

**Universidad Rovira I Virgili
Escuela Técnica Superior de Ingeniería**

**Tesis para la obtención del título de
Máster en Ingeniería Informática y de la Seguridad**

Directora: Dra. Aïda VALLS MATEU

**GENERACIÓN INTELIGENTE DE PROTOTIPOS DE SECUENCIAS DE
EVENTOS Y SU APLICACIÓN EN ANONIMIZACIÓN DE DATOS**

Marko Andrés Vargas Ángel

markingerr@hotmail.com

Tarragona, 5 de Junio de 2009

Agradecimientos

A Dios padre por brindarme el bienestar durante todo este proceso y poderlo culminar enhorabuena.

A la Universidad Rovira I Virgili (URV) por la oportunidad que me brindaron de realizar este master gracias a la beca que me otorgaron.

Durante el desarrollo de este trabajo conté con la colaboración, el conocimiento, la disponibilidad y la constante asesoría de mi directora de tesis quien fue guiando y afirmando cada paso que iba dando, los cuales, con su gran ayuda fraguaron el camino hacia el logro de este trabajo, a la Dra. Aïda, gracias por toda su colaboración, moltes gràcies.

A mi familia, que desde un principio me apoyaron en todo sentido para emprender este reto, y que desde la distancia (Villavicencio, Colombia) se convirtieron en la fuerza vital que mantienen los ánimos día a día, por todo esto, muchas gracias a mis padres y a mis hermanos por ser la base fundamental en todo sentimiento de apoyo y cariño.

Índice general

1. INTRODUCCION AL PROBLEMA	9
1.1. MINERÍA DE DATOS O <i>DATA MINING</i>	13
1.2. MICROAGREGACIÓN	16
1.3. OBJETIVOS	22
 2. CONCEPTOS BÁSICOS	 24
2.1. SECUENCIAS DE EVENTOS	24
2.1.1. <i>Representación de secuencias</i>	25
2.2. CARACTERÍSTICAS DE LAS SECUENCIAS	25
2.3. ALINEAMIENTO DE SECUENCIAS	26
2.4. PROTOTIPO DE UNA SECUENCIA	28
 3. METODOS EXISTENTES DE PROTOTIPO.....	 30
3.1. MATRIZ DE PESO POSICIONAL (PWM)	31
3.1.1. <i>Otras formas de calcular la matriz PWM</i>	33
3.1.2. <i>Cálculo de la matriz PWM con pseudo-cuentas</i>	35
3.2. MODELO CADENA DE MARKOV	36
3.3. MODELO MARKOV OCULTO (HMM).....	39
3.5. ANÁLISIS DE LOS MÉTODOS PRESENTADOS	41
3.5.1. <i>PWM (Position Weight Matrix)</i>	42
3.5.2. <i>Cadenas de Markov</i>	43
3.5.3. <i>Modelo Markov Oculto Hmm</i>	45
 4. NUEVAS PROPUESTAS.....	 46
4.1. EL MÉTODO OESP (ORDERED ELEMENT SCORING PROTOTYPING).....	46
4.1.1 <i>Consideraciones sobre el método OESP</i>	49
4.2. OESP CON NUEVAS PUNTUACIONES: NP	50
4.3. OESP SIMÉTRICO: OESP-EM	55
4.4. MÉTODO POR-PARES	56

5. DESARROLLO DE UN ENTORNO DE EVALUACIÓN	58
5.1. PWM (POSITION WEIGHT MATRIX)	59
5.2. CADENAS DE MARKOV	61
5.3. MÉTODO OESP	63
5.4. MÉTODO OESP-NP	64
5.5. MÉTODO OESP-EM.....	66
5.6. MÉTODO POR-PARES	67
 6. ESTUDIO COMPARATIVO.....	 70
6.1. MEDIDA DE COMPARACIÓN	70
6.2. RESULTADOS DE LOS MÉTODOS SEGÚN LA POLÍTICA DE COMPARACIÓN	73
6.2.1. Resultados con $\alpha=2$ (igual preferencia para ambos aspectos).....	75
6.2.2. Resultados con $\alpha=8$ (mayor preferencia para segunda clave)	76
6.3. ANÁLISIS DE LOS RESULTADOS	77
 7. CONCLUSIONES FINALES Y CONTRIBUCIONES	 79

Índice de Tablas

Tabla 1. Representación Símbolos/Datos	25
Tabla 2. PWM de 6 secuencias de Figura 4	31
Tabla 3. PWM de Figura 4 usando pseudo-cuentas	35
Tabla 4. Transición <i>nxt</i>	37
Tabla 5. Resultado de la suma de las puntuaciones de las secuencias	48
Tabla 6. Registros ordenados por puntuación	48
Tabla 7. Grupos de secuencias de itinerarios seguidos por turistas	74
Tabla 8. Resultados con $\alpha=2$	75
Tabla 9. Resultados con $\alpha=8$	76

Índice de Figuras

Figura 1. Selección microagregación grupos de tamaño K	20
Figura 2. Ejemplo con seis secuencias de distinta longitud	25
Figura 3. Ejemplo de alineamientos	28
Figura 4. Grupo de 6 secuencias de igual longitud	30
Figura 5. Representación de Secuencia de Logo de un PWM	32
Figura 6. HMM procesos ocultos/observables	40
Figura 7. Ejemplo de dos secuencias que empiezan por 'e'	44
Figura 8. Grupo de secuencias para ejemplos de alternativas OESP	51
Figura 9. Grupo de secuencias + prototipos obtenidos	54
Figura 10. Ejemplo representativo de método POR-PARES (base PWM)	57
Figura 11. Esquema conjuntos de secuencias a representar por prototipos	58
Figura 12. Prototipos generados PWM	60
Figura 13. Prototipos generados cadenas de Markov	62

Figura 14. Prototipos generados OESP.....	64
Figura 15. Prototipos generados OESP-NP.....	65
Figura 16. Prototipos generados OESP-EM	66
Figura 17. Prototipos generados POR-PARES (Base PWM)	68

1. INTRODUCCION AL PROBLEMA

La privacidad se enmarca como el ámbito de la vida personal de un individuo que se desarrolla en un espacio reservado y debe mantenerse confidencial. En el campo informático y sus implicaciones en sistemas de información, se trata información confidencial de individuos la cual debe ampararse dentro del marco de la privacidad. Bajo esta concepción surgen avances tecnológicos que conllevan el aumento del tratamiento automatizado de la información que busca lo que se denomina como "protección de datos", que se concluye hacia la "protección de la privacidad" donde se debe cumplir en términos generales con:

- Mantener información para si mismo.
- Tener la posibilidad de mantenerse "no identificado" (anónimo) respecto a ciertas actividades personales o públicas.

En el tratamiento de datos personales y/o comerciales, en el momento que se requiera publicar información de individuos, organizaciones, etc., se debe manejar un control de divulgación para salvaguardar la información dada en confianza.

Por tanto, cuando se realizan encuestas para obtención de información que nos permita conocer un cierto dominio, se almacena información de un conjunto de individuos que responden a la encuesta. Por motivos de protección de privacidad, la identidad de los encuestados se debe mantener de manera anónima en todas las posteriores publicaciones que se necesiten realizar, con la finalidad de evitar que terceros (usuarios no autorizados) puedan identificar un individuo y conocer información sensible que afecte la integridad del mismo.

Esta anonimidad parte de no publicar datos que directamente revelen la identidad de las persona (como el DNI o el nombre completo), y también de enmascarar datos que si que se publican y que al combinarlos pueden ser identificativos, por ejemplo, si en pueblo tenemos los siguientes datos de una persona, trabajo, lugar y fecha de nacimiento, estado civil, podríamos identificar a la persona en cuestión.

SDC (*Statistical Disclosure Control*) refiere a este tema como el conjunto de métodos para reducir el riesgo de divulgar información, que es el propósito dentro del ámbito de la protección de privacidad como necesidad al control de inferencia estadística.

El objetivo del riesgo de revelación estadística es controlar el riesgo de que la información acerca de individuos específicos pueda ser extraída de entre los

resultados estadísticos publicados. Estas determinaciones de seguridad surgen durante la recogida de información con fines estadísticos por organizaciones estadísticas, que deben garantizar que la información suministrada por las personas u organizaciones no se revele de forma no autorizada.

Para asegurar el control del riesgo de revelación de datos se tiene presente tres enfoques generales a la protección:

- I. Reducir detalle en las clasificaciones. Es decir, aumentar el tamaño de los grupos más pequeños. Esta aproximación tiende a resultar en la pérdida de una gran cantidad de información útil.
- II. Modelo basado en la imputación (perturbación). Ajustar un modelo a los datos, a continuación, reemplazar las variables sensibles¹ (o todas) de los registros con una estimación de la respuesta obtenida de la muestra del modelo [Polettini et al, 2002].
- III. Intercambio de datos. Se elige un conjunto de variables de intercambio (probabilísticamente implicadas en la identificación), y se seleccionan al azar (o basado en el riesgo) pares de registros de respuesta, en la que el canje de los valores de las variables se intercambiaron conjuntamente. En esta aproximación se pueden tener reglas para decidir sobre el canje establecido, es decir, no puede codificar los valores específicos de género. [Gomatan et al., 2003].

A partir de estos enfoques, se han definido diversos métodos para enmascarar la información, que se suelen clasificar en dos familias: perturbativos y no perturbativos [Torra&Domingo, 2001].

Perturbativos

Modifica los datos de manera que el riesgo de identificación de las personas se reduce a un nivel aceptable. El conjunto de datos se distorsiona antes de su publicación. Pueden incluirse nuevos datos, suprimir y/o modificar los existentes beneficiando la confidencialidad estadística.

Las principales técnicas perturbativas son:

- Ruido aditivo (*Additive noise*): Añade ruido con la misma estructura de correlación que los datos originales. Adecuado para datos numéricos.
- Distorsión de los datos mediante distribuciones de probabilidad (*Data distortion by probability distribution*): Distorsiona los datos con series estimadas en la función de densidad de las variables.

¹ Son datos no directos (DNI, nombre) que por inferencia pueden llevar a la identificación de un individuo (fecha nacimiento, trabajo, etc.)

- **Microagregación (*Microaggregation*):** Construye pequeños micro-clusters de datos muy similares, después sustituye los datos originales por los valores de la media del grupo (prototipo). El tamaño de los conjuntos (o clusters) debe ser igual o superior a una variable k para garantizar la no revelación.
- **Re-muestreado (*Re-sampling*):** Se toman t muestras independientes $X_1...X_t$ del mismo tamaño n de la variable original V_i . Se ordenan los datos dentro de cada muestra. Se hace la media de los primeros valores de cada muestra, y estos se sustituyen por esta media. Se hace lo mismo con los $n - 1$ valores de las siguientes posiciones.
- **Perdida de compresión (*Lossy compression*):** Considera el archivo de datos como una imagen y aplica algoritmos de compresión (por ejemplo JPEG).
- **Imputación múltiple (*Multiple imputation*):** Se genera una nueva versión con datos simulados creados a partir de múltiples técnicas de imputación a partir de los datos originales. Por ejemplo, un método de imputación consiste en hacer regresiones con una distribución aleatoria del error, para imputar valores 'desconocidos' a una variable continua.
- **Camuflaje (*Camouflage*):** Camufla la información original en un intervalo (conjunto infinito). Es un método adecuado para datos numéricos. Supone una pérdida de información elevada.
- **PRAM (*Post-Randomization Method*):** Sustituye los datos originales por otra información diferente de acuerdo a mecanismos de probabilidad, siguiendo los modelos de cadena de markov. Fusiona adición de ruido, supresión de datos y recodificación.
- **Intercambio ordenado (*Rank swapping*):** Intercambia aleatoriamente un porcentaje de los datos originales.
- **Redondeo (*Rounding*):** Sustituye los valores originales con valores redondeados. Se elige valores pertenecientes a un conjunto predefinido de redondeo. Adecuado para variables numéricas.

No-perturbativos

Estas técnicas no alteran los datos, sino que producen supresiones parciales o reducciones del nivel de detalle del conjunto original.

Las principales técnicas no-perturbativas son:

- **Muestreado (*Sampling*):** Publica una muestra de datos originales. Adecuado para variables categóricas. En variables continuas, riesgo de revelación elevado.
- **Recodificación global (*Global recoding*):** Combina categorías para formar nuevas categorías menos específicas. En variables continuas, discretiza valores (paso de infinito a finito).

- Codificación superior e inferior (*Top and bottom coding*): Determina un umbral superior e inferior y forma nuevas categorías con los valores extremos. Caso concreto de 'Recodificación global'.
- Supresión local (*Local suppression*): Elimina ciertos valores con el objetivo de incrementar el conjunto de registros concordantes. Orientado a variables categóricas.

En este trabajo vamos a centrarnos en los métodos perturbativos. Concretamente nos interesan los métodos denominados métodos de micro-agregación, que consisten en crear grupos de individuos similares y sustituir sus valores originales por los valores del prototipo del grupo al cual pertenecen.

Este trabajo de máster se enmarca dentro de dos proyectos de investigación financiados por el Gobierno Español: el proyecto E-AEGIS (TSi-2007-65406-c03) y el proyecto ARES (CSD-2007-00004). Una de las líneas de investigación de estos proyectos está centrada en el estudio de métodos de anonimización de datos no numéricos. En particular, en este trabajo nos vamos a centrar en el caso de secuencias temporales de valores categóricos, denominadas Secuencias de Eventos [Dong&Pei, 2007]. Este trabajo también está relacionado con el proyecto 2008 TURISME-02 financiado por la URV para estudiar la aplicación de técnicas de análisis de datos de turistas basados en su localización.

Así pues, trataremos el caso de valores categóricos encadenados uno detrás de otros (secuencias) por una relación temporal, de forma que cada valor corresponde a un evento producido después de su predecesor.

Hoy en día se ha incrementado el estudio en el desarrollo de técnicas para el análisis de secuencias de datos debido a la necesidad, en ciertos campos, de herramientas que permitan analizar este tipo de datos de forma automática y obtener conocimiento interesante que pueda ser utilizado para mejorar los procesos tradicionales.

Algunos ejemplos de secuencias de datos categóricos son las siguientes:

- Cadenas o secuencias biológicas, generalmente proteínas, ADN o ARN, las cuales llevan a indicaciones funcionales como por ejemplo representaciones de familias de proteínas y construcción de modelos para la identificación de miembros potenciales de la familia, así como también deducciones evolutivas.
- La lista de las páginas Web visitadas por un usuario durante un determinado tiempo o un rango establecido.

- La lista de lugares donde un turista ha estado durante sus vacaciones en cierto país o ciudad.
- Los datos del historial clínico de un paciente, donde figuren los diferentes tratamientos que ha recibido por una cierta enfermedad o durante estancias clínicas.

La disponibilidad de grandes volúmenes de información y el uso generalizado de herramientas informáticas ha transformado el análisis de datos. A los métodos de análisis automático de datos para extraer conocimiento relevante se les conoce como métodos de Minería de Datos (*Data Mining*). Este es un área relevante dentro del campo de investigación en Inteligencia Artificial.

Recientemente están surgiendo líneas de investigación centradas en desarrollar métodos de análisis para secuencias temporales de datos, que se denominan *Temporal Data Mining* [Hernández et al, 2004]. Sin embargo, hasta ahora, no se han preocupado mucho de desarrollar técnicas de protección de privacidad para datos que son secuencias de valores categóricos.

A continuación, se va a presentar una breve introducción a la minería de datos y a los métodos de protección de la privacidad, lo cual nos permitirá definir mas adelante los objetivos de este trabajo de máster.

1.1. MINERÍA DE DATOS O DATA MINING

Las técnicas de minería de datos persiguen el descubrimiento automático del conocimiento contenido en la información almacenada de modo ordenado en bases de datos. Estas técnicas tienen como objetivo descubrir patrones, perfiles y tendencias a través del análisis de los datos utilizando tecnologías de reconocimiento de patrones, redes neuronales, lógica difusa, algoritmos genéticos y otras técnicas avanzadas de análisis de datos[Pérez&Santín, 2007].

El análisis de la información recopilada puede hacerse de forma manual sólo si el conjunto de datos es pequeño. Sin embargo, cuando la cantidad de datos de los que disponemos aumenta, la resolución manual del problema se hace intratable. Aquí es donde entra en juego el conjunto de técnicas de análisis automático al que nos referimos al hablar de minería de datos o de procesos completos de extracción de información como KDD (*Knowledge Discovery in Databases*), que se encarga además de preparación de los datos y de la interpretación de los resultados obtenidos [Shapiro&Frawley, 1991].

El proceso de minería de datos, es el proceso de aplicar a una determinada base da datos las operaciones requeridas de selección, exploración, muestreo, transformación y métodos de modelado para extraer los patrones y

posteriormente evaluarlos para identificar el conjunto de ellos que representarán el conocimiento u objetivo final de la necesidad del usuario. Como todo proceso, representa una conjunción de pasos, que consta de cuatro fases [Pérez, 2006]:

- *Selección de los objetivos:* En esta fase hay que estudiar el problema y decidir cuál es la meta del proyecto. Así mismo es deseable formular unas expectativas de éxito o fracaso del proyecto dado que estos conceptos son relativos. Si se hace bien el planteamiento del problema, se descubren fácilmente las fuentes de datos y los algoritmos de minería de datos que se aplicarán.
- *Preparación de los datos:* Es una etapa previa al proceso de análisis de los datos, consta de tres pasos: selección de los datos, preproceso de los datos y transformación de los datos.
- *Construcción del modelo:* Es una fase de relevancia porque es en ella donde se aplican los diferentes algoritmos de análisis de datos sobre los datos transformados y preparados en las etapas anteriores.
- *Análisis de los resultados:* Esta fase conlleva la interpretación y evaluación de los resultados obtenidos en la fase anterior.

Según la especificación del problema existen técnicas de análisis diversas. Estas técnicas provienen de la Inteligencia artificial y de la estadística. Las técnicas más representativas son:

- *Redes neuronales:* Son un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas en una red que colabora para producir un estímulo de salida.
- *Árboles de decisión:* Un árbol de decisión es un modelo de predicción utilizado en el ámbito de la inteligencia artificial, dada una base de datos se construyen estos diagramas de construcciones lógicas que sirven para representar y categorizar una serie de condiciones que suceden de forma sucesiva, para la resolución de un problema. Estos métodos también se denominan Sistemas basados en reglas o Sistemas expertos.
- *Modelos estadísticos:* Es una expresión simbólica en forma de igualdad o ecuación que se emplea en los diseños experimentales y en la regresión para indicar los diferentes factores que modifican la variable de respuesta.
- *Agrupamiento o Clustering:* Es un procedimiento de agrupación de una serie de vectores según criterios habitualmente de distancia; se tratará de disponer los vectores de entrada de forma que estén más cercanos aquellos que tengan características comunes.

Según el objetivo del análisis de los datos, las técnicas utilizadas se clasifican en técnicas predictivas, técnicas descriptivas y técnicas auxiliares.

Las *técnicas predictivas* buscan el modelo para los datos con base a un conocimiento teórico previo. El modelo supuesto para los datos debe contrastarse después del proceso de minería de datos antes de aceptarlo como válido. Formalmente, la aplicación de todo modelo debe superar las fases de *identificación objetiva* (a partir de los datos se aplican reglas que permitan identificar el mejor modelo posible que ajuste los datos), *estimación* (proceso de cálculo de los parámetros del modelo elegido para los datos en la fase de identificación), *diagnosis* (proceso de contraste de la validez del modelo estimado) y *predicción* (proceso de utilización del modelo identificado, estimado y validado para predecir valores futuros de las variables dependientes). En algunos casos, el modelo se obtiene como mezcla del conocimiento obtenido antes y después del *Data Mining* y también debe contrastarse antes de aceptarse como válido.

Las *técnicas descriptivas* no asignan ningún papel predeterminado a las variables. No se supone la existencia de variables dependientes ni independientes y tampoco se supone la existencia de un modelo previo para los datos. Los modelos se crean automáticamente partiendo del reconocimiento de patrones. En este grupo se incluyen las técnicas de *clustering* y segmentación (que también son técnicas de clasificación en cierto modo), las técnicas de asociación y dependencia, las técnicas de análisis exploratorio de datos y las técnicas de reducción de la dimensión (factorial, componentes principales, correspondencias, etc.) y de escalamiento multidimensional.

Las *técnicas auxiliares* son herramientas de apoyo más superficiales y limitadas. Se trata de nuevos métodos basados en técnicas estadísticas descriptivas, consultas e informes y enfocados en general hacia la verificación [Pérez&Santín, 2007].

Como hemos visto, la minería de datos se ocupa de analizar grandes volúmenes de (a menudo no estructurados) datos para descubrir interesantes regularidades o relaciones que a su vez conduzcan a una mejor comprensión de los procesos subyacentes. Dentro de conjuntos de datos secuenciales, el campo de "*minería de datos temporales*" se refiere a este tipo de análisis en el caso de conjuntos de datos con interdependencias temporales.

Minería de datos temporales (*Temporal Data Mining*)

Minería de datos temporales se refiere a la minería de datos de grandes conjuntos de datos secuenciales. Por secuencia de datos, nos referimos a los

datos que se ordenan con respecto a algún índice. Por ejemplo, las secuencias temporales constituyen una clase popular de series de datos secuenciales, donde los registros son indexados por el tiempo.

Los atributos relacionados con el aspecto temporal de la información presente en algunos tipos de bases de datos necesitan de un trato diferente con respecto a otros tipos de atributos. Sin embargo, la mayoría de las técnicas de minería de datos tradicionales tienden a tratar los datos como una colección desordenada de acontecimientos, haciendo caso omiso de su información temporal.

En el proceso de minería de datos que contienen información temporal, se requiere que los datos deban considerarse como una secuencia de eventos con un orden de precedencia. Con base en esto, el objetivo de minería de datos temporales es descubrir relaciones ocultas entre las secuencias y sub-secuencias de los acontecimientos. El descubrimiento de las relaciones entre las secuencias de los acontecimientos implica principalmente tres etapas: la representación y modelado de los datos de secuencia, la definición de medidas de similitud entre las secuencias, y la solución de modelos y representaciones de los problemas reales de la minería.

Dependiendo de la naturaleza de la secuencia de eventos, los enfoques para resolver los problemas pueden ser muy diferentes. Una secuencia compuesta por una serie de símbolos nominales de un determinado alfabeto se conoce como una *secuencia temporal* y una secuencia continua de los elementos de valor real, que se conoce como *una serie temporal* [Antunes&Oliveira, 2001].

1.2. MICROAGREGACIÓN

La *microagregación* es una técnica perturbativa que consiste en distorsionar los datos empíricos (reales) a fin de garantizar el anonimato de los microdatos². El procedimiento de perturbación/sustitución consiste en construir grupos de k individuos cuya construcción se realiza bajo un criterio de máxima similitud entre los individuos que conformaran el grupo (o cluster). Para perturbar los datos, a los individuos pertenecientes a un grupo se les sustituye los valores originales por valores computados (un valor prototipo o valor medio del cluster). Así pues los pasos fundamentales del proceso de microagregación son los siguientes:

- Construcción de grupos con los datos originales según un criterio de máxima similaridad (donde en cada grupo debe haber al menos k individuos).
- Construcción de un prototipo representativo de cada grupo.

² Fichero donde cada registro contiene información sobre un individuo (ciudadano o compañía)

- Substitución de los registros originales por sus correspondientes prototipos.

En el primer paso, la formación de grupos se basa en criterios de máxima similitud, lo que implica que es necesaria una medida de similitud entre registros. Cada registro individual puede ser visto como un punto y todo el conjunto de datos como un conjunto de puntos multidimensionales (la dimensión dependerá del número de variables medidas dentro de cada registro). De esta forma, la semejanza entre registros puede ser medida utilizando una distancia.

En el siguiente paso, se utilizan operadores de agregación para unificar los valores de las variables de los individuos de cada cluster, obteniendo un valor prototipo final que representa los valores tomados por cada individuo en ese cluster. Generalmente suelen usarse medias o cálculos de frecuencias para la determinación del prototipo.

La tercera fase consiste simplemente en sustituir los valores originales por su correspondiente prototipo según el cluster al que pertenece el individuo en cuestión.

En el proceso de microagregación se debe tratar de minimizar la pérdida de información que se pueda generar en su aplicación o resultado final. Normalmente, esta pérdida se calcula de la forma siguiente:

Dado un conjunto de datos con p variables continuas y n registros resultantes de observar las p variables en n individuos. Con estos vectores formamos N grupos de tamaño k con n_i individuos en cada grupo i -ésimo ($n_i \geq k$ y $n = \sum n_i$).

Denotamos:

$x = (x_1, x_2, \dots, x_p)$ registro de un individuo, donde x_i son valores de variables.

x_{ij} j -ésimo registro del grupo i $j=1, \dots, n_i$

G_i prototipo del grupo i $i=1, \dots, N$

G prototipo del conjunto de datos

Suma de Cuadrados dentro de los grupos

$$SSE = \sum_{i=1}^N \sum_{j=1}^{n_i} (x_{ij} - G_i)(x_{ij} - G_i)$$

Suma de Cuadrados Total

$$SST = \sum_{i=1}^N \sum_{j=1}^{n_i} (x_{ij} - G)'(x_{ij} - G)$$

La partición óptima será aquella que minimice la suma de cuadrados dentro de los grupos (SSE) o de forma equivalente la que maximice la suma de cuadrados entre grupos (SSA). Estas sumas de cuadrados pueden utilizarse como medida de la pérdida de información:

$$L = \frac{SSE}{SST}$$

El valor de L es el que obtenemos como indicador de pérdida de información cuyo rango esta entre 0 y 1. Cuanto más cercano sea a 0 menor será la pérdida de información debida a la microagregación.

Por otro lado, se debe asegurar la calidad de la protección de la privacidad. En este sentido se define la propiedad de k -anonimidad, que evita relacionar un individuo con sus datos privados, para dar garantías de que las personas que son objeto de los datos no pueden ser re-identificados mientras que los datos siguen siendo prácticamente útil. Se dice que un fichero proporciona protección de k -anonimato si la información para cada persona no puede distinguirse de, al menos, otras $k-1$ personas cuya información aparece también en la publicación, a través de un conjunto de atributos que se consideran sensibles a la privacidad.

El concepto de *k-anonimato* [Samarati, 1998] establece uno de los principales requisitos que ha sido seguido por la comunidad estadística y por la liberación de organismos de los datos, puesto que los datos deben ser publicados de forma conjunta en relación con no menos de un cierto número de los encuestados.

En la definición de k -anonimidad se tiene en cuenta la clasificación de los tipos de atributos en un conjunto de microdatos [Domingo&Torra, 2005]:

- *Identificadores*: Identifican al sujeto de forma univoca.
- *Casi-identificadores*: Combinación de atributos que pueden ser vinculados con información externa para reidentificar a algunos sujetos a los que se refieren registros del fichero.
- *Atributos confidenciales*: Contienen información comprometida del sujeto.
- *Atributos no confidenciales*: No contienen información comprometida del sujeto.

La definición formal de esta propiedad dice [Sweeney, 2002]: Un conjunto se dice que satisface la k -anonimidad para $k > 1$ si, para cada combinación de valores de casi-identificadores, existen al menos k registros en el conjunto que comparten la combinación.

Cada versión de los datos debe ser de tal manera que cada combinación de valores de los cuasi-identificadores puede ser indistintamente correspondida a por lo menos k encuestados.

El k -anonimato intenta establecer el equilibrio entre la pérdida de información y el riesgo de revelación mediante la siguiente condición:

A continuación se presenta brevemente los métodos de anonimización básicos.

Métodos Tamaño Fijo

Estos métodos heurísticos ordenan de forma ascendente o descendente los registros de acuerdo con un criterio unidimensional determinado. Una vez ordenados, se forman grupos de tamaño fijo k con registros sucesivos. Si el número total de registros n no es múltiplo de k , entonces el último grupo contendrá más de k registros. Dentro de cada grupo se reemplaza el valor de cada variable por su media dentro del grupo.

- **Métodos del Eje-Único**

Resultan muy efectivos cuando las variables están fuertemente correlacionadas. La ordenación (ascendente o descendente) se puede realizar siguiendo varios criterios:

- *Ordenación por componentes principales.* Se proyectan los registros en un nuevo espacio con ejes ortogonales. Se ordenan los registros por la 1ª componente principal (eje), siendo ésta una variable transformada de forma que está altamente correlacionada con el resto de variables originales.

- *Ordenación por una variable particular.* Dicha variable debe reflejar de alguna forma el tamaño del registro.

- *Ordenación por la suma de los z-scores.* Todas las variables son estandarizadas y para cada registro se calculan las sumas de dichos valores estándar, ordenando por esta característica.

- **Método de Ordenación Individual**

Cada variable se considera de forma independiente de las demás. Los registros son ordenados por la 1ª variable, formándose grupos de k registros sucesivos y reemplazando los valores de cada variable por su media dentro de cada grupo. Un procedimiento similar se lleva a cabo con el resto de variables (ordenando por la 2ª, 3ª,...y así sucesivamente). Este método preserva gran cantidad de información pero existe un mayor riesgo de difusión de datos que pueden ser confidenciales. El usuario puede llegar a deducir que el verdadero valor de una variable en el grupo i se encuentra entre la media del grupo $i-1$ y la del $i+1$. Si dichas medias tienen valores próximos, el intervalo que se conoce puede ser demasiado estrecho.

Otra desventaja de éste método consiste en que los n registros no son agrupados en un único vector de medias básico, ya que la microagregación se realiza para cada variable cada vez por lo que se obtiene una partición diferente (y por lo tanto un registro de medias distinto) para cada variable del conjunto de datos.

También existen métodos de microagregación de tamaño fijo que utilizan múltiples variables simultáneamente (multi-variables), estos suelen ser métodos basados en algoritmos de clustering tradicionalmente usados en minería de datos.

Métodos con Tamaño Variable

Las técnicas de microagregación explicadas suelen utilizar grupos de registros de tamaño fijo. Sin embargo, la posibilidad de variar el tamaño de un grupo a otro de manera que dicha agrupación se adaptara mejor a la distribución inicial de los datos, tendería a perder menos información (conservando siempre la condición de ser al menos de tamaño k). La siguiente figura ilustra las ventajas de la microagregación con grupos de tamaño variable:

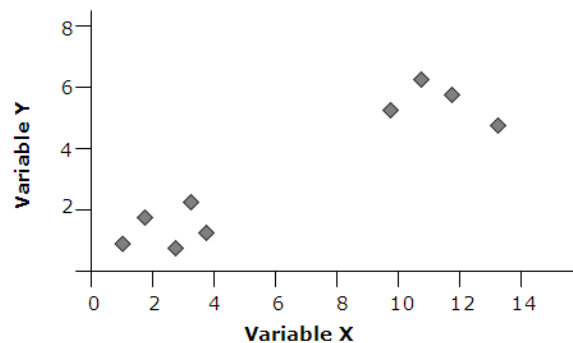


Figura 1. Selección microagregación grupos de tamaño K

El gráfico muestra dos variables y nueve datos. Si utilizáramos microagregación de tamaño fijo $k=3$, obtendríamos una partición de datos en tres grupos, lo que parece poco natural para la distribución dada. Si por el contrario utilizamos microagregación de tamaño variable $\geq k$, los cinco datos a la izquierda del gráfico podrían ser incluidos en un mismo grupo y los cuatro restantes en otro. De esta forma la agrupación de registros resulta más acorde con la distribución de los datos y la pérdida de información es menor.

Con este ejemplo se puede ver que la microagregación con tamaño variable representa una ventaja en casos como éste. Cuando se considera tamaño variable los clusters pueden tener entre k y $2k-1$ elementos, de forma que no pueda haber grupos demasiado pequeños, con riesgo de re-identificación, ni grupos demasiado grandes, lo cual implica una gran pérdida de información porque todos los individuos de este grupo se sustituyen por el mismo prototipo y por tanto se convierten en indistinguibles, perdiendo sus cualidades particulares y su información individual.

Los siguientes son dos métodos de microagregación con tamaños variables:

- Microagregación Genética

Debe su nombre a que representa las particiones como cadenas binarias (al igual que cromosomas) y combina una búsqueda directa y aleatoria para encontrar la agrupación óptima. Se trata de un método difícil de adaptar al caso multivariante ya que un espacio multidimensional está ordenado sólo parcialmente por lo que la representación binaria no resulta la más adecuada.

- Método de Ward

Se trata de un método bastante efectivo y proporciona un algoritmo recursivo que optimiza la solución en cada paso. En cada iteración se unen dos vectores (ó grupos de vectores), de forma que el incremento de la suma de cuadrados (SSE) debida a su unión, sea mínimo.

El método de Ward surge originariamente, como un método de agrupación jerárquica de vectores que optimiza SSE, sin considerar ningún tamaño mínimo k para los grupos. Por ello, ha sido adaptado para trabajar con microagregación introduciendo ligeras modificaciones en lo que denominamos el *Algoritmo Modificado de Ward* [Mas, 2000].

En [Mateo&Domingo, 1998] se establece un estudio sobre métodos de microagregación donde se puede denotar la descripción de microagregación óptima tomando así determinaciones sobre algoritmos de microagregación que pueden ser aplicables como desarrollo de solución.

1.3. OBJETIVOS

Esta tesis de master esta enmarcada dentro de los proyectos españoles E-AEGIS y CONSOLIDER, en los que el grupo ITAKA participa, donde se estudian y desarrollan métodos de protección de la privacidad para bases de datos complejas.

El creciente interés en el estudio de datos secuenciales temporales en distintos ámbitos, en estos proyectos se ha propuesto estudiar los métodos que trabajan con secuencias de eventos. En este caso, los valores secuenciales son valores categóricos (palabras o símbolos) relacionados por un orden de precedencia total.

Se dispone ya de un trabajo previo consistente en la adaptación de métodos de clustering jerárquicos tradicionales en minería de datos para su aplicación a la anonimización como métodos de microagregación [Valls&Gómez, 2008]. Los algoritmos clásicos fueron adaptados para garantizar que los clusters obtenidos tengan un tamaño variable que cumpla con la propiedad de k -anonimidad (el número de elementos de cada cluster está entre k y $2k-1$). Por otro lado, se ha definido una medida de similitud entre secuencias de eventos, que permite estimar su distancia. Esta medida se puede usar durante el proceso de microagregación para determinar qué secuencias son más parecidas y generar los clusters correspondientes [Gómez&Valls, 2008].

Con este trabajo previo se puede resolver la primera fase del proceso de microagregación detallado anteriormente. La fase siguiente consiste en calcular el prototipo de un grupo de secuencias. Este es el objetivo de este proyecto de máster.

Este objetivo general se puede desglosar en subobjetivos más específicos:

- Estudiar y realizar un estado del arte de qué métodos existen que permiten obtener un *prototipo* a partir de una secuencia de símbolos.
- Evaluar cuales son los puntos fuertes y los puntos débiles de cada método y analizar sus ventajas y desventajas en relación a la construcción de prototipos para microagregación.
- Adaptar alguno de los métodos existentes o diseñar un método alternativo que sea más adecuado para la protección de secuencias de símbolos.
- Implementar este método alternativo, analizar sus resultados con diversos conjuntos de datos y comparar estos resultados con los obtenidos en los métodos existentes previamente.

Es importante destacar, que aunque el enfoque del trabajo será construir un prototipo para ser usado en un método de microagregación para generar una versión protegida de una base de datos, los mismos métodos pueden usarse en los métodos de minería de datos para generar los prototipos de los clusters generados. En este caso, los prototipos sirven para describir las clases aprendidas, para interpretar los datos o para generar reglas de clasificación.

2. CONCEPTOS BÁSICOS

En esta sección se definen algunos conceptos básicos de este trabajo de máster, para facilitar la comprensión del documento y aclarar la nomenclatura que se va a utilizar.

2.1. SECUENCIAS DE EVENTOS

A los elementos de las secuencias que vamos a estudiar se les suele denominar Eventos, puesto que se corresponden a actividades realizadas en un cierto instante (o período) de tiempo. Así, por ejemplo, en las secuencias mencionadas anteriormente tenemos distintos tipos de eventos: página Web consultada, lugar turístico visitado, visita o tratamiento médico, etc.

Cualquier tipo de evento vendrá representado con un valor categórico, formado por una o más palabras. Estos valores categóricos van a ser sustituidos por símbolos más sencillos, para facilitar su tratamiento informático. Concretamente vamos a representar los eventos con caracteres en un alfabeto A.

Así pues, los algoritmos que vamos a estudiar en este trabajo van a manejar *secuencias de símbolos*.

Los símbolos de una secuencia están encauzados unos detrás de otros de manera encadenada cada uno después de un predecesor siguiendo un orden temporal.

Para facilitar la comprensión de los métodos que se expondrán en este trabajo, se va a usar un ejemplo concreto en el que las secuencias representan itinerarios por sitios turísticos de una ciudad, los cuales indican el trayecto que cada persona ha seguido para visitar la ciudad, y esto puede ser interesante para analizar el comportamiento del usuario en un contexto particular.

Para la facilidad en el análisis de resultados, los eventos se van a representar con símbolos formados por una sola letra en un alfabeto A. El orden de los símbolos indica el orden en que se han realizado los eventos. Cada símbolo corresponde a un único evento.

Un ejemplo con datos itinerarios de turistas sería el de la Tabla 1, donde tenemos un alfabeto A con 4 elementos.

<i>Símbolos</i>	<i>información</i>
E	Visita a la catedral
F	Visita del puerto
G	Visita al museo
K	Visita al ayuntamiento

Tabla 1. Representación Símbolos/Datos

2.1.1. Representación de secuencias

Vamos a indicar un conjunto de secuencias D , de diferente longitud w , dados los símbolos E-F-G-K dentro del alfabeto que indicaremos como A , los individuos son representados con Id el cual registra un determinado itinerario de visitas que se ve en su correspondiente *secuencia de símbolos*, como lo representamos en la siguiente Figura 2.

<i>Id</i>	<i>Secuencias</i>
1	F E E K F E G K G
2	F G E F F E E
3	F E E F F G
4	G E E F F F G
5	E G E K F E F K
6	E G E F F K

Figura 2. Ejemplo con seis secuencias de distinta longitud

2.2. CARACTERÍSTICAS DE LAS SECUENCIAS

En [Liao, 2005] se definen un conjunto de características sobre secuencias de eventos, que hacen referencia a la distribución de los símbolos de los eventos dentro de la secuencia. Siguiendo esta misma aproximación describiremos las

secuencias que van a ser estudiadas en este trabajo de máster de la siguiente forma:

- Los eventos son valores categóricos que pertenecen a un conjunto finito de etiquetas lingüísticas (ciudades, lugares, páginas Web).
- Las secuencias han sido incluidas en la muestra de manera uniforme en el sentido de que los tiempos entre eventos no son tomados en cuenta.
- Las secuencias son univariantes, sólo un concepto se estudia.
- Las longitudes de las secuencias de las personas no son iguales.
- Eventos se pueden repetir en la secuencia.

En este trabajo, la construcción de una secuencia prototipo que define lugares visitados por un grupo de personas para su posterior protección de confidencialidad a cada individuo, toma como base las secuencias de los acontecimientos que representan el comportamiento del usuario en un contexto particular, que se representa en los turistas que visitan la ciudad, donde las secuencias muestran el itinerario que cada persona ha seguido a la visita de lugares interesantes en esta ciudad, este desarrollo se documenta en [Valls&Gómez, 2008].

En este caso de estudio y siguiendo las características descritas, se va a trabajar con una base de datos con distintos turistas de los cuales sabemos la secuencia de lugares visitados aunque no el tiempo entre cada visita, cada turista puede visitar el número de lugares que considere oportuno y puede ir varias veces a un mismo sitio (por ejemplo, un turista visitó el mismo lugar, La Catedral, más de una vez durante sus vacaciones).

2.3. ALINEAMIENTO DE SECUENCIAS

El objetivo de una alineación de secuencias se fundamenta en la comparación de secuencias, donde dicha comparación tiene como finalidad identificar si las secuencias son homologas, es decir si estas tienen un origen común, lo cual puede generar información sobre su función, estructura e inferir con base en trozos o partes iguales de las secuencias comparadas para obtener conclusiones. En este sentido, con la alineación de secuencias obtenemos información sobre regiones que se parecen entre secuencias comparadas así como también encontrar el alineamiento que maximice el parecido.

La alineación de secuencias proviene de una actividad de la Biología, que los biólogos llaman 'biología comparada' y su desarrollo se ha dado en secuencias o cadenas biológicas, generalmente proteínas, ADN o ARN. Desde el punto de vista bioinformático, la alineación de secuencias es el procedimiento por el cual se comparan dos secuencias (o más), buscando los patrones de caracteres (nucleótidos o aminoácidos) que mantienen el mismo orden dentro de las secuencias comparadas, con lo cual se puede concluir información como qué proteínas tienen un origen común, predecir la estructura y/o función de proteínas, deducir historia evolutiva y mas interpretaciones biológicas que se hagan del resultado de la información[Mount, 2004].

Suponemos que tenemos N secuencias a estudiar. Se hace un '*alineamiento de secuencias*' para dos o mas secuencias, que corresponde a compararlas, lo que da información sobre regiones que se parecen entre dichas secuencias comparadas. Las secuencias pueden estar dadas en longitudes (tamaño de la secuencia) iguales o diferentes, el alineamiento de secuencias esta basado en métodos que nos ayudan a determinar un tramo de eventos de longitud w que agrupa, de manera congruente, la región que representa mejor intersección para las N secuencias.

El mejor alineamiento de todos los posibles dentro de la comparación de secuencias viene dado por la aplicación de un sistema de puntuación ("*score*") tal que a mayor valor, mayor es su significación. Esto se verá en la sección 3.

Dentro del alineamiento de secuencias se puede distinguir entre alineamientos globales y alineamientos locales. Los alineamientos globales trabajan sobre secuencias completas mientras que los locales trabajan sobre alineamientos de sub-secuencias. Una estrategia general de alineamiento global es el algoritmo Needleman-Wunsch [Needleman&Wunsch,1970] basado en programación dinámica. Los alineamientos locales son más útiles para secuencias diferenciadas en las que se sospecha que existen regiones muy similares o motivos de secuencias similares dentro de un contexto mayor. El algoritmo Smith-Waterman es un método general de alineamiento local basado en programación dinámica. Con secuencias suficientemente similares, no existe diferencia entre alineamientos globales y locales [Brudno&Malde, 2003].

En la Figura 3, tomamos dos secuencias como ejemplo de alineamiento global y local.

2 Secuencias No Alineadas										
	F	E	F	F	K	E	G	F		
	F	F	F	K	G	K	G	F		
Alineamiento global										
	F	E	F	F	K	E	-	G	F	
	F	-	F	F	K	G	K	G	F	
Alineamiento local										
	-	-	F	F	K	E	-	-	-	
	-	-	F	F	K	G	-	-	-	

Figura 3. Ejemplo de alineamientos

En este caso (Figura 3) tomamos $N = 2$ llamado *alineamiento de pares*, los métodos de alineamiento de pares, o emparejamientos, se utilizan para encontrar la mejor coincidencia en bloque (local) o alineamiento global de dos secuencias. Los alineamientos de pares sólo pueden utilizarse con dos secuencias a la vez, pero son eficientes de calcular, y son utilizados a menudo en métodos que no requieren precisión extrema. También encontramos $N > 2$ lo cual requiere un proceso llamado "*alineamiento múltiple de secuencias*"; el alineamiento múltiple de secuencias es una extensión del alineamiento de pares que incorpora más de dos secuencias al mismo tiempo. Los métodos de alineamiento múltiple intentan alinear todas las secuencias de un conjunto dado.

2.4. PROTOTIPO DE UNA SECUENCIA

Un prototipo es un ejemplar base o estándar de los elementos de un mismo grupo (o cluster). Un prototipo especifica cuales son los elementos comunes encontrados dentro de un grupo, combinan los valores más representativos de atributos de los elementos pertenecientes al grupo. En técnicas de minería de datos los prototipos se denominan centroides, que se define como el centro (o vector media) del grupo.

Con respecto a un conjunto N de secuencias, un prototipo es una representación de las N secuencias en una sola secuencia promedio que se puede relacionar como una secuencia intersección del grupo formado por las N secuencias.

El cálculo de prototipos es un proceso habitual en minería de datos. Los prototipos se utilizan para describir las clases generadas con algoritmos de aprendizaje automático (un perfil), para interpretar los resultados o para generar reglas de clasificación y discriminación [Dong&Pei, 2007].

Otros dominios también necesitan calcular prototipos, por ejemplo, en Biología todos los procesos o sistemas de alineamiento de secuencias se concluyen en definir una secuencia consenso que representa el empalme de N secuencias, y que corresponde a su *prototipo*.

En el caso de la privacidad de los individuos (p.e. turistas), los prototipos se son necesarios en los métodos de microagregación. En la última fase de este proceso, se sustituyen los datos originales (que se quieren proteger) de un grupo de individuos por los datos obtenidos en el *prototipo* que representa bien los individuos del grupo, sin revelar información que se pueda usar para re-identificación.

3. METODOS EXISTENTES DE PROTOTIPO

Para la especificación de cuáles son los elementos comunes en un grupo de secuencias obtenido, se estudian los métodos que nos llevan a la construcción de un prototipo.

Una vez realizado un estudio de la literatura existente sobre métodos de generación de prototipos en secuencias categóricas, se ha observado que este tipo de métodos se ha estudiado y aplicado muy mayoritariamente en genética. En este caso el prototipo indica una secuencia teórica de nucleótidos o aminoácidos en la cual cada nucleótido o aminoácido es el que se presenta más frecuentemente en ese sitio en las diferentes secuencias que ocurren en la naturaleza.

Debido a las características de este dominio, los métodos de prototipos consideran secuencias de la misma longitud, puesto que en genética, las secuencias de ADN previamente se alinean para encontrar un tramo óptimo que permita la comparación y el estudio de la parte más significativa de la secuencia, tal como se ha explicado en el capítulo anterior.

Con base en la determinación de secuencias de igual longitud y tomando como referencia las secuencias indicadas en la Figura 2, se tomara un segmento de dichas secuencias de tal forma que todas queden de la misma longitud. Este será el grupo de secuencias con el que se trabajara a continuación en la aplicación (ejemplos) de los métodos descritos en los siguientes apartados.

$$\left\{ \begin{array}{cccccc} F & E & E & K & F & E \\ F & G & E & F & F & E \\ F & E & E & F & F & G \\ G & E & E & F & F & F \\ E & G & E & K & F & E \\ E & G & E & F & F & K \end{array} \right\}$$

Figura 4. Grupo de 6 secuencias de igual longitud

3.1. MATRIZ DE PESO POSICIONAL (PWM)

El método PWM [Dong&Pei, 2007] se basa en la puntuación de los símbolos de las secuencias de acuerdo con su posición. Todas las secuencias deben tener la misma longitud. Estas puntuaciones se guardan en una matriz denominada Matriz de Peso Posicional (*Position Weight Matrix*, PWM).

Una matriz de peso posicional M : $A \times w$ contiene valores numéricos $M(x,p)$ para cada $(x \in L)$ y para $(1 \leq p \leq w)$. Cada fila de M es un símbolo del alfabeto L y cada columna es una posición de las secuencias. Dentro de la matriz M asociada a PWM se asocian los valores que indican la aparición de los símbolos (correspondientes a diferentes eventos) para determinadas posiciones. El número $M(x,p)$ es generalmente proporcional a la frecuencia de x en la posición p .

En la Tabla 2, se muestra la matriz PWM que se obtiene con el conjunto de secuencias presentado en la Figura 4.

	1	2	3	4	5	6
E	$\frac{2}{6}$	$\frac{3}{6}$	1	0	0	$\frac{3}{6}$
F	$\frac{3}{6}$	0	0	$\frac{4}{6}$	1	$\frac{1}{6}$
G	$\frac{1}{6}$	$\frac{3}{6}$	0	0	0	$\frac{1}{6}$
K	0	0	0	$\frac{2}{6}$	0	$\frac{1}{6}$

$M(x,p)$, frecuencia de F en posición 6.

Tabla 2. PWM de 6 secuencias de Figura 4

Estas matrices también pueden ser representadas en una forma gráfica donde se visualiza las frecuencias de cada x (símbolos) en su correspondiente posición p . Esta representación es denominada *secuencia de logos* [Schneider&Stephens, 1990] donde se indica en la altura total de la pila (posición) la conservación de la frecuencia en esa posición y en la altura de una x (símbolo) indica la frecuencia relativa de x en esa posición.

En la Figura 5 se muestra la secuencia de logos correspondiente a la Tabla 2. Podemos observar que en la posición 2 solo aparecen dos símbolos (E y G) con igual frecuencia, por tanto, con igual tamaño. Por otro lado, en la posición 4, aparecen dos símbolos (F y K) con distintos tamaños puesto que F es más frecuente que K en esta posición de las secuencias.



Figura 5. Representación de Secuencia de Logo de un PWM

A partir de una matriz PWM se puede dar una puntuación a cualquier otra secuencia, midiendo su parecido con el conjunto de secuencias representado en la matriz. Esto se utiliza en estudios de alineamiento de secuencias en ADN. Como se comentó en la sección 2.1, para obtener una selección optima dentro de los diferentes alineamientos y en este caso en el prototipo, nos basamos en sistemas de puntuaciones donde un alineamiento es optimo si su puntuación es la mas grande posible.

Ahora, para obtener una puntuación a partir de una matriz M y una secuencia S de longitud w , una puntuación (M,S) se define como:

$$Score(M,S) = \sum_{p=1}^w M(p, S[p])$$

Por ejemplo, tomemos una secuencia $S' = G E F F K E$, obtenemos su puntuación con respecto a la matriz de la Tabla 2 de la siguiente forma:

$$Score(M,S') = \frac{1}{6} + \frac{3}{6} + 0 + \frac{4}{6} + 0 + \frac{3}{6} = \frac{11}{6}$$

Un prototipo lo podemos construir a partir de un PWM, por ejemplo tomando los símbolos x mas frecuentes para cada posición p , obteniendo así el prototipo $F E E F F E$, también podemos obtener la secuencia de consenso incluyendo los

símbolos x cuya frecuencia esta por encima de 0.50, $M(x,p) \geq 0.50$, con lo que tendríamos $F[EG]EFFE$ notando en este caso una igualdad entre 2 símbolos $[EG]$ lo cual se puede solucionar a través de un promedio de fondo (promedio que se estipula para un símbolo según comportamiento general, en todas las secuencias) que con base en eventos producidos se puede asignar a cada símbolo.

Si se calcula la puntuación del prototipo obtenido $S = FEEFFE$ con respecto a la matriz PWM de referencia (Tabla 2), obtenemos el siguiente resultado:

$$Score(M,S) = \frac{3}{6} + \frac{3}{6} + 1 + \frac{4}{6} + 1 + \frac{3}{6} = \frac{25}{6}$$

Se observa que el prototipo $FEEFFE$ obtenido en la matriz M de PWM (Tabla 2) su puntuación es mayor que la puntuación obtenida para la otra secuencia $S'=GEFFKE$. Esta propiedad se cumple siempre: el prototipo es la secuencia con una puntuación más alta, por tanto, la más parecida (o mejor alineada) con respecto a las secuencias de origen.

3.1.1. Otras formas de calcular la matriz PWM

Veamos ahora diferentes métodos para calcular una matriz de pesos posicionales de un determinado número de secuencias de la misma longitud.

Tomamos $n(x,p)$ como el número de ocurrencias de x en la posición p de la secuencia y N el número de secuencias.

1. **Método de frecuencia directo** [Dong&Pei, 2007]: Definimos $M(x,p)$ como la frecuencia relativa de x en la posición p de la secuencia:

$$M(x,p) = \frac{n(x,p)}{N}$$

Por ejemplo, de la lista de secuencias de la Figura 4 tomemos el símbolo G y la posición 1, tenemos que $n(x,p) = 1$ porque el símbolo G solo aparece una vez en la primera posición de las secuencias y $N = 6$ porque tenemos 6 secuencias, luego $M(x,p) = 1 / 6$.

2. **Método de registro de probabilidad** [Dong&Pei, 2007]: En este método las probabilidades de puntuación se calculan como la frecuencia observada $n(x,p)$ dividida por la frecuencia esperada del símbolo x , b_x . Es decir, se toma una frecuencia relativa de antecedentes de la base, de forma que:

$$M(x,p) = \log (P(x,p) / b_x) = \left[\log \frac{n(x,p)}{N} / b_x \right]$$

Por ejemplo, tomando como frecuencia base observada 0,33 (frecuencia de símbolo E en posición 1, Figura 4) y una estimación de 0,25 como frecuencia esperada (secuencia de fondo), la puntuación de probabilidades sería $0,33/0,25 = 1,32$. Por ultimo las probabilidades las convertimos en puntuación de registro tomando el logaritmo en base 2.

Pero debemos tener presente aquellas frecuencias donde $n(x,p) = 0$ pues el logaritmo de cero es infinito y esto crea un problema [Hertz, 1999]. Para evitar esta particularidad se plantea la siguiente ecuación:

$$M(x,p) = \log [(C_{i,j} + P_i) / ((N+1) P_i)] \approx \log (Pr(x,p) / b_x)$$

Donde $C_{i,j}$ es el de contar con base en la posición i,j , N es el numero total de secuencias, y de esta manera se evita el logaritmo de cero.

3. **Método frecuencia + similitud** [Dong&Pei, 2007]: Este método define una PWM por combinación del cálculo de frecuencias en una matriz de pesos posicionales, W , y una matriz de semejanzas por pares, Y . La matriz W puede ser una PWM definida por cualquiera de los métodos descritos anteriormente. Un ejemplo de la matriz de Y es mutacional de Dayhoff distancia de la matriz [Dayhoff, 1978]. Para cada letra $x \in A$ y posición p , el uso de Y nos permite utilizar información (probabilidad) de la frecuencia promedio entre las letras (símbolos) obtenidas fuera del rango de las secuencias dadas.

$M(x,p)$ se define entonces:

$$M(x,p) = \sum_{y \in A} Y(x,y) \times W(y,p)$$

3.1.2. Cálculo de la matriz PWM con pseudo-cuentas

Los símbolos cuya frecuencia es cero no deben ser muy determinantes porque puede ser no se haya encontrado este símbolo en esta posición solo por una cuestión circunstancial, es decir, que las secuencias consideradas no sean una buena muestra que aproxima la distribución real de los datos.

Para resolver este problema, se consideran pseudo-cuentas (PC) que mantenga una línea promedio y que de esta manera nos aproximen al manejo de frecuencias con más relación en un entorno global y no en una toma sesgada de datos.

Para ello se define la matriz $M(x,p)$ de la siguiente manera:

$$M(x,p) = \frac{n(x,p) + PC(x,p)}{N + \sum_{y \in A} PC(y,p)}$$

Donde para cada posición p y cada letra y , dado $PC(y,p) \geq 0$ sea un número, llamemos $PC(x,p)$ la variabilidad (número, probabilidad) que se deba tener presente dentro de la frecuencia global de x , y $PC(y,p)$ la suma de dichas variabilidades dentro de la posición p .

Teniendo en cuenta los datos presentados en la Figura 4, tomamos como ejemplo una variabilidad (pseudo-cuenta) $PC(x,p) = 1$ para todos los símbolos, lo que representa para $PC(y,p) = 4$. Con este caso particular, la matriz PWM que obtenemos es la siguiente:

	1	2	3	4	5	6
E	$\frac{3}{10}$	$\frac{4}{10}$	$\frac{7}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{4}{10}$
F	$\frac{4}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{5}{10}$	$\frac{7}{10}$	$\frac{2}{10}$
G	$\frac{2}{10}$	$\frac{4}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{2}{10}$
K	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{3}{10}$	$\frac{1}{10}$	$\frac{2}{10}$

Tabla 3. PWM de Figura 4 usando pseudo-cuentas

3.2. MODELO CADENA DE MARKOV

El modelo de eventos presentado anteriormente (PWM) se desarrolla bajo la percepción de una serie de eventos independientes uno del otro, es decir, el método no tiene memoria, en el sentido que las posiciones en las secuencias son completamente independientes. Para capturar la dependencia entre las posiciones, hay que añadir los estados.

El modelo cadena de Markov es una serie de eventos, en la cual la probabilidad de que ocurra un evento depende del evento inmediato anterior. En efecto, las cadenas de este tipo tienen memoria. "Recuerdan" el último evento y esto condiciona las posibilidades de los eventos futuros. Esta dependencia del evento anterior distingue a las cadenas de Markov de las series de eventos independientes; este modelo fue introducido por Markov [Seneta, 1996]. Aunque este tipo de proceso presenta una forma de dependencia simple, se ha usado mucho para modelos variables en procesos estocásticos. El método está basado en que el estado presente resume toda la información importante para descubrir su probabilidad en el futuro.

Para una determinada distribución de datos y una secuencia $S = s_1 \dots s_m$, se puede estimar la probabilidad de S ocurrida en la distribución usando lo siguiente:

$$Pr(S) = Pr(S_m | S_{m-1}..S_1) Pr(S_{m-1} | S_{m-2}..S_1) \dots Pr(S_2 | S_1) Pr(S_1).$$

Cadena de Markov de primer orden asume que el próximo estado sólo depende del estado inmediatamente anterior, pero no del resto de eventos pasados. Con esa premisa, podemos reescribir $Pr(S)$ de la siguiente manera:

$$\begin{aligned} Pr(S) &= Pr(S_m | S_{m-1}) Pr(S_{m-1} | S_{m-2}) \dots Pr(S_2 | S_1) Pr(S_1) \\ &= Pr(S_1) \prod_{i=2}^m Pr(S_i | S_{i-1}) \end{aligned}$$

Un modelo de cadena de Markov se representa con cinco variables (A , Q , $q0$, nxt , emt) que indican:

- A es un conjunto finito (el alfabeto) con símbolos (o letras).
- Q es un conjunto finito de estados.
- $q0$ es el estado de inicio.

- *nxt* es el mapeo de transición; es una función total, asignando probabilidades para estados pares en $Q \times Q$, de manera que las siguientes dos condiciones se cumplan:

(a) $0 \leq \text{nxt}(q, q') \leq 1$ para todo $q, q' \in Q$

(b) El conjunto de transición a partir de cualquier estado q define una distribución de probabilidad sobre el posible siguiente estado, es decir:

$$\sum_{q' \in Q} \text{nxt}(q, q') = 1$$

- *emt* es una función de emisión; es una función parcial de Q para A ; q es un estado de silencio si *emt*(q) no es definida, y q es un estado emisor de otra manera.

Ahora, en un ejemplo de cadenas de Markov, se toman como base los datos de la Figura 4 donde tenemos, el alfabeto $A = \{E, F, G, K\}$, los estados son $Q = \{q0, E, F, G, K\}$, con $q0$ como estado de inicio, la función emisión es $\text{emt}(Q) = A$, $q0$ es un estado de silencio. Las transiciones resultantes se representan en la Tabla 4.

<i>qo</i>	E	14/36
<i>qo</i>	F	14/36
<i>qo</i>	G	5/36
<i>qo</i>	K	3/36

E	E	3/11
E	F	4/11
E	G	2/11
E	K	2/11

F	E	5/13
F	F	5/13
F	G	2/13
F	K	1/13

G	E	4/4
G	F	0
G	G	0
G	K	0

K	E	0
K	F	2/2
K	G	0
K	K	0

Tabla 4. Transición *nxt*

Como se denoto en las formulas anteriores y la transición de q_0 (estado inicial) corresponde al número de veces que esta representado un evento en todas las secuencias dividido en el número total de todos los eventos en todas las secuencias. En el ejemplo de la Tabla 4, el evento E esta representado 14 veces en todo el conjunto de secuencias y existen 36 eventos (símbolos E, F, G, K) en todas las secuencias, con lo cual $q_0 \rightarrow E = 14/36$. Las otras transiciones corresponden al número de veces que se encuentra el par indicado en todas las secuencias dividido sobre la suma de dichos pares, para el caso del par $[E-E]=3$ indica que este par se encuentra 3 veces en todas las secuencias y la suma de pares que empiezan con el evento E ($E-E, E-F, E-G, E-K$) equivale a 11 con lo se obtiene para la transición $E \rightarrow E = 3/11$.

Las cadenas de Markov se pueden usar para definir puntuaciones donde dada una secuencia $S = s_1 s_2 \dots s_m$ y un modelo de cadena de Markov M , la puntuación (*Score*) de S contra M se define como:

$$Score(S, M) = nxt(q_0, s_1) \prod_{i=1}^{m-1} nxt(s_i, s_{i+1}).$$

Para el modelo de Markov de la Tabla 4 (transición nxt) y la secuencia $S=FEFFFE$, la puntuación esta dada por:

$$Score(S, M) = 14/36 * 5/13 * 3/11 * 4/11 * 5/13 * 5/13.$$

Donde según la secuencia S , se inicia con el símbolo F , cuya puntuación viene representada del estado inicial q_0 que corresponde a $nxt(q_0, F)=14/36$, el siguiente símbolo E forma par con su símbolo anterior F , su valor es $nxt(F, E)=5/13$, el siguiente par formado es $nxt(E, E)=3/11$ y así sucesivamente se siguen tomando parejas en orden de la secuencia y se toma el valor de cada pareja indicado en la transición nxt (Tabla 4).

Para la secuencia $S' = GEFFKE$, tenemos

$$Score(S', M) = 5/36 * 4/4 * 4/11 * 5/13 * 1/13 * 0$$

En el caso de las frecuencias observadas en cero, también se puede utilizar pseudocuentas al igual que se observo en PWM en la sección anterior.

3.3. MODELO MARKOV OCULTO (HMM)

El modelo de Markov anterior considera que cada estado corresponde a un evento físicamente observable. Los modelos de Markov ocultos o *Hidden Markov Models* (HMM) se indican como modelos doblemente estocásticos en el cual hay un proceso subyacente que está oculto. En modelos de Markov ocultos se puede asociar en un determinado estado muchos símbolos según una forma probabilística, en lugar de asociar un solo símbolo como es el caso de una cadena de Markov.

El proceso oculto de los modelos HMM es un proceso estocástico bivariado que se puede indicar como:

$$\{(o_t, s_t)\} t \in T$$

Donde $S=s_1s_2...s_T$ es un proceso observado dependiente de $O=(O_1O_2...O_T)$ que es el proceso de Markov oculto.

Así pues modelo de Markov oculto se puede representar como una 6-tupla $(A, Q, q_0, q_e, nxt, emt)$ donde:

- A es un conjunto finito (alfabeto) de los símbolos (o letras)
- Q es un conjunto finito de estados
- q_0 es el estado de inicio y q_e es el estado final
- nxt es la matriz de transición, que es una función total de la asignación de probabilidades a pares en el estado $Q \times Q$, satisfaciendo las dos condiciones siguientes:
 - (a) $0 \leq nxt(q, q') \leq 1$ para todo $q, q' \in Q$, y
 - (b) el conjunto de transiciones a partir de cualquier estado q definida una probabilidad de distribución sobre el posible próximo estado, es decir:

$$\sum_{q' \in Q} nxt(q, q') = 1$$

- emt es la matriz de emisión, que es una función parcial de la asignación de probabilidades a pares en $Q \times A$, satisfaciendo las dos condiciones siguientes:
 - (a) $0 \leq emt(q, x) \leq 1$ si es definida, para todo $q \in Q$ y $x \in A$
 - (b) para cualquier estado q tal que $emt(q, y)$ es definida para algunos $y \in A$, es el caso de que $emt(q, x)$ es definido para todos los $x \in A$ que es:

$$\sum_{x \in A} emt(q, x) = 1$$

Un estado q es un estado silencioso si $emt(q,x)$ no está definida para cualquier $x \in A$, y q es un estado emisión de otra manera.

Para aplicar este método en el proceso observado cuya matriz de transición nxt es la indicada en la Tabla 4 faltaría tener como referencia un proceso no observado (oculto). Supongamos que este suceso hace referencia las condiciones metereológicas que se dan durante las visitas de los turistas, de forma que lo podemos definir por dos estados $I = \{\text{lluvioso}(v), \text{soleado}(w)\}$ como se referencia gráficamente en la siguiente Figura 6.

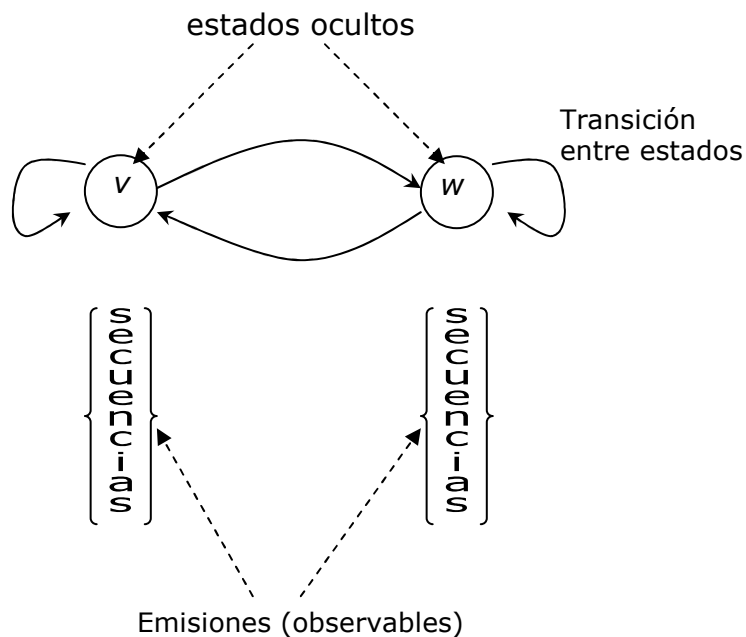


Figura 6. HMM procesos ocultos/observables

Una probabilidad de emisión con base en estos estados podría ser la que se muestra a continuación, donde a cada lugar turístico se le asocia una probabilidad de ser visitado según sea un día lluvioso o soleado.

$emt(v,E) = 0.3$
 $emt(v,F) = 0.4$
 $emt(v,G) = 0.2$
 $emt(v,K) = 0.1$
 $emt(w,E) = 0.5$
 $emt(w,F) = 0.3$
 $emt(w,G) = 0.1$
 $emt(w,K) = 0.1$

Puesto que no hay una correspondencia uno-a-uno entre la secuencia de emisión de símbolos y la secuencia de estados, habrá que distinguir las dos secuencias. En [Dong&Pei, 2007] se relacionan dos de los tres principales problemas para hacer frente a HMM de la determinación de la óptima correspondencia entre las dos secuencias. Los tres principales problemas son:

- Dado un modelo HMM y una secuencia, calcular la probabilidad de que la secuencia es emitida por el HMM. Este problema se resuelve por el procedimiento forwardbackward.
- Dado un modelo HMM y una secuencia, buscar la secuencia mas probable de estados ocultos que podría haber emitido la secuencia dada. Este problema es resuelto por el algoritmo de Viterbi.
- Dado una secuencia de salida o un conjunto de dichas secuencias, encontrar el estado de transición más probable y probabilidades de salida. En otras palabras, la formación de parámetros del HMM dado un conjunto de datos de secuencias Este problema es resuelto por el algoritmo de Baum-Welch.

3.5. ANÁLISIS DE LOS MÉTODOS PRESENTADOS

Una vez estudiados los métodos existentes de generación de secuencias consenso, se ha analizado su aplicabilidad para el caso de construir un prototipo que represente de forma adecuada la información dada por un conjunto de secuencias de eventos.

Dado que la principal aplicación que se pretende estudiar es la generación de prototipos con la finalidad de usarlos para anonimizar una base de datos, se considerará que el conjunto de secuencias es siempre pequeño. Además, debemos tener en cuenta que el conjunto de secuencias de las que se va a obtener un prototipo serán similares, puesto que son el resultado de un proceso automático de generación de clases, que agrupa siempre los elementos parecidos en una misma clase, y deja elementos diferentes en clases distintas. En este caso, los elementos a clasificar son las secuencias de eventos.

Vamos a considerar, también, que las secuencias representan trayectorias de individuos, que han realizado distintas actividades. En concreto, como ya se ha dicho, nos interesa la aplicación de estos algoritmos para el dominio turístico, donde se obtienen datos de seguimiento de turistas y de los lugares que visitan y actividades que hacen en un entorno cerrado (una ciudad, un parque temático, etc.).

3.5.1. PWM (Position Weight Matrix)

Este método calcula la frecuencia de los eventos para cada posición para así ir definiendo cada evento prototipo (el de mayor frecuencia) por posición, con base a la matriz de frecuencias PWM.

La independencia de eventos con respecto a su relación de visitas a sitios turísticos por individuo (turista) determina una ventaja en la aplicación de PWM pues se obtiene un buen sub-prototipo identificativo por posición que posteriormente la unión de estos sub-prototipos se generalizan en el prototipo final del conjunto de secuencias.

Cuando se habla de independencia de eventos, se refiere a que la unión de determinados eventos (por ejemplo unión de pares de 'ef' o 'gk') no tienen ninguna representación significativa para la toma de decisiones en la conjugación de un prototipo, lo que para el caso representa una coherencia particular de los datos que se concluye en menos pérdida de información.

Para el caso requerido, este método PWM representa un buen desarrollo aplicativo ya que sus características corresponden a las necesidades requeridas que atañen en independencia de eventos que se representan en encuestas individuales que forman las secuencias de sitios turísticos visitados como única referencia a tener presente dentro del conjunto de variables o datos identificativos.

Sin embargo, este método presenta algunas características que hacen que no sea directamente aplicable para construir un prototipo de secuencias de eventos como las que se estudian en este trabajo de máster. Concretamente existen dos condicionantes:

1. El método PWM considera que las longitudes de todas las secuencias son iguales.
2. Para una cierta posición p de las secuencias, se puede tener una misma frecuencia máxima igual para distintos símbolos. El método PWM no da ningún criterio para elegir un símbolo en caso de empates.

Después de estudiar estos dos problemas, se proponen las siguientes soluciones.

Para el primer punto, se determinan las longitudes de todas las secuencias pertenecientes a un grupo, posteriormente se halla la longitud máxima dentro de ese grupo y las secuencias que tienen menor longitud con respecto a la máxima se les agrega al final el símbolo '-' hasta que tenga la misma longitud de la secuencia determinada con mayor longitud. Como se ha mencionado, la

longitud de una secuencia corresponde al número de eventos que tiene dicha secuencia.

Por ejemplo, dado un grupo g de tres secuencias $g=[\{efegk\},\{ef\},\{fek\}]$ las longitudes de las secuencias corresponden a $[(5),(2),(3)]$ definiendo así la primera secuencia como longitud mayor o máxima ($l=5$) luego las otras dos secuencias deben llenarse con el símbolo '-' para que así todas las secuencias queden de igual longitud $l=5$ con lo cual se precisan las secuencias de menor longitud con respecto a la mayor como $\{e;f;-;-;- \}$, $\{f;e;k;-;- \}$ quedando definido el grupo g en la matriz PWM $g=[\{efegk\},\{ef---\},\{fek--\}]$, dejando de esta manera todas las secuencias del grupo g de igual longitud (tamaño) para ser procesadas.

Para el segundo punto, dado el caso que en una posición p las frecuencias de algunos eventos sean iguales, el usuario deberá definir un orden de prevalencia (prioridad) sobre los símbolos del alfabeto A , que identifica los distintos eventos. De esta manera para los eventos ordenados $\{e,f,g,k\}$, si en alguna posición p de la matriz PWM tienen la misma frecuencia el evento e con el evento f , por ejemplo, el evento e va ser el escogido, pues en el orden definido aparece primero el evento e que el evento f . Con esta heurística, se puede interpretar un orden de relevancia entre eventos definiendo por el usuario en cada aplicación concreta del algoritmo de generación de prototipos.

3.5.2. Cadenas de Markov

Tal como se ha dicho en el apartado anterior, el método PWM está especialmente indicado cuando hay independencia entre los eventos. Sin embargo, no garantiza dar buenos resultados cuando hay una representación significativa de la unión de determinados eventos. Por ejemplo en la predicción de genes de secuencias de DNA donde una región (unión de eventos) puede codificar para una proteína y/o sus correspondientes transcripciones o descubrir que proteínas tienen un origen común; como en estas indicaciones existen mas (y otros casos) donde la dependencia de eventos juegan un papel relevante en la definición de un prototipo. También existen casos donde la probabilidad de un evento depende de otro evento anterior o que la representación de un conjunto de variables dependa de la interacción que pueda tener con otro conjunto de variables que sea determinativo a la hora de hallar probabilidades, como el caso indicado en la sección 3.3 donde la probabilidad de que ocurra un evento interactúa o depende de otro conjunto de variables que son representados en proceso de estados tales como $\{\text{lluvioso}, \text{soleado}\}$ dados en el ejemplo de dicha sección.

En cambio, las cadenas de Markov precisamente permiten representar estas relaciones semánticas entre eventos.

Para generar un modelo de Markov partimos como base de un estado inicial q_0 de eventos, el cual se determina a partir de las frecuencias de cada evento de la posición 1 de un grupo g de secuencias, seguidamente se establecen las transiciones resultantes del conjunto de secuencias a operar, estas transiciones corresponden a la representación de la combinación (unión) de determinados eventos (que para esta aplicación se toma por pares de eventos $\{(ee),(ef)...\}$) y su frecuencia que corresponde a la cantidad de veces que se encuentra esta combinación (por ejemplo 'ef') en todas las secuencias del grupo relacionado.

Generalmente el estado inicial q_0 parte de las frecuencias resultantes por evento con relación a todos los eventos de todas las secuencias del grupo, es decir, la sumatoria (numero de representaciones de dicho evento en todo el grupo de secuencias) de un evento x sobre el número total de eventos que hay en todas las secuencias del grupo.

$$\sum x_g / \sum A_g$$

Donde x_g corresponde a la cantidad de veces que se encuentra representado un evento x en un grupo g , y A_g corresponde al total de eventos del alfabeto A que hay en el grupo g .

El problema de estado inicial q_0 tomado de la especificación anterior generalmente indicada para este estado q_0 , surge en que un evento x que se encuentra representado mayoritariamente en todo el grupo g , no se encuentre como frecuencia mayor en la primera posición del grupo y esto representaría que este evento x sea el primer evento indicado como prototipo de esta posición sin realmente serlo, en la Figura 7 se denota un ejemplo representativo para este caso.

e;f;g;f;k;f;f
e;g;f;f;k;f

Figura 7. Ejemplo de dos secuencias que empiezan por 'e'

En el ejemplo se observa que en la primera posición de este grupo g de 2 secuencias, el evento x con mayor frecuencia corresponde al evento 'e' que

sería el prototipo relativo para esta posición, pero tomando como estado inicial q_0 ($\sum X_g / \sum A_g$) el evento x con mayor frecuencia corresponde al evento ' f ' el cual no correspondería a un prototipo relativo para esta primera posición del grupo g indicado en este ejemplo.

Después de estudiar este problema, se ha diseñado una heurística que permite iniciar la cadena de Markov con un estado inicial más adecuado. Se propone calcular el estado inicial q_0 con las frecuencias resultantes por evento calculadas únicamente a partir de la primera posición de las secuencias a operar.

Sin embargo, el problema descrito también puede tener efecto en la referencia de transiciones creadas (combinación de pares de eventos). En el caso del estudio de trayectorias de individuos, puede que ciertas transiciones entre eventos tengan un significado para ciertos estudios. En tal situación, las cadenas de Markov podrán modelar estas transiciones en el prototipo obtenido.

3.5.3. Modelo Markov Oculto Hmm

Para esta aproximación se consideran unas secuencias de variables (símbolos) que no son independientes, sino que el valor de cada símbolo depende de los elementos que previamente han aparecido en las secuencias o en su caso valores dependientes de un proceso oculto. Por tanto, este modelo no es adecuado para el uso en métodos de protección de datos puesto que en los ficheros de datos a ocultar raramente se dispone de información de algún otro proceso oculto que se pueda relacionar con los datos de la secuencia. En este trabajo vamos a suponer que los eventos de las secuencias son independientes por individuo y no existe relación entre una secuencia y otra, debida algún proceso oculto.

Por este motivo descartamos este algoritmo para la comparación en el ámbito del estudio de trayectorias en el sector turístico.

4. NUEVAS PROPUESTAS

En esta sección se presenta una nueva propuesta para la construcción de un prototipo de un grupo de secuencias de eventos más apropiada para el tipo de secuencias que se han presentado en esta tesis de master, por ejemplo, las trayectorias de turistas dentro de un recinto acotado.

Recordemos que en los métodos de protección de la privacidad por microagregación, el último paso consiste en calcular un centroide o prototipo para cada uno de los clusters obtenidos en el paso anterior. Estos prototipos son los que se utilizan para enmascarar los datos originales y generar una nueva versión que no permita la re-identificación de los individuos. En el caso que nos ocupa, todos los individuos de un mismo cluster sustituyen sus respectivas secuencias de eventos por una secuencia prototipo.

Estas secuencias están formadas por un conjunto de eventos que se han producido en un cierto orden que es relevante para el estudio posterior de los datos. Sin embargo, no es tan relevante la posición absoluta dentro de la secuencia, como el orden relativo y el número de eventos en común en las secuencias que forman un mismo grupo.

A continuación se presenta el método OESP que se ha desarrollado en el grupo ITAKA [Valls&Gómez, 2008]. Después se presentan diversas mejoras al esquema básico del método **Ordered Element Scoring Prototyping** (OESP) con el fin de mejorar los resultados obtenidos y generar prototipos más representativos de las secuencias originales.

Además de las mejoras que se presentan al esquema básico del método OESP, también se presenta una modificación de mejora sobre la base del esquema Position Weight Matrix (PWM).

4.1. EL MÉTODO OESP (Ordered Element Scoring Prototyping)

Para el diseño de este algoritmo se han tenido en cuenta dos factores. Por un lado, en una secuencia, el orden de consecución en que se presentan los eventos es representativo, por lo que se asignan distintas puntuaciones a los símbolos (eventos) según su posición. Estas puntuaciones son acumulables por todas las secuencias del grupo, de forma que por otro lado, estamos teniendo

en cuenta el número de apariciones de cada símbolo en las secuencias del cluster. La puntuación total acumulada por eventos se toma como base para la formación del prototipo.

A continuación presentamos la formalización del método OESP.

Siendo $c(G\{i\})$ el prototipo de un cluster de secuencias categóricas $G\{i\}$, se considera que el prototipo debe cumplir las siguientes propiedades:

- $c(G\{i\})$ es una secuencia categórica.
- $c(G\{i\})$ solamente contiene símbolos (por ejemplo, etiquetas categóricas) que se pueden encontrar en $G\{i\}$.
- Los símbolos en $c(G\{i\})$ aparecen en un orden similar que los encontrados en las secuencias de $G\{i\}$.
- La longitud de $c(G\{i\})$ es similar a la media de las longitudes de las secuencias de $G\{i\}$.

El método propuesto para la generación del prototipo se basa en el cálculo de una tabla de puntuaciones **Element Scoring Table** (EST), donde los elementos pueden ser elementos individuales o patrones de eventos secuenciales considerados como unidades indivisibles.

Considerando cada elemento como un símbolo asociado al lenguaje L , el método OESP consiste en las siguientes etapas:

1. Los símbolos de $s \in L$ se transforman en $s' \in L'$, mediante la adición a cada símbolo (evento) s de un número, n , que indica su posición relativa en la secuencia. La primera vez que un símbolo aparece, se le asigna el número $n=1$, la segunda vez $n=2$, y así sucesivamente.

Esta recodificación permite distinguir la repetición de los elementos que se encuentren en una misma secuencia como elementos diferentes para poder evaluarlos correctamente y de forma que la secuencia prototipo también disponga de elementos repetidos.

Por ejemplo, si se dispone de un cluster con cinco secuencias $G=\{abc,ac,ab,cbac,cab\}$, cada uno de ellos se codificará de la forma siguiente:

$$\begin{aligned}r(abc) &= \{a1,b1,c1\} \\ r(ac) &= \{a1,c1\} \\ r(ab) &= \{a1,b1\}\end{aligned}$$

$$r(cbac) = \{c1, b1, a1, c2\}$$

$$r(cab) = \{c1, a1, b1\}$$

- Identificación de todos los símbolos L' que aparecen en r . Estos símbolos forman el conjunto $E \subset L'$.

Siguiendo con el ejemplo, la lista de símbolos diferentes es $E = \{a1, b1, c1, c2\}$.

- Construcción de la tabla *Element Scoring Table* (EST), con los símbolos de E y sus correspondientes puntuaciones. Para cada símbolo, se le van asignando diferentes puntuaciones según la posición en la que aparece en cada registro. Siendo p la posición del símbolo en uno de los registros, su resultado se incrementará en x puntos, proporcionalmente al tamaño del cluster $G\{i\}$:

$$x = |G_i| - p + 1$$

En el ejemplo, se obtiene el resultado en la Tabla 5.

Id	Suma	Resultado
a1	4+4+4+2+3	17
b1	3+3+3+2	11
c1	2+3+4+4	13
c2	1	1

Tabla 5. Resultado de la suma de las puntuaciones de las secuencias

- Ordenación decreciente de todos los elementos de E con base a su puntuación (frecuencia).

La tabla ordenada se muestra en la Tabla 6.

Id	Resultado
a1	17
c1	13
b1	11
c2	1

Tabla 6. Registros ordenados por puntuación

5. Creación del prototipo concatenando los símbolos iniciales siguiendo el orden obtenido en el paso anterior. Es decir, el símbolo con mayor puntuación aparece en la primera posición del prototipo.

Posteriormente, se selecciona un segundo símbolo para añadirlo al prototipo. La selección del símbolo depende de la naturaleza de los elementos. Se distinguen dos formas:

- **Elementos simples:** los símbolos se adicionan según el orden decreciente obtenido en los resultados de la tabla EST. En este ejemplo, el prototipo es *acbc*.
 - **Patrones:** el siguiente símbolo se fusiona al final del prototipo. Si el patrón tiene t eventos, el siguiente elemento de E debe compartir los $t-1$ símbolos previos con la última parte del prototipo. Por ejemplo, considerando que E tiene 5 patrones de símbolos en el siguiente orden decreciente: *dae*, *abe*, *ebc* y *aeb*. El prototipo inicial es *dae*. El siguiente símbolo debe comenzar por *ae* para fusionarse correctamente con la secuencia. Siguiendo el orden, se elige el símbolo *aeb*, y el prototipo se convierte en *daeb*. En el siguiente paso, se selecciona *ebc*, obteniendo *daebc*. Esta operación es necesaria para poder seguir manteniendo la correlación de orden que se ha deseado valorar inicialmente. En el caso de que no exista un símbolo que comparta los elementos requeridos y E todavía disponga de elementos no adicionados al prototipo, se añade el primero con mayor puntuación. Es decir, se fuerza la creación de un patrón no existente realmente, pero resultando imprescindible esta fusión si no se dispone de patrones con elementos simples comunes o si no quedan elementos no relacionados todavía por añadir. En el anterior ejemplo, el prototipo final sería *daebcabe*.
6. Reducción del prototipo para ajustarlo a la longitud media de las secuencias del cluster. Los elementos del final de la secuencia prototipo se eliminan hasta alcanzar la longitud deseada.

Para el cluster G , la longitud media es 3, así que el prototipo candidato *acbc* se reduce al prototipo $c(G)=acb$.

4.1.1 Consideraciones sobre el método OESP

La idea de este método es dar puntuaciones más altas a los eventos que aparecen siempre en las primeras posiciones de las secuencias, mientras que

los que siempre aparecen en posiciones finales de las secuencias, van a tener menor puntuación acumulada. Si al construir el prototipo se sigue el orden decreciente en las puntuaciones, el resultado esperado es reconstruir una secuencia prototipo que tendrá al principio los eventos que han aparecido en las primeras posiciones, al final, los que han aparecido en posiciones finales siempre, y en medio los que han obtenido puntuaciones intermedias. Por lo tanto, cabría esperar que la secuencia prototipo representara de forma adecuada la información de las secuencias originales.

Sin embargo, después de hacer un análisis de este método a partir de los resultados en distintos conjuntos de secuencias sintéticas, se observa que este método produce unas secuencias que mantienen el orden original de los eventos. Se puede dar el caso de que la secuencia prototipo tenga como evento inicial un evento que no se ha dado en primer lugar en ninguna de las secuencias tomadas como referencia. Por este motivo se ha decidido proponer distintas versiones de este método para intentar mejorar los resultados.

Por otro lado, este método presenta diversas ventajas respecto a los anteriores (PWM y Cadenas de Markov). OESP realiza un análisis de los eventos respecto a sus posiciones en las secuencias pero sin limitarse a una posición concreta o a una ventana de n pasos anteriores, sino que los eventos se puntúan respecto a cualquiera sea su posición en las secuencias. Además, se tiene en cuenta el orden relativo entre ellos dando distintas puntuaciones. Se trata de una aproximación más global al problema, mientras que los métodos tradicionales tienen un enfoque local, más focalizado (construyen la secuencia prototipo a base de analizar pequeños fragmentos de las secuencias).

4.2. OESP CON NUEVAS PUNTUACIONES: NP

En la representación del prototipo generado por el método OESP se observa que en algunos casos los eventos iniciales de los prototipos no son representativos del conjunto de secuencias, es decir, el elemento inicial de la secuencia no es un evento que aparezca al inicio de ninguna de las secuencias tomadas como base. Esto se presenta porque la diferencia entre los puntos otorgados a posiciones consecutivas es de 1 unidad, y por tanto, puedes acumular más puntos si estas muchas veces en la posición 2, y en cambio el evento de la posición 1 cambia de una secuencia a otra.

Ilustramos este caso con el siguiente ejemplo: Si tomamos el grupo de secuencias de la Figura 8, la puntuación obtenida con el método OESP corresponde a $\{b_1=11, a_1=10, c_1=9, a_2=5\}$, con lo que obtenemos como prototipo resultante ' *baca* ' donde se observa que el evento inicial (posición 1, evento *b*) del prototipo no es representativo en esa posición para el conjunto

de secuencias. Esto ocurre porque el evento 'b' obtiene mayor puntuación debido a que la diferencia entre posiciones que equivale a 1 no le es suficiente al evento 'a' para que se mantenga con mayor puntuación y de esta manera se ubique en la primera posición del prototipo resultante.

a;b;a;b
c;b;c
a;c;b;a;c

Figura 8. Grupo de secuencias para ejemplos de alternativas OESP

En las puntuaciones del método OESP explicado en la sección anterior (Sección 4.1) se toma como punto de partida para la primera posición puntuarla con la longitud máxima del grupo de secuencias y consecutivamente la siguiente posición se puntúa restando de uno el valor dado a la posición anterior.

$$(p_1 = \max_lon, p_2 = \max_lon - 1, p_3 = \max_lon - 2, \dots, p_{\max_lon} = 1)$$

$$p_i = n - i + 1$$

Sea p_i la puntuación de un evento en la posición i de la secuencia y $n = \max_lon$ la longitud máxima dentro de un grupo de secuencias.

Así pues estudiamos este efecto de la puntuación en el método OESP y decidimos proponer una modificación sobre la puntuación para efectos de mitigar este efecto. Se propone puntuar de manera diferente los eventos con respecto a su posición, de tal manera que se obtenga mayor y heterogénea diferenciación entre las puntuaciones de los eventos por posición.

A continuación se presentan una serie de alternativas hacia nuevas puntuaciones que permitan dar diferencias (valores) mayores entre las posiciones de los eventos consecutivos, de tal manera que los eventos en la posición $i+1$ necesiten notable representación en esta posición para superar puntuaciones de un evento de la posición i . Esta idea pretende lograr un prototipo más representativo al conjunto de secuencias de un grupo tratado.

Para explicar las distintas heurísticas propuestas se va a usar como ejemplo base el grupo de secuencias de la Figura 8. La puntuación OESP para las posiciones i de este grupo de secuencias corresponde a:

$$p_1 = 5$$

$$p_2 = 4$$

$$\begin{aligned}p_3 &= 3 \\p_4 &= 2 \\p_5 &= 1\end{aligned}$$

Sobre esta hipótesis de mejora de la puntuación para mejorar el prototipo, se presentan las siguientes alternativas:

- a) *Alternativa 1*: Sumar a las puntuaciones dadas en OESP un valor k_g igual al número de secuencias del grupo mas uno (num_sec+1). Se propone este valor como interpretación de que la diferenciación se tome en relación al número de secuencias existentes en el grupo.

$$p_i = k_g + (n - i + 1)$$

Para el ejemplo, tomamos el grupo de secuencias indicado en la Figura 8 cuya variable k_g equivalente en este grupo a 4.

La puntuación obtenida para las posiciones p_i correspondiente a esta alternativa es:

$$\begin{aligned}p_1 &= 9 \\p_2 &= 8 \\p_3 &= 7 \\p_4 &= 6 \\p_5 &= 5\end{aligned}$$

La sumatoria de puntuaciones por evento (dado con sub-indicador como lo indica el método OESP) es:

$$a_1=18 ; b_1=23 ; c_1 = 17 ; a_2 = 13 ; b_2 = 6 ; c_2 = 12$$

Al ordenar los eventos y obtener los 4 primeros (según longitud media del grupo de secuencias) el prototipo obtenido es 'baca'. El prototipo no cambia, puesto que en realidad solo se ha realizado un desplazamiento entre las puntuaciones pero no ha variado su relación entre ellas (la diferencia sigue siendo de una unidad).

- b) *Alternativa 2*: Multiplicar las puntuaciones dadas en OESP por una constante k . Esta constante puede ser dada bajo la característica de diferenciación que se quiera otorgar para diferenciar las posiciones. De esta forma a mayor k , mayor puntuación se le otorga a una posición concreta.

$$p_i = k * (n - i + 1)$$

Para el ejemplo, tomamos el grupo de secuencias indicado en la Figura 8 y una constante $k=10$, la puntuación obtenida para las posiciones p_i correspondiente a esta alternativa es:

$$\begin{aligned}p_1 &= 50 \\p_2 &= 40 \\p_3 &= 30 \\p_4 &= 20 \\p_5 &= 10\end{aligned}$$

Si aplicamos esta puntuación a los datos del ejemplo, la sumatoria de puntuaciones por evento es:

$$a_1=100 ; b_1=110 ; c_1 = 90 ; a_2 = 50 ; b_2 = 20 ; c_2 = 50$$

Al ordenar los eventos y obtener los 4 primeros (según longitud media del grupo de secuencias) el prototipo obtenido es '*baça*'. En este caso, observamos que seguimos obteniendo el mismo prototipo que con el método original de puntuación.

- c) Alternativa 3: Elevar un valor k (que corresponde al número de secuencias + 1) a la posición menos 1. El exponente se toma de esta manera para que su última posición corresponda a k^0 (exponente cero) y obtengamos una puntuación de uno en la última posición de la secuencia más larga, con lo que la puntuación queda definida de la siguiente forma:

$$p_i = k^{(n-i)}$$

Para el ejemplo, tomamos el grupo de secuencias indicado en la Figura 8 cuya variable k es equivalente en este grupo a 4. La puntuación obtenida para las posiciones p_i correspondiente a esta alternativa es:

$$\begin{aligned}p_1 &= 256 \\p_2 &= 64 \\p_3 &= 16 \\p_4 &= 4 \\p_5 &= 1\end{aligned}$$

La sumatoria de puntuaciones por evento (dado con sub-indicador como lo indica el método OESP) es:

$$a_1=512 ; b_1 = 144 ; c_1 = 320 ; a_2 = 20 ; b_2 = 4 ; c_2 = 17$$

Al ordenar los eventos y obtener los 4 primeros (según longitud media del grupo de secuencias) el prototipo obtenido es ahora diferente *'acba'*.

- d) Alternativa 4: Elevar la posición a una constante k , esta constante k cumple con la característica de diferenciación que se quiera otorgar para destacar mayor representación por posición entre los eventos, donde k a mayor valor mayor interpretación por representatividad de posición.

$$p_i = (n - i + 1)^k$$

Para el ejemplo, tomamos el grupo de secuencias indicado en la Figura 8 y una constante $k=2$. La puntuación obtenida para las posiciones p_i correspondiente a esta alternativa es:

$$\begin{aligned} p_1 &= 25 \\ p_2 &= 16 \\ p_3 &= 9 \\ p_4 &= 4 \\ p_5 &= 1 \end{aligned}$$

La sumatoria de puntuaciones por evento (dado con sub-indicador como lo indica el método OESP) es:

$$a_1=50 ; b_1 = 41 ; c_1 = 41 ; a_2 = 13 ; b_2 = 4 ; c_2 = 10$$

Al ordenar los eventos y obtener los 4 primeros (según longitud media del grupo de secuencias) el prototipo obtenido es *'abca'*.

Para interpretar y comparar estas alternativas propuestas con respecto al método OESP, se presenta la figura 9 donde se observa el grupo de secuencias tratado y los prototipos obtenidos.

Grupo de secuencias a b a b c b c a c b a c	<table> <tr> <th>Método</th><th>Prototipo</th></tr> <tr> <td>OESP</td><td><i>'baca'</i></td></tr> <tr> <td>OESP-NP(a)</td><td><i>'baca'</i></td></tr> <tr> <td>OESP-NP(b)</td><td><i>'baca'</i></td></tr> <tr> <td>OESP-NP(c)</td><td><i>'acba'</i></td></tr> <tr> <td>OESP-NP(d)</td><td><i>'abca'</i></td></tr> </table>	Método	Prototipo	OESP	<i>'baca'</i>	OESP-NP(a)	<i>'baca'</i>	OESP-NP(b)	<i>'baca'</i>	OESP-NP(c)	<i>'acba'</i>	OESP-NP(d)	<i>'abca'</i>
Método	Prototipo												
OESP	<i>'baca'</i>												
OESP-NP(a)	<i>'baca'</i>												
OESP-NP(b)	<i>'baca'</i>												
OESP-NP(c)	<i>'acba'</i>												
OESP-NP(d)	<i>'abca'</i>												

Figura 9. Grupo de secuencias + prototipos obtenidos

Dentro de las alternativas presentadas y viendo los prototipos obtenidos, al compararlos con el prototipo del método OESP, vemos que no hubo cambios para las alternativas OESP-NP(a) y OESP-NP(b) pero esto surge porque para conjuntos de secuencias pequeño con longitudes pequeñas no hay bastantes eventos que representen variación suficiente como para registrar modificaciones en los prototipos resultantes. Aun así, las alternativas OESP-NP(c) y OESP-NP(d) dentro de sus puntuaciones si alcanzan a registrar diferenciaciones posicionales con respecto a los eventos iniciales del conjunto de secuencias tratado, observando así prototipos diferentes de estas alternativas con respecto al prototipo OESP, consiguiendo de esta manera el objetivo planteado de mayor y heterogénea diferenciación entre posiciones a fin de obtener eventos mas representativos al conjunto de secuencias.

Entre estos dos prototipos obtenidos, el prototipo OESP-NP(d) tiene una mejor representación pues sus eventos iniciales 'ab', que son mas representativos que los eventos iniciales del prototipo OESP-NP(c) 'ac', ya que si vemos estas posiciones en el grupo de secuencias, los eventos con mayor representación en la posición 1 y posición 2 corresponde a los eventos 'a' y 'b' respectivamente.

Después de analizar estos resultados y observar la distribución de puntuaciones en las secuencias, tomamos como mejor heurística la de la alternativa 4, OESP-NP(d), donde se observa que dentro de las puntuaciones referenciadas a cada posición, la diferencia entre la primera posición y la segunda es de 9, la diferencia entre la posición2 y la posición3 es de 7, la diferencia entre la posición3 y la posición4 es de 5 y por ultimo la diferencia entre la posición4 y la posición5 es de 3; con lo que se evidencia mayor y heterogénea diferenciación (de mayor a menor) entre posiciones.

Por tanto, para el estudio comparativo posterior con datos reales de itinerarios de turistas, se considerará solo la alternativa OESP-NP(d) denominándola simplemente OESP-NP: OESP con Nueva Puntuación.

4.3. OESP SIMÉTRICO: OESP-EM

En el método OESP básico en el proceso de aunar y ordenar (de mayor a menor) los eventos puntuados y así representar el prototipo resultante, la longitud del prototipo se basa en la media entre los tamaños (longitudes) de las secuencias del grupo, con respecto a esta media, se eliminan los elementos que se encuentran después del valor medio obtenido, con ello se observó que se pierde representación con respecto a los elementos finales del conjunto de secuencias, y notándose que generalmente los elementos medios de las secuencias son variantes o no tienen representaciones comunes como el que se asemejan en los inicios o finales de las secuencias.

Con respecto a esta determinación, se ha modificado el proceso de obtención del prototipo de OESP eliminando elementos del medio de los eventos puntuados, en lugar de eliminar los eventos finales, hasta obtener el tamaño del prototipo con base en la longitud media calculada.

Dado un grupo de dos secuencias $G=\{(a\ c\ b\ c), (a\ a\ b\ b)\}$ con el método OESP la puntuación para la consecución del prototipo corresponde a 'abcabc' con lo que se obtiene un prototipo de longitud=6 y la media para este grupo corresponde a 4, por consiguiente en el método OESP se eliminan los dos elementos finales para conseguir el prototipo con la longitud media igual a 4, el prototipo obtenido es 'abca' donde los elementos eliminados 'bc' corresponderían a una mejor representación de los elementos finales del conjunto de secuencias.

Sin embargo, con la versión simétrica de eliminación de elementos medios (OESP-EM) se propone eliminar los elementos medios de la puntuación para la consecución del prototipo, con lo que el prototipo (eliminando los 2 elementos del medio 'ca') corresponde a 'abbc' cuyos elementos 'bc' dejados tienen mejor representación al conjunto de secuencias. De esta forma se pretende obtener prototipos que sean, por lo menos, representativos de los eventos iniciales de la secuencia y también de los finales. Esto puede ser interesante si se han de realizar estudios sobre cuáles son los inicios de los itinerarios de los turistas, o cuáles son los lugares que visitan al final de su trayecto.

4.4. MÉTODO POR-PARES

La determinación del prototipo en el método PWM (*Position Weight Matrix*) viene por la posiciones individuales dentro de la secuencia, donde el elemento con mayor frecuencia (numero de veces que se encuentra un evento en determinada posición) en cada posición, se registra para la construcción del prototipo.

Después de analizar este método y compararlo con el de las cadenas de Markov, se ha definido una nueva aproximación que permita relacionar los eventos con sus predecesores, de forma que no se traten como elementos aislados en la secuencia.

El esquema base del método POR-PARES corresponde al método PWM, pero los valores de las frecuencias se toman armando pares de posiciones y no una única posición como lo realiza el método PWM, esto con el fin de valorar pares de eventos como transiciones de desplazamiento del turista.

En este caso, tal como se comentó al analizar el PWM, si se presentan frecuencias iguales entre pares de posiciones, esta situación se resuelve tomando una matriz inicial que registra (por pares de eventos) todas las parejas de eventos que existen en todas las secuencias y a partir de esta matriz inicial se toma entre las parejas con igual frecuencia la que este mas representativa en todo el conjunto de secuencias. La matriz inicial que se toma en el caso de desempatar pares con igual frecuencia es equivalente a la matriz de transición que se utiliza en cadenas de Markov (Sección 3.2). De esta forma se combinan las dos aproximaciones para sacar mejor provecho de las ventajas de ambos enfoques.

Dado el siguiente grupo de secuencias:

a;c;b;a
a;c;a;b
c;b;b;a
c;b;a;b

Figura 10. Ejemplo representativo de método POR-PARES (base PWM)

Para este grupo de secuencias el prototipo generado por el método PWM corresponde a *'abaa'* que equivale al registro del evento con mayor frecuencia por posición. En cambio, si separamos las posiciones por pares (1-2 y 3-4) para el primer par de posiciones la pareja de eventos con mayor frecuencia es *'cb'*, para el segundo par de posiciones se encuentran con igual frecuencia las parejas *'ba'* y *'ab'* pues ambas se encuentran dos veces en ese par de posiciones (3-4); esta situación se resuelve tomando la pareja *'ba'* porque tiene mayor representación en todo el conjunto de secuencias, con lo que el prototipo conseguido es *'cbba'*.

Comparando los prototipos *'abba'* y *'cbba'* con las secuencias originales de la Figura 9, se observa que la segunda representa mejor al grupo que la primera.

5. DESARROLLO DE UN ENTORNO DE EVALUACIÓN

A fin de valorar el método que mejor aplique la representación de un prototipo que conjugue un grupo de secuencias que para el caso, cada secuencia representa un conjunto de sitios turísticos visitados por un individuo donde cada evento (símbolo) de la secuencia corresponde precisamente a un sitio turístico, se desarrollo y evaluó cada método los cuales generan prototipos y así evaluar los resultados. Estos prototipos se generan con base en un esquema representado en un archivo plano (txt) como el indicado en la Figura 11 (ejemplo).

efgk
4
e;f;g;k;f;f;e;k
e;k;g;e
e;k
k;k;f;g;e;g
2
f;g;g
e;f;g;e;k
3
e;f;g;g;e;k
e;g;f;k
g;k;f;f;e
0

Figura 11. Esquema conjuntos de secuencias a representar por prototipos

En este esquema base referenciado, la primera línea indica los eventos (sitios turísticos) a tener presente, que son los posibles destinos seguidos por un individuo (turista) indicados en una secuencia, los números (por ejemplo en la línea 2, número 4) indican el numero de secuencias correspondientes a un grupo del cual se va obtener un prototipo, el ultimo número (0) indica el fin del esquema. En las secuencias, cada evento esta separado por ";" cuya línea (por ejemplo línea 4, e;k;g;e) representa los sitios seguidos por un individuo (turista).

5.1. PWM (Position Weight Matrix)

Como lo indica este método, se halla la frecuencia de los eventos para cada posición para así ir definiendo cada evento (el de mayor frecuencia) por posición con base en la matriz de frecuencias desarrollada por grupo. Los pasos (algoritmo) que se codificaron con base en PWM, fueron (de manera generalizada):

1. Leer archivo .txt (esquema base) y capturarlo en un vector relacionando posición del vector por línea del archivo leído.
2. Almacenar en un vector los eventos a tener presentes indicados en la primera posición del vector anteriormente capturado, agregando como evento anexo el símbolo "-" que nos representa las posiciones donde no se produce ningún evento.
3. Capturar posiciones del primer vector (paso 1) donde se indican los números que identifican el numero de secuencias a trabajar por grupo.
4. Con base en el número de secuencias a trabajar por grupo (definidos en paso 3) formar y almacenar en un arreglo (matriz *eventos x posición*) la matriz PWM que define el grupo de secuencias a desarrollar.
5. Calculando la secuencia (longitud) mas larga del grupo (conjunto de secuencias) a trabajar, dejar todas las secuencias (de la matriz formada en paso 4) de igual tamaño agregando el parámetro "-" alineando dichas secuencias con base en la secuencia mas larga del grupo.
6. Con respecto a la matriz PWM formada y alineada de igual tamaño con respecto a posiciones p (columnas) para cada secuencia s (fila), ir calculando y almacenando por posición p la frecuencia de cada evento relacionado en el vector del paso 2.
7. Con base en las frecuencias calculadas por posición p para cada evento x , tomar el evento x con mayor frecuencia (calculadas en el paso 6) e ir formando el vector prototipo que se va definiendo por posición p , y así se va obteniendo el prototipo hasta llegar a la ultima posición de la matriz PWM con la que tenemos el prototipo final del grupo de secuencias trabajado.
8. Al obtener el prototipo final de un grupo G , borrar los eventos de la matriz PWM para que se forme nuevamente dicha matriz con respecto a un nuevo grupo de secuencias a generar el prototipo.
9. Se retorna al paso 3 y se realiza nuevamente la generación de un prototipo para otro grupo de secuencias, esto se realiza mientras el valor de este vector del paso 3 sea diferente de 0 (cero), que es el valor que nos concluye la finalización del esquema de grupos de secuencias planteado.
10. Con la indicación de finalización, se crea un archivo plano (txt) y se imprime en este los prototipos generados.

Con respecto a lo planteado y con base en el ejemplo de la Figura 11 (se vuelven a mostrar en la Figura 12), los prototipos generados fueron los siguientes, Figura 12.

efgk	
4	
e;f;g;k;f;f;e;k	
e;k;g;e	
e;k	
k;k;f;g;e;g	
2	
f;g;g	
e;f;g;e;k	
3	
e;f;g;g;e;k	
e;g;f;k	
g;k;f;f;e	
0	

PROTOTIPOS

ekg-----
efg--
efffe-

Figura 12. Prototipos generados PWM

Dado el caso que en una posición p las frecuencias de algunos eventos sean iguales, se toma como punto de partida o como punto de prevalencia el orden indicado de los eventos descritos en la primera línea del esquema a desarrollar (Figura 11), de tal manera que para los eventos dados en este esquema $\{e, f, g, k\}$ si en alguna posición p de la matriz PWM tienen la misma frecuencia el evento e con el evento f por ejemplo, el evento e va ser el indicado pues en el orden definido aparece primero el evento e que el evento f , de esta manera se puede interpretar un punto de relevancia entre eventos definiendo el orden que se requiera en el conjunto de eventos a manejar.

5.2. CADENAS DE MARKOV

Con base en el esquema de la Figura 11 y la implementación de un estado inicial q_0 y las transiciones correspondientes a la combinación de pares de eventos, en la aplicación de este método los pasos (algoritmo) que se codificaron fueron (de manera generalizada):

1. Leer archivo .txt (esquema base) y capturarlo en un vector relacionando posición del vector por línea del archivo leído.
2. Almacenar en un vector los eventos a tener presentes indicados en la primera posición del vector anteriormente capturado, agregando como evento anexo el símbolo "-" que nos representa las posiciones donde no se produce ningún evento.
3. Capturar posiciones del primer vector (paso 1) donde se indican los números que identifican el numero de secuencias a trabajar por grupo.
4. Con base en el numero de secuencias a trabajar por grupo (definidos en el paso 3) almacenar en un vector las secuencias, donde se relaciona posición del vector por secuencia.
5. Determinar la longitud de mayor tamaño (secuencia mas larga) para que con respecto a esta todas las secuencias queden de igual longitud donde se agrega el parámetro "-" y así alinearlas a fin de que todas queden de igual longitud.
6. Crear estado inicial q_0 en un vector el cual indicara el inicio a seguir dentro de las transiciones (frecuencia de pares de eventos) a crear.
7. Crear y almacenar en una matriz las transiciones que equivalen a la combinación de pares de los eventos almacenados en el vector del paso 2.
8. Calcular la frecuencia por transición creadas en el paso 7 que surge de la representación que tenga cada transición en cada una de las secuencias del grupo.
9. Ir calculando mayor frecuencia por transición teniendo presente que el paso a cada conjunto de transiciones lo proporciona el segundo evento de una transición (par de eventos) indicada como mayor frecuencia.
10. Almacenar en un vector las correspondientes transiciones de mayor frecuencia para ir creando el prototipo por grupo.
11. Al obtener el prototipo final de un grupo g , borrar las secuencias del vector a operar para que se forme nuevamente dicho vector con respecto a un nuevo grupo de secuencias a generar el prototipo.
12. Se retorna al paso 3 y se realiza nuevamente la generación de un prototipo para otro grupo de secuencias, esto se realiza mientras el valor de este vector del paso 3 sea diferente de 0 (cero), que es el valor que nos concluye la finalización del esquema de grupos de secuencias planteado.
13. Con la indicación de finalización, se crea un archivo plano (txt) y se imprime en este los prototipos generados.

Los resultados obtenidos con respecto a lo planteado de cadenas de Markov, fueron los siguientes prototipos, Figura 13.

efgk	
4	
e;f;g;k;f;f;e;k	
e;k;g;e	
e;k	
k;k;f;g;e;g	
2	
f;g;g	
e;f;g;e;k	
3	
e;f;g;g;e;k	
e;g;f;k	
g;k;f;f;e	
0	

PROTOTIPOS

ekfgekfg
efgef
efefef

Figura 13. Prototipos generados cadenas de Markov

Algunos sub-prototipos (por posición) obtenidos por este método, no relacionan representaciones relativas que sean esperadas con respecto a la particularidad de los datos por posición, con lo que se obtiene un prototipo final que concluye un porcentaje mayor en pérdida de información con respecto a PWM.

Como lo comentado en la sección 3.5.1 las ventajas o desventajas son relativas de acuerdo al valor significativo que asuma o no, una combinación de eventos.

Para las secuencias operadas no existe ninguna representación de unión o combinación que marque una dependencia o un subconjunto con un valor significativo, por tal, este método cadenas de Markov por su matriz de transición no tiene una correspondencia esperada cuando se opera con específicamente independencia de eventos como es el caso tratado de encuestas individuales que forman las secuencias de sitios turísticos visitados.

5.3. MÉTODO OESP

Teniendo presente la descripción de este método (Sección 4.1), se asignaron valores a los eventos de cada secuencia de un grupo g , donde el valor asignado a cada evento está relacionado con su ubicación dentro de la secuencia. De esta forma el prototipo se construye a partir de los valores referenciados de mayor a menor puntuación, es decir, el evento X_i de la posición 1 tiene mayor puntuación que el evento de la siguiente posición (posición 2) y así sucesivamente. Además, para los eventos que se encuentran repetidos en una secuencia se les diferencia con un número que se agrega como sub-indicador, por ejemplo, $X_1, X_2... X_n$ y como se ha mencionado van tomando un valor según su posición dentro de la secuencia.

Para la implementación de este método se ha diseñado el siguiente algoritmo:

1. Leer archivo .txt (esquema base) y capturarlo en un vector relacionando posición del vector por línea del archivo leído.
2. Se almacena en un vector los números que indican la cantidad de secuencias que hay para el grupo a operar.
3. Se almacena en una matriz las secuencias correspondientes al grupo a trabajar donde en dicha matriz las filas corresponden a las secuencias y las columnas a cada evento (posición) dentro de la secuencia.
4. Con respecto a la matriz anterior se crea una matriz donde se almacenan los eventos más su sub-indicador para posteriormente iniciar la puntuación.
5. Se realiza la puntuación con base en las dos matrices anteriores, cada evento con su sub-indicador (matriz punto 4) va puntuando con respecto a la ubicación en que se encuentre en la matriz del punto 3.
6. Obtenida la puntuación se organizan de mayor a menor los eventos con respecto a su puntuación.
7. Se calcula una longitud promedio entre las secuencias del grupo la cual va hacer la longitud del prototipo a obtener.
8. Se crea el vector prototipo con respecto al orden de puntuación (punto 6) y cuya longitud (número de eventos a representar) se obtiene en el punto 7.
9. Al conseguir el prototipo de un grupo g , se borran los arreglos (vectores y matrices) que se almacenaron durante este proceso, para así dar paso al proceso de obtención de un nuevo prototipo de otro grupo a trabajar.
10. Se retorna al paso 3 y se realiza nuevamente la generación de un prototipo para otro grupo de secuencias, esto se realiza mientras el valor del vector del paso 2 sea diferente de 0 (cero), que es el valor que nos concluye la finalización del esquema de grupos de secuencias planteado.
11. Con la indicación de finalización, se crea un archivo plano (txt) y se imprime en este los prototipos generados.

En la Figura 14 se observan los resultados (3 prototipos de los 3 grupos del ejemplo referenciado en la Figura 11) obtenidos por este método (OESP).

efgk	
4	
e;f;g;k;f;f;e;k	
e;k;g;e	
e;k	
k;k;f;g;e;g	
2	
f;g;g	
e;f;g;e;k	
3	
e;f;g;g;e;k	
e;g;f;k	
g;k;f;f;e	
0	

PROTOTIPOS

ekgfk
fgeg
gefk

Figura 14. Prototipos generados OESP

En este método, la representación por posición de algunos prototipos puede no ser identificativa del conjunto de secuencias para esa posición, o que ese evento no se encuentre en esa posición para ninguna secuencia como es el caso del prototipo 3 en la posición 2 (evento 'e') donde para ese grupo, ninguna secuencia tiene en la posición 2 el evento 'e', esto surge porque el evento 'e' en su sumatoria de puntuación es la segunda mayor (después de sumatoria de puntuación de evento g) es por ello que toma esta posición en el prototipo resultante obtenido.

5.4. MÉTODO OESP-NP

Este método se basa en el algoritmo del método anterior (OESP) pero con una modificación a la hora de puntuar los eventos con respecto a su posición.

Para su implementación se toman las secuencias del esquema referenciado en la Figura 11 cuyo algoritmo se mencionó anteriormente, con la siguiente modificación realizada en el punto 5 de este algoritmo (Sección 5.3):

- **Modificación punto 5:** Se realiza la puntuación para cada evento con su sub-indicador, va puntuando con respecto a la ubicación (elevada al cuadrado) en que se encuentre.

En la Figura 15 se observan los resultados (3 prototipos de los 3 grupos obtenidos con respecto a lo planteado por este método (OESP-NP).

efgk	
4	
e;f;g;k;f;f;e;k	
e;k;g;e	
e;k	
k;k;f;g;e;g	
2	
f;g;g	
e;f;g;e;k	
3	
e;f;g;g;e;k	
e;g;f;k	
g;k;f;f;e	
0	

PROTOTIPOS

ekgfk
fgeg
gefkg

Figura 15. Prototipos generados OESP-NP

Con este método se puede dar solución a la representación de algunos eventos del prototipo obtenido con respecto a sus posiciones identificables en el conjunto de secuencias, lo cual no se dan en algunos casos con el método OESP, pero esta diferenciación es más notoria con secuencias de mayor longitud.

5.5. MÉTODO OESP-EM

Al presentarse como una mejora del método OESP su base de proceso corresponde al algoritmo presentado en la sección 5.3 con la modificación en el momento de consolidar la longitud del prototipo a presentar, en la cual se eliminan los eventos del medio y no los eventos finales como nos presenta OESP.

Para su implementación se toman las secuencias del esquema referenciado en la Figura 11 cuyos pasos codificados (algoritmo) fueron basados como se menciona en el algoritmo del método OESP detallado, con la modificación realizada en el punto 8 de este algoritmo (Sección 5.3):

- Al momento de crear el vector prototipo con respecto al orden de puntuación, se eliminan los eventos medios hasta obtener la longitud promedio (número de eventos a representar) calculada con anterioridad

En la Figura 16 se observan los resultados (3 prototipos de los 3 grupos) obtenidos con respecto a lo planteado por este método (OESP-EM).

efgk	
4	
e;f;g;k;f;f;e;k	
e;k;g;e	
e;k	
k;k;f;g;e;g	
2	
f;g;g	
e;f;g;e;k	
3	
e;f;g;g;e;k	
e;g;f;k	
g;k;f;f;e	
0	

PROTOTIPOS

ekgfg
fgek
geffe

Figura 16. Prototipos generados OESP-EM

Lo identificable en este método es mantener la representatividad de los eventos finales del conjunto de secuencias, lo cual no se conseguía con el método OESP básico. Observamos que en este caso, el evento que queda en último lugar del prototipo era uno de los eventos en último lugar en las secuencias de referencia.

La desventaja puede surgir en que los eventos medios de las secuencias tengan bastante relación común entre ellos y que esta relación sea significativa, y de esta manera se estaría eliminando una información valorable en estas posiciones.

5.6. MÉTODO POR-PARES

Para implementar este método se tomó como base el algoritmo del método PWM estándar.

Con base en las frecuencias halladas en PWM, se calcula la frecuencia por pares de eventos para cada 2 posiciones, para así ir definiendo parejas de eventos (las de mayor frecuencia) por par de posiciones, con base en la matriz de frecuencias por pares desarrollada por grupo. Los pasos (algoritmo) que se codificaron, fueron los siguientes:

1. Leer archivo .txt (esquema base) y capturarlo en un vector relacionando posición del vector por línea del archivo leído.
2. Almacenar en un vector los eventos a tener presentes indicados en la primera posición del vector anteriormente capturado, agregando como evento anexo el símbolo "-" que nos representa las posiciones donde no se produce ningún evento.
3. Capturar posiciones del primer vector (paso 1) donde se indican los números que identifican el numero de secuencias a trabajar por grupo.
4. Con base en el número de secuencias a trabajar por grupo (definidos en paso 3) formar y almacenar en un arreglo (matriz par de *eventos* x cada 2 *posiciones*) la matriz PWM que define el grupo de secuencias a desarrollar.
5. Calculando la secuencia (longitud) mas larga del grupo (conjunto de secuencias) a trabajar, dejar todas las secuencias (de la matriz formada en paso 4) de igual tamaño agregando el parámetro "-" alineando dichas secuencias con base en la secuencia mas larga del grupo.
6. Calcular en una matriz de inicio, las veces que se encuentra cada par de eventos en todo el conjunto de secuencias, esto con el fin de desempatar cuando se presente igualdad de frecuencias, donde se tomara la puntuación de esta matriz para tomar la pareja más representativa.

7. Con respecto a la matriz PWM formada y alineada de igual tamaño con respecto a posiciones p (columnas) para cada secuencia s (fila), ir calculando y almacenando por cada 2 posiciones la frecuencia de cada pareja de eventos.
8. Con base en las frecuencias calculadas por cada 2 posiciones para cada pareja de eventos, tomar la pareja con mayor frecuencia (calculadas en el paso 7) e ir formando el vector prototipo que se va definiendo de dos posiciones, y así se va obteniendo el prototipo hasta llegar a la ultima posición de la matriz PWM con la que tenemos el prototipo final del grupo de secuencias trabajado.
9. Al obtener el prototipo final de un grupo G , borrar los eventos de la matriz PWM para que se forme nuevamente dicha matriz con respecto a un nuevo grupo de secuencias a generar el prototipo.
10. Se retorna al paso 3 y se realiza nuevamente la generación de un prototipo para otro grupo de secuencias, esto se realiza mientras el valor de este vector del paso 3 sea diferente de 0 (cero), que es el valor que nos concluye la finalización del esquema de grupos de secuencias planteado.
11. Con la indicación de finalización, se crea un archivo plano (txt) y se imprime en este los prototipos generados.

Con base en el ejemplo tratado hasta ahora, los prototipos generados fueron los siguientes, Figura 17.

efgk	
4	
e;f;g;k;f;f;e;k	
e;k;g;e	
e;k	
k;k;f;g;e;g	
2	
f;g;g	
e;f;g;e;k	
3	
e;f;g;g;e;k	
e;g;f;k	
g;k;f;f;e	
0	

PROTOTIPOS

ekfg----
fggek
efffek

Figura 17. Prototipos generados POR-PARES (Base PWM)

La representación de estos prototipos generados con respecto a la congruencia entre pares de eventos es distintiva para mantener además de elementos comunes, desplazamientos comunes que se hayan realizado entre turistas, como es el caso en que nos enfocamos en este trabajo.

Sin embargo, en este ejemplo se observa que para el primer grupo el prototipo obtenido es de menor longitud que en los casos anteriores y la pareja de símbolos vacíos '-' se ha tomado en distintas posiciones.

6. ESTUDIO COMPARATIVO

Un prototipo como conjugación de un conjunto de secuencias (grupo) pretende representar la información común de las secuencias que conforman un determinado grupo. Con respecto a esta representación, se puede obtener una medida valorativa que surge de la comparación de cada secuencia del grupo con respecto a la secuencia prototipo obtenida para ese grupo. Esta medida sirve para saber la similitud entre las secuencias originales y la secuencia prototipo.

Esta medida es muy importante si se quiere usar el prototipo para anonimizar un conjunto de datos por microagregación (tal como ya se ha explicado en la Sección 1.2). Lo que se consigue con esta comparación es tener un valor numérico que corresponde a la similitud entre la secuencia prototipo y cada secuencia del grupo, que nos ayuda a determinar si un prototipo es una buena representación de un grupo o conjunto de secuencias. Cuánto mayor sea la similitud, menor será la pérdida de información, y por tanto, más representativo será, y se podrá usar mejor para hacer análisis posteriores. Es decir, se mantendrá más cantidad de información.

La información con que se va trabajar corresponde a datos reales, cedidos por el Dr. Shoval, que proceden de un estudio que se hizo en una ciudad de Israel. Se ha seleccionado un subconjunto de secuencias correspondientes al itinerario dentro de esta ciudad. Previamente se ha usado una herramienta de clustering para construir una clasificación de las secuencias en pequeños grupos. A continuación se han aplicado todos los métodos de generación de prototipos explicados y diseñados en esta tesis de master. Se han evaluado los prototipos generados por cada método visto a fin de determinar qué método registra mejores resultados.

6.1. MEDIDA DE COMPARACIÓN

Dado un grupo G de secuencias, $S_1, S_2 \dots S_n$ como las secuencias que lo conforman y $c(G)$ un prototipo obtenido para el grupo G , se mide el error como una medida que corresponde a la suma de los cuadrados de las diferencias entre cada secuencia S_i y el prototipo $c(G)$, que se indica como SSE (*Sum of Squared Error*).

$$SSE = \sum_{i=1}^n (S_i - c(G))$$

donde:

- G es un grupo de secuencias
- n es el número de secuencias existentes en el grupo G
- S_i representa cada secuencia existente en el grupo G
- $c(G)$ es un prototipo definido para el grupo G

El valor *del error* SSE corresponde a valores entre 0 y 1, donde su rango valorativo indica que 0 (cero) indica que no hay error o pérdida de información, y por tanto el prototipo obtenido es la mejor representación; mientras que el valor 1 indica la distancia máxima, o máxima pérdida de información, obteniendo un prototipo con la peor representación. Dentro de estas valoraciones podemos inferir qué prototipos son mas o mejor representativos que otros para un grupo g , es decir, el método que obtiene menor valor dentro del rango 0-1 es el que ha generado un prototipo que representaría a los datos originales.

La diferencia $(S_i - c(G))$ entre secuencias S_i y la secuencia prototipo $c(G)$ se debe realizar de forma especial con una función de similitud adecuada para tratar con información simbólica, ya que se trata de eventos (letras) y por consiguiente no se puede obtener un resultado realizando llanamente la resta como correspondería para datos o valores numéricos.

Usaremos un enfoque para el calculo de similitudes conocido como OSS (*Ordered-based Sequence Similarity*) que con respecto a distancias entre eventos de una secuencia S_i y los eventos de una secuencia prototipo $c(G)$ nos ayuda a calcular $(S_i - c(G))$. Este enfoque OSS fue definido inicialmente en [Gómez&Valls, 2008] y mejorado posteriormente en [Valls et al, 2009].

Esta medida OSS permite al usuario definir diferentes políticas de preferencia para comparar la similitud de dos secuencias de eventos, para ello se utiliza el operador OWA (*Ordered Weighted Aggregation*), por lo tanto, se le denomina OSS_{OWA} .

OSS_{OWA} : Función de disimilitud para secuencias de eventos categóricos

Para el escenario que estamos tratando, evaluación de prototipos de los itinerarios de los turistas, hay dos cuestiones claves a considerar dentro de las políticas de preferencia para medir la similitud:

- El número de elementos (eventos) comunes dentro de las secuencias.
- El orden (posición) entre los elementos (eventos) comunes dentro de las secuencias.

La primera consideración, nos permite evaluar la parte común de dos secuencias diferentes, si esta parte común es grande, entonces las dos secuencias tienen que ser similares, ya que dichas secuencias muestran que sus propietarios (turistas) han visitado los mismos destinos. Por otra parte, en la segunda consideración se toman en cuenta si los hechos se han realizado en el mismo orden o no. Esta segunda información es muy útil para capturar el orden relativo entre los eventos y poder detectar las trayectorias y su direccionalidad. Esto puede servir posteriormente para planificar la gestión del flujo de los turistas dentro de la ciudad.

Es aquí donde el operador OWA se puede utilizar para introducir políticas de preferencia y realizar el cálculo de similitud. En el proceso de este cálculo se tiene en cuenta lo detallado a continuación.

Dado s y r dos secuencias de eventos, s_i es un evento i de la secuencia s , entonces la distancia de eventos de s_i con respecto a r se define como:

$$d(s_i, r) = \left\{ \frac{\min_{s_i=r_j} |i - j|}{|s| + |r|} \right\}$$

Donde $|s|$ y $|r|$ son las longitudes de s y r respectivamente.

Tomemos s como una secuencia de un grupo g y r como el prototipo el grupo g , lo que se realiza es una comparación de distancias entre los eventos comunes entre s y r que se denota $d(s_i, r)$, por ejemplo, si tenemos dentro de un grupo g una secuencia $s=(a,c,b)$ y un prototipo obtenido $r=(b,a,a,c)$ para la distancia $d(a_1, r)$ se tiene $\min a_1 = |2-1|$, $|s|=3$ y $|r|=4$, donde $|2-1|$ es la posición de a en secuencia r menos la posición de a en secuencia s , $|s|=3$ es la longitud de la secuencia s y $|r|=4$ es la longitud de la secuencia prototipo r , con lo que se obtiene como resultado $d(a_1, r) = (|2-1|)/(3+4) = 1/7$.

Este paso se realiza para todos los eventos de las dos secuencias, y seguidamente, las distancias parciales se agregan utilizando el operador OWA, que permite aplicar diferentes políticas de agregación.

Dadas secuencias s y r , se computan las distancias de eventos $d(s_i, r)$ para todo $i=1, \dots, |s|$ y $d(r_i, s)$ para todo $i=1, \dots, |r|$, a continuación se define el siguiente paso $d_{OOS-OWA}$.

Para s y r como dos secuencias de eventos con longitudes $|s|$ y $|r|$ respectivamente, la función disimilitud $d_{OOS-OWA}$ de s y r se define como:

$$d_{OOS-OWA(Q)}(s, r) = OWA_Q(d(s_1, r), \dots, d(s_{|s|}, r), d(r_1, s), \dots, d(r_{|r|}, s))$$

Donde Q es un cuantificador difuso no-decreciente definido de la siguiente forma:

$$Q^\alpha(x) = x^\alpha$$

De esta manera y con respecto a lo mencionado, se obtiene una medida de cálculo para $(S, c(G))$ donde el usuario puede agregar valores dando mas importancia a grandes o pequeños valores que correspondería al modelado del escenario a tener presente.

6.2. RESULTADOS DE LOS MÉTODOS SEGÚN LA POLÍTICA DE COMPARACIÓN

Teniendo presente las políticas de preferencia sobre los dos aspectos claves mencionados en la sección anterior, se han realizado un análisis utilizando los datos reales de itinerarios de turistas mencionados anteriormente.

- El número de elementos (eventos) comunes dentro de las secuencias.
- El orden (posición) entre los elementos (eventos) comunes dentro de las secuencias.

Para definir la política de agregación del operador OWA se deben dar distintos valores al parámetro α del cuantificador difuso. En este estudio se ha trabajado con los siguientes valores:

- $\alpha=2$ que indica igual preferencia para ambas claves, número de elementos comunes y orden (posición) entre los elementos comunes,
- $\alpha=8$ que indica mayor preferencia para la segunda clave, orden (posición) entre los elementos comunes.

Se ha considerado que esta segunda política de preferencia ($\alpha=8$) es la más representativa para este trabajo ya que dentro del itinerario de sitios visitados

por turistas, además del número de elementos comunes es aún mas relevante el orden (posición) en que se han realizado estas visitas pues enmarcan una rutina de visitas que siguen los turistas.

Las pruebas se realizaron con 4 grupos de secuencias de datos reales de itinerarios seguidos por turistas los cuales se detallan en la Tabla 7.

Id. del Grupo	Secuencias
1	a;b;a;b;j;b;w;b;k;o;g;e;s;g;s;r;n;r;z;c;m;d;z;r;n;s;e;h;b;w;a a;w;b;j;b;a;h;e;s;r;z;r;s;r;d;m;c;m;c;z;s;g;s;e;k;b;j;h;v;p;x;r;n;d;n;s;o;k;w;a a;b;e;s;r;n;r;z;c;m;d;z;x;s;d;g;e;k;b;w;a a;b;j;b;e;h;x;z;c;m;d;r;n;d;n;d;n;s;o;k;e;b;w;a
2	a;b;h;e;s;x;z;c;m;c;m;d;z a;h;s;x;z;m;z;c;m;d;r;s;e;w;a a;h;e;s;x;z;c;m;d;z;x;s;e;o;e;b;w;a a;h;e;s;x;z;c;m;d;z;x;h;b;w;a
3	a;h;b;j;b;e;h;e;s;r;z;c;m;d;z;r;s;e;k;b;w;a a;h;e;b;e;o;e;s;r;z;c;z;s;e;k;b;w;a a;h;e;z;c;m;d;r;s;o;k;e;b;w;a a;b;e;k;e;s;x;z;c;m;d;s;e;k;w;a
4	a;b;a;b;h;e;x;z;c;m;d;n;s;e;k;b;w;a a;b;a;b;h;e;s;x;z;c;m;d;z;x;s;o;k;e;b;w;a a;b;a;e;s;x;z;c;m;d;n;s;o;k;w;a a;b;a;h;a;b;k;e;s;x;z;c;m;d;n;s;o;k;w;a a;b;a;h;p;f;p;z;c;r;s;n;g;s;e;h;e;b;w;a a;b;a;h;e;s;x;z;c;m;d;n;d;n;s;g;e;b;w;a

Tabla 7. Grupos de secuencias de itinerarios seguidos por turistas

Podemos observar que en los 4 grupos tenemos un número reducido de secuencias, puesto que nos interesa construir prototipos en métodos de microagregación, y por tanto, se forman grupos pequeños para no distorsionar demasiado los datos originales, solo lo mínimo para preservar la privacidad.

El primer grupo contiene secuencias de longitudes bastante diferentes. Los grupos 2 y 3 tienen secuencias de menor longitud y de tamaños similares. Finalmente el grupo 4 contiene más secuencias que los anteriores, pero también de similar longitud.

Con respecto a las secuencias de estos 4 grupos referenciados en la Tabla 7, a cada grupo se le aplicaron todos los métodos generadores de prototipos evaluados, incluyendo las propuestas de mejoras, a fin de poder compararlos.

6.3. ANÁLISIS DE LOS RESULTADOS

Para los primeros resultados (Sección 6.2.1. Resultados con $\alpha=2$ (igual preferencia para ambos aspectos)) no se evidencia una mejora en el método OESP. Seguramente se debe a que se ha ponderado de forma igual la presencia de elementos en común y su orden, por tanto, los prototipos finales si tienen por lo menos los mismos elementos en común bien representados esto ya es suficiente, y no se captura con esta parametrización el hecho de mejorar el orden en algunos elementos, si en otros no se da este hecho.

Sin embargo, si parece haber una mejora con respecto al método POR-PARES y los métodos PWM y Markov, siendo este último el que siempre da peores resultados.

Concertándonos ahora en los segundos resultados (Sección 6.2.2. Resultados con $\alpha=8$ (mayor preferencia para segunda clave)) que ya habíamos previsto que serían más interesantes puesto que precisamente queremos medir si conseguimos representar correctamente el orden relativo entre los eventos, vale notar las mejoras de las nuevas propuestas respecto a los métodos originales. Las propuestas de mejoras sobre los métodos evaluados se centran en los métodos PWM y OESP respectivamente.

Se observan las siguientes mejoras:

- a) Mejora de PWM: La propuesta de mejora con base en este método denominada como POR-PARES, registra una mayoritaria representación de mejores resultados (SSE) con respecto a PWM como se evidencia en los grupos (Id) 1, 2 y 3 cuyos prototipos registran mejor similitud que los generados por PWM.
- b) Mejora de OESP: Para este caso, la propuesta denominada OESP-NP registra igualmente una mayoritaria representación de mejores resultados (SSE) con respecto a OESP, cuyos prototipos (de OESP-NP) registran mejor similitud que los prototipos generados por OESP.

Para el caso que nos compete, se puede concluir que más que centrarnos en elementos tomados de manera particular, se debe enfatizar los procesos de construcción de un prototipos que tomen en consideración el orden relativo que puede existir dentro de los elementos de un conjunto de secuencias que atañen en el itinerario que han seguido los turistas.

Dentro de las particularidades evidencias en los resultados obtenidos se puede reseñar:

- La matriz de transición concurrente en cadenas de Markov genera pares repetitivos de eventos (como el prototipo dado en el grupo 1) con los cuales no se obtiene buena puntuación de similitud del prototipo con respecto al conjunto de secuencias, lo que lo posiciona en últimas posiciones con respecto a los otros métodos.
- El método de mejoras sobre OESP, posiciona con mejores resultados a la nueva puntuación (OESP-NP) con respecto a la eliminación (eventos) de medios (OESP-EM).
- Para el método OESP-EM, la idea era que el prototipo correspondiera de manera representativa los eventos tanto iniciales como finales con respecto al conjunto de secuencias, pero los eventos finales pueden puntuar menos que algunos que solo se encuentran una vez como es el caso evidenciado para el grupo 3.
- La nueva puntuación (OESP-NP) presentada del método OESP suele mejorar con preferencia de $\alpha=8$ que es mas relevante para este trabajo, donde se tiene presente tanto los eventos comunes como su orden (posición) entre estos.
- Aunque el prototipo OESP-NP ha obtenido peor posición que OESP en el ranking del grupo 4, si comparamos los prototipos con las secuencias originales, podemos ver que OESP-NP coloca el par 'he' en el prototipo, mientras que OESP coloca el par 'eh'. El primero es más habitual en las secuencias iniciales, lo que intuitivamente nos haría elegir el prototipo del método OESP-NP al de OESP, esto nos ha hecho cuestionarnos la medida de error utilizada.
- El método POR-PARES basado en frecuencias del método PWM arroja mejores resultados que PWM teniendo presente que los cluster generados sean representativos de turistas con desplazamientos en común.

7. CONCLUSIONES FINALES Y CONTRIBUCIONES

El desarrollo de esta tesis representó un reto muy importante para mí. En el transcurso del trabajo trazado, iba acrecentando mi interés con adquisiciones de conocimientos totalmente nuevos que forjaban eslabones hacia el cumplimiento de los objetivos planteados.

En la culminación de este trabajo, se puede concluir que los objetivos planteados en la sección 1.1 se han cumplido acorde a los propósitos proyectados.

Se realizó un estudio del estado del arte de los métodos existentes para la obtención de un prototipo a partir de un conjunto de secuencias de eventos, los cuales fueron detallados en las secciones 3 y 4. Se evaluó cada uno de los métodos con ejemplos simples que nos ayudaron a denotar las ventajas y desventajas de cada método, determinando los puntos fuertes o débiles que son representativos para la óptima obtención de un prototipo que cumpla con la finalidad en relación a la microagregación, estos se detallan en la sección 3.5. Se propusieron mejoras a los algoritmos existentes en la sección 4. Se desarrolló una aplicación informática que implementa todos los métodos explicados, así como, realizar adaptaciones o modificaciones de los métodos existentes, esto se desarrolló en la sección 5. Finalmente, se desarrolló un estudio comparativo entre los métodos existentes y los alternativos propuestos donde fueron analizados sus resultados con datos sintéticos³ y con datos reales de itinerarios de turistas, esto se expuso en la sección 6.

Dentro de las investigaciones realizadas en el contexto de esta tesis, se apreció que son escasos los temas de investigación o aplicación en este tema y aún más relacionados a itinerarios de turistas. Hemos constatado que el enfoque de secuencias consenso "*prototipos*" hasta ahora se ha estudiado sobretodo en bioinformática, como en el caso de representación e interpretación de cadenas de ADN.

Sin embargo no se ha encontrado constancia de la aplicación de estos métodos para el propósito estudiado en este trabajo, o sea para la aplicación en protección de la privacidad en datos de secuencias de eventos. Con respecto a esto, se abre un punto de prelude hacia posibles líneas futuras de investigación de métodos de construcción de secuencias consenso más específicas para este ámbito.

³ Datos creados artificialmente como base para análisis de resultados

A partir del anàlisis comparatiu que se realitzà sobre els mètodes existents i els mètodes proposats caldria destacar les següents conclusions:

- Destacamos haber conseguido mejorar el método PWM con la propuesta POR-PARES, y tener alternativas que mejoran ligeramente el comportamiento del OESP, como el método OESP-NP. También parecer ser que el método OESP-EM no da buenos resultados.
- Los resultados obtenidos con datos sintéticos muestran algunas diferencias en relación a los resultados obtenidos con el conjunto de datos reales tratados en la sección 6. Con respecto a esto, creemos que se deben usar otros datos reales para validar las conclusiones presentadas en la sección 6, ya que el problema con datos sintéticos es que éstos, al ser generados artificialmente, puede que no representen sentido con respecto a cierto modelo de comportamiento del turista.
- Contar con otra(s) medida(s) de similitud entre secuencias de eventos que ayuden a afianzar conclusiones en el análisis de resultados partiendo de la base comparativa entre las diferentes medidas de similitud que se puedan obtener.
- En ciertas situaciones, el prototipo obtenido en estos métodos puede no garantizar la protección de alguno de los registros iniciales. Es importante encontrar el equilibrio entre representatividad de los registros iniciales frente a su adecuada protección. Para poder preservar la privacidad en las secuencias iniciales es necesario que el prototipo no sea igual a ninguna de ellas. En este análisis comparativo, este aspecto no se ha tenido en cuenta, puesto que la solución sería aplicar alguna heurística para cambiar una parte del prototipo, ya sea alterando algún parámetro del método (por ejemplo, las prioridades entre símbolos). Consideramos que introducir esta variación en el método y comparar después los prototipos alteraría los resultados y el valor del error asociado y, por tanto, la comparación de estos resultados no correspondería a la comparación de los métodos originales.

A partir del trabajo realizado y las conclusiones obtenidas queremos destacar las siguientes contribuciones de esta tesis:

1. Se detallaron a nivel explicativo y de análisis los métodos existentes de construcción de prototipos y se estudio su aplicabilidad al problema de la microagregación en protección de datos. Este estudio sirve como base para estudios futuros.
2. Con respecto a desventajas que se iban registrando como consecución del estudio, se presentaron nuevas propuestas para la construcción de mejores prototipos de secuencias de eventos.

3. Se evaluaron estas propuestas y se ha obtenido un primer estudio comparativo de los métodos que puede servir para diseñar nuevos algoritmos en este campo incipiente.

Posibles líneas futuras

Como ya se mencionó, hay muy pocos métodos que se puedan usar para la construcción de prototipos en el caso de secuencias de eventos, como las tratadas en este trabajo sobre itinerarios de turistas. Por tanto, esta constatación permite abrir una línea de investigación en este tema.

Como líneas de trabajo más concretas podríamos mencionar las siguientes:

1. Mejorar los algoritmos presentados para que sus resultados aseguren la privacidad siempre. Para ello se debe definir exactamente el propósito para el que se protegen los datos y analizar si los prototipos obtenidos aseguran la no re-identificación de los individuos en este contexto.
2. Puesto que las diferencias en el error estimado no son muy significativas, se debería estudiar si se puede medir la calidad del resultado de otra forma. Por ejemplo, evaluar si las conclusiones que se sacan después en los datos anonimizados son ciertas, ya que los eventos de los prototipos obtenidos pueden no corresponder a relaciones coherentes o posibles en el mundo real, y por tanto, podríamos generar una base de datos anonimizada pero sin sentido.
3. Considerar otro tipo de información sobre el contexto de los datos de las secuencias de eventos a tratar. Con esta información adicional se podría mejorar el cálculo de prototipos basados en información relevante para interpretar el significado de las secuencias. Por ejemplo, se pueden considerar transiciones no posibles entre eventos que no debería representar el prototipo final. Se debe ver como se puede introducir esta información dentro del método de protección de datos y concretamente dentro del método de generación de prototipos.

REFERENCIAS

Bibliografía

[Antunes&Oliveira, 2001] Antunes M. Claudia, Oliveira ArlindoL., *Temporal Data Mining: an overview*, KDD Workshop on Temporal Data Mining, 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, USA, 2001.

[Brudno&Malde, 2003] Brudno M., Malde S., «Glocal alignment: finding rearrangements during alignment». *Bioinformatics* 19 Suppl 1: i54-62. DOI:10.1093/bioinformatics/btg1005. PMID 12855437, 2003.

[Dayhoff, 1978] M. O. Dayhoff, M. Schwartz, B. C. Orcutt, *A model of evolutionary change in proteins*. Atlas of Protein Sequence and Structure, 5(Suppl 3):345-352, 1978.

[Domingo&Torra, 2005] Domingo-Ferrer J., Torra V., *Ordinal, continuous and heterogeneous k-anonymity through microaggregation*. Data Min. Knowl. Discov., 11(2):195-212, 2005.

[Dong&Pei, 2007] Dong Guozhu, Pei Jian, *Sequence Data Mining*, Springer 2007.

[Gomatan et al., 2003] Gomatan, S., Karr, A. F., Reiter, J. P., and Sanil, A. P., *Data Dissemination and Disclosure Limitation in a World Without Microdata: A Risk-Utility Framework for Remote Access Analysis Servers*, Statist. Sci., Submitted for publication, 2003.

[Gómez&Valls, 2008] Gómez-Alonso, C., Valls, A., *A similarity measure for sequences of categorical data based on the ordering of common elements*, Lecture Notes in Artificial Intelligence, Vol. 5285, pp.134-145 (2008).

[Hernández et al, 2004] Hernández Orallo José, Ramírez Quintana María, Ferri Ramírez César, *Introducción a la minería de Datos*, Pearson 2004.

[Hertz, 1999] Hertz G.Z., *Identificación de ADN y proteínas con patrones estadísticamente significativas alineaciones de múltiples secuencias*. Bioinformática, 15: 563-577, 1999.

[Liao, 2005] Liao T.W., *Clustering of time series data-a Surrey*, Pattern Recognition, 38(11):1857-1874, 2005.

[Mas, 2000] Mas, M., *Técnicas de Protección y Seguridad de Datos Estadísticos. Reporte del Instituto Vasco de Estadística, 2000.*

[Mateo&Domingo, 1998] Mateo Sanz J.M., Domingo Ferrer J., *A comparative study of microaggregation methods*, Questiió: Quaderns d'Estadística, Sistemes, Informàtica i Investigació Operativa, 22(3), 1998.

[Mount, 2004] Mount D.M., *Bioinformatics: Sequence and Genome Analysis 2nd ed.*. Cold Spring Harbor Laboratory Press: Cold Spring Harbor, NY. ISBN 0-87969-608-7, 2004.

[Needleman&Wunsch, 1970] S.B. Needleman, C.D. Wunsch, *A general method applicable to the search for similarities in the amino acid sequence of two proteins*, J. Mol. Biol. 48(3):443-453 (1970).

[Pérez&Santín, 2007] Pérez César, Santín Daniel, *Minería de Datos técnicas y herramientas*, Thomson 2007.

[Pérez, 2006] Pérez César, *Data Mining*, Ra-Ma 2006.

[Polettini et al, 2002] Polettini S., Franconi L., Stander J., *Inference Control in Statistical Databases, From Theory to Practice*, Lecture Notes In Computer Science; Vol. 2316, Pages: 83 – 96, Springer-Verlag 2002.

[Samarati, 1998] Samarati P., *Generalizing data to provide anonymity when disclosing information*, In Proc. of the 17th ACM-SIGMOD-SIGACT-SIGART Symposium on the Principles of Database Systems, p. 188, Seattle, WA, USA 1998.

[Schneider&Stephens, 1990] Schneider T.D., Stephens R.M., «Sequence logos: a new way to display consensus sequences». *Nucleic Acids Res* 18: 6097-6100. DOI:10.1093/nar/18.20.6097. PMID 2172928, 1990.

[Seneta, 1996] Seneta E., *Markov and the birth of chain dependence theory*, International Statistical Review, 64(3), pp 255-263, 1996.

[Shapiro&Frawley, 1991] Shapiro G. Piatetsky, Frawley W.J., *Knowledge Discovery In Databases: An Overview. In Knowledge Discovery In Databases*, AAAI Press/MIT Press, Cambridge, MA., pp. 1-30, 1991.

[Sweeney, 2002] Sweeney L., k-anonymity: a model for protecting privacy, International Journal of Uncertainty, Fuzziness and Knowledge Based Systems, 10, pp.557-570, 2002.

[Torra&Domingo, 2001] Torra V., Domingo Ferrer J., *Confidentiality, disclosure, and data access: Theory and practical application for statistical*

agencies, Chapter Disclosure control methods and information loss for microdata, Elsevier, 2001.

[Valls et al, 2009] Valls, A., Nin, J., Torra, V., *On the use of aggregation operators for location privacy*, IFSA-EUSFLAT Conference, Lisbon, Portugal, July 2009.

[Valls&Gómez, 2008] Valls Aida, Gómez A. Cristina, El algoritmo de microagregación KSHC para anonimización de secuencias de datos categóricos, Reporte de investigación de la URV, DEIM-RR-09-001, 2009.