Introducción al Modelamiento Espacio-Temporal

Jonathan Acosta

2022-11-16

Procesos Espacio-Temporal

Sea $\{Z(\boldsymbol{s},t)\}$ con $\boldsymbol{s} \in D \subset \mathbb{R}^d$ y $t \in \mathbb{Z}^+$ un proceso espacio-temporal de media

$$\mu(\boldsymbol{s},t) = \mathbb{E}[Z(\boldsymbol{s},t)]$$

y función de covarianza

$$C(\boldsymbol{s}_1,\boldsymbol{s}_2,t_1,t_2) = \operatorname{cov}(Z(\boldsymbol{s}_1,t_1),Z(\boldsymbol{s}_2,t_2)).$$

Diremos que el proceso $\{Z(\boldsymbol{s},t)\}$ es débilmente estacionario de media μ y función de covarianza $C^0(\boldsymbol{h},u)$ si

$$\mu(\boldsymbol{s},t) = \mu \ \ \text{(cte.)}, \ \ \forall \boldsymbol{s},t \quad \wedge \quad C(\boldsymbol{s}_1,\boldsymbol{s}_2,t_1,t_2) = C^0(\boldsymbol{h},u), \ \ \boldsymbol{h} = \boldsymbol{s}_2 - \boldsymbol{s}_1, \ \ u = t_2 - t_1.$$

Es decir, la función de covarianza depende exclusivamente de la diferencia espacial y temporal. Considerando un abuso de notación escribiremos simplemente $C^0(\boldsymbol{h},u) = C(\boldsymbol{h},u)$.

Análogo, al caso espacial, se define el semivariograma como

$$\gamma(\boldsymbol{h},u) = \mathbb{V}[Z(\boldsymbol{s}+\boldsymbol{h},t+u) - Z(\boldsymbol{s},t)] = C(\boldsymbol{0},0) - C(\boldsymbol{h},u)$$

Estimadores empíricos

Sea $\boldsymbol{Z}_{n,T} = Z(\boldsymbol{s}_i,t_j)$ para $i = 1,\ldots,n$ y $j = 1,\ldots,T$, se define el estimador de momentos del variograma y la covianza como

$$\widehat{\gamma}(\boldsymbol{h},u) = \frac{1}{2|N(\boldsymbol{h},u)|}\sum_{N(\boldsymbol{h},u)} [Z(\boldsymbol{s}_i,t_i) - Z(\boldsymbol{s}_j,t_j)]^2$$

$$\widehat{C}(\boldsymbol{h},u) = \frac{1}{2|N(\boldsymbol{h},u)|}\sum_{N(\boldsymbol{h},u)} \left[Z(\boldsymbol{s}_i,t_i) - \bar{Z}\right]\left[Z(\boldsymbol{s}_j,t_j) - \bar{Z}\right]$$

donde $N(\boldsymbol{s},u) = \{[(\boldsymbol{s}_i,t_i),(\boldsymbol{s}_j,t_j)] : \ \boldsymbol{s}_i - \boldsymbol{s}_j = \boldsymbol{h}, \ t_i - t_j = u\}$ considerando una tolerancia del mismo modo que en el caso espacial.

Librerias de R

```
library("sp")
library("spacetime")
library("ggplot2")
library("dplyr")
library("gstat")
library("RColorBrewer")
```

```
library("STRbook")
library("tidyr")
library("grid")
library("gridExtra")
```

Formato de la Base datos para gstat

```
data(wind)       # base de datos del viento
head(wind)
```

```
##   year month day   RPT   VAL   ROS   KIL   SHA  BIR   DUB   CLA   MUL   CLO
## 1   61     1   1 15.04 14.96 13.17  9.29 13.96 9.87 13.67 10.25 10.83 12.58
## 2   61     1   2 14.71 16.88 10.83  6.50 12.62 7.67 11.50 10.04  9.79  9.67
## 3   61     1   3 18.50 16.88 12.33 10.13 11.17 6.17 11.25  8.04  8.50  7.67
## 4   61     1   4 10.58  6.63 11.75  4.58  4.54 2.88  8.63  1.79  5.83  5.88
## 5   61     1   5 13.33 13.25 11.42  6.17 10.71 8.21 11.92  6.54 10.92 10.34
## 6   61     1   6 13.21  8.12  9.96  6.67  5.37 4.50 10.67  4.42  7.17  7.50
##     BEL   MAL
## 1 18.50 15.04
## 2 17.54 13.83
## 3 12.75 12.71
## 4  5.46 10.88
## 5 12.92 11.83
## 6  8.12 13.17
```

```
class(wind)
```

```
## [1] "data.frame"
```

```
dim(wind)
```

```
## [1] 6574   15
```

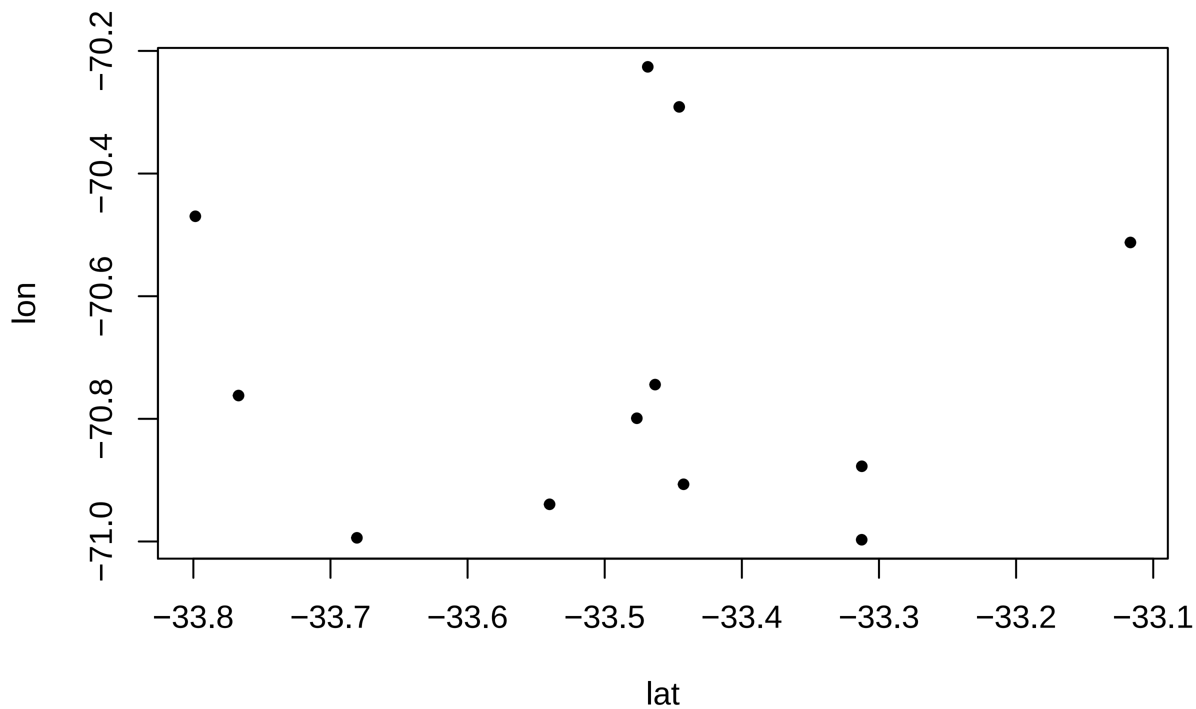
```
wind[c(1,nrow(wind)),1:3]     # inicio-fin del periodo
```

```
##      year month day
## 1      61     1   1
## 6574   78    12  31
```

```
time_part = as.POSIXct(paste( paste0(19,wind[,1]), wind[,2], wind[,3], sep="-" ), tz = "GMT")
head(time_part)
```

```
## [1] "1961-01-01 GMT" "1961-01-02 GMT" "1961-01-03 GMT" "1961-01-04 GMT"
## [5] "1961-01-05 GMT" "1961-01-06 GMT"
```

```
coords = cbind(lat= -33-runif(12), lon= -70-runif(12))
plot(coords, pch=20)
```




```
spat_part = SpatialPoints(coords = coords)

# formato STFDF
dfST <- STFDF(sp = spat_part,
              time = time_part,
              data = data.frame(wind=c(t(wind[,-(1:3)]))) )

proj4string(dfST) <- CRS("+proj=longlat +ellps=WGS84")
class(dfST)
```

```
## [1] "STFDF"
## attr(,"package")
## [1] "spacetime"
```

Kriging Espacio Temporal

Kriging universal espacio-temporal permite realizar predicciones cuando la media se modela linealmente y la covarianza viene dada por un proceso estacionario. Para realizar el ajuste del semivariograma y posterior predicción se utilizará la libreria *gstat*. Nos centramos en los datos de temperatura máxima del conjunto de datos de la NOAA (Tmax) en julio de 1993, disponible en la libreria *TSRBook*.

Base de datos para análisis

```
data("STObj3", package = "STRbook")           # Datos ya preparados en STRbook
STObj4 <- STObj3[, "1993-07-01::1993-07-31"]  # Se considera una parte de la Data
class(STObj4)
```

```
## [1] "STFDF"
## attr(,"package")
## [1] "spacetime"
```

```
head(STObj4@data, 10)      # El formato STFDF, requiere todos los campos
```

```
##    julian year month day   id  z
## 1  728111 1993     7   1 3804 82
## 2  728111 1993     7   1 3809 NA
## 3  728111 1993     7   1 3810 93
## 4  728111 1993     7   1 3811 91
## 5  728111 1993     7   1 3812 90
## 6  728111 1993     7   1 3813 95
## 7  728111 1993     7   1 3814 NA
## 8  728111 1993     7   1 3816 93
## 9  728111 1993     7   1 3817 NA
## 10 728111 1993     7   1 3820 96
```

```
head(STObj4@sp)            # coordenadas espaciales
```

```
## SpatialPoints:
##         lon      lat
## 1 -81.43333 39.35000
## 2 -89.40000 36.01667
## 3 -81.38333 35.73333
## 4 -88.91666 35.60000
## 5 -82.48333 35.43333
## 6 -83.65000 32.70000
## Coordinate Reference System (CRS) arguments: +proj=longlat +ellps=WGS84
```

```
head(STObj4@time)          # coordenadas temporales
```

```
##            timeIndex
## 1993-07-01      1278
## 1993-07-02      1279
## 1993-07-03      1280
## 1993-07-04      1281
## 1993-07-05      1282
## 1993-07-06      1283
```

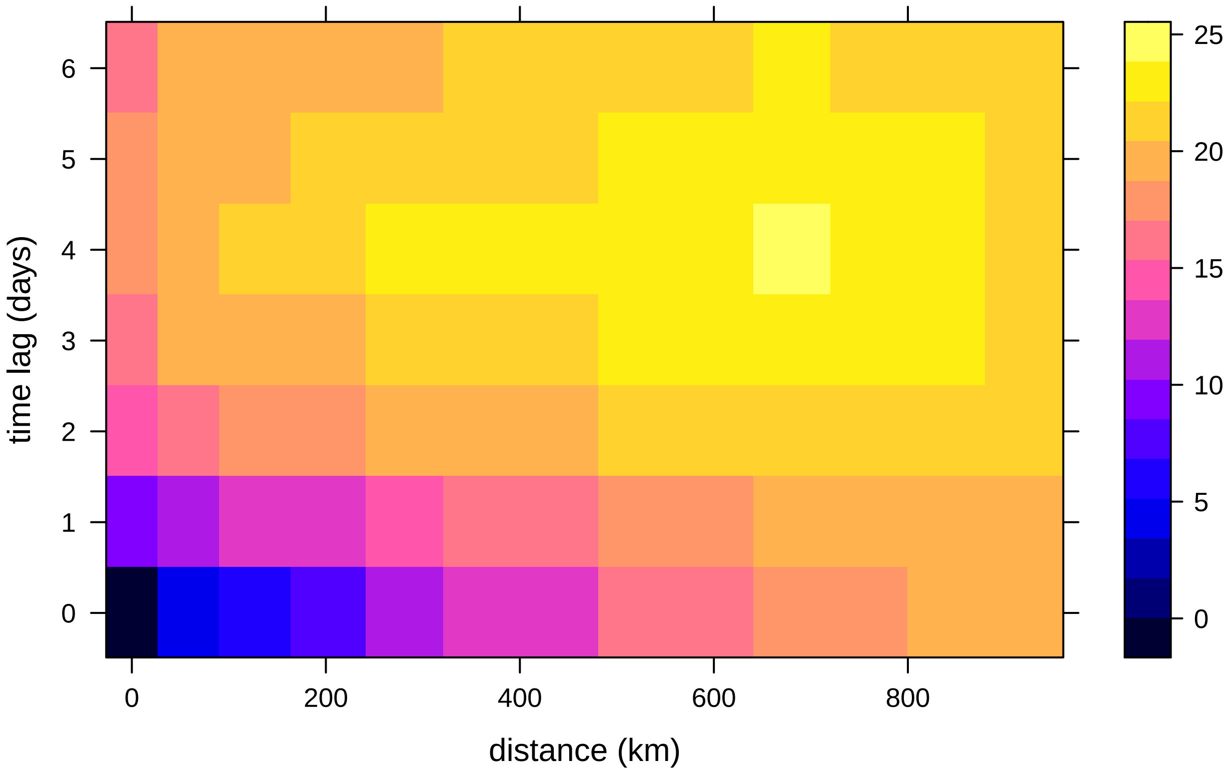
```
summary(STObj4)
```

```
## Object of class STFDF
##  with Dimensions (s, t, attr): (328, 31, 6)
## [[Spatial:]]
## Object of class SpatialPoints
## Coordinates:
##           min       max
## lon -99.96667 -80.00000
## lat  32.01667  45.86666
## Is projected: FALSE
## proj4string : [+proj=longlat +ellps=WGS84]
## Number of points: 328
## [[Temporal:]]
##      Index              timeIndex
##  Min.   :1993-07-01   Min.   :1278
##  1st Qu.:1993-07-08   1st Qu.:1286
##  Median :1993-07-16   Median :1293
##  Mean   :1993-07-16   Mean   :1293
##  3rd Qu.:1993-07-23   3rd Qu.:1300
##  Max.   :1993-07-31   Max.   :1308
## [[Data attributes:]]
##      julian             year          month         day            id
##  Min.   :728111   Min.   :1993   Min.   :7   Min.   : 1   Min.   : 3804
##  1st Qu.:728118   1st Qu.:1993   1st Qu.:7   1st Qu.: 8   1st Qu.:13879
##  Median :728126   Median :1993   Median :7   Median :16   Median :14820
##  Mean   :728126   Mean   :1993   Mean   :7   Mean   :16   Mean   :31856
##  3rd Qu.:728134   3rd Qu.:1993   3rd Qu.:7   3rd Qu.:24   3rd Qu.:14992
##  Max.   :728141   Max.   :1993   Max.   :7   Max.   :31   Max.   :94930
##
##        z
##  Min.   : 61.00
##  1st Qu.: 83.00
##  Median : 89.00
##  Mean   : 88.55
##  3rd Qu.: 95.00
##  Max.   :106.00
##  NA's   :6046
```

Análisis Descriptivo

Para explorar que la dependencia espacio-temporal es relevante debemos calcular el variograma empírico

```
vv <- variogram(object = z ~ 1 + lat,    # componente del efecto fijo
                data = STObj4,           # data
                width = 80,              # bin espacial (80 km)
                cutoff = 1000,           # considerar pts < 1000 km de distancia
                tlags = 0.01:6.01)       # 0 días a 6 días
plot(vv)
```



Modelos de Covarianza Paramétricos

Si $\{Z(\boldsymbol{s}, t\}$ es un proceso espacio-temporal con función de covarianza, $C(\boldsymbol{h}, u; \boldsymbol{\theta})$, parametrizada por $\boldsymbol{\theta}$. Estos modelos parametrizados se diferencian en no-separables y separables.

Modelos Separables

Sean $\rho_S(\boldsymbol{h}; \theta_1)$ y $\rho_T(u)$ funciones de correlación válidas en el espacio y tiempo, respectivamente, luego una función de correlación válida en el espacio-tiempo es

$$\rho_{ST}(\boldsymbol{h}, u) = \rho_S(\boldsymbol{h})\rho_T(u) \quad \wedge \quad \rho_{ST}(\boldsymbol{h}, u) = \rho_S(\boldsymbol{h}) + \rho_T(u)$$

Por ejemplo, un modelo de covarianza válido es

$$C(\boldsymbol{h}, u; \boldsymbol{\theta}) = \sigma^2 e^{-\|\boldsymbol{h}\|/\phi_s} e^{-|u|/\phi_t} \qquad \text{(Doble Exponencial)}$$

Modelos No-Separables

En este caso, la estructura de correlación $\rho_{ST}(\boldsymbol{h}, u)$ no se puede factorizar por el espacio y tiempo. Luego, algunos modelos de covarianza válidos son

$$C(\boldsymbol{h}, u; \boldsymbol{\theta}) = \frac{\sigma^2}{(|u|^{2\gamma} + 1)^{\nu}} \exp\left\{ \frac{-c\|\boldsymbol{h}\|^{2\gamma}}{(|u|^{2\gamma} + 1)^{\beta\gamma}} \right\} \qquad \text{(Gneiting)}$$

Un modelo general, corresponde a

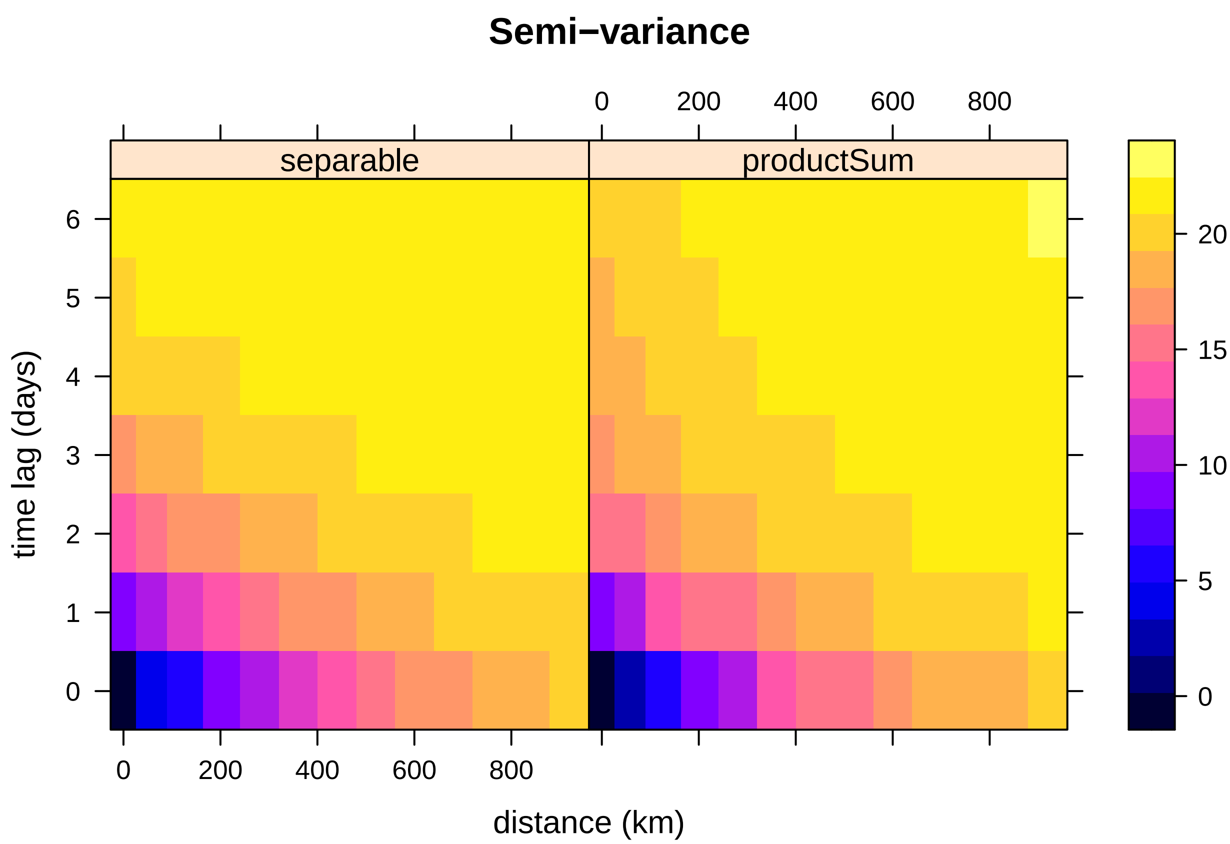
$$C(\boldsymbol{h}, u; \boldsymbol{\theta}) = pC_1^S(\boldsymbol{h})C_1^T(u) + qC_2^S(\boldsymbol{h}) + rC_2^T(u) \qquad \text{(Suma-Producto)}$$

Ajuste de Modelos de Covarianza

```
## Modelo Separable
sepVgm <- vgmST(stModel = "separable",
                space = vgm(10, "Exp", 400, nugget = 0.1),
                time = vgm(10, "Exp", 1, nugget = 0.1),
                sill = 20)
sepVgm <- fit.StVariogram(vv, sepVgm)

## Modelos No-Separable
sumprodVgm <- vgmST(stModel = "productSum",
                   space = vgm(10, "Exp", 400, nugget = 0.1),
                   time = vgm(10, "Exp", 1, nugget = 0.1),
                   k=15)
sumprodVgm <- fit.StVariogram(vv, sumprodVgm)

## Grafica de ambos variogramas
plot(vv, list(sepVgm, sumprodVgm), main = "Semi-variance")
```



Predicción

Considere el modelo

$$\boldsymbol{Z} = \boldsymbol{Y} + \boldsymbol{\epsilon} \quad \wedge \quad \boldsymbol{Y} = \boldsymbol{\mu} + \boldsymbol{\eta}$$

donde $\boldsymbol{\mu} = \boldsymbol{X}\boldsymbol{\beta}$. Notar que

$$\text{cov}(\boldsymbol{Y}) = \boldsymbol{C}_y = \boldsymbol{C}_\eta; \qquad \text{cov}(\boldsymbol{\epsilon}) = \boldsymbol{C}_\epsilon; \quad \wedge \quad \text{cov}(\boldsymbol{Z}) = \boldsymbol{C}_z = \boldsymbol{C}_y + \boldsymbol{C}_\eta$$

Sea $\boldsymbol{s}_0, t_0$ donde se requiere hacer la predicción y donde se tiene $\boldsymbol{x}_0 = \boldsymbol{x}(\boldsymbol{s}_0, t_0)$. Definiendo,

$$\boldsymbol{c}_0^\top = \text{cov}(Y(\boldsymbol{s}_0, t_0), \boldsymbol{Z}) \quad \wedge \quad c_{0,0} = \mathbb{V}(Y(\boldsymbol{s}_0, t_0)).$$

Asumiendo que se tiene la distribución Gaussiana conjunta, se tiene

$$Y(\boldsymbol{s}_0, t_0) \mid \boldsymbol{Z}\mathcal{N}(\boldsymbol{x}_0^\top \boldsymbol{\beta} + \boldsymbol{c}_0^\top \boldsymbol{C}_z^{-1}(\boldsymbol{Z} - \boldsymbol{X}\boldsymbol{\beta}); c_{0,0} - \boldsymbol{c}_0^\top \boldsymbol{C}_z^{-1}\boldsymbol{c}_0)$$

Los cuales corresponden al predictor Kriging simple, es decir,

$$\widehat{Y}(\boldsymbol{s}_0, t_0) = \boldsymbol{x}_0^\top \boldsymbol{\beta} + \boldsymbol{c}_0^\top \boldsymbol{C}_z^{-1}(\boldsymbol{Z} - \boldsymbol{X}\boldsymbol{\beta}) \quad \wedge \quad \sigma_{sk}^2(\boldsymbol{s}_0, t_0) = c_{0,0} - \boldsymbol{c}_0^\top \boldsymbol{C}_z^{-1}\boldsymbol{c}_0$$

En el caso que $\boldsymbol{\beta}$ sea desconocido (el caso más habitual) se tienen las ecuación de kriging universal

$$\widehat{Y}(\boldsymbol{s}_0, t_0) = \boldsymbol{x}_0^\top \widehat{\boldsymbol{\beta}} + \boldsymbol{c}_0^\top \boldsymbol{C}_z^{-1}(\boldsymbol{Z} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}) \quad \wedge \quad \sigma_{sk}^2(\boldsymbol{s}_0, t_0) = c_{0,0} - \boldsymbol{c}_0^\top \boldsymbol{C}_z^{-1}\boldsymbol{c}_0 + k$$

donde

$$\begin{aligned} \widehat{\boldsymbol{\beta}} &= (\boldsymbol{X}^\top \boldsymbol{C}_z^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^\top \boldsymbol{C}_z^{-1}\boldsymbol{Z} \\ k &= (\boldsymbol{x}_0 - \boldsymbol{X}^\top \boldsymbol{C}_z^{-1}\boldsymbol{c}_0)^\top (\boldsymbol{X}^\top \boldsymbol{C}_z^{-1}\boldsymbol{X})^{-1}(\boldsymbol{x}_0 - \boldsymbol{X}^\top \boldsymbol{C}_z^{-1}\boldsymbol{c}_0) \end{aligned}$$

Grilla de predicción

En primer lugar se remueven los datos donde no hay observaciones y se define la paleta de colores:

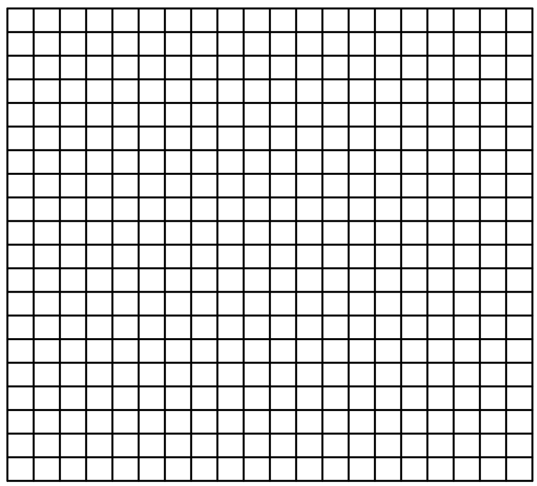
```
## ----------------------------------------------------------------------------
STObj5 <- as(STObj4[, -14], "STIDF")          # convert to STIDF
STObj5 <- subset(STObj5, !is.na(STObj5$z))    # remove missing data
head(STObj5@data)
```

```
##   julian year month day   id  z
## 1 728111 1993     7   1 3804 82
## 3 728111 1993     7   1 3810 93
## 4 728111 1993     7   1 3811 91
## 5 728111 1993     7   1 3812 90
## 6 728111 1993     7   1 3813 95
## 8 728111 1993     7   1 3816 93
```

```
## ----------------------------------------------------------------------------
color_pal <- rev(colorRampPalette(brewer.pal(11, "Spectral"))(16))
```

Ahora se procede a definir la grilla de predicción

```r
## parte espacial
spat_pred_grid <- expand.grid(
                      lon = seq(-100, -80, length = 20),
                      lat = seq(32, 46, length = 20)) %>%
            SpatialPoints(proj4string = CRS(proj4string(STObj3)))
gridded(spat_pred_grid) <- TRUE
plot(spat_pred_grid)
```



```r
## parte temporal
temp_pred_grid <- as.Date("1993-07-01") + seq(3, 28, length = 6)
temp_pred_grid
```

```
## [1] "1993-07-04" "1993-07-09" "1993-07-14" "1993-07-19" "1993-07-24"
## [6] "1993-07-29"
```

```r
## Formato para la grilla de predicción
DE_pred <- STF(sp = spat_pred_grid,    # spatial part
               time = temp_pred_grid)  # temporal part
head(DE_pred@sp)
```

```
## Object of class SpatialPixels
## Grid topology:
##     cellcentre.offset  cellsize cells.dim
## lon              -100 1.0526316        20
```

```
## lat                32 0.7368421        20
## SpatialPoints:
##             lon lat
## [1,] -100.00000  32
## [2,]  -98.94737  32
## [3,]  -97.89474  32
## [4,]  -96.84211  32
## [5,]  -95.78947  32
## [6,]  -94.73684  32
## Coordinate Reference System (CRS) arguments: +proj=longlat +ellps=WGS84
## +no_defs
```

```
head(DE_pred@time)
```

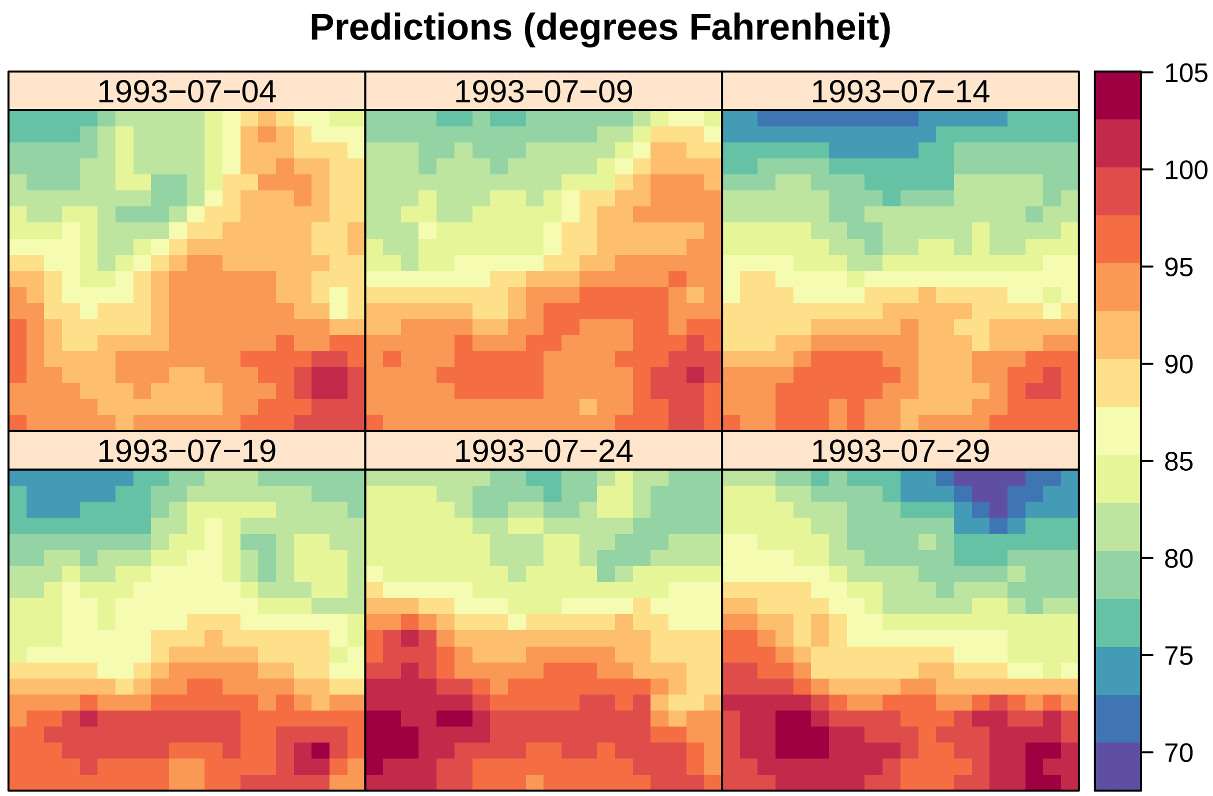
```
##            timeIndex
## 1993-07-04         1
## 1993-07-09         2
## 1993-07-14         3
## 1993-07-19         4
## 1993-07-24         5
## 1993-07-29         6
```

Predicción mediante el modelo separable

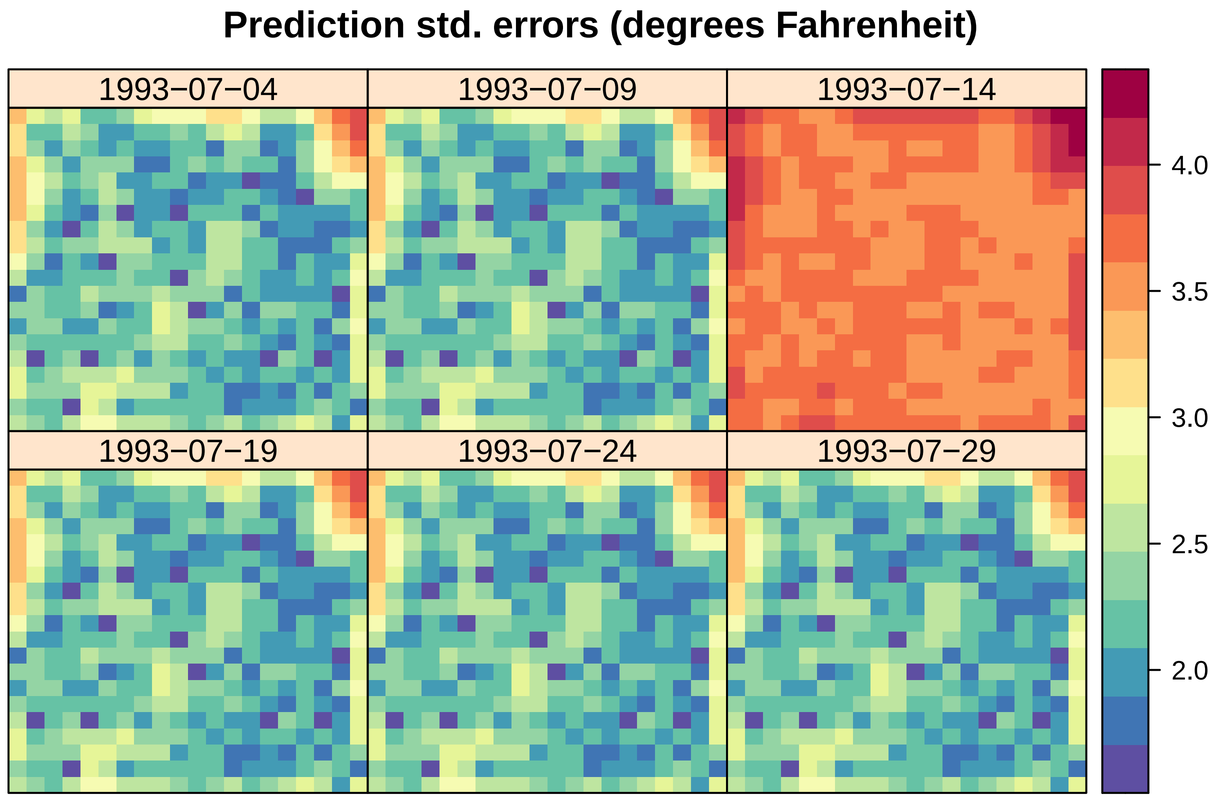
Para realizar la predicción mediante kriging universal con media lineal y función de covarianza separable se debe usar *krigeST*:

```
## ------------------------------------------------------------------------
pred_kriged <- krigeST(z ~ 1 + lat,         # latitude trend
                       data = STObj5,       # data set w/o 14 July
                       newdata = DE_pred,   # prediction grid
                       modelList = sepVgm,  # semivariogram
                       computeVar = TRUE)   # compute variances

## ------------------------------------------------------------------------
stplot(pred_kriged,
      main = "Predictions (degrees Fahrenheit)",
      layout = c(3, 2),
      col.regions = color_pal)
```




```
## ------------------------------------------------------------------------
pred_kriged$se <- sqrt(pred_kriged$var1.var)
stplot(pred_kriged[, , "se"],
      main = "Prediction std. errors (degrees Fahrenheit)",
      layout = c(3, 2),
      col.regions = color_pal)
```

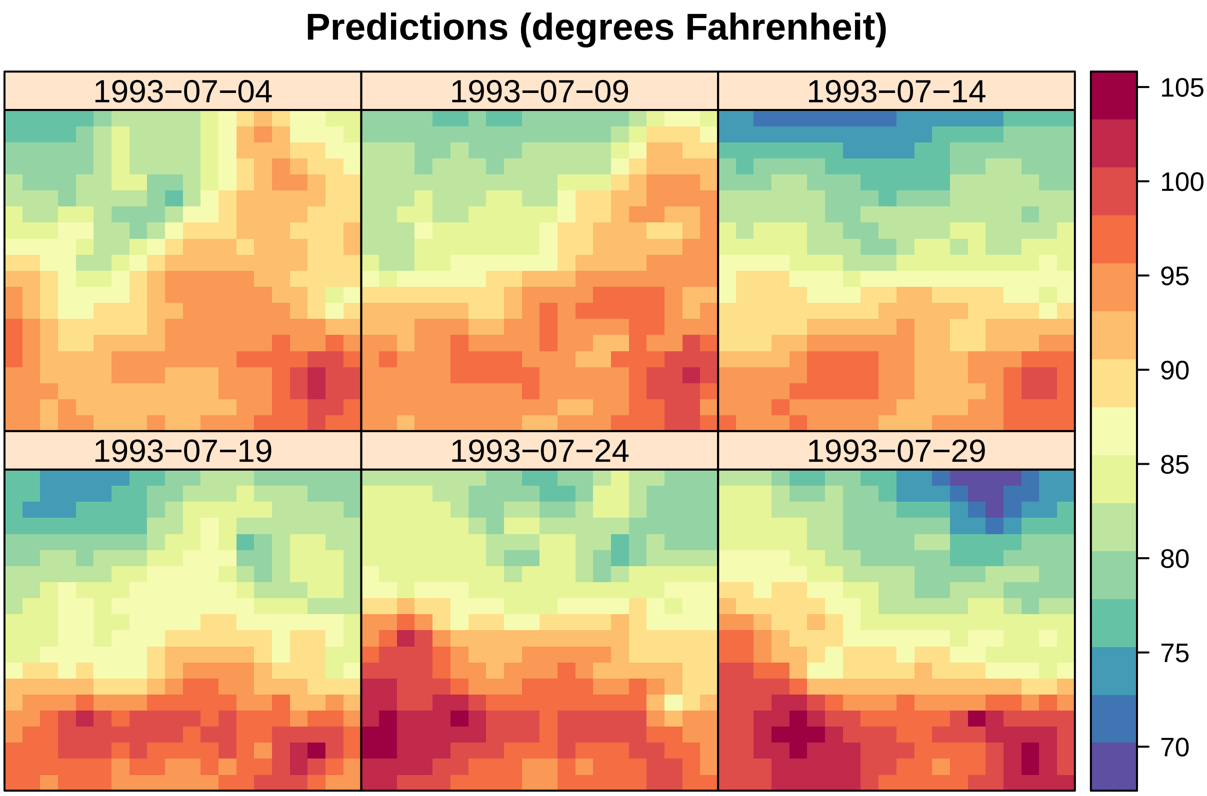


Predicción mediante el modelo sum-product

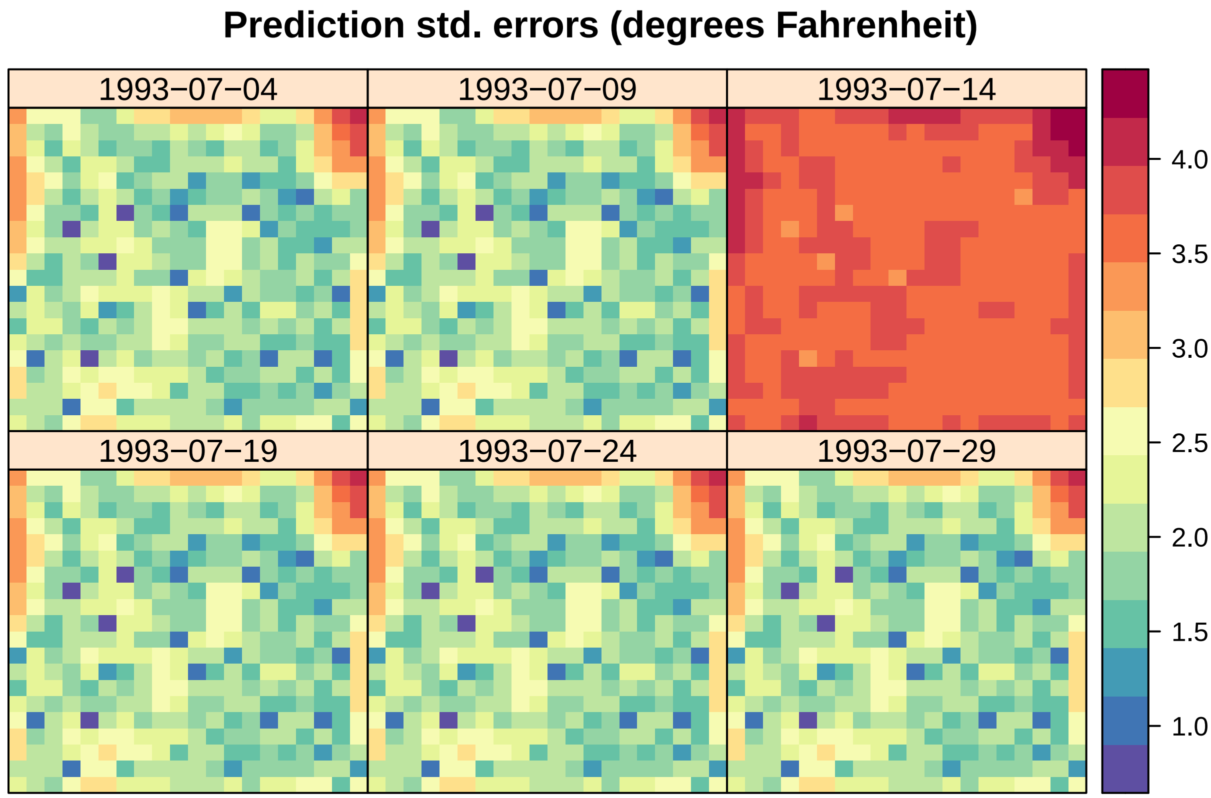
Para realizar la predicción mediante kriging universal con media lineal y función de covarianza no-separable dada por la estructura sum-product, también se debe usa *krigeST*, solo cambia el modelo de variograma:

```
## ------------------------------------------------------------------------
pred_kriged2 <- krigeST(z ~ 1 + lat,           # latitude trend
                        data = STObj5,          # data set w/o 14 July
                        newdata = DE_pred,      # prediction grid
                        modelList = sumprodVgm, # semivariogram
                        computeVar = TRUE)      # compute variances

## ------------------------------------------------------------------------
stplot(pred_kriged2,
      main = "Predictions (degrees Fahrenheit)",
      layout = c(3, 2),
      col.regions = color_pal)
```




```r
## ----------------------------------------------------------------------
pred_kriged2$se <- sqrt(pred_kriged2$var1.var)
stplot(pred_kriged2[, , "se"],
       main = "Prediction std. errors (degrees Fahrenheit)",
       layout = c(3, 2),
       col.regions = color_pal)
```



Prediction std. errors (degrees Fahrenheit)
1993−07−04
1993−07−09
1993−07−14
1993−07−19
1993−07−24
1993−07−29
4.0
3.5
3.0
2.5
2.0
1.5
1.0