UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Universidad de la República
Facultad de Ingeniería

Tesis de maestría
en ingeniería matemática

Análisis de componentes principales para datos genómicos en presencia de datos faltantes

Gerardo Martínez

Tutores:
María Inés Fariello e Ignacio Ramírez

20 de diciembre de 2021

Agradecimientos

La versión de este trabajo a la que el tribunal accedió no contenía una sección de agradecimientos. Y esto no era por falta de gente a quien agradecer. La fatiga sobre el final de este proceso de dos años que fue esta maestría me impidió tener un momento de inspiración para agradecer como hubiese gustado. Ahora, luego de corregir el trabajo y observándolo a la distancia, hallo por fin la inspiración para hacerlo.

Hace unos años, cuando escribía mi monografía de grado, decía que no me imaginaba estudiando otra cosa que no fuera genética de poblaciones. Suscribo a mis palabras del pasado. Decía en aquel entonces que parte de la culpa la tenía Maine por haberme metido en estos embrollos. También suscribo a estas otras palabras. Así como fue realmente valiosa la supervisión de Maine en ese entonces también lo fue en esta instancia durante el transcurso de esta maestría. Además, tuvo la capacidad de bancarme la cabeza en varias oportunidades y darme consejos en instancias donde otros tutores quizá no lo habrían hecho. Gracias, Maine.

Este trabajo tendría una forma muy distinta sin la supervisión de Nacho. Fueron muy preciados sus consejos sobre formas de trabajo y sobre cómo programar. Me consta que no siempre tomé estos consejos inicialmente pero el tiempo me hizo entender su valor. No puedo asegurarles que haya aprendido a programar en estos años pero estoy seguro de que si ahora repito algunas de las cosas que hacía antes, voy a tener la voz de Nacho en la cabeza, desaprobando lo que ve. Por otra parte, fueron también muy valiosos sus consejos sobre el mundo académico y la vida después de esta maestría. Gracias, Nacho.

Es interesante cómo los familiares pueden apoyarnos sin saber bien lo que uno hace; lo hacen con una confianza ciega de que estamos haciendo las cosas bien. Tampoco supe yo muy bien cómo explicarles y dudo que este trabajo ayude a esclarecer algo. Lo que sí es claro es que el hecho de que yo esté escribiendo este trabajo es para ustedes un motivo de orgullo. Espero entonces que esto cuente como "estar haciendo las cosas bien". Gracias a todos: padres, hermanos, tíos, primos, etc.

Escribía también hace unos años, en mi monografía de grado, que no me atrevía a dar una lista de amigos a quienes agradecer. Nuevamente, vuelvo a pecar de cobarde y me voy a negar a listarlos. Pero todos ellos saben el papel que tienen en mi vida por lo que no hace falta aclarar tampoco. La persona en la que me he convertido tiene parte de todos ustedes. Gracias, chikes.

Pensé mucho en si debería agradecerle a esta última persona. Pensaba en qué sentido tendría si quizá nunca lo leería. Pero por su importancia para mí y, en consecuencia, para el desarrollo de esta maestría, no puedo evitar hacerlo. Gracias, Dieguito.

Índice general

Capítulo 1

Introducción

La estructuración en subpoblaciones es un proceso usual que sufren las poblaciones naturales [1]. Las razones que llevan a que las poblaciones sufran subdivisiones son diversas: en algunos casos, son las propias especies las que se estructuran naturalmente en clanes o colonias; en otros casos, las características de los hábitats y/o las distancias geográficas aíslan a algunos individuos de otros [2]. Independientemente de la razón que la provoque, la existencia de subpoblaciones provoca una *diferenciación genética* entre los individuos que las componen; es decir, las frecuencias de algunos alelos serán distintas entre poblaciones.

En el estudio de poblaciones humanas la asignación en poblaciones es frecuentemente subjetiva y basada en factores culturales, fenotípicos o geográficos [3]. Es relevante, por tanto, la pregunta de si esta asignación es consistente con la asignación a la que se llegaría si uno estudiara la diversidad genética de los individuos. Esta pregunta podría ser analizada esencialmente de dos formas: en la primera, un investigador podría partir de una muestra de individuos que pertenecen a una población y estudiar la existencia de subpoblaciones; en la segunda, a partir de una muestra de invididuos y un conjunto predefinido de subpoblaciones, el investigado podría intentar asignar a los individuos a cada una de estas últimas. Una técnica matemática fundamental que permite vincular ambos enfoques es el *análisis de componentes principales* (PCA).

La aplicación del PCA en genética es una técnica conocida desde la década del 1970, aunque el objetivo de su utilización y su forma de aplicación era diferente al actual (por ejemplo, en [4]). Basados en el enfoque estadístico de PCA (el desarrollado por [5]) el objetivo de su utilización era la construcción de variables aleatorias de resumen (que Cavalli-Sforza denominaba *mapas sintéticos*) como combinaciones lineales de otras variables de interés en la genética. Con el advenimiento de los datos genómicos en la primera década del 2000, el PCA resultó una herramienta particularmente útil para responder preguntas sobre estructura poblacional; podemos destacar dos trabajos que son piedras angulares de este enfoque: [6] y [7]. La versatilidad y potencia de esta técnica fue analizada en [8] al mostrar que el utilizar datos genómicos de individuos y proyectar estos datos sobre las dos primeras componentes principales uno obtiene información sobre los patrones geográficos de las poblaciones estudiadas.

Es en el estudio de la estructura poblacional a través de datos genómicos que surge el problema clave de este trabajo: el de realizar PCA en presencia de datos faltantes. Podemos encontrar dos instancias en donde este problema surge de forma natural.

La primera instancia es en el análisis de ADN ancestral extraído de fósiles. La capa-

cidad de obtención de ADN de buena calidad está limitada por los costos y por el estado de la muestra de donde se extrae [9]. El resultado es, muchas veces, una muestra que presenta incertidumbre: ciertas entradas de las observaciones son desconocidas a causa de la ruptura de la molécula del ADN. En este caso podemos pensar que la pérdida de datos es un proceso al azar y que el conjunto de valores faltantes es aleatorio.

Distinta es la segunda instancia en donde podemos encontrar datos faltantes en genética de poblaciones: en esta encontraremos que el conjunto de datos faltantes es altamente estructurado. Nos referimos a la propuesta que se hace en [10] para analizar la estructura poblacional de individuos con mezcla. Moreno-Estrada et al. proponen que para estudiar la estructura poblacional de individuos mezclados se pueden *enmascarar* las regiones que no son de interés y realizar un PCA solo con aquellas que sí lo son. A modo de ejemplo, en ese artículo se realizó un PCA sobre genomas de individuos latinoamericanos, utilizando solo aquellas regiones que fueron previamente clasificadas como *nativas* y enmascarando aquellas regiones clasificadas como *europeas* y *africanas*. El resultado de aplicar esta máscara es la aparición de datos faltantes por bloques en lugar de datos faltantes completamente aleatorios (a pesar de que la ubicación de los bloques sí es al azar).

Una técnica usual para resolver el problema de datos faltantes es imputar las entradas desconocidas. Este enfoque es poco satisfactorio en genética de poblaciones, lo que nos motiva a realizar un análisis más profundo de este problema. El objetivo de este trabajo será el de estudiar y comparar diversas técnicas para realizar un PCA en el contexto de datos faltantes. Estaremos interesados en técnicas que recuperen la estructura de poblaciones *puras* y mezcladas de forma satisfactoria (es decir, cercana a la que obtendríamos sin datos faltantes) pero también nos interesará el problema de la escalabilidad y la velocidad de los algoritmos.

El capítulo 2 introducirá el problema de hallar componentes principales desde un punto de vista geométrico: este será el problema básico que atravesará todo el trabajo. Estudiaremos también cómo se vincula esto a la genética de poblaciones y mostraremos por qué debemos ser cuidadosos al trabajar con datos faltantes en genómica.

En el capítulo 3 realizaremos un estudio del PCA como un problema de factorización de matrices. Presentaremos un algoritmo de minimización alternada y un algoritmo basado en descenso por gradiente. Analizaremos la convergencia teórica de estos algoritmos y su implementación computacional.

La segunda técnica, estudiada en el capítulo 4, está basada en un supuesto sobre la distribución probabilística de los datos. La existencia de este supuesto nos permite realizar el llamado *análisis de componentes principales probabilístico*. Esto nos llevará a implementar un método iterativo basado en el algoritmo EM (*Expectation-Maximization*). El enfoque del capítulo 5 es ligeramente distinto al de los capítulos previos. En este nos apartaremos del estudio de las componentes principales y analizaremos una técnica para completar una matriz con entradas desconocidas. Para poder construir un algoritmo iterativo que resuelva este problema introduciremos algunos conceptos de funciones convexas y de problemas proximales.

En el capítulo 6 realizaremos un estudio comparativo de las implementaciones de estos algoritmos. Esto nos llevará a analizar distintos casos de prueba con el objetivo de entender cuál o cuáles de las técnicas previamente introducidas pueden ser aplicadas para datos genómicos con entradas desconocidas.

El código construido durante el desarrollo de este trabajo puede consultarse en este repositorio de GitHub.

Capítulo 2

Conceptos previos

El análisis de componentes principales

Supongamos que contamos con n puntos de $\mathbb{R}^d$, $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, y deseamos hallar el subespacio afín $\mathcal{S} \subset \mathbb{R}^d$ de dimensión p, con $p < d$ que los aproxima de la mejor manera en un sentido posteriormente a definir. En una primera instancia, consideremos que la aproximación es exacta, es decir, que cada $\mathbf{x}_j$ puede ser escrito como

$$\mathbf{x}_j = \boldsymbol{\mu} + \mathbf{U}\mathbf{y}_j, \quad j = 1, 2, \ldots, n, \tag{2.0.1}$$

donde $\boldsymbol{\mu} \in \mathcal{S}$ es un punto de este subespacio, $\mathbf{U} \in \mathbb{R}^{d\times p}$ es una matriz cuyas columnas forman una base de $\mathcal{S}$ e $\mathbf{y}_j \in \mathbb{R}^p$ es el vector de coordenadas de $\mathbf{x}_j$ en este subespacio.

Observemos que la representación de los puntos $\mathbf{x}_j$ expresada en (2.0.1) no es única. En efecto, consideremos un $\mathbf{y}_0 \in \mathbb{R}^d$ arbitrario. Podemos representar al punto $\mathbf{x}_j$ como

$$\mathbf{x}_j = (\boldsymbol{\mu} + \mathbf{U}\mathbf{y}_0) + \mathbf{U}(\mathbf{y}_j - \mathbf{y}_0).$$

Podemos encontrar así infinitos espacios afines $\mathcal{S}$ que resuelvan el problema. Una forma de resolver esta *ambigüedad traslacional* es eligiendo el subespacio afín $\mathcal{S}$ que cumpla la restricción de que los vectores $\mathbf{y}_1, \ldots, \mathbf{y}_n$ satisfagan

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{y}_j = \mathbf{0}.$$

Esta elección es por el momento arbitraria, pero será de utilidad para resolver el problema de optimización que especificaremos próximamente.

Por otra parte, para cualquier matriz $\mathbf{A} \in \mathbb{R}^{p\times p}$ invertible podemos representar a cualquier $\mathbf{x}_j$ como

$$\mathbf{x}_j = \boldsymbol{\mu} + \mathbf{U}\mathbf{A}\mathbf{A}^{-1}\mathbf{y}_j \coloneqq \boldsymbol{\mu} + \tilde{\mathbf{U}}\tilde{\mathbf{y}}_j, \quad \text{con } \tilde{\mathbf{U}} = \mathbf{U}\mathbf{A} \text{ e } \tilde{\mathbf{y}}_j = \mathbf{y}_j. \tag{2.0.2}$$

Por lo tanto la matriz $\mathbf{U}$ y los vectores $\mathbf{y}_j$ no serían únicos. Una forma de resolver esta *ambigüedad de cambio de base* es requerir que la matriz $\mathbf{U}$ sea ortonogonal, es decir, $\mathbf{U}^\top\mathbf{U} = \mathbf{I}_p$. En efecto, si $\mathbf{A}$ es una matriz invertible de dimensión p, no podríamos construir una representación como en (2.0.2) puesto que $\tilde{\mathbf{U}}$ no sería necesariamente ortogonal. Esta restricción sólo resuelve la ambigüedad de cambio de base a menos de isometrías.

Más específicamente, si $\mathbf{U}$ es como en la ecuación (2.0.1) y $\mathbf{R} \in \mathbb{R}^{p\times p}$ es una matriz ortogonal entonces podemos escribir $\mathbf{x}_j = \boldsymbol{\mu} + \mathbf{U}\mathbf{R}\mathbf{R}^{-1}\mathbf{y}_j$. Luego la matriz $\tilde{\mathbf{U}} = \mathbf{U}\mathbf{R}$ es también una solución posible.

El modelo expresado en (2.0.1) asume que los puntos $\mathbf{x}_j$ pertenecen al espacio afín $\mathcal{S}$ de forma perfecta. Esto no será cierto en la práctica: los puntos estarán contaminados por ruido. Una forma posible de modelar esto es asumiendo que cada punto pertenece al subespacio $\mathcal{S}$ pero que está perturbado por un error aditivo, es decir,

$$\mathbf{x}_j = \boldsymbol{\mu} + \mathbf{U}\mathbf{y}_j + \varepsilon_j, \quad j = 1, 2, \ldots, n. \tag{2.0.3}$$

Estamos interesados en que la aproximación de los puntos $\mathbf{x}_j$ por el espacio afín $\mathcal{S}$ sea tal que se minimice el error de representación. Una función objetivo posible a minimizar es, entonces,

$$\sum_{j=1}^{n} \|\varepsilon_j\|_2^2 = \sum_{j=1}^{n} \|\mathbf{x}_j - \boldsymbol{\mu} - \mathbf{U}\mathbf{y}_j\|_2^2 .$$

Así, podemos plantear el problema de optimización que consiste en hallar el subespacio afín óptimo $\mathcal{S}^\star$ como

$$\text{(P)} \quad \begin{aligned} \underset{\boldsymbol{\mu},\mathbf{U},\{\mathbf{y}_j\}_{j=1}^n}{\text{minimizar}} \quad & \sum_{j=1}^{n} \|\mathbf{x}_j - \boldsymbol{\mu} - \mathbf{U}\mathbf{y}_j\|_2^2 \\ \text{sujeto a} \quad & \mathbf{U}^\top\mathbf{U} = \mathbf{I}_p \\ & \sum_{j=1}^{n} \mathbf{y}_j = \mathbf{0}_p \end{aligned} \quad . \tag{2.0.4}$$

Las columnas de $\mathbf{U}$ son las llamadas p *componentes principales*. Al resolver el problema (2.0.4) estaremos realizando un *análisis de componentes principales* (PCA, por sus siglas en inglés) sobre la matriz $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$.

La solución al problema (2.0.4) está estréchamente ligada a la descomposición en valores singulares (SVD) de la matriz $\mathbf{X}$: la matriz $\mathbf{U}$ es obtenida reteniendo los p vectores asociados a los p valores singulares más grandes de $\mathbf{X} - \hat{\boldsymbol{\mu}}_n$, donde $\hat{\boldsymbol{\mu}}_n$ es la media muestral. Los detalles y la prueba de este hecho pueden consultarse en el apéndice (C).

Datos genómicos

En secciones posteriores estaremos interesados en el vínculo que tiene PCA con un tipo de datos en particular: los datos genómicos. En la presente sección introduciremos al lector algunos conceptos y terminologías necesarias para entender el desarrollo del trabajo. Las ideas presentadas en esta sección están basadas en el libro [11].

La información necesaria para el desarrollo y mantenimiento de un organismo se encuentra almacenada dentro del núcleo de nuestras células en una molécula lineal llamada ácido desoxirribonucleico (ADN) [1]. Cada molécula de ADN está formada por cuatro estructuras discretas llamadas *nucléotidos*. Podemos entonces pensar a la molécula de ADN

[1] Aquellos más versados en genética notarán que esto es una simplificación; no estamos contemplando al decir esto, por ejemplo, mecanismos de regulación epigenéticos. También estamos suponiendo, en un acto de antropocentrismo, que el lector no es una bacteria y, por lo tanto, no tiene ADN circular.

como una secuencia de cuatro letras, a saber, A, C, G y T. Estas secuencias son el plano para que la maquinaria celular utiliza para construir, entre otras moléculas, las proteínas necesarias para el funcionamiento de organismo. El ADN no se encuentra dentro de las células de forma lineal si no forma estructuras llamadas *cromosomas*. Al conjunto de los cromosomas (y, en consecuencia, al conjunto de todo el ADN nuclear) lo llamamos el *genoma* de un organismo.

Consideremos una secuencia en particular dentro de una población, por ejemplo,

$$\texttt{AAGCATTAGCAATT}.$$

Tomaremos esta secuencia como una secuencia *de referencia*. Los distintos individuos de esta población pueden tener todos la misma secuencia o pueden tener variantes. A las variantes de una secuencia le llamaremos *alelos*. En caso de que exista una única variante en la población diremos que este alelo está *fijado* en la población. Los alelos pueden darse por distintas situaciones. Una posibilidad es que un nucleótido esté cambiado por otro, es decir,

$$\texttt{AAGCA}\mathbf{\texttt{T}}\texttt{TAGCAATT} \rightarrow \texttt{AAGCA}\mathbf{\texttt{G}}\texttt{TAGCAATT}.$$

Si hay un cambio en un sólo nucleótido con respecto a una secuencia de referencia diremos que existe un *polimorfismo de un sólo nucleótido* (SNP, por sus siglas en inglés *single-nucleotide polymorphism*). También puede ocurrir que se pierda un nucléotido lo que llamaremos una *deleción*,

$$\texttt{AAGCA}\mathbf{\texttt{T}}\texttt{TAGCAATT} \rightarrow \texttt{AAGCA TAGCAATT}.$$

Es posible también que se gane un nucleótido lo que llamaremos una *inserción*:

$$\texttt{AAGCA}\mathbf{\texttt{T}}\texttt{TAGCAATT} \rightarrow \texttt{AAGCA}\mathbf{\texttt{TA}}\texttt{TAGCAATT}.$$

Algunos organismos, como la mayoría de los mamíferos, presentan dos copias de su material genético en la mayoría de sus células: una obtenida por vía materna y otra por vía paterna. Aquellas células con el doble del material genético son llamadas *células diploides* mientras que aquellas que solo cuentan con una copia son *células haploides*. En humanos, las células haploides son los óvulos y los espermatozoides y las diploides son todo el resto.

Por último, si un individuo presenta dos copias exactas de una misma secuencia dentro de una célula diploide, diremos que el individuo es *homocigota* para esta secuencia. De forma contraria, si el individuo presenta dos copias diferentes, diremos que el individuo es *heterocigota* para esta secuencia.

Aplicación de PCA a datos genómicos

En la siguiente sección estudiaremos cómo se vinculan el análisis de componentes principales a los datos genómicos. En esta sección seguiremos las ideas presentadas en [7].

Consideremos n individuos y d marcadores (posiciones dentro del genoma) bialélicos en sus genomas. Como nuestro objetivo es el de utilizar PCA para estudiar la relación que tienen individuos desde un punto de vista genético, es necesario que estos marcadores sean variables; aquellas posiciones en el genoma que se han fijado en el conjunto de datos no

nos aportarán información. Para cada uno de los marcadores seleccionemos un alelo de referencia (con respecto a un genoma de referencia prefijado) y alelos variantes.

Construiremos una matriz $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_d] \in \mathbb{R}^{d \times n}$ de la siguiente forma. Para cada individuo $\mathbf{x}_j$ contaremos la cantidad de copias que tiene del alelo variante en la posición i. Así, un individuo diploide tendrá 0, 1 o 2 copias (correspondientes a los casos en que el individuo sea homocigota para el alelo de referencia, que sea heterocigota o que sea homocigota para la variante de ese alelo, respectivamente). Un individuo haploide, por su parte, tendrá solamente valores 0 o 1 en la posición i.

En [7] se muestra que la proyección de los individuos $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ sobre componentes principales (es decir, los vectores $\mathbf{y}_1, \ldots, \mathbf{y}_n$ del modelo (2.0.3)) permite revelar la estructura poblacional. Esto nos permite realizar un análisis no supervisado para estudiar la estructura poblacional de un conjunto de datos.

La figura (2.1) muestra dos posibles escenarios obtenidos de la proyección de un conjunto de datos genómicos sobre un espacio de dimensión 2. En la figura (A) se simularon 1000 SNPs independientes de 150 individuos provenientes de tres poblaciones bien diferenciadas. El resultado, coherente con lo propuesto por [7], son tres grupos separados en este plano. Si bien en este caso los datos son simulados y podemos etiquetar y colorear a cada uno de los individuos, una imagen como la obtenida permitiría evidenciar la existencia de subpoblaciones distintas al utilizar datos genómicos reales. En la figura (B) se simularon 50 individuos para 5 poblaciones que forman un gradiente en cuanto a su mezcla entre dos poblaciones ancestrales. El resultado es también un gradiente en el plano formado por las dos primeras componentes principales. Este tipo de gráfico ha sido observado en numerosas ocasiones en datos no simulados y evidencia ancestría en común (por ejemplo, en [10] o [12]) o de aislamiento geográfico entre poblaciones (por ejemplo, en [13] o [14]).

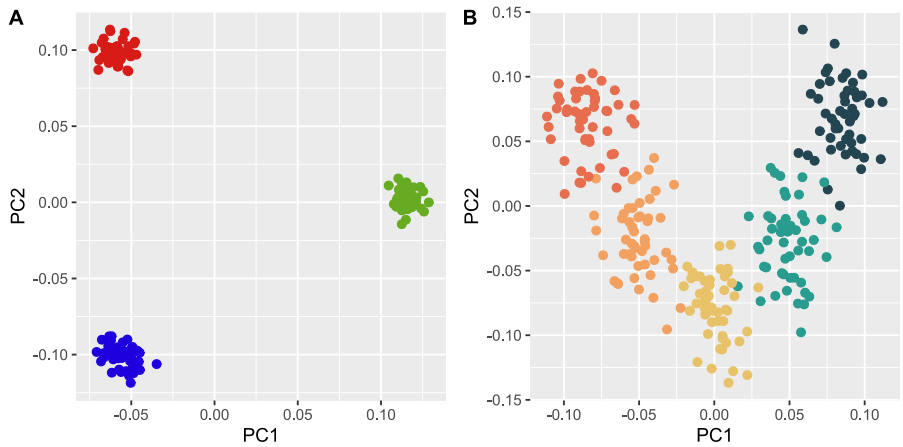

Figura 2.1: Individuos simulados y proyectados en el mejor subespacio afín de dimensión 2 de acuerdo con el problema (2.0.4). (A) Se simularon 150 individuos y 1000 SNPs de 3 poblaciones distintas. (B) Se simularon 250 individuos y 1000 SNPs de 5 poblaciones mezcladas.

Una forma usual de construir un PCA con entradas desconocidas es a través de la imputación con la media (ver por ejemplo la revisión [15]). La idea de esta técnica se basa en que que si la coordenada i de la j-ésima observación de una muestra $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$

es desconocida entonces se definirá

$$x_{ij} \leftarrow \frac{\sum_{k=1}^{n} w_{ik} x_{ik}}{\sum_{k=1}^{n} w_{ik}} \tag{2.0.5}$$

donde w_{ik} vale 0 si la entrada i, k es desconocida o 1 si no lo es. La figura (2.2) muestra por qué esta solución puede no ser apropiada en el estudio de poblaciones mezcladas. En estas figuras se simularon 300 individuos: 150 que forman parte de poblaciones bien distinguidas y 150 que son mezclas de las poblaciones de base. En la figura A, la proyección sobre las dos primeras componentes es un triángulo en donde los individuos de las poblaciones de base se hallan en los vértices y los individuos mezclados se hallan en vértices del triángulo (este diseño es coherente con el obtenido en poblaciones naturales, por ejemplo en [10]). El diseño de esta figura se debe a que los individuos mezclados fueron formados combinando solo dos poblaciones de base a la vez. Sin embargo, si los individuos fueron una mezcla de las tres poblaciones, la proyección de los individuos se hallaría cerca del baricentro de este triángulo. Para construir la figura B, se simularon datos faltantes al azar para los individuos mezclados y se imputaron sus entradas desconocidas mediante la ecuación (2.0.5). Para obtener una entrada faltante se simuló una variable aleatoria Bernoulli de parámetro $p = 0{,}5$ para cada una de las entradas de la matriz correspondientes a los individuos mezclados; si la variable simulada fue igual a 1, se borró esta posición. El resultado es que los individuos mezclados ahora aparecen cerca del baricentro del triángulo y cercanos al $(0, 0)$. Esto podría dar lugar a la interpretación de que los individuos son mezclas de las tres poblaciones pero debido a que conocemos el proceso generador de los datos, sabemos que esta interpretación es errónea.

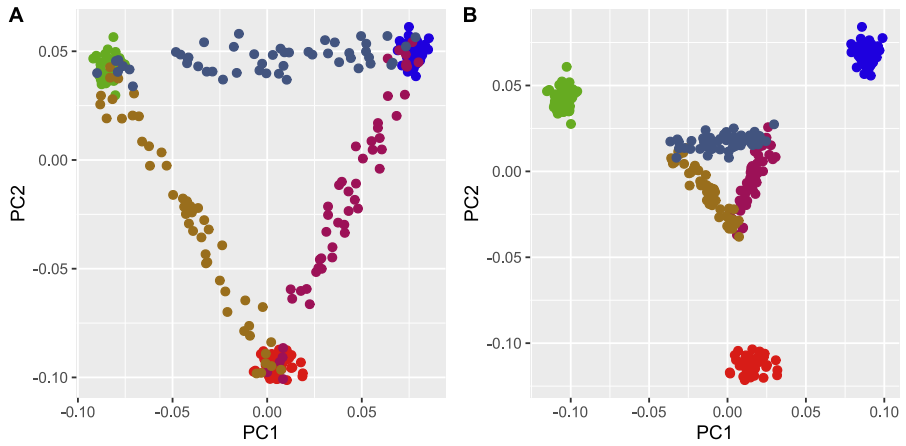

Figura 2.2: Simulación de 300 individuos y 1000 SNPs de seis poblaciones. Del total de individuos, 150 formaron parte de poblaciones base y los individuos restantes son individuos mezclados a partir de dos poblaciones base. (A) Proyección sobre las dos primeras componentes de la base de datos completa. (B) Proyección sobre las dos primeras componentes de la base de datos imputada. Se construyeron datos faltantes aleatorios a los individuos de poblaciones mezcla y se imputaron sus entradas mediante la ecuación (2.0.5).

Es pertinente, por lo tanto, el estudio de formas alternativas de realizar un PCA en el contexto de datos faltantes con el fin de evitar artefactos como el observado en la figura (2.2). El objetivo de este trabajo se centrará en el estudio de estas técnicas y en una comparación de las mismas.

Capítulo 3

El enfoque de PCA como un problema de factorización de matrices

Consideremos nuevamente un conjunto de puntos $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbb{R}^d$, un $p \in \mathbb{Z}^+$ fijo y el modelo afín

$$\mathbf{x}_j = \boldsymbol{\mu} + \mathbf{U}\mathbf{y}_j + \varepsilon_j, \tag{3.0.1}$$

con $\boldsymbol{\mu} \in \mathbb{R}^d$, $\mathbf{U} \in \mathbb{R}^{d\times p}$, $\mathbf{y}_j \in \mathbb{R}^p$. La diferencia en este modelo con respecto al presentado en el capítulo de introducción es que permitiremos que los puntos $\mathbf{x}_i$ tengan entradas desconocidas o faltantes. ¿Cómo podemos encontrar los parámetros del modelo (3.0.1) si no conocemos la totalidad de las observaciones?

Para poder definir la función objetivo del problema, definamos un conjunto de pesos $\{w_{ij}\}_{i,j=1}^n$ de tal forma que

$$w_{ij} = \begin{cases} 1 & \text{si la coordenada } i \text{ de } \mathbf{x}_j \text{ es conocida} \\ 0 & \text{si no} \end{cases} \tag{3.0.2}$$

Estamos interesados en minimizar el error de representación de los puntos $\mathbf{x}_i$ según la norma 2, es decir, queremos minimizar

$$\min_{\varepsilon_j} \sum_{j=1}^n \|\varepsilon_j\|^2 = \min_{\boldsymbol{\mu},\mathbf{U},\{\mathbf{y}_j\}} \sum_{j=1}^n \|\mathbf{x}_i - \boldsymbol{\mu} - \mathbf{U}\mathbf{y}_j\|_2^2 .$$

Ahora bien, como algunas de las entradas de los puntos $\mathbf{x}_i$ son eventualmente desconocidas, no las incluiremos en la función objetivo. Al igual que en el capítulo (2) consideraremos que $\mathbf{U}^\top\mathbf{U} = \mathbf{I}_p$ y $\sum_{j=1}^n \mathbf{y}_j = \mathbf{0}$. Luego el problema de optimización que construiremos es

$$\begin{aligned} & \underset{\boldsymbol{\mu},\mathbf{U},\{\mathbf{y}_j\}_{j=1}^n}{\text{minimizar}} && \sum_{i=1}^d \sum_{j=1}^n w_{ij}(x_{ij} - \mu_i - \mathbf{u}_i^\top \mathbf{y}_j)^2 \\ (\mathrm{P}) \quad & \text{sujeto a} && \mathbf{U}^\top\mathbf{U} = \mathbf{I}_p \\ & && \sum_{j=1}^n \mathbf{y}_j = \mathbf{0}_p \end{aligned} \tag{3.0.3}$$

donde $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)$ y $\{\mathbf{u}_1^\top, \ldots, \mathbf{u}_n^\top\}$ son las filas de la matriz $\mathbf{U}$. Observemos que esta función de costo es análoga a la del problema (2.0.4) con la diferencia de que los errores

$\varepsilon_{ij} = x_{ij} - \mu_i - \mathbf{u}_i^\top \mathbf{y}_j$ asociados a las entradas faltantes (y, por lo tanto, con $w_{ij} = 0$) no son contemplados.

Previo al estudio de cómo resolver el problema (3.0.3) daremos una forma alternativa del problema que será de utilidad. Construyamos la matriz $\mathbf{X} = [\mathbf{x}_1, \ldots, x_n] \in \mathbb{R}^{d \times n}$ que tiene nuestras observaciones como columnas. El modelo en (3.0.1) puede ser escrito entonces de forma matricial como

$$\mathbf{X} = \boldsymbol{\mu}\mathbf{1} + \mathbf{U}\mathbf{Y} + \boldsymbol{\varepsilon}, \tag{3.0.4}$$

donde $\mathbf{1} = (1, 1, \ldots, 1) \in \mathbb{R}^{1 \times n}$. Asociada a la matriz $\mathbf{X}$ podemos construir la matriz $\mathbf{W} = (w_{ij})$ de tamaño $d \times n$ tal que $w_{ij} = 1$ si la entrada (i, j) es conocida o 0 si no lo es. Esto nos permite escribir el problema de optimización (3.0.3) de forma matricial como

$$\text{(P)} \quad \begin{aligned} \underset{\boldsymbol{\mu}, \mathbf{U}, \mathbf{Y}}{\text{minimizar}} \quad & \|\mathbf{W} \odot (\mathbf{X} - \boldsymbol{\mu}\mathbf{1} - \mathbf{U}\mathbf{Y})\|_F^2 \\ \text{sujeto a} \quad & \mathbf{U}^\top \mathbf{U} = \mathbf{I}_p \\ & \sum_{j=1}^{n} \mathbf{y}_j = \mathbf{0}_p \end{aligned} \tag{3.0.5}$$

donde $\odot$ es el producto elemento a elemento.

La ecuación (3.0.4) nos muestra que si los puntos pueden ser reconstruidos de forma exacta por un espacio afín de dimensión p (es decir, si $\varepsilon = \mathbf{0}$), entonces la matriz de datos centrados puede ser escrita como el producto de dos matrices; es decir,

$$\mathbf{X} - \boldsymbol{\mu}\mathbf{1} = \mathbf{U}\mathbf{Y}.$$

En el siguiente capítulo presentaremos algoritmos para realizar PCA mediante esta factorización de matrices.

3.1. Algoritmo de minimización alternada

3.1.1. Construcción del algoritmo

El primer algoritmo que presentaremos está basado en el algoritmo de *PowerFactorization* presentado en [16] con las modificaciones realizadas en [17].

El objetivo de este algoritmo será construir una sucesión $\{(\boldsymbol{\mu}_{(k)}, \mathbf{U}_{(k)}, \mathbf{Y}_{(k)})\}_{k \in \mathbb{N}}$ que converja al óptimo del problema (3.0.3). Consideraremos en una primera instancia al problema (3.0.3) sin restricciones y, posteriormente, realizaremos un tratamiento *ad-hoc* de las soluciones halladas con el fin de que estas cumplan las restricciones. Derivemos entonces a la función objetivo con respecto a cada uno de los parámetros del modelo. Para todo $k = 1, \ldots, d$ tenemos que

$$\frac{\partial}{\partial \mu_k} \sum_{i=1}^{d} \sum_{j=1}^{n} w_{ij}(x_{ij} - \mu_i - \mathbf{u}_i^\top \mathbf{y}_j)^2 = -2 \sum_{j=1}^{n} w_{kj}(x_{kj} - \mu_k - \mathbf{u}_k^\top \mathbf{y}_j) = 0,$$

si y sólo si

$$\left(\sum_{j=1}^{n} w_{kj} \right) \mu_k = \sum_{j=1}^{n} w_{kj}(x_{kj} - \mathbf{u}_k^\top \mathbf{y}_j). \tag{3.1.1}$$

Por otra parte, utilizando la regla de la cadena y la proposición (B.1.1), tenemos que

$$\frac{\partial}{\partial \mathbf{u}_k} \sum_{i=1}^{d} \sum_{j=1}^{n} w_{ij}(x_{ij} - \mu_i - \mathbf{u}_i^\top \mathbf{y}_j)^2 = -2 \sum_{j=1}^{n} w_{kj}(x_{kj} - \mu_i - \mathbf{u}_k^\top \mathbf{y}_j)\mathbf{y}_j^\top = \mathbf{0}^\top,$$

si y sólo si

$$\sum_{j=1}^{n} w_{kj} \mathbf{u}_k^\top \mathbf{y}_j \mathbf{y}_j^\top = \sum_{j=1}^{n} w_{kj}(x_{kj} - \mu_i)\mathbf{y}_j^\top$$

o, equivalentemente,

$$\left(\sum_{j=1}^{n} w_{kj} \mathbf{y}_j \mathbf{y}_j^\top \right) \mathbf{u}_k = \sum_{j=1}^{n} w_{kj}(x_{kj} - \mu_i)\mathbf{y}_j. \tag{3.1.2}$$

Análogamente probamos que para todo $\ell = 1, \ldots, n$,

$$\frac{\partial}{\partial \mathbf{y}_\ell} \sum_{i=1}^{d} \sum_{j=1}^{n} w_{ij}(x_{ij} - \mu_i - \mathbf{u}_i^\top \mathbf{y}_j)^2 = \mathbf{0}^\top,$$

si y sólo si,

$$\left(\sum_{i=1}^{d} w_{i\ell} \mathbf{u}_i \mathbf{u}_i^\top \right) \mathbf{y}_\ell = \sum_{i=1}^{d} w_{i\ell}(x_{i\ell} - \mu_i)\mathbf{u}_i. \tag{3.1.3}$$

Los parámetros, $\boldsymbol{\mu}, \mathbf{U}$ e $\mathbf{Y}$ no pueden ser obtenidos de forma explícita a través de las ecuaciones (3.1.1), (3.1.2) y (3.1.3). Sin embargo, observemos que si conocemos $\mathbf{U}$ e $\mathbf{Y}$, podemos obtener $\boldsymbol{\mu}$ de la ecuación (3.1.1). Análogamente, conociendo $\boldsymbol{\mu}$ e $\mathbf{Y}$, podemos obtener $\mathbf{U}$ de la ecuación (3.1.2). De la misma forma, conociendo $\boldsymbol{\mu}$ y $\mathbf{U}$, podemos obtener $\mathbf{Y}$ de la ecuación (3.1.3). Esto nos llevará al algoritmo de *minimización alternada*: fijados dos parámetros, minimizaremos según el parámetro restante.

Los parámetros obtenidos, sin embargo, no contemplan las restricciones del problema (3.0.3). Para imponer que la matriz $\mathbf{U}$ cumpla que $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$, utilizaremos la descomposición QR de $\mathbf{U}$ (ver apéndice A.3). Para encontrar la matriz $\mathbf{U} \in \mathbb{R}^{d \times p}$ que resuelva la restricción de que $\mathbf{U}^\top \mathbf{U} = \mathbf{I}$, podemos, en virtud de la proposición (A.3.2) actualizar a la matriz $\mathbf{U}$ en cada iteración por el factor $\mathbf{Q}_1$ de la descomposición QR compacta.

Resta imponer la restricción de que $\sum_{j=1}^{n} \mathbf{y}_j = \mathbf{0}$. Observemos que si $\boldsymbol{\mu}$, $\mathbf{U}$ e $\mathbf{Y}$ son una solución de (3.0.3), y notando con $\mathbf{1} = (1, \ldots, 1) \in \mathbb{R}^{1 \times n}$, podemos escribir

$$\begin{aligned} \mathbf{X} = \boldsymbol{\mu}\mathbf{1} + \mathbf{U}\mathbf{Y} &= \boldsymbol{\mu}\mathbf{1} + \mathbf{U}\mathbf{Y} + \mathbf{U}\mathbf{Y}\frac{1}{n}\mathbf{1}^\top\mathbf{1} - \mathbf{U}\mathbf{Y}\frac{1}{n}\mathbf{1}^\top\mathbf{1} \\ &= \underbrace{\left(\boldsymbol{\mu} + \frac{1}{n}\mathbf{U}\mathbf{Y}\mathbf{1}^\top\right)}_{\tilde{\boldsymbol{\mu}}}\mathbf{1} + \mathbf{U}\underbrace{\mathbf{Y}(\mathbf{I} - \frac{1}{n}\mathbf{1}^\top\mathbf{1})}_{\tilde{\mathbf{Y}}} \end{aligned}$$

Esta nueva matriz $\tilde{\mathbf{Y}}$ está construida de tal forma que cumple que $\sum_{j=1}^{n} \mathbf{y}_j = \mathbf{0}$. Por lo tanto, la estrategia será la de obtener $\boldsymbol{\mu}$ y $\mathbf{Y}$ del problema sin restricciones y luego devolver $\tilde{\boldsymbol{\mu}} = \boldsymbol{\mu} + \frac{1}{n}\mathbf{U}\mathbf{Y}\mathbf{1}^\top$ e $\tilde{\mathbf{Y}} = \mathbf{Y}(\mathbf{I} - \frac{1}{n}\mathbf{1}^\top\mathbf{1})$

Podemos entonces definir el algoritmo de minimización alternada (que notaremos de ahora en más como ***Min-Alt***).

(1) Se toma como entrada una matriz $\mathbf{X} = (x_{ij}) \in \mathbb{R}^{d\times n}$ con entradas faltantes, la dimensión p de la reconstrucción según el modelo (3.0.1) y un umbral $\varepsilon > 0$. Se construye la matriz $\mathbf{W} = (w_{ij}) \in \mathbb{R}^{d\times n}$ según la ecuación (3.0.2).

(2) Se inicializan $\mathbf{U}_0 = \begin{pmatrix} \mathbf{u}_1^\top \\ \vdots \\ \mathbf{u}_d^\top \end{pmatrix} \in \mathbb{R}^{d\times p}$ e $\mathbf{Y}_{(0)} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{p\times n}$.

(3) Se fija una regla para detener la iteración. Una posible regla es la siguiente: definiendo

$$\varphi(\boldsymbol{\mu}, \mathbf{U}, \mathbf{Y}) = \|\mathbf{W} \odot (\mathbf{X} - \boldsymbol{\mu}\mathbf{1} - \mathbf{U}\mathbf{Y})\|_F^2 ,$$

se construye $\{\boldsymbol{\mu}_{(k)}, \mathbf{U}_{(k)}, \mathbf{Y}_{(k)}\}$ hasta el primer k_0 para el cual se cumple que

$$|\varphi(\boldsymbol{\mu}_{(k_0)}, \mathbf{U}_{(k_0)}, \mathbf{Y}_{(k_0)}) - \varphi(\boldsymbol{\mu}_{(k_0-1)}, \mathbf{U}_{(k_0-1)}, \mathbf{Y}_{(k_0)})| < \varepsilon \tag{3.1.4}$$

(4) Hasta la convergencia se actualizan los parámetros como

$$\begin{aligned}
\mu_i &\leftarrow \frac{\sum_{j=1}^n w_{ij}(x_{ij} - \mathbf{u}_i^\top \mathbf{y}_j)}{\sum_{j=1}^n w_{ij}}, \\
\mathbf{u}_i &\leftarrow \left(\sum_{j=1}^n w_{ij}\mathbf{y}_j\mathbf{y}_j^\top\right)^{-1} \sum_{j=1}^n w_{ij}(x_{ij} - \mu_i)\mathbf{y}_j, \\
\mathbf{U} &\leftarrow \mathbf{Q_1}, \quad \text{donde } \mathbf{U} = \mathbf{Q}_1\mathbf{R}_1 \text{ es la descomposición QR de } \mathbf{U}, y \\
\mathbf{y}_j &\leftarrow \left(\sum_{i=1}^d w_{ij}\mathbf{u}_i\mathbf{u}_i^\top\right)^{-1} \sum_{i=1}^d w_{ij}(x_{ij} - \mu_i)\mathbf{u}_i.
\end{aligned}$$

(5) Se devuelve como salida $\boldsymbol{\mu} + \frac{1}{n}\mathbf{U}\mathbf{Y}\mathbf{1}^\top$, $\mathbf{U}$ e $\mathbf{Y}(\mathbf{I} - \frac{1}{n}\mathbf{1}^\top\mathbf{1})$.

3.1.2. Análisis de la convergencia del algoritmo

Caso sin datos faltantes y con columnas centradas

Estamos interesados en analizar la convergencia a una solución global del problema del algoritmo descrito en (3.1). Para esto consideremos, en una primera instancia, el problema de la reconstrución de los puntos $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ sin datos faltantes y centrados. Es decir, utilizando la formulación de la ecuación (3.0.5), estamos interesados en estudiar el problema

$$(\mathbf{P}) \quad \begin{array}{ll} \underset{\mathbf{U}\in\mathbb{R}^{d\times p},\, \mathbf{Y}\in\mathbb{R}^{p\times n}}{\text{minimizar}} & \|\mathbf{X} - \mathbf{U}\mathbf{Y}\|_F^2 \\ \text{sujeto a} & \mathbf{U}^\top\mathbf{U} = \mathbf{I}_p \end{array}. \tag{3.1.5}$$

Derivando con respecto a $\mathbf{U}$ e $\mathbf{Y}$ igualando las derivadas a $\mathbf{0}$ tenemos que

$$\frac{\partial}{\partial \mathbf{U}} \|\mathbf{X} - \mathbf{U}\mathbf{Y}\|_F^2 = -2(\mathbf{X} - \mathbf{U}\mathbf{Y})\mathbf{Y}^\top = \mathbf{0} \tag{3.1.6}$$

y

$$\frac{\partial}{\partial \mathbf{Y}} \|\mathbf{X} - \mathbf{U}\mathbf{Y}\|_F^2 = -2\mathbf{U}^\top(\mathbf{X} - \mathbf{U}\mathbf{Y}) = \mathbf{0} \tag{3.1.7}$$

Conocida $\mathbf{Y}$, podemos despejar $\mathbf{Y}$ de (3.1.6) como $\mathbf{U} = \mathbf{X}\mathbf{Y}(\mathbf{Y}\mathbf{Y}^\top)^{-1}$ y conocida $\mathbf{U}$ podemos despejar $\mathbf{Y}$ de (3.1.7) como $\mathbf{Y} = (\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{U}^\top\mathbf{X}$. Esto nos motiva a definir la siguiente sucesión de matrices $\{\mathbf{U}_{(k)}, \mathbf{Y}_{(k)}\}_{k\in\mathbb{N}}$:

(1) Comenzar con $\mathbf{U}_0$ e $\mathbf{Y}_0$ arbitrarios.

(2) Definir $\tilde{\mathbf{U}}_{(k+1)} = \mathbf{X}\mathbf{Y}_{(k)}(\mathbf{Y}_{(k)}\mathbf{Y}_{(k)}^\top)^{-1}$.

(3) Para imponer que la solución cumpla $\mathbf{U}^\top\mathbf{U} = \mathbf{I}_p$, definir $\mathbf{U}_{(k)} = \mathbf{Q}_{(k)}$ donde $\tilde{\mathbf{U}}_{(k+1)} = \mathbf{Q}_{(k)}\mathbf{R}_{(k)}$ es la descomposición QR compacta de $\tilde{\mathbf{U}}_{(k+1)}$.

(4) Definir $\mathbf{Y}_{(k)} \leftarrow (\mathbf{U}_{(k+1)}^\top\mathbf{U}_{(k+1)})^{-1}\mathbf{U}_{(k)}^\top\mathbf{X} = \mathbf{U}_{(k+1)}^\top\mathbf{X}$.

En [16] se presenta la siguiente proposición que asegura la convergencia teórica de la iteración anteriormente descrita a la mejor aproximación de rango p de $\mathbf{X}$ si existe un intervalo entre el p-ésimo valor singular de $\mathbf{X}$ y el valor singular $(p+1)$-ésimo. Esto está dado por la siguiente proposición.

Proposición 3.1.1 (Convergencia del algoritmo de minimización alternada)**.** *Sea $\mathbf{X} \in \mathbb{R}^{d\times n}$. Sean $\sigma_1, \ldots, \sigma_j$ los valores singulares de $\mathbf{X}$ ordenados de forma creciente. Supongamos que $\sigma_p < \sigma_{p+1}$. Si $\mathbf{X}_p$ es la mejor aproximación de rango p de $\mathbf{X}$, entonces existe una constante C (que depende de $\mathbf{X}$ y $\mathbf{U}_0$) tal que para todo k*

$$\left\|\mathbf{X}_p - \mathbf{U}_{(k)}\mathbf{Y}_{(k)}\right\|_F \leq C\left(\frac{\sigma_{p+1}}{\sigma_p}\right)^{2k}.$$

Demostración. Ver [17]. ■

Caso con datos faltantes

En el caso de que existan datos faltantes en la matriz de entrada no podemos asegurar asegurar la convergencia a los parámetros óptimos del modelo (3.0.5) y si esta existe, está condicionada a la elección de los valores iniciales.

Para ejemplificar este fenómeno, consideremos la matriz

$$\mathbf{X} = \begin{pmatrix} 1 & 2 & -2 \\ 2 & 5 & 3 \\ 3 & 7 & \texttt{NA} \end{pmatrix},$$

donde `NA` indica que este valor es desconocido. Estamos interesados en reconstruir a la matriz $\mathbf{X}$ mediante

$$\hat{\mathbf{X}}_2 = \boldsymbol{\mu}\mathbf{1} + \mathbf{U}\mathbf{Y},$$

donde $\mathbf{U} \in \mathbb{R}^{3\times 2}$ e $\mathbf{Y} \in \mathbb{R}^{2\times 3}$; es decir, queremos una reconstrucción de rango 2. Construyamos la secuencia $\{\boldsymbol{\mu}_{(k)}, \mathbf{U}_{(k)}, \mathbf{Y}_{(k)}\}_{k\in\mathbb{N}}$ según el algoritmo Min-Alta tomando como regla de detención aquella dada por la ecuación (3.1.4) con $\varepsilon = 10^{-10}$.

Tomemos, para ejemplificar, los valores iniciales

$$\mathbf{U}_0 = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix} \quad \text{e} \quad \mathbf{Y}_0 = \begin{pmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{pmatrix}$$

llegamos a que la reconstrucción de $\mathbf{X}$ de rango 2 es

$$\mathbf{X} = \begin{pmatrix} 1 & 2 & -2 \\ 2 & 5 & 3 \\ 3 & 7 & 9{,}214081 \end{pmatrix}, \quad \left\| \mathbf{W} \odot (\hat{\mathbf{X}}_2 - \mathbf{X}) \right\|_F = 0.$$

Mientras que si comenzamos con valores iniciales

$$\mathbf{U}_{(0)} = \begin{pmatrix} -1{,}04 & 0{,}02 \\ 0{,}15 & 0{,}89 \\ -0{,}21 & 0{,}05 \end{pmatrix} \quad \text{e} \quad \mathbf{Y}_{(0)} = \begin{pmatrix} -1{,}93 & 1{,}02 & 0{,}42 \\ 0{,}78 & -1{,}09 & 0{,}28 \end{pmatrix},$$

que corresponden con matrices cuyas entradas son normales de media 0 y varianza 1, el resultado es

$$\mathbf{X} = \begin{pmatrix} 1 & 2 & -2 \\ 2 & 5 & 3 \\ 3 & 7 & 4{,}245046 \end{pmatrix}, \quad \left\| \mathbf{W} \odot (\hat{\mathbf{X}}_2 - \mathbf{X}) \right\|_F = 0$$

3.2. Algoritmo de aprendizaje de subespacios

3.2.1. Construcción del algoritmo para datos centrados

En esta sección presentaremos un algoritmo iterativo para resolver el problema (3.0.5) en el caso en que las columnas de $\mathbf{X}$ están centradas. Este algoritmo está basado en lo propuesto por [10] y [18].

La idea de este método para hallar $\mathbf{U}$ e $\mathbf{Y}$ es utilizar el método del gradiente descendente para obtener una sucesión $\{\mathbf{U}_{(k)}, \mathbf{Y}_{(k)}\}_{k\in\mathbb{N}}$ que resuelva el problema

$$(\mathrm{P}) \quad \begin{array}{ll} \underset{\mathbf{U}\in\mathbb{R}^{d\times p},\, \mathbf{Y}\in\mathbb{R}^{p\times n}}{\text{minimizar}} & \frac{1}{2} \left\| \mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y}) \right\|_F^2 \\ \text{sujeto a} & \mathbf{U}^\top \mathbf{U} = \mathbf{I}_p \end{array} \tag{3.2.1}$$

Calculemos entonces las derivadas con respecto a $\mathbf{U}$ e $\mathbf{Y}$ de la función objetivo. Estas son

$$\frac{\partial}{\partial \mathbf{U}} \left(\frac{1}{2} \left\| \mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y}) \right\|_F^2 \right) = -(\mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y}) \odot \mathbf{W})\mathbf{Y}^\top = -(\mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y}))\mathbf{Y}^\top$$

y

$$\frac{\partial}{\partial \mathbf{Y}} \left(\frac{1}{2} \left\| \mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y}) \right\|_F^2 \right) = -\mathbf{U}^\top (\mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y}) \odot \mathbf{W}) = -\mathbf{U}^\top (\mathbf{W} \odot (\mathbf{X} - \mathbf{U}\mathbf{Y})).$$

Por lo tanto, dado un inicial $(\mathbf{U}_0, \mathbf{Y}_0)$ y una sucesión $\{\delta_k\}_{k\in\mathbb{N}}$ actualizaremos

$$\mathbf{U}_{(k+1)} = \mathbf{U}_{(k)} + \delta_k (\mathbf{W} \odot (\mathbf{X} - \mathbf{U}_{(k)}\mathbf{Y}_{(k)}))\mathbf{Y}_{(k)}^\top \tag{3.2.2}$$

y

$$\mathbf{Y}_{(k+1)} = \mathbf{Y}_{(k)} + \delta_k \mathbf{U}_{(k)}^\top (\mathbf{W} \odot (\mathbf{X} - \mathbf{U}_{(k)}\mathbf{Y}_{(k)})) \tag{3.2.3}$$

La actualización en (3.2.2) y (3.2.3) presenta una dificultad asociada a la función objetivo del problema. La función objetivo de (3.2.1) no es convexa: en particular si $\mathbf{U}$ e $\mathbf{Y}$ son

solución, también lo serán $\frac{1}{\alpha}\mathbf{U}$ y $\alpha\mathbf{Y}$ para todo $\alpha \neq 0$. En consecuencia, puede ocurrir que $\left\|\mathbf{U}_{(k)}\right\|_F \to \infty$ o $\left\|\mathbf{Y}_{(k)}\right\|_F \to \infty$ cuando $k \to \infty$. Podemos plantear dos formas posibles de resolver este problema.

La primera es modificar $\mathbf{U}_{(k)}$ por la matriz $\mathbf{Q}$ correspondiente a la descomposición QR de $\mathbf{U}_{(k)}$. Esto, además de impedir que la norma Frobenius de $\mathbf{U}_{(k)}$ diverja, tiene el valor agregado de imponer la restricción del problema (3.2.1). La desventaja de este enfoque es que agregar un cálculo de una descomposición QR en cada iteración volvería al algoritmo intenso computacionalmente en detrimento de las bondades del algoritmo de gradiente descendente que es computacionalmente poco costoso (observemos que en las ecuaciones (3.2.2) y (3.2.3) solo hay sumas y multiplicaciones de matrices).

La segunda forma ampliamente utilizada de remediar este problema es modificar la función objetivo agregando unos factores de *regularización* (ver, por ejemplo, la revisión de estos métodos realizada en [19]). Dado un $\lambda > 0$ planteamos el problema

$$\begin{aligned} \underset{\mathbf{U}\in\mathbb{R}^{d\times p},\, \mathbf{Y}\in\mathbb{R}^{p\times n}}{\text{minimizar}} \quad & \frac{1}{2}\left(\|\mathbf{X}-\mathbf{U}\mathbf{Y}\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{Y}\|_F^2)\right) \\ \text{sujeto a} \quad & \mathbf{U}^\top\mathbf{U} = \mathbf{I}_p \end{aligned} \tag{3.2.4}$$

Tras derivar, podemos construir el siguiente algoritmo:

(1) Comenzar con $\mathbf{U}_0$ y $\mathbf{Y}_0$ arbitrarios y una constante $\lambda > 0$ de regularización.

(2) Realizar la actualización

$$\mathbf{U}_{(k+1)} = \mathbf{U}_{(k)} + \delta_k[(\mathbf{W}\odot(\mathbf{X}-\mathbf{U}_{(k)}\mathbf{Y}_{(k)}))\mathbf{Y}_{(k)}^\top - \lambda\mathbf{U}_{(k)}]$$

y

$$\mathbf{Y}_{(k+1)} = \mathbf{Y}_{(k)} + \delta_k[\mathbf{U}_{(k)}^\top(\mathbf{W}\odot(\mathbf{X}-\mathbf{U}_{(k)}\mathbf{Y}_{(k)}) - \lambda\mathbf{Y}_{(k)}]$$

(3) Una vez alcanzado un criterio de convergencia, devolver $\mathbf{Y}^\star$ y $\mathbf{Q}^\star$ donde $\mathbf{Q}^\star$ es la matriz $\mathbf{Q}$ de la descomposición QR de $\mathbf{U}^\star$.

3.2.2. Construcción del algoritmo general

Si las columnas de $\mathbf{X}$ no están centradas el algoritmo propuesto en la sección previa no arroja resultados satisfactorios. Por otra parte, si la matriz presenta datos faltantes, no es claro cuál es el centro de los datos. Por lo tanto, inspirados por el algoritmo presentado en (3.1) modificaremos el algoritmo de gradiente descendente para que este pueda ser utilizado para matrices no centradas.

Comenzaremos resolviendo el problema

$$\begin{aligned} & \underset{\boldsymbol{\mu},\mathbf{U},\mathbf{Y}}{\text{minimizar}} \quad \frac{1}{2}\left(\|\mathbf{W}\odot(\mathbf{X}-\boldsymbol{\mu}\mathbf{1}-\mathbf{U}\mathbf{Y})\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{Y}\|_F^2)\right) \\ (\mathrm{P}) \quad & \text{sujeto a} \quad \mathbf{U}^\top\mathbf{U} = \mathbf{I}_p \\ & \qquad\qquad \sum_{i=1}^{n}\mathbf{y}_i = \mathbf{0}_p \end{aligned} \tag{3.2.5}$$

Notando con $\varphi(\boldsymbol{\mu}, \mathbf{U}, \mathbf{Y})$ a la función objetivo de (3.2.5) obtenemos que las derivadas de la φ con respecto a $\mathbf{U}$ e $\mathbf{Y}$ son

$$\frac{\partial}{\partial \mathbf{U}}\varphi(\boldsymbol{\mu}, \mathbf{U}, \mathbf{Y}) = -(\mathbf{W} \odot (\mathbf{X} - \boldsymbol{\mu}\mathbf{1} - \mathbf{U}\mathbf{Y}))\mathbf{Y}^\top + \mathbf{U}$$

y

$$\frac{\partial}{\partial \mathbf{Y}}\varphi(\boldsymbol{\mu}, \mathbf{U}, \mathbf{Y}) = \mathbf{U}^\top(\mathbf{W} \odot (\mathbf{X} - \boldsymbol{\mu}\mathbf{Y} - \mathbf{U}\mathbf{Y})) + \mathbf{Y}.$$

Luego, podemos plantear una iteración de descenso por gradiente para $\mathbf{U}$ e $\mathbf{Y}$. Por otra parte, podemos estimar en cada paso de la iteración, el valor de $\boldsymbol{\mu}$ óptimo según la ecuación (3.1.1).

Esto nos lleva al siguiente algoritmo iterativo que llamaremos ***SLPCA*** por *Subspace Learning PCA*, como referencia al nombre que utilizó [18] en su artículo.

(1) Comenzar con $\mathbf{U}_0$ y $\mathbf{Y}_0$ arbitrarios y una constante $\lambda > 0$ de regularización.

(2) Realizar la actualización

$$\mathbf{U}_{(k+1)} = \mathbf{U}_{(k)} + \delta_k[(\mathbf{W} \odot (\mathbf{X} - \mathbf{U}_{(k)}\mathbf{Y}_{(k)}))\mathbf{Y}_{(k)}^\top - \lambda\mathbf{U}_{(k)}], \tag{3.2.6}$$

$$\mathbf{Y}_{(k+1)} = \mathbf{Y}_{(k)} + \delta_k[\mathbf{U}_{(k)}^\top(\mathbf{W} \odot (\mathbf{X} - \mathbf{U}_{(k)}\mathbf{Y}_{(k)}) - \lambda\mathbf{Y}_{(k)}] \tag{3.2.7}$$

y

$$\mu_i^{(k)} = \frac{\sum_{j=1}^n w_{ij}(x_{ij} - \mathbf{u}_i^\top \mathbf{y}_j)}{\sum_{j=1}^n w_{ij}}, \quad \text{con } \boldsymbol{\mu}_{(k)} = (\mu_1^{(k)}, \ldots, \mu_d^{(k)})^\top.$$

(3) Una vez alcanzado un criterio de convergencia, devolver $\boldsymbol{\mu}^\star$, $\mathbf{U}^\star$ y $\mathbf{Y}^\star$.

(4) Para imponer las restricciones del problema, devolver $\tilde{\mathbf{Y}} = \mathbf{Y}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)$ y $\mathbf{Q}^\star$ donde $\mathbf{Q}^\star$ es la matriz $\mathbf{Q}$ de la descomposición QR de $\mathbf{U}^\star$.

Capítulo 4

Análisis de componentes principales probabilístico

4.1. Modelo con datos completos

Consideremos una muestra de n datos $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ provenientes de un vector aleatorio $\mathbf{x} \in \mathbb{R}^d$. Estamos interesados en encontrar una representación $\{\mathbf{y}_i\}_{i=1}^n \subset \mathbb{R}^p$ de los puntos $\mathbf{x}_i$, con $p < d$ de la siguiente forma:

Sea $\mathbf{y}$ un vector aleatorio con función de densidad $p_{\mathbf{y}}$ y $\boldsymbol{\varepsilon}$ un vector aleatorio con función de densidad $p_{\boldsymbol{\varepsilon}}$ e independiente a $\mathbf{y}$. Consideraremos que el vector $\mathbf{x}$ proviene del modelo

$$\mathbf{x} = \boldsymbol{\mu} + \mathbf{U}\mathbf{y} + \boldsymbol{\varepsilon}, \tag{4.1.1}$$

con $\boldsymbol{\mu} \in \mathbb{R}^d$ y $\mathbf{U} \in \mathbb{R}^{d\times p}$, un vector y una matriz determinísticos, respectivamente. Notemos con $\boldsymbol{\mu}_{\mathbf{x}}$, $\boldsymbol{\mu}_{\mathbf{y}}$ y $\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}$ a los vectores de medias de los vectores $\mathbf{x}$, $\mathbf{y}$ y $\boldsymbol{\varepsilon}$, respectivamente, y con $\boldsymbol{\Sigma}_{\mathbf{x}}$, $\boldsymbol{\Sigma}_{\mathbf{y}}$ y $\boldsymbol{\Sigma}_{\boldsymbol{\varepsilon}}$ a las matrices de covarianza de $\mathbf{x}$, $\mathbf{y}$ y $\boldsymbol{\varepsilon}$, respectivamente. Si suponemos que $\boldsymbol{\mu}_{\varepsilon} = \mathbf{0}$ y $\boldsymbol{\Sigma}_{\varepsilon} = \sigma^2\mathbf{I}_d$ entonces

$$\boldsymbol{\mu}_{\mathbf{x}} = \boldsymbol{\mu} + \mathbf{U}\boldsymbol{\mu}_{\mathbf{y}} \quad \text{y} \quad \boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{U}\boldsymbol{\Sigma}_{\mathbf{y}}\mathbf{U}^\top + \sigma^2\mathbf{I}_d. \tag{4.1.2}$$

La idea será estimar los parámetros del modelo, a saber, $\boldsymbol{\mu}, \mathbf{U}, \boldsymbol{\mu}_{\mathbf{y}}, \boldsymbol{\Sigma}_{\mathbf{y}}$ y σ^2, utilizando la muestra $\{\mathbf{x}_i\}_{i=1}^n$ y vía el método de máxima verosimilitud.

Para poder obtener expresiones cerradas de la función de verosimilitud (y, consecuentemente, de los estimadores) asumiremos que el vector $\mathbf{y}$ tiene distribución normal media $\mathbf{0}$ y matriz de covarianza $\mathbf{I}_p$ y que el vector ε tiene distribución normal con la media y matriz de covarianza previamente discreta. De esto se sigue que, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ con

$$\boldsymbol{\mu}_{\mathbf{x}} = \boldsymbol{\mu} \quad \text{y} \quad \boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{U}\mathbf{U}^\top + \sigma^2\mathbf{I}_d. \tag{4.1.3}$$

Al modelo explicitado en (4.1.1) con las hipótesis sobre las distribuciones asociadas le llamaremos *análisis de componentes principales probabilístico*. La función de verosimilitud asociada es

$$\mathcal{L}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}} | \mathbf{x}_1, \ldots, \mathbf{x}_n) = \prod_{i=1}^n \frac{1}{\sqrt{(2\pi)^d \det(\boldsymbol{\Sigma}_{\mathbf{x}})}} \exp\left(\frac{(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})^\top \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})}{2}\right)$$

Luego, aplicando logaritmo, construimos la función de log-verosimilitud

$$\begin{aligned}\ell(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}}) &\coloneqq \ln(\mathcal{L}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}}|\mathbf{x}_1, \ldots, \mathbf{x}_n)) \\ &= \sum_{i=1}^{n} \ln\left(\frac{1}{\sqrt{(2\pi)^d \det(\boldsymbol{\Sigma}_{\mathbf{x}})}} \exp\left(-\frac{(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})^\top \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})}{2}\right)\right) \\ &= -\frac{nd}{2}\ln(2\pi) - \frac{n}{2}\ln(\det(\boldsymbol{\Sigma}_{\mathbf{x}})) - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})^\top \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})\end{aligned} \tag{4.1.4}$$

La forma de los estimadores por máxima verosimilitud están resumidos en la siguiente proposición, cuya demostración se encuentra en el apéndice (E).

Proposición 4.1.1 (Estimadores por máxima verosimilitud PPCA)**.** *Los parámetros del modelo de análisis de componentes principales probabilístico, $\boldsymbol{\mu}, \mathbf{U}$ y σ^2 obtenidos mediante el método de máxima verosimilitud son iguales a*

$$\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\mu}}_n, \quad \widehat{\mathbf{U}} = \mathbf{U}_p\left(\boldsymbol{\Lambda}_p - \hat{\sigma}^2\mathbf{I}\right)^{1/2}\mathbf{R} \quad y \quad \hat{\sigma}^2 = \frac{1}{d-p}\sum_{i=p+1}^{d}\lambda_i, \tag{4.1.5}$$

donde $\mathbf{U}_p$ es la matriz con los vectores propios asociados a los p valores propios más grandes de $\widehat{\boldsymbol{\Sigma}}_n$, $\boldsymbol{\Lambda}_p$ es la matriz diagonal con los p valores propios más grandes, $\mathbf{R}$ es una matriz ortogonal arbitraria, λ_i es el i-ésimo valor propio más grande de $\widehat{\boldsymbol{\Sigma}}_n$ y $\hat{\boldsymbol{\mu}}_n$ y $\widehat{\boldsymbol{\Sigma}}_n$ son la media y matriz de covarianza muestrales, respectivamente.

Como nuestro interés es el de obtener las proyecciones de datos $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ en un espacio de dimensión p, estamos interesados en conocer cómo obtener valores de y. La ventaja de plantear el modelo (4.1.1) asumiendo una distribución normal para los datos, es que podemos obtener la distribución de $\mathbf{y}|\mathbf{x}$. La distribución condicional de $\mathbf{y}$ dada $\mathbf{x}$ es $\mathbf{y}|\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}})$ con

$$\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}} = (\mathbf{U}^\top\mathbf{U} + \sigma^2\mathbf{I})^{-1}\mathbf{U}^\top(\mathbf{x} - \mu) \quad \text{y} \quad \boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}} = (\mathbf{U}^\top\mathbf{U} + \sigma^2\mathbf{I})^{-1}\mathbf{U}^\top(\mathbf{x} - \mu).$$

Dada una muestra $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ definiremos a los *componentes principales probabilísticos* como

$$\mathbf{y}_j = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x} = \mathbf{x}_j). \tag{4.1.6}$$

Para dar una forma explícita de los componentes principales probabilísticos, recordemos una propiedad que tienen las distribuciones normales.

Proposición 4.1.2 (Máximo de la distribución normal)**.** *Sea $\mathbf{X} \sim \mathcal{N}_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Notemos con $p_{\mathbf{X}}$ a la función de densidad conjunta de $\mathbf{X}$. Luego*

$$\arg\max_{\mathbf{x}} p_{\mathbf{X}}(\mathbf{x}) = \boldsymbol{\mu}.$$

Demostración. Como la función $x \mapsto \ln(x)$ es creciente, tenemos que

$$\arg\max_{\mathbf{x}} p_{\mathbf{X}}(\mathbf{x}) = \arg\max_{\mathbf{x}} \ln(p_{\mathbf{X}}(\mathbf{x}))$$

$$= \arg\max_{\mathbf{x}} \left(-\frac{k}{2} - \frac{1}{2}\det(\boldsymbol{\Sigma}) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) \right) \quad (4.1.7)$$

Derivando (4.1.7) con respecto a un vector $\mathbf{x}$ y de acuerdo con la proposición (B.1.4) obtenemos que

$$\frac{\partial}{\partial \mathbf{x}} \ln(p_{\mathbf{X}}(\mathbf{x})) = -(\mathbf{x}-\mu)\boldsymbol{\Sigma}^{-1} = \mathbf{0}_d^\top$$

si y sólo si $\mathbf{x} = \boldsymbol{\mu}$. Para probar que punto estacionario es efectivamente un candidato a máximo observemos que para todo $\mathbf{x}$,

$$\frac{\partial^2}{\partial^2 \mathbf{x}} \ln(p_{\mathbf{X}}(x)) = -\boldsymbol{\Sigma}^{-1}.$$

Luego, como $\boldsymbol{\Sigma}$ es definida positiva, también lo es $\boldsymbol{\Sigma}^{-1}$. Esto implica que $-\boldsymbol{\Sigma}^{-1}$ es definida negativa. Luego, $\mathbf{x} = \boldsymbol{\mu} = \arg\max_{\mathbf{x}} p_{\mathbf{X}}(\mathbf{x})$. ■

Vinculando, entonces, la proposición (4.1.2) y la ecuación (4.1.6) obtenemos una forma explícita para los componentes principales probabilísticos:

$$\mathbf{y}_j = (\mathbf{U}^\top \mathbf{U} + \sigma^2 \mathbf{I})^{-1} \mathbf{U}^\top (\mathbf{x}_j - \boldsymbol{\mu}) \quad (4.1.8)$$

Ahora bien, como los parámetros del modelo son desconocidos, sustituiremos a los parámetros de la ecuación (4.1.8) por los estimadores por máxima verosimilitud hallados según la proposición (4.1.1). Notemos en primer lugar que

$$\begin{aligned} \mathbf{U}^\top \mathbf{U} + \sigma^2 \mathbf{I} &= [(\mathbf{U}_p(\boldsymbol{\Lambda}_p - \hat{\sigma}^2 \mathbf{I})^{1/2}\mathbf{R})^\top (\mathbf{U}_p(\boldsymbol{\Lambda}_p - \hat{\sigma}^2 \mathbf{I})^{1/2}\mathbf{R})] + \sigma^2 \mathbf{I} \\ &= \mathbf{R}^\top (\boldsymbol{\Lambda}_p - \hat{\sigma}^2 \mathbf{I})^{1/2} \mathbf{U}_p^\top \mathbf{U}_p (\boldsymbol{\Lambda}_p - \hat{\sigma}^2 \mathbf{I})^{1/2} \mathbf{R} + \sigma^2 \mathbf{I} \\ &= \mathbf{R}^\top \boldsymbol{\Lambda}_p \mathbf{R} \end{aligned}$$

y, por lo tanto,

$$(\mathbf{U}^\top \mathbf{U} + \sigma^2 \mathbf{I})^{-1} = \mathbf{R}^\top \boldsymbol{\Lambda}_p^{-1} \mathbf{R}.$$

En consecuencia, los componentes principales pueden ser escritos como

$$\mathbf{y}_j = \mathbf{R}^\top \boldsymbol{\Lambda}_p^{-1} (\boldsymbol{\Lambda}_p - \hat{\sigma}^2 \mathbf{I})^{1/2} \mathbf{U}_p^\top (\mathbf{x}_j - \hat{\boldsymbol{\mu}}_n).$$

4.2. Modelo con datos faltantes

Al igual que en la sección anterior, consideraremos una muestra $\{\mathbf{x}_i\}_{i=1}^n$ de acuerdo al modelo $\mathbf{x} = \boldsymbol{\mu} + \mathbf{U}\mathbf{y} + \boldsymbol{\varepsilon}$, y donde, $\boldsymbol{\mu}$, $\mathbf{U}$ son un vector y una matriz desconocidos pero determinísticos, $\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p)$ y $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. De forma análoga al caso previamente estudiado surge el hecho de que $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ con $\boldsymbol{\mu}_{\mathbf{x}} = \boldsymbol{\mu}$ y $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{U}\mathbf{U}^\top + \sigma^2 \mathbf{I}_d$. La principal diferencia en esta sección es que permitiremos que existan datos faltantes, es decir, cada uno de los elementos de la muestra $\mathbf{x}_i$ puede tener entradas desconocidas.

Como cada elemento $\mathbf{x}_i$ de la muestra tiene potencialmente datos faltantes, podemos particionar a un punto $\mathbf{x}$ y a los parámetros $\boldsymbol{\mu}_{\mathbf{x}}$ y $\boldsymbol{\Sigma}_{\mathbf{x}}$ como

$$\begin{pmatrix} \mathbf{x}_N \\ \mathbf{x}_O \end{pmatrix} = \mathbf{P}\mathbf{x}, \quad \begin{pmatrix} \boldsymbol{\mu}_N \\ \boldsymbol{\mu}_O \end{pmatrix} = \mathbf{P}\boldsymbol{\mu}_{\mathbf{x}}, \quad \text{y} \quad \begin{pmatrix} \boldsymbol{\Sigma}_{NN} & \boldsymbol{\Sigma}_{NO} \\ \boldsymbol{\Sigma}_{ON} & \boldsymbol{\Sigma}_{OO} \end{pmatrix} = \mathbf{P}\boldsymbol{\Sigma}_{\mathbf{x}}\mathbf{P}^\top,$$

donde $\mathbf{x}_O$ y $\mathbf{x}_N$ son las partes observadas y no observadas de $\mathbf{x}$, respectivamente, y $\mathbf{P}$ es una matriz de permutación que reordena las entradas de $\mathbf{x}$ para que las entradas no observadas aparezcan primero. Cabe aclarar que esta matriz $\mathbf{P}$ puede no ser única y que cada entrada $\mathbf{x}_i$ puede necesitar una matriz de permutación distinta. Cuando sea estrictamente necesario, utilizaremos $\mathbf{x}_{iO}$ y $\mathbf{x}_{iN}$ para referirnos a las partes observadas y no observadas de un punto $\mathbf{x}_i$ y notaremos con $\mathbf{P}_i$ a la matriz que reordena el i-ésimo elemento de la muestra.

4.2.1. Construcción del algoritmo MAP-EM

Si conociéramos la verosimilitud completa, podríamos obtener los estimadores por máxima verosimilitud como los de la proposición (4.1.1). Sin embargo, como la función de verosimilitud está incompleta, no podremos utilizar esta estrategia.

Notemos con $\theta = (\boldsymbol{\mu}_{\mathbf{x}}, \mathbf{U}, \sigma^2)$ a los parámetros a estimar del modelo. Notaremos con $p_\theta(\mathbf{x})$ a la función de densidad del vector $\mathbf{x}$. Si conociéramos al vector $\mathbf{x}$ completo tendríamos que

$$p_\theta(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^d \det(\boldsymbol{\Sigma}_{\mathbf{x}})}} \exp\left(\frac{(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}})^\top \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}})}{2}\right),$$

donde $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{U}\mathbf{U}^\top + \sigma^2\mathbf{I}_d$. Como en este caso no necesariamente conocemos los vectores $\mathbf{x}_i$ completos, notaremos con $p_\theta(\mathbf{x}_{iN}, \mathbf{x}_{iO})$ a la densidad del vector $(\mathbf{x}_{iN}, \mathbf{x}_{iO})^\top$. La idea será entonces maximizar la log-verosimilitud incompleta $\ell(\theta, \{(\mathbf{x}_{iN}, \mathbf{x}_{iO})\})$ según θ y según las variables $\{(\mathbf{x}_{iN})\}_{i=1}^n$ desconocidas; es decir, queremos resolver el problema

$$\max_\theta \max_{x_{iN}} \ell(\theta, \{(\mathbf{x}_{iN}, \mathbf{x}_{iO})\}) = \max_\theta \max_{\{x_{iN}\}} \sum_{i=1}^n \ln\left(p_\theta(\mathbf{x}_{iN}, \mathbf{x}_{iO})\right).$$

Esto nos conduce a una estrategia de maximización alternada para estimar θ. Construiremos una sucesión de estimadores $\{\theta_{(k)}\}_{k\in\mathbb{N}}$ de la siguiente forma. Dado θ_k, hallaremos el estimador máximo *a posteriori* de las muestras no observadas en el paso k como

$$\mathbf{x}_{iN}^{(k)} = \arg\max_{\mathbf{z}} p_{\theta_k}(\mathbf{z}|\mathbf{x}_{iO}). \tag{4.2.1}$$

Luego, con las muestras completas podemos actualizar la estimación de θ como

$$\theta_{(k+1)} = \arg\max_\theta \sum_{i=1}^n \ln(p_\theta(\mathbf{x}_i)) \tag{4.2.2}$$

Supongamos que conocemos $\theta_{(k)}$ para algún k. Queremos hallar la solución de (4.2.1). Para poder hallar esto haremos uso de la siguiente proposición sobre la distribución condicional de un vector multivariado.

Proposición 4.2.1 (Distribución condicional de un vector multivariado)**.** *Consideremos $\mathbf{X} \in \mathbb{R}^d$ un vector aleatorio gaussiano descompuesto como $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)^\top$ con $\mathbf{X}_1 \in \mathbb{R}^p$ y $\mathbf{X}_2 \in \mathbb{R}^{d-p}$. Descompongamos el vector de medias de $\mathbf{X}$ como $\mathbb{E}(\mathbf{X}) = \begin{pmatrix}\boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2\end{pmatrix}$ y a la*

matriz de covarianzas como $\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}$. *Luego, la distribución del vector* $\mathbf{X}_1$ *condicionada a* $\mathbf{X}_2 = \mathbf{x}_2$ *es*

$$\mathbf{X}_1|\mathbf{X}_2 = \mathbf{x}_2 \sim \mathcal{N}(\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{+}\left(\mathbf{x}_2 - \boldsymbol{\mu}_2\right), \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{+}\boldsymbol{\Sigma}_{21}^{\top}),$$

donde $\boldsymbol{\Sigma}_{22}^{+}$ *es la inversa generalizada de* $\boldsymbol{\Sigma}_{22}$.

Demostración. Ver [20]. ■

La proposición (4.2.1) nos dice que $\mathbf{x}_N|\mathbf{x}_O \sim \mathcal{N}\left(\boldsymbol{\mu}_{N|O}, \boldsymbol{\Sigma}_{N|O}\right)$ con

$$\boldsymbol{\mu}_{N|O} = \boldsymbol{\mu}_N + \boldsymbol{\Sigma}_{NO}\boldsymbol{\Sigma}_{OO}^{+}\left(\mathbf{x}_O - \boldsymbol{\mu}_O\right) \tag{4.2.3}$$

y

$$\boldsymbol{\Sigma}_{U|O} = \boldsymbol{\Sigma}_{NN} - \boldsymbol{\Sigma}_{NO}\boldsymbol{\Sigma}_{OO}^{+}\boldsymbol{\Sigma}_{ON}. \tag{4.2.4}$$

Luego, combinando la ecuación (4.2.3) y la proposición (4.1.2) resolvemos que el vector no observado es

$$\mathbf{x}_N = \arg\max_{\mathbf{z}} p_\theta(\mathbf{z}|\mathbf{x}_O) = \boldsymbol{\mu}_{N|O} = \boldsymbol{\mu}_N + \boldsymbol{\Sigma}_{NO}\boldsymbol{\Sigma}_{OO}^{+}\left(\mathbf{x}_O - \boldsymbol{\mu}_O\right). \tag{4.2.5}$$

Podemos completar entonces un punto en el paso k como $\mathbf{x}^{(k)} = \mathbf{P}\begin{pmatrix} \boldsymbol{\mu}_{N|O} \\ \mathbf{x}_O. \end{pmatrix}$. Resta ahora resolver (4.2.2). Observemos que una vez completadas las muestras $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, la log-verosimilitud está totalmente especificada. Podemos entonces, hallar θ según la proposición (4.1.1).

Podemos entonces construir un algoritmo que llamaremos ***MAP-EM*** por las siglas en inglés *Maximum A Posteriori - Expectation Maximization* que está dado de la siguiente forma:

(1) Se toma como entrada del algoritmo una matriz $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{n\times d}$ y un parámetro $\varepsilon > 0$.

(2) Se extrae Ω, el conjunto de entradas no faltantes de la matriz $\mathbf{X}$.

(3) Se transforma $\mathbf{X}_{ij} \leftarrow 0$ si $(i, j) \notin \Omega$.

(4) Se inicializa

$$\boldsymbol{\mu}_{(0)} \leftarrow \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i \quad \text{y} \quad \boldsymbol{\Sigma}_{(0)} \leftarrow \frac{1}{n}\sum_{i=1}(\mathbf{x}_i - \boldsymbol{\mu}_{(1)})(\mathbf{x}_i - \boldsymbol{\mu}_{(1)})^{\top}$$

(5) Para cada $\mathbf{x}_i$ se construye una matriz $\mathbf{P}_i$ de permutaciones que ordena las entradas de la observación $\mathbf{x}_i$ de tal forma que sus entradas no observadas aparezcan primero:

$$\begin{pmatrix} \mathbf{x}_N^i \\ \mathbf{x}_O^i \end{pmatrix} = \mathbf{P}_i\mathbf{x}_i, \quad \begin{pmatrix} \boldsymbol{\mu}_N^i \\ \boldsymbol{\mu}_O^i \end{pmatrix} = \mathbf{P}_i\boldsymbol{\mu}_{\mathbf{x}}, \quad \text{y} \quad \begin{pmatrix} \boldsymbol{\Sigma}_{NN}^i & \boldsymbol{\Sigma}_{NO}^i \\ \boldsymbol{\Sigma}_{ON}^i & \boldsymbol{\Sigma}_{OO}^i \end{pmatrix} = \mathbf{P}_i\boldsymbol{\Sigma}_{\mathbf{x}}\mathbf{P}_i^{\top},$$

(6) Se construyen dos sucesiones $\{\boldsymbol{\mu}_{(k)}\}_{k\in\mathbb{N}}$ y $\{\boldsymbol{\Sigma}_{(k)}\}_{k\in\mathbb{N}}$ de la siguiente forma. Mientras se cumpla que la diferencia entre dos evaluaciones de la log-verosimilitud definida en (4.1.4) difiera más de una tolerancia ε, es decir, mientras

$$|\ell(\boldsymbol{\mu}_{(k)}, \boldsymbol{\Sigma}_{(k)}) - \ell(\boldsymbol{\mu}_{(k-1)}, \boldsymbol{\Sigma}_{(k-1)})| > \varepsilon,$$

se actualiza el valor de $\mathbf{x}_i$ como

$$\mathbf{x}_i \leftarrow \mathbf{P}_i^\top \begin{pmatrix} \boldsymbol{\mu}_N^i + \boldsymbol{\Sigma}_{NO}^i (\boldsymbol{\Sigma}_{OO}^i)^+ (\mathbf{x}_O^i - \boldsymbol{\mu}_O^i) \\ \mathbf{x}_O^i \end{pmatrix} \tag{4.2.6}$$

y, luego,

$$\boldsymbol{\mu}_{(k)} \leftarrow \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \quad \text{y} \quad \boldsymbol{\Sigma}_{(k)} \leftarrow \frac{1}{n}\sum_{i=1} (\mathbf{x}_i - \boldsymbol{\mu}_{(k)})(\mathbf{x}_i - \boldsymbol{\mu}_{(k)})^\top. \tag{4.2.7}$$

(7) El algoritmo devuelve $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{\Sigma}}$ y la matriz $\tilde{\mathbf{X}}$ sin entradas faltantes. Los parámetros $\mathbf{U}$ y σ^2 pueden ser estimados según la proposición (4.1.1).

4.2.2. Análisis e implementación del algoritmo MAP-EM

Para cada una de las observaciones y en cada iteración del algoritmo descrito en la sección anterior, se debe calcular el término $\boldsymbol{\Sigma}_{OO}^+$. En la implementación de este algoritmo construimos una función llamada `ginv()` en R que calcula la pseudoinversa de una matriz dada. Este cálculo es realizado a través de la SVD de $\mathbf{X}$. La justificación de este cálculo está dada por la siguiente proposición.

Proposición 4.2.2 (Inversa generalizada a través de SVD)**.** *Sea $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \in \mathbb{R}^{m\times n}$ una matriz escrita descompuesta a través de su SVD y de rango r. Luego la pseudoinversa de $\mathbf{X}$ puede ser calculada como*

$$\mathbf{X}^+ = \mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top, \quad \boldsymbol{\Sigma}^+ = \operatorname{diag}\{\sigma_1^{-1}, \ldots, \sigma_r^{-1}, 0, \ldots, 0\}. \tag{4.2.8}$$

Demostración. Probemos que la matriz $\mathbf{Y} = \mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top$ cumple las propiedades que definen a la pseudoinversa de $\mathbf{X}$. Tenemos que

$$\begin{aligned}
\mathbf{XYX} &= \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^+\boldsymbol{\Sigma}\mathbf{V}^\top = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top = \mathbf{X}, \\
\mathbf{YXY} &= \mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top = \mathbf{V}\boldsymbol{\Sigma}^+\boldsymbol{\Sigma}\boldsymbol{\Sigma}^+\mathbf{U}^\top = \mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top = \mathbf{Y}, \\
(\mathbf{XY})^\top &= (\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top)^\top = \mathbf{U}(\boldsymbol{\Sigma}^+)^\top\mathbf{V}^\top\mathbf{V}\boldsymbol{\Sigma}^\top\mathbf{U}^\top = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^+\mathbf{U}^\top = \mathbf{XY}, y \\
(\mathbf{YX})^\top &= (\mathbf{V}\boldsymbol{\Sigma}^+\mathbf{U}^\top\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top)^\top = \mathbf{V}\boldsymbol{\Sigma}^\top\mathbf{U}\mathbf{U}^\top(\boldsymbol{\Sigma}^+)^\top\mathbf{V}^\top = \mathbf{V}\boldsymbol{\Sigma}^+\boldsymbol{\Sigma}\mathbf{V} = \mathbf{YX}.
\end{aligned}$$

Luego, por la unicidad de la pseudoinversa, $\mathbf{X}^+ = \mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^\top$. ■

La complejidad del algoritmo implementado en la función `ginv()` que resuelve la pseudoinversa de una matriz está íntimamente ligada, por tanto, a la función que utilicemos para calcular la SVD de $\mathbf{X}$. Realicemos un cálculo aproximado de la complejidad de este algoritmo.

En nuestra implementación utilizamos la función `propack.svd()` del paquete `svd` de R. Esta función utiliza el algoritmo llamado bidiagonalización de Lanczos con reortogonalización parcial (ver [21]). Desafortunadamente, no es posible obtener un valor estimado de la complejidad de este algoritmo pero para entender la magnitud del número de operaciones supongamos que en nuestra implementación utilizamos el algoritmo que en [22] se llama R-SVD. En ese texto se estima que dada una matriz $\mathbf{X} \in \mathbb{R}^{m\times n}$, devolver $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}$ tiene un orden $O(6m^2n + 22n^3)$ con el algoritmo previamente mencionado. La función `ginv()` calcula la SVD de una matriz dada y realiza la multiplicación expuesta en la ecuación (4.2.8) para calcular la pseudoinversa de $\mathbf{X}$. El orden de complejidad de la multiplicación de dos matrices $\mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{B} \in \mathbb{R}^{n\times p}$ es $O(mnp)$. Por otra parte, dada una matriz $\mathbf{X} \in \mathbb{R}^{m\times n}$ con $q = \min\{m, n\}$, la función que calcula la SVD en R devuelve $(\mathbf{U}, \mathbf{\Sigma}, \mathbf{V})$ con $\mathbf{U} \in \mathbb{R}^{m\times q}$, $\mathbf{\Sigma} \in \mathbb{R}^{q\times q}$ y $\mathbf{V} \in \mathbb{R}^{q\times n}$. Por lo tanto, la función `ginv()` tiene una complejidad aproximada de $O(6m^2n + 22n^3 + mq^2 + q^2n)$.

El algoritmo MAP-EM entonces presenta un cuello de botella computacional al calcular $\mathbf{\Sigma}^{+}_{OO}$. Para ejemplificar, si una observación $\mathbf{x}_i \in \mathbb{R}^d$ presenta aproximadamente la mitad de las entradas faltantes, entonces la matriz $\mathbf{\Sigma}_{OO} \in \mathbb{R}^{(\frac{1}{2}d)\times(\frac{1}{2}d)}$. Por lo tanto, en cada iteración se deberán realizar una cantidad de operaciones en el orden de $\frac{19}{8}d^3$.

En cada paso del algoritmo los puntos $\mathbf{x}_i$ son actualizados de manera independiente entre sí. Esto se debe a que, de acuerdo con (4.2.6), los puntos solo son modificados por los valores observados y no observados de $\boldsymbol{\mu}$ y $\mathbf{\Sigma}$ y recordemos que estos últimos, a su vez, sólo dependen de cuáles entradas son conocidas y cuáles no en el punto $\mathbf{x}_i$. Por esta razón y para mejorar el tiempo de ejecución, implementamos en R una versión multihilo del algoritmo MAP-EM. Esta implementación fue realizada a través del paquete `parallel` que permite la construcción de algoritmos multihilo.

Capítulo 5

Completación de matrices

En este capítulo el enfoque no será el de reconstruir n puntos de dimensión en d con entradas faltantes a través de un espacio de dimensión menor. Por el contrario, estaremos interesados en estudiar cómo completar los puntos (es decir, cómo conocer sus entradas faltantes), a partir de las entradas conocidas. El desarrollo de este capítulo está basado en el artículo [23] junto con las modificaciones propuestas en el libro [17].

A diferencia de los capítulos anteriores notaremos con $\mathbf{M} \in \mathbb{R}^{d\times n}$ a la matriz con n observaciones $\{\mathbf{m}_1, \ldots, \mathbf{m}_n\} \subset \mathbb{R}^d$ pensadas como vectores columna. Consideremos a la matriz, $\mathbf{W} = (w_{ij}) \in \mathbb{R}^{d\times n}$, construida en el capítulo (3), como la matriz que cumple que

$$w_{ij} = \begin{cases} 1 & \text{si la entrada } (i,j) \text{ de } \mathbf{M} \text{ es conocida} \\ 0 & \text{si no lo es} \end{cases}. \tag{5.0.1}$$

La idea será la de reconstruir las entradas faltantes de $\mathbf{M}$ de la forma más *parsimoniosa* posible. Una forma de plasmar esta idea es a través de la función $\text{rango}()$. El problema que plantearemos es

$$\begin{aligned} &\underset{\mathbf{X}\in\mathbb{R}^{m\times n}}{\text{minimizar}} && \text{rango}(\mathbf{X}) \\ &\text{sujeto a} && \mathbf{W}\odot\mathbf{X} = \mathbf{W}\odot\mathbf{M} \end{aligned} \tag{5.0.2}$$

Es decir, queremos recuperar la estructura de la matriz $\mathbf{M}$ a través de una matriz $\mathbf{X}$ con una estructura *simple* (en el sentido del rango). El problema (5.0.2) presenta, sin embargo, una dificultad importante: la función rango$(\cdot)$ no es una función convexa y por lo tanto no podemos garantizar la globalidad del problema. Estudiemos con un ejemplo sencillo que la función rango no es convexa.

Ejemplo 5.0.1. Consideremos las matrices

$$\mathbf{A} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{y} \quad \mathbf{B} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

y $t \in (0,1)$. Como

$$t\,\text{rango}(\mathbf{A}) + (1-t)\,\text{rango}(\mathbf{B}) = 1$$

y

$$\text{rango}[t\mathbf{A} + (1-t)\mathbf{B})] = \text{rango}\left[\begin{pmatrix} t & 0 \\ 0 & 1-t \end{pmatrix}\right] = 2,$$

se concluye que rango$(\cdot)$ no es convexa.

Al margen de las características de la función $\text{rango}()$, existe una dificultad práctica en la resolución exacta del problema (5.0.2): de acuerdo con [24], la complejidad de todos los algoritmos conocidos crecen de forma exponencial con la matriz $\mathbf{M}$.

Una solución a este problema, propuesta por Maryam Fazel en su tesis de doctorado (ver [25]) es utilizar *relajación convexa*. Es decir, considerar un problema que sea equivalente en algún sentido al problema (5.0.2) pero que, además, sea convexo. Precisemos, a continuación, estos conceptos.

Definición 5.0.1 (Envolvente convexa)**.** Sean V un espacio vectorial, $E \subset V$ un conjunto convexo y $f : E \to \mathbb{R}$. Decimos que $f_{\text{env}} : E \to \mathbb{R}$ es la ***envolvente convexa*** de f si

$$f_{\text{env}}(x) = \sup\{h : E \to \mathbb{R} : h \text{ es convexa y } h(x) \leq f(x)\}.$$

Al estudiar la envolvente convexa de la función rango surge la llamada *norma núclear de una matriz.*

Definición 5.0.2 (Norma nuclear)**.** Sea $\mathbf{X} \in \mathbb{R}^{m\times n}$ una matriz escrita según la SVD

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top,$$

con $\mathbf{U} \in \mathbb{R}^{m\times r}$, $\mathbf{V} \in \mathbb{R}^{n\times r}$ $\mathbf{\Sigma} = \text{diag}(\{\sigma_1, \ldots, \sigma_r\})$, con $r = \text{mín}\{m, n\}$ y $\mathbf{U}^\top\mathbf{U} = \mathbf{V}^\top\mathbf{V} = \mathbf{I}_r$. Definimos la ***norma nuclear*** de $\mathbf{X}$ como

$$\|\mathbf{X}\|_* = \sum_{i=1}^{\text{mín}\{m,n\}} \sigma_i.$$

El vínculo entre rango y norma nuclear está dado por la siguiente proposición, que presentaremos sin demostración.

Proposición 5.0.1. *Sea $\mathcal{S} = \{\mathbf{X} \in \mathbb{R}^{m\times n} : \|\mathbf{X}\|_2 \leq 1\}$ el espacio de matrices acotadas. La envolvente convexa de la función rango $: \mathcal{S} \to \mathbb{N}$ es la norma nuclear.*

Demostración. Ver [25], página 56. ■

Consideremos $\Omega \subset \{1, \ldots, d\} \times \{1, \ldots, n\}$ el conjunto de índices de la matriz $\mathbf{M}$ asociados a las entradas conocidas. Siguiendo la notación de [23], notaremos con $\mathcal{P}_\Omega$ al operador tal que $\mathcal{P}_\Omega(\mathbf{X}) = \mathbf{W} \odot \mathbf{X}$. Proponemos entonces el siguiente problema de optimización:

$$(\text{P}) \quad \begin{array}{ll} \underset{\mathbf{X}\in\mathbb{R}^{d\times n}}{\text{minimizar}} & \|\mathbf{X}\|_* \\ \text{sujeto a} & \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{M}) \end{array} \tag{5.0.3}$$

En la siguiente sección realizaremos el desarrollo para construir un algoritmo iterativo que aproxima una solución del problema (5.0.3) de acuerdo con [23].

5.1. Construcción del algoritmo *Singular value thresholding*

Definimos $f_\tau(\mathbf{X}) = \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X}\|_F^2$ para todo $\tau > 0$. Queremos resolver

$$(\text{PProx}) \quad \begin{array}{ll} \underset{\mathbf{X}\in\mathbb{R}^{d\times n}}{\text{minimizar}} & \tau\|\mathbf{X}\|_* + \dfrac{1}{2}\|\mathbf{X}\|_F^2 \\ \text{sujeto a} & \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{M}) \end{array} \tag{5.1.1}$$

La vinculación entre el problema (PProx) y el problema (P) está dada por el siguiente teorema, cuya demostración es una adaptación de la demostración realizada en [23].

Teorema 5.1.1. *Sea* $\mathbf{X}_\tau^\star$ *la solución de* (PProx) *y* $\mathbf{X}_\infty$ *definido como*

$$\mathbf{X}_\infty \coloneqq \arg\min_{\mathbf{X}} \{\|\mathbf{X}\|_F : \mathbf{X} \text{ es solución de } (5.0.3)\}.$$

Luego, $\lim_{\tau\to\infty} \|\mathbf{X}_\tau^\star - \mathbf{X}_\infty\|_F = 0$.

Demostración. Para probar el límite de la tesis, consideraremos una sucesión $\{\mathbf{X}_{\tau_n}^\star\}_{n\in\mathbb{N}}$ arbitraria pero convergente en norma Frobenius y probaremos que este límite es $\mathbf{X}_\infty$.

Consideremos $\{\mathbf{X}_{\tau_n}^\star\}_{n\in\mathbb{N}}$ una sucesión convergente y definamos $\mathbf{X}_\ell = \lim_{n\to\infty} \mathbf{X}_{t_n}^\star$. Probaremos que $\mathbf{X}_\ell$ es solución (5.0.3). Probemos, en primer lugar que $\mathbf{X}_\ell$ cumple las restricciones del problema (5.0.3). Para todo $n \in \mathbb{N}$ tenemos que $\mathcal{P}_\Omega(\mathbf{X}_{\tau_n}^\star) = \mathcal{P}_\Omega(\mathbf{M})$ y como $\mathcal{P}_\Omega : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ es una función continua tenemos que

$$\mathcal{P}_\Omega(\mathbf{X}_\ell) = \mathcal{P}_\Omega(\lim_{n\to\infty} \mathbf{X}_{t_n}^\star) = \lim_{n\to\infty} \mathcal{P}_\Omega(\mathbf{X}_{t_n}^\star) = \mathcal{P}_\Omega(\mathbf{M}).$$

Probemos, a continuación, que $\mathbf{X}_\ell$ minimiza la norma nuclear. Para esto observemos que para todo $\tau > 0$ tenemos, por la definición de $\mathbf{X}_\tau^\star$, que

$$\tau \|\mathbf{X}_\tau^\star\|_* + \frac{1}{2}\|\mathbf{X}_\tau^\star\|_F^2 \le \tau \|\mathbf{X}_\infty\|_* + \frac{1}{2}\|\mathbf{X}_\infty\|_F^2$$

y, dividiendo por τ,

$$\|\mathbf{X}_\tau^\star\|_* + \frac{1}{2\tau}\|\mathbf{X}_\tau^\star\|_F^2 \le \|\mathbf{X}_\infty\|_* + \frac{1}{2\tau}\|\mathbf{X}_\infty\|_F^2. \tag{5.1.2}$$

Además, como $\mathbf{X}_\infty$ es solución de (5.0.3), tenemos que

$$\|\mathbf{X}_\infty\|_* \le \|\mathbf{X}_\tau^\star\|_*. \tag{5.1.3}$$

Combinando las desigualdades (5.1.2) y (5.1.3) obtenemos que

$$\|\mathbf{X}_\tau^\star\|_* + \frac{1}{2\tau}\|\mathbf{X}_\tau^\star\|_F^2 \le \|\mathbf{X}_\infty\|_* + \frac{1}{2\tau}\|\mathbf{X}_\infty\|_F^2 \le \|\mathbf{X}_\tau^\star\|_* + \frac{1}{2\tau}\|\mathbf{X}_\infty\|_F^2$$

y, por lo tanto,

$$\|\mathbf{X}_\tau^\star\|_F^2 \le \|\mathbf{X}_\infty\|_F^2, \tag{5.1.4}$$

para todo τ. Luego, aplicando las desigualdades (5.1.2) y (5.1.4) obtenemos

$$\limsup_{n\to\infty} \left\|\mathbf{X}_{\tau_n}^\star\right\|_* \le \|\mathbf{X}_\infty\|_*.$$

y, por la desigualdad (5.1.3),

$$\|\mathbf{X}_\infty\|_* \le \liminf_{n\to\infty} \left\|\mathbf{X}_{\tau_n}^\star\right\|_*.$$

Por lo tanto, $\lim_{n\to\infty} \left\|\mathbf{X}_{\tau_n}^\star\right\|_* = \|\mathbf{X}_\infty\|_*$. Luego, como las normas son funciones continuas, obtenemos que $\|\mathbf{X}_\ell\|_* = \|\mathbf{X}_\infty\|_*$. Por definición de $\mathbf{X}_\infty$ concluimos que $\mathbf{X}_\ell$ es solución de (5.0.3). Resta probar que $\mathbf{X}_\ell = \mathbf{X}_\infty$.

Por definición de $\mathbf{X}_\infty$ tenemos que $\|\mathbf{X}_\ell\|_F \ge \|\mathbf{X}_\infty\|_F$ mientras que por la desigualdad (5.1.4) tenemos que $\|\mathbf{X}_\ell\|_F \le \|\mathbf{X}_\infty\|_F$. Concluimos que $\|\mathbf{X}_\ell\|_F = \|\mathbf{X}_\infty\|_F$ y, luego, por la definición de X_∞ concluimos que $\mathbf{X}_\ell = \mathbf{X}_\infty$. ■

El objetivo, a continuación será el de resolver el problema (PProx) mediante un algoritmo iterativo. En primer lugar observemos que la función

$$f_\tau(\mathbf{X}) = \tau \|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X}\|_F^2$$

es una función convexa para todo $\tau > 0$. Esto se debe a que $\|\cdot\|_*$ es una norma (ver proposición D.3.1)), las normas son funciones convexas (ver proposición (D.1.1)) y f_τ es una combinación lineal con coeficientes positivos de funciones convexas (ver proposición (D.1.2)).

Estamos en condiciones entonces de construir un algoritmo que resuelva de forma iterativa el problema (5.1.1).

El lagrangiano del problema (PProx) es $\mathcal{L}(\mathbf{X}, \mathbf{Z}) = f_\tau(\mathbf{X}) + \mathbf{Z}^\top[\mathcal{P}_\Omega(\mathbf{M}) - \mathcal{P}_\Omega(\mathbf{X})]$ donde $\mathbf{Z} \in \mathbb{R}^{m\times n}$ es una matriz de multiplicadores de Lagrange. Como el problema (P) es un problema convexo, se cumple dualidad fuerte y existe un par $(\mathbf{X}^\star, \mathbf{Z}^\star)$ que cumple la desigualdad *min-max* (ver [26], página 237):

$$\sup_{\mathbf{Z}\succeq 0} \inf_{\mathbf{X}\in\mathbb{R}^{m\times n}} \mathcal{L}(\mathbf{X}, \mathbf{Z}) = \mathcal{L}(\mathbf{X}^\star, \mathbf{Z}^\star) = \inf_{\mathbf{X}\in\mathbb{R}^{m\times n}} \sup_{\mathbf{Z}\succeq 0} \mathcal{L}(\mathbf{X}, \mathbf{Z}).$$

Definimos $g(\mathbf{Z}) = \inf_{\mathbf{X}\in\mathbb{R}^{m\times n}} \mathcal{L}(\mathbf{X}, \mathbf{Z})$ la función dual del problema (P). Notemos que si conociéramos $\mathbf{X}^\star \in \arg\min_{\mathbf{X}\in\mathbb{R}^{m\times n}} \mathcal{L}(\mathbf{X}, \mathbf{Z})$ tendríamos que

$$\frac{\partial}{\partial \mathbf{Z}} g(\mathbf{Z}) = \frac{\partial}{\partial \mathbf{Z}} \mathcal{L}(\mathbf{X}^\star, \mathbf{Z}) = \frac{\partial}{\partial \mathbf{Z}} (f_\tau(\mathbf{X}^\star) + \mathbf{Z}^\top[\mathcal{P}_\Omega(\mathbf{M}) - \mathcal{P}_\Omega(\mathbf{X}^\star)]) = \mathcal{P}_\Omega(\mathbf{M}) - \mathcal{P}_\Omega(\mathbf{X}^\star).$$

Podríamos entonces plantear una iteración de gradiente ascendente para encontrar $\mathbf{Z}^\star$ de la forma

$$\mathbf{Z}_{(k)} = \mathbf{Z}_{(k-1)} + \delta_k \frac{\partial}{\partial \mathbf{Z}} g(\mathbf{Z}_{(k-1)}) = \mathbf{Z}_{(k-1)} + \delta_k[\mathcal{P}_\Omega(\mathbf{M}) - \mathcal{P}_\Omega(\mathbf{X}^\star)],$$

donde $\{\delta_k\}_{k\geq 1}$ es una sucesión de pasos positivos.

Por otra parte, tomemos $\mathbf{Z}$ fijo. Observemos que

$$\begin{aligned}
\mathcal{L}(\mathbf{X}, \mathbf{Z}) &= f_\tau(\mathbf{X}) + \mathbf{Z}^\top[\mathcal{P}_\Omega(\mathbf{M}) - \mathcal{P}_\Omega(\mathbf{X})] \\
&= f_\tau(\mathbf{X}) + \mathbf{Z}^\top[\mathcal{P}_\Omega(\mathbf{M} - \mathbf{X})] \\
&= f_\tau(\mathbf{X}) + \mathcal{P}_\Omega(\mathbf{Z})^\top(\mathbf{M} - \mathbf{X}) \\
&= \tau\|\mathbf{X}\|_* + \frac{1}{2}\mathbf{X}^\top\mathbf{X} + \mathcal{P}_\Omega(\mathbf{Z})^\top\mathbf{M} - \mathcal{P}_\Omega(\mathbf{Z})^\top\mathbf{X} \\
&\quad + \frac{1}{2}\mathcal{P}_\Omega(\mathbf{Z})\mathcal{P}_\Omega(\mathbf{Z})^\top - \frac{1}{2}\mathcal{P}_\Omega(\mathbf{Z})\mathcal{P}_\Omega(\mathbf{Z})^\top \\
&= \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \mathcal{P}_\Omega(\mathbf{Z})\|_F^2 + \mathcal{P}_\Omega(\mathbf{Z})^\top\mathbf{M} + \frac{1}{2}\mathcal{P}_\Omega(\mathbf{Z})\mathcal{P}_\Omega(\mathbf{Z})^\top
\end{aligned}$$

Luego

$$\arg\min_{\mathbf{X}} \mathcal{L}(\mathbf{X}, \mathbf{Z}) = \arg\min_{\mathbf{X}} \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \mathcal{P}_\Omega(\mathbf{Z})\|_F^2. \tag{5.1.5}$$

Para poder hallar (5.1.5) necesitaremos de la siguiente definición:

Definición 5.1.1 (Operador soft-thresholding)**.** Sea $\mathbf{X} \in \mathbb{R}^{n\times m}$ una matriz de rango r. Consideremos su descomposición compacta en valores singulares

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top,$$

donde $\mathbf{U}$ y $\mathbf{V}$ son matrices de tamaño $n \times r$ y $r \times m$, respectivamente, y que cumplen que $\mathbf{U}^\top\mathbf{U} = \mathbf{V}^\top\mathbf{V} = \mathbf{I}_r$, y $\mathbf{\Sigma} = \operatorname{diag}(\{\sigma_1, \ldots, \sigma_r\})$ donde $\{\sigma_i\}_{i=1}^r$ son los valores singulares positivos de la matrix $\mathbf{X}$. Para cada $\tau \geq 0$ definimos el operador **soft-thresholding** $\mathcal{D}_\tau$ tal que $\mathcal{D}_\tau : \mathbb{R}^{n\times m} \to \mathbb{R}^{n\times m}$ y

$$\mathcal{D}_\tau(\mathbf{X}) = \mathbf{U}\mathcal{D}_\tau(\mathbf{\Sigma})\mathbf{V}^\top, \quad \mathcal{D}_\tau(\mathbf{\Sigma}) = \operatorname{diag}((\sigma - \tau)_+) \tag{5.1.6}$$

donde $(\cdot)_+ = \text{máx}\{0, \cdot\}$.

El vínculo entre el operador *soft-thresholding* y la ecuación (5.1.5) lo da la siguiente proposición:

Proposición 5.1.1. *Para cada $\tau \geq 0$ y $\mathbf{Y} \in \mathbb{R}^{m\times n}$, $\mathcal{D}_\tau$ cumple que*

$$\mathcal{D}_\tau(\mathbf{Y}) = \arg\min_{\mathbf{X}} \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \mathbf{Y}\|_F^2 \tag{5.1.7}$$

Para poder dar una prueba de esta proposición necesitaremos la forma del subdiferencial de la norma nuclear. Esto está dado en la siguiente proposición que presentaremos sin demostración.

Proposición 5.1.2. *Dada una matriz $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ escrita a través de su SVD, el subdiferencial de la norma nuclear está dado por*

$$\partial\|\mathbf{X}\|_* = \{\mathbf{U}\mathbf{V}^\top + \mathbf{W} : \mathbf{W} \in \mathbb{R}^{m\times n}, \mathbf{U}^\top\mathbf{W} = 0, \mathbf{W}\mathbf{V} = 0, \sigma_1(\mathbf{W}) \leq 1\}$$

Demostración. Ver [27], página 40. ■

Prueba del teorema (5.1.1). A través de las proposiciones (D.3.1) y (D.1.2) probamos que

$$h(\mathbf{X}) = \frac{1}{2}\|\mathbf{X} - \mathbf{Y}\|_F^2 + \tau\|\mathbf{X}\|_*$$

es una función convexa. Por lo tanto, si existe el mínimo de h, este es único. Vamos a probar que este mínimo es igual a $\mathcal{D}_\tau(\mathbf{Y})$. Si $\mathbf{X}^\star$ minimiza a la función h en $\mathbb{R}^{m\times n}$, por la proposición (D.2.4) tenemos que

$$\mathbf{0} \in \partial h(\mathbf{X}^\star). \tag{5.1.8}$$

Por la proposición (D.2.2) tenemos que

$$\partial h(\mathbf{X}^\star) = \partial f(\mathbf{X}^\star) + \partial g(\mathbf{X}^\star),$$

donde $f(\mathbf{X}) = \frac{1}{2}\|\mathbf{X} - \mathbf{Y}\|_F^2$ y $g(\mathbf{X}) = \tau\|\mathbf{X}\|_*$. Por la proposición (D.2.3) y utilizando que f es diferenciable se tiene que

$$\partial f(\mathbf{X}^\star) = \{\nabla f(\mathbf{X}^\star)\} = \{\mathbf{X}^\star - \mathbf{Y}\}. \tag{5.1.9}$$

Dada una matriz arbitraria $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, por la proposición (5.1.2) tenemos que

$$\partial\|\mathbf{X}\|_* = \{\mathbf{U}\mathbf{V}^\top + \mathbf{W} : \mathbf{W} \in \mathbb{R}^{m\times n}, \mathbf{U}^\top\mathbf{W} = 0, \mathbf{W}\mathbf{V} = 0, \sigma_1(\mathbf{W}) \leq 1\}. \tag{5.1.10}$$

Combinando las ecuaciones (5.1.8), (5.1.9) y (5.1.10), resta probar que

$$\mathbf{X}^\star - \mathbf{Y} \in \partial \left\|\mathbf{X}\right\|_*,$$

con $\mathbf{X}^\star = \mathcal{D}_\tau(\mathbf{Y})$. Para esto, construimos la descomposición en valores singulares de $\mathbf{Y}$ como $\mathbf{Y} = \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^\top + \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^\top$ donde $\mathbf{U}_0, \mathbf{V}_0$ son los vectores singulares asociados a los valores singulares mayores que τ y $\mathbf{U}_1, \mathbf{V}_1$ son los vectores singulares asociados a los valores singulares menores o iguales que τ. Con esta notación tenemos que $\mathcal{D}_\tau(\mathbf{Y}) = \mathbf{U}_0(\mathbf{\Sigma}_0 - \tau\mathbf{I})\mathbf{V}_0^\top$ y luego

$$\begin{aligned}\mathbf{X}^\star - \mathbf{Y} &= \mathbf{U}_0(\mathbf{\Sigma}_0 - \tau\mathbf{I})\mathbf{V}_0^\top - \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^\top + \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^\top \\ &= -\tau\mathbf{U}_0\mathbf{V}_0^\top + \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^\top \\ &= -\tau(\mathbf{U}_0\mathbf{V}_0^\top + \mathbf{W}),\end{aligned}$$

con $\mathbf{W} = \frac{1}{\tau}\mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^\top$. Por la construcción de la SVD, $\mathbf{U}_0^\top\mathbf{W} = 0$ y $\mathbf{W}\mathbf{V}_0 = 0$. Además, por construcción, $\sigma(\mathbf{W}) \leq 1$. Concluimos, que $\mathbf{X}^\star - \mathbf{Y} \in \partial\tau \left\|\mathbf{X}\right\|_*$, como queríamos probar. ■

Esto conduce al siguiente algoritmo para hallar el par $(\mathbf{X}^\star, \mathbf{Z}^\star)$ óptimos:

Sea $\tau > 0$ fijo, una sucesión $\{\delta_k\}_{k\geq 0}$ de pasos positivos y $\mathbf{Z}_0$ una matriz de multiplicadores de Lagrange inicial. Construimos una sucesión $\{(\mathbf{X}_{(k)}, \mathbf{Z}_{(k)})\}_{k\geq 0}$ como

$$\begin{cases}\mathbf{X}_{(k)} = \mathcal{D}_\tau(\mathbf{Z}_{(k-1)}) \\ \mathbf{Z}_{(k)} = \mathbf{Z}_{(k-1)} + \delta_k[\mathcal{P}_\Omega(\mathbf{M}) - \mathcal{P}_\Omega(\mathbf{X}_{(k)})]\end{cases} \tag{5.1.11}$$

A la iteración previamente descrita la llamaremos ***SVTC*** por *singular value thresholding completion*.

5.2. Análisis e implementación del algoritmo

La convergencia analítica del algoritmo detallado en (5.1.11) está garantizada por la siguiente proposición.

Proposición 5.2.1. *Sea $\{\delta_k\}_{k\in\mathbb{N}}$ una sucesión de números reales positivos que cumplen que $0 < \inf_k \delta_k \leq \sup_k \delta_k < 2$. Luego, la sucesión de matrices $\{\mathbf{X}_{(k)}\}_{k\in\mathbb{N}}$ obtenidas a través del algoritmo (5.1.11) con los δ_k previamente especificados, converge a la única solución del problema (5.0.3).*

Demostración. Ver [23], página 1968. ■

Una implementación del algoritmo (5.1.11) en una cantidad de iteraciones razonable debe hacerse con cautela. Analicemos algunos puntos sobre este algoritmo y cómo fueron atacados en nuestra implementación.

En primer lugar observemos que desde un punto de vista de la complejidad, la complejidad del algoritmo (5.1.11) está dominada por el cálculo del operador *soft-thresholding* de la matriz $\mathbf{Z}_{(k-1)}$. Esto implica que en cada iteración del algoritmo se deberá calcular la SVD de una matriz con lo que esto implica a nivel computacional (recordemos el análisis realizado para el algoritmo MAP-EM en la sección (4.2.2)). En nuestro caso, la implementación del operador $\mathcal{D}_\tau$ que realizamos está basada en la función `propack.svd()`

en R. Esta implementación no es óptima puesto que `propack.svd()` calcula todos los valores singulares de la matriz. Por lo tanto, para valores de τ *grandes* y/o para δ_k *chicos*, existirán varias iteraciones del algoritmo en donde todos los valores singulares de $\mathbf{Z}$ serán menores que τ. Una forma de remediar este problema es a través de la utilización de formas iterativas de obtener los valores singulares de $\mathbf{Z}$: de esta manera, si el valor singular más grande es más chico que τ, no seguiremos calculando el resto.

En segundo lugar y vinculado al punto anterior, observemos que dependiendo de τ, δ, y $\mathbf{Z}_0$ pueden haber decenas (o centenas) de iteraciones del algoritmo donde no se producirá ningún cambio sobre $\mathbf{X}_{(k)}$. Para ejemplificar esta situación consideremos $\delta_k = \delta$ fijo para todo k y que la matriz de multiplicadores inicial es $\mathbf{Z}_{(0)} = \mathbf{0}$. Con estos valores iniciales tendremos que, $\mathbf{X}_{(1)} = \mathbf{0}$ y $\mathbf{Z}_{(1)} = \delta\mathcal{P}_\Omega(\mathbf{M})$. Además, mientras k sea tal que el valor singular más grande de $k\delta\mathcal{P}_\Omega(\mathbf{M}))$ sea menor que τ, $\mathbf{X}_{(k)}$ será igual a $\mathbf{0}$. Si el paso δ es pequeño con respecto a τ y a $\sigma_1(\mathcal{P}_\Omega(\mathbf{M}))$ habrá muchas iteraciones virtualmente inútiles. Por esta razón, y siguiendo la implementación de [23], comenzamos con una matriz $\mathbf{Z}_{(0)} = k_0\delta\mathcal{P}_\Omega(\mathbf{M})$ donde k_0 es el natural tal que

$$\frac{\tau}{\delta \left\|\mathcal{P}_\Omega(\mathbf{X})\right\|_2} \in (k_0 - 1, k_0].$$

En la implementación de este algoritmo consideramos como criterio de parada el recomendado por [23]: fijado un umbral $\varepsilon > 0$, detendremos el algoritmo si

$$\frac{\left\|\mathcal{P}_\Omega(\mathbf{X}_{(k)} - \mathbf{M})\right\|_F}{\mathcal{P}_\Omega(\mathbf{M})_F} < \varepsilon. \tag{5.2.1}$$

Los detalles de por qué este es un criterio de parada razonable pueden consultarse en la página 1972 de [23].

Capítulo 6

Comparación de algoritmos

6.1. Simulación de datos de prueba

Con el objetivo de analizar el desempeño de los algoritmos previamente estudiados, simulamos datos genómicos de prueba. El punto de partida para esta simulación es el *Human Genome Diversity Panel* en su versión del 2008 [28]. De los 525 910 SNPs de la base de datos completa extrajimos 49 408 correspondientes al cromosoma 1.

Seleccionamos 148 haplotipos correspondientes a individuos provenientes de tres poblaciones: 48 vascos, 56 japoneses y 42 yorubas. La elección de las poblaciones que llamaremos *de base* se debió a que estas poblaciones están distanciadas a nivel geográfico y esta distancia genera patrones bien reconocibles en las componentes principales [8]. Reconocimos asimismo que los SNPs elegidos del cromosoma 1 son lo suficientemente variables entre las poblaciones seleccionadas como para permitir distinguirlas.

A partir de estos haplotipos construimos vectores de frecuencias alélicas para cada una de la poblaciones bases. Una vez obtenida la frecuencia alélica $\hat{f}_{ij}$ de una la posición j para la población i, esto nos permitió simular nuevos individuos a través de variables aleatorias Bernoulli. Para obtener un individuo $\mathbf{x}$ de la población i, simulamos d variables aleatorias Bernoulli con frecuencias $\{f_{11}, f_{12}, \ldots, f_{1d}\}$.

Los conjuntos de datos simulados pueden ser divididos en dos de acuerdo a la medida que se quiere analizar de cada algoritmo: con uno de ellos se pretende estudiar la capacidad de los algoritmos de recuperar la estructura poblacional y la robustez a datos faltantes; con el otro, buscamos medir la velocidad y la escalabilidad de los algoritmos.

6.2. Robustez a datos faltantes

El objetivo del estudio de las distintas técnicas de PCA es el de encontrar una o más de una de ellas que sea aplicable a datos genómicos reales. Como habíamos observado en la sección (2), un tratamiento naíf de los datos faltantes en el contexto de PCA puede llevar a la construcción de hipótesis erróneas sobre la estructura de las poblaciones. En particular, como habíamos notado en ese capítulo, es relevante reconstruir de forma adecuada el efecto que ocurre en la proyección de individuos con mezcla sobre las primeras componentes principales. Por esta razón construimos distintos casos de prueba que incorporan individuos mezclados y tasas de datos faltantes variadas.

Los distintos casos de prueba que realizamos están inspirados en la construcción de datos de prueba realizada en [29]. En todos los escenarios que detallaremos a continuación simulamos $d = 1000$ SNPs para 100 individuos de cada una de las poblaciones base y de 150 individuos mezclados. Los individuos mezclados siguieron el siguiente esquema: 50 fueron individuos vasco-yorubas, 50 fueron vasco-japoneses y 50 serán yoruba-japoneses. Detallemos cuáles fueron los distintos esquemas de incertidumbre que desarrollamos. Una representación gráfica de estos escenarios puede ser vista en la figura (6.1).

- *Escenario 1*. Consideramos tasas bajas de datos faltantes que fueron variables por individuo y se construyeron sin estructura. Para cada uno de los individuos $\mathbf{x}_j$ se simuló una variable aleatoria, p, con distribución uniforme en $[0{,}05, 0{,}3]$. Luego, se simularon variables aleatorias Bernoulli X_{ij} con parámetro p_j. Este tipo de datos faltantes es el análogo al caso en que se cuente con una base de datos genómica proveniente de genotipado con poca profundidad y errores aleatorios de lectura.

- *Escenario 2*. El escenario 2 es análogo al escenario 1 con la diferencia de que en este se consideraron tasas moderadas de datos faltantes. La construcción resultó equivalente a la previamente descrita para el escenario 1 con la salvedad de que las variables p_j se construyeron uniformes en $[0, 3; 0, 7]$.

- *Escenario 3*. En este caso consideramos datos faltantes variables por individuo pero con estructura de bloque. Para construir los bloques de datos faltantes, para cada $\mathbf{x}_j$ simulamos dos variables aleatorias, p_j y θ_j con distribuciones uniformes en $[0{,}05; 0{,}3]$ y $(0, 1)$, respectivamente. Para cada individuo luego se muestrearon $n_j = \lfloor p_j d \rfloor$ posiciones y se simularon variables aleatorias $X_1^j, \ldots, X_{n_j}^j$ independientes con distribución geométrica de esperanza θ_j^{-1}. El objetivo de considerar este escenario es el de entender si la existencia de bloques de datos faltantes afecta al desempeño de los algoritmos.

- *Escenario 4*. En este esquema consideramos nuevamente datos faltantes por bloque como en el escenario 3 pero tasas diferenciales para cada población. En este caso consideramos variables p_j con distribución uniforme en $[0, 05; 0{,}1]$ para las poblaciones de base y variables p_j uniformes en $[0, 3; 0, 7]$ para las poblaciones mezcladas. El objetivo de este experimento es el de entender qué tan bien reconstruyen los algoritmos las relaciones entre individuos mezclados y las poblaciones ancestrales cuando los individuos mezclados presentan tasas más altas de incertidumbre.

En primer lugar aclararemos que no presentaremos los resultados para el algoritmo PPCA. Las razones de esto son sencillas: no fue posible obtener reconstrucciones de los datos con este algoritmo. Para entender el porqué tenemos que analizar la implementación del algoritmo EM-PPCA. Recordemos que la función objetivo de este problema es

$$\ell(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}}) = -\frac{nd}{2}\ln(2\pi) - \frac{n}{2}\ln(\det(\boldsymbol{\Sigma}_{\mathbf{x}})) - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})^{\top}\boldsymbol{\Sigma}_{\mathbf{x}}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})$$

En cada iteración del algoritmo se estima $\boldsymbol{\mu}_x$ y $\boldsymbol{\Sigma}_x$. Para nuestros datos observamos que la matriz $\widehat{\boldsymbol{\Sigma}}_n$ es numéricamente no invertible por lo que el término $-\frac{n}{2}\ln(\det(\widehat{\boldsymbol{\Sigma}}_n))$ es numéricamente igual a $+\infty$.

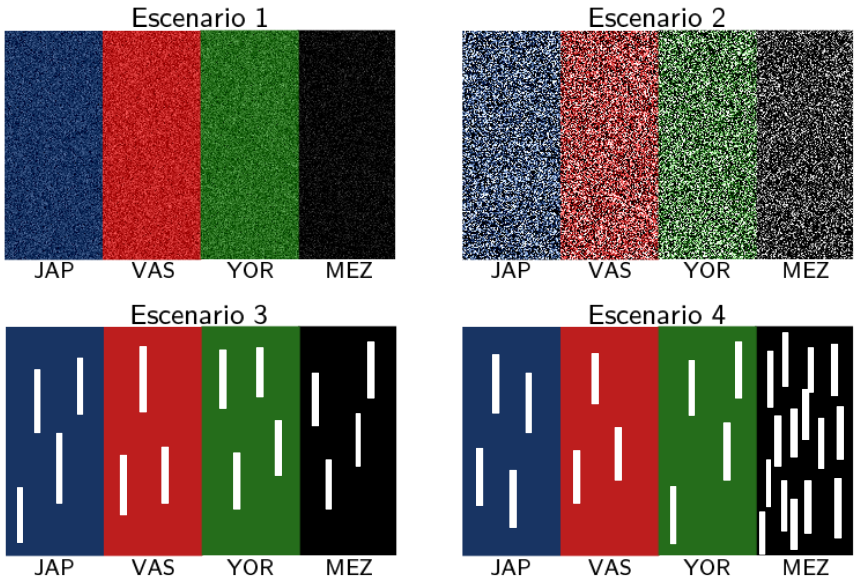

Figura 6.1: Representación gráfica de los escenarios de la sección 6.2. JAP: japonés, VAS: vasco, YOR: yoruba, MEZ: mezcla. Las columnas representan individuos y las filas representan SNPs.

Por otra parte, quitar este último de la función objetivo genera otro problema. El término $-\frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})^\top \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})$ es equivalente, si reemplazamos $\boldsymbol{\Sigma}_{\mathbf{x}}^{-1}$ por $\widehat{\boldsymbol{\Sigma}}_n$ y $\boldsymbol{\mu}_{\mathbf{x}}$ por $\bar{\mathbf{x}}_n$ es

$$-\frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}}_n)^\top \widehat{\boldsymbol{\Sigma}}_n^{-1}(\mathbf{x}_i - \bar{\mathbf{x}}_n) = -\frac{n}{2}\operatorname{tr}(\mathbf{I}_n),$$

donde en esta última ecuación utilizamos el corolario (A.1.1.1). Por lo tanto este término es constante en cada iteración del algoritmo. En conclusión, consideramos que este algoritmo no es apropiado para datos genómicos.

El resultado de la aplicación del resto de los algoritmos a los esquemas previamente detallados puede verse en la figura (6.2) y la figura (F.1) del apéndice.

En la figura (6.2) (A) tenemos representados los puntos originales y su reconstrucción en las dos primeras componentes principales según el teorema (C.0.1). La transparencia de los puntos indica la tasa de datos faltantes del punto considerado: a mayor cantidad de datos faltantes, mayor es la transparencia del punto. Esta figura actúa entonces como contraste para el resto de los algoritmos puesto que esta es la mejor representación de los puntos en dos dimensiones.

En las figuras 6.2 (B) y 6.2 (C) presentamos la reconstrucción que obtenida luego imputar los datos desconocidos por la media y tras utilizar el método SVTC, respectivamente. El desempeño de ambos procedimientos es similar para los escenarios 1 y 3, escenarios para los cuales la tasa de entradas desconocidas es baja. Asimismo, observemos que el desempeño no se ve influenciado por la estructura de los datos faltantes; tanto la imputación por la media como SVTC pueden reconstruir las relaciones entre individuos mientras la incertidumbre sea baja (recordemos que en estos dos escenarios cada individuo tiene una proporción de datos faltantes menor a $0{,}3$).

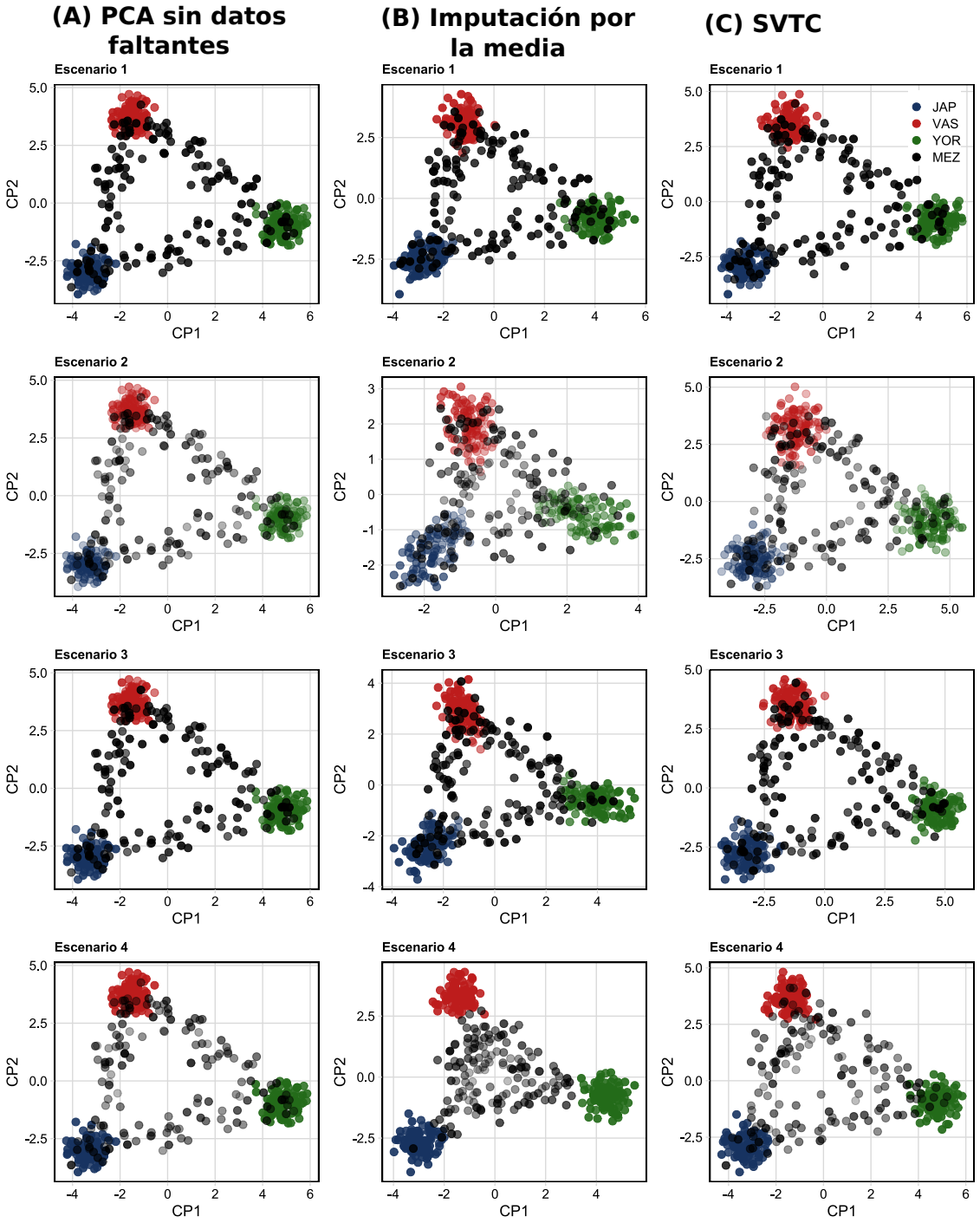

Figura 6.2: Proyección sobre las dos primeras componentes de los haplotipos simulados. (A) Se obtuvieron los puntos a través de la SVD de la matriz original. (B) Se imputaron los datos desconocidos a través de la ecuación (2.0.5). (C) El algoritmo SVTC fue ejecutado hasta la convergencia según el criterio de parada dado en la fórmula (5.2.1) con $\varepsilon = 0{,}05$. La iteración fue realizada con $\tau = 10^6$, con τ dado por (5.0.3).

Sin embargo, el valor de algoritmos más complejos comparados con la imputación por la media es revelado en los casos 2 y 4, en donde la tasa de datos faltantes aumenta. Mientras que SVTC recupera la estructura de triángulo con las poblaciones base en las esquinas y los individuos mezclados en las aristas (aunque con una dispersión mayor que en la figura original), imputar por la media distorsiona las figuras y, por lo tanto, la relación entre los individuos que podemos sustraer de estas. La peor de estas deformaciones es la

del escenario 4, en donde los individuos mezclados tienen una regresión al origen y, en consecuencia, las aristas del triángulo se pierden. De forma similar a lo esbozado en el capítulo (2) desaconsejamos por lo tanto la imputación a través de la media en este tipo de datos puesto que puede llevar a esconder estructuras relevantes entre los individuos estudiados.

Los algoritmos Alt-Min y SLPCA tienen un desempeño cualitativamente similar al de SVTC y el resultado puede consultarse en (F.1); ambos algoritmos conservan la relación entre los individuos base e individuos mezclados y no deforman significativamente, para estos escenarios, la figura original. La principal diferencia entre los resultados de estos dos algoritmos y los expuestos en la figura (F.1) radica en que las representaciones obtenidas corresponden a rotaciones y/o simetrías de la figura original. Esto no es, sin embargo, una desventaja de estos dos algoritmos puesto que, debido a la estructura del problema de optimización (3.0.5) dos soluciones son equivalentes si existe una isometría que transforme una solución en la otra.

Para poder determinar determinar cuán similares son entre sí las representaciones que realiza cada uno de los algoritmos y también para determinar su robustez a datos faltantes en los escenarios estudiados, definimos para cada punto $\mathbf{x}_j$ la medida

$$d_j = \frac{\|\hat{\mathbf{x}}_j\|}{\left\|\hat{\mathbf{x}}_j^{\text{orig}}\right\|},$$

donde $\hat{\mathbf{x}}_j$ es un punto reconstruido en las dos primeras componentes principales por un algoritmo dado y $\hat{\mathbf{x}}_j^{\text{orig}}$ es un punto reconstruido a través de la descomposición SVD de la matriz original y sin datos faltantes.

Para estudiar cuán similares son las representaciones entre sí, realizamos los modelos lineales

$$d_j^{\text{altmin}} = \beta_0 + \beta_1 d_j^{\text{slpca}} + \varepsilon_j, \quad d_j^{\text{altmin}} = \beta_0 + \beta_1 d_j^{\text{svtc}} + \varepsilon_j$$

y

$$d_j^{\text{slpca}} = \beta_0 + \beta_1 d_j^{\text{svtc}} + \varepsilon_j,$$

donde d_j^{altmin}, d_j^{slpca} y d_j^{svtc} son los medidas correspondientes a los puntos reconstruidos por Alt-Min, SLPCA y SVTC, respectivamente. El resultado es que los coeficientes β_j son significativos para todo los casos a un nivel de significación de 10^{-16}.

En la figura (6.3) tenemos representada la distribución de d_j para cada experimento y para cada método. Observamos que los puntos reconstruidos por el algoritmo SLPCA presentan una distribución diferente a la del resto de los algoritmos. Esto es un artefacto del algoritmo SLPCA implementado. Se consideró una grilla de posibles valores de λ según el problema (3.2.5) y se eligió aquel que minimiza el error de reconstrucción $\|\mathbf{W} \odot (\mathbf{X} - \boldsymbol{\mu}\mathbf{1} - \mathbf{U}\mathbf{Y})\|_F$. No obstante, el parámetro λ restringe la norma de las matrices $\mathbf{U}$ e $\mathbf{Y}$ y, en consecuencia, los puntos reconstruidos tienen normas pequeñas en comparación con las de los otros algoritmos. Para los algoritmos Alt-Min y SVTC, sus distribuciones tienen medias cercanas a 1 por lo que presentan menos deformaciones en la representación. En el escenario 2, sin embargo, existe un corrimiento en la distribución de SVTC que no se da Alt-Min: esto evidencia que Alt-Min deforma menos, para los parámetros escogidos, las distancias al origen. Es importante recordar que el algoritmo SVTC

fue detenido según el criterio de parada definido en (5.2.1) para un $\varepsilon = 0,05$; es posible, por lo tanto, que su desempeño sea más robusto a datos faltantes al tomar errores más pequeños.

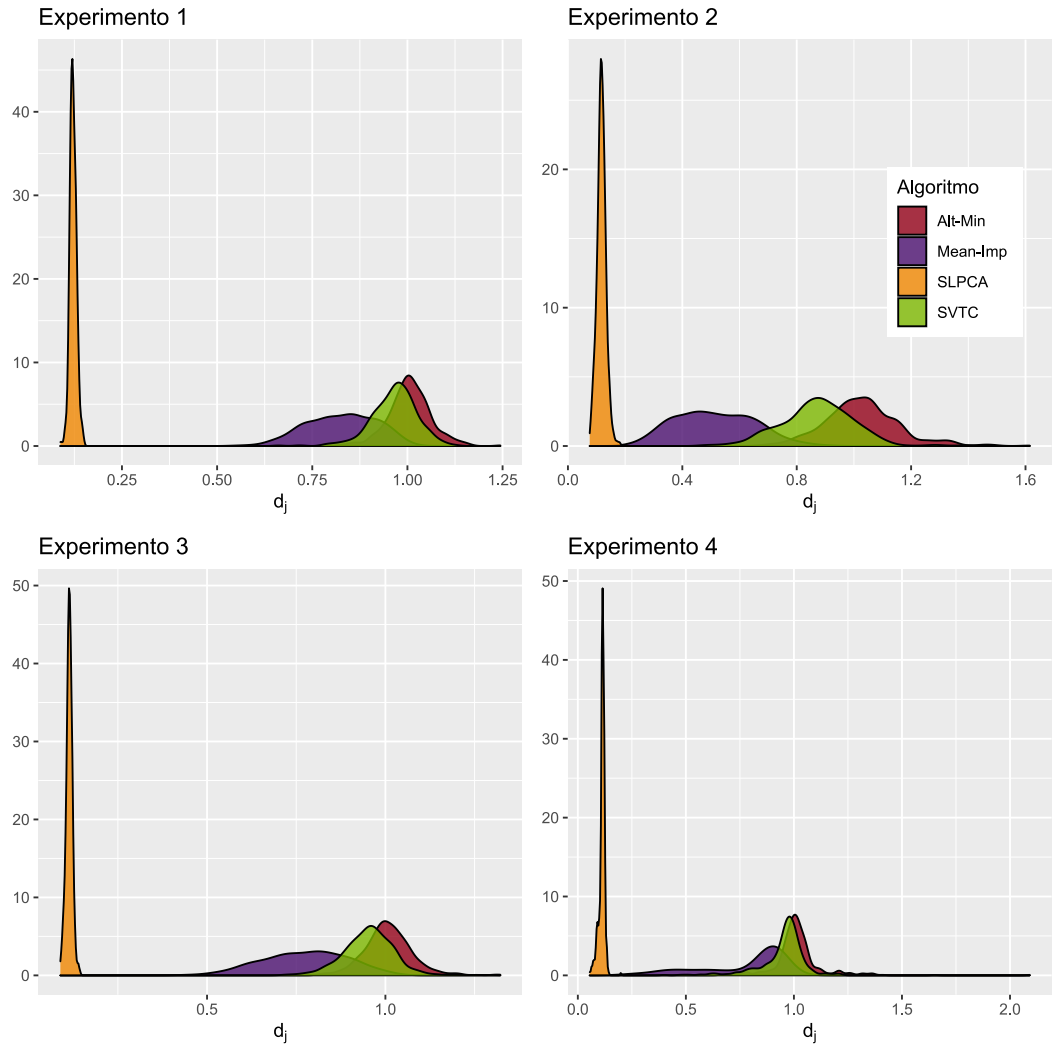

Figura 6.3: (A) Densidad estimada de los d_j para cada algoritmo y en cada escenario.

En experimentos previamente realizados observamos que en algunas instancias los algoritmos representaban a los puntos reconstruidos más cerca del $(0, 0)$ si la proporción de datos desconocidos para ese individuo era mayor. En la figura (6.4) observamos si existe una correlación lineal entre la norma de los puntos reconstruidos y la tasa de datos faltantes para el escenario 2. La inclusión de esta figura se debe a que en la figura (6.3) parece que es en este el escenario donde existe un mayor corrimiento hacia el origen. En este escenario y a un nivel de significación del $5\,\%$ todos las técnicas arrojan puntos cuyas normas están correlacionadas con la proporción de datos faltantes. Esta correlación es encontrada para todos los algoritmos y para todos los escenarios de datos faltantes salvo para dos: los puntos reconstruidos por Alt-Min para los escenarios 1 y 3 no tienen normas correlacionadas con la tasa de datos faltantes.

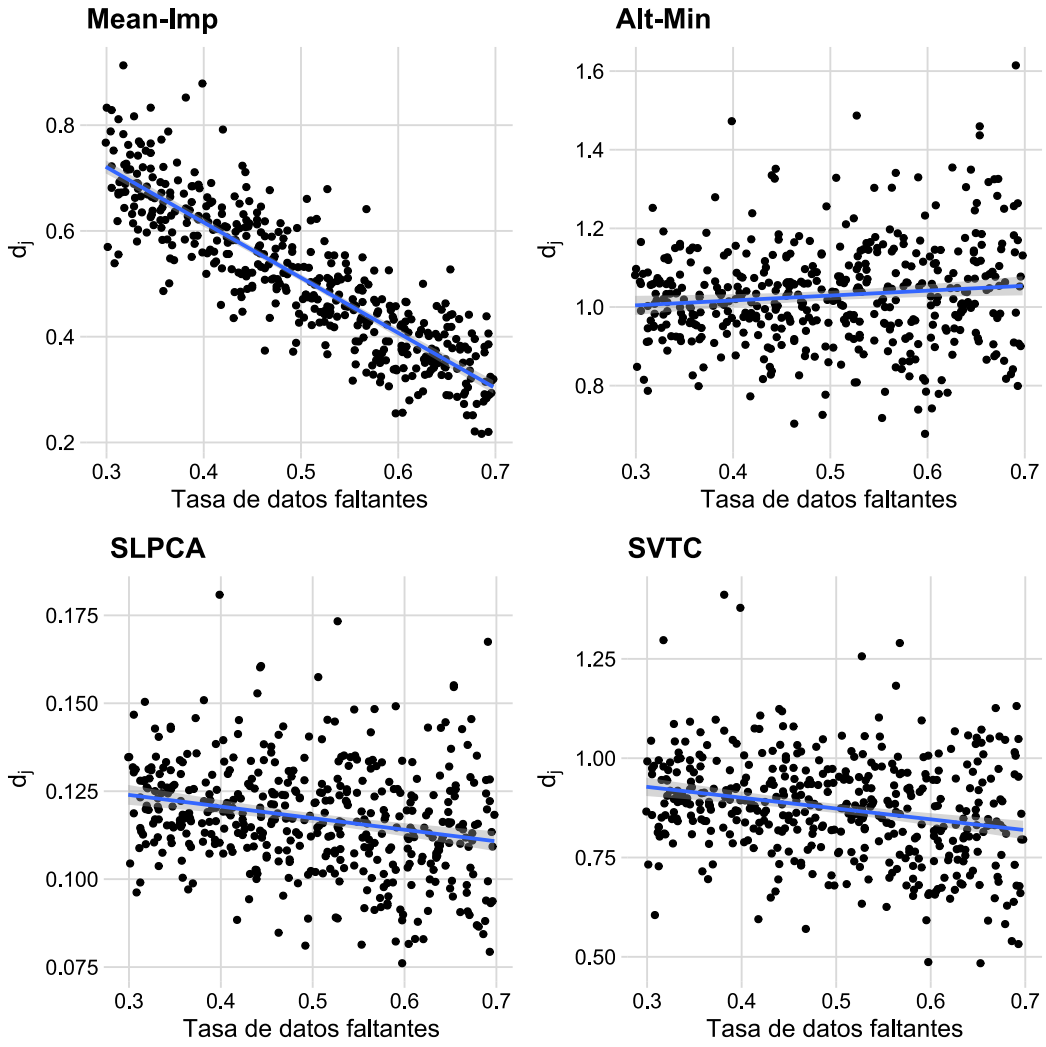

Figura 6.4: d_j en función de la tasa de datos faltantes para los puntos reconstruidos del experimento 2. En azul, la recta de regresión lineal.

6.3. Escalabilidad y tiempo de convergencia

En [30] se estima que un genoma típico difiere de un genoma de referencia en entre 4.1 y 5 millones de sitios. A pesar de que no todas las posiciones son SNPs, se estima en este mismo artículo que más del $99{,}9\,\%$ son SNPs. Si bien algunas variantes son comunes entre individuos de poblaciones del mundo, muchas variantes están restringidas a poblaciones geográficamente cercanas. Esto nos muestra que la utilización de un número grande de SNPs puede ser útil para realizar un análisis fino de la pertenencia de los individuos de una base de datos a subpoblaciones. Por esta razón estamos interesados en estudiar cuáles de los algoritmos implementados pueden ser aplicados a bases de datos similares a las que se utilizarían en un estudio real de estructura poblacional.

En una primera instancia simulamos cuatro bases de datos con una cantidad creciente de marcadores. Para las cuatro bases simulamos 50 individuos de cada una de las poblaciones base y un número de marcadores, d, igual a 250, 500, 1000 y 2000. Este número, considerablemente menor al que uno utilizaría en datos reales, nos permitió estudiar el factor de escalabilidad de los algoritmos. El esquema de datos faltantes utilizado en este

caso fue el de datos faltantes independientes: para cada entrada se simuló una variable aleatoria Bernoulli de parámetro $0{,}3$ de forma independiente.

Todos los resultados de esta sección y de las posteriores son analizados en base a la ejecución de los algoritmos en una PC con un procesador Xeon E3-1240v3 3.4GHz de 4 núcleos y 32 GB de RAM.

Comencemos el análisis estudiando aquellos algoritmos que son dependientes de los valores iniciales: nos referimos a Alt-Min y SLPCA.

Para estudiar el comportamiento de estos dos algoritmos se repitieron 50 iteraciones para cada una de los casos de prueba modificando los valores iniciales de $\mathbf{U}_{(0)}$ y $\mathbf{Y}_{(0)}$. Como estamos interesados en la capacidad de los algoritmos de reconstruir la matriz original de datos en dos dimensiones, fijamos $p = 2$ en (3.0.5) y (3.2.5). Para cada uno de los experimentos se construyeron las matrices $\mathbf{U}_{(0)} \in \mathbb{R}^{d\times p}$ y $\mathbf{Y}_{(0)} \in \mathbb{R}^{p\times n}$ según una distribución normal de varianza 1 y media igual a

$$\mu = \frac{1}{dn}\sum_{i=1}^{d}\sum_{j=1}^{n} w_{ij}x_{ij},$$

donde $\mathbf{X} = (x_{ij}) \in \mathbb{R}^{d\times n}$ es la matriz de entrada y $\mathbf{W} = (w_{ij}) \in \mathbb{R}^{d\times n}$ es la matriz definida en (3.0.2).

El resultado de estos experimentos puede verse en (6.5). En esta gráfica observamos una estimación por densidad del tiempo de ejecución medido en minutos y la cantidad de iteraciones necesaria hasta la convergencia de los algoritmos Alt-Min y SLPCA y en función de la cantidad de variables consideradas en el conjunto de datos. El tiempo necesario hasta la convergencia de SLPCA se ve apenas alterado al aumentar el número de marcadores considerados mientras que la distribución del tiempo de Alt-Min sí se observa alterada por el número de marcadores. Esto se debe a que en cada iteración de Alt-Min se debe calcular una descomposición QR y la inversión de dos matrices mientras que para SLPCA el tipo de operaciones que se deben realizar son computacionalmente menos costosas.

Por otra parte el número de iteraciones hasta la convergencia es menor en Alt-Min que en SLPCA. El análisis de esto debe, sin embargo, tomarse con cuidado puesto que el número de iteraciones de SLPCA depende de la sucesión $\{\delta_k\}_{k\in\mathbb{N}}$ de pasos de gradiente descendente (recordemos las iteraciones definidas en 3.2.6) y (3.2.7). En este caso se utilizó $\delta_k = \delta = 0{,}0001$ para todo $k \in \mathbb{N}$, valor que permitió repetir el algoritmo 50 veces. Para valores de δ más grandes no siempre se logró la convergencia en 50 repeticiones del algoritmo.

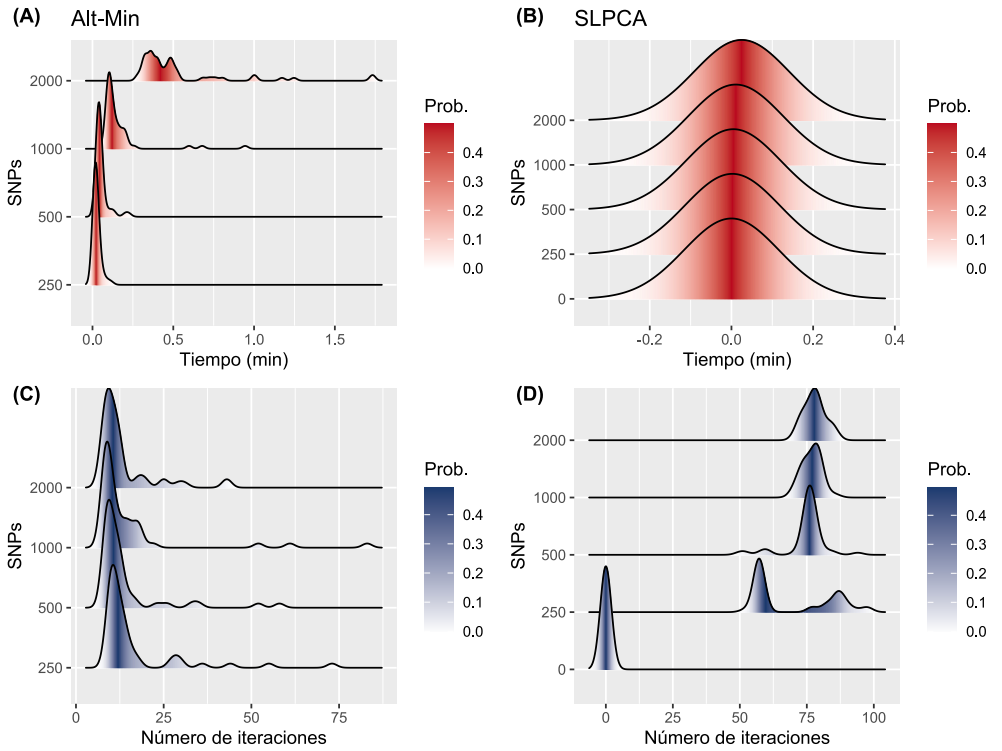

Figura 6.5: Tiempos de ejecución y número de iteraciones necesarias hasta la convergencia para los algoritmos Alt-Min y SLPCA. En las figuras (A) y (B) observamos el tiempo de ejecución y número de iteraciones para Alt-Min. En las figuras (C) y (D) observamos el tiempo de ejecución y número de iteraciones para SLPCA. Los gráficos se corresponden con estimadores de la densidad sobre un total de $n = 50$ repeticiones de los algoritmos con matrices iniciales variables. El degradado corresponde con la probabilidad acumulada hacia la izquierda para los puntos menores a la mediana o hacia la derecha para los puntos mayores a la mediana de la distribución.

En la tabla (6.1) vemos resumido el comportamiento de los algoritmos implementados. Para construir esta tabla se consideró un error de $\varepsilon = 10^{-5}$ como criterio de parada. En el caso de Alt-Min y SLPCA, las cifras consideradas corresponden a la mediana del tiempo y del número de iteraciones. Observamos que SLPCA es el algoritmo que presenta el tiempo hasta la convergencia más pequeño de entre los considerados. Por su parte, el algoritmo SVTC es el que presenta tiempos de convergencia considerablemente más altos que el resto. En particular, en esta tabla se consideró un número máximo de iteraciones de 3×10^5 y el algoritmo no había alcanzado el error predefinido antes de estas iteraciones; es por esto que el número de iteraciones no está presente en esta tabla para este algoritmo.

Número de marcadores (d)	250	500	1000	2000
Alt-Min	0.45 (285)	0.82 (218)	1.70 (146)	0.49 (13)
SLPCA	0.0018 (58)	0.005 (76)	0.01 (77)	0.024 (77)
SVTC	102.81	146.33	257.59	457.41

Cuadro 6.1: Tiempos de ejecución medidos en minutos y, en paréntesis, la cantidad de iteraciones hasta alcanzar la convergencia. La convergencia de todos los algoritmos fue analizada utilizando un umbral de $\varepsilon = 10^{-5}$.

6.4. Aplicación a una base de datos de poblaciones nativas americanas

Cualitativamente, los métodos estudiados fueron capaces de reconstruir en datos simulados las relaciones entre individuos de poblaciones bases e individuos mezclados. Este fue, sin embargo, un caso de estudio sencillo en términos de estructura poblacional. Las poblaciones que tomamos de base (japoneses, vascos y yorubas) forman parte de tres continentes diferentes y eso tiene una consecuencia directa en la divergencia genética [8]. Estamos interesados en estudiar si podemos reconstruir las distancias entre individuos que pertenecen a poblaciones de un mismo continente.

En el artículo [31] se estudió la similitud que existe entre 10 individuos que declararon ascendencia charrúa e individuos de otras poblaciones nativas americanas geográficamente cercanas. Para cada uno de los individuos con ascendencia autodeclarada charrúa se estudiaron sus genomas completos y cada sección de estos fue etiquetada como provenienten de señales *nativas*, *europeas* o *africanas*. En el espíritu del artículo [10], se construyó una base de datos con las señales europeas y africanas enmascaradas: es decir, las secciones de los genomas correspondientes a señales europeas y africanas se consideraron datos desconocidos. En el artículo de Spangenberg et al. se construyó un PCA entre los individuos con ascendencia charrúa y otras poblaciones nativas utilizando el algoritmo `smartpca` del paquete `EIGENSOFT`. De acuerdo con [29], este algoritmo utiliza imputación por la media para realizar un PCA con datos faltantes. Estamos interesados en estudiar si podemos mejorar el resultado de este PCA si utilizamos los algoritmos propuestos en este trabajo.

En lugar de utilizar los genomas completos, utilizamos una base de datos proveniente de una muestra de 363 578 SNPs de 210 individuos de 10 poblaciones. En la tabla (6.2) podemos observar las poblaciones que fueron consideradas en este análisis. Cabe resaltar que los individuos charrúas presentaban tasas promedio de datos faltantes significativamente mayores que las del resto de las poblaciones.

Población	Tasa de datos faltantes
Aimara	0.042
Chane	0.010
Diaguita	0.269
Guahibo	0.008
Guarani	0.100
Kaingang	0.172
Piapoco	0.028
Quechua	0.092
Toba	0.027
Charrúa	0.798

Cuadro 6.2: Poblaciones estudiadas y la tasa de datos faltantes promedio de los individuos de cada una de ellas.

El primero de los descubrimientos es que el algoritmo Alt-Min, a pesar de la robustez a los datos faltantes que observamos en la sección 6.2 presenta un problema al ser implementado que no había sido descubierto con los datos simulados: una implementación naíf

del algoritmo (como la que realizamos) utiliza grandes cantidades de memoria RAM. Por esta razón, no pudimos utilizar este algoritmo para estudiar estas poblaciones, aún utilizando un submuestreo de SNPs (para ejemplificar, un submuestreo de 75 000 SNPs obliga a guardar en memoria una matriz de 41,9 Gb lo que es imposible en la computadora que se utilizó para hacer estos experimentos).

El problema de la falta de memoria RAM no es tal para el método SLPCA. En la figura (6.6) observamos la proyección de los individuos estudiados en las dos primeras componentes principales de acuerdo al algoritmo SLPCA. Cuando consideramos todos los individuos observamos que algunos individuos charrúas quedan muy mal representados y lejos del resto de los puntos. Una posible razón para esto es que estos puntos corresponden con individuos con tasas muy altas de datos faltantes (aproximadamente 0.99); sin embargo esto no condice con lo hallado en la sección (6.2) puesto que en esta sección hallamos que la norma de los puntos reconstruidos disminuía con al aumentar el número de entradas desconocidas. Para entender si las distancias entre el resto de las poblaciones quedan bien reconstruidas por este método, construimos un nuevo PCA sin los charrúas. El resultado es desalentador: aún sin los individuos anómalos, el método falla en reconstruir una figura que dé cuenta de las distancias genómicas de los individuos. Una posible razón por la cual esto se esté dando puede ser que debido a la similitud entre las poblaciones, el algoritmo se estanque en mínimos locales que están lejos de la solución óptima. Por otra parte, en esta iteración utilizamos un λ fijo; es posible que una optimización más fina de este parámetro arroje mejores resultados.

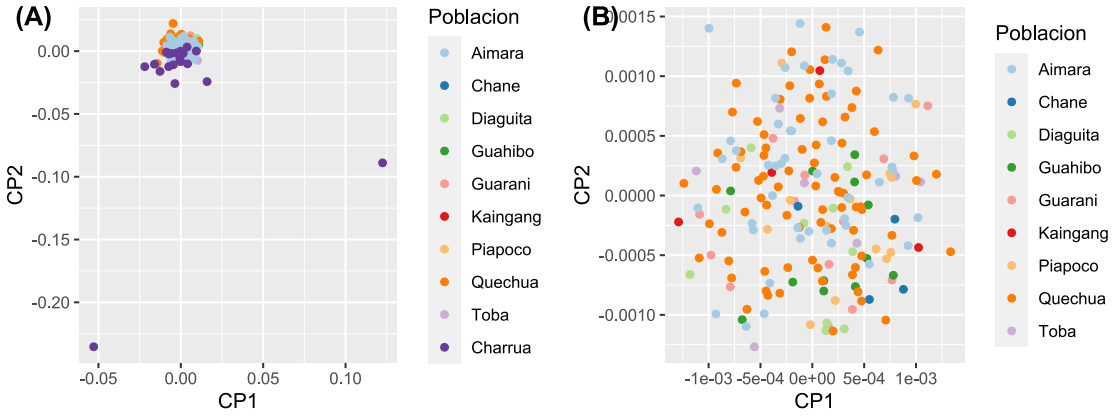

Figura 6.6: Proyección sobre las dos primeras componentes principales de los individuos correspondientes a poblaciones nativas americanas según el algoritmo SLPCA para $\delta = 10^{-8}$ y $\lambda = 1000$. (A) La base de datos completa. (B) La base de datos sin los individuos charrúas.

El algoritmo SVTC sí nos permitió realizar un PCA que revele las relaciones entre individuos de las poblaciones nativas estudiadas. El resultado de esto puede verse en la figura 6.7. Para construir esta figura utilizamos un submuestreo aleatorio de 75 000 SNPs. En la figura eliminamos aquellos puntos cuya tasa de datos faltantes era mayor a $0{,}95$ pero estos no fueron eliminados de la matriz a completar. El resultado es que la primera componente principal se ve fuertemente afectada por estos puntos. En el sentido de la segunda componente principal, sí podemos diferenciar las poblaciones.

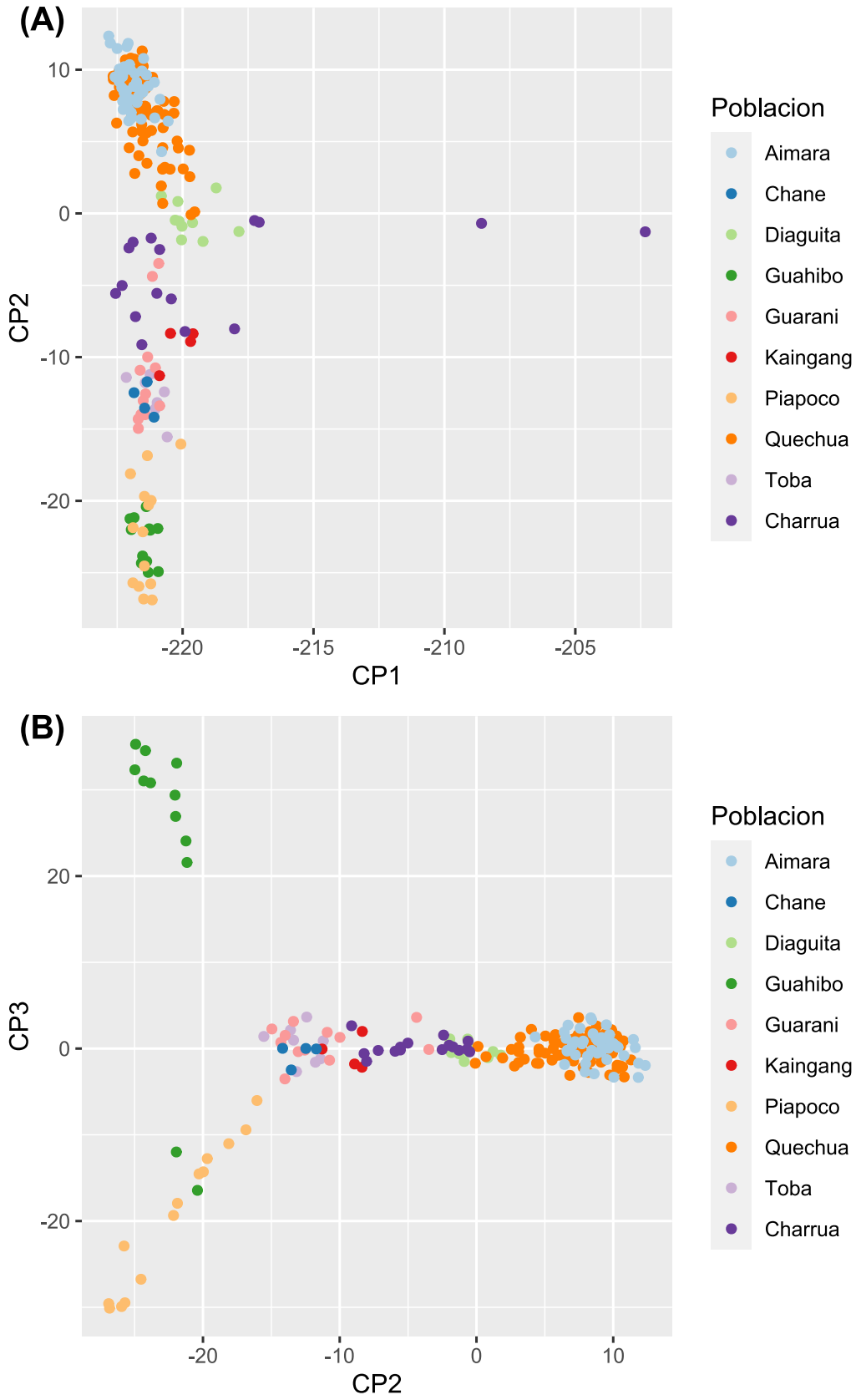

Figura 6.7: Reconstrucción de los individuos de la sección 6.4 según el algoritmo SVTC. La matriz que se utilizó para completar contenía 75 000 SNPs y todos los individuos. El algoritmo se ejecutó hasta obtener una convergencia con un error de $\varepsilon = 0,05$ y para $\tau = 10^6$.

En concordancia con lo hallado en [31], observamos una proximidad entre los individuos charrúas, diaguitas, kaingang y guaraníes. Según este artículo, la similaridad hallada a nivel genético entre charrúas y diaguitas, a pesar de parecer inesperada por razones geográficas, está bien documentada a nivel etnográfico.

Por otra parte, si proyectamos a los individuos sobre la segunda y tercera componente principal somos capaces de diferenciar a las poblaciones piapoco y guahibo, ambas poblaciones nativas de Colombia y Venezuela.

Capítulo 7

Conclusiones y trabajo futuro

Los métodos estudiados para construir un PCA en presencia de datos faltantes arrojan resultados razonables en los distintos escenarios simulados que construimos. En particular, en estos escenarios, no parece haber grandes diferencias en la estructura de los datos faltantes y los métodos recuperan las similitudes entre individuos sin importar si los datos faltantes se consideran aleatorios dentro de la base de datos o con estructura de bloque. Sin embargo, dados los resultados desalentadores que obtuvimos del método de SLPCA al aplicarlos sobre una base de datos no simulada es relevante cuestionarse si los escenarios que construimos son demasiados sencillos para los algoritmos. En este sentido, sería interesante trabajar en la construcción de nuevos escenarios en donde la tasa de datos faltantes sea aún más extrema (por ejemplo, cercana a 0.9) y estudiar cómo responden los métodos a estos escenarios. Incluso con datos simulados, sería interesante estudiar cómo es el desempeño de los métodos estudiados cuando la estructura de datos faltantes es similar a la de los charrúas de la sección 6.4.

El algoritmo PPCA no pudo ser aplicado, en la versión descrita en el capítulo 4, a datos genómicos. Artículos recientes como [29] proponen una variante del algoritmo EM que, de acuerdo con los autores, permite no solo recuperar estructuras que no son recuperadas por los paquetes usados ampliamente en el área de genética de poblaciones, si no que también prometen una escalabilidad apropiada para estudiar bases de datos de altas dimensiones. Sería interesante continuar en el estudio de estos métodos y compararlos con aquellos que no son probabilísticos.

Por otra parte, el algoritmo Alt-Min que tuvo un buen desempeño para datos simulados en términos de escalabilidad y de robustez a datos faltantes, no pudo ser aplicado a un conjunto de datos de individuos reales. Nos encontramos con esta dificultad debido a que el algoritmo internamente realiza operaciones costosas computacionalmente. Sería pertinente, por lo tanto, continuar en el estudio de implementaciones de este algoritmo computacionalmente menos costosas que permitan una mejor escalabilidad.

En cuanto a SVTC, su desempeño sobre datos simulados y sobre la base de datos de poblaciones nativas, parece prometedor e invita a realizar nuevos experimentos en otras bases de datos con estructuras complejas de datos faltantes. Entendemos, sin embargo, que de acuerdo a lo estudiado en la sección 6.3, el algoritmo tiene tiempos de ejecución elevados comparados con el resto de los métodos estudiados aún para matrices de dimensiones muy por debajo de aquellas utilizadas en estudios de estructura poblacional con datos reales. Es pertinente un estudio de nuevas técnicas que permitan realizar el problema de completar las matrices pero que aseguren tiempos de ejecución menores. No obstante, sería

interesante poder reconstruir la base de datos de poblaciones nativas utilizando todos los SNPs y no una muestra; si fuimos capaces de hallar similitudes entre individuos con 75 000 SNPs, cabe la pregunta de qué tipo de estructura fina podemos revelar al utilizar la totalidad de la base de datos.

Apéndice A

Elementos de álgebra lineal

A.1. Conceptos esenciales y propiedades

Traza de una matriz

Definición A.1.1 (Traza de una matriz)**.** Dada una matriz $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{n\times n}$ definimos ***traza*** como la suma de los elementos en la diagonal principal, es decir,

$$\operatorname{tr}(\mathbf{A}) = \sum_{i=1}^{n} a_{ii}.$$

Proposición A.1.1 (Propiedades de la traza)**.** *La función* $\operatorname{tr}(\cdot)$ *definida en el espacio de matrices cuadradas cumple las siguientes propiedades:*

(1) Para todas $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$, $\operatorname{tr}(\mathbf{A}+\mathbf{B}) = \operatorname{tr}(\mathbf{A}) + \operatorname{tr}(\mathbf{B})$.

(2) Para todo $\lambda \in \mathbb{R}$ *y* $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\operatorname{tr}(\lambda\mathbf{A}) = \lambda \operatorname{tr}(\mathbf{A})$.

(3) Para toda $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\operatorname{tr}(\mathbf{A}) = \operatorname{tr}(\mathbf{A}^\top)$.

(4) Para toda $\mathbf{A} \in \mathbb{R}^{m\times n}$ *y* $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\operatorname{tr}(\mathbf{AB}) = \operatorname{tr}(\mathbf{BA})$.

(5) Es invariante bajo permutaciones cíclicas de productos de matrices, es decir, dadas $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ *matrices,* $\operatorname{tr}(\mathbf{ABCD}) = \operatorname{tr}(\mathbf{BCDA}) = \operatorname{tr}(\mathbf{CDAB}) = \operatorname{tr}(\mathbf{DABC})$ *(siempre que estos productos estén bien definidos).*

Demostración. Daremos la prueba de la propiedad (5). El resto de las propiedades surge de forma inmediata de la definición de traza.

Probemos que $\operatorname{tr}(\mathbf{ABCD}) = \operatorname{tr}(\mathbf{BCDA})$; el resto de las igualdades se prueban de formas análogas. Considerando $\mathbf{U} = \mathbf{A}$ y $\mathbf{V} = \mathbf{BCD}$, tenemos, en virtud de la propiedad (4), que

$$\operatorname{tr}(\mathbf{ABCD}) = \operatorname{tr}(\mathbf{UV}) = \operatorname{tr}(\mathbf{VU}) = \operatorname{tr}(\mathbf{BCDA}). \qquad \blacksquare$$

Corolario A.1.1.1. *Sean* $\mathbf{x} \in \mathbb{R}^n$ *y* $\mathbf{A} \in \mathbb{R}^{n\times n}$. *Entonces*

$$\mathbf{x}^\top\mathbf{Ax} = \operatorname{tr}(\mathbf{x}^\top\mathbf{Ax}) = \operatorname{tr}(\mathbf{Axx}^\top) = \operatorname{tr}(\mathbf{xx}^\top\mathbf{A}).$$

Demostración. Puesto que $\mathbf{x}^\top \mathbf{A}\mathbf{x} \in \mathbb{R}$, tenemos que $\mathbf{x}^\top \mathbf{A}\mathbf{x} = \mathrm{tr}(\mathbf{x}^\top \mathbf{A}\mathbf{x})$. Luego, por la proposición (A.1.1) tenemos que

$$\mathrm{tr}(\mathbf{x}^\top \mathbf{A}\mathbf{x}) = \mathrm{tr}(\mathbf{x}^\top (\mathbf{A}\mathbf{x})) = \mathrm{tr}((\mathbf{A}\mathbf{x})\mathbf{x}^\top) = \mathrm{tr}(\mathbf{A}\mathbf{x}\mathbf{x}^\top)$$

y

$$\mathrm{tr}(\mathbf{x}^\top \mathbf{A}\mathbf{x}) = \mathrm{tr}((\mathbf{x}^\top \mathbf{A})\mathbf{x}) = \mathrm{tr}(\mathbf{x}(\mathbf{x}^\top \mathbf{A})) = \mathrm{tr}(\mathbf{x}\mathbf{x}^\top \mathbf{A}). \qquad \blacksquare$$

Determinante y matriz de cofactores

Definiciones A.1.1. Consideremos una matriz cuadrada $\mathbf{X} \in \mathbb{R}^{n\times n}$.

- Sean $I, J \subset \{1, \ldots, n\}$ tal que $\mathrm{card}(I) = \mathrm{card}(J) = k$. Definimos el ***determinante menor de orden*** k o ***menor de orden*** k como al determinante de la submatriz de tamaño $k \times k$ de $\mathbf{X}$ obtenida tras eliminar las filas de i que no pertenecen al conjunto I y las columnas que no pertenecen al conjunto J. Si $I = \{i\}^c$ y $J = \{j\}^c$ llamaremos al ***primer menor*** al menor de orden 1 obtenido y lo notaremos como $\mathbf{X}_{ij}$. Si $I = J = \{1, \ldots, k\}$, llamaremos ***menor principal superior***.

- Definimos la ***matriz de cofactores*** de $\mathbf{X}$ y la notamos como $\mathrm{cof}(\mathbf{X})$ a la matriz que cumple que $\mathrm{cof}(\mathbf{X}) = (-1)^{i+j}\mathbf{X}_{ij}$.

- Definimos la ***matriz adjunta*** de $\mathbf{X}$ como la traspuesta de la matriz de cofactores de $\mathbf{X}$ y la notamos como $\mathrm{adj}(\mathbf{X})$.

La matriz adjunta de una matriz $\mathbf{X}$ invertible está estrechamente vinculada con la matriz inversa y su determinante. Esto está dado por la siguiente proposición:

Proposición A.1.2 (Vínculo entre matriz adjunta e inversa)**.** *Para toda matriz* $\mathbf{X} \in \mathbb{R}^{n\times n}$ *se tiene que*

$$\mathbf{X}\,\mathrm{adj}(\mathbf{X}) = \mathrm{adj}(\mathbf{X})\,\mathbf{X} = \det(\mathbf{X})\mathbf{I}_n.$$

En particular, si $\mathbf{X}$ *es invertible se cumple que*

$$\mathbf{X}^{-1} = \frac{\mathrm{adj}(\mathbf{X})}{\det(\mathbf{X})}.$$

Demostración. Ver [32], página 192. $\blacksquare$

Pseudoinversa de Penrose-Moore

Definición A.1.2 (Pseudoinversa de Moore-Penrose)**.** Sea $\mathbf{X} \in \mathbb{R}^{m\times n}$. Decimos que $\mathbf{X}^+$ es una ***pseudoinversa de Moore-Penrose*** o, simplemente, ***pseudoinversa*** si satisface las siguientes propiedades

$$\mathbf{X}\mathbf{X}^+\mathbf{X} = \mathbf{X}, \tag{A.1.1}$$
$$\mathbf{X}^+\mathbf{X}\mathbf{X}^+ = \mathbf{X}^+ \tag{A.1.2}$$
$$(\mathbf{X}^+\mathbf{X})^\top = \mathbf{X}\mathbf{X}^+ \tag{A.1.3}$$
$$(\mathbf{X}\mathbf{X}^+)^\top = \mathbf{X}^+\mathbf{X} \tag{A.1.4}$$

Una propiedad fundamental de una pseudoinversa de una matriz es que siempre existe y es única:

Proposición A.1.3 (Existencia y unicidad de la pseudoinversa)**.** *Para toda matriz* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *existe una única pseudoinversa.*

Demostración. Ver [33]. ■

Las pseudoinversas generalizan el concepto de inversa de una matriz en el siguiente sentido:

Proposición A.1.4 (Pseudoinversa de una matriz invertible)**.** *Sea* $\mathbf{X} \in \mathbb{R}^{m\times m}$ *invertible. Luego* $\mathbf{X}^+ = \mathbf{X}^{-1}$.

Demostración. Probemos que $\mathbf{X}^{-1}$ cumple las cuatro propiedades que definen una pseudoinversa de $\mathbf{X}$. Tenemos que

$$\begin{aligned}\mathbf{X}\mathbf{X}^{-1}\mathbf{X} &= \mathbf{I}\mathbf{X} = \mathbf{X}\\ \mathbf{X}^{-1}\mathbf{X}\mathbf{X}^{-1} &= \mathbf{I}\mathbf{X}^{-1} = \mathbf{X}^{-1}\\ (\mathbf{X}^{-1}\mathbf{X})^\top &= \mathbf{I}^\top = \mathbf{I} = \mathbf{X}\mathbf{X}^{-1}\\ (\mathbf{X}\mathbf{X}^{-1})^\top &= \mathbf{I}^\top = \mathbf{I} = \mathbf{X}^{-1}\mathbf{X}\end{aligned}$$

Luego, por la proposición (A.1.3), $\mathbf{X}^+ = \mathbf{X}^{-1}$. ■

Matrices de entrada única

Definición A.1.3 (Matrices de entrada única)**.** Dado $m \in \mathbb{Z}_+$ y un par $(i, j) \in \mathbb{Z}_+^2$ con $i, j \in \{1, \dots, m\}$ definimos a la matriz $\boldsymbol{\delta}_{ij} \in \mathbb{R}^{m\times m}$ como

$$(\boldsymbol{\delta}_{ij})_{k\ell} = \begin{cases} 1 & \text{si } k = i \text{ y } \ell = j \\ 0 & \text{si no} \end{cases}$$

Las matrices de entrada única son *selectoras de filas y columnas* a través de la multiplicación con otras matrices como lo da la siguiente proposición.

Proposición A.1.5. *Sea* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *una matriz arbitraria.*

(1) Sean $i, j \in \{1, \dots, m\}$. *La matriz* $\mathbf{A} = \boldsymbol{\delta}_{ij}\mathbf{X} \in \mathbb{R}^{m\times n}$ *es tal que*

$$\mathbf{A}_{k\ell} = \begin{cases} x_{j\ell} & \textit{si } k = i \\ 0 & \textit{si no} \end{cases}.$$

Es decir, la matriz $\mathbf{A}$ *tiene en su fila* i *a las entradas de la fila* j *de* $\mathbf{X}$ *y el resto de sus entradas son* 0.

(2) Sean $i, j \in \{1, \dots, n\}$. *La matriz* $\mathbf{B} = \mathbf{X}\boldsymbol{\delta}_{ij} \in \mathbb{R}^{m\times n}$ *es tal que*

$$\mathbf{B}_{k\ell} = \begin{cases} x_{ki} & \textit{si } \ell = j \\ 0 & \textit{si no} \end{cases}.$$

Es decir, la matriz $\mathbf{B}$ *tiene en su columna* j *a las entradas de la columna* i *de* $\mathbf{X}$ *y el resto de sus entradas son* 0.

Demostración. Directo de la definición de $\boldsymbol{\delta}_{ij}$ ■

Valores propios de la suma de matrices

Proposición A.1.6 (Valores propios de la suma)**.** *Sean* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$.

(1) Si $\mathbf{A}$ *y* $\mathbf{B}$ *son diagonalizables y conmutan si y sólo si* $\mathbf{A}$ *y* $\mathbf{B}$ *son simultáneamente diagonalizables, es decir, si existen* $\mathbf{P}$ *invertible y* $\mathbf{D}_1, \mathbf{D}_1$ *matrices diagonales tales que*

$$\mathbf{A} = \mathbf{P}\mathbf{D}_1\mathbf{P}^{-1} \quad y \quad \mathbf{B} = \mathbf{P}\mathbf{D}_2\mathbf{P}^{-1}.$$

(2) $\mathbf{A}$ *y* $\mathbf{B}$ *conmutan entonces* $\mathbf{A}+\mathbf{B}$ *es diagonalizable y sus valores propios pueden ser escritos como* $\lambda = \lambda_A + \lambda_B$ *donde* λ_A *y* λ_B *son valores propios de* $\mathbf{A}$ *y* $\mathbf{B}$*, respectivamente.*

Demostración. (1) Supongamos que $\mathbf{A}$ y $\mathbf{B}$ son diagonalizables y conmutan y probemos que son simultáneamente diagonalizables. Sea v un vector propio de $\mathbf{A}$, es decir, existe $\lambda \in \mathbb{R}$ tal que $\mathbf{A}v = \lambda v$. Probemos que v es un vector propio de $\mathbf{B}$. Tenemos que

$$\mathbf{A}(\mathbf{B}v) = \mathbf{B}(\mathbf{A}v) = \mathbf{B}(\lambda v) = \lambda(\mathbf{B}v).$$

Esto implica que el vector $\mathbf{B}v$ es un vector propio asociado a λ y, por lo tanto, pertenece al subespacio generado por v. Luego, existe μ tal que $\mathbf{B}v = \mu v$. Esto implica que v es un vector propio asociado a $\mathbf{B}$.

Análogamente, probamos que si v es un vector propio de $\mathbf{B}$ entonces también lo es de $\mathbf{A}$. Luego, $\mathbf{A}$ y $\mathbf{B}$ se diagonalizan en la misma base de vectores propios.

Supongamos ahora que $\mathbf{A}$ y $\mathbf{B}$ son simultáneamente diagonalizables. Probemos que $\mathbf{A} = \mathbf{B}$. Tenemos que

$$\mathbf{A}\mathbf{B} = (\mathbf{P}\mathbf{D}_1\mathbf{P}^{-1})(\mathbf{P}\mathbf{D}_2\mathbf{P}^{-1}) = \mathbf{P}\mathbf{D_1}\mathbf{D_2}\mathbf{P}^{-1}$$

y

$$\mathbf{B}\mathbf{A} = (\mathbf{P}\mathbf{D}_2\mathbf{P}^{-1})(\mathbf{P}\mathbf{D}_1\mathbf{P}^{-1}) = \mathbf{P}\mathbf{D_2}\mathbf{D_1}\mathbf{P}^{-1}.$$

Luego, como las matrices diagonales conmutan entre sí, se cumple que $\mathbf{A}\mathbf{B} = \mathbf{B}\mathbf{A}$.

(2) Utilizando la hipótesis podemos escribir a la matriz $\mathbf{A}+\mathbf{B}$ como

$$\mathbf{A}+\mathbf{B} = \mathbf{P}\mathbf{D}_1\mathbf{P}^{-1} + \mathbf{B}\mathbf{P}\mathbf{D}_2\mathbf{P}^{-1} = \mathbf{P}(\mathbf{D}_1+\mathbf{D}_2)\mathbf{P}^{-1}.$$

Luego, $\mathbf{A}+\mathbf{B}$ es invertible y sus valores propios son de la forma $\lambda = \lambda_A + \lambda_B$. ■

A.2. Descomposición en valores singulares

Sea $\mathbf{X} \in \mathbb{R}^{n\times n}$ una matriz simétrica y semidefinida positiva. A través del teorema espectral para matrices simétricas y de la no negatividad de los valores propios de una matriz semidefinida positiva la matriz $\mathbf{X}$ puede ser escrita como

$$\mathbf{X} = \mathbf{P}\begin{pmatrix}\mathbf{D} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{pmatrix}\mathbf{P}^{\top},$$

donde $\mathbf{P}$ es una matriz ortogonal y $\mathbf{D}$ es una matriz diagonal que contiene a los valores propios estrictamente positivos. Podemos utilizar esta descomposición de las matrices simétricas para obtener una descomposición para matrices arbitrarias: la *descomposición en valores singulares* (SVD, por sus siglas en inglés).

Teorema A.2.1 (Descomposición en valores singulares)**.** *Dada* $\mathbf{X}$*, una matriz de rango* r*, existen una matriz ortogonal de dimensión* $n \times n$*,* $\mathbf{P}$*, una matriz ortogonal de dimensión* $m \times m$*,* $\mathbf{Q}$*, y una matriz diagonal* $\mathbf{D}$ *de dimensión* $r \times r$ *cuyas entradas son estrictamente positivas y tales que* $\mathbf{X}$ *puede ser escrita como*

$$\mathbf{X} = \mathbf{P} \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \mathbf{Q}^\top.$$

Para dar una prueba de esta proposición, probaremos un par de lemas.

Lema A.2.1 (Lema 1 para la descomposición en valores singulares)**.** *Sea* $\mathbf{X} \in \mathbb{R}^{m \times n}$*. Se cumple que* $\mathbf{X} = \mathbf{0}$ *si y sólo si* $\mathbf{X}^\top \mathbf{X} = \mathbf{0}$*.*

Demostración. La prueba del directo es inmediata. Probemos el recíproco. Para esto consideremos $\mathbf{e}_j \in \mathbb{R}^n$ el j-ésimo vector canónico. Por hipótesis tenemos que $\mathbf{X}^\top \mathbf{X}\mathbf{e}_j = \mathbf{0}$. Luego

$$\mathbf{e}_j^\top \mathbf{X}^\top \mathbf{X}\mathbf{e}_j = \langle \mathbf{X}\mathbf{e}_j, \mathbf{X}\mathbf{e}_j \rangle = \|\mathbf{X}\mathbf{e}_j\|_2^2 = 0.$$

Por lo tanto, para todo j, $\mathbf{X}\mathbf{e}_j = \mathbf{0}$. Resta probar que esto implica que $x_{ij} = 0$ para todo i, j. Notemos con $\mathbf{e}_k^{(p)}$ al k-ésimo vector canónico de dimensión p. Luego, para todo i, j concluimos que

$$x_{ij} = (\mathbf{e}_i^{(m)})^\top \mathbf{X}\mathbf{e}_j^{(n)} = 0. \qquad \blacksquare$$

Lema A.2.2 (Lema 2 para la descomposición en valores singulares)**.** *Sea* $\mathbf{X} \in \mathbb{R}^{m \times n}$ *una matriz de rango* r*. Sea* $\mathbf{P}$ *cualquier matriz ortogonal y* $\mathbf{D}$ *la matriz diagonal tal que*

$$\mathbf{Q}^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q} = \begin{pmatrix} \mathbf{D}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{A.2.1}$$

Particionemos $\mathbf{Q}$ *como* $\mathbf{Q} = (\mathbf{Q}_1, \mathbf{Q}_2)$*, donde* $\mathbf{Q}_1$ *tiene* r *columnas, y definamos* $\mathbf{P} = (\mathbf{P}_1, \mathbf{P}_2)$*, donde* $\mathbf{P}_1 = \mathbf{X}\mathbf{Q}_1\mathbf{D}^{-1}$ *y* $\mathbf{P}_2$ *es cualquier matriz de dimensión* $m \times (m - r)$ *tal que*

$$\mathbf{P_1}^\top \mathbf{P}_2 = \mathbf{0}. \tag{A.2.2}$$

(Si $r = 0$*,* $\mathbf{Q} = \mathbf{Q}_2$*,* $\mathbf{P} = \mathbf{P}_2$ *y* $\mathbf{P}_2$ *es arbitraria; si* $r = n$*,* $\mathbf{Q} = \mathbf{Q}_1$*; si* $r = m$*,* $\mathbf{P} = \mathbf{P}_1$*). Luego,*

$$\mathbf{P}^\top \mathbf{X}\mathbf{Q} = \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$$

Demostración. En primer lugar observemos que si descomponemos $\mathbf{Q} = (\mathbf{Q}_1, \mathbf{Q}_2)$ de forma tal que $\mathbf{Q}_1$ contenga a las primeras r columnas, entonces podemos escribir

$$\mathbf{Q}^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_1^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_1 & \mathbf{Q}_1^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_2 \\ \mathbf{Q}_2^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_1 & \mathbf{Q}_2^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_2 \end{pmatrix}.$$

Por la ecuación (A.2.1) obtenemos que $\mathbf{Q}_1^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_1 = \mathbf{D}^2$. Además, como $\mathbf{Q}_2^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_2 = \mathbf{0}$, por el lema (A.2.1) esto implica que $\mathbf{X}\mathbf{Q}_2 = \mathbf{0}$. Por otra parte, por definición $\mathbf{P}_1 = \mathbf{X}\mathbf{Q}_1\mathbf{D}^{-1}$. Por lo tanto, $\mathbf{P}_1^\top = \mathbf{D}^{-1}\mathbf{Q}_1^\top \mathbf{X}^\top$ y $\mathbf{X}\mathbf{Q}_1 = \mathbf{P}_1\mathbf{D}$. Luego

$$\begin{aligned}\mathbf{P}^\top \mathbf{X}\mathbf{Q} = \begin{pmatrix} \mathbf{P}_1^\top \mathbf{X}\mathbf{Q}_1 & \mathbf{P}_1^\top \mathbf{X}\mathbf{Q}_2 \\ \mathbf{P}_2^\top \mathbf{X}\mathbf{Q}_1 & \mathbf{P}_2^\top \mathbf{X}\mathbf{Q}_2 \end{pmatrix} &= \begin{pmatrix} \mathbf{D}^{-1}\mathbf{Q}_1^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_1 & \mathbf{P}_1^\top \mathbf{0} \\ \mathbf{P}_2^\top \mathbf{P}_1 \mathbf{D} & \mathbf{P}_2^\top \mathbf{0} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{D}^{-1}\mathbf{D}^2 & \mathbf{0} \\ (\mathbf{P}_2^\top \mathbf{P}_1)\mathbf{D} & \mathbf{0} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}\end{aligned}$$

■

Podemos a continuación dar una prueba de la existencia de la descomposición en valores singulares.

Prueba del teorema (A.2.1). La idea de la prueba será utilizar el lema (A.2.2) encontrando unas matrices $\mathbf{Q}$ y $\mathbf{P}$ que sean ortogonales y que estén bajo las hipótesis del lema.

Por el teorema espectral para matrices simétricas, existe una matriz ortogonal $\mathbf{Q}$ de dimensión $n \times n$ tal que

$$\mathbf{Q}^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q} = \begin{pmatrix} \mathbf{D}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0}, \end{pmatrix} \tag{A.2.3}$$

puesto que $\mathbf{X}^\top \mathbf{X}$ es una matriz simétrica y semidefinida positiva. Además, tenemos que $\mathbf{D}^2 = \operatorname{diag}\{\lambda_1, \ldots, \lambda_r\}$ es una matriz diagonal que tiene los valores propios estrictamente positivos de $\mathbf{X}^\top \mathbf{X}$.

Hallemos, a continuación, una matriz $\mathbf{P} \in \mathbb{R}^{m \times m}$ ortogonal, que esté bajo las hipótesis del lema anterior. Particionemos la matriz $\mathbf{Q} = (\mathbf{Q}_1, \mathbf{Q}_2)$, donde las columnas de $\mathbf{Q}_1$ son las primeras r columnas de $\mathbf{Q}_1$.

Definamos, a continuación, $\mathbf{P}_1 = \mathbf{X}\mathbf{Q}_1\mathbf{D}^{-1}$. En la prueba del lema (A.2.2) probamos que $\mathbf{Q}_1^\top \mathbf{X}^\top \mathbf{X}\mathbf{Q}_1 = \mathbf{D}^2$. Luego

$$\mathbf{P}_1^\top \mathbf{P}_1 = \mathbf{D}^{-1}\mathbf{Q}_1 \mathbf{X}^\top \mathbf{X}\mathbf{Q}_1 \mathbf{D}^{-1} = \mathbf{D}^{-1}\mathbf{D}^2\mathbf{D}^{-1} = \mathbf{I}_r.$$

Por el teorema de rango-nulidad tenemos que

$$\dim[\ker(\mathbf{P}_1^\top)] = m - \operatorname{rango}(\mathbf{P}_1) = m - \operatorname{rango}(\mathbf{P}_1^\top \mathbf{P}_1) = m - r.$$

Consideremos entonces $\mathbf{P}_2$ cualquier matriz de dimensión $m \times (m - r)$ cuyas columnas formen una base ortonormal de $\ker(\mathbf{P}_1^\top)$. Luego,

$$\mathbf{P}^\top \mathbf{P} = \begin{pmatrix} \mathbf{P}_1^\top \mathbf{P}_1 & \mathbf{P}_1^\top \mathbf{P}_2 \\ \mathbf{P}_2^\top \mathbf{P}_2 & \mathbf{P}_2^\top \mathbf{P}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{I}_r & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{m-r} \end{pmatrix} = \mathbf{I}_m.$$

Por lo tanto, $\mathbf{P}$ es una matriz ortogonal y está bajo las hipótesis del lema (A.2.2). Por lo tanto,

$$\mathbf{P}^\top \mathbf{X}\mathbf{Q} = \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}$$

o, equivalentemente,

$$\mathbf{X} = \mathbf{P} \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \mathbf{Q}^\top,$$

como queríamos probar. ■

La SVD es usualmente escrita como $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$, donde

$$\mathbf{U} = \mathbf{P}, \boldsymbol{\Sigma} = \mathbf{D} \quad y \quad \mathbf{Q} = \mathbf{V},$$

de acuerdo con el teorema (A.2.1). En este trabajo seguiremos esta notación. Por otra parte, observemos que realizando las particiones de $\mathbf{U} = (\mathbf{U}_1, \mathbf{U}_2)$ y $\mathbf{V} = (\mathbf{V}_1, \mathbf{V}_2)$ vistas en la proposición (A.2.1) podemos escribir

$$\mathbf{X} = \mathbf{U}\begin{pmatrix} \boldsymbol{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}\mathbf{V}^\top = \mathbf{U}_1\boldsymbol{\Sigma}\mathbf{V}_1^\top.$$

A esta última descomposición, en donde $\boldsymbol{\Sigma}$ es una matriz diagonal y cuadrada se la conoce como SVD *compacta* o *económica*.

La razón del nombre *valores singulares* está relacionada con el hecho de que elementos de la diagonal $\boldsymbol{\Sigma}$ son los *valores singulares* de la matriz $\mathbf{X}$.

Definición A.2.1 (Valor singular de una matriz)**.** Dada una matriz $\mathbf{X} \in \mathbb{R}^{m\times n}$ decimos que σ es un valor singular de $\mathbf{X}$ si σ es un valor propio de $\mathbf{X}^\top\mathbf{X}$ (o, equivalentemente, de $\mathbf{X}\mathbf{X}^\top$).

Este hecho no lo presentaremos en este trabajo pero el lector interesado puede consultarlo en [32] en las páginas 552 y 553.

A.3. Descomposición QR

Proposición A.3.1 (Existencia de la descomposición QR)**.** *Sea* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *arbitraria. Existen* $\mathbf{Q} \in \mathbb{R}^{m\times m}$ *ortogonal y* $\mathbf{R} \in \mathbb{R}^{m\times n}$ *triangular superior tales que*

$$\mathbf{X} = \mathbf{Q}\mathbf{R}.$$

Demostración. Ver [22], página 246. ■

La particularidad de la descomposición QR es debido a la estructura de la matriz $\mathbf{Q}$: el espacio generado por las columnas de una matriz $\mathbf{X}$ es el mismo que el generado por las columnas de $\mathbf{U}$. Esto estará dado por la siguiente proposición.

Proposición A.3.2 (Estructura del espacio de columnas de la matriz U)**.** *Sea* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *una matriz de rango completo por columnas. Consideremos* $\mathbf{X} = \mathbf{Q}\mathbf{R}$ *la descomposición QR de* $\mathbf{X}$ *y las particiones por columnas*

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 \,\middle|\, \dots \,\middle|\, x_n \end{pmatrix} \quad y \quad \mathbf{Q} = \begin{pmatrix} \mathbf{q}_1 \,\middle|\, \dots \,\middle|\, \mathbf{q}_n \end{pmatrix}.$$

Luego, para todo $k = 1, \dots, n$*,*

$$\operatorname{gen}\{\mathbf{x}_1, \dots, \mathbf{x}_k\} = \operatorname{gen}\{\mathbf{q}_1, \dots, \mathbf{q}_k\} \quad y \quad r_{kk} \neq 0. \tag{A.3.1}$$

Más aún, escribiendo $\mathbf{Q} = (\mathbf{Q}_1, \mathbf{Q}_2)$*, con* $\mathbf{Q}_1 \in \mathbb{R}^{m\times n}$ *y* $\mathbf{Q}_2 \in \mathbb{R}^{m\times(m-n)}$ *y* $\mathbf{R} = \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{pmatrix}$*, con* $\mathbf{R}_1 \in \mathbb{R}^{n\times n}$ *tenemos que*

$$\operatorname{rango}(\mathbf{X}) = \operatorname{rango}(\mathbf{Q}_1), \quad \operatorname{nu}(\mathbf{X}) = \operatorname{nu}(\mathbf{Q}_2) \quad y \quad \mathbf{X} = \mathbf{Q}_1\mathbf{R}_1. \tag{A.3.2}$$

Demostración. Probemos en primer lugar que se cumple (A.3.1). Consideremos la k-ésima columna de $\mathbf{X}$. Debido a que $\mathbf{X} = \mathbf{QR}$ y que $\mathbf{R}$ es una matriz triangular superior tenemos que

$$\mathbf{x}_k = \sum_{i=1}^{k} r_{ik}\mathbf{q}_i. \tag{A.3.3}$$

Por lo tanto, $\mathbf{x}_k \in \text{gen}\{\mathbf{q}_1, \ldots, \mathbf{q}_k\}$. Luego $\text{gen}\{\mathbf{x}_1, \ldots, \mathbf{x}_k\} \subset \text{gen}\{\mathbf{q}_1, \ldots, \mathbf{q}_k\}$. Resta probar la inclusión inversa. Para esto, supongamos que $r_{kk} = 0$. Luego $\mathbf{x}_k = \sum_{i=1}^{k-1} r_{ik}\mathbf{q}_i$. Por la ecuación (A.3.3) para $k = 1$ tenemos que

$$\mathbf{x}_1 = r_{11}\mathbf{q}_1$$

y, luego

$$\mathbf{q}_1 = \frac{1}{r_{11}}\mathbf{x}_1.$$

La ecuación (A.3.3) para $k = 2$ nos dice,

$$\mathbf{x}_2 = r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2 = \frac{r_{12}}{r_{11}}\mathbf{x}_1 + r_{22}\mathbf{q}_2$$

y, luego,

$$\mathbf{q}_2 = \frac{1}{r_{22}}\mathbf{x}_2 - \frac{r_{12}}{r_{11}r_{22}}\mathbf{x}_1.$$

Análogamente, podemos escribir $\mathbf{q}_1, \ldots, \mathbf{q}_{k-1}$ en función de $\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}$. Luego, $\mathbf{x}_k$ puede ser escrito como combinación lineal de $\{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$. Pero esto contradice el hecho de que la matriz $\mathbf{X}$ es de rango completo por columnas. Por lo tanto, no puede existir k para el cual $r_{kk} = 0$. Luego, $\text{gen}\{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$ tiene dimensión k lo que prueba la afirmación (A.3.1).

Para probar la afirmación (A.3.2) notemos que

$$\mathbf{X} = \mathbf{QR} = \begin{pmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{pmatrix}\begin{pmatrix}\mathbf{R}_1 \\ \mathbf{0}\end{pmatrix} = \mathbf{Q}_1\mathbf{R}_1. \qquad \blacksquare$$

La descomposición que nos da la proposición (A.3.2) en la ecuación (A.3.2) la llamaremos *descomposición QR compacta*. A diferencia de la descomposición QR de la proposición (A.3.1), esta descomposición es única. El lector interesado puede consultar el teorema 5.2.3 de [22] para la demostración de este hecho.

Apéndice B

Diferenciación con respecto a vectores y a matrices

B.1. Diferenciación con respecto a vectores

Consideremos una función $f : \mathbb{R}^n \to \mathbb{R}^m$, con $f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))$. La derivada de f con respecto a un vector $\mathbf{x} \in \mathbb{R}^n$ la definiremos como la matriz jacobiana de f en $\mathbf{x}$, es decir, $\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$ es una matriz de $m \times n$ tal que

$$\left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)_{ij} = \frac{\partial f_i(\mathbf{x})}{\partial x_j}. \tag{B.1.1}$$

En particular, si f tiene a $\mathbb{R}$ como codominio tenemos que

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} & \frac{\partial f(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f(\mathbf{x})}{\partial x_n} \end{pmatrix}.$$

Proposición B.1.1 (Derivada del producto interno)**.** *Sean $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ vectores fijos. Consideremos las funciones $f_\mathbf{u}, f_\mathbf{v}$ dadas por $f_\mathbf{v}(\mathbf{x}) = \mathbf{x}^\top \mathbf{v}$ y $f_\mathbf{u}(\mathbf{y}) = \mathbf{u}^\top \mathbf{y}$. Luego*

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^\top \mathbf{v} = \mathbf{v}^\top \quad y \quad \frac{\partial}{\partial \mathbf{y}} \mathbf{u}^\top \mathbf{y} = \mathbf{u}^\top$$

Demostración. Observemos que fijado $\mathbf{v}$, tenemos que $f_\mathbf{v}(\mathbf{x}) = \sum_{i=1}^n x_i v_i$. Luego, por definición de derivada con respecto a un vector:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{x}} f_\mathbf{v}(\mathbf{x}) &= \begin{pmatrix} \frac{\partial f_\mathbf{v}(\mathbf{x})}{\partial x_1} & \frac{\partial f_\mathbf{v}(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_\mathbf{v}(\mathbf{x})}{\partial x_n} \end{pmatrix} \\ &= \begin{pmatrix} v_1 & v_2 & \cdots & v_n \end{pmatrix} = \mathbf{v}^\top \end{aligned}$$

Dados vectores $\mathbf{x} = (x_1, \ldots, x_n)^\top$ e $\mathbf{y} = (y_1, \ldots, y_n)^\top$ tenemos que

$$\frac{\partial}{\partial x_j} f_\mathbf{v}(\mathbf{x}) = v_j \quad \text{y} \quad \frac{\partial}{\partial y_j} f_\mathbf{u}(\mathbf{y}) = u_j.$$

Luego, por definición de derivada con respecto a un vector, obtenemos lo pedido. ■

Proposición B.1.2 (Derivada de una transformación lineal)**.** *Consideremos una matriz $\mathbf{A} \in \mathbb{R}^{m \times n}$ y la función $f : \mathbb{R}^n \to \mathbb{R}^m$ dada por $f(\mathbf{x}) = \mathbf{A}\mathbf{x}$. Luego*

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{A}. \tag{B.1.2}$$

Demostración. Sea $f_i(\mathbf{x})$ la i-ésima entrada de $f(\mathbf{x}) = \mathbf{A}\mathbf{x}$. Tenemos que

$$f_i(\mathbf{x}) = \sum_{k=1}^{n} a_{ik}x_j.$$

Luego

$$\frac{\partial f_i(\mathbf{x})}{x_j} = a_{ij}.$$

Obtenemos así, por definición de derivada con respecto a $\mathbf{x}$, la ecuación (B.1.2). ■

Proposición B.1.3. *Consideremos una matriz $\mathbf{A} \in \mathbb{R}^{n\times n}$ y la función $f : \mathbb{R}^n \to \mathbb{R}$ tal que $f(\mathbf{x}) = \mathbf{x}^\top\mathbf{A}\mathbf{x}$. Luego*

$$\frac{\partial}{\partial \mathbf{x}} f(\mathbf{x}) = \mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top).$$

Demostración. Por definición de f tenemos que

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_ix_j. \tag{B.1.3}$$

Derivando con respecto a x_k la ecuación (B.1.3) obtenemos

$$\begin{aligned}
\frac{\partial f}{\partial x_k}(x_1, \ldots, x_n) &= \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial}{\partial x_k} a_{ij}x_ix_j \\
&= \frac{\partial}{\partial x_k}\left(\sum_{i\neq k}^{n}\sum_{j=1}^{n} a_{ij}x_ix_j + \sum_{j=1}^{n} a_{kj}x_kx_j\right) \\
&= \sum_{i\neq k}^{n} a_{ik}x_i + \sum_{j\neq k}^{n} a_{kj}x_j + 2a_{kk}x_k \\
&= \sum_{i=1}^{n} a_{ik}x_i + \sum_{j=1}^{n} a_{kj}x_j = (\mathbf{A}^\top)_{k\cdot}\mathbf{x} + \mathbf{A}_{k\cdot}\mathbf{x},
\end{aligned}$$

donde $\mathbf{M}_{k\cdot}$ simboliza la k-ésima fila de una matriz $\mathbf{M}$. En consecuencia,

$$\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}) = \mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top).$$ ■

La proposición anterior se simplifica en el caso en que $\mathbf{A}$ sea simétrica:

Corolario B.1.3.1 (Derivada de una forma cuadrática)**.** *Sea $\mathbf{A} \in \mathbb{R}^{m\times n}$ una matriz simétrica y la función $f(\mathbf{x}) = \mathbf{x}^\top\mathbf{A}\mathbf{x}$. Luego*

$$\frac{\partial f}{\partial \mathbf{x}} = 2\mathbf{x}^\top\mathbf{A}.$$

Demostración. Se sigue directamente de la proposición (B.1.3) utilizando que $\mathbf{A} = \mathbf{A}^\top$. ■

Proposición B.1.4. *Consideremos $\mathbf{A} \in \mathbb{R}^{n\times n}$ y la función $f : \mathbb{R}^p \to \mathbb{R}$ tal que $f(\mathbf{x}) = \mathbf{x}^\top\mathbf{A}\mathbf{x}$, donde $\mathbf{x}$ es una función de un vector $\mathbf{z} \in \mathbb{R}^p$. Luego*

$$\frac{\partial}{\partial \mathbf{z}} f(\mathbf{x}) = \mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)\frac{\partial \mathbf{x}}{\partial \mathbf{z}}.$$

Demostración. Por la regla de la cadena multivariada:

$$\frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{z}}.$$

Por la proposición (B.1.3), el primer término de la derecha en la igualdad anterior es igual a $\mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)$. Luego

$$\frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} = \mathbf{x}^\top(\mathbf{A} + \mathbf{A}^\top)\frac{\partial \mathbf{x}}{\partial \mathbf{z}}.$$ ■

Corolario B.1.4.1 (Derivada de la norma 2 de una función)**.** *La derivada de la norma* 2 *de un vector* $f(\mathbf{x})$ *es*

$$\frac{\partial}{\partial \mathbf{x}} \|f(\mathbf{x})\|_2^2 = 2\mathbf{x}^\top \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}. \tag{B.1.4}$$

Demostración. Por definición de norma 2 tenemos que

$$\|f(\mathbf{x})\|_2^2 = f(\mathbf{x})^\top f(\mathbf{x}) = f(\mathbf{x})^\top \mathbf{I}_n f(\mathbf{x}).$$

Luego, la ecuación (B.1.4) sigue de la proposición (B.1.4) utilizando $\mathbf{A} = \mathbf{I}_n$. ■

B.2. Diferenciación con respecto a matrices

Consideremos una función $f : \mathbb{R}^{m\times n} \to \mathbb{R}$. Definimos la derivada de f con respecto a una matriz $\mathbf{X} = (x_{ij})$ como la matriz de dimensión $n \times m$ tal que

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \begin{pmatrix} \frac{\partial f(\mathbf{X})}{\partial x_{11}} & \frac{\partial f(\mathbf{X})}{\partial x_{12}} & \cdots & \frac{\partial f(\mathbf{X})}{\partial x_{n1}} \\ \frac{\partial f(\mathbf{X})}{\partial x_{21}} & \frac{\partial f(\mathbf{X})}{\partial x_{22}} & \cdots & \frac{\partial f(\mathbf{X})}{\partial x_{n2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(\mathbf{X})}{\partial x_{21}} & \frac{\partial f(\mathbf{X})}{\partial x_{22}} & \cdots & \frac{\partial f(\mathbf{X})}{\partial x_{nm}} \end{pmatrix}.$$

Proposición B.2.1 (Regla de la cadena: versión matricial)**.** *Sea* $\mathbf{H} : \mathbb{R}^d \to \mathbb{R}^{m\times n}$ *una función tal que*

$$\mathbf{H}(\mathbf{x}) = \begin{pmatrix} h_{11}(\mathbf{x}) & h_{12}(\mathbf{x}) & \cdots & h_{1n}(\mathbf{x}) \\ h_{21}(\mathbf{x}) & h_{22}(\mathbf{x}) & \cdots & h_{2n}(\mathbf{x}) \\ \vdots & \vdots & \ddots & \vdots \\ h_{m1}(\mathbf{x}) & h_{m2}(\mathbf{x}) & \cdots & h_{mn}(\mathbf{x}) \end{pmatrix},$$

y $g : \mathbb{R}^{m\times n} \to \mathbb{R}$*. Luego*

$$\begin{aligned} \frac{\partial}{\partial x_j} g(\mathbf{H}(\mathbf{x})) &= \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{\partial}{\partial y_{ij}} g(\mathbf{H}(\mathbf{x})) \frac{\partial\, h_{ij}(\mathbf{x})}{\partial x_j} \\ &= \operatorname{tr}\left[\left(\frac{\partial g(\mathbf{H}(\mathbf{x}))}{\partial \mathbf{Y}} \right)^\top \frac{\partial \mathbf{H}(\mathbf{x})}{\partial x_j} \right] \end{aligned}$$

Demostración. Ver [32], página 303. ■

Derivada de la función traza

Proposición B.2.2 (Derivada de la función traza)**.** *1. Sean* $\mathbf{A} \in \mathbb{R}^{n\times m}$ *y* $\mathbf{X} \in \mathbb{R}^{m\times n}$. *Luego*

$$\frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial \mathbf{X}} = \frac{\partial \operatorname{tr}(\mathbf{XA})}{\partial \mathbf{X}} = \mathbf{A}^\top.$$

2. Sean $\mathbf{A} \in \mathbb{R}^{n\times m}$ *y* $\mathbf{X} \in \mathbb{R}^{n\times m}$. *Luego*

$$\frac{\partial \operatorname{tr}(\mathbf{AX}^\top)}{\partial \mathbf{X}} = \frac{\partial \operatorname{tr}(\mathbf{X}^\top\mathbf{A})}{\partial \mathbf{X}} = \mathbf{A}.$$

Demostración. Daremos la prueba de la proposición (1). La prueba de (2) es análoga a esta última. Observemos que como $\mathbf{AX}$ y $\mathbf{XA}$ están ambos bien definidos, por la proposición (A.1.1), tenemos que

$$\frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial \mathbf{X}} = \frac{\partial \operatorname{tr}(\mathbf{XA})}{\partial \mathbf{X}}.$$

Resta probar que $\frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial \mathbf{X}} = \mathbf{A}^\top$. La entrada i, j de la matriz $\frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial \mathbf{X}}$ cumple que

$$\begin{aligned}\left(\frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial \mathbf{X}}\right)_{ij} = \frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial x_{ij}} &= \frac{\partial}{\partial x_{ij}}\sum_{k=1}^{n}(\mathbf{AX})_{kk} \\ &= \frac{\partial}{\partial x_{ij}}\sum_{k=1}^{n}(\mathbf{AX})_{kk} \\ &= \frac{\partial}{\partial x_{ij}}\sum_{k=1}^{n}\sum_{l=1}^{m} a_{kl}x_{lk} = a_{ji} = (\mathbf{A}^\top)_{ij}.\end{aligned}$$

Concluimos así lo pedido. ■

Proposición B.2.3. *Sean* $\mathbf{A} \in \mathbb{R}^{m\times m}, \mathbf{X} \in \mathbb{R}^{m\times n}$ *y* $\mathbf{B} \in \mathbb{R}^{n\times n}$. *Luego*

$$\frac{\partial \operatorname{tr}(\mathbf{X}^\top\mathbf{AXB})}{\partial \mathbf{X}} = \mathbf{AXB} + \mathbf{A}^\top\mathbf{XB}^\top$$

Derivada de la función determinante

Proposición B.2.4 (Derivada de la función determinante)**.** *Sea* $\mathbf{X} \in \mathbb{R}^{n\times n}$ *una matriz no estructurada, es decir, una matriz con* n^2 *entradas independientes. Luego*

$$\frac{\partial \det(\mathbf{X})}{\partial \mathbf{X}} = \operatorname{cof}(\mathbf{X}). \tag{B.2.1}$$

Demostración. Calculando el determinante por la i-ésima fila, tenemos que $\det(\mathbf{X}) = \sum_{j=1}^{n} x_{ij}c_{ij}$. Luego

$$\frac{\partial \det(\mathbf{X})}{\partial x_{ik}} = \sum_{j=1}^{n}\frac{\partial}{\partial x_{ik}}x_{ij}c_{ij} = c_{ik}.$$

De donde se concluye (B.2.1). ■

Consideremos ahora una generalización del resultado de (B.2.4). Consideremos una función $f : \mathbb{R}^m \to \mathbb{R}^{n\times n}$ y $g : \mathbb{R}^{n\times n} \to \mathbb{R}$ tal que $g(\mathbf{X}) = \det(\mathbf{X})$. Obtenemos así la siguiente proposición.

Proposición B.2.5. *Sea* $\mathbf{F} : \mathbb{R}^d \to \mathbb{R}^{m\times n}$ *una función diferenciable entonces*

$$\frac{\partial}{\partial x_j} \det(\mathbf{F}(\mathbf{x})) = \operatorname{tr}\left(\operatorname{adj}(\mathbf{F}(\mathbf{x}))\frac{\partial}{\partial x_j}\mathbf{F}(\mathbf{x})\right). \tag{B.2.2}$$

En particular, si $\mathbf{F}(\mathbf{x})$ *es invertible, podemos escribir*

$$\frac{\partial}{\partial x_j} \det(\mathbf{F}(\mathbf{x})) = \det(\mathbf{F}(\mathbf{x}))\operatorname{tr}\left(\mathbf{F}(\mathbf{x})^{-1}\frac{\partial}{\partial x_j}\mathbf{F}(\mathbf{x})\right). \tag{B.2.3}$$

Demostración. Por la proposición (B.2.1) y utilizando que la función $g : \mathbb{R}^{m\times n} \to \mathbb{R}$ dada por $g(\mathbf{X}) = \det(\mathbf{X})$ es de clase C^∞ tenemos que

$$\frac{\partial}{\partial x_j} \det(\mathbf{F}(\mathbf{x})) = \operatorname{tr}\left(\left(\frac{\partial \det(\mathbf{F}(\mathbf{x}))}{\partial \mathbf{Y}}\right)^\top \frac{\partial}{\partial x_j}\mathbf{F}(\mathbf{x})\right).$$

Luego, utilizando la derivada de la función determinante calculada en la proposición (B.2.4) obtenemos la ecuación (B.2.2).

Supongamos ahora que $F(\mathbf{x})$ es invertible. Por la proposición (B.2.2) podemos escribir

$$\operatorname{adj}(\mathbf{F}(\mathbf{x})) = \mathbf{F}(\mathbf{x})^{-1}\det(\mathbf{F}(\mathbf{x})).$$

Concluimos así, debido a que $\det(\mathbf{F}(\mathbf{x}))$ es un número real y la traza es una transformación lineal (ver proposición (A.1.1)), que se cumple la ecuación (B.2.3). ■

Proposición B.2.6. *Sea* $\mathbf{X}$ *una matriz invertible. Luego*

$$\frac{\partial \ln|\det(\mathbf{X})|}{\partial \mathbf{X}} = (\mathbf{X}^{-1})^\top.$$

Demostración. Consideremos el caso en que $\det(\mathbf{X}) > 0$. Luego

$$\frac{\partial \ln|\det(\mathbf{X})|}{\partial x_{ij}} = \frac{\partial \ln\det(\mathbf{X})}{\partial x_{ij}} = \frac{1}{\det(\mathbf{X})}\frac{\partial \det(\mathbf{X})}{\partial x_{ij}}.$$

Para calcular el factor de la derecha, consideremos la función $\mathbf{F} : \mathbb{R}^{n^2} \to \mathbb{R}^{n\times n}$ que toma un vector columna $\mathbf{x}$ y devuelve las entradas del vector $\mathbf{x}$ estructuradas en una matriz $\mathbf{X} \in \mathbb{R}^{n\times n}$. Luego, utilizando la proposición (B.2.5) tenemos que

$$\frac{\partial \det(\mathbf{X})}{\partial x_{ij}} = \det(\mathbf{X})\operatorname{tr}\left(\mathbf{X}^{-1}\frac{\partial \mathbf{X}}{\partial x_{ij}}\right) = \det(\mathbf{X})\operatorname{tr}\left(\mathbf{X}^{-1}\mathbf{e}_i\mathbf{e}_j^\top\right),$$

donde $\mathbf{e}_i$ es el i-ésimo vector canónico de dimensión n. A través del corolario (A.1.1.1), podemos escribir $\operatorname{tr}\left(\mathbf{X}^{-1}\mathbf{e}_i\mathbf{e}_j^\top\right) = \mathbf{e}_j^\top\mathbf{X}^{-1}\mathbf{e}_i$. Luego

$$\frac{\partial \ln\det(\mathbf{X})}{\partial x_{ij}} = \frac{1}{\det(\mathbf{X})}\det(\mathbf{X})\mathbf{e}_j^\top\mathbf{X}^{-1}\mathbf{e}_i = (\mathbf{X}^{-1})_{ji} = (\mathbf{X}^{-1})^\top_{ij},$$

lo que prueba lo pedido. El caso en que $\det(\mathbf{X}) < 0$ es análogo. ■

Proposición B.2.7. *Sea* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *y* $\lambda \in \mathbb{R}$ *tal que* $\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}_m$ *es una matriz invertible. Luego*

$$\frac{\partial}{\partial \mathbf{X}} \ln\det(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}_m) = 2(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}_m)\mathbf{X}.$$

Demostración. Fijemos un par (i, j). Por las proposiciones (B.2.5) y (B.2.4) tenemos que

$$\frac{\partial}{\partial x_{ij}} \ln\det(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m) = \operatorname{tr}\left((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1} \frac{\partial\, \mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m}{\partial x_{ij}}\right). \tag{B.2.4}$$

Calculemos el segundo factor dentro de $\operatorname{tr}(\cdot)$. En primer lugar observemos que

$$\frac{\partial}{\partial x_{ij}}(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m) = \frac{\partial}{\partial x_{ij}}(\mathbf{X}\mathbf{X}^\top),$$

puesto que $\lambda \mathbf{I}_m$ no depende de x_{ij} para ningún par (i, j). Calculemos entonces $\frac{\partial}{\partial x_{ij}}(\mathbf{X}\mathbf{X}^\top)$. Para esto, consideremos $f : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times m}$ dada por $f(\mathbf{X}) = \mathbf{X}\mathbf{X}^\top$, es decir,

$$f(\mathbf{X}) = \begin{pmatrix} f_{11}(\mathbf{X}) & f_{12}(\mathbf{X}) & \dots & f_{1m}(\mathbf{X}) \\ f_{21}(\mathbf{X}) & f_{22}(\mathbf{X}) & \dots & f_{2m}(\mathbf{X}) \\ \vdots & \vdots & \ddots & \vdots \\ f_{m1}(\mathbf{X}) & f_{m2}(\mathbf{X}) & \dots & f_{mm}(\mathbf{X}) \end{pmatrix}, \quad f_{k\ell}(\mathbf{X}) = \sum_{h=1}^{n} x_{kh} x_{\ell h}.$$

Luego, la derivada con respecto a x_{ij} de $f_{k\ell}(\mathbf{X})$ es

$$\frac{\partial}{\partial x_{ij}} f_{k\ell}(\mathbf{X}) = \begin{cases} x_{\ell j} & \text{si } k = i, \ell \neq i \\ x_{kj} & \text{si } \ell = i, k \neq i \\ 2x_{ij} & \text{si } k = \ell = i \\ 0 & \text{si } k \neq i \text{ o } \ell \neq i \end{cases}.$$

Utilizando la proposición (A.1.5) podemos escribir eso último como

$$\frac{\partial}{\partial x_{ij}} \mathbf{X}\mathbf{X}^\top = \boldsymbol{\delta}_{ij}\mathbf{X}^\top + \mathbf{X}\boldsymbol{\delta}_{ji}. \tag{B.2.5}$$

Uniendo las ecuaciones (B.2.4) y (B.2.5) obtenemos que

$$\begin{aligned} \frac{\partial}{\partial x_{ij}} \ln\det(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m) &= \operatorname{tr}\left((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}(\boldsymbol{\delta}_{ij}\mathbf{X}^\top + \mathbf{X}\boldsymbol{\delta}_{ji})\right) \\ &= \operatorname{tr}\left((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\boldsymbol{\delta}_{ij}\mathbf{X}^\top\right) + \operatorname{tr}\left((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\mathbf{X}\boldsymbol{\delta}_{ji})\right) \\ &\coloneqq S_1 + S_2. \end{aligned}$$

Observando que para todas $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m\times m}$ se cumple que

$$\operatorname{tr}(\mathbf{A}\boldsymbol{\delta}_{ij}\mathbf{B}) = (\mathbf{B}\mathbf{A})_{ji} = (\mathbf{A}^\top\mathbf{B}^\top)_{ij} \quad \text{y} \quad \operatorname{tr}(\mathbf{A}\boldsymbol{\delta}_{ji}) = (\mathbf{A}^\top)_{ji} = \mathbf{A}_{ij}$$

tenemos que

$$S_1 = \operatorname{tr}\left((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\boldsymbol{\delta}_{ij}\mathbf{X}^\top\right) = ((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\mathbf{X})_{ij}$$

y

$$S_2 = \operatorname{tr}\left((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\mathbf{X}\boldsymbol{\delta}_{ji})\right) = ((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\mathbf{X})_{ij}.$$

Concluimos entonces que

$$\frac{\partial}{\partial x_{ij}} = 2((\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)^{-1}\mathbf{X})_{ij}$$

y, por definición de derivada con respecto a matriz,

$$\frac{\partial}{\partial \mathbf{X}} \ln\det(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m) = 2(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I}_m)\mathbf{X}. \qquad \blacksquare$$

Apéndice C

Análisis de componentes principales via SVD

En este capítulo mostraremos cómo resolver el problema problema (2.0.4). Con este fin, planteemos, en primer lugar, la función lagrangeana del problema:

$$\begin{aligned}\mathscr{L} &:= \mathscr{L}(\boldsymbol{\mu}, \mathbf{U}, \mathbf{y}_1, \ldots, \mathbf{y}_n, \gamma, \Lambda) \\ &= \sum_{j=1}^{n} \|\mathbf{x}_j - \boldsymbol{\mu} - \mathbf{U}\mathbf{y}_j\|_2^2 + \gamma^\top \sum_{j=1}^{n} \mathbf{y}_j + \operatorname{tr}\left(\left(\mathbf{I}_d - \mathbf{U}^\top\mathbf{U}\right)\Lambda\right),\end{aligned}$$

donde $\gamma \in \mathbb{R}^p$ y Λ es una matriz de dimensión $p \times p$.

Hallemos los puntos estacionarios del problema. Comencemos por derivar la función lagrangeana con respecto a $\boldsymbol{\mu}$. En virtud del corolario (B.1.4.1) tenemos que

$$\frac{\partial}{\partial \boldsymbol{\mu}}\mathscr{L} = -2\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu} - \mathbf{U}\mathbf{y}_i)^\top = \mathbf{0}_p^\top. \tag{C.0.1}$$

Puesto a que estamos considerando la restricción $\sum_{i=1}^{n}\mathbf{y}_i = \mathbf{0}$, obtenemos

$$\left(\sum_{i=1}^{n}\mathbf{x}_i\right) - n\boldsymbol{\mu} - \mathbf{U}\underbrace{\left(\sum_{i=1}^{n}\mathbf{y}_i\right)}_{=\mathbf{0}} = \mathbf{0}_p$$

y, por lo tanto,

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i. \tag{C.0.2}$$

A continuación, derivemos la función lagrangeana con respecto a $\mathbf{y}_i$ para cada $i = 1, \ldots, n$. En virtud del corolario (B.1.4.1) y de la proposición (B.1.2) obtenemos que

$$2(\mathbf{x}_i - \boldsymbol{\mu} - \mathbf{U}\mathbf{y}_i)^\top(-\mathbf{U}) + \gamma^\top = \mathbf{0}_p^\top \tag{C.0.3}$$

Sumando con respecto a i estas derivadas obtenemos

$$\underbrace{-2\sum_{i=1}^{n}\mathbf{x}_i\mathbf{U}^\top + 2\sum_{i=1}^{n}\boldsymbol{\mu}^\top\mathbf{U}}_{S_1} + \underbrace{2\sum_{i=1}^{n}\mathbf{y}_i^\top\mathbf{U}^\top\mathbf{U}}_{S_2} + \gamma^\top = S_1 + S_2 + \gamma^\top = \mathbf{0}_p^\top.$$

Utilizando la ecuación (C.0.2) obtenemos que

$$S_1 = -2\sum_{i=1}^{n} \mathbf{x}_i \mathbf{U}^\top + 2\sum_{i=1}^{n} \boldsymbol{\mu}^\top \mathbf{U} = -2n\boldsymbol{\mu}^\top \mathbf{U} + 2n\boldsymbol{\mu}^\top \mathbf{U} = \mathbf{0}_p^\top.$$

Por otra parte, como exigimos que la matriz $\mathbf{U}$ sea ortogonal y que $\sum_{i=1}^{n} \mathbf{y}_i = \mathbf{0}$ tenemos que

$$S_2 = 2\sum_{i=1}^{n} \mathbf{y}_i^\top \mathbf{I}_p = 2\sum_{i=1}^{n} \mathbf{y}_i^\top = \mathbf{0}_p^\top.$$

Por lo tanto, como $S_1 = S_2 = \mathbf{0}_p^\top$, obtenemos que el vector de multiplicadores de Lagrange es $\gamma = \mathbf{0}_p$. Sustituyendo esto último en la ecuación (C.0.3) obtenemos

$$\mathbf{y}_i = \mathbf{U}^\top(\mathbf{x}_i - \boldsymbol{\mu}). \tag{C.0.4}$$

Esto implica que el vector $\mathbf{y}_i \in \mathbb{R}^d$ es el vector que retiene las coordenadas de la proyección del vector $\mathbf{x}_i$ en el subespacio afín $\mathcal{S}$.

Antes de optimizar sobre $\mathbf{U}$ reemplazaremos los valores hallados para $\boldsymbol{\mu}$ e $\mathbf{y}_i$ en la función objetivo del problema (2.0.4). Esto nos lleva al siguiente problema

$$\begin{aligned} \underset{\mathbf{U}}{\text{minimizar}} \quad & \sum_{i=1}^{n} \left\| (\mathbf{x}_i - \boldsymbol{\mu}) - \mathbf{U}\mathbf{U}^\top(\mathbf{x}_i - \boldsymbol{\mu}) \right\|^2 \\ \text{sujeto a} \quad & \mathbf{U}^\top \mathbf{U} = \mathbf{I}_p \end{aligned} \tag{C.0.5}$$

Resta encontrar la matriz $\mathbf{U}$ óptima del problema. Para esto utilizaremos la SVD de la matriz $\tilde{\mathbf{X}} = \mathbf{X} - \boldsymbol{\mu}$, donde $\boldsymbol{\mu}$ está dado por (C.0.2). Esto lo dará el siguiente teorema:

Teorema C.0.1 (PCA via SVD)**.** *Sea $\tilde{\mathbf{X}} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ la matriz formada por los datos centrados puestos como vectores columnas. Consideremos $\mathbf{X} = \mathbf{U}_X \boldsymbol{\Sigma}_X \mathbf{V}_X^\top$ la SVD compacta de $\mathbf{X}$ de tal forma que*

$$\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \ldots, \sigma_r) \quad \textit{con } \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$

Luego para un $p < d$, la solución óptima de $\mathbf{U}$ del problema (C.0.5) está dada por las primeras p columnas de $\mathbf{U}_X$, la solución óptima para $\mathbf{y}_i$ está dada por la i-ésima columna de la submatriz $\boldsymbol{\Sigma}_X \mathbf{V}_X^\top$ y el valor óptimo de la función objetivo del problema (C.0.5) está dado por $\sum_{i=p+1}^{d} \sigma_i^2$, donde σ_i es el i-ésimo valor singular de $\mathbf{X}$.

Demostración. En primer lugar, reescribamos la función objetivo de la siguiente forma:

$$\begin{aligned} \sum_{i=1}^{n} \left\| \mathbf{x}_i - \mathbf{U}\mathbf{U}^\top \mathbf{x}_i \right\|^2 &= \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{U}\mathbf{U}^\top \mathbf{x}_i)^\top (\mathbf{x}_i - \mathbf{U}\mathbf{U}^\top \mathbf{x}_i) \\ &= \sum_{i=1}^{n} \mathbf{x}_i^\top (\mathbf{I}_d - \mathbf{U}\mathbf{U}^\top)^\top (\mathbf{I}_d - \mathbf{U}\mathbf{U}^\top) \mathbf{x}_i \\ &= \sum_{i=1}^{n} \mathbf{x}_i^\top (\mathbf{I}_d - \mathbf{U}\mathbf{U}^\top) \mathbf{x}_i \end{aligned}$$

En virtud del corolario (A.1.1.1), podemos escribir

$$\sum_{i=1}^{n} \mathbf{x}_i^\top (\mathbf{I}_d - \mathbf{U}\mathbf{U}^\top)\mathbf{x}_i = \sum_{i=1}^{n} \operatorname{tr}((\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{x}_i\mathbf{x}_i^\top)$$
$$= \operatorname{tr}((\mathbf{I} - \mathbf{U}\mathbf{U}^\top)\mathbf{X}\mathbf{X}^\top) = \operatorname{tr}(\mathbf{X}\mathbf{X}^\top) - \operatorname{tr}(\mathbf{U}\mathbf{U}^\top\mathbf{X}\mathbf{X}^\top).$$

Puesto que $\operatorname{tr}(\mathbf{X}\mathbf{X}^\top)$ no depende de $\mathbf{U}$, el problema (C.0.5) es equivalente a

$$\begin{array}{ll} \underset{\mathbf{U}}{\text{maximizar}} & \operatorname{tr}(\mathbf{U}\mathbf{U}^\top\mathbf{X}\mathbf{X}^\top) \\ \text{sujeto a} & \mathbf{U}^\top\mathbf{U} = \mathbf{I}_p \end{array}. \tag{C.0.6}$$

Utilizando la proposición (A.1.1), podemos escribir el lagrangeano del problema (C.0.6) como

$$\mathscr{L} = \operatorname{tr}(\mathbf{U}^\top\mathbf{X}\mathbf{X}^\top\mathbf{U}) + \operatorname{tr}((\mathbf{I}_d - \mathbf{U}^\top\mathbf{U})\Lambda)).$$

Para derivar el lagrangeano con respecto a $\mathbf{U}$ utilizaremos el corolario (B.2.3), obtenemos así

$$\frac{\partial}{\partial \mathbf{U}}\mathscr{L} = 2\mathbf{X}\mathbf{X}^\top\mathbf{U} - 2\mathbf{U}\Lambda = \mathbf{0}$$

de donde hallamos que Λ debe cumplir que

$$\Lambda = \mathbf{U}^\top\mathbf{X}\mathbf{X}^\top\mathbf{U} \tag{C.0.7}$$

y, por lo tanto, la función objetivo del problema (C.0.6) es igual a $\operatorname{tr}(\Lambda)$.

Probemos a continuación que, sin pérdida de generalidad, podemos elegir a Λ como una matriz diagonal. Recordemos que la matriz $\mathbf{U}$ está definida a menos de multiplicaciones por matrices ortogonales; esto implica que si $\mathbf{U}$ es solución del problema (C.0.6), también lo es $\tilde{\mathbf{U}} = \mathbf{U}\mathbf{R}$. Como la matriz Λ hallada en (C.0.7) es simétrica, puede escribirse como

$$\Lambda = \mathbf{P}\mathbf{D}\mathbf{P}^\top.$$

con $\mathbf{D}$ una matriz diagonal y $\mathbf{P}$ ortogonal. Consideremos $\mathbf{R}$ como la matriz de vectores propios de Λ. Luego la matriz

$$\tilde{\Lambda} = \mathbf{R}\mathbf{U}^\top\mathbf{X}\mathbf{X}^\top\mathbf{U}\mathbf{R}^\top = \mathbf{D},$$

es solución del problema y es diagonal.

Por lo tanto, podemos pensar a Λ, sin pérdida de generalidad, como una matriz diagonal. Se sigue de la ecuación (C.0.7) que las columnas de $\mathbf{U}$ deben ser los primeros p vectores propios de $\mathbf{X}\mathbf{X}^\top$. Como la función objetivo es igual a $\operatorname{tr}(\Lambda)$ concluimos que la solución óptima está dada por los primeros p valores propios de $\mathbf{X}\mathbf{X}^\top$ o, equivalentemente, por los primeros p valores singulares de $\mathbf{X} = \mathbf{U}_X\mathbf{\Sigma}_X\mathbf{V}_X^\top$.

Por otra parte, por lo hallado en la ecuación (C.0.4) tenemos que

$$\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] = \mathbf{U}^\top\mathbf{X} = \mathbf{U}^\top\mathbf{U}_X\mathbf{\Sigma}_X\mathbf{V}_X^\top = \mathbf{\Sigma}_X\mathbf{V}_X^\top.$$

Finalmente, como

$$\Lambda = \mathbf{U}^\top\mathbf{U}_X\mathbf{\Sigma}_X^2\mathbf{U}_X^\top\mathbf{U} = \mathbf{\Sigma}^2,$$

la función objetivo de (C.0.5) está dada por $\operatorname{tr}(\mathbf{\Sigma}_X^2) - \operatorname{tr}(\mathbf{\Sigma}^2) = \sum_{i=p+1}^{d} \sigma_i^2$, donde σ_i es el i-ésimo valor singular de $\mathbf{X}$. ■

Apéndice D

Funciones convexas

D.1. Definición y resultados fundamentales

Definición D.1.1 (Función convexa)**.** Una función $f : \mathbb{R}^n \to \mathbb{R}$ es convexa si para todo $t \in [0,1]$ y todo $x_1, x_2 \in \mathbb{R}^n$ se cumple que

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2).$$

Proposición D.1.1 (Las normas son funciones convexas)**.** *Sea $\|\cdot\|$ una norma en $\mathbb{R}^n$. Entonces $\|\cdot\|$ es convexa.*

Demostración. Sean $x, y \in \mathbb{R}^n$ y $t \in [0,1]$. Por definición de norma tenemos que

$$\|tx + (1-t)y\| \leq \|tx\| + \|(1-t)y\| = t\|x\| + (1-t)\|y\|,$$

lo que prueba que $\|\cdot\|$ es convexa. ■

Proposición D.1.2 (Combinación lineal de funciones convexas)**.** *Sean $f_1, \ldots, f_k : \mathbb{R}^n \to \mathbb{R}$ funciones convexas y $\{a_i\}_{i=1}^k \subset \mathbb{R}_{\leq 0}$. Luego $g(x) = \sum_{i=1}^k a_i f_i(x)$ es una función convexa.*

Demostración. Sean $x, y \in \mathbb{R}^n$ y $t \in [0,1]$. Se tiene que

$$\begin{aligned} g(tx + (1-t)y) = \sum_{i=1}^k a_i f_i(tx + (1-t)y) &\leq \sum_{i=1}^k a_i t f_i(x) + (1-t)f_i(y) \\ &= t\sum_{i=1}^k a_i f_i(x) + (1-t)\sum_{i=1}^k a_i f_i(y) \\ &= tg(x) + (1-t)g(y), \end{aligned}$$

donde en la primer desigualdad se utilizó que f_i es convexa y $a_i \geq 0$ para todo $i = 1, \ldots, k$. ■

Proposición D.1.3 (Desigualdad de Jensen: versión finita)**.** *Sean $\{a_i\}_{i=1}^n \subset \mathbb{R}_+$ y $f : D \subset \mathbb{R} \to \mathbb{R}$ convexa. Luego para todos $x_1, \ldots, x_n \in D$ tenemos que*

$$f\left(\frac{\sum_{i=1}^n a_i x_i}{\sum_{i=1}^n a_i}\right) \leq \frac{\sum_{i=1}^n a_i f(x_i)}{\sum_{i=1}^n a_i}. \tag{D.1.1}$$

Si f es cóncava, la desigualdad es

$$f\left(\frac{\sum_{i=1}^n a_i x_i}{\sum_{i=1}^n a_i}\right) \leq \frac{\sum_{i=1}^n a_i f(x_i)}{\sum_{i=1}^n a_i}. \tag{D.1.2}$$

Demostración. La prueba será por inducción en n. Consideremos en primer lugar el caso $n = 2$. Como f es convexa, para todos $x_1, x_n \in D$ tenemos que

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2), \quad t \in [0,1]$$

En particular, si tomamos $0 < t = \frac{a_1}{a_1+a_2} \leq 1$ tenemos que

$$f\left(\frac{a_1 x_1}{a_1+a_2} + \frac{a_2 x_1}{a_1+a_2}\right) \leq \frac{a_1 f(x_1)}{a_1+a_2} + \frac{a_1 f(x_2)}{a_1+a_2}$$

como queríamos probar. Supongamos ahora que la ecuación (D.1.1) se cumple para $n = k$ y probemos que se cumple para $n = k+1$. En primer lugar notemos que

$$\begin{aligned}
\frac{\sum_{i=1}^{k+1} a_i x_i}{\sum_{i=1}^{k+1} a_i} &= \frac{\sum_{i=1}^{k} a_i x_i}{\sum_{i=1}^{k+1} a_i} + \frac{a_{k+1}x_{k+1}}{\sum_{i=1}^{k+1} a_i} \\
&= \frac{\sum_{i=1}^{k} a_i}{\sum_{i=1}^{k+1} a_j}\left(\frac{1}{\sum_{j=1}^{k} a_j}\sum_{i=1}^{k} a_i x_i\right) + \frac{a_{k+1}x_{k+1}}{\sum_{i=1}^{k+1} a_i} \\
&:= (1-t)\left(\frac{1}{\sum_{j=1}^{k} a_j}\sum_{i=1}^{k} a_i x_i\right) + tx_{k+1}, \quad t = \frac{a_{k+1}}{\sum_{i=1}^{k+1} a_i}
\end{aligned}$$

Luego

$$\begin{aligned}
f\left(\frac{\sum_{i=1}^{k+1} a_i x_i}{\sum_{i=1}^{n} a_i}\right) &= f\left((1-t)\left(\frac{1}{\sum_{j=1}^{k} a_j}\sum_{i=1}^{k} a_i x_i\right) + tx_{k+1}\right) \\
&\leq (1-t) f\left(\frac{\sum_{i=1}^{k} a_i x_i}{\sum_{j=1}^{k} a_j}\right) + tf(x_{k+1}) \\
&\leq (1-t)\frac{\sum_{i=1}^{k} a_i f(x_i)}{\sum_{i=1}^{k} a_i} + tf(x_{k+1}) \\
&= \frac{\sum_{i=1}^{k+1} a_i f(x_i)}{\sum_{i=1}^{k+1} a_i},
\end{aligned}$$

donde en la primera desigualdad utilizamos que f es convexa y en la segunda desigualdad utilizamos la hipótesis inductiva. Queda así probada la ecuación (D.1.1). La ecuación (D.1.2) se sigue directamente de la definición de función cóncava y la ecuación (D.1.1). ■

D.2. Preliminares sobre subdiferenciales

Las funciones convexas pueden no tener definido al gradiente en todo punto. Sin embargo, existe un concepto que generaliza el concepto de gradiente: el de subgradiente. Presentaremos en esta sección su definición y algunos resultados esenciales. El desarrollo de esta sección sigue el capítulo 3 de [34].

Dados un espacio vectorial V y una función $f : V \to \mathbb{R} \cup \{-\infty, +\infty\}$, diremos que f es una función *propia* si el elemento $-\infty$ no tiene preimagen por la función f.

Definición D.2.1 (Subgradiente y subdiferencial)**.** Sean V un espacio vectorial con producto interno, $f : V \to (-\infty, \infty]$ una función propia y $x_0 \in V$. Decimos que un vector $v \in V$ es un ***subgradiente*** de f por el punto x_0 si

$$f(x) \geq f(x_0) + \langle v, x - x_0 \rangle$$

Al conjunto de subgradientes de f por un punto x_0 lo llamaremos el ***subdiferencial*** de f por x_0 y lo notaremos como $\partial f(x_0)$:

$$\partial f(x_0) \coloneqq \{v \in V : \forall x \in V, f(x) \geq f(x_0) + \langle v, x - x_0 \rangle\}$$

La derivada direccional de una función f con respecto a un vector v está estréchamente ligada al concepto de subgradiente, como lo muestra la siguiente proposición.

Proposición D.2.1. *Sea $f : V \to (-\infty, +\infty]$ una función convexa y propia. Para todo $x_0 \in V$ y $v \in V$ se tiene que*

$$\frac{\partial f}{\partial v}(x_0) = \max_{d \in \partial f(x_0)} \{\langle d, v \rangle\}$$

Demostración. Ver [34], teorema 3.26. ■

Proposición D.2.2. *Sean $f_1, \ldots, f_m : V \to (-\infty, +\infty]$ funciones convexas y propias. Sea $x \in \bigcap_{i=1}^m dom(f_i)$. Se cumple que*

$$\sum_{i=1}^m \partial f_i(x_0) \subset \partial \left(\sum_{i=1}^m f_i \right)(x_0),$$

donde la suma del lado derecho debe entenderse como suma de Minkowski. Si además se tiene que $x \in \bigcap_{i=1}^n int(dom(f_i))$ entonces

$$\sum_{i=1}^m \partial f_i(x_0) = \partial \left(\sum_{i=1}^m f_i \right)(x_0),$$

Demostración. Probaremos la primera de las tesis. La prueba de la segunda tesis, ligéramente más técnica, puede consultarse en [34].

Sea $v \in \sum_{i=1}^m \partial f_i(x_0)$. Por definición, existen $v_1 \in \partial f_1(x_0), \ldots, v_m \in \partial f_m(x_0)$ tales que $v = v_1 + \ldots v_m$. Luego

$$\begin{aligned} f_1(x) + \ldots f_m(x) &\geq f_1(x_0) + \langle v_1, x - x_0 \rangle + \cdots + f_m(x_0) + \langle v_m, x - x_0 \rangle \\ &= f_1(x_0) + \cdots + f_m(x_0) + \langle v_1 + \cdots + v_m, x - x_0 \rangle, \end{aligned}$$

para todo $x \in V$. Esto muestra que $v = v_1 + \cdots + v_m$ es un subgradiente de $\sum_{i=1}^m f_i$ en el punto x_0. ■

El concepto de subgradiente extiende el concepto de gradiente en el siguiente sentido: si una función es diferenciable entonces existe un único subgradiente (el gradiente propiamente dicho) pero si no lo es, el subdiferencial es un conjunto de más de un elemento.

Proposición D.2.3. *Sea $f : V \to (-\infty, +\infty]$ una función convexa y propia. Sea $x_0 \in V$. Si f es diferenciable en x_0 entonces se cumple que $\partial f(x_0) = \{\nabla f(x_0)\}$. A la inversa, si el subdiferencial de f en x_0 tiene un solo elemento, entonces f es diferenciable en x_0 y $\partial f(x_0) = \{\nabla f(x_0)\}$.*

Demostración. Probaremos la primera de las tesis. Una prueba de la segunda puede verse en [34], teorema 3.33.

Sea $v \in V$. Como f es diferenciable se tiene que

$$\frac{\partial f}{\partial v}(x_0) = \langle \nabla f(x_0), v \rangle. \tag{D.2.1}$$

Sea $d \in \partial f(x_0)$. Vamos a probar que $d = \nabla f(x_0)$. Combinando la ecuación (D.2.1) con la proposición (D.2.1) se tiene que

$$\langle \nabla f(x_0), v \rangle = \frac{\partial f}{\partial v}(x_0) \geq \langle d, v \rangle.$$

Luego, utilizando la linealidad del producto interno

$$\langle d - \nabla f(x_0), v \rangle \leq 0$$

Tomando máximo sobre todas las direcciones tales que $\|v\| \leq 1$ se tiene que

$$\|d - \nabla f(x_0)\|_* = \max_{\|v\| \leq 1} \langle d - \nabla f(x_0), v \rangle \leq 0,$$

lo que implica que $d = \nabla f(x_0)$. ■

Recordemos que si $f : \mathbb{R}^n \to \mathbb{R}$ es diferenciable y presenta un mínimo en un punto x^* entonces $\nabla_f(x_0) = \mathbf{0}$. La siguiente proposición nos da una condición análoga para el caso de funciones con subdiferenciales.

Proposición D.2.4 (Condición de optimalidad)**.** *Sea $f : V \to (-\infty, +\infty]$ una función convexa y propia. Son equivalentes las siguientes afirmaciones:*

1) $x^* \in \arg\min_{x \in V}\{f(x)\}$

2) $\mathbf{0} \in \partial f(x^*)$.

Demostración. Supongamos que $x^* \in \arg\min_{x \in V}\{f(x)\}$. Por definición de mínimo se tiene que $f(x) \geq f(x^*)$ para todo $x \in V$. Luego,

$$f(x) \geq f(x^*) = f(x^*) + \langle 0, x - x^* \rangle,$$

para todo $x \in V$, lo que implica que $\mathbf{0} \in \partial f(x^*)$.

La prueba de la equivalencia contraria es análoga. ■

D.3. Norma nuclear

En esta sección probaremos que la función norma nuclear es efectivamente una norma. Para esto probemos, en primer lugar, el siguiente lema técnico.

Lema D.3.1. *La norma nuclear de una matriz $\mathbf{X} \in \mathbb{R}^{m \times n}$ puede ser escrita como*

$$\|\mathbf{X}\|_* = \sup_{\sigma_1(\mathbf{Q}) \leq 1} \operatorname{tr}(\mathbf{Q}^\top \mathbf{X}) = \sup_{\sigma_1(\mathbf{Q}) \leq 1} \langle \mathbf{Q}, \mathbf{X} \rangle_F,$$

donde $\sigma_1(\mathbf{Q})$ es el valor singular más grande.

Demostración. Para demostrar este lema probaremos las dos desigualdades de la tesis.

($\leq$) Sea $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ la descomposición en valores singulares de $\mathbf{X}$. Definimos

$$\mathbf{Q}_0 \coloneqq \mathbf{U}\mathbf{V}^\top = \mathbf{U}\mathbf{I}_r\mathbf{V}^\top,$$

donde $\mathbf{I}_r$ es la matriz identidad en $\mathbb{R}^{r\times r}$. Tenemos que $\sigma_1(\mathbf{Q}_0) = 1$ lo que implica que $\mathbf{Q}_0$ pertenece al espacio de matrices $\mathbf{Q} \in \mathbb{R}^{m\times n}$ tales que $\sigma_1(\mathbf{Q}) \leq 1$. Además, utilizando el corolario (A.1.1.1),

$$\begin{aligned}\operatorname{tr}(\mathbf{Q}_0^\top\mathbf{X}) = \operatorname{tr}((\mathbf{U}\mathbf{V}^\top)^\top(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top)) &= \operatorname{tr}(\mathbf{V}\mathbf{U}^\top\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top)\\ &= \operatorname{tr}(\mathbf{V}^\top\mathbf{V}\mathbf{U}^\top\mathbf{U}\mathbf{\Sigma})\\ &= \operatorname{tr}(\mathbf{\Sigma}) = \sum_{i=1}^{r}\sigma_i(\mathbf{X}) = \|\mathbf{X}\|_*.\end{aligned}$$

Luego

$$\sup_{\sigma_1(\mathbf{Q})\leq 1}\operatorname{tr}(\mathbf{Q}^\top\mathbf{X}) \geq \operatorname{tr}(\mathbf{Q}_0^\top\mathbf{X}) = \|\mathbf{X}\|_*.$$

($\geq$) Consideremos nuevamente la SVD de $\mathbf{X}$, es decir, escribamos $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = \sum_{i=1}^{r}\sigma_i u_i v_i^\top$. Tenemos que

$$\begin{aligned}\sup_{\sigma_1(\mathbf{Q})\leq 1}\operatorname{tr}(\mathbf{Q}^\top\mathbf{X}) = \sup_{\sigma_1(\mathbf{Q})\leq 1}\operatorname{tr}(\mathbf{Q}^\top\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top) &= \sup_{\sigma_1(\mathbf{Q})\leq 1}\operatorname{tr}(\mathbf{V}^\top\mathbf{Q}^\top\mathbf{U}\mathbf{\Sigma})\\ &= \sup_{\sigma_1(\mathbf{Q})\leq 1}\sum_{i=1}^{n}\sigma_i\sigma_i(\mathbf{U}^\top\mathbf{Q}\mathbf{V})_{ii}\\ &= \sup_{\sigma_1(\mathbf{Q})\leq 1}\sum_{i=1}^{n}\sigma_i\mathbf{u}_i^\top\mathbf{Q}\mathbf{v}_i\\ &\leq \sup_{\sigma_1(\mathbf{Q})\leq 1}\sum_{i=1}^{n}\sigma_i\mathbf{u}_i^\top\sigma_1(\mathbf{Q})\mathbf{v}_i = \|\mathbf{X}\|_*\end{aligned}$$

■

Proposición D.3.1 (La norma nuclear es una norma)**.** $\|\cdot\|_*\ \mathbb{R}^{m\times n}\to\mathbb{R}$ *es una norma.*

Demostración. • $\|\mathbf{X}\|_* = \mathbf{0}$ si y sólo si $\mathbf{X} = \mathbf{0}$

($\Leftarrow$) La matriz nula $\mathbf{0} \in \mathbb{R}^{m\times n}$ puede ser escrita en su descomposición SVD como

$$\mathbf{0} = \mathbf{U}\mathbf{0}\mathbf{V}$$

donde $\mathbf{U} \in \mathbb{R}^{m\times r}$ y $\mathbf{V} \in \mathbb{R}^{m\times r}$ son matrices tales que $\mathbf{U}^\top\mathbf{U} = \mathbf{V}^\top\mathbf{V} = \mathbf{I}_r$, con $r = \min\{m, n\}$. Luego

$$\|\mathbf{0}\|_* = \sum_{i=1}^{\min\{m,n\}}\sigma_i(\mathbf{0}) = \sum_{i=1}^{\min\{m,n\}}0 = 0$$

($\Rightarrow$) Sea $\mathbf{X} \in \mathbb{R}^{m\times n}$ tal que $\|\mathbf{X}\| = 0$. Por definición

$$\sum_{i=1}^{\min\{m,n\}}\sigma_i(\mathbf{X}) = 0.$$

Como $\sigma_i(\mathbf{X}) \geq 0$ para todo $i = 1, \ldots, \text{mín}\{m, n\}$, se debe cumplir que $\sigma_i(\mathbf{X}) = 0$ para todo i. Luego, escribiendo a $\mathbf{X}$ en su SVD obtenemos

$$\mathbf{X} = \mathbf{U0V}^\top = \mathbf{0}.$$

- $\|\mathbf{X}\|_* \geq 0$ para todo $\mathbf{X} \in \mathbb{R}^{m \times n}$

  Como los valores singulares $\sigma_i(\mathbf{X})$ son no negativos para todo $i = 1, \ldots, \text{mín}\{m, n\}$ se concluye que $\|\mathbf{X}\|_* \geq 0$.

- $\|\lambda\mathbf{X}\|_* = |\lambda| \, \|\mathbf{X}\|_*$ para todo $\lambda \in \mathbb{R}$ y $\mathbf{X} \in \mathbb{R}^{m \times n}$

  Sea $\mathbf{X} \in \mathbb{R}^{m \times n}$ y $\lambda \in \mathbb{R}$. Los valores singulares de $\lambda\mathbf{X}$ son las raíces cuadradas de $(\lambda\mathbf{X})^\top(\lambda\mathbf{X}) = \lambda^2\mathbf{X}^\top\mathbf{X}$. Luego

$$\|\lambda\mathbf{X}\|_* = \sum_{i=1}^{r} \sigma_i(\lambda\mathbf{X}) = \sum_{i=1}^{r} |\lambda|\sigma_i(\mathbf{X}) = |\lambda| \, \|\mathbf{X}\|_* .$$

- $\|\mathbf{X} + \mathbf{Y}\|_* \leq \|\mathbf{X}\|_* + \|\mathbf{Y}\|_*$

  Utilizando el lema (D.3.1) tenemos que

$$\begin{aligned} \|\mathbf{X} + \mathbf{Y}\|_* &= \sup_{\sigma_1(\mathbf{Q}) \leq 1} \langle \mathbf{Q}, \mathbf{X} \rangle \\ &= \sup_{\sigma_1(\mathbf{Q}) \leq 1} \langle \mathbf{Q}, \mathbf{X} \rangle + \langle \mathbf{Q}, \mathbf{Y} \rangle \\ &\leq \sup_{\sigma_1 \leq 1} \langle \mathbf{Q}, \mathbf{X} \rangle + \sup_{\sigma_1 \leq 1} \langle \mathbf{Q}, \mathbf{Y} \rangle = \|\mathbf{X}\|_* + \|\mathbf{Y}\|_*. \end{aligned}$$

■

Apéndice E

Estimadores de PPCA

En este capítulo daremos la prueba de la proposición (4.1.1). Esta prueba está basada en las pruebas de [35] y [17].

Demostración. Calculemos las derivadas de la función de log-verosimilitud descrita en (4.1.4) con respecto a sus parámetros.

Utilizando la proposición (B.1.4) y el hecho de que $\mathbf{\Sigma_X}$ es simétrica, tenemos que

$$\frac{\partial}{\partial \mu_{\mathbf{x}}} \ell(\boldsymbol{\mu}_{\mathbf{x}}, \mathbf{\Sigma}_{\mathbf{x}}) = \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}})^\top \mathbf{\Sigma}_{\mathbf{x}}^{-1} = \mathbf{0}_d^\top$$

si y solo sí

$$\boldsymbol{\mu}_{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i = \bar{\mathbf{x}}_n.$$

Sustituyendo esto en la log-verosimilitud obtenemos

$$\begin{aligned} \ell(\bar{\mathbf{x}}_n, \mathbf{\Sigma}_{\mathbf{x}}) &= -\frac{nd}{2} \ln(2\pi) - \frac{n}{2} \ln(\det(\mathbf{\Sigma}_{\mathbf{x}})) - \frac{1}{2} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}}_n)^\top \mathbf{\Sigma}_{\mathbf{x}}^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_n) \\ &= -\frac{nd}{2} \ln(2\pi) - \frac{n}{2} \ln(\det(\mathbf{\Sigma}_{\mathbf{x}})) - \frac{n}{2} \operatorname{tr}(\mathbf{\Sigma}_{\mathbf{x}}^{-1} \widehat{\mathbf{\Sigma}}_n) \\ &\coloneqq S_1 + S_2 + S_3, \end{aligned} \tag{E.0.1}$$

donde en la segunda igualdad se utilizó que, debido a la proposición (A.1.1) y al corolario (A.1.1.1) tenemos que

$$\begin{aligned} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}}_n)^\top \mathbf{\Sigma}_{\mathbf{x}}^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_n) &= \operatorname{tr}\left[\sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}}_n)^\top \mathbf{\Sigma}_{\mathbf{x}}^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_n) \right] \\ &= \operatorname{tr}\left[\left(\sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}}_n)^\top \mathbf{\Sigma}_{\mathbf{x}}^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_n) \right)^\top \right] \\ &= \operatorname{tr}\left[\mathbf{\Sigma}_{\mathbf{x}}^{-1} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}}_n)(\mathbf{x}_i - \bar{\mathbf{x}}_n)^\top \right] \\ &= \operatorname{tr}(\mathbf{\Sigma}_{\mathbf{x}}^{-1} \widehat{\mathbf{\Sigma}}_n). \end{aligned}$$

Recordemos, por la ecuación (4.1.3) que la matriz $\mathbf{\Sigma}_{\mathbf{x}}$ depende de $\mathbf{U}$ y de σ^2. Derivemos, a continuación, la log-verosimilitud con respecto a $\mathbf{U}$. Como S_1 no depende de $\mathbf{U}$

tenemos que $\frac{\partial}{\partial \mathbf{U}} S_1 = 0$. Por otra parte, como asumimos que $\mathbf{U}$ es de rango p tenemos que $\mathbf{U}\mathbf{U}^\top$ es invertible y, por lo tanto, por la proposición (B.2.7),

$$\frac{\partial}{\partial \mathbf{U}} S_2 = \frac{\partial}{\partial \mathbf{U}} \left(-\frac{n}{2} \ln(\det(\mathbf{U}\mathbf{U}^\top + \mathbf{I}_p)) \right) = -n(\mathbf{U}\mathbf{U}^\top + \mathbf{I}_p)^{-1}\mathbf{U} = -n\boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\mathbf{U}.$$

Resta calcular la derivada de S_3. Tenemos que

$$\begin{aligned} \frac{\partial}{\partial \mathbf{U}} S_3 &= \frac{\partial}{\partial \mathbf{U}} \left(-\frac{n}{2} \operatorname{tr}\left((\mathbf{U}\mathbf{U}^\top + \mathbf{I}_p)^{-1} \widehat{\boldsymbol{\Sigma}}_n \right) \right) \\ &= n(\mathbf{U}\mathbf{U}^\top + \mathbf{I}_p)^{-1} \widehat{\boldsymbol{\Sigma}}_n (\mathbf{U}\mathbf{U}^\top + \mathbf{I}_p)^{-1}\mathbf{U} = n\boldsymbol{\Sigma}_{\mathbf{x}}^{-1} \widehat{\boldsymbol{\Sigma}}_n \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\mathbf{U}. \end{aligned}$$

Por lo tanto,

$$\frac{\partial}{\partial \mathbf{U}} \ell(\bar{\mathbf{x}}_n, \boldsymbol{\Sigma}_{\mathbf{x}}) = -n\boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\mathbf{U} + n\boldsymbol{\Sigma}_{\mathbf{x}}^{-1} \widehat{\boldsymbol{\Sigma}}_n \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\mathbf{U} = \mathbf{0}.$$

y, esto implica que

$$\mathbf{U} = \widehat{\boldsymbol{\Sigma}}_n \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\mathbf{U}. \tag{E.0.2}$$

La ecuación (E.0.2) tiene tres posibles soluciones.

Solución 1: $\mathbf{U} = \mathbf{0}$. Una posible solución es que $\mathbf{U}$ sea la matriz nula. Pero esto viola el hecho de que estamos buscando que $\mathbf{U}$ sea una matriz de rango completo. Por lo tanto, esta solución no será considerada.

Solución 2: $\widehat{\boldsymbol{\Sigma}}_n \boldsymbol{\Sigma}_{\mathbf{x}}^{-1} = \mathbf{I}$. Esta solución implica que $\boldsymbol{\Sigma}_{\mathbf{x}} = \widehat{\boldsymbol{\Sigma}}_n$. Para hallar $\mathbf{U}$, notemos que como $\boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{U}\mathbf{U}^\top + \sigma^2\mathbf{I}_p$, a través de la proposición (A.1.6) podemos afirmar que los valores propios de $\boldsymbol{\Sigma}_{\mathbf{x}}$ son aquellos de $\mathbf{U}\mathbf{U}^\top$ más σ^2. Por otra parte, como $\mathbf{U}$ es de rango p y $\mathbf{U}\mathbf{U}^\top$ es semidefinida positiva, $\mathbf{U}\mathbf{U}^\top$ tiene $d - p$ valores propios iguales a 0. Luego, utilizando estas ideas y el teorema espectral para matrices simétricas, podemos escribir

$$\widehat{\boldsymbol{\Sigma}}_n = \begin{pmatrix} \mathbf{U}_p & \mathbf{U}_{d-p} \end{pmatrix} \begin{pmatrix} \Lambda & \mathbf{0} \\ \mathbf{0} & \sigma^2\mathbf{I}_{d-p} \end{pmatrix} \begin{pmatrix} \mathbf{U}_p & \mathbf{U}_{d-p} \end{pmatrix}^\top,$$

donde $\Lambda = \operatorname{diag}(\{\sigma_1, \ldots, \sigma_p\})$ es una matriz diagonal con los p valores más grandes de $\widehat{\boldsymbol{\Sigma}}_n$ y las columnas $\mathbf{U}_p$ son los vectores propios asociados. Luego

$$\mathbf{U}\mathbf{U}^\top = \widehat{\boldsymbol{\Sigma}}_n - \sigma^2\mathbf{I}_p = \begin{pmatrix} \mathbf{U}_p & \mathbf{U}_{d-p} \end{pmatrix} \begin{pmatrix} \Lambda & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}_p & \mathbf{U}_{d-p} \end{pmatrix}^\top = \mathbf{U}_p(\Lambda - \sigma^2\mathbf{I})\mathbf{U}_p^\top.$$

Como $\mathbf{U}$ y $\mathbf{U}_p$ son de rango p debe darse que todas las soluciones de $\mathbf{U}$ sean de la forma $\mathbf{U} = \mathbf{U}_p(\Lambda - \sigma^2\mathbf{I})^{1/2}\mathbf{R}$ donde $\mathbf{R}$ es una matriz ortogonal arbitraria.

Observemos también que probamos que si $\lambda_{d-p}, \ldots, \lambda_d$ son los $d - p$ valores propios más chicos de $\widehat{\Sigma}_n$, tenemos que $\sigma^2 = \lambda_{d-p} = \cdots = \lambda_d$ lo que prueba la ecuación (4.1.1).

Solución 3: $\mathbf{U} \neq \mathbf{0}$ y $\boldsymbol{\Sigma}_{\mathbf{x}} \neq \widehat{\boldsymbol{\Sigma}}_n$. Consideremos $\mathbf{U} = \mathbf{P}\Gamma\mathbf{V}^\top$ la SVD compacta de $\mathbf{U}$, donde $\mathbf{P} \in \mathbb{R}^{d\times p}$ es una matriz con columnas ortonormales, $\Gamma \in \mathbb{R}^{p\times p}$ es una matriz diagonal y $\mathbf{V} \in \mathbb{R}^{p\times p}$ es ortogonal. Sea $\mathbf{P}^\perp \in \mathbb{R}^{d\times(d-p)}$ una matriz con columnas ortonormales tal que $\mathbf{P}^\top\mathbf{P}^\perp = \mathbf{0}$. Con esta elección de $\mathbf{P}$ tenemos que la matriz $\begin{pmatrix} \mathbf{P} & \mathbf{P}^\perp \end{pmatrix}$ es ortonormal y cumple que $\mathbf{P}\mathbf{P}^\top + \mathbf{P}^\perp(\mathbf{P}^\perp)^\top = \mathbf{I}_d$. Luego

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathbf{x}} = \mathbf{U}\mathbf{U}^\top + \sigma^2\mathbf{I}_d &= \mathbf{P}\Gamma^2\mathbf{P}^\top + \sigma^2(\mathbf{P}\mathbf{P}^\top + \mathbf{P}^\perp(\mathbf{P}^\perp)^\top) \\ &= \mathbf{P}(\Gamma^2 + \sigma^2\mathbf{I}_d)\mathbf{P}^\top + \sigma^2\mathbf{P}^\perp(\mathbf{P}^\perp)^\top. \end{aligned} \tag{E.0.3}$$

Luego, combinando la ecuación (E.0.3) con (E.0.2) tenemos que

$$\begin{aligned}\widehat{\boldsymbol{\Sigma}}_n\boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\mathbf{U} &= \widehat{\boldsymbol{\Sigma}}_n(\mathbf{P}(\Gamma^2+\sigma^2\mathbf{I}_d)\mathbf{P}^\top+\sigma^2\mathbf{P}^\perp(\mathbf{P}^\perp)^\top)^{-1}\mathbf{P}\Gamma\mathbf{V}^\top\\ &= \widehat{\boldsymbol{\Sigma}}_n\mathbf{P}(\Gamma^2+\sigma^2\mathbf{I}_d)^{-1}\mathbf{P}^\top\mathbf{P}\Gamma\mathbf{V}^\top+\widehat{\boldsymbol{\Sigma}}_n(\sigma^{-2}\mathbf{P}^\perp(\mathbf{P}^\perp)^\top)\mathbf{P}\Gamma\mathbf{V}^\top\\ &= \widehat{\boldsymbol{\Sigma}}_n\mathbf{P}(\Gamma^2+\sigma^2\mathbf{I}_d)^{-1}\mathbf{P}^\top\mathbf{P}\Gamma\mathbf{V}^\top = \widehat{\boldsymbol{\Sigma}}_n\mathbf{P}(\Gamma^2+\sigma^2\mathbf{I}_d)^{-1}\Gamma\mathbf{V}^\top\\ &= \mathbf{P}\Gamma\mathbf{V}^\top\end{aligned}$$

si y sólo si

$$\widehat{\boldsymbol{\Sigma}}_n\mathbf{P} = \mathbf{P}(\Gamma^2+\sigma^2\mathbf{I}_d). \tag{E.0.4}$$

Escribiendo $\mathbf{P} = (\mathbf{p}_1,\dots,\mathbf{p}_d)$ y $\Gamma = \operatorname{diag}(\gamma_1,\dots,\gamma_d)$, la ecuación (E.0.4) puede ser escrita como

$$\widehat{\boldsymbol{\Sigma}}_n\mathbf{p}_i = (\gamma_i^2+\sigma^2)\mathbf{p}_i, \quad i=1,\dots,d. \tag{E.0.5}$$

Por lo tanto, $\mathbf{P}$ es una matriz que contiene d vectores propios de $\widehat{\boldsymbol{\Sigma}}_n$ con valores propios asociados $\lambda_i = \gamma_i^2+\sigma^2$. Luego $\gamma_i = (\lambda_i-\sigma^2)^{1/2}$. Consideremos entonces la descomposición en valores propios de $\hat{\boldsymbol{\Sigma}}_n$,

$$\hat{\boldsymbol{\Sigma}}_n = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top = \begin{pmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{pmatrix}\operatorname{diag}\{\Lambda_1,\Lambda_2\}\begin{pmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{pmatrix}^\top,$$

donde $\mathbf{U}_1$ es una matriz con d vectores propios y Λ_1 es una matriz con sus d valores propios asociados. Luego, las soluciones óptimas de $\mathbf{U}$ son de la forma

$$\mathbf{U} = \mathbf{P}\Gamma\mathbf{V}^\top = \mathbf{U}_1(\Lambda_1-\sigma^2\mathbf{I}_d)^{1/2}\mathbf{V}^\top.$$

Para determinar σ^2 reemplacemos la solución de $\mathbf{U}$ en la función de verosimilitud presentada en (E.0.1). El resultado de esto es, por un lado, que

$$\begin{aligned}\det(\boldsymbol{\Sigma}_{\mathbf{x}}) &= \det(\mathbf{U}\mathbf{U}^\top+\sigma^2\mathbf{I}_d)\\ &= \det(\mathbf{U}_1(\Lambda_1-\sigma^2\mathbf{I}_d)^{1/2}\mathbf{V}^\top(\mathbf{U}_1(\Lambda_1-\sigma^2\mathbf{I}_d)^{1/2}\mathbf{V}^\top)^\top+\sigma^2\mathbf{I}d)\\ &= \det\left(\mathbf{U}_1(\Lambda_1-\sigma^2\mathbf{I}_d)\mathbf{U}_1+\sigma^2(\mathbf{U}_1\mathbf{U}_1^\top+\mathbf{U}_2\mathbf{U}_2^\top)\right)\\ &= \det\left(\mathbf{U}_1\Lambda_1\mathbf{U}_1^\top+\sigma^2\mathbf{U}_2\mathbf{U}_2^\top\right) = \det(\Lambda_1)\sigma^{2(d-p)}.\end{aligned}$$

y, por otro lado,

$$\begin{aligned}\operatorname{tr}(\boldsymbol{\Sigma}_{\mathbf{x}}^{-1}\widehat{\boldsymbol{\Sigma}}_n) &= \operatorname{tr}\left((\mathbf{U}_1\Lambda_1^{-1}\mathbf{U}_1^\top+\sigma^{-2}\mathbf{U}_2\mathbf{U}_2^\top)(\mathbf{U}_1\Lambda_1\mathbf{U}_1^\top+\mathbf{U}_2\Lambda_2\mathbf{U}_2^\top\right)\\ &= \operatorname{tr}\left(\mathbf{U}_1\mathbf{U}_1^\top+\sigma^{-2}\mathbf{U}_2\Lambda\mathbf{U}_2\right) = d+\sigma^{-2}\operatorname{tr}(\Lambda_2),\end{aligned}$$

lo que implica que

$$\ell(\bar{\mathbf{x}}_n,\mathbf{U},\sigma^2) = -\frac{n}{2}\left(\ln(2\pi)+\ln(\det(\Lambda_1))+(d-p)\ln(\sigma^2)+p+\sigma^{-2}\operatorname{tr}(\Lambda_2)\right). \tag{E.0.6}$$

Derivando la ecuación (E.0.6) obtenemos que

$$\frac{\partial}{\partial\sigma^2}\ell(\bar{\mathbf{x}}_n,\mathbf{U},\sigma^2) = -\frac{n}{2}\left(\frac{d-p}{\sigma^2}-\frac{\operatorname{tr}(\Lambda_2)}{\sigma^4}\right) = 0$$

si y sólo si $\sigma^2 = \frac{\operatorname{tr}(\Lambda_2)}{d-p}$. Por lo tanto σ^2 es el promedio de los valores propios de $\hat{\boldsymbol{\Sigma}}_n$. Resta probar cuáles de los d valores propios de $\hat{\boldsymbol{\Sigma}}_n$ debemos retener y cuáles descartar. Para esto

notemos en primer lugar que $\det(\Lambda_1) = \frac{\det(\Lambda)}{\det(\Lambda_2)}$. Luego, sustituyendo por el σ^2 hallado en la ecuación (E.0.6) observamos que maximizar

$$\ell(\bar{\mathbf{x}}_n, \mathbf{U}, \sigma^2) = -\frac{n}{2}\left(\ln(2\pi) + \ln(\frac{\det(\Lambda)}{\det(\Lambda_2)}) + (d-p)\ln\left(\frac{\mathrm{tr}(\Lambda_2)}{d-p}\right) + d\right)$$

es equivalente a minimizar

$$\mathcal{M}(\pi) = \ln\left(\frac{1}{d-p}\sum_{i=p+1}^{d}\lambda_{\pi[i]}\right) - \frac{1}{d-p}\sum_{i=p+1}^{d}\ln\lambda_{\pi[i]},$$

con respecto a una permutación π de los valores propios de $\widehat{\boldsymbol{\Sigma}}_n$ de tal forma que los valores propios retenidos son $\lambda_{\pi[1]}\ldots,\lambda_{\pi[p]}$ y $\lambda_{\pi[p+1]}\ldots,\lambda_{\pi[d]}$ son aquellos descartados. Como la función $x \mapsto \ln(x)$ es cóncava tenemos, por la proposición (D.1.3) que para toda permutación π,

$$\ln\left(\frac{1}{d-p}\sum_{i=p+1}^{d}\lambda_{\pi[i]}\right) \geq \frac{1}{d-p}\sum_{i=p+1}^{d}\ln\lambda_{\pi[i]}$$

y, por lo tanto, $\mathcal{M}(\pi) \geq 0$. Además, podemos ver que $\mathcal{M}(\pi)$ se minimiza cuando los valores propios descartados se eligen de forma contigua en el espectro de $\widehat{\boldsymbol{\Sigma}}_n$. Afirmamos que los valores descartados son los $d-p$ valores más pequeños de $\widehat{\boldsymbol{\Sigma}}_n$.

Supongamos que esto no es así, es decir, supongamos que uno de los $d-p$ valores propios más chicos de $\widehat{\boldsymbol{\Sigma}}_n$ es elegido. Entonces $\lambda_{\min}$ debe ser también uno de los valores elegidos y tendríamos que $\lambda_{\text{mín}} < \sigma^2$. Pero esto es una contradicción con la ecuación (E.0.5). La contradicción surge de suponer que uno de los $d-p$ valores propios más chicos fue elegido y no descartado.

Concluimos entonces que en este caso, los estimadores son los de la ecuación (4.1.5), como queríamos probar. ■

Apéndice F

Gráficos suplementarios

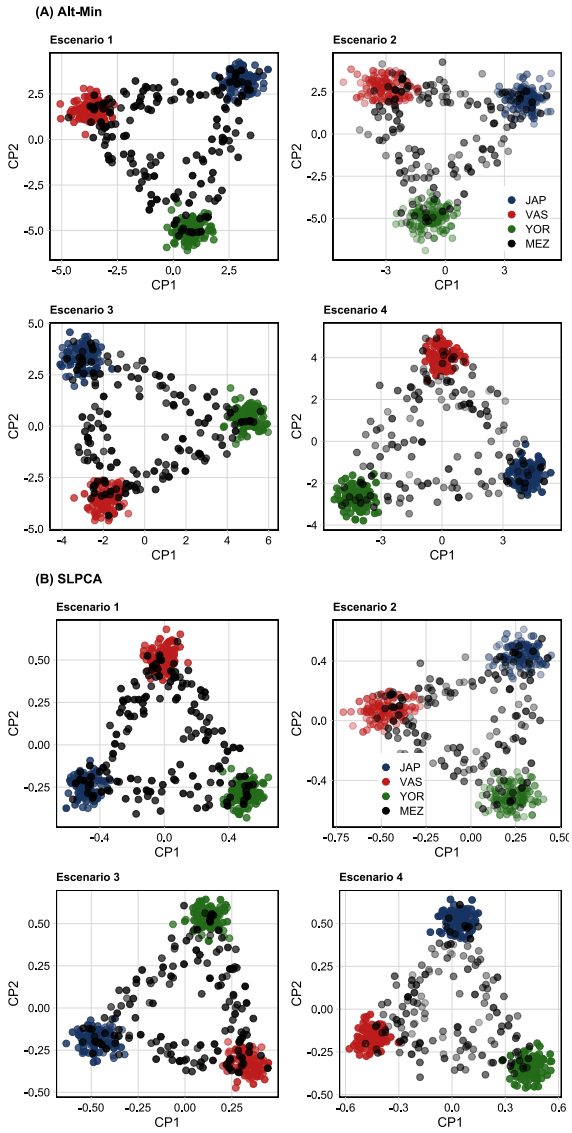

Figura F.1: Reconstrucción sobre las dos primeras componentes de los experimentos de la sección (6.2). (A) Resultados para el algoritmo Alt-Min. Se consideró el criterio de parada descrito en la sección (3.1) para $\varepsilon = 10^{-5}$. (B) Resultados para el algoritmo SLPCA. Se consideró el criterio de parada descrito en la sección (3.2) con $\varepsilon = 10^{-5}$.

Bibliografía

[1] D. L. Hartl and A. G. Clark, *Principles of Population Genetics*. Sinauer Associates, Inc, 1997.

[2] J. Relethford, "Global Patterns of Isolation by Distance Based on Genetic and Morphological Data," *Human Biology*, vol. 76, 2004.

[3] J. K. Pritchard, M. Stephens, and P. Donnelly, "Inference of Population Structure Using Multilocus Genotype Data," *Genetics*, vol. 155, no. 2, pp. 945–959, 2000.

[4] P. Menozzi, A. Piazza, and L. Cavalli-Sforza, "Synthetic Maps of Human Gene Frequencies in Europeans," *Science*, vol. 201, no. 4358, pp. 786–792, 1978.

[5] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[6] A. L. Price, N. J. Patterson, R. M. Plenge, M. E. Weinblatt, N. A. Shadick, and D. Reich, "Principal Components Analysis Corrects for Stratification in Genome-Wide Association Studies," *Nature Genetics*, vol. 38, no. 8, pp. 904–909, 2006.

[7] N. Patterson, A. L. Price, and D. Reich, "Population Structure and Eigenanalysis," *PLOS Genetics*, vol. 2, pp. 1–20, 12 2006.

[8] J. Novembre and M. Stephens, "Interpreting Principal Component Analyses of Spatial Population Genetic Variation.," *Nature Genetics*, vol. 40, pp. 646–9, May 2008.

[9] R. Hui, E. D'Atanasio, L. M. Cassidy, C. L. Scheib, and T. Kivisild, "Evaluating genotype imputation pipeline for ultra-low coverage ancient genomes," *Scientific Reports*, vol. 10, no. 1, p. 18542, 2020.

[10] A. Moreno-Estrada, S. Gravel, F. Zakharia, J. L. McCauley, J. K. Byrnes, C. R. Gignoux, P. A. Ortiz-Tello, R. J. Martínez, D. J. Hedges, R. W. Morris, C. Eng, K. Sandoval, S. Acevedo-Acevedo, P. J. Norman, Z. Layrisse, P. Parham, J. C. Martínez-Cruzado, E. G. Burchard, M. L. Cuccaro, E. R. Martin, and C. D. Bustamante, "Reconstructing the Population Genetic History of the Caribbean," *PLOS Genetics*, vol. 9, pp. 1–19, 11 2013.

[11] L. H. Hartwell, M. L. Goldberg, J. A. Fischer, and L. Hood, *Genetics: From Genes to Genomes*. McGraw-Hill Education, sixth ed., 2018.

[12] P. Luisi, A. García, J. M. Berros, J. M. B. Motti, D. A. Demarchi, E. Alfaro, E. Aquilano, C. Argüelles, S. Avena, G. Bailliet, J. Beltramo, C. M. Bravi, M. Cuello, C. Dejean, J. E. Dipierri, L. S. J. Medina, J. L. Lanata, M. Muzzio, M. L. Parolin, M. Pauro, P. B. P. Sepúlveda, D. R. Golpe, M. R. Santos, M. Schwab, N. Silvero, J. Zubrzycki, V. Ramallo, and H. Dopazo, "Fine-scale Genomic Analyses of Admixed Individuals Reveal Unrecognized Genetic Ancestry Components in Argentina," *PLOS ONE*, vol. 15, no. 7, p. e0233808, 2020.

[13] J. Novembre, T. Johnson, K. Bryc, Z. Kutalik, A. R. Boyko, A. Auton, A. Indap, K. S. King, S. Bergmann, M. R. Nelson, M. Stephens, and C. D. Bustamante, "Genes Mirror Geography Within Europe," *Nature*, vol. 456, no. 7218, pp. 98–101, 2008.

[14] C. Sabatti, S. K. Service, A.-L. Hartikainen, A. Pouta, S. Ripatti, J. Brodsky, C. G. Jones, N. A. Zaitlen, T. Varilo, M. Kaakinen, U. Sovio, A. Ruokonen, J. Laitinen, E. Jakkula, L. Coin, C. Hoggart, A. Collins, H. Turunen, S. Gabriel, P. Elliot, M. I. McCarthy, M. J. Daly, M.-R. Järvelin, N. B. Freimer, and L. Peltonen, "Genome-wide Association Analysis of Metabolic Traits in a Birth Cohort from a Founder Population," *Nature Genetics*, vol. 41, no. 1, pp. 35–46, 2009.

[15] S. Dray and J. Josse, "Principal component analysis with missing values: a comparative survey of methods," *Plant Ecology*, vol. 216, no. 5, pp. 657–667, 2015.

[16] R. Hartley and F. Schaffalitzky, "PowerFactorization : 3D Reconstruction with Missing or Uncertain Data," 01 2003.

[17] R. Vidal, Y. Ma, and S. S. Sastry, *Generalized Principal Component Analysis*. Springer, 2016.

[18] T. Raiko, A. Ilin, and J. Karhunen, "Principal Component Analysis for Large Scale Problems with Lots of Missing Values," in *Machine Learning: ECML 2007* (J. N. Kok, J. Koronacki, R. L. d. Mantaras, S. Matwin, D. Mladenič, and A. Skowron, eds.), (Berlin, Heidelberg), pp. 691–698, Springer Berlin Heidelberg, 2007.

[19] A. M. Buchanan and A. W. Fitzgibbon, "Damped Newton Algorithms for Matrix Factorization with Missing Data," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, pp. 316–322 vol. 2, 2005.

[20] M. L. Eaton, *Multivariate Statistics: A Vector Space Approach*. Institute of Mathematical Statistics, 2007.

[21] R. M. Larsen, *Efficient Algorithms for Helioseismic Inversion*. PhD thesis, Århus University, 1998.

[22] G. H. Golub and C. F. van Loan, *Matrix Computations*. The Johns Hopkins University Press, 2013.

[23] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, pp. 1956–1982, 2010.

[24] E. J. Candès and B. Recht, "Exact Matrix Completion via Convex Optimization," *Foundations of Computational Mathematics*, vol. 9, 2009.

[25] M. Fazel, *Matrix Rank Minimization with Applications*. PhD thesis, Stanford University, 2002.

[26] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[27] G. Watson, "Characterization of the Subdifferential of Some Matrix Norms," *Linear Algebra and its Applications*, vol. 170, pp. 33–45, 1992.

[28] M. Jakobsson, S. W. Scholz, P. Scheet, J. R. Gibbs, J. M. VanLiere, H.-C. Fung, Z. A. Szpiech, J. H. Degnan, K. Wang, R. Guerreiro, J. M. Bras, J. C. Schymick, D. G. Hernandez, B. J. Traynor, J. Simon-Sanchez, M. Matarin, A. Britton, J. van de Leemput, I. Rafferty, M. Bucan, H. M. Cann, J. A. Hardy, N. A. Rosenberg, and A. B. Singleton, "Genotype, Haplotype and Copy-number Variation in Worldwide Human Populations," *Nature*, vol. 451, no. 7181, pp. 998–1003, 2008.

[29] J. Meisner, S. Liu, M. Huang, and A. Albrechtsen, "Large-scale Inference of Population Structure in Presence of Missingness Using PCA," *Bioinformatics*, 01 2021. btab027.

[30] A. Auton, G. R. Abecasis, D. M. Altshuler, R. M. Durbin, G. R. Abecasis, D. R. Bentley, A. Chakravarti, A. G. Clark, P. Donnelly, E. E. Eichler, P. Flicek, S. B. Gabriel, R. A. Gibbs, E. D. Green, M. E. Hurles, B. M. Knoppers, J. O. Korbel, E. S. Lander, C. Lee, H. Lehrach, E. R. Mardis, G. T. Marth, G. A. McVean, D. A. Nickerson, J. P. Schmidt, S. T. Sherry, J. Wang, R. K. Wilson, R. A. Gibbs, E. Boerwinkle, H. Doddapaneni, Y. Han, V. Korchina, C. Kovar, S. Lee, D. Muzny, J. G. Reid, Y. Zhu, J. Wang, Y. Chang, Q. Feng, X. Fang, X. Guo, M. Jian, H. Jiang, X. Jin, T. Lan, G. Li, J. Li, Y. Li, S. Liu, X. Liu, Y. Lu, X. Ma, M. Tang, B. Wang, G. Wang, H. Wu, R. Wu, X. Xu, Y. Yin, D. Zhang, W. Zhang, J. Zhao, M. Zhao, X. Zheng, E. S. Lander, D. M. Altshuler, S. B. Gabriel, N. Gupta, N. Gharani, L. H. Toji, N. P. Gerry, A. M. Resch, P. Flicek, J. Barker, L. Clarke, L. Gil, S. E. Hunt, G. Kelman, E. Kulesha, R. Leinonen, W. M. McLaren, R. Radhakrishnan, A. Roa, D. Smirnov, R. E. Smith, I. Streeter, A. Thormann, I. Toneva, B. Vaughan, X. Zheng-Bradley, D. R. Bentley, R. Grocock, S. Humphray, T. James, Z. Kingsbury, H. Lehrach, R. Sudbrak, M. W. Albrecht, V. S. Amstislavskiy, T. A. Borodina, M. Lienhard, F. Mertes, M. Sultan, B. Timmermann, M.-L. Yaspo, E. R. Mardis, R. K. Wilson, L. Fulton, R. Fulton, S. T. Sherry, V. Ananiev, Z. Belaia, D. Beloslyudtsev, N. Bouk, C. Chen, D. Church, R. Cohen, C. Cook, J. Garner, T. Hefferon, M. Kimelman, C. Liu, J. Lopez, P. Meric, C. O'Sullivan, Y. Ostapchuk, L. Phan, S. Ponomarov, V. Schneider, E. Shekhtman, K. Sirotkin, D. Slotta, H. Zhang, G. A. McVean, R. M. Durbin, S. Balasubramaniam, J. Burton, P. Danecek, T. M. Keane, A. Kolb-Kokocinski, S. McCarthy, J. Stalker, M. Quail, J. P. Schmidt, C. J. Davies, J. Gollub, T. Webster, B. Wong, Y. Zhan, A. Auton, C. L. Campbell, Y. Kong, A. Marcketta, R. A. Gibbs, F. Yu, L. Antunes, M. Bainbridge, D. Muzny, A. Sabo, Z. Huang, J. Wang, L. J. M. Coin, L. Fang, X. Guo, X. Jin, G. Li, Q. Li, Y. Li, Z. Li, H. Lin, B. Liu, R. Luo, H. Shao, Y. Xie, C. Ye,

C. Yu, F. Zhang, H. Zheng, H. Zhu, C. Alkan, E. Dal, F. Kahveci, G. T. Marth, E. P. Garrison, D. Kural, W.-P. Lee, W. Fung Leong, M. Stromberg, A. N. Ward, J. Wu, M. Zhang, M. J. Daly, M. A. DePristo, R. E. Handsaker, D. M. Altshuler, E. Banks, G. Bhatia, G. del Angel, S. B. Gabriel, G. Genovese, N. Gupta, H. Li, S. Kashin, E. S. Lander, S. A. McCarroll, J. C. Nemesh, R. E. Poplin, S. C. Yoon, J. Lihm, V. Makarov, A. G. Clark, S. Gottipati, A. Keinan, J. L. Rodriguez-Flores, J. O. Korbel, T. Rausch, M. H. Fritz, A. M. Stütz, P. Flicek, K. Beal, L. Clarke, A. Datta, J. Herrero, W. M. McLaren, G. R. S. Ritchie, R. E. Smith, D. Zerbino, X. Zheng-Bradley, P. C. Sabeti, I. Shlyakhter, S. F. Schaffner, J. Vitti, D. N. Cooper, E. V. Ball, P. D. Stenson, D. R. Bentley, B. Barnes, M. Bauer, R. Keira Cheetham, A. Cox, M. Eberle, S. Humphray, S. Kahn, L. Murray, J. Peden, R. Shaw, E. E. Kenny, M. A. Batzer, M. K. Konkel, J. A. Walker, D. G. MacArthur, M. Lek, R. Sudbrak, V. S. Amstislavskiy, R. Herwig, E. R. Mardis, L. Ding, D. C. Koboldt, D. Larson, K. Ye, S. Gravel, T. . G. P. Consortium, C. authors, S. committee, P. group, B. C. of Medicine, BGI-Shenzhen, M. I. T. of Broad Institute, Harvard, C. I. for Medical Research, E. B. I. European Molecular Biology Laboratory, Illumina, M. P. I. for Molecular Genetics, M. G. I. at Washington University, U. S. N. I. of Health, U. of Oxford, W. T. S. Institute, A. group, Affymetrix, A. E. C. of Medicine, B. University, B. College, C. S. H. Laboratory, C. University, E. M. B. Laboratory, H. University, H. G. M. Database, I. S. of Medicine at Mount Sinai, L. S. University, M. G. Hospital, M. University, and N. I. H. National Eye Institute, "A Global Reference for Human Genetic Variation," *Nature*, vol. 526, no. 7571, pp. 68–74, 2015.

[31] L. Spangenberg, M. I. Fariello, D. Arce, G. Illanes, G. Greif, J.-Y. Shin, S.-K. Yoo, J.-S. Seo, C. Robello, C. Kim, J. Novembre, M. Sans, and H. Naya, "Indigenous ancestry and admixture in the Uruguayan population," *bioRxiv*, 2021.

[32] D. A. Harville, *Matrix Algebra From a Statistician's Perspective*. Springer, 1997.

[33] R. Penrose, "A Generalized Inverse for Matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, no. 3, p. 406–413, 1955.

[34] A. Beck, *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, 2017.

[35] M. E. Tipping and C. M. Bishop, "Probabilistic Principal Component Analysis," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 611–622, 1999.

[36] P. J. Dhrymes, *Mathematics for Econometrics*. Springer, 2013.

[37] C. Chiang, J. Marcus, C. Sidore, A. Biddanda, H. Al-Asadi, M. Zoledziewska, M. Pitzalis, F. Busonero, A. Maschio, G. Pistis, M. Steri, A. Angius, K. Lohmueller, G. Abecasis, D. Schlessinger, F. Cucca, and J. Novembre, "Genomic History of the Sardinian Population," *Nature Genetics*, vol. 50, Oct. 2018.

[38] J. R. Homburger, A. Moreno-Estrada, C. R. Gignoux, D. Nelson, E. Sanchez, P. Ortiz-Tello, B. A. Pons-Estel, E. Acevedo-Vasquez, P. Miranda, C. D. Langefeld, S. Gravel,

M. E. Alarcón-Riquelme, and C. D. Bustamante, "Genomic Insights into the Ancestry and Demographic History of South America," *PLOS Genetics*, vol. 11, pp. 1–26, 12 2015.

[39] K. Ausmees, "Evaluation of methods handling missing data in PCA on genotype data: Applications for ancient DNA.," tech. rep., Universidad de Uppsåla, 2019.