

Recuperando coordenadas geográficas de endereços públicos e locais com o uso do software R



**Empresa Brasileira de Pesquisa Agropecuária
Embrapa Monitoramento por Satélite
Ministério da Agricultura, Pecuária e Abastecimento**

DOCUMENTOS 114

Recuperando coordenadas geográficas de endereços públicos e locais com o uso do software R

Fernando Antônio de Pádua Paim

Exemplares desta publicação podem ser adquiridos na:

Embrapa Monitoramento por Satélite

Av. Soldado Passarinho, nº 303
Fazenda Jardim Chapadão
13070-115, Campinas, SP
Fone: (19) 3211.6200
www.embrapa.br/territorial
www.embrapa.br/fale-conosco/sac

Comitê Local de Publicações
da Embrapa Monitoramento por Satélite

Presidente
Sérgio Gomes Tosto

Secretário-Executivo
Bibiana Teixeira de Almeida

Membros
André Luiz dos Santos Furtado, Bibiana Teixeira de Almeida, Carlos Fernando Quartaroli, Daniela Maciel Pinto, Fabio Enrique Torresan, Gustavo Bayma Siqueira da Silva, Janice Freitas Leivas, Marcelo Fernando Fonseca, Vera Viana dos Santos Brandão

Supervisão editorial
Suzi Carneiro e Bibiana Teixeira de Almeida

Revisão de texto
Bibiana Teixeira de Almeida

Normalização bibliográfica
Vera Viana dos Santos Brandão

Projeto gráfico da coleção
Carlos Eduardo Felice Barbeiro

Editoração eletrônica e Tratamento das ilustrações
Suzi Carneiro

Ilustração da capa
Suzi Carneiro

1ª edição
1ª impressão (2017): versão on-line

Todos os direitos reservados.

A reprodução não autorizada desta publicação, no todo ou em parte, constitui violação dos direitos autorais (Lei nº 9.610).

Dados Internacionais de Catalogação na Publicação (CIP)
Embrapa Monitoramento por Satélite

Paim, Fernando Antônio de Pádua.

Recuperando coordenadas geográficas de endereços públicos e locais com o uso do software R / Fernando Antônio de Pádua Paim. - Campinas: Embrapa Monitoramento por Satélite, 2017. 24 p. : il. ; (Documentos / Embrapa Monitoramento por Satélite, ISSN 0103-7811; 114).

1. Informação georreferenciada. 2. Programa de computador. 3. Sistema de informação geográfica. I. Título. II. Série.

CDD 006

© Embrapa, 2017

Autor

Fernando Antônio de Pádua Paim

Analista de Sistemas, Especialista em Análise de Sistemas, analista da Embrapa Monitoramento por Satélite, Campinas, SP

Sumário

Apresentação	9
A família ggplot2 e o pacote ggmap	10
A função geocode.....	10
Os repositórios Google Maps e Data Science Toolkit	10
Os dados de exemplo.....	11
A função gsub (substituição de caracteres).....	13
A função iconv (conversão de caracteres)	13
A função get_map (uso de uma imagem real).....	16
A função ggmap.....	18
A função readShapePoly	20
Literatura consultada	21

Apresentação

A localização de qualquer posição no planeta é dada em função de dois números: a latitude e a longitude. Conhecidos como coordenadas geográficas, eles têm sua origem na perspectiva do planeta dividido em fatias de circunferências horizontais e verticais, relativamente ao seu eixo de rotação. A imagem mental resultante é a do planeta envolto em uma rede. As circunferências horizontais são conhecidas como paralelos e dão origem à latitude, e as circunferências verticais, conhecidas como meridianos, dão origem às longitudes. O cruzamento entre um paralelo e um meridiano define a localização de uma posição, isto é, as suas coordenadas geográficas, sobre a superfície do planeta. O Google Maps utiliza as coordenadas geográficas (latitude e longitude) na indicação de localizações no planeta. A localização de qualquer entidade sobre a superfície do planeta é hoje uma informação operacional e estratégica. Os mais diversos problemas, análises e soluções são nela baseados. O Google Maps disponibiliza acesso a esses dados por meio de uma interface de programação de aplicativo (API, do inglês *Application Programming Interface*) própria que pode ser utilizada pelo software R.

O presente documento ilustra como utilizar o software R para recuperar as informações de localização dadas em coordenadas geográficas a partir de endereços locais e como utilizar esses dados para plotar um gráfico georreferenciado. Para isso, é usado o pacote `ggmap`, da família `ggplot2`, o qual contém funcionalidades para essa atividade.

A família ggplot2 e o pacote ggmap

Um pacote é carregado no ambiente do R com a instrução `library (nome-do-pacote)`. Assim, `library(ggplot2)` e `library(ggmap)` executam essa função. Após carregar as bibliotecas `ggplot2` e `ggmap`, utiliza-se a função `geocode`, que recebe uma cadeia de caracteres no parâmetro `location` e devolve as coordenadas na forma de longitude e latitude. A sintaxe básica da função é `geocode(location)`.

Para a Prefeitura Municipal de Campinas, por exemplo, foi usada a seguinte sintaxe:

```
geocode(location="Av. Anchieta, 200 - Cambui, Campinas - SP")
```

A resposta do R foi:

```
      lon      lat
1 -47.05733 -22.90094
```

O endereço informado no parâmetro `location` não pode conter caracteres acentuados ou cedilha. São válidos apenas os caracteres especiais do teclado em inglês (norte-americano).

A função geocode

A função `geocode` faz com que o R faça uma chamada para a API do Google Maps (<https://developers.google.com/maps/?hl=pt-br>), para recuperar as coordenadas geográficas, como é mostrado na linha abaixo, que é parte integrante da resposta do R.

Information from URL:

```
http://maps.googleapis.com/maps/api/geocode/json?address=Av.%20Anchieta,%20200%20-%20Cambui,%20Campinas%20%96%20SP&sensor=false
```

O endereço da Prefeitura Municipal de Campinas foi obtido digitando-se “Prefeitura Municipal de Campinas” no Google.

Uma sintaxe mais completa da função `geocode` é:

```
geocode(location, source)
```

`Source` refere-se ao local utilizado para resgatar as coordenadas geográficas.

Os repositórios Google Maps e Data Science Toolkit

Atualmente há dois lugares disponíveis para uso como `source`: o Google Maps e o Data Science Toolkit (<http://www.datasciencetoolkit.org/about>). Ao utilizar a API do Google Maps, o usuário declara estar de acordo com os termos do serviço¹.

O uso da API do Google Maps tem limitação do número de consultas. Limites são comuns em APIs de acesso gratuito, e existem planos pagos que resolvem essas limitações.

¹ Descritos no documento <https://developers.google.com/maps/terms>.

Para usar o Data Science Toolkit como fonte da informação:

```
lo = geocode(
  location="Av. Anchieta, 200 - Cambui, Campinas - SP",
  source = "dsk")

lo
  lon lat
1 -49 -22
>
```

Os dados de exemplo

O arquivo `hospitais_campinas.txt`², disponível em http://repositorio.cnpm.embrapa.br/ftp/fernando/hospitais_campinas.txt, contém vários endereços e apresenta o seguinte formato: duas colunas de dados separadas por um ponto-e-vírgula. Não existem cabeçalhos para as colunas. Este é um exemplo de uma listagem com o conteúdo do arquivo:

```
Hospital Vera Cruz; Av. Andrade Neves, 402
Hospital Santa Sofia; Av. Andrade Neves, 611
Campinas Day Hospital; Av. Benjamin Constant, 1991
Casa de Saúde Campinas; Praça Dr. Toffoli, 28
Hospital Samaritano Campinas; Rua Eng. Monlevade, 206
Hospital e Maternidade Santa Tereza; Rua Bernardino de Campos, 804
Hospital Municipal Dr.Mario Gatti; Av. Pref. Faria Lima, 340
Centro Clínico Vera Cruz; Rua Gonçalves César, 158
Hospital Renascença Campinas; Av. Barão de Itapura, 1444
Hospital Madre Theodora; Rua José Geraldo Cerebino Christófaró, 175
Hospital do Coração de Campinas; Av. Benjamin Constant, 1725
Unimed Campinas; Av. Barão de Itapura, 1123
Maternidade de Campinas; Rua Barão de Parnaíba, 690
Saúde Ocupacional; Av. Barão de Itapura, 3371
San Francisco Hospital; Rua Eng. Artur Canguçu, 455
Hospital Vera Cruz; Av. Andrade Neves, 402
```

Para plotá-los em um gráfico georreferenciado, deve-se fazer o download do arquivo e lê-lo usando a seguinte instrução:

```
wf =
"http://repositorio.cnpm.embrapa.br/ftp/fernando/hospitais_campinas.txt"
hosp = read.table(file=wf, header=FALSE, sep=";")
```

² Os endereços dos hospitais de Campinas foram obtidos na web e colocados em arquivo de texto para uso nesta publicação.

É exibida uma listagem para simples conferência:

> hosp

	V1	V2
1	Hospital Vera Cruz	Av. Andrade
Neves, 402		
2	Hospital Santa Sofia	Av. Andrade
Neves, 611		
3	Campinas Day Hospital	Av. Benjamin
Constant, 1991		
4	Casa de Saã°de Campinas	Praãa Dr.
Toffoli, 28		
5	Hospital Samaritano Campinas	Rua Eng.
Monlevade, 206		
6	Hospital e Maternidade Santa Tereza	Rua Bernardino de
Campos, 804		
7	Hospital Municipal Dr.Mario Gatti	Av. Pref. Faria
Lima, 340		
8	Centro Clãnico Vera Cruz	Rua Gonãsalves
Cãsar, 158		
9	Hospital Renascenãa Campinas	Av. Barão de
Itapura, 1444		
10	Hospital Madre Theodora	Rua Josã Geraldo Cerebino
Christãfaro, 175		
11	Hospital do Coraãão de Campinas	Av. Benjamin
Constant, 1725		
12	Unimed Campinas	Av. Barão de
Itapura, 1123		
13	Maternidade de Campinas	Rua Barão de
Parnaãba, 690		
14	Saã°de Ocupacional	Av. Barão de
Itapura, 3371		
15	San Francisco Hospital	Rua Eng. Artur
Canguãu, 455		
16	Hospital Vera Cruz	Av. Andrade
Neves, 402		

>

Fazendo-se uma alteraão conveniente nos nomes das colunas, tem-se:

```
names(hosp) = c("nome", "endereço")
```

A função `gsub` (substituição de caracteres)

Como há caracteres estranhos nos nomes dos hospitais e nos endereços, é necessário fazer a mudança desses caracteres, principalmente nos endereços. A função `gsub` faz esse trabalho:

```
hosp$nome = gsub(pattern="Ã°", replacement="u", x=hosp$nome)
hosp$nome = gsub(pattern="Ã", replacement="i", x=hosp$nome)
hosp$nome = gsub(pattern="Ã§", replacement="c", x=hosp$nome)
hosp$nome = gsub(pattern="Ã©", replacement="e", x=hosp$nome)
hosp$nome = gsub(pattern="Ã£", replacement="a", x=hosp$nome)
```

```
hosp$endereço = gsub(pattern="Ã§", replacement="c", x=hosp$endereço)
hosp$endereço = gsub(pattern="Ã©", replacement="e", x=hosp$endereço)
hosp$endereço = gsub(pattern="Ã£", replacement="a", x=hosp$endereço)
hosp$endereço = gsub(pattern="Ã", replacement="i", x=hosp$endereço)
hosp$endereço = gsub(pattern="Ã³", replacement="o", x=hosp$endereço)
```

Também é possível fazer essa alteração utilizando dois vetores, `ce` (caractere estranho) e `cn` (caractere normal):

```
ce = c( "Ã£", "Ã©", "Ã", "Ã³", "Ã°", "Ã§" )
```

```
cn = c( "a", "e", "i", "o", "u", "c" )
```

Quando a ordem esperada dos parâmetros (`pattern`, `replacement`, `x`) é conhecida, a função `gsub` permite economizar na digitação usando o seguinte código:

```
for (i in 1:6) {
  hosp$nome = gsub(ce[i], cn[i], hosp$nome)
  hosp$endereço = gsub(ce[i], cn[i], hosp$endereço)
}
```

A função `iconv` (conversão de caracteres)

O R disponibiliza recursos adicionais para processar o texto e deixá-lo da forma adequada aos propósitos aos quais se destina. Por exemplo, a função `hosp$endereço=iconv(hosp$endereço,to="ASCII//TRANSLIT")` remove todos os acentos e a função `hosp$endereço=gsub("[[:punct:]]", "", hosp$endereço)` retira todos os caracteres especiais da variável `hosp$endereço`.

Mais informações sobre essas funções são obtidas por meio da função `help(iconv)` no *prompt* do R.

Após ajustar os caracteres que aparecem nos endereços, no caso específico do exemplo usado aqui, deve-se acrescentar a eles a expressão “ - Campinas Sao Paulo”:

```
hosp$endereco = paste(hosp$endereco, "Campinas, Sao Paulo")
```

Uma listagem para simples conferência mostra que os dados do endereço estão prontos para serem submetidos à função `geocode`:

```
> hosp
      nome                                     endereco
1      Hospital Vera Cruz                      Av. Andrade
Neves, 402 Campinas, Sao Paulo
2      Hospital Santa Sofia                    Av. Andrade
Neves, 611 Campinas, Sao Paulo
3      Campinas Day Hospital                  Av. Benjamin Constant,
1991 Campinas, Sao Paulo
4      Casa de Saude Campinas                  Praca Dr.
Toffoli, 28 Campinas, Sao Paulo
5      Hospital Samaritano Campinas           Rua Eng. Monlevade,
206 Campinas, Sao Paulo
6      Hospital e Maternidade Santa Tereza    Rua Bernardino de
Campos, 804 Campinas, Sao Paulo
7      Hospital Municipal Dr.Mario Gatti      Av. Pref. Faria
Lima, 340 Campinas, Sao Paulo
8      Centro Clinico Vera Cruz               Rua Goncalves
Cesar, 158 Campinas, Sao Paulo
9      Hospital Renascenca Campinas          Av. Barao de Itapura,
1444 Campinas, Sao Paulo
10     Hospital Madre Theodora Rua Jose Geraldo Cerebino Christofaro,
175 Campinas, Sao Paulo
11     Hospital do Coracao de Campinas        Av. Benjamin Constant,
1725 Campinas, Sao Paulo
12     Unimed Campinas                       Av. Barao de Itapura,
1123 Campinas, Sao Paulo
13     Maternidade de Campinas               Rua Barao de Parnaiba,
690 Campinas, Sao Paulo
14     Saude Ocupacional                     Av. Barao de Itapura,
3371 Campinas, Sao Paulo
15     San Francisco Hospital                Rua Eng. Artur
Cangucu, 455 Campinas, Sao Paulo
16     Hospital Vera Cruz                    Av. Andrade
Neves, 402 Campinas, Sao Paulo
>
```

Para aplicar à pesquisa a função `geocode` utilizando o Data Science Toolkit como fonte da informação:

```
longlat = geocode(location=hosp$endereco, source="dsk")
```

O resultado obtido foi este:

```
longlat
      lon      lat
1 -47.06083 -22.90556
2 -47.06083 -22.90556
3 -47.06083 -22.90556
4 -47.06083 -22.90556
5 -47.06083 -22.90556
6 -47.06083 -22.90556
7 -47.06083 -22.90556
8 -47.06083 -22.90556
9 -47.06083 -22.90556
10 -47.06083 -22.90556
11 -47.06083 -22.90556
12 -47.06083 -22.90556
13 -47.06083 -22.90556
14 -47.06083 -22.90556
15 -47.06083 -22.90556
16 -47.06083 -22.90556
>
```

Observa-se que os dados de longitude (lon) e latitude (lat) são iguais, não houve variações, embora sejam localizações distintas. Foi feita uma tentativa usando o Google:

```
longlat = geocode(location=hosp$endereco, source="google")
```

O resultado obtido foi:

```
> longlat
      lon      lat
1 -47.06863 -22.90489
2 -47.06991 -22.90333
3 -47.05321 -22.90021
4 -47.05482 -22.90821
5 -47.05994 -22.91755
6 -47.06218 -22.90415
7 -47.06821 -22.91683
8 -47.06729 -22.88821
9 -47.06400 -22.89263
10 -47.05151 -22.84048
11 -47.05553 -22.90113
12 -47.06601 -22.89427
13 -47.07160 -22.90177
14 -47.05305 -22.87791
15 -47.07489 -22.89537
16 -47.06863 -22.90489
>
```

O uso do Google como fonte das localizações produziu um conjunto de dados (longitudes e latitudes) mais preciso.

A Figura 1 mostra o resultado da plotagem de um gráfico simples para visualização usando a função:

```
plot(longlat,main="Hospitais em Campinas").
```

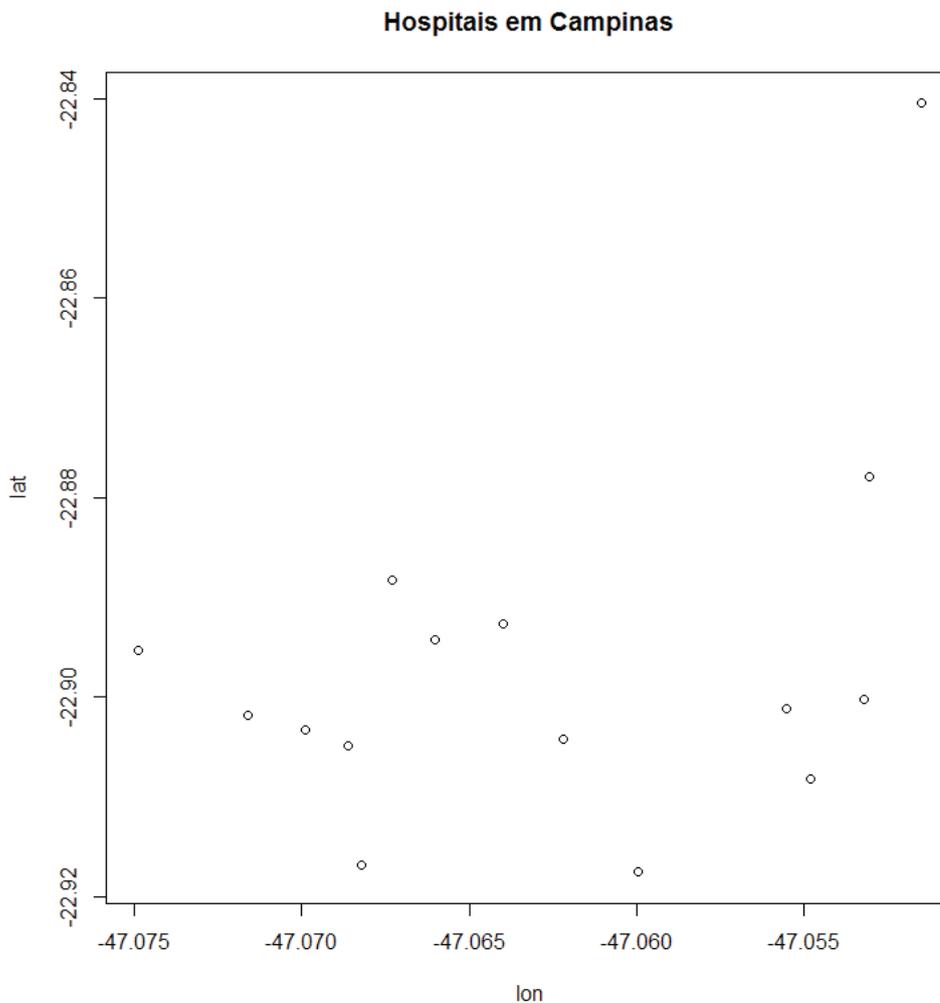


Figura 1. Hospitais em Campinas, SP.

A função `get_map` (uso de uma imagem real)

A função `get_map` do pacote `ggmap` acessa repositórios públicos de mapas como Google Maps, OpenStreetMap, Statmen Maps e Naver Map. Ela copia a imagem geocodificada da região desejada, caso exista, e retorna como um objeto do R. Mais informações são obtidas por meio da função `help(get_map)` no *prompt* do R.

Na chamada abaixo, foi utilizado o parâmetro `location` com o valor `Campinas`, `zoom`, que é um número inteiro no intervalo [3,21] e que estabelece uma referência à proximidade desejada. O valor 3 refere-se à dimensão continental e o valor 21, às construções. Foram usados três valores para o teste – 11, 12 e 13 –, e as imagens resultantes podem ser vistas nas Figuras 2, 3 e 4.

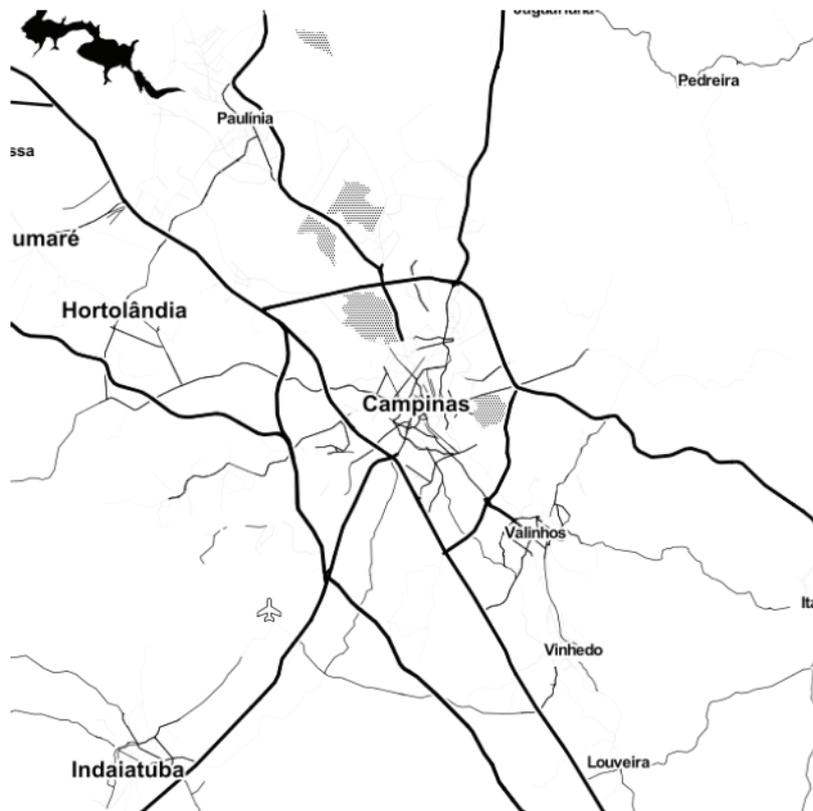


Figura 2. Função `get_map` com `zoom = 11`.



Figura 3. Função `get_map` com `zoom = 12`.



Figura 4. Função `get_map` com `zoom = 13`.

O parâmetro `source` indica de qual repositório a imagem é resgatada, o parâmetro `maptype` define um tema para a imagem. Para o Google Maps, as opções são `terrain`, `terrain-background`, `satellite`, `roadmap` e `hybrid`. Para o Stamen Maps, as opções são `terrain`, `watercolor` e `toner`. O parâmetro `color` define uma cor, a especificação `color` refere-se a preto e branco. Mais informações sobre esses recursos são obtidas por meio da função `help(get_cloudmademap)` no *prompt* do R.

```
campmap = get_map(
  location = "Campinas",
  zoom = 13,
  source = "stamen",
  maptype = "toner",
  color = "color")
```

As imagens resgatadas usando `plot(campmap)` foram plotadas para cada zoom utilizado.

A função `ggmap`

A função `ggmap` foi usada para agregar os dados disponíveis na variável `longlat` com a imagem disponível na variável `campmap`. O parâmetro `extent` define como a agregação entre a imagem e as coordenadas geográficas, definidas em `longlat`, ocorrerá. As opções são `normal`, `device` e `panel`. A opção `device` utiliza o tamanho da janela gráfica. A variável `campmap` é do tipo `ggmap` e `raster`, informação que é obtida com `class(campmap)`. A variável `longlat` é do tipo `"data.frame"`.

Mais informações sobre esses recursos são obtidas por meio da função `help(ggmap)` no *prompt* do R.

```
campmap = ggmap(  
  ggmap = campmap,  
  base_layer = ggplot(aes(x=lon, y=lat), data = longlat),  
  extent = "device")
```

Finalmente são plotados os resultados dos zooms iguais a 12 e 13, mostrados nas Figuras 5 e 6, usando a função:

```
campmap + geom_point(size = I(5), colour="red")
```



Figura 5. Mapa com zoom igual a 12.

Para melhor visualização, a distribuição dos hospitais foi plotada em pontos vermelhos e a prefeitura municipal, em azul, em um *shapefile*. As seguintes bibliotecas são necessárias para o acesso às funcionalidades para arquivos do tipo *shape*:

```
library(sp)  
library(rgeos)  
library(maptools)
```

O *shapefile* do Município de Campinas está disponível no endereço `< wzip = http://repositorio.cnpm.embrapa.br/ftp/fernando/campinas.zip >`.



Figura 6. Mapa com zoom igual a 13.

A função `download.file`, como o nome sugere, é utilizada para fazer o download de arquivos. Para baixar o arquivo compactado (`campinas.zip`) que contém o mapa (`.shp`, `.shx` e `.dbf`), é utilizada a função:

```
download.file(url=wzip, destfile="campinas.zip")
```

Para extrair os arquivos do pacote `.zip`, é utilizada a função `unzip("campinas.zip")`.

A função `readShapePoly`

A função `readShapePoly` do pacote `MapTools` é utilizada para ler um arquivo do tipo `shape`. O arquivo `.shp` faz referência implícita aos arquivos `.shx` e `.dbf` durante a leitura:

```
mshape = readShapePoly(fn="campinas.shp")
plot(mshape)
title("Município de Campinas SP\nHospitais")
points(longlat, col="red")
points(x=-47.05733, y=-22.90094, col="blue", pch=20)
```

Os resultados obtidos pelo procedimento exemplificado nesta publicação revelaram uma concentração dos hospitais na área central do Município de Campinas, SP (Figura 7).

Município de Campinas SP Hospitais

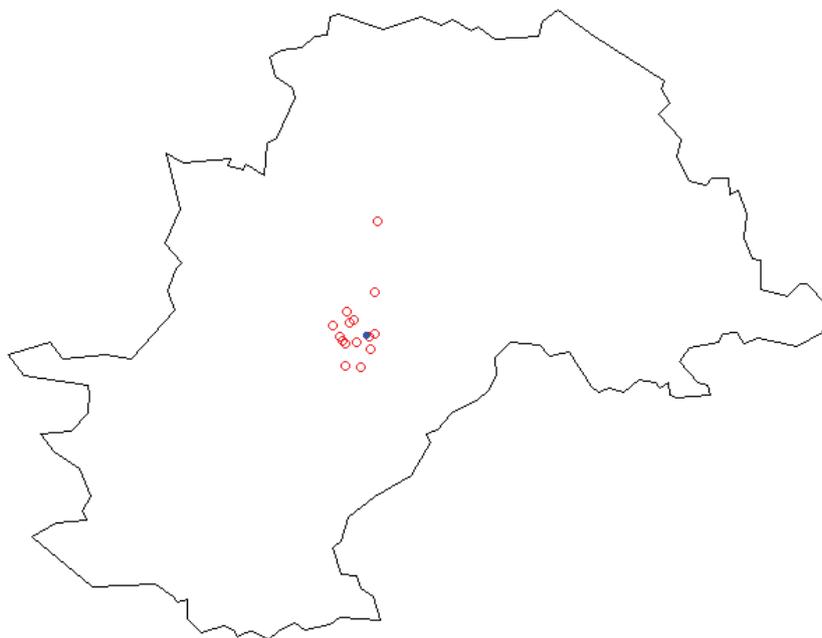


Figura 7. Localização dos hospitais plotados no mapa do Município de Campinas, SP.

Literatura consultada

R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.

Arquivos locais (na máquina) de ajuda (help) do R

<http://127.0.0.1:10531/library/base/html/grep.html>

<http://127.0.0.1:10531/library/base/html/iconv.html>

<http://127.0.0.1:10531/library/ggmap/html/geocode.html>

http://127.0.0.1:10531/library/ggmap/html/get_map.html

<http://127.0.0.1:10531/library/ggmap/html/ggmap.html>

<http://127.0.0.1:10531/library/mapproj/html/readShapePoly.html>

<http://127.0.0.1:10531/library/utils/html/read.table.html>

<http://127.0.0.1:10531/library/utils/html/unzip.html>

<http://127.0.0.1:14249/library/utils/html/download.file.html>



Monitoramento por Satélite