

Converting geographic coordinate system in R

<https://gis.stackexchange.com/questions/45263/converting-geographic-coordinate-system-in-r>

```
# References:
# http://lists.maptools.org/pipermail/proj/2001-September/000248.html (has typos)
# http://www.remotesensing.org/geotiff/proj\_list/swiss\_oblique\_cylindrical.html
#
# Input coordinates.
#
x <- c(7.173500, 7.172540, 7.171636, 7.180180, 7.178070, 7.177229, 7.175240,
       7.181409, 7.179299)
y <- c(45.86880, 45.86887, 45.86924, 45.87158, 45.87014, 45.86923, 45.86808,
       45.87177, 45.87020)

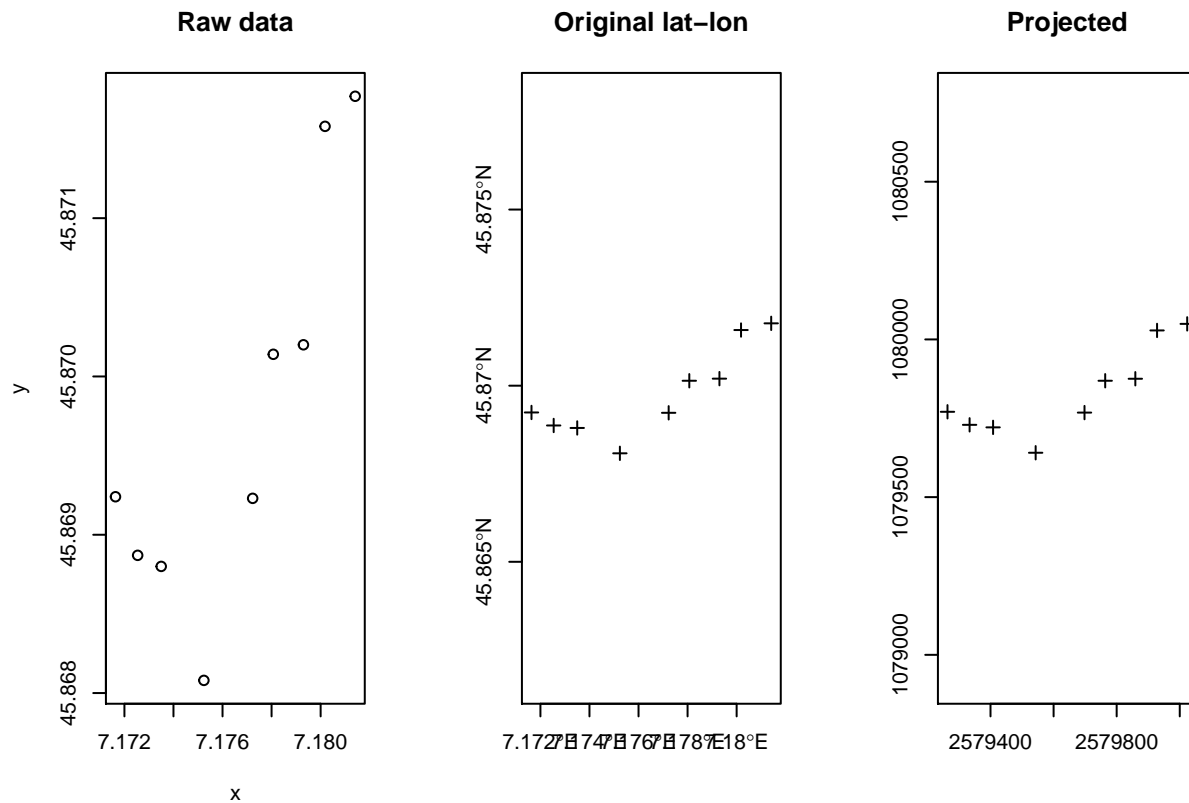
#
# Define the coordinate systems.
#
library(rgdal)
#> Loading required package: sp
#> rgdal: version: 1.3-4, (SVN revision 766)
#> Geospatial Data Abstraction Library extensions to R successfully loaded
#> Loaded GDAL runtime: GDAL 2.2.3, released 2017/11/20
#> Path to GDAL shared files: C:/Users/msfz751/Documents/R/win-library/3.5/rgdal/gdal
#> GDAL binary built with GEOS: TRUE
#> Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ_VERSION: 493]
#> Path to PROJ.4 shared files: C:/Users/msfz751/Documents/R/win-library/3.5/rgdal/proj
#> Linking to sp version: 1.3-1
d <- data.frame(lon=x, lat=y)

coordinates(d) <- c("lon", "lat")
proj4string(d) <- CRS("+init=epsg:4326") # WGS 84

CRS.new <- CRS("+proj=somerc +lat_0=46.9524056 +lon_0=7.43958333 +ellps=bessel +x_0=2600000 +y_0=1200000")

# (@mdsummer points out that
#   CRS.new <- CRS("+init=epsg:2056")
# will work, and indeed it does. See http://spatialreference.org/ref/epsg/2056/proj4/.)
d.ch1903 <- spTransform(d, CRS.new)

# Plot the results.
#
par(mfrow=c(1,3))
plot.default(x,y, main="Raw data", cex.axis=.95)
plot(d, axes=TRUE, main="Original lat-lon", cex.axis=.95)
plot(d.ch1903, axes=TRUE, main="Projected", cex.axis=.95)
```



```

unclass(d.ch1903)
#> <S4 Type Object>
#> attr(,"bbox")
#>      min      max
#> lon 2579264 2580024
#> lat 1079641 1080049
#> attr(,"proj4string")
#> CRS arguments:
#> +proj=somerc +lat_0=46.9524056 +lon_0=7.43958333 +ellps=bessel
#> +x_0=2600000 +y_0=1200000 +towgs84=674.374,15.056,405.346 +units=m
#> +k_0=1 +no_defs
#> attr(,"coords")
#>      lon      lat
#> [1,] 2579408 1079721
#> [2,] 2579334 1079729
#> [3,] 2579264 1079771
#> [4,] 2579928 1080028
#> [5,] 2579764 1079869
#> [6,] 2579698 1079768
#> [7,] 2579543 1079641
#> [8,] 2580024 1080049
#> [9,] 2579859 1079875

```

```

library(rgdal)
sputm <- SpatialPoints(randompoints, proj4string=CRS("+proj=utm +zone=47 +datum=WGS84"))
spgeo <- spTransform(sputm, CRS("+proj=longlat +datum=WGS84"))

```

```
library(rgdal)

# Make a two-column matrix, col1 = long, col2 = lat
xy <- cbind(c(107), c(26))
# Convert it to UTM coordinates (in units of meters)
project(xy, "+proj=utm +zone=51 ellps=WGS84")
```

```
library(rgdal)

# Make a two-column matrix, col1 = long, col2 = lat
xy <- cbind(c(118, 119), c(10, 50))

# Convert it to UTM coordinates (in units of meters)
project(xy, "+proj=utm +zone=51 ellps=WGS84")

#           [,1]      [,2]
# [1,] -48636.65 1109577
# [2,] 213372.05 5546301
```