3D\_spatial\_network

Suraj Vidyadaran

Wednesday, December 23, 2015

### Data Description

This dataset was constructed by adding elevation information to a 2D road network in North Jutland, Denmark (covering a region of 185 x 135 km^2). Elevation values where extracted from a publicly available massive Laser Scan Point Cloud for Denmark (available at : [Web Link] (Bottom-most dataset)). This 3D road network was eventually used for benchmarking various fuel and CO2 estimation algorithms. This dataset can be used by any applications that require to know very accurate elevation information of a road network to perform more accurate routing for eco-routing, cyclist routes etc. For the data mining and machine learning community, this dataset can be used as 'ground-truth' validation in spatial mining techniques and satellite image processing. It has no class labels, but can be used in unsupervised learning and regression to guess some missing elevation information for some points on the road. The work was supported by the Reduction project that is funded by the European Comission as FP7-ICT-2011-7 STREP project number 288254.

### Attribute Information

1. OSM\_ID: OpenStreetMap ID for each road segment or edge in the graph.
2. LONGITUDE: Web Mercaptor (Google format) longitude
3. LATITUDE: Web Mercaptor (Google format) latitude
4. ALTITUDE: Height in meters.

Note: OSM\_ID is the ID assigned by OpenStreetMaps ([Web Link]) to the road segments. Each (long,lat,altitude) point on a road segment (with unique OSM ID) is sorted in the same order as they appear on the road. So a 3D-polyline can be drawn by joining points of each row for each OSM\_ID road segment.

### Data Cleaning and Exploratory Analysis

spatial\_network <- read.csv("F:/Analysis\_Practice/Regression/3D\_spatial\_network/3D\_spatial\_network.txt", header=FALSE)  
  
colnames(spatial\_network)<-c("OSM\_ID","longitude","latitude","altitude")  
  
#Basic structure of the data  
str(spatial\_network)

## 'data.frame': 434874 obs. of 4 variables:  
## $ OSM\_ID : int 144552912 144552912 144552912 144552912 144552912 144552912 144552912 144552912 144552912 144552912 ...  
## $ longitude: num 9.35 9.35 9.35 9.35 9.35 ...  
## $ latitude : num 56.7 56.7 56.7 56.7 56.7 ...  
## $ altitude : num 17.1 17.6 18.1 18.3 18.4 ...

#Summary of data  
summary(spatial\_network)

## OSM\_ID longitude latitude altitude   
## Min. : 4482444 Min. : 8.146 Min. :56.58 Min. : -8.608   
## 1st Qu.: 82678969 1st Qu.: 9.338 1st Qu.:56.85 1st Qu.: 7.028   
## Median :101979668 Median : 9.887 Median :57.04 Median : 17.575   
## Mean : 97869978 Mean : 9.732 Mean :57.08 Mean : 22.185   
## 3rd Qu.:125954704 3rd Qu.:10.172 3rd Qu.:57.31 3rd Qu.: 31.810   
## Max. :157742416 Max. :11.199 Max. :57.75 Max. :134.442

#Changing the ID into factors  
spatial\_network$OSM\_ID<-as.factor(spatial\_network$OSM\_ID)  
str(spatial\_network)

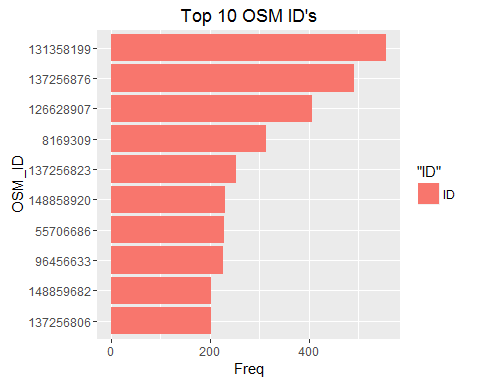
## 'data.frame': 434874 obs. of 4 variables:  
## $ OSM\_ID : Factor w/ 57329 levels "4482444","4550137",..: 49705 49705 49705 49705 49705 49705 49705 49705 49705 49705 ...  
## $ longitude: num 9.35 9.35 9.35 9.35 9.35 ...  
## $ latitude : num 56.7 56.7 56.7 56.7 56.7 ...  
## $ altitude : num 17.1 17.6 18.1 18.3 18.4 ...

### Data Analysis

# Create a table of ID's  
ID\_table<-as.data.frame(table(spatial\_network$OSM\_ID))  
ID\_table<-ID\_table[order(-ID\_table$Freq),]  
rownames(ID\_table)<-NULL  
colnames(ID\_table)<-c("ID","Freq")  
#Top 10 ids  
top\_ids<-ID\_table[1:10,]  
library(ggplot2)

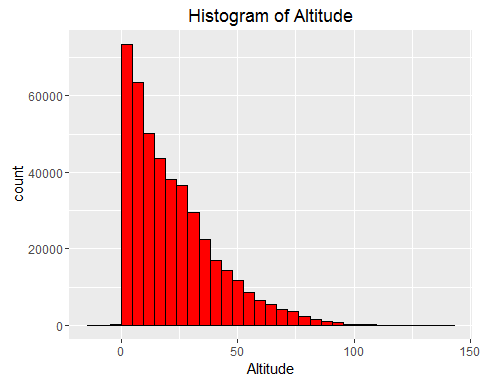
## Warning: package 'ggplot2' was built under R version 3.2.3

ggplot(top\_ids,aes(x=reorder(ID,Freq),y=Freq,fill="ID"))+geom\_bar(stat = "identity")+coord\_flip()+xlab("OSM\_ID")+ylab("Freq")+ggtitle("Top 10 OSM ID's")



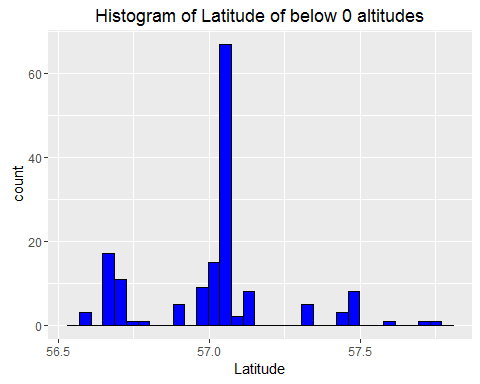
#Inspecting the altitude  
library(ggplot2)  
qplot(spatial\_network$altitude,geom="histogram",fill=I("red"),col=I("black"),main="Histogram of Altitude",xlab="Altitude")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



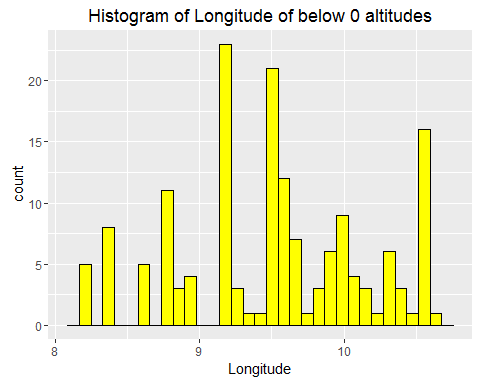
below\_0\_altitude<-subset(spatial\_network,spatial\_network$altitude<0)  
Above\_100\_altitude<-subset(spatial\_network,spatial\_network$altitude>100)  
  
qplot(below\_0\_altitude$latitude,geom="histogram",fill=I("blue"),col=I("black"),main="Histogram of Latitude of below 0 altitudes",xlab="Latitude")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



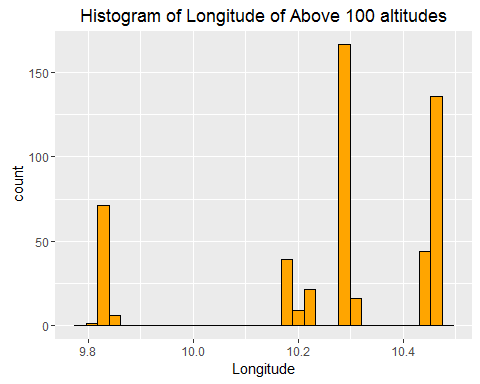
qplot(below\_0\_altitude$longitude,geom="histogram",fill=I("Yellow"),col=I("black"),main="Histogram of Longitude of below 0 altitudes",xlab="Longitude")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



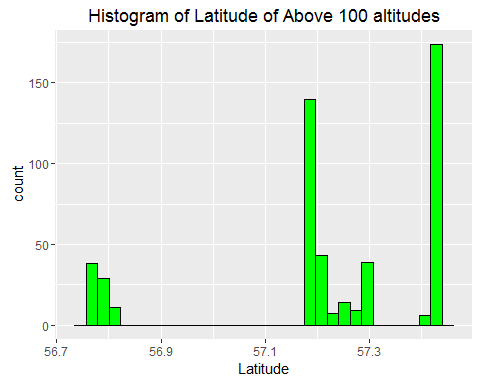
qplot(Above\_100\_altitude$longitude,geom="histogram",fill=I("Orange"),col=I("black"),main="Histogram of Longitude of Above 100 altitudes",xlab="Longitude")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

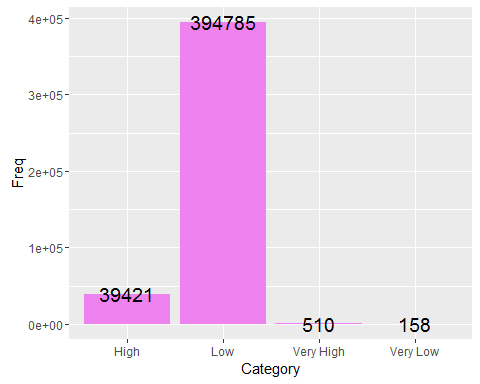


qplot(Above\_100\_altitude$latitude,geom="histogram",fill=I("green"),col=I("black"),main="Histogram of Latitude of Above 100 altitudes",xlab="Latitude")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#Create a new Category for Altitudes  
spatial\_network$altitude\_category<-NA  
spatial\_network$altitude\_category[spatial\_network$altitude<0]<-"Very Low"  
spatial\_network$altitude\_category[spatial\_network$altitude>0 & spatial\_network$altitude<50]<-"Low"  
spatial\_network$altitude\_category[spatial\_network$altitude>50 & spatial\_network$altitude<100]<-"High"  
spatial\_network$altitude\_category[spatial\_network$altitude>100]<-"Very High"  
  
#Analysis of Category of altitudes  
altitude\_table<-as.data.frame(table(spatial\_network$altitude\_category))  
colnames(altitude\_table)<-c("Category","Freq")  
ggplot(altitude\_table,aes(x=Category,y=Freq,fill="Category"))+geom\_bar(stat = "identity",position = "dodge",fill="violet")+geom\_text(aes(label=Freq),size=5)



# Plotting on the Google Maps  
library(googleVis)  
spatial\_network$latlon <- paste(spatial\_network$latitude, spatial\_network$longitude,   
 sep = ":")  
map <- gvisMap(spatial\_network, locationvar = "latlon", tipvar = "altitude\_category",   
 options = list(showTip = T, showLine = F, enableScrollWheel = TRUE, useMapTypeControl = TRUE,   
 width = 1400, height = 800, allowHtml = TRUE))  
plot(map)

#Plotting using plotGoogleMaps  
library(plotGoogleMaps)

## Loading required package: sp  
## Loading required package: spacetime

spatial2<-spatial\_network[1:50000,]  
coordinates(spatial2)<- ~ longitude+latitude  
proj4string(spatial2) = CRS("+proj=longlat +datum=WGS84")  
ic<-iconlabels(attribute = spatial\_network$altitude\_category,colPalette = rainbow(4),icon=TRUE,at=NULL,height=10,scale=0.6)  
spatial3<-SpatialPointsDataFrame(spatial2,data = data.frame(ID=row.names(spatial2)))  
m<-plotGoogleMaps(spatial3,filename = "myMap1.html",iconMarker = ic)