UNIVERSITAT JAUME·I

Máster en Matemática Computacional

Proyecto Final de Máster

Procesos puntuales espacio-temporales con aplicación a la modelización de accidentes de tráfico

Autor:
Javier Calahorra Tovar

Tutor académico:
PhD: Jorge Mateu Mahiques

Fecha de lectura: 27 de septiembre de 2017
Curso académico 2016/2017

A mi abuela María.

Agradecimientos

A todas las personas que forman o han formado parte de nuestro grupo de investigación. En especial a Jorge, por haberme brindado la oportunidad de vivir esta experiencia. A los expertos en *GIS*, Pau y Óscar, a Manoli, Francisco, Jonathan y Raquel. Gracias por haberme recibido como uno más desde el primer día. Gracias por los almuerzos, los consejos, la ayuda y todos esos ratos que hacen que madrugar sea menos duro.

A mis padres, a mi hermano y a Amparo, por su apoyo incondicional.

A mi compañera de viaje. Gracias por acompañarme durante estos años con amor, comprensión, paciencia y fortaleza. Como en todo lo que escribo, estás presente en mi mente y en el alma de estas líneas. Como dijo Benedetti, *eres mi cómplice y mi todo, y en la calle codo a codo, somos mucho más que dos.*

Resumen

El objetivo final de este *Proyecto Final de Máster* es crear un modelo espacio-temporal capaz de estimar la función de intensidad de un proceso estocástico puntual.

En el primer capítulo, desarrollaremos los fundamentos teóricos que nos permitirán crear dicho modelo; a saber: proceso estocástico puntual, *modelo lineal generalizado* (GLM), *modelos aditivos generalizados* (GAM), ... En el segundo y último capítulo mostraremos el modelo construído y analizaremos la bondad de ajuste del mismo, es decir, cómo es de fiable.

La base de datos utilizada para acometer este proyecto es *Motor Vehicle Traffic Accident*. En dicha base de datos están registrados todos los accidentes de Houston, Texas, desde el año 1999 hasta el año 2001. Además, *Motor Vehicle Traffic Accident* contiene 180.970 observaciones con 591 covariables asociadas a cada accidente, afrontando de esta forma un problema de *big data* cuyo dominio se reduce a las calles o ejes de Houston.

Abstract

The final objective of this Final Master Project is to create a space-time model capable of estimating the intensity function of a stochastic point process.

In the first chapter, we will develop the theoretical foundations that will allow us to create the model; namely: stochastic point process, *generalized linear model* (GLM), *generalized additive models* (GAM), ... In the second and last chapter we will show the constructed model and analyze the goodness of fit of the same, that is, how reliable it is.

The database used to undertake this project is Motor Vehicle Traffic Accident. In that database are registered all the accidents of Houston, Texas, from the year 1999 to the year 2001. In addition, Motor Vehicle Traffic Accident contains 180.970 observations with 591 covariables associated with each accident, thus facing a big data problem whose domain is reduced to the streets or network of Houston.

Palabras clave

Proceso puntual, Proceso estocástico, Modelo lineal generalizado, Modelos aditivos generalizados, GLM, GAM, Modelo espacio-temporal, Función de intensidad, red.

Keywords

Point process, Stochastic process, Generalized linear model, Generalized additive model, GLM, GAM, Space-time model, Intensity function, network.

Índice general

Capítulo 1

Procesos puntuales espacio-temporales

En este capítulo definiremos los conceptos necesarios y mostraremos algunos resultados importantes que proporcionarán al lector una base matemática sólida. Comenzaremos estudiando los procesos puntuales espaciales. Posteriormente, trabajaremos sobre los procesos puntuales espacio-temporales como una extensión de los primeros.

1.1. Introducción

Los procesos espaciales puntuales son modelos estocásticos que describen la localización de eventos de interés. Los más comunes son aquellos en los cuales la localización viene señalada por dos dimensiones. En nuestro caso, trabajaremos con procesos puntuales univariantes, es decir, incluyen sólo la localización de un tipo de evento (accidentes de tráfico).

Esta tipología de datos y modelos resultan útiles para resolver las siguientes cuestiones en el ámbito científico:

- averiguar si los datos del modelo espacial son agrupados, si se distribuyen de manera regular o de forma aleatoria;
- hallar relaciones entre los diferentes tipos de eventos en los procesos marcados;
- calcular funciones de densidad (número de eventos por unidad de área).

En lo que sigue, nos centraremos en esta última cuestión con el objetivo de detectar la tendencia de los accidentes de tráfico en la ciudad de Houston. Esta tendencia vendrá influida y/o determinada por las covariables observadas en cada evento (edad del conductor, estado de la vía, visibilidad,...)[1]

Los procesos puntuales espacio-temporales empezaron a ser estudiados en profundidad debido a la necesidad de modelar y predecir factores ambientales; en particular, terremotos. En 1999, *Ogata*[18] escribió una publicación en la cual resumía las técnicas paramétricas de máxima verosimilitud. Una año más tarde, *Choy* y *Hall*[4] añadieron estimadores no paramétricos de la función de intensidad utilizando un estimador Kernel. En 1994, *Rathbun* y *Cressie*[19] discutieron acerca de los procesos puntuales espacio-temporales en el contexto del crecimiento de los árboles. Existe también literatura sobre procesos puntuales para datos tipo lattice y geoestadísticos, véase por ejemplo, *Kooperberg* y *O'Sullivan*[15] y *Haas*[13].

Como consecuencia de que la localización espacial de un evento siempre puede considerarse como una componente más en un sistema multidimensional -véase *Daley* y *Vere-Jones*[6], 2002-, la evolución de las características espaciales de un evento con respecto al tiempo es, en muchas ocasiones, de gran interés en diversos ámbitos científicos. Sin embargo, el estudio y desarrollo de modelos espacio-tiempo se ha producido de forma tardía respecto a los temporales e incluso los puramente espaciales. Esto se debe a la dificultad computacional que existe para trabajar con buenos conjuntos de datos espacio-tiempo y la inviabilidad de realizar los cálculos necesarios para analizar dichos datos.

Una forma de entender los procesos puntuales espacio-temporales desarrollados por *Daley* y *Vere-Jones*[7] (2008) es considerar la ubicación espacial como una marca de un proceso puntual simple en el tiempo proporcionando, de esta forma, un análisis probabilístco de los modelos espacio-tiempo. Otras características como la magnitud, extensión o, incluso, la duración se pueden adicionar al modelo como marcas de los eventos. Así, el estudio de los procesos puntuales en el espacio-tiempo conduce, de forma inevitable, al estudio más general de la evolución de los campos espaciales. En este sentido, el modelado es todavía limitado y específico del sujeto.

Siguiendo este enfoque, en 1995, *Diggle* y *Hoggkvist*[11] extendieron los métodos existentes de segundo orden para modelos espaciales a la configuración espacio-temporal. Esta extensión permite estimar la interacción espacio-tiempo como una función del espacio y el tiempo, separables.

Por último, notar que los sucesos de un proceso puntual espacio-tiempo forman un conjunto numerable, $\mathbb{P} = \{(s_i, t_i) : i = 1, 2, \ldots\}$ donde $s_i \in \mathbb{R}^2$ es la localización y $t_i \in \mathbb{R}^+$ es el tiempo de ocurrencia del evento i-ésimo. En la práctica, los datos disponibles para el análisis son los puntos $(s_i, t_i), i = 1, 2, \ldots$, restringidos a un dominio espacio temporal, $W \times T$, donde W es, habitualmente, un polígono cerrado y T un único intervalo cerrado.

En lo que sigue, denotaremos como $Y(A)$ al número de eventos acontecidos en una región arbitraria A.

1.2. Propiedades de primer y segundo orden

Las propiedades de primer orden se describen por la intensidad del proceso

$$\lambda(s,t) = \lim_{|ds|\to 0, |dt|\to 0} \frac{\mathbb{E}\left[Y(\mathrm{d}s, \mathrm{d}t)\right]}{|\mathrm{d}s|\,|\mathrm{d}t|}, \tag{1.1}$$

donde ds define una pequeña región espacial alrededor de s ($|\mathrm{d}s|$ es su área), dt es un pequeño intervalo que contiene el tiempo t ($|\mathrm{d}t|$) es la longitud de dicho intervalo y $Y(\mathrm{d}s, \mathrm{d}t)$ se refiere al

número de eventos acontecidos en $\mathrm{d}s \times \mathrm{d}t$. Así, de forma informal, $\lambda(s,t)$ es el número medio de eventos por unidad de volumen en (s,t). Un proceso en el cual $\lambda(s,t) = \lambda$ para todo (s,t) se llama *homogéneo*.

Las propiedades de segundo orden describen la relación entre el número de eventos en pares de subregiones dentro de $S \times T$. La intensidad de segundo orden se define como

$$\lambda_2((s_i,t_i),(s_j,t_j)) = \lim_{|D_i|\to 0, |D_j|\to 0} = \frac{\mathbb{E}\left[Y(D_i,D_j)\right]}{|D_i|\,|D_j|}, \tag{1.2}$$

donde $D_i = \mathrm{d}s_i \times \mathrm{d}t_i$ y $D_j = \mathrm{d}s_j \times \mathrm{d}t_j$ son pequeños cilindros que contienen los puntos (s_i,t_i) y (s_j,t_j), respectivamente.

Otros descriptores de propiedades de segundo orden, esencialmente equivalentes, incluyen la densidad de covarianza,

$$\gamma((s_i,t_i),(s_j,t_j)) = \lambda_2((s_i,t_i),(s_j,t_j)) - \lambda(s_i,t_i)\lambda(s_j,t_j) \tag{1.3}$$

y la *función de distribución radial* o *función de correlación de pares* de puntos (*pair correlation*)

$$g((s_i,t_i),(s_j,t_j)) = \frac{\lambda_2((s_i,t_i),(s_j,t_j))}{\lambda(s_i,t_i)\lambda(s_j,t_j)}. \tag{1.4}$$

La densidad de covarianza es el proceso puntual análogo a la función de covarianza de un proceso estocástico evaluado en los reales. La función de correlación de pares o *pair correlation* puede interpretarse como la función de densidad de probabilidad estándar aplicada a que un evento ocurra en cada uno de dos cilindros centrados en los puntos (s_i,t_i), (s_j,t_j). Para un proceso Poisson espacio-temporal (que se definirá formalmente más adelante), la densidad de covarianza es idénticamente cero y la función de correlación de pares es idénticamente 1.

1.3. Estacionariedad

Un proceso puntual espacio-temporal $\{(\mathbf{s},t), \mathbf{s} \in W, t \in T\}$ es estacionario de primer y segundo orden

- en el espacio, si: $\lambda(\mathbf{s},t) \equiv \lambda(t)$ y $\lambda_2((\mathbf{s},t),(\mathbf{s}',t)) = \lambda_2(\mathbf{s}-\mathbf{s}',t)$;
- en el tiempo, si: $\lambda(\mathbf{s},t) \equiv \lambda(s)$ y $\lambda_2((\mathbf{s},t),(\mathbf{s},t')) = \lambda_2(\mathbf{s},t-t')$;
- en el tiempo y el espacio, si $\lambda(\mathbf{s},t) \equiv \lambda$ y $\lambda_2((\mathbf{s},t),(\mathbf{s}',t')) = \lambda_2(\mathbf{s}-\mathbf{s}',t-t')$.

Un proceso puntual estacionario en el espacio-tiempo es isotrópico si $\lambda_2((\mathbf{s},t),(\mathbf{s}',t')) = \lambda_2(u,v)$, donde (u,v) es el vector de diferencia espacio-temporal, $u = \left\|\mathbf{s}-\mathbf{s}'\right\|$ y $v = \left|t-t'\right|$.

Un proceso puntual espacio-temporal es *estacionario isotrópico de segundo orden e intensidad reponderada* si su función de intensidad está acotada, alejada de cero, y su función de correlación de pares depende sólo de la diferencia del vector (u, v).

1.4. Separabilidad

En lo que sigue, supondremos que los efectos de primer orden son separables (véase Diggle and Gabriel[8], 2009). Un proceso puntual espacio-temporal es separable de primer orden si su intensidad $\lambda(s,t)$ puede ser factorizado como

$$\lambda\left(\mathbf{s},t\right)=\lambda\left(\mathbf{s}\right)\lambda\left(t\right),\ \text{para todo}\ (s,t)\in W\times T. \tag{1.5}$$

donde $\lambda(\mathbf{s})$ y $\lambda(t)$ son funciones no negativas. Bajo este supuesto, cualquier efecto de no separabilidad podrá interpretarse como de segundo orden.

Un proceso puntual espacio-temporal es separable de segundo orden si la densidad de la convarianza, $\gamma(u,v)=\lambda(u,v)-\lambda^2$, factoriza como

$$\gamma(u,v)=\gamma_s(u)\gamma_t(v). \tag{1.6}$$

Obsérvese que, en general, la separabilidad de segundo orden está implícita en la independencia de las componentes espaciales y temporales del proceso. Sin embargo, un proceso de Poisson tiene componentes independientes si y sólo si es de primer orden separable.

1.4.1. Estimación no paramétrica de la función de intensidad

Supongamos ahora que hemos obtenido estimaciones de $\hat{\lambda}_{\mathcal{S}}$ y $\hat{\lambda}_{\mathcal{T}}$. Si $\hat{\lambda}_{\mathcal{S}}$ y $\hat{\lambda}_{\mathcal{T}}$ nos proporcionan estimaciones no sesgadas del número esperado de puntos observados, i.e.,

$$\int_W \hat{\lambda}_{\mathcal{S}}\left(\mathbf{u}\right)d\mathbf{u}=n=\int_T \hat{\lambda}_{\mathcal{T}}\left(t\right)dt,$$

entonces la estimación de la función de intensidad espacio-temporal viene dada por

$$\hat{\lambda}\left(\mathbf{u},t\right)=\frac{1}{n}(\hat{\lambda}_{\mathcal{S}}\left(\mathbf{u}\right)\hat{\lambda}_{\mathcal{T}}\left(t\right)). \tag{1.7}$$

Esto nos proporciona una estimación no sesgada del número esperado de puntos observados, es decir,

$$\int_{W\times T}\hat{\lambda}\left(\mathbf{u},t\right)d(\mathbf{u},t)=n$$

Para la estimación no paramétrica de la función de intensidad espacial, emulamos el desarrollo que *Diggle*[10] (1985) y *Berman y Diggle*[3] (1989) usando la estimación kernel de tal forma que

$$\hat{\lambda}_{\mathcal{S}}\left(\mathbf{u}\right)=\sum_{i=1}^{n}\frac{k_\epsilon\left(\mathbf{u}-\mathbf{u}_i\right)}{c_{W,\epsilon}\left(\mathbf{u}_i\right)},\qquad \mathbf{u}\in W, \tag{1.8}$$

donde

$$k_\epsilon(\mathbf{u}) = \frac{1}{\epsilon^2} k\left(\frac{\mathbf{u}}{\epsilon}\right)$$

es un kernel con ancho de banda $\epsilon > 0$, esto es, k es una función de densidad dada. Además,

$$c_{W,\epsilon}(\mathbf{u}_i) = \int_W k_\epsilon(\mathbf{u} - \mathbf{u}_i)\, d\mathbf{u}$$

es un factor de corrección que asegura $\int_W \hat{\lambda}_{\mathcal{S}}(\mathbf{u})\, d\mathbf{u} = n$.

Una estimación kernel similar también puede utilizarse para la estimación no paramétrica de $\lambda_{\mathcal{T}}(t)$. Si la cola de la función de distribución empírica de los tiempos observados t_i es pesada, es más razonable usar la log-transformación, retransformando el sistema como en *Moller* y *Ghorbani*[17], 2012, donde primero se obtiene una estimación kernel $\hat{h}_\delta$ para la función de intensidad de los tiempos observados log-transformados y, luego,

$$\hat{\lambda}_{\mathcal{T}}(t) = \frac{1}{t}\left(\hat{h}_\delta(\log t)\right) \tag{1.9}$$

es utilizado como estimación no paramétrica del tiempo; aunque estos procedimientos de estimación no paramétricos de las funciones de intensidad espacial y temporal sólo nos pueden conducir a estimaciones no sesgadas.

1.5. Propiedades de segundo orden

Las propiedades de segundo orden descritas en la sección anterior se pueden emplear para analizar la estructura espacio-temporal de un proceso puntual. En particular, la función de correlación entre pares no homogénea en el espacio-tiempo y la K-función se puede usar como medida de clustering y de interacción entre los eventos en el espacio y el tiempo.

1.5.1. K-función inomogénea en el espacio-tiempo

Para un proceso puntual *estacionario isotrópico de segundo orden e intensidad reponderada* la función K inhomogénea espacio-tiempo (función $STIK$) definida por Diggle y Gabriel[8] es

$$K_{ST}(u, v) = 2\pi \int_0^v \int_0^u g(u', v') u' \mathrm{d}u' \mathrm{d}v', \tag{1.10}$$

donde $g(u, v) = \lambda_2(u, v)/(\lambda(\mathbf{s}, t)\lambda(\mathbf{s}', t'))$, $u = \|\mathbf{s} - \mathbf{s}'\|$ y $v = \|t - t'\|$. Gabriel y Diggle también dieron una segunda definición que considera los acontecimientos pasados y futuros

$$K^*_{ST}(u, v) = 2\pi \int_{-v}^v \int_0^u g(u', v') u' \mathrm{d}u' \mathrm{d}v'. \tag{1.11}$$

La función $STIK$ caracteriza las propiedades de segundo orden de un proceso *estacionario isotrópico de segundo orden e intensidad reponderada* y puede usarse como una medida de agregación o regularidad espaciotemporal. Para cualquier proceso de Poisson espacio-temporal no homogéneo con intensidad acotada, lejos de cero, $K_{ST}(u, v) = \pi u^2 v$.

1.5.2. Estimación de la función pair correlation inomogénea en el espacio-tiempo

Un estimador de la función espacio-temporal pair correlation es

$$\hat{g}(u,v) = \frac{1}{W \times T} \sum_{i=1}^{n} \sum_{j \neq i} \frac{1}{w_{ij} v_{ij}} \frac{k_s(u - \|s_i - s_j\|) k_t(v - |t_i - t_j|)}{\lambda(s_i, t_i) \lambda(s_j, t_j)}, \tag{1.12}$$

donde w_{ij} y v_{ij} son los factores de corrección de borde y $k_s(\cdot)$, $k_t(\cdot)$ son funciones con ancho de banda h_s y h_t.

1.5.3. Estimación de la K-función no parmétrica en el espacio-tiempo

La estimación no paramétrica de las funciones de correlación de pares normalmente se basa en métodos kernel, donde la especificación del ancho de banda del núcleo es discutible. Alternativamente, una estimación no paramétrica y no sesgada de la función K viene dada por

$$\hat{K}(r,t) = \frac{1}{|W|\,|T|} \sum_{i \neq j} \frac{\mathbf{1}\left[\|\mathbf{u}_i - \mathbf{u}_j\| \leq r\right] \mathbf{1}\left[|t_i - t_j| \leq t\right] e^2\left(\mathbf{u}_i, \mathbf{u}_j\right) e^1\left(t_i, t_j\right)}{\lambda\left(\mathbf{u}_i, t_i\right) \lambda\left(\mathbf{u}_j, t_i\right)} \tag{1.13}$$

donde $\sum_{i \neq j}$ significa que la suma se realiza sobre todos los pares $(\mathbf{u}_i, t_i) \neq (\mathbf{u}_j, t_i)$ de los puntos de la base de datos, e^2 y e^1 denota los factores de corrección de Ripley. Notar que en el caso temporal $e^1\left(t_i, t_j\right) = 1$ si ambos extremos del intervalo de longitud $2\,|t_i - t_j|$ centrados en t_i y 2 en otro caso. En la práctica, λ tiene que estimarse.

1.6. Modelos

Los modelos espacio-temporales son una extensión natural de los modelos espaciales. Para establecer el contexto adecuado nos centramos en las definiciones dadas por Gabriel, Rowlingson, and Diggle[12], 2012.

1.6.1. Procesos Poisson

Procesos Poisson homogéneos

El proceso homogéneo de Poisson es el mecanismo estocástico más simple posible para la generación de patrones puntuales espacio-temporales. Raramente es plausible como modelador de las diferentes bases datos pero proporciona un punto de referencia de aleatoriedad espacio-temporal completa, *CSTR*. De forma informal, en cualquier región espacio-temporal $W \times T$ de X, los eventos generan una muestra aleatoria independiente de la distribución uniforme en $W \times T$. Más formalmente, el proceso homogéneo de Poisson se define por el siguientes postulados:

i. Para algún $\lambda > 0$, el número $Y(W \times T)$ de eventos dentro de la región $W \times T$ sigue una distribución Poisson con media $\lambda |W| |T|$ donde $|\cdot|$ denota la medida de Lebesgue.

ii. Dado $Y(W \times T) = n$, los n eventos en $W \times T$ forman una muestra aleatoria e independiente de la distribución uniforme sobre $W \times T$.

Las intensidades de primer orden y de segundo orden de un proceso de Poisson homogéneo se reducen a constantes $\lambda(\mathbf{s}, t) = \lambda$ y $\lambda_2((\mathbf{s}_i, t_i)(\mathbf{s}_j, t_j)) = \lambda^2$. Además, la densidad de la covarianza es idénticamente cero, la pair correlation es idénticamente 1, y la función STIK es $K_{ST}(u, v) = \pi u^2 v$.

Procesos Poisson inhomogéneos

El proceso Poisson inhomogéneo es el proceso puntual no estacionario más simple. Se obtiene sustituyendo la función de intensidad constante de un proceso de Poisson homogéneo por una función de intensidad que varía espacial y/o temporalmente. Los procesos de Poisson no homogéneos se definen por los siguientes postulados:

i. El número $\mathrm{Y}(W \times T)$ de eventos acontecidos dentro de la región $W \times T$ sigue una distribución Poisson con media $\int_W \int_T \lambda(\mathbf{s}, t) \mathrm{d}s\mathrm{d}t$.

ii. Dado $Y(W \times T) = n$, los n eventos en $W \times T$ forman una muestra aleatoria e independiente de la distribución en $W \times T$ y con función de densidad de probabilidad $f(\mathbf{s}, t) = \lambda(\mathbf{s}, t) / \int_W \int_T \lambda(\tilde{s}, \tilde{t}) \mathrm{d}\tilde{s}\mathrm{d}\tilde{t}$.

Para un proceso Poisson con intensidad $\lambda(\mathbf{s}, t)$, la intensidad de segundo orden es $\lambda_2((s_i, t_i)(s_j, t_j)) = \lambda(s_i, t_i)\lambda(s_j, t_j)$. Por tanto, la función de densidad de la convarianza es idénticamente 0, la función pair correlation es idénticamente 0 y la función STIK es $K_{ST}(u, v) = \pi u^2 v$ en el caso homogéneo.

1.6.2. Procesos de interacción

Procesos inhibitorios

Los procesos de inhibición impiden (inhibición estricta) o hacen improbable la aparición de pares de eventos cercanos, dando lugar a patrones que son más regulares en el espacio y/o en el tiempo que un proceso de Poisson de la misma intensidad.

En un proceso de inhibición secuencial espacial simple (inhibición estricta), δ_s denota la distancia mínima permisible entre eventos y $\lambda(\mathbf{s})$ la intensidad espacial del proceso. La proporción del plano cubierto por discos no superpuestos de radio $\delta_s/2$ es $\rho = \lambda(s)\pi\delta_s^2/4$ y ρ se denomina densidad de empaquetamiento.

Los procesos de inhibición secuencial simple en el espacio y el tiempo se definen mediante el siguiente algoritmo. Consideremos una secuencia de m eventos (s_i, t_i) en $W \times T$. Entonces,

1. s_1 y t_1 están uniformemente distribuidos en W y T respectivamente.

2. En el $k-ésimo$ paso del algoritmo, $k = 2, \ldots, m$, s_k está uniformemente distribuido en la intersección de W con $\{s : \|s - s_j\| \geq \delta_s, j = 1, \ldots, k-1\}$ y t_k está uniformemente distribuido en la intersección de T con $\{t : |t - t_j| \geq \delta_t, j = 1, \ldots, k-1\}$.

Para obtener una mayor cantidad de clases de procesos de inhibición que el definido anteriormente, podemos extender la condición 2. de la definición algorítmica anterior introduciendo las funciones $p_s(u)$ y $p_t(v)$ que determinarán de forma conjunta la probabilidad de que un punto con posición s en el tiempo t sea aceptado como punto del proceso, de acuerdo con el siguiente algoritmo, en el cual las funciones $g_s(\cdot)$, $g_t(\cdot)$, $h_s(\cdot)$, $h_t(\cdot)$ y el parámetro r deben estar definidos.

1. s_1 y t_1 están uniformemente distribuidos en W y T respectivamente.

2. En el $k-ésimo$ paso del algoritmo, $k = 2, \ldots, m$,

- Generar un evento (uniforme) con $\mathbf{s} \in W$ en el tiempo $t \in T$;
- Generar $u_s \sim U[0,1]$ y $u_t \sim U[0,1]$.
- Si $\|s - s_j\| \geq \delta_s$ para todo $j = 1, \ldots, k-1$, entonces fijamos $p_s = 1$. En otro caso, $p_s = g_s(h_s((\|s - s_j\|)_{j=1,\ldots,k-1}), \theta_s, \delta_s, r)$.
- Si $|t - t_j| \geq \delta_t$ para todo $j = 1, \ldots, k-1$, entonces fijamos $p_t = 1$. En otro caso, $p_t = g_t(h_t((|t - t_j|)_{j=1,\ldots,k-1}), \theta_t, \delta_t, r)$.
- Si $u_s < p_s$ y $u_t < p_t$, seleccionar (s, t).

Procesos contagiosos

Un proceso contagioso simple en el espacio-tiempo se puede definir algorítmicamente como sigue. Consideremos una secuencia de m eventos (s_i, t_i) en $W \times T$. Entonces,

1. s_1 y t_1 están uniformemente distribuidos en W y T respectivamente.

2. En el $k-ésimo$ paso del algoritmo, dados $\{(s_j, t_j), j = 1, \ldots, k-1\}$, s_k está uniformemente distribuido en la intersección de W con el círculo centrado en s_{k-1} y de radio δ_s, mientras que t_k está uniformemente distribuido en la intersección de T y el segmento $[t_{k-1}, t_{k-1} + \lambda(t)]$.

Como en el caso de procesos inhibitorios, podemos amplir la clase de procesos contagiosos introduciendo las funciones p_s y p_t, que dependen de $\|s - s_j\|$ y $|-t_j|$, respectivamente. Entonces, el $k-ésimo$ paso del algoritmo es

- Generar un evento (uniforme) con $s \in W$ en el tiempo $t \in T$;

- Generar $u_s \sim U[0,1]$ y $u_t \sim U[0,1]$.

- Si $\|s - s_j\| \geq \delta_s$ para todo $j = 1, \ldots, k-1$, entonces fijamos $p_s = 1$. En otro caso, $p_s = g_s(h_s((\|s - s_j\|)_{j=1,\ldots,k-1}), \theta_s, \delta_s, r)$.

- Si $|t - t_j| \geq \delta_t$ para todo $j = 1, \ldots, k-1$, entonces fijamos $p_t = 1$. En otro caso, $p_t = g_t(h_t((|t - t_j|)_{j=1,\ldots,k-1}), \theta_t, \delta_t, r)$.

- Si $u_s < p_s$ y $u_t < p_t$, seleccionar (s,t).

Las funciones g y h tienen la misma interpretación que para los procesos inhibitorios.

1.6.3. Procesos infecciosos

La diferencia entre una enfermedad infecciosa y una enfermedad contagiosa es que la primera puede ser contraída por una persona sin que haya entrado en contacto directo con una persona infectada, mientras que la segunda se transmite únicamente por contacto directo. Todas las enfermedades contagiosas son infecciosas, pero muchas enfermedades infecciosas no son contagiosas.

Aquí, usamos el término proceso contagioso para significar que la existencia de un evento en la localización $\mathbf{s}$ y en el tiempo t aumenta la probabilidad de que haya eventos adicionales del proceso cercanos a $(\mathbf{s}, t)$ en el espacio-tiempo. Por otro lado, denotamos mediante el término proceso infeccioso al proceso en el cual a cada individuo infectado en un momento t le hacemos corresponder una tasa de infección $h(t)$, que asumiremos depende de tres parámetros: un período de latencia α, la tasa máxima de infección β y el período de infección γ. Notar que un proceso infeccioso definido de esta forma puede presentar una combinación de propiedades de los procesos contagiosos e inhibitorios. *Diggle, Kaimi y Abellana*[9], en 2010, propusieron un ejemplo de esta clase de procesos para describir el patrón de colonización de un terreno de anidación, en el cual cada nueva llegada tiende a elegir un lugar de anidación cercano al resto pero no tan cerca como para invadir territorios ya establecidos.

Definimos un proceso infeccioso en espacio-tiempo como sigue. Consideremos una secuencia de m eventos (s_i, t_i) en $W \times T$. Entonces,

1. Escogemos la localización s_1 y el tiempo t_1 en el que se produce el primer hecho.

2. Dado $\{(s_j, t_j), j = 1, \ldots, k-1\}$, s_k se distribuye de forma simétrica y radial alrededor de s_{k-1} o es un punto de un proceso de Poisson con intensidad $\lambda(\mathbf{s})$, y t_k está distribuido uniforme o exponencialmente a partir de t_{k-1}. Denotamos por f_s y f_t a la distribución de s_k y t_k respecto s_{k-1} y t_{k-1}, respectivamente.

1.6.4. Procesos Log-Gaussianos de Cox

El proceso Log-Gaussiano de Cox es un proceso puntual doblemente estocástico formado como un proceso de Poisson inhomogéneo con una intensidad estocástica. Estos procesos fueron introducidos por *Cox*[5] (1955) en una dimensión temporal. Su definición en el espacio y el tiempo es:

1. $\{\Lambda(s,t), s \in S, t \in T\}$ es un proceso estocástico de valor no negativo

2. Condicionado a $\{\Lambda(s,t) = \lambda(s,t), s \in S, t \in T\}$, los eventos forman un proceso Poisson inhomogéneo con intensidad $\lambda(\mathbf{s}, t)$.

Supongamos que $Z = \{Z(s,t), s \in S, t \in T\}$ es un proceso Gaussiano evaluado en los reales con media $\mu(s,t) = E[Z(s,t)]$ y función de covarianza $c((s_i,t_i),(s_j,t_j)) = \mathrm{Cov}(Z(s_i,t_i), Z(s_j,t_j))$. Si la función de intensidad está definida como $\Lambda(s,t) = \exp Z(s,t)$, entonces el correspondiente proceso Y es, también, un proceso Log-Gaussiano de Cox.

1.7. Modelo lineal generalizado (GLM)

1.7.1. Componentes de un modelo generalizado lineal (GLM)

Un modelo lineal generalizado [16] tiene tres componentes básicos:

- componente aleatoria: identifica la variable respuesta y su distribución de probabilidad;
- componente sistemática: especifica las variables explicativas (independientes o predictoras) utilizadas en la función predictora lineal;
- función link: es una función del valor esperado de Y, $\mathbb{E}(Y)$, como una combinación lineal de las variables predictoras.

Componente aleatoria

La componente aleatoria de un GLM consiste en una variable aleatoria Y con observaciones independientes $(y_1, \ldots, y_N)$.

En muchas aplicaciones, las observaciones de Y son binarias y se identifican como éxito y fracaso. Aunque de modo más general, cada Y_i indicaría el número de éxitos de entre un número fijo de ensayos, y se modelizaría como una distribución binomial.

En otras ocasiones cada observación es un recuento, con lo que se puede asignar a Y una distribución de Poisson o una distribución binomial negativa. Finalmente, si las observaciones son continuas se puede asumir para Y una distribución normal.

Todos estos modelos se pueden incluir dentro de la llamada familia exponencial de distribuciones

$$f(y_i\|\theta_i) = a(\theta_i) \cdot b(y_i) \cdot \exp[y_i Q(\theta_i)],$$

de modo que $Q(\theta_i)$ recibe el nombre de parámetro natural.

Componente sistemática

La componente sistemática de un GLM especifica las variables explicativas, que entran en forma de efectos fijos en un modelo lineal, es decir, las variables x_j se relacionan como

$$\alpha + \beta_1 x_1 + \cdots + \beta_k x_k.$$

Esta combinación lineal de variables explicativas se denomina predictor lineal. Alternativamente, se puede expresar como un vector $(\eta_1, \ldots, \eta_N)$ tal que

$$\eta_i = \sum_j \beta_i x_{ij},$$

donde x_{ij} es el valor del $j - ésimo$ predictor en el $i - ésimo$ individuo e $i = 1, \ldots, N$. El término independiente α se obtendrá con esta notación haciendo que todos los x_{ij} sean igual a 1 para todos los i.

En cualquier caso, se pueden considerar variables que están basadas en otras variables como $x_3 = x_1 x_2$ ó $x_3 = x_{2_3}$, para modelizar interacciones entre variables o efectos curvilíneos de x_2.

Función link

Se denota el valor esperado de Y como $\mathbb{E}(Y)$. Entonces, la función link especifica una función $g(\cdot)$ que relaciona μ con el predictor lineal como

$$g(\mu) = \alpha + \beta_1 x_1 + \cdots + \beta_k x_k.$$

Así, la función link $g(\cdot)$ relaciona las componentes aleatoria y sistemática. De este modo, para $i = 1, \ldots, N$,

$$\mu_i = \mathbb{E}(Y_i);$$
$$\eta_i = g(\mu_i) = \sum_j \beta_j x_{ij}.$$

La función g más simple es $g(\mu) = \mu$, esto es, la identidad que da lugar al modelo de regresión lineal clásico.

Los modelos de regresión lineal típicos para respuestas continuas son un caso particular de los GLM. Estos modelos generalizan la regresión ordinaria de dos modos: permitiendo que Y tenga distribuciones diferentes a la normal y, por otro lado, incluyendo distintas funciones link de la media. Esto resulta bastante útil para datos categóricos.

Los modelos GLM permiten la unificación de una amplia variedad de métodos estadísticos como la regresión, los modelos ANOVA y los modelos de datos categóricos. Se usa el mismo algoritmo para obtener los estimadores de máxima verosimilitud en todos los casos. Dicho algoritmo es la base de la función *glm* de *R*.

1.7.2. Modelos GLM para recuentos

En nuestro caso, las variables respuesta son recuentos. En el modelo más simple se asume que el componente aleatorio Y sigue una distribución de Poisson. Esta distribución es unimodal y su propiedad más destacada es que la media y la varianza coinciden

$$\mathbb{E}(Y) = Var(Y) = \mu,$$

de modo que cuando el número de recuentos es mayor en media, también tienden a tener mayor variabilidad.

En el modelo GLM se usa habitualmente el logaritmo de la media para la función link, de modo que el modelo loglineal con una variable explicativa X se puede expresar como

$$\log(\mu) = \mu + \beta x,$$

de modo que

$$\mu = \exp[\mu + \beta x] = e(e^{\beta})^{x}.$$

1.7.3. Sobredispersión en GLM Poisson

En una distribución de Poisson la media y la varianza son iguales. Sin embargo, cuando trabajamos con recuentos reales no suele ser cierta esta hipótesis. Con frecuencia, la varianza es mayor que la media. A esto se le llama *sobredispersión*.

Habitualmente, esta situación se debe a la existencia de heterogeneidad entre las observaciones. Esto se puede interpretar como una mezcla o mixtura de distribuciones de Poisson. No es un problema cuando Y sigue una distribución normal ya que dicha distribución posee parámetro específico que modeliza la variabilidad.

Notar que la estimación del parámetro de dispersión no es más que la suma de los residuos dividida entre sus grados de libertad.

1.7.4. Modelos GLM binomiales negativos

Si una varibale aleatoria Y se distribuye como una binomial negativa, entonces la función de probabilidad es

$$P(y\|k,\mu) = \frac{\Gamma(y+k)}{\Gamma(k)\Gamma(y+1)}\left(\frac{k}{\mu+k}\right)^{k}\left(1-\frac{k}{\mu+k}\right)^{y} \tag{1.14}$$

con $y = 0, 1, 2, \ldots$ donde k y μ son los parámetros de la distribución.

Se tiene que:

$$\mathbb{E}(Y) = \mu$$
$$Var(Y) = \mu + \frac{\mu^2}{k}$$

El parámetro $\frac{1}{k}$ es un parámetro de dispersión, de modo que si $\frac{1}{k} \to 0$ entonces $Var(Y) \to \mu$ y la distribución binomial negativa converge a una distribución de Poisson. Por otro lado, para un valor fijo de k, esta distribución pertenece a la familia exponencial natural, de modo que se puede definir un modelo GLM binomial negativo. En general, se usa una función link de tipo logaritmo.

La regresión binomial negativa se puede utilizar para datos sobredispersos de recuentos, es decir, cuando la varianza condicional es mayor que la media condicional. Se puede considerar como una generalización de la regresión de Poisson ya que tiene su misma estructura de medias y, además, un parámetro adicional para el modelo de sobredispersión. Si la distribución condicional de la variable observada es más dispersa, los intervalos de confianza para la regresión binomial negativa es probable que sean más estrechos que los correspondientes a un modelo de regresión de Poisson.

1.8. Modelo aditivo generalizado (GAM)

El nombre Generalized Additive Model ha sido acuñado por *Hastie y Tibshirani*[14]F. Fueron los primeros en proponer este tipo de modelos así como diversos procedimientos para su estimación y contraste. La técnica específica de estimación que propusieron se llama *backfitting*. Posee la ventaja de poder integrar una gran variedad de procedimientos de suavización habituales (como los árboles de regresión) pero su punto débil reside en la estimación del grado de suavización del modelo[2].

Definición 1.1. *Un modelo lineal generalizado (GAM) es un modelo lineal con un predictor lineal definido a través de una suma de funciones suaves de las covariables,*

$$(\mathbb{E}\|\boldsymbol{X}, U, V, W, Z, \ldots) \sim \text{ distribución perteneciente a la familia exponencial,}$$

donde

$$\mu = \mathbb{E}(Y\|\boldsymbol{X}, U, V, W, Z, \ldots)$$
$$g(\mu) = \boldsymbol{X}\beta + f_1(U) + f_2(V) + f_3(W, Z) + \cdots$$

siendo:

- Y*: variable respuesta;*
- $\boldsymbol{X}$*: matriz de diseño correspondiente a las covariables que definen las componentes paramétricas del modelo;*
- β*: vector de los coeficientes de regresión;*

- $U, V, W, Z, \ldots$*: covariables;*
- $f_j(\cdot)$*: funciones suaves de las covariables.*

Notar que las distribuciones de probabilidad que pertenecen a la familia exponencial son:

- distribuciones discretas: Bernoulli, Binomial, Binomial negativa, Geométrica, Multinomial y Poisson;
- distribuciones contínuas: Beta, Chi-cuadrado, Dirichlet, Exponencial,....

Capítulo 2

Modelo espacio-temporal de los accidentes de tráfico en la ciudad de Houston

En este capítulo detallaremos cómo hemos construido el modelo espacio-temporal de los accidentes de tráfico en la ciudad de Houston sobre el network.

Para ello usaremos R y ArcGis. Este último nos proporcionará una amplia posibilidad de recursos relacionados con el análisis espacial de datos.

Para la creación del modelo disponemos de una base de datos que contiene los registros de los accidentes de tráfico de la ciudad de Houston desde el año 1999 hasta el 2001. Disponemos, por tanto, de 180.970 observaciones con 591 covariables asociadas a cada accidente. Además, disponemos también de otro archivo ESRI Shapefile (.shp) que almacena información espacial de las calles de la ciudad de Houston.

El formato ESRI Shapefile es un formato de archivo capaz de almacenar datos espaciales desarrollado por la compañía ESRI, creadores también de ArcGis.

2.1. Análisis y preparación de los datos

Esta ha sido, sin lugar a dudas, la parte más costosa del proyecto. Debido a la gran cantidad de información, cualquier intento de gestionar la base de datos y/o los archivos espaciales *.shp* ha conllevado un problema computacional (*big data*).

La información de la base de datos está registrada en dos archivos de la siguiente forma:

- el archivo *eventos.shp* contiene un objeto del tipo *SpatialPointsDataFrame* de *R* que localiza espacialmente todos los eventos. Además, en el slot *data* del objeto *SpatialPointsDataFrame* está almacenada la información acerca de las 591 covariables asociadas a cada accidente;
- el archivo *ejes.shp* contiene un objeto del tipo *SpatialLinesDataFrame* de *R* que localiza espacialmente las calles o ejes de Houston.

Este último archivo es el que ha presentado mayores dificultades ya que el objeto *SpatialLines-DataFrame* estaba mal construido. Dicho objeto estaba formado por pequeños segmentos y no por calles, de tal forma que estimar el número de eventos por eje no era posible. Resolver este problema de *big data* ha sido costoso.

La primera opción para resolver esta situación fue implementar un código en R encargado de recorrer el grafo y reconstruirlo; sin embargo, dicho código era capaz de reconstruir subconjuntos de Houston pero no Houston completo. Tras aplicar varias técnicas diferentes, pudimos resolver el problema insertando la información de los ejes de Houston en una base de datos y reconstuyendo el objeto *SpatialLinesDataFrame* mediante ArgGis.

Por último, notar que todos los eventos no estaban sobre los ejes. Con el fin de simplificar la tarea, añadimos un idenfiticador único a cada calle de tal forma que, mediante ArcGis, pudimos asociar cada evento a un eje y relacionarlos mediante dicho indentificador.

Análisis de los datos

Análisis de los datos

Una vez solventadas las cuestiones anteriores, queda examinar con detenimiento las características de nuestra base de datos.

El archivo *ejes.shp* contiene un objeto del tipo *SpatialLinesDataFrame* que localiza espacialmente los ejes de Houston. Gráficamente queda como sigue:

```
plot(capaEjes)
```

La base de datos de dicha capa es de dimension 4145×1, es decir, registra 4145 calles de Houston y a cada una de ellas le hemos asociado un identificador único llamado EID:

```r
length(capaEjes)
```

```
## [1] 4145
```

```r
names(capaEjes@data)
```

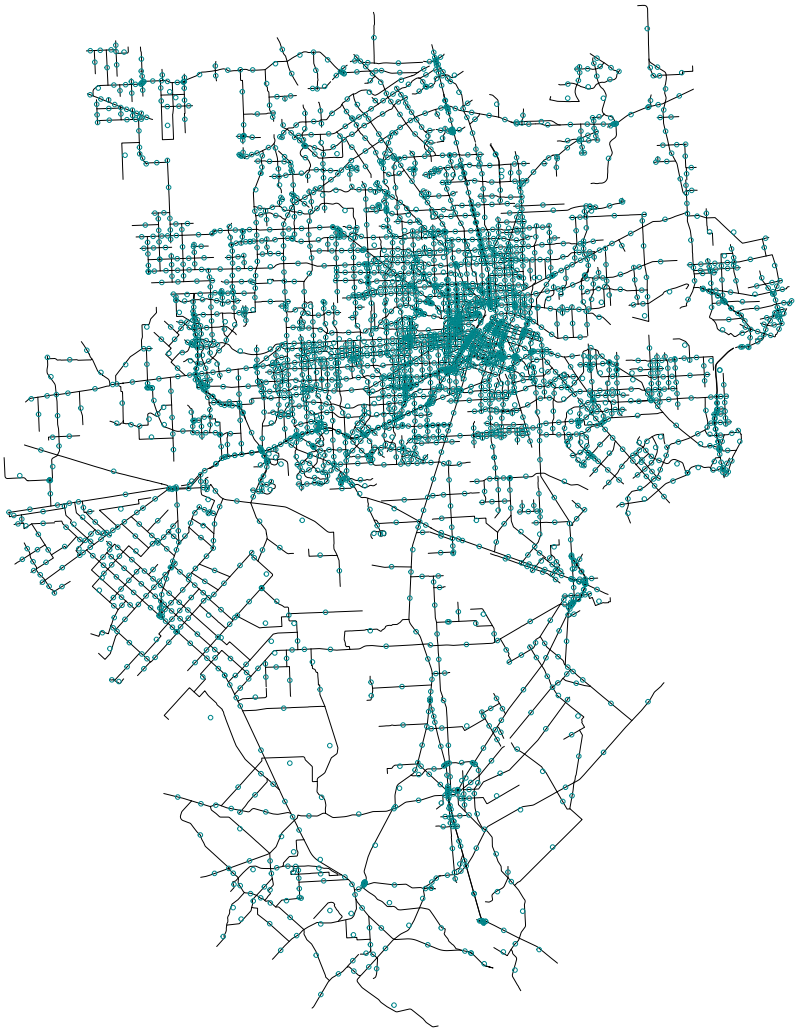
```
## [1] "EID"
```

```r
head(capaEjes@data)
```

```
##   EID
## 1   1
## 2   2
## 3   3
## 4   4
## 5   5
## 6   6
```

Por último, calcularemos el centroide de cada calle de tal forma que podamos comprobar y visualizar que hemos reconstruido el network correctamente:

```
centroides_capa = gCentroid(capaEjes, byid=TRUE)@coords
plot(capaEjes)
points(centroides_capa[,1],centroides_capa[,2], col="turquoise4")
```



Podemos ver en negro las calles de la ciudad de Houston y, en turquesa, los centroides de cada una de dichas calles.

El archivo *eventos.shp* contiene un objeto del tipo SpatialPointsDataFrame que localiza espacialmente todos los eventos. Además, en esta capa está almacenada la información acerca de las 591 covariables asociadas a cada accidente. Gráficamente, los accidentes están situados de la siguente forma:

```
plot(capaHechos, col="turquoise4")
```

```
dim(capaHechos@data)
```

```
## [1] 180970    593
```

La base de datos asociada a esta capa es de dimension 180970×593, esto es, en dicha base de datos están registrados 180.970 accidentes acontecidos en Houston durante el año 1999, 2000 y 2001. Las covariables asociadas a cada hecho son:

```r
names(capaHechos@data)
```

```
##   [1] "RECTYPE"    "DPSNUMBR"   "RECCNTRL"   "COUNTY"     "CITY"
##   [6] "DISTRICT"   "POPGROUP"   "ROADCLAS"   "MONTH"      "DAYMONTH"
##  [11] "WEEKDAY"    "TIME"       "LIGHT"      "HARMFUL1"   "SEVERITY"
##  [16] "WEATHER"    "SURFACE"    "ROADCOND"   "INVESTIG"   "ALIGNMNT"
##  [21] "TRAFCNTL"   "ONOFFRD"    "INTERSEC"   "INTRROAD"   "ENTRROAD"
##  [26] "VEHMOVE"    "OBJECT"     "OTHFACTR"   "MAINSTRT"   "BLOCKNUM"
##  [31] "ROADPART"   "CURVE"      "BRIDGNUM"   "BRIDGDET"   "DIRECTN1"
##  [36] "DIRECTN2"   "REFSTRT"    "CTYACCNO"   "RRCROSSN"   "CNTYCODE"
##  [41] "NUMVEH"     "YEAR"       "MSTREET"    "RSTREET"    "ADDRESS"
##  [46] "X"          "Y"          "RECCTRL1"   "VEHYEAR1"   "VEHMAKE1"
##  [51] "VEHBODY1"   "VEHTYPE1"   "DAMAGE1"    "DEFECT1"    "AGE1"
##  [56] "RACESEX1"   "LICENSE1"   "STATUS1"    "INSURNC1"   "DISABIL1"
##  [61] "FACTOR1A"   "FACTOR1B"   "VEHNUM1"    "SEVERE1"    "BELT1"
##  [66] "EJECTED1"   "INJPART1"   "INJBODY1"   "EMS1"       "HELMET1"
##  [71] "GOGGLE1"    "LENSCLR1"   "DUISPEC1"   "DUITEST1"   "CASUALT1"
##  [76] "VEHYEAR2"   "VEHMAKE2"   "VEHBODY2"   "VEHTYPE2"   "DAMAGE2"
##  [81] "DEFECT2"    "AGE2"       "RACESEX2"   "LICENSE2"   "STATUS2"
##  [86] "INSURNC2"   "DISABIL2"   "FACTOR2A"   "FACTOR2B"   "VEHNUM2"
##  [91] "SEVERE2"    "BELT2"      "EJECTED2"   "INJPART2"   "INJBODY2"
##  [96] "EMS2"       "HELMET2"    "GOGGLE2"    "LENSCLR2"   "DUISPEC2"
## [101] "DUITEST2"   "CASUALT2"   "RECCTRL3"   "VEHYEAR3"   "VEHMAKE3"
## [106] "VEHBODY3"   "VEHTYPE3"   "DAMAGE3"    "DEFECT3"    "AGE3"
## [111] "RACESEX3"   "LICENSE3"   "STATUS3"    "INSURNC3"   "DISABIL3"
## [116] "FACTOR3A"   "FACTOR3B"   "VEHNUM3"    "SEVERE3"    "BELT3"
## [121] "EJECTED3"   "INJPART3"   "INJBODY3"   "EMS3"       "HELMET3"
## [126] "GOGGLE3"    "LENSCLR3"   "DUISPEC3"   "DUITEST3"   "CASUALT3"
## [131] "VEHYEAR4"   "VEHMAKE4"   "VEHBODY4"   "VEHTYPE4"   "DAMAGE4"
## [136] "DEFECT4"    "AGE4"       "RACESEX4"   "LICENSE4"   "STATUS4"
## [141] "INSURNC4"   "DISABIL4"   "FACTOR4A"   "FACTOR4B"   "VEHNUM4"
## [146] "SEVERE4"    "BELT4"      "EJECTED4"   "INJPART4"   "INJBODY4"
## [151] "EMS4"       "HELMET4"    "GOGGLE4"    "LENSCLR4"   "DUISPEC4"
## [156] "DUITEST4"   "CASUALT4"   "RECCTRL5"   "VEHYEAR5"   "VEHMAKE5"
## [161] "VEHBODY5"   "VEHTYPE5"   "DAMAGE5"    "DEFECT5"    "AGE5"
## [166] "RACESEX5"   "LICENSE5"   "STATUS5"    "INSURNC5"   "DISABIL5"
## [171] "FACTOR5A"   "FACTOR5B"   "VEHNUM5"    "SEVERE5"    "BELT5"
## [176] "EJECTED5"   "INJPART5"   "INJBODY5"   "EMS5"       "HELMET5"
## [181] "GOGGLE5"    "LENSCLR5"   "DUISPEC5"   "DUITEST5"   "VEHYEAR6"
## [186] "VEHMAKE6"   "VEHBODY6"   "VEHTYPE6"   "DAMAGE6"    "DEFECT6"
## [191] "AGE6"       "RACESEX6"   "LICENSE6"   "STATUS6"    "INSURNC6"
## [196] "DISABIL6"   "FACTOR6A"   "FACTOR6B"   "VEHNUM6"    "SEVERE6"
## [201] "BELT6"      "EJECTED6"   "INJPART6"   "INJBODY6"   "EMS6"
## [206] "HELMET6"    "GOGGLE6"    "LENSCLR6"   "DUISPEC6"   "DUITEST6"
## [211] "CASUALT6"   "RECCTRL7"   "VEHYEAR7"   "VEHMAKE7"   "VEHBODY7"
## [216] "VEHTYPE7"   "DAMAGE7"    "DEFECT7"    "AGE7"       "RACESEX7"
## [221] "LICENSE7"   "STATUS7"    "INSURNC7"   "DISABIL7"   "FACTOR7A"
## [226] "FACTOR7B"   "VEHNUM7"    "SEVERE7"    "BELT7"      "EJECTED7"
## [231] "INJPART7"   "INJBODY7"   "EMS7"       "HELMET7"    "GOGGLE7"
## [236] "LENSCLR7"   "DUISPEC7"   "DUITEST7"   "CASUALT7"   "VEHYEAR8"
## [241] "VEHMAKE8"   "VEHBODY8"   "VEHTYPE8"   "DAMAGE8"    "DEFECT8"
## [246] "AGE8"       "RACESEX8"   "LICENSE8"   "STATUS8"    "INSURNC8"
## [251] "DISABIL8"   "FACTOR8A"   "FACTOR8B"   "VEHNUM8"    "SEVERE8"
## [256] "BELT8"      "EJECTED8"   "INJPART8"   "INJBODY8"   "EMS8"
```

```
## [261] "HELMET8"    "GOGGLE8"    "LENSCLR8"   "DUISPEC8"   "DUITEST8"
## [266] "CASUALT8"   "TRUCK"      "COMMVEH"    "PICKUP"     "PANELVAN"
## [271] "SUV"        "DUI"        "REDLIGHT"   "P_REDLIGHT" "SPEEDING"
## [276] "FAILSTOP"   "NOYIELD"    "BADTURN"    "TOOCLOSE"   "FROMPARK"
## [281] "NOPEDROW"   "NOSIGNAL"   "BADPASS"    "EMS"        "TEENAGE"
## [286] "TWENTIES"   "THIRTIES"   "FORTIES"    "FIFTIES"    "ELDERLY"
## [291] "CHILD"      "MALE"       "FEMALE"     "WHITE"      "BLACK"
## [296] "FATAL"      "TYPEA"      "TYPEB"      "TYPEC"      "PDO"
## [301] "TANKER"     "BUS"        "CELLPHON"   "RECCTRL2"   "RECCTRL4"
## [306] "PASSAGE1"   "PASSSEX1"   "PASSNUM1"   "PASSPOS1"   "PASSSEV1"
## [311] "PASSDEV1"   "PASSEJC1"   "PASSPRT1"   "PASSBOD1"   "PASSEMS1"
## [316] "PASSAGE2"   "PASSSEX2"   "PASSNUM2"   "PASSPOS2"   "PASSSEV2"
## [321] "PASSDEV2"   "PASSEJC2"   "PASSPRT2"   "PASSBOD2"   "PASSEMS2"
## [326] "PASSAGE3"   "PASSSEX3"   "PASSNUM3"   "PASSPOS3"   "PASSSEV3"
## [331] "PASSDEV3"   "PASSEJC3"   "PASSPRT3"   "PASSBOD3"   "PASSEMS3"
## [336] "PASSAGE4"   "PASSSEX4"   "PASSNUM4"   "PASSPOS4"   "PASSSEV4"
## [341] "PASSDEV4"   "PASSEJC4"   "PASSPRT4"   "PASSBOD4"   "PASSEMS4"
## [346] "NONPASS1"   "BACTEST1"   "PEDACTN1"   "PEDALCO1"   "PEDVIOL1"
## [351] "PASSAGE5"   "PASSSEX5"   "PASSNUM5"   "PASSPOS5"   "PASSSEV5"
## [356] "PASSDEV5"   "PASSEJC5"   "PASSPRT5"   "PASSBOD5"   "PASSEMS5"
## [361] "NONPASS2"   "BACTEST2"   "PEDACTN2"   "PEDALCO2"   "PEDVIOL2"
## [366] "PASSAGE6"   "PASSSEX6"   "PASSNUM6"   "PASSPOS6"   "PASSSEV6"
## [371] "PASSDEV6"   "PASSEJC6"   "PASSPRT6"   "PASSBOD6"   "PASSEMS6"
## [376] "PASSAGE7"   "PASSSEX7"   "PASSNUM7"   "PASSPOS7"   "PASSSEV7"
## [381] "PASSDEV7"   "PASSEJC7"   "PASSPRT7"   "PASSBOD7"   "PASSEMS7"
## [386] "PASSAGE8"   "PASSSEX8"   "PASSNUM8"   "PASSPOS8"   "PASSSEV8"
## [391] "PASSDEV8"   "PASSEJC8"   "PASSPRT8"   "PASSBOD8"   "PASSEMS8"
## [396] "PASSAGE9"   "PASSSEX9"   "PASSNUM9"   "PASSPOS9"   "PASSSEV9"
## [401] "PASSDEV9"   "PASSEJC9"   "PASSPRT9"   "PASSBOD9"   "PASSEMS9"
## [406] "NONPASS3"   "BACTEST3"   "PEDACTN3"   "PEDALCO3"   "PEDVIOL3"
## [411] "PASSAGE10"  "PASSSEX10"  "PASSNUM10"  "PASSPOS10"  "PASSSEV10"
## [416] "PASSDEV10"  "PASSEJC10"  "PASSPRT10"  "PASSBOD10"  "PASSEMS10"
## [421] "NONPASS4"   "BACTEST4"   "PEDACTN4"   "PEDALCO4"   "PEDVIOL4"
## [426] "PASSAGE11"  "PASSSEX11"  "PASSNUM11"  "PASSPOS11"  "PASSSEV11"
## [431] "PASSDEV11"  "PASSEJC11"  "PASSPRT11"  "PASSBOD11"  "PASSEMS11"
## [436] "PASSAGE12"  "PASSSEX12"  "PASSNUM12"  "PASSPOS12"  "PASSSEV12"
## [441] "PASSDEV12"  "PASSEJC12"  "PASSPRT12"  "PASSBOD12"  "PASSEMS12"
## [446] "PASSAGE13"  "PASSSEX13"  "PASSNUM13"  "PASSPOS13"  "PASSSEV13"
## [451] "PASSDEV13"  "PASSEJC13"  "PASSPRT13"  "PASSBOD13"  "PASSEMS13"
## [456] "PASSAGE14"  "PASSSEX14"  "PASSNUM14"  "PASSPOS14"  "PASSSEV14"
## [461] "PASSDEV14"  "PASSEJC14"  "PASSPRT14"  "PASSBOD14"  "PASSEMS14"
## [466] "NONPASS5"   "BACTEST5"   "PEDACTN5"   "PEDALCO5"   "PEDVIOL5"
## [471] "PASSAGE15"  "PASSSEX15"  "PASSNUM15"  "PASSPOS15"  "PASSSEV15"
## [476] "PASSDEV15"  "PASSEJC15"  "PASSPRT15"  "PASSBOD15"  "PASSEMS15"
## [481] "NONPASS6"   "BACTEST6"   "PEDACTN6"   "PEDALCO6"   "PEDVIOL6"
## [486] "CHILD1"     "CHILD2"     "CHILD3"     "CHILD4"     "CHILD5"
## [491] "CHILD6"     "CHILD7"     "CHILD8"     "CHILD9"     "CHILD10"
## [496] "CHILD11"    "CHILD12"    "CHILD13"    "CHILD14"    "CHILD15"
## [501] "KILLED1"    "KILLED2"    "KILLED3"    "KILLED4"    "KILLED5"
## [506] "KILLED6"    "KILLED7"    "KILLED8"    "KILLED9"    "KILLED10"
## [511] "KILLED11"   "KILLED12"   "KILLED13"   "KILLED14"   "KILLED15"
## [516] "INJURED1"   "INJURED2"   "INJURED3"   "INJURED4"   "INJURED5"
## [521] "INJURED6"   "INJURED7"   "INJURED8"   "INJURED9"   "INJURED10"
## [526] "INJURED11"  "INJURED12"  "INJURED13"  "INJURED14"  "INJURED15"
```

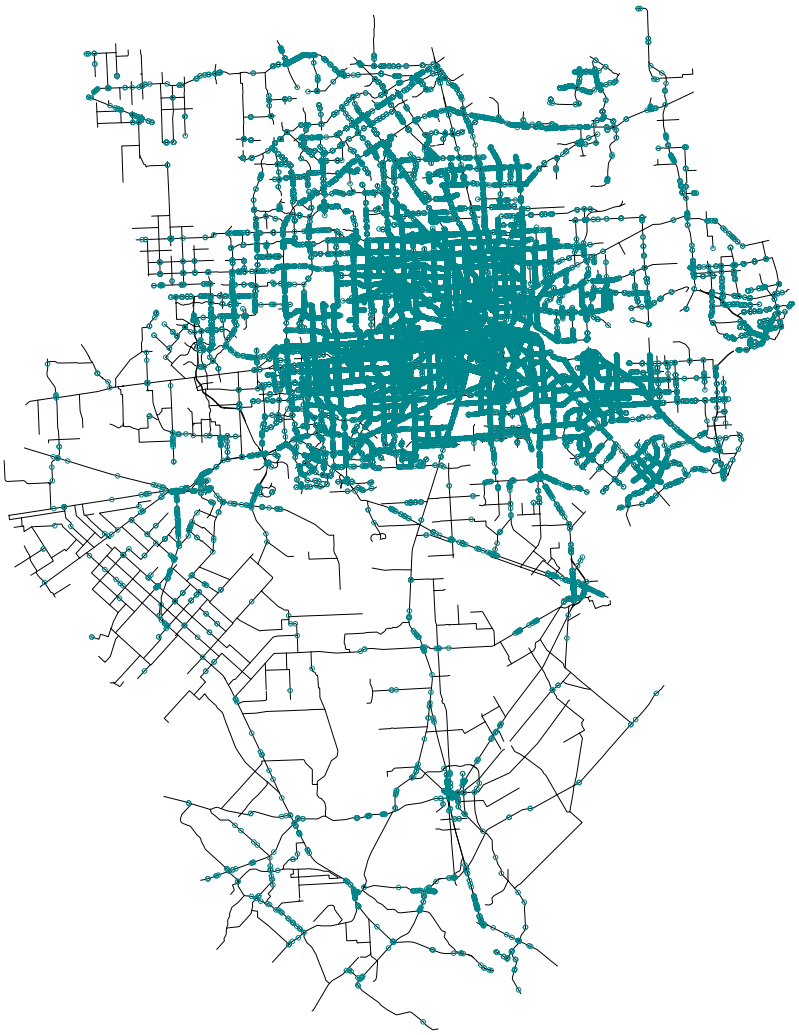
```
## [531] "DEVICE1"    "DEVICE2"    "DEVICE3"    "DEVICE4"    "DEVICE5"
## [536] "DEVICE6"    "DEVICE7"    "DEVICE8"    "DEVICE9"    "DEVICE10"
## [541] "DEVICE11"   "DEVICE12"   "DEVICE13"   "DEVICE14"   "DEVICE15"
## [546] "NODEVICE"   "DUMMY"      "FNODE_"     "TNODE_"     "LPOLY_"
## [551] "RPOLY_"     "LENGTH"     "ROAD_"      "ROAD_ID"    "PRETYPE"
## [556] "NAME"       "BOUND"      "GDC_FLAG"   "ONEWAY"     "STARMAP_ID"
## [561] "SM_SUBTYPE" "GDC_CLASS"  "LABEL_FIEL" "LOWER_RIGH" "LOWER_LEFT"
## [566] "UPPER_RIGH" "UPPER_LEFT" "PREFIX"     "ST_TYPE"    "SUFFIX"
## [571] "CITY_RIGHT" "CITY_LEFT"  "ZIP_RIGHT"  "ZIP_LEFT"   "PARITY_RIG"
## [576] "PARITY_LEF" "LAMBERT"    "DOQQ"       "TIGER90_ID" "FHWA_CLASS"
## [581] "ADD_DATE"   "CHANGE_DAT" "EDITOR"     "SOURCE"     "LENGTH_MIL"
## [586] "dummy_2"    "SHAPE_LEN"  "i"          "j"          "networkGrp"
## [591] "GID"        "diftime"    "EID"
```

Notar que las dos últimas covariables, $diftime$ y EID, son variables artificiales creadas por nosotros y que nos ayudarán a modelizar el problema de tal forma que:

- $diftime$ es una variable entera que almacena el día que se produjo el accidente siendo el valor 1 correspondiente al *1 de enero de 1999* y el valor 1096 correspondiente al *31 de diciembre de 2001*;
- EID es el identificador del eje en el cual se ha producido el accidente. Esta covariable nos simplificará la tarea de reubicar cada evento sobre un eje.

Por tanto, superponiendo ambas capas queda como sigue:

```
plot(capaEjes)
points(capaHechos, col="turquoise4")
```



En negro, las calles de la ciudad de Houston. En turquesa, los accidentes ocurridos desde 1999 hasta 2001.

Por último, una vez detallada la naturaleza y estructura de la base de datos original, estamos en condiciones de calcular de una forma sencilla (véase el código en el anexo) las funciones de intensidad espacial (número accidentes por eje) y temporal (número de accidentes por día) empíricas, que serán las variables respuesta de los modelos.

Función de intensidad espacial

```
fIntEmpiricaEsp[1:100]
```

```
##   [1]  36 224 136   0   3 174 160  32  71  30   7   0 145  93   2  29  26
##  [18] 140  12   7  17   7  66  29   1  26   0   0   8   8  20   9   8   0
##  [35]  36  79   0   0   0  17  20  20   2  10  76 116  44   0   5  86   8
##  [52]   8  14  20  37   0  32   8   0   0  12   8  20  28 139   0   7  21
##  [69]   0  22   0   0  14   0   3  41 125  12   0  16   0 100   0   0  11
##  [86]  13   0  16  65 106   0   0   1  26   0  56  24  12   4 100
```

donde la coordenada $i-$ésima del vector representa el número de accidentes ocurridos en el eje i durante los 1096 días.

Función de intensidad temporal

```
fIntEmpiricaTemp[1:100]
```

```
##   [1] 177 106  73 134 138 186 218 181 130 111 149 121 223 151 141 111 150
##  [18]  91 129 139 142 178 145  78 139 155 177 247 232 176  96 154 113 142
##  [35] 116 178 129 116 138 109 137 166 185 192 126 135 140 132 138 143 203
##  [52] 116 135 174 155 142 126 234 144 138 119 158 119 154 230 126 178 198
##  [69] 124 174 347 159  99 159 147 151 103 295 164 123 167 124 149 128 195
##  [86] 121 122 323 257 145 157 183 173 132 157 167 156 134 170 173
```

donde la coordenada $j-$ésima del vector representa el número de accidentes durante el día j en toda la ciudad de Houston.

2.2. Desarrollo teórico del modelo

En lo que sigue, nuestro objetivo es modelar la función de intensidad $\lambda(\mathbf{s}, t),\ \mathbf{s} \in W,\ t \in T$ dependiente del espacio y del tiempo, esto es, el número de accidentes por unidad de área y tiempo.

Asumiendo la separabilidad espacio-temporal de primer orden vista en 1,2,2., podemos reescribir $\lambda(\mathbf{s}, t)$ de la siguiente forma:

$$\lambda(\mathbf{s}, t) = \lambda(\mathbf{s}) \cdot \lambda(t), \quad \mathbf{s} \in W,\ t \in T, \tag{2.1}$$

reduciendo el problema original a

1. crear un modelo espacial que estime la función de intensidad espacial $\lambda(\mathbf{s}),\ \mathbf{s} \in W$, es decir, número de accidentes por unidad de área;
2. crear un modelo temporal capaz de estimar la función de intensidad temporal $\lambda(t),\ t \in T$, es decir, número de eventos por unidad de tiempo.

2.3. Modelo temporal

Mediante este modelo queremos estimar la función de intensidad temporal, es decir, el número de eventos por unidad de tiempo. En nuestro caso, tomaremos como unidad de tiempo el día. Por tanto, el objetivo es obtener $\lambda_{estimada}(t),\ t \in T$ de tal forma que:

$$\lambda_{empirica}(t) = \lambda_{estimada}(t) \cdot errorTemporal(t), \quad t \in T, \tag{2.2}$$

siendo *errorTemoral(t)* una función dependiente del tiempo lo más *pequeña* posible.

Para ello, supondremos que el modelo idóneo para modelizar la función de intesidad temporal $\lambda_{empirica}(t)$ es el *Modelo lineal geranalizado*, *GLM*, para recuentos donde la variable respuesta Y es el número de accidentes por día, la función de distribución es la Poisson y el componente sistémico o variables explicativas son: estado de la superfície de la carretera (*Sup*), día de la semana (*DiaSemana*), visibilidad (*Luz*), mes del año (*Mes*), hora a la que se produce el accidente (*Hora*), estado de la carrtera (*EstadoCarretera*), estado del tiempo atmosférico (*Tiempo*) y una función seno y coseno definida como sen $\left(\frac{\text{día en el que se produce accidente}}{\text{total de días}}\right)$ y cos $\left(\frac{\text{día en el que se produce accidente}}{\text{total de días}}\right)$.

Para crear el modelo, empezamos suponiendo que sólo puede existir una varibale explicativa. A continuación, vemos cuál de todas ellas minimiza el error cuadrático, es decir, cuál de todas ellas explica mejor la variable respuesta Y y fijamos dicha variable. Ahora, suponemos que existen dos variables explicativas. Fijada la primera variable, vemos cuál es el modelo que minimiza el error cuadrático siendo la primera variable explicativa la fijada en el paso primero y la segunda cualquiera de las restantes. Fijamos la segunda variable. Procediendo de esta forma obtenemos que los mejores modelos son lo siguientes:

Modelo temporal		
$\lambda(t)$	$\sim$	Sup
$\lambda(t)$	$\sim$	Sup+DiaSemana
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine+Luz
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine+Luz+Mes
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine+Luz+Mes+cosine
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine+Luz+Mes+cosine+Hora
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine+Luz+Mes+cosine+Hora+EstadoCarretera
$\lambda(t)$	$\sim$	Sup+DiaSemana+sine+Luz+Mes+cosine+Hora+EstadoCarretera+Tiempo

Finalmente, el modelo que minimiza el error cuadrático es

$$\lambda(t) \sim \text{Sup+DiaSemana+sine+Luz+Mes+cosine+Hora+EstadoCarretera+Tiempo}$$

y el valor de dicho error es 0.03405221 teniendo el modelo una bondad de ajuste de 0.8546831. Esto quiere decir que, de cada 100 accidentes, en 85 de ellos clasifica correctamente el día en el cual acontenció dicho evento.

Gráficamente, la serie temporal y la función *errorTemoral(t)* queda como sigue:

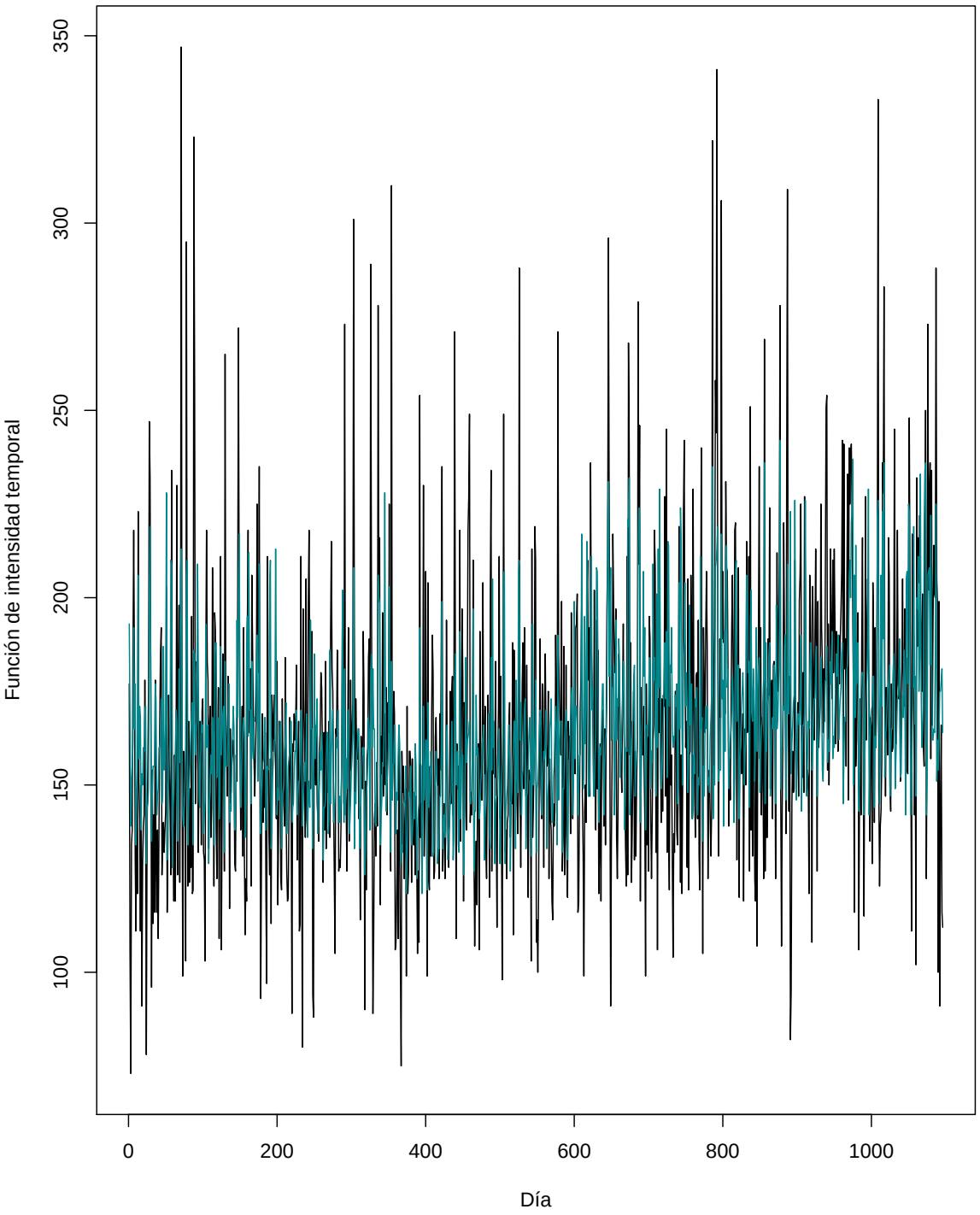


Figura 2.1: En negro, la serie temporal empírica. En turquesa, la serie temporal estimada.

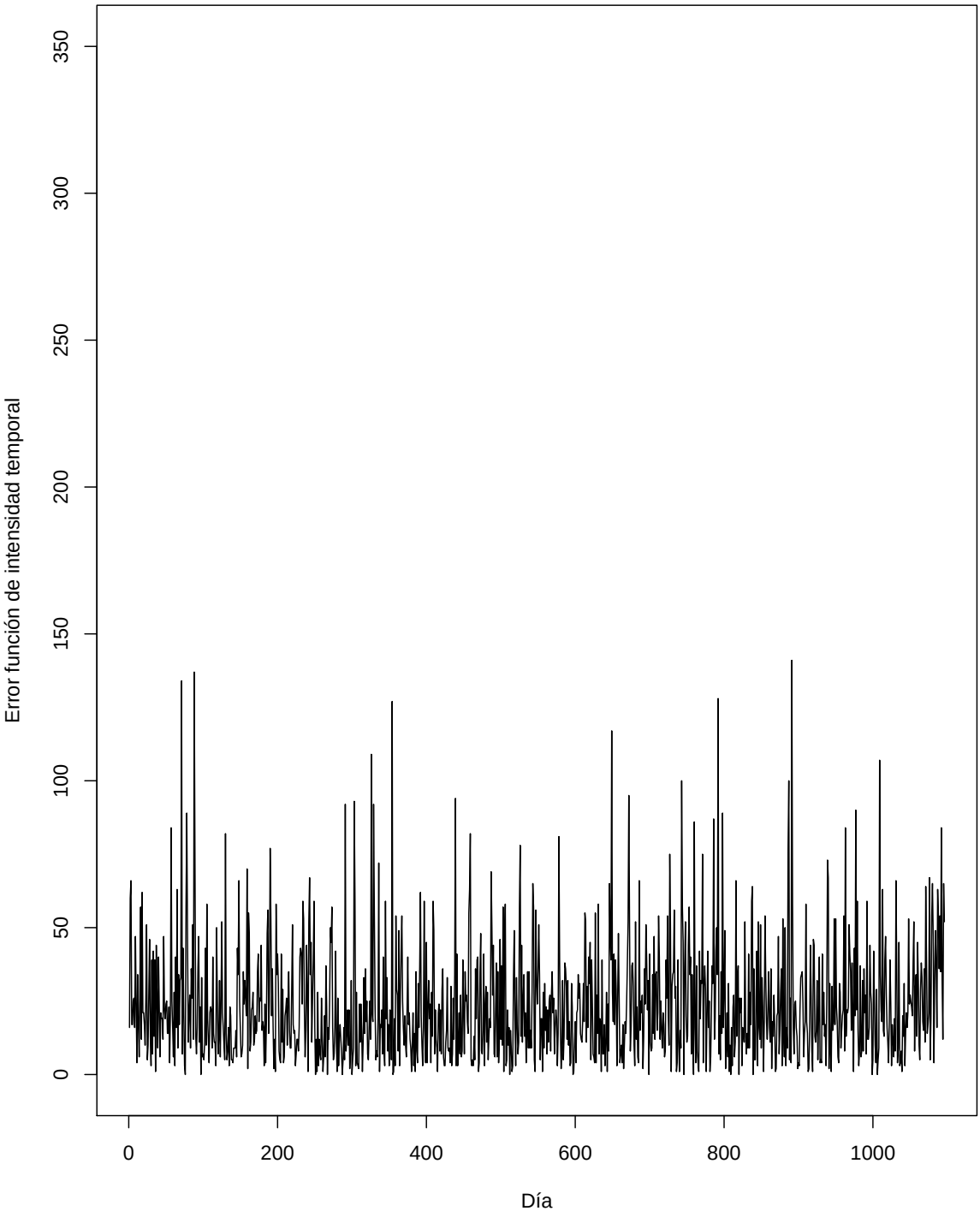


Figura 2.2: En negro, la función *errorTemoral(t)*, $t \in T$.

2.4. Modelo espacial

Mediante este modelo queremos estimar la función de intensidad espacial, es decir, el número de eventos por unidad de área. En nuestro caso, tomaremos como unidad de área cada una de las calles o ejes de Houton. Por tanto, el objetivo es obtener $\lambda_{estimada}(\mathbf{s}),\ \mathbf{s} \in W$ de tal forma que:

$$\lambda_{empirica}(\mathbf{s}) = \lambda_{estimada}(\mathbf{s}) \cdot errorEspacial(\mathbf{s}), \quad \mathbf{s} \in W, \tag{2.3}$$

siendo la función errorEspacial(**s**) lo más *pequeña* posible.

Para ello, supondremos que el modelo idóneo para modelizar la función de intesidad espacial $\lambda_{empirica}(\mathbf{s})$ es el *Modelo aditivo geranalizado*, *GAM*, para recuentos donde la variable respuesta Y es el número de accidentes por eje, la función de distribución es la Poisson y el componente sistémico o variables explicativas son las covariables espaciales sin datos faltantes. Procediendo de la misma forma que en el modelo temporal, el modelo que minimiza el error cuadrático es

$\lambda(\mathbf{s}) \sim$ FEMALE + RACESEX1 + INSURNC1 + INTERSEC + NOYIELD + SPEEDING + BLACK + POPGROUP + ROADCLAS + TRUCK + ENTRROAD + PICKUP + EMS + TEENAGE + BADTURN + INVESTIG + PANELVAN + BADPASS + NOSIGNAL + FROMPARK + P_REDLIGHT + REDLIGHT + FATAL + VEHNUM1

y el valor de dicho error es 0.968989 teniendo el modelo una bondad de ajuste de 0.3486804.

Notar que hemos realizado un modelo alternativo con 134.904 observaciones pero con 20 covariables más eliminando observaciones con datos faltantes. Algunas de estas variables añadidas explican de una forma significativa la variables respuesta Y, de tal forma que la bondad de ajuste de este modelo

$\lambda(\mathbf{s}) \sim$ "VEHBODY2 + FEMALE + NOYIELD + AGE1 + ENTRROAD + TRUCK + FACTOR1A + INSURNC2 + EMS + P_REDLIGHT + BADTURN + INVESTIG + VEHTYPE2 + AGE2 + TRAFCNTL + PANELVAN + SUV + RACESEX1 + PICKUP + TYPEC + RACESEX2 + WHITE + ROADCLAS + TWENTIES + VEHYEAR2 + NOSIGNAL + POPGROUP + TANKER + CHILD + FATAL + FROMPARK + FACTOR2B + BUS + DISABIL1 + CELLPHON + FAILSTOP + REDLIGHT + DISABIL2 + VEHNUM2 + VEHNUM1 + NOPEDROW + VEHMAKE2 + INTRROAD + VEHYEAR1 + ONOFFRD + BLACK + SPEEDING + INTERSEC + VEHMOVE + TEENAGE + SEVERE2 + SEVERITY + MALE + STATUS2

es 0.4390455.

A diferencia del modelo temporal, el ajuste del modelo espacial no es bueno a priori. Sin embargo, podemos ver gráficamente como la función de intensidad estimada es proporcional a la empírica de tal forma que, valores bajos o altos en $\lambda_{empirica}(\mathbf{s})$, produce valores bajos o altos en $\lambda_{estimada}(\mathbf{s})$, respectivamente. Gráficamente el modelo y el error de dicho modelo queda como sigue:

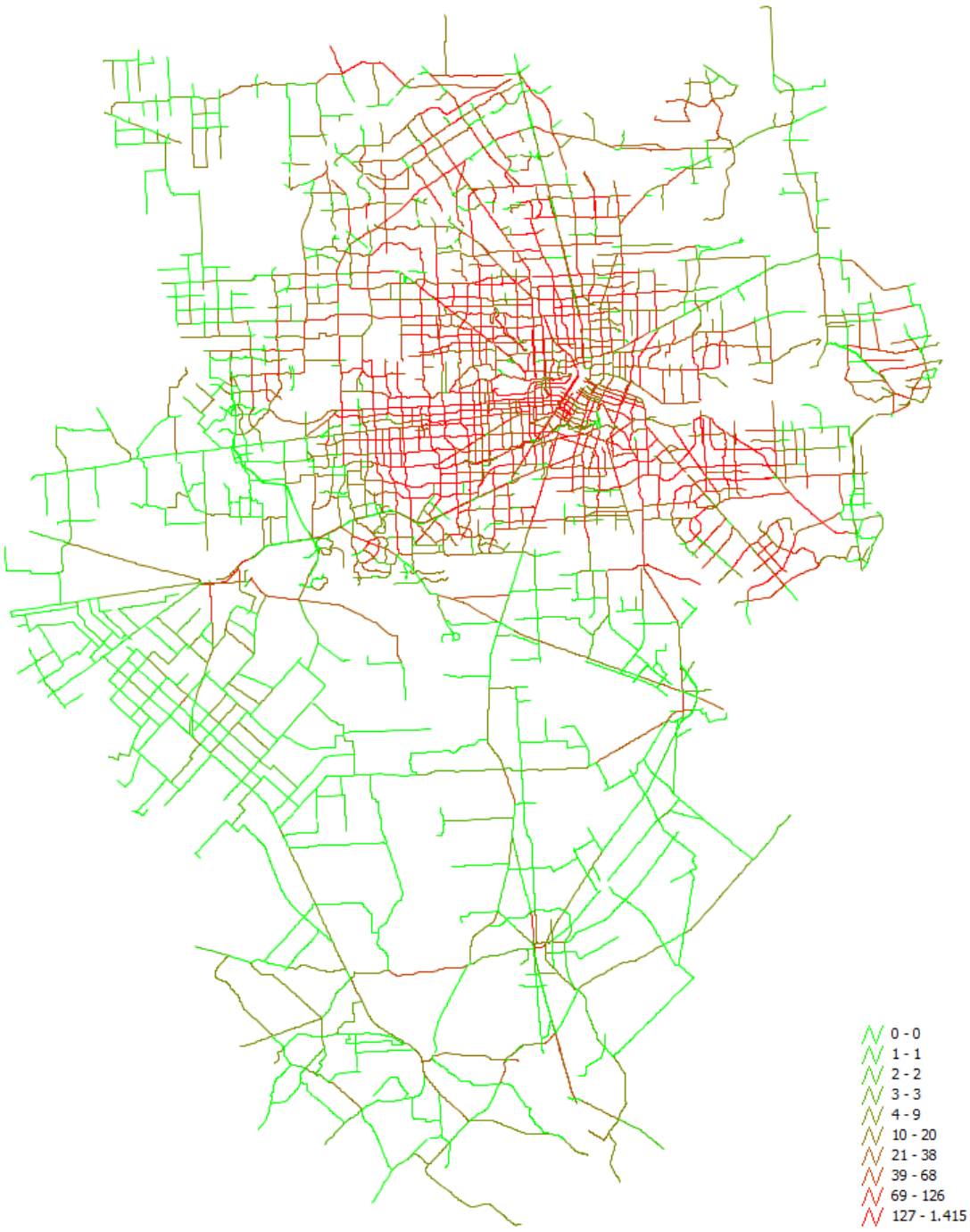


Figura 2.3: Función de intensidad espacial empírica, $\lambda_{empirica}(\mathbf{s})$, $\mathbf{s} \in \mathrm{W}$

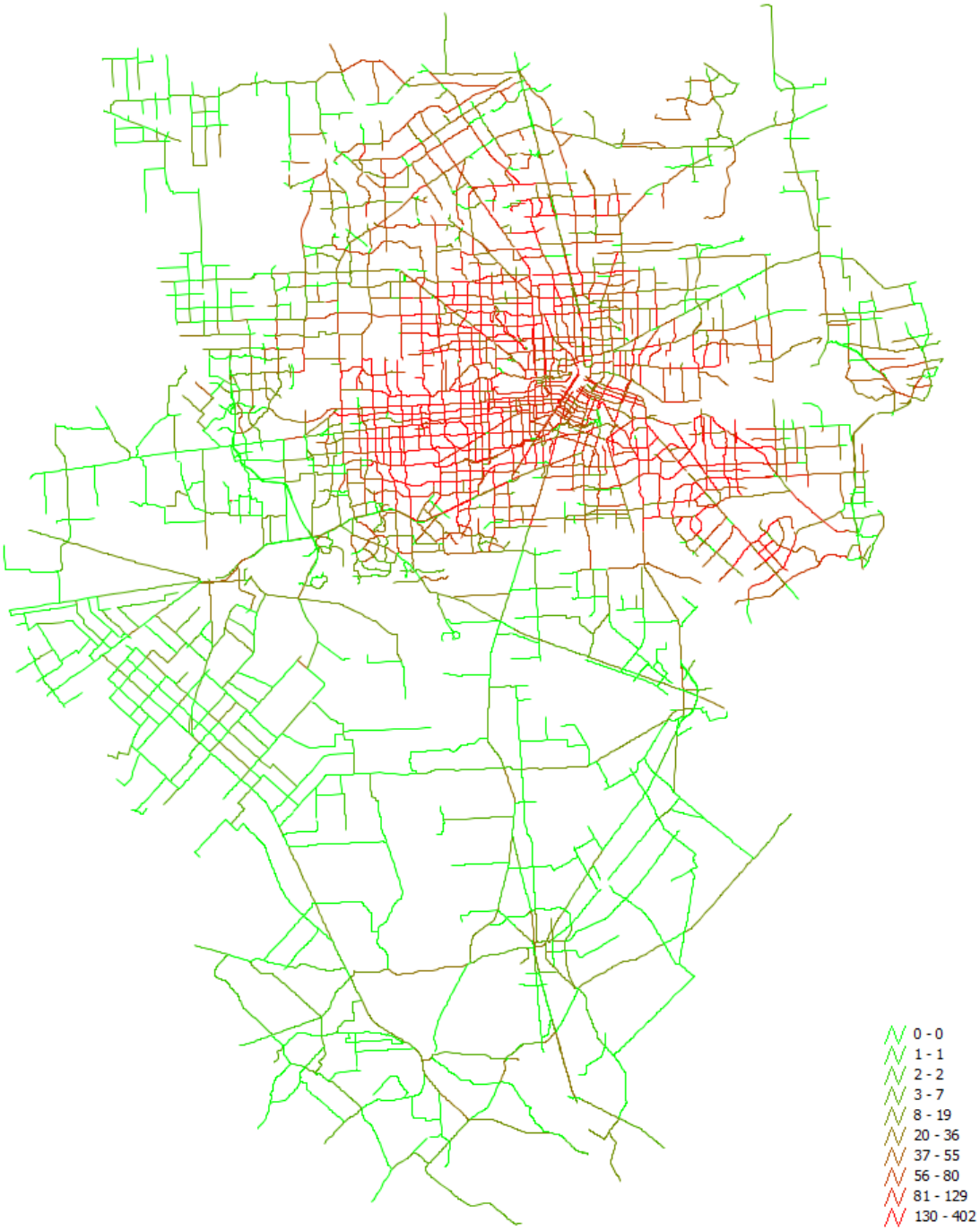


Figura 2.4: Función de intensidad espacial estimada, $\lambda_{estimada}(\mathbf{s})$, $\mathbf{s} \in \mathrm{W}$

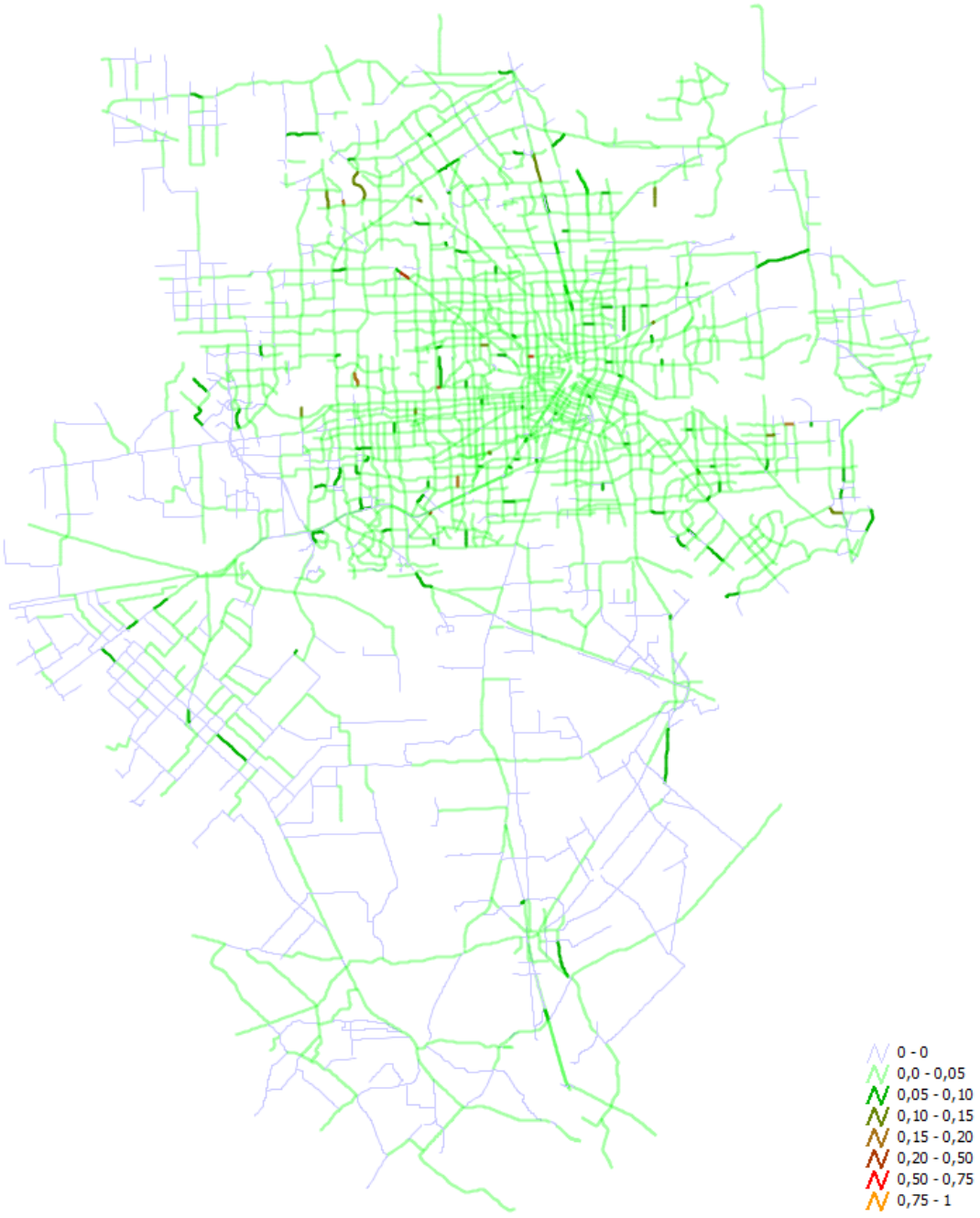


Figura 2.5: Función errorEspacial($\mathbf{s}$), $\mathbf{s} \in$ W

2.5. Modelo espacio-temporal

Hasta el momento, hemos modelado:

- la función de intensidad temporal $\lambda(t)$ de tal forma que

$$\lambda_{empirica}(t) = \lambda_{estimada}(t) \cdot errorTemporal(t), \quad t \in T, \tag{2.4}$$

con $\lambda_{empirica}(t),\ \lambda_{estimada}(t) \in \mathbb{R}^{1\times 1096}$ ya que la base de datos contiene observaciones de 1096 días (año 1999, 2000 y 2001);

- la función de intensidad espacial $\lambda(\mathbf{s})$ de tal forma que

$$\lambda_{empirica}(\mathbf{s}) = \lambda_{estimada}(\mathbf{s}) \cdot errorEspacial(\mathbf{s}), \quad \mathbf{s} \in W, \tag{2.5}$$

con $\lambda_{empirica}(\mathbf{s}),\ \lambda_{estimada}(\mathbf{s}) \in \mathbb{R}^{1\times 4145}$ ya que el archivo *ejes.shp* registra 4145 calles de la ciudad de Houston.

.

Por tanto, el modelo espacio-temporal es como sigue:

$$\begin{aligned}\lambda_{emp}(\mathbf{s},t) = \lambda_{emp}(\mathbf{s}) \cdot \lambda_{emp}(t) = (\lambda_{est}(\mathbf{s}) \cdot errorEspacial(\mathbf{s})) \cdot (\lambda_{est}(t) \cdot errorTemporal(t)) = \\ = (\lambda_{est}(\mathbf{s}) \cdot \lambda_{est}(t)) \cdot (errorEspacial(\mathbf{s}) \cdot errorTemporal(t)) = \\ = \lambda_{est}(\mathbf{s},t) \cdot errorModelo(\mathbf{s},t) \quad \mathbf{s},t \in W,\ t \in T.\end{aligned}$$

En conclusión, la función de intensidad $\lambda(\mathbf{s},t)$ que estima nuestro modelo espacio-temporal es el producto de $T(\lambda_{estimada}(\mathbf{s})) \in \mathbb{R}^{4145\times 1}$ y $\lambda_{estimada}(t) \in \mathbb{R}^{1\times 1096}$, esto es, la función espacio-temporal estimada $\lambda_{estimada}(\mathbf{s},t) \in \mathbb{R}^{1096\times 4145}$ es una matriz de 4145 filas y 1096 columnas en la cual la posición (i,j) indica el número de accidentes que acontecieron en la calle de Houston con identificador i durante el día j.

Este modelo tiene una bondad de ajuste de 0.3486804 debido al error cometido en el modelo espacial. Pese a esto, de la misma forma que ocurría en dicho modelo, valores bajos o altos en $\lambda_{emp}(\mathbf{s},t)$ produce valores bajos o altos en $\lambda_{est}(\mathbf{s},t)$.

Teniendo en cuenta que el objetivo final del estudio es predecir las áreas o ejes con más riesgo de accidentes en tiempos futuros podemos aseverar que el modelo es válido.

2.6. Error del modelo

Denotemos por Z(**s**,t) al error del modelo, sea $Z(\mathbf{s},t) = \exp(A(\mathbf{s},t))$ y $\lambda_{emp}(\mathbf{s},t) = \lambda_{est}(\mathbf{s},t) \cdot Z(\mathbf{s},t)$. Tomando logarítmos a ambos costados de la expresión obtenemos que

$$A(\mathbf{s},t) = log(\lambda_{emp}(\mathbf{s},t)) - \log(\lambda_{est}(\mathbf{s},t)).$$

De esta forma es computacionalmente sencillo obtener $Z(\mathbf{s},t)$ sin tener que dividir matrices. Notar que la matriz $Z(\mathbf{s},t)$ es de la misma dimensión que la función de intesidad espacio-temporal estimada.

2.7. Función predict temporal

Una vez construido el modelo espacio-temporal, nos interesa estimar el número de accidentes en tiempos futuros. Es razonable pensar que, en esta situación, una predicción espacial no tiene lugar ya que significaría añadir nuevas calles a la ciudad de Houston.

Mediante la función *predict.glm* de *R* podemos predecir fácilmente una nueva observación, incluso revalidar nuestro modelo. Para ello, podemos retirar una o varias observaciones del modelo e intentar predecirlas. En nuestro caso, al predecir el último día del modelo obtenemos un valor estimado de 115 accidentes a lo largo de dicho día mientras que el dato real son 112. Esto vuelve a poner de manifiesto que hemos creado un modelo válido.

Capítulo 3

Conclusiones y futuras líneas de investigación

3.1. Conclusiones

- Una de las partes más importantes a la hora de modelar es la recogida de datos. El hecho de disponer de información completa y bien estructurada hace que podamos obtener, de forma más sencilla, un modelo ajustado a la realidad. En nuestro caso, reestructurar y adecuar la información recabada por la policía de Houston ha sido la parte más costosa y dificultosa del proyecto, habiendo llegado a tener que descartar ejes inconexos, covariables releventes asociados a los eventos por datos faltantes,... Solventar estas cuestiones a supuesto un reto a nivel computacional.
- No hay una solución única ni óptima para resolver nuestro problema. Existen multitud de modelos posibles que son capaces de modelar la realidad de los accidentes de tráfico en Houston. También hay diversas formas de seleccionar las covariables más descriptivas, muchas combinaciones posibles de dichas covariables,... Siempre existen alternativas para tratar de mejorar el modelo.

3.2. Futuras líneas de invesitgación

Tras la exposción de este proyecto final de máster seguiremos trabajando en la mejora de este modelo. Algunas de las cuestiones a ampliar son:

- Mejorar y reajustar el modelo espacial. Para ello podemos
  - examinar de forma más exhaustiva si hay sobredispersión en los datos;
  - analizar si otras distribuciones pueden ajustarse mejor a la realidad espacial, como la binomial negativa;
  - reestructurar de forma distinta las covariables usadas en la creación del modelo con el fin de utilizar el máximo de la información disponible;

- Reacondicionar el código de tal forma que se puedan añadir observaciones a la base de datos y que esta nueva informacion pueda usarse para alimentar al modelo. Esta parte será interesante, más adelante, a la hora de trabajar con bases de datos actuales de interés policial de tal forma que, la nueva información que se introduzca en dicha base de datos, pueda ser usada para crear modelos actuales y más ajustados.
- Formalizar la función estocástica del residuo a través de la estructura de convarianza espacio-temporal.

Bibliografía

[1] Aragó, P.; Salvador, P.; Juan P.; Díaz-Avalos, C.; Serra, L.; Sáez, M. VARGA, D.5 TRILLES, S.6. MATEU J.2. Aplicación de modelos de procesos puntuales para la caracterizacón espacio-temporal del régimen de incendios en el este de espa´ a (provincias de castellón y girona). https://www.google.es/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwiVyvfhi5bWAhVE6RQKHVEID68QFggvMAA&url=https%3A%2F%2Fs3-eu-west-1.amazonaws.com%2Fpfigshare-u-files%2F999758%2Fppuntualesincendios.pdf&usg=AFQjCNFMIbYp6ZUfiIzZv1vt7GepqH0P-Q.

[2] Armero, Carmen. Modelos aditivos generalizados, gam. https://www.google.es/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwi7_LWu2ZzWAhXIfhoKHZgTAxEQFggtMAA&url=http%3A%2F%2Fwww.uv.es%2Farmero%2Ftemes_msuavitzats%2Ftema2.pdf&usg=AFQjCNFvBG-2WBDbscAeUeKfNklsIfNcew.

[3] P.J. Berman, M.; Diggle. Estimating weighted integrals of the second-order intensity of a spatial point process.

[4] Choi, E.; Hall,P. On the estimation of poles in intensity functions. Biometrika 87,2:251-263.

[5] D. Cox. Some statistical methods connected with series of events.

[6] Daley, D.J.; Vere-Jones, D. . An introduction to the theory of point processes. 2. volume first: Elementary theory and methods. Springer.

[7] Daley, D.J.; Vere-Jones, D. . An introduction to the theory of point processes. 2.volume second: General theory and structure. Springer.

[8] E. Diggle, P.J.; Gabriel. Second-order analysis of inhomogeneous spatiotemporal point process data.

[9] I.; Abellana R. Diggle, P.; Kaimi. Partial likelihood analysis of spatio-temporal point process data.

[10] P.J. Diggle. A kernel method for smoothing point process data.

[11] Diggle, P.J.; Høggkvist, R. . Second-order analysis of space-time clustering. Statistical Methods in Medical Research 4:124-136.

[12] B.; Diggle P.J. Gabriel, E.; Rowlingson. stpp: An r package for plotting, simulating and analysing spatio-temporal point patterns.

[13] Haas, T.C. . Local prediction of a spatio-temporal process with an application to wet sulfate deposition. Journal of the American Statistical Assoctiation 90,432:1189-1199.

[14] R. Hastie, T; Tibshirani. The elements of statistical learning.

[15] Kooperberg, C.; O'Sullivan, F. . Predictive oscillation patterns: A synthesis of methods for spatial-temporal decomposition of random fields. Journal of the American Statistical Association 91,436:1485-1496.

[16] Marín Diazaraque, Juan Miguel. Modelos lineales generalizados. https://www.google.es/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwjSzoD52pzWAhWGOhoKHeCRCz4QFggtMAA&url=http%3A%2F%2Fhalweb.uc3m.es%2Fesp%2FPersonal%2Fpersonas%2Fjmmarin%2Fesp%2FCategor%2FTema3Cate.pdf&usg=AFQjCNHDNSdyTFMlZvwVpIHGXvZXkNSGew.

[17] Møller, J.; Ghorbani, G . Aspects of second-order analysis of structured inhomogeneous spatio-temporal point processes. Statistica Neerlandica 66:472-491.

[18] Ogata, Y. Seismicity analysis through point-process modeling: a review. Pure and applied Geophysics 155:471-507.

[19] Rathbun, S.L.; Cressie, N.A.C. . A space-time survival point procss for a longleaf pine forest in southern georgia. Journal of the American Statistical Association 89,428:1164-1174.

Código para la creación del modelo espacio-temporal de los accidentes de tráfico en la ciudad de Houston

Javier Calahorra Tovar

11 de septiembre de 2017

```
#####################################################################
#   Modelo predicción espacio temporal Houston ( GAM * GLM )    #
#####################################################################

# Guardar la sesión de R
save.image("C:/Users/User/Desktop/TFM/capa.Rdata")
# Cargar la sesión de R
load("C:/Users/User/Desktop/TFM/capa.Rdata")

# Instalamos los paquetes necesarios:
install.packages("rgeos")
install.packages("mgcv")
install.packages("spatstat")
install.packages("splancs")
install.packages("stpp")
install.packages("sp")
install.packages("raster")
install.packages("rgdal")
install.packages("maptools")
install.packages("glm.predict")
install.packages("knitr")

# Cargamos los paquetes necesarios:
library('rgeos')
library('mgcv')
library('spatstat')
library('splancs')
library('stpp')
library('sp')
library('raster')
library('rgdal')
library('maptools')
library('glm.predict')
library('knitr')

#--- Cargarmos el .shp de los ejes ---#
wd2 = "C:/Users/User/Desktop/CapasHoustonDefinitivas"
shpname2 = "ejes_id_unico"
capaEjes = readOGR(dsn = wd2, layer = shpname2)
plot(capaEjes)
head(capaEjes@data)
length(capaEjes)

#--- Centroides ---#
centroides_capa = gCentroid(capaEjes, byid=TRUE)@coords
```

```
cenx=centroides_capa[,1]
ceny=centroides_capa[,2]
plot(capaEjes)
points(cenx,ceny)

#--- Cargarmos el .shp de los hechos ---#
shpname = "crash_all"
capaHechos = readOGR(dsn = wd2, layer = shpname)
length(capaHechos@data)
plot(capaEjes)
points(capaHechos, col="turquoise4")
length(capaHechos@data)

#--- Convertimos la variable difftime y EID ---#
capaHechos@data[,592] = as.numeric(as.character(capaHechos@data[,592]))
capaHechos@data[,593] = as.numeric(as.character(capaHechos@data[,593]))

#--- Función de intensidad espacioTemporal ---#
f=matrix(rep(0,length(capaEjes)*1096), nrow = (max(capaHechos@data[,592])+1),+
+ncol = length(capaEjes))

for ( i in 1:length(capaHechos@data[,1])) {
  f[(capaHechos@data[i,592]+1),capaHechos@data[i,593]] =+
  +f[(capaHechos@data[i,592]+1),capaHechos@data[i,593]]+1
}

#--- Función de Intensidad Empírica Temporal y Espacial ---#
fIntEmpiricaTemp=apply(f,1,sum)
fIntEmpiricaEsp=apply(f,2,sum)
boxplot(fIntEmpiricaTemp)
plot(fIntEmpiricaTemp,type="l")


#########################
#    Modelo Temporal    # ----------------------------------------------
#########################

# Seleccionamos las covariables temporales:
dfTemporal=capaHechos@data[,c(9,11,12,13,16,17,18,592)]
write.csv(dfTemporal, file = "dfTemporal.csv")
dfTemporal=read.csv("C:/Users/User/Desktop/dfTemporal.csv",sep=";")
dfTemporal=dfTemporal[,2:9]
dfTemporal[,7]=dfTemporal[,7]+1
dfTemporal[,8]=dfTemporal[,8]+1

### --- Vectores auxiliares --- ###
Mes = c()
DiaSemana = c()
Hora = c()
Luz = c()
Tiempo = c()
Sup = c()
EstadoCarretera = c()
```

```
diftime2 = c()
### ----------------------------###
i=1
for(k in 1:1096){
  ### ----------------------------###
  nuevaMatriz = dfTemporal[(dfTemporal[,8]==k),]
  ### ----------------------------###
  Mes[i] = nuevaMatriz[1,1]
  ### ----------------------------###
  DiaSemana[i] = nuevaMatriz[1,2]
  ### ----------------------------###
  Hora[i] = round(sum(nuevaMatriz[,3])/length(nuevaMatriz[,3]))
  ### ----------------------------###
  aux = margin.table(t(nuevaMatriz[,4]),2)
  aux2 = unique(aux)
  num = length(aux2)
  def = aux2[1]
  count = sum( aux == 1 )
  j = 1
  for(l in j:num){
    count2 = sum( aux == aux2[l] )
    if (count2>count) {
      count = count2
      def = aux2[l]
    }
  }
  Luz[i] = def
  ### ----------------------------###
  aux = margin.table(t(nuevaMatriz[,5]),2)
  aux2 = unique(aux)
  num = length(aux2)
  def = aux2[1]
  count = sum( aux == 1 )
  j = 1
  for(l in j:num){
    count2 = sum( aux == aux2[l] )
    if (count2>count) {
      count = count2
      def = aux2[l]
    }
  }
  Tiempo[i] = def
  ### ----------------------------###
  aux = margin.table(t(nuevaMatriz[,6]),2)
  aux2 = unique(aux)
  num = length(aux2)
  def = aux2[1]
  count = sum( aux == 1 )
  j = 1
  for(l in j:num){
    count2 = sum( aux == aux2[l] )
    if (count2>count) {
      count = count2
```

```
      def = aux2[l]
    }
  }
  Sup[i] = def
  ### ----------------------------###
  aux = margin.table(t(nuevaMatriz[,7]),2)
  aux2 = unique(aux)
  num = length(aux2)
  def = aux2[1]
  count = sum( aux == 1 )
  j = 1
  for(l in j:num){
    count2 = sum( aux == aux2[l] )
    if (count2>count) {
      count = count2
      def = aux2[l]
    }
  }
  EstadoCarretera[i] = def
  ### ----------------------------###
  diftime2[i] = i
  ### ----------------------------###
  i = i+1
}

### ----------------------------###
auxTemp=NULL

cosine<-cos(2*pi*diftime2/1096)
sine<-sin(2*pi*diftime2/1096)
auxTemp = cbind( cosine, auxTemp )
auxTemp = cbind( sine, auxTemp )
auxTemp = cbind( Mes, auxTemp )
auxTemp = cbind( DiaSemana, auxTemp )
auxTemp = cbind( Hora, auxTemp )
auxTemp = cbind( Luz, auxTemp )
auxTemp = cbind( Tiempo, auxTemp )
auxTemp = cbind( Sup, auxTemp )
auxTemp = cbind( EstadoCarretera, auxTemp )
auxTemp = cbind( diftime2, auxTemp )

### --- Creamos el modelo temporal ----###
modeloTemporal = as.data.frame(auxTemp[,2:10]) # sólo variables independientes
ErrorCuad = numeric(0)
ErrorCuadAcum = ErrorCuad
for(i in names(modeloTemporal)) { # recorremos var. indep.
  reg = glm(formula=paste("fIntEmpiricaTemp~", i),family='poisson')
  ErrorCuad[i] = var(reg$residuals) # calculamos error
}
VarElegida = names(which.min(ErrorCuad)) # var. elegida
VarQuedan = names(modeloTemporal)[ names(modeloTemporal) != VarElegida ] # la quitamos
Forms = character(0) # aquÃ??? guardaremos las fÃ³rmulas
ErrorCuadAcum = c(ErrorCuadAcum, min(ErrorCuad))
```

```
VarQuedan = VarQuedan[ VarQuedan != VarElegida ]
FormAnt = paste("fIntEmpiricaTemp ~ ", VarElegida)
Forms = c(Forms, FormAnt)
while( length(VarQuedan)>0 ) {
  ErrorCuad = numeric(0)
  for( i in VarQuedan ) {
    reg = glm(formula=paste(FormAnt, i, sep="+"),family='poisson')
    # EXTRAER ERROR CUADRATICO DEL MODELO
    ErrorCuad[i] = var(reg$residuals)
  }
  VarElegida = names(which.min(ErrorCuad))
  VarQuedan = VarQuedan[ VarQuedan != VarElegida ]
  FormAnt = paste(FormAnt, VarElegida, sep="+")
  Forms = c(Forms, FormAnt)
  ErrorCuadAcum = c(ErrorCuadAcum, min(ErrorCuad))
}

# Resultados
Forms
ErrorCuadAcum[which.min(ErrorCuadAcum)]
Forms[which.min(ErrorCuadAcum)]

regresionTemp = glm(formula=Forms[which.min(ErrorCuadAcum)],family='poisson')
estimacionTemp = predict(regresionTemp,type="response")
estimacionTemp = round(estimacionTemp)

# Ajuste temporal, gráficamente:
plot(fIntEmpiricaTemp,type="l",xlab="Día",ylab="Función de intensidad temporal")
lines(estimacionTemp,col="turquoise4")

# Bondad del ajuste gráfico
plot(abs(fIntEmpiricaTemp-estimacionTemp),type="l",xlab="Día",+
+ylab="Error función de intensidad temporal", ylim=c(0,350))

# Bondad Ajuste numérico
numMedioAccidentesPorDiaEmpirico=180970/1096
numMedioAccidentesErradosPorDiaModelo=sum(estimacionTemp)/1096
ajusteModeloTemporal=1-(sum((abs(fIntEmpiricaTemp-estimacionTemp)))/180970)
numMedioAccidentesPorDiaEmpirico
numMedioAccidentesErradosPorDiaModelo
ajusteModeloTemporal


#########################
#    Modelo Espacial    # ------------------------------------------------
#########################

newCapaHechos=NULL
newCapaHechos=capaHechos@data[,c(7,8,15,19:27,41,46,47,49:52,54:56,58:65,75:79,81:83,+
+85:87,89:92,267:303,591:593)]
write.csv(newCapaHechos, file = "newCapaHechos.csv")
newCapaHechos=read.csv("C:/Users/User/Desktop/cHechos.csv", sep=";")
dim(newCapaHechos)
```

```
ejesConAccidentes=length(sort(unique(newCapaHechos[,63])))
ejesAccidentes=sort(unique(newCapaHechos[,63]))
todosLosEjes=length(fIntEmpiricaEsp)
ejesConAccidentes
todosLosEjes

auxEspacial = matrix(0, ejesConAccidentes, 63)
auxEspacial = data.frame(auxEspacial)

i=1
for( k in sort(unique(newCapaHechos[,63])) ) {
  nuevaMatriz = newCapaHechos[(newCapaHechos[,63]==k),]
  q = 1
  for (t in 1:63){
    aux = nuevaMatriz[,t]
    aux2 = unique(aux)
    num = length(aux2)
    def = aux2[1]
    count = sum( aux == aux[1] )
    j = 1
    for(l in j:num) {
      count2 = sum( aux == aux2[l] )
      if (count2>count) {
        count = count2
        def = aux2[l]
      }
    }
    auxEspacial[i,q]=def
    q = q + 1
  }
  i = i+1
}
names(auxEspacial)=c("POPGROUP",    "ROADCLAS", "SEVERITY", "INVESTIG", "ALIGNMNT",+
+"TRAFCNTL",    "ONOFFRD",  "INTERSEC", "INTRROAD", "ENTRROAD", "VEHMOVE",  "OBJECT",+
+"NUMVEH",  "X",    "Y",    "DEFECT1",  "RACESEX1", "STATUS1",  "INSURNC1", "DISABIL1",+
+"FACTOR1A",    "FACTOR1B", "VEHNUM1",  "SEVERE1",  "BELT1",    "CASUALT1", "TRUCK",+
+"COMMVEH", "PICKUP",   "PANELVAN", "SUV",  "DUI",  "REDLIGHT", "P_REDLIGHT",   +
+"SPEEDING",    "FAILSTOP", "NOYIELD",  "BADTURN",  "TOOCLOSE", "FROMPARK", "NOSIGNAL",+
+"BADPASS", "EMS",  "TEENAGE",  "TWENTIES", "THIRTIES", "FORTIES",  "FIFTIES",+
+"ELDERLY", "CHILD",    "MALE", "FEMALE",   "WHITE",    "BLACK",    "FATAL",    "TYPEA",+
+"TYPEB",   "TYPEC",    "PDO",  "BUS",  "CELLPHON", "difftime", "EID")

fIntEmpiricaEspSinCeros = c()
cenxSinCeros = c()
cenySinCeros = c()
i=1
for (s in 1:length(fIntEmpiricaEsp)) {
  if(fIntEmpiricaEsp[s]!=0){
    fIntEmpiricaEspSinCeros[i]=fIntEmpiricaEsp[s]
    cenxSinCeros[i]=cenx[s]
    cenySinCeros[i]=ceny[s]
    i = i + 1
  }
```

```
}
auxEspacial = cbind(auxEspacial,cenxSinCeros)
auxEspacial = cbind(auxEspacial,cenySinCeros)
auxEspacial = cbind(auxEspacial,fIntEmpiricaEspSinCeros)
attach(auxEspacial)
head(auxEspacial)
dim(auxEspacial)

modeloEspacial = as.data.frame(auxEspacial[,1:61]) # sólo variables independientes
ErrorCuadratico = numeric(0)
ErrorCuadraticoAcum = ErrorCuadratico


for(i in names(modeloEspacial)) { # recorremos var. indep.
  gamfunction = paste("reg=gam(fIntEmpiricaEspSinCeros~",i,"+s(cenxSinCeros,cenySinCeros),+
  +family='poisson')",sep="")
  eval(parse(text=gamfunction))
  ErrorCuadratico[i] = var(reg$residuals) # calculamos error
}
VarElegida = names(which.min(ErrorCuadratico)) # var. elegida
VarQuedan = names(modeloEspacial)[ names(modeloEspacial) != VarElegida ] # la quitamos
Formulas = character(0) # aquÃ??? guardaremos las fÃ³rmulas
ErrorCuadraticoAcum = c(ErrorCuadraticoAcum, min(ErrorCuadratico))

VarQuedan = VarQuedan[ VarQuedan != VarElegida ]
FormulaAnt = paste(VarElegida)
Formulas = c(Formulas, FormulaAnt)
while( length(VarQuedan)>0 ) {
  ErrorCuadratico = numeric(0)
  for( i in VarQuedan ) {
    newFormula = paste(FormulaAnt,i,sep="+")
    gamfunction = paste("reg=gam(fIntEmpiricaEspSinCeros~",newFormula,+
    +"+s(cenxSinCeros,cenySinCeros),family='poisson')",sep="")
    eval(parse(text=gamfunction))
    ErrorCuadratico[i] = var(reg$residuals) # calculamos error
  }
  VarElegida = names(which.min(ErrorCuadratico))
  VarQuedan = VarQuedan[ VarQuedan != VarElegida ]
  FormulaAnt = paste(FormulaAnt, VarElegida, sep="+")
  Formulas = c(Formulas, FormulaAnt)
  ErrorCuadraticoAcum = c(ErrorCuadraticoAcum, min(ErrorCuadratico))
  print(FormulaAnt)
}

# Resultados
Formulas
ErrorCuadraticoAcum[which.min(ErrorCuadraticoAcum)]
Formulas[which.min(ErrorCuadraticoAcum)]

gamfunction = paste("regresionEsp=gam(fIntEmpiricaEspSinCeros~",+
+Formulas[which.min(ErrorCuadraticoAcum)],"+s(cenxSinCeros,cenySinCeros),+
+family='poisson')",sep="")
eval(parse(text=gamfunction))
```

```r
var(regresionEsp$residuals)
resultado = round(predict(regresionEsp,type="response"))

estimacionEspacial = c()
for (i in 1:ejesConAccidentes){
  estimacionEspacial=c(estimacionEspacial,resultado[[i]])
}
estimacionEspacial
comparacion = cbind(estimacionEspacial,fIntEmpiricaEspSinCeros)
comparacion

# Bondad de ajuste espacial
1-sum(abs(comparacion[,1]-comparacion[,2]))/180970


# Creamos los archivos .shp para representar gráficamente el modelo Temporal ---
resultadosEstimados=as.data.frame(estimacionEsp.)
resultadosEmpíricos=as.data.frame(fMatIntEmpiricaEsp)
capaEjes@data=cbind(capaEjes@data,resultadosEmpíricos)
capaEjes@data=cbind(capaEjes@data,resultadosEstimados)
names(capaEjes@data)=c("EID","resultadosEmpíricos","resultadosEstimados")
head(capaEjes@data)
writeOGR(capaEjes, "C:/Users/User/Desktop/TFM" ,"resEspacial", driver="ESRI Shapefile")
b = readOGR(dsn = "C:/Users/User/Desktop/TFM", layer = "resEspacial")
# ----------------------------------------------------------------


### --- MATRICES FINALES --- ###
fMatIntEmpiricaTemp=as.matrix(fIntEmpiricaTemp)
fMatIntEmpiricaEsp=as.matrix(fIntEmpiricaEsp)
dim(fMatIntEmpiricaTemp)
dim(fMatIntEmpiricaEsp)

funcIntEspacioTemporalEmpirica=fMatIntEmpiricaTemp%*%t(fMatIntEmpiricaEsp)
dim(funcIntEspacioTemporalEmpirica)
sum(funcIntEspacioTemporalEmpirica)


j=1
ejesAccidentes
fIntEspacialEstimadaCompleta = c()
for(i in 1:4145){
  if (i==ejesAccidentes[j]){
    fIntEspacialEstimadaCompleta = c(fIntEspacialEstimadaCompleta, estimacionEspacial[j])
    j=j+1
  }
  else{
    fIntEspacialEstimadaCompleta = c(fIntEspacialEstimadaCompleta, 0)
  }
}
# Añadimos los últimos
fIntEspacialEstimadaCompleta=c(fIntEspacialEstimadaCompleta,estimacionEspacial[2836:2843])
```

```
estimacionTemp=as.matrix(estimacionTemp)
estimacionEsp.=as.matrix(fIntEspacialEstimadaCompleta)
dim(estimacionTemp)
dim(estimacionEsp.)

funcIntEspacioTemporalEstimada = estimacionTemp%*%t(estimacionEsp.)

# Bondad del ajuste del modelo espaciop-temporal:
1-sum(abs(funcIntEspacioTemporalEmpirica-funcIntEspacioTemporalEstimada))/180970^{2}

# Calculamos el error Z del modelo:
estEspSinCeros = as.matrix(resultado)
empEspSinCeros = as.matrix(fIntEmpiricaEspSinCeros)

funcIntEspacioTemporalEstimadaSinCeros =  estimacionTemp%*%t(estEspSinCeros)
funcIntEspacioTemporalEsmSinCeros = fMatIntEmpiricaTemp%*%t(empEspSinCeros)

A=log(funcIntEspacioTemporalEsmSinCeros)-log(funcIntEspacioTemporalEstimadaSinCeros)
Z=exp(abs(A))

errorEspacialSinCeros = apply(Z,2,sum)/180970
errorTemporal = apply(Z,1,sum)/180970

j=1
errorEspacial = c()
for(i in 1:4145){
  if (i==ejesAccidentes[j]){
    errorEspacial = c(errorEspacial, errorEspacialSinCeros[j])
    j=j+1
  }
  else{
    errorEspacial = c(errorEspacial, 0)
  }
}
# Añadimos los últimos
errorEspacial=c(errorEspacial,errorEspacialSinCeros[2836:2843])

# Creamos el archivo .shp para representar gráficamente el error espacial del modelo ---
errorEspacial=as.data.frame(errorEspacial)
capaEjes@data=cbind(capaEjes@data,errorEspacial)
names(capaEjes@data)=c("EID","resultadosEmpíricos","resultadosEstimados", "errorEspacial")
head(capaEjes@data)
writeOGR(capaEjes, "C:/Users/User/Desktop/TFM" ,"errorEspacial", driver="ESRI Shapefile")
# b = readOGR(dsn = "C:/Users/User/Desktop/TFM", layer = "errorEspacial")


### --- Predecimos un nuevo valor temporal --- ###
diaPredecir=1096
modeloTemporalPred = modeloTemporal[-diaPredecir,]
nuevaObservacion=modeloTemporal[diaPredecir,]

regresionPred = glm(formula=fIntEmpiricaTemp[1:1095]~modeloTemporalPred[,2]+
+modeloTemporalPred[,6]+modeloTemporalPred[,8]+modeloTemporalPred[,4]+modeloTemporalPred[,7]+
```

```
+modeloTemporalPred[,9]+modeloTemporalPred[,5]+modeloTemporalPred[,1]+modeloTemporalPred[,3],+
+family='poisson')
estimacionPuntual = predict.glm(regresionPred, nuevaObservacion, type="response")

estimacionPuntual[1096]
fIntEmpiricaTemp[1096]
```