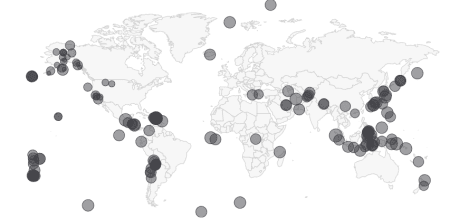
**Map Bubble**Map bubble is type of map chart where **bubble or circle position  indicates geoghraphical location** and **bubble size is used to show differences in magnitude of quantitative variables** like population.

We will be using Highcharter package to show earthquake magnitude and depth . Highcharter is a versatile charting library to build interactive charts, one of the easiest to learn and for shiny integration.

Bubble Map

**About dataset**Dataset used here is from [US Geological survey](https://earthquake.usgs.gov/earthquakes/feed/v1.0/csv.php) website of recent one week earthquake events. There are about 420 recorded observation with magnitude more than 2.0 globally. Dataset has 22 variables, of which we will be using *time, latitude, longitude, depth, magnitude(mag) and nearest named place of event.*

**Shiny Application**This application has single *app.R* file and *earthquake* dataset. Before we start with UI function, we will load dataset  and fetch *world* json object from highcharts map collection with *hcmap* function.

library(shiny)

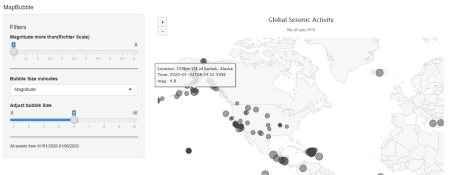
library(highcharter)

library(dplyr)

edata <- read.csv('earthquake.csv') %>% rename(lat=latitude,lon = longitude)

wmap <- hcmap()

Using dplyr package latitude and longitude variables are renamed as *lat* and *lon* with *rename* verb. Column names are important.



**ui**It has sidebar panel with 3 widgets and main panel for displaying map.

* Two *sliders*, one for filtering out low magnitude values and other for adjusting bubble  size.
* One *select* widget for bubble size variable. User can select magnitude or depth of earthquake event. *mag* and *depth* are columns in dataset.
* Widget output function *highchartOutput* for use in shiny.

ui <- fluidPage(

titlePanel("MapBubble"), # Application title

sidebarLayout(

sidebarPanel(

   sliderInput('mag','Magnitude more than(Richter Scale)', min = 1,max = 6,step = 0.5,value = 0),

   selectInput('bubble','Bubble Size indicates',choices = c('Magnitude'= 'mag','Depth(in Km)' = 'depth')),

   sliderInput('bublesize','Adjust bubble Size',min = 2,max = 10,step = 1,value = 6)

        ),

      # Display a Map Bubble

      mainPanel(

        highchartOutput('eqmap',height = "500px")

      )

   )

)

**Server**Before rendering, we will filter the dataset within *reactive* context. Any numeric column that we want to indicate with bubble size should be named *z. input$bubble* comes from *select* widget.

*renderHighchart* is render function for use in shiny. We will pass the filtered data and chart type as *mapbubble* in *hc\_add\_series* function. *Place, time* and *z* variable are displayed in the tooltip with “*point*” format.    
Sub-title is used to show no. of  filtered observation  in the map

  server <- function(input, output) {

  data <- reactive(edata %>%

               filter(mag >= input$mag) %>%

               rename(z = input$bubble))

output$eqmap <-renderHighchart(

wmap %>% hc\_legend(enabled = F) %>%

  hc\_add\_series(data = data(), type = "mapbubble", name = "", maxSize = paste0(input$bublesize,'%')) %>% #bubble size in perc %

hc\_tooltip(useHTML = T,headerFormat='',pointFormat = paste('Location :{point.place}

Time: {point.time}

',input$bubble,': {point.z}')) %>%

hc\_title(text = "Global Seismic Activity") %>%

hc\_subtitle(text = paste('No of obs:', nrow(data()),sep = '')) %>%

hc\_mapNavigation(enabled = T)%>%

)

}

# Run the application

shinyApp(ui = ui, server = server)