UNIVERSIDAD DE GRANADA

DEPARTAMENTO DE ESTADÍSTICA E INVESTIGACIÓN OPERATIVA

MÁSTER OFICIAL EN ESTADÍSTICA APLICADA

TRABAJO FINAL DE MÁSTER:

VALIDACIÓN CRUZADA PARA EL CONTROL DE LA DEFORMACIÓN DEL ESPACIO EN LA ESTIMACIÓN DE LA DISPERSIÓN ESPACIAL DE PROCESOS NO ESTACIONARIOS MEDIANTE SMACOF

Presentado Por:

IVÁN DARÍO PEÑARANDA ARENAS

Director:

Dr. JOSÉ FERNANDO VERA VERA

Granada, septiembre de 2016

TABLA DE CONTENIDO

1. INTRODUCCIÓN 1

2. ESCALAMIENTO MULTIDIMENSIONAL 3

2.1. MDS clásico 4

2.2. Variantes del MDS 5

2.2.1. MDS métrico 6

2.2.2. MDS no-métrico 6

2.3. Algoritmo SMACOF 9

3. EL ENFOQUE DE SAMPSON & GUTTORP PARA LA ESTIMACIÓN DE LA ESTRUCTURA DE COVARIANZA ESPACIAL NO ESTACIONARIA 12

3.1. Descripción del procedimiento 13

4. IMPLEMENTACIÓN DEL MÉTODO DE SAMPSON Y GUTTORP EN R 18

4.1. El paquete EnviroStat 18

4.2. Tratamiento computacional mediante smacof y un procedimiento de validación cruzada 24

4.2.1. Selección del parámetro de suavizado mediante validación cruzada 27

5. EJEMPLOS DE APLICACIÓN 32

5.1. Comparación del paquete EnviroStat original con el modificado 32

5.1.1. Implementación mediante la versión original de EnviroStat 36

5.1.2. Implementación mediante la versión modificada de EnviroStat 40

5.2. Ejemplo de implementación completa del método de Sampson y Guttorp 51

6. CONCLUSIONES Y TRABAJO FUTURO 71

7. REFERENCIAS BIBLIOGRÁFICAS 74

1. INTRODUCCIÓN

Una práctica común en geoestadística consiste en asumir que el proceso espacial subyacente es estacionario e isotrópico, suposición que es necesaria para aplicar diversos métodos de krigeado. No obstante, en problemas ambientales estas suposiciones no son realistas, dado que existen influencias locales en la estructura de correlación del proceso espacial y por consiguiente no es válido asumir un comportamiento homogéneo de la covarianza a través del dominio completo del campo aleatorio.

Entre los diversos enfoques desarrollados para abordar procesos espacio-temporales no estacionarios se destaca el de Sampson y Guttorp (1992), siendo éste un método no paramétrico que permite la estimación de la estructura de covarianza del proceso espacial, la cual es un requisito fundamental para la solución de problemas de interpolación espacial usando métodos como el krigeado y para el diseño de redes de monitoreo.

En términos generales, el procedimiento de Sampson y Guttorp utiliza escalamiento multidimensional no-métrico para obtener una representación de las estaciones de muestreo donde es válida la suposición de estacionariedad del proceso; de este modo se transforma el problema de estimar la estructura de covarianza, ya que al expresar ésta en términos de dispersiones espaciales en el nuevo espacio pasa a ser estacionaria e isotrópica. Posteriormente, se calculan splines de placa delgada como funciones que permiten relacionar los puntos de la representación geográfica con los correspondientes a la representación obtenida mediante MDS conocida también como espacio de dispersión. Estas funciones suavizadas apropiadamente junto con el modelo de variograma isotrópico obtenido a partir de las correlaciones observadas y las distancias en el espacio de dispersión, permiten la estimación de la correlación espacial entre cualquier par de ubicaciones de interés.

El paquete *EnviroStat* de R proporciona funciones que permiten la estimación de la matriz de covarianza espacial entre las estaciones mediante el algoritmo EM y una vez calculada ésta la extienden utilizando el método de Sampson y Guttorp en un procedimiento que consta de 5 etapas, generando así correlaciones espaciales entre sitios de interés, incluyendo aquellos que no han sido monitoreados aún.

El objetivo central de este trabajo es presentar una metodología de selección del parámetro de suavizado del spline de placa delgada usando el enfoque de validación cruzada con el fin de mejorar la implementación del método de Sampson y Guttorp mediante los paquetes *smacof* y *EnviroStat*. Para ello se incluye una normalización de la matriz de dispersión (obtenida a partir de la matriz de correlación espacial) y de la matriz de coordenadas de las estaciones. En cuanto a esta última, el método de normalización permite eliminar la arbitrariedad en la selección de su escala, ya que en la versión original de *EnviroStat* es necesario multiplicar la matriz por un factor definido por el usuario para que su escala sea lo suficientemente pequeña, de modo que no se presenten problemas en el cálculo de los splines de placa delgada. Otra modificación introducida es la incorporación del paquete *smacof* en la función que permite generar el espacio de dispersión mediante escalamiento multidimensional en la etapa 1 del procedimiento. El último cambio hecho al paquete *EnviroStat* es la incorporación de una metodología para la selección del parámetro de suavizado λ adoptando el enfoque de validación cruzada, dado que en la versión original del paquete no se proporciona un criterio claro para escogerlo.

El trabajo se organiza del siguiente modo: en las secciones 2 y 3 se presentan los fundamentos teóricos del escalamiento multidimensional y del método de Sampson y Guttorp, la siguiente sección discute las principales características del paquete *EnviroStat* y las modificaciones introducidas. En la sección 5 se comparan en primer lugar la versión original con la versión modificada en lo referente a la ejecución de las dos primeras etapas y luego se ilustra con otro ejemplo la implementación completa del procedimiento de Sampson y Guttorp mediante la versión modificada del paquete *EnviroStat*. La última parte presenta las conclusiones y sugiere trabajo a futuro.

2. ESCALAMIENTO MULTIDIMENSIONAL

El escalamiento multidimensional (MDS) es un método de análisis estadístico multivariante que representa mediciones de similaridad (o disimilaridad) entre pares de objetos como distancias entre puntos de un espacio de dimensión reducida. La técnica tiene sus orígenes en los estudios de psicología experimental en la década de 1950, llevados a cabo para descubrir la similaridad entre estímulos aplicados a distintos individuos; y es en el área de las ciencias sociales donde preferentemente se han aplicado muchos de los avances de las investigaciones. No obstante, el MDS ha encontrado aplicación en una amplia gama de disciplinas científicas, entre otras razones porque admite una gran variedad de datos de entrada como tablas de contingencia, matrices de proximidad y correlaciones.

El objetivo fundamental del MDS consiste en generar un mapa o representación gráfica de los objetos en un espacio de modo que sus posiciones relativas en tal configuración sean el reflejo del grado de proximidad percibida entre los objetos. Otros propósitos de este método que vale la pena destacar son los siguientes (Borg & Groenen, 2005):

- Facilitar el análisis exploratorio de los datos generando una representación en un espacio de dimensión reducida haciendo que estos sean accesibles a la inspección visual del investigador, de modo que pueda apreciarse la estructura de los datos y se encuentren reglas que ayuden a describir su distribución.
- Permitir el contraste de hipótesis estructurales, es decir, probar si y cómo ciertos criterios mediante los cuales se pueden distinguir entre diferentes objetos de interés se reflejan en las diferencias empíricas que corresponden entre observaciones en estos ítems.
- Descubrir las dimensiones que subyacen a los juicios de similaridad o disimilaridad, su número e importancia relativa.
- Proporcionar un modelo que explique los juicios de similaridad en términos de una regla que imita un tipo particular de función de distancia.

2.1. MDS clásico

Este método de escalamiento multidimensional asume que las proximidades se comportan como si fueran distancias medidas en realidad (generalmente euclídeas), suposición que puede aceptarse para aquellos datos que se derivan de matrices de correlación, pero raramente para valoraciones de disimilaridad directas. La ventaja de esta técnica es que proporciona una solución analítica y que tampoco requiere procedimientos iterativos (Wickelmaier, 2003).

El algoritmo de este método depende del hecho de que la matriz de coordenadas $\boldsymbol{X}$ puede derivarse mediante descomposición en autovalores a partir de la matriz de producto escalar $\boldsymbol{B} = \boldsymbol{X}\boldsymbol{X}'$. El problema de obtener $\boldsymbol{B}$ a partir de la matriz de distancias $\boldsymbol{D}_{n\times n}$ entre n objetos se resuelve multiplicando las distancias al cuadrado por la matriz $\boldsymbol{H} = \boldsymbol{I} - n^{-1}\mathbf{1}\mathbf{1}'$, procedimiento conocido como doble centrado. La matriz $\boldsymbol{D}$ es simétrica con $d_{rr} = 0$ y $d_{rs} \geq 0$ para $r \neq s$.

El algoritmo del MDS clásico comprende los siguientes pasos (Mardia *et al.*, 1979):

1. Obtener la matriz de distancias al cuadrado $\boldsymbol{D}^2$.
2. Aplicar el doble centrado: $\boldsymbol{B} = -\frac{1}{2}\boldsymbol{H}\boldsymbol{D}^2\boldsymbol{H}$ usando la matriz $\boldsymbol{H} = \boldsymbol{I} - n^{-1}\mathbf{1}\mathbf{1}'$.
3. Extraer los k autovalores positivos mayores $\lambda_1, \dots, \lambda_k$ de $\boldsymbol{B}$ y los correspondientes k autovectores $e_1, \dots, e_k$.
4. Derivar una configuración espacial en k dimensiones de los n objetos a partir de la matriz de coordenadas $\boldsymbol{X} = \boldsymbol{E}_k\boldsymbol{\Lambda}_k^{1/2}$, donde $\boldsymbol{E}_k$ es la matriz de los k autovectores y $\boldsymbol{\Lambda}_k$ es la matriz diagonal de los k autovalores de $\boldsymbol{B}$, respectivamente. Las filas de $\boldsymbol{X}$ corresponden a las coordenadas de los puntos en la configuración requerida.

Si λ_p representa el menor autovalor positivo de $\boldsymbol{B}$, con $p \geq k$; al emplear $k = p$ se obtiene la solución de escalamiento multidimensional que proporciona el mejor ajuste posible a las distancias. En la práctica se utiliza un valor más pequeño de k, el cual puede seleccionarse con base en la siguiente magnitud (Mardia *et al.*, 1979):

$$\alpha_k = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} |\lambda_i|}; \quad k = 1, \dots, p \quad (1)$$

Esta medida puede entenderse como "la proporción de la matriz de distancias ***D*** explicada" por la solución de escalamiento multidimensional. Dado que los autovalores se ordenan por tamaño, α_k será mayor a medida que aumenta k, pero el incremento de $k-1$ a k es menor para valores mayores de k. Un criterio para seleccionar k consiste en tomar aquel valor que permita obtener un α_k de al menos el 90% del valor máximo (α_p). Por consiguiente, se puede afirmar que escoger el número de dimensiones del MDS es análogo a la selección del número de componentes en el análisis de componentes principales.

Existen también consideraciones de orden práctico en la selección de k; en efecto, si $k = 1$ o $k = 2$ el cómputo se efectúa de forma más rápida y se obtiene una configuración más fácil de visualizar (Løland & Høst, 2003).

2.2. Variantes del MDS

El escalamiento multidimensional es una familia de modelos que tienen en común el esquematizar las proximidades mediante distancias entre puntos de un espacio de dimensión k. Las variantes surgen debido a las diferentes suposiciones sobre la escala de medida de las proximidades y al uso de diferentes funciones para calcular la distancia; esto último da lugar a distintas geometrías en los modelos, ya que la distancia euclídea al igual que otras distancias de Minkowski implican una geometría plana, mientras que la distancia geodésica implica una geometría curvada. A continuación se hará un breve análisis de las dos principales variantes del escalamiento multidimensional: el MDS métrico y el MDS no-métrico u ordinal.

2.2.1. MDS métrico

Esta variante del MDS especifica una función analítica que relaciona las distancias en la configuración $\boldsymbol{X}$ con las proximidades, es decir, $d_{ij}(\boldsymbol{X}) = f(p_{ij})$. Uno de los requisitos que usualmente cumple la función f es ser monótona, esto implica que:

$$f: p_{ij} < p_{kl} \rightarrow d_{ij}(\boldsymbol{X}) \leq d_{kl}(\boldsymbol{X}) \quad (2)$$

Generalmente este modelo de MDS se aplica a proximidades cuya escala de medida es intervalo o razón. El modelo estándar del MDS métrico es el MDS intervalo, en el cual se asume que la relación entre las proximidades y las distancias es lineal:

$$p_{ij} \rightarrow a + bp_{ij} = d_{ij}(\boldsymbol{X}) \quad (3)$$

De este modo, toda la información sobre los datos que permanece invariante bajo tales transformaciones lineales se considera significativa, mientras que las demás acotaciones, como aquellas referentes a la proporción de ciertos valores de datos, no son significativas.

El modelo de MDS métrico más restrictivo es el MDS razón, en este caso la constante aditiva a del MDS intervalo es eliminada y busca una solución que preserva las proximidades en función de un factor de escalamiento b.

2.2.2. MDS no-métrico

La suposición del MDS métrico de que las proximidades se comportan como distancias puede ser muy restrictiva cuando se aplica el MDS a la exploración del espacio perceptual de los sujetos humanos. Con el fin de resolver este problema, Shepard y Kruskal desarrollaron esta variante del MDS que a diferencia de la anterior, asume que las proximidades están en escala ordinal. De esta forma, en la construcción de la configuración espacial se utiliza únicamente la

información ordinal de las proximidades, es decir, solamente su rango u orden se considera información confiable y válida.

De forma análoga a la versión anterior, el MDS no-métrico transforma las proximidades mediante una función monótona, obteniéndose así proximidades escaladas que son optimizadas, a las cuales se conoce también como disparidades $\hat{d} = f(p)$.

El problema que aborda el MDS no-métrico es el de hallar una configuración de puntos $\boldsymbol{X}$ tal que las distancias sobre ésta queden ordenadas tan cerca como sea posible a las proximidades y se logren minimizar las diferencias al cuadrado entre las disparidades y las distancias entre los puntos. Este problema es equivalente al de encontrar las coordenadas que minimicen el *stress*, el cual puede calcularse con la siguiente expresión:

$$STRESS = \sqrt{\frac{\sum(f(p) - d)^2}{\sum d^2}} \quad (4)$$

Tal como se aprecia en la ecuación anterior, la magnitud del stress es proporcional a la diferencia entre las disparidades y las distancias, siendo entonces un indicador de la bondad de ajuste del modelo; en efecto, un valor bajo del stress indica que se obtuvo un buen ajuste con la solución, mientras que un valor alto corresponde a un mal ajuste. Con el fin de interpretar el valor del stress respecto a la bondad de ajuste de la solución suele emplearse la siguiente guía sugerida por Kruskal (Wickelmaier, 2003):

Stress	*Bondad de ajuste*
> 0.20	pobre
0.1	aceptable
0.05	bueno
0.025	excelente
0.00	perfecto

Tabla 1. Stress y bondad de ajuste

Frecuentemente la guía mostrada en la tabla anterior es malinterpretada, ya que se considera que estos criterios pueden aplicarse a cualquier valor del stress independientemente de la fórmula empleada para calcularlo, cuando en realidad esta guía solo es aplicable a la medida del stress obtenido mediante la ecuación (4), conocido también como *Stress 1*. Otro aspecto a tener en cuenta es que el stress disminuye a medida que aumenta el número de dimensiones de la solución.

Dado que la magnitud del stress no proporciona una indicación clara de la bondad del ajuste, existen dos técnicas adicionales que comúnmente se usan para juzgar la idoneidad del modelo: el gráfico de sedimentación y el diagrama de Shepard. El primero de ellos representa la cantidad de stress frente al número de dimensiones de la solución, se busca en éste el menor número de dimensiones asociado a un valor aceptable del stress; un codo en este diagrama indica que la adición de dimensiones a la solución producirá solo una mejora menor en términos del stress, por consiguiente, el mejor ajuste se logra con aquel modelo que utiliza el número de dimensiones que corresponde al codo en esta gráfica. El diagrama de Shepard muestra la relación entre las proximidades y las distancias entre los puntos de la configuración, entre menor sea la dispersión mejor es el ajuste; en el MDS no-métrico la ubicación ideal de los puntos en este diagrama es una línea que aumenta de forma monótona y que describe a las disparidades (Wickelmaier, 2003).

Existen dos formas de MDS no-métrico cuya diferencia reside en el modo en el que tratan los empates o valores iguales en los datos. El enfoque primario que es el adoptado por defecto en la mayoría de programas, consiste en considerar que las proximidades iguales no necesariamente corresponden a distancias iguales; mientras que el enfoque secundario lleva a mantener los empates, es decir, que las proximidades iguales correspondan a iguales distancias en la solución (Borg *et al*., 2013).

El algoritmo del MDS no-métrico comprende un proceso de optimización dual en el que debe encontrarse primero una transformación monótona óptima de las proximidades y posteriormente

debe arreglarse óptimamente la configuración de los puntos, de manera que sus distancias correspondan a las proximidades escaladas o disparidades lo más cerca que sea posible. Básicamente las etapas del algoritmo del MDS no-métrico son las siguientes (Wickelmaier, 2003):

1. Hallar una configuración aleatoria de puntos, por ejemplo mediante una muestra tomada de una distribución normal.
2. Calcular las distancias d entre los puntos.
3. Hallar la transformación monótona óptima de las proximidades, con el fin de obtener datos óptimamente escalados o disparidades $\hat{d} = f(p)$.
4. Minimizar el stress entre las disparidades y las distancias encontrando una nueva configuración de puntos.
5. Comparar el stress con algún criterio. En caso que el stress sea lo suficientemente pequeño se termina el algoritmo y en caso contrario se retorna al paso 2.

2.3. Algoritmo SMACOF

Este algoritmo minimiza el stress mediante mayorización, en efecto SMACOF significa escalamiento vía mayorización de una función complicada. En sentido estricto, la mayorización no es un algoritmo sino una prescripción para construir algoritmos de optimización. La idea de la mayorización es optimizar una función sustituta más simple que la función original, se garantiza que la función sustituta tiene un valor mayor al de la función original y es igual a esta última en un punto de soporte. En cada iteración, la configuración final es usada como el punto de inicio para la próxima iteración (De Leeuw & Mair, 2009).

El algoritmo SMACOF converge a un punto fijo y es equivalente a un algoritmo de gradiente descendente ponderado con tamaño de paso constante. En la versión simple que corresponde al caso de una matriz de disimilaridad Δ simétrica, la función stress $\sigma(\boldsymbol{X})$ se define así:

$$\sigma(\boldsymbol{X}) = \sum_{i<j} w_{ij}\,(\delta_{ij} - d_{ij}(\boldsymbol{X}))^2 \quad (5)$$

El algoritmo ubica los $i, j = 1, \ldots, n$ puntos en un espacio euclídeo de baja dimensión de tal manera que las distancias entre los elementos en la configuración $d_{ij}(\boldsymbol{X})$ se aproximan a las disimilaridades δ_{ij}.

Otro dato de entrada del algoritmo es la matriz de ponderaciones $\boldsymbol{W}_{n\times n}$, que al igual que la matriz de disimilaridades se asume no-negativa y con elementos iguales a cero en la diagonal. Una de las aplicaciones de esta matriz es facilitar la manipulación de los valores faltantes; por ejemplo, $w_{ij} = 0$ si el dato falta y 1 en el caso contrario; otro uso de dicha matriz es definir el énfasis que tendrán en el análisis las disimilaridades.

El algoritmo inicia asignando en el paso $t = 0$ el punto de soporte $\boldsymbol{Y} = \boldsymbol{X}^{(0)}$ donde $\boldsymbol{X}^{(0)}$ es una configuración inicial. Dentro de cada iteración t se calcula $\overline{\boldsymbol{X}}^{(t)}$, siendo esta última la transformada de Guttman de la configuración, la cual se calcula mediante la siguiente ecuación:

$$\overline{\boldsymbol{X}}^{(t)} = \boldsymbol{V}^{+}\boldsymbol{B}(\boldsymbol{Y})\boldsymbol{Y} \quad (6)$$

Donde $\boldsymbol{V}^{+}$representa la inversa de Moore-Penrose: $\boldsymbol{V}^{+} = (\boldsymbol{V} + n^{-1}\mathbf{1}\mathbf{1}')^{-1} - n^{-1}\mathbf{1}\mathbf{1}'$. La matriz $\boldsymbol{V}$ se define así:

$$\boldsymbol{V} = \sum_{i<j} w_{ij}\,A_{ij} \quad (7)$$

Los elementos de la matriz $\boldsymbol{A}$ son iguales a 1 cuando $a_{ii} = a_{jj}$, -1 en el caso que $a_{ij} = a_{ji}$, y 0 en las demás posiciones. Por otro lado, la matriz $\boldsymbol{B}$ evaluada en el punto de soporte $\boldsymbol{Y}$ es igual a:

$$\boldsymbol{B}(\boldsymbol{Y}) = \sum_{i<j} w_{ij}\,s_{ij}(\boldsymbol{Y})A_{ij} \quad (8)$$

$$s_{ij}(\boldsymbol{Y}) = \begin{cases} \delta_{ij}/d_{ij}(\boldsymbol{Y}) & si\ d_{ij}(\boldsymbol{Y}) > 0 \\ 0 & si\ d_{ij}(\boldsymbol{Y}) = 0 \end{cases} \quad (9)$$

En la versión simple de SMACOF la actualización que corresponde a la iteración t, es decir $X^{(t)}$, se hace equivalente a $\bar{X}^{(t)}$. El siguiente paso es calcular el stress $\sigma\left(X^{(t)}\right)$, el proceso de iteración continúa hasta que la diferencia entre el stress calculado en una iteración y la anterior sea menor a una determinada tolerancia, o en otros términos: $\sigma\left(X^{(t)}\right) - \sigma\left(X^{(t-1)}\right) < \epsilon$, o se alcance un límite en el número de iteraciones. El algoritmo garantiza que en cada iteración el stress disminuye. Otra característica del algoritmo es que a medida que aumenta el número de dimensiones de la solución disminuye la probabilidad de presencia de mínimos locales.

Un aspecto a tener en cuenta del algoritmo SMACOF es que cada vez que se calculan las disparidades óptimas $\hat{d}_{ij}$ para las distancias de la actualización de la iteración t, $d_{ij}(X^{(t)})$, éstas se normalizan de la siguiente forma (Borg & Groenen, 2005):

$$\sum_{i<j} w_{ij} \, (\hat{d}_{ij})^2 = \frac{n(n-1)}{2} \quad (10)$$

3. EL ENFOQUE DE SAMPSON & GUTTORP PARA LA ESTIMACIÓN DE LA ESTRUCTURA DE COVARIANZA ESPACIAL NO ESTACIONARIA

El método de Sampson y Guttorp es un enfoque no paramétrico que pretende estimar y representar gráficamente la estructura de covarianza espacial de un campo aleatorio sin asumir que el proceso es estacionario. El método construye en primer lugar una función que relacione las ubicaciones del espacio geográfico, donde no se asume estacionariedad del campo aleatorio, y las ubicaciones de un nuevo espacio en el que la suposición de un proceso estacionario e isotrópico es válida. Al espacio geográfico se le conoce como *espacio-G*, mientras que al nuevo espacio se le denomina *espacio de dispersión* o *espacio-D*. De esta forma, es posible ajustar un modelo de variograma isotrópico empleando las correlaciones observadas y las distancias en el espacio-*D*, el cual es utilizado junto con una función suavizada para estimar las correlaciones entre cualquier par de ubicaciones de interés del espacio-*G*.

Al enfoque de Sampson y Guttorp también se le conoce como "deformación", ya que distorsiona el espacio-*G* para obtener el espacio-*D* a través de transformaciones suavizadas de las coordenadas geográficas en coordenadas del espacio de dispersión, de modo que la distancia entre pares de sitios en el espacio deformado es inversamente proporcional a la correlación existente entre estos sitios. El método incorpora un parámetro de suavizado en la función de mapeo para controlar la distorsión entre los dos espacios, lo cual asegura que la cuadrícula no esté doblada en el espacio-*D* y mantenga por tanto la interpretabilidad espacial de las correlaciones, es decir, a menor distancia entre las estaciones corresponde una mayor correlación (Le & Zidek, 2006).

La aplicación del método de Sampson y Guttorp requiere de dos herramientas fundamentales: el escalamiento multidimensional y la interpolación mediante splines de placa delgada. La primera herramienta implica el desarrollo de un modelo de MDS no-métrico de la matriz de covarianza espacial con el fin de conseguir una configuración bidimensional de las estaciones de monitoreo

con distancias entre los puntos que representan versiones suavizadas de la dispersión espacial muestral (Vera *et al*., 2016). La otra herramienta clave para el éxito del método es un conjunto de funciones suavizadas biyectivas que permiten el enlace entre los dos espacios: splines de placa delgada, los cuales asignan puntos del espacio-*G* a puntos del espacio-*D*. Es así que el proceso se convierte en isotrópico, ya que la correlación espacial entre dos puntos en el espacio-*D* es una función monótona de la distancia euclídea entre ellos.

3.1. Descripción del procedimiento

Con el fin de dar una descripción más precisa del método se supone que f: $R^2 \rightarrow R^2$ es una función biyectiva no lineal suavizada que asigna puntos del espacio-*G* al espacio-*D*. De esta forma, para una ubicación p_i en el espacio-*G* la ubicación correspondiente en el espacio-*D* q_i se obtiene como $q_i = f(p_i)$ e igualmente $p_i = f^{-1}(q_i)$ siendo f^{-1} la inversa de f.

El variograma del campo aleatorio Y entre un par de sitios p_i y p_j puede expresarse en términos de ubicaciones en el espacio-*D* así:

$$2\gamma(p_i, p_j) \equiv var\left[Y(p_i) - Y(p_j)\right]$$

$$= var\left[Y(q_i) - Y(q_j)\right]$$

$$= 2g\left(\left|h_{ij}^{D}\right|\right) \quad (11)$$

Siendo q_i y q_j las ubicaciones correspondientes al espacio-*D*, $\left|h_{ij}^{D}\right|$ es la distancia en el espacio-*D*, y g representa el semi-variograma en el espacio-*D*, el cual depende únicamente de la distancia ya que en dicho espacio se asume que el proceso es estacionario e isotrópico. Respecto al semi-variograma g se puede emplear un modelo como el exponencial o el gaussiano, entre otros, para lograr un ajuste satisfactorio de los datos del semi-variograma muestral.

El enfoque de Sampson y Guttorp comprende dos etapas para la estimación de g y f en las que se emplean dispersiones muestrales entre las ubicaciones $p_1, \dots, p_n$ en el espacio-*G* representadas por d_{ij}^2. En la primera etapa se aplica el MDS para obtener una representación bidimensional $(q_1, \dots, q_N)$ de los sitios, tal que:

$$\delta(d_{ij}) \equiv \delta_{ij} \approx |q_i - q_j| \quad (12)$$

Siendo δ una función monótona. Las δ_{ij} se conocen como disimilaridades, estas pueden estimarse del siguiente modo:

$$\delta_{ij}^2 = s_{ii} + s_{jj} - 2s_{ij} \quad (13)$$

Donde los s_{ij} para $i, j = 1, \dots, N$ son los elementos de la matriz de covarianza muestral y N es el número de estaciones. De esta forma se proporciona una medida empírica de las disimilaridades para determinar la dispersión espacial (Vera *et al*., 2016). La matriz de covarianza muestral puede obtenerse así:

$$\boldsymbol{S} = (1/T)\, \boldsymbol{Z}\boldsymbol{H}_T\boldsymbol{Z}' \quad (14)$$

$$\boldsymbol{H}_T = \boldsymbol{I} - (1/T)\, \mathbf{1}\mathbf{1}' \quad (15)$$

Siendo $\boldsymbol{Z}_{N\times T}$ la matriz muestral que corresponde a una función aleatoria $Z(x,t)$ con estacionariedad temporal de segundo orden y espacialmente no estacionaria, dicha función es observada regularmente en tiempos $t = 1, \dots, T$, en un número de estaciones de monitoreo con ubicaciones $x_i, i = 1, \dots, N$; en la ecuación 15, $\mathbf{1}$ es un vector columna de unos cuyo número de filas es T.

El semi-variograma g se obtiene mediante las distancias entre las nuevas ubicaciones en el espacio-*D*, $(q_1, \dots, q_N)$ resolviendo la relación mostrada en la ecuación 12, en efecto:

$$d_{ij}^2 = \left(\delta^{-1}(\delta_{ij})\right)^2 \approx g(|q_i - q_j|) \quad (16)$$

Para hallar la representación de las ubicaciones q_i se busca que las distancias entre los sitios en el espacio-D $\left|h_{ij}^{D}\right|$ minimicen el siguiente criterio de stress:

$$min_{\delta}\left[\sum_{i<j}\frac{\left(\delta_{ij}-h_{ij}^{D}\right)^{2}}{\sum_{i<j}(h_{ij}^{D})^{2}}\right] \quad (17)$$

En la segunda etapa del método se aplica el enfoque de los splines de placa delgada con el fin de estimar la función suavizada f que permite relacionar las ubicaciones originales p_i y las obtenidas mediante MDS q_i. Dicha función se plantea del siguiente modo:

$$f(p)=\alpha_0+\alpha_1 p^{(1)}+\alpha_2 p^{(2)}+\sum_{i=1}^{N}\beta_i\, u_i(p) \quad (18)$$

Siendo $u_i(p)=|p-p_i|^2 log|p-p_i|$, $p^{(j)}$ indica la j-ésima coordenada de la ubicación p y los $\alpha's$ y $\beta's$ son parámetros que deben ajustarse. Para el problema del espacio bidimensional el método desarrolla la función f como dos splines de placa delgada, f_1 y f_2 para las dos coordenadas de q_i. En esta etapa se incorpora un parámetro de suavizado λ, que le permite al usuario seleccionar un nivel deseado de suavidad.

Los parámetros $\alpha's$ y $\beta's$ se ajustan de modo que minimicen:

$$\sum_{i=1}^{N}\sum_{m=1}^{2}\left(q_{im}-f_j(p_i)\right)^2+\lambda[J_2(f_1)+J_2(f_2)] \quad (19)$$

Donde q_{i1} y q_{i2} corresponden a la primera y segunda coordenada de q_i, mientras que J_2 mide la suavidad de la función y se define así:

$$J_2(f)=\int_{R^2}\left[\left(\frac{\partial^2 f}{\partial x_1^2}\right)^2+2\left(\frac{\partial^2 f}{\partial x_1 \partial x_2}\right)^2+\left(\frac{\partial^2 f}{\partial x_2^2}\right)^2\right]dx_1 dx_2 \quad (20)$$

Cuando el parámetro de suavizado λ es igual a cero la función f se convierte en un spline de interpolación, mientras que hacer $\lambda \to \infty$ trae como efecto que los coeficientes β tiendan a cero

de modo que f sea tan solo una función lineal de mínimos cuadrados (Sampson & Guttorp, 1992).

Una vez se obtienen las funciones estimadas $\hat{f}$ y $\hat{g}$, es posible estimar el variograma entre cualquier par de ubicaciones p_1 y p_2 del espacio-G mediante los siguientes pasos (Le & Zidek, 2006):

- Obtener las ubicaciones correspondientes en el espacio-*D* mediante

$$q_j = f(p_j) \; j = 1{,}2$$

- Calcular la distancia en el espacio-D $\left|h_{12}^{D}\right|$ entre q_1 y q_2
- Evaluar el variograma, $2\gamma(h) = 2\hat{g}\left(\left|h_{12}^{D}\right|\right)$.

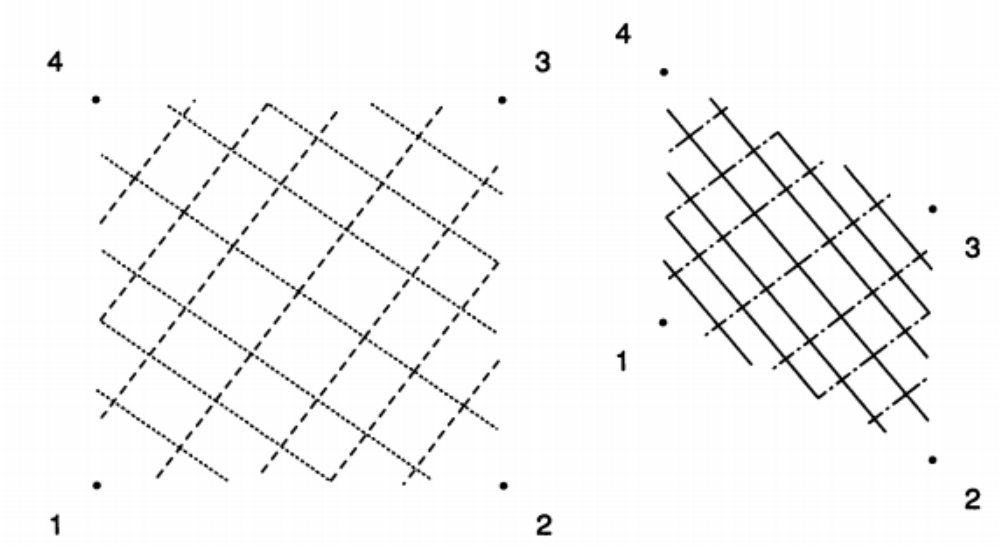


Figura 1. Transformación lineal del espacio-*G* (cuadrado) en el espacio-*D* (paralelogramo).

Otra herramienta utilizada en el método de Sampson y Guttorp es la cuadrícula biortogonal, la cual proporciona una representación gráfica de la estructura de covarianza espacial del proceso espacio-temporal. En la figura 1 se observa un ejemplo de una transformación lineal del espacio-*G* (cuadrado) en el espacio-*D* (paralelogramo). La red biortogonal rectangular que aparece a la derecha indica un estiramiento relativo del plano a lo largo del eje noroeste – sureste y una contracción en la dirección ortogonal; para un proceso espacial esto implica que

la covarianza espacial es más débil en la dirección noroeste – sureste y más fuerte en la dirección suroeste – noreste (Sampson & Guttorp, 1992).

En la figura 2 se aprecia un ejemplo de aplicación del método de Sampson y Guttorp, la gráfica de la izquierda muestra la representación del espacio-G con la ubicación de un grupo de estaciones de monitoreo de radiación solar en Columbia Británica (Canadá) y cómo esta configuración se transforma en el plano-D, mostrado en el centro, el cual se obtuvo aplicando escalamiento multidimensional. Las líneas punteadas representan la función de interpolación mediante splines de placa delgada que relacionan las ubicaciones de ambos planos. La gráfica de la derecha muestra la red biortogonal sobre el plano-G que representa la estructura de dispersión espacial de los datos de radiación solar (Sampson & Guttorp, 1992).

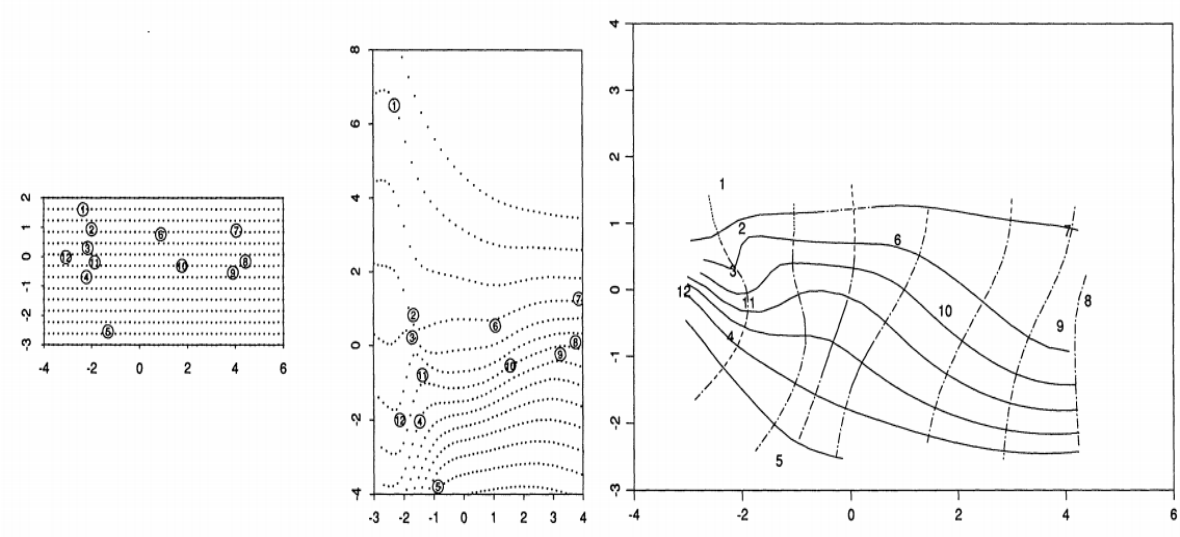


Figura 2. Transformación del espacio-*G* (izquierda) en el espacio-*D* (centro) y red biortogonal para el mapeo interpolado del plano-G al plano-D.

4. IMPLEMENTACIÓN DEL MÉTODO DE SAMPSON Y GUTTORP EN R

4.1. El paquete EnviroStat

El paquete *EnviroStat* permite implementar el procedimiento de deformación espacial de Sampson y Guttorp en R, proporcionando funciones para el modelamiento espaciotemporal de procesos y el diseño de redes de monitoreo para estos. Algunas de las características del paquete son las siguientes (Le & Zidek, 2013):

- Se asume que el proceso es un campo aleatorio gaussiano.
- En toda ubicación espacial, el proceso puede generar una multiplicidad de respuestas aleatorias, por ejemplo concentraciones de sustancias que contaminan la atmósfera.
- El enfoque usado en el paquete se ubica dentro de un marco de modelamiento jerárquico bayesiano. Sin embargo, por razones de conveniencia computacional se adopta un método simplificado en niveles altos de la configuración jerárquica. De esta forma se elimina la necesidad de especificación por parte del usuario de algunos parámetros en algunas funciones, lo cual permite al paquete manejar campos de redes de monitoreo grandes, conformados por 600 o más ubicaciones espaciales.
- En este paquete, al igual que en el método de Sampson y Guttorp, no se asume que el campo aleatorio gaussiano sea estacionario. En lugar de ello, se adopta un enfoque no-paramétrico en el que la matriz de covarianza espacial se deja completamente sin especificar y se dota con una distribución previa con una matriz de hipercovarianza que se puede modelar en el segundo nivel de la jerarquía, lo que hace que el método sea bastante robusto frente a la no-estacionariedad del campo aleatorio.

- Se permite la existencia de datos faltantes, siempre y cuando estos datos estén ausentes en bloques de tiempo, los cuales después de ajustar una tendencia regional, se tornan intercambiables. De ahí en adelante los bloques de residuos pueden permutarse para obtener un patrón en forma de escalera creciente o decreciente en la matriz de datos, siendo este un requisito para aplicar algunas funciones del paquete.
- El enfoque aplicado en el paquete genera intervalos de predicción bien calibrados; por ejemplo, un intervalo del 95% cubrirá sus predictandos alrededor de 95% de las veces.

La implementación del método de Sampson y Guttorp mediante el paquete *EnviroStat* comprende cinco etapas que son descritas a continuación (Le & Zidek, 2006).

Etapa 1. Determinación de una nueva configuración de las ubicaciones conocida como espacio-*D*. Para lograr este objetivo se siguen estos pasos:

1. Iniciar con la matriz de correlación espacial
2. Usar la función `Falternate3`, la cual emplea un algoritmo iterativo alternante que trata de reubicar óptimamente las estaciones en el espacio-D usando MDS y luego ajusta el variograma. La opción por defecto para esta función es el variograma exponencial, pero también existe la opción de emplear el semi-variograma gaussiano.

Los argumentos de la función `Falternate3` son los siguientes:

- `disp`: matriz de dispersión cuya dimensión es $n \times n$, esta se obtiene a partir de la matriz de correlación espacial mediante la siguiente expresión:

$$\mathbb{D} = 2 - 2\boldsymbol{C} \quad (21)$$

  Siendo $\mathbb{D}$ la matriz de dispersión y $\boldsymbol{C}$ la matriz de correlación.

- `coords`: matriz de coordenadas de dimensión $n \times 2$. Generalmente no se emplean las coordenadas geográficas originales de las estaciones sino una transformación de estas a un sistema de coordenadas rectangular mediante una proyección de Lambert usando la función `Flamb2`, incorporada en el paquete.

- `model`: tipo de variograma: 1 para el exponencial o 2 para el gaussiano.
- `a0, t0`: estimados iniciales de los parámetros del variograma.
- `max.iter, max.fcal`: parámetros de control para las llamadas a las rutinas de optimización no-lineal, se utilizan los mismos valores en la etapa de MDS y en la etapa del variograma.
- `alter.lim`: número máximo de iteraciones de pares de llamados alternantes a las funciones `Fmdsfit3`, la cual estima las coordenadas de las ubicaciones en el espacio-D mediante MDS, y `Fvariogfit3`, encargada de realizar el ajuste del variograma.
- `tol`: criterio de convergencia para los estimados de las coordenadas.
- `dims`: número de dimensiones utilizadas en el escalamiento multidimensional.
- `lambda`: parámetro de suavizado
- `ncoords`: matriz opcional de coordenadas iniciales en caso de no disponer de plano-G.
- `dev.mon`: función para abrir el dispositivo usado para las gráficas que monitorean la convergencia del objetivo.
- `verbose`: cuando asume el valor TRUE muestra los resultados de cada iteración en la consola.

Luego de aplicar la función `Falternate3` se obtiene una lista que contiene los siguientes elementos:

- `variogfit`: parámetros del variograma ajustado con las nuevas ubicaciones.
- `ncoords`: matriz de coordenadas de las nuevas ubicaciones cuya dimensión es $n \times 2$.

Etapa 2. Ajuste de un spline de placas delgadas suavizado entre las ubicaciones originales y las del espacio-D identificadas en la fase anterior. En esta fase se permite al usuario observar la deformación del espacio geográfico en el espacio-D y seleccionar un valor adecuado para el parámetro de suavizado del spline de placas delgadas. Esto se logra mediante la función

`Ftransdraw`, la cual es interactiva, muestra el variograma ajustado y la función de transformación del espacio geográfico en el espacio-D.

La función `Ftransdraw` utiliza los siguientes argumentos:

- `disp`: matriz de dispersión cuya dimensión es $n \times n$.
- `Gcrds`: matriz de coordenadas de las ubicaciones en el espacio-G, de dimensión $n \times 2$. Generalmente se emplea una transformación de las coordenadas geográficas a un sistema de coordenadas rectangular mediante una proyección de Lambert.
- `MDScrds:` coordenadas de las ubicaciones en el espacio-D obtenidas mediante la función `Falternate3`.
- `gridstr`: coordenadas de la cuadrícula, obtenidas a partir de la función `Fmgrid` usando como argumento `Gcrds`.
- `sta.names`: nombres de las ubicaciones, en caso de no proporcionarse las ubicaciones se numeran de 1 a n.
- `lambda`: valor inicial del parámetro de suavizado.
- `lsq`: señal lógica utilizada en el método de Sampson y Guttorp.
- `eye`: perspectiva visual, en caso de no ser suministrada las ubicaciones serán seleccionadas usando los datos proporcionados.
- `model`: tipo de variograma: 1 para el exponencial o 2 para el gaussiano.
- `a0, t0`: estimados iniciales de los parámetros del variograma.

En esta fase se crea una cuadrícula de puntos que abarca el rango de estaciones con la función `Fmgrid`, cuyas coordenadas son utilizadas por la función `Ftransdraw` para ajustar splines de placa delgada entre las ubicaciones del espacio-G y del espacio-D y aplica el spline ajustado a los puntos de la cuadrícula del espacio-G. Posteriormente grafica los puntos correspondientes a la cuadrícula en el espacio-D permitiendo al usuario escoger de forma interactiva un valor apropiado para el parámetro de suavizado. El objetivo al escoger el parámetro λ es conseguir un

balance entre la búsqueda de estacionariedad en el modelo frente a la necesidad de mantener una superficie que no esté tan deformada para que se pierda la interpretabilidad. En efecto, no es deseable que el espacio-D esté doblado, ya que implica que dos sitios más lejanos podrían tener mayor correlación que la correspondiente a aquellos ubicados entre ellos.

Etapa 3. Obtención de un spline de placa delgada óptimo combinando los resultados de las etapas anteriores. La función `sinterp` se utiliza para ajustar el spline de placa delgada con el parámetro de suavizado seleccionado en el paso anterior.

La función `sinterp` emplea los siguientes argumentos:

- `x`: matriz de nodos de dimensión $nq \times nk$, siendo nq la dimensión del espacio de dominio, y nk el número de nodos.
- `y`: matriz de valores de la función en cada nodo cuya dimensión es $np \times nk$, donde np es la dimensión del espacio de imagen.
- `m`: un entero tal que el orden del spline es igual a $2 * m$, el valor usado por defecto es 2.
- `lam`: vector de parámetros de suavizado. En caso de faltar este argumento o cuando es igual a 0 se lleva a cabo interpolación.
- `lsq`: si es igual a TRUE se sustrae el ajuste de mínimos cuadrados de `y`, generando los coeficientes del polinomio como coeficientes de la porción de mínimos cuadrados.

Los coeficientes estimados α's y β's de la ecuación 18 que corresponden al spline de placa delgada optimizado, se guardan en el resultado `sol` de la función `sinterp`. Estos coeficientes son utilizados como argumento por la función `bgrid` para evaluar la cuadrícula biortogonal, la cual representa la contracción y expansión del spline de placa delgada que caracteriza la deformación del espacio-G.

Etapa 4. Estimación de la dispersión entre las nuevas ubicaciones de interés y las estaciones usando el ajuste efectuado en los pasos anteriores. Esto se logra mediante el spline de placa delgada de la etapa 3 y el variograma correspondiente ajustado en la etapa 1. En primer lugar es

necesario convertir las coordenadas geográficas de las ubicaciones de interés a coordenadas de Lambert usando el mismo punto de referencia de la etapa 1, a continuación se evalúan sus ubicaciones correspondientes en el espacio-D mediante el spline de placa delgada optimizado, finalmente se calculan las correlaciones usando los parámetros del variograma ajustado y las distancias entre los sitios en el espacio-D.

La función `corrfit` se utiliza para estimar las correlaciones entre todas las ubicaciones, tanto las nuevas como las estaciones, dicha función tiene los siguientes argumentos:

- `crds`: coordenadas de todas las ubicaciones comenzando con las de los sitios nuevos de interés.
- `Tspline`: ajuste mediante el spline de placa delgado desarrollado en las etapas anteriores.
- `sg.fit`: resultado obtenido en la etapa 1.
- `model`: tipo de variograma: 1 para el exponencial o 2 para el gaussiano.

Etapa 5. Interpolación del campo de varianza. En esta fase se estiman las varianzas del campo aleatorio en todas las ubicaciones y luego se combinan con la matriz de correlación estimada en la etapa anterior para obtener un estimado de la matriz de covarianza.

Las varianzas en cada ubicación se estiman mediante la función `seval`, usando el spline de placa delgada obtenido mediante la función `sinterp`. Los argumentos usados por la función `seval` son los siguientes:

- `x`: matriz de dimensión $n \times 2$, la cual contiene las coordenadas de las ubicaciones.
- `tpsp`: solución de spline de placa delgada, retornada típicamente por la función `sinterp`, la cual estima coeficientes para splines de placa delgada suavizados de dimensión arbitraria.

Después de la estimación de las varianzas se emplea nuevamente la función `corrfit` para estimar la matriz de covarianza para todas las ubicaciones y completar de este modo el procedimiento de Sampson y Guttorp.

4.2. Tratamiento computacional mediante smacof y un procedimiento de validación cruzada

Para el uso de la función `Falternate3` en la etapa 1 de la implementación del método de Sampson y Guttorp descrita en el apartado anterior, es necesario que las coordenadas de las ubicaciones tengan una escala razonablemente pequeña antes de utilizar la función.

En caso de no cumplirse este requisito es probable que no funcione la inversión de la matriz en el cálculo de la matriz de energía de flexión (Le *et al.*, 2015), la cual hace parte de la obtención del spline de placa delgada. En efecto, la medida de suavizado J_2 (definida en la ecuación 20) de la función f que permite relacionar las ubicaciones del espacio-G con las del espacio-D, es proporcional a la energía de flexión de una placa delgada idealizada de extensión infinita. Los interpoladores que minimizan esta energía de flexión son funciones no-lineales definidas por combinaciones lineales de funciones base centradas en las N observaciones del espacio-G, estos interpoladores son los que aparecen en el término de la sumatoria de la ecuación 18 (Sampson & Guttorp, 1992).

Con el fin de obtener coordenadas cuya escala sea pequeña, los creadores del paquete *EnviroStat* sugieren dividir las coordenadas de las ubicaciones del espacio-G obtenidas mediante la proyección de Lambert por un factor, en el caso del ejemplo de aplicación del paquete este factor es igual a 10 (Le & Zidek, 2013).

Dado que la elección del factor es arbitraria por parte del usuario y no se define en la documentación del paquete que tan pequeña debe ser la escala de las coordenadas de las

ubicaciones, se propone a continuación un método de normalización que permite obtener una escala adecuada en las coordenadas para implementar el procedimiento de Sampson y Guttorp en R.

El método propuesto –el cual ha sido empleado en trabajos como el de Vera, Angulo y Roldán (2016)–, consiste en crear una matriz de distancias $\mathcal{D}$ entre las ubicaciones del espacio-G cuyas coordenadas han sido transformadas mediante la proyección de Lambert. Las distancias se calculan mediante la función `Fdist`, disponible en el paquete *EnviroStat*. Para la normalización de las coordenadas se asume que existe una constante c tal que:

$$c^2 \sum_{i<j} (d_{ij})^2 = \frac{n(n-1)}{2} \quad (22)$$

Donde d_{ij} es cada uno de los elementos de la matriz de distancias $\mathcal{D}$ y n es el número de estaciones de monitoreo. Tal como se aprecia en la ecuación anterior los términos incluidos en la sumatoria son únicamente aquellos que están por debajo de la diagonal principal de la matriz $\mathcal{D}$.

Una vez calculada la constante c mediante la ecuación 22, esta se multiplica por la matriz de coordenadas de las ubicaciones del espacio-G transformadas mediante la proyección de Lambert y este resultado es el que se emplea como argumento `coords` en la función `Falternate3`. El mismo procedimiento de normalización se aplica directamente a la matriz de dispersión $\mathbb{D}$ definida en la ecuación 21, la cual una vez normalizada pasa a ser el argumento `disp` de esta misma función. Estos procedimientos de estandarización evitan entre otras cosas que se obtengan soluciones degeneradas en el MDS (Vera *et al*., 2016).

Otro cambio efectuado en la implementación del método de Sampson y Guttorp fue la incorporación del paquete *smacof* para obtener la configuración conocida como espacio-D mediante escalamiento multidimensional en la etapa 1. Para ello se modificó el código del

programa SG.R utilizado por *EnviroStat* para implementar el método; el programa que incluye los cambios se denomina SG_nuevo.R.

En primer lugar se añadieron los argumentos type y spline.degree a la función Falternate3_nuevo, de modo que el usuario pueda seleccionar con el argumento type la variante de MDS a implementar mediante el paquete *smacof*; en caso de seleccionar la versión mspline es necesario especificar el grado del spline para esta variante con el argumento spline.degree. Por defecto, la función Falternate3_nuevo implementa la variante de MDS mspline usando spline de grado 4.

```
require(smacof)
Falternate3_nuevo <- function(disp, coords, model= 1., a0=0.1, t0=0.5,
      max.iter= 50., max.fcal=100., alter.lim=50., tol= 1e-05, prt=0.,
      dims = 2., lambda = 0., ncoords, dev.mon = NULL, verbose =FALSE,
      type = 'mspline', spline.degree = 4)
```

Estos argumentos de la función Falternate3_nuevo pasan a la función auxiliar Fmdsfit3_nuevo, la cual es usada para ajustar las coordenadas de las ubicaciones en el espacio-D dado un conjunto de parámetros del variograma.

```
Fmdsfit3_nuevo <- function(disp.lt, coords, model = 1., a, t0,
      max.iter= 25., max.fcal = 100., prt = 0., dims = 2., lambda= 0.,
      bem, verbose = FALSE, type =type, spline.degree = spline.degree)
```

En la última parte de la función Fmdsfit3_nuevo se añadieron las siguientes líneas de código que permiten obtener la configuración de MDS mediante el paquete *smacof*.

```
coordenadas <-smacofSym(disp, ndim = dims, type = type,
                        spline.degree = spline.degree)
ncoords <- (coordenadas$conf)
```

La función smacofSym del paquete *smacof* implementa una de las versiones del algoritmo SMACOF retornando una lista cuyo principal resultado conf, es la matriz de la configuración ajustada cuyas filas son las coordenadas de las ubicaciones en el espacio-D. Esta función utiliza

como argumentos la matriz de dispersión normalizada a través del procedimiento descrito anteriormente, el número de dimensiones usadas en el MDS `dims`, y el tipo de MDS que será implementado para conseguir el ajuste, `type`. Existen básicamente tres variantes de MDS aceptadas por la función `smacofSym`: *razón* (opción por defecto), *intervalo* (transformación polinómica) y *ordinal* o *no-métrica*. El MDS tipo *mspline* corresponde a un spline monótono que permite transformar suavemente las proximidades (en este caso los elementos de la matriz de dispersión) para obtener las disparidades.

4.2.1. Selección del parámetro de suavizado mediante validación cruzada

La última modificación introducida al paquete *EnviroStat* es la incorporación de una metodología para la selección del parámetro de suavizado λ. En la versión original del paquete la selección es efectuada por el usuario de forma interactiva utilizando la función `Ftransdraw`, la cual ajusta splines de placas delgadas entre las ubicaciones del espacio-G y del espacio-D y aplica este spline ajustado a los puntos de una cuadrícula en el espacio-G (creada con anterioridad mediante la función `Fmgrid`) que es observada en primer lugar cuando se ejecuta la función. Al hacer clic sobre la cuadrícula en el espacio-G el usuario puede observar la cuadrícula correspondiente en el espacio-D y el variograma ajustado cuando $\lambda = 0$. A continuación se pide al usuario ingresar un nuevo valor para λ o pulsar *enter* para detener la ejecución. Cada vez que se introduce un valor para λ y se hace clic en la ventana de gráficos se puede apreciar la deformación del plano-G en el plano-D, el variograma correspondiente a dicho valor del parámetro de suavizado y la magnitud de la raíz cuadrada del error cuadrático medio (rmse) que corresponde al ajuste del variograma.

El único criterio dado en la versión original del paquete para la selección del parámetro λ consiste en ensayar valores de dicho parámetro de modo que se pueda obtener un balance entre la búsqueda de estacionariedad en el modelo frente a la necesidad de mantener una superficie

cuya deformación no sea tan alta de forma que permita su interpretación adecuada. Expresado en otros términos, cuando $\lambda = 0$ la magnitud de la raíz cuadrada del error cuadrático medio es mínima, ya que los splines de placa delgada se convierten en splines de interpolación y por lo tanto se consigue el mejor ajuste del variograma, pero esto trae como consecuencia la generación de un espacio-D cuya deformación es máxima, lo cual no es deseable tal como se explicó anteriormente. Por otro lado, cuando λ es muy grande se minimiza la deformación del espacio-G en el espacio-D, pero empeora el grado de ajuste conseguido con el variograma y se corre el riesgo de obtener un modelo sobreajustado, de modo que la estimación o predicción de la estructura de covarianza espacial en lugares que no han sido monitoreados será menos confiable.

Dado que en la versión original del paquete no se proporciona al usuario un criterio más claro para la selección del parámetro de suavizado, se desarrolló una metodología que consiste en adoptar el enfoque de validación cruzada. En este procedimiento se calcula para cada valor de λ el error cuadrático medio en la predicción de las dispersiones (Sampson & Guttorp, 1992). Es decir, si temporalmente se deja por fuera[1] el i-ésimo elemento por debajo de la diagonal principal de la matriz de dispersión se calcula la función $g^{(i)}$ y luego se evalúa la sumatoria de errores cuadráticos dada por:

$$\sum_{k} \left(\mathbb{d}_k^2 - g^{(i)}\left(\left|h_k^D\right|\right) \right)^2 \quad (23)$$

En la ecuación anterior $\mathbb{d}$ es un vector creado con los elementos situados por debajo de la diagonal principal de la matriz de dispersión, $|h^D|$ representa un vector obtenido con los elementos situados por debajo de la diagonal principal de la matriz de distancias entre las ubicaciones en el plano-D, $g^{(i)}$ corresponde al modelo de variograma que permite el cálculo de las dispersiones obtenido al eliminar temporalmente la i-ésima observación y el índice k

[1] Este método es conocido como LOOCV o validación cruzada dejando uno fuera. Para mayor información se puede consultar la siguiente página: https://www.autonlab.org/tutorials/overfit10.pdf

representa cada uno de los términos restantes en los vectores luego de descartar temporalmente el i-ésimo dato.

Una vez se calculan las sumas de errores cuadráticos al dejar por fuera una de las observaciones, estas se promedian y se obtiene así el error de validación cruzada que corresponde al parámetro λ en consideración.

El algoritmo implementado para la selección del parámetro de suavizado λ se describe con mayor detalle a continuación:

- *Paso 1*. Definir un conjunto de valores de λ para ser ensayados.
- *Paso 2*. Calcular para cada valor de λ el error de validación cruzada mediante el siguiente procedimiento:
  - Ajustar para este valor de λ el spline de placa delgada que relaciona los puntos del plano-G con los del plano-D.
  - Evaluar las coordenadas de los puntos en el plano-D correspondientes al spline desarrollado en la etapa anterior.
  - Obtener la matriz de distancias entre los puntos del plano-D.
  - Crear el vector de distancias $|h^D|$ seleccionando los elementos que están por debajo de la diagonal principal de la matriz de distancias entre los puntos del plano-D.
  - Crear el vector de dispersiones $\mathbb{d}^2$ seleccionando los elementos situados por debajo de la diagonal principal de la matriz de dispersión.
  - Eliminar temporalmente el i-ésimo elemento tanto del vector de distancias como del vector de dispersiones.
  - Ajustar con el resto de elementos de ambos vectores un modelo de variograma isotrópico para obtener estimados de las dispersiones. Se puede seleccionar el modelo exponencial o el gaussiano.
  - Calcular la suma de errores cuadráticos en la estimación de las dispersiones usando la ecuación 23.

  - Reintegrar los elementos excluidos a los vectores, eliminar temporalmente el siguiente elemento tanto del vector de distancias como del vector de dispersiones y repetir los dos pasos anteriores hasta que hayan sido eliminados de forma transitoria todos los elementos.
  - Obtener el error de validación cruzada promediando las sumas de errores cuadráticos en la estimación de las dispersiones.

- *Paso 3*. Generar una gráfica que represente el error de validación cruzada frente a λ.
- *Paso 4*. Seleccionar como parámetro de suavizado del modelo un valor de λ próximo a la asíntota de la curva generada en la fase anterior.

Para el cálculo del error de validación cruzada se creó la función `LOOCVcal`, la cual fue incorporada al programa `SG_nuevo.R`, sus argumentos son los siguientes:

- `disp`: matriz de dispersión cuya dimensión es $n \times n$.
- `Gcrds`: matriz de coordenadas de las ubicaciones en el espacio-G, de dimensión $n \times 2$. Generalmente se emplea una transformación de las coordenadas geográficas a un sistema de coordenadas rectangular mediante una proyección de Lambert.
- `MDScrds:` coordenadas de las ubicaciones en el espacio-D obtenidas mediante la función `Falternate3_nuevo`.
- `lam`: valor del parámetro de suavizado, por defecto éste es igual a 0.
- `lsq`: señal lógica utilizada en el método de Sampson y Guttorp.
- `model`: tipo de variograma: 1 para el exponencial o 2 para el gaussiano.
- `a0, t0`: estimados iniciales de los parámetros del variograma.

Esta función es utilizada para generar una gráfica de errores de validación cruzada frente a valores del parámetro de suavizado la cual proporciona un criterio más claro para la selección de λ que el disponible en la versión original de *EnviroStat*, ya que el error de validación cruzada es un indicador de la tendencia del modelo al sobreajuste. De esta manera se facilita la

obtención de un balance adecuado entre el ajuste conseguido con el variograma y la deformación del espacio-G en el espacio-D.

Una ventaja del método de normalización de las matrices de dispersión y de coordenadas transformadas de las ubicaciones en el espacio-G introducido en la versión modificada de *EnviroStat*, es que el intervalo de parámetros de suavizado necesario para obtener el balance adecuado entre el ajuste del variograma y la deformación espacial está entre 0 y 1, lo cual simplifica el procedimiento de validación cruzada tal como se mostrará en los ejemplos de aplicación. En el caso de la versión original del paquete la magnitud del parámetro de suavizado requerido para conseguir dicho balance es considerablemente mayor, complicando así la tarea de elección de λ, ya que es necesario en este caso ensayar una gama más amplia de valores de λ.

Después de escoger el parámetro λ teniendo en cuenta el diagrama de error de validación cruzada se utiliza la función `Ftransdraw_nuevo`, la cual emplea los mismos argumentos que la función `Ftransdraw` de la versión original. Estas funciones cumplen roles distintos, ya que en la versión original la función `Ftransdraw` es la única que sirve para seleccionar λ, mientras que en la versión modificada la función `Ftransdraw_nuevo` se usa para confirmar que el parámetro λ escogido mediante validación cruzada permite obtener el balance adecuado entre el ajuste del variograma y la deformación espacial; en efecto, una vez que se ejecuta esta función el usuario puede observar en la ventana de gráficos la deformación del plano-G en el plano-D y el variograma correspondiente al parámetro λ seleccionado y compararlos con los que corresponden a $\lambda = 0$.

5. EJEMPLOS DE APLICACIÓN

5.1. Comparación del paquete EnviroStat original con el modificado

Con el fin de llevar a cabo la comparación de los resultados obtenidos al implementar el procedimiento de Sampson y Guttorp mediante el paquete *EnviroStat* original y la versión que incluye las modificaciones descritas en el apartado anterior se efectuaron varias pruebas utilizando el conjunto de datos `ozone.NY` que hace parte del paquete. Estos datos corresponden a mediciones de la concentración de ozono en ppb (partes por billón) realizadas cada hora en 9 estaciones de monitoreo del Estado de Nueva York descargadas de la base de datos AIRS (Aerometric Information Retrieval System) de la Agencia de Protección Ambiental de Estados Unidos. Cada fila del conjunto de datos representa un registro diario que inicia en abril 1 de 1995 y finaliza en septiembre 30 de 1995 abarcando un período de 183 días. El análisis inicia con la carga del paquete *EnviroStat* y la observación de la estructura del conjunto de datos:

```
> library(EnviroStat)
> data(ozone.NY)
> str(ozone.NY)
```

```
'data.frame':  183 obs. of  38 variables:
 $ month  : int  4 4 4 4 4 4 4 4 4 4 ...
 $ weekday: int  7 8 2 3 4 5 6 7 8 2 ...
 $ sqO3.1 : num  6.16 6.24 5.92 5.74 5.74 ...
 $ sqO3.2 : num  6.32 6.63 6.71 5.92 5.83 ...
 $ sqO3.3 : num  6.4 6.93 7.07 5.83 5.83 ...
 $ sqO3.4 : num  6.56 7 7.07 6.56 5.92 ...
 $ sqO3.5 : num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.6 : num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.7 : num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.8 : num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.9 : num  5.74 5.83 5.39 5.48 5.57 ...
 $ sqO3.10: num  6.08 6.16 5.92 5.48 5.57 ...
 $ sqO3.11: num  6.16 6.48 6.16 5.39 5.66 ...
 $ sqO3.12: num  6.16 6.56 6.4 5.83 5.66 ...
 $ sqO3.13: num  5.48 5.2 2.45 2.24 4.69 ...
 $ sqO3.14: num  5.92 6.16 4.36 3.61 5 ...
 $ sqO3.15: num  6.16 6.56 5.1 3.87 5.1 ...
 $ sqO3.16: num  6.16 6.63 5.37 4.36 5.1 ...
 $ sqO3.17: num  NA NA NA 5.29 5.88 ...
 $ sqO3.18: num  NA NA NA 5.1 6.69 ...
 $ sqO3.19: num  NA NA NA 5 6.61 ...
 $ sqO3.20: num  NA NA NA 5.39 5.48 ...
 $ sqO3.21: num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.22: num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.23: num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.24: num  NA NA NA NA NA NA NA NA NA NA ...
 $ sqO3.25: num  6.24 5.39 4.8 6.08 5.57 ...
 $ sqO3.26: num  6.48 6.24 6.24 6.08 6.31 ...
 $ sqO3.27: num  6.56 6.63 6.56 6.16 6.63 ...
 $ sqO3.28: num  6.63 6.63 6.78 6.32 6.67 ...
 $ sqO3.29: num  6.32 5.57 6.32 6.16 5.48 ...
 $ sqO3.30: num  6.48 6.48 6.71 6.32 5.66 ...
 $ sqO3.31: num  6.56 6.63 6.78 6.48 5.66 ...
 $ sqO3.32: num  6.56 6.71 6.86 6.51 5.74 ...
 $ sqO3.33: num  5.92 4.47 5.66 2.24 3.09 ...
 $ sqO3.34: num  6.16 5.39 5.74 3.87 5.39 ...
 $ sqO3.35: num  6.32 6.16 6.08 5.29 5.48 ...
 $ sqO3.36: num  6.4 6.4 6.48 5.39 5.48 ...
```

La primera variable es el mes en el que se efectuó la medición, cuyo valor es un entero comprendido en el intervalo [4, 9]; la siguiente variable es el día de la semana, un entero que es igual a 2 si el día de la medición es lunes y así sucesivamente hasta 8 para el día domingo. En este conjunto de datos se extraen únicamente las mediciones registradas entre las 8:00 y 12:00 de la mañana, de modo que a cada estación le corresponden cuatro columnas, puesto que son 9 las estaciones de monitoreo el número de variables restantes es 36. El análisis preliminar del conjunto de datos omitido en el presente apartado, muestra que es conveniente utilizar la raíz cuadrada de las concentraciones de ozono en lugar de las concentraciones originales con el fin de obtener distribuciones más simétricas, por tal motivo las variables `sqO3.1–sqO3.4`

representan la raíz cuadrada de las mediciones de la concentración de ozono entre las 8 – 12 horas para la estación 1, mientras que las variables `sqO3.33–sqO3.36` corresponden a la raíz cuadrada de las mediciones del nivel de ozono entre las 8 – 12 horas para la estación 9.

La hoja de datos `location.NY` que también hace parte del paquete, contiene las coordenadas geográficas de cada estación de monitoreo.

```
> data(location.NY)
> location.NY
      lat     long
1 42.8975 -73.2508
2 42.6367 -73.1686
3 43.0122 -73.6492
4 42.6769 -73.7555
5 42.7244 -73.4316
6 42.1378 -74.5147
7 43.4556 -74.5150
8 42.7306 -75.7861
9 43.3047 -75.7216
```

Los creadores del paquete *EnviroStat* sugieren efectuar una permutación en el orden de los sitios en la hoja de datos, de manera que se obtenga un patrón monótono de datos faltantes que luzca como una escalera ascendente o descendente; este patrón simplifica considerablemente el análisis. Para obtener dicho patrón se calcula en primer lugar el número de observaciones faltantes en cada sitio.

```
> sqO3 <- ozone.NY[,3:38]
> sitenumber <- apply(is.na(sqO3), 2, sum)
> missingnum <- {}
> k <- 0
> for (i in 1:9){
+   for (j in 1:4){
+     k <- k + sitenumber[4*i-j+1]
+   }
+   missingnum[i] <- k
+   k <- 0
+ }
> order(missingnum,decreasing = T)
[1] 2 6 5 1 3 4 7 8 9
```

El conteo anterior muestra que el orden de las estaciones debe permutarse a (2, 6, 5, 1, 3, 4, 7, 8, 9) para conseguir el patrón monótono requerido.

```
> norder <- c(order(missingnum,decreasing = T))
> tt <- NULL
> for (i in 1:9) tt <- c(tt, c(1:4) + 4 * (norder[i]-1))
```

```
> ndata <- sqO3[,tt]
> nloc <- location.NY[norder, ]
```

El mapa de la figura 3, el cual muestra la ubicación de las estaciones en el espacio-G, se generó utilizando la función `map` del paquete maps.

```
> library(maps)
> dev.new(width=5, height=4)
> map('state',region='new york')
> points(nloc[,2],nloc[,1],pch=19,col="red",cex=0.6)
> text(x=nloc[,2]+0.01,y=nloc[,1]+0.01,labels=c(1:9),cex=1)
```

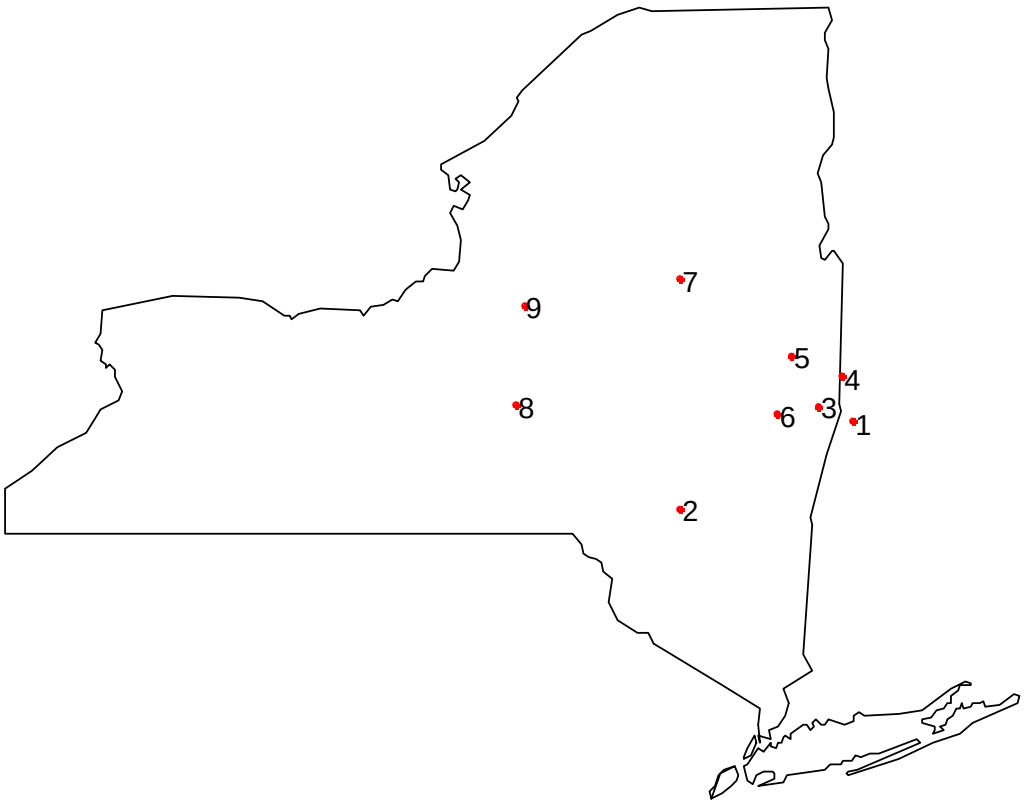


Figura 3. Ubicación geográfica de las estaciones de monitoreo

Antes de implementar la primera etapa del método de Sampson y Guttorp es necesario transformar las coordenadas mediante su proyección sobre una superficie plana, ya que esto permite un mejor cálculo de las distancias entre las ubicaciones. El paquete dispone de la función `Flamb2` para obtener la proyección de Lambert de las coordenadas, cuyo resultado es una lista que contiene la matriz de las coordenadas de la proyección `(xy)` así como las

coordenadas de un punto utilizado como referencia y el rango de latitudes empleado en la proyección.

```
> coords <- Flamb2(abs(nloc))
> coords$xy
            x          y
2  106.949788 -16.943913
6   -3.076874 -73.195581
5   85.338117  -7.502412
1   99.811622  11.923342
3   67.267631  24.269406
4   58.952310 -13.055823
7   -3.036232  73.196942
8 -106.787916  -6.514160
9 -100.584494  57.174524
```

A continuación se mostrará la implementación de las etapas 1 y 2 del método de Sampson y Guttorp en la versión original del paquete *EnviroStat* y posteriormente en la versión modificada, ya que en éstas es donde se aprecian las modificaciones introducidas en el paquete.

5.1.1. Implementación mediante la versión original de EnviroStat

En la versión original del paquete es necesario que las coordenadas de las ubicaciones tengan una escala razonablemente pequeña antes de utilizar la función `Falternate3` para que el escalamiento multidimensional y el spline de placa delgada funcionen adecuadamente. Esto se logra dividiendo las coordenadas transformadas mediante la proyección de Lambert por un factor que en este caso es igual a 10.

```
> coords.lamb <- coords$xy / 10
```

La matriz de dispersión se obtiene a partir de la matriz de correlación. Por ahora se omiten los detalles del cálculo de la matriz de correlación, los cuales serán mostrados en el próximo ejemplo de aplicación.

```
> disp <- 2 - 2 * corr.est
```

La determinación de una nueva configuración de las ubicaciones conocida como espacio-*D* se consigue con la función `Falternate3.`

```
> sg.est <- Falternate3(disp, coords.lamb, max.iter = 100,
+                       alter.lim = 100, model = 1)
```

A medida que esta función se ejecuta se muestran en la ventana de gráficos el plano-D y el variograma correspondientes a cada iteración. En la figura 4 se aprecian el plano-D y el variograma iniciales. Al hacer la comparación con el mapa de la figura 3 se observa que en la iteración 0 el plano-D coincide con el plano-G. La figura 5 muestra los resultados de la última iteración, las flechas representan el movimiento de las estaciones a partir de la ubicación original. Se aprecia que luego de 82 iteraciones el error cuadrático medio en el ajuste del variograma disminuye de 0.0967 a 0.0417. En la figura 6 se aprecia el plano-D obtenido mediante MDS luego de utilizar la función `Falternate3.`

```
> plot(sg.est$ncoords,type="n")
> text(sg.est$ncoords[,1], sg.est$ncoords[,2], c(1:9))
```

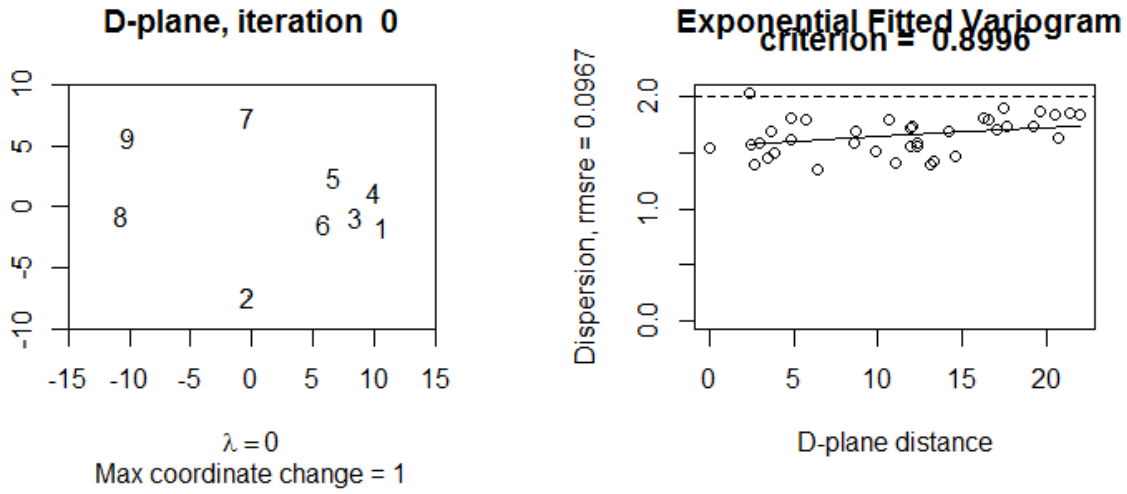


Figura 4. Plano-D y variograma exponencial ajustado iniciales

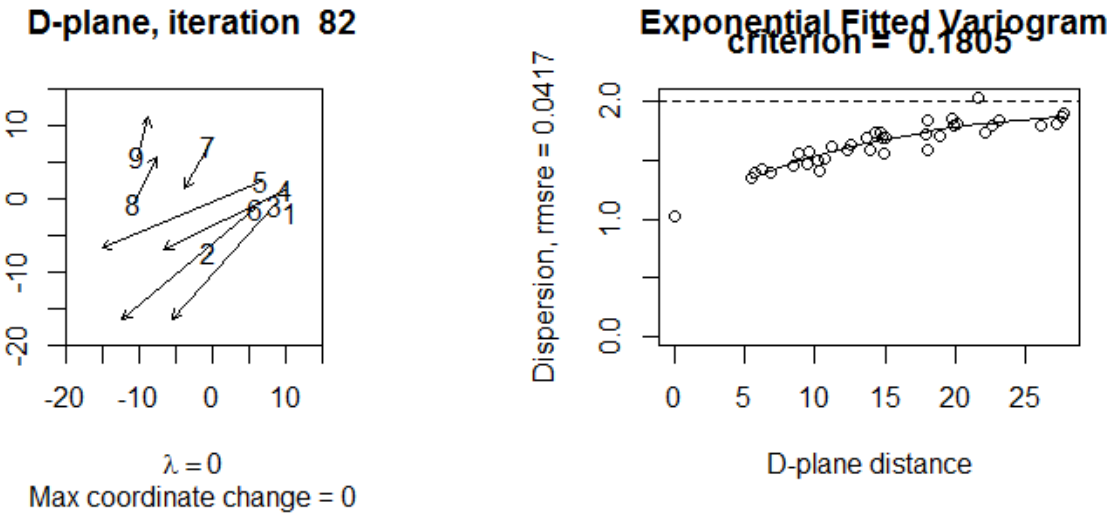


Figura 5. Plano-D y variograma exponencial ajustado en la última iteración

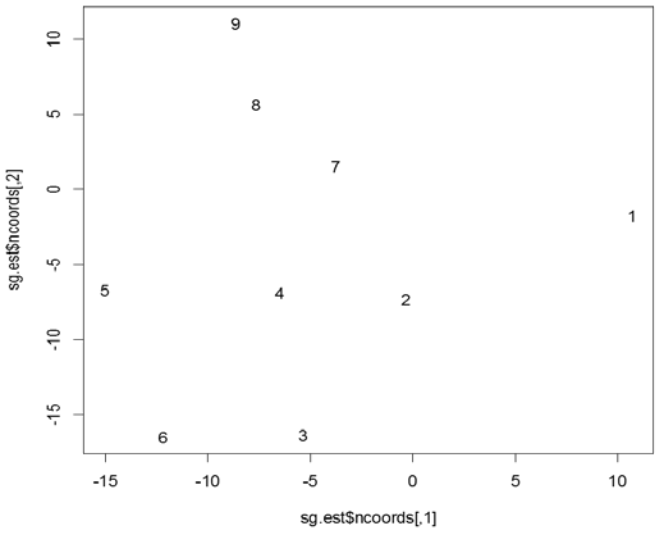


Figura 6. Plano-D obtenido mediante MDS utilizando la función `Falternate3`

En la etapa 2 del método de Sampson y Guttorp se ajusta un spline de placas delgadas suavizado entre las ubicaciones del plano-G y las del plano-D identificadas en el paso anterior. La función `Ftransdraw` se emplea para este propósito, permitiendo observar la deformación del espacio geográfico en el espacio-D. Antes de aplicar esta función es necesario crear una cuadrícula de puntos que abarca el rango de coordenadas de las estaciones con la función `Fmgrid`.

```
> coords.grid <- Fmgrid(range(coords.lamb[,1]),
```

```
+                        range(coords.lamb[,2]))
> par(mfrow = c(1, 2))
> temp <- setplot(coords.lamb, axis = TRUE)
> deform <- Ftransdraw(disp = disp, Gcrds = coords.lamb,
+                      MDScrds = sg.est$ncoords,
+                      gridstr = coords.grid)
Click anywhere on plot to continue (Left button)
Enter value for new lambda (Hit return to stop)
1:
```

Esta función es interactiva, ya que demanda al usuario ingresar un valor del parámetro de suavizado λ, mostrando en primer lugar la deformación de la cuadrícula del plano-G (figura 7) al plano-D cuando $\lambda = 0$ (figura 8). La selección del parámetro de suavizado implica encontrar un balance entre el ajuste logrado en el variograma (medido por el error cuadrático medio) y la suavidad de la deformación. La selección de $\lambda = 50$ muestra ser apropiada según la figura 9.

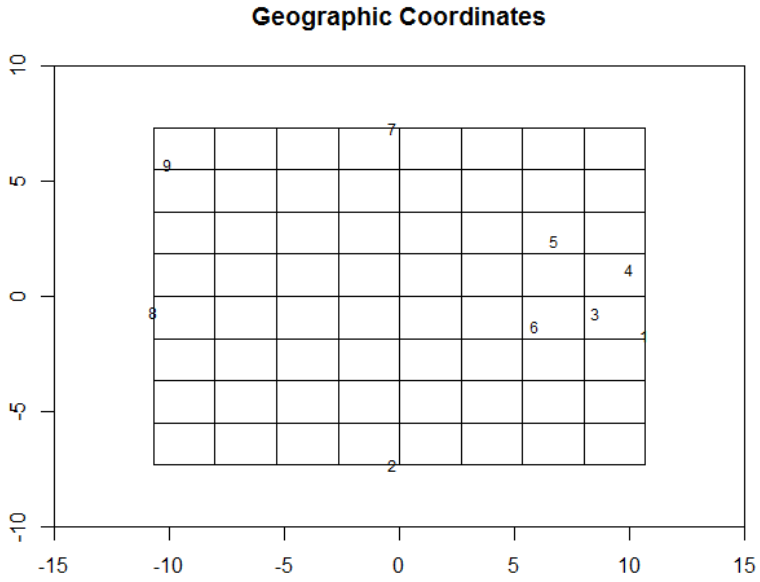


Figura 7. Cuadrícula del plano-G mostrando la ubicación de las estaciones

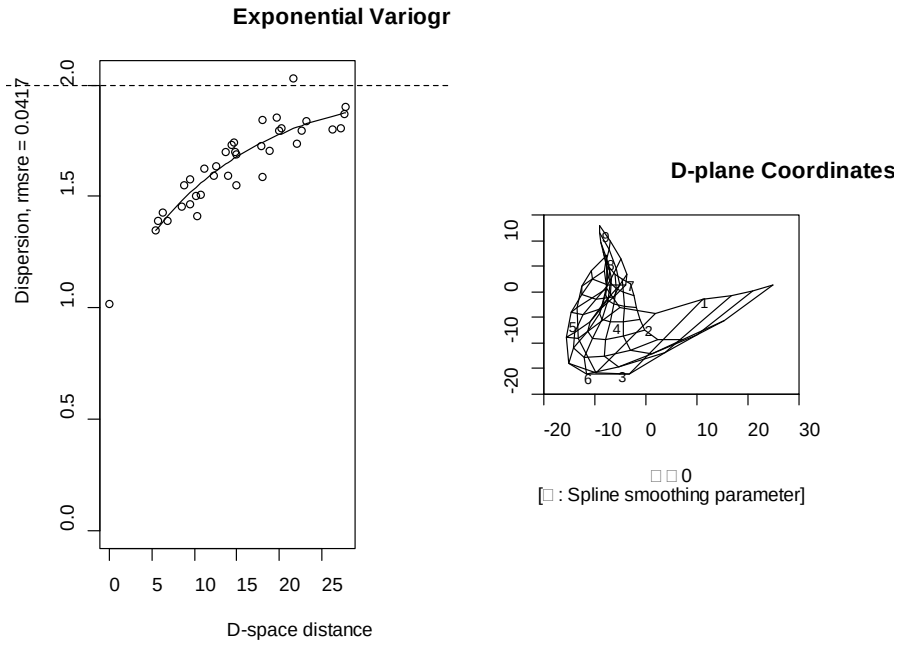


Figura 8. Cuadrícula del plano-D y variograma ajustado para $\lambda = 0$.

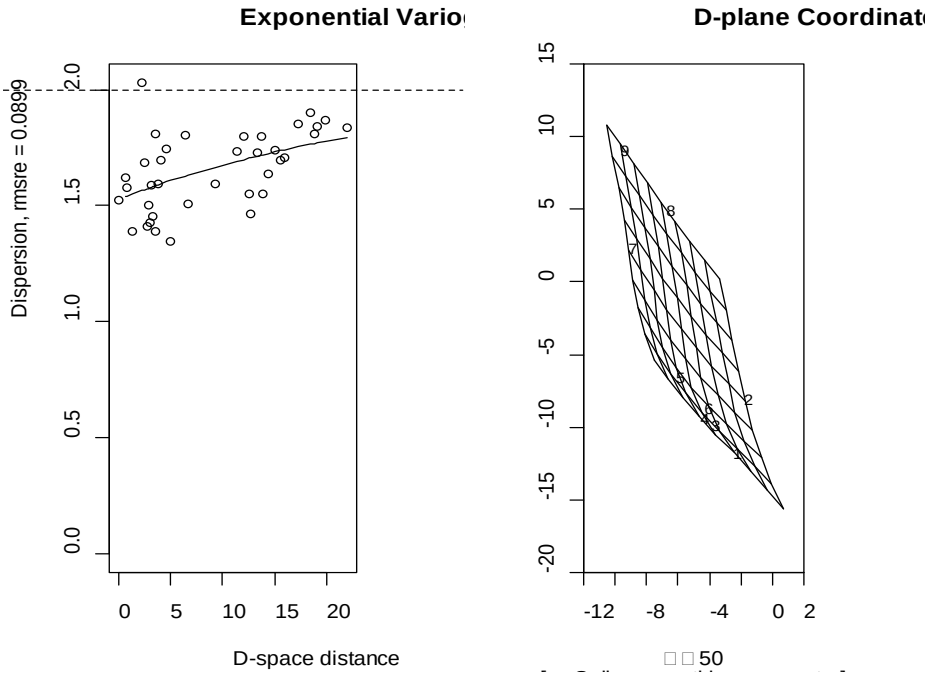
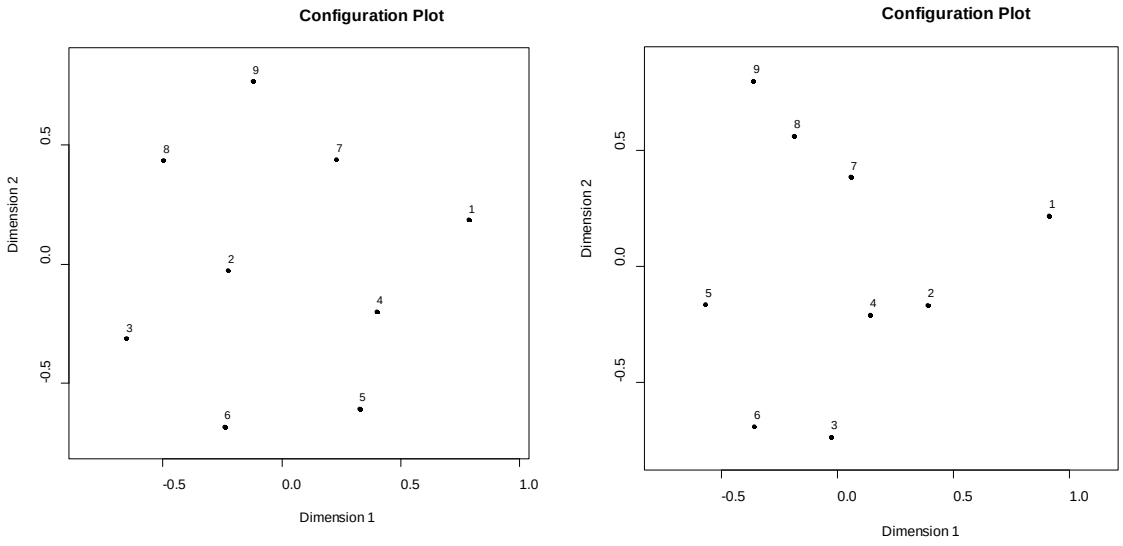


Figura 9. Cuadrícula del plano-D y variograma ajustado para $\lambda = 50$.

5.1.2. Implementación mediante la versión modificada de EnviroStat

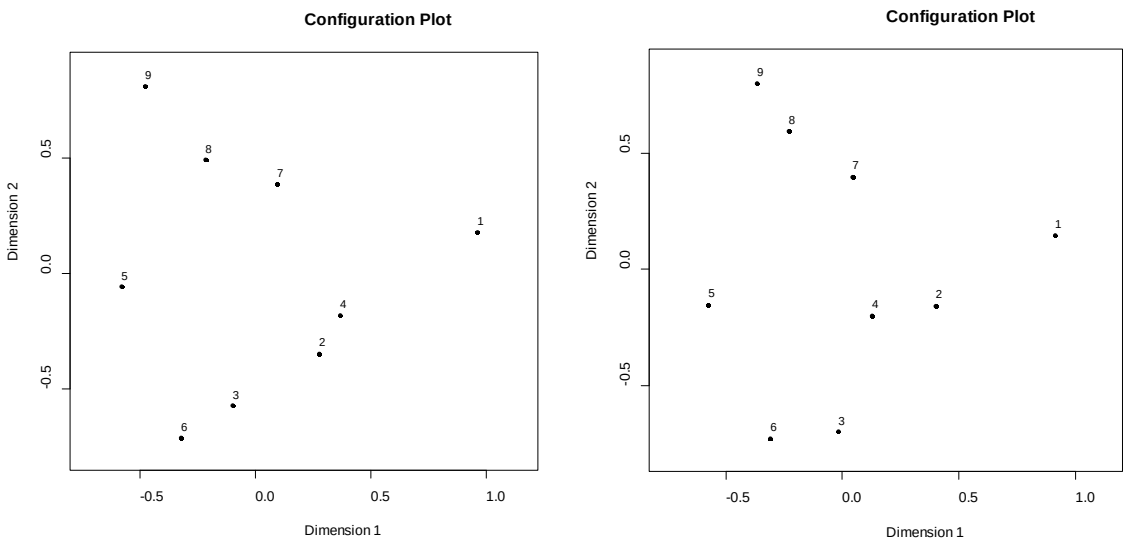
Antes de mostrar la implementación de las primeras dos etapas del método de Sampson y Guttorp con la versión modificada del paquete *EnviroStat*, es necesario probar el paquete smacof con el fin de determinar cuál variante de MDS es la más adecuada para obtener el plano-D en la etapa 1.

En primer lugar fueron ensayadas las variantes tipo razón e intervalo del MDS, las cuales producen las configuraciones de la figura 10, estas fueron rotadas con el fin de facilitar la comparación con el plano-D obtenido en el primer paso de la implementación con la versión original del paquete. Luego de comparar las gráficas de la figura 10 con la figura 6 se aprecia que la variante tipo intervalo del MDS genera una configuración más similar a la mostrada en la figura 6.



(A) (B)

Figura 10. Espacio de dispersión o plano-D obtenido mediante las variantes tipo razón (A) y tipo intervalo (B) del paquete *smacof*.



(A) (B)

Figura 11. Espacio de dispersión o plano-D obtenido mediante las variantes tipo ordinal (A) y tipo mspline (B) del paquete *smacof*.

```
> library(smacof)

> coordenadas <-smacofSym(disp, ndim=2, type="ratio")
> x <- coordenadas$conf[,1]
> y <- coordenadas$conf[,2]
> coordenadas$conf[,1]<- y
> coordenadas$conf[,2]<- x
> plot(coordenadas)

> coordenadas <-smacofSym(disp, ndim=2, type="interval")
> x <- coordenadas$conf[,1]
> y <- coordenadas$conf[,2]
> coordenadas$conf[,1]<- y
> coordenadas$conf[,2]<- x
> plot(coordenadas)
```

Posteriormente se ensayaron las variantes tipo ordinal y mspline de MDS, las cuales producen las configuraciones de la figura 11, estas fueron rotadas al igual que las dos anteriores con el fin de facilitar la comparación con el plano-D obtenido en el primer paso de la implementación con la versión original del paquete. Luego de comparar las gráficas de la figura 11 y de la figura 10

con la figura 6 se aprecia que las variantes intervalo y mspline del MDS generan las configuraciones que más se asemejan a la mostrada en la figura 6.

```
> coordenadas <-smacofSym(disp, ndim=2, type="ordinal")
> x <- coordenadas$conf[,1]
> y <- coordenadas$conf[,2]
> coordenadas$conf[,1]<- y
> coordenadas$conf[,2]<- x
> plot(coordenadas)

> coordenadas <-smacofSym(disp, ndim=2, type="mspline",
+                         spline.degree = 4)
> x <- coordenadas$conf[,1]
> y <- coordenadas$conf[,2]
> coordenadas$conf[,1]<- y
> coordenadas$conf[,2]<- x
> plot(coordenadas)
```

En la tabla 2 se muestra la magnitud del stress-1 obtenida para cada uno de los modelos desarrollados. Se aprecia que la variante ordinal es la que permite obtener un stress más bajo, lo que indica un mejor ajuste; sin embargo, la configuración obtenida presenta algunas diferencias notables con la obtenida en la etapa 1 de implementación mediante la versión original del paquete *EnviroStat*, una de ellas es la ubicación de las estaciones 2 y 4 que cambian de posición respecto a las observadas en la figura 6. Por este motivo se escogió el MDS tipo mspline en la implementación del método de Sampson y Guttorp con la versión modificada, ya que además de generar una configuración más semejante a la mostrada en la figura 6 proporciona un mejor ajuste que otras variantes.

Variante	*Stress-1*
razón	0.2632602
intervalo	0.1691994
ordinal	0.1094753
mspline	0.1563073

Tabla 2. Magnitud del stress-1 para cada uno de los modelos desarrollados mediante la función smacofSym

Para implementar el método de Sampson y Guttorp mediante la versión modificada del paquete *EnviroStat* es necesario normalizar tanto la matriz de coordenadas geográficas transformadas mediante la proyección de Lambert y la matriz de dispersión, aplicando la ecuación 22.

```
> coords <- Flamb2(abs(nloc))
> coords.lamb <- coords$xy
> dist  <- Fdist(coords.lamb)
> kdist <- sqrt((nrow(dist)*(nrow(dist)-1)/2)/
+                 sum(dist[lower.tri(dist)]^2))
> coords.lamb <- kdist*coords.lamb

> disp  <- 2 - 2 * corr.est
> dispn <- sqrt((nrow(disp)*(nrow(disp)-1)/2)/
+                 sum(disp[lower.tri(disp)]^2))*disp
> disp  <- dispn
```

Una vez normalizadas las matrices de coordenadas transformadas y de dispersión se procede a llevar a cabo la primera etapa del procedimiento de Sampson y Guttorp, para ello es necesario cargar el programa `SG_nuevo.R`, el cual contiene las funciones necesarias para implementar el método con las modificaciones incluidas.

```
> source('SG_nuevo.R')
> sg.est_nuevo <- Falternate3_nuevo(disp, coords.lamb, max.iter = 100,
+                                   alter.lim = 100, model = 1)
```

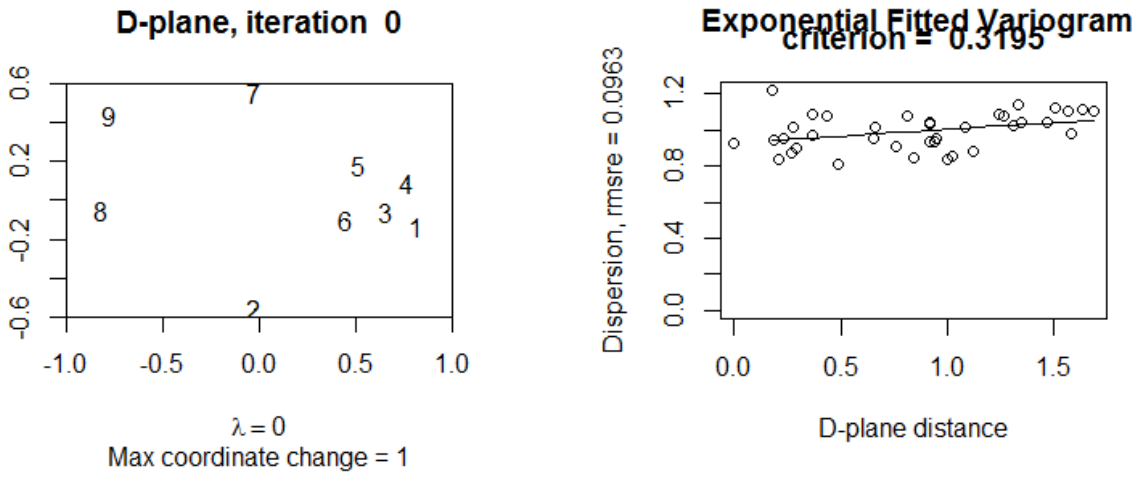


Figura 12. Plano-D y variograma exponencial ajustado iniciales para la versión modificada

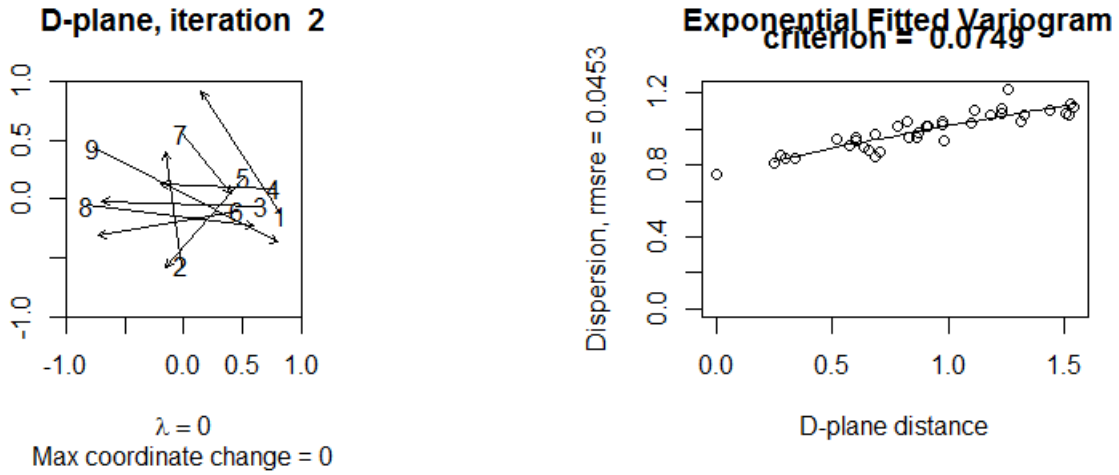


Figura 13. Plano-D y variograma exponencial ajustado en la última iteración para la versión modificada

La figura 12 muestra el plano-D y el variograma inicial obtenidos con la versión modificada del paquete *EnviroStat*, por otro lado, en la figura 13 se aprecia los resultados de la última iteración, las flechas representan el movimiento de las estaciones a partir de la ubicación original. Se observa que luego de 2 iteraciones el error cuadrático medio en el ajuste del variograma disminuye de 0.0963 a 0.0453, estos valores son similares a los obtenidos con la versión original.

Al comparar ambos procedimientos llama la atención la diferencia en el número de iteraciones necesarias para obtener el plano-D: 82 en la versión original y sólo 2 en la versión modificada. Esto se debe a que en la versión modificada se obtiene desde la primera iteración un plano-D mucho más cercano a la configuración óptima que en la versión original, el cual permite conseguir un grado satisfactorio de ajuste del variograma gracias a la incorporación del paquete *smacof*.

No. de Réplicas	10		100	
Versión del paquete	Original	Modificada	Original	Modificada
Tiempo (min)	1.191938	0.078979	11.193630	0.828163

Tabla 3. Comparación de los tiempos de ejecución de la primera etapa del método de Sampson y Guttorp

En la tabla 3 se observa el tiempo que tarda la implementación de un determinado número de réplicas de la primera etapa del método de Sampson y Guttorp con las versiones del paquete *EnviroStat*. Las diferencias en los tiempos de ejecución saltan a la vista, en efecto, la implementación de 100 réplicas de la etapa 1 con la versión modificada tarda aproximadamente 14 veces menos que con la versión original para el caso de una red conformada por 9 estaciones de monitoreo. Esta reducción considerable en el tiempo de ejecución constituye una ventaja en el momento de aplicar la versión modificada del paquete a grandes volúmenes de datos.

```
> inicio <- Sys.time()
> replicate(10, Falternate3(disp, coords.lamb, max.iter = 100,
+                           alter.lim = 100, model = 1))
          [,1]       [,2]       [,3]       [,4]       [,5]
variogfit List,4     List,4     List,4     List,4     List,4
ncoords   Numeric,18 Numeric,18 Numeric,18 Numeric,18 Numeric,18
          [,6]          [,7]       [,8]       [,9]    [,10]
variogfit List,4     List,4     List,4     List,4     List,4
ncoords   Numeric,18 Numeric,18 Numeric,18 Numeric,18 Numeric,18
> print(Sys.time()-inicio)
Time difference of 1.191938 mins
```

La implementación del método de Sampson y Guttorp en la etapa 2 difiere en la versión modificada de la versión original en el procedimiento para seleccionar el parámetro de suavizado λ. En efecto, tal como se explicó en la sección anterior se incorpora en la versión modificada un procedimiento de *validación cruzada dejando uno fuera* en el cual se obtiene una gráfica del error de validación cruzada frente al parámetro de suavizado para facilitar la selección de éste último. Para obtener esta gráfica se emplea la función `LOOCVcal` que permite obtener el error de validación cruzada dado un valor del parámetro λ.

```
> dev.new(width=5, height=4)
> lambdas <- seq(0,1,0.02)
> lenlam <- length(lambdas)
> vecv = NULL
> for (i in 1:lenlam){
+   vecv <- c(vecv,LOOCVcal(disp=disp,Gcrds=coords.lamb,
+                           MDScrds=sg.est_nuevo$ncoords,
+                           lam=lambdas[i]))
+ }
> plot(lambdas,vecv,xlab='lambda',ylab='MSE CrossVal')
> lines(lambdas,vecv)
```

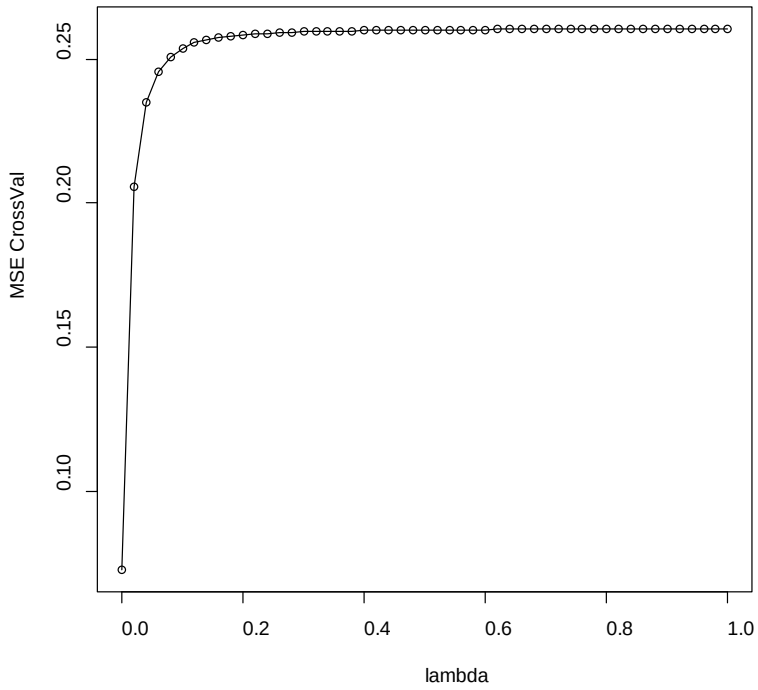


Figura 14. Error de validación cruzada frente a λ en la versión modificada del paquete

En la figura 14 se observa el diagrama que representa el error de validación cruzada frente a λ para cada modelo. Se aprecia que cuando $\lambda = 0$ se obtiene el menor error de validación cruzada, no obstante, tal como se explicó en la sección anterior este valor no puede seleccionarse, ya que genera una máxima deformación del espacio-G en el espacio-D que dificulta la interpretación del modelo. La curva presenta un comportamiento asintótico y confirma que al efectuar la normalización de las matrices de dispersión y de coordenadas transformadas de las ubicaciones en el espacio-G el valor del parámetro de suavizado necesario para obtener el balance adecuado entre el ajuste del variograma y la deformación espacial está entre 0 y 1. Teniendo en cuenta este diagrama, se decidió utilizar $\lambda = 0.18$ como valor del parámetro de suavizado con el fin de garantizar que la deformación espacial sea mínima y por su ubicación próxima a la asíntota de la curva de error de validación cruzada.

Si se utiliza la versión original de *EnviroStat* con las matrices de dispersión y de coordenadas transformadas de las ubicaciones en el plano-G sin normalizar se obtiene el diagrama de error de validación cruzada mostrado en la figura 15, el cual permite apreciar nuevamente un comportamiento asintótico de la curva pero a partir de un valor considerablemente más alto que el observado en la figura 14. Por consiguiente, en la versión original del paquete la selección del parámetro de suavizado implica ensayar una gama más amplia de valores de λ.

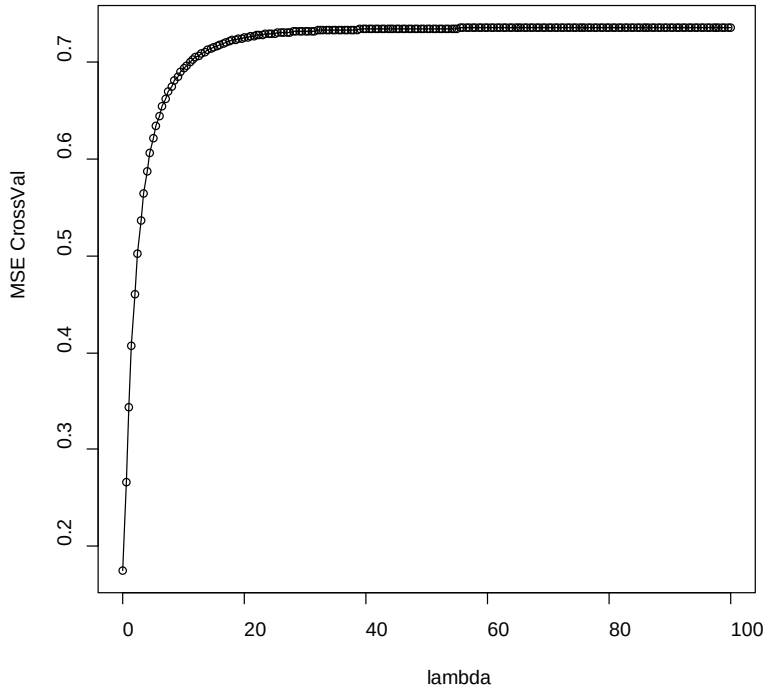


Figura 15. Error de validación cruzada frente a λ en la versión original del paquete

Después de seleccionar el valor de λ se utiliza la función `Ftransdraw_nuevo` para confirmar que el valor escogido del parámetro de suavizado permite obtener un grado de deformación satisfactoria del plano-G en el plano-D y un ajuste adecuado del modelo.

```
> apply(coords.lamb, 2, range)
              x          y
[1,] -0.8156602 -0.5590775
[2,]  0.8168966  0.5590879
> coords.grid <- Fmgrid(range(coords.lamb[,1]),
+                       range(coords.lamb[,2]))
> par(mfrow = c(1, 2))
> temp <- setplot(coords.lamb, axis = TRUE)
```

```
> deform <- Ftransdraw_nuevo(disp = disp, Gcrds = coords.lamb,
+                            MDScrds = sg.est_nuevo$ncoords,
+                            gridstr = coords.grid, lambda = 0.18)
```

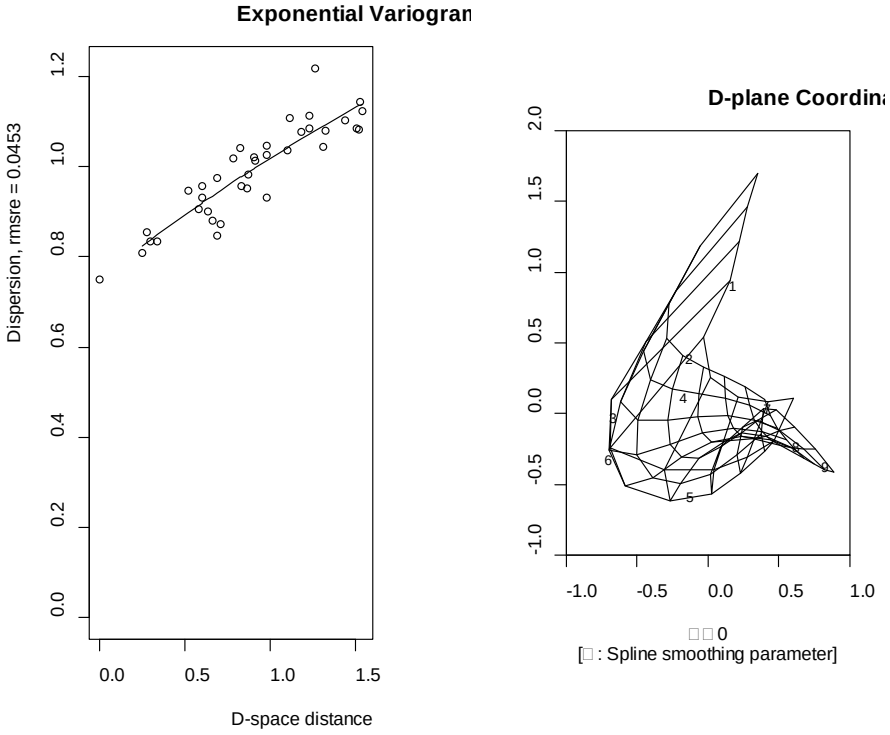


Figura 16. Cuadrícula del plano-D y variograma ajustado para $\lambda = 0$.

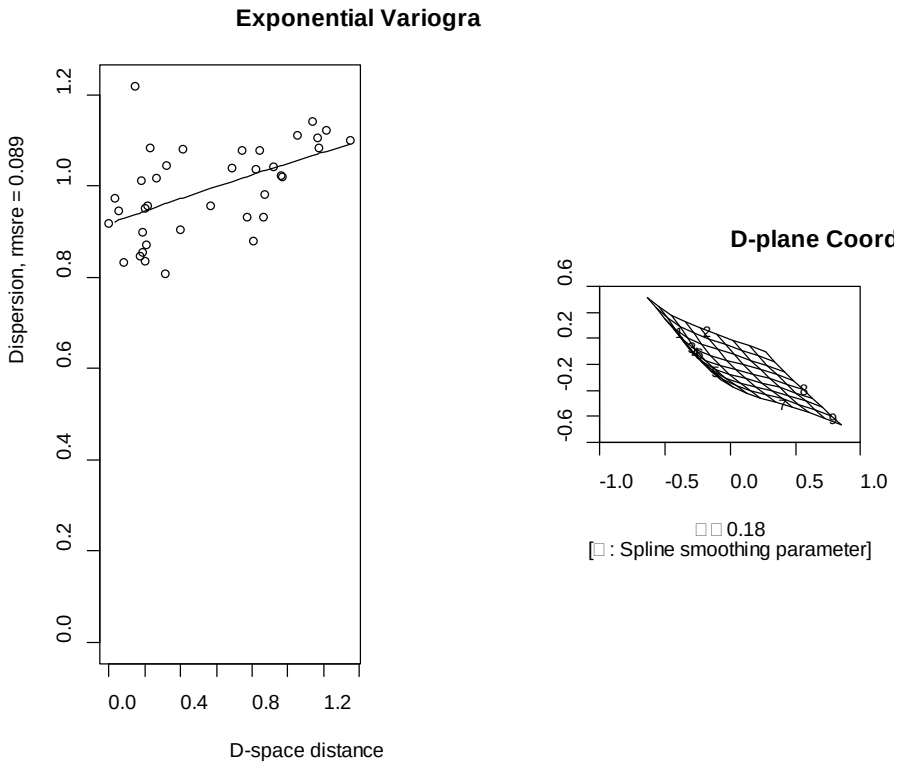


Figura 17. Cuadrícula del plano-D y variograma ajustado para $\lambda = 0.18$

En la figura 16 se observa la deformación de la cuadrícula del plano-G al plano-D cuando $\lambda = 0$. A diferencia de la implementación mediante la versión original donde el parámetro de suavizado necesario es igual a 50, en la versión modificada solamente se requiere $\lambda = 0.18$ para obtener una deformación suave del plano-D (figura 17) y un ajuste del variograma que inclusive es mejor al conseguido mediante la versión original, puesto que el error cuadrático medio es ligeramente inferior.

5.2. Ejemplo de implementación completa del método de Sampson y Guttorp

Para mostrar la implementación completa del método de Sampson y Guttorp mediante las modificaciones introducidas al paquete *EnviroStat* se utilizó la hoja de datos `datum_ST` creada a partir de información extraída de la base de datos sobre calidad del aire de la Agencia Ambiental Europea[2]. Los datos corresponden a mediciones de la concentración de dióxido de azufre (SO_2) en ppb (partes por billón) realizadas cada hora en 9 estaciones de monitoreo ubicadas en Estonia. Cada fila de la hoja de datos representa una medición efectuada cada hora a partir del 1 de agosto de 2009 hasta el 30 de noviembre de 2009 abarcando un período de 122 días, es así que la hoja de datos contiene $24 \times 122 = 2928$ filas o registros para cada estación. El análisis comienza con la carga del paquete *EnviroStat* y la observación de la estructura del conjunto de datos.

```
> library(EnviroStat)
> datos_ST <- read.table("datum_ST.txt",sep="\t")

> str(datos_ST)
'data.frame':  2928 obs. of  15 variables:
 $ date: Factor w/ 122 levels "2009-08-01","2009-08-02",..: 1 1 1 1 1
1 1 1 1 1 ...
 $ mm  : int  8 8 8 8 8 8 8 8 8 8 ...
 $ dd  : int  1 1 1 1 1 1 1 1 1 1 ...
 $ dw  : int  6 6 6 6 6 6 6 6 6 6 ...
 $ wy  : int  30 30 30 30 30 30 30 30 30 30 ...
 $ hr  : int  1 2 3 4 5 6 7 8 9 10 ...
 $ S1  : num  1.2 0.9 0.9 4.1 8 14.7 14.2 10.7 12.1 60.3 ...
 $ S2  : num  0.3 0.3 0.3 0.3 0.3 0.2 0.4 0.5 0.5 0.5 ...
 $ S3  : num  0 0.1 0.1 0.1 0.1 0.1 0 0.1 0.1 0.1 ...
 $ S4  : num  4.6 5.8 4.3 2.1 2.3 1.2 0.9 1 0.8 1.2 ...
 $ S5  : num  0.1 0.1 0.1 0.1 0.1 0 0.1 0.1 0.1 0.1 ...
 $ S6  : num  0.1 0 0 0 0 0 0 0.1 0.2 0.2 ...
 $ S7  : num  0.4 0.4 0.3 0.3 0.3 0.4 0.5 0.5 0.1 0 ...
 $ S8  : num  0.1 0.4 0.2 0.2 0.2 0 0.1 0.2 0.1 0.1 ...
 $ S9  : num  0.2 0.2 0.1 0.1 0.2 0.2 0.1 0.2 0.1 0.1 ...
```

La primera variable es la fecha de la medición, luego aparece el mes cuyo valor es un entero comprendido en el intervalo [8, 11]; la siguiente variable es el día, siendo su valor un entero que pertenece al intervalo [1, 31]; luego aparece el día de la semana, un entero igual a 0 si el día de la medición es domingo y así sucesivamente hasta 6 para el día sábado. La siguiente

[2] http://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-6

variable es la semana del año, un entero comprendido entre 30 y 48; la sexta variable es la hora en la que se efectuó la medición, cuyo valor es un entero comprendido en el intervalo [1, 24]; las demás variables: S1, S2, …, S9, corresponden a las concentraciones de SO_2 medidas en cada estación de monitoreo. Con el fin de obtener distribuciones más simétricas de estas últimas variables fue necesario aplicar la siguiente transformación: $\ln(x+1)$, donde x representa el valor de la observación original.

```
> for(i in 7:15){
+   datos_ST[,i] <- log(datos_ST[,i]+1)
+ }
```

Las coordenadas de las estaciones de monitoreo se encuentran en la hoja de datos: `AirBase_EE_v6_stations.csv`, la cual muestra que existen 11 estaciones en Estonia que miden la concentración de SO_2, no obstante, dado que las estaciones de las filas 4 y 10 no registran mediciones entre agosto y noviembre de 2009 se eliminaron estas filas.

```
> estaciones<-read.csv('AirBase_EE_v6_stations.csv',header=T,sep="\t")
> nrow(estaciones)
[1] 11
> estaciones_wk <- estaciones[-c(4,10),]
```

A partir de esta última hoja de datos se generó la matriz con las coordenadas geográficas de las estaciones, estas aparecen representadas en el mapa de la figura 18, el cual fue generado mediante la función `map` del paquete maps.

```
> location2 <- matrix(0,nrow=9,ncol=2)
> for (i in 1:2){
+   location2[,i] <- estaciones_wk[,15-i]
+ }
> colnames(location2) <- c('Lat','Long')
> location2
          Lat     Long
 [1,] 59.40973 27.27862
 [2,] 59.49445 25.93057
 [3,] 59.43111 24.76056
 [4,] 59.37616 28.17917
 [5,] 59.45694 24.69862
 [6,] 58.70278 26.75890
 [7,] 58.29361 26.57390
 [8,] 58.37611 21.84501
 [9,] 59.41417 24.64946
```

```
> dev.new(width=5, height=4)
> map(regions='estonia',xlim = c(20, 29), ylim = c(57,60))
> points(location2[,2],location2[,1],pch=19,col="red",cex=0.6)
> text(x=location2[,2]+0.01,y=location2[,1]+0.01,
+      labels=colnames(datos_ST)[7:15],cex=1)
```

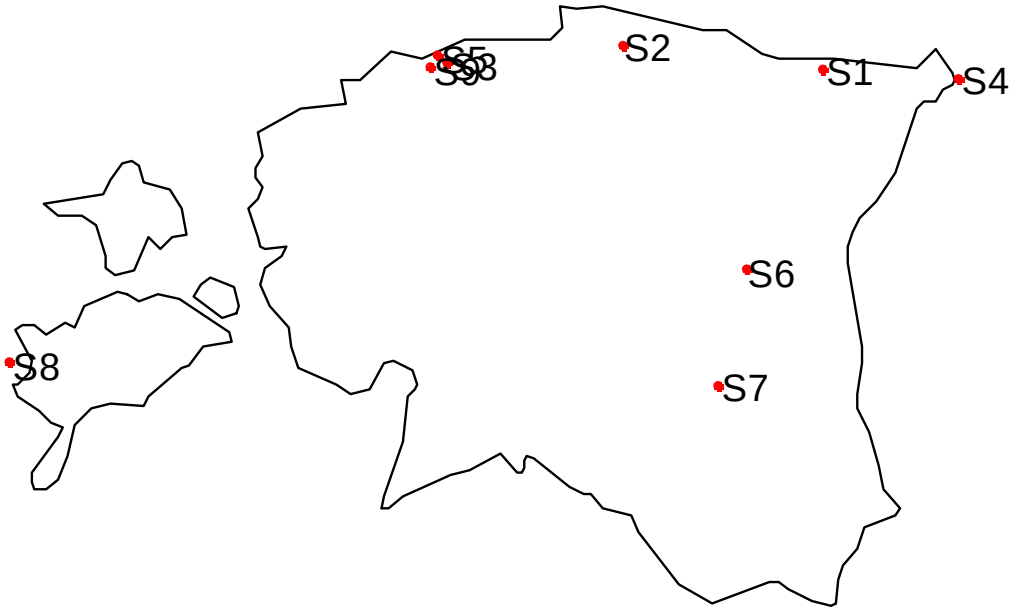


Figura 18. Ubicación geográfica de las estaciones de monitoreo

El análisis exploratorio de los datos inicia en este caso examinando la tendencia determinista de los niveles de SO2 ajustando un modelo lineal con las variables hora, día de la semana y semana como factores para cada estación por separado. La función `model.matrix` se utiliza para obtener la correspondiente matriz de diseño y el ajuste lineal se consigue mediante la función `lm`.

```
> hr <- as.factor(datos_ST[,6])
> wkday <- as.factor(datos_ST[,4])
> week <- as.factor(datos_ST[,5])
> y <- datos_ST[,7:15]
> x <- model.matrix(~hr + wkday + week)

> fit <- list()
> for (i in 1:9)
+   fit[[i]] = lm(y[,i] ~ x -1, singular.ok =T , na.action=na.omit)
```

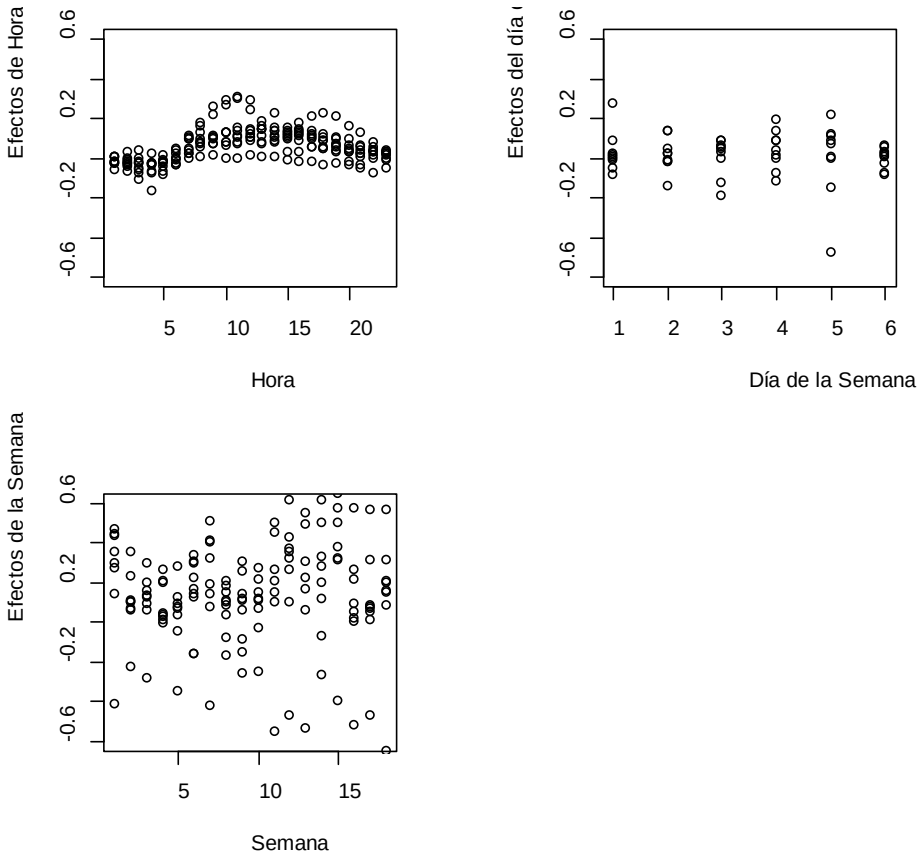


Figura 19. Gráficas de efectos estimados

Los efectos estimados se representan en la figura 19, estos resultados muestran patrones consistentes debido a los efectos de la hora y del día de la semana para todas las estaciones, mientras que los efectos de la semana no muestran una tendencia temporal.

```
> par(mfrow=c(2,2))
> par(mar=c(4,4,4,4))
> plot(fit[[1]]$coef[2:24],ylim=c(-.6,.6),type="n",xlab="Hora",
+        ylab="Efectos de Hora")
> for (i in 1:9) points(fit[[i]]$coef[2:24])

> plot(fit[[1]]$coef[25:30],ylim=c(-.6,.6),type="n",xlab="Día de la
+      Semana", ylab="Efectos del día de la Semana")
> for (i in 1:9) points(fit[[i]]$coef[25:30])


> plot(fit[[1]]$coef[31:48],ylim=c(-.6,.6),type="n",xlab="Semana",
+      ylab="Efectos de la Semana")
> for (i in 1:9) points(fit[[i]]$coef[31:48])
```

Las gráficas Q-Q para los residuos ajustados se muestran en la figura 20, estas indican que si bien existen algunas ligeras desviaciones en algunas estaciones, la suposición de normalidad es razonable.

```
> par(mfrow=c(3,3))
> for (i in 1:9) qqnorm(fit[[i]]$resid)
```

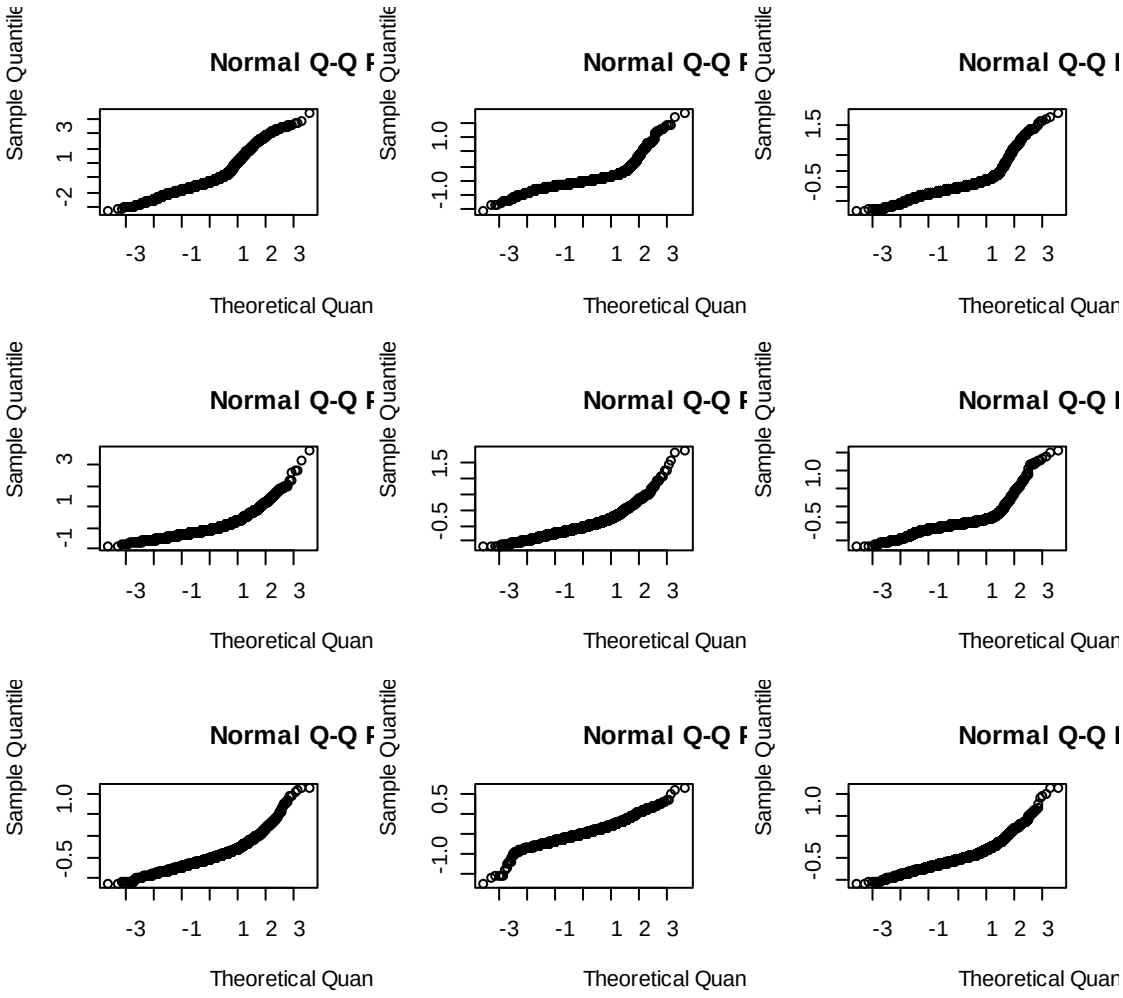


Figura 20. Gráficas Q-Q para los residuos ajustados

El análisis mostrado a continuación tiene como propósito determinar si existe una estructura de autorregresión de segundo orden AR(2) para cada una de las estaciones. Para ello es necesario obtener las correlaciones espaciales entre las estaciones antes y después de eliminar la estructura AR(2).

```
> missing.num <- apply(is.na(datos_ST[,7:15]),2,sum)
> missing.num
S1 S2 S3 S4 S5 S6 S7 S8 S9
57 20  0 50 50 26  1 63  0

> # Obtención de las correlaciones sin eliminar la estructura AR(2)
> lmfit.resid <- NULL
> for (i in 1:9){
+   lmfit.resid <- cbind(lmfit.resid,c(rep("NA",missing.num[i]),
+                                      fit[[i]]$resid))
+ }
> lmfit.resid <-apply(lmfit.resid,2,as.numeric)
> lmfit.resid.corr <- cor(lmfit.resid, use="pairwise.complete.obs")


> # Obtención de las correlaciones luego de eliminar la estructura AR(
2)
> arfit <- list()
> for (i in 1:9){
+   arfit[[i]] <- ar(fit[[i]]$resid,aic=F,order=2)
+ }
> ar.resid = NULL
> for (i in 1:9){
+   ar.resid <- cbind(ar.resid,c(rep("NA",missing.num[i]),
+                                            arfit[[i]]$resid))
+ }
> ar.resid <-apply(ar.resid,2,as.numeric)
> ar.resid.corr <- cor(ar.resid,use = "pairwise")
```

El efecto producido por la eliminación de la estructura AR(2) se aprecia mejor al graficar las correlaciones espaciales estimadas antes y después de eliminar la estructura AR(2) frente a las distancias entre las estaciones. Para calcular las distancias es necesario al igual que en el ejemplo de aplicación anterior, transformar las coordenadas de las estaciones mediante la proyección de Lambert usando la función `Flamb2` y luego utilizar la función `Fdist` para estimar las distancias.

```
> coords2 <- Flamb2(abs(location2))
> dist2 <- Fdist(coords2$xy)

> par(mfrow=c(1,2))
> plot(-.2,0,xlim=c(0,300),ylim=c(-.2,1),xlab="Dist",
+        ylab="Correlación espacial (pre-AR(2) logSO2)",type="n")
> for (i in 1:8) for (j in (i+1):9)
+   points(dist2[i,j],lmfit.resid.corr[i,j])

> plot(-.2,0,xlim=c(0,300),ylim=c(-.2,1),xlab="Dist",
+        ylab="Correlación espacial (AR(2) resid)",type="n")
> for (i in 1:8) for (j in (i+1):9)
+   points(dist2[i,j],ar.resid.corr[i,j])
```

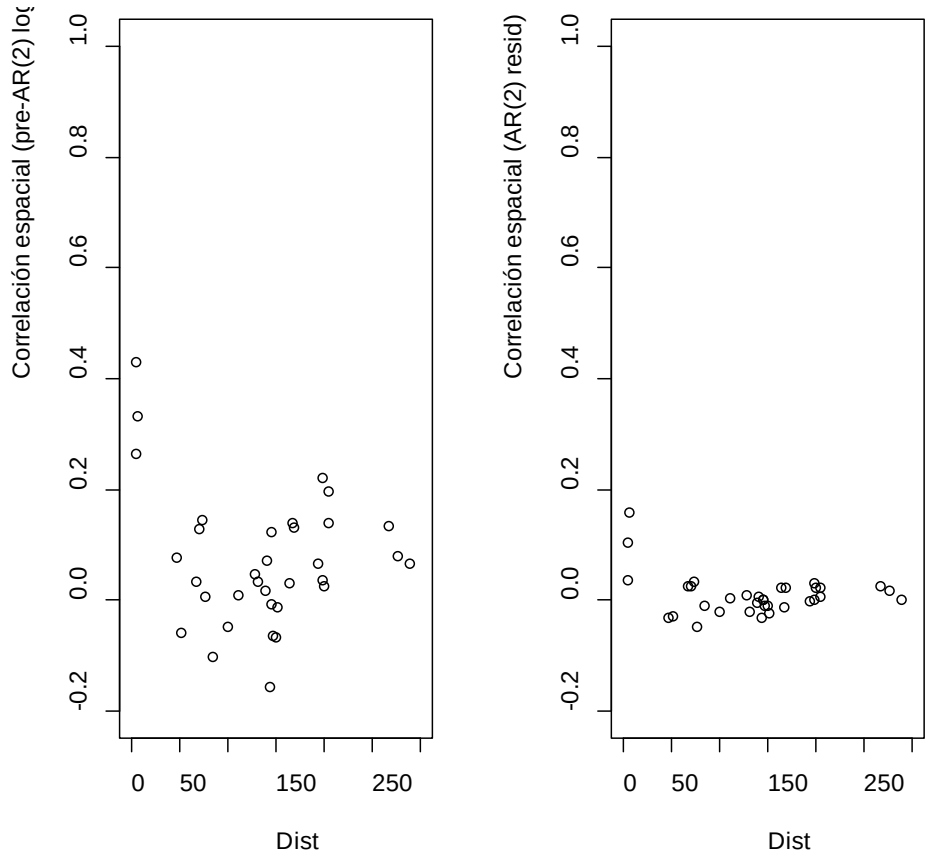


Figura 21. Correlaciones espaciales estimadas antes y después de eliminar la estructura AR(2)

Los resultados muestran que la mayoría de correlaciones espaciales antes de remover la estructura de autorregresión AR(2) se encuentran en el rango entre – 0.2 y 0.2 pero estas se reducen a valores próximos a cero cuando se elimina dicha estructura. Dado que no hay un cambio excesivo en las correlaciones espaciales al remover la estructura de autorregresión se puede afirmar que en este caso la autocorrelación no es fuerte. El enfoque adoptado, el cual será ilustrado a continuación, modela simultáneamente los componentes temporales y espaciales juntando las respuestas a cada hora en un solo vector de respuestas para cada día usando el algoritmo EM. Este enfoque puede utilizarse inclusive en casos donde la autocorrelación es marcada, ya que la estructura de autorregresión se deja arbitraria evitando la necesidad de modelar tal dependencia a la escala temporal fina de una hora (Le & Zidek, 2013).

Para aplicar este enfoque y en particular la función `staircase.EM` necesaria para la obtención de la matriz de covarianza y por consiguiente de la matriz de dispersión, se requiere que la matriz de datos usada como argumento de esta función tenga una estructura de bloques conformados por estaciones que tienen el mismo número de observaciones faltantes, de manera que se obtenga un patrón monótono de datos faltantes que luzca como una escalera ascendente o descendente, siendo preciso para ello efectuar una permutación en el orden de los sitios en la hoja de datos. Luego de probar varias permutaciones y de eliminar algunas filas de la matriz de datos no fue posible obtener el patrón requerido para conseguir una ejecución de la función `staircase.EM` sin errores. Por tal motivo, se llevó a cabo un proceso de imputación mediante el paquete *mice* que consiste en asignar valores a las observaciones faltantes para evitar inconvenientes en la estimación de la matriz de covarianza entre las estaciones al usar el algoritmo EM.

```
> library(mice)
> datos_STno <- mice(datos_ST,m=5,maxit=50,meth='pmm',seed=500)
> datos_ST_comp2 <- complete(datos_STno,2)
```

En el análisis mostrado a continuación se supone que se desea interpolar los niveles de SO_2 a cada hora en ubicaciones que no han sido monitoreadas para una hora específica del día, siendo esta las 3:00 pm, designando así el período comprendido entre 3:00 y 4:00 pm. El método aplicado permite emplear vectores de respuesta multivariante que constan de los niveles de SO_2 a las 3:00 pm al igual que varias de las horas precedentes. De esta forma el enfoque de interpolación espacial aprovecha la fuerza no solamente de las concentraciones medidas cada hora en los sitios de monitoreo a las 3:00 pm, sino también de las mediciones efectuadas en aquellos sitios en las horas precedentes. En el presente ejemplo de aplicación se utilizan cuatro horas consecutivas en el vector de respuesta multivariante para cada día, donde la última hora es la que inicia a las 3:00 pm. Este enfoque asegura que los vectores de respuesta sean estocásticamente independientes a una aproximación razonable y que pueda aplicarse aún en casos donde exista una estructura AR(2) fuerte ya que el lapso de 18 horas entre cada par de

vectores de respuesta reduce considerablemente la correlación temporal. Otra característica del enfoque es la suposición de la separabilidad de la covarianza, esto equivale a afirmar que la covarianza para aquellas cuatro horas es la misma de un día al siguiente (Le & Zidek, 2013).

Con el fin de aplicar el enfoque descrito anteriormente es necesario organizar los datos de cada estación en una matriz de dimensión 122×24, ya que las mediciones registradas cada hora comprenden un período de 122 días, estas matrices se yuxtaponen creando así una matriz más grande denominada series24hr cuya dimensión es 122×216.

```
> series24hr <- NULL
> for (i in 7:15) {
+   x <- datos_ST_comp2[,i]
+   temp <- t(matrix(x,nrow=24))
+   series24hr <- cbind(series24hr,temp)
+ }
```

Las respuestas multivariantes que constan de cuatro horas consecutivas desde las 12:00 m hasta las 3:00 pm se extraen de la matriz creada en el paso anterior, obteniendo así una matriz denotada por hr12.15.

```
> n <- 4
> tt <- c(1:n)
> for (i in 2:9){
+   tt <- c(tt,c(1:n)+24*(i-1))
+ }
> hr12.15 <- series24hr[,tt+3*n]
```

La estimación de la matriz de covarianza espacial entre las estaciones y por consiguiente de la matriz de dispersión necesaria para implementar el procedimiento de Sampson y Guttorp, se lleva a cabo mediante el algoritmo EM usando la función `staircase.EM` en la que las variables *month* y *weekday* se utilizan como factores categóricos. El resultado de esta función es una lista en la que el elemento *Psi* corresponde a la matriz de covarianza incondicional entre las estaciones de monitoreo, mientras que el elemento *Omega* representa la matriz de covarianza entre horas.

```
> month <- as.factor((matrix(datos_ST_comp2[,2],byrow=T,ncol=24))[,1])
```

```
> weekday<-as.factor((matrix(datos_ST_comp2[,4],byrow=T,ncol=24))[,1])
```

```
> ZZ <- model.matrix(~month + weekday)
> em.fit <- staircase.EM(hr12.15, p = 4, covariate = ZZ,
+                        maxit = 400, tol = .000001)

> cov.est <- em.fit$Psi[[1]]
> dim1 <- nrow(cov.est)
> dim2 <- nrow(em.fit$Omega)
> corr.est <- cov.est /sqrt( matrix(diag(cov.est), dim1, dim1) *
+                              t(matrix(diag(cov.est), dim1, dim1)))

> disp <- 2 - 2 * corr.est
```

Tal como se mostró en el ejemplo de aplicación anterior la implementación del método de Sampson y Guttorp mediante la versión modificada del paquete *EnviroStat* requiere de la normalización de la matriz de dispersión y de la matriz de coordenadas transformadas mediante la proyección de Lambert.

```
> dispn <- sqrt((nrow(disp)*(nrow(disp)-1)/2)/
+                 sum(disp[lower.tri(disp)]^2))*disp
> disp <- dispn

> kdist <- sqrt((nrow(dist2)*(nrow(dist2)-1)/2)/
+                 sum(dist2[lower.tri(dist2)]^2))
> coords.lamb <- kdist*coords2$xy
```

La primera etapa del método de Sampson y Guttorp, ya descrita en el ejemplo de aplicación anterior, procede con las modificaciones introducidas al paquete *EnviroStat*.

```
> source('SG_nuevo.R')
> sg.est_nuevo <- Falternate3_nuevo(disp, coords.lamb, max.iter = 100,
+                                   dims=2, alter.lim = 100, model= 1)
```

En las figuras 22 y 23 se aprecian el plano-D y el variograma exponencial ajustado iniciales y en la iteración final, respectivamente. Al igual que en el ejemplo de aplicación anterior, solamente fueron necesarias 2 iteraciones con la versión modificada para conseguir una configuración óptima del plano-D y un grado de ajuste satisfactorio del variograma, confirmado por la magnitud baja del error cuadrático medio.

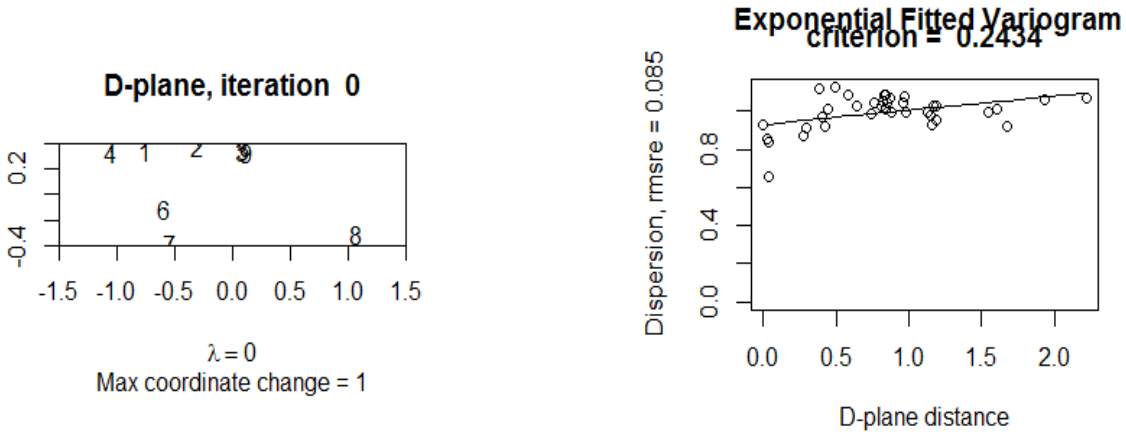


Figura 22. Plano-D y variograma exponencial ajustado iniciales

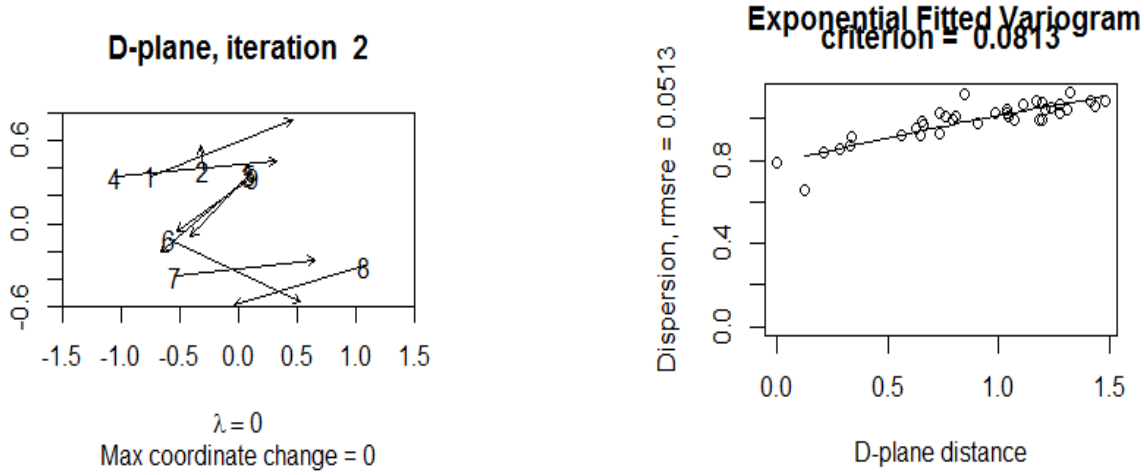


Figura 23. Plano-D y variograma exponencial ajustado en la última iteración

La etapa 2 del método de Sampson y Guttorp, tal como se mostró en el primer ejemplo de aplicación, busca el ajusta del spline de placa delgada entre el plano-G y el plano-D obtenido en la fase 1. En la versión modificada de *EnviroStat* se implementa un procedimiento de validación cruzada para seleccionar el parámetro de suavizado λ, generando en primer lugar la curva de error de validación cruzada frente a λ.

```
> dev.new(width=5, height=4)
```

```
> lambdas <- seq(0,1,0.02)
> lenlam <- length(lambdas)
> vecv = NULL
```

```
> for (i in 1:lenlam){
+   vecv <- c(vecv,LOOCVcal(disp=disp,Gcrds=coords.lamb,
+                           MDScrds=sg.est_nuevo$ncoords,
+                           lam=lambdas[i]))
+ }
> plot(lambdas,vecv,xlab='lambda',ylab='MSE CrossVal')
> lines(lambdas,vecv)
```

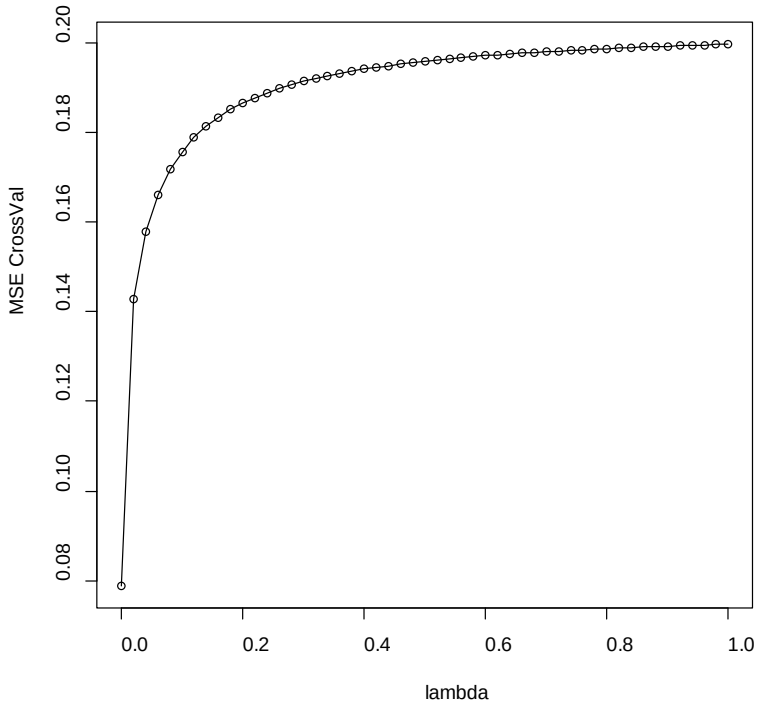


Figura 24. Error de validación cruzada frente a λ

En la figura 24 se observa el diagrama que representa el error de validación cruzada frente a λ para cada modelo. Adoptando un criterio similar al del primer ejemplo de aplicación se decidió utilizar $\lambda = 0.50$ como valor del parámetro de suavizado con el fin de garantizar que la deformación espacial sea mínima y porque aproximadamente a partir de este valor se empieza a observar el comportamiento asintótico de la curva de error de validación cruzada.

Luego de seleccionar el valor de λ se utiliza la función `Ftransdraw_nuevo` para confirmar que el valor escogido del parámetro de suavizado permite obtener un grado de deformación satisfactoria del plano-G en el plano-D y un ajuste adecuado del modelo.

```
> apply(coords.lamb, 2, range)
             x          y
[1,] -1.044385 -0.3826198
[2,]  1.075024  0.3908950

> coords.grid <- Fmgrid(range(coords.lamb[,1]),
+                       range(coords.lamb[,2]))
> par(mfrow = c(1, 2))
> temp <- setplot(coords.lamb, axis = TRUE)

> deform <- Ftransdraw_nuevo(disp = disp, Gcrds = coords.lamb,
+                            MDScrds = sg.est_nuevo$ncoords,
+                            gridstr = coords.grid, lambda = 0.50)
```

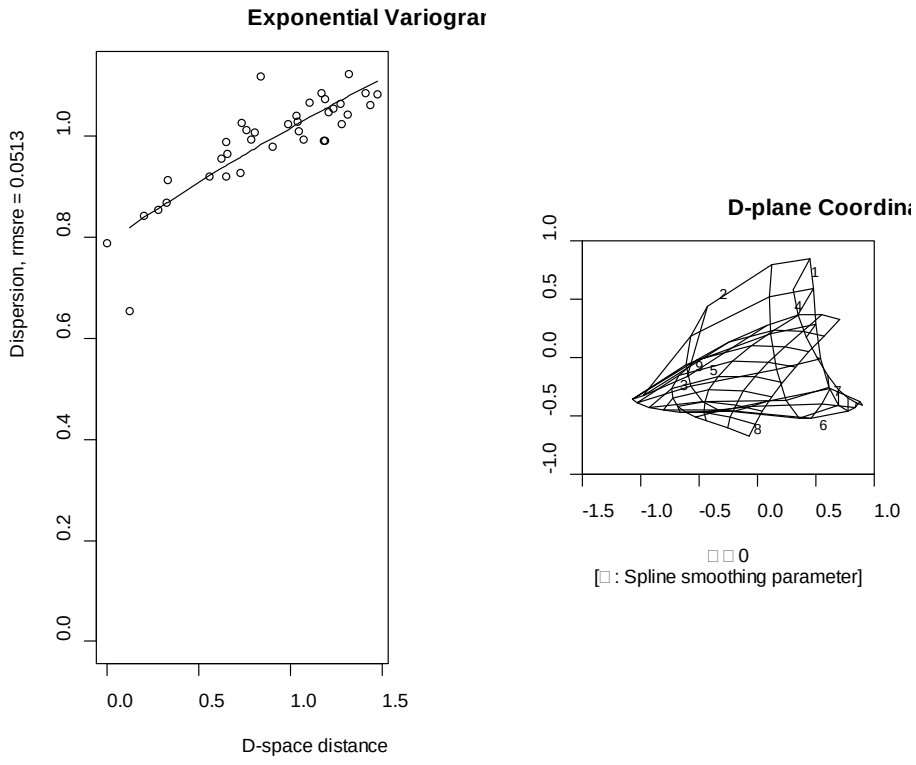


Figura 25. Cuadrícula del plano-D y variograma ajustado para $\lambda = 0$.

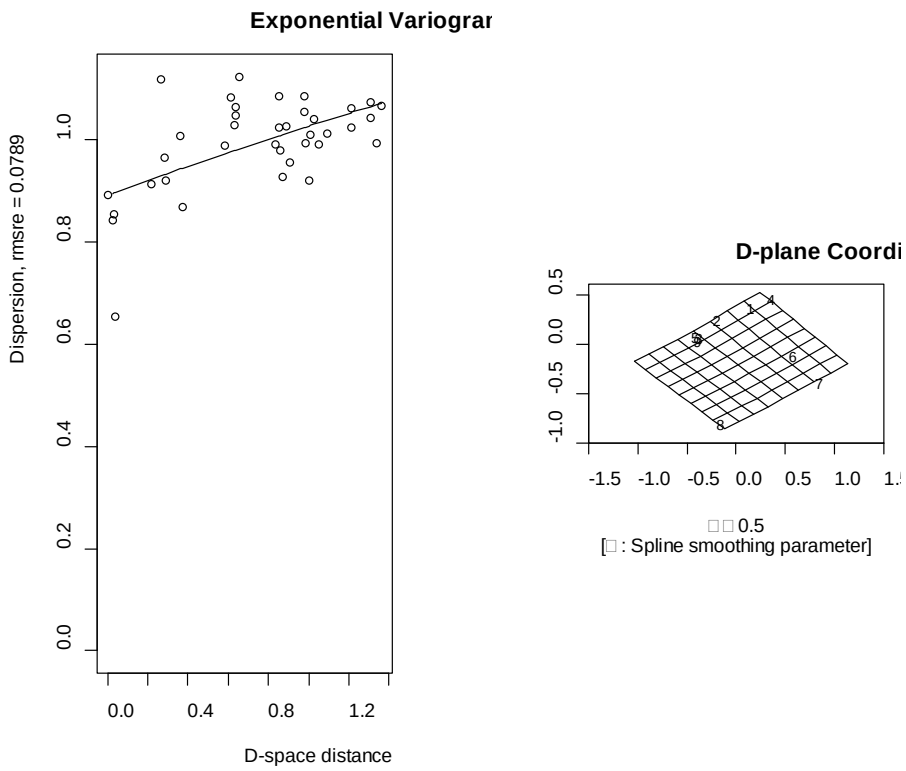


Figura 26. Cuadrícula del plano-D y variograma ajustado para $\lambda = 0.5$

En las figuras 25 y 26 se aprecia el efecto del parámetro de suavizado en la deformación del espacio-G en el espacio-D cuando este pasa de 0 a 0.5. En este ejemplo de aplicación se necesitó un valor menor del parámetro de suavizado (comparado al usado en la versión original) para obtener un espacio-D no doblado y un grado de ajuste del variograma incluso mejor que el del primer ejemplo.

En la etapa 3 se combinan los resultados de las dos etapas precedentes para obtener un spline de placa delgada óptimo. La función `sinterp` se utiliza para ajustar el spline de placa delgada con el parámetro de suavizado seleccionado en el paso anterior. Los coeficientes del spline de placa delgada optimizado (almacenados en el resultado `sol`) se utilizan como argumento de la función `bgrid` para evaluar la cuadrícula biortogonal.

```
> Tspline <- sinterp(coords.lamb, sg.est_nuevo$ncoords, lam = 0.5)

> par(mfrow = c(1, 1))
> par(mar=c(1,1,1,1))
> Tgrid <- bgrid(start=c(0,0), xmat=coords.lamb,
+                coef=Tspline$sol)
> tempplot <- setplot(coords.lamb, maxdim=c(9,10), ax=T)
> text (coords.lamb, labels = 1:nrow(coords.lamb))
> draw(Tgrid, fs=T)
```

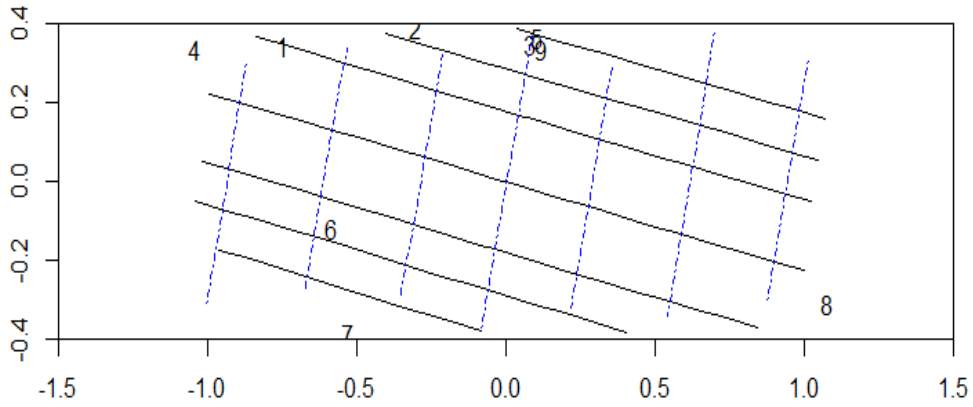


Figura 27. Red biortogonal para el spline de placa delgada

En la figura 27 se aprecia la cuadrícula biortogonal para el spline de placa delgada, esta representa la estructura de covarianza espacial del proceso espacio-temporal. En este caso las líneas discontinuas azules indican un estiramiento relativo del plano a lo largo del eje noreste – suroeste y una contracción en la dirección ortogonal, esto implica que la covarianza espacial es más débil en la dirección noreste – suroeste y más fuerte en la dirección sureste – noroeste.

En la etapa 4 se usa el spline de placa delgada del paso 3 y el variograma ajustado en la etapa 1 para estimar la dispersión entre las estaciones de monitoreo y las nuevas ubicaciones de interés, éstas últimas se crean empleando una cuadrícula de 100 puntos entre las estaciones y convirtiendo las coordenadas geográficas de las ubicaciones de interés a coordenadas de Lambert usando el mismo punto de referencia de la etapa 1.

```
> lat10 <- seq(min(location2[,1]), max(location2[,1]), length = 10)
> long10 <- seq(min(location2[,2]), max(location2[,2]), length = 10)
> llgrid <- cbind(rep(lat10, 10), c(outer(rep(1, 10), long10)))
```

Las coordenadas transformadas de las nuevas ubicaciones se normalizan y luego se combinan con las coordenadas de las estaciones de monitoreo.

```
> z <- coords2
> newcrds.lamb <- (Flamb2(llgrid, latrf1 = z$latrf1, latrf2= z$latrf2,
+                         latref=z$latref, lngref=z$lngref)$xy) *kdist

> allcrds <- rbind(newcrds.lamb, coords.lamb)
```

La función `corrfit` permite estimar las correlaciones entre todas las ubicaciones usando los parámetros del variograma ajustado y las distancias entre los sitios en el espacio-D; en el caso de las nuevas ubicaciones su posición correspondiente en el espacio-D se evalúa mediante el spline de placa delgada optimizado. A continuación se muestran las correlaciones entre las primeras cinco ubicaciones.

```
> corr.est <- corrfit(allcrds, Tspline = Tspline,
+                     sg.fit = sg.est_nuevo, model = 1)

> round(corr.est$cor[1:5, 1:5], 2)
     [,1] [,2] [,3] [,4] [,5]
[1,] 1.00 0.59 0.57 0.56 0.55
[2,] 0.59 1.00 0.59 0.57 0.56
[3,] 0.57 0.59 1.00 0.59 0.57
[4,] 0.56 0.57 0.59 1.00 0.59
[5,] 0.55 0.56 0.57 0.59 1.00
```

La última etapa del método de Sampson y Guttorp estima las varianzas del campo aleatorio en todas las ubicaciones y luego las combina con la matriz de correlación estimada en la etapa anterior para estimar la matriz de covarianza.

En primer lugar se observan los elementos de la diagonal de dicha matriz.

```
> diag(cov.est)
[1] 1.95538614 0.14955074 0.16859152 0.62513836 0.14150010 0.12339957
    0.16038352 0.08206694
[9] 0.13732444
```

Esto sugiere la ausencia de homogeneidad en el campo la cual puede ser suavizada empleando el mismo spline de placa delgada.

```
> Tspline.var <- sinterp(allcrds[101:109,],
+                        matrix(diag(cov.est), ncol = 1),lam = 0.5)
```

Luego se utiliza la función `seval` para obtener estimados de la varianza en las ubicaciones y los resultados de los cálculos se combinan para obtener la matriz de covarianza en todas las ubicaciones, finalizando así la implementación del método de Sampson y Guttorp para extender a sitios sin monitorear la matriz de covarianza, la cual es almacenada en `covfit.`

```
> varfit <- abs(seval(allcrds, Tspline.var)$y)
> temp <- matrix(varfit, length(varfit), length(varfit))

> covfit <- corr.est$cor * sqrt(temp * t(temp))
```

Los resultados anteriores son la base para efectuar la interpolación espacial del campo aleatorio. Para ello es necesario obtener una distribución predictiva que implica una estimación de los hiperparámetros asociados con las nuevas ubicaciones mediante la función `staircase.hyper.est`.

```
> u <- 100 # número de ubicaciones nuevas
> p <- 4   # dimensión de la respuesta multivariante
> hyper.est <- staircase.hyper.est(emfit = em.fit,
+                                  covfit = covfit, u = u, p = p)
```

A continuación se utiliza la distribución predictiva para generar realizaciones del campo, lo cual permite efectuar la interpolación de forma empírica. La función `pred.dist.simul` es usada para obtener 1000 realizaciones en el día 122 (30 de noviembre de 2009).

```
> simu <- pred.dist.simul(hyper.est, tpt = 122, N = 1000)
```

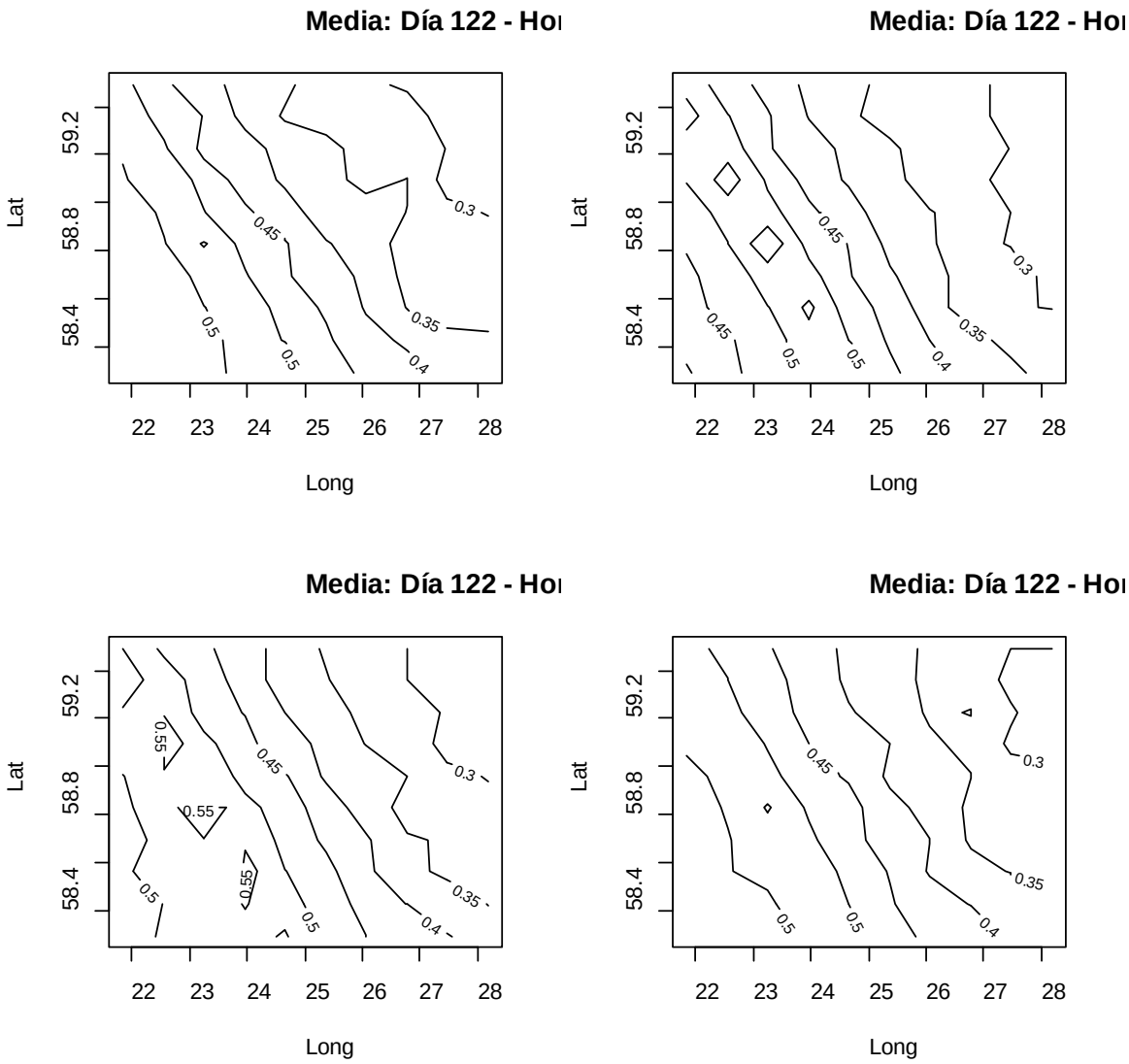


Figura 28. Gráficas de contorno de la media entre las 12:00 m y 3:00 pm del día 30/11/2009

Los datos de esta simulación pueden extraerse en las estaciones para generar gráficas de contorno de la media por horas, las cuales se aprecian en la figura 28.

```
> x <- apply(simu, 2, mean)[1:400]

> par(mfrow = c(2, 2))
> for (i in 1:4) {
+   tt <- i + 4 * 0:99
+   x1 <- x[tt]
+   hr <- matrix(x1, byrow = TRUE, ncol = 10)
+   print(range(x1 ))
+   contour(long10, lat10, hr, xlab = "Long", ylab = "Lat",
+           main = paste("Media: Día 122 - Hora ", 11+i))
+ }
```

```
[1] 0.2655740 0.5543951
[1] 0.2559844 0.5881726
[1] 0.2520632 0.5943345
[1] 0.2631018 0.5544370
```

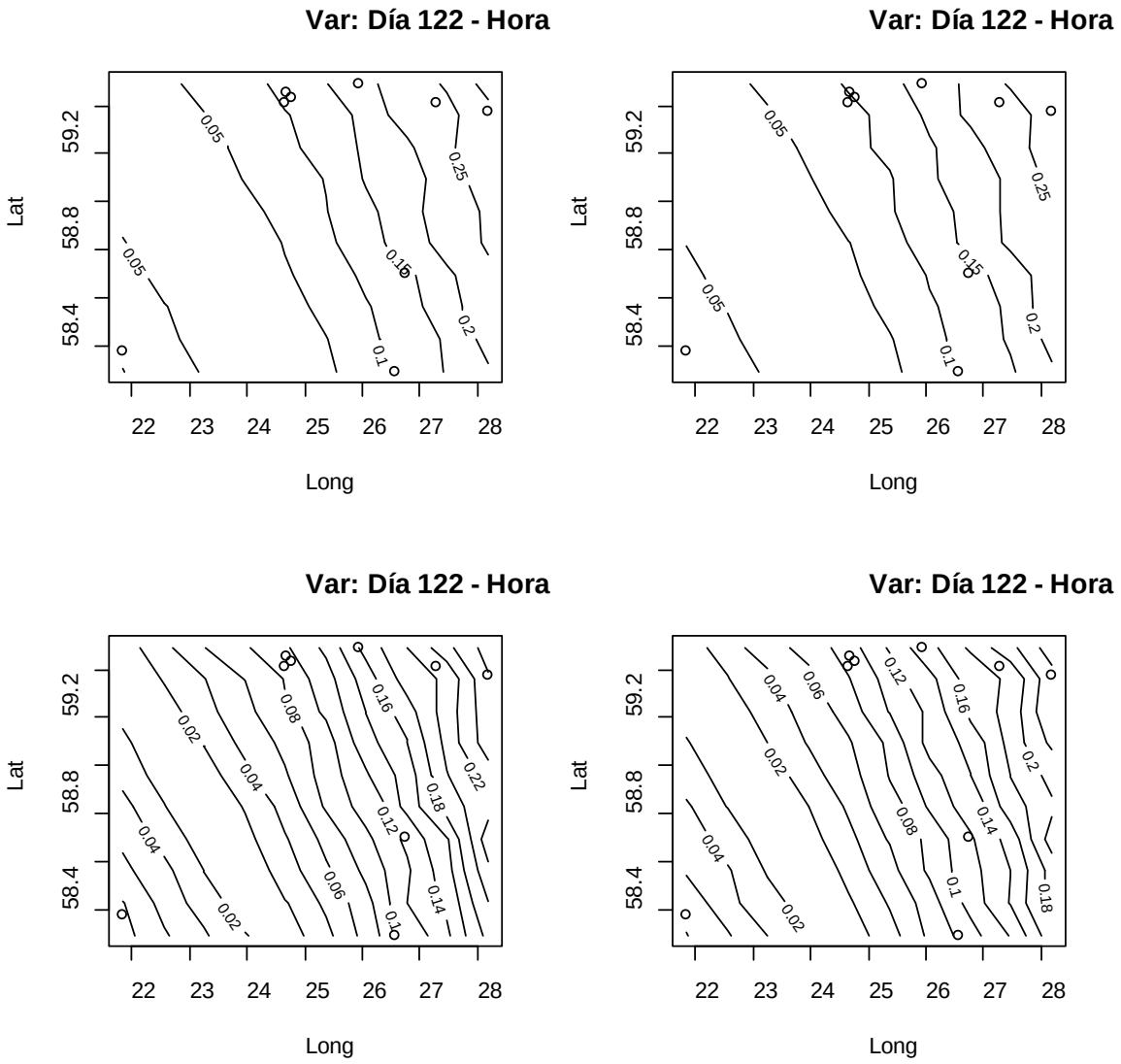


Figura 29. Gráficas de contorno de la varianza entre las 12:00 m y 3:00 pm del día 30/11/2009

Por último, los contornos correspondientes a la varianza del campo para cada hora se muestran en la figura 29.

```
> x <- simu[,1:400]
> par(mfrow = c(2, 2))

> for (i in 1:4) {
```

```
+   tt <- i + 4 * 0:99
+   x1 <- x[,tt]
+   x2 <- diag(var(x1))
+   vv <- matrix(x2, byrow = TRUE, ncol = 10)
+   contour(long10, lat10, vv, xlab = "Long", ylab = "Lat",
+           main = paste("Var: Día 122 - Hora ", 11+i))
+   points(location2[,2], location2[,1])
+ }
```

6. CONCLUSIONES Y TRABAJO FUTURO

El paquete *EnviroStat* permite la implementación del procedimiento de Sampson y Guttorp en R para la estimación de la estructura de covarianza espacial de un proceso espacio-temporal no estacionario empleando dos herramientas fundamentales: el escalamiento multidimensional (MDS) y la interpolación mediante spline de placa delgada.

Una de las limitaciones encontradas en la versión original del paquete *EnviroStat* es la ausencia de un criterio que determine la escala de las coordenadas de las ubicaciones transformadas mediante la proyección de Lambert. En efecto, la documentación del paquete solo sugiere que el usuario debe seleccionar un factor que al ser multiplicado por la matriz de coordenadas transformadas permita obtener una escala lo suficientemente pequeña para que no hayan problemas en el cálculo del spline de placa delgada.

Otro inconveniente que surge al aplicar el paquete es el tiempo que consume la ejecución de la primera etapa de la implementación del método de Sampson y Guttorp debido al elevado número de iteraciones requeridas para hallar un plano-D óptimo y un grado de ajuste satisfactorio del variograma.

También se encontró como limitación de la versión original del paquete la ausencia de un criterio claro para elegir el parámetro de suavizado del spline de placa delgada λ, el cual controla la deformación del espacio-G en el espacio-D al tiempo que mantiene un ajuste satisfactorio del modelo.

Para resolver estos problemas se introdujeron varias modificaciones en el paquete *EnviroStat*. La primera de ellas consiste en un método de normalización de la matriz de coordenadas de las ubicaciones en el espacio-G obtenidas después de utilizar la proyección de Lambert, esta normalización limita la arbitrariedad en la definición de la escala de la matriz de coordenadas y también se aplicó a la matriz de dispersión.

La siguiente modificación introducida al paquete fue la incorporación del paquete *smacof* en la primera etapa de implementación del método de Sampson y Guttorp para obtener un plano-D optimizado en un menor número de iteraciones, ya que la versión modificada permite obtener desde la primera iteración un plano-D mucho más cercano a la configuración óptima.

Las modificaciones hechas al paquete demostraron ser altamente efectivas para reducir el tiempo de ejecución de la primera etapa del método de Sampson y Guttorp; en efecto, en una de las pruebas realizadas con datos reales el número de iteraciones requeridas pasa de 82 en la versión original a solamente 2 en la versión modificada, lo cual representa un tiempo de ejecución aproximadamente 14 veces menor en la versión modificada. En la prueba efectuada se emplearon datos de 9 estaciones de monitoreo, es de esperar que la reducción de tiempo sea incluso mayor en un conjunto de datos proveniente de una red conformada por un mayor número de estaciones, lo que constituye una ventaja a la hora de aplicar la versión modificada del paquete a grandes volúmenes de datos.

El último cambio hecho al paquete *EnviroStat* es la incorporación de una metodología para la selección del parámetro de suavizado λ adoptando el enfoque de validación cruzada. Una ventaja adicional de la versión modificada del paquete es que la normalización de las matrices de dispersión y de coordenadas transformadas de las ubicaciones en el espacio-G hace posible que el intervalo de parámetros de suavizado necesario para obtener el balance adecuado entre el ajuste del variograma y la deformación espacial está entre 0 y 1, lo cual simplifica el procedimiento de validación cruzada tal como se mostró en los ejemplos de aplicación. En la versión original del paquete la magnitud del parámetro de suavizado requerido para conseguir dicho balance es considerablemente mayor, esto hace que sea más difícil la elección de λ.

Como trabajo futuro se propone ensayar la implementación del método de Sampson y Guttorp mediante la versión modificada de *EnviroStat* en un conjunto de datos que involucre un mayor número de estaciones de monitoreo, la comparación del enfoque utilizado en este trabajo con otros procedimientos de validación cruzada en la selección del parámetro λ y el desarrollo de

una función biyectiva que asigne puntos del espacio-G a puntos del espacio-D en R^3, ya que el spline de placa delgada utilizado por el paquete *EnviroStat* no sirve en este caso. Los splines de Duchon (Duchon, 1977) constituyen una generalización del spline de placa delgada y representan una alternativa factible en problemas de dimensión más elevada, la cual ha sido aplicada con éxito en casos donde el dominio del campo aleatorio posee fronteras complicadas (Miller & Wood, 2014).

7. REFERENCIAS BIBLIOGRÁFICAS

Borg, I., Groenen, P. (2005). Modern multidimensional scaling. Nueva York, Estados Unidos: Springer.

Borg, I., Groenen, P., Mair, P. (2013). Applied multidimensional scaling. Berlín, Alemania: Springer.

Duchon, J. (1977). Splines minimizing rotation-invariant semi-norms in Sobolev spaces. Constructive theory of functions of several variables, pp. 85-100.

Le, N., Zidek, J. (2006). Statistical analysis of environmental space-time processes. Nueva York, Estados Unidos: Springer.

Le, N., Zidek, J. (2013). [En línea]. EnviroStat. [Fecha de consulta: 6 de julio de 2016]. Disponible en: https://cran.r-project.org/web/packages/EnviroStat/vignettes/EnviroStat.pdf

Le, N., *et al*. (2015). [En línea]. Package 'EnviroStat'. [Fecha de consulta: 25 de agosto de 2016]. Disponible en: https://cran.r-project.org/web/packages/EnviroStat/EnviroStat.pdf

Leeuw, J., Mair, P. (2009). Multidimensional scaling using majorization: SMACOF in R. *Journal of Statistical Software*, *Vol*. 31 (3).

Løland, A., Høst, G. (2003). Spatial covariance modelling in a complex coastal domain by multidimensional scaling. *Environmetrics, Vol*. 14, pp. 307-321.

Mardia, K., Kent, J., Bibby, J. (1979). Multivariate Analysis. Londres, Gran Bretaña: Academic Press.

Miller, D., Wood, S. (2014). Finite area smoothing with generalized distance splines. *Environmental and Ecological Statistics, Vol*. 21, pp. 715-731.

Sampson, P., Guttorp, P. (1992). Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association, Vol*. 87, pp. 108-119.

Vera, J. F., Angulo, J. M., Roldán, J. A. (2016). Stability analysis in nonstationary spatial covariance estimation. *Stochastic Environmental Research and Risk Assessment*. [Fecha de consulta: 25 de agosto de 2016]. Disponible en: http://link.springer.com/article/10.1007/s00477-016-1228-4

Wickelmaier, F. (2003). [En línea]. An introduction to MDS. [Fecha de consulta: 10 de agosto de 2016]. Disponible en: https://homepage.uni-tuebingen.de/florian.wickelmaier/pubs/Wickelmaier

2003SQRU.pdf