NUEVAS PROPUESTAS PARA BÚSQUEDAS POR SIMILITUD EN BASES DE DATOS MÉTRICAS

Karina Figueroa

Nora Reyes

Verónica Ludueña

Patricia Roggero

Contenido del curso (1/2)

- Conceptos Fundamentales de Espacios Métricos
  - Introducción y motivación
  - Definición de espacios métricos.
  - Funciones de Distancia: propiedades.
  - Tipos de búsquedas pos similitud más comunes.
  - Maldición de la dimensión.
- Índices para Bases de Datos Métricas
  - Taxonomía de los índices
  - Principales referentes de índices basados en particiones compactas.
  - Principales referentes de índices basados en pivotes.
  - Principales referentes de índices basados en permutaciones
  - Índices estáticos y dinámicos. Ejemplos.
  - Índices para memoria secundaria. Ejemplos.
- Algoritmos Exactos y Aproximados
  - Algoritmos Exactos.
  - Algoritmos Aproximados.
  - Medidas de evaluación de calidad de respuesta.

Contenido del curso (2/2)

- Otras operaciones de Interés sobre Bases de Datos Métricas:
  - Join por Similitud, variantes.
  - Algoritmos para Join: con índices y sin índices.
  - Ejemplos de soluciones existentes.
  - Medidas de evaluación de la dimensionalidad.

Operaciones de interés

- Join por similitud: dadas dos bases de datos $A, B \subseteq U$, encontrar todos los pares de objetos (un objeto desde cada base de datos) que satisfacen algún predicado de similitud Φ.

- El algoritmo de ***fuerza bruta*** para calcular el join por similitud necesita $|A| \times |B|$ cálculos de distancia: Iteración Anidada (NL).

- Se puede indexar uno o ambos conjuntos independientemente, y luego resolver consultas por rango para los elementos involucrados.

Join por similitud

- Formalmente, dados A, $B \subseteq U$ y siendo Φ un *criterio de similitud*, el ***join por similitud entre A y B*** se define como:

$$A \bowtie_{\Phi} B = \{(x, y) / (x \in A \wedge y \in B) \wedge \Phi(x, y)\} \subseteq A \times B$$

- Cuando A = B $\subseteq$ U, la operación se denomina ***auto-join***:

$$A \bowtie_{\Phi} A = \{(x, y) \in A^2 / \Phi(x, y)\} \subseteq A^2$$

Join por similitud

- Las variantes más comunes del join por similitud entre *A*, *B* $\subseteq$ *U*, son:

***Join por rango*:** dado un radio $r \geq 0$, se denota como $A \bowtie_r B$.

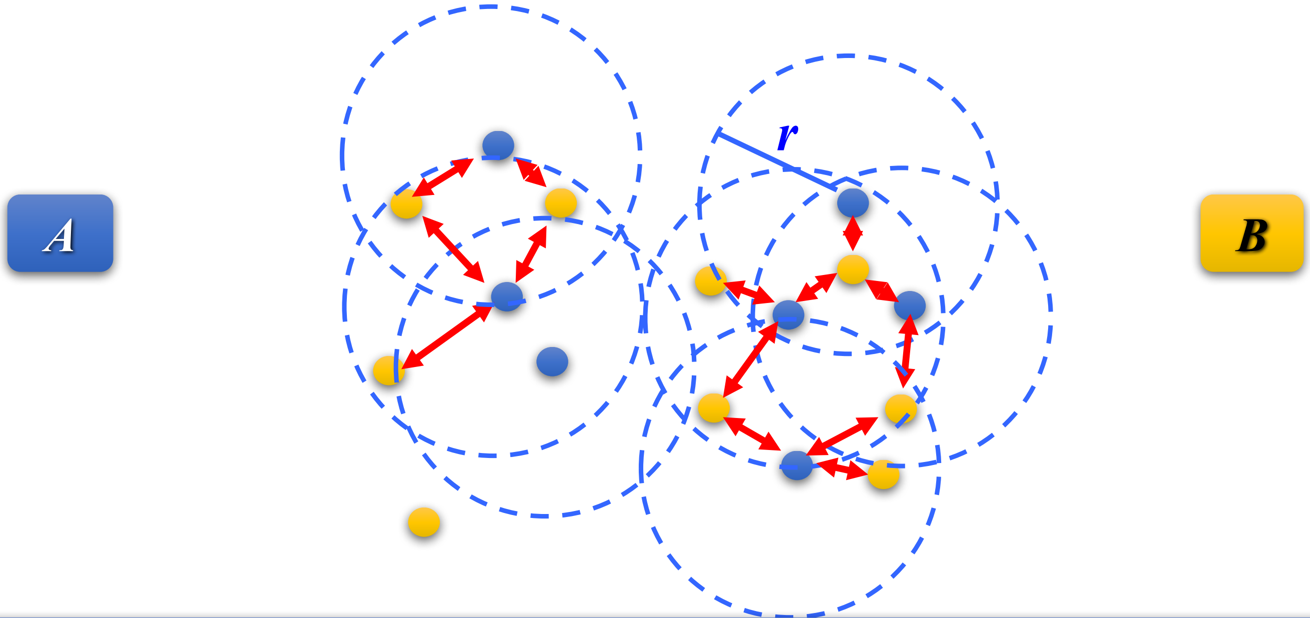
***Join de k-vecinos más cercanos*:** dado $k \in \mathbb{N}$, se denota como $A \bowtie_{kNN} B$.

***Join de k pares de vecinos más cercanos*:** dado $k \in \mathbb{N}$, se denota como $A \bowtie_k B$.

- Cuando, dado *A* $\subseteq$ *U* y *k* $\in \mathbb{N}$ se quiere obtener $A \bowtie_{kNN} A$, se habla del ***problema de All-k-NN***.

Join por similitud

- Tiene aplicaciones como minería de datos, limpieza de datos, integración de datos, entre otras.



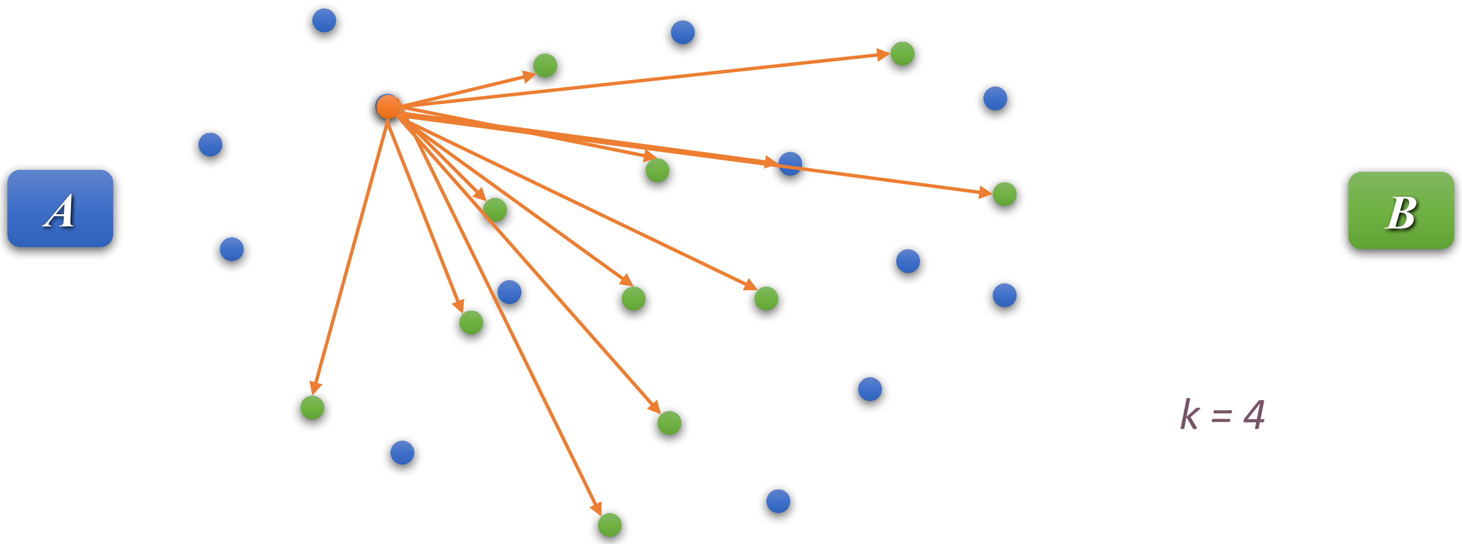
Se resuelven consultas por rango con radio r en B para todo elemento de A

Join por similitud

- Join de vecino más cercano:

Algoritmos de join sin índices

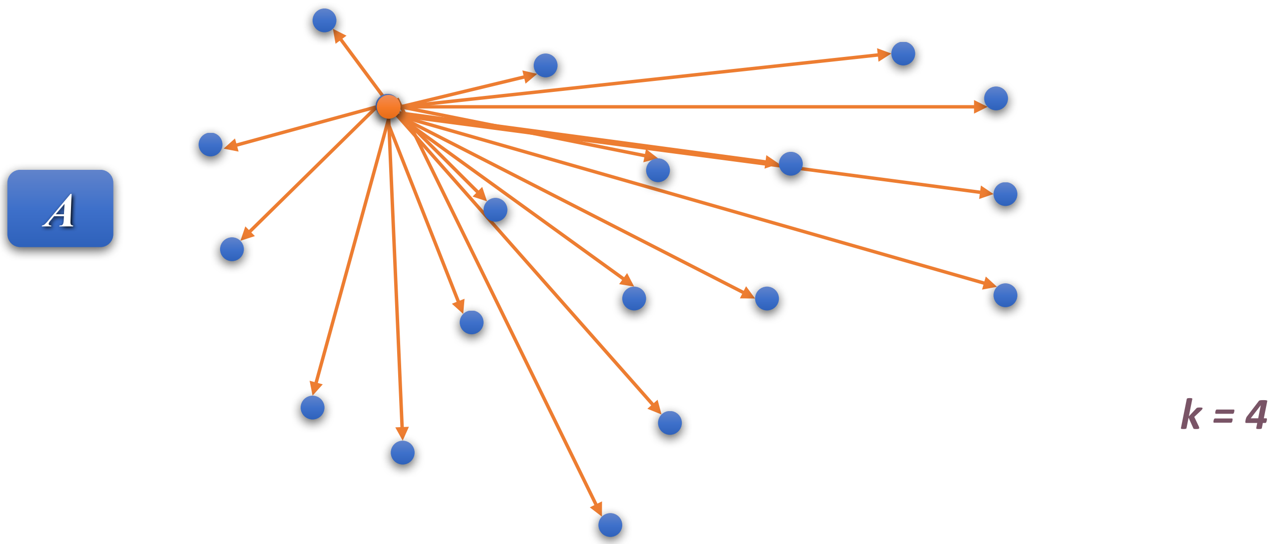
- Si dados $A, B \subseteq U$, se desea resolver el join y no se cuenta con índices, ¿cómo podría resolverse por ejemplo el join $A \bowtie_{kNN} B$?



¿Cómo podría continuar, tratando de aprovechar las distancias calculadas?

Algoritmos de join sin índices

- Si dada $A \subseteq U$, se desea resolver el auto-join y no se cuenta con índices, ¿cómo podría resolverse por ejemplo el join $A \bowtie_{kNN} A$?



¿Cómo podría continuar, tratando de aprovechar las distancias calculadas?

Algoritmos de join con índices

- Dados A, $B \subseteq U$, tales que $|A| > |B|$, existen distintas maneras de usar índices para join:
  - Construir un índice para la base de datos A (I_A) y resolver consultas para cada uno de los $b \in B$ en el índice I_A.
  - Construir un índice para la base de datos A (I_A) y otro para la base de datos B (I_B) y luego aprovechar la información del índice I_B para resolver consultas para cada uno de los $b \in B$ en el índice I_A.
  - Construir un índice para $A \cup B$ ($I_{A \cup B}$) y resolver el join aprovechando la información del índice $I_{A \cup B}$.

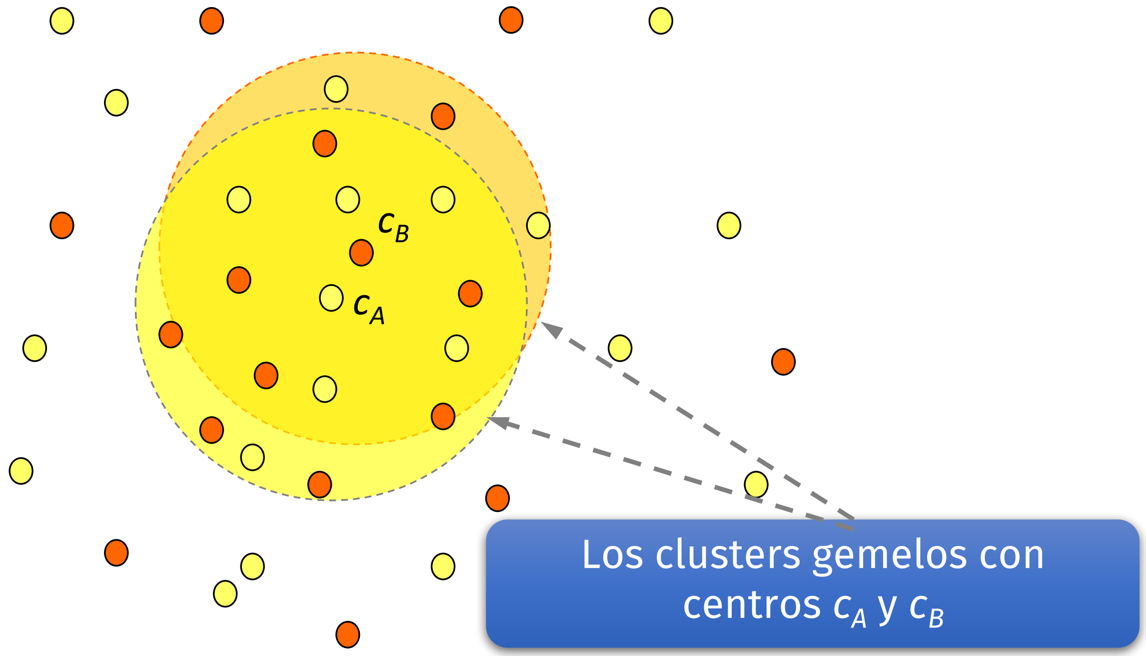
Solving similarity joins and range queries in metric spaces with the list of twin clusters

[Paredes and Reyes, 2009] Rodrigo Paredes and Nora Reyes. ***Solving similarity joins and range queries in metric spaces with the list of twin clusters***. Journal of Discrete Algorithms (JDA), 7:18–35, March 2009.

.

Lista de Clusters Gemelos (LTC)

- La LTC está basada en la variante de la LC que usa particiones de radio fijo.
- Se consideran dos listas de clusters que se solapan, por ello se denominan ***clusters gemelos***.



Lista de Clusters Gemelos

- Los clusters con centros de *A* encierran a elementos de *B* y viceversa.
- Se considera un radio de construcción *R* y se almacena en cada clúster su radio efectivo.
- En la LTC los elementos de las bases de datos pueden clasificarse en tres grupos:
  - Objetos que actúan como ***centros***.
  - Objetos que ***pertenecen a un clúster***.
  - Objetos ***no indexados***.

Lista de Clusters Gemelos

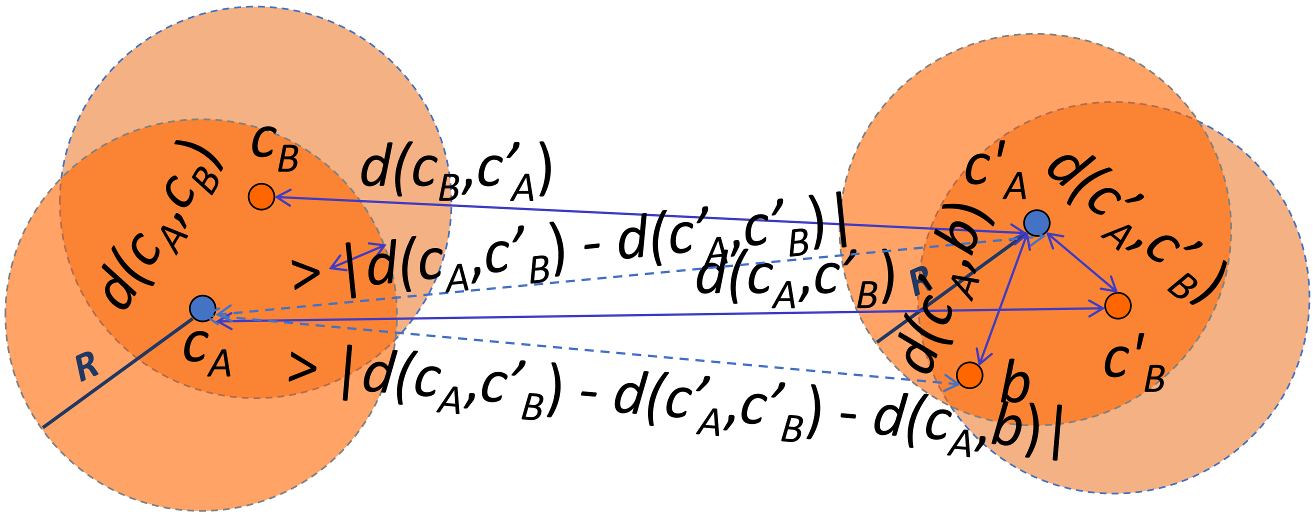
- La LTC se puede usar para calcular:
  - el join por rango
  - el join de *k* pares de vecinos más cercanos
  - consultas por rango sobre la unión de ambas bases de datos.
- Para construir el índice o para obtener el join ***nunca*** se calculan *distancias internas* entre elementos de la misma base de datos.
- Para evaluar el desempeño se consideran la aproximación ingenua (NL) y usar LC sobre una o las dos bases de datos.

Lista de Clusters Gemelos

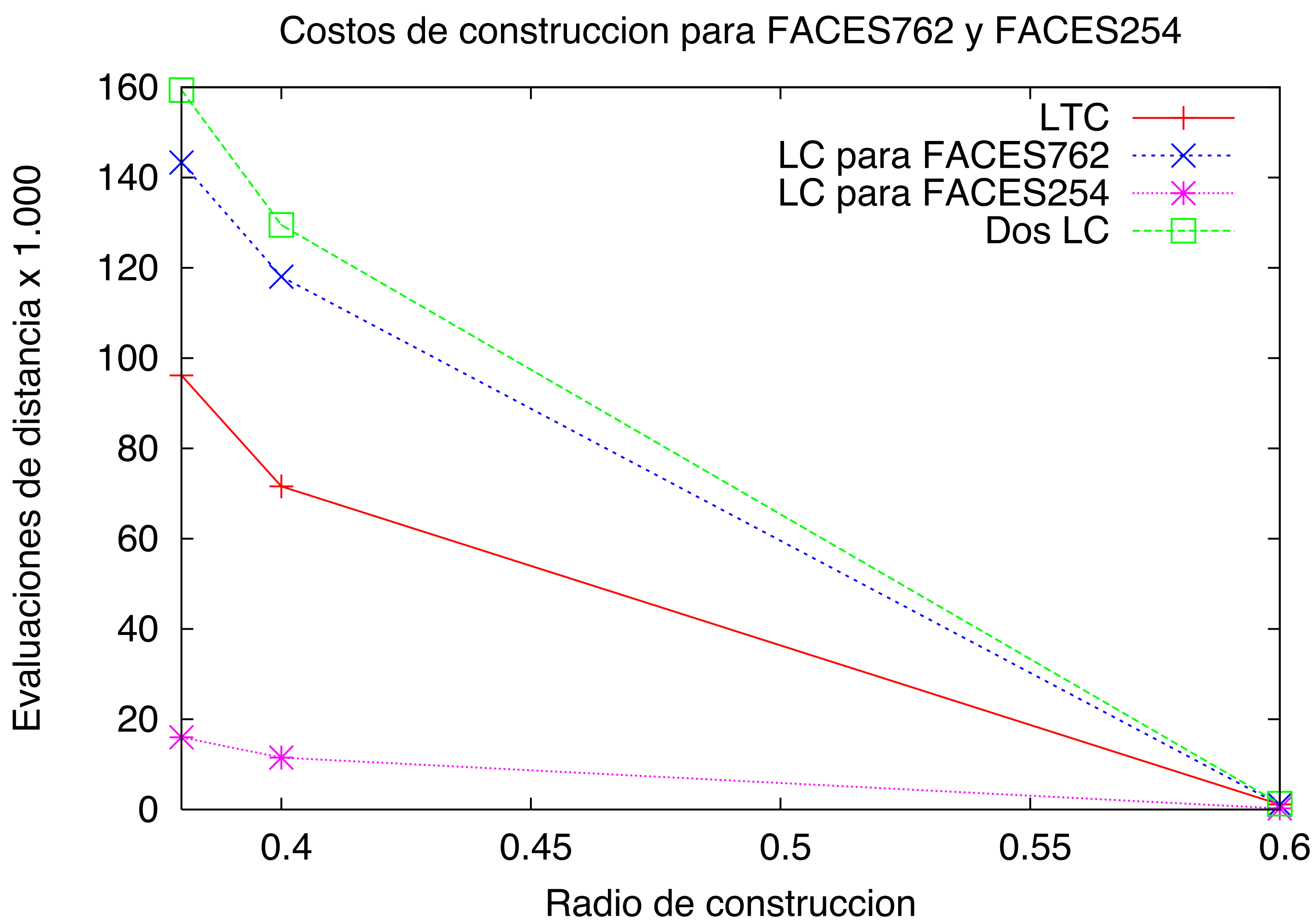
- La estructura consta de:
  - Dos listas de clusters gemelos C_A y C_B. Los centros de los clusters C_A y C_B pertenececen a la base de datos A y B, respectivamente y los objetos en sus clusters internos pertenecen a las bases de datos B y A respectivamente.
  - Una matriz D con las distancias calculadas entre todos los centros de la base de datos A a los centros de la base de datos B.
  - Vectores de distancias máximas y mínimas de los objetos no indexados (pueden ser de la base de datos A o bien de la base de datos B) a los centros de la otra base de datos.

No se calcula ninguna distancia entre elementos de la misma base de datos

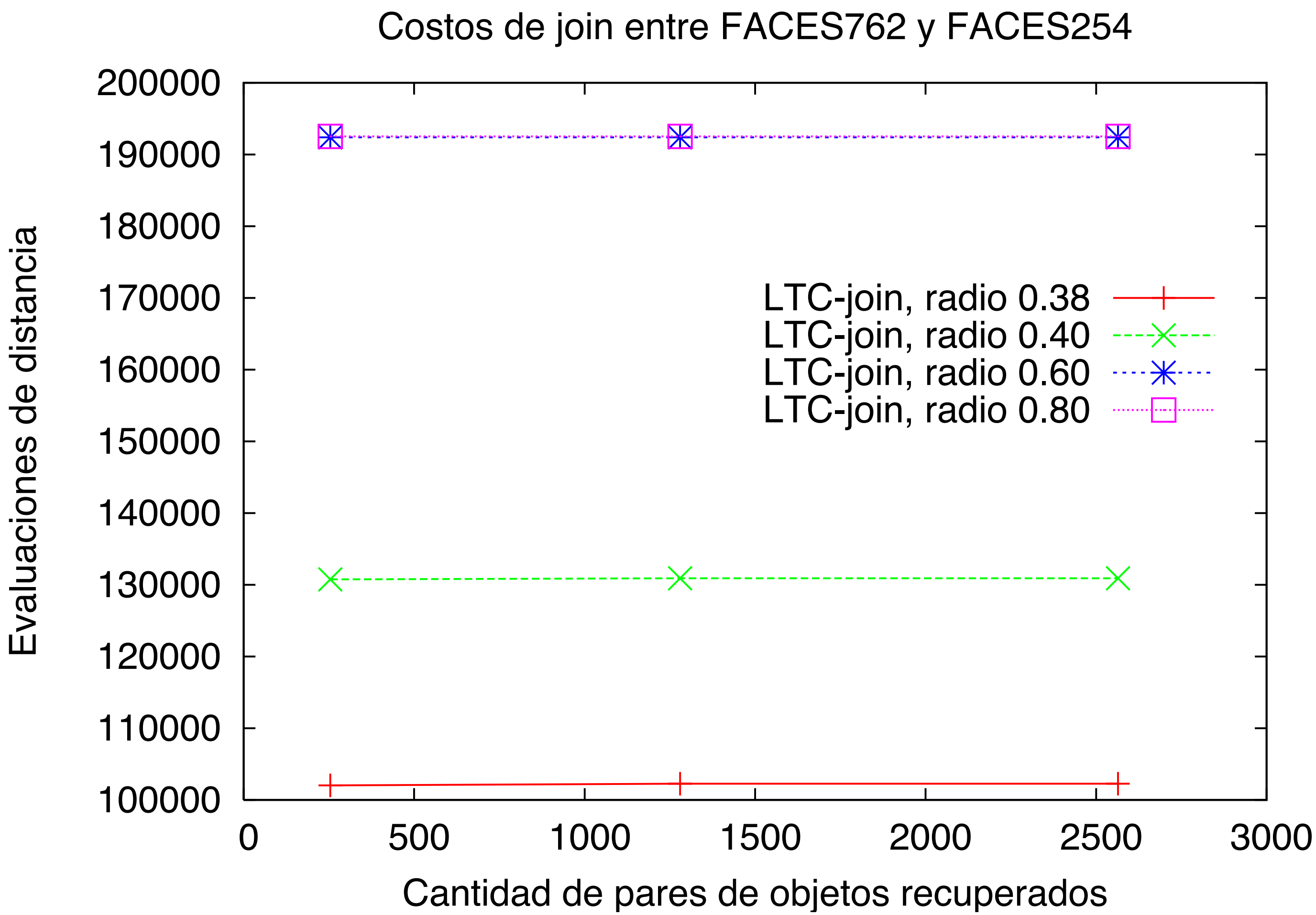
Lista de Clusters Gemelos



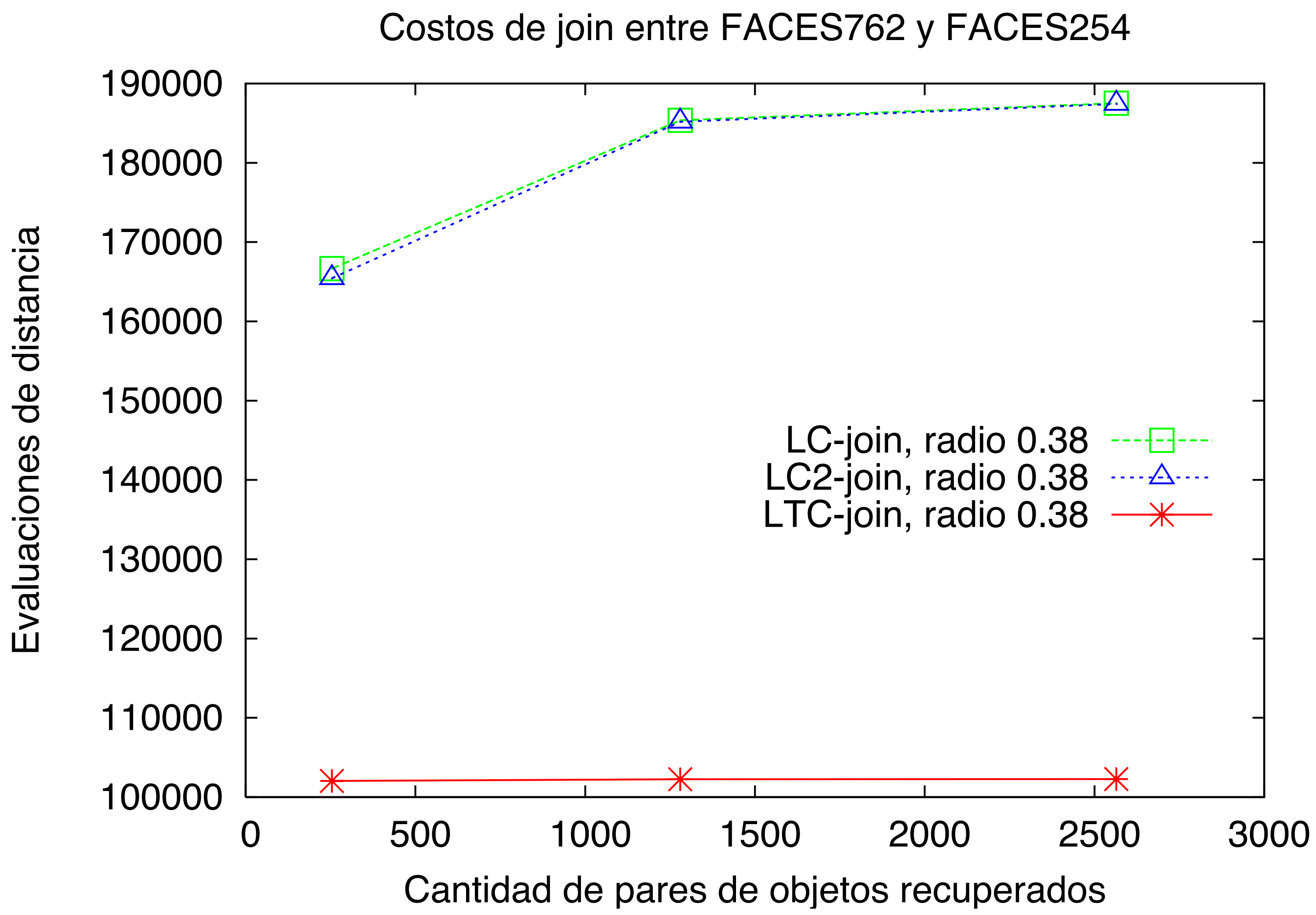
Lista de Clusters Gemelos



Lista de Clusters Gemelos



Lista de Clusters Gemelos



Similarity join in metric spaces using eD-index

[Dohnal, Gennaro, and Zezula, 2003] Vlatislav Dohnal, Claudio Gennaro, and Pavel Zezula. ***Similarity join in metric spaces using eD-index***. In *Proc. 14th Intl. Conf. on Database and Expert Systems Applications (DEXA'03)*, LNCS 2736, pages 484–493, 2003.

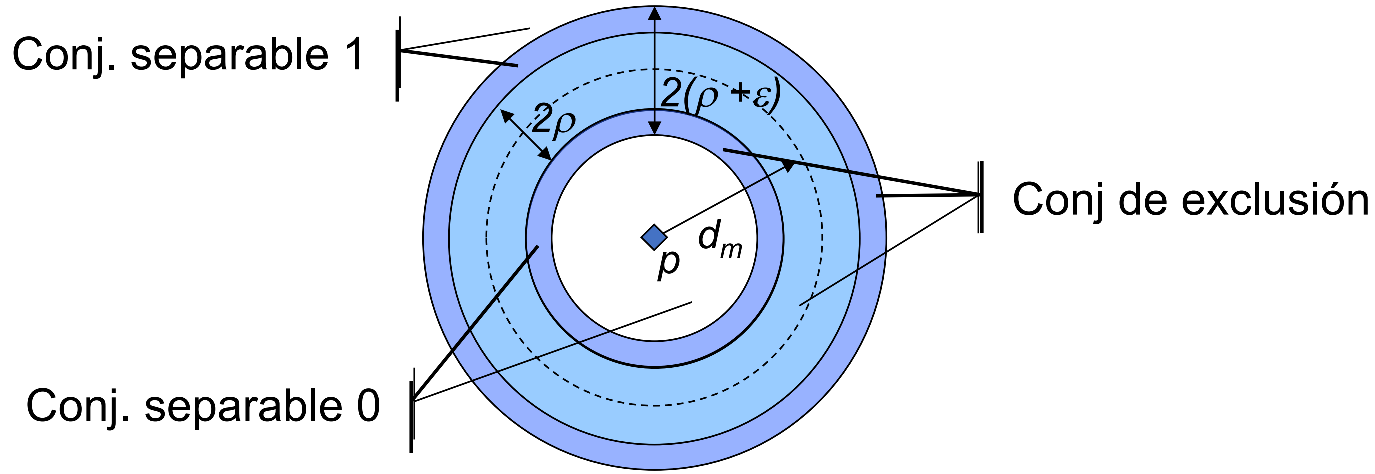
eD-Index

- Es una variante del D-index con un algoritmo especializado para joins por similitud.

- Las funciones *split* manejan replicación de objetos.

- Modifica levemente los algoritmos de búsqueda por rango y de *k*-NN del D-index.

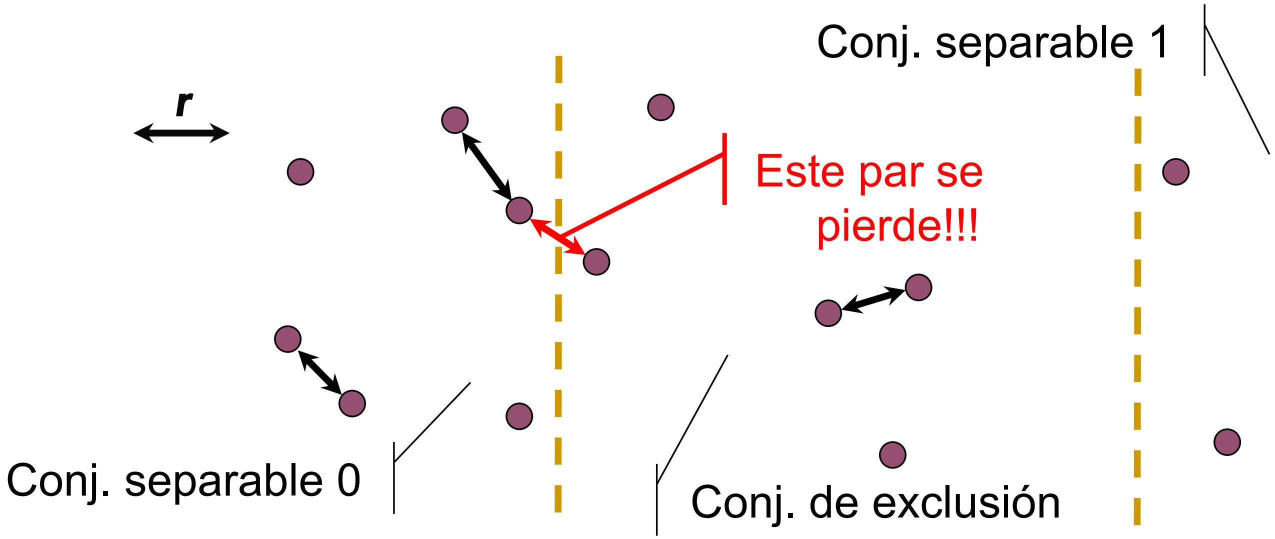
eD-Index



$$bsp^{1,\rho}(x) = \begin{cases} 0 & \text{if } d(x,p) \leq d_m - \rho - \varepsilon \\ 0 \quad \textit{copy} & \text{if } d(x,p) > d_m - \rho - \varepsilon \wedge d(x,p) \leq d_m - \rho \\ 1 & \text{if } d(x,p) > d_m + \rho + \varepsilon \\ 1 \quad \textit{copy} & \text{if } d(x,p) > d_m + \rho \wedge d(x,p) \leq d_m + \rho + \varepsilon \\ - & \text{otherwise} \end{cases}$$

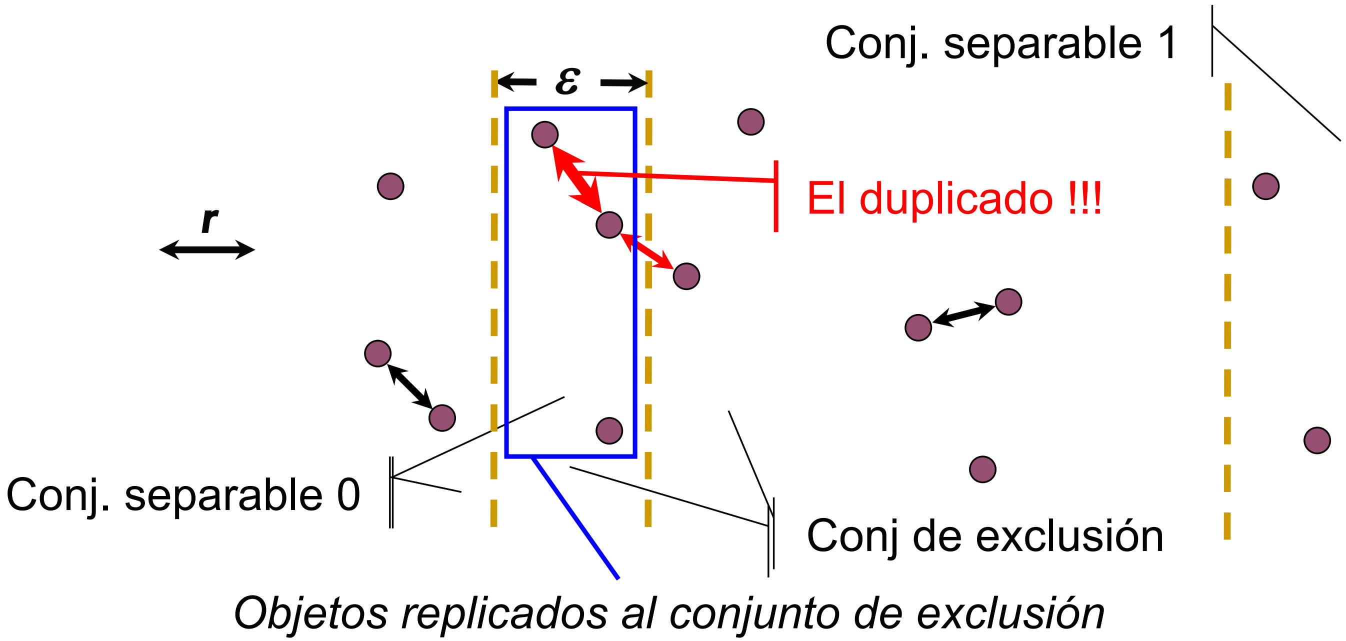
eD-Index

- El auto join por similitud se obtiene independientemente en cada bucket, y el resultado final es la unión de respuestas a las todas las subconsultas de cada bucket.



eD-Index

- Áreas de ancho ε se replican en el conjunto de exclusión.



An Empirical Evaluation of Intrinsic Dimension Estimators

[Navarro, Paredes, Reyes, Bustos, 2017] Gonzalo Navarro, Rodrigo Paredes, Nora Reyes, Cristian Bustos, ***An empirical evaluation of intrinsic dimension estimators***, Information Systems, Volume 64, March 2017, Pages 206-218, ISSN 0306-4379.

Dimensión intrínseca

- **Dimensión de Representación**: cantidad de coordenadas de los vectores.

- **Dimensión Intrínseca**: número real de dimensiones en las cuales los elementos pueden ser embebidos manteniendo las distancias entre ellos.

Ambas pueden ser muy distintas entre sí

Dimensión intrínseca

- Un mapeo $\mathbf{\Phi}$ de un espacio *S* al espacio embebido *R* preserva las distancias si se cumple que:

$$d(x,y) \leq d(x,z) \Rightarrow D(\Phi(x),\Phi(y)) \leq D(\Phi(x),\Phi(z))$$

donde:

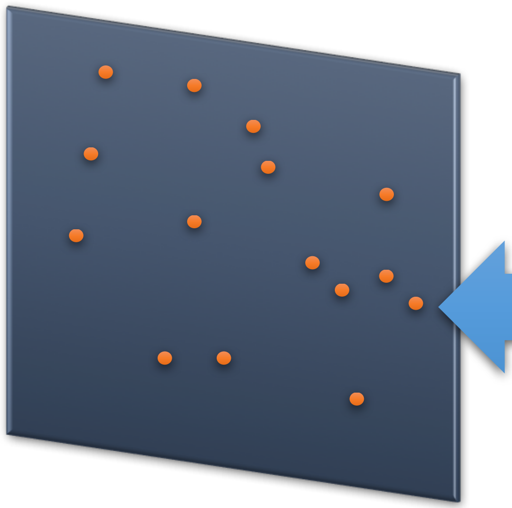
$$\Phi : S \mapsto R$$

$$d : S \times S \mapsto \mathbb{R}^+$$

$$D : R \times R \mapsto \mathbb{R}^+$$

Dimensión intrínseca

- Ejemplo: los elementos se pueden embeber en un plano.



Cada elemento se representa por un vector de 100 coordenadas

Dimensión Intrínseca = 2

Dimensión Representacional = 100

Dimensión intrínseca

- En espacios vectoriales uniformes, la maldición de la dimensión describe el conocido *incremento exponencial del costo de los algoritmos de búsqueda a medida que aumenta la dimensión del espacio*.

- En espacios métricos generales, a pesar de la ausencia de coordenadas en los objetos que se manejan, en algunos espacios las búsquedas son *intrínsecamente más difíciles* que en otros.

Dimensión intrínseca

- Esto ha guiado al concepto de ***dimensión intrínseca*** (**DI**) de un espacio métrico, como una medida de la dificultad de buscar en él.

- En muchos casos los elementos están representados como vectores de R^D.

- ***Maldición de la dimensión***: todas las técnicas se degradan cuando la dimensión intrínseca del espacio crece, *pero no con la dimensión de representación.*

Dimensión intrínseca

- Ejemplo de caso extremo:

$$\forall x,\ \ d(x,x) = 0$$

$$\forall x, y \ x \neq y \Rightarrow d(x,y) = 1$$

No es posible evitar una búsqueda secuencial en este caso

Dimensión intrínseca

- Calcular la DI de un espacio métrico puede ser útil por ejemplo para:
  - Determinar si es posible realmente indexar el espacio: si la DI del espacio fuera demasiado alta, entonces se deberían aplicar directamente soluciones de *fuerza bruta* o *algoritmos de búsqueda aproximados*
  - Para decidir qué clase de índice usar: los métodos basados en pivotes funcionan mejor sobre espacios de dimensión baja, mientras que los métodos basados en particiones compactas los superan en dimensiones altas.

Estimadores de DI para espacios vectoriales

- La DI de un espacio es el *mínimo número de variables libres necesarias para representar los datos sin pérdida de información*.

- En términos generales, un espacio X $\subseteq$ R^D tiene DI $M \leq D$, si sus elementos caen completamente dentro de un subespacio *M*-dimensional de R^D.

- Otra noción intuitiva es el logaritmo del costo de búsqueda, pero este costo crece exponencialmente a medida que crece la dimensión.

Estimadores de DI para espacios vectoriales

- Un conjunto de datos $X \subseteq \mathrm{R}^D$, con $|X| = n$, requiere almacenar $n \times D$ coordenadas reales. Si DI(X) = $M \leq D$, los puntos pueden ser mapeados en R^M, y en ese caso se requiere almacenar sólo $n \times M$ coordenadas reales.

- El costo de calcular cualquier distancia *se reduce*.

- Existen dos aproximaciones para estimar la DI de un espacio vectorial:

  - La ***local*** usa la información contenida en vecindarios de muestra.

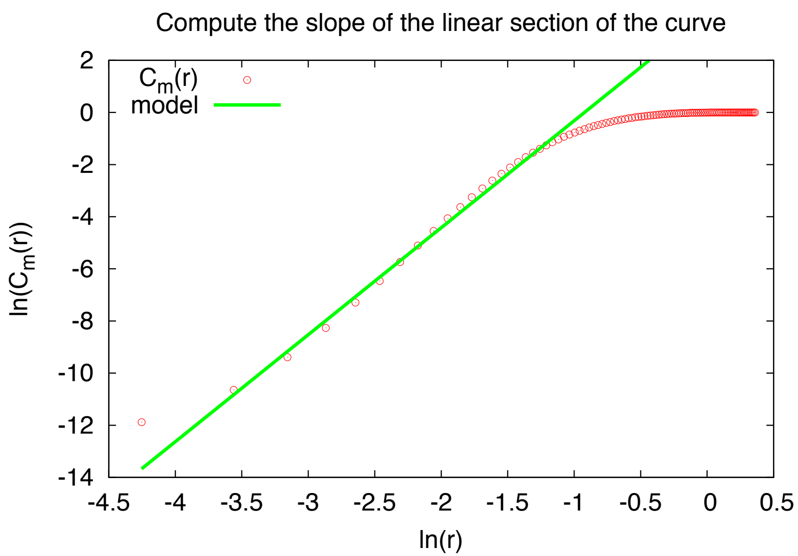
  - La ***global*** despliega el conjunto de datos sobre un espacio *M*-dimensional usando toda la información del conjunto de datos.

Estimadores de DI para espacios vectoriales

- Existen varios métodos para estimar la DI de espacios vectoriales y otras para espacios métricos generales:
  - Correlación
  - Análisis de Componente Principal
  - Métodos basados en fractales
  - Exponente de distancia
  - Sparse Spatial Selection (SSS)
  - Fastmap
  - Dificultad intrínseca de la búsqueda

Correlación

- Consiste en graficar el ln(*Cm*(*r*)) versus ln(*r*), donde *Cm*(*r*) es la fracción de pares de objetos cuya distancia es menor que *r.*
- La ***dimensión por correlación*** es la pendiente de la sección lineal de la curva.



Análisis de componente principal (PCA)

- Es un procedimiento estadístico que proyecta los datos sobre nuevos ejes, llamados las ***componentes principales***, donde los ejes se ordenan de máxima a mínima varianza.
- Como las primeras componentes acumulan la mayoría de la varianza, los datos originales se pueden proyectar usando las primeras componentes controlando cuánta información se desea preservar o perder.
- Una aplicación común de PCA es reducir la dimensión de un conjunto de vectores, descartando las componentes con pequeña varianza.

Análisis de componente principal (PCA)

- Se eligen pivotes al azar y se representa cada objeto por un vector de distancias a los pivotes.
- Se computan las componentes principales desde la tabla de pivotes.
- Las componentes se ordenan por importancia: *en orden decreciente de la varianza de los datos*.
- Se usa la cantidad de componentes que acumule el 90% de la varianza del conjunto de datos como estimación de la DI del espacio.

Métodos basados en fractales

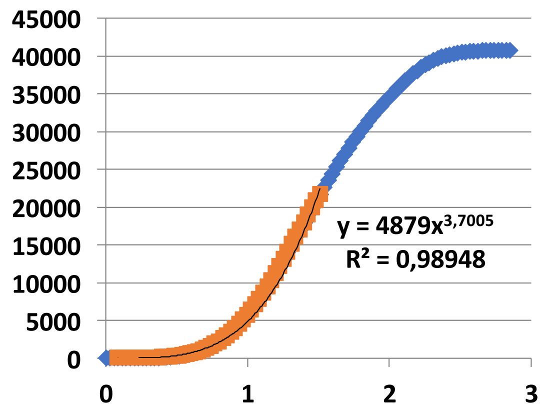
- La estimación de la dimensión por ***Conteo de Cajas*** D_B de un conjunto $\Omega \subseteq \mathrm{R}^D$ está definido como sigue: si $v(r)$ es el número de cajas de tamaño r necesario para cubrir Ω, entonces:

$$D_B = \lim_{r \to 0} \frac{\ln(v(r))}{\ln\left(\frac{1}{r}\right)}$$

- En el caso métrico, se estima D_B por ***Conteo de Bolas.***
  Se considera $B(r)$, el número de bolas de radio r de una *Lista de Clusters* para cubrir a Ω, y se usa $B(r)$ en lugar de $v(r)$.

Exponente de distancia

- Varios conjuntos cumplen con la ***Ley de Potencia.***
- Se denomina ***Exponente de Distancia*** al exponente de la ley de potencia, y permite derivar fórmulas para estimar la selectividad de las consultas por rango.
- ***Ley de Distancia*** – *Dado un conjunto de n objetos de un espacio métrico con una función de distancia d(x, y), el número promedio de distancias menores o iguales que un radio r sigue una ley de potencia; es decir, el número de vecinos nb(r) dentro de una distancia r es proporcional a* r^D.



Sparse Spatial Selection (SSS)

- En este método se eligen los pivotes de manera tal que alrededor de cada pivote cubre al menos una bola de radio αd^{+}.
- El método selecciona nuevos pivotes si cubren alguna parte del espacio aún no cubierta por algún pivote previo (≈ método Fractal)
- Se puede usar una técnica similar para estimar DI.
- Sea $P(\alpha)$ el número de pivotes producido por SSS para un valor α dado. Se grafica $\ln P(\alpha)$ contra $\ln\frac{1}{\alpha}$ y se obtiene la pendiente de la parte lineal de la curva usando regresión lineal con mínimos cuadrados sobre los datos experimentales $(\ln(P(\alpha)), \ln(\frac{1}{\alpha}))$.

Fastmap

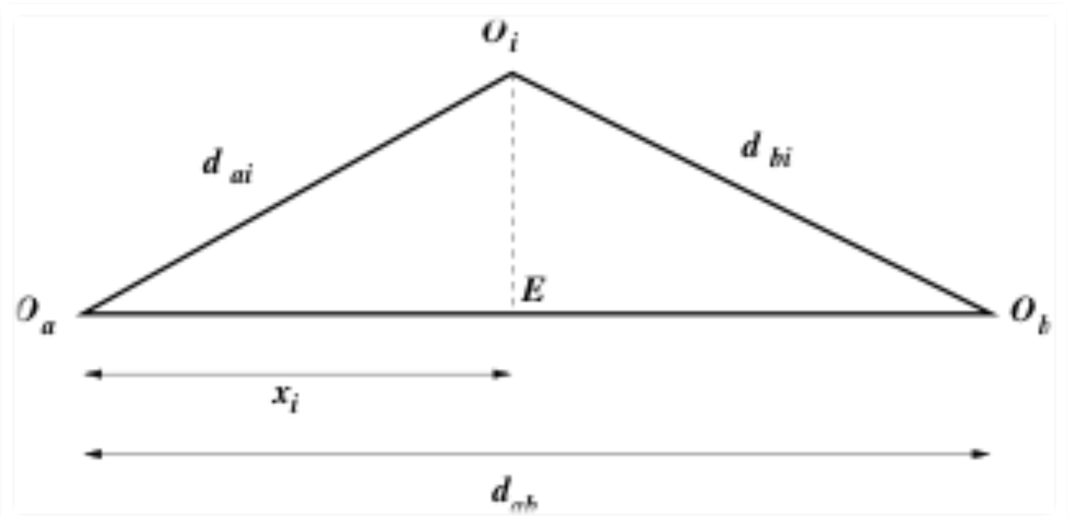
- Es un algoritmo rápido que mapea objetos de cualquier espacio métrico en puntos de un espacio k-dimensional, de manera tal que las disimilitudes se preserven.
- Para evaluar la preservación de la disimilitud en el espacio objetivo, se define una función ***stress*** como sigue:

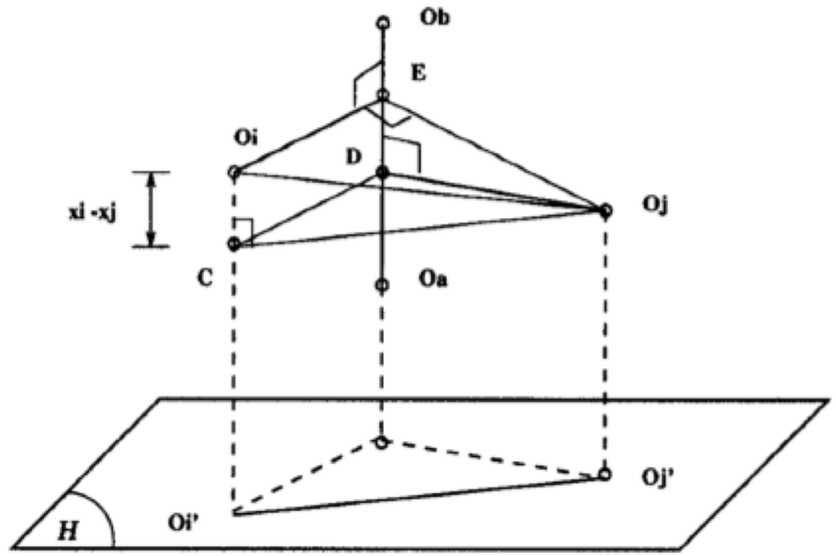
$$stress^2 = \frac{\left(\sum_{i,j}(\hat{d}_{ij} - d_{ij})^2\right)}{\left(\sum_{i,j} d_{ij}^2\right)}$$

donde d_{ij} es la distancia original entre los objetos o_i y o_j, y and $\hat{d}_{ij}$ es la distancia euclidiana entre sus imágenes p_i y p_j.

- Se estima el número de proyecciones necesarias para que el espacio objetivo alcance un mapeo con un *stress* suficientemente pequeño.

Fastmap





Ley de Coseno: proyección sobre la recta $\overline{O_a O_b}$

Proyección sobre un hiperplano *H*, perpendicular a la recta $\overline{O_a O_b}$

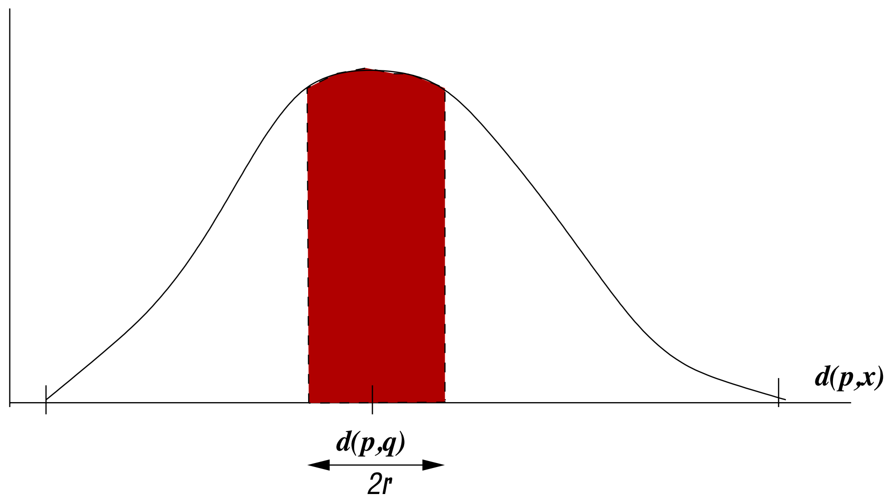
Dificultad intrínseca de la búsqueda

- Existe una medida para la complejidad intrínseca de las búsquedas en espacios métricos generales, fácil de estimar e independiente del algoritmo de búsqueda utilizado.

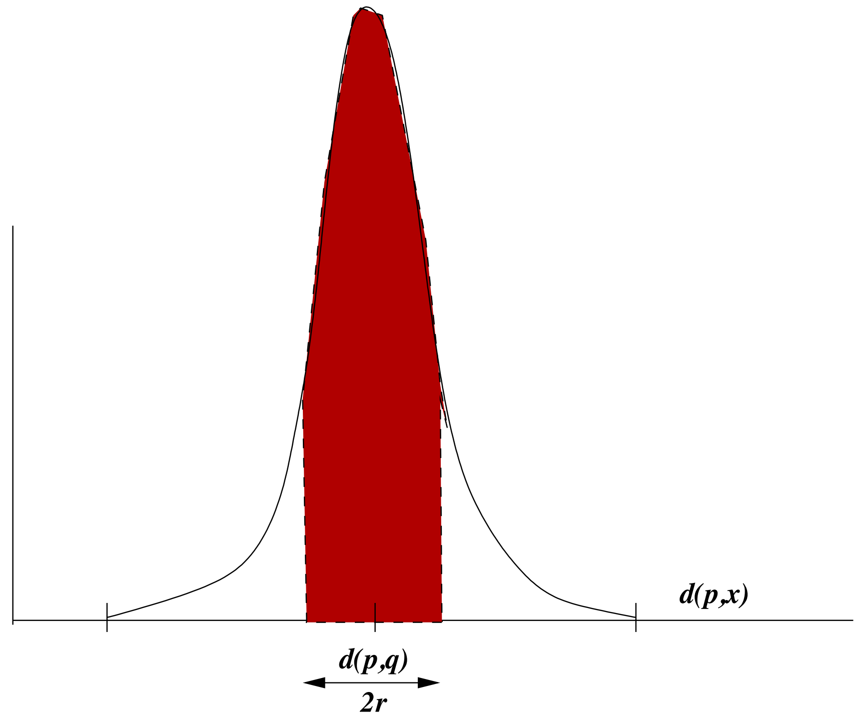
- *Definición:* Sean μ la media y σ^2 la varianza del histograma de distancias de un espacio métrico. La dificultad intrínseca de la búsqueda es:

$$\rho = \frac{\mu^2}{2\sigma^2}$$

Dificultad intrínseca de la búsqueda

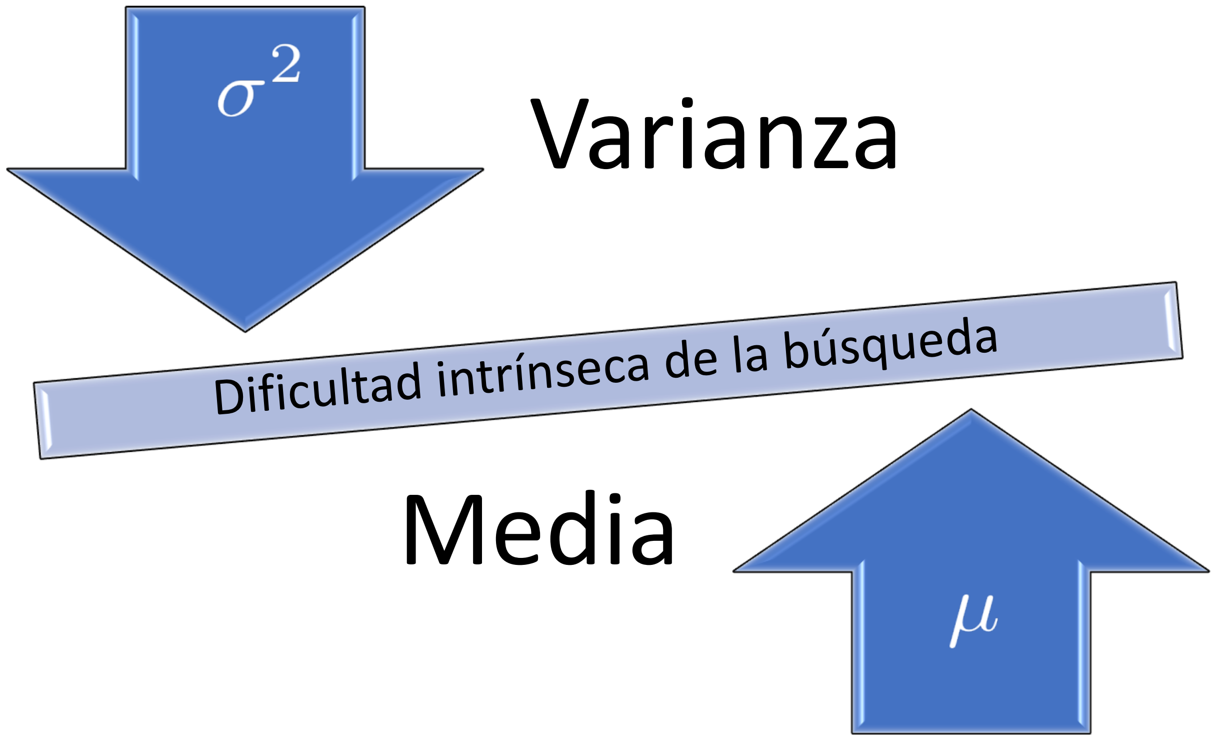


Espacio de baja dimensión



Espacio de alta dimensión

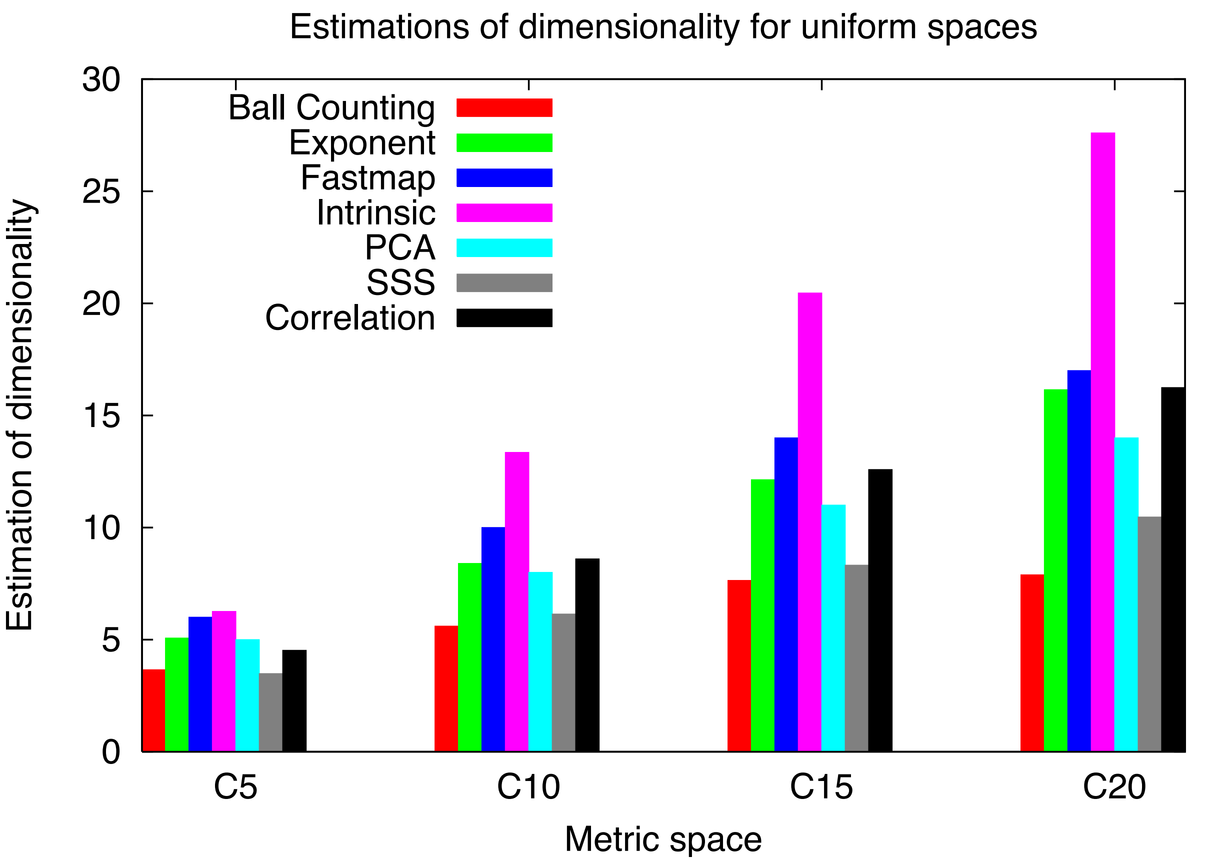
Dificultad intrínseca de la búsqueda



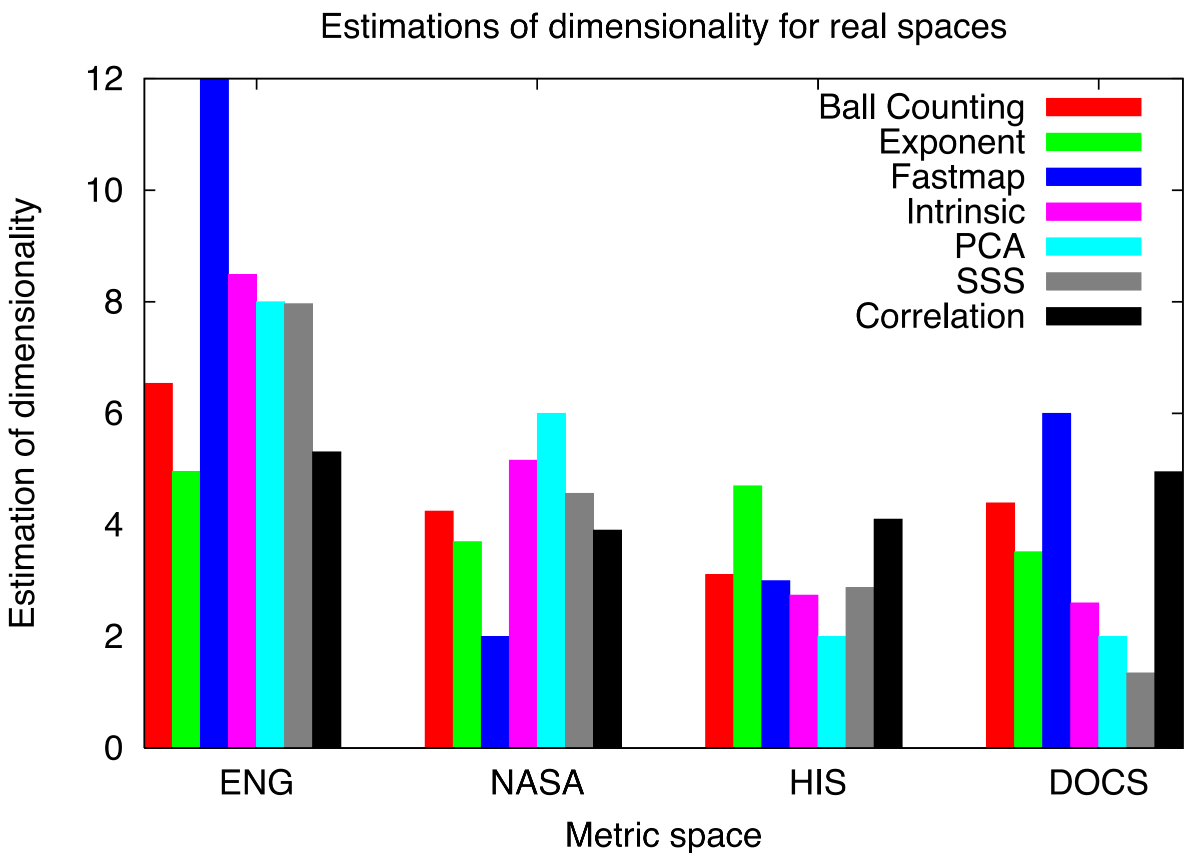
Resultados experimentales

- Consideramos dos clases de espacios métricos dependiendo de la fuente de los datos:
  - Sintéticos: estos espacios se generan artificialmente de manera tal que presenten características interesantes de evaluar. Por ejemplo, vectores uniformemente distribuidos en R^D con dimensión conocida.
  - Reales: estos espacios se obtienen desde aplicaciones del mundo real. Por ejemplo, un espacio de vectores de características de imágenes obtenidas desde un conjunto de imágenes de la NASA.

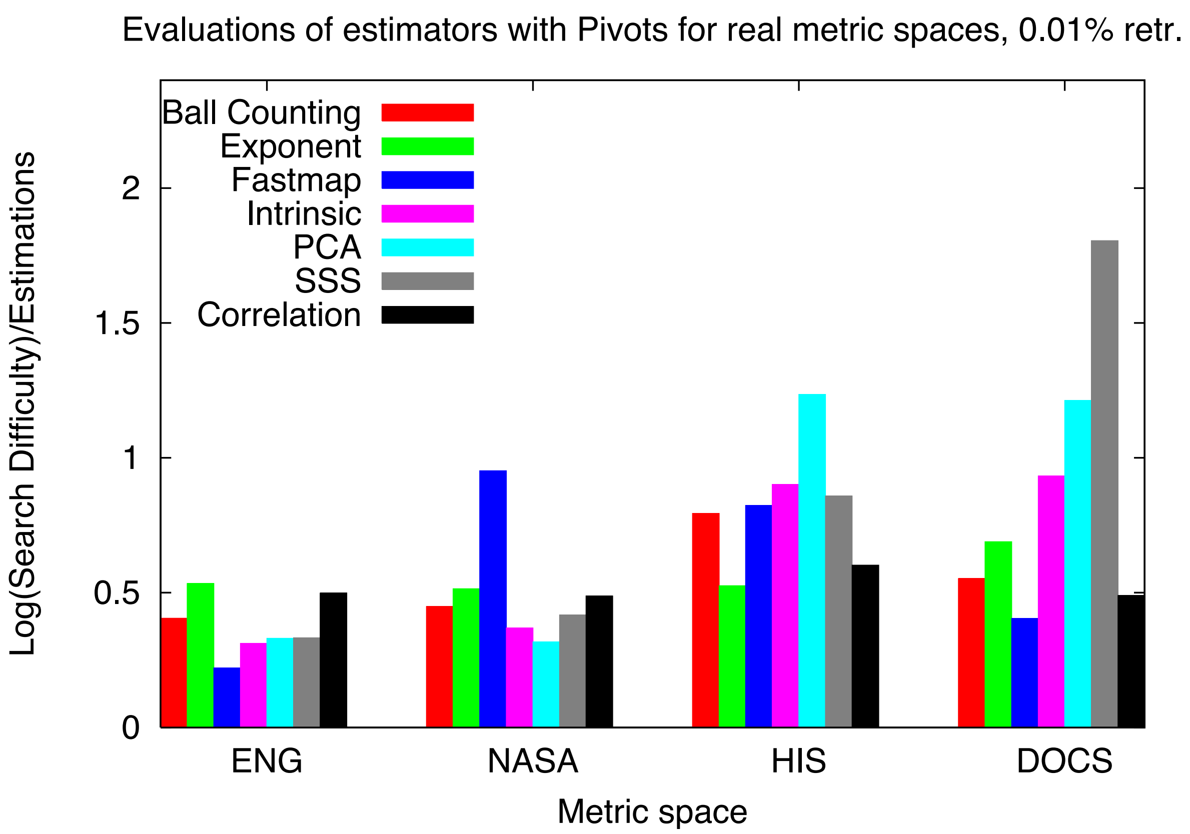
Resultados experimentales



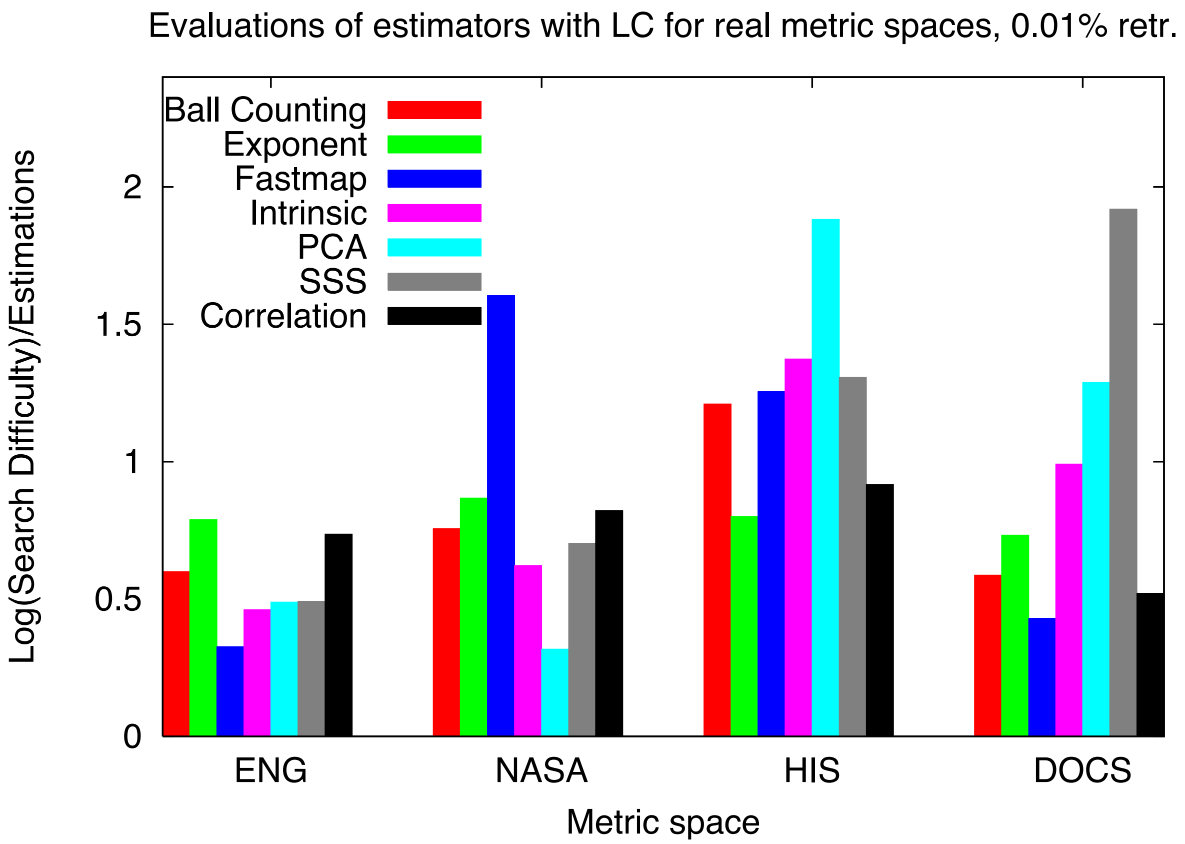
Resultados experimentales



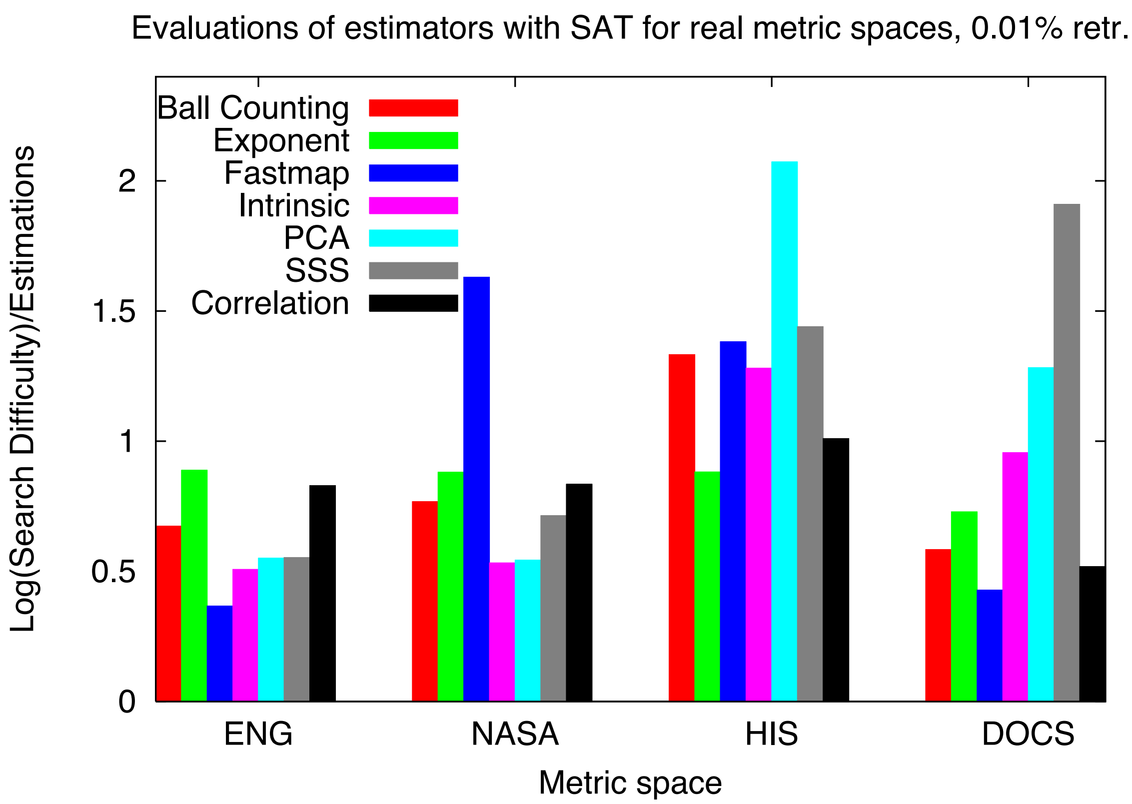
Resultados experimentales



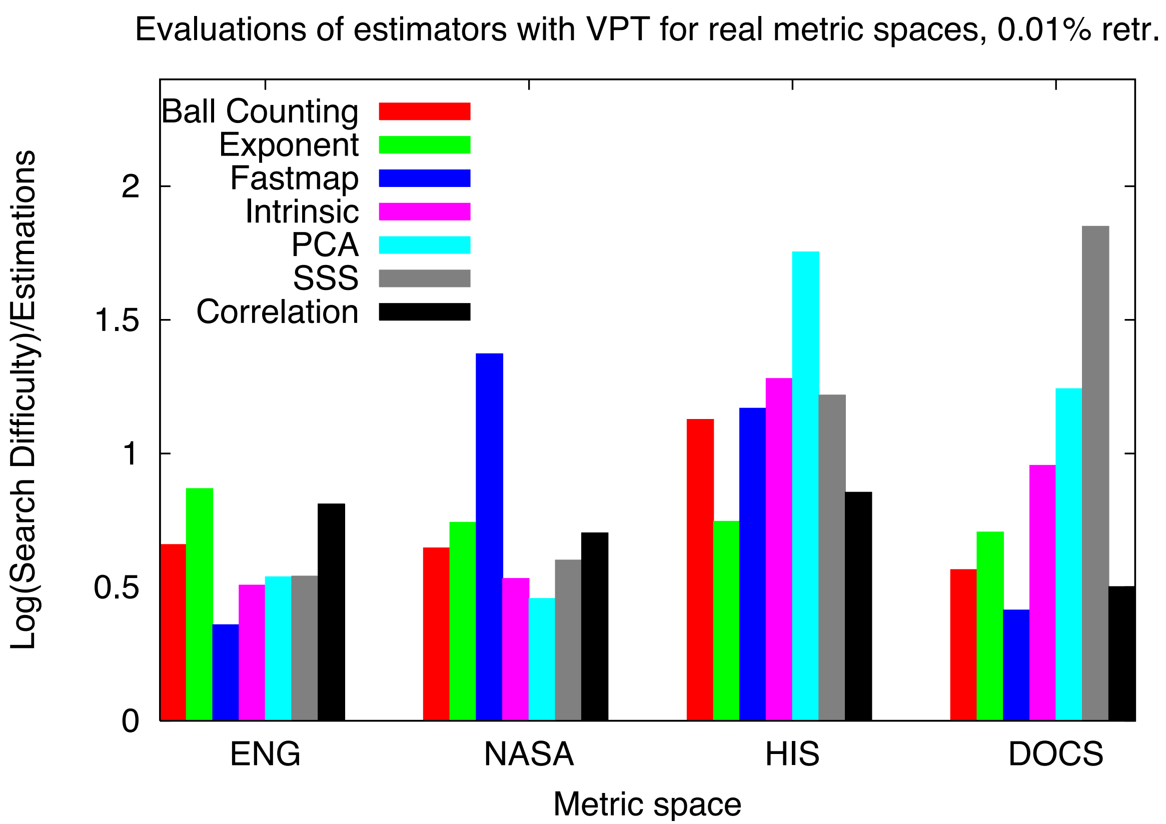
Resultados experimentales



Resultados experimentales



Resultados experimentales



Resultados experimentales

- Comparación de log(Dif. Búsqueda)/Estimación para los siete estimadores de DI.

IDim Estimator	**Mean**	**Standard Deviation**
Ball Counting	0.810	0.289
Distance Exponent	0.822	0.135
Fastmap	0.924	0.592
Intrinsic Search Difficulty	0.861	0.364
PCA	1.068	0.605
SSS	1.121	0.556
Correlation	0.773	0.200

Resultados experimentales

- Comparación de log(Dif. Búsqueda/Tamaño de la BD)/Estimación para los siete estimadores de DI.

IDim Estimator	**Mean**	**Standard Deviation**
Ball Counting	0.218	0.175
Distance Exponent	0.212	0.149
Fastmap	0.287	0.302
Intrinsic Search Difficulty	0.255	0.196
PCA	0.279	0.247
SSS	0.283	0.186
Correlation	0.204	0.156

Resultados experimentales

- Comparación de log(Dif. Búsqueda/Tamaño de la Rta)/Estimación para los siete estimadores de DI.

IDim Estimator	**Mean**	**Standard Deviation**
Ball Counting	0.427	0.173
Distance Exponent	0.448	0.164
Fastmap	0.398	0.281
Intrinsic Search Difficulty	0.491	0.280
PCA	0.599	0.406
SSS	0.681	0.553
Correlation	0.407	0.137

¡¡Muchas gracias!!