



Universidad Miguel Hernández – Facultad de Ciencias Sociales y Jurídicas

Grado en Estadística Empresarial

Curso académico 2021/2022



Autor: Cristina Oliver Elices
Directora: Mercedes Landete Ruiz

ÍNDICE

RESUMEN	2
INTRODUCCIÓN	3
OBTENCIÓN DE DATOS	6
ESCALAMIENTO MULTIDIMENSIONAL	8
EJEMPLO DE APLICACIÓN DEL ESCALAMIENTO MULTIDIMENSIONAL CON SPSS	10
DESPLEGAMIENTO MULTIDIMENSIONAL.....	14
PROBLEMA DE COMPARACIÓN POR PARES	16
EJEMPLO.....	17
PROBLEMA DE ORDENAMIENTO LINEAL	21
PROBLEMA DE ORDENAMIENTO LINEAL CON RSTUDIO.....	23
CONCLUSIÓN	34
BIBLIOGRAFÍA	36



RESUMEN

Actualmente los rankings se han convertido en nuestro día a día, en muchos ámbitos como la música, el deporte, en las empresas, en las universidades, etc. son utilizados. Por ejemplo, el ranking ATP (Asociación de Tenistas Profesionales) clasifica a los mejores tenistas del momento o los rankings de universidades del mundo QS que evalúan la calidad o prestigio de cada universidad.

El objetivo de este trabajo es crear un ranking de cinco jugadores que pertenecen a la selección española de fútbol a través de tres problemas diferentes de estadística multivariante que analizaremos con profundidad. Los resultados obtenidos nos han permitido identificar las preferencias futbolísticas en tres zonas diferentes de nuestro país.



INTRODUCCIÓN

Los datos se han convertido en algo muy importante para nuestra sociedad, incluso algunos lo llegan a definir como “el petróleo del siglo XXI”. En el siglo pasado todos aquellos que pudieron extraer petróleo del suelo acumularon una gran riqueza, ya que construyeron la economía futura a partir del petróleo, pues empresas como Facebook o Google han hecho algo similar con los datos.

Gracias a la gran cantidad de datos que generamos ahora con las redes sociales, las búsquedas de internet, los GPS, los e-mails o muchos aspectos más podemos hacer miles de cosas como anticiparnos a la realidad simulando procesos, detectar patrones, tendencias o reglas que nos expliquen el comportamiento de la realidad con la minería de datos, analizar mercados, optimizar procesos o recursos, análisis multivariante etc.

En el siguiente trabajo nos vamos a centrar con el análisis multivariante, que se trata de diferentes métodos que estudian y examinan el efecto simultáneo de múltiples variables. Este método se utiliza para analizar el comportamiento conjunto de más de una variable aleatoria.

Existe una amplia gama de técnicas multivariadas y vamos a estudiar en profundidad tres de ellas, en las que su principal objetivo es ordenar las variables. Ordenar variables o crear un ranking de ellas. Un ranking es una clasificación que ordena distintos elementos en función de un criterio de evaluación.

Los problemas que vamos a estudiar a continuación son el escalamiento y desplegamiento multidimensional, el problema de comparación por pares y el problema de ordenamiento lineal.

El escalamiento y desplegamiento multidimensional son dos técnicas estadísticas. La estadística siempre nos ha marcado prácticamente todos los aspectos de nuestra vida cotidiana, a partir de todas nuestras actividades recopilamos datos que luego analizamos y nos permiten tomar decisiones. Pero más allá de nuestra vida cotidiana, gracias a la digitalización y globalización puede ser una ventaja estratégica y competitiva para muchos ámbitos como puede ser la medicina, la economía, la agricultura, la ciencia o la política.

El escalamiento multidimensional, más conocido como MultiDimensional Scaling (MDS) tiene como objetivo representar en un espacio geométrico de pocas dimensiones las proximidades existentes entre un conjunto de objetos. Esta técnica nace a principios del siglo XX en el campo de la psicología. Surge cuando se pretendía estudiar la relación que existía entre la intensidad física de ciertos estímulos con su intensidad subjetiva.

El MDS, parte de una matriz de distancias entre individuos, la cual produce una representación de los individuos en una escala euclídeana de modo que las distancias en dicha escala se aproximen lo mejor posible a las distancias de partida.

Para añadir información al resultado del MDS, utilizamos el despliegamiento multidimensional. El despliegamiento multidimensional tiene como objetivo buscar una escala cuantitativa común que permita examinar visualmente las relaciones entre dos conjuntos de objetos. Utilizamos unas preguntas de preferencia que nos proporcionan una serie de encuestados para buscar la relación con el resultado del escalamiento multidimensional.

Estos dos problemas los resolvemos a través de SPSS, este programa tiene ya más de 30 años de existencia y en la actualidad ha alcanzado difusión a nivel mundial. Gracias a su intuitiva interfaz gráfica, este programa se hace fácil de utilizar frente a otras herramientas de análisis de datos.

Seguimos con el problema de comparación por pares que se trata de un problema de inteligencia artificial. La inteligencia artificial es la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un humano.

El sistema de comparación por pares es una poderosa herramienta para argumentar la importancia relativa de varias opciones, cuando tal importancia no puede evaluarse mediante una calificación directa. Es decir, cuando no se dispone de criterios objetivos que hagan posible una calificación numérica de valor de cada opción.

En la práctica, la mayor parte de las decisiones se fundamentan en relaciones de preferencia donde cada pareja de alternativas supone un grado de predilección de una sobre otra. Cabe destacar que este problema está diseñado para pocas variables.

Por último, analizamos el problema de ordenamiento lineal que se trata de un problema de optimización, más concretamente optimización combinatoria.

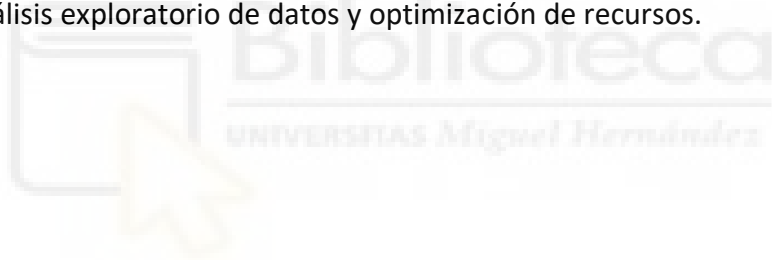
La optimización es el proceso buscar la mejor solución posible a un problema. Estos problemas se pueden expresar como encontrar el valor de unas variables de decisión para los que una determinada función objetivo alcanza su valor máximo y mínimo. El valor de las variables generalmente está sujeto a unas restricciones.

Existen distintas clases de problemas dentro de la optimización, encontramos por ejemplo los problemas lineales que son problemas muy fáciles de resolver ya que su función objetivo y sus restricciones son expresiones lineales, estos problemas suelen ser resueltos con el conocido método Simplex. Luego existen problemas con más complejidad algorítmica, es decir, problemas más difíciles de resolver, que su término científico es NP-hard. Un ejemplo de optimización NP-hard es la optimización combinatoria que consiste en encontrar en un conjunto un subconjunto que contenga las “mejores soluciones”.

El problema de ordenamiento lineal parte de una matriz cuadrada $C_{n \times n}$, en la cual se permutan sus filas y columnas simultáneamente hasta que la suma de los valores que están por encima de la diagonal llegan a su máximo, una vez llegados a este punto podemos observar el orden de las variables.

Si el problema tiene n variables, tiene $n!$ iteraciones, ya que tendríamos que permutar todas las filas y columnas con todas las combinaciones posibles entre variables. Por tanto, es un problema muy costoso de hacer a mano, si disponemos de 3 variables tendríamos 6 iteraciones, pero si tenemos 5 variables serían 120 iteraciones, es decir, solo con aumentar 2 variables se suman 114 iteraciones más. Por esta misma razón, lo resolvemos también a través de R-Studio. RStudio es un software diseñado para hacer análisis estadísticos y gráficas.

Para finalizar cabe destacar las disciplinas impartidas en el grado de estadística empresarial que tenemos en cuenta para este trabajo. En minería de datos estudiamos la representación gráfica de datos multivariantes en SPSS. Técnicas de asociación, agrupación y reducción de la dimensión como el análisis clúster, análisis de componentes principales, análisis factorial... y entre ellas el escalamiento multidimensional. En técnicas estadísticas de análisis de mercados también estudiamos diversas técnicas multivariantes, aunque son analizadas en RStudio. En modelos de optimización se estudian diversos modelos como el de asignación, secuenciación, emparejamiento, etc. en los cuales nos ayudan a entender mejor cómo funciona el problema de ordenamiento lineal. En introducción a la programación se estudia como programar bucles, funciones recursivas y de condicionamiento, etc. esto nos ha ayudado a programar y entender a la perfección el problema de ordenamiento lineal en RStudio. Otras materias que también nos han servido, aunque más indirectamente serían: cálculo numérico, análisis exploratorio de datos y optimización de recursos.



OBTENCIÓN DE DATOS

Antes de empezar cualquier estudio estadístico necesitamos obtener los datos, en nuestro caso hemos hecho una encuesta a través de Google Forms y le hemos preguntado a 34 personas por sus preferencias entre 5 jugadores de fútbol. Cabe destacar que la población encuestada el 65% son de la provincia de Castellón, el 25% de la provincia de Alicante y el 10% de otros lugares.

Para realizar la encuesta en primer lugar hemos cogido a 5 jugadores de la selección española (Álvaro Morata, Jordi Alba, Marcos Llorente, Pau Torres y Busquets), y hemos hecho todas las combinaciones por pares posibles, teniendo un total de 10 combinaciones:

1	Busquets – Morata
2	Jordi Alba – Busquets
3	Jordi Alba – Morata
4	Llorente – Busquets
5	Pau Torres – Busquets
6	Pau Torres – Jordi Alba
7	Llorente – Jordi Alba
8	Pau – Morata
9	Llorente – Morata
10	Pau – Llorente

Una vez hechas estas combinaciones hemos encuestado a la población preguntándole por la preferencia de cada par de jugadores, siendo el 1 el jugador preferido:

Encuestado	BUSQUETS	MORATA	JORDI ALBA	BUSQUETS	JORDI ALBA	MORATA	LLORENTE	BUSQUETS	PAU TORRES	BUSQUETS
1	1			1	1			1	1	
2	1			1	1			1		1
3	1		1		1			1	1	
4	1			1	1			1		1
5	1		1		1		1			1
6	1			1		1		1	1	
7	1			1	1			1	1	
8		1	1			1	1		1	
9	1			1	1			1		1
10	1			1	1			1		1
11	1			1		1	1		1	
12	1			1		1		1	1	
13	1			1	1		1		1	
14		1	1			1	1		1	
15		1	0	1		1	1		1	0
16	1		0	1	1	0	0	1		1
17		1	1			1	1		1	
18		1		1		1	1			1
19	1		1		1			1		1
20	1			1		1		1		1
21		1		1		1	1		1	
22		1	1		1		1		1	
23		1		1		1		1		1
24		1		1		1		1		1
25	1			1	1		0	1	0	1
26	1		1		1			1		1
27	1			1	1			1		1
28		1		1		1	1		1	
29	1			1		1	1		1	
30	1		1		1			1	1	
31		1		1		1	1		1	
32	1		1		1			1		1
33	1		1		1			1	1	
34	1		1		1			1		1
SUMA VOTOS	23	11	12	22	19	15	13	21	18	16

Encuestado	PAU TORRES	JORDI ALBA	LLORENTE	JORDI ALBA	PAU TORRES	MORATA	LLORENTE	MORATA	PAU TORRES	LLORENTE
1	1			1	1			1	1	
2	1			1	1		1		1	
3	1			1	1		1		1	
4		1		1		1		1	1	
5		1	1			1	1			1
6	1			1	1			1	1	
7	1			1	1			1	1	
8	1		1		1			1	1	
9	1			1	1		1		1	
10	1		1		1		1		1	
11	1		1		1		1		1	
12	1		1		1			1	1	
13	1		1		1		1		1	
14	1			1	1			1	1	
15	1		1			1		1		1
16	0	1		1		1		1	0	1
17	1		1			1		1	1	1
18		1	1			1		1		1
19		1		1		1		1	1	
20	1			1	1			1	1	
21	1		1			1		1	1	
22	1		1		1		1		1	
23	1			1		1		1	1	
24		1		1		1		1		1
25	0	1	0	1	1		1	0	1	
26		1		1	1		1		1	
27		1		1	1		1		1	
28	1		1		1			1	1	
29	1		1		1		1		1	
30	1			1	1			1	1	
31	1		1		1		1		1	
32		1		1		1		1		1
33	1			1	1		1		1	
34		1		1	1			1	1	
SUMA VOTOS	23	11	14	20	23	11	14	20	27	7

Y este ha sido el resultado que nos hemos obtenido en la encuesta. A partir de esta tabla sacaremos todos los datos que vamos a necesitar durante el trabajo.

Simplificaremos el nombre de las variables para que nos resulte más fácil trabajar con ellas, reduciremos el nombre dejando solamente su inicial. Por ejemplo: Morata → M o Jordi Alba → J.

ESCALAMIENTO MULTIDIMENSIONAL

El escalamiento multidimensional es una técnica de representación espacial (normalmente en dos o tres dimensiones) de las percepciones y preferencias de los encuestados hacia una serie de objetos. El mapa espacial representa la información como posición o distancia entre los diferentes objetos estudiados.

Para nuestro estudio vamos a utilizar los datos de la encuesta que hemos realizado, aunque los vamos a poner de forma en que cada individuo puntúe del 0 al 4 por orden de preferencia de cada jugador. Es decir, sumamos el número de veces que ha votado cada encuestado a cada jugador.

Encuestado	PAU TORRES	BUSQUETS	JORDI ALBA	MORATA	LLORENTE
1	4	3	2	1	0
2	3	4	2	0	1
3	4	2	3	0	1
4	1	4	3	2	0
5	0	2	3	1	4
6	4	3	1	2	0
7	4	3	2	1	0
8	4	0	1	3	2
9	3	4	2	0	1
10	3	4	1	0	2
11	4	2	0	1	3
12	4	3	0	2	1
13	4	2	1	0	3
14	4	0	2	3	1
15	2	1	0	4	3
16	0	4	3	2	1
17	2	0	1	4	3
18	0	2	1	4	3
19	1	3	4	2	0
20	3	4	1	2	0
21	3	1	0	4	2
22	4	0	2	1	3
23	2	3	1	4	0
24	0	3	2	4	1
25	2	4	3	0	1
26	2	3	4	0	1
27	2	4	3	0	1
28	4	1	0	3	2
29	4	2	0	1	3
30	4	2	3	1	0
31	4	1	0	2	3
32	0	3	4	2	1
33	4	2	3	0	1
34	2	3	4	1	0
TOTAL	91	82	62	57	48

*Partimos de 5 variables que son los jugadores, por tanto $n=5$.

Empezamos con el escalamiento multidimensional, lo primero que hacemos es identificar cómo son los datos, en nuestro caso disponemos de una matriz de preferencias y tenemos como objetivo transformarla en una matriz de distancias que denominamos $D \in M_{n \times n}$:

$$D = \begin{pmatrix} d_{11} & d_{12} & \dots & d_{1n} \\ d_{21} & d_{22} & \dots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \dots & d_{nn} \end{pmatrix}$$

Para crear la matriz de distancias calculamos la distancia existente entre cada par de variables, en nuestro caso las calcularemos a partir del método de la distancia euclídea. La matriz debe cumplir los siguientes requisitos:

- No negatividad
- Simetría
- Desigualdad triangular

Una vez obtenida la matriz de distancias es necesario transformarla en una matriz que se llama de productos escalares $B \in M_{n \times n}$ y que es el resultado de:

$$b_{ij} = -0.5(d_{ij}^2 - d_{i.}^2 - d_{.j}^2 + d_{..}^2)$$

donde:

$$d_{i.}^2 = \frac{1}{n} \sum_{j=1}^n d_{ij}^2 \quad d_{.j}^2 = \frac{1}{n} \sum_{i=1}^n d_{ij}^2 \quad d_{..}^2 = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n d_{ij}^2$$

Luego, transformamos la matriz $B \in M_{n \times n}$ en una matriz $X \in M_{n \times m}$ tal que $B = X \cdot X'$, siendo X la matriz que nos da las coordenadas de cada uno de los n estímulos en cada una de las m dimensiones.

Para finalizar, nos aseguramos que el modelo se ajusta bien a los datos. En el caso del escalamiento multidimensional busca que haya la máxima correspondencia entre las distancias generadas y las observadas, para ello usamos el Stress, este indicador adquiere un valor que oscila entre el 0 y el 1. Cuanto menor es el stress, mejor es el ajuste.

Kruskal sugiere las siguientes interpretaciones del Stress:

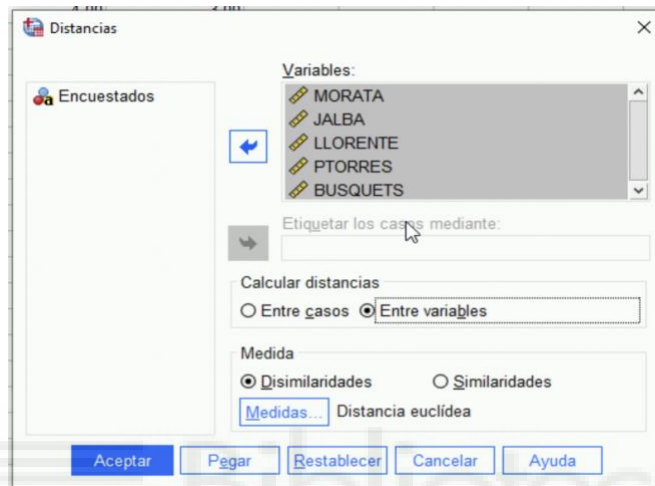
0,2	Malo
0,1	Regular
0,05	Bueno
0,025	Excelente
0,0	Perfecto

EJEMPLO DE APLICACIÓN DEL ESCALAMIENTO MULTIDIMENSIONAL CON SPSS

Utilizaremos el Spss para realizar el estudio, ya que nos va a resultar más fácil visualmente y nos proporcionará unos cálculos más exactos. Explicaremos paso a paso lo que hacemos.

Llevamos a la practica lo explicado anteriormente siguiendo los siguientes pasos:

ANALIZAR → CORRELACIONAR → DISTANCIAS



Resultado:

Resumen de procesamiento de casos

Válido		Casos Perdidos		Total	
N	Porcentaje	N	Porcentaje	N	Porcentaje
34	100,0%	0	0,0%	34	100,0%

Matriz de proximidades

Distancia euclídea

	MORATA	JALBA	LLORENTE	PTORRES	BUSQUETS
MORATA	,000	13,454	10,149	14,629	14,036
JALBA	13,454	,000	12,806	14,595	9,055
LLORENTE	10,149	12,806	,000	13,229	14,071
PTORRES	14,629	14,595	13,229	,000	13,153
BUSQUETS	14,036	9,055	14,071	13,153	,000

Esto es una matriz de disimilaridad.

Se nos queda la siguiente matriz de distancias:

	M	J	L	P	B
M	0	13,454	10,149	14,629	14,036
J	13,454	0	12,806	14,595	9,055
L	10,149	12,806	0	13,229	14,071
P	14,629	14,595	13,229	0	13,153
B	14,036	9,055	14,071	13,153	0

Esta matriz nos explica el grado de proximidad que tienen las variables, es decir, cuanto menor es el valor más se parecen. Por esta misma razón la diagonal es 0.

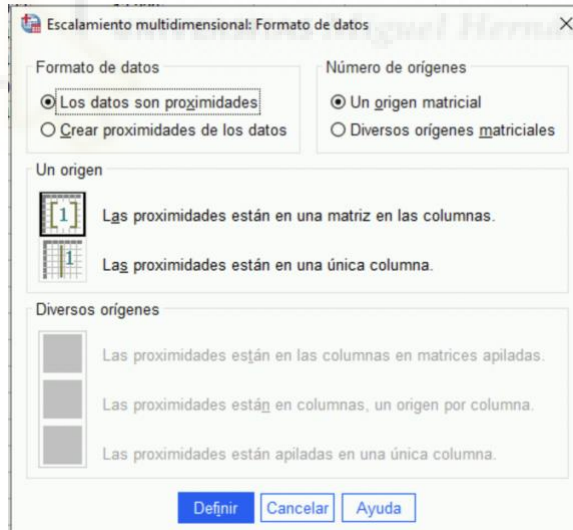
También podemos ver como por ejemplo Jordi Alba y Busquets tienen el grado más pequeño, esto quiere decir que según la población encuestada son los más parecidos. O Pau Torres y Morata son los que menos se parecen.

Una vez tenemos la matriz de proximidades, la copiamos en una nueva base de datos y realizamos el análisis de escalamiento multidimensional. Utilizaremos el método PROXSCAL, ya que partimos de una matriz de proximidades.

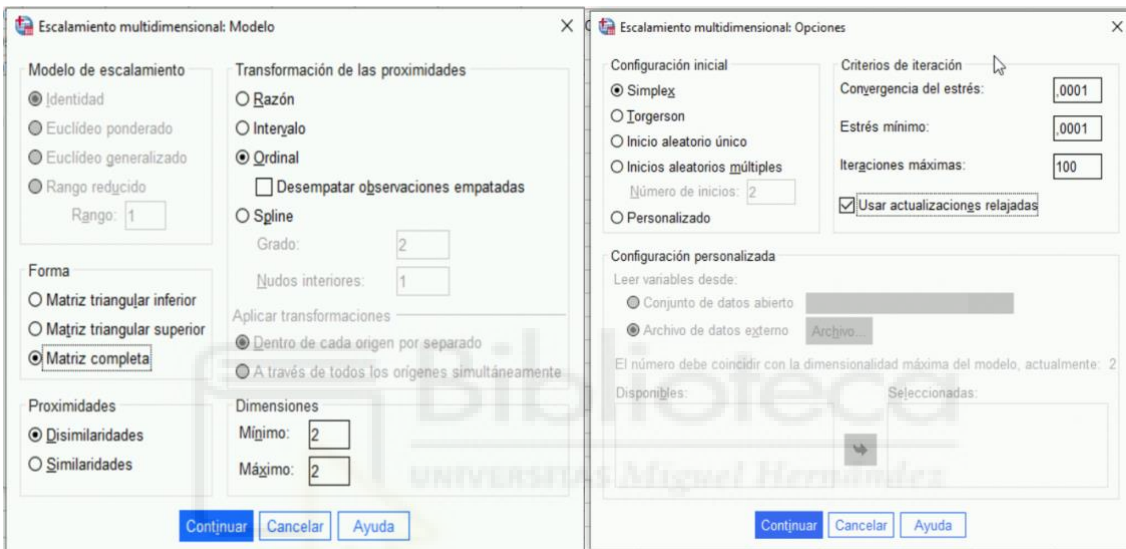
ANALIZAR → ESCALA → ESCALAMIENTO MULTIDIMENSIONAL (PROXSCAL)

1. Pulsamos “Definir” .
2. Pasamos todas las variables a proximidades.
3. En “Modelo..” modificamos que la forma es matriz completa, que las proximidades son disimilaridades y que la transformación de las proximidades es ordinal.
4. En “Opciones..” marcamos “usar actualizaciones relajadas”.

1º



2º



3º

4º

Resultado:

→ Medidas de Stress y de ajuste:

Estrés bruto normalizado	,00021
Estrés-I	,01444 ^a
Estrés-II	,04339 ^a
S-Estrés	,00073 ^b
Dispersión contada para (D.A.F.)	,99979
Coefficiente de congruencia de Tucker	,99990

PROXSCAL minimiza el estrés bruto normalizado.

a. Factor de escalamiento óptimo = 1,000.

b. Factor de escalamiento óptimo = 1,000.

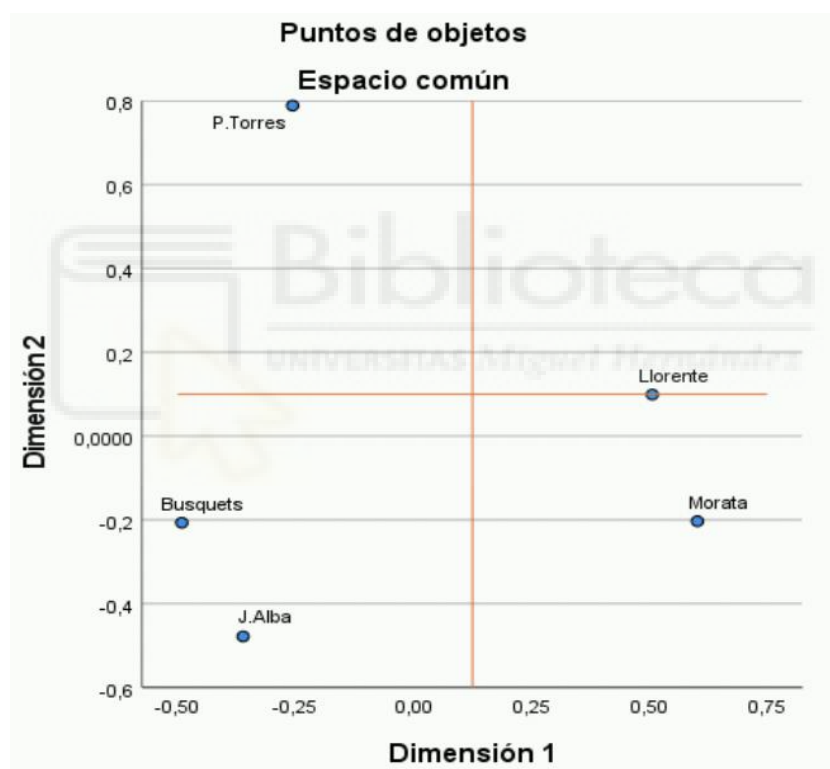
El Stress es muy cercano al 0 por tanto decimos el la bondad de ajuste es perfecta.

El coeficiente de dispersión y el coeficiente de Tucker deben estar cerca del 1. Explica que estamos realizando un modelo con un buen ajuste.

→ Coordenadas finales:

	Dimensión	
	1	2
Morata	,602	-,203
J.Alba	-,361	-,478
Llorente	,506	,099
P.Torres	-,256	,789
Busquets	-,491	-,207

→ Gráfico:



En el gráfico observamos las variables no siguen ningún orden, la información que nos aporta es la similitud entre los jugadores que claramente se dividen en tres grupos. En primer lugar tenemos a Pau Torres que se sitúa en el segundo cuadrante. Luego tenemos en el tercer cuadrante a Busquets y Jordi Alba y en el cuarto cuadrante tenemos a Morata y a Llorente.

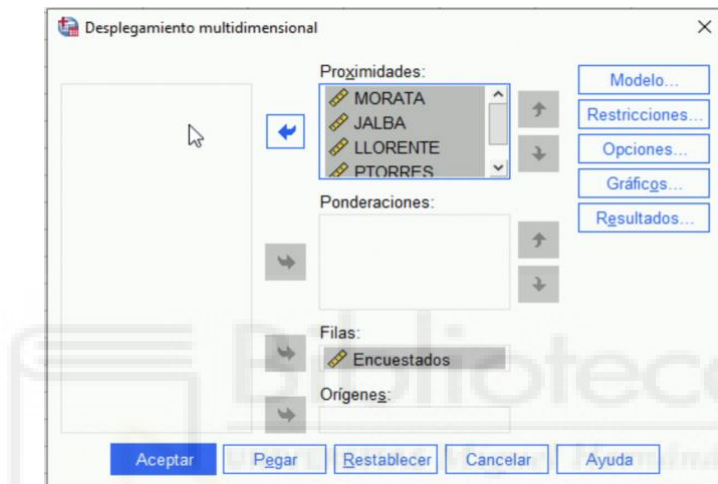
Algunos de estos parecidos ya se podían predecir anteriormente con la matriz de distancias, ya que el valor más pequeño es el de Busquets y Jordi Alba con 9,055, por tanto, son los más parecidos y luego le sigue Morata y Llorente con 10,149.

DESPLÉGAMIENTO MULTIDIMENSIONAL

El procedimiento del desplegamiento multidimensional intenta buscar una escala cuantitativa común que permita examinar visualmente las relaciones entre dos conjuntos de objetos. En nuestro caso examinaremos las relaciones entre los encuestados y los jugadores. Seguimos los siguientes pasos:

ANALIZAR → ESCALA → DESPLÉGAMIENTO MULTIDIMENSIONAL (PREFSCAL)

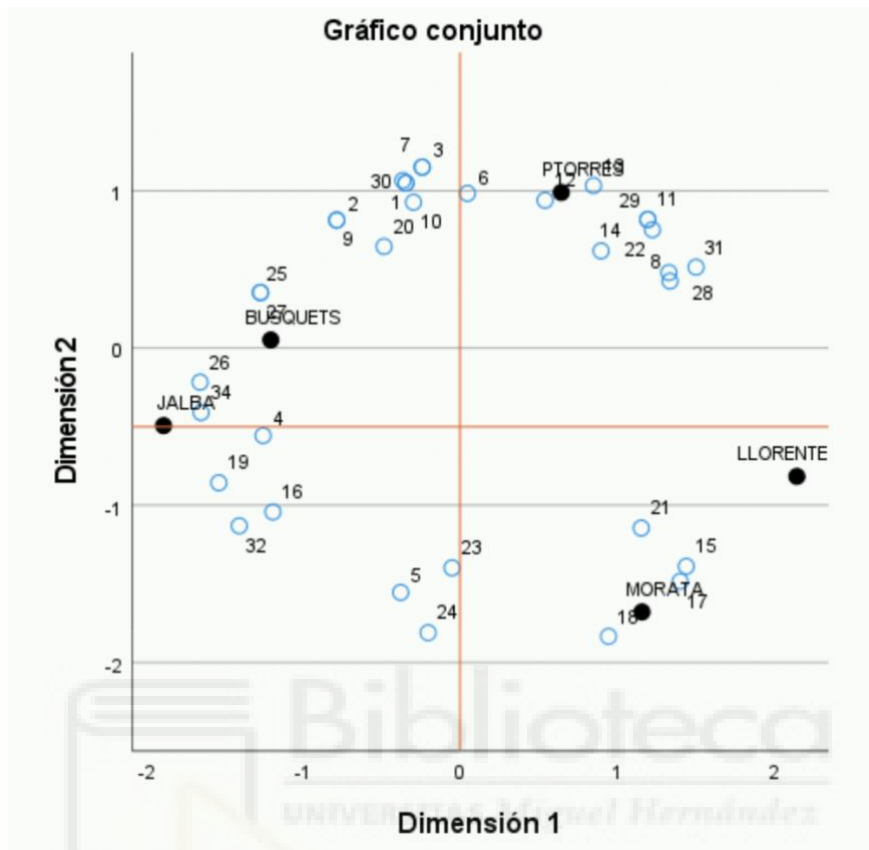
Para continuar cargamos la variable “Encuestados” a “Filas:” y el resto de variables a proximidades:



En “Modelo..” ponemos “ninguna” en transformación de las proximidades e incluimos la intersección.

En “opciones...” tomaremos en cuenta la configuración clásica de spearman, que es un método clásico no paramétrico que nos permite examinar la intensidad de asociación entre dos variables, en este caso sería por un lado los encuestados y por el otro los 5 jugadores.

Aceptamos todo y vemos los resultados, que lo que realmente nos interesa es el gráfico conjunto:



En este gráfico tenemos los puntos azules que se trata de los encuestados y los puntos negros de cada jugador. Podemos observar como la gran mayoría de los encuestados están más cercanos a Pau Torres, luego se van extendiendo entre Busquets y Jordi alba y finalmente los menos votados serian Morata y Llorente.

Si comparamos el gráfico del escalamiento multidimensional con este gráfico observamos que ambos siguen el mismo patrón, aunque este gráfico nos aporta más información, ya que incluye a la población encuestada. La cercanía de la población con las variables (jugadores) nos informan de la preferencia de cada encuestado.

Para concluir el escalamiento multidimensional, diríamos que se ha quedado el siguiente ranking de preferencia:

1	Pau Torres
2	Busquets
3	Jordi Alba
4	Morata
5	Llorente

PROBLEMA DE COMPARACIÓN POR PARES

El problema de comparación por pares basado en regresión binaria bayesiana consiste en describir un orden de preferencia para el grupo, que denominaremos \geq_g y en el que se combinan las preferencias individuales (\geq_i). Suponiendo que cada individuo tiene un orden de preferencia distinto y que el grupo es capaz de ordenar algunos pares de alternativas.

Disponemos de un conjunto finito de \mathcal{A} alternativas y un grupo de decisores donde cada individuo tiene un criterio. Sea U^i la función que modela las preferencias de cada individuo con $i = 1, \dots, n$. Tal que:

$$U^i(a_j) \geq U^i(a_k)$$

El i -ésimo individuo cree que la alternativa a_j es al menos tan buena como a_k . Con esto se define una preferencia estricta, es decir, $a_j \geq a_k$, no puede ocurrir que $a_j \leq a_k$.

En este proceso se le asigna a los decisores un subconjunto de alternativas iniciales \mathcal{S} (subconjunto de referencia) para que las valoren comparándolas todas de dos en dos, es decir dadas dos alternativas saber cual es la preferida del grupo. Con esta ordenación de grupo se podrá establecer un orden y así poder formar una matriz de pares.

Para presentar la preferencia del grupo entre las dos alternativas, se define una variable binaria y:

$$Y_{a_j a_k} = \begin{cases} 1 & \text{si } a_j \geq_g a_k \\ 0 & \text{si } a_j <_g a_k \end{cases}$$

Si $a_j >_g a_k$ entonces $Y_{a_j a_k} = Y_{a_k a_j} = 1$.

Sean a_j y a_k dos alternativas del conjunto de referencia $\mathcal{S} \subset \mathcal{A}$. Se define para cada decisor $i = 1, \dots, n$.

$$X_{a_j a_k}^i = U^i(a_j) - U^i(a_k)$$

De este modo $a_j \geq_i a_k \rightarrow X_{a_j a_k}^i \geq 0$. El vector que representa para los decisores las preferencias individuales entre a_j y a_k es $X_{a_j a_k} = (X_{a_j a_k}^1, X_{a_j a_k}^2, \dots, X_{a_j a_k}^n)^T$.

Partiendo del subconjunto \mathcal{S} , donde $\mathcal{S} \subset \mathcal{A}$, que contiene m alternativas donde conocemos las preferencias individuales y las de grupo, llevaremos un proceso de agregación de preferencias. Si no se alcanza un consenso de grupo, un mecanismo arbitrario externo deberá resolver el conflicto. El método consiste en una ordenación global de las alternativas, basándose tan solo en las utilidades individuales.

Si comparamos cada par de alternativas en \mathcal{S} , obtendremos $m(m - 1)/2$ vectores ampliados de datos, que al agruparse dan lugar a la matriz de pares:

$$\Lambda = \begin{pmatrix} Y_{a_1 a_2} & Y_{a_1 a_3} & \cdots & Y_{a_{\frac{m(m-1)}{2}-1} a_{\frac{m(m-1)}{2}}} \\ X_{a_1 a_2}^1 & X_{a_1 a_3}^1 & \cdots & X_{a_{\frac{m(m-1)}{2}-1} a_{\frac{m(m-1)}{2}}}^1 \\ \vdots & \vdots & \ddots & \vdots \\ X_{a_1 a_2}^n & X_{a_1 a_3}^n & \cdots & X_{a_{\frac{m(m-1)}{2}-1} a_{\frac{m(m-1)}{2}}}^n \end{pmatrix}$$

El objetivo es conseguir la estimación de la probabilidad asociada a que la alternativa a_j sea preferida a a_k , que se denota $\pi_{a_j a_k}$. Se propone utilizar el siguiente criterio:

- $\pi_{a_j a_k} \geq 1/2 \rightarrow a_j \geq_g a_k$.
- $\pi_{a_j a_k} \leq 1/2 \rightarrow a_k \geq_g a_j$.
- $\pi_{a_j a_k} = 1/2 \rightarrow a_j \sim_g a_k$.

Estas probabilidades son las que finalmente nos permitirán ordenar las alternativas.

EJEMPLO

El conjunto finito \mathcal{A} lo formarían los 22 jugadores que forman la selección española, aunque para que el problema resulte más sencillo cogeremos un subconjunto de 5 jugadores (Pau Torres, Lorente, Jordi Alba, Busquets y Morata), los cuales llevamos estudiando durante todo el trabajo.

Cogemos una muestra de 7 encuestados (los siete primeros) para realizar el problema:

Encuestado	BUSQUETS	MORATA	JORDI ALBA	BUSQUETS	JORDI ALBA	MORATA	LLORENTE	BUSQUETS	PAU TORRES	BUSQUETS
1	1			1	1			1	1	
2	1			1	1			1		1
3	1		1		1			1	1	
4	1			1	1			1		1
5	1		1		1		1			1
6	1			1		1		1	1	
7	1			1	1			1	1	

Encuestado	PAU TORRES	JORDI ALBA	LLORENTE	JORDI ALBA	PAU TORRES	MORATA	LLORENTE	MORATA	PAU TORRES	LLORENTE
1	1			1	1			1	1	
2	1			1	1		1	1	1	
3	1			1	1		1		1	
4		1		1		1		1	1	
5		1	1			1	1			1
6	1			1	1			1	1	
7	1			1	1			1	1	

Sumamos las veces que cada encuestado haya votado a cada jugador, para que así se nos quede el orden de preferencia de cada encuestado:

Encuestado	PAU TORRES	BUSQUETS	JORDI ALBA	MORATA	LLORENTE
1	4	3	2	1	0
2	3	4	2	0	1
3	4	2	3	0	1
4	1	4	3	2	0
5	0	2	3	1	4
6	4	3	1	2	0
7	4	3	2	1	0

A continuación, sumamos el total de votos que tiene cada jugador para así establecer el orden de preferencia de grupo:

ÓRDEN	JUGADOR	TOTAL VOTOS
1	BUSQUETS	21
2	PAU TORRES	20
3	JORDI ALBA	16
4	MORATA	7
5	LLORENTE	6

Definimos las variables binarias, para presentar la preferencia de grupo:

$$Y_{BM} = 1, Y_{JB} = 0, Y_{JM} = 1, Y_{LB} = 0, Y_{PB} = 0,$$

$$Y_{PJ} = 1, Y_{LJ} = 0, Y_{PM} = 1, Y_{LM} = 0, Y_{PL} = 1,$$

Por otro lado, definimos las preferencias individuales:

$$X_{BM}^1 = 1, X_{JB}^1 = 0, X_{JM}^1 = 1, X_{LB}^1 = 0, X_{PB}^1 = 1, \quad (\text{Encuestado 1})$$

$$X_{PJ}^1 = 1, X_{LJ}^1 = 0, X_{PM}^1 = 1, X_{LM}^1 = 0, X_{PL}^1 = 1,$$

$$X_{BM}^2 = 1, X_{JB}^2 = 0, X_{JM}^2 = 1, X_{LB}^2 = 0, X_{PB}^2 = 0, \quad (\text{Encuestado 2})$$

$$X_{PJ}^2 = 1, X_{LJ}^2 = 0, X_{PM}^2 = 1, X_{LM}^2 = 1, X_{PL}^2 = 1,$$

$$X_{BM}^3 = 1, X_{JB}^3 = 1, X_{JM}^3 = 1, X_{LB}^3 = 0, X_{PB}^3 = 1, \quad (\text{Encuestado 3})$$

$$X_{PJ}^3 = 1, X_{LJ}^3 = 1, X_{PM}^3 = 1, X_{LM}^3 = 0, X_{PL}^3 = 0,$$

$$X_{BM}^4 = 1, X_{JB}^4 = 0, X_{JM}^4 = 1, X_{LB}^4 = 0, X_{PB}^4 = 0, \quad (\text{Encuestado 4})$$

$$X_{PJ}^4 = 0, X_{LJ}^4 = 0, X_{PM}^4 = 0, X_{LM}^4 = 0, X_{PL}^4 = 1,$$

Ordenamos las alternativas:

	PAU TORRES	BUSQUETS	JORDI ALBA	MORATA	LLORENTE
π_{BM}		1		0	
π_{JB}		1	0		
π_{JM}			1	0	
π_{LB}		1			0
π_{PB}	0,5	0,5			
π_{PJ}	1		0		
π_{LJ}			1		0
π_{PM}	1			0	
π_{LM}				1	0
π_{PL}	1				0
TOTAL	3,5	3,5	2	1	0

Finalmente obtenemos el ranking con un empate en el primer puesto entre Pau Torres y Busquets. El ranking es el siguiente:

1	Pau Torres
1	Busquets
3	Jordi Alba
4	Morata
5	Llorente

PROBLEMA DE ORDENAMIENTO LINEAL

El problema de ordenamiento lineal es un problema de optimización combinatoria. Se parte de una matriz cuadrada $C_{n \times n} = (C_{ij})$ no negativa y el objetivo es encontrar la permutación simultánea de filas y columnas que maximice la suma de los valores que están por encima de la diagonal.

Vamos a introducir un ejemplo de ordenamiento lineal de $n=3$, para poder entender mejor el problema. Cogemos a 3 jugadores Morata (M), Jordi Alba (J) y Llorente (L) y buscamos todas las combinaciones posibles entre estos tres, llegando a un total de $3!=6$. Las combinaciones son las siguientes:

e^1	Morata (M) – Jordi Alba (J) – Llorente (L)
e^2	Morata (M) – Llorente (L) – Jordi Alba (J)
e^3	Jordi Alba (J) – Morata (M) – Llorente (L)
e^4	Jordi Alba (J) – Llorente (L) – Morata (M)
e^5	Llorente (L) – Morata (M) – Jordi Alba (J)
e^6	Llorente (L) – Jordi Alba (J) – Morata (M)

Una vez hechas las combinaciones creamos una matriz inicial e^1 y sumamos la diagonal superior de esta:

		(1)	(2)	(3)
		M	J	L
(1)	M	0	15	20
(2)	J	19	0	20
(3)	L	14	14	0

La matriz inicial está representada por la permutación de identidad $e^1 = (M, J, L)$ y su aptitud $f(e^1) = 55$, que es la suma de la parte superior de la diagonal.

El propósito del problema es crear un ranking de estos tres jugadores. En el caso de la matriz e^1 , proponemos el siguiente ranking: 1º Morata, 2º Jordi Alba y 3º Llorente. El orden en que los jugadores se posicionan en el ranking se llama identidad.

Los tres factores que tiene la diagonal superior en este caso, dos pertenecen a Morata (Nº1 en el ranking) y uno a Jordi Alba (Nº2), esto supone que cuanto mayor sea la puntuación de Morata o Jordi Alba más alta será la suma de la diagonal superior, es decir, nos interesa que el jugador que más puntuación tenga esté en primera posición y así consecutivamente para poder lograr el máximo buscado. Morata al tener más factores dentro de este espacio aumentará la suma más que Jordi Alba. Este es el motivo por el que queremos maximizar la diagonal superior.

Por esta razón buscamos todas las combinaciones posibles y las sumamos, para ordenar a los jugadores de más puntuación a menos. La puntuación que sumamos por encima de la diagonal la llamaremos aptitud.

A continuación, formulamos todas las matrices con todas las combinaciones posibles, indicando su identidad y calculando su aptitud:
(Hemos numerado las posiciones de la matriz inicial para poder observar con mayor facilidad estas permutaciones).

		(1)	(3)	(2)
		M	L	J
(1)	M	0	20	15
(3)	L	14	0	14
(2)	J	19	20	0

$$e^2 = (M, L, J)$$

$$f(e^2) = 49$$

		(2)	(1)	(3)
		J	M	L
(2)	J	0	19	20
(1)	M	15	0	20
(3)	L	14	14	0

$$e^3 = (J, M, L)$$

$$f(e^3) = 59$$

		(2)	(3)	(1)
		J	L	M
(2)	J	0	20	19
(3)	L	14	0	14
(1)	M	15	20	0

$$e^4 = (J, L, M)$$

$$f(e^4) = 53$$

		(3)	(1)	(2)
		L	M	J
(3)	L	0	14	14
(1)	M	20	0	15
(2)	J	20	19	0

$$e^5 = (L, M, J)$$

$$f(e^5) = 43$$

		(3)	(2)	(1)
		L	J	M
(3)	L	0	14	14
(2)	J	20	0	19
(1)	M	20	15	0

$$e^6 = (L, J, M)$$

$$f(e^6) = 47$$

La solución óptima de este ejercicio sería $e^3 = (J, M, L)$, con aptitud $f(e^3) = 59$. Es decir, la población encuestada prefiere a Morata antes que a Llorente pero después que a Jordi Alba. Comprobamos con la encuesta inicial esta evidencia:

Jugadores	Jordi Alba	Morata	Morata	Llorente	Jordi Alba	Llorente
Votos	19	15	20	14	20	14

Observamos que Jordi Alba tiene más votos que Morata y Llorente y que Morata tiene más votos que Llorente. Por tanto, finalmente se nos quedaría el siguiente ranking: 1º Jordi Alba, 2º Morata y 3º Llorente.

El problema de ordenación lineal también se puede formular como un problema de programación lineal entera.

Sea $[n] = \{1, 2, \dots, n\}$ donde $n \geq 2$, y $C_{ij} \in \mathbb{R}_{\geq 0}$ para $i, j \in [n]$

$$\text{Max} \quad \sum_{i \neq j: i, j \in [n]} C_{ij} X_{ij}$$

$$X_{ij} + X_{ji} = 1 \quad \forall i < j: i, j \in [n] \quad (1)$$

$$X_{ij} + X_{jk} + X_{ki} \leq 2 \quad \forall i < j, i < k, j \neq k: i, j, k \in [n] \quad (2)$$

$$X_{ij} \in \{0, 1\} \quad \forall i \neq j: i, j \in [n] \quad (3)$$

Donde cada celda X_{ij} indica la relación de precedencia entre el elemento "i" y el elemento "j". Si la celda $X_{ij} = 1$ entonces "i" precede de "j" y 0 en caso contrario. Por otro lado, C_{ij} es la matriz del problema.

La función objetivo busca maximizar la diagonal superior de la matriz.

La primera restricción (tipo 1) captura el hecho de que o bien "i" antecede a "j", o "j" es el que antecede a "i".

La segunda asegura la transitividad, de modo que si "i" antecede a "j" y "j" antecede a "k" entonces "i" antecede a "k".

La tercera restricción nos dice que X_{ij} son variables binarias que toman valor 1 cuando i antecede a j en la permutación a

Problema de ordenamiento lineal con RStudio

Para resolver este problema vamos a usar el método simplex aplicado a RStudio, este método es un método iterativo que permite ir mejorando la solución en cada paso.

Usamos las librerías "lpsolve" y "matrix":

- El paquete "lpsolve" resuelve problemas de programación lineal y entera.
- El paquete "matrix" permite crear una matriz, pasando como input un vector numérico, de caracteres o lógico.

Una vez cargamos las librerías, escribimos la matriz que queremos ordenar y le ponemos nombre a las filas y columnas:

```
cij<-matrix(c(0,15,20,11,11,
              19,0,20,11,12,
              14,14,0,7,13,
              23,23,27,0,18,
              23,22,21,16,0),nrow = 5, ncol = 5, byrow = TRUE)
```

```
#Nombramos las filas y columnas
colnames(cij)<-c("M", "J", "L", "P", "B")
rownames(cij)<-c("M", "J", "L", "P", "B")
cij
```

```
> cij
  M J L P B
M 0 15 20 11 11
J 19 0 20 11 12
L 14 14 0 7 13
P 23 23 27 0 18
B 23 22 21 16 0
```

En segundo lugar, calculamos el tamaño de la matriz llamándolo “n” y llamamos también “matriz” a la matriz inicial transpuesta:

```
n<-ncol(cij);n
matriz<-t(cij);matriz
```

```
> n<-ncol(cij);n
[1] 5
> matriz<-t(cij);matriz
  M J L P B
M 0 19 14 23 23
J 15 0 14 23 22
L 20 20 0 27 21
P 11 11 7 0 16
B 11 12 13 18 0
```

A continuación, creamos un vector de 0 de tamaño $(n*n)-n$, esto es el número de dígitos que tiene la matriz inicial exceptuando los de la diagonal, es decir, en nuestro caso la matriz es 5 x 5 teniendo un número total de 25 dígitos, si le restamos los 5 dígitos que se encuentran en la diagonal nos quedamos con 20. Luego creamos un bucle que rellene el vector con los números de la matriz inicial:

```
f.objetivo<-c(rep(0,(n*n)-n))

# Bucle
t=1
s=0

for (i in 1:n){
  for (j in 1:n){
    s=s+1
    if(i != j){
      f.objetivo[t]=matriz[s]
      t=t+1
    }
  }
}
f.objetivo

> f.objetivo
[1] 15 20 11 11 19 20 11 12 14 14 7 13 23 23 27 18 23 22 21 16
```

El algoritmo simplex depende de la cantidad de variables y de restricciones. Por esta razón, calculamos el número de restricciones.

El primer tipo de restricción que es $X_{ij} + X_{ji} = 1$, es decir, si "i" es antecedente de "j", "j" no puede ser antecedente de "i". Por tanto, la variable X_{ij} es dependiente de X_{ji} . Esto implica que se puede reescribir la restricción de tipo 1 de la manera siguiente:

$$\begin{aligned} X_{ij} + X_{ji} &= 1, \forall i, j \\ X_{ji} &= 1 - X_{ij}, \forall i, j \end{aligned}$$

Por tanto, podríamos reducir a la mitad la cantidad de variables, debido a que basta con representar los casos donde $i < j$. Vale aclarar que también podría haberse elegido las variables X_{ij} donde $i > j$ y hubiera podido continuarse con implicaciones equivalentes. La cantidad de permutaciones de este tipo serían:

$$\frac{(n-1) * n}{2}$$

Qué adaptado a nuestro problema:

$$\frac{(5 - 1) * 5}{2} = 10$$

Todo esto aplicado a RStudio sería:

```
contador1 = 0
for(i in 1:n){
  for (j in 1:n){
    if (i<j){
      contador1=contador1+1
    }
  }
}
contador1
> contador1
[1] 10
```

En el caso de la restricción de tipo 2, que es $X_{ij} + X_{jk} + X_{ki} \leq 2$, si se agrupan todas las restricciones correspondientes a las $3! = 6$ permutaciones asociadas a la misma combinación de valores (i, j, k) para los cuales se cumple $i < j < k$ estas restricciones tendrían la forma siguiente:

$$\begin{aligned} X_{ij} + X_{jk} + X_{ki} &\leq 2, \text{ para } i, j, k \\ X_{ik} + X_{kj} + X_{ji} &\leq 2, \text{ para } i, k, j \\ X_{ji} + X_{ik} + X_{kj} &\leq 2, \text{ para } j, i, k \\ X_{jk} + X_{ki} + X_{ij} &\leq 2, \text{ para } j, k, i \\ X_{ki} + X_{ij} + X_{jk} &\leq 2, \text{ para } k, i, j \\ X_{kj} + X_{ji} + X_{ik} &\leq 2, \text{ para } k, j, i \end{aligned}$$

Se puede ver qué, aunque están escritas en distinto orden, las restricciones primera, cuarta y quinta son equivalentes, pues suman los valores X_{ij} , X_{jk} , X_{ki} y las restricciones segunda tercera y sexta también son equivalentes. De este modo, de las 6 restricciones anteriores solo serían necesarias dos. Por tanto, las restricciones de tipo 2 ahora serían dos para cada una de las combinaciones de valores. La cantidad de permutaciones de este tipo de quedaría:

$$\frac{n * (n - 1) * (n - 2)}{3}$$

Qué aplicado a nuestro problema:

$$\frac{5 * (5 - 1) * (5 - 2)}{3} = 20$$

En RStudio sería:

```
contador2=0
for (i in 1:n){
  for(j in 1:n){
    for (k in 1:n){
      if (i<j && i<k && j!=k)
        contador2 = contador2 +1
    }
  }
}
contador2
> contador2
[1] 20
```

Una vez calculado el número de restricciones por cada tipo, sumamos todas las restricciones del problema llamándole “dim”:

```
dim<-contador1+contador2; dim
> dim<-contador1+contador2; dim
[1] 30
```

A continuación creamos una matriz de 0 nxn:

```
vector<- matrix(0, nrow = n, ncol = (n)); vector
> vector<- matrix(0, nrow = n, ncol = (n)); vector
      [,1] [,2] [,3] [,4] [,5]
[1,]    0    0    0    0    0
[2,]    0    0    0    0    0
[3,]    0    0    0    0    0
[4,]    0    0    0    0    0
[5,]    0    0    0    0    0
```

```

t=1
for (i in 1:n) {
  for(j in 1:n){
    if (j != i){
      vector[i,j]<-t
      t=t+1
    }
  }
}
colnames(vector)<-c("M", "J", "L", "P", "B")
rownames(vector)<-c("M", "J", "L", "P", "B")
vector

```

```
> vector
```

	M	J	L	P	B
M	0	1	2	3	4
J	5	0	6	7	8
L	9	10	0	11	12
P	13	14	15	0	16
B	17	18	19	20	0

Volvemos a crear una matriz de 0 llamándola “mad”, pero esta vez si tamaño será el número de restricciones que hay como fila y el número de dígitos que hay en la matriz inicial (sin contar la diagonal) como columnas. Es decir, una matriz 20x30:

```
mad <- matrix(0, nrow = dim, ncol = ((n*n)-n));mad
```

```

> mad <- matrix(0, nrow = dim, ncol = ((n*n)-n));mad
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20]
[1,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[2,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[3,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[4,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[5,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[6,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[7,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[8,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[9,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[10,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[11,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[12,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[13,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[14,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[15,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[16,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[17,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[18,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[19,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[20,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[21,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[22,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[23,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[24,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[25,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[26,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[27,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[28,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[29,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[30,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Partiendo de que las 10 restricciones de tipo 1 son:

$$\begin{array}{ll}
 X_{MJ} + X_{JM} = 1 & (1) \\
 X_{MP} + X_{PM} = 1 & (3) \\
 X_{JL} + X_{LJ} = 1 & (5) \\
 X_{JB} + X_{BJ} = 1 & (7) \\
 X_{LB} + X_{BL} = 1 & (9) \\
 X_{ML} + X_{LM} = 1 & (2) \\
 X_{JP} + X_{PJ} = 1 & (4) \\
 X_{BP} + X_{PB} = 1 & (6) \\
 X_{LP} + X_{PL} = 1 & (8) \\
 X_{MB} + X_{BM} = 1 & (10)
 \end{array}$$

Vamos a rellenar las primeras 10 filas de la matriz de 10, que corresponderán a las 10 restricciones escritas anteriormente:

```

t=1
for( i in 1:n){
  for(j in 1:n){
    if (i < j ){
      mad[t, vector[i,j]] = 1
      mad[t, vector[j,i]] = 1
      t=t+1
    }
  }
}
mad

```

```

> mad
[ ,1] [ ,2] [ ,3] [ ,4] [ ,5] [ ,6] [ ,7] [ ,8] [ ,9] [ ,10] [ ,11] [ ,12] [ ,13] [ ,14] [ ,15] [ ,16] [ ,17] [ ,18] [ ,19] [ ,20]
[1,] 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[2,] 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
[3,] 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
[4,] 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
[5,] 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
[6,] 0 0 0 0 0 1 0 1 0 0 0 0 1 0 0 0 0 0 0 0
[7,] 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 0
[8,] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0
[9,] 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0
[10,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1
[11,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[12,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[13,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[14,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[15,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[16,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[17,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[18,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[19,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[20,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[21,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[22,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[23,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[24,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[25,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[26,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[27,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[28,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[29,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[30,] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
> |

```

Ejemplo: La primera fila de la matriz representa la restricción número 1 que es $X_{MJ} + X_{JM} = 1$, donde X_{MJ} se sitúa en la posición 1 de la matriz inicial (cij) y X_{JM} que se sitúa en la posición número 5.

Las 20 restricciones de tipo 2 son:

$$\begin{array}{ll}
 X_{MJ} + X_{JL} + X_{LM} \leq 2 & (1) \\
 X_{MJ} + X_{JB} + X_{BM} \leq 2 & (3) \\
 X_{ML} + X_{JP} + X_{PM} \leq 2 & (5) \\
 X_{MP} + X_{JM} + X_{PJ} \leq 2 & (7) \\
 X_{MP} + X_{PB} + X_{BM} \leq 2 & (9) \\
 X_{MB} + X_{LM} + X_{BL} \leq 2 & (11) \\
 X_{JL} + X_{LP} + X_{PJ} \leq 2 & (13) \\
 X_{JP} + X_{LJ} + X_{PL} \leq 2 & (15) \\
 X_{JB} + X_{LJ} + X_{BL} \leq 2 & (17) \\
 X_{LP} + X_{PB} + X_{BL} \leq 2 & (19) \\
 X_{MJ} + X_{JP} + X_{PM} \leq 2 & (2) \\
 X_{ML} + X_{JM} + X_{LJ} \leq 2 & (4) \\
 X_{ML} + X_{LB} + X_{BM} \leq 2 & (6) \\
 X_{MP} + X_{LM} + X_{PL} \leq 2 & (8) \\
 X_{MB} + X_{JM} + X_{BJ} \leq 2 & (10) \\
 X_{MB} + X_{PM} + X_{BP} \leq 2 & (12) \\
 X_{JL} + X_{LB} + X_{BJ} \leq 2 & (14) \\
 X_{JP} + X_{PB} + X_{BL} \leq 2 & (16) \\
 X_{JB} + X_{PJ} + X_{BP} \leq 2 & (18) \\
 X_{LB} + X_{PL} + X_{BP} \leq 2 & (20)
 \end{array}$$

Rellenamos de la fila 11 a la 30, que corresponden a las 20 restricciones de tipo 2

```

for( i in 1:n){
  for(j in 1:n){
    for(k in 1:n){
      if (i<j && i<k && j!=k){
        mad[t, vector[i,j]] = 1
        mad[t, vector[j,k]] = 1
        mad[t, vector[k,i]] = 1
        t=t+1
      }
    }
  }
}
mad

```

```

> mad
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20]
[1,]  1  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
[2,]  0  1  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0
[3,]  0  0  1  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0
[4,]  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0
[5,]  0  0  0  0  0  1  0  0  0  1  0  0  0  0  0  0  0  0  0  0
[6,]  0  0  0  0  0  0  1  0  0  0  0  0  0  1  0  0  0  0  0  0
[7,]  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  1  0  0
[8,]  0  0  0  0  0  0  0  0  0  0  1  0  0  0  1  0  0  0  0  0
[9,]  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  1  0
[10,] 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  1
[11,] 1  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0
[12,] 1  0  0  0  0  0  0  1  0  0  0  0  1  0  0  0  0  0  0  0
[13,] 1  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0  1  0  0  0
[14,] 0  1  0  0  1  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0
[15,] 0  1  0  0  0  0  0  0  0  0  1  0  1  0  0  0  0  0  0  0
[16,] 0  1  0  0  0  0  0  0  0  0  0  1  0  0  0  0  1  0  0  0
[17,] 0  0  1  0  1  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0
[18,] 0  0  1  0  0  0  0  0  0  1  0  0  0  0  1  0  0  0  0  0
[19,] 0  0  1  0  0  0  0  0  0  0  0  0  0  0  1  1  0  0  0  0
[20,] 0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0
[21,] 0  0  0  1  0  0  0  0  1  0  0  0  0  0  0  0  0  0  1  0
[22,] 0  0  0  1  0  0  0  0  0  0  0  1  0  1  0  0  0  0  0  1
[23,] 0  0  0  0  0  1  0  0  0  0  1  0  0  1  0  0  0  0  0  0
[24,] 0  0  0  0  0  1  0  0  0  0  0  1  0  0  0  0  0  1  0  0
[25,] 0  0  0  0  0  0  1  0  0  1  0  0  0  0  1  0  0  0  0  0
[26,] 0  0  0  0  0  0  1  0  0  0  0  0  0  0  1  0  1  0  0  0
[27,] 0  0  0  0  0  0  0  1  0  1  0  0  0  0  0  0  0  0  1  0
[28,] 0  0  0  0  0  0  0  0  1  0  0  0  1  0  0  0  0  0  0  1
[29,] 0  0  0  0  0  0  0  0  0  0  1  0  0  0  1  0  0  1  0  0
[30,] 0  0  0  0  0  0  0  0  0  0  0  1  0  1  0  0  0  0  1  1

```

Ejemplo: la restricción 1 de tipo 2 que es $X_{MJ} + X_{JL} + X_{LM} \leq 1$, esta representada en la fila 11 de la tabla anterior, nos indica que sus tres variables se sitúan en la posición 1, 6 y 9 de la matriz inicial cij.

```
restricciones<-mad
```


Los unos representan los números que van por encima de la matriz, por tanto ordenamos la matriz:

	P	B	J	M	L
P	0	1	1	1	1
B	0	0	1	1	1
J	0	0	0	1	1
M	0	0	0	0	1
L	0	0	0	0	0

Matriz óptima:

	P	B	J	M	L
P	0	18	23	23	27
B	16	0	22	23	21
J	11	12	0	19	20
M	11	11	15	0	20
L	7	13	14	14	0

Solución óptima (La suma de los valores que están por encima de la diagonal):

```
solucion$objval
> solucion$objval
[1] 216
```

Finalmente, el ranking de la preferencia de jugadores se quedaría:

1	Pau Torres
2	Busquets
3	Jordi Alba
4	Morata
5	Llorente

CONCLUSIÓN

A modo de cierre de este trabajo, podemos señalar que los tres problemas de orden que hemos realizado coinciden con el mismo resultado.

Como ya comentamos anteriormente en el apartado de “obtención de datos”, el 65% de la población encuestada es de la provincia de Castellón, el 25% de la provincia de Alicante y el otro 10% de otros lugares. Esto supone de gran relevancia en el resultado de los ejercicios.

En nuestro país, la mayoría de la gente prefiere un equipo de fútbol por delante de otros, normalmente suele ser el equipo que le pertenece al aficionado por cercanía territorial. Pero hay algo que pasa muy curioso, a parte de ese equipo preferido que hemos comentado anteriormente, la mayoría de la población española también es simpatizante del FC Barcelona o del Real Madrid, esto se debe a la gran importancia mundial que han tenido en la historia estos dos clubs.

La mayoría de los encuestados son de la provincia de Castellón, en esta provincia pertenece un equipo llamado Villarreal c.f, situado en la pequeña ciudad de Vila-real, la ciudad natal de Pau Torres. Pau es el defensa del equipo de fútbol de su ciudad y en este momento se encuentra en pleno auge de su carrera profesional. Estos motivos podrían ser determinantes para la primera posición del ranking en la que se encuentra Pau Torres.

En la segunda y tercera posición del Ranking se encuentra Busquets y Jordi Alba respectivamente, estos dos jugadores juegan en el FC Barcelona que como hemos comentado anteriormente es uno de los equipos más queridos de nuestro país.

Después, en penúltima y última posición tenemos a Morata y a Llorente, el primero de ellos juega en un equipo italiano, la Juventus, y luego tenemos a Llorente que es jugador del Atlético de Madrid. Estos equipos son secundarios para nuestra población encuestada, ya que no son clubs geográficamente cercanos y tampoco son el FC Barcelona o el Real Madrid. Este puede ser uno de los motivos por lo que estos jugadores han quedado en penúltima y última posición del ranking.

Para concluir, cabe destacar como en este trabajo de final de grado hemos comprobado la utilidad para ordenar de la programación matemática, en el caso del ordenamiento lineal o de los algoritmos, en el caso de la comparación por pares o el escalamiento multidimensional.

También hemos observado algunas ventajas y desventajas de los métodos estudiados. El escalamiento multidimensional es un método muy complejo para hacerlo a mano en cambio la comparación por pares y el ordenamiento lineal son más sencillos, siempre y cuando sean pocas variables las que analizar.

Dada la complejidad del escalamiento multidimensional, utilizamos el Spss que nos proporciona una solución muy visual y sencilla.

El método de comparación por pares es un método muy fácil, pero está diseñado para problemas sencillos y con pocas variables. Es decir, problemas del día a día.

Finalmente, el ordenamiento lineal, que en mi opinión es el problema más viable de los tres y gracias a la programación matemática podemos rankear todas las variables que queramos.



BIBLIOGRAFÍA

- Rajan, A. (2017, 14 octubre). ¿Es la acumulación de datos el petróleo del siglo XXI?. . . Piénsalo de nuevo. *BBC*. <https://www.bbc.com/mundo/noticias-41600755>
- García, M. (2021, 3 septiembre). *¿Qué es el análisis multivariado? Aprende a dominar datos y variables.* <https://www.crehana.com>.
<https://www.crehana.com/blog/desarrollo-web/analisis-multivariado/>
- Westreicher, G. (2021, 21 enero). *Ranking.* Economipedia.
<https://economipedia.com/definiciones/ranking.html>
- Torres, H. A. (2014b, julio 15). *ESCALAMIENTO MULTIDIMENSIONAL 1. INTRODUCCIÓN.* academia.edu.
https://www.academia.edu/7677902/ESCALAMIENTO_MULTIDIMENSIONAL_1_INTRODUCCI%C3%93N
- Escalamiento multidimensional.* (2021, 10 junio). IBM.
<https://www.ibm.com/docs/es/spss-statistics/SaaS?topic=features-multidimensional-scaling>
- Consultores, A. (2019, 30 septiembre). *Comparación por Pares para Tomar Decisiones.* Aiteco Consultores. <https://www.aiteco.com/comparacion-por-pares-para-tomar-decisiones//>
- Casas, F. M. G. (2012, 20 marzo). *EL ANÁLISIS DE ESCALAMIENTO MULTIDIMENSIONAL: UNA ALTERNATIVA Y UN COMPLEMENTO A OTRAS TÉCNICAS MULTIVARIANTES.* | *La Sociología en sus Escenarios.*
<https://revistas.udea.edu.co/index.php/ceo/article/view/11450>
- Martí, Reinelt & Duarte. (s. f.). *Linear Ordering Problem.* Linear Ordering Problem. Recuperado 2009, de <http://grafo.etsii.urjc.es/opticom/lolib/>

Métodos Cuantitativos & Econometría. (2014). *SPSS Escalamiento Multidimensional* [Vídeo]. YouTube.

https://www.youtube.com/results?search_query=escalamiento+multidimensional+spss

Despliegamiento multidimensional (PREFSCAL). (2021, 10 junio). IBM.

<https://www.ibm.com/docs/es/spss-statistics/SaaS?topic=categories-multidimensional-unfolding-prefscal>

Hdez., C. P. E. M. A. (2022, 23 mayo). *Método de comparación por pares*: Blogspot.

<http://masterpablocabrera.blogspot.com/2019/02/metodo-de-comparacion-por-pares.html>

Martin Grötschel, Michael Jünger, Gerhard Reinelt, (1984) A Cutting Plane Algorithm for the Linear Ordering Problem. *Operations Research* 32(6):1195-1220.

<https://doi.org/10.1287/opre.32.6.1195>

Chanas, S., Kobylański, P. A new heuristic algorithm solving the linear ordering problem. *Comput Optim Applic* **6**, 191–205 (1996).

<https://doi.org/10.1007/BF00249646>

Massó Ibarra, V. (2019, 19 junio). *UMH: El problema de ordenamiento lineal:*

Aplicación a un caso real. TFG. <http://dspace.umh.es/handle/11000/7845>

Rosete-Suárez, A. (2017, 1 febrero). *Reformulación eficiente del problema de*

programación lineal de agregación de rankings. redalyc. Recuperado 9 de julio

de 2018, de <https://www.redalyc.org/journal/3604/360458817004/html/>

Cebreiro, J., Mendiburu, A., & Lozano, J. A. (2014, enero). *The Linear Ordering Problem*

Revisited. UNIVERSITY OF THE BASQUE COUNTRY.

<https://addi.ehu.es/bitstream/handle/10810/11178/tr14->

[1b.pdf;jsessionid=5015F9DDC72056782744FDCA168B9563?sequence=4](https://addi.ehu.es/bitstream/handle/10810/11178/tr14-1b.pdf;jsessionid=5015F9DDC72056782744FDCA168B9563?sequence=4)