UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE INGENIERÍA
Departamento de Ingeniería Informática

ALGORITMO GENÉTICO PARA EL PROBLEMA DE ESCALAMIENTO MULTIDIMENSIONAL MULTI-OBJETIVO

Juan Cristián Giglio Gutiérrez

Profesor guía: Manuel Villalobos Cid
Profesor co-guía: Mario Inostroza Ponta

Trabajo de titulación presentado en conformidad a los requisitos para obtener el título de Ingeniero de Ejecución en Computación e Informática

Santiago – Chile
2019

RESUMEN

El escalamiento multidimensional (MDS) permite visualizar la similitud entre diferentes objetos reduciendo el número de dimensiones. MDS ha sido ampliamente utilizado para efectuar análisis exploratorios en distintas áreas del conocimiento. Las estrategias de MDS actuales utilizan exclusivamente una medida de similitud, sin embargo, la mayor parte de los problemas de la vida real requieren analizar más de una medida simultáneamente. Las técnicas de optimización multi-objetivo han sido utilizadas con éxito para resolver problemas con múltiples criterios (dos o tres criterios). En este trabajo se propone un algoritmo genético para resolver el problema de MDS multi-objetivo que es evaluado utilizando conjuntos de datos clásicos de la literatura relacionada. Los resultados muestran que la estrategia propuesta es capaz de identificar un conjunto de soluciones pertenecientes a la frontera de Pareto que incluye nuevas representaciones que no son dominadas por los enfoques de optimización mono-objetivo del estado del arte, y nuevas soluciones que combinan las características de las diferentes entradas. Estos resultados hacen que esta propuesta sea una alternativa real para encontrar soluciones a problemas que requieran visualizar diferentes medidas de similitud.

Palabras Claves: escalamiento multidimensional, algoritmo genético, optimización multi-objetivo, visualización de datos.

Agradezco la oportunidad de obtener nuevos conocimientos y a las personas que me ayudaron durante este proceso. Manuel, Angelo, Joyce, familia y amigos gracias por su apoyo y paciencia.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE ILUSTRACIONES

ÍNDICE DE ALGORITMOS

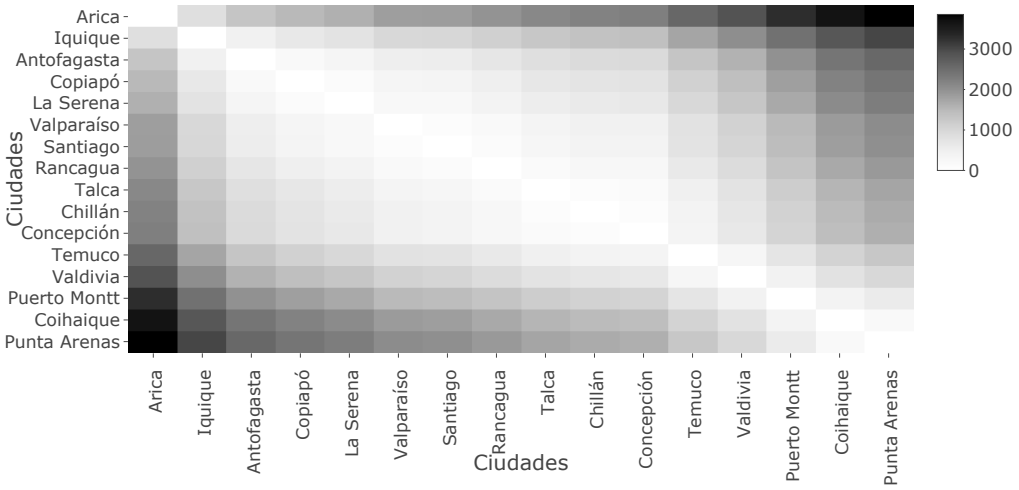
CAPÍTULO 1. INTRODUCCIÓN

1.1 ANTECEDENTES Y MOTIVACIÓN

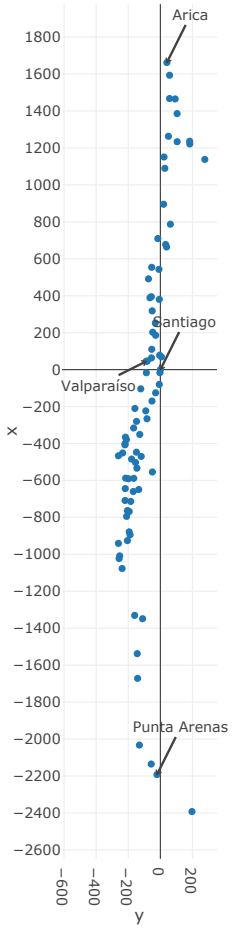
Reducir la dimensionalidad de los datos sin perder información es una etapa importante del pre-proceso dentro del análisis de datos. El objetivo de la reducción de dimensionalidad es representar muestras de datos multi-dimensionales en un espacio de menor dimensionalidad, preservando la mayor parte de la información contenida en los datos (Roweis, 2000). Por ejemplo, los píxeles que componen una imagen se pueden considerar como un objeto de múltiples dimensiones y la imagen que se percibe como una representación de este objeto con menor dimensionalidad. Una vez que la reducción de dimensionalidad es llevada a cabo correctamente, esta nueva representación compacta de los datos puede ser utilizada en tareas de visualización, clasificación u otras, con el fin de extraer nuevo conocimiento acerca de los datos. Uno de los métodos utilizados para la reducción de dimensionalidad es el de escalamiento multi-dimensional (Buja et al., 2008; Tenenbaum, De Silva y Langford, 2000).

El **escalamiento multi-dimensional**, desde ahora MDS por sus siglas en inglés (*MultiDimensional Scaling*), es un conjunto de métodos para visualizar el nivel de similitud o disimilitud entre objetos multidimensionales en un espacio de menor dimensión. Corresponde a un conjunto de procedimientos que tienen relación con la construcción de una configuración de n puntos, usualmente en el espacio euclidiano, a partir de la información acerca de su similitud representada por una distancia entre un conjunto de objetos (Mead, 1992), donde el nivel de similitud entre ellos está dado por la cercanía entre unos y otros, a menor distancia mayor similitud. Para entender en qué consiste, se ha planteado el siguiente ejemplo. La Figura 1.1a muestra un mapa de calor con la distancia entre las principales ciudades de Chile. Se puede intuir qué ciudades son más distantes de otras, sin embargo, resulta difícil determinar la posición de cada ciudad en un plano, ya que no existen datos asociados a su latitud y longitud. Utilizando MDS se puede obtener una representación en dos dimensiones como la presentada en la Figura 1.1b, que se aproxima a las ubicaciones reales de las ciudades, ver Figura 1.1c. Este método es utilizado generalmente para comparar objetos con múltiples características donde resulta difícil observar *a priori* si son similares.

MDS es una herramienta de análisis general utilizado en distintas áreas del conocimiento. Al consultar el tópico *"multidimensional scaling"* en la base de datos bibliográfica SCOPUS (2019), se obtienen 11.596 documentos distribuidos por área para las últimas seis décadas como se muestra en la Figura 1.2. En el campo de la **psicología**, la noción de modelar la similitud como una distancia en un espacio cartesiano no es nueva, ya que cuando una persona



(a) Mapa de calor de la distancia (km) entre capitales regionales de Chile.



(b) MDS distancia (km) entre ciudades de Chile.

(c) MDS (círculos) v/s ubicación real (puntos).

Figura 1.1: Ejemplo MDS ciudades de Chile, datos extraídos de (Simplemaps, 2019), Fuente: Elaboración propia (2019).

emite un juicio de similitud entre objetos, crea una representación de los objetos en un “espacio psicológico” considerando sus atributos y calcula una distancia global (Groenen y Borg, 2013). Para un observador este espacio es desconocido, una forma de descifrarlo es mediante un análisis de MDS. En **sociología** ha sido utilizado para investigar redes sociales (Scott, 1988) y para entender la estructura que relaciona los valores humanos entre distintas culturas (Schwartz, 1994). En el área de **lingüística** se ha usado para determinar las dimensiones asociadas a la percepción de un tono lingüístico considerando el lenguaje nativo de cada individuo (Gandour y Harshman, 1978). En **biología**, el MDS ha sido usado como método para aplicaciones tan específicas como el descubrimiento de patrones no jerárquicos entre estructuras genéticas y su ubicación geográfica (Lessa, 1990), o para determinar el grado de dolor en personas con distintos orígenes étnicos y culturales, utilizando MDS para descubrir las dimensiones subyacentes al problema (Cleeland et al., 1996). También el MDS ha contribuído en otras áreas como **turismo y marketing** (Gartner, 1989), **medicina** (Schiffman, Musante y Conger, 1978; Roux, 2008) y **educación** (Ding, 2018). En resumen, MDS es un método versátil y ampliamente utilizado para apoyar el análisis de datos en las diferentes áreas del conocimiento.

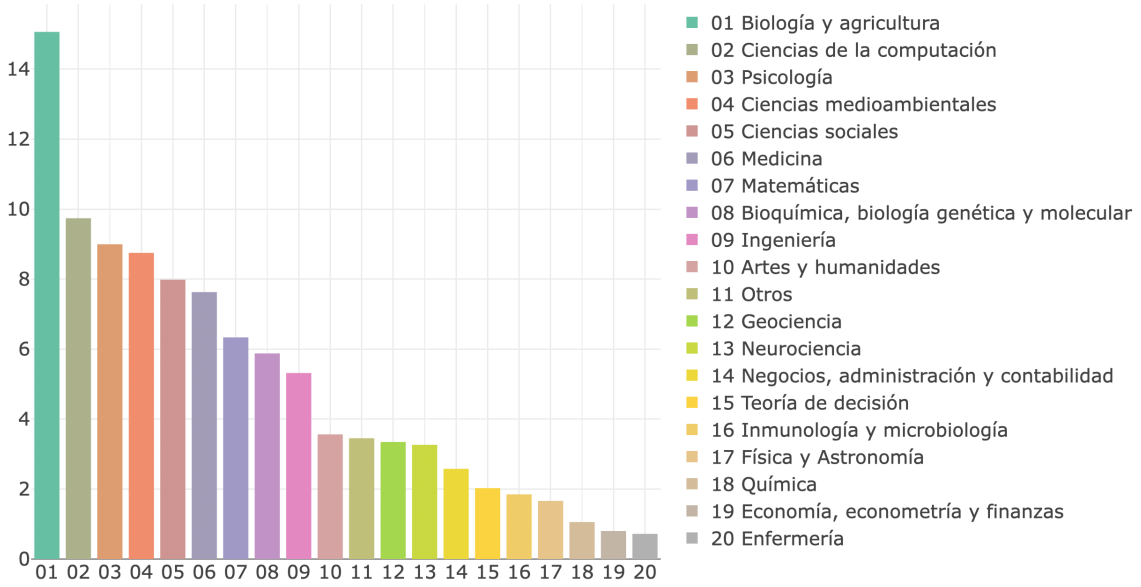


Figura 1.2: Porcentaje de documentos por área del conocimiento, datos obtenidos de SCOPUS (2019). Fuente: Elaboración propia (2019).

Los métodos de MDS actuales son capaces de crear representaciones de los objetos en un espacio de menor dimensionalidad utilizando una medida de similitud (optimización mono-objetivo). Sin embargo, la mayoría de los problemas pueden contener múltiples medidas de similitud conflictivas entre sí. Por ejemplo, Machado, Duarte y Duarte (2011) estudiaron por separado distancia de correlación y de histograma entre acciones del mercado de valores, Choi

(2012) crea dos representaciones MDS para comparar similitudes asimétricas entre países, y Villalobos-Cid, Dorn e Inostroza-Ponta (2018b) compararon dos distancias filogenéticas para estudiar árboles filogenéticos. Si las medidas de similitud son analizadas separadamente ¿La información acerca la combinación e interacción entre las medidas de similitud pueden ser descartadas?.

El problema de MDS multiobjetivo se define como la obtención de una o más representaciones MDS a partir de dos medidas de similitud que pueden ser conflictivas entre sí. Como entrada se tienen dos matrices de similitud y como salida se obtiene una o más representaciones MDS que minimizan el error con respecto a ambas medidas de similitud al mismo tiempo. Estas representaciones MDS contienen información acerca de la combinación e interacción de ambas medidas de similitud, lo que es interesante desde el punto de vista de análisis de datos y que permite la aplicación de MDS a problemas que involucran más de una medida de similitud.

Los métodos de MDS de diferencias individuales (Carroll y Chang, 1970) y Bai, Bai, Latecki y Tian (2017) enfrentan el problema de combinar múltiples matrices de entrada en una representación MDS ponderando las entradas conflictivas, sin embargo, la escala de las entradas usualmente no está relacionada, por lo que las entradas no son comparables, o los ponderadores no son siempre conocidos, lo que podría introducir sesgo.

En este trabajo se propone una nuevo modelo multi-objetivo para afrontar el problema de MDS multi-objetivo. Está basado en el algoritmo genético de ordenamiento no-dominado (NSGA-II, por sus siglas en inglés *Non-dominated Sorting Genetic Algorithm II*) (Deb et al., 2002) que es capaz de combinar dos medidas de similitud que pueden ser conflictivas entre sí. Las soluciones pueden aportar información acerca de las características individuales de las entradas y su combinación.

1.2 DESCRIPCIÓN DEL PROBLEMA

Dada la existencia de problemas con diversas métricas de similitud que pueden ser conflictivas entre sí y que los métodos de MDS que reciben como entrada más de una matriz de similitud recurren a ponderar dimensiones o distancias. El problema se puede enunciar de la siguiente manera: ¿Es posible resolver el problema de escalamiento multidimensional multi-objetivo con dos métricas de similitud que pueden ser conflictivas entre sí?

1.3 SOLUCIÓN PROPUESTA

1.3.1 Características de la solución

Para abordar el problema de MDS multi-objetivo se propone un algoritmo genético evolutivo que sea capaz de combinar dos matrices de similitud en una representación MDS, para esto se analizan distintas operaciones de inicialización, genéticas y de selección que permitan obtener un conjunto de soluciones “buenas” en términos de métricas de calidad multi-objetivo. Por lo tanto, la solución propuesta consiste en un algoritmo genético que como parámetros de entrada reciba dos matrices de $n \times n$ con medidas de similitud conflictivas entre sí, y retorne una configuración de n puntos en el plano cartesiano, minimizando la función de *stress* para ambas matrices simultáneamente.

1.3.2 Propósitos de la solución

El propósito de la solución es ampliar el espectro de problemas a los cuales se puede aplicar MDS, adicionando la posibilidad de utilizar dos métricas de similitud, que podrían ser conflictivas entre sí, simultáneamente.

1.4 OBJETIVOS Y ALCANCES DEL PROYECTO

1.4.1 Objetivo general

Diseñar e implementar un algoritmo genético multi-objetivo para efectuar escalamiento multidimensional considerando como entrada dos matrices de similitud que pueden ser conflictivas entre sí.

1.4.2 Objetivos específicos

1. Buscar en la literatura conjuntos de datos de prueba y reales que han sido tratados con estrategias para MDS.
2. Evaluar relaciones entre diferentes funciones de error (*stress*) entre la matriz de similitudes y la representación obtenida al aplicar MDS.
3. Diseñar e implementar un algoritmo genético multi-objetivo en base al contexto del problema.
4. Comparar el rendimiento y la calidad de soluciones de la estrategia propuesta y técnicas mono-objetivo *SMACOF* y *CMDSCALE* en términos de métricas de calidad multi-objetivo.
5. Aplicar la estrategia propuesta sobre los conjuntos de datos *glass*, *iris*, *breast cancer Wisconsin*, *ionosphere*, *FluTrees* y hospitales.

1.4.3 Alcances

- El algoritmo genético recibe como parámetros dos matrices triangulares inferiores, con el mismo tamaño que contienen métricas de similitud numéricas y mayores a cero.
- No se consideran valores en las diagonales de la matriz, se asume que la distancia entre el mismo objeto es cero.
- La salida es una matriz de $n \times 2$ donde n es el número de objetos.

1.5 ORGANIZACIÓN DEL DOCUMENTO

En el Capítulo 2 se revisan los conceptos generales de MDS, los tipos de MDS y algunos de los algoritmos de la literatura. El algoritmo propuesto es descrito en el Capítulo 3. En el Capítulo 4 se mencionan los métodos y materiales utilizados en este trabajo. Los resultados se exponen en el Capítulo 5. Las conclusiones acerca de los resultados y objetivos se encuentran en el Capítulo 6. Finalmente, en los Anexos A hasta el F se presentan la frontera de Pareto y las representaciones MDS obtenidas para los conjuntos de datos iris, *breast cancer wisconsin*, diabetes, *ionosphere*, *flutress* y hospitales2014 respectivamente y en el Anexo F la portada del artículo basado en este trabajo presentado en las jornadas chilenas de la computación 2019 (Giglio, Villalobos-Cid e Inostroza-Ponta, 2019).

CAPÍTULO 2. ESCALAMIENTO MULTIDIMENSIONAL

2.1 ESCALAMIENTO MULTIDIMENSIONAL MÉTRICO

El MDS métrico considera n objetos y un conjunto de medidas de similitud entre pares de puntos (δ_{ij}) para crear una representación de baja-dimensionalidad (dos o tres dimensiones), donde cada punto corresponde a un objeto y cada objeto está separado por una distancia $d_{ij} = f(\delta_{ij})$ con los otros objetos, siendo f una función continua, monótona y paramétrica (Cox y Cox, 2008). El MDS clásico es la primera versión de MDS y asume que las similitudes son distancias (Young y Householder, 1938; Torgerson, 1952; Cox y Cox, 2008). Puede ser considerado el primer acercamiento algebraico a MDS. Fue propuesto independientemente por Torgerson (1952) y Gower (1966). La idea básica del MDS clásico es asumir que las similitudes son distancias y buscar las coordenadas que puedan explicarlas. Borg y Groenen (2005) resumen el procedimiento en los siguientes pasos:

1. Se calcula la matriz de disimilitudes al cuadrado $\Delta^{(2)}$ de $n \times n$
2. Se multiplica el lado izquierdo y derecho de $-\frac{1}{2}\Delta^{(2)}$ por una matriz centradora $J = I - n^{-1}uu'$ donde I es la matriz identidad de $n \times n$ y u es un vector de n unos, ecuación (2.1)

$$B_{\Delta} = -\frac{1}{2} J \Delta^{(2)} J \tag{2.1}$$

3. Se calcula la descomposición en valores propios de B_{Δ} según la ecuación (2.2), donde Q es una matriz unitaria y Λ es una matriz diagonal de valores propios

$$B_{\Delta} = Q \Lambda Q' \tag{2.2}$$

4. Finalmente, se calcula la matriz de coordenadas X según la ecuación (2.3) donde Λ_{+} es la matriz con los primeros m valores propios mayores a cero y Q_{+} una matriz con las m primeras columnas de Q.

$$X = Q_{+} \Lambda_{+}^{\frac{1}{2}} \tag{2.3}$$

Si Λ es una matriz de distancia euclidiana, entonces las coordenadas X son encontradas por rotación. En el punto 4 se pueden generar valores propios negativos, pero no cuando Λ es una matriz de distancia euclidiana. Si existen valores propios negativos se ignoran.

El MDS clásico minimiza la función de costo en la ecuación (2.4), a veces llamada *strain* (Borg y Groenen, 2005).

$$\textit{Strain}(X) = ||XX' - B_{\Delta}||^2 \tag{2.4}$$

2.2 ESCALAMIENTO MULTIDIMENSIONAL NO MÉTRICO

En investigaciones prácticas frecuentemente se cuenta sólo con el orden de similitud, por ejemplo se conoce que los objetos 1 y 2 son más similares entre sí que los objetos 1 y 3 ($\delta_{12} < \delta_{13}$). En este caso las similitudes δ_{ij} son transformadas en disparidades $\hat{d}_{ij}$. Las disparidades son transformaciones admisibles de las similitudes, por ejemplo si solo importa el orden de similitud entonces las disparidades deben respetar el mismo orden. Si las disparidades están relacionadas a las similitudes por una función continua específica entonces es MDS métrico. El MDS no métrico es una extensión del MDS métrico que permite la visualización de objetos lo mejor posible en un espacio de baja-dimensionalidad a partir de un orden de similitud entre objetos (Cox y Cox, 2008), fue propuesto por Shepard (1962a) y mejorado por Kruskal (1964), proponiendo distintas funciones de *stress* (S).

2.3 DIFERENCIAS INDIVIDUALES EN MDS

En estudios de MDS acerca de juicios de similitud emitidos por múltiples individuos sobre los mismos objetos se busca encontrar un espacio psicológico común y sobre este buscar las diferencias de cada individuo asumiendo que cada individuo da un peso distinto a cada dimensión (Carroll y Chang, 1970). Por ejemplo Borg y Groenen (2005) muestran utilizando los datos un estudio de similitud de color, donde 14 individuos juzgan la similitud entre 10 fichas de distintos colores, que al comparar las representaciones MDS individuales con una representación MDS de consenso es posible identificar a los individuos con daltonismo.

Para explicar las diferencias de juicios de similitud emitidos por múltiples individuos, representados por K matrices de similitud, Horan (1969) y Carroll y Chang (1970) propusieron una extensión de MDS, el “modelo euclidiano ponderado”, que asume que cada matriz de similitud se puede ajustar a una representación común, considerando operaciones de aumento, reducción y rotación. Bai et al. (2017) proponen MVMDS (*Multi-View Multi-dimensional Scaling*) un método que busca crear un MDS a partir de múltiples matrices de similitud, para esto incorporan un ponderador

dentro de la función estrés para cada matriz de entrada.

2.4 STRESS

Modelar medidas de similitud (δ_{ij}) directamente como disparidades en una función de costo y permitir transformaciones métricas y no métricas de similitudes en disparidades $\hat{d}_ij$ son dos de los avances más importantes en MDS (Groenen y Borg, 2013). Shepard (1962a y 1962b) propuso un método heurísico para ambos aspectos sin proponer una función de costo. Fue Kruskal (1964) quien propuso la función de costo "*Stress*", ver ecuación (2.5) donde $\hat{d}_{ij}$ es una disparidad y d_{ij} es la distancia entre los objetos posicionados en el espacio de representación MDS y considerando que las matrices de distancia son triangulares inferiores y con diagonal 0, la sumatoria $\sum_{i<j}$ representa la suma de los elementos bajo la diagonal.

$$\textit{Stress}(X, \hat{d}) = \frac{\sum_{i<j}(\hat{d}_{ij} - d_{ij}(X))^2}{\sum_{i<j} d_{ij}^2(X)} \tag{2.5}$$

Donde $\hat{d}_{ij}$ es una medida de similitud transformada en disparidad. Por el momento se asume que $\hat{d}_{ij} = \delta_{ij}$. Entonces la función de de costo Stress ajusta la distancia $d_{ij}(X)$ directamente a las similitudes δ_{ij} y simplemente minimiza el error cuadrático para todas las combinaciones de i, j. Kruskal (1964) propuso un método basado en gradiente para obtener las coordenadas. Una de las transformaciones de similitudes es la transformación lineal $\hat{d}_{ij} = a + b\delta_{ij}$ para un a y b desconocidos. Con un intercepto positivo a y una pendiente b las medidas de similitud pueden ser reemplazas por disparidades, permitiendo una gran variedad de aplicaciones basadas en medidas de similitud entre objetos. Kruskal (1964) también propuso la transformación ordinal. Que considera que los $\hat{d}_{ij}$ deben ser escogidos de tal forma que $\delta_{ij} \leq \delta_{kl}$ implica que $\hat{d}_{ij} \leq \hat{d}_{kl}$ para cualquier combinación de pares ij y kl. Para una matriz X fija, la minimización de la ecuación (2.5) sobre $\hat{d}$ corresponde a una ecuación cuadrática con restricciones de desigualdad lineal en $\hat{d}$. Kruskal (1964) propuso una solución llamada regresión monótona que provee un mínimo global para este problema de optimización (Groenen y Borg, 2013).

Los algoritmos diseñados para resolver el problema de MDS utilizan distintas funciones de *stress*. En la literatura se han propuesto distintas funciones de *stress*: *Stress* (ecuación 2.6), *Raw stress* (ecuación 2.7), *Stress* normalizado (ecuación 2.8) y S-stress (ecuación 2.9) (Kruskal, 1964). Por ejemplo MDSCAL utiliza la optimización de gradiente para minimizar la función de *stress* (Tecuanhuehue-Vera, Carrasco-Ochoa y Martínez-Trinidad, 2012) (ecuación 2.6) y SMACOF utiliza mayorización de *stress* (De Leeuw, Barra, Brodeau, Romier y Van Cutsem,

1977; De Leeuw y Mair, 2009). Otras propuestas están basadas en heurística, como recocido simulado (Dzwinel, 1994), algoritmo genético (Tecuanhuehue-Vera et al., 2012; Dzidolikaite, 2015) y simulador virtual de partículas de N-cuerpos (Dzwinel, 1997).

Para explicar al lector el funcionamiento y relaciones entre funciones de *stress* se diseñó el siguiente experimento preliminar. Se creó una matriz X de 100×2 con valores aleatorios entre 0 y 1 (d_ij). Luego se calcularon mil matrices simulando disparidades ($\hat{d}_ij$) (triangulares inferiores) $\hat{d} \in \hat{D}$ de 100×100 con valores aleatorios entre 0 y 1, asumiendo que la disparidad entre los mismos objetos es 0 (diagonal en 0). Para cada $\hat{d} \in \hat{D}$ se calcularon las funciones de *stress*, raw stress, *stress* normalizado, y *S-Stress* con respecto a X. Como resultado se obtuvo la Figura 2.1 donde se presenta la correlación de Pearson entre las distintas funciones de *stress*, en la sección superior derecha de forma numérica, en la sección inferior izquierda como gráficos de dispersión y en la diagonal la distribución de las variables. Se observa que en general que las funciones *stress*, *stress* normalizado y *S-Stress* están correlacionadas positivamente entre sí, mientras *raw stress* presenta una menor correlación con todas las demás funciones. Este se debe a que la función de *raw stress* no considera ningún tipo de normalización.

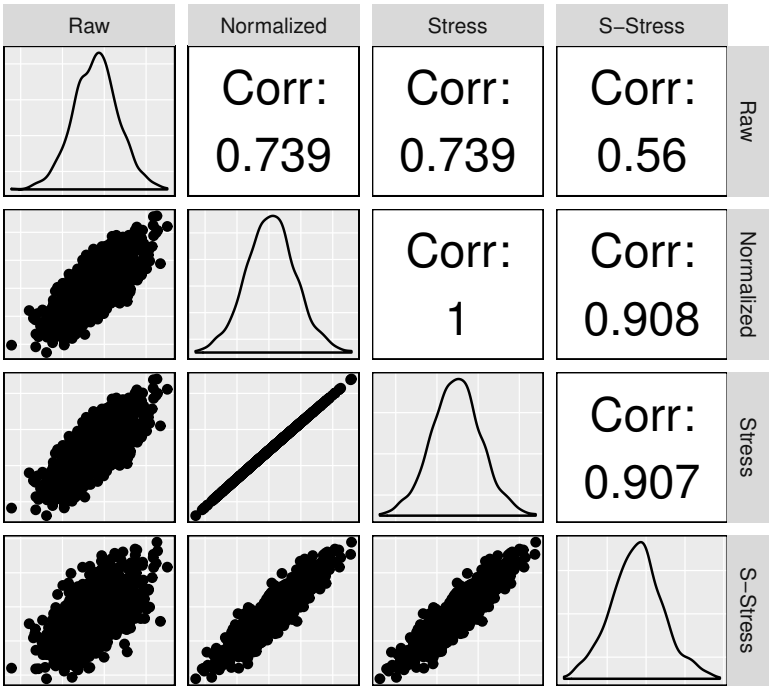


Figura 2.1: Correlación entre funciones de *stress* obtenida comparando 1.000 matrices de similitud creadas aleatoriamente. Fuente: Elaboración propia (2019).

$$Stress = \sqrt{\frac{\sum_{i<j} (d_{ij} - \hat{d}_{ij})^2}{\sum_{i<j} d_{ij}^2}} \tag{2.6}$$

$$S_{raw} = \sum_{i<j} (d_{ij} - \hat{d}_{ij})^2 \tag{2.7}$$

$$S_{norm} = \frac{\sum_{i<j} (d_{ij} - \hat{d}_{ij})^2}{\sum_{i<j} d_{ij}^2} \tag{2.8}$$

$$S - Stress = \sqrt{\frac{\sum_{i<j} (d_{ij}^2 - \hat{d}_{ij}^2)^2}{\sum_{i<j} (d_{ij}^2)^2}} \tag{2.9}$$

2.5 MDS ACTUAL

MDS es computacionalmente demandante. Para generar la visualización de los objetos es necesario minimizar el error entre la distancia de los objetos en un espacio de M dimensiones $\Omega = \{\omega_1, \omega_2 ... \omega_M\}$ y su proximidad en un espacio $X = \{x_1, x_2, ..., x_N\}$ de N dimensiones donde $N < M$. La función de error llamada función de *Stress* se presenta en la ecuación (2.10) donde D_{ij} es la matriz de similitud entre los objetos ω_i, ω_j en el espacio Ω, d es la matriz de distancia en el espacio X donde $d_{ij} = ((x_i - x_j)^T (x_i - x_j))^{1/2}$, k.

$$V(X) = \sum_{i<j} w_{ij} (D_{ij}^k - d_{ij}^k)^m \tag{2.10}$$

Asumiendo en la ecuación (2.10) que: $k = 1, m = 2$ y el peso $w_{ij} = 1/D_{ij}^k$ se obtiene de la versión de MDS clásico (ecuación 2.5). Para encontrar el mínimo de la ecuación (2.10) se debe resolver un sistema no lineal de ecuaciones de $N \times M$. El número de soluciones es infinito dado que la configuración objetivo de X no varía cuando es rotada. Además, la función de *stress* es multidimensional de $R^{NM} \rightarrow R^1$ que, usualmente, es multimodal. En general este problema no tiene solución directa, la complejidad aumenta junto con el número de mínimos locales de la ecuación (2.10). Puede ser resuelto parcialmente utilizando métodos de optimización como mayorización de *stress* (Kruskal, 1964; De Leeuw y Mair, 2009) o heurísticas como templado simulado (Dzwinel, 1994), algoritmo genético (Dzidolikaite, 2015; Tecuanhuehue-Vera et al., 2012), simulador de n cuerpos (Dzwinel, 1997), u otros. La complejidad en tiempo y memoria sigue limitada superiormente por $O(M^2)$ (Pawliczek, Dzwinel y Yuen, 2014). Para un computador moderno promedio generar una visualización de 10^4 objetos es demandante computacionalmente (Pawliczek et al., 2014).

2.6 MÉTODO CMDSCALE

El método *Cmdscale* del paquete *stats* de R (R Core Team, 2018) utiliza el procedimiento de MDS clásico que se describe en la Sección 2.1 con mejoras propuestas por Mardia (1978) que permiten utilizar la lógica del MDS clásico en matrices de distancia no euclidiana y estimar los valores faltantes.

2.7 MÉTODO SMACOF

El método *smacofSym* del paquete de R utiliza una estrategia llamada mayorización de *stress* que fue introducida por De Leeuw et al. (1977) y elaborada por De Leeuw y Heiser (1977), De Leeuw y Heiser (1980) y De Leeuw y Mair (2009). Suponiendo que se busca minimizar una función $f(x)$ para la que no es fácil obtener una solución analítica. El principio de mayorización sugiere buscar una función auxiliar $g(x, y)$ donde y es una constante llamada "punto de soporte" y se cumple la ecuación (2.11).

$$\forall x : f(x) \leq g(x, y) \tag{2.11}$$

La función auxiliar $g(x, y)$ debe tocar la superficie de $f(x)$ en el punto de soporte y, ver ecuación (2.12).

$$f(y) = g(y, y) \tag{2.12}$$

Teniendo $x^\star$ el valor que minimiza $g(x, y)$ sobre x se puede obtener la desigualdad (2.13) conocida como la "desigualdad del *sandwich*".

$$f(x^\star) \leq g(x^\star, y) \leq g(y, y) = f(y) \tag{2.13}$$

La mayorización es un proceso iterativo con los siguientes pasos:

1. Se escoge un valor inicial $y := y_0$
2. Se busca una actualización $x^{(t)}$ tal que $g(x^{(t),y)} \leq g(y, y)$
3. Si $f(y) - f(x^{(t)})$ se para, sino $y = x^{(t)}$
4. Se continua en el punto 2.

Este procedimiento puede ser extendido a espacios de múltiples dimensiones y siempre que se cumpla la desigualdad del *sandwich* (ecuación 2.13) puede ser utilizado para minimizar la función objetivo correspondiente. En MDS la función objetivo *stress* es una función multivariada de las disparidades y distancias entre objetos. SMACOF utiliza la mayorización para minimizar la función de *stress* (De Leeuw y Mair, 2009).

2.8 PROBLEMA DE OPTIMIZACIÓN DE MDS MULTI-OBJETIVO

Los métodos actuales de MDS utilizan solo una medida de similitud, sin embargo, existen problemas en los cuales se requiere analizar dos o más medidas de similitud conflictivas entre sí simultáneamente, además de la existencia de múltiples métricas que difieren entre sí para medir una misma similitud entre objetos y estudios que analizan por separado distintas medidas de similitud. Hacen necesaria una técnica capaz de combinar distintas medidas de similitud en una representación MDS en la cual se pueda observar la combinación e interacción de distintas medidas de similitud.

El problema de optimización multi-objetivo para el problema de MDS o MDSMO es definido como la obtención de representaciones MDS a partir de dos medidas de similitud, minimizando la función de *stress* sobre dos matrices de disparidades al mismo tiempo ($\hat{d}_{1ij}$ y $\hat{d}_{2ij}$), resultando en una o varias representaciones para cada matriz de entrada y su combinación. El problema de MDSMO puede ser definido según la ecuación (2.14), donde x es una solución (una representación en dos dimensiones) en un conjunto de todas las posibles soluciones X, y $z = \vec{f}(x)$ es un objetivo, donde f_1 corresponde a la evaluación de la función de *stress* usando una disparidad $\hat{d}_{1ij}$, y f_2 la función de *stress* normalizado es aplicada sobre una segunda disparidad $\hat{d}_{2ij}$.

$$\text{minimizar } \vec{z} = \vec{f}(x) = (f_1(x), f_2(x)), x \in X \tag{2.14}$$

2.9 OPTIMIZACIÓN MULTI-OBJETIVO

Un problema de optimización mono-objetivo optimiza sólo una función objetivo. En cambio un problema de optimización multi-objetivo considera múltiples objetivos, dos o tres criterios, los cuales usualmente tienen algún grado de conflicto (Villalobos-Cid, Dorn, Ligabue-Braun e Inostroza-Ponta, 2019).

En la optimización multi-objetivo (MOO), por sus siglas en inglés (*MultiObjective Optimization*) tiene relación con la optimización de problemas con múltiples objetivos que a menudo son parcial o completamente conflictivos entre sí. Handl, Kell y Knowles (2007) dividen los MOO en cinco categorías. La primera categoría es MOO estándar donde todos los objetivos son claros y las soluciones se encuentran en la frontera de Pareto. La segunda es contrapeso de sesgo, donde MOO es utilizada para disminuir el sesgo de una métrica en un problema de un objetivo. La tercera categoría integración de múltiples fuentes, MOO es utilizada para integrar datos con ruido desde múltiples fuentes. La cuarta categoría *performance approximation by proxies* es usada cuando una función de un objetivo no es optimizable y se hace necesario utilizar otra función intermedia que se aproxime. La quinta y última categoría *multiobjectivisation* (MOO) es utilizada para obtener una mejor "guía de búsqueda" en un problema de un objetivo donde la búsqueda de un óptimo global se ve dificultada por el exceso de mínimos locales en el espacio de búsqueda y la existencia de regiones en el espacio de búsqueda donde la gradiente se indefine. Según esta clasificación el problema de MDS multi-objetivo pertenecería a las categorías tercera y quinta.

2.9.1 Solución no dominada

Se dice que una solución es no dominada cuando es mejor con respecto a cualquiera de los objetivos al compararla con todas las demás soluciones. Si se minimizan los objetivos $f1$ y $f2$ y la solución a tiene valores $a_{f1} = 1$ y $a_{f2} = 0,5$ y la solución b valores $b_{f1} = 0,9$ y $b_{f2} = 0,5$, entonces b no es dominada por a porque $b_{f1} \leq a_{f1}$ y $b_{f2} \leq a_{f2}$. Esto se puede ver gráficamente en la Figura 2.2 donde las soluciones no dominadas están representadas con un círculo y las soluciones dominadas con una cruz.

2.9.2 Frontera de Pareto

La frontera de Pareto corresponde al conjunto de todas las soluciones no dominadas. En la Figura 2.2 la frontera de Pareto corresponde a las soluciones representadas por un círculo y unidas por una línea punteada.

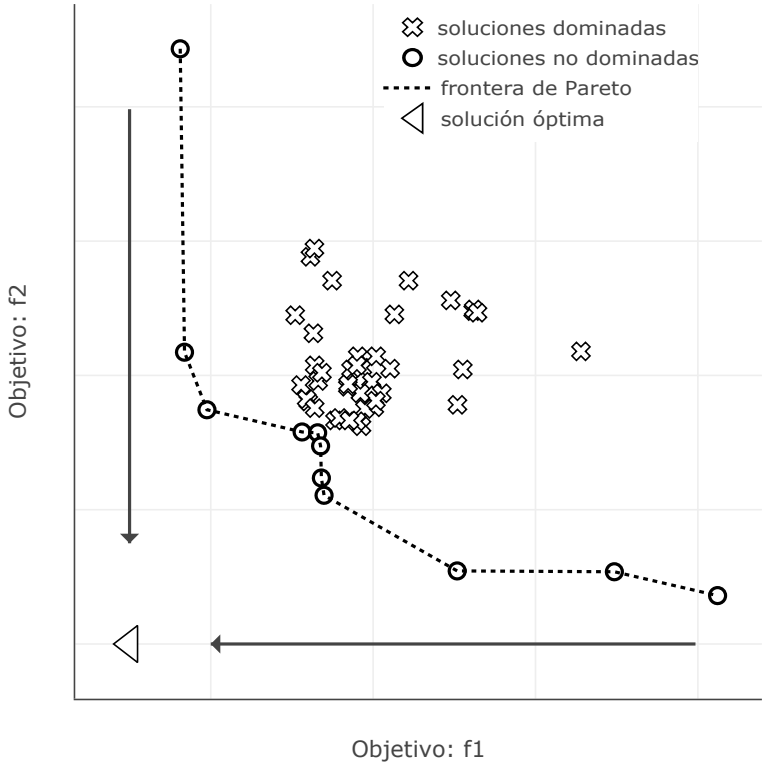


Figura 2.2: Conjunto de soluciones de Pareto para un problema multi-objetivo. Fuente: Elaboración propia (2019).

2.10 ALGORITMO NSGA-II

NSGA-II (del inglés, *Non-dominated Sorting Genetic Algorithm II*) es un algoritmo genético de ordenamiento no dominado; es utilizado para encontrar soluciones pertenecientes al conjunto de Pareto en problemas de optimización multi-objetivo . Deb et al. (2002) introduce un enfoque de ordenamiento no dominado rápido elitista para mantener las soluciones no dominadas y un operador de hacinamiento para mantener la diversidad de las soluciones (Deb et al., 2002).

2.10.1 Ordenamiento no dominado

El ordenamiento no dominado consiste en asignar un *ranking* de no dominación a las soluciones de acuerdo a la frontera de Pareto a la que pertenecen, para esto se obtienen las soluciones pertenecientes a la frontera de Pareto y se les asigna el ranking 1, luego excluyendo las soluciones con ranking ya asignado, se obtienen las soluciones pertenecientes a una nueva

frontera de Pareto y se les asigna el ranking 2, esta operación se repite incrementando el ranking en 1 hasta que todas las soluciones tienen un ranking de no dominancia asignado (Deb et al., 2002).

2.10.2 Distancia de hacinamiento

Para estimar la densidad de las soluciones que rodean a otra, se calcula para cada solución la longitud promedio de los lados del cuboide formado entre las dos soluciones más cercanas pertenecientes a la misma frontera de Pareto. Esta es la distancia de hacinamiento. A las soluciones en los extremos de cada frontera de Pareto se les asigna un valor infinito como distancia de hacinamiento, de esta forma siempre son seleccionadas (Deb et al., 2002).

2.10.3 Operador de comparación de hacinamiento

El operador de comparación de hacinamiento guía el proceso de selección. Si dos soluciones pertenecen a distintos *ranking* de no dominación, selecciona la solución con menor *ranking*. Si las soluciones pertenecen al mismo *ranking* de no dominación, prefiere la solución que se encuentra en un área con menor hacinamiento (Deb et al., 2002). En la Figura 2.3 se muestra una frontera de Pareto para los objetivos $f1$ y $f2$, la distancia de hacinamiento para el punto i se calcula como el perímetro del cuboide formado entre sus vecinos más cercanos $i-1$ e $i+1$, en el caso de los puntos 0 y 1 se les asigna un valor infinito.

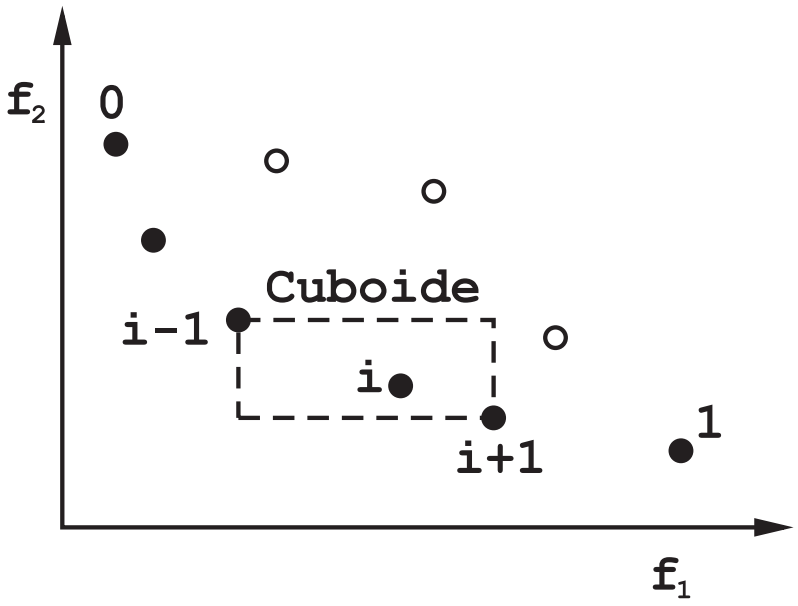


Figura 2.3: Distancia de hacinamiento del algoritmo NSGA-II. Fuente: Deb et al. (2002)

2.10.4 Ciclo principal

- Se crea una población P_0 de tamaño N aleatoriamente.
- Se asignan los rankings de no dominación.
- Se aplican los operadores de torneo binario, cruzamiento y mutación para crear una población Q_0 de tamaño N.
- Se forma una población $R_t = P_t \cup Q_t$ de tamaño $2N$.
- R_t se ordena de acuerdo a ranking de no dominancia.
- Se forma la población P_{t+1} agregando secuencialmente los conjuntos completos de soluciones pertenecientes a los menores rankings de dominancia, finalizando antes de agregar un conjunto incompleto.
- P_{t+1} se completa hasta tener tamaño N con los individuos con mayor distancia de hacinamiento del último conjunto que quedó sin agregar.
- La nueva población P_{t+1} de tamaño N es utilizada para crear una nueva población Q_{t+1}.

2.10.5 Ejemplo

En la Figura 2.4 se puede observar al lado izquierdo la población anterior P_t y Q_t que contiene los individuos creados con las operadores genéticos, los individuos de P_t y Q_t se juntan en R_t y se agrupan en los *rankings* de Pareto $F = F_1, F_2, ..., F_n$, luego se crea la población P_{t+1} con los primeros *rankings* hasta que no se pueda agregar un *ranking* completo sin superar el tamaño de la población N, en este caso son las fronteras $F_1 y F_2$. Finalmente, se agregan a p_{t+1} los individuos pertenecientes a F_3 con mayor distancia de hacinamiento hasta completar el tamaño de la población N.

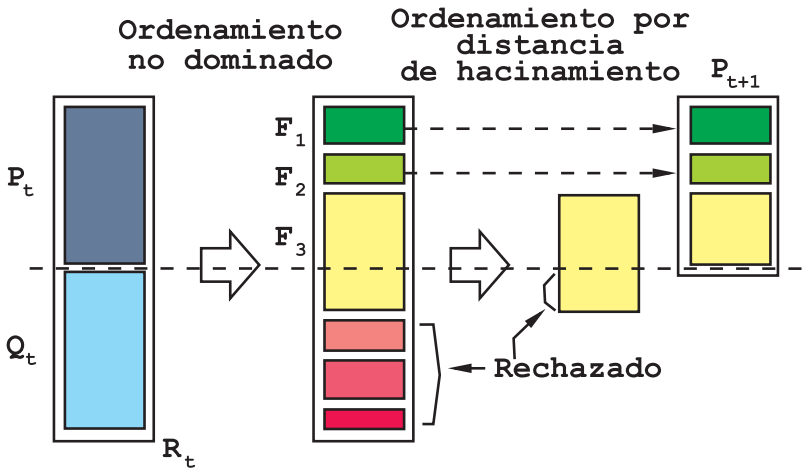


Figura 2.4: Ordenamiento no dominado del algoritmo NSGA-II. Fuente: Deb et al. (2002)

2.11 MÉTRICAS DE EVALUACIÓN MULTI-OBJETIVO

A diferencia de las estrategias mono-objetivo (SO) que buscan minimizar o maximizar un objetivo único, donde se puede comparar la calidad de las soluciones directamente, las estrategias multi-objetivo (MO) buscan minimizar o maximizar más de un objetivo, por lo que la calidad de las soluciones se debe medir en relación a todos los objetivos.

2.11.1 Hipervolumen

El hipervolumen (HV) es una métrica para evaluar la calidad de las soluciones multi-objetivo, corresponde al espacio dominado por un conjunto de soluciones pertenecientes a la

frontera de Pareto con respecto a un punto de referencia. Es la métrica más utilizada para comparar el rendimiento de estrategias multi-objetivo (Riquelme, Von Lücken y Baran, 2015). En la Figura 2.5 se puede ver el área (color azul) entre las soluciones pertenecientes a la frontera de Pareto y un punto de referencia, esta área corresponde a la métrica de hipervolumen.

En este trabajo se utiliza el método *computeHV* del paquete erc (Bossek, 2017) de R para calcular la métrica de hipervolumen.

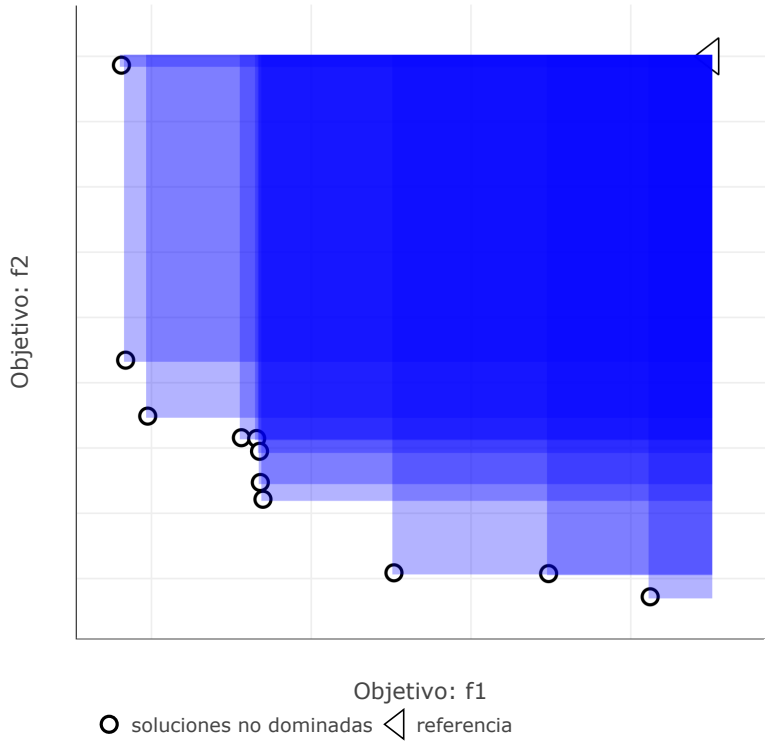


Figura 2.5: Hiper volumen para un conjunto de soluciones de Pareto. Fuente: Elaboración propia (2019)

2.11.2 Contribución de hipervolumen por punto

La contribución de hipervolumen por punto corresponde al área que aporta cada punto al área dominada con respecto a un punto de referencia, en la Figura 2.5 corresponde al área desde un punto de la frontera de Pareto hasta el punto de referencia. Para este trabajo se calcula la contribución de HV con el método *computeHVContr* del paquete erc (Bossek, 2017) de R.

CAPÍTULO 3. ALGORITMO PROPUESTO

Se propone un algoritmo evolutivo multi-objetivo basado en el algoritmo NSGA-II (Deb et al., 2002) (por sus siglas en inglés: *Non-sorting dominated sorting genetic algorithm II*) para abordar el problema de MDSMO. NSGA-II es uno de los algoritmos más utilizados en problemas de optimización multi-objetivo, incluye una estrategia para aumentar la diversidad de las soluciones (Villalobos-Cid, Dorn e Inostroza-Ponta, 2018a).

El pseudo-código del algoritmo propuesto se puede ver en el Algoritmo 3.1, donde $\hat{d}_{1ij}$ y $\hat{d}_{2ij}$ corresponden a matrices de similitud, ps es el tamaño de la población, cr y mr son los ratios de entrecruzamiento y mutación respectivamente. Los parámetros r y p_{mr} están asociados al operador de mutación. En las siguientes secciones se describen los detalles del algoritmo.

Algoritmo 3.1: Propuesta basada en NSGA-II.

entrada: $\hat{d}_{1ij}$, $\hat{d}_{2ij}$, ps, cr, mr, r, p_{mr}
salida : P población de representaciones MDS

```
1  a
                                                  // Crear población inicial
2  P ← CreaPoblaciónInicial (d̂1ij,d̂2ij,ps)
3  mientras no se cumpla la condición de término hacer
4      para cada p ∈ P hacer
5          inicio Operaciones genéticas
6              [S1, S2] ← SelecciónPorTorneo (P)
7              Q[p] ← Cruzamiento (S1,S2,cr)
8              Q[p] ← Mutación (Q[p],mr,r,pmr)
9          fin
10     fin
                                              // Ordenar soluciones por dominancia
11     P ← OrdenamientoNoDominado (P,Q,ps)
12 fin
13 devolver (P)
```

3.1 CRITERIOS DE OPTIMALIDAD

El algoritmo propuesto recibe como entrada dos matrices de similitud ($\hat{d}_{1ij}$ y $\hat{d}_{2ij}$) que contienen una medida de similitud entre n objetos. Una **solución** corresponde a una representación de los objetos en el espacio euclidiano. Las funciones objetivo $f1$ y $f2$ buscan minimizar el error entre la distancia entre la representación de los objetos en el espacio euclidiano y las medidas de similitud en las matrices $\hat{d}_{1ij}$-$\hat{d}_{2ij}$ utilizando la función de *stress* normalizado (Ecuación 2.8). Esta función es independiente de la escala de las entradas.

3.2 POBLACIÓN INICIAL

Se implementaron cinco estrategias para crear la población inicial P con ps soluciones:

1. Método aleatorio. Esta estrategia posiciona aleatoriamente los n objetos en el espacio euclidiano. Se repite ps veces.
2. Método radial. Se selecciona aleatoriamente un punto de referencia y una matriz de similitud. Los puntos se posicionan a una distancia igual a la similitud de la matriz seleccionada en un ángulo aleatorio.
3. Método PCA. Las ps soluciones son creadas utilizando análisis de componentes principales seleccionando aleatoriamente una de las matrices de entrada.
4. Método *Cmdscale*. Las ps soluciones son creadas utilizando *Cmdscale* utilizando una matriz de entrada aleatoria.
5. Método *Cmdscale* ponderado. Crea dos soluciones *Cmdscale* y pondera sus coordenadas aleatoriamente. Se repite ps veces.

Estas estrategias son comparadas en la etapa de parametrización, donde una de las estrategias es seleccionada para configurar el algoritmo propuesto. Luego de crear la población inicial P, una segunda población Q es construida utilizando operadores genéticos.

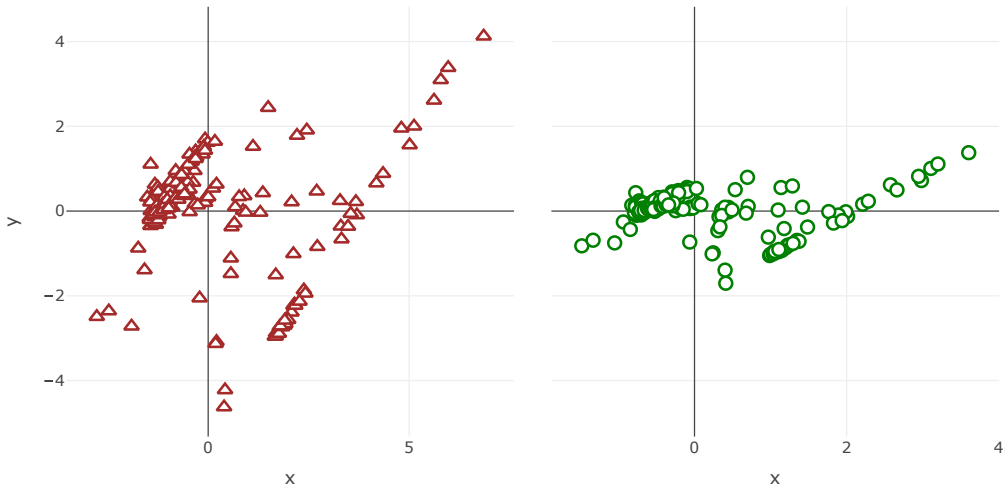
3.3 OPERADOR DE CRUZAMIENTO

3.3.1 Selección de padres

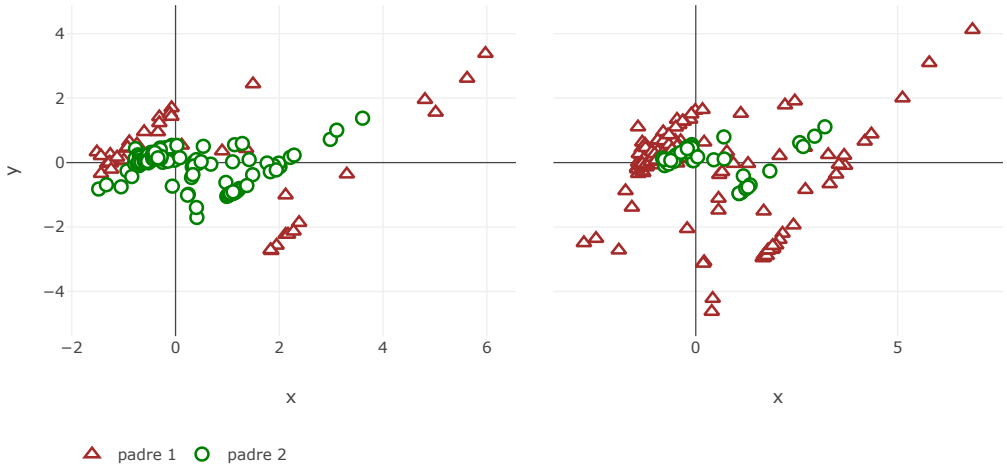
El operador de cruzamiento selecciona dos de las soluciones utilizando una selección por torneo-4 estocástico (S_1, S_2) como padres. El torneo-4 estocástico consiste en seleccionar cada padre como la solución con mejor *fitness* de cuatro soluciones obtenidas aleatoriamente.

3.3.2 Cruzamiento

Un porcentaje cr de los puntos de cada padre es intercambiado para crear dos nuevas soluciones (descendientes), véase Figura 3.1. Se calcula el *stress* de las nuevas soluciones y se selecciona la solución dominante. Si ambas soluciones son no-dominadas, se selecciona una al azar. Esta operación es repetida ps veces.



(a) Padres.



(b) Descendientes

Figura 3.1: Operador de cruzamiento, conjunto de datos glass. Fuente: Elaboración propia (2019).

En la Figura 3.1 se puede observar un ejemplo del operador de cruzamiento para el conjunto de datos glass. En la primera etapa se seleccionan dos soluciones como padres como se indica en la Sub-sección 3.3.1, ver Figura 3.1a. Luego se selecciona un porcentaje cr de puntos del padre 1 (Figura 3.1a, triángulos rojos) y se traspasa al hijo 1 (Figura 3.1b, izquierda) y el

complemento $(1 - cr)$ se traspasa al hijo 2 (Figura 3.1b derecha). Luego se traspasan $cr - 1$ objetos del padre 2 (Figura 3.1a, círculos verdes) al hijo 1 (Figura 3.1b, izquierda) y cr al hijo 2 (Figura 3.1b derecha). De esta forma se crean dos nuevas soluciones con el mismo número de elementos.

3.4 OPERADOR DE MUTACIÓN

El operador de mutación es aplicado de acuerdo al ratio mr. Cuando una solución es seleccionada, p_{mr} puntos del espacio euclidiano son modificados aleatoriamente en un ratio r entre $(1 - r)/2$ y $(1 + r)/2$. De esta forma se incorpora ruido aleatorio a la solución.

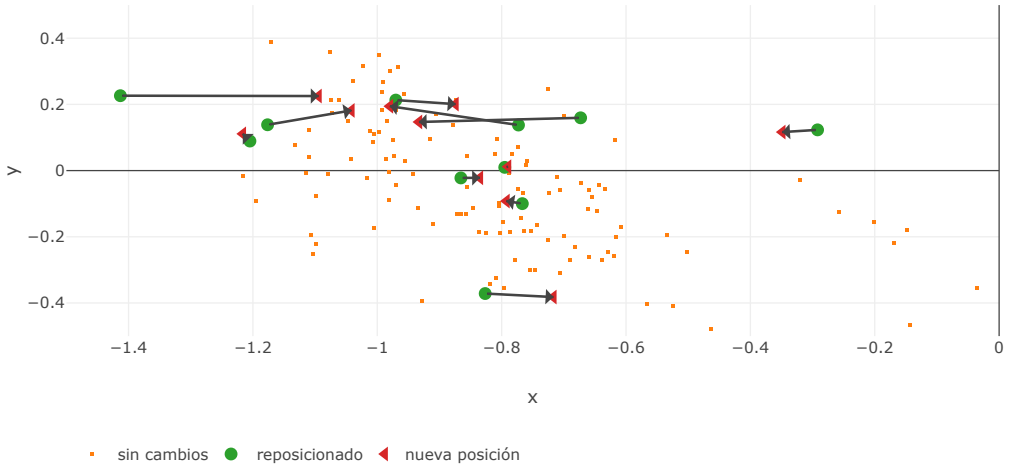


Figura 3.2: Operador de mutación, Fuente: Elaboración propia (2019).

En la Figura 3.2 se puede observar una solución del conjunto de datos *glass* seleccionada para mutar de acuerdo al ratio mr. Los puntos amarillos corresponden a la posición original de los objetos. Los puntos verdes son los objetos que fueron seleccionados de acuerdo al ratio p_{mr} y se desplazan aleatoriamente de acuerdo a r, siguiendo la trayectoria de las flechas negras hasta una nueva posición demarcada por un triángulo rojo.

3.5 SELECCIÓN DE SOLUCIONES

Finalmente, las poblaciones P y Q son combinadas seleccionando los mejores ps individuos para la siguiente generación. Las soluciones son seleccionadas utilizando

ordenamiento no dominado y distancia de hacinamiento (*crowding distance* en inglés) (Deb et al., 2002). El ordenamiento no dominado consiste en encontrar la frontera de Pareto como se muestra en la Figura (2.2), asignarle el primer nivel a las soluciones pertenecientes a la frontera de Pareto, luego se excluyen estas soluciones y se repite el proceso asignando el nivel siguiente a cada frontera de Pareto. Luego sobre cada frontera se calcula la distancia de hacinamiento; la distancia de hacinamiento es una métrica que estima la densidad de las soluciones cercanas a una solución. Se busca que las soluciones seleccionadas sean no dominadas, es decir que cumplan con los objetivos $f1$ y $f2$ mejor que las demás soluciones y que las soluciones sean distintas entre sí, lo que mejora la probabilidad de encontrar nuevas soluciones y disminuye la posibilidad de que el algoritmo quede estancado en un mínimo local.

Todas las etapas anteriores son repetidas hasta que se cumple con la condición de término.

CAPÍTULO 4. MATERIALES Y MÉTODOS

4.1 MATERIALES

4.1.1 Herramientas de desarrollo

Para la construcción del algoritmo genético se utiliza **R** (R Core Team, 2018) versión 3.5.0. **R** es un lenguaje de programación y un entorno para computación estadística y gráficos. Es un proyecto GNU que provee una amplia variedad de técnicas estadísticas como: modelamiento lineal y no lineal, pruebas estadísticas clásicas, análisis de series de tiempo, *clustering*, y técnicas gráficas, entre otras, incluyendo distintas implementaciones de MDS. Una de sus fortalezas es que puede producir fácilmente gráficos de calidad, incluyendo fórmulas y símbolos matemáticos. Es flexible, sus funcionalidades pueden ser extendidas instalando paquetes desde CRAN (por sus siglas en inglés: *Comprehensive R Archive Network*), un repositorio mantenido por la fundación de R (Foundation, 2019).

Para el almacenamiento de resultados se utiliza una base de datos MySQL edición de la comunidad (licencia GLP) en su versión 14.14. MySQL es la base de datos de código abierto más popular en el mundo. Con comprobado rendimiento, confiabilidad y facilidad, MySQL se ha convertido en la primera opción de base de datos para aplicaciones web, es usada por Facebook, Twitter, Youtube, Yahoo!, entre otras (MySQL, 2019).

Los conjuntos de datos serán obtenidos desde el repositorio UCI. El repositorio UCI es una colección de bases de datos, teorías de dominio y generadores de datos que son utilizados por la comunidad de aprendizaje de máquina para el análisis empírico de algoritmos de aprendizaje de máquina. Este archivo fue creado en 1987 por David Aha y otros estudiantes de la Universidad de California. Desde su creación ha sido ampliamente utilizado por estudiantes, educadores e investigadores alrededor del mundo como la principal fuente de conjuntos de datos para aprendizaje de máquina. Como un indicador del impacto de este repositorio, ha sido citado más de 1.000 veces, alcanzando un lugar dentro de los 100 “papers” más citados en ciencias de la computación (Dua y Graff, 2017).

Para documentar se utiliza el editor de documentos en línea de overleaf. Overleaft es un *startup* y una empresa social que construye herramientas de creación de documentos colaborativas. Su principal producto es un editor de documentos escritos en el lenguaje de etiquetas para composición tipográfica LaTeX en tiempo real y colaborativo (Overleaf, 2019). LaTeX es un sistema de composición de documentos que incluye funcionalidades diseñadas para la producción de documentación técnica y científica, LaTeX es el estándar *de facto* para

la comunicación y publicación de documentos científicos. Está disponible bajo una licencia de software gratuito (LaTeX, 2019).

El control de versiones se realizará con git. Git es un sistema de control de versiones libre y de código abierto (Git, 2019).

4.1.2 Ambiente de desarrollo

El ambiente de desarrollo es local en un computador portátil con sistema operativo OSX, procesador de doble núcleo Intel Core i5 de 2,6 GHz y 8GB de memoria RAM.

4.1.3 Conjuntos de datos

Para evaluar el algoritmo propuesto se utilizaron siete conjuntos de datos (Tabla 4.1). Cinco de ellos corresponden a conjuntos de datos conocidos utilizados en la literatura para aplicar métodos de minería de datos y aprendizaje de máquina (*Glass*, *Iris*, *Pima Indians Diabetes*, *Breast Cancer Wisconsin*) e *Ionosphere* (Dua y Graff, 2017). Los demás conjuntos de datos *Flu Trees* (Jombart, Kendall, Almagro-Garcia y Colijn, 2017) y *Hospitals2014* (Villalobos-Cid, Chacón, Zitko y Inostroza-Ponta, 2016) han sido estudiados usando dos diferentes medidas de similitud en estudios aplicados.

Tabla 4.1: Conjuntos de datos utilizados en experimentos. D_1 y D_2 corresponden a la distancia utilizada: Eu: euclideana, CB: city-block, RF: Robinson-Foulds, KF: branch score y Co: correlación. Fuente: Elaboración propia (2019).

Conjunto de datos	Nº Obs.	Nº Atr.	D_1	D_2	Referencia
Glass	214	10	Eu	CB	(Dua y Graff, 2017)
Iris	150	4	Eu	CB	(Dua y Graff, 2017)
Pima Indians Diabetes	768	9	Eu	CB	(Dua y Graff, 2017)
Breast Cancer Wisconsin	569	32	Eu	CB	(Dua y Graff, 2017)
Ionosphere	351	34	Eu	CB	(Dua y Graff, 2017)
Flu Trees	200	165	RF	KF	(Jombart et al., 2017)
Hospitals2014	187	683	Eu	Co	(Villalobos-Cid et al., 2016)

Conjunto de datos glass

El conjunto de datos *Glass* contiene 214 muestras de vidrios, con las siguientes características: índice de refracción, cantidad presente de: sodio, magnesio, aluminio, silicio, potasio, calcio, bario y hierro. La recolección de este conjunto de datos fue motivada por investigaciones criminológicas (Dua y Graff, 2017). Los tipos de vidrios se dividen en las siguientes categorías:

1. ventana de edificio de vidrio flotado
2. ventana de edificio de vidrio no flotado
3. ventana de vehículo de vidrio flotado
4. ventana de vehículo de vidrio flotado (ninguno presente en el conjunto de datos)
5. contenedores
6. vajilla
7. lámparas.

Conjunto de datos iris

El conjunto de datos *iris* está compuesto por tres clases: lirio *setosa*, *versicolour* y *virginica*; por cada clase hay 50 instancias en el conjunto de datos, sus atributos son: longitud del sépalo (cm), ancho del sépalo (cm), longitud del pétalo (cm) y ancho del pétalo (cm).

Conjunto de datos pima indians diabetes

El conjunto de datos pima indians diabetes contiene 768 instancias de diagnósticos de diabetes practicados a mujeres de al menos 21 años, con antecedentes familiares, con residencia cercana a Phoenix, Arizona, Estados Unidos, con un resultado binario, positivo o negativo de acuerdo al criterio de la organización mundial de la salud (Dua y Graff, 2017). Los atributos son los siguientes:

1. número de embarazos
2. concentración de glucosa en la sangre a las 2 horas de una prueba de tolerancia a la glucosa oral
3. presión arterial diastólica (mm Hg)
4. grosor del pliegue de la piel del tríceps
5. insulina sérica de 2 horas (mu U / ml)
6. índice de masa corporal (peso en kg / (altura en m^2)
7. probabilidad de diabetes basada en antecedentes familiares
8. edad (años).

Conjunto de datos breast cancer Wisconsin

El conjunto de datos *breast cancer Wisconsin* tiene relación con el cáncer de mamas, se extrajeron características de tumores desde imágenes digitalizadas de un aspirado con aguja fina (FNA) de una masa mamaria (Dua y Graff, 2017). Los tumores se dividen en dos categorías, benigno o maligno, las características extraídas son las siguientes:

1. radio (media de las distancias desde el centro a los puntos del perímetro)
2. textura (desviación estándar de escalas de grises)
3. perímetro
4. regularidad
5. densidad
6. concavidad
7. número de porciones cóncavas en el contorno
8. simetría
9. dimensión fractal.

Conjunto de datos ionosphere

La ionosfera es una parte de la atmósfera terrestre. El conjunto de datos *ionosphere* contiene datos acerca de señales electromagnéticas enviadas por un sistema de antenas en el área de Goose Bay en Canadá, en busca de electrones libres en la ionosfera. Tiene 351 instancias divididas en dos clases, buena y mala; buena es cuando se encontró algún tipo de estructura en la ionosfera y mala cuando las señales traspasaron la ionosfera. Cada instancia tiene 34 atributos con valores continuos asociados a las señales recibidas (Dua y Graff, 2017).

Conjunto de datos fluTrees

Un árbol filogenético contiene la información acerca de la relación evolutiva de conjuntos de organismos. El conjunto de datos fluTrees contiene 200 árboles filogenéticos creados utilizando el software BEAST (Suchard et al., 2018) sobre secuencias de hemaglutinina (HA) en muestras de influenza estacional A/H3N2 recolectada en la ciudad de Nueva York en Estados Unidos entre los años 2000 y 2003 (Jombart et al., 2017). Cada árbol tiene 165 ramas.

Conjunto de datos hospitales2014

Este conjunto de datos contiene 193 hospitales chilenos con 683 atributos asociados a producción y finanzas del año 2014. Fue recolectado por Villalobos-Cid et al. (2016) desde el sitio web del Departamento de Estadística e Información de Salud del Ministerio de Salud de Chile (www.deis.cl) en febrero de 2015 (Villalobos-Cid et al., 2016). Los hospitales están clasificados en cinco categorías de acuerdo a complejidad:

1. baja complejidad
2. alta complejidad adulto
3. mediana complejidad
4. alta complejidad pediatría
5. mediana complejidad psiquiatría.

4.2 PARAMETRIZACIÓN

En una primera etapa se parametrizó el algoritmo propuesto utilizando el paquete *Irace* de R (López-Ibáñez, Dubois-Lacoste, Cáceres, Birattari y Stützle, 2016). *Irace* es un método de configuración automática de algoritmos de optimización que busca la mejor configuración de parámetros dado un conjunto de instancias del problema, para esto implementa una carrera iterativa que es una extensión del procedimiento F-race (Birattari, Stützle, Paquete y Varrentrapp, 2002) y aplica pruebas estadísticas al final de cada iteración (López-Ibáñez et al., 2016). *Irace* toma como entradas una definición del espacio de parámetros del algoritmo a ajustar, un conjunto de instancias del problema, y las opciones de configuración (López-Ibáñez et al., 2016). Luego busca en el espacio de parámetros configuraciones con mejor rendimiento, ejecutando el algoritmo objetivo con distintas configuraciones e instancias, evaluando los resultados y aplicando una prueba estadística para seleccionar las mejores configuraciones, repitiendo este proceso hasta que una condición de término definida es alcanzada (López-Ibáñez et al., 2016).

Para evaluar el algoritmo propuesto el espacio de parámetros se definió dentro de los siguientes intervalos: generaciones [1:500], tamaño de población ps [1:500], método de inicialización (ver Capítulo 3), ratio de cruzamiento cr [0.0,1.0], ratio de mutación mr [0.0,1.0], rango de mutación r [0.0,2.0], y probabilidad de mutación p_{mr} [0.0,1.0]. Como instancias de pruebas se utilizaron los conjuntos de datos *glass*, *iris* y *FluTrees*. La métrica de evaluación de rendimiento se realizó con la métrica de hipervolumen calculado sobre el *stress* normalizado de cada objetivo. Como condición de término se definió un presupuesto de 1.000 ejecuciones .Se aplicó la prueba de Friedman o *F-test*.

4.3 EVALUACIÓN DE RENDIMIENTO

No existen estrategias en la literatura para resolver el problema de MDS multi-objetivo, por lo que se compara el rendimiento del algoritmo propuesto con soluciones obtenidas con los métodos de optimización mono-objetivo. Se seleccionaron dos estrategias recomendadas por la literatura: una versión mejorada del método MDSCAL (*Cmdscale*, véase Sección 2.6) (R Core Team, 2018) y SMACOF (Sección 2.7) (Vermeesch, 2019). Ambos métodos se aplican sobre las entradas $\hat{d}_{1ij}$-$\hat{d}_{2ij}$. El objetivo es determinar si la estrategia MO puede encontrar nuevas soluciones dominantes u obtener representaciones intermedias que combinen las características de cada entrada.

La métrica de hipervolumen (HV) (ver Sección 2.11) corresponde al espacio

dominado por un conjunto de soluciones (frontera de Pareto), es la métrica más utilizada para comparar el rendimiento de estrategias multi-objetivo (Riquelme et al., 2015). Sin embargo, HV no se puede utilizar directamente para comparar estrategias MO y SO (Ishibuchi, Nojima y Doi, 2006). Esto porque para las estrategias SO las soluciones son un punto por cada objetivo y para las estrategias MO es una frontera de Pareto con múltiples soluciones.

Para realizar una comparación justa entre las estrategias MO y SO se utiliza el procedimiento descrito por Villalobos-Cid et al. (2018b) donde se utiliza la contribución de hipervolumen por punto dividida por la suma acumulada de la contribución de hipervolumen para todas las soluciones en la frontera de Pareto de la estrategia MO y las soluciones obtenidas por la estrategia SO. Como resultado se obtiene una contribución porcentual de hipervolumen por cada solución MO y SO, lo que permite comparar ambas estrategias.

Si la estrategia multi-objetivo encuentra soluciones que dominan a todas las soluciones encontradas por las estrategias mono-objetivo, la contribución de HV alcanza el 100 %. En el caso contrario, si la estrategia multi-objetivo es dominada completamente por las soluciones mono-objetivo la contribución de HV es 0 %. Si la frontera de Pareto contiene soluciones de ambas estrategias mono y multi objetivo, la contribución de HV es proporcional a la contribución de cada método para construir la frontera de Pareto (Ishibuchi et al., 2006). Cada estrategia se ejecuta 31 veces.

4.4 SELECCIÓN DE SOLUCIONES

La selección de soluciones se realiza de la siguiente forma. Con respecto a un objetivo f se selecciona la solución con menor valor de f. Para elegir una solución con respecto a los objetivos $f1$ y $f2$ se utiliza la métrica L_2 (Villalobos-Cid, Vega-Araya e Inostroza-Ponta, 2017) que corresponde a la distancia euclideana entre la solución y un punto de referencia en el espacio objetivo. Se utiliza como referencia el punto $(0, 0)$ y se selecciona la solución con menor L_2.

CAPÍTULO 5. RESULTADOS EXPERIMENTALES

5.1 PARAMETRIZACIÓN

Utilizando la estrategia *Irace* para obtener los parámetros que optimizan HV sobre el *stress* normalizado de ambos objetivos se obtuvieron los siguientes valores: $generaciones = 458$, $ps = 345$, método de inicialización = Cmdscale ponderado, $cr = 0{,}8$, $mr = 0{,}9$, $r = 0{,}6$, y $p_{mr} = 0{,}07$. Estos parámetros fueron utilizados en las pruebas de rendimiento del algoritmo propuesto.

5.2 EVALUACIÓN DE RENDIMIENTO

La Tabla 5.1 resume la contribución de HV porcentual por punto de las estrategias mono y multi-objetivo para cada conjunto de datos en las 31 ejecuciones. En todas ejecuciones el método multi-objetivo alcanza un 100 % de contribución porcentual de HV, dominando las soluciones mono-objetivo. Esto significa que no existen soluciones multi-objetivo dominadas por soluciones mono-objetivo en ningún conjunto de datos.

Los valores en la Tabla 5.2 muestran la mediana de las soluciones obtenidas ejecutando las estrategias SO y MO. En el caso de la estrategia multi-objetivo, la ejecución mediana fue determinada en base a HV. Para los métodos mono-objetivo las medianas fueron seleccionadas utilizando los valores de f_1 y f_2. De acuerdo a estos resultados, la estrategia multi-objetivo obtuvo el menor *stress* con respecto a f_1 (entrada $\hat{d}_{1ij}$) (Tabla 5.2, columnas 1-2, valores sombreados), dominando todas las soluciones obtenidas por las estrategias mono-objetivo que optimizan la misma función. Para f_2 (entrada $\hat{d}_{2ij}$) la estrategia multi-objetivo no pudo encontrar nuevas soluciones con respecto a *Cmdscale*; sin embargo, dominó todas las soluciones obtenidas por SMACOF (Tabla 5.2, columnas 5-6, valores sombreados).

Para todos los conjuntos de datos el enfoque MO encontró soluciones intermedias que representan el *tradeoff* entre $f1$ y $f2$ (Tabla 5.2, columnas 3-4). Estas soluciones intermedias fueron seleccionadas utilizando la métrica L2 (Villalobos-Cid et al., 2017). La Figura 5.1 muestra los resultados asociados al conjunto de datos *glass*. Las soluciones obtenidas con un enfoque MO dominan a las soluciones generadas por los enfoques SO, encontrando la misma solución que *Cmdscale* con respecto al objetivo f_2.

Los resultados para el conjunto de datos *glass* para la ejecución mediana se pueden observar en la Figura 5.1. En la Figura 5.1a se encuentran las soluciones pertenecientes a la

Tabla 5.1: Contribución de HV promedio obtenido utilizando las estrategias mono y multi-objetivo, considerando 31 ejecuciones. Fuente: Elaboración propia (2019).

Resultados	Conjuntos de datos						
Contribución de HV (%)	Glass	Iris	PID	BCW	Io	FT	H
cmdscale	0	0	0	0	0	0	0
SMACOF	0	0	0	0	0	0	0
Enfoque MO	100	100	100	100	100	100	100

Tabla 5.2: Soluciones para la ejecución mediana obtenida por las estrategias mono y multi-objetivo para todos los conjuntos de datos. Fuente: Elaboración propia (2019).

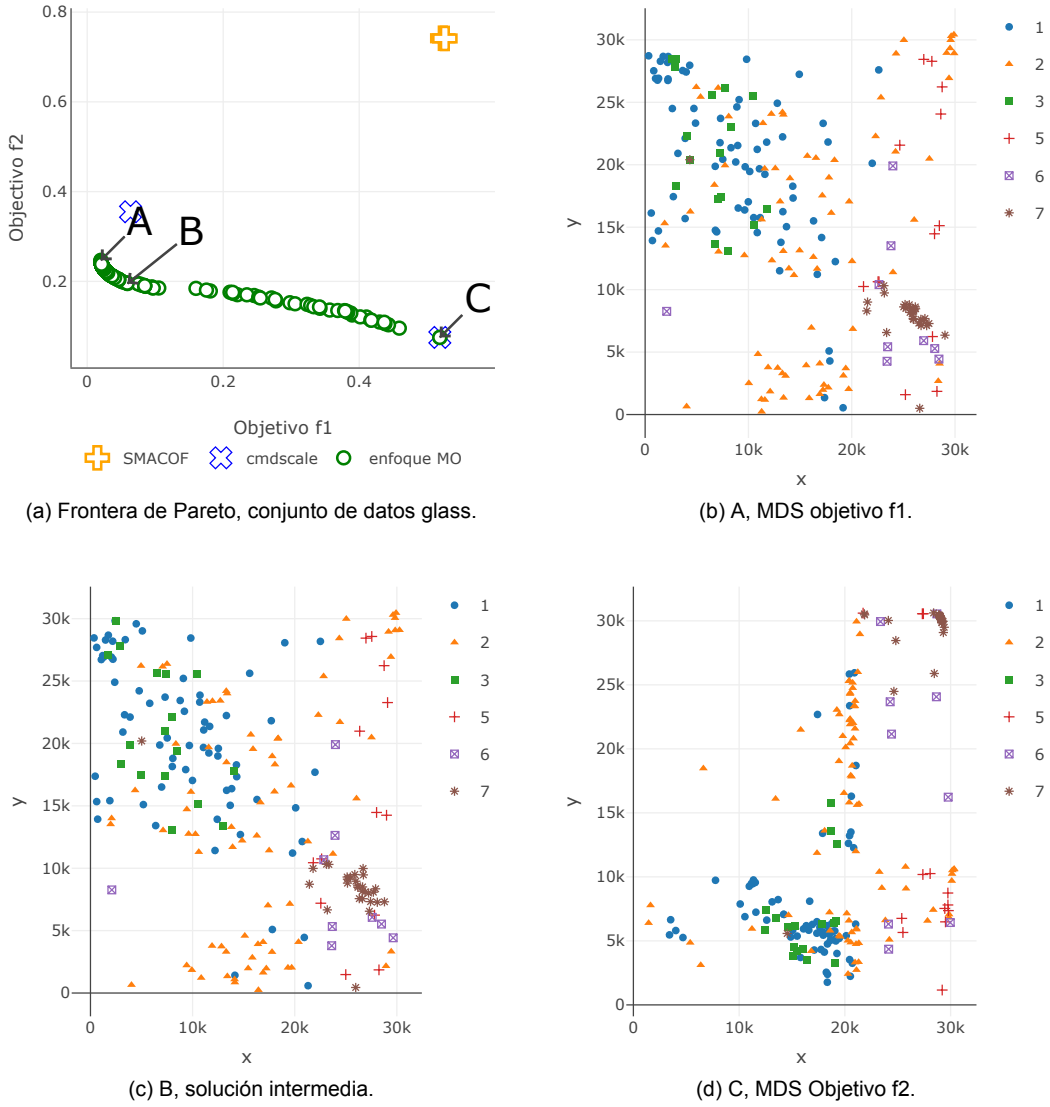
Conjunto de datos	Mejor f1		Solución intermedia		Mejor f2	
	f1	f2	f1	f2	f1	**f2**
cmdscale						
glass	**0.064**	0.355	-	-	0.519	**0.075**
iris	**0.002**	0.171	-	-	0.461	**0.003**
diabetes	**0.009**	0.131	-	-	0.138	**0.019**
bcw	**0.051**	0.454	-	-	1.000	**0.026**
ionosphere	**0.205**	0.757	-	-	1.000	**0.181**
fluTrees	**0.243**	0.985	-	-	1.000	**0.652**
hospitals	**0.051**	0.927	-	-	1.000	**0.000**
SMACOF						
glass	**0.523**	0.742	-	-	0.526	**0.741**
iris	**0.449**	0.646	-	-	0.449	**0.645**
diabetes	**0.989**	0.992	-	-	0.989	**0.992**
bcw	**0.833**	0.933	-	-	0.833	**0.933**
ionosphere	**0.610**	0.908	-	-	0.610	**0.908**
fluTrees	**0.486**	0.991	-	-	0.486	**0.991**
hospitals	**1.000**	0.000	-	-	1.000	**0.000**
Enfoque MO						
glass	**0.020**	0.243	0.052	0.206	0.519	**0.075**
iris	**0.002**	0.168	0.031	0.134	0.461	**0.003**
diabetes	**0.008**	0.106	0.040	0.069	0.138	**0.019**
bcw	**0.042**	0.364	0.063	0.334	1.000	**0.026**
ionosphere	**0.085**	0.604	0.153	0.556	1.000	**0.181**
fluTrees	**0.000**	0.971	0.171	0.832	1.000	**0.652**
hospitals	**0.025**	1.000	0.287	0.265	1.000	**0.000**

frontera de Pareto obtenidas con el enfoque MO (puntos verdes) y las soluciones obtenidas con los enfoques SO: SMACOF representadas como cruces amarillas y cmdscale como equis azules. Las anotaciones A, B y C corresponden a:

- Solución A: es la solución con mejor $f1$. La representación MDS que correspondiente a esta solución se presenta en la Figura 5.1b. Se pueden observar las instancias del conjunto de datos *glass* representadas en el espacio cartesiano separadas por su similitud en términos de distancia euclidiana entre sus características.

- Solución B: corresponde a una solución intermedia, seleccionada con la métrica L_2 (ver Sección 4.4). En la Figura 5.1c es posible ver las instancias del conjunto de datos *glass* representadas en el espacio cartesiano separadas por su similitud en términos de distancia euclidiana y *city-block*.

- Solución C: es la solución con mejor $f2$. La representación MDS correspondiente a esta solución se presenta en la Figura 5.1b. Se pueden observar las instancias del conjunto de datos *glass* representadas en el espacio cartesiano separadas por su similitud en términos de distancia *city-block*.



(a) Frontera de Pareto, conjunto de datos glass.

(b) A, MDS objetivo f1.

(c) B, solución intermedia.

(d) C, MDS Objetivo f2.

Figura 5.1: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *glass* $\hat{d}_{1_{ij}}$: distancia euclideana, $\hat{d}_{2_{ij}}$: distancia *city-block*. Color y número representa el tipo de vidrio, Fuente: Elaboración propia (2019).

De las soluciones A, B y C se puede desprender que:

- A, B y C no son iguales.
- A y B son similares. Esto ocurre porque las soluciones A y B minimizan en mayor medida el objetivo $f1$.
- El tipo de vidrio 7 en general está agrupado en todas las soluciones. En A y B en el cuadrante inferior derecho y en C en el cuadrante superior derecho.
- El tipo de vidrio 6 se mantiene cercano al tipo de vidrio 7 en todas las soluciones.
- Los tipos de vidrio 1 y 3 se mantienen juntos en A y B en el sector superior izquierdo y en C en el sector inferior izquierdo.

Con respecto al objetivo $f1$ las soluciones encontradas por el enfoque MO no son dominadas y son mejores en términos de $f1$ que las soluciones obtenidas por el enfoque SO.

En relación al objetivo $f2$ la solución C con mejor $f2$ es obtenida por el método cmdscale de la estrategia SO y por el enfoque MO. Al ser dos soluciones idénticas una no domina a la otra.

Para ambos objetivos $f1$ y $f2$, la estrategia MO encontró múltiples soluciones pertenecientes a la frontera de Pareto. Dentro de estas encontró las soluciones extremas (A y B, Figuras 5.1b-5.1d) y soluciones que representan el *tradeoff* entre las matrices de entrada (C, Figura 5.1c).

Finalmente, la estrategia MO propuesta, además de no ser dominada por las soluciones de las estrategias SO, es capaz de encontrar soluciones intermedias que minimizan simultáneamente los objetivos $f1$ y $f2$.

Los resultados de los demás conjuntos de datos están disponibles como anexos:

- Iris: Anexo A.
- Breast cancer Wisconsin: Anexo B.
- Diabetes: Anexo C.
- Ionosphere: Anexo D.
- FluTrees: Anexo E.
- Hospitales 2014: Anexo F.

CAPÍTULO 6. CONCLUSIONES Y TRABAJO FUTURO

6.1 CONCLUSIONES

En este trabajo se propuso una estrategia MO para resolver el problema de MDS multi-objetivo, considerando dos matrices de similitud o distancia como entrada.

En una primera etapa se buscaron conjuntos de datos clásicos y bien conocidos que han sido utilizados en problemas de aprendizaje de máquina y conjuntos de datos de la literatura en los cuales se utilizan dos métricas de similitud por separado (ver Sección 4.1.3). Estos conjuntos de datos se utilizaron para comparar el rendimiento de la estrategia MO y la estrategia SO en múltiples ejecuciones. Cumpliendo así con el primer objetivo específico planteado en la Sección 1.4.

Se compararon múltiples funciones de *stress* presentes en la literatura en términos de correlación. Esto permitió definir una función de *fitness* para utilizar en la etapa de selección del algoritmo genético propuesto (Sección 3.5), en general las funciones de *stress* están correlacionadas entre sí (Sección 2.4). Finalmente, se seleccionó la métrica de *stress* normalizado por no depender de la escala de las entradas, cumpliendo así el segundo objetivo específico (Sección 1.4).

Para abordar el problema de MDSMO con dos objetivos, se construyó un algoritmo genético evolutivo basado en NSGA-II (Sección 3) considerando operaciones de inicialización, mutación, cruzamiento y selección. Cumpliendo el tercer objetivo específico (Sección 1.4). El enfoque MO fue comparado en términos de contribución de hipervolumen con algoritmos SO utilizando distintos y bien conocidos conjuntos de datos de la literatura relacionada. Los resultados mostraron que el enfoque MO obtiene soluciones que no son dominadas por las estrategias SO y que, además, encuentra en algunos casos mejores soluciones con respecto a uno de los objetivos. Cumpliendo con el cuarto y el quinto objetivo específico propuesto en la Sección 1.4.

Finalmente, se puede dar respuesta a la pregunta planteada al inicio de este trabajo (Sección 1.2) ¿Es posible resolver el problema de escalamiento multidimensional con dos métricas de similitud utilizando un enfoque multi-objetivo?. La respuesta es sí, la estrategia MO propuesta fue capaz de obtener como soluciones representaciones MDS con menor o igual *stress* normalizado que las representaciones obtenidas con métodos mono-objetivo de la literatura con respecto a un objetivo, además de encontrar múltiples representaciones MDS diferentes entre sí que combinan dos medidas de similitud, al mismo tiempo para todos los conjuntos de datos utilizados.

Estos resultados hacen que la propuesta sea una buena alternativa para estudiar

dos matrices de similitud distintas, cuando la relación entre ellas es desconocida. Sin embargo, la mayoría de problemas en la vida real, por ejemplo, el estudio de juicios de similitud en psicología, incluye muestras de más de tres personas, esto significa que el problema de MDS podría ser tratado como un problema de optimización de múltiples objetivos, mediante la aplicación de estrategias de computación evolutiva.

6.2 TRABAJO FUTURO

El estudio de estrategias de optimización multi-objetivo y computación evolutiva aplicadas al problema de escalamiento dimensional han dejado en evidencia tópicos para estudios posteriores, entre ellos:

- **MDS muchos-objetivos.** En este trabajo se abordó el problema de MDSMO con dos objetivos, quedando abierto el problema MDSMO con tres o más objetivos, que podría ser abordado con estrategias de optimización de muchos objetivos como NSGA-III (Jain y Deb, 2014).
- **MDS multi-modal.** La estrategia MO propuesta en este trabajo busca soluciones distintas en relación a los valores de los objetivos utilizando distancia de hacinamiento y otras métricas, empleando exclusivamente el espacio objetivo. Esto no asegura la diversidad de las soluciones según su disposición sobre el espacio de decisiones. La optimización multi-modal busca encontrar todos los mínimos o máximos locales y globales en una sola ejecución, incorpora distintas técnicas para asegurar que las soluciones no sean similares entre sí, distancia de hacinamiento (*crowding distance*, *fitness sharing*, conservación de especies, *covariance matrix adaptation*, entre otras (Wong, 2015). Este enfoque podría encontrar un conjunto de soluciones más diversas para el problema de MDSMO.
- **Diferencias individuales multi-objetivo.** Actualmente, la aplicación de MDS para caracterizar diferencias individuales entre juicios de similitud emitidos por distintas personas, en una primera etapa define una representación MDS de consenso, rotando y ponderando las representaciones MDS individuales, para luego comparar el MDS individual con el MDS de consenso. Esta primera etapa se podría reemplazar con una estrategia multi-objetivo que integre las representaciones MDS de todos los individuos en una representación MDS de consenso.

LISTADO DE ACRÓNIMOS

HV Hiper-Volumen. 18, 19, 30–32

MDS del inglés, *MultiDimensional Scaling*. 1, 3–6, 8, 9, 12, 13, 25, 30, 36

MDSMO MDS Multi-Objetivo. 13, 20, 36, 37

MO Multi-Objetivo. 18, 30–33, 35–37

MOO del inglés, *Multi-Objective Optimisation*. 14

NSGA-II del inglés, *Non-dominated Sorting Genetic Algorithm II*. 4, 15, 20, 36

SMACOF del inglés, *Scaling by MAjorizing a COmplicated Function*. 9, 13, 30, 32, 33

SO del inglés, *Single-Objective*. 18, 31–33, 35, 36

REFERENCIAS BIBLIOGRÁFICAS

Bai, S., Bai, X., Latecki, L. J. y Tian, Q. (2017). Multidimensional scaling on multiple input distance matrices. En *Thirty-first AAAI conference on artificial intelligence.*

Birattari, M., Stützle, T., Paquete, L. y Varrentrapp, K. (2002). A racing algorithm for configuring metaheuristics. En *Proceedings of the 4th annual conference on genetic and evolutionary computation* (pp. 11–18). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Borg, I. y Groenen, P. J. (2005). *Modern multidimensional scaling: Theory and applications*. 233 Spring Street, New York, NY 10013, USA: Springer Science & Business Media.

Bossek, J. (2017). Ecr 2.0: A Modular Framework for Evolutionary Computation in R. En *Proceedings of the genetic and evolutionary computation conference (gecco) companion* (pp. 1187–1193). Berlin, Germany: ACM. Recuperado de `http://doi.acm.org/10.1145/3067695.3082470` Doi: 10.1145/3067695.3082470

Buja, A., Swayne, D. F., Littman, M. L., Dean, N., Hofmann, H. y Chen, L. (2008). Data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics*, *17*(2), 444–472.

Carroll, J. D. y Chang, J.-J. (1970). Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. *Psychometrika*, *35*(3), 283–319. Doi: 10.1007/bf02310791

Choi, H. H. (2012). Dynamic learning for visual representation of asymmetric proximity. En *Conference proceedings - IEEE international conference on systems, man and cybernetics* (p. 1972-1977). Doi: 10.1109/ICSMC.2012.6378027

Cleeland, C. S., Nakamura, Y., Mendoza, T. R., Edwards, K. R., Douglas, J. y Serlin, R. C. (1996). Dimensions of the impact of cancer pain in a four country sample: new information from multidimensional scaling. *Pain*, *67*(2), 267–273. Doi: 10.1016/0304-3959(96)03131-4

Cox, M. A. A. y Cox, T. F. (2008). Multidimensional scaling. En *Handbook of data visualization* (pp. 315–347). Berlin, Heidelberg: Springer Berlin Heidelberg. Doi: 10.1007/978-3-540-33037-0_14

De Leeuw, J., Barra, I. J. R., Brodeau, F., Romier, G. y Van Cutsem, B. (1977). Applications of convex analysis to multidimensional scaling. En *Recent developments in statistics: proceedings of the european meeting of statisticians* (pp. 133–146). Grenoble: North-Holland Pub. Co. : distributors for the U.S.A. and Canada, Elsevier/North-Holland.

De Leeuw, J. y Heiser, W. J. (1977). Convergence of correction matrix algorithms for multidimensional scaling. En *Geometric representations of relational data: Readings in multidimensional scaling* (pp. 735–752). Ann Arbor, Mathesis Press.

De Leeuw, J. y Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. *Multivariate analysis*, *5*(1), 501–522.

De Leeuw, J. y Mair, P. (2009). Multidimensional scaling using majorization: SMACOF in R. *Journal of Statistical Software*, *31*(3), 1–30. Recuperado de `http://www.jstatsoft.org/v31/i03/`

Deb, K., Pratap, A., Agarwal, S. y Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182-197. Doi: 10.1109/4235.996017

Ding, C. (2018). *Fundamentals of applied multidimensional scaling for educational and psychological research*. Springer. Doi: 10.1007/978-3-319-78172-3

Dua, D. y Graff, C. (2017). *UCI machine learning repository.* Recuperado el 2019-05-18, de `http://archive.ics.uci.edu/ml`

Dzidolikaite, A. (2015). Genetic algorithms for multidimensional scaling. *Mokslas: Lietuvos Ateitis*, *7*(3), 275–279. Doi: 10.3846/mla.2015.781

Dzwinel, W. (1994). How to make sammon's mapping useful for multidimensional data structures analysis. *Pattern Recognition*, *27*(7), 949 - 959. Doi: https://doi.org/10.1016/0031-3203(94)90160-0

Dzwinel, W. (1997). Virtual particles and search for global minimum. *Future Generation Computer Systems*, *12*(5), 371 - 389. (HPCN96) Doi: https://doi.org/10.1016/S0167-739X(96)00024-6

Foundation, T. R. (2019). *R: What is R?* Recuperado el 2019-05-18, de https://www.r-project.org/about.html

Gandour, J. T. y Harshman, R. A. (1978). Crosslanguage differences in tone perception: a multidimensional scaling investigation. *Language and Speech*, *21*(1), 1–33. Doi: 10.1177/002383097802100101

Gartner, W. C. (1989). Tourism image: Attribute measurement of state tourism products using multidimensional scaling techniques. *Journal of Travel Research*, *28*(2), 16–20. Doi: 10.1177/004728758902800205

Giglio, J., Villalobos-Cid, M. e Inostroza-Ponta, M. (2019). A multi-objective optimisation evolutionary approach for the multidimensional scaling problem. En *Proceedings - International Conference of the Chilean Computer Science Society, SCCC.*

Git. (2019). *Git.* Recuperado el 2019-05-18, de https://git-scm.com/

Gower, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, *53*(3-4), 325–338.

Groenen, P. y Borg, I. (2013). The past, present, and future of multidimensional scaling. *Econometric Institute Research Papers*, *EI 2013-07*.

Handl, J., Kell, D. B. y Knowles, J. (2007). Multiobjective optimization in bioinformatics and computational biology. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, *4*(2), 279–292.

Horan, C. B. (1969). Multidimensional scaling: Combining observations when individuals have different perceptual structures. *Psychometrika*, *34*(2), 139–165. Doi: 10.1007/bf02289341

Ishibuchi, H., Nojima, Y. y Doi, T. (2006). Comparison between single-objective and multi-objective genetic algorithms: Performance comparison and performance measures. En *IEEE Int Conf on Evol Comp (CEC)* (Vol. 1, p. 1143-1150). Vancouver, BC, Canada.

Jain, H. y Deb, K. (2014). An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, *18*(4), 602-622. Doi: 10.1109/TEVC.2013.2281534

Jombart, T., Kendall, M., Almagro-Garcia, J. y Colijn, C. (2017). Treespace: statistical exploration of landscapes of phylogenetic trees. *Molecular Ecology Resources*, *17*(6), 1385-1392. Recuperado de https://onlinelibrary.wiley.com/doi/abs/10.1111/1755-0998.12676 Doi: 10.1111/1755-0998.12676

Kruskal, J. B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, *29*(1), 1–27.

LaTeX. (2019). *LaTeX - A document preparation system.* Recuperado el 2019-05-18, de https://www.latex-project.org/

Lessa, E. P. (1990). Multidimensional analysis of geographic genetic structure. *Systematic Zoology*, *39*(3), 242. Doi: 10.2307/2992184

López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M. y Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, *3*, 43 - 58. Recuperado de http://www.sciencedirect.com/science/article/pii/S2214716015300270 Doi: https://doi.org/10.1016/j.orp.2016.09.002

Machado, J. T., Duarte, F. B. y Duarte, G. M. (2011). Analysis of stock market indices through multidimensional scaling. *Communications in Nonlinear Science and Numerical Simulation*, *16*(12), 4610 - 4618. (SI:Complex Systems and Chaos with Fractionality, Discontinuity, and Nonlinearity) Doi: https://doi.org/10.1016/j.cnsns.2011.04.027

Mardia, K. V. (1978). Some properties of clasical multi-dimesional scaling. *Communications in Statistics-Theory and Methods*, *7*(13), 1233–1241.

Mead, A. (1992). Review of the development of multidimensional scaling methods. *Journal of the Royal Statistical Society: Series D (The Statistician)*, *41*(1), 27–39.

MySQL. (2019). *MySQL :: About MySQL.* Recuperado el 2019-05-18, de https://www.mysql.com/about/

Overleaf. (2019). *About us - Overleaf, Online LaTeX Editor.* Recuperado el 2019-05-18, de https://www.overleaf.com/about

Pawliczek, P., Dzwinel, W. y Yuen, D. (2014). Visual exploration of data by using multidimensional scaling on multicore cpu, gpu, and mpi cluster. *Concurrency and Computation Practice and Experience*, 1-21. Doi: 10.1002/cpe.3027

R Core Team. (2018). R: A language and environment for statistical computing [Manual de software informático]. Vienna, Austria. Recuperado de https://www.R-project.org/

Riquelme, N., Von Lücken, C. y Baran, B. (2015). Performance metrics in multi-objective optimization. En *2015 Latin American Computing Conference (CLEI)* (Vol. 1, p. 1-11).

Roux, I. (2008). *Application of cluster analysis and multidimensional scaling on medical schemes data* (Tesis Doctoral no publicada). Stellenbosch: Stellenbosch University, South Africa.

Roweis, S. T. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*(5500), 2323–2326. Doi: 10.1126/science.290.5500.2323

Schiffman, S., Musante, G. y Conger, J. (1978). Application of multidimensional scaling to ratings of foods for obese and normal weight individuals. *Physiology & Behavior*, *21*, 417-422. Doi: 10.1016/0031-9384(78)90102-6

Schwartz, S. H. (1994). Are there universal aspects in the structure and contents of human values? *Journal of Social Issues*, *50*(4), 19–45. Doi: 10.1111/j.1540-4560.1994.tb01196.x

SCOPUS. (2019). *Scopus - analyze search results.* Recuperado el 2019-05-15, de https://www-scopus-com.ezproxy.usach.cl/term/analyzer.uri?sid=3aee018397e7dae280b3f2b8efca7754&origin=resultslist&src=s&s=TITLE-ABS-KEY%28%22multidimensional+scaling%22%29&sort=cp-f&sdt=b&sot=b&sl=41&count=11596&analyzeResults=Analyze+results&txGid=ac4a6bcba5b0771e1452e39ebbfe2984

Scott, J. (1988). Social network analysis. *Sociology*, *22*(1), 109–127. Doi: 10.1177/0038038588022001007

Shepard, R. N. (1962a). The analysis of proximities: Multidimensional scaling with an unknown distance function I. *Psychometrika*, *27*(2), 125–140. Doi: 10.1007/bf02289630

Shepard, R. N. (1962b). The analysis of proximities: Multidimensional scaling with an unknown distance function II. *Psychometrika*, *27*(3), 219–246. Doi: 10.1007/bf02289621

Simplemaps. (2019). *World Cities Database | Simplemaps.com.* Recuperado el 2019-05-18, de https://simplemaps.com/data/world-cities

Suchard, M. A., Lemey, P., Baele, G., Ayres, D. L., Drummond, A. J. y Rambaut, A. (2018). Bayesian phylogenetic and phylodynamic data integration using BEAST 1.10. *Virus Evolution*, *4*(1). Recuperado de https://doi.org/10.1093/ve/vey016 Doi: 10.1093/ve/vey016

Tecuanhuehue-Vera, P., Carrasco-Ochoa, J. A. y Martínez-Trinidad, J. F. (2012). Genetic algorithm for multidimensional scaling over mixed and incomplete data. En *Mexican conference on pattern recognition* (pp. 226–235).

Tenenbaum, J. B., De Silva, V. y Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, *290*(5500), 2319–2323. Doi: 10.1126/science.290.5500.2319

Torgerson, W. S. (1952). Multidimensional scaling: I. theory and method. *Psychometrika*, *17*(4), 401–419.

Vermeesch, P. (2019). Exploratory analysis of provenance data using r and the provenance package. *Minerals*, *9*(3), art. no. 193.

Villalobos-Cid, M., Dorn, M. e Inostroza-Ponta, M. (2018a). Performance comparison of multi-objective local search strategies to infer phylogenetic trees. En *2018 IEEE Congress on Evolutionary Computation (CEC)* (p. 1-8).

Villalobos-Cid, M., Dorn, M. e Inostroza-Ponta, M. (2018b). Understanding the relationship between decision and objective space in the multi-objective phylogenetic inference problem. En *2018 IEEE congress on evolutionary computation (cec)* (p. 1-8). Doi: 10.1109/CEC.2018.8477689

Villalobos-Cid, M., Dorn, M., Ligabue-Braun, R. e Inostroza-Ponta, M. (2019). A memetic algorithm based on an NSGA-II scheme for phylogenetic tree inference. *IEEE Transactions on Evolutionary Computation*, *23*(5), 776–787.

Villalobos-Cid, M., Vega-Araya, D. e Inostroza-Ponta, M. (2017). Application of different multi-objective decision making techniques in the phylogenetic inference problem. En *2017 36th international conference of the chilean computer science society (SCCC)* (pp. 1–9). Arica, Chile.

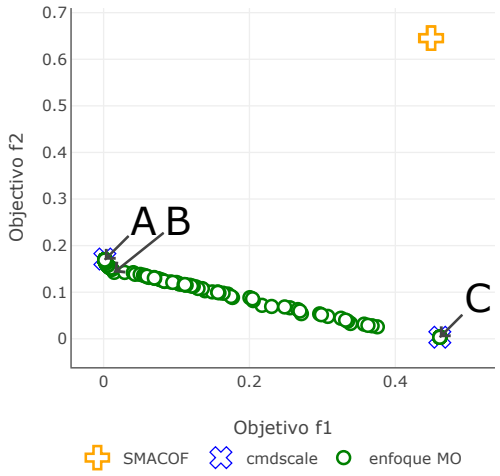
Villalobos-Cid, M., Chacón, M., Zitko, P. e Inostroza-Ponta, M. (2016). A new strategy to evaluate technical efficiency in hospitals using homogeneous groups of casemix. *Journal of Medical Systems*, *40*(4), art. no. 103. Doi: 10.1007/s10916-016-0458-9

Wong, K.-C. (2015). Evolutionary multimodal optimization: A short survey. *arXiv preprint arXiv:1508.00457*.

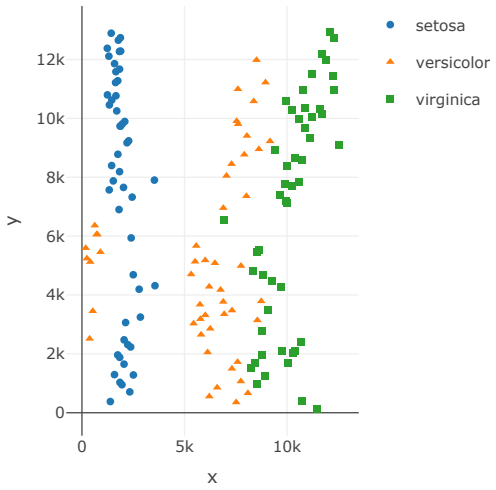
Young, G. y Householder, A. S. (1938). Discussion of a set of points in terms of their mutual distances. *Psychometrika*, *3*(1), 19–22. Doi: 10.1007/BF02287916

Anexos

A. RESULTADOS PARA EL CONJUNTO DE DATOS IRIS

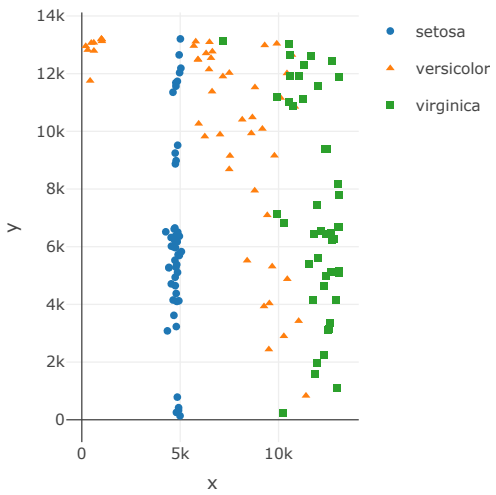


(a) Frontera de Pareto.



(b) A, MDS objetivo f1.

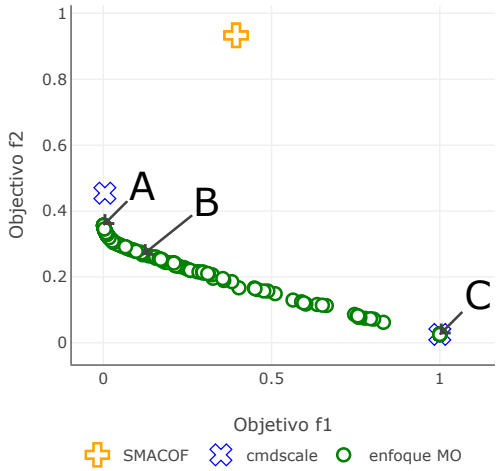
(c) B, solución intermedia



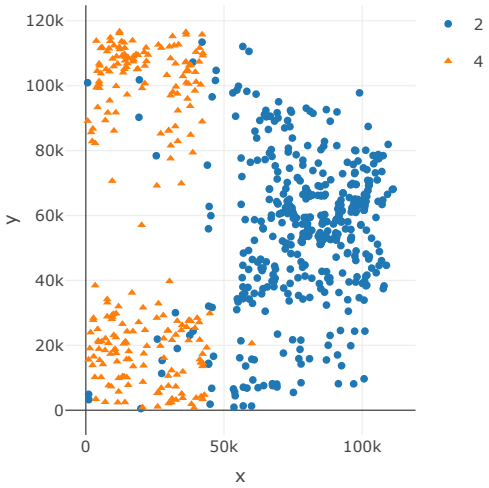
(d) C, MDS objetivo f2.

Figura 1: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *iris* $\hat{d}_{1ij}$: distancia euclideana, $\hat{d}_{2ij}$: distancia *city-block*. Color representa el tipo de planta. Fuente: Elaboración propia (2019).

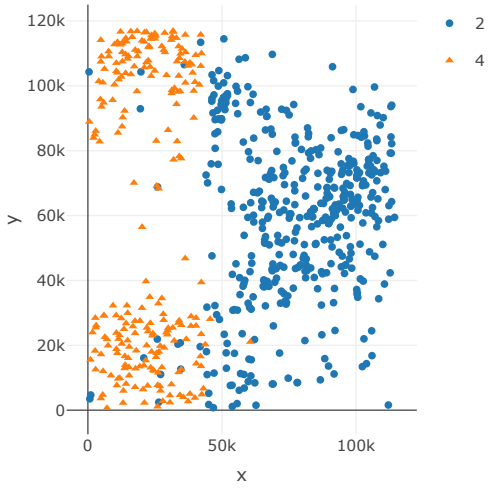
B. RESULTADOS PARA EL CONJUNTO DE DATOS BREAST CANCER WISCONSIN



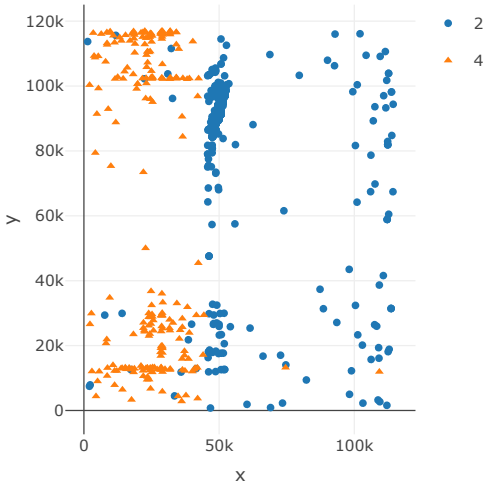
(a) Frontera de Pareto.



(b) A, MDS objetivo f1.



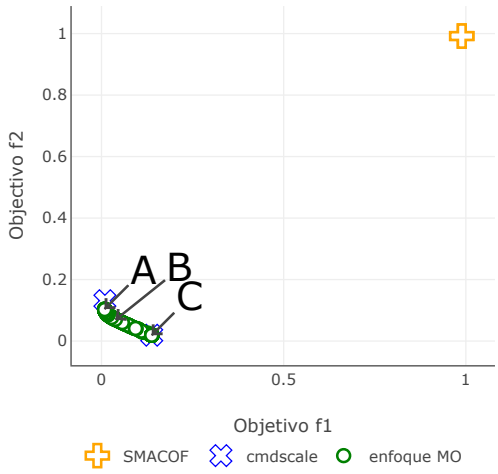
(c) B, solución intermedia.



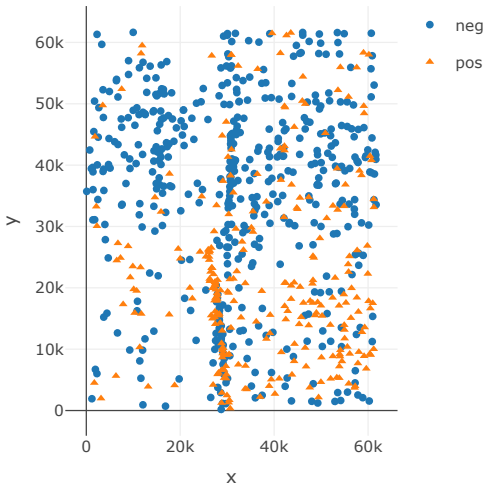
(d) C, MDS objetivo f1.

Figura 2: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *bcw* $\hat{d_1}_{ij}$: distancia euclideana, $\hat{d_2}_{ij}$: distancia *city-block*. 2 es tumor benigno y 4 es tumor maligno. Fuente: Elaboración propia (2019).

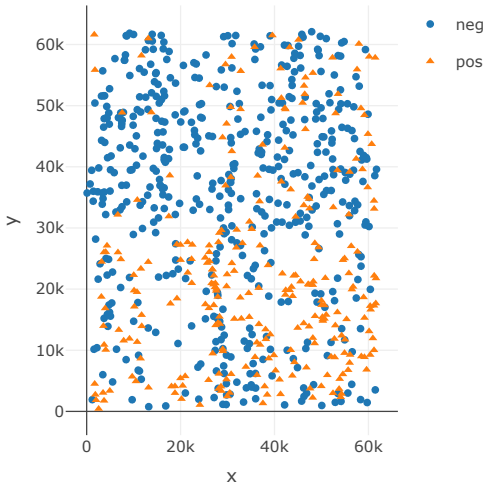
C. RESULTADOS PARA EL CONJUNTO DE DATOS DIABETES



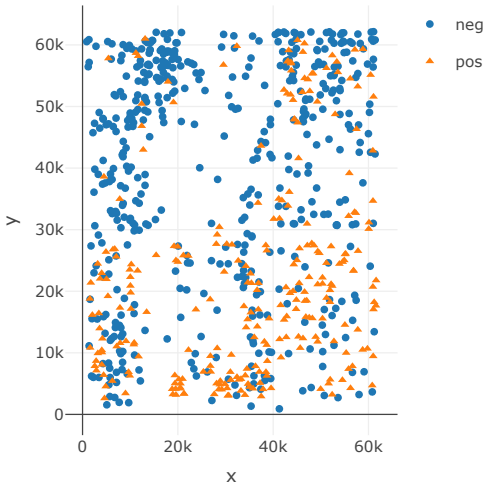
(a) Frontera de Pareto.



(b) A, MDS objetivo f1.



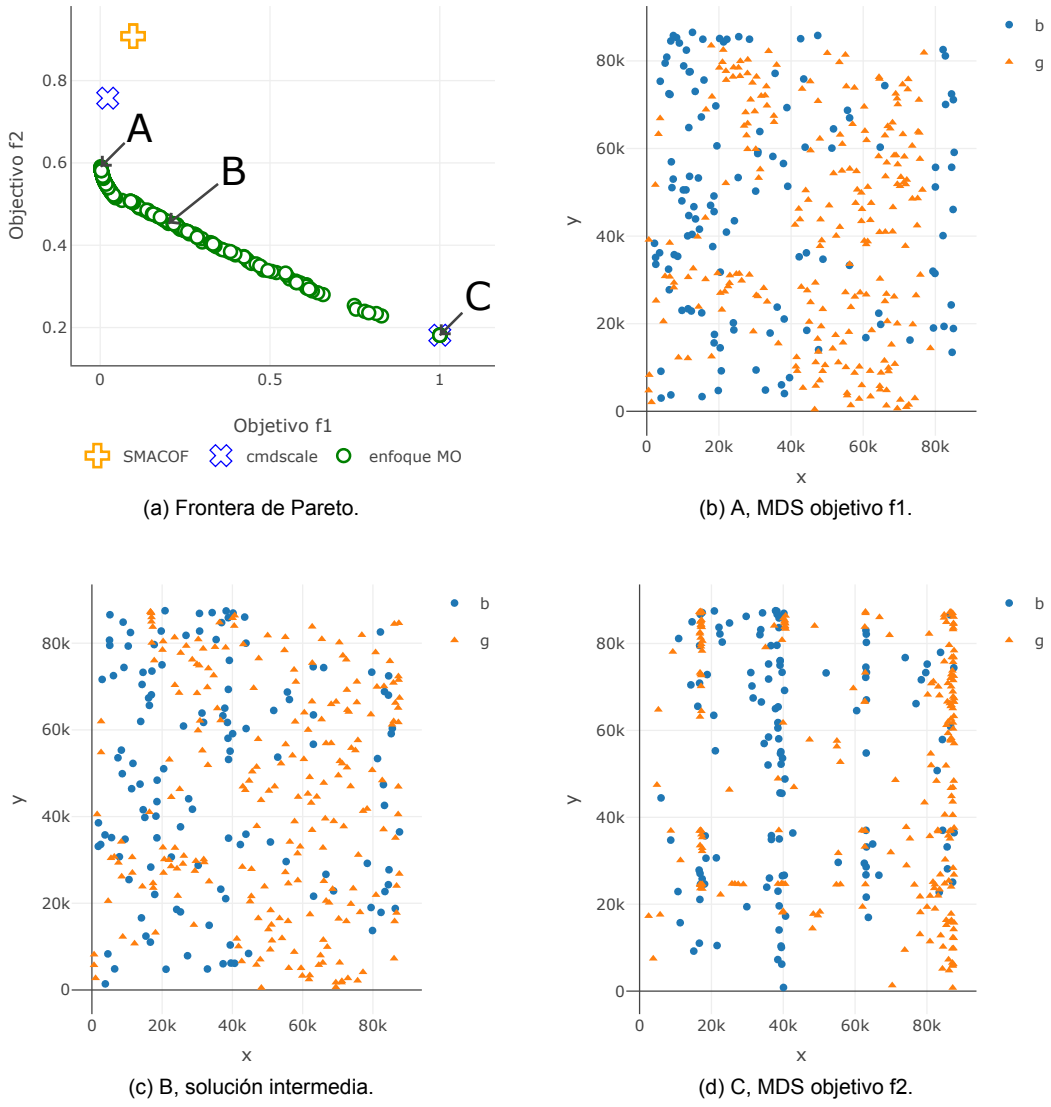
(c) B, solución intermedia.



(d) C, MDS objetivo f2.

Figura 3: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *diabetes* $\hat{d}_{1ij}$: distancia euclideana, $\hat{d}_{2ij}$: distancia *city-block*. *pos* tiene diabetes y *neg* no tiene diabetes. Fuente: Elaboración propia (2019).

D. RESULTADOS PARA EL CONJUNTO DE DATOS IONOSPHERE



(a) Frontera de Pareto.

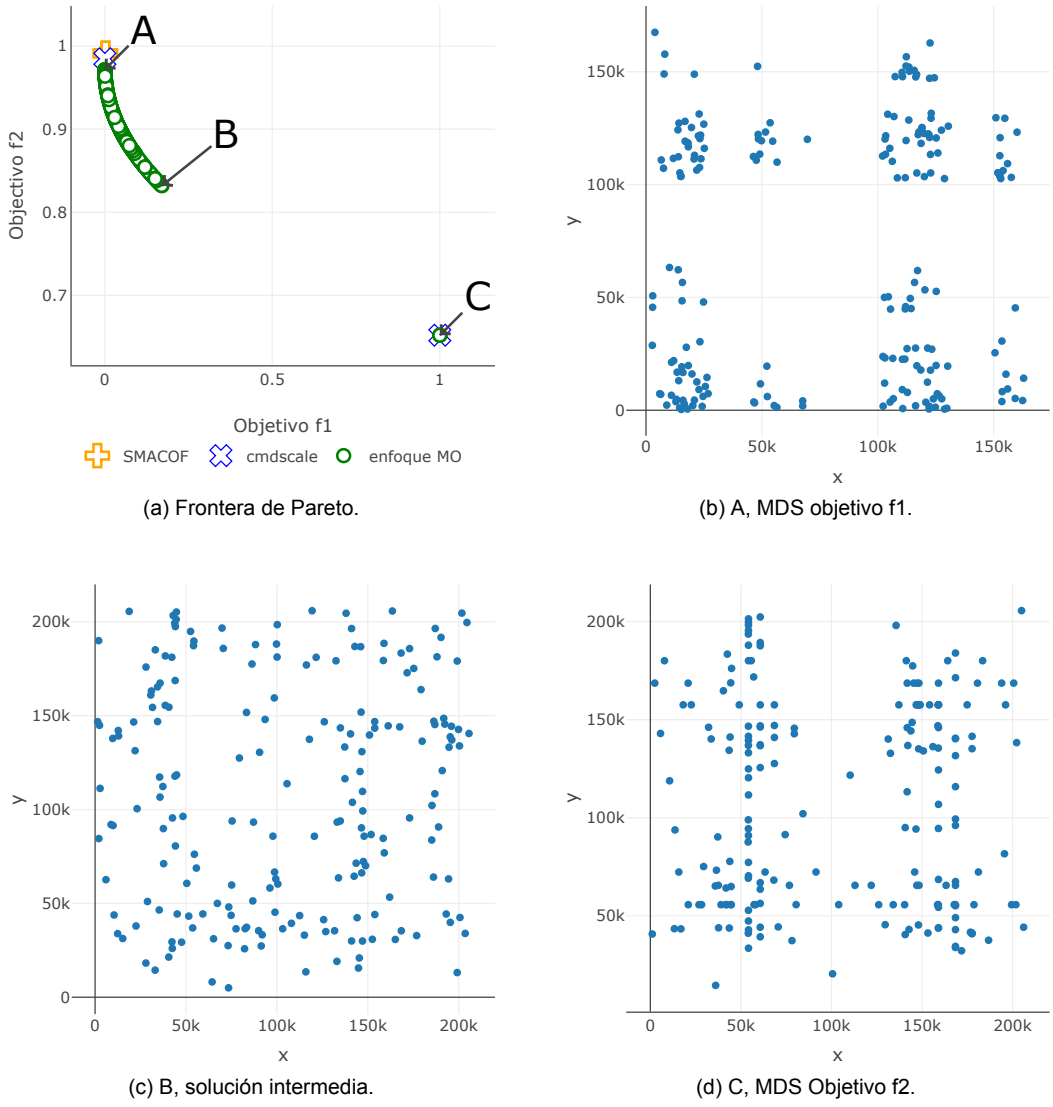
(b) A, MDS objetivo f1.

(c) B, solución intermedia.

(d) C, MDS objetivo f2.

Figura 4: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *ionosphere* $\hat{d}_{1ij}$: distancia euclideana, $\hat{d}_{2ij}$: distancia *city-block*. *b* malo y *g* bueno. Fuente: Elaboración propia (2019).

E. RESULTADOS PARA EL CONJUNTO DE DATOS FLUTREES



(a) Frontera de Pareto.

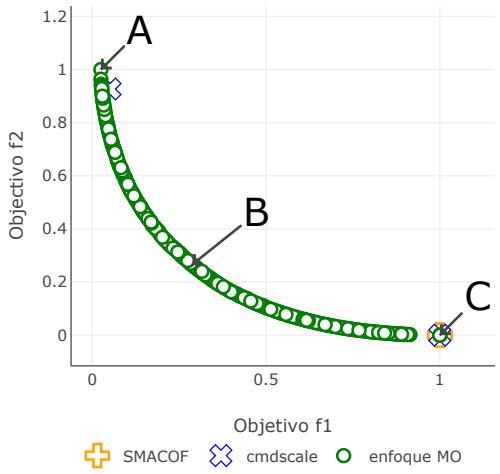
(b) A, MDS objetivo f1.

(c) B, solución intermedia.

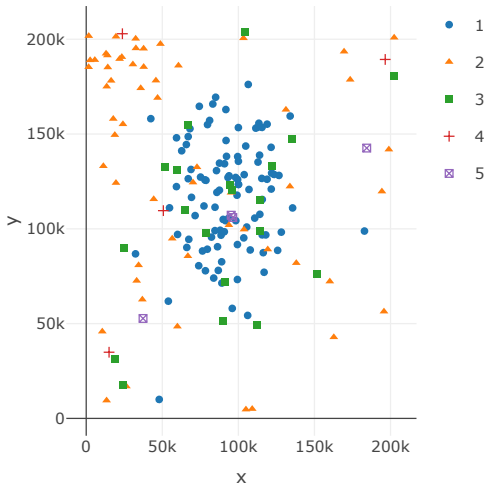
(d) C, MDS Objetivo f2.

Figura 5: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *fluTrees* $\hat{d}_{1ij}$: distancia euclideana, $\hat{d}_{2ij}$: distancia *city-block*. Fuente: Elaboración propia (2019).

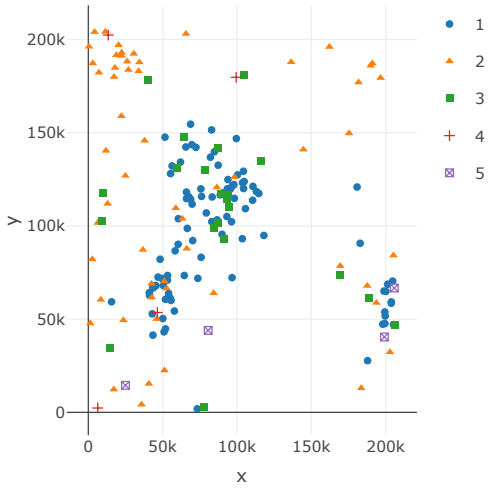
F. RESULTADOS PARA EL CONJUNTO DE DATOS HOSPITALES2014



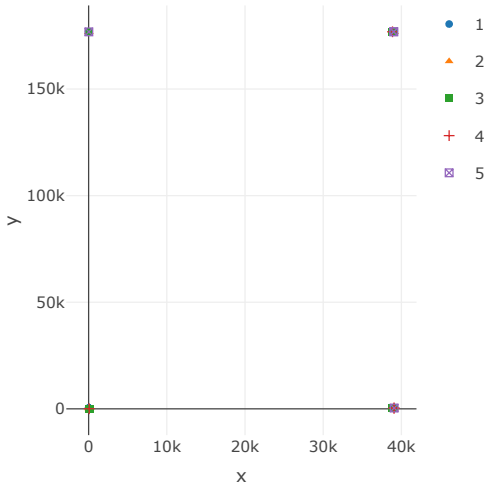
(a) Frontera de Pareto.



(b) A, MDS objetivo f1.



(c) B, solución intermedia.



(d) C, MDS objetivo f2.

Figura 6: Representaciones MDS obtenidas utilizando enfoque MO y SO para el conjunto de datos *hospitales2014* $\hat{d}_{1ij}$: distancia euclideana, $\hat{d}_{2ij}$: distancia de correlación. 1-5 clasificación de hospital. Fuente: Elaboración propia (2019).

G. PORTADA ARTÍCULO JORNADAS CHILENAS DE LA COMPUTACIÓN 2019

A multi-objective optimisation evolutionary approach for the Multidimensional Scaling Problem

Juan Giglio*, Mario Inostroza-Ponta*, Manuel Villalobos-Cid*†
*Departamento de Ingeniería Informática
Universidad de Santiago de Chile, Santiago, Chile
{*juan.giglio,mario.inostroza,manuel.villalobos*}*@usach.cl*
† Corresponding author

***Abstract*—The Multidimensional Scaling (MDS) strategies allow visualising the similarity between different objects reducing the number of dimensions. MDS has been widely used to perform exploratory analyses in different fields of the knowledge. The current strategies designed to deal with the MDS problem are able to consider exclusively one measure in a same time, however, most of the real-life problems usually require to analyse more than one measure simultaneously. The multi-objective optimisation techniques have been successfully used to deal with in problems from different areas considering multiples criteria (two or three criteria). In this work, we propose a genetic algorithm to deal with the multi-objective MDS problem being evaluated by using classical data sets from the related literature. The results show that the proposed strategy is able to identify a Pareto set of solutions that include new representations which were non-dominated by solutions from the current state of the art single-objective optimisation approaches, and new solutions which combine the features of the different inputs. These results make our proposal a real alternative to deal with problems which require to visualise different similarity inputs.**

***Index Terms*—Multidimensional scaling problem, evolutionary algorithm, multi-objective optimisation, data visualisation.**

I. Introduction

Multidimensional scaling (MDS) is a set of methods designed to represent similarity (or dissimilarity) between objects in a low-dimensional space allowing their visualisation [1, 2].

The MDS was designed to deal with problems from the **psychology** area, attempting to discover hidden attributes by considering human similarity judgements between objects, which conform an abstract and diffuse space called "*psychological space*". MDS transforms this mental representation into an Euclidean space [3]. Also, the MDS has been used in **sociology**, to research social networks [4] and to understand the relationships between different cultures [5]; in **linguistics**, to determine the attributes associated to the perception of the linguistic tone, understanding the association between the language background influences and the human perception [6]. In **biology**, the MDS has been used to analyse the genetic geographic structure [7], to compare different evolutionary hypotheses [8, 9], and to demonstrate how the cultural and linguistic background affect the relationships among pain, mood, and sleep [10]. The MDS also has contributed in other areas, such as **tourism** and **marketing** [11], **medicine** [12, 13] and **education** [14] facilitating the visualisation and exploration of similarity between multidimensional objects.

There are two ways to treat the MDS problem: (1) the **metric scaling** considers that the objects are related to distances according to a continuous function and (2) the **non-metric or ordinal scaling** assumes that the configuration of points in the space is associated to similarities [2]. Multiple measures to represent multi-dimensional objects on an Euclidean space have been proposed in the literature for the non-metric MDS, allowing the use of similarity instead of distance metrics: Stress, raw Stress, normalised Stress [15], S-Stress [16], among others.

In general, the MDS methods uses iterative optimisation over an initial distribution of points in a low-dimensional space, moving the points until a minimum stress function will be minimised [17].

The current methods that perform MDS are able to analyse one similarity metric at time (single optimisation). However, most problems may contain several similarity measures conflicting with each other. For example, Machado [18] studied separately time and histogram's distance between stock market shares, Choi [19] created two conflicting MDS representations to compare asymmetrical similarity between countries, and Villalobos-Cid [9] compared two different phylogenetic distances to study phylogenetic trees. If the similarity metrics are analysed separately, information about the combination and interaction between metrics could be discarded, adding bias to the resultant visualisation.

Bai [20] deals with this problem weighting the conflicting inputs, however, the scale of the inputs are usually unrelated and can not be merged, or the weights are not always known.

Models based on **multi-objective optimisation** (MOO) have demonstrated advantages compared to the single-optimisation methods: minimisation of the probability of stagnation in local minimum and areas without gradients, reduction of noise effect from the data, and incorporation of **multiple sources** which are in conflict with each other [21]. Hence, a MOO strategy could be used to merge different input in the MDS context.

In this work, we propose a MO novel strategy to deal with the Non-metric MDS. It is based on the Non-dominated sorting