El presente libro se propone estimar el rango apropiado en la factorización matricial no negativa NMF, aplicada a la restauración de imágenes de colposcopía. Para ello, se presentan los preliminares que contiene el conjunto de definiciones y resultados necesarios para sustentar los métodos MAD, SURE y MDL. Se describen los tres métodos citados; se ilustran con ejemplos que permiten analizarlos; y se exponen los pseudocódigos de los algoritmos de su implementación en Matlab. Finalmente, se desarrolla la experimentación y el análisis de resultados. Los aportes fundamentales que se ponen a consideración son el Software en Matlab para la detección de los pixeles de brillo de una colposcopía; el estudio y la reproducción de métodos de la literatura sobre la determinación del rango en NMF, incluido software en Matlab de implementación; y la aplicación de los métodos en la restauración de imágenes de colposcopía.

Haro

Lascano

Coronel

Paladines

Villón

9 780311 000722

Estudio de la determinación del rango en las Factorizaciones de Matrices No Negativas y su aplicación en la Restauración de Imágenes

ETECAM

Estudio de la determinación del rango en las Factorizaciones de matrices no negativas y su aplicación en la restauración de imágenes

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay

Book Citation Index
Web of Sciece/Core Collection

Estudio de la determinación del rango en las Factorizaciones de Matrices No Negativas y su aplicación en la Restauración de Imágenes
Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Estudio de la determinación del rango en las factorizaciones de matrices no negativas y su aplicación en la restauración de imágenes

Study of rank determination in factorizations of nonnegative matrixes and its application in image restoring

Diseño: Ing. Erik Marino Santos Pérez.
Traducción: Prof. Dr. C. Ernan Santiesteban Naranjo.
Corrección de estilo: Prof. Dra. C. Leydis Iglesias Triana.
Diagramación: Prof. Dr. C. Ernan Santiesteban Naranjo.
Director de Colección Ciencias naturales & matemáticas: Prof. Dr. C. Carlos Manuel Caraballo Carmona.
Jefe de edición: Prof. Dra. C. Kenia María Velázquez Avila.
Dirección general: Prof. Dr. C. Ernan Santiesteban Naranjo.

Sobre la presente edición:
Primera edición
Esta obra ha sido evaluada por pares académicos a doble ciegos
Lectores/Pares académicos/Revisores: 0065 & 0100
Editorial Tecnocientífica Americana
Domicilio legal: calle 613sw 15th, en Amarillo, Texas. **ZIP**: 79104, EEUU
Teléfono: 7867769991
Fecha de publicación: 23 septiembre de 2024
Código BIC: PBW **Código EAN: 9780311000722**
Código UPC: 978031100072 **ISBN: 978-0-3110-0072-2**
La Editorial Tecnocientífica Americana se encuentra indizada en, referenciada en o tiene convenios con, entre otras, las siguientes bases de datos:

Book Citation Index
Web of Sciece/Core Collection

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Contenido

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Recepción: 15-06-2024
Aprobación: 01-09-2024

Acerca de los autores

Socrates Emilio Haro Guanga

Doctor en Ciencias de la Educación. Doctor en Matemática, con especialización en Modelización Matemática. Máster en Ciencias Matemáticas, con mención en Matemática Numérica, por la Universidad de La Habana. Máster en Nuevas Tecnologías Aplicadas a la Educación, por la Universidad Autónoma de Barcelona. Actualmente, se desempeña como docente e investigador a tiempo completo en la Universidad de Guayaquil.

Edisson Lascano Mora

Magíster en Matemática, con mención en Modelación y Docencia. Magíster en Pedagogía de las Ciencias Experimentales, mención Matemática y Física. Ingeniero en Electricidad. Actualmente, se desempeña como docente de la Universidad de Guayaquil, en la que ejerce la cátedra a tiempo completo. Es, además, profesor invitado de la maestría en Pedagogía de las Ciencias Experimentales, de la Universidad Técnica de Manabí.

Verónica Coronel Pérez

Máster en Business Administration del Endicott College, Massachussetts y Glion, en Suiza. Economista, con mención en Gestión Empresarial de la Escuela Superior Politécnica del Litoral (ESPOL). Se desempeñó como Talent Coach en W Hotels, en Qatar y Trump Organization, en Miami. Ha sido docente de Matemáticas Aplicada, en la Universidad de Guayaquil.

Leonardo Paladines Zurita

Máster en Matemáticas, especializado en Modelación Matemática. Ingeniero Mecánico. Se ha desempeñado como docente en bachillerato y en la Escuela Superior Politécnica del Litoral (ESPOL), en la cual ha liderado laboratorios de física. Su experiencia también abarca la

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

enseñanza en institutos de educación superior, en áreas de razonamiento abstracto, matemáticas y física.

Víctor Hugo Villón Pincay

Máster en Modelación Matemática. Licenciado en Educación, en la especialidad Física y Matemática. Fue profesor en nivelación, en la Universidad de Guayaquil, y miembro del grupo de capacitadores para el proyecto SIPROFE del Ministerio de Educación en Didáctica de la Matemática. Actualmente, se desempeña como docente de Termofluidos, Electromagnetismo y Ecuaciones Diferenciales.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
Recepción: 15-06-2024
Aprobación: 01-09-2024

Resumen

El presente libro se propone estimar el rango apropiado en la factorización matricial no negativa NMF, aplicada a la restauración de imágenes de colposcopía. Para ello, se presentan los preliminares que contiene el conjunto de definiciones y resultados necesarios para sustentar los métodos MAD, SURE y MDL. Se describen los tres métodos citados; se ilustran con ejemplos que permiten analizarlos; y se exponen los pseudocódigos de los algoritmos de su implementación en Matlab. Finalmente, se desarrolla la experimentación y el análisis de resultados. Los aportes fundamentales que se ponen a consideración son el Software en Matlab para la detección de los pixeles de brillo de una colposcopía; el estudio y la reproducción de métodos de la literatura sobre la determinación del rango en NMF, incluido software en Matlab de implementación; y la aplicación de los métodos en la restauración de imágenes de colposcopía.

Palabras clave: colposcopía, MAD, SURE y MDL

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Abstract

This book aims to estimate the appropriate rank in the non-negative matrix factorization NMF, applied to the restoration of colposcopy images. For this purpose, the preliminaries containing the set of definitions and results necessary to support the MAD, SURE and MDL methods are presented. The three methods are described; they are illustrated with examples that allow to analyze them; and the pseudocodes of the algorithms of their implementation in Matlab are exposed. Finally, experimentation and analysis of results are developed. The main contributions to be considered are the Matlab software for the detection of the brightness pixels of a colposcopy; the study and reproduction of methods from the literature on the determination of the range in NMF, including implementation software in Matlab; and the application of the methods in the restoration of colposcopy images.

Key words: colposcopy, MAD, SURE and MDL

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Capítulo 1. Introducción la factorización matricial no negativa NMF

Uno de los motivos de este libro es mostrar las líneas de investigación del Grupo de Procesamiento de Imágenes del departamento de Matemática Aplicada, de la Universidad de La Habana en colaboración con el Ministerio de Salud Pública de Cuba. Dichas investigaciones, entre otras, han sido orientadas al estudio de imágenes de colposcopía, con el objetivo de desarrollar herramientas y algoritmos basados en el procesamiento de imágenes para apoyar a los médicos en la detección temprana del cáncer cérvico-uterino.

El presente libro tiene como objetivo general estimar el rango apropiado en la factorización matricial no negativa NMF, aplicada a la restauración de imágenes de colposcopía. Para ello se propusieron diferentes acciones: la identificación y eliminación de zonas de brillo en imágenes de colposcopía, siguiendo los resultados previos; el estudio e implementación, en Matlab, de los métodos de selección del rango, elegidos de la literatura; experimentar con métodos de selección del rango de la literatura para decidir cuál es el mejor en la restauración de imágenes de colposcopía; comparar los métodos elegidos y estimar cuál proporciona el rango más adecuado en un grupo de imágenes de colposcopía; y medir la calidad de las imágenes (mediante SSIM) obtenidas con el rango mínimo estimado (con los tres métodos), ya sea, tanto a las imágenes obtenidas con el producto de factores WH de NMF, como a las imágenes restauradas mediante NMF.

Las acciones anteriores conducen al cumplimiento del objetivo general y a la comprobación empírica de la hipótesis que plantea la posibilidad de determinar cuál de los tres métodos estudiados es el más idóneo para la selección del rango interno de una matriz, y con este lograr

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

una adecuada restauración de imágenes de colposcopía, utilizando la factorización matricial no-negativa.

El aporte de este libro se centra en cuatro dimensiones fundamentales: un Software en Matlab para la detección de los pixeles de brillo de una colposcopía; presentar una sistematización de métodos de la literatura sobre la determinación del rango en NMF, incluido el software en Matlab de implementación; determinar las conclusiones sobre la experimentación de los métodos; y ejemplificar la aplicación de los métodos en la restauración de imágenes de colposcopía.

Desde la aparición de los computadores, el manejo de grandes cantidades de datos ya no es una novedad, la importancia de este manejo radica en la necesidad de encontrar una aplicación óptima en diferentes áreas. Esta cantidad enorme de información o datos se los puede ordenar, organizar y codificar en tablas o matrices. Un ejemplo de aplicación en la actualidad son las imágenes digitales, sean estas a color o en blanco y negro.

En el caso de una imagen en blanco y negro, se puede modelar mediante una matriz con m filas y n columnas $V=\left(v^{ij};i=1,...,m;j=1,...,n\right);V\in\mathbb{R}_{+}^{m\times n}$, donde el valor de cada elemento v^{ij} de la matriz representa la tonalidad en escala de grises del pixel ij de la imagen. La escala de grises, está representada dentro de un rango de $0\le v_{ij}\le 255$ donde el menor valor $v_{ij}=0$ le corresponde al pixel negro y el máximo $v^{ij}=255$ al pixel blanco, y los demás valores intermedios corresponden a la graduación de la escala de grises (Gómez, 2018). Por ejemplo, observe la imagen de colposcopía y una parte de los valores de su matriz $A_{480x720}$ correspondiente en escala de grises.

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 1. Colposcopía de la base de datos de la Universidad de La Habana (izquierda) y su equivalente en escala de grises (derecha)

Fuente: Elaboración propia, mediante una función de Matlab

Figura 2. Matriz de códigos numéricos en escala de grises de una colposcopía

$$v=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 5 & 5 & 7 & 7 & 9 & 9 & 74 & 208 & 254 & 254 & \cdots \\ 5 & 5 & 6 & 6 & 6 & 6 & 83 & 214 & 255 & 255 & \cdots \\ 5 & 5 & 5 & 5 & 5 & 5 & 69 & 208 & 255 & 255 & \cdots \\ 5 & 5 & 5 & 5 & 5 & 5 & 88 & 200 & 255 & 255 & \cdots \\ 5 & 5 & 5 & 5 & 5 & 5 & 73 & 202 & 255 & 255 & \cdots \\ 5 & 5 & 5 & 5 & 5 & 5 & 83 & 210 & 255 & 255 & \cdots \\ 5 & 5 & 5 & 5 & 5 & 5 & 77 & 214 & 255 & 255 & \cdots \\ 5 & 5 & 5 & 5 & 5 & 5 & 81 & 202 & 255 & 255 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \end{bmatrix}_{480\times720}$$

Fuente: Elaboración propia, a partir de un código de Matlab

Las imágenes de colposcopía se obtienen de la observación del cérvix (parte interior del útero) con la ayuda de un colposcopio que consta de un potente lente y una luz acoplados a una computadora (Gómez, 2018; Palmer, 2015). Para el efecto, se han desarrollado dos trabajos de

Recepción: 15-06-2024
Aprobación: 01-09-2024

diploma, en los que se aplicaron diferentes técnicas de eliminación del brillo y luego de restauración de un grupo de imágenes de colposcopías, usando como método base la factorización matricial no negativa (NMF) definido por Lee & Seung (1999), tema que será explicado más adelante.

Desde la publicación del primer algoritmo de factorización matricial no negativa NMF (Lee & Seung, 1999) ha ido creciendo el interés por él. Uno de sus problemas más importantes es la selección del rango (Ulfarsson & Solo, 2013). Este fue uno de los temas pendientes o sugeridos en los trabajos de diploma de Palmer (2015) y Gómez (2018).

Considere una imagen modelada con $V \in \mathbb{R}_{+}^{m \times n}$, donde $v^{ij} \geq 0$ son no negativos, por la naturaleza de sus datos. NMF es una técnica de factorización que reduce la dimensionalidad lineal y proporciona una representación de los datos basada en partes (Muzzarelli et al., 2019); la diferencia de NMF con otras técnicas como PCA (*Principal Component Analysis*) o SVD (*Singular Value Descomposition*), radica en que, NMF requiere que sus factores W y H sean no negativos; mientras que en PCA y SVD exige que sus factores sean ortogonales. NMF permite aproximar la matriz de datos originales $V^{m \times n}$ como $V^{m \times n} \approx W^{m \times k} H^{k \times n}$; donde generalmente la dimensión k se reduce, siendo $k < \min(m, n)$.

Las columnas de la matriz W representan los vectores base del nuevo subespacio y las columnas de la matriz H representan los coeficientes de cada uno de los puntos en ese nuevo subespacio de proyección. Las columnas de W representan las características de la imagen V y las columnas de H son los pesos que distinguen unas características de otras. La combinación lineal o producto de

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

estas dos matrices representan las características aproximadas a las características reales de la imagen original V (Lee & Seung, 1999; Squires et al., 2017).

Un problema importante en NMF es la selección del rango adecuado para la factorización (Ulfarsson & Solo, 2013). En las colposcopías de la figura 1 de arriba, para detectar y delimitar las regiones brillantes (o especulares) se han utilizado diferentes algoritmos como se muestran en los trabajos hechos por Palmer (2015) y Gómez (2018). Ellos, una vez que definieron las denominadas regiones brillantes, estudiaron el problema de eliminarlas y reconstruir la imagen incompleta que representa una aproximación de la imagen original. Si la imagen incompleta es modelada mediante una matriz de valores conocidos y valores incógnitos, entonces, el problema nuestro es, ¿cómo hallar el rango k adecuado de factorización que, a su vez, reproduzca los valores adecuados, que reemplacen a las incógnitas?, esto se conoce como problema de determinación del rango de la matriz V. En este estudio, los valores incógnitos representan los pixeles de las zonas brillosas que han sido eliminados a propósito y reemplazadas por valores $[X]=0$.

$$\begin{bmatrix} [X] & 102 & 23 & 99 & 250 & 120 \\ [X] & [X] & 30 & 23 & 50 & 39 \\ [X] & [X] & 167 & 87 & 37 & 45 \\ [X] & 97 & 205 & [X] & [X] & [X] \\ 245 & 234 & 200 & [X] & [X] & [X] \\ 178 & 204 & 190 & 209 & [X] & [X] \end{bmatrix} \approx \begin{bmatrix} [100] & 102 & 23 & 99 & 250 & 120 \\ [108] & [203] & 30 & 23 & 50 & 39 \\ [123] & [234] & 167 & 87 & 37 & 45 \\ [63] & 97 & 205 & [67] & [105] & [204] \\ 245 & 234 & 200 & [98] & [49] & [25] \\ 178 & 204 & 190 & 209 & [120] & [12] \end{bmatrix}$$

En la revisión de la literatura, se encuentran varios estudios respecto al problema de la determinación del rango. A continuación, se muestran algunos de ellos que serán abordados con mayor profundidad en próximos capítulos.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
Recepción: 15-06-2024
Aprobación: 01-09-2024

Un estudio realizado por Guillis (2014) propone tres métodos basados en (i) la opinión del experto, (ii) prueba y error, (iii) la descomposición de valores singulares. En el primer caso, es posible que no haya expertos calificados para elegir un rango k adecuado; por otro lado, los expertos también suelen equivocarse; y por último si el experto logra seleccionar un rango adecuado, su decisión no deja de ser subjetiva. En el segundo caso, la prueba y error cae en la selección manual del rango de parte del investigador interesado en adaptar su modelo y estudio de manera subjetiva. En el tercer caso, el uso de la descomposición de valores singulares sería un desafío si los valores singulares no presentan un claro decrecimiento hacia cero, la elección de k cuando los valores singulares se vuelven muy pequeños no es fácil de hacer (Squires et al., 2017).

Un estudio reciente, sobre la determinación del rango de factorización, plantean una técnica basada en la compresión lineal de la longitud de un mensaje, la llaman longitud mínima de descripción MDL (*Minimum Description Length*, de sus siglas en inglés). Ellos aplican esta técnica a datos reales y a datos sintéticos. Para los datos reales MDL proporciona una selección de un valor plausible del rango k, y para los datos sintéticos con rango conocido k MDL estima razonablemente los valores de k (Squires et al., 2017). Esta técnica será estudiada y explicada posteriormente.

En un estudio realizado para enfrentar el problema del análisis inteligente de una base de datos de *Twitter* (red social que permite a los usuarios compartir mensajes cortos del tipo SMS denominados tweets), se reconoce que la selección del rango de factorización es de suma importancia, pues determina el número de agrupaciones y los temas ocultos en los millones de tweets. Al ser el rango un parámetro de entrada, ellos fijan de manera manual el rango de factorización en k=4, entendiéndose este parámetro como el número de palabras clave usadas

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

para filtrar los tweets. De esta manera, logran comparar los temas ocultos, que han sido extraídos automáticamente por los algoritmos NMF. La red social Twitter produce alrededor de 500 millones de tweets diarios, es evidente que la capacidad humana de análisis se ve rebasada, por esto plantean una técnica de análisis y clasificación mediante el uso de NMF (Casalino et al., 2017).

En un estudio reciente se propone una nueva métrica para la determinación del rango, basado en la validación cruzada de imputación (Kanagal & Sindhwani, s/a.). Sus experimentos los realizan utilizando conjuntos de datos sintéticos con diferentes propiedades y también con datos reales. Verificaron que, los resultados dependen de las propiedades de los datos. A pesar de que su técnica (MADimput) les acercó una mejor precisión frente a otras técnicas, ellos manifiestan que se debe evaluar con cuidado las características del conjunto de datos antes de decidir e identificar qué método es el más adecuado para la selección del rango (Muzzarelli et al., 2019).

En el estudio realizado por Ulfarsson & Solo (2013) se propone un método basado en un estimador de riesgo no sesgado de Stein denominado SURE (*Stein's Unbiased Risk Estimator*) (Stein, 1981). Esta técnica consiste en desarrollar un estimador computable no sesgado del error cuadrático medio (MSE) que se pueda usar para estimar el rango (Ulfarsson & Solo, 2013).

Luego de revisar la literatura, se han seleccionado tres métodos para reproducirlos y compararlos, estos son: la técnica denominada MAD desarrollado por Muzzarelli et al. (2019); el método llamado SURE aportado por Ulfarsson & Solo (2013); y el método denominado MDL desarrollado por Squires et al. (2017).

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Capítulo 2. Definiciones preliminares

En este capítulo se presentan tanto las definiciones y notaciones necesarias para entender el contenido de este trabajo, como las notaciones debidas para matrices y los algoritmos utilizados.

Matrices

Se propone usar la notación $V=\left[v^{ij};i=1,...,m;j=1,...,n\right]$ para representar una matriz con m filas y n columnas donde $v^{ij}\in\square_{+}$(números reales no negativos). Si la dimensión de la matriz es obvia por el contexto, se escribirá simplemente $V=\left[v^{ij}\right]$. Al conjunto de todas las matrices con m filas y n columnas se denotará por $\mathcal{M}^{m\times n}$. La matriz $V=\left[v^{ij}\right]$se suele visualizar de la siguiente manera.

$$V=\begin{bmatrix} v_{11} & \cdot & \cdot & \cdot & v_{1n} \\ \cdot & \cdot & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ v_{m1} & \cdot & \cdot & \cdot & v_{mn} \end{bmatrix}$$

La matriz transpuesta de una matriz $V=\left[v^{ij}\right]$se define como $V^{T}=\left[v^{ji}\right]$, se obtiene yuxtaponiendo filas por columnas, esto es:

$$V^{T}=\begin{bmatrix} v_{11} & \cdot & \cdot & \cdot & v_{m1} \\ \cdot & \cdot & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ v_{1n} & \cdot & \cdot & \cdot & v_{mn} \end{bmatrix}$$

Si $m=n$, se dice que la matriz$V$ es cuadrada, y el conjunto de todas las matrices cuadrada se denotará por $\mathcal{M}^{n}$.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
Recepción: 15-06-2024
Aprobación: 01-09-2024

Los vectores $v = \begin{bmatrix} v_1 \\ . \\ . \\ . \\ v_n \end{bmatrix} \in \mathbb{R}^n$ serán considerados en este trabajo como matrices con una sola columna; es decir $v \in \mathcal{M}^{n\times 1}$.

La $i-ésima$ fila de la matriz $V \in \mathcal{M}^{m\times n}$ estará representada por el vector $v_{i:}$, y la $j-ésima$ columna se representará por el vector $v_{:j}$; es decir

$$v_{i:} = \begin{bmatrix} v_{i1} & . & . & . & v_{in} \end{bmatrix} \quad \text{y} \quad v_{:j} = \begin{bmatrix} v_{1j} \\ . \\ . \\ . \\ v_{mj} \end{bmatrix}$$

La matriz nula de dimensión $m\times n$ está formada por todos sus elementos ceros

$$O = \begin{bmatrix} 0 & . & . & . & 0 \\ . & . & & & . \\ . & & . & & . \\ . & & & . & . \\ 0 & . & . & . & 0 \end{bmatrix}$$

La matriz identidad de dimensión $n\times n$ es una matriz diagonal con de valores unos

$$I_n = \begin{bmatrix} 1 & . & ... & . & 0 \\ . & 1 & & & . \\ . & & ... & & . \\ . & & & 1 & . \\ 0 & . & ... & . & 1 \end{bmatrix}$$

Imagen. - Todas las imágenes utilizadas serán en escala de grises. Mediante la función de Matlab rgb2gray(imagen) se transforman las imágenes a color, en imágenes a escala de grises.

Pixel. - El pixel es la unidad básica de una imagen.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Colposcopio. - aparato con tubo y cámara conectado a una computadora para tomar fotos internas del útero.

Colposcopía. - foto digitalizada de la parte interior del útero.

Zonas acetoblancas. - zonas de la colposcopía con tejidos pálidos menos brillosos.

Zonas especulares. - zonas de la colposcopía con tejidos brillosos producido por el flash de la cámara del colposcopio al tomar la foto en una zona húmeda al interior del útero.

La descomposición matricial SVD (Singular Value Descomposition)

La SVD es una forma de factorización matricial donde cada matriz $A=\left[a^{ij};i=1,...,m;\ j=1,...,n\right]$ puede ser descompuesta de la forma siguiente:

$$A_{m\times n}=U_{m\times k}\sum_{k\times k}V^{T}_{k\times n} \qquad Ec.(1.1.1)$$

Donde:

- U tiene una dimensión $m\times k$ y $U^{T}U=I$,
- $\sum$ tiene una dimensión $k\times k$ es una matriz diagonal de números reales no negativos $\sigma_1\geq\sigma_2\geq\cdots\geq\sigma_n\geq 0$,
- V tiene una dimensión $k\times n$ y $V^{T}V=I$.

Cómo se hace esto, se calcula el problema más pequeño. ¿Cuál es más pequeño m o n?

$$\begin{aligned}A^{T}A&=V\sum^{2}V^{T}\ si\ \text{n}<m\\ AA^{T}&=U\sum^{2}U^{T}\ si\ m<n\end{aligned} \qquad Ec.(1.1.2)$$

Los vectores de V son los vectores propios de $A^{T}A$ y los vectores de U son los vectores propios de AA^{T}; los valores singulares al cuadrado $\sum^{2}$ son los son los valores propios de $A^{T}A$ y también de AA^{T}.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Observe que:

$$A^T A = \left(U \sum V^T\right)^T U \sum V^T = V \sum^T U^T U \sum V^T = V \sum^T \sum V^T \qquad Ec.(1.1.3)$$

Así que tenemos una descomposición de la matriz simétrica real $A^T A$ como una matriz V por una matriz diagonal (tiene entradas σ_i^2) por la transpuesta de V. Por lo tanto, por la descomposición SVD tenemos los siguientes hechos.

- Los valores singulares σ_i de A son las raíces cuadradas de los valores propios de $A^T A$.
- Las columnas de V son los vectores singulares derechos de A y los vectores propios de $A^T A$.
- Al formar $A^T A$, veríamos que las columnas de U, son los vectores singulares izquierdos de A, son los vectores propios de $A^T A$.

En Matlab se calcula el SVD con la función $[U, S, V] = svd(A)$.

Definición elemental de probabilidad (Laplace). - Sea N el número de veces que se observa un evento o ensayo de un experimento aleatorio, y sea n el número de veces que ocurre un resultado favorable al evento A. La probabilidad del evento A, es la frecuencia relativa observada cuando el número total de observaciones crece indefinidamente (Bianco, s. f.):

$$p(A) = \lim_{n \to \infty} \frac{n}{N} \qquad Ec.(1.1.4)$$

La definición anterior no es correcta desde el punto de vista matemático, debido a que la probabilidad depende de la convergencia o no, del límite (Scheaffer & McClave, 1993).

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Definición formal de probabilidad.- Suponga que un experimento está asociado a un espacio muestral S Una probabilidad es una función $p:S\to$ que asigna un valor numérico $p(A)$ a cada evento A, tal que se cumplen los siguientes axiomas (Scheaffer & McClave, 1993):

(i) $p(A)\geq 0$

(ii) $p(S)=1$

(iii)Si $A_1, A_2, ...,$ es una sucesión de eventos mutuamente excluyentes, es decir, $A_i \cap A_j = \phi$, $\forall$i $\neq$ j, entonces

$$p\left(\bigcup_{i=1}^{\infty} A_i\right)=\sum_{i=1}^{\infty} p(A_i) \qquad Ec.(1.1.5)$$

Distribución Gamma

Es una distribución de probabilidad adecuada para modelar el comportamiento de variables aleatorias con asimetría positiva y/o experimentos que tengan que ver con el tiempo. Una variable aleatoria X tiene una distribución gamma si su función de densidad está dada por (Muñoz, 2014; Squires et al., 2017)):

$$f(x\mid\alpha,\beta)=\begin{cases}\dfrac{1}{\beta^{\alpha}\Gamma(\alpha)}x^{\alpha-1}e^{\frac{-x}{\beta}}, & para\ \ x>0;\alpha,\beta>0\\ 0, & para\ cualquier\ otro\ caso\end{cases} \qquad Ec.(1.1.6)$$

donde $\Gamma(\square)$ es la función Gamma $\Gamma:(0,\infty)\to$

$$\Gamma(\alpha)=\int_0^{\infty} t^{\alpha-1}e^{-t}dt \text{ para } \alpha>0 \qquad Ec.(1.1.7)$$

$$\Gamma(n)=(n-1)!, \forall n\in^{+} \qquad Ec.(1.1.8)$$

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Momentos absolutos y centrados

Los momentos de una variable aleatoria X son un promedio ponderado de una función de dicha variable $h(x)$, donde los ponderadores son los valores de su función de probabilidad, sea esta discreta o continua. El resultado se llama valor esperado de la función $h(x)$ (Bianco, s. f.):

$$E\left[h(x)\right]=\sum_{\forall x} h(x)\cdot p(x)$$

$$o \qquad \textit{Ec.}(1.1.9)$$

$$E\left[h(x)\right]=\int_{\forall x} h(x)\cdot f(x)\,dx$$

Si se define la función $h(x)=x^r$, entonces surge el denominado momento absoluto de orden r de la variable aleatoria (Bianco, s. f.):

$$E\left[X^r\right]=\sum_{\forall x} x^r\cdot p(x)$$

$$o \qquad \textit{Ec.}(1.1.10)$$

$$E\left[X^r\right]=\int_{\forall x} x^r\cdot f(x)\,dx$$

Esperanza matemática o Valor esperado de una variable aleatoria. - Es el momento absoluto de orden uno, y se lo denota por μ:

$$E(X)=\sum_{\forall x} x\cdot p(x)$$

$$o \qquad \textit{Ec.}(1.1.11)$$

$$E(X)=\int_{\forall x} x\cdot p(x)\,dx$$

El resultado es un promedio ponderado de los valores de la variable aleatoria, donde los ponderadores le otorgan mayor peso a los valores con mayor probabilidad, de ahí surge el nombre de valor esperado (Bianco, s. f.).

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
Recepción: 15-06-2024
Aprobación: 01-09-2024

Momento centrado de orden r

El momento centrado de orden r de una variable aleatoria resulta de definir $h(x)=\left[x-E(X)\right]^r$ en la *Ec.*(1.1.9) (Bianco, s. f.):

$$E\left\{\left[X-E(X)\right]^r\right\}=\sum_{\forall x}\left[X-E(X)\right]^r\cdot p(x)$$

o

$$E\left\{\left[X-E(X)\right]^r\right\}=\int_{\forall x}\left[X-E(X)\right]^r\cdot f(x)\,dx \qquad Ec.(1.1.12)$$

Varianza.- Es el momento centrado de orden 2 (Bianco, s. f.):

$$Var(X)=\sum_{\forall x}\left[X-E(X)\right]^2\cdot p(x)$$

o

$$Var(X)=\int_{\forall x}\left[X-E(X)\right]^2\cdot f(x)\,dx \qquad Ec.(1.1.13)$$

La Paradoja de Stein

La predicción óptima de un suceso futuro se elabora promediando los sucesos pasados. La paradoja de Stein determina las condiciones de existencia de estimadores que son mejores que la media aritmética. El primer paso para aplicar el método de Stein es calcular el promedio de los promedios, se lo denomina “gran promedio” que se lo denota por $\bar{y}$.

El proceso esencial del método de Stein consiste en contraer todos los promedios individuales hacia este gran promedio, este estimador se lo denota por z. La paradoja de Stein cosiste, simplemente, en los valores z, es decir los estimadores de James-Stein (en honor a Stein y James

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

que 1961 pusieron una versión sencilla del método) dan mejores estimaciones que los promedios individuales. El estimador de James-Stein se define así:

$$z = \bar{y} + c\left(y - \bar{y}\right) \qquad Ec.(1.1.14)$$

Donde el término $\left(y - \bar{y}\right)$ expresa la diferencia entre el promedio de cada individuo y el gran promedio, multiplicada por una constante $c < 1$ denominada "constante de constricción", se la calcula a partir de la colección de todos los promedios observados (Efron & Morris, 1977).

Teorema (Lema de Stein). Si $X \sim N\left(\mu, \sigma^2\right)$ y si g es una función diferenciable tal que $E\left[\left|g'\left(X\right)\right|\right] < \infty$, entonces

$$E\left[g\left(X\right)\left(X - \mu\right)\right] = \sigma^2 E\left[g'\left(X\right)\right] \qquad Ec.(1.1.15)$$ (Herazo, 2014).

Ejemplo 1. Si $X \sim N\left(\mu, \sigma^2\right)$, entonces el cuarto momento μ_4, es:

$$\mu_4 = E\left[\left(X - \mu\right)^4\right]$$
$$= E\left[\left(X - \mu\right)^3\left(X - \mu\right)\right]$$
$$= \sigma^2 E\left[3\left(X - \mu\right)^2\right]$$
$$= 3\sigma^2 E\left[\left(X - \mu\right)^2\right]$$
$$= 3\sigma^2 \cdot \sigma^2$$

$$= 3\sigma^4.$$

Recepción: 15-06-2024
Aprobación: 01-09-2024

2.1. Detección de los pixeles de brillo

Para la detección de los pixeles de brillo, existen algunos métodos y técnicas presentes en la literatura y que ya han sido desarrollados (Gómez, 2018; Palmer, 2015). En este trabajo se usará la técnica del umbral de intensidad, que se describe a continuación.

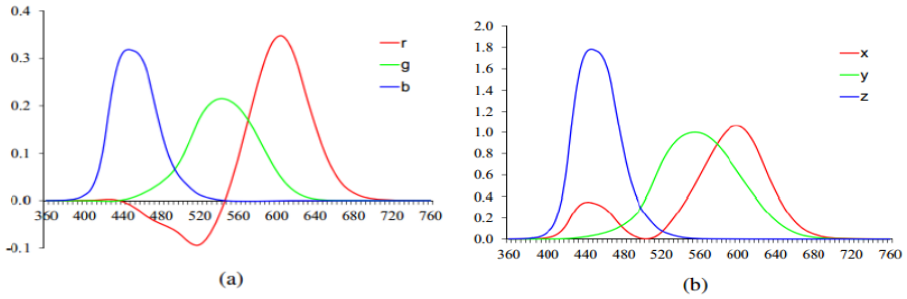
2.1.1. Técnica usando un umbral de intensidad, basada en el espacio de color CIE-xyY

En los años 30, la Comisión Internacional de la Iluminación (CIE) (de sus siglas en francés *Comission Internationale de l´Éclairage*, Viena- Austria 1931) fijó los estándares para la representación del espacio de color RGB Rojo, Verde y Azul (de siglas en inglés), basándose en una serie de experimentos realizados en 1920 por David Wright y John Guild. Esta comisión definió los tres colores primarios a partir de los cuales se puede crear todos los demás colores.

Los estándares vienen dados con la longitud de onda de cada color: 700.0nm (Red), 546.1nm (Green), 435.8nm (Blue). Debido a un problema asociado a cierto espectro del rango azul-verde, para solucionarlo había que agregarle una cantidad de luz roja negativa. La CIE desarrolló el actual sistema de color estándar XYZ que se sigue usando como referencia para definir los colores que percibe el ojo humano. Este sistema contiene todo el espectro de colores puros dentro de su octante positivo (Palmer, 2015).

Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 3. Funciones colorimétricas, en función de la longitud de onda λ para el observador patrón CIE 1931: (a) RGB, (b) XYZ



Fuente: Palmer (2015)

Para el cálculo y transformación del espacio RGB al nuevo espectro XYZ, Szeliski (2011) utiliza las siguientes matrices:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17696 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad Ec.(1.2.1)$$

Las coordenadas cromáticas se obtienen dividiendo la matriz XYZ para la suma $X+Y+Z$ (Szeliski, 2011):

$$x = \frac{X}{X+Y+Z} \quad Ec.(1.2.2)$$

$$y = \frac{Y}{X+Y+Z} \quad Ec.(1.2.3)$$

$$z = \frac{Z}{X+Y+Z} \quad Ec.(1.2.4)$$

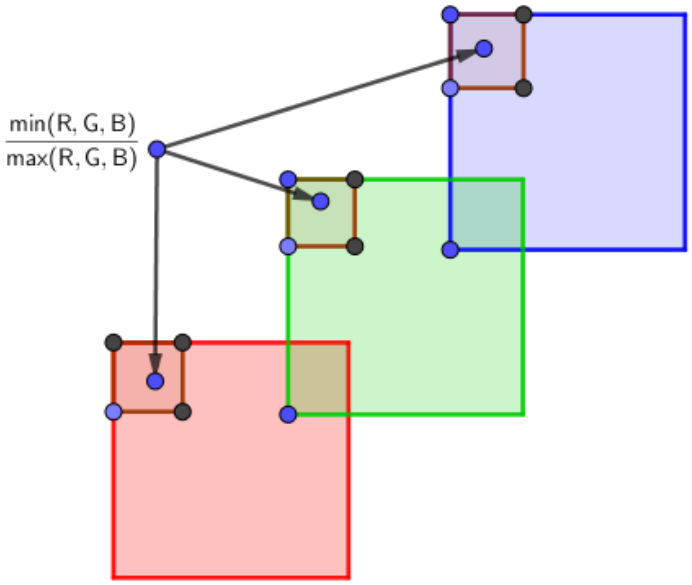
Cuando se requiera separar la luminosidad (Y) de la cromaticidad, se escoge el espacio de color xyY, y está formado por la luminosidad y los componentes cromáticas más distintivas (x, y).

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

En una colposcopía existen zonas con pixeles brillosos (zonas especulares) y zonas con tejidos pálidos y menos brillosos (zonas acetoblancas). Para distinguir una zona de la otra, se aumenta la diferencia entre ellas con la siguiente transformación aplicada a la imagen original:

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \frac{\min(R,G,B)}{\max(R,G,B)} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \qquad Ec.(1.2.5)$$

Figura 4. Máximos y mínimos de los pixeles de una imagen en el color RGB



Fuente: Elaboración propia

Para determinar y seleccionar la región especular (zonas de pixeles brillosos no pálidos) Meslouhi (2011) se utiliza el espacio de color xyY, debido a la ventaja de que en este espacio se separa la luminosidad de la cromaticidad. El criterio de selección es eligiendo los pixeles que tienen mayor luminosidad Y que la luminosidad cromática y. Esto se realiza con los siguientes pasos: (i) se realza la región especular utilizando la ecuación $Ec.(1.2.1)$, (ii) se transforma la

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

imagen del espacio de color RGB al espacio CIE-XYZ y se calcula la luminosidad usando la ecuación $Ec.(1.2.3)$, (iii) se procede a tomar como pixeles pertenecientes a la región especular aquellos que cumplen la condición de luminosidad $y<Y$ (Meslouhi et al., 2011). A continuación, se presenta el algoritmo.

Algoritmo 1.2.1: Extraer Región Especular (SR) usando espacio de color CIE-xyY

Entrada: Img - Imagen a extraer región especular

Salida: región especular SR

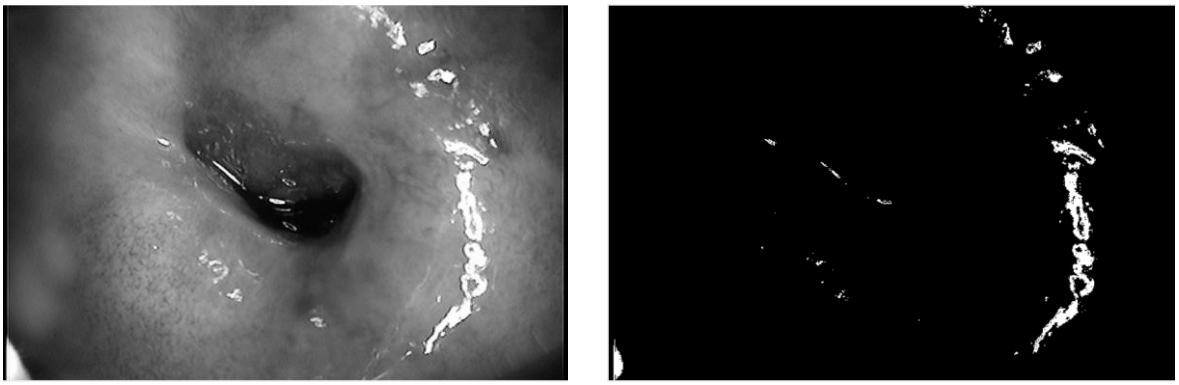
1. $SR = [\ ]$ región especular
2. Realzar región especular de $\mathrm{Im}g$ con $Ec.(1.2.5)$
3. $\mathrm{Img}XYZ = transformar\ \mathrm{Img}$ a espacio de color XYZ
4. For each p in $\mathrm{Img}XYZ$ do:
5. $Y = p.Y$
6. $y = \dfrac{p.Y}{p.X + p.Y + p.Z}$
7. if $y < Y$ do:
8. $SR.add(p)$
9. return SR

Fuente: Palmer, 2015)

El resultado de aplicar el algoritmo a una colposcopía mediante un código de Matlab, se presenta en la siguiente figura.

Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 5. Colposcopía equivalente detectada los pixeles de brillo por la técnica de la intensidad del umbral (derecha).



Fuente: Elaboración propia, mediante un código de Matlab

2.2. Modelo clásico de NMF

El modelo

Dada una matriz de entrada no negativa $V \in R_+^{m\times n}$ y un número natural $k < \min(m,n)$. Se busca $W \in R_+^{m\times n}$ y $H \in R_+^{m\times n}$ tal que

$$V \approx WH \quad Ec.(1.3.1)$$

El problema general (Pg) es:

$$Pg \begin{cases} \min \Phi(V, WH) \\ s.a\ W \geq 0, H \geq 0 \end{cases} \quad Ec.(1.3.2)$$

Donde, $\Phi(A,WH): \square^{m\times n} \times \square^{m\times n} \to \square_+$ es una distancia métrica conveniente o la divergencia de Kullback-Liebler (Baguer-Romañach, 2018).

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 6. Estructura geométrica de la factorización

Fuente: Elaboración propia

En este trabajo se usa la solución de Lee y Seung (1999) minimizando la norma de Frobenius entre V y WH ; además de las reglas de actualización multiplicativa.

Función costo $\Phi(V,WH)$:

$$\begin{aligned} \arg\min_{W,H} \quad & F(W,H)=\frac{1}{2}\left|V-WH\right|_F^2 \\ s.a. \quad & 0\le W,H \end{aligned} \qquad Ec.(1.3.3)$$

Reglas de actualización multiplicativa.

$$\begin{cases} H^{(nueva)} = H^{(vieja)} \square \dfrac{\left(W^{(vieja)}\right)^T V}{\left(W^{(vieja)}\right)^T W^{(vieja)} H^{(vieja)}} \\ W^{(nueva)} = W^{(vieja)} \square \dfrac{V\left(H^{(nueva)}\right)^T}{W^{(vieja)} H^{(nueva)} \left(H^{(nueva)}\right)^T} \end{cases} \qquad Ec.(1.3.4)$$

Donde el superíndice *vieja* se refiere a la iteración actual; *nueva* se refiere a la siguiente iteración; y T se refiere a la transpuesta de la matriz.

Norma de Frobenius escalada: $\left\|V-WH\right\|_F^2 = \dfrac{traza\left[\left(V-WH\right)^T\left(V-WH\right)\right]}{m\cdot n}$ $Ec(1.3.5)$

En otros trabajos, abordan la solución del problema minimizando la divergencia de Kullback-Leibler entre V y WH (Chih-Jen Lin, 2007).

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

$$\arg\min_{W,H} f(W,H) = \sum_{ij} A_{ij} \log \frac{A_{ij}}{(WH)_{ij}} - \sum_{ij} A_{ij} + \sum_{ij} (WH)_{ij} \quad Ec.(1.3.6)$$

$$\text{s.a. } 0 \le W, H$$

Las características del Pg (problema general) son las siguientes.

- Problema de optimización no lineal con restricciones de no negatividad.
- La función objetivo es no convexa(Rezaei et al., 2011).

Se usan métodos iterativos que comienzan con una aproximación inicial $W^{(0)}, H^{(0)}$ obtenida aleatoriamente o con cualquier otra estrategia y se generan sucesivamente aproximaciones con las reglas de actualización multiplicativa u otras reglas $W^{(1)}, H^{(1)}, W^{(2)}, H^{(2)}, .., W^{(k)}, H^{(k)}$ hasta que se satisfaga un determinado criterio de parada de la función objetivo (Baguer Díaz-Romañach, 2018).

2.3. La inicialización de NMF

El algoritmo original de NMF utiliza como datos iniciales las matrices W_0, H_0 cuyas matrices se generan aleatoriamente(Lee & Seung, 1999). Esto significa que, cada vez que se empieza se obtengan nuevas matrices W_0, H_0, lo que provoca que los resultados finales se alteren debido a los datos iniciales cada vez diferentes. Con este tipo de comienzo, no se puede esperar que haya consistencia, ni continuidad en los resultados finales. Esto se conoce como el problema de inicialización (Boutsidis & Gallopoulos, 2008).

Para resolver el problema de la inicialización de NMF, Boutsidis y Gallopoulos (2008) plantean un algoritmo de inicialización basado en SVD. Al contrario del método aleatorio sus resultados finales no dependen del inicio, puesto que cada vez que repetimos la inicialización, las matrices

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

iniciales W_0, H_0 permanecen invariantes. En su trabajo consideran que su algoritmo proporciona una reducción del error y una convergencia rápida.

Algoritmo NNDSVD

Nonnegative unit rank aproximation of arbitrary matrix
Input: Matrix $C \in \mathbb{R}^{m \times n}$
Output: Nonnegative $g \in \mathbb{R}_{+}^{m \times n}, h \in \mathbb{R}_{+}^{m \times n}$ so that $C \approx gh^T$
1. Compute the largest singular triplet of $C: [\sigma, u, v]$
2. Set $g = u_{+}, h = \sigma v_{+}$ where u_{+}, v_{+} are the nonnegative sections of u, v
for $j = 1, \ldots,$
3. Compute $g = \frac{Ch}{h^T h}$ and set $g = g_{+}$;
4. Compute $h = \frac{g^T C}{g^T g}$ and set h = h_{+};
end

El costo aritmético del cálculo del SVD es del orden $O\left(mn^2\right)$ cuando $m \geq n$, el SVD es caro computacionalmente pero la información que proporciona es bastante confiable (O'Leary, 2009).

2.4. Inpainting o restauración mediante NMF ponderado

El modelo

La idea para resolver el problema de restauración de imágenes usando NMF, se basa en encontrar la mejor factorización teniendo en cuenta solamente las componentes conocidas de la matriz que modela la imagen. Para esto supone una modificación al problema original y puede ser planteado de la siguiente manera(Gómez, 2018) (Baguer, 2018)

$$P_{m \times n} \square V_{m \times n} \approx P_{m \times n} \square \left(W_{m \times r} H_{r \times n}\right) \quad Ec.(1.5.1)$$

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Donde, V representa la imagen; W y H son los dos factores de V mediante NMF, y $P=\left[p_{ij} \right]$ es una matriz de unos y ceros que identifican los elementos de la matriz V, conocidos y desconocidos, siendo $p_{ij}=\begin{cases}1 \text{ si } v_{ij} \text{ es conocido} \\ 0 \text{ si } v_{ij} \text{ es desconocido}\end{cases}$ *Ec.*(1.5.2)

Este problema se resuelve minimizando el error cuadrático medio con la distancia de Frobenius entre la matriz original A y sus aproximaciones WH, la función objetivo es:

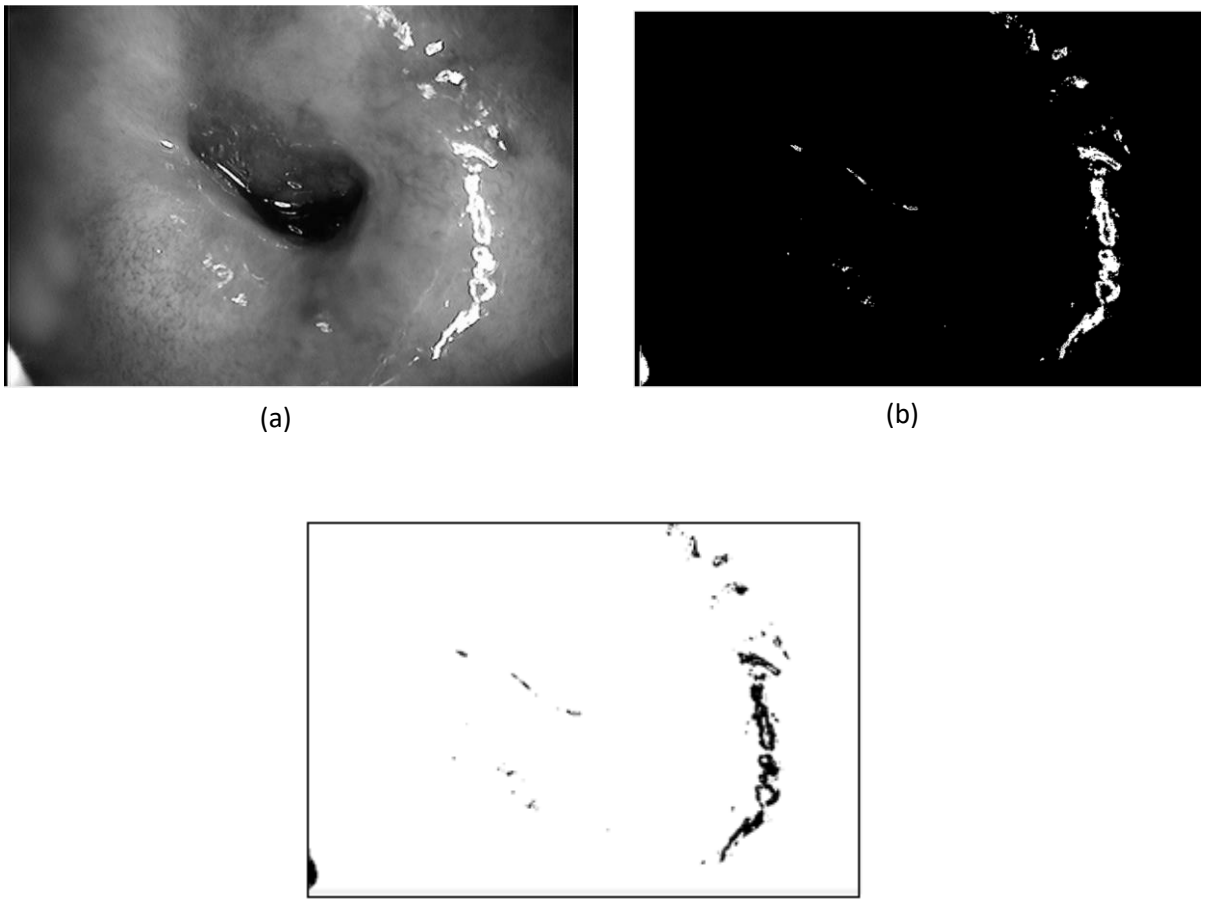
$$\begin{aligned} \arg\min_{W,H} \quad & F(W,H)=\frac{1}{2}\left|P\square\left(A-WH\right)\right|_F^2 \\ s.a. \quad & 0\le W,H \end{aligned} \qquad Ec.(1.5.3)$$

Reglas de actualización multiplicativa ponderado:

$$\begin{cases} H^{(nueva)}=H^{(viejo)}\square\dfrac{\left(W^{(vieja)}\right)^T\left(P\square V\right)}{\left(W^{(vieja)}\right)^T\left(P\square\left(W^{(vieja)}H^{(vieja)}\right)\right)} \\ W^{(nueva)}=W^{(vieja)}\square\dfrac{\left(P\square V\right)\left(H^{(nueva)}\right)^T}{\left(P\square W^{(vieja)}H^{(nueva)}\right)\left(H^{(nueva)}\right)^T} \end{cases} \qquad Ec.(1.5.4)$$ (Gómez, 2018)

Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 7. (a) Colposcopía de la Universidad de La Habana (izquierda), (b) la colposcopía anterior detectada los pixeles de brillo, (c) el complemento de la colposcopía anterior, retirada los pixeles de brillo (representa la matriz P del modelo anterior) (abajo)



(a) (b)

(c)

2.5. Métrica de calidad para las imágenes

El índice de similitud estructural SSIM (*Structural Similarity Index Measure*) es una métrica de calidad de imagen, que evalúa tres características: luminosidad, contraste y estructura. También se lo conoce como índice de calidad universal (UQI), o índice de Wang-Bovik, debidos a sus autores Zhou Wang y Alan Bovik en 2001. Con ayuda de otros autores esta medida evolucionó hasta lo que hoy conocemos. La adopción de SSIM por la comunidad de procesamiento de

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

imágenes e ingeniería de video, marcó el inicio del uso de una medida cuantitativa de la calidad de una imagen o un video (Bovik et al., 2005; Wang et al., 2004)

Algoritmos

El índice general es una combinación multiplicativa de los tres términos.

$$SSIM\left(x,y\right)=\left[l\left(x,y\right)\right]^{\alpha}\cdot\left[c\left(x,y\right)\right]^{\beta}\cdot\left[s\left(x,y\right)\right]^{\gamma}$$

Donde

$$l\left(x,y\right)=l\left(\mu_x,\mu_y\right)=\frac{2\mu_x\mu_y+C_1}{\mu_x{}^2\mu_y{}^2+C_1}\ \ ;\ \ C_1=\left(K_1L\right)^2;\ K_1 \square\ 1\ \text{y}\ L=255\ \text{(para 8-bits imagen en escala de grises)}$$

$$\mu_x=\frac{1}{N}\sum_{i=1}^{N}x_i$$

$$c\left(x,y\right)=c\left(\sigma_x,\sigma_y\right)=\frac{2\sigma_x\sigma_y+C_2}{\sigma_x{}^2+\sigma_y{}^2+C_2}\ \ ;\ \ C_2=\left(K_2L\right)^2;\ K_2 \square\ 1\ \text{y}\ L=255\ \text{(para 8-bits imagen en esc}$$

$$\sigma_x=\left(\frac{1}{N-1}\sum_{i=1}^{N}\left(x_i-\mu_x\right)^2\right)^{1/2}$$

$$s\left(x,y\right)=s\left(\frac{x-\mu_x}{\sigma_x},\frac{y-\mu_y}{\sigma_y}\right)=\frac{2\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3}$$

$$\sigma_{xy}=\frac{1}{N-1}\sum_{i=1}^{N}\left(x_i-\mu_x\right)\left(y_i-\mu_y\right)$$

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Capítulo 3. Estudio de los métodos

3.1. Método MAD

Este método se ocupa del problema de selección del rango en NMF, utilizando el procedimiento denominado validación cruzada CV (*cross validation*), la idea más común de este proceso se basa en la extracción de filas de la matriz original, como en el caso de aprendizaje supervisado. Luego de un aprendizaje con algún modelo elegido, los datos extraídos son reemplazados, para su posterior medición del error mediante una métrica adecuada (Kanagal & Sindhwani, s. f.).

En este trabajo, revisan una idea diferente a la idea original de CV en el contexto de NMF. La idea consiste en la retención del 10% de los datos siguiendo un patrón aleatorio de elección de los índices del total de datos originales. Los valores correspondientes a los índices seleccionados aleatoriamente se retiran a propósito de la matriz original y son reemplazados por cero; luego, se aplica un método de aproximación basado en NMF para ser calculados y reemplazados (imputados). Este proceso se repite muchas veces, mediante una función objetivo y cierto criterio de parada. Para la selección del rango, usan como parámetro la desviación media absoluta (MDA) del conjunto de errores de imputación (o de aproximación) entre la matriz V (matriz original de datos) y la matriz de aproximación V (matriz recalculada con el algoritmo MAD para un rango de rangos). Una vez que se seleccionan aleatoriamente el 10% de los datos de la matriz original V, se los retiene y los sustituyen por 0. Esto se realiza en la práctica experimental, definiendo una matriz de valores binarios P, donde se asigna el valor 1 para los valores presentes y 0 para los retenidos o ausentes. Luego se calcula el NMF ponderado sobre los datos originales con estos pesos, obteniendo así, la matriz de aproximación o imputación V que representa la

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

reconstrucción o restauración de la matriz original V, imputando así a los valores faltantes (ausentes). La matriz P se la denomina matriz de pesos y se define como sigue (Muzzarelli et al., 2019):

$$P = \left[p_{ij} \right], \text{ siendo } p_{ij} = \begin{cases} 1 & \text{si } v_{ij} \text{ está presente} \\ 0 & \text{si } v_{ij} \text{ está ausente o retenido} \end{cases} \quad Ec.(2.1.1)$$

Para la reconstrucción y posterior imputación de la matriz V se usa el método NMF ponderado, donde se ha modificado el problema original de Lee y Seung (1999) y puede ser planteado como en la ecuación 1.5.1. La matriz de pesos P permite hacer hincapié en dominos específicos para reconstruir ciertas entradas de la matriz V con preferencia de otras (Kanagal & Sindhwani, s. f.-a).

El NMF ponderado se resuelve minimizando la función objetivo de la ecuación 1.3.3. y con las reglas de actualización multiplicativa dadas en la ecuación 1.3.4. Este proceso se repite muchas veces (100 veces en el trabajo original, igual en este trabajo) para cada rango; con el afán de reducir la varianza debida a la aleatoriedad en la selección de los índices de ubicaciones de los valores retenidos. Al finalizar el proceso de repeticiones se obtiene la matriz de aproximación (o de imputación) $V = WH$, con esta se calcula el error de aproximación comparando la matriz original V y la matriz de imputación V, mediante la función objetivo para $mse[i]$ abajo definida en $Ec.(2.1.3)$. Se procede aplicar el mismo procedimiento por cada rango (dentro del rango de rangos) y se obtiene el conjunto denominado “conjunto de los errores de imputación”. En este conjunto se calcula la desviación media absoluta (MDA) de todos los errores y este estimador es el que indica el rango a ser elegido (Muzzarelli et al., 2019).

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

3.1.1. Función objetivo

Dada una matriz de entrada no negativa $V \in R_{+}^{m \times n}$ y un número natural $k < \min(m,n)$. Se busca una matriz de imputación $V_k^{m \times n}$ tal que:

$$\begin{cases} \min f(V, V_k) \\ s.a\ V \geq 0, V_k \geq 0 \end{cases} \qquad Ec.(2.1.2)$$

Donde, $f(V,V_k): \mathbb{R}^{m \times n} \times \mathbb{R}^{m \times n} \rightarrow \mathbb{R}_{+}$ es una distancia definida así:

$$f(V,V_k) = \frac{\sum_{t=1}^{m \times n} \left(V_k[muestra] - V[muestra]\right)^2}{n_{imp}} \qquad Ec.(2.1.3)$$

Donde n_{imp} es el número de datos retirados de la matriz original para ser imputados. Así $f(V,V_k)$ representa el error cuadrático medio entre ambas matrices, pero solo de los datos imputados.

El vector *muestra* contiene las coordenadas que indica los índices de los elementos de V que van a ser retirados aleatoriamente.

$V[muestra]$ contiene los elementos (y sus coordenadas) de la matriz original retirados a propósito para su posterior reemplazo en la imputación.

$V_k[muestra]$ contiene los elementos (y sus coordenadas) de la matriz de imputación o aproximación mediante NMF ($V_k[muestra] = W_k H_k$); k representa al rango de la simulación.

Recepción: 15-06-2024
Aprobación: 01-09-2024

3.1.2. Algoritmo

Tabla I. Pseudocode cálculo para MADIMPUT y MSEIMPUT

Entrada: matriz de datos V, número de valores faltantes m, rango de factores de interés f, número de iteraciones de validación cruzada v

Salida: MSE_{salida} y MAD_{salida}

Procedimiento:

Para cada factor de interés k en f hacer

$mse \leftarrow$ Vector de longitud v

Para cada iteración v hacer

$im \leftarrow$ Seleccionar aleatoriamente m índices de elementos

$Vm \leftarrow V$

$Vm[im] \leftarrow NAN$

$W, H \leftarrow NMF(Vm, k)$

$Vr \leftarrow W * H$

$mse[i] \leftarrow media(suma(Vr[im] - V[im])\wedge 2)$

end for

$MSE_{imput}^{k} \leftarrow median(mse)$

$MAD_{imput}^{k} \leftarrow mad(mse)$

end for

$MSE_{imput} \leftarrow \min(MSE_{imput}^{k})$

$MAD_{imput} \leftarrow \min(MAD_{imput}^{k})$

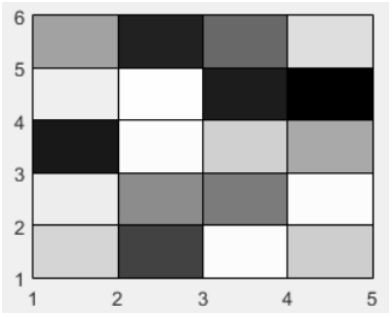
Recepción: 15-06-2024
Aprobación: 01-09-2024

3.1.3. Ejemplo pequeño del método MAD

Para explicar el método usaremos una matriz de entrada $V^{6\times5} \in \mathbb{R}_{+}^{m\times n}$ formada por números aleatorios que están en el rango $[0, 255]$. Estos valores representarán a los colores en la escala de grises, donde el mínimo valor $0 = negro$ y el más alto valor $255 = blanco$.

Figura 8. Matriz de números enteros no negativos aleatorios (izquierda), y sus tonalidades en escala de grises (derecha)

$$V = \begin{bmatrix} 208 & 71 & 245 & 202 & 173 \\ 231 & 140 & 124 & 245 & 193 \\ 32 & 245 & 204 & 167 & 190 \\ 233 & 247 & 36 & 9 & 100 \\ 161 & 40 & 107 & 217 & 167 \\ 24 & 248 & 234 & 239 & 43 \end{bmatrix} \Longrightarrow$$



La matriz original V es de dimensión 6x5, luego contiene 30 elementos. Estos los enumera con índices del 1 al 30, en orden de arriba hacia abajo, iniciando por la primera columna del 1 al 6, la segunda columna del 7 al 12 y así sucesivamente, es decir: $i_1 = v_{11}, i_2 = v_{21}, i_3 = v_{31}, ..., i_{30} = v_{65}$. En el trabajo original, eligen aleatoriamente el 10% de los 30 elementos de V. Para esto se usa la función de matlab *datasample*, la misma que elige automáticamene tres índices aleatorios de los 30 posibles. Los índices elegidos se los guarda en el vector *muestra*, por ejemplo:

$$muestra = \begin{bmatrix} 7 \\ 15 \\ 26 \end{bmatrix}$$

Los elementos de V que serán retirados, son los que corresponden con los índices 7, 15 y 26; es decir, $i_7 = v_{12}$ (fila 1 columna 2) , $i_{15} = v_{33}$ (fila 3 columna 3) y $i_{26} = v_{25}$ (fila 2 columna 5). A

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

estos tres elementos se los retira y en sus posiciones se coloca el valor cero en la matriz original V.

$$V=\begin{bmatrix} 208 & [71] & 245 & 202 & 173 \\ 231 & 140 & 124 & 245 & [193] \\ 32 & 245 & [204] & 167 & 190 \\ 233 & 247 & 36 & 9 & 100 \\ 161 & 40 & 107 & 217 & 167 \\ 24 & 248 & 234 & 239 & 43 \end{bmatrix} \Longrightarrow V^{(nueva)}=\begin{bmatrix} 208 & [0] & 245 & 202 & 173 \\ 231 & 140 & 124 & 245 & [0] \\ 32 & 245 & [0] & 167 & 190 \\ 233 & 247 & 36 & 9 & 100 \\ 161 & 40 & 107 & 217 & 167 \\ 24 & 248 & 234 & 239 & 43 \end{bmatrix}$$

La matriz de pesos P que será usada para el NMF ponderado se define como una matriz formada por unos y ceros; donde 1 corresponde al elemento presente y 0 al elemento retirado o ausente en la matriz V.

$$P=(p_{ij})\rightarrow p_{ij}=\begin{cases} 1 \text{ si } v_{ij} \text{ está presente} \\ 0 \text{ si } v_{ij} \text{ está retirado} \end{cases} \Longrightarrow Pesos=\begin{bmatrix} 1 & [0] & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & [0] \\ 1 & 1 & [0] & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Los parámetros de entrada faltantes son las matrices de inicialización de NMF W_0 y H_0. Estas se calculan usando la función de Matlab $[W,H] = NNDSVD(V,2,0)$ que corresponde al algoritmo NNDSVD (Boutsidis & Gallopoulos, 2008), donde V es la matriz original; “2” es el rango $(k=2)$ de inicio del intervalo de rangos; y “0” es una de tres variantes que posee el algoritmo NNDSVD (relacionada a una de las tres formas de hacer ceros a los números negativos presentes en el proceso de svd). Por notación usaremos: $W_{vieja}=W_0$, $H_{vieja}=H_0$

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Para $k = 2$ se calcula las matrices iniciales de NMF usando la función $[W, H] = NNDSVD(V, 2, 0)$ y se obtiene

$$W_0 = \begin{bmatrix} 13.5026 & 4.3751 \\ 14.0136 & 6.7105 \\ 12.7408 & 0.1000 \\ 8.9210 & 6.1902 \\ 10.4653 & 6.0531 \\ 12.3909 & 0.1000 \end{bmatrix} \qquad H_0 = \begin{bmatrix} 11.8984 & 13.2822 & 13.5632 & 15.3918 & 12.0338 \\ 11.1357 & 0.1000 & 0.1000 & 0.1000 & 3.8893 \end{bmatrix}$$

Con la matriz P y con las matrices $W_0 = W_{vieja}$ y $H_0 = H_{vieja}$, se procede hallar la primera restauración de $V^{(nueva)}$, esta es la imagen estimada $V = W_{nueva} H_{nueva}$ usando el NMF ponderado de las ecuaciones *Ec.*(1.3.3) y *Ec.*(1.3.4)

Para la primera iteración r=1, se obtiene:

$$Wnueva = \begin{bmatrix} 12.7835 & 9.0811 \\ 10.7157 & 13.7897 \\ 9.8245 & 19.9102 \\ 0.4155 & 18.2708 \\ 10.0673 & 6.8755 \\ 12.5001 & 4.9000e^{-324} \end{bmatrix}$$

$$Hnueva = \begin{bmatrix} 4.7765 & 3.5511e^{-83} & 15.8551 & 19.0092 & 7.7897 \\ 13.4785 & 11.9895 & 0.8449 & 1.9725e^{-04} & 6.4880 \end{bmatrix}$$

Se multiplica W_{nueva} y H_{nueva} y se obtiene la imagen estimada (o restaurada) $V = W_{nueva} H_{nueva}$

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

$$V=\begin{bmatrix} 183.459762382217 & [108.876893233277] & 210.355871608893 & 243.005153091386 & 158.497291761063 \\ 237.049155300077 & 165.331343147558 & 181.549647596186 & 203.699564803944 & [172.939945585509] \\ 315.287979926929 & 238.712868456925 & [172.591314345279] & 186.760358288244 & 205.707876640295 \\ 248.248781202868 & 219.057436006597 & 22.0242428485169 & 7.90107039915893 & 121.777595843271 \\ 140.758090562308 & 82.4333818769175 & 165.427001988190 & 191.372273131836 & 123.029329884199 \\ 59.7066052300012 & 4.43884244225042e^{-82} & 198.189565929836 & 237.615767956614 & 97.3715991763257 \end{bmatrix}$$

Se calcula el estimador del método MAD, en el código de Matlab de este trabajo se indica por la fórmula:

$$aux=\frac{\left(\sum_{t}\left(imagengris\left[muestra_t\right]-imagenestimada\left[muestra_t\right]\right)^2\right)}{n\_imputaciones} \qquad (3)$$

Donde, por un lado, el número de imputaciones n_imputaciones=3; mientras que $imagengris\left[muestra_t\right]$ contiene los tres elementos originales de V que fueron retirados a propósito:

$$imagengris\left[muestra_t\right]=\begin{bmatrix}71\\204\\193\end{bmatrix}$$

Por otro lado, $imagenestimada\left[muestra_t\right]$ contiene los tres elementos estimados o de imputación, luego de la primera iteración (r=1);

$$imagenestimada\left[muestra_t\right]=\begin{bmatrix}108.876893233277\\172.591314345279\\172.939945585509\end{bmatrix}.$$

Se calcula el estimador del método MAD mediante la ecuación 3. Este es el primer componente del vector de estimaciones $mse\_vector=aux$: $aux_1=47465.9188389131.$

Para r=2, se repite el proceso y se obtiene el segundo componente de estimación $aux_2=10800.3818092509$, así sucesivamente hasta r=100 se obtiene el componente 100-ésima

Recepción: 15-06-2024
Aprobación: 01-09-2024

$aux_{100} = 139262808.460024$. Una vez cumplidas las 100 repeticiones, el conjunto de las soluciones quedan almacenadas en el vector $mse\_vector = (aux_1, aux_2, \ldots, aux_{100})$. De este conjunto se calcula la desviación media absoluta, usando la función de Matalab mad, así se obtiene la solución del método que denominaremos MAD_2 para el respectivo rango; en este caso para $k = 2$.

Este proceso se repite nuevamente desde el principio para k=3,…,kmax; para algún rango (intervalo) de rangos, y lo resultados son almacenados en el vector $MADvector = (MAD_1, MAD_2, \ldots, MAD_k)$. Calculamos el mínimo de todos estos resultados de $MADvector$ y este reultado lo utilizamos para la selección del rango. El rango k seleccionado es igual a la posición de aquel que corresponde el mínimo del vector $MADvector$. En este ejemplo el valor mínimo es: $MAD_{\min} = 32080566{,}5099767$ que se encuentra en la posición 1. Por lo tanto, el rango mínimo seleccionado es $k = 1$.

Ejemplo 2. Observe ahora un ejemplo corrido para una matriz aleatoria un poco más grande $V^{10\times 20}$

Mediante la función *randi* se genera una matriz aleatoria de dimensión 10×20 que se la denomina imagengris,

```
imagengris=randi([0,255],10,20)
```

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

```
imagengris =

   251   253   158    33    89   122    25   139    31   161   155   179    38   214    29   158   193   127   153   111
     8    14    44    57   248   230    62    19     3   148   181   113    94   147   217   158   127   238   146   107
   105    48   162   204   250    61    93    59   143   140   130   230   173   132   146   199     7   120    49   138
   152   163   242   119   103    26    80   102    60    42    56    10   214   149     0    40   190   111   221   117
    63   250    62   113   249    53   103    61     3    40   164   146     0   140    52   143   118   190    84   124
   219   235   218   171    63   103   109    26    40   221    98    56   241   133    15   147   218   231   151   161
   169   167    40   138   178     1   178   118    26   222   103   111   105    90   193   185   164   203    35    42
    13    12   125   144   184    51    57   124   107   145     5   242    46   217   253    58   200   113    41   233
   252   226   225    44    59   140   175    14   132    86   122    93    50   117    47   227   123   246   108    10
   120    56   102    55   144   203   254   105   155   245   106   236   171   237    45   186    50   165    30    10
```

$MAD\min = 0.7753 \rightarrow k = 2$. El resultado del rango mínimo de esta matriz es 2.

3.2. Método MDL

La longitud mínima de descripción (MDL) es un método para seleccionar el rango entre modelos de complejidad variable (se refiere a la cantidad de datos), basada en la teoría de telecomunicaciones creada por Shanon en los años 50 y aplicada por Wallace y Bulton (1968) en el modelo MDL. La idea se centra en comprimir un mensaje de datos, de tal manera que, el mejor modelo MDL es aquel que comprima de mejor manera; es decir, el reto es enviar la mayor cantidad de información con la menor longitud posible; o sea, con el menor costo posible.

La compresión de los datos y la transmisión del mensaje son equivalentes, puesto que, la mejor forma de comprimir los datos, implica el menor costo de envío de un mensaje comprimido (codificado). Hay un emisor, un receptor y un acuerdo entre ambos, lo que se denomina precisión, al que deben ajustarse los datos. Para este trabajo, el mensaje original es la matriz $V^{m\times n} \in \square_{+}^{m\times n}$ de datos no negativos. El modelo de mensaje comprimido es la multiplicación de dos matrices no negativas WH, $W^{m\times k} \geq 0$, $H^{k\times n} \geq 0$, que aproxima al mensaje mediante $V \approx WH$.

La compresión del mensaje se realiza usando el método de factorización matricial no negativa NMF de Lee y Seung (1999), ecuación 1.3.3. El parámetro que determina la precisión del mensaje es el rango k (representa el nivel o grado de compresión de los datos) de la

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

factorización NMF. Cuando k es pequeño (lo que implica que W y H tenga pocos elementos que son baratos de codificar) la aproximación WH de V es pobre, requiriendo una adición al mensaje para corregir la mala aproximación. El principio de MDL es elegir un modelo que minimice la longitud total del mensaje. Al intercambiar ente la complejidad y la precisión, se espera encontrar un modelo que minimice la transmisión del ruido (características redundantes o menos importantes) y, a la vez, que maximice la transmisión de las características reales (Squires et al., 2017).

Mackay (2003) definió la función costo en MDL, usando probabilidades: $h(x) = -\log_2 P(x)$ donde x es un elemento y $P(x)$ es la probabilidad de que este ocurra.

La longitud del mensaje es igual a la suma de la longitud de envío, más la longitud del error (precisión). La longitud está asociada con el costo, mientras más largo es el mensaje más costoso será su envío y viceversa.

Longitud o costo del mensaje: $L(V,WH) = L(WH) + L(V \mid WH)$ $Ec.(2.2.1)$

Donde $L(WH)$ es la suma de las longitudes de los factores del modelo NMF

$$L(WH) = L(W) + L(H) \quad Ec.(2.2.2)$$

$L(V \mid WH)$ es la longitud del error, $L(V \mid WH) = L(E)$ $Ec.(2.2.3)$

$$\text{El error es } E = V - WH \quad Ec.(2.2.4).$$

Por lo tanto, la longitud o costo del mensaje es:

$$L(V,WH) = L(W) + L(H) + L(E) \quad Ec.(2.2.5)$$

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
Recepción: 15-06-2024
Aprobación: 01-09-2024

Squires et al. (2017) sugiere dos métodos para aplicar MDL a NMF. Un método usa la probabilidad clásica de Laplace al que le llaman método del histograma; y el otro se basa en una función de densidad de probabilidad Gamma. Ambos métodos se ajustan a las condiciones de no negatividad de NMF. Para lograr mejores resultados, realizan una partición de los datos en dos y aplican el método del histograma a una parte y el método de la probabilidad Gamma a la otra.

La partición la realizan de la siguiente manera, separan las matrices en dos grupos, los ceros y los no ceros mediante un corte o umbral. Esto lo realizan por dos razones. En la primera hay una oportunidad y un desafío, NMF tiende, de manera natural, a dar lugar a matrices escasas con una alta proporción de términos ceros. La *oportunidad* es que estos términos ser podrían ser enviados de manera muy barata como matrices de tasas separadas, y el *desafío* que estos ceros pueden dar lugar a distribuciones de probabilidad muy imprecisas. En la segunda, la distribución Gamma o bien cae a 0 o tiende al ∞ en cero, eso dependiendo del parámetro de la distribución. Si los datos que no son cero se ajustan mejor a una distribución que tiende a 0, entonces las estimaciones de las probabilidades serán muy pobres. Lo más grave es que podrá sobreestimar el costo de enviar los términos cero.

La partición se realiza no al mensaje original V, sino al modelo de aproximación WH; es decir, a ambos factores W y H, cada uno por separado. Ambos se separan en dos grupos, el uno de términos ceros y el otro de términos no ceros. Para esto, es necesario fijar un valor de corte o umbral. Por un lado, los valores por debajo del corte se consideran ceros (incluidos los ceros), estos se modelan con el método del histograma y la probabilidad clásica de Laplace; mientras que los valores por encima del corte se consideran no ceros y se modelan con la distribución de probabilidad Gamma.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

La separación de las matrices se hace de la manera siguiente.

$$W \rightarrow \begin{cases} W_0 \ , \mathrm{w}_{ij} < corte \text{ (ceros)} \\ W_m, \mathrm{w}_{ij} \geq corte \text{ (no ceros)} \end{cases} \qquad Ec.(2.2.6)$$

$$H \rightarrow \begin{cases} H_0, \mathrm{h}_{ij} < corte \text{ (ceros)} \\ H_m, \mathrm{h}_{ij} \geq corte \text{ (no ceros)} \end{cases} \qquad Ec.(2.2.7)$$

La nueva función coste o longitud del método MDL es:

$$L(V, WH) = L(W_0) + L(W_m) + L(H_0) + L(H_m) + L(E) \qquad Ec.(2.2.8)$$

La función que usa Squires et al. (2017) en el método del histograma, para modelar las matrices cuyos términos menores al corte (ceros) es:

$$L(X_0) = -nX_0 * \log_2\left(\frac{nX_0}{ntX}\right) - (ntX - nX0) * \log_2\left(\frac{ntX - nX_0}{ntX}\right) \qquad Ec.(2.2.9)$$

donde X_0 representa a W_0 y H_0, aplicadas a cada una de las matrices en cuestión, resulta:

$$LW_0 = -nW_0 * \log_2\left(\frac{nW_0}{ntW}\right) - (ntW - nW_0) * \log_2\left(\frac{ntW - nW_0}{ntW}\right) \qquad Ec.(2.2.10)$$

$$LH_0 = -nH_0 * \log_2\left(\frac{nW_0}{ntH}\right) - (ntH - nH_0) * \log_2\left(\frac{ntH - nH_0}{ntH}\right) \qquad Ec.(2.2.11)$$

La función que usa Squires et al. (2017) en el método de la distribución Gamma, para modelar las matrices cuyos términos son mayores al corte es:

$$L(Wm, Hm) = \sum_i \sum_j \log_2 P(Wm_{ij}) + \sum_i \sum_j \log_2 P(Hm_{ij}) + \sum_i \sum_j \log_2 P(E_{ij}) \qquad Ec.(2.2.12)$$

Book Citation Index
Web of Sciece/Core Collection
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Donde $P(Wm_{ij})$ y $P(Hm_{ij})$ representan las probabilidades de la distribución Gamma ajustadas a las matrices Wm y Hm; y el último sumando tiene $P(E_{ij})$ representa la probabilidad de la matriz de errores $E = V - WH$, ajustada a una distribución Normal o de Gauss con media $\mu = 0$ y varianza σ^2.

Fusionando las dos funciones, en una sola, se llega a la función objetivo del método MDL.

3.2.1. Función objetivo

Dada una matriz de entrada no negativa $V \in R_+^{m \times n}$ y un número natural $k < \min(m,n)$. Se busca $W \in R_+^{m \times n}$ y $H \in R_+^{m \times n}$ tal que

$$V \approx WH \quad Ec.(2.2.1.1)$$

$$\begin{cases} \min L(V, WH) \\ s.a\ W \geq 0, H \geq 0 \end{cases} \quad Ec.(2.2.1.2)$$

Donde, $L(A, WH) : \mathbb{R}^{m \times n} \times \mathbb{R}^{m \times n} \rightarrow \mathbb{R}_+$ es una distancia métrica definida así:

$$\begin{aligned} L(V,WH) = & -nW_0 * \log_2\left(\frac{nW_0}{ntW}\right) - (ntW - nW_0) * \log_2\left(\frac{ntW - nW_0}{ntW}\right) + \sum_i \sum_j \log_2 P(Wm_{ij}) \\ & - nH_0 * \log_2\left(\frac{nW_0}{ntH}\right) - (ntH - nH_0) * \log_2\left(\frac{ntH - nH_0}{ntH}\right) + \sum_i \sum_j \log_2 P(Hm_{ij}) \\ & + \sum_i \sum_j \log_2 P(E_{ij}) \end{aligned} \quad Ec.(2.2.3)$$

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

3.2.2. Algoritmo

A continuación, el seudocódigo original del método MDL de Squires et al. (2017)

Algorithm 1 MDL algorithm for each k value automatic moving zero threshold

---

Input: $V, W, H, \partial D$

Output: Description lengths for each k

1: for zero threshold values de $W\ y\ H$

2: Separate out zero values, calculate $L(W_0)\ and\ L(H_0)$

3: Apply gamma distributions to $W_+\ and\ H_+$, calculate $L(W_+)\ and\ L(H_+)$

4: Calculate $E\ then\ L(E)$

5: Calculate $L\ (\mathcal{D}, \mathcal{H})$

6: If $L\ (\mathcal{D}, \mathcal{H})$ is smaller than previous smallets, then store description lengths end if

7: end for

8: Return $L\ (\mathcal{D}, \mathcal{H})$

Donde: $\mathcal{D} = V$; $\mathcal{H} = WH$; $W_+ = W_m$; $H_+ = H_m$; ∂D=precisión (asociado al rango k)

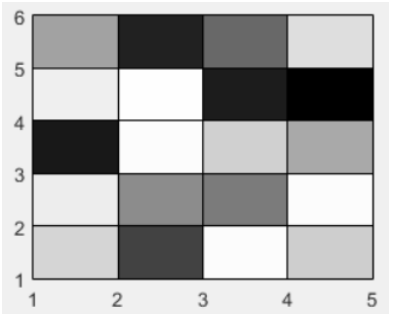
3.2.3. Ejemplo pequeño del método MDL

Para explicar el método usaremos una matriz de entrada $V^{6\times5} \in \mathbb{R}_+^{m\times n}$ formada por números aleatorios que están en el rango $[0, 255]$, estos valores representarán a los colores en la escala de grises, donde el mínimo valor $0 = negro$ y el más alto valor $255 = blanco$.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 9. Matriz de números enteros no negativos aleatorios (izquierda), y sus tonalidades en escala de grises (derecha)

$$V = \begin{bmatrix} 208 & 71 & 245 & 202 & 173 \\ 231 & 140 & 124 & 245 & 193 \\ 32 & 245 & 204 & 167 & 190 \\ 233 & 247 & 36 & 9 & 100 \\ 161 & 40 & 107 & 217 & 167 \\ 24 & 248 & 234 & 239 & 43 \end{bmatrix} \Longrightarrow$$



Como parámetros de entrada es necesario la matriz original V (arriba), un intervalo de rangos $k = k_1,\ \mathrm{k}_2,\ \mathrm{k}_3,\ \ldots,\ k_n$ de V, y las matrices de inicialización de NMF W_0 y H_0. Estas se calculan usando la función de Matlab $[W,H] = NNDSVD(V,1,0)$ que corresponde al algoritmo NNDSVD (Boutsidis & Gallopoulos, 2008), donde V es la matriz original, "1" es el rango de inicio del intervalo de rangos, y "0" es una de tres variantes que posee el algoritmo NNDSVD (relacionada con una de las tres formas de hacer ceros a los números negativos presentes en el proceso de SVD). Por notación usaremos:

$$W_{vieja} = W_0\ ,\ H_{vieja} = H_0$$

Por ejemplo, para $k = 1$ las matrices iniciales calculadas con NNSVD son:

$$W_{vieja} = \begin{bmatrix} 13.5026 \\ 14.0136 \\ 12.7408 \\ 8.9210 \\ 10.4653 \\ 12.3903 \end{bmatrix} \qquad H_{vieja} = \begin{bmatrix} 11.8994 & 13.2822 & 13.5632 & 15.3918 & 12.0338 \end{bmatrix}$$

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Se factoriza la matriz V usando el algoritmo NMF (Lee & Seung, 1999) y utilizando las ecuaciones 1.3.2 y 1.3.3. Obtenemos, la primera factorización W_{nueva} y H_{nueva} y sus productos, se denomina imagen restaurada $W_{nueva}H_{nueva} = imagen\_restaurada$:

$$imagen\_restaurada = \begin{bmatrix} 160.6603 & 179.3454 & 183.1385 & 207.8296 & 162.4886 \\ 166.7399 & 186.1320 & 190.0687 & 215.6942 & 168.6374 \\ 151.5952 & 169.2260 & 172.8052 & 196.1031 & 153.3204 \\ 106.1461 & 118.4911 & 120.9971 & 137.3102 & 107.3541 \\ 124.5205 & 139.0025 & 141.9424 & 161.0794 & 125.9376 \\ 147.4318 & 164.5783 & 168.0592 & 190.7173 & 149.1096 \end{bmatrix}$$

Calculamos la matriz error E usando la ecuación 2.2.4. Se fija un *corte* que depende del tipo de datos que se tenga (Squires et al., 2017). Para este ejemplo, se tomó un $corte = 12.05$ (para obtener un par de valores por debajo del corte al inicio del experimento). Mediante este corte se dividen las matrices nuevas W_{nueva} y H_{nueva} (después de cada factorización) en dos grupos de matrices. En el primer grupo están las matrices menores al corte W_0 y H_0 (matrices de ceros); en el segundo grupo se hallan las matrices complementarias con los valores mayores al corte Wm y Hm (matrices de valores distinitos de cero). Los dos grupos de matrices del ejemplo quedan conformadas así:

$$W_0 = \begin{bmatrix} 8.9210 \\ 10.4653 \end{bmatrix} \qquad H_0 = \begin{bmatrix} 11.8984 & 12.0338 \end{bmatrix}$$

$$W_m = \begin{bmatrix} 13.5026 \\ 14.0136 \\ 12.7408 \\ 12.3909 \end{bmatrix} \qquad H_m = \begin{bmatrix} 13.2822 & 13.5632 & 15.3918 \end{bmatrix}$$

Recepción: 15-06-2024
Aprobación: 01-09-2024

Siguiendo el modelo planteado por Squires et al.,(2017), se calculan los siguientes parámetros para su modelo. Se supone que los datos de las nuevas matrices W y H se ajustan a la función Gamma; y en cambio que la matriz de errores E se ajusta (definida arriba) a la distribución Normal o de Gauss con media 0. Los parámetros se calculan con el uso de algunas funciones de Matlab; las matrices W y H de los parámetros escritos se refieren a las matrices actualizadas W_{nueva} y H_{nueva}.

$nt = M * N = 6*5 = 30$ →Número total de elementos de V

$ntW = M * k = 6*1 = 6$ →Número de elementos de W

$ntH = k * N = 1*5 = 5$ →Número de elementos de H

$sigma2 = 5.5628e+03$ → Varianza de la matriz de error E (usamos $sum(sum(E.*E))$ de Matlab)

$abw = [430.6437 \quad 0.0306]$ → Cálculo de los parámetros de entrada “alpha” y “beta” para la función Gamma de la matriz W_m (mediante la función de Matlab gamfit (W_m)), donde

$\alpha = abw(1) = 430.6437; \beta = abw(2) = 0.0306$

$abh = [232.08319 \quad 0.0605]$→ Cálculo de los parámetros “alpha” y “beta” para la función Gamma de la matriz H_m (mediante la función de Matlab gamfit (H_m))

$nW_0 = 2$ →Número de elementos por debajo del corte (ceros) en la primera iteración (usamos length (W_0))

$ntW - nW_0 = 6 - 2 = 4$ →Número de elementos por encima del corte (no ceros W)

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Según el modelo de Squires et al., calculamos las longitudes o costos para cada una de las matrices creadas: W_0, W_m, H_0, H_m y E mediante las ecuaciones 2.2.10 y 2.2.11 sus valores son:

$$LW0 = 5.5098; \quad LH0 = 4.8548$$

Para determinar las longitudes o costos de las matrices Wm y Hm (cuyos valores son mayores que el *corte)*, se usa la función objetivo $Ec.(2.2.12)$ (develada por el autor en un segundo trabajo, dos años después del primero sobre MDL-NMF).

De la ecuación $Ec.(2.2.12)$, separando los sumados de la función, se calculan cada uno de los términos por separado mediante Matlab, así:

Primer sumando $L(Wm) = \sum_i \sum_j \log_2 P(Wm_{ij})$ se calcula con:

$$LWm = -sum\left(sum\left(\log_2\left(gampdf\left(Wm, abw(1), abw(2)\right)\right)\right)\right)$$

Segundo sumando $L(Hm) = \sum_i \sum_j \log_2 P(Hm_{ij})$ se calcula con:

$$LHm = -sum\left(sum\left(\log_2\left(gampdf\left(Hm, abh(1), abh(2)\right)\right)\right)\right)$$

Tercer sumando $L(E) = \sum_i \sum_j \log_2 P(E_{ij})$ se calcula con:

$$L(E) = -sum\left(sum\left(\log_2\left(normpdf\left(E, 0, sqrt\left(sigma2\right)\right)\right)\right)\right)$$

Sus resultados son:

$$LWm = 5.5564; \quad LHm = 5.7868; \quad LE = 248.0369$$

Recepción: 15-06-2024
Aprobación: 01-09-2024

Sumándolos todos tal como dice la ecuación 2.2.13, se obtiene el valor de la métrica MDL-NMF para la el rango $k=1$:

$$MDL = 269.7446.$$

Se repite el proceso desde el inicio para el rango $k=2$. Se factoriza V mediante NMF (ecuaciones 1.3.2 y 1.3.3) y se obtinen las nuevas matrices iniciales de factorización:

$$H_{nueva} = \begin{bmatrix} 0.0156 & 16.3167 & 16.5691 & 14.9236 & 6.7840 \\ 14.6926 & 0.0002 & 0.0393 & 3.9792 & 8.0285 \end{bmatrix} \qquad w_{nueva} = \begin{bmatrix} 9.6363 & 14.0219 \\ 9.1673 & 16.4485 \\ 13.2353 & 3.8817 \\ 4.9557 & 12.4758 \\ 6.5714 & 12.8982 \\ 14.5552 & 0.0000 \end{bmatrix}$$

Se particionan las matrices por encima y por debajo del $corte = 12.05$

$$W_0 = \begin{bmatrix} 9.6363 \\ 9.1673 \\ 4.9557 \\ 6.5714 \\ 3.8817 \\ 0.000 \end{bmatrix} \quad W_m = \begin{bmatrix} 13.2353 \\ 14.5552 \\ 14.0219 \\ 16.4485 \\ 12.4758 \\ 12.8982 \end{bmatrix} \quad H_0 = \begin{bmatrix} 0.0156 \\ 0.0002 \\ 0.0393 \\ 3.9792 \\ 6.7840 \\ 8.0285 \end{bmatrix} \quad H_m = \begin{bmatrix} 14.6926 \\ 16.3167 \\ 16.5691 \\ 14.9236 \end{bmatrix}$$

Se realizan todos los cálculos y se obtiene el valor de la métrica MDL-NMF para $k=2$:

$$MDL = 277.2305.$$

Si repetimos el proceso desde el inicio para $k=3$, y así sucesivamente, para un intervalo de rangos fijados, (en este caso como el ejemplo es pequeño se corrió para todos los rangos), entonces se obtienen los valores de la métrica en el vector MDLvector:

$$\text{MDLvector} = [269{,}744624965437 \quad 277{,}230529242417 \quad 251{,}651163537597 \quad 253{,}820069114855 \quad 36{,}2147719026729]$$

Recepción: 15-06-2024
Aprobación: 01-09-2024

El problema consistía en minimizar esta métrica, por lo tanto se toma el valor mínimo de MDL vector y este es 36.21 y corresponde al rango $k = 5$; por lo tanto el rango mínimo seleccionado es $\min k = 5$.

Ejemplo 2. MDL-NMF

Sea

$$V = \begin{bmatrix}
60 & 140 & 237 & 164 & 53 & 79 & 152 & 21 & 246 & 9 & 27 & 7 & 46 & 15 & 168 & 44 & 74 & 243 & 77 & 8 \\
90 & 75 & 198 & 96 & 77 & 236 & 67 & 67 & 139 & 226 & 167 & 190 & 61 & 174 & 132 & 100 & 110 & 235 & 179 & 143 \\
210 & 190 & 124 & 207 & 120 & 110 & 154 & 205 & 133 & 233 & 126 & 128 & 226 & 10 & 249 & 212 & 3 & 13 & 170 & 225 \\
3 & 48 & 111 & 136 & 59 & 47 & 182 & 7 & 59 & 203 & 199 & 122 & 7 & 18 & 166 & 205 & 251 & 188 & 138 & 171 \\
11 & 175 & 114 & 89 & 216 & 231 & 56 & 237 & 125 & 25 & 183 & 231 & 125 & 133 & 204 & 15 & 42 & 68 & 178 & 48 \\
43 & 46 & 78 & 240 & 49 & 250 & 30 & 186 & 159 & 67 & 231 & 156 & 42 & 24 & 116 & 102 & 27 & 108 & 170 & 94 \\
166 & 94 & 130 & 224 & 57 & 112 & 75 & 125 & 173 & 85 & 228 & 158 & 250 & 209 & 110 & 134 & 95 & 140 & 45 & 117 \\
187 & 160 & 130 & 140 & 43 & 28 & 81 & 148 & 101 & 174 & 85 & 220 & 182 & 209 & 211 & 106 & 50 & 241 & 32 & 251 \\
165 & 199 & 209 & 159 & 58 & 66 & 108 & 60 & 94 & 34 & 178 & 206 & 128 & 184 & 21 & 168 & 125 & 106 & 255 & 40 \\
115 & 20 & 203 & 150 & 111 & 104 & 130 & 117 & 252 & 184 & 50 & 147 & 120 & 38 & 34 & 160 & 86 & 251 & 43 & 219
\end{bmatrix}$$

Los valores de la métrica para $k = 1, 2, 3, ..., 10$ están en el vector MDLvector:

MDLvector = [NaN 1631,00446676432 1605,86461779847 1589,31930897531

1542,61035191103 1496,20892315308 1434,60387711309 NaN NaN NaN]

El mínimo de MDLvector es $1.4346e^{+03}$ y corresponde al rango $k = 7$ por lo tanto el rango mínimo seleccionado es $\min k = 7$.

3.3. Método SUR E

Este método está basado en el estimador de riesgo no sesgado de Stein (1981) SURE (de sus siglas en inglés *Stein's Unbiased Risk Estimator*), denominado así en honor a Charles Stein, quien formuló un teorema sobre la distribución normal estándar, cuya principal aplicación es en la inferencia estadística (Tibshirani, 2015).

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

La idea del método SURE consiste en desarrollar un estimador computable no sesgado del error cuadrático medio (MSE de sus siglas en inglés) que puede ser útil en la determinación del rango (Ulfarsson & Solo, 2013).

Se considera el modelo

$$V^{m\times n} = W^{m\times k} H^{k\times n} + \varepsilon^{m\times n} \text{ donde } V, W, H \geq 0 \qquad Ec.(2.3.1)$$

$$v^{ij} = \mu^{ij} + \varepsilon^{ij}, \quad i = 1,...,m, \ \ j = 1,...,n \ \text{ donde } \varepsilon^{ij} \square N\left(0, \sigma^2\right) \qquad Ec.(2.3.2)$$

El MSE para $\mu^{ij} = \mu^{ij,k}(V)$ es un estimador de $\mu^{ij} = [WH]^{ij}$ donde k es el rango. Se calcula mediante la siguiente ecuación (ver demostración en el apéndice de Ulfarsson & Solo, 2013):

$$R_k = \sum_{ij} E\left[\left(\mu^{ij} - \mu^{ij}\right)^2\right] = \sum_{ij}\left(e^{ij}\right)^2 - 2\sum_{ij} E\left[e^{ij}\varepsilon^{ij}\right] + nm\sigma^2 \qquad Ec.(2.3.3)$$

Donde $e^{ij} = v^{ij} - u^{ij}$ es el residuo (error de aproximación); $\varepsilon^{ij} = v^{ij} - \mu^{ij}$ y

$$E\left[e^{ij}\varepsilon^{ij}\right] = E\left[e^{ij}\left(v^{ij} - \mu^{ij}\right)\right] \qquad Ec.(2.3.4)$$

$$= \sigma^2 E\left[\frac{de^{ij}}{dv^{ij}}\right] \quad \text{Lema de (Stein, 1981)}$$

$$Ec.(2.3.5) \qquad = \sigma^2 - \sigma^2 E\left[\frac{d\mu^{ij}}{dv^{ij}}\right].$$

$$Ec.(2.3.6)$$

Reemplazando en R_k se obtiene

$$R_k = \sum_{ij}\left(e^{ij}\right)^2 + 2\sigma^2 \sum_{ij} E\left[\frac{d\mu^{ij}}{dv^{ij}}\right] - nm\sigma^2 \qquad Ec.(2.3.7)$$

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Al hacer reajustes a los términos irrelevantes se obtiene SURE

$$R_k = \sum_{ij}\left(e^{ij}\right)^2 + 2\sigma^2 \sum_{ij} \frac{d\mu^{ij}}{dv^{ij}} \qquad Ec.(2.3.8)$$

Se puede demostrar (ver el apéndice de (Ulfarsson & Solo, 2013)) que la derivada $\frac{d\mu^{ij}}{dv^{ij}}$ es igual a:

$$\frac{d\mu^{ij}}{dv^{ij}} = (m+n)k - \sum_{i=1}^{k}\sum_{j=1}^{n-k} \frac{2\left(\lambda^j\right)^2}{\left(\lambda^j\right)^2 - \left(\rho^i\right)^2} \qquad Ec.(2.3.9)$$

Donde $\left(\rho^i\right)^2$ es el i-ésimo valor propio de $K^T K W^T W$; $K = H^T$ y $\left(\lambda^j\right)^2$ es el j-ésimo valor propio de $E^T E$ siendo $E = V - WH$.

Juntando todo en la $Ec.(2.3.3)$ se obtiene la función objetivo de SURE para NMF

$$R_k = \sum_{ij}\left(e^{ij}\right)^2 + 2\sigma^2 \left((m+n)k - \sum_{i=1}^{k}\sum_{j=1}^{n-k} \frac{2\left(\lambda^j\right)^2}{\left(\lambda^j\right)^2 - \left(\rho^i\right)^2} \right) \qquad Ec.(2.3.10)$$

3.3.1. Función objetivo

Dada una matriz de entrada no negativa $V \in R_+^{m\times n}$ y un número natural $k < \min(m,n)$. Se busca un estimador $R_k \in \mathbb{R}_+$ tal que:

$$\begin{cases} \min f(V, R_k) \\ s.a\ V \geq 0, R_k \geq 0 \end{cases} \qquad Ec.(2..3.1)$$

Donde, $f(V, R_k): \mathbb{R}^{m\times n} \to \mathbb{R}_+$ es una distancia definida así:

$$R_k = \sum_{ij}\left(e^{ij}\right)^2 + 2\sigma^2 (m+n)k - 2\sigma^2 \left(\sum_{i=1}^{k}\sum_{j=1}^{n-k} \frac{2\left(\lambda^j\right)^2}{\left(\lambda^j\right)^2 - \left(\rho^i\right)^2} \right) \qquad Ec.(2..3.2)$$

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Donde

$V = \left[v^{ij}; i = 1,...,m; j = 1,...,n \right]$

$V \approx WH$ es la aproximación de V con WH

W y H son los factores de V con NMF y rango k

$W = \left[w^{ij}; i = 1,...,m; j = 1,...,k \right]$; $H = \left[h^{ij}; i = 1,...,k; j = 1,...,n \right]$

$E = V - WH$ es la matriz de errores de la factorización con NMF; $E = \left(e^{ij}; i = 1,...,m; j = 1,...,n \right)$

σ es una varianza de ajuste de E

λ^j j-ésimo valor propio de $E^T E$

ρ^i i-ésimo valor propio de $K^T K W^T W$; $K = H^T$

3.3.2. Algoritmo

Datos de entrada: V, rango de rangos k

1. W_0, H_0 : Inicializar NMF(V, k)
2. Calcular los valores propios al cuadrado $\left(\rho^i \right)^2$ de $K^T K W^T W$
3. Calcular los valores propios al cuadrado $\left(\lambda^j \right)^2$ de $E^T E$
4. Calcular la varianza $\sigma = \dfrac{\frac{1}{n} \sum_{p=1}^{n} mediana\left(\left| E_{ij} \right| \right)}{0.6745}$
5. Calcular el estimador R_k

Salida: min R_k

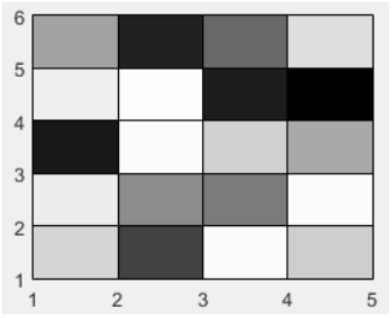
Recepción: 15-06-2024
Aprobación: 01-09-2024

3.3.3. Ejemplo pequeño del método SURE

Para explicar el método usaremos una matriz de entrada $V^{6\times5} \in \mathbb{R}_{+}^{m\times n}$ formada por números aleatorios que están en el rango $[0, 255]$. Estos valores representarán a los colores en la escala de grises, donde el mínimo valor $0 = negro$ y el más alto valor $255 = blanco$.

Figura 10. Matriz de números enteros no negativos aleatorios (izquierda), y sus tonalidades en escala de grises (derecha)

$$V = \begin{bmatrix} 208 & 71 & 245 & 202 & 173 \\ 231 & 140 & 124 & 245 & 193 \\ 32 & 245 & 204 & 167 & 190 \\ 233 & 247 & 36 & 9 & 100 \\ 161 & 40 & 107 & 217 & 167 \\ 24 & 248 & 234 & 239 & 43 \end{bmatrix} \Longrightarrow$$



Como parámetros de entrada es necesario la matriz original V (arriba), un intervalo de rangos $k = k_1,\ \mathrm{k}_2,\ \mathrm{k}_3,\ \ldots,\ k_n$ de V; y las matrices de inicialización de NMF W_0 y H_0. Estas se calculan usando la función de Matlab $[W, H] = NNDSVD(V, 1, 0)$ que corresponde al algoritmo NNDSVD (Boutsidis & Gallopoulos, 2008), donde V es la matriz original, "1" es el rango de inicio del intervalo de rangos, y "0" es una de tres variantes que posee el algoritmo NNDSVD (relacionada con una de las tres formas de hacer ceros a los números negativos presentes en el proceso de SVD).

Recepción: 15-06-2024
Aprobación: 01-09-2024

Se calcula el estimador SURE con la ecuación $Ec.(2.3.10)$ con los datos de entrada V y un rango de rangos $k=1,2,3,4,5$. Para cada valor del rango se obtiene un valor de SURE, el criterio de selección del rango será de acuerdo al mínimo de todos los valores SURE obtenidos.

Se recuerda que la notación en las matrices cuyo subíndice sea "*vieja*" indicará que es la iteración actual y el subíndice "*nueva"* que es la iteración siguiente: W_{vieja}, H_{vieja} y W_{nueva}, H_{nueva}

Luego de inicializar la matriz V se hace con el código NNSVD, se obtienen: W_{vieja}, H_{vieja}

$$W_{vieja}=\begin{bmatrix}13.5026\\14.0136\\12.7408\\8.9210\\10.4653\\12.3909\end{bmatrix}\quad H_{vieja}=\begin{bmatrix}11.8984 & 13.2822 & 13.5632 & 15.3918 & 12.0338\end{bmatrix}$$

Se factoriza la matriz V usando las ecuaciones $Ec.(1.3.3)$ y $Ec.(1.3.4)$ y se obtiene, la primera factorización y sus productos, la denominamos imagen estaurada $W_{nueva}H_{nueva}=imagen\_restaurada$, esto es:

$$imagen\_restaurada=\begin{bmatrix}160.6603 & 179.3454 & 183.1385 & 207.8296 & 162.4886\\166.7399 & 186.1320 & 190.0687 & 215.6942 & 168.6374\\151.5952 & 169.2260 & 172.8052 & 196.1031 & 153.3204\\106.1461 & 118.4911 & 120.9971 & 137.3102 & 107.3541\\124.5205 & 139.0025 & 141.9424 & 161.0794 & 125.9376\\147.4318 & 164.5783 & 168.0592 & 190.7173 & 149.1096\end{bmatrix}$$

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Calculamos el resto de parámetros necesarios para hallar el estimador SURE, usando la ecuación $Ec.(2.3.11)$. Primero se calcula $E = V - imagen\_restaurada$, y con esto se realiza $E.*E$ que produce los elementos al cuadrado de E, usando la función *sum* de Matlab para matrices, se calcula la suma de todos los elementos al cuadrado:

$$\sum_{ij} e_{ij}^2 = sum(sum(E.*E))$$

La primera operación *sum* calcula la suma de los elementos de $E.*E$ por columnas y los almacena en un vector. La segunda operación *sum* calcula la suma de todos los elementos del vector de la primera operación, se obtiene:

$$\sum_{ij} e_{ij}^2 = 8.7198e{+}04$$

Para calcular la varianza de este método se usa la fórmula $\sigma = \dfrac{\frac{1}{n}\sum_{p=1}^{n} mediana\left(\left|E_{ij}\right|\right)}{0.6745}$, para su cálculo se combinan las funciones de Matlab *median* y *mean*, su valor es:

$$\sigma = 49.017270265626685$$

Para el cálculo de los valores propios se usa la función de Matlab *eig*, pero antes se calculan los productos de matrices que se requieren:

$$m1 = E^T * E$$
$$m2 = H_{nueva} * H_{nueva}^{\mathrm{T}} * W_{nueva}^{\mathrm{T}} * W_{nueva}$$

$$\lambda = eig\left(E^T * E\right) \qquad \rightarrow \rho = 781110{,}466302475$$

$$\rho = eig\left(H_{nueva} * H_{nueva}^{\mathrm{T}} * W_{nueva}^{\mathrm{T}} * W_{nueva}\right) \qquad \rightarrow \lambda = 8{,}46720950103433e{-}06$$

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Los valores de $m\ y\ n$ salen de la dimensión de V, $m=6, n=5$. Con eso se tienen todos los parámetros para calcular el estimador $R_k = sure$ de la ecuación (7), y el resultado, para la primera iteración (k=1) es $sure = 360731,962283341$. Repetimos el proceso para k=2, 3,4,5, obteniendo como resultado el vector sure almacenando todos los resultados por los rangos del 1 al 5.

sure =[360731,962283341 193078,312043689 62803,0141227537
46081,6091254569 0,714531339969778]

Se toma el valor mínimo del vector sure, mediante la función *min* de matlab y su posición corresponde al rango mínimo. Si $\min(sure) = R_5 = 0,714531339969778$, entonces el rango mínimo es $k = 5$.

Ejemplo 3. Un ejemplo mucho más grande con una foto real

Figura 11. Pato

Fuente: Banco de imágenes de la Universidad de Berkeley

Estudio de la determinación del rango en las factorizaciones de matrices no negativas y su aplicación en la restauración de imágenes

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

$$V^{321\times 481} = \begin{bmatrix} 6 & 6 & 7 & 13 & 10 & 17 & 17 & 20 & 22 & \cdots & 60 \\ 6 & 7 & 9 & 9 & 17 & 15 & 20 & 25 & 39 & \cdots & 62 \\ 5 & 7 & 13 & 10 & 13 & 16 & 35 & 46 & 52 & \cdots & 63 \\ 9 & 6 & 10 & 12 & 16 & 41 & 53 & 55 & 54 & \cdots & 63 \\ 9 & 7 & 11 & 10 & 26 & 57 & 54 & 54 & 55 & \cdots & 63 \\ 6 & 12 & 14 & 9 & 39 & 57 & 50 & 55 & 55 & \cdots & 64 \\ 12 & 10 & 11 & 20 & 51 & 56 & 51 & 56 & 55 & \cdots & 64 \\ 10 & 10 & 19 & 48 & 54 & 50 & 55 & 55 & 55 & \cdots & 64 \\ 12 & 7 & 24 & 52 & 49 & 51 & 53 & 55 & 55 & \cdots & 64 \\ 11 & 8 & 26 & 53 & 48 & 51 & 53 & 54 & 54 & \cdots & 63 \\ 11 & 10 & 29 & 52 & 47 & 51 & 53 & 52 & 53 & \cdots & 63 \\ & & & & & & & & & \cdots & 62 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \ddots & 62 \\ & & & & & & & & & \cdots & 63 \\ & & & & & & & & & & 61 \\ 61 & 60 & 60 & 60 & 61 & 62 & 62 & 61 & 62 & 62 & 60 \end{bmatrix}$$

$$sure\left[i=1,...,321\right] = [1367{,}66969326153 \quad 657{,}548040774312 \quad 465{,}626061679886$$
$$313{,}996600907573 \quad 252{,}649846403112 \quad 219{,}991650799182$$
$$191{,}459435037405 \quad 173{,}127934259542 \quad 159{,}105575320661$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$135{,}538631728415 \quad 136{,}594990996126 \quad 136{,}982168701286$$
$$137{,}399970024862 \quad 137{,}509615318649 \quad 137{,}721519106813]$$

El menor valor del vector sures es $\min\left(sure\right) = R_{66} = -257{,}307142775921$. Por lo tanto, el rango mínimo es $k = 66$.

Recepción: 15-06-2024
Aprobación: 01-09-2024

Capítulo 4. Experimentación y análisis de resultados

Se realizaron 36 simulaciones, 12 imágenes de colposcopía por cada método. Como se afirmó, para cada simulación se aplicó un rango de rangos k que está entre $50 \leq k \leq 200$. En cada simulación se eligió el rango correspondiente al mínimo de los tres métodos: SURE, MDL y MAD.

Para la inicialización de NMF, en los tres métodos se usa el algoritmo NNDSVD. Dentro del algoritmo NNDSVD se pudo encontrar en la literatura que, para el cálculo de los valores propios, usan la función de Matlab denominada *SVDS*, la misma que no basa su cálculo en el polinomio característico, más bien se basa en la bidiagonalización mediante rutinas de funciones compiladas del paquete LAPACK (Doxigen, 2019; Monahan, 2011). Matlab utiliza la rutina del paquete LAPACK a partir del año 2001, sus rutinas vienen compiladas en forma de funciones de código privado (MathWorks, 2021).

4.1. Experimentación 1. Estudio de la convergencia de algoritmo de factorización NMF y NMF ponderado

Se implementó en Matlab los tres métodos y se probó mediante un ejemplo pequeño. Luego al implementarlos con matrices más grandes que representan las colposcopías, se encuentra un problema común en los tres métodos. Este radica en la demora y lentitud de convergencia. Se considera que no es normal que esto ocurra; dos de los tres (SURE y MDL) presentan una convergencia lenta, y el otro método (MAD) presenta una convergencia extremadamente lenta. Esto hace pensar que hay algo dentro de los tres métodos, que los ralentiza. Para descubrir lo que

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

está ocurriendo es necesario estudiarlos internamente, corriendo paso a paso, controlando el número de iteraciones en cada bucle.

Del estudio experimental se descube que el causante de esta ralentización es el bucle que corresponde al cálculo de factorización H_{nueva} *y* W_{nueva} mediante NMF, cuyo algoritmo no detiene su ciclo repetitivo, o bien, hasta alcanzar el óptimo, o bien, hasta alcanzar el umbral de parada (número máximo de iteraciones). Todo ello gira entorno al valor óptimo de la función objetivo y la condición del umbral de parada. Por tanto, se inicia el estudio realizando 400 repeticiones de factorización NMF para distintos valores de la función objetivo (F.O.) y distintos umbrales de parada; los resultados se presentan la tabla 2.

Tabla 2. Convergencia de NMF en función del valor óptimo y el número máximo de iteraciones.

NMF k=50 (Rango)	**F.O. = 0.02; umbral de parada= 1 000 iteraciones**		**F.O. = 0.02; umbral de parada= 5 000 iteraciones**		**F.O. = 0.05; umbral de parada= 1 000 iteraciones**		**F.O. = 0.08; umbral de parada= 1 000 iteraciones**	
r (repeticiones)	F.O. (valor alcanzado)	No. iteraciones	F.O. (valor alcanzado)	No. iteraciones	F.O. (valor alcanzado)	No. iteraciones	F.O. (valor alcanzado)	No. iteraciones
1	0.0339	1000	0.0327	5000	0.0498	50	0.0796	3
2	0.0338	2000	0.0326	10000	0.05	148	0.0796	6
3	0.0339	3000	0.0327	15000	0.05	197	0.0796	9
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
100	0.0337	100000	0.0327	500000	0.0498	4941	0.0796	300

Fuente: Elaboración propia

Al observar la tabla se puede deducir que el algoritmo del método NMF es extremadamente lento cerca del valor óptimo de la función objetivo, por ejemplo, con un valor de 0.02 o 0.05, el

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

algoritmo no converge a pesar de ocupar más de 5000 iteraciones, lo cual no habla bien del algoritmo. Este cuadro contiene 100 repeticiones, puesto que el método MAD debe repetir igual número de veces el NMF, para cumplir con el propósito de su algoritmo, como se vio anteriormente. Esto anticipa a pensar que, la convergencia de MAD va a ser extremadamente lenta y poco operativa, para un rango de valores del óptimo de NMF, por tanto, hay que solucionarlo con un adecuado valor del óptimo para NMF.

Al seguir con el análisis de la tabla se puede observar que, entre la segunda y la cuarta columna, se corrió el programa NMF con un valor óptimo mínimo deseado de 0.02 y con un umbral de parada ente 1000 a 5000 iteraciones; sin embargo, la función objetivo del NMF no alcanza el valor óptimo deseado (se probó con 10 mil iteraciones como umbral, y tampoco converge), se queda estancado en un valor mínimo de 0.0327. El total de iteraciones, luego de las 100 repeticiones, oscilan entre 100 mil y 500 mil iteraciones. Con esto se confirma la sospecha que la causa de la lentitud de los algoritmos es NMF.

Por otro lado, se observa que, entre la quinta y la sexta columna se fija un valor óptimo de 0.05, y un umbral de 1000 iteraciones. En este caso la función alcanza el óptimo y converge con valores mínimos de 0.0498, necesitando para esta convergencia, alrededor de 50 a 100 iteraciones por repetición, lo cual ya es aceptable. Finalmente, se puede visualizar que, entre la 7ª y la 8ª columna, se fija un óptimo de 0.08, la función objetivo de NMF converge al mínimo con apenas 3 iteraciones por repetición, que es mucho mejor.

Para continuar con el estudio de la convergencia de NMF, se procede a realizar una experimentación variando el rango con valores menores que 50. Se observa que la lentitud de

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

convergencia se mantiene y, en muchos casos, para valores muy pequeños del rango el método NMF no converge. Los resultados se considera no publicarlos, pues no están en el intervalo de interés de este estudio. Por el contrario, se experimenta variando el valor del rango, subiéndolo gradualmente, para valores mayores a 50, con un valor óptimo de 0.08 para NMF y una condición de parada 1000 iteraciones como máximo. Esto es aplicado en los tres métodos en igualdad de condiciones. Podemos ver algunos resultados en la tabla 3, para los rangos de 80 y 150.

Tabla 3. Estudio de la convergencia de NMF en los tres métodos, junto con las variables de número de iteraciones, tiempo de corrida, calidad del producto WH, y calidad de restauración PWH.

F.O. NMF = 0.08; Umbral de parada NMF= 1000 iteraciones					
Rango=80	**F.O. NMF**	**iteraciones**	**tiempo (min)**	**calidad ssim WH**	**calidad ssim PWH**
MAD	0.0705	404	0.10	0.5412	0.5371
MDL	0.0706	9	0.03	0.5448	0.5386
SURE	0.0705	9	0.03	0.5412	0.5371
Rango=150	F.O. NMF	iteraciones	tiempo (min)	calidad ssim WH	calidad ssim PWH
MAD	0.0705	404	0.140	0.5412	0.5371
MDL	0.71	9	0.064	0.5553	0.5487
SURE	0.0705	9	0.030	0.5412	0.5371

Fuente: Elaboración propia

En esta tabla se puede observar que el algoritmo NMF es convergente bajo estos parámetros. Esto es lo que se buscó en este momento del estudio, que su convergencia no sea un problema y por ende no influya en el desenvolvimiento de los algoritmos de los tres métodos, al ser parte de estos. Por lo tanto, ahora se puede realizar en los tres métodos, la medición y estudio del comportamiento de las variables, el número de iteraciones y el tiempo de ejecución, así como la

Recepción: 15-06-2024
Aprobación: 01-09-2024

calidad. En la misma tabla anterior, se puede visualizar que el MAD ocupa más iteraciones que SURE y MDL, y esto es lógico pensar que iba a ser así, puesto que su algoritmo repite 100 la ejecución de la factorización de NMF, frente a los otros métodos que solamente lo hacen unas cuantas veces. Vale aclarar que, las mismas condiciones de valor óptimo de 0.08 y umbral de parada 1000 iteraciones, serán aplicados al algoritmo MNF ponderado, puesto que su algoritmo es el mismo NMF aumentado un producto por una matriz binaria, denominada pesos.

4.2. Experimentación 2. Estudio del rango con los tres métodos en función del valor óptimo de NMF

Con los tres softwares implementados correctamente se realizan varias corridas de los programas de MDL, SURE y MAD, variando los valores óptimos de la función objetivo (FO) de NMF y manteniendo fijo el umbral de parada en 1000 iteraciones. En dos de los métodos que permiten, se varían también ciertos parámetros propios del método sin cambiar su estructura y modelo de convergencia. El objetivo de estas corridas es medir la principal variable que es el rango y adicional a esta, el costo de máquina y de tiempo, para poder juzgar su competitividad al final. En estas corridas, se mantiene un rango de rangos fijo, en el intervalo $50 \leq k \leq 200$ para los tres métodos. Los resultados los puede visualizar en la tabla 4.

Tabla 4. Corrida de MDL variando F.O. de NMF y el parámetro de corte de la Partición de MDL

ESTUDIO DEL RANGO CON EL MÉTODO MDL					
colposcopía	**Partición**	**FO NMF**	**Rango**	iteraciones	Tiempo (min)
colpos1	corte=0.02	0.050	200	6892	3.10
colpos1	corte=0.01	0.075	200	902	2.18
colpos1	corte=0.02	0.075	200	902	2.31
colpos1	corte=0.005	0.080	200	596	2.20
colpos1	corte=0.02	0.080	200	596	3.75

Fuente: Elaboración propia

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Estudio de rango que proporciona MDL variando el óptimo de la F.O. de NMF y el parámetro de corte de la Partición de MDL

En la tabla 4 se puede observar que el valor del rango no varía, y toma siempre el mismo valor 200. Esto sucede a pesar de transformar dos variables: (i) el valor óptimo de NMF con valores de: 0.05, 0.075, 0.08; (ii) el valor del parámetro corte en MDL que divide a la matriz original de datos de la colposcopía en "ceros" (los que están por debajo del corte) y "no ceros" (los que están por encima de este). Lo único que se diferencia es el elevado número de iteraciones para un corte de 0.02 de MDL y un óptimo deseado de 0.05 para NMF, frente a las otras variantes. En general, el número de iteraciones y el tiempo de ejecución son pequeños, pero el valor del rango no es un mínimo esperado, sino el valor máximo posible del intervalo. Los resultados los puede visualizar en la tabla 5.

Tabla 5. Corrida de SURE variando F.O. de NMF

ESTUDIO DEL RANGO CON EL MÉTODO SURE				
colpos	**FO NMF**	**Rango**	iter	t min
colpos1	0.050	175	6894	3.27
colpos1	0.075	153	903	2.31
colpos1	0.080	123	597	2.24
colpos2	0.050	114	3685	2.51
colpos2_2ª	0.050	114	3685	3.89
colpos2	0.075	104	884	5.29
colpos2_2ª	0.075	104	884	2.39
colpos2	0.080	168	668	2.08

Fuente: Elaboración propia

Estudio de rango que proporciona SURE variando el óptimo de la F.O. de NMF

En la tabla 5 se observa que el valor del rango cambia al variar el valor del óptimo de NMF, y al hacer dos corridas en la misma colposcopía, bota el mismo valor del rango, y el mismo número

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

de iteraciones, con un tiempo parecido. Esto muestra una consistencia de resultados del método SURE. En general, se puede rescatar de estas experimentaciones que, el rango mínimo en la colposcopía 1 es 123 con un óptimo de 0.08 en NMF; en cambio, el rango mínimo en la colposcopía 2 es de 104, esto con un óptimo de 0.075 en NMF. Los tiempos y número de iteraciones son menores con el valor óptimo de 0.075 y 0.08 para NMF, que con el valor 0.05. Los resultados se pueden visualizar en la tabla 6.

Tabla 6. Cálculo del Rango con MAD al variar el porcentaje de retención de los datos.

RANGO MAD con 0.08 FO NMF			
porcentaje de retención	**repeticiones=75**	**repeticiones=100**	**repeticiones=125**
5%	156	138	164
10%	186	149	108
15%	142	163	166
20%	50	192	164
		Rango min=	50

Fuente: Elaboración propia

Estudio de rango que proporciona MAD fijando el óptimo de la F.O. de NMF en 0.08 y variando dos parámetros de MAD: porcentaje de retención y número de repeticiones

El método MAD para los valores menores de 0.08 es muy lento, por ende, el costo de máquina es alto. Luego, no se considera necesario estudiar el valor del rango proporcionado por MAD variando la función objetivo de NMF, sino más bien, variando el número de repeticiones y el porcentaje de retención propios de MAD, con el objetivo de ver si se logra bajar el costo de máquina, sin perjudicar el resultado del rango. Para esto se fija, el valor de 0.08 como óptimo deseado para NMF y un umbral de 1000 iteraciones como máximo en NMF.

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

En la tabla se puede observar que el mínimo rango se obtiene para un valor de retención de datos del 20% y un número de repeticiones de 75. En el modelo original usan una retención del 10% de los datos y un número de repeticiones de 100. En la práctica, el porcentaje de retención es relativo, puesto que dependerá de la cantidad de pixeles brillosos que sean retirados de la colposcopía; en cambio, el número de repeticiones sí es importante porque de este dato depende el costo de máquina y el tiempo de ejecución del modelo MAD.

No es concluyente, el rango mínimo 50 que proporciona la tabla, puesto que, debido a la naturaleza probabilística de la retención de los datos, al correrlo nuevamente, dará otro resultado diferente que puede ser mayor o menor. De esta tabla se puede significar que los valores del rango obtenidos en general están por debajo del máximo que es 200 y ninguno lo alcanza, en el intervalo de estudio $50 \leq k \leq 200$

Estudio del cálculo del costo máquina con MAD al variar el porcentaje de retención de los datos

En la tabla 7, se puede apreciar que, el número de iteraciones es directamente proporcional al número de repeticiones. Esto es lógico, pues cada repetición representa un ciclo completo de factorización NMF, esto multiplicado por 75, 100, o 125 aumenta las operaciones al final. En la experimentación final, se deja fijo el número de repeticiones en 100 y la retención del 10%, tal cual el trabajo original. Para calcular los resultados que serán comparados y evaluados en los tres métodos, se deja fijo el valor óptimo de NMF en 0.08, con un umbral de 1000 iteraciones como máximo (Figura 12).

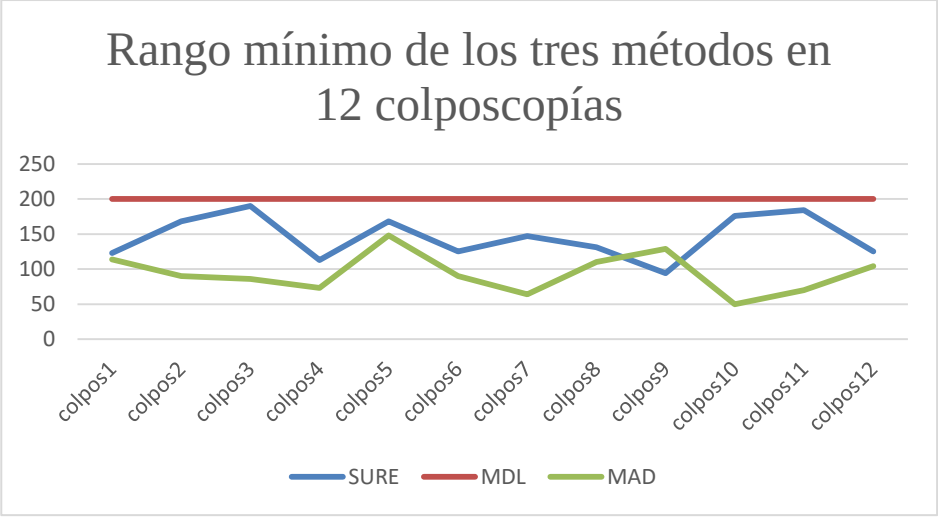
Recepción: 15-06-2024
Aprobación: 01-09-2024

Tabla 7. Cálculo del costo máquina con MAD al variar el porcentaje de retención de los datos.

NÚMERO DE ITERACIONES-MAD con 0.08 FO NMF			
porcentaje de retención	**repeticiones=75**	**repeticiones=100**	**repeticiones=125**
5%	44373	59161	73941
10%	44403	59210	74016
15%	44453	59264	74085
20%	44493	59317	74156

Fuente: Elaboración propia

Figura 12. Gráfica comparativa de los resultados de los rangos mínimos con los tres métodos



Fuente: Elaboración propia

4.3. Experimentación 3. Resultados del estudio del rango en los tres métodos

Los resultados se recogen en la tabla 8. Como se puede apreciar en la Figura 13, el método MDL proporciona en promedio como rango mínimo 200; SURE proporciona en promedio un rango mínimo de 145, se observa que el método SURE proporciona como rango mínimo un valor cercano a la cota superior de rangos de experimentación que es 200; y el MDL proporciona como

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

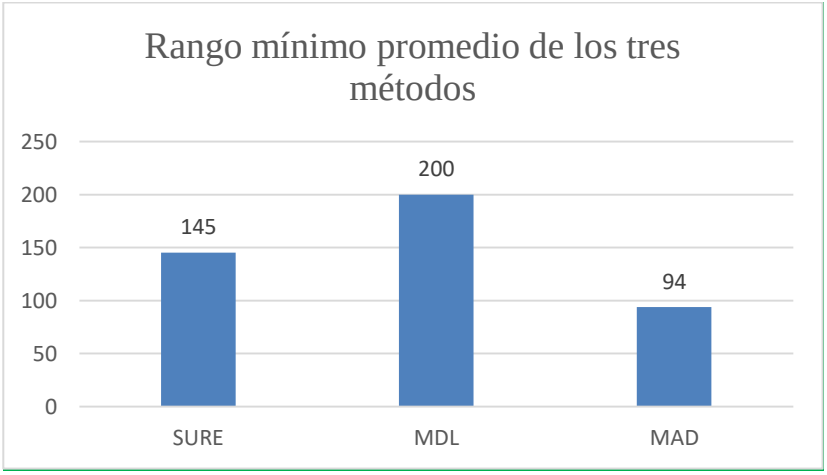
rango mínimo el mismo valor de 200; en cambio, el método MAD proporciona en promedio un rango mínimo de 94 (una diferencia de 106 puntos con MDL y 51 puntos con SURE) (ver tabla 8).

Tabla 8. Resultados de los rangos mínimos con los tres métodos

$50 \leq k \leq 200$	**SURE**	**MDL**	**MAD**
colpos1	123	200	114
colpos2	168	200	90
colpos3	190	200	86
colpos4	113	200	73
colpos5	168	200	148
colpos6	125	200	90
colpos7	147	200	64
colpos8	131	200	110
colpos9	94	200	129
colpos10	176	200	50
colpos11	184	200	70
colpos12	125	200	104
Rango promedio	145	200	94

Fuente: Elaboración propia

Figura 13. Histograma con los promedios de los rangos mínimos con los tres métodos



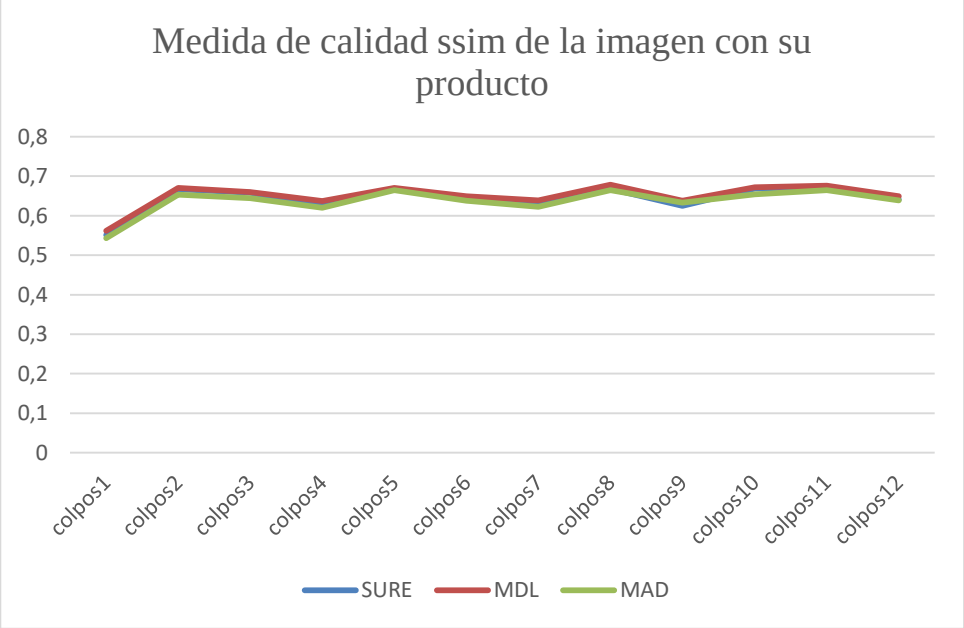
Fuente: Elaboración propia

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay
Recepción: 15-06-2024
Aprobación: 01-09-2024

4.3.1. Medición de la calidad SSIM al comparar V con el producto WH. Resultados y análisis

En la figura 15 se puede observar que, al comparar la imagen original V con su producto WH, el método MDL proporciona una medida de calidad SSIM en promedio del 65.01% y el método SURE del 64.32%; en cambio, el método MAD con un rango mucho menor, proporciona una medida de calidad SSIM en promedio de 63.68% (una diferencia pequeña de 1.33% con MAD y 0.64% con SURE).

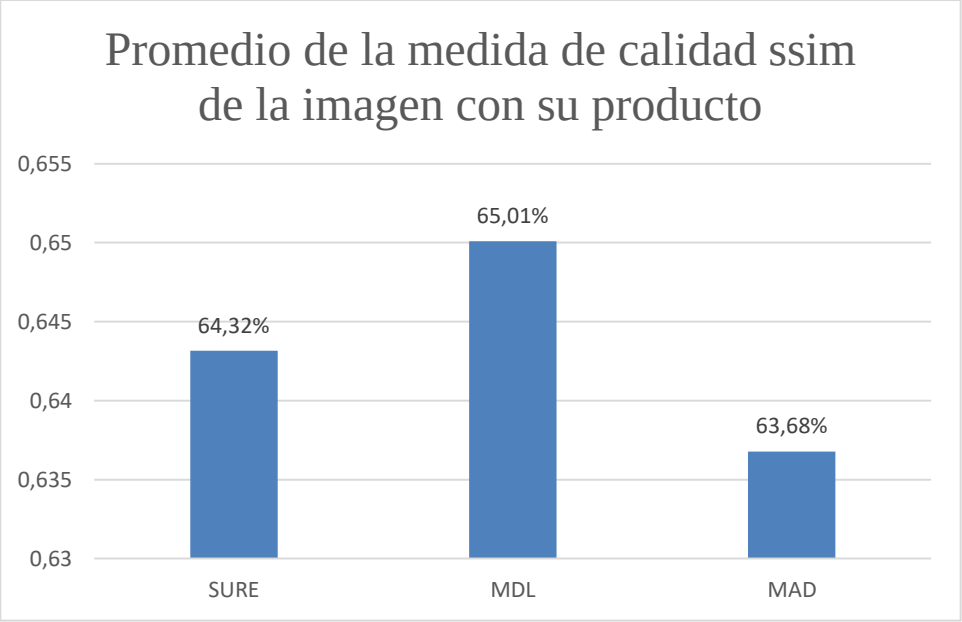
Figura 14. Gráfica comparativa de la medida de calidad SSIM, mide la aproximación $V \approx WH$



Fuente: Elaboración propia

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 15. Histograma con los promedios de calidad de aproximación SSIM de los tres métodos.



Fuente: Elaboración propia

4.3.2. Medición de la calidad SSIM al comparar V con su restauración (inpainting) $V \approx P\square WH$. Resultados y análisis

En la figura 17 se puede apreciar que, al comparar la imagen original V con su imagen restaurada $P\square WH$, el método MDL proporciona una medida de calidad SSIM en promedio del 64.15% y el método SURE del 63.46%; en cambio, el método MAD con un rango mucho menor proporciona una medida de calidad SSIM en promedio de 62.99% (una pequeña diferencia de 1.16% con MDL y 0.47% con SURE) (Tabla 9).

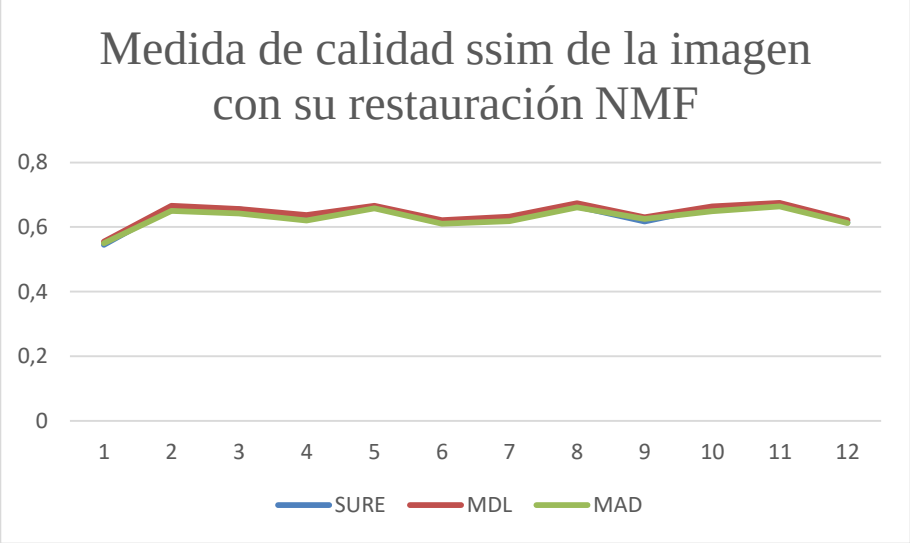
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Tabla 9. Medición de la calidad SSIM de los resultados al comparar V con su restauración $P\square WH$, es decir, $V \approx P\square WH$

$50 \le k \le 200$	SURE	MDL	MAD
colpos1	0.5448	0.5548	0.550
colpos2	0.6604	0.6656	0.6497
colpos3	0.655	0.656	0.6418
colpos4	0.6257	0.6369	0.6202
colpos5	0.6604	0.6656	0.658
colpos6	0.6139	0.6213	0.6102
colpos7	0.6253	0.6326	0.6179
colpos8	0.6639	0.6742	0.661
colpos9	0.6176	0.6304	0.6251
colpos10	0.6612	0.6642	0.6492
colpos11	0.6732	0.6751	0.6642
colpos12	0.6139	0.6213	0.6122
Promedio SSIM	0.63460833	0.6415	0.629941667

Fuente: Elaboración propia

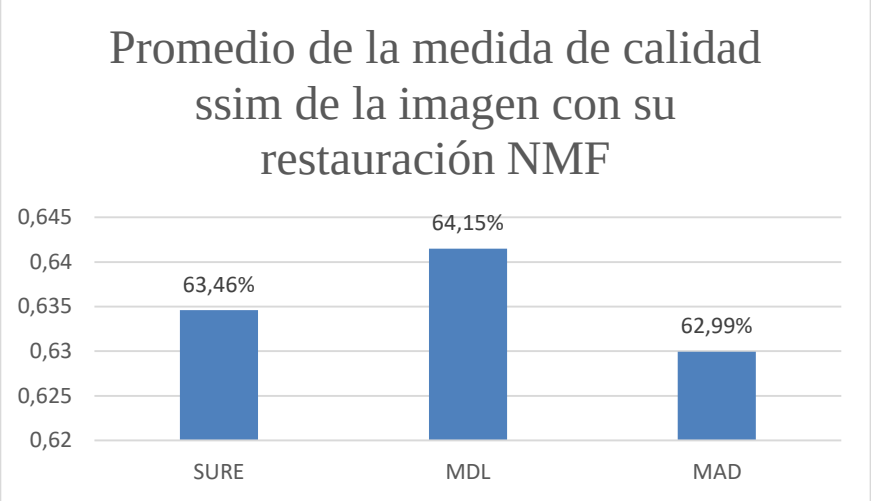
Figura 16. Gráfica comparativa de la medida de calidad SSIM, mide la aproximación $V \approx P\square W*H$



Fuente: Elaboración propia

Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 17. Histograma con los promedios de calidad de aproximación SSIM de los tres métodos



Fuente: Elaboración propia

4.3.3. Medición del tiempo de ejecución de los tres métodos. Resultados y análisis.

La medición del tiempo de ejecución de los códigos se hizo mediante la ayuda de la función *tic - toc* de Matlab. Para darle mayor consistencia a los tiempos de ejecución obtenidos, se procedió a hacer varias corridas en aquellas imágenes que presentaban tiempos elevados en comparación al promedio del resto de imágenes en cada uno de los métodos. Por ejemplo, en la colposcopía 1 con el método MAD en la primera simulación se obtuvo un tiempo de 74.48 minutos, en la segunda de 39.02 minutos, en la tercera de 36.84 minutos y la cuarta se obtiene un tiempo de 36.89 minutos; en este caso se eligió el promedio de los tres mejores tiempos que es 36.89 minutos. De esta manera, se controló los tiempos elevados por encima del promedio en los tres métodos.

En la figura 19 se puede apreciar que el método MDL ocupa un tiempo promedio de 3.23 minutos en su ejecución y el método SURE lo hace en 2.86 minutos; mientras que el método

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

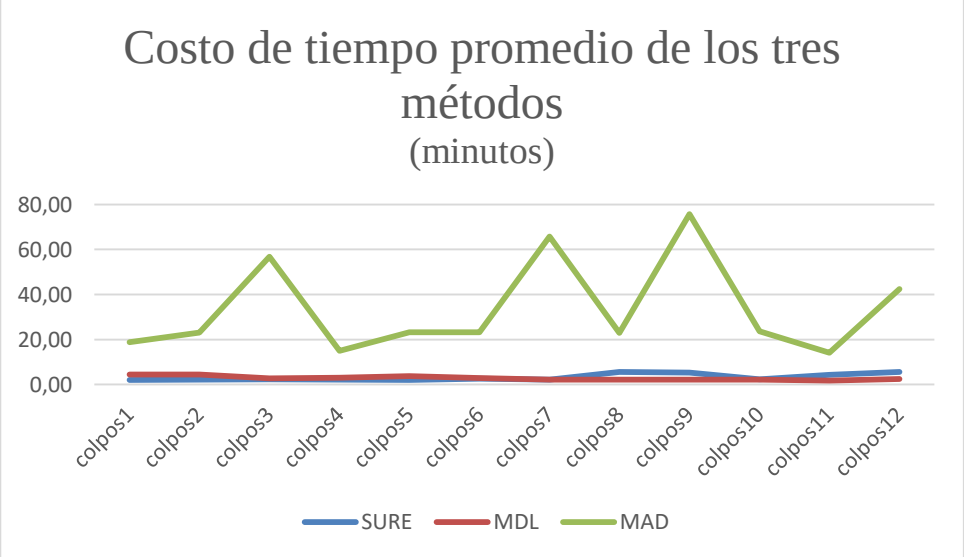
MAD ocupa un tiempo promedio de 33.72 minutos (una desventaja en el tiempo de 30.86 minutos con MDL y 30.49 con SURE) (Tabla 10).

Tabla 10. Resultados de los tiempos de ejecución en minutos de las 36 simulaciones

	SURE	MDL	MAD
colpos1	2.10	4.53	18.78
colpos2	2.19	4.41	23.11
colpos3	2.36	2.74	56.74
colpos4	2.14	2.97	14.99
colpos5	2.03	3.81	23.26
colpos6	2.54	2.90	23.26
colpos7	2.13	2.16	65.82
colpos8	5.62	2.16	22.93
colpos9	5.38	2.22	75.65
colpos10	2.36	2.17	23.66
colpos11	4.35	1.77	14.13
colpos12	5.54	2.51	42.38
Promedio minutos	3.23	2.86	33.72

Fuente: Elaboración propia

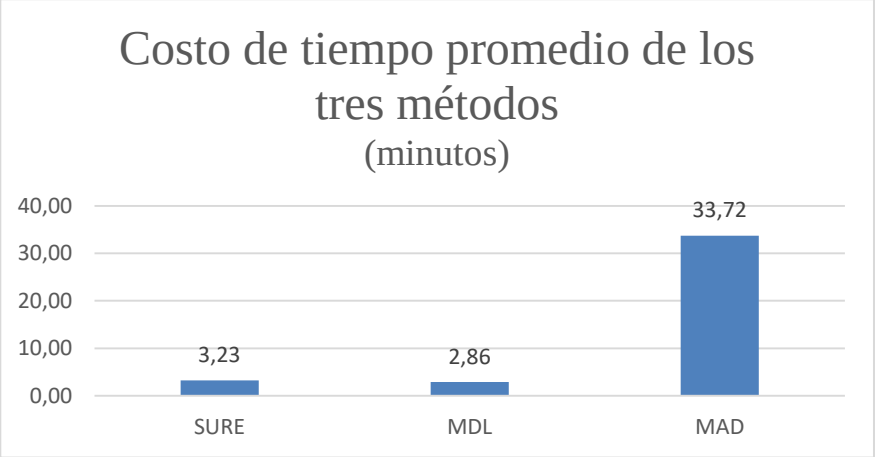
Figura 18. Gráfica comparativa de los tiempos de ejecución de los códigos en Matlab (minutos).



Fuente: Elaboración propia

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 19. Histograma con el promedio de los tiempos (min) de ejecución de los códigos en Matlab



Fuente: Elaboración propia

4.3.4. Medición del número de iteraciones de los tres métodos. Resultados y análisis

En la figura 21 se puede apreciar que el método MDL ocupa en promedio 1370 iteraciones, y el método SURE lo hace con 1218 iteraciones; mientras que el método MAD ocupa 120197 iteraciones (una desventaja de 119 mil iteraciones con MDL y SURE) (Tabla 11).

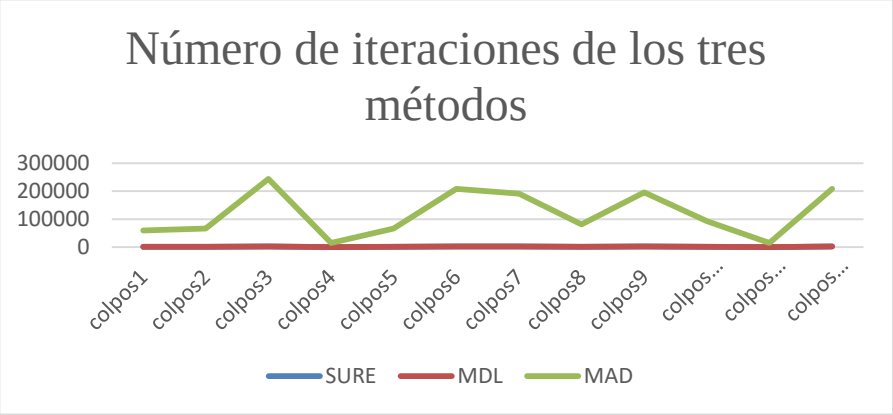
Tabla 11. Resultados del número de iteraciones de las 36 simulaciones

	SURE	MDL	MAD
colpos1	598	749	59211
colpos2	669	821	65933
colpos3	2476	2627	243606
colpos4	155	306	15104
colpos5	669	821	65937
colpos6	2111	2264	208428
colpos7	1940	2092	191230
colpos8	826	979	81437
colpos9	1980	2133	195802
colpos10	930	1081	92133
colpos11	155	306	15104
colpos12	2111	2264	208421
Promedio iteraciones	1218	1370	120196

Fuente: Elaboración propia

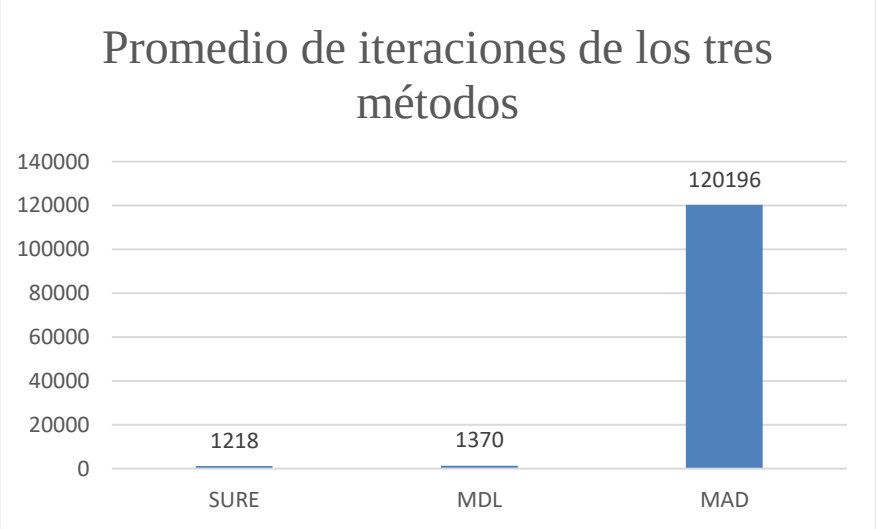
9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Figura 20. Gráfica comparativa del número de iteraciones de los códigos en Matlab (minutos).



Fuente: Elaboración propia

Figura 21. Histograma con el promedio del número de iteraciones de ejecución de los tres métodos.



Fuente: Elaboración propia

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Epílogo

Se implementó un programa en Matlab basado en la técnica del *umbral* para la detección, y posterior eliminación de los pixeles de brillo, que es el punto de entrada para la restauración de las imágenes de colposcopía. Se implementó, además, un software en Matlab de cada uno de los tres métodos de selección del rango, para su estudio y experimentación.

De la experimentación realizada se seleccionó como mejor método al método MAD, pues tiene la ventaja de que proporciona el mejor rango mínimo entre los tres métodos (siendo en promedio 106 veces menor a MDL y 56 veces menor a SURE) y la desventaja de que su costo de tiempo de ejecución y número de iteraciones es mucho mayor, frente a los otros dos métodos MDL y SURE (MAD se demora alrededor de 30 minutos más que los otros; MDL y SURE terminan su ejecución promedio con alrededor de1400 iteraciones, mientras que MAD lo hace en promedio con 121 mil iteraciones).

El método MAD utilizando una cantidad muchísimo menor de datos de la imagen original logra una restauración con una calidad SSIM, apenas menor al 2%. Esto al comparar, tanto la *imagen producto*, como la *imagen restaurada*. El método MAD se lo puede mejorar para que sea menos costoso computacionalmente, y se vuelva muy competitivo frente a los otros dos métodos SURE y MDL, ajustando el parámetro “número de repeticiones” de imputación. Este ajuste (creciente o decreciente) dependerá del tipo de datos que se tenga.

En el método MDL se sugiere probar con otras distribuciones de probabilidad (en vez de Gauss y Gamma), para ver si se obtienen mejores resultados en este tipo de imágenes. Por su parte, en el método SURE se sugiere utilizar la varianza de la matriz original en lugar de la varianza utilizada

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

por el autor del método (específica para el procesamiento de señales), para ver si se obtienen mejores resultados en este tipo de imágenes.

Recepción: 15-06-2024
Aprobación: 01-09-2024

Referencias

Baguer, M. L. (2018). *Nonnegative matrix factorizations: Ideas and applications. Anchor Institutions Advancing Local and Global Sustainable Community Development Through Teaching, Scholarship, and Research, 2017-2018*. Conference Proceedings Rutgers University–Camden, 78-88.

Bianco, M. J. (s. f.). *Introducción a la Probabilidad y a la Estadística*.

Boutsidis, C., & Gallopoulos, E. (2008). SVD based initialization: A head starts for nonnegative matrix factorization. *Pattern Recognition, 41*(4), 1350-1362. https://doi.org/10.1016/j.patcog.2007.09.010

Bovik, A., Wang, Z., & Sheikh, H. (2005). Structural Similarity Based Image Quality Assessment. In H. Wu & K. Rao (eds.), *Digital Video Image Quality and Perceptual Coding* (pp. 225-241). CRC Press. https://doi.org/10.1201/9781420027822.ch7

Casalino, G., Castiello, C., Del Buono, N., & Mencar, C. (2017). Intelligent Twitter Data Analysis Based on Nonnegative Matrix Factorizations. In O. Gervasi, B. Murgante, S. Misra, G. Borruso, C. M. Torre, A. M. A. C. Rocha, D. Taniar, B. O. Apduhan, E. Stankova, & A. Cuzzocrea (eds.), *Computational Science and Its Applications – ICCSA 2017* (pp. 188-202). Springer International Publishing. https://doi.org/10.1007/978-3-319-62392-4_14

Chih-Jen Lin. (2007). On the Convergence of Multiplicative Update Algorithms for Nonnegative Matrix Factorization. *IEEE Transactions on Neural Networks, 18*(6), 1589-1596. https://doi.org/10.1109/TNN.2007.895831

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Doxigen. (2019, noviembre 21). *LAPACK 3.9.0*. http://www.netlib.org/lapack/explore-html/d1/d7e/group__double_g_esing_ga84fdf22a62b12ff364621e4713ce02f2.html

Efron, P. & Morris, B. (1977). *La paradoja de Stein en Estadística*.

Meslouhi, O., Kardouchi, M., Allali, H., Gadi, T., & Benkaddour, Y. A. (2011). Automatic detection and inpainting of specular reflections for colposcopic images. *Central European Journal of Computer Science, 1*(3), 341-353. https://doi.org/10.2478/s13537-011-0020-2

Gómez, D. (2018). *Eliminación de zonas especulares en imágenes de colposcoía utilizando Factorizaciones Matriciales No-negativas* [Tesis de Diploma]. Universidad de La Habana.

Herazo, C. A. (2014). *Probabilidad y estadística: Una introducción* (Primera edición). Editorial Universitaria.

Kanagal, B., & Sindhwani, V. (s. f.-a). *Rank Selection in Low-rank Matrix Approximations: A Study of Cross-Validation for NMFs*.

Lee, D. D., & Seung, H. S. (1999a). Learning the parts of objects by non-negative matrix factorization. *Nature, 401*(6755), 788-791. https://doi.org/10.1038/44565

MathWorks. (2021). *MathWorks. Accelerating the pace of engineering and science*. https://la.mathworks.com/help/matlab/math/lapack-in-matlab.html

Monahan, J. F. (2011). *Numerical Methods of Statistics (2nd ed.)*. Cambridge University Press. https://doi.org/10.1017/CBO9780511977176

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Muñoz, F. (2014). Distribuciones Poisson y Gamma: Una Discreta y Continua Relación. *Prospectiva, 12*(1), 99-109. https://doi.org/10.15665/rp.v12i1.156

Muzzarelli, L., Weis, S., Eickhoff, S. B., & Patil, K. R. (2019). Rank Selection in Non-negative Matrix Factorization: Systematic comparison and a new MAD metric. 2019 *International Joint Conference on Neural Networks (IJCNN),* 1-8. https://doi.org/10.1109/IJCNN.2019.8852146

O'Leary, D. P. (2009). *Scientific computing with case studies*. Society for Industrial and Applied Mathematics.

Palmer, A. (2015). *Eliminación de regiones especulares en imágenes colposcópicas de cuello de útero* [Tesis de Diploma]. Universidad de La Habana.

Rezaei, M., Boostani, R., & Rezaei, M. (2011). An Efficient Initialization Method for Nonnegative Matrix Factorization. *Journal of Applied Sciences, 11*(2), 354-359. https://doi.org/10.3923/jas.2011.354.359

Scheaffer & McClave, R. (1993). *Probabilidad y Estadística para Ingeniería*. Grupo Editorial Iberoamérica S. A. de C. V.

Squires, S., Prügel-Bennett, A., & Niranjan, M. (2017). Rank Selection in Nonnegative Matrix Factorization using Minimum Description Length. *Neural Computation, 29*(8), 2164-2176. https://doi.org/10.1162/neco_a_00980

Stein, C. (1981). *Estimation of the mean of a multivariate normal distribution*.

Szeliski, R. (2011). *Algorithms and Applications*. Computer Vision.

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Tibshirani, R. (2015). *Stein's Unbiased Risk Estimate*.

Ulfarsson, M. O., & Solo, V. (2013). Tuning parameter selection for nonnegative matrix factorization. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 6590-6594. https://doi.org/10.1109/ICASSP.2013.6638936

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, *13*(4), 600-612. https://doi.org/10.1109/TIP.2003.819861

Book Citation Index
Web of Sciece/Core Collection

Socrates Emilio Haro Guanga
Edisson Lascano Mora
Verónica Coronel Pérez
Leonardo Paladines Zurita
Víctor Hugo Villón Pincay

9 780311 000722
Recepción: 15-06-2024
Aprobación: 01-09-2024

Estudio de la determinación del rango en las factorizaciones de matrices no negativas y su aplicación en la restauración de imágenes

Study of rank determination in factorizations of nonnegative matrixes and its application in image restoring

Sobre la presente edición:
Primera edición
Esta obra ha sido evaluada por pares académicos a doble ciegos
Lectores/Pares académicos/Revisores: 0065 & 0100
Editorial Tecnocientífica Americana
Domicilio legal: calle 613sw 15th, en Amarillo, Texas. **ZIP**: 79104, EEUU
Teléfono: 7867769991
Fecha de publicación: 23 septiembre de 2024
Código BIC: PBW **Código EAN: 9780311000722**
Código UPC: 978031100072 **ISBN: 978-0-3110-0072-2**
La Editorial Tecnocientífica Americana se encuentra indizada en, referenciada en o tiene convenios con, entre otras, las siguientes bases de datos: