

Universidad de Salamanca
Grado en Matemáticas

REVISIÓN DE MÉTODOS
MULTIVARIANTES
SUPERVISADOS Y NO SUPERVISADOS

Trabajo Fin de Grado



VNiVERSiDAD
D SALAMANCA

Alumno: Pedro Ángel Fraile Manzano

Tutoras: Ana Belén Nieto Librero y Nerea González García

Salamanca, Julio de 2023

*A Loreto, que me inspiró a no abandonar
mis sueños aunque las tormentas fueran duras.*

*A Paula, la que me iluminó
en los momentos más oscuros.*

*A mi familia, que empujó mis aspiraciones
hasta que se convirtieron en realidades.*

*A Moloka'i, Titania y Pangea que me lo
dieron todo cuando el mundo se paró.*

Índice general

1	Métodos Supervisados	3
1.1	Métodos Lineales para regresión	4
1.1.1	Métodos de ajuste	5
1.1.2	Inferencias Estadísticas sobre $\hat{\beta}$	7
1.1.3	Regresión multivariante	10
1.2	Clasificación supervisada	13
1.2.1	Formalización	13
1.2.2	Análisis mediante Variables Canónicas discriminantes	14
1.3	Redes Neuronales	18
1.3.1	Projection Pursuit Regression	19
1.3.2	Red Neuronal de 2 capas	20
1.3.3	Ajuste y uso de una Red Neuronal	21
1.4	Árboles de Decisión y Bosques Aleatorios	22
1.4.1	Árboles de Regresión	23
1.4.2	Árboles de Clasificación	25
1.4.3	Bosques Aleatorios	26
2	Métodos no supervisados	29
2.1	Análisis de Componentes Principales	30
2.1.1	Definición y cálculo de las Componentes	31
2.1.2	PCA muestral	33
2.1.3	Reconstrucción de los datos	33
2.2	Análisis Factorial	38
2.2.1	Formalización	38
2.2.2	Estimación de la matriz de cargas	40
2.2.3	Unicidad del modelo	40
2.2.4	Interpretación del modelo factorial	40
2.3	Análisis de Clusters	41
2.3.1	Algoritmos Jerárquicos	41
2.3.2	Algoritmos Particionales	43

3	Aplicación sobre datos	45
3.1	Clasificación de Iris	45
3.2	Alubias secas	46
3.3	Comparación regresión lineal, árboles de regresión y redes neuronales.	46
	Bibliografía	47

Introducción

El Análisis Multivariante es, según *Martínez Arias, R.* [21], el conjunto de técnicas y métodos que busca describir y extraer información de las relaciones entre variables medidas en una o varias muestras u observaciones. Dentro de esta definición, entran todos los procedimientos que analizan de manera simultánea más de una variable.

La clasificación que se dará en esta memoria de los métodos multivariantes será la siguiente

- **Métodos supervisados:** Son aquellos que exploran la relación estocástica entre varias variables, divididas en variables respuesta y variables predictoras, estas técnicas a su vez se pueden dividir según la naturaleza de las variables respuesta [20]:
 - *Regresión:* Cuando las variables de respuesta son continuas, dentro de esta tipología, entran métodos como las *Redes Neuronales* [30], *los Árboles de Regresión* [31] etc...
 - *Clasificación:* Cuando las variables de respuesta son discretas, como por ejemplo, el *Análisis discriminante* [24], *Regresión logística* [32],
- **Métodos no supervisados:** Los métodos no supervisados son aquellos que buscan analizar la estructura y las relaciones entre las distintas variables [9]. Las distintas técnicas se distinguen según la medida en la que se centren, por ejemplo la variabilidad común [28], la homogeneidad de grupos [29] etc...

Estas técnicas han visto un auge en los últimos años, debido a su utilidad en el análisis de grandes bases de datos y la creciente capacidad de recogida de datos que se tiene en la actualidad. Basta con ver que la mayoría de artículos recopilados en los que se aplican estas técnicas se han publicado en las dos primeras décadas de siglo. Además, el llevar a buen término el estudio anteriormente era complejo debido a la gran cantidad de cálculos que se necesitaban realizar. Actualmente esos cálculos son automatizados, en algunos casos, con una única línea de código es posible llevar a cabo ciertas técnicas de las descritas.

Muchas veces, el uso de estas técnicas se combinan con el aprendizaje automático como proponen *Hastie T., Tibshirani R. y Friedman J.* [9] o *James G, Witten D, Hastie T, y Tibshirani R.* [20]. De esta manera, los parámetros se estiman extrayendo la información de los propios datos, comparando distintos modelos etc... Utilizando esta aproximación se pueden generar Este aspecto no se desarrollará en esta memoria.

La estructura de esta memoria se divide en tres partes, una dedicada a los métodos supervisados en la que se detallan los más importantes, como la regresión lineal, el análisis discriminante, los árboles de decisión y las redes neuronales. En la segunda parte se desarrollan los métodos no supervisados con técnicas como el análisis de componentes principales, el análisis factorial y el análisis de conglomerados. Por último, se realizarán aplicaciones en las que se desarrollarán las interpretaciones de las técnicas descritas durante la memoria.

En resumen, el principal objetivo de este trabajo es describir las principales técnicas de análisis de multivariante y fundamentar su aplicación e interpretación sobre cualquier tipo de datos. Además se busca dar ejemplos de aplicación sobre datos, sacando conclusiones que sean provechosas y que con técnicas univariantes

Capítulo 1

Métodos Supervisados

Sea un conjunto de variables aleatorias observables de manera simultánea en una población. Estas variables se pueden dividir en dos tipos [9]:

- **Variables de entrada o predictoras:** Son las variables independientes que determinarán de manera aleatoria al segundo conjunto de variables. Al conjunto de variables aleatorias de entrada se la denotará como el vector aleatorio $\mathbf{x} = [X_1, \dots, X_p]$.

Definición 1.0.1. Tomando N observaciones de estas variables de manera simultánea se obtiene la *matriz de datos* \mathbf{X} . Esta matriz es de tamaño $N \times p$ y contiene como filas los vectores de longitud p que representan los datos de cada observación, se denotarán a lo largo de la memoria como $\mathbf{x}_i, i = 1 \dots N$. Por ejemplo, en el caso de que se recojan datos sobre distintos modelos de coches las observaciones serían los valores medidas de las distintas variables consideradas en cada uno de los coches.

- **Variables de salida :** Son las variables dependientes de las anteriores. A este conjunto de variables se les denota con el vector aleatorio $\mathbf{y} = [Y_1, \dots, Y_K]$, en el caso de que se tenga una única variable respuesta se denotará como Y . Como en el caso de las variables de entrada, se tomarán N observaciones de las K variables, dando como resultado la matriz de respuestas \mathbf{Y} de tamaño $N \times K$, donde cada observación es una fila y se denota como $\mathbf{y}_i, i = 1 \dots K$, en el caso de que $K > 1$ y como y_i cuando $K = 1$. Siguiendo el ejemplo anterior, se podrían tener variables respuesta como el tipo de etiqueta medioambiental siendo esta una variable discreta o el precio del vehículo que tiene naturaleza continua.

Por ende, se recogen observaciones simultáneas de las variables de entrada y de salida formando parejas $(\mathbf{y}_i, \mathbf{x}_i), i = 1 \dots N$, de manera que se obtiene un vector fila de tamaño $p + K$. Estas observaciones de las variables forman una muestra aleatoria de la población .

Definición 1.0.2. Se llama *muestra aleatoria* de una variable aleatoria con una cierta distribución de probabilidad F , a un conjunto de N variables aleatorias con la misma distribución.

Definición 1.0.3. Se llaman métodos supervisados [41] a aquellos métodos que buscan inferir una relación estocástica entre dos grupos de variables, predictoras y respuesta en un conjunto de datos contiene observaciones simultaneas de ambos conjuntos de datos.

Durante la memoria, se establecerán los fundamentos a nivel poblacional, solo teniendo en cuenta las propiedades de las variables aleatorias y luego en caso de que sea necesario se tomarán N observaciones o realizaciones del experimento para formar las matrices de datos correspondientes.

En conclusión, el objetivo de estos métodos supervisados es encontrar una relación estocástica que llamaremos predictor, de tal manera que para una nueva observación de las variables de entrada, que denotaremos con \mathbf{x}_0 , se pueda hacer una predicción $\hat{\mathbf{y}}_0$ del valor real \mathbf{y}_0 de la variable respuesta siempre con un cierto error que será una variable aleatoria.

1.1. Métodos Lineales para regresión

El objetivo de la regresión es encontrar como un conjunto de variables respuesta se ven afectadas por otro conjunto de variables predictoras. En este caso, se estudia como una combinación lineal de las variables puede afectar a las variables respuesta, que en el caso de la regresión es una variable aleatoria continua. Esto se puede hallar con fines predictivos o con el fin de analizar cómo cada una de las variables predictivas afectan a las variables respuesta [34].

Para ello, se tendrá en cuenta que las observaciones de la variable respuesta siguen la siguiente relación estocástica [9, 34]:

$$Y = f(\mathbf{x}) + \varepsilon \quad (1.1.1)$$

Donde ε es una variable aleatoria con $\mathbb{E}(\varepsilon) = 0$ y $Var(\varepsilon) = \sigma^2$, habitualmente $\varepsilon \sim N(0, \sigma^2)$. Además supóngase que el vector de variables de entrada sigue una distribución normal multivariante $N_p(\mu, \Sigma)$ donde Σ es una matriz semidefinido positiva y simétrica, [6]. A esta última suposición se le podría añadir que sea diagonal, de manera que las variables aleatorias predictoras no estén correladas, pero no es necesario.

En el caso que nos atañe actualmente, se supondrá que f es una función lineal de las variables de entrada del vector aleatorio $\mathbf{x} = [X_1 \dots X_p]$. De esta manera, se tiene que:

$$f(\mathbf{x}) = \beta_0 + \sum_{j=1}^p X_j \beta_j \quad (1.1.2)$$

Definición 1.1.1. Se llaman *parámetros de regresión* al vector columna $\beta = [\beta_0, \beta_1, \dots, \beta_p]^T$ de tamaño $(p+1)$ con los coeficientes necesarios para la regresión.

Añadiendo al vector \mathbf{x} una nueva variable aleatoria X_0 que sea constantemente 1, se puede dar la siguiente expresión matricial de la función f :

$$f(\mathbf{x}) = \mathbf{x}\beta \quad (1.1.3)$$

De esta manera, tomando la matriz de datos de entrada de tamaño $N \times (p + 1)$ entonces se puede generar un vector de predicciones de tamaño N de la siguiente manera:

$$\hat{\mathbf{y}} = \mathbf{X}\beta \quad (1.1.4)$$

Si en particular se quiere hacer una predicción para una nueva observación \mathbf{x}_0 basta con calcular $f(\mathbf{x}_0)$.

Una vez establecida la notación y los supuestos que tomaremos de ahora en adelante, hay que estimar los parámetros de regresión, para ello utilizaremos distintos métodos como el método de los mínimos cuadrados o el de Máxima verosimilitud, a parte de esto, veremos las características inferenciales de los estimadores obtenidos y su interpretación geométrica.

1.1.1. Métodos de ajuste

Sea una matriz de datos \mathbf{X} de tamaño $N \times (p+1)$ resultado de hacer N observaciones de p variables aleatorias y añadir a la primera componente de cada observación un 1. Sea también un vector de respuestas $\mathbf{y} = [y_1, \dots, y_N]^T$ de tamaño N , resultado de observar la variable respuesta Y .

Sea el vector de parámetros de regresión β de tamaño $p + 1$ definido como antes, entonces podemos definir el concepto de función de pérdida.

Definición 1.1.2. Se llama *función de pérdida* [9] a una función $L(\mathbf{y}, \hat{\mathbf{y}})$ monótona creciente de la diferencia entre la predicción y el valor real de una variable

Aunque el término sea más utilizado en el marco del aprendizaje automático [20], el concepto de error cuadrático o la función de pérdida cuadrática es el que se suele usar en el ajuste de los parámetros de regresión, Ya que esta pérdida al minimizarse, es equivalente al método de los mínimos cuadrados, muy utilizado en distintos campos. En particular, la pérdida cuadrática se define de la siguiente manera [2]:

Definición 1.1.3. Se llama *error de predicción cuadrático* entre una variable observada, y , y la predicción obtenida por un cierto modelo \hat{y} a la expresión $(y - \hat{y})^2$

Tomando la suma de los errores cuadrados cometidos en todas las observaciones y minimizando se puede obtener una estimación del vector de parámetros β . En particular, si se utilizan las expresiones matriciales anteriores:

$$RSS(\beta) = \sum_{i=1}^N (y_i - \hat{y}_i)^2 = \sum_{i=1}^N (y_i - \mathbf{x}_i\beta)^2 = (\mathbf{y} - \mathbf{X}\beta)(\mathbf{y} - \mathbf{X}\beta)^T \quad (1.1.5)$$

Para obtener el mínimo se debe hallar los puntos estacionarios, lo que implica calcular la derivada de la suma de residuos cuadrados respecto del vector β , y como esta es una forma cuadrática general respecto de β , (Para el desarrollo de la expresión véase [17]):

$$\frac{\partial RSS(\beta)}{\partial \beta} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta)^T \quad (1.1.6)$$

La segunda derivada respecto del vector de parámetros es $2\mathbf{X}^T\mathbf{X}$, entonces es semi-definido positiva, ya que todos los valores propios son positivos o nulos. Entonces los β en los que $-2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta)^T = 0$ son mínimos.

Asumiendo que \mathbf{X} es una matriz de rango máximo $p + 1$, entonces la matriz $\mathbf{X}^T\mathbf{X}$ es invertible, por tanto la siguiente expresión tiene solución única:

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0 \quad (1.1.7)$$

Esa solución es :

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \quad (1.1.8)$$

Para el cálculo de este parámetro se ha utilizado lo que se conoce por el método de los mínimos cuadrados. Aún así hay otros métodos de obtención como el geométrico o el de máxima verosimilitud.

Con la siguiente proposición se puede ver la equivalencia entre los mínimos cuadrados y el de máxima verosimilitud:

Proposición 1.1.1. Esta estimación de los parámetros β por mínimos cuadrados es equivalente a la estimación de estos mediante el método de máxima verosimilitud [9].

Demostración. Sea una muestra de tamaño N , $y_i, i = 1, \dots, N$ de una variable aleatoria Y , de manera que su función de probabilidad, $\mathbb{P}_\theta(y)$ depende de los parámetros θ . Entonces el método de máxima verosimilitud busca maximizar la siguiente función.

$$L(\theta) = \sum_{i=1}^n \log(\mathbb{P}_\theta(y_i)) \quad (1.1.9)$$

Suponiendo que la variable respuesta cumple como antes que $Y = f_\theta(\mathbf{x}) + \varepsilon$, donde $\varepsilon \sim N(0, \sigma^2)$, provoca que si se suponen conocidos a priori el parámetro θ y el vector aleatorio \mathbf{x} entonces :

$$\mathbb{P}(Y|\mathbf{x}, \theta) = N(f_\theta(\mathbf{x}), \sigma^2) \quad (1.1.10)$$

Teniendo esto en cuenta, la función $L(\theta)$ tiene la siguiente expresión:

$$L(\theta) = -\frac{n}{2}\log(2\pi) - n\log(\sigma) - \frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - f_\theta(x_i))^2 \quad (1.1.11)$$

Por tanto, teniendo en cuenta que el último término es $RSS(\theta)$, entonces maximizar $L(\theta)$ es equivalente a minimizar $RSS(\theta)$. \square

Corolario 1.1.1. Las estimaciones para cualquier f_θ obtenidas por el método de los mínimos cuadrados son equivalentes a las del método de máxima verosimilitud.

Por tanto, los valores predichos obtenidos de las observaciones recogidas en la matriz de datos $\hat{\mathbf{y}}$ se calculan de la siguiente manera:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \quad (1.1.12)$$

De esta manera, se está calculando la predicción de la respuesta \hat{y}_i para cada una de las observaciones $\mathbf{x}_i, i = 1, \dots, N$.

Otra forma de considerar el problema del ajuste sería desde el punto de vista geométrico para el que hay que desarrollar las siguientes definiciones y proposiciones.

En lo sucesivo se considera a no ser que se diga lo contrario que $N > p + 1$ y que la matriz de datos \mathbf{X} es de rango máximo, $p + 1$.

Definición 1.1.4. Se llama $C_{p+1}(\mathbf{X})$ al subespacio lineal de \mathbb{R}^N generado por las columnas de la matriz \mathbf{X} [8].

Proposición 1.1.2. El vector $\hat{\varepsilon} = \mathbf{y} - \mathbf{X}\hat{\beta}$ es ortogonal al subespacio $C_{p+1}(\mathbf{X})$.

Demostración. Premultiplicando por \mathbf{X}^T :

$$\mathbf{X}^T \hat{\varepsilon} = \mathbf{X}^T (\mathbf{y} - \mathbf{X}\hat{\beta}) = \mathbf{X}^T \mathbf{y} - \mathbf{X}^T \mathbf{X} \hat{\beta} = 0 \quad \square$$

Abdi, H [2] desarrolla este tipo de ajuste de manera superficial, mientras que *Hastie et.al.* [9] detallan este marco geométrico utilizando el proceso de ortonormalización de Graham-Schmidt, para encontrar combinaciones ortonormales de las variables originales. Este proceso se basa como se puede ver, en el hecho de que el vector de residuo generado debe ser ortogonal al espacio $C_{p+1}(X)$. Aunque no se desarrolle el ajuste en su totalidad, la proposición anterior será utilizada para demostrar una propiedad de las distribuciones de los estimadores hallados.

1.1.2. Inferencias Estadísticas sobre $\hat{\beta}$

En esta parte se estudian las propiedades de los estimadores obtenidos mediante el método de los mínimos cuadrados. Tras ello, se formarán estadígrafos para realizar los contrastes de hipótesis necesarios.

Hay que tener en cuenta los siguientes supuestos el vector $\mathbf{x} \sim N_p(\mu, \Sigma)$ una distribución multivariante de dimensión p , del cual tomamos N observaciones independientes obteniendo la matriz de datos \mathbf{X} , que a no ser que se diga lo contrario, se supondrá conocida. Además, supóngase el vector de N observaciones de la variable respuesta sigue el siguiente modelo $\mathbf{y} = \mathbf{X}\beta + \varepsilon$ donde $\varepsilon \sim N(0, \sigma^2 \mathbf{I}_N)$, es un vector de longitud N en el que cada componente es una normal $N(0, \sigma^2)$ independientes entre sí. Esto nos permite dar la siguiente proposición [8]

Proposición 1.1.3. El vector aleatorio formado por N observaciones de la variable respuesta conocida la matriz de datos y el vector de parámetros β sigue una distribución $\mathbf{y} \sim N_N(\mathbf{X}\beta, \sigma^2 \mathbf{I}_N)$, donde \mathbf{I}_N es la matriz identidad de tamaño N .

Demostración. Basta con comprobar que:

$$\mathbb{E}(\mathbf{y}) = \mathbb{E}(\mathbf{X}\beta) + \mathbb{E}(\varepsilon) = \mathbb{E}(\mathbf{X}\beta) = \mathbf{X}\beta \quad (1.1.13)$$

Y la varianza :

$$\begin{aligned} \mathbb{E}((\mathbf{X}\beta + \varepsilon)(\mathbf{X}\beta + \varepsilon)^T) &= \mathbf{X}\beta\beta^T \mathbf{X}^T + \text{Var}(\varepsilon) \\ \text{Var}(\mathbf{y}) &= \mathbf{X}\beta\beta^T \mathbf{X}^T + \text{Var}(\varepsilon) - \mathbf{X}\beta\beta^T \mathbf{X}^T = \text{Var}(\varepsilon) = \sigma^2 \mathbf{I}_N \end{aligned} \quad (1.1.14)$$

Y se concluye que el vector $\mathbf{y} \sim N_N(\mathbf{X}\beta, \sigma^2 \mathbf{I}_N)$, ya que el vector $\varepsilon \sim N_N(0, \sigma^2 \mathbf{I}_N)$ por hipótesis. \square

Añadamos a las suposiciones que la matriz \mathbf{X} es de rango máximo y por tanto, $\mathbf{X}^T \mathbf{X}$ es semidefinido positiva. A continuación se detallarán las cualidades inferenciales del vector $\hat{\beta}$, como que es insesgado, su varianza y su distribución.

Proposición 1.1.4. El estimador $\hat{\beta}$ es un estimador insesgado [33]

Demostración. Teniendo en cuenta las anteriores hipótesis anteriores, incluyendo que la variable respuesta es:

De esta manera, se obtiene que $\hat{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T (\mathbf{X}\beta + \varepsilon) = \beta + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \varepsilon$

Entonces, se tiene que conocida $\mathbf{X} \Rightarrow \mathbb{E}(\hat{\beta}|\mathbf{X}) = \mathbb{E}(\beta + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \varepsilon|\mathbf{X}) = \beta + \mathbb{E}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \varepsilon|\mathbf{X}) = \beta$ debido a la distribución de ε .

Por tanto como $\mathbb{E}(\mathbb{E}(\hat{\beta}|\mathbf{X})) = \mathbb{E}(\beta) = \beta$ queda demostrado. \square

Proposición 1.1.5. La varianza del estimador $Var(\hat{\beta}) = \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1}$ [9], [33]

Demostración.

$$\begin{aligned} Var(\hat{\beta}) &= \mathbb{E}((\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \varepsilon \varepsilon^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} | \mathbf{X}) \\ &= \mathbb{E}(\varepsilon \varepsilon^T) (\mathbf{X}^T \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{X}) (\mathbf{X}^T \mathbf{X})^{-1} \\ &= \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1} \end{aligned} \quad (1.1.15)$$

\square

Por último, tenemos que el parámetro de estimadores $\hat{\beta} = \beta + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \varepsilon$, es una combinación afín de ε una variable con distribución por tanto, tendremos que cada uno de los componentes del vector $\hat{\beta}$, $\hat{\beta}_j \sim N(\beta_j, \sigma^2 (\mathbf{X}^T \mathbf{X}_{jj}^{-1}))$ donde $j = 1 \dots p$ y $(\mathbf{X}^T \mathbf{X}_{jj}^{-1})$ es el j-ésimo elemento de la diagonal de la matriz $(\mathbf{X}^T \mathbf{X})^{-1}$ [34].

Si el parámetro σ^2 fuera conocido, entonces se podría llevar a cabo los contrastes de manera sencilla. En este caso, $z_j = \frac{\hat{\beta}_j}{\sigma \sqrt{v_j}} \sim N(0, 1)$

Para poder hacer los estadígrafos de contrastes se debe construir un estimador de esta varianza, ya que habitualmente no es conocida. Para ello *Hastie et. al.* [9] y *Cuadras C.M* [8] proponen el siguiente estimador de la varianza, suponiendo que unicamente hay una variable respuesta:

$$\hat{\sigma}^2 = \frac{1}{N - p - 1} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (1.1.16)$$

Proposición 1.1.6. El estimador $\hat{\sigma}^2$ cumple lo siguiente :

- $\mathbb{E}(\hat{\sigma}^2) = \sigma^2$ (Es un estimador insesgado de la varianza)
- $\hat{\sigma}^2 \sim \frac{\sigma^2}{N-p-1} \chi_{N-p-1}^2$

Demostración. El término $\sum_{i=1}^N (y_i - \hat{y}_i)^2 = RSS(\hat{\beta})$, se puede expresar como un producto escalar, tomando el vector residuo $\hat{\varepsilon} = \mathbf{y} - \mathbf{X}\hat{\beta} = \mathbf{y} - \hat{\mathbf{y}}$, se puede ver que el $RSS(\hat{\beta}) = \hat{\varepsilon}^T \hat{\varepsilon}$.

Sabiendo que $\hat{\varepsilon} \in \mathbb{R}^N$, consideremos una nueva base del espacio, $\{t_1, \dots, t_{p+1}, t_{p+2}, \dots, t_N\}$, de tal manera que los $p+1$ primeros son una base de $C_{p+1}(\mathbf{X})$ y que sean ortogonales entre si, entonces, se puede tomar la matriz de cambio de base \mathbf{T} , es una matriz ortogonal, de manera que $\mathbf{T}^T \mathbf{T} = \mathbf{T} \mathbf{T}^T = \mathbf{I}$.

Entonces, el vector $\mathbf{T}\hat{\varepsilon} = (\overbrace{0 \dots 0}^{p+1}, e'_{p+2}, \dots, e'_N)^T$ por la proposición 1.1.2, y se cumple que $\mathbb{E}(\hat{\varepsilon}) = \mathbb{E}(\mathbf{T}\hat{\varepsilon}) = 0$, por la distribución que tiene el vector ε además $\mathbb{E}((e'_i)^2) = \sigma^2, i = p+2 \dots N$ ya que para los anteriores es 0 y por último, $\mathbb{E}(\hat{\varepsilon}^T \hat{\varepsilon}) = \mathbb{E}(\hat{\varepsilon}^T \mathbf{T}^T \mathbf{T} \hat{\varepsilon}) = \sum_{p+2}^N (e'_i)^2 = (N - p - 1)\sigma^2$

Por tanto:

$$\mathbb{E}(\hat{\sigma}^2) = \frac{1}{N - p - 1} \mathbb{E}(RSS(\hat{\beta})) = \sigma^2 \quad (1.1.17)$$

Se concluye entonces que el estimador $\hat{\sigma}^2$, es insesgado.

Y para terminar, se puede ver que el estimador es suma de $N - p - 1$ normales estándar multiplicadas por σ^2 y divididas entre $N - p - 1$, por tanto, tenemos que además $\hat{\sigma}^2 \sim \frac{\sigma^2}{N - p - 1} \chi_{N-p-1}^2$. \square

Observación: en el caso de que la matriz de datos \mathbf{X} sea de rango $r < p+1$ se debe construir un estimador distinto, para el cual se puede seguir el mismo razonamiento:

$$\hat{\sigma}^2 = \frac{1}{N - r} \sum_{i=1}^N N(y_i - \hat{y}_i)^2 \quad (1.1.18)$$

Por tanto, ahora ya podemos establecer, conocidas las distribuciones de los estimadores construidos que [33], [9]:

$$t_j = \frac{\hat{\beta}_j}{\hat{\sigma}^2 \sqrt{v_j}}, j = 1, \dots, p+1 \sim t_{N-p-1} \quad (1.1.19)$$

Donde v_j es como antes el j -ésimo elemento de la diagonal de $\mathbf{X}^T \mathbf{X}$

Conocer la distribución de los estimadores permite realizar contrastes sobre los distintos parámetros. De manera habitual, se plantea la hipótesis nula $H_0 : \beta_j = 0$. Esta hipótesis busca comprobar si la variable X_j es importante en el modelo lineal. En caso de que se acepte la hipótesis esa variable puede ser eliminada del modelo.

Teniendo en cuenta esto, hay ocasiones en las que se desea comprobar si un subconjunto de variables es más significativo estadísticamente que otro, de manera que se pueda eliminar variables que no aporten información en pos de la sencillez del modelo y su posterior interpretación.

Sea $p_1 + 1$ el número del conjunto más grande de parámetros o de variables a considerar y RSS_1 su error cuadrático respectivo, sean respectivamente RSS_0 y $p_0 + 1$ para el segundo conjunto o subconjunto menor, si además contamos que :

$$\frac{RSS(\hat{\beta})}{N - p - 1} = \frac{1}{N - p - 1} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \sim \sigma^2 \chi_{N-p-1} \quad (1.1.20)$$

Se puede definir el estadígrafo:

$$F = \frac{\frac{(RSS_0 - RSS_1)}{p_1 - p_0}}{\frac{RSS_1}{N - p_1 - 1}} \quad (1.1.21)$$

Que cumple:

Proposición 1.1.7. El estadígrafo $F \sim F_{(p_1 - p_0), (N - p_1 - 1)}$

Demostración. Teniendo en cuenta lo anterior, se puede comprobar que:

$$F \sim \frac{\chi_{p_1 - p_0}^2}{\chi_{N - p_1 - 1}^2} = F_{(p_1 - p_0), (N - p_1 - 1)} \quad (1.1.22) \quad \square$$

Este estadígrafo se referencia en secciones sucesivas con el objetivo de reducir las variables involucradas en la regresión.

1.1.3. Regresión multivariante

Hasta ahora, se ha considerado una única variable aleatoria respuesta, sea \mathbf{y} el vector aleatorio de variables respuesta $\mathbf{y} = [Y_1, \dots, Y_K]$ y un vector aleatorio de variables de entrada $\mathbf{x} = [X_0, X_1, \dots, X_p]$. Se puede establecer el siguiente modelo análogo, donde :

$$Y_k = \beta_{0k} + \sum_{j=1}^p X_j \beta_{jk} + \varepsilon_k = f_k(\mathbf{x}) + \varepsilon_k, \quad k = 1, \dots, K \quad (1.1.23)$$

Donde el término $\varepsilon_k \sim N(0, \sigma_k^2)$ es el error inevitable para la variable $Y_k, k = 1 \dots K$. De esta manera, si se toman N observaciones se puede crear las matrices de datos de respuesta y de entrada y obtener la siguiente expresión:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \quad (1.1.24)$$

Dentro de esa expresión matricial se tiene que:

- \mathbf{Y} es la matriz de tamaño $N \times K$ que contiene los valores observados de las variables respuesta
- \mathbf{X} matriz de datos habitual de tamaño $N \times (p + 1)$
- \mathbf{B} matriz de tamaño $(p + 1) \times K$ que contiene los parámetros de la regresión
- \mathbf{E} es la matriz de tamaño $N \times K$ que contiene los errores cometidos en cada uno de las variables respuesta.

El proceso de ajuste, en el caso de que los errores $\varepsilon^T = [\varepsilon_1, \dots, \varepsilon_K]$ no estén correlados, de la matriz de parámetros es análogo al de una sola variable respuesta de tal manera que minimizando el error cuadrático acumulado se obtiene $\hat{\mathbf{B}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}$. En caso de que los errores tengan una matriz de covarianzas conocida, Σ , entonces es necesario hacer la siguiente modificación en el RSS :

$$RSS(\mathbf{B}, \Sigma) = \sum_{i=1}^n (y_i - f(\mathbf{x}_i))^T \Sigma^{-1} (y_i - f(\mathbf{x}_i)) \quad (1.1.25)$$

En la última expresión se usa la *distancia de Mahalanobis*.

Definición 1.1.5. La *distancia de Mahalanobis* entre dos observaciones $\mathbf{x}_i, \mathbf{x}_j$ extraídas de una misma población con matriz de covarianzas Σ se define de la siguiente manera[8]:

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T \Sigma^{-1} (\mathbf{x}_i - \mathbf{x}_j)} \quad (1.1.26)$$

En el caso anterior, $(y_i - f(\mathbf{x}_i)) = \varepsilon_i$ y se está haciendo la distancia de Mahalanobis respecto de su media, el 0.

Selección de subconjuntos y métodos penalizados

Anteriormente, se ha detallado un estadígrafo que permitía hacer un contraste sobre la cantidad de variables a considerar en la regresión. Esta reducción de características a considerar permite hacer mucho más interpretable el modelo obtenido durante todo el proceso de ajuste y en algunos casos incluso aumentar la precisión del modelo ya que puede ocurrir que se eliminen variables que introduzcan ruido en el modelo.

Se podría seguir un método exhaustivo que calcule cada uno de los subconjuntos posibles para cada número de variables que creamos necesarias y calcular el error cuadrático acumulado de cada uno de los subconjuntos posibles. Pero este método es de una complejidad computacional alta.

Otra posibilidad sería utilizar el estadígrafo de contraste F definido en la Ecuación (1.1.21) empezando con una sola variable e ir añadiendo cada variable que mejore el ajuste. También se puede hacer al revés, empezando con el modelo con todas las variables e ir reduciendo la cantidad de estas. A estos algoritmos se les llaman *algoritmos voraces* los cuales buscan una solución óptima en cada paso y no en global.

El problema de estos métodos de selección de variables es que es un proceso discreto, una variable es o no considerada para el modelo siguiente y esto puede generar sobreajuste o infraajuste, no habiendo un término medio.

Para ello, existen los métodos penalizados o de encogimiento, que añaden un término de penalización para que los parámetros β no sean muy grandes.

Definición 1.1.6. Llamamos *Errores Cuadrados Acumulados Penalizados*, $PRSS$, a la suma de los cuadrados de los errores cometidos con el modelo lineal que usa el vector de parámetros β , añadiendo un término regulador $\lambda > 0$:

$$PRSS(\beta) = \sum_{i=1}^n (\mathbf{y}_i - \mathbf{x}_i^T \beta)^2 + \lambda \sum_{j=1}^p \beta_j^2 \quad (1.1.27)$$

El último término hace que parámetros β_j grandes sean considerados perjudiciales, y el parámetro λ es una forma de regular cuanta importancia tiene dicha penalización, cuanto mayor sea, este provocará un encogimiento mayor de los parámetros. De esta manera, se tiene una forma continua de considerar los pesos no eliminando en de manera total las variables o haciendo el β_j correspondiente cercano a 0.

1.2. Clasificación supervisada

Supóngase un conjunto de observaciones de un vector aleatorio \mathbf{x} de tamaño p las cuales pertenecen a dos o más poblaciones conocidas a priori. El objetivo es dada una nueva observación \mathbf{x}_0 poder asignarla a una de las poblaciones. Otro objetivo que se puede buscar con estas técnicas es determinar que variables son determinantes y tienen mayor importancia a la hora de discriminar entre varias categorías.

Este problema se le conoce como *Clasificación Supervisada*, ya que se conoce un conjunto de observaciones de las cuales se sabe a que población pertenece. También se le conoce como *Análisis Discriminante*, ya que se pueden establecer funciones que permitan discriminar observaciones unas de otras.

Alguna de las aplicaciones que utilizan las técnicas que se describen en esta sección son el reconocimiento de patrones, la concesión de créditos, o asignación de textos a autores.

Las primeras aproximaciones que se dieron de estas técnicas fueron dadas por *Fisher R.* en 1936 introduciendo el conjunto de datos *Iris de Fisher* para la clasificación.

Durante esta sección se seguirán *Hastie T. et.al.*[9], *Cuadras C.M.*[8] y *Peña D.* [22]

1.2.1. Formalización

Supóngase dadas dos poblaciones P_1, P_2 de las cuales son conocidas sus funciones de densidad f_1, f_2 . Si además se conocen las probabilidades a priori de pertenencia a cada una de las poblaciones π_1, π_2 Entonces la probabilidad conociendo el valor de \mathbf{x}_0 de pertenencia a $i = 1, 2$ es :

$$P(i|\mathbf{x}_0) = \frac{P(\mathbf{x}_0|i)\pi_i}{\pi_1 P(\mathbf{x}_0|1) + \pi_2 P(\mathbf{x}_0|2)} = \frac{f_i(\mathbf{x}_0)\pi_i}{f_1(\mathbf{x}_0)\pi_1 + f_2(\mathbf{x}_0)\pi_2} \quad (1.2.1)$$

De esta manera se puede clasificar como una población dependiendo de que $f_i(\mathbf{x}_0)\pi_i$ sea mayor (*Esto es generalizable a más poblaciones.*)

Definición 1.2.1. Se llama *función discriminante* aquella que:

$$f_i : \Omega \longrightarrow \mathbb{R} \quad (1.2.2)$$

Donde Ω es el espacio de observaciones posibles. Definida de tal manera que si $f_i(\mathbf{x}_0) > 0 \Rightarrow \mathbf{x}_0 \in P_i$ y en caso contrario $\mathbf{x}_0 \notin P_i$.

En el caso anterior de dos poblaciones, $f_1(\mathbf{x}_0)\pi_1 - f_2(\mathbf{x}_0)\pi_2$ sería la función discriminante para discernir si \mathbf{x}_0 pertenece a P_1 . Ya que representa la probabilidad de ser de dicha clase conociendo el valor de las variables predictoras.

Definición 1.2.2. Se llama *Error de clasificación* al coste de clasificar de manera errónea una observación y se denota como $c(i|j) = \text{“Error de clasificar como } P_i \text{ una observación perteneciente a } P_j \text{”}$

De esta manera, si el coste de mala clasificación en una y otra población son distintas podemos trasladarlo a la función discriminante de la siguiente manera:

$$\frac{f_1(\mathbf{x}_0)\pi_1}{c(1|2)} - \frac{f_2(\mathbf{x}_0)\pi_2}{c(2|1)} \quad (1.2.3)$$

Por consiguiente, a igualdad de los otros términos escogeremos el que menos coste, el que mayor verosimilitud o el que mayor probabilidad a priori tenga.

Supóngase ahora que tanto f_1, f_2 son densidades normales con medias distintas μ_1, μ_2 pero con una matriz de covarianzas común Σ :

$$f_i(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{p}{2}}|\Sigma|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2}(\mathbf{x} - \mu_i)^T \Sigma^{-1}(\mathbf{x} - \mu_i) \right\} \quad (1.2.4)$$

De esta manera la función discriminante se puede transformar sustituyendo las densidades por la expresión anterior y tomando logaritmos :

$$(\mathbf{x} - \mu_1)^T \Sigma^{-1}(\mathbf{x} - \mu_1) + \log \left(\frac{\pi_1}{c(1|2)} \right) - (\mathbf{x} - \mu_2)^T \Sigma^{-1}(\mathbf{x} - \mu_2) - \log \left(\frac{\pi_2}{c(2|1)} \right) \quad (1.2.5)$$

Hay que tener en cuenta que el término $D_i^2 = (\mathbf{x} - \mu_i)^T \Sigma^{-1}(\mathbf{x} - \mu_i)$ se puede interpretar como la *distancia de Mahalanobis* de la observación \mathbf{x} a la i -ésima población. Por otro lado, se tiene un término que relaciona probabilidad de pertenencia con el coste de clasificación errónea. En el caso de que tanto las probabilidades como el coste fueran iguales, entonces la función discriminante para la población 1 sería:

$$D_2^2 > D_1^2 \quad (1.2.6)$$

Definición 1.2.3. Llamaremos *frontera de decisión* al hiperplano que divide el espacio de observaciones en tantas regiones como poblaciones haya. En el caso de una función discriminante como las desarrolladas la frontera son las observaciones tales que $f(\mathbf{x}) = 0$

Para obtener la frontera de decisión se igualan ambos términos y desarrollando ambas distancias podríamos desechar el término $\mathbf{x}^T \Sigma^{-1} \mathbf{x}$ ya que es común a ambas, obteniendo la siguiente expresión:

$$(1.2.7)$$

1.2.2. Análisis mediante Variables Canónicas discriminantes

Hasta ahora, se han supuesto conocidas las distribuciones de las poblaciones estudiadas de manera que utilizando las probabilidades de pertenencia y los costes se desarrollan funciones discriminantes de manera sencilla. Pero la realidad pocas veces se ajusta a estos supuestos, es por ello que surgen las técnicas que descomponen la varianza en dos términos la varianza dentro de los grupos y fuera de ellos de tal manera que se busca minimizar la primera y maximizar la segunda.

Es por ello, que a continuación se detallará el método por el cual se va a calcular una nueva base del espacio de observaciones que cumpla lo anterior, minimizar la

variación entre poblaciones y maximizar la variabilidad entre grupos siguiendo el desarrollo dado por *Lebart L., Morineau, A. y Warwick K.M.* [16]

Sea \mathbf{X} la matriz de datos de tamaño $n \times p$ donde las filas \mathbf{x}_i son cada una de las observaciones de las p variables. Dichas observaciones están particionadas en general por q grupos, sea I_k el conjunto de observaciones pertenecientes al k -ésimo grupo, sea también n_k el número de observaciones que pertenecen al k -ésimo grupo. Se definen las medias muestrales $\bar{x}_j = \frac{1}{n} \sum_{i=1}^n x_{ij}$ de la j -ésima variable en la población en total. También se define la media muestral dentro de cada grupo que es $\bar{x}_{jk} = \frac{1}{n_k} \sum_{i \in I_k} x_{ij}$

Por ende podemos dar la distancia entre dos variables como:

$$Cov(X_j, X_{j'}) = \frac{1}{n} \sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}) \quad (1.2.8)$$

Esto se puede particionar por grupos de la siguiente manera:

$$Cov(X_j, X_{j'}) = \frac{1}{n} \sum_{k=1}^q \sum_{i \in I_k} (x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}) \quad (1.2.9)$$

y a su vez cada uno de los $(x_{ij} - \bar{x}_j)$ se pueden dividir en la parte intergrupos e intragrupos:

$$(x_{ij} - \bar{x}_j) = (x_{ij} - \bar{x}_{jk}) + (\bar{x}_{jk} - \bar{x}_j) \quad (1.2.10)$$

Sustituyendo y simplificando lo necesario:

$$Cov(X_j, X_{j'}) = \frac{1}{n} \sum_{k=1}^q \sum_{i \in I_k} (x_{ij} - \bar{x}_{jk})(x_{ij'} - \bar{x}_{j'k}) + \sum_{k=1}^q \frac{n_k}{n} (\bar{x}_{jk} - \bar{x}_j)(\bar{x}_{j'k} - \bar{x}_{j'}) \quad (1.2.11)$$

Esto nos permite dar una descomposición de la matriz de covarianzas total de la siguiente forma :

$$\mathbf{T} = \mathbf{B} + \mathbf{W} \quad (1.2.12)$$

Donde:

- \mathbf{T} es la matriz que expresa la covarianza total y sus coeficientes $t_{jj'} = Cov(X_j, X_{j'})$
- \mathbf{B} es la matriz que expresa la covarianza entre los grupos y sus coeficientes son $b_{jj'} = \sum_{k=1}^q \frac{n_k}{n} (\bar{x}_{jk} - \bar{x}_j)(\bar{x}_{j'k} - \bar{x}_{j'})$
- \mathbf{W} es la matriz que expresa la covarianza dentro de los grupos y sus coeficientes son $w_{jj'} = \frac{1}{n} \sum_{k=1}^q \sum_{i \in I_k} (x_{ij} - \bar{x}_{jk})(x_{ij'} - \bar{x}_{j'k})$

Para cualquier combinación lineal que se quiera hacer de las variables de entrada de la forma $\mathbf{a}^T \mathbf{x}$, donde el vector \mathbf{a} es un vector de p constantes, entonces la varianza se transforma de la siguiente manera:

$$Var(\mathbf{a}^T \mathbf{x}) = \mathbf{a}^T \Sigma \mathbf{a} \quad (1.2.13)$$

Entonces transformando por el vector \mathbf{a} tenemos que la Ecuación (1.2.12) se transforma de la siguiente manera:

$$\mathbf{a}^T \mathbf{T} \mathbf{a} = \mathbf{a}^T \mathbf{B} \mathbf{a} + \mathbf{a}^T \mathbf{W} \mathbf{a} \quad (1.2.14)$$

Recopilando, el objetivo del análisis discriminante lineal es encontrar combinaciones lineales que maximicen la varianza entre grupos y minimicen la varianza dentro de los grupos. Eso es equivalente a encontrar el vector \mathbf{a} tal que:

$$f(\mathbf{a}) = \frac{\mathbf{a}^T \mathbf{B} \mathbf{a}}{\mathbf{a}^T \mathbf{T} \mathbf{a}} \quad (1.2.15)$$

Es máxima. Si además utilizamos la restricción $\mathbf{a}^T \mathbf{T} \mathbf{a} = 1$. En principio la función objetiva es homogénea, es decir, $f(\mu \mathbf{a}) = f(\mathbf{a})$ Utilizando el método de los multiplicadores de Lagrange derivamos respecto del vector \mathbf{a} tendremos que:

$$L(\mathbf{a}) = \mathbf{a}^T \mathbf{B} \mathbf{a} - \lambda(\mathbf{a}^T \mathbf{T} \mathbf{a} - 1) \quad (1.2.16)$$

al derivarla respecto de \mathbf{a} se obtiene que:

$$\frac{\partial L(\mathbf{a})}{\partial \mathbf{a}} = 2\mathbf{B} \mathbf{a} - 2\lambda \mathbf{T} \mathbf{a} \quad (1.2.17)$$

En consecuencia:

$$\mathbf{B} \mathbf{a} = \lambda \mathbf{T} \mathbf{a} \quad (1.2.18)$$

Si además \mathbf{T} es no singular

$$\mathbf{T}^{-1} \mathbf{B} \mathbf{a} = \lambda \mathbf{a} \quad (1.2.19)$$

Es decir, el vector \mathbf{a} es el vector de valor propio λ , tomando el valor propio máximo de la matriz $\mathbf{T}^{-1} \mathbf{B}$.

Definición 1.2.4. Al valor λ se le conoce como *potencia discriminante* de la combinación \mathbf{a} .

Proposición 1.2.1. El desarrollo antes detallado es análogo para $f(\mathbf{a}) = \frac{\mathbf{a}^T \mathbf{B} \mathbf{a}}{\mathbf{a}^T \mathbf{W} \mathbf{a}}$

En el caso de que tomamos el desarrollo para obtener los vectores propios de $\mathbf{W}^{-1} \mathbf{B}$ es una matriz con rango $r = \min(p, q - 1)$ entonces podemos utilizar la matriz de cambio de base \mathbf{U}_r que tiene como columnas los vectores propios de la matriz $\mathbf{W}^{-1} \mathbf{B}$ y \mathbf{x} el vector a transformar, de esta manera, tenemos que $\mathbf{z} = \mathbf{U}_r^T \mathbf{x}$.

A continuación para determinar si una nueva observación \mathbf{x}_0 basta con calcular la distancia con las medias de los grupos transformadas, es decir, $\|\mathbf{z}_0 - \mathbf{U}_r \mu_i\|^2$ y comprobar cual es la menor.

Para determinar cuantas variables canónicas se puede utilizar la varianza entre grupos que explica cada variable canónica.

Proposición 1.2.2. La varianza explicada por cada variable canónica es el valor propio de la matriz $\mathbf{W}^{-1}\mathbf{B}$ de su vector propio asociado, su *potencia discriminante*.

Demostración. Sea el vector propio \mathbf{a}_i con valor propio λ_i entonces tendremos que por construcción $\mathbf{a}_i^T \mathbf{W} \mathbf{a}_i = 1$, entonces los vectores son unitarios respecto a la métrica que induce la matriz de covarianzas intra-grupos, entonces la varianza explicada por cada variable es $VE(\mathbf{a}_i) = \mathbf{a}_i^T \mathbf{B} \mathbf{a}_i$, por ser valores propios de $\mathbf{W}^{-1}\mathbf{B}$ se cumple que $\mathbf{B} \mathbf{a}_i = \lambda_i \mathbf{W} \mathbf{a}_i$, entonces

$$VE(\mathbf{a}_i) \mathbf{a}_i^T \mathbf{B} \mathbf{a}_i = \lambda_i \mathbf{a}_i^T \mathbf{W} \mathbf{a}_i = \lambda_i \quad (1.2.20)$$

□

Para elegir un número de variables canónicas a usar se puede fijar un umbral por el cual tomemos las m variables cuya varianza explicada acumulada sea superior. De esta manera, se puede llevar a cabo una reducción de la dimensionalidad de los datos de manera parecida a la que se desarrollará en el *Análisis de Componentes Principales*. La principal diferencia con este último es que mientras uno busca representar de una manera simple la variabilidad de los datos, esta busca las direcciones en las cuales los grupos o clases se distinguen de manera más significativa

1.3. Redes Neuronales

Las redes neuronales artificiales, son un algoritmo basado en el funcionamiento de las propias neuronas del cerebro que reciben señales de entradas de las neuronas con las cuales están conectadas, las procesan y envían el resultado a las neuronas con las que estén conectadas.

El algoritmo de aprendizaje que oculta una red neuronal es la optimización de algún funcional, de manera que estas se pueden aplicar a problemas que consistan en la resolución de dichos problemas. La ventaja de este tipo de algoritmos es que en esencia, son un conjunto de parámetros que pueden ser ajustados para cualquier tarea y cualquier tipo de función a aproximar solo hace falta la complejidad del modelo adecuada según *Hornik, K., Stinchcombe, M., y White, H.* [26].

Una neurona artificial es mucho más simple que una neurona, *López R. E. Balcanto y E. Oñate* [15] lo define de la siguiente manera:

Definición 1.3.1. Una neurona procesa una entrada \mathbf{x} de acuerdo con unos pesos sinápticos (b, ω) que luego es transformada por una función de activación $g(u)$, las más habituales son la función sigmoide, una función lineal, la tangente hiperbólica o directamente la identidad.

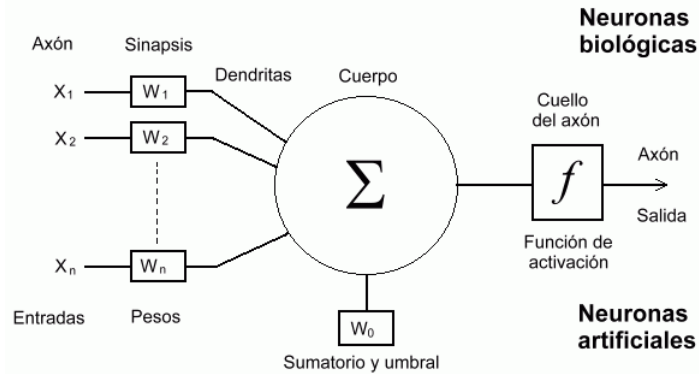
Una vez definidos los elementos que forman una neurona artificial estos se utilizan de manera que se utiliza la función:

$$\begin{aligned} f : \mathbb{R}^p &\longrightarrow \mathbb{R} \\ f(\mathbf{x}) &\longrightarrow f(\mathbf{x}; b, \omega) \end{aligned} \quad (1.3.1)$$

Donde

$$f(\mathbf{x}) = g\left(b + \sum_{i=1}^p \omega_i x_i\right) \quad (1.3.2)$$

El siguiente diagrama proporciona una forma sencilla de entender el funcionamiento de dicho modelo, incluyendo la analogía de las neuronas biológicas.



El potencial de este tipo de algoritmos es poder conectar neuronas unas con otras, de manera que la salida que produce una funcione de entrada para otra.

Definición 1.3.2. Se llama capa de neuronas a un conjunto de neuronas que tienen en común las señales de entrada y producen un vector de salidas que pueden o no servir como señal de entrada para otra.

La siguiente imagen es un esquema de una red neuronal con 7 capas de neuronas interconectadas donde la primera capa es de escalado y la última de desescalado. Se puede observar que se predicen las variables y_1, y_2, y_3 usando como entrada las variables x_1, x_2, x_3, x_4

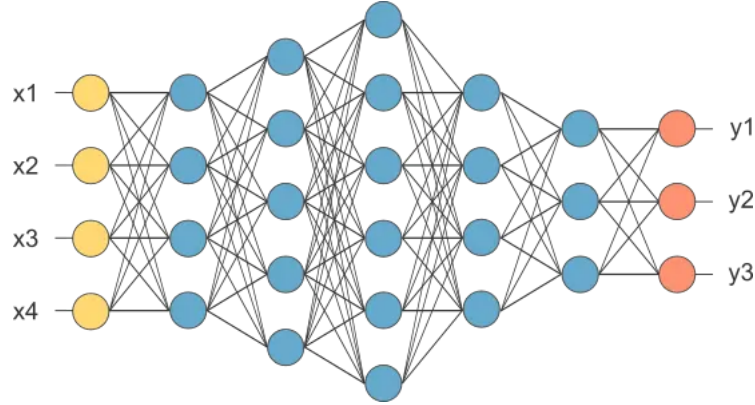


Figura 1.1: Imagen extraída directamente de www.neuraldesigner.com

1.3.1. Projection Pursuit Regression

Sean un vector aleatorio \mathbf{x} de longitud p , una variable objetivo Y y una familia de vectores de parámetros de longitud p , $\{\omega_m\}_{m=1}^M$. Entonces el modelo de Regresión por Búsqueda de Proyecciones (*PPR en inglés*) es de la forma :

$$f(\mathbf{x}) = \sum_{m=1}^M g_m(\omega_m^T \mathbf{x}) \quad (1.3.3)$$

En este modelo las funciones g_m no son especificadas y son estimaciones a lo largo de las direcciones de los vectores $\omega_m^T \mathbf{x}$. Según *Hastie, Tibshirani y Friedman* [9] para un M lo suficiente grande y utilizando las g_m apropiadas este modelo puede ayudar a la predicción de cualquier función continua real.

Para ajustar este tipo de modelos tenemos un ajuste en dos pasos, primero se estiman las funciones g_m en función de los parámetros ω_m del paso anterior y luego con las g_m estimadas se optimiza respecto de los parámetros ω_m . Teniendo en cuenta lo siguiente :

- Los métodos de obtención de las g_m deben producir funciones derivables.
- Se pueden ir actualizando tanto las g_m como los ω_m . Aún así, ω_m no se suele actualizar para evitar un gasto computacional excesivo.
- Se puede usar validación cruzada o ir comprobando si hay mejora para estimar la cantidad total de sumandos M .

El uso de este tipo de modelos suele estar restringido así como el de las redes neuronales a la predicción, ya que suelen producir modelos de alta complejidad y difícilmente interpretables.

Sobre este modelo se sustentan las redes neuronales, que establecen previamente las funciones de activación g_m .

En el caso de las redes neuronales, el método de ajuste varía al modelo previamente descrito ya que las funciones están prefijadas.

1.3.2. Red Neuronal de 2 capas

En pos de la sencillez de los resultados, se detallará la de una red neuronal de dos capas. El resto de casos $m \geq 2$, el proceso es análogo. Añadir que la situación es aquella en la que se quiere predecir K variables objetivo a partir de observaciones de p variables.

La primera capa tiene los siguientes elementos:

- Como datos de entrada cada una de las observaciones \mathbf{x} de tamaño p e incluimos en cada uno el término inicial $x_0 = 1$ de tal manera que hay $p + 1$ datos de entrada.
- Un total de M unidades lo que provocará un conjunto $\{z_m\}$ de datos de salida.
- Cada unidad de las M tiene unos pesos α_m de dimensión $p + 1$, donde la primera componente α_{m0} se denomina **sesgo**.
- Una función de activación (*Que puede o no ser lineal*) $g^{(1)}$.

De esta manera, tenemos que los datos de salida de esta primera capa son de la forma:

$$z_m = g^{(1)}(\alpha_m^T \mathbf{x}) \quad (1.3.4)$$

De manera análoga tenemos una segunda capa de neuronas con los siguientes elementos:

- Como datos de entrada los M resultados de la capa anterior, z_m al que añadimos $z_0 = 1$ y denotaremos como el vector \mathbf{z} de longitud $M + 1$.
- Un total de K unidades lo que provocará un conjunto $\{t_k\}$ de datos de salida.
- Cada unidad de las K , tiene unos pesos β_m de dimensión $M + 1$ igual que en la anterior.
- Una función de activación (*Que puede o no ser lineal*) $g^{(2)}$ que puede ser la misma que en la anterior o no.

De esta manera, tenemos que los datos de salida de esta primera capa son de la forma:

$$t_k = g^{(2)}(\beta_k^T \mathbf{z}) \quad (1.3.5)$$

Hay que destacar que los datos deben estar escalados, o pueden serlo mediante una capa que los escale per sé.

Definición 1.3.3. Sea una matriz de datos \mathbf{X} de tamaño $n \times p$, resultado de observar las variables $X_1 \dots X_p$, se llama *capa de escalado* a la capa de neuronas con los siguientes elementos:

- Un vector de pesos con una sola componente igual a 1 y el resto nulas.
- Una función de activación $g(z) = \frac{z - \mu_i}{\sigma_i}$ donde μ_i, σ_i son las media y desviación típica muestrales de cada variable.

El resultado de que la matriz de datos sea procesada por este tipo de capa es una matriz \mathbf{X}' centrada y estandarizada.

1.3.3. Ajuste y uso de una Red Neuronal

Una vez formulado el funcionamiento de una red neuronal, el ajuste de los pesos y sesgos de esta se puede plantear como un problema de minimización de la pérdida en función de los propios pesos, utilizando métodos de optimización numérica como el método del gradiente o el método de Quasi-Newton.

Hay que tener en cuenta para seleccionar el modelo, es decir, el número de neuronas y capas, que las redes neuronales son proclives al sobreajuste. Esto es debido a la gran cantidad de parámetros que se pueden tener en una red con una complejidad media. Teniendo en cuenta esta problemática, se introduce una penalización a términos muy grandes y hace que los pesos sean parecidos a 0.

Una vez ajustados los pesos el modelo resultante puede ser complejo de interpretar, pero proporciona predicciones que potencialmente pueden ser todo lo precisas que se requieran según el número de capas y neuronas que utilicemos.

En esta memoria se han detallado los tipos más simples de redes neuronales y de neuronas, con las cuales ya se obtiene una gran capacidad de predicción. Sin embargo, hay estructuras neuronales como las redes convolucionales o las capas LSTM cuya estructura (*Llamadas así por sus siglas en inglés Long-Short Term Memory*) que permiten modelizar sistemas en que los estados futuros son afectados por los estados anteriores como pudiera ser el precio de una acción bursátil o el tiempo atmosférico.

Para evaluar el rendimiento de una red neuronal se pueden utilizar técnicas de validación cruzada o separar el conjunto de entrenamiento en entrenamiento, validación y test. En este proceso, se puede también hacer comparaciones entre varios modelos para elegir el número de neuronas y de capas.

Para mejorar el rendimiento de la propia red neuronal se pueden aplicar técnicas previamente a los datos como el Análisis de Componentes Principales o seleccionar las variables más significativas como se detalla en la sección de regresión que trata de ello.

1.4. Árboles de Decisión y Bosques Aleatorios

Sea un vector aleatorio \mathbf{x} con p los datos de entrada, e Y la variable respuesta. Supóngase también que se toman n observaciones obteniéndose parejas (\mathbf{x}_i, y_i) . De esta manera, tenemos que las $\mathbf{x}_i \in \mathbb{R}^p$.

Los métodos de árboles son un método divisivo, ya que tras aplicarlos, se obtiene una partición del espacio de observaciones \mathbb{R}^p y luego en cada región del espacio se ajusta un modelo más simple, incluso una constante.

La ventaja de este tipo de métodos es que son fácilmente interpretables, ya que pueden ser representados mediante un diagrama de tipo árbol. De esta manera, pueden ser utilizados según *Brown et.al.*[5] tanto para análisis exploratorio, detectando características de grandes cantidades de datos, como para tareas de regresión y clasificación.

Las siguientes imágenes procedentes de *Hastie et. al.*[9] muestran el diagrama resultante tras dividir el espacio de observaciones mediante un árbol.

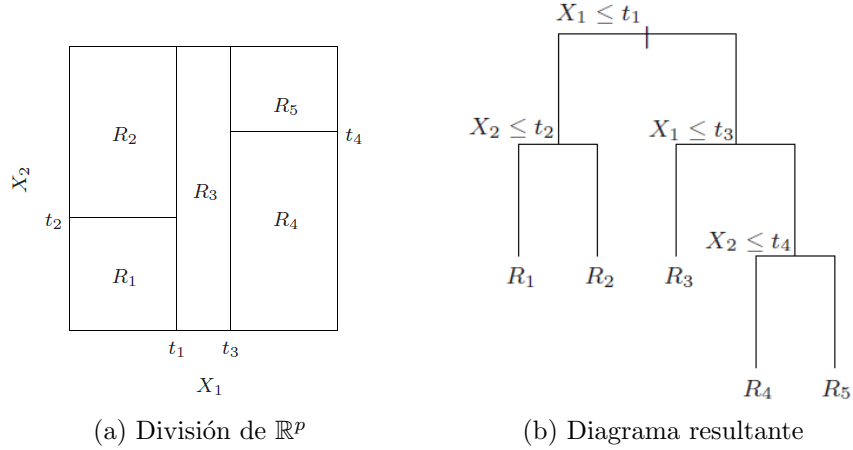


Figura 1.2: Representación de la división de \mathbb{R}^p y el diagrama de árbol resultante

Otra ventaja de este tipo de métodos es que permite trabajar con conjuntos de datos en los que se estudian más variables en comparación de las observaciones, es decir, $p > n$. Por ejemplo, el ejemplo que desarrollan *Díaz-Uriarte y De Andrés* [7], en el que trabajan con un microarray genético.

Otro de los problemas que surgen del propio planteamiento del modelo en sí es que el algoritmo para obtener las divisiones están diseñados para sobreajustarse a los datos por tanto, surgen métodos como la “poda”. Se desarrollará más adelante, pero el idea es poder modelizar toda la variación del conjunto de entrenamiento.

1.4.1. Árboles de Regresión

Sea un vector aleatorio \mathbf{x} de variables predictoras como antes y una variable Y de respuesta. Supóngase que queremos dividir el espacio de observaciones en M regiones, R_1, \dots, R_M , se puede modelar en cada una de las regiones como la constante c_m . Por tanto teniendo en cuenta, la siguiente definición.

$$\mathbf{1}_m(\mathbf{x}) = \begin{cases} 0 & \text{si } x \notin R_m \\ 1 & \text{si } x \in R_m \end{cases} \quad (1.4.1)$$

Es decir, la función $\mathbf{1}(\mathbf{x})$ es la función característica de la región R_m , por tanto puede construirse el siguiente predictor:

$$f(\mathbf{x}) = \sum_{m=1}^M c_m \mathbf{1}_m(\mathbf{x}) \quad (1.4.2)$$

Un problema que surge es que en principio cualquier tipo de región sería aceptable. Esta libertad llevaría a tener que explorar cualquier región posible, lo cual no es viable computacionalmente hablando. Es por ello, que se toma un tipo de división en particular, las divisiones binarias en semihiperplanos. De esta manera, se generan regiones que son rectángulos de altas dimensiones.

Si se toma como criterio de ajuste los mínimos cuadrados obtendremos que $\hat{c}_m = \text{ave}(y_i | x_i \in R_m)$, ya que es la media condicional de la y sabiendo el valor de x .

Para obtener las regiones se toma una variable X_j y un valor de separación s , es decir, cada separación se puede identificar con una pareja (j, s) . Por ejemplo, supóngase una separación (j, s) entonces se generan dos regiones del espacio de las observaciones R_1, R_2 .

$$R_1 = \{\mathbf{x} | X_j \leq s\} \quad R_2 = \{\mathbf{x} | X_j > s\} \quad (1.4.3)$$

Hay que establecer un criterio para elegir que separaciones se deben hacer en particular, para ello se utilizan los mínimos cuadrados:

$$\min_{j,s} \left[\min_{c_1} \sum_{x_i \in R_1} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2} (y_i - c_2)^2 \right] \quad (1.4.4)$$

Es decir, teniendo en cuenta que se va a predecir el valor de la variable respuesta y_i como una constante, la función anterior es la función RSS

Las mejores estimaciones de las constantes según el criterio de los mínimos cuadrados sería la media de las respuestas dentro de las regiones, es decir:

$$\hat{c}_m = \frac{1}{n_m} \sum_{\mathbf{x}_i \in R_m} y_i \quad (1.4.5)$$

De esta manera, con estas estimaciones de las constantes de cada región, es posible encontrar la mejor división del espacio de observaciones.

Se puede conseguir la mejor pareja (j, s) fijando la j -ésima variable, encontrar el mejor valor s para cada variable, de manera sencilla.

Definición 1.4.1. Diremos que un algoritmo es *voraz* o *greedy* si en cada paso local busca la solución óptima.

En cada paso, el árbol busca en cada paso la división que mayor disminución produce en el RSS , por tanto, se utiliza un algoritmo voraz.

El siguiente problema es controlar la complejidad del modelo, en particular, cuanto debe crecer el árbol, ya que un árbol demasiado grande puede pecar de problemas de sobre ajuste.

La estrategia más común es tener un criterio de parada, en particular hacer crecer un árbol grande T_0 y “podarlo” cuando en los nodos terminales tenga menos de 5 individuos según *Díaz-Uriarte y De Andrés* [7], además este es el valor por defecto que otorga la librería de **R**, `randomForest`. Si se hace más grande esto provoca que los árboles sean más pequeños.

Definición 1.4.2. Llamaremos tamaño de un árbol T y se denota por $|T|$ al número de nodos terminales, es decir, al número de regiones en las que se particiona el espacio de observaciones.

Se puede entonces definir el error cuadrático medio de un nodo de un árbol de la siguiente manera:

$$Q_m(T) = \frac{1}{n_m} \sum_{\mathbf{x}_i \in R_m} (y_i - \hat{c}_m)^2 \quad (1.4.6)$$

Con la anterior función y el concepto del tamaño del árbol, el cual está directamente relacionado con la complejidad del modelo, se puede establecer un criterio para un árbol que penalice la complejidad:

$$C_\alpha(T) = \sum_{m=1}^{|T|} n_m Q_m(T) + \alpha |T| \quad (1.4.7)$$

El valor de α es un parámetro que es elegido y su valor hace que se penalice más o menos la complejidad del modelo. A valores más altos, tendremos un modelo menos complejo y con $\alpha = 0$ es el modelo obtenido por mínimos cuadrados.

El objetivo es encontrar el árbol de menor tamaño para cada α que minimice la función (1.4.7). Para ello se empieza con el árbol completo o con un árbol inicial T_0 y se van juntando regiones de tal manera que el aumento provoquen sobre la función $\sum_{m=1}^{|T|} n_m Q_m(T)$, sea el menor posible, resultando en una secuencia de árboles entre los cuales se encuentra el que minimiza la función $C_\alpha(T)$ para un α determinado.

Para llevar a cabo el proceso completo, se deben aplicar procesos de validación cruzada, por ejemplo en *Divakaran S.* [19] o en *James G. et.al.* [20] se detalla el algoritmo de Breiman. En este algoritmo se usa la validación cruzada de K-folds, (*Para más detalles sobre dicho proceso, léase la sección 7.10 de Hastie et.al* [9] o la sección 5.1.3 *James et.al.* [20]) para poder elegir el α que minimice el error.

1.4.2. Árboles de Clasificación

Gran parte de lo anterior se puede detallar de manera análoga a los árboles de regresión. En esencia, un árbol de clasificación es un conjunto de funciones discriminantes utilizadas de manera recurrente obteniéndose en el caso óptimo una partición del espacio de observaciones en el que cada región contiene observaciones que pertenecen únicamente a una categoría.

A partir de este punto, la variable respuesta será una variable categórica con valores posibles $1, \dots, K$.

Sea un nodo m que representa una de las regiones R_m en las que se han particionado el espacio con n_m observaciones en ella. Entonces, la proporción de observaciones que son de la clase k -ésima en el nodo m es:

$$\hat{p}_{mk} = \frac{1}{n_m} \sum_{\mathbf{x}_i \in R_m} \mathbf{1}(y_i = k) \quad (1.4.8)$$

En cada región R_m se tomará como predicción la respuesta cuya \hat{p}_{mk} sea máxima.

Una vez definido esto se ha de definir el concepto de pureza de un nodo

Definición 1.4.3. Diremos que un nodo es *puro u homogéneo* si todas las observaciones que pertenecen a la región establecida por dicho nodo pertenecen a la misma clase. En caso contrario se dice que el nodo es *impuro*

Es este concepto de la impureza lo que permite establecer un criterio para realizar las divisiones del espacio. Es decir, en cada paso se elegirá la división (j, s) que produzca los nodos más puros.

Teniendo en cuenta la definición que dimos de \hat{p}_{mk} se pueden dar distintas medidas de la impureza de un nodo. Como detallan tanto *Hastie et.al*[9] como *Brown et.al*. [5]

Definición 1.4.4. Sea un nodo y su región correspondiente, R_m , sea además k el valor para el cual \hat{p}_{mk} es mayor que el resto, se define el *Error de Clasificación errónea* como:

$$\frac{1}{n_m} \sum_{\mathbf{x}_i \in R_m} \mathbf{1}(y_i \neq k) = 1 - \hat{p}_{mk} \quad (1.4.9)$$

Es la proporción de observaciones en el nodo que no tienen la misma categoría que el más frecuente. Esta medida no es la más usada debido a que no es derivable y no es muy sensible a los cambios.

Por otro lado, el *índice Gini* no solo tiene en cuenta la proporción máxima o la categoría máxima sino que da una medida de la varianza total entre las K categorías posibles. Lo cual arroja una mejor visión de la pureza de cada nodo.

Definición 1.4.5. Se llama *índice Gini* a la siguiente medida:

$$G = \sum_{k=1}^K \hat{p}_{mk}(1 - \hat{p}_{mk}) \quad (1.4.10)$$

Otra medida alternativa sería la *entropía* del nodo que se define

Definición 1.4.6. Se llama *entropía* de un nodo m o de una región R_m a:

$$D = - \sum_{k=1}^K \hat{p}_{mk} \log(\hat{p}_{mk}) \quad (1.4.11)$$

Ambas medidas son bastante similares.

El algoritmo para construirlos es análogo al de los árboles de regresión. Tomando la medida de impureza o pureza que se desee se puede construir un árbol de clasificación teniendo en cuenta que el error de clasificación produce de manera general según *James et.al.* [20] árboles con una precisión mayor.

Aunque en la mayoría de casos según *Brown et.al*[5] lo que ocurre al utilizar distintas medidas es que los modelos no obtienen errores de predicción muy alejados, sin embargo, pueden provocar que los modelos tengan complejidades muy diferentes.

1.4.3. Bosques Aleatorios

Los bosques aleatorios y otros métodos de juntar árboles de decisión son maneras de solventar ciertos problemas que tienen los árboles de decisión individuales como puede ser la precisión o la dependencia del conjunto de datos, ya que una pequeña variación en estos puede provocar grandes cambios.

Durante esta sección no se detallarán los métodos de construcción de cada árbol individual ya que se han descrito anteriormente y se hará referencia a ellos como árboles simplemente.

Estos métodos se basan en el Bagging y el Bootstrap, ambos métodos se pueden utilizar cuando la cantidad de datos es pequeña o cuando el modelo sufre de *sobreajuste*.

El proceso de Bagging tiene como base el hecho de que si se tienen n observaciones independientes de una población con varianza σ^2 entonces la media de dichas observaciones tendrá varianza $\frac{\sigma^2}{n}$, es decir, tomar la media de distintas observaciones provoca una disminución en la varianza.

De esta manera, partiendo de un conjunto inicial de datos, se toman muestras aleatorias de los mismos y se entrenan distintos modelos para luego hacer la media entre todos los modelos. Es decir, se hace un muestreo aleatorio entre las n observaciones obteniendo B conjuntos de entrenamiento distintos y se estiman los predictores $\hat{f}_1(\mathbf{x}), \dots, \hat{f}_B(\mathbf{x})$ y el modelo final es:

$$\hat{f}_{bag}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B \hat{f}_b(\mathbf{x}) \quad (1.4.12)$$

Esto se puede aplicar perfectamente al caso de la modelización mediante árboles de decisión aunque esta técnica puede ser útil para cualquier tipo de modelos.

En este caso, estamos utilizando en cada división de los árboles todos los predictores, sin embargo, cuando una variable tiene demasiada importancia a la hora de predecir la variable de respuesta, la mayoría de divisiones en casi todos los árboles

van a utilizar ese predictor, ya que va a provocar mejoras sustanciales en el *RSS* o en el *índice Gini* dependiendo del tipo de árbol.

Para evitar dicho problema, en el algoritmo del bosque aleatorio se toma en cada división no el total de predictores, si no una muestra aleatoria de tamaño m que habitualmente es $m = \sqrt{p}$ de esta manera, se puede tener modelos con mayor variedad.

En *Biau G. y Scornet E.*[3] se detalla el algoritmo para la creación de un bosque aleatorio. En este algoritmo se deben definir los siguientes parámetros:

- \mathcal{M} : Número de árboles que se incluirán en el bosque aleatorio.
- m_{try} , es el número de predictores que entre los cuales se pueden usar en cada división del espacio.
- a_n : Número de observaciones del conjunto de entrenamiento que se utilizarán para entrenar cada árbol.

El algoritmo en sí es el siguiente:

for $j = 1 \dots \mathcal{M}$ **do**:

- Se extrae de manera uniforme un subconjunto de tamaño a_n del conjunto de observaciones de entrenamiento.
 - Se extrae aleatoriamente un conjunto de predictores posibles \mathcal{M}_{try} de tamaño m_{try} y se construye el árbol, utilizando como posibles predictores únicamente los que están en el conjunto \mathcal{M}_{try}
-

Para hacer la predicción para una nueva observación \mathbf{x}_0 , hay que calcular la predicción para cada uno de los \mathcal{M} árboles. Una vez calculadas en el caso de que la variable respuesta sea numérica se hace la media de todas las predicciones. Si por el contrario, Y es una variable categórica, se toma como predicción aquella que haya salido con mayor frecuencia, lo que se llama predicción por *voto de mayoría*.

En el artículo de *Biau, G y Scornet, E.* [3] se detallan más en profundidad las distintas características de estos modelos. En particular, se detalla la necesidad de utilizar modelos simplificados para poder analizar las características de los bosques aleatorios. Estos pueden ser los bosques puramente aleatorios que hace particiones totalmente aleatorias sin tener un criterio de acuerdo a los datos y otras modificaciones.

En particular, *Breiman L.* [4], utiliza los siguientes resultados empíricos para poder analizar la consistencia y el ratio de convergencia de los bosques aleatorios utilizando un modelo más simplificado de los mismos.

- El muestreo aleatorio que se hace del conjunto de entrenamiento no tiene efecto en el ratio de error.
- Si para conseguir una partición en la que únicamente en cada nodo haya una sola observación se utiliza la mediana de las variables aleatorias esto no tiene efecto en el ratio de error.

- Hacer divisiones sin tener en cuenta las clases que conocemos, sí aumenta el error.

De esta manera, establece un modelo simplificado en el las variables predictoras se pueden dividir en dos grupos, las variables fuertes \mathcal{S} y las variables débiles \mathcal{W} . Las variables fuertes son las que tienen una influencia directa sobre las predicciones y las débiles se pueden interpretar como ruido.

Teniendo en cuenta lo anterior, las variables aleatorias fuertes se dividen en la mediana, mientras que si es débil se toma un valor aleatorio para la división.

Como resultado de dichas modificaciones *Breiman, L.* [4] concluye que el ratio de convergencia es $\mathcal{O}\left(n^{\left(\frac{-0,75}{|\mathcal{S}|\log 2+0,75}\right)}\right)$.

Por tanto, el ratio de error depende del número de variables fuertes.

Capítulo 2

Métodos no supervisados

Los métodos no supervisados son aquellos que utilizan la matriz de datos en la cual no se tiene un conocimiento previo, es decir, no intenta ajustarse a una variable respuesta conocida. Esto provoca que no se tenga una *función de pérdida* en el sentido que se ha definido anteriormente [9].

El objetivo de la mayoría de métodos no supervisados es estudiar la estructura de los datos, con el objetivo de encontrar información subyacente a los datos. En este proceso, se pueden observar distintas características, como puede ser la variabilidad del conjunto de los datos, la homogeneidad de las observaciones etc...

Al estudiar la estructura de los datos y dar los puntos comunes a todos los datos, se puede llevar a cabo una reducción de la dimensionalidad, es decir, nuestro problema puede pasar a explicarse con menos variables de manera que se simplifica la compresión [1] y a veces los sucesivos análisis.

En particular métodos como el Análisis factorial o el Análisis de componentes principales, permiten en casos en los que las ciertas variables estén muy correladas [35], crear nuevas variables que representen la misma información o gran parte de ella, en el caso de las componentes la variabilidad general y en el caso de los factores la variabilidad común a todas las variables. En cambio en el análisis de clústers centra el foco la variabilidad de las observaciones, teniendo en cuenta la homogeneidad dentro de los clusters y heterogeneidad entre clusters distintos, de esta manera, se pueden identificar las distintas observaciones como el cluster al que pertenecen.

Aunque no se tenga una función de pérdida como se daba en el caso de los métodos supervisados, se puede dar en estos casos también una medida de la bondad de ajuste que suele depender de la característica observada en cada uno de los casos. Esto puede ser en el caso de los factores la variabilidad común explicada [22], o la proporción de variabilidad explicada en cada componente [6].

Por falta de tiempo y de extensión del trabajo, no se desarrollarán técnicas como las reglas de asociación, el escalado multidimensional para los cuales *Hastie et.al.* aporta una visión introductoria a los mismos [9]. Por otro lado *James et.al.*[20] desarrolla de manera similar con la particularidad que además lo implementa en el entorno R con ejemplos. (Veánse también [35] y [34]).

2.1. Análisis de Componentes Principales

El Análisis de componentes principales, es una de las técnicas multivariantes más antiguas. En primera instancia, esta técnica fue desarrollada en paralelo por *Pearson, K.* [42] y *Hotelling* [43] con distintos enfoques.

Pearson, K.[42] en su aproximación, buscó la forma de ajustar de mejor manera puntos de un espacio de dimensión p a una recta o plano. Es decir, Pearson buscó un enfoque de optimización geométrica que llevó a las componentes principales.

Por el otro lado, *Hotelling, H.* [43] buscaba un conjunto de variables menor que pudiera determinar de igual manera o parecida los datos. Hotelling maximizó la contribución que aportaba cada componente a la varianza. Al utilizar para este fin el método de los multiplicadores de Lagrange, obtuvo el problema de valores propios que se desarrolla en esta sección. Aún así, el trabajo de Hotelling tenía ciertas diferencias con el enfoque que actualmente se le da actualmente a las componentes principales [10], por ejemplo, consideraba que las variables originales debían ser combinaciones de las componentes y no al revés. Tampoco usaba la notación matricial ni la matriz de covarianzas, en su lugar utilizaba la de correlaciones. Añadir que aunque en esta memoria se trabajará con la matriz de covarianza, *Chatfield, C. y Collins A.J.*[6] desarrollan ambos enfoques.

Según *Abdi, H* [1] esta técnica tiene varios objetivos, entre los cuales están el extraer la información más importante de los datos, reduciendo la dimensionalidad y simplificando la descripción en el proceso de los mismos. Y por último, analizar la estructura de los datos.

De esta manera, variables bastante correladas, se simplificarán en una que tenga toda la variabilidad de ambas [6], [35] ya que a fin de cuentas la información que expresan estaría repetida.

En esta sección se detallará en principio cómo se calculan las componentes principales solo teniendo en cuenta las propiedades del vector aleatorio \mathbf{x} de longitud p . Tras esto, tomaremos N observaciones con las que poder construir estimadores de las componentes y se analizará como esta reducción de la dimensionalidad es la que permite una mejor reconstrucción de los datos mediante el Teorema de Eckart-Young [38].

Para desarrollar esta última parte en la que se habla de la reconstrucción de los datos se da una idea de una medida sobre las matrices, en particular, la norma de Frobenius que detallan *Eckart C. y Young, G* [38] sin llamarla de esa manera y demuestran que es invariante ante transformaciones ortogonales.

Y por último, se detallan varios ejemplos para ilustrar la interpretación de los resultados obtenidos tras aplicar las técnicas detalladas y hallar ciertos parámetros de bondad de ajuste, además de mostrar aplicaciones reales de este método.

2.1.1. Definición y cálculo de las Componentes

Sea un vector aleatorio de longitud p , $\mathbf{x} = [X_1, \dots, X_p]$ con una distribución normal p -multivariante $N_p(\mu, \Sigma)$

Definición 2.1.1. *Cuadras C.M.,[8]* define las componentes principales

$$Z_j = v_{1j}X_1 + \dots v_{pj}X_p = \mathbf{v}_j^T \mathbf{x}^T \quad j = 1 \dots p \quad (2.1.1)$$

Donde \mathbf{v}_j es un vector columna con p escalares y la nueva variable aleatoria Z_j cumple lo siguiente:

- Si $j = 1$ $Var(\mathbf{z}_1)$ es máxima restringido a $\mathbf{v}_1^T \mathbf{v}_1 = 1$
- Si $j > 1$ debe cumplir:
 - $Cov(\mathbf{z}_j, \mathbf{z}_i) = 0 \quad \forall i \neq j$
 - $\mathbf{v}_j^T \mathbf{v}_j = 1$
 - $Var(Z_j)$ es máxima.

De esta manera lo que se busca es una nueva base que reúna las direcciones de máxima variación y que sean ortogonales respecto a la matriz de covarianzas, es decir, que sean no correladas.

Chatfield C. y Collins A.J [6], utilizan el método de los multiplicadores de Lagrange para resolver el problema de maximizar $Var(Z_1)$ sujeto a la restricción $\mathbf{v}_1^T \mathbf{v}_1 = 1$. Todo esto con el objetivo de calcular el vector \mathbf{v}_1 , el cálculo de sucesivas componentes cambia en ciertos aspectos.

Aplicando el método de los multiplicadores, la función objetivo es la varianza de la combinación lineal, es decir, $f(\mathbf{x}) = \mathbf{x}^T \Sigma \mathbf{x}$ y la restricción aplicada es $g(\mathbf{x}) = \mathbf{x}^T \mathbf{x} = 1$.

Tomando $\mathbf{x} = \mathbf{v}_1$ se puede establecer $L(\mathbf{v}_1) = \mathbf{v}_1^T \Sigma \mathbf{v}_1 - \lambda[\mathbf{v}_1^T \mathbf{v}_1 - 1]$. Que al derivarla se obtiene:

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{v}_1} &= 2\Sigma \mathbf{v}_1 - 2\lambda \mathbf{v}_1 \\ &= 2(\Sigma - \lambda I) \mathbf{v}_1 \end{aligned}$$

Igualando a 0 tenemos la siguiente ecuación:

$$(\Sigma - \lambda I) \mathbf{v}_1 = 0 \quad (2.1.2)$$

Para que \mathbf{v}_1 sea un vector no trivial, se elige λ de tal manera que $|\Sigma - \lambda I| = 0$, es decir, λ es un vector propio de la matriz de covarianzas, Σ . Al ser ésta una matriz semidefinido positiva y simétrica, los valores propios son reales y positivos. Por tanto, \mathbf{v}_1 es un vector propio de la matriz de covarianza.

La función a maximizar es $Var(Z_1) = Var(\mathbf{v}_1^T \mathbf{x}) = \mathbf{v}_1^T \Sigma \mathbf{v}_1 = \mathbf{v}_1^T \lambda \mathbf{v}_1 = \lambda$, y para maximizarla basta tomar $\lambda = \max \{\lambda_1 \dots \lambda_p\}$. Reordenando si es necesario, se tiene

que $\lambda = \lambda_1$, por tanto la primera componente es el vector propio Z_1 y además la varianza de la nueva variable cumple $Var(Z_1) = \lambda_1$

Una vez calculada la primera componente principal Z_1 , la segunda componente se calcula de manera análoga, maximizando $Var(Z_2) = Var(\mathbf{v}_2^T \mathbf{x}^T)$ condicionada por $\mathbf{v}_2^T \mathbf{v}_2 = 1$. A esta restricción tenemos que añadir la restricción $Cov(Z_1, Z_2) = 0$

Proposición 2.1.1. La condición $Cov(Z_1, Z_2) = 0$ equivale a la condición $\mathbf{v}_2^T \mathbf{v}_1 = 0$.

Demostración. Utilizando que $Z_j = \mathbf{v}_j^T \mathbf{x}^T \quad \forall j = 1, \dots, p$, se tiene entonces que :

$$\begin{aligned} Cov(Z_2, Z_1) &= Cov(\mathbf{v}_2^T \mathbf{x}^T, \mathbf{v}_1^T \mathbf{x}) \\ &= \mathbb{E}(\mathbf{v}_2^T (\mathbf{x}^T - \mu^T) (\mathbf{x}^T - \mu^T)^T \mathbf{v}_1) \\ &= \mathbf{v}_2^T \mathbb{E}((\mathbf{x}^T - \mu^T) (\mathbf{x}^T - \mu^T)^T) \mathbf{v}_1 \\ &= \mathbf{v}_2^T \Sigma \mathbf{v}_1 \\ &= \mathbf{v}_2^T \lambda_1 \mathbf{v}_1 \end{aligned}$$

De manera que, si $\mathbf{v}_2^T \lambda_1 \mathbf{v}_1 = 0 \Rightarrow \mathbf{v}_2^T \mathbf{v}_1 = 0$, luego son vectores ortogonales entre sí. \square

Observación: Esta proposición se puede extender de manera simple al caso de tener que calcular la j -ésima componente principal habiendo calculado las anteriores de las cuales se sepan los valores propios asociados.

Corolario 2.1.1. Las componentes principales son todas ortogonales entre sí.

Para $k = 2$, se dan dos restricciones, $\mathbf{v}_2^T \mathbf{v}_2 = 1$ y además $\mathbf{v}_1^T \mathbf{v}_2 = 0$. Para este caso existen λ, ϕ de manera que la función a maximizar es:

$$L(\mathbf{v}_2) = \mathbf{v}_2^T \Sigma \mathbf{v}_2 - \lambda[\mathbf{v}_2^T \mathbf{v}_2 - 1] - \phi(\mathbf{v}_1^T \mathbf{v}_2) \quad (2.1.3)$$

Que al ser derivado respecto \mathbf{v}_2 obtenemos:

$$2\Sigma \mathbf{v}_2 - 2\lambda \mathbf{v}_2 - \phi \mathbf{v}_1 = 0 \quad (2.1.4)$$

Que al multiplicar todo por \mathbf{v}_1^T resulta que $\phi = 0$. De esta manera, se obtiene en la ecuación lo mismo que en el cálculo de la primera.

Por tanto, $\lambda = \lambda_2$ que es el segundo valor propio más grande, y \mathbf{v}_2 es el vector propio de valor propio λ_2 .

El proceso para calcular el resto de componentes principales es análogo únicamente hay que tener en cuenta que se debe dar la ortogonalidad entre las distintas componentes respecto de la varianza.

Por ende, se obtiene que las componentes principales vienen dadas por los vectores propios de la matriz de covarianzas Σ . Además sabemos que $Var(\mathbf{v}_j^T \mathbf{x}^T) = \lambda_j$, $j = 1, \dots, p$, donde λ_j es el j -ésimo valor propio más grande.

En esencia, calcular las componentes principales es calcular una base ortonormal que cumple una ciertas condiciones. Por lo tanto, podemos definir lo siguiente:

Definición 2.1.2. Se llama matriz de cargas \mathbf{V} a la matriz ortogonal que tiene por columnas a los \mathbf{v}_j . Dichas columnas forman una base ortonormal del espacio \mathbb{R}^p .

Observación: Se utilizará la condición de que al ser \mathbf{V} una matriz ortogonal $\mathbf{V}\mathbf{V}^T = \mathbf{V}^T\mathbf{V} = \mathbf{I}$ y por tanto $\mathbf{V}^{-1} = \mathbf{V}^T$

Esa matriz nos permite calcular las componentes de la siguiente manera

$$\mathbf{z} = \mathbf{x}\mathbf{V} \quad (2.1.5)$$

Donde el vector $\mathbf{z} = Z_1, \dots, Z_p$, y donde cada $Z_j = \mathbf{v}_j^T \mathbf{x}$

2.1.2. PCA muestral

Ahora, veamos como esto se puede aplicar a una matriz de datos con la matriz de cuasivarianzas muestral o con la matriz de covarianzas.

Sea \mathbf{x} el vector aleatorio de longitud p definida de la misma manera. A continuación se toman N observaciones independientes de este vector y se obtiene la matriz de datos \mathbf{X} de tamaño $N \times p$, en estos casos las componentes principales se definen de la misma manera.

Definición 2.1.3. Dado el vector aleatorio de longitud p del cual se han realizado N observaciones entonces se puede definir la j -ésima componente de la i -ésima observación $\mathbf{z}_{ij} = \mathbf{x}_i \mathbf{v}_j, i = 1 \dots N, j = 1, \dots p$.

Donde el vector \mathbf{v}_j cumple las mismas condiciones que para las variables aleatorias.

Por tanto, el proceso que se detalla para un vector aleatorio \mathbf{x} con matriz de covarianzas Σ se puede extender a este caso en el conocemos la matriz de covarianzas muestrales \mathbf{S} .

Con el objetivo de hacer las demostraciones más sencillas y compactas tomaremos la matriz \mathbf{X} como la matriz centrada $\bar{\mathbf{X}}$, es decir:

$$\bar{x}_{ij} = x_{ij} - \bar{x}_j \quad (2.1.6)$$

Donde \bar{x}_j es la media muestral de la j -ésima variable. Esto hace que $\mathbf{S} = \frac{1}{n-1} \mathbf{X}^T \mathbf{X}$. Esto permite hablar de los valores y vectores propios de \mathbf{S} y de $\mathbf{X}^T \mathbf{X}$ indistintamente, ya que los vectores propios son los mismos y los valores propios son proporcionales.

2.1.3. Reconstrucción de los datos

Lo más importante de esta técnica es que si la varianza se puede explicar por las m primeras componentes, entonces se puede afirmar que la dimensionalidad efectiva de los datos es $m < p$ [6].

Otra visión del problema es que se quiere buscar una proyección sobre una sub-variedad de dimensión $m < p$ que contenga la máxima variabilidad posible de los datos y que brinde una mayor capacidad de interpretación de los datos, ya que en el caso de que $m = 2$ o $m = 3$ se podrán hacer representaciones gráficas de manera sencilla. En virtud de conseguir esto se deben definir los siguientes conceptos:

Definición 2.1.4. Dada una matriz $\mathbf{X} \in \mathbb{M}_{N \times p}(\mathbb{R})$ existe la descomposición en valores singulares (*SVD en inglés*)[1]:

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T \quad (2.1.7)$$

Donde:

- \mathbf{U} matriz ortogonal y de tamaño $N \times N$
- \mathbf{D} matriz de tamaño $N \times p$ diagonal, cuyos elementos no nulos son los valores singulares $\sigma_1 \geq \dots \geq \sigma_r \geq 0$ que son los valores propios de la matriz $\mathbf{X}^T\mathbf{X}$ y $r = \text{rg}(\mathbf{X})$
- \mathbf{V} matriz ortogonal y de tamaño $p \times p$.

Observación: En lo restante, se considerará que $N > p$ y que el rango de la matriz de datos es máximo, es decir, que el rango es p . En el caso contrario, se puede reducir la matriz \mathbf{D} a una matriz de tamaño $r \times p$, con las columnas no nulas y por otro lado la matriz \mathbf{U} de tamaño $N \times r$ a la que se le pueden quitar las columnas correspondientes a los vectores propios de valor propio nulo.

Proposición 2.1.2. La matriz \mathbf{V} de tamaño $(p \times p)$ es la matriz que contiene los vectores para hacer la combinación lineal que definen las componentes principales. Es decir es la matriz con la base de diagonalización de $\mathbf{X}^T\mathbf{X}$ [39].

Demostración. La matriz $\mathbf{X}^T\mathbf{X}$ es la matriz de covarianzas $(p \times p)$ por la descomposición en valores singulares tenemos que:

$$\begin{aligned} \mathbf{X}^T\mathbf{X} &= (\mathbf{U}\mathbf{D}\mathbf{V}^T)^T(\mathbf{U}\mathbf{D}\mathbf{V}^T) \\ &= \mathbf{V}\mathbf{D}^T\mathbf{U}^T\mathbf{U}\mathbf{D}\mathbf{V}^T \\ &= \mathbf{V}\mathbf{D}^T\mathbf{D}\mathbf{V}^T \end{aligned}$$

Donde la matriz $\mathbf{D}^T\mathbf{D}$ es una matriz diagonal de tamaño $p \times p$ cuyos elementos son los cuadrados de los valores singulares de \mathbf{X} , que son a su vez los valores propios de $\mathbf{X}^T\mathbf{X}$.

Añadiendo la condición de ortogonalidad de $\mathbf{V} \Rightarrow \mathbf{V}^{-1} = \mathbf{V}^T$ es fácil ver que la matriz \mathbf{V} es la matriz cuyas columnas son los vectores propios de $\mathbf{X}^T\mathbf{X}$. \square

Proposición 2.1.3. La matriz \mathbf{U}^T es la matriz que contiene la base de diagonalización de la matriz $\mathbf{X}\mathbf{X}^T$ [39].

Demostración. Se sigue un razonamiento análogo pero con la matriz $\mathbf{X}\mathbf{X}^T$

$$\begin{aligned} \mathbf{X}\mathbf{X}^T &= (\mathbf{U}\mathbf{D}\mathbf{V}^T)(\mathbf{U}\mathbf{D}\mathbf{V}^T)^T \\ &= \mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{V}\mathbf{D}^T\mathbf{U}^T \\ &= \mathbf{U}\mathbf{D}\mathbf{D}^T\mathbf{U}^T \end{aligned}$$

Entonces, tenemos que como es ortonormal $\mathbf{U}^T = \mathbf{U}^{-1}$ de esta manera, se puede comprobar al ser $\mathbf{D}\mathbf{D}^T$ una matriz diagonal $N \times N$ que efectivamente se cumple la proposición. \square

Corolario 2.1.2. El cálculo de las componentes principales mediante los valores propios de la matriz de datos $\mathbf{X}^T \mathbf{X}$ es equivalente a calcular la descomposición en valores singulares de la misma.

El problema de la reconstrucción es demostrar que la matriz reducida, que se definirá más tarde, es la mejor aproximación de rango menor a la matriz de datos, para ello, tenemos que definir la siguiente norma sobre las matrices adecuada a la situación.

Definición 2.1.5. Sea $\mathbf{A} \in \mathbb{M}_{N \times p}(\mathbb{R})$ definimos la *norma de Frobenius* de la matriz \mathbf{A} como [40]:

$$\|\mathbf{A}\|_F = (tr(\mathbf{A}^T \cdot \mathbf{A}))^{\frac{1}{2}} = \left(\sum_{i=1}^N \sum_{j=1}^p a_{ij}^2 \right)^{\frac{1}{2}} \quad (2.1.8)$$

Proposición 2.1.4. La norma de Frobenius es invariante a transformaciones ortogonales

Demostración. Sea \mathbf{U} una matriz ortogonal, que cumple $\mathbf{U}^T \cdot \mathbf{U} = \mathbf{U} \cdot \mathbf{U}^T = \mathbf{I}$, sea una matriz cualquiera \mathbf{A} , entonces:

$$\begin{aligned} \|\mathbf{U} \cdot \mathbf{A}\|_F^2 &= tr((\mathbf{U}\mathbf{A})^T \cdot (\mathbf{U}\mathbf{A})) \\ &= tr((\mathbf{A}^T \mathbf{U}^T) \cdot \mathbf{U}\mathbf{A}) \\ &= tr(\mathbf{A}^T \mathbf{A}) \\ &= \|\mathbf{A}\|_F^2 \end{aligned} \quad \square$$

Se ha elegido la norma de Frobenius en particular por la siguiente propiedad:

Proposición 2.1.5. Dada una matriz de datos \mathbf{X} de tamaño $N \times p$ entonces

$$\|\mathbf{X}\|_F^2 = (N-1) \sum_{j=1}^p s_{jj}^2 \quad (2.1.9)$$

Donde las s_{ii}^2 son las varianzas muestrales.

Demostración. Debido a la centralidad impuesta a la matriz \mathbf{X} , sabemos que la matriz de covarianzas es $\mathbf{S} = \frac{1}{n-1} \mathbf{X}^T \mathbf{X}$ por tanto, se tiene que utilizar la definición de la norma:

$$\begin{aligned} \|\mathbf{X}\|_F^2 &= tr(\mathbf{X}^T \mathbf{X}) \\ &= (N-1) tr(\mathbf{S}) \\ &= (N-1) \sum_{i=1}^p s_{ii}^2 \end{aligned} \quad \square$$

Por tanto, la norma de Frobenius da una imagen del tamaño de la matriz de datos en función de la varianza total de los datos, lo que concuerda con la idea de buscar una matriz que aproxime la matriz de datos con la mínima pérdida de variación de los datos.

Definición 2.1.6. Se llama matriz reducida de orden $m \leq p$ de \mathbf{X} y se denota como \mathbf{X}_m , a la matriz $N \times m$ resultado de:

$$\mathbf{X}_m = \mathbf{U}_m \mathbf{D}_m \mathbf{V}_m^T \quad (2.1.10)$$

Donde:

- \mathbf{U}_m matriz ortogonal de tamaño $N \times m$, resultado de tomar de \mathbf{U} únicamente la matriz las m primeras columnas.
- \mathbf{D}_m matriz cuadrada de tamaño m diagonal con los m primeros valores singulares.
- \mathbf{V}_m matriz ortogonal de tamaño $p \times m$ obtenida al tomar las m primeras columnas de \mathbf{V} .

Es decir, es la matriz de rango m tras tomar las m primeras columnas y el resto hacerlas nulas o combinaciones lineales, en este caso.

Teorema 2.1.1 (De Eckart-Young). Sea \mathbf{A} una matriz de coeficientes reales de tamaño $N \times p$ y rango r entonces se cumple que [38, 40]:

$$\|\mathbf{A} - \mathbf{B}\|_F \leq \|\mathbf{A} - \mathbf{A}_m\|_F \quad \forall \mathbf{B} / \text{rg}(\mathbf{B}) = m \leq r \quad (2.1.11)$$

Johnson R.M. [39] desarrolla la demostración del teorema.

Por tanto, la matriz reducida brinda la mejor aproximación de la matriz de datos teniendo un criterio de aproximación basado en la variación de los datos. En consecuencia se puede

Como conclusión, se puede definir un criterio para elegir el orden de la matriz reducida m . Se puede entonces definir la variación acumulada de la siguiente manera [6]:

$$t_m = \frac{\sum_{i=1}^m \lambda_i}{\sum_{i=1}^p \lambda_j} \quad (2.1.12)$$

Donde los λ_i son los valores propios de la matriz $\mathbf{S} = \frac{1}{N-1} \mathbf{X}^T \mathbf{X}$.

Ahora la selección del número de componentes m que se van a seleccionar se puede hacer de varias maneras [22], [10], ya sea representando en un gráfico el número de componentes frente a la proporción de variabilidad explicada $t_j, j = 1, \dots, p$. Tomar hasta la primera componente con la cual se supere una cota de variabilidad explicada como el 0,8 o 0,9 [10] o desechar las componentes que no expliquen más de una cierta cota inferior. Sin embargo, estas reglas o condiciones suelen ser arbitrarias.

Todos estas condiciones se suelen resumir en desechar las componentes que no aporten a la representación.

Los valores de la matriz de cargas se pueden interpretar como las contribuciones de cada una de las variables a explicar la mayoría de la variabilidad de los datos.

Johnson, R. A., y Wichern, D. W [34] desarrollan un ejemplo sencillo en el cual se estudian 5 variables socioeconómicas de ciertas zonas de Estados Unidos, como

puede ser la población en miles, el porcentaje de graduados universitarios, el porcentaje de empleo de mayores de 16 años, el porcentaje de funcionariado y la valor medio de la vivienda.

En este ejemplo, las dos primeras componentes explican el 93 % de la variabilidad y la primera es una diferencia ponderada de el empleo y el empleo público, mientras que la segunda una suma ponderada de las mismas. Por tanto, se puede entender que las variables que mayor importancia tienen en este caso son el porcentaje de funcionarios y de empleados en general.

Otro ejemplo destacable lo desarrolla *Ringnér, W.*[36] en el que se reduce un conjunto de datos con más de 8000 variables y 105 observaciones a uno de solo 63 variables para retener el 90 % y 105 para retener casi el 100 % de la variabilidad total de los datos. Es decir, no siempre se se busca la reducción para representar los datos, en este ejemplo únicamente se busca eliminar datos redundantes en el conjunto total.

Otro ejemplo de aplicación sería el que desarrollan *Gottumukkal, R., y Asari, V.* [37] que mediante una pequeña modificación del algoritmo habitual de las componentes principales en el campo del reconocimiento facial con el objetivo de adaptarse mejor a ciertas condiciones, en las que se tienen grandes variaciones de posición y expresiones. Esto lo consiguen dividiendo cada foto y aplicando de manera independiente a cada parte la técnica.

2.2. Análisis Factorial

En primera instancia, *Galton*, intentó ver como una causa común podía afectar a dos variables aleatorias o medidas [44].

Fue el psicólogo inglés Charles Spearman el que tomó la idea de estudiar como una causa común podía afectar a un conjunto de variables. Es por ello que en 1904 se publica el primer artículo en el que aparece la palabra factor como la conoceremos en esta sección. Pero no se da ningún método específico para calcularlos.

El análisis factorial es una técnica de análisis multivariante no supervisado que analiza la covarianza con el objetivo de otorgar un conjunto de nuevas variables que expliquen la mayoría de la variabilidad común de los datos llamados factores, además de calcular la variabilidad que es específica de cada una de las variables estudiadas.

2.2.1. Formalización

Supóngase un vector aleatorio \mathbf{x} de longitud p con una distribución $N_p(0, \Sigma)$, centrada sin pérdida de generalidad, donde la matriz de covarianzas Σ es simétrica y definido positiva. El modelo factorial afirma lo siguiente [6]:

$$X_j = \lambda_{1j}F_1 + \dots + \lambda_{mj}F_m + \psi_j U_j \quad j = 1 \dots p \quad (2.2.1)$$

Donde:

- λ_{jk} son las *cargas del factor común* k -ésimo de la j -ésima variable.
- F_k es el k -ésimo factor común
- ψ_j es la *carga específica* de la variable X_j
- U_j es el *factor específico* para la variable X_j

En este caso haremos las siguientes suposiciones:

- Los factores comunes F_k son variables aleatorias que siguen una distribución marginal $N(0, 1)$. Además se supondrá que $Cov(F_k, F_{k'}) = 0, k \neq k'$, de manera que el vector aleatorio \mathbf{f} de longitud m con distribución $N_m(0, \mathbf{I})$. Y son completamente independientes de los factores específicos.
- Los factores específicos U_j son variables aleatorias con una distribución normal $N(0, 1)$ no correladas, entre ellas que forman el vector aleatorio $\mathbf{u} \sim N_p(0, \mathbf{I})$ de longitud p . Son completamente independientes de los factores comunes.

De esta manera se pueden definir las siguientes matrices:

Definición 2.2.1. Llamaremos *matriz de cargas de los factores o matriz de cargas*, Λ de tamaño $p \times m$ que la matriz es:

$$\Lambda = \begin{pmatrix} \lambda_{11} & \cdots & \lambda_{1m} \\ \vdots & \ddots & \vdots \\ \lambda_{p1} & \cdots & \lambda_{pm} \end{pmatrix} \quad (2.2.2)$$

Definición 2.2.2. Llamaremos matriz específica a la matriz diagonal Ψ de tamaño $p \times p$ a aquella que tiene los términos ψ_j donde $j = 1, \dots, p$:

$$\Psi = \begin{pmatrix} \psi_1 & 0 & \cdots & 0 \\ 0 & \psi_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \psi_p \end{pmatrix} \quad (2.2.3)$$

De esta manera, se puede dar una versión matricial de la expresión 2.2.1:

$$\mathbf{x}^T = \Lambda \mathbf{f}^T + \Psi \mathbf{u}^T \quad (2.2.4)$$

Ya que se consideran los vectores aleatorios $\mathbf{x}, \mathbf{f}, \mathbf{u}$ vectores ordenados como filas.

Proposición 2.2.1. La varianza de la variable aleatoria X_j , σ_j^2 se puede descomponer de la siguiente manera:

$$\sigma_j^2 = \sum_{k=1}^m \lambda_{kj}^2 + \psi_j^2 \quad (2.2.5)$$

Demostración.

$$\begin{aligned} \mathbb{E}(X_j^2) = \sigma_j^2 &= \sum_{k=1}^m \lambda_{kj}^2 \mathbb{E}(F_k^2) + \sum_{k=1}^m \sum_{n=1, n \neq k}^m \lambda_{kj} \lambda_{nj} \mathbb{E}(F_k \cdot F_n) \\ &+ \sum_{k=1}^m \lambda_{kj} \psi_j \mathbb{E}(F_k \cdot U_j) + \psi_j^2 \mathbb{E}(U_j^2) \end{aligned}$$

Utilizando las hipótesis del modelo:

$$\sigma_j^2 = \sum_{k=1}^m \lambda_{kj}^2 + \psi_j^2$$

□

Teniendo en cuenta esta descomposición se forma el siguiente concepto.

Definición 2.2.3. Se llama *comunalidad* de la variable X_j , $h_j^2 = \sum_{k=1}^m \lambda_{kj}^2$ [22]

Esto permite interpretar la varianza de la variable como la varianza explicada por los factores comunes por un lado y la variabilidad específica de la propia variable.

Proposición 2.2.2. La covarianza de dos variables $Cov(X_j, X'_j)$, $\sigma_{jj'}$ cumple que [17, 6]:

$$\sigma_{jj'} = \sigma_{j'j} = \sum_{k=1}^m \lambda_{kj} \lambda_{kj'} \quad (2.2.6)$$

Demostración.

$$\sigma_{jj'} = \sum_{k=1}^m \lambda_{kj} \lambda_{kj'} \mathbb{E}(F_k^2) + \sum_{k=1}^m \sum_{n=1, n \neq k}^m \lambda_{kj} \lambda_{nj'} \mathbb{E}(F_k \cdot F_n) \quad (2.2.7)$$

$$+ \sum_{k=1}^m \lambda_{kj} \psi_{j'} \mathbb{E}(F_k \cdot U_{j'}) + \sum_{k=1}^m \lambda_{kj'} \psi_j \mathbb{E}(F_k \cdot U_j) + \psi_j \psi_{j'} \mathbb{E}(U_j \cdot U_{j'}) \quad (2.2.8)$$

De nuevo, teniendo en cuenta las hipótesis

$$\sigma_{jj'} = \sigma_{j'j} = \sum_{k=1}^m \lambda_{kj} \lambda_{kj'} \quad (2.2.9)$$

□

Y tomando estas dos proposiciones se puede tener que la matriz Σ , cumple lo siguiente:

Corolario 2.2.1. La matriz $\Sigma = \Lambda \Lambda^T + \Psi$

Bajo esta idea en la qu

2.2.2. Estimación de la matriz de cargas

Supóngase el vector aleatorio x como antes del cual se recogen N observaciones para obtener la matriz de datos \mathbf{X} que se supondrá centrada sin pérdida de generalidad.

El objetivo de esta sección es estimar la matriz de cargas Λ a partir de la matriz de covarianzas. Para empezar, supóngase conocido el número de factores m que se desea establecer en el modelo

2.2.3. Unicidad del modelo

2.2.4. Interpretación del modelo factorial

2.3. Análisis de Clusters

Según *Jain, A.K. y Richard, C.D.*; [12] el análisis de clusters tiene como objetivo conseguir una clasificación de las observaciones en subconjuntos que tengan sentido de acuerdo al contexto de la investigación. El clustering es un tipo de clasificación de los datos que tiene las siguientes características:

- *Exclusividad*: Una observación no puede pertenecer a más de un cluster a la vez.
- *Es intrínseco*: No hay ninguna etiqueta que permita clasificar las observaciones, son las características de las propias muestras las que servirán como diferenciadores. Lo que hace que sea un método *no supervisado*.

Una vez clarificado las características esenciales del clustering, este se puede dar de manera *jerárquica* o *particional*.

Definición 2.3.1. Se dice que un método de clustering es *jerárquico* si genera una secuencia de particiones del espacio de observaciones las cuales están anidadas

Definición 2.3.2. Un método de clustering *particional* genera una única partición del espacio.

Por otro lado, los algoritmos se pueden clasificar según sean *aglomerativos*, en los que se empieza teniendo cada una de las observaciones y se van formando los clusters hasta tener un solo cluster con todas las muestras. Por el contrario, los *algoritmos divisivos*, se empieza con un solo cluster y se van haciendo subconjuntos del mismo.

Otra clasificación puede venir dada por el modo en el que se consideran las variables, si es *Monotético*, en cada paso se centra en una variable, mientras que si es *Politético* se utilizan todas las variables a la vez.

También hay que distinguir métodos que usan como fundamento el álgebra matricial, centrándose en los datos que nos puede dar la matriz de distancias entre observaciones o en la Teoría de Grafos.

En primer lugar, es necesario definir los conceptos principales de esta sección que es un clustering y un cluster:

Definición 2.3.3. Se llama *clustering* a la partición que provoca la relación \mathcal{R} y se definen los *clusters* como las clases de equivalencia de dicha relación, $\{c_i\}$.

Por tanto, el análisis de clusters es equivalente a establecer una relación de equivalencia entre las observaciones.

2.3.1. Algoritmos Jerárquicos

Los algoritmos jerárquicos son aquellos que utilizan como entrada no la matriz de los datos per sé, en su lugar, estas utilizan como entrada la matriz de distancias o similitud.

Uno de los inconvenientes mayores de este tipo de algoritmos es como calcular la matriz de similaridades o distancias. En el caso de que las variables sean numéricas y continuas entonces se puede utilizar la distancia euclídea entre elementos de \mathbb{R}^p . Además hay que tener en cuenta si se va a estandarizar ya que en el caso de que si se estandarice se dará el mismo peso a todas las variables, dando igual su escala, en cambio si no se estandariza, la escala provoca que las características con mayores valores tengan mayor peso en las distancias. Por otro lado, si se tienen variables binarias o categóricas hay que tener en cuenta cómo medir sus distancias, todo esto se soluciona con las siguiente definición y consideraciones:

Definición 2.3.4. Llamaremos similaridad entre dos muestras i, h según la variable j a una función s_{jih} la cual cumpla que:

- La similaridad de una muestra consigo misma es igual a la unidad $s_{jii} = 1 \forall j, i$
- $0 \leq s_{jih} \leq 1 \quad \forall j, i, h$
- Simetría: $s_{jih} = s_{jhi}$

Una vez considerado esto, se puede crear un coeficiente de similaridad entre dos mediante el coeficiente de *Gower*:

$$s_{ih} = \frac{\sum_{j=1}^p w_{jih} s_{jih}}{\sum_{j=1}^p w_{jih}} \quad (2.3.1)$$

Donde la variable w_{jih} puede ser 0 ó 1, dependiendo de si se quiere tomar o no dicha variable o no en la comparación de dichas muestras.

Lo más habitual para formalizar las similaridades es tomar lo siguiente:

- Para continuas se puede tomar el valor $s_{jih} = 1 - \frac{|x_{ji} - x_{hi}|}{\text{rango}(X_j)}$
- En cambio para variables binarias en el caso de que coincida el atributo $x_{ji} = x_{jh}$ la similaridad es 1 y 0 en el caso contrario. También se puede decir que si hay presencia del atributo y coinciden es 1 y en caso contrario 0.

Una vez decidida la forma de decir de inicialmente dar la matriz de distancias o similaridades, se da el algoritmo aglomerativo de generación de la jerarquía:

- Se comienza con un *clustering* que tiene n *clusters* con una observación cada uno y se calcula la matriz de distancias
- Se toman los elementos que más cercanos están y se forman una nueva clase.
- Se sustituyen los dos elementos anteriores por la clase y se calcula la distancia entre la clase y el resto de observaciones de acuerdo con uno de los criterios preestablecidos.
- Repetir los dos pasos anteriores hasta obtener una única clase.

Dependiendo de la forma que se de unir los grupos se obtendrá un algoritmo u otro de clustering:

- Si $f(x, y) = \min(x, y)$ es decir, se toma como distancia entre la clase y el resto de observaciones como el mínimo de las que se den, a este algoritmo se le llama *single linkage*, este tipo de enlace tiende a formar.
- Si $f(x, y) = \max(x, y)$ es análogo al anterior, pero tomando la distancia máxima, a este método se le llama *complete linkage*.
- Si $f(x, y) = \frac{x + y}{2}$ se hace la media de las distancias de los dos grupos antes de unirlos, a este método se le llama *average linkage*.

Este tipo de métodos aporta una jerarquía en la agrupación de los datos en función de su similitud, de manera que a distintos grados de similitud se pueden establecer distintas reparticiones del espacio de observaciones. Además esto se puede representar fácilmente con un dendograma o diagrama de árboles otorgando una manera sencilla de establecer un criterio de similitud máxima.

2.3.2. Algoritmos Particionales

El clustering particional busca dado un conjunto de datos heterogéneos dividirlos en grupos homogéneos, como por ejemplo la segmentación de clientes o votantes. , buscaremos un criterio por el cual podamos afirmar que una partición del espacio es mejor que la anterior o no. En estos casos la variabilidad total de todas las observaciones se puede descomponer como la variabilidad intra-cluster y la inter-cluster como se detalla en la sección del análisis discriminante.

Definición 2.3.5. Se llama *centroide* de un cluster, C_k , $k = 1 \dots K$ al elemento que resulta de hacer la media de todos los elementos del cluster C_k :

$$\bar{x}_{jk} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij} \quad (2.3.2)$$

Definición 2.3.6. Se llama variabilidad o variación intra-cluster del cluster C_k $k = 1 \dots K$ a la siguiente expresión:

$$\mathbf{W}(C_k) = \sum_{i \in C_k} \sum_{j=1}^p (x_{ij} - \bar{x}_{jk})^2 \quad (2.3.3)$$

Esta medida también se podría dar como la media de las diferencias cuadradas entre las observaciones del cluster, pero se seleccionará esta expresión.

Una vez dadas estas definiciones y criterios se puede construir un algoritmo iterativo como es el método de K-means que se puede resumir de la siguiente manera, sabiendo el número inicial de clusters, por ejemplo K :

1. Se inicializa la asignación de manera aleatoria, agrupando en los K clusters observaciones aleatoriamente.
2. Se itera lo siguiente hasta que no haya cambio en los clusters:
 - a) Se calculan los centroides de cada uno de los K clusters

- b) Se reasignan las observaciones de acuerdo a qué centroide es más cercano según la distancia euclidiana.

Proposición 2.3.1. El algoritmo de K -means sólo puede mejorar el criterio de la variación dentro de clusters.

Demostración. El paso 2a minimiza la suma de las desviaciones cuadradas y en el paso 2b la reasignación solo puede mejorar eso ya que se busca que la distancia euclídea al *centroide* sea la mínima. \square

Esto provoca que el algoritmo no para hasta que no se da una mejora en el criterio. Sin embargo, este mínimo que alcanza la función no tiene por qué ser global, ya que esta no es convexa y por tanto la consecución de este mínimo puede estar condicionada por la asignación inicial. Para solucionar estos problemas, se deben hacer varias ejecuciones del algoritmo para poder compararlas entre sí, por ejemplo *Scikit-Learn* implementa el método *k-means++* que no es más que realizar multiples ejecuciones con asignaciones iniciales distintas y luego las compara y otorga la mejor. Para poder compararlas se puede utilizar el parámetro inercia.

Definición 2.3.7. Se llama *inercia* de un clustering a la suma de las variaciones intra-clusters:

$$inercia = \sum_{k=1}^K \mathbf{W}(C_k) \quad (2.3.4)$$

La inercia representa como de compacta es nuestra partición a menor inercia más compacta es y por tanto más homogéneos son.

Otro de los problemas de este tipo de algoritmos es que se deben conocer a priori el número de clusters en los que se quieren dividir las observaciones. Para ello *Peña D.*[22] propone un contraste de hipótesis utilizando la inercia haciéndola depender del número de clusters

$$F = \frac{inercia(K) - inercia(K + 1)}{\frac{inercia(K+1)}{n-K-1}} \quad (2.3.5)$$

Este cociente compara la disminución de la variación entre una partición con K y $K + 1$ clústers, de esta manera, puede compararse con una F con $p, p(n - K - 1)$ grados de libertad, pero que en la práctica no se puede asumir las hipótesis, por tanto, en la práctica si se da un valor mayor que 10 se asume que es conveniente añadir un clúster más.

Capítulo 3

Aplicación sobre datos

A lo largo de este capítulo se aplicarán algunas de las técnicas desarrolladas anteriormente a ejemplos prácticos con bases de datos reales obtenidas de distintos repositorios.

La principal herramienta usada ha sido el lenguaje de programación Python, utilizando librerías como Pandas para el manejo de las bases de datos, Scikit-Learn para la implementación de los modelos y otras librerías de visualización como Matplotlib para la inclusión de gráficos sencillos.

3.1. Clasificación de Iris

Uno de los ejemplos básicos del análisis discriminante es el conjunto de datos de distintas clases de flores Iris, en particular *Iris Setosa*, *Iris virginica*, e *Iris versicolor*. Nuestro objetivo con este conjunto de datos es encontrar variables latentes o factores que pueden darse debido a la alta correlación que se observan en un estudio inicial y luego realizar un análisis discriminante para obtener un modelo predictivo, usando las variables transformadas por los factores y las originales para comprobar como esta transformación puede afectar a las predicciones.

El *dataset* que se va a utilizar recoge datos de 150 muestras y toma las siguientes 5 medidas, ancho y largo del pétalo y ancho y largo del sépalo, además de su clasificación dentro de 1 las 3 especies antes detalladas.

El primer paso es intentar encontrar los factores latentes que puedan explicar la mayor parte de la variabilidad de forma común. En caso de encontrar esa variable común, que pudiera interpretarse en este caso como un *factor tamaño de la flor*, se podría simplificar el conjunto de datos.

Tras aplicar el análisis factorial con la funcionalidad implementada en la biblioteca *Scikit-Learn*, se ha podido obtener un modelo de un sólo factor común que acumula casi el 92% de la variabilidad de las variables, siendo el resto repartido como variabilidad específica de cada variable.

Por otro lado, tras ajustar las funciones discriminantes con los datos sin transformar, se consigue un `accuracy_score` del 100%, sin embargo, utilizando los datos proyectados sobre el factor común, se puede observar que el modelo entrenado tiene una precisión del 95,5% lo cual no es una pérdida significativa de precisión. Por

tanto, en este caso, se podría utilizar tanto el conjunto de datos original como el transformado para realizar las predicciones.

3.2. Alubias secas

3.3. Comparación regresión lineal, árboles de regresión y redes neuronales.

El objetivo de esta aplicación es comparar el rendimiento de los distintos algoritmos para la obtención de modelos predictivos que se han visto durante la memoria para ello, se utilizarán varias aproximaciones. Entre las aproximaciones usadas están el modelo lineal, que es la más sencilla de todas, los árboles de decisión para regresión que son los que un mayor *sobreajuste* pueden tener y las redes neuronales que son las que mayor adaptabilidad tienen a la hora de crear un modelo predictivo debido a la gran cantidad de parámetros que se manejan.

Los resultados son los siguientes,

Bibliografía

- [1] **Abdi, H., and Williams, L. J. (2010).** Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics* 2(4), 433–459.
- [2] **Abdi, H. (2007).** The method of least squares. *Encyclopedia of measurement and statistics*, 1, 530-532.
- [3] **Biau, G., and Scornet, E. (2016).** *A random forest guided tour*. *Test*, 25, 197-227.
- [4] **Breiman, L. (2004).** Consistency for a simple model of random forests. *University of California at Berkeley. Technical Report*, 670.
- [5] **Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., and Brown, S. D. (2004).** An introduction to decision tree modeling. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(6), 275-285.
- [6] **Chatfield, C and Collins A.J (1989).** *Introduction to multivariate analysis*, Chapman and Hall.
- [7] **Díaz-Uriarte, R., and Alvarez de Andrés, S. (2006).** Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7, 1-13.
- [8] **Cuadras, C.M. (2014),** *Nuevos métodos de Análisis Multivariante*, CMC Editions, Barcelona.
- [9] **Hastie, T., Tibshirani, R. and Friedman J. (2001),** *The Elements of Statistical Learning, Data Mining, Inference and Prediction* Springer
- [10] **Jolliffe I.T.(1986).** *Principal Component Analysis*, Springer-Verlag.
- [11] **Köppen, M. (2000)** The curse of dimensionality. *5th online world conference on soft computing in industrial applications (WSC5)* (Vol. 1, pp. 4-8).
- [12] **Jain, A.K., Richard, C.D., (1988)** *Algorithms for clustering data*. Prentice-Hall, Inc.
- [13] *Scikit Learn Documentation, Clustering Methods*. <https://scikit-learn.org/stable/modules/clustering.html#clustering>
- [14] Neural Designer Blog <https://www.neuraldesigner.com/>

- [15] **Lopez, R., Balsa-Canto, E. and Oñate, E. (2008)** Neural networks for variational problems in engineering *Int. J. Numer. Meth. Engng.*, 75 <https://doi.org/10.1002/nme.2304> 1341-1360.
- [16] **Lebart, L., Morineau, A., Warwick, K. M. (1984).** *Multivariate Descriptive Analysis: Correspondence analysis and related techniques for large matrices.* Wiley.
- [17] **Morrison. D.(1976).** *Multivariate statistical methods (2nd ed).* McGraw-Hill Kogakusha.
- [18] **Batista Foguet, J.M. y Martínez Arias, R. (1989)***Análisis Multivariante: Análisis en Componentes Principales.* Editorial Hispano Europea.
- [19] **Divakaran, S. (2022).** Data Science: Principles and Concepts in Modeling Decision Trees. *Data Science in Agriculture and Natural Resource Management*, 55-74.
- [20] **James G, Witten D, Hastie T, Tibshirani R.** *An Introduction to Statistical Learning: With Applications in R.* New York, NY: Springer; 2021.
- [21] **Martínez Arias R.** *El análisis multivariante en la investigación científica.* Madrid: La Muralla; 1999.
- [22] **Peña D.** *Análisis de datos multivariantes.* Madrid: McGraw-Hill; 2002.
- [23] **Andrade-Sánchez, A. I., Galindo-Villardón, M. P., y Cuevas Romo, J. (2015).** Análisis multivariante del perfil psicológico de los deportistas universitarios: Aplicación del CPRD en México. *Educación Física y Ciencia*, 17(2), 00-00.
- [24] **Hernandis, S. P., Diez, J. P., y Rouma, A. C. (2002).** El consumo de inhalables y cannabis en la preadolescencia: Análisis multivariado de factores predisponentes. *Anales de Psicología/Annals of Psychology*, 18(1), 77-93.
- [25] **Machado-Duque, M. E., Echeverri Chabur, J. E., y Machado-Alba, J. E. (2015).** Somnolencia diurna excesiva, mala calidad del sueño y bajo rendimiento académico en estudiantes de Medicina. *Revista colombiana de Psiquiatría*, 44(3), 137-142.
- [26] **Hornik, K., Stinchcombe, M., and White, H.** Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5), 359-366, 1989. [https://doi.org/10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8)
- [27] **Dry Bean Dataset. (2020).** *UCI Machine Learning Repository.* <https://doi.org/10.24432/C50S4B>.
- [28] **Pagès, J. (2005).** Collection and analysis of perceived product inter-distances using multiple factor analysis: Application to the study of 10 white wines from the Loire Valley. *Food quality and preference*, 16(7), 642-649. <https://doi.org/10.1016/j.foodqual.2005.01.006>

- [29] **Okazaki, S. (2006).** What do we know about mobile Internet adopters? A cluster analysis. *Information & Management*, 43(2), 127-141.
- [30] **Mamidi, R. R. (2021).** Application of Neural Networks for Prediction of Double Fed Induction Generator's Equivalent Circuit Parameters used in Wind Generators. *IRJCS:: International Research Journal of Computer Science*, Volume VIII, 23-31.
- [31] **Nerini, D., & Ghattas, B. (2007).** Classifying densities using functional regression trees: Applications in oceanology. *Computational Statistics & Data Analysis* 51(10), 4984-4993.
- [32] **Ensum, J., Pollard, R., & Taylor, S. (2005).** Applications of logistic regression to shots at goal at association football. In *Science and football V: the proceedings of the Fifth World Congress on Science and Football* (pp. 214-221).
- [33] **Greene, W.H. (2008).** *Econometric analysis (6th ed.)* Pearson Prentice Hall
- [34] **Johnson, R. A., & Wichern, D. W. (2007).** *Applied multivariate statistical analysis.(6th edition)* Pearson Prentice Hall.
- [35] **Everitt, B., and Hothorn, T. (2011).** *An introduction to applied multivariate analysis with R.* Springer Science & Business Media.
- [36] **Ringnér, M. (2008).**What is principal component analysis?.*Nature biotechnology*, 26(3), 303-304.
- [37] **Gottumukkal, R., & Asari, V. K. (2004).** An improved face recognition technique based on modular PCA approach. *Pattern Recognition Letters*, 25(4), 429-436.
- [38] **Eckart, C., & Young, G. (1936).** The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3), 211-218.
- [39] **Johnson, R. M. (1963).** On a theorem stated by Eckart and Young. *Psychometrika*, 28(3), 259-263.
- [40] **Golub, G. H., Hoffman, A., & Stewart, G. W. (1987).** A generalization of the Eckart-Young-Mirsky matrix approximation theorem. *Linear Algebra and its applications*, 88, 317-327.
- [41] **Mahesh, B. (2020).** Machine learning algorithms-a review. *International Journal of Science and Research (IJSR)*, 9, 381-386.
- [42] **Pearson, K. (1901).** LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11), 559-572.
- [43] **Hotelling H. (1933)** Analysis of a complex of statistical variables into principal components. *J Educ Psychol* 25:417–441.
- [44] **Vincent, D. F. (1953).** The origin and development of factor analysis. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 2(2), 107-117.