

Mapeamiento visual interactivo de la evolución temporal de  
datos multidimensionales usando árboles filogenéticos  
enfocado en artículos científicos

*Roberto Josué Rodríguez Urquiaga*

**Asesor:** *Dra. Ana Maria Cuadros Valdivia*

Tesis presentada a la Universidad Nacional de San  
Agustín como parte de los requisitos para obtener  
el grado académico de Maestro en Informática con  
Mención en Tecnologías de Información.

**UNSA - Arequipa**  
**Agosto - 2016**

---

Visualización de la evolución temporal de datos  
multidimensionales usando árboles filogenéticos  
enfocado en artículos científicos

*Roberto Josué Rodríguez Urquiaga*

---

# Resumen

---

En la actualidad la cantidad de datos producida por los medios electrónicos a tenido un alza importante tanto en número de registros como en complejidad, como consecuencia identificar información útil de estos grandes conjuntos de datos se ha vuelto un reto, una gran parte de esta aumento en la cantidad de datos viene relacionado a las colecciones de artículos científicos que recientemente su exploración a despertado el interés de la comunidad científica, como consecuencia se han realizado trabajos para facilitar el análisis de estas colecciones de documentos usando técnicas de minería de datos y visualización de información, dentro de las técnicas de visualización de datos multidimensionales se encuentran las *Proyección multidimensional* que permiten reducir de una dimensiolidad alta a un espacio de dimensión ya sea 1,2,3 conservando las características de similitud de los datos en la dimensión original haciendo posible encontrar patrones a través de la capacidad visual humana. La mayoría de enfoques de proyecciones multidimensional de datos no considera el componente temporal de las colecciones de artículos científicos a pesar de tener el componente temporal un papel crucial en muchos tipos de datos, en este trabajo se propone incorporar el tratamiento del componente temporal en una proyección multidimensional basado en árboles filogenéticos de modo que sea apropiado para tareas de análisis exploratoria envolviendo la evolución de temas en colecciones de artículos científicos.

**Palabras Clave:** Visualización temporal de documentos, evolución temporal de temas, árboles filogenéticos, proyección de datos multidimensionales.



# Abstract

---

Currently the amount of data produced by the electronic media had a significant increase in number of records and complexity, as a result identify useful information from these large data sets has become a challenge, a large part of this increase in the amount of data is related to collections of scientific papers recently its exploration aroused the interest of the scientific community, following work has been done to facilitate analysis of these collections of documents using techniques of data mining and information visualization within visualization techniques multidimensional data are the textit multidimensional projection that reduce a high dimensiolidad to a space dimension either 1, 2, 3 preserving the characteristics of similarity of the data in the dimension Original making it possible to find patterns through the human visual capacity. Most approaches to multidimensional data projections does not consider the temporal component of the collections of scientific papers despite having the time component a crucial role in many types of data, this paper intends to incorporate the treatment of temporal component in a projection multidimensional based on phylogenetic trees so that it is appropriate for exploratory analysis tasks involving the evolution of issues in collections of scientific articles.

**Keywords:** Temporal visualization of documents, temporal evolution of topics, phylogenetic trees, multidimensional data projection



# Índice general

---

Resumen . . . . .	I
Abstract . . . . .	III
Sumario . . . . .	IV
Lista de Figuras . . . . .	VII
Lista de Tablas . . . . .	IX
<b>1. Introducción</b>	<b>1</b>
1.1. Consideraciones Iniciales y Motivación . . . . .	1
1.2. Descripción del problema . . . . .	3
1.3. Objetivos de la investigación . . . . .	4
1.3.1. Objetivo Principal . . . . .	4
1.3.2. Objetivos Secundario . . . . .	4
1.4. Metodología . . . . .	4
1.5. Organización del Trabajo . . . . .	5
<b>2. Conceptos Previos</b>	<b>7</b>
2.1. Minería de datos temporales . . . . .	7
2.1.1. Series de tiempo . . . . .	8
2.1.2. Representación de series temporales . . . . .	9
2.1.3. Medidas de similitud . . . . .	14
2.2. Evolución temporal de temas . . . . .	19
2.3. Proyección de datos multi-dimensionales . . . . .	19
2.3.1. Técnicas de proyecciones de datos multidimensionales . . . . .	20
2.4. árboles filo-genéticos . . . . .	20
<b>3. Resultados</b>	<b>21</b>
3.1. Evaluación de Resultados . . . . .	21
3.1.1. Synthetic Control Chart . . . . .	21
3.1.2. Gun Point . . . . .	21
3.1.3. Alessandro Vespignani dataset . . . . .	23
<b>Referencias Bibliográficas</b>	<b>29</b>





# Índice de figuras

---

1.1. Proceso de construcción de mapas de textos evolutivos usando NJ	4
2.1. Jerarquía de diferentes enfoques de representación de series de tiempo	10
2.2. Una tabla de consulta que contiene los puntos de interrupción que dividen una distribución de Gauss en un número arbitrario (3-10) de las regiones equiprobables (Lin et al., 2007)	11
2.3. Una serie de tiempo es discretizado mediante la obtención de una primera aproximación PAA y luego usando los <i>breakpoints</i> predeterminados para asignar los coeficientes de PAA en símbolos SAX. (Lin et al., 2007)	11
2.4. Representación PAA de una serie de tiempo	12
2.5. Una serie de tiempo de longitud 64, denotado por $C$ , se convierte en la representación <i>clipped</i> , denotado por $c$ , simplemente observando los elementos de $C$ , que los puntos encima de cero son 1 caso contrario son 0.	13
2.6. Ejemplo de <i>warping path</i>	16
3.1. Proyección IDMAP con distancia euclidiana	22
3.2. Proyección IDMAP con distancia DTW	22
3.3. Proyección NJ con distancia euclidiana	22
3.4. Proyección NJ con distancia DTW	22
3.5. Proyección LSP con distancia DTW	23
3.6. Proyección NJ con distancia DTW	23
3.7. Proyección temporal LSP usando de dataset colecciones de documentos	24
3.8. Proyección NJ usando de dataset colecciones de documentos	24



# Índice de cuadros

---



---

# Introducción

---

## 1.1. Consideraciones Iniciales y Motivación

En la actualidad la cantidad de datos producida por los medios electrónicos a tenido un alza importante tanto en el número de registros como en complejidad, de tal forma que se ha creado en los últimos dos años mas datos que en toda la historia anterior de la raza humana [Marr \(2015\)](#) este exceso de información es conocido como "sobrecarga de información" como se menciona en ([Paulovich et al., 2008](#)) y esto ocurre cuando una persona no es capaz de localizar o hacer uso de la información que necesita ([Christian et al., 2001](#)), como consecuencia identificar información útil de estos grandes conjuntos de datos se ha vuelto un reto.

Una gran parte de esta aumento en la cantidad de datos viene relacionado a las colecciones de artículos científicos, como se menciona en [Ware and Mabe \(2015\)](#) la publicación colectiva es alrededor 2,5 millones de artículos y su crecimiento a sido del 3% y un 3,5% al año con tendencia al crecimiento, como consecuencia su exploración a despertado el interés de la comunidad científica dada la importancia al momento de iniciar una nueva investigación donde las referencias bibliográficas, las fechas de las investigaciones guardan una vital relación con el tema que se esta investigando ([Sun et al., 2013](#)). A pesar que los motores de bases de datos científicas ayudan en esta labor sus resultados son mostrados en una lista textual ([Sakai et al., 2012](#)) y si bien pueden mostrar un orden cronológico, resultando difícil la exploración conjunta del contenido temático, relaciones referenciales de bibliografía y la jerarquía en cuanto al tiempo que fueron publicadas. Esto genera problemas cuando se van creando con el tiempo nuevas investigaciones ocasionando dificultados en el proceso de

exploración por parte de un investigador.

A razón de estos problemas surgieron técnicas de mapeamiento de texto, que se define como la creación de un mapa de una colección de documentos basándose en el análisis de su contenido y sus similitudes, pasando primero por una etapa de preprocesamiento, proyección de los datos en dos dimensiones seguido de algún método de clustering como se define en [Lopes et al. \(2006\)](#).

En [Börner et al. \(2003\)](#) muestra diferentes técnicas para realizar mapeamiento de texto, para dar solución a dos problemas importantes en el proceso de análisis y visualización de textos [Cuadros et al. \(2007\)](#): Enormes cantidades de datos deben ser mostrados en un espacio limitado y que los datos multidimensionales necesitan ser mostrados en espacios de dos dimensiones. Se presentan entonces técnicas de clustering, posicionamiento basado en fuerza, y reducción de dimensionalidad. Dentro de las técnicas de reducción de dimensionalidad se encuentran las proyecciones multidimensionales que son mas rápidas e intentan preservar el máximo posible las relaciones de similitud entre objetos quedando mapeados en espacios de bajas dimensiones [Cuadros et al. \(2007\)](#).

Como consecuencia se han realizado trabajos que usan proyecciones multidimensionales para facilitar el análisis de estas colecciones de documentos como por ejemplo en los trabajos de ([Valdivia, 2007](#)), ([Paulovich et al., 2008](#)), ([Paulovich et al., 2012](#)).

La técnica de proyección *Neighbor joining* propuesta en ([Saitou and Nei, 1987](#)) para mostrar la evolución de organismos a través de la reconstrucción de árboles filogenéticos y utilizada también en proyección de documentos ([Valdivia, 2007](#)). Esta técnica ha demostrado ser adecuada para el mapeamiento de texto, además de tener ventajas sobre otras técnicas en cuanto a los problemas que surgen en el proceso de visualización pues visualmente la técnica *Neighbor joining* puede partir de una visión local o global del árbol. ([Valdivia, 2007](#)).

Las técnicas de proyección antes mencionadas han demostrado tener un buen desempeño en cuanto al mapeamiento de artículos científicos. Estas colecciones sin embargo en el mayor número de casos tienen propiedades temporales, como en una conferencia indexada anualmente o en el caso de nuevos avances publicados recientemente con referencias a artículos anteriores. A pesar de tener el componente temporal un papel importante en muchos tipos de datos, inclusive en texto, muchas técnicas de proyección multidimensional no lo consideran explícitamente en su proceso de visualización como lo muestra ([Alencar, 2012](#)) en las siguientes técnicas de proyección multidimensionales dinámicas ([Alsakran et al., 2012](#)), ([Cruz, 2012](#)), ([Joia et al., 2011](#)), ([de Pinho et al., 2010](#)).

En ([Alencar, 2012](#)) propone la técnica *Time-based Least Square Projection* que solucionan parcialmente algunas de las desventajas que poseen las proyecciones

dinámicas como la incorporación del componente temporal en el proceso de proyección y además ofrece un buen equilibrio entre la preservación de la información global y de las relaciones de vecindad local como también un costo computacional inferior a otras técnicas. A pesar de tener buenos resultados muestra limitaciones como la susceptibilidad a la última proyección debido a su esquema retroactivo, este modelo está limitado a intervalos de tiempo anuales, además de que un usuario puede tener dificultades en acompañar la evolución de tópicos por medio de los efectos de la animación ([Alencar, 2012](#)).

Los enfoques antes considerados ofrecen una visualización de los datos en este caso colecciones de artículos científicos, donde las similitudes de documentos es por contenido, y tienen la desventaja de no ofrecer una evolución de sus temas a través del tiempo, por ejemplo si un investigador quiere saber que cuáles son las investigaciones más recientes en un campo de estudio, las técnicas antes mencionadas no proporcionarían ayuda, para solucionar esto en ([Alencar, 2007](#)) ofrece un enfoque en donde aborda este problema haciendo uso también de técnicas conocidas como *Temporal text mining* tiene que ver con el descubrimiento de patrones temporales en información de texto a través del tiempo y una tarea en particular que es *Evolutionary theme patterns* ([Mei and Zhai, 2005](#)).

## 1.2. Descripción del problema

Como antes se había mencionado, identificar información útil de conjuntos grandes de datos constituye una ardua tarea. Las colecciones de artículos científicos forman parte de esta creciente producción de datos. Este tipo de datos tienen características propias como son las referencias bibliográficas de trabajos relacionados, la temporalidad (fecha de publicación del artículo científico). Cuando una persona desea iniciar una nueva investigación toda esta carga de información es mostrada de forma textual ([Sakai et al., 2012](#)).

Trabajos como los de ([Paulovich et al., 2008](#)), ([Valdivia, 2007](#)) entre otros, mapean la información, logrando una visión más comprensible de las colecciones de artículos científicos basados en su similitud por contenido. Estas técnicas no consideran el tiempo siendo este atributo importante, como tampoco muestran la evolución temporal de los temas. El enfoque ([Alencar, 2007](#)) donde aborda este problema haciendo uso también de técnicas conocidas como *Temporal text mining* y *Evolutionary theme patterns* muestra también algunas limitaciones que fueron mencionadas anteriormente. Estas desventajas hacen que la exploración de colecciones de artículos científicos sea una tarea difícil, como por ejemplo: como también conocer cuáles son las últimas investigaciones en el estado del arte (íntimamente relacionado con el tiempo), que artículos están relacionados entre sí y su similitud por contenido como por último explorar la

evolución temática en la colección de artículos.

### 1.3. Objetivos de la investigación

#### 1.3.1. Objetivo Principal

Investigar como incorporar el tratamiento del componente temporal en una proyección multidimensional *Neighbor joining* de modo que sea apropiado para tareas de análisis exploratoria envolviendo la evolución de temas en colecciones de artículos científicos.

#### 1.3.2. Objetivos Secundario

1. Obtener una proyección multidimensional con componente temporal basado en *Neighbor joining*.
2. Detectar e incluir una representación visual sobre la evolución de tópicos o temas.

### 1.4. Metodología

La metodología a seguir constará de las siguientes partes como lo muestra a continuación la figura

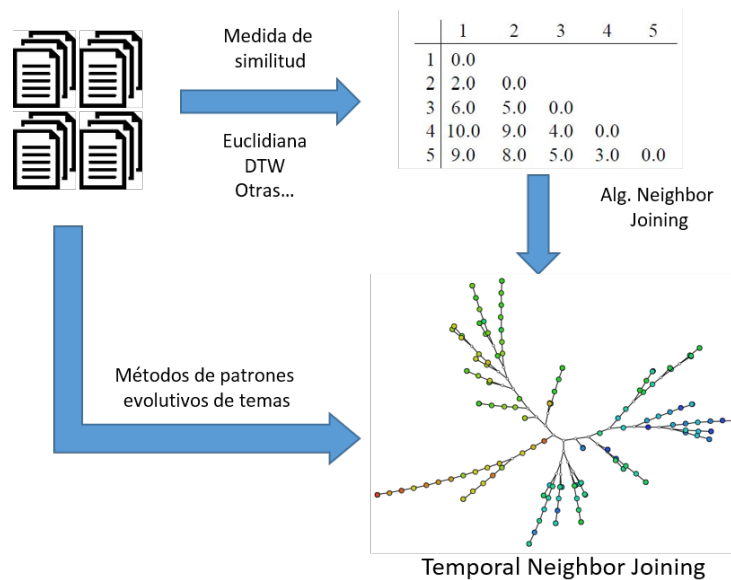


Figura 1.1: Proceso de construcción de mapas de textos evolutivos usando NJ



## 1.5. Organización del Trabajo

Este trabajo está organizado en 5 capítulos, incluyendo esta introducción, y la siguiente estructura:

- En el **Capítulo 2** se presenta los conceptos previos de la investigación, el cual incluye Minería de datos temporales, Evolución temporal de temas, Proyección de datos multidimensionales y Árboles filogenéticos.



---

# Conceptos Previos

---

## 2.1. Minería de datos temporales

Recientemente, el uso creciente de datos temporales, en particular los datos de series de tiempo, ha puesto en marcha varios intentos de investigación y desarrollo en el campo de la minería de datos. Las series temporales son una clase importante de objetos de datos temporales, y se puede obtener fácilmente de las aplicaciones científicas y financieras (por ejemplo, el electrocardiograma (ECG), la temperatura diaria, los totales de ventas semanales, y los precios de los fondos de inversión y acciones). Una serie de tiempo es un conjunto de observaciones realizadas por orden cronológico. La naturaleza de los datos de series de tiempo incluye: datos de gran tamaño, de alta dimensionalidad y actualización de forma continua. series temporales de datos por otra parte, que se caracteriza por su naturaleza numérica y continua, siempre se considera como un todo en lugar de campo numérico individual. Por lo tanto, a diferencia de las bases de datos tradicionales, donde la similitud de búsqueda se basa en coincidencia exacta, búsqueda de similitudes en los datos de series de tiempo se lleva a cabo normalmente de forma aproximada ([Fu, 2011](#)).

La diversidad de dominios es bastante significativo y se extiende desde el ámbito médico al financiero. Algunos ejemplos de estos datos son los siguientes ([Fayyad et al., 2002](#)):

- Datos de sensores: Datos de sensores suele ser recolectada por una amplia variedad de hardware y otros dispositivos de vigilancia. Por lo general, estos datos contienen lecturas continuas sobre el objetos. Por ejemplo, los datos del medio ambiente se recoge comúnmente con diferente tipos de sensores que miden la temperatura, la presión, la humedad, y así sucesivamente.

datos de sensores es la forma mas común de los datos de series de tiempo.

- **Dispositivos médicos:** Muchos dispositivos médicos tales como el electrocardiograma (ECG) y el electroencefalograma (EEG) producen flujos continuos de datos de series de tiempo. estos representan mediciones del funcionamiento del cuerpo humano, tales como el latido del corazón, la frecuencia del pulso, la presión sanguínea, etc. en tiempo real de datos también se obtiene de los pacientes en la unidad de cuidados intensivos (UCI) para supervisar su estado.
- **Datos de los mercados financieros:** datos financieros, tales como precios de las acciones, a menudo es temporal. Otro formas de datos temporales incluyen precios de los productos, las tendencias industriales y económica indicadores.

### 2.1.1. Series de tiempo

Una serie de tiempo es una colección de valores obtenidos de medidas secuenciales a través del tiempo y pueden ser clasificados en tres tipos según (Mitsa, 2010):

- *Time series.* Representan medidas ordenandas de valores reales en intervalos de tiempos regulares.
- *Temporal sequences.* Estas pueden ser marcas de tiempo en intervalos regulares o irregulares. Un ejemplo de una secuencia temporal es una secuencia de marcas de tiempo de las compras de un cliente en un sitio Web.
- *Semantic temporal data.* Se definen en el contexto de una ontología. Por ejemplo, "Señor" "mediana edad." están definidos en el contexto de una definición formal de tipos, propiedades, y relaciones entre entidades que realmente o fundamentalmente existen para un dominio de discusión, en particular de la vida humana.

Los datos de series de tiempo según su naturaleza pueden ser univariante o multivariante. En los datos de series temporales univariantes, un atributo de comportamiento individual está asociado a cada instante de tiempo. En series temporales de datos multivariados, múltiples atributos de comportamiento estan asociados a cada instante de tiempo. La dimensionalidad de la serie de tiempo, por lo tanto, se refiere al número de atributos de comportamiento que están siendo rastreados (Aggarwal, 2015) y se pueden definir formalmente de la siguiente forma:

**Definición 2.1** (Serie de tiempo univariante) Una serie de tiempo  $T$  es una secuencia ordenada de  $n$  variables de valores reales.

$$T = (t_1, \dots, t_n), \quad t_i \in \mathbb{R}$$

**Definición 2.2** (Serie de tiempo Multivariante) una serie de tiempo de longitud  $n$  y dimensionalidad  $d$ , contiene  $d$  características numéricas en cada una de las  $n$  marcas de tiempo  $t_1, \dots, t_n$ . Por lo tanto, el conjunto de valores recibidos en una marca de tiempo  $t_i$  es  $\bar{Y}_i = (y_i^1, \dots, y_i^d)$ . El valor de el  $j$ th series de una marca de tiempo  $t_i$  es  $y_j^i$ .

### 2.1.2. Representación de series temporales

Una de las características de las series de tiempo es su alta dimensionalidad, lo que generalmente ocasiona problemas en el procesamiento como consecuencia a su alto costo computacional, debido a esto se necesita aplicar métodos de reducción de dimensión (es decir reducir el número de puntos) de los datos originales, los beneficios ganados son el fácil almacenamiento, rapidez en el procesamiento y eliminación de ruido (Esling and Agon, 2012), una de las desventajas también es que dependiendo el método empleado se tendrá una pérdida de información según cuanto de dimensión se quiera reducir. Existen muchas técnicas para la representación de series temporales y es posible clasificar esos enfoques según la transformación aplicada, Tanto en (Aghabozorgi et al., 2015a) como también en (Lovric, 2011) muestra una taxonomía para dividir las representaciones de series de tiempo en cuatro categorías como lo muestra en la figura 2.1

**Definición 2.3** (Representación de series de tiempo) dado una serie de tiempo  $F_i = f_1, \dots, f_t, \dots, f_T$ , la representación es la transformación de la serie de tiempo a un vector de dimensionalidad reducida  $F_i = f_i, \dots, f_x$  donde  $x < T$  y si dos series son similares en su espacio original entonces sus representaciones deberían ser similares en el espacio de transformación también.

#### Data adaptive

Este enfoque implica que el parámetro de transformación se ira modificando dependiendo de la naturaleza de los datos disponibles, en otras palabras usan una longitud no igual de segmentación para la representación de series de tiempo.

##### *Symbolic Aggregate Approximation (SAX)*

Este método de representación permite una reducción de dimensionalidad de una longitud  $n$  a otra cadena de caracteres de longitud  $w$  donde se entiende

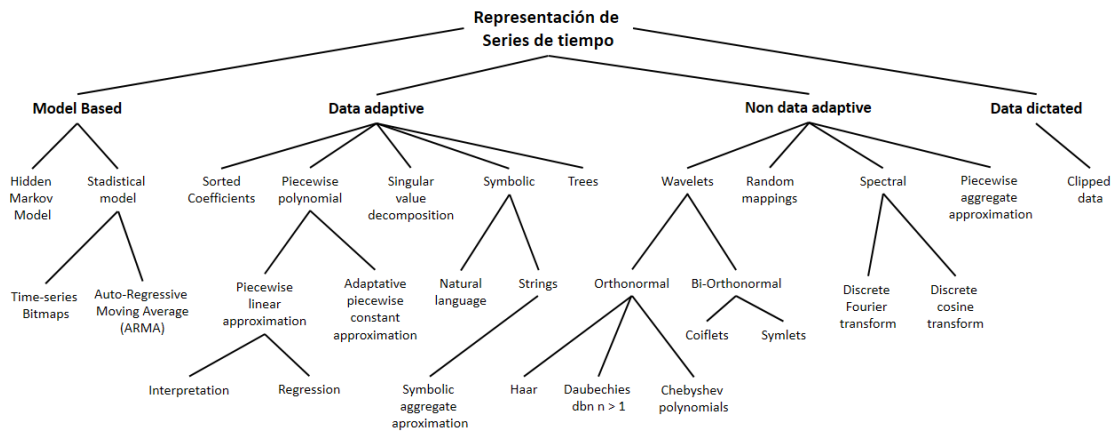


Figura 2.1: Jerarquía de diferentes enfoques de representación de series de tiempo

que siempre  $w$  será menor que  $n$ , este proceso de discretización usa como paso intermedio el algoritmo de representación Piecewise Aggregate Approximation (PAA) para luego simbolizar esta representación en una cadena discreta. Dado que una serie de tiempo tiene una distribución Gaussiana (RJaM, 2000) que es la base del algoritmo SAX, se puede determinar "breakpoints" que producirán áreas  $a$  de igual tamaño bajo la curva de Gaussiana (RJaM, 2000), el algoritmo hace uso de los siguientes conceptos:

- *Breakpoints*: los *Breakpoints* áreas de igual tamaño que dividen una distribución Gaussiana para ser aplicadas en SAX estos valores ya están determinados en el siguiente (cuadro 2.2).
- *Word*: representan los símbolos que estarán en cada intervalo del *Breakpoint*.

En la (figura 2.3) se muestra un ejemplo gráfico de como funciona SAX.

### Non-data adaptive

los parámetros de la transformación permanecen siendo los mismos para toda la serie de tiempo ignorando la naturaleza de la misma, en *Non-data adaptive* la longitud de la segmentación es de igual longitud, algunos de estos métodos son:

#### *Piecewise Aggregate Approximation (PAA)*

En este método, las series de tiempo se divide en segmentos  $k$  de igual longitud y luego cada segmento se sustituye con un valor constante, que es el valor medio del segmento. A continuación, estos valores medios se agrupan en un vector, que representa la marca del segmento (Mitsa, 2010). Un ejemplo se muestra en la (Figura 2.4).

$\beta_i \backslash a$	3	4	5	6	7	8	9	10
$\beta_1$	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28
$\beta_2$	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84
$\beta_3$		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52
$\beta_4$			0.84	0.43	0.18	0	-0.14	-0.25
$\beta_5$				0.97	0.57	0.32	0.14	0
$\beta_6$					1.07	0.67	0.43	0.25
$\beta_7$						1.15	0.76	0.52
$\beta_8$							1.22	0.84
$\beta_9$								1.28

Figura 2.2: Una tabla de consulta que contiene los puntos de interrupción que dividen una distribución de Gauss en un número arbitrario (3-10) de las regiones equiprobables (Lin et al., 2007)

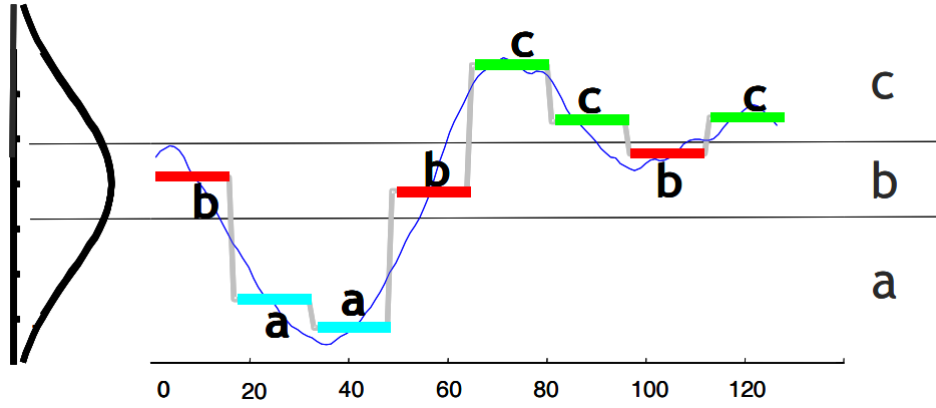


Figura 2.3: Una serie de tiempo es discretizado mediante la obtención de una primera aproximación PAA y luego usando los *breakpoints* predeterminados para asignar los coeficientes de PAA en símbolos SAX. (Lin et al., 2007)

Una definición más formal de PAA se da a continuación, se tiene una serie de tiempo  $C$  de longitud  $n$  puede ser representado en un espacio de dimensión  $w$  por un vector  $\bar{C} = \bar{c}_1, \dots, \bar{c}_w$ . el elemento  $i^{th}$  de  $\bar{C}$  es calculado por la siguiente ecuación:

$$\bar{c}_i = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} c_j \quad (2.1)$$

En pocas palabras, para reducir la serie de tiempo de  $n$  dimensiones a  $w$  dimensiones, los datos se divide en  $w$  "marcos" de igual tamaño. El valor medio de los datos incluidos en una trama se calcula y un vector de estos valores se convierte en la representación reducida de datos (Lin et al., 2007).

*Discrete Fourier Transform (DFT)*

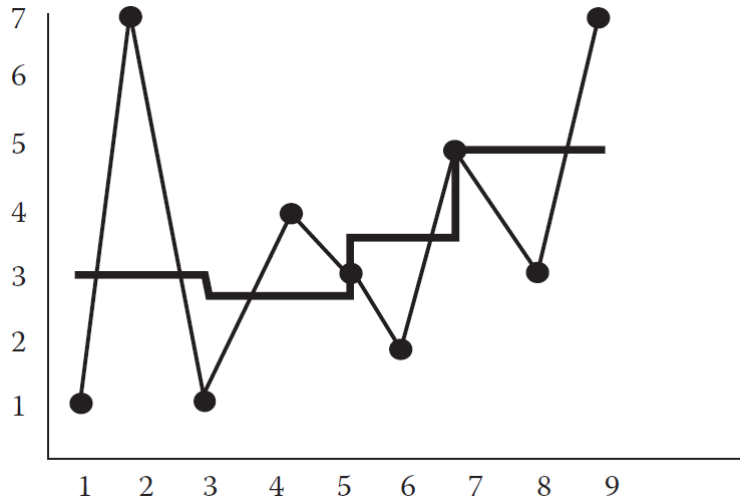


Figura 2.4: Representación PAA de una serie de tiempo

Wavelets son más efectivos cuando la mayoría de variación in la serie puede ser capturado en una región específica local de la serie. En casos donde la serie contiene periodicidad global, el DFT es más efectivo ([Aggarwal, 2015](#)). El DFT representa una serie de tiempo en el dominio frecuencia, el coeficiente  $F_k$  de una serie de tiempo  $X = x_0, x_2, \dots, x_{n-1}$  es un número complejo dado por:

$$F_k = \sum_{i=0}^{N-1} x_i e^{-j2\pi i k / N} \quad (2.2)$$

Donde:  $k = 0, 1, \dots, N-1$ .

Una de las ventajas de usar DFT en procesamiento de señales es que existe un algoritmo rápido para su calculo, conocido como *fast fourier transform* (FFT) su complejidad computacional es  $O(n \log n)$ .

### Model-based

Representan una series de tiempo de una forma estocástica, en estadística es un concepto matemático que sirve para tratar con magnitudes aleatorias o más exactamente para caracterizar una sucesión de variables aleatorias(estocásticas) que evolucionan en función de otra variable, generalmente el tiempo.

#### *Markov Models Representation*

modelos de Markov representan una serie de tiempo de una manera estocástica. *Hidden Markov models* (HMM) son modelos de Markov cuyos parámetros son desconocidos. Una descripción detallada de los HMM se puede encontrar en ([Rabiner, 1989](#)). Un HMM de primer orden se describe completamente con los siguientes parámetros:

- El número de estados.

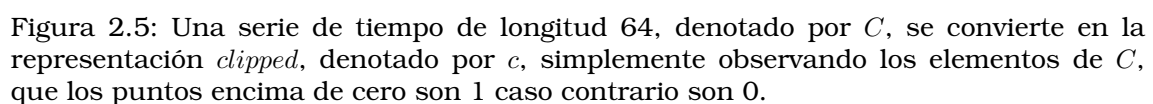


- Para que un *Hidden Markov models* sea totalmente modelada probabilística-mente, tenemos que especificar los estados anteriores y actuales.

algoritmos iterativos, tales como el algoritmo de Baum-Welch, sólo se garantizan la convergencia a un máximo local. Los parámetros que se estiman por este algoritmo son la probabilidad del estado inicial, los parámetros de transición de estado, la media y la covarianza de la gaussiana de cada estado.

En este enfoque la tasa de comprensión es definido automáticamente de una serie de tiempo sin procesar (Aghabozorgi et al., 2015a) tal como Clipped propuesto en (Ratanamahatana et al., 2005).

En (Ratanamahatana et al., 2005) presenta el método de representación que trabaja remplazando cada dato del valor real por un único bit. como se muestra en la siguiente (figura 2.5)



Más formalmente, podemos definir  $c$ , la representación *clipped* de  $C$  como:

$$c(i) = \begin{cases} 1 & \text{if } C(i) > \mu \\ 0 & \text{otherwise} \end{cases} \quad (2.3)$$

Donde  $\mu$  es el valor medio de  $C$ , y en el caso que la serie de tiempo este normalizada se puede asumir que  $\mu = 0$ .

### 2.1.3. Medidas de similitud

Las medidas de similitud son una tarea importante y además la base de donde surgen técnicas más sofisticadas como la minería y análisis de series de tiempo, a diferencia del clustering tradicional donde una distancia entre objetos estáticos es una comparación exacta, en clustering en series de tiempo la distancia es calculada de manera aproximada. En el dominio de series de tiempo, la elaboración de una función de similitud apropiada no se hace de forma trivial. Hay esencialmente dos maneras en que los datos podrían ser organizados y procesados (Agrawal et al., 1993), en *whole series matching* donde se considera la longitud completa de toda la serie de tiempo durante la búsqueda de similitud. Se requiere la comparación de la secuencia de consulta para cada serie candidata mediante la evaluación de una función de distancia y hacer el seguimiento de la secuencia con la distancia más pequeña (Fu, 2011). Este problema de similitud puede ser resumido en dos formas (Mitsa, 2010) de la siguiente manera:

1. Encontrar todos los pares de series de tiempo que tienen una distancia que es menor que un número  $n$ .
2. Indexación y búsqueda de contenidos. Hay dos enfoques para este tipo de búsqueda: (1) por rango: Encuentre todas las series temporales que tienen distancia inferior a un número  $n$  a partir de una serie de tiempo específica. (2) Busca los  $m$  vecinos más cercanos para una serie de tiempo específico.

la otra manera es *subsequence matching* donde una secuencia corta es comparada con una secuencia más larga, deslizando el anterior a lo largo de este ultimo. Una forma para calcular la distancia entre dos series de tiempo es considerándolos como series de tiempo univariadas, una serie de tiempo univariada es cuando solo tiene una dimensión que depende del tiempo, y luego calculando la medida de distancia a través de todos los puntos.

La elección de una medida de distancia depende netamente de las características de la serie de tiempo como son: la longitud, el método de representación y el objetivo del clustering (Aghabozorgi et al., 2015a), las

medidas de distancia o similitud pueden ser divididas en cuatro categorías (Esling and Agon, 2012) que son las siguientes:

Shape-based:

compara en general la forma de las series de tiempo, los métodos basados en esta categoría son:

*Dynamic Time Warping (DTW)*

Cuando se hace desea hacer una medida de similitud generalmente se asume que las dos series de tiempo que se quiere evaluar están alineadas en el eje-X, para solucionar este problema esta el algoritmos Dynamic Time Warping que consiste algorítmicamente como se explica en (Keogh and Pazzani, 2001):

Tenemos dos series de tiempo  $Q$  Y  $C$  de longitud  $n$  y  $m$  respectivamente donde:

$$Q = q_1, q_2, \dots, q_i, \dots, q_n \quad (2.4)$$

$$C = c_1, c_2, \dots, c_j, \dots, c_m \quad (2.5)$$

para alinear estas dos secuencias usando DTW se construye un matriz  $n$  por  $m$  donde el elemento  $(i^{th}, j^{th})$  de la matriz contiene las distancias  $d(q_i, c_j)$  entre los dos puntos  $q_i$  y  $c_j$  (comúnmente se usa la distancia euclidiana así que  $d(q_i, c_j) = (q_i - c_j)^2$ ). cada elemento de la matriz  $(i, j)$  corresponde a la alineación entre los puntos  $q_i$  y  $c_j$ . Esto se ilustra en la (figura 2.6). Un Warping path  $W$  es un conjunto de elementos de la matriz que define un mapeo entre  $Q$  Y  $C$ . El elemento  $k^{th}$  de  $W$  es definido como  $w_k = (i, j)_k$  así que tenemos:

$$W = w_1, w_2, \dots, w_k, \dots, w_k \quad \max(m, n) \leq k < m + n + 1 \quad (2.6)$$

El path warping esta sujeto a varias restricciones como son, condiciones de limites, continuidad, nonotonicidad, que están hechas para optimizar el rendimiento de su calculo.

*Spatial Assembling (SpADe)*

En este algoritmo, la similaridad es calculado por búsqueda de patrones coincidentes entre dos series de tiempo, El algoritmo (Chen et al., 2007) describe que teniendo dos secuencias  $Q[0 : m]$  y  $D[0 : n]$  de las cuales se extrae un conjunto de pequeños patrones de la serie de tiempo usando una ventaja deslizando de tamaño fijo. Esos pequeños patrones de una misma longitud son llamados patrones locales. Al utilizar el tamaño fijo de ventana deslizando en dos secuencias de tiempo, se obtienen dos conjuntos de patrones locales.

Un patrón local  $lp = (\theta_{pos}, \theta_{amp}, \theta_{shp}, \theta_{tscl}, \theta_{ascl})$ , que son las que son la posición de  $lp$  en  $Q$  que significan:

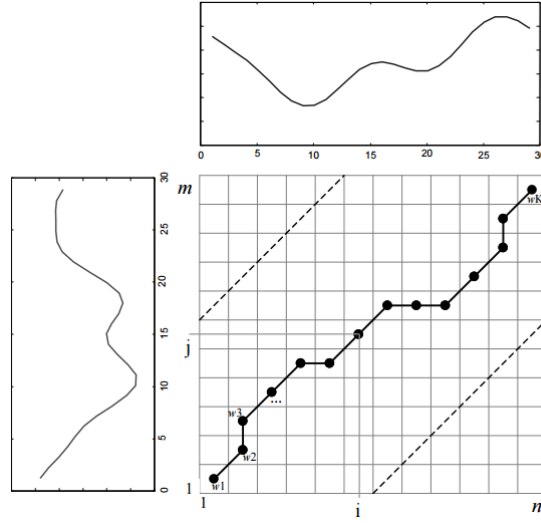


Figura 2.6: Ejemplo de *warping path*

- $\theta_{pos}$ : Es la posición de  $lp$  en  $Q$ .
- $\theta_{amp}$ : Es la amplitud media de los datos item en  $lp$ .
- $\theta_{shp}$ : La forma que caracteriza de  $lp$ .
- $\theta_{tscl}$ : La escala temporal de  $lp$  (igual a 1 si  $Q$  no esta escalado).
- $\theta_{ascl}$ : La escala de amplitud de  $lp$  (igual a 1 si  $Q$  no esta escalado).

La distancia de los dos patrones locales  $lp$  en  $Q$  y  $lp'$  en  $D$ , pueden ser medidos como:

$$D_1(lp', lp) = f(|\theta'_{amp} - \theta_{amp}|, |\theta'_{shp} - \theta_{shp}|) \quad (2.7)$$

Que es una suma ponderada de las diferencias de las amplitudes y características de forma de los dos patrones locales, finalmente se procede a comparar estos patrones para hallar el grado de similitud.

#### *DISSIM Distance*

Ha sido introducido para manejar la similitud de varias tasas de sampling. Esto es definido como una aproximación de la integral de la distancia euclidiana que también se puede definir (Frentzos et al., 2007) como: La disimilaridad  $DISSIM(Q, T)$  entre dos trayectorias  $Q$  y  $T$  que ha sido validado durante un periodo  $[t_1, t_n]$  se expresa mediante la siguiente ecuación:

$$DISSIM(Q, T) \approx \frac{1}{2} \sum_{k=1}^{n-1} ((D_{Q,T}(t_k) + D_{Q,T}(t_{k+1})) \cdot (t_{k+1} - t_k)) \quad (2.8)$$

Donde  $D_{Q,T}(t)$  es una función de la distancia euclidiana entre las trayectorias  $Q$  y  $T$  con el tiempo.

Edit-based:

compara dos series de tiempo en base al mínimo número de operaciones necesarias para transformar una serie a la otra.

*Edit Distance with Real Penalty (ERP)*

La distancia ERP es una medida de distancia elástica para *matching* en series de tiempo (Chen and Ng, 2004). Durante el calculo de la distancia ERP de dos series de tiempo  $R$  y  $S$  con longitudes  $M$  y  $N$ , están alineados a la misma longitud mediante la adición de algunos símbolos (también llamados *gaps*) a ellos. A continuación, cada elemento en una serie de tiempo o bien se corresponde con un *gap* o un elemento en la otra serie de tiempo. finalmente la distancia ERP se define (Chen, 2005), dado dos series de tiempo  $R$  y  $S$  de longitud  $M$  y  $N$ , respectivamente la distancia ERP de  $R$  a  $S$ ,  $ERP(R, S)$ , se define como:

$$ERP(R, S) = \begin{cases} \sum_{i=1}^M |r_i - g| & \text{if } N = 0, \\ \sum_{i=1}^N |s_i - g| & \text{if } M = 0, \\ \min \begin{cases} ERP(Rest(R), Rest(S)) + dist_{erp}(r_1, s_1), \\ ERP(Rest(R), S) + dist_{erp}(r_1, g), \\ ERP(R, Rest(S)) + dist_{erp}(s_1, g) \end{cases} & \text{otherwise} \end{cases} \quad (2.9)$$

Donde:

$$dist_{erp}(r_i, s_i) = \begin{cases} |r_i - s_i| & \text{if } r_i, s_i \text{ not gaps} \\ |r_i - s_g| & \text{if } s_i \text{ is a gaps} \\ |s_i - s_g| & \text{if } r_i \text{ is a gaps} \end{cases} \quad (2.10)$$

y  $g$  denota los *gap* que se asigna un valor constante.

*The Longest Common Subsequence (LCSS)*

LCSS es una medida que es tolerante a la ausencia de puntos en la serie de tiempo(gaps) en las dos series que se quiere comparar, este algoritmo asume la misma base y escala para las dos series de tiempo y es superior a DTW en las siguientes aspectos (Vlachos et al., 2003):

- LCSS maneja mejor el ruido y los *outliers*.
- El DTW puede distorsionar la distancia actual entre puntos en la serie de tiempo por *over fitting*.
- La complejidad computacional de DTW es significativa y además su escalabilidad no es muy buena.

Una definición formal de LCSS la encontramos en (cs.umd, 1998) y explica que teniendo dos secuencias de caracteres  $X = x_1, x_2, \dots, x_m$  y  $Z = z_1, z_2, \dots, z_k$

decimos que  $Z$  es una subsecuencia de  $X$  si existe una secuencia estrictamente incremental de  $k$  índices  $i_1, i_2, \dots, i_k$  tal que  $Z = X_{i_1}, X_{i_2}, \dots, X_{i_k}$ , por ejemplo, tenemos  $X = ABRACADABRA$  y  $Z = AADAA$ , luego  $Z$  es una subsecuencia de  $X$ .

Feature-based:

Se extraen las características describiendo aspectos de la serie que son luego comparadas con algún tipo de función de distancia, entre los cuales tenemos los siguientes métodos:

#### *Autocorrelation Function*

Como se explica en (Baggenstoss, 2008), la función de autocorrelación (ACF) es una muy importante conjunto de características para el análisis de series de tiempo. Teniendo un conjunto de series de tiempo  $X = [x_1, x_2, \dots, x_N]$  de longitud  $N$  definimos un conjunto de características  $Z$  de  $M$ -dimensiones como también un conjunto arbitrario de retardos (*lags*)  $k_1, k_2, \dots, k_M$ . Tenemos  $Z = [r_{k_1}, r_{k_2}, \dots, r_{k_M}]$ , donde:

$$r_k = \frac{1}{N} \sum_{i=1}^N x_i x_{[i+k]_N} \quad (2.11)$$

Donde los corchetes  $[i - k]_N$  indican el modulo- $N$ . Estos son conocidos como estimaciones ACF circular debido a la indexación de módulo- $N$ . Se elige esta forma de ACF porque simplifica el análisis. Finalmente se tiene el calculo de características como:

$$\text{Feature Calculation : } Z = [r_{k_1}, r_{k_2}, \dots, r_{k_M}], \text{ Donde } r_k = \frac{1}{N} \sum_{i=1}^N x_i x_{[i+k]_N} \quad (2.12)$$

Notar que es posible reescribir el calculo de las características como:

$$r_k = \frac{1}{N^2} \sum_{i=0}^{N/2} \epsilon_i y_i \cos \left\{ \frac{2\pi i k}{N} \right\}, \quad k = 0, 1, \dots, P, \quad (2.13)$$

Donde  $y = y_0, y_1, \dots, y_{N/2}$  son los coeficientes DFT de magnitud al cuadrado,  $\epsilon_i = 1$  para  $i = 0, N/2$ , y  $\epsilon_i = 2$  para  $i = 1, 2, \dots, N/2 - 1$ .

#### *Histogram distance*

Dado un conjunto de series de tiempo  $D = R_1, R_2, \dots, R_L$ , donde cada elemento  $R$  esta representado por  $R = [(r_1, t_1), \dots, (r_N, t_N)]$  donde  $N$  es el numero de puntos en  $R_i$  y cada par de datos  $(r, t)$  significa tanto  $r$  el valor en la marca de tiempo  $t$ . Para obtener el histograma del conjunto de datos  $D$  se normaliza cada  $R_i$  según la formula mostrada en (Chen et al., 2005), luego los Histogramas son

obtenidos dando un máximo ( $max_D$ ) y un mínimo ( $min_D$ ), el rango  $[min_D, max_D]$  es dividido en  $\tau$  subregiones de igual tamaño disjuntos, llamados *histogram bins*. Dado una serie de tiempo  $R$ , su histograma  $H_R$  puede ser calculado por conteo de número de puntos  $h_i (1 \leq i \leq \tau)$  que son ubicaciones en cada *histogram bin*  $i$ :  $H_R = [h_1, \dots, h_\tau]$ . Finalmente L1-norm o L2-norm (Swain and Ballard, 1991) pueden ser usados para medir la distancia entre dos histogramas.

Structure-based:

Dividimos esta categoría en dos subcategorías específicas. *Model-based distances* que trabaja mediante el ajuste de un modelo para las diferentes series y, luego comparando los parámetros de los modelos subyacentes. *Compression-based distances*. analizar qué tan bien dos series se puede comprimir juntos. Similitud es reflejada por relaciones de compresión más altas.

*Compression-based dissimilarity (CDM)*

Dado dos cadena de características,  $x$  y  $y$ , definimos el CDM como (Keogh et al., 2007):

$$CDM(x, y) = \frac{C(xy)}{C(x) + C(y)} \quad (2.14)$$

La disimilitud CDM se aproxima a 1 cuando  $x$  e  $y$  no están relacionados, y es más pequeño que 1 si  $x$  e  $y$  están relacionados. Cuanto menor sea el CDM  $x$  e  $y$  son más estrechamente relacionados aunque nunca será cero. El algoritmo comprime tanto la cadena  $x$  como la cadena  $y$ , luego concatena las cadenas originales obteniendo  $xy$  procediendo a comprimir, para aplicar finalmente la formula antes mencionada.

## 2.2. Evolución temporal de temas

## 2.3. Proyección de datos multi-dimensionales

Debido al incremento de datos, no solo en el número de registros, sino también en las dimensiones que poseen, como por ejemplo un vector de características de un documento, el vector característica estará conformado por el número de ocurrencias de cada palabra que contiene de tal forma si tenemos  $m$  atributos se tendrá un vector  $m$  dimensional.

Métodos convencionales de visualización de datos fallan cuando son aplicados directamente sobre datos de alta dimensionalidad como en el caso de identificación de patrones (Berkhin, 2006).

Una forma de manejar esta alta dimensionalidad de forma que puedan ser visualizadas de forma correcta son las *proyecciones multidimensionales*, estas técnicas permiten reducir la dimensionalidad de un espacio original  $m$  a uno espacio  $p$ -dimensional donde  $p \ll m$  pudiendo ser las dimensiones de  $p$ : 1, 2, 3,

ademas logran conservar en lo mas posible las relaciones de distancia del espacio original

**Definición 2.4** (Proyección Multidimensional ([Tejada-Gamero et al., 2003](#))) Sea  $X$  un conjunto de objetos en  $\mathbb{R}^m$  con  $\delta : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$  un criterio de proximidad entre objetos en  $\mathbb{R}^m$ , y  $Y$  un conjunto de puntos en  $\mathbb{R}^p$  para  $p = 1, 2, 3$  y  $d : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}$  un criterio de proximidad en  $\mathbb{R}^p$ . Una técnica de proyección multidimensional puede ser descrita como una función  $f : X \rightarrow Y$  cuyo objetivo es hacer  $|\delta(x_i, x_j) - d(f(x_i), f(x_j))|$  o mas proximo posible a cero,  $\forall x_i, x_j \in X$ .

### 2.3.1. Técnicas de proyecciones de datos multidimensionales

Existen variedad de técnicas aplicadas a diferentes campos para la realizar proyecciones multidimensionales, según ([Tejada-Gamero et al., 2003](#)) se pueden clasificar en tres grandes grupos: (1) *Force-Direct Placement(FDP)*; (2) *Multidimensional Scaling(MDS)*; y (3) *Técnicas para reducción de dimensionalidad*.

## 2.4. árboles filo-genéticos



---

# Resultados

---

## 3.1. Evaluación de Resultados

Los siguientes resultados fueron evaluaciones de métodos de proyección usando una parte de la herramienta (Paulovich et al., 2007), haciendo también uso de conjuntos de datos temporales. Se contrastaron los métodos de proyección con mediciones de distancia o similitud comunes como son la distancia euclidiana y específicas para series temporales como son *Dynamic time warping*.

### 3.1.1. Synthetic Control Chart

Este conjunto de datos temporales fueron extraídos de (Chen et al., 2015), y consta de 6 clases: *Normal*, *Cyclic*, *Increasing trend*, *Decreasing trend*, *Upward shift* y *Downward shift* con 600 muestras de 60 dimensiones cada serie de tiempo. Las figuras 3.1, 3.2 muestran las propuestas hechas en (Alencar, 2007), donde se hace uso del método de proyección IDMAP con datos temporales usando distancia euclidiana y el *dynamic time warping* respectivamente.

En las figuras 3.3, 3.4, se observa el comportamiento del método de proyección *Neighbours joining* tanto con distancia euclidiana como *dynamic time warping*.

### 3.1.2. Gun Point

Disponibles también en (Chen et al., 2015), contiene 2 clases, con un total de 200 registros y una longitud de series de tiempo de 150. Este conjunto de datos posee como se había mencionado antes posee dos clases las cuales son *Point* y *Gun*, siendo la principal diferencia que en una el personaje apunta con su

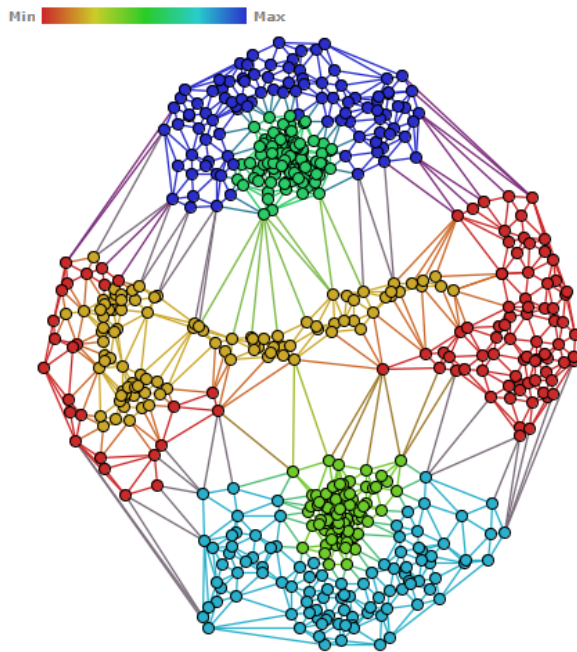


Figura 3.1: Proyección IDMAP con distancia euclidiana

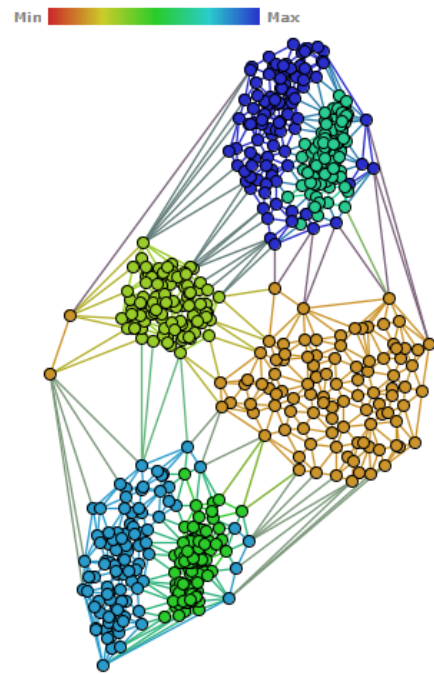


Figura 3.2: Proyección IDMAP con distancia DTW

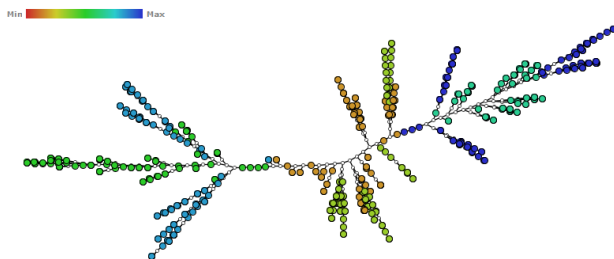


Figura 3.3: Proyección NJ con distancia euclidiana

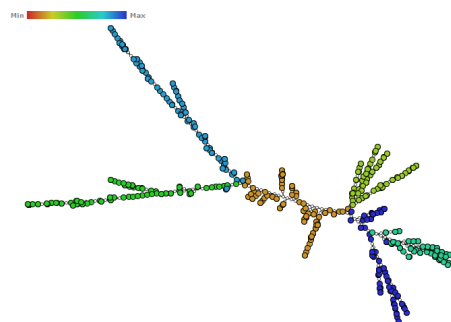


Figura 3.4: Proyección NJ con distancia DTW

Min  Max

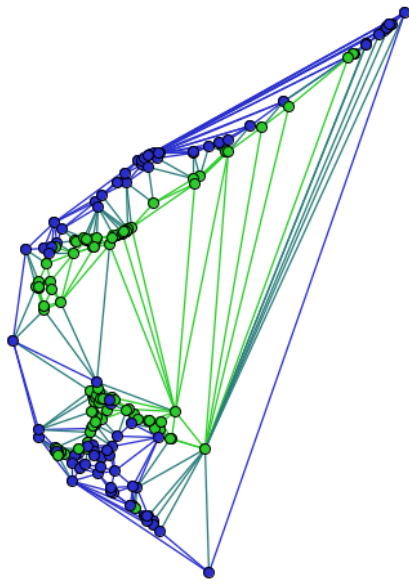


Figura 3.5: Proyección LSP con distancia DTW

Min  Max

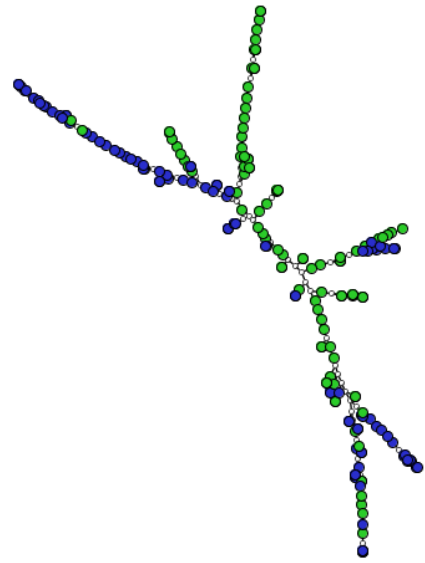


Figura 3.6: Proyección NJ con distancia DTW

dedo mientras en el otro grupo con un arma. Este conjunto es de difícil análisis pues las dos clases no presentan grandes diferencias. En la figuras 3.5, 3.6 se muestran las proyecciones LSP y NJ usando la distancia DTW.

### 3.1.3. Alessandro Vespignani dataset

Es un conjunto de resúmenes de artículos relacionados que tienen como autor a Alessandro Vespignami entre los años 1990 al 2012. Cuenta con información como año de publicación referencias, resumen etc. Este conjunto de datos a sido usado en (Alencar, 2012). La figura 3.7 muestra los resultados obtenidos por (Alencar, 2012), mientras en la figura 3.8 se utilizo esta misma base de datos usando la proyección *Neighbours joining*. En 3.7 los puntos son artículos científicos, los enlaces son las referencias entre artículos, y el color esta dado por la fecha de publicación de cada articulo. En la figura 3.8 se pierde las referencias bibliográficas, pero en su modelo de árbol permite ver la relación local entre documentos, aun así el año de publicación aun se muestra de manera desordenada.

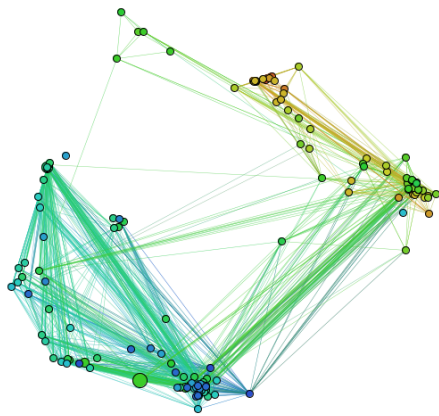


Figura 3.7: Proyección temporal LSP usando de dataset colecciones de documentos

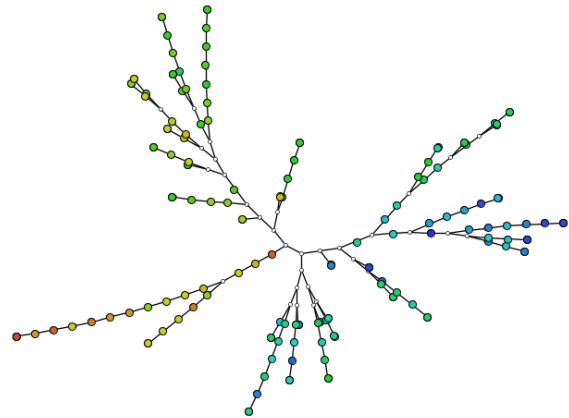


Figura 3.8: Proyección NJ usando de dataset colecciones de documentos

# Bibliografia

---

- Aggarwal, C. C. (2015). *Data Mining: The Textbook*, chapter Mining Time Series Data, pages 457–491. Springer International Publishing, Cham.
- Aghabozorgi, S., Shirkhorshidi, A. S., and Wah, T. Y. (2015a). Time-series clustering—a decade review. *Information Systems*, 53:16–38.
- Aghabozorgi, S., Shirkhorshidi, A. S., and Wah, T. Y. (2015b). Time-series clustering—a decade review. *Information Systems*, 53:16–38.
- Agrawal, R., Faloutsos, C., and Swami, A. (1993). *Efficient similarity search in sequence databases*. Springer.
- Alencar, A. B. (2007). *Mineração e visualização de coleções de séries temporais*. PhD thesis, Instituto de Ciências Matemáticas e de Computação.
- Alencar, A. B. (2012). *Visualização da evolução temporal de coleções de artigos científicos*. PhD thesis, Universidade de São Paulo.
- Alsakran, J., Chen, Y., Luo, D., Zhao, Y., Yang, J., Dou, W., Liu, S., et al. (2012). Real-time visualization of streaming text with a force-based dynamic system. *IEEE Computer Graphics and Applications*, 32(1):34–45.
- Baggenstoss, P. (2008). Autocorrelation function (circular), spa solution, arbitrary lags. In <http://class-specific.com/csf/html/doc/node141.html>.
- Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data*, pages 25–71. Springer.
- Börner, K., Chen, C., and Boyack, K. W. (2003). Visualizing knowledge domains. *Annual review of information science and technology*, 37(1):179–255.
- Chen, L. (2005). *Similarity search over time series and trajectory data*. PhD thesis, University of Waterloo.

- Chen, L. and Ng, R. (2004). On the marriage of lp-norms and edit distance. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 792–803. VLDB Endowment.
- Chen, L., Ozsu, M. T., and Oria, V. (2005). Using multi-scale histograms to answer pattern existence and shape match queries over time series data. In *Proceedings of 17th Int Conf on Scientific and Statistical Database Management. Piscataway: IEEE Press*, pages 217–226.
- Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., and Batista, G. (2015). The ucr time series classification archive. [www.cs.ucr.edu/~eamonn/time\\_series\\_data/](http://www.cs.ucr.edu/~eamonn/time_series_data/).
- Chen, Y., Nascimento, M. A., Ooi, B. C., and Tung, A. K. (2007). Spade: On shape-based pattern detection in streaming time series. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 786–795. IEEE.
- Chiş, M., Banerjee, S., and Hassanien, A. E. (2009). Clustering time series data: an evolutionary approach. In *Foundations of Computational, Intelligence Volume 6*, pages 193–207. Springer.
- Christian, G. R., Blumenthal, C., and Patterson, M. (2001). The information explosion and the adult learner: Implications for reference librarians. *The Reference Librarian*, 33(69-70):19–30.
- Cruz, C. J. D. W. (2012). *Explorando conjuntos de dados volumétricos multidimensionais variantes no tempo usando projeções*. PhD thesis, Universidade de São Paulo.
- cs.umd (1998). Lecture 25: Longest common subsequence. In *Lecture 25*. CLR.
- Cuadros, A. M., Paulovich, F. V., Minghim, R., and Telles, G. P. (2007). Point placement by phylogenetic trees and its application to visual analysis of document collections. In *IEEE VAST*, pages 99–106.
- de Pinho, R. D., de Oliveira, M. C. F., and de Andrade Lopes, A. (2010). An incremental space to visualize dynamic data sets. *Multimedia Tools and Applications*, 50(3):533–562.
- Esling, P. and Agon, C. (2012). Time-series data mining. *ACM Computing Surveys (CSUR)*, 45(1):12.
- Fayyad, U. M., Wierse, A., and Grinstein, G. G. (2002). *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann.

- Frentzos, E., Gratsias, K., and Theodoridis, Y. (2007). Index-based most similar trajectory search. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 816–825. IEEE.
- Fu, T.-c. (2011). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181.
- Joia, P., Coimbra, D., Cuminato, J. A., Paulovich, F. V., and Nonato, L. G. (2011). Local affine multidimensional projection. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2563–2571.
- Keogh, E., Lonardi, S., Ratanamahatana, C. A., Wei, L., Lee, S.-H., and Handley, J. (2007). Compression-based data mining of sequential data. *Data Mining and Knowledge Discovery*, 14(1):99–129.
- Keogh, E. J. and Pazzani, M. J. (2001). Derivative dynamic time warping. In *Sdm*, volume 1, pages 5–7. SIAM.
- Liao, T. W. (2005). Clustering of time series data a survey. *Pattern recognition*, 38(11):1857–1874.
- Lin, J., Keogh, E., Lonardi, S., Lankford, J. P., and Nystrom, D. M. (2004). Visually mining and monitoring massive time series. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 460–469. ACM.
- Lin, J., Keogh, E., Wei, L., and Lonardi, S. (2007). Experiencing sax: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15(2):107–144.
- Lopes, A. d. A., Minghim, R., Melo, V., and Paulovich, F. V. (2006). Mapping texts through dimensionality reduction and visualization techniques for interactive exploration of document collections. In *Proceedings of SPIE*, volume 6060, pages 271–282.
- Lovric, M. (2011). *International encyclopedia of statistical science*. Springer London.
- Marr, B. (2015). Big data: 20 mind-boggling facts everyone must read. *Forbes Magazine*. URL: <http://www.forbes.com/sites/bernardmarr/2015/09/30/big-data-20-mind-boggling-facts-everyone-must-read>.
- Mei, Q. and Zhai, C. (2005). Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 198–207. ACM.

- Mitsa, T. (2010). *Temporal data mining*. CRC Press.
- Paulovich, F. V., Nonato, L. G., Minghim, R., and Levkowitz, H. (2008). Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping. *IEEE Transactions on Visualization and Computer Graphics*, 14(3):564–575.
- Paulovich, F. V., Oliveira, M. C. F., and Minghim, R. (2007). The projection explorer: A flexible tool for projection-based multidimensional visualization. In *Computer Graphics and Image Processing, 2007. SIBGRAPI 2007. XX Brazilian Symposium on*, pages 27–36. IEEE.
- Paulovich, F. V., Toledo, F., Telles, G. P., Minghim, R., and Nonato, L. G. (2012). Semantic wordification of document collections. In *Computer Graphics Forum*, volume 31, pages 1145–1153. Wiley Online Library.
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.
- Rani, S. and Sikka, G. (2012). Recent techniques of clustering of time series data: a survey. *International Journal of Computer Applications*, 52(15).
- Ratanamahatana, C., Keogh, E., Bagnall, A. J., and Lonardi, S. (2005). A novel bit level time series representation with implication of similarity search and clustering. In *Advances in knowledge discovery and data mining*, pages 771–777. Springer.
- RJaM, M. (2000). An introduction to mathematical statistics and its applications.
- Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425.
- Sakai, T., Flanagan, B., Zeng, J., Nakatoh, T., and Hirokawa, S. (2012). Search engine focused on multiple features of scientific articles. In *Advanced Applied Informatics (IIAIAI), 2012 IIAI International Conference on*, pages 214–217. IEEE.
- Shavlik, J. W. and Deitterich, T. E. (1991). *Readings in Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition.
- Sun, J., Ma, J., Liu, X., Liu, Z., Wang, G., Jiang, H., and Silva, T. (2013). A novel approach for personalized article recommendation in online scientific communities. In *System Sciences (HICSS), 2013 46th Hawaii International Conference on*, pages 1543–1552. IEEE.



- Swain, M. J. and Ballard, D. H. (1991). Color indexing. *International journal of computer vision*, 7(1):11–32.
- Tejada-Gamero, E., Minghim, R., and Nonato, L. G. (2003). On improved projection techniques to support visual exploration of multi-dimensional data sets.
- Valdivia, A. M. C. (2007). *Mapeamento de dados multidimensionais usando árvores filogenéticas: foco em mapeamento de textos*. PhD thesis, Universidade de São Paulo.
- Vlachos, M., Hadjieleftheriou, M., Gunopulos, D., and Keogh, E. (2003). Indexing multi-dimensional time-series with support for multiple distance measures. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 216–225. ACM.
- Ware, M. and Mabe, M. (2015). The stm report: An overview of scientific and scholarly journal publishing.