
***Facultad de Ciencias***

# Un ejemplo de uso del Análisis Estadístico de Datos Funcionales (An example of the use of Statistical Functional Data Analysis)

Trabajo de Fin de Grado
para acceder al

**GRADO EN MATEMÁTICAS**

**Autor: Celia de la Cueva Rodríguez**

**Director: Juan Antonio Cuesta Albertos**

**Septiembre - 2021**

# Resumen

Se realiza una introducción a la teoría del análisis estadístico de datos funcionales. Posteriormente se estudia una aplicación de una de las técnicas de clasificación más usadas, la de los $k$-vecinos más próximos, para clasificar a ciertos tumores de seno como benigno o maligno atendiendo a los datos obtenidos mediante un espectrograma de cada paciente. En esta última parte se reproduce el estudio que hicieron Cuesta-Albertos y Fraiman (ver [7]).
**Palabras clave:** Análisis de Datos Funcionales, Espectrograma, $k$-vecinos más próximos, Problemas de clasificación, Tumores de seno, ...

# Abstract

An introduction to the Theory of Statistical Analysis of Functional Data is given. Subsequently, an application of one of the most widely used techniques, called $k$-nearest neighbors, is presented: this method will be used to classify certain breast tumors as benign or malignant based on data obtained from a spectrogram of each patient. In this last part I am reproducing the study carried out by Cuesta-Albertos and Fraiman (see [7]).
**Key words:** Funcional Data Analysis, Classification problems, $k$-nearest neighbors, Breast tumour, Spectrogram...

# Índice general

# Capítulo 1

# Introducción

## 1.1. El cáncer de mama

El cáncer es una enfermedad debida a un crecimiento sin control de ciertas células anómalas de forma que sobrepasan en número a las células normales. Las células anormales pueden formar masas sólidas que llamamos tumores. En ocasiones después de aparecer de forma localizada, se extienden a otros tejidos adyacentes (la metástasis). Si el paciente no recibe un tratamiento adecuado o no se detecta a tiempo, con frecuencia conduce al fallecimiento. Desde hace muchos años, el cáncer es una de las enfermedades de mayor importancia en salud pública. Concretamente se trata de la segunda causa de muerte a nivel mundial y se estima que en los países desarrollados acabe siendo la principal. (Ver [10])

El cáncer de mama es el tumor maligno más frecuente entre las mujeres de todo el mundo con más de 2.2 millones de casos en 2020, de los cuales entorno a 685000 mujeres no sobrevivieron. En España es el tipo de cáncer que más muertes produce en mujeres con casi 6500 fallecimientos cada año. Si tenemos en cuenta ambos sexos, es el segundo tipo de cáncer más común, tan solo por detrás del de pulmón. La REDECAN[1] estima alrededor de 33000 nuevas incidencias en 2021 (ver [12]).

Con esta introducción quería resaltar la importancia de poder tratar esta enfermedad, pero, sobre todo, la de poder detectarla a tiempo. El ser capaces de localizar el cáncer en sus inicios es una gran ventaja ya que las probabilidades de supervivencia son más altas. Esto se debe a que no da tiempo a que la enfermedad se haya desarrollado mucho, bajando las opciones de que se haya extendido a otros órganos del cuerpo. Así, los diferentes tratamientos que se aplican para combatir el cáncer son más efectivos. Según la AECC[2], en el caso

[1] Red Española de Registros de Cáncer
[2] Asociación Española Contra el Cáncer

del cáncer de mama, la detección temprana puede reducir entre el 25 % y el 31 % la mortalidad (ver [3]). Como se comenta en la tesis doctoral del D. Juan Bayo Calero de la Universidad de Huelva en [8], las matemáticas, así como la informática, han ayudado a la hora de la detección precoz del cáncer diseñando una fórmula con la que con una simple analítica en una mujer se pueda asegurar si es o no cancerosa, además del extendido uso de los modelos matemáticos.

## 1.2. El papel de la estadística en el diagnóstico del cáncer

Este trabajo entra en el diagnóstico del tumor maligno. Pretende ilustrar algunas de las posibilidades de la aplicación de las matemáticas en general, y de la estadística en particular, a temas relevantes de la actualidad. Para poder abordar este problema se hace uso de diferentes partes de las matemáticas. Principalmente se va a basar en la teoría del Análisis de Datos Funcionales que se apoyará en la teoría de los espacios de Hilbert y en las técnicas de clasificación.

La importancia de los datos funcionales ha ido creciendo durante los últimos años debido, principalmente, a que los avances tecnológicos han posibilitado la toma de datos de forma prácticamente continua durante largos periodos de tiempo. Esto ha provocado la necesidad de resolver nuevos problemas estadísticos en los que los datos son funciones.

Los objetivos del análisis de datos funcionales son los mismos que los de la estadística convencional: formular un problema de forma que sea manejable para posteriormente analizarlo con mayor facilidad, conseguir representar los datos de forma que destaquen sus aspectos más relevantes, construir modelos,...

Para poder tratar los problemas de datos funcionales, se ha hecho uso de los espacios de Hilbert. Esto es debido a que tienen muy buenas propiedades matemáticas, es decir, la estructura de Espacio de Hilbert facilita los cálculos. En este contexto también nos interesa tener un espacio que no solo sea de Hilbert, sino también separable, ya que todo Espacio de Hilbert separable admite bases numerables y la existencia de estas bases simplifica notablemente el manejo de estos datos.

Se van a estudiar funciones aleatorias, medidas sobre un intervalo compacto. Por tanto, dado el intervalo $[a, b] \subset \mathbb{R}$, podríamos utilizar el conjunto de las funciones continuas en $[a, b]$, o el de las acotadas,... con una topología adecuada. Sin embargo, con el objetivo de disponer de las propiedades de los espacios de Hilbert (los conjuntos mencionados con las métricas adecuadas no son espacios de Hilbert), estos conjuntos no nos sirven y debemos tomar otro parecido a uno de estos. Para ello, tomaremos una compleción de uno de estos conjuntos, en

concreto del de las funciones continuas, $C[a,b]$, con respecto a la topología de la norma. Esta topología es la topología definida por la distancia asociada a su norma. Este va a ser el conjunto $L^2[a,b]$, que se define como:

$$L_2[a,b] := \left\{ f : [a,b] \to \mathbb{R} \text{ medible y tal que } \int_a^b f^2(t)dt < +\infty \right\}. \qquad (1.1)$$

Pasemos a exponer el objetivo del estudio. Supongamos que disponemos de $n$ datos. Existen varios objetivos en los análisis con estas características. Para este trabajo nos vamos a centrar en el caso en que se estudia la relación entre los datos. Estudiaremos el problema de clasificar datos funcionales, que consiste en agrupar elementos en grupos homogéneos, es decir, clasificar los datos en ciertos grupos.

Hay diversos problemas de clasificación de datos, pero en este trabajo trataremos el análisis discriminante, también conocido como clasificación supervisada. Este modelo trata de clasificar los datos observados en distintos grupos que han sido definidos previamente.

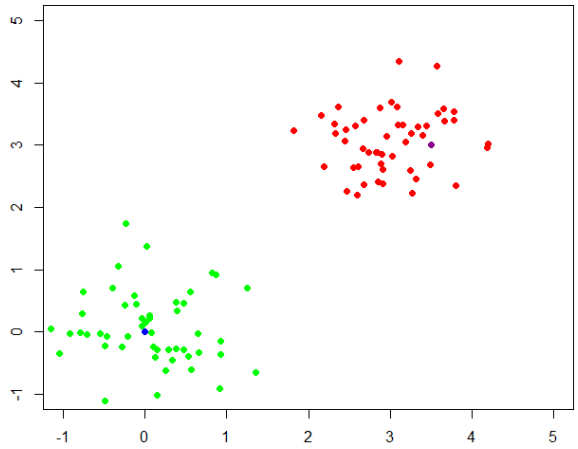


Figura 1.1: Gráfica donde podemos observar una *training sample* y dos datos a clasificar

Hablando un poco más precisamente, el problema al que nos enfrentamos es el siguiente. A partir de una base de datos perfectamente clasificada en dos distribuciones, el objetivo es predecir la distribución a la que pertenece una nueva observación basándonos en la semejanza de esta con los elementos de la muestra de entrenamiento, llamada *training sample*, que se trata de la muestra de datos con lo que se ha probado el modelo. Existen muchos procedimientos para llevar a cabo esta tarea (árboles de decisión, modelos de regresión, *suporting vector machine*, redes neuronales,...), pero en este trabajo nos vamos a limitar a utilizar

el algoritmo de los $k$-vecinos más próximos, que de ahora en adelante llamaremos $k$-NN[3] por sus siglas en inglés. Es uno de los algoritmos más usados para problemas de clasificación. Se basa en la idea de que los datos procedentes de una misma distribución van a estar próximos entre sí y relativamente alejados de los pertenecientes a otras. Esta técnica depende de que esta suposición sea lo suficientemente cierta para que el algoritmo sea útil. En la Figura 1.1 podemos observar una *trainig sample* dividida en dos poblaciones. Una de ellas está representada por puntos verdes y la otra por puntos rojos. Aparecen también dos puntos en otro color, estos son los puntos a clasificar.

Con la teoría matemática que se va a presentar a lo largo de este trabajo, se estará en condiciones de llevar a cabo un estudio que ayuda a detectar si un tumor es maligno o benigno. Se hará uso de espectrogramas para efectuar el estudio. Un espectrograma es el resultado de tomar una muestra de un tejido, dividirlo hasta el nivel molecular y medir el número de moléculas que contiene para cada posible relación masa/carga eléctrica. Para ello se utiliza un aparato llamado espectrómetro que, en los datos analizados, está graduado para medir en el intervalo $[699.99, 12000]$.
En particular, el estudio consistirá en, dada una muestra de entrenamiento adecuadamente clasificada, tratar de clasificar otras observaciones mediante el $k$-NN con la ayuda de la herramienta RStudio.

## 1.3. En este trabajo

En el presente trabajo se analizarán los resultados de espectrogramas que se obtuvieron en varias mujeres con tumores de pecho para poder concluir si los tumores eran malignos o no. Para ello, se utilizará el análisis estadístico de datos funcionales. Es de resaltar que, como veremos perfectamente en la práctica, este ejemplo no permite aplicar directamente la teoría bajo consideración, sino que tendremos que aplicar ciertas modificaciones que permiten mejorar los resultados de análisis. Este estudio está estructurado en cinco capítulos en los que mostraremos el procedimiento para hacer dicho análisis. El resto del trabajo se organiza como sigue:
En el segundo capítulo explicaremos las nociones necesarias de espacios de Hilbert.
En el Capítulo 3 entramos en materia: en él se exponen las principales nociones de clasificación de datos que debemos conocer para el estudio que realizaremos más adelante y explicaremos el método de los $k$-vecinos más próximos.
El siguiente capítulo incluye nociones básicas de la teoría del Análisis Funcional de Datos, que en adelante denotaremos FDA[4].
Y por último en el Capítulo 5, analizaremos los datos mencionados utilizando conceptos explicados previamente. Este capítulo contiene la explicación detallada del procedimiento utilizado. Primero se prepara la muestra para poder

[3] $k$-Nearest Neighbours
[4] Functional Data Analysis

realizar un mejor estudio de los datos. A continuación se especifica la forma de elegir una métrica que se adecue bien a nuestro problema. Más adelante se propone un modo de trabajar con el método de los $k$-vecinos más próximos, que parece el idóneo para nuestra situación. Para finalizar, se exponen los resultados y se presentan algunas conclusiones que se deducen de los mismos.

# Capítulo 2

# Espacios de Hilbert

A lo largo de este capítulo se presentan los conceptos matemáticos básicos para el manejo de los datos funcionales.

En matemáticas, cuando se trabaja en dimensiones finitas, se utilizan espacios Euclídeos. Pero, cuando trabajamos con dimensiones infinitas, los espacios euclídeos no son suficientes para entender cómo medir y tomar distancias. Aquí es donde entra los Espacios de Hilbert, ya que generalizan la noción de espacio euclídeo, en otras palabras, nos da algunas reglas para poder trabajar con dimensiones infinitas.

Los espacios de Hilbert se definen a partir de un producto escalar, por ello vamos a empezar definiendo el producto escalar. Aunque se puede definir sobre cualquier cuerpo, aquí nos limitamos al caso del cuerpo de los números reales.

**Definición 2.1** *Sea $\mathcal{H}$ un espacio vectorial sobre el cuerpo de los números reales, $\mathbb{R}$. Un producto escalar en $\mathcal{H}$ es una operación*

$$\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R} \tag{2.1}$$

*que satisface:*

1. $\langle y, x \rangle = \langle x, y \rangle$, $\forall x, y \in \mathcal{H}$
2. $\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$ $\forall x, y, z \in \mathcal{H}$
3. $\langle \lambda x, y \rangle = \lambda \langle x, z \rangle$, $\forall x, y \in \mathcal{H}$ $y$ $\forall \lambda \in \mathbb{R}$
4. $\langle x, x \rangle > 0$, $\forall x \neq 0 \in \mathcal{H}$

Se dice que el par $(\mathcal{H}, \langle \cdot, \cdot \rangle)$ es un *espacio preHilbert*.

**Definición 2.2** *Dado un espacio vectorial E, una norma en E es una aplicación $\| \cdot \| : E \to \mathbb{R}$ que satisface:*

1. $\|x\| \geq 0$, $\forall x \in E$

2. $\|x\| = 0$ *si y solo si* $x = 0$

3. $\|\lambda x\| = |\lambda|\|x\|$, $\forall x \in E$ *y* $\lambda \in \mathbb{R}$

4. *Desigualdad triangular* $\|x + y\| \leq \|x\| + \|y\|$ $\forall x, y \in E$.

Se puede comprobar que todo producto escalar $\langle \cdot, \cdot \rangle$, tiene asociada la norma:

$$\|x\| := \sqrt{\langle x, x \rangle} \tag{2.2}$$

A continuación definimos una distancia.

**Definición 2.3** *Dado un conjunto $M$, una distancia es una aplicación $d : M \times M \to \mathbb{R}$ que satisface:*

1. $d(x, y) \geq 0$, $\forall x, y \in M$ *y* $d(x, y) = 0$ *si y solo si* $x = y$

2. $d(x, y) = d(y, x)$, $\forall x, y \in M$

3. *Desigualdad triangular* $d(x, y) \leq d(x, z) + d(z, y)$ $\forall x, y, z \in M$

Se llama *Espacio métrico* al par $(M, d)$. Es bien sabido que toda norma es una distancia. Por lo tanto, podemos definir la distancia:

$$d(x, y) = \|x - y\|, \forall x, y \in \mathcal{H} \tag{2.3}$$

Para poder hablar de completitud será necesario recordar la definición de sucesión de Cauchy:

**Definición 2.4** *Se dice que una sucesión $(x_n)$ en un espacio métrico $(M, d)$ es una sucesión de Cauchy si para todo $\epsilon > 0$ existe $n_0 \in \mathbb{N}$ tal que $d(x_n, x_m) < \epsilon$ para todo n,m $\geq n_0$.*

En otras palabras, a partir de cierto término de la sucesión de Cauchy, todos los términos están tan cerca como se quiera. Coloquialmente se puede decir que los términos se "apelotonan".

**Definición 2.5** *Un espacio métrico se dice completo si toda sucesión de Cauchy es convergente*

Ya estamos en condiciones de definir un espacio de Hilbert.

**Definición 2.6** *Un espacio preHilbert es un espacio de Hilbert si es completo con respecto a la métrica asociada a su norma.*

Una propiedad importante para nosotros es la separabilidad.

**Definición 2.7** *Un espacio métrico es separable si contiene un subconjunto denso y numerable.*

Se va a introducir lo que son las bases ortonormales numerables, ya que su uso es de gran utilidad a la hora de resolver problemas de FDA.

**Teorema 2.1** *Existe* $\{\phi_n\}_{n=1}^{\infty} \subset \mathcal{H}$*, espacio de Hilbert, tal que,*

1. $\int_0^1 \phi_n^2(t)dt = 1$, $\forall n$
2. $\int_0^1 \phi_n(t)\phi_m(t)dt = 0$, $\forall n \neq m$
3. $\forall f \in \mathcal{H}$, $\exists\{a_n\}_{n=1}^{\infty}$ *tal que* $f(t) = \sum_{n=1}^{\infty} a_n\phi_n(t)$*, para todo* $t \in [a,b]$ *donde* $a_i = \langle f, \phi_i\rangle$.
   *Se dice que* $\{\phi_n\}$ *es una base ortonormal de* $\mathcal{H}$.

El siguiente teorema presenta una útil caracterización de la separabilidad en Espacios de Hilbert

**Teorema 2.2** *Un espacio de Hilbert es separable si y solamente si admite una base ortonormal numerable.*

Vamos a pasar a centrarnos más en el espacio que se utiliza para este trabajo.

## 2.1. El espacio $L_2[a,b]$

Como hemos indicado, para este estudio, sea $[a,b]$ un intervalo, se va a trabajar con el espacio $L_2[a,b]$ de funciones.
El producto escalar más popular definido en este espacio es: dadas $f, g \in L_2[a,b]$, se toma

$$\langle f, g\rangle = \int_a^b f(t)g(t)dt, \tag{2.4}$$

y, por lo tanto, la norma asociada tiene la expresión:

$$\|f\|_2 = \sqrt{\int_a^b |f(t)|^2 dt} \tag{2.5}$$

Este espacio es separable, por lo tanto, contamos con bases ortonormales que van a ser de gran utilidad. El uso de estas bases funcionales en $L_2[a,b]$ permite transformar un problema con datos funcionales a uno de estadística multivariada finito dimensional, lo que facilita el manejo de datos. Esto es debido a que una forma de reconstruir los datos que se han obtenido para poder tratarles con mayor facilidad es a partir de bases funcionales. Para ello, por ejemplo, dada una base ortonormal $\{\phi_n\}_{n=1}^{\infty} \subset L_2[a,b]$ se puede aplicar el punto 3 de la Proposición 2.1.
Además, se verifica que $\sum_i a_i^2 < \infty$, lo que implica que a partir de un índice $k_f$ en adelante las componentes de dicha $f$ pueden ser despreciadas. De aquí se deducen los siguientes resultados:

**Propisoción 2.1** *Si se cumple que* $\sum_i a_i^2 < \infty$*, entonces*

$$Si\ f \in L_2[a,b] \Rightarrow \|f\|^2 = \sum_i a_i^2$$

**Corolario 2.1** *Si* $\{\phi_n\}$ *es una base ortonormal de* $L_2[a,b]$ *y se cumple que* $\sum_i a_i^2 < \infty$*, entonces*

$$\lim_{n \longrightarrow \infty} \|f - \sum_i^n a_i \phi_i\| = 0$$

A partir de la definición de $L^2[a,b]$ se encuentra un conjunto de funciones que permiten aproximar las curvas del FDA por medio de suavización.

# Capítulo 3

# Problemas de clasificación de datos

Un problema de clasificación supervisada de datos consiste en lo siguiente: Dado $h \geq 2$, se dispone de una población donde los puntos que la integran provienen de $h$ distribuciones de probabilidad. Tenemos un par de variables aleatorias $(Y, X)$ medidas sobre los individuos que componen la población. Aquí, $X$ es una función e $Y \in \{1, \ldots, h\}$ indica a que distribución pertenece el individuo. Se tiene una *training sample*, que es el conjunto de cierto número $N \in \mathbb{N}$ de parejas $(Y_{i,j}, X_{i,j})$ conocidas, donde $i = 1, \ldots, h$, $j = 1, \ldots, n_i$ y $N = \sum_{i=1}^{h} n_i$. Por tanto, cada $n_i$ es el tamaño de la muestra extraída de la distribución $i$. Se tiene que $Y_{i,j} = i$, para todo $j = 1, \ldots, n_i$. Además, se dispone de otro par $(y, x)$ del que solo se conoce el valor de $x$. El objetivo es predecir el valor de $y$. Para solucionar este problema deberemos construir una función que asocie a cada uno de los posibles valores que pueda tomar $x$, un valor concreto de $y$. La llamaremos función discriminante:

$$D : \mathbb{R}^p \to \{1, \ldots, h\} \tag{3.1}$$

Como nuestro objetivo es acertar el valor de $Y$ para los distintos $X$ que observamos, lo que buscamos es maximizar el valor de $\mathbb{P}[Y = D(X)]$.

En este trabajo nos vamos a centrar en el método de los $k$-vecinos más próximos, que es el que vamos a utilizar para nuestro estudio. Esto es debido a que es un algoritmo que tiene numerosas ventajas. Es una técnica de aprendizaje no paramétrico, es decir, no hace suposiciones explícitas sobre la forma funcional de los datos. Se trata de un algoritmo simple, no solo a la hora de ejecutarlo sino también a la hora de interpretarlo (ver [4]).

## 3.1. $k$-vecinos más próximos

Se trata de un algoritmo *lazy learning*, esto quiere decir que no aprende del modelo sino que memoriza los datos de entrenamiento y los usa para la predicción. Vamos a pasar a explicar detalladamente esta técnica para el caso de que $x \in \mathbb{R}^1$, que es extensible al caso $\mathbb{R}^p$ para $p \geq 1$ sin dificultad, pero no al caso infinito dimensional.

### 3.1.1. Motivación

La justificación de este clasificador se basa en los métodos bayesianos. Vamos a empezar tratando este tema. Se va a dar la idea de este razonamiento en el caso en que las distribuciones tienen función de densidad y además son continuas. Como introducción de la teoría de Bayes, tenemos el siguiente teorema:

**Teorema 3.1 (Teorema de Bayes)** *Dado el espacio probabilístico* $(\Omega, \sigma, \mathbb{P})$ *y* $\{A_i\}$, $B \in \sigma$ *con las* $\{A_i\}_{i=1}^n$ *formando una partición de* $\Omega$, *y* $j = 1, \ldots, n$ *se tiene que:*

$$\mathbb{P}[A_j/B] = \frac{\mathbb{P}[B/A_j] \cdot \mathbb{P}[A_j]}{\sum_{i=1}^{n} \mathbb{P}[B/A_i] \cdot \mathbb{P}[A_i]} \tag{3.2}$$

**Demostración**:
Por la definición de esperanza condicionada tenemos que

$$\mathbb{P}[A_j/B] = \frac{\mathbb{P}[A_j \cap B]}{\mathbb{P}[B]}$$

pero como $\mathbb{P}[A_i \cap B] = \mathbb{P}[B/A_j] \cdot \mathbb{P}[A_j]$ entonces tenemos que

$$\mathbb{P}[A_j/B] = \frac{\mathbb{P}[B/A_j] \cdot \mathbb{P}[A_j]}{\mathbb{P}[B]}$$

y por la ley de probabilidad total

$$\mathbb{P}[B] = \sum_{i=1}^{n} \mathbb{P}[B/A_i] \cdot \mathbb{P}[A_i]$$

ya tenemos el resultado.

■

Para la aplicación de estos métodos, es necesario contar con una distribución a priori sobre los posibles valores de los parámetros. Por tanto, supondremos que se verifica que $\pi_i = \mathbb{P}_i[Y = i]$ para $i = 1, \ldots, h$, donde los $\pi_i$ son conocidos y satisfacen que $\pi_i \geq 0$ y que $\sum_{i=1}^{h} \pi_i = 1$. En estas condiciones, el *teorema de*

*Bayes* consiste en adjudicar a $Y$ el valor en que se alcanza el máximo de las probabilidades condicionadas, o a posteriori.

Sea la función discriminante $D : \mathcal{H} \to \{1, ..., k\}$, dado $x \in \mathbb{R}$, un valor de $X$, calculamos $D(X)$. Nuestro objetivo es que $D(X) = Y$. Por tanto, el problema al que nos enfrentamos es ver lo que vale $\mathbb{P}[D(X) = Y]$ para hacer esta probabilidad lo más grande posible. Luego, se tiene que:

$$\mathbb{P}[D(X) = Y] \leq \int_{\mathcal{H}} \max_{y} \mathbb{P}[Y = y/X = x] \cdot \mathbb{P}_X(dx) \tag{3.3}$$

Para entender como llegamos a esta desigualdad, primero vamos a dar varias propiedades respecto a probabilidades conjuntas y condicionadas.

**Propisoción 3.1** *Propiedades*

1. *Función de densidad conjunta:*

$$f_{X,Y}(x, y) = f_{Y/X}(y) \cdot f_X(x)$$

2. *Esperanza conjunta: Sea $H(X, Y)$ una función en $X$ e $Y$, entonces definimos la esperanza de $H(X, Y)$ como*

$$E[H(X, Y)] = \int H(X, Y) \cdot f_{(X,Y)}(x, y)dxdy$$

3. *Esperanza condicionada:*

$$E[Y/X = x)] = \int y \cdot f_Y(y/X = x)dy$$

Veamos ahora como se llega a la desigualdad (3.3). Primero aplicamos la definición de esperanza y continuamos desarrollando.

$$\begin{aligned}\mathbb{P}[D(X) = Y] = E(\mathbb{P}[D(X) = Y]) &= \int \mathbb{P}[D(X) = Y] \cdot f_{X,Y}(x, y)dxdy \\ &= \int \mathbb{P}[D(X) = Y] \cdot f_{Y/X}(y) \cdot f_X(x)dxdy = E(\mathbb{P}[D(X) = Y/X])\end{aligned} \tag{3.4}$$

Para la tercera igualdad que aparece, aplicamos la definición de función de densidad conjunta y en la última igualdad la de esperanza condicionada. Por último, aplicando la definición de esperanza tenemos que:

$$E(\mathbb{P}[D(X) = Y/X]) = \int_{\mathcal{H}} \mathbb{P}[D(X) = Y/X = x] \cdot \mathbb{P}_X(dx) \tag{3.5}$$

Así concluimos y conseguimos la desigualdad (3.3).

Poder maximizarlo nos lleva a definir la función $D$ de Bayes de la siguiente forma:

$$D_B(X) = arg\,\max_y P[Y = y/X = x] \tag{3.6}$$

Por tanto, con esta definición de función discriminante de Bayes, $D_B$, en (3.6), tenemos una regla de clasificación. Aún así, podría pasar que nos encontremos con dificultades a la hora de clasificar algunos elementos.

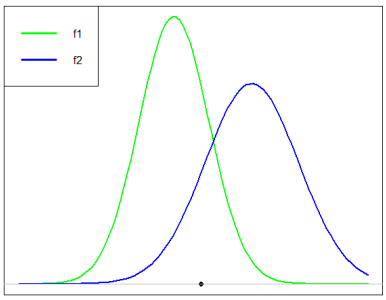


Figura 3.1: Gráfica donde podemos observar dos funciones de densidad, $f_1$ y $f_2$, de dos distribuciones de probabilidad

En la Figura 3.1 tenemos dos funciones de densidad, $f_1$ y $f_2$, asociadas a dos distribuciones de probabilidad, $\mathbb{P}_1$ y $\mathbb{P}_2$ respectivamente. Además, de una de estas dos distribuciones se ha generado un punto y se debe decidir de cual de las dos ha sido, es decir, lo que se quiere es clasificar un dato cuyo valor es la abscisa del punto negro. Decidiremos que una observación dada viene de una distribución si en esa coordenada la curva de dicha distribución tiene un valor mayor que la otra. El problema vendrá cuando debamos clasificar un dato que se encuentre en el punto de corte entre ambas curvas. Supongamos que $f_1$ corre más rápido que $f_2$ en un entorno de un punto de corte entre ambas, y que pretendemos clasificar un punto $x$ posicionado a la izquierda de dicho punto de corte. Deberemos por tanto, seleccionar un intervalo pequeño.

Para solucionar esto, vamos a empezar por calcular la probabilidad de que un punto $x$ caiga en un intervalo cualquiera $(a, b)$. Supondremos que tenemos $h = 2$ distribuciones donde clasificar los datos para ajustarnos a la gráfica de la Figura 3.1 y simplificar la justificación, pero el seguimiento es fácilmente extensible para $h > 2$.

Sea $x$ el elemento a clasificar que es conocido, en el intervalo $(a, b)$ pequeño, y sean $f_1$ y $f_2$ las funciones de densidad continuas de las distribuciones $\mathbb{P}_1$ y $\mathbb{P}_2$ respectivamente. Entendiendo que $\mathbb{P}[x \in \mathbb{P}_i]$ quiere decir que dicho punto $x$ se

ha obtenido mediante la probabilidad $\mathbb{P}_i$, tenemos lo siguiente:

$$\begin{aligned}\mathbb{P}[x \in (a,b)] &= \mathbb{P}[x \in \mathbb{P}_1] \cdot \mathbb{P}[x \in (a,b)/x \in \mathbb{P}_1] + \mathbb{P}[x \in \mathbb{P}_2] \cdot \mathbb{P}[x \in (a,b)/x \in \mathbb{P}_2] \\ &= \pi_1 \int_a^b f_1(x)dx + \pi_2 \int_a^b f_2(x)dx \\ &\simeq \pi_1 f_1(x)(b-a) + \pi_2 f_2(x)(b-a)\end{aligned} \tag{3.7}$$

Definamos $\pi_i$ de otra forma que nos conviene más para este caso: $\pi_i = \mathbb{P}[x \in \mathbb{P}_i]$, es decir, es la probabilidad total de puntos que viene de $\mathbb{P}_i$. Por tanto, lo que estamos haciendo para calcular dicha probabilidad es ver la probabilidad de puntos que vienen de $\mathbb{P}_1$ y $\mathbb{P}_2$. La segunda igualdad se cumple por hipótesis, ya que las distribuciones tienen funciones de densidad. Para la última igualdad, como ambas funciones son continuas, en particular en el intervalo pequeño $(a,b)$, podemos aplicar el *Teorema del Valor Medio* y así tenemos que la integral se puede aproximar al valor de la función en cualquier punto por el intervalo en el que está definida (ver [1]). Tomamos el punto $x$ porque es el que nos conviene. Por Bayes, el punto se clasificará en la distribución donde la proporción de puntos sea mayor. Por tanto, lo que nos va a decir en que grupo debemos clasificar el punto $x$ será ver que cantidad $\pi_i f_1(x)$ es mayor. Para ello, tenemos el siguiente teorema:

**Teorema 3.2** *Llamemos a $\pi_i$ probabilidad a priori y a $\mathbb{P}[Y = i/X = x]$ probabilidad a posteriori. Sea $f_i(t) = \mathbb{P}[X = x/Y = i]$ la función de probabilidad condicional de $X$ para una observación $x$ que pertenece a la distribución $\mathbb{P}_i$. Entonces el Teorema de Bayes (3.1) establece que:*

$$P[Y = i_0/X = x] = \frac{\pi_{i_0} f_{i_0}(x)}{\sum_{i=1}^{h} \pi_i f_i(x)}$$

*(Ver [2])*

Del Teorema 3.2 se deduce que

$$D_B(X) = i_0 \Leftrightarrow \pi_{i_0} f_{i_0}(x) \geq \pi_i f_i(x),\ \forall i$$

Si tomamos el intervalo $(a,b)$ donde se encuentra $x$ pequeño, se tiene que la proporción de observaciones que esperamos de la población $i$ es:

$$\pi_i \int_a^b f_i(x)dx$$

Lo que nos interesa es ver cuantas observaciones esperamos que vengan de $i_0$ en ese intervalo. Para simplificar, suponemos que $i_0$ es $D_B(x)$. De esta forma, tenemos que, sea $i_0$ único, $\pi_{i_0} f_{i_0}(x) > \pi_i f_i(x)\ \forall i \neq i_0$,

$$\pi_{i_0} \int_a^b f_{i_0}(x)dx > \pi_i \int_a^b f_i(x)dx,\ \forall i \neq i_0$$

Por estos resultados se deduce que el *teorema de Bayes* consiste en asignar a $x$ la distribución $\mathbb{P}_i$ de forma que $P[Y = i/X = x]$ sea máxima.

En conclusión, el Teorema 3.3 da conocimientos suficientes para que el $k$-NN coincida asintóticamente con el clasificador de Bayes. Por lo tanto, el $k$-NN es, asintóticamente, un clasificador óptimo en este contexto. Nótese que aquí estamos suponiendo que $k$ varía con el tamaño muestral.

**Teorema 3.3** *Sea $N$ el tamaño muestral, se toma $k_N$, es decir $k$ dependiendo del tamaño muestral, de forma que*

$$\frac{k_N}{N} \longrightarrow 0 \quad con \quad k_N \longrightarrow \infty \tag{3.8}$$

Este teorema está tomado de [13].

El inconveniente al que nos enfrentamos ahora es que este argumento solo nos vale para $R^p$, es decir, el clasificador es óptimo en $R^p$. Esto es debido que en espacios de Hilbert de dimensión infinita no hay funciones de densidad. Por tanto, lo que nos atañe ahora es extender la regla de Bayes a dimensión infinita.

Para empezar, vamos a hablar de la maldición de la dimensión. Abarca los distintos fenómenos que surgen al analizar y organizar datos de espacios de altas dimensiones que no ocurren en el espacio físico, descrito normalmente con solo tres dimensiones. La causa de estos problemas es que al aumentar la dimensionalidad, el volumen del espacio aumenta exponencialmente provocando que los datos se vuelvan dispersos. En otras palabras, la maldición de la dimensión se basa en que a medida que aumenta la dimensionalidad de los datos, necesitamos más muestras de datos para asegurar la precisión de un modelo.

Estos fenómenos obstaculizan la búsqueda de los vecinos más cercanos en un espacio de alta dimensión. No solo por la necesidad de ampliar la muestra de datos para conseguir un algoritmo eficiente, sino que también afectan a la hora de tomar la distancia. No es posible rechazar rápidamente a los candidatos utilizando la diferencia en una coordenada para una distancia basada en todas las dimensiones. La solución para este problema es la reducción de dimensionalidad. Este proceso se basa en identificar las dimensiones que no son relevantes para el análisis del estudio, que llamamos "ruido", y agruparlas para crear nuevas eliminando las originales. De esta forma, habrá que transformar los datos para que estén representados por las nuevas dimensiones. (Ver [14])

Esto nos llega para dimensiones muy altas pero no se extiende a dimensión infinita. Esto es debido a que el volumen de la bola de radio fijo converge a 0 al aumentar la dimensión del espacio. Afecta considerablemente al método $k$-NN, ya que esta técnica se basa en la necesidad de tener puntos cercanos en la muestra para poder hacer estimaciones precisas, lo que depende de la convergencia de estas bolas. El volumen de una esfera $d$-dimensional de radio R

es:

$$V_d = \frac{\pi^{d/2}}{\Gamma(1+\frac{d}{2})} R^d \approx \pi^{d/2} R^d \frac{1}{\sqrt{\pi d}} \left(\frac{2e}{d}\right)^{d/2} = \frac{1}{\sqrt{\pi d}} \left(\frac{2e\pi R^2}{d}\right)^{d/2} = \frac{C^d}{\sqrt{\pi d} d^{d/2}} \longrightarrow 0 \tag{3.9}$$

donde $\Gamma$ es la función gamma que se define como: Dado $n \in \mathbb{N}$, $\Gamma(n) = (n-1)!$. Para la aproximación se ha aplicado la fórmula de Stirling.

Supongamos que, para cada $n \in \mathbb{N}$, se tiene una probabilidad $d$-dimensional, $\mathbb{P}_d$, con función de densidad, $f_d$, uniformemente acotada por $C > 0$. Si $x_0 \in \mathbb{R}^d$ y $q > 0$, se tiene que:

$$\mathbb{P}_d\{x : \|x - x_0\| \leq q\} = \int_{\{x:\|x-x_0\|\leq q\}} f_d(t)dt \leq CVol(B_d(0,q)) \longrightarrow 0 \tag{3.10}$$

donde $B_d(0,q)$ es la bola $d$-dimensional centrada en el origen de radio $q$. Con lo cual, si $d$ es alto, dado un punto cualquiera, es difícil encontrar un punto producido por la distribución $\mathbb{P}_d$ cerca de él.

Por ello, es lógico pensar que en dimensión infinita, es decir en espacios funcionales, el algoritmo $k$-NN puede ser muy problemático. Sin embargo, esto no es así. Esto puede ser debido a que los datos funcionales usuales no tienen una dispersión muy grande y dada una curva, es fácil que haya otras muy parecidas a ella. Se podría entender que las curvas en realidad no pertenecen a un espacio infinito, sino a uno de dimensión moderada. (Ver [6])

### 3.1.2. El método en la práctica

Un problema clave en el $k$-NN es la elección del $k$ apropiado. Este punto se analiza en el apartado 3.1.3. Por tanto, aprovechamos que tenemos fijado un $k$ adecuado (que, de acuerdo con el Teorema 3.3, debería ser pequeño).

Recordemos que disponemos de una base de datos de tamaño $N$ procedentes de $h$ poblaciones, esto es, la *training sample* $(Y_{i,j}, X_{i,j})$. Pretendemos adivinar $y$ en una nueva observación $(y, x)$ de la que solo conocemos $x$. Primero tomamos los $k$ puntos de la muestra cuya $X_{i,j}$ sea más cercana a $x$. A estos $k$ puntos les llamamos los $k$ *vecinos más próximos*. Comprobamos los valores de $i$ para cada una de ellas. Cada vez que aparezca el valor $i$, el caso $\{y = i\}$ "recibe un voto". Finalmente, estimamos $y$ con el valor entre $i = 1, \ldots, h$ que más votos haya recibido. En caso de empate se clasificará al azar.

**Nota 3.1** *La clasificación que estamos haciendo es por mayoría simple. Hay estudios en la que esta clasificación no es la adecuada. Es decir, hay ocasiones en las que fallar al clasificar un dato es más razonable si se clasifica en el grupo $i$ que en el grupo $j$. Por ejemplo, si debemos clasificar unos datos que provienen de una central nuclear donde se debe decidir si va a explotar o no, es preferible*

*que ante al duda se elija el caso en que explota que al revés. En este caso se debería definir desde el principio otra forma de "apuntarse" a uno de los casos que se ajuste más a dicho estudio.*

Vamos a pasar a detallarlo analíticamente. Dado $i = 1, \ldots, h$ y $j = 1, \ldots, n_i$ denotaremos

$$d_{i,j}(x) = \|X_{i,j} - x\| \tag{3.11}$$

Ordenamos estos valores de menor a mayor y les nombramos $d_{(1)} \leq d_{(2)} \leq \cdots \leq d_{(N)}$. En caso de empate, se ordenaría al azar entre los que tengan el mismo valor. Solo nos interesan los $k$ primeros, que determinan el conjunto de los $k$-NN

$$\mathcal{N}_k(x) = \{(i,j) : d_{i,j} = d_{(t)}, t = 1, \ldots, k\} \tag{3.12}$$

A continuación contamos la cantidad de elementos que hay de cada grupo entre los $k$-NN. Para cada $i \in \{1, ..., h\}$, vamos a llamar

$$V_i(x) = \#\{i : (i,j) \in \mathcal{N}_k(x)\}. \tag{3.13}$$

Así, $D_k(x) = i_0$ donde $i_0 = \arg\max_i V_i(x)$.

### 3.1.3. ¿Cómo fijar $k$?

El único punto pendiente a la hora de aplicar el algoritmo es como debemos tomar $k$ para que sea lo más eficaz posible. El problema es que si se toma un valor grande se corre el riesgo de hacer la clasificación favoreciendo a la población mayoritaria sin atender a la similitud entre puntos y, si el valor es necesariamente pequeño, puede haber inexactitud en la clasificación debido a los pocos datos seleccionados para la comparación.

En la práctica, lo más habitual es usar el método de *cross-validation*, CV, o validación cruzada en español. Es una técnica empleada para la evaluación de métodos y la elección de parámetros en múltiples métodos estadísticos, incluyendo el aprendizaje automático, como es el caso. Para ello, lo primero que se hace es dividir la muestra en dos conjuntos complementarios. Luego, consiste en realizar el análisis de uno de ellos (*training sample*) y validar el análisis en el otro conjunto (*test sample*).

Existen muchas variaciones del CV, de ahora en adelante lo llamaremos así por sus siglas en inglés. Vamos a explicar como funciona el CV mediante el *Leave-one-out cross-validation.*

Comencemos por fijar el conjunto $\mathcal{C}$ de los valores candidatos para $k$. Para ello, fijamos el máximo, $C$, que obviamente cumple $C \leq N$ y, luego fijamos los candidatos. Una elección razonable puede ser $\{1, 2, 3, 4, 5, 7, 10, 25, 35, \ldots, C\}$. Es decir, tomar valores mas seguidos para $k$'s pequeños, debido a que puede haber mucha diferencia en las clasificaciones si añadimos un voto más cuando

$k = 1$, pero no tanta si $k = 7$. Por ello, a medida que $k$ es más grande hay menos diferencia entre $k$ y $k + 1$. Es recomendable utilizar solo números impares para evitar empates cuando se trata de clasificar datos para el caso $h = 2$.

Tomamos la *training sample* $(Y_1, X_1), \ldots, (Y_N, X_N)$ y eliminamos de ella el elemento $(Y_1, X_1)$, en este caso la *test sample* estará compuesta de un solo dato en cada iteración. Ahora aplicamos $k$-NN para clasificar el elemento que acabamos de sacar, tomando $k$ igual a cada uno de los valores de $\mathcal{C}$.
Para cada valor de $k$ habremos obtenido un fallo o un acierto, dependiendo de si hemos clasificado bien o no el punto. Vamos a tomar una variable que guarde estos datos:

$$A_1(k) = \begin{cases} 1 \text{ si los } k\text{-NN determinan que } D(X_1) = Y_1 \\ 0 \text{ si los } k\text{-NN determinan que } D(X_1) \neq Y_1 \end{cases} \tag{3.14}$$

Repetimos el proceso metiendo en la muestra $(Y_1, X_1)$ y sacando el valor $(Y_2, X_2)$, con lo que obtendríamos los valores $A_2(k)$, con $k \in \mathcal{C}$ y así sucesivamente. Haremos esto con todos los puntos y elegiremos el menor de los valores de $k$ que más aciertos haya obtenido. Esto es, tomamos como $k$ el menor valor que cumpla:

$$\sum_{i=1}^{N} A_i(k) = \max_{r \in \mathcal{C}} \sum_{i=1}^{N} A_i(r) \tag{3.15}$$

Una vez explicado en que consiste esta técnica con el *Leave-one-out cv*, vamos a mencionar brevemente algunas de las variantes más populares:

- *Leave-p-out cross-validation*: Este proceso es igual que el anterior con la salvedad de que en este caso se tiene como *test sample* un subconjunto de la muestra de $p$ puntos en vez de uno solo. Esto quiere decir que se deben tomar todos los subconjuntos posibles de $p$ puntos de la muestra de datos. Por ello, aplicar esta variante puede hacer la tarea muy larga y tediosa, aunque siempre se puede tomar una elección aleatoria de los posibles subconjuntos. En todo caso es más popular el *Leave-one-out*.

- *n-fold cross-validation*: Se suele utilizar cuando el tamaño muestral es grande y/o el procedimiento de cálculo es especialmente costoso. Se divide la muestra, al azar, en $n$ submuestras diferentes del mismo tamaño y se hace el proceso de CV dejando fuera cada vez uno de estos grupos. Los números más comunes para este procedimiento suelen ser el 4, 5, 10...

- *Stratified cross-validation*: Aplicando los métodos anteriores, puede suceder que los subconjuntos de la muestra tomados para el análisis no representan adecuadamente a la muestra total debido a que les tomamos de forma aleatoria. Una manera de evitar esto es usando esta técnica. La estratificación es un proceso de reorganización de los datos utilizando variables auxiliares como la edad, estadio de desarrollo de la enfermedad,... dependiendo del estudio, para garantizar que cada subconjunto sea un buen representante de la muestra total.

Una variación relevante de la técnica $k$-NN es el *weighted* $k$-NN, que es el que se utilizará para nuestro análisis. En este caso, no se elige un $k$ concreto, sino que se tiene en cuenta un número finito de ellos al mismo tiempo. Se basa en la suposición de que los puntos más cercanos al que se desea clasificar deberían de tener más importancia que los más lejanos a la hora de decidir a qué grupo pertenece el dato. Por tanto, se da un mayor peso a la clase de pertenencia de los puntos más próximos que a la de los más lejanos (ver [5]).

# Capítulo 4

# Análisis estadístico de datos funcionales

Vamos a comenzar describiendo unos ejemplos para mostrar la importancia que los problemas con datos funcionales tienen en el día a día.

**Ejemplo 1.** *Consumo eléctrico:* El precio de la luz se determina en subasta pública y depende fundamentalmente de la relación entre oferta y demanda. Tanto los proveedores como ciertos consumidores están interesados en estimar este precio con antelación. Para ello, habrá que estimar valores como nubosidad, viento,... (que afectan a la producción fotovoltaica, eólica,...) y temperatura, producción industrial... (que afectan al consumo). Una posibilidad para realizar esta estimación es utilizar las series a tiempo continuo de las variables que influyen en el precio.

**Ejemplo 2.** *Meteorología:* Es recurrente mirar la página del tiempo en Internet para planificar tu día, tanto para decidir que plan hacer como para saber como vestirte. El pronóstico del tiempo es una ciencia que predice el estado de la atmósfera para un período futuro y un lugar dado. Esto se lleva a cabo recolectando la mayor cantidad posible de datos acerca del estado de la atmósfera, como la temperatura, presión atmorférica, vientos, humedad, precipitaciones... y haciendo el estudios de estos datos recogidos para los días anteriores a la predicción.

**Ejemplo 3.** *Crecimiento:* Llevar a cabo una buena documentación del crecimiento humano es esencial para poder controlar las condiciones normales de crecimiento. Esto ayuda a poder detectar lo antes posible una anomalía en el proceso de crecimiento. Este tipo de datos son muy complicados de recoger debido a que los niños deben ser llevados a un laboratorio en ciertas fechas asignadas previamente. Por tanto, contamos para realizar esta estimación con medidas tomadas en tiempo discreto, que ni si quiera están equiespaciadas. La Figura 4.1

nos muestra una representación de esto, donde se puede ver claramente la separación entre cada dato (ver [16]).

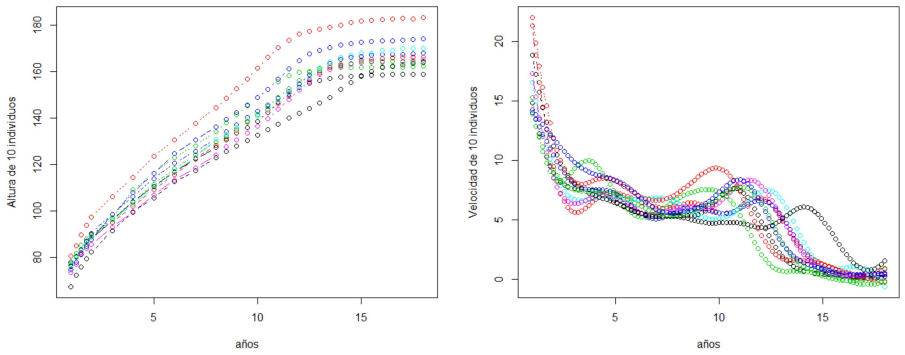


Figura 4.1: Datos del crecimiento de la altura de 10 de los niños de la muestra del experimento junto con estimaciones de la velocidad

Un dato funcional se puede identificar con una función $X(s)$, $s \in S$, donde $S$ corresponde a un intervalo continuo de tiempo, aunque realmente todos se miden de modo discreto, como hemos visto en el Ejemplo 3.
Independientemente de como se consiga, lo importante es que cada dato funcional se entiende como una función o curva. De esta manera, podría decirse que el análisis de datos funcionales estudia problemas estadísticos cuyos datos son estas curvas o funciones.

En la práctica, pueden pasar dos cosas.
Cuando se trata de medidas tomadas en experimentos de laboratorio, es frecuente la obtención de datos medidos en los mismos puntos. Un ejemplo de esto es el estudio que vamos a realizar con los datos de proteómica. En estos casos, no es necesario manipular dichos datos para obtener funciones.
En cambio, lo normal es que no se observe como tal la forma funcional del dato $X(s)$, sino su valor para ciertos puntos en el intervalo $S$. Es más, podría suceder que ni siquiera tengamos los datos de diferentes individuos observados en los mismos puntos del intervalo $S$. En el caso del Ejemplo 3, estamos observando la altura de ciertos niños cada año. Lo más probable, es que no se tome el dato en el mismo momento para todos los niños, es decir, puede que para uno se tome justo a los 3 años y en cambio a otro se le coja unos días después por razones ajenas al experimento. Otro ejemplo es la toma de los datos de la electricidad que consume una vivienda. En este caso, podría pasar que se produzca un fallo que haga que se salten algunas observaciones.
Cuando todos los datos no tengan el mismo número de mediciones o no estén realizados en los mismos instantes, no es razonable considerarlos como elemen-

tos de un espacio euclídeo $n$-dimensional. Parece más razonable considerarlos como funciones de variables continuas que han sido observadas únicamente en un conjunto discreto de puntos. La primera tarea es tratar de reconstruir la función de la que provienen las mediciones disponibles.

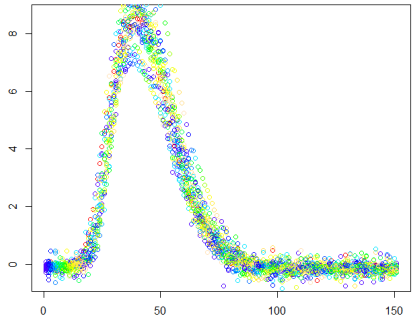

Figura 4.2: Ejemplo de mediciones de fuerza de pellizco

En la Figura 4.2 tenemos un ejemplo de la representación de unos datos funcionales. Consiste en 151 mediciones de la fuerza de un pellizco en 20 repeticiones. Para llevar a cabo este experimento, se ha medido la fuerza ejercida sobre un contador durante un breve pellizco por el pulgar y el índice. Se puede observar una separación entre puntos. Esto es debido a la toma de los datos espaciados en diferentes momentos en el tiempo. (Ver [16])

Pasemos a describir el problema matemáticamente: Sea $X$ la función a determinar partiendo de $m$ observaciones en $S$. Se dispone de los valores $X(s_1), ..., X(s_m)$ para algunos $s_1, ..., s_m \in S$ que suponemos conocidos. Por lo tanto, el primer paso para realizar un análisis de datos funcionales consiste en reconstruir la función $X(S)$ a partir de dichas observaciones puntuales. Un proceso muy común es el que se basa en la estructura de espacio de Hilbert de $L_2(S)$ mediante bases funcionales. Y es aquí donde nos interesan tanto las buenas propiedades de los espacios de Hilbert y más en concreto las propiedades del espacio $L_2$.

## 4.1. Preparación de los datos para su posterior análisis: Ejemplo

En esta sección vamos a utilizar el Ejemplo 3 para explicar como trabajar con datos tomados en tiempo discreto y como reducir la dimensión de estos. Los datos que se van a mostrar, llamados "*Berkeley Growth Data*", están sacados del paquete de la librería de R "fda.usc" (ver [16]).

Vamos a centrarnos en el suavizado de funciones, que es una de las formas de transformar los datos para conseguir un mejor manejo de los mismos. Consiste

en la conversión de los datos en funciones suaves que den buenas aproximaciones. Por tanto, necesitamos un conjunto de funciones base que podamos sumar y de esta forma conseguir una buena aproximación de nuestros datos. Es decir, sean $\{\phi_i\}_{i=1}^{\infty}$ el conjunto de las funciones base, se quiere obtener la función $f(t)$, que será la combinación lineal de estas. Tendremos entonces que:

$$f(t) = a_1\phi_1(t) + a_2\phi_2(t) + \cdots = \sum_{i=1}^{\infty} a_i\phi_i(t) \tag{4.1}$$

Si tomamos límites:

$$\lim_{k\longrightarrow\infty} f(t) = \lim_{k\longrightarrow\infty} \sum_{i=1}^{k} a_i\phi_i(t),\ \forall t \tag{4.2}$$

Lo que implica que:

$$\forall t \quad \exists k_t / f(t) \approx \sum_{i=1}^{k} a_i\phi_i(t),\ \forall k \geq k_t \tag{4.3}$$

Es decir, para cada $t$ tendríamos que tomar un $k_t$ distinto que sería a partir del cual se podría dar una buena aproximación de dicho $f(t)$. Además, podemos aplicar el corolario 2.1

$$\lim_{k\longrightarrow\infty} \|f(t) - \sum_{i=1}^{k} a_i\phi_i(t)\|_2 = 0 \tag{4.4}$$

Finalmente se tiene que $\exists k_0$ que no depende de ningún otro valor de forma que:

$$\|f(t) - \sum_{i=1}^{k} a_i\phi_i(t)\|_2 \approx 0,\ \forall k \geq k_0 \tag{4.5}$$

En conclusión, de la ecuación (4.5) se concluye que $\forall t$ se puede conseguir una aproximación lo suficientemente buena como para que no haya diferencia entre ambas en términos de normas, esto es:

$$f(t) \approx \sum_{i=1}^{k} a_i\phi_i(t),\ \forall k \geq k_0 \tag{4.6}$$

Existen numerosos conjuntos de funciones base que se pueden tomar para construir $f(t)$. En un primer momento se piensa en polinomios, pero estos no son muy flexibles. Por ello, en lugar de polinomios, se utilizan distintos sistemas de bases, Kernel, series de Fourier, Wavelet,..., pero en este apartado, y con objetivo de tratar con los datos de crecimiento, vamos a hablar del sistema llamado *B-spline*[1].

[1] B viene de Bases

**Definición 4.1** *Dados k valores reales $t_i$, llamados nodos, con $t_1 \leq t_2 \leq \cdots \leq t_k$. La i-ésima función base B-spline $b_{i,j}(t)$ de orden k está definida por las relaciones recursivas (ver [11]):*

$$b_{i,1}(t) = \begin{cases} 1 \text{ si } t_i \leq t \leq t_i + 1 \\ 0 \text{ en otro caso} \end{cases}, \; i = 1, ..., k \tag{4.7}$$

*y*

$$b_{i,j}(t) = \frac{t - t_i}{t_{i+j-1} - t_i} b_{i,j-1}(t) + \frac{t_{i+j} - t}{t_{i+j} - t_{i+1}} b_{i+1,j-1}(t), \; j > 1 \tag{4.8}$$

A continuación vamos a describir el procedimiento que se lleva a cabo tras la toma de datos para poder hacer un buen análisis de estos. Se van a utilizar para este caso, las funciones *B-splines* (ver [9]).

En este experimento se observaron las diferentes alturas durante el crecimiento de 54 niñas y 39 niños, es decir, un total de 93 individuos. Para cada uno, se toman 31 observaciones a lo largo de 18 años. Estos datos no están igualmente espaciados ya que las medidas no se tomaron en exactamente los mismos momentos. Se deben manipular los datos para que así puedan ser analizados correctamente.

Una vez recogidos los datos, se deben convertir en datos funcionales. En este caso, se requiere de dos pasos para ello: transformar los datos en funciones y calcular la mejor combinación lineal para el conjunto de medidas discretas de la altura de cada individuo. Para este análisis, convertiremos los datos en funciones mediante el uso de bases. Usamos el sistema de bases *B-Spline* porque más adelante querremos observar la velocidad y la aceleración de la altura. Para ello, será necesario controlar la suavidad de nuestras funciones base. A pesar de que otros sistemas también permiten esto, concretamente este es muy flexible y por ello, será más sencillo trabajar con él. Los datos de una sola niña van desde una altura de 75 centímetros más o menos al año de edad hasta unos 165 centímetros en su plena estatura. Mientras que en el caso de los niños puede alcanzar alrededor de los 180 centímetros.

Una vez elegido el sistema de funciones que vamos a utilizar, se debe definir el conjunto de las funciones base. Para ello, hay que especificar dos características. En primer lugar, debido a que las *B-splines* son segmentos polinómicos unidos entre sí, debemos describir el grado de los segmentos polinómicos, teniendo en cuenta que $orden = grado + 1$ . Se tomará un orden al menos cuatro veces mayor que el orden más alto de la derivada que se tendrá que manejar. En este caso necesitaremos calcular la aceleración, es decir, la segunda derivada, así que vamos con el orden 6, lo que significa que las *B-splines* son polinomios de quinto grado a trozos.
La segunda característica es especificar los nodos, es decir, el conjunto de $t_i$ para $i = 1, \ldots, k$. Para simplificar, se utilizarán las propias observación como nodos. Hay 31 observaciones por individuo, por lo que un buen $k_0$ a tomar para este

ejemplo, siguiendo la notación introducida en (4.5), sería $k_0 = 31$.
Con estos dos pasos estamos en condiciones de dibujar las funciones que mejor se ajustan a los datos de las niñas. En la primera gráfica de la Figura 4.1 podemos observar la representación de estas funciones para la altura de 10 niños, donde cada punto ilustra las medidas tomadas.

# Capítulo 5

# Proteómica en la la clasificación de tumores

En este capítulo vamos a clasificar ciertos tumores de mama como benignos o malignos mediante el análisis estadístico de unas muestras de dichos tumores.

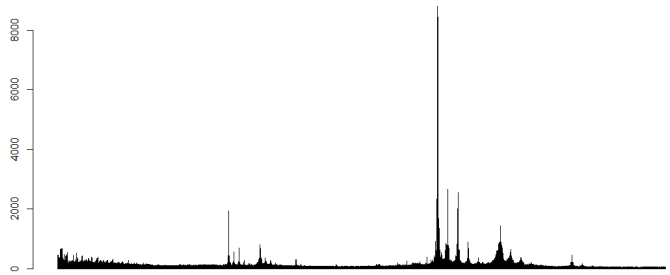


Figura 5.1: Espectrograma de un tumor

El procedimiento es el siguiente. Se toma una muestra del tejido a analizar. Se divide hasta el nivel de proteína y, entonces, un experto llamado espectrógrafo mide el número de proteínas que contiene la muestra para cada relación entre su masa y carga eléctrica. Se llama espectrograma a la curva $S : \mathbb{R}^+ \to \mathbb{N}$ que asocia a cada $t \in \mathbb{R}^+$ el número de proteínas del tejido con relación masa/carga igual a $t$. Podemos pensar que son la realización de una muestra de una v.a. definida en el espacio probabilístico $(\Omega, \sigma, \mathbb{P})$ con llegada en un espacio formado por funciones reales definidas en el intervalo $[699.99, 12000]$. Por tanto, los espectrogramas pertenecen al espacio de Hilbert $L_2[699.99, 12000]$. Esto es debido a que son funciones medibles por ser combinaciones lineales finitas de indicadores de conjuntos de la $\sigma$-álgebra de Borel, $\mathcal{B}$, en concreto de intervalos, que pertenecen a $\mathcal{B}$. En la Figura 5.1 podemos observar un ejemplo de una de estas curvas. Estos gráficos son 373.401-dimensionales. Es decir, el aparato de medida

detecta $d = 373.401$ posibles relaciones masa/carga diferentes.

Se utiliza una base de datos con 216 tumore, donde se ha determinado, mediante la técnica apropiada con total seguridad, que 121 son malignos, mientras que el resto son benignos. Utilizando el método de los $k$-vecinos más próximos, se procederá de la siguiente forma.
Primero dividiremos la base de datos en dos grupos aleatoriamente, uno de los grupos con 150 datos que se utilizará de *training sample* y el otro, la *test sample*, con los 66 restantes que será el que analicemos utilizando la información del primero, de esta forma podremos comprobar la eficacia de este método de clasificación.
Para poder clasificar los nuevos datos se deberán hacer algunas modificaciones. Como ya comentamos, no es necesario manipular los datos para poder reconstruir las funciones, ya que los datos vienen de un experimento de laboratorio y por tanto no hay diferencias en los puntos de toma de las medidas. Es decir, como al final los datos se utilizan para los cálculos, porque reemplazamos las integrales por sumas finitas, no tiene sentido convertir los datos que no tienen problemas de cálculo en funciones. Sin embargo, si que debemos normalizarlos, ya que todas las muestras no van a ser iguales: Al tomar una muestra a cada mujer, no se obtiene el mismo numero de moléculas. También reduciremos la dimensión de los datos (mediante agrupaciones) para hacerlos más manejables. A continuación, describiré los diferentes pasos que se han realizado para poder hacer la mejor clasificación posible.

Este análisis se lleva a cabo mediante la herramienta RStudio (ver [16]). En el Apéndice A aparecen todos los programas que se han ido utilizando para la realización de este estudio.

## 5.1. Preparación de la muestra

Como contamos con observaciones de dimensiones muy altas, más de 350000 para cada espectrograma, vamos a empezar reduciendo la dimensión de estos. Nuestro objetivo va a ser reducir la dimensión a 2000, de esta forma los datos son mucho más manejables. Para ello, se lleva a cabo el siguiente procedimiento: Los datos pertenecen al intervalo $[699.99, 12000]$, por tanto, lo que haremos será dividir el intervalo en 2000 trozos. De esta forma, se agruparán las dimensiones sumando, para cada nuevo intervalo, las moléculas de los valores que lo componen.

Una vez hemos reducido la dimensión de estos datos, el siguiente paso debe ser normalizarlos. Para ello, calcularemos la proporción de moléculas para cada relación masa/carga. Es decir, se dividirán las moléculas para cada relación entre la suma total de moléculas de cada espectrograma. Tras la normalización de los datos, los espectrogramas están definidos en el intervalo $[0, 0.06017885]$

Con estas modificaciones ya estamos en condiciones de aplicar a los datos el algoritmo $k$-NN. Pasaremos ahora a describir las características relevantes del método para su empleo en este estudio.

## 5.2. Elección de la métrica adecuada

La mejor forma de comparar dos curvas es midiendo la distancia entre ellas mediante una métrica. La más usada por su estructura es la $L_2$ en el intervalo en que estén definidas las funciones.

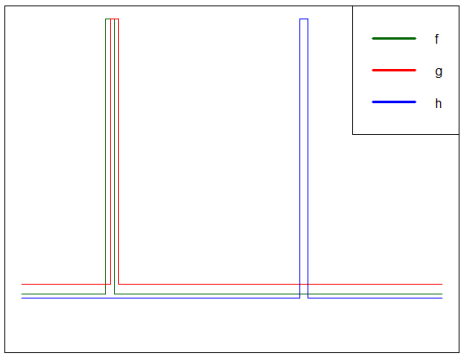


Figura 5.2: Gráfica donde observamos tres espectrogramas simplificados para medir sus distancias

Esta distancia no nos vale para nuestro caso ya que si tomamos la distancia entre dos de nuestras curvas siempre nos va a quedar muy parecida y cerca de cero. Me explico: si nos fijamos en la Figura 5.1 podemos entender que nuestras curvas son muy próximas a cero en casi todo punto, menos en los picos de la curva, es decir, los valores más altos de la función. La información relevante se va a encontrar en los picos y por tanto, todo lo demás es ruido. Este ruido podemos atribuirle, por ejemplo, a que en la muestra habrá moléculas que todos tenemos y que no tengan que ver con el cáncer. Por tanto para nuestro estudio son irrelevantes. Estos picos no representan un único punto sino un intervalo con una base muy estrecha. Por tanto, al calcular la distancia de $L_2$, la diferencia entre cualesquiera dos curvas, aunque tuvieran picos muy próximos, se distinguiría por las diferencias en las zonas bajas de la curva, que entendemos como ruido. Esto es un problema, ya que necesitamos que se pueda medir la distancia entre dos de nuestras funciones.
Se entiende que los picos de tejidos similares deberían estar próximos, pero no necesariamente en exactamente el mismo punto. Supongamos entonces que tenemos nuestra función a clasificar $f$ y tomamos $g$ y $h$ funciones de modo que

un pico en $g$ está muy próximo a otro de $f$ mientras que el pico en $h$ está muy alejado, como en la Figura 5.2. Queremos calcular la distancia de $f$ a $g$ y de $f$ a $h$, es decir, $d(f,g)$ y $d(f,h)$. Con esta norma, tendremos que $d(f,g) \approx d(f,h)$ y no se tiene en cuenta donde se encuentra el pico de la curva. Es más, podría pasar incluso que $d(f,g) < (f,h)$ si el ruido de la función $f$ se parece más al de la función $h$ que al de la $g$. Sin embargo, es obvio que $d(f,g)$ debería ser mucho menor que $d(f,h)$. Por tanto, no nos sirve esta distancia, debemos buscar otra distancia que se ajuste más a nuestro problema.

Antes de nada, vamos a proponer un método para deshacernos de este ruido. Para ello, lo primero que vamos a hacer es ordenar las medidas que hemos obtenido. Partimos de los datos $x_1, \ldots, x_d$ que han sido observados en los puntos $s_1, \ldots, s_d$. Es decir, mediante el espectrograma se ha obtenido $X = (x_1, \ldots, x_d)$, siendo $d$ la dimensión de los datos, es decir, el número de datos tomados de cada tumor.
Primero ordeno estas medidas mediante una permutación: $x_{i_1} \leqslant \cdots \leqslant x_{i_d}$.
A continuación, se elige $p \in [0,1]$ como la proporción de valores pequeños (proporción de ruido) que voy a eliminar. Como hay más ruido que información, muchos de los valores no van a servir de nada, por lo tanto $p$ será alto. Ahora definimos:

$$x^*_{i_j} := \begin{cases} 0 & \text{si} \quad j \leq pd \\ x_{i_j} & \text{si} \quad j > pd \end{cases} \tag{5.1}$$

Si por ejemplo tomo $p = 0.8$, esto quiere decir que hago cero 80 % de los valores, más concretamente, llevo a cero los primeros $[0,8 \cdot d]$ valores donde $[\cdot]$ denota la parte entera del número que corresponda.
Entonces sustituimos $X$ por $X^* = (x^*_1, \ldots, x^*_d)$.
Además, dada la función $f$, vamos a aproximar $\int f(s)dt$ por $\sum_{i=1} f(s_i) \cdot \delta$ donde $\delta$ representa la distancia entre puntos de la observación. Esto se puede hacer porque que las funciones con las que trabajamos están definidas en un conjunto discreto. Como $\delta$ es constante no se va a tener en cuenta ya que solo hace el papel de un cambio de unidades. De hecho, en este problema no hay aproximación ninguna porque las funciones a considerar con constantes en intervalos. Entonces, se define:

$$\|X - X'\| = \sqrt{\sum_{i=1}^{d} |X^*(s_i) - X'^*(s_i)|^2} \tag{5.2}$$

Pero para usar los $k$-NN podemos reemplazar las distancias por sus transformaciones usando funciones estrictamente crecientes. Por ello, para simplificar los cálculos, vamos a trabajar con el cuadrado de la fórmula (5.2).

Esta distancia podría funcionar, pero debemos hacer algunos cambios para que dos espectrogramas con máximos parecidos en puntos cercanos disten poco. Es decir, nuestros problemas vienen cuando dos picos, como les hemos

nombrado previamente, están próximos pero no coinciden en el mismo punto. Esta distancia no nos asegura que la diferencia entre dos espectrogramas con máximos muy próximos no vaya a ser grande. Entonces, necesito una métrica que tenga en cuenta, no solo la diferencia entre picos sino la proximidad ente las localizaciones de los picos. Para ello vamos a utilizar una distancia tipo Levy-Prohonov. (Ver [6])

**Definición 5.1** *Dadas dos funciones $f, g$ definidas en un intervalo $I$, la distancia de Levy-Prohonov se define como:*

$$\text{ínf}\{q > 0 : |f(t) - g(s)| \leq q, \forall t, s \in I \text{ tal que } |t - s| \leq q\}. \tag{5.3}$$

Sean $X$ e $Y$ dos espectrogramas, el procedimiento a seguir es el siguiente: Fijando $q > 0$ tomo $u$ un punto cualquiera de $S$, entonces pretendo buscar $v$ próximo a $u$ de modo que $X(u)$ e $Y(v)$ sean lo más parecidos posible. Entonces medimos la diferencia con la distancia del estilo de (5.3) pero con unas modificaciones para ajustarlo a nuestro caso:

$$\inf_{v:|v-u|\leq q} |X(u) - Y(v)|. \tag{5.4}$$

Por ejemplo, si el punto $u$ que hemos tomado es un máximo de $X$ y que a su vez, es mayor que el máximo de $Y$, entonces, utilizando esta norma, $Y(v)$ será el máximo de $Y$ en $[v - q, v + q]$ para $q$ pequeño.

Definamos ahora a partir de (5.4):

$$d_q^*(X, Y) := \sum_{i=1}^{d} \inf_{v:|v-t_i|\leq q} |X(t_i) - Y(v)|^2 \tag{5.5}$$

**Nota 5.1** *No tomamos raíces cuadradas, por lo que no sale una verdadera distancia, pero esto es irrelevante para nuestro caso.*

Obsérvese que, además, $d_q^*(X, Y)$ no es simétrica. Lo que estamos haciendo es fijar puntos de $X$ y buscar de entre los puntos de $Y$ el que mejor se acomoda. Pero, por otro lado, si se hace al revés, fijar un punto de $Y$ y buscar el más próximo a este entre los puntos de $X$, no tienen por qué salir los mismos emparejamientos. Vamos a verlo con un ejemplo:

En la Figura 5.3 vemos que si fijamos el punto $X(u)$ en $X$ y buscamos el punto más próximo en $Y$, este punto será $Y(v)$. En cambio, si tomamos el punto $Y(v)$ en $Y$, el punto más cercano a este en $X$ será el punto $X(v)$, puesto que $X(v) = Y(v)$. Con esto se visualiza que, efectivamente, no es simétrica.
Debemos hacer algo para arreglarlo. Para ello, lo que haremos será tomar el mínimo entre ambas funciones:

$$d_q(X, Y) := \text{ínf}(d_q^*(X, Y),\, d_q^*(Y, X)) \tag{5.6}$$

Y así es como resolvemos el problema de la asimetría y por tanto, para nuestro análisis utilizaremos la fórmula (5.6).

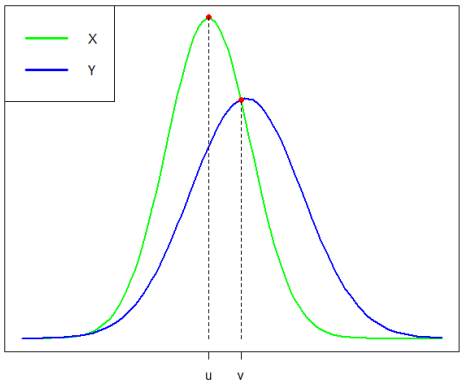

Figura 5.3: Gráfica que muestra la asimetría de (5.4)

## 5.3. Uso de la técnica $k$-vecinos más próximos

Ahora pasamos a analizar los datos, y como ya hemos adelantado, utilizaremos el método de los $k$-NN para $h = 2$, ya que contamos con dos poblaciones, la de los tumores benignos y la de los malignos.

Recordemos en que consiste este método. Tenemos una muestra de datos que están correctamente clasificados y que llamaremos base de datos. Esta técnica busca para cada dato a clasificar, los $k$ más cercanos de la base de datos, mira sus clases de pertenencia y clasifica el dato como la mayoría de estos valores, teniendo en cuenta, posiblemente, ciertos puntos que afectan a la proximidad.

En nuestro caso, tenemos unos espectrogramas obtenidos con técnicas proteómicas y una base de datos donde sé que tumor es benigno y cual es maligno. Ahora la tarea que nos atañe es elegir con que $k$, $p$ y $q$ vamos a trabajar.

Empecemos por elegir el $k$. La forma más común es hacerlo mediante la *cross-validation* ya mencionada anteriormente. En este caso vamos a hacerlo de otra forma, vamos a aplicar el algoritmo *weighted* $k$-NN. Entonces el método va a seguir los siguientes pasos:
Analizaremos los datos mediante $k$-NN para $k = 1, 3, 5, 7, 9$. Tomando estos $k$'s, estamos dando peso 5 al punto más próximo, peso 4 a los dos siguientes, y así sucesivamente. Hemos prescindido de los pares para evitar empates, ya que estamos trabajando con dos grupos, $h = 2$, y hemos tomado de 1 a 9, que además la suma de todos ellos, 25, también es impar, y es lo que realmente vamos a utilizar a la hora de decidir a que grupo pertenece cada muestra. Lo que vamos a hacer para clasificar el espectrograma $X_{i,j}$ es aplicarle la técnica de los $k$-vecinos más próximos para cada uno de los $k$'s que estamos considerando. Si continuamos con la notación utilizada en la sección 3.1, predeciremos que

un tumor es benigno dando el valor 0 a $D(Y_{i,j})$ y maligno dando el valor 1. De esta forma si sumamos todos los valores resultantes de cada análisis decidiremos que es benigno si la suma es menor que 13 y que tiene cáncer de lo contrario. Adjudicamos la igualdad al caso en que el tumor es maligno ya que puestos a confundirnos, es mejor pensar que se tiene cáncer cuando no es así que lo contrario. Sea

$$V_{i,k}(x) = V_i(X_{i,j}) \text{ para cada } k \in 1, 3, 5, 7, 9 \tag{5.7}$$

entonces tenemos que:

$$D(X_{i,j}) = \begin{cases} 0 \text{ si } \sum_k V_{1,k}(X_{i,j}) < 13 \\ 1 \text{ en otro caso} \end{cases} \tag{5.8}$$

Obviamente de esta forma, estamos dando mayor importancia a los puntos que están más próximos a $X_{i,j}$. Esto se debe a que cada punto que tomamos al aplicar el método para $k$, también le estaremos tomando cuando lo hagamos para $k+1$. Lo que parece razonable, ya que se espera que cuanto más cercanos estén dos elementos, más fácil será que sigan la misma distribución.

Ahora que ya hemos elegido los $k$ que utilizaremos, vamos a escoger el $p$ y $q$ por *leave-one-out CV* aplicado a la *training sample*. Probaremos para los valores $p \in \{0, 0.5, 0.8, 0.9\}$ y $q \in \{0, 10, 25, 35\}$. La siguiente tabla muestra la cantidad de datos acertados para cada combinación entre los posibles valores de ambos parámetros.

| Métrica | Aciertos |
|---|---|
| D0N0 | 114 |
| D10N0 | 110 |
| D25N0 | 119 |
| D35N0 | 111 |
| D0N0.5 | 117 |
| D10N0.5 | 114 |
| D25N0.5 | 120 |
| D35N0.5 | 113 |
| D0N0.8 | 120 |
| D10N0.8 | 115 |
| D25N0.8 | 118 |
| D35N0.8 | 111 |
| D0N0.9 | 118 |
| D10N0.9 | 118 |
| D25N0.9 | 122 |
| D35N0.9 | 115 |
| | |

En la Tabla 5.3 , llamaremos D$q$N$p$ a la métrica obtenida cuando se recorta una proporción $p$ de puntos y se utiliza una ventana de amplitud $q$ para el

cálculo de las distancias.

En vista de los aciertos de cada combinación, se decide tomar los siguientes parámetros, $q = 25$ y $p = 0.9$, es decir, eliminamos el 90 % de los valores, que son considerados como ruido. Estos son los valores que han producido los mejores resultado con una proporción de 122/150 aciertos, esto es, un 81.33 %.

## 5.4. Resultados

Tras preparar los datos para que sean más manejables para su estudio, y se hayan elegido los parámetros necesarios para utilizar el algoritmo, se procede a realizar el análisis de los espectrogramas mediante el método de los $k$-NN. Se va a analizar una *test sample* de 66 espectrogramas.

Una vez analizados los datos se obtienen 51 aciertos. Esto quiere decir que se clasifican bien 51 datos de los 66 que analizamos, es decir, hay una proporción de 77.27 % de éxito. Se puede observar que la efectividad ha bajado con respecto a la proporción de aciertos que teníamos, 81.33 %. Esto se debe a que estamos analizando la *test sample*, que contiene muchos menos datos que la *training sample*, que es la que se había usado previamente.

En conclusión, la proporción de aciertos no parece muy alta, sería más razonable llegar al 90 % de aciertos, como en el estudio realizado por Cuesta-Albertos y Fraiman (ver [7]). En este caso, obtenemos peores resultados porque hemos reducido la dimensión de los datos más que en dicho estudio, perdiendo así mucha información. En el trabajo de Cuesta-Albertos y Fraiman se reducía la dimensión a 15000, mientras que en nuestro caso se ha reducido a 2000.
Por tanto, se puede concluir que, para un buen análisis de estos espectrogramas mediante la técnica de los $k$-NN, sería más razonable trabajar con una dimensión $d$ al menos mayor o igual que 15000. De esta forma, los resultados que se obtienen son considerablemente mejores.

# Bibliografía


[1] Alba Valverde Colmeiro (2020) *Tema 6 Integral definida de Riemann*, Análisis matemático, Economía, Universidad Autónoma de Madrid.

[2] Amat J. R. (Septiembre, 2016) *Análisis discriminante lineal (LDA) y análisis discriminante cuadrático (QDA)*

[3] Asociación Española contra el cáncer https://www.aecc.es/es/todo-sobre-cancer/prevencion/deteccion-precoz [Internet; Visitado 25/08/2021]

[4] Bafandeh S. y Bolandraftar M. (2013) *Application of K-Nearest Neighbor (KNN) Approach for Predicting Economic Events: Theoretical Background*, S B Imandoust et al. Int. Journal of Engineering Research and Applications

[5] Bicego M. y Loog M. (2016) *Weighted K-Nearest Neighbor Revisited*, 23rd International Conference on Pattern Recognition (ICPR)

[6] Cuesta-Albertos J.A. (2021) *Análisis de Datos Funcionales*, Universidad de Cantabria

[7] Cuesta-Albertos J.A. y Fraiman R. (2006) Análisis de espectrogramas, *Comunicación personal*

[8] D. Juan Bayo Calero (2015) *Estudio de casos y controles de marcadores sanguíneos para el diagnóstico precoz de cáncer de mama* , Universidad de Huelva.

[9] Funtional Data Analysis https://www.psych.mcgill.ca/misc/fda/ex-growth-a1.html [Internet; Visitado 05/08/2021]

[10] Instituto Nacional del Cáncer https://www.cancer.gov/espanol/cancer/naturaleza/que-es [Internet; Visitado 25/06/2021]

[11] Ipanaqué R. *B-splines* http://www.unp.edu.pe/pers/ripanaque/bsplinesanaliti/bs.html [Internet; Visitado 05/08/2021]

[12] Organización Mundial de la Salud https://www.who.int/es/news-room/fact-sheets/detail/breast-cancer [Internet; Visitado 25/06/2021]

[13] Quezada M. (2017) *K-vecinos más próximos en una aplicación de clasificación y predicción en el Poder Judicial del Perú*, Universidad Nacional Mayor de San Marcos.

[14] Rahman F.n (2020) *k-Nearest Neighbors and the Curse of Dimensionality* https://towardsdatascience.com/k-nearest-neighbors-and-the-curse-of-dimensionality-7d64634015d9 [Internet; Visitado 05/09/2021]

[15] Ramsay J.O. y Silverman B.W. (1997) *Funcional Data Analysis*, Springer

[16] Ramsay J. O., Graves S. and Hooker G. (2020). fda: Functional Data Analysis. R package version 5.1.9. https://CRAN.R-project.org/package=fda

# Apéndice A

# Algoritmos para el análisis

Aquí se encuentran todos los códigos de cada algortimo que se ha utilizado para realizar el análisis del Capítulo 5.

1.

```
#  ------------------------------------------------------------------------
#                    Código para la reducción de la dimensión
#  ------------------------------------------------------------------------
DIVIS=seq(from=699.99, to=12000, length.out=2001)
N1=121 #malignos
N2=95 #benignos
N=N1+N2
#DATOS: matriz donde vamos a guardar los datos
DATOS=matrix(rep(NA, N*2000), ncol=N)
#cargamos, reducimos la dimension y guardamos los espectogramas en DATOS
for(j in 1:216){
  aux=paste0("daf- (",j,").txt")
  aux=read.table(aux, quote="\"")
  for (i in 2:2000){
  DATOS[i,j]=sum(aux[aux[,1] <= DIVIS[i] & aux[,1] > DIVIS[i-1],2])
  }
  DATOS[1,j]=sum(aux[aux[,1] <= DIVIS[1],2])
}
```

2.

```
#  ------------------------------------------------------------------------
#                        Código para normalizar los datos
#  ------------------------------------------------------------------------
N1=121 #malignos
N2=95 #benignos
N=N1+N2
#DATOSNORM: matriz donde se van a guardar los datos normalizados
DATOSNORM=matrix(rep(NA, N*2000), ncol=N)
for(j in 1:216){
  sj=sum(DATOS[,j])
  DATOSNORM[,j]=DATOS[,j]/sj
}
```

3.

```
#  ------------------------------------------------------------------------
#                    Código k-NN para determinar parametros
#  ------------------------------------------------------------------------

#Tomamos al azar los datos de la muestra para la training sample
set.seed(1)
nTrain=150 #cantidad de espectogramas en la traininf sample
ind = sample(x=1:216, nTrain)
trainData = DATOSNORM[,ind]

#Tenemos cada dato con su coordenada benigno o maligno (analizados)
N1=121 #malignos
N2=95 #benignos
N=N1+N2 #total
estados = rep(0,N)
estados[96:N] = 1

#Etiquetamos la training
trainData_labels = estados[ind]

#posibles valores de p que vamos a analizar
pvalores = c(0,0.5,0.8,0.9)
#posibles valores de q que vamos a analizar
qvalores = c(0,10,25,35)
#clasificacion: matriz donde vamos a guardar como clasificamos la training
clasificacion =
matrix(nrow=nTrain,rep(0,nTrain),ncol=length(pvalores)*length(qvalores))
#comprobacion: matriz donde guardamos si hemos acertado o no en la
clasificacion
comprobacion =
matrix(nrow=nTrain,rep(0,nTrain),ncol=length(pvalores)*length(qvalores))

#creamos la distancia que vamos a usar
distancia.prov <- function(a,b,q){
```

```
  sumando1 = rep(NA,2000)
  for(i in 1:2000){
    aux = 1:2000
    aux = aux[abs(DIVIS-DIVIS[i])<=q]
    diferencia = rep(NA,length(aux))
    for(j in 1:length(aux)){
      diferencia[j] = abs(trainData[i,a]-trainData[aux[j],b])
    }
    sumando1[i] = min(diferencia,na.rm = TRUE)^2
  }
  return(sum(sumando1^2))
}

distancia <- function(a,b,q){
  min(distancia.prov(a,b,q),distancia.prov(b,a,q))
}

contp=1
contq=0
for(p in pvalores){
#Recortamos el ruido con el p%, porcentaje de datos que determinamos como
#ruido y por tanto eliminamos
  if(p>0){
    for(i in 1:nTrain){
      aux = order(trainData[,i])
      trainData[aux[1:(p*2000)],i] = 0
    }
  }

for(q in qvalores){

#Creamos la matriz para las distancias 2 a 2 entre la training con ella
#misma
DIST = matrix(rep(0, nTrain*nTrain), ncol=nTrain)
```

```
#Calculamos las distancias 2 a 2 y las guardamos en la matriz DIST
for(a in 1:(nTrain-1)){
  for(b in (a+1):nTrain){
    DIST[a,b] = distancia(a, b, q)
    DIST[b,a] = DIST[a,b]
  }
  DIST[a,a] = sum(DIST[a,])
}
if(contq == 0){
  DIST0=DIST
}

#Buscamos los k vecinos más próximos para cada dato de la train
for(i in 1:nTrain){
#indicesCercanos: indices donde se encuentran los 9 vecinos cercanos para
#cada dato de la training
  indicesCercanos = order(DIST[i,])[1:9]
#Comprobamos el grupo al que pertenecen los k-vecinos para cada dato de
#la train
  grupoi = trainData_labels[indicesCercanos]

 #Vamos a ver como clasificariamos cada dato
 #para cada dato de train nos dice si es benigno o maligno
  aux = c(5,4,4,3,3,2,2,1,1)*grupoi
  #k-NN pesado, por tanto nos importa el valor de la suma de todos
  if(sum(aux)>=13){
    clasificacion[i,contp+contq] = 1 #tumor maligno
  }

  #Comparamos los resultados obtenidos con el programa k-NN y los valores
#reales
  if(clasificacion[i,contp+contq]==trainData_labels[i]){
    comprobacion[i,contp+contq] = 1
  }
}
```

```
  contq=contq+1
  print(paste("q=",q))
} #for q
  contp=contp+1
  if(p==0){
    contq=3
  }else if(p==0.5){
    contq=6
  }else if(p==0.8){
    contq=9
  }

} #for p

#comprobamos cual es la mejor combinacion de parametros calculando los
#aciertos obtenidos
aciertos = rep(NA, (length(pvalores)*length(qvalores)))
for(i in 1:(length(pvalores)*length(qvalores))){
  aciertos[i] = sum(comprobacion[,i])
}
```

4.

```
#  ------------------------------------------------------------------------
#                    Código k-NN para analizar la test
#  ------------------------------------------------------------------------

#Separamos la muestra en la training sample y la test sample
set.seed(1)
nTrain=150
nTest = 66
ind = sample(x=1:216, nTrain)
trainData = DATOSNORM[,ind]
testData = DATOSNORM[,-ind]
nTrain=length(trainData[1,])
nTest = length(testData[1,])

#Tenemos cada dato con su coordenada benigno o maligno (analizados)
N1=121 #malignos
N2=95 #benignos
N=N1+N2 #total
espectogramas = rep(0,N)
espectogramas[96:N] = 1

#Etiquetamos la muestra de entrenamiento y de test
trainData_labels = espectogramas[ind]
testData_labels = espectogramas[-ind]

#definimos la distancia que vamos a usar
distancia.1 <- function(a,b,q){
  sumando1 = rep(NA,2000)
  for(i in 1:2000){
    aux = 1:2000
    aux = aux[abs(DIVIS-DIVIS[i])<=q]
    diferencia = rep(NA,length(aux))
    for(j in 1:length(aux)){
```

```
      diferencia[j] = abs(testData[i,a]-trainData[aux[j],b])
    }
    sumando1[i] = min(diferencia,na.rm = TRUE)^2
  }
  return(sum(sumando1,na.rm = TRUE))
}
distancia.2 <- function(a,b,q){
  sumando2 = rep(NA,2000)
  for(i in 1:2000){
    aux = 1:2000
    aux = aux[abs(DIVIS-DIVIS[i])<=q]
    diferencia = rep(NA,length(aux))
    for(j in 1:length(aux)){
      diferencia[j] = abs(trainData[i,b]-testData[aux[j],a])
    }
    sumando2[i] = min(diferencia,na.rm = TRUE)^2
  }
  return(sum(sumando2,na.rm = TRUE))
}

distancia <- function(a,b,q){
  min(distancia.1(a,b,q),distancia.2(a,b,q))
}

#seleccionamos los valores de p y q
p=0.9
q=25
#clasificacion: matriz donde vamos a guardar como clasificamos la test
clasificaciontest = rep(0,nTest)
#comprobacion: matriz donde guardamos si hemos acertado o no en la
clasificacion
comprobaciontest = rep(0,nTest)

#Recortamos el ruido con el p%, porcentaje de datos que determinamos como
#ruido y por tanto eliminamos
```

```
if(p>0){
  for(i in 1:nTrain){
    aux = order(trainData[,i])
    trainData[aux[1:(p*1000)],i] = 0
  }
  for(i in 1:nTest){
    aux = order(testData[,i])
    trainData[aux[1:(p*1000)],i] = 0
  }
}

#Creamos la matriz para las distancias 2 a 2 entre train y test
DIST = matrix(rep(0, nTrain*nTest), ncol=nTrain)

#Calculamos las distancias 2 a 2 y las guardamos en la matriz DIST
for(a in 1:(nTest-1)){
  for(b in 1:nTrain){
    DIST[a,b] = distancia(a, b, q)
  }
}
#Buscamos los k vecinos más próximos para cada dato de la test
for(i in 1:nTest){
  #indicesCercanos: indices donde se encuentran los 9 vecinos cercanos
#para cada dato de la test
  indicesCercanos = order(DIST[i,])[1:9]
  #Comprobamos el grupo al que pertenecen los k-vecinos para cada dato de
#la test
  grupoi = trainData_labels[indicesCercanos]


#Vamos a ver como clasificariamos cada dato
#para cada dato de train nos dice si es benigno o maligno
  aux = c(5,4,4,3,3,2,2,1,1)*grupoi
  # k-NN pesado, por tanto no importa el valor de la suma de todos
  if(sum(aux)>=13){
```

```
    clasificaciontest[i] = 1 #tumor maligno
  }


#Comparamos los resultados obtenidos con el programa k-NN y los valores
#reales
  if(clasificaciontest[i]==testData_labels[i]){
    comprobaciontest[i] = 1
  }
}


#comprobamos los aciertos que hemos obtenido en el analisis de la test
aciertostest = sum(comprobaciontest)
```