior

```
> writeOGR(pr, "../mapas/", "prdata", "ESRI Shapefile")
```

Podemos visualizar os dados no mapa, isto é, criar um mapa temático.

3.4 Mapa temático

Um mapa temático é um mapa de uma região geográfica colorida de acordo com o valor de um atributo. Por exemplo, podemos colorir o mapa do Paraná dividido por município em tons de cinza, onde cinza mais escuro indica maior mortalidade infantil. Nesta seção vamos aprender a criar um mapa temárico.

Vamos considerar os dados de população em 2000 e 2010 da seção anterior. Vamos considerar que queremos colorir o mapa de forma que os municípios que tiveram a sua população aumentada estejam de uma cor e os demais de outra cor.

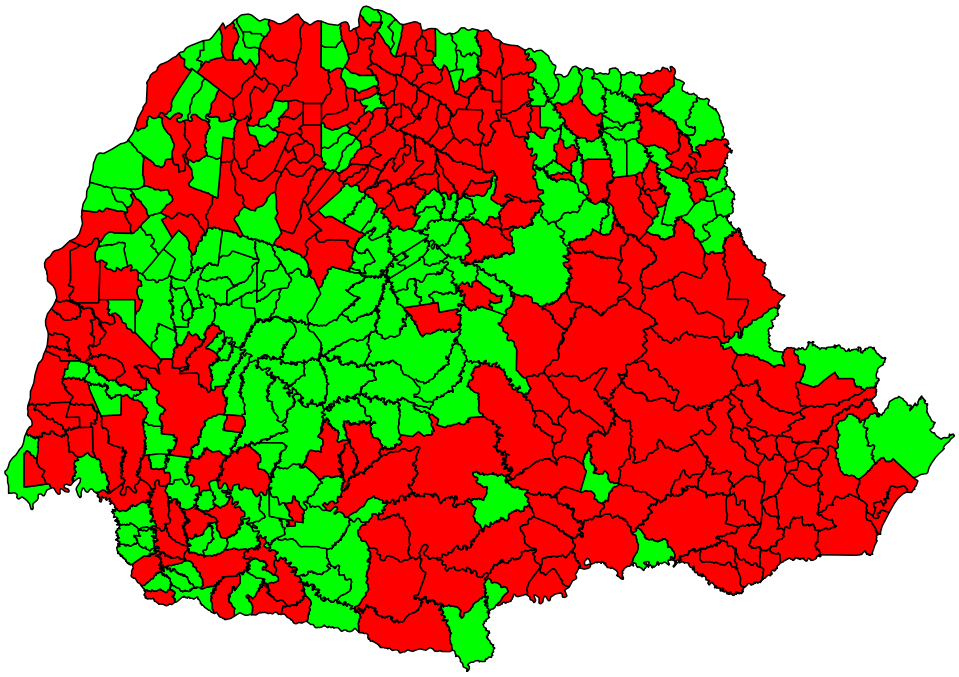
Inicialmente, vamos criar uma variável que indica se a população de cada município aumentou ou não

```
> table(aum <- opop$pop2010>opop$pop2000)
```

```
FALSE  TRUE
  178   221
```

Podemos visualizar o mapa colorindo os municipios que tiveram aumento em vermelho e os demais em verde, assim

```
> par(mar=c(0,0,0,0))
> plot(pr, col=ifelse(aum, 'red', 'green'))
```



Podemos criar um mapa com mais cores, por exemplo, tons diferentes de acordo com o percentual de aumento da população. Vamos criar inicialmente o vetor que indica o aumento percentual da população

```
> summary(paum <- opop$pop2010/opop$pop2000-1)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.
-0.38480 -0.05991  0.01574  0.02251  0.08695
    Max.
 0.73250
```

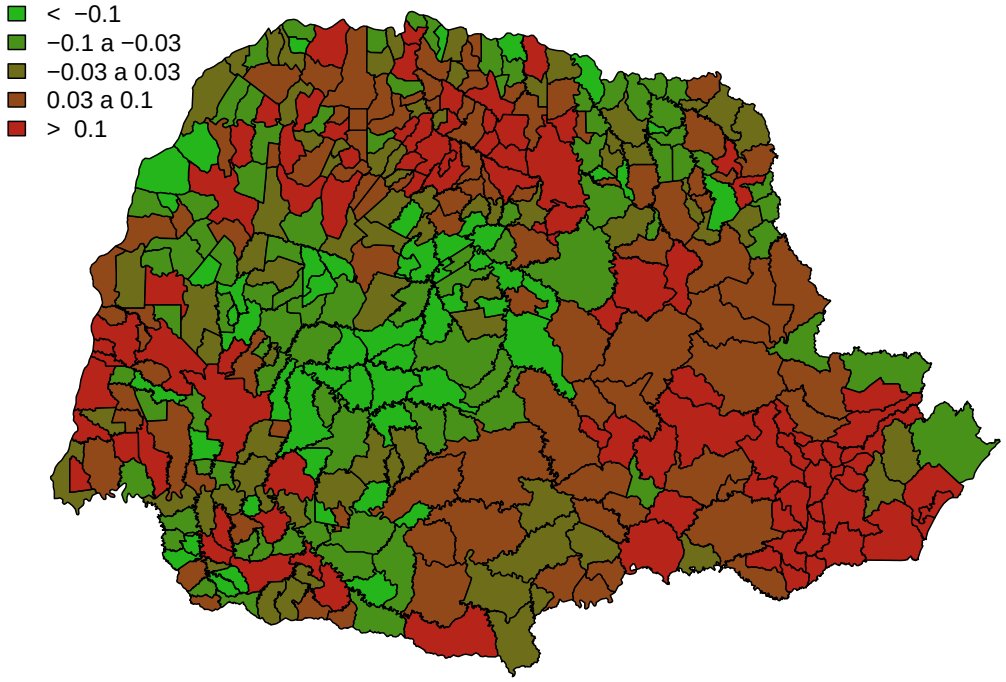
Vamos categorizar essa variável em cinco níveis: menor que -0.02; de -0.02; de 0.02 a 0.05; de 0.05 a 0.15; e maiores que 0.15. Assim,

```
> br <- c(-Inf, -0.1, -0.03, 0.03, 0.1, Inf)
> table(cl5 <- findInterval(paum, br))
```

```
 1  2  3  4  5
54 85 74 95 91
```

Agora vamos definir as cores. Vamos usar tons RGB (red, green e blue) de forma que as primeiras cores tenham um tom mais avermelhado e as ultimas, um tom mais esverdeado.

```
> labelfun <- function(brea) {
+   k <- length(brea)
+   paste(c("<", brea[2:(k-2)], ">"),
+         c("", rep("a", k-3), ""), brea[c(2:(k-1),k-1)])
+ }
> par(mar=c(0,0,0,0))
> plot(pr, col=cores5[cl5])
> legend('topleft', labelfun(br), fill=cores5, bty='n')
```




```
> cores5 <- rgb(1:5/7, 5:1/7, 0.1)
```

O mapa com mais de duas cores pode ser feito com o comando abaixo e adicionamos uma legenda para as cores, resultando num mapa temático.

Vamos considerar os dados de salário médio mensal por município. Podemos fazer um mapa temático usando umaa funçã propria para isso

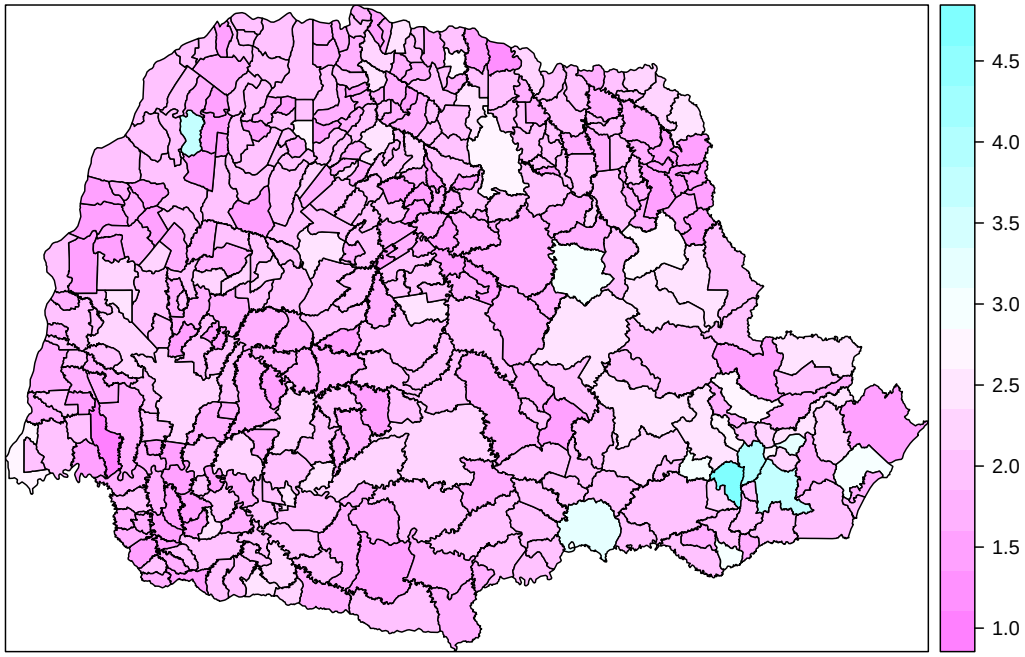
Podemos salvar em arquivo o mapa com as duas colunas de população. Para isso, vamos substituir o elemento `data` do mapa pelo `data.frame` com as duas colunas de população

```
> writeOGR(pr, "../mapas/", "prdata", "ESRI Shapefile")
```

3.5 Exercícios

1. Entre no site do IBGE. Clique em Geociências. Clique em Mapeamento das unidades territoriais. Clique em Produtos. Clique em Malha municipal digital

```
> spplot(pr, 's2010')
```



2007 (clicar no disquete de 2007). Na nova janela escolha E500, E100 ou E2500. Escolha uma projeção. (Proj_Geografica por exemplo). Escolha um formato. (Recomendo ArcView_shp) Escolha o nível territorial de UF. Escolha a unidade territorial MG.

(a) Importe esse mapa em R.

(b) Visualize esse mapa.

2. Considere o mapa de Minas Gerais dividido em Municípios.

(a) Encontre o município de Belo Horizonte e visualize-o.

(b) Encontre a microregião de Belo Horizonte e visualize-a.

(c) Adicione o nome a cada município.

3. Considere o mapa de Minas Gerais dividido em Municípios.

(a) Encontre a lista de vizinhança.

(b) Visualize esse grafo.

(c) Encontre quem são os municípios vizinhos de Belo Horizonte.

4. Considere os dados do exercío 3 do capítulo anterior

(a) Associe esses dados ao mapa de Minas Gerais

(b) Faça um mapa temático colorindo-o de acordo com a proporção calculada

5. Considere os exercídio anterior.

   (a) Exporte o mapa para o formato *shapefile*, incluindo uma coluna com a proporção calculada.

Capítulo 4

Georeferenciamento

Eventualmente trabalhamos com dados que incluem endereço. Esse endereço pode ser o logradouro (Rua, Avenida, etc.) com o número ou apenas o CEP. Os dados de internações do DATASUS, por exemplo, contém o CEP de residência dos pacientes. Neste caso, um CEP pode referir-se a todas as residências de uma rua interia, ou seja, a localização não é exata. Neste capítulo, vamos mostrar como fazer o georeferenciamento de endereços usando uma API de georeferenciamento do googleMaps. Também vamos obter uma imagem de mapa de ruas e sobrepor endereços georeferenciados a essa imagem.

4.1 Georeferenciamento de Logradouro

Um endereço especificado por logradouro é completo quando temos o tipo de logradouro (Av., Rua, etc.), o nome do logradouro, o número do endereço no logradouro, a cidade, estado e país. Suponha que temos alguns endereços, guardados na forma de logradouro, nome do logradouro, número e cidade. No ambiente **R**, teríamos, por exemplo:

```
> ltipo <- c("av", "rua", NA)
> lnome <- c("Brasil", "joubert", "prefeitura municipal")
> enum <- c(3500, 100, NA)
```

todos estes em Belo Horizonte. Note que o terceiro endereço é simplesmente "prefeitura", sem logradouro.

Nós vamos usar uma API do `google.maps` para fazer o georeferenciamento desses endereços, isto é, obter a latitude e longitude desses endereços. Para entender o que essa API faz, digite

```
http://maps.google.com/maps/geo?q=
av+parana+100+Maringa+PR
&output=csv&sensor=true_or_false&key=abcdefg
```

num navegador de internet.

O resultado exibito deve ser

```
200,8,-23.4266623,-51.9431892
```

que nada mais é que uma linha com quatro colunas separadas por ",". A primeira coluna é um código de sucesso da operação (200 é sucesso), cujo significado esta em

```
http://code.google.com/intl/pt-BR/apis/maps/documentation/
javascript/v2/reference.html#GGeoStatusCode
```

A segunda coluna é um código de qualidade (acurácia) do georeferenciamento cujo significado está em

```
http://code.google.com/intl/pt-BR/apis/maps/documentation/
javascript/v2/reference.html#GGeoAddressAccuracy
```

As duas últimas colunas são a latitude e longitude, respectivamente. O código de acurácia de georeferenciamento considera a divisão política dos EUA, isto é, estado, condado, cidade. Mas no Brasil, temos estado, município, bairro. Além disso, como veremos num exemplo, aveninas longas possuem mais de um CEP, sendo mais preciso neste caso georeferenciar por CEP que por logradouro.

Note que esse resultado é nada mais que um arquivo em formato de texto (muito simples) num servidor. Nós podemos ler esse arquivo no **R** com funções do tipo `read.table()`, `read.csv()`, etc. Particularmente eu uso a função `readLines()`, assim:

```
> url <- paste("http://maps.google.com/maps/geo?q=",
+              "av+parana+100+Maringa+PR",
+              "&output=csv&sensor=true_or_false&key=abcdefg", sep="")
> readLines(url, warn=FALSE)

[1] "200,8,-23.4266623,-51.9431892"
```

que retorna uma *string*. Basta quebrar essa `string` nas ","

```
> strsplit(readLines(url, warn=FALSE), ",")

[[1]]
[1] "200"          "8"            "-23.4266623"
[4] "-51.9431892"
```

e converter este resultado para número. Então, temos

```
> as.numeric(strsplit(readLines(url, warn=FALSE), ",")[[1]])

[1] 200.00000   8.00000 -23.42666 -51.94319
```

que pode ser armazenado num vetor.

Muitas vezes temos mais de um endereço para georeferenciar. Por isso, vamos definir uma função para fazer o georeferenciamento, a partir de um logradouro.

```
> fGetLatLonLog <- function(tipo, nome, num, mun, uf, pais) {
+   end <- paste(tipo, nome, num, mun, uf, pais, sep="+")
+   end <- gsub(" ", "+", end, fixed=TRUE)
+   end <- gsub("NA", "", end, fixed=TRUE)
+   end <- gsub("++", "+", end, fixed=TRUE)
+   end <- paste("http://maps.google.com/maps/geo?q=",
+                end,
+                "&output=csv&sensor=true_or_false&key=abcdefg", sep="")
+   end <- sapply(end, readLines, warn=FALSE)
+   end <- t(sapply(strsplit(end, ","), as.numeric))
+   colnames(end) <- c("Status", "Acuracia", "Latitude", "Longitude")
+   rownames(end) <- 1:nrow(end)
+   return(as.data.frame(end))
+ }
```

Aplicando essa função aos três endereços

```
> ll <- fGetLatLonLog(ltipo, lnome, enum, "Maringa", "PR", "BR")
> ll

  Status Acuracia  Latitude Longitude
1    200        8 -23.42084 -51.93622
2    200        8 -23.41993 -51.93421
3    200        9 -25.91839 -53.47374
```

4.2 Localização de Endereço num Setor Censitário

Após georeferenciar, suponha que precisamos encontrar a qual setor censitário cada endereço pertence. Neste exemplo, precisamos do mapa de Belo Horizonte subdividido por setor censitário.

4.2.1 Mapa de Setores Censitários

Os mapas de setores censitários do censo 2010 estão disponívei em

`ftp://geoftp.ibge.gov.br/malhas_digitais/`
`censo_2010/setores_censitarios/`

Podemos, por exemplo, fazer o download do conjunto de mapas do Parná.

Um *shapefile* pode ser lido em **R**, entre outras, com a função `readOGR()` do pacote **rgdal**. No conjunto de mapas de 2010 disponíveis, o *shapefile* '41SEE250GC_SIR' armazena os polígonos do 17691 setores censitários 2010 do Paraná

```
> require(rgdal)
> sepr <- readOGR("../mapas/pr/", "41SEE250GC_SIR",
+                 input_field_name_encoding='latin1')
```

Como são muitos setores, vamos selecionar apenas os setores pertencentes a Maringá. Para isso, observamos que nos atributos desse *shapefile* a coluna 'NM_MUNICIP' contém o nome dos municípios

```
> head(sepr@data$'NM_MUNICIP')

[1] ALTAMIRA DO PARAN\xc1 ALTAMIRA DO PARAN\xc1
[3] ALTAMIRA DO PARAN\xc1 ALTAMIRA DO PARAN\xc1
[5] ALTAMIRA DO PARAN\xc1 ALTAMIRA DO PARAN\xc1
399 Levels: ABATI\xc1 ... XAMBR\xca

> sepr@data$munsname <- as.character(sepr$'NM_MUNICIP')
> Encoding(sepr$munsname) <- 'latin1'
> head(id.mar <- which(substr(sepr@data$munsname, 1, 6)=='MARING'))

[1] 10864 10865 10866 10867 10868 10869

> semar <- sepr[id.mar,]
> dim(semar@data)

[1] 562  14
```

Podemos obter as coordenadas do polígono de cada setor censitário. Também podemos obter o centróide de cada um com

```
> head(coordinates(semar))

           [,1]      [,2]
10863 -52.03733 -23.40853
10864 -52.02065 -23.35273
10865 -51.94619 -23.37422
10866 -51.96480 -23.30397
10867 -51.92155 -23.33257
10868 -51.87804 -23.32335
```

Note que as coordenadas desse mapa estão em graus de latitude e longitude.

4.2.2 Endereço Pontual em Setor Censitários

Podemos verificar a qual setor censitário pertencem os três endereços georeferenciados. Inicialmente, vamos coverter o objeto `ll` num formato adequado

```
> llxy <- data.frame(x=ll[,4], y=ll[,3])
> coordinates(llxy) = ~x+y
> llxy@proj4string <- semar@proj4string
> class(llxy)
```

```
[1] "SpatialPoints"
attr(,"package")
[1] "sp"
```

Após isso, usamos a função `over()` do pacote **sp** para encontrar os polygonos do mapa que contém esses pontos.

```
> o <- over(llxy, semar)
> o
```

```
     ID      CD_GEOCODI   TIPO CD_GEOCODB
1 14643 411520005010008 URBANO       <NA>
2 14653 411520005010018 URBANO       <NA>
3    NA            <NA>   <NA>       <NA>
  NM_BAIRRO  CD_GEOCODS NM_SUBDIST CD_GEOCODD
1      <NA> 41152000501     ZONA 1  411520005
2      <NA> 41152000501     ZONA 1  411520005
3      <NA>        <NA>       <NA>       <NA>
  NM_DISTRIT CD_GEOCODM NM_MUNICIP   NM_MICRO
1 MARING\xc1    4115200 MARING\xc1 MARING\xc1
2 MARING\xc1    4115200 MARING\xc1 MARING\xc1
3       <NA>       <NA>       <NA>       <NA>
                   NM_MESO munsname
1 NORTE CENTRAL PARANAENSE  MARINGÁ
2 NORTE CENTRAL PARANAENSE  MARINGÁ
3                     <NA>     <NA>
```

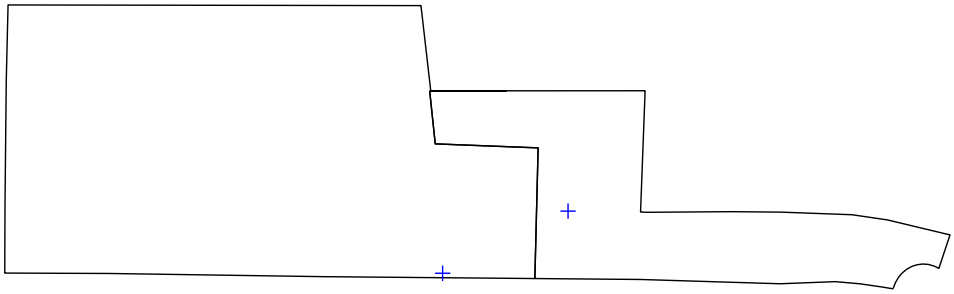
Precisamos saber a posição desses poligonos no mapa

```
> id <- pmatch(o$ID, semar$ID, duplicates.ok=TRUE)
> id
```

```
[1] 21 31 NA
```

e após isso, usar essa informação. Por exemplo, podemos encontrar o sub-mapa desses setores e visualizar

```
> ssel <- semar[id[1:2],]
> par(mar=c(0,0,0,0))
> plot(ssel)
> points(llxy, pch=3, col=4)
```



4.3 Georeferenciamento de CEP

Um interesse particular é fazer o georeferenciamento de CEPs. Podemos fazer o georeferenciamento de CEP usando a API de georeferenciamento do GoogleMaps. Notemos que o código de acurácia de CEP é 5. Exemplo:

```
http://maps.google.com/maps/geo?
q=cep=87013260+Maringa+PR
&output=csv&sensor=true_or_false&key=abcdefg
```

Vamos criar uma função para georeferenciar a partir de CEP

```
> fGetLatLonCEP <- function(cep, mun, uf, pais) {
+   end <- paste(cep, mun, uf, pais, sep="+")
+   end <- gsub(" ", "+", end, fixed=TRUE)
+   end <- gsub("NA", "", end, fixed=TRUE)
+   end <- gsub("++", "+", end, fixed=TRUE)
+   end<- paste("http://maps.google.com/maps/geo?q=cep=",
+               end,
+               "&output=csv&sensor=true_or_false&key=abcdefg", sep="")
+   end <- sapply(end, readLines, warn=FALSE)
+   end <- t(sapply(strsplit(end, ","), as.numeric))
+   colnames(end) <- c("Status", "Acuracia", "Latitude", "Longitude")
+   rownames(end) <- 1:nrow(end)
+   return(as.data.frame(end))
+ }
> ceps3 <- c(87013260, 87010380, 87013070)
> ll2 <- fGetLatLonCEP(ceps3, "Maringa", "PR", "Brasil")
> ll2

  Status Acuracia  Latitude Longitude
1    200        5 -23.42482 -51.94082
2    200        5 -23.42444 -51.93289
3    200        5 -23.41463 -51.94810
```

Há uma grande vantagem em georeferenciar por CEP em vez de logradouro. Isso porque algumas ruas ou avenidas muito longas possuem mais de um CEP. No exemplo da Av. Amazonas, que é uma avenida muito longa, o georeferenciamento ao nível de rua é menos preciso que ao nível de CEP. Nesses casos, a coordenada geográfica é o ponto central do logradouro ou do CEP.

4.4 Visualizando no GogleMaps

Podemos visualizar dados georeferenciados no google Maps. Isso é feito adicionando um *layer*. Podemos visualizar dessa forma com o pacote **plotGoogleMaps**, [16].

Podemos visualizar os pontos representando endereços. Para isso os pontos devem estar representados por um objeto da classe `SpatialPointsDataFrame`. Além disso é necessário a informação da projeção cartográfica na qual as coordenadas estão. Assim, vamos considerar o `SpatialPoints` com os três endereços georeferenciados e adicionar um `data.frame` apenas com uma coluna de identificação para visualizar esses dados no googleMaps.

```
> require(plotGoogleMaps)
> dat <- data.frame(Tipo=ltipo, Logradouro=lnome, Numero=enum)
> llxydf <- SpatialPointsDataFrame(llxy, dat)
> plotGoogleMaps(llxy)
```

O resultado pode é visualizado num navegador de internet. Mapas de áreas também podem ser visualizados dessa forma.

4.5 Imagens do GoogleMaps

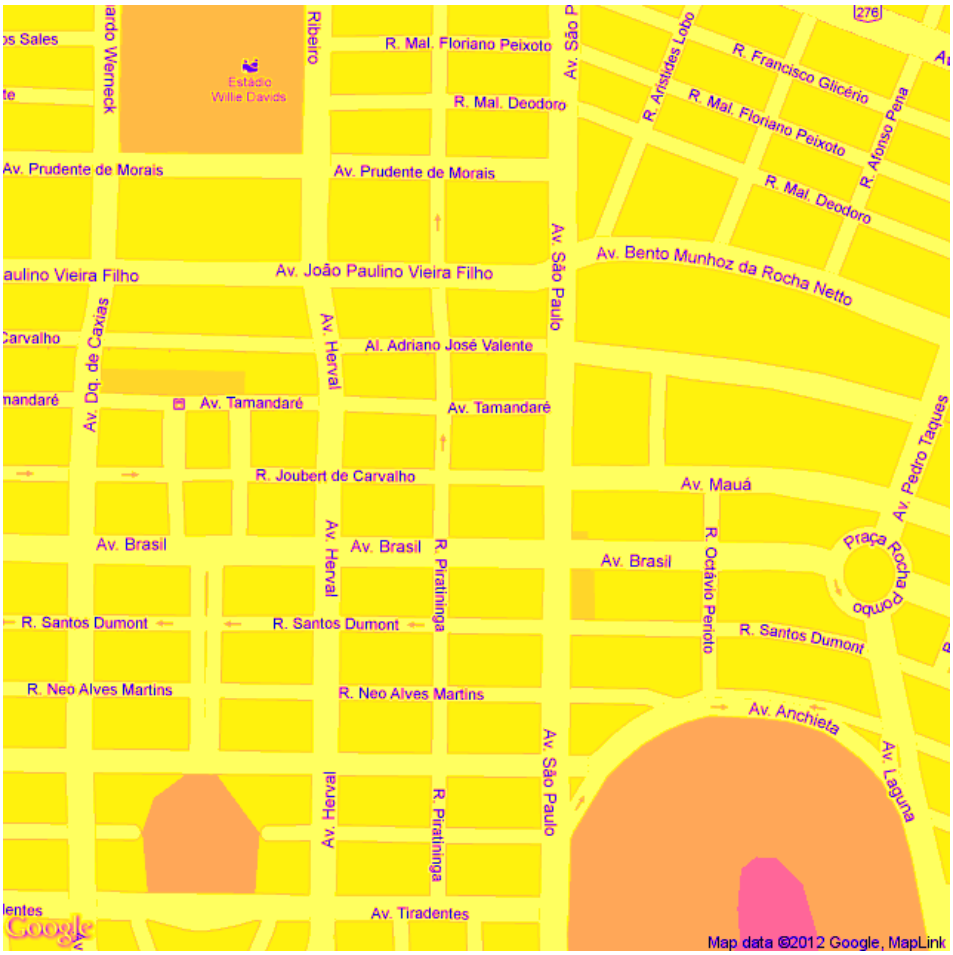
O googleMaps é muito utilizado para a localização de um endereço. Esse endereço é mostrado num mapa com imagens de ruas ou numa imagem de satélite ou ambas sobrepostas. Uma imagem do googleMaps pode ser armazenada em formato de arquivos de imagens. Estes arquivos podem ser carregados em **R** onde podemos georeferenciar uma imagem. Assim, podemos trabalhar com mapas de setores censitários, endereços georeferenciados e imagens georeferenciadas.

4.5.1 Obtendo Imagens do GoogleMaps

Nesta seção nós vamos obter uma imagem do googleMaps e vizualizá-la usando o **R**. Para obter uma imagem, vamos utilizar o pacote **GoogleMaps**. Vamos obter uma imagem com centro o primeiro endereço georeferenciado

```
> require(RgoogleMaps)
> cooimg <- GetMap(center=c('lon'=ll[2,4], 'lat'=ll[2,3]), zoom=16,
+   maptype="roadmap", destfile="marimg.png", format="png32")
```

A imagem obtida foi gravada num arquivo de imagem. Vamos agora usar um pacote para ler a imagem




```
> require(rgdal)
> imgmar <- readGDAL("marimg.png", half.cell=c(0,0))
```

e vamos vizualizá-la com

```
> par(mar=c(0,0,0,0))
> image(imgmar, col=bpy.colors(50))
```

4.5.2 Georeferenciamento de imagens

Nesta seção, vamos georeferenciar a imagem obtida e sobrepor os endereços georeferenciados anteriormente a essa imagem. Inicialmente vamos observar as coordenadas limites da imagem, armazenadas ao buscar a imagem no googleMaps

```
> cooimg[['BBOX']]

$ll
          lat       lon
```

```
[1,] -23.42624 -51.94107

$ur
           lat       lon
[1,] -23.41364 -51.92733
```

e vamos observar as coordenadas da imagem lida

```
> bbox(imgmar)

   min   max
x -0.5 639.5
y -0.5 639.5
```

Vamos alterar os limites das coordenadas da imagem usando os limites da imagem buscada

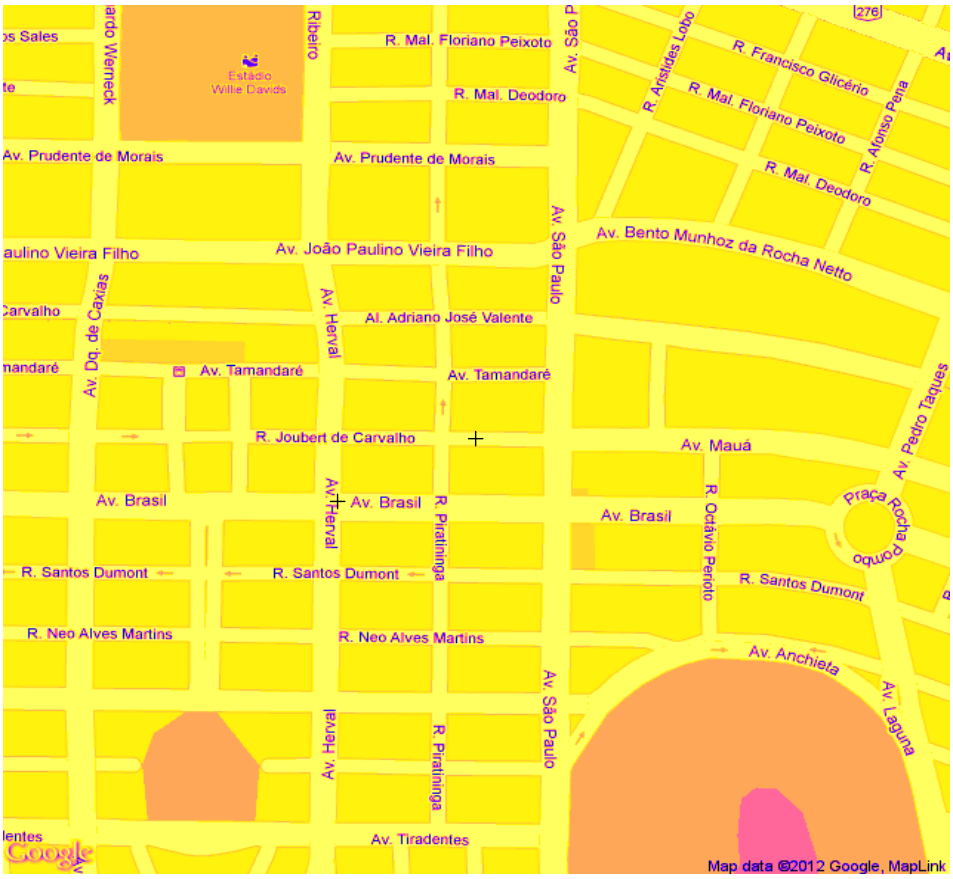
```
> coobb <- t(do.call('rbind', cooimg[['BBOX']]))[2:1,]
> imgmar@bbox <- coobb
```

Assim, ja podemos visualizar a imagem e sobrepor os três endereços georeferenciados.

```
> par(mar=c(0,0,0,0))
> image(imgmar, col=bpy.colors(50))
> points(ll[,4:3], pch=3)
```

Mas, para considerar o mapa de setores, precisamos projetar essa imagem no sistema de coordenadas do mapa de setores censitários. Para isso, podemos alterar novamente os limites da imagem

```
> par(mar=c(0,0,0,0))
> image(imgmar, col=bpy.colors(50))
> plot(semar, border="blue", add=TRUE)
> points(llxy, pch=3)
```


Estádio
Willie Davids
R. Mal. Floriano Peixoto
R. Mal. Deodoro
Av. Prudente de Morais
Av. João Paulino Vieira Filho
Av. São Paulo
Av. Herval
Al. Adriano José Valente
Av. Tamandaré
R. Joubert de Carvalho
Av. Brasil
R. Piratininga
R. Santos Dumont
R. Neo Alves Martins
Av. Tiradentes
R. Aristides Lobo
R. Francisco Glicério
R. Afonso Pena
Av. Bento Munhoz da Rocha Netto
Av. Mauá
R. Octávio Peixoto
Av. Pedro Taques
Praça Rocha Pombo
Av. Anchieta
Av. Laguna
Av. Dq. de Caxias
276
Google
Map data ©2012 Google, MapLink


Estádio
Willie Davids
R. Mal. Floriano Peixoto
R. Mal. Deodoro
Av. Prudente de Morais
Av. João Paulino Vieira Filho
Av. Dq. de Caxias
Av. Tamandaré
Al. Adriano José Valente
Av. Herval
R. Joubert de Carvalho
Av. Brasil
R. Santos Dumont
R. Neo Alves Martins
Av. São Paulo
R. Pratininga
Av. Tiradentes
R. Francisco Glicério
R. Mal. Floriano Peixoto
Av. Bento Munhoz da Rocha Netto
Av. Mauá
Av. Pedro Taques
Praça Rocha Pombo
Av. Anchieta
Map data ©2012 Google, MapLink

Capítulo 5

Dependência Espacial em Dados de Áreas

Neste capítulo introduzimos algumas técnicas de detecção de dependência espacial em dados de áreas. Abordamos a necessidade de especificação de uma matriz de vizinhança espacial e como usá-la em alguns testes de dependência espacial. Também apresentamos dois modelos comumente utilizados em análise de dados de áreas. Algumas aplicações são feitas para ilustração.

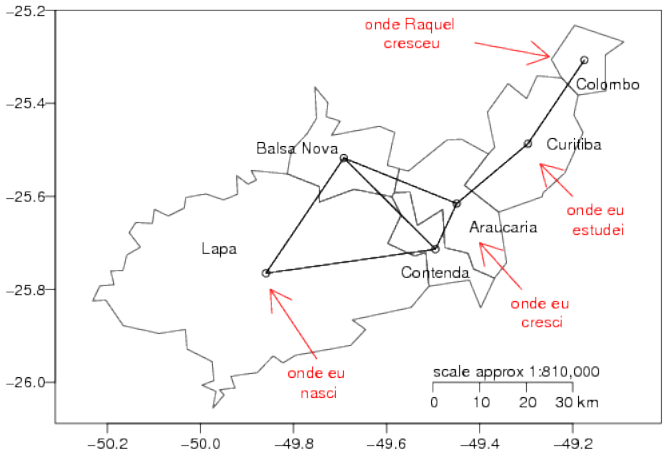
5.1 Matriz de Vizinhança

Assim como os métodos de análise de séries temporais, os métodos de análise de dados de áreas devem levar em conta a relação de vizinhança entre as observações. Nas séries temporais basta que os dados estejam ordenados na ordem do tempo. Nos dados espaciais, não é possível uma ordenação, assim precisamos sempre associar aos dados uma matriz que indica a vizinhança dos dados.

Suponha, por exemplo, que temos um mapa como o da Figura 5.1. Nessa Figura temos cinco municípios cujos nomes estão indicados. Podemos considerar vizinhos os municípios que partilham de fronteira comum. Desta forma, temos um grafo associado que representa essa estrutura de vizinhança.

Como vimos, a estrutura de vizinhança pode ser representada por um grafo. Ainda, a estrutura de vizinhança pode ser representada pela matriz de adjacência, $\boldsymbol{A}$, associada, que é uma matriz cujo elemento $\boldsymbol{A}_{i,j}$ é 1 (um) se a área i é vizinha da área j e 0 (zero) caso contrário. Ainda, $\boldsymbol{A}_{i,i} = 0$, a diagonal principal é zero, ou seja, uma área não é vizinha dela mesma. Uma propriedade importante desta matriz é sua simetria, ou seja, se a área i é vizinha da área j, logo a área j é vizinha da área i.

Para o grafo da Figura 5.1 temos a matriz de vizinhança abaixo:



Lapa	0	1	1	0	0	0
Balsa Nova	1	0	1	1	0	0
Contenda	1	1	0	1	0	0
Araucária	0	1	1	0	1	0
Curitiba	0	0	0	1	0	1
Colombo	0	0	0	0	1	0

É comum em muitos métodos estatísticos especificar uma matriz de 'vizinhança padronizada', $\boldsymbol{W}$. Por exemplo, de forma que as linhas somem 1. Para o caso de $\boldsymbol{W}$ obtida de $\boldsymbol{A}$, temos $\boldsymbol{W}_{i,j} = 1/d_i$ se a área i é vizinha da área j, $d_i = \sum_j \boldsymbol{A}_{i,j}$ é o número de vizinho da área i, e 0 (zero) caso contrário. Neste caso, $\boldsymbol{W}$ não é simétrica sempre que houver algum par $i \neq j$ tal que $d_i \neq d_j$. Ou seja, se houver algum par de áreas com número de vizinhos diferentes, $\boldsymbol{W}$ não é simétrica.

No exemplo da matriz de adjacência do grafo da Figura 5.1, temos a seguinte matriz $\boldsymbol{W}$:

Lapa	0.00	0.50	0.50	0.00	0.00	0.00
Balsa Nova	0.33	0.00	0.33	0.33	0.00	0.00
Contenda	0.33	0.33	0.00	0.33	0.00	0.00
Araucária	0.00	0.33	0.33	0.00	0.33	0.00
Curitiba	0.00	0.00	0.00	0.50	0.00	0.50
Colombo	0.00	0.00	0.00	0.00	1.00	0.00

Vários modelos usam $\boldsymbol{W}$ como definida acima. Devido ao fato de $\boldsymbol{W}$ definida acima não ser simétrica, alguns *softwares* usam outra padronização de $\boldsymbol{A}$. Seja $\lambda_1, \lambda_2, ..., \lambda_n$ os autovalores ordenados em ordem decrescente de $\boldsymbol{A}$. Podemos considerar uma matriz dada por $\boldsymbol{A}_{i,j}/\lambda_1$. Em geral, λ_1 é próximo, mas menor, que o maior número de vizinhos.

Tanto $\boldsymbol{A}$ quanto $\boldsymbol{W}$, geralmente, são esparsas, isto é, a maioria de seus elementos são iguais a 0 (zero). Desta forma é comum representá-las, em **R**, por um objeto do tipo lista. Assim, podemos ter uma lista de tamanho n e cada elemento i um vetor indicando as áreas vizinhas da área i. Suponha, por exemplo, que o número médio de vizinhos é cinco, assim em vez de usar memória para uma matriz com $n \times n$ elementos, usamos memória para armazenar apenas $5 \times n$ elementos.

Vamos considerar como exemplo o mapa do estado do Paraná dividido em 399 municípios.

```
> require(rgdal)
> prm <- readOGR("../mapas/", "prdata", input_field_name_encoding='latin1')

OGR data source with driver: ESRI Shapefile
Source: "../mapas/", layer: "prdata"
with 399 features and 51 fields
Feature type: wkbPolygon with 2 dimensions
```

e considerar a função `poly2nb()` do pacote **spdep**, [7], para construir a lista de vizinhança.

```
> require(spdep)
> nbprm <- poly2nb(prm)
```

E notamos alguns dados sucintos da lista com

```
> nbprm

Neighbour list object:
Number of regions: 399
Number of nonzero links: 2226
Percentage nonzero weights: 1.398232
Average number of links: 5.578947
```

e

```
> summary(nbprm)
```

```
Neighbour list object:
Number of regions: 399
Number of nonzero links: 2226
Percentage nonzero weights: 1.398232
Average number of links: 5.578947
Link number distribution:

 2  3  4  5  6  7  8  9 10 11 12
 7 40 72 86 83 56 27 16  9  2  1
7 least connected regions:
30 49 174 255 275 332 359 with 2 links
1 most connected region:
68 with 12 links
```

5.2 Índice de Moran

Temos de estatística básica o coeficiente de correlação de Pearson dado por

$$\frac{1}{n}\sum_{i=1}^{n}(\frac{y_i-\overline{y}}{n})(\frac{x_i-\overline{x}}{n})$$

em que $\overline{y}$ e $\overline{x}$ são, respectivamente, a média aritmética de y e x.

Análogamente, poderíamos pensar num índice de autocorrelação. No caso espacial, temos o índice I de Moran é dado por:

$$I = \frac{1}{\sum_{i\neq j} w_{ij}} \sum_{i\neq j} w_{ij} \left(\frac{y_i-\bar{y}}{s_y}\right)\left(\frac{y_j-\bar{y}}{s_y}\right) \tag{5.1}$$

qu é o índice autocorrelação espacial de Moran ou simplesmente índice de Moran, em homenagem a P. A. P. Moran, estatístico australiano que estudou o seu comportamento, Moran (1950).

Vamos considerar os dados de salário médio por município do Paraná e calcular o índice de Moran. Para isso, precisamos dos dados da variável e da lista de vizinhança padronizada. Esta, pode ser obtida pela função `nb2listw()` do pacote **spdep**.

```
> nbw <- nb2listw(nbprm)
> names(prm)

 [1] "cod6"       "GEOCODIG_M" "UF"
 [4] "Sigla"      "Nome_Munic" "RegiÃ.o"
 [7] "MesorregiÃ" "Nome_Meso"  "Microrregi"
[10] "Nome_Micro" "codMun"     "pop2000"
[13] "pop2010"    "s2006"      "s2007"
[16] "s2008"      "s2009"      "s2010"
[19] "MunicÃ.pio" "n1994"      "n1995"
```

```
[22] "n1996"      "n1997"      "n1998"
[25] "n1999"      "n2000"      "n2001"
[28] "n2002"      "n2003"      "n2004"
[31] "n2005"      "n2006"      "n2007"
[34] "n2008"      "n2009"      "n2010"
[37] "o1996"      "o1997"      "o1998"
[40] "o1999"      "o2000"      "o2001"
[43] "o2002"      "o2003"      "o2004"
[46] "o2005"      "o2006"      "o2007"
[49] "o2008"      "o2009"      "o2010"
```

```
> args(moran)

function (x, listw, n, S0, zero.policy = NULL, NAOK = FALSE)
NULL
```

```
> n <- length(nbprm)
> names(nbw)

[1] "style"      "neighbours" "weights"
```

```
> mopop <- moran(prm$s2010, nbw, n, Szero(nbw))
> mopop

$I
[1] 0.2048081

$K
[1] 14.65432
```

Dois testes de significância estão implementados. Um baseado na Normalidade assintótica, função `moran.test()` e outro baseado num teste via simulações de Monte Carlo `moran.mc()`.

```
> args(moran.test)

function (x, listw, randomisation = TRUE, zero.policy = NULL,
    alternative = "greater", rank = FALSE, na.action = na.fail,
    spChk = NULL, adjust.n = TRUE)
NULL
```

```
> moran.test(prm$s2010, nbw)

        Moran's I test under randomisation

data:  prm$s2010
weights: nbw
```

```
Moran I statistic standard deviate = 6.8686,
p-value = 3.241e-12
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation
     0.2048081103     -0.0025125628
         Variance
     0.0009110581
```

```
> args(moran.mc)

function (x, listw, nsim, zero.policy = NULL, alternative = "greater",
    na.action = na.fail, spChk = NULL, return_boot = FALSE)
NULL
```

```
> moran.mc(prm$s2010, nbw, 999)

        Monte-Carlo simulation of Moran's I

data:  prm$s2010
weights: nbw
number of simulations + 1: 1000

statistic = 0.2048, observed rank = 1000,
p-value = 0.001
alternative hypothesis: greater
```

Vamos aplicar também a dados de salário médio por município do Paraná em 2006 a 2010.

5.3 Índice de Moran Bayesiano Empírico

Em muitas situações, os dados de áreas analisados são taxas, por exemplo, taxa de mortalidade infantil. Nesta situação, é melhor usar o índice bayesiano empírico, [3]. Este índice é dado por:

$$EBI = \frac{1}{S_0} \sum_{i \neq j} w_i z_i z_j \tag{5.2}$$

em que

- $S_0 = \sum_{i \neq j} \boldsymbol{W}_{i,j}$,
- $z_i = \frac{r_i - \hat{r}}{\sqrt{v_i}}$,
- $r_i = y_i/N_i$, y_i é o número de casos na área i e N_i é a população sob risco na área i,

- $v_i = \hat{r}/N_i + \tilde{r}I(\tilde{r} > 0)$,
- $\hat{r} = \sum_i y_i / \sum_i N_i$ é a taxa da região toda de estudo
- $\tilde{r} = s^2 - \hat{r}/m$, $m = \sum_{i=1}^{n} N_i/n$,
- $s^2 = \sum_i N_i(r_i - \hat{r})^2 / \sum_i N_i$.

Como exemplo, vamos considerar os dados de mortalidade infantil por município no Paraná. Vamos considerar o total de nascidos vivos de 1996 a 2010 e o total de ótitos infantis nesse período, por município.

```
> names(prm@data)
```

```
 [1] "cod6"       "GEOCODIG_M" "UF"
 [4] "Sigla"      "Nome_Munic" "RegiÃ.o"
 [7] "MesorregiÃ" "Nome_Meso"  "Microrregi"
[10] "Nome_Micro" "codMun"     "pop2000"
[13] "pop2010"    "s2006"      "s2007"
[16] "s2008"      "s2009"      "s2010"
[19] "MunicÃ.pio" "n1994"      "n1995"
[22] "n1996"      "n1997"      "n1998"
[25] "n1999"      "n2000"      "n2001"
[28] "n2002"      "n2003"      "n2004"
[31] "n2005"      "n2006"      "n2007"
[34] "n2008"      "n2009"      "n2010"
[37] "o1996"      "o1997"      "o1998"
[40] "o1999"      "o2000"      "o2001"
[43] "o2002"      "o2003"      "o2004"
[46] "o2005"      "o2006"      "o2007"
[49] "o2008"      "o2009"      "o2010"
```

```
> summary(t.nv <- rowSums(prm@data[, 22:36], na.rm=TRUE))
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    260    1112    2131    6273    4452  400200
```

```
> summary(t.obi <- rowSums(prm@data[,37:51], na.rm=TRUE))
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    1.0    17.5    38.0   104.5    84.0  5153.0
```

O teste de autocorrelação de taxas usando o índice de Moran bayesiano empírico pode ser feito com a função `EBImoran.mc()` do pacote **spdep**.

```
> args(EBImoran.mc)
```

```
function (n, x, listw, nsim, zero.policy = NULL, alternative = "greater",
    spChk = NULL, return_boot = FALSE)
NULL
```

```
> EBImoran.mc(t.nv, t.obi, nbw, 999)

        Monte-Carlo simulation of Empirical Bayes
        Index

data:  cases: t.nv, risk population: t.obi
weights: nbw
number of simulations + 1: 1000

statistic = 0.1806, observed rank = 1000,
p-value = 0.001
alternative hypothesis: greater
```

Capítulo 6

Introdução aos Modelos Espaciais para Dados de Áreas

Os modelos de regressão podem ser estendidos para analisar esses dados espaciais. As extensões são necessárias para modelar a estrutura de dependência espacial dos dados. Essa dependência espacial pode ser modelada diretamente utilizando modelos de regressão espacial para dados gaussianos, na abordagem frequentista [1], ou considerando efeitos aleatórios espaciais para dados gaussianos ou não, na abordagem bayesiana, [4].

Neste capítulo introduzimos alguns modelos mais comuns para análise de dados de áreas, os modelos SAR e CAR. Também, fazemos uma aplicação ao conjunto de dados para ilustrar o uso dos modelos CAR e SAR.

6.1 Os modelos SAR e CAR

Seja uma região $\boldsymbol{D}$, particionada em n áreas disjuntas, $A_1, ..., A_n$ com $A_i \cup A_j =$ e $\cup_{i=1}^{n} A_i = \boldsymbol{D}$. Os 399 municípios que fazem parte do estado do Paraná, por exemplo, formam uma partição do estado do Paraná. Seja y_i o valor observado de um determinado fenômeno na área A_i . O interesse é modelar o processo estocástico $Y(A_i)$, $i = 1, ..., n$ ou simplesmente $\boldsymbol{y} = (y_1, ..., y_n)$. Dois modelos para esse processo foram propostos e são muito utilizados: o modelo autoregressivo simultâneo - SAR [30] e o modelo autoregressivo condicional - CAR [5].

O modelo SAR é determinado pela solução simultânea do sistema de equações dado por:

$$y_i = \mu_i + \sum_{j=1}^{n} b_{ij}(y_j - \mu_j) + \epsilon_i \tag{6.1}$$

onde $\epsilon = (\epsilon_1, ..., \epsilon_n)' \sim N(0, \Lambda)$ com Λ diagonal, $E(y_i) = \mu_i$, e b_{ij} são constantes conhecidas ou desconhecidas e $b_{ii} = 0, i = 1, ..., n$. A distribuição conjunta de $\boldsymbol{y}$ é

$$\boldsymbol{y} \sim N(\mu, (I_n - B)^{-1}\Lambda(I_n - B)^{-1'}) \,, \tag{6.2}$$

em que $B_{ij} = (b_{ij})$.

O modelo CAR é determinado por um conjunto de distribuições condicionais

$$y_i|y_{-i} \sim N(\mu_i + \sum_{j=1}^{n} c_{ij}(y_j - \mu_j), \tau/d_i) \tag{6.3}$$

em que $y_{-i} = \{y_j : j \leq i\}$ são os valores de $\boldsymbol{y}$ nas áreas vizinhas de i, $E(y_i) = \mu_i$, τ e d_i c_{ij} são constantes conhecidas ou desconhecidas $c_{ij} = 0$, $i = 0, ..., n$. Este modelo é *condicional* por que τ/d_i é a variância condicinal.

$$Z \sim N(\mu, (I_n - C)^{-1}T^{-1}) \tag{6.4}$$

em que $C_{ij} = c_{ij}$ e $T = \tau diag\{d_1, ..., d_n\}$.

Geralmente adota-se $B = \rho_s \boldsymbol{W}$ e $C = \rho_c \boldsymbol{W}$, em que ρ_s e ρ_c são parâmetros de correlação espacial dos modelos CAR e SAR, respectivamente, e $\boldsymbol{W}$ reflete a estrutura de vizinhança entre as áreas. Uma definição bastante comum de $\boldsymbol{W}$ é feita a partir da matriz de adjacência $\boldsymbol{A}$. A matriz de adjacência é definida fazendo $\boldsymbol{A}_{ij} = 1$ se as áreas i e j tem borda comum e $\boldsymbol{A}_{ij} = 0$ caso contrario. Por definição $\boldsymbol{A}_{ii} = 0$. Neste trabalho consideramos que $\boldsymbol{W}$ é definida de forma que suas linhas somem 1, fazendo $\boldsymbol{W}_{ij} = (a_{ij}/a_{i.})$, em que $a_{i.} = d_i$ é o número de vizinhos da área i.

6.2 Um Exemplo de Aplicação

Vamos considerar os dados de salário médio mensal por município do Paraná em 2006 a 2010, considerados no capítulo anterior.

```
> require(rgdal)
> alld <- readOGR("../mapas/", "prdata",
+                 input_field_name_encoding='latin1')
```

```
OGR data source with driver: ESRI Shapefile
Source: "../mapas/", layer: "prdata"
with 399 features and 51 fields
Feature type: wkbPolygon with 2 dimensions
```

Nesses dados temos também população, óbitos infantis e nascidos vivos.

```
> names(alld)
```

```
 [1] "cod6"       "GEOCODIG_M" "UF"         "Sigla"      "Nome_Munic"
 [6] "RegiÃ.o"    "MesorregiÃ" "Nome_Meso"  "Microrregi" "Nome_Micro"
[11] "codMun"     "pop2000"    "pop2010"    "s2006"      "s2007"
[16] "s2008"      "s2009"      "s2010"      "MunicÃ.pio" "n1994"
[21] "n1995"      "n1996"      "n1997"      "n1998"      "n1999"
[26] "n2000"      "n2001"      "n2002"      "n2003"      "n2004"
[31] "n2005"      "n2006"      "n2007"      "n2008"      "n2009"
```

```
[36] "n2010"      "o1996"      "o1997"      "o1998"      "o1999"
[41] "o2000"      "o2001"      "o2002"      "o2003"      "o2004"
[46] "o2005"      "o2006"      "o2007"      "o2008"      "o2009"
[51] "o2010"
```

```
> require(spdep)
> nbpr <- poly2nb(alld)
> args(spautolm)
```

```
function (formula, data = list(), listw, weights, na.action,
    family = "SAR", method = "full", verbose = NULL, interval = NULL,
    zero.policy = NULL, tol.solve = .Machine$double.eps, llprof = NULL,
    control = list())
NULL
```

```
> nbw <- nb2listw(nbpr)
> msar <- spautolm(s2010~I(pop2010/1000), alld@data, nbw)
> summary(msar)
```

```
Call: spautolm(formula = s2010 ~ I(pop2010/1000), data = alld@data,
    listw = nbw)

Residuals:
      Min        1Q    Median        3Q       Max
-0.790754 -0.194293 -0.024074  0.092814  2.494365

Coefficients:
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)     1.90391860 0.02050302  92.860 < 2.2e-16
I(pop2010/1000) 0.00139186 0.00016719   8.325 < 2.2e-16

Lambda: 0.19772 LR test value: 6.8879 p-value: 0.0086783

Log likelihood: -115.4983
ML residual variance (sigma squared): 0.10369, (sigma: 0.32202)
Number of observations: 399
Number of parameters estimated: 4
AIC: 239
```

```
> mcar <- spautolm(s2010~I(pop2010/1000), alld@data, nbw, family='CAR')
> summary(mcar)
```

```
Call: spautolm(formula = s2010 ~ I(pop2010/1000), data = alld@data,
    listw = nbw, family = "CAR")

Residuals:
```

```
      Min        1Q    Median        3Q       Max
-1.125070 -0.192067 -0.020721  0.093780  2.486310

Coefficients:
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)     1.89399399 0.01809358 104.678 < 2.2e-16
I(pop2010/1000) 0.00172849 0.00016699  10.351 < 2.2e-16

Lambda: 0.14552 LR test value: 2.4931 p-value: 0.11434

Log likelihood: -117.6957
ML residual variance (sigma squared): 0.10541, (sigma: 0.32467)
Number of observations: 399
Number of parameters estimated: 4
AIC: 243.39
```

Usando a matriz de adjacencia, em lugar da padronizaca

```
> nbwb <- nb2listw(nbpr, style='B')
> msarb <- spautolm(s2010~I(pop2010/1000), alld@data, nbwb)
> summary(msarb)

Call: spautolm(formula = s2010 ~ I(pop2010/1000), data = alld@data,
    listw = nbwb)

Residuals:
      Min        1Q    Median        3Q       Max
-0.776217 -0.187197 -0.023629  0.095119  2.484615

Coefficients:
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)     1.90292009 0.02077825 91.5823 < 2.2e-16
I(pop2010/1000) 0.00132760 0.00016851  7.8787 3.331e-15

Lambda: 0.038582 LR test value: 7.0709 p-value: 0.0078344

Log likelihood: -115.4068
ML residual variance (sigma squared): 0.10349, (sigma: 0.32171)
Number of observations: 399
Number of parameters estimated: 4
AIC: 238.81

> mcarb <- spautolm(s2010~I(pop2010/1000), alld@data, nbwb, family='CAR')
> summary(mcarb)

Call: spautolm(formula = s2010 ~ I(pop2010/1000), data = alld@data,
    listw = nbwb, family = "CAR")
```

```
Residuals:
      Min        1Q    Median        3Q       Max
-0.749958 -0.194680 -0.039350  0.090763  2.412780

Coefficients:
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)     1.90157378 0.02252063 84.4370 < 2.2e-16
I(pop2010/1000) 0.00126539 0.00016898  7.4884 6.972e-14

Lambda: 0.087465 LR test value: 8.8652 p-value: 0.0029066

Log likelihood: -114.5096
ML residual variance (sigma squared): 0.10131, (sigma: 0.31829)
Number of observations: 399
Number of parameters estimated: 4
AIC: 237.02
```

Padronizando pelo autovalor

```
> eig <- eigen(nb2mat(nbpr, style='B'), only.values=TRUE)$values
> range(eig)

[1] -3.297341  6.419280

> e1 <- max(eig)
> nbw2 <- nb2listw(nbpr, lapply(nbpr, function(x)
+                                rep(1/e1, length(x))), style='B')
> nbw2$w[[1]]

[1] 0.1557807 0.1557807 0.1557807 0.1557807 0.1557807 0.1557807

> nbw2$w[[2]]

[1] 0.1557807 0.1557807 0.1557807

> msar2 <- spautolm(s2010~I(pop2010/1000), alld@data, nbw2)
> mcar2 <- spautolm(s2010~I(pop2010/1000), alld@data, nbw2, family='CAR')
> c(msar$lambda, msarb$lambda, msar2$lambda)

    lambda     lambda     lambda
0.19771650 0.03858235 0.24767091

> c(mcar$lambda, mcarb$lambda, mcar2$lambda)

    lambda     lambda     lambda
0.14551577 0.08746461 0.56145985
```

```
> c(msar$LL0, msarb$LL0, msar2$LL0,
+   mcar$LL0, mcarb$LL0, mcar2$LL0) ### iguais p ser msmo mod

[1] -118.9422 -118.9422 -118.9422 -118.9422 -118.9422 -118.9422

> c(msar$LL, msarb$LL, msar2$LL,
+   mcar$LL, mcarb$LL, mcar2$LL) ### iguais dentro da classe

[1] -115.4983 -115.4068 -115.4068 -117.6957 -114.5096 -114.5096

>
```

Capítulo 7

Introdução aos modelos bayesianos espaciais

A abordagem bayesiana de análise de dados georeferenciados por áreas tem sido a mais utilizada, principalmente devido ao fato de ser mais facilmente estendida a modelos mais complexos do que a abordagem frequentista. Nessa abordagem, a componente espacial é geralmente considerada como um efeito aleatório cuja distribuição a priori autogregressiva condicional ou CAR, do inglês ***C**onditional **A**uto**R**egressive*, para os efeitos aleatórios espaciais desconhecidos.

7.1 Modelo espacial básico

7.1.1 Introdução

Nesta seção nós vamos considerar que temos observações y_i, $i = 1, ..., n$, feitas em n áreas geográficas. Podemos considerar também um vetor de covariáveis $\boldsymbol{X}_i$ observadas em cada área. Um modelo que pode ser considerado para esses dados é o modelo linear generalizado misto, ou *Generalized Linear Mixed Model* - GLMM. Esse modelo é utilizado para modelar a esperança da variável de interesse de forma linear através de uma função de ligação:

$$\mu_i = E(y_i) = g(\eta_i), \text{ onde, } \eta_i = \boldsymbol{X}_i\boldsymbol{\beta} + b_i$$

onde, μ_i é a esperança da média, $g(.)$ é uma função da média da variável chamada de função de ligação, η_i é o preditor linar para a média da variável na área i, β são parâmetros de regressão, $\boldsymbol{X}_i$ é a linha i da matriz $n \times p$ com o vetor $\underline{1}$ e as $p-1$ covariáveis e b_i é o efeito aleatório da área i. Na forma vetorial temos

$$\boldsymbol{\mu} = E(\boldsymbol{y}) = g(\boldsymbol{\eta}), \text{ onde, } \boldsymbol{\eta} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b} \ .$$

A priori CAR é usada para modelar $\boldsymbol{b}$, o vetor de efeito aleatório.

O modelo CAR para o vetor $\boldsymbol{b}$, é definido por autoregressões de b_i em seus vizinhos. Assim,

$$b_i|\boldsymbol{b}_{-i} \sim N\left(\sum_{j\sim i}\frac{\rho b_j}{d_i}, \frac{1}{d_i\tau_b}\right) \ ,$$

em que $\boldsymbol{b}_{-i}$ são todos os elementos de $\boldsymbol{b}$ exceto o elemento b_i, $j \sim i$ indica que a área j é vizinha da área i, d_i é o número de vizinhos da área i, ρ é um parâmetro de autocorrelação e mede a força da dependência de b_i nos seus vizinhos e $\tau_b > 0$ é o parâmetro de precisão. Esse modelo faz com que $\boldsymbol{b}$ tenha uma variação suave no espaço.

É possível obter a distribuição conjunta de $\boldsymbol{b}$, [4]. Assumindo que E(b_i)=0, a distribuição de $\boldsymbol{b}$ é normal multivariada

$$\boldsymbol{b} \sim MVN(0, (\boldsymbol{I} - \rho\boldsymbol{W})^{-1}\boldsymbol{T}^{-1}) \ ,$$

em que $\boldsymbol{T}$ é uma matriz diagonal com $\boldsymbol{T}_{ii} = \tau_b d_i$e $\boldsymbol{W}$ é definida a partir da matriz de adjacência. A matriz de adjacência $\boldsymbol{A} = (a_{ij})$ de dimensão $n \times n$ é definida fazendo $a_{ij} = 1$ se $i \sim j$ e $a_{ij} = 0$ caso contrário ($a_{ii} = 0$). A matriz $\boldsymbol{W} = (w_{ij})$ é tal

que $w_{ij} = a_{ij}/a_{i.}$ onde $a_{i.} = \sum_j a_{ij} = d_i$ é o número de áreas vizinhas da área i. É necessário fazer $\rho \in (\lambda_n^{-1}, \lambda_1^{-1})$ em que $\lambda_1, ..., \lambda_n$ são os autovalores de $\boldsymbol{W}$ ordenados em ordem decrescente, ou $\rho \in (\lambda_n^{-1}, 1)$, pois $\lambda_1 = 1$.

Nós vamos considerar que, $y_1, ..., y_n$ são observações feitas nas áreas $1, ..., n$ de um processo gaussiano condicionalmente independente com média μ_i e variância τ_y^{-1}. Neste caso temos que o vetor n dimensional $\boldsymbol{y}|\boldsymbol{b}, \tau_y \sim MVN(\boldsymbol{b}, \frac{1}{\tau_y}\boldsymbol{I})$. Com $\boldsymbol{y}$ gaussiano, $g(.)$ costuma ser a identidade e $E(y_i) = \mu_i = \eta_i = \boldsymbol{X}_i\boldsymbol{\beta}$, pois $E(b_i) = 0$.

Esse modelo foi estendido em várias direções e para obter uma intuição da estrutura a posteriori desses modelos, nós vamos considerar $\boldsymbol{y}$ gaussiano e estudar detalhadamente sua distribuição à posteriori. Como veremos, muito pode ser aprendido sobre as consequências da adoção de uma distribuição a priori CAR para o efeito aleatório $\boldsymbol{b}$ neste caso. No caso de assumir $\boldsymbol{y}$ gaussiano, podemos encontrar distribuições conhecidas para a distribuição a posteriori. Porém, para outras distribuições, tal como a de Poisson, mesmo nos casos mais simples a distribuição a posteriori não possui forma conhecida e os métodos MCMC são utilizados para fazer inferência.

7.1.2 Distribuição à posteriori

Nós queremos obter a distribuição a posteriori de $\boldsymbol{b}$ considerando uma distribuição a priori CAR para $\boldsymbol{b}$:

$$\boldsymbol{b}|\tau_b, \rho \propto exp\left\{-\frac{1}{2}\boldsymbol{b}'[\boldsymbol{T}(\boldsymbol{I} - \rho\boldsymbol{W})]\boldsymbol{b}\right\} .$$

com ρ, $\boldsymbol{T}$ e $\boldsymbol{W}$ como definidos anteriormente. Para $\boldsymbol{\beta}$ nós adotamos uma distribuição a priori Normal p variada:

$$\boldsymbol{\beta} \propto exp\left\{-\frac{1}{2}(\boldsymbol{\beta}-)\boldsymbol{V}^{-1}(\boldsymbol{\beta}-)\right\} ,$$

com o vetor de médias e $\boldsymbol{V}$ a matriz de covariância. Para τ_y nós consideramos uma distribuição a priori $Gamma(\alpha, \beta)$, $\alpha > 0$ e $\beta > 0$:

$$\tau_y|\alpha, \beta \propto \tau_y^{\alpha-1}exp(-\tau_y\beta) .$$

A verossimilhança para os dados é o produto de distribuições normais:

$$y \propto \tau_y^{-n/2}exp\left\{-\frac{\tau_y}{2}[\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b})]'[\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b})]\right\} .$$

Assumindo independência entre as prioris, temos então que

$$\begin{aligned}\boldsymbol{\beta}, \tau_y, \boldsymbol{b}|\boldsymbol{y}, \tau_b, \boldsymbol{,} \boldsymbol{V}, \alpha, \beta \quad \propto \quad & exp\left\{-\frac{1}{2}(\boldsymbol{\beta}-)\boldsymbol{V}^{-1}(\boldsymbol{\beta}-)\right\} \times \tau_y^{\alpha-1}exp(-\tau_y\beta) \\ & \times exp\left\{-\frac{1}{2}\boldsymbol{b}'[\boldsymbol{T}(\boldsymbol{I} - \rho\boldsymbol{W})]\boldsymbol{b}\right\} \\ & \times \tau_y^{n/2}exp\left\{-\frac{\tau_y}{2}(\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b}))'(\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b}))\right\} .\end{aligned}$$

Podemos obter as distribuições condicionais para cada um dos parâmetros ($\boldsymbol{\beta}$, τ_y e $\boldsymbol{b}$).

Temos que $\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{b}, \tau_y, , \boldsymbol{V}$

$$\begin{aligned}
&\propto \; exp\left\{-\frac{1}{2}[(\boldsymbol{\beta}-)^{'}\boldsymbol{V}^{-1}(\boldsymbol{\beta}-) + \tau_y(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))]\right\} \\
&\propto \; exp\left\{-\frac{1}{2}[\boldsymbol{\beta}^{'}\boldsymbol{V}^{-1}\boldsymbol{\beta} - 2\boldsymbol{\beta}^{'}\boldsymbol{V}^{-1} - 2\tau_y(\boldsymbol{y}-\boldsymbol{b})^{'}(\boldsymbol{X}\boldsymbol{\beta}) + \tau_y(\boldsymbol{X}\boldsymbol{\beta})^{'}(\boldsymbol{X}\boldsymbol{\beta})]\right\} \\
&= \; exp\left\{-\frac{1}{2}[\boldsymbol{\beta}^{'}\boldsymbol{V}^{-1}\boldsymbol{\beta} - 2^{'}\boldsymbol{V}^{-1}\boldsymbol{\beta} - 2\tau_y[\boldsymbol{X}^{'}(\boldsymbol{y}-\boldsymbol{b})]^{'}\boldsymbol{\beta} + \tau_y\boldsymbol{\beta}^{'}(\boldsymbol{X}^{'}\boldsymbol{X})\boldsymbol{\beta}]\right\} \\
&= \; exp\left\{-\frac{1}{2}[\boldsymbol{\beta}^{'}(\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})\boldsymbol{\beta} - 2[^{'}\boldsymbol{V}^{-1} + \tau_y(\boldsymbol{X}^{'}(\boldsymbol{y}-\boldsymbol{b}))^{'}]\boldsymbol{\beta}]\right\}
\end{aligned}$$

Vamos considerar um resultado de algebra linear, a *multivariate completion of squares* ou *ellipsoidal retification* onde

$$\boldsymbol{u}^{'}\boldsymbol{A}\boldsymbol{u} - 2\boldsymbol{\alpha}^{'}\boldsymbol{u} = (\boldsymbol{u}-\boldsymbol{A}^{-1}\boldsymbol{\alpha})^{'}\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{A}^{-1}\boldsymbol{\alpha}) - \boldsymbol{\alpha}^{'}\boldsymbol{A}^{-1}\boldsymbol{\alpha}\ .$$

Portanto

$$\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{b}, \tau_y, , \boldsymbol{V} \propto exp\left\{-\frac{1}{2}[(\boldsymbol{\beta}-^{*})^{'}(\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})(\boldsymbol{\beta}-^{*})\right\}$$

onde $^{*} = (\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})^{-1}[\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}(\boldsymbol{y}-\boldsymbol{b})]$ e $\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{b}, \tau_y, , \boldsymbol{V}$ tem distribuição normal-p variada com média * e variância $(\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})^{-1}$.

A distribuição condicional de τ_y é obtida de

$$\tau_y|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_b, \rho \propto \tau_y^{n/2+\alpha-1} exp\left\{-\tau_y[\beta + \frac{1}{2}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))]\right\}\ ,$$

ou seja, que $\tau_y|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{b}, \alpha, \beta$ tem distribuição $Gamma(\alpha + n/2,\ \beta + \frac{1}{2}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b})))$.

A distribuição a posteriori de $\boldsymbol{b}$ condicionada aos demais parâmetros é obtida de

$$\begin{aligned}
\boldsymbol{b}|\boldsymbol{y}, \boldsymbol{\beta}, \tau_y, \tau_b, \rho \;&\propto\; exp\left\{-\frac{1}{2}[\boldsymbol{b}^{'}[\boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W})]\boldsymbol{b} + \tau_y(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))\right\} \\
&\propto\; exp\left\{-\frac{1}{2}[\boldsymbol{b}^{'}[\boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W})]\boldsymbol{b} - 2\tau_y(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})^{'}\boldsymbol{b} + \boldsymbol{b}^{'}(\tau_y\boldsymbol{I})\boldsymbol{b}]\right\} \\
&=\; exp\left\{-\frac{1}{2}[\boldsymbol{b}^{'}[\boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W}) + \tau_y\boldsymbol{I}]\boldsymbol{b} - 2\tau_y(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})^{'}\boldsymbol{b}]\right\}
\end{aligned}$$

Completando quadrados e considerando $\boldsymbol{C} = [\tau_y\boldsymbol{I} + \boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W})]$ temos

$$\boldsymbol{b}|\boldsymbol{y}, \boldsymbol{\beta}, \tau_y, \tau_b, \rho \propto exp\left\{-\frac{1}{2}[(\boldsymbol{b}-\tau_y\boldsymbol{C}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}))^{'}\boldsymbol{C}(\boldsymbol{b}-\tau_y\boldsymbol{C}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}))]\right\}$$

Portanto, temos que

$$\boldsymbol{b}|\boldsymbol{y}, \boldsymbol{\beta}, \tau_y, \tau_b, \rho \sim MVN\left(\tau_y\boldsymbol{C}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}),\ \boldsymbol{C}^{-1}\right)\ . \tag{7.1}$$

Se consideramos o caso particular em que não temos covariáveis no modelo e $E(y_i) = 0$, nós podemos utilizar um resultado geral de inferência bayesiana para dados normais multivariados. Neste caso, condicionando a τ_b e ρ conhecidos, temos que a distribuição marginal de $\boldsymbol{\mu}$ a posteriori, integrando sobre τ_y, é t-multivariada com 2α graus de liberdade, vetor de locação $\boldsymbol{\mu}$ e matriz de precisão $(\alpha/\beta)[\boldsymbol{T}(\boldsymbol{I} - \rho\boldsymbol{W})]$, [13]. Neste caso particular, não é necessário utilizar MCMC para simular amostras da distribuição a posteriori de $\boldsymbol{b}$.

Na prática, é comum atribuir distribuições a priori para os hiperparâmetros , $\boldsymbol{V}$, τ_b e ρ, ou seja, utilizar hiperprioris para considerar a incerteza sobre esses hiperparâmetros. Porém, não é possível obter distribuições condicionais com forma conhecida para eles. Como temos condicionais completas para os demais parâmetros com forma conhecida, podemos utilizar o procedimento *Gibbs Sampler* com passos de Metropolis-Hastings para esses parâmetros.

7.1.3 Inferência

Geralmente o uso da distribuição CAR como priori requer o uso de técnicas de Monte Carlo via Cadeia de Markov, ou do inglês *Monte Carlo via Markov Chain* - MCMC, para obter amostras da distribuição à posteriori. Apenas em poucos casos particulares podemos encontrar distribuições a posteriori conhecidas a aproximação necessária. Atualmente vários modelos considerando o modelo CAR como distribuição à priori foram implementados e estão disponíveis em programas computacionais populares , tais como WinBUGS, [18], e BayesX, [17]. Devido à facilidade de simular amostras da distribuição a posteriori, os modelos espaciais bayesianos foram estendidos a modelos espaço-temporais, modelos multivariados e modelos de sobrevivência espaciais, [4]; modelos com parâmetros variando no espaço [2]; e modelos aditivos generalizados, [14].

Os métodos MCMC são bem fundamentados na teoria de Cadeias de Markov e fornecem bons resultados na obtenção de aproximações para as distribuições à posteriori. Recentemente, tem sido proposto o uso de aproximações analíticas como alternativa à aproximação via MCMC. A mais recente e largamente difundida é a combinação de duas aproximações itegradas de Laplace aninhadas, ou *Integrated Nested Laplace Aproximations* - **INLA**, proposta por [24]. Esta técnica foi desenvolvida para aplicação em modelos de regressão com efeitos aleatórios gaussianos. Assim, podemos aplicá-la a vários modelos espaciais e espaço temporais.

7.2 Regressão com priori CAR

A abordagem bayesiana de análise de dados georeferenciados por áreas tem sido a mais utilizada, principalmente devido ao fato de ser mais facilmente estendida a modelos mais complexos do que a abordagem frequentista. Nessa abordagem, a componente espacial é geralmente considerada como um efeito aleatório cuja distribuição a priori autogregressiva condicional ou CAR, do inglês ***C**onditional **A**uto**R**egressive*, para os efeitos aleatórios espaciais desconhecidos.

7.2.1 Introdução

Nesta seção nós vamos considerar que temos observações y_i, $i = 1, ..., n$, feitas em n áreas geográficas. Podemos considerar também um vetor de covariáveis $\boldsymbol{X}_i$ observadas em cada área. Um modelo que pode ser considerado para esses dados é o modelo linear generalizado misto, ou *Generalized Linear Mixed Model* - GLMM. Esse modelo é utilizado para modelar a esperança da variável de interesse de forma linear através de uma função de ligação:

$$\mu_i = E(y_i) = g(\eta_i), \text{ onde, } \eta_i = \boldsymbol{X}_i\boldsymbol{\beta} + b_i$$

onde, μ_i é a esperança da média, $g(.)$ é uma função da média da variável chamada de função de ligação, η_i é o preditor linar para a média da variável na área i, β são parâmetros de regressão, $\boldsymbol{X}_i$ é a linha i da matriz $n \times p$ com o vetor $\underline{1}$ e as $p-1$ covariáveis e b_i é o efeito aleatório da área i. Na forma vetorial temos

$$\boldsymbol{\mu} = E(\boldsymbol{y}) = g(\boldsymbol{\eta}), \text{ onde, } \boldsymbol{\eta} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b}\ .$$

A priori CAR é usada para modelar $\boldsymbol{b}$, o vetor de efeito aleatório.

O modelo CAR para o vetor $\boldsymbol{b}$, é definido por autoregressões de b_i em seus vizinhos. Assim,

$$b_i|\boldsymbol{b}_{-i} \sim N\left(\sum_{j\sim i}\frac{\rho b_j}{d_i}, \frac{1}{d_i\tau_b}\right)\ ,$$

em que $\boldsymbol{b}_{-i}$ são todos os elementos de $\boldsymbol{b}$ exceto o elemento b_i, $j \sim i$ indica que a área j é vizinha da área i, d_i é o número de vizinhos da área i, ρ é um parâmetro de autocorrelação e mede a força da dependência de b_i nos seus vizinhos e $\tau_b > 0$ é o parâmetro de precisão. Esse modelo faz com que $\boldsymbol{b}$ tenha uma variação suave no espaço.

É possível obter a distribuição conjunta de $\boldsymbol{b}$, [4]. Assumindo que E(b_i)=0, a distribuição de $\boldsymbol{b}$ é normal multivariada

$$\boldsymbol{b} \sim MVN(0, (\boldsymbol{I} - \rho\boldsymbol{W})^{-1}\boldsymbol{T}^{-1})\ ,$$

em que $\boldsymbol{T}$ é uma matriz diagonal com $\boldsymbol{T}_{ii} = \tau_b d_i$e $\boldsymbol{W}$ é definida a partir da matriz de adjacência. A matriz de adjacência $\boldsymbol{A} = (a_{ij})$ de dimensão $n \times n$ é definida fazendo $a_{ij} = 1$ se $i \sim j$ e $a_{ij} = 0$ caso contrário ($a_{ii} = 0$). A matriz $\boldsymbol{W} = (w_{ij})$ é tal que $w_{ij} = a_{ij}/a_{i.}$ onde $a_{i.} = \sum_j a_{ij} = d_i$ é o número de áreas vizinhas da área i. É

necessário fazer $\rho \in (\lambda_n^{-1}, \lambda_1^{-1})$ em que $\lambda_1, ..., \lambda_n$ são os autovalores de $\boldsymbol{W}$ ordenados em ordem decrescente, ou $\rho \in (\lambda_n^{-1}, 1)$, pois $\lambda_1 = 1$.

Nós vamos considerar que, $y_1, ..., y_n$ são observações feitas nas áreas $1, ..., n$ de um processo gaussiano condicionalmente independente com média μ_i e variância τ_y^{-1}. Neste caso temos que o vetor n dimensional $\boldsymbol{y}|\boldsymbol{b}, \tau_y \sim MVN(\boldsymbol{b}, \frac{1}{\tau_y}\boldsymbol{I})$. Com $\boldsymbol{y}$ gaussiano, $g(.)$ costuma ser a identidade e $E(y_i) = \mu_i = \eta_i = \boldsymbol{X}_i\boldsymbol{\beta}$, pois $E(b_i) = 0$.

Esse modelo foi estendido em várias direções e para obter uma intuição da estrutura a posteriori desses modelos, nós vamos considerar $\boldsymbol{y}$ gaussiano e estudar detalhadamente sua distribuição à posteriori. Como veremos, muito pode ser aprendido sobre as consequências da adoção de uma distribuição a priori CAR para o efeito aleatório $\boldsymbol{b}$ neste caso. No caso de assumir $\boldsymbol{y}$ gaussiano, podemos encontrar distribuições conhecidas para a distribuição a posteriori. Porém, para outras distribuições, tal como a de Poisson, mesmo nos casos mais simples a distribuição a posteriori não possui forma conhecida e os métodos MCMC são utilizados para fazer inferência.

7.2.2 Distribuição à posteriori

Nós queremos obter a distribuição a posteriori de $\boldsymbol{b}$ considerando uma distribuição a priori CAR para $\boldsymbol{b}$:

$$\boldsymbol{b}|\tau_b, \rho \propto exp\left\{-\frac{1}{2}\boldsymbol{b}'[\boldsymbol{T}(\boldsymbol{I} - \rho\boldsymbol{W})]\boldsymbol{b}\right\} .$$

com ρ, $\boldsymbol{T}$ e $\boldsymbol{W}$ como definidos anteriormente. Para $\boldsymbol{\beta}$ nós adotamos uma distribuição a priori Normal p variada:

$$\boldsymbol{\beta} \propto exp\left\{-\frac{1}{2}(\boldsymbol{\beta}-)\boldsymbol{V}^{-1}(\boldsymbol{\beta}-)\right\} ,$$

com o vetor de médias e $\boldsymbol{V}$ a matriz de covariância. Para τ_y nós consideramos uma distribuição a priori $Gamma(\alpha, \beta)$, $\alpha > 0$ e $\beta > 0$:

$$\tau_y|\alpha, \beta \propto \tau_y^{\alpha-1}exp(-\tau_y\beta) .$$

A verossimilhança para os dados é o produto de distribuições normais:

$$y \propto \tau_y^{-n/2}exp\left\{-\frac{\tau_y}{2}[\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b})]'[\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b})]\right\} .$$

Assumindo independência entre as prioris, temos então que

$$\begin{aligned}\boldsymbol{\beta}, \tau_y, \boldsymbol{b}|\boldsymbol{y}, \tau_b, , \boldsymbol{V}, \alpha, \beta \quad \propto \quad & exp\left\{-\frac{1}{2}(\boldsymbol{\beta}-)\boldsymbol{V}^{-1}(\boldsymbol{\beta}-)\right\} \times \tau_y^{\alpha-1}exp(-\tau_y\beta) \\ & \times exp\left\{-\frac{1}{2}\boldsymbol{b}'[\boldsymbol{T}(\boldsymbol{I} - \rho\boldsymbol{W})]\boldsymbol{b}\right\} \\ & \times \tau_y^{n/2}exp\left\{-\frac{\tau_y}{2}(\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b}))'(\boldsymbol{y} - (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{b}))\right\} .\end{aligned}$$

Podemos obter as distribuições condicionais para cada um dos parâmetros ($\boldsymbol{\beta}$, τ_y e $\boldsymbol{b}$).

Temos que $\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{b}, \tau_y, , \boldsymbol{V}$

$$\begin{aligned}
&\propto \; exp\left\{-\frac{1}{2}[(\boldsymbol{\beta}-)^{'}\boldsymbol{V}^{-1}(\boldsymbol{\beta}-) + \tau_y(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))]\right\} \\
&\propto \; exp\left\{-\frac{1}{2}[\boldsymbol{\beta}^{'}\boldsymbol{V}^{-1}\boldsymbol{\beta} - 2\boldsymbol{\beta}^{'}\boldsymbol{V}^{-1} - 2\tau_y(\boldsymbol{y}-\boldsymbol{b})^{'}(\boldsymbol{X}\boldsymbol{\beta}) + \tau_y(\boldsymbol{X}\boldsymbol{\beta})^{'}(\boldsymbol{X}\boldsymbol{\beta})]\right\} \\
&= \; exp\left\{-\frac{1}{2}[\boldsymbol{\beta}^{'}\boldsymbol{V}^{-1}\boldsymbol{\beta} - 2^{'}\boldsymbol{V}^{-1}\boldsymbol{\beta} - 2\tau_y[\boldsymbol{X}^{'}(\boldsymbol{y}-\boldsymbol{b})]^{'}\boldsymbol{\beta} + \tau_y\boldsymbol{\beta}^{'}(\boldsymbol{X}^{'}\boldsymbol{X})\boldsymbol{\beta}]\right\} \\
&= \; exp\left\{-\frac{1}{2}[\boldsymbol{\beta}^{'}(\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})\boldsymbol{\beta} - 2[^{'}\boldsymbol{V}^{-1} + \tau_y(\boldsymbol{X}^{'}(\boldsymbol{y}-\boldsymbol{b}))^{'}]\boldsymbol{\beta}]\right\}
\end{aligned}$$

Vamos considerar um resultado de algebra linear, a *multivariate completion of squares* ou *ellipsoidal retification* onde

$$\boldsymbol{u}^{'}\boldsymbol{A}\boldsymbol{u} - 2\boldsymbol{\alpha}^{'}\boldsymbol{u} = (\boldsymbol{u}-\boldsymbol{A}^{-1}\boldsymbol{\alpha})^{'}\boldsymbol{A}(\boldsymbol{u}-\boldsymbol{A}^{-1}\boldsymbol{\alpha}) - \boldsymbol{\alpha}^{'}\boldsymbol{A}^{-1}\boldsymbol{\alpha}\ .$$

Portanto

$$\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{b}, \tau_y, , \boldsymbol{V} \propto exp\left\{-\frac{1}{2}[(\boldsymbol{\beta}-^{*})^{'}(\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})(\boldsymbol{\beta}-^{*})\right\}$$

onde $^{*} = (\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})^{-1}[\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}(\boldsymbol{y}-\boldsymbol{b})]$ e $\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{b}, \tau_y, , \boldsymbol{V}$ tem distribuição normal-p variada com média * e variância $(\boldsymbol{V}^{-1}+\tau_y\boldsymbol{X}^{'}\boldsymbol{X})^{-1}$.

A distribuição condicional de τ_y é obtida de

$$\tau_y|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{b}, \tau_b, \rho \propto \tau_y^{n/2+\alpha-1} exp\left\{-\tau_y[\beta + \frac{1}{2}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))]\right\}\ ,$$

ou seja, que $\tau_y|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{b}, \alpha, \beta$ tem distribuição $Gamma(\alpha + n/2,\ \beta + \frac{1}{2}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b})))$.

A distribuição a posteriori de $\boldsymbol{b}$ condicionada aos demais parâmetros é obtida de

$$\begin{aligned}
\boldsymbol{b}|\boldsymbol{y}, \boldsymbol{\beta}, \tau_y, \tau_b, \rho \;&\propto\; exp\left\{-\frac{1}{2}[\boldsymbol{b}^{'}[\boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W})]\boldsymbol{b} + \tau_y(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))^{'}(\boldsymbol{y}-(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{b}))\right\} \\
&\propto\; exp\left\{-\frac{1}{2}[\boldsymbol{b}^{'}[\boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W})]\boldsymbol{b} - 2\tau_y(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})^{'}\boldsymbol{b} + \boldsymbol{b}^{'}(\tau_y\boldsymbol{I})\boldsymbol{b}]\right\} \\
&=\; exp\left\{-\frac{1}{2}[\boldsymbol{b}^{'}[\boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W}) + \tau_y\boldsymbol{I}]\boldsymbol{b} - 2\tau_y(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta})^{'}\boldsymbol{b}]\right\}
\end{aligned}$$

Completando quadrados e considerando $\boldsymbol{C} = [\tau_y\boldsymbol{I} + \boldsymbol{T}(\boldsymbol{I}-\rho\boldsymbol{W})]$ temos

$$\boldsymbol{b}|\boldsymbol{y}, \boldsymbol{\beta}, \tau_y, \tau_b, \rho \propto exp\left\{-\frac{1}{2}[(\boldsymbol{b}-\tau_y\boldsymbol{C}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}))^{'}\boldsymbol{C}(\boldsymbol{b}-\tau_y\boldsymbol{C}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}))]\right\}$$

Portanto, temos que

$$\boldsymbol{b}|\boldsymbol{y}, \boldsymbol{\beta}, \tau_y, \tau_b, \rho \sim MVN\left(\tau_y\boldsymbol{C}^{-1}(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}),\ \boldsymbol{C}^{-1}\right)\ . \tag{7.2}$$

Se consideramos o caso particular em que não temos covariáveis no modelo e $E(y_i) = 0$, nós podemos utilizar um resultado geral de inferência bayesiana para dados normais multivariados. Neste caso, condicionando a τ_b e ρ conhecidos, temos que a distribuição marginal de $\boldsymbol{\mu}$ a posteriori, integrando sobre τ_y, é t-multivariada com 2α graus de liberdade, vetor de locação $\boldsymbol{\mu}$ e matriz de precisão $(\alpha/\beta)[\boldsymbol{T}(\boldsymbol{I} - \rho\boldsymbol{W})]$, [13]. Neste caso particular, não é necessário utilizar MCMC para simular amostras da distribuição a posteriori de $\boldsymbol{b}$.

Na prática, é comum atribuir distribuições a priori para os hiperparâmetros , $\boldsymbol{V}$, τ_b e ρ, ou seja, utilizar hiperprioris para considerar a incerteza sobre esses hiperparâmetros. Porém, não é possível obter distribuições condicionais com forma conhecida para eles. Como temos condicionais completas para os demais parâmetros com forma conhecida, podemos utilizar o procedimento *Gibbs Sampler* com passos de Metropolis-Hastings para esses parâmetros.

7.2.3 Inferência

Geralmente o uso da distribuição CAR como priori requer o uso de técnicas de Monte Carlo via Cadeia de Markov, ou do inglês *Monte Carlo via Markov Chain* - MCMC, para obter amostras da distribuição à posteriori. Apenas em poucos casos particulares podemos encontrar distribuições a posteriori conhecidas a aproximação necessária. Atualmente vários modelos considerando o modelo CAR como distribuição à priori foram implementados e estão disponíveis em programas computacionais populares , tais como WinBUGS, [18], e BayesX, [17]. Devido à facilidade de simular amostras da distribuição a posteriori, os modelos espaciais bayesianos foram estendidos a modelos espaço-temporais, modelos multivariados e modelos de sobrevivência espaciais, [4]; modelos com parâmetros variando no espaço [2]; e modelos aditivos generalizados, [14].

Os métodos MCMC são bem fundamentados na teoria de Cadeias de Markov e fornecem bons resultados na obtenção de aproximações para as distribuições à posteriori. Recentemente, tem sido proposto o uso de aproximações analíticas como alternativa à aproximação via MCMC. A mais recente e largamente difundida é a combinação de duas aproximações itegradas de Laplace aninhadas, ou *Integrated Nested Laplace Aproximations* - **INLA**, proposta por [24]. Esta técnica foi desenvolvida para aplicação em modelos de regressão com efeitos aleatórios gaussianos. Assim, podemos aplicá-la a vários modelos espaciais e espaço temporais.

7.3 Regressão dinâmica via INLA

A inferência bayesiana ganhou um enorme salto de usabilidade a partir da facilidade de recursos computacionais disponibilizados para implementar MCMC. Porém, em modelos mais complexos, usar MCMC pode ser proibitivo do ponto de vista computacional. Isto porque crescendo a complexidade do modelo, geralmente cresce o número de parâmetros, crescendo também o número de distribuições a posteriori para buscar. Isso torna a convergência lenta e necessário muitas amostras da posteriori ou algoritmos específicos. Há casos em que demoram-se dias para obter uma amostra satisfatória de MCMC em modelos com alguma complexidade, e em especial os que estruturas de dependências cuja dimensionalidade das operações cresce com as observações tais como modelos espaciais e/ou temporais.

Diante dessas dificuldades, uma alternativa atrativa para uma grande classe de modelos é o uso de aproximações analíticas da distribuição a posteriori. O uso das aproximações de Laplace aninhadas e integradas, *Integrated Nested Laplace Aproximations* - INLA, proposto por [24], tem se provado útil, relativamente geral e operacional em uma diversidade de contextos. Essa difusão está se dando de forma rápida, motivada principalmente pela disponibilidade de software, na forma do pacote **INLA** [25] do R, para aplicação dessa técnica a uma grande variedade de modelos. No restante desta sessão vamos considerar que o pacote **INLA** está instalado[1] e carregado.

```
require(INLA)
```

A inferência via INLA pode ser extendida para modelos dinâmicos como demostrado em [26] e nos exemplos que acompanham o artigo. Estimar modelos de regressão dinâmica usando o INLA em R pode ser feito de forma bastante direta a partir das funções disponíveis no pacote **INLA** com um arranjo adequado da estrutura de dados. Outros modelos dinâmicos, tais como modelos de crescimento, multivariados e espaço-temporais, também podem ser estimados via INLA [26].

A modelagem usando o pacote **INLA** envolve a especificação de modelos na classe de modelos lineares generalizados e de sobrevida com efeitos mistos. Os modelos podem conter efeitos fixos e efeitos aleatórios gaussianos. Além disso, pode-se especificar efeitos aleatórios gaussianos para os coeficientes das covariáveis de tal forma que temos os coeficientes variando. É exatamente esta possibilidade que permite a estimação modelos de regressão dinâmica usando o **INLA** .

Dentre os modelos gaussianos para efeitos aleatórios estruturados no tempo, há o autoregressivo de ordem 1, o passeio aleatório de ordem 1 e o passeio aleatório de ordem 2. Assim, basta especificar um destes para os coeficientes de regressão de um modelo de regressão para séries temporais e teremos um modelo de regressão dinâmica.

Um efeito aleatório é declarado no **INLA** usando a função `f()` deste pacote. Nesta função é necessário informar um vetor de índices com valores variando na dimensão do efeito aleatório. Para os dados da seção anterior, vamos declarar um

[1] disponível para *download* em www.r-inla.org

vetor de índices para o intercepto e para cada covariável. As covariáveis entram como pesos multiplicando as estimativas do efeito aleatório. Esse efeito aleatório pode ou não ter restrição de soma zero. No nosso caso não vamos ter essa restrição.

A seguir simulamos um conjunto de dados ($n = 100$) segundo um modelo dinâmico como definido em **??**.

Para fazer comparação com os resultados através do algorítmo de MCMC, da Seção **??**, vamos considerar as mesmas distribuições a priori para ψ. Para utilizar o INLA definimos a estrutura de dados em um `data-frame` que inclui variáveis `ID` identificadoras das unidades às quais serão atribuídos efeitos aleatórios. Defimos também uma lista com informações sobre os parâmetros e finalmente utilizamos a função `inla()` declarando o modelo desejado por uma `formula` em Rna qual os termos considerados aleatórios são definidos dentro de `f()` sempre utilizando a variável indicadora das unidades.

```
dad <- data.frame(y=y, i0=1:n, i1=1:n,
                  i2=1:n, x1=x[2,], x2=x[2,])
hyp <- list(theta1=list(param=c(.5,.1), initial=0),
            theta2=list(param=c(.5,.3)))
mod <- inla(y ~ 0 + f(i0, model="ar1", constr=FALSE, hyper=hyp) +
         f(i1, x1, model="ar1", constr=FALSE, hyper=hyp) +
         f(i2, x2, model="ar1", constr=FALSE, hyper=hyp), data=dad,
         control.data=list(hyper=list(theta=list(initial=0,
                                          param=c(.5,.1)))))
```

O objeto `mod` guarda um sumário e a distribuição marginal a posteriori de cada efeito aleatório, θ_t, e dos parâmetros ψ.

Na Figura 7.1 nós visualizamos a densidade a posteriori de $1/V$, $diag\{G\}$, $1/diag\{W\}$, considerando as 2000 amostras simuladas por MCMC (curvas em linhas contínuas) e as densidades marginais a posteriori obtidas via INLA (curvas tracejadas). A linha pontilhada na vertical corresponde ao valor verdadeiro usado para simular os dados. Também, visualizamos a região de 95% credibilidade para cada valor de θ_t considerando as 2000 amostras simuladas por MCMC (área em cinza) e o intervalo de 95% de credibilidade obtido via INLA (linhas tracejadas). A linha vertical é o valor verdadeiro de θ.

Observamos na Figura 7.1 que os resultados são satisfatórios embora apresentem diferenças para alguns parâmetros. Por exemplo, para $diag\{W\}[1]$ temos que a distribuição a posteriori obtida via INLA é mais concentrada que a obtida via MCMC. No caso de G, também são observadas algumas diferenças. Neste caso, nós consideramos um algoritmo MCMC com possibilidade de simular valores menores que -1 e maiores que 1. Já considerando a abordagem via INLA, o modelo considerado a priori para θ é autoregressivo de ordem 1, fazendo com que os elementos de $diag\{G\}$ sejam restritos ao intervalo $(0, 1)$. Portanto neste caso a diferença parece estar no modelo e não no algoritmo de inferência.

7.4 Modelo dinâmico espaço temporal via INLA

Nesta seção vamos considerar que temos dados observados em diferentes localizações geográficas e em diferentes momentos no tempo, ou seja, temos um dado espaço-temporal. Na dimensão temporal é mais comum se ter dados em variação temporal discreta, por exemplo, dados diários, mensais, anuais, etc. Na dimensão espacial podemos ter dados em alguns pontos de uma região geográfica ou dados de áreas geográficas (setores censitários, municípios, estados, etc). Nesta última situação, as regiões são fixas, assim como os tempos. Na situação de dados observados em alguns pontos, pode-se ter os pontos fixados, por exemplo observações climáticas feitas em estações meteorológicas, ou ter a localização dos pontos aleatória, por exemplo a localização da residência de pessoas acometidas de uma certa doença. Para cada uma dessas três situações, os modelos para os termos espaciais são diferentes. [**?**] apresentam e discutem os fundamentos e resultados centrais da especificação condicional de modelos espaciais.

7.4.1 Um modelo para mortalidade infantil

Nesta seção, vamos considerar um formato de dados bastante comum na área de mapeamento de doenças. Nessa área, é comum a análise de dados do número de casos de uma doença agrupados em região geográfica, como, por exemplo, um município. Desta forma, vamos considerar um modelo de variação espacial discreta. Além disso, vamos considerar, neste caso, uma distribuição de Poisson para a resposta.

Vamos analisar dados de mortalidade infantil nos 37 municípios da mesoregião de Curitiba. A partir do site do DATASUS, conseguimos dados de nascidos vivos de 1994 a 2009 e número de óbitos infantis de 1996 a 2009. Também consideramos três possíveis covariáveis: cobertura de imunização total, cobertura de imunização oral contra tuberculose (BCG) e cobertura de imunização contra poliomelite. Para estras três covariáveis, há dados de cada município para o período de 1995 a 2011. Assim, vamos considerar os dados no período de 1995 a 2009. Consideramos os óbitos em 1995 e 2009 como dados faltantes e podemos estimá-los utilizando o modelo. Para o ano de 2009 vamos comparar o números estimados com os observados.

Seja $o_{i,t}$ e $n_{i,t}$ os números de óbitos infantis e de nascidos vivos, respectivamente, no município i no ano t. Temos $i = 1, 2, \ldots, N = 37$ e $t = 1, 2, \ldots, T = 15$. Sob a suposição de que a taxa de mortalidade infantil é igual para todos o municípios em todo o período, essa taxa única é estimada por

$$R = \frac{\sum_{t=1}^{T}\sum_{i=1}^{N} o_{i,t}}{\sum_{t=1}^{T}\sum_{i=1}^{N} n_{i,t}}.$$

Nessa hipótese, o número esperado de óbitos infantis é dado por $E_{i,t} = R * n_{i,t}$. Com isso vamos considerar o que

$$o_{i,t} \sim Poisson(\psi_{i,t} E_{i,t})$$

em que $\psi_{i,t}$ é o risco relativo do município i no ano t. Esse risco mede o quanto o $o_{i,t}$ está acima ou abaixo do número esperado $E_{i,t}$.

Também podemos considerar que a taxa em cada ano muda na mesma proporção para todos os municípios. Assim, podemos considerar

$$R_t = \frac{\sum_{i=1}^{N} o_{i,t}}{\sum_{i=1}^{N} n_{i,t}}.$$

Desta forma elimina-se qualquer tendência global existente ao longo do período.

Vamos considerar o seguinte modelo para $\psi_{i,t}$

$$\log(\psi_{i,t}) = \eta_{i,t} = \alpha_{i,t} + \beta_{i,t} X_{i,t}$$

em que $\eta_{i,t}$ é o preditor linear; $X_{i,t}$ é uma variável observada e varia em cada município e em cada ano; $\beta_{i,t}$ mede o efeito da covariável e varia para cada município e para cada ano, $\alpha_{i,t}$ é um intercepto aleatório variando para cada município e no tempo e captura a variação em $\psi_{i,t}$ não explicada pela covariável. Se $o_{i,t}$ é próximo de $E_{i,t}$, ou seja, o observado não difere muito do esperado, $\psi_{i,t} = 1$ e $\eta_i = 0$.

Considerando a classe de modelos proposta em [29] e [28], Temos que

$$\alpha_{i,t} = \phi_\alpha \alpha_{i,t-1} + w1_{i,t} \beta_{i,t} = \phi_\beta \beta_{i,t-1} + w2_{i,t}$$

em que $w1_{,t}$ e $w_{,t}$ são erros gaussianos multivariados com média zero e matriz de covariância com estrutura espacial. Para estimar esses modelos podemos considerar a abordagem proposta em [26].

Inicialmente vamos considerar um caso particular, quando $w1_{,t}$ e $w2_{,t}$ são ICAR, autoregressivo intrínsico, a priori [6]. Neste caso,

$$w1_{i,t} | w1_{-i,t} \sim N(\sum_{j \sim i} w1_{j,t}/d_i, \sigma^2_{w1}/d_i)$$

em que $w1_{-i,t}$ indica o vetor $w1_{,t}$ sem a área i, $j \sim i$ é o conjunto de áreas vizinhas da área i, d_i é o número de áreas vizinhas da área i e $\sigma^2_{w1} = \tau^{-1}_{w1}$ é o parâmetro de variância de α. Portanto, a distribuição condicional de $w1_{i,t}$ é Normal com média sendo a média de $w1$ nas áreas vizinhas e a variância inversamente proporcional ao número de áreas vizinhas.

A estimação de alguns modelos dinâmicos usando o pacote **INLA**, pode ser simplificada para alguns casos com recentes opções de sintaxe no pacote para definir efeitos aleatórios por grupos. Podemos definir um modelo espacial para cada tempo e considerar cada tempo como um grupo e propor um modelo autoregressivo de ordem 1 para grupos. [10] desenvolve uma aplicação desta abordagem em modelagem espaço-temporal. No nosso exemplo, suponha uma matriz Q_S de precisão para a componente espacial de $\alpha_{i,t}$, $w1_{,t}$. Considerando Q_T a matriz de estrutura da precisão de um processo univariado temporal autoregressivo de ordem 1, temos que a matriz de precisão de $w1$ é $Q = Q_S \otimes Q_T$. Na situação em que Q_S é a matriz de precisão resultante do modelo ICAR, podemos usar essa estratégia no pacote **INLA**.

Modelos sem covariáveis

Na modelagem com efeitos aleatórios espaciais para dados de áreas, precisamos definir a matriz (ou lista) de vizinhança entre as áreas. Assim, vamos inicialmente usar um mapa do Paraná dividido em municípios, obtido do site do IBGE. Utilizamos o pacote **spdep** [7] e suas dependência para importar e representar os mapas no R.

```
require(spdep)
pr.m <- readShapePoly("~/mapas/41mu2500gc")
```

Após importar o mapa para para o R, selecionamos a parte do mapa com apenas os municípios da mesoregião de Curitiba, criamos a lista de vizinhança e salvamos em arquivo para uso pelo **INLA**:

```
cwbm <- pr.m[pr.m$MESOREGIAO=="METROPOLITANA DE CURITIBA",]
nbm <- poly2nb(cwbm)
nb2INLA(file="dados/mesoc.g", nbm)
```

Para a análise usando o pacote **INLA**, empilhamos os dados de cada ano, isto é, as N primeiras linhas são dados do primeiro ano. Os dados possuem as seguinte colunas: `ano`, `código do município`, `número de nascidos vivos`, `número de óbitos infantis`, `imuno`, `bcg`, `polio`, `i` (um índice de município variando de 1 a 37) e `t` (um índice de tempo variando de 1 a 15). Os dados (atributos dos municípios) são lidos e verificados com os comandos a seguir.

```
micwb <- read.csv2("dados/micwb.csv", fileEnc="latin1")
head(micwb, 2) ## primeiras linhas
```

```
  ano    mun nasc obi imuno   bcg  polio i t
1 1995 411320  445  NA 17.65 39.11  39.00 1 1
2 1995 412010   58  NA 45.23 95.29 102.35 2 1
```

```
tail(micwb, 2) ## ultimas linhas
```

```
    ano    mun nasc obi  imuno    bcg  polio  i  t
554 2009 410950  104   1  92.19 109.01 117.12 36 15
555 2009 411995  278   3 100.72 124.58 125.00 37 15
```

Precisamos calcular o número esperado de óbitos. Vamos considerar uma taxa para cada ano. Assim, eliminamos a tendência temporal comum a todos os municípios e focamos a modelagem nas particularidades de cada município. Alguns municípios pequenos não tiveram nascidos vivos em alguns dos anos. Para calcular o número esperado, consideramos $n_{ij} = 0.5$ nesses casos. Para o ano de 1994, consideramos taxa de 1995.

```
ob.a <- tapply(micwb$obi, micwb$t, sum, na.rm=TRUE)
nc.a <- tapply(micwb$nasc, micwb$t, sum, na.rm=TRUE)
tx.a <- ob.a/nc.a
tx.a[1] <- tx.a[2]
micwb$nasc[micwb$nasc==0] <- 0.5
micwb$esperado <- micwb$nasc * rep(tx.a, each=length(nbm))
```

A SMR (da sigla em inglês para razão de mortalidade padronizada) observada pode ser calculada por

```
smro <- micwb$obi/micwb$esperado
```

Para avaliação, vamos definir uma coluna adicional nos dados com os óbitos até 2008 e colocando NA's para o ano 2009. Esta será a variável resposta na modelagem.

```
micwb$y <- micwb$obi
micwb$y[micwb$ano==2009] <- NA
```

Vamos considerar um conjunto de modelos de complexidade crescente para $\eta_{i,t}$:

$m_0:$	α_0
$m_1:$	$\alpha_0 + \alpha_{0,t}$
$m_2:$	$\alpha_0 + \alpha_{i,0}$
$m_3:$	$\alpha_0 + \alpha_{0,t} + \alpha_{i,0}$
$m_4:$	$\alpha_0 + \alpha_{i,t}$

em que:

- m_0: não temos modelagem do risco relativo;
- m_1: $\alpha_{0,t}$ é um processo autoregressivo no tempo, comum a todas as áreas;
- m_2: $\alpha_{i,0}$ é um processo autoregressivo no espaço, comum a todos os tempos;
- m_3: temos a soma de ambos os anteriores
- m_4: temos o modelo dinâmico para $\alpha_{i,t}$

Vamos criar uma lista de cinco fórmulas para especificar o conjunto de cinco modelos.

```
forms <- list(m0=y ~ 1,
              m1=y ~ 1 + f(i, model="ar1"),
              m2=y ~ 1 + f(i, model="besag", graph="dados/mesoc.g"),
              m3=y ~ 1 + f(t, model="ar1") +
              f(i, model="besag", graph="dados/mesoc.g"),
              m4=y ~ 1 + f(i, model="besag", graph="dados/mesoc.g",
                group=t, control.group=list(model="ar1",
                           hyper=list(theta=list(param=c(0,1))))))
```

e usar a função `inla()` para aproximar as distribuições a posteriori e o cálculo é feito em poucos segundos nos computadores atuais.

```
require(INLA)
mods <- lapply(forms, inla, family="Poisson",
               data=micwb, E=micwb$esperado,
               control.predictor=list(compute=TRUE),
               control.compute=list(dic=TRUE))
sapply(mods, function(x) x$cpu.used)
```

```
                        m0        m1        m2        m3         m4
Pre-processing  0.5307660 0.4286566 0.4650400 0.5396781  0.4900405
Running inla    1.3110819 2.7041285 1.8977787 6.9944375 24.4648252
Post-processing 0.3522794 0.4051874 0.4052265 0.4419107  0.9067664
Total           2.1941273 3.5379725 2.7680452 7.9760263 25.8616321
```

Podemos ver o valor do DIC para cada modelo e verificamos que o modelo dinâmico é o que apresentou o menor DIC.

```
sapply(mods, function(x) x$dic[[1]])
```

```
      m0       m1       m2       m3       m4
2951.815 2499.458 2495.070 2496.099 2470.631
```

É recomendável verificar o ajuste do modelo aos dados. Um diagnóstico simples é visualizar os dados observados e o valor estimado pelo modelo dado por $E_{i,t}\mathrm{e}\{\psi_{i,t}\}$. Vamos inicialmente organizar o risco relativo estimado pelo modelo 4 e o número de óbitos estimado.

```
smr4 <- matrix(exp(mods$m4$summary.linear.pred$m), 37)
est <- micwb$esperado*smr4
```

Também podemos comparar o comportamento da SMR observada e a SMR estimada pelo modelo em relação ao número de óbitos.

Na Figura 7.2 temos o diagrama de dispersão entre o número de casos e a SMR calculada pela razão entre o número observado de casos e o número esperado de casos, isto é, a SMR observada. Observamos neste gráfico uma grande variação na SMR quando o número de óbitos é pequeno. Esta variação na SMR decai a medida que o número de óbitos aumenta. Esse fato expressa bem a elevada variação em taxas de municípios pequenos, com poucos nascidos vivos e, consequentemente, poucos óbitos. No gráfico do meio, temos a dispersão entre o número observado de casos e a SMR estimada pelo modelo. Neste caso, notamos padrão parecido, porém, agora a SMR estimada não apresenta valores muito extremos como a observada. No gráfico da direita temos a dispersão entre o número de casos estimado pelo modelo e o número observado de casos. Neste gráfico, observamos que há um bom ajuste do modelo.

Na Figura 7.3 temos a densidade à posteriori para α_0, τ_{w1} e para ϕ_α, considerando o modelo m_4. A seguir, temos a média, desvio padrão e intervalo de 95% de credibilidade.

```
mods$m4$summary.fix[ ,c(1,2,3,5)]
```

```
      mean         sd 0.025quant 0.975quant
0.17372634 0.01705562 0.14022568 0.20713809
```

```
mods$m4$summary.hyp[ ,c(1,2,3,5)]
```

```
                        mean         sd 0.025quant 0.975quant
Precision for i 12.1344197 2.94144908  7.4410901 18.9158045
GroupRho for i   0.8934278 0.04029005  0.7968966  0.9525941
```

Os valores de α_0 estão em torno de 0.14 a 0.21. A evolução de α no tempo é bastante suave, com ϕ_α a posteriori em torno de 0.80 a 0.95. Com τ_α em torno de 7.4 a 18.9, $w1$ tem variância em torno de 0.05 a 0.2.

Temos a distribuição marginal a posteriori de cada $\alpha_{i,t}$. Podemos visualizar as séries para alguns municípios ou visualizar o mapa para alguns tempos. Também temos estatísticas resumo de $\psi_{i,t}$, o logaritmo do risco relativo, a posteriori.

Na Figura 7.4 podemos visualizar a SMR estimada pelo modelo 4. A linha mais larga é para o município de Curitiba. Espera-se que a média da SMR seja igual a 1. Como Curitiba tem mais nascidos vivos e apresenta SMR bem abaixo dos demais, a maioria dos municípios apresenta SMR maior que 1.

Na Figura 7.5 temos a SMR estimada pelo modelo 4 para os anos 1995, 2000, 2004 e 2009. Como havíamos visto, Curitiba tem o risco mais baixo. Embora o número de áreas seja pequeno, é perceptível uma suavidade na variação espacial do risco. Os municípios a leste (do litoral) apresentam um valor menor. E alguns municípios ao norte de Curitiba são os que apresentam os maiores valores.

```
q <- quantile(smr4, 0:5/5)
par(mfrow=c(2,2), mar=c(0,0,1,0))
for (i in c(1, 6, 10, 15)) {
  plot(cwbm, col=gray(1-(smr4[,1]-q[1])/(q[6]-q[1])))
  title(main=paste("Ano", 1995:2009)[i])
}
image(x=c(-50.3,-50.2), y=seq(-25.5,-24.5,leng=21),
      z=matrix(quantile(smr4, 1:20/21), 1),  col=gray(19:0/19),
      brea=quantile(smr4, 0:20/20), add=TRUE)
text(rep(-50.1,6), seq(-25.5, -24.5, leng=6), format(q,dig=2))
```

Os seis municípios que apresentam o maior risco médio estimado ao longo dos anos pelo modelo 4 podem ser extraídos.

```
    GEOCODIGO              NOME
376   4105201        Cerro Azul
391   4127882   Tunas do Paraná
375   4122206 Rio Branco do Sul
385   4103107   Bocaiúva do Sul
369   4128633    Doutor Ulysses
390   4100202      Adrianópolis
```

Agora, vamos considerar Curitiba e mais três municípios selecionados aleatoriamente. e visualizar as séries de dados observados para esses municípios, juntamente com o valor estimado pelo modelo. Vamos considerar também intervalo de 95% de credibilidade multiplicando $E_{i,t}$ pelo exponencial dos quantis 2,5% e 97,5% de $\psi_{i,t}$. As séries para esses quatro municípios são visualizadas na Figura 7.6.

```
 set.seed(123)
 (msel <- c("Curitiba", sample(setdiff(cwbm$NOME, "Curitiba"),3)))
```

```
[1] "Curitiba"          "Itaperuçu"         "Adrianópolis"
[4] "Rio Branco do Sul"
```

```
 li.e <- matrix(micwb$esperado*exp(mods$m4$summary.linear.pred[,3]), 37)
 ls.e <- matrix(micwb$esperado*exp(mods$m4$summary.linear.pred[,5]), 37)
```

Na Figura 7.6, notamos que, em geral, número de óbitos estimados está razoavelmente próximo dos observados. Observamos que para os municípios com número de casos menor, a discrepância entre o observado e o estimado é maior. Isso também pode ser visto no gráfico da direita da Figura 7.2.

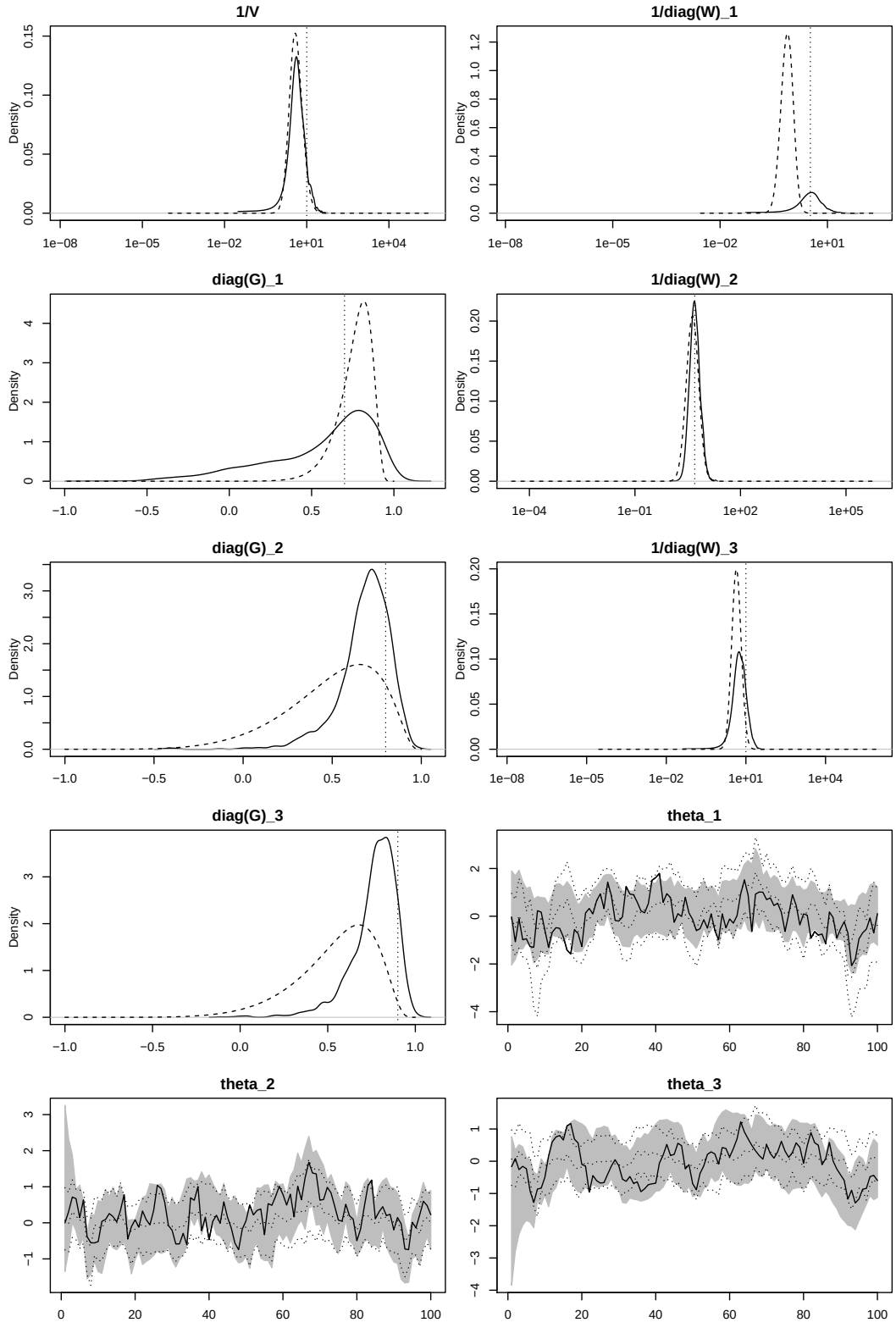


Figura 7.1: Gráficos comparativos dos resultados via MCMC e via INLA para o modelo de regressão dinâmica.

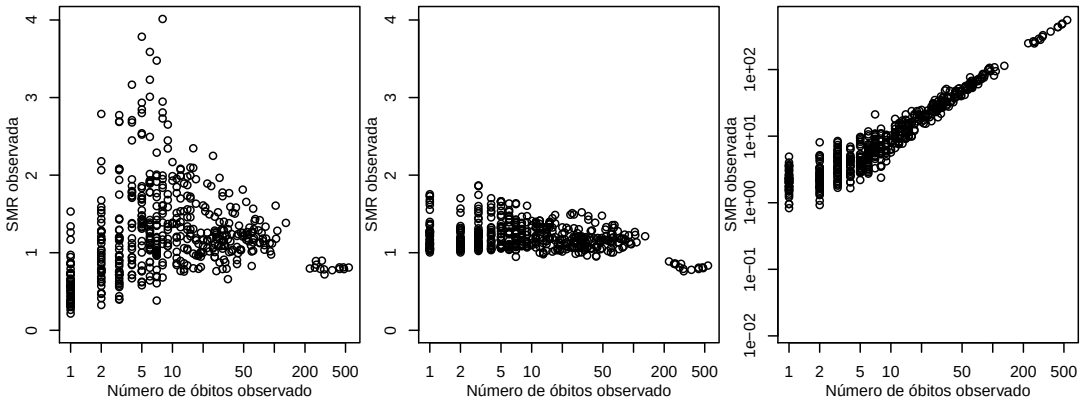


Figura 7.2: SMR observada pelo número de óbitos (esquerda), SMR estimada pelo número de óbitos (meio) e número estimado pelo número de óbitos observado.

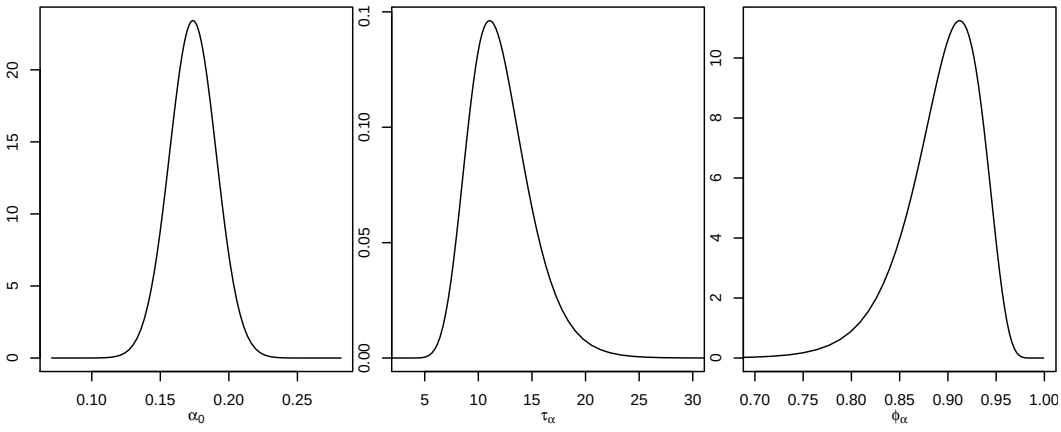


Figura 7.3: Densidade dos hiperparâmetros a posteriori.

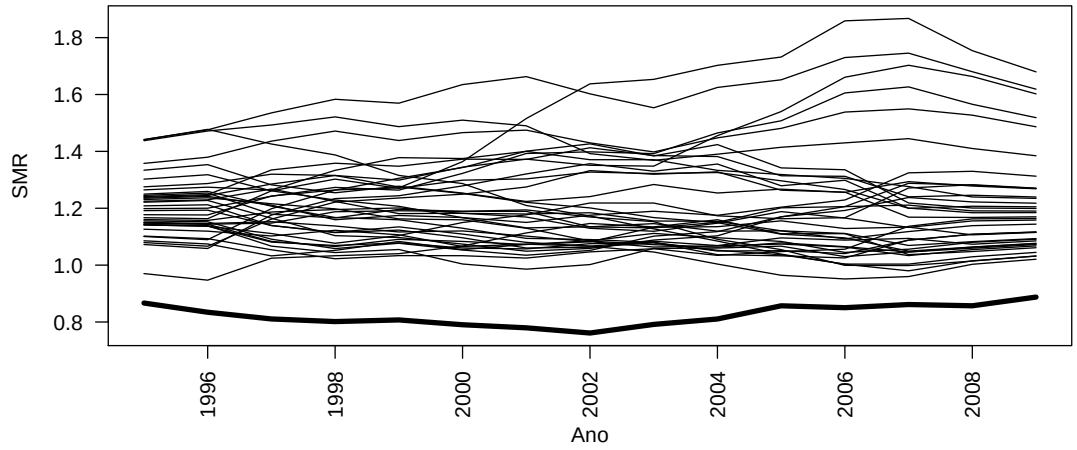


Figura 7.4: Séries temporais da SMR estimada pelo modelo 4. Linha grossa indica o município de Curitiba.

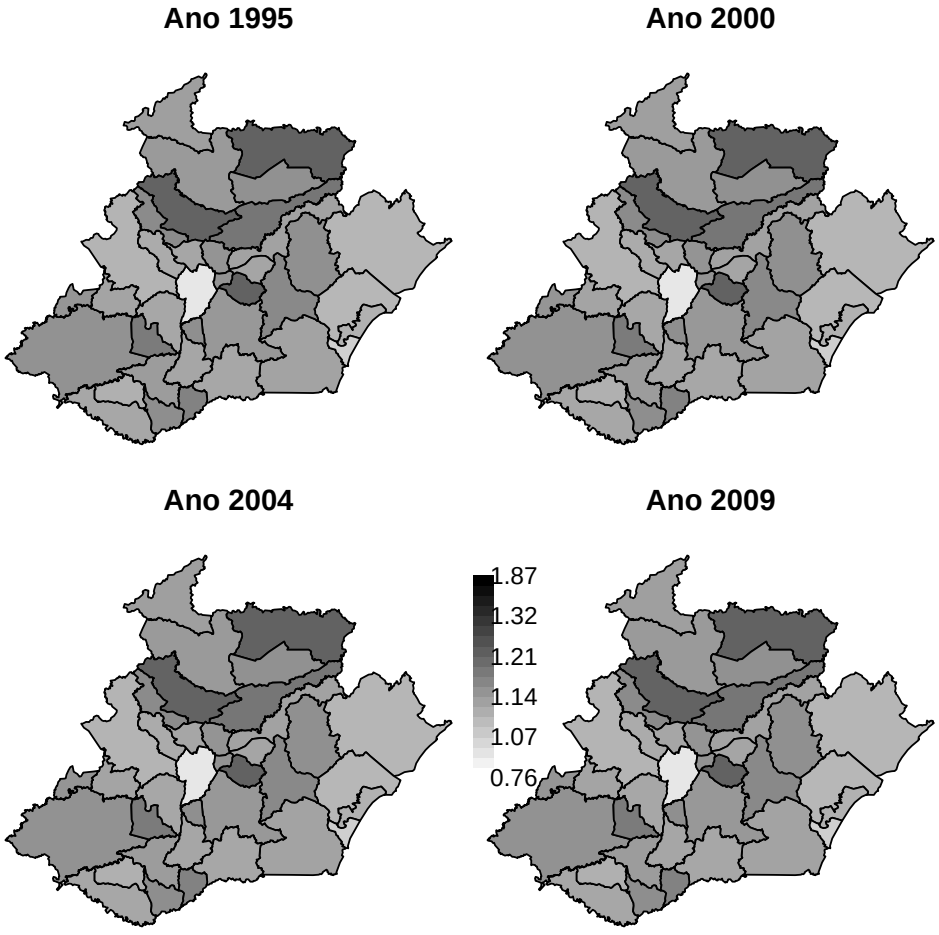


Figura 7.5: Mapas do risco relativo estimado pelo modelo 4 para alguns dos anos.

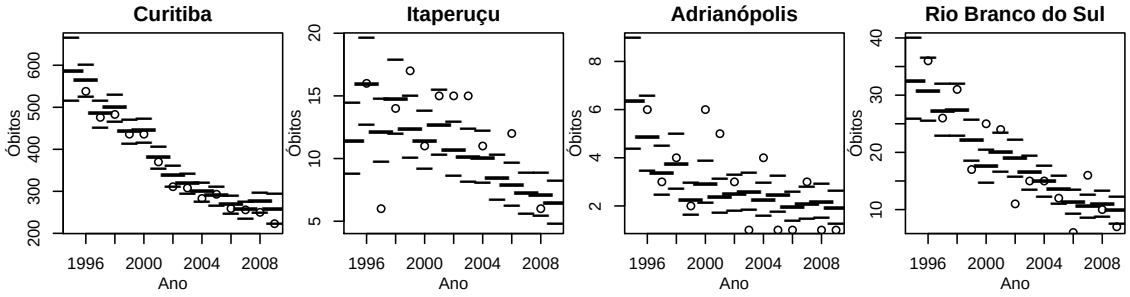


Figura 7.6: Séries de número de óbitos infantis para os quatro municípios, valor predito e intervalo de 95% de credibilidade.

Referências Bibliográficas

[1] L. Anselin, R. J. G. M Florax, and S. J Rey. *Advances in Spatial Econometrics: Methodology, Tools and Applications*. Springer, 2004.

[2] R. M. Assunção. Space varing coefficient models for small area data. *Environmetrics*, 14:453–473, 2003.

[3] R. M. Assunção and E. A. Reis. A new proposal to adjust moran's i for population density. *Statistics in Medicine*, (18):2147–2162, 1999.

[4] S. Banerjee, B. P. Carlin, and A. E. Gelfand. *Hierarchical Modeling and Analysis for Spatial Data*. CRC Press, 2004.

[5] J. Besag. Spatial interaction and the statistical analisys of lattice systems (with discussion). *Journal of the Royal Statistical Society, Series B*, 36:192–236, 1974.

[6] J. Besag and C. Kooperberg. On conditional and intrinsic autoregressions. *Biometrika*, 82(4):733–746, 1995.

[7] Roger Bivand, with contributions by Micah Altman, Luc Anselin, Renato Assunço, Olaf Berke, Andrew Bernat, Eric Blankmeyer, Marilia Carvalho, Yongwan Chun, Bjarke Christensen, Carsten Dormann, Stéhane Dray, Rein Halbersma, Elias Krainski, Nicholas Lewin-Koh, Hongfei Li, Jielai Ma, Giovanni Millo, Werner Mueller, Hisaji Ono, Pedro Peres-Neto, Gianfranco Piras, Markus Reder, Michael Tiefelsdorf, , and Danlin Yu. *spdep: Spatial dependence: weighting schemes, statistics and models*, 2010. R package version 0.5-10.

[8] G. Camara, R.C.M. Souza, U.M. Freitas, and J. Garrido. Spring: Integrating remote sensingand gis by object-oriented data modelling. *Computers & Graphics*, 20(3):395–403, May-Jun 1996.

[9] G. Câmara, L. Vinhas, K. R. Ferreira, G. R. Queiroz, R. C. M. Souza, A. M. V. Monteiro, M. T. Carvalho, and U. M. Casanova, M. A. Freitas. *Open Source Approaches for Spatial Data Handling*, chapter TerraLib: An Open Source GIS Library for Large-scale Environmental and Socio-economic Applications. Springer, 2007.

[10] M. Cameletti, F. Lindgren, D. Simpson, and H. Rue. Spatio-temporal modeling of particulate matter concentration through the spde approach. Technical report, Norwegian University of Science and Technology, 2011.

[11] Noel A. C. Cressie. *Statistics For Spatial Data*. Wiley, 1995. Revised Edition.

[12] DATASUS. *TabWin*. http://www.datasus.gov.br/tabwin.

[13] Morris H. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill Book Company, New York, 1970.

[14] L. Fahrmeir and S. Lang. Bayesian inference for generalized additive mixed models based on markov random field priors. *Applied Statistics - JRSS C*, 50:201–220, 2001.

[15] Timothy H. Keitt, Roger Bivand, Edzer Pebesma, and Barry Rowlingson. *rgdal: Bindings for the Geospatial Data Abstraction Library*, 2010. R package version 0.6-27.

[16] Milan Kilibarda. *plotGoogleMaps: Plot SP data as HTML map mashup over Google Maps*, 2012. R package version 1.3.

[17] Brezger Lang. BayesX - Software for Bayesian Inference Based on Markov Chain Monte Carlo Simulation Techniques, 2000.

[18] D.J Lunn, A. Thomas, N. Best, and D. Spiegelhalter. WinBUGS – a Bayesian Modelling Framework: Concepts, Structure, and Extensibility. *Statistics and Computing*, 10:325–337, 2000.

[19] C. Oliveira. *Curso de Cartografia Moderna*. Ed. IBGE, 1988.

[20] E.J. Pebesma and R.S Bivand. Classes and methods for spatial data in r. *R News*, 2(5), 2005. http://cran.r-project.org/doc/Rnews/.

[21] R Core Team. *foreign: Read Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, dBase, ...*, 2012. R package version 0.8-50.

[22] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[23] Virgilio Gomez-Rubio Roger S. Bivand, Edzer J. Pebesma. *Applied spatial data analysis with R*. Springer, New York, 2008. http://www.asdar-book.org/.

[24] H. Rue, S. Martino, and N. Chopin. Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *JRSS: Series B*, 71(2):319–392, 2009.

[25] Havard Rue, Sara Martino, and Finn Lindgren. *INLA: Functions which allow to perform a full Bayesian analysis of structured (geo-)additive models using Integrated Nested Laplace Approximaxion*, 2009. R package version 0.0.

[26] R. Ruiz-Cárdenas, E. T. Krainski, and H. Rue. Direct fitting of dynamic models using integrated nested laplace approximations - inla. *Computational Statistics and Data Analysis*, 56(6):1808–1828, 2012.

[27] L. Vinhas, K. R. Ferreira, G. R. de de Queiroz, M. da Motta, L. Hara, and Regina L de S. de Garrido, J. P. Bruno. *TerraView 3.1.4 Tutorial*. Instituto Nacional de Pesquisas Espaciais, 2008.

[28] J. C. Vivar. *Modelos espaÃ§o-temporais para dados de Ã¡rea na famÃlia exponencial*. PhD thesis, Universidade Federal do Rio de Janeiro, 2007.

[29] J. C. Vivar and M. A. R. Ferreira. Spatio-temporal models for gaussian areal data. *Journal of Computational and Graphical Statistics*, 18(3):658–674, 2009.

[30] P. Whittle. On stationary process in the plane. *Biometrika*, 41:434–449, 1954.