



UNIVERSIDAD DE GRANADA

Doble Grado en Ingeniería Informática y Matemáticas

*Distribución de puntos en la esfera.
Competición en Kaggle: TalkingData AdTracking
Fraud Detection Challenge.*

Daniel López García

Curso 2017/18

UNIVERSIDAD DE GRANADA

Doble Grado en Ingeniería Informática y Matemáticas

***Distribución de puntos en la esfera.
Competición en Kaggle: TalkingData AdTracking
Fraud Detection Challenge.***

Trabajo Fin de Grado presentado por
Daniel López García

Curso 2017/18

Miguel A. Piñar González
Antonio Bautista Bailón Morillas

A mis padres, siempre, por su ejemplo, su cariño y por no permitirme abandonar.

A mi familia y amigos.

Finalmente, mi más sincero agradecimiento a mis tutores, Miguel Piñar y Antonio Bailón por su ayuda durante este proyecto.

Yo, Daniel López García, con DNI 74742492J, alumno de la titulación *Doble Grado en Ingeniería Informática y Matemáticas* de la *Universidad de Granada*, asumo la originalidad de este trabajo, entendida en el sentido de que no se han utilizado fuentes sin citarlas debidamente

Para que conste, firmo la presente declaración.

A handwritten signature in black ink, appearing to be 'Daniel López García', written in a cursive style.

Granada, a 19 de junio de 2018

Índice

I	Distribución de puntos en la esfera.	5
1.	Armónicos Esféricos	7
1.1.	Preliminares	7
1.1.1.	Notación	7
1.2.	Armónicos Esféricos y Espacios Primitivos.	8
1.2.1.	Espacios de Polinomios Homogéneos.	9
1.2.2.	Armónicos de Legendre y Polinomios de Legendre	10
1.2.3.	Armónicos Esféricos	11
1.3.	Teorema de Adición. Consecuencias.	12
1.4.	El Operador de Proyección	17
1.5.	Generando Bases Ortonormales.	19
2.	Cálculo del Gradiente	23
2.1.	El gradiente de los armónicos esféricos.	23
2.1.1.	Caso particular $d=3$	27
2.2.	Puntos críticos del gradiente.	27
3.	Integración Numérica	33
3.1.	Fórmulas de una variable.	33
3.2.	Métodos de Gauss de Orden Superior.	37
3.3.	Integración puntos dispersos	38
3.4.	Integración sobre el disco unidad.	38
II	Competición en Kaggle: TalkingData AdTracking Fraud Detection Challenge.	41
4.	Introducción	43
4.1.	Descripción del problema.	43
4.2.	Herramientas utilizadas.	44
4.3.	Proceso a seguir.	45
5.	Estudio de los datos.	47

6. Preprocesamiento.	53
6.1. Proceso llevado a cabo.	53
7. Soluciones planteadas	57
7.1. Boosting	57
7.1.1. Optimización de parámetros	59
7.2. RUSBoosting.	60
7.3. CUSBoosting	61
7.4. Resultados obtenidos.	61
8. Conclusiones.	63
A. La Función Gamma	65
B. Resultados básicos de la esfera.	67
C. Polinomios de Legendre	69
C.1. Fórmulas de Representación	69
C.1.1. Fórmula de Rodrigues	69
C.1.2. Fórmulas de Representación Integral.	69
C.2. Propiedades	70
D. Polinomios de Gegenbauer	71
E. Funciones de Legendre Asociadas	73
F. El problema de clasificación	75
G. Visualización de los puntos críticos.	77
Bibliografía	81

Índice de figuras

2.2.1.Puntos críticos del gradiente para $n=20$ y $k=9$	30
2.2.2.Puntos críticos del gradiente para $n=25$ y $k=20$	30
2.2.3.Puntos críticos del gradiente para $n=30$ y $k=5$	31
5.0.1.Distribución de <i>ip</i> y <i>app</i>	47
5.0.2.Distribución de <i>channel</i> y <i>device</i>	48
5.0.3.Distribución de <i>os</i>	48
5.0.4.Distribución de <i>day</i> y <i>month</i>	49
5.0.5.Distribución de <i>year</i> y <i>timestamp</i>	49
5.0.6.Balanceo de clases	50
5.0.7.Matriz de correlación.	51
F.0.1Ejemplo validación cruzada.	75

Índice de cuadros

5.1. Valores perdidos.	50
6.1. Pruebas realizadas durante el preprocesamiento.	55
7.1. Resultados obtenidos con los diferentes algoritmos.	61

Introducción

Summary

In this work, we consider systems of points on the unit sphere related to problems of approximation and numerical integration. These systems are useful in various fields ranging from global climate models for the Earth, mapping of the stationary and magnetic fields of the Earth, crystallography, geodesic domes, virus modeling, computational geometry, etc.

Starting with the construction of the spherical harmonic space. Next, we study the Addition Theorem and its consequences. This result will allow us to generate an orthonormal basis of the space of spherical harmonics. Later, we get an expression for critical points of the gradient of a spherical harmonic. In particular, we get critical points of the 3-dimensional sphere.

Finally, we get some results about numeric quadrature.

In the second part of the work, we explain the process followed for participation in a data mining competition. This process consists of the parts related to the understanding of the problem, the visualization of the data with which we will work, the application of preprocessing techniques to improve the data set and the choice of the algorithm to use to build a data model. Finally, we discuss the results obtained.

keywords: sphere, spherical harmonic, gradient, preprocessing, machine learning.

Resumen

Uno de los propósitos de este trabajo es determinar conjuntos de puntos para aproximación, interpolación e integración sobre la esfera y sus propiedades geométricas. Las distribuciones de puntos en la esfera unidad tienen aplicaciones que van desde modelos climáticos globales para la Tierra, mapeado de los campos estacionales y magnéticos de la Tierra, cristalografía, cúpulas geodésicas, modelado de virus, geometría computacional, etc.

Para ello, en primer lugar, introduciremos de forma constructiva el espacio de los armónicos esféricos sobre el que trabajaremos. Una vez construido dicho espacio, trataremos el Teorema de adición y sus consecuencias. Estos ingredientes nos permitirán generar una base ortonormal del espacio de esféricos armónicos.

Una vez asentadas las bases, estudiaremos el gradiente de dichos polinomios y calcularemos sus puntos críticos. Como caso particular, estudiaremos los puntos críticos de dichos polinomios sobre la esfera de dimensión tres y obtendremos una visualización de los resultados obtenidos.

Finalmente, trataremos varios resultados sobre integración numérica.

En la segunda parte del trabajo, explicaremos el proceso seguido durante la participación en una competición de data mining. Dicho proceso consta de las partes relativas a la comprensión del problema, la visualización de los datos con los que vamos a trabajar, la aplicación de técnicas de preprocesamiento para mejorar el conjunto de datos y la elección del algoritmo a usar para construir un modelo de datos. Finalmente, estudiaremos los resultados obtenidos y realizaremos un análisis sobre los mismos.

Palabras clave: esfera, armónicos, gradiente, preprocesamiento, clasificación.

Objetivos.

Los objetivos propuestos al inicio del trabajo fueron los siguientes:

- Simulación y visualización de distribuciones de puntos sobre la esfera
- Estimación numérica de la calidad de las aproximaciones obtenidas mediante estas distribuciones de puntos.
- Enfrentarse a un problema real de machine learning, estudiando las posibles soluciones y los resultados obtenidos.

El primer objetivo se ha logrado satisfactoriamente, siendo desarrollado en el Capítulo 2. El segundo objetivo se ha alcanzado durante el desarrollo del Capítulo 3. Finalmente, el tercer objetivo se ha cumplido, siendo desarrollado en la segunda parte del trabajo.

Organización de la memoria.

El Capítulo 1 contiene todo lo referente a la definición y construcción del espacio de polinomios armónicos esféricos. En el Capítulo 2 se calculará el gradiente de dichos polinomios y se estudiarán sus puntos críticos. Se estudiará el caso particular de dimensión 3, obteniendo una visualización de los puntos obtenidos sobre la esfera. Finalmente, en el capítulo 3 se estudiarán diferentes resultados de integración numérica.

En cuanto a la segunda parte del trabajo, en el Capítulo 4 se realiza una introducción al problema a tratar. En los Capítulos 5 y 6 se describe el proceso de visualización y posterior preprocesamiento del conjunto de datos. En el Capítulo 7 se describen los algoritmos usados para resolver el problema. Finalmente, en el Capítulo 8 se resumen los resultados obtenidos y se describen las conclusiones obtenidas.

Para la redacción de los Capítulos 1 y 3 se ha tomado como referencia [1], mientras que en el Capítulo 2 se ha tomado como referencia [3]. Para la segunda parte del trabajo se han tomado como referencia los libros [4] y [11]. Además, para desarrollar el Capítulo 7 se ha tomado como referencia [5], [6] y [7].

Parte I

Distribución de puntos en la esfera.

Capítulo 1

Armónicos Esféricos

1.1. Preliminares

1.1.1. Notación

Para empezar fijaremos la notación que seguiremos durante el capítulo. Usaremos $d \in \mathbb{N}$ para representar la dimensión de un conjunto; en particular, el conjunto $\mathbb{R}^d = \{x = (x_1, \dots, x_d)^T : x_j \in \mathbb{R}, 1 \leq j \leq d\}$ es el espacio euclídeo de dimensión d con el producto escalar y la norma definidos por

$$(x, y) = \sum_{j=1}^d x_j y_j \quad |x| = (x, x)^{1/2} \quad x, y \in \mathbb{R}^d$$

En \mathbb{R}^d usaremos la base canónica

$$e_1 = (1, 0, \dots, 0)^T, \dots, e_d = (0, 0, \dots, 1)^T$$

y escribiremos $x = \sum_{j=1}^d x_j e_j, x \in \mathbb{R}^d$.

Para indicar la dimensión explícitamente usaremos $x_{(d)}$ en lugar de x . En tal caso, $x_{(d)} = x_{(d-1)} + x_d e_d$ siendo $x_{(d-1)} = (x_1, \dots, x_{d-1}, 0)^T$. También usaremos $x_{(d-1)}$ para referirnos al vector $(d-1)$ -dimensional $(x_1, \dots, x_{d-1})^T$.

Trabajaremos sobre la esfera unidad $\mathbb{S}^{d-1} = \{\xi \in \mathbb{R}^d : |\xi| = 1\}$. Por simplicidad, llamaremos esfera a \mathbb{S}^{d-1} .

Definición 1.1. Sean $\xi, \eta \in \mathbb{S}^{d-1}$, definimos las siguientes distancias:

- La distancia euclídea $|\xi - \eta| = \sqrt{2(1 - \xi \cdot \eta)}$
- La distancia geodésica $\theta(\xi, \eta) = \arccos(\xi \cdot \eta)$

Nota 1.2. Usando que $\frac{2}{\pi} \leq \sin t \leq t, t \in [0, \pi/2]$ se deduce la siguiente relación entre ambas distancias:

$$\frac{2}{\pi} \theta(\xi, \eta) \leq |\xi - \eta| \leq \theta(\xi, \eta)$$

esto es, ambas distancias son equivalentes.

Para $x = (x_1, \dots, x_d)$ definimos $x^\alpha = x_1^{\alpha_1} \dots x_d^{\alpha_d}$. Análogamente, para el operador gradiente $\nabla = (\partial_{x_1}, \dots, \partial_{x_d})^T$ definimos

$$\nabla^\alpha = \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \dots \partial x_d^{\alpha_d}}$$

Y finalmente definimos el operador laplaciano como

$$\Delta = \nabla \cdot \nabla = \sum_{j=1}^d \left(\frac{\partial}{\partial x_j} \right)^2.$$

1.2. Armónicos Esféricos y Espacios Primitivos.

Consideramos \mathbb{O}^d el conjunto de matrices ortogonales de orden d . Para cualquier $\eta \in \mathbb{O}^d$ vector no nulo, $\mathbb{O}^d(\eta) = \{A \in \mathbb{O}^d : A\eta = \eta\}$ es el subconjunto de matrices ortogonales que deja el subespacio $\text{span}\{\eta\} = \{\alpha\eta : \alpha \in \mathbb{R}\}$ invariante.

Definición 1.3. Sea $f : \mathbb{R}^d \rightarrow \mathbb{C}$ y $A \in \mathbb{R}^{d \times d}$, se define f_A como:

$$f_A(x) = f(Ax), \forall x \in \mathbb{R}^d$$

Consideremos un subespacio \mathcal{V} de funciones definidas de \mathbb{R}^d en un subconjunto de \mathbb{R}^d .

Definición 1.4. Sea \mathcal{V} un subespacio de funciones definidas de \mathbb{R}^d en $K \subseteq \mathbb{R}^d$. Se dice que \mathcal{V} es invariante si para $f \in \mathcal{V}$ y $A \in \mathbb{O}^d$, entonces $f_A \in \mathcal{V}$. Considerando \mathcal{V} un subespacio invariante de un espacio proveniente de un producto escalar se define:

- \mathcal{V} es reducible si $\mathcal{V} = \mathcal{V}_1 + \mathcal{V}_2$ con $\mathcal{V}_1 \neq \emptyset$, $\mathcal{V}_2 \neq \emptyset$ verificando $\mathcal{V}_1, \mathcal{V}_2$ irreducibles y $\mathcal{V}_1 \perp \mathcal{V}_2$.
- \mathcal{V} es irreducible si no es reducible.
- \mathcal{V} es primitivo si es invariante e irreducible.

Proposición 1.5. Si $f_A = f$ para cualquier $A \in \mathbb{O}^d$ entonces $f(x)$ depende de x por medio de $|x|$, luego f es constante en una esfera de radio arbitrario.

Demostración. Sean $x, y \in \mathbb{R}^d$ con $|x| = |y|$, podemos encontrar una matriz $A \in \mathbb{O}^d$ tal que $Ax = y$. Entonces $f(x) = f_A(x) = f(y)$. □

Definición 1.6. Dado $f : \mathbb{R}^d \rightarrow \mathbb{C}$ se define $\text{span}\{f_A : A \in \mathbb{O}^d\}$ como el espacio de las series $\sum c_j f_{A_j}$ convergentes con $A_j \in \mathbb{O}^d, c_j \in \mathbb{C}$

De la definición se deduce que $\text{span}\{f_A : A \in \mathbb{O}^d\}$ es un subespacio de funciones. Además, si \mathcal{V} es un espacio finito dimensional $\mathcal{V} = \text{span}\{f_A\}$

Introduciremos los espacios de armónicos esféricos de diferentes órdenes como subespacios primitivos de $C(\mathbb{S}^{d-1})$.

1.2.1. Espacios de Polinomios Homogéneos.

Consideramos \mathcal{H}_n^d el espacio de polinomios homogéneos de grado n en d dimensiones. Estas funciones son de la forma:

$$\sum_{|\alpha|=n} a_\alpha x^\alpha, a_\alpha \in \mathbb{C}$$

Ejemplo 1.7.

$$\begin{aligned}\mathbb{H}_2^2 &= \{a_1 x_1^2 + a_2 x_1 x_2 + a_3 x_2^2\} \\ \mathbb{H}_3^2 &= \{a_1 x_1^3 + a_2 x_2^3 + a_3 x_1^2 x_2 + a_4 x_1 x_2^2\}\end{aligned}$$

A continuación vamos a estudiar la dimensión de \mathcal{H}_n^d , llegando a la conclusión de que es un espacio invariante finito dimensional. Para determinar $\dim \mathcal{H}_n^d$ contamos los monomios de grado n , es decir, x^α con $\alpha_i \geq 0$ y verificando $\alpha_1 + \alpha_2 + \dots + \alpha_d = n$. Tomamos un conjunto $S = \{1, 2, \dots, n+d-1\}$. Seleccionamos $d-1$ números de dicho conjunto y los llamamos $\beta_i, 1 \leq i \leq d-1$. Definimos $\beta_0 = 0$ y $\beta_d = n+d$.

Ahora, tomamos α_i como el número elementos de S entre dos β_i consecutivos, es decir, $\alpha_i = \beta_i - \beta_{i-1} - 1, 1 \leq i \leq d$. Tenemos que

$$\sum_{i=1}^d \alpha_i = \sum_{i=1}^d \beta_i - \beta_{i-1} - \sum_{i=1}^d 1 = \beta_d - d = n + d - d = n$$

Por tanto tenemos una biyección entre el conjunto de α_i que suman n y el conjunto de β_i . Finalmente, contamos las distintas elecciones posibles de los β_i y tenemos que

$$\dim \mathbb{H}_n^d = \binom{n+d-1}{d-1} = \binom{n+d-1}{n}$$

Definición 1.8. Una función f es armónica si $\Delta f(x) = 0$.

Lema 1.9. Si $\Delta f = 0$, entonces $\Delta f_A = 0, \quad \forall A \in \mathbb{O}^d$

Demostración. Sea $y = Ax$, entonces $\nabla_x = A \nabla_y$. Como $A \in \mathbb{O}^d$ se tiene que

$$\Delta_x = \nabla_x \cdot \nabla_x = \nabla_y \cdot \nabla_y = \Delta_y$$

□

A continuación, vamos a ver un subespacio de \mathcal{H}_n^d importante.

Definición 1.10. Llamamos $\mathbb{Y}_n(\mathbb{R}^d)$ al espacio de los polinomios homogéneos de grado n en \mathbb{R}^d que son armónicos.

Ejemplo 1.11. $\mathbb{Y}_n(\mathbb{R}^d) = \mathbb{H}_n^d$ si $n = 0$ o $n = 1$

Para $d = 1$, $\mathbb{Y}_n(\mathbb{R}) = \emptyset$ para $n \geq 2$

Para $d = 2$, $\mathbb{Y}_n(\mathbb{R}^2)$, los polinomios de la forma $(x_1 + ix_2)^n$ pertenecen a $\mathbb{Y}_n(\mathbb{R}^2)$. En particular, $\mathbb{Y}_2(\mathbb{R}^2)$ está formado por polinomios de la forma $a(x_1^2 - x_2^2) + bx_1x_2$, $a, b \in \mathbb{C}$.

Calculamos ahora la dimensión de $\mathbb{Y}_n(\mathbb{R}^d)$. Llamaremos $N_{n,d}$ a la dimensión de $\mathbb{Y}_n(\mathbb{R}^d)$. Sea $H_n \in \mathbb{H}_n^d$, dicho polinomio puede ser escrito de la forma

$$H_n(x_1, \dots, x_d) = \sum_{j=0}^n (x_d)^j h_{n-j}(x_1, \dots, x_{d-1}), \quad h_{n-j} \in \mathbb{H}_{n-j}^{d-1}$$

Aplicamos el operador laplaciano a H_n ,

$$\Delta_{(d)} H_n(x_{(d)}) = \sum_{j=0}^{n-2} (x_d)^j [\Delta_{(d-1)} h_{n-j}(x_{(d-1)}) + (j+2)(j+1)h_{n-j-2}(x_{(d-1)})]$$

Luego, si $H_n \in \mathbb{Y}_n(\mathbb{R}^d)$ entonces $\Delta_{(d)} H_n(x_{(d)}) \equiv 0$ y

$$h_{n-j-2} = -\frac{1}{(j+2)(j+1)} \Delta_{(d-1)} h_{n-j}, \quad 0 \leq j \leq n-2$$

En consecuencia un armónico homogéneo está únicamente determinado por $h_n \in \mathbb{H}_n^{d-1}$ y $h_{n-1} \in \mathbb{H}_{n-1}^{d-2}$. De este modo, obtenemos la siguiente relación

$$N_{n,d} = \dim \mathbb{H}_n^{d-1} + \dim \mathbb{H}_{n-1}^{d-2}$$

Usando la formula obtenida para $\dim \mathbb{H}_n^d$ se tiene que para $d \geq 2$,

$$N_{n,d} = \frac{(2n+d-2)(n+d-3)!}{n!(d-2!)}, \quad n \in \mathbb{N}$$

1.2.2. Armónicos de Legendre y Polinomios de Legendre

Ahora, nos centraremos en unos armónicos homogéneos especiales, los armónicos de Legendre de grado n en d dimensiones.

Definición 1.12. Se define los armónicos de Legendre, $L_{n,d} : \mathbb{R}^d \rightarrow \mathbb{R}$ verificando las siguientes condiciones:

- $L_{n,d} \in \mathbb{Y}_n(\mathbb{R}^d)$
- $L_{n,d}(Ax) = L_{n,d}(x) \quad \forall A \in \mathbb{O}^d(e_d), \forall x \in \mathbb{R}^d$
- $L_{n,d}(e_d) = 1$

Nota 1.13. La segunda condición implica que $h_{n-j}(A_1 x_{(d-1)}) = h_{n-j}(x_{(d-1)})$, $\forall A_1 \in \mathbb{O}^{(d-1)}, \quad x_{(d-1)} \in \mathbb{R}^{d-1}, \quad 0 \leq j \leq n$

De la Proposición 1.5 se deduce que por ser h_{n-j} polinomio homogéneo, $(n-j)$ es par y

$$h_{n-j}(x_{(d-1)}) = \begin{cases} c_k |x_{(d-1)}|^{2k} & \text{si } n-j = 2k \\ 0 & \text{si } n-j = 2k+1 \end{cases} \quad (1.2.1)$$

Por tanto,

$$L_{n,d}(x) = \sum_{k=0}^{[n/2]} c_k |x_{(d-1)}|^{2k} (x_d)^{n-2k}$$

Determinamos ahora los coeficientes c_k

$$c_k = -\frac{(n-2k+2)(n-2k+1)}{2k(2k+d-3)} c_{k-1}, \quad 1 \leq k \leq [n/2]$$

Usando la condición de normalidad se tiene que $c_0 = 1$ y

$$c_k = (-1)^k \frac{n! \Gamma(\frac{d-1}{2})}{4^k k! (n-2k)! \Gamma(k + \frac{d-1}{2})}, \quad 0 \leq k \leq [n/2]$$

Finalmente, obtenemos la siguiente expresión

$$L_{n,d}(x) = n! \Gamma(\frac{d-1}{2}) \sum_{k=0}^{[n/2]} (-1)^k \frac{|x_{(d-1)}|^{2k} (x_d)^{n-2k}}{4^k k! (n-2k)! \Gamma(k + \frac{d-1}{2})}$$

Usando coordenadas polares (véase apéndice B) $x_{(d)} = r\xi_{(d)}$, $\xi_{(d)} = te_d + \sqrt{1-t^2}\xi_{(d-1)}$, definimos el polinomio de Legendre de grado n en d dimensiones, $P_{n,d}(t) = L_{n,d}(\xi_{(d)})$ como la restricción a la esfera unidad del armónico de Legendre. Por tanto

$$P_{n,d}(t) = n! \Gamma(\frac{d-1}{2}) \sum_{k=0}^{[n/2]} (-1)^k \frac{(1-t^2)^k t^{n-2k}}{4^k k! (n-2k)! \Gamma(k + \frac{d-1}{2})}$$

Nota 1.14. $P_{n,d}(1) = 1$ y $L_{n,d}(x) = L_{n,d}(r\xi_{(d)}) = r^n P_{n,d}(t)$

1.2.3. Armónicos Esféricos

Definición 1.15. Se llama espacio de armónicos esféricos de orden n en d dimensiones a $\mathbb{Y}_n^d = \mathbb{Y}_n(\mathbb{R}^d)|_{\mathbb{S}^{d-1}}$

De la definición se deduce que un armónico esférico $\mathbb{Y}_n \in \mathbb{Y}_n^d$ está asociado a un armónico homogéneo $\mathbb{H}_n \in \mathbb{Y}_n^d$ de la siguiente forma:

$$\mathbb{H}_n(r\xi) = r^n \mathbb{Y}_n(\xi)$$

En consecuencia, $\dim \mathbb{Y}_n^d = N_{n,d}$

Teorema 1.16. Sea $Y_n \in \mathbb{Y}_n^d$ y $\xi \in \mathbb{S}^{d-1}$. Entonces Y_n es invariante respecto a $\mathbb{O}^d(\xi)$, si y sólo si, $Y_n(\eta) = Y_n(\xi)P_{n,d}(\xi \cdot \eta)$, $\forall \eta \in \mathbb{S}^{d-1}$

Demostración. (\Rightarrow) Dado que ξ es un vector unitario podemos encontrar $A_1 \in \mathbb{O}^d$ tal que $\xi = A_1 e_d$. Sea $\tilde{Y}_n(\eta) = Y_n(A_1 \eta)$, $\eta \in \mathbb{S}^{d-1}$, que es invariante respecto a $\mathbb{O}^d(e_d)$. De la definición de armónico de Legendre sabemos que $r^n \tilde{Y}_n(\eta) = c_1 L_{n,d}(r^n \eta)$, $r \geq 0$, $\eta \in \mathbb{S}^{d-1}$ con c_1 una constante.

Por tanto, $\tilde{Y}_n(\eta) = c_1 L_{n,d}(\eta)$ y tomando $\eta = e_d$ tenemos que $c_1 = \tilde{Y}_n(e_d)$. Finalmente como

$$\tilde{Y}_n(\eta) = \tilde{Y}_n(e_d)P_{n,d}(\eta \cdot e_d)$$

se tiene que

$$\begin{aligned} Y_n(\eta) &= \tilde{Y}_n(A_1^T \eta) = Y_n(A_1^T e_d)P_{n,d}(A_1^T \eta \cdot e_d) = \\ &= Y_n(A_1^T e_d)P_{n,d}(\eta \cdot A_1 e_d) = Y_n(\xi)P_{n,d}(\xi \cdot \eta) \end{aligned}$$

(\Leftarrow) Obvio □

1.3. Teorema de Adición. Consecuencias.

Teorema 1.17. Sea $\{Y_{n,j} : 1 \leq j \leq N_{n,d}\}$ una base ortonormal de \mathbb{Y}_n^d , es decir,

$$\int_{\mathbb{S}^{d-1}} Y_{n,j}(\eta) \overline{Y_{n,k}(\eta)} d\mathbb{S}^{d-1} = \delta_{j,k}, \quad 1 \leq j, k \leq N_{n,d}$$

Entonces,

$$\sum_{j=1}^{N_{n,d}} Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)} = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} P_{n,d}(\xi \cdot \eta), \quad \forall \xi, \eta \in \mathbb{S}^{d-1}$$

Demostración. Sean $A \in \mathbb{O}^d$ y $1 \leq k \leq N_{n,d}$, $Y_{n,k}(A\xi) \in \mathbb{Y}_n^d$ podemos escribir

$$Y_{n,k}(A\xi) = \sum_{j=1}^{N_{n,d}} c_{kj} Y_{n,j}(\xi), \quad c_{kj} \in \mathbb{C}$$

Como

$$\int_{\mathbb{S}^{d-1}} Y_{n,j}(A\xi) \overline{Y_{n,k}(A\xi)} d\mathbb{S}^{d-1}(\xi) = \int_{\mathbb{S}^{d-1}} Y_{n,j}(\eta) \overline{Y_{n,k}(\eta)} d\mathbb{S}^{d-1}(\eta) = \delta_{j,k}$$

tenemos que

$$\delta_{jk} = \sum_{l,m=1}^{N_{n,d}} c_{j,l} \overline{c_{k,m}} (Y_{n,l}, Y_{n,m}) = \sum_{l,m=1}^{N_{n,d}} c_{j,l} \overline{c_{k,m}}$$

Sea $C = (c_{j,l})$ y C^H su matriz conjugada transpuesta. Se verifica que $CC^H = I$ y $C^H C = I$ luego C es unitaria y

$$\sum_{j=1}^{N_{n,d}} \overline{c_{jl}} c_{jk} = \delta_{lk} \quad 1 \leq l, k \leq N_{n,d}$$

Ahora, consideramos la suma

$$Y(\xi, \eta) = \sum_{j=1}^{N_{n,d}} Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)}, \quad \xi, \eta \in \mathbb{S}^{d-1}$$

Para $A \in \mathbb{O}^d$, se tiene que

$$\begin{aligned} Y(A\xi, A\eta) &= \sum_{j=1}^{N_{n,d}} Y_{n,j}(A\xi) \overline{Y_{n,j}(A\eta)} = \sum_{j,k,l=1}^{N_{n,d}} c_{jk} \overline{c_{jl}} Y_{n,k}(\xi) \overline{Y_{n,l}(\eta)} \\ &= \sum_{j=1}^{N_{n,d}} Y_{n,k}(\xi) \overline{Y_{n,k}(\eta)} = Y(\xi, \eta) \end{aligned}$$

luego, fijado ξ , $Y(\xi, \cdot) \in \mathbb{Y}_n^d$ es invariante respecto a $\mathbb{O}^d(\xi)$. Por el Teorema 1.16 $Y(\xi, \eta) = Y(\xi, \xi) P_{n,d}(\xi, \eta)$. Análogamente, $Y(\xi, \eta) = Y(\eta, \eta) P_{n,d}(\xi, \eta)$. En consecuencia, $Y(\xi, \xi) = Y(\eta, \eta)$ y es constante en \mathbb{S}^{d-1} . Para determinar dicha constante, integramos la igualdad $Y(\xi, \xi) = \sum_{j=1}^{N_{n,d}} |Y_{n,j}(\xi)|^2$ sobre la esfera, obteniendo que

$$Y(\xi, \xi) |\mathbb{S}^{d-1}| = \sum_{j=1}^{N_{n,d}} \int_{\mathbb{S}^{d-1}} |Y_{n,j}(\xi)|^2 d\mathbb{S}^{d-1} = N_{n,d}$$

Por tanto, $Y(\xi, \xi) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|}$ y se cumple

$$\sum_{j=1}^{N_{n,d}} Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)} = Y(\xi, \eta) = Y(\xi, \xi) P_{n,d}(\xi, \eta) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} P_{n,d}(\xi, \eta)$$

□

Ejemplo 1.18. En el caso $d = 2$

$$\sum_{j=1}^2 Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)} = \frac{1}{\pi} P_{n,2}(\xi, \eta) \quad \forall \xi, \eta \in \mathbb{S}^1$$

Si tomamos $\xi = (\cos \theta, \sin \theta)^T$, $\eta = (\cos \psi, \sin \psi)^T$. Entonces $\xi \cdot \eta = \cos(\theta - \psi)$ y podemos tomar

$$Y_{n,1}(\xi) = \frac{1}{\sqrt{\pi}} \cos(n\theta) \quad (1.3.1)$$

$$Y_{n,2}(\xi) = \frac{1}{\sqrt{\pi}} \sin(n\theta) \quad (1.3.2)$$

como base ortonormal de \mathbb{Y}_n^2 .

Ejemplo 1.19. Si $d = 3$

$$\sum_{j=1}^{2n+1} Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)} = \frac{2n+1}{4\pi} P_{n,3}(\xi \cdot \eta) \quad \forall \xi, \eta \in \mathbb{S}^2$$

Veamos ahora algunas aplicaciones del teorema de adición. En primer lugar, aplicaremos el teorema para encontrar una expresión reducida del núcleo de \mathbb{Y}_n^d .

Cada $Y_n \in \mathbb{Y}_n^d$ puede escribirse de la forma

$$Y_n(\xi) = \sum_{j=1}^{N_{n,d}} (Y_n, Y_{n,j})_{\mathbb{S}^{d-1}} Y_{n,j}(\xi)$$

Aplicando el Teorema 1.17,

$$\begin{aligned} Y_n(\xi) &= \int_{\mathbb{S}^{d-1}} Y_n(\eta) \sum_{j=1}^{N_{n,d}} Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)} d\mathbb{S}^{d-1}(\eta) \\ &= \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \int_{\mathbb{S}^{d-1}} P_{n,d}(\xi \cdot \eta) Y_n(\eta) d\mathbb{S}^{d-1}(\eta) \end{aligned}$$

Por tanto,

$$K_{n,d}(\xi \cdot \eta) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} P_{n,d}(\xi \cdot \eta)$$

es el núcleo reproductor de \mathbb{Y}_n^d , es decir,

$$Y_n(\xi) = (Y_n, K_{n,d}(\xi, \cdot))_{\mathbb{S}^{d-1}} \quad \forall Y_n \in \mathbb{Y}_n^d, \xi \in \mathbb{S}^{d-1}$$

Definimos $\mathbb{Y}_{0:m}^d = \bigoplus_{n=0}^m \mathbb{Y}_n^d$ como el espacio de todos los armónicos esféricos de orden menor o igual a m . Entonces

$$K_{0:m,d}(\xi, \eta) = \frac{1}{|\mathbb{S}^{d-1}|} \sum_{n=0}^m N_{n,d} P_{n,d}(\xi \cdot \eta)$$

es el núcleo reproductor de $\mathbb{Y}_{0:m}^d$.

A continuación, obtendremos cotas para los armónicos esféricos y los polinomios de Legendre.

Proposición 1.20. *Se verifican las siguientes desigualdades:*

$$\|Y_n\|_\infty \leq \left(\frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \right)^{\frac{1}{2}} \|Y_n\|_{L^2(\mathbb{S}^{d-1})} \quad (1.3.3)$$

$$|P_{n,d}(t)| \leq 1 = P_{n,d}(1) \quad (1.3.4)$$

Demostración. Tomando $\xi \in \mathbb{S}^{d-1}$ por el teorema de adición

$$\sum_{j=1}^{N_{n,d}} |Y_{n,j}(\xi)|^2 = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} P_{n,d}(|\xi|^2) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \quad (1.3.5)$$

Por tanto, $\max\{|Y_{n,j}(\xi)|\} \leq \left(\frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \right)^{1/2}$.

Por otro lado,

$$\begin{aligned} \int_{\mathbb{S}^{d-1}} |Y_n(\xi)|^2 dS^{d-1}(\xi) &= \int_{\mathbb{S}^{d-1}} \sum_{j=1}^{N_{n,d}} \sum_{k=1}^{N_{n,d}} (Y_n, Y_{n,j})_{\mathbb{S}^{d-1}} (Y_n, Y_{n,k})_{\mathbb{S}^{d-1}} Y_{n,j} Y_{n,k} dS^{d-1} \\ &= \sum_{j=1}^{N_{n,d}} |(Y_n, Y_{n,j})_{\mathbb{S}^{d-1}}|^2 \end{aligned}$$

Finalmente uniendo lo anterior se tiene que

$$\begin{aligned} |Y_n(\xi)|^2 &\leq \sum_{j=1}^{N_{n,d}} |(Y_n, Y_{n,j})_{\mathbb{S}^{d-1}}|^2 \sum_{j=1}^{N_{n,d}} |Y_{n,j}|^2 \\ &= \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \int_{\mathbb{S}^{d-1}} |Y_n|^2 dS^{d-1} = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \|Y_n\|_{L^2(\mathbb{S}^{d-1})}^2 \end{aligned}$$

y en consecuencia

$$\|Y_n\|_\infty \leq \left(\frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \right)^{1/2} \|Y_n\|_{L^2(\mathbb{S}^{d-1})}$$

Ahora, usando (1.3.5) y el teorema de adición tenemos que

$$\begin{aligned} \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} |P_{n,d}(\xi \cdot \eta)| &= \sum_{j=1}^{N_{n,d}} |Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)}| \\ &\leq \left(\sum_{j=1}^{N_{n,d}} Y_{n,j}^2(\xi) \right)^{1/2} \left(\sum_{j=1}^{N_{n,d}} Y_{n,j}^2(\eta) \right)^{1/2} = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \end{aligned}$$

es decir,

$$|P_{n,d}(\xi \cdot \eta)| \leq 1 = P_{n,d}(1)$$

□

Proposición 1.21. *Se verifica la siguiente igualdad*

$$\int_{\mathbb{S}^{d-1}} |P_{n,d}(\xi \cdot \eta)|^2 dS^{d-1}(\eta) = \frac{|\mathbb{S}^{d-1}|}{N_{n,d}}$$

Demostración.

$$\begin{aligned} \int_{\mathbb{S}^{d-1}} |P_{n,d}(\xi \cdot \eta)|^2 dS^{d-1}(\eta) &= \left(\frac{|\mathbb{S}^{d-1}|}{N_{n,d}} \right)^2 \int_{\mathbb{S}^{d-1}} \left| \sum_{j=1}^{N_{n,d}} Y_{n,j}(\xi) \overline{Y_{n,j}(\eta)} \right|^2 dS^{d-1}(\eta) \\ &= \left(\frac{|\mathbb{S}^{d-1}|}{N_{n,d}} \right)^2 \sum_{j=1}^{N_{n,d}} |Y_{n,j}(\xi)|^2 = \frac{|\mathbb{S}^{d-1}|}{N_{n,d}} \end{aligned}$$

□

Teorema 1.22. *Para cualquier $n \in \mathbb{N}_0$ y $d \in \mathbb{N}$ el espacio \mathbb{Y}_n^d es irreducible*

Demostración. Razonamos por reducción al absurdo. Supongamos que \mathbb{Y}_n^d es reducible entonces $\exists V_1, V_2$ no vacíos, verificando que $\mathbb{Y}_n^d = V_1 + V_2$ y $V_1 \perp V_2$. Tomamos una base ortonormal de \mathbb{Y}_n^d tal que las primeras N_1 funciones recubren V_1 y las restantes $N_2 = N_{n,d} - N_1$ recubren V_2 . Podemos aplicar el teorema de adición a V_1 y V_2 con las funciones de Legendre $P_{n,d,1}$ y $P_{n,d,2}$.

Como $V_1 \perp V_2$

$$\int_{\mathbb{S}^{d-1}} P_{n,d,1}(\xi \cdot \eta) P_{n,d,2}(\xi \cdot \eta) d\mathbb{S}^{d-1}(\eta) = 0 \quad \forall \xi \in \mathbb{S}^{d-1} \quad (1.3.6)$$

Fijamos $\xi \in \mathbb{S}^{d-1}$ y sea ϕ una función tal que $\phi(\eta) = P_{n,d,1}(\xi \cdot \eta)$. Tomamos $A \in \mathbb{O}^d(\xi)$ y se cumple que $A^T \xi = \xi$. Entonces

$$P_{n,d,1}(\xi \cdot A\eta) = P_{n,d,1}(A^T \xi \cdot \eta) = P_{n,d,1}(\xi \cdot \eta)$$

es decir, ϕ es invariante respecto a $\mathbb{O}^d(\xi)$. Por el Teorema 1.16

$$P_{n,d,1}(\xi \cdot \eta) = P_{n,d,1}(\xi \cdot \xi) \cdot P_{n,d}(\xi \cdot \eta) = P_{n,d}(\xi \cdot \eta)$$

Razonando de forma análoga para $P_{n,d,2}$ se tiene que

$$P_{n,d,2}(\xi \cdot \eta) = P_{n,d}(\xi \cdot \eta)$$

Sin embargo, tenemos que

$$0 = \int_{\mathbb{S}^{d-1}} P_{n,d,1}(\xi \cdot \eta) P_{n,d,2}(\xi \cdot \eta) d\mathbb{S}^{d-1}(\eta) = \int_{\mathbb{S}^{d-1}} |P_{n,d}(\xi \cdot \eta)|^2 d\mathbb{S}^{d-1}(\eta) = \frac{|\mathbb{S}^{d-1}|}{N_{n,d}}$$

Hemos llegado a una contradicción, por tanto, \mathbb{Y}_n^d es irreducible

□

1.4. El Operador de Proyección

Buscamos la mejor aproximación de una función $f \in L^2(\mathbb{S}^{d-1})$ en \mathbb{Y}_n^d , es decir, $\inf\{\|f - Y_n\|_{L^2(\mathbb{S}^{d-1})} : Y_n \in \mathbb{Y}_n^d\}$. Si $\{Y_{n,j} : 1 \leq j \leq N_{n,d}\}$ es una base ortonormal de \mathbb{Y}_n^d entonces la solución es la proyección de f en \mathbb{Y}_n^d que está definida para $f \in L^1(\mathbb{S}^{d-1})$

$$(\mathcal{P}_{n,d}f)(\xi) = \sum_{j=1}^{N_{n,d}} (f, Y_{n,j})_{\mathbb{S}^{d-1}} Y_{n,j}(\xi)$$

Definición 1.23. Se define la proyección de $f \in L^1(\mathbb{S}^{d-1})$ en \mathbb{Y}_n^d como

$$(\mathcal{P}_{n,d}f)(\xi) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \int_{\mathbb{S}^{d-1}} P_{n,d}(\xi \cdot \eta) f(\eta) d\mathbb{S}^{d-1}(\eta), \quad \xi \in \mathbb{S}^{d-1}$$

Nota 1.24. El operador $\mathcal{P}_{n,d}$ es lineal

Proposición 1.25. Sea $f \in L^1(\mathbb{S}^{d-1})$ entonces $\|\mathcal{P}_{n,d}f\|_{L^1(\mathbb{S}^{d-1})} \leq N_{n,d}\|f\|_{L^1(\mathbb{S}^{d-1})}$

Demostración. Como $|\mathcal{P}_{n,d}(t)| \leq 1$ entonces dado $\xi \in \mathbb{S}^{d-1}$

$$|\mathcal{P}_{n,d}f(\xi)| \leq \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \int_{\mathbb{S}^{d-1}} |f(\eta)| d\mathbb{S}^{d-1}(\eta) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \|f\|_{L^1(\mathbb{S}^{d-1})}$$

Por tanto,

$$\|\mathcal{P}_{n,d}f\|_{L^1(\mathbb{S}^{d-1})} \leq N_{n,d}\|f\|_{L^1(\mathbb{S}^{d-1})}$$

□

Proposición 1.26. Sea $f \in L^2(\mathbb{S}^{d-1})$ entonces

$$\|\mathcal{P}_{n,d}f\|_{L^2(\mathbb{S}^{d-1})} \leq (N_{n,d})^{1/2} \|f\|_{L^2(\mathbb{S}^{d-1})}$$

Demostración. Sea $\xi \in \mathbb{S}^{d-1}$

$$|(\mathcal{P}_{n,d}f)(\xi)|^2 \leq \left(\frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \right)^2 \int_{\mathbb{S}^{d-1}} |P_{n,d}(\xi \cdot \eta)|^2 d\mathbb{S}^{d-1}(\eta) \int_{\mathbb{S}^{d-1}} |f(\eta)|^2 d\mathbb{S}^{d-1}(\eta)$$

Usando la Proposición 1.21 tenemos que

$$|(\mathcal{P}_{n,d}f)(\xi)|^2 \leq \left(\frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \right)^2 \frac{|\mathbb{S}^{d-1}|}{N_{n,d}} \int_{\mathbb{S}^{d-1}} |f(\eta)|^2 d\mathbb{S}^{d-1}(\eta) = \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \|f\|_{L^2(\mathbb{S}^{d-1})}^2$$

Por tanto,

$$\begin{aligned} \|\mathcal{P}_{n,d}f\|_{L^2(\mathbb{S}^{d-1})}^2 &\leq \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \|f\|_{L^2(\mathbb{S}^{d-1})}^2 \\ \|\mathcal{P}_{n,d}f\|_{L^2(\mathbb{S}^{d-1})} &\leq N_{n,d}^{1/2} \|f\|_{L^2(\mathbb{S}^{d-1})} \end{aligned}$$

□

Proposición 1.27. *El operador proyección $\mathcal{P}_{n,d}$ conmuta con las transformaciones ortogonales, es decir, $\mathcal{P}_{n,d}f_A = (\mathcal{P}_{n,d}f)_A \quad \forall A \in \mathbb{O}^d$*

Demostración.

$$\begin{aligned} (\mathcal{P}_{n,d}f_A)(\xi) &= \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \int_{\mathbb{S}^{d-1}} P_{n,d}(\xi \cdot \eta) f(A\eta) d\mathbb{S}^{d-1}(\eta) \\ &= \frac{N_{n,d}}{|\mathbb{S}^{d-1}|} \int_{\mathbb{S}^{d-1}} P_{n,d}(A\xi \cdot \zeta) f(\zeta) d\mathbb{S}^{d-1}(\zeta) = (\mathcal{P}_{n,d}f)_A(\xi) \end{aligned}$$

□

Corolario 1.28. *Si \mathbb{V} es un espacio invariante, entonces $\mathcal{P}_{n,d}\mathbb{V} = \{\mathcal{P}_{n,d}f : f \in \mathbb{V}\}$ es un subespacio invariante de \mathbb{Y}_n^d .*

Teorema 1.29. *Si \mathbb{V} es un espacio primitivo de $C(\mathbb{S}^{d-1})$ entonces $\mathbb{V} \perp \mathbb{Y}_n^d$ o $\mathcal{P}_{n,d}$ es una biyección de \mathbb{V} sobre \mathbb{Y}_n^d . En el último caso, $\mathbb{V} = \mathbb{Y}_n^d$*

Demostración. Veamos que si $\mathcal{P}_{n,d} : \mathbb{V} \rightarrow \mathbb{Y}_n^d$ es una biyección entonces $\mathbb{V} = \mathbb{Y}_n^d$.

Ambos espacios son de dimensión finita y tienen la misma dimensión, $N_{n,d} = \dim(\mathbb{Y}_n^d)$. Sea $\{V_j : 1 \leq j \leq N_{n,d}\}$ una base ortonormal de \mathbb{V} . Por ser \mathbb{V} primitivo, para cada $A \in \mathbb{O}^d$

$$V_j(A\xi) = \sum_{k=1}^{N_{n,d}} c_{jk} V_k(\xi), \quad c_{jk} \in \mathbb{C}$$

siendo la matriz (c_{jk}) unitaria. Definimos la función

$V(\xi, \eta) = \sum_{k=1}^{N_{n,d}} V_j(\xi) \overline{V_j(\eta)}$ y $V(A\xi, A\eta) = V(\xi, \eta)$, $\forall A \in \mathbb{O}^d$. Dados $\xi, \eta \in \mathbb{S}^{d-1}$ podemos encontrar $A \in \mathbb{O}^d$ tal que, $A\xi = e_d, A\eta = te_d + (1-t^2)^{\frac{1}{2}}e_{d-1}$ para $t = \xi \cdot \eta$. Entonces $V(\xi, \eta) = V(e_d, te_d + (1-t^2)^{\frac{1}{2}}e_{d-1})$ es una función de $t = \xi \cdot \eta$. Llamaremos a esta función $P_d(t)$. Fijado ξ , la aplicación $\eta \rightarrow \overline{P_d(\xi \cdot \eta)}$ es una función en \mathbb{V} , del mismo modo, fijado ζ la aplicación $\eta \rightarrow P_{n,d}(\zeta \cdot \eta)$ es una función en \mathbb{Y}_n^d . Consideramos la función $\phi(\xi, \zeta) = \int_{\mathbb{S}^{d-1}} \overline{P_d(\xi \cdot \eta)} P_{n,d}(\zeta \cdot \eta) d\mathbb{S}^{d-1}(\eta)$ con $\phi(A\xi, A\zeta) = \phi(\xi, \zeta), \forall A \in \mathbb{O}^d$. Es decir, $\phi(\xi, \zeta)$ depende sólo de $\xi \cdot \zeta$. ϕ pertenece a \mathbb{V} y a \mathbb{Y}_n^d , luego o $\mathbb{V} = \mathbb{Y}_n^d$ o $\phi \equiv 0$. En el último caso tenemos que

$$\sum_{j,k=1}^{N_{n,d}} \overline{V_j(\xi)} Y_{n,k}(\zeta) (V_j, Y_{n,k})_{L^2(\mathbb{S}^{d-1})} = 0 \quad \forall \xi, \zeta \in \mathbb{S}^{d-1}$$

donde $\{Y_{n,k} : 1 \leq k \leq N_{n,d}\}$ es una base ortonormal de \mathbb{Y}_n^d . Como cada elemento de los conjuntos $\{V_j : 1 \leq j \leq N_{n,d}\}$ y $\{Y_{n,j} : 1 \leq j \leq N_{n,d}\}$ son linealmente independientes, deducimos de la igualdad anterior que

$$(V_j, Y_{n,k})_{L^2(\mathbb{S}^{d-1})} = 0, \quad 1 \leq j, k \leq N_{n,d}$$

Por tanto, $\mathbb{V} \perp \mathbb{Y}_n^d$.

□

Corolario 1.30. Para $m \neq n$, $\mathbb{Y}_m^d \perp \mathbb{Y}_n^d$

Demostración. Sean $Y_m \in \mathbb{Y}_m^d$ e $Y_n \in \mathbb{Y}_n^d$ restricciones sobre la esfera de $H_m \in \mathbb{Y}_m(\mathbb{R}^d)$ y $H_n \in \mathbb{Y}_n(\mathbb{R}^d)$ respectivamente. Como $\Delta H_m(x) = \Delta H_n(x) = 0$ tenemos que

$$\int_{||x|| < 1} (H_m \Delta H_n - H_n \Delta H_m) dx = 0$$

Aplicando la fórmula de Green

$$\int_{\mathbb{S}^{d-1}} (H_m \frac{\partial H_n}{\partial r} - H_n \frac{\partial H_m}{\partial r}) d\mathbb{S}^{d-1} = 0$$

Además, por ser H_m un polinomio homogéneo de grado m

$$\frac{\partial H_m(x)}{\partial r} \Big|_{x=\xi} = m Y_m(\xi), \quad \xi \in \mathbb{S}^{d-1}$$

Análogamente,

$$\frac{\partial H_n(x)}{\partial r} \Big|_{x=\xi} = m Y_n(\xi), \quad \xi \in \mathbb{S}^{d-1}$$

Por tanto,

$$\int_{\mathbb{S}^{d-1}} (n - m) Y_m(\xi) Y_n(\xi) d\mathbb{S}^{d-1}(\xi) = 0$$

Finalmente, como $m \neq n$,

$$\int_{\mathbb{S}^{d-1}} Y_m(\xi) Y_n(\xi) d\mathbb{S}^{d-1}(\xi) = 0$$

□

1.5. Generando Bases Ortonormales para Espacios de Armónicos Esféricos.

A continuación, generaremos una base ortonormal de \mathbb{Y}_n^d a partir de bases ortonormales de dimensión $d-1$. Para ello, haremos uso de las funciones de Legendre asociadas (Apéndice E).

Proposición 1.31. Si $Y_{j,d-1} \in \mathbb{Y}_j^{d-1}$ entonces $P_{n,d,j}(t) Y_{j,d-1}(\xi_{(d-1)}) \in \mathbb{Y}_n^d$ en coordenadas polares (véase Apéndice B).

Demostración. Tomamos $d \geq 3$ y

$$f(x) = \frac{i^{-j}}{|\mathbb{S}^{d-2}|} \int_{\mathbb{S}^{d-2}} (x_d + i x_{(d-1)} \cdot \eta)^n Y_{j,d-1}(\eta) d\mathbb{S}^{d-2}(\eta)$$

es un polinomio homogéneo de grado n . Usando coordenadas polares $x = |x|\xi$, $\xi = te_d + \sqrt{1-t^2}\xi_{(d-1)}$, $|t| \leq 1$, $\xi_{(d-1)} \in \mathbb{S}^{d-1}$. La restricción de $f(x)$ a la esfera es

$$f(\xi) = \frac{i^{-j}}{|\mathbb{S}^{d-2}|} \int_{\mathbb{S}^{d-2}} (t + i\sqrt{1-t^2}\xi_{(d-1)} \cdot \eta)^n Y_{j,d-1}(\eta) dS^{d-2}(\eta)$$

Ahora, aplicamos la fórmula de Funk-Hecke [1, sec. 2.5]

$$\int_{\mathbb{S}^{d-2}} (t + i\sqrt{1-t^2}\xi_{(d-1)} \cdot \eta)^n Y_{j,d-1}(\eta) dS^{d-2}(\eta) = \lambda Y_{j,d-1}(\xi)$$

siendo $\lambda = |\mathbb{S}^{d-3}| \int_{-1}^1 P_{j,d-1}(s) (t + i\sqrt{1-t^2}s)^j (1-t^2)^{\frac{d-4}{2}} dt$.

Por tanto, $f(\xi) = P_{n,d,j}(t) Y_{j,d-1}(\xi_{(d-1)})$ es un esférico armónico de orden n y dimensión d . \square

Este resultado nos permite construir una base de \mathbb{Y}_n^d a partir de bases de $\mathbb{Y}_0^{d-1}, \dots, \mathbb{Y}_n^{d-1}$

Definición 1.32. Para $d \geq 3$ y $m \leq n$ definimos el operador

$$\tilde{P}_{n,m} : \mathbb{Y}_m^{d-1} \rightarrow \mathbb{Y}_n^d$$

como

$$(\tilde{P}_{n,m} Y_{m,d-1})(\xi) = \tilde{P}_{n,d,m}(t) Y_{m,d-1}(\xi_{(d-1)}), \quad Y_{m,d-1} \in \mathbb{Y}_m^{d-1}$$

Llamaremos a $\mathbb{Y}_{n,m}^d = \tilde{P}_{n,m}(\mathbb{Y}_m^{d-1})$ el espacio de orden m en \mathbb{Y}_n^d .

El siguiente resultado nos permite descomponer \mathbb{Y}_n^d como suma ortogonal de espacios asociados.

Teorema 1.33. Para $d \geq 3$ y $n \geq 0$ se tiene que

$$\mathbb{Y}_n^d = \mathbb{Y}_{n,0}^d \oplus \dots \oplus \mathbb{Y}_{n,n}^d$$

Demostración. En primer lugar, veamos que los subespacios $\mathbb{Y}_{n,i}^d$ son ortogonales 2 a 2. Sea $0 \leq k, m \leq n$ con $k \neq m$. Para cualesquiera $Y_{k,d-1} \in \mathbb{Y}_k^{d-1}$, $Y_{m,d-1} \in \mathbb{Y}_m^{d-1}$,

$$(\tilde{P}_{n,k} Y_{k,d-1}, \tilde{P}_{n,m} Y_{m,d-1})_{L^2(\mathbb{S}^{d-1})} \quad (1.5.1)$$

$$= (Y_{k,d-1}, Y_{m,d-1})_{L^2(\mathbb{S}^{d-2})} \int_{-1}^1 \tilde{P}_{n,d,k}(t) \tilde{P}_{n,d,m}(t) (1-t^2)^{\frac{d-3}{2}} dt = 0 \quad (1.5.2)$$

Por tanto, $\mathbb{Y}_{n,k}^d \perp \mathbb{Y}_{n,m}^d$ para $k \neq m$.

Para cada $0 \leq m \leq n$, $\mathbb{Y}_{m,n}^d$ es un subespacio de \mathbb{Y}_n^d y

$$\mathbb{Y}_n^d \supset \mathbb{Y}_{n,0}^d \oplus \dots \oplus \mathbb{Y}_{n,n}^d$$

Como $\tilde{P}_{n,m} : \mathbb{Y}_m^{d-1} \rightarrow \mathbb{Y}$, es una biyección entonces

$$\dim \mathbb{Y}_{n,m}^d = \dim \mathbb{Y}_m^{d-1} = N_{m,d-1}$$

Por otro lado, $\sum_{m=0}^n \dim \mathbb{Y}_{n,m}^d = \sum_{m=0}^n N_{m,d-1} = N_{n,d} = \dim \mathbb{Y}_n^d$. Es decir, ambos lados de la igualdad son espacios de dimensión finita con la misma dimensión. \square

Nota 1.34. Si $\{Y_{m,d-1,j} : 1 \leq j \leq N_{m,d-1}\}$ es una base ortonormal de \mathbb{Y}_m^{d-1} , $0 \leq m \leq n$ entonces $\{\tilde{P}_{n,d,m} Y_{m,d-1,j}(\xi_{(d-1)}) : 1 \leq j \leq N_{m,d-1}, 0 \leq m \leq n\}$ es una base ortonormal de \mathbb{Y}_n^d

A partir de la base ortonormal de \mathbb{Y}_n^2 obtenida en (1.3.1) y del resultado anterior, construiremos una base ortonormal de \mathbb{Y}_n^3 .

$$\text{Usaremos que } \xi_{(3)} = te_3 + \sqrt{1-t^2} \begin{pmatrix} \xi_{(2)} \\ 0 \end{pmatrix} \text{ con } t = \cos \theta, \quad 0 \leq \theta \leq \pi,$$

$$\xi_{(2)} = (\cos(\phi), \sin(\phi))^T, 0 \leq \phi \leq 2\pi \text{ Por lo visto anteriormente,}$$

$$\left\{ Y_{m,2,1}(\xi_{(2)}) = \frac{1}{\sqrt{\pi}} \cos(m\phi), Y_{m,2,2}(\xi_{(2)}) = \frac{1}{\sqrt{\pi}} \sin(m\phi) \right\}$$

es una base ortonormal de \mathbb{Y}_m^2 .

Por otro lado tenemos que (E.7)

$$\tilde{P}_{n,3,m}(t) = \left[\frac{(n + \frac{1}{2})(n-m)!}{(n+m)!} \right]^{\frac{1}{2}} (1-t^2)^{\frac{m}{2}} P_{n,3}^{(m)}(t)$$

Entonces una base ortonormal viene dada por las funciones

$$\left[\frac{(2n+1)(n-m)!}{2\pi(n+m)!} \right]^{\frac{1}{2}} (\sin \theta)^m P_{n,3}^{(m)}(\cos \theta) \cos(m\phi) \quad 0 \leq m \leq n \quad (1.5.3)$$

$$\left[\frac{(2n+1)(n-m)!}{2\pi(n+m)!} \right]^{\frac{1}{2}} (\sin \theta)^m P_{n,3}^{(m)}(\cos \theta) \sin(m\phi), \quad 1 \leq m \leq n \quad (1.5.4)$$

Esta base también puede ser escrita de otra forma más cómoda para realizar cálculos

$$(-1)^{(m+|m|)/2} \left[\frac{(2n+1)(n-|m|)!}{4\pi(n+|m|)!} \right]^{\frac{1}{2}} (\sin \theta)^m P_{n,3}^{(m)}(\cos \theta) e^{im\phi}, \quad -n \leq m \leq n \quad (1.5.5)$$

Capítulo 2

Cálculo del Gradiente

2.1. El gradiente de los armónicos esféricos.

Para el cálculo del gradiente usaremos una expresión de la base en términos de los polinomios de Gegenbauer. De la Proposición D.2, se deduce que esta base es equivalente a la calculada anteriormente.

Teorema 2.1. Sean, $T_n(t), U_n(t)$ los polinomios de Chebyshev de 1ª y 2ª clase respectivamente. Y definimos

$$g_{0,n}(x_1, x_2) = (x_1^2 + x_2^2) T_n^2(x_2(x_1^2 + x_2^2)^{-1/2})$$
$$g_{1,n-1}(x_1, x_2) = x_1(x_1^2 + x_2^2)^{\frac{n-1}{2}} U_{n-1}(x_2(x_1^2 + x_2^2)^{-1/2})$$

entonces, si tomamos $\mathbf{n} = (n_1, \dots, n_d)$ con $n_1 = \{0, 1\}$ se define

$$Y_{\mathbf{n}} = g_{n_1, n_2}(x_1, x_2) \prod_{j=3}^d (x_1^2 + \dots + x_j^2)^{n_j/2} C_{n_j, \lambda_j}(x_j(x_1^2 + \dots + x_j^2)^{-1/2})$$

donde $\lambda_j = \lambda_j(n_1, \dots, n_{j-1}) = \sum_{i=1}^{j-1} n_i + \frac{j-2}{2}$. Entonces $\{Y_{\mathbf{n}}, |\mathbf{n}| = n\}$ es una base de \mathbb{Y}_n^d .

Tomamos,

$$F_n^\lambda(x) = (x_1^2 + \dots + x_d^2)^{\frac{n}{2}} C_{n, \lambda} \left(\frac{x_d}{\sqrt{(x_1^2 + \dots + x_d^2)}} \right)$$

. Si $x = (x_1, \dots, x_d), n = (n_1, \dots, n_d)$ y $x' = (x_1, \dots, x_{d-1}), n = (n_1, \dots, n_{d-1})$.
 $Y_n(x) = Y_{n'}(x') F_{n_d}^{\lambda_d}(x)$ siendo $Y_{n'}(x')$ un esférico armónico de dimensión $d-1$ y grado $n - n_d$.

Proposición 2.2. Para $i = 1, \dots, d-1$

$$\begin{aligned}\partial_i F_n^\lambda(x) &= -2\lambda x_i F_{n-2}^{\lambda+1}(x) \\ \partial_d F_n^\lambda(x) &= (n+2\lambda-1)F_{n-1}^\lambda(x)\end{aligned}$$

Demostración. Sean $r = \sqrt{x_1^2 + \dots + x_d^2}$, $i = 1, 2, \dots, d-1$. Usando que

$$\frac{d}{dx} C_{n,\lambda}(x) = 2\lambda C_{n-1,\lambda+1}(x)$$

y los apartados (I) y (II) de la Proposición D.5 entonces

$$\begin{aligned}\partial_i F_n^\lambda &= x_i r^{n-2} \left[n C_{n,\lambda}\left(\frac{x_d}{r}\right) - 2\lambda \frac{x_d}{r} C_{n+1,\lambda+1}\left(\frac{x_d}{r}\right) \right] \\ &= -2\lambda x_i r^{n-2} C_{n-2,\lambda+1}\left(\frac{x_d}{r}\right)\end{aligned}$$

Además,

$$\begin{aligned}\partial_d F_n^\lambda(x) &= r^{n-1} \left[n \frac{x_d}{r} C_{n,\lambda}\left(\frac{x_d}{r}\right) + 2\lambda \left(1 - \frac{x_d^2}{r^2}\right) C_{n-1,\lambda+1}\left(\frac{x_d}{r}\right) \right] \\ &= (n+2\lambda-1) r^{n-1} C_{n-1,\lambda}\left(\frac{x_d}{r}\right)\end{aligned}$$

□

Ahora, tomamos la proyección del espacio de los polinomios homogéneos al espacio de los armónicos esféricos $proj_{n,\mathbb{S}^{d-1}}^d : \mathbb{H}_n^d \rightarrow \mathbb{Y}_n^d$ tal que para $P \in \mathbb{H}_n^d$ está definida por

$$proj_n^d P = \sum_{j=0}^{\lfloor \frac{n}{2} \rfloor} \frac{1}{4^j j! (-n+2-\frac{d}{2})} \|x\|^{2j} \triangle^j P$$

lo que implica que para Y_n se tiene

$$proj_{n,\mathbb{S}^{d-1}}^d(x_i Y_n(x)) = x_i Y_n(x) - \frac{1}{2(n+(d-2)/2)} \|x\|^2 \partial_i Y_n(x)$$

Proposición 2.3. Sea $n' = |n'| = n - n_d$ e $i = 1, 2, \dots, d-1$

$$\begin{aligned}\partial_i Y_n(x) &= -2\lambda_d proj_{n'+1,\mathbb{S}^{d-1}}^{d-1}(x Y_{n'}(x')) F_{n_d-2}^{\lambda_d+1}(x) \\ &\quad + \frac{(n_d+2\lambda_d-1)(n_d+2\lambda_d-2)}{(2\lambda_d-1)(2\lambda_d-2)} \partial_i Y_{n'}(x') F_{n_d}^{\lambda_d-1}(x) \\ \partial_d Y_n(x) &= (n_d+2\lambda_d-1) Y_{n'}(x') F_{n_d-1}^{\lambda_d}(x)\end{aligned}$$

Demostración. Usando los resultados anteriores y que $2\lambda_d - 1 = 2n' + d - 3$

$$\begin{aligned}\partial_i Y(x) &= \partial_i Y_{n'}(x') F_{n_d}^{\lambda_d}(x) - 2\lambda_d x_i Y_{n'}(x') F_{n_d-2}^{\lambda_d+1}(x) \\ &= -2\lambda_d \text{proj}_{n'+1, \mathbb{S}^{d-1}}^{d-1}(x_i Y_{n'}(x')) F_{n_d-2}^{\lambda_d+1}(x) \\ &\quad + \partial_i Y_{n'}(x') \left[F_{n_d}^{\lambda_d}(x) - \frac{2\lambda_d}{2\lambda_d - 1} \|x'\|^2 F_{n_d-2}^{\lambda_d+1}(x) \right]\end{aligned}$$

Además, como $\|x'\|^2 = r^2 - x_d^2$ y en virtud de los apartados (I) y (IV) de la Proposición D.5

$$\begin{aligned}F_{n_d}^{\lambda_d}(x) - \frac{2\lambda_d}{2\lambda_d - 1} \|x'\|^2 F_{n_d-2}^{\lambda_d+1}(x) \\ &= r_d^n \left[C_{n_d}^{\lambda_d} \left(\frac{x_d}{r} \right) - \frac{2\lambda_d}{2\lambda_d - 1} \left(1 - \frac{x_d}{r^2} \right) C_{n_d-2}^{\lambda_d+1} \left(\frac{x_d}{r} \right) \right] \\ &= \frac{(n_d + 2\lambda_d - 1)(n_d + 2\lambda_d - 2)}{(2\lambda_d - 1)(2\lambda_d - 2)} r^{n_d} C_{n_d}^{\lambda_d-1} \left(\frac{x_d}{r} \right)\end{aligned}$$

Sustituyendo esta igualdad en la obtenida anteriormente, se prueba el resultado. \square

Proposición 2.4. Sea $n' = |\mathbf{n}'| = n - n_d$ e $i = 1, \dots, d - 1$

$$\begin{aligned}\text{proj}_{n+1, \mathbb{S}^{d-1}}^d(x_i Y_n(x)) &= \frac{\lambda_d}{n_d + \lambda_d} \text{proj}_{n'+1, \mathbb{S}^{d-1}}^{d+1}(x_i Y_{n'}(x')) F_{n_d}^{\lambda_d+1}(x) \\ &\quad + \frac{(n_d + 1)(n_d + 2)}{(2\lambda_d - 1)(2\lambda_d - 2)2(n_d + \lambda_d)} \partial_i Y_{n'}(x') F_{n_d+2}^{\lambda_d-1}(x) \\ \text{proj}_{n+1, \mathbb{S}^{d-1}}^d(x_d Y_n(x)) &= \frac{n_d + 1}{2(n_d + \lambda_d)} Y_{n'}(x') F_{n_d+1}^{\lambda_d}(x)\end{aligned}$$

Demostración.

$$\text{proj}_{n+1, \mathbb{S}^{d-1}}^d(x_i Y_n(x)) = x_i Y_{n'} F_{n_d}^{\lambda_d}(x) - \frac{r^2}{2(n_d + \lambda_d)} \partial_i (Y_{n'}(x') F_{n_d}^{\lambda_d}(x))$$

Usando la Proposición 2.3 tenemos que

$$\begin{aligned}\text{proj}_{n+1, \mathbb{S}^{d-1}}^d(x_i Y_n(x)) &= \\ \text{proj}_{n'+1, \mathbb{S}^{d-1}}^{d-1}(x_i Y_{n'}(x')) &\left[F_{n_d}^{\lambda_d} + \frac{\lambda_d}{n_d + \lambda_d} r^2 F_{n_d-2, \lambda_d+1}(x) \right] \\ + \partial_i Y_{n'}(x') &\left[\frac{\|x'\|^2}{2\lambda_d - 1} F_{n_d}^{\lambda_d}(x) - \frac{(n_d + 2\lambda_d - 1)(n_d + 2\lambda_d) - 2}{2(n_d + \lambda_d)(2\lambda_d - 1)(2\lambda_d - 2)} r^2 F_{n_d}^{\lambda_d-1}(x) \right]\end{aligned}$$

Ahora, aplicando el apartado (IV) de la Proposición D.5

$$\begin{aligned}F_{n_d}^{\lambda_d} + \frac{\lambda_d}{n_d + \lambda_d} r^2 F_{n_d-2, \lambda_d+1}(x) &= r^{n_d} \left[C_{n_d}^{\lambda_d} \left(\frac{x_d}{r} \right) + \frac{\lambda_d}{n_d + \lambda_d} C_{n_d-2}^{\lambda_d+1} \left(\frac{x_d}{r} \right) \right] \\ &= \frac{\lambda_d}{n_d + \lambda_d} r^{n_d} C_{n_d}^{\lambda_d+1} \left(\frac{x_d}{r} \right)\end{aligned}$$

Por otro lado, aplicando el apartado (I) de la Proposición D.5

$$\begin{aligned}
& \frac{\|x'\|^2}{2\lambda_d - 1} F_{n_d}^{\lambda_d}(x) - \frac{(n_d + 2\lambda_d - 1)(n_d + 2\lambda_d) - 2}{2(n_d + \lambda_d)(2\lambda_d - 1)(2\lambda_d - 2)} r^2 F_{n_d}^{\lambda_d-1}(x) \\
&= \frac{r^{n_d+2}}{2\lambda_d - 1} \left[\left(1 - \frac{x_d^2}{r^2}\right) C_{n_d}^{\lambda_d}\left(\frac{x_d}{r}\right) - \frac{(n_d + 2\lambda_d - 1)(n_d + 2\lambda_d - 2)}{2(n_d + \lambda_d)(2\lambda_d - 2)} C_{n_d}^{\lambda_d-1}\left(\frac{x_d}{r}\right) \right] \\
&= \frac{(n_d + 1)(n_d + 2)}{2(n_d + \lambda_d)(2\lambda_d - 1)(2\lambda_d - 2)} r^{n_d+2} C_{n_d+2}^{\lambda_d-1}\left(\frac{x_d}{r}\right)
\end{aligned}$$

Uniendo ambas igualdades se prueba la primera igualdad de la proposición. Finalmente,

$$\begin{aligned}
\text{proj}_{n+1, \mathbb{S}^{d-1}}^d(x_d Y_n(x)) &= x_d Y_{n'}(x') F_{n_d}^{\lambda_d}(x) - \frac{r^2}{2(n_d + \lambda_d)} \partial_d(Y_{n'}(x') F_{n_d}^{\lambda_d}(x)) \\
&= Y_{n'}(x') \left[x_d F_{n_d}^{\lambda_d}(x) - \frac{r^2}{2(n_d + \lambda_d)} (n_d + 2\lambda_d - 1) F_{n_d-1}^{\lambda_d}(x) \right] \\
&= \frac{n_d + 1}{2(n_d + \lambda_d)} Y_{n'}(x') F_{n_d+1}^{\lambda_d}(x)
\end{aligned}$$

□

Teorema 2.5. Sea $n = (n_1, n_2, \dots, n_d) \in \mathbb{N}_0^d$ con $n_1 = \{0, 1\}$ y $|n| = n$. Entonces $\partial_i Y_n(x)$ es un armónico esférico de grado $n-1$

$$(\partial_i Y_n, Y_m)_{L^2} \neq 0 \quad |m| = n - 1$$

para sólo 2^{d-2} índices, $m \in \mathbb{N}_0^d$ con $m_1 = \{0, 1\}$

Demostración. La afirmación del teorema equivale a $\partial_i Y_n = \sum_m a_m Y_m^{n-1}$ siendo a_m una constante real. El resultado se prueba por inducción sobre la dimensión d usando las proposiciones anteriores. Para $d = 2$,

$$\begin{aligned}
\partial_1 Y_n^{(1)}(x) &= n Y_{n-1}^{(1)}(x) & \partial_2 Y_n^{(1)}(x) &= -n Y_{n-1}^{(2)}(x) \\
\partial_1 Y_n^{(2)}(x) &= n Y_{n-1}^{(2)}(x) & \partial_2 Y_n^{(2)}(x) &= n Y_{n-1}^{(1)}(x)
\end{aligned}$$

Supongamos cierto el resultado para dimensión $d - 1$. Entonces $\partial_i Y_{n'}(x')$ puede escribirse como combinación lineal de a lo sumo 2^{d-3} esféricos $Y_m^{n'-1}$. Como $Y_m^{n'-1} F_{n_d}^{\lambda_d-1} = Y_{m_1, \dots, m_{d-1}, n_d}^{n-1}$, el resultado se obtiene aplicando la Proposición 2.2. □

2.1.1. Caso particular $d=3$

El espacio de los armónicos esféricos de grado n en dimensión 3 tiene dimensión $2n+1$. Tomando coordenadas esféricas,

$$\begin{aligned}x_1 &= r \sen \theta \sen \phi \\x_2 &= r \sen \theta \cos \phi \\x_3 &= r \cos \theta \\0 &\leq \theta \leq \pi, 0 \leq \phi \leq 2\pi, r > 0\end{aligned}$$

una base ortogonal de \mathbb{Y}_n^d viene dada por

$$\begin{cases} Y_{k,1}^n = r^n (\sen \theta)^k C_{n-k,k+1/2}(\cos \theta) \cos(k\phi), & 0 \leq k \leq n \\ Y_{k,2}^n(x) = r^{n-k} (\sen \theta)^k C_{n-k,k+1/2}(\cos \theta) \sen(k\phi), & 1 \leq k \leq n \end{cases} \quad (2.1.1)$$

Proposición 2.6. Para $k = 0, \dots, n$

$$\begin{aligned}\partial_1 Y_{k,1}^n(x) &= -\frac{(n+k)(n+k-1)}{2(2k-1)} Y_{k-1,2}^{n-1}(x) - (k + \frac{1}{2}) Y_{k+1,2}^{n-1}(x) \\ \partial_2 Y_{k,1}^n(x) &= \frac{(n+k)(n+k-1)}{2(2k-1)} Y_{k-1,1}^{n-1}(x) - (k + \frac{1}{2}) Y_{k+1,1}^{n-1}(x) \\ \partial_3 Y_{k,1}^n(x) &= (n+k) Y_{k,1}^{n-1}(x)\end{aligned}$$

Para $k = 1, \dots, n$

$$\begin{aligned}\partial_1 Y_{k,2}^n(x) &= \frac{(n+k)(n+k-1)}{2(2k-1)} Y_{k-1,1}^{n-1}(x) + (k + \frac{1}{2}) Y_{k+1,1}^{n-1}(x) \\ \partial_2 Y_{k,2}^n(x) &= \frac{(n+k)(n+k-1)}{2(2k-1)} Y_{k-1,2}^{n-1}(x) - (k + \frac{1}{2}) Y_{k+1,2}^{n-1}(x) \\ \partial_3 Y_{k,2}^n(x) &= (n+k) Y_{k,2}^{n-1}(x)\end{aligned}$$

2.2. Puntos críticos del gradiente.

Finalmente, queremos conocer el número de puntos que anulan el gradiente, para ello haremos uso de las siguientes igualdades trigonométricas

$$\begin{aligned}\cos(k\phi) &= \sen\left(k\phi + \frac{\pi}{2}\right) \\ \sen(k\phi) &= \cos\left(k\phi - \frac{\pi}{2}\right) \\ \cos(k+1)\phi &= \cos(k\phi) \cos \phi - \sen k\phi \sen \phi \\ \cos(k-1)\phi &= \cos(k\phi) \cos \phi + \sen k\phi \sen \phi \\ \sen(k+1)\phi &= \sen(k\phi) \cos \phi + \sen(\phi) \cos(k\phi) \\ \sen(k-1)\phi &= \sen(k\phi) \cos \phi - \sen(\phi) \cos(k\phi)\end{aligned}$$

Llamaremos $c_{n,k}, d_k$ a las constantes $\frac{(n+k)(n+k-1)}{2(2k-1)}, k + \frac{1}{2}$ respectivamente. Igualando a 0 las parciales calculadas anteriormente tenemos que para $k \geq 0$

$$\partial_3 Y_{k,1}^n(x) = (n+k)Y_{k,1}^{n-1}(x) = (n+k)(\sin \theta)^k C_{n-k-1,k+1/2}(\cos \theta) \cos k\phi = 0 \quad (2.2.1)$$

implica que ha de verificarse una de las siguientes igualdades

$$\begin{cases} \sin \theta = 0 \\ \cos k\phi = 0 \\ C_{n-k-1,k+1/2}(\cos \theta) = 0 \end{cases} \quad (2.2.2)$$

Si $\sin \theta = 0$ tendremos que $\theta = 0$ o $\theta = \pi$.

Ahora, suponemos que $\cos k\phi = 0$, entonces

$$\begin{aligned} \partial_1 Y_{k,1}^n(x) &= (\sin \theta)^{k-1} [-c_{n,k} C_{n-k,k-1/2}(\cos \theta) \sin(k\phi) \cos \phi \\ &\quad + d_k \sin^2 \theta C_{n-k-2,k+3/2}(\cos \theta) \sin(k\phi) \cos \phi] \\ &= (\sin \theta)^{k-1} \sin(k\phi) \cos \phi [-c_{n,k} C_{n-k,k-1/2}(\cos \theta) \\ &\quad + d_k \sin^2 \theta C_{n-k-2,k+3/2}(\cos \theta)] \end{aligned} \quad (2.2.3)$$

$$\begin{aligned} \partial_2 Y_{k,1}^n(x) &= (\sin \theta)^{k-1} [-c_{n,k} C_{n-k,k-1/2}(\cos \theta) \sin(k\phi) \sin \phi \\ &\quad + d_k \sin^2 \theta C_{n-k-2,k+3/2}(\cos \theta) \sin(k\phi) \sin \phi] \\ &= (\sin \theta)^{k-1} \sin(k\phi) \sin \phi [-c_{n,k} C_{n-k,k-1/2}(\cos \theta) \\ &\quad + d_k \sin^2 \theta C_{n-k-2,k+3/2}(\cos \theta)] \end{aligned} \quad (2.2.4)$$

Ahora, igualando ambas expresiones a 0

$$\begin{aligned} \partial_1 Y_{k,1}^n(x) &= \partial_2 Y_{k,1}^n(x) \\ &= (\sin \theta)^{k-1} \sin(k\phi) \cos \phi [c_{n,k} C_{n-k,k-1/2}(\cos \theta) \\ &\quad + d_k \sin^2 \theta C_{n-k-2,k+3/2}(\cos \theta)] \\ &= (\sin \theta)^{k-1} \sin(k\phi) \sin \phi [c_{n,k} C_{n-k,k-1/2}(\cos \theta) \\ &\quad + d_k \sin^2 \theta C_{n-k-2,k+3/2}(\cos \theta)] \\ &= 0 \end{aligned} \quad (2.2.5)$$

Como $\cos(k\phi) = 0$ entonces $\sin(k\phi) \neq 0$. Además, $\sin \phi$ y $\cos \phi$ no se anulan simultáneamente, luego de la expresión anterior se verifica que

$$c_{n,k} C_{n-k}^{k-1/2}(\cos \theta) + d_k \sin^2 \theta C_{n-k-2}^{k+3/2}(\cos \theta) = 0$$

Haciendo el cambio $\sin^2 \theta = 1 - \cos^2 \theta$, y tomando como variable $t = \cos \theta$, se tiene que la expresión anterior es un polinomio de grado a lo sumo $n - k$. Por tanto, tiene a lo sumo $n - k$ raíces.

Consideremos el polinomio:

$$Q_{n,k}(t) = (n-k)(n+k-1)C_{n-k,k-\frac{1}{2}}(t) + (2k-1)(1-t^2)C_{n-k-2,k+\frac{3}{2}}(t)$$

De la ecuación (I) de la Proposición D.5 deducimos que:

$$(2k-1)(2k+1)C_{n-k-2,k+\frac{3}{2}}(t) = \left(C_{n-k,k-\frac{1}{2}}\right)''(t)$$

De la ecuación diferencial de los polinomios de Gegenbauer (Proposición D.4) obtenemos:

$$(1-t^2)\left(C_{n-k,k-\frac{1}{2}}\right)''(t) = 2kt\left(C_{n-k,k-\frac{1}{2}}\right)'(t) - (n-k)(n-k-1)C_{n-k,k-\frac{1}{2}}(t)$$

Por tanto,

$$\begin{aligned} Q_{n,k}(t) &= [(n+k)(n+k-1) - (n-k)(n+k-1)]C_{n-k,k-\frac{1}{2}}(t) \\ &\quad + 2kt\left(C_{n-k,k-\frac{1}{2}}\right)'(t) \\ &= 2k\left[(n+k-1)C_{n-k,k-\frac{1}{2}}(t) + t\left(C_{n-k,k-\frac{1}{2}}\right)'(t)\right] \end{aligned}$$

y usando de (III) en la Proposición D.5 queda

$$Q_{n-k}(t) = 2k\left(C_{n-k+1,k-\frac{1}{2}}\right)'(t) = 2k(2k-1)\left(C_{n-k,k+\frac{1}{2}}\right)$$

que tiene exactamente $n-k$ raíces reales y distintas contenidas en el intervalo $[1, -1]$.

En resumen, hemos encontrado $2k(n-k) + 2$ valores que anulan las 3 parciales.

Razonando análogamente, se obtiene la misma conclusión para las parciales de los armónicos esféricos del segundo tipo.

A continuación, se muestran algunos ejemplos de los puntos críticos obtenidos para distintos valores de n y k . El código programado para obtener esta visualización se puede consultar en el Apéndice G.

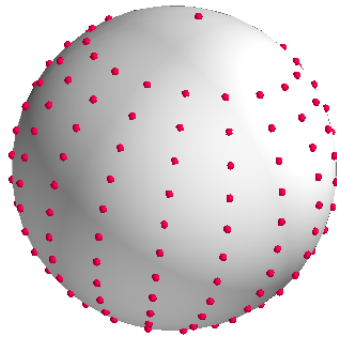


Figura 2.2.1: Puntos críticos del gradiente para $n=20$ y $k=9$.

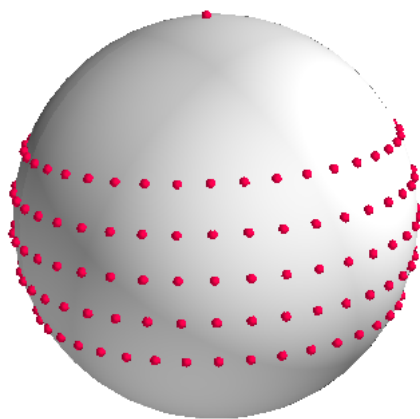


Figura 2.2.2: Puntos críticos del gradiente para $n=25$ y $k=20$.

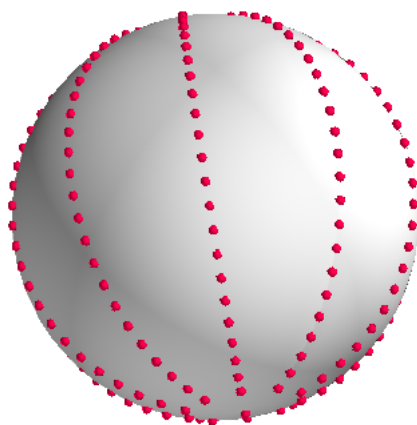


Figura 2.2.3: Puntos críticos del gradiente para $n=30$ y $k=5$.

Capítulo 3

Integración Numérica

En este capítulo vamos a obtener una aproximación numérica de la integral

$$I(f) = \int_{\mathbb{S}^2} f(\eta) dS^2(\eta).$$

3.1. Fórmulas de una variable.

En primer lugar tomaremos la siguientes coordenadas esféricas

$$\eta \mapsto (\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta), \quad 0 \leq \phi \leq 2\pi, 0 \leq \theta \leq \pi$$

Ahora,

$$I(f) = \int_0^{2\pi} \int_0^\pi f(\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta) \sin \theta d\theta d\phi.$$

Una vez hemos simplificado la expresión de la integral podemos aplicar métodos de integración numérica de una variable a cada una de las integrales. Comenzaremos integrando respecto a ϕ .

Nota 3.1. Regla del trapecio compuesta.

$$\int_a^b f(x) dx \approx \frac{b-a}{n} \left[\frac{f(a) + f(b)}{2} + \sum_{k=1}^{n-1} f\left(a + k \frac{b-a}{n}\right) \right].$$

Como el integrando es periódico en π con periodo 2π , usando la regla del trapecio tenemos que

$$\tilde{I}(g) \equiv \int_0^{2\pi} g(\phi) d\phi \approx \widetilde{I}_m(g) \equiv \frac{2\pi}{m} \sum_{j=1}^m g\left(j \frac{2\pi}{m}\right)$$

Lema 3.2. Para $m \geq 2, k \geq 0$

$$\begin{aligned}\int_0^{2\pi} \cos(k\phi) d\phi &= \begin{cases} 2\pi, & k = 0 \\ 0, & k > 0 \end{cases} \\ \frac{2\pi}{m} \sum_{j=0}^{m-1} \cos(k \frac{2j\pi}{m}) &= \begin{cases} 2\pi, & k = 0, m, 2m, \dots \\ 0, & \text{en otro caso} \end{cases} \\ \int_0^{2\pi} \sin(k\phi) d\phi &= \frac{2\pi}{m} \sum_{j=1}^{m-1} \sin(k \frac{2j\pi}{m}) = 0\end{aligned}$$

Demostración. Basta con sustituir las siguientes expresiones en la fórmula del trapecio

$$\begin{aligned}\cos(iw) &= \frac{e^{iw} + e^{-iw}}{2} \\ \sin(iw) &= \frac{e^{iw} - e^{-iw}}{2i}\end{aligned}$$

□

Finalmente estudiaremos la convergencia de $\tilde{I}(g)$ a $I(g)$. Para estudiar la convergencia de funciones periódicas, introduciremos el espacio $H^q(2\pi)$ como aquel de las funciones de cuadrado integrable en $(0, 2\pi)$ que verifican que

$$\|f\|_q = \sqrt{|a_0|^2 + \sum |k| |a_k|^2} < +\infty$$

siendo a_k los coeficientes de la serie de Fourier. El espacio $H^q(2\pi)$ es un espacio de Hilbert con el producto escalar

$$(f, g)_q = a_0 b_0 + \sum_{k=1}^{\infty} |k|^{2q} a_k b_k$$

siendo a_k, b_k los coeficientes de la serie de Fourier para f y g respectivamente.

Teorema 3.3. Sean $q > \frac{1}{2}, g \in H^q(2\pi)$, entonces

$$|I(g) - I_m(g)| \leq \frac{\sqrt{4\pi\zeta(2q)}}{m^q} \|g\|_q, \quad m \geq 1$$

siendo ζ la función zeta de Riemann,

$$\zeta(s) = \sum_{j=1}^{\infty} \frac{1}{j^s}$$

Por otro lado, estudiamos el valor de la integral

$$\int_0^{2\pi} f(\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta) \sin \theta d\theta$$

. Para ello, hacemos el cambio de variable $z = \cos \theta$, la integral queda:

$$\int_{-1}^1 f(\cos \phi \sqrt{1-z^2}, \sin \phi \sqrt{1-z^2}, z) dz$$

Nota 3.4. Integración de Gauss-Legendre

$$\int_{-1}^1 f(x) dx \approx \sum_{i=1}^n w_i f(x_i)$$

Aplicamos la integración de Gauss-Legendre en $-1 < z < 1$.

$$\begin{aligned} I_n(f) &= h \sum_{j=0}^{2n-1} \sum_{k=1}^n w_k f(\cos \phi_j \sqrt{1-z^2}, \sin \phi_j \sqrt{1-z^2}, z) \\ &= h \sum_{j=0}^{2n-1} \sum_{k=1}^n w_k f(\cos \phi_j \sin \theta_k, \sin \phi_j \sin \theta_k, \cos \theta_k) \end{aligned} \quad (3.1.1)$$

siendo z_i, w_i los nodos y los pesos de la fórmula de Gauss-Legendre respectivamente.

Teorema 3.5. *Sea f un polinomio esférico de grado menor o igual a $2n-1$. Entonces $I(f) = I_n(f)$. Además, para $f(x, y, z) = z^{2n}, I(f) \neq I_n(f)$*

Demostración. Supongamos $f(x, y, z) = x^r y^s z^t$, $r + s + t \leq 2n - 1$. Haciendo el cambio a coordenadas esféricas:

$$I = \int_{\mathbb{S}^2} x^r y^s z^t dS^2 = \int_0^{2\pi} \int_0^\pi \cos^r \phi \sin^{r+s+1} \theta \sin^s \phi \cos^t \theta d\phi d\theta$$

Sean

$$\begin{aligned} I &= \int_{\mathbb{S}^2} x^r y^s z^t dU = J^{r,s} K^{r,s,t} \\ J^{r,s} &= \int_0^{2\pi} \cos^r \phi \sin^s \phi d\phi \\ K^{r,s,t} &= \int_0^\pi \sin^{r+s+1} \theta \cos^t \theta d\theta \end{aligned}$$

Para la correspondiente integral numérica tenemos y la correspondiente aproximación numérica

$$\begin{aligned} I_m &= \sum_{j=1}^{2n} \sum_{k=1}^n w_k x_{j,k}^r y_{j,k}^s z_k^t = J^{r,s} K^{r,s,t} \\ J_n^{r,s} &= h \sum_{j=1}^{2n} \cos^r \phi_j \sin^s \phi_j \\ K_n^{r,s,t} &= \sum_{k=1}^n w_k \sin^{r+s+1} \theta_k \cos^t \theta_k \end{aligned} \quad (3.1.2)$$

Los valores $\{x_{j,k}, y_{j,k}, z_k\}$ representan las coordenadas cartesianas de los puntos obtenidos mediante las coordenadas esféricas $\{\phi_j\}$ y $\{\theta_j\}$.

Ahora, analizaremos el error,

$$E_n = I - I_n = J^{r,s} K^{r,s,t} - J_n^{r,s} K_n^{r,s,t}$$

Usaremos las siguientes propiedades trigonométricas, suponiendo que r es impar y s es par

$$\begin{aligned}\cos^r(\pi + w) &= \cos^r(\pi - w) \\ \sen^s(\pi + w) &= \sen^s(w) \\ \sen^s\left(\frac{\pi}{2} + w\right) &= \sen^s\left(\frac{\pi}{2} - w\right) \\ \cos^r\left(\frac{\pi}{2} + w\right) &= -\cos^r\left(\frac{\pi}{2} - w\right)\end{aligned}$$

De estas igualdades se deduce que

$$J^{r,s} = J_n^{r,s} = 0$$

Razonando análogamente se obtiene la misma igualdad cuando r es par y s impar o cuando ambos son impares. En consecuencia $I - I_n = 0$.

En [1, p.170] se demuestra que si $f(x, y, z) = z^{2n}$ entonces $I(f) \neq I_n(f)$. \square

Finalmente, obtendremos una cota del error. Para ello haremos uso del Teorema 4.5 [1, p.142] y del corolario del Teorema 4.11 [1, p.149].

El error minimax de la aproximación de una función $f \in C(\mathbb{S}^2)$ por un polinomio esférico de grado menor o igual a m , se define como

$$E_{m,\infty}(f) = \min \|f - p\|_\infty$$

Sea p_m^* un polinomio esférico de grado menor o igual a m cuyo error minimax se alcanza. La existencia de p_m^* es consecuencia de los resultados citados anteriormente.

De (3.1.1) se deduce que $I(p_{2n-1}^*) = I_n(p_{2n-1}^*)$ Ahora, para $g \in C(\mathbb{S}^2)$

$$|I(g)| \leq 4\pi \|g\|_\infty \quad (3.1.3)$$

$$|I_n(g)| \leq 4\pi \|g\|_\infty \quad (3.1.4)$$

y

$$I(f) - I_n(f) = I(f - p_{2n-1}^*) - I_n(f - p_{2n-1}^*) \quad (3.1.5)$$

$$|I(f) - I_n(f)| \leq |I(f - p_{2n-1}^*)| + |I_n(f - p_{2n-1}^*)| \leq 8\pi \|f - p_{2n-1}^*\|_\infty \quad (3.1.6)$$

3.2. Métodos de Gauss de Orden Superior.

En el caso de integración en una variable los métodos gaussianos se basan en pedir que la fórmula sea exacta para polinomios del mayor grado posible. Si tenemos n nodos es posible alcanzar esa exactitud para polinomios de grado $2n - 1$. Este enfoque se generaliza para la integración en varias variables.

Sea

$$I(f) = \int f(\eta) dS^2(\eta) \approx I_N(f) = \sum_{k=1}^N w_k f(\eta_k)$$

Los nodos n_k , y los pesos w_k se eligen de forma que la fórmula sea exacta para los armónicos esféricos de mayor grado posible.

Teorema 3.6. *Sea \mathcal{G} un grupo finito de rotaciones sobre la esfera. Y supongamos que el sistema anterior es invariante para todos los elementos de \mathcal{G} , es decir, $I_N(f_\gamma) = I_N(f)$, $\forall \gamma \in \mathcal{G}$. Entonces*

$$\{\eta_i : i = 1, \dots, N\} = \{\gamma(\eta_i) : i = 1, \dots, N\}$$

Además, para cada nodo η_i sea $\mathcal{S}_i = \{\gamma(\eta_i) : \gamma \in \mathcal{G}\}$. Entonces los pesos asociados a los nodos son iguales. Es condición necesaria y suficiente que f sea invariante para todo elemento de \mathcal{G} para que se verifique $I(f) = I_N(f)$. En tal caso, decimos que $I_N(f)$ tiene grado de precisión d , siendo d la dimensión de f .

Demostración. Como la esfera es simétrica por rotaciones se verifica que

$$I(f) = \int_{\mathbb{S}^2} f(\eta) dS^2(\eta) = \int_{\mathbb{S}^2} f_\gamma(\eta) dS^2(\eta) = I(f_\gamma)$$

. Sea $f^* = \frac{1}{|\mathcal{G}|} \sum_{\gamma \in \mathcal{G}} f_\gamma$. f^* es invariante por \mathcal{G} e $I(f) = I(f^*)$. Por hipótesis del teorema,

$$I(f) - I_N(f) = I(f^*) - I_N(f^*)$$

Por tanto, si queremos que $I(f) = I_N(f)$ para todo polinomio esférico f , basta con mostrar que el resultado es cierto para todo polinomio esférico que sea invariante por la acción de \mathcal{G} . \square

Razonando análogamente al apartado anterior se prueba que

$$|I(f) - I_N(d)| \leq 8\pi E_d(f)$$

y la velocidad con la que $I_N(f)$ converge a $I(f)$ depende de la regularidad de f .

Los nodos y los pesos han de ser elegidos de forma que sean invariantes por \mathcal{G} . Además, el error $I(f) - I_N(d)$ debe ser 0 para polinomios del mayor grado posible, lo que requiere la resolución de un sistema de ecuaciones no lineal. Estas condiciones hacen que la elección de los nodos y los pesos sea una tarea complicada.

3.3. Integración puntos dispersos

Supongamos que tenemos N nodos, $P = \{\eta_1, \dots, \eta_N\}$ y sus valores aproximados $f_i \approx f(\eta_i)$. Queremos aproximar la integral $I(f) = \int_{\mathbb{S}^2} f(\eta) dS^2(\eta)$.

Tomamos $T_N = \{\triangle_1, \dots, \triangle_{M(N)}\}$ la triangulación de \mathbb{S}^2 , donde los vértices de la triangulación son los nodos.

$$I(f) = \sum_{k=1}^M \int_{\triangle_k} f(n) dS^2(n) \approx \sum_{k=1}^M \frac{1}{3} [f(n_{k,1}) + f(n_{k,2}) + f(n_{k,3})] \text{area}(\triangle_k)$$

Realizando un análisis del error cometido similar al realizado anteriormente, obtenemos que el método tiene grado de precisión 0 y $|I(f) - I_n(f)| \leq 4\pi c_f h$ siendo $h = \max \text{diam}(\triangle)$, $\triangle \in T_N$. Luego, f es lipschitziana en \mathbb{S}^2 con constante c_f .

Finalmente, se nos plantean dos cuestiones: elegir una triangulación y un conjunto de nodos. Una de las opciones más frecuentes para la primera cuestión es la triangulación de Delaunay. En cuanto al conjunto de nodos, es conocido que se obtienen buenos resultados tomando un conjunto en el que los puntos están bien distribuidos.

3.4. Integración sobre el disco unidad.

Finalmente, integraremos sobre el disco unidad $\mathbb{D} = \{(x, y) : x^2 + y^2 \leq 1\}$. La semiesfera superior es la imagen de $z = \sqrt{1 - x^2 - y^2}$ $(x, y) \in \mathbb{D}$

$$\begin{aligned} & \int_D f(x, y, \sqrt{1 - x^2 - y^2}) \sqrt{1 + \left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2} dx dy \\ &= \int_D f(x, y, \sqrt{1 - x^2 - y^2}) \frac{dx dy}{\sqrt{1 - x^2 - y^2}} \end{aligned}$$

Por tanto,

$$\begin{aligned} & \int_{\mathbb{S}^2} f(\eta) dS^2(\eta) = \\ & \int_D \left[f(x, y, \sqrt{1 - x^2 - y^2}) + f(x, y, -\sqrt{1 - x^2 - y^2}) \right] \frac{dx dy}{\sqrt{1 - x^2 - y^2}} \end{aligned}$$

Es decir, la integración sobre la esfera es equivalente a una integración con pesos sobre el disco unidad.

$$I(f) = \int_D f(x, y) dx dy = \int_0^{2\pi} \int_0^1 r f(r \cos \theta, r \sin \theta) dr d\theta$$

Para integrar sobre θ usamos la regla del trapecio y para hacerlo respecto de r usamos la integración de Gauss-Legendre al integrando.

$$I_n(f) = h \sum_{j=0}^{2n} \sum_{k=0}^n w_k r_k f(r_k \cos \theta_j, r_k \sin \theta_j)$$

Teorema 3.7. *Sea $f(x,y)$ un polinomio de grado menor o igual a $2n$, entonces $I(f) = I_n(f)$. Además, la fórmula anterior tiene exactitud $2n$.*

Demostración. Supongamos $f(x,y) = x^\alpha y^\beta$ con α, β enteros positivos y tales que $\alpha + \beta \leq n$, entonces

$$I(f) = \left(\int_0^{2\pi} (\cos \theta)^\alpha (\sin \theta)^\beta d\theta \right) \left(\int_0^1 r^{\alpha+\beta+1} \right) \equiv J^{\alpha,\beta} K^{\alpha+\beta+1}$$

$$I_n(f) = \left(h \sum_{j=0}^{2n} (\cos \theta_j)^\alpha (\sin \theta_j)^\beta \right) \left(\sum_{k=0}^n w_k r_k^{\alpha+\beta+1} \right) \equiv J_n^{\alpha,\beta} K_n^{\alpha+\beta+1}$$

siendo

$$K^t = \int_0^1 r^t \quad K_n^t = \sum_{k=0}^n w_k r_k^t$$

Por otro lado, si β es impar las integrales $J^{\alpha,\beta}, J_n^{\alpha,\beta} = 0$ (visto en la demostración del Teorema 3.5). Si β es par podemos transformar el integrando $(\cos \theta)^\alpha (\sin \theta)^\beta$ en un polinomio de potencias de $\cos \theta$ con grado $\alpha + \beta$. Usando el Lema 3.2 se tiene que $J_n^{\alpha,\beta} = J^{\alpha,\beta}, \alpha + \beta \leq 2n$. La fórmula de Gauss-Legendre para $(n+1)$ puntos tiene exactitud $2n+1$, luego $K_n^{\alpha+\beta+1} = K^{\alpha+\beta+1}, \alpha + \beta \leq 2n$.

Por tanto, hemos probado que $I(f) = I_n(f) \quad \forall \alpha, \beta \geq 0, 0 \leq \alpha + \beta \leq 2n$. Para comprobar que $I_n(f)$ tiene exactitud $2n$ basta considerar la función $f(x,y) = r(r \cos \theta)^{2n+1}$. En este caso, $J^{2n+1,0} = 0$ y $J_n^{2n+1,0}, K_n^{2n+2}$ no se anulan. \square

Parte II

Competición en Kaggle: TalkingData AdTracking Fraud Detection Challenge.

Capítulo 4

Introducción

4.1. Descripción del problema.

En este trabajo se abordará la resolución de un problema planteado en la plataforma web Kaggle. Esta plataforma ofrece a sus usuarios la posibilidad de participar en distintas competiciones relacionadas con la Ciencia de Datos. Muchas de estas competiciones son problemas reales planteados por empresas, esto hace que el nivel de los participantes sea muy elevado ya que las primeras plazas de estos retos suelen estar recompensados con cuantiosas sumas de dinero o con la posibilidad de obtener un puesto de trabajo en dichas empresas.

Debido a las limitaciones temporales y de preprocesamiento, el objetivo de este trabajo no es sólo obtener la mejor clasificación posible sino estudiar cómo de importante es realizar un buen preprocesamiento y una buena elección del algoritmo a usar, frente a la capacidad de procesamiento de una máquina más potente.

En nuestro caso, he participado en la competición TalkingData AdTracking Fraud Detection Challenge que estuvo activa desde el 5 de marzo de 2018 hasta el 7 de mayo de 2018. El problema fue planteado por la empresa china TalkingData, siendo su objetivo reducir el fraude en los anuncios de apps. Este fraude se produce cuando se registran clicks en los anuncios y estos no conllevan la instalación de la app. Por tanto, se produce una facturación de los canales publicitarios que no concuerda con la realidad. TalkingData cubre más del 70 % de los dispositivos móviles activos en China. Manejan 3 mil millones de clics por día, de los cuales el 90 % son potencialmente fraudulentos. El desafío propuesto consiste en obtener un modelo para predecir si un usuario descargará una aplicación después de hacer clic en un anuncio de aplicación móvil.

Para llevar a cabo esta tarea han proporcionado un conjunto de datos que cubre aproximadamente 200 millones de clics. Estos datos contienen los siguientes atributos.

- **ip**: dirección IP de click.
- **app**: id de la aplicación
- **device**: identificación del tipo de dispositivo del teléfono móvil del usuario
- **channel**: id del canal del editor publicitario móvil
- **so**: id de la versión del OS del teléfono móvil del usuario
- **click_time**: marca de tiempo del click
- **attributed_time** : momento de la descarga de la aplicación
- **is_attributed** : el objetivo que se va a pronosticar, indica si la aplicación se descargó

Las soluciones obtenidas se evalúan usando la métrica área bajo la curva ROC.

La curva ROC representa la tasa de verdaderos positivos frente a la tasa de falsos positivos. Por tanto, cuanto mayor sea el área bajo dicha curva mejor será nuestro clasificador.

4.2. Herramientas utilizadas.

La primera decisión a tomar, es el lenguaje de programación en el que vamos a programar. Esta decisión debe ser tomada teniendo en cuenta los datos que necesitamos leer de los archivos proporcionados, los algoritmos que vamos a usar y la facilidad para generar los resultados que subiremos a la plataforma. Me he decantado por usar Python debido a que es un lenguaje en el que me siento cómodo, cuya sintaxis permite un código legible y que posee numerosas librerías de código abierto que satisfacen las necesidades del problema.

En concreto las principales librerías usadas son:

- **numpy**: para el uso de arrays y el tratamiento de los datos.
- **pandas**: para leer y escribir los archivos con los resultados.
- **matplotlib**: para la generación de gráficos
- **xgboost**[9]: implementa una forma específica de algoritmo de clasificación basado en árboles llamado Boosting.
- **Scikit Learn**[8]: librería para aprendizaje automático en Python

4.3. Proceso a seguir.

En primer lugar estudiaremos los datos proporcionados, de este modo podremos obtener algunas pistas sobre las operaciones a realizar durante la fase de preprocesamiento. Para ello, visualizaremos cómo se distribuyen los distintos datos, el número de valores vacíos de cada atributo y las posibles relaciones entre los distintos atributos.

A continuación, probaremos distintas técnicas de preprocesamiento con el fin de obtener el mejor conjunto de datos posible.

Finalmente, ejecutaremos distintos algoritmos para obtener el mejor clasificador posible.

Capítulo 5

Estudio de los datos.

Para poder elegir una estrategia para el preprocesamiento es necesario realizar una visualización de los datos. De esta forma, podremos obtener cómo están distribuidos los valores de cada uno de los atributos o si existe alguna relación de correlación entre ellos. En primer lugar, veamos la distribución de las variables categóricas *ip*, *app*, *channel*, *device* y *os*.

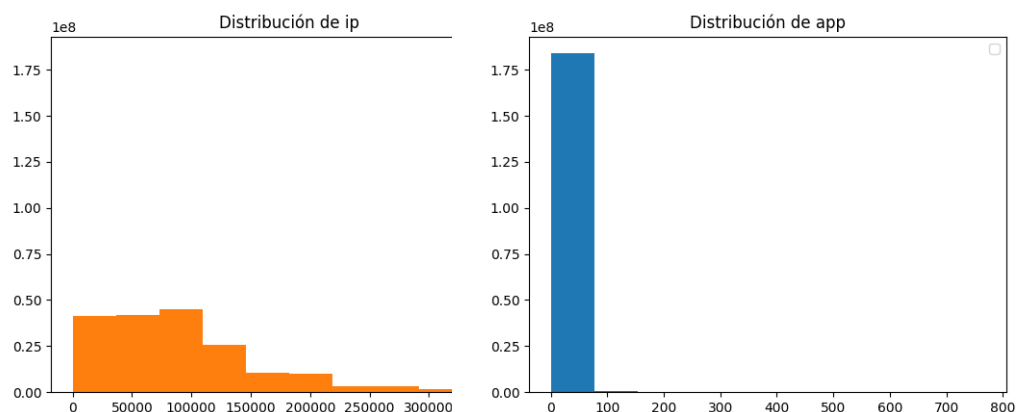
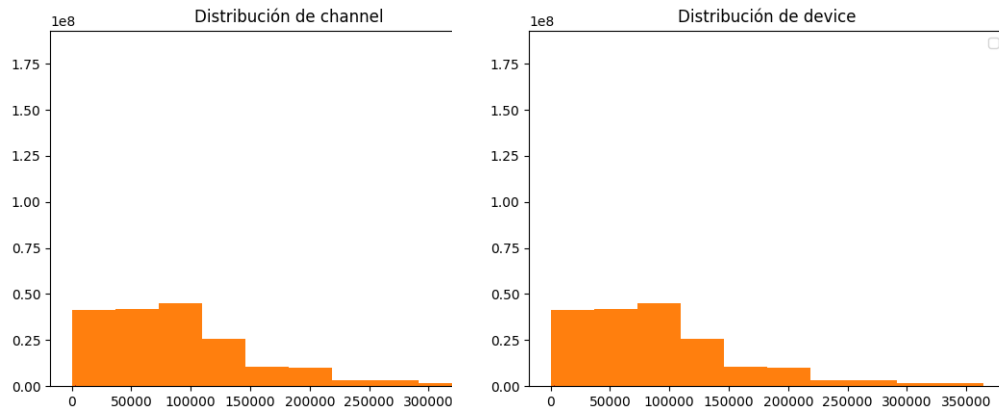
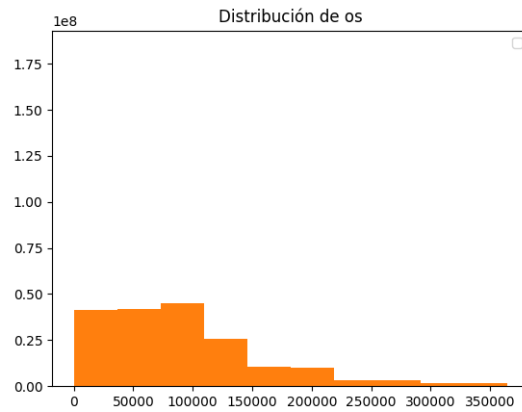


Figura 5.0.1: Distribución de *ip* y *app*

Figura 5.0.2: Distribución de *channel* y *device*Figura 5.0.3: Distribución de *os*

Podemos observar que aunque los valores no se concentran en torno a un valor, la distribución no es uniforme.

Ahora, estudiaremos la variable *click_time*. Al tratarse de una variable que representa una hora y fecha, la dividiremos en día, mes, año y valor timestamp(segundos transcurridos desde una fecha fijada por el sistema operativo).

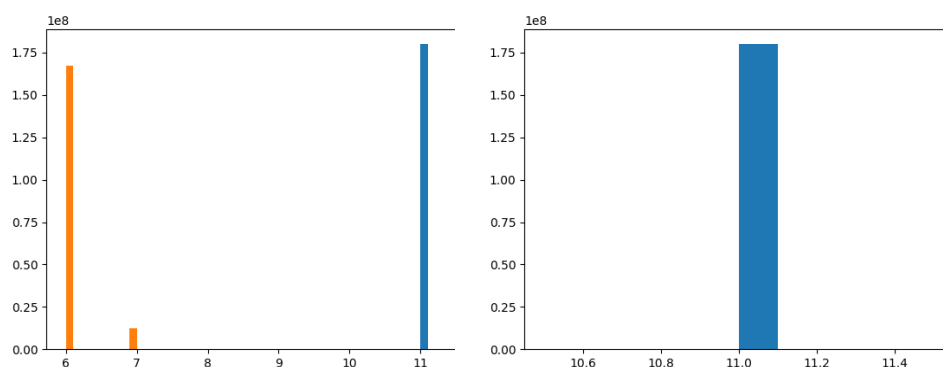


Figura 5.0.4: Distribución de *day* y *month*

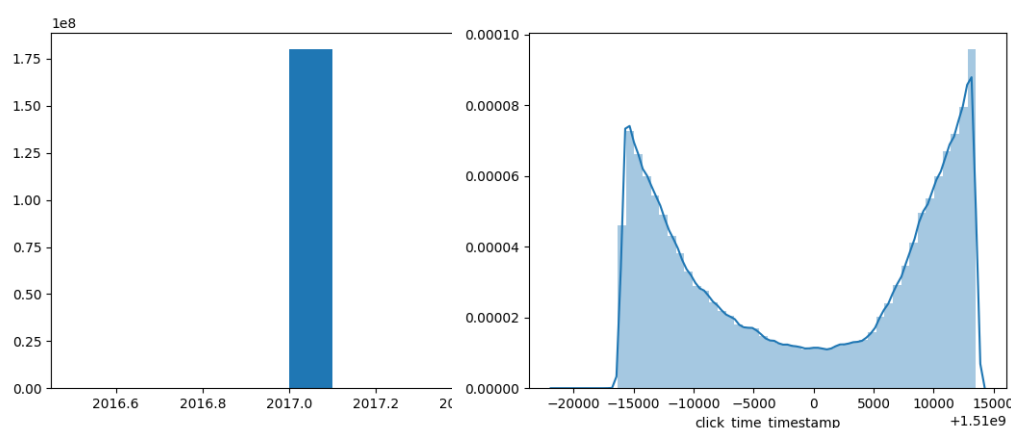


Figura 5.0.5: Distribución de *year* y *timestamp*

En esta ocasión sí podemos establecer valores fijos para algunos campos:

- Los datos corresponden a los días 6,7 y 8.
- Todos los valores corresponden al mes 11.
- Todos los registros son del año 2017.

Finalmente, estudiaremos el balanceo de clases de la variable a clasificar. Cabe recordar que los clasificadores estándar tienden a sobreaprender la clase mayoritaria, ignorando la minoritaria. En el siguiente gráfico podemos observar que el desbalanceo de clases es más que evidente. Por tanto, los algoritmos a desarrollar deberán tener en cuenta esta característica del conjunto de datos.

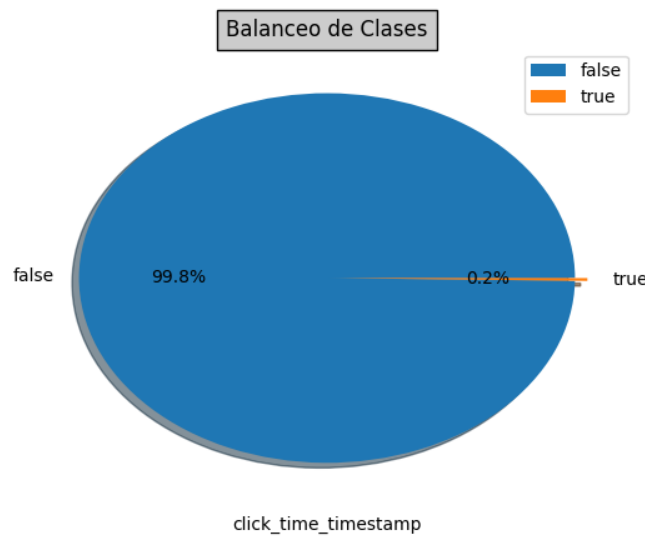


Figura 5.0.6: Balanceo de clases

Una vez hemos visto cómo se distribuyen los datos, a continuación calcularemos el número de valores desconocidos de cada columna. Esta información nos será de ayuda para detectar si una variable no aporta información. Los resultados obtenidos se muestran en la siguiente tabla.

Atributo	Total	%
ip	0	0
app	0	0
os	0	0
chanel	0	0
device	0	0
click_time	0	0
attributed_time	184447044	99.7529

Cuadro 5.1: Valores perdidos.

A la vista de los resultados obtenidos, podemos afirmar que la variable `attributed_time` no nos aporta información debido a que el número de instancias para las que existen valores, es ínfimo. En consecuencia, una de las decisiones a tomar durante la fase de preprocesamiento será no tener en cuenta estos datos en nuestro conjunto de entrenamiento.

Finalmente, obtenemos la matriz de correlación entre las variables categóricas usando el coeficiente de correlación de Spearman.

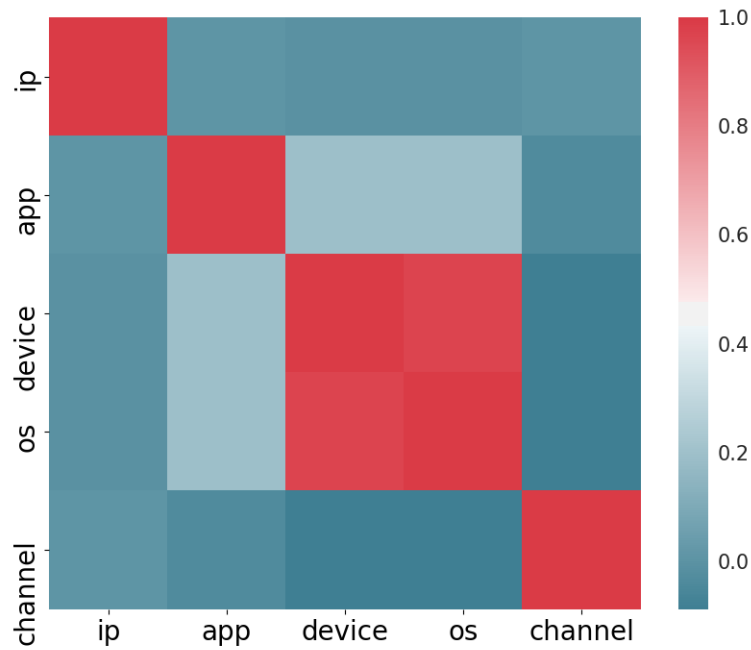


Figura 5.0.7: Matriz de correlación.

Observando el gráfico y teniendo en cuenta que el grado de correlación varía desde 0 hasta 1. Para este estudio, sólo vamos a tener en cuenta aquellas variables con una relación fuerte (valores cercanos a 1). Por tanto, podemos concluir que las variables device y os están relacionadas, mientras que no existen relación entre las demás variables.

Capítulo 6

Preprocesamiento.

Durante esta sección nos centraremos en tratar el conjunto de datos proporcionado, con el objetivo de extraer la máxima información posible que nos permita la construcción de un buen modelo de aprendizaje. En este tipo de competiciones la fase de preprocesamiento suele ser la que marca más la diferencia. Esto es debido a que a igualdad de capacidad de procesamiento, se puede obtener la configuración de parámetros óptima para los algoritmos usados, siendo los algoritmos usados similares entre los participantes.

El preprocesamiento de datos consiste en aplicar técnicas para manipular y transformar el conjunto de datos con el objetivo de mejorar la calidad de los datos. La mejora de dicho conjunto nos permitirá obtener mayor y mejor información de nuestro modelo de datos.

6.1. Proceso llevado a cabo.

La primera medida a tomar es asignar a cada columna el tipo de dato correcto, de esta forma conseguimos reducir notablemente la memoria ocupada por el conjunto de datos. Una de las mayores limitaciones con las que he tenido que lidiar durante la competición es la dificultad para tratar el conjunto de datos debido a su gran tamaño(7.7 GB en su versión inicial). Esto unido a que el uso de un lenguaje como Python genera un proceso pesado, ha provocado que en la mayoría de los casos la memoria RAM de mi ordenador fuera insuficiente para poder procesar todo el conjunto de datos. Por ello, en todas las decisiones tomadas he optado por elegir el enfoque más simple.

A continuación, vamos a obtener la máxima información posible de los datos proporcionados. Para ello vamos a realizar un proceso iterativo de preprocesamiento, manteniendo los cambios que proporcionen una mejora significativa y descartando aquellos cuya mejora no compense añadir una columna más. Para evaluar la bondad de las decisiones tomadas haremos uso

de la clasificación mediante boosting usando la librería *xgboost*. Gracias a esto podremos obtener la importancia de cada atributo a la hora de construir el modelo de aprendizaje. Este dato nos permitirá discriminar qué cambios nos proporcionan mejoras.

A partir de los resultados obtenidos en la sección anterior concluimos que:

- La columna *attributed_time* puede ser eliminada ya que la mayoría de sus valores son vacíos.
- De la variable *click_time* podemos obviar los datos relativos al mes y al año.
- La variable *app* concentra sus valores en torno al intervalo $[0, 100]$, por lo que podemos realizar agrupaciones sobre dicha variable, añadiendo una columna que contabilice el número de instancias coincidentes.
- Las variables *os* y *device* representan la misma información.

El primer cambio que vamos a realizar es eliminar la columna *attributed_time*, y obtener los datos *timestamp* y día de la variable *click_time*. Al efectuar este cambio hemos obtenido una mejora. A continuación, probaremos a considerar el día de la semana en vez del día. Tras comparar los resultados obtenidos en la construcción de ambos clasificadores podemos concluir que no existen diferencias entre ambos atributos.

Ahora, vamos a realizar agrupaciones en torno a las variables cuyos valores se repiten frecuentemente. Comenzaremos, añadiendo una columna para contar el número de veces que se repiten las combinaciones ip-app y ip-app-os. En esta ocasión hemos obtenido peor resultado que en las pruebas anteriores. Haremos lo mismo para las combinaciones ip-hora y ip-día-hora. A diferencia de la prueba anterior, este cambio sí conlleva una mejora del resultado.

Finalmente, realizaremos los cambios anteriores añadiendo el valor medio de la variable *channel*. Al igual que en el apartado anterior se obtienen mejores resultados cuando se manipulan las variables día y hora. Además, en general, considerar el valor medio conlleva mejores resultados que la simple cuenta de los valores repetidos.

Para obtener el mejor conjunto de entrenamiento posible, vamos a combinar aquellas variables creadas que más influyen a la hora de construir el modelo. Esta valiosa información la podemos obtener del clasificador construido por la librería *xgboost*. El conjunto de datos resultante nos lleva a mejorar el resultado final.

La siguiente tabla recoge los resultados obtenidos en las distintas fases del preprocesamiento.

Cambio realizado	Resultado
Conjunto inicial	0.8385028
Eliminar mes,año y click_time	0.8471159
Añadir día y hora	0.8471159
Cambiar día por día de la semana	0.8471159
count(channel) tras agrupar ip-app e ip-app-os	0.8289424
count(channel) tras agrupar ip-day e ip-day-hour	0.8569927
media(channel) tras agrupar ip-app e ip-app-os	0.8559159
media(channel) tras agrupar ip-day e ip-day-hour	0.8628768
Conjunto final	0.8649459

Cuadro 6.1: Pruebas realizadas durante el preprocesamiento.

Como podemos observar, durante esta fase hemos mejorado el rendimiento de nuestro clasificador un 3% aproximadamente.

Capítulo 7

Soluciones planteadas

. Como hemos visto en la sección de visualización nos encontramos ante un problema en el que existe un gran desbalanceo de clases. Por ello, las soluciones buscadas trataran de mitigar esta situación. Los algoritmos clásicos, no tienen un buen comportamiento en estos casos ya que tienden a clasificar como clase mayoritaria todos los ejemplos. Por tanto, durante esta fase vamos a enfocar nuestros esfuerzos en aplicar técnicas que tratan problemas de este tipo.

Dos de las técnicas más usadas para aliviar el problema de desequilibrio de clase son undersampling y oversampling. En el primer caso, se eliminan instancias de la clase mayoritaria siguiendo un criterio. Por otro lado, el oversampling consiste en generar nuevos ejemplos de la clase minoritaria hasta conseguir un número de instancias que generen un conjunto de datos balanceado. El principal inconveniente de undersampling es la pérdida de información asociada con la supresión de ejemplos. Sin embargo, tiene el beneficio de disminuir el conjunto de datos y en consecuencia, el tiempo necesario para construir el modelo. El sobremuestreo puede producir un sobreentrenamiento (el modelo se ajusta demasiado a los datos de entrenamiento) y conlleva un aumento del tiempo de entrenamiento.

A continuación, explicaremos los diferentes algoritmos que han sido usados en las distintas pruebas y los resultados proporcionados por cada uno de ellos.

7.1. Boosting

Como punto de partida usaremos el algoritmo Boosting.

Boosting es una técnica que combina varios clasificadores de forma que cada clasificador complementa a los demás. El proceso de construcción de los modelos es iterativo, cada modelo tiene en cuenta los errores de los modelos construidos anteriormente. Finalmente, la clase a la que pertenece

una instancia sin clasificar se obtiene mediante un voto ponderando entre los clasificadores construidos.

A continuación se detalla el proceso de entrenamiento y el de clasificación.

Algoritmo 1 Algoritmo boosting. Construcción del modelo.

1. Se inicializan los pesos, asignándole a cada instancia el mismo valor.
 2. Para cada iteración
 - a) Aplicar el clasificador
 - b) Obtener el error
 - c) Si el error es 0 o mayor que 0.5, finalizar.
 - d) Para cada instancia del conjunto de entrenamiento
 - Si la instancia es clasificada correctamente, multiplicar el peso de dicha instancia por $\text{error}/(1-\text{error})$
 - e) Normalizar los pesos de las instancias
-

Una vez obtenido el modelo de datos, para clasificar una nueva instancia se sigue el siguiente proceso:

Algoritmo 2 Algoritmo boosting. Clasificación

1. Asignar peso 0 a todas las clases.
 2. Para cada clasificador:
 - Sumar $-\log(\text{error} / (1 - \text{error}))$ al peso de la clase que devuelve el clasificador.
-

7.1.1. Optimización de parámetros

Para obtener los parámetros óptimos del algoritmo aplicaremos el concepto Grid Search[10]. Esto significa que ejecutaremos todas las combinaciones posibles de clasificadores a partir de los valores posibles introducidos. Este método es el ideal para obtener la configuración óptima para cada problema, sin embargo es un proceso computacionalmente costoso. Por ello, sólo llevaremos a cabo este proceso para algunos parámetros(`n_estimators`, `max_depth`, `subsample`, `scale_pos_weight`).

A continuación se detalla el significado de cada uno de ellos.

- `n_estimators`: número de clasificadores a usar.
- `max_depth`: la profundidad máxima de un árbol, aumentar este valor hará que el modelo sea más complejo.
- `subsample`: ratio de instancias usadas del conjunto de entrenamiento.
- `scale_pos_weight`: controla el balanceo de clases.

Además de los mencionados anteriormente, se han establecido los parámetros que controlan el tipo de clasificación y la métrica para evaluar. Tras obtener los mejores parámetros hemos mejorado la puntuación un 2,73 % .

7.2. RUSBoosting.

RUSBoost es un algoritmo que combina boosting con muestreo aleatorio. De esta forma el conjunto con el que se construye cada clasificador es un conjunto de datos balanceado. Además, para cada clasificador las instancias de la clase mayoritaria son diferentes. El algoritmo RUSBoosting se detalla a continuación.

Algoritmo 3 RUSBoosting

Sea S un conjunto de ejemplos con clase minoritaria $y \in Y$, un T un clasificador, k el número de iteraciones y N el porcentaje de instancias a ser representadas por la clase minoritaria.

1. Inicializar $D_1(i) = \frac{1}{m} \quad \forall i$
2. Para $t = 1, 2, \dots, N$
 - a) Crear un conjunto de entrenamiento S'_t con distribución D'_t .
 - b) Construir un modelo con T usando los ejemplos S'_t y sus pesos D'_t
 - c) Obtener una hipótesis $h_t : X \times Y \rightarrow [0, 1]$
 - d) Calcular la pérdida para $S_y D_t$:

$$\epsilon_t = \sum_{(i,y): y_i \neq y} D_t(i)(1 - h_t(x_i, y_i) + h_t(x_i, y))$$

- e) Calcular el nuevo valor del peso

$$\alpha_t = \frac{\epsilon_t}{1 - \epsilon_t}$$

- f) Actualizar D_t :

$$D_{t+1}(i) = D_t(i) \alpha_t^{\frac{1}{2}(1 + h_t(x_i, y_i) - h_t(x_i, y \neq y_i))}$$

- g) Normalizar D_{t+1} :

$$D_{t+1}(i) = \frac{D_{t+1}(i)}{\sum_i D_{t+1}(i)}$$

- h) Devolver la hipótesis final:

$$H(x) = \operatorname{argmax} \sum_{t=1}^N h_t(x, y) \log \frac{1}{\alpha_t}$$

La aplicación de este algoritmo no ha mejorado los resultados obtenidos anteriormente.

7.3. CUSBoosting

El algoritmo CUSBoosting sigue la misma filosofía que RUSBoosting. La diferencia entre ambos es la obtención del conjunto de datos balanceados. En este caso, dicho conjunto se obtiene usando el método de los centroides.

A continuación se detalla el algoritmo:

Algoritmo 4 CUSBoosting

1. inicializar los pesos $x_i \in D$ a $1/d$
 2. para $i = 1, \dots, k$
 - a) Crear un conjunto de datos balanceado con distribución D usando undersampling basado en clusters.
 - b) Obtener M_i de D_i
 - c) Calcular el error
 - d) si $error \geq 0,5$ entonces volver a a)
 - e) para cada $x_i \in D_i$ multiplicar el error asociado a x_i por $(\frac{error(M_i)}{1-error(M_i)})$ y actualizar los pesos.
 - f) Normalizar el peso de cada instancia.
-

Este algoritmo no ha sido de utilidad durante la competición, debido a cuestiones de capacidad de procesamiento.

7.4. Resultados obtenidos.

La siguiente tabla recoge los resultados obtenidos.

Algoritmo	Resultado
Boosting base	0.9509598
Boosting con parámetros optimizados	0.9769301
RUSBoosting	0.8763027
CUSBoosting	-
Boosting con conjunto balanceado	0.9441415

Cuadro 7.1: Resultados obtenidos con los diferentes algoritmos.

Capítulo 8

Conclusiones.

Los resultados obtenidos en esta última sección no han sido tan satisfactorios como a priori esperaba, ya que la aplicación de algoritmos alternativos no ha mejorado los resultados. Esto pone en relieve el porqué la mayoría de los participantes se decanta por el uso de xgboost, ya que al ser una librería de código abierto tiene un gran soporte que permite que el algoritmo esté muy optimizado. Sin embargo, los resultados obtenidos por RUSBoost parecen prometedores.

Por otro lado, hemos constatado la importancia de la fase de preprocesamiento ya que hemos conseguido una gran mejora sin importar la capacidad de procesamiento de nuestras máquinas. Esto nos lleva a concluir que en situaciones donde la importancia de procesamiento no fuera tan relevante, habríamos obtenido una mejor clasificación final.

Tras la finalización de la competición se ha obtenido la posición 1327 de 3951 participantes. Personalmente me ha parecido muy interesante afrontar este problema y poder comparar mis soluciones con las propuestas por gente experimentada en este ámbito. Haber participado en esta competición me ha servido para ganar experiencia, que me será de gran ayuda en el futuro. El proyecto, con los scripts usados son accesibles públicamente en https://github.com/daniL16/kaggle_talkingdata.

Apéndices A

La Función Gamma

Definición A.1. Dado $x \in \mathbb{R}^+$ definimos la función Gamma[1, sec. 1.2] como

$$\Gamma(x) := \int_0^\infty t^{x-1} e^{-t} dt$$

Proposición A.2. Se verifican las siguientes formulas:

$$\begin{aligned} \int_0^\infty t^{x-1} e^{-at^b} dt &= b^{-1} a^{-x/b} \Gamma(x/b), x, a, b \in \mathbb{R}^+ \\ \int_0^1 |\ln t|^{x-1} dt &= \Gamma(x), x \in \mathbb{R}^+ \\ \Gamma(x+1) &= x\Gamma(x), x \in \mathbb{R}^+ \\ \Gamma^{(k)}(x) &= \int_0^\infty (\ln t)^k t^{x-1} e^{-t} dt, k \in \mathbb{N}_0, x \in \mathbb{R}^+ \end{aligned} \tag{A.0.1}$$

Nota A.3. $\Gamma(1) = 1$ y de la tercera fórmula se deduce que $\Gamma(n+1) = n!$, $n \in \mathbb{N}_0$. Es decir, la función Γ extiende el operador factorial de los números naturales a los reales positivos.

Lema A.4.

$$\begin{aligned} \Gamma\left(\frac{1}{2}\right) &= \sqrt{\pi} \\ \Gamma\left(n + \frac{1}{2}\right) &= \frac{(2n)!}{2^{2n} n!} \sqrt{\pi} \end{aligned}$$

Definición A.5. Sea $x \in \mathbb{R}$ y $n \in \mathbb{N}$, el símbolo de Pochhammer se define como

$$(x)_0 = 1, (x)_n = x(x+1)(x+2)\dots(x+n-1)$$

Proposición A.6. Sea $x \in \mathbb{R}^+$ entonces

$$(x)_n = \frac{\Gamma(x+n)}{\Gamma(x)}$$

Apéndices B

Resultados básicos de la esfera.

En esta sección presentaremos algunos resultados referentes a la esfera que nos serán de utilidad [1, sec 1.3].

En \mathbb{R}^d usaremos la base canónica

$$e_1 = (1, 0, 0, \dots, 0)^T, \dots, e_d = (0, \dots, 0, 1)^T$$

Usaremos dV^d para elemento diferencial de volumen y dS^{d-1} para elemento diferencial de superficie de la esfera \mathbb{S}^{d-1} .

Proposición B.1. (*Coordenadas polares*)

$$x_1 = r \cos \theta \tag{B.0.1}$$

$$x_2 = r \sin \theta$$

$$0 \leq \theta \leq 2\pi, r \geq 0 \tag{B.0.2}$$

Proposición B.2. (*Coordenadas esféricas*)

$$x_1 = r \sin \theta \cos \phi$$

$$x_2 = r \sin \theta \sin \phi$$

$$x_3 = r \cos \theta$$

$$0 \leq \theta \leq \pi, 0 \leq \phi \leq 2\pi, r > 0$$

Proposición B.3. Para $d \geq 3$ y $\xi \in \mathbb{S}^{d-1}$, con $\xi_{(d)} = t e_d + \sqrt{1-t^2} \xi_{(d-1)}$, $t \in [-1, 1]$, se tiene que

$$dS^{d-1}(t e_d + \sqrt{1-t^2} \xi_{(d-1)}) = (1-t^2)^{\frac{d-3}{2}} dt dS^{d-2}(\xi_{(d-1)})$$

Equivalentemente,

$$dS^{d-1} = (1-t^2)^{\frac{d-3}{2}} dt dS^{d-2}$$

Ejemplo B.4. Sea $d=3$ y ξ un punto genérico de la esfera. Usando coordenadas esféricas

$$\xi_{(3)} = \begin{pmatrix} \cos \phi \sin \theta \\ \sin \phi \sin \theta \\ \cos \theta \end{pmatrix} \quad 0 \leq \phi \leq 2\pi, 0 \leq \theta \leq \pi$$

Sea $t = \cos \theta$ entonces

$$\xi_{(2)} = \begin{pmatrix} \cos \phi \\ \sin \phi \\ 0 \end{pmatrix}$$

Por tanto, $\xi_{(3)} = t e_3 + \sqrt{1-t^2} \xi_{(2)}$ y $dS^1 = d\phi$, $dS^2 = dt d\phi$

Podemos usar la anterior proposición para el cálculo del área de la superficie de la esfera.

Proposición B.5. *Se verifica que*

$$|\mathbb{S}^{d-1}| = \int_{\mathbb{S}^{d-1}} dS^{d-1} = \frac{2\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2})}$$

Proposición B.6. *Sea $A \in \mathbb{R}^{d \times d}$ ortogonal entonces*

$$dS^{d-1}(A\xi) = dS^{d-1}(\xi)$$

$$dV^d(A\xi) = dV^d(\xi)$$

Llamamos $C(S^{d-1})$ al espacio de funciones continuas sobre S^{d-1} . Este espacio es un espacio de Banach con la norma $\|f\|_\infty = \sup\{|f(\xi)| : \xi \in S^{d-1}\}$. Llamaremos $L^2(S^{d-1})$ al espacio de funciones con cuadrado integrable en S^{d-1} . Dicho espacio es un espacio de Hilbert con el producto escalar

$$(f, g) = \int_{S^{d-1}} f \bar{g} dS^{d-1}$$

Consideramos el espacio $C(S^{d-1})$ con la norma inducida por el producto escalar de $L^2(S^{d-1})$. Este espacio no es completo. Además, el cierre de $C(S^{d-1})$ respecto a dicha norma es $L^2(S^{d-1})$. Es decir, dado una función $f \in L^2(S^{d-1})$ existe una sucesión $\{f_n\} \subset C(S^{d-1})$ tal que $f_n \rightarrow f$

Proposición B.7. *Sean $\Omega_\delta = \{x \in \mathbb{R}^d : |x| \in [1-\delta, 1+\delta]\}$ y $f^*(x) = f(\frac{x}{|x|})$, $x \in \Omega_\delta$ y $k \in \mathbb{N}$. Entonces f es k veces diferenciable en S^{d-1} cuando f^* lo es.*

Definición B.8. Definimos $C^k(S^{d-1})$, $k \in \mathbb{N} \cup 0$ como el espacio de funciones k veces diferenciables en S^{d-1}

Proposición B.9. $C^k(S^{d-1})$ es un espacio de Banach con la norma

$$\|f\|_{C^k(S^{d-1})} = \|f^*\|_{C^k(\Sigma_\delta)}$$

Nota B.10. Usaremos $\|f\|_\infty = \|f\|_{C(S^{d-1})}$

Apéndices C

Polinomios de Legendre

C.1. Fórmulas de Representación

C.1.1. Fórmula de Rodrigues

Teorema C.1.

$$P_{n,d}(t) = (-1)^n \frac{\Gamma(\frac{d-1}{2})}{2^n \Gamma(n + \frac{d-1}{2})} (1-t^2)^{\frac{3-d}{2}} \left(\frac{d}{dt}\right)^n (1-t^2)^{n+\frac{d-3}{2}}, \quad d \geq 2$$

Nota C.2. A la constante $R_{n,d} = \frac{\Gamma(\frac{d-1}{2})}{2^n \Gamma(n + \frac{d-1}{2})}$ se le llama constante de Rodrigues

Nota C.3. Estos polinomios verifican $P_{n,d}(1) = 1$ y $\int_{-1}^1 P_{n,d}(t) P_{m,d}(t) (1-t^2) dt = 0$ para $m \leq n$

Ejemplo C.4. ■ Si $d = 2$,

$$P_{n,2}(t) = (-1)^n \frac{2^n n!}{(2n)!} (1-t^2)^{\frac{1}{2}} \left(\frac{d}{dt}\right)^n (1-t^2)^{n-\frac{1}{2}}, \quad n \in \mathbb{N}_0$$

Una forma reducida se obtiene usando el polinomio de Chebyshev obteniendo que $P_{n,2}(t) = \cos(n \arccos t)$, $t \in [-1, 1]$

■ Si $d=3$,

$$P_{n,3}(t) = \frac{1}{2^n n!} \left(\frac{d}{dt}\right)^n (t^2-1)^n, \quad n \in \mathbb{N}_0$$

C.1.2. Fórmulas de Representación Integral.

Teorema C.5. Sea $n \in \mathbb{N}_0$ y $d \geq 3$,

$$P_{n,d}(t) = \frac{|\mathbb{S}^{d-3}|}{|\mathbb{S}^{d-2}|} \int_{-1}^1 [t + i(1-t^2)^{\frac{1}{2}} s]^n (1-s^2)^{\frac{d-4}{2}} ds, \quad t \in [-1, 1]$$

Nota C.6. Como consecuencia de la fórmula anterior se tiene que $P_{n,d}(-t) = (-1)^n P_{n,d}(t)$, $t \in [-1, 1]$, es decir $P_{n,d}(t)$ tiene la misma paridad que n .

C.2. Propiedades

Proposición C.7. Si $f \in C^n([-1, 1])$ entonces

$$\int_{-1}^1 f(t) P_{n,d}(t) (1-t^2)^{\frac{d-3}{2}} dt = R_{n,d} \int_{-1}^1 f^{(n)}(t) (1-t^2)^{n+\frac{d-3}{2}} dt$$

siendo $R_{n,d}$ la constante de Rodrigues (Nota C.2)

Proposición C.8. (Norma)

$$\int_{-1}^1 [P_{n,d}(t)]^2 (1-t^2)^{\frac{d-3}{2}} dt = \frac{\sqrt{\pi} \Gamma(\frac{d-1}{2})}{\frac{(2n+d-2)(n+d-3)!}{n!(d-2)!} \Gamma(\frac{d}{2})}$$

Proposición C.9. (Ecuación diferencial)

$$(1-t^2)P''_{n,d}(t) - (d-1)tP'_{n,d}(t) + n(n+d-2)P_{n,d}(t) = 0$$

Proposición C.10. $P_{n,d}(t)$ tiene n raíces distintas en $(-1, 1)$

Proposición C.11. Los polinomios de Legendre satisfacen la siguiente relación de recurrencia

$$P_{n,d}(t) = \frac{2n+d-4}{n+d-3} t P_{n-1,d}(t) - \frac{n-1}{n+d-3} P_{n-2,d}(t), \quad n \geq 2, d \geq 2$$

$$P_{0,d}(t) = 1, P_{1,d}(t) = t$$

Proposición C.12.

$$(1-t^2)P'_{n,d}(t) = n[P_{n-1,d}(t) - tP_{n,d}(t)], \quad n \geq 1, d \geq 2, t \in [-1, 1]$$

Proposición C.13. Para $d \geq 2$

$$\sum_{n=0}^{\infty} \frac{(2n+d-2)(n+d-3)!}{n!(d-2)!} r^n P_{n,d}(t) = \frac{1-r^2}{(1+r^2-2rt)^{\frac{d}{2}}}, \quad |r| < 1, t \in [-1, 1]$$

Proposición C.14.

$$P_{n,d}(0) = \frac{|\mathbb{S}^{d-3}|}{|\mathbb{S}^{d-2}|} \int_{-1}^1 i^n s^n (1-s^2)^{\frac{d-4}{2}} ds$$

$$P_{n,d}(-1) = (-1)^n$$

Proposición C.15.

$$|P_{n,d}(t)| < \frac{\Gamma(\frac{d-1}{2})}{\sqrt{\pi}} \left[\frac{4}{n(1-t^2)} \right]^{\frac{d-2}{2}}, \quad n \in \mathbb{N}_0, d \geq 2, t \in (-1, 1)$$

Apéndices D

Polinomios de Gegenbauer

En esta sección estudiaremos los polinomios de Gegenbauer [1, sec. 2.9]

Definición D.1. Sean $v \geq 0, n \in \mathbb{N}_0$ se define el polinomio de Gegenbauer de grado n e índice v , como:

$$C_{n,v}(t) = \binom{n+2v-1}{n} \frac{\Gamma(v+\frac{1}{2})}{\sqrt{\pi}\Gamma(v)} \int_{-1}^1 [t + i(1-t^2)^{1/2}s]^n (1-s^2)^{v-1} ds$$

Los polinomios de Gegenbauer coinciden con los polinomios de Legendre salvo una constante de normalización.

Proposición D.2. *Se verifica la siguiente relación*

$$C_{n, \frac{d-2}{2}}(t) = \binom{n+d-3}{d} P_{n,d}(t)$$

Proposición D.3. *(Función generatriz)*

$$\sum_{n=0}^{\infty} C_{n,v}(t) r^n = \frac{1}{(1+r^2-2rt)^v}, \quad |r| < 1, t \in [-1, 1]$$

Proposición D.4. *(Ecuación diferencial, [2, sec 4.7])*

$$(1-x^2)C_{n,v}''(x) - (2v+1)x C_{n,v}'(x) + n(n+2v)C_{n,v}(x) = 0$$

Proposición D.5. *Se verifican las siguientes igualdades:*

- (I) $\frac{d}{dx} C_{n,v}(x) = 2v C_{n-1,v+1}(x)$
- (II) $n C_{n,v}(x) = x \frac{d}{dx} C_{n,v}(x) - \frac{d}{dx} C_{n-1,v}(x)$
- (III) $(n+2v) C_{n,v}(x) = \frac{d}{dx} C_{n-1,v}(x) - x \frac{d}{dx} C_{n,v}(x) \quad n \geq 0$
- (IV) $\frac{d}{dx} [C_{n+1,v}(x) - C_{n-1,v}(x)] = 2(n+v) C_{n,v}(x)$
 $= 2v [C_{n,v+1}(x) - C_{n-2,v+1}(x)] \quad n \geq 1$

Apéndices E

Funciones de Legendre Asociadas

Las funciones asociadas de Legendre [1, sec. 2.10] nos permiten construir armónicos esféricos a partir de otros de menor dimensión.

Definición E.1. Sea $d \geq 3$ y $n, j \in \mathbb{N}_0$ se define la función asociada de Legendre de grado n y orden j en dimensión d , como

$$P_{n,d,j}(t) = \frac{|\mathbb{S}^{d-3}|}{|\mathbb{S}^{d-2}|} i^{-j} \int_{-1}^1 \left[t + i(1-t^2)^{1/2}s \right]^n P_{j,d-1}(s) (1-s^2)^{\frac{d-4}{2}}, \quad t \in [-1, 1]$$

Nota E.2. Si $j = 0$, $P_{n,d,0}(t) = P_{n,d}(t)$

Proposición E.3. Sea $d \leq 3$ y $0 \leq j \leq n$

$$P_{n,d,j}(t) = \frac{n! \Gamma(\frac{d-1}{2})}{2^j (n-j)! \Gamma(j + \frac{d-1}{2})} (1-t^2)^{1/2} P_{n-j,d+2j}(t), \quad t \in [-1, 1]$$

El siguiente resultado nos proporciona una relación entre las funciones asociadas de Legendre y las derivadas de los polinomios de Legendre.

Proposición E.4. Sea $d \leq 3$ y $0 \leq j \leq n$

$$P_{n,d,j}(t) = \frac{(n+d-3)!}{(n+j+d-3)!} (1-t^2)^{1/2} P_{n,d}^{(j)}(t), \quad t \in [-1, 1]$$

Proposición E.5.

$$\int_{-1}^1 P_{m,d,j}(t) P_{n,d,j}(t) (1-t^2)^{\frac{d-3}{2}} dt = 0, \quad m \neq n$$

Proposición E.6. Las funciones $\tilde{P}_{n,d,j}$ definidas como

$$\tilde{P}_{n,d,j}(t) = \frac{[(2n+d-2)(n-j)!(n+d+j-3)!]^{1/2}}{2^{\frac{d-2}{2}n} \Gamma(\frac{d-1}{2})} P_{n,d,j}(t), \quad t \in [-1, 1]$$

están normalizadas, es decir $\int_{-1}^1 [\tilde{P}_{n,d,j}]^2 (1-t^2)^{\frac{d-3}{2}} dt = 1$

Nota E.7. Las funciones $\tilde{P}_{n,d,j}$ pueden ser escritas en función de las derivadas de los polinomios de Legendre

$$\tilde{P}_{n,d,j}(t) = \frac{(n+d-3)!}{n!\Gamma(\frac{d-1}{2})} \frac{[(2n+d-2)(n-j)!]^{1/2}}{2^{d-2}(n+d+j-3)!} (1-t^2)^{j/2} P_{n,d}^{(j)}(t), \quad t \in [-1, 1]$$

Apéndices F

El problema de clasificación

El problema de clasificación[11] consiste en predecir la clase a la que pertenece una instancia basándonos en una serie de características disponibles. Es un tipo de aprendizaje supervisado, es decir, se conoce de antemano la clases a las que pertenecen los ejemplos usados para construir el clasificador. El proceso de clasificación consta de las siguientes etapas:

1. **Construir el clasificador.** A partir de un conjunto de ejemplos ya clasificados. El modelo resultante establece
2. **Validación.** Se usa un conjunto distinto al de entrenamiento ya clasificado. Para cada instancia de este conjunto se compara el valor devuelto con el valor real.

Una de las técnicas para realizar la evaluación del modelo es la conocida como validación cruzada. Esta técnica consiste en dividir el conjunto de entrenamiento en k partes, usando $k-1$ partes para construir el clasificador y usar la restante para validar. Este proceso se repite durante k iteraciones de forma que cada parte es usada como conjunto de prueba. Finalmente, se ponderan los resultados de cada iteración para obtener el resultado final.

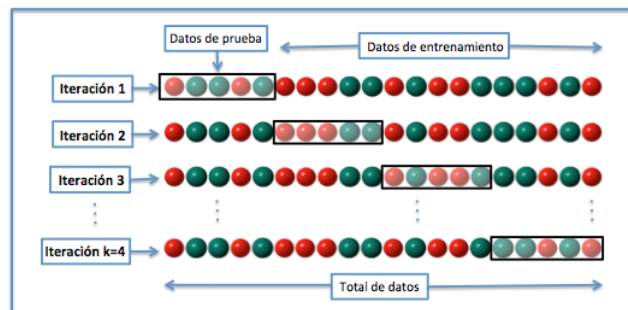


Figura F.0.1: Ejemplo validación cruzada.

Apéndices G

Visualización de los puntos críticos del gradiente sobre la esfera

```
# uso: python3 gradient.py n k [anim]

import numpy as np
import math
from sympy import cos, sin, acos, gegenbauer, var
from mayavi import mlab
from sympy.solvers import solve
from sympy import Symbol
import sys
import time

class Gradient:
def __init__(self, n, k):
    self.points = []
    self.n = int(n)
    self.k = int(k)
    self.plt = None
    self.current_k = 0

def pol(self, t):
    c = ((self.n + self.k) * (self.n + self.k - 1))
        / (2 * (2 * self.k - 1))
    d = (self.k + 0.5)
    return c *
        gegenbauer(self.n - self.k, self.k - 0.5, t)
```



```

+ d * (1 - t * t) *
gegenbauer(self.n - self.k - 2, self.k + 1.5, t)

def getPoints(self):
    self.points = []
    # polos
    polo_norte = (math.pi, 0)
    polo_sur = (0, 0)
    self.points.append(polo_norte)
    self.points.append(polo_sur)
    if self.k > 0 :
        # soluciones de cos kfi= 0
        phis = []
        for i in range(0, 2 * self.k):
            phis.append(-math.pi / (2 * self.k)
                        + (i * math.pi) / self.k)

        # soluciones del gegenbauer
        print("calculating")
        t = Symbol('t')
        solved = solve(self.pol(t),
                       t, quintics=False, quartics=False, cubics=False)
        # solucion de polinomio n-k = 0
        thetas = [acos(point) for point in solved]

        # generar los puntos cruzando los thetas y los phis
        for phi in phis:
            for theta in thetas:
                self.points.append([theta, phi])

def pintarEsfera(self):
    theta, phi = var('theta_phi')
    theta, phi = np.linspace(0, 2 * np.pi, 50),
                 np.linspace(0, np.pi, 25)

    THETA, PHI = np.meshgrid(theta, phi)

    X = np.sin(PHI) * np.cos(THETA)
    Y = np.sin(PHI) * np.sin(THETA)
    Z = np.cos(PHI)

    mlab.figure(1, bgcolor=(1, 1, 1),
                 fgcolor=(0, 0, 0), size=(600, 600))
    mlab.clf()

```

```

mlab.mesh(X, Y, Z, color=(0.9, 0.9, 0.9))

def printPoints(self, color=(1, 0, 0.3)):
    curva_pt = np.array([(sin(theta) * sin(phi),
                           sin(theta) * cos(phi),
                           cos(theta))
                          for theta, phi in self.points])
    xx = np.array([np.float(pt[0]) for pt in curva_pt])
    yy = np.array([np.float(pt[1]) for pt in curva_pt])
    zz = np.array([np.float(pt[2]) for pt in curva_pt])
    self.plt = mlab.points3d(xx, yy, zz,
                             scale_factor=0.05, color=color)

def pintarPuntosGradiente(self, color=(1, 0, 0.3),
                           animacion=False):
    self.getPoints();
    self.pintarEsfera()
    self.printPoints(color)

    if animacion:
        self.anim()
        mlab.show()

#funcion para generar la animacion
mlab.animate(delay=100)
def anim(self):
    msplt = self.plt.mlab_source
    k = self.k
    start_time = time.time()
    for i in range(self.current_k, self.n):
        print(str(i) + "/" + str(self.n))
        # paramos el proceso 2 segundos
        # para mejorar la visualizacion
        time.sleep(2)
        self.k = i
        self.getPoints()
        curva_pt = np.array(
            [(sin(theta) * sin(phi),
              sin(theta) * cos(phi), cos(theta))
             for theta, phi in self.points])
        xx = np.array([np.float(pt[0]) for pt in curva_pt
                        ])
        yy = np.array([np.float(pt[1]) for pt in curva_pt
                        ])

```

```

        zz = np.array([np.float(pt[2]) for pt in curva_pt
                        ])
        msplt.reset(x=xx, y=yy, z=zz)
        print("——%s seconds——" % (time.time() -
            start_time))
        self.current_k = i
    yield

n = sys.argv[1]
k = sys.argv[2]
animacion = False
if (len(sys.argv) > 3):
    animacion = sys.argv[3]
k = 0

gradiente = Gradient(n, k)
gradiente.pintarPuntosGradiente(animacion=animacion)

```

Bibliografía

- [1] K. Atkinson, Weimin Han *Spherical Harmonics and Approximations on the Unit Sphere: An Introduction*, 2nd ed., Springer.
- [2] Gabor Szegő *Orthogonal Polynomials* American Mathematical Society, 1939.
- [3] Miguel A. Piñar, Yuan Xu *Best Polynomial Approximation on the Unit Ball*
- [4] Ian H. Witten, Eibe Frank, *Data Mining Practical Machine Learning Tools and Techniques*, 2nd edition Morgan Kaufmann Publishers, 2005.
- [5] Y. Freund and R. Schapire, *Experiments with a new boosting algorithm. In Proceedings of the Thirteenth International Conference on Machine Learning* pages 148–156, 1996.
- [6] Chris Seiffert, Taghi M. Khoshgoftaar, Jason Van Hulse, Amri Napolitano *RUSBoost: Improving Classification Performance when Training Data is Skewed*
- [7] Farshid Rayhan, Sajid Ahmed, Asif Mahbub, Md. Rafsan Jani, Swakkhar Shatabda, Dewan Md. Farid *CUSBoost: Cluster-based Under-sampling with Boosting for Imbalanced Classification* December 2017
- [8] <http://scikit-learn.org/stable/>
- [9] <http://xgboost.readthedocs.io/en/latest/>
- [10] Kevin Lemagnen *Hyperparameter tuning in XGBoost* <https://cambridgespark.com/content/tutorials/hyperparameter-tuning-in-xgboost/index.html>
- [11] G. Shmueli, N.R. Patel, P.C. Bruce *Data mining for business intelligence* Wiley, 2010 (2nd. edition)