Centro de Investigación en Matemáticas, A.C.

Proyecciones aleatorias para aproximar métodos kernel

T E S I S

Que para obtener el grado académico de:

Maestra en Ciencias con especialidad en Probabilidad y Estadística

Presenta

Flor de María Martinez Sermeño

Dr. Johan Jozef Lode Van Horebeek

Director de Tesis

29 de Enero de 2015

Centro de Investigación en Matemáticas, A.C.

Acta de Examen de Grado

CIMAT
CENTRO DE INVESTIGACION EN MATEMATICAS A. C.

Acta No.: 086

Libro No.: 002

Foja No.: 086

En la Ciudad de Guanajuato, Gto., siendo las 12:30 horas del día 29 de enero del año 2015, se reunieron los miembros del jurado integrado por los señores:

DR. MIGUEL NAKAMURA SAVOY (CIMAT)
DR. ROGELIO RAMOS QUIROGA (CIMAT)
DR. JOHAN JOZEF LODE VAN HOREBEEK (CIMAT)

bajo la presidencia del primero y con carácter de secretario el segundo, para proceder a efectuar el examen que para obtener el grado de

MAESTRO EN CIENCIAS
CON ESPECIALIDAD EN PROBABILIDAD Y ESTADÍSTICA

sustenta

FLOR DE MARÍA MARTÍNEZ SERMEÑO

en cumplimiento con lo establecido en los reglamentos y lineamientos de estudios de posgrado del Centro de Investigación en Matemáticas, A.C., mediante la presentación de la tesis

"PROYECCIONES ALEATORIAS PARA APROXIMAR MÉTODOS KERNEL "

Los miembros del jurado examinaron alternadamente al (la) sustentante y después de deliberar entre sí resolvieron declararlo (a):

Aprobada

DR. MIGUEL NAKAMURA SAVOY
Presidente

DR. ROGELIO RAMOS QUIROGA
Secretario

DR. JOHAN JOZEF LODE VAN HOREBEEK
Vocal

CIMAT
DIRECCIÓN GENERAL

DR. JOSÉ ANTONIO STEPHAN DE LA PEÑA MENA
Director General

A mi fortaleza e inspiración:

Flor Sermeño y Gerardo.

Agradecimientos

Agradezco a mi amado país El Salvador por la identidad y fortaleza que me ha brindado. A Guanajuato, mi segundo hogar, por recibirme con los brazos abiertos y haber sido la cuna de muchas alegrías en mi vida.

Al Dr. Johan Van Horebeek por haber guiado esta tesis en direcciones interesantes, por su paciencia, su atención a los "detallitos", su apoyo, sus consejos y por encaminarme en el arte de la investigación.

A mis sinodales, el Dr. Miguel Nakamura Savoy y el Dr. Rogelio Ramos Quiroga por todo el apoyo, conocimiento y consejos que me han brindado durante mi formación académica, y por el tiempo y esfuerzo invertido en revisar este trabajo y enriquecerlo con sus comentarios.

A Gerardo, Roxie y Lennon por ser mi felicidad e inspiración y por siempre animarme con palabras, ladridos, abrazos y besos.

A mi madre por ser mi apoyo incondicional, por impulsarme a perseguir mis sueños y no abandonarlos y por estar siempre a mi lado sin importar la distancia. A mis hermanos y a Noyo Toño por su apoyo y por todos los momentos a su lado que llenan mi corazón de felicidad. A mi padre por haberme ayudado a recibir una buena educación.

A toda mi familia, Sermeño y Ortega por toda la alegría, amor y palabras de aliento que me han brindado siempre.

A mis amigos y seres queridos (humanos y no humanos) por todos los buenos momentos compartidos, por siempre tenderme la mano y por todo su cariño, que de alguna u otra manera ayudaron a que este trabajo haya sido posible.

Al Centro de Investigación en Matemáticas (CIMAT) y al Consejo Nacional de Ciencia y Tecnología (CONACYT) por los apoyos económicos recibidos para mi formación como profesional.

Resumen

Los métodos kernel o métodos de transformaciones implícitas han sido de gran utilidad para el reconocimiento de patrones en diversos tipos de datos. Dichos métodos son convenientes en dos situaciones. La primera es cuando se tienen datos sin una representación vectorial natural. Por ejemplo, cuando las observaciones son cadenas de carácteres. La segunda, la cual es de particular interés en esta tesis, es cuando se requiere mapear los datos a un (otro) espacio de Hilbert en el cual se puedan analizar más fácilmente mediante técnicas estadísticas clásicas como: regresión lineal, análisis de discriminante lineal (LDA), análisis de componentes principales (PCA), clustering, entre otros.

Se tiene un interés particular en funciones kernel para análisis de componentes principales. Dichas funciones serán útiles para el caso en que los datos tienen patrones no lineales y por lo tanto, PCA no es una herramienta adecuada para analizarlos. La idea es emplear las funciones kernel para mapear los datos a un espacio donde tengan estructura lineal y posteriormente realizar PCA sobre los datos transformados. El procedimiento anterior es un método kernel conocido como Kernel análisis de componentes principales (KPCA).

Los métodos kernel pueden implicar trabajo con matrices de gran dimensión, lo cual computacionalmente puede ser un problema. Por lo anterior, en esta tesis se presentan diversos métodos aleatorizados para aproximación de matrices, los cuales han recibido gran atención en el área de álgebra lineal. Dichos métodos utilizan un elemento aleatorio para aproximar matrices mediante otras cuya estructura es más sencilla (menor dimensión, rango menor, etc.). En la presente tesis se puede encontrar una descripción y reseña de los siguientes métodos aleatorizados para aproximación de matrices: el de columnas, el de Nyström y el Random Fourier Features (RFF). Los primeros dos métodos son bastante similares en su estructuración ya que la idea básica de estos es utilizar un subconjunto o una muestra de los datos para obtener una aproximación. El método Random Fourier Features es diferente a estos en cuanto a su formulación, ya que está basado en aproximaciones probabilísticas de las funciones kernel. Como se explica en esta tesis, la formulación del método RFF es independiente de la distribución los datos y se intuye que dicha característica hace que el método sea menos eficiente para algunas tareas. Por dicha razón, se presenta una modificación realizada al algoritmo RFF, de tal manera que pueda aprovechar la distribución de los datos.

Como puede inferirse, tanto los métodos kernel como los aleatorizados son muy útiles en áreas como big data, minería de datos, bioinformática, entre otras. Debido a ello, esta tesis se enfoca en utilizar los métodos aleatorizados para aproximar una matriz obtenida mediante una transformación kernel de un conjunto grande de datos que tiene patrones no lineales. En particular, se compara el desempeño de los métodos aletorizados cuando se utilizan para aproximar el resultado obtenido por KPCA mediante resultados teóricos y a través de dos experimentos: uno con datos sintéticos y otro con datos reales. Se presentan resultados muy satisfactorios en cuanto al desempeño de los tres métodos aleatorizados (columnas, Nyström y RFF) en diversas tareas. Sin embargo, se verá que si se desea aproximar el resultado de KPCA, los métodos de columnas, RFF y una modificación de este último son más adecuados.

Índice

1. Introducción

En este capítulo se explica el problema que se desea resolver, se presentan los objetivos de la tesis y algunos resultados del área de álgebra lineal que son necesarios para los capítulos posteriores.

1.1. Motivación

El reconocimiento de patrones consiste en encontrar relaciones generales entre los elementos de un conjunto de datos. Uno de los métodos clásicos de análisis multivariado utilizados para encontrar dichas relaciones es el análisis de componentes principales (PCA). Mediante un análisis de componentes principales se encuentran patrones que explican la variabilidad de los datos, los cuales son llamados componentes principales. Se toman como componentes principales combinaciones lineales de las variables y generalmente bastan pocas componentes para explicar bien la variabilidad de los datos. Los datos se proyectan sobre algunas de las componentes principales para obtener una matriz de menor dimensión que representa bastante bien a los datos. Desafortunadamente, cuando los datos tienen una estructura no lineal, PCA no es una técnica muy adecuada.

Una idea bastante natural cuando los datos tienen una estructura no lineal, es utilizar una función para mapear los datos a un espacio en donde sí la tengan y luego analizarlos con métodos multivariados clásicos como PCA. En lugar de transformar los datos explícitamente se hace de manera implícita, lo que da raíz a métodos kernel.

Los métodos kernel son muy utilizados en ciencias de la computación, especialmente en el área de aprendizaje máquina. Estos métodos se caracterizan a partir de dos ingredientes principales: una función kernel y un método de aplicación de análisis multivariado. A pesar de que el uso de dichos métodos conlleva el uso de métodos de análisis estadístico clásicos, los métodos kernel no han contado con mucha atención en el área de estadística. Reciententemente ha surgido el interés en dichos métodos debido al tipo de datos de gran tamaño y estructura no lineal que surgen en las diferentes áreas y que requieren ser analizados estadísticamente. Sin embargo, aún no se cuenta con mucha literatura en el área de estadística para estos métodos.

Los métodos kernel realizan un análisis estadístico a partir de la matriz de Gram K, la cual se utiliza en lugar de la matriz de datos X y tiene la estructura $K = [k(x_i, x_j)]$, con k una función kernel (se explicará a detalle más adelante). Dicha matriz es de dimensión $n \times n$, para un conjunto de datos de tamaño n. Cuando n es muy grande, los métodos kernel pueden ser muy costosos computacionalmente y algunas veces imposibles de utilizar. Para solucionar este problema lo que se hace es aproximar el resultado de los métodos kernel aproximando la matriz K. Para aproximar la matriz K, se utilizarán los métodos aleatorizados de aproximación de matrices, los cuales provienen del área de álgebra lineal.

En resumen, como se muestra en la Figura 1, la tesis se centra en el problema de aproximar una matriz mediante otra con una estructura específica. Esta matriz proviene de utilizar una función kernel y es la base de un método kernel. Bajo algunas circunstancias puede ser muy costoso computacionalmente trabajar con dicha matriz, lo que implicaría grandes costos para los métodos kernel. Por dicha razón, se utilizarán métodos aleatorizados para aproximar la matriz K con una matriz más sencilla y así obtener una aproximación al resultado que se obtendría de los métodos kernel.

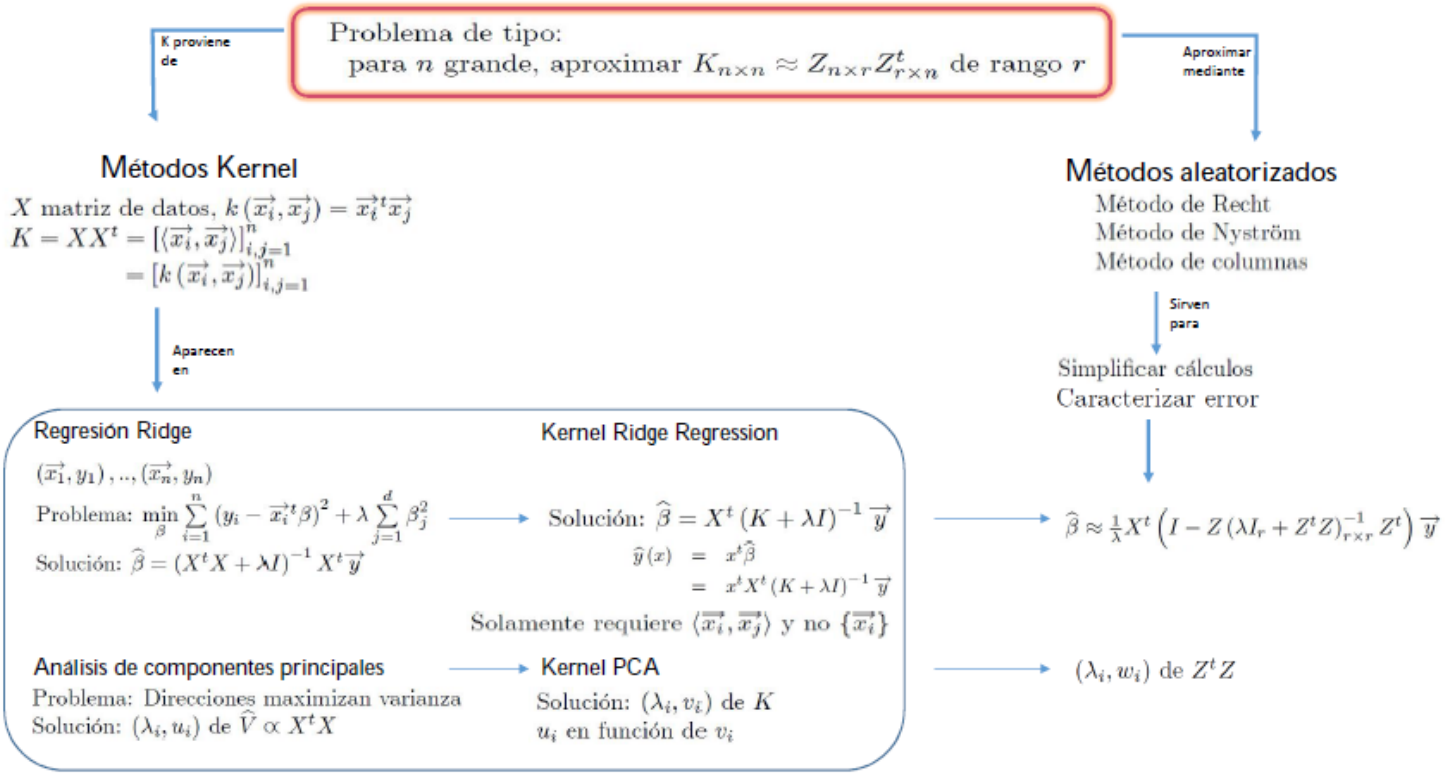

Figura 1: Diagrama del panorama general de la tesis.

Se tiene un interés particular en el método kernel PCA (KPCA), el cual consiste en realizar un análisis de componentes principales sobre la matriz K.

Como puede observarse en la Figura 1, el objetivo principal es utilizar los métodos aleatorizados para aproximar la matriz kernel de la forma $K \approx Z_{n\times r} Z^t_{n\times r}$, donde $r < n$ es el rango de Z. Como se verá más adelante, si se aproxima K con dicha estructura, el resultado de KPCA se puede aproximar haciendo PCA sobre la matriz Z, lo cual es menos costoso computacionalmente.

1.2. Objetivos

Los objetivos de esta tesis son:

- Hacer una investigación general acerca de Métodos kernel y Kernel PCA.
- Proporcionar una reseña acerca de los diferentes métodos aleatorizados para aproximación de matrices.
- Implementar los diferentes métodos aleatorizados para aproximación de matrices.

- Estudiar y comparar el desempeño de los métodos aleatorizados para aproximación de matrices cuando se utilizan para aproximar el resultado de Kernel PCA.

1.3. Preliminares de álgebra lineal

En esta sección se presentarán algunos resultados relevantes del área de álgebra lineal que de alguna manera se utilizan más adelante para aproximar matrices por otras de menor rango. Se presentarán algunas propiedades generales para matrices, la descomposición en valores singulares y resultados importantes para aproximar matrices. Además, se presentará la notación que se utilizará en esta tesis.

1.3.1. Notación

Para cualquier matriz A se utilizará la siguiente notación. $A[,j]$ denotará la columna j-ésima de A, $A[i,]$ denotará su renglón i-ésimo y $A[i,j]$ denota su entrada i,j. La traspuesta de A se denotará por A^t, la pseudo-inversa o inversa de Penroose de A por A^+ y $A^{\frac{1}{2}}$ denota la matriz C tal que $CC = A$. Las normas Euclidiana y de Frobenius de una matriz A se denotarán respectivamente por $\|A\|_2$ y $\|A\|_F$. Span(A) denota el espacio generado por los eigenvectores de A y Span$(A)^\perp$ denota el espacio ortogonal a este. El rango de la matriz A se denotará por ran(A). El vector cuyas entradas son todas cero se denotará por $\mathbf{0}$. Los espacios real, natural y complejo se denotarán por R, N y C respectivamente. La traza de A se denotará por tr(A). X denotará la matriz de datos y K, la matriz de Gram o matriz kernel .

1.3.2. Normas vectoriales y matriciales

Las normas vectoriales y matriciales sirven como una medida de qué tan cercano o lejano está un objeto de otro. Dicha medida se utilizará más adelante para medir qué tan bien se aproxima una matriz mediante otra y por lo tanto, comparar de alguna manera los métodos aletorizados entre sí. En esta subsección se recordará el concepto de producto punto, norma vectorial, algunas de sus propiedades importantes y en particular, las normas Euclidiana y de Frobenius.

Definición 1 ***Producto punto*** *(sobre R).* *Sea E un espacio vectorial sobre R. El producto punto o producto escalar es una función $\langle , \rangle : E \times E \to R$ que cumple que:*

- Para todo $v, w \in E$, $\langle v, w\rangle = \langle w, v\rangle$ (Simetría).
- Para todo v, w y $z \in E$ y $a, b \in R$ se tiene que $\langle av + bw, z\rangle = a\langle v, z\rangle + b\langle w, z\rangle$ (linealidad).
- Para todo $v \in E$, $\langle v, v\rangle \geq 0$ y $\langle v, v\rangle = 0$ si y sólo si $v = \mathbf{0}$ (Definida positiva).

Definición 2 ***Norma vectorial.*** *Sea E un espacio vectorial sobre R ó C. Una función $f : E \to R$ es una norma en E si satisface las siguiente propiedades:*

- Para todo $v \in E$, si $v \neq \mathbf{0}$ entonces $f(v) \neq 0$.

- Para todo $a \in R$ y $v \in E$, $f(av) = |a| f(v)$.
- Para todo $v, w \in E$, $f(v+w) \leq f(v) + f(w)$ (Desigualdad del triángulo).

Utilizando las propiedades anteriores, se tiene la propiedad de que $f(\mathbf{0}) = 0$.

Todo producto punto induce una norma sobre el espacio en el que está definido. En esta tesis se considerará el producto punto sobre R, ya que se utilizará la norma vectorial euclídea la cual para $v \in R^d$ está definida como:

$$\begin{aligned} \|v\| &= \sqrt{\langle v, v \rangle} \\ &= \sqrt{v_1^2 + v_2^2 + \cdots + v_d^2}. \end{aligned}$$

Utilizando el concepto de normal vectorial, se puede inducir una norma en un espacio matricial. Las propiedades de las normas matriciales son semejantes a las presentadas para la norma vectorial, excepto porque se pide que la multiplicación de matrices cumpla con algo parecido a la desigualdad triangular. Las normas matriciales Euclidiana y de Frobenius que se utilizarán más adelante, se definen a continuación. Sea A una matriz en $R^{n \times d}$, con $n, d \in N$. Sea $\{\lambda_i\}$ el conjunto de valores propios de la matriz $A^t A$.

Norma Euclidiana

La norma euclidiana de A está dada por

$$\|A\|_2 = \sqrt{\max_i \lambda_i}.$$

Es decir, la norma euclidiana de A es la raíz cuadrada del máximo valor propio de la matriz $A^t A$.

Norma de Frobenius

La norma de Frobenius de A está dada por

$$\begin{aligned} \|A\|_F &= \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{d} |A[i,j]|^2} \\ &= tr\left(A^t A\right) \\ &= \sqrt{\sum_i \lambda_i}. \end{aligned}$$

Es decir, la norma de Frobenius de A es la raíz cuadrada de la suma de los valores propios de $A^t A$.

De las definiciones anteriores, se deriva la siguiente propiedad: $\|A\|_2 \leq \|A\|_F$.

1.3.3. Propiedades de matrices de proyección

Una proyección es una transformación lineal de un espacio vectorial a otro que tiene la propiedad de ser idempotente. Las proyecciones pueden representarse utilizando matrices, a las cuales se les conoce como matrices de proyección. Una matriz P es matriz de proyección si es idempotente, es decir si $P = P^2$. Las matrices de proyección pueden entenderse como funciones que mandan un

punto en un cierto espacio a otro punto en el espacio de la imagen. Sea $P^{(A)} = A\left(A^tA\right)^{-1}A^t$ la matriz de proyección sobre Span(A) (el espacio generado por los eigenvectores de la matriz A). Para $P^{(A)}$ y cualquier vector v se tiene que:

- Los valores propios de $P^{(A)}$ son 1 ó 0. En esta tesis se considerará el caso en que al menos un valor propio debe ser 1, el caso en el que todos son cero no es de interés ya que la proyección sería al origen únicamente.
- $I-\ P^{(A)}$ es matriz de proyección.
- Si $v \in$Span(A) entonces $P^{(A)}v = v$. Esta propiedad significa que si se proyecta un vector sobre el subespacio en el que vive, entonces la proyección es el mismo vector. Es decir, la proyección no cambia los vectores que viven en el subespacio sobre el que se proyecta.
- Si $v \in$Span$(A)^{\perp}$ entonces $P^{(A)}v = \mathbf{0}$. Esta propiedad significa que para los vectores que se encuentran en el subespacio ortogonal al generado por los vectores propios de A, su proyección es el origen.
- $\left\|I - P^{(A)}\right\|_2 = 1$. Esta característica es producto de las primeras dos propiedades enunciadas en esta lista.

1.3.4. Propiedades de matrices simétricas y semidefinidas positivas

En esta tesis se presta especial atención a las matrices simétricas y semidefinidas positivas ya que, como se verá después existe una relación entre estas matrices y la matriz K que se obtiene de las transformaciones kernel. Una matriz B es simétrica si cumple que $B = B^t$ y es semidefinida positiva si $v^tBv \geq 0$ para todo $v \in R^n$ no nulo. De estas características se derivan algunas propiedades para este tipo de matrices, de las cuales las más relevantes para esta tesis se presentan a continuación:

- Si B es simétrica y positiva definida, entonces sus valores propios son reales positivos.
- Si B es simétrica entonces los vectores propios de B^tB y BB^t coinciden puesto que $B^tB = BB^t$. Como se verá en la siguiente sección, lo anterior implica que la descomposición en valores singulares de B es de la forma $B = U^BD^B\left(U^B\right)^t = V^BD^B\left(V^B\right)^t$, donde U^B y V^B denotan los vectores propios de BB^t y B^tB respectivamente.
- Para cualquier matriz simétrica y semidefinida positiva B se tiene que existe una matriz A tal que $B = AA^t$.

1.3.5. Factorizaciones o descomposiciones importantes de matrices

En esta subsección se mostrarán diferentes factorizaciones o descomposiciones de matrices que serán de gran utilidad para obtener aproximaciones de dichas matrices que tengan menor rango. Primero se mostrará la descomposición en valores singulares (SVD) la cual se ha utilizado bastante en áreas como estadística y computación. Posteriormente se mostrarán algunas descomposiciones que se obtienen utilizando SVD.

1.3.5.1. Descomposición en valores singulares (SVD) La descomposición en valores singulares de una matriz A es una descomposición o factorización de esta que ha resultado útil para diferentes tareas. Dicha factorización utiliza los vectores y valores propios de AA^t para reescribir A. Sea A una matriz de dimensión $n \times d$ de rango r. La descomposición en valores singulares de A es

$$A = U^A D^A \left(V^A\right)^t, \tag{1}$$

donde U^A de dimensión $n \times r$ es la matriz cuyas columnas son los vectores propios $\left\{u_i^A\right\}_{i=1}^r$ de la matriz AA^t y son llamados vectores singulares izquierdos de A, V^A de dimensión $d \times r$ es la matriz cuyas columnas son los vectores propios $\left\{v_i^A\right\}_{i=1}^r$ de la matriz A^tA y son llamados vectores singulares derechos de A y D^A de dimensión $r \times r$ es una matriz diagonal cuyas entradas corresponden a la raíz cuadrada de los valores propios $\{\lambda_i\}_{i=1}^r$ de AA^t ordenados de mayor a menor. Es decir, la matriz D es de la forma

$$D = \begin{pmatrix} \sqrt{\lambda_1} & 0 & 0 & 0 \\ 0 & \sqrt{\lambda_2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \sqrt{\lambda_r} \end{pmatrix}.$$

Se debe recordar que los valores propios no nulos de AA^t coinciden con los de A^tA puesto que, si λ_i es valor propio de AA^t con vector propio asociado v_i^A entonces $AA^tv_i^A = \lambda_i v_i^A$ por lo que

$$\begin{aligned} A^tAA^tv_i^A &= A^t\lambda_i v_i^A \\ &\Rightarrow \left(A^tA\right) w = \lambda_i w, \end{aligned} \tag{2}$$

donde w es el vector $A^tv_i^A$. Por lo tanto, λ_i también es valor propio de A^tA con vector propio asociado w. Además los valores propios son reales positivos ya que A^tA es simétrica y positiva definida.

Por la descomposición en valores singulares, se tiene que cualquier matriz A de rango r puede escribirse como una combinación lineal de matrices de rango 1 de la forma:

$$A = \sum_{i=1}^{r} \sqrt{\lambda_i^A} u_i^A \left(v_i^A\right)^t, \tag{3}$$

donde λ_i^A corresponde el $i-$ésimo valor singular de AA^t, u_i^A el $i-$ésimo vector singular izquierdo de A y v_i^A su $i-$ésimo vector singular derecho. Como se verá después, la expresión (3) se utiliza para aproximar la matriz A mediante otra de menor rango k tomando la suma hasta $k < r$.

1.3.5.2. Descomposiciones importantes de una matriz simétrica y semidefinida positiva

Se presentan tres caracterizaciones o descomposiciones de una matriz para el caso particular de una matriz simétrica y semidefinida positiva. Dichas descomposiciones serán útiles para aproximar matrices mediante otras de menor rango. La idea es caracterizar las matrices mediante cierta descomposición y aproximar la matriz aproximando los elementos de la descomposición.

Sea B una matriz semidefinida positiva y simétrica, las descomposiciones para esta matriz son:

Descomposición en valores singulares

Como B es simétrica, $B^tB = BB^t$ y por lo tanto, los vectores propios U^B y V^B coinciden. Por lo anterior se tiene que la descomposición en valores singulares de B es

$$B = U^B D^B \left(U^B\right)^t.$$

Descomposición en matriz proyección

Utilizando la descomposición en valores singulares de B se tiene que

$$\begin{aligned} B &= U^B D^B \left(U^B\right)^t \\ &= U^B \left[\left(U^B\right)^t U^B\right] D^B \left(U^B\right)^t \qquad \text{(pues } U^B \text{ ortonormal)} \\ &= U^B \left(U^B\right)^t B \qquad\qquad\qquad \text{(SVD de } B\text{).} \end{aligned}$$

A esta factorización de B se le llamará descomposición de B en matriz proyección. El nombre "descomposición en matriz proyección" proviene del hecho que la factorización es la proyección de los elementos de B sobre el subespacio generado por los vectores singulares derechos U^B.

Descomposición en vectores propios

Como B es semidefinida positiva se tiene que existe A tal que $B = AA^t$. Si $A = U^A D^A \left(V^A\right)^t$ es la descomposición en valores singulares de A entonces

$$\begin{aligned} AV^A &= U^A D^A \left(V^A\right)^t V^A \\ &= U^A D^A \qquad (V^A \text{ ortonormal}). \end{aligned} \tag{4}$$

Utilizando (4) se puede factorizar la matriz B de la siguiente manera:

$$\begin{aligned} B &= AA^t \\ &= U^A D^A \left(V^A\right)^t V^A D^A \left(U^A\right)^t \\ &= AV^A \left(AV^A\right)^t \qquad \text{por (4) y ortonormalidad de } V^A. \end{aligned}$$

Es decir, se caracteriza B en función de la matriz A y los vectores propios de la matriz A^tA. En el capítulo 3 se verá por qué es relevante esta caracterización a la que se le llamará descomposición de B en vectores propios.

Nótese que las tres descomposiciones coinciden, ya que la descomposición en vectores propios y en matriz proyección solamente son reescrituras de la descomposición en valores singulares.

1.3.6. Aproximación de matrices mediante otras de menor rango

En esta subsección se mostrará la forma en que se obtiene una aproximación de una matriz mediante otra menor rango utilizando las descomposición en valores singulares y se ejemplifica el efecto de dicha aproximación para una matriz que representa una imagen. También se presentan las aproximaciones para matrices simétricas y semidefinidas positivas utilizando las factorizaciones mostradas en la subsección anterior. Se utilizarán sólo las primeras k entradas de las matrices D^B y U^B. Debido a que las aproximaciones basadas en las tres descomposiciones coinciden al igual que en el caso en que $k =$ran(B), no se hará ninguna distinción de notación entre dichas aproximaciones.

Aproximación utilizando la descomposición en valores singulares

Tomando los primeros k elementos de la suma (3) se obtiene una aproximación de rango k para A. Por el teorema de Eckart-Young [22] se tiene que dicha aproximación es la mejor aproximación de rango k de la matriz A con respecto a las normas Euclidiana y de Frobenius. De manera matricial, la aproximación de rango k mediante SVD se escribe como

$$A_k = U_k^A D_k^A \left(V_k^A\right)^t,$$

donde U_k^A denota la matriz cuyas columnas están formadas por los vectores singulares izquierdos de A correspondientes a los primeros k mayores valores propios de AA^t y D_k^A es una matriz diagonal cuyas entradas corresponden a la raíz de los primeros k mayores valores propios $\left\{\sqrt{\lambda_i^A}\right\}_{i=1}^{k}$ de AA^t ordenados de mayor a menor.

Cualquier matriz puede descomponerse y aproximarse mediante otra de rango $k < r$ utilizando SVD, no obstante la forma en que cambia la matriz no es fácil de visualizar en la mayoría de los casos. Un ejemplo en el que queda bastante claro el cambio que se hace en la matriz es cuando dicha matriz representa una imagen. En la Figura 2 se presenta el cambio que produce reducir el rango en una matriz que representa una imagen. La imagen original corresponde a una matriz de dimensión 556×414 y de rango $r = 414$.

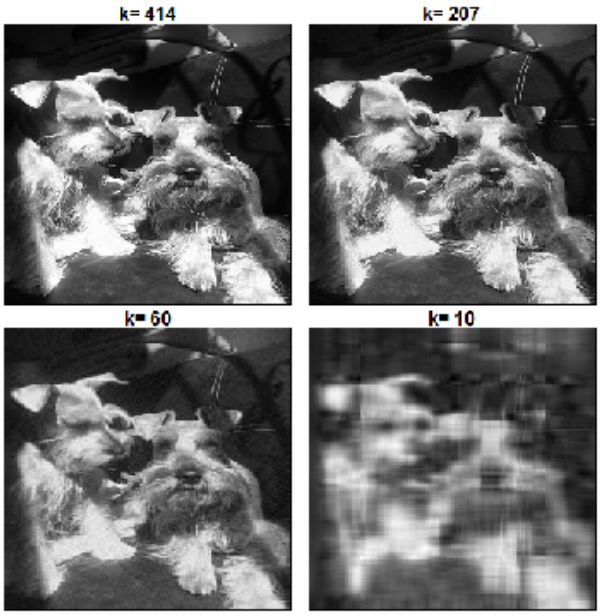

Figura 2: Efecto de la aproximación de una matriz que representa una imagen por una de menor rango.

La primera imagen corresponde a la imagen original sin reducción en el rango matricial. Se utilizó la descomposición en valores singulares para aproximar esta matriz mediante otras de rango $k = 200$, $k = 60$ y $k = 10$. Como se puede observar, en el caso en que $k = 200$ el cambio es casi imperceptible a pesar de que se disminuyó el rango a más de la mitad. Ésto podría significar que esta matriz puede ser bien representada mediante otra de menor rango. Como se verá en el capítulo 3, esta característica es bastante importante cuando lo que se desea hacer es aproximación de matrices mediante otra de menor rango. Cabe recalcar que no todas las matrices cuentan con esta peculiaridad por lo que existen casos en los que no es buena idea aproximar mediante otra matriz de menor rango.

En el caso en que $k = 60$ se puede percibir más fácilmente el cambio; no debe olvidarse que se redujo el rango original a casi su séptima parte. De manera empírica lo que se ha observado cuando se aproximan matrices mediante otras de menor rango es que si la imagen no tiene bordes muy marcados, como en este caso, la aproximación generalmente es buena. Es decir, en una imagen con bordes más marcados generalmente el rango por el que se puede aproximar sin perder mucha calidad en la imagen no suele ser tan pequeño.

En el caso en que $k = 10$ ya es muy difícil distinguir la imagen original. Lo anterior era de esperarse puesto que se está reduciendo a más de 40 veces el rango original. En el caso de imágenes, aproximar la matriz correspondiente mediante otra de menor rango se percibe como bajar la calidad de la imagen de cierta manera ya que se nota más borrosa.

1.3.6.1. Aproximaciones de menor rango utilizando los vectores y valores propios Sea B una matriz simétrica, semidefinida positiva y de rango r. Las aproximaciones de menor rango para B basadas en las descomposiciones presentadas en la subsección anterior son:

Aproximación utilizando la descomposición en valores singulares
Como se explicó anteriormente, tomando los primeros k valores y vectores singulares izquierdos (o derechos) de B se obtiene la siguiente aproximación de rango k:

$$B_k = U_k^B D_k^B \left(V_k^B\right)^t \tag{5}$$

Aproximación utilizando la descomposición en matriz proyección
De manera análoga al caso anterior, tomando los primeros k vectores singulares izquierdos de B se obtiene la siguiente aproximación de rango k

$$B_k = U_k^B \left(U_k^B\right)^t B.$$

Aproximación utilizando la descomposición en vectores propios
Debido a la estructura de esta descomposición, lo que se utilizará serán los primeros k vectores singulares izquierdos de la matriz A. Es decir, la aproximación será de la forma:

$$B_k = AV_k^A \left(AV_k^A\right)^t.$$

1.3.6.2. Aproximaciones utilizando estimaciones de los vectores y valores propios En lo anterior se mostraron las aproximaciones de rango k para la matriz B que provienen de utilizar los primeros k vectores singulares de B o de A según la descomposición con la que se cuenta. En el caso en que no es posible conocer o calcular los vectores que se necesitan, se pueden utilizar aproximaciones o estimaciones de estos. Es decir, se va a aproximar la aproximación B_k estimando los vectores necesarios y sustituyéndolos en las expresiones de B_k. Dichas aproximaciones se utilizarán bastante en los siguientes capítulos.

Aproximación de B_k por descomposición espectral

$$\widehat{B}_k^{\text{esp}} = \widehat{U_k^B}\widehat{D_k^B}\left(\widehat{U_k^B}\right)^t. \tag{6}$$

A esta aproximación se le llamará aproximación de rango k para B por descomposición espectral.

Aproximación de B_k por descomposición en matriz proyección

$$\widehat{B}_k^{\text{proy}} = \widehat{U_k^B}\left(\widehat{U_k^B}\right)^t B. \tag{7}$$

A esta aproximación se le llamará aproximación de rango k para B por descomposición en matriz proyección.

Aproximación de B_k por descomposición en vectores propios

En este caso se van a aproximar o estimar los primeros k vectores singulares izquierdos de la matriz A, para obtener el estimador:

$$\widehat{B}_k^{\text{vp}} = A\widehat{V_k^A}\left(A\widehat{V_k^A}\right)^t. \tag{8}$$

A esta aproximación se le llamará aproximación de rango k para B por descomposición en vectores propios.

Se debe recalcar que en este caso las aproximaciones $\widehat{B}_k^{\text{esp}}$, $\widehat{B}_k^{\text{proy}}$ y $\widehat{B}_k^{\text{vp}}$ no necesariamente coinciden. Debido a que no se utilizan los vectores propios reales sino aproximaciones de estos, se pueden perder ciertas propiedades como la ortogonalidad sin la cual no se tiene la equivalencia entre las matrices.

2. Métodos kernel o métodos de transformaciones implícitas

La idea de los métodos kernel es primero mapear los datos a un (otro) espacio dotado de producto punto, en el cual los datos tengan una estructura más sencilla para despúes analizarlos mediante técnicas clásicas de análisis de datos. Por ejemplo, mapearlos a un espacio en donde las observaciones presenten una estructura aproximadamente lineal. Como se verá después, en lugar de trabajar con los datos transformados explícitamente, se trabajará con los productos punto entre las observaciones en el nuevo espacio. Como puede verse en Cristianini *et al.* (2004) [7], la matriz que contiene dichos productos punto es llamada matriz kernel o matriz de Gram K.

Al reescribir alguna técnica de análisis de datos de tal forma que en lugar de trabajar con la matriz de datos se trabaje con la matriz de productos punto entre los datos se dice que se está *kernelizando* dicha técnica. Es decir, un método kernel puede verse como un método que resulta de reescribir (kernelizar) una técnica de análisis de datos en función de productos punto. Es posible kernelizar los métodos estadísticos clásicos como: PCA, regresión, clustering, entre otros.

Como ya se mencionó, en esta tesis se eligió PCA como método a kernelizar. Los métodos de análisis estadístico no supervisado como PCA, buscan un patrón en los datos a partir de una función de interés (en PCA, es una función de proyección). Se buscará en su versión kernelizada, un patrón en los datos transformados, a partir de una función de interés formulada en términos de productos punto (K). Así, lo que lo que se obtendrá de su versión kernelizada será un patrón para los datos en el espacio transformado. En la Figura 3 se muestra el procedimiento que se sigue cuando se utiliza un método de análisis estadístico no supervisado kernelizado.

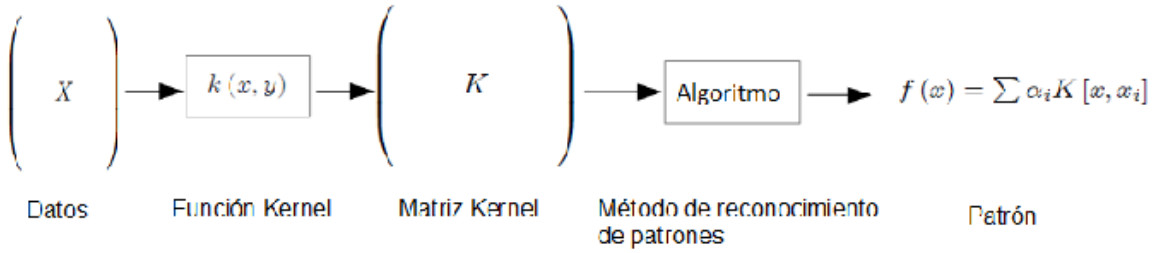

Figura 3: Etapas de la aplicación de un método Kernel que se utiliza para reconocimiento de patrones.

Para obtener un método kernel son necesarios dos ingredientes principales: una función kernel y un método de análisis de datos que se desea kernelizar. Por dicha razón, este capítulo contiene cuatro subsecciones principales. En las primeras dos subsecciones se definen las funciones kernel y los métodos kernel, se presentan algunas de sus propiedades y funciones kernel particulares. En la tercera subsección se recuerdan los resultados y suposiciones importantes de PCA y en la cuarta,

se presenta su versión kernelizada Kernel PCA (KPCA). Se consideró una quinta subsección, en la cual se presenta una aplicación específica tanto de PCA como de KPCA.

2.1. Función kernel

En esta sección se presenta la definición formal de una función kernel y algunas de sus propiedades.

Definición 3 *(Función kernel) Sean x, $y \in X$, donde X denota el espacio de entrada. Un kernel k es una función que calcula el producto punto entre x y y transformados mediante cierta función ϕ,*

$$k(x,y) = \langle \phi(x), \phi(y) \rangle , \tag{9}$$

donde ϕ es un mapeo de X a un espacio de Hilbert H dotado de producto punto y al cual se le llamará espacio de características.

Es importante observar dos cosas acerca de esta definición. La primera es que x y y no deben ser vectores necesariamente. Por ejemplo, x y y pueden ser cadenas de carácteres. La segunda, es que si se conoce el kernel k asociado a la función ϕ, entonces no es necesario calcular explícitamente $\phi(x)$, $\phi(y)$ y luego calcular sus productos puntos, basta calcular $k(x,y)$. Debido a lo anterior, las funciones kernel también son conocidas como transformaciones implícitas. Esta característica de las funciones kernel implica, en ciertas situaciones, una gran ventaja computacional. El ejemplo clásico de dicha ventaja se presenta a continuación.

Considérese el mapeo de R^2 a R^3 definido como $\phi(x) = \phi((x_1, x_2)) = \left(x_1^2, x_2^2, \sqrt{2}x_1x_2\right)$. Sean x y y dos puntos en R^2, su producto punto en el espacio transformado está dado por:

$$\begin{aligned} \langle \phi(x), \phi(y) \rangle &= \left\langle \left(x_1^2, x_2^2, \sqrt{2}x_1x_2\right), \left(y_1^2, y_2^2, \sqrt{2}y_1y_2\right) \right\rangle \\ &= x_1^2y_1^2 + x_2^2y_2^2 + 2x_1x_2y_1y_2 \\ &= (x_1y_1 + x_2y_2)^2 \\ &= \langle x, y \rangle^2 \\ &= k(x,y). \end{aligned} \tag{10}$$

Es decir, la función $k(x,y) = \langle x, y \rangle^2$ permite calcular el producto punto de dos datos transformados mediante ϕ calculando solamente el producto punto entre los dos datos en el espacio original elevado al cuadrado. La ventaja computacional radica en que al utilizar $k(,)$ solamente se necesitan realizar 3 multiplicaciones y una suma, en cambio si se hace el cálculo explícitamente se deben realizar 10 multiplicaciones y 3 sumas.

Si se aplica la función kernel a un conjunto de datos surge de manera natural el concepto de una matriz cuyas entradas sean el valor del kernel entre los datos. A esta matriz, cuya definición se presenta a continuación, se le llama matriz kernel o matriz de Gram.

Definición 4 *(Matriz kernel o matriz de Gram) Dado un kernel* $k(,)$ *y un conjunto de datos* $x_1, x_2, ..., x_n$ *, la matriz* K *de dimensión* $n \times n$ *definida como:*

$$K = \begin{pmatrix} k(x_1, x_1) & k(x_1, x_2) & \cdots & k(x_1, x_n) \\ k(x_2, x_1) & k(x_2, x_2) & \cdots & k(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & \cdots & k(x_n, x_n) \end{pmatrix}$$

es llamada *matriz kernel* o *matriz de Gram de* $k(,)$ *con respecto a* $x_1, x_2, ..., x_n$.

Por la definición del kernel se tiene que la matriz de Gram es simétrica y semidefinida positiva. Es simétrica puesto que $k(x, y) = \langle \phi(x), \phi(y) \rangle = \langle \phi(y), \phi(x) \rangle = k(y, x)$. Es semidefinida positiva ya que para cualquier vector v se tiene que

$$\begin{aligned} v^t K v &= \sum_{i,j=1}^{n} v_i v_j K_{i,j} \\ &= \sum_{i,j=1}^{n} v_i v_j \langle \phi(x_i), \phi(x_j) \rangle \\ &= \left\langle \sum_{i=1}^{n} v_i \phi(x_i), \sum_{j=1}^{n} v_j \phi(x_j) \right\rangle \\ &= \left\| \sum_{j=1}^{n} v_i \phi(x_i) \right\|_2^2 \geq 0. \end{aligned}$$

Como puede verse en Smola *et al.* (2014) [14] y Muñiz (2011) [25], ser simétrica y semidefinida positiva es también una condición suficiente para ser una matriz de Gram.

Ejemplos de funciones kernel

- **Kernel lineal**

Este kernel es el más simple de todos y está dado por

$$\begin{aligned} k(x, y) &= \langle x, y \rangle \\ &= x^t y. \end{aligned}$$

Para este kernel se tiene que la función ϕ es la identidad, es decir $\phi(x) = x$ y por lo tanto el espacio de características tiene la misma dimensión que el espacio original.

- **Kernel polinomial de grado p**

Este kernel está definido por

$$k(x,y) = (\langle x,y\rangle + c)^p,$$

donde $p \in N$ y $c \geq 0$. A c se le conoce como offset y si $c = 0$ entonces el kernel es llamado kernel polinomial homogéneo de grado p.

Como puede observarse, el kernel lineal es un caso particular del kernel polinomial tomando $c = 0$ y $p = 1$. El kernel $k(x,y) = \langle x,y\rangle^2$ que se utilizó en (10) es un kernel polinomial homogéneo de grado 2. Puede verse que el espacio de características está formado por todos los monomios de grado p y que su dimensión está dada por $\binom{d+p}{p}$ [7].

- **Kernel gaussiano de parámetro σ**

Este kernel también es conocido como kernel de base radial y está definido por

$$k(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right),$$

donde $\sigma > 0$. Como puede verse en Muñiz (2011) [25], este kernel cuenta con algunas características entre las cuales resaltan las siguientes:

- Cuando se tienen n diferentes obervaciones, la matriz de Gram correspondiente de dimensión $n \times n$ es de rango completo.

- Los datos transformados tienen norma 1 ya que$\|\phi(x_i)\|_2 = \sqrt{\langle \phi(x_i), \phi(x_i)\rangle} = \sqrt{k(x_i,x_i)} = \sqrt{\exp(0)} = \sqrt{1} = 1$. Por lo tanto, los datos transformados están sobre una hiperesfera de radio 1.

- El kernel gaussiano es un kernel invariante bajo traslaciones o estacionario, lo cual significa que es una función basada en la diferencia $x - y$. Esta última propiedad resulta necesaria para algunos métodos de aproximación de matrices que se presentarán en el capítulo 3.

Una manera de hacerse una idea de cómo afecta el parámetro σ del kernel gaussiano es pensando que la diferencia entre dos datos transformados es $\|\phi(x_i) - \phi(x_i)\|^2 = 2 - 2\exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma^2}\right)$. Por lo tanto, el parámetro σ determina la escala en que se mide la distancia entre los datos transformados.

2.2. Génesis de los métodos kernel

En esta subsección se presentará una idea de cómo surgieron los métodos kernel a partir de las funciones kernel. El propósito es dar un esbozo del planteamiento. Los detalles técnicos pueden encontrarse en Tibshirani *et al.* (2001) [12] y Muñiz (2011) [25]. Pueden considerarse dos diferentes ideas que hicieron que los métodos kernel surgieran de manera natural. La primera fue la de reescribir los métodos clásicos de análisis de datos de forma diferente; la segunda fue llevar los métodos clásicos lineales de análisis de datos al terreno no lineal.

2.2.1. Los métodos kernel como reescritura de métodos clásicos

Considérese, como ejemplo, la función que se minimiza en regresión ridge:

$$\min_{\beta} \sum_{i=1}^{n} \left(y_i - X[i,]^t\beta\right)^2 + \lambda \sum_{j=1}^{d} \beta_j^2, \tag{11}$$

donde X es la matriz de datos y $y = (y_i)_{i=1}^n$ los valores de la variable respuesta.

La solución es $\widehat{\beta} = \underbrace{\left(X^tX + \lambda I\right)^{-1} X^t}_{a} y$. Se puede reescribir a utilizando la siguiente propiedad para matrices: sean F^{-1} y H^{-1} matrices invertibles y G cualquier matriz, entonces

$$\left(F^{-1} + G^tH^{-1}G\right)^{-1} G^tH^{-1} = FG^t\left(GFG^t + H\right)^{-1}. \tag{12}$$

Utilizando (12) con $F^{-1} = \lambda I$, $G = X$ y $H^{-1} = I$, se reescribe a de la siguiente manera:

$$\begin{aligned}
\left(X^tX + \lambda I\right)^{-1} X^tI &= \frac{1}{\lambda} IX^t\left(I + X\frac{1}{\lambda}IX^t\right)^{-1} \\
&= \frac{1}{\lambda} X^t\left(\frac{1}{\lambda}\left(\lambda I + XX^t\right)\right)^{-1} \\
&= X^t\left(\lambda I + XX^t\right)^{-1} \\
&= X^t\left(\lambda I + K\right)^{-1},
\end{aligned}$$

donde K es la matriz kernel para los datos utilizando un kernel lineal.
Es decir, $\widehat{\beta}$ puede reescribirse en función de la matriz kernel de la forma $\widehat{\beta} = X^t\left(\lambda I + K\right)^{-1} y$. Además, las predicccciones están dadas por:

$$\begin{aligned}
\widehat{Y} &= X\widehat{\beta} \\
&= XX^t\left(\lambda I + K\right)^{-1} y \\
&= K\left(\lambda I + K\right)^{-1} y.
\end{aligned}$$

Por lo tanto, para calcular las predicciones solamente se necesita conocer la matriz kernel. Así, los métodos kernel surgen como una forma de reescribir la solución dada por un método de análisis de datos en función de un kernel. Esta es quizás la forma más sencilla de ver un método kernel.

2.2.2. Los métodos kernel como generalización al campo no lineal

Como se había mencionado, los métodos kernel también pueden verse como una solución al problema de generalizar un método de análisis de datos lineal al campo no lineal. Considérese de nuevo el problema de regresión ridge (11), el cual es de la forma:

$$\min_{f \in H} L\left(f\left(X\left[1,\right]\right), ..., f\left(X\left[n,\right]\right)\right) + g\left(\|f\|^2\right), \tag{13}$$

donde $L\left(f\left(X\left[1,\right]\right), ..., f\left(X\left[n,\right]\right)\right)$ es una función de costo, $\|f\|^2$ es un funcional de penalización (si $f(x) = \beta^t x$ entonces $\|f\|^2 = \|\beta\|^2$), H es un espacio de funciones equipado con una norma $\|f\|^2$ y

g es una función no decreciente. Al considerar funciones f no lineales, se obtiene una extensión de regresión ridge.

Puede verse en Tibshirani *et al.* (2011) [12] que la solución de (13) es de la forma:

$$f^* = \sum_{i=1}^{n} \alpha_i k\left(X[i,], .\right). \tag{14}$$

Es decir, la solución al problema de regresión ridge (no lineal) solamente depende de los datos a través de la función kernel. Así, los métodos kernel surgen como una generalización al contexto no lineal de métodos de análisis de datos clásicos.

En el Apéndice A se presenta un bosquejo de por qué la solución al problema (13) tiene la forma (14). Una demostración detallada puede verse en Wahba (1990) [32].

2.3. Análisis de componentes principales (PCA)

En esta subsección se recordará PCA de manera burda, ya que el objetivo es kernelizar este método de tal forma que se haga PCA sobre los datos transfomados mediante un kernel. Una explicación más detallada puede verse en Izenman (2008) [16].

El análisis de componentes principales es una herramienta de análisis multivariado la cual generalmente es utilizada para reducir la dimensionalidad de un conjunto de datos. Con dicha técnica, se trata de obtener una representación de los datos que mantiene gran parte de la información de estos y con la que es más fácil trabajar y analizar.

Sea X la matriz de datos de dimensión $n \times d$. Por simplicidad se supondrá que los datos están centrados por columna. Si no están centrados, simplemente para cada columna de X en lugar de utilizar $X[,j]$ se utilizará $X[,j] - \frac{1}{n}\sum_{i=1}^{n} X[i,j]$, $j = 1, .., d$. Así, la matriz de datos X es de la forma:

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}_{n\times d} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}_{n\times d},$$

donde x_{ij} representa la medición de la característica j hecha al individuo i y x_i representa el vector $(x_{i1}, x_{i2}, ..., x_{id})$.

Lo que se pretende al hacer análisis de componentes principales es encontrar direcciones de proyección que maximicen la varianza y además sean ortogonales entre sí. Intuitivamente, la razón por la que se buscan las direcciones que maximizan la varianza es debido a que una dirección en la cual los valores proyectados son casi constantes (tienen poca variabilidad) no ofrece mayor información.

A las direcciones que maximizan la varianza se les conoce como componentes principales. Sea $\widehat{Var}$ el estimador de la covarianza para estos datos. Las r componentes principales están dadas por los r eigenvectores asociados a los r eigenvalores más grandes de la matriz $\widehat{Var}$ ordenados de manera descencente. Se denotará por $(\lambda_1, w_1), ..., (\lambda_r, w_r)$ a las r parejas eigenvalor-eigenvector de la matriz

$\widehat{Var}$. La proyección de los datos en el i-ésimo componente principal está dada por

$$P_{w_i}(X) = Xw_i.$$

La proyección de los datos en los primeros r componentes principales puede verse de manera matricial. Si se denota por W_r la matriz cuyas columnas están formadas por los primeros r eigenvectores de $\widehat{Var}$ de la siguiente manera:

$$W_r = [w_1, \ldots, w_r],$$

entonces la proyección de los datos en los primeros r componentes principales está dada por

$$Y_r = XW_r.$$

A Y_r se le llama *matriz de datos proyectados*.

PCA es una herramienta bastante eficiente en el sentido que generalmente con pocos componentes principales se captura bastante bien el comportamiento de los datos. Como se puede notar, dichos componentes principales son combinaciones lineales de las variables. En este sentido, si el comportamiento o patrón de los datos no es lineal entonces no se logrará capturar bien utilizando PCA. Cuando se tienen datos para los cuales se sospecha que tienen una estructura no lineal, una solución es mapear los datos utilizando una función kernel y luego hacer análisis de componentes principales a estos datos mapeados. De esta manera, se kerneliza PCA.

2.4. Kernel PCA

En esta subsección se describe el procedimiento que se utiliza para hacer Kernel PCA, así como algunas características de esta técnica. Como ya se mencionó, la idea es mapear los datos a un espacio dotado de producto punto en donde las datos tengan una estructura aproximadamente lineal y luego utilizar PCA. Así, Kernel PCA consiste en hacer PCA a partir de la matriz de Gram K.

La idea del procedimiento para hacer Kernel PCA es la siguiente:

1. Se mapean los datos mediante una función $\phi(x) = (\phi_1(x), \phi_2(x), ...)$ a otro espacio.

2. Se hace implícitamente un análisis de componentes principales a los datos transformados, es decir a

$$\widetilde{X} = \begin{pmatrix} \phi(x_1) \\ \phi(x_2) \\ \vdots \\ \phi(x_n) \end{pmatrix}_{n \times \widetilde{d}},$$

donde $\widetilde{d}$ es la dimensión del espacio de características y $\phi(x_i) = \left(\phi_1(x_i), ..., \phi_{\widetilde{d}}(x_i)\right)^t$.

Hacer PCA sobre $\widetilde{X}$ significa encontrar las parejas eigenvalor-eigenvector de la matriz estimada de covarianzas. Una elección que se emplea usualmente para dicho estimador es:

$$\widetilde{Var} = \frac{1}{n}\widetilde{X}^t\widetilde{X}.$$

Nótese que $\widetilde{Var}$ es de la forma A^tA. Utilizando la propiedad que se enuncia a continuación, se obtiene una relación entre las parejas eigenvalor-eigenvector de $\widetilde{Var}$ y las de la matriz kernel K.

Propiedad. Las matrices AA^t y A^tA tienen los mismos eigenvalores, y si v es vector propio de A^tA y u es vector propio de AA^t con valor propio λ, entonces

$$v = \frac{A^t u}{\sqrt{\lambda}}. \tag{15}$$

Utilizando (15) con $A = \widetilde{X}$ se tiene que si (λ_j, w_j) y (τ_j, v_j) representan las parejas eigenvalor-eigenvector j-ésima de $\widetilde{Var}$ y K respectivamente, entonces

$$\begin{aligned} w_j &= \sum_{i=1}^{n} \tau_j^{-\frac{1}{2}} v_j[i] \phi(x_i) && (16) \\ &= \tau_j^{-\frac{1}{2}} \widetilde{X}^t v_j. && (17) \end{aligned}$$

La proyección de un punto x en el j-ésimo componente principal está dada por

$$\begin{aligned} P_{w_j}(x) &= \phi(x) w_j && (18) \\ &= \sum_{i=1}^{n} \tau_j^{-\frac{1}{2}} v_j[i] \langle \phi(x_i), \phi(x) \rangle \qquad \text{(por (16))} \\ &= \sum_{i=1}^{n} \tau_j^{-\frac{1}{2}} v_j[i] K(x_i, x). \end{aligned}$$

Consecuentemente, no es necesario calcular la transformación ϕ explícitamente sino solamente calcular la matriz K. Aún más, para calcular la $j-$ésima columna de la matriz de datos proyectados es más conveniente utilizar la siguiente igualdad:

$$\begin{aligned} Y_r[,j] &= \widetilde{X} W_r[,j] \\ &= \widetilde{X} \tau_j^{-\frac{1}{2}} \widetilde{X}^t v_j \qquad \text{(por (17))} \\ &= \tau_j^{-\frac{1}{2}} \widetilde{X} \widetilde{X}^t v_j \\ &= \tau_j^{-\frac{1}{2}} K v_j \\ &= \tau_j^{-\frac{1}{2}} \tau_j v_j \qquad \text{(pues } v_j \text{ v.p. de } K) \\ &= \tau_j^{\frac{1}{2}} v_j. \end{aligned}$$

Así, para obtener las proyecciones dadas por KPCA basta con calcular los vectores y valores propios de la matriz de Gram.

En lo anterior, se está suponiendo que los datos transformados $\widetilde{X}$ están centrados. Generalmente la condición anterior no se cumple por lo que se mostrará la idea del procedimiento para obtener la K correspondiente a los datos transformados centrados. Si se denota por $k_c(x_i, x_j)$ el kernel

centrado evaluado en (x_i, x_j), entonces para cualquier $i, j = 1, ..., n$ se tiene que:

$$\begin{aligned} k_c(x_i, x_j) &= \left\langle \phi(x_i) - \frac{1}{n}\sum_{l=1}^{n}\phi(x_l), \phi(x_j) - \frac{1}{n}\sum_{l=1}^{n}\phi(x_l) \right\rangle \\ &= \langle \phi(x_i), \phi(x_j)\rangle - \frac{1}{n}\sum_{l=1}^{n}\langle \phi(x_i), \phi(x_l)\rangle - \frac{1}{n}\sum_{l=1}^{n}\langle \phi(x_l), \phi(xj)\rangle + \frac{1}{n^2}\sum_{l=1}^{n}\sum_{m=1}^{n}\langle \phi(x_l), \phi(x_m)\rangle \\ &= k(x_i, x_j) - \frac{1}{n}\sum_{l=1}^{n}k(x_i, x_l) - \frac{1}{n}\sum_{l=1}^{n}k(x_l, x_j) + \frac{1}{n^2}\sum_{l=1}^{n}\sum_{m=1}^{n}k(x_l, x_m) \\ &= K[i,j] - K[i,]\frac{\mathbf{1}}{n} - \frac{\mathbf{1}^t}{n}K[,j] + \frac{\mathbf{1}^t}{n}K\frac{\mathbf{1}}{n}. \end{aligned} \tag{19}$$

Utilizando (19), se tiene que la matriz de Gram sobre los datos transformados centrados se obtiene calculando:

$$K_c = K - K\frac{1}{n}J - \frac{1}{n}JK + \frac{1}{n^2}JKJ, \tag{20}$$

donde J denota la matriz de dimensión $n \times n$ cuyas entradas son todas 1.

2.4.1. Ventajas y desventajas de KPCA

En la subsección anterior se mostró el procedimiento a seguir cuando se desea analizar datos mediante KPCA. En esta sección se presentarán algunas características y resultados importantes acerca de este método.

Como ya se mencionó, una de las formas en que surge KPCA es como una solución al caso en que la estructura de los datos es no lineal. Así, KPCA puede considerarse como una generalización de PCA al campo no lineal.

Una desventaja de Kernel PCA es que no siempre cuenta con una interpretación evidente en el espacio original de los datos. Cuando se realiza PCA se tiene una interpretación bastante clara del resultado en base a los datos, puesto que los componentes principales son combinaciones lineales de las variables. Así, dependiendo de las combinaciones obtenidas se puede deducir qué variable o conjunto de variables son las que explican la variabilidad de los datos. En el caso de KPCA esto no es posible ya que se está trabajando en el espacio generado por los productos puntos de los datos transformados. Trabajo en torno a dicha interpretación puede encuentrarse en Muñiz (2011) [25].

Para el caso en que se utiliza KPCA con un kernel gaussiano σ, Muñiz (2011) [25] dió la interpretación de que el procedimiento busca un *contraste* en las densidades de los datos. Por lo anterior, KPCA con un kernel gaussiano puede resultar útil cuando se desea detectar grupos o clusters en los datos. Como se observará en el siguiente ejemplo, en dicho caso, las componentes principales generalmente separan los datos que corresponden a diferentes grupos.

Ejemplo Se generaron dos grupos de datos provenientes de distribuciones normales con diferente media e igual matriz de covarianza. La matriz de covarianza para ambas distribuciones es $\Sigma = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix}$ y las medias son $\mu_1 = (1, 1)$ y $\mu_2 = (-1, -1)$.

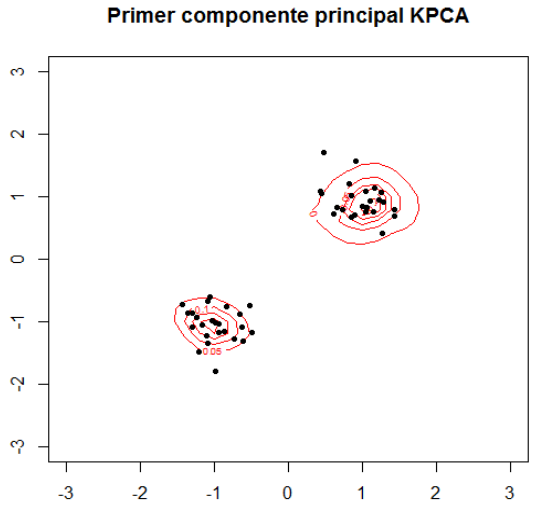

Figura 4: Gráfica de contornos de la función de proyección sobre el primer componente principal obtenido con KPCA con un kernel gaussiano.

En la Figura 4 se puede observar que el primer componente principal separa ambos grupos de datos. Los datos para las cuales la función de proyección es positiva forman un grupo y los datos para los cuales la función de proyección es negativa, forman otro. Este ejemplo es bastante sencillo pero sirve para formarse una idea en cuanto al comportamiento del método. Ejemplos más interesantes se pueden encontrar en Muñiz (2011) [25].

Dependiendo del tipo de datos con que se cuente, KPCA puede representar una ventaja o desventaja computacional con respecto a PCA. En el caso en que se cuenta con menos observaciones que variables, es decir que $d > n$ KPCA con un kernel lineal produce una ventaja computacional ya que en lugar de trabajar con una matriz estimada de covarianzas de dimensión $d \times d$, como lo hace PCA, trabaja con una matriz de dimensión $n \times n$. Claramente, la desventaja ocurre en el momento en que se trabaja con un conjunto de datos en el cual se tienen más observaciones que variables, es decir que $n > d$. Una desventaja es que no para cualquier $\phi(\cdot)$, se tiene que $k(x, y)$ es más fácil de calcular que $\langle \phi(x), \phi(y) \rangle$ explícitamente.

2.5. Reconstrucción de la proyección obtenida con PCA y Kernel PCA en el espacio original

En esta subsección se presentará el procedimiento que se sigue cuando a partir de las matrices de datos proyectados obtenidas con PCA y KPCA, se desea obtener una aproximación del dato en el espacio original. PCA y KPCA producen nuevas coordenadas (proyecciones) para cada dato en un espacio generalmente de menor dimensión al original. Sin embargo, existen algunas aplicaciones

en las cuales se desea regresar al espacio original de los datos utilizando dichas proyecciones. Se empleará el caso en que se utiliza PCA para explicar en qué consiste la reconstrucción del dato en el espacio original ya que, en este caso, el procedimiento es sencillo.

Cada renglón de la matriz de datos proyectados Y, obtenida mediante PCA, representa las coordenadas de cada dato conforme a la base dada por los componentes principales $\{w_1, ..., w_r\}$. Para obtener las coordenadas de cada uno de los $Y[i,]_{i=1}^n$ conforme a la base del espacio original lo único que se debe hacer es multiplicar cada entrada de $Y[i,]_{i=1}^n$ por el componente principal correspondiente. Es decir, la preimagen se obtiene utilizando la ecuación:

$$\widehat{x_i} = \sum_{j=1}^{r} Y[i,j] w_j. \tag{21}$$

Lo anterior es posible debido a que en PCA los componentes principales pertenecen al espacio generado por los datos ya que son los vectores propios de la matriz estimada de covarianza $\widehat{Var} \propto X^t X$.

Una aplicación es en el caso en que cada objeto representa una imagen. En este caso, se debe encontrar la forma de reconstruir la imagen a partir de las nuevas coordenadas obtenidas. El resultado deseable de dicho procedimiento es que la preimagen (imagen o dato reconstruido) mantenga las características importantes de la imagen que se está aproximando.

Kernel PCA provee proyecciones de los datos transformados sobre los vectores de la matriz estimada de covarianzas $\widetilde{Var}$. Las proyecciones son suficientes cuando se desea utilizar alguna herramienta de clasificación o encontrar patrones en los datos, pero cuando se desea la preimagen de estas proyecciones se necesita de otros cálculos. Se describirá el procedimiento para calcular la preimagen de un dato que ha sido transformado usando un kernel gaussiano de parámetro σ y proyectado sobre los primeros r vectores singulares izquierdos (o derechos) de K. Sin embargo, como puede verse en Mika *et al.* (1999) [23], este procedimiento puede hacerse para cualquier kernel conocido. A continuación se describe el procedimiento cuando se usa un kernel gaussiano de parámetro σ.

Supóngase que se realizó kernel PCA con un kernel gaussiano de parámetro σ y se obtuvo la matriz de datos proyectados Y. Es decir, Y es una matriz que contiene las proyecciones de los datos transformados y no de los datos en el espacio original. Debido a que se hizo PCA para una matriz de datos transformados, se puede obtener una estimación de la forma (21) para $\phi(x)$. Así,

$$\widehat{\phi(x)} = \sum_{j=1}^{r} y_j(x) w_j, \tag{22}$$

donde w_j representa el $j-$ésimo vector propio de la matriz de covarianza estimada para los datos transformados $\widetilde{Var} = \frac{1}{n}\widetilde{X}^t\widetilde{X}$ y $y_j(x)$ denota la $j-$ésima entrada de la proyección del dato x.

$\widehat{\phi(x)}$ es una aproximación al valor real $\phi(x)$. Se desea encontrar un z en el espacio original tal que $\phi(z) \approx \widehat{\phi(x)}$. En el caso de PCA, ϕ es la función identidad y por lo tanto $\phi(z) = z = \sum_{j=1}^{r} y_j(x) w_j$. En el caso de KPCA, lo anterior generalmente no se cumple.

Debido a que es difícil encontrar la preimagen x exacta, lo que se hace es encontrar un z tal que $\phi(z) \approx \phi(x)$. Para encontrar dicho z lo que se hace es utilizar el método del gradiente para

minimizar la función:

$$\begin{aligned}\rho(z) &= \left\|\phi(z) - \widehat{\phi(x)}\right\|^2 \\ &= \|\phi(z)\|^2 - 2\left\langle \phi(z), \widehat{\phi(x)} \right\rangle + \left\|\widehat{\phi(x)}\right\|^2 \\ &= k(z,z) - 2\left\langle \phi(z), \widehat{\phi(x)} \right\rangle + c \\ &= 1 - 2\left\langle \phi(z), \widehat{\phi(x)} \right\rangle + c,\end{aligned}$$

donde c es una constante que no depende de z y la última igualdad se obtuvo recordando que para el caso del kernel gaussiano se cumple que $k(z,z) = 1$ para cualquier z.

Así,

$$\begin{aligned}\rho(z) &= 1 - 2\left\langle \phi(z), \widehat{\phi(x)} \right\rangle + c \\ &= -2\left\langle \phi(z), \widehat{\phi(x)} \right\rangle + c' \\ &= -f(z),\end{aligned}$$

donde c' representa una constante que no depende de z. Por lo tanto, para minimizar $\rho(z)$ con respecto a z basta maximizar la función $f(z)$.

Sustituyendo (16) en (22) y a su vez en la expresión de $f(z)$ con $a_{ij} = \tau_j^{-\frac{1}{2}} v_j[i]$ se tiene que:

$$\begin{aligned}f(z) &= 2\left\langle \phi(z), \widehat{\phi(x)} \right\rangle + c' \\ &= 2\sum_{j=1}^{m} y_j(x) \sum_{i=1}^{n} a_{ij} k(z, x_i) + c'.\end{aligned}$$

Si se define

$$\gamma_i(x) = \sum_{j=1}^{m} y_j(x)\, a_{ij},$$

se tiene que

$$f(z) = 2\sum_{i=1}^{n} \gamma_i(x)\, k(z, x_i) + c'.$$

Se debe recalcar que para cada renglón x_l en el conjunto de datos se tiene un vector $\gamma(x_l) = (\gamma_1(x_l), ..., \gamma_n(x_l))$.

Supóngase que se quiere encontrar la preimagen del objeto x_l, $l \in \{1, ..., n\}$. Debido a que para los datos se desea hacer KPCA y por lo tanto se utilizará un kernel centrado, en el caso de la estimación también se debe utilizar. Es decir para cada x_l, $l \in \{1, ..., n\}$ se tiene que maximizar la función:

$$f(z) = 2\sum_{i=1}^{n} \gamma_i(x_l)\, k_c(z, x_i) + c'.$$

Por lo que, se quiere encontrar el z que cumple que:

$$\nabla_z f(z) = 2\sum_{i=1}^{n} \gamma_i(x_l)\, k_c'(z, x_i) = 0$$

Se puede observar que es necesario calcular la derivada del kernel centrado para poder encontrar z. Derivando la expresión (19) para el kernel gaussiano de parámetro σ se tiene que:

$$\begin{aligned}
\nabla_z k_c(z, x_i) &= \nabla_z k(z, x_i) - \nabla_z \left(\frac{1}{n}\sum_{s=1}^{n} k(x_i, x_s)\right) - \nabla_z \left(\frac{1}{n}\sum_{s=1}^{n} k(z, x_s)\right) + \nabla_z \left(\frac{1}{n^2}\sum_{s=1}^{n}\sum_{m=1}^{n} k(x_s, x_m)\right) \\
&= -\frac{1}{\sigma^2}\left[k(z, x_i)(z - x_i) - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s)(z - x_s)\right] \\
&= -\frac{1}{\sigma^2}\left[k(z, x_i) - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s)\right] z + \frac{1}{\sigma^2}\left[k(z, x_i)\, x_i - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s) x_s\right].
\end{aligned}$$

Por lo tanto,

$$\begin{aligned}
\nabla_z f(z) &= 2\sum_{i=1}^{n} \gamma_i(x_l)\, \nabla_z k_c(z, x_i) \\
&= 2\sum_{i=1}^{n} \gamma_i(x_l) \left(-\frac{1}{\sigma^2}\left[k(z, x_i) - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s)\right] z + \frac{1}{\sigma^2}\left[k(z, x_i)\, x_i - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s) x_s\right]\right).
\end{aligned} \tag{23}$$

Igualando a cero la ecuación anterior (23), se obtiene que el z que maximice $f(z)$ debe cumplir:

$$z = \frac{\sum_{i=1}^{n} \gamma_i(x_l) \left(k(z, x_i)\, x_i - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s) x_s\right)}{\sum_{i=1}^{n} \gamma_i(x_l) \left(k(z, x_i) - \frac{1}{n}\sum_{s=1}^{n} k(z, x_s)\right)}.$$

Puede verse en Mika *et al.* (1999) [23], que para el kernel no centrado, se cumplen ciertas condiciones necesarias para garantizar la convergencia al óptimo z. Se supondrá que dichas condiciones se cumplen para el kernel centrado. Así, para calcular la preimagen z a partir de las proyecciones obtenidas con KPCA cuando se utiliza un kernel gaussiano centrado, se utilizará el método del punto fijo. Se calcula z para cada x_l, $l \in \{1, ..., n\}$ de manera iterativa de la siguiente forma:

$$z_{t+1} = \frac{\sum_{i=1}^{n} \gamma_i (x_l) \left(k(z_t, x_i) x_i - \frac{1}{n} \sum_{s=1}^{n} k(z_t, x_s) x_s \right)}{\sum_{i=1}^{n} \gamma_i (x_l) \left(k(z_t, x_i) - \frac{1}{n} \sum_{s=1}^{n} k(z_t, x_s) \right)}.$$

De esta manera, se obtiene una aproximación del dato x_l en el espacio original a partir de las proyecciones dadas por KPCA. Los cálculos anteriores se hicieron para el kernel gaussiano sin embargo, como puede verse en Mika *et al.* (1999) [23] y Wang (2012) [33], se pueden hacer para otras funciones kernel.

2.5.1. Ejemplo

Se consideró un subconjunto de los datos MNIST[34]. Dichos datos son digitalizaciones de dígitos manuscritos los cuales fueron escaneados y estandarizados a un cuadro de 16×16 pixeles, de forma tal que cada imagen corresponde a un vector de longitud 256. Para este análisis se consideraron solamente las imágenes que corresponden a unos, cuatros u ochos obteniendo una matriz de datos X de dimensión 2199×256 y de rango 256. En la figura 5 se presenta una muestra de los datos que se utilizarán en este ejemplo.

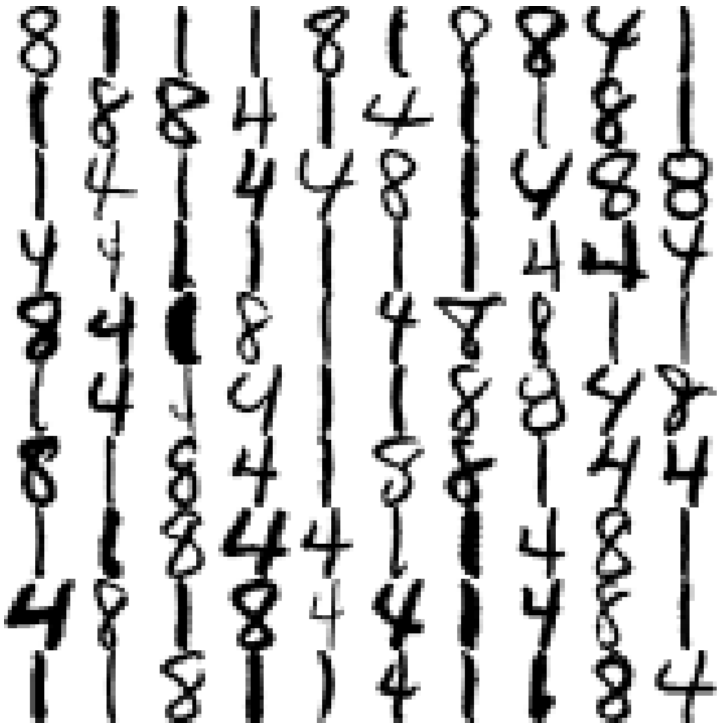

Figura 5: Muestra de los datos.

Como ya se mencionó, cada imagen está representada por un vector de tamaño 256. Las entradas de este vector, son números que se encuentran entre -1 y 1. Para mostrar la imagen se utiliza una escala de grises, por lo que el valor -1 corresponde al color blanco y el 1 al negro.

Se realizó KPCA a los datos utilizando un kernel gaussiano con parámetro $\sigma = 1$, $\sigma = 5$ y $\sigma = 10$ y se proyectó sobre un espacio de dimensión dos. Sería deseable obtener proyecciones que reflejen que en el conjunto de datos existe tres tipos diferentes o grupos diferentes de datos, los cuales corresponderían al dígito. En la Figura 6 se presentan las gráficas de las primeras dos coordenadas de las proyecciones obtenidas con KPCA.

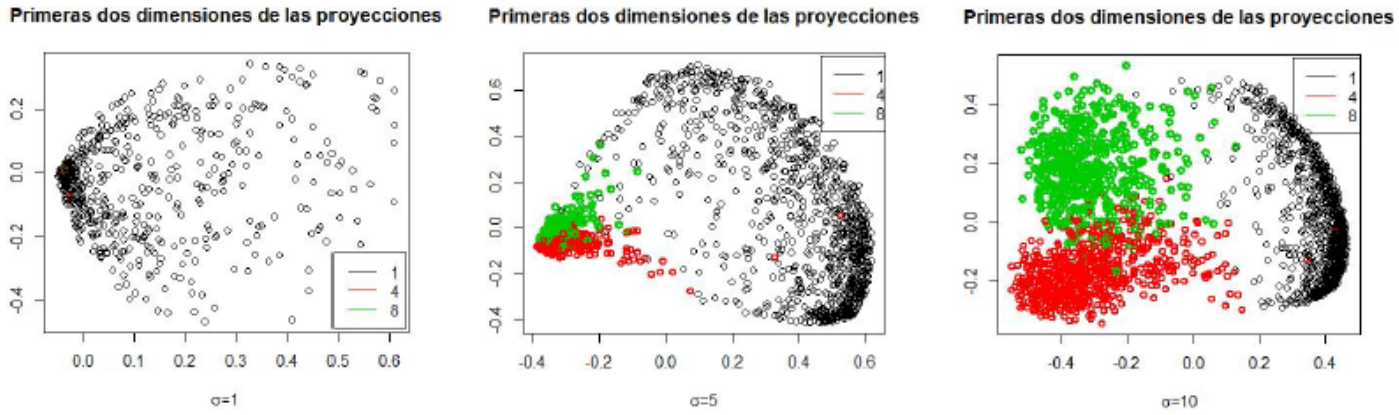

Figura 6: Primeras dos dimensiones de las proyecciones obtenidas mediante KPCA con un kernel gaussiano con $\sigma = 1$, 5 y 10.

Se puede observar que cuando se utiliza el parámetro $\sigma = 1$ no se separan los tres dígitos tan claramente como en el caso en que $\sigma = 10$. Para el caso en que el parámetro σ es 5 parece que existe mayor posibilidad de confusión entre dígitos que para cuando $\sigma = 10$. Sin embargo, debe recordarse que nada más se están observando las primeras dos dimensiones de la proyección por lo que no se puede concluir que $\sigma = 5$ no sea una buena elección. Debido a lo observado, puede decirse que el parámetro $\sigma = 1$ no es una buena elección para hacer KPCA a estos datos y que $\sigma = 5$ y $\sigma = 10$ parecen ser adecuados.

En este caso, los datos corresponden a imágenes por lo que sería útil obtener la reconstrucción de las proyecciones. Existen diversos parámetros que se deben elegir cuando se quiere reconstruir la proyección obtenida con KPCA con un kernel gaussiano en el espacio original. Tres parámetros que se considera que afectan bastante el desempeño del procedimiento de reconstrucción son: el parámetro σ del kernel gaussiano, el número de vectores sobre el que se proyectan las entradas de la matriz de Gram y el punto inicial que se elige para el método del punto fijo. A continuación se estudiará el efecto de dichos parámetros.

Efecto del parámetro σ del kernel gaussiano y del número de vectores que se utilizan para la proyección

En esta sección se presentan las preimágenes obtenidas cuando se lleva a cabo la reconstrucción de las proyecciones obtenidas con KPCA en el espacio original. Además, se presenta una comparación entre dichas preimágenes y las obtenidas a partir de PCA. Se realizó KPCA a los datos variando el valor del parámetro σ del kernel gaussiano y al realizar el cálculo de las preimágenes se varió el número de vectores de proyección tanto para PCA como para KPCA.

Se consideraron diferentes valores para el parámetro σ del kernel gaussiano, ya que no es nada claro como debe hacerse su elección. Los valores que se tomaron para el parámetro son: $\sigma = 1, 5, 10, 20, 50$. Debido a problemas numéricos no se eligió un parámetro con valores muy grandes.

Considerando que el rango de la matriz de datos X y por lo tanto de K es 256, se tomaron los siguientes valores para el número de vectores sobre el que se proyecta: $vec = 256, \frac{256}{2} = 128, \frac{256}{5} \approx 51, \frac{256}{10} \approx 26$ y $\frac{256}{20} \approx 13$. Se debe recordar que para el caso en que se obtiene la preimagen a partir de PCA y $vec = 256$, la preimagen coincide con la imagen original ya que no se está realizando ninguna reducción de dimensión.

Otro parámetro que influye en la preimagen obtenida es el punto inicial que se utiliza para el método del punto fijo. El efecto de este parámetro se estudiará después de manera burda, sin embargo debe tenerse en cuenta que una mala preimagen puede ser resultado de una mala elección del punto inicial. Para esta sección se utilizó como punto inicial una variable aleatoria normal multivariada generada con media igual a la media de los datos y matriz de covarianzas dada por uI, donde u se genera de una distribución uniforme en $(0, 1)$. Además se tomaron las entradas que corresponden a las esquinas $1, 16, 241$ y 256 igual a -1. Más detalles sobre esta elección se verá en la siguiente subsección.

A continuación se presentan algunas preimágenes obtenidas a partir de las proyecciones de PCA y KPCA y en base a los diferentes valores de σ y vec.

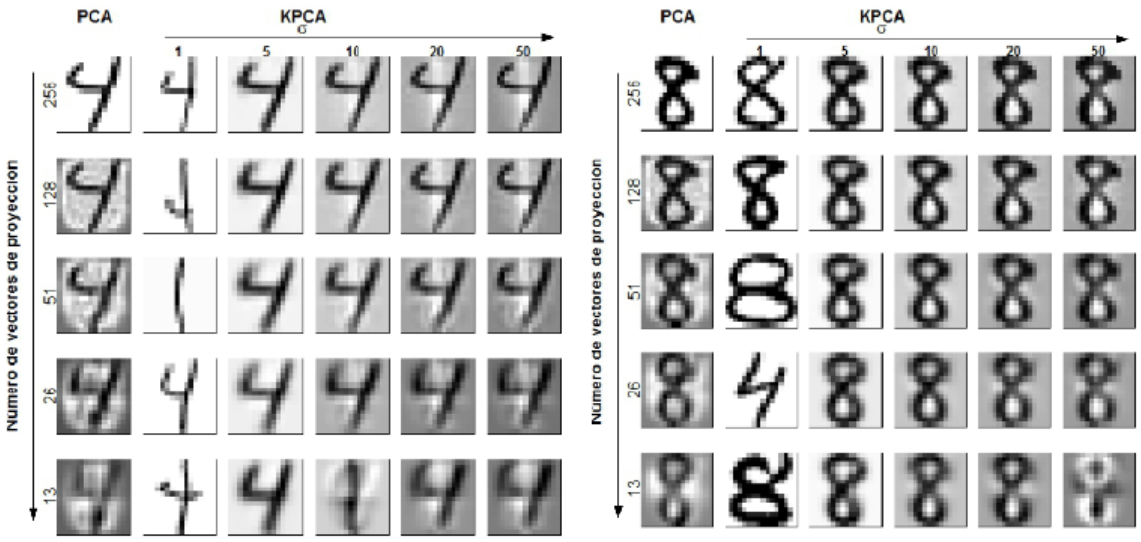

Figura 7: Preimágenes de los dígitos 4 y 8 obtenidas a partir de las proyecciones dadas por PCA y KPCA con diferentes valores para σ y vec.

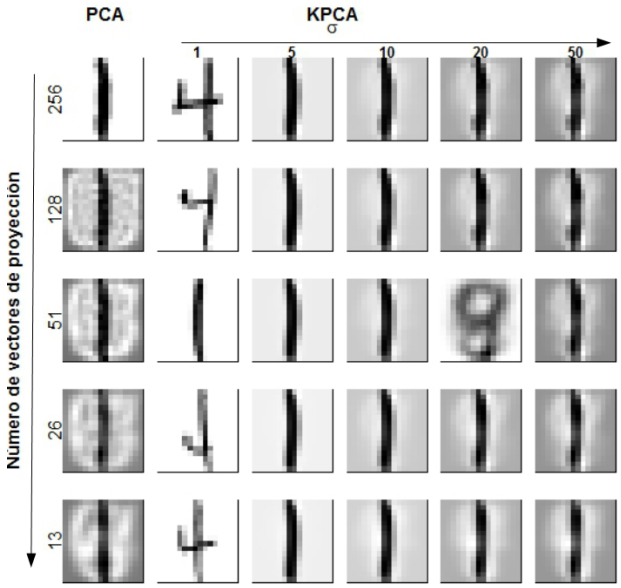

Figura 8: Preimágenes del dígito 1 obtenidas a partir de las proyecciones dadas por PCA y KPCA con diferentes valores para σ y vec.

En las Figuras 7 y 8 se puede observar que para $\sigma = 5$, $\sigma = 10$ y $\sigma = 20$ se obtienen resultados muy similares y buenos con respecto a la imagen original aún en el caso en que se disminuye el número de vectores de proyección a solamente 13. Cuando $\sigma = 50$ se nota un buen desempeño cuando se proyecta sobre muchos vectores. Sin embargo, en los últimos dos casos en los que se proyecta sobre pocos vectores cuesta un poco más distinguir el número. Con $\sigma = 1$ se tienen resultados malos ya que confunde muchos dígitos entre sí.

En cuanto al desempeño en base al número de vectores de proyección, se puede observar que el dígito se ve más borroso o con menor calidad conforme se disminuye este número. Este comportamiento era de esperarse ya que, lo que se hace con la proyección es limitarse a un subespacio del espacio generado por los datos transformados. Se puede observar que PCA es más sensible que KPCA (cuando se elige un parámetro adecuado) en cuanto a la disminución del número de vectores de proyección. Es más difícil distinguir el dígito cuando solamente se utilizan 13 ó 26 vectores de proyección para calcular la preimagen a partir de las proyecciones obtenidas con PCA.

El hecho de que se obtenga una preimagen con un dígito diferente al que se está aproximando puede deberse a que KPCA no recuperó bien la estructura del dato o al punto inicial dado al algoritmo del punto fijo (como sucedió cuando se obtuvo una preimagen con el número 8 en lugar de 1 con $\sigma = 20$ y $vec = 51$). No obstante, el hecho de que con $\sigma = 1$ confunda casi siempre los números es un indicio de que el valor para el parámetro no fue una buena elección, lo cual se observó anteriormente en la Figura 6.

Como referencia, se utilizó también la función sigest de R [27] que contiene un procedimiento

para estimar el parámetro σ óptimo y dió el valor $\sigma \approx 11$ para estos datos. Dicha función estima un valor para el parámetro que puede ser bueno cuando se desea utilizar una máquina de soporte vectorial para analizar los datos. Este valor parece ser una buena elección dado el desempeño que se observó para $\sigma = 10$.

Se realizó la reconstrucción de la proyección obtenida con KPCA para otros valores de σ y no se observó una tendencia o patrón claro con respecto al desempeño del método en base al parámetro. Sin embargo, para σ mucho menor que 5 se obtuvieron estimaciones muy malas. Es claro que el parámetro σ del kernel gaussiano influye bastante en la calidad de las estimaciones obtenidas; sería deseable tener una forma adecuada de elegirlo.

Efecto del punto inicial

Teóricamente, el punto inicial que se elige para el método del punto fijo, influye solamente en el tiempo que tarda el método en encontrar una solución z. No obstante, durante los diferentes experimentos realizados se observaron algunos comportamientos no deseables en el algoritmo. El primero, es que una mala elección del punto inicial combinado con un parámetro σ del kernel gaussiano muy grande, puede ocasionar problemas computacionales y de convergencia. Se observó que cuando se utilizó $\sigma = 20$ y $\sigma = 50$ el método tardaba más en converger y algunas veces producía errores computacionales del siguiente tipo. Considérese la expresión para z_{t+1} que se utiliza para el método del punto fijo:

$$z_{t+1} = \frac{\sum_{i=1}^{n} \gamma_i (x_l) \left(k(z_t, x_i) x_i - \frac{1}{n} \sum_{s=1}^{n} k(z_t, x_s) x_s \right)}{\underbrace{\sum_{i=1}^{n} \gamma_i (x_l) \left(k(z_t, x_i) - \frac{1}{n} \sum_{s=1}^{n} k(z_t, x_s) \right)}_{den}}.$$

Si el parámetro σ de $k(,)$ es muy grande entonces para todo i se tiene que $k(z_t, x_i) \approx 1$. Por lo anterior, el denominador de z_{t+1} se convierte aproximadamente en

$$den \approx \sum_{i=1}^{n} \gamma_i (x_l) \left(1 - \frac{1}{n} \sum_{s=1}^{n} 1 \right) = 0.$$

Es decir, no se obtiene una expresión para z_{t+1} con la que se pueda seguir iterando. Se decidió que al presentarse este caso, el método volviera a inicializarse con un punto diferente.

El otro problema, como se observó en la subsección anterior, es que para algunos dígitos el método converge a una preimagen que no corresponde al dígito. Debido a que este problema podría bien ser producido por KPCA o por el método del punto fijo, debe tenerse cautela con la utilización de ambos métodos. No obstante, se observó que en la mayoría de los casos, para algunas elecciones particulares del punto inicial se tuvo convergencia al dígito que se deseaba aproximar. Por esta razón, se infiere que el problema puede deberse a que el método del punto fijo busca los puntos en los cuales la derivada de la función que se quiere maximizar (o minimizar) es cero. Dichos puntos pueden ser máximos, mínimos o puntos silla locales. En este caso, no se sabe que la función que

se está maximizando, la cual depende del dato, solamente tenga un máximo. Así, dependiendo de la elección del punto inicial que se haga, el método puede converger al punto más cercano que sea máximo, mínimo o punto silla.

Al realizar los diferentes experimentos se observó que cuando el método convergía rápidamente (en menos de 30 iteraciones), convergía a una preimagen que sí correspondía al dígito que se estaba aproximando. En cambio, si se tardaba más en converger se obtenía una preimagen mala en el sentido en que no correspondía al dígito que se quería estimar.

Debido a dichos problemas, se decidió inicializar el método con un vector cuyas entradas correspondientes a las esquinas fueran -1. En cierta forma, se está introduciendo el hecho de que en las esquinas no se encuentra el dígito. Además, se restringió a que el número máximo de iteraciones fuera 30 y en caso de que aún no convergiera, se volviera a iniciar el método en un punto diferente. Se obtuvieron resultados muy buenos introduciendo las condiciones anteriores, aunque para algunos dígitos aún se obtuvo convergencia a una mala preimagen. Para evitar este problema, podría introducirse un parámetro de regularización a la función que se desea maximizar, de tal forma que la función fuese convexa. Sin embargo, dicha idea no se implementó debido a que el método del punto fijo no es uno de los objetivos principales de estudio de esta tesis.

3. Métodos aleatorizados para aproximar matrices de Gram mediante otras de menor rango

3.1. Introducción

Como ya se había mencionado, esta tesis se enfoca en datos con estructura no lineal y con gran número de observaciones. Se estudiaron los métodos Kernel como una herramienta para mapear los datos a espacios donde tengan estructura aproximadamente lineal y en lugar de trabajar con la matriz de datos, se trabaje con la matriz K. Como se vió en el capítulo anterior, la matriz K es de dimensión $n \times n$ por lo que el costo computacional de trabajar con esta matriz puede ser muy grande. Por lo anterior, la motivación de este capítulo es aprender algunos métodos aleatorizados para aproximar la matriz K mediante una matriz de menor rango.

Los métodos aleatorizados en álgebra lineal se remontan al teorema de Johnson Lindenstrauss. Este teorema demuestra que un conjunto de puntos en un espacio de suficientemente alta dimensión puede ser mapeado a otro de menor dimensión preservando casi completamente las distancias entre los puntos, utilizando proyecciones aleatorias. Una demostración sencilla de este teorema puede verse en Dasgupta *et al.* (2003) [8]. Así, la idea de los métodos aleatorizados es proyectar los datos a espacios en donde se mantenga la mayor información posible.

En general, la ventaja más clara de utilizar los métodos aleatorizados es que se obtienen algoritmos más rápidos. Sin embargo, como lo menciona Mahoney (2012) [21], también se tienen otras ventajas entre las que se puede resaltar: obtener algoritmos que sean fáciles de analizar, cuyos resultados se puedan interpretar de manera más fácil y que permiten aprovechar técnicas computacionales modernas, por ejemplo el cálculo en paralelo.

En esta tesis, se utilizan los métodos aleatorizados para aproximar la matriz K. Por ejemplo, algunos métodos aproximan K proyectándola sobre un subespacio generado aleatoriamente, a partir de diferentes distribuciones de probabilidad. En este capítulo se mostraran tres métodos de proyección aleatoria para aproximación de matrices: el método de columnas, el método de Nyström y el método Random Fourier Features (RFF). Además, se presenta una modificación al método RFF a la que se le llamó RFF PCA.

Como se vió en el primer capítulo, se puede aproximar una matriz utilizando diferentes descomposiciones de ésta. En este capítulo, se presentan las aproximaciones dadas por las diferentes descomposiciones de K_k obtenidas mediante los métodos aleatorizados antes mencionados. En particular se tiene el objetivo de reescribir alguna de las aproximaciones en la forma:

$$\widehat{K} = \widehat{Z}\widehat{Z}^t, \tag{24}$$

donde $\widehat{Z}$ es una matriz de rango k y de dimensión $n \times k$, con $k < n$. En lo que sigue, a esta aproximación se le llamará *aproximación objetivo* de K.

El interés en la aproximación objetivo radica en que si se cumple (24) entonces KPCA puede ser aproximado haciendo PCA sobre $\widehat{Z}$. Como se vió en el capítulo anterior, para calcular las proyecciones que se obtienen con KPCA basta con calcular los eigenvalores y eigenvectores de K. Si se hace PCA sobre $\widehat{Z}$, las componentes principales están dados por los eigenvalores y eigenvectores de la matriz de covarianzas estimada $\widehat{Z}^t\widehat{Z}$ y utilizando (15) se obtienen los eigenvectores de $\widehat{Z}\widehat{Z}^t$ que

es K. La ventaja de hacer PCA sobre $\widehat{Z}$ para aproximar el resultado de KPCA, es que en lugar de trabajar con la matriz K de dimensión $n \times n$ se trabajará con la matriz $\widehat{Z}^t\widehat{Z}$ de (menor) dimensión $k \times k$.

3.2. Método de columnas

3.2.1. Idea detrás del método

El método de columnas fue introducido en 1998 por Frieze et al. [13]. La idea detrás del método es quizás la más natural de los tres métodos que se presentarán: si no se puede trabajar con un conjunto grande de datos, muestrear una parte de ellos y analizarlos para obtener información del conjunto total. Así, el método de columnas utiliza una muestra o subconjunto de K para obtener estimadores tanto de la matriz como de sus eigenvectores y eigenvalores. El nombre método de columnas proviene del hecho que se utiliza la descomposición en valores singulares de la matriz de columnas muestreadas C para aproximar K.

La pregunta obvia es cómo se debe elegir las columnas de manera que representen bien al conjunto total de datos. El problema de cómo hacer esta elección es conocido en la literatura como "column selection subset problem" o CSSP [4] [3] [10] [13]. Diferentes soluciones o algoritmos para resolver este problema se han propuesto [4] [3] [10] [11] [13].

Esta tesis se enfocará en el método utilizado por Talwallkar *et al.* (2013) [18] por ser un método de aproximación bueno y sencillo. A dicho método, el cual se presenta a continuación, se le llamará simplemente método de columnas.

3.2.2. Algoritmo para calcular $\widehat{K}^{Col}$

En esta sección se presentan los pasos para calcular las diferentes aproximaciones de rango k para la matriz de Gram K utilizando el método de columnas. En la sección 3.2.4 se dará una justificación de este algoritmo. El procedimiento es:

1. Muestrear aleatoriamente con una distribución uniforme y sin reemplazo $l \leq n$ columnas (c_1, c_2,..., c_l) de K .

2. Construir la matriz C
$$C\left[, j\right] = K\left[, c_j\right] \qquad j \in (1, 2, ..., l)\,.$$

3. Las aproximaciones de rango k ($k \leq n$) de la matriz de Gram K están dadas por
$$\widehat{K}_k^{Col,\ esp} = \frac{\sqrt{n}}{l} C \left(\left(C^t C \right)_k^{\frac{1}{2}} \right)^+ C^t \tag{25}$$
y
$$\widehat{K}_k^{Col,proy} = C \left(\left(C^t C \right)_k \right)^+ C^t K. \tag{26}$$

3.2.3. Fórmulas para $\widehat{Z}^{Col}$ y los eigenvectores y eigenvalores estimados de K

La expresión (25) corresponde a la aproximación para K de rango k por descomposición espectral y (26) corresponde a la aproximación para K de rango k por descomposición en matriz proyección. En esta sección, se reescribirá (25) de tal manera que se encuentre una expresión para la aproximación objetivo (24). Además, se presentarán fórmulas cerradas que provee el método de columnas para estimar los eigenvectores y eigenvalores de la matriz de Gram.

Utilizando la descomposición en valores singulares de C, se tiene que

$$\begin{aligned} C^tC &= \left[V^CD^C\left(U^C\right)^t\right]\left[U^CD^C\left(V^C\right)^t\right] \\ &= V^CD^CD^C\left(V^C\right)^t \qquad \text{(pues } U^C \text{ es matriz ortonormal)} \\ &= V^C\left(D^C\right)^2\left(V^C\right)^t \end{aligned} \tag{27}$$

y

$$\begin{aligned} \left[V^CD^C\left(V^C\right)^t\right]\left[V^CD^C\left(V^C\right)^t\right] &= V^CD^CD^C\left(V^C\right)^t \quad \text{(pues } V^C \text{ es matriz ortonormal)} \\ &= C^tC \qquad \text{(por (27)).} \end{aligned}$$

De lo anterior se obtiene que:

$$\left(C^tC\right)^{\frac{1}{2}} = \left[V^CD^C\left(V^C\right)^t\right]$$

y su aproximación de rango k es

$$\left(C^tC\right)_k^{\frac{1}{2}} = \left[V_k^CD_k^C\left(V_k^C\right)^t\right].$$

Reemplazando en (25) se tiene que:

$$\begin{aligned} \widehat{K}_k^{Col,\,esp} &= \frac{\sqrt{n}}{l}C\left(\left(C^tC\right)_k^{\frac{1}{2}}\right)^+C^t \\ &= \frac{\sqrt{n}}{l}C\left(V_k^C\left(D_k^C\right)^+\left(V_k^C\right)^t\right)C^t \\ &= \frac{\sqrt{n}}{l}CV_k^C\left[\left(D_k^C\right)^{\frac{1}{2}}\right]^+\left[\left(D_k^C\right)^{\frac{1}{2}}\right]^+\left(V_k^C\right)^tC^t \\ &= \frac{\sqrt{n}}{l}\left\{CV_k^C\left[\left(D_k^C\right)^{\frac{1}{2}}\right]^+\right\}\left\{CV_k^C\left[\left(D_k^C\right)^{\frac{1}{2}}\right]^+\right\}^t \\ &= \left\{\underbrace{\frac{\sqrt[4]{n}}{\sqrt{l}}CV_k^C\left[\left(D_k^C\right)^{\frac{1}{2}}\right]^+}_{\widehat{Z}^{Col}}\right\}\left\{\underbrace{\frac{\sqrt[4]{n}}{\sqrt{l}}CV_k^C\left[\left(D_k^C\right)^{\frac{1}{2}}\right]^+}_{\widehat{Z}^{Col}}\right\}^t \\ &= \widehat{Z}^{Col}\left(\widehat{Z}^{Col}\right)^t. \end{aligned}$$

Por lo tanto, la matriz de interés $\widehat{Z}^{Col}$ está dada por:

$$\widehat{Z}^{Col} = \frac{\sqrt[4]{n}}{\sqrt{l}} C V_k^C \left[\left(D_k^C \right)^{\frac{1}{2}} \right]^+ .$$

Como puede verse en Talwalkar *et al.* (2013) [18], los estimadores de los eigenvectores y eigenvalores de K dados por el método de columnas son:

$$\widehat{\Sigma}^{Col} = \sqrt{\frac{n}{l}} D_k^C \qquad \text{y} \qquad \widehat{U}^{Col} = U^C = C V_k^C \left(D_k^C \right)^+ .$$

El hecho de contar con fórmulas cerradas para los eigenvectores aproximados $\widehat{U}^{Col}$ es una ventaja computacional en el caso de Kernel PCA ya que, como se verá más adelante, cuesta menos calcular $\widehat{U}^{Col}$ que hacer PCA sobre $\widehat{Z}^{Col}$.

3.2.4. ¿Cómo surge el método de columnas?

En esta subsección se mostrará que la aproximación de K por descomposición en matriz proyección $\widehat{K}^{Col,proy}$ dada por el método de columnas, es la matriz que minimiza el error de aproximación $\left\| K - \widehat{K} \right\|_F$, sujeto a que $\widehat{K}$ sea de un cierto rango [11]. Es decir, se verá que el método de columnas surge como la solución a un problema de regresión lineal multivariado.

Supóngase que se selecciona un subconjunto de columnas de K el cual se representa matricialmente mediante C. Una idea para aproximar la matriz de Gram es suponer que las entradas de la matriz K pueden aproximarse mediante combinaciones lineales de las columnas de C. Es decir, suponer que el subconjunto de columnas escogidas genera todo el espacio en el que viven las columnas de K. Así,

$$\widehat{K} = CT, \tag{28}$$

donde T de dimensión $c \times n$ es la matriz de coeficientes de las combinaciones lineales.

El objetivo es minimizar el error de aproximación $\left\| K - \widehat{K} \right\|_F$ por lo que se deben escoger los coeficientes de tal forma que

$$T^* = \arg\min_T \left\| K - CT \right\|_F .$$

Lo anterior corresponde al problema de regresion lineal multivariado cuya solución si $(C^tC)^{-1}$ existe, es

$$T^* = \left(C^t C \right)^{-1} C^t K. \tag{29}$$

Sustituyendo (29) en (28) se tiene

$$\widehat{K} = C \left(C^t C \right)^{-1} C^t K.$$

Si la inversa $(C^tC)^{-1}$ no existe, se obtiene una solución con la inversa generalizada:

$$\begin{aligned} \widehat{K} &= C \left(C^t C \right)^+ C^t K \\ &= P^{(C)} K \\ &= \widehat{K}^{\text{Col,proy}} \quad \text{(por (26))}. \end{aligned}$$

Es decir, el método de columnas es un método que se obtiene de aproximar K a través de mínimos cuadrados basado en las columnas seleccionadas de K.

Calculando los vectores y valores propios de $\widehat{K}^{Col,proy}$ y sustituyéndolos en (6) se obtiene $\widehat{K}_k^{Col,\ esp}$ (25).

3.2.5. El método de columnas como proyección aleatoria

La aproximación de K por descomposición en matriz proyección obtenida mediante el método de columnas tiene la estructura:

$$\widehat{K}^{\text{Col,proy}} = P^{(C)}K.$$

Lo anterior muestra que la estimación de la matriz de Gram se obtiene proyectando las entradas de la matriz K sobre el subespacio generado por las columnas de C. Cada columna de la matriz C es escogida de manera aleatoria con probabilidad uniforme y sin reemplazo. Es decir, la matriz C puede considerarse una matriz aleatoria y por lo tanto $\widehat{K}$ se puede considerar como una proyección aleatoria.

3.2.6. Cotas para el error de aproximación

A continuación se presenta una cota en probabilidad para el error de aproximación de $\widehat{K}^{\text{Col,proy}}$. Cabe resaltar que en la literatura no se encontró fácilmente cotas para el método de columnas cuando se utiliza un muestreo aleatorio simple sin reemplazo para ninguna de las dos aproximaciones (25) y (26). Por lo anterior, se calculó una cota no muy estricta solamente para obtener una idea del comportamiento del método.

Para el caso en que se utiliza un muestreo aleatorio simple con reemplazo (o basado en algunas otras distribuciones), sí hay cotas para el error de aproximación, las cuales pueden verse en Frieze (1998) [13] y Mahoney (2012) [21]. Resulta bastante sorprendente que no se encuentren cotas para el método de columnas aunque sea el método más sencillo. Como se verá después, la razón puede deberse al hecho que su desempeño en aproximar la matriz K no es tan bueno como el del método de Nyström.

La cota en probabilidad que se presenta a continuación se obtuvo utilizando resultados de la literatura para el método de Nyström y propiedades de las normas vectoriales. Probablemente esta cota no es muy estricta pero se obtuvo para tener una idea de cómo es el error de aproximación.

Se encontrará el error de aproximación cuando se utiliza la aproximación por descomposición en matriz proyección $\widehat{K}^{Col,proy}$. Es decir, se quiere una cota para el error:

$$\left\| K - \widehat{K}^{Col,proy} \right\|_2^2 = \left\| K - P^{(C)}K \right\|_2^2.$$

Se tiene que

$$\begin{aligned}
\left\|K - P^{(C)}K\right\|_2^2 &= \left\|\left(I - P^{(C)}\right)K\right\|_2^2 \\
&= \left\|\left(I - P^{(C)}\right)KK^t\left(I - P^{(C)}\right)\right\|_2 \\
&\leq \left\|\left(I - P^{(C)}\right)\left(KK^t - CC^t\right)\left(I - P^{(C)}\right)\right\|_2 + \qquad (30)\\
&\quad + \left\|\left(I - P^{(C)}\right)CC^t\left(I - P^{(C)}\right)\right\|_2 \qquad \text{(Des. triángulo)} \qquad (31)\\
&= \left\|\left(I - P^{(C)}\right)\left(KK^t - CC^t\right)\left(I - P^{(C)}\right)\right\|_2 + 0 \quad (P^{(C)} \text{ matriz de proy.}) \quad (32)\\
&\leq \left\|\left(I - P^{(C)}\right)\right\|_2 \left\|KK^t - CC^t\right\|_2 \left\|\left(I - P^{(C)}\right)\right\|_2 \text{(Norma submultiplicativa)} (33)\\
&\leq \left\|KK^t - CC^t\right\|_2 \qquad \left(\left\|\left(I - P^{(C)}\right)\right\|_2 = 1\right) \quad (34)\\
&\leq \left\|KK^t - CC^t\right\|_F . \qquad (35)
\end{aligned}$$

Así, la idea es utilizar resultados que se tienen para $\left\|KK^t - CC^t\right\|_F$ para encontrar una cota para $\left\|K - P^{(C)}K\right\|_2^2$. Un poco más acerca de cómo se obtiene la cota para $\left\|KK^t - CC^t\right\|_F$ se presenta en la subsección correspondiente a la cota para el método de Nyström. Sin embargo, la idea es acotar la variable $\left\|KK^t - CC^t\right\|_F$ utilizando una desigualdad de concentración de medida (Teorema de McDiarmid) para la esperanza $E\left(\left\|KK^t - CC^t\right\|_F\right)$. De manera general, la concentración de medida con respecto a la media se refiere a que el promedio de una función de variables aleatorias independientes no se aleja mucho de su media.

Utilizando el corolario 2 del artículo de Kumar *et al.* (2009) [19] y haciendo la aproximación $\alpha(l, n-l) \approx l\left(1 - \frac{l}{n}\right)$, se tiene que si $\delta \in (0,1)$ y $\eta = \sqrt{\log(2/\delta)\left(1 - \frac{l}{n}\right)}$ entonces

$$P\left(\left\|KK^t - CC^t\right\|_F \leq (1+\eta)\frac{n}{\sqrt{l}}\left(\max_j \|K[,j]\|\right)^2\right) \geq 1 - \delta. \qquad (36)$$

Combinando (34) y (36) se tiene que si $\delta \in (0,1)$ y $\eta = \sqrt{\log(2/\delta)\left(1 - \frac{l}{n}\right)}$ entonces

$$P\left(\left\|K - P^{(C)}K\right\|_2^2 \leq (1+\eta)\frac{n}{\sqrt{l}}\left(\max_j \|K[,j]\|\right)^2\right) \geq 1 - \delta.$$

Se puede observar que conforme n crece la cota se hace más grande, lo cual es de esperarse puesto que se tiene más error cuando la matriz es más grande. Si consideramos n fija y l crece, la cota se hace más pequeña lo cual nos dice que entre mayor tamaño de muestra se tome el error de reducirá. Sin embargo, la mejora o la reducción que se hace irá siendo cada vez menor lo cual queda muy claro en las gráficas de desempeño del método que se mostraran en el siguiente capítulo. En dichas gráficas se observará que la aproximación mejora conforme crece el porcentaje de columnas muestreado, sin embargo la diferencia entre la mejora de porcentajes altos es menor que la diferencia en la mejora entre porcentajes bajos.

3.3. Método de Nyström

3.3.1. Idea detrás del método

El método de Nyström fue propuesto en 1928 por Nyström [26] como un método para integrar numéricamente. En el año 2000, Williams y Seeger [31] lo presentaron como una solución para optimizar el tiempo de algoritmos kernel. Este método muestrea de forma aleatoria columnas de la matriz kernel $K_{n\times n}$ para producir una estimación de dicha matriz, de sus eigenvalores y de sus eigenvectores.

Sin detallar mucho, podría decirse que el método de Nyström aproxima K reemplazando sus vectores y valores propios por estimaciones en su descomposición espectral (6). Lo que caracteriza al método de Nyström es la estructura de $\widehat{K}$ y la forma en que se estiman los eigenvectores. Como se verá después, a diferencia de otros métodos, la aproximación para K_k dada por la descomposición espectral (6) coincide con la aproximación para K_k dada por la descomposición en vectores propios (8) . Es decir,

$$\widehat{K}_k^{Nys,\ esp} = \widehat{K}_k^{Nys,\ vp} \tag{37}$$

$$= A\widehat{V_k^A}\left(A\widehat{V_k^A}\right)^t. \tag{38}$$

Se debe recalcar que no basta estimar los eigenvectores y crear una matriz $\widehat{K}$ con la estructura (38) ya que se desconoce A. Si A fuese conocida ya se tendría la factorización deseada para K. El método de Nyström resulta bastante interesante con respecto a la estructura de $\widehat{K}$, ya que obtiene una aproximación de la forma (38) sin que sea necesario conocer explícitamente V_k ni A.

3.3.2. Algoritmo para obtener $\widehat{K}$

En esta subsección se mostraran los pasos para calcular las estimaciones de K obtenidas mediante el método de Nyström. Se muestran las expresiones para la aproximaciones de K_k por descomposición espectral y en matriz proyección.

El procedimiento es:

1. Muestrear aleatoriamente con una distribución uniforme y sin reemplazo $l \leq n$ columnas de K. Se denotarán las columnas muestreadas por c_1, c_2,..., c_l.

2. Construir las matrices C y W definidas por:

$$C\,[,j] = K\,[,c_j] \qquad j \in (1,2,...,l)$$

y

$$W[i,j] = K\,[c_i,c_j] \qquad i,\ j \in (1,2,...,l)\,.$$

Es decir, las columnas de C están conformadas por las l columnas muestreadas de K y W consiste de la intersección de dichas columnas con los correspondientes renglones de K.

3. Las aproximaciones de rango k ($k \leq n$) de la matriz de Gram K están dadas por

$$\widehat{K}_k^{Nys,\ esp} = CW_k^+ C^t \tag{39}$$

y

$$\widehat{K}_k^{Nys,\ proy} = \frac{l}{n} C \left(W_k^2\right)^+ C^t K. \tag{40}$$

3.3.3. Fórmulas para Z^{Nys}, los eigenvectores y eigenvalores estimados

En esta subsección se reescribirá (39) de tal forma que se encuentre una expresión para la aproximación objetivo (24). Además, se presentarán fórmulas cerradas para los eigenvectores y eigenvalores estimados obtenidos mediante el método de Nyström.

Se utilizará el SVD de W para encontrar la matriz Z^{Nys} de la aproximación objetivo (24). Como W es simétrica, por lo visto en el Capítulo 1, la mejor aproximación de rango k de W es:

$$W_k = U_k^W D_k^W \left(U_k^W\right)^t.$$

Reemplazando lo anterior en (39) se tiene que

$$\begin{aligned}
\widehat{K}_k^{Nys,\ esp} &= CW_k^+ C^t \qquad (41)\\
&= CU_k^W \left(D_k^W\right)^+ \left(U_k^W\right)^t C^t \\
&= \left\{ CU_k^W \left[\left(D_k^W\right)^{\frac{1}{2}}\right]^+ \right\} \left[\left[\left(D_k^W\right)^{\frac{1}{2}}\right]^+ \left(U_k^W\right)^t C^t \right] \\
&= \left\{ \underbrace{CU_k^W \left[\left(D_k^W\right)^{\frac{1}{2}}\right]^+}_{\widehat{Z}^{Nys}} \right\} \left\{ \underbrace{CU_k^W \left[\left(D_k^W\right)^{\frac{1}{2}}\right]^+}_{\widehat{Z}^{Nys}} \right\}^t \\
&= \widehat{Z}^{Nys} \left(\widehat{Z}^{Nys}\right)^t. \qquad (42)
\end{aligned}$$

Por lo tanto, la matriz de interés $\widehat{Z}^{Nys}$ está dada por

$$\widehat{Z}^{Nys} = CU_k^W \left[\left(D_k^W\right)^{\frac{1}{2}}\right]^+.$$

Reordenando la expresión (41) se tiene que la matriz $\widehat{K}$ también puede escribirse como

$$\widehat{K}_k^{Nys,\ esp} = \left[\sqrt{\frac{l}{n}} CU_k^W \left(D_k^W\right)^+\right] \left[\frac{n}{l} D_k^W\right] \left[\sqrt{\frac{l}{n}} CU_k^W \left(D_k^W\right)^+\right]^t$$

y por lo tanto, los eigenvalores y eigenvectores estimados están dados por

$$\widehat{\Sigma}^{Nys} = \frac{n}{l} D_k^W \qquad \text{y} \qquad \widehat{U}^{Nys} = \left(\sqrt{\frac{l}{n}} CU_k^W \left(D_k^W\right)^+\right).$$

3.3.4. ¿Cómo surge el método de Nyström?

El método de Nyström surge originalmente como un método de integración numérica [26], es decir un método que sirve para aproximar una ecuación integral mediante una suma. Este método discretiza un problema continuo y encuentra solución para un cierto número de puntos contenidos en el dominio de la integral. Después de encontrar dicha solución, interpola el resultado para los demás puntos. Esta subsección se divide en dos partes. En la primera, se explica la forma en que Seeger y Williams (2000) [31] utilizaron el método de Nyström para aproximar los eigenvectores de K. En la segunda, se muestra cómo a partir de las aproximaciones obtenidas en la primera parte se llega a (39).

3.3.4.1. Aproximación de los eigenvectores de K

A continuación se presentará la idea de Seeger y Williams (2000) [31] de utilizar el método de Nyström para encontrar una aproximación de los eigenvectores de K.

El método de Nyström de integración numérica sirve para encontrar la aproximación numérica a la solución de problemas de eigenfunciones de la forma:

$$\int_a^b k(x,y) f_r(y)\, dy = \lambda_r f_r(x), \qquad r \in \{1,2,...,k\}. \tag{43}$$

donde k denota un kernel, $f(\cdot)$ una función y λ un número real.

Para obtener una estimación $\widehat{f_r}$ de la eigenfunción f_r se aproxima la integral anterior mediante una suma sobre l puntos equidistantes en el intervalo $[a,b]$, es decir:

$$\int_a^b k(x,y) f_r(y)\, dy \approx \frac{(b-a)}{l} \sum_{j=1}^{l} k(x,y_j) f_r(y_j).$$

Se debe recalcar que los puntos se toman equidistantes para ejemplificar el procedimiento sin embargo, dicha suposición no es necesaria. De manera general, se pueden tomar puntos no equidistantes en el intervalo $[a,b]$ y se introduce una función de pesos, que en el caso equidistante es $\frac{1}{l}$. El problema (43) conduce al problema:

$$\frac{(b-a)}{l} \sum_{j=1}^{l} k(x,y) f_r(y_j) = \lambda_r f_r(x). \tag{44}$$

Para resolverlo, se toma $x = y_i$, lo cual da el sistema de ecuaciones:

$$\frac{(b-a)}{l} \sum_{j=1}^{l} k(y_i,y_j) f_r(y_j) = \lambda_r f(y_i) \qquad \forall i \in \{1,...,l\}.$$

En particular, si se consideran los l puntos como los l números de columnas muestreadas, $k(y_i,y_j)$ de la integral anterior corresponde a una parte de la matriz K cuyos elementos son las intersecciones de los renglones correspondientes a las columnas muestreadas y_j. Lo anterior se definió como W. Así, el sistema queda de la forma:

$$\frac{n}{l}\sum_{j=1}^{l} W[i,j]\, f_r(y_j) = \lambda_r f_r(y_i) \qquad \forall i \in \{1,...,l\}.$$

De forma matricial esto puede escribirse como:

$$W f_r \propto \lambda_r f_r.$$

Es decir, se tiene un problema de eigenvalores: λ_r es el eigenvalor correspondiente al vector propio f_r de la matriz W. Al resolver este problema, se encuentran valores para λ_r y $f_r(\cdot)$. Se debe notar que $f_r(y_j)$ corresponde a la j-ésima entrada del r-ésimo vector propio de W.

Sustituyendo $f_r(\cdot)$ en la ecuación (44) se obtiene

$$\begin{aligned} \widehat{f_r}(x) &= \frac{n}{\lambda_r l}\sum_{j=1}^{l} k(x,y_j)\, f_r(y_j) \\ &\propto \frac{1}{\lambda_r}\sum_{j=1}^{l} k(x,y_j)\, f_r(y_j). \end{aligned} \tag{45}$$

La expresión (45) permite estimar $f_r(\cdot)$ en valores diferentes a los $y_j's$. La entrada j-ésima del vector propio estimado correspondiente al renglón x de la matriz K que no formó parte de la muestra, se obtiene de una combinación lineal de los eigenvectores de W. Lo que se debe calcular para cada x que no perteneció a la muestra es $k(x,y_j)$ con y_j que sí perteneció a la muestra.

Así, lo que se hace para encontrar una aproximación de los eigenvectores de K es calcular los eigenvectores de W y luego interpolar el resultado para todas las entradas de K. El resultado que se obtiene es que una parte del eigenvector aproximado de K es el eigenvector para la matriz de entradas muestreadas y la otra es una proyección sobre el espacio generado por las entradas muestreadas.

3.3.4.2. Aproximación de K

Antes de mostrar la forma en que se obtiene (39), se reescribirán K y C de manera conveniente. Sin pérdida de generalidad, se pueden reordenar las entradas de la matriz K de la siguiente manera:

$$K = \begin{bmatrix} W & K_{21} \\ K_{21}^t & K_{22} \end{bmatrix}, \tag{46}$$

donde W de dimensión $l \times l$ es como se definió anteriormente, K_{21}^t es la matriz de dimensión $(n-l)\times l$ formada por la función kernel entre las entradas muestreadas y todas las demás, y K_{22} es la matriz de dimensión $(n-l)\times(n-l)$ formada por la función kernel entre todos las entradas que no fueron muestreadas. Si se escribe K de este manera, por la definición de C se tiene que:

$$C = \begin{bmatrix} W \\ K_{21}^t \end{bmatrix}.$$

Se definen los eigenvalores y eigenvectores estimados de K dados por el método de Nyström como

$$\widehat{D} = D^*. \tag{47}$$

y

$$\widehat{U} = \begin{bmatrix} U^* \\ K_{21}U^* \left(D^*\right)^{-1} \end{bmatrix}, \tag{48}$$

donde las columnas de U^* son los eigenvectores de W y $K_{21}U^* \left(D^*\right)^{-1}$ corresponde a la interpolación de estos eigenvectores para los puntos que no pertenecen a la muestra. Dicha interpolación se obtiene utilizando la ecuación (45), K_{21} corresponde a la función kernel entre los puntos muestreados y los restantes, U^* corresponde a la parte de los eigenvectores $f_r\left(y_j\right)$ de W y $\left(D^*\right)^{-1}$ corresponde al $\frac{1}{\lambda_r}$ de la ecuación. Observando $\widehat{U}$ en la forma (48) es claro que los eigenvectores estimados no son necesariamente ortonormales entre sí, ya que algunas entradas son combinaciones lineales de los eigenvectores de W.

Reemplazando los vectores y valores propios de K en su descomposición espectral por los estimadores (47) y (48) se puede encontrar una aproximación $\widehat{K}$. A continuación se mostrará que $\widehat{K}$ corresponde a (39).

$$\begin{aligned}
\widehat{K} &= \widehat{U}\widehat{D}\widehat{U}^t \\
&= \begin{bmatrix} U^* \\ K_{21}^t U^* \left(\widehat{D}\right)^{-1} \end{bmatrix} \widehat{D} \begin{bmatrix} \left(U^*\right)^t & \left(\widehat{D}\right)^{-1} \left(U^*\right)^t K_{21} \end{bmatrix} \\
&= \begin{bmatrix} U^*\widehat{D}\left(U^*\right)^t & U^*\widehat{D}\widehat{D}^{-1}\left(U^*\right)^t K_{21} \\ K_{21}^t U^* \widehat{D}^{-1}\widehat{D}\left(U^*\right)^t & K_{21}^t U^* \widehat{D}^{-1}\widehat{D}\widehat{D}^{-1}\left(U^*\right)^t K_{21} \end{bmatrix} \\
&= \begin{bmatrix} W & K_{21} \\ K_{21}^t & K_{21}^t W^{-1} K_{21} \end{bmatrix} \\
&= \begin{bmatrix} W \\ K_{21}^t \end{bmatrix} W^{-1} \begin{bmatrix} W & K_{21} \end{bmatrix} \\
&= CW^{-1}C^t \\
&= \widehat{K}^{Nys,\ esp}.
\end{aligned} \tag{49}$$

De los cálculos anteriores, es de gran importancia tener presente la expresión (49) ya que es muy útil cuando se desea calcular el error de aproximación.

3.3.5. El método de Nyström como proyección aleatoria

En esta subsección se demostrará que (39) coincide con (37) y por lo tanto tiene la misma interpretación. Es decir, se mostrará que:

$$\begin{aligned}
\widehat{K}^{Nys,\ esp} &= \widehat{K}^{Nys,\ vp} \\
&= \left(A\widehat{V}_k\right)\left(A\widehat{V}_k\right)^t.
\end{aligned}$$

Como se había mencionado, por ser K semidefinida positiva se tiene que $K = AA^t$. Sea $A^t = \left[\begin{array}{cc} A_1^t & A_2^t \end{array}\right]$. Así,

$$\begin{aligned} K &= AA^t \\ &= \left[\begin{array}{c} A_1 \\ A_2 \end{array}\right] \left[\begin{array}{cc} A_1^t & A_2^t \end{array}\right] \\ &= \left[\begin{array}{cc} A_1A_1^t & A_1A_2^t \\ A_2A_1^t & A_2A_2^t \end{array}\right] \\ &= \left[\begin{array}{cc} W & K_{21} \\ K_{21}^t & K_{21}^tW^{-1}K_{21} \end{array}\right] \quad \text{(por (49))}. \end{aligned}$$

De lo anterior, $W = A_1A_1^t$ y $C = \left[\begin{array}{c} A_1A_1^t \\ A_2A_1^t \end{array}\right]$. Reemplazando lo anterior en la expresión para $\widehat{K}^{Nys,\,esp}$ se tiene

$$\begin{aligned} \widehat{K}^{Nys,\,esp} &= CW^{-1}C^t \\ &= \left[\begin{array}{c} A_1A_1^t \\ A_2A_1^t \end{array}\right] \left(A_1A_1^t\right)^{-1} \left[\begin{array}{cc} A_1A_1^t & A_1A_2^t \end{array}\right] \\ &= \left[\begin{array}{c} A_1 \\ A_2 \end{array}\right] A_1^t \left(A_1A_1^t\right)^{-1} A_1 \left[\begin{array}{cc} A_1^t & A_2^t \end{array}\right] \\ &= AA_1^t \left(U^{A_1} D^{A_1} \left(V^{A_1}\right)^t \left(V^{A_1}\right) D^{A_1} \left(U^{A_1}\right)^t\right)^{-1} A_1A^t \quad \text{(SVD de } A_1\text{)} \\ &= AA_1^t \left(U^{A_1} \left(D^{A_1}\right)^2 \left(U^{A_1}\right)^t\right)^{-1} A_1A^t \\ &= AA_1^tU^{A_1} \left(D^{A_1}\right)^{-2} \left(U^{A_1}\right)^t A_1A^t \\ &= A \left(V^{A_1}\right) D^{A_1} \left(U^{A_1}\right)^t U^{A_1} \left(D^{A_1}\right)^{-2} \left(U^{A_1}\right)^t U^{A_1} D^{A_1} \left(V^{A_1}\right)^t A^t \\ &= A \left(V^{A_1}\right) \left(V^{A_1}\right)^t A^t. \end{aligned}$$

Es decir, para el método de Nyström se tiene que la matriz de aproximación se puede ver como una proyección sobre el espacio generado por las entradas muestreadas. Debido a que la muestra que se toma es aleatoria, la proyección puede considerarse aleatoria. El hecho que $\widehat{K}^{Nys,\,esp}$ coincida con $\widehat{K}^{Nys,\,vp}$ es una característica del método de Nyström que permite una interpretación más sencilla y que ayuda a que el desempeño del método sea mejor en cuanto a la reconstrucción de la matriz [18].

3.3.6. Supuestos del método

Como lo mencionan Talwallkar *et al.* (2010) [29] la efectividad del método de Nyström radica en los siguientes dos supuestos sobre la matriz K:

1. Una aproximación de K de menor rango es útil para la técnica de análisis de datos que se desea utilizar.

2. La aproximación de K puede ser hecha de manera precisa con la información extraída de un pequeño subconjunto de las columnas de K.

Los autores mencionan que el primer supuesto se ha cumplido en muchas aplicaciones en las que el método de Nyström se utiliza para hacer SVD o PCA sobre los datos. El segundo supuesto tiene que ver con el hecho de si K puede ser bien representada mediante una matriz de rango menor y con el método de muestreo. Es decir, se supone que la matriz K es de rango menor a sus dimensiones o que su estructura puede ser bien representada por otra matriz de menor rango. En este sentido, el método de Nyström es muy efectivo cuando la parte K_{22} de la matriz de Gram es una combinación lineal de W. Esto puede verse mediante un teorema presentado por Kumar *et al.* en 2009 [19] el cual se presentará en la siguiente subsección.

Esta tesis no se enfoca en medir la capacidad de extraer suficiente información con pocas columnas. Sin embargo, a continuación se presenta una medida para que el lector tenga conocimiento de las herramientas que se pueden utilizar para saber si el método de Nyström es adecuado para sus datos.

Talwallkar *et al.* (2010) [29] introdujeron una medida para caracterizar la capacidad de extraer la información necesaria con pocas columnas a la que llamaron "matrix coherence", que se traducirá como coherencia matricial. Los autores demostraron que dicha medida, que depende de los datos, está ligada con el desempeño del método de Nyström.

Definición 5 *Coherencia matricial. La coherencia de una matriz V de dimensión $n \times r$ con columnas ortonormales es*

$$Coh\left(V_r\right) = \sqrt{n} \max_{i,j} V\left[i,j\right].$$

La coherencia se encuentra en el intervalo $[1, \sqrt{n}]$.

Para el caso en que se tiene una matriz de datos, la coherencia se calcula en base a la matriz cuyas columnas son los primeros r vectores propios de la matriz de datos. Talwakar et al. vieron que se puede recuperar la información muestreando de manera aleatoria columnas de las matrices con poca coherencia, en particular el método de Nyström funciona muy bien estos casos.

El problema obvio es calcular la coherencia ya que si se contara con los eigenvectores de la matriz K no hay necesidad de usar una aproximación. En 2010, los autores publicaron un artículo [24] en el cual proponen una medida de coherencia más robusta y proponen un método de estimación para ella.

Cuando la matriz de interés puede ser representada por otra de rango menor, es importante contar con un método de muestreo que extraiga la mayor información posible. Una pregunta natural es si al muestrear de una distribución uniforme, se recupera la información necesaria para la reconstrucción de la matriz. Otros esquemas de muestreo que incorporan más información acerca de los datos han sido propuestos, como los de Tropp *et al.* (2011) [15]. Sin embargo, como se puede ver en Kumar *et al.* (2012) [20] el más aconsejable es el muestreo aleatorio uniforme sin reemplazo. Kumar *et al.* (2012) [20] compararon la eficiencia de diferentes métodos de muestreo para el método de Nyström y propusieron un esquema de muestreo adaptativo más eficaz. No obstante, aseguran que de los muestreos fijos el más eficiente es el muestreo aleatorio simple sin reemplazo. Otros esquemas son más costosos debido a que requieren hacer cálculos sobre la matriz K y no han superado en

eficiencia al muestreo uniforme sin reemplazo. Además, no es posible utilizar estos esquemas para datos de gran tamaño. Es por estas razones que esta tesis se enfoca en muestreo aleatorio simple sin reemplazo para el método de Nyström.

3.3.7. Cotas para el error de aproximación

Como se vió anteriormente, la matriz K se particionó de la forma:

$$K = \begin{bmatrix} W & K_{21} \\ K_{21}^t & K_{22} \end{bmatrix}$$

y

$$\widehat{K}^{Nys,\ esp} = \begin{bmatrix} W & K_{21} \\ K_{21}^t & K_{21}^t W^{-1} K_{21} \end{bmatrix}. \tag{50}$$

Tomando en cuenta las expresiones anteriores, se puede observar que el error que se comete con la aproximación depende solamente de qué tan bien se aproxima la parte de la matriz K_{22}. Es decir el error de aproximación es $\|K_{22} - K_{21}^t W^{-1} K_{21}\|$. ¿Por qué es relevante ver el error de esta forma? Se debe recordar que los eigenvectores estimados de K se obtuvieron como una interpolación de los eigenvectores de la matriz W. Lo anterior nos dice que, si el rango de K fuera menor que sus dimensiones y al muestrear se formara una matriz W con columnas que son linealmente independientes entonces el método de Nyström sería muy eficiente. En este caso, se tendría que sería muy probable (dependiendo de cuál es el rango) que K_{22} se pueda escribir como combinación lineal de los eigenvectores de W por lo que la aproximación $K_{21}^t W^{-1} K_{21}$ sería buena. En el caso en que la matriz W tiene el mismo rango que K, tenemos que K_{22} pertenece al espacio generado por los eigenvectores de W por lo cual la aproximación es exacta. Dicho resultado fue presentado por Kumar *et al.* (2009) [19] y se muestra a continuación.

Teorema 1 *Supongase que ran$(K) = r \leq k \leq l$ y ran$(W) = r$. Entonces la aproximación por el método de Nyström de rango k es exacta, i.e.* $\left\| K - \widehat{K}_k^{Nys,\ esp} \right\|_F = 0$.

El teorema anterior sugiere que si se muestrea una parte de K que tiene el mismo rango que K, entonces no se comete error de aproximación. Lo anterior, como ya se mencionó, tiene sentido puesto que el espacio generado por los eigenvectores de W coincidiría con el generado por los eigenvectores de K y por lo tanto se recupera la información. Talwalkar et al. [18] mostraron el siguiente teorema para el tamaño de muestra necesario en términos de la coherencia para el caso en que ran$(K) = r$ y se desea obtener W tal que ran$(W) = r$, de forma tal que el método de Nyström sea exacto.

Teorema 2 *Sea K de dimensión $n \times n$ una matriz simétrica y semidefinida positiva de rango r y V la matriz cuyas columnas están formadas por sus r eigenvectores. Sea $\widehat{K}_k^{Nys}$ la aproximación de rango k de K obtenida mediante el método de Nyström muestreando $l \geq k \geq r$ columnas. Entonces es suficiente muestrear $l \geq r\ Coh^2(V_r)\, \text{máx}\left(c_1 \log(r), c_2 \log\left(\frac{3}{\delta}\right)\right)$ columnas, donde c_1 y c_2 son constantes positivas, para que*

$$P\left(\left\| K - \widehat{K}_k^{Nys,\ esp} \right\|_2 = 0\right) \geq 1 - \delta.$$

Como ya se mencionó, calcular la coherencia de una matriz de Gram con n grande no es posible por lo que Talwalkar *et al.* (2010) [24] propusieron una forma de estimarla. Así, en el teorema anterior podría utilizarse, con cierto grado de error, la estimación de la coherencia.

De manera general, el error de aproximación es $\|K_{22} - K_{21}^t W^{-1} K_{21}\|$. Kumar *et al.* (2009) [19] obtuvieron cotas para dicho error de aproximación. A continuación se presentan las cotas para el método de Nyström con muestreo aleatorio simple sin reemplazo y en el apéndice B se presenta una idea general de la forma en que se obtuvieron.

Para cualquier tamaño de muestra l se cumple que:

$$P\left(\left\|K - \widehat{K_k}^{Nys}\right\|_2 \leq \left\|K - \widehat{K_k}\right\|_2 + \frac{2n}{\sqrt{l}} K_{\text{máx}} \left[1 + \frac{\sqrt{\frac{n-l}{n-1/2}\left(1 - \frac{1}{2\,\text{máx}\{l,n-l\}}\right)^{-1} \log\frac{1}{\delta}}\, d_K}{K_{\text{máx}}^{\frac{1}{2}}}\right]\right) \geq 1 - \delta,$$

donde $d_K = \left(\underset{i,j}{\text{máx}} \|K[,i] - K[,j]\|\right)$ y $K_{\text{máx}} = \left(\underset{i}{\text{máx}} K_{ii}\right)$.

Kumar *et al.* (2012) [20] obtuvieron la siguiente cota para el error de aproximación con respecto a la norma de Frobenius:

$$P\left(\left\|K - \widehat{K_k}^{Nys}\right\|_F \leq \left\|K - \widehat{K_k}\right\|_F + \left(\frac{64k}{l}\right)^{\frac{1}{4}} n K_{\text{máx}} \left[1 + \frac{\sqrt{\frac{n-l}{n-1/2}\left(1 - \frac{1}{2\,\text{máx}\{l,n-l\}}\right)^{-1} \log\frac{1}{\delta}}\, d_K}{K_{\text{máx}}^{\frac{1}{2}}}\right]^{\frac{1}{2}}\right) \geq 1-\delta.$$

Las cotas de aproximación presentadas son para el caso de muestreo aleatorio simple sin reemplazo, sin embargo Drineas y Mahoney (2005) [9] obtuvieron cotas para cuando se realiza otro tipo de muestreo con reemplazo. El comportamiento de esta cota es bastante similar al que se observó para el método de columnas. Conforme n crece la cota se hace más grande, lo cual es de esperarse puesto que se tiene más error cuando hay más entradas en la matriz K. Si se fija n y l crece, la cota se hace más pequeña lo cual nos dice que entre mayor tamaño de muestra se tome el error de reducirá. Sin embargo, la mejora o la reducción que se hace irá siendo cada vez menor lo cual queda muy claro en las gráficas de desempeño del método que se mostraran en el siguiente capítulo. En dichas gráficas se observará que la aproximación mejora conforme crece el porcentaje de columnas muestreado, sin embargo la diferencia entre la mejora de porcentajes altos es menor que la diferencia en la mejora entre porcentaje bajos.

3.4. Random Fourier Features

3.4.1. Idea detrás del método

El método Random Fourier Features (RFF) es más reciente que el de columnas y el de Nyström, fue propuesto en 2007 por Rahimi y Recht [28]. Este método, a diferencia de los dos anteriores, fue

ideado específicamente para aproximar una matriz de Gram en la forma (24) de manera eficiente y utilizando resultados probabilísticos para las funciones kernel. La idea básica es expresar la función kernel como una esperanza matemática y luego aproximarla mediante una media muestral. Para poder hacer esto, se utiliza una transformación aleatoria z_θ de los datos tal que:

$$\begin{aligned} k(x,y) &= E_\theta\left(z_\theta(x)\, z_\theta(y)\right) \\ &\approx \frac{1}{l}\sum_{i=1}^{l} z_{\theta_i}(x)\, z_{\theta_i}(y). \end{aligned} \tag{51}$$

La transfomación aleatoria, como se verá después, depende solamente de la función kernel que se desea aproximar. Utilizando la aproximación (51) se aproxima cada entrada de la matriz kernel.

3.4.2. Algoritmo para obtener $\widehat{K}$ y $\widehat{Z}$

La transformación aleatoria que se hace a los datos depende del kernel que se desea aproximar. Como se verá en la siguiente subsección, lo que cambia es la distribución de la cual se generan ciertas variables aleatorias. Así, el método Random Fourier Features puede describirse de manera general; sin embargo esta tesis se enfocará al kernel gaussiano por lo que el procedimiento se describirá para dicho kernel.

El procedimiento es:

1. Generar l variables aleatorias i.i.d. $w_1, ..., w_l \in R^d$ de una distribución normal $N(0, \sigma^{-2}I)$.
2. Generar l variables aleatorias i.i.d. $b_1, ..., b_l \in R$ de una distribución uniforme $U[0, 2\pi]$.
3. Sean $\theta_i = \begin{pmatrix} w_i & b_i \end{pmatrix}$, $i = 1, ..., d$, $z_{\theta_i}(x) = \sqrt{2}\cos\left(w_i^t x + b_i\right)$ y $\overrightarrow{z_\theta}(x) = \frac{1}{\sqrt{l}}\left(z_{\theta_1}(x), ..., z_{\theta_l}(x)\right)$. Calcular la matriz de datos mapeados

$$Z^{rff} = \begin{pmatrix} \overrightarrow{z_\theta}(x_1) \\ \overrightarrow{z_\theta}(x_2) \\ \vdots \\ \overrightarrow{z_\theta}(x_n) \end{pmatrix}_{n\times l}.$$

4. Construir la matriz

$$\begin{aligned} \widehat{K}^{rff} &= Z^{rff}\left(Z^{rff}\right)^t \\ &= \begin{pmatrix} \langle \overrightarrow{z_\theta}(x_1), \overrightarrow{z_\theta}(x_1)\rangle & \langle \overrightarrow{z_\theta}(x_1), \overrightarrow{z_\theta}(x_2)\rangle & \cdots & \langle \overrightarrow{z_\theta}(x_1), \overrightarrow{z_\theta}(x_n)\rangle \\ \langle \overrightarrow{z_\theta}(x_2), \overrightarrow{z_\theta}(x_1)\rangle & \langle \overrightarrow{z_\theta}(x_2), \overrightarrow{z_\theta}(x_2)\rangle & \cdots & \langle \overrightarrow{z_\theta}(x_2), \overrightarrow{z_\theta}(x_n)\rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \overrightarrow{z_\theta}(x_n), \overrightarrow{z_\theta}(x_1)\rangle & \langle \overrightarrow{z_\theta}(x_n), \overrightarrow{z_\theta}(x_1)\rangle & \cdots & \langle \overrightarrow{z_\theta}(x_n), \overrightarrow{z_\theta}(x_n)\rangle \end{pmatrix}. \end{aligned}$$

3.4.3. Teoría detrás del método

La formulación de este método está basado en resultados de convergencia en probabilidad. Para poder aplicar dichos resultados, se hace el supuesto de que el kernel $k(,)$ que genera la matriz de Gram K es continuo, positivo definido e invariante bajo traslaciones. Bajo el supuesto anterior, se utiliza el siguiente teorema para poder escribir $k(,)$ como una esperanza.

Teorema 3 *(Bochner) Un kernel continuo $k(,)$ e invariante bajo traslaciones es positivo definido si y sólo si es la transformada de Fourier de una medida no negativa.*

Por el teorema anterior, se tiene que $k(,)$ es la transformada de Fourier de una medida no negativa. Reescalando dicho kernel de manera adecuada, se tiene que $k(,)$ es la transformada de Fourier de una densidad p, es decir

$$k(x,y) = k(x-y) = \int_{R^d} p(w) \exp(-i(x-y)^t w) dw.$$

Utilizando la formula de Euler $\exp(iz) = \cos(z) + i\sin(z)$ se tiene que:

$$\begin{aligned} k(x,y) &= \int_{R^d} p(w) \exp(-i(x-y)^t w) dw \\ &= \int_{R^d} p(w) \cos((x-y)^t w) dw - i \int_{R^d} p(w) \sin(-(x-y)^t w) dw. \end{aligned}$$

Como $k(,)$ es una función kernel real entonces:

$$\begin{aligned} k(x,y) &= \int_{R^d} p(w) \cos((x-y)^t w) dw \\ &= E_w \left[\cos\left[(x-y)^t w\right]\right]. \end{aligned} \tag{52}$$

Es decir, el kernel es la esperanza de cierta función de la variable aleatoria w. En RFF, se quiere escribir $k(x,y)$ como una esperanza del tipo $E_\theta(z_\theta(x) z_\theta(y))$, de forma tal que se obtenga una matriz Z^{rff} para la aproximación objetivo (24). Es decir, que se tenga que $\widehat{K}^{rff} = Z^{rff}\left(Z^{rff}\right)^t$. La proyección aleatoria que cumple lo anterior está dada por:

$$z_\theta(x) = \sqrt{2}\cos\left(w^t x + b\right), \tag{53}$$

donde $\theta = (w,b)$, w se genera con densidad p y b se genera de manera independiente de una distribución uniforme en $[0, 2\pi]$. En el apéndice B se presenta la demostración de (53) como aparece en Muñiz (2011) [25].

Así, se tiene que

$$k(x,y) = E_\theta\left(z_\theta(x)\, z_\theta(y)\right).$$

Lo anterior implica que en lugar de calcular $k(x,y)$ como el producto punto de dos datos transformados mediante ϕ, se puede calcular como la esperanza de dos datos transformados mediante z_θ.

La relación que existe entre la función kernel y la transformación que debe hacerse a los datos está dada por la densidad p con la que se genera w. Es decir, lo único que cambia en la formulación para z_θ dependiendo del kernel que se está aproximando, es la densidad p. Se ha demostrado que la transformada de Fourier de un kernel gaussiano de parámetro σ es una distribución normal $N(0, \sigma^{-2}I)$. Así, para el caso de interés de la tesis, las variables w se generan de la distribución normal $N(0, \sigma^{-2}I)$.

Hasta este momento, se escribió la función kernel $k(,)$ como una esperanza de variables aleatorias, sin embargo dicha esperanza es desconocida. Por lo tanto, el siguiente paso es el usual cuando no se conoce la esperanza de variables aleatorias: aproximarla mediante la media muestral. De aquí es de donde proviene el procedimiento de generar l proyecciones aleatorias $z_{\theta_1}, ..., z_{\theta_l}$ y definir $\overrightarrow{z_\theta}(x) = \frac{1}{\sqrt{l}}\left(z_{\theta_1}(x), ..., z_{\theta_l}(x)\right)$.

Se aproxima cada entrada de la matriz K de la siguiente manera:

$$\begin{aligned} k(x_k, x_j) &= E_\theta\left(z_\theta(x_k)\, z_\theta(x_j)\right) \\ &\approx \frac{1}{l}\sum_{i=1}^{l} z_{\theta_i}(x_k)\, z_{\theta_i}(x_j) \\ &= \overrightarrow{z_\theta}(x_k)\, \overrightarrow{z_\theta}(x_j) \\ &= \left\langle \overrightarrow{z_\theta}(x_k), \overrightarrow{z_\theta}(x_j) \right\rangle. \end{aligned}$$

Por lo anterior, se aproximará la matriz K mediante

$$\widehat{K}^{rff} = Z^{rff}\left(Z^{rff}\right)^t,$$

con

$$Z^{rff} = \begin{pmatrix} \overrightarrow{z_\theta}(x_1)^t \\ \overrightarrow{z_\theta}(x_2)^t \\ \vdots \\ \overrightarrow{z_\theta}(x_n)^t \end{pmatrix}_{n\times l}.$$

Debe tenerse en cuenta que l debe ser suficientemente grande para aproximar bien la media ya que lo que se está utilizando es la ley fuerte de los grandes números.

Es importante notar que las direcciones de proyección se eligen de manera independiente a los datos. La distribución de las direcciones solamente depende del kernel que se desea aproximar y no de los datos que se desea analizar. Como se verá después esto puede representar una desventaja al método, por lo que sería deseable encontrar una forma de hacer dichas elecciones dependientes de los datos sin que pierda muchas características de convergencia.

3.4.4. El método Random Fourier Features como proyección aleatoria

El método RFF consiste en aproximar el kernel mediante un promedio de variables aleatorias. Dichas variables aleatorias se generan primero proyectando los datos x sobre rectas w generadas por la densidad p, calculando $w^t x$. Es decir, el método RFF está basado en la proyección de los datos sobre direcciones aleatorias. Después de proyectar los datos sobre alguna dirección aleatoria, se calcula la función coseno sobre estos puntos (ver (53)). Así, el método RFF puede verse como una proyección aleatoria. En la Figura 9 presentada en Muñiz (2011) [25] se muestra dicho mapeo.

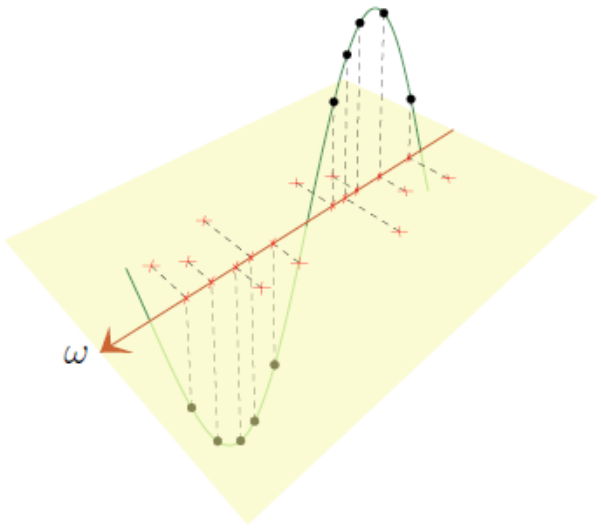

Figura 9: Gráfica del coseno de la proyección de los datos sobre la dirección w.

Como se mostró en la sección anterior, la razón por la cual esta proyección tiene sentido para generar una matriz que aproxime K, es que la función kernel que define dicha matriz es la transformada de Fourier de la distribución de la cual se generan las direcciones aleatorias.

3.4.5. Medidas del error de aproximación

Recht *et al.* (2007) [28] además de proponer el método anterior, encontraron una expresión para l que asegura que $\langle \overrightarrow{z_\theta}(x_k), \overrightarrow{z_\theta}(x_j) \rangle$ está cercano a $k(x, y)$. A continuación se presenta la cota más sencilla obtenida por ellos sin embargo en su artículo [28] puede verse una cota más estricta.

Se tiene que $z_\theta(x) = \sqrt{2}\cos(w^t x + b)$. Debido a que la función coseno siempre se encuentra entre -1 y 1 entonces $z_\theta(x)$ siempre estará en el intervalo $\left[-\sqrt{2}, \sqrt{2}\right]$. Así, para cualquier par de puntos fijos (x_k, x_j), usando (51) se tiene por la desigualdad de Hoeffding que

$$P\left(\left|\overrightarrow{z_\theta}(x_k)^t \overrightarrow{z_\theta}(x_j) - k(x_k, x_j)\right| \geq \in\right) \leq 2\exp\left(-\frac{l \in^2}{4}\right). \tag{54}$$

Como se puede observar, el error decae exponencialmente conforme el número de variables aleatorias generadas l crece. Intuitivamente es de esperarse que entre más variables aleatorias se generen con mayor probabilidad se tenga un mejor estimador de la esperanza. Puede observarse que, a diferencia de las cotas presentadas para los métodos de Nyström y de columnas, la cota para este método no depende del número de datos n sino de su dimensión d.

La cota (54) es para cada par de puntos pero es de interés el error que se comete en la aproximación de la matriz completa K. Tomando en cuenta la desigualdad (54) Recht *et al.* (2007) [28] construyeron la siguiente cota para todos los puntos simultáneamente.

Teorema 4 *(Convergencia uniforme del Fourier Features) Sea M un compacto en R^d con diámetro $diam(M)$. Entonces, para el mapeo $z_\theta(\cdot)$ se tiene que*

$$P\left(\sup_{x_k,x_j\in M}\left|\overrightarrow{z_\theta}(x_k)^t\overrightarrow{z_\theta}(x_j)-k(x_k,x_j)\geq\in\right|\right)\leq 2^8\left(\frac{\sigma_p\ diam(M)}{\in}\right)^2\exp\left(-\frac{l\in^2}{4(d+2)}\right), \quad (55)$$

donde $\sigma_p^2=E_p[w^tw]$ es el segundo momento de la transformada de Fourier del kernel $k(,)$.

En el apéndice B se presenta la idea de cómo se obtiene esta cota; la demostración completa puede verse en Recht *et al.* (2007) [28].

3.5. RFF PCA: una modificación del método RFF

En esta subsección se presenta una modificación hecha al método Random Fourier Features la cual, de cierta manera, introduce información de la distribución de los datos al algoritmo del RFF.

En la subsección anterior se vió que la función kernel se podía escribir como una esperanza de la siguiente manera:

$$k(x,y)=E_\theta[z_\theta(x)z_\theta(y)]. \quad (56)$$

La variable $z_\theta(x)$ se genera como una función de la proyección del dato x sobre el vector w. La dirección w que se utiliza para la proyección se genera de manera independiente a los datos, ya que ni la distribución ni sus parámetros cambian dependiendo de estos. Yang *et al.* (2012) [17] mencionaron que probablemente esta característica del método es lo que lo hace un poco ineficiente en comparación de otros cuya formulación se basa en los datos. Por ejemplo, aunque el método de Nyström utilice una distribución uniforme para elegir las columnas de K, el espacio que generan estas columnas está contenido en el espacio que generan todos los datos. De esta manera, el método introduce información de los datos. En cambio, el método RFF, para generar la matriz Z^{rff}, genera una dirección aleatoria w que no depende de los datos y luego los proyecta en esta dirección.

Tomando en cuenta la observación de Yang *et al.* (2012) [17], se realizó un cambio al método RFF el cual se presenta en esta subsección y con el cual se pretende introducir información de los datos al método. Debido a la construcción que se utilizó, al método resultante de dicha modificación se le llamó RFF PCA.

3.5.1. Idea detrás de la modificación

No es muy útil proyectar los datos en una dirección en la cual no tienen mucha variabilidad. Considérese el caso extremo en que los datos viven en un subespacio. Si se elige $w \in \left(\text{Span}\left\{vp_i\right\}_{i=1}^{d}\right)^{\perp}$ donde $\left\{vp_i\right\}_{i=1}^{d}$ son los vectores propios de la matriz X^tX, entonces $w^tx = 0$ y por lo tanto la proyección $z_\theta(x) = \sqrt{2}\cos(w^tx + b) = \sqrt{2}\cos(b)$. Es decir, la columna de Z^{rff} correspondiente a w estará constante, o sea, con varianza mínima, entonces la columna tendra un peso ("loading") cero en los componentes principales de Z^{rff}. Por lo anterior, no tiene sentido elegir w que pertecene al espacio ortogonal al generado por los datos.

Para evitar que $w^tx \approx 0$ y por lo tanto $z_\theta(x) \approx \sqrt{2}\cos(b)$, se decidió dar preferencia a direcciones en las cuales la proyección tenga mayor variabilidad. Lo anterior se hará realizando PCA sobre la matriz X, tomando los primeros d^* componentes principales y forzando a que w pertenezca al espacio generado por dichos componentes.

Supóngase que $\{vp_1, ..., vp_{d^*}\}$, $d^* \leq d$, son las componentes principales que se obtiene de hacer PCA sobre X. A continuación se presenta la expresión para w cuando se restringe al espacio generado por $\{vp_1, ..., vp_{d^*}\}$, la cual se denotará esta expresión por w^*.

La nueva dirección w^* está dada por:

$$\begin{aligned} w^* &= \sum_{i=1}^{d^*} \langle w, vp_i \rangle vp_i \\ &= \sum_{i=1}^{d^*} w^t vp_i vp_i. \end{aligned} \tag{57}$$

La expresión (57) corresponde a la dirección de proyección que se considerará para el nuevo algoritmo. Por su construcción, dicha dirección pertenece al espacio generado por los datos. Se puede utilizar (57) para generar las nuevas direcciones de proyección. No obstante, es fácil obtener la distribución de w^tvp_i la cual se presenta a continuación y hace más fácil la generación de w^*.

Para aproximar un kernel gaussiano de parámetro σ, cuando se utiliza el método RFF, w se genera de una distribución normal. Como $w \sim N(\mathbf{0}, \sigma^{-2}I)$ y los $\left\{vp_i\right\}_{i=1}^{d^*}$ están dados, entonces $w^tvp_i \sim N(vp_i\mathbf{0}, vp_i^t\sigma^{-2}Ivp_i)$, para $i = 1, ..., d^*$. Puesto que los $\left\{vp_i\right\}_{i=1}^{d^*}$ son ortonormales se tiene que $vp_i^tvp_i = 1$ y por lo tanto:

$$\begin{aligned} N\left(vp_i\mathbf{0}, vp_i^t\sigma^{-2}Ivp_i\right) &= N\left(0, \sigma^{-2}vp_i^tvp_i\right) \\ &= N\left(0, \sigma^{-2}\right). \end{aligned}$$

Es decir $w^tvp_i \sim N(0, \sigma^{-2})$. Por lo tanto, las nuevas direcciones de proyección se generarán calculando

$$w^* = \sum_{i=1}^{d^*} n_i vp_i. \tag{58}$$

donde $n_1, ..., n_{d^*}$ son variables aleatorias i.i.d. con distribución normal $N(0, \sigma^{-2})$.

El procedimiento a seguir para obtener las aproximaciones de la matriz kernel, de sus eigenvalores y de sus eigenvectores es el mismo que para RFF simplemente cambiando las w por w^*.

3.5.2. Comportamiento del algoritmo con base en su estructura

El comportamiento del método RFF PCA no es evidente por lo que sería bueno analizar su estructura para observar posibles deficiencias en cuanto a sus aproximaciones. Para obtener un panorama general de las aproximaciones dadas por este método primero se considerará un caso sencillo.

Caso sencillo

Sea $X = [x_i^t]_{i=1}^n$ la matriz de datos centrados de dimensión $n \times d$. Supóngase que los primeros d^* ($d^* \leq d$) vectores propios de la matriz $X^t X$ corresponden a la base canónica $\{e_i\}_{i=1}^{d^*}$. Bajo la base canónica, $x_i = \sum_{l=1}^{d} x_i[l]\, e_l$ y $w = \sum_{i=1}^{d} n_i e_i$, donde $x_i[l] = X[i,l]$.

Por (58) se tiene que $w^* = \sum_{i=1}^{d^*} n_i e_i$. Así,

$$\begin{aligned}
(w^*)^t x &= \left\langle \sum_{i=1}^{d^*} n_i e_i, \sum_{j=1}^{d} x_j e_j \right\rangle \\
&= \sum_{i=1}^{d^*} n_i x_i \\
&= \left\langle \sum_{i=1}^{d} n_i e_i, \sum_{i=1}^{d^*} x_i e_i \right\rangle \\
&= w^t x_{d^*}, \qquad (59)
\end{aligned}$$

donde x_{d^*} denota el vector que contiene las primeras d^* entradas del vector x.

Recuérdese la transformación $\cos\left((w^*)^t x + b\right)$ que se usa para calcular z_θ y aproximar el kernel gaussiano mediante el método RFF PCA. Por (59), se tiene que

$$\cos\left((w^*)^t x + b\right) = \cos\left(w x_{d^*} + b\right).$$

Por lo tanto, utilizar $\cos\left((w^*)^t x + b\right)$ para aproximar el kernel gaussiano basado en x será equivalente a utilizar $\cos(w x_{d^*} + b)$. Es decir, la modificación RFF PCA puede verse como el algoritmo exacto RFF tomando como matriz de datos a $X_{d^*} = \left[x_{i,d^*}^t\right]_{i=1}^n$. Así, la aproximación $\widehat{K}^{rff\ pca}$ converge a la matriz de gram obtenida a partir de los datos X_{d^*}.

El kernel gaussiano entre dos elementos x_{i,d^*} y x_{j,d^*} es

$$\begin{aligned} k\left(x_{i,d^*}, x_{j,d^*}\right) &= \exp\left(\frac{-\left\|x_{i,d^*} - x_{j,d^*}\right\|^2}{2\sigma^2}\right) \\ &= \exp\left(\frac{-\sum_{l=1}^{d^*}\left(x_{i,d^*}[l] - x_{j,d^*}[l]\right)^2}{2\sigma^2}\right) \\ &= \exp\left(\frac{-\sum_{l=1}^{d}\left(x_i[l] - x_j[l]\right)^2}{2\sigma^2}\right) \underbrace{\exp\left(\frac{\sum_{l=d^*+1}^{d}\left(x_i[l] - x_j[l]\right)^2}{2\sigma^2}\right)}_{\alpha_{i,j}} \\ &= k\left(x_i, x_j\right)\alpha_{i,j}. \end{aligned} \tag{60}$$

Lo anterior implica que la aproximación para el kernel dada por RFF PCA corresponde a una estimación del kernel multiplicado por una constante que depende de las observaciones x_i y x_j. En este sentido, la entrada i, j de la matriz kernel aproximada mediante la modificación es de la forma:

$$\widehat{K}^{rff\ pca}[i,j] = K[i,j]\alpha_{i,,j}. \tag{61}$$

Por lo tanto, si $\alpha_{i,,j} \neq 1$ las aproximaciones para cada entrada de la matriz kernel están siendo mal aproximadas. Por la definición de $\alpha_{i,,j}$ se tiene que:

$$\alpha_{i,,j} \begin{cases} = 1, & \text{si } i = j \\ \geq 1, & \text{si } i \neq j \end{cases}.$$

Antes de discutir el efecto de (61) sobre las estimaciones, se mostrará que (61) es cierto en general.

Caso general

Supóngase que $\{vp_i\}_{i=1}^{d^*}$ son los vectores propios de X^tX y $\{y_i\}_{i=1}^{n}$ las proyecciones de los datos en los componentes principales. Sea X^P la matriz X escrita bajo la base $\{vp_i\}_{i=1}^{d^*}$.

Bajo un cambio de base las distancias se preservan, por lo que $\left\|x_i^p - x_j^p\right\| = \left\|x_i - x_j\right\|$ y $k\left(x_i^p, x_j^p\right) = k\left(x_i, x_j\right)$. Por lo anterior, se trabajará con la matriz X^P en lugar de X.

Nótese que (59) también es cierto para la base $\{vp_i\}_{i=1}^{d^*}$, debido a su ortonormalidad. Por lo tanto, en este caso también se tiene que $\left(w^*\right)^t x^p = w^t x_{d^*}^p$ y se cumple:

$$k\left(x_{i,d^*}^p, x_{j,d^*}^p\right) = k\left(x_i^p, x_j^p\right)\alpha_{i,j} \quad \text{(por (60))}, \tag{62}$$

con $\alpha_{i,j} = \exp\left(\frac{\sum_{l=d^*+1}^{d}\left(x_i^p[l] - x_j^p[l]\right)^2}{2\sigma^2}\right) = \exp\left(\frac{\sum_{l=d^*+1}^{d}\left(x_i[l] - x_j[l]\right)^2}{2\sigma^2}\right)$.

Por (62) se tiene la expresión similar a (61):

$$\widehat{K}^{rff\ pca}[i,j] = K[i,j]\alpha_{i,,j}.$$

Error en la aproximación

En lo anterior se vió que, excepto por los elementos de la diagonal, las aproximaciones dadas por RFF PCA están estimando un múltiplo de las entradas de K. Por lo anterior, $\widehat{K}^{rff\ pca}$ difícilmente será una buena aproximación para K y además es de esperarse que la estimación de los valores propios también se vea afectada.

Supóngase que se tuviera que $\alpha_{i,,j} = c$, para $i \neq j$, $i, j \in \{1, ..., n\}$ y para alguna constante c. Se sabe que si $i = j$ entonces $\alpha_{i,,j} = 1$, por lo que se tendría que $\widehat{K}^{rff\ pca} = cK + (1-c)I$. Sea λ valor propio de K con vector propio v. Entonces:

$$\begin{aligned} \widehat{K}^{rff\ pca} v &= cKv + (1-c)Iv \\ &= c\lambda v + (1-c)v \\ &= (c\lambda + 1 - c)\, v. \end{aligned}$$

Por lo tanto, las aproximaciones de los valores propios de K también son afectadas por la constante multiplicativa pero los vectores propios, no.

Se realizaron diversos experimentos con la modificación y se observó que efectivamente, la estimación de la matriz de Gram y de los valores propios se veía bastante perjudicada por la característica del método de estimar un múltiplo de cada entrada de la matriz de Gram. No se observó ningún problema obvio en las estimaciones para los vectores propios dadas por RFF PCA. Esto no debería ser tan sorprendente por dos razones. La primera es que no existe una relación explícita entre los vectores propios de una matriz y la matriz en el sentido en que se puede cambiar una matriz bastante y tener los mismos vectores propios. Es por esto que, aunque se piense que si un método estima bien la matriz debe estimar bien los vectores propios o viceversa, no es cierto. La segunda razón es la estructura del algoritmo que probablemente ayuda a captar la información de los vectores propios de manera más rápida. Es decir, el método se diseñó de forma tal que se obtuviera una buena estimación de los vectores propios de K y no necesariamente de la matriz K.

Lo observado en esta sección podría ser útil para encontrar un cambio en el algoritmo que provea mejores estimaciones y cuyo costo computacional no sea grande. Dicho cambio no es obvio ya que se debe tomar en cuenta que calcular $\alpha_{i,,j}$ explícitamente, cuando se cuenta con un conjunto grande de datos, es muy costoso.

Se decidió estimar los $\alpha_{i,,j}$ tomando un subconjunto pequeño de los datos y calculando la media de los $\alpha_{i,,j}$ para ese subconjunto. Así, si se denota dicha media por α se tiene que la nueva estimación de la matriz de Gram dada por el método RFF PCA será de la forma:

$$\widehat{K}^{rff\ pca2} = \left(\widehat{K}^{rff\ pca} - (1-\alpha)I\right)\alpha^{-1}.$$

y los valores propios aproximados

$$\widehat{\lambda}^{rff\ pca2} = \left(\widehat{\lambda}^{rff\ pca} - 1 + \alpha\right)\alpha^{-1}.$$

3.5.3. Fórmulas para $\widehat{K}$ y $Z^{rff\ pca}$ obtenidas mediante el método RFF PCA

El procedimiento para obtener la aproximación de la matriz de Gram cuando esta es producida por un kernel gaussiano de parámetro σ es:

1. Elegir un número deseado de vectores propios d^* de X^tX ,$V = [vp_1, ..., vp_{d^*}]$.
2. Elegir un tamaño de muestra s y calcular la media $\alpha = \exp\left(\frac{\sum_{i=1}^{s-1}\sum_{j=d^*+1}^{d}(X[i,,j]-X[i+1,,j])^2}{2\sigma^2 s}\right)$
3. Generar l variables aleatorias i.i.d. $n_1, ..., n_l \in R^{d^*}$ de una distribución normal $N(\mathbf{0}, \sigma^{-2}I)$.
4. Calcular $w_j^* = \sum_{i=1}^{d^*} n_j[i]\, vp_i$ para $j = 1, ..., l$.
5. Generar l variables aleatorias i.i.d. $b_1, ..., b_l \in R$ de una distribución uniforme $U[0, 2\pi]$.
6. Sean $\theta_i = \begin{pmatrix} w_i^* & b_i \end{pmatrix}$, $i = 1, ..., d$, $z_{\theta_i}(X[i,]) = \sqrt{2}\cos\left((w_i^*)^t X[i,] + b_i\right)$ y $\overrightarrow{z_\theta}(X[i,]) = \frac{1}{\sqrt{l}}(z_{\theta_1}(X[i,]), ..., z_{\theta_l}(X[i,]))$. Calcular la matriz de datos transformados:

$$Z^{rff\ pca} = \begin{pmatrix} \overrightarrow{z_\theta}(X[1,]) \\ \overrightarrow{z_\theta}(X[2,]) \\ \vdots \\ \overrightarrow{z_\theta}(X[2,]) \end{pmatrix}_{n\times l},$$

7. Construir la matriz

$$\widehat{K}^{rff\ pca} = \alpha^{-1}\left(Z^{rff\ pca}\left(Z^{rff\ pca}\right)^t - (1-\alpha)I\right).$$

Observaciones

El método RFF PCA pierde la propiedad de convergencia a la verdadera matriz kernel. Sin embargo, se desea ver si para cuando el número de variables aleatorias l que se genera es pequeño, el hecho de que la elección de w depende de los datos ayuda a capturar mejor la esencia de estos.

Además de esta modificación se intentó otra basada en la varianza de las columnas de Z. Consistía en rechazar las columnas de Z que tuvieran varianza menor a una cierta cota. Si la columna tenía una varianza menor, se generaba una nueva variable aleatoria θ y por lo tanto una nueva columna. En los diferentes experimentos realizados, se observó que el desempeño del método utilizando esta modificación no era tan bueno y cuando era suficientemente bueno era muy similar al RFF y al RFF PCA. Por lo anterior, dicho método no se presenta en esta tesis.

3.6. Comparación de los métodos aleatorizados de aproximación de matrices

Se presentaron cuatro métodos para aproximar una matriz Kernel, sus eigenvalores y eigenvectores. En esta sección se comparan los métodos y se presentan algunos resultados teóricos con respecto al desempeño de cada uno. Para el método RFF PCA no se tienen resultados teóricos por lo que no se tomó en cuenta en algunas comparaciones. Se consideraron diferentes tareas y formas de medir qué tan bueno es un método en comparación del otro encontradas en la literatura.

Como ya se ha mencionado, si el procedimiento que se desea hacer es aproximar el resultado de Kernel PCA, el mayor interés está en la aproximación objetivo de K (24) y en la estimación de los eigenvectores de la matriz K. Sin embargo, también se estudiará el desempeño de los métodos en la estimación de la matriz Kernel K en sus diferentes descomposiciones y la complejidad del algoritmo para obtener la aproximación de la matriz kernel, los eigenvectores y eigenvalores.

Talwalkar *et al.* (2013) [18] estudiaron y compararon los métodos de columnas y Nyström. Yang *et al.* (2012) [17] realizaron un estudio comparativo entre el método de Nyström y el de Random Fourier Features. En esta sección se presentarán los resultados relevantes de estos artículos además de algunas conclusiones basadas en estos. En el último capítulo se presentan los experimentos y comparaciones empíricas de los diferentes métodos.

3.6.1. Diferencias estructurales entre los métodos

Como se había mencionado anteriormente, el método Random Fourier Features (así como su modificación) puede considerarse bastante diferente en cuanto a su formulación a los métodos de columnas y Nyström. Como ya se vió, la idea de RFF es expresar la función kernel como una expansión de Fourier para luego aproximarla como método Monte Carlo generando proyecciones aleatorias basadas en la transformada de Fourier del kernel. La idea básica de los métodos de columnas y Nyström es utilizar una muestra de la matriz K para encontrar aproximaciones de la matriz de Gram, de sus eigenvectores y sus eigenvalores.

Quizás la diferencia estructural más importante entre los métodos es que el método RFF no toma en cuenta las entradas de la matriz K y los otros tres, sí. RFF solamente necesita conocer el kernel que se desea aproximar para generar proyecciones aleatorias que no dependen de los datos. Esta diferencia podría no parecer tan importante pero como se verá después sí influye bastante en los resultados.

3.6.2. Desempeño en aproximar los eigenvectores de la matriz K

Las expresiones para las estimaciones de los eigenvalores y eigenvectores cuando se aproxima por una matriz de rango k dados por el método de columnas y por el Nyström son:

$$\widehat{\Sigma}^{Col} = \sqrt{\frac{n}{l}} D_k^C \qquad \text{y} \qquad \widehat{U}^{Col} = U^C = C V_k^C \left(D_k^C\right)^+$$

y

$$\widehat{\Sigma}^{Nys} = \frac{n}{l} D_k^W \qquad \text{y} \qquad \widehat{U}^{Nys} = \left(\sqrt{\frac{l}{n}} C U_k^W \left(D_k^W \right)^+ \right).$$

Como se puede observar, los estimadores para los eigenvectores de K dados por el método de columnas son los eigenvectores de la matriz de columnas muestreadas C. En cambio, para el caso del método de Nyström se tiene que los eigenvectores estimados son una extrapolación de los eigenvectores de la matriz W. Por ser $\widehat{U}^{Col}$ un conjunto de eigenvectores de una submatriz de K tienen las propiedades de una base, es decir son ortonormales. A diferencia de estos, $\widehat{U}^{Nys}$ no necesariamente cumple con esta propiedad. Lo anterior es debido a que no se sabe si al extrapolar se va a obtener la ortogonalidad entre los "eigenvectores". La forma de obtener la propiedad de norma unitaria es obvia pues simplemente se divide por la norma del vector. En cambio, la forma de obtener la ortogonalidad entres los vectores no es tan obvia. Como puede verse en Talwalkar *et al.* (2013) [18] los estimadores $\widehat{U}^{Nys}$ podrían ortogonalizarse usando una descomposición QR sin embargo, esto hace más costoso el procedimiento y no se ha visto que produzca mejores resultados en las estimaciones que el método de columnas.

Talwalkar *et al.* (2013) [18] investigaron de manera empírica cómo era el desempeño de los métodos de Nyström y de columnas para aproximar eigenvectores y eigenvalores. Para hacer esto utilizaron 6 conjuntos de datos diferentes y aproximaron utilizando diferentes rangos y número de columnas muestreadas. Las conclusiones que obtuvieron de su estudio fue que el método de columnas fue superior al método de Nyström en aproximar tanto los eigenvectores como los eigenvalores. El mejor desempeño del método de columnas en cuanto a la estimación de los eigenvectores se lo asignaron al hecho de que las estimaciones dadas por el método de Nyström no son ortogonales.

El método Random Fourier Features es más difícil de comparar en este sentido debido a que, a diferencia de los otros dos, no tiene una expresión cerrada para sus eigenvectores y eigenvalores estimados. Sin embargo, Yang *et al.* (2012) [17] analizaron el caso particular en el que se tiene una gran brecha entre los eigenvalores de la matriz K. Con una gran brecha se refieren a que hay unos cuantos valores propios que son mucho más grandes que el resto de eigenvalores. Esto lo que nos dice en función del espacio generado por los eigenvectores de la matriz K es que pocos eigenvectores generan un subespacio informativo. Debido a que el método RFF utiliza un subespacio para sus proyecciones que no depende de los datos, lo que se vió es que necesita generar más variables z_θ para poder captar la brecha. En cambio, el método de Nyström no necesita muchas muestras para poder captarla.

En general se esperaría que el método de columnas se desempeñe mejor que el método de Nyström y el RFF para estimar los eigenvectores de K. Lo anterior es de gran importancia para el caso en que se desea aproximar el resultado de Kernel PCA.

3.6.3. Desempeño en aproximar la matriz K

Para aproximar la matriz de Gram se utilizó la aproximación mediante la descomposición espectral y la de proyección, para los métodos de Nyström y de columnas. Para todos los métodos se utilizó la aproximación objetivo $\widehat{K} = \widehat{Z}\widehat{Z}^t$. En el caso de los métodos de Nyström y de columnas, esta última factorización corresponde también a la aproximación mediante descomposición espectral. Por lo anterior, se comparará el desempeño de todos los métodos con respecto a la aproximación

objetivo y el desempeño de los métodos de Nyström y de columnas con respecto a la aproximación dada por la descomposición de proyección.

Aproximación mediante la descomposición en proyección

Las aproximaciones de rango k para K mediante la descomposición de proyección dadas por el método de columnas y de Nyström son:

$$\widehat{K}^{Col,proy} = C\left(\left(C^tC\right)_k\right)^+ C^tK \tag{63}$$

y

$$\widehat{K}^{Nys,\,proy} = \frac{l}{n} C\left(W_k^2\right)^+ C^tK. \tag{64}$$

Como se puede observar, las expresiones anteriores son bastantes parecidas y coincidirían en el caso en que $\frac{l}{n}\left(W_k^2\right)^+$ y $\left(\left(C^tC\right)_k\right)^+$ fuesen iguales.

Talwalkar *et al.* (2013) [18] presentaron el siguiente teorema con respecto al desempeño de ambos métodos para aproximar K utilizando la descomposición en matriz proyección.

Teorema 5 *Las matrices de aproximación mediante la descomposición en matriz proyección dadas por el método de columnas y de Nyström son de la forma $U_CRU_C^tK$ donde R es una matriz de dimensión $l\times l$. Además, sobre todas las aproximaciones de la forma anterior, el método de columnas produce el menor error de aproximación en la norma de Frobenius si $k = l$.*

El teorema anterior provee una situación bajo la cual se tiene optimalidad por parte del método de columnas. Se mostrará a continuación que cuando $k = l$ entonces C es reconstruida de manera exacta. Recuérdese las siguientes factorizaciones obtenidas para K y C:

$$K = \begin{bmatrix} W & K_{21} \\ K_{21}^t & K_{22} \end{bmatrix} \tag{65}$$

y

$$C = \begin{bmatrix} W \\ K_{21} \end{bmatrix}. \tag{66}$$

Por (65) y (66), se tiene que la matriz de Gram se puede escribir como $K = \left[C, \widetilde{C}\right]$, donde $\widetilde{C} = \begin{bmatrix} K_{21} \\ K_{22} \end{bmatrix}$. Así, si $k = l$ la aproximación (63) es:

$$\begin{aligned} \widehat{K}^{Col,proy} &= C\left(\left(C^tC\right)_k\right)^+ C^tK \\ &= C\left(C^tC\right)^{-1} C^tK \quad (\text{Pues } k = l) \\ &= \left[C\left(C^tC\right)^{-1} C^tC, C\left(C^tC\right)^{-1} C^t\widetilde{C}\right] \\ &= \left[C, C\left(C^tC\right)^{-1} C^t\widetilde{C}\right]. \end{aligned}$$

Por lo anterior, C se reconstruye de manera exacta y el error de aproximación sólo depende de qué tan bien se aproxime $\widetilde{C}$.

Como lo mencionan Talwalkar *et al.* (2013) [18] la aproximación de la matriz de Gram utilizando la descomposición en matriz proyección, cuando n es demasiado grande, no representa una mejora computacional ya que se requiere guardar la matriz K y además hacer multiplicación con esta. Sin embargo, se estudia la aproximación por dicha descomposición ya que teóricamente se tienen resultados interesantes.

Aproximación mediante la descomposición espectral

A continuación se presentarán resultados importantes con respecto a la aproximación mediante la descomposición espectral de la matriz K. Como ya se mencionó estos resultados aplican para la forma equivalente $\widehat{K} = \widehat{Z}\widehat{Z}^t$ en que se escribió la aproximación para el método de columnas y de Nyström.

Talwalkar *et al.* (2013) [18] presentaron dos teoremas con respecto al desempeño de la aproximación mediante la descomposición espectral del método de columnas y el de Nyström. El primer teorema muestra que ambas aproximaciones pueden escribirse de cierta forma general y que sobre todas las aproximaciones de esa forma ni el método de columnas ni el Nyström es óptimo con respecto a la norma de Frobenius. Este teorema es de interés ya que expone que de manera general no se puede saber teóricamente cuál método será mejor para los datos si se desea utilizar la aproximación mediante la descomposición espectral.

El segundo teorema, que se enuncia a continuación y corresponde al Teorema 3 de Talwalkar *et al.* (2013) [18], provee circunstancias bajo las cuales sí se tiene optimalidad utilizando el método de Nyström.

Teorema 6 *Sea* $r = ran(K) \leq k \leq l$ *y* $ran(W) = r$*. Entonces, la aproximación mediante descomposición espectral utilizando el método de Nyström es exacta, mientras que el método de columnas es exacto sí y sólo si* $W = \left(\frac{l}{n}C^tC\right)^{\frac{1}{2}}$.

El teorema anterior hace bastante sentido, ya que lo que se hace es muestrear al menos tantas columnas de K como su rango y el hecho de que W también tiene el mismo rango, significa que la muestra que se tomó genera el espacio en el que viven las columnas de K. Debido a que se utilizan los eigenvectores de W para aproximar K mediante la descomposición espectral y dichos eigenvectores generan el mismo espacio que los eigenvectores reales de K, entonces tiene sentido que la aproximación sea exacta.

De manera empírica Talwalkar *et al.* (2013) [18] observaron que el método de Nyström se desempeña mejor que el método de columnas cuando se utiliza la aproximación mediante descomposición espectral para la matriz K. Sin embargo, este resultado tiene mucho que ver con el teorema anterior ya que la mayoría de las matrices de Gram que consideraron fueron de rango pequeño.

No se encontró en la literatura comparaciones entre la aproximación dada por el método Random Fourier Features y los métodos Nyström y de columnas. Yang *et al.* (2012) [17] realizaron un estudio comparativo entre el método RFF y el de Nyström, sin embargo sus comparaciones fueron hechas para el desempeño en estimar vectores propios y utilizando los métodos para problemas de regresión y clasificación. La conclusión a la que llegaron es que el método de Nyström se desempeña mejor que el RFF en diferentes tareas debido a que en el muestreo introduce información de los datos.

3.6.4. Complejidad de los algoritmos

Un aspecto importante que se debe considerar en la comparación de los métodos es su costo computacional. Dependiendo de la tarea que se desee realizar la complejidad del algoritmo puede variar. Las tres tareas que se estudiaron en esta tesis son: obtener la matriz de aproximación $\widehat{K}$, obtener las estimaciones $\widehat{U}$ de los vectores propios de K y obtener las proyecciones Y en los primeros k eigenvectores estimados de $\widetilde{Var}$. Recuérdese que las proyecciones Y se obtienen como un múltiplo de los vectores propios U de K, por lo tanto la complejidad para obtener las proyecciones coincide con la complejidad de obtener los vectores propios de K. Se seguira denotando por l al número de columnas muestreadas o variables aleatorias generadas según sea el método y se considerará que las aproximaciones son de rango k. Para la modificación del método RFF se tiene que el rango de la aproximación depende del número d^* de vectores propios de X^tX que se eligen para hacer la proyección.

Método	$\widehat{K}$	$\widehat{U}$ o Y
KPCA		$O(n^2k)$
Nyström	$O(nlk)$	$O(nlk)$
Columnas	$O(nlk)$	$O(nlk)$
RFF	$O(n^2l)$	$O(nlk)$
RFF PCA	$O(n^2l)$	$O(nld^*)$

Se puede observar que los método de columnas y Nyström coinciden en el orden de las complejidades. No obstante, como se puede ver en Talwakar *et al.* (2013) [18], el método de columnas es más costoso que el de Nyström por las constantes que no se toman en cuenta. El método RFF y su modificación no representan una ventaja computacional en cuanto a aproximar la matriz K, lo cual era de esperarse ya que no fue diseñado para aproximar la matriz de Gram sino el resultado de KPCA. Todos los métodos representan una gran ventaja cuando se desea calcular las proyecciones, el cual generalmente es el propósito cuando se hace KPCA. La modificación RFF PCA cuesta menos que RFF debido a que el rango d^* siempre es menor o igual que k.

4. Experimentos

En este capítulo se comparan los diferentes métodos aleatorizados de aproximación de matrices, presentados en el capítulo anterior, mediante algunos experimentos. En la primera subsección se presentan algunas medidas que se utilizarán para evaluar el desempeño de los métodos en diversas tareas. Posteriormente se presentan dos ejemplos: uno con datos artificiales, inspirado en el realizado por Yang *et al.* (2012) [17] y uno con datos reales, inspirado en el realizado por Smola *et al.* (2014) [14]. El número de datos que se consideró para los ejemplos no es muy grande ya que desea compararse con el resultado exacto de KPCA.

4.1. Medidas del desempeño

En esta sección se presentarán las diferentes medidas de desempeño que se utilizarán en los experimentos. La forma de medir este desempeño no es tan obvia pero se debe elegir una forma de medirlo. Se valorará el desempeño de los métodos en aproximar la matriz de Gram, sus eigenvalores y sus eigenvectores.

Para medir la calidad de la aproximación de la matriz de Gram se utilizarán las siguientes dos medidas. La primera es bastante natural: la diferencia entre la estimación obtenida y K menos la diferencia de K con su mejor aproximación de rango k. Es decir,

$$\text{Error matricial}= \left\| K - \widehat{K_k}^{\text{esp}} \right\|_F - \|K - K_k\|_F ,$$

donde $\widehat{K_k}^{\text{esp}}$ es la aproximación de rango k obtenida mediante alguno de los diferentes métodos. El error matricial está acotado por abajo por cero y no está acotado por arriba. La segunda medida, propuesta por Talwalkar *et al.* (2013) [18] es:

$$\text{Precisión relativa} = \frac{\|K - K_k\|_F}{\left\| K - \widehat{K_k}^{\text{esp}} \right\|_F}.$$

Esta medida siempre se encuentra entre cero y uno ya que el numerador es siempre más chico que el denominador puesto que K_k es la mejor aproximación de rango k para K. Entre más cercana a 1 sea la precisión relativa, mejor es la aproximación $\widehat{K_k}^{\text{esp}}$. La precisión relativa da una idea de qué tan mala o buena es una aproximación $\widehat{K_k}^{\text{esp}}$ con respecto a la mejor aproximación K_k de rango k.

Para valorar si los métodos captan bien la estructura de los vectores propios se presentarán las gráficas de estos y sus estimaciones. Además, se medirá el desempeño de los métodos en cuanto a la estimación del $i-$ésimo vector propio u_i de K mediante:

$$\text{Diferencia vectorial} = \left| u_i^t \widehat{u}_i \right| .$$

De esta manera, si la diferencia vectorial es cero quiere decir que los vectores u_i y $\widehat{u}_i$ son ortogonales y por lo tanto, muy diferentes. Si la diferencia vectorial es uno, quiere decir que coinciden, ya que los vectores tienen norma 1. Entonces, mientras más cercana a uno sea la diferencia vectorial la aproximación se considerará mejor.

Para los valores propios se utilizará la medida más natural que es simplemente la diferencia entre estos. Es decir, la diferencia entre el $i-$ésimo valor propio λ_i y su estimación está dada por:

$$\text{Diferencia en valores propios} = \left|\lambda_i - \widehat{\lambda}_i\right|.$$

Así, entre más cercana a cero sea la diferencia en valores propios la estimación es mejor.

También se desea explorar el desempeño de los diferentes métodos en aproximar las proyecciones dadas por KPCA y luego reconstruir el dato en el espacio original a partir de estas. Recuérdese que a dicha reconstrucción se le llamó preimagen. El procedimiento que se realizará es el presentado en la sección 2.4 con la diferencia que, cuando se utilice un método de aproximación, la proyección no será la dada por KPCA sino la estimación de esta. Se desea ver qué tanta diferencia hay entre la preimagen obtenida cuando se utiliza un método de aproximación y la obtenida cuando se realiza KPCA. Para medir dicha diferencia para el dato $i-$ésimo se utilizó la medida del error:

$$\text{Error en la preimagen} = \left\|\widehat{X}_{KPCA}[i,] - \widehat{X}[i,]\right\|_1,$$

donde $\widehat{X}_{KPCA}[i,]$ denota la preimagen obtenida a partir de las proyecciones dadas por KPCA y $\widehat{X}[i,]$ denota la preimagen obtenida al utilizar las proyecciones estimadas mediante alguno de los métodos de interés.

4.2. Ejemplo 1

El primer experimento está inspirado en el realizado por Yang *et al.* (2012) [17] y se pretende investigar qué tan buenas son las estimaciones de la matriz de Gram K, de sus eigenvectores y eigenvalores dados por los métodos de Nyström, de columnas, RFF y RFF PCA. Además, se desea utilizar KPCA como una herramienta de clasificación por lo que se crearán datos con una estructura particular.

Para este ejemplo se generaron $n = 5,000$ observaciones de dimensión $d = 102$. Las primeras dos dimensiones separarán las observaciones en dos grupos, como se muestra en la Figura 10 y las últimas 100 dimensiones corresponden a ruido generado de una distribución uniforme en $(0, 1)$. Los grupos fueron creados de la siguiente manera: la mitad de los puntos se generaron de una distribución uniforme en un círculo de radio 0.5 centrado en $(0.5, 0.5)$ y la otra mitad, en un círculo de radio 0.5 y centrado en $(-0.5, 0.5)$. Así, la matriz generada X es de dimensión $5,000 \times 102$ y de rango 102.

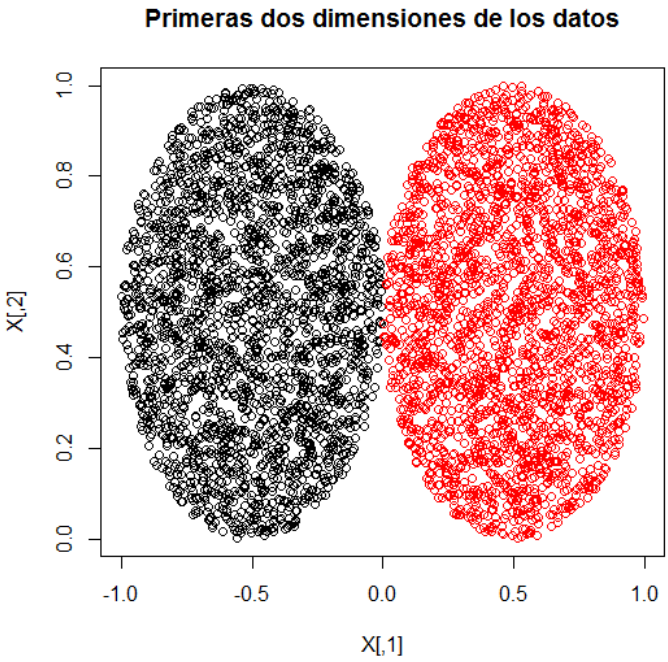

Figura 10: Gráfica de las primeras dos dimensiones de los datos, las cuales forman dos grupos que son diferenciados mediante el color.

La matriz de Gram para estos datos se produjo utilizando un kernel gaussiano de parámetro $\sigma = 8.69$, el cual se estimó mediante la función sigest de R [27]. Se calcularon las aproximaciones por descomposición espectral de rango $k = 60$ dadas por el método de Nyström, el de columnas, el Random Fourier Features (RFF) y de rango $k = d^* = 50$ para la modificación RFF PCA.

Recuérdese que para los métodos de Nyström y de columnas se debe elegir el número de columnas l que se tomará en la muestra y para el método RFF y su modificación RFF PCA se debe elegir el número de variables aleatorias l que se generarán. Por lo anterior, la complejidad de los métodos es similar y fueron elegidos como un porcentaje del número de columnas de K. Yang *et al.* (2012) [17] mostraron que el método RFF no se desempeñaba bien para clasificación cuando se utilizan solamente $l = 100$ variables aleatorias. Se desea observar si la modificación RFF PCA tiene un mejor desempeño que RFF en los casos en que se generan pocas variables aleatorias. Por lo anterior, los porcentajes de columnas de K elegidos fueron 2 %, 3 %, 5 % y 10 %, de forma tal que $l = 100$, 150, 250 y 500.

Como se vió en el capítulo anterior, el método RFF PCA estima un múltiplo de cada entrada de la matriz de Gram. Se vió que esta propiedad afecta las aproximaciones de K y de los valores propios dadas por este método. Se decidió estimar el múltiplo con la media α de los múltiplos $\alpha_{i,j}$ calculados para un subconjunto pequeño de los datos. En este caso se tomó el subconjunto dado por los primeros 101 renglones y la primer columna de X.

Por la estructura creada en los datos, se tiene una matriz de Gram con una gran brecha entre los eigenvalores, es decir que tiene algunos eigenvalores mucho más grandes que los demás. Los primeros dos eigenvalores son 1847.11 y 65.89 respectivamente. Se tiene interés en estimar el segundo vector propio de K porque el grupo al que se asigna un dato, se verá reflejado en el signo de la entrada correspondiente a dicho dato del segundo vector propio de K. Es decir, la finalidad es construir un clasificador basado en el segundo vector propio.

Para obtener los diagramas de caja que se presentan a continuación, se repitió el experimento 30 veces.

4.2.1. Aproximación de la matriz de Gram mediante su descomposición espectral

En esta subsección se estudiará el desempeño de los métodos al aproximar la matriz de Gram en su descomposición espectral. La Figura 11 muestra los diagramas de caja para el error matricial de aproximación a la matriz de Gram obtenida para los diferentes métodos.

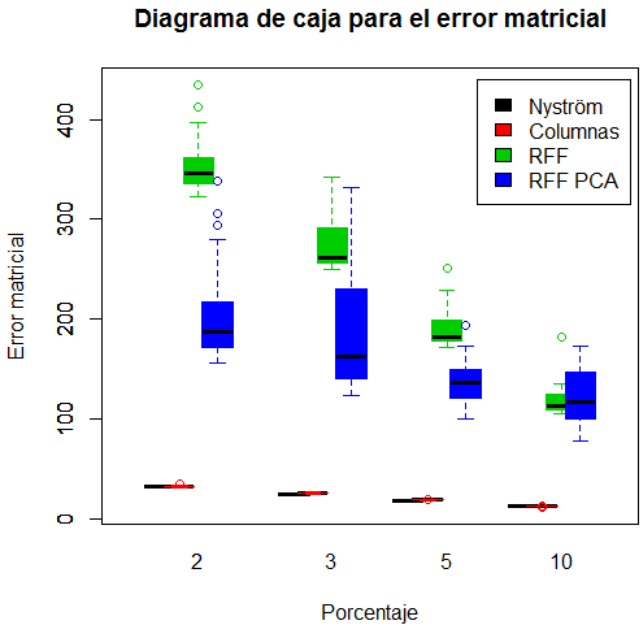

Figura 11: Diagrama de caja para el error matricial obtenida para diferentes métodos y diferentes porcentajes de columnas muestreadas o v.a. generadas.

Se puede observar que los métodos de Nyström y de columnas son mejores que los otros dos, ya que el error matricial es muy cercano a cero. Además, se debe notar que para todos los métodos el error decae conforme se aumenta el número de columnas muestreadas o variables aleatorias generadas según sea el caso. Esta propiedad es algo muy deseable en los métodos, ya que de esta

forma se observa empíricamente que conforme se aumenta el porcentaje, las estimaciones mejoran. La varianza de los métodos de Nyström y de columnas es pequeña por lo que se podría decir que sus estimaciones además de buenas son bastante estables. En este caso, se tuvo una mejora del método RFF en la estimación de la matriz de Gram mediante la modificación. Puede observarse que conforme el porcentaje aumenta, el método RFF mejora y en el 10 % ya es muy similar a su modificación.

En la Figura 12 se muestra el diagrama de caja para la precisión relativa obtenida para los diferentes métodos.

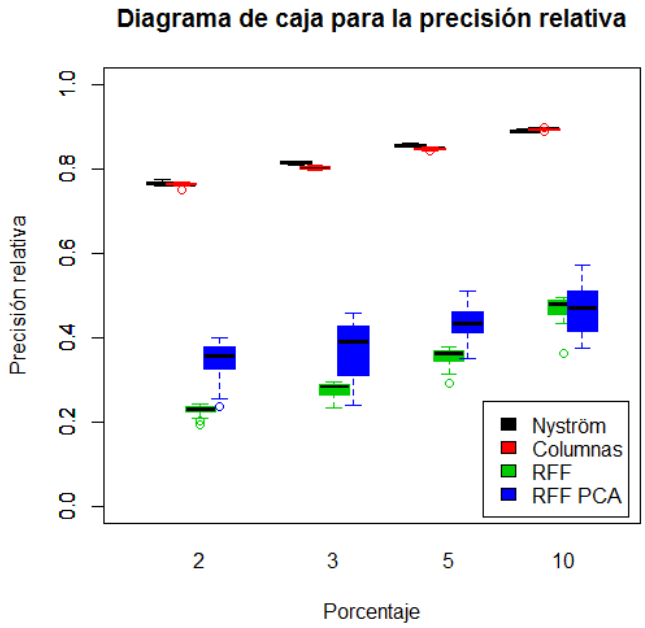

Figura 12: Diagrama de caja para la precisión relativa de las aproximaciones de la matriz de Gram obtenidas mediante los diferentes métodos.

Como es de esperarse, el comportamiento que se observa para los métodos es similar al observado con la medida anterior. El método de Nyström y el de columnas fueron sustancialmente mejores en aproximar K que los otros dos métodos. El métodos de Nyström es el mejor en aproximar la matriz de Gram.

4.2.2. Aproximación de los vectores propios

Como ya se mencionó, los datos se construyeron de forma tal que se obtenga una gran brecha entre los eigenvalores de K. La brecha en los eigenvalores es creada a través de los grupos y lo que se espera es poder detectar los dos grupos mediante los vectores propios de K. A continuación

se muestran las gráficas para los primeros dos eigenvectores de K y sus estimaciones obtenidas mediante diferentes métodos utilizando solamente el 2 % de las columnas de K, de forma tal que $l = D = 100$.

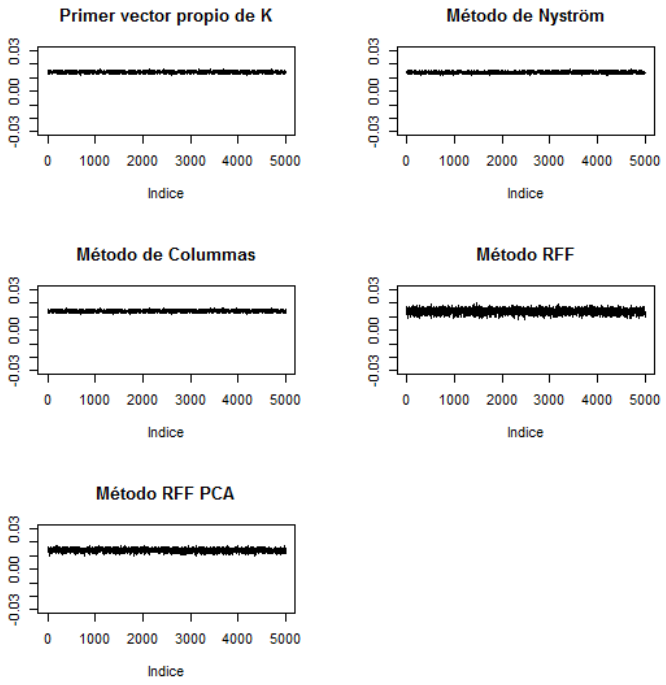

Figura 13: Primer vector propio de la matriz de Gram y sus estimaciones obtenidas mediante los diferentes métodos (con $l = 100$).

De las gráficas se observa que el comportamiento del primer vector propio es bien captado por todos los métodos. Sin embargo, vemos que los métodos de Nyström, columnas y RFF PCA parecen producir una mejor estimación que el RFF. En la Figura 14 se presentan las gráficas del segundo vector propio de K y sus estimaciones.

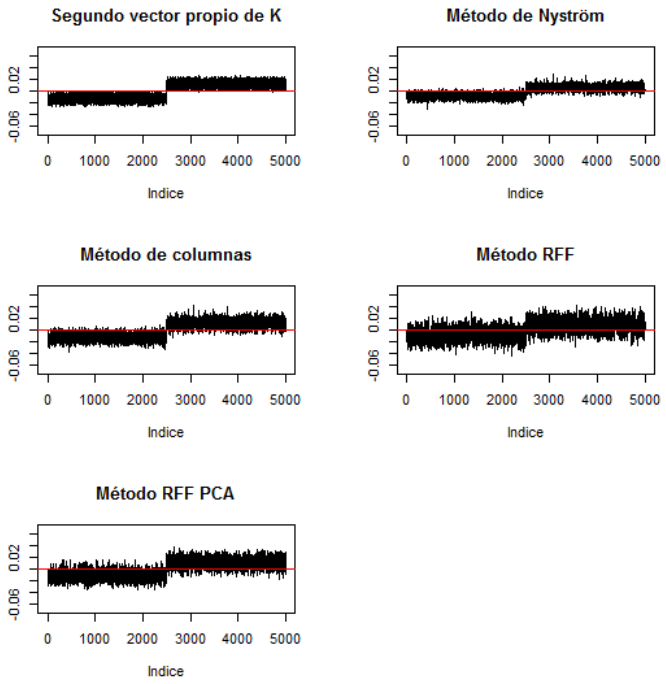

Figura 14: Segundo vector propio de la matriz de Gram y sus estimaciones obtenidas mediante los diferentes métodos (con $l = 100$).

Se puede observar que los dos grupos creados o la brecha en los eigenvalores se refleja en el segundo vector propio como un "salto". Es deseable que el método aleatorizado de aproximación de matrices que se utilice, detecte estos saltos ya que indicarían posibles grupos en los datos. El grupo al que se asigna la j-ésima observación, está dado por el signo de la entrada $j-$ésima del segundo vector propio. En este caso simple, todos los métodos parecen rescatar el salto con tan sólo el 2 % de columnas de K muestreadas. Sin embargo, el método RFF no capta muy bien el salto, lo cual provoca malas clasificaciones. Los métodos de columnas y Nyström marcan muy bien el salto y estiman mejor el segundo vector propio. El método RFF PCA está en un punto intermedio.

En las Figuras 15 y 16 se muestran los diagramas de caja para las diferencias vectoriales obtenidas para los primeros dos vectores propios de K y sus estimaciones mediante los diferentes métodos y dependiendo del porcentaje de columnas muestreadas o variables aleatorias generadas.

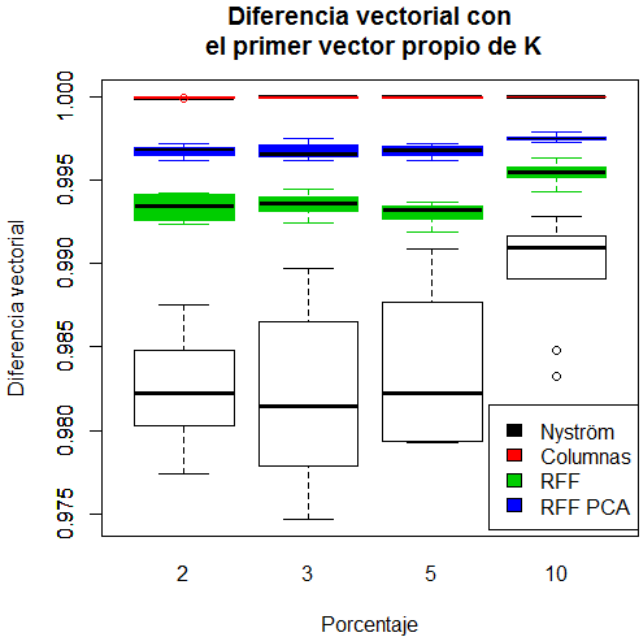

Figura 15: Diagrama de caja para la diferencia vectorial entre las estimaciones obtenidas con los diferentes métodos y el primer vector propio de K, para diferentes porcentajes de columnas muestreadas o v.a. generadas.

Como se puede observar en la Figura 15, para el primer vector propio se tiene que el desempeño de los métodos es bastante bueno ya que todos se encuentran muy cercanos a uno. Como se vió en las gráficas anteriores, este vector propio no tenía "saltos" por lo que es más fácil de estimar. Se puede observar que el método RFF PCA da una mejor estimación del vector propio que RFF y que el de Nyström bajo esta medida. Para todos los métodos se observa poca varianza y además esta disminuye conforme el porcentaje de columnas muestreadas aumenta. El método de columnas es el que mejor estima el primer vector propio.

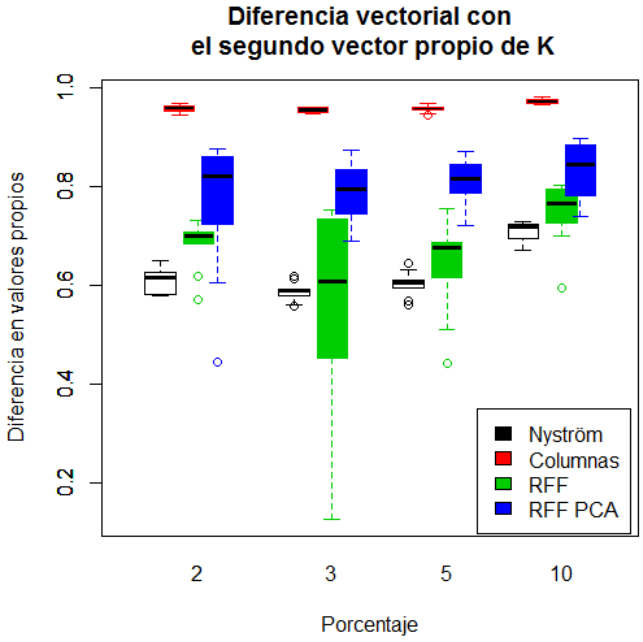

Figura 16: Diagrama de caja para la diferencia vectorial entre las estimaciones obtenidas con los diferentes métodos y el primer vector propio de K, para diferentes porcentajes de columnas muestreadas o v.a. generadas.

En la Figura 16 se observa que el método de columnas estima el segundo vector propio de manera eficiente con pocas columnas, ya que aún cuando se utiliza solamente el 2 % para el número de variables aleatorias generadas, la diferencia vectorial es bastante cercana a 1. Además, la varianza para este método en cuanto a sus estimaciones es muy pequeña, lo cual es bastante bueno. La modificación RFF PCA obtiene una mejor aproximación del segundo vector propio que los método RFF y el de Nyström, los cuales necesitan un mayor número de variables aleatorias generadas para producir un mejor resultado.

Las gráficas anteriores corresponden a una medida de desempeño particular para las aproximaciones de los vectores propios. En este sentido, bajo esta medida se tiene que el método de Nyström es el peor de todos. Sin embargo, de simplemente observar las Figuras 13 y 14 se puede concluir que el método de Nyström y el de columnas son los mejores. Se desea discernir qué método capta mejor la estructura de los vectores propios, principalmente la del segundo vector propio. Por dicha razón, se decidió contar el número de observaciones mal clasificadas de forma tal que entre mayor sea este número, el método de aproximación es peor. En la Figura 17 se presenta el diagrama de caja para el número de observaciones mal clasificadas en base al segundo vector propio estimado de K por los diferentes métodos.

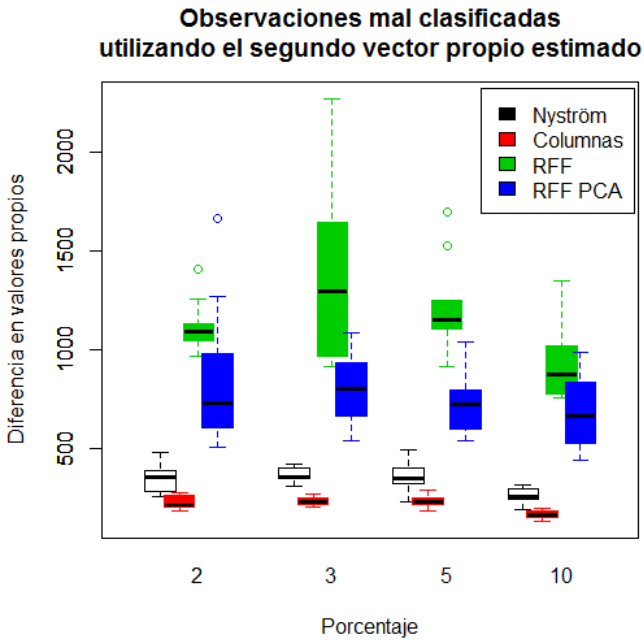

Figura 17: Observaciones mal clasificadas a partir de la estimación del segundo vector propio mediante diferentes métodos.

Se puede observar que, los métodos de Nyström y de columnas son bastante buenos y estables en su clasificación. En este caso, se observa que la modificación RFF PCA sí produce mejores resultados que el método RFF, ya que clasifica bien un mayor número de observaciones que el RFF.

Lo que se puede concluir de las Figuras presentadas en cuanto al desempeño de los métodos en la estimación del vector propio, es que los métodos de columnas y Nyström estiman bastante bien los vectores propios y la modificación RFF PCA es mejor que el método RFF cuando el porcentaje es bajo. Como ya se mencionó, para el caso en que se desea aproximar el resultado dado por KPCA, el interés principal es la aproximación de los vectores propios.

4.2.3. Aproximación de los valores propios

A continuación se muestra la diferencia entre los primeros dos valores propios y sus estimaciones obtenidas mediante los diferentes métodos.

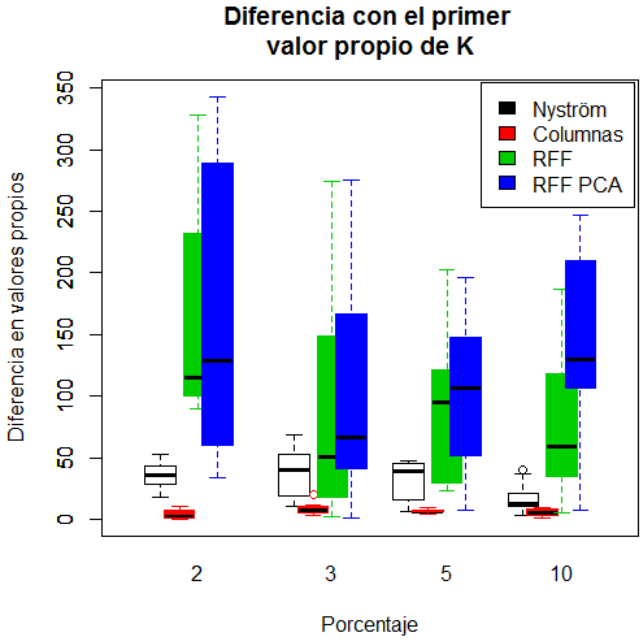

Figura 18: Diagrama de caja para la diferencia con el primer valor propio para los diferentes métodos dependiendo del porcentaje de columnas muestreadas.

En la Figura 18 se puede observar que a pesar de que el método RFF PCA es bastante malo en estimar el primer valor propio, para porcentajes bajos es muy similar al RFF. Recuérdese que en este caso, para estimar las constantes multiplicativas que afectan la aproximación, se utilizó una muestra de 101 elementos (de un total de $5,000$) para calcular la media α. Por lo anterior, no es tan sorprendente que el método no sea tan bueno en estimar el primer valor propio. El error para los métodos es bastante grande pero debe considerarse que el primer valor propio es 1847, el cual es bastante grande. Puede observarse una tendencia de mejora (excepto para RFF PCA) conforme el porcentaje aumenta. El método de columnas es el que estima mejor el primer valor propio, con un error muy bajo aún cuando solamente se utiliza el $2\,\%$ de las columnas de K.

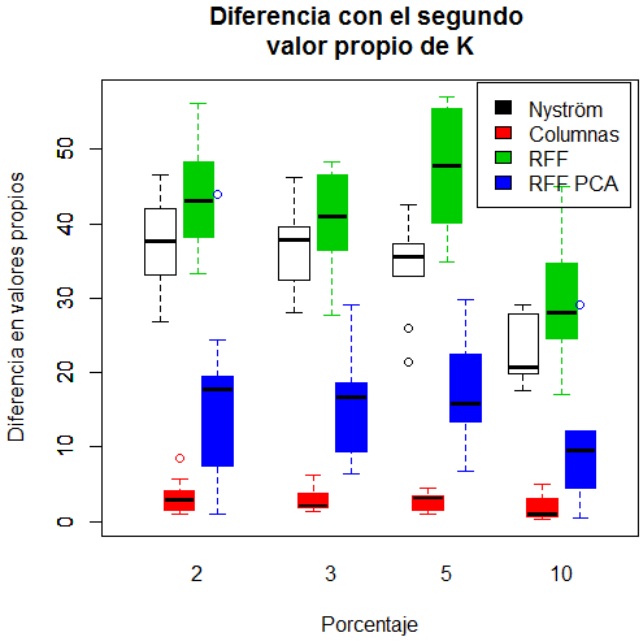

Figura 19: Diagrama de caja para la diferencia con el segundo valor propio para los diferentes métodos dependiendo del porcentaje de columnas muestreadas.

Se puede observar en la Figura 19 que el método RFF PCA no es tan malo en estimar el segundo valor propio, inclusive es mejor que el método RFF y que el Nyström. Posiblemente la razón de que esta estimación no sea tan mala es que en este caso el valor propio es 65.89 (mucho más pequeño que el primer valor propio) por lo que las diferencias o errores que se van sumando por cada observación ($\alpha_{i,j}$) son más pequeños que en el primer caso. En cuanto a la estimación de los valores propios, el método de columnas es considerablemente mejor que los demás métodos, ya que también en este caso su error es casi cero aún cuando se utiliza solamente el 2 % de columnas muestreadas.

Observaciones

En el experimento anterior se observó que el método de Nyström es superior a los demás en aproximar la matriz de Gram y el de columnas en estimar los valores propios. Los métodos de Nyström y de columnas se desempeñan bastante bien en todas las tareas. El método RFF PCA resultó mejor que el RFF en la mayoría de los casos, a pesar de que las estimaciones de los valores propios y de la matriz de Gram se ven afectadas por los valores $\alpha_{i,j}$.

En este experimento, al igual que en el realizado por Yang *et al.* (2012) [17], se obtuvo una gran diferencia entre el desempeño del método de Nyström y RFF. Smola *et al.* (2014) [14] mencionaron que no encontraron esta gran diferencia entre los métodos obtenida Yang *et al.* (2012) [17].

Además se observó que, como lo mostraron empíricamente Talwalkar *et al.* (2013) [18], el método de Nyström es mejor que el de columnas en aproximar K utilizando la descomposición espectral pero

el método de columnas es mejor en estimar los eigenvectores de K que el de Nyström. Talwalkar *et al.* mostraron que dicho comportamiento hacía sentido con los resultados teóricos que presentaron en su artículo [18]. En este ejemplo puede notarse dicha relación entre los métodos de Nyström y de columnas.

Se debe recalcar que los experimentos se realizaron considerando valores mayores para el número de variables aleatorias generadas o columnas muestreadas l, y se observaron comportamientos similares a los aquí presentados. Para porcentajes muy altos se tuvo un desempeño de los métodos de Nyström, de columnas y RFF bastante bueno (errores muy bajos), pero como era de esperarse no se tuvo ese desempeño para la modificación RFF PCA. Recuérdese que RFF PCA no cuenta con la propiedad de convergencia a la matriz kernel, además de que sus estimaciones se ven afectadas por los valores $\alpha_{i,j}$.

4.3. Ejemplo 2

En el segundo ejemplo se desea explorar el desempeño de los diferentes métodos en aproximar las proyecciones dadas por KPCA y luego reconstruir la proyección en el espacio de los datos. El procedimiento que se realizará es el presentado en la sección 2.4 con la diferencia que, cuando se utilice un método de aproximación, la proyección no será la obtenida por KPCA sino la estimación de esta.

Este ejemplo está inspirado en el realizado por Smola *et al.* (2014) [14]. Los datos que se utilizarán son los del conjunto MNIST [34]. A diferencia del ejemplo del capítulo dos, en este caso se consideran los datos completos. Como ya se había mencionado, cada dato es la digitalización de un dígito manuscrito del 0 al 9. La idea es mapear los datos a un espacio de menor dimensión que los caracterice bien para posteriormente reconstruirlos, como se muestra en la Figura 20. Para lo anterior se utiliza una función kernel para transformar los datos, se proyecta la matriz kernel sobre los vectores propios de $\widetilde{Var}$ y se emplean dichas proyecciones para reconstruir la imagen. Recuérdese que a las imágenes reconstruidas mediante diferentes métodos se le llamó preimagen.

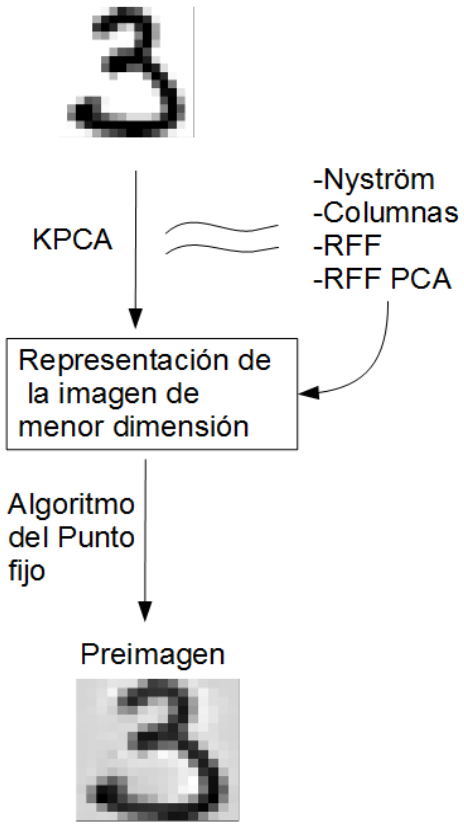

Figura 20: Reconstrucción de la preimagen mediante randomized autoencoders.

Como se muestra en la Figura 20, el resultado de KPCA se puede aproximar utilizando los métodos aleatorizados de aproximación de matrices. Al procedimiento por el cual la reconstrucción de la imagen en el espacio original se obtiene a partir de las proyecciones aproximadas mediante los métodos aleatorizados de aproximación de matrices, se le conoce en Ciencias de la computación como "*randomized autoencoders*". En este ejemplo se pretende mostrar el desempeño de los diferentes métodos de aproximación presentados en el Capítulo 3 cuando se realiza *randomized autoencoders* y además se reduce bastante algunas variables que afectan la calidad de la imagen.

4.3.1. Descripción del conjunto de datos

Para este ejemplo se eligieron las imágenes del conjunto de datos MNIST, el cual puede obtenerse en [34]. Como ya se había mencionado, cada renglón de la matriz de datos representa un dígito. Se consideró un subconjunto de entrenamiento que cuenta con 7291 observaciones. Así, la matriz de datos X que se utilizó es de dimensión 7291×256 y cada renglón de dicha matriz corresponde a la imagen de un dígito entre el cero y el nueve. El rango de la matriz X es 256. En la Figura 21 se presenta una muestra de los datos.

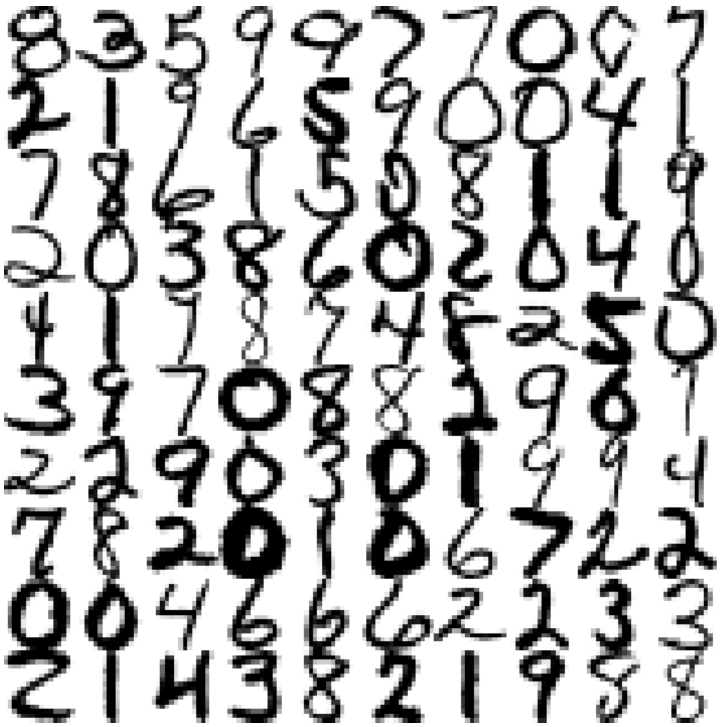

Figura 21: Muestra de los datos del conjunto MNIST.

4.3.2. Desempeño de los métodos

En esta subsección se presentarán los resultados obtenidos cuando se realiza *randomized autoencoders* utilizando los métodos de aproximación presentados en el Capítulo 3 para aproximar la matriz de Gram. Esta subsección se divide en dos partes. En la primera, se presentan los resultados obtenidos para todos los métodos cuando se realiza una compresión alta del autoencoder (rango chico de la matriz de aproximación) y se trabaja con un número limitado de proyecciones (l pequeño). En la segunda, se comparan los métodos RFF y RFF PCA.

4.3.2.1. Comparación del desempeño de los métodos aleatorizados

En los capítulos anteriores y el primer ejemplo de este capítulo se mostró que los parámetros que tienen gran influencia en la calidad de la preimagen son: porcentaje de columnas muestreadas o variables aleatorias generadas, rango de la matriz de aproximación, parámetro σ del kernel gaussiano y el número de vectores propios sobre el que se proyecta . Se desea observar el desempeño de los diferentes métodos de aproximación presentados en el capítulo 3. Para esto, se utilizará un caso especial en el cual se hace una compresión alta del autoencoder y se trabaja con un número reducido de proyecciones.

El valor que se tomó para el parámetro del kernel gaussiano fue $\sigma = 11$. El porcentaje de columnas muestreadas o variables aleatorias generadas fue $2\,\%$ que equivale a $l = 146$ y el rango de

la matriz de aproximación se tomó como $k = 100$, lo cual es menos de la mitad del rango original. Para ambos parámetros del procedimiento lo anterior representa una gran reducción. El número de vectores propios sobre el que se proyecta se tomó como 50. Para el método RFF PCA se tomó $d^* = 50$, el cual se vió en la sección anterior que daba buenos resultados.

A continuación se presentan los resultados obtenidos para los diferentes métodos de aproximación. También se muestra la imagen original que se desea estimar y la preimagen que se obtendría si se hiciera Kernel PCA sin utilizar aproximaciones. Se debe señalar que se realizó el procedimiento con diferentes dígitos y los resultados son similares a los aquí presentados.

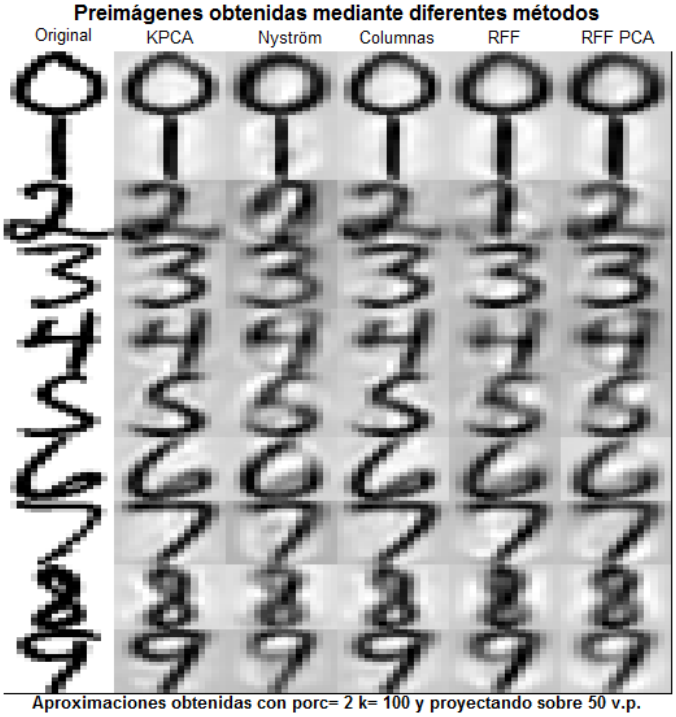

Figura 22: Preimágenes de diferentes dígitos obtenidas a partir de randomized autoencoders.

Se puede observar que aunque las preimágenes han perdido calidad, el desempeño de los métodos es muy bueno. El método de Nyström es el peor de todos ya que no logra rescatar la esencia de todos los dígitos y confunde un poco más el fondo con el dígito. Los métodos de columnas, RFF y RFF PCA tienen un desempeño bastante similar. Sin embargo, por ejemplo para el dígito 2, se obtiene una mejor aproximación con RFF PCA que con RFF. Se debe recalcar que el rango de la matriz de aproximación utilizada para el método RFF PCA es 50, lo cual es la mitad del rango que se utilizó para los demás métodos. Se puede percibir muy poca diferencia entre el desempeño de Kernel PCA y las aproximaciones de su resultado dadas por los diferentes métodos. Lo anterior es muy bueno ya que se obtiene un resultado similar al de KPCA utilizando aproximaciones que implican una gran reducción de costos computacionales.

En los diferentes experimentos realizados se observó que algunos datos eran más difíciles de distinguir debido a su forma, pero los métodos mantuvieron en gran parte la esencia del dígito. En este caso extremo, se puede notar que los métodos de aproximación parecen captar la esencia de cada dígito. Se debe enfatizar que se está utilizando solamente el 1 % del número de columnas para generar la aproximación de K, una matriz de aproximación cuyo rango es menor que la mitad del rango de K y se proyecta sobre el 0.5 % del número de vectores de K.

Observaciones

Se puede decir que todos los métodos de aproximación trabajan bastante bien aún cuando se hace una gran reducción en las variables que afectan bastante la calidad de la imagen. Se obtuvo un desempeño similar para todos los métodos, siendo un poco menos eficiente el método de Nyström.

En cuanto al método RFF y su modificación RFF PCA se observó que cuando se utilizan pocas variables aleatorias generadas para aproximar la matriz de Gram y realizar *randomized autoencoders*, la modificación puede tener un mejor desempeño.

4.3.2.2. Comparación detallada de los métodos RFF y RFF PCA

Uno de los parámetros que se deben elegir cuando se trabaja con el método RFF PCA es el número de vectores propios d^* de la matriz X^tX que se utiliza para generar las direcciones w^*. En este caso, debido a que la matriz X es de rango 256, el máximo número de vectores propios que se pueden utilizar es 256. Cuando se utilizan todos los vectores propios la modificación RFF PCA coincide con RFF. Se desea comparar el desempeño de ambos métodos cuando se realiza *randomized autoencoders* y se varía el parámetro d^*. Por esta razón, se consideraron los siguientes valores para dicho parámetro $d^* = 200, 100, 50$ y 20.

Se sabe que conforme el número de variables aleatorias l que se generan para el método RFF tiende a infinito, se tiene convergencia a la matriz K. Así, intuitivamente si l es muy grande la forma en que se eligen las direcciones w no afecta tanto la aproximación de KPCA obtenida. En cambio, si se generan pocas variables aleatorias sería deseable obtener direcciones w que sí contengan información sobre los datos y ayuden a obtener una aproximación buena. En otras palabras, existe una relación entre l y d^*. Por dicha razón, se decidió variar también el número de variables aleatorias que se genera, tomando este número como un porcentaje del número de columnas de K, como se hizo anteriormente. Los valores que se consideraron para la gráfica que se presenta son: $l = 3\,\%, 5\,\%, 10\,\%$ y $20\,\%$. Se realizaron experimentos con valores mayores de l y lo que se observó es que para porcentajes mayores a 20 % el error que produce el método RFF es menor que su modificación. Lo anterior era de esperarse ya que la modificación está pensada para casos en los que no se pueden elegir muchas direcciones para la proyección, es decir que l es chico.

El parámetro σ que se utilizó para el kernel gaussiano fue $\sigma = 11$, el cual se vió en el Capítulo 2 que funcionaba bien para los datos. El rango se tomó como 200 y el número de vectores de proyección elegido fue 20. El error en la preimagen depende del dígito que se desea aproximar, en la Figura 23 se muestran los diagramas de caja para el error en la preimagen obtenidos para algunas observaciones escogidas al azar, las cuales corresponden al digito cero.

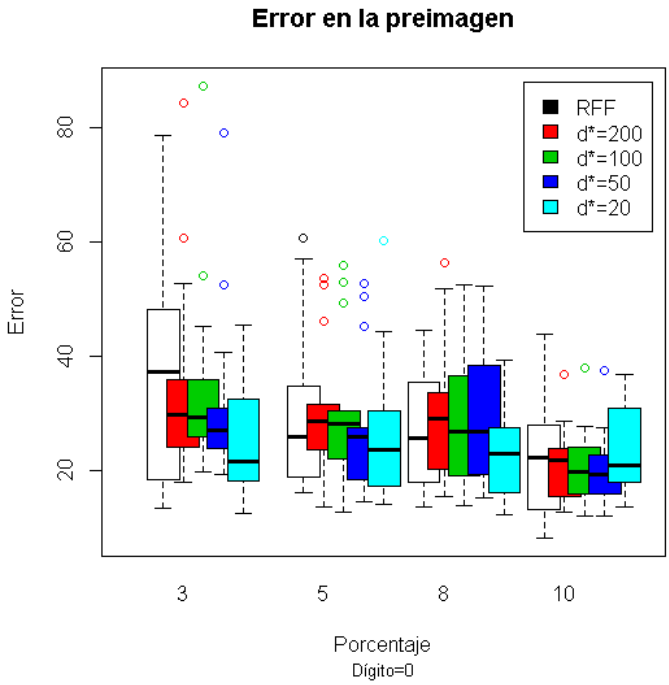

Figura 23: Diagrama de caja para el error en la preimagen obtenida mediante diferentes métodos de aproximación y diferentes valores del número de v.a. generadas para una muestra de los datos correspondientes al dígito cero.

Se debe recalcar que la muestra que se tomó fue de tamaño 20. De las 20 imágenes seleccionadas, se tuvo que para 14 de ellas el algoritmo del punto fijo convergió a una aproximación de la imagen que se estaba aproximando. Para las 6 imágenes restantes, como se observó en el capítulo 2, el método convergió a una aproximación de una imagen que no correspondía a la imagen que se estaba aproximando. Por dicha razón, para la Figura 23, no se tomaron en cuenta estas 6 observaciones.

Se puede observar que para porcentajes menores a 10 %, se obtiene una ganancia con la modificación RFF PCA. Sobretodo, para los casos en que el porcentaje es 3 y 5, se puede observar que el método RFF tiene mayor variabilidada que las modificaciones y además da probabilidad diferente de cero a valores muy grandes para el error. Para estos porcentajes bajos, el método RFF PCA con $d^* = 50$, parece ser más conveniente que los demás ya que a pesar de tener un poco más de variabilidad en algunos casos, considera valores mucho menores para el error. Como se vió anteriormente, en la gráfica se refleja la propiedad de que el método RFF mejora conforme se aumenta el porcentaje.

De los diferentes experimentos realizados para otros dígitos, se observó un comportamiento similar al del dígito aquí presentado para los métodos. Se observó una tendencia de mejora del método RFF PCA en cuanto a la estimación de KPCA, para cuando se generan pocas variables aleatorias. Lo anterior es de gran interés ya que cuando no se pueda generar un gran número de

variables para el algoritmo RFF, lo recomendable sería utilizar la modificación RFF PCA.

Conclusiones

El uso de los métodos aleatorizados produce una gran ventaja computacional cuando se desea analizar una gran cantidad de datos. En base a los experimentos realizados se puede concluir que todos los métodos aproximan bastante bien la matriz K, sus eigenvectores y sus eigenvalores. Sin embargo, el método de Nyström es el mejor en estimar la matriz K y el método de columnas, en estimar los eigenvalores y eigenvectores de K. Los métodos de Nyström y de columnas producen estimaciones bastante buenas, no obstante es necesario trabajar con la matriz K para sacar la muestra. En cambio, los métodos RFF y su modificación RFF PCA, no necesitan trabajar con la matriz de Gram en ningún momento lo cual para algunos casos es muy beneficioso.

La implementación en R [27] tanto de los métodos kernel como de los métodos aleatorizados es realtivamente sencilla. Sin embargo, debe cuidarse mucho la estructura de programación para que el análisis de datos con muchas observaciones no sea muy costoso computacionalmente. Para el caso en que se realiza kernel análisis de componentes principales y se desea obtener la expresión de las proyecciones en el espacio original de los datos, la implementación es más complicada.

La modificación RFF PCA representa una mejora al método RFF cuando se generan pocas variables aleatorias en el algoritmo. Lo anterior da el indicio de que el método RFF puede mejorarse de alguna otra manera en la que se incluya información de la distribución de los datos. Aún existe trabajo estadístico por hacer en esta área de investigación así como en cuanto a la interpretación de los métodos kernel.

A. Métodos Kernel

Una de las propiedades que se utiliza para la demostración de que (14) es solución del problema (13), es la propiedad de reproducción del kernel.

Definición 6 *(Propiedad de reproducción) Sea $k(,)$ un kernel y E un espacio de Hilbert de funciones con producto punto $\langle,\rangle_E$. k es un kernel con la propiedad de reproducción si para cualquier vector x y función $f \in E$, se tiene que:*

$$\langle k(x,\cdot), f(\cdot)\rangle_E = f(x). \tag{67}$$

A partir de funciones kernel con la propiedad de reproducción, se puede construir un espacio de Hilbert, en el cual valga (67). Más acerca de este tema puede verse en Cristianini *et al.* (2004) [7] y Muñiz (2011) [25].

A continuación se presenta un bosquejo de por qué (14) es la solución al problema (13).

Sea $H_0 = \text{Span}\{k(x,),\ x \in X\}$ y $H_o = \text{Span}\{K[x,],\ x \in X\}$el subespacio de H generado por las combinaciones lineales de la función kernel evaluada en los datos. Sea $\|\cdot\|$ la norma definida en este espacio a partir de $\langle\rangle_E$, la cual cuenta con la propiedad de reproducción del kernel. Si se proyecta f sobre el subespacio H_0, se tiene que $f = f_{H_0} + f_{H_0^\perp}$ y por lo tanto:

$$\|f\|^2 = \|f_{H_0}\|^2 + \left\|f_{H_0^\perp}\right\|^2 \geq \|f_H\|^2. \tag{68}$$

Como g es no decreciente entonces

$$g\left(\|f\|^2\right) \geq g\left(\|f_{H_0}\|^2\right). \tag{69}$$

Utilizando (69), propiedades del producto punto y la propiedad de reproducción del kernel se tiene que, para $i \in \{1,\ldots,n\}$:

$$\begin{aligned} f(x_i) &= \langle f, k(X[i,],.)\rangle \\ &= \langle f_{H_0}, k(X[i,],.)\rangle + \left\langle f_{H_0^\perp}, k(X[i,],.)\right\rangle \\ &= \langle f_{H_0}, k(X[i,],.)\rangle \\ &= f_{H_0}(X[i,]). \end{aligned}$$

Es decir $f(X[i,]) = f_{H_0}(X[i,])$ para $i \in \{1,\ldots,n\}$ y por lo tanto:

$$L(f(X[1,]),...,f(X[n,])) = L(f_{H_0}(X[1,]),...,f_{H_0}(X[n,])).$$

En otras palabras, la función L solamente depende de la parte de f que pertenece a H y por (69) se tiene que $g\|f\|^2$ es minimizado por f_{H_0}.

Por lo tanto, (13) es minimizado por una función de la forma:

$$f^* = f_{H_0} = \sum_{i=1}^{n} \alpha_i k(X[i,],.).$$

B. Métodos aleatorizados

B.1. Cotas para el error de aproximación para el método de Nyström

Se presenta una idea de la forma en que se obtienen las cotas para el método de Nyström.

El primer paso es escribir el error de aproximación en función de los vectores propios de W. Se vió que $\widehat{K}^{Nys,\ esp} = A\left(V^{A_1}\right)\left(V^{A_1}\right)^t A^t$, por lo que

$$\left\|K - \widehat{K}_k^{Nys,\ esp}\right\|_{2,F} = \left\|AA^t - A\left(V_k^{A_1}\right)\left(V_k^{A_1}\right)^t A^t\right\|_{2,F},$$

donde $A_1 A_1^t = W$ y por lo tanto $\left(V_k^{A_1}\right)\left(V_k^{A_1}\right)^t A^t$ puede verse como una proyección de A^t sobre los primeros k eigenvectores de W.

En particular, para la norma euclidiana tomando en cuenta que para cualquier matriz B, $\|B\|_2^2 = \|B^t B\|_2$ se tiene que

$$\begin{aligned}\left\|K - \widehat{K}_k^{Nys,\ esp}\right\|_2 &= \left\|AA^t - A\left(V_k^{A_1}\right)\left(V_k^{A_1}\right)^t A^t\right\|_2 \\ &= \left\|A^t - \left(V_k^{A_1}\right)\left(V_k^{A_1}\right)^t A^t\right\|_2^2.\end{aligned}$$

Utilizando propiedades de la descomposición en valores singulares a partir de la expresión anterior Drineas y Mahoney (2005) [9] demostraron que:

$$\left\|K - \widehat{K}_k^{Nys,\ esp}\right\|_2 \leq \left\|K - \widehat{K_k}\right\|_2 + 2\left\|A^t A - C_A C_A^t\right\|_2, \tag{70}$$

donde C_A contiene las columnas muestreadas y reescaladas de A^t. Sea M la matriz que indica si una columna fue elegida en el muestreo o no, es decir $M_{ii} = 1$ si la columna i fue elegida en el muestreo y $M_{ij} = 0$ en otro caso. Así, $C_A = \sqrt{\frac{n}{l}} A^t M$. Cabe destacar que el resultado anterior se cumple para cualquier tipo de muestreo de columnas que se realice simplemente cambiando el factor de reescalamiento $\sqrt{\frac{n}{l}}$.

La idea es acotar el segundo término de la suma (70) para tener una mejor idea del error. Para esto, se utiliza una generalización de la desigualdad de McDiarmid. Esta desigualdad proviene de la llamada concentración de medida en probabilidad. La concentración de medida con respecto a la media se refiere a que el promedio de una función de variables aleatorias independientes no se aleja mucho de su media. Bajo este concepto, lo que se hace es acotar la variable $\|A^t A - C_A C_A^t\|_2$ utilizando una desigualdad de concentración de medida para la esperanza de dicha variable. El teorema es una generalización de la cota de concentración de medida de McDiarmid para el caso de muestreo uniforme sin reemplazo demostrado por Cortes *et al.* (2008) [6].

Teorema 7 *Desigualdad de McDiarmid para muestreo aleatorio simple sin reemplazo. Sea $Z_1, Z_2, ..., Z_l$ una secuencia de variables aleatorias muestreadas uniformemente y sin reemplazo de un conjunto fijo de $l + u$ elementos, y sea $g : Z^l \to R$ una función simétrica tal que para todo $i \in \{1, ..., l\}$ y para*

todo $z_1, ..., z_l \in Z$ *y* $z'_1, ..., z'_l \in Z$, $|g(z_1, ..., z_l) - g(z_1, ..., z'_i, ..., z_l)| \leq c$. *Entonces para todo* $\varepsilon > 0$ *se tiene que*

$$P(g - E(g) \geq \varepsilon) \leq \exp\left(\frac{-2\varepsilon^2}{\frac{lu}{l+u-1/2}\frac{c^2}{1-1/2\max\{l,u\}}}\right).$$

Lo que se hace es aplicar el teorema anterior a la función $g(M) = \|A^tA - C_AC_A^t\|$. Para esto, primero se debe obtener una cota para $\left|g\left(M'\right) - g(M)\right|$ la cual correspondería al valor c del teorema anterior. Kumar *et al.* (2012) [20] obtuvieron la siguiente cota:

$$\left|g\left(M'\right) - g(M)\right| \leq 2\frac{n}{l}\left(\max_{i,j}\|K[,i] - K[,j]\|\right)\left(\max_i K_{ii}\right)^{\frac{1}{2}}. \tag{71}$$

En lo que sigue se denotará por d_K a $\left(\max_{i,j}\|K[,i] - K[,j]\|\right)$ y por $K_{\max}$ a $\left(\max_i K_{ii}\right)$.

Ya que se tiene la cota c del teorema, entonces se acota la esperanza $E(g)$, para que al aplicar el teorema se obtenga una cota en probabilidad para $\|A^tA - C_AC_A^t\|$. Lo que se utiliza son resultados de concentración de medida para la media de funciones que cumplen algunos supuestos de suavidad. Se puede ver en Kumar *et al.* (2009) [19] que la cota que se obtiene para $E(g)$ es

$$E(g) \leq \frac{n}{\sqrt{l}}K_{\max}. \tag{72}$$

Combinando (71), (72) y el teorema de McDiarmid para muestreo aleatorio simple sin reemplazo se llega a que para cualquier tamaño de muestra l se cumple que:

$$P\left(\left\|K - \widehat{K_k}^{Nys}\right\|_2 \leq \left\|K - \widehat{K_k}\right\|_2 + \frac{2n}{\sqrt{l}}K_{\max}\left[\frac{1 + \sqrt{\frac{n-l}{n-1/2}\left(1 - \frac{1}{2\max\{l,n-l\}}\right)\log\frac{1}{\delta}}d_K}{K_{\max}^{\frac{1}{2}}}\right]\right) \geq 1 - \delta.$$

De forma similar, Kumar *et al.* (2012) [20] obtuvieron la siguiente cota para el error de aproximación con respecto a la norma de Frobenius:

$$P\left(\left\|K - \widehat{K_k}^{Nys}\right\|_F \leq \left\|K - \widehat{K_k}\right\|_F + \left(\frac{64k}{l}\right)^{\frac{1}{4}}nK_{\max}\left[\frac{1 + \sqrt{\frac{n-l}{n-1/2}\left(1 - \frac{1}{2\max\{l,n-l\}}\right)\log\frac{1}{\delta}}d_K}{K_{\max}^{\frac{1}{2}}}\right]^{\frac{1}{2}}\right) \geq 1 - \delta.$$

B.2. Obtención de la proyeccion aleatoria z_θ para el método RFF

Se presenta la demostración de que

$$z_\theta(x) = \sqrt{2}\cos\left(w^t x + b\right)$$

donde $\theta = (w, b)$, w se genera con densidad p y b se genera de manera independiente de una distribución uniforme en $[0, 2\pi]$, dada en Muñiz (2011) [25].

Utilizando las propiedades trigonométricas del seno y coseno para la proyección aleatoria $z_\theta(\cdot)$ se tiene que

$$\begin{aligned} z_\theta(x)\, z_\theta(y) &= \sqrt{2}\cos\left(w^t x + b\right)\sqrt{2}\cos\left(w^t y + b\right) \\ &= 2\left[\cos\left(w^t x\right)\cos(b) - \sin\left(w^t x\right)\sin(b)\right]\left[\cos\left(w^t y\right)\cos(b) - \sin\left(w^t y\right)\sin(b)\right] \\ &= 2[\cos\left(w^t x\right)\cos\left(w^t y\right)\cos(b)^2 - \sin\left(w^t x\right)\sin(b)\cos\left(w^t y\right)\cos(b) - \\ &\quad - \cos\left(w^t x\right)\cos(b)\sin\left(w^t y\right)\sin(b) + \sin\left(w^t x\right)\sin\left(w^t y\right)\sin(b)^2]. \end{aligned}$$

Si w y b se generan de manera independiente entonces $E_\theta\left(z_\theta(x)\, z_\theta(y)\right) = E_w\left[E_b\left[z_\theta(x)\, z_\theta(y)\right]\right]$, por lo que se calcula la esperanza sobre b para dejar una esperanza en términos de w.

$$\begin{aligned} E_b\left[z_\theta(x)\, z_\theta(y)\right] &= E_b\left[\begin{array}{l} 2[\cos(w^t x)\cos(w^t y)\cos(b)^2 - \sin(w^t x)\sin(b)\cos(w^t y)\cos(b) + \\ \quad - \cos(w^t x)\cos(b)\sin(w^t y)\sin(b) + \sin(w^t x)\sin(w^t y)\sin(b)^2] \end{array}\right] \\ &= 2\cos\left(w^t x\right)\cos\left(w^t y\right)E_b\left(\cos(b)^2\right) - 2\sin\left(w^t x\right)\cos\left(w^t y\right)E_b\left[\sin(b)\cos(b)\right] - \\ &\quad -2\cos\left(w^t x\right)\sin\left(w^t y\right)E_b\left[\cos(b)\sin(b)\right] + \sin\left(w^t x\right)\sin\left(w^t y\right)E_b\left[\sin(b)^2\right]. \end{aligned}$$

Integrando por partes se tiene que

$$E_b\left[\cos(b)^2\right] = \int_0^{2\pi} \frac{1}{2\pi}\cos(b)^2\, db = \frac{1}{2},$$

$$E_b\left[\sin(b)^2\right] = E_b\left[1 - \cos(b)^2\right] = \frac{1}{2}$$

y

$$E_b\left[\sin(b)\cos(b)\right] = \int_0^{2\pi} \frac{1}{2\pi}\sin(b)\cos(b)\, db = 0.$$

Con esto,

$$\begin{aligned} E_b\left[z_\theta(x)\, z_\theta(y)\right] &= \cos\left(w^t x\right)\cos\left(w^t y\right) + \sin\left(w^t x\right)\sin\left(w^t y\right) \\ &= \cos\left[(x-y)^t w\right]. \end{aligned}$$

Sustituyendo en la expresión para la esperanza sobre θ se tiene que

$$\begin{aligned} E_\theta\left(z_\theta(x)\, z_\theta(y)\right) &= E_w\left[E_b\left[z_\theta(x)\, z_\theta(y)\right]\right] \\ &= E_w\left[\cos\left[(x-y)^t w\right]\right], \end{aligned}$$

que es la expresión que se tenía para el kernel en la ecuación (52). Por lo tanto,

$$\begin{aligned} k(x,y) &= E_w\left[\cos\left[(x-y)^t w\right]\right] \\ &= E_\theta\left(z_\theta(x) z_\theta(y)\right). \end{aligned}$$

B.3. Convergencia uniforme del Fourier Features

Lo primero que se hace es definir la función $f(x_k,x_j) = \overrightarrow{z_\theta}(x_k)^t \overrightarrow{z_\theta}(x_j) - k(x_k,x_j)$. Por construcción se tiene que $|f(x_k,x_j)| \leq 2$ y que $E(f(x_k,x_j)) = 0$ para todo par (x_k,x_j). Se tiene que el espacio M donde viven los $\{x_i\}_{i=1}^n$ es compacto y por lo tanto se puede cubrir M con a lo más $T = (4diam(M)/r)^d$ bolas de radio r. Se denota por $\{\Delta_i\}_{i=1}^T$ los centros de las bolas. Sea L el supremo del valor absoluto de la derivada de f con respecto a x. L es conocida como la constante de Lipschitz de f y su interpretación es la de la cota más chica que limita qué tan rápido la función f puede cambiar. Se tiene que $|f(x_k,x_j)| \leq \in$ para todos los $(x_k,x_j) \in M$ siempre y cuando sucedan los eventos $|f(\Delta_i)| \leq \frac{\in}{2}$ y $L \leq \frac{\in}{2r}$ para todo i , por lo que se acota la probabilidad de dichos eventos.

Para acotar el evento $L \leq \frac{\in}{2r}$ se utiliza el hecho de que f es diferenciable y la linealidad del valor esperado para obtener que $E(L^2) \leq E_p[w^t w] = \sigma_p^2$. Utilizando la desigualdad de Markov se tiene que

$$P\left(L \geq \frac{\in}{2r}\right) \leq \left(\frac{2r\sigma_p}{\in}\right)^2. \tag{73}$$

Por la desigualdad de Boole o la cota de la unión se tiene que

$$P\left(\cup_{i=1}^T |f(\Delta_i)| \geq \frac{\varepsilon}{2}\right) \leq \sum_{i=1}^T P\left(|f(\Delta_i)| \geq \frac{\varepsilon}{2}\right).$$

Utilizando la desigualdad de Hoeffding se tiene que

$$P\left(|f(\Delta_i)| \geq \frac{\varepsilon}{2}\right) \leq 2\exp\left(-\frac{l\varepsilon^2}{8}\right),$$

por lo que

$$P\left(\cup_{i=1}^T |f(\Delta_i)| \geq \frac{\varepsilon}{2}\right) \leq 2T\exp\left(-\frac{l\varepsilon^2}{8}\right). \tag{74}$$

Utilizando (73) y (74) se tiene que

$$\begin{aligned} P\left(\sup_{(x_k,x_j)\in M}|f(x_k,x_j)|\leq\varepsilon\right) &= P\left(|f(\Delta_i)|\leq\frac{\varepsilon}{2}\right)+P\left(L\leq\frac{\varepsilon}{2r}\right) \\ &= 1-P\left(|f(\Delta_i)|\geq\frac{\varepsilon}{2}\right)-P\left(L\geq\frac{\varepsilon}{2r}\right) \\ &\geq 1-\left(\frac{2r\sigma_p}{\varepsilon}\right)^2-2T\exp\left(-\frac{l\varepsilon^2}{8}\right) \\ &= 1-\left(\frac{2r\sigma_p}{\varepsilon}\right)^2-2\left(\frac{4\text{diam}(M)}{r}\right)^d\exp\left(-\frac{l\varepsilon^2}{8}\right). \end{aligned}$$

Lo cual da una cota en función de r. Eligiendo $r = \left(2\left(4\text{diam}(M)\right)^d\exp\left(-\frac{l\varepsilon^2}{8}\right)/\left(\frac{2\sigma_p}{\varepsilon}\right)^2\right)$ y asumiendo que $(\sigma_p\text{diam}(M)/\varepsilon)\geq 1$ y que $\text{diam}(M)\geq 1$ se obtiene la cota deseada 55.

A grandes rasgos, se puede decir que lo que se ocupa es una construcción inteligente f y resultados de concentración de medida de variables aleatorias con respecto a su media.

Referencias

[1] Bingham, E. y Heikki M. (2001), "Random Projection in dimensionality reduction: applications to imagen and text data", *Proceedings of the Seventeenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* , 245-250.

[2] Boutsidis C., Sun J. y Anerousis N. (2008), "Clustered subset selection and its applications on it service metrics", *Proceedings of the Seventeenth ACM CIKM Conference on Information and Knowledge Management,* 599–608.

[3] Boutsidis C., Drineas P. y Mahoney M. (2009), "An improved approximation algorithm for the column subset selection problem", *AMC SIAM Symposium on Discrete Algorithms.*

[4] Boutsidis C., Drineas P. y Magdon-Ismail M. (2013),"Near optimal column based matrix reconstruction",*SIAM Journal on Computing, special issue of Symposium on Foundations of Computer Science, 2011.*

[5] Chitta R., Jin R. y Jain A. K.(2012), "Efficient kernel clustering using random Fourier features", *IEEE International Conference on Data Mining*, 161–170.

[6] Cortes C., Mohri M., Pechyony D y Rastogi A. (2008), "Stability of transductive regression algorithms", *International Conference on Machine Learning.*

[7] Cristianini N. y Shawe-Taylor J. (2004), *Kernel Methods for Pattern Analysis*, Cambridge University Press.

[8] Dasgupta S. y Gupta A. (2003), "An elementary proof of a theorem of Johnson and Lindenstrauss", *Random Structures & Algorithms,* 22,1, 60–65.

[9] Drineas P. y Mahoney M. (2005), "On the Nyström method for approximating a Gram matrix for improved kernel.based learning", *Journal of Machine Learning Research*, 6, 2153-2175.

[10] Drineas P., Mahoney M. y Muthukrishnan S. (2006), "Subspace sampling and relative error matrix approximation: Column-row based methods", *Proceedings of the fourteenth Annual* European Symposium on Algorithms

[11] Elgohary A. Farahat A., Ghodsi A. y Kamel M.(2013), "Greedy Column Subset Selection for Large-scale Data Sets", arXiv:1312.6838.

[12] Friedman J., Hastie T. y Tibshirani R.(2001), *The Elements of Statistical Learning*, Springer series in statistics.

[13] Frieze A., Kannan R. y Santosh V. (1998), "Fast Monte-Carlo algorithms for finding low rank approximations", *Foundation of Computer Science.*

[14] Ghahramani Z., Lopez-Paz D., Schölkopf. B., Smola A. y Sra S. (2014), "Randomized nonlinear component analysis", *International Conference on Machine Learning*, arXiv:1402.0119.

[15] Halko N., Martinsson P. y Tropp J. (2011), "Finding structure with randomness: Stochastic algorithms for constructing approximate matrix decompositions", *SIAM Review 53,* 2, 217-288.

[16] Izenman A. (2008), *Modern multivariate statistical techniques: regression, classification and manifold learning*, Springer Verlag, Springer Texts in Statistics.

[17] Jin R., Li Y., Mahdavi M., Yang T. y Zhou Z. (2012), "Nyström method vs Random Fourier Features: a theoretical and empirical comparison", *Neural Information Processing Systems*, 485–493.

[18] Kumar S., Mohri M., Rowley H. y Talwalkar A. (2013), "Large-scale SVD and manifold learning", *Journal of Machine Learning Research*, 14, 3129-3152.

[19] Kumar S., Mohri M. y Talwalkar A. (2009), "Sampling techniques for the Nyström method", *Conference on Artificial Intelligence and Statistics*, 2009a.

[20] Kumar S., Mohri M. y Talwalkar A. (2012), "Sampling methods for the Nyström Method", *Journal of Machine Learning Research,* 13, 981-1006.

[21] Mahoney M. (2012), "Randomized algorithms for matrices and data", *Advances in Machine Learning and Data Mining for Astronomy*, 647-672.

[22] Martin C. y Porter M. (2011), "The extraordinary SVD", *Proceedings of the 11th International Conference on Intelligent Systems Design and Applications*, 587-592.

[23] Mika S., Müller K., Ratsch G., Schölkopf. B., Scholz M. y Smola A. (1999), "Kernel PCA and de-noising in feature spaces", *Neural Information Processing Systems,* 11, 536-542.

[24] Mohri M. y Talwalkar A. (2010), "On the estimation of coherence", *Journal of Machine Learning Research*, arXiv:1009.0861.

[25] Muñiz, V. (2011), Tesis de Doctorado: "Métodos de clasificación y exploración para datos complejos", CIMAT.

[26] Nyström E. (1928), "Über die praktische auflösung von linearen integralgleichungen mit anwendungen auf randwertaufgaben der potentialtheorie", *Commentationes Physico-Mathematicae*, 4(15), 1–52.

[27] R Core Team (2014), R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

[28] Rahimi A. y Recht B. (2007), "Random features for large-scale kernel machines", *Neural Information Processing Systems*, 1177–1184.

[29] Rostamizadeh A. y Talwalkar A. (2010), "Matrix coherence and the Nyström method", *Proceedings of the 26th Conference in Uncertainty in Artificial Intelligence.*

[30] Rudin, W. (1994), *Fourier analysis on groups*, Wiley Classics Library, Wiley Intersciencie, 36.

[31] Seeger M. y Williams C. (2000), "Using the Nyström method for speed up kernel machines", *Neural Information Processing Systems,* 682-688.

[32] Wahba G. (1990), *Spline models for obervational data*, SIAM.

[33] Wang Q. (2012), "Kernel principal component analysis and its applications in face recognition and active shape models", arXiv:1207.3538.

[34] http://yann.lecun.com/exdb/mnist/