



Universidad Miguel Hernández

**Facultad de Ciencias Sociales y Jurídicas de
Elche**

**Grado en Administración y Dirección de
Empresas**

Trabajo Fin de Grado

Curso 2015/2016

**OPTIMIZACIÓN Y MODELOS ECONÓMICOS
EN LA TOMA DE DECISIONES EMPRESARIALES.**

Alumno: Cristian Estañ Guirao.

Tutora: María Josefa Cánovas Cánovas.

RESUMEN

En este trabajo se analizan diferentes modelos matemáticos y econométricos orientados hacia la toma de decisiones empresariales. En términos generales, el objetivo primordial es la búsqueda de soluciones óptimas para los modelos planteados, con la finalidad última de facilitar la toma de decisiones en la empresa.

El trabajo está estructurado en tres capítulos. Los dos primeros desarrollan sendos modelos clásicos que pueden enmarcarse en el amplio campo de la Optimización Lineal. La Optimización (o Programación) Lineal es la rama de las matemáticas que se ocupa del estudio de problemas de optimización en los cuales tanto el objetivo (e.g., minimización de costes) como las restricciones que aparecen (e.g. limitaciones de capital, materia prima, mano de obra, etc.) pueden describirse mediante funciones lineales. Cuando las variables de estos problemas toman valores enteros, nos encontramos ante un problema de Programación Lineal Entera, como ocurre en el capítulo 2. Además, los dos primeros capítulos tienen en común el hecho de que las aplicaciones desarrolladas se ambientan en la empresa irlandesa Primark. Por su parte, el tercer capítulo hace uso de determinadas técnicas de análisis de series temporales.

Concretamente, el capítulo 1 aborda el problema logístico de planificación óptima del transporte de mercancías en la empresa. El modelo clásico de transporte o distribución plantea el objetivo de transportar, con el mínimo coste, determinada mercancía desde unos puntos (denominados nodos) de origen, hasta otros de destino. Además de los orígenes y destinos, el modelo puede incorporar otros nodos denominados de transbordo. En nuestro caso, nos ocuparemos de analizar un modelo de transporte de mercancía en la empresa Primark, contando con 4 nodos de origen, 44 destinos y 18 transbordos.

El capítulo 2 se ocupa del estudio de un modelo denominado del viajante de comercio, conocido como modelo TSP (del inglés Travelling Salesman Problem). El problema TSP plantea la búsqueda del camino más corto (o de menor coste, o menor tiempo, etc.) que parte de un punto inicial, recorre diferentes nodos (visitando una única vez cada uno de ellos) y regresa al mismo punto de partida. Como aplicación del mundo real, abordamos la cuestión de recorrer con el menor coste posible los diferentes centros logísticos de la empresa Primark (concretamente, 18 centros logísticos).

En el capítulo 3 se introducen diferentes herramientas básicas en el estudio de una serie temporal. Informalmente hablando, una serie temporal es una colección de datos correspondientes a una característica poblacional (e.g., tasa de ocupación hotelera, tasa de paro, etc.) que han sido extraídos en distinto instantes de tiempo. El objetivo de dichas herramientas es el de formular un modelo capaz de explicar el comportamiento de la serie y obtener predicciones futuras. Como aplicación real, el capítulo plantea el análisis del número de hipotecas solicitadas en la provincia de Alicante entre los años 1994 a 2014 (los datos han sido recogido del Instituto Nacional de Estadística).

Para concluir, destacamos el hecho de que se han resuelto de manera satisfactoria dos cuestiones logísticas en el seno de la empresa Primark, la primera relativa al transporte óptimo de su mercancía, y la segunda en relación con el recorrido óptimo de sus centros logísticos por parte del empresario. Por otro lado, se ha analizado una serie temporal de datos relativos a hipotecas. La finalidad última de los estudios contenidos en el presente trabajo es servir de ayuda en la toma de decisiones empresariales y financieras.



Contenido

INTRODUCCIÓN	1
CAPÍTULO 1.- MODELO DEL TRANSPORTE O DISTRIBUCIÓN	5
1.1 - Introducción	5
1.2. - Planteamiento del Modelo Básico de Transporte	7
1.3. - Formato Matricial	8
1.4. - Ejemplo Modelo de Transporte Simple.....	10
1.5. - Resolución del Modelo en Matlab.....	12
1.6. – Modelo de Transporte con Transbordo.....	13
1.6.1. - Ejemplo de Modelo de Transporte con Transbordo	13
1.7. – Resolución de un caso real: Un modelo de transporte en la empresa Primark	16
APÉNDICE 1.A: PROGRAMACIÓN EN MATLAB (Modelo de Transporte General)	25
APÉNDICE 1.B: CASO PARTICULAR DEL PRIMARK CON MATLAB.....	26
CAPÍTULO 2. - MODELO DEL VIAJANTE DE COMERCIO	29
2.1. - Introducción	29
2.2. – Planteamiento del Modelo Viajante de Comercio.....	31
2.3. – Formato Matricial.....	33
2.4. – Resolución del Modelo en Matlab	34
2.5. – Un caso real: Ruta óptima entre los centros logísticos de la empresa Primark	35
APÉNDICE 2.A: PROGRAMACIÓN EN MATLAB (Modelo del Viajante de Comercio General)	39
APÉNDICE 2.B: CASO PARTICULAR DEL PRIMARK CON MATLAB.....	39
CAPÍTULO 3.- SERIES TEMPORALES	41
3.1. - Introducción	41
3.2. – Herramientas Básicas del Estudio de una Serie Temporal	42
3.3. – Ejemplo y Resolución con Matlab.....	45
3.4. – Un caso real: Estudio de la evolución del número de hipotecas	50
APÉNDICE 3.A: MODELOS.....	54

APÉNDICE 3.B: Identificación del modelo del ejemplo introducido en la sección 3.3.	55
APÉNDICE 3.C: Identificación del modelo del ejemplo real introducido en la sección 3.4.	57
APÉNDICE 3.D: PROGRAMACIÓN EN MATLAB	59
1. - Programación de los correlogramas	59
2. - Programación de autocorrelación parcial	59
3. – Tipo de Transformaciones	60
APÉNDICE 3.E: SALIDAS DEL MATLAB.....	60
BIBLIOGRAFÍA.....	63



INTRODUCCIÓN

El presente trabajo de fin de grado se desarrolla en el seno del Grado de Administración y Dirección de Empresas. El objetivo del trabajo es doble; por un lado se pretenden diseñar estrategias de ayuda a la toma de decisiones en determinados ámbitos de la logística empresarial, y por otro, desarrollar herramientas de análisis de la evolución en el tiempo de determinadas variables, concretamente herramientas relacionadas con el estudio de series temporales.

Respecto del primer objetivo, que se aborda en los dos primeros capítulos del trabajo, en este momento comentamos que para su desarrollo emplearemos técnicas del campo de la Optimización Lineal. La Optimización Lineal es la rama de las Matemáticas que se ocupa del estudio de modelos consistentes en la optimización de un objetivo (e.g., minimización de costes, maximización de beneficios, etc.) que se encuentra sujeto a una serie de restricciones (e.g., limitaciones presupuestarias, de mano de obra, limitaciones derivadas de la oferta y demanda, entre otras). La Optimización Lineal es una rama activa de investigación desde mediados de los años 50, y constituye un núcleo importante dentro de la amplia disciplina de la Investigación Operativa, que también tuvo su origen a mediados del siglo XX. Por su parte, la Investigación Operativa podríamos definirla muy concisamente como la ciencia de la toma de decisiones. Dirigimos al lector, por ejemplo, a los textos de Bazaraa, Jarvis y Sherali, (2005), y Bertsimas y Tsitsiklis (1997) para una revisión de la teoría y los métodos de la Optimización Lineal, y a los de Hillier y Lieberman (2010), Viejo (1996), Taha (2004) para una perspectiva general de diferentes modelos de la Investigación Operativa.

Los dos primeros capítulos desarrollan determinados modelos de optimización ambientados en la empresa irlandesa Primark. Concretamente, el capítulo 1 se ocupa del estudio de un modelo de transporte en dicha empresa orientado hacia la búsqueda del mínimo coste de transporte, y el capítulo 2 pretende determinar el mínimo coste

empleado en la visita de diferentes centros logísticos de Primark ubicados en distintos países de la Unión Europea. En lo que sigue, emplearemos los términos ‘modelo’ y ‘problema’ como sinónimos. Asimismo, en ocasiones, alternativamente a la terminología ‘Optimización Lineal’, emplearemos ‘Programación Lineal’.

Para el desarrollo del segundo objetivo emplearemos determinadas herramientas provenientes de la teoría de los procesos estocásticos y series temporales, la cual puede ubicarse en el amplio campo de la Econometría. Concretamente en relación con este objetivo presentamos un estudio sobre la evolución del número de hipotecas concedidas para la compra de viviendas urbanas en la provincia de Alicante en los últimos 20 años.

El objetivo último del estudio consiste en buscar modelos explicativos que se ajusten aceptablemente a nuestra serie numérica (número de hipotecas anuales) y proporcionen estimaciones adecuadas del comportamiento futuro de la serie.

Seguidamente comentamos la estructura típica de un capítulo que comienza con una introducción, donde se motiva el modelo que vamos a estudiar y donde se mencionan algunas notas históricas para situarnos en el tiempo. Después de la introducción, pasamos al planteamiento del modelo explicando la formulación y la notación matemática necesaria para la comprensión del problema. A continuación, aparecen una serie de ilustraciones del modelo considerado mediante ejemplos sencillos que pueden encontrarse en la literatura. Seguidamente vendrá la aportación más importante del capítulo: la resolución de un caso real.

En cuanto a la resolución de los problemas que surgirán a lo largo del trabajo, en este momento adelantamos que será imprescindible el uso de un software informático para facilitar los cálculos. El programa informático escogido es el Matlab, que fue creado por el matemático C. Moler en 1984, y puede considerarse como una gran herramienta de cálculo y programación matemática usada tanto para fines didácticos, como profesionales.

Las principales contribuciones a destacar en líneas generales son: en primer lugar hemos hecho una pequeña recopilación bibliográfica sobre los diferentes temas a tratar, incluyendo diferentes ejemplos para comprender mejor los conceptos utilizados. Por motivos de completitud, y teniendo en cuenta que el programa Matlab no se estudia en la carrera, hemos incluido las herramientas básicas del manejo de dicho programa

necesarias para el desarrollo del trabajo. También destacamos como contribución original el hecho de que en cada capítulo se implementan unos programas específicos en Matlab capaces de resolver la situación considerada de manera genérica; esto es, de forma que el programa pueda ser utilizado por cualquier usuario simplemente introduciendo los datos numéricos que hubiera en cada caso.

La ventaja que tiene el empleo del Matlab a nivel formativo es, precisamente, el hecho de que nos permite diseñar los programas necesarios para la resolución de un caso práctico. En este sentido, nos proporciona una alta versatilidad, dado que nos permite modificar los elementos necesarios para adaptarnos a cada situación concreta.

Como conclusión podemos decir que en este trabajo se presentan modelos que expresan matemáticamente determinadas situaciones concretas procedentes del mundo empresarial y se resuelven con Matlab, creando para ello programas genéricos capaces de ofrecer soluciones a dichos modelos. De este modo, se obtienen conclusiones que facilitan la toma de decisiones en la empresa.



CAPÍTULO 1.- MODELO DEL TRANSPORTE O DISTRIBUCIÓN

1.1 - Introducción

Para el comienzo de dicho capítulo es necesario comprender a que nos referimos cuando se nombra el problema o modelo de transporte. El modelo del transporte es un problema de redes en Programación Lineal que se encarga de averiguar las cantidades a transportar de cierta mercancía desde unos puntos de origen hacia otros puntos de destino con el menor coste posible. Las restricciones que de modo natural presenta este modelo provienen de las cantidades ofertadas en los puntos de origen y las demandadas en los destinos.

En nuestro caso nos centraremos en el ámbito de la economía empresarial y concretamente en un modelo de transporte ubicado en la empresa Primark. Específicamente, consideramos el problema de transportar la mercancía procedente de 4 centros de producción (situados en Shanghái, Daca, Bombay y Karachi) hacia 44 puntos de venta de la cadena situados en diferentes países de la Unión Europea, teniendo en cuenta también la existencia de 18 centros logísticos de transbordo de mercancía.

Seguidamente, resumimos los núcleos temáticos del capítulo:

- **Modelo de transporte simple:** Esta sección introduce el planteamiento del problema, establece la notación necesaria y presenta la formulación general para su resolución. Asimismo, proporciona una ilustración del problema del transporte mediante un ejemplo académico.
- **Modelo de transporte con transbordo:** Ahora se completa el modelo anterior añadiendo nodos intermedios de transbordo por los cuales pasan las cantidades de los nodos de origen a sus destinos. Presentamos un ejemplo como ilustración de este modelo.

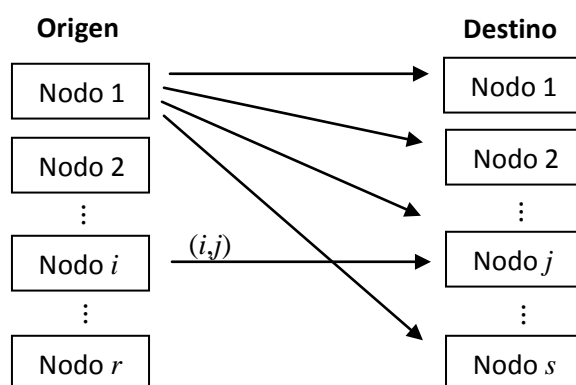
- Resolución en Matlab: Todos los ejemplos planteados en el capítulo están resueltos a través de este software, para ello debemos de aprender los diferentes comandos del programa para crear un modelo genérico capaz de resolver nuestro problema de transporte.
- Programación en Matlab: Destacamos el hecho de que este capítulo introduce un programa genérico elaborado con Matlab para resolver situaciones reales (véase el Apéndice 1.A). El programa ha sido desarrollado para que pueda ser utilizado por cualquier usuario. En nuestro caso, hemos resuelto el caso real de un modelo ambientado en la compañía irlandesa Primark.

Atendiendo a la evolución histórica del modelo de transporte, podríamos decir que en 1942 Kantorovich y Koopmans realizaron estudios de manera independiente sobre la planificación del transporte de ciertos productos desde sus centros de producción a sus centros de consumo; ambos proponen el llamado problema del transporte o también es conocido como problema de Koopmans-Kantorovich.

Durante la II Guerra Mundial dicho modelo se utilizó para resolver problemas militares y económicos de la época. Tras la guerra, se desarrollaron también técnicas de computación, que permitieron una importante reducción del tiempo de resolución de los problemas. Como por ejemplo en 1958 se aplicaron los métodos de la programación lineal a un problema concreto: el cálculo del plan óptimo del transporte de arena de construcción a las obras de edificación de la ciudad de Moscú, donde existían 10 puntos de origen y 230 de destino. El resultado gracias al ordenador Strena tardó un periodo de 10 días, pero rebajó un 11% los gastos respecto a los costes previstos.

En el modelo de transporte hay r orígenes y s destinos, cada uno se representa por un nodo, como se ilustra en la figura 1. En dicha figura, las líneas representan los caminos que unen los orígenes con los destinos. La línea (i, j) que une el origen i con el destino j transporta una cantidad de producto que denotaremos por x_{ij} , con un coste de transporte asociado por unidad de c_{ij} (en ciertas unidades monetarias).

Figura 1: Esquema del problema básico de transporte. Elaboración propia



La cantidad ofertada en el origen i es o_i y la cantidad demandada en el destino j es d_j . El objetivo del modelo es minimizar el coste de transporte total al mismo tiempo que se satisfacen las restricciones de la oferta y la demanda.

En nuestro caso, resolvemos un problema específico de un modelo de transporte ambientado en la empresa Primark donde se puede decir que gran parte de los datos son reales y otros han sido elaborados atendiendo a determinados criterios de sentido común (véase el Apéndice 1.B).

1.2. - Planteamiento del Modelo Básico de Transporte

En nuestro caso el modelo de transporte con coste mínimo viene dado por:

$$(P) \quad \text{Min} \sum_{j=1}^s \sum_{i=1}^r c_{ij} x_{ij}$$

$$\text{s. a} \quad x_{i1} + x_{i2} + \dots + x_{is} = o_i, i = 1, \dots, r,$$

$$x_{1j} + x_{2j} + \dots + x_{rj} = d_j, j = 1, \dots, s,$$

$$x_{ij} \geq 0, i = 1, \dots, r, j = 1, \dots, s,$$

donde, recordemos que estamos empleando la siguiente notación:

- Cada $o_i, i = 1, \dots, r$, representa la cantidad ofertada o capacidad del i - ésimo nodo de origen.
- Cada $d_j, j = 1, \dots, s$, representa la cantidad demandada en el j - ésimo nodo de destino.

- $c_{ij}, i = 1, \dots, r, j = 1, \dots, s$, representa el coste de transportar una unidad del producto considerado desde el i -ésimo nodo de origen hasta el j -ésimo nodo de destino.
- $x_{ij}, i = 1, \dots, r, j = 1, \dots, s$, representa la cantidad de producto transportada desde el i -ésimo nodo de origen hasta el j -ésimo nodo de destino. Estas son las variables del modelo.

Nótese que el primer bloque de restricciones (asociado a los índices $i=1, \dots, r$) hace referencia a las cantidades ofertadas de cada uno de los orígenes. El segundo bloque ($j=1, \dots, s$) se corresponde con la satisfacción de la demanda en cada uno de los destinos. Finalmente el modelo recoge la condición de que todas las variables son no negativas, por tratarse de cantidades a transportar.

En nuestro modelo supondremos que la suma de las cantidades ofertadas coincide con las suma de las demandas, formalmente:

$$\sum_{i=1}^r o_i = \sum_{j=1}^s d_j.$$

El lector puede consultar modelos más generales en los que no se verifica dicha igualdad en Hillier y Lieberman (2010).

1.3. - Formato Matricial

El modelo de transporte en formato matricial quedaría:

$$(P) \quad \begin{aligned} & \text{Min } c'x \\ & \text{s. a } Aeq \ x = beq, \\ & \quad \quad x \geq 0, \end{aligned}$$

donde:

- x es el vector de variables del modelo:

$$x = (x_{11}, x_{12}, \dots, x_{1s}, x_{21}, \dots, x_{2s}, \dots, x_{r1}, \dots, x_{rs})' \in R^{r \times s},$$

- c representa al vector de coeficientes de la función objetivo:

$$c = (c_{11}, \dots, c_{1s}, c_{21}, \dots, c_{2s}, \dots, c_{r1}, \dots, c_{rs})' \in R^{r \times s},$$

- beq denota al vector de términos independientes del sistema de ecuaciones del modelo:

$$beq = (o_1, \dots, o_r, d_1, \dots, d_s)'$$

- Aeq es la matriz de coeficientes del sistema de ecuaciones del modelo (matriz del transporte):

$$Aeq = \left(\begin{array}{cccc|cccc| \dots & |cccc} 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 & 1 & \dots & 1 \\ \hline 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & 1 \end{array} \right)$$

En lo que sigue utilizamos la siguiente notación: 1_s representa al vector fila de dimensión s cuyas coordenadas son todas iguales a 1, esto es,

$$1_s = \underbrace{(1, \dots, 1)}_{(s)}$$

0_s denotará al vector fila de dimensión s cuyas coordenadas son todas iguales a 0, esto es,

$$0_s = \underbrace{(0, \dots, 0)}_{(s)}$$

Además, I_s denotará a la matriz identidad de orden s , esto es,

$$I_s = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 \end{pmatrix}$$

Tras la explicación de la notación podemos proceder a escribir la matriz Aeq :

$$Aeq = \underbrace{\left(\begin{array}{c} (r \text{ filas}) \\ I_s \end{array} \right)}_{(r \text{ bloques})} \begin{pmatrix} 1_s & 0_s & \dots & 0_s \\ 0_s & 1_s & \dots & 0_s \\ \dots & \dots & \dots & \dots \\ 0_s & 0_s & \dots & 1_s \\ I_s & I_s & \dots & I_s \end{pmatrix}$$

En general la matriz del transporte (A_{eq}) y su estructura dependen del número de orígenes y destinos. Esta matriz tiene $r + s$ filas y $r \times s$ columnas.

1.4. - Ejemplo Modelo de Transporte Simple

Imaginamos que la empresa FRIGOCON tiene 3 plantas en Alicante, Valencia y Castellón encargadas de la fabricación de frigoríficos. Sus centros de distribución se encuentran en Catarroja y Denia.

Las capacidades semanales de las 3 plantas son respectivamente:

P.Alicante	1000
P.Valencia	1300
P.Castellón	1400
Total	3700

Las demandas a la semana de los dos centros de distribución:

P.Catarroja	2100
P.Denia	1600
Total	3700

Las distancias en kilómetros entre las plantas y los centros de distribución son:

	Catarroja	Denia
Alicante	170	90
Valencia	15	105
Castellón	90	190

La compañía de transporte nos cobra 5 cent/km, así que, los costes de transporte por frigorífico entre cada dos nodos serían:

	Catarroja	Busot
Alicante	8,5	4,5
Valencia	0,75	5,25
Castellón	4,5	9,5

Sea x_{ij} el nº de frigoríficos enviados desde el origen i al destino j .

El problema quedaría:

$$\begin{aligned} (P) \quad & \text{Min } 8,5x_{11} + 4,5x_{12} + 0,75x_{21} + 5,25x_{22} + 4,5x_{31} + 9,5x_{32} \\ & \text{s. a } x_{11} + x_{12} = 1000 \\ & \quad x_{21} + x_{22} = 1300 \\ & \quad x_{31} + x_{32} = 1400 \\ & \quad x_{11} + x_{21} + x_{31} = 2100 \\ & \quad x_{12} + x_{22} + x_{32} = 1600 \\ & \quad x_{ij} \geq 0 \end{aligned}$$

En este caso las restricciones se pueden escribir con igualdad porque la suma de las ofertas es igual a la suma de demandas.

Tras la aplicación del programa Matlab para dicho problema con coste mínimo obtenemos las siguientes soluciones óptimas:

$$\begin{aligned} x_{11} &= 0 \\ x_{12} &= 1000 \\ x_{21} &= 700 \\ x_{22} &= 600 \\ x_{31} &= 1400 \\ x_{32} &= 0 \end{aligned}$$

Coste mínimo conjunto=14475€

Dichos datos nos dicen como conclusión:

- Del origen 1 (Alicante) tiene que ir al destino 1 (Catarroja) la cantidad 0 y al destino 2 (Denia) 1000 unidades de frigoríficos.
- Del origen 2 (Valencia) tiene que ir al destino 1 (Catarroja) la cantidad 700 y al destino 2 (Denia) 600 unidades.
- Del origen 3 (Castellón) tiene que ir al destino 1 (Catarroja) la cantidad 1400 y al destino 2 (Denia) 0 unidades.

Todas estas cantidades distribuidas desde los orígenes a sus respectivos destinos son para conseguir obtener un coste mínimo para la empresa de 14475€

1.5. - Resolución del Modelo en Matlab

En la introducción hacemos referencia a que los problemas planteados se van a desarrollar y analizar con el fin de ser resueltos a través de programas informáticos desarrollados para la resolución de este tipo de problema, en nuestro caso vamos a utilizar el software Matlab.

El modelo general de Programación Lineal en Matlab presenta la siguiente forma matricial:

$$(P) \quad \begin{aligned} & \text{Min } c'x \\ & \text{s. a } Ax \leq b, \\ & \quad \quad Aeqx = beq, \\ & \quad \quad LB \leq x \leq UB, \\ & \quad \quad x \geq 0, \end{aligned}$$

donde $x \in R^n$ es el vector de variables de decisión, $c \in R^n$ es el vector de coeficientes de la función objetivo, ' $Ax \leq b$ ' y ' $Aeqx = beq$ ' representan el sistema de desigualdades lineales y de ecuaciones lineales en formato matricial, respectivamente, y para terminar LB (*lowerbound*) y UB (*upperbound*) hacen referencia a las cotas inferiores y las cotas superiores de las variables del problema.

Para la ejecución de este problema se utilizara la función de Matlab "*linprog*" en su forma más desarrollada dependiendo inicialmente de las restricciones y datos que tengamos.

Forma más simplificada $\longrightarrow x = \text{linprog}(c, A, b)$

Se utiliza cuando el sistema de restricciones contiene solamente desigualdades lineales.

Forma más desarrollada $\longrightarrow x = \text{linprog}(c, A, b, Aeq, beq, LB, UB)$

Dependiendo del tipo de restricciones que presente el problema tendremos que añadir más o menos variables a dicha fórmula. En caso de que el problema no presente cotas inferiores o superiores lo dejaremos como una matriz vacía, esto es, $LB = []$ o $UB = []$.

También es posible añadir otra opción de salida a la función *linprog*. Aparte de la solución óptima, nos puede calcular el coste mínimo del problema añadiendo *fval* como se muestra a continuación:

$$[x, fval] = \text{linprog}(c, A, b, Aeq, beq, LB, UB).$$

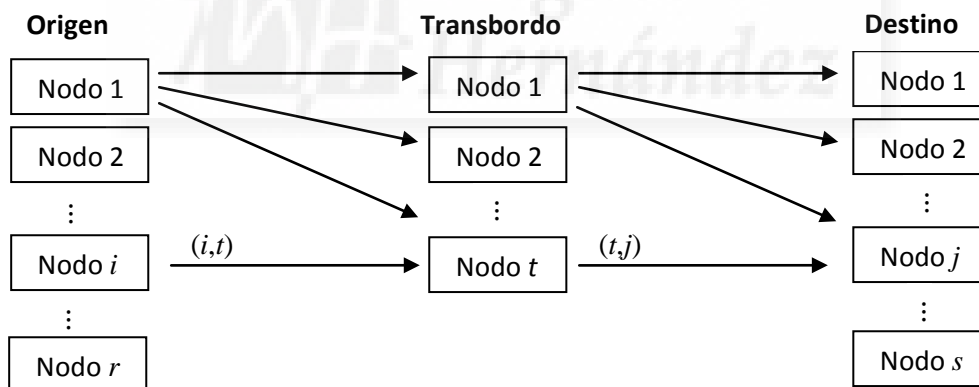
1.6. – Modelo de Transporte con Transbordo

La aplicación del modelo de transbordo trata de variar el modelo original del transporte añadiendo una serie de nodos intermedios que facilitan el intercambio entre los distintos nodos orígenes y destinos.

En el modelo supondremos que hay r orígenes y s destinos, y que aparecen t nodos de transbordo, que sirven de intermediarios entre orígenes y destinos (como se ilustra en la figura 2). En la figura 2, las líneas representan los caminos que unen los orígenes con los transbordos y a su vez los transbordos con los distintos destinos.

La línea (i, t) que une el origen i con el transbordo t transporta una cantidad de producto desde el origen i al transbordo t que denotaremos por x_{it} y la línea (t, j) que se dirige desde el transbordo t al destino j transporta la cantidad y_{tj} , todo ello lleva asociado un coste de transporte por unidad de c_{it} y c_{tj} (en ciertas unidades monetarias).

Figura 2: Esquema del problema de transporte con transbordo. Elaboración propia.



1.6.1. - Ejemplo de Modelo de Transporte con Transbordo

La empresa EDITOELX se dedica a la impresión de libros de lectura, tiene 3 plantas de fabricación en Elche, Mallorca y Pamplona; los almacenes de dicha empresa se encuentran en Valencia y Madrid; y sus destinos finales son Sevilla, Barcelona, Andorra y Oporto.

Supongamos que las cantidades semanales de dichas plantas son respectivamente:

P.Elche	P.Mallorca	P.Pamplona	Total
6500	3000	700	16500

Supondremos que las cantidades demandadas para dichos destinos son las siguientes:

Sevilla	Barcelona	Andorra	Oporto	Total
1500	5000	2500	7500	16500

El coste unitario de transporte por libro independientemente de su género sería:

Tabla 1: Origen-Transbordo. Elaboración propia

Origen-Transbordo

	Valencia	Madrid
Elche	0,1	0,75
Mallorca	0,9	1,2
Pamplona	0,5	0,4

Tabla 2: Transbordo-Destino. Elaboración propia

Transbordo-Destino

	Sevilla	Barcelona	Andorra	Oporto
Valencia	0,25	0,5	0,6	0,3
Madrid	0,4	0,3	0,7	0,15

Sea x_{it} el nº de libros enviados desde el origen i al transbordo t .

Sea y_{tj} el nº de libros enviados desde el transbordo t al destino j .

El objetivo es:

$$\begin{aligned}
 (P) \quad & \text{Min } 0,1x_{11} + 0,75x_{12} + 0,9x_{21} + 1,2x_{22} + 0,5x_{31} + 0,4x_{32} + 0,25y_{11} + \\
 & 0,5y_{12} + 0,6y_{13} + 0,3y_{14} + 0,4y_{21} + 0,3y_{22} + 0,7y_{23} + 0,15y_{24} \\
 \text{s. a} \quad & x_{11} + x_{12} = 6500 \\
 & x_{21} + x_{22} = 3000 \\
 & x_{31} + x_{32} = 7000 \\
 & y_{11} + y_{21} = 1500 \\
 & y_{12} + y_{22} = 5000 \\
 & y_{13} + y_{23} = 2500 \\
 & y_{14} + y_{24} = 7500 \\
 & x_{11} + x_{21} + x_{31} = y_{11} + y_{12} + y_{13} + y_{14} \\
 & x_{12} + x_{22} + x_{32} = y_{21} + y_{22} + y_{23} + y_{24} \\
 & x_{it} \geq 0 \\
 & y_{tj} \geq 0
 \end{aligned}$$

Tras la aplicación del programa Matlab para dicho problema con coste mínimo obtenemos las siguientes soluciones óptimas:

	Valencia	Madrid
Elche	6500	0
Mallorca	3000	0
Pamplona	0	7000

	Sevilla	Barcelona	Andorra	Oporto
Valencia	1500	0	2500	5500
Madrid	0	5000	0	2000

Coste mínimo conjunto=11475€

Dichos datos nos dicen como conclusión:

- Del origen 1 (Elche) tiene que ir al transbordo 1 (Valencia) la cantidad 6500 y al transbordo 2 (Madrid) 0 unidades.
- Del origen 2 (Mallorca) tiene que ir al transbordo 1 (Valencia) la cantidad 3000 y al transbordo 2 (Madrid) 0 unidades.
- Del origen 3 (Pamplona) tiene que ir al transbordo 1 (Valencia) la cantidad 0 y al transbordo 2 (Madrid) 7000 unidades.
- Del transbordo 1 (Valencia) tiene que ir al destino 1 (Sevilla) la cantidad 1500 y al destino 2 (Barcelona) 0, al destino 3 (Andorra) 2500 y al destino 4 (Oporto) 5500 unidades.
- Del transbordo 2 (Madrid) tiene que ir al destino 1 (Sevilla) la cantidad 0 y al destino 2 (Barcelona) 5000, al destino 3 (Andorra) 0 y al destino 4 (Oporto) 2000 unidades.

Todas estas cantidades distribuidas desde los orígenes a sus transbordos y de sus transbordos a sus destinos son para conseguir obtener un coste mínimo para la empresa de 11475€

1.7. – Resolución de un caso real: Un modelo de transporte en la empresa Primark

El ejemplo real que queremos desarrollar para dar solución al problema de transporte con transbordo corresponde a un modelo inspirado en la compañía irlandesa Primark.

Nuestro negocio tiene 4 plantas de fabricación (orígenes) en los siguientes países: China, Bangladesh, India, Pakistan. Para ello estas deben de satisfacer la demanda de sus respectivos centros de logística que se encuentran repartidos en Reino Unido, Alemania, España, Irlanda, Países Bajos, Portugal, Francia, Austria, Belgica (destinos) todo esto realizado por transporte aéreo.

A su vez los centros logísticos serán los encargados de abastecer a las diferentes tiendas más importantes del país para así satisfacer la demanda de los clientes; dicho transporte se realizará por carretera. Toda la operativa relacionada con el transporte de la mercancía desde sus orígenes a los destinos se realizara a través de la empresa DHL.

Seguidamente, presentamos los datos relativos a las capacidades mensuales de las plantas de producción. Concretamente, la primera columna indica las ciudades consideradas, la segunda contiene las cantidades de producción (en kilogramos), y la tercera corresponde al número de palés, teniendo en cuenta que cada uno pesa aproximadamente 350 kilogramos.

P.Shanghái	98000	(280 P)
P.Daca	63000	(180 P)
P.Bombay	73500	(210 P)
P.Karachi	49000	(140 P)
Total	283500	

Supongamos, además, que las demandas de los destinos (tiendas) son las siguientes:

Berlín	24500	(70 P)
Dortmund	8750	(25 P)
Frankfurt	14000	(40 P)
Stuttgart	10500	(30 P)
Bremen	5250	(15 P)
Manchester	7000	(20 P)
Liverpool	10500	(30 P)
Londres	21000	(60 P)
Newcastle	3500	(10 P)
Glasgow	8750	(25 P)

Barcelona	12250	(35 P)
Bilbao	5250	(15 P)
Madrid	17500	(50 P)
Elche	3500	(10 P)
Málaga	7000	(20 P)
Valladolid	3500	(10 P)
Belfast	3500	(10 P)
Cork	1750	(5 P)
Dublín	5250	(15 P)
Galway	1750	(5 P)
Lodonderry	1750	(5 P)
Eindhoven	1750	(5 P)
Rotterdam	7000	(20 P)
Ámsterdam	8750	(25 P)
Arnhem	1750	(5 P)
Den Haag	3500	(10 P)
Coimbra	1750	(5 P)
Lisboa	7000	(20 P)
Oporto	3500	(10 P)
Braga	1750	(5 P)
Marsella	8750	(25 P)
París	14000	(40 P)
Dijon	3500	(10 P)
Lyon	7000	(20 P)
Montpellier	7000	(20 P)
Toulouse	5250	(15 P)
Innsbruck	1750	(5 P)
Wien	5250	(15 P)
Linz	1750	(5 P)
Graz	3500	(10 P)
Lieja	1750	(5 P)
Bruselas	7000	(20 P)
Anderlecht	1750	(5 P)
Genk	1750	(5 P)
Total	283500	



Por completitud, y a efecto orientativos, seguidamente incluimos las distancias en kilómetros entre las plantas y los centros logísticos:

Tabla 3: Distancias orígenes-transbordos. Elaboración propia

	P.Shanghái	P.Daca	P.Bombay	P.Karachi
Frankfurt	8819	7433	6566	5679
Hannover	8611	7322	6530	5643
Londres	9722	8003	7194	6307
Edinburgh	9000	8059	7401	6518

Madrid	10254	8654	7533	6665
Valencia	10126	8424	7261	6398
Dublín	9346	8337	7606	6720
Limerick	9508	8512	7770	6883
Ámsterdam	8873	7646	6856	5969
Utrecht	8880	7636	6837	5950
Lisboa	10709	9157	8022	7157
Braga	10431	8973	7908	7035
París	9262	7910	7012	6125
Nantes	9599	8240	7303	6419
Viena	8475	6906	5980	5093
Salzburgo	8707	7158	6214	5328
Brujas	9044	7786	6961	6073
Bruselas	9009	7718	6879	5992

La ruta en kilómetros entre los distintos nodos de origen y transbordos son obtenidas de la siguiente página: <http://es.distance.to/>

Para que el problema fuera con aplicación real solicitamos a DHL datos orientativos para hacer posible nuestro estudio. Concretamente, la empresa DHL nos facilitó una estimación del coste de transportar un palé entre dos de las ciudades consideradas. A partir de esa información, hemos completado la tabla que aparece a continuación siguiendo como criterio principal la proporcionalidad con la distancia entre cada dos nodos.

Tabla 4: Costes unitarios de transporte. Elaboración propia

Costes unitarios de transporte

	P.Shanghái	P.Daca	P.Bombay	P.Karachi
Frankfurt	800	650	400	500
Hannover	900	600	500	600
Londres	1350	1000	900	1100
Edinburgh	1200	1050	950	1000
Madrid	1400	1200	1100	950
Valencia	1500	1100	1000	900
Dublín	1000	800	950	700
Limerick	1150	1000	1000	850
Ámsterdam	950	1100	600	550
Utrecht	1000	950	700	600
Lisboa	1300	900	1000	1150
Braga	1200	700	900	1100
París	1250	1300	1200	900

Nantes	1400	1100	1300	1050
Viena	900	950	1050	1200
Salzburgo	1100	800	1100	900
Brujas	600	700	850	400
Bruselas	700	600	800	500

A continuación, les presentamos los costes unitarios de transportar un palé por carretera realizados entre el centro logístico (transbordo) y las distintas tiendas (destino) a las que va dirigido el producto demandado.

	Berlín	Dortmund	Frankfurt	Stuttgart	Bremen
Frankfurt	275	110	20	100	225
Hannover	140	105	170	255	60

	Manchester	Liverpool	Londres	Newcastle	Glasgow
Londres	170	180	20	225	330
Edinburgh	175	180	330	95	40

	Barcelona	Bilbao	Madrid	Elche	Málaga	Valladolid
Madrid	310	200	10	205	270	95
Valencia	175	300	180	85	310	270

	Belfast	Cork	Dublín	Galway	Lodonderry
Dublín	80	130	15	100	120
Limerick	180	50	100	50	180

	Eindhoven	Rotterdam	Ámsterdam	Arnhem	Den Haag
Ámsterdam	60	35	10	50	30
Utrecht	45	30	20	30	30

	Coímbra	Lisboa	Oporto	Braga
Lisboa	100	15	160	180
Braga	85	185	30	10

	Marsella	París	Dijon	Lyon	Montpellier	Toulouse
París	390	20	155	185	370	340
Nantes	480	190	320	340	410	290

	Innsbruck	Wien	Linz	Graz
Viena	220	40	90	100
Salzburgo	90	150	70	140

	Lieja	Bruselas	Anderlecht	Genk
Brujas	100	50	50	100
Bruselas	50	20	30	50

En lo que sigue, por claridad en la notación, emplearemos subíndices y comas para distinguir transbordos y destinos. Concretamente, denotaremos por $y_{t,j}$ al nº de palés transportados desde el transbordo t al destino j .

El objetivo es:

$$\begin{aligned}
(P) \quad \text{Min} \quad & 800x_{1,1} + 900x_{1,2} + 1350x_{1,3} + 1200x_{1,4} + 1400x_{1,5} + 1500x_{1,6} + \\
& 1000x_{1,7} + 1150x_{1,8} + 950x_{1,9} + 1000x_{1,10} + 1300x_{1,11} + 1200x_{1,12} + \\
& 1250x_{1,13} + 1400x_{1,14} + 900x_{1,15} + 1100x_{1,16} + 600x_{1,17} + 700x_{1,18} + \\
& 650x_{2,1} + 600x_{2,2} + 1000x_{2,3} + 1050x_{2,4} + 1200x_{2,5} + 1100x_{2,6} + \\
& 800x_{2,7} + 1000x_{2,8} + 1100x_{2,9} + 950x_{2,10} + 900x_{2,11} + 700x_{2,12} + \\
& 1300x_{2,13} + 1100x_{2,14} + 950x_{2,15} + 800x_{2,16} + 700x_{2,17} + 600x_{2,18} + \\
& 400x_{3,1} + 500x_{3,2} + 900x_{3,3} + 950x_{3,4} + 1100x_{3,5} + 1000x_{3,6} + \\
& 950x_{3,7} + 1000x_{3,8} + 600x_{3,9} + 700x_{3,10} + 1000x_{3,11} + 900x_{3,12} + \\
& 1200x_{3,13} + 1300x_{3,14} + 1050x_{3,15} + 1100x_{3,16} + 850x_{3,17} + \\
& 800x_{3,18} + 500x_{4,1} + 600x_{4,2} + 1100x_{4,3} + 1000x_{4,4} + 950x_{4,5} + \\
& 900x_{4,6} + 700x_{4,7} + 850x_{4,8} + 550x_{4,9} + 600x_{4,10} + 1150x_{4,11} + \\
& 1100x_{4,12} + 900x_{4,13} + 1050x_{4,14} + 1200x_{4,15} + 900x_{4,16} + 400x_{4,17} + \\
& 500x_{4,18} + \\
& 275y_{1,1} + 110y_{1,2} + 20y_{1,3} + 100y_{1,4} + 225y_{1,5} + 140y_{2,1} + 105y_{2,2} + \\
& 170y_{2,3} + 255y_{2,4} + 60y_{2,5} + 170y_{3,6} + 180y_{3,7} + 20y_{3,8} + 225y_{3,9} + \\
& 330y_{3,10} + 175y_{4,6} + 180y_{4,7} + 330y_{4,8} + 95y_{4,9} + 40y_{4,10} + 310y_{5,11} + \\
& 200y_{5,12} + 10y_{5,13} + 205y_{5,14} + 270y_{5,15} + 95y_{5,16} + 175y_{6,11} + \\
& 300y_{6,12} + 180y_{6,13} + 85y_{6,14} + 310y_{6,15} + 270y_{6,16} + 80y_{7,17} + \\
& 130y_{7,18} + 15y_{7,19} + 100y_{7,20} + 120y_{7,21} + 180y_{8,17} + 50y_{8,18} + \\
& 100y_{8,19} + 50y_{8,20} + 180y_{8,21} + 60y_{9,22} + 35y_{9,23} + 10y_{9,24} + 50y_{9,25} + \\
& 30y_{9,26} + 45y_{10,22} + 30y_{10,23} + 20y_{10,24} + 30y_{10,25} + 30y_{10,26} + \\
& 100y_{11,27} + 15y_{11,28} + 160y_{11,29} + 180y_{11,30} + 85y_{12,27} + 185y_{12,28} + \\
& 30y_{12,29} + 10y_{12,30} + 390y_{13,31} + 20y_{13,32} + 155y_{13,33} + 185y_{13,34} + \\
& 370y_{13,35} + 340y_{13,36} + 480y_{14,31} + 190y_{14,32} + 320y_{14,33} + \\
& 340y_{14,34} + 410y_{14,35} + 290y_{14,36} + 220y_{15,37} + 40y_{15,38} + 90y_{15,39} + \\
& 100y_{15,40} + 90y_{16,37} + 150y_{16,38} + 70y_{16,39} + 140y_{16,40} + 100y_{17,41} + \\
& 50y_{17,42} + 50y_{17,43} + 100y_{17,44} + 50y_{18,41} + 20y_{18,42} + 30y_{18,43} + \\
& 50y_{18,44} +
\end{aligned}$$

$$\begin{aligned}
s. a \quad & x_{1,1} + x_{1,2} + x_{1,3} + x_{1,4} + x_{1,5} + x_{1,6} + x_{1,7} + x_{1,8} + x_{1,9} + x_{1,10} + x_{1,11} + \\
& x_{1,12} + x_{1,13} + x_{1,14} + x_{1,15} + x_{1,16} + x_{1,17} + x_{1,18} = 98000, \\
& x_{2,1} + x_{2,2} + x_{2,3} + x_{2,4} + x_{2,5} + x_{2,6} + x_{2,7} + x_{2,8} + x_{2,9} + x_{2,10} + x_{2,11} + \\
& x_{2,12} + x_{2,13} + x_{2,14} + x_{2,15} + x_{2,16} + x_{2,17} + x_{2,18} = 63000, \\
& x_{3,1} + x_{3,2} + x_{3,3} + x_{3,4} + x_{3,5} + x_{3,6} + x_{3,7} + x_{3,8} + x_{3,9} + x_{3,10} + x_{3,11} + \\
& x_{3,12} + x_{3,13} + x_{3,14} + x_{3,15} + x_{3,16} + x_{3,17} + x_{3,18} = 73500,
\end{aligned}$$

$$x_{4,1} + x_{4,2} + x_{4,3} + x_{4,4} + x_{4,5} + x_{4,6} + x_{4,7} + x_{4,8} + x_{4,9} + x_{4,10} + x_{4,11} + x_{4,12} + x_{4,13} + x_{4,14} + x_{4,15} + x_{4,16} + x_{4,17} + x_{4,18} = 49000,$$

$$\begin{aligned} y_{1,1} + y_{2,1} &= 24500, y_{1,2} + y_{2,2} = 8750, y_{1,3} + y_{2,3} = 14000, \\ y_{1,4} + y_{2,4} &= 10500, y_{1,5} + y_{2,5} = 5250, y_{3,6} + y_{4,6} = 7000, \\ y_{3,7} + y_{4,7} &= 10500, y_{3,8} + y_{4,8} = 21000, y_{3,9} + y_{4,9} = 3500, \\ y_{3,10} + y_{4,10} &= 8750, y_{5,11} + y_{6,11} = 12250, y_{5,12} + y_{6,12} = 5250, \\ y_{5,13} + y_{6,13} &= 17500, y_{5,14} + y_{6,14} = 3500, y_{5,15} + y_{6,15} = 7000, \\ y_{5,16} + y_{6,16} &= 3500, y_{7,17} + y_{8,17} = 3500, y_{7,18} + y_{8,18} = 1750, \\ y_{7,19} + y_{8,19} &= 5250, y_{7,20} + y_{8,20} = 1750, y_{7,21} + y_{8,21} = 1750, \\ y_{9,22} + y_{10,22} &= 1750, y_{9,23} + y_{10,23} = 7000, y_{9,24} + y_{10,24} = 8750, \\ y_{9,25} + y_{10,25} &= 1750, y_{9,26} + y_{10,26} = 3500, y_{11,27} + y_{12,27} = 1750, \\ y_{11,28} + y_{12,28} &= 7000, y_{11,29} + y_{12,29} = 3500, y_{11,30} + y_{12,30} = 1750, \\ y_{13,31} + y_{14,31} &= 8750, y_{13,32} + y_{14,32} = 14000, y_{13,33} + y_{14,33} = 3500, \\ y_{13,34} + y_{14,34} &= 7000, y_{13,35} + y_{14,35} = 7000, y_{13,36} + y_{14,36} = 5250, \\ y_{15,37} + y_{16,37} &= 1750, y_{15,38} + y_{16,38} = 5250, y_{15,39} + y_{16,39} = 1750, \\ y_{15,40} + y_{16,40} &= 3500, y_{17,41} + y_{18,41} = 1750, y_{17,42} + y_{18,42} = 7000, \\ y_{17,43} + y_{18,43} &= 1750, y_{17,44} + y_{18,44} = 1750 \end{aligned}$$

$$\begin{aligned} x_{1,1} + x_{2,1} + x_{3,1} + x_{4,1} &= y_{1,1} + y_{1,2} + y_{1,3} + y_{1,4} + y_{1,5} \\ x_{1,2} + x_{2,2} + x_{3,2} + x_{4,2} &= y_{2,1} + y_{2,2} + y_{2,3} + y_{2,4} + y_{2,5} \\ x_{1,3} + x_{2,3} + x_{3,3} + x_{4,3} &= y_{3,6} + y_{3,7} + y_{3,8} + y_{3,9} + y_{3,10} \\ x_{1,4} + x_{2,4} + x_{3,4} + x_{4,4} &= y_{4,6} + y_{4,7} + y_{4,8} + y_{4,9} + y_{4,10} \\ x_{1,5} + x_{2,5} + x_{3,5} + x_{4,5} &= y_{5,11} + y_{5,12} + y_{5,13} + y_{5,14} + y_{5,15} + y_{5,16} \\ x_{1,6} + x_{2,6} + x_{3,6} + x_{4,6} &= y_{6,11} + y_{6,12} + y_{6,13} + y_{6,14} + y_{6,15} + y_{6,16} \\ x_{1,7} + x_{2,7} + x_{3,7} + x_{4,7} &= y_{7,17} + y_{7,18} + y_{7,19} + y_{7,20} + y_{7,21} \\ x_{1,8} + x_{2,8} + x_{3,8} + x_{4,8} &= y_{8,17} + y_{8,18} + y_{8,19} + y_{8,20} + y_{8,21} \\ x_{1,9} + x_{2,9} + x_{3,9} + x_{4,9} &= y_{9,22} + y_{9,23} + y_{9,24} + y_{9,25} + y_{9,26} \\ x_{1,10} + x_{2,10} + x_{3,10} + x_{4,10} &= y_{10,22} + y_{10,23} + y_{10,24} + y_{10,25} + y_{10,26} \\ x_{1,11} + x_{2,11} + x_{3,11} + x_{4,11} &= y_{11,27} + y_{11,28} + y_{11,29} + y_{11,30} \\ x_{1,12} + x_{2,12} + x_{3,12} + x_{4,12} &= y_{12,27} + y_{12,28} + y_{12,29} + y_{12,30} \\ x_{1,13} + x_{2,13} + x_{3,13} + x_{4,13} &= y_{13,31} + y_{13,32} + y_{13,33} + y_{13,34} + y_{13,35} + \\ & y_{13,36} \\ x_{1,14} + x_{2,14} + x_{3,14} + x_{4,14} &= y_{14,31} + y_{14,32} + y_{14,33} + y_{14,34} + y_{14,35} + \\ & y_{14,36} \\ x_{1,15} + x_{2,15} + x_{3,15} + x_{4,15} &= y_{15,37} + y_{15,38} + y_{15,39} + y_{15,40} \\ x_{1,16} + x_{2,16} + x_{3,16} + x_{4,16} &= y_{16,37} + y_{16,38} + y_{16,39} + y_{16,40} \\ x_{1,17} + x_{2,17} + x_{3,17} + x_{4,17} &= y_{17,41} + y_{17,42} + y_{17,43} + y_{17,44} \\ x_{1,18} + x_{2,18} + x_{3,18} + x_{4,18} &= y_{18,41} + y_{18,42} + y_{18,43} + y_{18,44} \\ x_{it} &\geq 0 \\ y_{tj} &\geq 0 \end{aligned}$$

Tras la aplicación del programa Matlab para dicho problema con coste mínimo nos salen las siguientes soluciones óptimas.

La siguiente tabla muestra las cantidades a transportar desde cada uno de los orígenes a sus transbordos:

	O1: P.Shanghái	O2: P.Daca	O3: P.Bombay	O4: P.Karachi
T1: Frankfurt	0	0	33250	0
T2: Hannover	0	20600	9150	0
T3: Londres	0	22358	8350	0
T4: Edinburgh	20042	0	0	0
T5: Madrid	0	0	0	26250
T6: Valencia	0	0	0	22750
T7: Dublín	12292	1708	0	0
T8: Limerick	0	0	0	0
T9: Ámsterdam	0	0	22750	0
T10: Utrecht	0	0	0	0
T11: Lisboa	0	0	0	0
T12: Braga	0	14000	0	0
T13: París	42916	0	0	0
T14: Nantes	0	2584	0	0
T15: Viena	10500	0	0	0
T16: Salzburgo	0	1750	0	0
T17: Brujas	12250	0	0	0
T18: Bruselas	0	0	0	0

Las tablas que aparecen a continuación representan las cantidades a transportar desde cada uno de los transbordos a sus respectivos destinos:

	D1: Berlín	D2: Dortmund	D3: Frankfurt	D4: Stuttgart	D5: Bremen
T1: Frankfurt	0	8750	14000	10500	0
T2: Hannover	24500	0	0	0	5250

	D6: Manchester	D7: Liverpool	D8: Londres	D9: Newcastle	D10: Glasgow
T3: Londres	7000	2708	21000	0	0
T4: Edinburgh	0	7792	0	3500	8750

	D11: Barcelona	D12: Bilbao	D13: Madrid	D14: Elche	D15: Málaga	D16: Valladolid
T5: Madrid	0	5250	17500	0	0	3500
T6: Valencia	12250	0	0	3500	7000	0

	D17: Belfast	D18: Cork	D19: Dublín	D20: Galway	D21: Londonderry
T7: Dublín	3500	1750	5250	1750	1750
T8: Limerick	0	0	0	0	0

	D22: Eindhoven	D23: Rotterdam	D24: Ámsterdam	D25: Arnhem	D26: Den Haag
T9: Ámsterdam	1750	7000	8750	1750	3500
T10: Utrecht	0	0	0	0	0

	D27: Coímbra	D28: Lisboa	D29: Oporto	D30: Braga
T11: Lisboa	0	0	0	0
T12: Braga	1750	7000	3500	1750

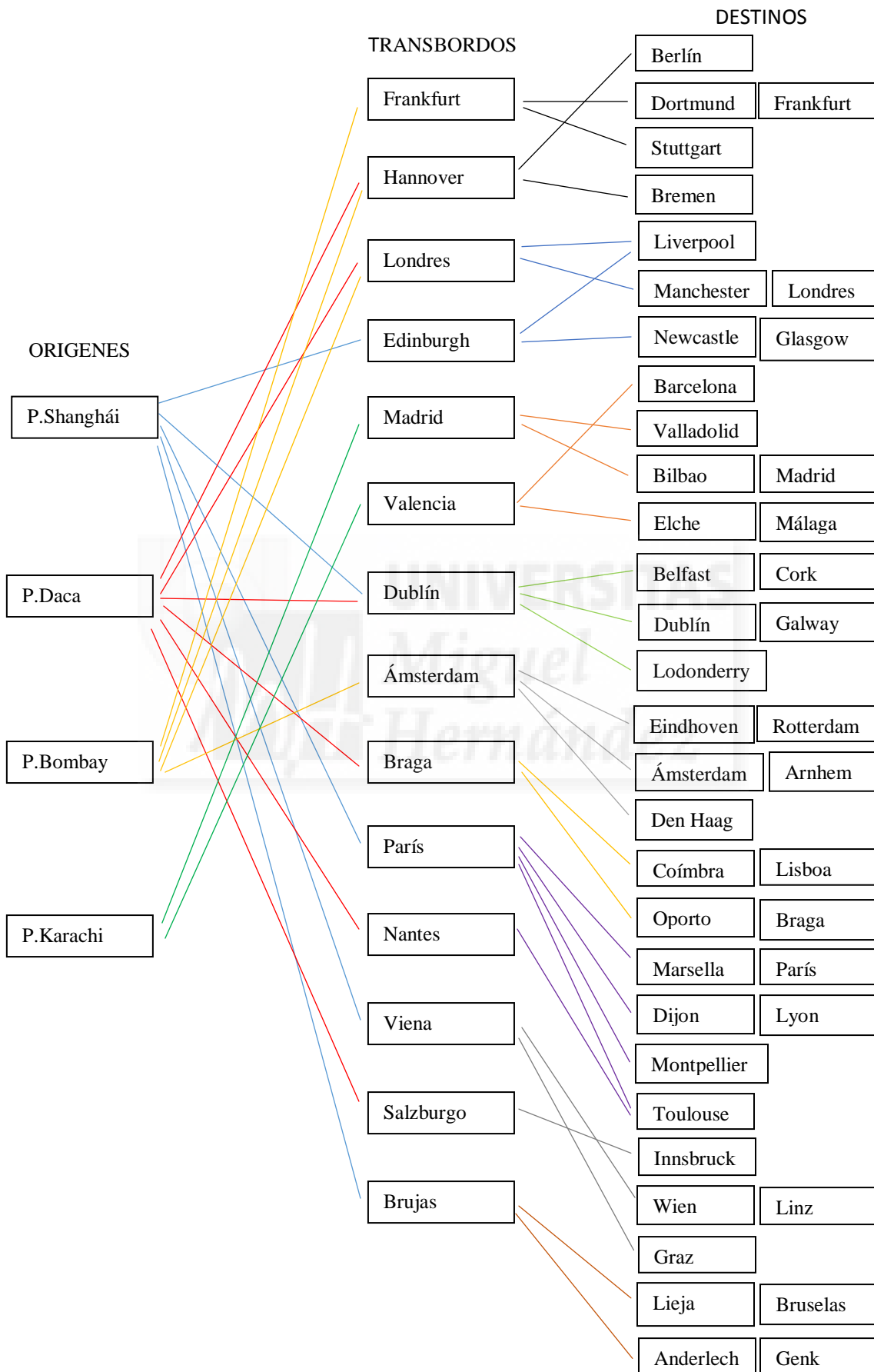
	D31: Marsella	D32: París	D33: Dijon	D34: Lyon	D35: Montpellier	D36: Toulouse
T13: París	8750	14000	3500	7000	7000	2666
T14: Nantes	0	0	0	0	0	2584

	D37: Innsbruck	D38: Wien	D39: Linz	D40: Graz
T15: Viena	0	5250	1750	3500
T16: Salzburgo	1750	0	0	0

	D41: Lieja	D42: Bruselas	D43: Anderlecht	D44: Genk
T17: Brujas	1750	7000	1750	1750
T18: Bruselas	0	0	0	0

Todas estas cantidades distribuidas desde los orígenes a sus transbordos y de sus transbordos a sus destinos son para conseguir obtener un coste mínimo para la empresa de 272.360.000€

Seguidamente incluimos un esquema con de las diferentes rutas en las cuales se transporta mercancía atendiendo a la solución óptima del modelo de transporte.



APÉNDICE 1.A: PROGRAMACIÓN EN MATLAB (Modelo de Transporte General)

```
%Este programa resuelve de manera genérica el problema de transporte
%con transbordo. Suponemos que existen o orígenes, d destinos y t
%transbordos. Las variables serán consideradas en el orden siguiente:
%En primer lugar, se colocan las variables  $x_{it}$  (cantidad transportada
%desde el  $i$ -ésimo origen hasta el  $t$ -ésimo transbordo) y a continuación
%la variables  $y_{tj}$  (cantidad transportada desde el  $t$ -ésimo transbordo
%hasta el destino  $j$ ). Nótese que las coordenadas del vector de costes
%han de ordenarse atendiendo a la posición de las variables
c=input('Introduzca el vector de costes (vector columna)');
A=[];
b=[];
b1=input('Introduzca las ofertas de los orígenes (columna)');
b2=input('Introduzca las demandas de los destinos (columna)');
b3=input('Número de transbordos');
s=size(c);
nvariable=s(1);
s1=size(b1);
s2=size(b2);
o=s1(1);
d=s2(1);
t=b3;
X=[];
%Elaboración de la matriz de coeficientes del sistema de restricciones
%correspondiente a los diferentes nodos de origen
H=[];
for i=1:o
    z=zeros(o,t);
    z(i,:)=ones(1,t);
    H=[H,z]
end
for i=1:o
    z=zeros(o,t*d);
    X=[H,z];
end
%Elaboración de la matriz de coeficientes del sistema de restricciones
%correspondiente a los diferentes nodos de destino
for i=1:d
    z=zeros(d,o*t);
end
D=[];
for i=1:t
    D=[D,eye(d)];
end
W=[z,D]
%Elaboración de la matriz de coeficientes del sistema de restricciones
```

%correspondiente a los diferentes nodos de transbordo

```
Y1=[]
for i=1:o
    Y1=[Y1,eye(t)]
end
Y2=[]
for i=1:t
    z=zeros(t,d)
    z(i,:)=(-1)*ones(1,d)
    Y2=[Y2,z]
end
Y=[Y1,Y2]
Aeq=[X;W;Y];
beq=[b1;b2;zeros(b3,1)]';
lb=zeros(1,nvariable);
ub=[];
[Solucion,coste]=linprog(c,A,b,Aeq,beq,lb,ub)
```

APÉNDICE 1.B: CASO PARTICULAR DEL PRIMARK CON MATLAB

La introducción de la constante M con una cantidad muy superior en referencia a los datos del problema se realiza para que en algunos trayectos los costes salgan altos y entonces no los tengamos en cuenta como una solución óptima del problema. Este dato M, que es introducido por el usuario de manera manual se hace tomando como referencia los datos del problema inicial, esto se pone para obligar a que las variables que acompañan a dicha constante sean 0 para así cumplir las restricciones previamente dadas del problema.

Para resolver dicho problema en particular tras la introducción de los datos del Primark debemos de ejecutarlos en el Matlab y posteriormente ejecutar el programa general para conseguir la solución del problema de manera óptima con coste mínimo.

```
M=1000000;
o=[98000 63000 73500 49000 ]';
d=[24500 8750 14000 10500 5250 7000 10500 21000 3500 8750 12250 5250 17500,...
    3500 7000 3500 3500 1750 5250 1750 1750 1750 7000 8750 1750 3500 1750,...
    7000 3500 1750 8750 14000 3500 7000 7000 5250 1750 5250 1750 3500,...
    1750 7000 1750 1750]';
c =[800 900 1350 1200 1400 1500 1000 1150 950 1000 1300 1200 1250 1400 900 1100
600 700,...
    650 600 1000 1050 1200 1100 800 1000 1100 950 900 700 1300 1100 950 800 700
600,...
    400 500 900 950 1100 1000 950 1000 600 700 1000 900 1200 1300 1050 1100 850
800,...
```


500 600 1100 1000 950 900 700 850 550 600 1150 1100 900 1050 1200 900 400
 500,...
 275 110 20 100 225 M*ones(1,39),...
 140 105 170 255 60 M*ones(1,39),...
 M M M M M 170 180 20 225 330 M*ones(1,34),...
 M M M M M 175 180 330 95 40 M*ones(1,34),...
 M*ones(1,10) 310 200 10 205 270 95 M*ones(1,28),...
 M*ones(1,10) 175 300 180 85 310 270 M*ones(1,28),...
 M*ones(1,16) 80 130 15 100 120 M*ones(1,23),...
 M*ones(1,16) 180 50 100 50 180 M*ones(1,23),...
 M*ones(1,21) 60 35 10 50 30 M*ones(1,18),...
 M*ones(1,21) 45 30 20 30 30 M*ones(1,18),...
 M*ones(1,26) 100 15 160 180 M*ones(1,14),...
 M*ones(1,26) 85 185 30 10 M*ones(1,14),...
 M*ones(1,30) 390 20 155 185 370 340 M*ones(1,8),...
 M*ones(1,30) 480 190 320 340 410 290 M*ones(1,8),...
 M*ones(1,36) 220 40 90 100 M M M M,...
 M*ones(1,36) 90 150 70 140 M M M M,...
 M*ones(1,40) 100 50 50 100,...
 M*ones(1,40) 50 20 30 50]'



CAPÍTULO 2. - MODELO DEL VIAJANTE DE COMERCIO

2.1. - Introducción

En el presente capítulo vamos a examinar el modelo del viajante de comercio o también llamado agente viajero (en inglés, *Travelling Salesman Problem*, abreviado por TSP). Este modelo es un problema clásico de Programación Lineal Entera relacionado con la teoría de redes (conjuntos de nodos que están interconectados entre sí y que intercambian flujos de información) y que se puede aplicar en situaciones muy diversas de la vida real. El modelo pretende averiguar la ruta más corta posible para un viajante que partiendo de un lugar original, tiene que visitar varias ciudades de la red pasando una única vez por cada una de ellas y al finalizar el recorrido tiene que volver al punto de partida. Para ello se deberá conocer la distancia existente entre cada dos ciudades o el coste de transporte de una ciudad a otra, gracias a esto podremos proceder a la resolución de dicho problema.

Cuando hablamos de viajante de comercio, como su nombre indica se puede interpretar como un individuo que realiza diferentes trayectos por el mundo generalmente por relaciones comerciales. En el ámbito de la economía, todo empresario pretende conocer la situación actual que mantienen las empresas que se encuentran a su cargo, por lo tanto, suele encontrarse en constante movimiento por los diversos lugares en los que se encuentran establecidas. El empresario suele visitar sus centros logísticos bien sea para conocer cómo va el funcionamiento de sus plantas en distintos lugares o bien para cerrar cualquier tema comercial referido a la compraventa de materias primas, firma de contratos con distintos distribuidores, etc. Así pues, se plantea el modelo que pretende minimizar la distancia, el tiempo o el coste del recorrido total.

En cuanto a la evolución histórica seguidamente citamos algunos acontecimientos relevantes. En el año 1857 el matemático W.R. Hamilton elaboró un juego matemático llamado Icosian cuyo objetivo era encontrar un camino que pasará por todos los vértices de un dodecaedro con la condición de terminar en el mismo vértice donde comenzó el

camino y teniendo en cuenta que cada vértice tenía que ser visitado sólo una vez. En términos generales, la situación puede modelizarse como un grafo, esto es, un par formado por un conjunto de vértices o nodos y un conjunto de aristas (conexiones entre nodos). Comentamos, además, que un camino de las características anteriores (que recorre todos los nodos una y sólo una vez y termina en el mismo nodo donde comienza) recibe el nombre camino hamiltoniano.

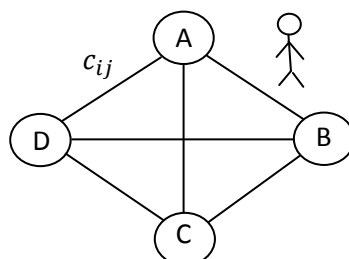
En 1954 G. Dantzig, R. Fulkerson, y S. Johnson hicieron una importante contribución ya que expresaron el problema como un modelo de Programación Lineal entera, y publicaron un método de solución del TSP titulado “*Solutions of a large scale traveling salesman problema*” (“Soluciones de gran escala para el problema del agente viajero”) para resolver una instancia de 49 ciudades donde encontraron el recorrido óptimo comprobando que no había ningún otro más corto.

A lo largo de los años muchos investigadores, matemáticos, científicos, etc., se han encargado de estudiar dicho problema, ello ha permitido generar continuos avances para la resolución óptima del TSP, y obtener cada vez soluciones con un mayor número de nodos (ciudades). En la actualidad, se han resuelto problemas a gran escala, por ejemplo con 85900 ciudades por D. Applegate, R. Bixby, V. Chvátal, W. Cook en el año 2006.

El objetivo principal del capítulo es encontrar una ruta óptima que empiece y termine en una ciudad en concreto, y que pase una sola vez por cada una de las ciudades del recorrido. En nuestro caso vamos a centrarnos en el objetivo de minimizar el coste de traslado del viajante.

En el modelo hay n ciudades, cada una se representa por un nodo, como se ilustra a continuación en la figura 3. Por cada dos nodos, pongamos i y j , una arista (i, j) que las conecta. Cada arista (i, j) lleva asociado un coste C_{ij} (coste que acarrea el traslado desde el nodo i al nodo j).

Figura 3: Esquema del problema viajante de comercio. Elaboración propia.



En nuestro caso, al igual que en el capítulo 1 vamos a solucionar un ejemplo real de la empresa Primark. Todos los datos recogidos son reales, estos han sido escogidos en base a determinados criterios como el menor número de escalas (vuelos directos entre ciudades) y el horario flexible preferiblemente en jornada laboral para dar mayor facilidad al empresario. El modelo, ha sido resuelto mediante un proceso iterativo de naturaleza técnica que viene recogido en el Apéndice 2.B.

Seguidamente resumimos los principales núcleos temáticos del capítulo:

- Modelo del viajante de comercio: Se introduce el planteamiento del modelo, establece la notación necesaria y presenta la formulación general para su resolución.
- Resolución en Matlab: El modelo planteado en el capítulo está resuelto a través de este software. Para resolver dicho problema hemos elaborado un programa genérico (véase el Apéndice 2.A), capaz de resolver el modelo planteado en este capítulo para cualquier conjunto de datos numéricos.
- Como aplicación, resolvemos una situación real de búsqueda del recorrido óptimo de visitas de los centros logísticos de Primark.

2.2. – Planteamiento del Modelo Viajante de Comercio

Para aplicar dicho modelo, el número de ciudades, n , ha de ser mayor o igual que 2. La colección de costes entre cada dos ciudades vienen recogidas en una matriz

$$C = (c_{ij})_{i,j=1,\dots,n}$$

donde c_{ij} representa el coste de traslado desde i hacia j . En este momento, observamos que los costes entre i y j podrían no ser simétricos; esto es, c_{ij} y c_{ji} podrían ser distintos.

En una primera impresión, la solución más directa para el problema consiste en valorar todas las posibles combinaciones obtenidas de los recorridos y quedarse con aquella cuyo trazado utiliza el menor coste. El problema se encuentra en el número de posibles combinaciones, que viene dado por el factorial del número de ciudades ($n!$), lo cual es impracticable con los medios computacionales actuales, incluso para valores de n moderados.

Para modelizar el problema del viajante de comercio como un problema de Programación Lineal Entera (véase Hillier y Lieberman (2010), pág. 428 para detalles), consideramos las variables $x_{ij}, i, j = 1, \dots, n$ definidas de la siguiente manera:

$$x_{ij} := \begin{cases} 1, & \text{si el viajante realiza el trayecto desde } i \text{ hasta } j, \\ 0, & \text{en caso contrario.} \end{cases}$$

En nuestro modelo, supondremos que $x_{ii} = 0, \forall i$, lo que se interpreta como que el individuo que se encuentra realizando el trayecto total no puede dirigirse de una ciudad a la misma ciudad.

Trabajaremos con el siguiente modelo para resolver nuestro problema del viajante de comercio:

$$\begin{aligned} (P_0) \text{ Min } & \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \\ \text{s. a } & \sum_{i=1}^n x_{ii} = 0, \\ & \sum_{j=1}^n x_{ij} = 1, i = 1, \dots, n, \quad \leftarrow x_{i1} + x_{i2} + x_{i3} + \dots + x_{in} = 1, \\ & \sum_{i=1}^n x_{ij} = 1, j = 1, \dots, n, \quad \leftarrow x_{1j} + x_{2j} + x_{3j} + \dots + x_{nj} = 1, \\ & x_{ij} \in \{0, 1\}, i, j = 1, \dots, n. \end{aligned}$$

Analizando las restricciones, la primera, $\sum_{i=1}^n x_{ii} = 0$, en nuestro caso de variable binarias, resulta ser equivalente a la condición " $x_{ii} = 0, \forall i$ ". A continuación, el segundo bloque de restricciones expresa el hecho de que solo debe haber un trayecto de salida de cada ciudad, y el tercero formaliza la idea de que solo debe haber un trayecto de llegada a cada una de las ciudades. Con esto se garantiza que cada ciudad es visitada exactamente una vez.

Formalmente, el problema (P_0) constituye una relajación del problema del viajante de comercio, dado que no incluye todas las restricciones de dicho problema. Concretamente, las restricciones de (P_0) no son suficientes para garantizar que sus soluciones sean recorridos que cumplan las características deseadas, puesto que, pueden aparecer subrutas (no conectadas entre sí). Por este motivo, en ocasiones, es necesario añadir más restricciones que eviten la formación de estas subrutas. Existen diferentes propuestas de resolución de este problema. En nuestro caso utilizaremos la siguiente estrategia:

- Si la solución óptima de (P_0) constituye un recorrido completo sin subrutas, dicha solución es la solución óptima del TSP.

- En caso contrario, si se produce alguna subruta, el valor óptimo de (P_0) será una cota inferior para el valor óptimo del TSP, es decir, el coste de la resolución del problema (P_0) es inferior al coste de TSP, debido a que en (P_0) se ha tenido en cuenta un mayor número de posibilidades. En este caso en el que se ha producido alguna subruta, añadiremos nuevas restricciones, y formamos de este modo un nuevo problema (P_1) . Las nuevas restricciones se añaden con la finalidad de romper las subrutas anteriores. Así pues, formalmente se escribirán como sigue:
 - ✓ Para evitar la subruta (formada por 2 ciudades) $i \rightarrow j \rightarrow i$, añadiremos $x_{ij} + x_{ji} \leq 1$.
 - ✓ Cuando tengamos que evitar la subruta (formada por 3 ciudades) $i \rightarrow j \rightarrow k \rightarrow i$, añadiremos $x_{ij} + x_{jk} + x_{ki} \leq 2$.
 - ✓ Así sucesivamente, cuando se trate de evitar una subruta formada por M ciudades en el miembro derecho pondremos $M - 1$.

2.3. – Formato Matricial

El problema inicial de optimización del viajante de comercio en formato matricial quedaría de la siguiente manera:

$$(P_0) \quad \begin{array}{l} \text{Min } c'x \\ \text{s. a } Aeq \ x = beq, \\ \quad \quad x \geq 0, \end{array}$$

donde:

- x representa al vector de variables del modelo, esto es,

$$x = (x_{11}, x_{12}, \dots, x_{1n}, x_{21}, \dots, x_{2n}, \dots, x_{n1}, \dots, x_{nn})' \in R^{n \times n},$$

- c denota al vector de coeficientes de la función objetivo, esto es, el vector de costes:

$$c = (c_{11}, \dots, c_{1n}, c_{21}, \dots, c_{2n}, \dots, c_{n1}, \dots, c_{nn})' \in R^{n \times n}$$

- Aeq representa a la matriz de coeficientes del sistema de restricciones (ecuaciones, en este caso). Así pues, será de la forma:

$$Aeq = \left(\begin{array}{c|c|c|c} \overbrace{1 \ 0 \ \dots \ 0}^n & \overbrace{0 \ 1 \ \dots \ 0}^n & \dots & \overbrace{0 \ \dots \ 0 \ 1}^n \\ \hline 1 \ 1 \ \dots \ 1 & 0 \ 0 \ \dots \ 0 & & 0 \ \dots \ 0 \ 0 \\ 0 \ 0 \ \dots \ 0 & 1 \ 1 \ \dots \ 1 & & 0 \ \dots \ 0 \ 0 \\ & \vdots & \dots & \vdots \\ 0 \ 0 \ \dots \ 0 & 0 \ 0 \ \dots \ 0 & & 1 \ \dots \ 1 \ 1 \\ \hline 1 \ 0 \ \dots \ 0 & 1 \ 0 \ \dots \ 0 & & 1 \ \dots \ 0 \ 0 \\ 0 \ 1 \ \dots \ 0 & 0 \ 1 \ \dots \ 0 & & 0 \ \dots \ 1 \ 0 \\ & \vdots & \dots & \vdots \\ 0 \ 0 \ \dots \ 1 & 0 \ 0 \ \dots \ 1 & & 0 \ \dots \ 0 \ 1 \end{array} \right)$$

(En la programación de este modelo en MATLAB hemos utilizado la notación K, Z e Y para los tres bloques que constituyen la matriz Aeq).

- beq denota al vector columna de términos independientes del sistema de ecuaciones del modelo, que viene dado por:

$$beq = (0, 1_n, 1_n)'$$

Como se ha comentado anteriormente, la solución propuesta por Matlab para este problema (P_0) puede contener alguna subruta. En este caso, resolveremos un nuevo problema (P_1) que se forma añadiendo al anterior un sistema de desigualdades. Así pues, el nuevo problema puede escribirse de la forma:

$$(P_1) \quad \begin{array}{l} \text{Min } c'x \\ \text{s. a } Ax \leq b, \\ \quad Aeq \ x = beq, \\ \quad x_{ij} \in \{0, 1\}, \ i, j = 1, \dots, n. \end{array}$$

donde el sistema de desigualdades añadido se corresponde con la colección de desigualdades necesarias para romper las subrutas que se hubieran formado en la solución de (P_0), atendiendo al procedimiento descrito al final del epígrafe anterior.

2.4. – Resolución del Modelo en Matlab

Para la ejecución de este problema se utilizará la función de Matlab “bintprog” (*binary integer programming*), al tratarse de un problema de optimización con variables binarias, que pueden tomar únicamente los valores 0 (no se realiza el trayecto) o 1 (si se realiza el trayecto).

Sintaxis(*) \rightarrow $x = \text{bintprog}(c, A, b, Aeq, beq)$

(*)Sintaxis con la versión R2014a de Matlab, en versiones posteriores, *bintprog* ha sido reemplazado por *intlinprog*.

Dependiendo de las subrutinas que vayan apareciendo, se irán actualizando las matrices *A* y *b*, hasta encontrar una solución óptima al TSP.

Es posible añadir otra opción de salida a la función *bintprog*; aparte de la solución óptima, si añadimos una segunda salida (denominada aquí *fval*), ésta almacenará el valor óptimo del problema, esto es, el coste del trayecto total. Formalmente, quedaría:

$$[x, fval] = \text{bintprog}(c, A, b, Aeq, beq).$$

2.5. – Un caso real: Ruta óptima entre los centros logísticos de la empresa Primark

El problema que abordamos en esta sección se contextualiza de nuevo (como en el capítulo 1) en la empresa Primark. En esta ocasión, deseamos resolver el problema del viajante de comercio que surge cuando un directivo de la empresa desea recorrer todos sus centros logísticos con el menor coste posible. Así pues, hemos de determinar la ruta menos costosa y que pase exactamente una vez por cada uno de los centros.

Los precios obtenidos para el coste de traslado de una ciudad a otra se han conseguido a través de internet en la página web www.rumbo.es a fecha de 19 de enero de 2016, puesto que nuestro cliente quiere a principio de año conocer de primera mano cómo se encuentran sus empresas y dar unos objetivos o pautas anuales para la organización.

La tabla que se muestra a continuación son los costes de traslado:

	Dublín	Frankfurt	Hannover	Londres	Edinburgh	Madrid	Valencia	Limerick	Ámsterdam	Utrecht	Lisboa	Braga	París	Nantes	Viena	Salzburgo	Brujas	Bruselas
Dublín	0	87	144	34	24	40	184	161	29	39	21	33	39	310	169	123	81	66
Frankfurt	99	0	105	93	345	87	339	228	114	122	311	366	48	245	211	191	289	274
Hannover	182	191	0	118	128	145	237	182	211	219	203	171	225	248	139	259	139	124
Londres	84	108	87	0	93	129	90	103	83	91	124	115	77	78	128	102	112	97
Edinburgh	35	294	133	68	0	154	149	252	56	64	152	167	98	304	165	149	177	162
Madrid	46	94	127	69	133	0	71	344	110	118	54	61	49	85	140	265	80	65
Valencia	142	208	138	109	101	70	0	159	86	94	80	129	136	125	145	602	108	93
Limerick	154	200	137	39	204	257	209	0	144	152	148	648	127	877	177	147	162	147
Ámsterdam	31	120	209	76	56	109	110	169	0	8	76	148	181	279	192	74	99	84
Utrecht	39	128	217	84	64	117	118	177	8	0	84	156	189	287	200	82	107	92
Lisboa	21	109	149	73	123	72	230	170	54	62	0	77	59	62	87	216	234	219
Braga	160	354	156	70	160	146	117	641	188	196	76	0	97	101	145	230	119	104
París	62	49	89	45	97	49	156	166	49	57	81	102	0	50	140	238	127	112
Nantes	110	295	303	59	103	80	315	870	96	104	106	51	48	0	301	1185	89	74
Viena	35	234	162	133	164	151	133	194	137	145	292	192	176	308	0	112	398	383
Salzburgo	153	196	280	108	144	219	437	175	50	58	329	375	144	1662	132	0	295	280
Brujas	91	299	89	99	150	75	96	183	63	71	248	116	125	90	80	228	0	15
Bruselas	76	284	74	84	135	60	81	168	48	56	233	101	110	75	65	213	15	0

Los costes se han tenido en cuenta en base a un precio medio de los encontrados en la web, ya que hay muchísimas opciones para escoger. La mayoría de los vuelos son en clase turista, y el criterio para elegirlo ha sido el siguiente:

- Elegir el menor número de escalas posibles para los trayectos establecidos, sin tener preferencia por una aerolínea en concreto.
- Preferentemente, vuelos en horario laboral.
- Flexibilidad horaria para tener facilidad para cambiar el billete.
- Escoger el trayecto que disminuya el tiempo de recorrido y el coste del viaje.

La fecha del 19 de enero de 2016, se ha escogido al azar para tener una orientación de los precios. En cuanto a la matriz de costes siempre va a ser cuadrada de tamaño $n \times n$, pero no va a ser simétrica debido a que los costes de traslado de las ciudades i y j , son diferentes de los adquiridos entre j e i .

El objetivo es:

$$(P_0) \quad \text{Min } 0x_{1,1} + 87x_{1,2} + 144x_{1,3} + 34x_{1,4} + 24x_{1,5} + 40x_{1,6} + 184x_{1,7} + 161x_{1,8} + 29x_{1,9} + 39x_{1,10} + 21x_{1,11} + 33x_{1,12} + 39x_{1,13} + 310x_{1,14} + 169x_{1,15} + 123x_{1,16} + 81x_{1,17} + 66x_{1,18} + 99x_{2,1} + 0x_{2,2} + 105x_{2,3} + 93x_{2,4} + 345x_{2,5} + 87x_{2,6} + 339x_{2,7} + 228x_{2,8} + 114x_{2,9} + 122x_{2,10} + 311x_{2,11} + 366x_{2,12} + 48x_{2,13} + 245x_{2,14} + 211x_{2,15} + 191x_{2,16} + 289x_{2,17} + 274x_{2,18} \\ \vdots \\ 76x_{18,1} + 284x_{18,2} + 74x_{18,3} + 84x_{18,4} + 135x_{18,5} + 60x_{18,6} + 81x_{18,7} + 168x_{18,8} + 48x_{18,9} + 56x_{18,10} + 233x_{18,11} + 101x_{18,12} + 110x_{18,13} + 75x_{18,14} + 65x_{18,15} + 213x_{18,16} + 15x_{18,17} + 0x_{18,18}$$

$$\text{Min } \sum_{i,j=1}^{18} C_{ij} X_{ij}$$

$$s.a \quad x_{1,1}, x_{2,2}, \dots, x_{18,18} = 0,$$

$$x_{ii} = 0, i = 1, \dots, 18$$

$$x_{1,1} + x_{1,2} + x_{1,3} + x_{1,4} + x_{1,5} + x_{1,6} + x_{1,7} + x_{1,8} + x_{1,9} + x_{1,10} + x_{1,11} + x_{1,12} + x_{1,13} + x_{1,14} + x_{1,15} + x_{1,16} + x_{1,17} + x_{1,18} = 1 \\ x_{2,1} + x_{2,2} + x_{2,3} + x_{2,4} + x_{2,5} + x_{2,6} + x_{2,7} + x_{2,8} + x_{2,9} + x_{2,10} + x_{2,11} + x_{2,12} + x_{2,13} + x_{2,14} + x_{2,15} + x_{2,16} + x_{2,17} + x_{2,18} = 1 \\ \vdots$$

$$x_{18,1} + x_{18,2} + x_{18,3} + x_{18,4} + x_{18,5} + x_{18,6} + x_{18,7} + x_{18,8} + x_{18,9} + x_{18,10} + x_{18,11} + x_{18,12} + x_{18,13} + x_{18,14} + x_{18,15} + x_{18,16} + x_{18,17} + x_{18,18} = 1$$

$$\sum_{i=1}^{18} x_{ij} = 1, j = 1, \dots, 18$$

$$\begin{aligned} x_{1,1} + x_{2,1} + x_{3,1} + x_{4,1} + x_{5,1} + x_{6,1} + x_{7,1} + x_{8,1} + x_{9,1} + x_{10,1} + x_{11,1} + x_{12,1} + x_{13,1} + x_{14,1} + x_{15,1} + x_{16,1} + x_{17,1} + x_{18,1} &= 1 \\ x_{1,2} + x_{2,2} + x_{3,2} + x_{4,2} + x_{5,2} + x_{6,2} + x_{7,2} + x_{8,2} + x_{9,2} + x_{10,2} + x_{11,2} + x_{12,2} + x_{13,2} + x_{14,2} + x_{15,2} + x_{16,2} + x_{17,2} + x_{18,2} &= 1 \\ &\vdots \\ x_{1,18} + x_{2,18} + x_{3,18} + x_{4,18} + x_{5,18} + x_{6,18} + x_{7,18} + x_{8,18} + x_{9,18} + x_{10,18} + x_{11,18} + x_{12,18} + x_{13,18} + x_{14,18} + x_{15,18} + x_{16,18} + x_{17,18} + x_{18,18} &= 1 \end{aligned}$$

$$\sum_{j=1}^{18} x_{ij} = 1, i = 1, \dots, 18$$

Lo que pretendemos con el Matlab es crear un programa que esté preparado para crear una matriz de manera sistemática, donde al usuario únicamente se le pidan los costes y que automáticamente, conociendo los costes, te resuelva el problema con la ejecución del “*bintprog*”.

Tras la ejecución del programa general, como es habitual han aparecido una serie de subrutas que a pesar de dar el valor óptimo para P_0 puesto que cumple todas las restricciones del problema, no ha conseguido llegar al óptimo del TSP. Las subrutas que se han formado han sido las siguientes:

- Londres → Limerick → Londres
- Madrid → Valencia → Madrid
- Frankfurt → París → Frankfurt
- Brujas → Bruselas → Brujas
- Amsterdam → Utrech → Salzburgo → Amsterdam
- Lisboa → Braga → Nantes → Lisboa
- Dublín → Hannover → Edimburgo → Viena → Dublín

En este caso, añadimos unas nuevas restricciones de dichas subrutas a las anteriores ya establecidas; se genera de esta forma un nuevo problema (P_1) que reemplazara al (P_0) y

volvemos a ejecutar el programa. Tras romper todas las subrutas anteriormente mencionadas, aparecen otras que introducimos seguidamente:

Dublín → Edimburgo → Dublín
Amsterdam → Utrech → Amsterdam

Es necesario realizar otra vez el paso anterior, añadiendo estas subrutas a las anteriores y sustituyendo (P_1) por un nuevo problema (P_2).

Por último, tras la aplicación de las subrutas (véase Apéndice 2.B) hemos logrado alcanzar nuestro valor óptimo para TSP, y por lo tanto, nuestro objetivo final. Nuestro cliente puede realizar dicho circuito con un coste mínimo de traslado de 1183€

Dublín → Braga → Lisboa → Nantes → París → Frankfurt → Hannover →
Limerick → Londres → Valencia → Madrid → Bruselas → Brujas → Viena →
Salzburgo → Utrech → Ámsterdam → Edinburgo → Dublín



APÉNDICE 2.A: PROGRAMACIÓN EN MATLAB (Modelo del Viajante de Comercio General)

%Este programa resuelve de manera genérica el problema del viajante de comercio. Suponemos que existen unos c costes de traslado introducidos por el cliente. En primer lugar, se colocan las variables x_{ij} (Coste del trayecto realizado desde i hasta j).

```
c=input('Introduzca la matriz de costes');
s=size(c)
t=s(1)^2 %Número de trayectos posibles
n=s(1) %Número de ciudades
A=[];
b=[];
k=zeros(1,t);
for i=1:n
    k((i-1)*18+i)=1;
end
```

```
z=zeros(n,t);
for i=1:n
    z(i,((i-1)*18+1):(i*18))=ones(1,18);
end
y=[];
for i=1:n
    y=[y,eye(n)];
end
```

```
Aeq=[k;z;y];
beq=[0;ones(2*n,1)];
[Solucion,fval]=bintprog(c,A,b,Aeq,beq)
```

APÉNDICE 2.B: CASO PARTICULAR DEL PRIMARK CON MATLAB

Este apéndice hace referencia a la representación de las subrutas, para resolver dicho problema en particular es necesaria la introducción de las subrutas en el problema general debido a que debemos ejecutarlo en el Matlab para conseguir la solución óptima del problema TSP con coste mínimo.

```
A=[zeros(1,61),1,zeros(1,67),1,zeros(1,194);...
    zeros(1,96),1,zeros(1,16),1,zeros(1,210);...
    zeros(1,30),1,zeros(1,186),1,zeros(1,106);...
    zeros(1,305),1,zeros(1,16),1,0;...
    zeros(1,159),1,zeros(1,10),1,zeros(1,108),1,zeros(1,44);...
    zeros(1,193),1,zeros(1,14),1,zeros(1,36),1,zeros(1,78);...
    zeros(1,4),1,zeros(1,45),1,zeros(1,23),1,zeros(1,177),1,zeros(1,71);...
    zeros(1,4),1,zeros(1,67),1,zeros(1,251);...
    zeros(1,153),1,zeros(1,16),1,zeros(1,153)];
b=[ones(1,4)';2;2;3;1;1];
```


CAPÍTULO 3.- SERIES TEMPORALES

3.1. - Introducción

Este capítulo se centra en el estudio de una serie temporal, la cual será analizada a través del programa informático Matlab. Para ello debemos saber que una serie temporal es un conjunto de valores observados de una característica poblacional (variable) que se han observado de forma secuencial en el tiempo y que están ordenados de manera cronológica. Estas observaciones se suelen recoger en instantes de tiempo equiespaciados (intervalos de tiempo de la misma longitud).

Así, en el estudio de una serie temporal observamos la evolución que ha experimentado una variable a lo largo del tiempo con el objetivo de obtener un modelo que permita predecir sus valores futuros. Todo esto nos informa ante posibles alteraciones, cambios de ciclo o crisis económicas a lo largo del tiempo, para así poder adelantarnos y tener una visión orientativa de nuestro futuro gracias a los estudios exhaustivos de los datos estadísticos recogidos de manera periódica.

Nuestras principales aportaciones para este capítulo serán dos, hacer el estudio de una serie temporal compuesta con datos reales del *INE* (Instituto Nacional de Estadística) y otro hecho a destacar es crear unos programas en Matlab que ayuden en la identificación de modelos. Destacamos el hecho de introducir un programa para obtener las llamadas función de autocorrelación simple (*fas*) y función de autocorrelación parcial (*fap*), las cuales constituyen ingredientes fundamentales en el estudio de una serie temporal (véase la sección 3.2). El programa ha sido desarrollado para que pueda ser utilizado por cualquier usuario, en nuestro caso hemos analizado un ejemplo muy conocido, el clásico de los pasajeros de avión (1970) y también un caso real del INE en referencia al número de hipotecas solicitadas en la provincia de Alicante de 1994-2014.

El capítulo contiene tres núcleos temáticos:

- Herramientas básicas para la comprensión de una serie: Comenzamos describiendo brevemente ciertos pasos básicos a seguir para el análisis de una serie temporal. Concretamente, se da un breve repaso del análisis descriptivo, así como de ciertos tipos de transformaciones, y determinados modelos que ampliaremos más adelante.
- Resolución en Matlab: Los ejemplos que han sido planteados en el capítulo están resueltos a través de este software, para ello debemos aprender los diferentes comandos del programa para así crear los códigos necesarios para obtener la *fap* y *fas* que posteriormente comentaremos.
- Desarrollo de una aplicación con datos reales relativos a las solicitudes de hipotecas en los últimos 20 años en la provincia de Alicante.

3.2. – Herramientas Básicas del Estudio de una Serie Temporal

En esta sección describimos las herramientas necesarias para la comprensión de la aplicación real que desarrollamos en el apartado 3.4. Todas estas herramientas serán imprescindibles para extraer las conclusiones de las distintas series temporales que procedamos analizar.

Dada una serie temporal, el objetivo principal del capítulo es describir su comportamiento mediante un modelo que utilizaremos para realizar predicciones. En nuestro caso trabajaremos con modelos estacionarios cuya explicación está en el 1^{er} paso, de manera que, en ocasiones, tendremos que realizar diferentes transformaciones a nuestra serie con el fin de ajustarla a uno de los modelos considerados (véase detalles en el Apéndice 3.A). Así pues estructuraremos nuestro análisis en tres pasos.

1^{er} Paso: Análisis descriptivo. Observando la representación gráfica de una serie, veremos si se trata de una serie estacionaria o no, y si presenta o no un carácter estacional. De modo intuitivo, una **serie es estacionaria** cuando no se le aprecia ninguna tendencia, creciente o decreciente, y además cuando la franja en la que se mueven los datos tiene una anchura constante. Estas dos características se corresponden con la estacionariedad en media y en varianza, respectivamente.

En caso contrario si apreciamos que la **serie no es estacionaria** (por ejemplo, vemos claramente que su tendencia global es creciente o decreciente), hemos de plantearnos qué tipo de transformaciones podríamos practicarle a nuestra serie original con el fin de obtener una serie estacionaria.

Por otro lado, en términos informales, entendemos que una **serie es estacional** cuando presenta un comportamiento periódico con una periodicidad menor que un año (por ejemplo, que se aprecian ciertos comportamientos repetitivos por meses, por estaciones, por trimestres, etc.), en caso de no observar un comportamiento similar en el estilo de nuestra **serie no será estacional**.

2º Paso: En numerosas ocasiones, nuestra serie no será estacionaria, en cuyo caso, como hemos comentado anteriormente, tendremos que realizar alguna transformación sobre los datos originales. Emplearemos dos tipos de transformaciones:

- Transformaciones orientadas a estabilizar la varianza:

- En nuestro caso emplearemos la **transformación logarítmica**, que es una de las transformaciones pertenecientes a la familia de transformaciones de BOX-COX (véase Peña (2005) para detalles).

Dicha transformación será necesaria cuando la variabilidad entre los datos va variando a lo largo del tiempo de la serie. Esta transformación se representa de la siguiente manera:

Formalmente, a partir de la serie original $\{Z_t, t = 1, 2, \dots, N\}$ se define una nueva serie $\{Y_t, t = 1, 2, \dots, N\}$ como sigue:

$$Y_t = \log(Z_t), \quad t = 1, 2, \dots, N.$$

-Transformaciones orientadas a estabilizar la media:

- La llamada **diferencia regular** (a cada dato restamos el anterior) que emplearemos cuando apreciemos una tendencia creciente o decreciente.

Esta transformación se realiza para crear una serie estacionaria en media, la cual trata de eliminar la tendencia (creciente o decreciente) de la serie para así mantener la tendencia constante en el tiempo. Esta diferencia se presenta de la siguiente forma: a partir de una serie $\{Z_t, t = 1, 2, \dots, N\}$,

$$R_t = Z_t - Z_{t-1}, \quad t = 2, \dots, N.$$

- Cuando observemos un comportamiento periódico, en primer lugar, trataremos de averiguar el periodo de la serie (mensual, trimestral, por estaciones, etc.) y, utilizando este dato, practicaremos la llamada **diferencia estacional**. Esta transformación se aplica de manera similar a la anterior operación solo que la diferencia se realiza entre las observaciones equiespaciadas por el periodo l . Esta diferenciación se escribe como:

$$S_t = Z_t - Z_{t-l}, \quad t = l + 1, \dots, N.$$

En ocasiones, tendremos que realizar de manera secuencial distintas transformaciones, además de una diferencial estacional, diferentes diferencias regulares.

3ºPaso: Una vez que tenemos una serie estacionaria, buscaremos un modelo adecuado dentro de la familia de modelos denominados ARMA (que viene del inglés *autoregressive moving average*).

Dada una serie temporal original de datos Z_t , el modelo ARMA nos ayuda a predecir los futuros valores que puede tener la serie. Este modelo se encuentra formado por dos partes; por un lado está la parte autorregresiva (AR) y por otro lado la parte de media móvil (MA) que se encuentran explicados en el Apéndice 3.A.

Las herramientas necesarias para identificar el modelo: *fas* y *fap*.

La función de autocorrelación simple (*fas*), para cada i , indica la correlación lineal entre Z_t y Z_{t-i} (correlación entre dos instantes con un retardo de i unidades), por ejemplo *fas* (1): correlación entre lo que ocurre en un instante y el instante anterior.

La función de autocorrelación parcial (*fap*), para cada i , proporciona una medida de la correlación lineal que hay entre las variables Z_t y Z_{t-i} eliminando el efecto de todas las variables intermedias.

En cuanto al enfoque que vamos a utilizar en dicho capítulo, es el predictivo, ya que estudiaremos los modelos ARMA que mejor describen el comportamiento de nuestra serie, y la finalidad es hacer predicciones futuras.

NOTA: No es nuestro objetivo realizar un estudio exhaustivo de las series temporales sino presentar únicamente las herramientas necesarias para desarrollar la aplicación real de la sección 3.4. Simplemente comentaremos que si la serie original ha requerido una

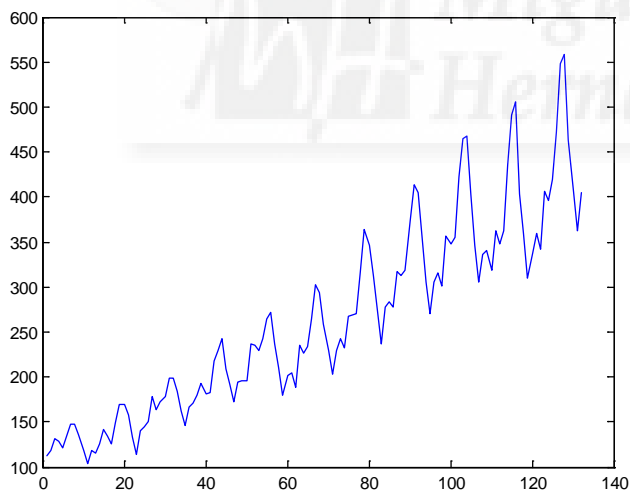
diferencia regular para llegar a un modelo estacionario, la serie original se dice que se ajusta a un modelo ARIMA (Véase Peña (2005) para detalles) y en el caso de tener que aplicar una diferencia estacional, la serie original se dice que se ajusta a un modelo SARIMA (Véase Peña (2005) para detalles). Por cuestiones de espacio, el análisis o estudio de las predicciones lo explicaremos directamente sobre la aplicación que vamos a desarrollar, y dando unas pinceladas en el momento necesario para así comprenderla en su totalidad.

3.3. – Ejemplo y Resolución con Matlab

Seguidamente presentamos un ejemplo clásico con el que se pretenden ilustrar los conceptos anteriores. A continuación vamos a proceder al desarrollo de la serie de pasajeros de avión en vuelos internacionales (1970).

Los datos se encuentran recogidos de manera mensual a lo largo de 12 años, luego el total de datos que disponemos es de 144 datos.

Gráfica 1: Representación de la serie. Elaboración propia



Con el comando “mean” calculamos la media de nuestra serie:

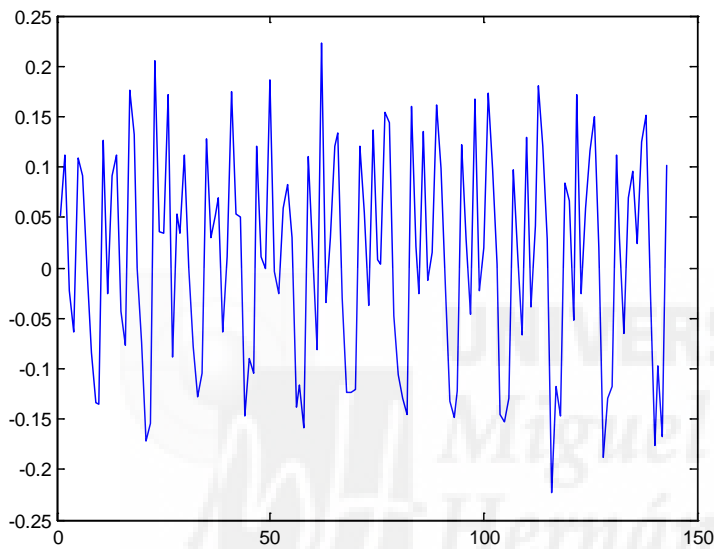
```
mean (Aline)= 280.2986
```

De modo intuitivo, fijándonos en el gráfico proporcionado, podemos decir que nuestra serie tiene una tendencia creciente y no es estacionaria en varianza, ya que su

variabilidad crece a lo largo del tiempo. Nótese que la franja en la que se encuentran los valores va incrementándose en anchura.

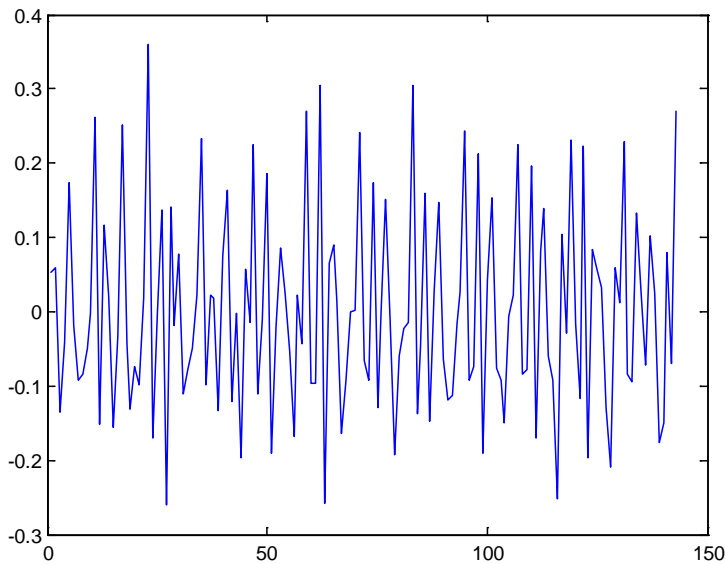
En primer lugar, debemos corregir la no estacionariedad de nuestra serie temporal, tanto en media, como en varianza. Para ello, comenzaremos realizando una transformación logarítmica, y posteriormente una diferenciación regular, y obtenemos la siguiente figura.

Gráfica 2: Transformación logarítmica y diferencia regular. Elaboración propia



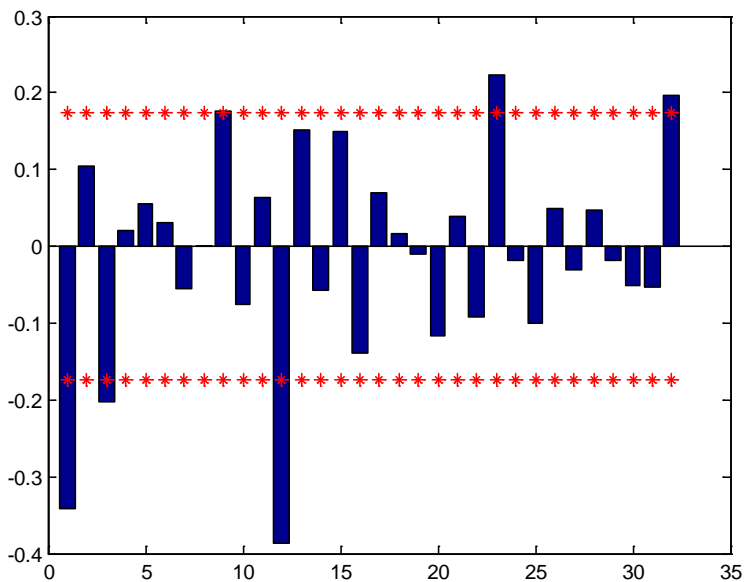
En el gráfico anterior, se aprecia que la serie transformada es estacionaria en media y varianza, hemos conseguido corregir la no estacionariedad de la serie en varianza. Seguidamente realizamos una diferenciación estacional.

Gráfica 3: Diferencia estacional. Elaboración propia



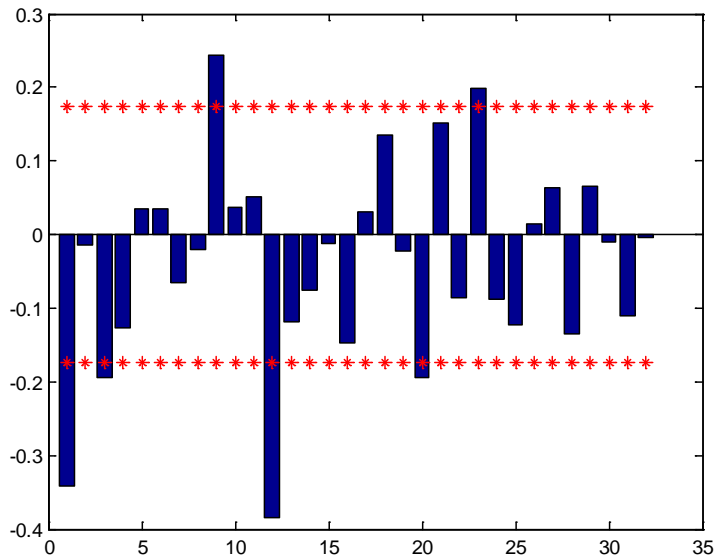
Una vez realizadas ambas diferencias, se pasará a estudiar las autocorrelaciones. Para el desarrollo del correlograma (*fas* muestral, la obtenida a partir de datos muestrales) y las autocorrelaciones parciales (*fap* muestral), se creará un script que se encuentra en el Apéndice 3.D y que nos facilitará la representación de ambas en unos gráficos. En primer lugar, nos centramos en el correlograma, que ejecutando su respectivo script se obtiene lo siguiente:

Gráfica 4: Autocorrelaciones simples. Elaboración propia



Para representar la gráfica de las autocorrelaciones parciales también es necesario ejecutar el script propuesto en el Apéndice 3.D.

Gráfica 5: Autocorrelaciones parciales. Elaboración propia



Tras la realización de dichos gráficos ya tenemos todo lo necesario para conocer de manera intuitiva cual es el tipo modelo que puede ajustarse a dicha serie temporal. Las transformaciones empleadas han sido: la transformación logarítmica, para corregir el crecimiento de la varianza y, posteriormente, la diferenciación regular para corregir la tendencia creciente de la serie, y finalmente una diferenciación estacional. Como conclusión podemos decir que nos encontramos ante un modelo SARIMA. Sin entrar en detalles técnicos, comentaremos que dichos modelos incluyen una serie de parámetros por determinar para realizar una identificación completa del modelo. En nuestro caso, haciendo uso de las gráficas de la *fas* y la *fap*, y comparándolas con las que se encuentra en el texto de Peña (2005), en la pág. 207, llegamos a la conclusión de que se trata de un modelo que se expresa simbólicamente como: SARIMA (0,1,1)(0,1,1)₁₂ (véase el apéndice 3.B para detalles).

A la fase de identificación del modelo, le sigue la de estimación de los parámetros y la de obtención de estimaciones. Esta última etapa la hemos llevado a cabo directamente con el programa estadístico SPSS, obteniendo el modelo:

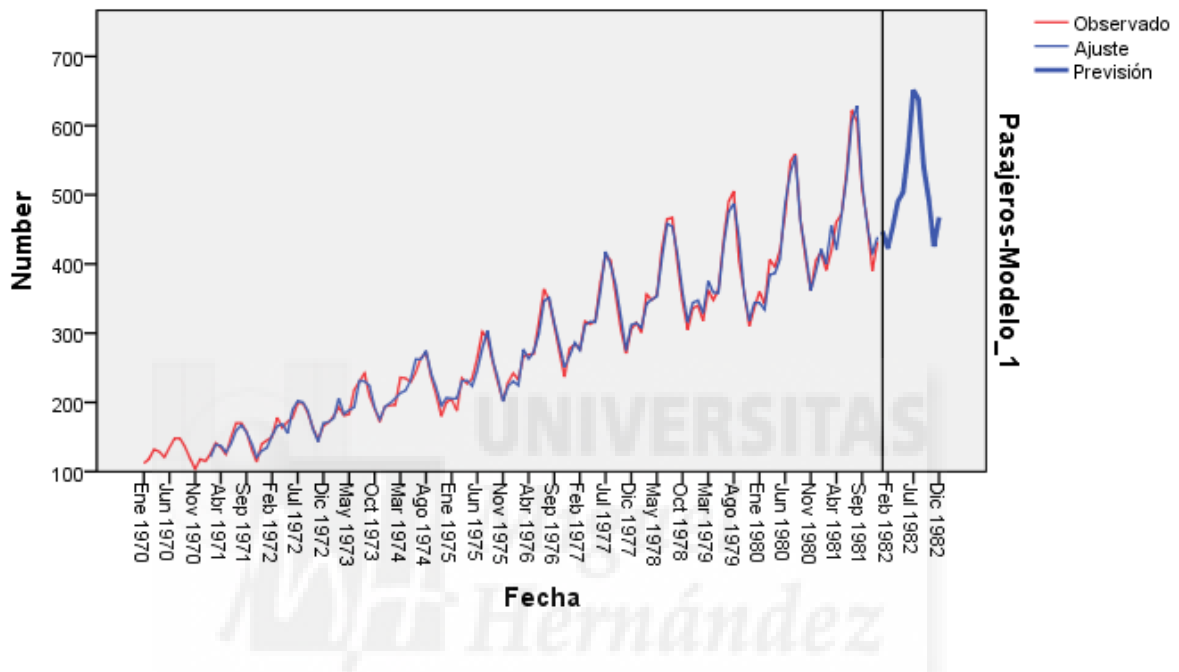
$$Z_t = Z_{t-1} + Z_{t-12} - Z_{t-13} + a_t - 0,107a_{t-12} - 0,310a_{t-1} + 0,03317a_{t-13} + 0,199,$$

donde Z_t representa el valor de la serie en el instante t , el cual se expresa en función de los valores en los instantes $t-1$, $t-12$ y $t-13$. Por su parte, $\{a_t\}$ representa un ruido

blanco, esto es, una sucesión de variables aleatorias de media (esperanza) 0, varianza constante e incorreladas dos a dos. Dirigimos al lector al apéndice 3.B para algunos detalles técnicos sobre la elección de este modelo.

Seguidamente mostramos la gráfica que nos permite comparar intuitivamente la serie de valores observados y la de sus estimaciones a partir del modelo seleccionado.

Gráfica 6: Valores estimados. Elaboración propia



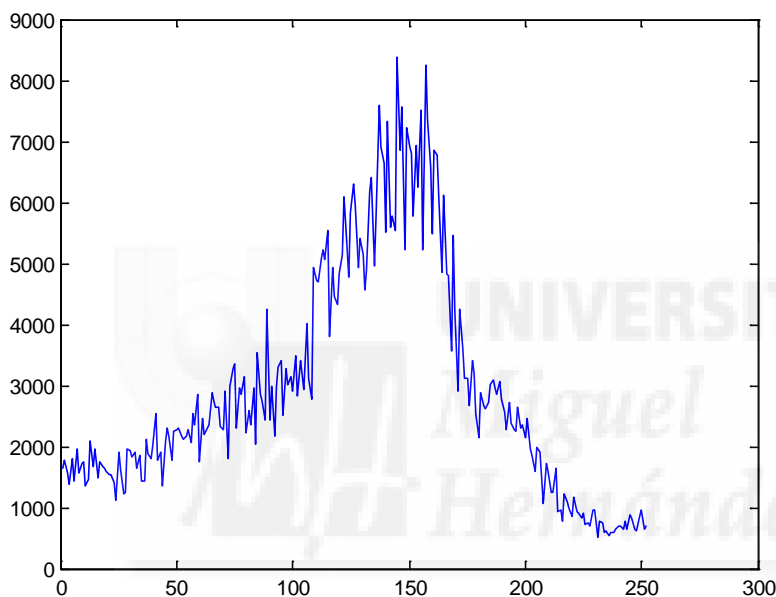
En el gráfico se observan dos líneas, una roja que corresponde a los valores observados y la azul que es la que ha estimado nuestro modelo, incluyendo estimaciones futuras. Obsérvese que en el eje vertical se encuentra el número de pasajeros de avión que realizaron vuelos internacionales, mientras que en el eje horizontal aparece el tiempo en meses entre los años 1970-1982. En cuanto a la interpretación de dicho gráfico comentamos que el número de pasajeros aumenta a final de año, en fiestas o puentes nacionales y que sobretodo se dispara en los meses de verano, entiendo esto como el periodo entre Junio-Agosto. También al mirar la ilustración se puede llegar a la conclusión que con el paso de los años el número de pasajeros ha ido aumentando gracias a la competitividad entre las distintas compañías de vuelo. Además, se aprecia un aumento de la variabilidad de los datos conforme pasa el tiempo.

3.4. – Un caso real: Estudio de la evolución del número de hipotecas

En este apartado vamos a proceder al análisis de una base de datos escogida del INE (Instituto Nacional de Estadística), más concretamente una serie relacionada con el número de hipotecas realizadas en la provincia de alicante entre los años 1994-2014.

Los datos se encuentran recogidos de manera mensual a lo largo de 20 años, luego el total de datos que disponemos es de 240 datos.

Gráfica 7: Representación de la serie. Elaboración propia



Con el comando “mean” calculamos la media de nuestra serie:

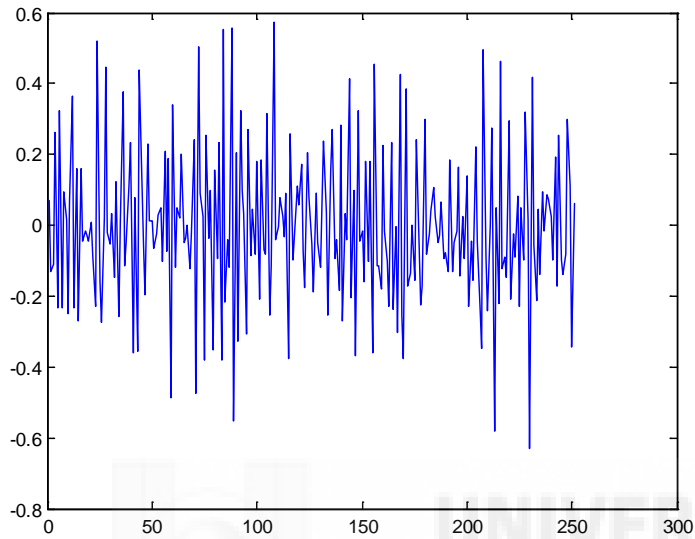
```
mean (hipoteca)= 2939,40476
```

De modo intuitivo fijándonos en el gráfico proporcionado, observamos que nuestra serie presenta una tendencia que primero crece y después decrece, por lo que intuimos que no es estacionaria en media. Por otra parte, de nuevo de forma intuitiva, apreciamos que la variabilidad de las observaciones va aumentando hasta determinado momento en el que comienza a decrecer, por lo que entendemos que tampoco es estacionaria en varianza.

En primer lugar, debemos corregir la no estacionariedad de nuestra serie temporal en media y en varianza. Para ello, como hemos comentado anteriormente, realizamos una

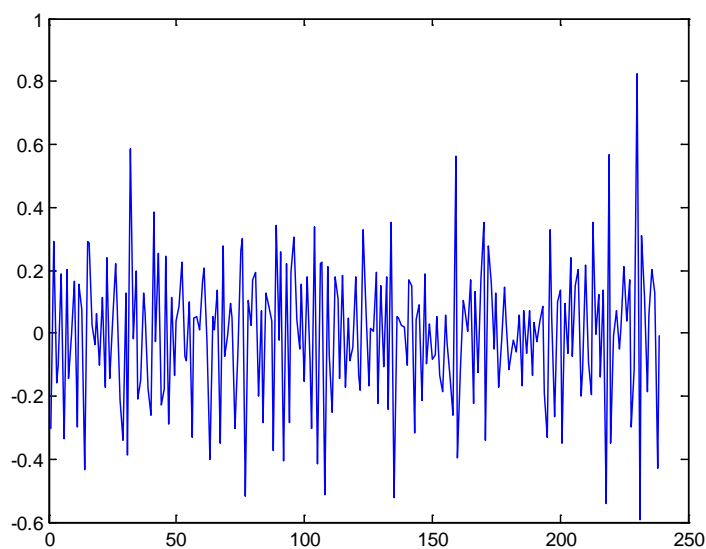
transformación logarítmica y posteriormente una diferenciación regular (el desarrollo de dicha diferenciación la encontramos en el Apéndice 3.E). De este modo, obtenemos la siguiente gráfica.

Gráfica 8: Transformación logarítmica y diferencia regular. Elaboración propia



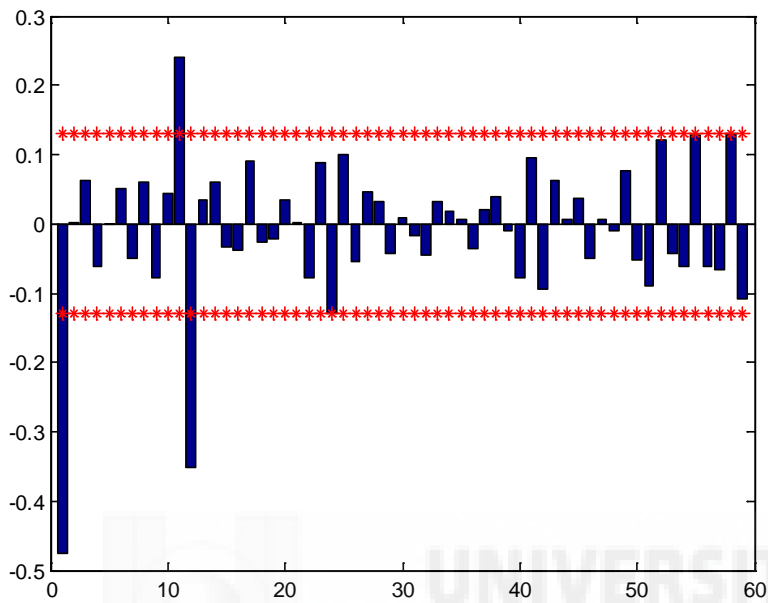
Nuestra serie se ha convertido en estacionaria en media y en varianza, a continuación realizamos una diferenciación estacional.

Gráfica 9: Diferencia estacional. Elaboración propia

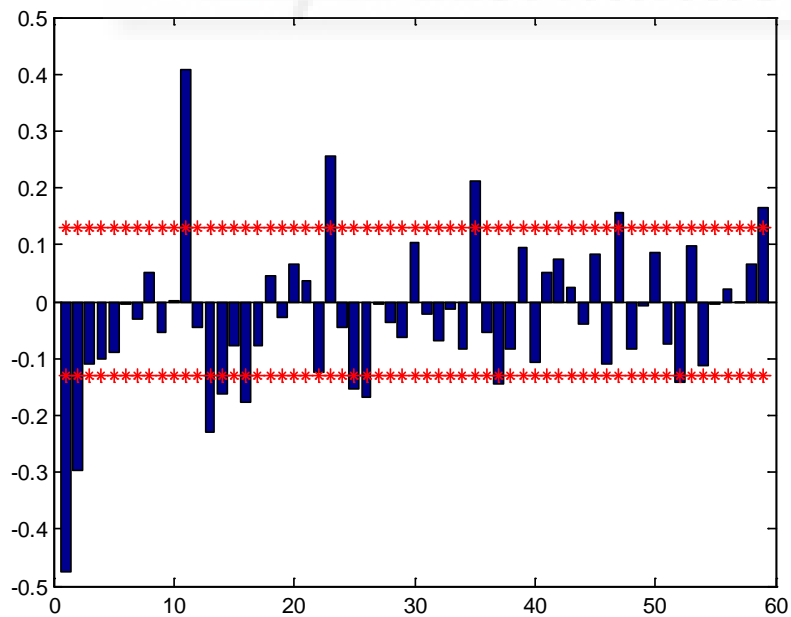


Ejecutamos la opción serierecorrelograma y la *fap* del Apéndice 3.D. para la representación de las autocorrelaciones, y obtenemos lo siguiente:

Gráfica 10: Autocorrelaciones simples finales. Elaboración propia



Gráfica 11: Autocorrelaciones parciales finales. Elaboración propia



Tras la representación de las autocorrelaciones podemos conocer de manera intuitiva que modelos pueden ajustarse mejor a nuestra serie. La utilización de las

transformaciones, diferencias regulares y estacionales hacen que nos situemos de nuevo ante un modelo SARIMA. La identificación completa del modelo nos conduce a un SARIMA(0,1,1)(0,1,1)₁₂. De nuevo, la identificación la hemos realizado de forma intuitiva comparando la *fas* y la *fap* obtenidas con las que encontramos como modelo en el libro de Peña (2005).

Finalmente, con ayuda del programa SPSS, hemos realizado la estimación de parámetros del modelo y la obtención de predicciones.

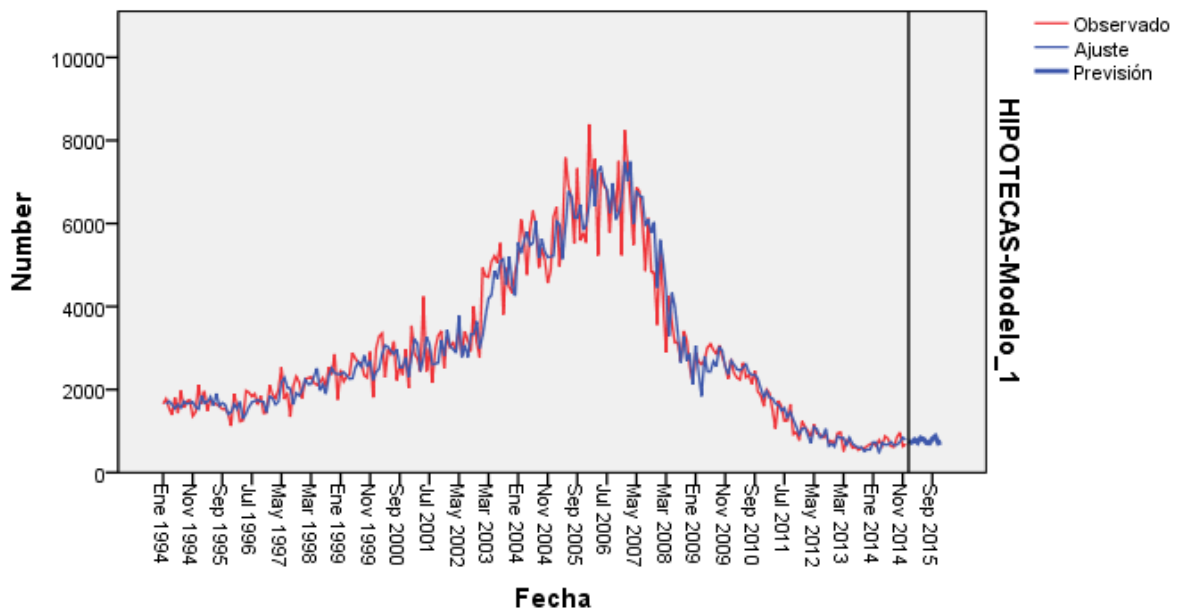
Seguidamente, mostramos el modelo empleado para explicar el comportamiento de nuestra serie y una gráfica donde figuran las estimaciones realizadas con este modelo y las predicciones futuras. El modelo quedaría de la siguiente forma (véase el apéndice 3.C para detalles técnicos sobre la elección del modelo):

$$Z_t = Z_{t-1} + 0,493Z_{t-12} - 0,493Z_{t-13} + a_t - 0,624a_{t-1} - 2,334,$$

donde podemos concluir que la estimación de la variable en el instante t , Z_t , se obtiene a partir de los valores en los instantes $t-1$, $t-12$ y $t-13$. Recordemos que $\{a_t\}$ representa un ruido blando (una sucesión de variables aleatorias de media 0, varianza constante e incorreladas dos a dos).

En la siguiente figura, de nuevo, la gráficas de color rojo y azul se corresponden con los valores observados y estimados, respectivamente.

Gráfica 12: Valores estimados. Elaboración propia



APÉNDICE 3.A: MODELOS

Los modelos autorregresivos son modelos de regresión, en los que se pretende explicar una variable estadística como función de otras variables.

El modelo autorregresivo AR (p): Este modelo presenta ciertas características que comentaremos a un nivel intuitivo, lo que hace es tratar de explicar lo que ocurre en un instante determinado en función de lo que viene ocurriendo en los $t - p$ instantes anteriores. Siempre denotaremos por a_t a un conjunto de variables aleatorias, de media cero, varianza constante, e incorreladas dos a dos.

La formulación general de AR (p) es la que aparece a continuación:

$$Z_t = \phi_1 Z_{t-1} + \dots + \phi_p Z_{t-p} + a_t$$

O equivalentemente

$$\phi(B)Z_t = a_t$$

Donde $\phi(B) = 1 - \phi_1 B - \dots - \phi_p B^p$. La notación ϕ_1, \dots, ϕ_p hace referencia a los parámetros del modelo.

El modelo de regresión explica la evolución de Z_t como una variable dependiente, en función de otras variables Z_{t-1}, Z_{t-2} hasta Z_{t-p} , donde consideramos un modelo lineal con unos parámetros de ϕ_1 y ϕ_p que son constantes que tenemos que determinar.

En nuestro caso nos vamos a centrar como máximo en modelos de orden AR (1) y AR (2).

Dos herramientas clave son las llamadas *fas* y *fap* que implementaremos en Matlab. La *fas* de un modelo autorregresivo se caracteriza por tener un decrecimiento exponencial (el estudio de una característica en el tiempo se va distanciando entre dos instantes de tiempo donde van perdiendo correlación entre ellos); pero con la *fas* sola no somos capaces de distinguir entre los distintos tipos de modelos autorregresivos y por lo tanto nos hace falta fijarnos en la segunda herramienta, la *fap*. Observaremos los picos significativos de la *fap* para saber el tipo de modelo que debemos de seleccionar. De manera intuitiva, si nos aparecen 2 barras que sobresalen del intervalo de confianza, nos encontraríamos ante un modelo AR (2) (véase Peña(2005) pág.127).

El modelo de media móvil MA (q): Dicho modelo presenta una serie de características que explicaremos a nivel intuitivo. La formulación general de MA (q) es la que aparece a continuación:

$$Z_t = a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q}$$

O equivalentemente

$$Z_t = \theta(B)a_t$$

Donde $\theta(B) = 1 - \theta_1 B - \dots - \theta_q B^q$.

La notación $\theta_1, \dots, \theta_q$ hace referencia a los parámetros del modelo.

En nuestro caso nos vamos a centrar como máximo en modelos de orden MA (1) y MA (2). La *fas* de un modelo de media móvil se caracteriza por tener una cantidad finita de picos significativos, mientras que la *fap* tiene que presentar un decrecimiento exponencial, como apreciamos, la *fas* y la *fap* se intercambian los papeles con respecto a lo que ocurre en los modelos autorregresivos (véase Peña(2005) pág.150).

El combinado de ambos es el modelo ARMA (p, q) se conoce por tener orden p en la parte autorregresiva y orden q en la de media móvil. Formalmente el modelo, se escribe como sigue:

$$(1 - \phi_1 B - \dots - \phi_p B^p)Z_t = (1 - \theta_1 B - \dots - \theta_q B^q)a_t$$

Para conocer la *Fas* y *Fap* de los procesos ARMA (véase Peña(2005) pág.157).

APÉNDICE 3.B: Identificación del modelo del ejemplo introducido en la sección 3.3.

Para la identificación del modelo ha sido necesario conocer el significado de cada uno de los componentes que forma un modelo SARIMA, este se representa de la siguiente manera, $(p, d, q)(P, D, Q)_s$, donde:

- El primer paréntesis corresponde a la parte no estacional, donde p y q son respectivamente la parte autorregresiva y de media móvil, y la d es el número de diferencias regulares que se han aplicado a la serie.

- El segundo paréntesis hace referencia a la parte estacional, donde P y Q son también la parte autorregresiva y de media móvil, y la D son el número de diferencias estacionales que le han practicado a la serie.

Para averiguar dichos parámetros habrá que fijarse en las gráficas de autocorrelaciones. En cuanto a la parte no estacional, atendiendo al correlograma (autocorrelacion simple), observamos que tenemos un pico significativo, y un decrecimiento exponencial (decrece rápidamente en el tiempo). Por otro lado si miramos las autocorrelaciones parciales obtenemos los mismos resultados que en el correlograma. Por lo tanto, se aprecian dos modelos, un modelo autorregresivo AR(1) con $p = 1$ y un modelo de media móvil MA(1) con $q = 1$.

En cuanto a la parte estacional (véase Peña (2005) pág.207 para detalles) observamos en el correlograma obtenido que estamos ante un MA(1) estacional con $Q = 1$.

En resumen, atendiendo a las transformaciones que han sido significativas y a las características que presentan las autocorrelaciones, podemos concluir que los modelos que mejor se ajustan a la serie son el modelo SARIMA (0,1,1)(0,1,1) y el (1,1,0)(0,1,1).

DESARROLLO DE LOS MODELOS EN SPSS

Los modelos identificados en el apartado anterior deben de cumplir las tres hipótesis básicas de toda serie temporal que son la homocedasticidad, normalidad e independencia con un nivel de significación de 5%. La finalidad consiste en conocer cuál es el modelo que mejor se ajusta a nuestra serie, como ambos lo cumplen todo quedarían de la siguiente forma:

$$1) \text{ SARIMA}(0,1,1)(0,1,1)_{12}$$

$$(1 - B)(1 - B^{12}) Z_t = (1 - \phi_1 B)(1 - \phi_{12} B^{12}) a_t + C$$

$$Z_T = Z_{t-1} + Z_{t-12} - Z_{t-13} + a_t - \phi_{12} a_{t-12} - \phi_1 a_{t-1} + \phi_1 \phi_{12} a_{t-13} + C$$

$$Z_T = Z_{t-1} + Z_{t-12} - Z_{t-13} + a_t - 0,107 a_{t-12} - 0,310 a_{t-1} + 0,03317 a_{t-13} + 0,199$$

Para calcular el error del modelo es necesario la utilización de la tabla que nos proporciona SPSS al crear el modelo, la llamada “Residuos I” que corresponde a la

diferencia entre los valores observados y los esperados o estimados que nos proporciona el programa. Para obtener el error aplicamos la fórmula que aparece a continuación:

$$Error = \frac{((Residuos1)' \times Residuos1)}{H}$$

donde H representa al número de elementos del vector que aparece en el numerador, en este caso, el que denotamos por *Residuos1*. Tras ejecutar la fórmula en Matlab nos da el siguiente resultado:

Error = 135.0916

2) SARIMA(1,1,0)(0,1,1)₁₂

$$(1 - B)(1 - B^{12})(1 - \vartheta_1 B)Z_t = (1 - \phi_{12} B^{12})a_t + C$$

$$Z_t = Z_{t-1} + \vartheta_1 Z_{t-1} - \vartheta_1 Z_{t-2} + Z_{t-12} - Z_{t-13} - \vartheta_1 Z_{t-13} + \vartheta_1 Z_{t-14} + a_t + \phi_{12} a_{t-12} + C$$

$$Z_t = Z_{t-1} - 0,297Z_{t-1} + 0,297Z_{t-2} + Z_{t-12} - Z_{t-13} + 0,297Z_{t-13} - 0,297Z_{t-14} + a_t + 0,103a_{t-12} + 0,186$$

Ejecutamos la misma fórmula que anteriormente pero esta vez con los residuos correspondientes al segundo modelo planteado.

$$Error = \frac{((Residuos2)' \times Residuos2)}{H}$$

Error = 136.0458

APÉNDICE 3.C: Identificación del modelo del ejemplo real introducido en la sección 3.4.

Para averiguar los parámetros del modelo habrá que fijarse en las autocorrelaciones. En cuanto a la parte no estacional, corresponde a un modelo de media móvil MA(1) con $q = 1$.

En cuanto a la parte no estacional (véase Peña (2005) pág.207 para detalles) observamos en el correlograma obtenido un MA(1) estacional con $Q = 1$.

Por último, cabe destacar la aparición de muchos modelos que no se encuentran nombrados en nuestro ejemplo pues no cumplen las hipótesis básicas de una serie

temporal. De acuerdo al análisis de los datos, como conclusión, los modelos que cumplen todos los criterios básicos bajo comprobación por el programa SPSS y que son modelos que se ajustan a nuestra serie son el SARIMA (0,1,1)(1,0,0) y (0,1,1)(0,1,1).

DESARROLLO DE LOS MODELOS EN SPSS

Una vez que identificamos los modelos procedemos a comprobar si estos cumplen las hipótesis básicas, con una significación del 5%. Los modelos se pueden representar de la siguiente manera:

1) SARIMA(0,1,1)(1,0,0)₁₂

$$(1 - B)(1 - \vartheta_{12}B^{12}) Z_t = (1 - \phi_1 B)a_t + C$$

$$Z_t = Z_{t-1} + \vartheta_{12}Z_{t-12} - \vartheta_{12}Z_{t-13} + a_t - \phi_1 a_{t-1} + C$$

$$Z_t = Z_{t-1} + 0,493Z_{t-12} - 0,493Z_{t-13} + a_t - 0,624a_{t-1} - 2,334$$

Para calcular el error del modelo utilizamos la tabla que nos proporciona el SPSS llamada “Residuos1” que corresponde a la diferencia entre los valores observados y los esperados o estimados que nos proporciona el programa.

$$Error = \frac{((Residuos1)' \times Residuos1)}{H}$$

Tras ejecutar la fórmula en Matlab nos da el siguiente resultado:

Error = 2.4770e+05

2) SARIMA(0,1,1)(0,1,1)₁₂

$$(1 - B)(1 - B^{12}) Z_t = (1 - \phi_1 B)(1 - \phi_{12}B^{12})a_t + C$$

$$Z_t = Z_{t-1} + Z_{t-12} - Z_{t-13} + a_t - \phi_{12}a_{t-12} - \phi_1 a_{t-1} + \phi_1 \phi_{12} a_{t-13} + C$$

$$Z_t = Z_{t-1} + Z_{t-12} - Z_{t-13} + a_t - 0,673a_{t-12} - 0,591a_{t-1} + 0,397743a_{t-13} - 2,2$$

Ejecutamos la misma fórmula que en el modelo anterior escogiendo los residuos pertenecientes a este modelo.

$$Error = \frac{((Residuos2)' \times Residuos2)}{H}$$

Error = 2.5936e+05

APÉNDICE 3.D: PROGRAMACIÓN EN MATLAB

1. - Programación de los correlogramas

```
hold off
clear rho
clear c
z=input('Introduzca la serie de la se desea representar el correlograma')
numero=size(z)
T=numero(1)
disp('La media muestral es: ')
media=mean(z)
disp('La varianza muestral es: ')
varianza=(z-media)*(z-media)/T

for i=1:(T/4)
    %Autocovarianzas entre zt y z(t-h)
    c(i)=(z(1:T-i)-media)*(z(1+i:T)-media)/T;
end
disp('Vease la grafica del correlograma ')
%Autocorrelaciones
rho=c/varianza;
bar(rho)
hold on
plot(1:T/4,2/sqrt(T),'r*')
plot(1:T/4,-2/sqrt(T),'r*')
hold off
```

2. - Programación de autocorrelación parcial

```
z=input('Introduzca la serie temporal en una columna para calcular su fap')
s=size(z)
T=s(1)
%Obtendremos los coeficientes de autocorrelaciones parcial para los T/4
%primeros retardos.
X=[]
for i=1:T/4
    X=[X,[zeros(i,1);z(1:T-i)]];
    alpha=quadprog(2*X'*X,-2*X'*z,[],[]);
    fapmuestral(i)=alpha(i);
end
bar(fapmuestral)
hold on
plot(1:T/4,2/sqrt(T),'r*')
plot(1:T/4,-2/sqrt(T),'r*')
hold off
```

3. – Tipo de Transformaciones

Transformación logarítmica

>>Y=log(Z); % Aplicación del logaritmo a la serie original (Z).

>>plot(Y) % Gráfica de la serie con la transformación logarítmica

Transformación diferencia regular

>> s=size(Y); % Tamaño de la serie

>> nvariables=s(1); % Conocer el número de datos

>> T=nvariables

>> R=Y-[0;Y(1:T-1)]; % Se aplica la diferencia regular, donde el segundo bloque corresponde a un vector columna que empieza en 0 seguido de la serie original quitando el último número.

>> dif=R(2:T); % La serie diferenciada tendrá un tamaño igual a T-1, ya que no tiene en cuenta el primer dato.

>> plot(dif) % Gráfica de la serie con la diferencia regular

Transformación diferencia estacional

>>difestacional=Z(13:T)-Z(1:T-12);

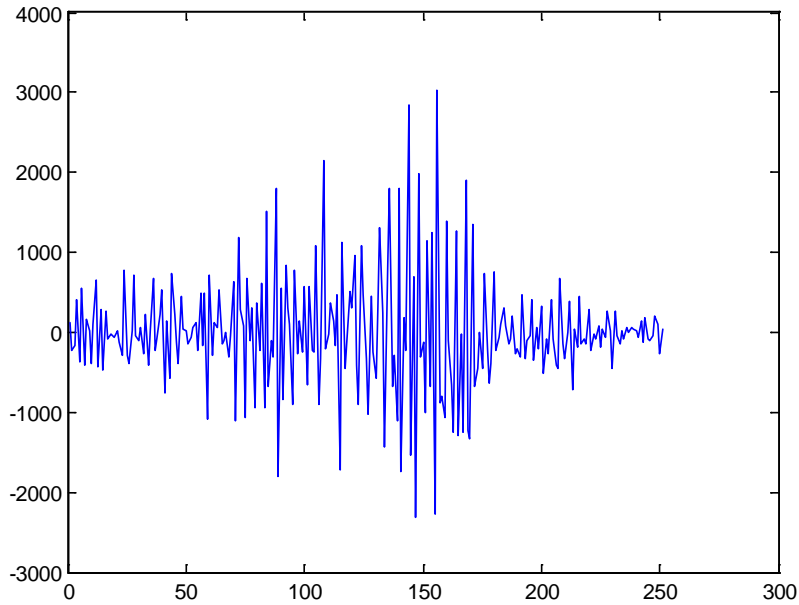
APÉNDICE 3.E: SALIDAS DEL MATLAB

La representación gráfica de los datos de nuestra serie nos hace llegar a la determinación de que nos encontramos ante una serie no estacionaria en media ni en varianza. Primero vamos a corregir la no estacionariedad en media y para ello debemos de realizar una diferenciación regular.

$$R_{hipo} = Y_{hipo} - Y_{hipo-1}, \quad t = 1, 2, \dots, N.$$

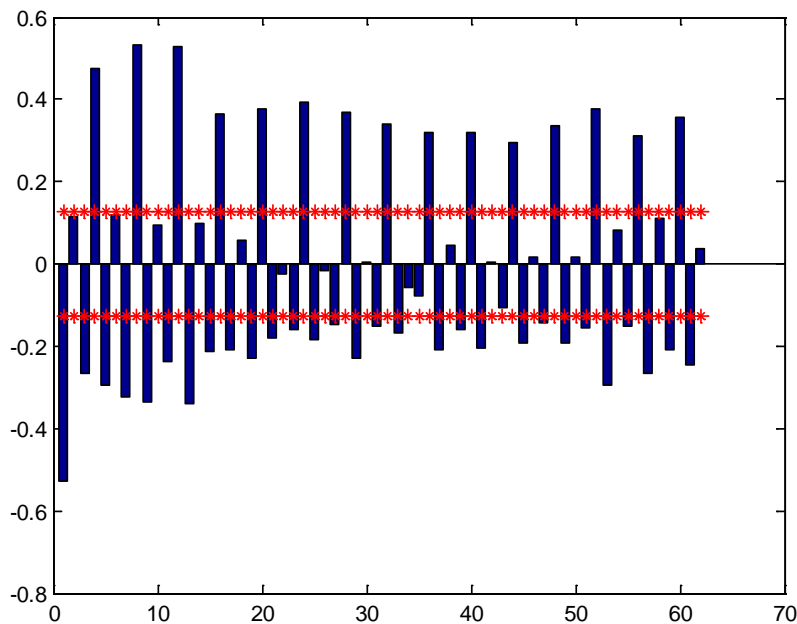
Siendo “hipo”, el nombre establecido para hacer referencia a nuestra serie original (número de hipotecas).

Gráfica 13: Diferencia regular. Elaboración propia



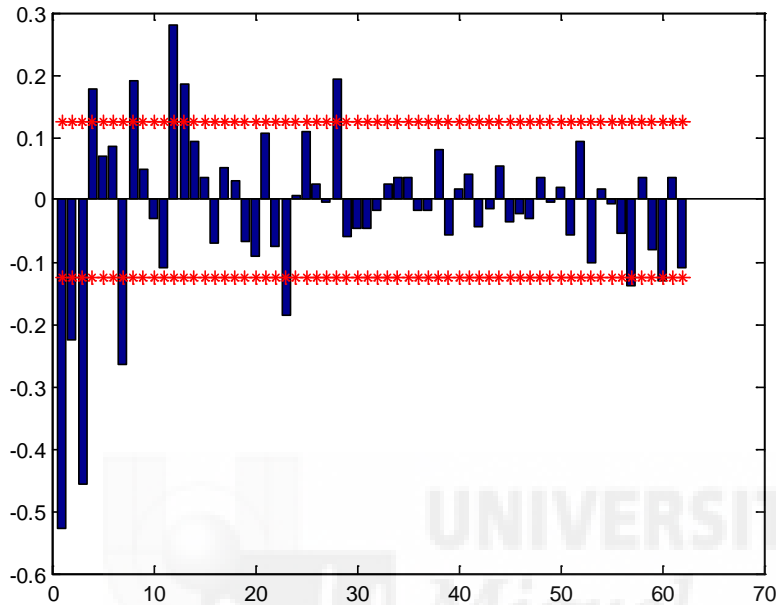
Como podemos apreciar nuestra serie es estacionaria en media, ya que nuestra tendencia es constante en el tiempo. Suponemos que es estacionario en varianza, puesto que la variabilidad de los datos va creciendo a lo largo del tiempo, dicha decisión se ha de tomar para no dejar ningún modelo por el camino. Ahora pasamos a ejecutar nuestras autocorrelaciones simples y parciales del Apéndice 3.B.

Gráfica 14: Autocorrelaciones simples de la diferencia regular. Elaboración propia



Como podemos apreciar en la gráfica los datos presentan comportamientos similares en determinados momentos del tiempo por lo que dicha serie tiene un componente estacional que tenemos que estudiarlo posteriormente.

Gráfica 15: Autocorrelaciones parciales de la diferencia regular. Elaboración propia



Como observamos en nuestras autocorrelaciones, y gracias a la Figura 7.6. del libro de Peña (2005), prestando atención a las explicaciones dadas en el ejemplo anterior de pasajeros de avión podemos llegar a la terminación de que el modelo que debemos de estudiar es $(0,1,1)(1,0,0)$.

Pero este no es el único modelo que podemos sacar de dicha serie, ahora pasamos a conseguir una serie completamente estacionaria en media y varianza aplicando la transformación logarítmica, cuyo resultado se puede mirar en el gráfico 11.

BIBLIOGRAFÍA

[1] M.S.Bazaraa, J.J.Jarvis y H.D.Sherali, Programacion Lineal y Flujo de Redes, Limusa,2005.

[2] D.Bertsimas y J.N. Tsitsiklis, Introduction to Linear Optimization, Athena Scientific,1997.

[3] F.S. Hillier y G.J. Lieberman, Introducción a la Investigación de Operaciones, McGraw Hill, México, 2010.

[4] D. Peña, Análisis de series temporales, Alianza Editorial, 2005.

[5] H.A.Taha, Investigación de Operaciones, Pearson Educación, 2004. URL:

https://books.google.es/books?id=3oHztjMSuL8C&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

[6] A.S. Viejo, La Investigación Operativa: una herramienta para la adopción de decisiones, Universidad Pontificia Comillas (ICAI-ICADE),1996. URL:

https://books.google.es/books?id=sA1dSQko3PAC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false