

問題解決技法入門

2. Graph Theory

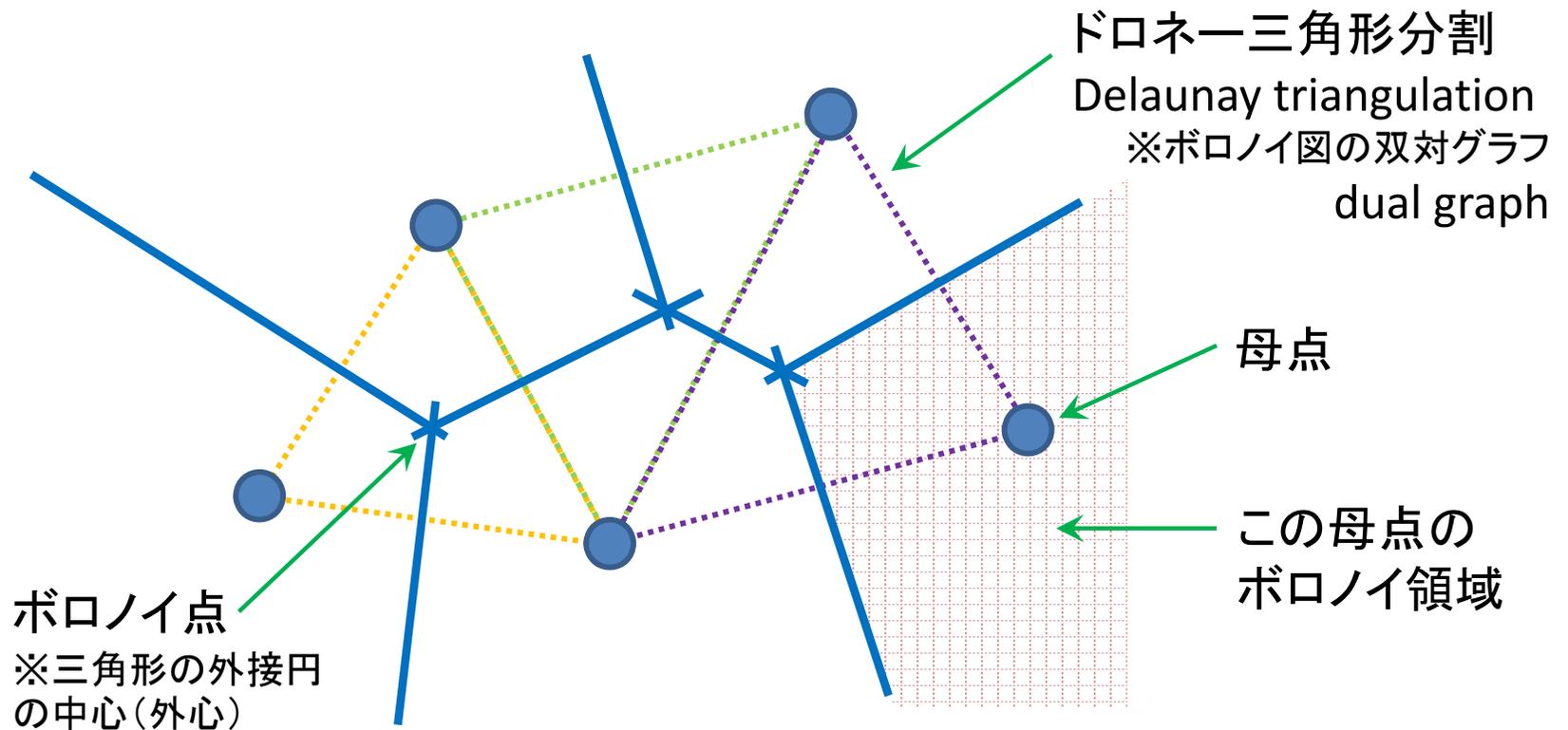
4. GIS, Voronoi diagram

堀田 敬介

Voronoi diagram とは？

- ボロノイ図 Voronoi diagram

- 空間上の複数個の点(母点)をもとに領域分割
- 各領域(ボロノイ領域)から最も近い母点はその領域内の母点
- 2次元ユークリッド空間上でユークリッド距離を考えると, 各領域は2点の垂直二等分線で分割される その際, 3つの母点を分ける境界線の交点は1点で交わる(ボロノイ点)



RでVoronoi図を簡単に描く1

- Rを起動し, R Console で以下を入力

```
> install.packages("deldir")
```

← package(deldir)のインストール

```
> library(deldir)
```

← package(deldir)の読み込み

- 母点を適当につくり voronoi図を描く

```
> x <- c(2, 3, 7, 1, 3, 8)
```

← 6点の(x,y)座標設定

```
> y <- c(2, 3, 2, 5, 8, 9)
```

(2,2), (3,3), (7,2), (1,5),...

```
> dd <- deldir(x, y, plot=T)
```

※ <- : 右のモノを左に代入

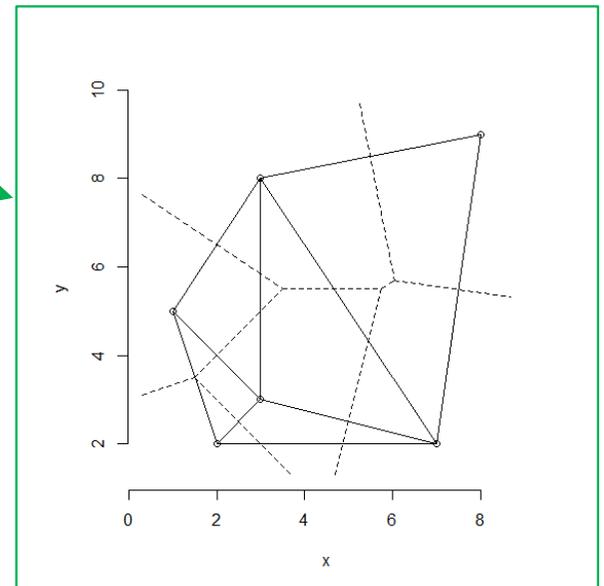
※ c(...) : ベクトルを作成

ポロノイ図 Voronoi diagram【点線】 と
ドロネー三角形分割 Delaunay triangulation【実線】
を描く

```
> tl <- tile.list(dd)
```

```
> plot(tl)
```

ポロノイ図 Voronoi diagram【実線】のみを描く



RでVoronoi図を簡単に描く2

- 座標を用意

```
> x <- runif(20)
> y <- runif(20)
```

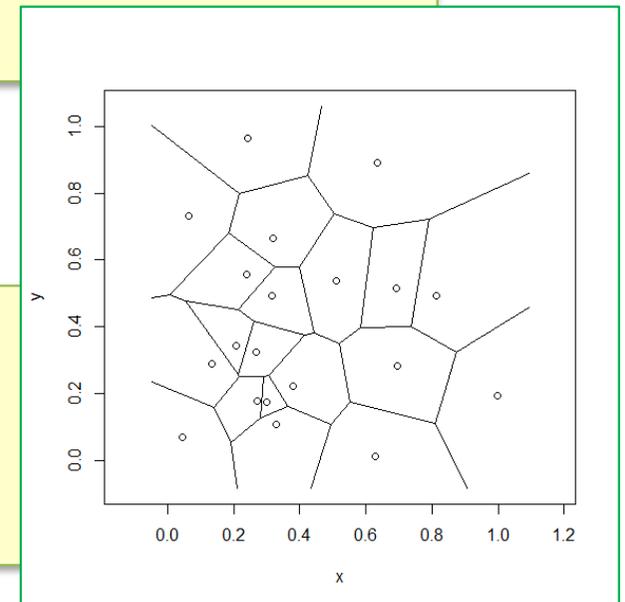
← [0,1)一様乱数を20個生成し,
(x,y)座標とする
※ `runif`: the `uniform` distribution
r = `random`

- 座標の確認

```
> x
[1] 0.04693021 0.20785805 0.32963935 0.51061063 0.81478018 0.13452010
[7] 0.26855775 0.30166600 0.24344253 0.62985486 0.37873759 0.31884216
[13] 0.63481753 0.31565712 0.99787164 0.06323913 0.24069670 0.69282271
[19] 0.27066383 0.69622948
```

- ボロノイ図を描く

```
> dd <- deldir(x, y)
> tl <- tile.list(dd)
> plot(tl)
```



GIS, Voronoi diagram

RでVoronoi図を描く【csv編】

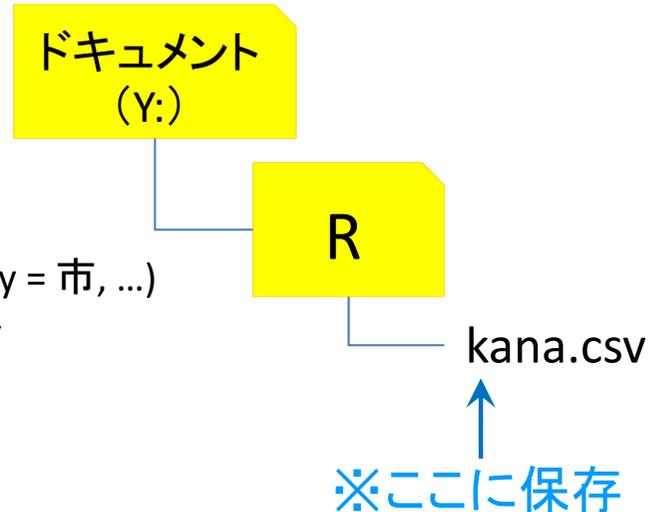
RでVoronoi図を描く【csv編0】

- データとしてcsvファイル (*.csv)を準備
 - 「マイドキュメント(Y:)」に「R」フォルダをつくり保存

“kana.csv”ファイルの中身(Excelで開いた場合)

name	type	lon	lat
鶴見区	ward	139.6825	35.5084
神奈川区	ward	139.6294	35.4769
西区	ward	139.6170	35.4536
中区	ward	139.6422	35.4447
...

- ※name: 名前
- ※type: 種類 (ward = 区, city = 市, ...)
- ※lon: longitude(経度)の略
- ※lat: latitude(緯度)の略



“kana.csv”をテキストエディタ(TeraPadやメモ帳)で開くと...

```
name,type,lon,lat
鶴見区,ward,139.6825,35.5084
神奈川区,ward,139.6294,35.4769
西区,ward,139.617,35.4536
中区,ward,139.6422,35.4447
南区,ward,139.6088,35.4315
保土ヶ谷区,ward,139.596,35.46
磯子区,ward,139.6184,35.4024
金沢区,ward,139.6244,35.3373
港北区,ward,139.633,35.5191
```

※csvファイルとは？

- データがコンマ(,)で区切られたテキストファイルのこと
- csvファイルを開くとExcelが自動で起動し、コンマ(,)で区切られたデータが各セルに表示される
- TeraPad等のテキストエディタで開くと、左のように表示される

RでVoronoi図を描く【csv編1】

• csvファイル(*.csv)を読み込みVoronoi図を描く

✓ csvファイルの読み込み

※csvファイルの1行目はheader(項目名)が記述されているのでT(True)を指定する

```
> df <- read.csv("Y:/R/kana.csv", header=T)
```

("... ")に指定したcsvファイルを読みという命令
※ファイルはフルパス("Y:/R/kana.csv")で指定
例では、YドライブのRフォルダ内にあるkana.csvという名前のファイルをフルパスで指定している

ファイルを読み込んだ後、変数dfに代入する

✓ 読み込んだファイルの中身確認

```
> df
```

← 変数dfの中身確認は変数名を書いてEnter

✓ Voronoi図を描く ※plot=T(True) とすると描く

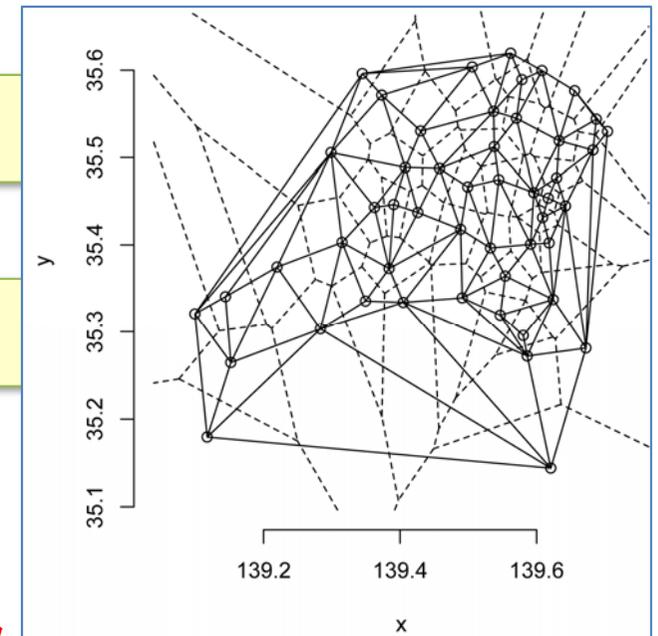
```
> dd <- deldir(df$lon, df$lat, plot=T)
```

指定した(x,y)座標でVoronoi図をつくり、
結果をddに代入

x座標として変数dfのlonを指定
y座標として変数dfのlatを指定

「変数名\$項目名」
と書くと列のデータを意味する
例) df\$lon ... 変数dfの項目lonのデータ

描画したボロノイ図【点線】
とドローネー三角形分割【実線】



RでVoronoi図を描く【csv編2】

- Voronoi図だけを描きたい場合

(ドローネー三角形分割は表示したくない場合)

```
> df <- read.csv("Y:/R/kana.csv", header=T)
```

```
> df
```

変数dfの中身を確認

変数dfの中身

```
> dd <- deldir(df$lon, df$lat)
```

```
> tl <- tile.list(dd)
```

```
> plot(tl)
```

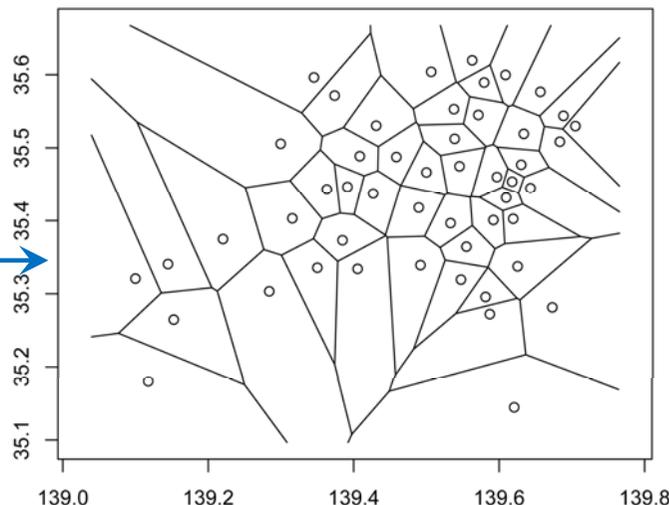
x座標として変数dfのlonを指定
y座標として変数dfのlatを指定

voronoi図だけを作り、変数tlに代入

変数tlをplot(描画)

※ボロノイ図【実線】
のみを描いた

name	type	lon	lat
鶴見区	ward	139.6825	35.5084
神奈川区	ward	139.6294	35.4769
西区	ward	139.6170	35.4536
中区	ward	139.6422	35.4447
...



「変数名\$項目名」

と書くと列のデータを意味する

例) df\$lon ... 変数dfの項目lonのデータ

例) df\$type ... 変数dfの項目typeのデータ

例) df\$name ... 変数dfの項目nameのデータ

RでVoronoi図を描く【csv編3】

- 母点のtype毎にVoronoi領域を色分けしたいとき

```
> df <- read.csv("Y:/R/kana.csv", header=T)
```

```
> df
```

変数dfの中身を確認 → 変数dfの中身

```
> dd <- deldir(df$lon, df$lat, z=df$type)
```

```
> tl <- tile.list(dd)
```

```
> plot(tl)
```

x座標として変数dfのlonを指定
y座標として変数dfのlatを指定
zとして変数dfのtypeを指定

name	type	lon	lat
鶴見区	ward	139.6825	35.5084
神奈川区	ward	139.6294	35.4769
西区	ward	139.6170	35.4536
中区	ward	139.6422	35.4447
...

df\$type

df\$lon

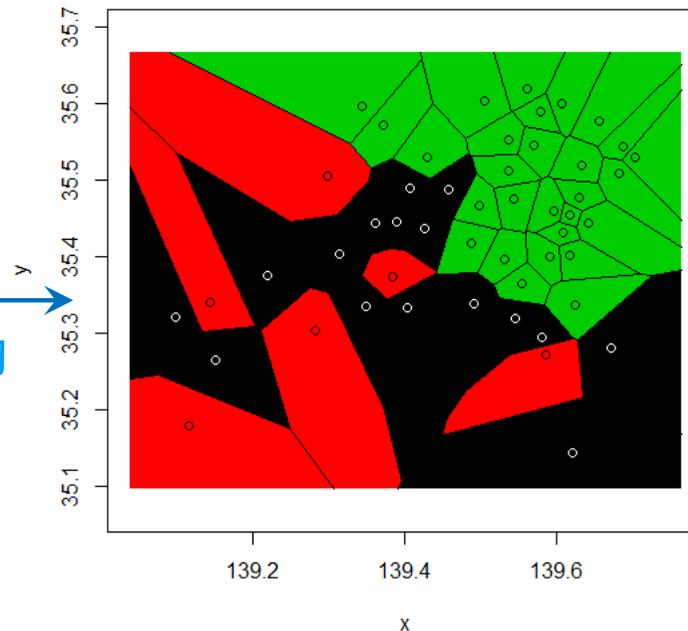
df\$lat

※ボロノイ図【実線】の各母点のボロノイ領域をtype毎に色分けする
色は指定しなければ自動例では、

ward (区) = 緑色

city (市) = 黒色

county (郡) = 赤色



注) county は主に米国の郡をさす
「高座郡寒川町」と言いたい時は
Samukawa Town, Koza District等

GIS, Voronoi diagram

RでVoronoi図を描く【[google地図編](#)】

RでVoronoi図を描く【地図編1】

- Voronoi図とGoogle地図を重ねて描く
- 準備

```
> install.packages("deldir")
```

```
> install.packages("ggmap")
```

```
> library(deldir)
```

```
> library(ggmap)
```

packageのインストール
(deldir と ggmap)

※インストール済みなら不要

パッケージの読み込み
(deldir と ggmap)

※読み込み済みなら不要

– Tips!(裏技・小ネタ)

```
> library()
```

※現在インストールされているパッケージのリストが表示される

```
> setwd("C:/Users/〇〇/Desktop/R")
```

※wd(working directory)設定 設定しておくで、ここにあるファイルはファイル名だけで指定できる

※例はWindowsのデスクトップにRフォルダを作っている場合(〇〇にはユーザー名が入る)

[ctrl]+[p] ...1つ前の既出命令

[ctrl]+[n] ...1つ後の既出命令

をそれぞれ表示

※同じ命令を少し修正して書きたい
場合などに便利

RでVoronoi図を描く【地図編2】

- ファイル(*.csv)の読み込み

```
> df <- read.csv("Y:/R/kana.csv", header=T)
```

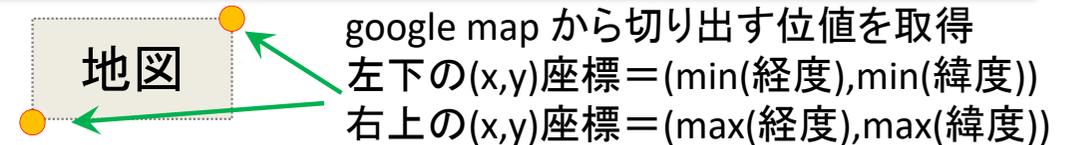
name	type	lon	lat
鶴見区	ward	139.6825	35.5084
神奈川区	ward	139.6294	35.4769
西区	ward	139.6170	35.4536
中区	ward	139.6422	35.4447
...

kana.csv

- 経度・緯度の最小値・最大値を取得

```
> loc <- c(min(df$lon), min(df$lat), max(df$lon), max(df$lat))
```

- Google map の取得



```
> GM <- ggmap(get_map(location=loc, zoom=10, source="google"))
```

※location ... 切り出す地図の位置(左下(x,y)座標-右上(x,y)座標の4点で指定)

※zoom ... 地図の縮尺指定:1市が対象なら12ぐらい, 1県が対象なら10ぐらいに設定

※source ... 取得する地図のソース(源, 元, 情報源, 出所)

- 地図描画確認

```
> GM
```

※欲しいサイズになっているかどうか確認

縮尺はzoomで調整するので, 値を小さくしたり大きくしたりして再度読み込み直す

RでVoronoi図を描く【地図編3】

• 母点の描画準備

```
> GP <- geom_point(data=df, aes(x=lon, y=lat, colour=factor(type)), size=3)
```

母点の大きさを3に設定

dataとしてdfを使う

x座標としてdfのlon

y座標としてdfのlatを使う

母点の色分けにdfのtypeを利用する

• ボロノイ領域の描画準備

```
> dd <- deldir(df$lon, df$lat)
> GS <- geom_segment(data=dd$dirsgs, aes(x=x1, y=y1, xend=x2, yend=y2), size=0.5, linetype=1)
```

dataとして変数ddのdirsgsを使う

(x,y)座標としてdd\$dirsgsの(x1,y1)を
(xend,yend)としてdd\$dirsgsの(x2,y2)を使う

※線の太さ

※線種

実線 = 1

各種点線 = 2,3,4,5,...

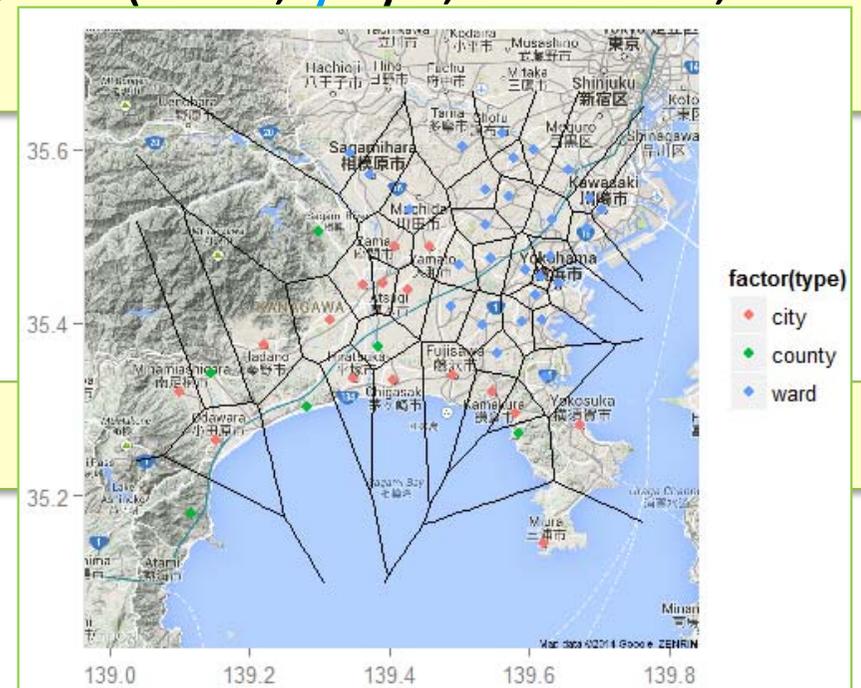
• 描画

```
> GM+GP+GS
```

※google map

※母点

※ボロノイ領域



RでVoronoi図を描く【地図編】

Tips! たくさんの命令を打つのは大変だし間違えちゃう！
一度にまとめて命令したい！

- ① 「TeraPad」「メモ帳」などのテキストエディタを起動. まとめて実行したい命令を, 以下の様に書く

```
df <- read.csv("kana.csv", header=T)
loc <- c(min(df$lon), min(df$lat), max(df$lon), max(df$lat))
GM <- ggmap(get_map(location=loc, zoom=10, source="google"))
GP <- geom_point(data=df, aes(x=lon, y=lat, colour=factor(type)), size=3)
dd <- deldir(df$lon, df$lat)
GS <- geom_segment(data=dd$dirsgs, aes(x=x1, y=y1, xend=x2, yend=y2), size=0.5,
linetype=1)
GM+GP+GS
```

- ② ファイルの種類を「全てのファイル」にし, 保存する場所(フォルダ, ディレクトリ)を選び, ファイル名を「ファイル名.R」として保存. ファイル名の部分は半角アルファベットで好きな名前を付ける (例: ファイル名「vormap.R」とし「Y:/R/」フォルダに保存)
- ③ R(R Studio)で以下を実行

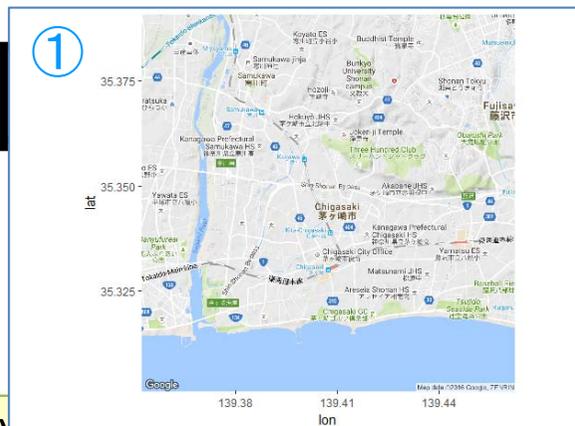
```
> source("Y:/R/vormap.R")
```

RでVoronoi図を描く【地

Tips! ggmapで読み込む地図の種類を変える

① default (`mptype="terrain"`)

```
> GM <- ggmap(get_map(location=loc, zoom=13, source="google"))
```



② roadmap (`mptype="roadmap"`)

```
> GMr <- ggmap(get_map(location=loc, zoom=14, source="google", mptype="roadmap"))
```

③ hybrid (`mptype="hybrid"`)

```
> GMh <- ggmap(get_map(location=loc, zoom=14, source="google", mptype="hybrid"))
```

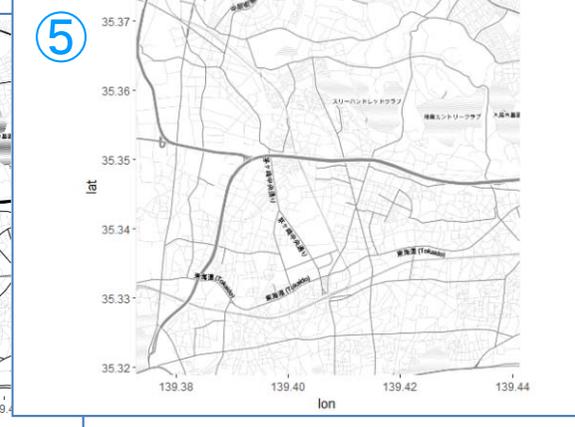
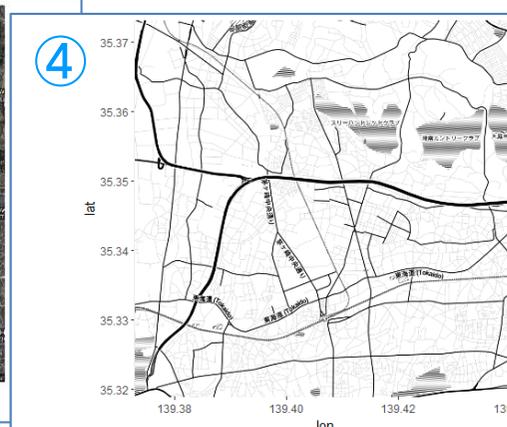
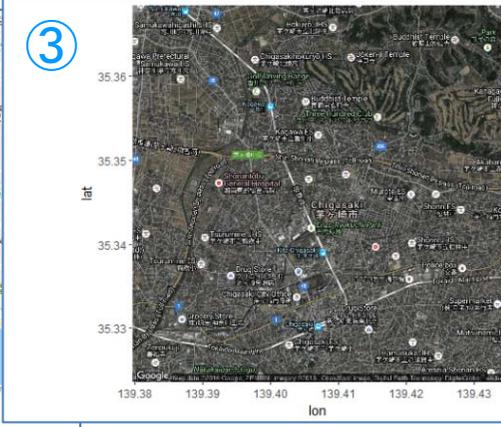
④ toner (`mptype="toner"`)

注) zoomの数値が大きいと、読み込みに相当時間がかかる。まずは11程度で試そう

```
> GMt <- ggmap(get_map(location=loc, zoom=14, source="stamen", mptype="toner"))
```

⑤ toner-lite (`mptype="toner-lite"`)

```
> GMl <- ggmap(get_map(location=loc, zoom=14, source="stamen", mptype="toner-lite"))
```



もっと詳しく・参考文献

- 参考文献

- 谷村晋「Rで学ぶデータサイエンス7地理空間データ分析」共立出版(2010)
- 山本他「Rで学ぶデータサイエンス12統計データの視覚化」共立出版(2013)
- J.P.Lander「みんなのR」マイナビ(2015)
- W.Chang「Rグラフィックスクックブック」オライリー(2013)