Practica 3

Viacheslav Shalisko

3/9/2021

Objetivo: Realizar un analisis de datos sobre temperatura dunte un mes del año en México y constuir un modelo de relación entre temperatura y variable de elevación sobre nivel del mar. Aplicar el modelo para contruir una predicción de patron de temperaturas.

1) Cargar bibliotecas requeridas: sp, raster, rgdal, foreign

```
library(sp)
library(raster)
library(rgdal)
```

```
## rgdal: version: 1.5-23, (SVN revision 1121)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 3.2.1, released 2020/12/29
## Path to GDAL shared files: C:/Users/vshal/Documents/R/win-library/4.1/rgdal/gdal
## GDAL binary built with GEOS: TRUE
## Loaded PROJ runtime: Rel. 7.2.1, January 1st, 2021, [PJ_VERSION: 721]
## Path to PROJ shared files: C:/Users/vshal/Documents/R/win-library/4.1/rgdal/proj
## PROJ CDN enabled: FALSE
## Linking to sp version:1.4-5
## To mute warnings of possible GDAL/OSR exportToProj4() degradation,
## use options("rgdal_show_exportToProj4_warnings"="none") before loading rgdal.
## Overwritten PROJ_LIB was C:/Users/vshal/Documents/R/win-library/4.1/rgdal/proj
```

```
library(foreign)

sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19043)
## 
## Matrix products: default
## 
## locale:
## [1] LC_COLLATE=Spanish_Mexico.1252  LC_CTYPE=Spanish_Mexico.1252   
## [3] LC_MONETARY=Spanish_Mexico.1252 LC_NUMERIC=C                   
## [5] LC_TIME=Spanish_Mexico.1252    
## 
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base     
## 
## other attached packages:
## [1] foreign_0.8-81 rgdal_1.5-23   raster_3.4-13  sp_1.4-5      
## 
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.7        codetools_0.2-18 lattice_0.20-44  digest_0.6.27   
##  [5] grid_4.1.1        R6_2.5.1         jsonlite_1.7.2   magrittr_2.0.1  
##  [9] evaluate_0.14     rlang_0.4.11     stringi_1.7.4    jquerylib_0.1.4 
## [13] bslib_0.3.0       rmarkdown_2.10   tools_4.1.1      stringr_1.4.0   
## [17] xfun_0.25         yaml_2.2.1       fastmap_1.1.0    compiler_4.1.1  
## [21] htmltools_0.5.2  knitr_1.33       sass_0.4.0
```

2) Realizar la lectura de datos fuente en formato DBF

```
tabla_temp <- read.dbf("Practica_3_insumos/Temp.dbf")

head(tabla_temp)
```

```
##   NUMERO_DE_ TENE TFEB TMAR TABR TMAY TJUN TJUL TAGO TSEP TOCT TNOV TDIC TMEDIA
## 1   00001003 15.4 16.7 19.2 21.7 23.7 24.0 22.0 22.1 22.2 21.0 18.5 16.3   20.2
## 2   00001004 11.8 13.1 15.6 18.0 19.9 20.7 19.4 19.2 18.6 16.7 14.2 12.5   16.6
## 3   00001005 13.4 14.7 17.2 19.5 21.9 22.6 21.0 20.7 20.2 18.6 16.2 14.2   18.4
## 4   00001006 12.6 13.8 16.7 19.5 21.7 22.0 20.5 20.3 19.8 18.0 15.1 13.4   17.8
## 5   00001007 12.2 13.4 16.4 18.8 21.0 21.4 20.0 19.6 19.5 17.5 14.8 13.1   17.3
## 6   00001008 13.0 13.8 16.0 18.1 20.0 19.5 18.3 18.3 17.6 16.4 14.9 13.4   16.6
```

3) Realizar la lectura de datos fuente en formato SHP

```
estaciones_smn <- readOGR("Practica_3_insumos/Estaciones_SMN_alt_ok.shp")
```

```
## OGR data source with driver: ESRI Shapefile 
## Source: "D:\GD\UdeG_Docencia\Metadatos\Semestre_2021B\Practica_3\Practica_3_insumos\Estaciones_SMN_alt_ok.shp", layer: "E
staciones_SMN_alt_ok"
## with 3089 features
## It has 10 fields
```

```
estaciones_smn
```

```
## class       : SpatialPointsDataFrame 
## features    : 3089 
## extent      : -117.0472, -86.82028, 14.61778, 32.665  (xmin, xmax, ymin, ymax)
## crs         : +proj=longlat +datum=WGS84 +no_defs 
## variables   : 10
## names       : OBJECTID, NUMERO_DE_,             NOMBRE_DE_,    Z, CVE_ENT,  Z_DELTA,   Z_MDE,  Z_MDE1, Z_DELTA1,   Z_FIN 
## min values  :        1,   00001003,          ?ADO - ACULCO,    0,      01, -382.393,     -14,     -14, -382.393,     -14 
## max values  :     3089,   00032187, ZUMPANGO DEL RIO (SMN), 4110,      32,  313.404, 4297.51, 4297.51,  313.404, 4297.51
```

4) Realizar la lectura de datos fuente en formato TIFF

```
raster_mde <- raster("Practica_3_insumos/MDE_occidente_500m_float_GEO.tif")

raster_mde
```

```
## class      : RasterLayer 
## dimensions : 1293, 1324, 1711932  (nrow, ncol, ncell)
## resolution : 0.004709448, 0.004709448  (x, y)
## extent     : -107.0781, -100.8428, 16.96142, 23.05073  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs 
## source     : MDE_occidente_500m_float_GEO.tif 
## names      : MDE_occidente_500m_float_GEO 
## values     : 0, 4096  (min, max)
```

5) Verificar contenido de datos fuente

```
str(tabla_temp)
```

```
## 'data.frame':    3724 obs. of  14 variables:
##  $ NUMERO_DE_: Factor w/ 3724 levels "00001003","00001004",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ TENE      : num  15.4 11.8 13.4 12.6 12.2 13 0 10.3 14.9 13.9 ...
##  $ TFEB      : num  16.7 13.1 14.7 13.8 13.4 13.8 0 11.4 15.9 15.2 ...
##  $ TMAR      : num  19.2 15.6 17.2 16.7 16.4 16 0 13.7 18.2 17.4 ...
##  $ TABR      : num  21.7 18 19.5 19.5 18.8 18.1 0 16.4 20.4 20 ...
##  $ TMAY      : num  23.7 19.9 21.9 21.7 21 20 0 18.2 22.6 22.8 ...
##  $ TJUN      : num  24 20.7 22.6 22 21.4 19.5 0 18 22.6 23.8 ...
##  $ TJUL      : num  22 19.4 21 20.5 20 18.3 0 16.1 21 21.9 ...
##  $ TAGO      : num  22.1 19.2 20.7 20.3 19.6 18.3 0 16 20.8 21.5 ...
##  $ TSEP      : num  22.2 18.6 20.2 19.8 19.5 17.6 0 15.3 20.4 21.3 ...
##  $ TOCT      : num  21 16.7 18.6 18 17.5 16.4 0 14.2 19.3 19.8 ...
##  $ TNOV      : num  18.5 14.2 16.2 15.1 14.8 14.9 0 12.5 17.1 16.8 ...
##  $ TDIC      : num  16.3 12.5 14.2 13.4 13.1 13.4 0 10.9 15.6 14.8 ...
##  $ TMEDIA    : num  20.2 16.6 18.4 17.8 17.3 16.6 0 14.4 19.1 19.1 ...
##  - attr(*, "data_types")= chr [1:14] "C" "N" "N" "N" ...
```

```
tabla_temp$NUMERO_DE_ <- as.character(tabla_temp$NUMERO_DE_)
str(tabla_temp)
```

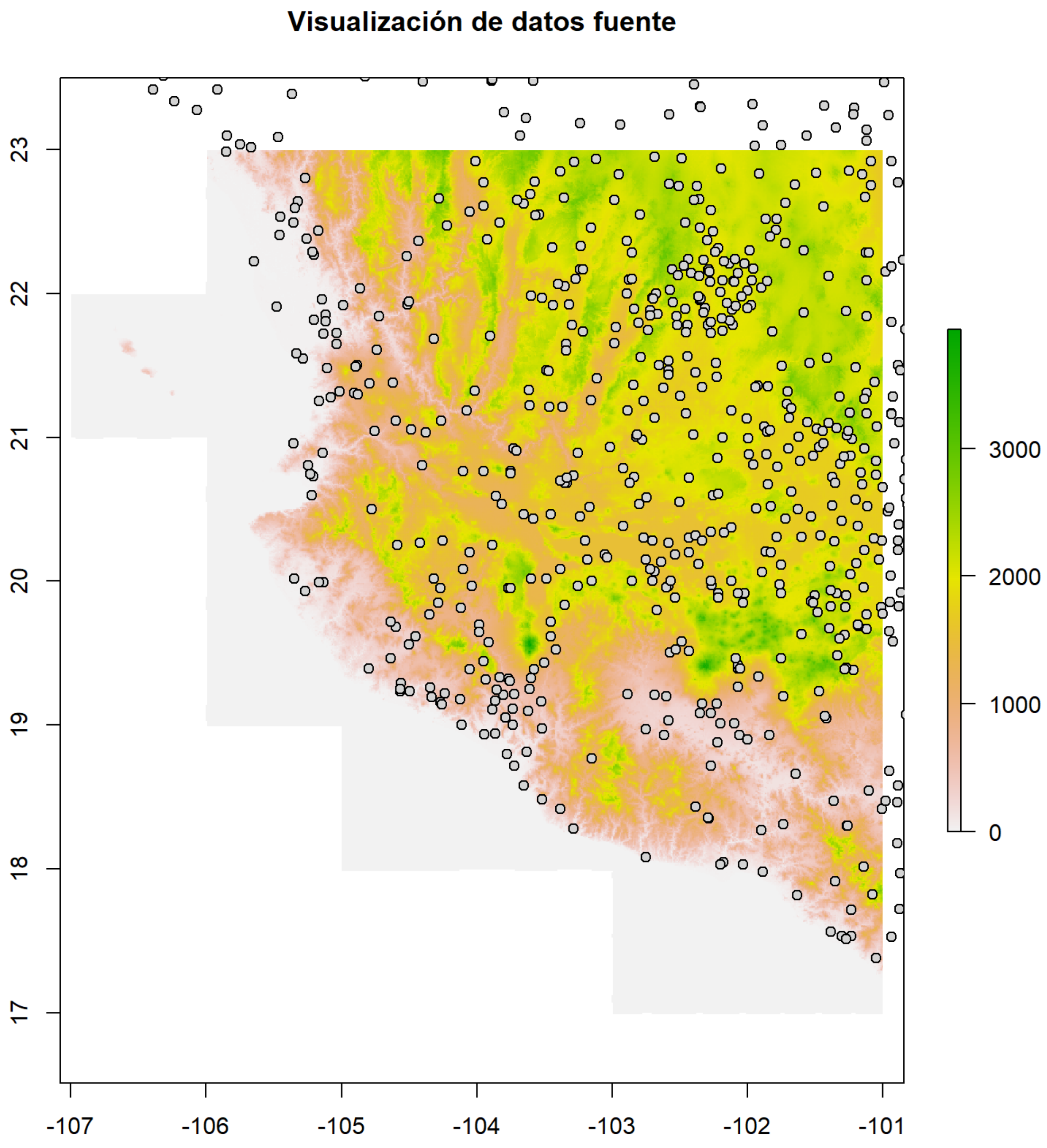
```
## 'data.frame':    3724 obs. of  14 variables:
##  $ NUMERO_DE_: chr  "00001003" "00001004" "00001005" "00001006" ...
##  $ TENE      : num  15.4 11.8 13.4 12.6 12.2 13 0 10.3 14.9 13.9 ...
##  $ TFEB      : num  16.7 13.1 14.7 13.8 13.4 13.8 0 11.4 15.9 15.2 ...
##  $ TMAR      : num  19.2 15.6 17.2 16.7 16.4 16 0 13.7 18.2 17.4 ...
##  $ TABR      : num  21.7 18 19.5 19.5 18.8 18.1 0 16.4 20.4 20 ...
##  $ TMAY      : num  23.7 19.9 21.9 21.7 21 20 0 18.2 22.6 22.8 ...
##  $ TJUN      : num  24 20.7 22.6 22 21.4 19.5 0 18 22.6 23.8 ...
##  $ TJUL      : num  22 19.4 21 20.5 20 18.3 0 16.1 21 21.9 ...
##  $ TAGO      : num  22.1 19.2 20.7 20.3 19.6 18.3 0 16 20.8 21.5 ...
##  $ TSEP      : num  22.2 18.6 20.2 19.8 19.5 17.6 0 15.3 20.4 21.3 ...
##  $ TOCT      : num  21 16.7 18.6 18 17.5 16.4 0 14.2 19.3 19.8 ...
##  $ TNOV      : num  18.5 14.2 16.2 15.1 14.8 14.9 0 12.5 17.1 16.8 ...
##  $ TDIC      : num  16.3 12.5 14.2 13.4 13.1 13.4 0 10.9 15.6 14.8 ...
##  $ TMEDIA    : num  20.2 16.6 18.4 17.8 17.3 16.6 0 14.4 19.1 19.1 ...
##  - attr(*, "data_types")= chr [1:14] "C" "N" "N" "N" ...
```

```
str(estaciones_smn@data)
```

```
## 'data.frame':    3089 obs. of  10 variables:
##  $ OBJECTID  : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ NUMERO_DE_: chr  "00001003" "00001004" "00001005" "00001006" ...
##  $ NOMBRE_DE_: chr  "CALVILLO - CALVILLO" "CA?ADA HONDA - AGUASCALIE" "PRESA EL NIAGARA - AGUASC" "EL TULE - ASIENTOS"
...
##  $ Z         : int  1702 1925 1828 1970 1800 2300 1600 2425 1775 1585 ...
##  $ CVE_ENT   : chr  "01" "01" "01" "01" ...
##  $ Z_DELTA   : num  -0.666 8.635 -75.76 -30.882 -85.542 ...
##  $ Z_MDE     : num  1703 1916 1904 2001 1886 ...
##  $ Z_MDE1    : num  1703 1916 1904 2001 1886 ...
##  $ Z_DELTA1  : num  -0.666 8.635 -75.76 -30.882 -85.542 ...
##  $ Z_FIN     : num  1703 1916 1904 2001 1886 ...
```

6) Visualizar datos fuente en forma geoespacial

```
plot(raster_mde, main = "Visualización de datos fuente")
plot(estaciones_smn, pch = 21,
     col = "black", bg = "lightgray", add = TRUE)
```



7) Realizar JOIN de temperaturas en tabla DBF y puntos de observación SHP

```
smn_temp <- merge(x = estaciones_smn@data, y = tabla_temp,
                  by = "NUMERO_DE_")

head(smn_temp)
```

```
##   NUMERO_DE_ OBJECTID                NOMBRE_DE_    Z CVE_ENT    Z_DELTA   Z_MDE
## 1   00001003        1       CALVILLO - CALVILLO 1702      01  -0.666382 1702.67
## 2   00001004        2 CA?ADA HONDA - AGUASCALIE 1925      01   8.635380 1916.36
## 3   00001005        3 PRESA EL NIAGARA - AGUASC 1828      01 -75.760100 1903.76
## 4   00001006        4        EL TULE - ASIENTOS 1970      01 -30.881600 2000.88
## 5   00001007        5 JESUS MARIA - JESUS MARIA 1800      01 -85.542100 1885.54
## 6   00001008        6 PUERTO DE LA CONCEPCION - 2300      01 -22.656500 2322.66
##    Z_MDE1   Z_DELTA1   Z_FIN TENE TFEB TMAR TABR TMAY TJUN TJUL TAGO TSEP TOCT
## 1 1702.67  -0.666382 1702.67 15.4 16.7 19.2 21.7 23.7 24.0 22.0 22.1 22.2 21.0
## 2 1916.36   8.635380 1916.36 11.8 13.1 15.6 18.0 19.9 20.7 19.4 19.2 18.6 16.7
## 3 1903.76 -75.760100 1903.76 13.4 14.7 17.2 19.5 21.9 22.6 21.0 20.7 20.2 18.6
## 4 2000.88 -30.881600 2000.88 12.6 13.8 16.7 19.5 21.7 22.0 20.5 20.3 19.8 18.0
## 5 1885.54 -85.542100 1885.54 12.2 13.4 16.4 18.8 21.0 21.4 20.0 19.6 19.5 17.5
## 6 2322.66 -22.656500 2322.66 13.0 13.8 16.0 18.1 20.0 19.5 18.3 18.3 17.6 16.4
##   TNOV TDIC TMEDIA
## 1 18.5 16.3   20.2
## 2 14.2 12.5   16.6
## 3 16.2 14.2   18.4
## 4 15.1 13.4   17.8
## 5 14.8 13.1   17.3
## 6 14.9 13.4   16.6
```

```
dim(smn_temp)
```

```
## [1] 3089   23
```

```
summary(smn_temp$TMEDIA)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00   16.80   20.60   19.75   24.50   29.70
```

8) Preparar variables para el modelo lineal

```
selector_valores <- (smn_temp$TMEDIA != 0) & (smn_temp$Z_FIN > 0)

summary(selector_valores)
```

```
##    Mode   FALSE    TRUE 
## logical     163    2926
```

```
temperatura <- smn_temp[selector_valores,"TSEP"]

summary(temperatura)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    4.20   18.80   23.50   22.71   27.00   32.60
```

```
length(temperatura)
```

```
## [1] 2926
```

```
elevacion <- smn_temp[selector_valores,"Z_FIN"]
elevacion_2 <- elevacion ^ 2

summary(elevacion)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##    0.051  142.309  960.114 1068.659 1899.405 4297.510
```

```
length(elevacion)
```

```
## [1] 2926
```

9) Construir un modelo lineal (LM)

10) Revisar propiedades del modelo construido

```
modelo_lm <- lm(temperatura ~ elevacion + elevacion_2)

summary(modelo_lm)
```

```
##
## Call:
## lm(formula = temperatura ~ elevacion + elevacion_2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5760  -0.7062   0.0446   0.8444   7.9939
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.768e+01  5.260e-02  526.32   <2e-16 ***
## elevacion   -3.557e-03  1.148e-04  -30.98   <2e-16 ***
## elevacion_2 -5.969e-07  4.608e-08  -12.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.564 on 2923 degrees of freedom
## Multiple R-squared:  0.8931, Adjusted R-squared:  0.8931
## F-statistic: 1.221e+04 on 2 and 2923 DF,  p-value: < 2.2e-16
```

```
coef(modelo_lm)
```

```
##   (Intercept)     elevacion   elevacion_2
##  2.768394e+01 -3.556515e-03 -5.969455e-07
```

```
confint(modelo_lm)
```

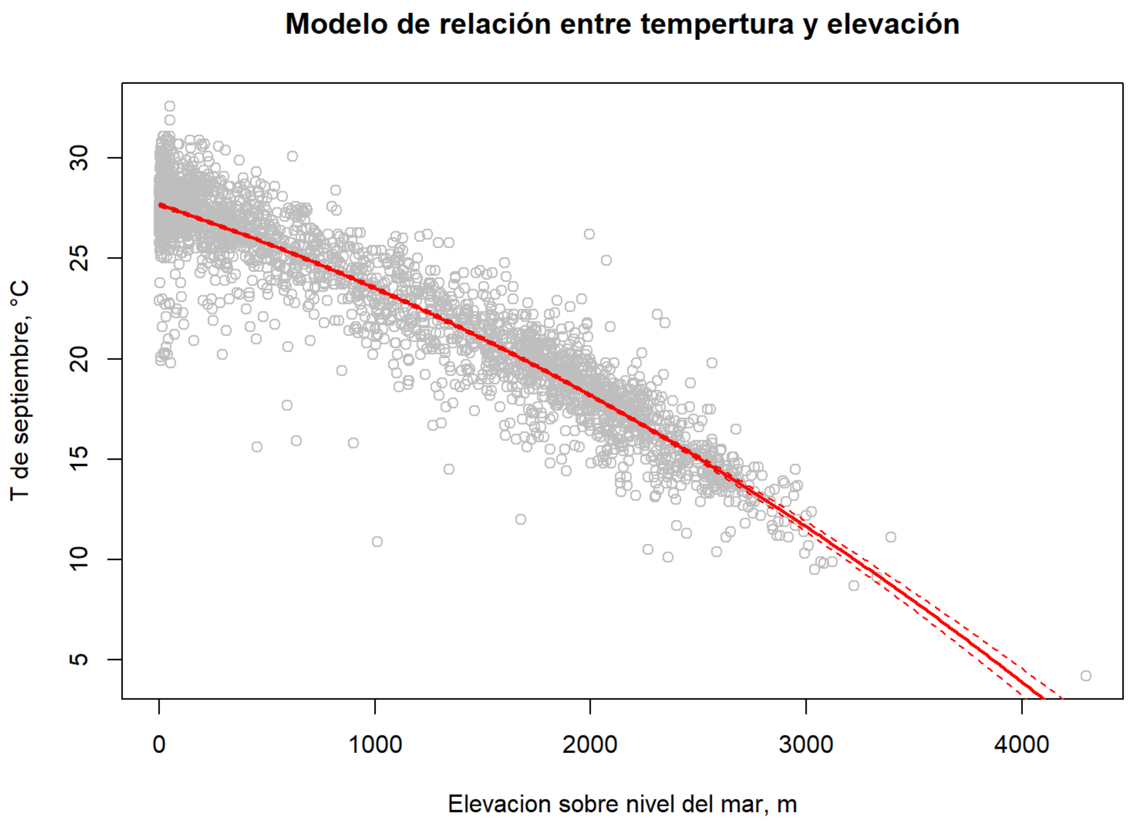
```
##                     2.5 %        97.5 %
## (Intercept)  2.758080e+01  2.778707e+01
## elevacion   -3.781613e-03 -3.331418e-03
## elevacion_2 -6.873054e-07 -5.065856e-07
```

11) Realizar visualización del modelo

```
plot(x = elevacion, y = temperatura, col = "gray",
     main = "Modelo de relación entre tempertura y elevación",
     ylab = "T de septiembre, °C", xlab = "Elevacion sobre nivel del mar, m")

elevacion_simulada <- seq(0, 4300, 10)
temperatura_simulada <- predict(modelo_lm, data.frame(elevacion = elevacion_simulada,
                                                      elevacion_2 = elevacion_simulada^2),
                                interval = c("confidence"))

str(temperatura_simulada)
```

```
##  num [1:431, 1:3] 27.7 27.6 27.6 27.6 27.5 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:431] "1" "2" "3" "4" ...
##   ..$ : chr [1:3] "fit" "lwr" "upr"
```

```
lines(x = elevacion_simulada, y = temperatura_simulada[,"fit"],
      col = "red", lty = 1, lwd = 2)
lines(x = elevacion_simulada, y = temperatura_simulada[,"upr"],
      col = "red", lty = 2, lwd = 1)
lines(x = elevacion_simulada, y = temperatura_simulada[,"lwr"],
      col = "red", lty = 2, lwd = 1)
```



12) Realizar predicción de las temperaturas en formato Raster

```
raster_mde2 <- raster_mde ^2
raster_mde_stack <- stack(raster_mde, raster_mde2)
names(raster_mde_stack) <- c("elevacion","elevacion_2")
raster_mde_stack
```

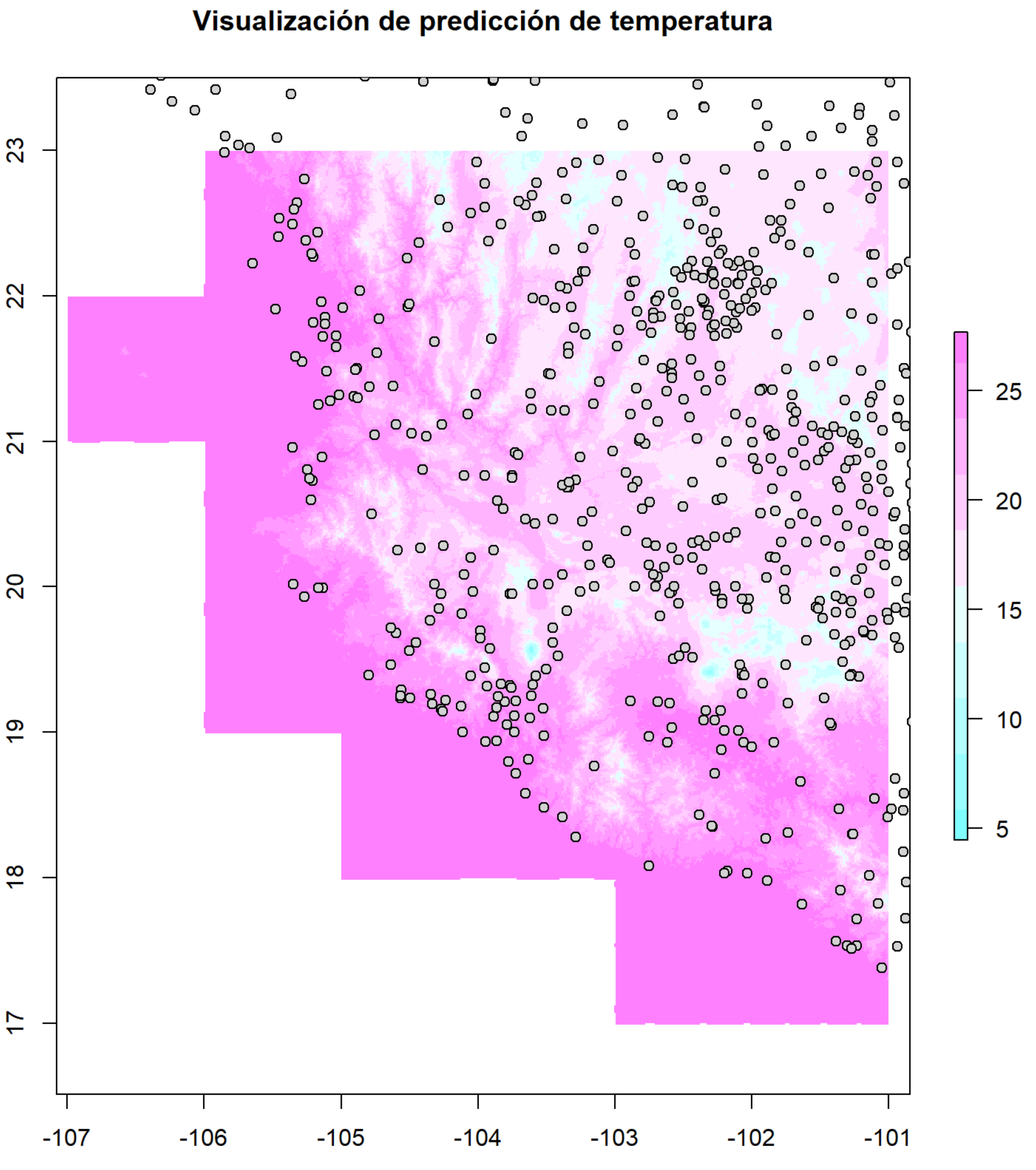
```
## class      : RasterStack
## dimensions : 1293, 1324, 1711932, 2  (nrow, ncol, ncell, nlayers)
## resolution : 0.004709448, 0.004709448  (x, y)
## extent     : -107.0781, -100.8428, 16.96142, 23.05073  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs
## names      : elevacion, elevacion_2
## min values :         0,           0
## max values :      4096,    16777216
```

```
temperatura_simulada_raster <- predict(raster_mde_stack, modelo_lm)
temperatura_simulada_raster
```

```
## class      : RasterLayer
## dimensions : 1293, 1324, 1711932  (nrow, ncol, ncell)
## resolution : 0.004709448, 0.004709448  (x, y)
## extent     : -107.0781, -100.8428, 16.96142, 23.05073  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs
## source     : memory
## names      : layer
## values     : 3.101365, 27.68394  (min, max)
```

13) Visualizar y guardar el la superficie resultante

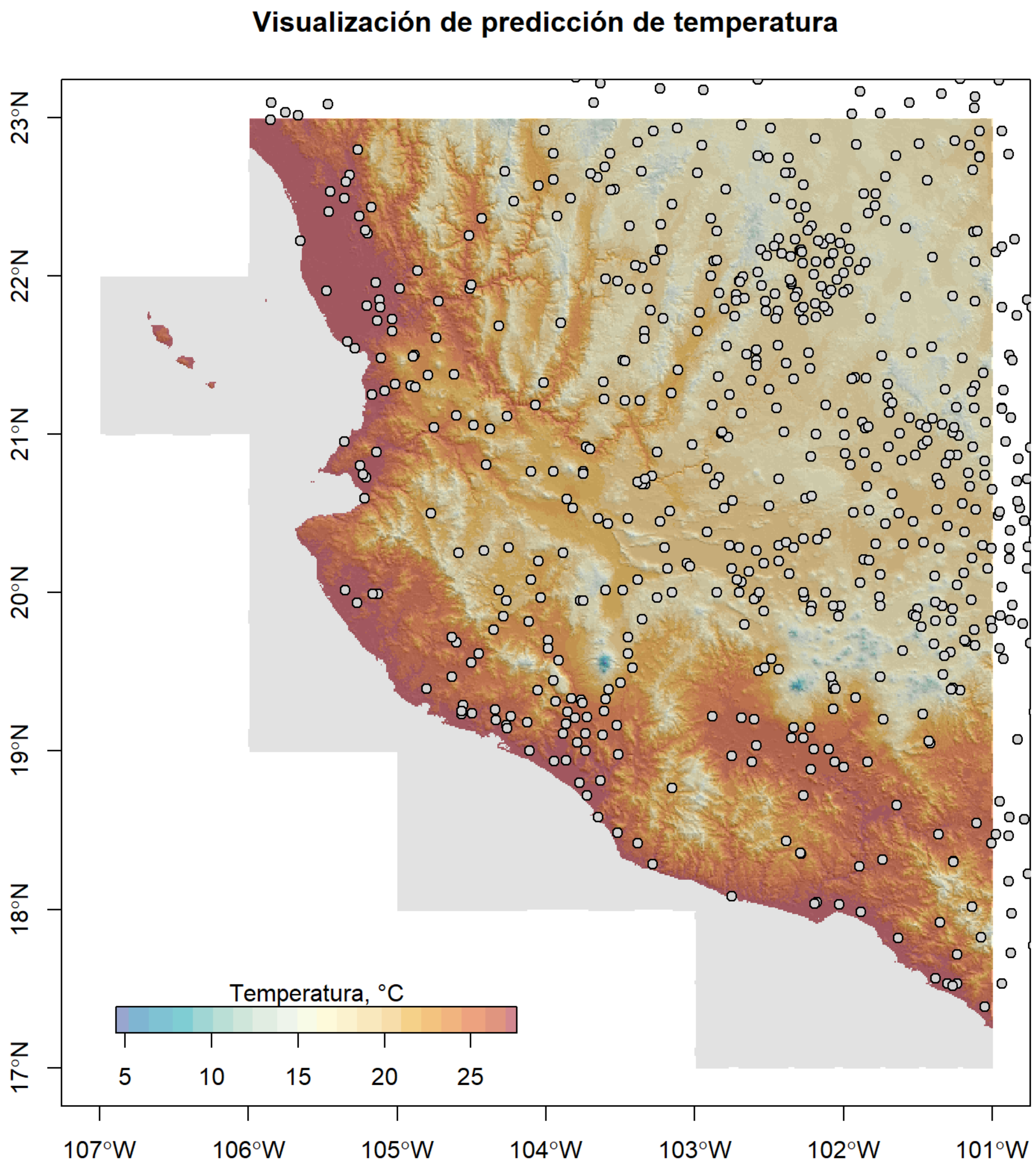
```
plot(temperatura_simulada_raster,
     col = cm.colors(10),
     main = "Visualización de predicción de temperatura")
plot(estaciones_smn, pch = 21,
     col = "black", bg = "lightgray", add = TRUE)
```



```
writeRaster(temperatura_simulada_raster, filename = "temperatura_septiembre.tif",
            format = "GTiff", datatype = "FLT4S", overwrite = TRUE)
```

14) Visualización de raster con relieve sombreado

```
## generar superficie de relieve sombreado
slope <- terrain(raster_mde, 'slope')
aspect <- terrain(raster_mde, 'aspect')
ethHS <- hillShade(slope=slope, aspect=aspect,
                       angle=20, direction=30)
m <- c(-Inf, 0, 1,  1, Inf, NA)
rclmat <- matrix(m, ncol = 3, byrow = TRUE)
ocean <- reclassify(raster_mde, rclmat)


## generar grafica de puntos para definición de área de visualización
plot(estaciones_smn, pch = 21,
     col = "black", bg = "lightgray",
     xlim = c(-107,-101), ylim = c(17,23),
     main = "Visualización de predicción de temperatura",
     axes = TRUE)

## agregar capa de relieve sombreado
mi_plot <- plot(ethHS, add = TRUE,
     col = hcl.colors(n = 20, pal = "Grays"),
     legend = FALSE)

## agregar predicción de temperatura en modo de capa semi-transpatente
plot(temperatura_simulada_raster, add = TRUE,
     legend = FALSE,
     col = rev(hcl.colors(n = 20, pal = "RdYlBu", alpha = 0.5)))

## agregar oceano
plot(ocean, add = TRUE,
     legend = FALSE,
     col = hcl.colors(n = 2, pal = "Blues 2"))

## colocar puntos por encima
plot(estaciones_smn, pch = 21,
     col = "black", bg = "lightgray", add = TRUE)

## agregar leyenda para predicción de temperatura
plot(temperatura_simulada_raster, add = TRUE,
     legend.only = TRUE, horizontal = TRUE,
     smallplot= c(0.15, 0.5, 0.17, 0.19),
     legend.args = list(text='Temperatura, °C'),
     col = rev(hcl.colors(n = 20, pal = "RdYlBu", alpha = 0.5)))
```


Visualización de predicción de temperatura
Temperatura, °C
5
10
15
20
25
23°N
22°N
21°N
20°N
19°N
18°N
17°N
107°W
106°W
105°W
104°W
103°W
102°W
101°W