

Haversine formula/ XY Conversion

Distance between two points

```
custom_hav_dist <- function(lat1, lon1, lat2, lon2) {  
  R <- 6371  
  Radian_factor <- 0.0174533  
  lat_1 <- (90 - lat1) * Radian_factor  
  lat_2 <- (90 - lat2) * Radian_factor  
  diff_long <- (lon1 - lon2) * Radian_factor  
  
  distance_in_km <- 6356.137 * acos((cos(lat_1) * cos(lat_2)) + (sin(lat_1) * sin(lat_2) * cos(diff_long)))  
  rm(lat1, lon1, lat2, lon2)  
  return(distance_in_km * 10^3)  
}
```

Conversion

To convert latitudinal and longitudinal coordinates into Kilo-meters first you must convert those coordinates into radians. Then we compute the x and y in meters.

```
convq <- function(lat, long) {  
  # Convert Lat Long to Radians  
  lat1rad = lat/180 * 3.14159  
  long1rad = long/180 * 3.14159  
  # Convert (R,T,r) (Rho, Theta, radius) Polar coordinates  
  # into standard (x,y) coordinates  
  x = sin(lat1rad) * cos(long1rad) * 6356.137 * 10^3 # Rad is 6356.752 at poles  
  y = sin(lat1rad) * sin(long1rad) * 6356.137 * 10^3  
  return(list(x, y))  
  # Calc straight line distance between 2 points  
  # straightLineDist = sqrt((x2 - x1) ^ 2 + (y2 - y1) ^ 2)  
  # Use law of cosines to calculate arc distance  
  # arcDist = acos(1 - (straightLineDist ^ 2 / (2 * 6356.137 * 10^3 ^ 2))) * 6356.137 * 10^3  
  # return arc distance  
  # return(arcDist)  
}
```

```
qq1 <- convq(46.07151828435007, 11.12053317266584)  
qq1<- unlist(qq1)
```

```
qq2 <- convq(46.071618166286214, 11.120960037103822)  
qq2<- unlist(qq2)
```

```
(qq <- qq2-qq1)
```

```
## [1] 0.9647629 34.9472237
```

```
custom_hav_dist(46.07151828435007, 11.12053317266584,46.071618166286214, 11.120960037103822 )
```

```
## [1] 34.67082
```