# Hurricane - Storm Path

*Enqun Wang*

*April 21, 2016*

**What areas were along the storm path?**

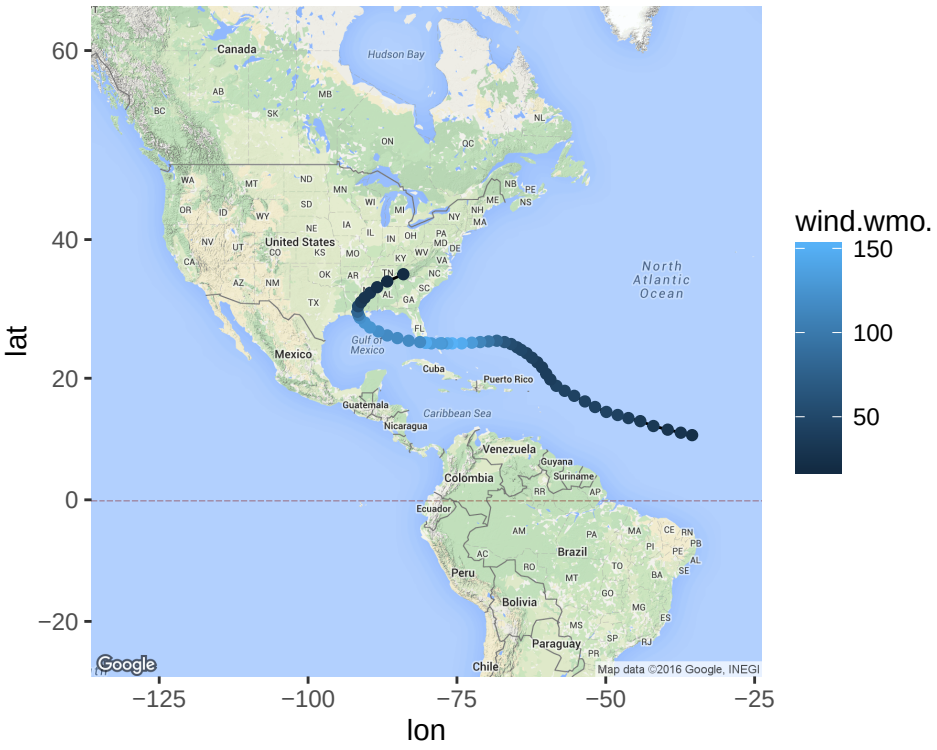
### Storm Path Function

EW: This a function to plot the storm path. You need to know the `name`, `year`, and `location` of the storm you would like to analyze. The function provides two types of map: the whole map or the local map.

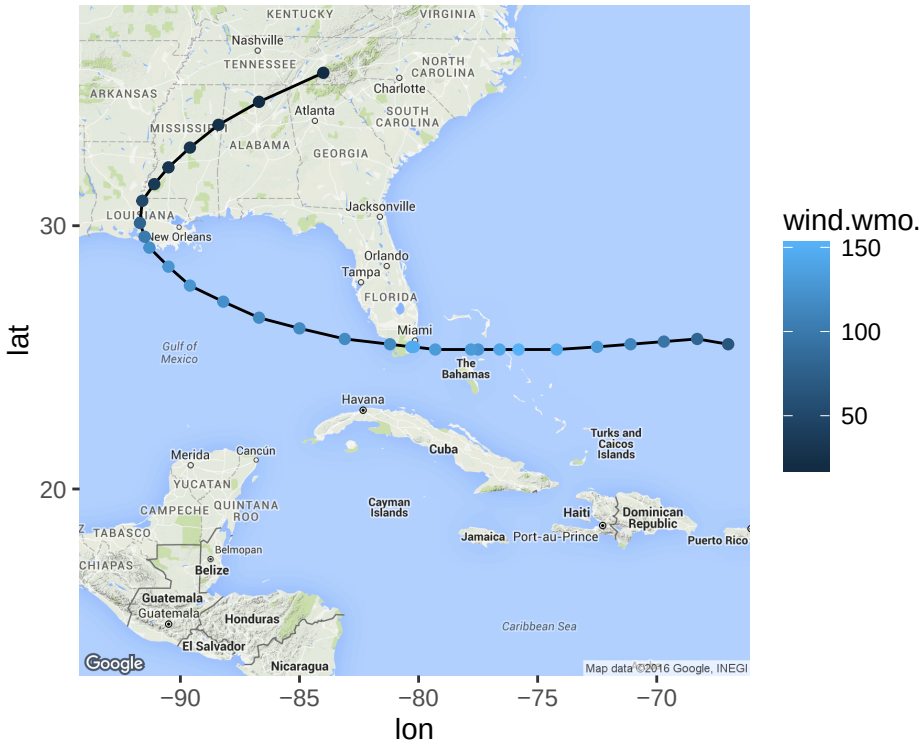
```
require(rnoaa)
require(dplyr)
require(ggplot2)
require(ggmap)

storm_path <- function(name, year, location, local = TRUE){
  # name: name of the storm
  # year: year of the storm
  # location: location of the storm
  # local: if true, plot the local map; else, plot the whole plot

  hurrs_year <- storm_data(year = year)
  storm <- hurrs_year$data[which(hurrs_year$data$name == name), ]
  if(sum(storm$latitude == -999)){
    storm <- storm[-c(which(storm$latitude == -999)),]
  }
  storm_loc <- select(storm, iso_time, latitude, longitude, wind.wmo.)

  if(local){
    map <- get_map(location = location, zoom = 5)
  }else{
    map <- get_map(location = location, zoom = 3, source='google')
  }
  ggmap(map) +
    geom_path(data = storm_loc, aes(x = longitude, y = latitude)) +
    geom_point(data = storm_loc, aes(x = longitude, y = latitude,
                                     color = wind.wmo.))
}
```

**Hurricane Andrew:**

```
storm_path(name = "ANDREW", year = 1992, location = "Miami", local = F)
```



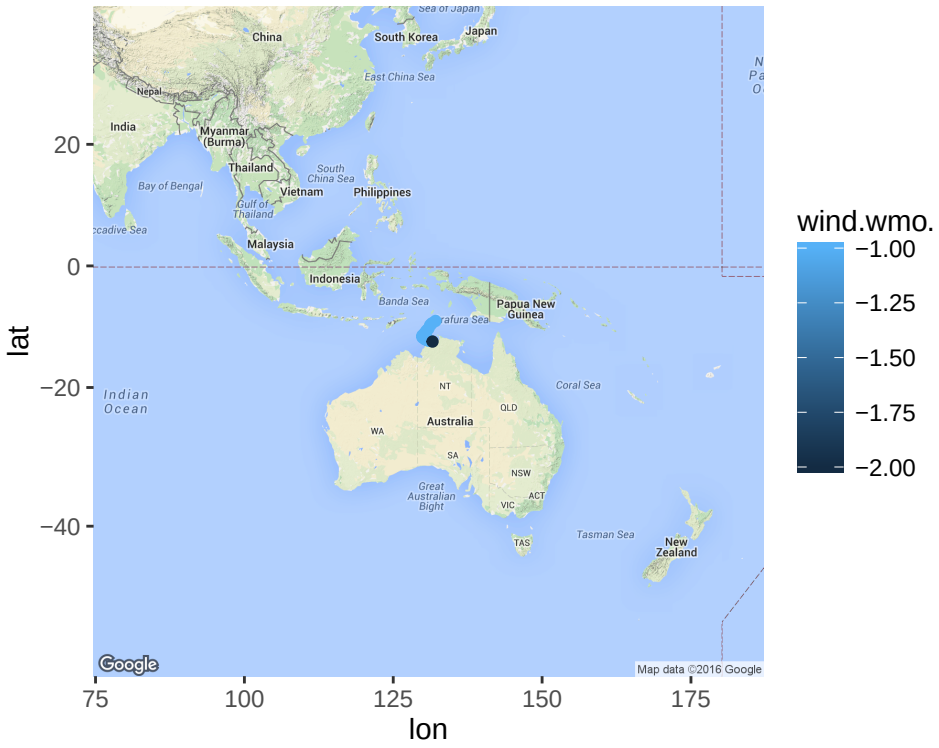

```
storm_path(name = "ANDREW", year = 1992, location = "Miami", local = T)
```



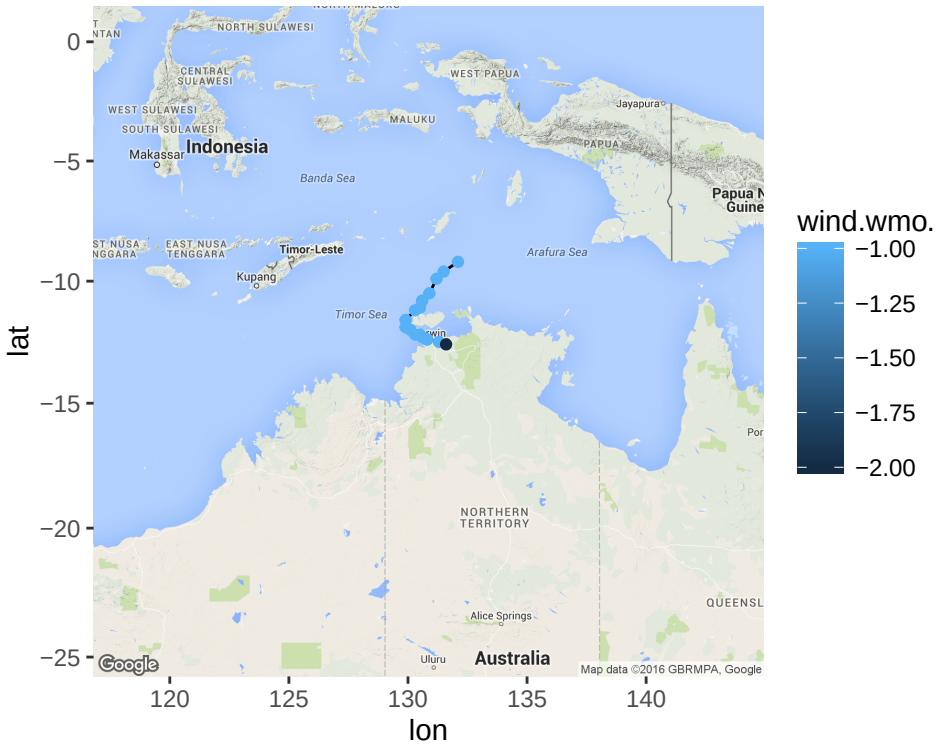
**Labor Day Storm:** cannot find the data.

**Cyclone Tracy:**

```
storm_path(name = "07S:TRACY", year = 1975, location = "Darwin", local = F)
```

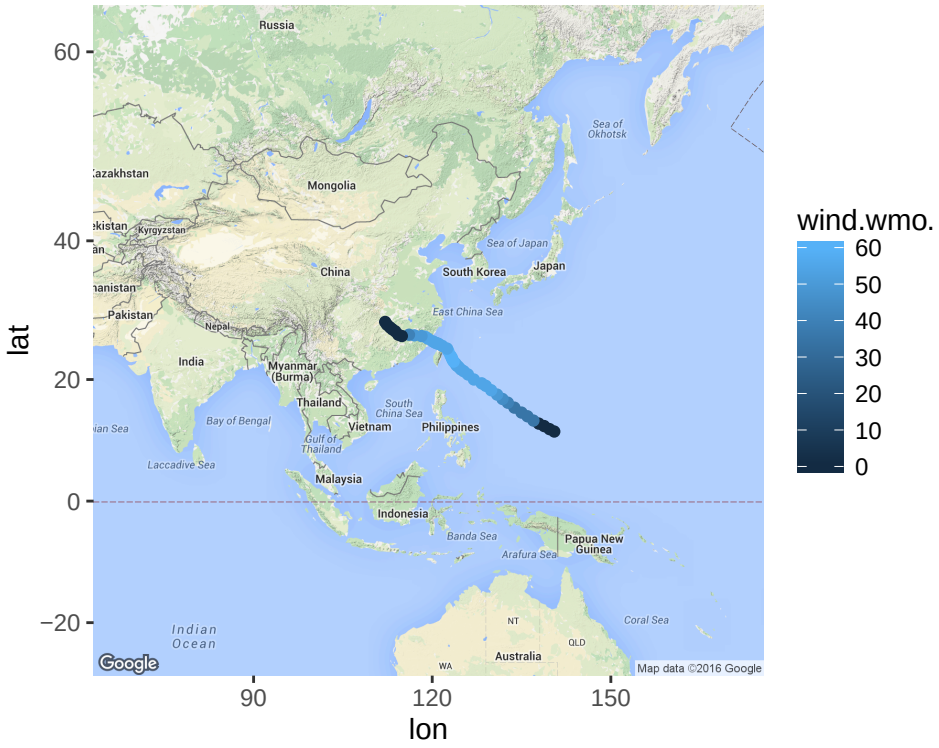



```
storm_path(name = "07S:TRACY", year = 1975, location = "Darwin", local = T)
```




**Tropical Storm Bilis:**

```
storm_path(name = "BILIS", year = 2006, location = "Fuzhou", local = F)
```



```
storm_path(name = "BILIS", year = 2006, location = "Fuzhou", local = T)
```