UNIVERSITAT JAUME I

Grado en Matemática Computacional

Trabajo Final de Grado

Espacios de Hilbert con Núcleo Reproductor

Autor:
Olga Girona Cutillas

Tutor académico:
Jorge Galindo Pastor

Fecha de lectura: Julio de 2023
Curso académico 2022/2023

Resumen

A lo largo de este trabajo de fin de grado nos centraremos en el estudio de los espacios de Hilbert con núcleo reproductor y la aplicación de Máquinas de Vectores de Soporte (SVM).El trabajo comienza con una introducción al algoritmo SVM, proporcionando una visión más general de los objetivos que se abordarán a lo largo del trabajo.

A continuación, se hará una introducción a los espacios de Hilbert, incluyendo concpetos fundamentales como el producto interno, ortogonalidad, teorema de Riesz. Además se estudiarán los espacios de Hilbert con núcleo reproductor, centrándonos en el método del núcleo, la separabilidad entre hiperplanos y el concepto de hiperplano de margen máximo.

Por último, se aplica a un caso práctico real el algoritmo SVM utilizando el lenguaje de programación Python. El objetivo es resolver un problema de clasificación en el que se predente determinar si un estudiante se graduará o abandonará sus estudios.

Palabras clave

Espacios de Hilbert, Hiperplano de margen máximo, Núcleo reproductor, Método del núcleo, Máquinas de Vector de Soporte.

Keywords

Hilbert Spaces, Maximal margin hyperplane, Reproducing Kernel,Kernel method, Support Vector Machines (SVM).

Agradecimientos

En primer lugar, quería agradecer a mi tutor Jorge Galindo Pastor por tener la paciencia de enseñarme y guiarme para desarrollar el trabajo que pondría fin a la carrera que he estado estudiando durante estos cuatro años. Además, de agradecer a María Victoria Ibáñez Gual por guiarme a desarrollar parte del trabajo. En definitiva, gracias a los dos por hacer que no me resulte imposible un trabjo que he temido desde que empecé la carrera.

Gracias a mis amigos de allí por haber aguantado mis dramas 'Matemáticos' aunque no entendierais nada. Pero sobre todo, gracias por estar ahí desde el principio, y hacerme sentir que vuelvo a lo de siempre cada vez que volvía.

Ahora, quiero agradecer a toda mi familia que ha hecho que durante estos cuatro años sienta que soy capaz de esto y más. Gracias por ser mi soporte y mi refuerzo cada día. Gracias a mi padre también y gracias a mi hermano, por haberme enseñado que aunque las cosas no salgan como lo establecido siempre salen.

Por otra parte, quiero agradecer a toda la gente que he conocido en Castellón y que ha hecho que a pesar de estar a 282 km de mi casa, sienta que estoy en ella. Y, gracias por hacer que venir a Castellón sea el mejor error que he podido cometer en mi vida. Gracias por preocuparos y reir conmigo como si fuera de vuestra familia.

Lorena y Paula, gracias por darme la oportunidad de vivir la experiencia de vivir con amigas. Y, gracias a Marta y Adriana por haber sido las mejores vecinas que por casualidad nos pudimos encontrar.

Gracias también 'a los que no iban a salir', a todos y cada uno de vosotros. Gracias por aportarme la parte de risas, de fiestas, cervecitas... Y, sobre todo, gracias por haberme creado el mejor recuerdo de estos cuatro años, siempre nos quedará otro crucero...

Sobre todo, agradecer a Lola, sinceramente, no sé si este TFG lo hubiera terminado sin ti. No sé que habría pasado si no hubiera tenido todos los días el consuelo de que iba a poder tener

momento roble y, por supuesto, cervecita y olivas.

Gracias también a Iván, quien iba a decir que un Kahoot podía hacer nuevas amistades y no enemigos. Gracias por estar ahí siempre, por estar pendiente de que estuviera bien, y ser esa persona que pasaba de mi parte seca y darme abrazos cuando más lo necesitaba.

Ahora quiero agradecer a Clara. Gracias por haber confiado en mi para todo, espero haberte ayudado aunque sea un 1 % de lo que tú me has ayudado estos años a mí. Gracias por ser mi compañera de llanto estos años, y aguantar todos y cada uno de mis dramas que no eran pocos.

También, agradecer a Júlia. Gracias por ser compañera, amiga y hermana. Gracias por todos los trabajos de 1 % trabajo y 99 % chisme, y gracias por haber sido mi compañera de vida estos cuatro años. Gracias por enseñarme tanto a nivel personal, y hacerme ver que el mundo no es un drama continuo. Gracias por ser mi casa, por demostrarme que el dicho 'los amigos son la familia que escogemos' es real.

Pero sobre todo, agradecer a mi madre. Gracias por confiar en mi desde el principio, por haber sido mi pilar y mi empuje siempre, por enseñarme que las cosas siempre pueden salir con mayor o menor esfuerzo. Gracias por todo el esfuerzo que has hecho durante toda mi vida para que hoy en día yo esté donde estoy, y sobre todo ser quien soy. Yo habré estudiado y aprobado los examenes, pero este logro no es solo mío, es de las dos.

Por último, agradecer a mis 4 abuelos. Por estar siempre, y por ofrecerme comida y refugio cada vez que volvía. Gracias a todos por cogerme el teléfono cada vez que llamaba, y por escuchar mis cosas e interesarse aunque no entendierais mucho. Y sí, al final, lo he conseguido abuelo, esto va por ti.

Índice general

Índice de figuras

Capítulo 1

Introducción a SVM

Para el desarrollo de las siguientes secciones haré uso de las referencias: [13] y [3]. Además, de usar las referencias [10] y [11] para el cumplimiento de la información.

1.1. Separación mediante hiperplanos

En esta sección nos basaremos en la idea de la separación de hiperplanos. La idea principal es hallar una separación lineal que divida los datos en dos o más clases. En el caso más sencillo, que veremos a lo largo de este capítulo, consideramos datos bidimensionales que están separados perfectamente en dos clases y buscamos un hiperplano separador.

Figura 1.1: Gráfica de datos linealmente separables (caso 1).

En general, si los datos son perfectamente linealmente separables, existirán infinitos hiperplanos que puedan clasificar el conjunto de datos en dos clases distintas (Ver en Fígura 1.2).

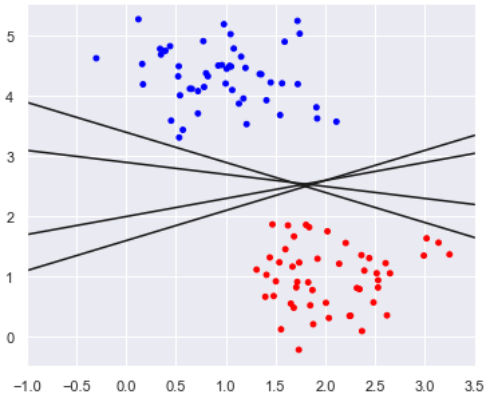


Figura 1.2: Generación de hiperplanos (caso 1).

Sin embargo, en esta sección, nos centramos en elegir el mejor hiperplano, al que se le conoce como *hiperplano de margen máximo*.

La definición del *margen*, y por tanto, del hiperplano de margen máximo, no depende directamente del conjunto de datos, sino de unos vectores a los que se conocen como *vectores de soporte*. Estos, forman parte del conjunto de datos que se quiere clasificar, y son aquellos que se encuentran más cercanos al hiperplano de separación, y que además, tienen una influencia directa en la orientación y posición del hiperplano (ver Figura 1.3).

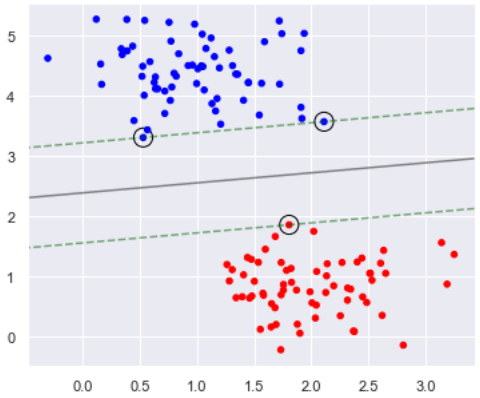


Figura 1.3: Visualización del hiperplano de margen máximo y los márgenes y vectores de soporte.

De esta manera, en caso de que se produzca alguna modificación en el conjunto de datos, así como añadir nuevos datos o eliminar, el hiperplano de margen máximo no sufrirá ninguna modificación.

Aunque la idea de hallar un hiperplano que separe perfectamente los datos en dos clases suena ideal cuando este conjunto de datos es perfectamente linealmente separables, en la práctica rara vez se encuentra un conjunto de datos que cumplan esta condición. Esto significa a que a menudo resulta difícil, e incluso imposible, hallar ese hiperplano que clasifique los datos de manera óptima.

Además, existe el riesgo de caer en problemas de *overfitting*, donde el hiperplano separador se ajusta excesivamente a unas características específicas del conjunto de datos impidiendo que se pueda generalizar a nuevos datos.

1.2. Clasificador de vectores de Soporte

Para resolver el problema comentado en la sección anterior, ampliaremos el concepto de hiperplano separador con el fin de ser capaz de hallar un hiperplano que logre separar lo mejor posible el conjunto de datos en dos clases distintos, aunque no perfectamente. Para lograr esto, utilizaremos el *margen blando* mediante el *Clasificador de vectores de soporte* o *Clasificador de margen suave*.

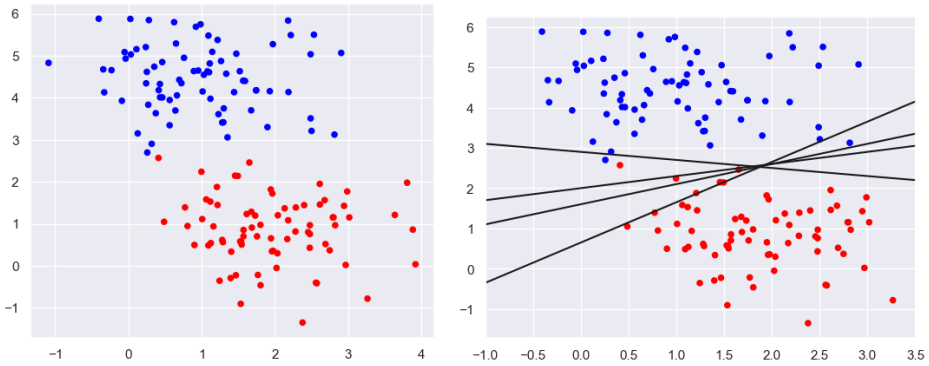


Figura 1.4: Conjunto de datos linealmente no separables.

La idea de esta técnica es clasificar los datos teniendo en cuenta que algunas observaciones pueden estar en el lado incorrecto del hiperplano, e incluso en el espacio que hay entre los vectores de soporte y el hiperplano. Es decir, se permite una clasificación errónea de algunas observaciones del conjunto.

El enfoque del clasificador de vectores de soporte permite encontrar un equilibrio entre la separación óptima de los datos y la tolerancia de algunos errores de la clasificación. De esta manera, se consigue una mayor felixibilidad y capacidad de generalización en modelo de clasificación, evitando sobreajustes.

Para ello, se puede ajustar la 'dureza' de este margen mediante un parámetro que denotaremos como $\mathcal{C}$ (ver Figura 1.5). Este permite controlar ese equilibrio que hemos mencionado.

Cuando este parámetro tiene un valor alto (, resulta en un margen más rígido, por lo que se impone una mayor penalización por cada error de clasificación. Esto puede llevar a un mejor rendimiento en el conjunto de entrenamiento, a pesar de que puede tener problemas de sobreajuste y para generalizar a nuevos datos.

Por otro lado, si tiene un valor bajo, encontramos un margen más suave, se permite un margen más amplio, y por tanto, se tolerarán más los errores de clasificación. De esta manera, resulta en un modelo con más flexibilidad para clasificar las generalizaciones. Sin embargo, baja el rendimiento del modelo al permitir mayor número de errores.

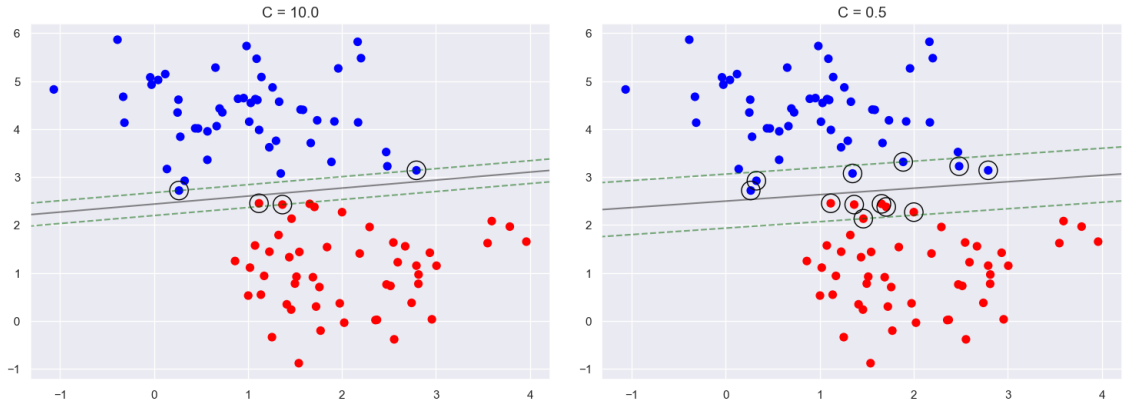


Figura 1.5: Hiperplanos de margen máximo según el parámetro C

1.3. Máquinas de Vectores de Soporte

Las *máquinas de vectores de soporte (SVM)* es una extensión del clasificador de vectores de soporte que resulta de ampliar varias dimensiones, e incluso a dimensión infinita, el espacio de características mediante el uso de *kernels*. Se utiliza en casos donde es imposible encontrar un hiperplano separador, como el que vemos en la figura 1.6.

Un *kernel* es una función que permite calcular el producto escalar entre dos vectores en un espacio de características de mayor dimensión. En otras palabras, el kernel define una medida de solicitud o distancia entre dos vectores en un espacio de carracteríticas de mayor dimensión.

Por lo que, el uso de estos es crucial para abordar problemas de clasificiación en el que

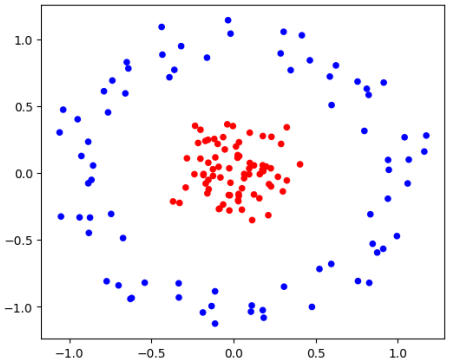


Figura 1.6: Datos no separables

los datos no son linealmente separables en el espacio original, ya que permite realizar una trasnformación implícita de los datos a un espacio de características de dimensionalidad superior donde los datos sí lo sean.

La ventaja de utilizar kernels es que nos permite evitar el cálculo explícito de las coordenadas de los vectores en el espacio de mayor dimensión. En cambio, podemos realizar los cálculos necesarios en el espacio original utilizando esta función. Esto resulta en una considerable reducción en el coste computacional y nos permite aplicar transformaciones no lineales sin necesidad de conocer explícitamente dichas tranformaciones en el espacio de mayor dimensión.

Existen diferentes tipos de *kernels* y su elección puede marcar la diferencia entre un modelo SVM efectivo y uno que no se ajuste correctamente a los datos. En nuestro caso, nos centraremos en los siguientes:

- **Kernel lineal:** Este es el más simple de todos y es altamente utilizado cuando los datos son linealmente separables en el espacio original. Este no realiza una transformación explícita de los datos, sino que calcula el producto interno entre los datos en el espacio original, lo que permite definir una similitud entre ellos. De esta manera, poder definir la posición del hiperplano. El kernel lineal se define como:

  $$K(x, y) = v^T x + c = \langle v, x \rangle + c$$

  donde v y c son parámetros aprendidos durante el entrenamiento del SVM, y serán aquellos que ayudar a determinar la dirección y posición del hiperplano de margen máximo respectivamente.

- **El kernel Gaussiano:** Se trata de uno de los kernels más utilizados en el algoritmo SVM. Y es realmente útil cuando los datos tienen una estructura compleja. Este kernel

en particular, se define como:

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

siendo $\|\cdot\|$ la norma Euclidia y σ un parámetro que regula la dispersión.

- **Kernel polinómico:** Se define como:

$$K(x, y) = (x^T y + c)^d$$

donde $c \geq 0$. Cuando se emplea $d = 1$ y $c = 0$, es lo mismo que emplear el kernel lineal: $K(x, y) = x^T y$

A medida que va aumentando el valor de d, se aumenta la no linealidad de los límites de decision. Por lo que no es recomendable utilizar valores elevados para la variable d.

A continuación, se va a mostrar un ejemplo en el que se aplica el *Kernel Gaussiano* para hallar un hiperplano separador en el espacio original. Y, como he comentado anteriormente el aumento de dimensión no se hace explícitamente, por lo que representar el espacio con la dimensión aumentada no es posible, ya que puede llegar hasta dimensión infinita.

Y ahora, vemos que sí que es linealmente separables en 2D:

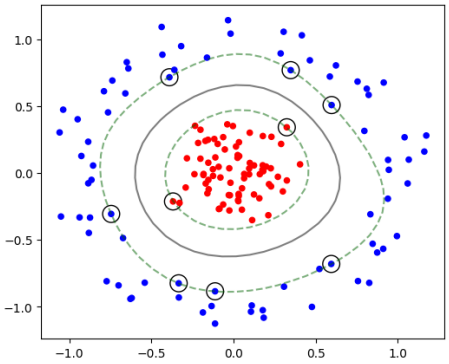


Figura 1.7: Hiperplano separado utilizando el Kernel Gaussiano

Capítulo 2

Introducción a los espacios de Hilbert

A lo largo de este capítulo se introducirán aquellos conceptos teóricos que serán necesarios para poder comprender correctamente todo el fundamento teórico contenido a lo largo de todo el trabajo. Se profundizará en el significado de los espacios de Hilbert y se explorará su utilidad.

Para desarrollar todo este capítulo usaremos las referencias: [4], [14]

2.1. Espacios de Banach

A lo largo de esta sección, profundizaremos en los espacios normados. Las normas permiten medir convenientemente distancias entre datos.

En la sección siguient veremos que las normas inducidas por una estructura adicional, el producto interno, introducen en el espacio de los datos una estructura geométrica muy útil.

Definición 2.1.1 (**Norma**)**.** Sea x un espacio vectorial de $\mathbb{R}$. Se define la *norma de* X como la función $\|\cdot\| : X \longrightarrow \mathbb{R}$ tal que dados los vectores $x, y \in X$ y el escalar $\alpha \in \mathbb{R}$, se cumple:

- $\|x\| \geq 0$.
- $\|x\| = 0$ si y solo si $v = 0$.
- $\|\alpha \cdot x\| = |\alpha| \cdot \|x\|$.
- Desigualdad triangular: $\|x + y\| \leq \|x\| + \|y\|$.

A un espacio vectorial X junto con la norma $\|\cdot\|$ se le denomina espacio normado.

Definición 2.1.2 (**Espacio acotado**)**.** Sea un espacio normado $(X, \|\cdot\|)$. Se dice que un subconjunto A de ese espacio es *acotado* si $\exists M$, tal que:

$$\|x\| \leq M, \ \ \forall x \in A.$$

Asociado a toda norma hay una distancia que se define como: $d(x, y) = \|x - y\|$. Esto nos permite utilizar los conceptos de sucesiones de Cauchy, sucesiones convergentes, y espacio completo.

Definición 2.1.3 (**Espacio de Banach**)**.** Un espacio normado X es un espacio de Banach si es completo, es decir, si toda sucesión de Cauchy en X converge a un elemento de X.

2.2. Productos Internos y Espacios de Hilbert

A lo largo de esta sección, se explorará en detalle los conceptos de *producto interno* y *ortogonalidad*, los cuales son fundamentales en el estudio de espacios de Hilbert.

La introducción de productos internos permite dotar de una estructura geométrica a los espacio de Banach lo que nos permitirá establecer relaciones de perpendicularidad entre vectores, así como definir conceptos como la proyección y componentes ortogonales.

2.2.1. Producto Interno

El concepto de producto interno es una generalización del producto escalar clásico en espacios euclídeos. La peculiaridad del producto interno, es que este nos permite trabajar en espacios generales más grandes, incluyendo espacios vectoriales complejos, donde las coordenadas de los vectores que lo componen son números complejos.

Definición 2.2.1 (**Producto interno**)**.** Sea $\mathcal{H}$ un espacio vectorial. Un *producto interno* en $\mathcal{H}$ es una función escalar $\langle \cdot, \cdot \rangle$ en $\mathcal{H} \times \mathcal{H}$ con las siguientes propiedades:

- Nunca es negativo: $\langle x, x \rangle \geq 0$, $\forall x \in \mathcal{H}$.
- Simetría conjugada: $\langle x, y \rangle = \overline{\langle y, x \rangle}$, $\forall x, y \in \mathcal{H}$.
- Linealidad en su primera variable: $\langle ax + by, z \rangle = \mathrm{a}\langle x, z \rangle + \mathrm{b}\langle y, z \rangle$, $\forall x, y, z \in \mathcal{H}$.
- Unicidad $\langle x, x \rangle = 0$ si y solo si $x = 0$, $\forall x \in \mathcal{H}$.

En este caso, a $\mathcal{H}$ se le conoce como un *espacio con producto interno.*

En particular, el producto interno $\langle \cdot, \cdot \rangle$ siempre define una norma, la cual está dada por la siguiente fórmula:

$$\|x\| = \langle x, x \rangle^{\frac{1}{2}}$$

Además, esta norma también define una distancia entre los productos del espacio.

$$d(x, y) = \langle x - y, x - y \rangle^{\frac{1}{2}}$$

Esta conexión entre el producto interno, la norma y la distancia nos permite medir la similitud, longitud y distancia entre vectores del espacio de Hilbert. Además, de permitir definir conceptos como la convergencia de sucesiones.

Teorema 2.2.2 (**Desigualdad de Cauchy-Bunyakowski-Schwarz**)**.** Dado el producto interno $\langle \cdot, \cdot \rangle$ en un espacio vectorial X, entonces:

$$|\langle x, y \rangle| \leq \|x\| \|y\|, \;\; x, y \in X$$

Corolario 2.2.3. Si $\mathcal{H}$ es un espacio de producto interno, entonces se cumple lo siguiente:

- Continuidad del producto interno. Si x_n converge a x, e y_n converge a y en $\mathcal{H}$, entonces: $\langle x_n, y_n \rangle$ converge a $\langle x, y \rangle$.
- Si la serie $\sum_{n=1}^{\infty} x_n$ converge en $\mathcal{H}$, entonces:

$$\langle \sum_{n=1}^{\infty} x_n, y \rangle = \sum_{n=1}^{\infty} \langle x_n, y \rangle, \;\; y \in \mathcal{H}.$$

A continuación vamos a ver algunas propiedades importantes del producto interno.

- *Ley del paralelogramo*:

$$\|x + y\|^2 + \|x - y\|^2 = 2(\|x\|^2 + \|y\|^2), \;\; x, y \in X.$$

- *Identidad de Polarización*:

$$\|x + y\|^2 = \|x\|^2 + 2Re\langle x, y \rangle + \|y\|^2, \;\; x, y \in X.$$

Como consecuencia directa de la Ley del paralegramo tenemos que se cumple el *Teorema de Pitágoras.* Que dados $x, y \in X$ ortogonales se cumple que $\langle x, y \rangle = 0$, y por tanto:

$$\|x \pm y\|^2 = \|x\|^2 + \|y\|^2, \;\; x, y \in X.$$

Definición 2.2.4 (**Espacio de Hilbert**)**.** Se dice que un espacio con producto interno (H, $\langle \cdot, \cdot \rangle$) es un *espacio de Hilbert* cuando (H, $\|\cdot\|$) es un espacio de Banach para la norma inducida.

Ahora, vamos a ver algunos ejemplos tanto de espacios que sí cumplen las condiciones para ser espacios de Hilbert como de espacios que no las cumplen.

Ejemplo 2.2.5 (ℓ_n^2)**.** El espacio ℓ_n^2 es un espacio de Hilbert de dimensión finita n, se define de la siguiente manera:

$$\ell_n^2 = \{(x_1, \cdots, x_n) : x_i \in \mathbb{C}\} = \mathbb{C}^n$$

Este espacio se convierte en un espacio de Hilbert con el siguiente producto interno. Dados x, $y \in \ell_n^2$, tenemos que:

$$\langle x, y \rangle = \sum_{i=1}^{n} x_i \cdot \overline{yi}, \;\; x_i, y_i \in \ell_n^2$$

La norma inducida es la siguiente:

$$\|x\|_2 = \left(\sum_{i=1}^{n} |x_i^2| \right)^{1/2}$$

Por lo que es un espacio normado dotado de un producto interno.

Además se trata de un espacio completo, ya que se puede demostrar que toda sucesión de Cauchy de ℓ_n^2 converge a un elemento del mismo espacio.

Ejemplo 2.2.6 (ℓ^2)**.** El espacio ℓ^2 es un espacio de Hilbert de dimesión infinita. Se define igual que el ejemplo anterior, pero teniendo en cuenta que ahora no es finito.

$$\ell^2 = \{(x_n)_n : \sum_{i=1}^{\infty} |x_i|^2 < +\infty \, , x_n \in \mathbb{C}\}$$

Al igual que antes, este espacio se convierte en un espacio de Hilbert al introducir la estructura de producto interno. En este caso, se define de manera similar al caso de ℓ^2:

$$\langle x, y \rangle = \sum_{i=1}^{\infty} x_i \cdot \overline{yi}, \;\; x_i, y_i \in \ell_n^2$$

Además, sabemos que este producto está bien definido, ya que dado $N \in \mathbb{N}$ la desigualdad de Cauchy-Schwartz prueba que:

$$\left(\sum |x_i \overline{y_i}| \right)^2 \leq \left(\sum |x_i| \right)^2 \cdot \left(\sum |y_i| \right)^2 \leq \|x\|_2^2 \cdot \|y\|_2^2$$

lo que asegura que la serie $\sum_{i=1}^{\infty} x_i y_i$ es convergente.

Y su norma es la siguiente:

$$\|x\|_2 = \left(\sum_{i=1}^{\infty} |x_i^2| \right)^{1/2}$$

Además, se puede demostrar que se trata de un espacio completo. Y, por tanto, ℓ^2 también es un espacio de Hilbert.

Ejemplo 2.2.7 ($C[a,b]$)**.** Sea el espacio $C([a,b])$ definido como:

$$C[a,b] = \{f : [a,b] \longrightarrow \mathbb{C} : \text{f es continua}\}.$$

No todos los espacios de fnunciones naturales son espacios de Hilbert. Por ejemplo, en este caso la norma $\|\cdot\|_2$ no es completa y la norma $\|\cdot\|_1$ no está inducida por el producto escalar.

En el Anexo A.1 veremos las comprobaciones de estos hechos.

Ejemplo 2.2.8 ($L^2[a,b]$)**.** El espacio $L^2[a,b]$, también conocido como espacio de Lebesgue, está formado por clases de equivalencia de funciones definidas en el intervalo $[a,b]$. Estas clases de equivalencia se forman considerando equivalentes a las funciones que cumplen que $\int |f(x) - g(x)| = 0$

Está dotado de un producto interno definido de la siguiente manera:

$$\langle f, g \rangle = \int_a^b f(x)\overline{g(x)}\, dx, \quad f, g \in L^2[a,b]$$

Y, la norma inducida por este producto interno es la siguiente:

$$\|f\|_2 = \langle f, f \rangle^{1/2} = \left(\int_a^b |f(x)|^2\, dx \right)^{1/2}, \quad f, g \in L^2[a,b].$$

Además, se puede demostrar que cumple la propiedad de completitud. Por tanto, $L^2[a,b]$ es un espacio de Hilbert.

2.2.2. Ortogonalidad

Definición 2.2.9 (**Ortogonalidad**)**.** Sea $\mathcal{H}$ un espacio de producto interno, y sean A y B subconjuntos de $\mathcal{H}$.

- Dos vectores $x, y \in \mathcal{H}$ son *ortogonales*, denotado como $x \perp y$, si $\langle x, y \rangle = 0$.

- Decimos que $x \in \mathcal{H}$ es *ortogonal* al conjunto A, $(x \perp A)$, si $x \perp y$ para todo vector $y \in A$.
- Decimos que A y B son *subconjuntos ortogonales* $(A \perp B)$, si $x \perp y$ para cada $x \in A$ y cada $y \in B$.

Definición 2.2.10 (**Complemento Ortogonal**)**.** Sea A un subconjunto perteneciente a un espacio de producto interno, $\mathcal{H}$. El *complemento ortogonal* de A es el mayor conjunto posible ortogonal a A, es decir:

$$A^{\perp} = \{x \in \mathcal{H} : x \perp A\} = \{x \in \mathcal{H} : \langle x, y\rangle = 0, \forall y \in A\}\,. \tag{2.1}$$

Definición 2.2.11 (**Conjunto ortonormal**)**.** Dado un espacio arbitrario de indices I, y un conjunto de vectores $A = \{x_i\}_{i\in I}$. Se dice que A es ortonormal si es ortogonal y cada vector x_i es unitario. En otras palabras, $A = \{x_i\}_{i\in I}$ es un *conjunto ortonormal* si para todo $i, j \in I$ se cumple que:

$$\langle x_i, y_i\rangle = \delta_{ij} = \begin{cases} 1, & \text{si } i = j \\ 0, & \text{si } i \neq j \end{cases}$$

2.2.3. Teorema del punto más cercano

En general, dado un punto x y un conjunto A no siempre hay un punto en el conjunto que sea más cercano a x. Por ejemplo, si en la recta real cogemos el conjunto $A = (0, 1)$ y $x = 2$, no hay ningún punto de A que sea el más cercano a x. Da igual que elemento de A escogamos, siempre habrá otro elemento distinto que está más cerca de x. Sin embargo es razonable decir que la distancia es de x a cualquier punto de S es al menos 1, por lo que se puede decir que la "distancia de x a S es 1. Por tanto, podemos establecer una distancia mínima entre ellos aunque no haya un punto único en A que sea el más cercano.

En esta sección, vamos a extender esta idea a los espacios normados, donde se puede definir la distancia de un punto a un conjunto.

Definición 2.2.12 (**Distancia de un punto a un conjunto**)**.** Sea (X, d) un espacio normado. La *distancia* entre $x \in X$ y el subconjunto $S \subseteq X$ es el ínfimo de todas las distancias entre $x \in X$y los puntos $y \in S$:

$$dist(x, S) = \inf_{y\in S} \|x - y\|.$$

Cuando el ínfimo se alcanza en un vector $y \in S$, entonces decimos que *y es el punto más cercano de S a x.* Esto es, $y \in S$ es el más cercano a x si y solo si $\|x - y\| \leq \|x - z\|$ para cada $z \in S$.

A continuación, demostraremos que dado un conjunto convexo y cerrado en un espacio de Hilbert y un punto no perteneciente a él, siempre hay un único punto que es el más próximo.

Teorema 2.2.13 (**Teorema del punto más cercano [4], [6]**). Sea $\mathcal{H}$ un espacio de Hilbert, y sea S un subconjunto no vacío cerrado, convexo de $\mathcal{H}$. Dada cualquier $x \in \mathcal{H}$ existe un único vector $y \in S$ que es *el más cercano* a x. Es decir, existe un único vector $y \in S$ que satisface:

$$\|x - y\| = dist(x, S) = \inf_{z \in S} \|x - z\|.$$

Demostración. Consideramos $d = d(x, S)$.

Por definición de ínfimo, dado un $\epsilon = \frac{1}{n} > 0$, $d^2 + \frac{1}{n}$ no es cota inferior del conjunto $\{\|x - y\| : y \in S\}$. Por tanto, tendremos que existe un $y_n \in S$ tal que:

$$d^2 \leq \|x - y_n\|^2 \leq d^2 + \frac{1}{n} \tag{2.2}$$

Luego, por definición de ínfimo otra vez,, tenemos que para cualquier otro vector $y_n \in S$, $d \leq \|x - y_n\|$.

Ahora, para ver que el vector y es el que estamos buscando. Vamos a ver que la sucesión $\{y_n\}_{n \in \mathbb{N}}$ es una sucesión de Cauchy. Para ello, para dos elementos arbitrarios $y_n, y_m \in S$ haremos uso del punto medio $p = \frac{y_n + y_m}{2}$. Y, por definición de ínfimo, sabemos también que:

$$d(x, p) = \|x - p\| \geq d \longrightarrow \|x - p\|^2 \geq d^2. \tag{2.3}$$

A continuación, emplearemos la expresión $\|y_n - y_m\| + 4d^2$ para establecer una desigualdad que nos permitirá demostrar el resultado deseado.

$$\begin{aligned} \|y_n - y_m\|^2 + 4d^2 &\leq \|y_n - y_m\|^2 + 4\|x - p\|^2 \\ &= \|(x - y_n) - (x - y_m)\|^2 + 4\|(x - \frac{y_m + y_n}{2})\|^2 \\ &= \|(x - y_n) - (x - y_m)\|^2 + \|(x - y_n) + (x - y_m)\|^2 \\ &= 2\left(\|(x - y_n)\|^2 + \|(x - y_m)\|^2\right) \end{aligned} \tag{2.4}$$

Luego, tenemos que, $\forall \epsilon > 0$ tomando N tal que $\frac{2}{N} \leq \frac{\epsilon}{2}$, si $n, m > N$ tendremos que:

$$\text{si } \|y_n - y_m\|^2 + 4d^2 \leq 4d^2 + \frac{2}{n} + \frac{2}{m} \longrightarrow \|y_n - y_m\| \leq \frac{2}{n} + \frac{2}{m} \leq \epsilon. \tag{2.5}$$

Por tanto, queda demostrado que $\{y_n\}_{n \in \mathbb{N}}$ es una sucesión de Cauchy. Y, como $\mathcal{H}$ es un espacio completo, sabemos que existe un vector $y \in S$ tal que:

$$\lim_{n \to \infty} y_n = y \quad \equiv \quad \lim_{n \to \infty} \|y_n - y\| = 0 \tag{2.6}$$

Además, también se cumple que:

$$\lim_{n\to\infty}\|(x-y_n)-(x-y)\| = \lim_{n\to\infty}\|y_n-y\| = 0 \longrightarrow \lim_{n\to\infty}\|(x-y_n)\| = \|x-y\| \tag{2.7}$$

Aplicando límites a la desigualdad 2.2 tenemos que:

$$d^2 = \lim_{n\to\infty} d^2 \leq \lim_{n\to\infty}\|(x-y_n)\|^2 = \|x-y\|^2 \leq \lim_{n\to\infty}(d^2+\frac{1}{n^2}) = d^2$$

Se deduce entonces que $d = \|x-y\|$. Y, por tanto, se demuestra que, para el vector y, dado que hemos considerado $d = \text{ínf}_{z\in S}\|x-z\|$, se cumple que:

$$\|x-y\| = dist(x,S) = \inf_{z\in S}\|x-z\|$$

Por último, falta ver que ese vector y es único. Para ello, suponemos que existe otro vector $z \in S$ que también es el más cercano a x. Por ello:

$$\|x-y\| = d = \|x-z\|$$

Ahora, considerando el punto intermedio $p = \frac{y+z}{2}$, y teniendo en cuenta 2.5 $y_n = y$, y $y_m = z$, tenemos que:

$$2d^2 + 2d^2 = 2\|x-z\|^2 + 2\|x-y\|^2 \overset{(L.P)}{=}$$
$$\|(x-y)+(x-z)\|^2 + \|(x-y)-(x-z)\|^2 \overset{2,4}{\geq} 4d^2 + \|y-z\|^2$$

Por tanto, se deduce que:

$$4d^2 \geq 4d^2 + \|y-z\|^2 \longrightarrow 0 \geq \|y-z\| \longrightarrow y = z.$$

□

2.2.4. Proyección ortogonal

Para el cálculo del punto más cercano que hemos ido hablando hasta el momento, recurriremos a la proyección ortogonal. Este concepto es fundamental en el contexto de los espacios de Hilbert ya que nos permite encontrar el vector más cercano a un subespacio cerrado.

Definición 2.2.14 (**Proyección ortogonal**)**.** Sea M un subespacio cerrado de un espacio de Hilbert $\mathcal{H}$.

- Dado $x \in \mathcal{H}$, al único vector $p \in M$ que es el más cercano a x se le llama *proyección ortogonal* de x en M.

- La función $P : \mathcal{H} \longrightarrow \mathcal{H}$ definida por $Px = p$, donde p es la proyección ortogonal de x en M, es le llamada proyección ortogonal de $\mathcal{H}$ sobre M.

Dado que la proyección ortogonal p es el vector más cercano a x, podemos pensar que p es la mejor aproximación a x mediante vectores de M. El vector error se definide como la diferencia entre el vector original y su proyección p en el subespacio M, es decir, $e = x - p$.

El siguiente lema nos da expresiones equivalentes a la proyección ortogonal, que nos permitirán profundizar más en el concepto.

Teorema 2.2.15. Sea M un subespacio cerrado del espacio de Hilbert $\mathcal{H}$. Dados los vectores x y p pertenecientes a $\mathcal{H}$, las 4 siguientes afirmaciones son equivalentes:

- p es la proyección ortogonal de x en M, es decir, p es el único punto de M que es el más cercano a x.
- $p \in M$ y $x - p \in M^{\perp}$.
- $x = p + e$, donde $p \in M$ y $e \in M^{\perp}$.
- $e = x - p$ es la proyección ortogonal de x en $M^{\perp}$.
- $\mathcal{H} = M \oplus M^{\perp}$.

Demostración. Todos los apartados anteriores se deducen de demostrar que: $x = p + (x - p)$ con $p \in M$ y $(x - p) \in M^{\perp}$.

Para ello, es necesario ver que:

- $M \cap M^{\perp} = \{0\}$. Para ello, consideramos $x \in M \cap M^{\perp}$:

$$\|x\|^2 = \langle x, x \rangle = 0 \longrightarrow x = 0.$$

- $\mathcal{H} = M + M^{\perp}$. Si tenemos $x \in \mathcal{H}$, $x - p \in M^{\perp}$. Luego, tendremos que:

$$x = p + (x - p) \longrightarrow \text{con} p \in M, \text{ y} (x - p) \in M^{\perp}.$$

  Luego, todo elemento de $\mathcal{H}$ se puede escribir como la suma de alguien de M y alguien de $M^{\perp}$.

□

Lema 2.2.16. Sea $\mathcal{H}$ un espacio de Hilbert.

- Si M es un subespacio cerrado de $\mathcal{H}$, entonces $(M^{\perp})^{\perp} = M$.

- Si A es un subespacio de $\mathcal{H}$, entones:

$$A^{\perp} = span(A)^{\perp} = \overline{span}(A)^{\perp} \text{ y } (A^{\perp})^{\perp} = \overline{span}(A)$$

siendo $span(A)$ el el conjunto de todas las combinaciones lineales finitas de los elementos de A.

La proyección ortogonal tiene las siguientes propiedades:

- $P(v) = v \longleftrightarrow v \in M$.
- $P \circ P = P$.
- Dado $x \in \mathcal{H}$, $x = v+w$ tal que $v \in M$ y $w \in M^{\perp}$, entonces: $P(v+w) = v, v \in M, w \in M^{\perp}$.
- Preserva el producto interno: $\langle x, Py\rangle = \langle Px, y\rangle$, $\forall x, y \in \mathcal{H}$

2.2.5. Teorema de Riesz

En esta sección introduciremos el concepto de los *funcionales*, que son operadores con imagen en $\mathbb{R}$ o $\mathbb{C}$. Estas permiten asignar un número complejo o real a cada vector del espacio normado. Son aplicaciones de la forma: $\mu : X \longrightarrow \mathbb{F}$, lo que nos permitiá analizar el comportamiento de los vectores en el espacio.

A continuación, vamos a definir unos conceptos previos de los que haremos uso a lo largo de esta sección.

Definición 2.2.17 (**Espacio Dual**)**.** El conjunto de todas las funcionales lineales y acotadas en un espacio normado X es conocido como *espacio dual* de X. Este se denota como:

$$X^* = \{\mu : X \to \mathbb{F} : \text{A es acotada y lineal }\}.$$

X^* es un espacio normado, cuya norma es la siguiente:

$$\|\mu\| = \sup_{\|x\|=1} |\mu(x)|, \;\; x \in X, \;\; \mu \in X^*$$

Definición 2.2.18 (**Funcional asociado a** $y \in X$)**.** Dado un espacio con producto interno $(X, \langle\cdot,\cdot\rangle$ y dado el vector $y \in \mathcal{X}$, definimos el siguiente *funcional* que depende del vector y:

$$\mu_y : \mathcal{H} \longrightarrow \mathbb{F}, \;\; \mu_y(x) = \langle x, y\rangle, \;\; x \in \mathcal{H}.$$

Esta funcional cumple las siguientes propiedades:

- Como el producto interno es lineal en la primera variable, es lineal:

$$\mu_y(ax + bz) = \langle ax + bz, y\rangle = a\langle x, y\rangle + b\langle z, y\rangle = a\mu_y(x) + b\mu_y(z).$$

- Por la desigualdad de Cauchy-Bunyakowski-Schwarz sabemos que μ_y es acotado:

$$|\mu_y(x)| = |\langle x, y\rangle| \leq \|x\|\|y\|, \;\; x \in H.$$

  siendo $\|y\|$ la constante M correspondiente a la definicion de acotada.

- Resumiendo, se obtiene que si $y \in X$, entonces $\mu_y \in X^*$ y se cumple que:

$$\|\mu_y\| = \sup_{\|x\|=1} |\mu_y(x)| = \sup_{\|x\|=1} |\langle x, y\rangle| = \|y\|.$$

El teorema de Riesz establece una relación importante entre los elementos de un espacio y su espacio dual, de tal forma que cada funcional lineal y continuo, μ en $\mathcal{H}^*$ puede ser representado de forma única con un elemento de $y \in \mathcal{H}$.

Teorema 2.2.19 (**Teorema de Riesz**)**.** Sea $\mathcal{H}$ un espacio de Hilbert y $\mathcal{H}^*$ su dual. Si $\mu \in H^*$, entonces existe un único vector $y \in \mathcal{H}$ tal que $\mu(x) = \langle x, y\rangle, \;\; \forall x \in \mathcal{H}$.

Demostración. Para esta demostración asumiremos que μ es un operador distinto de 0, lo que implica que μ no asigna todos los vectores a 0. Por tanto, $\ker(\mu) \subset \mathcal{H}$.

Dado que el $\ker(\mu)$ no abarca todo $\mathcal{H}$, sabemos que hay elementos en $\mathcal{H}$ que no pertenecen al $\ker(\mu)$. Esto implica que existen vectores distintos de 0 en $\mathcal{H}$ que son ortogonales a todos los vectores del $\ker(\mu)$.

Como $\mathcal{H} = Ker(\mu) \oplus Ker(\mu)^\perp$ (y $Ker(\mu) \neq \mathcal{H}$), existirá $z \neq 0$ con $z \in Ker(\mu)^\perp$.

Por tanto, sea $z \neq 0 \in \ker(\mu)^\perp$. Entonces $z \notin \ker(\mu)$, es decir, $\mu(z) \neq 0$. Luego, podemos definir $w \in \mathcal{H}$ como $w = \frac{1}{\mu(z)}z$. Y, como μ es un operador linear, tenemos que:

$$\mu(w) = \frac{1}{\mu(z)}\mu(z) = 1$$

Ahora, definimos el vector $y \in \ker(\mu)^\perp$ de la siguiente manera:

$$y = \frac{w}{\|w\|^2}$$

En primer lugar, demostraremos que $\mu(x) = \mu_y(x)$. Para empezar, consideremos el vector arbitrario $x \in \mathcal{H}$ y usando la linearidad del operador μ comprobaremos:

$$\mu(x - \mu(x)w) = \mu(x) - \mu(x)\mu(w) = \mu(x) - \mu(x) \cdot 1 = 0.$$

Concluimos entonces que $x - \mu(x)w \in \ker(\mu)$ y sabemos que $y \in \ker(\mu)^{\perp}$, luego $y \perp (x - \mu(x)w)$.

Por tanto, si son perpendiculares sabemos que su producto escalar es 0, y tenemos que:

$$\begin{aligned} 0 = \langle x - \mu(x)w, y\rangle = & \\ & \langle x, y\rangle + \langle -\mu(x)w, y\rangle = \langle x, y\rangle - \mu(x)\langle w, y\rangle = \\ & \langle x, y\rangle - \mu(x)\langle w, \frac{w}{\|w\|^2}\rangle = \langle x, y\rangle - \mu(x)\frac{\langle w, w\rangle}{\|w\|^2} \\ & \langle x, y\rangle - \mu(x) \cdot 1 = \langle x, y\rangle - \mu(x). \end{aligned}$$

Luego, $0 = \langle x, y\rangle - \mu(x) \rightarrow \langle x, y\rangle = \mu(x)$. Y, por tanto, $\mu(x) = \langle x, y\rangle = \mu_y(x), \forall x \in \mathcal{H}$. Concluimos entonces, que: $\mu = \mu_y$.

Por último nos falta ver que el vector y buscado es único. Para ello consideramos que existe un vector $z \neq y \in \mathcal{H}$ tal que $\mu(x) = \mu(z)$ también. Por lo que $\mu(x) = \langle x, y\rangle = \langle x, z\rangle$. Luego $\langle x, y\rangle - \langle x, z\rangle = 0$.

Ahora, aplicando las propiedades del producto interno tenemos que: $\langle x, y - z\rangle = 0$. Y, eligindo $x = y - z$, por propiedades del producto interno sabemos que: $y - z = 0 \rightarrow y = z$.

Por tanto, queda demostrado que dado un y el operador μ se define como $\mu(x) = \langle x, y\rangle$, para cualquier $x \in \mathcal{H}$ y además es único.

□

Capítulo 3

Espacios de Hilbert con Núcleo Reproductor

A lo largo de este capítulo se va a profundizar en la definición de espacios de Hilbert con núcleo reproductor, así como algunas de sus propiedades importantes.

Recordemos, que como hasta ahora, usaremos la notación $\mathbb{F}$ para denotar el conjunto de los números complejos ($\mathbb{C}$) o los números reales ($\mathbb{R}$), dependiendo del contexto.

Para el desarrollo de este capítulo se ha utilizado la referencia: [7]

3.1. El método del núcleo

A lo largo de esta sección haremos uso del subconjunto $\mathcal{F} := \mathcal{F}(X, \mathbb{F})$. Se trata del espacio de funciones: $f : X \longrightarrow \mathbb{F}$. Además, es un espacio vectorial con las siguientes operaciones:

- $(f + g)(x) = f(x) + g(x)$, $\forall f, g \in \mathcal{F}$, $\forall x \in X$.
- $(\lambda \cdot f)(x) = \lambda \cdot (f(x))$, $\forall f \in \mathcal{F}$, $\lambda \in \mathbb{R}$, $\forall x \in X$.

Definición 3.1.1 (**Espacio de Hilbert con núcleo reproductor**)**.** Sea X un conjunto. Llamamos a un subespacio $\mathcal{H} \subseteq \mathcal{F}(X, \mathbb{F})$ *espacio de Hilbert con núcleo reproductor*, o más brevemente RKHS si:

- $\mathcal{H}$ admite un producto interno que lo convierte en espacio de Hilbert. Está dotatado pues de una norma $\|\cdot\|$ y una distancia d.

- Para cada $z \in X$ la función de evaluación lineal y acotada, definida de la siguiente manera, $E_z : \mathcal{H} \to \mathbb{F}$, $E_z(f) = f(z)$, es acotada para la norma de $\mathcal{H}$.

La función evaluación de la que hemos hablado en la definición previa nos permite evaluar la función f en cualquier $z \in X$ en $\mathcal{H}$.

Una vez tenemos definida la funcion de evaluación lineal y acotada, por el Teorema de Riesz (2.2.19), sabemos que existe un único vector, al que llamaremos $k_z \in \mathcal{H}$ tal que para cada $f \in \mathcal{H}$ tenemos que:

$$E_z(f) = f(z) = \langle f, k_z \rangle \tag{3.1}$$

De esta manera, la función de evaluación es siempre un producto escalar con un vector fijo. Es decir, se puede evaluar el valor de la función f en un punto z utilizando el producto interno con un vector k_z que pertenece al mismo espacio de Hilbert.

Basándonos en (3.1), si escogemos un vector $k_x \in \mathcal{H}$, tenemos que:

$$k_x(z) = \langle k_x, k_z \rangle \tag{3.2}$$

Esta expresión establece que el valor del núcleo reproductor k_x evaluado en el punto z se obtiene calculando el producto interno entre los vectores k_x y k_z en el espacio de Hilbert. Por tanto, ahora sí se puede definir lo siguiente.

Definición 3.1.2 (**Núcleo reproductor**)**.** Dado $\mathcal{H} \subseteq \mathcal{F}(X, \mathbb{F})$ un RKHS, podemos definir la función:

$$K : X \times X \to \mathbb{F}, \;\; K(x, z) = k_x(z) \;\; \forall x, z \in X.$$

a la que se conoce como *núcleo reproductor* de todo el espacio de funciones, es decir, de $\mathcal{H}$.

La función $K : X \times X \to \mathbb{F}$ cumple las siguientes propiedades

- $K(x, y) = k_x(y) = \langle k_x, k_y \rangle$.
- $K(x, y) = \langle k_x, k_y \rangle = \overline{\langle k_y, k_x \rangle} = \overline{K(y, x)}$, si estamos trabajando con complejos.
- $K(x, y) = K(y, x)$ si estamos trabajando con reales.
- $\|E_y\|^2 = \|k_y\|^2 = \langle k_y, k_y \rangle = K(y, y)$

3.2. Ejemplos básicos

3.2.1. $\mathbb{C}^n$ como un RKHS

Veremos que el espacio $\mathbb{C}^n$ es un espacio de Hilbert con núcleo reproductor, ya que cumple lo siguiente:

- Podemos hallar un conjunto X de modo que podamos considerar $\mathbb{C}^n$ como un espacio de funciones de X en $\mathbb{C}$. Este conjunto será $X = \{1, 2, \cdots, n\}$. Para ello, simplemento vemos cada $v = (v_1, \cdots, v_n)$ como una función $v : X \longrightarrow \mathbb{C}$, definida como $v(j) = v_j$.

- Necesitamos además que el espacio $\mathbb{C}^n$ sea un espacio de Hilbert. Para ello, dados los vectores v_i, w_i, es suficiente con definir el siguiente producto interno:

  $$\langle v, w \rangle = \sum_{i=1}^{n} v_i \overline{w_i}$$

  Este producto interno induce la norma euclídea, luego se trata efectivamente de un espacio de Hilbert.

- Por último, es necesario que la función de evaluación $E_i : \mathbb{C}^n \longrightarrow \mathbb{C}$ sea acotada:

  $$|E_i(v)| = |v_i| \leq \sqrt{v_1^2 + \cdots + v_n^2} = \|v\|$$

  Ahora, ya sabemos que es un RKHS, por tanto, se puede definir el núcleo reproductor del espacio considerando la base ortonormal de $\mathbb{C}^n$.

  Por tanto, ya tenemos estructura de RKHS y sabemos que exite un núcleo reproductor $K : X \times X \longrightarrow \mathbb{C}$ que por (Núcelo reproductor) sabemos que se define:

  $$K(i, j) = k_i(j)$$

  Siendo $k_i \in \mathbb{C}^n$, sabemos que para cualquier función del espacio $\mathbb{C}^n$, en este caso, cogemos e_j, tenemos que:

  $$k_i(j) = \langle k_i, e_j \rangle = e_j(i) = \delta_{ij}$$

  Y, por tanto el núcleo reproductor se define como:

  $$K(i, j) = k_i(j) = \delta_{ij} = \begin{cases} 1, & \text{si } i = j \\ 0, & \text{si } i \neq j \end{cases}$$

3.2.2. $L^2[a,b]$ es un no ejemplo

Consideremos el espacio de Lebesgue $L^2[0,1]$, junto a su norma:

$$\|f\|_2 = \langle f, f\rangle^{1/2} = \left(\int_0^1 |f(t)|^2\, dt\right)^{1/2}, \quad f, g \in L^2[a,b].$$

Veremos que este espacio de Hilbert visto como espacio de funciones de $[a,b]$ en $\mathbb{C}$ no es un ejemplo de RKHS. Lo veremos observando que la función de evaluación no está acotada. Por lo que, queremos buscar unas funciones f_n para el que la función de evaluación cumpla eso.

Para ello, consideraremos el espacio X como el conjunto de todos los posibles valores de x en el intervalo $[0,1]$. Dado un $x \in X$ fijo, consideramos la siguiente sucesión de funciones:

$$f_n(t) = \begin{cases} (\frac{t}{x})^n, & \text{si } 0 \le t \le x \\ (\frac{1-t}{1-x})^n, & \text{si } x < t \le 1. \end{cases}$$

donde se cumple que $f_n(x) = 1$ para todo n. Sin embargo, veremos que $\lim_{n\longrightarrow\infty}\|f_n\| = 0$. Para ello, calculamos su norma:

$$\int_0^x \left(\frac{t}{x}\right)^{2n} dt + \int_x^1 \left(\frac{1-t}{1-x}\right)^{2n} dt =$$

$$\frac{1}{x^{2n}}\int_0^x t^{2n}\, dt + \frac{1}{(1-x)^{2n}}\int_x^1 (1-t)^{2n}\, dt =$$

$$\frac{x}{2n+1} + \frac{1-x}{2n+1} = \frac{1}{2n+1}$$

Ahora, calculando su límite cuando $n \longrightarrow \infty$, tenemos que:

$$\lim_{n\longrightarrow\infty} \frac{1}{2n+1} = 0.$$

Entonces, tenemos que $\|f_n\| = 0$, cuando n tiende a infinito.

Ahora, la función de evaluación $E_t : L^2([0,1]) \longrightarrow \mathbb{C}$, está definida como: $f \longrightarrow f(t)$. Como sabemos que $f_n(x) = 1$, entonces $|E_x(f_n)| = |f_n(x)| = 1$.

Si E_x fura acotada, existiría un $M > 0$ tal que: $|E_x(f_n)| \le M \cdot \|f_n\|$, Sin embargo esto no es posible, ya que:

$$|E_x(f_n)| = 1, \;\; \|f_n\| = 0. \text{ Por tanto, tomando límites } 1 \le M \cdot 0$$

Por tanto, no se puede encontrar ninguna función f_n de manera que la función de evaluación esté acotada. Entonces, se puede afirmar que $L^2([0,1])$ no es un RKHS.

3.3. Teorema de Moore-Aronszajn

Hasta ahora hemos visto que todo espacio de Hilbert con núcleo reproductor define una función núcleo. Sin embargo, esta sección veremos que también ocurre en la otra dirección.

Primero de todo, veremos una propiedad que la función núcleo ha de cumplir para garantizar que el RKHS generado a partir de este sea un espacio de Hilbert válido.

Definición 3.3.1. Sea X un conjunto y sea el núcleo $K : X \times X \longrightarrow \mathbb{F}$. Este es *definido positivo* si cumple las siguientes propiedades:

- K cumple la simetría, es decir, $K(x, y) = K(x, y)$.
- La matriz definida como $k_{ij} = K(x_i, x_j)$ es positiva definida, es decir, $v^T kv \geq 0$, $\forall v \in \mathbb{R}^m$.

Teorema 3.3.2 (**Teorema de Moore-Aronszajn**)**.** Si K es un núcleo positivo definido sobre un conjunto X, entonces existe un espacio de Hilbert $\mathcal{H}$ con núcleo reproductor cuyo núcleo es K.

Este teorema es realmente útil en la teoría de los RKHS porque establece una conexión fundamental entre los núcleos definidos positivos y los espacios de Hilbert. Garantiza que para cada núcleo definido positivo existe un RKHS asociado y viceversa.

Esto proporciona una forma de reconstruir un RKHS a aprtir de núcleos definidos positivos y, a su vez, de verificar si un núcleo dado puede ser utilizado para construir un RKHS.

Ejemplo 3.3.3. [1] En este ejemplo, veremos un ejemplo de núcleo positivo definido con el que se podrá aplicar el teorema 3.3.2.

Consideramos el núcleo polinómico definido como:

$$K(x, y) = (x^T z + c)^d, c \geq 0.$$

Se trate de un núcleo positivo definido ya que cumple que:

- Simetría: $K(x, z) = (x^T z + c)^d = (\langle x, z\rangle + c)^d = (\langle z, x\rangle + c)^d = (z^T x + c)^d = K(z, x)$.
- La matriz definida como $A = \sum_{k=1}^{n} (x_k^T v)^2$ es definida positiva, es decir $A \geq 0$.

Por tanto, el kernel polinómico es un núcleo positivo definido. Y, con esto aplicando el teorema anterior podemos afirma la existencia de un RKHS cuyo núcleo es $K(x, y) = (x^T z + c)^d$.

El espacio de Hilbert $\mathcal{H}$ generado por el núcleo polinómico será un espacio de funciones en $\mathbb{R}^n$ que satisface las propiedades de un espacio de Hilbert. Además, el producto interno de este espacio estará relacionado con el núcleo polinómico.

Capítulo 4

Fundamentos del método del núcleo

En el campo de la estadística y el aprendizaje automático, los espacios de Hilbert con núcleo reproductor tienen diversas aplicaciones.

En muchos casos, los conjuntos de datos pueden ser no lineales y, por tanto, no pueden ser separados de manera efectiva por un hiperplano en el espacio original. Es ahí, cuando entra el concepto del 'Kernel Trick', con el que se permite realizar una transformación impllícita de los datos a un espacio de Hilbert de mayor dimensión, donde los datos pueden ser más fácilmente separables.

Una vez se elige la función núcleo adecuada, se pueden utilizar algoritmos de aprendizaje automático como las Máquina de Vectores de Soporte (SVM), que es el que nos centraremos en este trabajo.

4.1. Definiciones previas

A lo largo de esta sección, explicaremos el fundamento teórico de muchos de los conceptos utilizados en el Capítulo 1.

Definición 4.1.1 (**Aplicación de características**)**.** Una *aplicación de características* ϕ es una incrustación del conjunto X donde los datos residen en un posible espacio de Hilbert $\mathcal{H}$ que puede tener incluso dimensión infinita. En otras palabras, se refiere a una función que toma una entrada un conjunto de datos y la representa en un espacio de características de mayor dimensión.

La aplicación de características, $\phi : X \longrightarrow \mathcal{H}$ permite transformar los datos de entrada del conjunto X en elementos de $\mathcal{H}$ de mayor dimensión, lo cual nos da flexibilidad (ver sección 4.2) para clasificarlos. La estructura de espacio de Hilbert nos permitirá hacer los cálculos de manera eficaz.

Este aplicación ϕ induce un núcleo que, según lo visto en la sección 3.1, se define como el producto interno en el espacio de Hilbert, $K(x, y) = \langle \phi(x), \phi(y) \rangle$. El uso de este núcleo nos permite realizar los cálculos en el espacio de mayor dimensión de manera más eficiente sin necesidad de conocer explícitamente la función ϕ.

Una vez se define la aplicación de características y el núcleo, se puede abordar la predicción, visto en el Capítulo 1, de la clasificación de nuevos datos, a través de los predictores. Estos son funciones definidas en el espacio de Hilbert $\mathcal{H}(K)$ inducido por el núcleo K, para aprender y generalizar a partir de los datos. Estos predictores permitirán entrenar el modelo con los datos de entrenamiento, de manera que se puedan encontrar generalizaciones de nuevos datos no entrenados, ya sea para problemas de clasificación o de regresión.

En otras palabras, la idea final es hallar una función $f : X \longrightarrow Y$ que nos permitan clasificar correctamente los datos de entrenamiento x_i. Para su clasificación, recurriremos a las etiquetas λ_i asociadas a cada elemento x_i, que nos ayudarán a saber a que clase pertenece cada uno de ellos.

4.2. Separabilidad entre hiperplanos

Como hemos visto anteriormente, el objetivo es hallar una función que permita separar los datos de entrenamiento. Para poder comprender correctamente el significado de esta frase, se van a introducir una serie de definiciones.

Definición 4.2.1 (**Hiperplano**)**.** Sea un espacio de Hilbert $\mathcal{H}$. Entonces, un *hiperplano* V en $\mathcal{H}$ se define como:

$$V = \{x \in \mathcal{H} : \langle x, v \rangle = c\}, \;\; v \in V, c \in \mathbb{C}.$$

En este caso, v se puede interpretar como un funcional lineal y continuo en $\mathcal{H}^*$.

Cualquier hiperplano V particiona el espacio en tres conjuntos:

- El propio hiperplano: $V = \{x \in \mathcal{H} : \langle x, v \rangle = c\}$
- Los lados del hiperplano dados por las ecuaciones:

$$V_+ = \{x \in \mathcal{H} : \langle x, v \rangle > c\}$$

$$V_- = \{x \in \mathcal{H} : \langle x, v \rangle < c\}$$

Definición 4.2.2 (**Separabilidad**). Sea $X = \{x_i : i \in I\}$ un conjunto de datos en un espacio de características $\mathcal{H}$, y sean $X_+ = \{x_i : \lambda_i = +1\}$ y $X_- = \{x_i : \lambda_i = -1\}$, donde $\lambda_i \in \pm 1$ son las etiquetas que indican a que lado del hiperplano se halla el dato. Decimos que los datos son separables si existe un hiperplano V en $\mathcal{H}$ tal que X_+ está contenido en un lado de V y X_- está contenido en el otro lado de V. En otras palabras, los datos son separables si es posible encontrar un hiperplano que pueda perfectamente discriminar entre las dos clases. Si no existe un hiperplano que pueda separar perfectamente los datos, se dice que los datos son no separables.

Es importante destacar que no siempre es posible hallar un hiperplano que separe completamente el conjunto de datos.

Proposición 4.2.3 (**Criterio para saber si es separable o no**). Si $X = \{x_i : i = 1, \cdots, n\}$ es un conjunto de n puntos en un espacio vectorial y sea W el subespacio que generan. Entonces:

1. Si $dim(W) = n$ y el conjunto X se divide en dos conjuntos X_+ y X_-, entonces siempre existe un hiperplano que separa a ambos conjuntos.
2. Si $dim(W) < n - 1$ podemos encontrar particiones X_+ y X_-, que no pueden separarse mediante hiperplano alguno.

Demostración. A lo largo de esta demostración, veremos que se cumplen las dos afirmaciones. En ambos casos nuestro objetivo será hallar un vector $v \in \mathbb{C}^n$ y una constante $c \in \mathbb{C}$ de manera que si se puede hallar el hiperplano $V = \{x \in \mathbb{C}^n : \langle x, v\rangle = c\}$ se tenga que: $X_+ \subseteq V_+$ y $X_- \subseteq V_-$.

1. En primer lugar, demostraremos que se puede hallar un hiperplano V que separe ambos conjuntos cuando la $dim(W) = n$. Para ello, sean las etiquetas $\lambda_i = \pm 1$ dependiento si $x_i \in X_\pm$.

   Ahora sabemos que si X contiene n elementos y la dimensión del espacio es n, entonces los elementos x_i son linealmente independientes y, por tanto, $x_1, \cdots, x_n$ es una base de W. Entonces, podemos considerar la base dual asociada $\{v_1, \cdots, v_n\}$ con la relación: $\langle x_i, v_j\rangle = \delta_{i,j}$.

   Ahora, definimos el vector v como:

   $$v = \sum_{j=1}^{n} \lambda_j v_j, \text{ donde } \begin{cases} \lambda_j = 1, & \text{si } x_j \in X_+ \\ \lambda_j = -1, & \text{si } x_j \in X_- \end{cases} \tag{4.1}$$

   Luego, siento $c = 0$ sustituyendo la esxpresión anterior y por linealidad del producto interno tenemos que:

   $$\langle x_i, v\rangle - c = \langle x_i, \sum_{j=1}^{n} \lambda_j v_j\rangle = \sum_{j=1}^{n} \lambda_j \langle x_i, v_j\rangle = \lambda_i \langle x_i, v_i\rangle = \lambda_i.$$

Por tanto, por la definición de las etiquetas λ_i el producto interno entre x_i y v será:

$$\langle x_i, v\rangle > 0, \text{ si } x_i \in X_+.$$
$$\langle x_i, v\rangle < 0, \text{ si } x_i \in X_-.$$

Podemos concluir que sí que existe un hiperplano $V = \{x \in \mathcal{H} : \langle -x, v\rangle = c\}$ tal que, $X_+ \subseteq V_+$ y $X_- \subseteq V_-$.

2. Finalmente demostraremos que no es posible hallar un hiperplano que separe el conjunto de datos si la $dim(W) \leq n-2$.

   En este caso, tendremos la base $\{v_1, \cdots, v_n\}$ linealmente dependientes. Además, podemos encontrar los coeficientes $\alpha_1, \cdots, \alpha_n$ tal que:

$$\sum_{j=1}^{n} \alpha_j v_j = 0$$

$$\sum_{j=1}^{n} \alpha_j, \text{ (y algún } \alpha_j \neq 0). \tag{4.2}$$

   Para ver esto, consideramos la aplicación $\phi(x) = v \oplus 1$, es decir, $(v_1, 1), \cdots, (v_n, 1) \subseteq W \oplus \mathbb{C}$.

   Como $dim(W \oplus \mathbb{C}) \leq n-1$, sabemos que $(v_1, 1), \cdots, (v_n, 1)$ son linealmente dependientes. Por tanto, existen $\alpha_1, \cdots, \alpha_n \in C$, ($\alpha_i \neq 0$ para algún i), tal que:

$$\sum_{i=1}^{n} \alpha_i v_i = 0$$

$$\sum_{i=1}^{n} \alpha_i, \text{ (y algún } \alpha_i \neq 0).$$

   En este caso, queremos ver que hay particiones del conjunto X que no se pueden separar. Para ello, creamos las siguientes particiones y veremos que no son separables:

$$X_+ = \{x_i : \alpha_i > 0\} \neq \emptyset.$$

$$X_- = \{x_i : \alpha_i < 0\} \neq \emptyset.$$

   Ahora, para poder utilizar la defínción de hiperplano, consideramos la diferencia:

$$\sum_{x_i \in X} \alpha_i \langle x_i, v\rangle - \sum_{x_i \in X} \alpha_i c$$

   Luego, tendremos que:

$$\sum_{x_i \in X_+} \alpha_i (\langle x_i, v\rangle - c) > 0$$

$$\sum_{x_i \in X_-} \alpha_i(\langle x_i, v\rangle - c) > 0$$

Por otra parte:

$$\sum_{x_i \in X_+} \alpha_i(\langle x_i, v\rangle - c) + \sum_{x_i \in X_-} \alpha_i(\langle x_i, v\rangle - c) =$$

$$\sum_{i=1}^{n} \alpha_i(\langle x_i, v\rangle - c) = \sum_{i=1}^{n} \alpha_i \langle x_i, v\rangle - \sum_{i=1}^{n} \alpha_i c =$$

$$\langle \sum_{i=1}^{n} \alpha_i x_i, v\rangle - \left(\sum_{i=1}^{n} \alpha_i\right) c \overset{4,2}{=} 0.$$

Por tanto, si la suma de dos cosas estrictamente positivas da 0, eso solo es posible si se cumple que: $\sum \alpha_i = 0$, para todos los $\alpha_i = 0$. Sin embargo, esto es una contradicción.

Concluimos que, cuando $dim(W) \leq n-2$, no es posible encontrar un hiperplano separador.

□

Este teorema no asegura la existencia de un hiperplano que separe el conjunto X si la $dim(W) = n - 1$ para un conjunto de n puntos, entonces puede que haya o no haya hiperplano que separe el conjunto.

4.2.1. Hiperplano de margen máximo

Hasta ahora, hemos visto la necesidad de enontrar un hiperplano en el espacio que nos permita clasificar el conjunto de datos en un espacio de mayor dimensión.

Definición 4.2.4 (Hiperplano de margen máximo). Supongamos que $X \subseteq \mathcal{H}$ se divide en X_+ y X_- mediante un hiperplano. Llamamos *hiperplano de margen máximo* al hiperplano en el que la mínima distancia a los puntos es máxima, lo que disminuye el margen de error.

Basándonos en la definición de un punto a un conjunto (2.2.12) y la forma del hiperplano, tenemos que la distancia de un punto x al hiperplano V queda definida mediante la fórmula:

$$d(x, V) = \frac{|\langle x, v\rangle - c|}{\|v\|}$$

Luego, para hallar este hiperplano, como se ha comentado antes, será suficiente con hallar un vector v que permita definir ese hiperplano de manera que la clasificación o la separación de los datos sea óptima. Para hallar este vector, se ha de tener en cuenta el siguiente teorema.

Lema 4.2.5. Dados $x_1, \cdots, x_n \in V$, $\lambda_1, \cdots, \lambda_n$. El conjunto $C = \{v : \lambda_i \langle x_i, v\rangle - c \geq 1\}$ es convexo y cerrado.

Demostración. Se define para cada i: $\phi_i : \mathcal{H} \longrightarrow \mathbb{C}^n$, definida como$\phi_i(v) = \lambda_i \langle v, x_i\rangle - 1$

Sabemos que ϕ_i es continua porque $v \mapsto \langle x_i, v\rangle$ lo es. Y, además:

$$C = \bigcap_i^n \phi_i\left([+1, +\infty)\right)$$

Como $[+1, +\infty)$ es cerrado en C y la intersección de cerrados es cerrado, podemos concluir que C es cerrado.

Ahora, falta demostrar que también es convexo. Para ello, sabemos que: $\lambda_i \langle v_1, x_i\rangle - c \geq 1$ y que $\lambda_i \langle v_2, x_i\rangle - c \geq 1$. Y, que cumple la condición de convexo, ya que:

$$(\alpha \lambda_i \langle \alpha v_1 + (1-\alpha) v_2, x_i\rangle - c) \in C.$$

□

Para el siguiente teorema consideramos el conjunto $X = \{x_1, x_2, \cdots, x_n\}$ y sus particiones X_+ y X_-. Además, utilizaremos las etiquetas $\lambda_i = \{\pm 1\}$ asociadas a si x_i pertenece a un lado o a otro del hiperplano.

Teorema 4.2.6 (**Teorema de existencia del hiperplano de margen máximo**)**.** Consideramos el conjunto X de un espacio de Hilbert $\mathcal{H}$ y sus particiones. Si obtenemos el vector w que soluciona el problema de minimización: $\frac{1}{2}\|v\|^2$ sujeto a $\lambda_i(\langle x_i, v\rangle - c) \geq 1$, entonces el hiperplano determinado por w y por c es un hiperplano de margen máximo. El vector v es combinación lineal de los vectores $x_1, \cdots, x_n$.

Demostración. Sea el hiperplano $V = \{x \in \mathcal{H} : \langle x, v\rangle = c\}$ que separa X_+ y X_-. Nuestro objetivo es hallar v y c.

Luego, consideramos la distancia de cada uno de los puntos $x_i \in X$ definida como:

$$d(x_i, V) = \frac{|\langle x_i, v\rangle - c|}{\|v\|} = |A_i|, \text{ donde } \begin{cases} A_i > 0, & \text{si } x_i \in V_+ \\ A_i < 0, & \text{si } x_i \in V_- \end{cases} \tag{4.3}$$

Ahora, definimos el índice i_o como aquel que hace que $A_{i_0} = min\{A_1, \cdots, A_n\}$.

Y, podemos definir nuestro hiperplano buscado como:

$$V = \{x \in \mathcal{H} : \langle x, v^{'} \rangle = c^{'}\}$$

donde:

$$v^{'} = \frac{v}{|A_{i_0}| \cdot \|v\|}, \;\; c^{'} = \frac{c}{|A_{i_0}| \cdot \|v\|}$$

Para este hiperplano V, x_{i_0} sería el vector soporte. Por tanto, $d(x_{i_0}, V)$ será tan grande como sea posible. Ahora, deberemos elegir ese $v^{'}$ y $c^{'}$ que consigue esto.

Ahora, con esta reparametrización tenemos que la distancia es:

$$d(x_{i_0}, V) = \frac{|\langle x_{i_0}, v^{'} - c^{'}|}{\|v\|^{'}} = \frac{\frac{1}{A_{i_0}} \cdot |\langle x_{i_0}, v\rangle - c|}{\|v^{'}\|} \overset{4,3}{=} \frac{A_{i_0} \cdot \|v\|}{A_{i_0} \cdot \|v\|} \cdot \frac{1}{\|v^{'}\|} = \frac{1}{\|v^{'}\|}$$

Luego, por como hemos definido el índico i_0 tenemos que:

$$d(x_i, V) \geq d(x_{i_0}, V) = \frac{1}{\|v^{'}\|}$$

Por tanto, hemos convertido en problema en encontrar un vector v y una constante c, de modo que $|A_{i_0}| = \frac{1}{\|v^{'}\|}$ sea lo más grande posible teniendo en cuenta que:

$$\frac{|\langle x_i, v^{'} - c^{'}|}{\|v^{'}\|} \geq \frac{1}{\|v^{'}\|}$$

y que:

$$\text{si } \; x_i \in V_+, \langle x_i, v^{'} - c^{'}\rangle = \langle x_i, v^{'} - c^{'}\rangle$$
$$\text{si } \; x_i \in V_-, \langle x_i, v^{'} - c^{'}\rangle = -(\langle x_i, v^{'} - c^{'}\rangle)$$

O, lo que es lo mismo:

$$\frac{\lambda_i \cdot |\langle x_i, v^{'} - c^{'}|}{\|v^{'}\|} \geq \frac{1}{\|v^{'}\|} \longrightarrow \lambda_i \cdot |\langle x_i, v^{'} - c^{'}| \geq \frac{1}{\|v^{'}\|}. \tag{4.4}$$

A continuación, demostraremos la unicidad de la existenccia de ese vector solución al que denotaremos w. Para ello, supongamos que el problema anterior tiene dos soluciones.

Definimos el conjunto $C = \{v : \lambda_i \langle x_i, v\rangle c \geq 1\}$, es decir, aquel conjunto que contiene los vectores que cumplen las restricciones 4.4.

Por el lema 4.2.5 sabemos que C es un conjunto convexo y cerrado. Y, por el teorema del punto más cercano (2.2.13) hay un único elemento de C que es el más cercano al 0, que es el elemento w, el de norma mínima.

Por último, demostraremos que w, que define el hiperplano de margen máximo, se puede escribir como combinación lineal de los vectores $x_1, \cdots, x_n$. Para ello, hacemos uso de la proyección P sobre la combinación lineal de vectores $x_1, \cdots, x_n$ al vector w.

Como hemos visto en el 2.2.16, la proyección ortogonal preserva el producto interno, además, se cumple que $Px_i = x_i$ por ser P proyección sobre los vectores x_i. Por tanto, se tiene que:

$$\langle x_i, Pw\rangle = \langle Px_i, w\rangle = \langle Px_i, w\rangle = \langle x_i, w\rangle$$

Luego, vemos que el vector Pw también resuelve el problema de minimización. Sin embargo, w era el vector cuya norma era la mínima, y por la unicidad demostrada anteriormente, concluimos que: $Pw = w$. Y, como P es la proyección sobre el espacio vectorial $\langle x_1, \cdots, x_n\rangle$, se puede afirmar que w se puede escribir como combinación lineal de los vectores $x_1, \cdots, x_n$. Luego, existe un hiperplano de margen máximo definido por w y la constante c.

Por tanto, el problema de optimización se basa en encontrar un vector v tal que se minimice $\frac{1}{2}\|v\|^2$ sujeto a $\lambda_i(\langle x_i, v\rangle - c) \geq 1$.

□

Corolario 4.2.7. Si obtenemos $\alpha_1, \cdots, \alpha_n$ que solucione el problema de:
Minimizar:

$$\frac{1}{2}\sum_{i,j=1}^{n} \alpha_i\alpha_j\langle x_i, x_j\rangle$$

sujeto a

$$\lambda_1(\sum_{j=1}^{n} \alpha_j\langle x_1, x_j\rangle - c) \geq 1,$$

$$\vdots$$

$$\lambda_n(\sum_{j=1}^{n} \alpha_j\langle x_n, x_j\rangle - c) \geq 1$$

entonces, $w = \sum \alpha_j x_j$ es el vector que determina el hiperplano de margen máximo.

Demostración. Basta aplicar el Teorema teniendo en cuenta que sabemos que $w = \sum \alpha_j x_j$. □

4.3. Teorema del representante

Hasta ahora, hemos tratado el caso de hallar un hiperplano de margen máximo que separe correctamente X_+ y X_-. Ahora, en lugar de buscar un hiperplano de separación que separe los datos de forma precisa, nos centraremos en minimizar una función de pérdida que cuantifica el error.

Anteriormente hemos visto que el problema de encontrar un hiperplano de margen máximo estaba sujeto a las restricciones $\lambda_i(\langle x_i, v\rangle - c) \geq 1$. Entonces, se puede decir que existe un hiperplano que separa los datos si y solo si existe una función $f \in \mathcal{H}$ que cumple que: $\lambda i f(x_i) \geq 1$. o lo que es lo mismo, $(1 - \lambda_i f(x_i)) \leq 0$.

En este caso, el problema será minimizar la función de pérdida $L(f(x_1), \cdots, f(x_n)) = \sum_{i=1}^{n} \text{máx}\{0 - \lambda_i f(x_i)\}$.

Esta función de pérdida funciona bien para los datos que ya están en el conjunto, sin embargo, tiene a clasificar erróneamente los nuevos datos. Por ello, será necesario contrarrestar ese error añadiendo una especie de penalización a aquellas normas que son muy grandes e impden que se cumpla la restricción a la que está sujeta la función f. Esta restricción la aplicaremos sobre la norma de f, y se denota de la siguiente manera: $W(\|f\|_{\mathcal{H}})$

A continuación, presentaremos el teorema del representante, el cual establece una relación entre la función de pérdida y la penalización utilizada.

Teorema 4.3.1 (**Teorema del representante**)**.** Si $\mathcal{H}$ es un RKHS sobre X, $L : \mathbb{R}^n \longrightarrow \mathbb{R}$ es una función continua y $W : \mathbb{R} \longrightarrow \mathbb{R}$ es creciente, entonces la función f^* con

$$W(\|f^*\|^2) + L(f^*(x_1), \cdots, f^*(x_n)) = \inf_{f \in \mathcal{H}} W(\|f\|^2) + L(f(x_1, \cdots, f(x_n) \tag{4.5}$$

es una combinación lineal de $k_{x_1}, \cdots, k_{x_n}$. Si W es lineal y L es convexa, entonces ese ínfimo se alcanza.

Demostración. Sea S el conjunto formado por funciones que son combinación lineal de $k_{x_1}, \cdots, k_{x_n}$. Luego, sea $f^* = g + h$, siendo $g \in S$ y $h \perp g$.

Vamos a demostrar que es necesario que $h = 0$ para que la función óptima f^* sea combinación lineal de $k_{x_1}, \cdots, k_{x_n}$. Vamos a suponer que $h \neq 0$.

Sabemos que la función de pérdida se puede escribir de la siguiente manera:

$$J(f^*) = W(\|f^*\|^2) + L(f^*(x_1), \cdots, f^*(x_n)) = W(\|g + h\|^2) + L((g + h)(x_1), \cdots, (g + h)(x_n))$$

Ahora, como $h \in S^{\perp}$. Por Pitágoras, se tiene que: $\|g+h\|^2 = \|g\|^2 + \|h\|^2$. Luego:

$$J(f^*) = W(\|g\|^2 + \|h\|^2) + L(g(x_1), \cdots, g(x_n))$$

Además, sabemos que W es monótonamente creciente. Por tanto, $\|h\| \geq 0$. Por lo que sí o sí, se cumple que: $\|h\|^2 + \|g\|^2 \geq \|g\|$. Luego:

$$J(f^*) \geq W(\|g\|^2) + L(g(x_1), \cdots, g(x_n)) = J(g)$$

Entonces, se tiene que $J(f^*) > J(g)$. Es decir, se obtiene un valor de pérdida mayor para f^*. Sin embargo, esto contradice el hecho de que f^* es la función que minimiza la función de pérdida entre todas las funciones. Por ello, no puede existir ningún $h \neq 0$, tal que f^* sea la óptima.

Por tanto, $h = 0$. Y, se tiene que:

$$f^* = g + h = g + 0 \longrightarrow f^* = g \longrightarrow f^* \in S.$$

Como S está formado por la combinación lineal de las funciones de $K_{x_1}, \cdots, k_{x_n}$, queda demostrado que la función f^* también es una combinación lineal de estas. □

Luego, todas las soluciones del problema tendrán la forma: $f = \sum_{i=1}^{n} \alpha_i k_{x_i}$, es decir, la combinación lineal de las funciones núcleo. Entonces, deducimos que para hallar la solución del problema no será necesario salir de esta combinación de funciones, por lo que la solcuión está en un espacio finito-dimensional.

Capítulo 5

Aplicación Caso Práctico

En este capítulo se va aplicar el fundamento teórico visto a lo largo de todos los capítulos anteriores a un caso concreto real. Para ello, se va a utilizar una base de datos extraída del repositorio UC Irvine Machine Learning [9]. El objetivo será el mismo que se expuso en el Capítulo 1, partiremos de un conjunto de datos separados en dos clases distintas, de manera que se pueda encontrar el clasificador óptimo que las separe.

En este caso se va a utilizar la base de datos '*Predict students dropout and academic success*'. El objetivo de esta base de datos es predecir el abandono y el éxito académico de los estudiantes. Para ello, utilizaremos técnicas de clasificarción para desarrollar un clasificador basado en el algoritmo SVM para separar las dos clases de estudiantes: los que abandonan (Dropout) y los que tienen éxito académico (Graduate).

En primer lugar, exploraremos con detalle la base de datos explicando cada uno de las variables utilizadas para predecir nuestro objetivo. Y, en segundo lugar, aplicaremos el algoritmo de clasificación, comparando el rendimiento con diferentes Kernels.

5.1. Descripción Variables

La base de datos está formada por datos de 4424 individuos recopilados a partir de una institución de educación superior relacionado con estudiantes de diferentes títulos universitarios.

El conjunto de datos incluye información conocida al momento de la inscripción de los estudiantes, como su ruta académica, factores demográficos y factores socioeconómicos. Además, del rendimiento académico de los estudiantes a final del primer y segundo semestre.

A continuación, se introducirá una descripción breve de cada uno de estas variables. Para ver los distintos valores que puede tomar cada variable ver el Anexo B.2. La base de datos en total consta de 37 variables distintas:

- **Marital status:** *(Discreta)* Indica el estado civil del alumno. Toma valores del 1 al 6 donde:
- **Application mode**: *(Discreta)* Forma en la que el alumno ha presentado su solicitud. Toma un total de 17 valores distintos en el rango numérico de 1 a 57.
- **Application order**: *(Discreta)* Hace referencia a la posición de la solucitud en relación con otras solicitudes presentadas. Toma valores entre el 0 y el 9, siendo el 0 la elección más deseada y 9 la menos deseada.
- **Course**: *(Discreta)* Carrera universitaria que el alumno está cursando. En la base de datos hay 17 carreras distintas representadas por un código numérico, donde los más frecuentes son:
- **Daytime/evening attendance (0-1)**: *(Discotómicas)* Indica si el estudiante tiene clase de tarde o de mañana, donde 0 significa 'Daytime' y 1 'evening'.
- **Previous qualification:** *(Discreta)* Indica el grado alcanzado en sus estudios anteriores. Toma 17 valores distintos en un rango numérico entre 1 y 43.
- **Previous qualification (grade):** *(Continua)* Indica la nota obtenida antes de la actual, cuyo valor está entre el 0 y 200, siendo 0 la nota más baja y 200 la más alta.
- **Nacionality:** *(Discreta)* Hace referencia al país de origen del estudiante. En la base de datos hay registros de 21 nacionalidades distintas representadas por un valor numérico entre 1 y 109.
- **Mother's qualification/Father's qualification:** *(Discreta)* Ambos indican los estudios de la madre y el padre, respectivamente, del estudiante. En caso de la primera variable hay 29 trabajos distintos y la segunda variable puede tomar 34 valores distintos. Ambas variables están representadas por números entre 1 y 44.
- **Mother's occupation:** *(Discreta)* Hace referencia a l(a profesión de la madre y del padre, respectivamente, del estudiante. En caso de la primera variable se pueden encontrar 32 valores distintos, y para la segunda variable 47 valores distintos. Ambas variables están representadas por números entre 1 y 195.
- **Admission grade:** *(Continua)* Nota obtenida por el estudiante para ser admitido en la carrera universitaria, cuyo valor está entre el 0 y 200, siendo 0 la nota más baja y 200 la más alta.
- **Displaced (0-1)**: *(Discotómicas)* Se refiere a si la persona ha tenido que desplazarse de su lugar de residencia para realizar sus estudios, donde 0 significa 'no' y 1 'yes'.

- **Educational special needs (0-1)**: *(Discotómicas)* Indica si el alumno tiene necesidades espaciales que requieren apoyo educativo adicional o no, donde 0 significa 'no' y 1 'yes'.
- **Debtor (0-1)**: *(Discotómicas)* Hace referencia a si la persona tiene deudas pendientes hasta la fecha actual, donde 0 significa 'no' y 1 'yes'.
- **Tuition fees up to date (0-1)**: *(Discotómicas)* Indica si la persona ha pagado todas sus cuotas de matrícula hasta la fecha, donde 0 significa 'no' y 1 'yes'.
- **Gender (0-1)**: *(Discotómicas)* Se refiere al género del alumno, siendo '1' hombre y '0' mujer.
- **Scholarship holder (0-1)**: *(Discotómicas)* Si el alumno es beneficiario de una beca para sus estudios, donde 0 significa 'no' y 1 'yes'.
- **Age at enrollment:** *(Continua)* Indica la edad del estudiante en el momento de comenzar los estudios.
- **International (0-1)**: *(Discotómicas)* Hace referencia a si la persona es de nacionalidad extranjera o no, donde 0 significa 'no' y 1 'yes'.
- **Curricular units 1st/2nd sem (credited):** *(Discreta)* Indica el número de créditos reconocidos por la institución en cada semestre.
- **Curricular units 1st/2nd sem (enrolled):** *(Discreta)* Número de crétidos en los que el estudiante está actualmente matriculado en cada semestre.
- **Curricular units 1st/2nd sem (evaluations):** *(Discreta)* Número de crétidos en los que se ha realizado exámenes en cada semestre.
- **Curricular units 1st/2nd sem (approved):** *(Discreta)* Número de crétidos que el estudiante ha aprobado en cada semestre.
- **Curricular units 1st/2nd sem (grade) (0-20):** *(Continua)* Indica la nota obtenida en las asignaturas cursadas en cada semestre.
- **Curricular units 1st/2nd sem (without evaluations):** *(Discreta)* Número de créditos en los que aún no se han realizado exámenes en cada semestre.
- **Unemployment rate:** *(Continua)* Porcentaje de personas desempleadas. En este caso, no se indica si este representa la tasa de desempleo en un año, en un mes concreto. En cualquier caso, no es un dato específico de cada estudiante.
- **Inflation rate:** Medida que se utiliza para cuantificar el aumento promedio de los precios de bienes y servisios en una economía durante un periodo específico de tiempo. Por lo que, no es un dato específico de cada estudiante.

- **GDP:** Medida del valor total de todos los bienes o servicion producidos por el estudiante. Puede tomar valores negativos, que indica una disminución en la producción económica, o un valor positivo que indica todo lo contrario.

- **Target:** Indica si el alumno ha tenido éxito o no, los valores que puede tomar son: 'Dropout' (1) o 'Graduate'(0).

5.2. Análisis de la Base de Datos

A lo largo de esta sección, se describirá de forma detallada la base de datos comentada anteriormente. Así como su aplicación a un algoritmo de aprendizaje supervisado, SVM, para poder predecir si un estudiante acabará graduándose o no. El código se puede ver en B.3.

En primer lugar, ha sido necesario la lectura de la base de datos CSV. (ver Figura 5.1) Debido a la gran cantidad de valores de los que dispone la base de datos mostraremos las 5 primeras filas a falta de algunas variables.

	Marital status	Application mode	Application order	Course	Daytime/evening attendance	Previous qualification	Curricular units 2nd sem (grade)	Curricular units 2nd sem (without evaluations)	Unemployment rate	Inflation rate	GDP	Target
0	1	17	5	171	1	1	0.000000	0	10.8	1.4	1.74	Dropout
1	1	15	1	9254	1	1	13.666667	0	13.9	-0.3	0.79	Graduate
2	1	1	5	9070	1	1	0.000000	0	10.8	1.4	1.74	Dropout
3	1	17	2	9773	1	1	12.400000	0	9.4	-0.8	-3.12	Graduate
4	2	39	1	8014	0	1	13.000000	0	13.9	-0.3	0.79	Graduate

Figura 5.1: Primeras 5 filas de la base de datos.

A continuación, se va a detallar una descripción de las dos clases de variables disponibles en la base de datos. En primer lugar, se va a mostrar un histograma (ver figura 5.2), y detalles de la media y desviación típica (ver cuadro 5.1) de cada una de las variables cuantitativas.

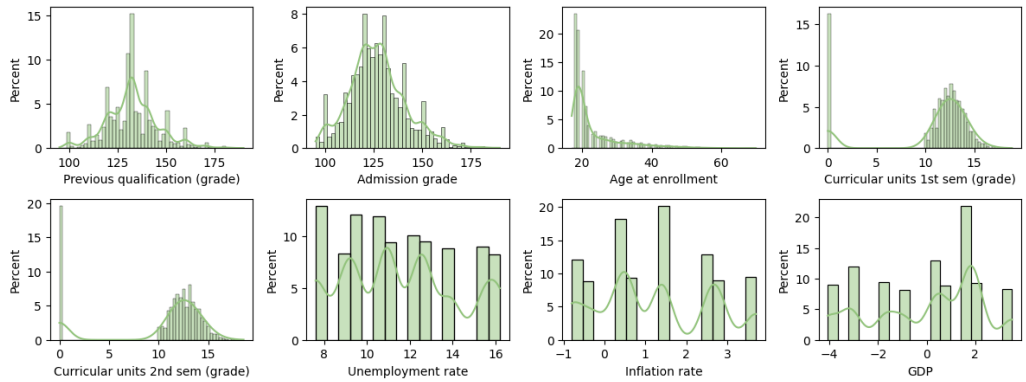


Figura 5.2: Distribución de los atributos cuantitativos.

Variable	Media	σ
Previous qualification (grade)	132.61	13.19
Admission grade	126.98	14.48
Age at enrollment	23.27	7.59
Curricular units 1st sem (grade)	10.64	4.84
Curricular units 2nd sem (grade)	10.23	5.21
Unemployment rate	11.57	2.66
Inflation rate	1.23	1.38
GDP	0.0	2.27

Cuadro 5.1: Media y Desviación típica de los atributos cuantitativos

A continuación, crearemos la matriz de correlación [12] (ver figura 5.3) . Gracias a ella podremos ver que tan bien se correlaciona cada uno de estos atributos cuantitativos entre sí. El valor devuelto será un número entero entre -1 y 1, de manera que aquellas correlaciones más altas estarán cerca de esos valores y las correlaciones más bajas estarán cerca del 0 indicando que las variables no tienen mucha relación entre sí.

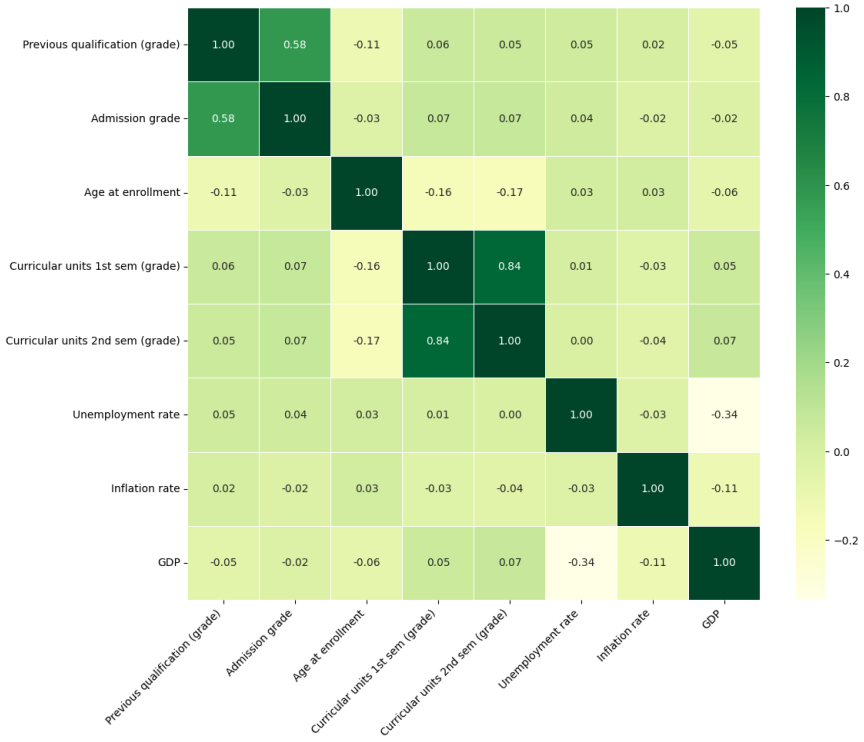


Figura 5.3: Matriz de correlación de las variables cuantitativas.

En cuanto a las variables cualitativas se mostrará en primer lugar tablas de frecuencias para aquellas variables cualitativas dicotómicas:

Displaced		
Valor	**Frecuencia**	**%**
1: yes	2426	54.84
0: no	1998	45.16

Educational special needs		
Valor	**Frecuencia**	**%**
0: no	4373	98.15
1: yes	51	1.15

Debtor		
Valor	**Frecuencia**	**%**
0: no	3921	88.63
1: yes	503	11.37

Tuition fees up to date		
Valor	**Frecuencia**	**%**
1: yes	3896	88.07
0: no	528	11.93

Gender		
Valor	**Frecuencia**	**%**
0: femenino	2868	64.83
1: masculino	1556	35.17

Scholarship holder		
Valor	**Frecuencia**	**%**
0: no	3325	75.16
1: yes	1099	24.84

International		
Valor	**Frecuencia**	**%**
0: no	4314	97.51
1: yes	110	2.49

DayTime/evening attendance		
Valor	**Frecuencia**	**%**
1: daytime	3941	89.08
0: evening	483	10.92

Target		
Valor	**Frecuencia**	**%**
Dropout	4314	50.07
Graduate	2209	49.93

Cuadro 5.2: Tabla de frecuencias de las variables dicotómicas

En cuanto a las variables *'Marital status'*, *'Application mode'*, *'Application order'*, *'Course'*, *'Previous qualification'*, *'Nacionality'*, *'Mother's qualification'*, *'Father's qualification'*, *'Mother's occupation'*, *'Father's occupation'* mostraremos su gráfico de frecuencias (ver figuras 5.4). Para una mejor visualización, en este caso, no mostraremos todos los atributos disponibles en la base de datos, mostraremos solo aquellos cuyas frecuencias superan 20, excepto algunas variables.

Para las variables que indican el número de créditos mostraremos un gráfico de frecuencias agrupados por intervalos de 5 (ver Figura 5.5). De esta manera, se podrá ver de manera más compacta toda la información.

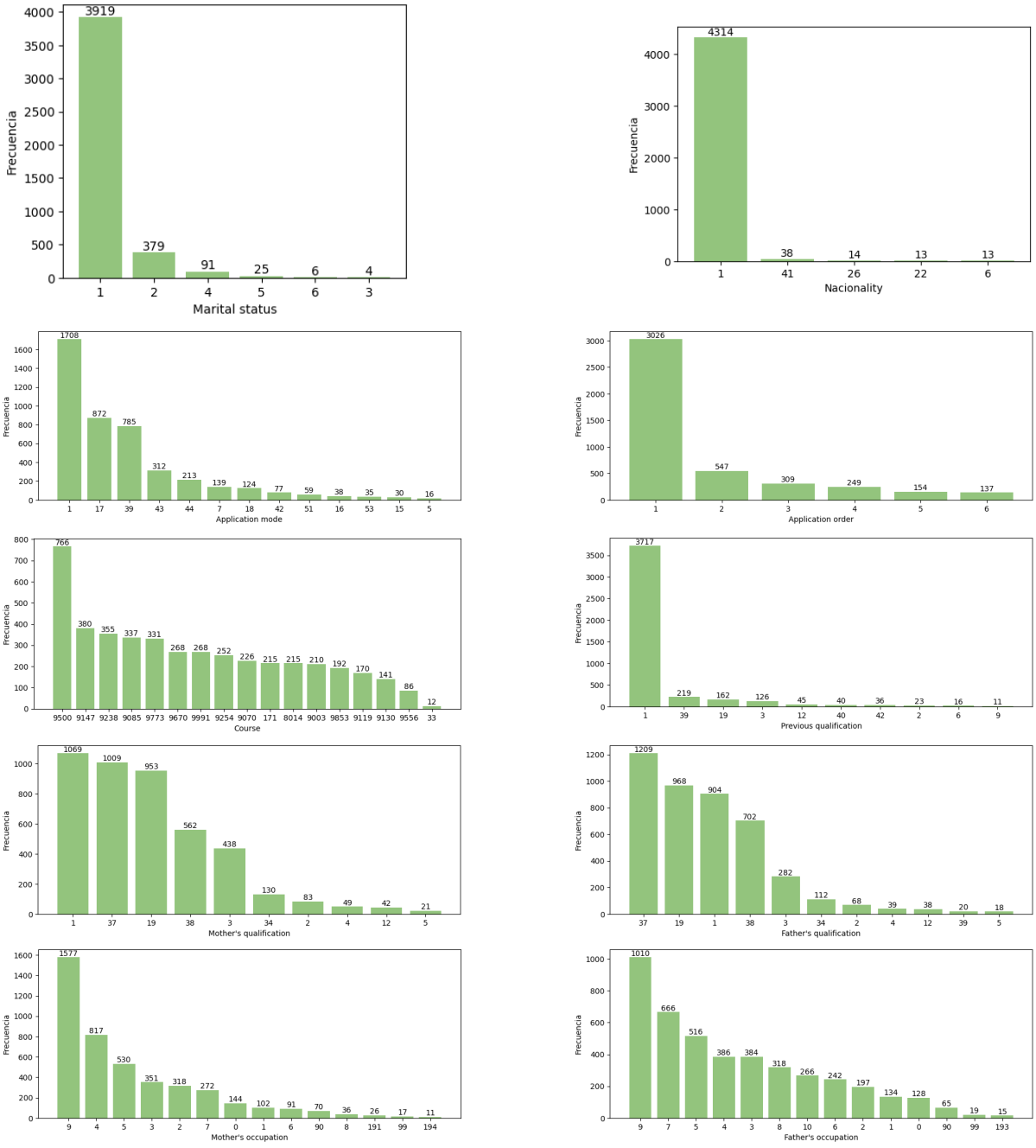


Figura 5.4: Gráfica de frecuencias de las variables cualitativas

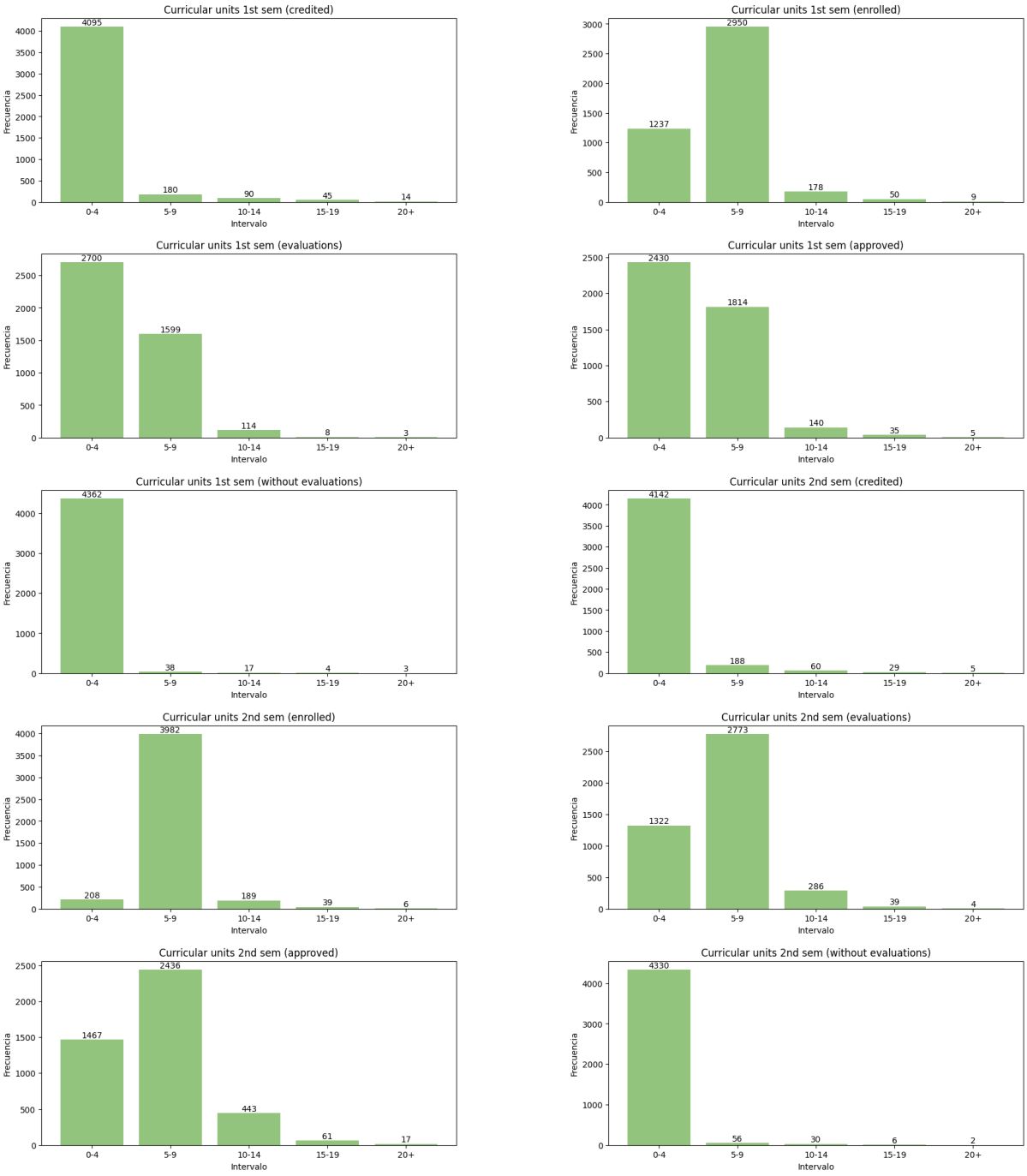


Figura 5.5: Gráfica de frecuencias de las variables cualitativas 2

5.2.1. Evaluación independencias

Como hemos podido ver la base de datos está definida por una gran cantidad de variables. Por ello, en esta sección, vamos a evaluar la relación de cada una de ellas con la variable 'Target', y de esta manera, ver si realmente todas pueden contribuir a discriminar y lograr un buen modelo para predecir el objetivo.

En el caso de las variables continuas se utilizará el test t-Student ([2]) para ver si hay diferencia en la media de las variables entre el grupo de alumnos qe acaban graduándose y los que no.

En este caso, se ha considerado que las diferencias eran significativas si el p-valor es menor a 0,05. Por lo que cualquier p-valor inferior a este implicará un rechazo de la hipótesis nula de que las dos variables son independientes.

En el cuadro 5.3 observamos que las variable *'Unemployment rate'* y *'Inflation rate'* tiene un p-valor superior a 0,05, y por tanto, podemos prescindir de ellas a la hora de aplicar el SVM.

Variable	Dropout		Graduate		p-valor
	Media	σ	Media	σ	
Previous qualification (grade)	131.14	12.87	134.08	13.34	1.15e-13
Admission grade	125.16	14.66	128.79	14.07	6.15e-17
Age at enrollment	24.74	8.12	21.78	6.69	3.61e-39
Curricular units 1st sem (grade)	8.64	5.62	12.64	2.69	8.00e-182
Curricular units 2nd sem (grade)	7.76	5.90	12.69	2.68	2.41e-245
Unemployment rate	11.49	2.72	11.63	2.60	6.79e-02
Inflation rate	1.25	1.39	1.19	1.37	1.48e-01
GDP	-0.07	2.27	0.08	2.26	1.94e-02

Cuadro 5.3: Resultado t-Student para las variables cuantitativas.

En el caso de las variables cualitativas utilizaremos el test 'chi-cuadrado' ([8]) para contrastar su independencia con la variable 'Target'. Tendremos que la hipótesis nula es que ambas variables son independientes, y la alternativa que sí hay relación. Al igual que en el caso anterior, rechazaremos la hipótesis nula en caso de que el p-valor $< 0{,}05$.

En el caso de las variables dicotómicas (ver Cuadro 5.4), dividiremos el conjunto de datos en personas que se graduaron y en personas que abandonaron sus estudios. Y, calcularemos el porcentaje de gente que tiene el valor "1"de cada uno de los dos grupos. Podemos observar que solo hay dos variables que pueden considerarse independientes de la variable *'Target'*: *'Educational special needs'* e *'International'*.

Variable	Dropout (1)	Graduate (1)	p-valor
Daytime/evening attendance	87.27	90.65	1.30e-04
Displaced	49.75	59.77	1.23e-11
Educational special needs	98.74	1.04	5.79e-01
Debtor	18.15	4.56	1.28e-45
Tuition fees up to date	77.47	98.42	1.43e-104
Gender	45.51	24.74	6.38e-47
Scholarship holder	11.92	37.70	5.50e-88
International	2.53	2.44	9.34e-01

Cuadro 5.4: Resultado chi-cuadrado para las variables dicotómicas.

Por último, aplicaremos el chi-cuadrado al resto de variables cualitativas que no hemos usado hasta ahora. Podemos observar en el Cuadro 5.5 que *'Nacionality'* tiene un valor inferior a 0,05. Luego, todas las variables son dependientes de la variable objetivo, y por tanto, importantes para decidir si un estudiante se gradua o no.

Variable	p-valor
Marital status	5.40e-06
Application mode	5.30e-51
Application order	1.31e-10
Course	1.55e-94
Previous qualification	8.36e-11
Nacionality	2.82e-01
Mother's qualification	4.44e-08
Father's qualification	2.012e-09
Mother's occupation	6.27e-06
Father's occupation	2.36e-04

Variable	p-valor
Curricular units 1st sem (credited)	4.34e-03
Curricular units 1st sem (enrolled)	2.75e-61
Curricular units 1st sem (evaluations)	3.99e-100
Curricular units 1st sem (approved)	0
Curricular units 1st sem (without evaluations)	6.26e-07
Curricular units 2nd sem (credited)	1.51e-03
Curricular units 2nd sem (enrolled)	8.84e-69
Curricular units 2nd sem (evaluations)	1.52e-94
Curricular units 2nd sem (approved)	0
Curricular units 2nd sem (without evaluations)	6.07e-08

Cuadro 5.5: Resultado chi-cuadrado para las variables cualitativas.

5.3. Aplicación SVM y Resultados

Una vez tenemos finalizado la descripción de los atributos nos adentramos en el algoritmo SVM para poder realizar las predicciones sobre nuevos estudiantes. En primer lugar, definimos las variables que formarán parte de las 'features', que serán todas aquellas que se utilizarán para predecir la variable objetivo ('Target').

Tomaremos como *'features'* todas aquellas varaiables que han resultado significativas en el apartado anterior.

En primer lugar, evaluaremos el comportamiento de los diferentes kernels descritos en el capítulo 1: gaussiano (*'rbf'*), polinómico (*'poly'*) y lineal (*'linear'*). En este caso, estudiaremos

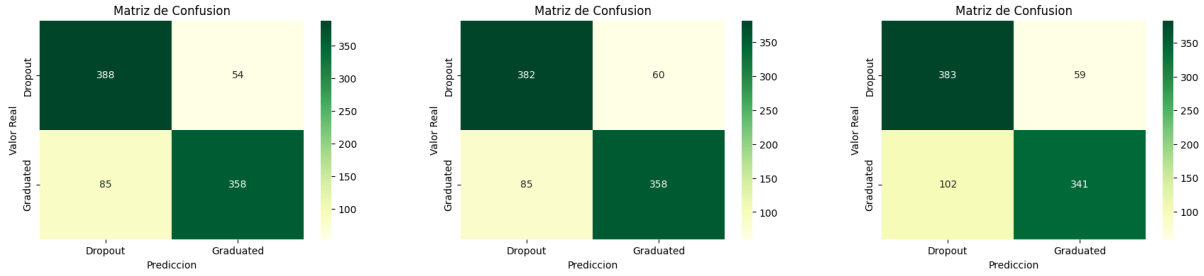
los kernels gaussiano y lineal teniendo en cuenta sus mejores hiperparámetros obtenido por la función `GridSearchCV`. Sin embargo, debido a la gran magnitud de la base de datos, se ha dificultado buscar los mejores hiperparámetros para el polinómico, por lo que tras realizar varias pruebas, consideramos los resultados del cuadro 5.6.

Kernels				
kernel	C	gamma	degree	Precision
rbf	100	0.001	-	84.30 %
linear	10	0.01	-	83.61 %
poly	100	0.01	3	81.80 %

Cuadro 5.6: Evaluación Kernels

Además, mostremos una matriz de confusión para cada uno de ellos (ver figura 5.6), donde podremos ver la precisión de cada uno de ellos observando el número de aciertos y fallos de cada kernel.

En todas ellas, podemos observar en la diagonal el número de aciertos y fallos para cada uno de los Kernels. Podemos observar que el Kernel Gaussiano es ligeramente mejor que el Kernel Lineal y el Kernel Polinómico teniendo 388 aciertos para la clase 'Dropout' (1) y 358 aciertos para la clase 'Graduated' (0).



(a) Kernel Gaussiano (b) Kernel lineal (c) Kernel Polinómico

Figura 5.6: Matriz de Confusión con parámatros por defecto

Capítulo 6

Conclusiones

En conclusión, este trabajo ha demostrado que los espacios de Hilbert con núcleo reproductor son una herramienta fundamental en la resolución de problemas de clasificación en el ámbito del aprendizaje automático y la inteligencia artificial. Su flexibilidad en la representación de datos y su capacidad de generalización les permite abordar problemas complejos de la vida real.

El estudio del fundamento matemático y teórico de los RKHS ha permitido construir una base matemática sólida para comprender y analizar los algoritmos de clasificación basados en estos espacios. Esta ha permitido avanzar en la teoría y la práctica de la clasificación, abriendo nuevas posibilidades de aplicación en diversos ámbitos.

En este trabajo, se ha aplicado la combinación de RKHS y el algoritmo SVM a un problema de clasificación de estudiantes. Sin embargo, las aplicaciones de esta técnica son mucho más amplias. Por ejemplo, en el ámbito de la medicina, donde los RKHS pueden desempeñar un papel crucial en la detección temprana de enfermedades mortales, entre otras.

En resumen, este trabajo me ha dado la oportunidad de explorar y comprender en profundidad los problemas de clasificación y su importancia en la resolución de problemas en la vida real actuales de diversos campos. Esta investigación abre nuevas oportunidades para futuras investigaciones y aplicaciones prácticas, promoviendo el avance y la mejora continua en la resolución de problemas complejos en el mundo actual.

Bibliografía

[1] My Datta Science Blog. Reproducing kernel hilbert spaces machine learning, February.

[2] Ciencia de Datos. Artículo: T-test en python. `https://www.cienciadedatos.net/documentos/pystats10-t-test-python`.

[3] Trevor Hastie Robert Tibshirani Gareth James, Daniela Witten. *An Introduction to Statistical Learning: with Applications in R.* Springer, 8th edition edition, 2017.

[4] Cristopher Heil. *Metrics, Norms, Inner Products, and Operator Theory.* Birkhäuser, 2018.

[5] Sergio Macario Vives Juan José Font Ferrandis, Salvador Hernández Muñoz. *Cálculo.* Universidad Jaume I - Sapientia.

[6] Francisco Heredia Mañas. Los espacios de hilbert en la transformada de fourier, 2019.

[7] Vern I. Paulsen. *An Introduction to the Theory of Reproducing Kernel Hilbert Spaces.* Cambridge University Press, 2016.

[8] ProgrammerClick. Article: Understanding python decorators. `https://programmerclick.com/article/7791535299/`.

[9] Vieira Martins Mónica Machado Jorge Realinho, Valentim and Luís Baptista. Predict students' dropout and academic success. UCI Machine Learning Repository, 2021. DOI: https://doi.org/10.24432/C5MC89.

[10] Joaquín Amat Rodrigo. Máquinas de vector de ssporte (support vector machines, svms, April 2017.

[11] I. Santamaría S. Van Vaerenbergh. Métodos kernel para clasificación, marzo 2018.

[12] Towards Data Science. Article: Seaborn heatmap for visualising data correlations. `https://towardsdatascience.com/seaborn-heatmap-for-visualising-data-correlations-66cbef09c1fe`, Agosto 2022.

[13] Jerome Friedman Trevor Hastie, Robert Tibhirani. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, second edition edition, 2008.

[14] Óscar Blanco. Análisis funcional. PDF.

Anexo A

Anexo I

A.1. Caso $C([a,b])$

En esta sección veremos la demostración de los hechos afirmados en el ejemplo de $C(a,b)$

Veremos la no completitud de la $L2$-norma y que la $L1$-norma no está inducida por el producto interno.

- En primer lugar consideremos la $L2$-norma:

$$\|f\|_2 = \left(\int_a^b |f(x)|^2\, dt\right)^{\frac{1}{2}}$$

Esta norma si que proviene de un producto interno, concretamente del producto definido de la siguiente manera:

$$\langle f, g\rangle = \int_a^b f(t)\cdot\overline{f(t)}\, dt$$

Veremos que no cumple la propiedad de completitud.

Para ello, consideraremos la sucesión $x_n(t) = t^n, \forall n \in \mathbb{N}$.

Queremos demostrar que es una sucesión de Cauchy en el espacio $(C[0,1], \|\cdot\|_2)$. Para ello, asumimos que $n < m$ para asegurar que sea positivo, y se tiene que:

$$\|x_n - x_m\|_2 = \left(\int_0^1 (t^n - t^m)^2\right)^{1/2} dt = \left(\int_0^1 (t^{2n} - 2t^{n+m} + t^{2m})\right)^{1/2} dt =$$

$$= \left[\frac{t^{2n+1}}{2n+1} - \frac{t^{n+m+1}}{n+m+1} + \frac{t^{2m+1}}{2m+1} \right]_0^1 = \left(\frac{1}{2n+1} - \frac{1}{n+m+1} + \frac{1}{2m+1} \right)^{1/2}$$

Entonces, si escogemos N con $\frac{1}{N} < \frac{\epsilon^2}{3}$, y tenemos que $n, m > N$, se cumple que:

$$\|x_n - x_m\|_2 = \left(\frac{1}{2n+1} - \frac{1}{n+m+1} + \frac{1}{2m+1} \right)^{1/2} < \left(\frac{\epsilon^2}{3} + \frac{\epsilon^2}{3} + \frac{\epsilon^2}{3} \right)^{1/2} = \left(\epsilon^2\right)^{1/2} = \epsilon$$

Luego, la sucesión $x_n(t) = t^n$ es de Cauchy. Sin embargo, que sea una sucesión de Cauchy no implica necesariamente que converja en el espacio $(C[0,1], \|\cdot\|_2)$. Esto es fácil verlo, al estudiar el límite de la sucesión cuando $n \to \infty$.

- Si $t = 1$, entonces $\lim_{n\to\infty} x_n(t) = \lim_{n\to\infty} 1^n = 1$.
- Si $0 \leq t < 1$, entonces $\lim_{n\to\infty} x_n(t) = \lim_{n\to\infty} t^n = 0$.

Por lo tanto, la sucesión (x_n) converge puntualmente en el intervalo $[0,1]$ a la función $f(t)$ definida como:

$$f(t) = \begin{cases} 1, & \text{si } 0 \leq t < 1 \\ 0, & \text{si } t = 1 \end{cases}$$

Pero esta función no es continua, por lo que no pertence al espacio $(C[0,1])$. Como la convergencia en $\|\cdot\|_2$ implica convergencia puntual, esta sucesión no puede ser convergente y el espacio no es completo.

Por tanto el espacio $C([a,b])$ que adquiere la $L2$-norma no cumple las propiedades para ser un espacio de Hilbert.

- Ahora, consideramos la L^1-norma:

$$\|f\|_1 = \int_a^b |f(x)|\, dt$$

En este caso, Vamos a ver que la $\|\cdot\|_1$ no está inducida por ningún producto interno. Para ello, considerarmos el espacio $(C[0,1], \|\cdot\|_1)$. Y sean dos funciones pertenecientes a ese espacio, claramente continuas y definidas en el intervalo $[0,1]$:

$$x(t) = 1 - t^2$$

$$y(t) = t^2 - 1$$

Veremos en primer lugar, si está dotado de un producto interno. Para ello, comprobaremos la Ley del Paralelogramo.

$$\|x+y\|^2 + \|x-y\|^2 = \left(\int_0^1 |(1-t^2) + (t^2-1)|\, dt \right)^2 + \left(\int_0^1 |(1-t^2) - (t^2-1)|\, dt \right)^2 =$$

$$= \left(\int_0^1 |2 - 2t^2| \, dt \right)^2 = \left(\left[2t - \frac{2}{3}t^3 \right]_0^1 \right)^2 = \frac{16}{9}$$

Por otro lado, tenemos que:

$$\|x\|^2 + \|y\|^2 = \left(\int_0^1 |(1 - t^2)| \, dt \right)^2 + \left(\int_0^1 |(t^2 - 1)| \, dt \right)^2 =$$

$$= \left(2 \cdot \left[t - \frac{1}{3}t^3 \right]_0^1 \right)^2 = 2 \cdot \frac{16}{9} = \frac{16}{9}$$

Luego, sustituyendo en la Ley del paralelogramo tenemos que:

$$\|x + y\|^2 + \|x - y\|^2 = 2(\|x\|^2 + \|y\|^2) \longrightarrow \frac{16}{9} \neq 2 \cdot \frac{16}{9}$$

Luego, podemos concluir que no está dotado de un producto interno. Y, por tanto, la $L1-$ norma no proviene de un producto interno.

A.1.1. Multiplicadores de Lagrange

A continuación, presentamos un teorema fundamental clásico que nos permite abordar problemas de optimización con restricciones mediante el uso de los multiplicadores de Lagrange. Este procedimiento nos ayuda a encontrar los máximos y mínimos relativos de funciones de múltiples variables bajo ciertas restricciones.

Teorema A.1.1. [5] Sea la función $f : A \subset \mathbb{R}^n \longrightarrow \mathbb{R}$ con A abierto de $\mathbb{R}^n$ y $f \in C^1(A)$, es decir es una función continua que admite una primera derivada. Sea $X = \{x \in A : g(x) = 0\}$, con $g = (g_1, \cdots, g_m)$ de clase C^1. Sea $x_0 \in X$. Si $f_{|_x}$ tiene un extremo local en x_0, entonces existen m escalares $\alpha_1, \cdots, \alpha_n$ tales que el punto x_0 es punto crítico de:

$$L(x) := f(x) + \alpha_1 g_1(x) + \alpha_2 g_2(x) + \cdots + \alpha_m g_m(x)$$

Introduciendo esos escalares como variablees adicionelaes a la función L anterior, se reduce el problema de determinar los puntos críticos de f que cumplen las restricciones al problema de determinar los puntos críticos de la función *lagrangiana*:

$$L(x, \alpha_1, \cdots, \alpha_n) := f(x) + \alpha_1 g_1(x) + \alpha_2 g_2(x) + \cdots + \alpha_m g_m(x)$$

El teorema anterior se puede aplicar a nuestro problema de optimización. Para ello se define la función objetivo f como $f(v) = \frac{1}{2}\|v\|^2$, que estarán sujetas a las resticciones $g_i(v) =$

$\lambda_i(\langle x_i, v\rangle - c) - 1$. Por tanto, nuestro problema de optimización se simplifica resolviendo la función Lagrangiana siguiente:

$$L(x, \alpha_i) = \frac{1}{2}\|v\|^2 - \sum_{i=1}^{n}(\alpha_i \cdot \lambda_i\langle x_i, v\rangle - c) \tag{A.1}$$

Anexo B

Anexo II

B.1. Explicación código

En esta sección, se van a comentar el código realizado en Python en el entorno jupyter para el desarrollo de las gráficas del Capítulo 1, y el procesamiento de la base de datos del Capítulo 5.

En cuanto a las librerías que han sido necesarias se han importado librerías tanto para el desarrollo del análisis como librerías de Machine Learning.

En cuando a las librerías útiles para el análisis se ha utilizado la librería 'Numpy' que posee funciones matemáticas de alto nivel. También, se ha utilizado la librería 'Matplotib' necesaria para generar los gráficos en 2D, concretamente se ha utilizado la función `scatter`.

Y, por último se han utilizado las funciones `'train_test_split'`, `'fit'`, `'score'` de la librería de Machine Learning 'sklearn'. Estas nos han permitido dividir el conjunto de datos en datos de entrenamiento y datos de prueba, a entrenar los datos pertenecientes al conjunto de entrenamiento, y por último a medir la precisión del modelo.

B.1.1. Capítulo 1

```
#**********************IMPORTACION LIBRERIAS**********************#
#Importacion de las librerias necesarias
%matplotlib inline
import numpy as np
```

```
import matplotlib.pyplot as plt
from scipy import stats
import seaborn as sns; sns.set()
from sklearn.datasets import make_blobs
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split

#**********************GENERACION DATOS ALEATORIOS**********************#

#Generacion de datos aleatorios seccion 1.1
X, y = make_blobs(n_samples=130, centers=2, random_state=0, cluster_std=0.50)


#Generacion de datos aleatorios seccion 1.2
X, y = make_blobs(n_samples=150 , centers=2, random_state=0, cluster_std = 0.81)

#Generacion de datos aleatorios seccion 1.3
X, y = make_circles(150, factor=0.2, noise=0.1)

#**********************REPRESENTACION GR\'AFICA**********************#
# Crear una figura con tamanyo personalizado
plt.figure(figsize=(6, 5))

#Graficar los datos
plt.scatter(X[:, 0], X[:, 1], c=y, s=20, cmap='bwr')
plt.show()

#Representacion de varios hiperplanos que separen los datos
xfit = np.linspace(-1, 3.5)
plt.scatter(X[:, 0], X[:, 1], c=y, s=20, cmap='bwr')

for m, b in [(1, 0.65), (0.5, 1.6), (-0.2, 2.9), [0.3, 2]]:
    plt.plot(xfit, m * xfit + b, '-k')

plt.xlim(-1, 3.5);

#**********************ESPECIFICO DE SVM**********************#
#Entrenamiento de los datos
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
    random_state=19)

#Crear clasificador vectores desoporte seccion 1.1 y 1.2
model = SVC('linear')

#Crear clasificador vectores desoporte seccion 1.3
model = SVC(kernel='rbf', C=1E6)

#Entrenamiento modelo
model.fit(X_train, y_train)

#Obtener precision del modelo
precision = model.score(X_test, y_test)
```

```
print(precision)

#**********************PREDICCIONES**********************#

# Crear un diccionario de etiquetas
etiquetas = {0: "azul", 1: "rojo"}

# Preparar el nuevo dato
indice_aleatorio = np.random.randint(len(X_test))
new_data = X_test[indice_aleatorio].reshape(1, -1)

# Realizar la prediccion
predicted_label = model.predict(new_data)

# Imprimir la etiqueta predicha
etiqueta_predicha = etiquetas[predicted_label[0]]
print("Etiqueta predicha:", etiqueta_predicha)

# Graficar el nuevo dato
plt.scatter(new_data[0][0], new_data[0][1], color='yellow', marker='o', s=20,
    label='Nuevo Dato', zorder = 2)

#Graficar el resto de datos
plt.scatter(X[:, 0], X[:, 1], c=y, s=20, cmap='bwr', zorder = 1)

plt.show()


#**********************VISUALIZACION HIPERPLANO DE MARGEN MAXIMO
    **********************#
def visualizacion_hiperplano_optimo(model, ax=None, plot_support=True):
    if ax is None:
        ax = plt.gca()

    # Crear una cuadricula para evaluar el modelo
    xlim = ax.get_xlim()
    ylim = ax.get_ylim()

    x = np.linspace(xlim[0], xlim[1], 30)
    y = np.linspace(ylim[0], ylim[1], 30)
    X, Y = np.meshgrid(x, y)
    xy = np.vstack([X.ravel(), Y.ravel()]).T

    P = model.decision_function(xy).reshape(X.shape)

    #Trazar el limite de decision y los margenes
    ax.contour(X, Y, P, colors=['darkgreen', 'k', 'darkgreen'], levels=[-1, 0,
        1], alpha=0.5, linestyles=['--', '-', '--'])

    #Graficar los vectores de soporte
    if plot_support:
        ax.scatter(model.support_vectors_[:, 0], model.support_vectors_[:, 1], s
```

```
            =300, linewidth=1, facecolors='none')

    ax.set_xlim(xlim)
    ax.set_ylim(ylim)

#Obtenemos los vectores de soporte
vectores = model.support_vectors_

# Puntos que deseas resaltar
highlight_points = vectores;

#Redimensiono la grafica
plt.figure(figsize=(6, 5))


# Graficar los puntos resaltados
plt.scatter(highlight_points[:, 0], highlight_points[:, 1], s=200, facecolors='
    none', edgecolors='black')

plt.scatter(X[:, 0], X[:, 1], c=y, s=20, cmap='bwr')
visualizacion_hiperplano_optimo(model);
```

B.2. Descripción de la base de datos

En este caso, veremos solo aquellos valores que aparecen en las gráficas, es decir, aquellos con una frecuencia mayor a 10 en caso de la variable 'Nacionality' y aquellos con frecuencia mayor a 20 en caso del resto de variables cualitativas

- **Marital status:** *(Discreta)*
  - 1: single.
  - 2: married.
  - 3: widower.
  - 4: divorced.
  - 5: facto union.
  - 6: legally separated.
- **Application mode**: *(Discreta)*
  - 1: 1st phase - general contingent 2 - Ordinance No. 612/93.
  - 17: 2nd phase - general contingent 18.
  - 39: Over 23 years old.

- 43: Change of course.
- 44: Technological specialization diploma holders.
- 7: Holders of other higher courses.
- 18: 3rd phase - general contingent.
- 42: Transfer.
- 51: Change of institution/course 53 - Short cycle diploma holders.
- 16: 1st phase - special contingent (Madeira Island)
- 53: Short cycle diploma holders.
- 15: International student (bachelor).
- 5: - 1st phase - special contingent (Azores Island).

- **Course**:
  - 9500: Nursing.
  - 9147: Management.
  - 9238: Social Service.
  - 9085: Veterinary Nursing.
  - 9773: Journalism and Communication.
  - 9670: Advertising and Marketing Management.
  - 9991: Management (evening attendance).
  - 9254: Tourism
  - 9070: Communication Design
  - 171: Animation and Multimedia Design
  - 8014: Social Service (evening attendance)
  - 9003: Agronomy
  - 9853: Basic Education
  - 9119: Informatics Engineering
  - 9130: Equinculture
  - 9556: Oral Hygiene
  - 33: Biofuel Production Technologies
- **Previous qualification:**
  - 1: Secondary education.
  - 39: Technological specialization course
  - 19: Basic education 3rd cycle (9th/10th/11th year) or equiv.

- 3: Higher education - degree
- 12: Other
- 40: Higher education - degree (1st cycle)
- 42: Professional higher technical course.
- 2: Higher education - bachelor's degree
- 6: Frequency of higher education
- 9: 12th year of schooling - not completed

- **Nacionality:** *(Discreta)*
  - 1: Portuguese.
  - 41: Brazilian.
  - 26: Santomean.
  - 22: Cape Verdean.
  - 6: Spanish.

- **Mother's qualification/Father's qualification:** *(Discreta)*
  - 1: Secondary Education - 12th Year of Schooling or Eq.
  - 37: Basic education 1st cycle (4th/5th year) or equiv.
  - 19: Basic Education 3rd Cycle (9th/10th/11th Year) or Equiv.
  - 38: Basic Education 2nd Cycle (6th/7th/8th Year) or Equiv.
  - 3: Higher Education.
  - 34: Unknown.
  - 2: Higher Education - Bachelor's Degree.
  - 4: Higher Education - Master's.
  - 12: Other.
  - 5: Higher Education - Doctorate.
  - 39: Technological specialization course.

- **Mother's occupation:** *(Discreta)*
  - 9: Unskilled Workers.
  - 4: Administrative staff.
  - 5: Personal Services, Security and Safety Workers and Sellers.
  - 3: Intermediate Level Technicians and Professions.
  - 2: Specialists in Intellectual and Scientific.
  - 7: Skilled Workers in Industry, Construction and Craftsmen.

- 0: Student.
- 1: Representatives of the Legislative Power and Executive Bodies, Directors, Directors and Executive Managers.
- 6: Farmers and Skilled Workers in Agriculture, Fisheries and Forestry.
- 90: Other Situation.
- 8: Installation and Machine Operators and Assembly Workers.
- 191: cleaning workers.
- 99: (blank), que quiere decir que no se indica el trabajo de ninguno de los padres.
- 194: Meal preparation assistants.
- 10: Armed Forces Professions.
- 193: Unskilled workers in extractive industry, construction, manufacturing and transport.

B.3. Código Caso Práctico

Para llevar a cabo el caso práctico haremos uso de las librerías comentadas en el B.1, necesarias para aplicar el algoritmo SVM.

Además, en este caso también se ha realizado un análisis exhaustivo de la base de datos utilizada. Para ello, ha sido necesario la generación de gráficas y tablas de frecuencias para las variables cualitativas, e histogramas y matriz de correalación para el caso de las cuantitativas, para lo que se ha utilizado la librería 'seaborn', la cual pemite crear gráficos estadísticos.

En caso de las variables cuantitativas se ha utilizado la función `'histplot'` disponible en la librería `seaborn` para la generación de los histogramas 5.2. Y, para la creación de la matriz de correlación se ha utilizado la función `'heatmap'`. Y, en el caso de las gráficas (5.4 , 5.5) y tablas de frecuencia (5.2) para las cualitativas simplemente se ha utilizado el método `value_counts()`, el cual cuenta la cantidad de ocurrencias únicas de cada valor en la columna, disponible en la librería 'pandas'. Además se ha utilizado el módulo `'plt'` disponible en la librería 'Matplotib'.

Para el cálculo de las medias y desviaciones típicas de las variables cuantitativas (5.1) se han utilizado las funciones `'mean()'` y `'std()'` disponible en el módulo `'scipy.stats'`.

Luego, debido a la gran cantidad de variables que contiene la base de datos, ha sido necesario realizar una serie de tests que nos han permitido conocer si realmente todas las columnas podían contribuir al problema de clasificación.

Tanto en las variables cuantitativas como cualitativas se ha hecho uso del módulo `'scipy.stats'`, este contiene una gran cantidad de ditribuciones de probabilidad, estadísticas de resumen. En

este caso, se ha utilizado la función `'ttest_ind'` (5.3), disponible en este módulo, para realizar la pueba de hipótesis de la independencia de medias. En caso de las variables cualitativas se ha utilizado la función `'chi2_contingency'` (5.4, 5.5) del mismo módulo, con el que se ha calculado el p-valor con el fin de ver cuales son las variables que son dependientes de la variable 'Target'.

Por último, se ha aplicado el algoritmo SVM con distintos Kernels con tal de resolver el problema de clasificación deseado.

En primer lugar, hemos eliminado todas aquellas columnas que han resultado en los tests anteriores independientes de la variable `'Target'`, con la función `'df.drop(columnas_eliminar, axis = 1)'`.

A continuación, con la base de datos resultante hemos realizado una serie de transformaciones a los datos antes de crear el modelo SVM. En primer lugar, hemos aplicado a las variables dicotómicas la función `'get_dummies'` con el fin de convertir la variable en dos variables ficiticias de 1 y 0. Y, además, se han escalado las variables cuantitativas con la función `'StandardScaler()'` para tener todas ellas en la misma escala, es decir, centrarlas todas alrededor de 0 y con la misma varianza.

Una vez, transformados los datos y dejar la base de datos preparada seleccionamos nuestras variables 'features' y nuestra variable objetivo 'Target'. A continuación con la función `'train_test_split'` dividimos el conjunto de datos en un 80 % destinado a conjunto de entrenamiento y un 20 % conjunto de prueba.

Luego, se han buscado los mejores hiperparámetros para el caso del 'Kernel lineal' y el 'Kernel gaussiano (rbf)' con el uso de la función `GridSearchCV` disponible en la librería de Machine Learning 'sklearn'. Una vez, encontrados los mejores hiperparámetros se ha entrenado los modelos con la función `'fit'` de la misma librería.

En caso del kernel polinómico no ha sido posible ejecutar la misma función debido al gran tamaño de la base de datos, por ello he cogido unos parámetros elegidos tras varias pruebas de tal manera que la predicción sea lo mejor posible.

Finalmente, se han generado las matrices de confusión para ver la cantidad de aciertos y fallos de cada uno de los Kernels con los hiperparámetros encontrados de cada uno de ellos. Para ello, se ha generado la matriz con la función `'confusion_matrix(y_test)'` disponible también en la librería 'sklearn'. Y, para la visualización de esta se ha hecho uso de la función `'heatmap'` disponible en la librería 'seaborn'.

```
#**********IMPORTACION LIBRERIAS**********#
import pandas as pd
import numpy as np
```

```
import seaborn as sns
import re
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import accuracy_score
from sklearn.model_selection import GridSearchCV
from scipy import stats
from scipy.stats import chi2_contingency

#**********LECTURA BASE DE DATOS**********#
df = pd.read_csv("data2.csv", sep = ';')


#Mostrar las primeras filas de la base de datos
df.head()

#**********DESCRIPCION DE LAS VARIABLES CUANTITATIVAS
    **********#
cuantitativas = [ "Previous qualification (grade)", "Admission grade", "Age at
    enrollment",  "Curricular units 1st sem (grade)","Curricular units 2nd sem (
    grade)",  "Unemployment rate",
                "Inflation rate", "GDP"]

aux = df[cuantitativas]
#Distribuciones de los atributos cuantitativos
plt.figure(figsize=(12,13))
for i in range(1, len(aux.columns) + 1 ):
    plt.subplot(6, 4, i)
    sns.histplot(aux.iloc[:, i-1], kde=True, stat = 'percent', color='#93c47d')
    plt.tight_layout()
    plt.plot()

#Media y desviacion tipica
for col in aux.columns:
    media = round(df[col].mean(), 2)
    desviacion_tipica = round(df[col].std(), 2)

    print(col)
    # Imprimir los resultados
    print("Media: ", media)
    print("Desviacion estandar: ", desviacion_tipica)

#Matriz de correlaciones
corr_matrix = aux.corr()

fig, ax = plt.subplots(figsize=(12, 10))
heatmap = sns.heatmap(corr_matrix, annot=True, linewidths=0.5, fmt=".2f", cmap="
    YlGn")
```

```
# Rotar las etiquetas del eje x
heatmap.set_xticklabels(heatmap.get_xticklabels(), rotation=45,
    horizontalalignment='right')

plt.show()

#**********DESCRIPCION DE LAS VARIABLES CUALITATIVAS
    **********#


#Tablas de las variables binarias
tablas = ["Daytime/evening attendance", "Displaced", "Educational special needs",
     "Debtor",
        "Tuition fees up to date", "Gender", "Scholarship holder",
        "International", "Target"]

aux2 = df[tablas]

for col in aux2.columns:
    # Crea la tabla de frecuencias
    tabla_frecuencias = df[col].value_counts().reset_index()

    # Renombra las columnas de la tabla de frecuencias
    tabla_frecuencias.columns = ['Valor', 'Frecuencia']

    # Calcula el porcentaje
    tabla_frecuencias['Porcentaje'] = round((tabla_frecuencias['Frecuencia'] /
        len(df)) * 100, 2)

    print(col)
    # Imprime la tabla de frecuencias
    print(tabla_frecuencias)

#Graficas de frecuencias de las variables con muchos valores distintos
graficas = ["Marital status", "Application mode", "Application order", "Course","
    Previous qualification", "Nacionality",
        "Mother's qualification", "Father's qualification",
        "Mother's occupation", "Father's occupation"]
aux3 = df[graficas]

for i in aux3.columns:
    frecuencias = aux3[i].value_counts()
    atributos_frecuentes = frecuencias[frecuencias > 20].index

    if len(atributos_frecuentes) > 0:
        valores_str = atributos_frecuentes.astype(str)
        plt.figure(figsize=(9, 4))
        plt.bar(valores_str, frecuencias[atributos_frecuentes], color='#93c47d')

        # Agregar etiquetas numericas
        for x, y in enumerate(frecuencias[atributos_frecuentes]):
            plt.text(x, y, str(y), ha='center', va='bottom')
```

```
        # Agregar etiquetas y titulo
        plt.xlabel(i)
        plt.ylabel('Frecuencia')


        # Mostrar la grafica
        plt.show()



#Grafica de frecuencias de la variable 'Marital Status'

aux4 = df["Marital_status"]
frecuencias = pd.Series(data=aux4.value_counts(), name = i)

valores_str = frecuencias.index.astype(str)
plt.figure(figsize=(9,4))

plt.bar(valores_str, frecuencias.values, color = '#93c47d')

# Agregar etiquetas numericas
for x, y in enumerate(frecuencias.values):
    plt.text(x, y, str(y), ha='center', va='bottom')

# Agregar etiquetas y titulo
plt.xlabel('Marital_status')
plt.ylabel('Frecuencia')

# Mostrar la grafica
plt.show()


#Grafica de frecuencias de la variable 'Nacionality'

aux5 = df["Nacionality"]
frecuencias = aux5.value_counts()
atributos_frecuentes = frecuencias[frecuencias > 10].index

if len(atributos_frecuentes) > 0:
    valores_str = atributos_frecuentes.astype(str)
    plt.figure(figsize=(9, 4))
    plt.bar(valores_str, frecuencias[atributos_frecuentes], color='#93c47d')

    # Agregar etiquetas numericas
    for x, y in enumerate(frecuencias[atributos_frecuentes]):
        plt.text(x, y, str(y), ha='center', va='bottom')

    # Agregar etiquetas y titulo
    plt.xlabel(i)
    plt.ylabel('Frecuencia')
```

```
    # Mostrar la grafica
    plt.show()

#Grafica de frecuencia por intervalos

creditos = ["Curricular units 1st sem (credited)",
       "Curricular units 1st sem (enrolled)",
       "Curricular units 1st sem (evaluations)",
       "Curricular units 1st sem (approved)",
       "Curricular units 1st sem (without evaluations)",
       "Curricular units 2nd sem (credited)",
       "Curricular units 2nd sem (enrolled)",
       "Curricular units 2nd sem (evaluations)",
       "Curricular units 2nd sem (approved)",
       "Curricular units 2nd sem (without evaluations)"]

aux6 = df[creditos]

for i in aux6.columns:
    # Obtener los valores de la columna actual como strings
    valores = aux6[i].astype(int)

    # Agrupar los valores en intervalos numericos
    grupos = pd.cut(valores, bins=5, labels=False, right=False)

    # Definir las etiquetas de los grupos en el orden deseado
    etiquetas = ['0-4', '5-9', '10-14', '15-19', '20+']

    # Crear una columna categorizada con los intervalos y el orden deseado
    intervalos_categorizados = pd.Categorical.from_codes(grupos, etiquetas,
        ordered=True)

    # Crear un DataFrame con la columna categorizada y las frecuencias
    df_frecuencias = pd.DataFrame({'Intervalo': intervalos_categorizados})
    frecuencias = df_frecuencias['Intervalo'].value_counts().sort_index()

    plt.figure(figsize=(9,4))
    # Mostrar la grafica de barras
    plt.bar(frecuencias.index, frecuencias.values, color='#93c47d')

    # Agregar etiquetas numericas
    for x, y in enumerate(frecuencias.values):
        plt.text(x, y, str(y), ha='center', va='bottom')

    # Agregar etiquetas y titulo
    plt.xlabel('Intervalo')
    plt.ylabel('Frecuencia')
    plt.title(i)

    # Mostrar la grafica
    plt.show()
```

```
#*********CHI CUADRADO VARIABLES DICOTOMICAS***********#

#Tabla de frecuencias para el grupo de graduados y el grupo de abandonos.

Dropout = df.loc[df['Target'] == "Dropout"]
datos_drop = Dropout[tablas]

frec_drop = []

for i in datos_drop.columns:
    # Crea la tabla de frecuencias
    tabla_frecuencias = datos_drop[i].value_counts().reset_index()

    # Renombra las columnas de la tabla de frecuencias
    tabla_frecuencias.columns = ['Valor', 'Frecuencia']

    # Calcula el porcentaje
    tabla_frecuencias['Porcentaje'] = round((tabla_frecuencias['Frecuencia'] /
        len(datos_drop)) * 100, 2)
    frec_drop.append((i, tabla_frecuencias))

# Imprime las tablas de frecuencias
for i, tabla in frec_drop:
    print("Columna:", i)
    print(tabla)
    print()

Graduate = df.loc[df['Target'] == "Graduate"]

datos_grad = Graduate[tablas]


frec_grad = []
for i in datos_grad.columns:
    # Crea la tabla de frecuencias
    tabla_frecuencias = datos_grad[i].value_counts().reset_index()

    # Renombra las columnas de la tabla de frecuencias
    tabla_frecuencias.columns = ['Valor', 'Frecuencia']

    # Calcula el porcentaje
    tabla_frecuencias['Porcentaje'] = round((tabla_frecuencias['Frecuencia'] /
        len(datos_drop)) * 100, 2)
    frec_grad.append((i, tabla_frecuencias))

# Imprime las tablas de frecuencias
for i, tabla in frec_grad:
    print("Columna:", i)
    print(tabla)
    print()
```

```
#Chi-cuadrado para evaluar la independencia con la variable "Target"
var = datos.drop('Target', axis = 1)

res_chi = []

for col in datos.columns:
    if col != 'Target':
        # Crear una tabla de contingencia para las variables 'Target' y '
            Respuesta'
        tabla_contingencia = pd.crosstab(datos['Target'], var[col])

        # Calcular el test de chi-cuadrado
        chi2, p_valor, _, _ = chi2_contingency(tabla_contingencia)

        res_chi.append((col, chi2, p_valor))

# Imprimir los resultados
tabla_res = pd.DataFrame(res_chi, columns = ['Variable', 'Estadistico chi-
    cuadrado','p-valor'])

print(tabla_res)

#********CHI-CUADRADO VARIABLES CUALITATIVAS***********#
Dropout = df.loc[df['Target'] == "Dropout"]
datos_drop1 = Dropout[d]


Graduate = df.loc[df['Target'] == "Graduate"]
datos_grad2 = Graduate[d]


datos2 = pd.concat([datos_drop1, datos_grad2], ignore_index = True)


from scipy.stats import chi2_contingency

var1 = datos2.drop('Target', axis = 1)

res_chi1 = []

for col in datos2.columns:
    if col != 'Target':
        # Crear una tabla de contingencia para las variables 'Target' y '
            Respuesta'
        tabla_contingencia = pd.crosstab(datos2['Target'], var1[col])

        # Calcular el test de chi-cuadrado
        chi2, p_valor, _, _ = chi2_contingency(tabla_contingencia)

        res_chi1.append((col, chi2, p_valor))
```

```
# Imprimir los resultados
tabla_res = pd.DataFrame(res_chi1, columns = ['Variable', 'Estadistico chi-
    cuadrado','p-valor'])

print(tabla_res)

#********CHI-CUADRADO VARIABLES CUANTITATIVAS**********#
from scipy import stats
from bioinfokit.analys import stat

Dropout = df1.loc[df1['Target'] == 1]
Graduate = df1.loc[df1['Target'] == 0]

medias_drop = Dropout[cuantitativas].mean()
sd_drop = Dropout[cuantitativas].std()

medias_grad = Graduate[cuantitativas].mean()
sd_grad = Graduate[cuantitativas].std()


resultado = []

for v in cuantitativas:
    t_stat , p_value = stats.ttest_ind(Dropout[v], Graduate[v])
    resultado.append((v, medias_drop[v], medias_grad[v], sd_drop[v], sd_grad[v],
        p_value))

tabla = pd.DataFrame(resultado,  columns=['Variable', 'Media Abandono', 'Media
    Graduacion', 'sd Abandono', 'sd Graduado','p-valor'])
print(tabla)

#************************ESPECIFICO SVM************************#

#Se eliminan las columnas no relevantes
columnas_eliminar = [ 'Nacionality', 'Unemployment rate', 'Inflation rate','
    Educational special needs', 'International']

df2 = df.drop(columnas_eliminar, axis = 1)

columnas_categoricas = ["Daytime/evening attendance", "Displaced", "Debtor", "
    Tuition fees up to date", "Gender", "Scholarship holder"]
df2 = pd.get_dummies(df2, columns = columnas_categoricas)


# Seleccionar las caracteristicas y la variable objetivo
X = df2.drop('Target', axis = 1)
y = df2['Target']
```

```
#Definimos el conjunto de entrenamiento y de prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
    random_state = 150, stratify = y)

columnas_escalar = [ "Previous qualification (grade)", "Admission grade", "Age at
    enrollment",
                "Curricular units 1st sem (grade)","Curricular units 2nd sem (
                    grade)"]

from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train_array = sc.fit_transform(X_train.values)
X_train = pd.DataFrame(X_train_array, index=X_train.index, columns=X_train.
    columns)
X_test_array = sc.transform(X_test.values)
X_test = pd.DataFrame(X_test_array, index=X_test.index, columns=X_test.columns)


# Define los hiperparametros a ajustar
param_grid = {'C': [0.01, 0.1, 1, 10, 100], 'gamma': [0.001, 0.0001, 0.01, 0.1,
    1, 10, 100],
              'kernel': ['rbf', 'linear']}
# Crear un clasificador SVC con kernel gaussiano
svc = SVC()

# Realizar la busqueda de hiperparametros
grid_search = GridSearchCV(svc, param_grid, cv=5)
grid_search.fit(X_train, y_train)

# Obtener los mejores hiperparametros y el mejor modelo
best_params = grid_search.best_params_
best_model = grid_search.best_estimator_

# Realizar predicciones en el conjunto de prueba utilizando el mejor modelo
y_pred = best_model.predict(X_test)

# Calcular la precision
accuracy = accuracy_score(y_test, y_pred)
print("Mejores hiperparqmetros:", best_params)
print("Precision:", accuracy)

#Resultados
df_results = grid_search.cv_results_
df_results = pd.DataFrame(df_results)
df_results = df_results[['param_C', 'param_gamma', 'param_kernel', '
    mean_test_score']]
df_results.columns = ['C', 'gamma', 'kernel', 'Accuracy']
df_results = df_results.sort_values('Accuracy', ascending=False)

print(df_results)

# Entrenamiento del modelo Linear y Gaussiano con los hiperparametros
```

```
model_linear = SVC(kernel='linear', C=10, gamma = 0.01)
model_linear.fit(X_train, y_train)

#Entrenamiento kernel polinomico con parametros
model_poly = SVC(kernel = 'poly', C = 100, gamma = 0.01, degree = 3)
model_poly.fit(X_train, y_train)

print(accuracy_score(y_test, model_rbf.predict(X_test)))
print(accuracy_score(y_test, model_linear.predict(X_test)))

print(accuracy_score(y_test, model_poly.predict(X_test)))



#**********MATRIZ CONFUSION**********#
# Crear un nuevo clasificador SVM con los mejores hiperpar metros encontrados
param_rbf = model_rbf.predict(X_test)
param_lineal = model_linear.predict(X_test)
param_poly = model_poly.predict(X_test)

y_pred = []

# Obt n las predicciones del modelo ajustado
y_pred.append(param_rbf)
y_pred.append(param_lineal)
y_pred.append(param_poly)

for i in y_pred:
    # Calcular la matriz de confusion
    confusion = confusion_matrix(y_test, i)
    fig, ax = plt.subplots(figsize=(6,4))

    # Visualizar la matriz de confusion
    labels = ['Dropout', 'Graduated'] # Etiquetas de las clases
    sns.heatmap(confusion, annot=True, fmt='d', cmap='YlGn', xticklabels=labels,
        yticklabels=labels)
    plt.xlabel('Prediccion')
    plt.ylabel('Valor Real')
    plt.title('Matriz de Confusion')
    plt.show()
```