Project 1

Jiawen Qi (jiq10)

1/30/2017

Spatial Data Manipulation

a. Find the total number of road segments

Totally, there are 22,222 road segments in pgh_street shapefile.

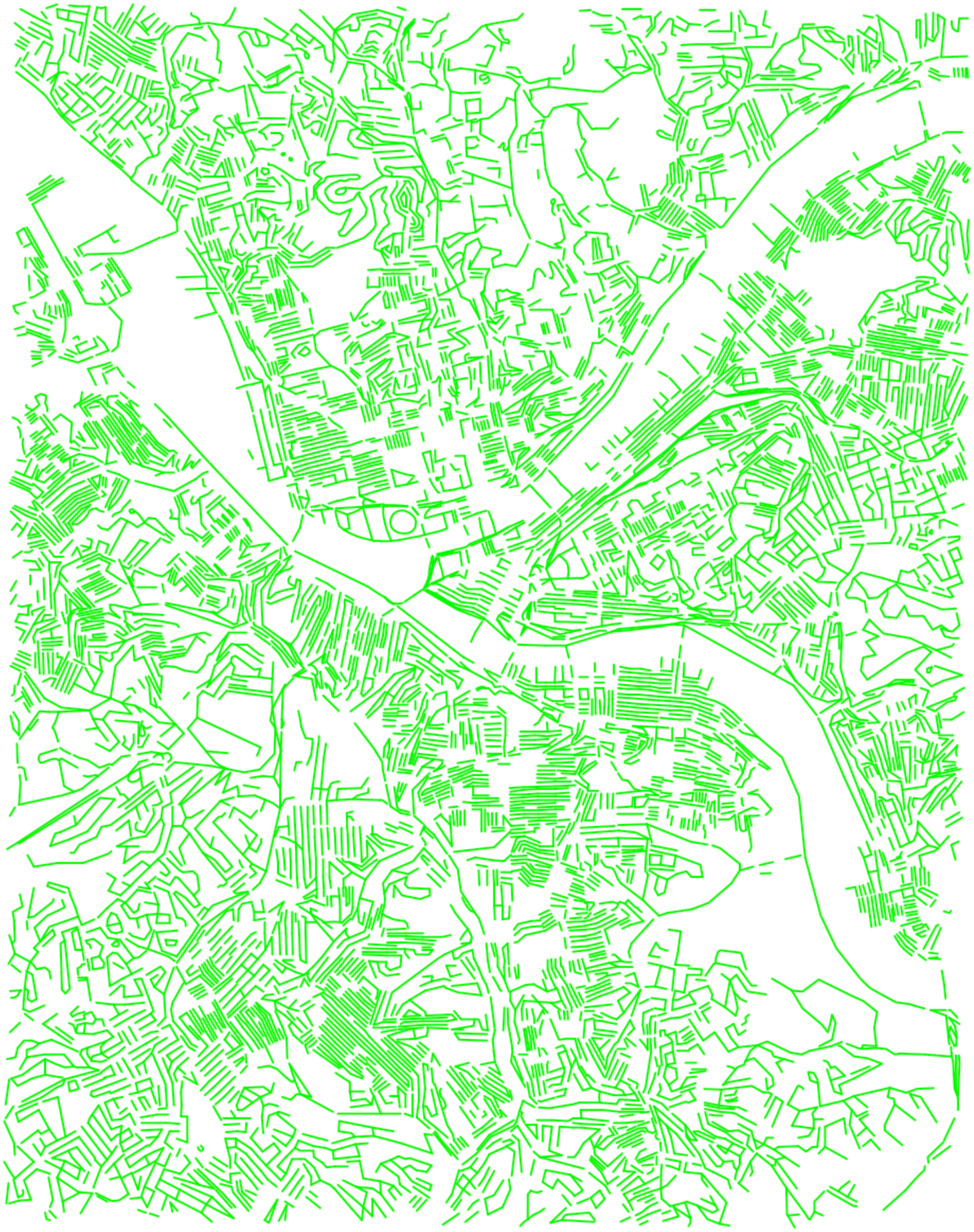
b. Calculate minimum, maximum, and mean segment lengths

The minimum segment length is 3e-04.
The maximum segment length is 1.46654.
The mean segment length is 0.05979852.

c. Filter out the segments that are below the mean length that you calculated in (b) and then create a map showing the remaining segments

(next page)

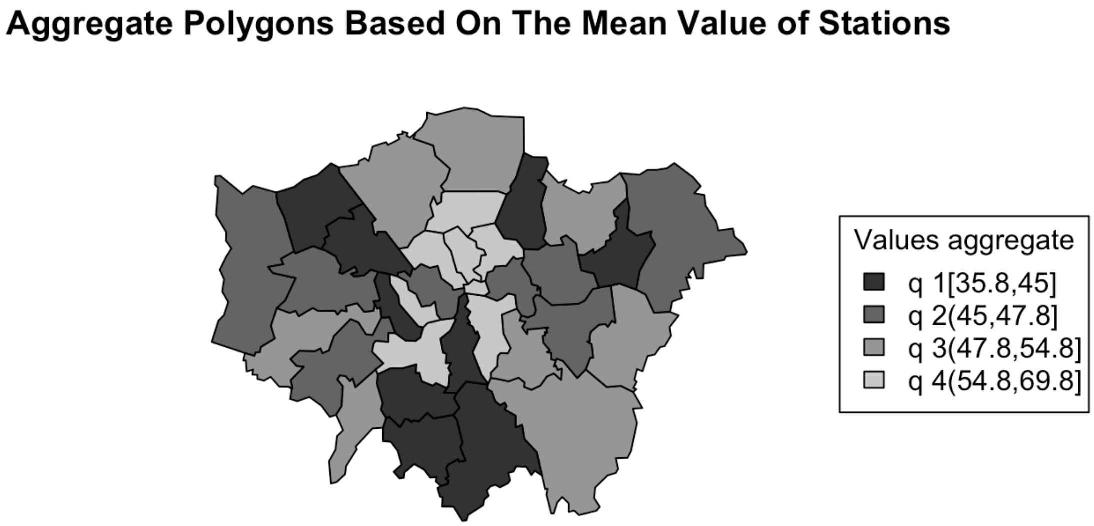
Pittsburgh streets



roads segments longer than mean (1.46654)

Spatial Data Aggregation

a. Aggregate the data based on the mean of the point values. Create a map and prepare a report on the result



Aggregate Polygons Based On The Mean Value of Stations

Report: Let's call the map 'Area'. In this area, most cities located at center have a larger station mean value than remainings. Bottom center, left top, and right top areas have lower station mean values.

b. Run regression on the point values before and after aggregation. Prepare a report on the result.

Regression Result:

```
Call:

lm(formula = stations$Value ~ x)

Residuals:

   Min      1Q  Median      3Q     Max

-47.727 -24.071  -0.744  25.054  51.998

Coefficients:

           Estimate Std. Error t value Pr(>|t|)

(Intercept) 49.04916    7.89595   6.212 8.79e-10 ***

x          -0.02601    0.16367  -0.159    0.874

---
```

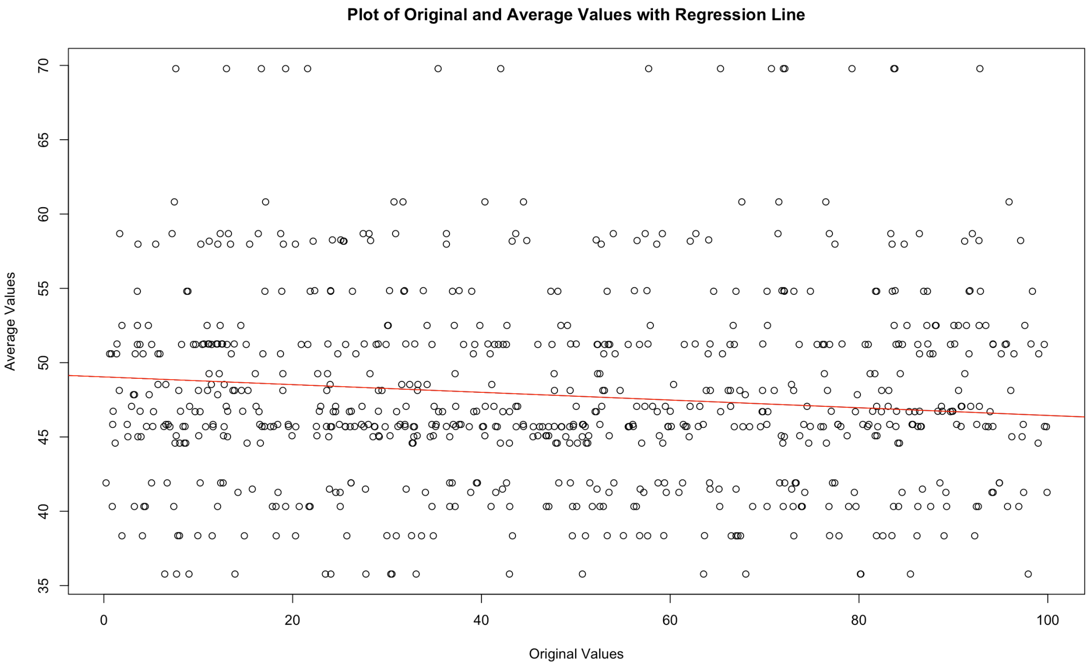
```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 28.65 on 729 degrees of freedom

Multiple R-squared:  3.463e-05,	Adjusted R-squared:  -0.001337

F-statistic: 0.02525 on 1 and 729 DF,  p-value: 0.8738
```

Plot the result:



Explanation:

Let's call the values before aggregation 'Original Values' and values after aggregation 'Average Values'. From the summary of this regression. We can generate a formula: OriginalValues = -0.02601 * AverageValues + 49.04916.

For example, if the AverageValues is 1, the OriginalValues will be predicted as 49.02315. If the AverageValues is 100, the OriginalValues will be predicted as 46.44816. As you can see from the plot above, low values will be averaged higher and high values will be averaged lower. This is how the mean works in statistics. When looking at the p-value in summary, it's 0.8738. Roughly speaking, when p-value > 0.10, there is little or no evidence against null hypothsis. In this case, x, the average values is not/less significant to predict the original values.

(Code is attached on next page)

Code

```r
# Load libraries
library(sp)  # Classes and Methods for Spatial Data

library(spdep)  # Spatial Dependence: Weighting Schemes, Statistics and Models

library(classInt)  # Choose Univariate Class Intervals

library(rgeos)  # Interface to Geometry Engine - Open Source (GEOS)

library(maptools)  # Tools for Reading and Handling Spatial Objects

library(rgdal)  # Binding for the Geospatial Data Abstraction Library

library(ggplot2)  # Create Elegant Data Visualisations Using the Grammar of Graphics
library(weights)  # Weighting and Weighted Statistics

##### Spatial Data Manipulation #####
getwd()  # get the working directory
list.files()  # List the Files in a Directory
pghstreet <- readOGR(dsn = "/Users/qijiawen/Desktop/2017 Spring/Spatial Data Analytics/Project 1",
    layer = "pgh_streets")  # read pgh_streets shapefile

summary(pghstreet)  # get a summary of pghsteet

###### a ######
length(pghstreet)  # get the length of pghstreet

###### b ######
min(pghstreet$LENGTH)  # min
max(pghstreet$LENGTH)  # max
mean(pghstreet$LENGTH)  # mean

###### c ######
pghstreet.filtered <- pghstreet[pghstreet$LENGTH > mean(pghstreet$LENGTH), ]
# filter out the roads segments that are below the mean length
summary(pghstreet.filtered)  # get a summary
plot(pghstreet.filtered, lwd = 1, col = "green")  # plot roads segments, line width is 1 and color is green
title(main = "Pittsburgh streets", sub = "roads segments longer than mean (1.46654)")  # add a title

##### Spatial Data Aggregation #####
load("stations.RData")  # load stations dataset
summary(stations)  # get a summary of station
load("lnd.RData")  # load lnd dataset
summary(lnd)  # get a summary of lnd
```

```r
plot(lnd)  # plot the polygons
plot(stations, add = T, col = "red")

###### a ######
stations.mean <- aggregate(stations[c("Value")], by = lnd, FUN = mean)  # station.mean is the mean value of stations for each polygon(city)

q <- cut(stations.mean$Value, breaks = c(quantile(stations.mean$Value)), include.lowest = T)
summary(q)
clr <- as.character(factor(q, labels = paste0("gray", seq(20, 80, 20))))  # color
plot(stations.mean, col = clr)
legend(legend = paste0("q ", 1:4, levels(q)), fill = paste0("gray", seq(20,
    80, 20)), "right", title = "Values aggregate")  # add legend
title(main = "Aggregate Polygons Based On The Mean Value of Stations")

###### b ######
stations.c <- aggregate(stations, by = lnd, FUN = length)  # aggregation all the stations by lnd, function is length()

stations.c@data[, 1]  # shows the numbers of stations falling in each polygon
x <- rep.int(stations.mean$Value, stations.c@data[, 1])  # repeate the mean value, times = numbers of stations falling in each polygon
str(x)
reg.model <- lm(stations$Value ~ x)  # do linear regression
summary(reg.model)  # get the summary
plot(stations$Value, x, main = "Plot of Original and Average Values with Regression Line",
    xlab = "Original Values", ylab = "Average Values")
abline(reg.model, col = "red")

##### end #####
```