Predicción de homicidios a patir de estructuras de grafos y características secuenciales de procesamiento de señales

Juan Moreno, Sebastian Quintero, Cristian Sánchez, Álvaro Riascos, Luis G. Nonato

Quantil - Matemáticas Aplicadas

5 de junio de 2020

quantil

Contenido

Análisis de Crimen

Un reto importante dentro del análisis de patrones en datos de crimen es la predicción por tipo de crimen. Homicidios cobra una particular importancia en la prioridades de la policía metropolitana de Bogotá. Los datos tienen un subreporte casi inexistente, y una calidad excepcional debido a cotejos semanales entre la SSCJ, Medicina Legal, Policía y Fiscalía para refinar los datos.

Análisis de Crimen

- Los homicidios ocurren de manera muy esporádica en comparación con otros tipos de delitos, como hurtos y riñas.
- El análisis estadístico y los modelos de predicción son particularmente complejos de implementar cuando se enfrentan datos de baja frecuencia.
- Hay alrededor de mil homicidios al año, relativo a hurtos donde ocurren cerca de 40 mil.

Datos de Homicidios

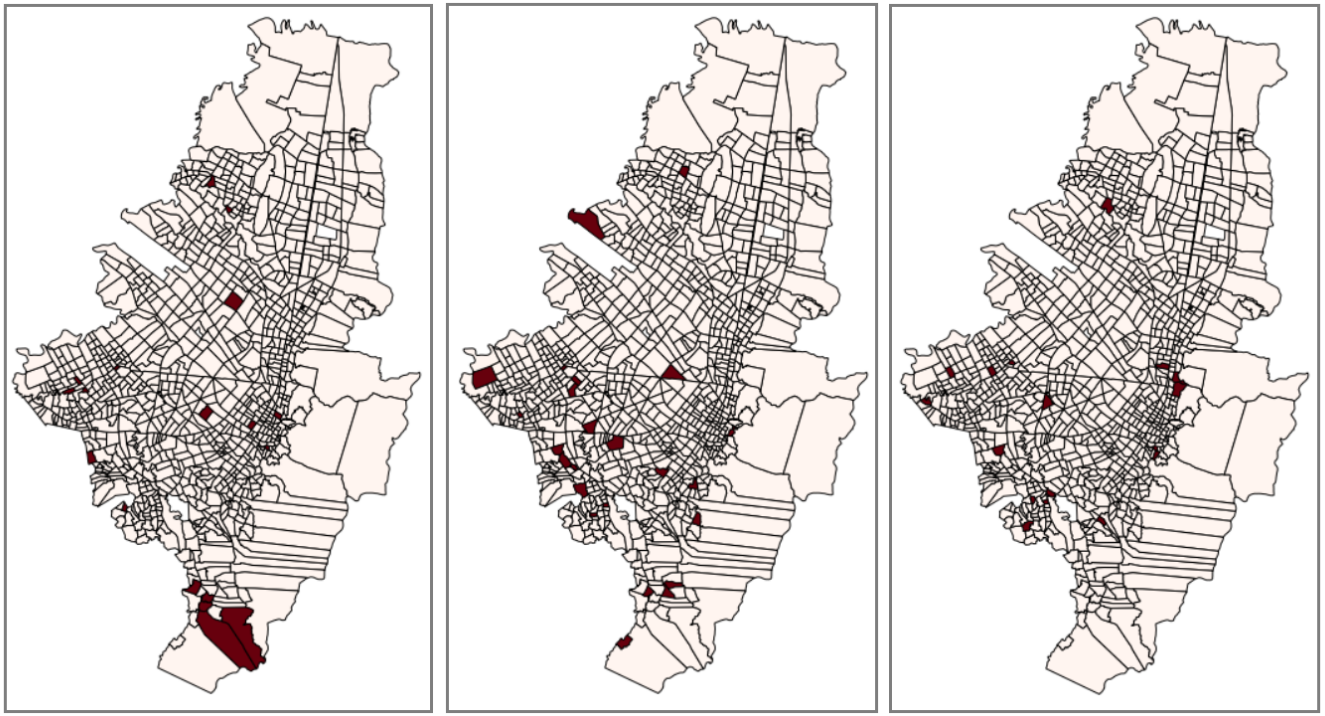


Figura: Cuadrantes en los que ocurrieron homicidios durante las semanas 12, 14, y 16 del año 2018

Acercamiento

- Esta trabajo predice homicidios a partir de:
  - La estructura de la representación espacial de la ciudad como un grafo.
  - La teoría del procesamiento de señales en grafos.
- La investigación presentada por Nonato et al (2020) propone la metodología del filtrado laplaciano de Gauss para señales (*GLoG*).
- *GloG* permite la definición de un filtro para la detección de límites en el dominio de gráfos basado en el desplazamiento de las señales.
- *DeepWalk* se usa para obtener la representación de un nodo en menor dimensionalidad, capturando su rol en la red, como su comunidad, centralidad, y la posibilidad de evaluar similitud entre nodos

Acercamiento

- Para representar Bogotá como un grafo se usan los cuadrantes de la policía.
- Las aristas representan si los cuadrante comparten límites.
- Hay 1051 cuadrantes en Bogotá.

Figura: Cuadrantes Bogotá

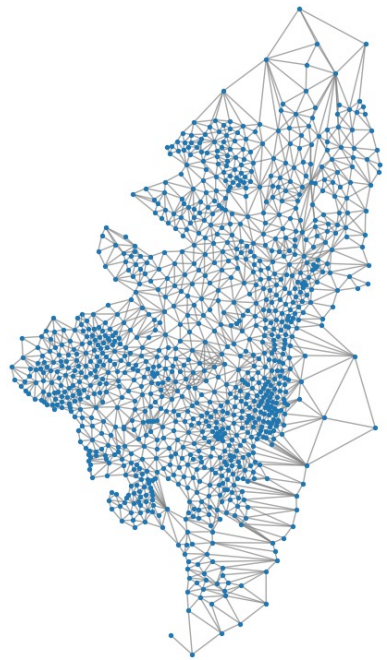


Figura: Representación de Bogotá en grafo

Contenido

Principio de *GLoG*

- El operador GloG busca detectar **nodos borde** en cada segmento de tiempo
- Los límites tienen lugar en lugares donde el Laplaciano de la señal es cero. Se denominan nodos limite zero-crossing

Procesamiento de Señales en Grafos

Un grafo puede definirse como $G = (V, E, w)$, donde:

- V es el conjunto de nodos $V = \tau_1, \tau_2, ..., \tau_n$,
- E es el conjunto de enlaces $E = (\tau_i, \tau_j), \tau_i, \tau_j \in V, i \neq j$
- w es una función peso $w : E \rightarrow R$ la cual asocia un escalar a cada enlace en G.

Procesamiento de Señales en Grafos

- El grafo Laplaciano es una matriz simétrica, la cual nos asegura un conjunto de vectores propios u_l, con los correspondientes valores propios $\lambda_l, l = 1, 2, ..., n$.
- En este contexto, los valores propios representan las frecuencias.
- The Graph Fourier Transform (GFT) de una señal f, denotada por $\hat{f} : \Lambda \rightarrow R$, donde Λ es el dominio espectral (valores propios), se define como:

$$\hat{f}(\lambda_l) = \sum_{j=1}^{n} u_l(\tau_j) f(\tau_j) \tag{1}$$

- Donde f es la señal definida en los nodos de G.

Procesamiento de Señales en Grafos

Denotando U como la matriz (ortogonal) cuyas columnas están dadas por los vectores propios u_l, GFT y iGFT se pueden obtener a través de una multiplicación:

$$GFT = U^T f \tag{2}$$

$$iGFT = U\hat{f} \tag{3}$$

Detección de Límites

Los límites de una señal corresponden a los puntos donde se presenta un cambio relevante.

- El Laplaciano de Gauss (LoG) figura entre los enfoques más importantes para identificar límites o bordes
- Funciona como un filtro, el cual identifica cambios abruptos donde el Laplaciano de la señal es cero (donde el gradiente de la señal es máximo)

Procesamiento de Señales en Grafos

El filtro LoG clásico se puede definir como:

$$LoG(f) = \nabla^2 G * f \tag{4}$$

Donde ∇^2 es el operador Laplaciano, G es la función Gausiana, f es la señal, y $*$ es el operador de convolución.

Procesamiento de Señales en Grafos

En la teoría procesamiento de señales en grafos, la convolución entre dos funciones f y g se puede definir como:

$$f * g = iGFT(\hat{f} \cdot \hat{g}) \tag{5}$$

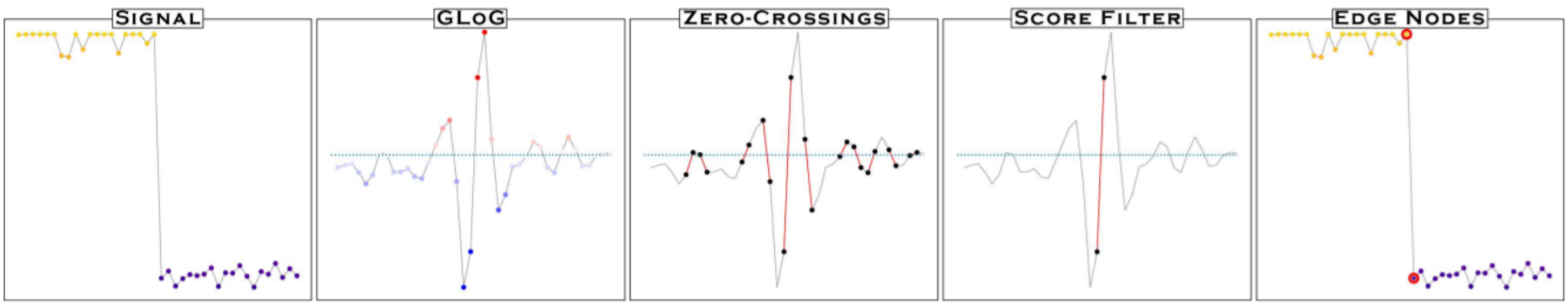
Donde $\cdot$ es la multiplicación elemental, $\hat{f}$ y $\hat{g}$ son los GFT de f y g

Procesamiento de Señales en Grafos

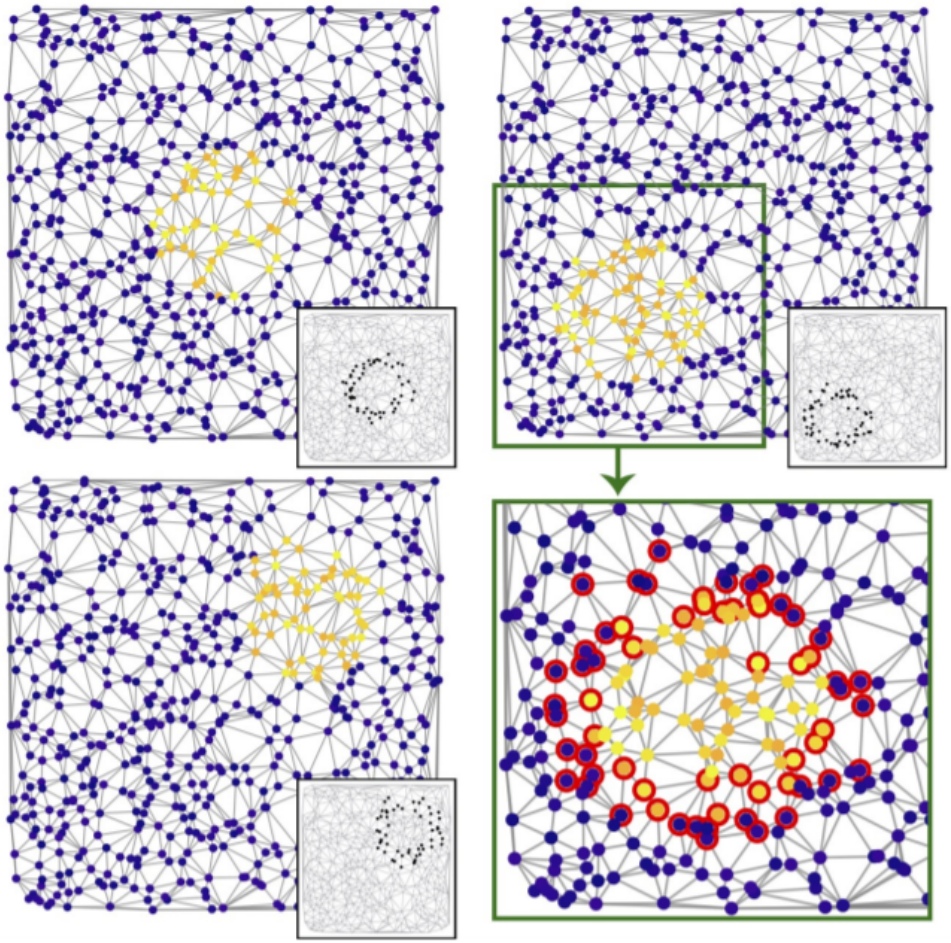
El filtro Graph Laplacian of Gaussian (GLoG) puede entonces definirse como:

$$GLoG(f) = iGFT(\nabla^2 G \cdot \hat{f}) \tag{6}$$

Principio de GloG



Procesamiento de Señales en Grafos



Datos de Homicidios

- Utilizamos registros de homicidios de 2013 a 2019.
- Los registros están georeferenciados, por lo que construimos un grafo donde los nodos corresponden al cuadrante donde ocurrió el homicidio.
- Los enlaces entre los nodos indican si los cuadrantes se cruzan geográficamente entre sí.
- La cantidad total de nodos fue de 1051.
- Aplicamos agregación semanal a los datos, lo que resulta en 366 segmentos de tiempo.

Resultados de GLOG

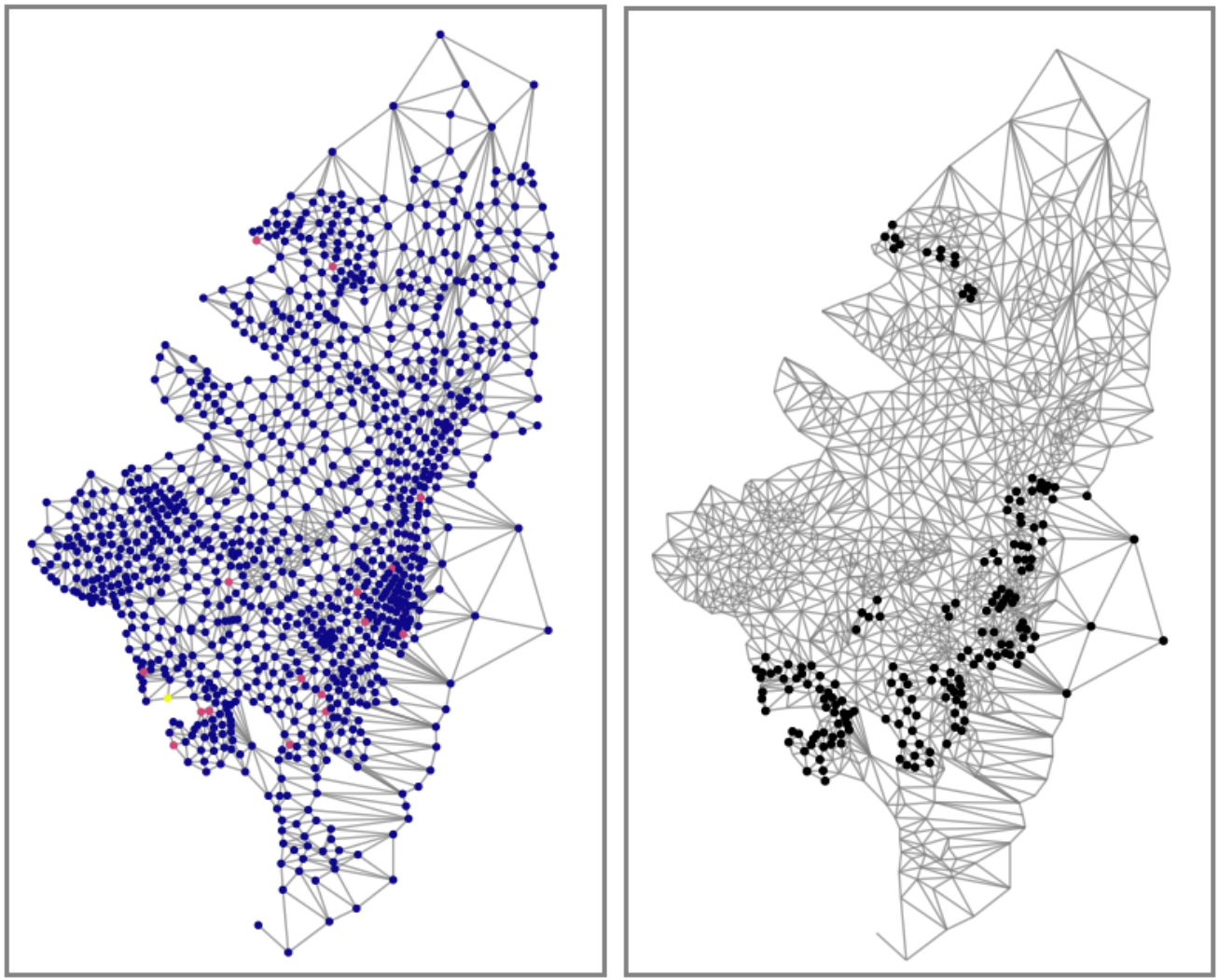
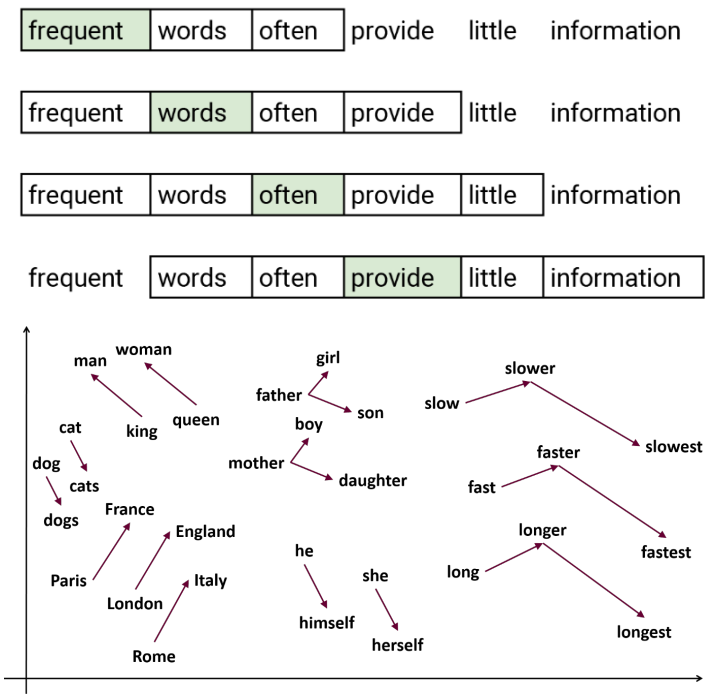


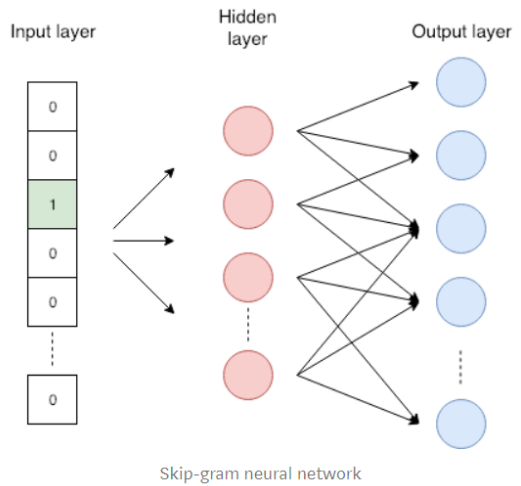
Figura: Número de homicidios en Bogotá de 2013 a 2019 (izquierda). Nodos borde (derecha)

Contenido

DeepWalk

- Esta inspirada en Word2Vec, metodología utilizada para representar en una menor dimensionalidad las palabras, capturando el sentido de estas a través de su contexto.

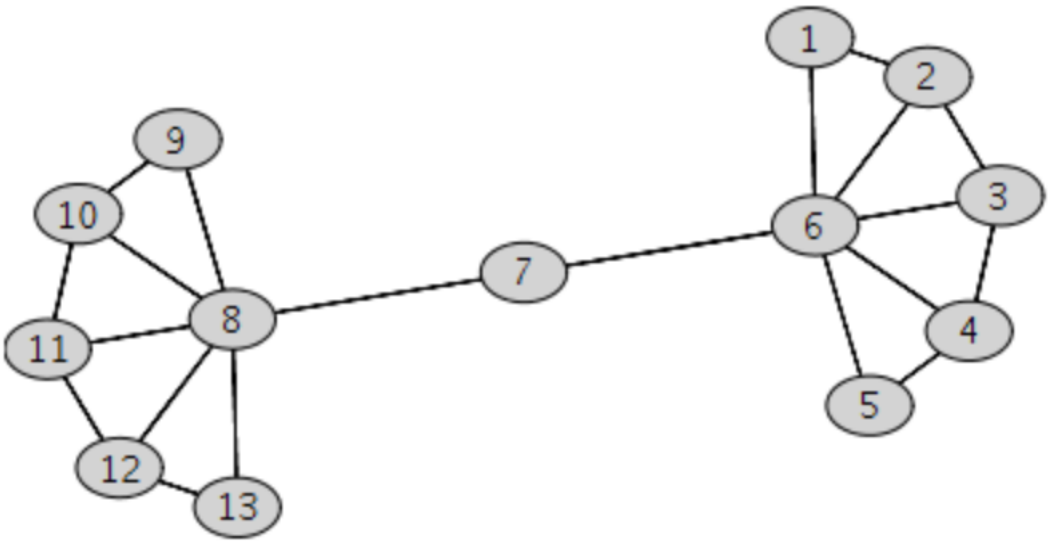




Skip-gram neural network

DeepWalk

- Debemos construir el análogo al "contexto" para cada nodo.
- Para cada nodo en el grafo se simularan caminatas aleatorias.



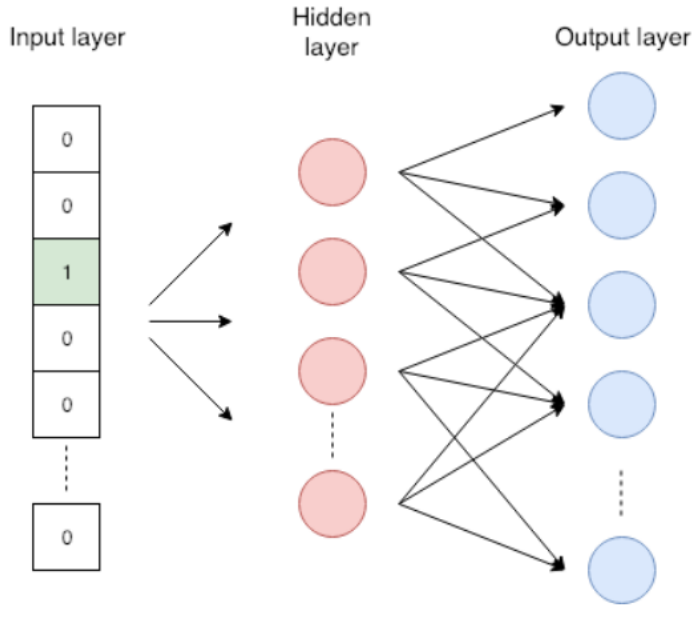
DeepWalk

1	2	3	4	5	4	3	6	4	3	6	4	5	6	1	6	7	6	4	3
2	1	6	3	6	2	1	2	6	7	8	10	9	10	9	10	11	12	8	13
3	4	6	2	3	6	3	2	3	2	1	6	7	8	10	8	13	12	8	11
4	3	6	3	4	6	2	3	2	3	4	6	2	1	2	3	6	1	6	1
5	6	5	4	3	4	3	2	3	4	6	4	6	7	8	13	12	11	8	9
6	1	2	3	6	1	6	1	6	4	5	4	3	4	5	4	6	7	8	12
7	8	10	8	11	8	7	6	5	4	6	5	6	5	6	1	2	6	7	8
8	9	10	9	10	11	12	11	12	11	12	13	12	8	11	10	9	10	8	9
9	8	12	13	12	11	10	11	8	9	10	8	12	11	10	11	8	13	8	7
10	8	13	8	11	12	13	12	11	12	13	8	7	6	1	6	1	6	3	6
11	8	10	8	13	12	13	12	8	10	11	12	11	10	8	10	8	12	13	12
12	8	10	9	8	10	8	13	8	13	8	7	6	7	8	13	12	11	10	8
13	8	7	6	5	4	5	6	1	2	1	6	4	3	2	3	2	6	3	6

1	2	3	4	5	4	3	6	4	3	6	4	5	6	1	6	7	6	4	3
2	1	6	3	6	2	1	2	6	7	8	10	9	10	9	10	11	12	8	13
3	4	6	2	3	6	3	2	3	2	1	6	7	8	10	8	13	12	8	11
4	3	6	3	4	6	2	3	2	3	4	6	2	1	2	3	6	1	6	1
5	6	5	4	3	4	3	2	3	4	6	4	6	7	8	13	12	11	8	9
6	1	2	3	6	1	6	1	6	4	5	4	3	4	5	4	6	7	8	12
7	8	10	8	11	8	7	6	5	4	6	5	6	5	6	1	2	6	7	8
8	9	10	9	10	11	12	11	12	11	12	13	12	8	11	10	9	10	8	9
9	8	12	13	12	11	10	11	8	9	10	8	12	11	10	11	8	13	8	7
10	8	13	8	11	12	13	12	11	12	13	8	7	6	1	6	1	6	3	6
11	8	10	8	13	12	13	12	8	10	11	12	11	10	8	10	8	12	13	12
12	8	10	9	8	10	8	13	8	13	8	7	6	7	8	13	12	11	10	8
13	8	7	6	5	4	5	6	1	2	1	6	4	3	2	3	2	6	3	6

DeepWalk

1	2	3	4	5	4	3	6	4	3	6	4	5	6	1	6	7	6	4	3
2	1	6	3	6	2	1	2	6	7	8	10	9	10	9	10	11	12	8	13
3	4	6	2	3	6	3	2	3	2	1	6	7	8	10	8	13	12	8	11
4	3	6	3	4	6	2	3	2	3	4	6	2	1	2	3	6	1	6	1
5	6	5	4	3	4	3	2	3	4	6	4	6	7	8	13	12	11	8	9
6	1	2	3	6	1	6	1	6	4	5	4	3	4	5	4	6	7	8	12
7	8	10	8	11	8	7	6	5	4	6	5	6	5	6	1	2	6	7	8
8	9	10	9	10	11	12	11	12	11	12	13	12	8	11	10	9	10	8	9
9	8	12	13	12	11	10	11	8	9	10	8	12	11	10	11	8	13	8	7
10	8	13	8	11	12	13	12	11	12	13	8	7	6	1	6	1	6	3	6
11	8	10	8	13	12	13	12	8	10	11	12	11	10	8	10	8	12	13	12
12	8	10	9	8	10	8	13	8	13	8	7	6	7	8	13	12	11	10	8
13	8	7	6	5	4	5	6	1	2	1	6	4	3	2	3	2	6	3	6



Skip-gram neural network

DeepWalk

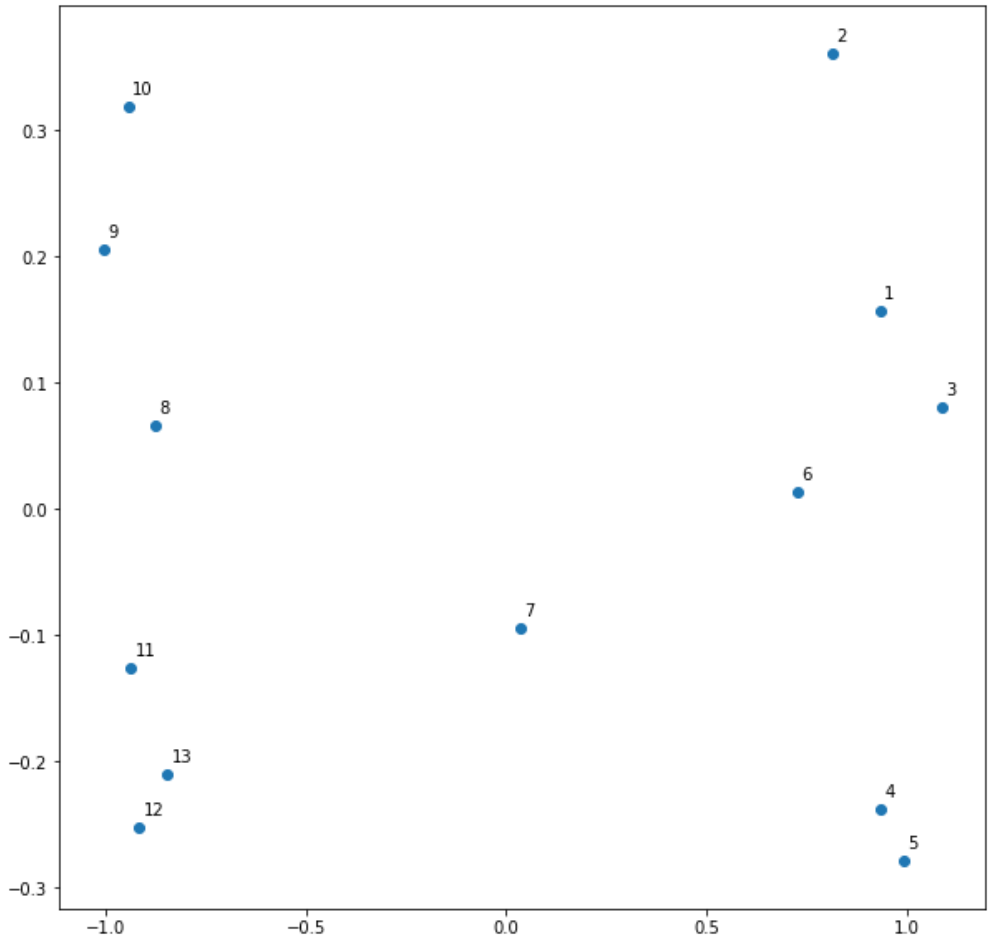


Figura: Representación T-SNE de la capa escondida

DeepWalk

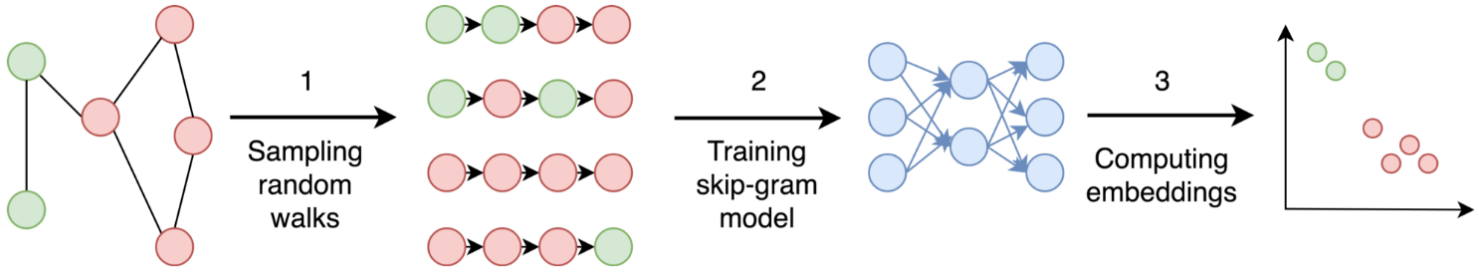


Figura: Resumen

Se escogen los hiperparámetros:

- Dimensión de la representación (cantidad de neuronas en la capa escondida).
- El largo de las caminatas.
- La cantidad de caminatas por nodo.
- Ancho de la ventana considerado en el contexto.

Contenido

El modelo y la base de entrenamiento

- El modelo escogido para realizar las predicciones fue una regresión logística.
- El conjunto de entrenamiento, que corresponde al 70 % de la base, contiene datos entre 2013 y 2018.
- Dado que el numero de segmentos de tiempo es tan bajo, tuvimos la necesidad de designar los datos de cada nodo como una observación distinta.
- Se tienen en total 233.322 observaciones (1051 nodos por 222 semanas).

Los features dinámicos

Para cada segmento de tiempo se tienen los nodos que son borde. Con esta información se crean los features dinámicos, es decir, aquellos que dependen del tiempo. Estos son:

- El indicador de si un nodo es borde.
- El numero de vecinos que también son borde.
- La fracción de vecinos que son borde.
- La probabilidad de ser borde.
- El minimo, el máximo y la media de la probabilidad de ser borde entre los vecinos.
- El numero de homicidios que ocurrieron previamente.

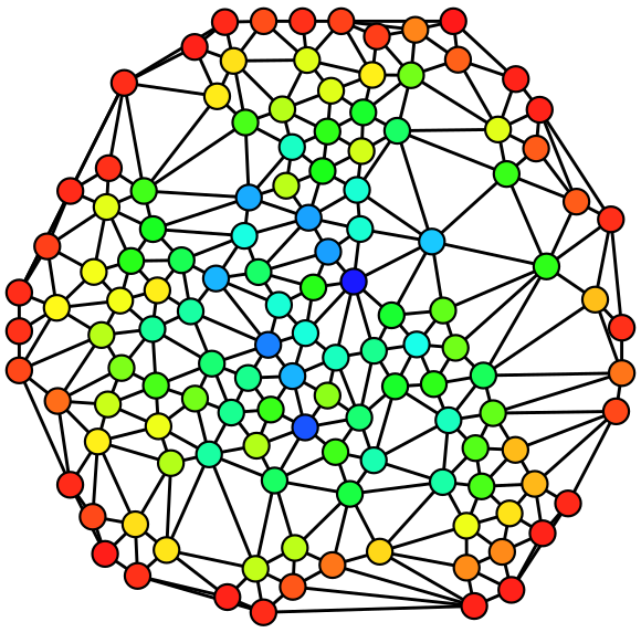
Tanto para la probabilidad como para el numero de homicidios se calcularon usando los datos de los 24 segmentos de tiempo previos.

Los features estáticos

Los features estáticos caracterizan la estructura del grafo, la cual es constante. Los tres mas importantes fueron los siguientes:

Betweenness centrality:

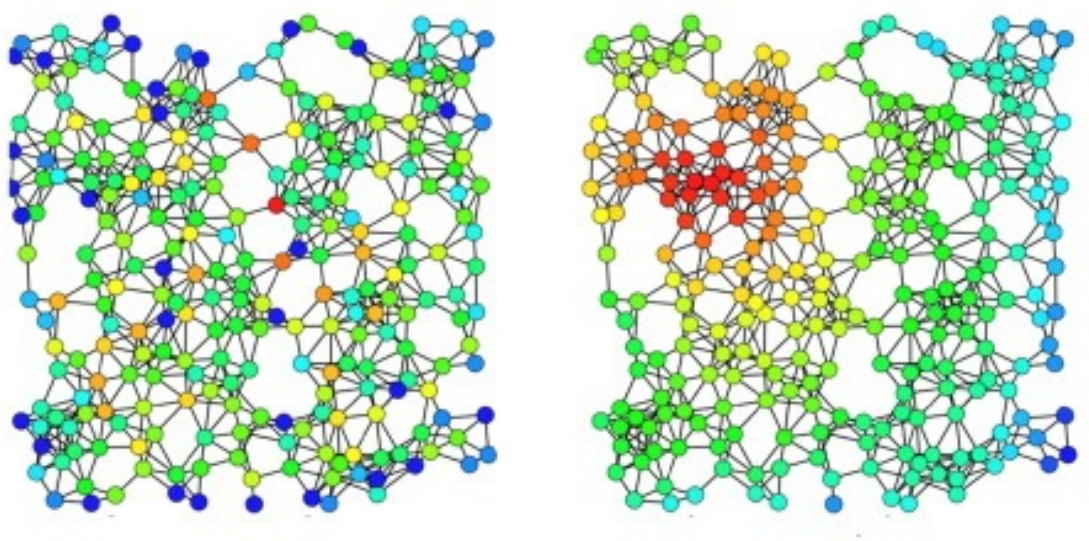
$$C_B(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma st} \tag{7}$$



Los features dinámicos

Eigenvector centrality:

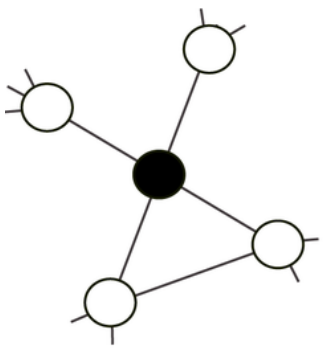
$$x_v = \frac{1}{\lambda} \sum_{t \in G} a_{vt} x_t \tag{8}$$



Los features dinámicos

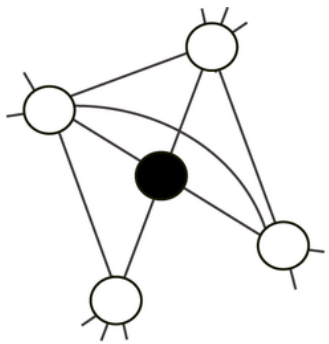
Local clustering coefficient:

$$C_i = \frac{2|e_{jk} : v_j, v_k \in N_i|}{k_i(k_i - 1)} \tag{9}$$



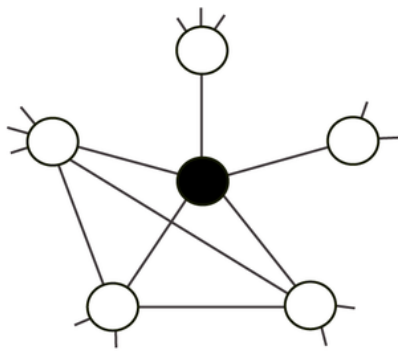
$k_i = 4$

$C_i = \frac{1}{6}$



$k_i = 4$

$C_i = \frac{2}{3}$

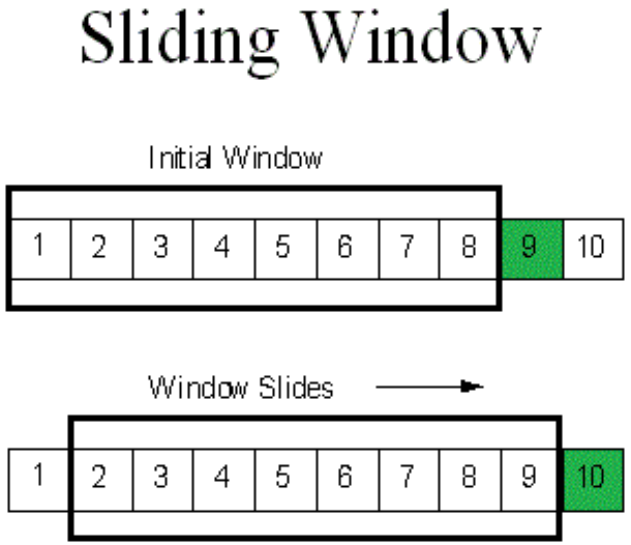


$k_i = 5$

$C_i = \frac{3}{10}$

Los features

Para identificar los diferentes nodos dentro del modelo, se crearon variables dummies que indiquen a que nodo pertenece cada observación.
Finalmente, con los features dinamicos , se crearon secuencias con ventanas móviles de 25 segmentos de tiempo para predecir el siguiente.



Métricas

- La naturaleza del modelo permite usar la curva ROC. Aterrizando el modelo al uso en la realidad ¿Es una buena métrica?

Métricas

- La naturaleza del modelo permite usar la curva ROC. Aterrizando el modelo al uso en la realidad ¿Es una buena métrica?
- Se usa Hit Rate, el cual parte de de hotspots y la cantidad de crimen que se captura en los señalados como Hotspots

$$HR = \frac{\text{homicidios dentro de los hostpots}}{\text{Total de homicidios}}$$

Métricas

- La naturaleza del modelo permite usar la curva ROC. Aterrizando el modelo al uso en la realidad ¿Es una buena métrica?
- Se usa Hit Rate, el cual parte de de hotspots y la cantidad de crimen que se captura en los señalados como Hotspots

$$HR = \frac{\text{homicidios dentro de los hostpots}}{\text{Total de homicidios}}$$

- **Trampa:** si digo que toda la ciudad es un hotspots el HR sería 1.

Métricas

- La naturaleza del modelo permite usar la curva ROC. Aterrizando el modelo al uso en la realidad ¿Es una buena métrica?
- Se usa Hit Rate, el cual parte de de hotspots y la cantidad de crimen que se captura en los señalados como Hotspots

$$HR = \frac{\text{homicidios dentro de los hostpots}}{\text{Total de homicidios}}$$

- **Trampa:** si digo que toda la ciudad es un hotspots el HR sería 1.
- Definimos el PAC (percentage of area covered)

$$PAC = \frac{\text{Área determinada como hotspot}}{\text{Área total}}$$

Métricas

- La naturaleza del modelo permite usar la curva ROC. Aterrizando el modelo al uso en la realidad ¿Es una buena métrica?
- Se usa Hit Rate, el cual parte de de hotspots y la cantidad de crimen que se captura en los señalados como Hotspots

$$HR = \frac{\text{homicidios dentro de los hostpots}}{\text{Total de homicidios}}$$

- **Trampa:** si digo que toda la ciudad es un hotspots el HR sería 1.
- Definimos el PAC (percentage of area covered)

$$PAC = \frac{\text{Área determinada como hotspot}}{\text{Área total}}$$

- Se puede interpreta similar al área bajo la curva ROC.

Contenido

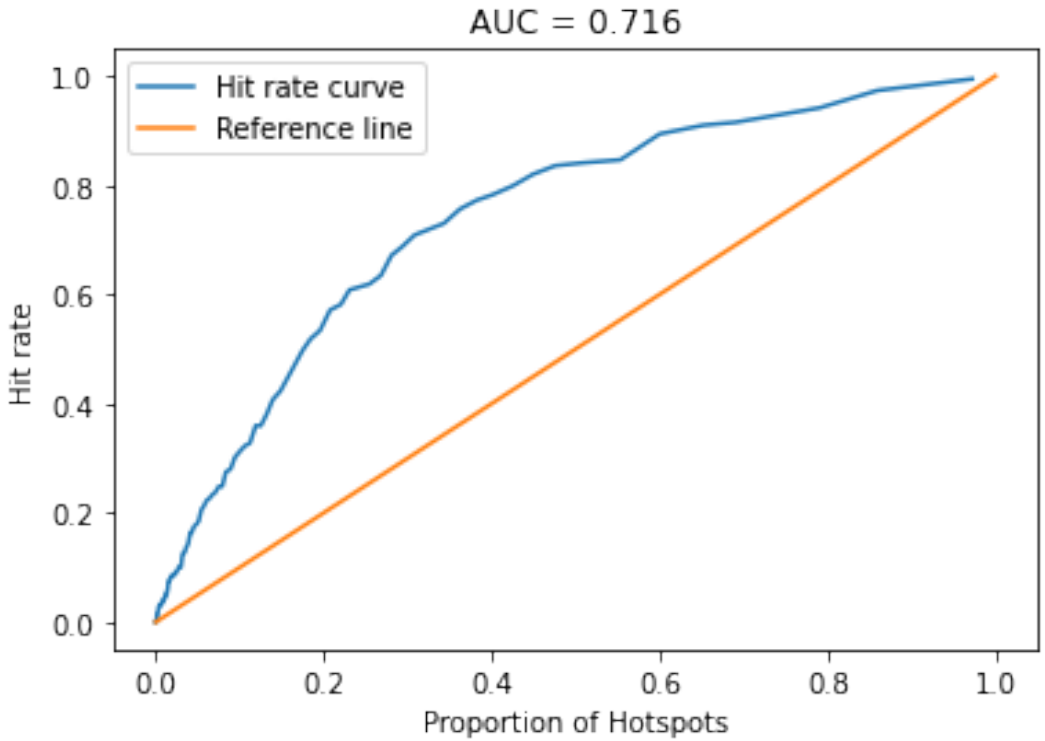
Resultados

En la siguiente tabla se presentan las métricas obtenidas con los diferentes modelos que se probaron.

Modelo	ROC AUC	Nodos para 10 % Hit Rate	Hit rate con 10 % cobertura
Logit solo homicidios	0.612	39	26.9 %
Logit con GLoG homicidios	0.714	34	31.7 %
Gradient Boosting con GLoG homicidios	0.707	40	33.3 %
Logit con GLoG homicidios y riñas	0.712	32	33.3 %
Logit con GLoG riñas	0.590	69	15.0 %
Logit con GLoG homicidios y dummies	0.729	28	30.1 %
Logit con GLoG homicidios, dummies y DeepWalk	0.732	28	31.7 %

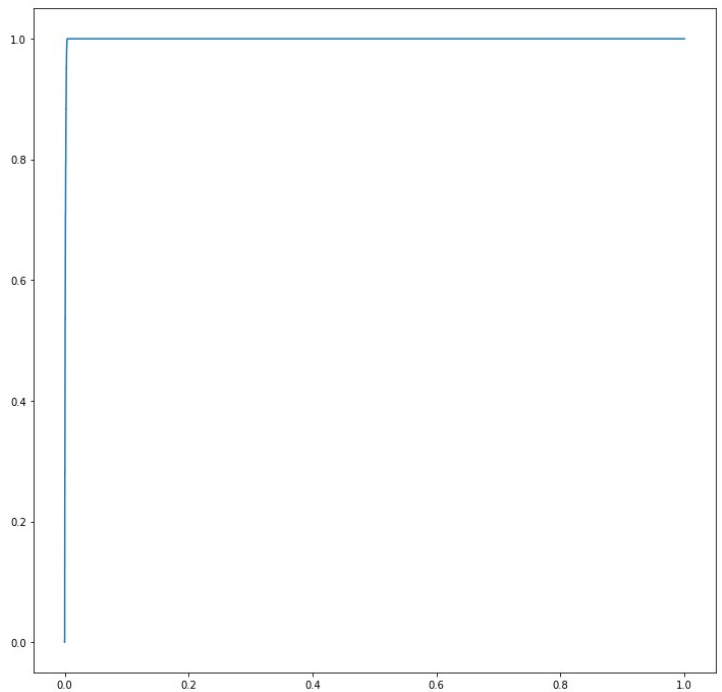
Resultados

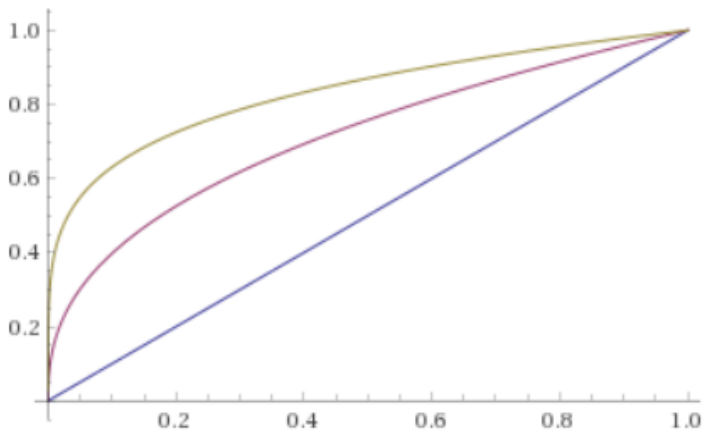
La siguiente gráfica muestra la curva de Hit Rate para el modelo con el área bajo la curva ROC mas alta, es decir, la regresión logística con GLoG de homicidios y DeepWalk.



Discusión sobre las métricas

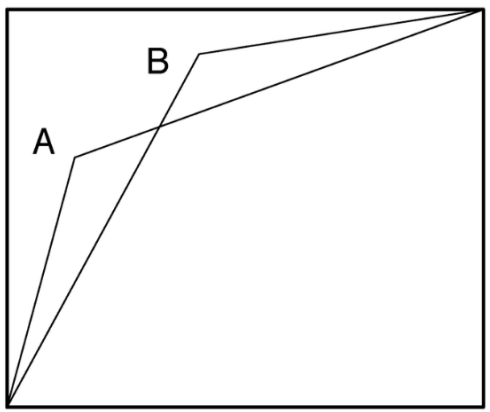
- Tal vez la interpretación de la curva ROC no es la más óptima para la curva HR.
- Suponga que en la realidad el 30 % del crimen se concentra en el 10 % de la ciudad.
- Luego el área bajo la curva HR nunca podrá ser 1.
- Debemos construir la curva teórica a partir de los datos reales.





Discusión sobre las métricas

- Otro problema es que en la práctica no importa el área bajo la curva completa!
- Lo que importa es lo que la policía pueda suplir.
- Para distintos modelos la concavidad de la curva HR es distinta.
- Entonces el área bajo la curva HR que importa no es toda, si no hasta donde la policía pueda suplir.
- E.g. modelo A tiene mejor AUC HR que modelo B, pero modelo B tiene mejor AUC HR hasta el uso del 10 % de la ciudad.



Referencias Principales

- Nonato, Luis Gustavo, Fabiano Petronetto Carmo, and Claudio T. Silva. GLoG: Laplacian of Gaussian for Spatial Pattern Detection in Spatio-Temporal Data.
- Perozzi, B., Al-Rfou, R., Skiena, S. (2014, August). Deepwalk: Online learning of social representations.
- Mohler, G. O., Short, M. B., Brantingham, P. J., Schoenberg, F. P., Tita, G. E. (2011). Self-exciting point process modeling of crime.
- Mohler, G. (2014). Marked point process hotspot maps for homicide and gun crime prediction in Chicago.

GRACIAS