



Universidade de Vigo

Trabajo Fin de Máster

Evaluación del impacto en los costes de los diversos artículos en modelos cooperativos de inventario con múltiples agentes

Emma Domínguez González

Máster en Técnicas Estadísticas

Curso 2024-2025

Propuesta de Trabajo Fin de Máster

<p>Título en galego: Evaluación do impacto nos costes dos diversos artigos en modelos cooperativos de inventario con múltiples agentes</p>
<p>Título en español: Evaluación del impacto en los costes de los diversos artículos en modelos cooperativos de inventario con múltiples agentes</p>
<p>English title: Evaluation of the Impact on Costs of Various Items in Cooperative Inventory Models with Multiple Agents</p>
<p>Modalidad: Modalidad A</p>
<p>Autor/a: Emma Domínguez González</p>
<p>Director/a: Ignacio García Jurado, Universidad de A Coruña</p>
<p>Tutor/a: Ignacio García Jurado, Universidad de A Coruña</p>
<p>Breve resumen del trabajo:</p> <p>Este Trabajo Fin de Máster (TFM) se centra en los modelos cooperativos de inventario con múltiples agentes. Los objetivos del trabajo incluyen una revisión de los modelos de inventario de revisión continua en un contexto cooperativo con múltiples agentes. Además, se adaptará la regla propuesta por Fiestras-Janeiro et al. (2024) para evaluar el impacto de los diversos artículos en el modelo básico EOQ, considerando varios artículos, múltiples agentes y la posibilidad de costes fijos eximibles. También se desarrollará un código en R para calcular las reglas resultantes, y se analizarán las propiedades de estas reglas desde un enfoque teórico, aplicándolas a ejemplos prácticos.</p>
<p>Recomendaciones:</p> <p>Es recomendable haber cursado las siguientes materias del MTE: Introducción a la teoría de juegos, Juegos cooperativos y Medelos interactivos de la IO. También es recomendable haber cursado un grado en matemáticas.</p>
<p>Otras observaciones:</p> <p>Dado que los objetivos son amplios, se deja abierta la posibilidad de enfocar el trabajo según los intereses de quien lo realice, permitiendo profundizar en áreas específicas, ya sea en la revisión teórica de los modelos, la adaptación de la regla de reparto, el desarrollo del código o el análisis de las propiedades de las reglas propuestas.</p>

Don/doña Ignacio García Jurado, Catedrático de la Universidad de A Coruña informa que el Trabajo Fin de Máster titulado

Evaluación del impacto en los costes de los diversos artículos en modelos cooperativos de inventario con múltiples agentes

fue realizado bajo su dirección por don/doña Emma Domínguez González para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, da su conformidad para su presentación y defensa ante un tribunal. Además, don/doña Ignacio García Jurado y don/doña Emma Domínguez González autorizan a la publicación de la memoria en el repositorio de acceso público asociado al Máster en Técnicas Estadísticas.

En Santiago de Compostela, a 13 de Febrero de 2025.

El/la director/a:
Don/doña Ignacio García Jurado

El/la tutor/a:
Don/doña Ignacio García Jurado

El/la autor/a:
Don/doña Emma Domínguez González

Declaración responsable. Para dar cumplimiento a la Ley 3/2022, de 24 de febrero, de convivencia universitaria, referente al plagio en el Trabajo Fin de Máster (Artículo 11, [Disposición 2978 del BOE núm. 48 de 2022](#)), **el/la autor/a declara** que el Trabajo Fin de Máster presentado es un documento original en el que se han tenido en cuenta las siguientes consideraciones relativas al uso de material de apoyo desarrollado por otros/as autores/as:

- Todas las fuentes usadas para la elaboración de este trabajo han sido citadas convenientemente (libros, artículos, apuntes de profesorado, páginas web, programas, . . .)
- Cualquier contenido copiado o traducido textualmente se ha puesto entre comillas, citando su procedencia.

- Se ha hecho constar explícitamente cuando un capítulo, sección, demostración, . . . sea una adaptación casi literal de alguna fuente existente.

Y, acepta que, si se demostrara lo contrario, se le apliquen las medidas disciplinarias que correspondan.

Índice

Resumen	VIII
Introducción	XI
1. Conceptos previos	1
1.1. Modelos de inventario	1
1.2. Teoría de juegos cooperativos	2
1.2.1. Juegos de utilidad transferible	2
1.2.2. Núcleo de un juego TU	3
1.2.3. Valor de Shapley de un juego TU	5
1.2.4. Valor de Owen de un juego TU	10
2. Modelos deterministas EOQ	17
2.1. Modelo básico determinista	17
2.1.1. Modelo con costes eximibles	19
2.1.2. Modelo sin costes eximibles	20
2.2. Modelo determinista con múltiples artículos	21
2.2.1. Modelo con costes eximibles	21
2.2.2. Modelo sin costes eximibles	24
2.2.3. Ejemplo ilustrativo	26
2.3. Modelo determinista con múltiples agentes	34

2.3.1. Ejemplo ilustrativo	36
2.4. Modelo determinista con múltiples artículos y agentes	38
2.4.1. Modelo con costes eximibles	39
2.4.2. Modelo sin costes eximibles	41
2.4.3. Ejemplo ilustrativo	42
Conclusiones	51

Resumen

El presente trabajo aborda el análisis de modelos de inventario deterministas en el contexto de la teoría de juegos cooperativos, explorando su aplicación a problemas de distribución de costes. Se presentan los conceptos fundamentales necesarios para el estudio, incluyendo una introducción a la teoría de juegos cooperativos, los juegos de utilidad transferible (TU) y sus principales soluciones, como el núcleo, el valor de Shapley y el valor de Owen.

En el marco de los modelos de inventario, se analizan diferentes configuraciones del modelo EOQ (*Economic Order Quantity*), comenzando por el modelo básico determinista y extendiéndose a casos con múltiples artículos y agentes. Dos variantes clave serán examinadas: modelos con costes eximibles, que contemplan eximir a ciertas coaliciones de los costes fijos de pedido y modelos sin costes eximibles, que no lo contemplan.

A lo largo del trabajo, se estudiará un ejemplo ilustrativo que destaca las implicaciones de los modelos y sus soluciones en la práctica. Además, se analizará cómo las diferentes reglas de distribución de costes afectan al reparto entre agentes o artículos, considerando tanto la ecuanimidad como la estabilidad del sistema.

Abstract

This work addresses the analysis of deterministic inventory models in the context of cooperative game theory, exploring their application to cost distribution problems. The fundamental concepts necessary for the study are presented, including an introduction to cooperative game theory, transferable utility (TU) games, and their main solutions, such as the core, the Shapley value, and the Owen value.


Within the framework of inventory models, different configurations of the EOQ (*Economic Order Quantity*) model are analyzed, starting with the basic deterministic model and extending to cases involving multiple items and agents. Two key variants are examined: models with exemptible costs, which include constraints allowing certain coalitions to be excluded from cost contributions, and models without exemptible costs, where all coalitions fully participate.

Throughout the work, an illustrative example is studied to highlight the implications of the models and their solutions in practice. Additionally, the impact of different cost distribution rules on the allocation among agents or items is analyzed, considering both fairness and system efficiency.

Introducción

La gestión de sistemas de inventarios es un campo fundamental dentro de la investigación operativa, cuyo objetivo es optimizar la producción, almacenamiento y distribución de productos, equilibrando la oferta y la demanda para minimizar los costos operativos. Entre los modelos más relevantes se encuentra el modelo de Cantidad Económica de Pedido (EOQ), ampliamente utilizado para determinar el tamaño y la frecuencia de los pedidos que minimizan los costos de almacenamiento y reposición. Este modelo ha sido la base de varios desarrollos, desde su formulación clásica hasta variaciones que abordan situaciones con demandas deterministas y aleatorias, así como políticas de revisión continua o periódica.

Este trabajo se inspira fundamentalmente en el artículo [Fiestras-Janeiro et al. \(2024\)](#), que analiza el uso de modelos EOQ en el marco de la teoría de juegos cooperativos. En particular, el artículo se enfoca en modelos en los que es posible eximir a ciertas coaliciones de los costes fijos de pedido. Sin embargo, en este trabajo se profundiza en la comparación entre modelos con costes eximibles y modelos sin costes eximibles, lo cual no es tratado en el artículo citado. Este análisis permitirá evaluar cómo la exención de costes afecta a la distribución de los mismos entre los agentes.

Además, en el artículo mencionado no se aborda el cálculo del valor de Owen, una solución importante en el contexto de juegos cooperativos. En este trabajo, se propone una implementación práctica para aproximar este valor, junto con otras soluciones clásicas como el núcleo y el valor de Shapley. A través del desarrollo de un código propio en el programa , se proporcionará una herramienta útil para la implementación de estas soluciones en modelos de inventario deterministas, permitiendo su aplicación a escenarios reales de gestión de costes.

Este trabajo tiene como objetivo ofrecer una visión completa de cómo los modelos deterministas de inventario, combinados con las herramientas de la teoría de juegos cooperativos, pueden ser utilizados para resolver problemas complejos de distribución de costes, proponiendo tanto un análisis teórico como una herramienta práctica para su implementación en la gestión de inventarios.

Capítulo 1

Conceptos previos

1.1. Modelos de inventario

La gestión de los sistemas de inventarios constituye una disciplina clave dentro de la investigación operativa, con el objetivo de optimizar la producción, almacenamiento y distribución de bienes. Estos sistemas permiten a las empresas mantener niveles adecuados de stock para satisfacer la demanda de los clientes, minimizando al mismo tiempo los costos asociados al mantenimiento, reposición y posibles desabastecimientos. Desde los primeros avances de Harris y Taft en el siglo XX, hasta los modelos probabilísticos desarrollados posteriormente, los sistemas de inventarios han evolucionado para adaptarse a las necesidades dinámicas de las organizaciones modernas.

Entre los modelos clásicos de inventarios, destaca el modelo de Cantidad Económica de Pedido (EOQ), que proporciona una solución matemática para determinar el tamaño y momento óptimos de los pedidos, minimizando costos de almacenamiento y reposición. Variaciones de este modelo permiten abordar distintos escenarios, como demandas deterministas o aleatorias, y políticas de revisión continua o periódica.

En este trabajo, el enfoque principal será el estudio de diferentes tipos de modelos EOQ y la implementación de soluciones de reparto asociadas a ellos, particularmente en contextos cooperativos. Para ello, utilizaremos herramientas provenientes de la teoría de juegos cooperativos, específicamente los juegos de utilidad transferible (juegos TU), que permiten modelar y analizar situaciones en las que agentes cooperan para reducir costos.

Antes de profundizar en los modelos EOQ y sus aplicaciones, será necesario definir conceptos clave de los juegos TU, como sus características principales, las soluciones de reparto y los criterios de justicia utilizados para asignar los costos entre los agentes. Esto proporcionará una

base sólida para analizar las soluciones propuestas en los distintos escenarios planteados.

1.2. Teoría de juegos cooperativos

Los juegos cooperativos constituyen una rama fundamental de la teoría de juegos, centrada en el análisis de situaciones donde los participantes tienen la posibilidad de coordinarse y tomar acuerdos vinculantes. A diferencia de los juegos no cooperativos, donde se busca maximizar beneficios individuales bajo restricciones estratégicas, los juegos cooperativos se ocupan de la colaboración entre agentes para alcanzar objetivos comunes. Una de las principales características de estos modelos es la formación de coaliciones, que son grupos de jugadores que acuerdan actuar juntos para mejorar sus resultados colectivos.

En este contexto, los juegos de utilidad transferible (TU) son especialmente importantes, ya que permiten la redistribución de beneficios dentro de una coalición. En los apartados siguientes, realizaremos una revisión de algunos conceptos clave para el análisis de los juegos TU, como el núcleo, el valor de Shapley, y el valor de Owen, los cuales constituyen herramientas fundamentales para garantizar repartos equitativos y sostenibles entre los jugadores en distintos escenarios de cooperación.

1.2.1. Juegos de utilidad transferible

Los juegos de utilidad transferible (TU) constituyen una de las clases más importantes dentro de los juegos cooperativos, pues introducen la posibilidad de redistribuir libremente los beneficios generados por una coalición entre sus miembros. Este tipo de juegos se centra en situaciones donde los jugadores pueden coordinarse para alcanzar un resultado conjunto que supere lo que obtendrían actuando de manera independiente. Una vez obtenidos estos beneficios colectivos, su transferencia permite diseñar reglas de reparto justas y eficientes.

El concepto de coalición juega un papel fundamental en los juegos TU. Cada subgrupo de jugadores puede asociarse para obtener un valor colectivo, representado por una función característica que asigna un beneficio a cada coalición. Este valor es independiente de las decisiones tomadas por los jugadores que están fuera del subgrupo.

A continuación se presentan una serie de definiciones que permitirán establecer el marco teórico necesario para introducir herramientas clave como el núcleo, el valor de Shapley y el valor de Owen en los apartados siguientes.

Definición 1. Un **juego TU** es un par (N, v) , donde N es un conjunto finito de jugadores $N = \{1, \dots, n\}$ y $v : 2^N \rightarrow \mathbb{R}$ es una función que verifica $v(\emptyset) = 0$.

La función v se denomina la **función característica** del juego. Para cada coalición $S \subseteq N$, $v(S)$ denota los **beneficios**¹ asociados a los jugadores de S , independientemente de cómo actúe el resto de jugadores. En lo que sigue, se denotará por $G(N)$ el conjunto de todos los juegos TU con conjunto de jugadores N . Es frecuente identificar el juego $(N, v) \in G(N)$, con su función característica: $v \in G(N)$.

Definición 2. Un juego $v \in G(N)$ se dice **superadictivo** si para cada par de coaliciones $S, T \subseteq N$ tales que $S \cap T = \emptyset$, se verifica: $v(S) + v(T) \leq v(S \cup T)$.

En otras palabras, un juego (N, v) se dice superaditivo cuando la cooperación entre los jugadores resulta ventajosa, es decir, la unión de coaliciones disjuntas nunca es perjudicial.

Como ya se mencionó, la teoría de los juegos TU se enfoca en la definición de reglas para asignar los beneficios o costes, y en la identificación de conjuntos de posibles asignaciones que sean aceptables para todos los jugadores involucrados. Para alcanzar este objetivo, se pueden adoptar diferentes enfoques. Uno de ellos está basado en la **estabilidad**: se busca encontrar conjuntos de asignaciones que sean estables, en el sentido de que el acuerdo final debe formar parte de dicho conjunto. Ejemplos de soluciones que siguen esta perspectiva son el **núcleo**, los conjuntos estables y el conjunto de negociación. Otro enfoque alternativo es la **ecuanimidad**, cuyo objetivo es desarrollar reglas que distribuyan los beneficios de manera justa entre los jugadores de la coalición. Conceptos como el **valor de Shapley**, el nucleolus, el τ -valor y el **valor de Owen** son ejemplos donde la ecuanimidad está estrechamente relacionada con las propuestas de reparto de costes. A continuación, se desarrollarán de manera más detallada tres de esos conceptos mencionados: el núcleo, el valor de Shapley y el valor de Owen de un juego TU.

1.2.2. Núcleo de un juego TU

El núcleo de un juego de utilidad transferible (TU) proporciona una solución que refleja la estabilidad interna de una coalición. En términos sencillos, el núcleo describe el conjunto de posibles asignaciones de beneficios que no serán rechazadas por ningún grupo de jugadores dentro del juego. Es decir, no habrá ninguna coalición que pueda mejorar su situación mediante un acuerdo distinto que no involucre a los jugadores fuera de ella.

Este concepto refleja la idea de estabilidad, garantizando que las soluciones no sean vulnerables a subcoaliciones que puedan formar acuerdos más favorables.

Definición 3. Dado un juego $v \in G(N)$, una **imputación** de v será un vector $\mathbf{x} \in \mathbb{R}^N$ verificando:

¹Las definiciones y/o resultados de este trabajo, podrían reescribirse considerando una función de **costes** $c(S)$, en lugar de beneficios.

1. $x_i \geq v(i), \forall i \in N$.
2. $\sum_{i \in N} x_i = v(N)$.

Denotando por $I(v)$ el conjunto de imputaciones del juego v , se tendrá que $I(v) \neq \emptyset$ cuando v sea superadictivo y contendrá las asignaciones de $v(N)$ que verifican la racionalidad individual. Como se define a continuación, el núcleo estará formado por las imputaciones que verifican que para cada coalición $S \subseteq N$, la suma $\sum_{i \in N} x_i \geq v(S)$.

Definición 4. Sea $v \in G(N)$, el **núcleo del juego** v , también conocido como **core**, vendrá dado por:

$$C(v) = \{ \mathbf{x} \in I(v) : \sum_{i \in N} x_i \geq v(S) \quad \forall S \in 2^N \} \quad (1.1)$$

Como conclusión, las asignaciones que pertenecen al núcleo garantizan que a cada coalición se le asigne un coste que no exceda el que podría asegurar actuando de manera independiente. Esto hace que dichas asignaciones sean estables, ya que ninguna coalición tendría incentivos para rechazar el reparto propuesto.

Además del razonamiento anterior, la relación entre el núcleo y el concepto de estabilidad se puede estudiar a través de una serie de resultados que se enumeran a continuación y cuyas demostraciones pueden consultarse en [Casas-Méndez et al. \(2012\)](#). Para la comprensión de éstos, será necesario conocer las definiciones de: **dominancia** entre dos imputaciones y un **juego equilibrado**.

Definición 5. Sea $v \in G(N)$ un juego TU y sean $\mathbf{x}, \mathbf{y} \in I(v)$ dos imputaciones. Se dice que \mathbf{x} **domina** \mathbf{y} a través de la coalición $S \subseteq N$ si:

1. $x_i > y_i, \forall i \in S$
2. $\sum_{i \in S} x_i \leq v(S)$

En particular, si \mathbf{x} domina \mathbf{y} a través de S , todos los agentes de la coalición S preferirán la primera asignación. Cuando no exista \mathbf{x} que sea preferible a \mathbf{y} a través de una coalición $S \subseteq N$, se dirá que \mathbf{y} es **no dominada**. Para buscar la estabilidad en una imputación, interesará perseguir aquellas que sean no dominadas. Los elementos del núcleo pueden caracterizarse de la siguiente manera:

Proposición 6. Sea $v \in G(N)$ un juego TU, se tendrá que:

- Si $\mathbf{x} \in C(v)$, necesariamente \mathbf{x} será **no dominada**.

- Si además v es **superaditivo**, entonces $C(v) = \{x \in I(v) : x \text{ es no dominada}\}$.

Para finalizar este apartado, la definición de **juego equilibrado**, determinará, a través del resultado de Bondareva-Shapley, una condición necesaria y suficiente para saber cuándo el core es no vacío.

Definición 7. Una familia de **coaliciones** $\mathcal{F} \subseteq 2^N \setminus \{\emptyset\}$ se dice **equilibrada** si existen números reales positivos $\{y_S : S \in \mathcal{F}\}$, denominados **coeficientes de equilibrio**, que cumplen $\sum_{S \in \mathcal{F}, i \in S} y_S = 1, \forall i \in N$.

Definición 8. Un juego $v \in G(N)$ se dirá que es **equilibrado** si para toda familia de coaliciones equilibradas \mathcal{F} con coeficientes de equilibrio $\{y_S : S \in \mathcal{F}\}$, se tiene que $\sum_{S \in \mathcal{F}} y_S \cdot v(S) \leq v(N)$.

Teorema 9. (Bondareva-Shapley). Sea $v \in G(N)$

$$C(v) \neq \emptyset \iff v \text{ es un juego equilibrado} \quad (1.2)$$

1.2.3. Valor de Shapley de un juego TU

El valor de Shapley es una de las soluciones más importantes en la teoría de juegos cooperativos, diseñada para asignar beneficios o costes de manera justa entre los jugadores de una coalición. Este concepto se basa en las contribuciones marginales que cada jugador aporta a las diferentes coaliciones a las que puede pertenecer, proporcionando una asignación que combina ecuanimidad y eficiencia.

A continuación, se definirá formalmente el valor de Shapley y se introducirá una breve explicación sobre cómo puede aproximarse mediante simulación cuando el número de jugadores es considerablemente grande, lo que resultará especialmente útil en el desarrollo de la sección 2.3 del presente trabajo.

Antes de conocer la definición del valor de Shapley, es necesario establecer ciertas propiedades que pueden ser impuestas sobre cualquier **regla de asignación** de costes, que no es más que una aplicación $f : G(N) \rightarrow \mathbb{R}^N$.

1. Se dice que la regla de asignación f es **eficiente** si:

$$\sum f_i(v) = v(N) \quad \forall v \in G(N) \quad (1.3)$$

2. La asignación f verifica la propiedad de **jugador nulo** si $\forall v \in G(N)$ y para todo jugador $i \in N$ nulo, es decir, aquel que verifica $v(S \cup \{i\}) = v(S)$ con $S \subseteq N$, se tiene $f_i(v) = 0$.

3. f será **simétrica** cuando $\forall v \in G(N)$ y para todo par de jugadores $i, j \in N$ simétricos, es decir, aquellos que verifican $v(S \cup \{i\}) = v(S \cup \{j\})$ con $S \subseteq N \setminus \{i, j\}$, se cumple $f_i(v) = f_j(v)$.
4. Podrá decirse que la asignación f verifica la propiedad de **aditividad** si para cada $v, w \in G(N)$ verifica $f(v) + f(w) = f(v + w)$.

Estamos entonces en condiciones de definir el valor de Shapley en base al siguiente resultado.

Teorema 10. *Existe una única regla de asignación $\phi(v) : G(N) \rightarrow \mathbb{R}^N$ que satisface las propiedades de eficiencia, jugador nulo, simetría y aditividad. Para todo juego $v \in G(N)$ y cada jugador $i \in N$, dicha regla se puede obtener como:*

$$\phi_i(v) = \sum_{S \subseteq N \setminus i} \frac{s!(n-s-1)!}{n!} (v(S \cup i) - v(S)) \quad (1.4)$$

siendo los cardinales $s = |S|$ y $n = |N|$.

Se define el **valor de Shapley** de un juego TU como la regla 1.4 identificada en el teorema anterior. Alternativamente, se puede definir el valor de Shapley asociado a un juego v a través del concepto de vector de contribución marginal.

Definición 11. Sea $\pi \in \Pi(N)$ una posible ordenación de los elementos de N . Dado un jugador $i \in N$, si el conjunto de jugadores $A^\pi(i)$ denota aquellos jugadores “anteriores” según el orden π : $A^\pi(i) = \{j \in N : \pi(j) < \pi(i)\}$, el **la contribución marginal de i** asociada al orden π se define como:

$$v(A^\pi(i) \cup \{i\}) - v(A^\pi(i)) \quad (1.5)$$

En la práctica, cada vector de contribuciones marginales representa una distribución del coste y/o beneficio total $v(N)$ que asigna a cada jugador su contribución marginal a los jugadores que le preceden, de acuerdo con el orden de la permutación π . De esta manera, el valor de Shapley de un juego $v \in G(N)$:

$$\phi_i(v) = \frac{1}{n!} \sum_{\pi \in \Pi(N)} (v(A^\pi(i) \cup \{i\}) - v(A^\pi(i))); \quad \forall i \in N \quad (1.6)$$

Las fórmulas 1.4 y 1.6 evidencian que calcular el valor de Shapley puede ser un proceso computacionalmente exigente, especialmente en juegos con un gran número de jugadores. Para abordar esta dificultad, se pueden utilizar técnicas de muestreo que permitan aproximar el valor de Shapley de manera computacionalmente viable. A continuación, se presenta una propuesta de

implementación basada en el enfoque descrito en [Castro et al. \(2009\)](#). Los pasos a seguir serán los siguientes:

1. Se extrae una muestra de m permutaciones $\{\pi_j\}_{j=1}^m$ de la población con reemplazamiento. En `R`, se puede hacer con ayuda de la función `sample`.
2. Para cada permutación π_j se calcula la contribución marginal de todos los jugadores.
3. La estimación del valor de Shapley de cada jugador se obtendrá como la media muestral, componente a componente, de las contribuciones marginales obtenidas sobre la muestra.

Ejemplo 12. Se considera el conjunto $N = \{1, 2, 3\}$ de jugadores que colaboran para generar un beneficio común. La función característica $v(S)$, que asigna un valor a cada coalición S se define como el beneficio generado por los jugadores que forman la coalición.

Coalición S	Valor $v(S)$
$\{1\}$	0
$\{2\}$	0
$\{3\}$	0
$\{1, 2\}$	1
$\{1, 3\}$	1
$\{2, 3\}$	0
$\{1, 2, 3\}$	1

Tabla 1.1: Función característica del juego con $N = \{1, 2, 3\}$.

Si el objetivo es calcular el **valor de Shapley exacto**, siguiendo la fórmula 1.6, será necesario obtener primero las **contribuciones marginales** 1.5.

Si $N = \{1, 2, 3\}$, las posibles ordenaciones $\{\pi_j\}_{j=1}^6$ serán $\{1, 3, 2\}$, $\{2, 1, 3\}$, $\{1, 2, 3\}$, $\{2, 3, 1\}$, $\{3, 1, 2\}$, $\{3, 2, 1\}$. Para el jugador $i = 1$ en la permutación $\pi_2 = \{2, 1, 3\}$, el conjunto de jugadores que llegan antes que él a la permutación es $A^{\pi_2}(1) = \{2\}$. Así entonces, se puede calcular la contribución marginal de acuerdo a la ordenación π_2 como:

$$v(A^{\pi_2}(1) \cup \{1\}) - v(A^{\pi_2}(1)) = v(\{2, 1\}) - v(\{2\}) = 1 - 0 = 1$$

Para los restantes jugadores de π_2 se tienen $A^{\pi_2}(2) = \emptyset$ y $A^{\pi_2}(3) = \{1, 2\}$, por lo tanto:

$$v(A^{\pi_2}(2) \cup \{2\}) - v(A^{\pi_2}(2)) = v(\{2\}) - v(\{\emptyset\}) = 0 - 0 = 0$$

$$v(A^{\pi_2}(3) \cup \{3\}) - v(A^{\pi_2}(3)) = v(\{1, 2, 3\}) - v(\{1, 2\}) = 1 - 1 = 0$$

Realizando este procedimiento para cada una de las 6 permutaciones posibles, se obtienen las siguientes contribuciones marginales recogidas en la Tabla 1.2:

Coaliciones π_j	$i = 1$	$i = 2$	$i = 3$
{1, 2, 3}	0	1	0
{1, 3, 2}	0	0	1
{2, 1, 3}	1	0	0
{2, 3, 1}	1	0	0
{3, 1, 2}	1	0	0
{3, 2, 1}	1	0	0

Tabla 1.2: Contribuciones marginales de cada jugador en las posibles permutaciones de $N = \{1, 2, 3\}$.


Sin más que utilizar la fórmula 1.6, se obtienen las asignaciones de los beneficios a cada jugador:

$$\Phi_1(v) = \frac{1}{3!}(0 + 0 + 1 + 1 + 1 + 1) = \frac{2}{3}$$

$$\Phi_2(v) = \frac{1}{3!}(1 + 0 + 0 + 0 + 0 + 0) = \frac{1}{6}$$

$$\Phi_3(v) = \frac{1}{3!}(0 + 1 + 0 + 0 + 0 + 0) = \frac{1}{6}$$

De acuerdo con la eficiencia 1.3 $\sum_{i=1}^3 \Phi_i(v) = v(N) = 1$

Un posible código para ejecutar el anterior proceso en , podría ser:

```

1 # Definir la función característica del juego
2 v <- function(S) {
3   if (setequal(S, c(1))) return(0)
4   if (setequal(S, c(2))) return(0)
5   if (setequal(S, c(3))) return(0)
6   if (setequal(S, c(1, 2))) return(1)
7   if (setequal(S, c(1, 3))) return(1)
8   if (setequal(S, c(2, 3))) return(0)
9   if (setequal(S, c(1, 2, 3))) return(1)
10  return(0)
11 }
12
13 # Lista de jugadores
14 players <- c(1, 2, 3)
15
16 # Generar todas las permutaciones
17 library(gtools)
18 permutations <- permutations(length(players), length(players), players)

```

```

19
20 # Calcular el valor de Shapley exacto
21 shapley_values <- rep(0, length(players))
22 names(shapley_values) <- players
23
24 for (player in players) {
25
26   marginal_contributions <- c()
27
28   for (perm in 1:nrow(permutations)) {
29
30     # Obtener la permutación actual
31     current_perm <- permutations[perm,]
32
33     # Determinar los jugadores previos a "player" en la permutación
34     player_index <- which(current_perm == player)
35     predecessors <- current_perm[1:(player_index - 1)]
36
37     # Calcular la contribución marginal
38     with_player <- union(predecessors, player)
39     without_player <- predecessors
40     marginal_contributions <- c(marginal_contributions,
41                               v(with_player) - v(without_player))
42
43   }
44   # Promediar las contribuciones marginales
45   shapley_values[player] <- mean(marginal_contributions)
46 }
47
48 shapley_values

```

```

1           1           2           3
2 0.6666667 0.1666667 0.1666667

```

```

1 # Comprobación eficiencia
2 sum(shapley_values)

```

```

1 [1] 1

```

Si se deseara ahora **aproximar** el valor de Shapley a través del algoritmo descrito anteriormente, podría utilizarse este otro código:

```

1 n_iterations = 100000
2 n <- length(players)
3 shapley_aprox <- numeric(n)
4
5 # Iterar a través de un número grande de simulaciones
6 for (iter in 1:n_iterations) {
7
8   permutation <- sample(players)
9   cost_prev <- 0
10
11   # Calcular contribuciones marginales para la permutación
12   for (j in 1:n) {

```

```

13   current_player <- permutation[j]
14   coalition <- permutation[1:j] # Coalición actual en la permutación
15
16   # Evaluar la función característica para la coalición actual
17   cost_current <- v(coalition)
18
19   # Contribución marginal del jugador actual
20   marginal_contribution <- cost_current - cost_prev
21   shapley_aprox [current_player] <- ( shapley_aprox [current_player]
22                                     + marginal_contribution )
23
24   # Actualizar el costo previo
25   cost_prev <- cost_current
26 }
27
28 # Dividir por el número de iteraciones para obtener el promedio
29 shapley_aprox <- shapley_aprox / n_iterations
30 shapley_aprox

```

```
1 [1] 0.66529 0.16899 0.16572
```

Para finalizar este apartado, se presentará un resultado relevante que establece las condiciones bajo las cuales el valor de Shapley de un juego TU pertenece al *core*. Este análisis vincula dos conceptos fundamentales en la teoría de juegos cooperativos: la estabilidad de las asignaciones, garantizada por el *core*, y la ecuanimidad en el reparto, representada por el valor de Shapley.

Definición 13. Un juego $v \in G(N)$ se dice **convexo** si $\forall i \in N$ y para todo par de subconjuntos $S \subseteq T \subseteq N \setminus \{i\}$ se verifica: $v(S \cup \{i\}) - v(S) \leq v(T \cup \{i\}) - v(T)$.

Teorema 14. Sea $v \in G(N)$:

$$v \text{ convexo} \implies \Phi(v) \in C(v) \quad (1.7)$$

Además, si v es un juego **convexo**, el núcleo $C(v)$ se podrá calcular como la envoltura convexa de los vectores de contribuciones marginales dado por 1.5.

1.2.4. Valor de Owen de un juego TU

El valor de Owen es una extensión del valor de Shapley que se aplica en contextos donde los jugadores están organizados en uniones con estructuras internas específicas, de modo que el sistema de uniones constituye una partición del conjunto de jugadores N . Este valor proporciona una manera justa de distribuir los beneficios de la cooperación, considerando no solo las contribuciones individuales de los jugadores, sino también la influencia de las uniones en el resultado global.

Al igual que se hizo en el apartado anterior, se definirá formalmente el valor de Owen, se presentará su fórmula de asignación y se incluirá un código de simulación que permite aproximar este valor cuando el número de jugadores o uniones es elevado, el cual servirá para el desarrollo de la sección 2.4.

De acuerdo con la notación utilizada hasta ahora, se considera (N, v) un juego cooperativo con utilidad transferible, donde N es el conjunto de jugadores y v es la función característica. Además, en este apartado, es necesario introducir una partición P del conjunto N , que represente la estructura de coaliciones.

Definición 15. Un **juego TU con uniones a priori** es una terna (N, v, P) , donde $(N, v) \in G(N)$ $P = \{P_1, \dots, P_t\}$ es una partición de N que proporciona el sistema de uniones asociado.

Denotaremos por $G(N)^U$ a los juegos TU de N jugadores con uniones a priori. En este nuevo contexto podemos definir la regla de reparto según el valor de Owen, que se calcula adaptando la fórmula del valor de Shapley, ponderando las contribuciones marginales de los jugadores teniendo en cuenta el sistema de uniones.

El valor de Owen para un juego con uniones a priori puede expresarse utilizando permutaciones. No obstante, a diferencia del valor de Shapley, este concepto requiere introducir la idea de **permutaciones compatibles con la partición P** .

Definición 16. Una permutación de los elementos del conjunto N , $\pi \in \Pi(N)$, se dice **compatible con la partición P** si, en dicha ordenación, todos los jugadores que pertenecen a una misma unión de la partición aparecen de forma consecutiva.

Formalmente, si denotamos por $P(i)$ la coalición a la que pertenece el jugador i una permutación π es compatible con P si y solo si para cualquier $i, j, k \in N$ se cumple que: si $P(i) = P(j)$ con $\pi(i) < \pi(k) < \pi(j)$, entonces necesariamente $P(i) = P(j) = P(k)$.

Estamos entonces en condiciones de definir el valor de Owen:

Definición 17. El valor de Owen de un juego con uniones a priori $(N, v, P) \in G(N)^U$ se define como:

$$\gamma_i(N, P) = \frac{1}{|\Pi(N)^P|} \sum_{\pi \in \Pi(N)^P} (v(A^\pi(i) \cup \{i\}) - v(A^\pi(i))) \quad (1.8)$$

para todo $i \in N$, donde $\Pi(N)^P$ representa el conjunto de ordenaciones compatibles con la partición P del conjunto N .

Como conclusión, mientras que el valor de Shapley asigna las ganancias de forma ecuánime sin considerar restricciones en las coaliciones, el valor de Owen resulta más adecuado en contextos

donde las alianzas o estructuras predefinidas juegan un papel determinante. Esto lo convierte en una herramienta indispensable en problemas de reparto en cadenas de suministro con acuerdos preexistentes o en mercados con empresas que colaboran estratégicamente.

Considerando de nuevo el Ejemplo 12 de 3 jugadores, se propone el siguiente código para el cálculo del valor de Owen de manera exacta.

Para comenzar, es necesario definir un juego con uniones a priori, es decir, dado el juego TU con función característica v mostrado en la Tabla 1.1, determinando una partición $P = \{P_1, P_2\}$ tal que $P_1 = \{1, 3\}$ y $P_2 = \{2\}$, se tendrá el nuevo juego $G(N)^U$ del cual se desea conocer el valor de Owen.

Las posibles ordenaciones $\pi \in \Pi(N)^P$ serán ahora aquellas donde los elementos $\{1, 3\}$ sean consecutivos, es decir: $(1, 3, 2)$, $(2, 1, 3)$, $(3, 1, 2)$ y $(2, 3, 1)$. Considerando entonces las contribuciones marginales 1.2 se obtienen las siguientes asignaciones:

$$\gamma_1(N, P) = \frac{1}{4}(0 + 1 + 1 + 1) = \frac{3}{4}$$

$$\gamma_2(N, P) = \frac{1}{4}(0 + 0 + 0 + 0) = 0$$

$$\gamma_3(N, P) = \frac{1}{4}(1 + 0 + 0 + 0) = \frac{1}{4}$$

A continuación se proponen dos bloques de código, análogos a los planteados para el cálculo del valor de Shapley, para el cálculo exacto del valor de Owen y una posible aproximación por muestreo que se usará cuando el juego contenga un número elevado de jugadores.

```

1 # Definir la partición
2 P1 <- c(1, 3) # Primera unión a priori
3 P2 <- c(2)    # Segunda unión a priori
4 partition <- list(P1, P2)
5
6 library(gtools)
7 compatible_permutations <- list()
8
9 # Generar todas las permutaciones de cada unión a priori
10 perms_P1 <- permutations(length(P1), length(P1), P1)
11 perms_P2 <- permutations(length(P2), length(P2), P2)
12
13 # Combinar las permutaciones de las uniones a priori
14 for (perm1 in 1:nrow(perms_P1)) {
15   for (perm2 in 1:nrow(perms_P2)) {
16     compatible_permutations <- append(
17       compatible_permutations,
18       list(c(perms_P1[perm1,], perms_P2[perm2,]))
19     )
20     compatible_permutations <- append(
21       compatible_permutations,

```



```

22     list(c(perms_P2[perm2,], perms_P1[perm1,]))
23   )
24 }
25 }
26
27 # Calcular el valor de Owen
28 owen_values <- rep(0, length(players))
29 names(owen_values) <- players
30
31 for (player in players) {
32   marginal_contributions <- c()
33
34   for (perm in compatible_permutations) {
35
36     # Determinar los jugadores previos a "player" en la permutación
37     player_index <- which(perm == player)
38     predecessors <- perm[1:(player_index - 1)]
39
40     # Calcular la contribución marginal
41     with_player <- union(predecessors, player)
42     without_player <- predecessors
43     marginal_contributions <- c(marginal_contributions,
44                               v(with_player) - v(without_player))
45   }
46
47   # Promediar las contribuciones marginales
48   owen_values[player] <- mean(marginal_contributions)
49 }
50
51 # Resultado
52 owen_values

```

```

1   1   2   3
2 0.75 0.00 0.25

```

```

1 # Comprobación eficiencia
2 sum(owen_values)

```

```

1 [1] 1

```

Si se desea **aproximar** el valor de Owen mediante permutaciones, como se desarrolla en [Saavedra-Nieves et al. \(2018\)](#), bastaría con ejecutar el siguiente código:

```

1 # Número de iteraciones para la aproximación
2 n_iterations <- 100000
3 n <- length(players)
4 owen_approx <- numeric(n) # Se almacenarán las contribuciones
5 names(owen_approx) <- players
6
7 # Definir la partición
8 P1 <- c(1, 3) # Primera unión a priori
9 P2 <- c(2)   # Segunda unión a priori
10 partition <- list(P1, P2)
11

```

```

12 # Generar todas las permutaciones de cada subconjunto
13 library(gtools)
14 perms_P1 <- permutations(length(P1), length(P1), P1)
15 perms_P2 <- permutations(length(P2), length(P2), P2)
16
17 # Iterar a través de un número grande de simulaciones
18 for (iter in 1:n_iterations) {
19
20   # Generar una permutación compatible con la partición
21   perm_P1 <- perms_P1[sample(1:nrow(perms_P1), 1), ] # Aleatoria de P1
22   perm_P2 <- perms_P2[sample(1:nrow(perms_P2), 1), ] # Aleatoria de P2
23   if (runif(1) < 0.5) {
24     permutation <- c(perm_P1, perm_P2) # Combinar P1 seguido de P2
25   } else {
26     permutation <- c(perm_P2, perm_P1) # Combinar P2 seguido de P1
27   }
28
29   cost_prev <- 0 # Valor de la coalición previa
30
31   # Calcular contribuciones marginales para la permutación
32   for (j in 1:n) {
33     current_player <- permutation[j]
34     coalition <- permutation[1:j] # Coalición actual en la permutación
35
36     # Evaluar la función característica para la coalición actual
37     v <- function(S) {
38       if (setequal(S, c(1))) return(0)
39       if (setequal(S, c(2))) return(0)
40       if (setequal(S, c(3))) return(0)
41       if (setequal(S, c(1, 2))) return(1)
42       if (setequal(S, c(1, 3))) return(1)
43       if (setequal(S, c(2, 3))) return(0)
44       if (setequal(S, c(1, 2, 3))) return(1)
45       return(0)
46     }
47
48     cost_current <- v(coalition) # Valor de la coalición actual
49
50     # Contribución marginal del jugador actual
51     marginal_contribution <- cost_current - cost_prev
52     owen_approx[current_player] <- (owen_approx[current_player]
53                                   + marginal_contribution)
54
55     # Actualizar el costo previo
56     cost_prev <- cost_current
57   }
58 }
59
60 # Dividir por el número de iteraciones para obtener el promedio
61 owen_approx <- owen_approx / n_iterations
62 owen_approx

```

```

1      1      2      3
2 0.75072 0.00000 0.24928

```

En conclusión, los resultados obtenidos para el juego planteado de 3 jugadores muestran la coherencia y utilidad de los métodos exactos y aproximados en la asignación de valores tanto de Shapley como de Owen. Para el valor de Shapley, el cálculo exacto $(\frac{2}{3}, \frac{1}{6}, \frac{1}{6})$ y la aproximación $(0.66266, 0.16710, 0.17024)$ son bastante cercanos, evidenciando la precisión del método de aproximación con un número elevado de iteraciones. Por otro lado, para el valor de Owen con la partición específica $\{\{1, 3\}, \{2\}\}$, los resultados exactos $(\frac{3}{4}, 0, \frac{1}{4})$ y aproximados $(0.75152, 0.00000, 0.24848)$ también reflejan una alta concordancia, validando la adaptación del algoritmo a las permutaciones compatibles con la partición. Estas similitudes subrayan la estabilidad de las herramientas aproximativas y su aplicabilidad en problemas donde el cálculo exacto puede ser computacionalmente costoso. Además, los valores obtenidos para Owen destacan la influencia de las restricciones estructurales de las coaliciones a priori en la distribución de las ganancias del juego, modificando significativamente las asignaciones respecto al valor de Shapley.

Capítulo 2

Modelos deterministas EOQ

La gestión óptima de inventarios constituye una de las áreas centrales en la investigación operativa, siendo los modelos de cantidad económica de pedido (EOQ, por sus siglas en inglés) una herramienta fundamental para su análisis en entornos deterministas. Estos modelos, basados en la suposición de una demanda lineal en el tiempo y conocida, permiten determinar la cantidad y el momento óptimo de reposición, minimizando los costes totales asociados al mantenimiento de inventarios y a la realización de pedidos.

En este capítulo, se presentan distintos modelos EOQ deterministas que abordan situaciones clave en la gestión de inventarios. Para comenzar, el modelo básico representa el caso más sencillo y general. A continuación, se explorarán variantes más complejas que consideran múltiples artículos, múltiples agentes, y finalmente la combinación de ambos. Para cada caso, se analizan escenarios con y sin costes eximibles, proporcionando un enfoque completo y detallado.

El desarrollo y análisis de estos modelos están fundamentados en las ideas presentadas en [Fiestras-Janeiro et al. \(2024\)](#). Este trabajo sirve como referencia principal para la formulación matemática y la interpretación de los resultados. Las metodologías y conceptos propuestos en dicho artículo han sido adaptados en este capítulo para abordar problemas más específicos, ampliando su aplicabilidad en diferentes contextos logísticos.

2.1. Modelo básico determinista

En el modelo básico EOQ una única empresa busca satisfacer la demanda de un producto que comercializa a sus clientes. Para ello, adquiere el producto de un proveedor y lo almacena en un inventario de capacidad ilimitada. La empresa tiene la libertad de realizar pedidos al proveedor en cualquier momento, y el proveedor puede decidir eximir, o no, los costes de pedido si el tamaño

del pedido supera un umbral predefinido. Este modelo asume que no se permiten faltantes y que la demanda es determinista y lineal.

El tiempo de entrega (**lead time**), definido como el intervalo entre la realización de un pedido y su recepción, se considera determinista y constante. Para simplificar el análisis, se asumirá sin pérdida de generalidad que este tiempo es igual a cero. Las decisiones de la empresa en este modelo giran en torno a determinar el tamaño óptimo del pedido que minimice el coste medio por unidad de tiempo.

El problema se caracteriza por los siguientes parámetros clave:

- $d > 0$: **demanda** del producto por unidad de tiempo, conocida y constante.
- $h > 0$: **coste de almacenamiento** por unidad de producto y por unidad de tiempo.
- $a > 0$: **coste de realizar un pedido**, independiente del tamaño del mismo.
- $A > 0$: **tamaño mínimo** del pedido a partir del cual el proveedor **exime** a la empresa del coste de pedido.

La **variable de decisión** en este modelo es el tamaño del pedido, Q , que representa la cantidad de producto que la empresa solicita al proveedor cuando el nivel de inventario alcanza cero. Este tamaño de pedido debe elegirse de manera que se minimice el coste medio total por unidad de tiempo.

En este modelo, se supone que el coste de adquisición por unidad de producto es constante e independiente del tamaño del pedido. Dado que toda la demanda debe ser satisfecha, el coste de adquisición no afecta la optimización del problema.

Es importante tener en cuenta que el **coste medio por ciclo** incluye 2 tipos de costes:

1. Costes de **almacenamiento**: Proporcionales al inventario promedio acumulado.
2. Costes de **pedido**: Determinados por el número de pedidos por unidad de tiempo y, en algunos casos, reducidos si el pedido excede el tamaño A .

Dado que el ciclo de inventario se repite de manera uniforme en el tiempo, el coste medio por unidad de tiempo puede expresarse en función del tamaño del pedido Q . En las secciones siguientes, se calculará el tamaño de pedido que minimiza este coste medio y se analizará el impacto de la exención de costes cuando el pedido supera el umbral A .

2.1.1. Modelo con costes eximibles

Como primera situación, se plantea el caso en que el proveedor puede eximir a la empresa de los costes de pedido si el tamaño del pedido supera un umbral predeterminado, A . Este supuesto resulta considerablemente significativo en la función de coste.

La **función de coste medio** para un tamaño de pedido Q se define como:

$$C(Q) = \begin{cases} \frac{hQ}{2} + \frac{ad}{Q} & \text{si } Q < A, \\ \frac{hQ}{2} & \text{si } Q \geq A. \end{cases} \quad (2.1)$$

Como se puede observar, el coste medio por unidad de tiempo $C(Q)$ incluye dos componentes principales:

1. Coste de almacenamiento: $\frac{hQ}{2}$, que resulta proporcional al inventario promedio $\frac{Q}{2}$.
2. Coste de pedido: $\frac{ad}{Q}$. Este término es proporcional al número de pedidos realizados por unidad de tiempo $\frac{d}{Q}$.

Con el objetivo de minimizar el coste medio, puede considerarse cada uno de los casos donde está definida $C(Q)$ y calcular los mínimos locales.

Por un lado, cuando la cantidad del pedido no es suficiente como para que el proveedor exima de gastos a la empresa: $Q < A$, la función resulta:

$$C(Q) = \frac{hQ}{2} + \frac{ad}{Q}$$

Calculando la primera de las derivadas es posible calcular los puntos críticos de dicha función:

$$\frac{dC(Q)}{dQ} = \frac{h}{2} - \frac{ad}{Q^2} = 0$$

$$Q^2 = \frac{2ad}{h}$$

$$Q = \sqrt{\frac{2ad}{h}}$$

Como $\frac{dC(Q^*)}{dQ^2} > 0$, el tamaño de pedido óptimo para el caso $Q^* < A$ resultará:

$$Q^* = \sqrt{\frac{2ad}{h}}$$

Sea ahora $Q \geq A$. Como ahora el tamaño del pedido es suficiente como para despreciar el coste del pedido, la función de coste resulta:

$$C(Q) = \frac{hQ}{2}$$

Como $C(Q)$ es una función creciente linealmente en Q , el tamaño óptimo del pedido en este caso será $Q^* = A$.

En resumen, si el tamaño de pedido óptimo según el modelo EOQ clásico es menor que una cantidad prefijada A , la empresa debe seguir la política de pedido tradicional, asumiendo los costes de pedido. En otro caso, el proveedor exime los costes de pedido, y el **tamaño óptimo del pedido** será igual a A . Formalmente:

$$Q^* = \begin{cases} \sqrt{\frac{2ad}{h}} & \text{si } \sqrt{\frac{2ad}{h}} < A, \\ A & \text{si } \sqrt{\frac{2ad}{h}} \geq A. \end{cases} \quad (2.2)$$

Como ya se mencionó, la cantidad $\frac{d}{Q}$ corresponde al **número de pedidos** que deben realizarse de una cantidad Q de producto con demanda d . Por lo tanto, la **longitud del ciclo** o intervalo de tiempo que transcurrirá entre dos pedidos vendrá dada por la fracción $\frac{Q}{d}$.

Una vez calculado el tamaño óptimo de pedido que hace mínimo los costes de inventario Q^* , se definirá la **longitud del ciclo óptima** como:

$$T^* = \begin{cases} \sqrt{\frac{2a}{dh}} & \text{si } \sqrt{\frac{2ad}{h}} < A, \\ \frac{A}{d} & \text{si } \sqrt{\frac{2ad}{h}} \geq A. \end{cases} \quad (2.3)$$

Para finalizar este apartado, evaluando la función de coste 2.1 en el tamaño 2.2, se define el **coste mínimo** del pedido como:

$$C(Q^*) = \begin{cases} \sqrt{2adh} & \text{si } \sqrt{\frac{2ad}{h}} < A, \\ \frac{hA}{2} & \text{si } \sqrt{\frac{2ad}{h}} \geq A. \end{cases} \quad (2.4)$$

2.1.2. Modelo sin costes eximibles

En este caso, se considera que el proveedor no ofrece la posibilidad de eliminar los costes de pedido, sin importar el tamaño del pedido realizado. Este escenario corresponde al modelo

clásico de EOQ, en el que los costes de pedido siempre se mantienen presentes. La **función de coste medio** para un tamaño de pedido Q se expresa como:

$$C(Q) = \frac{hQ}{2} + \frac{ad}{Q} \quad (2.5)$$

Como se desarrolló anteriormente, calculando la derivada de la función $C(Q)$ respecto a Q se puede obtener el **tamaño óptimo** del pedido:

$$Q^* = \sqrt{\frac{2ad}{h}} \quad (2.6)$$

y logitud óptima: $T^* = \sqrt{\frac{2a}{dh}}$.

Por último, el **coste mínimo** para este modelo sin gastos eximibles será $C(Q^*) = \sqrt{(2adh)}$.

2.2. Modelo determinista con múltiples artículos

Una vez analizado el modelo básico para un único artículo, se planteará ahora el enfoque para considerar el caso de múltiples artículos. Esta extensión es particularmente relevante en situaciones donde una empresa debe gestionar simultáneamente diferentes productos, optimizando el inventario de forma conjunta para minimizar los costes totales. Al igual que en el apartado anterior, se diferenciará el caso en el que el proveedor puede eximir a la empresa de gastos y el caso en el que no existe tal opción.

2.2.1. Modelo con costes eximibles

En este escenario, el proveedor puede eximir a la empresa de los costes de pedido si el coste total del pedido, considerando todos los artículos, supera un umbral predefinido A .

Se considera un conjunto $N = \{1, \dots, n\}$ de artículos gestionados por una única empresa. En este nuevo modelo, los parámetros a tener en cuenta para cada artículos $i \in N$ serán:

- $d_i > 0$: **demanda** del producto i por unidad de tiempo.
- $h_i > 0$: **coste de almacenamiento** por unidad del producto i y por unidad de tiempo.
- $c_i > 0$: **coste unitario** del artículo i .
- $a > 0$: **coste de realizar un pedido**, compartido por todos los artículos.

- $A > 0$: **tamaño mínimo** del total pedido a partir del cual el proveedor **exige** a la empresa del coste de pedido.

La decisión clave será determinar el tamaño del pedido Q_i para cada artículo i de forma que se minimice el coste total.

Si Q_i denota la cantidad de pedido de un artículo i adquirida con el fin de cubrir la demanda d_i del agente en cuestión, el cociente $\frac{Q_i}{d_i}$ será el período entre dos pedidos consecutivos y $\frac{d_i}{Q_i}$ el número de pedidos por unidad de tiempo.

Como se comentó en el modelo básico, el coste medio por ciclo está compuesto por el coste de almacenamiento $\frac{h_i Q_i}{2}$ y el coste de realización del pedido $\frac{ad_i}{Q_i}$.

Para el desarrollo de este trabajo, se ha escogido el caso en el que la empresa interesada, opta por realizar pedidos conjuntos para todos los artículos seleccionados de un conjunto $S \subseteq N$, en lugar de permitir que cada artículo tenga pedidos independientes. Este supuesto puede ayudar a la reducción de costes de administración de pedidos o restricciones logísticas.

En este escenario, la condición $\frac{Q_i}{d_i} = \frac{Q_j}{d_j}$ para todos los $i, j \in N$, implica que los artículos comparten el mismo ciclo de reabastecimiento. Es decir, cada artículo se repone al mismo tiempo para sincronizar los pedidos.

Esta restricción afecta la función de coste promedio por ciclo, que ahora debe considerar el conjunto completo de artículos que se gestionan conjuntamente. En lugar de minimizar costes para cada artículo por separado, el problema optimiza el coste total de todos los artículos simultáneamente, ajustando el tamaño de los pedidos conjuntos

La función de **coste promedio por ciclo** para una colección de artículos $S \subseteq N$ puede expresarse como:

$$C(\{Q_j\}_{j \in S}) = \begin{cases} \frac{ad_i}{Q_i} + \sum_{j \in S} \frac{h_j Q_j}{2} & \text{si } \sum_{j \in S} c_j Q_j < A, \\ \sum_{j \in S} \frac{h_j Q_j}{2} & \text{si } \sum_{j \in S} c_j Q_j \geq A. \end{cases} \quad (2.7)$$

En esta expresión, el término $\sum_{j \in S} \frac{h_j Q_j}{2}$ corresponde al **coste de almacenamiento**, que se calcula como la suma de los costes de mantener inventario para cada artículo en el conjunto S . El término $\frac{ad_i}{Q_i}$ corresponde al **coste de pedido**. Nótese que como $\frac{d_k}{Q_k} = \frac{d_j}{Q_j}$ para cualesquiera $j, k \in N$, el cociente $\frac{d_i}{Q_i}$ no dependerá de i .

Como el objetivo es realizar un único pedido conjunto para todos los artículos en el conjunto S , entonces el **coste de pedido total** será $\frac{ad_i}{Q_i}$, para cualquier $i \in N$.

Otra expresión alternativa de la función 2.7 resulta:

$$C(T) = \begin{cases} \frac{a}{T} + \frac{T}{2} \sum_{j \in S} h_j d_j & \text{si } \sum_{j \in S} c_j Q_j < A, \\ \frac{T}{2} \sum_{j \in S} h_j d_j & \text{si } \sum_{j \in S} c_j Q_j \geq A. \end{cases} \quad (2.8)$$

donde $T = \frac{Q_i}{d_i}$ denota el ciclo o período de reposición.

Derivando respecto a T la expresión 2.8 e igualando a cero, se obtiene el **ciclo óptimo** del pedido.

Por un lado, cuando $\sum_{j \in S} c_j Q_j < A$ se tiene:

$$\frac{dC(T)}{dT} = -\frac{a}{T^2} + \frac{1}{2} \sum_{j \in S} h_j d_j = 0$$

con lo que el mínimo es:

$$T^* = \sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}}$$

ya que $\frac{dC(T^*)}{dT^2} > 0$.

Por otro lado, si $\sum_{j \in S} c_j Q_j \geq A$ la función $C(T)$ es creciente en T , por que T^* será el mínimo valor que verifique $\sum_{j \in S} c_j Q_j^* \geq A$:

$$\sum_{j \in S} c_j Q_j^* \geq A \iff \frac{Q_i^*}{d_i} \sum_{j \in S} c_j d_j \geq A$$

$$T^* \geq \frac{A}{\sum_{j \in S} c_j d_j}$$

Así entonces, el **ciclo óptimo** de reposición del pedido resulta:

$$T^* = \begin{cases} \sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} & \text{si } \sum_{j \in S} c_j Q_j < A, \\ \frac{A}{\sum_{j \in S} c_j d_j} & \text{si } \sum_{j \in S} c_j Q_j \geq A. \end{cases} \quad (2.9)$$

A partir del ciclo óptimo T^* , se llega al **tamaño óptimo del pedido** $Q_i^* = d_i T^*$:

$$Q_i^* = \begin{cases} d_i \sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} & \text{si } 2\sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} < \frac{A}{\sum_{j \in S} c_j d_j}, \\ \frac{d_i A}{\sum_{j \in S} c_j d_j} & \text{si } 2\sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} \geq \frac{A}{\sum_{j \in S} c_j d_j}. \end{cases} \quad (2.10)$$

Con el tamaño óptimo 2.10 y la función de coste 2.7, se llega al coste mínimo del pedido:

$$C(Q_i^*) = \begin{cases} \sqrt{2a \sum_{j \in S} h_j d_j} & \text{si } 2\sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} < \frac{A}{\sum_{j \in S} c_j d_j}, \\ \frac{A}{2} \frac{\sum_{j \in S} h_j d_j}{\sum_{j \in S} c_j d_j} & \text{si } 2\sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} \geq \frac{A}{\sum_{j \in S} c_j d_j}. \end{cases} \quad (2.11)$$

2.2.2. Modelo sin costes eximibles

En este escenario, se asume que el proveedor no ofrece la posibilidad de eliminar los costes de pedido, independientemente del coste total del pedido. Esto implica que la empresa debe optimizar los costes manteniendo el coste de realización de pedidos presente en el análisis.

Como en el caso anterior, se asume que todos los artículos comparten el mismo ciclo de reabastecimiento T , lo que implica que:

$$T = \frac{Q_i}{d_i} \quad \forall i \in N$$

De nuevo, el coste total por unidad de tiempo puede descomponerse en dos componentes principales: el coste de almacenamiento $\frac{h_i Q_i}{2}$, con $\frac{Q_i}{2}$ el inventario promedio del artículo i , y el coste del pedido $\frac{ad_i}{Q_i} = \frac{a}{T}$, bajo la condición de realizar un único pedido conjunto para todos los artículos por ciclo.

Sea $S \subseteq N$, la función de coste del pedido de esos artículos será

$$C(\{Q_j\}_{j \in S}) = \frac{ad_i}{Q_i} + \sum_{j \in S} \frac{h_j Q_j}{2} \quad (2.12)$$

Si se desea expresar el coste total en función del ciclo de reposición, de manera análoga a 2.8, se tendrá

$$C(T) = \frac{a}{T} + \frac{T}{2} \sum_{j \in S} h_j d_j \quad (2.13)$$

Derivando la expresión 2.13 respecto a T , es posible encontrar el período de reposición óptimo:

$$T^* = \sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}}$$

Una vez obtenido el **ciclo óptimo** $T^* = \sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}}$, puede calcularse el **tamaño óptimo** del pedido como:

$$Q_i^* = T^* d_i = d_i \sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}}$$

Por último, el **coste mínimo** del pedido vendrá dado por:

$$C(Q_i^*) = \sqrt{2a \sum_{j \in S} h_j d_j}$$

Es fácil ver cómo el coste mínimo total en este caso, refleja un equilibrio entre los costes de pedido y almacenamiento, sin la influencia de descuentos por coste de pedido, como ocurría en 2.11.

Desde la perspectiva de optimización conjunta, este modelo también puede ser analizado bajo el marco de la **teoría de juegos cooperativos**. Cada artículo del conjunto S puede considerarse un jugador que contribuye a los costes totales de almacenamiento y pedido. La cooperación permite alcanzar un coste mínimo global, pero surge la cuestión de cómo repartir estos costes de manera justa entre los artículos. Para abordar este problema, herramientas como el valor de Shapley pueden proporcionar un reparto ecuánime basado en las contribuciones individuales al ahorro total.

En el capítulo introductorio, se definió un juego con utilidad transferible (TU) 1 como una estructura matemática (N, v) donde N es un conjunto de jugadores y v la función característica. En el contexto de este modelo de inventario, N corresponde al conjunto de artículos, y $v(S)$ representa el coste mínimo al gestionar los artículos de S de manera conjunta.

$$v(S) = C(Q_i^*)$$

Además, en el mismo capítulo se introdujo el **valor de Shapley** 1.4 como una herramienta ampliamente utilizada en la teoría de juegos cooperativos para repartir de manera justa los beneficios o costes derivados de la cooperación entre los jugadores. Este valor asigna a cada artículo $i \in N$ un coste que captura su impacto individual en el ahorro global. Este reparto puede ser de utilidad para el gestor de los inventarios en este contexto por motivos contables o para posibles altas/bajas de artículos con los que comerciar.

Por lo tanto, al aplicar el valor de Shapley a este problema de inventario, cada artículo recibe una asignación que refleja su impacto individual en la reducción de costes totales, considerando todas las posibles configuraciones de cooperación entre artículos. Esto permite a la empresa obtener no solo una solución óptima global, sino también un reparto equitativo de los costes que facilita decisiones informadas y equitativas para la gestión del inventario.

Para contrastar y visualizar claramente las diferencias entre las situaciones con y sin costes eximibles, a continuación se presenta un ejemplo práctico. Implementando las fórmulas y los conceptos desarrollados anteriormente, se proporcionará un posible código que permitirá analizar ambos escenarios de manera cuantitativa y comprensible. Este ejemplo ilustrará paso a paso cómo calcular el tamaño óptimo del pedido, el coste mínimo del ciclo para cada caso y el reparto de costes proporcionado por el valor de Shapley.

2.2.3. Ejemplo ilustrativo

El presente ejemplo está basado en el caso práctico recogido en el artículo [Fiestras-Janeiro et al. \(2024\)](#). En dicho trabajo, se aborda un modelo de inventario en el que los costes de pedido pueden ser eximidos bajo ciertas condiciones. Sin embargo, en este trabajo se compararán los resultados cuando no se considera la posibilidad de eximir dichos costes, proporcionando una perspectiva complementaria al modelo original.

El caso práctico se basa en datos reales de una empresa dedicada a la distribución de materiales eléctricos, que busca optimizar la gestión de su inventario. Esta empresa tiene un grupo de clientes para los cuales necesita garantizar la disponibilidad constante de los materiales que solicitan. La empresa gestiona un total de 100 productos, cada uno con características particulares como demanda mensual, coste de almacenamiento y coste unitario. El coste de pedido es compartido entre todos los productos.

En este contexto, se emplea el valor de Shapley como una regla para repartir ecuánimamente los costes asociados a la gestión del inventario. Esta metodología, basada en la teoría de juegos, es una herramienta ampliamente utilizada para problemas de reparto, ya que garantiza una asignación justa y proporcional según la contribución marginal de cada producto al ahorro de costes cuando se gestionan de manera conjunta.


A continuación, se presenta en la Tabla 2.1 información clave sobre los 100 productos del inventario de la empresa. Cada fila corresponde a un producto identificado por un número único (ítem), y las columnas contienen las correspondientes demandas mensuales d_i , costes unitarios de almacenamiento h_i , y coste unitario de cada producto c_i .

Ítem	d	h	c
1	419	0.45	4.03
2	467	0.46	4.13
3	183	0.32	94.01
4	18	0.23	68.20
5	199	0.12	64.41
6	430	0.47	1.18
7	310	0.19	5.53
8	122	0.08	58.53
9	303	0.48	8.84
10	233	0.37	58.61
11	248	0.11	74.10
12	18	0.30	62.65
13	378	0.48	1.06
14	157	0.31	60.81
15	94	0.23	83.72
16	95	0.34	52.38
17	260	0.19	85.04
18	201	0.19	67.59
19	95	0.15	70.45
20	445	0.22	1.65
21	352	0.49	2.48
22	141	0.12	91.05
23	481	0.09	7.93
24	359	0.11	7.62
25	332	0.36	9.75
26	30	0.33	95.94
27	11	0.45	64.13
28	430	0.35	5.20
29	141	0.38	98.06
30	258	0.28	86.42
31	215	0.35	84.32
32	424	0.42	1.67
33	82	0.40	52.64
34	85	0.49	69.76
35	47	0.25	73.89

Ítem	d	h	c
36	363	0.19	6.84
37	377	0.23	7.83
38	336	0.05	2.23
39	147	0.13	78.01
40	36	0.43	84.91
41	494	0.15	4.57
42	113	0.16	95.78
43	267	0.08	80.92
44	397	0.16	3.02
45	429	0.38	1.52
46	334	0.43	4.56
47	27	0.27	71.42
48	415	0.22	1.58
49	448	0.16	3.03
50	313	0.10	1.49
51	139	0.23	77.10
52	313	0.31	7.03
53	415	0.15	3.68
54	228	0.25	52.92
55	170	0.15	63.04
56	221	0.28	69.86
57	294	0.21	59.89
58	481	0.34	1.91
59	73	0.22	91.60
60	76	0.21	57.64
61	80	0.29	90.17
62	67	0.38	77.34
63	145	0.15	83.12
64	214	0.24	58.58
65	357	0.17	1.65
66	351	0.33	8.92
67	157	0.13	81.65
68	298	0.44	65.59
69	281	0.39	86.23

Ítem	d	h	c
70	467	0.35	7.79
71	45	0.33	69.95
72	435	0.22	8.35
73	94	0.29	98.47
74	320	0.44	9.84
75	227	0.31	98.37
76	20	0.43	86.34
77	120	0.19	62.86
78	98	0.37	61.09
79	266	0.17	79.65
80	239	0.32	63.38
81	460	0.27	1.93
82	90	0.17	76.55
83	346	0.30	1.89
84	43	0.46	89.26
85	163	0.46	58.40
86	244	0.17	70.22
87	213	0.19	73.58
88	378	0.49	8.19
89	310	0.33	8.06
90	38	0.47	93.41
91	8	0.26	96.29
92	17	0.23	94.10
93	73	0.35	83.71
94	387	0.12	1.08
95	247	0.31	97.51
96	413	0.16	8.01
97	312	0.48	7.56
98	282	0.32	75.82
99	93	0.28	78.83
100	420	0.23	6.67

Tabla 2.1: Datos del inventario divididos en tablas paralelas.

Para calcular el **tamaño óptimo 2.10** del pedido, diferenciando el caso de poder eximir de gastos a la empresa o no, puede implementarse en  el siguiente código.

```

1 # Lectura de datos
2 datos = read.csv2("datos_tfm.csv", header=TRUE, dec=".")
3 d=datos$d
4 c=datos$c
5 h=datos$h
6 bien = datos$X...item
7 a=2000
8 A=200000

```

```

1 # Caso CON costes eximibles
2 #####
3
4 Q_gorro <- numeric(length(d))
5
6 for (i in 1:length(d)) {
7   if (2 * sqrt((2*a) / (sum(h*d))) < A/(sum(c*d))) {
8     Q_gorro[i] <- sqrt( (2*a*(d[i]^2)) / (sum(h*d)) )
9   }
10   Q_gorro[i] <- (A*d[i]) / (sum(c*d))
11 }

```

```

1 # Caso SIN costes eximibles
2 #####
3
4 Q_gorro_sin_c <- numeric(length(d))
5
6 for (i in 1:length(d)) {
7   Q_gorro_sin_c[i] <- sqrt( 2*a*(d[i]^2) / (sum(h*d)) )
8 }

```

En la Figura 2.1 se muestran los valores Q_i^* ordenados de manera creciente. A pesar de que los tamaños óptimos Q_i^* del modelo con costes eximibles y del modelo sin costes eximibles no coinciden para ninguno de los artículos i , la ordenación de los valores es la misma en ambos casos. La desigualdad de los valores Q_i^* en cada uno de los modelos se debe a que, para este modelo, siempre se verifica la condición $2\sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} \geq \frac{A}{\sum_{j \in S} c_j d_j}$.

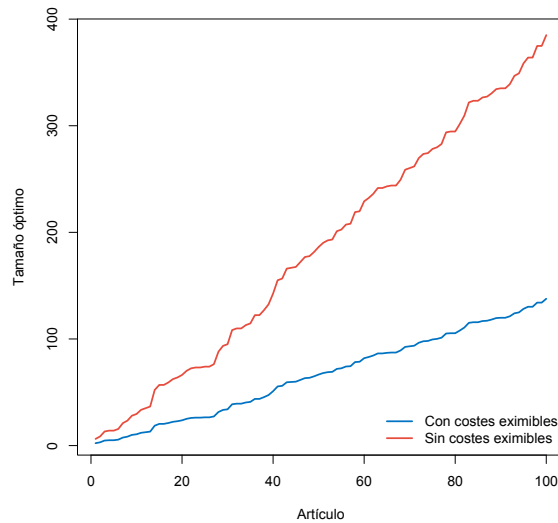


Figura 2.1: Representación de los tamaños óptimos Q_i^* .

Para calcular el **período óptimo** de reposición del pedido bastará considerar:

```

1 # Caso CON costes eximibles
2 #####
3
4 { if (sum(c*Q_gorro) < A) {
5     ciclo_opt <- sqrt(2*a/sum(h*d))
6 }
7     ciclo_opt <- A / sum(c*d)
8 }
9 ciclo_opt

```

```
1 [1] 0.2787984
```

```

1 # Caso SIN costes eximibles
2 #####
3
4 ciclo_opt_sin_c <- A / sum(c*d)
5 ciclo_opt_sin_c

```

```
1 [1] 0.2787984
```

Por lo tanto, para el ejemplo que se está analizando, lo más favorable será reponer el pedido cada 0.28 meses, es decir, cada 8 días aproximadamente, independientemente de considerar o no gastos eximibles. En este caso particular no está afectando considerar o no la posibilidad de exención de coste porque $\sum(c_j * Q_j) = A$. Considerando $\frac{1}{T^*}$, se tendrá que el número óptimo de pedidos a realizar por mes estará entre 3 y 4.

Una vez calculados los tamaños y ciclos óptimos del pedido, es posible conocer el **coste total** 2.7 que tendrá su realización:

```

1 # Caso CON costes eximibles
2 #####
3
4 { if (2 * sqrt((2*a) / (sum(h*d))) < A/(sum(c*d))) {
5   coste <- sqrt(2*a*sum(h*d))
6 }
7   coste <- A/2 * sum(h*d)/sum(c*d)
8 }
9 coste

```

```
1 [1] 918.1305
```

```

1 # Caso SIN costes eximibles
2 #####
3
4 coste_sin_c <- sqrt(2*a*sum(h*d))
5 coste_sin_c

```

```
1 [1] 5132.773
```

Si se considera la posibilidad de eximir a la empresa de gastos, se obtiene un coste total $C(Q_i^*) = 918.13$, mientras que si no hay dicha posibilidad resultará $C(Q_i^*) = 5132.77$. Como era de esperar, para el ejemplo considerado, beneficiará contar la posibilidad de eximir a la empresa de gastos.

En caso de que el coste total de gestión del inventario exceda las posibilidades presupuestarias de la empresa, se puede plantear la posibilidad de excluir algunos productos del inventario para reducir los costes. En este contexto, el **valor de Shapley 1.4** puede ser una herramienta útil para analizar qué productos contribuyen menos al ahorro total de costes y, por tanto, podrían ser considerados para su exclusión. Esto facilita la toma de decisiones informadas para optimizar los recursos disponibles.

Como se mencionó al inicio del trabajo, el cálculo exacto del valor de Shapley puede resultar computacionalmente costoso cuando el número de jugadores (en este caso, los productos) es elevado. Sin embargo, este valor se puede aproximar utilizando técnicas de simulación 1.6. A continuación, se presenta un código que adapta el enfoque mostrado en el primer capítulo del presente trabajo para realizar esta aproximación en el caso del inventario de la empresa.

```

1 # Caso CON costes eximibles
2 #####
3
4 shapley_montecarlo_decastro <- function(h, d, c, K, n_iterations = 10000) {
5
6   n <- length(h)
7   shapley <- numeric(n) # se almacenarán las contribuciones
8
9   # Iterar a través de un número grande de simulaciones
10  for (iter in 1:n_iterations) {
11    # Generar una permutación aleatoria de los jugadores

```

```

12 permutation <- sample(1:n)
13 cost_prev <- 0
14
15 # Calcular contribuciones marginales para la permutación
16 for (j in 1:n) {
17   current_player <- permutation[j]
18   coalition <- permutation[1:j] # Coalición actual en la permutación
19
20   # Calcular el costo con el jugador actual
21   h_coalicion <- h[coalition]
22   d_coalicion <- d[coalition]
23   c_coalicion <- c[coalition]
24   sum_h_d <- sum(h_coalicion* d_coalicion)
25   sum_c_d <- sum(c_coalicion* d_coalicion)
26
27   cost_current <- min(sqrt(2*a*sum(h_coalicion * d_coalicion)),
28                       K / 2 * ( sum_h_d / sum_c_d))
29
30
31   # Contribución marginal del jugador actual
32   marginal_contribution <- cost_current - cost_prev
33   shapley[current_player] <- shapley[current_player] + marginal_contribution
34
35   # Actualizar el costo previo
36   cost_prev <- cost_current
37 }
38 }
39
40 # Dividir por el número de iteraciones para obtener el promedio
41 shapley <- shapley / n_iterations
42 return(shapley)
43 }
44
45 datos$shapley_values <- shapley_montecarlo_decastro(datos$h, datos$d, datos$c, K=200000,
46   n_iterations = 10000)
47 summary(datos$shapley_values)

```

```

1   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
2 -75.816 -23.364  -1.875   9.181  40.462 112.150

```

```

1 # Caso SIN costes eximibles
2 #####
3
4 shapley_montecarlo_decastro_new <- function(h, d, c, n_iterations = 10000) {
5
6   n <- length(h)
7   shapley <- numeric(n) # se almacenarán las contribuciones
8
9   # Iterar a través de un número grande de simulaciones
10  for (iter in 1:n_iterations) {
11    # Generar una permutación aleatoria de los jugadores
12    permutation <- sample(1:n)
13    cost_prev <- 0
14

```

```

15 # Calcular contribuciones marginales para la permutación
16 for (j in 1:n) {
17   current_player <- permutation[j]
18   coalition <- permutation[1:j] # Coalición actual en la permutación
19
20   # Calcular el costo con el jugador actual
21   h_coalicion <- h[coalition]
22   d_coalicion <- d[coalition]
23   c_coalicion <- c[coalition]
24   sum_h_d <- sum(h_coalicion* d_coalicion)
25   sum_c_d <- sum(c_coalicion* d_coalicion)
26
27   cost_current <- sqrt(2*a*sum(h_coalicion * d_coalicion))
28
29   # Contribución marginal del jugador actual
30   marginal_contribution <- cost_current - cost_prev
31   shapley[current_player] <- shapley[current_player] + marginal_contribution
32
33   # Actualizar el costo previo
34   cost_prev <- cost_current
35 }
36 }
37
38 # Dividir por el número de iteraciones para obtener el promedio
39 shapley <- shapley / n_iterations
40 return(shapley)
41 }
42
43 datos$shapley_values_new <- shapley_montecarlo_decastro_new(datos$h, datos$d, datos$c, n_
44   iterations = 10000)
45 summary(datos$shapley_values_new)

```

```

1  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
2   2.35   18.87   40.87   51.33   73.98  158.39

```

Como principal diferencia, considerar la posibilidad de eximir de gastos a la empresa, produce que existan valores de Shapley negativos, lo que refleja que ciertos artículos contribuyen menos que el coste promedio esperado en la cadena de suministro conjunta, mientras que si no se cuenta con dicha posibilidad, los valores de Shapley son siempre positivos, indicando que todos los productos contribuyen de manera proporcional al coste global.

Puede comprobarse que en efecto se verifica la condición de eficiencia:

```
1 sum(datos$shapley_values_new); coste_sin_c
```

```
1 [1] 5132.773
```

```
2 [1] 5132.773
```

```
1 sum(datos$shapley_values) ; coste
```

```
1 [1] 918.1305
```

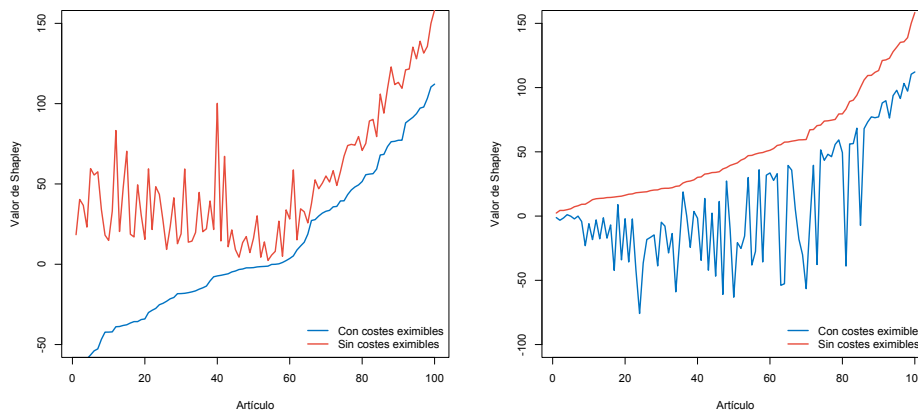
```
2 [1] 918.1305
```

La comprobación de eficiencia confirma que el modelo con costes eximibles asigna un valor total idéntico al coste conjunto (918.13). Esto valida la implementación del algoritmo y refuerza la consistencia teórica del modelo. Sin embargo, en el modelo sin costes eximibles, el valor total es significativamente mayor (5132.77).

Este resultado sugiere que los costes eximibles actúan como un factor limitante que concentra los costes en menos productos, mientras que eliminarlos distribuye las cargas de manera más uniforme. Desde una perspectiva práctica, esto podría indicar que los costes eximibles son más efectivos para fomentar la cooperación en grupos pequeños, mientras que su eliminación beneficia a grandes conjuntos.

Cabe mencionar que, como se muestra en las Tablas 2.2, los artículos con menor valor de Shapley en el primero de los modelos (con costes eximibles) tienden a tener demandas d_i más altas y no coinciden con los que tienen menor valor de Shapley en el segundo modelo (sin costes eximibles). No obstante, en las dos situaciones, los valores de Shapley más bajos, tienden a tener a artículos con costes c_i elevados.

En la Figura 2.2 se puede observar como para este ejemplo siempre se verifica la condición $2\sqrt{\frac{2a}{\sum_{j \in S} h_j d_j}} \geq \frac{A}{\sum_{j \in S} c_j d_j}$, y por ello, al igual que ocurría con tamaños óptimos Q_i^* , los valores de Shapley calculados para el modelo con y sin costes eximibles, no coinciden para ninguno de los artículos i .



(a) Artículos ordenados según el ϕ_i del modelo con costes eximibles (b) Artículos ordenados según el ϕ_i del modelo sin costes eximibles

Figura 2.2: Representación de los valores de Shapley ϕ_i .

Ítem	d_i	c_i	ϕ_i	Ítem	d_i	c_i	ϕ_i
43	267	80.92	-75.81559	91	8	96.29	2.349875
17	260	85.04	-63.12315	92	17	94.10	4.375746
79	266	79.65	-60.93498	4	18	68.20	4.378106
11	248	74.10	-58.97242	27	11	64.13	4.973714
95	247	97.51	-56.49728	12	18	62.65	5.683607
75	227	98.37	-53.89408	47	27	71.42	7.263055
30	258	86.42	-52.70013	76	20	86.34	8.017788
86	244	70.22	-46.74324	26	30	95.94	9.164871
67	157	81.65	-42.27411	8	122	58.53	9.230771
22	141	91.05	-42.25400	35	47	73.89	10.898995

(a) Modelo con costes eximibles. (b) Modelo sin costes eximibles.

Tabla 2.2: 10 primeros valores de Shapley con y sin costes eximibles.

2.3. Modelo determinista con múltiples agentes

En esta sección se analiza una situación en la que varias empresas deciden realizar pedidos de manera conjunta, introduciendo un modelo de inventario en el que cada empresa gestiona un único artículo. Este modelo, denominado problema EOQ multientidad con costes de pedido eximibles, permite explorar cómo la cooperación influye en la optimización de los costes totales.

El modelo se caracteriza por los siguientes parámetros fundamentales:

- N : conjunto de empresas que cooperan en la gestión de pedidos.
- $d_i > 0$: **demanda** del artículo de la empresa i por unidad de tiempo.
- $h_i > 0$: **coste de almacenamiento** por unidad del producto de la empresa i por unidad de tiempo.
- $c_i > 0$: **coste unitario** del artículo de la empresa i .
- $a > 0$: **coste de realizar un pedido**, compartido por todos los artículos de las diferentes empresas.
- $A > 0$: **coste mínimo** del total pedido a partir del cual el proveedor **exime** a las empresas del coste de pedido.

En este contexto las variables de decisión son los tamaños óptimos de pedidos de las empresas Q_i para todo $i \in N$. Además se asume que las empresas acuerdan realizar pedidos conjuntos,

unificando sus demandas individuales para beneficiarse de economías de escala. En definitiva, al igual que en el apartado anterior, se cumplirá la condición $\frac{Q_i}{d_i} = \frac{Q_j}{d_j}$ para todas las $i, j \in N$.

A partir de esta relación, y aplicando principios similares a los utilizados en el análisis de modelos de inventario individuales, se pueden determinar el tamaño óptimo de pedido conjunto y los costes mínimos asociados a cualquier coalición de empresas $S \subseteq N$. Esto permite construir un juego de costes cooperativo en el que se analiza cómo repartir el coste total de manera justa y eficiente.

El principal desafío en este modelo es garantizar que la distribución de costes sea aceptable para todas las empresas participantes, evitando que algún subgrupo decida abandonar la cooperación. Desde el punto de vista de la teoría de juegos, esta estabilidad coalicional puede lograrse si el núcleo del juego de costes asociado es no vacío.

El **núcleo** (*core*), definido formalmente como el conjunto de asignaciones que cumplen con la eficiencia y con las restricciones de estabilidad para todas las coaliciones, asegura que ninguna subcoalición pueda reducir su coste actuando de forma independiente. Sin embargo, en ciertos casos, el valor de Shapley, una regla de reparto ampliamente utilizada, puede no pertenecer al núcleo. Por esta razón, es útil considerar alternativas de asignación que garanticen la estabilidad, como la **regla de asignación proporcional hd** que ajusta los costes según la contribución marginal de cada empresa.

Definición 18. La **regla hd-proporcional** es una regla de asignación para juegos de inventario cooperativos, que distribuye los costes totales entre los participantes proporcionalmente al producto de dos factores clave: las demandas d_i y los costes de almacenamiento h_i . Dada una coalición $S \subseteq N$ se define para cada jugador i :

$$\phi_i^{hd}(v) = \frac{h_i d_i}{\sum_j h_j d_j} C(Q_i^*) \quad (2.14)$$

donde la función de coste total $C(Q_i^*)$ variará según se considere o no la posibilidad de eximir de costes a las empresas.

Al igual que el valor de Shapley y el valor de Owen, la regla hd-proporcional verificará la propiedad de **eficiencia**:

$$\sum_{i \in N} \phi_i^{hd}(v) = C(Q_i^*)$$

Aunque los fundamentos matemáticos del modelo en esta sección son similares al caso anterior (donde se gestionaban múltiples artículos dentro de una única empresa), el enfoque y la interpretación son diferentes. Aquí, el análisis se centra en la cooperación entre empresas y cómo repartir los costes totales derivados de la gestión conjunta de inventarios.

En este contexto, cuando los costes de pedido pueden ser eximidos, la regla hd-proporcional resulta especialmente adecuada, ya que garantiza repartos dentro del *core*, ofreciendo soluciones estables y equitativas para las empresas participantes (Fiestras-Janeiro et al. (2024)). Sin embargo, en el caso en que los costes de pedido no puedan ser eximidos, tanto la regla hd-proporcional como el valor de Shapley se encuentran dentro del *core*, y por ello, ambos enfoques son válidos (Meca et al. (2004)).


En el ejemplo que se desarrolla a continuación, se contrastarán los resultados del valor de Shapley y de la regla proporcional hd utilizando los mismos datos para facilitar la comparación entre ambos enfoques y analizar sus implicaciones en diferentes escenarios.

2.3.1. Ejemplo ilustrativo

Para ilustrar el modelo con múltiples empresas, adaptaremos el ejemplo previamente analizado de la empresa de materiales eléctricos, donde se gestionaban múltiples productos. En esta ocasión, consideraremos una situación en la que varias sucursales están interesadas en evaluar el coste de realizar pedidos conjuntos de un único producto: un material eléctrico específico.

Supongamos que cada sucursal tiene diferentes demandas para este material, así como costes individuales asociados al almacenamiento y adquisición de tal producto. Además, se mantiene la posibilidad de que el proveedor exima a las empresas de los costes de pedido.

Este planteamiento nos permitirá analizar cómo la cooperación entre las sucursales afecta al coste total y cómo distribuir de forma equitativa los beneficios de realizar un pedido conjunto. Para ello, utilizaremos los mismos principios y herramientas matemáticas que en el modelo anterior, adaptados a este nuevo contexto de múltiples sucursales y un único producto.

A continuación, se propone un posible código de  que ayudará al cálculo de la regla hd-proporcional.

```

1 # Caso CON costes eximibles
2 #####
3 hp_prop <- numeric(length(d))
4
5 # Calcular el valor total de sum(h*d)
6 sumatorio <- sum(c * d)
7
8 # Bucle para calcular hp
9 for (i in 1:length(d)) {
10   hp_prop[i] <- (200000 * h[i] * d[i]) / (2 * sumatorio)
11 }
12
13 # Mostrar los resultados
14 datos$hp_prop <- hp_prop
15 summary(datos$hp_prop)

```



```

1   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
2   0.2899  3.0274  7.0229  9.1813 13.3719 29.9457

1 # Caso SIN costes eximibles
2 #####
3 hp_prop_new <- numeric(length(d))
4
5 # Calcular el valor total de sum(h*d)
6 sumatorio_2 <- sum(h * d)
7
8 # Bucle para calcular hp
9 for (i in 1:length(d)) {
10  hp_prop_new[i] <- h[i] * d[i] * sqrt ( (2 * a) / (sumatorio_2) )
11 }
12
13 # Mostrar los resultados
14 datos$hp_prop_new <- hp_prop_new
15 summary(datos$hp_prop_new)

1   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
2   1.621  16.925  39.261  51.328  74.755 167.410

1 # Comprobación de la eficiencia
2 sum(hp_prop_new) ; coste_sin_c

1 [1] 5132.773
2 [1] 5132.773

1 # Comprobación de la eficiencia
2 sum(hp_prop) ; coste

1 [1] 918.1305
2 [1] 918.1305

```

Como se muestra en las Tablas 2.3, los valores de la regla proporcional (hd) en el modelo con costes eximibles no coinciden con los valores obtenidos en el modelo sin costes eximibles, no obstante, la ordenación creciente en ambos casos sí, coincide.

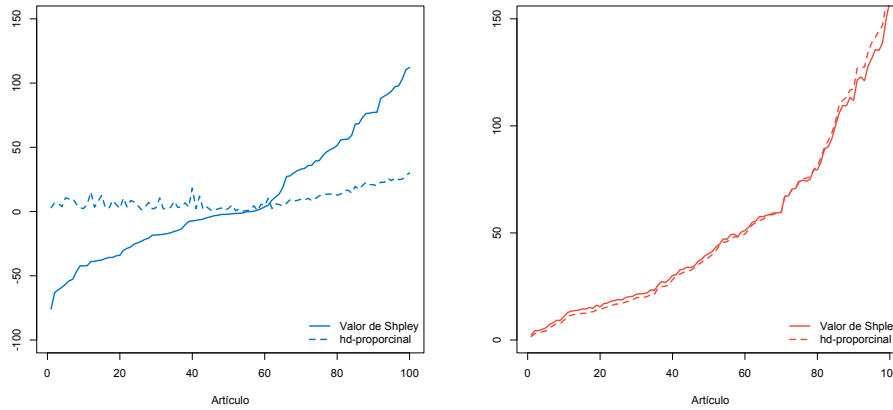
Es interesante observar cómo la ordenación creciente según los valores de ϕ_i^{hd} coincide con la ordenación creciente de los valores de Shapley ϕ_i del modelo sin costes eximibles 2.2. Los artículos con menores valores de hd-proporcional corresponden de nuevo a agentes con demandas más altas, lo cual refleja una distribución proporcional del coste total basado en la relevancia de cada agente dentro de la coalición.

En la Figura 2.3 se observa cómo se comportan los valores de Shapley y hd-proporcionales.

Ítem	d_i	c_i	ϕ_i	ϕ_i^{hd}	Ítem	d_i	c_i	ϕ_i	ϕ_i^{hd}
91	8	96.29	-1.17571591	0.2899503	91	8	96.29	2.349875	1.620956
92	17	94.10	-3.27063419	0.5450509	92	17	94.10	4.375746	3.047086
4	18	68.20	-1.46715218	0.5771127	4	18	68.20	4.378106	3.226326
27	11	64.13	1.10816453	0.6900260	27	11	64.13	4.973714	3.857564
12	18	62.65	-0.10350849	0.7527557	12	18	62.65	5.683607	4.208251
47	27	71.42	-2.22310734	1.0162202	47	27	71.42	7.263055	5.681139
76	20	86.34	-0.02673051	1.1988331	76	20	86.34	8.017788	6.702030
26	30	95.94	-23.04271667	1.3605362	26	30	95.94	9.164871	7.606025
8	122	58.53	-4.20874200	1.3800521	8	122	58.53	9.230771	7.715128
35	47	73.89	-5.92733277	1.6379406	35	47	73.89	10.898995	9.156843

(a) Modelo con costes eximibles.

(b) Modelo sin costes eximibles.

Tabla 2.3: 10 primeros valores de Shapley y hd-proporcianl con y sin costes eximibles.(a) Artículos ordenados según el ϕ_i del modelo con costes eximibles(b) Artículos ordenados según el ϕ_i del modelo sin costes eximibles**Figura 2.3:** Representación de los valores de Shapley y regla hd-proporcional ϕ_i .

2.4. Modelo determinista con múltiples artículos y agentes

En esta sección, se considera un escenario más general y complejo que los discutidos previamente: múltiples artículos y múltiples empresas. En este caso, varias empresas cooperan para realizar pedidos conjuntos de diferentes artículos, buscando minimizar el coste total de inventario y gestión. Este modelo refleja situaciones reales donde varias compañías, que comparten provee-

dores o intereses comunes, buscan optimizar sus cadenas de suministro de manera coordinada.

El problema de múltiples artículos y múltiples empresas introduce una mayor heterogeneidad en las demandas, costes de adquisición y costes de almacenamiento. Cada empresa tiene necesidades específicas para cada artículo, lo que genera un sistema interconectado de decisiones que afecta a todos los participantes. Este tipo de interacción requiere no solo calcular el coste total óptimo, sino también definir un mecanismo justo y estable para repartir dichos costes entre las empresas y los artículos implicados.

Como ya se ha visto en las secciones anteriores, el reparto entre artículos trata de asignar a cada uno de ellos su influencia en los costes conjuntos de pedido e inventario, por lo que puede basarse en una regla de reparto que busque ecuanimidad, como el valor de Shapley. El reparto entre las empresas trata de asignar a cada una de ellas lo que debe contribuir a los costes conjuntos de pedido e inventario, por lo que debe hacerse en una regla de reparto que busque la estabilidad, es decir, que proponga repartos en el core. En el caso del problema con costes de pedido eximibles, el valor de Shapley puede no estar en el core (Fiestras-Janeiro et al. (2024)), por lo que se sugirió usar la regla hd-proporcional que sí lo está. En cambio, en el caso del problema sin costes de pedido eximibles, tanto el valor de Shapley como la regla hd-proporcional proporcionan repartos en el core (Meca et al. (2004)).

En esta sección, se pretende repartir los costes conjuntos de pedido e inventario primero entre diferentes empresas y después, dentro de cada empresa, entre sus productos. El reparto entre las empresas debe ser estable y el reparto entre los productos de cada empresa debe reflejar su influencia en los costes asignados a la empresa. Para ello, en esta sección será necesario usar reglas para juegos con uniones a priori y, en concreto, usaremos el valor de Owen, que proporciona repartos entre las empresas estables en el caso de los costes de pedido no eximibles.

Al igual que en el modelo básico y el modelo con múltiples artículos/agentes, se desarrollarán dos escenarios: el caso donde el proveedor puede eximir de los costes de pedido si se supera un cierto umbral de pedido total, y el caso donde esta exención no es posible.

2.4.1. Modelo con costes eximibles

En este caso, se permite que las empresas reciban una exención de los costes de pedido si el coste total del pedido, sumando los artículos y empresas, supera un umbral predefinido A . Los parámetros a tener en cuenta serán:

- $N = \{1, \dots, n\}$: conjunto de artículos.
- $M = \{1, \dots, m\}$: conjunto de empresas.

- d_{ij} : demanda del artículo j por parte de la empresa i .
- h_{ij} : coste de almacenamiento por unidad del artículo j gestionado por la empresa i .
- c_j : coste unitario del artículo j .
- a : coste de realizar un pedido conjunto (compartido entre las empresas).
- A : coste mínimo del pedido total para que el coste fijo de pedido sea eximido.

El objetivo de este nuevo modelo será determinar el tamaño del pedido Q_{ij} del artículo j solicitado por la empresa i .

Se asumirá que todas las empresas coordinan sus pedidos de manera que los ciclos de reposición son iguales para todos los artículos. Esto implica que: $\frac{Q_{ij}}{d_{ij}} = T$ para toda empresa i y todo artículo j .

La función de **coste promedio por ciclo**, considerando los artículos de un subconjunto $S \subseteq N$ gestionados por un subconjunto $U \subseteq M$, puede expresarse como:

$$C(\{Q_{ij}\}_{i \in S, j \in U}) = \begin{cases} \frac{a}{T} + \frac{T}{2} \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}, & \text{si } \sum_{i \in S} \sum_{j \in U} c_i Q_{ij} < A, \\ \frac{T}{2} \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}, & \text{si } \sum_{i \in S} \sum_{j \in U} c_i Q_{ij} \geq A. \end{cases} \quad (2.15)$$

Aquí, el término $\frac{a}{T}$ representa el **coste de pedido promedio**, mientras que el término $\frac{T}{2} \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}$ corresponde al **coste de almacenamiento promedio**, calculado como la suma ponderada de los costes de mantener inventario para cada artículo y empresa.

Al derivar respecto al ciclo T e igualar a cero, se puede encontrar el **ciclo óptimo de reposición** en el primero de los casos; en el segundo hay que tener en cuenta el carácter creciente de la función. Se obtiene entonces:

- Si $\sum_{i \in S} \sum_{j \in U} c_i Q_{ij} < A$:

$$T^* = \sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}}.$$

- Si $\sum_{i \in S} \sum_{j \in U} c_i Q_{ij} \geq A$:

$$T^* = \frac{A}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}.$$

El **tamaño óptimo del pedido** para cada artículo i y cada empresa j puede calcularse a partir del ciclo óptimo T^* como:

$$Q_{ij}^* = \begin{cases} d_{ij} \sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}}, & \text{si } 2\sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}} < \frac{A}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}, \\ \frac{d_{ij} A}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}, & \text{si } 2\sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}} \geq \frac{A}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}. \end{cases} \quad (2.16)$$

El **coste mínimo total** puede expresarse finalmente como:

$$C(Q_{ij}^*) = \begin{cases} \sqrt{2a \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}, & \text{si } 2\sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}} < \frac{A}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}, \\ \frac{A}{2} \frac{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}, & \text{si } 2\sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}} \geq \frac{A}{\sum_{i \in S} \sum_{j \in U} c_i d_{ij}}. \end{cases} \quad (2.17)$$

2.4.2. Modelo sin costes eximibles

En este caso, no se contempla un umbral mínimo A para eximir los costes de pedido. Esto implica que el coste de realizar un pedido, a , se aplica independientemente del tamaño total del pedido conjunto.

En este caso, la **función de coste promedio por ciclo** para los artículos gestionados por un subconjunto $S \subseteq N$ y un subconjunto $U \subseteq M$ se expresa como:

$$C(\{Q_{ij}\}_{i \in S, j \in U}) = \frac{a}{T} + \frac{T}{2} \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}, \quad (2.18)$$

donde $\frac{a}{T}$ es el **coste de pedido promedio** y $\frac{T}{2} \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}$ representa el **coste de almacenamiento promedio**.

El **ciclo óptimo** T^* se obtiene minimizando la función de coste promedio respecto a T . Derivando y resolviendo, se tiene:

$$T^* = \sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}}. \quad (2.19)$$

El **tamaño óptimo del pedido** para cada artículo i y cada empresa j se calcula como:

$$Q_{ij}^* = d_{ij} T^* = d_{ij} \sqrt{\frac{2a}{\sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}}. \quad (2.20)$$

Sustituyendo T^* en la función de coste promedio, se obtiene el **coste mínimo total**:

$$C(Q_{ij}^*) = \sqrt{2a \sum_{i \in S} \sum_{j \in U} h_{ij} d_{ij}}. \quad (2.21)$$

En este modelo, el coste de pedido siempre contribuye al coste total, independientemente del tamaño del pedido conjunto. Por lo tanto, los resultados son más sencillos en comparación con el caso donde existen costes eximibles.

En los apartados anteriores se ha discutido cómo calcular el tamaño óptimo de los pedidos y los costes mínimos asociados a los diferentes modelos de inventario. Una vez obtenidos estos costes, surge la cuestión de cómo repartirlos entre los participantes del sistema: múltiples empresas y múltiples artículos.

A continuación, con el ejemplo utilizado en los anteriores apartados, se ilustrará cómo calcular el valor de Owen con \mathbb{R} y se analizará su utilización para repartir los costes óptimos obtenidos en los modelos de inventario discutidos previamente. Compararemos este reparto con los obtenidos mediante el valor de Shapley y la regla hd-proporcional para destacar las ventajas y diferencias del enfoque propuesto.

Es importante señalar que el artículo [Fiestras-Janeiro et al. \(2024\)](#) aborda la regla de reparto de Shapley proporcional en el contexto de modelos con costes eximibles. No obstante, dado que el objetivo principal de este trabajo es analizar los modelos sin costes eximibles, y por razones de extensión, este concepto no se desarrollará en detalle aquí. Sin embargo, se deja como una posible línea futura de investigación explorar en profundidad esta regla de reparto en el marco de los modelos con costes eximibles, para lo cual dicho artículo sirve como referencia clave.

2.4.3. Ejemplo ilustrativo

Para ilustrar el modelo con múltiples empresas y múltiples artículos, adaptaremos el enfoque previamente analizado. En esta ocasión, consideraremos una situación en la que varias empresas están interesadas en evaluar el coste de realizar pedidos conjuntos para una variedad de productos.

Supongamos que cada empresa gestiona un conjunto específico de artículos, como se muestra en la Tabla 2.4 con demandas, costes de almacenamiento y precios unitarios que varían entre las empresas y los productos. Además, el proveedor puede ofrecer la posibilidad de eximir los costes de pedido si el volumen total del pedido conjunto alcanza un umbral determinado.

i	j	d_{ij}	h_{ij}
1	47	27	0.27
1	67	157	0.13
1	90	38	0.47
1	93	73	0.35
1	24	359	0.11
1	23	481	0.09
1	14	157	0.31
1	31	215	0.35
1	36	363	0.19
1	68	298	0.44
1	74	320	0.44
1	46	334	0.43
1	70	467	0.35
1	58	481	0.34
1	21	352	0.49
2	92	17	0.23
2	76	20	0.43
2	43	267	0.08
2	59	73	0.22
2	15	94	0.23
2	94	387	0.12
2	7	310	0.19
2	30	258	0.28
2	48	415	0.22
2	66	351	0.33
2	28	430	0.35
2	6	430	0.47
3	26	30	0.33
3	19	95	0.15
3	5	199	0.12
3	55	170	0.15
3	16	95	0.34
3	33	82	0.40
3	50	313	0.10
3	3	183	0.32
3	37	377	0.23
3	69	281	0.39
3	100	420	0.23
3	20	445	0.22
3	83	346	0.30
3	9	303	0.48
4	22	141	0.12
4	40	36	0.43
4	63	145	0.15
4	77	120	0.19
4	11	248	0.11
4	51	139	0.23
4	34	85	0.49
4	95	247	0.31
4	80	239	0.32
5	39	147	0.13
5	60	76	0.21
5	61	80	0.29
5	18	201	0.19
5	54	228	0.25
5	65	357	0.17
5	44	397	0.16
5	96	413	0.16
5	10	233	0.37
5	72	435	0.22
5	52	313	0.31
5	25	332	0.36
5	97	312	0.48
6	91	8	0.26
6	4	18	0.23
6	12	18	0.30
6	8	122	0.08
6	35	47	0.25
6	82	90	0.17
6	71	45	0.33
6	86	244	0.17
6	57	294	0.21
6	85	163	0.46
6	98	282	0.32
6	81	460	0.27
6	32	424	0.42
6	13	378	0.48
7	38	336	0.05
7	73	94	0.29
7	99	93	0.28
7	87	213	0.19
7	78	98	0.37
7	64	214	0.24
7	29	141	0.38
7	75	227	0.31
7	53	415	0.15
7	49	448	0.16
7	41	494	0.15
7	45	429	0.38
7	88	378	0.49
7	1	419	0.45
7	2	467	0.46
8	27	11	0.45
8	42	113	0.16
8	84	43	0.46
8	62	67	0.38
8	79	266	0.17
8	17	260	0.19
8	56	221	0.28
8	89	310	0.33

Tabla 2.4: Distribución de los ítems por grupos con sus características.

Este planteamiento nos permitirá analizar cómo la cooperación entre las empresas, considerando sus respectivos conjuntos de artículos, afecta al coste total de la cadena de suministro y cómo distribuir de forma ecuánime los beneficios derivados de esta cooperación. En este caso, será fundamental tener en cuenta la interacción entre los niveles jerárquicos (empresas y sus artículos), lo que hace necesario un enfoque basado en el valor de Owen para el reparto de costes.

A continuación se propone un posible código que ayudará al cálculo del valor de Owen de manera aproximada como se ha desarrollado en el primer capítulo de este trabajo 1.8:

```

1 # Definición de los grupos (artículos de las empresas)
2 grupo_1 <- c(47, 67, 90, 93, 24, 23, 14, 31, 36, 68, 74, 46, 70, 58, 21)
3
4 grupo_2 <- c(92, 76, 43, 59, 15, 94, 7, 30, 48, 66, 28, 6)
5
6 grupo_3 <- c(26, 19, 5, 55, 16, 33, 50, 3, 37, 69, 100, 20, 83, 9)
7
8 grupo_4 <- c(22, 40, 63, 77, 11, 51, 34, 95, 80)
9
10 grupo_5 <- c(39, 60, 61, 18, 54, 65, 44, 96, 10, 72, 52, 25, 97)
11
12 grupo_6 <- c(91, 4, 12, 8, 35, 82, 71, 86, 57, 85, 98, 81, 32, 13)
13
14 grupo_7 <- c(38, 73, 99, 87, 78, 64, 29, 75, 53, 49, 41, 45, 88, 1, 2)
15
16 grupo_8 <- c(27, 42, 84, 62, 79, 17, 56, 89)
17 grupos_list <- list(grupo_1 = grupo_1, grupo_2 = grupo_2, grupo_3 = grupo_3, grupo_4 =
    grupo_4, grupo_5 = grupo_5, grupo_6 = grupo_6, grupo_7 = grupo_7, grupo_8 = grupo_8)

1 # Owen Caso CON costes eximibles
2 #####
3
4 owen_value <- function(grupos_list, h, d, c, K, a, n_iterations = 10000) {
5   n <- length(h)
6   owen_values <- numeric(n) # Para almacenar el valor de Owen
7
8   # Iterar a través de un número de simulaciones
9   for (iter in 1:n_iterations) {
10
11     # Generar una permutación aleatoria de los grupos
12     permuted_groups <- sample(seq_along(grupos_list))
13
14     # Lista para almacenar la permutación total de los
15     # jugadores respetando los grupos
16
17     permuted_players <- c()
18
19     # Crear permutación total respetando los grupos
20     for (k in permuted_groups) {
21       grupo <- grupos_list[[k]]
22
23       # Generar una permutación aleatoria de los jugadores dentro del grupo
24       permuted_players <- c(permuted_players, sample(grupo))
25     }

```



```

26
27   cost_prev <- 0
28
29   # Calcular contribuciones marginales para la permutación de jugadores
30   for (j in seq_along(permuted_players)) {
31     current_player <- permuted_players[j]
32     coalition <- permuted_players[1:j] # Coalición actual
33
34     # Calcular el costo actual
35     h_coalicion <- h[coalition]
36     d_coalicion <- d[coalition]
37     c_coalicion <- c[coalition]
38
39     # Cálculo de la función de costo
40     sum_h_d <- sum(h_coalicion * d_coalicion)
41     sum_c_d <- sum(c_coalicion * d_coalicion)
42
43     # cost_current <- sum_h_d * sqrt(2 * a / sum_h_d)
44     cost_current <- sum_h_d * min ( sqrt(2 * a / sum_h_d), K/(2* sum_c_d) )
45
46     # Contribución marginal
47     marginal_contribution <- cost_current - cost_prev
48     owen_values[current_player] <- owen_values[current_player] + marginal_contribution
49
50     # Actualizar el costo previo
51     cost_prev <- cost_current
52   }
53 }
54
55 # Promediar el valor de Owen
56 owen_values <- owen_values / n_iterations
57 return(owen_values)
58 }
59
60
61 owen <- owen_value(grupos_list, h, d, c, K=B, a, n_iterations = 10000)
62 datos$owen <- owen
63 summary(datos$owen)

```

```

1   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
2 -80.265 -23.374  -1.481   9.181  39.617 114.625

```

```

1 # Owen SIN costes eximibles
2 #####
3
4
5 owen_value_new <- function(grupos_list, h, d, c, a, n_iterations = 10000) {
6   n <- length(h)
7   owen_values <- numeric(n) # Para almacenar el valor de Owen
8
9   # Iterar a través de un número de simulaciones
10  for (iter in 1:n_iterations) {
11
12    # Generar una permutación aleatoria de los grupos

```

```

13   permuted_groups <- sample(seq_along(grupos_list))
14
15   # Lista para almacenar la permutación total de los
16   # jugadores respetando los grupos
17
18   permuted_players <- c()
19
20   # Crear permutación total respetando los grupos
21   for (k in permuted_groups) {
22     grupo <- grupos_list[[k]]
23
24     # Generar una permutación aleatoria de los jugadores dentro del grupo
25     permuted_players <- c(permuted_players, sample(grupo))
26   }
27
28   cost_prev <- 0
29
30   # Calcular contribuciones marginales para la permutación de jugadores
31   for (j in seq_along(permuted_players)) {
32     current_player <- permuted_players[j]
33     coalition <- permuted_players[1:j] # Coalición actual
34
35     # Calcular el costo actual
36     h_coalicion <- h[coalition]
37     d_coalicion <- d[coalition]
38     c_coalicion <- c[coalition]
39
40     # Cálculo de la función de costo
41     sum_h_d <- sum(h_coalicion * d_coalicion)
42     sum_c_d <- sum(c_coalicion * d_coalicion)
43
44     cost_current <- sum_h_d * sqrt(2 * a / sum_h_d)
45     # cost_current <- sum_h_d * min ( sqrt(2 * a / sum_h_d), K/(2* sum_c_d) )
46
47     # Contribución marginal
48     marginal_contribution <- cost_current - cost_prev
49     owen_values[current_player] <- owen_values[current_player] + marginal_contribution
50
51     # Actualizar el costo previo
52     cost_prev <- cost_current
53   }
54 }
55
56 # Promediar el valor de Owen
57 owen_values <- owen_values / n_iterations
58 return(owen_values)
59 }
60
61
62 owen_new <- owen_value_new(grupos_list, h, d, c, a, n_iterations = 10000)
63 datos$owen_new <- owen_new
64 summary(datos$owen_new)

```

```

1   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.

```

```
2 2.491 19.908 41.769 51.328 73.443 152.172
```

```
1 # Compruebo la eficiencia
2 sum(datos$owen); coste
```

```
1 [1] 918.1305
```

```
2 [1] 918.1305
```

```
1 # Compruebo la eficiencia
2 sum(datos$owen_new); coste_sin_c
```

```
1 [1] 5132.773
```

```
2 [1] 5132.773
```

Los resultados obtenidos muestran que, en el modelo con costes eximibles, los valores de Owen pueden tomar valores negativos. Esto refleja que ciertos jugadores o coaliciones contribuyen menos que el coste promedio esperado en la cadena de suministro conjunta. En el contexto del problema, estos valores negativos podrían interpretarse como un incentivo adicional para las empresas de menor tamaño o con menores demandas para cooperar, ya que se beneficiarían de una asignación más favorable.

En contraste, en el modelo sin costes eximibles, los valores son consistentemente positivos, indicando que todos los participantes contribuyen al coste global. Este resultado sugiere la idea de que eliminar los costes eximibles podría desincentivar la cooperación de aquellas con parámetros menores.

Comparando los valores extremos y promedio entre los modelos, observamos diferencias significativas. En el modelo con costes eximibles, el rango es más amplio (de -80.265 a 114.625), lo que indica una mayor disparidad en las contribuciones individuales. Esto sugiere que las características específicas de cada jugador (demanda, coste de almacenamiento, y precio) tienen un impacto más acentuado.

En el modelo sin costes eximibles, los valores oscilan entre 2.491 y 152.172, reflejando una distribución más homogénea. Este comportamiento sugiere que, al eliminar los costes eximibles, el sistema tiende a valorar más equitativamente a los participantes, independientemente de sus características iniciales.

En la Tabla 2.5 se pueden ver los 10 primeros artículos ordenados de manera creciente según el valor de Owen. Gráficamente, en la Figura 2.4 aparece el listado completo de cada uno de los ítems con sus correspondientes valores de Shapley, Owen y hd-proporcional para cada uno de los modelos estudiados.

i	j	ϕ_{ij}	ϕ_{ij}^{hd}	γ_{ij}	i	j	ϕ_{ij}	ϕ_{ij}^{hd}	γ_{ij}
2	43	-75.81559	2.977567	-80.26467	6	91	2.349875	1.620956	2.490737
7	75	-53.89408	9.809522	-62.97273	2	92	4.375746	3.047086	4.338295
2	30	-52.70013	10.070198	-55.21741	6	4	4.378106	3.226326	4.466792
8	17	-63.12315	6.886320	-49.79303	6	12	5.683607	4.208251	5.441458
8	79	-60.93498	6.303632	-47.86368	8	27	4.973714	3.857564	6.292076
1	67	-42.27411	2.845138	-46.98926	1	47	7.263055	5.681139	6.607892
7	87	-42.10494	5.641486	-46.79512	2	76	8.017788	6.702030	8.135918
6	86	-46.74324	5.782279	-46.27662	3	26	9.164871	7.715128	8.984638
3	69	-38.89538	15.276758	-44.47780	6	8	9.230771	7.606025	9.184204
4	11	-58.97242	3.802810	-42.63536	6	35	10.898995	9.156843	11.243221

(a) Modelo con costes eximibles.

(b) Modelo sin costes eximibles.

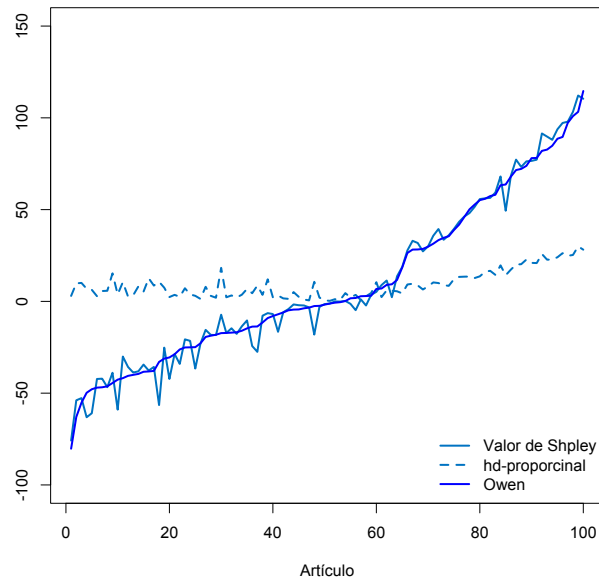
Tabla 2.5: 10 primeros valores Owen con y sin costes eximibles.

Con ayuda de la Figura 2.4 en el modelo con costes eximibles, se observa que gran cantidad de los valores de Shapley y Owen son negativos. Esto indica que algunas coaliciones están asumiendo cargas inferiores al promedio, lo cual puede atribuirse a los ajustes derivados de los costes eximibles. Este comportamiento sugiere que las coaliciones con menor demanda o influencia están siendo beneficiadas en este modelo, a costa de penalizar a otras coaliciones con mayores responsabilidades en la estructura de costes.

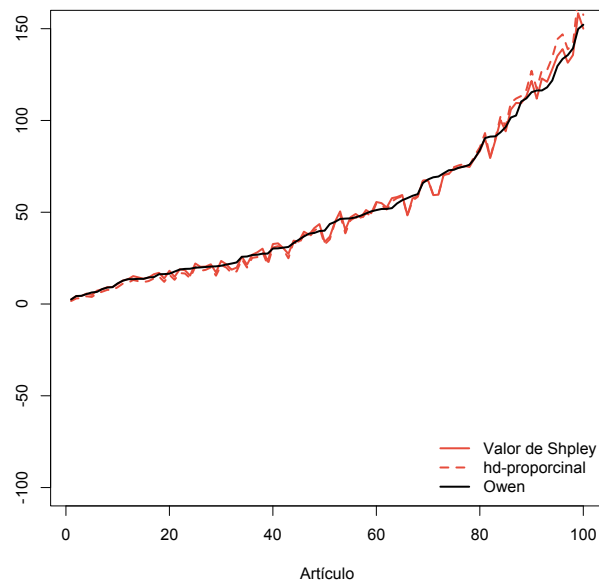
En este modelo, los valores positivos pero relativamente pequeños de la regla hd-proporcional reflejan que la alta demanda tiene un impacto reducido en este modelo, posiblemente debido a las restricciones impuestas por los costes eximibles. Como se muestra en la representación 2.4a las reglas hd-proporcionales se comportan de forma notablemente uniforme, mostrando valores que permanecen prácticamente constantes y formando una línea casi horizontal. Este patrón sugiere que esta regla no está adaptándose significativamente a las diferencias en la estructura de las coaliciones o la alta demanda, sino que distribuyen los costes de manera uniforme. Aunque esta uniformidad puede percibirse como una ventaja en contextos donde se busca minimizar disparidades, también evidencia una falta de sensibilidad a las particularidades del sistema, lo que podría considerarse una limitación en términos de equidad distributiva.

Por otro lado, en el modelo sin costes eximibles, todos los valores calculados son positivos, lo que indica que cada coalición contribuye de manera proporcional al coste total del sistema. En este modelo, la alta demanda juega un papel más importante, como lo evidencian los mayores valores de Shapley en comparación con el modelo con costes eximibles. Esto se debe a que, al eliminar las restricciones de los costes eximibles, los factores como la demanda y la influencia relativa de cada coalición tienen un mayor impacto en la distribución de los costes. Además,

destaca que los valores finales de Owen, Shapley y regla hd-proporcional están estrechamente alineados.



(a) Artículos ordenados según el ϕ_i del modelo con costes eximibles



(b) Artículos ordenados según el ϕ_i del modelo sin costes eximibles

Figura 2.4: Representación de los valores de Owen, Shapley y regla hd-proporcional.

Conclusiones

Para finalizar este trabajo, se expone a continuación un breve análisis y valoración de los resultados obtenidos, así como algunas limitaciones encontradas.

El primer capítulo está dedicado a establecer el marco teórico necesario para abordar el análisis de los modelos de inventario desde la perspectiva de la teoría de juegos cooperativos. En él se recopilan y desarrollan conceptos fundamentales como los juegos de utilidad transferible (TU), el núcleo, el valor de Shapley y el valor de Owen. Estos conceptos proporcionan una base sólida para el desarrollo del trabajo y permiten modelar y analizar la cooperación entre agentes en problemas de distribución de costes. La selección de una notación clara, basada principalmente en los trabajos de [Fiestras-Janeiro et al. \(2024\)](#) y [Casas-Méndez et al. \(2012\)](#), ha facilitado la comprensión y el desarrollo de las definiciones y soluciones planteadas.

En el segundo capítulo, se presentan distintas configuraciones del modelo EOQ (Economic Order Quantity), comenzando por el modelo básico determinista y extendiéndose a casos más complejos que consideran múltiples artículos, múltiples agentes, y la combinación de ambos. En cada caso, se han analizado variantes con y sin costes eximibles, permitiendo evaluar cómo la exención de costes afecta al reparto entre los agentes y a la estabilidad de las soluciones propuestas.

En este capítulo, se implementan los conceptos teóricos y los modelos de inventario mediante ejemplos prácticos y casos ilustrativos. Se hace especial énfasis en el cálculo del valor de Shapley y el valor de Owen en escenarios cooperativos, utilizando el software R para desarrollar herramientas computacionales que permitan calcular estas soluciones en contextos reales. Aunque el enfoque principal ha sido la distribución ecuánime de costes, los resultados obtenidos subrayan también la importancia de garantizar la estabilidad de los acuerdos entre agentes. No obstante, el análisis ha estado limitado por la complejidad computacional de algunos métodos para un elevado número de agentes o artículos.

Finalmente para cada uno de los modelos considerados, se analizan los conceptos desarrollados a un caso práctico basado en datos tomados de [Fiestras-Janeiro et al. \(2024\)](#). Se analiza cómo los modelos propuestos pueden ser utilizados para optimizar la gestión de inventarios en una

empresa con múltiples artículos y agentes. El estudio muestra cómo herramientas como el valor de Shapley proporcionan una base matemática robusta para el reparto de costes, destacando las contribuciones individuales de cada agente al ahorro global. Además, se ha demostrado que los modelos con costes eximibles ofrecen oportunidades significativas de reducción de costes en comparación con aquellos en los que no se contemplan estas exenciones.

En cuanto a las conclusiones principales del trabajo, los resultados obtenidos confirman que el uso de la teoría de juegos cooperativos en la gestión de inventarios no solo optimiza los costes totales, sino que también facilita acuerdos más justos y estables entre los participantes. Esto resulta particularmente útil en entornos empresariales donde la cooperación es esencial para reducir gastos y mejorar la eficiencia. No obstante, se recomienda para futuros estudios incorporar variabilidad en la demanda, análisis probabilísticos y herramientas de aprendizaje automático, que podrían ampliar significativamente la aplicabilidad de los modelos aquí desarrollados.

Bibliografía

- Casas-Méndez, B., Fiestras-Janeiro, M. G., García-Jurado, I., y González-Díaz, J. (2012). *Introducción a la Teoría de Juegos*. Servicio de Publicaciones e Intercambio Científico, Universidad de Santiago de Compostela.
- Castro, J., Gómez, D., y Tejada, J. (2009). Polynomial calculation of the shapley value based on sampling. *Computers & Operations Research*, 36:1726–1730.
- Fiestras-Janeiro, M. G., García-Jurado, I., Meca, A., y Mosquera, M. (2024). Evaluating the impact of items and cooperation in inventory models with exemptable ordering costs. *International Journal of Production Economics*, 269:109151.
- Meca, A., Timmer, J., García-Jurado, I., y Borm, P. (2004). Inventory games. *European Journal of Operational Research*, 156:127–139.
- Saavedra-Nieves, A., García-Jurado, I., y Fiestras-Janeiro, M. G. (2018). Estimation of the owen value based on sampling. In *The mathematics of the uncertain: A tribute to Pedro Gil*, pages 347–356. Springer.