Homework1

Laha Ale

1/27/2019

Exercise 7

a)

$P1 = (\theta_1, \lambda_1), P2 = (\theta_2, \lambda_2)$

$D = R\phi = Rarccos[sin\theta_1 sin\theta_2 + cos\theta_1 cos\theta_2 cos(\lambda_1 - \lambda_2)]$

```
#R code block
geodesic <- function(P1,P2,R=6371)
{
  P1 <- P1*pi/180
  P2 <- P2*pi/180
  theta_1 <- P1[2]
  lambda_1 <- P1[1]
  theta_2 <- P2[2]
  lambda_2 <- P2[1]
  D <- R*acos(sin(theta_1)*sin(theta_2)+cos(theta_1)*cos(theta_2)*cos(lambda_1-lambda_2))
  return(D)
}
```

b)

```
#R code block
Chicago <- c(87.63,41.88)
Minneapolis <- c(93.22,44.89)
geo_D_CM <- geodesic(Chicago,Minneapolis)
print(paste("The distance between Chicago and Minneapolis is",geo_D_CM))
```

```
## [1] "The distance between Chicago and Minneapolis is 561.994673102893"
```

```
#R code block
New_York <- c(73.97,40.78)
New_Orleans <- c(90.25,29.98)
geo_D_NN <- geodesic(New_York,New_Orleans)
print(paste("The distance between New York and New Orleans is",geo_D_NN))
```

```
## [1] "The distance between New York and New Orleans is 1897.21478244142"
```

Exercise 8

Suppose, $P1 = (\theta_1, \lambda_1), P2 = (\theta_2, \lambda_2)$
Naive Euclidean Distance:
$||P_1 - P_2||\pi R/180 = \pi R/180\sqrt{(\theta_2 - \theta_1)^2 + (\lambda_2 - \lambda_1)^2}$

```r
#R code block
Naive_Euclidean<-function(P1,P2,R=6371)
{
  theta_1 <- P1[2]
  lambda_1 <- P1[1]
  theta_2 <- P2[2]
  lambda_2 <- P2[1]
  D <- sqrt((theta_2- theta_1)^2+(lambda_2-lambda_1)^2)*pi*R/180
  return(D)
}
```

```r
#R code block
NE_D_CM<- Naive_Euclidean(Chicago,Minneapolis)
print(paste("The Naive Euclidean distance between Chicago and Minneapolis is",NE_D_CM))
```

```
## [1] "The Naive Euclidean distance between Chicago and Minneapolis is 705.962569354093"
```

```r
#R code block
NE_D_NN <- Naive_Euclidean(New_York,New_Orleans)
print(paste("The Naive Euclidean distance between New York and New Orleans is",NE_D_NN))
```

```
## [1] "The Naive Euclidean distance between New York and New Orleans is 2172.36983756042"
```

As we can see from above, Naive Euclidean distance larger than geodesic distance.

Exercise 9

Chordal distnace:
$chordal = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$

```r
#R code block
Chordal_D<-function(P1,P2,R=6371)
{
  P1 <- P1*pi/180
  P2 <- P2*pi/180
  theta_1 <- P1[2]
  lambda_1 <- P1[1]
  theta_2 <- P2[2]
  lambda_2 <- P2[1]
  x_1 <- R*cos(theta_1) * cos(lambda_1)
  y_1 <- R*cos(theta_1)* sin(lambda_1)
  z_1 <- R*sin(theta_1)
  x_2 <- R*cos(theta_2) * cos(lambda_2)
  y_2 <- R*cos(theta_2)* sin(lambda_2)
  z_2 <- R*sin(theta_2)
  D <- sqrt((x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2)
  return(D)
}
```

```r
#R code block
C_D_CM <- Chordal_D(Chicago,Minneapolis)
print(paste("The Chordal distance between Chicago and Minneapolis is",C_D_CM))
```

```
## [1] "The Chordal distance between Chicago and Minneapolis is 561.812481690168"
```

```
C_D_NN <- Chordal_D(New_York,New_Orleans)
print(paste("The Chordal distance between New York and New Orleans is",C_D_NN))
```

```
## [1] "The Chordal distance between New York and New Orleans is 1890.21246768842"
```

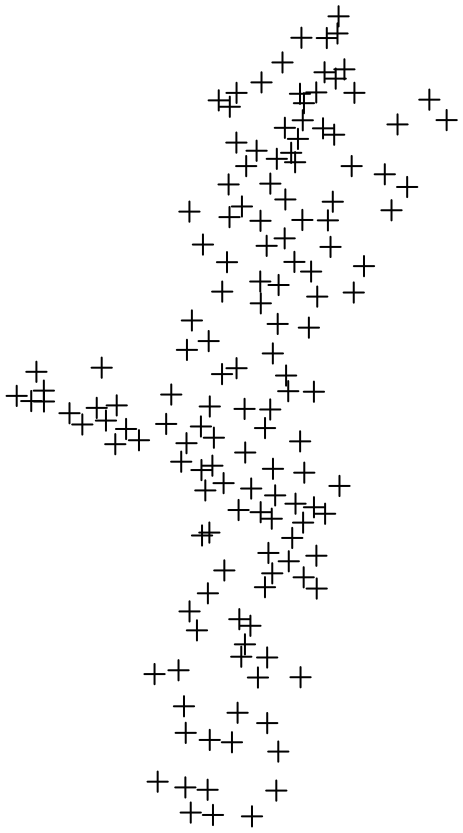
Cities	Geodesic	Naive Euclidean	Chordal
Chicago,Minneapolis	561.99	705.96	561.81
New York,New Orleans	1897.21	2172.37	1890.21

As we can see from above, Naive Euclidean distance larger than geodesic distance, and geodesic distance than Chordal distance.

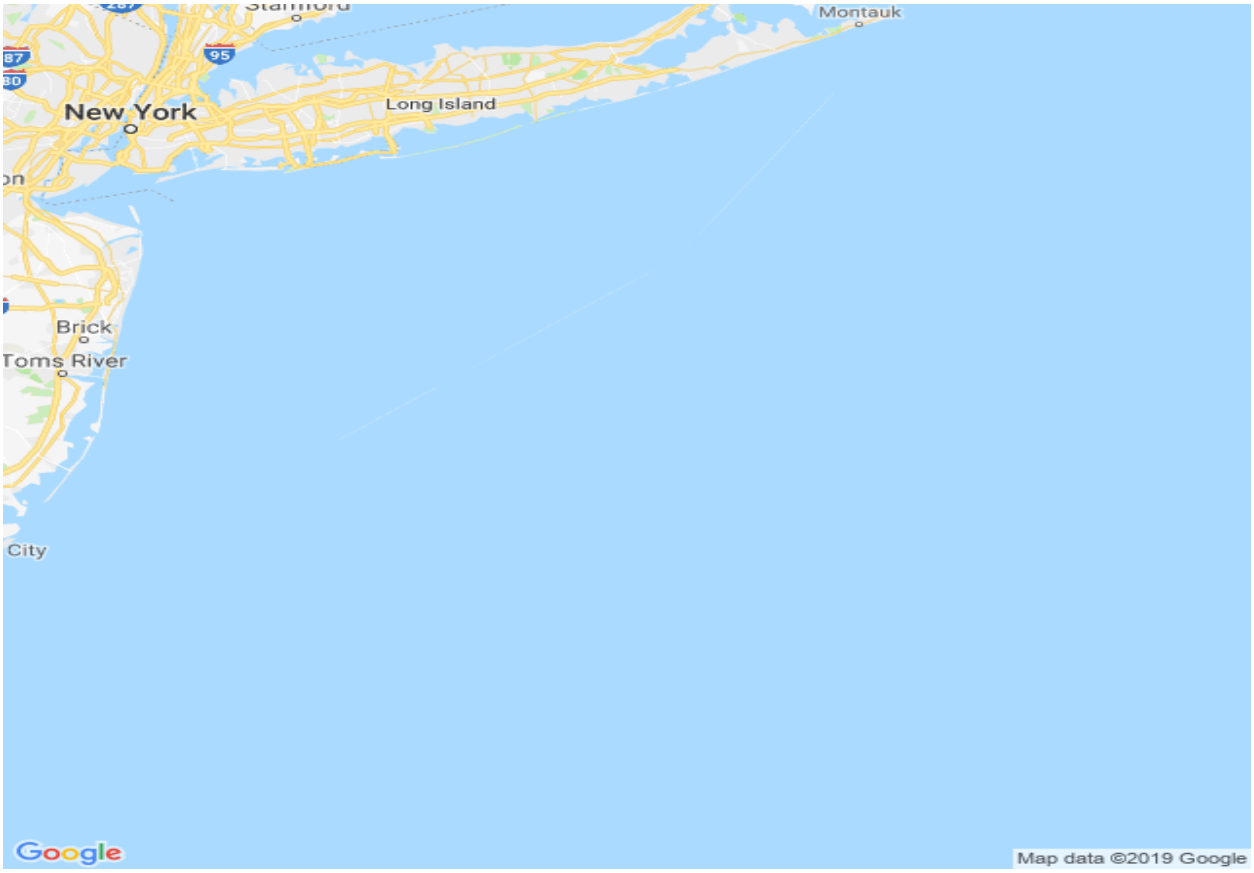
Exercise 11

```
#R code block
#Sys.setenv(GOOGLE_MAPS_API_KEY = "may google api key")
library(RgoogleMaps)
library(sp)
library(maptools)
library(rgdal)
scallops_url="https://www.counterpointstat.com/uploads/1/1/9/3/119383887/scallops.txt"
data <- read.table(scallops_url,header=TRUE)
hybrid <- GetMap.bbox(lonR = range(data$long), latR = range(data$lat),
                      size=c(640,640), maptype = "hybrid")
PlotOnStaticMap(hybrid)
```

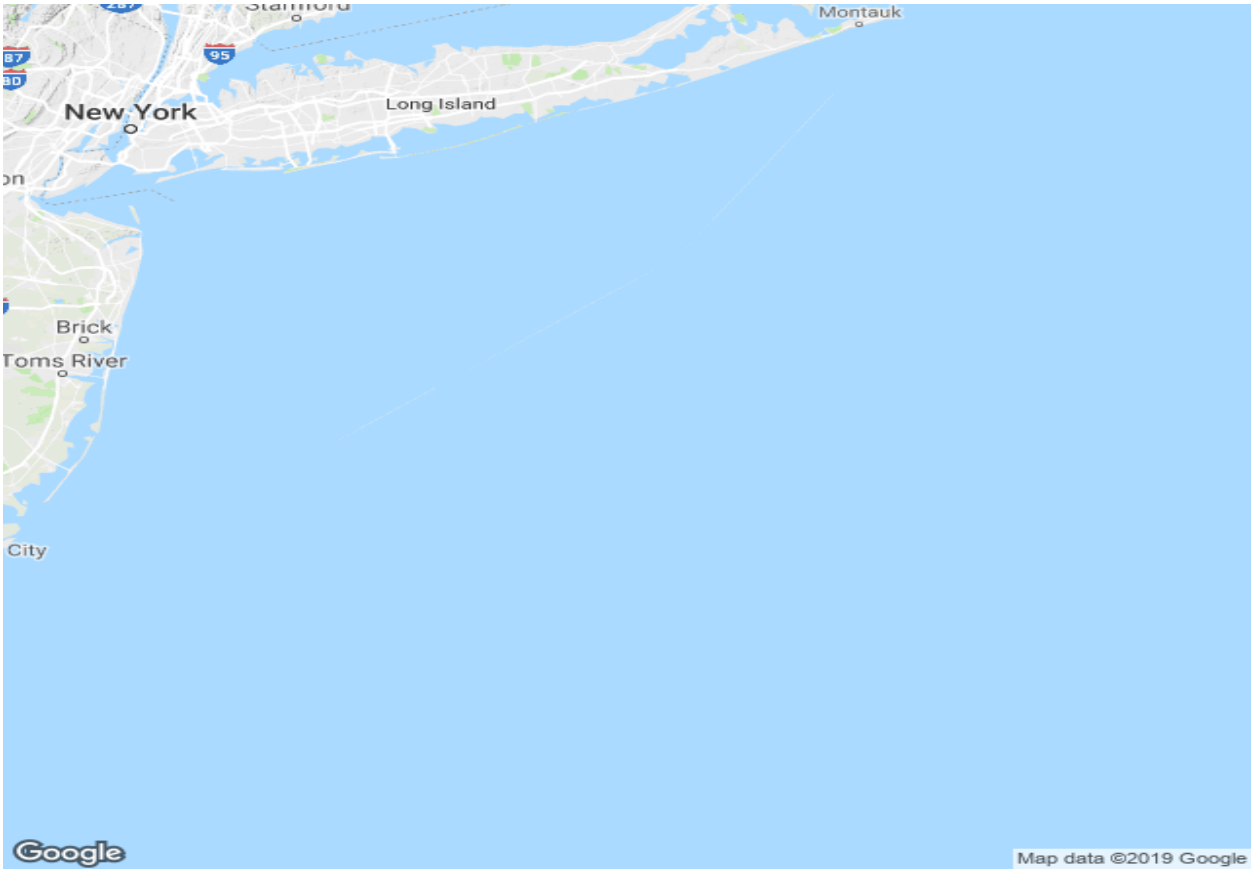
```
SP_longlat <- SpatialPoints(coords = cbind(data$long,data$lat),
                            proj4string = CRS("+proj=longlat +ellps=WGS84"))
SP_utm <- spTransform(SP_longlat, CRS("+proj=utm +zone=13 +datum=WGS84"))
plot(SP_utm)
```



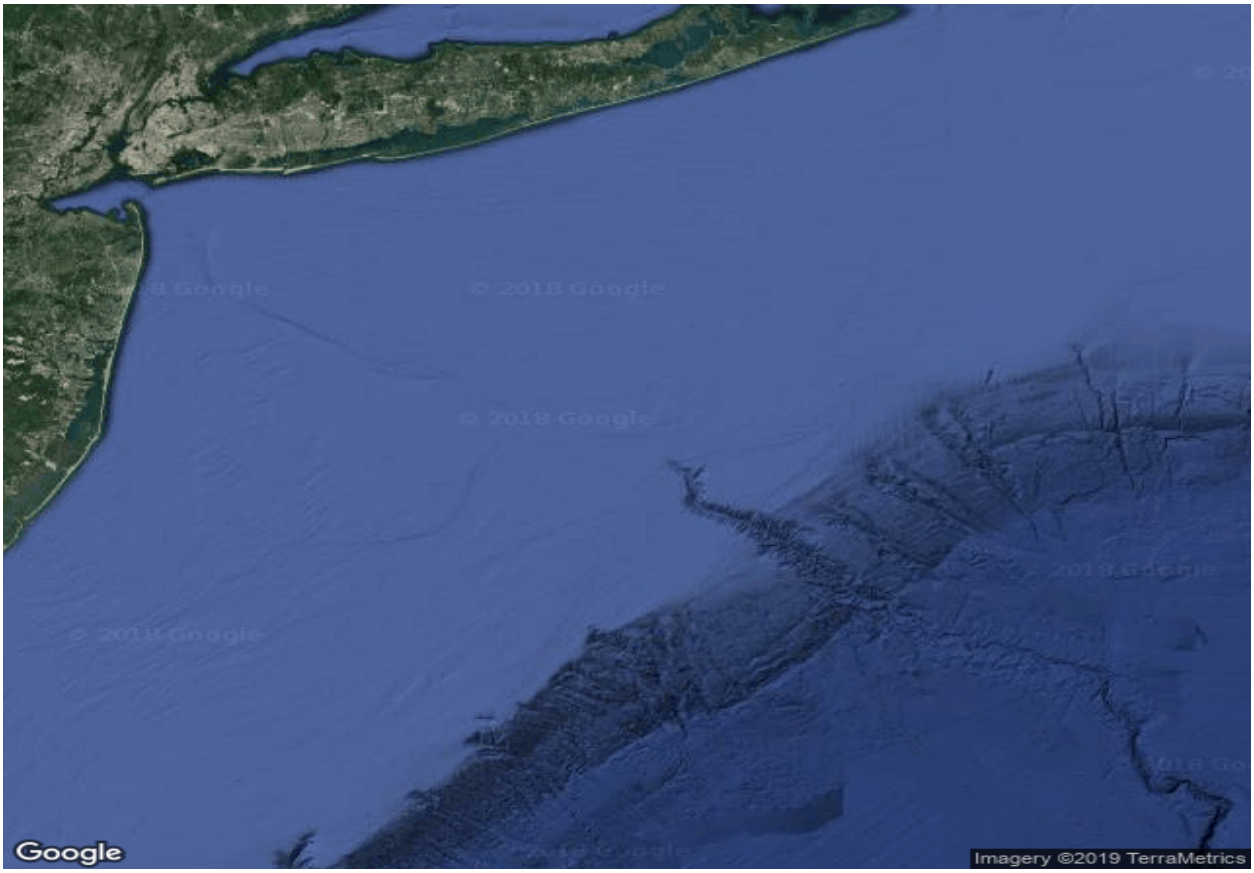
```
mobile <- GetMap.bbox(lonR = range(data$long), latR = range(data$lat),
                      size=c(640,640), maptype = "mobile")
PlotOnStaticMap(mobile)
```




```
terrain <- GetMap.bbox(lonR = range(data$long), latR = range(data$lat),
                       size=c(640,640), maptype = "terrain")
PlotOnStaticMap(terrain)
```




```
satellite <- GetMap.bbox(lonR = range(data$long), latR = range(data$lat),
                         size=c(640,640), maptype = "satellite")
PlotOnStaticMap(satellite)
```


Google
Imagery ©2019 TerraMetrics