

UNIVERSIDAD SAN FRANCISCO DE QUITO

Colegio de Posgrados

**Generación de superficies climáticas usando datos funcionales de
temperatura y precipitación por medio de métodos geoestadísticos para el
Valle del Río Cauca, Colombia**

Jorge Antonio Cardona Rios

Richard Resl, Ph.D.(c), Director de Tesis

Tesis de grado presentada como requisito
para la obtención del título de Magister en Sistemas de Información Geográfica

Quito, febrero de 2015

Universidad San Francisco de Quito

Colegio de Postgrados

HOJA DE APROBACIÓN DE TESIS

Generación de superficies climáticas usando datos funcionales de temperatura y precipitación por medio de métodos geoestadísticos para el Valle del Río Cauca, Colombia

Jorge Antonio Cardona Rios

Richard Resl, Ph.D.(c)

Director de la tesis

Pablo Cabrera, Msc.

Miembro del Comité de Tesis

Richard Resl, Ph.D.(c)

Director del Programa de Maestría en

Sistemas de Información Geográfica

Stella de la Torre, Ph.Dc.

Decana de Colegio de Ciencias

Biológicas y Ambientales

Victor Viteri Breedy, Ph. Dc.

Decano del colegio de Postgrados

Quito, febrero de 2015

©DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído la Política de Propiedad Intelectual de la Universidad San Francisco de Quito y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo de investigación quedan sujetos a lo dispuesto en la Política.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo de investigación en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

.....
Jorge Antonio Cardona Rios
Pasaporte AP693486

Quito, febrero de 2015

DEDICATORIA

A las personas que han estado junto a mí, que me han amado, comprendido y ayudado, a mis padres, mi esposa, mis hijos, hermanos, amigos y compañeros.

AGRADECIMIENTOS

Este trabajo fue desarrollado dentro del Área de Análisis de Políticas y Decisiones (DAPA) por su sigla en inglés liderado por Andrew Jarvis en el Centro Internacional de Agricultura Tropical (CIAT).

A Andrew Jarvis por las oportunidades de mejorar a través del estudio.

A Carlos Meneses por la oportunidad de mejorar continuamente.

A Glenn Hyman y Anton Eitzinger por darme el apoyo y la guía para desarrollar este trabajo

A David Arango por darme sus aportes desinteresados.

A Victor Hugo Patiño por su colaboración.

A Peter Jones por sus conocimientos.

A Juan Guillermo Perez, Idris Jone y Antonio Pantoja por su gran ayuda y colaboración.

Muy especialmente a todos los amigos que tengo en el proyecto que me han brindado su ayuda y guía durante este recorrido, además de los foros de discusión donde he podido encontrar muchos consejos, ya que sin ellos no hubiera podido llevar a cabo este proyecto

RESUMEN

Sin duda alguna el clima es un fenómeno que cada día se torna con mayor importancia, ya que todas las actividades sobre la tierra están limitadas o influenciadas por él y su estudio siempre será tema de investigación y discusión; para el siguiente estudio se analizará las variables climáticas desde un punto de vista de valor-datos funcional, en donde por medio de técnicas geoestadísticas se crea un modelo para predecir el comportamiento climático en sitios no muestreados y generar así las superficies.

El enfoque de este estudio es teórico-práctico, ya que constituye una técnica estadística reciente; empezando por la identificación y exploración de los datos obtenidos por las estaciones climáticas, además de realizarse un análisis descriptivo del clima, el cual muestra el estilo de los datos de manera individual (por estación climática) , siguiendo el proceso de control de calidad y la generación de datos faltantes por medio del software estadístico R y el paquete RMAWGEN *multi-sites auto-regressive weather generator*. A continuación, con el apoyo de la librería *functional data analysis* FDA (J. O. Ramsay & Silverman, 2005) se realizó el análisis funcional de datos, encontrándose mediante métodos geo-estadísticos de evaluación el mejor ajuste de las curvas. Posteriormente, con la aplicación de la librería de R (geofd), se realizó el análisis final y la generación de las curvas del modelo; generándose las superficies climáticas de tipo raster. Todos los procesos se acompañan de argumentos teóricos, con el fin de apoyar el entendimiento de las técnicas estadísticas espaciales aplicadas.

PALABRAS CLAVES:

Funciones bases, Cokriging, Validación cruzada, Modelación de datos funcionales, Predicción Espacial Multivariante

ABSTRACT

Certainly the climate is a phenomenon that every day becomes more important, as all activities on land are limited or influenced by it. Its study will always be subject of research and discussion; the following project will analyze climatic variables from the point of view of valuable-Functional Data, where using geostatistical techniques create a model to predict climate behavior at unsampled sites and generate surfaces. The focus of this study is purely theoretical and practical, since it is a new statistical technique; beginning with the identification and exploration of the data obtained by weather stations, and a descriptive analysis of the climate, which shows the style of data individually (for weather station) done, following the process of quality control and the missing data generation package through Multi-Sites Auto-regressive Weather Generator statistical software R and RMAWGEN. Then, with the support of the Functional Data Analysis (fda) library (JO Ramsay & Silverman, nd) (JO Ramsay & Silverman, nd) a functional data analysis was performed by finding geo-statistical methods of evaluating the best fit of the curves. Later, with the implementation of the R library (geofd) , the final analysis and the generation of the model curves was performed; generating the gridded climate surfaces. All processes are accompanied by theoretical arguments to support the understanding of spatial statistical techniques applied.

KEYWORDS:

Functional bases, Cocriging, Cross Validation, Functional Data Modelling, Multivariate Spatial Prediction

TABLA DE CONTENIDO

RESUMEN	7
ABSTRACT	8
LISTA DE TABLAS	11
LISTA DE FIGURAS	12
INTRODUCCION	13
1.1 Antecedentes	15
1.2 El problema	15
1.2.1 Pregunta de investigación	16
1.3 Hipótesis	16
1.4 Contexto y Marco Teórico	16
1.4.1 Propósito y significado del estudio.....	17
MARCO DE REFERENCIA	18
1.5 Análisis Exploratorio de Datos (AED)	18
1.6 R multi-site auto-regressive weather generator (RMAWGEN)	18
1.7 Tipos de datos espaciales	18
1.7.1 Datos geoestadísticos o georeferenciados (geostatistical data).....	18
1.7.2 Datos en rejilla o datos en un área (lattice data).....	19
1.7.3 Datos de procesos puntuales (point process data).....	19
1.8 Estadística espacial	20
1.9 Geoestadística	20
1.10 Estudio del Variograma y Semivariograma	21
1.11 Isotropía	23
1.11.1 Anisotropía geométrica.....	23
1.11.2 Anisotropía zonal.....	24
1.12 Kriging	24
1.12.1 Kriging ordinario (KO).....	26
1.12.2 Varianza de predicción de Kriging ordinario (KO).....	27
1.12.3 Validación del Kriging ordinario (KO).....	27
1.12.4 Semivariogramas.....	28
1.12.1 Estacionariedad.....	30
1.13 Análisis de datos funcionales (ADF)	33
1.14 Bases para datos funcionales	34
MATERIALES Y MÉTODOS	35
1.15 Bases de datos climáticos	35
1.16 Equipos	35
1.17 Programas	35
1.18 Descripción del proceso	36
RESULTADOS Y discusión	39
1.19 Localización del área de estudio	39
1.20 Distribución de pisos térmicos	40
1.21 Caracterización del clima	40

1.22	Temperatura	42
1.23	Precipitación	42
1.24	Estadística descriptiva de los datos de temperatura mínima	42
1.25	Análisis Gráfico.....	43
1.26	Análisis descriptivo	44
1.27	Control de calidad y generación de datos faltantes en las series diarias.....	46
1.28	Predicción espacial Geoestadística para datos funcionales	48
CONCLUSIONES.....		61
BIBLIOGRAFIA.....		64
ANEXOS.....		66
Anexo 1. Listado de tablas		66
Anexo 2. Histogramas de series históricas de temperatura máxima, mínima y temperatura.....		71
Anexo 3. Listado de código en lenguaje R		78

LISTA DE TABLAS

Tabla 1. Tipos de predictores kriging y sus propiedades.....	24
Tabla 2. Principales tipos de kriging lineal y sus propiedades evaluados en la estación Aeropuerto.....	53
Tabla 3. Resumen de evaluación del desempeño de los modelos SSE.....	53
Tabla 4 Listado de estaciones climáticas	66
Tabla 5 Estadísticos de Resumen Temperatura Máxima.....	67
Tabla 6 Estadísticos de Resumen Temperatura Mínima.....	69
Tabla 7 Estadísticos de Resumen Precipitación.....	70

LISTA DE FIGURAS

Figura 1. Comportamiento típico de un semivariograma acotado con una representación de los parámetros básicos. SEMEXP corresponde al semivariograma experimental y al ajuste de un modelo (Giraldo, 2009a)	22
Figura 2 Representación de una superficie interpolada para una variable.....	31
Figura 3 Representación de una superficie interpolada para una variable regionalizada no estacionaria (Giraldo, 2009b)	32
Figura 4. Diagrama de flujo, del proceso metodológico seguido, adicionalmente en las siguientes secciones se detalla el procedimiento seguido. En este esquema en recuadros azules se muestra los procesos desarrollados en ArcGis, y en rojo cada una de las herramientas utilizadas.....	36
Figura 5. Ubicación área de estudio.....	39
Figura 6. Zona de Confluencia Intertropical (ZCIT) www.wikipedia.com	41
Figura 7 Distribución de registros por estación	43
Figura 8 Ejemplo de datos faltante en las series	44
Figura 9. Gráficos de las series resumen de temperatura y precipitación.....	49
Figura 10. Representación de las bases evaluadas.....	51
Figura 11. Comparación temperatura normal y suavizada b-spline.	51
Figura 12. Comparación de precipitación promedio y suavizada.....	52
Figura 13. Comparación de los modelos Exponencial, Esférico, Gaussiano y Matérn	53
Figura 14 Validación Cruzada General de Modelo estación aeropuerto.	54
Figura 15 Comparación de curvas estimada y observada en la estación de aeropuerto.	54
Figura 16. Cambio de resolución del Modelo de Elevación.....	55
Figura 17 Función resultante estimada, en la selección en un punto.....	55
Figura 18 Resultado de superficies rasterizadas de temperatura máxima	57
Figura 19 Resultado de superficies rasterizadas de temperatura mínima	57
Figura 20 Resultado de superficies rasterizadas de precipitación	58

INTRODUCCION

Para el diseño y desarrollo de proyectos relacionados con agricultura, medio ambiente y el calentamiento global; el componente climático se hace un factor fundamental, ya que en el caso específico de la agricultura no se podría realizar ningún análisis completo, sin tener en cuenta este factor. No obstante, se puede presentar situaciones que no se pueda contar con datos climáticos, o que éstos no estén disponibles parcial o totalmente para el área requerida, debido a la dificultosa cobertura o localización. A menudo, esta situación se puede deber a diferentes factores, que pueden ir desde la falta de estaciones meteorológicas en el área de interés, hasta un corto presupuesto para adquirirlos; lo anteriormente expuesto genera la necesidad de tener información en sitios no accesibles o sin cobertura. De esta necesidad nace la motivación de este trabajo que ofrece una solución al problema de predecir y generar datos climáticos en sitios no muestreados bajo la ausencia de estacionariedad, aplicando métodos estadísticos estándar que se han generalizados para ser utilizados en el análisis de datos funcionales (J. Ramsay & Silverman, 1997) y técnicas geoestadísticas que se han adaptado a este tipo de datos con el fin de realizar la predicción espacial (Jim Ramsay & Silverman, 2009).

Tomando los aportes de Goulard y Voltz (1993) (Gouillard M. & Volts M., 1993) que trataron el problema de predicción espacial funcional en sitios no muestreados bajo la existencia de estacionariedad, en cuyo trabajo las funciones son conocidas a partir de un conjunto finito de puntos, y un modelo paramétrico los ajusta para reconstruir la curva completa. Giraldo y colaboradores (2010) (Delicado, Giraldo, & Mateu, 2007) (Comas, Delicado, Giraldo, & Mateu, 2009) retoman las contribuciones de hechas por Goulard y Volts (1993) y proponen aplicar un ajuste no-paramétrico al pre-proceso de las funciones observadas llamado suavización por bases de funciones, donde el parámetro de suavización es elegido mediante el proceso llamado validación cruzada funcional (VCF) , proponiendo la metodología de kriging ordinario para datos funcionales donde la función a predecir es una combinación lineal de las curvas observadas, que igualmente fue propuesto por Goulard y Voltz (1993).

Aplicando los métodos geo-funcionales propuestos por Giraldo y colaboradores, en este trabajo se evaluaron 28 estaciones climáticas puntuales de series de temperaturas y precipitación, registradas por un periodo de 16 años (1997-2012) continuos y distribuidas a lo

largo del Valle del Río Cauca, Colombia. Por medio de análisis estadísticos donde se aplicó un proceso de calidad, eliminándolos datos anómalos y llenado de faltantes, seguido de una etapa de resumen de series. La transformación a función-valor se realizó promedio de un proceso de suavización usando bases de Fourier; luego se hallaron los parámetros del variograma usando una técnica de validación cruzada y por último la etapa de predicción por medio de kriging ordinario usando los parámetros en la etapa anteriormente hallados y finalmente la generación de las diferentes superficies climáticas.

1.1 Antecedentes

El interés de la comunidad estadística en el desarrollo en temas de análisis de datos funcionales ha permitido que nuevos métodos hoy estén disponibles, permitiendo la interpolación de curvas cuando se dispone de una muestra de las curvas en una región con continuidad espacial, este problema fue estudiado inicialmente por Goulard y Voltz (1993), en su caso, las funciones eran conocidas solamente por un conjunto finito de puntos y el modelo paramétrico se ajustaba para reconstruir la curva entera, en este trabajo, el modelo paramétrico se suponía conocido y tanto el total de puntos conocidos para cada función como el número de los parámetros en el modelo paramétrico, se asumía que eran pequeños. Este estudio fue aplicado inicialmente a estudios de ciencias del suelos véase (Gourlard M. & Volts M., 1993). Los estudios más actualizados como el trabajo de Jim Ramsay y Bernar Silverman que proponen nuevas técnicas para análisis de datos funcionales aplicado a diferentes áreas del conocimiento respalda esta afirmación. En Giraldo y colaboradores (2010), se propone el uso de la librería `geofd` dentro del software R versión 2.5. Este trabajo implementa una técnica geoestadística de kriging ordinario para este tipo de curvas, suavizado por medio de funciones base de Fourier y b-splines, para ilustrar la técnica usa parte de los datos de temperatura del estudio citado por Ramsay y Silverma (2005 y 2009), cubriendo tres provincias (Nueva Scotia, New Brunswick y Prince Edward Island), en particular se analizó información de medias en series diarias de temperatura para 30 años, observadas por 35 estaciones. La integración de estos métodos a los sistemas de información geográfica (SIG) tendría gran importancia ya que aportaría un grupo de técnicas aplicables a las ciencias en temas de medio ambiente, ciencias del suelo, entre otros.

1.2 El problema

Con el propósito de entender los diversos fenómenos naturales, que normalmente son registrados tanto en el espacio como en el tiempo, estos fenómenos se ubican usualmente en una región con continuidad espacial y suelen representarse por medio de curvas, que usando técnicas de suavizado se convierten en datos funcionales; así mismo aplicando técnicas funcionales que permiten identificar la tendencia en media y varianza, explorar la estructura de autocorrelación inherente al conjunto de datos medidos en esta región y poder así dar

solución al problema de predicción de curvas en zonas no muestreadas en esa región. Actualmente los sistemas de información geográfica (SIG) no contemplan los nuevos desarrollos en los datos y análisis funcional, posiblemente se debe a que son técnicas estadísticas recientes y muchas de las metodologías aún están en desarrollo. Con la necesidad de entender y predecir los diferentes fenómenos naturales, como en este caso temperatura y precipitación en el Valle del Río Cauca, se recurre a estos nuevos avances en estadística; estos son el análisis de datos funcionales y la geoestadística y dar solución a la generación de datos climáticos en zonas no muestreadas, enfocando la solución desde una visión funcional.

1.2.1 Pregunta de investigación

Las preguntas concretas planteadas para la actual investigación son:

¿Se puede tener disponibles datos climáticos diarios interpolados a partir de series de datos de estaciones puntuales por medio de métodos funcionales?

¿Qué tipo de datos se genera a partir de las estimaciones?

¿Qué resolución mínima se logra con el método propuesto en las nuevas superficies?

¿Cuál es la distribución espacial resultante?

¿Cómo es el comportamiento de la temperatura y la precipitación en la zona de estudio?

¿Se pueden usar las superficies en algún software SIG ejemplo Arcgis y que formato tienen las superficies resultantes?

1.3 Hipótesis

Las técnicas de modelación geoestadísticas y análisis funcional, pueden aplicarse a series de datos diarios climatológicos de estaciones puntuales en el Valle del Cauca, Colombia y producir superficies, donde cada celda resultante es una función obtenida por interpolación de curvas.

1.4 Contexto y Marco Teórico

Se requiere analizar y generar las superficies de datos climatológicos del Valle del Río Cauca, para ello se cuenta con una red de estaciones meteorológicas las cuales proporcionan las series de datos climáticos, luego estas series serán analizadas y

transformadas en datos funcionales, para lo cual se usa el software R y Arcgis, que tiene todas las herramientas para llevar a cabo este análisis y así generar las superficies requeridas.

1.4.1 Propósito y significado del estudio

En el siguiente trabajo tiene el propósito de sistematizar, analizar y producir superficies climáticas interpoladas, mediante la aplicación de técnicas funcionales y geoestadísticas, usando el software R. Para demostrarlo se contó con series de temperatura y precipitación, de un grupo de 28 estaciones meteorológicas, en el área del Valle del Río Cauca, Colombia.

MARCO DE REFERENCIA

1.5 Análisis Exploratorio de Datos (AED)

Es un conjunto de técnicas estadísticas cuyo objetivo es entender los datos, consiste en detectar propiedades, tendencias, anomalías y la evaluación de datos ausentes por medio de métodos sencillos a través de gráficos y técnicas cuantitativas; así, la secuencia de un análisis exploratorio inicia en gráficas simples (diagramas de dispersión, gráficos condicionales y diagramas de cajas) seguido de técnicas para organizar y preparar los datos. No obstante, para realizarlo se puede seguir las siguientes etapas.

- Preparar los datos (formatear, eliminar campos no útiles para que sean accesibles a cualquier técnica.
- Realizar un examen gráfico de la naturaleza de las variables a evaluar, realizando un análisis descriptivo numérico que permita cuantificar algunos aspectos gráficos de los datos.
- Realizar un examen gráfico de las relaciones entre las variables analizadas y un análisis descriptivo numérico.
- Identificar los posibles datos atípicos.
- Evaluar el impacto de los datos ausentes o faltantes.

1.6 R multi-site auto-regressive weather generator (RMAWGEN)

Es un generador de series de precipitación y temperatura diaria de valores promedios mensuales, funciona bajo el ambiente R como un paquete o librería asociado; este paquete contiene funciones S3 y S4 para reproducción estocástica espacial multi-sitios, con el fin de generar series de tiempo para variables de precipitación y temperatura (Cordano & Eccel, 2013). Además, dicha herramienta implementa un sistema de modelo de Vector Auto Regresivo (VAR's), ya que es calibrado por un instrumento de series de tiempo diarias.

1.7 Tipos de datos espaciales

1.7.1 Datos geoestadísticos o georeferenciados (geostatistical data)

Los datos espacialmente continuos son mediciones tomadas en puntos fijos con localizaciones continuas en el espacio. Las mediciones clásicas en salud ambiental, tales

como contaminación de suelos y de aire o la radiación natural, suelen pertenecer a esta categoría. La variable medida, sin embargo, puede ser tanto continua como discreta. El objetivo es el de, dados los valores tomados en puntos de muestreo fijos, extender la distribución espacial de los valores de un atributo sobre la región total de estudio. Por lo tanto el análisis de los datos geoestadísticos puede contemplar tanto la modelización del patrón de variabilidad, la determinación de los factores con los que pueda estar relacionado, como la de obtener una buena predicción de la variable en puntos donde no se ha muestreado. Estos métodos son muy utilizados en estudios del área de geociencias (suelo, clima, hidrología, geología minera, etc.) por lo que también suelen ser denominados datos geoestadísticos (Giraldo, 2009b).

1.7.2 Datos en rejilla o datos en un área (lattice data)

Los datos en rejilla son observaciones procedentes de un proceso aleatorio, observadas sobre una colección contable de regiones espaciales, que pueden estar regular o irregularmente distribuidas, complementados con lo que se denomina estructura de vecindad, es decir, información sobre las regiones vecinas. Matemáticamente una rejilla queda definida como un conjunto de lados y de vértices, es decir, un conjunto de índices de localizaciones con un conjunto asociado de vecinos. Ya que los datos en rejilla están definidos en regiones espaciales, las localizaciones concretas especificadas por el vector suelen referirse al centroide de la región (Giraldo, 2009b).

1.7.3 Datos de procesos puntuales (point process data)

Se denominan datos patrones de puntos o procesos puntuales, cuando las localizaciones (y no las mediciones) son las variables de interés. Consisten en un número finito de localizaciones observadas en una región determinada. El objetivo de los procesos puntuales es el de conocer la variación de la intensidad de los eventos sobre la región de estudio y el de buscar modelos que ayuden a explicar o comprender el fenómeno. Tan importante como la variable estudiada suele ser el patrón de variación espacial (Giraldo 2009b).

1.8 Estadística espacial

Es una colección de métodos estadísticos, en que la localización espacial, juega un rol implícito en el análisis de datos que corresponden a la medición de variables aleatorias en diversos sitios (puntos del espacio o agregaciones espaciales) de una región (Giraldo, 2007). De manera más formal se puede decir que la estadística espacial trata con el análisis de realizaciones de un proceso estocástico $\{Z(\mathbf{s}) : \mathbf{s} \in D\}$, en el que $\mathbf{s} \in \mathbf{R}^d$ representa una ubicación en el espacio euclidiano dimensional, $Z(\mathbf{s})$ es una variable aleatoria en la ubicación \mathbf{s} y \mathbf{s} varía sobre un conjunto de índices $D \subset \mathbf{R}^d$ (Giraldo, 2009b).

1.9 Geoestadística

La geoestadística es una rama de la estadística que trata fenómenos espaciales (Journel & Ch. J. Huijbregts, 1978), desarrollada originalmente para predecir probabilidad de distribución en operaciones mineras, es aplicada hoy día a diferentes disciplinas como geología de petróleos, hidrología, meteorología, geoquímica, forestales, control ambiental, ecología, epidemiología, entre otros. Así mismo, se ocupa principalmente de la "representación espacial de variables", asociando la representación de superficies geográfica con los puntos no muestreados. La modelación espacial es la adición más reciente a la literatura estadística; cualquier disciplina que trabaja con datos colectados en diferentes localizaciones espaciales necesita desarrollar modelos que indiquen cuando hay dependencia entre las medidas de los diferentes sitios. Usualmente, dicha modelación concierne con la predicción espacial, pero hay otras áreas importantes como la simulación y el diseño de muestreo (Cressie, 1989).

Cuando el objetivo es hacer predicción, la geoestadística opera en dos etapas: La primera concierne al análisis, en la cual se describe la correlación entre puntos en el espacio. La segunda fase hace la predicción en sitios de la región no muestreados por medio de kriging, siendo este proceso el que calcula un promedio ponderado de las observaciones muestrales. Los pesos asignados a los valores muestrales son apropiadamente determinados

por la estructura espacial de correlación establecida en la primera etapa y por la configuración del muestreo (Petitgas, 1996).

1.10 Estudio del Variograma y Semivariograma

Son herramientas que caben en la categoría de diagramas estadísticos, que permiten analizar el comportamiento espacial de una variable sobre una área definida, se obtiene como resultado un gráfico de los datos a diferentes distancias, A partir de los datos proporcionados por el variograma experimental se realiza la estimación del kriging. Las razones por las cuales se estudió estos fenómenos son:

- La hipótesis es intrínsecamente estacionaria e incluye la condición de estacionaria de segundo orden.
- Para el cálculo del semivariograma no es necesario conocer la media de la variable.

Por otra parte, el variograma denotado por $2\gamma(h)$, se define como la varianza de la diferencia entre variables separadas por una distancia $h = \|s_i - s_j\|$ asumiendo que $\mathbb{E}(Z(s)) = \mu$ se tiene que:

$$\begin{aligned} 2\gamma(h) &= \mathbb{V}(Z(s+h) - Z(s)) \\ &= \mathbb{E}(Z(s+h) - Z(s))^2 \end{aligned} \quad \text{Ecuación 1}$$

La mitad del variograma es el semivariograma, el cual se caracteriza por las propiedades de dependencia espacial de un fenómeno espacial. Esta función es usualmente empleada para tratar datos de un fenómeno que presente continuidad espacial (datos geoestadísticos). Usando el método de momentos se tiene que un estimador del semivariograma es:

$$\gamma(h) = \frac{1}{n(h)} \sum^{n(h)} (Z(s+h) - Z(s))^2 \quad \text{Ecuación 2}$$

Donde $n(h)$ representa el número de parejas de sitios (S_i, S_j) , que se encuentran separados por una distancia h . En la práctica, debido a irregularidad en el muestreo y por ende en las distancias entre los sitios, se toman intervalos de distancia $[0, h], [h, 2h], [2h, 3h], \dots\}$ y el semivariograma experimental corresponde a una distancia promedio entre parejas de

sitios dentro de cada intervalo y no a una distancia h específica. Obviamente el número de parejas de puntos n dentro de los intervalos no es constante. Para interpretar el semivariograma experimental se parte del criterio de que a menor distancia entre los sitios mayor similitud o correlación espacial entre las observaciones. Por lo tanto, en presencia de autocorrelación se espera que para valores de h pequeños el semivariograma experimental, tenga magnitudes menores a las que este toma cuando las distancias se incrementan. La solución del problema de predicción espacial requiere del conocimiento de la estructura de autocorrelación para cualquier posible distancia entre sitios dentro del área de estudio. De la *Ecuación 1* es claro que el semivariograma muestral es calculado sólo para algunas distancias promedios particulares. Por ello se hace necesario el ajuste de modelos que generalicen la dependencia espacial para cualquier distancia. No obstante, existen diversos modelos teóricos de semivarianza que pueden ajustarse al semivariograma muestral (Giraldo, 2009b)

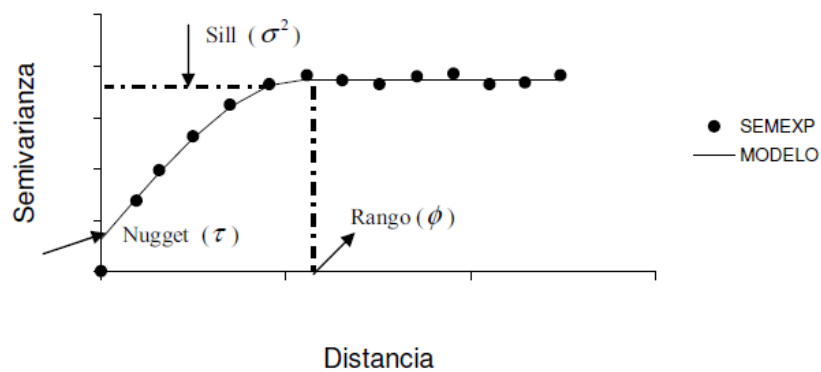


Figura 1. Comportamiento típico de un semivariograma acotado con una representación de los parámetros básicos. SEMEXP corresponde al semivariograma experimental y al ajuste de un modelo (Giraldo, 2009a)

En general los modelos pueden dividirse en no acotados (lineal, logarítmicos, potencial) y acotados (esféricos, exponencial, gaussiano, matérn) (Samper & Carrera, 1993). Los del segundo grupo garantizan que la covarianza de los incrementos es finita, por lo cual son ampliamente usados cuando hay evidencia de que presentan buen ajuste. La mayoría de modelos empleados para ajustar el semivariograma maestro, tiene tres parámetros en común (Figura 1), los cuales son descritos a continuación:

- **Pepita (Nugget) (τ):** Representa una discontinuidad puntual del semivariograma en el origen (Figura 1). Puede ser debido a errores de medición en la variable o a la

escala de la misma. En algunas ocasiones puede ser indicativo de que parte de la estructura espacial se concentra a distancias inferiores a las observadas.

- **Meseta (*Sill*) (σ^2):** Varianza a priori, es el máximo valor que alcanza un semivariograma, ya que es un estimador de la varianza de las variables del proceso. También puede definirse como el límite del semivariograma cuando la distancia h tiende a infinito, indicando la escala bajo la cual los datos definen un proceso estacionario de segundo orden.
- **Rango (ϕ):** En términos prácticos corresponde a la distancia a partir de la cual dos observaciones son independientes. El rango se interpreta como la zona de influencia.

Existen algunos modelos de semivariograma en los que no existe una distancia finita para la cual dos observaciones sean independientes; es por ello que se llama rango efectivo a la distancia para la cual el semivariograma alcanza el 95% de la meseta. Si la variable es estacionaria de segundo orden, siempre puede obtenerse la función de covariograma a partir de la función de semivariograma. De ahora en adelante, por facilidad de su notación se denota el variograma isotrópico $\gamma(h)$, donde h representa un índice de cambio que muestra una variable con la distancia. Su formato describe el patrón de variación espacial en términos de su magnitud y forma general.

1.11 Isotropía

Un proceso isotrópico es aquel cuya correlación entre los datos no depende de la dirección en la que ésta se calcule, de lo contrario se hablará de anisotropía (Giraldo, 2009b).

1.11.1 Anisotropía geométrica

Esta se refiere a un semivariograma de la siguiente forma.

$$\gamma(h) = \gamma_0(\| Ah \|) \quad \text{Ecuación 3}$$

Donde $\gamma_0(\| Ah \|)$ es un variograma isotrópico y A una matriz de dimensión $d \times d$ representando una transformación lineal de \mathbb{R}^d . Lógicamente si A es la identidad, esto se reduce al caso isotrópico. La idea subyacente para esta situación, es que el proceso no es isotrópico en el espacio original, pero sí en algún espacio transformado linealmente, el cual puede, por ejemplo, corresponder a una transformación de las coordenadas. En el caso más

lógico y usual, en el que A es una matriz definida positiva, los contornos de igual covarianza se corresponden con elipses inscritas en círculos.

1.11.2 Anisotropía zonal

Una posible generalización de anisotropía surge de la simple observación; si Z_1, \dots, Z_p son procesos independientes intrínsecamente estacionarios, entonces $Z = Z_1 + \dots + Z_p$, es también intrínsecamente estacionario, con semivariograma dado por $\gamma(\mathbf{h}) = \gamma_1(\mathbf{h}) + \dots + \gamma_p(\mathbf{h})$, denotando $\gamma_1, \dots, \gamma_p$ los semivariogramas de Z_1, \dots, Z_p respectivamente así:

$$\gamma(\mathbf{h}) = \sum_{i=1}^p \gamma_0(A_i \mathbf{h}) \quad \text{Ecuación 4}$$

Siendo γ_0 un semivariograma isotrópico y A_1, \dots, A_p matrices, es un semivariograma válido que generaliza la anisotropía geométrica. Esta anisotropía se llama anisotropía zonal. Una idea más complicada es asumir que, para alguna función no lineal $g(u)$, el proceso $Z(g(u))$, en lugar del original $Z(u)$, es un proceso isotrópico estacionario. Esta idea puede, de hecho analizarse, tanto la no estacionariedad como la no isotropía.

1.12 Kriging

La palabra kriging procede del nombre del geólogo sudafricano D. G. Krige (Justiniano & Junior, 2004), cuyo trabajo en la predicción de reservas de oro realizada en la época del cincuenta, se consideran como pioneros en los métodos de interpolación espacial. Kriging encierra un conjunto o familia de métodos de predicción espacial que se fundamentan en la minimización del error cuadrático medio de predicción. En la Tabla 1 se mencionan los tipos de kriging y algunas de sus propiedades.

Tabla 1. Tipos de predictores kriging y sus propiedades

TIPO DE PREDICTOR	NOMBRE	PROPIEDADES
Lineal	<ul style="list-style-type: none"> • Simple • Ordinario • Universal 	<ul style="list-style-type: none"> • Son óptimos si hay normalidad multivariada. • Independiente de la distribución son los mejores predictores linealmente insesgados.

No Lineal	<ul style="list-style-type: none"> • Indicador • Probabilístico • Log Normal, Trans-Gaussiano • Disyuntivo 	Son predictores óptimos.
--------------	--	--------------------------

Supóngase que se tiene información sobre cierto atributo físico z en diferentes posiciones de un dominio D . Este es un problema típico que en esta situación es tratar de predecir el valor de z en aquellas coordenadas donde no hubo mediciones, teniendo en cuenta la estructura de covarianza de las variables aleatorias $Z(s)$ definidas en los sitios donde fue posible hacer mediciones. El método utilizado es muy similar a una regresión lineal múltiple aplicada pero en el contexto espacial, en donde las variables aleatorias $Z(s)$ se comportan como variables de regresión, y la variable aleatoria en el punto donde interesa la predicción, $Z(s_0)$ se comporta como la variable dependiente. Al conjunto de algoritmos de regresión lineal cuyo propósito es ese, se le conoce como kriging. Esta es una técnica de estimación local que tiene la cualidad de ser el mejor estimador lineal sin sesgo de z (Reyes, 2010).

El predictor kriging depende del modelo que se adopte para la función aleatoria $Z(s)$: Por lo general, $Z(s)$ se suele descomponerse en una componente de tendencia y una componente residual, tal como lo expresa la ecuación:

$$Z(s) = m(s) + \varepsilon(s) \quad \text{Ecuación 5}$$

Donde se supone conocido el variograma o el covariograma de $\varepsilon(s)$:

El valor esperado de Z en la posición s representa el valor de la tendencia en dicha posición:

$$E[Z(s)] = m(s) \quad \text{Ecuación 6}$$

Las variantes de kriging dependen del modelo que se adopte para la tendencia $m(s)$:

- El kriging simple (KS), supone $m(s) = m$ es decir, que la media $m(s)$ es conocida en todo el dominio D .

- El kriging ordinario (KO), supone que la tendencia $m(s) = m$ es constante pero desconocida. Además, se ciñe a fluctuaciones locales de la media dentro de una vecindad $W(s)$, dentro del cual se pueda considerar la media estacionaria.
- El kriging universal (KU), considera que la media $m(s)$ es una función que varía suavemente en todo el dominio D . La tendencia se suele modelar generalmente como una combinación lineal de funciones los cuales resultan ser combinaciones lineales de las coordenadas espaciales. Es decir, $f_i(s)$:

$$m(s) = \sum_{i=1}^K a_i f_i(s) \quad \text{Ecuación 7}$$

Donde los coeficientes a_i se desconocen. Se considera que $f_0(s) = 1$, de tal manera que cuando $K = 0$, se tiene el caso particular del kriging ordinario (Diaz, 2002).

1.12.1 Kriging ordinario (KO)

Se supone que se hacen mediciones de la variable de interés Z en los puntos $s_i = 1, \dots, n$, de la región de estudio, es decir, se tienen realizaciones de las variables de interés Z en los puntos $s_i = 1, \dots, n$, de la región de estudio, es decir, se tienen realizaciones de las variables $Z(s_1), \dots, Z(s_n)$, y se desea predecir $Z(s_0)$, en el punto s_0 donde no hubo mediciones. Bajo las consideraciones anteriores, la metodología kriging ordinario propone que el valor de la variable en el sitio no muestreado puede predecirse como una combinación lineal de la n variables aleatorias, como se muestra a continuación:

$$Z^*(s_0) = \sum_{i=1}^n \lambda_i Z(s_i) \quad \text{Ecuación 8}$$

En donde los λ_i representan los pesos o ponderaciones de los valores de las variables en los sitios muestreados. Dichos pesos se calculan en función de la distancia entre los puntos muestreados y el punto donde se va a llevar a cabo la correspondiente predicción. La suma de los pesos debe ser igual a uno para que la esperanza del predictor sea igual a la esperanza de la variable. Esto último se reconoce como el requisito de insesgamiento [Giraldo, 2009a]. Se

dice que $Z^*(s_0)$ es el mejor predictor lineal en este caso, porque los pesos se obtienen de tal manera que minimicen la varianza del error de predicción sujeto a que se cumpla el requisito de insesgamiento, es decir, que se minimice la siguiente expresión:

$$V(Z^*(s_0) - Z(s_0)) \quad \text{sujeto a} \quad \sum_{i=1}^n \lambda_i = 1 \quad \text{Ecuación 9}$$

La aplicación del método de los multiplicadores de LaGrange como técnica de optimización en conjunto con la determinación de la matriz de covarianzas a partir de la estructura de autocorrelación espacial nos permite la determinación de los pesos λ_i óptimos.

1.12.2 Varianza de predicción de Kriging ordinario (KO)

Viene dada por la siguiente ecuación:

$$\sigma_k^2 = \sigma^2 - \sum_{i=1}^n \lambda_i C_{i0} - \mu, \quad \text{Ecuación 10}$$

Siendo $\sigma^2 = V[Z(s_0)]$, $C_{i0} = \text{Cov}(Z(s_i), Z(s_0))$ y μ el multiplicador de LaGrange.

Observación: Una expresión equivalente a partir de la función de semivarianza, viene dada por:

$$\sigma_k^2 = \sigma^2 \left(1 - \sum_{i=1}^n \lambda_i \gamma_{i0} + \mu \right), \quad \text{Ecuación 11}$$

1.12.3 Validación del Kriging ordinario (KO)

Existen diversos métodos que permiten evaluar la bondad de ajuste del modelo de semivariograma elegido con respecto a los datos muestrales y por ende de las predicciones hechas con kriging. El más completo es la validación cruzada, que consiste en excluir la observación de uno de los puntos muestrales y con los $n - 1$ valores restantes y el modelo de semivariograma escogido, es predecir vía kriging el valor de la variable en estudio en la ubicación del punto que se excluyó. Se piensa que si el modelo de semivarianza elegido describe bien la estructura de autocorrelación espacial, entonces la diferencia entre el valor observado y el valor predicho debe ser pequeña. Este procedimiento se realiza en forma secuencial con cada uno de los puntos muestrales y así se obtiene un conjunto de n errores de predicción (Giraldo, 2009b).

Lo usual es calcular medidas que involucren a estos errores de predicción como por ejemplo el mínimo error cuadrático medio (MECM). Una forma descriptiva de hacer validación cruzada es mediante un gráfico de dispersión de los valores observados contra los valores predichos. En la medida en que la nube de puntos se aproxime más a una recta que pase por el origen, mejor será el modelo de semivariograma utilizado para realizar el kriging (Giraldo, 2009a).

1.12.4 Semivariogramas

Algunos ejemplos de modelos de semivariogramas isotrópicos son:

Modelo esférico.

Tiene un crecimiento rápido cerca al origen, pero los incrementos marginales van decreciendo para distancias grandes, hasta que para distancias superiores al rango los incrementos son nulos. Su expresión matemática es la siguiente:

$$\gamma(|h|) = \begin{cases} C_0 + C_1 \left[1.5 \cdot \frac{h}{a} - 0.5 \left(\frac{|h|}{a} \right)^3 \right] & \text{si } |h| \leq a \\ C_1 + C_1 & \text{si } |h| \geq a \end{cases} \quad \text{Ecuación 12}$$

En donde C_1 representa la meseta, al rango y h la distancia

Modelo Exponencial

Este modelo se aplica cuando la dependencia espacial tiene un crecimiento exponencial respecto a la distancia entre las observaciones. El valor del rango es igual a la distancia para la cual el semivariograma toma un valor igual al 95% de la meseta. Este modelo es ampliamente usado. Su expresión matemática es la siguiente:

$$\gamma(|h|) = C_1 + C_1 \cdot [1 - \exp(-\frac{3|h|}{a})] \quad \text{Ecuación 13}$$

Modelo Gaussiano

Al igual que en el modelo exponencial, la dependencia espacial se desvanece solo en una distancia que tiende a infinito. El principal distintivo de este modelo es su forma parabólica cerca al origen. Su expresión matemática es:

$$\gamma(|h|) = C_0 + C_1 \cdot [1 - \exp(-\frac{(3|h|)^2}{a^2})] \quad \text{Ecuación 14}$$

Modelo familia matérn

Esta clase se define mejor en términos de la función de covarianza definida por:

$$C_0(|h|) = \frac{1}{2^{\theta_2-1}\Gamma(\theta_2)} \left(\frac{2\sqrt{\theta_2}|h|}{\theta_1}\right)^{\theta_2} k_{\theta_2}\left(\frac{2\sqrt{\theta_2}|h|}{\theta_1}\right) \quad \text{Ecuación 15}$$

En este caso, $\theta_1 > 0$ define un parámetro de dependencia espacial y $\theta_2 > 0$ es un parámetro de forma. La función $\Gamma(\cdot)$ representa la función gamma mientras que k_{θ_2} es la función modificada de Bessel de tercera clase de orden θ_2 . Como casos especiales se tiene que $\theta_2 \rightarrow \frac{1}{2}$ tiende a la forma exponencial y el límite de $\theta_2 \rightarrow \infty$ daría lugar a la forma Gaussiana. Todos los modelos anteriores son acotados, lo que significa que el *sill* (umbral) se alcanza realmente en el límite a una cierta distancia marcada por el rango.

Además se tienen las siguientes observaciones:

- Para el modelo de efecto *nugget*, el *sill* es alcanzado tan pronto la distancia se hace positiva.
- El modelo esférico alcanza realmente el *sill* a una distancia equivalente a su rango.
- El modelo exponencial y gaussiano alcanzan asintóticamente el *sill*, definiéndose un rango práctico como aquella distancia para la que el modelo se encuentra al 95% del *sill*.

Los modelos acotados también se suelen llamar modelos de transición. Por otra parte, los modelos potencial y lineal no tienen *sill*, y por tanto, carece de su correspondiente función de covarianza.

Alrededor del origen podemos distinguir tres tipos de comportamiento:

1. Comportamiento parabólico: (modelo Gaussiano), tal comportamiento es característico de fenómenos altamente regulares.

2. Comportamiento lineal: (modelos esférico o exponencial), tal comportamiento es característico de aquellos fenómenos que muestran dependencias espaciales a cortas distancias aumentándolas linealmente con las mismas.

3. Comportamiento discontinuo: (modelo de efecto *nugget*), evidencia de gran variabilidad, intrínseca a los datos que enmascara cualquier posible dependencia espacial.

El comportamiento cerca del origen del modelo potencial depende de los valores del parámetro ω , siendo lineal para $\omega = 1$ y parabólico para valores de ω cercanos a 2.

1.12.1 Estacionariedad

La variable regionalizada es estacionaria si su función de distribución conjunta es invariante respecto a cualquier translación del vector h , o lo que es lo mismo, la función de distribución del vector aleatorio $\vec{Z}(X) = [Z(x_1), Z(x_2), \dots, Z(x_n)]^T$ es idéntica a la del vector $\vec{Z}(X) = [Z(x_1 + h), Z(x_2 + h), \dots, Z(x_n + h)]^T$ para cualquier h . La teoría geoestadística se basa en los momentos arriba descritos y la hipótesis de estacionariedad puede definirse en términos de estos:

Estacionariedad de Segundo Orden. Sea $\{Z(x) : x \in D \subset R^d\}$ una variable regionalizada definida en un dominio D contenido en R^d (generalmente una variable medida en la superficie de una región) se dice que $Z(x)$ es estacionario de segundo orden si cumple:

$$E[Z(x)] = m, \quad k \in R, \quad \forall x \in D \subset R^d \quad \text{Ecuación 16}$$

El valor esperado de la variable aleatoria es finito y constante para todo punto en el dominio.

$$COV [Z(x) , | Z(x + h)]] = C(h) < \infty \quad \text{Ecuación 17}$$

Para toda pareja $\{Z(x), Z(x + h)\}$ la covarianza existe y es función única del vector de separación h . En la Figura 2 se muestra el gráfico de una variable regionalizada estacionaria. Exceptuando fluctuaciones aleatorias, el valor promedio de la variable no muestra una tendencia definida en alguna dirección. La existencia de la covarianza implica que la varianza existe, es finita y no depende de h , es decir $\{V(Z(x)) = C(0) = \sigma^2\}$. Así mismo la estacionariedad de segundo orden implica la siguiente relación entre la función de semivarianza y la de auto covarianza:

$$\begin{aligned} \gamma(Z(x+h), Z(x)) &= \gamma(h) = \frac{1}{2} E[Z(x+h) - m - Z(x) + m] && \text{Ecuación 18} \\ &= \frac{1}{2} \sigma^2 + \frac{1}{2} \sigma^2 - E[(Z(x+h) - m)(Z(x) - m)] \\ &= \sigma^2 - C(h) \end{aligned}$$

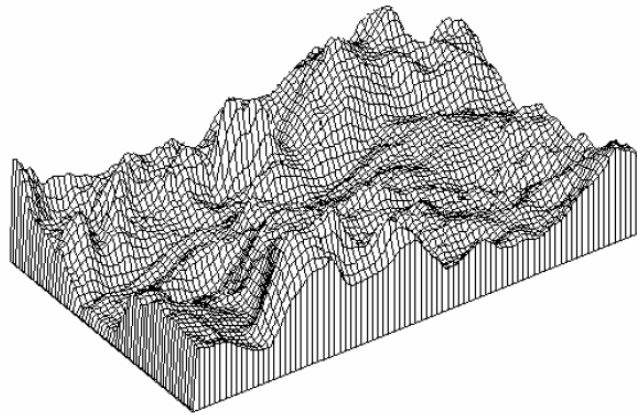


Figura 2 Representación de una superficie interpolada para una variable regionalizada estacionaria (Giraldo, 2009b)

Estacionariedad débil o intrínseca. Existen algunos fenómenos físicos reales en los que la varianza no es finita. En estos casos se trabaja sólo con la hipótesis que pide que los incrementos $[Z(x+h) - Z(x)]$ sean estacionarios, esto es (Clark, 1979):

$Z(x)$ tiene esperanza finita y constante para todo punto en el dominio. Lo que implica que la esperanza de los incrementos es cero como se muestra.

$$V[(Z(x+h) - Z(x))] = E[Z(x+h) - E[Z(x+h) - Z(x)]]^2 = 2\gamma(h) \quad \text{Ecuación 19}$$

Es claro que si una variable regionalizada es estacionaria fuerte entonces también será estacionaria débil. El concepto de estacionariedad es muy útil en la modelación de series temporales (Box & Jenkins, 1976). En este contexto es fácil la identificación, puesto que sólo hay una dirección de variación (el tiempo). En el campo espacial existen múltiples direcciones y por lo tanto se debe asumir que en todo el fenómeno es estacionario. Cuando la esperanza de la variable no es la misma en todas las direcciones o cuando la covarianza o correlación dependan del sentido en que se determinan, no habrá estacionariedad. Si la correlación entre los datos no depende de la dirección en la que esta se calcule se dice que el fenómeno es isotrópico, en caso contrario se hablará de anisotropía. En Isaaks y Srivastava (1989) se definen los posibles tipos de anisotropía y se proponen algunas soluciones. Cressie (1993) discute cual debe ser el tratamiento en caso de que la media no sea constante.

En casos prácticos resulta compleja la identificación de la estacionariedad. Suelen emplearse gráficos de dispersión de la variable respecto a las coordenadas, de medias móviles y de valores clasificados según puntos de referencia, con el propósito de identificar posibles tendencias de la variable en la región de estudio. La isotropía es estudiada a través del cálculo de funciones de auto covarianza o de semivarianza muestrales en varias direcciones. Si estas tienen formas considerablemente distintas puede no ser válido el supuesto de isotropía. Finalmente una variable regionalizada será no estacionaria si su esperanza matemática no es constante, esto es si $E[Z(x)] = m(x)$. En la Figura 3 se representa una variable regionalizada en la que existe tendencia en el valor promedio de la variable, lo cual es claro indicador de no estacionariedad (Giraldo, 2009b).

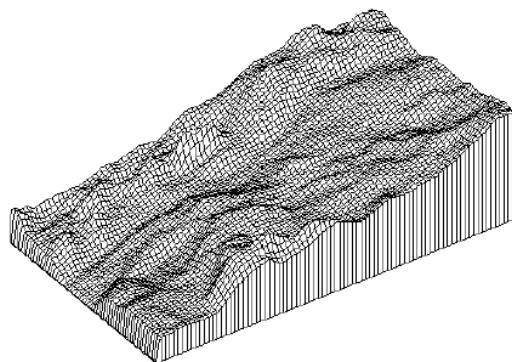


Figura 3 Representación de una superficie interpolada para una variable regionalizada no estacionaria
(Giraldo, 2009b)

1.13 Análisis de datos funcionales (ADF)

En muchas áreas de que trabajan con grandes datos como economía, ingeniería, medio ambiente las observaciones y datos existentes (bases de datos), corresponden a observaciones de variables aleatorias tomadas a lo largo de un intervalo continuo cuyo resultado de la medición corresponde a curvas. Este tipo de datos lo llamaremos datos funcionales. En el ADF la unidad básica de información es la función completa, más que un conjunto de valores (Ramsay & Dalzell 1991). Los problemas a los que se debe enfrentar la estadística con datos funcionales corresponden a la misma necesidad que la estadística clásica.

- a) Explorar y describir el conjunto de datos funcionales resaltando sus características más importantes.
- b) Explicar y modelar la relación entre una variable dependiente y una independiente (modelos de regresión).
- c) Métodos de Clasificación Supervisada o no Supervisada de un conjunto de datos respecto a alguna característica.
- d) Contraste, validación y predicción.

Definición 1. Una variable aleatoria χ se llama variable funcional si toma valores en un espacio infinito dimensional (espacio funcional). Una observación χ de \mathcal{X} se llama un dato funcional (Ferraty & Vieu, 2006).

Definición 2. Un conjunto de datos funcionales χ_1, \dots, χ_n es la observación de n variables funcionales $\mathcal{X}_1, \dots, \mathcal{X}_n$ con igual distribución que \mathcal{X} funcionales:

Donde $T = [a, b] \subset \mathbb{R}$. Normalmente se asume que se tiene elementos de:

$$L_2(T) = \{f : T \rightarrow \mathbb{R}, \text{ tal que } \int_T f(t)^2 dt < \infty\} \quad \text{Ecuación 20}$$

Donde $L_2(T)$ con el producto interno usual $\langle f, g \rangle = \int_T f(t)g(t)dt$ es un espacio euclidiano (He, Muller, & Wang, 2000).

1.14 Bases para datos funcionales

Una base es un conjunto de funciones conocidas independientes $\{\phi_k\}_{k \in \mathbb{N}}$ tales que cualquier función puede ser aproximada, tan bien como se quiera mediante una combinación lineal de K de ellas, como K suficientemente grande. De esta forma, la observación funcional puede aproximarse como:

$$x^{(q)}(t) \approx \sum_{k=1}^K c_k \phi_k^{(q)} \quad \text{Ecuación 21}$$

Básicamente la idea clave cuando se pueden usar ortogonales es representar, datos funcionales en la base usando aquellas coordenadas que son más significativas, debido a la alta dimensión de los datos funcionales, se elige en general un número K para representar los datos en el sub-espacio, convirtiendo el problema de dimensión infinita a un problema multidimensional. La elección del parámetro K y de la base más adecuada para los datos observados se muestra crucial y en principio no hay regla que permita hacer selección óptima de forma universal. El parámetro K es de cierto modo un parámetro de suavización de los datos funcionales. Si K es bajo tendremos un modelo muy manejable pero posiblemente habremos perdido información relevante. Si K es alto representaremos muy bien los datos pero el problema de la dimensión cobra importancia. Si atendemos a la elección de la base, para datos periódicos se suele emplear la base de Fourier y para los datos no periódicos la base b-spline o la wavelets. Una base muy popular está basada en la expansión Karhunen-Loève que no es más que la extensión del análisis de componentes principales multivariante a procesos estocásticos y por añadidura a datos funcionales (J. O. Ramsay & Silverman, 2005).

MATERIALES Y MÉTODOS

1.15 Bases de datos climáticos

Los datos obtenidos para el presente proyecto, son registros de la red sistematizada de datos climáticos del Centro de Investigación de la Caña de Azúcar CENICAÑA, teniendo como base el acuerdo suscrito entre el Centro Internacional de Agricultura Tropical (CIAT) y CENICAÑA, que a la fecha está vigente. La red está distribuida en 34 estaciones de las cuales solo se usaron 26, esta red cuenta con datos de estaciones desde el año 1993, estos fueron descargados desde su base de datos central, igualmente se pudo acceder a las estaciones automatizadas que el CIAT tiene en Palmira (Valle del Cauca), y Santander de Quilichao (departamento del Cauca).

1.16 Equipos

Este trabajo fue llevado a cabo en su totalidad en un computador de arquitectura Intel PC, de doble Núcleo de 2 GHz Con memoria de 8GB, disco duro de 500GB marca LENOVO Sistema Operativo Windows 7 X64 Bits.

1.17 Programas

El trabajo de procesamiento estadístico fue realizado con el software estadístico R (Versión 2.15) de dominio público y código abierto y su complemento *RStudio*, adicional a este fueron instaladas las librerías *fda*, *fda.usc* y *geofd*, las cuales fueron empleadas para el tratamiento de los datos funcionales y *raster*. Para el geo-procesamiento y rutinas de transformación de datos, fue usando el software ArcGIS Desktop de ESRI (Versión 10.1), junto con las licencias de *Arcmap*, *ArcCatalogue*, *ArcGIS 10 Spatial Analysis* y *Geostatistical Analysis*.

1.18 Descripción del proceso

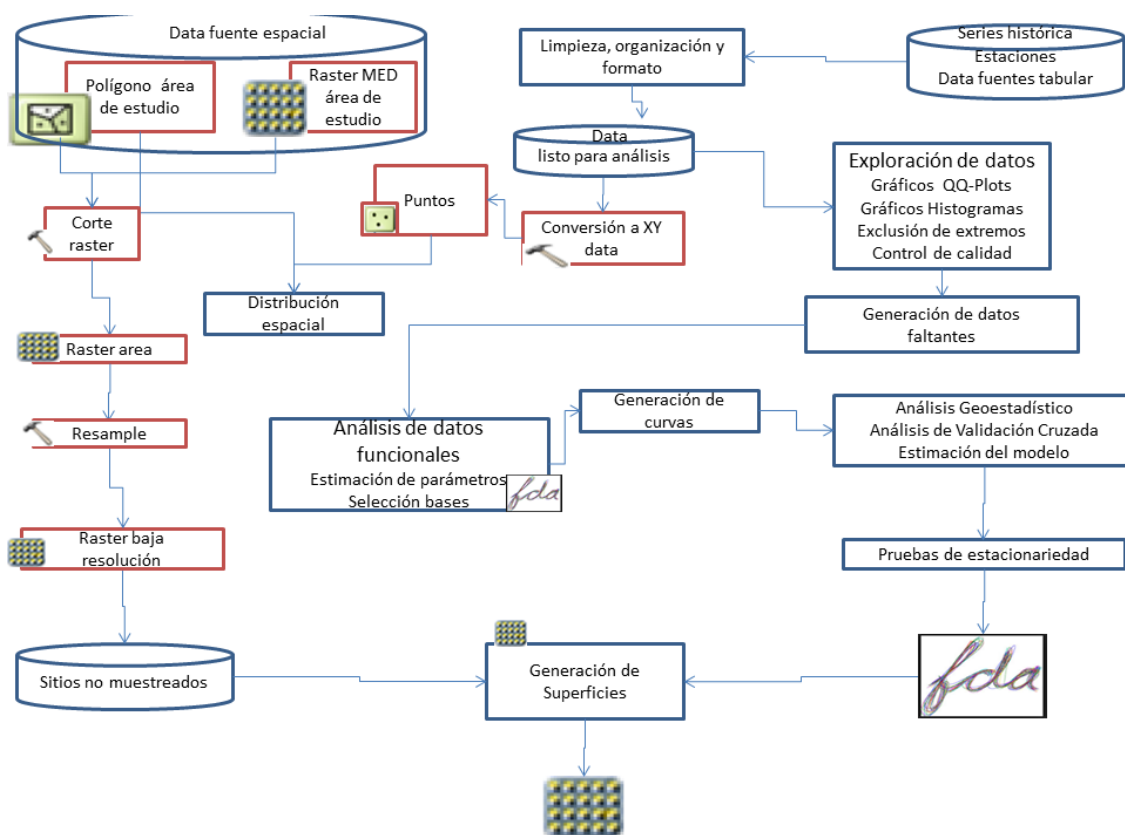


Figura 4. Diagrama de flujo, del proceso metodológico seguido, adicionalmente en las siguientes secciones se detalla el procedimiento seguido. En este esquema en recuadros azules se muestra los procesos desarrollados en ArcGis, y en rojo cada una de las herramientas utilizadas.

El diagrama de flujo permite representar, el análisis realizado, el cual parte de obtener los datos de las series climáticas de las estaciones meteorológicas, accediendo a las bases de datos de registros históricos del CIAT y CENICANA, posteriormente estos datos fueron limpiados y organizados, para poder ser procesados por el software estadístico R. En el proceso de limpieza se eliminó los campos restantes aquellos que no participarían del análisis, solo se utilizaron las variables de temperatura (C°) máxima y mínima y precipitación (mm). Posteriormente se procedió a identificar diferencias climáticas e identificar las características topográficas de la zona de estudio, a partir de mapas e información del Valle del Cauca y exhaustiva consulta de literatura detallada acerca del clima predominante; con el propósito de delimitar el área de estudio (Figura 4).

Para identificar la distribución espacial de las estaciones climáticas en la zona de estudio, se usaron las coordenadas geográficas obtenidas por el convenio (Latitud y

Longitud), las cuales mediante el software ArcGis Desktop y Arcmap (Versión 10.1), se convirtieron en un formato de puntos (*shape points.shp*) representadas junto con el polígono administrativo de la zona.

El siguiente paso fue el análisis descriptivo de las series climáticas que se realizó para identificar picos máximos y mínimos y determinar si estas eran homogéneas, incluyó una etapa de análisis simple y visual de gráficos que fueron necesarias (gráficos de series, diagramas de cajas, histogramas) y análisis estadístico básicos, todo usando el software estadístico R, aplicando métodos para detectar datos anómalos, discontinuidad y por último completar las series generando datos faltantes, para ello se usó la extensión *rmapgen* (Cordano & Eccel, 2013), por último la fase de modelización espacial está compuesta de dos etapas, la primera en la cual se detectan los parámetros de las funciones y una segunda los parámetros de la interpolación. La primera etapa tiene enfoque no paramétrico basado en la expansión en término de bases de funciones que se usa para obtener las curvas a partir de las series, también llamados datos discretos. La segunda es una etapa previa de validación a *posteriori* de resultados, basada en la estimación de valores conocidos, bajo condiciones del modelo construido, dicha validación incluye el modelo del variograma y el tipo de Kriging, se aplica Validación Cruzada (CV) para evaluar la bondad de ajuste del modelo del semivariograma elegido, con respecto a los datos muestrales para las predicciones con Kriging, esto se hizo con el propósito de comparar valores estimados por el modelo con valores reales, ya que es un proceso iterativo donde cada vez se excluye datos de la muestra y se estima con el resto de los datos usando el modelo de semivariograma, la predicción kriging del valor de la variable escogida en la ubicación del punto que se excluyó, teóricamente cada uno de estos valores se compara, por medio de regresión lineal, con el valor real. El concepto es sencillo buenos coeficientes de correlación /determinación fueron indicativos de una correcta modelización. Ya que el modelo de semivarianza elegido definió bien la estructura de autocorrelación espacial, donde la diferencia entre valor observado y el valor predicho debe ser mínima, este proceso se realiza con cada una de las muestras para este caso estaciones climáticas, obteniendo un conjunto de errores de predicción. Se obtuvo para el modelo el mínimo error cuadrático medio (MECM), la forma descriptiva de hacer la validación cruzada fue mediante un gráfico de dispersión de los valores observados contra los

valores predichos, en la medida que la nube de puntos se ajuste a una línea recta que pase por el origen, mejor será el modelo del semivariograma utilizado para realizar el kriging.

Después de ser hallados los parámetros del modelo se procedió a la generación de las superficies utilizando las librerías, *fda* (*functional data analysis*) (Ramsay y Silverman 2009), *geofd* (Giraldo, Ramon, Mateu y Delicado 2012).

RESULTADOS Y DISCUSIÓN

1.19 Localización del área de estudio

El entorno del área de estudio es el Departamento del Valle del Cauca está localizado al sur occidente de Colombia $3^{\circ} 05'$ y $5^{\circ} 01'$ de latitud norte, y los $75^{\circ} 42'$ y $77^{\circ} 33'$ de longitud oeste, América del Sur, el Valle Del Cauca es uno de los 32 departamentos que componen el país, limita al occidente con el Océano Pacífico y al oriente con la región Andina, gran parte de la región está ubicado entre la cordillera central y la oriental es el valle geográfico del río cauca de donde proviene su nombre, limita al norte con Chocó y Risaralda, al sur con Cauca, al este con Quindío y Tolima, y al oeste con el Océano pacífico.

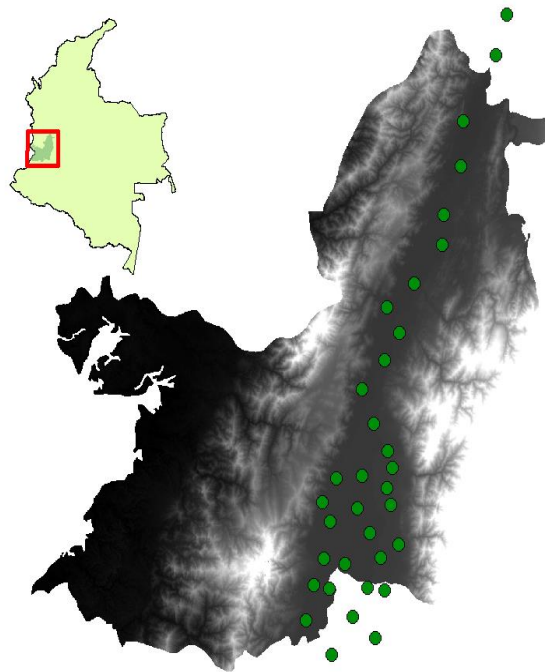


Figura 5. Ubicación área de estudio

1.20 Distribución de pisos térmicos

Cálido: 0-1000 msnm 47,9%

Templado: 1000-2000 msnm 33,6%

Frío: 2000-3000 msnm 13,6%

Páramo: 3000 y más 4,9%

Valle físico del río Cauca está ubicado entre las cordilleras Central y Occidental de los Andes colombianos en el departamento del Valle del Cauca, es una región plana homogénea. Tiene aproximadamente 240 km de largo y su ancho varía entre 32 km (Yumbo y Palmira) y 12 km (Yotoco, Guadalajara de Buga, La Victoria). Se encuentra localizado a una altura de 1.000 msnm en promedio entre los 75° 50' - 76° 39' Longitud Oeste y -5° 7' Latitud Norte y abarca una superficie aproximada de 3.000 km², en su gran mayoría dedicadas al cultivo de la caña de azúcar.

1.21 Caracterización del clima

Debido a que su ubicación geográfica (zona ecuatorial) donde influyen notoriamente en la circulación las corrientes de aire cálido y húmedo, lo que significa que se encuentra bajo la influencia los vientos Alisios del Noreste y sureste que convergen en la Zona de Confluencia Intertropical (ZCIT) favorece el desarrollo de nubosidad y lluvias, de clima intertropical, donde la temperatura promedio de la región fluctúa entre los 19 y 32 °C, la humedad relativa fluctúa en el rango 65%-75%. Es una región cuyo régimen de distribución de precipitación es bimodal. Este comportamiento de la precipitación es el resultado de la actividad de la ZCIT esta se localiza en el sur durante los meses de enero y febrero, comienza su desplazamiento hacia el norte a mediados de marzo, concluyendo su recorrido a mediados del año. Durante el mes de septiembre inicia el desplazamiento hacia el sur el cual termina en el mes de enero (figura 6).

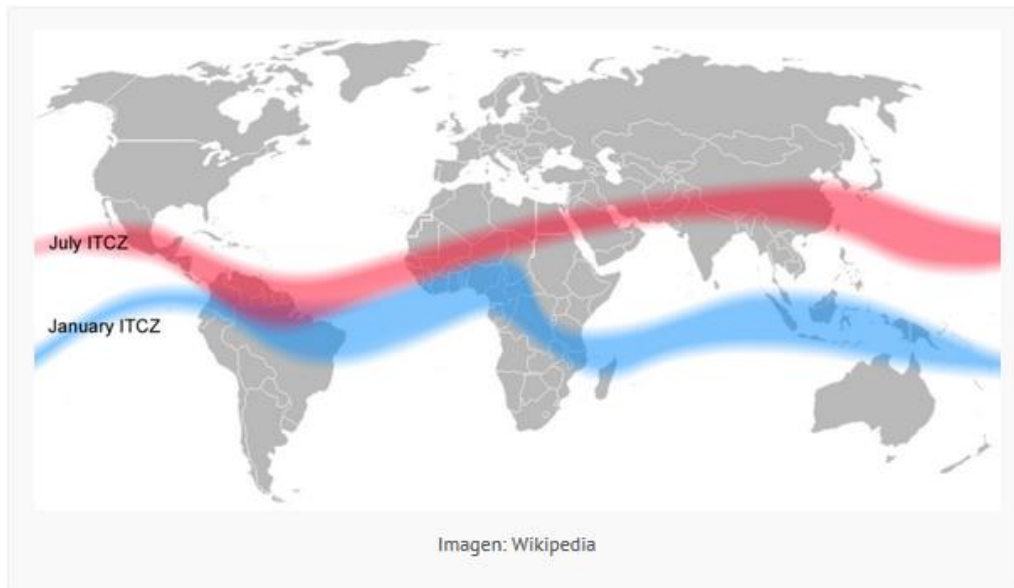


Figura 6. Zona de Confluencia Intertropical (ZCIT) www.wikipedia.com

1.22 Temperatura

Las temperaturas medias están cercanas a **24°C**, la temperatura mínima media (asociada a la temperatura de la noche) de **19°C** y temperatura máxima media (asociada a temperaturas en el día) de 30 °C que corresponde al piso térmico cálido; no obstante el registro histórico muestra una temperatura en Cali máxima de **37.5°C** y una temperatura mínima de **13°C** (Enrique Cortes y Barrios 2010) .

1.23 Precipitación

El régimen de distribución de precipitación es bimodal con dos estaciones lluviosas y dos secas al año; los periodos secos corresponde a diciembre - enero - febrero y la segunda junio - julio - agosto y dos periodos lluviosos la primera de marzo - abril - mayo y la segunda de septiembre - octubre - noviembre. En la parte plana del valle del rio cauca las lluvias, en general son inferiores a los 1,800 mm anuales. Las temporadas lluviosas en la región están relacionadas con el paso del ZCIT sobre la misma: abril-mayo, cuando va migrando hacia el norte y octubre-noviembre cuando va migrando hacia el sur. De igual forma para las temporadas secas los índices de precipitación anual son: 1,589 mm en el norte (133 días de lluvias), 1,882 mm al sur (109 días de lluvias) y 938 mm en el centro (100 días de lluvias) (Enrique Cortes y Barrios 2010).

1.24 Estadística descriptiva de los datos de temperatura mínima

Usando el programa estadístico R se llevó a cabo el análisis preliminar de las series, por medio de representaciones gráficas y parámetros estadísticos, elaborando una inspección inicial de los datos con el fin de hallar valores atípicos y así obtener sus descripciones y comprobar supuestos. Esta inspección nos mostró al validar la existencia de valores inusuales, extremos, discontinuidades, entre otros y permitió tomar decisiones respecto y saber si son adecuados para aplicar las técnicas requeridas a dichas series. A continuación se presenta los resultados de esta etapa.

Número de casos: Del total de 39 estaciones seleccionadas para el estudio fueron descartadas por datos faltantes nueve estaciones y por datos incompletos dos con un total de 11 estaciones descartadas para el análisis quedando solamente 28 estaciones disponibles para el análisis. La Figura 5, muestra que cerca del 75% de las estaciones, las series incluyen datos

desde 1997 ya que el periodo de estudio establecido fue un periodo de 16 años (desde 1997 al 2012), se incluye el año inicial.

En las tablas resumen (Tabla 5, Tabla 6, Tabla 7) se muestran el número total de casos y cada una de las estadísticas descriptivas correspondientes a cada una de las variables organizadas por estación y que fueron incluidas en el estudio, en estas el campo NA y NA% corresponde a la cantidad de datos faltantes y porcentaje de datos faltantes por estación correspondientemente, de lo cual se puede resumir que las series tienen un porcentaje de datos faltantes bajo el cual corresponde al 5.9% en la variable temperatura máxima, 5.9 en la variable temperatura mínima y 5.6 en precipitación; el porcentaje mayor de datos faltantes corresponde a la estación ptalcali con 12%.



Figura 7 Distribución de registros por estación

1.25 Análisis Gráfico

Este análisis corresponde al realizado a las estaciones climáticas de datos de temperatura máxima, temperatura mínima y precipitación en el Valle del Cauca, mediante una serie de histogramas, como se puede notar en las imágenes del ANEXO2, las secuencias de las series de precipitación todas tienen un comportamiento gráfico típico, el cual corresponden a una distribución unimodal asimétrica positiva, teóricamente este comportamiento corresponde a que el valor del coeficiente de asimetría es mayor a cero, y esto corresponde a la distribución esperada de las precipitaciones ya que la mayor parte del año no se presentan lluvias, también se puede notar que ésta tendencia es más marcada en

unas estaciones que en otras, este comportamiento se afirma dado los valores del coeficiente de asimetría son altos en la tablas de resumen del ANEXO 1.

Igualmente en las imágenes del ANEXO2, la secuencia de series de temperatura máxima y mínima tiene un comportamiento gráfico típico, el cual corresponde a una distribución normal.

Diagrama de Cajas: En el anexo 2 correspondiente a los diagramas de cajas, se determina la existencia de valores extremos en las series de temperatura, así también la concentración de ellas desde el rango de 25 - 35 C° en máxima, en mínima desde 16 a 22 C°, y de precipitación entre 0 y 10 o 15 mm.

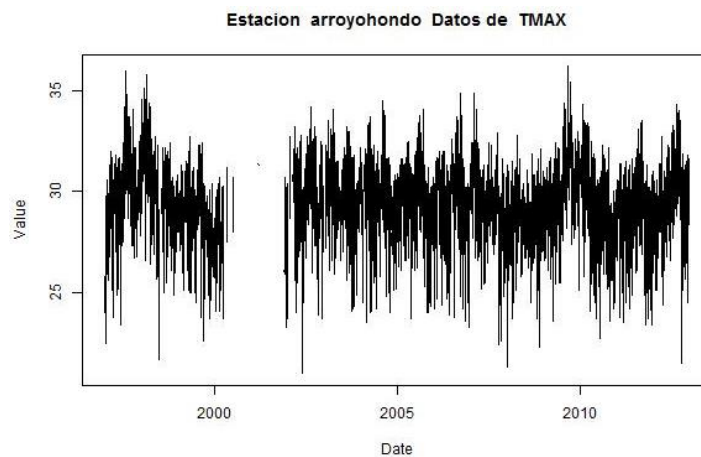


Figura 8 Ejemplo de datos faltante en las series

1.26 Análisis descriptivo

En las tablas Tabla 5, Tabla 6, Tabla 7, que corresponden a tablas de estadísticos de resumen, se organizaron los principales parámetros los cuales permiten realizar el análisis se las series de datos estadísticos: mínimo (Min), máximo (Max), media (Media), varianza (Variance), desviación estándar (Stand Desv.), coeficiente de variación (CV), datos NA (NA), porcentaje NA (NA %). Se observa que con respecto a las medidas de localización, la media y la mediana que tienen valores muy similares la media oscila entre los rangos de 18.361 hasta 19.937 y la mediana entre 18.4 - 19.9 dejando prever un buen ajuste a un modelo de

distribución normal, teniendo en cuenta que, la normalidad para nuestro modelo no se tendrá en cuenta.

Las medidas de Dispersión: En lo referente a la varianza y la desviación estándar de las series de datos de las estaciones, se puede mencionar que no existe varianzas muestrales grandes que estas van desde 37.48 mm a 122.40 mm y las desviaciones estándar 6.12 a 11.06 se puede decir que no hay diferencias muy grandes en la lluvia en el Valle del río Cauca, lo que hace que las series de datos tengan un comportamiento homogéneo.

Coefficiente de Variación: Se pudo observar que los coeficientes de variación de las variables TMAX entre 6.02 y 7.64 y TMIN 4.72 y 20.85 afirmando que son relativamente bajos, esto quiere decir que la distribución de los datos es homogéneo, los coeficientes de variación de la variable PREC que están entre 180.31 y 275.80, los valores altos corresponde a la parte baja del Valle, esto quiere decir que la distribución de los datos no es homogénea para estas estaciones.

Datos extremos: En el proceso de hallar los valores extremos, las estadísticas básicas que se muestran en las tablas resumen están registrados los valores extremos máximos y mínimos que están presentes en cada una de las series evaluadas, hay que tener en cuenta que algunos de ellos fueron clasificados como fuera de rango y fueron eliminados posteriormente, en el caso de los valores máximos y mínimos se puede afirmar que coinciden con los meses típicos del año, con el propósito de refinar el procedimiento de detección de atípicos se hace una selección de todos los valores fuera del rango de cuatro DEV (desviaciones estándar).

Cálculo de faltantes y llenado de datos: Partiendo de un formato desagregado por variable, en el cual todas las estaciones involucradas en el análisis se tabularon y organizaron por fecha, se marcaron los faltantes con NA, formato que reconoce el software estadístico R para los valores nulos, la información se agrupó de acuerdo a la variable a analizar por archivo y se le dio la extensión csv (que corresponde a un archivo separado por comas), se generaron varios script en el software estadístico R para esta etapa, para ello se utilizó la librería denominada rmaxgen (Cordano & Eccel, 2013), con la cual llevó a cabo el proceso

de llenado de faltantes, para lo cual se debe proveer a la herramienta de las variable temperatura máxima, temperatura mínima y precipitación, la herramienta se basa en los modelos VAR y la información de las estaciones cercanas, dado la homogeneidad encontrada en los procesos previos se decide usar dicha herramienta, teniendo en cuenta la homogeneidad de la información.

1.27 Control de calidad y generación de datos faltantes en las series diarias

A continuación se muestra procedimiento que se siguió para la generación de los datos faltantes en las series diarias, el cual comprende procesos adicionales de control de calidad, con el fin de lograr la generación completa de datos, cabe señalar que este paquete fue seleccionado gracias a que permite automatizar el proceso. Con el fin de llenar datos nulos se ha utilizado desde el programa estadístico R la librería *rmawgen* (Cordano & Eccel, 2013), este paquete implementa herramientas que utilizan modelos (VAR) como se explicó anteriormente. El proceso comprende una primera de instalación de la librería o herramienta, y una segunda que es activarla al inicio de cada script previo a su uso, la siguiente es la descripción de los scripts que describen las tareas llevadas a cabo para lograr la tarea, los script inician con 04 que corresponde a la etapa cuatro de los procesos realizados.

Script “04_01 QCDiaria.R”

La primera tarea es organizar los datos de acuerdo al formato que entiende el paquete *RMAWGEN*, se ordena el formato de las fechas y crea el archivo de extensión *csv* correspondiente de cada una de las variables, asignando a cada una un archivo correspondiente.

Para que la librería trabaje mejor se crea un pequeño módulo que inicia con un proceso de control de calidad, la primera tarea dentro de este proceso es definir que los valores los cuales se encuentran a cuatro desviaciones estándar (desviación típica) por fuera del promedio total sean marcadas, esto ayuda a que en un siguiente paso se pueda contar con esta selección y poder evaluar si los casos son por fallo del instrumento de captura de datos, o por lo contrario corresponden a algún factor diferente.

Se genera un reporte el cual contiene los valores atípicos, duplicados y perdidos, Dejando lista la estructura la base de datos, permitiendo separar cada variable en un archivo diferente.

Script “04_02Coher_TmaxTmin.R”

Toma las variables climáticas temperatura máxima y mínima de una misma estación de nivel diario y hace una comparación de fecha a fecha de las variables chequeando coherencia, es decir que las temperaturas máximas efectivamente sean mayores a las temperaturas mínimas, en ocasiones temperatura mínima es mayor a la máxima lo cual es obviamente un caso de error, el proceso elimina los casos para dichas fechas donde se presente la anomalía. Si resulta el caso que existió un error genera una nueva serie con el siguiente nombre generando archivos de texto con la siguiente nomenclatura id_tmax2.txt y id_tmin2.txt, también genera unos archivos de Excel id unificado, en donde está la estación con temperaturas máxima y mínimas, ordenado por fecha para que se pueda verificar que efectivamente hubo un error o no en algún momento, otras tareas que corresponden a este script es detecta y estandariza el formato de la fecha, para lograrse utiliza otras estaciones para comparar y arreglar problemas de la fecha. Además para convertir a escala diaria en caso de existir estaciones horarias.

Script “04_04_03CreandoInputsRmaw.R”

Los scripts anteriores ordenan los datos pero las salidas generalmente quedan en archivos separados. En general se hace un proceso en que se hace coincidir las fechas de todos los archivos para que todos queden en uno solo formato estándar y se pueda procesar con WMAWGEN; lo mismo con las etiquetas, que sirven para otras partes del procesamiento.

Script 04_04generarDatosVic.R

Se realiza el proceso de datos faltantes, en el que se selecciona las estaciones que van a ser usadas para análisis y luego se inicia el proceso primero estimando las temperaturas, seguido por las precipitaciones en la que usa las temperaturas como variables exógenas. Al final se realiza solo un proceso de organización de los datos.

1.28 Predicción espacial Geoestadística para datos funcionales

La predicción espacial por medio de métodos geoestadísticos se llevó a cabo, tomando como base las curvas producidas con el conjunto de datos obtenidos por las estaciones meteorológicas, para cada uno de los tres eventos de estudio, se encontró que, el comportamiento observado en los valores de las series, pertenecen a una función continua x a lo largo de un periodo de tiempo $x(t)$ 16 años, que corresponde a 5.846 registros diarios, en registros de temperaturas y precipitación, el proceso continuo es registrado de manera discreta y transformado a intervalos diarios, para cada día y noche, la temperatura del día es registrada como temperatura máxima y la nocturna es registrada como temperatura mínima y las unidades son grados centígrados, para la precipitación el registro diario está dado en acumulados diarios y sus unidades son milímetros, aunque es bien conocido que la precipitación tiene una connotación diferente a la temperaturas, cabe resaltar que en este análisis le daremos una “tratamiento funcional igual”, con el fin de estandarizar el método y aplicarlo así a todas las variables, los datos anteriormente fueron sometidos a un proceso de gráficos, análisis, depuración y llenado de series, que fueron anteriormente descritos, seguido se aplica a las tres variables un proceso de resumen, como se muestra en la Figura 9, la parte izquierda corresponde a temperatura mínima, para el caso de precipitación lo muestra la misma figura al lado derecho. Terminado el proceso de resumen se inicia la creación y afinamiento de curvas, donde se propone aplicar un ajuste no paramétrico a las funciones de las series de datos (curvas) y usar bases de Fourier o spline para dicha suavización. Esta etapa de pre-proceso se hace para transformar los datos a continuos, es decir, representándolos en una base para cada una de las variables. Usando para ello las librerías fda (J. O. Ramsay & Silverman, 2005) y geofd (Giraldo, Ramon, Mateu & Delicado, 2012), aplicando la metodología propuesta por Giraldo y Mateu, se procede a dar explicación al proceso propuesto.

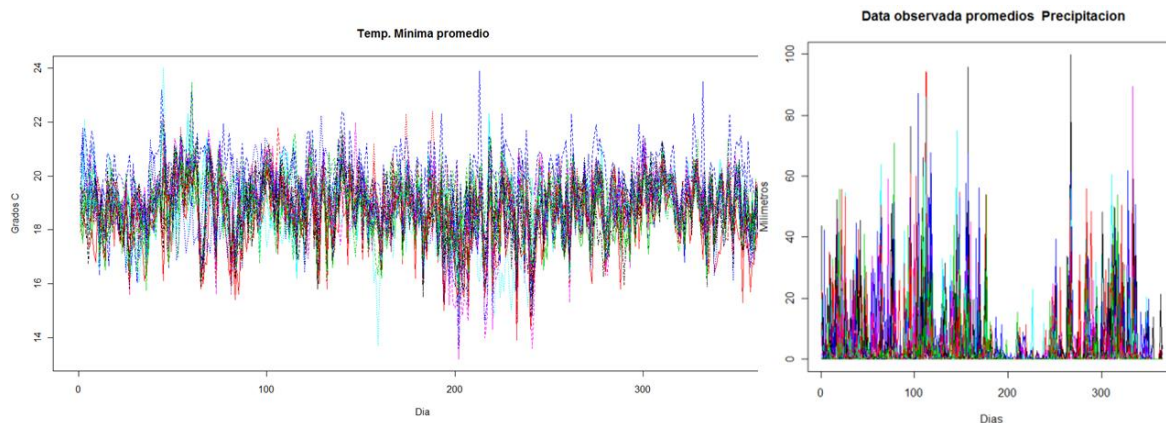


Figura 9. Gráficos de las series resumen de temperatura y precipitación

La primera etapa del análisis comprendió el resumir las series multianuales, por tanto se aplicó a todas las variables un proceso de resumen, para el caso de las temperaturas consistió en convertir los 16 años representados en 5.846 registros de series de datos diarios, en promedios diarios de un año (365 días), cada fecha correspondiente de la dupla mes/día en todos los años de las series, se aplicó la función de promedio y el resultado se llevó a una tabla nueva de resumen mes/día donde se separaron las variables por archivos independientes, en análisis previos se concluyó que la situación es diferente para la precipitación, debido a la cantidad de datos con cero, hallando que debido a la cantidad de días sin lluvia (registros en cero), en el proceso de suavización se produjo muchas curvas negativas, por lo tanto el proceso de resumen que se aplicó a esta variable fue el siguiente, adicional al proceso de promedio se sumó a las series el valor de 0.5 y luego se elevó a raíz cuadrada, este proceso fue realizado en hoja de cálculo, y su utilidad se justificó con el fin de elaborar los análisis y minimizar obtener resultados negativos, después de realizados los análisis las cifras resultante se deben invertir, elevar al cuadrado toda la serie y restar 0.5.

La metodología seleccionada para la interpolación geostatística de los datos funcionales fue Kriging Ordinario para datos valor-función (*function-value data*), esta metodología considera los métodos de suavización spline y Fourier. Las librerías *fda* y *geofd* cubren todo el proceso, iniciando con la estimación y generación de curvas, estimación del variograma y finalizando en la predicción (Giraldo, Ramon, Mateu & Delicado, 2012). El primer paso en el proceso es la instalación de las librerías, desde el programa R, *fda* (J. O. Ramsay & Silverman, 2005), *geofd* (Giraldo, Ramon, Mateu & Delicado, 2012) teniendo en

cuenta que ellas a su vez instalan las librerías necesarias precedentes, requeridas para correr de manera exitosa. En esta etapa son dos los factores claves a tener en cuenta para lograr los objetivos estos son:

- Definir el método de bases funcional que mejor se ajuste y represente los datos (esto se refiere a las curvas) observados.
- Determinar el tipo variograma o covarianza experimental para llevar a cabo la predicción.

Estos dos procesos nos permitirán crear el modelo adecuado que poseerá la mayoría de las características para cada uno de los fenómenos estudiados, en nuestro caso temperatura máxima y mínima y precipitación. En la parte funcional cuando es aplicada al conjunto datos, la primera preocupación es siempre en que espacio de bases funcional se trabaja, esto determina decisivamente el conjunto de herramientas que se podrá usar. La métrica del espacio funcional que se elija para sincronizar estos datos debe ser coherente con la interpretación física del fenómeno estudiado. Ya que los datos corresponden a grupo de series repetitivas, por lo cual se usarán bases de Fourier para llevar a cabo el proceso de suavización, los diferentes parámetros probados para estas base fueron 30, 65 y 95 bases de Fourier, de los cuales el mejor ajuste se dio en la base de 65, por lo tanto se decidió conservarlo para el estudio de los datos colectados, la Figura 10 muestra la diferentes bases probadas en la fase de ajuste.

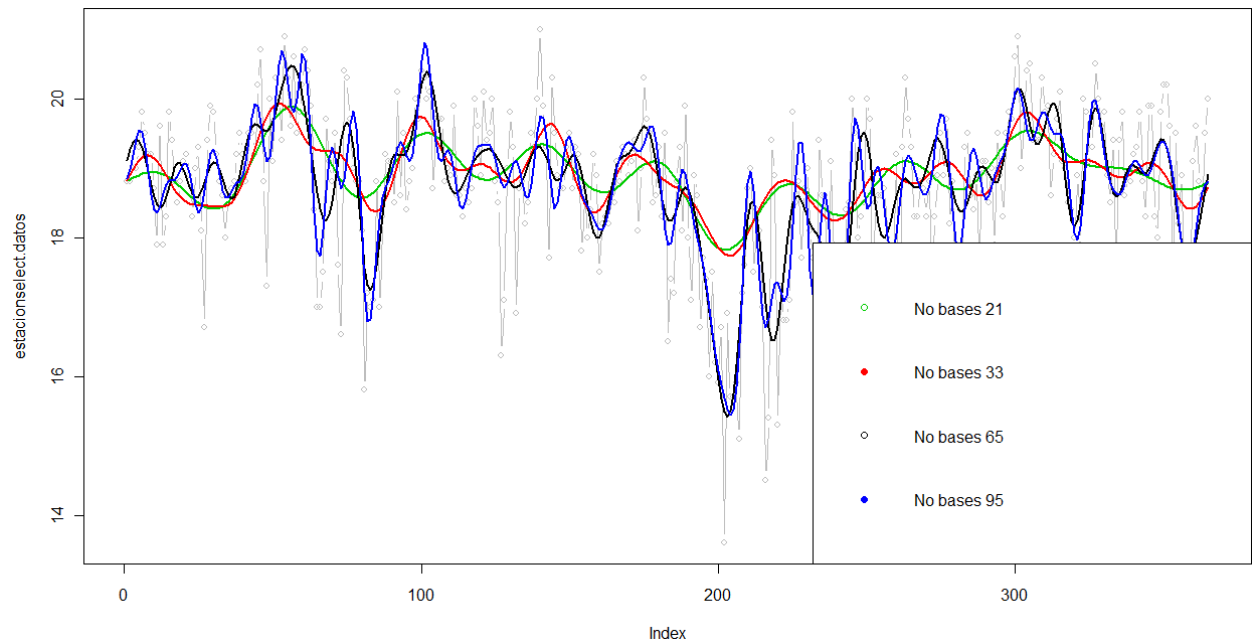


Figura 10. Representación de las bases evaluadas.

Para la suavización de las series de temperatura se usaron los parámetros: $\lambda=0$, número de bases=65, días año =365 (Giraldo, Ramon, Mateu & Delicado, 2012).

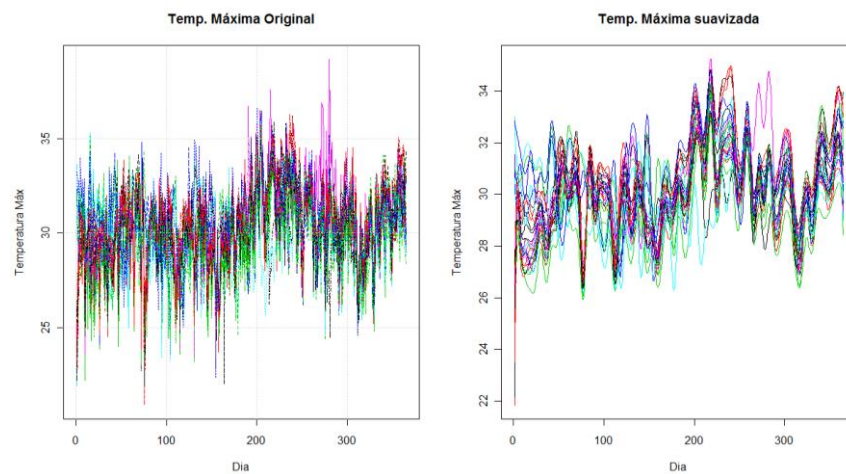


Figura 11. Comparación temperatura normal y suavizada b-spline.

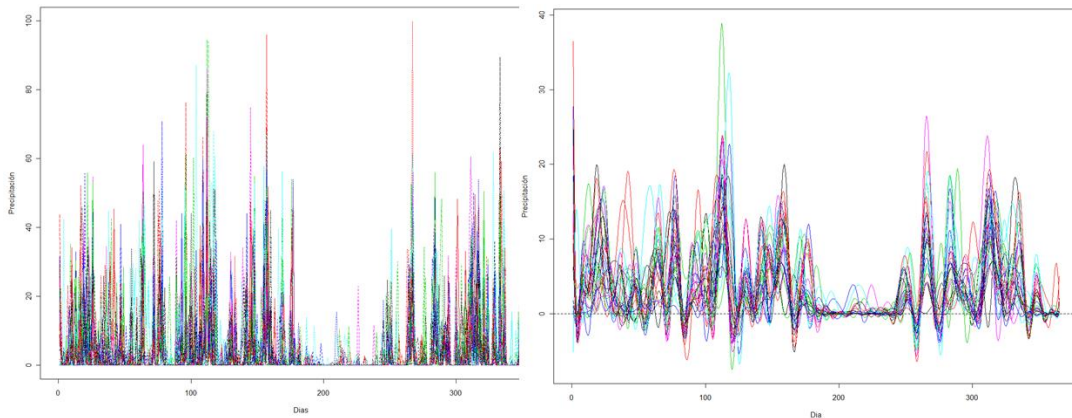


Figura 12. Comparación de precipitación promedio y suavizada.

Algunas decisiones importantes en la modelización se centran en el número y tipo de estructuras bases que constituyen el variograma y cómo obtener el modelo isotrópico o anisotrópico, al analizar los posibles variogramas permisibles de acuerdo a su comportamiento, la elección más acertada de los parámetros asociados a los variogramas (meseta, rango, pepita) son lo más importante; en la práctica se suele comparar diferentes variogramas calculados en distintas direcciones, para decidir si existe anisotropía geométrica, por lo menos tres direcciones deben ser consideradas.

En los problemas de modelación, el procedimiento ideal de evaluación consiste en dividir todo el conjunto de datos en dos partes y usar una para entrenamiento y la otra para pruebas. La idea detrás de este es la validación cruzada, donde los papeles se cambian y pasan de muestra a prueba y lo contrario, en el contexto de fda espacialmente correlacionados, donde el objetivo es predecir las curvas en determinadas localizaciones no muestreadas. La técnica de validación cruzada dejando unas estaciones en nuestro caso por fuera, se aplica de la siguiente manera: del conjunto de datos se extraen algunas estaciones y se aplica una función de suavización y se predice en dicha localización usando para ello kriging funcional como el predictor, a este proceso se le da el nombre de validación cruzada. Por último se calcula la suma de los cuadrados del error (SSE, por sus siglas en inglés) del FDA comparando los datos reales con los datos modelados. Para predecir el modelo de

semivariograma se probaron mediante el método de validación cruzada cuatro ejemplos de semivariogramas isotrópicos: matérn, exponencial, esférico y circular.

Tabla 2. Principales tipos de kriging lineal y sus propiedades evaluados en la estación Aereopuerto.

Estadístico SSE	Temperatura Máxima			Temperatura Mínima			Precipitación		
	Expo.	Esfér.	Mater.	Expo.	Esfér.	Mater.	Expo.	Esfer.	Mater.
Min. :	707.1	706.8	707.1	232.5	249	232.5	335.2	336.8	335.2
1st Qu.:	869.2	872.7	869.2	309.6	295	309.6	466.3	468.7	466.3
Median :	933.6	931.8	933.6	335.8	327.8	335.8	501.1	505.1	501.1
Mean :	1056.8	1059.3	1056.8	373.5	366.4	373.5	535.8	538.5	535.8
3rd Qu.:	1242.9	1243.2	1242.9	389.1	368.5	389.1	568.2	575.7	568.2
Max. :	1659.4	1655.2	1659.4	965.5	965.6	965.5	1088.1	1101.6	1088.1
Min. :	706.3	705.4	706.3	232.5	248.1	232.5	305.1	305.7	305.1

Tabla 3. Resumen de evaluación del desempeño de los modelos SSE

SSE	Exponencial	Esférico	Matérn
Tmax	1039.4	1042.2	1039.4
Tmin	1039.0	1041.7	1039.0
Prec	1038.4	1041.1	1038.4

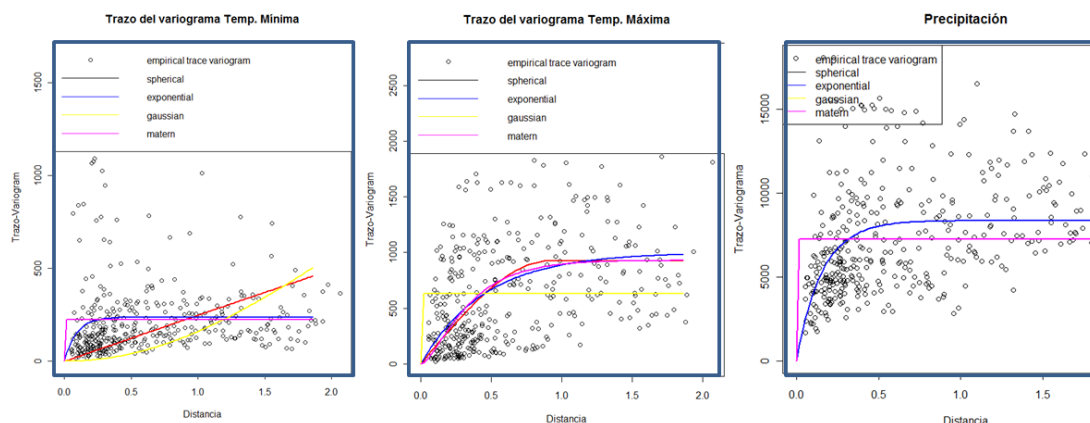


Figura 13. Comparación de los modelos Exponencial, Esférico, Gaussiano y Matérn

Se evaluó las variables de temperatura máxima y mínima y precipitación de forma independiente, se asumió que el proceso aleatorio funcional es estacionario de segundo orden e isotrópico, significa que la media de la función de varianza es constante y la covarianza depende solo de la distancia entre los puntos muestreados; para decidir qué modelo de semivariograma usar en la predicción geostatística por KO. Se probó mediante VC cuatro

tipos de semivariogramas isotrópicos (matérn, exponencial, esférico y circular) anteriormente descritos Capítulo 1. En la Figura 13 se aprecian el resultado de la predicción obtenidos mediante VC para elegir el modelo más apropiado. En la Tabla 2 se muestran los estadísticos en la estación 1 corresponde a la estación de aeropuerto, en la Tabla 3 muestra la los promedios resumen GCV de todas las estaciones, se concluye que los modelos exponencial, y matérn son similares y tienen un comportamiento similar en cada una las 3 variables evaluadas independientemente, el modelo esférico es el menos apropiado y se encontró que el modelo gaussiano no aplicó pues no presentó resultados en las evaluaciones de las temperaturas. Por tanto se trabajará con el exponencial que aparece en color azul. Suponemos que no se aprecia gran diferencia con el modelo matérn debido a que las distancias entre las estaciones no es muy grande, la máxima distancia de una estación a otra son 539.37 Km.

En la Figura 14 se muestra el variograma obtenido al calcular la predicción en el punto no muestreado con el método exponencial, como se explicó, con el modelo matérn los resultado fueron análogos.

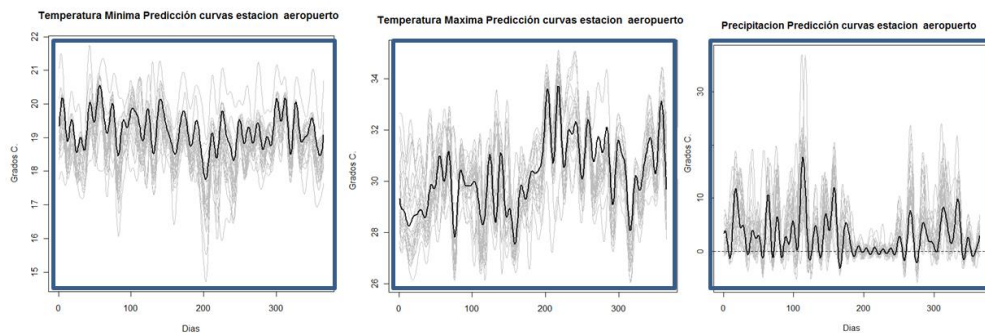


Figura 14 Validación Cruzada General de Modelo estación aeropuerto.

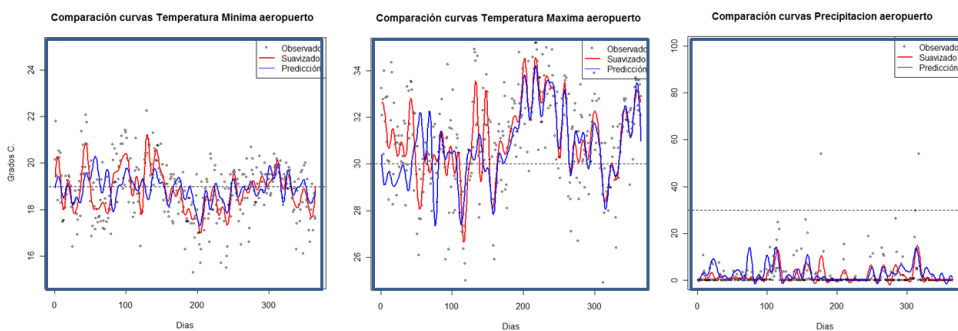


Figura 15 Comparación de curvas estimada y observada en la estación de aeropuerto.

Se realizó el análisis por medio de validación cruzada y la estación de aeropuerto. Los coeficientes de Kriging se resolvieron usando la librería geofd con la cual se dio solución a la estimación mediante variograma exponencial como queda justificado en la Figura 15. El método de validación cruzada (FCV) usado para elegir el número de los parámetros de suavización, es igualmente referido como una herramienta muy útil para predecir curvas observadas y estimadas como es mostrado en la Figura 15.

Debido a que los procesos se requería que corrieran de manera más rápida se bajó de resolución a los grid por medio del software de *Arcgis* usando el módulo de *Spatial Analysis* y la herramientas de *Resample* como lo muestra la Figura 16 generando 9,668 puntos válidos para interpolación de los sitios no muestreados.

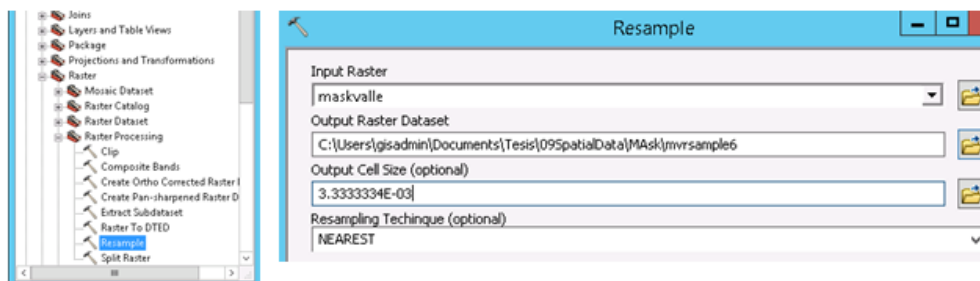


Figura 16. Cambio de resolución del Modelo de Elevación.

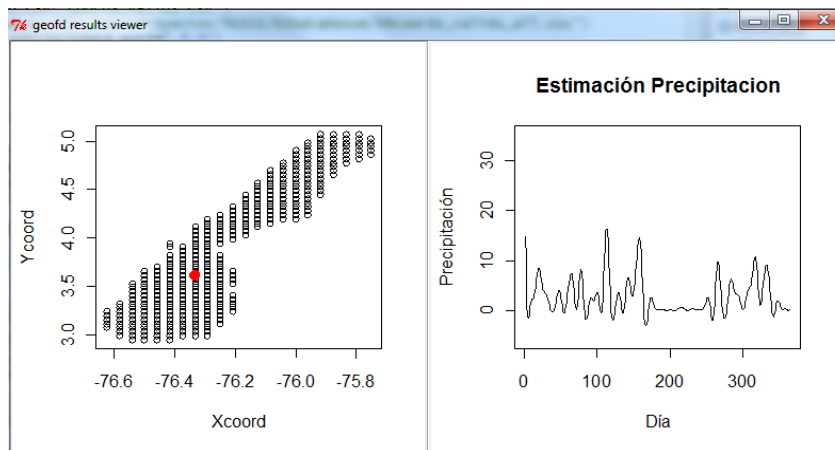


Figura 17 Función resultante estimada, en la selección en un punto.

Debido a la resolución del grid que inicialmente era de 0.00083333335 y la cantidad de 619,093 celdas (1091 col, 2377 filas), se transformó a 38,699 (273 columnas, 594 filas) con resolución de $3.333334E^{-03}$ (0.00333333334) se multiplico por 4 la resolución original y se realizó con la herramienta *Resample (Data Management)*, la

nueva resolución=0.0133333336, luego se extraen los puntos válidos por medio de R de este grid y se hace el enlace con la librería de geofd con la función **geofd.viewe**, aquí se generan las superficies por cada una de los días en los puntos válidos generando la interpolación.

Realizado este proceso, se procedió a generar las superficies, se creó un script **tranfromacioagrie.R**, usando para ello el modelo de elevación digital de la zona de estudio, la cual fue cortada debida al su tamaño inicial, después del proceso de re-muestreado, la resolución final fue $3.3333334E^{-03}$ pixel la cual se obtuvieron 9,680 celdas válidas con las cuales se realizó el proceso de generación de las superficies interpoladas como se ilustra en las mapas correspondientes a las siguientes figuras Figura 18, Figura 19, Figura 20, todas ellas realizadas con el software de Arcgis 9.1 y 10.1. De estas superficies se creó igualmente un video que ilustra todo el proceso del comportamiento modelado de las tres variables que puede ser consultado en las siguientes direcciones

ftp://ftp.ciat.cgiar.org/DAPA/users/jcardona/Valle_climate/Precipitacion_Valle_Del_Rio_Cauca_Kriging_O_1997-2012.gif

ftp://ftp.ciat.cgiar.org/DAPA/users/jcardona/Valle_climate/TemperaturaMaximaValle_Del_Rio_Cauca_Kriging_O_1997-2012.gif

ftp://ftp.ciat.cgiar.org/DAPA/users/jcardona/Valle_climate/TemperaturaMinimaValle_Del_Rio_Cauca_Kriging_O_1997-2012.gif

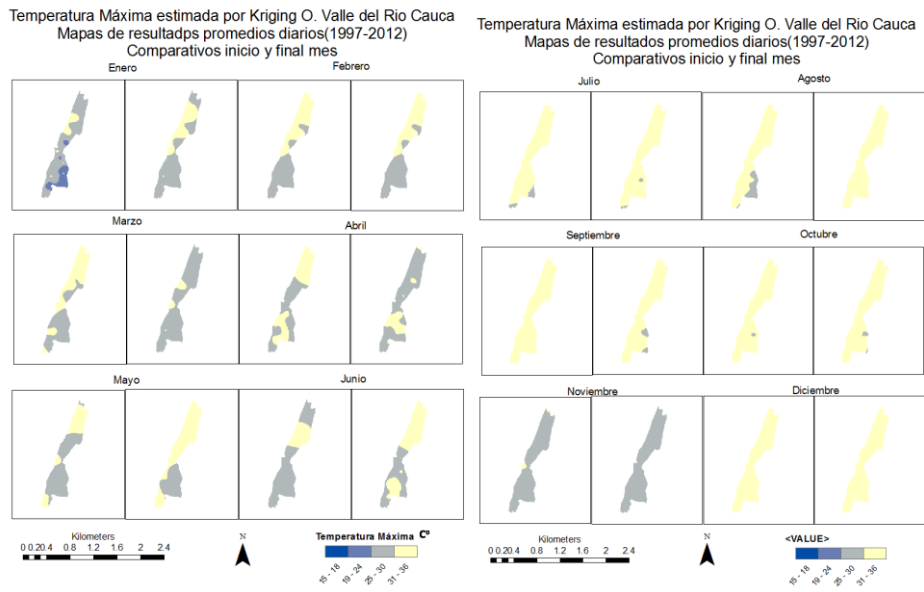


Figura 18 Resultado de superficies rasterizadas de temperatura máxima

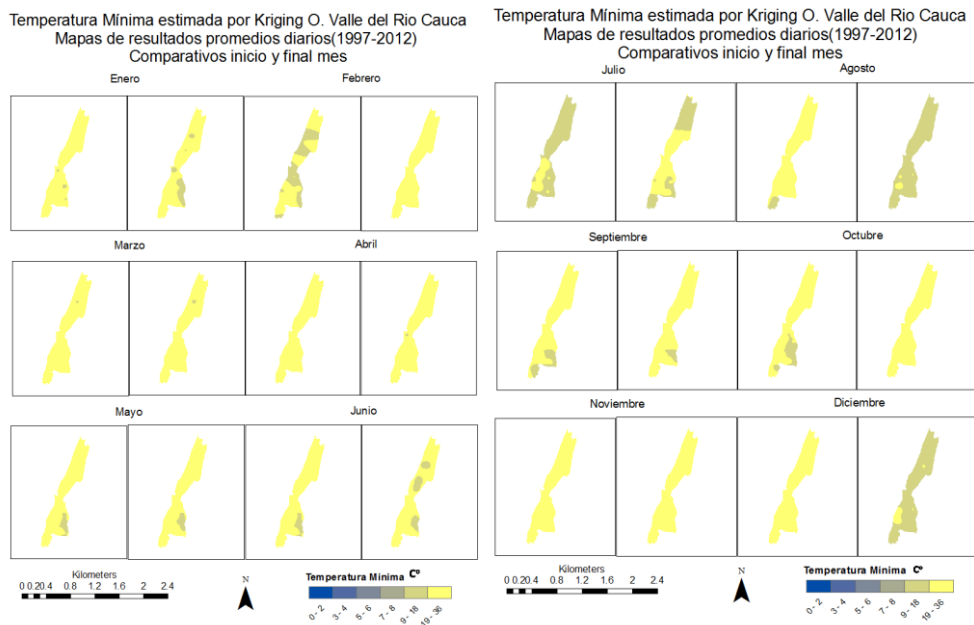


Figura 19 Resultado de superficies rasterizadas de temperatura mínima

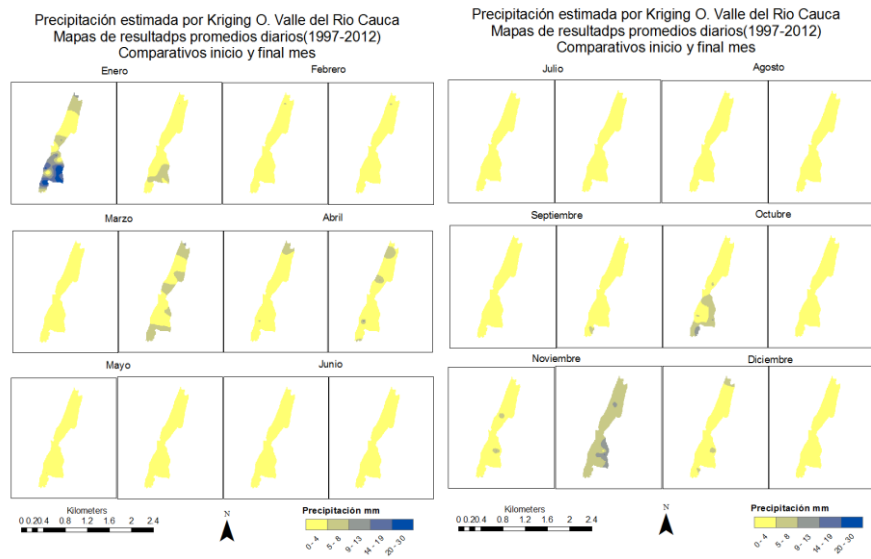


Figura 20 Resultado de superficies rasterizadas de precipitación

Finalmente los resultados obtenidos en este estudio se enfocaron en la evaluación y estimación de las curvas climáticas en sitios no muestreados y la selección del modelo de acuerdo a las variables, por tal motivo se usó un predictor kriging ordinario para datos funcionales teniendo como base datos reales de temperatura y precipitación de promedios de 16 años en 28 estaciones climáticas de una área con características homogéneas, se usó el modelo de validación cruzada funcional. Se puede afirmar que no es fácil proponer un modelo para modelar este tipo de curvas, y se debe aceptar que hay un alto grado de ruido observacional en los resultados, por tanto se puede concluir que las curvas obtenidas con las variables de temperaturas producen mejores resultados funcionales que con la precipitación y este enfoque permite abrir una posibilidad a estudiar este modelo.

Debido a que los datos funcionales que se generan (curvas) corresponden a resultados de curvas suavizadas, estas en algunos sitios no son representativas, es decir, que no representan realmente la variabilidad de los datos originales, este fenómeno se puede deber justamente al parámetro de suavización λ , con el cual se puede tomar la decisión de cuanta variabilidad se quiere tener en el modelo resultante, ya que si se suaviza mucho, se está dejando de lado las variaciones; dándole mayor importancia a la tendencia de los mismos datos. Una buena decisión, que se debe tomar antes de hacer cualquier modelo es definir qué parámetros se quieren resaltar, se puede investigar más en este sentido.

El proceso se podría mejorar, generando curvas anuales en vez de llevar a cabo el proceso de resumen de datos, este proceso representaría más la variabilidad interanual que se presenta en las series reales y que no es muy evidente en el resumen. Los métodos tradicionales de geoestadística univariada o algunos determinísticos como es el inverso de la distancia, no contemplan las estructuras de dependencia temporal (formas de las funciones), que también deben guardar cierto patrón o comportamiento lógico por ejemplo no pasar de 15 grados a 25 grados de temperatura en un día, por lo tanto hacer uso de esas técnicas en el caso de estimación de curvas de clima (temperatura máxima o mínima) no es lo más adecuado. Es por eso que se deben contemplar otro tipo de métodos, como los métodos funcionales, que son más adecuados ante este tipo de datos.

También se debe resaltar el tema de la mejora en la eficiencia de los tiempos de procesamiento ya que, en el caso que se use técnicas univariadas se debe construir modelos para cada día, correrlos y luego unir los datos lo cual es más costoso computacionalmente. Igualmente el hecho de usar el software R es una ventaja pues es de libre acceso y no tiene costo, está en constante actualización y las diferentes escuelas están aportando continuamente a su desarrollo y actualización constante, este hecho resulta ventajoso al momento de desarrollar el proyecto, pues no se requiere una inversión alta en paquetes de software estadísticos.

CONCLUSIONES

En este trabajo se usa un predictor Kriging ordinario para datos funcionales, que permite realizar la predicción espacial en ausencia de estacionariedad (Giraldo, 2009), aplicado al conjunto de datos climáticos de temperatura máxima y mínima y precipitación recolectados diariamente por estaciones a lo largo de 16 años en el Valle del Río Cauca, primero se hace un análisis descriptivo eliminando datos atípicos y aplicando control de calidad sobre ellos, seguido se hace el relleno de faltantes, luego en un proceso se resumen las series pasan a 365 días, se selecciona el número apropiado de bases y parámetros del modelo que mejor representa los datos y se transforma a dato funcional, seguido se realiza la predicción en cada uno de los sitios usando el método de validación cruzada, por último se realiza el proceso de interpolación en toda el área de estudio, generando así las superficies.

Las metodologías usadas en el actual proyecto son de gran utilidad para estudios ambientales, ya que se logró comprobar a partir de la hipótesis planteada en el inicio de este trabajo se estableciéndose los conceptos y herramientas del Análisis Funcional, y lográndose aplicar a series de tiempo de registros históricos de datos climáticos. En síntesis, estas metodologías estadísticas pueden ser de gran utilidad para posteriores estudios, aun estando en desarrollo hay muchos métodos que podrían aplicarse para identificar tendencias, correlaciones, etc.

En el desarrollo de este trabajo se puede observar que los datos generados por los modelos funcionales a partir de series de tiempo de estaciones puntuales, se destacan los datos climáticos ya que tienen una figura de curvas periódicas y que las 65 bases de Fourier fue la opción más apropiada para representar estas curvas de comportamiento. Posteriormente, en la fase de validación cruzada para evaluar qué el modelo de variograma era el más indicado, la suma de los errores cuadráticos en las 28 estaciones meteorológicas de acuerdo a la Tabla 3, muestra que los promedios resumen de todas las estaciones su error cuadrático es parecido, concluyéndose que los modelos exponencial y matérn son similares, y tienen un comportamiento similar en cada una las tres (3) variables evaluadas independientemente, sin embargo, el modelo esférico es el menos apropiado, ya que el modelo gaussiano no aplicó, debido a que no se presentó resultados en las evaluaciones. Siendo así, la predicción en sitios

no muestreados bajo la ausencia de estacionariedad, conlleva a la aplicación del modelo de variograma exponencial aplicando la técnica de kriging ordinario.

De acuerdo al formato y resolución de los datos climáticos generados a partir de la metodología empleada (datos funcionales, variogramas, kriging, etc.), se encontró que se pueden generar superficies climáticas continuas en serie de tiempo, con una resolución que depende de la fuente de datos. Para este caso el modelo de elevación digital se usó con fines de conversión, donde se redujo el tamaño de la grilla a una resolución de 0.0133333336 pixels, no se logró una mejor resolución ya que no se cuenta con una gran capacidad de memoria.

¿Se pueden usar las superficies en algún software GIS ejemplo Arcgis y que formato tienen las superficies resultantes?, el formato final es un raster de tipo ASCII estándar que puede ser leído y procesado por cualquier software GIS.

¿Cuál es la distribución espacial resultante?, ¿Cómo es el comportamiento de la temperatura y la precipitación en la zona de estudio?; Se determinó que los resultados obtenidos van de acuerdo a las referencias bibliográficas encontradas y citadas anteriormente, en las cuales se muestra que las lluvias son una masa cambiante en dirección sur y norte, donde las áreas más lluviosas están en el norte, sur y centro del valle del río Cauca. No obstante, se puede suponer que estas estaciones están localizadas en la parte montañosa, encontrándose que los periodos más secos están ubicados en el mes de julio y el más lluvioso en diciembre, este comportamiento se puede ver de manera evidente en los videos resultantes de la plataforma creada. Con respecto a las temperaturas se puede observar como el fenómeno se localiza en los primeros meses del año en el sur y se va trasladando. Aunque el método de resumen puede ser un buen mecanismo para definir indicadores, la variabilidad interanual no queda evidente, el poder generar las curvas anualmente y luego hacer una comparación interanual podría entregar mejores resultados respecto a los fenómenos globales ENSO (El niño y La niña) que afectan esta zona.

Finalmente, un enfoque válido para este tipo de análisis es que los factores climáticos van correlacionados con la altura, por lo tanto, es importante tener en cuenta que se podría

hacer una comparación entre diferentes tipos de análisis geoestadísticos tipo kriging ordinario y el cokriging.

Limitaciones

La principal limitación para llevar a cabo el desarrollo de esta investigación fue la obtención de los datos y la cantidad de los mismos.

La existencia y buena calidad de los mismos dan buenos resultados de los modelos con mayor aproximación cercanas a la realidad, con lo cual se favorece al aplicar otros procesos o modelos sobre dichos resultados.

BIBLIOGRAFIA

- Comas, C., Delicado, P., Giraldo, R., & Mateu, J. (2009). Statistics for spatial functional data: some recent contributions. *Environmetrics*.
- Cordano, E., & Eccel, E. (2013). Package “rmawgen.”
- Cressie, N. (1989). geostatistics. *The american statistician*, 43(4) : 611(23).
- Delicado, P., Giraldo, R., & Mateu, J. (2007). Geostatistics for functional data: an ordinary kriging approach. *Environmental and ecological statistics*.
- Diaz, V. M. (2002). Geoestadística aplicada, (instituto de geofísica, UNAM. y instituto de geofísica y astronomía, CITMA).
- Enrique Cortes, & Barrios, C. (2010). Nuevo calendario de temporadas secas y lluviosas en el valle del río cauca, 4–6.
- Ferraty, & Vieu. (2006). Non parametric functional data analysis. *theory and practice.*, (springer).
- Giraldo, R. (2007). Análisis exploratorio de variables regionalizadas con métodos funcionales introducción teoría, (1), 115–127.
- Giraldo, R. (2009a). Estadística espacial.
- Giraldo, R. (2009b). Introducción a la geoestadística.
- Giraldo, Ramon, Mateu, J., & Delicado, P. (2012). geofd : an r package for function-valued geostatistical prediction, (3).
- Gourlard M., & Volts M. (1993). Geostatistical interpolation of curves: A case study on soil science. *Academic Press*, 2, 805–816.
- He, Muller, G., & Wang, J. (2000). Extending correlation and regression from multivariate to functional data. *asymptotics in statistics and probability*, (brill academic publisher), 1–14.
- Journel, A. G., & Ch. J. Huijbregts. (1978). Mining geostatistics. *academics press, new york*.
- Justiniano, P., & Junior, R. (2004). Fundamentos de geoestadística Parte I :
- Petitgas, P. (1996). Geostatistics and their applications to fisheries survey aata. *computers and fisheries research*, (chapman-hall, londres), 114–142.
- Ramsay, J. O., & Silverman, B. W. (2005). applied functional data analysis: methods and case studies.
- Ramsay, J., & Silverman, B. (1997). *Functional data analysis*. *springer*.
- Ramsay, J., & Silverman, B. (2009). Functional data analysis with r and matlab.

Reyes, C. M. A. (2010). Estimación paramétrica y no paramétrica de la tendencia en datos con dependencia espacial. un estudio de simulación. *Estimación paramétrica y no paramétrica de la tendencia en datos con dependencia espacial*.

Samper, F., & Carrera, J. (1993). *Geoestadística aplicada a la hidrología subterránea*. barcelona: centro internacional de métodos numéricos en ingeniería.

ANEXOS

Anexo 1. Listado de tablas

Tabla 4 Listado de estaciones climáticas

ESTACIONES		
Estación	Inicio	Final
El naranjo	9/1/1993	10/10/2013
Miranda	9/1/1993	10/10/2013
Cenicana	9/1/1993	10/10/2013
melendez	9/1/1993	10/10/2013
Candelaria	9/1/1993	10/10/2013
Amaine	9/1/1993	10/10/2013
san marcos	9/1/1993	10/10/2013
Guacari	9/1/1993	10/10/2013
tulua	9/1/1993	10/10/2013
la paila	9/1/1993	10/10/2013
La virginia	9/1/1993	10/10/2013
Palmira la rita	9/1/1993	10/10/2013
Corinto	9/1/1995	10/10/2013
Boca del Palo	9/1/1995	10/10/2013
pradera	9/1/1995	10/10/2013
Cartago	11/1/1995	10/10/2013
Jamundi	12/1/1995	10/10/2013
Ortugal	1/1/1996	10/10/2013
Viterbo3	6/1/1996	10/10/2013
Aeropuerto	7/6/1996	10/10/2013
Palmira san jose	12/1/1996	10/10/2013
Arroyohondo	12/1/1996	10/10/2013
Ginebra	2/1/1997	10/10/2013
Ptar cali	5/1/1997	10/10/2013
Bugalagrande	5/9/1997	10/10/2013
Zarzal	6/1/1997	10/10/2013
Yotoco	7/1/1997	10/10/2013
Sant Quilichao	8/29/1997	10/10/2013
Riofrio	11/1/2001	10/10/2013

Tabla 5 Estadísticos de Resumen Temperatura Máxima

TEMPERATURA MÁXIMA										
Estación	Reg .	Min	Max	Mediana	Variancia	Stand . Desv.	Mediana	CV %	NA	NA %
smarcos	584 4	22	37	29.96 5	3.255	1.804	30.1	6.02 1	46	0.787
aeropuerto	584 4	21	39	29.87 9	3.397	1.843	30	6.16 9	144	2.464
guacari	584 4	22	37	29.85 6	3.622	1.903	30	6.37 4	45	0.77
melendez	584 4	22	36	29.74 2	3.595	1.896	29.8	6.37 5	57	0.975
palrita	584 4	21	36	29.43	3.555	1.885	29.5	6.40 7	146	2.498
arroyohondo	584 4	21	36	29.54 5	3.589	1.894	29.7	6.41 2	796	13.62 1
ptarcali	584 4	22	38	30.05 6	3.791	1.947	30.1	6.47 8	124 1	21.23 5
ginebra	584 4	21	37	29.43 2	3.652	1.911	29.5	6.49 3	692	11.84 1
cenicana	584 4	21	36	29.1	3.576	1.891	29.2	6.49 8	70	1.198
yotoco	584 4	22	38	30.04 1	3.853	1.963	30.2	6.53 4	641	10.96 9
ortigal	584 4	22	51	29.53 1	3.737	1.933	29.6	6.54 6	266	4.552
quilichao	584 4	20	36	29.14 9	3.674	1.917	29.3	6.57 6	551	9.428
jamundi	584 4	21	36	29.51 6	3.77	1.942	29.6	6.57 8	96	1.643
candelaria	584 4	21	39	29.92 7	3.876	1.969	30	6.57 9	51	0.873
amaime	584 4	21	36	29.17 1	3.737	1.933	29.3	6.62 7	330	5.647
naranjo	584 4	21	52	29.30 1	3.813	1.953	29.4	6.66 4	235	4.021
bugalagra	584 4	22	38	30.34 7	4.098	2.024	30.5	6.67 1	590	10.09 6
lapaila	584 4	22	43	30.06 2	4.03	2.007	30.2	6.67 8	157	2.687
pradera	584 4	21	37	29.02 4	3.76	1.939	29.2	6.68 1	60	1.027

corinto	584 4	21	35	28.46 5	3.625	1.904	28.6	6.68 9	67	1.146
miranda	584 4	22	36	29.25 9	3.833	1.958	29.4	6.69 1	101	1.728
tutua	584 4	22	37	29.60 3	3.983	1.996	29.7	6.74 2	175	2.995
lavirginia	584 4	22	37	30.48 9	4.294	2.072	30.6	6.79 7	105	1.797
bdelpalo	584 4	20	39	30.01 7	4.172	2.043	30.2	6.80 5	708	12.11 5
cartago	584 4	20	38	30.58 5	4.521	2.126	30.7	6.95 2	401	6.862
viterbo	584 4	22	37	30.14 7	4.427	2.104	30.3	6.97 9	251	4.295
palsjose	584 4	16	76	29.47	4.856	2.204	29.5	7.47 8	675	11.55
zarzal	584 4	21	94	30.39 6	5.395	2.323	30.5	7.64 2	995	17.02 6

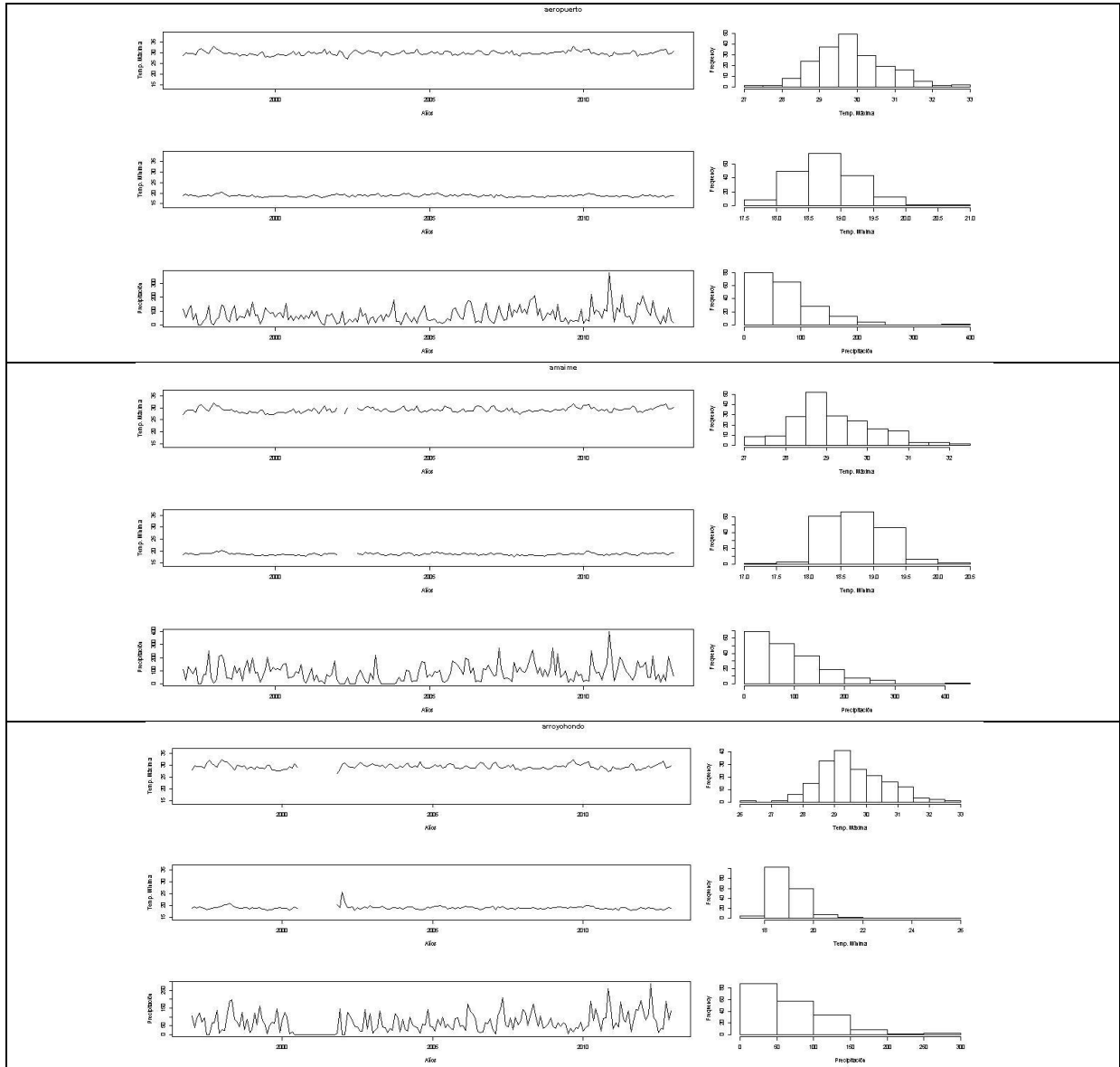
Tabla 6 Estadísticos de Resumen Temperatura Mínima

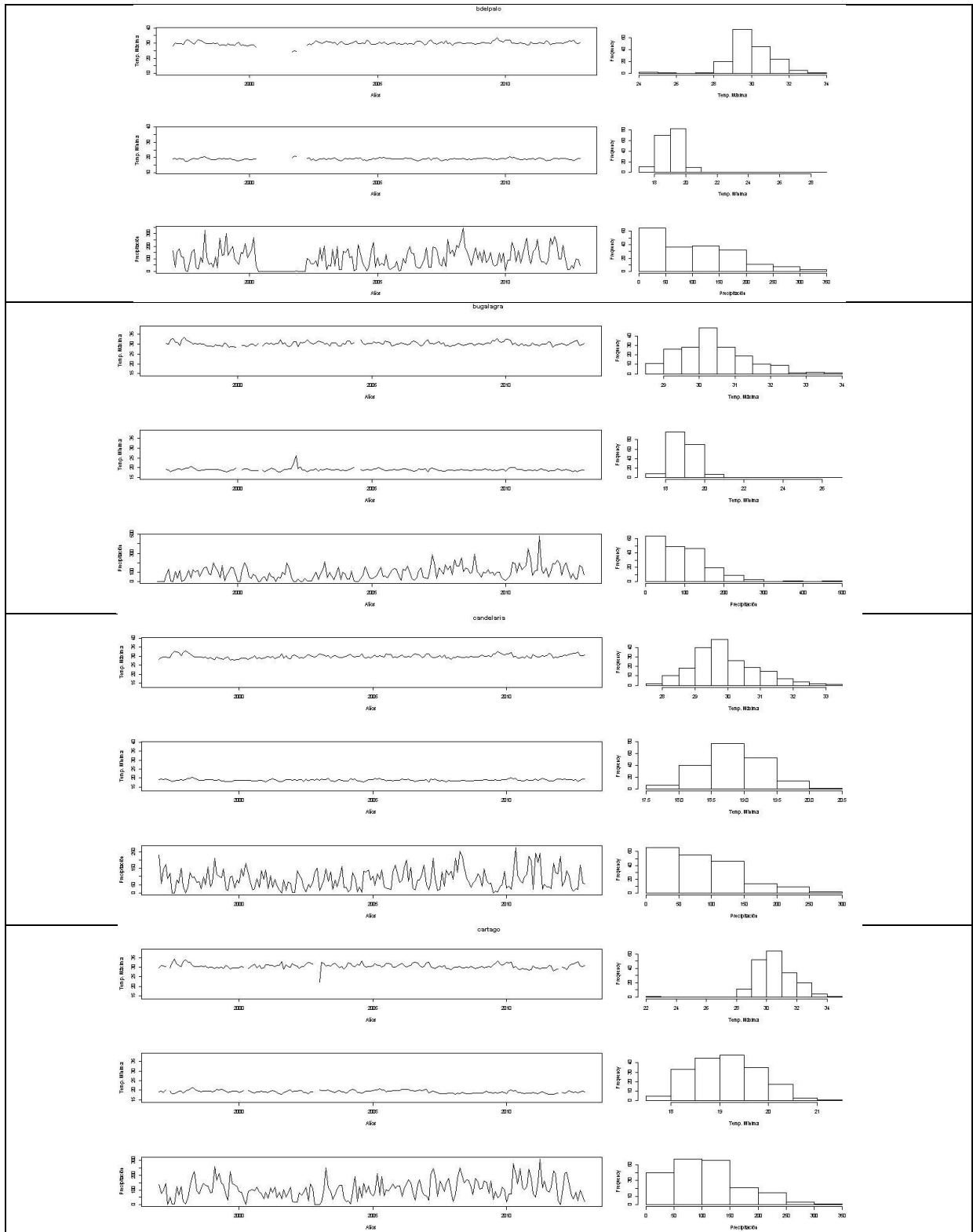
TEMPERATURA MÍNIMA										
Estación	Reg.	Min	Max	Media	Variance	Stand. Desv.	Mediana	CV %	NA	NA %
aeropuerto	5844	14	22	18.757	1.262	1.123	18.9	5.989	149	2.55
amaime	5844	14	22	18.729	0.997	0.998	18.8	5.331	330	5.647
arroyohondo	5844	14	26	18.905	1.207	1.099	19	5.811	812	13.895
bdelpalo	5844	10	31	19.012	1.276	1.13	19.1	5.942	705	12.064
bugalagra	5844	15	54	18.959	1.439	1.2	19	6.327	589	10.079
candelaria	5844	14	23	18.817	1.138	1.067	18.9	5.669	51	0.873
cartago	5844	14	24	19.13	1.146	1.071	19.2	5.596	402	6.879
cenicana	5844	14	312	19.114	15.888	3.986	19.1	20.854	73	1.249
corinto	5844	15	22	18.793	0.787	0.887	18.8	4.721	55	0.941
ginebra	5844	14	23	18.865	1.176	1.084	18.9	5.748	688	11.773
guacari	5844	14	23	19.107	1.166	1.08	19.2	5.651	46	0.787
jamundi	5844	7	22	18.372	1.768	1.33	18.5	7.237	94	1.608
lapaila	5844	14	22	19.047	1.052	1.026	19.1	5.385	155	2.652
lavirginia	5844	15	22	19.168	0.936	0.967	19.3	5.047	108	1.848
melendez	5844	13	22	18.872	1.208	1.099	19	5.824	54	0.924
miranda	5844	14	22	18.674	1.063	1.031	18.8	5.521	104	1.78
naranja	5844	10	31	18.824	1.18	1.086	18.9	5.771	251	4.295
ortigal	5844	13	54	18.493	1.568	1.252	18.6	6.771	270	4.62
palrita	5844	14	22	18.642	1.053	1.026	18.7	5.505	142	2.43
palsjose	5844	13	22	18.412	1.262	1.123	18.5	6.101	713	12.201
pradera	5844	14	22	18.523	1.058	1.029	18.6	5.553	55	0.941
ptarcali	5844	16	54	19.977	1.66	1.288	20	6.449	1242	21.253
quilichao	5844	13	24	18.39	1.409	1.187	18.5	6.455	554	9.48
smarcos	5844	14	23	19.217	1.069	1.034	19.3	5.38	46	0.787
tutua	5844	13	54	18.778	1.458	1.207	18.8	6.43	175	2.995
viterbo	5844	14	54	18.55	1.339	1.157	18.7	6.238	254	4.346
yotoco	5844	12	23	19.039	1.243	1.115	19.1	5.856	645	11.037
zarzal	5844	14	40	18.731	1.725	1.313	18.8	7.012	998	17.077

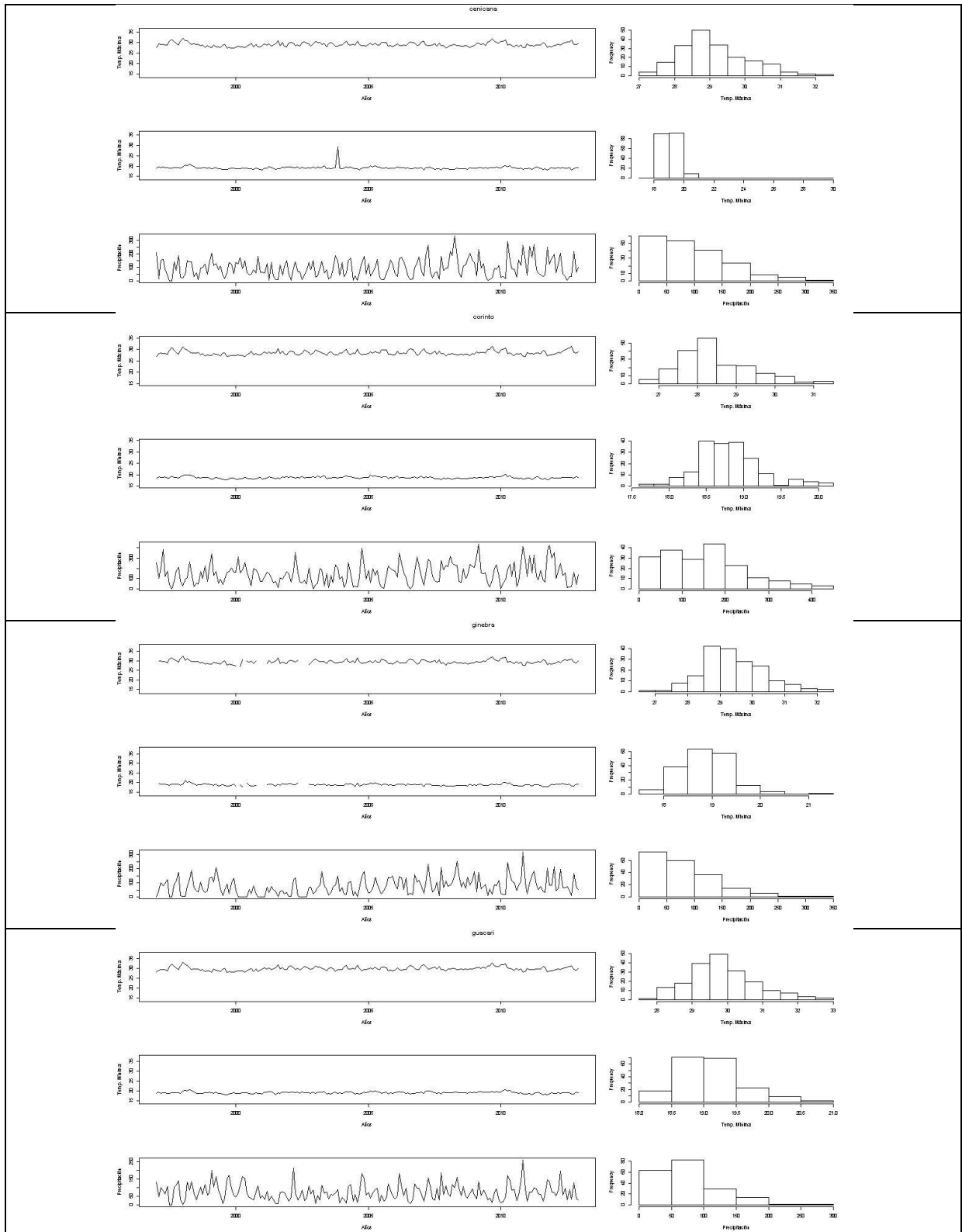
Tabla 7 Estadísticos de Resumen Precipitación

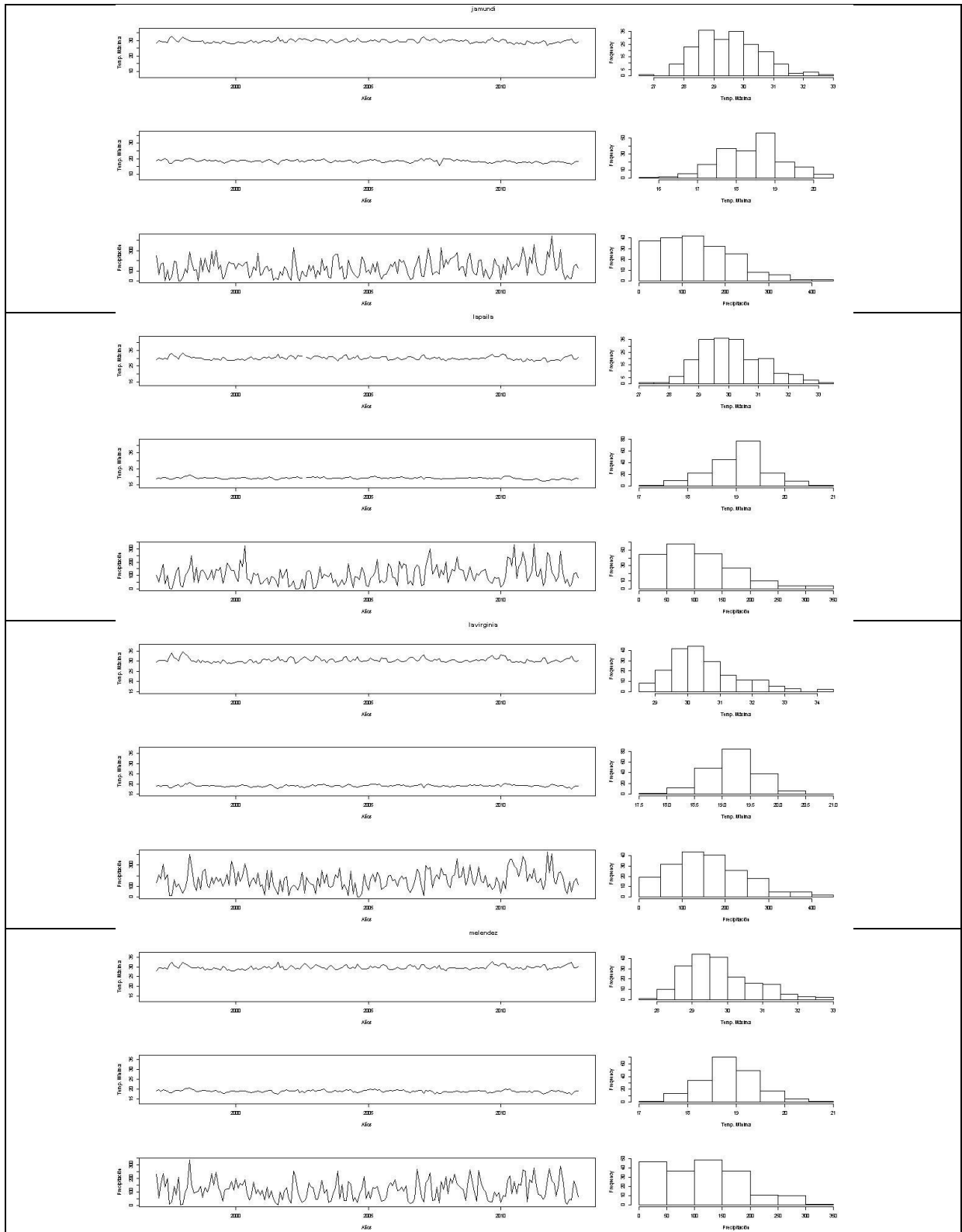
PRECIPITACIÓN										
Estación	Reg.	Min	Max	Media	Variance	Stand. Desv.	Median	CV %	NA	NA %
aeropuerto	5844	0	87	2.42	42.873	6.548	0	270.568	137	2.344
amaime	5844	0	78	3.024	58.296	7.635	0	252.486	326	5.578
arroyohondo	5844	0	91	2.556	46.724	6.835	0	267.429	765	13.09
bdelpalo	5844	0	92	3.725	74.578	8.636	0.1	231.835	699	11.961
bugalagra	5844	0	93	3.38	66.899	8.179	0.1	241.988	589	10.079
candelaria	5844	0	82	2.832	59.204	7.694	0	271.695	49	0.838
cartago	5844	0	74	3.727	60.456	7.775	0.2	208.622	397	6.793
cenicana	5844	0	102	3.189	68.461	8.274	0	259.458	58	0.992
corinto	5844	0	142	4.911	122.404	11.064	0.2	225.283	57	0.975
ginebra	5844	0	83	2.831	51.52	7.178	0	253.541	676	11.567
guacari	5844	0	76	2.441	38.005	6.165	0	252.553	44	0.753
jamundi	5844	0	81	4.381	93.319	9.66	0.1	220.502	113	1.934
lapaila	5844	0	99	3.621	67.673	8.226	0.1	227.185	145	2.481
lavirginia	5844	0	197	5.457	111.077	10.539	0.5	193.134	214	3.662
melendez	5844	0	98	3.817	82.742	9.096	0	238.309	53	0.907
miranda	5844	0	92	3.78	84.732	9.205	0.1	243.518	98	1.677
naranja	5844	0	94	4.085	88.085	9.385	0.1	229.752	226	3.867
ortigal	5844	0	127	3.445	76.413	8.741	0.1	253.743	203	3.474
palrita	5844	0	96	2.972	57.684	7.595	0	255.552	110	1.882
palsjose	5844	0	106	2.901	56.497	7.516	0.1	259.099	498	8.522
pradera	5844	0	91	2.897	63.842	7.99	0	275.807	52	0.89
ptarcali	5844	0	105	2.631	49.187	7.013	0	266.566	1267	21.68
quilichao	5844	0	87	4.477	97.061	9.852	0.1	220.057	549	9.394
smarcos	5844	0	86	2.25	37.485	6.122	0	272.111	45	0.77
tutua	5844	0	78	3.341	57.329	7.572	0.1	226.627	171	2.926
viterbo	5844	0	90	5.835	110.701	10.521	0.7	180.316	199	3.405
Yotoco	5844	0	71	2.888	49.14	7.01	0	242.728	530	9.069
Zarzal	5844	0	101	3.012	49.618	7.044	0.1	233.865	984	16.838

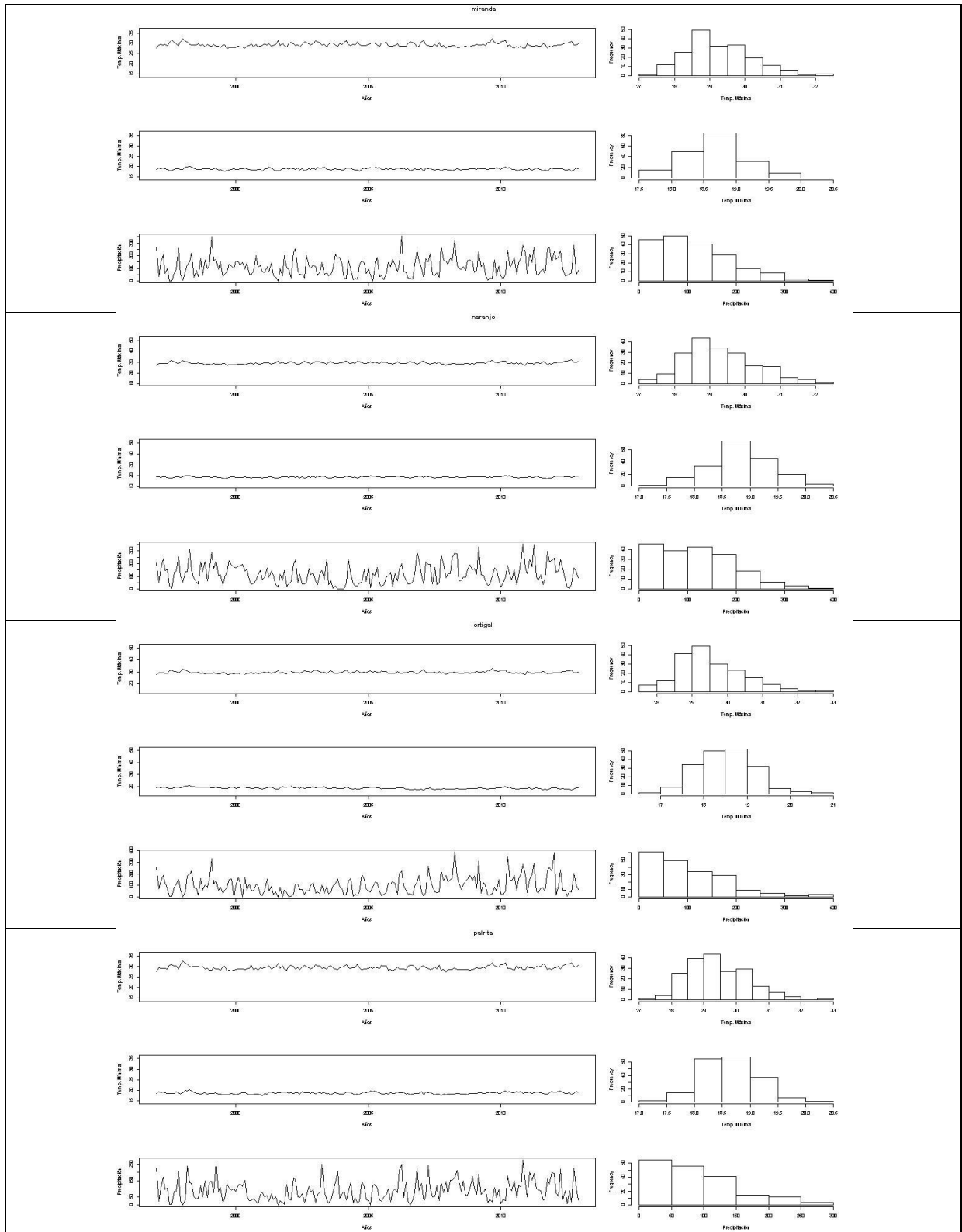
Anexo 2. Histogramas de series históricas de temperatura máxima, mínima y temperatura

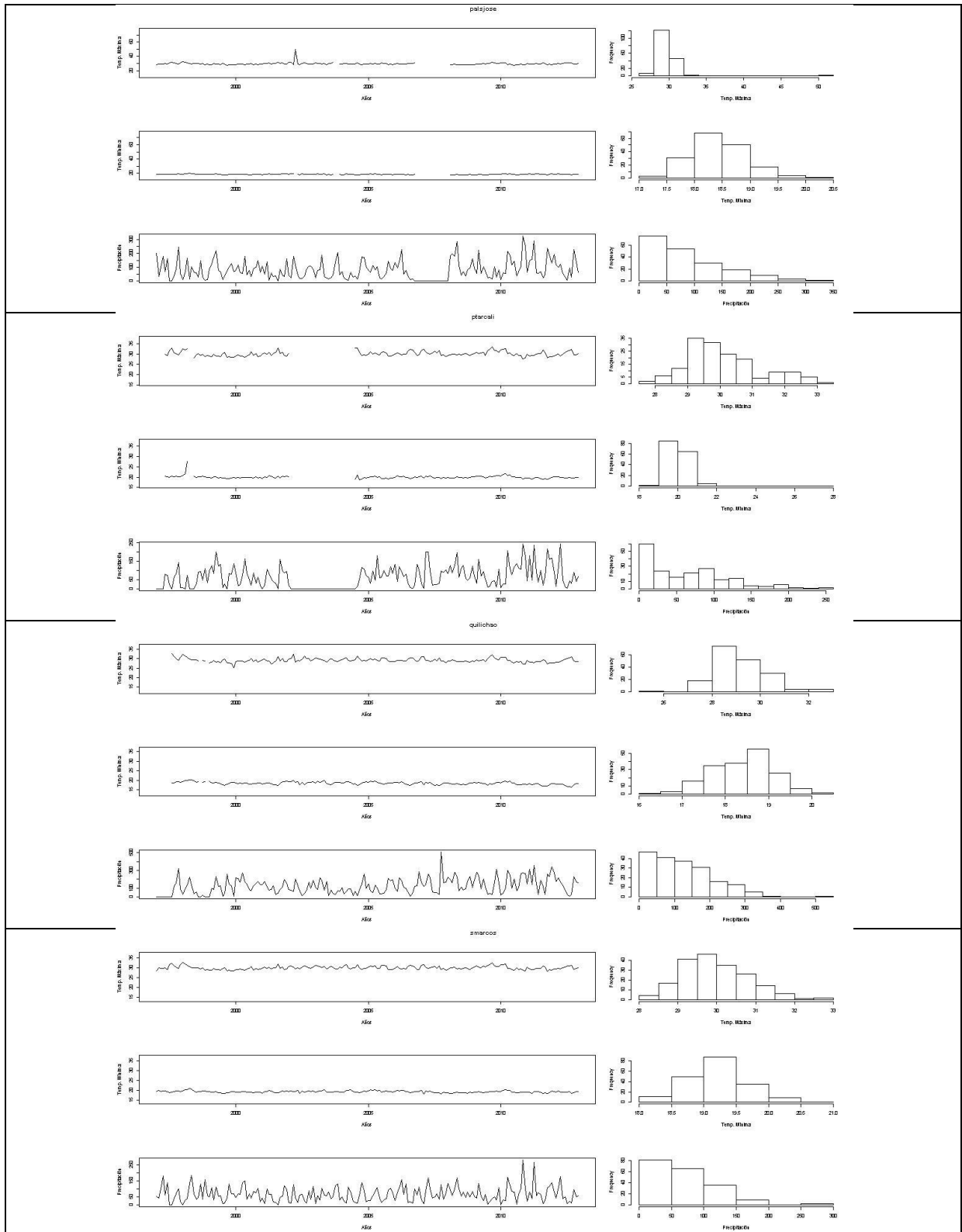


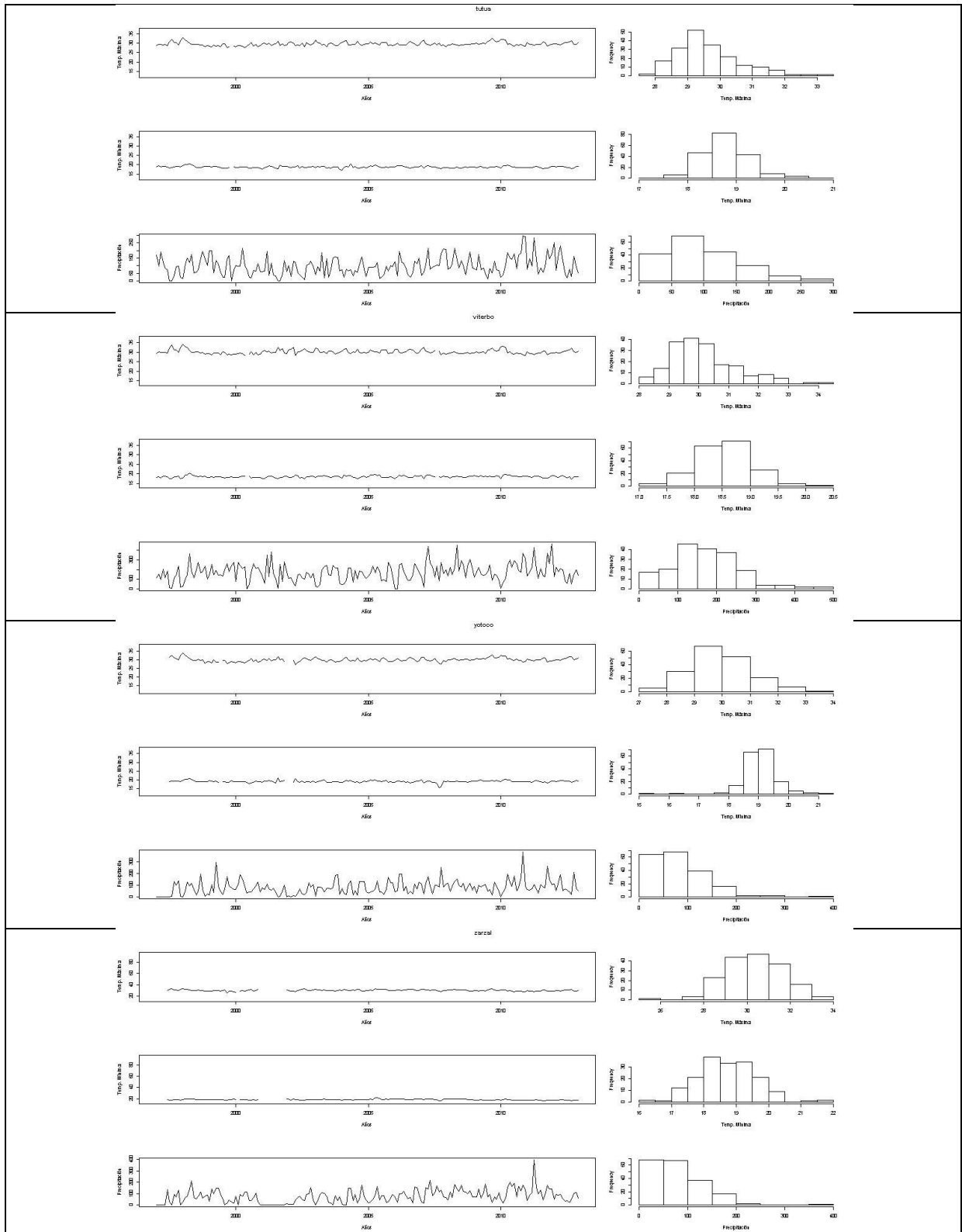












Anexo 3. Listado de código en lenguaje R

```

# Programa      = 00INSTALACIONLIBRERIAS.R
# Autor         = Jorge CArdon
# Actualziacion =2014-0501
# Nota         = Este programa instala todas las librerías necesarias para llevar a cabo
#              = el análisis de las series climáticas
#
install.packages("maptools", dep=TRUE)
install.packages("rgeos", dep=TRUE)
install.packages("raster", dep=TRUE)
install.packages("SDMTools", dep=TRUE)
install.packages("snowfall", dep=TRUE)
install.packages("rgdal", dep=TRUE)
install.packages("ggplot2", dep=TRUE)
install.packages("BIOMOD", repos="http://R-Forge.R-project.org")
install.packages("ggplot2", dep=TRUE)
install.packages("rtf", dep=TRUE)
install.packages("gdata", dep=TRUE)
install.packages("gWidgets", dep=TRUE)
install.packages("gWidgetsRGtk2", dep=TRUE)
install.packages("RMAWGEN", dep=TRUE)
install.packages("Kendall", dep=TRUE)
install.packages("tseries", dep=TRUE)
options("guiToolkit"="RGtk2")
install.packages("fda", dep=TRUE)
install.packages("geofd", dep=TRUE)

## Geo climate server climate wizard
install.packages("Ncdf", dep=TRUE)
install.packages("nlme", dep=TRUE)
install.packages("RColorBrewer", dep=TRUE)
install.packages("raster", dep=TRUE)
install.packages("rgdal", dep=TRUE)
install.packages("maptools", dep=TRUE)
# Programa      = 01_Integraciondatos1.R
# Autor         = Jorge CArdon
# Actualziacion = 2014-0501
# Nota         = Este programa integra todos los archivos descargados de estaciones
climáticas
#              = los registros estan compuestos de nombre, fecha, variables
#              = seré integrados en un archivo para luego análisis de las series

```

```

reemplazofecha <- function(){
  Df2.1=Df.1[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.2=Df.2[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.3=Df.3[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.4=Df.4[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.5=Df.5[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.6=Df.6[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.7=Df.7[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.8=Df.8[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.9=Df.9[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.10=Df.10[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.11=Df.11[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.12=Df.12[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.13=Df.13[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.14=Df.14[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.15=Df.15[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.16=Df.16[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.17=Df.17[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.18=Df.18[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.19=Df.19[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.20=Df.20[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.21=Df.21[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.22=Df.22[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.23=Df.23[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.24=Df.24[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.25=Df.25[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.26=Df.26[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.27=Df.27[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.28=Df.28[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.29=Df.29[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.30=Df.30[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.31=Df.31[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.32=Df.32[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.33=Df.33[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]
  Df2.34=Df.34[c("nm_ema","fc_obsrvcion","dia","mes","ano", "tmp_mnma" )]

```

```

setwd(PATH_VALLERESUMEFILES)

```

```

getwd()
write.csv(Df2.1 , file = LIST_ESTACIONES[1])
write.csv(Df2.2 , file = LIST_ESTACIONES[2])
write.csv(Df2.3 , file = LIST_ESTACIONES[3])
write.csv(Df2.4 , file = LIST_ESTACIONES[4])
write.csv(Df2.5 , file = LIST_ESTACIONES[5])
write.csv(Df2.6 , file = LIST_ESTACIONES[6])
write.csv(Df2.7 , file = LIST_ESTACIONES[7])
write.csv(Df2.8 , file = LIST_ESTACIONES[8])
write.csv(Df2.9 , file = LIST_ESTACIONES[9])

```

```

write.csv(Df2.10 , file = LIST_ESTACIONES[10])
write.csv(Df2.11 , file = LIST_ESTACIONES[11])
write.csv(Df2.12 , file = LIST_ESTACIONES[12])
write.csv(Df2.13 , file = LIST_ESTACIONES[13])
write.csv(Df2.14 , file = LIST_ESTACIONES[14])
write.csv(Df2.15 , file = LIST_ESTACIONES[15])
write.csv(Df2.16 , file = LIST_ESTACIONES[16])
write.csv(Df2.17 , file = LIST_ESTACIONES[17])
write.csv(Df2.18 , file = LIST_ESTACIONES[18])
write.csv(Df2.19 , file = LIST_ESTACIONES[19])
write.csv(Df2.20 , file = LIST_ESTACIONES[20])
write.csv(Df2.11 , file = LIST_ESTACIONES[21])
write.csv(Df2.12 , file = LIST_ESTACIONES[22])
write.csv(Df2.13 , file = LIST_ESTACIONES[23])
write.csv(Df2.14 , file = LIST_ESTACIONES[24])
write.csv(Df2.15 , file = LIST_ESTACIONES[25])
write.csv(Df2.16 , file = LIST_ESTACIONES[26])
write.csv(Df2.17 , file = LIST_ESTACIONES[27])
write.csv(Df2.18 , file = LIST_ESTACIONES[28])
write.csv(Df2.19 , file = LIST_ESTACIONES[29])
write.csv(Df2.20 , file = LIST_ESTACIONES[30])
write.csv(Df2.11 , file = LIST_ESTACIONES[31])
write.csv(Df2.12 , file = LIST_ESTACIONES[32])
write.csv(Df2.13 , file = LIST_ESTACIONES[33])
write.csv(Df2.14 , file = LIST_ESTACIONES[34])
}

```

```

frename <- function(){
  Df2.1= rename(Df2.1,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="quilichao"))
  Df2.2= rename(Df2.2,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="corinto"))
  Df2.3= rename(Df2.3,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="guachinte"))
  Df2.4= rename(Df2.4,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="naranja"))
  Df2.5= rename(Df2.5,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="bdelpalo"))
  Df2.6= rename(Df2.6,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="miranda "))
  Df2.7= rename(Df2.7,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="ortigal"))
  Df2.8= rename(Df2.8,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="jamundi"))
  Df2.9= rename(Df2.9,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="cenicana"))
  Df2.10= rename(Df2.10,c("ano" = "year", "mes" = "month","dia" = "day",

```



```

"tmp_mnma"="eltiple"))
Df2.11= rename(Df2.11,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="melendez"))
Df2.12= rename(Df2.12,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="pradera"))
Df2.13= rename(Df2.13,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="candelaria"))
Df2.14= rename(Df2.14,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="ptarcali"))
Df2.15= rename(Df2.15,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="palsjose"))
Df2.16= rename(Df2.16,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="amaime"))
Df2.17= rename(Df2.17,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="rozo"))
Df2.18= rename(Df2.18,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="smarcos"))
Df2.19= rename(Df2.19,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="ginebra"))
Df2.20= rename(Df2.20,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="guacari"))
Df2.21= rename(Df2.21,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="yotoco"))
Df2.22= rename(Df2.22,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="buga"))
Df2.23= rename(Df2.23,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="tutua"))
Df2.24= rename(Df2.24,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="riofrio"))
Df2.25= rename(Df2.25,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="bugalagra"))
Df2.26= rename(Df2.26,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="lapaila"))
Df2.27= rename(Df2.27,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="zarzal"))
Df2.28= rename(Df2.28,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="distrut"))
Df2.29= rename(Df2.29,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="cartago"))
Df2.30= rename(Df2.30,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="lavirginia"))
Df2.31= rename(Df2.31,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="viterbo"))
Df2.32= rename(Df2.32,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="aeropuerto"))
Df2.33= rename(Df2.33,c("ano" = "year", "mes" = "month","dia" = "day",
"tmp_mnma"="arroyohondo"))

```

```

Df2.34= rename(Df2.34,c("ano" = "year", "mes" = "month", "dia" = "day",
"tmp_mnma"="palrita"))
}

Resumen=data.frame()
resume=read.csv("C:/DatosProyectos/TESIS-Datos/EstacionesValle/Detalle-
al/Resumen.csv")
resume$fc_obsrvcion <- as.Date(resume$fc_obsrvcion,"%m-%d-%Y")

Df2.1$fc_obsrvcion <- as.Date(Df2.1$fc_obsrvcion,"%m-%d-%Y")
Df2.2$fc_obsrvcion <- as.Date(Df2.2$fc_obsrvcion,"%m-%d-%Y")
Df2.3$fc_obsrvcion <- as.Date(Df2.3$fc_obsrvcion,"%m-%d-%Y")
Df2.4$fc_obsrvcion <- as.Date(Df2.4$fc_obsrvcion,"%m-%d-%Y")
Df2.5$fc_obsrvcion <- as.Date(Df2.5$fc_obsrvcion,"%m-%d-%Y")
Df2.6$fc_obsrvcion <- as.Date(Df2.6$fc_obsrvcion,"%m-%d-%Y")
Df2.7$fc_obsrvcion <- as.Date(Df2.7$fc_obsrvcion,"%m-%d-%Y")
Df2.8$fc_obsrvcion <- as.Date(Df2.8$fc_obsrvcion,"%m-%d-%Y")
Df2.9$fc_obsrvcion <- as.Date(Df2.9$fc_obsrvcion,"%m-%d-%Y")
Df2.10$fc_obsrvcion <- as.Date(Df2.10$fc_obsrvcion,"%m-%d-%Y")

Df2.11$fc_obsrvcion <- as.Date(Df2.11$fc_obsrvcion,"%m-%d-%Y")
Df2.12$fc_obsrvcion <- as.Date(Df2.12$fc_obsrvcion,"%m-%d-%Y")
Df2.13$fc_obsrvcion <- as.Date(Df2.13$fc_obsrvcion,"%m-%d-%Y")
Df2.14$fc_obsrvcion <- as.Date(Df2.14$fc_obsrvcion,"%m-%d-%Y")
Df2.15$fc_obsrvcion <- as.Date(Df2.15$fc_obsrvcion,"%m-%d-%Y")
Df2.16$fc_obsrvcion <- as.Date(Df2.16$fc_obsrvcion,"%m-%d-%Y")
Df2.17$fc_obsrvcion <- as.Date(Df2.17$fc_obsrvcion,"%m-%d-%Y")
Df2.18$fc_obsrvcion <- as.Date(Df2.18$fc_obsrvcion,"%m-%d-%Y")
Df2.19$fc_obsrvcion <- as.Date(Df2.19$fc_obsrvcion,"%m-%d-%Y")
Df2.20$fc_obsrvcion <- as.Date(Df2.20$fc_obsrvcion,"%m-%d-%Y")

Df2.21$fc_obsrvcion <- as.Date(Df2.21$fc_obsrvcion,"%m-%d-%Y")
Df2.22$fc_obsrvcion <- as.Date(Df2.22$fc_obsrvcion,"%m-%d-%Y")
Df2.23$fc_obsrvcion <- as.Date(Df2.23$fc_obsrvcion,"%m-%d-%Y")
Df2.24$fc_obsrvcion <- as.Date(Df2.24$fc_obsrvcion,"%m-%d-%Y")
Df2.25$fc_obsrvcion <- as.Date(Df2.25$fc_obsrvcion,"%m-%d-%Y")
Df2.26$fc_obsrvcion <- as.Date(Df2.26$fc_obsrvcion,"%m-%d-%Y")
Df2.27$fc_obsrvcion <- as.Date(Df2.27$fc_obsrvcion,"%m-%d-%Y")
Df2.28$fc_obsrvcion <- as.Date(Df2.28$fc_obsrvcion,"%m-%d-%Y")
Df2.29$fc_obsrvcion <- as.Date(Df2.29$fc_obsrvcion,"%m-%d-%Y")
Df2.30$fc_obsrvcion <- as.Date(Df2.30$fc_obsrvcion,"%m-%d-%Y")

Df2.31$fc_obsrvcion <- as.Date(Df2.31$fc_obsrvcion,"%m-%d-%Y")
Df2.32$fc_obsrvcion <- as.Date(Df2.32$fc_obsrvcion,"%m-%d-%Y")
Df2.33$fc_obsrvcion <- as.Date(Df2.33$fc_obsrvcion,"%m-%d-%Y")
Df2.34$fc_obsrvcion <- as.Date(Df2.34$fc_obsrvcion,"%m-%d-%Y")

```

```

merge1<-merge(Df2.1, Df2.11 ,by="fc_obsrvcion", all=TRUE)
merge2<-merge(Df2.2, Df2.12 ,by="fc_obsrvcion", all=TRUE)
merge3<-merge(Df2.3, Df2.13 ,by="fc_obsrvcion", all=TRUE)
merge4<-merge(Df2.4, Df2.14 ,by="fc_obsrvcion", all=TRUE)
merge5<-merge(Df2.5, Df2.15 ,by="fc_obsrvcion", all=TRUE)
merge6<-merge(Df2.6, Df2.16 ,by="fc_obsrvcion", all=TRUE)
merge7<-merge(Df2.7, Df2.17 ,by="fc_obsrvcion", all=TRUE)
merge8<-merge(Df2.8, Df2.18 ,by="fc_obsrvcion", all=TRUE)
merge9<-merge(Df2.9, Df2.19 ,by="fc_obsrvcion", all=TRUE)
merge10<-merge(Df2.10, Df2.20 ,by="fc_obsrvcion", all=TRUE)

```

```

merge11<-merge(Df2.21, Df2.28 ,by="fc_obsrvcion", all=TRUE)
merge12<-merge(Df2.22, Df2.29 ,by="fc_obsrvcion", all=TRUE)
merge13<-merge(Df2.23, Df2.30 ,by="fc_obsrvcion", all=TRUE)
merge14<-merge(Df2.24, Df2.31 ,by="fc_obsrvcion", all=TRUE)
merge15<-merge(Df2.25, Df2.32 ,by="fc_obsrvcion", all=TRUE)
merge16<-merge(Df2.26, Df2.33 ,by="fc_obsrvcion", all=TRUE)
merge17<-merge(Df2.27, Df2.34 ,by="fc_obsrvcion", all=TRUE)

```

```

require(plyr)
require(dostats)

```

```
getwd()
```

```
PATH_LOCAL_LAPTOP="C:/DatosProyectos/"
```

```
PATH_LOCALSERVER="C:/Users/Administrator/Documents/"
```

```
PATH_DATA="TESIS-Datos/EstacionesValle/01Base/"
```

```
PATH_RESUMEN="TESIS-Datos/EstacionesValle/02Base/"
```

```
PATH_tmp_mnma="TESIS-Datos/EstacionesValle/tmp_mnma/"
```

```
PATH_VALLEDATAFILES=paste(PATH_LOCAL_LAPTOP,
PATH_DATA,sep=" ",collapse=NULL)
```

```
PATH_VALLERESUMEFILES=paste(PATH_LOCAL_LAPTOP,
PATH_RESUMEN,sep=" ",collapse=NULL)
```

```
setwd(PATH_VALLEDATAFILES)
```

```
getwd()
```

```
LIST_ESTACIONES <- list.files(pattern = ".csv")
```

```
###Dataframe
```

```
for (i in seq_along(LIST_ESTACIONES)) {
```

```
  DFVar01= paste("Df",i, sep = ".")
```

```
  assign(DFVar01, read.csv(LIST_ESTACIONES[i]))
```

```
}
```

```
=====
```

```
=
```

```
for(i in seq_along(LIST_ESTACIONES)){
```

```
  DFVar01= paste("Df" , i, sep = ".")
```

```

DFVar02= paste("Df" , i, sep = "" )
DFVar05<-paste(DFVar01, "$", "fc_obsrvcion[]", sep = "")
DFVar06<-paste(DFVar01, "$", "nm_ema[]", sep = "")
# DFVar02= data.frame(DFVar05,DFVar06)

DFVar02 <- paste(DFVar01,"[c("nm_ema","fc_obsrvcion")]") , sep="" )
colnames(mat.m) <- c("data","tmax","event","day")
matf.m[[i]] <- mat.m

}

row.names(aav)
# Programa = 04_01QCDiarias.R
# Autor = Victor Hugo Patiño Bravo
# Modificado = Jorge Cardona
#Nota = Este programa selecciona QC para información a nivel diario, hallan o no
pasado por proceso de conversión a
# escala diaria.
## Componente 2
##
## i=0
####Cargar paquetes
library(chron)
library(stringr)

####Lectura de los archivos
# *****Definicion de RUTAS de datos de entrada y resultados*****
ruta="C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/ejemplo2/"
Destino="C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script1/"

# ***** Identifica archivos seleccionados
files <- list.files(ruta, pattern="\\.csv$")

# *** Carga los archivos seleccionados
nom.files<-substring(files,1,nchar(files)-4)
Data.all.files <- lapply(paste(ruta,"/",files,sep="",dec="."),function(x){read.csv(x)})
# ***Asigna nombres a las series
names(Data.all.files)=nom.files

Data.all.filesNAFree=Data.all.files
Serie.diaria=Data.all.files
getwd()
#
# Tabla de valores de referencia
#REF=read.csv("//dapadfs/workspace_cluster_6/TRANSVERSAL_PROJECTS/MADR/CO
MPONENTE_2/CLIMA/QC_scripts_R/Valores_REF_QCDiario.csv", header=T,

```

```

row.names=1)
REF=read.csv("C:/DatosProyectos/TESIS/03R-
Scripts/scriptVH/ejemplo/Valores_REF_QCDiario.csv", header=T, row.names=1)

head(Data.all.files[[1]][4]) # Muestra lista1 campo4 "Date"
setwd("C:/DatosProyectos/TESIS/03R-Scripts/scriptVH/Results_script1")
#i=i+1
for(i in 1:length(Data.all.files)){
  ProbdUP=FALSE
  Station=Data.all.files[[i]]
  Station.Data.all.filesNAFree = Data.all.files[[i]]
  #Data.all.filesNAFree=Data.all.files[[i]]
  TOP=length(Station)-1
  j=4
  while (j <TOP){
    j=j+1
    #Data.all.filesNAFree=NA
NAMEFIELD=NA;DateOK=NA;VAR=NA;Bonnes1=NULL;TMP_A=NA;TMP_B=NA;MI
NIMO=NA;MAXIMO=NA
    TMP_C=NA;Dev=NA;PROMEDIO=NA;PLOT=NA

Data.all.filesNAFree[[i]]=Data.all.files[[i]][which(!is.na(Data.all.files[[i]][[j]])),]
  NAMEFIELD=names(Data.all.files[[i]][j])

  DateOK=as.Date(as.character(Data.all.filesNAFree[[i]]$Date), "%m/%d/%Y") #Formato
de fecha SIN separadores
  Data.all.filesNAFree[[i]]$Date=DateOK

  VAR=substring(nom.files[i],nchar(nom.files[i])-3, nchar(nom.files[i]))
  VAR=substring(nom.files[i],nchar(nom.files[i])-3, nchar(nom.files[i]))
VAR=toupper(VAR)

  # Detectar valores fuera de rango

  TMP_A=paste0("REF$",VAR,"[2]")
  TMP_B=paste0("REF$",VAR,"[1]")
  MINIMO=eval(parse(text=TMP_A))
  MAXIMO=eval(parse(text=TMP_B))
TMP_C=paste0("subset(Data.all.filesNAFree[[i]],Data.all.filesNAFree[[i]][j]>=MINIMO &
Data.all.filesNAFree[[i]][j]<=MAXIMO)")
Bonnes1=eval(parse(text=TMP_C))
  # Detectar valores fuera de 4 dev
Dev=sd(Bonnes1[[j]]) #colnames(Bonnes1[[j]]) ; head(Bonnes1[[j]]);names(Bonnes1)
PROMEDIO=mean(Bonnes1[[j]])

# camponuevo=paste(names(Bonnes1[j]),"CUATRODEV", sep="_")

```

```

# Bonnes1=Bonnes1[c("hole")]
# Bonnes1=Bonnes1[c(camponuevo)]
Bonnes1$CUATRODEV=NA
Bonnes1$CUATRODEV[(Bonnes1[[j]]>(PROMEDIO+4*Dev)|Bonnes1[[j]]<(PROMEDIO-
4*Dev))]=c("Fuera de 4DEV")

# Deteccion rango de fechas
Date.min=min(Bonnes1$Date)
Date.max=max(Bonnes1$Date)

##### Construir serie resultante
SEQ=NULL;POSlimpio=NULL #
SEQ=as.data.frame(seq(Date.min,Date.max,1)) # names(SEQ) # crea vector con fecha
final e inicial
colnames(SEQ)=c("Date") # Le da nombre a la columna

POSpot=match(SEQ$Date, Bonnes1$Date) # Crea vector con los encuentros de la fecha
POSlimpio=POSpot[!is.na(POSpot)] #
Bonnes3=Bonnes1[POSlimpio,]
POS=match(Bonnes3$Date,SEQ$Date)

TMP_A=paste0("SEQ$",NAMEFIELD,"=NA"); MINIMO=eval(parse(text=TMP_A)) #
crea los nuevo campos
SEQ$CUATRODEV=NA # names(SEQ)

SEQ[[2]][POS]=Bonnes3[[j]] # head(Bonnes3[[j]]) head(SEQ[[2]][POS])
SEQ$CUATRODEV[POS]=Bonnes3$CUATRODEV
Serie.diaria[[i]]=SEQ # names(Serie.diaria)

# Generar reporte
size=dim(Data.all.files[[i]])[1]
Good=dim(Bonnes3)[1]
Duplicados=dim(Bonnes1)[1]-Good
Error=dim(Data.all.filesNAFree[[i]])[1]-(Good+Duplicados)
NAentrada=size-(dim(Data.all.filesNAFree[[i]])[1])
POSdecal=POS[-1]
POSlimpio.small=POS[1:length(POSdecal)]
MAYORSalto=max(POSdecal-POSlimpio.small)

Fuera4DEV=sum(!is.na(Bonnes3$CUATRODEV))

resul=c(size, Duplicados, Duplicados/size, Error, Error/size, Fuera4DEV, NAentrada,
NAentrada/size, ProbDUP, MAYORSalto)
Tabla.fin=as.data.frame(resul,row.names=c("Numero de datos totales", "Numero de
Duplicados", "% de duplicados", "Errores", "% errores", "Numero de valores fuera de 4

```

```

DEV", "NA en entrada", "% de NA en entrada", "Problema de duplicados con valores
distintos", "Mayor hueco"))
  write.csv(Tabla.fin,paste0(Destino,nom.files[i],NAMEFIELD ,"_ResumenQCdiario.csv"))
  write.table(Serie.diaria[[i]],paste0(Destino,nom.files[i],NAMEFIELD
, "_ReadyPaRelleno.txt"), sep="\t", row.names=FALSE)

jpeg(paste0(Destino,nom.files[i],NAMEFIELD , "_Plotserie.jpg"), width=600, height=400,
units="px")
#####**names(Serie.diaria[[i]])
  YLAB="Value";XLAB="Date"; TYPE="l" ; MAIN=paste("Estacion ",NAMEFIELD,"
Datos de ",VAR)
  TMP_C=paste0("plot(Serie.diaria[[i]]$",NAMEFIELD,
"~Serie.diaria[[i]]$Date,main=MAIN, ylab=YLAB,xlab=XLAB, type=TYPE)" )

  PLOT=eval(parse(text=TMP_C))

  dev.off()

  jpeg(paste0(Destino,nom.files[i],NAMEFIELD , "_Boxplotserie.jpg"), width=600,
height=400, units="px")
  TMP_C=paste0("boxplot(Serie.diaria[[i]]$",NAMEFIELD,"main=MAIN)" )
  PLOT=eval(parse(text=TMP_C))
#####**boxplot(Serie.diaria[[i]]$aeropuerto)
dev.off()
# j
# NAMEFIELD
} #end While
} #end for

# Programa = 04_02Coher_TmaxTmin.R
# Autor = Victor Hugo Patiño Bravo
# Modificado = Jorge Cardona
# Fec Actualziacion = 2014-0501
# Nota = Este programa selecciona QC para información a nivel diario,
# hallan o no pasado por proceso de conversión a escala
# diaria. Estaciones con información desagregada o un archivo por variable
## TMAX>TMIN, DIARIA
## IMPORTANTE: en ocasiones una misma estación no cuenta

rm(list=ls())

#####Cargar paquetes
library(chron)
library(stringr)

#rutOrigen="./ejemplo"
rutOrigen="C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script1"

```

```

destino=paste0(rutOrigen)
destino2="C:/DatosProyectos/TESIS/03R-Scpts/scriptVH/Result_script2/"
setwd(destino2)
files <- list.files(rutOrigen, pattern="\\.txt$")

nom.files<-substring(files,1,nchar(files)-4)#
fil=files[grep("ReadyPaRelleno",nom.files)]#
files=fil[grep("TMAX|TMIN",fil)]#

nom.files<-substring(files,1,nchar(files)-19)
Datos <-
lapply(paste(rutOrigen,"/",files,sep=""),function(x){read.table(x,header=T,sep="\t")})
names(Datos)=nom.files

#i=1
for(i in 1:length(Datos)){
#si esta condicion se cumple quiere decir que la fecha original viene con separadores
if(class(Datos[[i]][,1])=="factor"){
Datos[[i]][,1]=as.Date(Datos[[i]][,1])#,format="%d/%m/%Y") #El [i][1] es el campo Date
}
}

#Cuales de las estaciones contienen info sobre Tmax y Tmin
TempMax=Datos[grep("TMAX",nom.files)] # names(TempMax[1])
TempMin=Datos[grep("TMIN",nom.files)]
#ID's de las estaciones de cada variable
#IDtmax=substring(names(TempMax),1,nchar(names(TempMax))-5) #IDtmax[1]
#IDtmin=substring(names(TempMin),1,nchar(names(TempMin))-5)

IDtmax=substring(names(TempMax),19,nchar(names(TempMax))-0) #IDtmax[1]
IDtmin=substring(names(TempMin),19,nchar(names(TempMin))-0)
cbind(IDtmin)
vec=(IDtmax==IDtmin)
vec1=length(vec[vec==FALSE])

#Para hacer la comparacion correctamente de todas las estaciones debe haber igual cantidad
#de estaciones, y que sean efectivamente las mismas estaciones entre Tmax y Tmin
if(length(TempMax)==length(TempMin) & vec1==0){
print("Puede realizarse la comparacion")
}else{print("Error, una o más estaciones no coinciden")}

#colnames(TempMax[[1]])=c("Date","Doc","Tres")
#TempMax$Valle19972012_TMAXaeropuerto$Date
#TempMax[[1]][,1]

```



```

cant.error=0
percent=0
#i=0
#i=i+1
for(i in 1:max(length(TempMax),length(TempMin))){
#Posicion que ocupa la estacion de Tmax entre las estaciones de Tmin
pos=grep(IDtmax[i],names(TempMin))
#Fecha mínima y máxima entre las dos bases de datos

Datostemin=min(min(TempMax[[i]][,1]),min(TempMin[[pos]][,1]))
Datemax=max(max(TempMax[[i]][,1]),max(TempMin[[pos]][,1]))
#Dates=seq(Datemin,Datemax,1)
Dates=seq(Datostemin,Datemax,1)

newTmax=0
for(j in 1:length(Dates)) {
  datasub=TempMax[[i]][TempMax[[i]][,1]==Dates[j],] #TempMax[[i]][,1]
  if(dim(datasub)[1]!=0){
    newTmax[j]=TempMax[[i]][TempMax[[i]][,1]==Dates[j],2]
  }else{
    newTmax[j]=NA
  }
}

newTmin=0
for(j in 1:length(Dates)) {
  data_sub=TempMin[[pos]][TempMin[[pos]][,1]==Dates[j],]
  if(dim(data_sub)[1]!=0){
    newTmin[j]=TempMin[[i]][TempMin[[pos]][,1]==Dates[j], 2]
  }else{
    newTmin[j]=NA
  }
}

Station.Unit=data.frame(Dates,newTmax,newTmin)
write.csv(Station.Unit,paste0(destino2,IDtmax[i],"_unificado.csv"))

Station.Unit.Error=which(Station.Unit$newTmax<=Station.Unit$newTmin)
originalTMAX=paste0(IDtmax[i],"_TMAX")
originalTMIN=paste0(IDtmax[i],"_TMIN")
#Reemplazando los valores de las inconsistencias en las fechas de las estaciones involucradas
if(length(Station.Unit.Error)>0){
  originalTMAX=paste0("Valle19972012_TMAX",IDtmax[i])
  originalTMIN=paste0("Valle19972012_TMIN",IDtmax[i])
Datos[[originalTMAX]][Station.Unit.Error,2]=NA #names(Datos[4])
names(Datos[["Valle19972012_TMAXbdelpalo"]])
Datos[[originalTMIN]][Station.Unit.Error,2]=NA

```

```

##Reescribiendo con las correcciones
write.table(Datos[[originalTMAX]],paste0(destino2,originalTMAX,"_02.txt"),sep="\t",row.names=F)

write.table(Datos[[originalTMIN]],paste0(destino2,originalTMIN,"_02.txt"),sep="\t",row.names=F)
}
  cant.error[i]=length(Station.Unit.Error)
  percent[i]=round((cant.error[i]/dim(Station.Unit)[1])*100,3)
} # end for i

cant.err=paste(percent,"%")
cant.errorf=data.frame(IDtmax,cant.error,cant.err)

write.csv(cant.errorf,paste0(destino2,"Coherencia,Tmax_Tmin.csv"),row.names=F)
# Programa      = 04_02Coher_TmaxTmin.R
# Autor         = Victor Hugo Patiño Bravo
# Modificado    = Jorge Cardona
# Fec Actualziacion = 2014-0501
# Nota         = Este programa selecciona ## Rmawgen necesita que los archivos de entrada
tengan un formato especial
##            con este script se logra dicha estructura.

require(chron)
library(chron)

rutOrigen=("C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script1")
rutDestin=("C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script3")

#Input=function(h){
#rutOrigen=paste0(rutAOrigen,h,"/")

files <-list.files(rutOrigen,pattern="\\.txt$")

nom.files<-substring(files,1,nchar(files)-4)
files=files[grep("ReadyPaRelleno",nom.files)]

nom.files<-substring(files,1,nchar(files)-19)
Datos <-
lapply(paste(rutOrigen,"/",files,sep=""),function(x){read.table(x,header=T,sep="\t")})
names(Datos)=nom.files

for(i in 1:length(Datos)){
#si esta condicion se cumple quiere decir que la fecha original viene con separadores

```

```

if(class(Datos[[i]][,1])=="factor"){Datos[[i]][,1]=as.Date(Datos[[i]][,1])} #para reconocer
las fechas
  Datos[[i]][,2]=as.numeric(Datos[[i]][,2]) #Para garantizar que existan
datos numericos
}
#Cuales de las estaciones contienen info sobre Tmax, Tmin y Rain
TempMax=Datos[grep("TMAX",nom.files)]
TempMin=Datos[grep("TMIN",nom.files)]
Rain =Datos[grep("RAIN",nom.files)]

#ID's de las estaciones involucradas en cada variable
IDtmax=paste0( substring(names(TempMax),19,nchar(names(TempMax))-0) , "")
IDtmin=paste0( substring(names(TempMin),19,nchar(names(TempMin))-0) , "")
IDrain=paste0( substring(names(Rain),19,nchar(names(Rain))-0) , "")

date.minTmax=0
date.maxTmax=0
for(i in 1:length(TempMax)){
  date.minTmax[i]=as.Date(min(TempMax[[i]][,1]))
  date.maxTmax[i]=as.Date(max(TempMax[[i]][,1]))
}
date.minTmax=as.Date(chron(min(date.minTmax),format="y-m-d"))
date.maxTmax=as.Date(chron(max(date.maxTmax),format="y-m-d"))

DatesTmax=seq(date.minTmax,date.maxTmax,1)
tmax_to=matrix(NA,nrow=length(DatesTmax),ncol=length(TempMax),byrow=F)

for(j in 1:length(TempMax)){
  for(i in 1:length(DatesTmax)) {
    data_sub=TempMax[[j]][TempMax[[j]][,1]==DatesTmax[i],]
    if(dim(data_sub)[1]!=0){
      tmax_to[i,j]=TempMax[[j]][TempMax[[j]][,1]==DatesTmax[i],2]
    }else{
      tmax_to[i,j]=NA
    }
  }
}
year=as.numeric(substr(DatesTmax,1,4))
month=as.numeric(substr(DatesTmax,6,7))
day=as.numeric(substr(DatesTmax,9,10))

tmaxtofin=data.frame(month,day,year,tmax_to)
colnames(tmaxtofin)=c("month","day","year",IDtmax)
write.csv(tmaxtofin,paste0(rutDestin,"/","tmax_to",".csv"),row.names=F)

#####para Tmin
date.minTmin=0

```

```

date.maxTmin=0
for(i in 1:length(TempMin)){
  date.minTmin[i]=as.Date(min(TempMin[[i]][,1]))
  date.maxTmin[i]=as.Date(max(TempMin[[i]][,1]))
}
date.minTmin=as.Date(chron(min(date.minTmin),format="y-m-d"))
date.maxTmin=as.Date(chron(max(date.maxTmin),format="y-m-d"))

DatesTmin=seq(date.minTmin,date.maxTmin,1)
tmin_to=matrix(NA,nrow=length(DatesTmin),ncol=length(TempMin),byrow=F)

for(j in 1:length(TempMin)){
  for(i in 1:length(DatesTmin)) {
    data_sub=TempMin[[j]][TempMin[[j]][,1]==DatesTmin[i],]
    if(dim(data_sub)[1]!=0){
      tmin_to[i,j]=TempMin[[j]][TempMin[[j]][,1]==DatesTmin[i],2]
    }else{
      tmin_to[i,j]=NA
    }
  }
}
year=as.numeric(substr(DatesTmin,1,4))
month=as.numeric(substr(DatesTmin,6,7))
day=as.numeric(substr(DatesTmin,9,10))

tmintofin=data.frame(month,day,year,tmin_to)
colnames(tmintofin)=c("month","day","year",IDtmin)
write.csv(tmintofin,paste0(rutDestin,"/", "tmin_to", ".csv"),row.names=F)

#Rain
date.minRain=0
date.maxRain=0
for(i in 1:length(Rain)){
  date.minRain[i]=as.Date(min(Rain[[i]][,1]))
  date.maxRain[i]=as.Date(max(Rain[[i]][,1]))
}
date.minRain=as.Date(chron(min(date.minRain),format="y-m-d"))
date.maxRain=as.Date(chron(max(date.maxRain),format="y-m-d"))

DatesRain=seq(date.minRain,date.maxRain,1)
prec_to=matrix(NA,nrow=length(DatesRain),ncol=length(Rain),byrow=F)

for(j in 1:length(Rain)){
  for(i in 1:length(DatesRain)) {
    data_sub=Rain[[j]][Rain[[j]][,1]==DatesRain[i],]
    if(dim(data_sub)[1]!=0){

```

```

    prec_to[i,j]=Rain[[j]][Rain[[j]][,1]==DatesRain[i],2]
  }else{
    prec_to[i,j]=NA
  }
}
}
year=as.numeric(substr(DatesRain,1,4))
month=as.numeric(substr(DatesRain,6,7))
day=as.numeric(substr(DatesRain,9,10))

prectofin=data.frame(month,day,year,prec_to)
colnames(prectofin)=c("month","day","year",IDrain)
write.csv(prectofin,paste0(rutDestin,"/", "prec_to", ".csv"),row.names=F)

#print(paste0("Estimado ",h," listo"))

#}

#for (h in 1:1000){
# tryCatch({
#   Input(h)
#print(paste0("simulación ",z," existosa"))
# }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")}
# )
#}

# Programa      = 04_04GenerarDatosRMAWGEN.R
# Autor         = Victor Hugo Patiño Bravo
# Modificado    = Jorge Cardona
# Fec Actualziacion = 2014-0501
# Nota         = Este programa selecciona

#####
#####
##### ## ## ### ## ## ##### ##### ## ##
## ## ### ### ## ## ## ## ## ## ## ## ## ## ##
## ## ##### ##### ## ## ## ## ## ## ## ## ## ##
##### ## ### ## ## ## ## ## ## ## ## ## ## ## ## ## ##
## ## ## ## ##### ## ## ## ## ## ## ## ## ## ##
## ## ## ## ## ## ## ## ## ## ## ## ## ## ## ##
## ## ## ## ## ## ## ## ## ## ## ## ## ## ## ##
#####
#####

rm(list=ls())
library(RMAWGEN)
fix(ComprehensivePrecipitationGenerator)
#ojo jorge

```

```

#rutOrigen2=setwd("//dapadfs/workspace_cluster_6/TRANSVERSAL_PROJECTS/MADR/
COMPONENTE_2/CLIMA/SERIES_DIARIAS_SIN_LLENAR/TallerFed/")
#rutOrigen2=setwd("//dapadfs/workspace_cluster_6/TRANSVERSAL_PROJECTS/MADR/
COMPONENTE_2/CLIMA/SERIES_DIARIAS_SIN_LLENAR/TallerFed/")
#rutDestino2="//dapadfs/workspace_cluster_6/TRANSVERSAL_PROJECTS/MADR/COM
PONENTE_2/CLIMA/SERIES_DIARIAS_RELLENAS/TallerFed/2010-2013/"

rutOrigen2=setwd("C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script4/")
rutOrigen2=setwd("C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script4/")
rutDestino2= ("C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script4/")

rutOrigen=("C:/DatosProyectos/TESIS/03R-Scrpts/scriptVH/Results_script3/")

#IMPUTACION<-function(h){

#rutOrigen2=setwd(paste0(rutOrigen,"/",h))###No borrar alguna de estas dos lineas repetidas
#rutOrigen2=setwd(paste0(rutOrigen,"/",h))###Debido a un inconveniente, se coloca dos
veces la especificación de la ruta para que realmente sea reconocida
#rutDestino2=(paste0(rutDestino,h,"/"))

year_max <- 2012
year_min <- 1997
origin <- paste0(year_min,"-1-1")
valmin <- 1.0

n_GPCA_iter <- 10
n_GPCA_iteration_residuals <- 10
n_GPCA_iter_prec <- 20
n_GPCA_iteration_residuals_prec <- 20
ttmaxvn=paste0(rutOrigen,"tmax_to.csv")
ttminn=paste0(rutOrigen,"tmin_to.csv")
pprecc=paste0(rutOrigen,"prec_to.csv")

tmax <- read.csv(ttmaxvn, header = TRUE, sep = ",", dec=".")
tmin <- read.csv(ttminn, header = TRUE, sep = ",", dec=".")
preci<- read.csv(pprecc, header = TRUE, sep = ",", dec=".")

tmax_1=subset(tmax,tmax$year>=year_min & tmax$year<=year_max)
tmin_1=subset(tmin,tmin$year>=year_min & tmin$year<=year_max)
precipitation=subset(preci,preci$year>=year_min & preci$year<=year_max)

station <- names(tmax_1)[-1:3]

stationUSE=station[c(1:28)]

summary(tmax_1[,4:ncol(tmax_1)]);summary(tmin_1[,4:ncol(tmin_1)]);summary(precipitati
on[,4:ncol(precipitation)])

```

```

dim(tmax_1);dim(tmin_1);dim(precipitation)

# generation of temperature max and min
generation00_temp <- ComprehensiveTemperatureGenerator(station=stationUSE,
              Tx_all=tmax_1,
              Tn_all=tmin_1,
              year_min=year_min,
              year_max=year_max,
p=1,n_GPCA_iteration=n_GPCA_iter,
n_GPCA_iteration_residuals=n_GPCA_iteration_residuals,
              sample="monthly")

#Use of measured and observed temperature as exogenous variables
exogen_sim <-
cbind(generation00_temp$output$Tx_gen,generation00_temp$output$Tn_gen)
names(exogen_sim) <-
cbind(paste(names(generation00_temp$output$Tx_gen),"_Tx",sep=""),paste(names(generati
on00_temp$output$Tn_gen),"_Tn",sep=""))
exogen <- cbind(generation00_temp$input$Tx_mes,generation00_temp$input$Tn_mes)
names(exogen) <-
cbind(paste(names(generation00_temp$input$Tx_mes),"_Tx",sep=""),paste(names(generatio
n00_temp$input$Tn_mes),"_Tn",sep=""))

# Precipitation Generator (temperture enters as exogenous variable)
generation00_prec <- ComprehensivePrecipitationGenerator(station=stationUSE,
              prec_all=precipitation,
              year_min=year_min,
              year_max=year_max,
              exogen=exogen,
              exogen_sim=exogen_sim,
              p=1,n_GPCA_iteration=n_GPCA_iter_prec,

n_GPCA_iteration_residuals=n_GPCA_iteration_residuals_prec,
              sample="monthly",valmin=1)

#-----Gerar Arquivos-----
prec_mes <- generation00_prec$prec_mes
prec_gen <- generation00_prec$prec_gen

tmin_mes <-generation00_temp$input$Tn_mes
tmin_gen <-generation00_temp$out$Tn_gen

tmax_mes <-generation00_temp$input$Tx_mes
tmax_gen <-generation00_temp$out$Tx_gen

test = names(prec_gen)
teste_min <- names(tmin_gen)

```

```

teste_max <- names(tmax_gen)

data_genPrec <- extractmonths(data=generation00_prec$prec_gen,when=c("Jan", "Feb",
"Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"),origin)
data_genTmin <- extractmonths(data=generation00_temp$out$Tn_gen,when=c("Jan", "Feb",
"Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"),origin)
data_genTmax <- extractmonths(data=generation00_temp$out$Tx_gen,when=c("Jan",
"Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"),origin)

#-----Salvando Arquivo-----
archPrec=list(0)
archTmax=list(0)
archTmin=list(0)
  archPrec2=list(0)
  archTmax2=list(0)
  archTmin2=list(0)
precipitation_Comp=precipitation[stationUSE]
tmax_Comp=tmax_1[stationUSE]
tmin_Comp=tmin_1[stationUSE]

  for(i in 1:length(stationUSE)){

    Modif=rep("NO",dim(tmax_1)[1])
    Modif1=rep("NO",dim(tmax_1)[1])
    Modif2=rep("NO",dim(tmax_1)[1])

    posPrec=which(is.na(precipitation[stationUSE[i]]))
    precipitation_Comp[[i]][posPrec]=data_genPrec[[i]][posPrec]#Fusion
    Modif[posPrec]="SI"
    archPrec[[i]]=cbind(precipitation[,1:3],precipitation_Comp[[i]],Modif)#para llevar las
    fuciones con indicador a varios txt

    postmax=which(is.na(tmax_1[stationUSE[i]]))
    tmax_Comp[[i]][postmax]=data_genTmax[[i]][postmax]
    Modif1[postmax]="SI"
    archTmax[[i]]=cbind(tmax_1[,1:3],tmax_Comp[[i]],Modif1)

    postmin=which(is.na(tmin_1[stationUSE[i]]))
    tmin_Comp[[i]][postmin]=data_genTmin[[i]][postmin]
    Modif2[postmin]="SI"
    archTmin[[i]]=cbind(tmin_1[,1:3],tmin_Comp[[i]],Modif2)

    datPrec=as.Date(paste0(archPrec[[i]][,1],"/",
    archPrec[[i]][,2],"/",archPrec[[i]][,3]),format="%m/%d/%Y")
    archPrec2[[i]] =data.frame(datPrec,archPrec[[i]][4:5]);
    colnames(archPrec[[i]])=c("Date", "Value", "Modificado")
    data_genPrec[[i]]=data.frame(datPrec,data_genPrec[[i]]);
  }

```



```

colnames(data_genPrec[[i]])=c("Date","Value")

  datTmax=as.Date(paste0(archTmax[[i]][,1],"/",
archTmax[[i]][,2],"/",archTmax[[i]][,3]),format="%m/%d/%Y")
  archTmax2[[i]] =data.frame(datTmax,archTmax[[i]][4:5]);
colnames(archTmax[[i]])=c("Date","Value","Modificado")
  data_genTmax[[i]]=data.frame(datTmax,data_genTmax[[i]]);
colnames(data_genTmax[[i]])=c("Date","Value")

  datTmin=as.Date(paste0(archTmin[[i]][,1],"/",
archTmin[[i]][,2],"/",archTmin[[i]][,3]),format="%m/%d/%Y")
  archTmin2[[i]] =data.frame(datTmin,archTmin[[i]][4:5]);
colnames(archTmin[[i]])=c("Date","Value","Modificado")
  data_genTmin[[i]]=data.frame(datTmin,data_genTmin[[i]]);
colnames(data_genTmin[[i]])=c("Date","Value")

  write.table(archPrec2[[i]],paste0(rutDestino2,test[i],"_RAIN_FUS.txt"), sep = "\t",
row.names = F,col.names = TRUE)
  write.table(archTmax2[[i]],paste0(rutDestino2,test[i],"_TMAX_FUS.txt"), sep = "\t",
row.names = F,col.names = TRUE)
  write.table(archTmin2[[i]],paste0(rutDestino2,test[i],"_TMIN_FUS.txt"), sep = "\t",
row.names = F,col.names = TRUE)

  write.table(data_genPrec[[i]],paste0(rutDestino2,test[i],"_RAIN_EST.txt"), sep = "\t",
row.names = F,col.names = TRUE)
  write.table(data_genTmax[[i]],paste0(rutDestino2,test[i],"_TMAX_EST.txt"), sep = "\t",
row.names = F,col.names = TRUE)
  write.table(data_genTmin[[i]],paste0(rutDestino2,test[i],"_TMIN_EST.txt"), sep = "\t",
row.names = F,col.names = TRUE)
}
#####Aqui guardo los *.csv de las fuciones, donde se conservan los valores originales

precipitacionCSV=do.call(cbind,precipitation_Comp);tempMaxi=do.call(cbind,tmax_Comp);
tempMini=do.call(cbind,tmin_Comp)

precipitacionCSV <- data.frame(precipitation[,1:3],precipitacionCSV)
tempMaxi <- data.frame(tmax_1[,1:3],tempMaxi)
tempMini <- data.frame(tmin_1[,1:3],tempMini)

write.csv(tempMini,paste0(rutDestino2,"tmin_to_FUS.csv"), row.names = F)
write.csv(tempMaxi,paste0(rutDestino2,"tmax_to_FUS.csv"), row.names = F)
write.csv(precipitacionCSV,paste0(rutDestino2,"prec_to_FUS.csv"), row.names = F)

#####
# for(i in 1:length(stationUSE)){
#   jpeg(paste0(rutDestino2,"Images/", "prec_",stationUSE[i],"_RAIN_2007-

```

```

2012.jpeg",sep=""),width = 1150, height = 500)
#
plot(1:dim(data_genPrec)[1],archPrec[[i]][,2],type="l",col="blue",lwd=1.5,main=paste("prec
_",stationUSE[i]),ylim=c(0,100))
#   lines(1:dim(data_genPrec)[1],precipitation[stationUSE[i]][,1],col="red",lwd=1.5,lty=2)
#   legend("topright",c("generada","original"),lwd=c(1.5,1.5),col=c("blue","red"))
#   dev.off()

#   jpeg(paste0(rutDestino2,"Images/","tmax_",stationUSE[i],"_TMAX_2007-
2012.jpeg",sep=""),width = 1150, height = 500)
#
plot(1:dim(data_genTmax)[1],archTmax[[i]][,2],type="l",col="blue",lwd=1.5,main=paste("t
max_",stationUSE[i]),ylim=c(22,40))
#   lines(1:dim(data_genTmax)[1],tmax_1[stationUSE[i]][,1],col="red",lwd=1.5,lty=2)
#   legend("topright",c("generada","original"),lwd=c(1.5,1.5),col=c("blue","red"))
#   dev.off()

#   jpeg(paste0(rutDestino2,"Images/","tmin_",stationUSE[i],"_TMIN_2007-
2012.jpeg",sep=""),width = 1150, height = 500)
#
plot(1:dim(data_genTmin)[1],archTmin[[i]][,2],type="l",col="blue",lwd=1.5,main=paste("tm
in_",stationUSE[i]),ylim=c(15,30))
#   lines(1:dim(data_genTmin)[1],tmin_1[stationUSE[i]][,1],col="red",lwd=1.5,lty=2)
#   legend("topright",c("generada","original"),lwd=c(1.5,1.5),col=c("blue","red"))
#   dev.off()
# }

#}

#for (z in 1:1000){
# tryCatch({
# IMPUTACION(z)
# print(paste0("simulación ",z," existosa"))
# }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")}
# )
#}

# Programa      = 08_1PRECCumulative.R
# Autor         = Jorge CArdon
# Actualziacion = 2014-0501
# Nota         =
# Programas de tres pasos 1
#PASO uNO(1)----- FUNCIONO jcardona
#Carga el archivo de estaciones con las series completas por el RMWAGEN
# y genera las mismas usando una funcion cumulativa por ano lo grafica y
#salva a un archivos
rm(list=ls()) # Elimina variables
#Carga los datos de los resultados de RMWAGEN

```

```

PREC=read.csv("C:/DatosProyectos/TESIS/03R-
Scripts/scriptVH/Results_script4/04prec_to_FUS.csv")
names(PREC)
length(PREC) #31 variables
length(PREC$year) #5844 registros
#rec_resum= data.frame(years=rep(1997:2012), x=rep(1:34))
#YEAR=rep(1997:2012)
#Crea el vector de 5844 registros
STATION=rep(1:length(PREC$year))
rec_resum=data.frame(year=STATION) #crea DF con 5844 rows
#i=5
#crea
jahre <- factor(PREC$year) #crea vector al extraer el factor de los años, unicos anos, no repite
#hacer un dataframe de
#ano, 28 variable =25 columnas
#copiar el año y luego cada resultado del for
rec_resum["year"]=NA
rec_resum["year"]=PREC["year"]
rec_resum["month"]=NA
rec_resum["month"]=PREC["month"]
rec_resum["day"]=NA
rec_resum["day"]=PREC["day"]
names(rec_resum)
for(i in 4:length(PREC)){ #(31-4) 28 variables, excluye los 3 primeros
  COLSNAME= colnames(PREC[i][1]) # le da el nombre de la columna
  rec_resum[COLSNAME]=NA #crea la nueva columna con el nombre
  tpks1 <- unlist(tapply(PREC[,i], jahre, cumsum))
  # VAR=paste0( COLSNAME)
  rec_resum[[COLSNAME]]=tpks1 #copia el nuevo vector a data frame
  plot(tpks1, type="l", xlab="Anos", ylab="Acumulado anual pitación",
  main=colnames(PREC[i][1]))
}
names(rec_resum)
#getwd()
write.csv(rec_resum ,
"C:/DatosProyectos/TESIS/02DataResum/08PREC_cumsum_5844X31.csv")
setwd()
names(PREC[[1]])
#Paso dos(2) ----- FUNCIONO jcardona
## elaborar una matriz de estaciones por año cada estaciones tiene 16 anos y
# 365 registros
rm(list=ls()) # Elimina variables
PREC=read.csv("C:/DatosProyectos/TESIS/02DataResum/08PREC_cumsum_5844X31.csv"
)
names(PREC) # aparece X al inicio
length(PREC) #32 variables
PREC=PREC[,2:32] # quita el primer campo X

```

```

names(PREC)
length(PREC) #31 variables
YEARS=rep(1997:2012) # lo mismo de YEARS=factor(PREC$year)
#REGISTROS=rep(1:5844) # lo mismo de REGISTROS= length(PREC$year)
#STATIONS=rep(1:32) #lo mismo que length(PREC)
#PREC[PREC$year==YEAR[1],]
rec_resum=data.frame(JULDAYS=rep(1:366))
names(rec_resum)
for(i in 4:length(PREC)){ # 28 columns estaciones variables
  for(j in 1:length(YEARS)) { #16 x 365 registros rows
COLSNAME=paste(colnames(PREC[i]), "_", YEARS[j], sep="") # le da el nombre de la
columna1
  # COLSNAME=colnames(PREC[i]) # le da el nombre de la columna
  rec_resum[COLSNAME]=NA #crea la nueva columna con el nombre
  jahre=PREC[PREC$year==YEARS[j],i] # crea una lista con los registros del año
seleccionado
length(jahre)
  if(length(jahre)==365){
    c(jahre[366], 0)
    jahre[366]= 0
  }
  length(jahre)
  names(jahre)
  rec_resum[[COLSNAME]]=jahre
  }}
#rev(rec_resum)
names(rec_resum)
length(rec_resum)
write.csv(rec_resum ,
"C:/DatosProyectos/TESIS/02DataResum/08PREC_cumsum_anual_366X449.csv")

#Paso tres(3)----- FUNCIONO jcardona
#Carfa el archivo con las series acumuladas por cada año y genera una matriz
#de cada estación por 365 días promedios de series diarias
rm(list=ls()) # Elimina variables
##promedio
PREC=read.csv("C:/DatosProyectos/TESIS/02DataResum/08PREC_cumsum_anual_366X44
9.csv")
length(PREC)
names(PREC)
PREC=PREC[,2:450]
length(PREC)
names(PREC)
AVERAGES=data.frame(DAY=rep(1:365))
K=1
# Este for inicia en el campo #3
for(i in seq(2,length(PREC), 16)){ #450 variables(cadauno 16 años) inicia 3,final 450,salto

```

```

16
K=K+1
COLSNAME=paste(colnames(PREC[i]),sep="") # le da el nombre de la columna
AVERAGES[COLSNAME]=NA
#COLSNAME<-substring(COLSNAME,1,nchar(COLSNAME)-5)
for(j in 1:365) { #16 x 365 registros rows

AVERAGES[j,K]=mean(PREC[j,i],PREC[j,i+1],PREC[j,i+2],PREC[j,i+3],PREC[j,i+4],PRE
C[j,i+5],PREC[j,i+6],PREC[j,i+7],PREC[j,i+8],

PREC[j,i+9],PREC[j,i+10],PREC[j,i+11],PREC[j,i+12],PREC[j,i+13],PREC[j,i+14],PREC[j
,i+15])
}}
names(AVERAGES)
length(AVERAGES)
write.csv(AVERAGES ,
"C:/DatosProyectos/TESIS/02DataResum/09PREC_Cumsum_Mean_365X29.csv")

#----- jcardona
#Hace la comparacionde las series cumulaticas y series suavizadas

rm(list=ls()) # Elimina variables
library(fda)
library(geofd)
#windows()
PREC_julian=read.csv("C:/DatosProyectos/TESIS/02DataResum/09PREC_Cumsum_Mean_
365X29.csv")
length(PREC_julian)
names(PREC_julian)
PREC_julian=PREC_julian[,3:30]
length(PREC_julian) # 28 variualbles
length(PREC_julian$aeropuerto_1997) #365 dias
names(PREC_julian)

#PREC_julian=cbind(tpks1[1:365],tpks1[366:730])
n<-dim(PREC_julian)[1]
argvals<-seq(1,n, by=1)
s<-28
rangeval <- range(argvals)
norder <- 4
nbasis <- 65
bspl.basis <- create.bspline.basis(rangeval, nbasis, norder)
lambda <-0
datafdPar <- fdPar(bspl.basis, Lfdobj=2, lambda)
smfd <- smooth.basis(argvals,as.matrix(PREC_julian),datafdPar)
datafd <- smfd$fd
M <- bsplinepen(bspl.basis,Lfdobj=0)

```

```

windows()
par(mfrow=c(1,2))
matplot(PREC_julian,type="l",xlab="Dia", ylab="Precipitación Maxima Original")
grid()
plot(datafd, lty=1, xlab="Dia", ylab="Precipitación Maxima Suavizada")
grid()

# Programa      = 08_1TMAXCumulative.R
# Autor         = Jorge Cardona
# Actualización = 2014-0501
# Nota          =
# *****TEMPERATURA MAXIMA*****
##Elaborar una matriz de estaciones por año cada estaciones tiene 16 años de registro y
# 365 registros por cada año
rm(list=ls()) # Elimina variables
TMAX=read.csv("C:/DatosProyectos/TESIS/03R-
Scripts/scriptVH/Results_script4/04tmax_to_FUS.csv")
names(TMAX)
length(TMAX) #31 28 estaciones + month,day,year
YEARS=rep(1997:2012) # lo mismo de YEARS=factor(TMAX$year)
#REGISTROS=rep(1:5844) # lo mismo de REGISTROS= length(TMAX$year)
#STATIONS=rep(1:32) #lo mismo que length(TMAX)
#TMAX[TMAX$year==YEAR[1],]
rec_resum=data.frame(JULDAYS=rep(1:366))
names(rec_resum)
for(i in 4:length(TMAX)){ # 28estaciones variables
  for(j in 1:length(YEARS)) { #16 x 365 registros rows
    COLSNAME=paste(colnames(TMAX[i]),"_",YEARS[j],sep="") # le da el nombre de la
columna1
    # COLSNAME=colnames(TMAX[i]) # le da el nombre de la columna
    rec_resum[COLSNAME]=NA #crea la nueva columna con el nombre
    jahre=TMAX[TMAX$year==YEARS[j],i] # crea una lista con los registros del año
seleccionado
    length(jahre)
    if(length(jahre)==365){
      c(jahre[366], 0)
      jahre[366]= 0
    }
    length(jahre)
    names(jahre)
    rec_resum[[COLSNAME]]=jahre
  }
}
#rev(rec_resum)
names(rec_resum)
length(rec_resum) #449
write.csv(rec_resum ,
"C:/DatosProyectos/TESIS/02DataResum/08TMAX_anual_366X449.csv")

```

```

#----- FUNCIONO jcardona
#Carfa el archivo con las series acumuladas por cada año y genera una matriz
#de cada estacion por 365 días promedios de series diarias
rm(list=ls()) # Elimina variables
##promedio
TMAX=read.csv("C:/DatosProyectos/TESIS/02DataResum/08TMAX_anual_366X449.csv")
names(TMAX)
length(TMAX) #450 variables
length(TMAX$JULDAYS) #366 registros
TMAX=TMAX[,2:450]
names(TMAX)
length(TMAX) #433 variables
AVERAGES=data.frame(DAY=rep(1:365))
K=1
# Este for inicia en el campor #3
for(i in seq(2,length(TMAX), 16)){ #450 variables(cadauno 16 años) inicia 3,final 450,salto
16
  K=K+1
  COLSNAME=paste(colnames(TMAX[i]),sep="") # le da el nombre de la columna
AVERAGES[COLSNAME]=NA
  #COLSNAME<-substring(COLSNAME,1,nchar(COLSNAME)-5)
  for(j in 1:365) { #16 x 365 registros rows

AVERAGES[j,K]=mean(TMAX[j,i],TMAX[j,i+1],TMAX[j,i+2],TMAX[j,i+3],TMAX[j,i+4
],TMAX[j,i+5],TMAX[j,i+6],TMAX[j,i+7],TMAX[j,i+8],

TMAX[j,i+9],TMAX[j,i+10],TMAX[j,i+11],TMAX[j,i+12],TMAX[j,i+13],TMAX[j,i+14],T
MAX[j,i+15])
  }}
names(AVERAGES)
length(AVERAGES)
write.csv(AVERAGES ,
"C:/DatosProyectos/TESIS/02DataResum/09TMAX_Mean_365X29.csv")

#----- FUNCIONO jcardona
#Hace la comparacionde las series cumulaticas y series suavizadas
#install.packages("C:/DatosProyectos/Statistical/R/Software/Librerias/geofd_0.4.6.zip",
repos = NULL)
rm(list=ls()) # Elimina variables
library(fda)
library(geofd)
#windows()
TMAX_julian=read.csv("C:/DatosProyectos/TESIS/02DataResum/09TMAX_Mean_365X29
.csv")
names(TMAX_julian)
length(TMAX_julian) #30 variables

```

```

TMAX_julian=TMAX_julian[,3:30]
names(TMAX_julian)
length(TMAX_julian) #28 variables

#TMAX_julian=cbind(tpks1[1:365],tpks1[366:730])
n<-dim(TMAX_julian)[1]
argvals<-seq(1,n, by=1)
s<-28 #cantidad de estaciones o suaviza
rangeval <- range(argvals)
norder <- 4
nbasis <- 65
bspl.basis <- create.bspline.basis(rangeval, nbasis, norder)
lambda <-0
datafdPar <- fdPar(bspl.basis, Lfdobj=2, lambda)
smfd <- smooth.basis(argvals,as.matrix(TMAX_julian),datafdPar)
datafd <- smfd$fd
M <- bsplinepen(bspl.basis,Lfdobj=0)
windows()
par(mfrow=c(1,2))
matplot(TMAX_julian,type="l",xlab="Dia", ylab="Temperatura Maxima Original")
grid()
plot(datafd, lty=1, xlab="Dia", ylab="Temperatura Maxima Suavizada")
grid()

# Programa      = 08_1_TMINCumulative.R
# Autor         = Jorge CArdon
# Actualziacion = 2014-0501
# Nota         =
# *****TEMPERATURA MINIMA*****
##Elaborar una matriz de estaciones por año cada estaciones tiene 16 anos de registro y
# 365 registros por cada año
rm(list=ls()) # Elimina variables
TMIN=read.csv("C:/DatosProyectos/TESIS/03R-
Scripts/scriptVH/Results_script4/04tmin_to_FUS.csv")
YEARS=rep(1997:2012) # lo mismo de YEARS=factor(TMIN$year)
#REGISTROS=rep(1:5844) # lo mismo de REGISTROS= length(TMIN$year)
#STATIONS=rep(1:32) #lo mismo que length(TMIN)
#TMIN[TMIN$year==YEAR[1],]

rec_resum=data.frame(JULDAYS=rep(1:366))
names(rec_resum)
for(i in 5:length(TMIN)){ # 28 columns estaciones variables
  for(j in 1:length(YEARS)) { #16 x 365 registros rows
COLSNAME=paste(colnames(TMIN[i]),"_",YEARS[j],sep="") # le da el nombre de la
columna1
  # COLSNAME=colnames(TMIN[i]) # le da el nombre de la columna
rec_resum[COLSNAME]=NA #crea la nueva columna con el nombre

```



```

    jahre=TMIN[TMIN$year==YEARS[j],i] # crea una lista con los registros del año
seleccionado
length(jahre)
  if(length(jahre)==365){
    c(jahre[366], 0)
    jahre[366]= 0
  }
  length(jahre)
  names(jahre)
  rec_resum[[COLSNAME]]=jahre
}}
#rev(rec_resum)
names(rec_resum)
length(rec_resum)
write.csv(rec_resum ,
"C:/DatosProyectos/TESIS/02DataResum/08TMIN_anual_366X449.csv")

#----- FUNCIONO jcardona
#Carfa el archivo con las series acumuladas por cada año y genera una matriz
#de cada estacion por 365 días promedios de series diarias
rm(list=ls()) # Elimina variables
##promedio
TMIN=read.csv("C:/DatosProyectos/TESIS/02DataResum/08TMIN_anual_366X449.csv")
AVERAGES=data.frame(DAY=rep(1:365))

K=1
# Este for inicia en el campo #3
for(i in seq(3,length(TMIN), 16)){ #450 variables(cadauno 16 años) inicia 3,final 450,salto
16
  K=K+1
  COLSNAME=paste(colnames(TMIN[i]),sep="") # le da el nombre de la columna
AVERAGES[COLSNAME]=NA
  #COLSNAME<-substring(COLSNAME,1,nchar(COLSNAME)-5)
  for(j in 1:365) { #16 x 365 registros rows

AVERAGES[j,K]=mean(TMIN[j,i],TMIN[j,i+1],TMIN[j,i+2],TMIN[j,i+3],TMIN[j,i+4],TM
IN[j,i+5],TMIN[j,i+6],TMIN[j,i+7],TMIN[j,i+8],

TMIN[j,i+9],TMIN[j,i+10],TMIN[j,i+11],TMIN[j,i+12],TMIN[j,i+13],TMIN[j,i+14],TMIN[
j,i+15])
  }
}
names(AVERAGES)
length(AVERAGES)
write.csv(AVERAGES ,
"C:/DatosProyectos/TESIS/02DataResum/09TMIN_Mean_365X29.csv")

#----- FUNCIONO jcardona

```

```

#Hace la comparacionde las series cumulaticas y series suavizadas
#install.packages("C:/DatosProyectos/Statistical/R/Software/Librerias/geofd_0.4.6.zip",
repos = NULL)
rm(list=ls()) # Elimina variables
library(fda)
library(geofd)
#windows()
TMIN_julian=read.csv("C:/DatosProyectos/TESIS/02DataResum/09TMIN_Mean_365X29.c
sv")
#TMIN_julian=cbind(tpks1[1:365],tpks1[366:730])
n<-dim(TMIN_julian)[1]
argvals<-seq(1,n, by=1)
s<-36
rangeval <- range(argvals)
norder <- 4
nbasis <- 65
bspl.basis <- create.bspline.basis(rangeval, nbasis, norder)
lambda <-0
datafdPar <- fdPar(bspl.basis, Lfdobj=2, lambda)
smfd <- smooth.basis(argvals,as.matrix(TMIN_julian),datafdPar)
datafd <- smfd$fd
M <- bsplinepen(bspl.basis,Lfdobj=0)
windows()
par(mfrow=c(1,2))
matplot(TMIN_julian,type="l",xlab="Dia", ylab="Temperatura Minima Original")
grid()
plot(datafd, lty=1, xlab="Dia", ylab="Temperatura Minima Suavizada")
grid()
# Programa      = 10_Modelo_PREC.R
# Autor         = Jorge Cardona
# Actualziacion = 2014-0501
# Nota         =

require(fda)
require(geofd)

#llama los datos TMAX 31 variables incluido month, day, year
datos_clima=read.csv("C:/DatosProyectos/TESIS/02DataResum/09PREC_Cumsum_Mean_3
65X29.csv")
names(datos_clima)
#julian_days=array(NA,length(datos_clima$year)) # crea arreglo de 5844 registros
for(i in 1:length(unique(datos_clima$year))){
julian_days[datos_clima$year==unique(datos_clima$year)[i]]=1:sum(datos_clima$year==un
ique(datos_clima$year)[i])}

plot(tapply(datos_clima$samaim, julian_days, mean),type="l")

```

```

tmax_julian=matrix(NA,ncol=28,nrow=366)
for(j in 4:31){
tmax_julian[,j-3]=tapply(datos_clima[,j],julian_days,mean)}

tmax_julian=as.data.frame(tmax_julian)
names(tmax_julian)=names(datos_clima)[4:31]

##Functional Data
length(tmax_julian[1])

n<-dim(tmax_julian)[1]
argvals<-seq(1,n, by=1)

#s es el numero de variables (28 estacionesValle) Giraldo 2012,Rmateu pg 391 of 23 (antes
s=33)
s<-28
rangeval <- range(argvals)
norder <- 4
nbasis <- 65
bspl.basis <- create.bspline.basis(rangeval, nbasis, norder)
lambda <-0

datafdPar <- fdPar(bspl.basis, Lfdobj=2, lambda)
tmax_julian=as.matrix(tmax_julian)
smfd <- smooth.basis(argvals,tmax_julian,datafdPar)
datafd <- smfd$fd
M <- bsplinepen(bspl.basis,Lfdobj=0)

windows()
par(mfrow=c(1,2))
matplot(tmax_julian,type="l",xlab="Dia", ylab="Temperatura Maxima Original")
grid()
plot(datafd, lty=1, xlab="Dia", ylab="Temperatura Maxima Suavizada")
grid()

L2norm <- l2.norm(s, datafd, M)

cafe_coords=read.csv("C:/DatosProyectos/TESIS/02DataResum/09Coordenadas_Ubicacione
s.csv",header=T)
#cafe_coords=cbind(cafe_coords$latitud,cafe_coords$longitud)
cafe_coords=cbind(cafe_coords$longitud,cafe_coords$latitud)
cafe_coords=as.data.frame(cafe_coords)
names(cafe_coords)

```

```

dista=max(dist(cape_coords[,1:2]))*0.9
tracev=trace.variog(cape_coords[,1:2], L2norm, bin=FALSE,
                    max.dist=dista,uvec="default",breaks="default",nugget.tolerance)

models=fit.tracevariog(tracev, models=c("spherical","exponential",
                                       "gaussian","matern"),sigma2.0=2000, phi.0=4, fix.nugget=TRUE,
                       nugget=0, fix.kappa=TRUE, kappa=1, max.dist.variogram=dista)

plot(tracev, xlab="Distancia", ylab="Trace-Variogram")
lines(models$fitted[[1]], lwd=2)
lines(models$fitted[[2]], lwd=2, col=4)
lines(models$fitted[[3]], lwd=2, col=7)
lines(models$fitted[[4]], lwd=2, col=6)
legend("topleft", c("empirical trace variogram", "spherical",
                   "exponential", "gaussian", "matern"), lty=c(-1,1,1,1,1),
       col=c(1,1,4,7,6), pch=c(1,-1,-1,-1,-1))

#coord.estim=read.csv("D:Proyectos/Geo-Functional/coords_valids.csv")
coord.estim=read.csv("C:/DatosProyectos/TESIS/02DataResum/09coords_valids_all.csv")
coord.estim=as.matrix(coord.estim[,2:3])

okfd.res<-okfd(new.coords=coord.estim, coords=cape_coords,
              cov.model="exponential", data=tmax_julian, nbasis=65,
              argvals=argvals, fix.nugget=TRUE)

source("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/a.r")
getwd()
geofd.viewer(okfd.res, argnames=c("Estimación","Día","Brillo Solar"))
dista=max(dist(cape_coords[,1:2]))*0.9
tracev=trace.variog(cape_coords[,1:2], L2norm, bin=FALSE,
                    max.dist=dista,uvec="default",breaks="default",nugget.tolerance)
models=fit.tracevariog(tracev, models=c("spherical","exponential",
                                       "gaussian","matern"),sigma2.0=2000, phi.0=4, fix.nugget=TRUE,
                       nugget=0, fix.kappa=TRUE, kappa=1, max.dist.variogram=dista)

plot(tracev, xlab="Distancia", ylab="Trace-Variogram")
lines(models$fitted[[1]], lwd=2)
lines(models$fitted[[2]], lwd=2, col=4)
lines(models$fitted[[3]], lwd=2, col=7)
lines(models$fitted[[4]], lwd=2, col=6)
legend("topleft", c("empirical trace variogram", "spherical",
                   "exponential", "gaussian", "matern"), lty=c(-1,1,1,1,1),
       col=c(1,1,4,7,6), pch=c(1,-1,-1,-1,-1))

```

```

#coord.estim=read.csv("D:Proyectos/Geo-Functional/coords_valids.csv")
coord.estim=read.csv("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/coords_valids.csv")
coord.estim=as.matrix(coord.estim[,2:3])

okfd.res<-okfd(new.coords=coord.estim, coords=cafe_coords[,2:3],
               cov.model="exponential", data=promedios, nbasis=65,
               argvals=argvals, fix.nugget=TRUE)

source("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/a.r")
getwd()
geofd.viewer(okfd.res, argnames=c("Estimación","Día","Brillo Solar"))
=====
# Programa      = 10_Modelo_TMAX.R
# Autor         = Jorge Cardona
# Actualizacion = 2014-0501
# Nota          =

require(fda)
require(geofd)
#llama los datos TMAX 31 variables incluido month, day, year
datos_clima=read.csv("C:/DatosProyectos/TESIS/03R-
Scripts/scriptVH/Results_script4/04tmax_to_FUS.csv")

julian_days=array(NA,length(datos_clima$year)) # crea arreglo de 5844 registros
for(i in 1:length(unique(datos_clima$year))){
julian_days[datos_clima$year==unique(datos_clima$year)[i]]=1:sum(datos_clima$year==un
ique(datos_clima$year)[i])}

plot(tapply(datos_clima$samaime,julian_days,mean),type="l")

tmax_julian=matrix(NA,ncol=28,nrow=366)
for(j in 4:31){
tmax_julian[,j-3]=tapply(datos_clima[,j],julian_days,mean)}

tmax_julian=as.data.frame(tmax_julian)
names(tmax_julian)=names(datos_clima)[4:31]

##Functional Data
length(tmax_julian[1])

n<-dim(tmax_julian)[1]
argvals<-seq(1,n, by=1)

```

```

s<-28 #numero de variables (28 estaciones Valle) Giraldo 2012,Rmateu pg 391 of 23 (antes
s=33)
rangeval <- range(argvals)
norder <- 4
nbasis <- 65
bspl.basis <- create.bspline.basis(rangeval, nbasis, norder)
lambda <- 0

datafdPar <- fdPar(bspl.basis, Lfdobj=2, lambda)
tmax_julian=as.matrix(tmax_julian)
smfd <- smooth.basis(argvals,tmax_julian,datafdPar)
datafd <- smfd$fd
M <- bsplinepen(bspl.basis,Lfdobj=0)

windows()
par(mfrow=c(1,2))
matplot(tmax_julian,type="l",xlab="Dia", ylab="Temperatura Maxima Original")
grid()
plot(datafd, lty=1, xlab="Dia", ylab="Temperatura Maxima Suavizada")
grid()

L2norm <- l2.norm(s, datafd, M)

cafe_coords=read.csv("C:/DatosProyectos/TESIS/02DataResum/09Coordenadas_Ubicacione
s.csv",header=T)
#cafe_coords=cbind(cafe_coords$latitud,cafe_coords$longitud)
cafe_coords=cbind(cafe_coords$longitud,cafe_coords$latitud)
cafe_coords=as.data.frame(cafe_coords)
names(cafe_coords)

dista=max(dist(cafe_coords[,1:2]))*0.9
tracev=trace.variog(cafe_coords[,1:2], L2norm, bin=FALSE,
                    max.dist=dista,uvec="default",breaks="default",nugget.tolerance)

models=fit.tracevariog(tracev, models=c("spherical","exponential",
                    "gaussian","matern"),sigma2.0=2000, phi.0=4, fix.nugget=TRUE,
                    nugget=0, fix.kappa=TRUE, kappa=1, max.dist.variogram=dista)

plot(tracev, xlab="Distancia", ylab="Trace-Variogram")
lines(models$fitted[[1]], lwd=2)
lines(models$fitted[[2]], lwd=2, col=4)
lines(models$fitted[[3]], lwd=2, col=7)

```

```

lines(models$fitted[[4]], lwd=2, col=6)
legend("topleft", c("empirical trace variogram", "spherical",
                    "exponential", "gaussian", "matern"), lty=c(-1,1,1,1,1),
      col=c(1,1,4,7,6), pch=c(1,-1,-1,-1,-1))

#coord.estim=read.csv("D:Proyectos/Geo-Functional/coords_valids.csv")
coord.estim=read.csv("C:/DatosProyectos/TESIS/02DataResum/09coords_valids_all.csv")
coord.estim=as.matrix(coord.estim[,2:3])

okfd.res<-okfd(new.coords=coord.estim, coords=cafe_coords,
              cov.model="exponential", data=tmax_julian, nbasis=65,
              argvals=argvals, fix.nugget=TRUE)

source("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/a.r")
getwd()
geofd.viewer(okfd.res, argnames=c("Estimación","Día","Brillo Solar"))
dista=max(dist(cafe_coords[,1:2]))*0.9
tracev=trace.variog(cafe_coords[,1:2], L2norm, bin=FALSE,
                  max.dist=dista,uvec="default",breaks="default",nugget.tolerance)
models=fit.tracevariog(tracev, models=c("spherical","exponential",
                                       "gaussian","matern"),sigma2.0=2000, phi.0=4, fix.nugget=TRUE,
                  nugget=0, fix.kappa=TRUE, kappa=1, max.dist.variogram=dista)

plot(tracev, xlab="Distancia", ylab="Trace-Variogram")
lines(models$fitted[[1]], lwd=2)
lines(models$fitted[[2]], lwd=2, col=4)
lines(models$fitted[[3]], lwd=2, col=7)
lines(models$fitted[[4]], lwd=2, col=6)
legend("topleft", c("empirical trace variogram", "spherical",
                    "exponential", "gaussian", "matern"), lty=c(-1,1,1,1,1),
      col=c(1,1,4,7,6), pch=c(1,-1,-1,-1,-1))

#coord.estim=read.csv("D:Proyectos/Geo-Functional/coords_valids.csv")
coord.estim=read.csv("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/coords_valids.csv")
coord.estim=as.matrix(coord.estim[,2:3])

okfd.res<-okfd(new.coords=coord.estim, coords=cafe_coords[,2:3],
              cov.model="exponential", data=promedios, nbasis=65,
              argvals=argvals, fix.nugget=TRUE)

source("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/a.r")
getwd()
geofd.viewer(okfd.res, argnames=c("Estimación","Día","Brillo Solar"))
=====

```

```

# Programa      = 10_MODELO_TMIN.R
# Autor         = Jorge CArdona
# Actualziacion = 2014-0501
# Nota          =

require(fda)
require(geofd)
#llama los datos TMAX 31 variables incluido month, day, year
datos_clima=read.csv("C:/DatosProyectos/TESIS/03R-
Scripts/scriptVH/Results_script4/04tmin_to_FUS.csv")

julian_days=array(NA,length(datos_clima$year)) # crea arreglo de 5844 registros
for(i in 1:length(unique(datos_clima$year))){
julian_days[datos_clima$year==unique(datos_clima$year)[i]]=1:sum(datos_clima$year==un
ique(datos_clima$year)[i])}

plot(tapply(datos_clima$samaime,julian_days,mean),type="l")

tmax_julian=matrix(NA,ncol=28,nrow=366)
for(j in 4:31){
tmax_julian[,j-3]=tapply(datos_clima[,j],julian_days,mean)}

tmax_julian=as.data.frame(tmax_julian)
names(tmax_julian)=names(datos_clima)[4:31]

##Functional Data
length(tmax_julian[1])

n<-dim(tmax_julian)[1]
argvals<-seq(1,n, by=1)

#s es el numero de variables (28 estacionesValle) Giraldo 2012,Rmateu pg 391 of 23 (antes
s=33)
s<-28
rangeval <- range(argvals)
norder <- 4
nbasis <- 65
bspl.basis <- create.bspline.basis(rangeval, nbasis, norder)
lambda <-0

datafdPar <- fdPar(bspl.basis, Lfdobj=2, lambda)
tmax_julian=as.matrix(tmax_julian)
smfd <- smooth.basis(argvals,tmax_julian,datafdPar)
datafd <- smfd$fd
M <- bsplinepen(bspl.basis,Lfdobj=0)

```



```

windows()
par(mfrow=c(1,2))
matplot(tmax_julian,type="l",xlab="Dia", ylab="Temperatura Mnina Original")
grid()
plot(datafd, lty=1, xlab="Dia", ylab="Temperatura Mnima Suavizada")
grid()

L2norm <- l2.norm(s, datafd, M)

cafe_coords=read.csv("C:/DatosProyectos/TESIS/02DataResum/09Coordenadas_Ubicacione
s.csv",header=T)

cafe_coords=cbind(cafe_coords$longitud,cafe_coords$latitud)
cafe_coords=as.data.frame(cafe_coords)
names(cafe_coords)

dista=max(dist(cafe_coords[,1:2]))*0.9
tracev=trace.variog(cafe_coords[,1:2], L2norm, bin=FALSE,
                    max.dist=dista,uvec="default",breaks="default",nugget.tolerance)

models=fit.tracevariog(tracev, models=c("spherical","exponential",
                                       "gaussian","matern"),sigma2.0=2000, phi.0=4, fix.nugget=TRUE,
                       nugget=0, fix.kappa=TRUE, kappa=1, max.dist.variogram=dista)

plot(tracev, xlab="Distancia", ylab="Trace-Variogram")
lines(models$fitted[[1]], lwd=2)
lines(models$fitted[[2]], lwd=2, col=4)
lines(models$fitted[[3]], lwd=2, col=7)
lines(models$fitted[[4]], lwd=2, col=6)
legend("topleft", c("empirical trace variogram", "spherical",
                  "exponential", "gaussian", "matern"), lty=c(-1,1,1,1,1),
      col=c(1,1,4,7,6), pch=c(1,-1,-1,-1,-1))

#coord.estim=read.csv("D:Proyectos/Geo-Functional/coords_valids.csv")
coord.estim=read.csv("C:/DatosProyectos/TESIS/02DataResum/09coords_valids_all.csv")
coord.estim=as.matrix(coord.estim[,2:3])

okfd.res<-okfd(new.coords=coord.estim, coords=cafe_coords,
              cov.model="exponential", data=tmax_julian, nbasis=65,
              argvals=argvals, fix.nugget=TRUE)

source("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/a.r")
getwd()

```

```

geofd.viewer(okfd.res, argnames=c("Estimación","Día","Brillo Solar"))
dista=max(dist(cafes_coords[,1:2]))*0.9
tracev=trace.variog(cafes_coords[,1:2], L2norm, bin=FALSE,
                    max.dist=dista,uvec="default",breaks="default",nugget.tolerance)
models=fit.tracevariog(tracev, models=c("spherical","exponential",
                    "gaussian","matern"),sigma2.0=2000, phi.0=4, fix.nugget=TRUE,
                    nugget=0, fix.kappa=TRUE, kappa=1, max.dist.variogram=dista)

plot(tracev, xlab="Distancia", ylab="Trace-Variogram")
lines(models$fitted[[1]], lwd=2)
lines(models$fitted[[2]], lwd=2, col=4)
lines(models$fitted[[3]], lwd=2, col=7)
lines(models$fitted[[4]], lwd=2, col=6)
legend("topleft", c("empirical trace variogram", "spherical",
                    "exponential", "gaussian", "matern"), lty=c(-1,1,1,1,1),
                    col=c(1,1,4,7,6), pch=c(1,-1,-1,-1,-1))

#coord.estim=read.csv("D:Proyectos/Geo-Functional/coords_valids.csv")
coord.estim=read.csv("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/coords_valids.csv")
coord.estim=as.matrix(coord.estim[,2:3])

okfd.res<-okfd(new.coords=coord.estim, coords=cafes_coords[,2:3],
               cov.model="exponential", data=promedios, nbasis=65,
               argvals=argvals, fix.nugget=TRUE)

source("C:/Users/jcardona/Desktop/FDA/BRILLO_SOLAR/a.r")
getwd()
geofd.viewer(okfd.res, argnames=c("Estimación","Día","Brillo Solar"))
=====

```