



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INFORMÁTICOS

UNIVERSIDAD POLITÉCNICA DE MADRID

Modelo dinámico de predicción y ajuste de los precios de las habitaciones de hoteles

TRABAJO FIN DE MÁSTER
MÁSTER UNIVERSITARIO EN INTELIGENCIA ARTIFICIAL

AUTOR: Andrés Herranz González
TUTORES: Alfonso Mateos Caballero y
Antonio Jiménez Martín

2019

Agradecimientos

El presente Trabajo Fin de Máster se enmarca dentro de un proyecto de investigación realizado por el *Centro Tecnológico Mixto AI.nnovation Space* (UPM - Accenture) desarrollado por mis directores, que forman parte del departamento de Inteligencia Artificial de la UPM dentro del grupo de investigación de *Análisis de Decisiones y Estadística* y por mí, joven entusiasta de la Inteligencia Artificial.

En primer lugar, quiero agradecer la labor de mis tutores Alfonso Mateos Caballero y Antonio Jiménez Martínez, por creer en mí desde el primer momento, por todo el esfuerzo realizado y por todo lo que me han enseñado y he podido aprender a su lado.

Por otro lado, a todos los profesores de la Escuela de Informática de la UPM por todo lo que me han enseñado durante el máster, que constituye un hito fundamental en mi formación técnica.

Agradecer en especial a mis padres, que han estado siempre a mi lado, a mis abuelos, que me enseñaron a ser como soy, a mis amigos que siempre están ahí, y a Lore por apoyarme y ayudarme a llegar hasta aquí. Andrés

Resumen

La gestión dinámica de los precios de un producto para venderlo de la manera más adecuada es un campo de estudio de gran interés en el momento actual. En este área surge el concepto de *revenue management* (en adelante, RM), es decir, la capacidad de vender un producto al cliente adecuado, en el momento adecuado, al precio correcto y en el canal correcto.

El RM se ha aplicado a gran cantidad de mercados donde se cuenta con un *stock* limitado y una competencia que ofrece el mismo producto o similar. Este es el caso, por ejemplo, de las reservas de billetes para vuelos, trenes, autobuses. Como posible caso de estudio de este tipo de metodologías, en este proyecto se ha decidido desarrollar un sistema de RM para el sector hotelero.

El trabajo forma parte de un proyecto del *Centro Tecnológico Mixto AI.nnovation Space* (UPM - Accenture) y surge como propuesta para optimizar la gestión de precios de ciertas cadenas hoteleras asociadas con Accenture. Por este motivo se ha decido utilizar este sector, aunque la investigación realizada es aplicable a cualquier campo que tenga unas características similares a las del hotelero.

Este tipo de gestión cuenta con ciertas ventajas frente a los modelos tradicionales: en primer lugar, la automatización del proceso, lo cual permite una respuesta más rápida del sistema en la propuesta de precios en caso de ser necesario; y en segundo lugar, la gestión objetiva y matemática de las incertidumbres, ya que una persona podría caer en errores de juicio o no estimar correctamente los riesgos.

Por todo ello, se va a proponer una nueva metodología de RMS (del inglés, *revenue management system*) que estime el precio que se debe poner para cada tipo de habitación en un cierto horizonte de estudio. La idea es que este sistema contemple las acciones de la competencia y el histórico de nuestro hotel y, a partir de ello, determine los precios óptimos para obtener el mayor beneficio posible. La metodología propuesta se ha basado en el Análisis de Riesgo Adversario (en adelante, ARA) para el estudio de la competencia y del riesgo que asume nuestro hotel, y el uso de otras técnicas para evaluar la evolución de la ocupación en el tiempo.

Finalmente, dado que no se ha podido contar con los datos necesarios para aplicar la metodología propuesta de forma completa, se ha desarrollado un prototipo que aplica algunas de las propuestas presentes en la metodología. Por lo tanto, se han generado módulos independientes que permiten comprobar si los resultados son coherentes

Abstract

Dynamic management of a product's price to sell it in the most adequate manner is a field of great interest nowadays. In this area the concept of revenue management (RM) is important. It describes the capacity of selling a product to the adequate customer, at the optimum time and price and by the correct channel.

The RM has been applied in many markets where there is a limited stock and competition with other companies selling the same or a similar product. An example can be found in the tickets for flights, trains or buses. In this project it has been decided to develop a system of RM for the hotel sector, as a case study for this type of methodologies. This work is part of a project of Centro Tecnológico Mixto AI.nnovation Space (UPM – Accenture), which was originated as a proposal to optimise the price management of some hotel companies associated with Accenture. For this reason, the hotel sector has been chosen, but the research performed in this work can be applied to any other field with similar characteristics.

This kind of management offers some advantages compared to the traditional models. First, the automation of the process, which allows a faster response of the system in the proposal of prices if needed and second, the impartial and mathematical management of the uncertainties, since a person could make judgement errors or not estimate well the risks.

Due to the previous reasons, a new methodology of revenue management system (RMS) is proposed to estimate the price that should be given for each type of room in a defined forecast horizon. The objective is that the system observes the actions of the other companies and the background of our hotel and, based in this data, determine the optimum prices to obtain the maximum possible benefit. The proposed methodology is based in the Analysis of Adversary Risk (Análisis de Riesgo Adversario, ARA) for the study of the competitors and the risk assumed by our hotel; and other techniques have been used to evaluate the evolution of the occupancy rate over time.

Finally, due to the absence of a set of data adequate to fully implement this methodology, a prototype has been developed to apply some of the features proposed. Therefore, independent modules have been generated allowing to evaluate the consistency of the results.

Índice general

Agradecimientos	III
Resumen	V
Abstract	VII
1. Introducción	1
1.1. Objetivos	2
2. Estado del Arte	5
2.1. <i>Revenue management</i>	5
2.1.1. Programación dinámica	7
2.2. <i>Revenue management</i> en el sector hotelero	8
2.3. Técnicas de <i>revenue management</i> aplicadas al sector hotelero	9
2.3.1. Modelos de programación dinámica	9
2.3.2. Modelos <i>multinomial logit</i> basados en las decisiones del cliente	10
2.4. Proceso de <i>revenue management</i> del hotel	11
2.5. Métricas para la evaluación de los beneficios	11
3. Metodología	15
3.1. Modelo de <i>Análisis de riesgo adversario</i>	15
3.1.1. Teoría de juegos y teoría del equilibrio de Nash	16
3.1.2. Formulación del <i>Análisis de riesgo adversario</i>	18
3.1.3. Adaptación del <i>Análisis de riesgo adversario</i> a la industria hotelera	20
3.1.4. Modelo de datos para el <i>Análisis de riesgo adversario</i>	21
3.2. Modelo estimador de demanda	22
3.2.1. Fuentes de información	23
3.2.2. Segmentación de la demanda a lo largo del día	24
3.3. Modelo de regresión	25
3.3.1. Modelo de regresión lineal	26
3.3.2. Métrica de calidad para el modelo	26
3.4. Modelo basado en la ocupación	27
3.4.1. Propuesta y adaptación	29
3.4.2. Ejemplo de ejecución	30

3.5. Optimización de la ecuación	31
4. Arquitectura y funcionamiento del sistema	35
4.1. Limitaciones del sistema	36
4.2. Histórico del sistema: Tablas	37
4.2.1. Configuración de las tablas	37
4.2.2. Estructura de las tablas	38
4.3. Modos de ejecución del sistema	39
4.3.1. Versión preventiva del sistema	39
4.3.2. Versión proactiva del sistema	40
4.4. Eventos del sistema y actualización de tablas	41
4.4.1. Establecimiento de precios al finalizar el día y actualización de tablas	41
4.4.2. Modificación de los precios de la competencia	42
4.4.3. Nueva reserva del hotel	43
4.5. Configuración del sistema	44
5. Caso de uso y desarrollo del prototipo	45
5.1. Entorno de desarrollo	45
5.2. Los datos	46
5.2.1. El dataset original	46
5.2.2. Limpieza y procesado de los datos	47
5.2.3. Estructura final de los datos	52
5.3. Prototipo ARA	52
5.3.1. Modelo de regresión implementado	53
5.3.2. Funciones implementadas	54
5.3.3. Las tablas	54
5.4. Prototipo para la carga de datos	56
5.5. Resultados	56
5.5.1. Resultados del <i>modo preventivo</i>	58
5.5.2. Resultados del <i>modo proactivo</i>	58
5.5.3. Resultados del modelo de regresión	60
6. Conclusiones y líneas futuras de trabajo	63
6.1. Trabajo futuro	64
A. Tablas de ejemplo	67
Bibliografía	69

Índice de figuras

2.1. Proceso de <i>revenue management</i> del hotel.	12
3.1. Diagrama de influencia básico.	18
3.2. Diagrama de influencia con evaluación de riesgos.	19
3.3. Diagrama de influencia para el sector hotelero.	21
3.4. Diagrama de influencia para el sector hotelero (simplificado).	22
3.5. Esquema general de un algoritmo genético.	32
4.1. Arquitectura del sistema.	35
4.2. Establecimiento de precios al finalizar el día.	42
4.3. Modificación de los precios de la competencia.	43
4.4. Nueva reserva del hotel.	43
5.1. Transformación de las reservas en tabla temporal: número de clientes por día.	48
5.2. Adaptación de las tablas temporales a la competencia.	49
5.3. Adaptación de la competencia a las configuraciones de precios.	50
5.4. Unión de todas las tablas para calcular la probabilidad de que los clientes elijan cada hotel.	51
5.5. División de las tablas para el prototipo.	57
5.6. Variación del ARA_{valor} en función al precio de la habitación.	60

Índice de tablas

3.1. Dilema de prisionero.	17
3.2. Ejemplo de clientes por configuración de precios en un día.	24
4.1. Ejemplo de tabla: histórico general	38
4.2. Ejemplo de tabla auxiliar	39
5.1. Clases del dataset original: reservas de hoteles.	47
5.2. Ejemplo de tabla final del prototipo.	52
5.3. Valores de ARA para caso del <i>modo preventivo</i>	58
5.4. Valores de ARA para el caso del <i>modo proactivo</i>	59
A.1. Ejemplo de tabla temporal del prototipo con valores generados por el regresor	67
A.2. Ejemplo de tabla final del prototipo	68

Capítulo 1

Introducción

El *Revenue Management* (en adelante, RM) es uno de los aspectos más importantes en la gestión de cualquier empresa que ofrezca un producto o servicio limitado. Una referencia clave a esta literatura es [56], donde se señala que a pesar del hecho de que las decisiones sobre los precios fijados “suelen ser determinantes y críticas para la rentabilidad, ... las decisiones sobre los precios suelen estar mal administradas (o incluso no gestionadas)”.

Por este motivo, surge el proyecto del *Centro Tecnológico Mixto AI.nnovation Space* (UPM - Accenture), para desarrollar un RMS (del inglés, *revenue management system*) aplicable a este tipo de situaciones que son habituales hoy en día. El proyecto se inicia en el curso académico 2017 / 2018 con un alumno del máster para realizar su Trabajo de Fin de Master y debido a que los resultados no resultaban suficientes, se decide en el curso 2018 / 2019 (curso actual) continuar con el proyecto dándole una nueva perspectiva más cercana al concepto de RMS.

Para trabajar sobre algo concreto y dado que Accenture tiene contactos en gran cantidad de sectores comerciales, desde el principio (en el curso anterior) se decide aplicar el sistema a desarrollar al sector hotelero.

Dentro de este sector se dispone de un número limitado de habitaciones, con diferentes características, se venden a lo largo del tiempo e influye la competencia y sus acciones. Por tanto, se trata de un campo de aplicación idóneo para la metodología que se pretende desarrollar. Además, gracias a los clientes de Accenture, el proyecto puede contar en un principio, con una gran base de datos de hoteles que permite el desarrollo del sistema.

La fuente de datos proviene del proyecto iniciado el curso 2017 / 2018, mencionado anteriormente. El objetivo de este proyecto era generar una estimación de la demanda en un horizonte de estudio concreto utilizando para ello el histórico de las reservas de los hoteles. Para ello, se adoptó un planteamiento basado en series temporales utilizando como histórico las reservas de un hotel concreto, y así estimar el número de reservas que recibiría un día concreto.

Los resultados obtenidos en este trabajo podrían servir en un futuro como complemento para el módulo estimador de demanda. Sin embargo, en el proyecto actual, no se utilizarán sus resultados por motivos de alcance del proyecto.

1.1. Objetivos

El objetivo fundamental del sistema es diseñar un sistema completo y una metodología para la gestión de los beneficios (RM) de un hotel concreto a estudiar. El objetivo del sistema es optimizar los precios y las ventas dentro de un horizonte de estudio para así maximizar los beneficios del hotel.

En las últimas décadas, como ya se ha comentado, se ha extendido la aplicación de las técnicas de RM a la industria hotelera. El objetivo principal del proyecto es generar un sistema a modo de RMS, capaz de calcular y recomendar de forma dinámica los precios óptimos para los diferentes tipos de habitaciones de un determinado hotel.

La idea es que el sistema reaccione ante ciertos eventos de manera que pueda gestionar dinámicamente los precios de cada tipo de habitación del hotel. Esto significa que cuando haya cambios en el entorno, el sistema es capaz de reaccionar y generar unas nuevas recomendaciones para adaptarse y optimizar el beneficio.

La definición del entorno es otra de las principales claves del proyecto. En este enfoque se ha decidido tener en cuenta la competencia y sus cambios en los precios, debido a que este hecho afecta en gran medida a la decisión que tomará el cliente y por tanto, repercutirá en el beneficio final de nuestro hotel. En segundo lugar, para la definición del entorno se ha decidido tener en cuenta la ocupación. El objetivo del trabajo es mejorar el beneficio del hotel en cuestión, y por tanto, el RevPar (del inglés, *revenue per available room*). Esto significa que se debe optimizar la forma en que se venden las habitaciones, es decir, el momento y el precio al que se venden.

El sistema propuesto pretende optimizar los precios en un horizonte de estudio H (que pueda ser configurado por el cliente), dentro del cual las ventas se distribuyen de tal forma que se maximiza el beneficio total. Esto significa que si dentro del horizonte de estudio, se quiere evaluar el día I , habrá que estudiar cómo evolucionará la ocupación dentro de ese *lead time* para que estén correctamente distribuidas las ventas y se maximice el RevPar. Para lograr este objetivo, el sistema cuenta con un estimador de demanda que calcula la demanda esperada en el horizonte de estudio.

Para alcanzar estos objetivos ha sido necesario el estudio del funcionamiento con los hoteles de manera que ha sido posible entender su gestión del riesgo, la distribución de las ventas, los márgenes en los precios, etc. Para ello, además de reuniones con perfiles de Accenture que contaban con conocimiento en el sector, tuvimos una reunión con un experto del sector hotelero que nos dio ciertas pautas sobre el funcionamiento del RM en los hoteles y las principales claves que tienen en cuenta los gestores.

Además, se ha hecho un estudio de la literatura existente sobre RM y la gestión de los precios de los hoteles para estudiar las aproximaciones que se han realizado. Las diferentes técnicas de RM y de DP aplicadas al sector han demostrado mejorar los beneficios. Sin embargo, la mayoría de los estudios dejan partes del problema sin analizar.

En este proyecto se propone utilizar la metodología de Análisis de Riesgo Adversario (ARA), planteada por primera vez en [31], que ha ido evolucionando en los

últimos años y se ha aplicado a diferentes casos de uso. Debido a los prometedores resultados de la metodología y a las carencias de algunos de los anteriores estudios de la literatura, el objetivo es plantear un sistema a modo de RMS capaz de aplicar ARA al sector hotelero.

La metodología que se pretende modelizar debe ser capaz de conectarse con un hotel y un Channel Manager (que nos enviará la información de los precios de la competencia) y actualizarse con estas informaciones. Por lo tanto, otro de los objetivos es crear un sistema de actualización que, partiendo de un formato de datos previamente adaptado por un conector, se encargue de alimentarlo.

Por último, se pretende desarrollar un prototipo que ejecute el modelo y pruebe y verifique que la metodología planteada redunde en un beneficio para el hotel estudiado. Sin embargo, tal y como se explicará más adelante, no se ha podido contar con los datos necesarios para ello. Por tanto, se ha desarrollado un prototipo más reducido que prueba algunos de los componentes propuestos y verifica que se obtienen resultados coherentes. Queda por tanto, una línea futura de trabajo para la aplicación de toda la metodología sobre una base de datos completa para poder probar los resultados y la eficiencia del sistema.

Por motivos de alcance del proyecto, algunos módulos del modelo propuesto no se han desarrollado. Principalmente el módulo estimador de demanda, dado que tiene una gran complejidad y su desarrollo podría ser otro proyecto por si solo. Además, dado que no se ha podido contar con la información de un hotel ni del *channel manager* (que se explicarán más adelante), no se han podido desarrollar los conectores (componente software que transforma los datos de origen en el formato de datos que pueda interpretar nuestro sistema) con estas dos plataformas, dado que no conocemos el modelo origen de los datos ni el método de envío de los mismos.

Capítulo 2

Estado del Arte

Para el proyecto se ha realizado un búsqueda bibliográfica sobre técnicas de optimización de beneficio así como técnicas para la gestión de hoteles y otro tipo de servicios y productos de características similares. El objetivo es comprobar qué técnicas se han utilizado y si las metodologías y resultados obtenidos, cubren las necesidades del sector.

Además, dado que se pretende utilizar un enfoque basado en la ocupación y el análisis de la competencia aplicando la metodología de Análisis de Riesgo Adversario, también se pretende buscar otros enfoques similares a este en la literatura.

Se ha buscado, en primer lugar, información sobre los sistemas de *revenue management* que tradicionalmente se han utilizado para este tipo de escenarios. Y, en segundo lugar, técnicas aplicadas en la industria hotelera (que en realidad, también son sistemas de RM, pero los autores pueden haberlos clasificado como tal, o no).

2.1. *Revenue management*

El *revenue management* (en adelante, RM) [70], también conocido como *yield management*, es una técnica de gestión de herramientas para fijar precios, gestionar reservas y cerrar ventas. Se define como la venta del producto adecuado, al cliente adecuado, en el momento adecuado, al precio correcto y en el canal correcto. Es decir, usar herramientas de análisis, cruzar datos y tomar decisiones para optimizar las ventas y obtener el máximo beneficio.

Para entender mejor los RMS se van a explicar brevemente algunos de los conceptos más importantes con los que se trabaja en este tipo de sistemas y que aparecerán referenciados más adelante.

- **RevPar** (*revenue per available room*): Son los ingresos por habitación disponible.
- **ADR** (*average daily rate*): Se calcula como los ingresos obtenidos de las habitaciones divididos por el número de habitaciones ocupadas.

- **Lead time:** Periodo de tiempo desde que se hace la reserva hasta que llegan los clientes al hotel.

Las técnicas de RM comerciales surgieron en la década de los 60 pero tuvo su auge en la década de los 80, cuando American Airlines se vio amenazada por la entrada del operador de bajo coste PeopleExpress. En [56] se describe:

“En respuesta, American Airlines desarrolló un programa de gestión basado en la diferenciación de precios entre viajeros de placer y de negocios. Un elemento clave de este programa era un sistema de “administración de rendimiento” que usaba algoritmos de optimización para determinar el número idóneo de asientos a guardar para reservas de última hora en cada vuelo mientras seguían aceptando pasajeros con tarifas bajas. Este enfoque fue un éxito rotundo para American Airlines, que resultó en la desaparición definitiva de PeopleExpress.”

En estos momentos, los RMS comerciales son ampliamente utilizados tanto por las aerolíneas como en el sector hotelero y otros debido a la naturaleza similar del problema. Es decir, la reserva anticipada y la asignación óptima de un “inventario” finito y perecedero a los clientes que llegan estocásticamente con una disposición diferente a pagar. Los ejemplos incluyen *IDeaS* [30] (una subsidiaria de *SAS*), *JDA* [36], *PROS* [59] y *Revenue Analytics* [61].

Según [5] “en su nivel más básico, RM se trata de la capacidad de un hotel para segmentar a sus consumidores y el inventario de precios de manera diferente en esos segmentos, en esencia practicando alguna forma de discriminación de precios. En muchos casos, se ha demostrado que el RM utilizada en la industria hotelera aumenta los ingresos de un 2 a un 5 por ciento”.

Los sistemas de RM son propietarios, por lo que no se sabe qué tipo de optimización utilizan y qué tipo de datos y métodos económicos emplean. Tal y como se menciona en [46], el mecanismo que determina los precios de las aerolíneas sigue siendo un misterio y se siguen haciendo investigaciones en este campo dada la importancia de unos buenos resultados que permitan ganar clientes al resto de competidores.

Por otro lado, en [56] se señala que es más probable que las herramientas que utilizan los gestores de los precios de estas agencias a diario estén basadas en técnicas estadísticas y se acredita el marketing (que el autor considera un subcampo de investigación de operaciones y ciencia de la gestión) señalando que el “marketing science” ha llevado algo de ciencia a lo que antes se consideraba un “black art”. Sin embargo, también menciona que el marketing aún tiene diferencias entre el modelo teórico y el práctico. Por ejemplo, algunos de estos modelos se han construido sobre una visión idealizada del comportamiento del cliente y otros parecen estar limitados por suposiciones simplistas y poco realistas.

La literatura relacionada con los RMS contiene muchas ideas prácticas importantes y ofrece principios heurísticos para el RM. Una de las claves en cualquier sistema de RM es la gestión del riesgo, es decir, subir el precio lo máximo posible,

arriesgándose a perder clientes pero ganando lo suficiente con los que reservan. Este aspecto se comenta en [5] y los autores reflexionan acerca del caso contrario, es decir, la posibilidad de que al reducir tarifas no se aumente la ocupación, y por tanto se reduzca el ADR.

Sin embargo, en estos estudios no se menciona una de las herramientas más utilizadas para calcular los precios óptimos dinámicamente: la Programación Dinámica (PD). Existe una gran cantidad de artículos en la literatura [46, 24] de RM y sistemas de gestión que utiliza PD para caracterizar estrategias de precios dinámicos para inventarios perecederos en un horizonte finito. Sin embargo, la mayoría de estos estudios son mayoritariamente teóricos (por ejemplo, [81]) y pocos de ellos proporcionan una evidencia empírica de la calidad de los algoritmos PD en la práctica. En el caso de [82] se llevaron a la práctica este tipo de algoritmos y concluyen que, a pesar de que la mejora en los beneficios es pequeña (aproximadamente 0.19%), estas técnicas pueden suponer una mejora significativa, especialmente si se hace una buena descomposición para PD.

En un estudio reciente [78] se propuso utilizar un modelo de programación dinámica estimado de forma empírica para demostrar que la asignación de precios dinámica frente a la demanda estocástica complementa la discriminación de precios intertemporal en el sector de las aerolíneas. El trabajo determina que, al hacer frente a los cambios en la demanda, las aerolíneas pueden guardar asientos para los clientes de última hora y subirles el precio. Los autores concluyen que las aerolíneas utilizan RMS comerciales que les han ayudado a aumentar sus beneficios.

Otros trabajos recientes, como por ejemplo [32], se han centrado en el cálculo numérico de estrategias dinámicas para el cálculo óptimo de los precios, específicamente para el RM de los hoteles.

2.1.1. Programación dinámica

La programación dinámica (PD) está relacionada con la toma secuencial de decisiones en entornos inciertos. Estas decisiones se toman en etapas y cada decisión, además de ofrecer un beneficio inmediato, modifica el contexto de las decisiones futuras. En consecuencia, esto afecta también los beneficios futuros (y por tanto, a los beneficios totales).

Debido a la naturaleza incierta del entorno, existe escasa información acerca del beneficio inmediato y el estado en el que se encontrará el sistema después de haber tomado una decisión. Por lo tanto, para lograr un buen desempeño en todas las etapas de la toma de decisiones, se tiene que evaluar si es preferible recibir el beneficio de forma inmediata o esperar los pagos futuros, es decir, gestionar el riesgo y las probabilidades asociadas.

La incertidumbre del entorno es modelada por un proceso de Markov, cuyas probabilidades de transición dependen tanto de la información del estado como de la decisión escogida. Estas probabilidades de transición que corresponden a cada par estado-acción, se asumen conocidas para el decisor y el objetivo de este es elegir una política (conjunto de decisiones) que maximice la función de beneficio esperada.

2.2. *Revenue management* en el sector hotelero

Las técnicas de RM se han convertido en un campo de investigación popular en la literatura sobre gestión hotelera. El marco y la evolución de los antecedentes del sector y el uso generalizado de las nuevas tecnologías han permitido adoptar un enfoque orientado al cliente para la asignación de precios y el desarrollo de herramientas de RM; al tiempo que se contribuye a la mejora de los procesos de gestión hotelera. Por lo tanto, los métodos de optimización de precios que buscan maximizar los ingresos del hotel se basan en la escasez de inventario, la segmentación de clientes y los precios.

La primera generación de sistemas RM se basó en gran medida en datos históricos para pronosticar la demanda (sin restricciones) con el fin de optimizar la disponibilidad de los productos. Los hoteles comenzaron a agrupar a sus clientes de acuerdo con los mismos criterios utilizados por las aerolíneas y a segmentarlos en función de la sensibilidad a los precios [70]. Las aportaciones también procedían de los investigadores de la hostelería, que se interesaron cada vez más (a menudo exclusivamente) en los aspectos técnicos de la gestión del riesgo como la previsión, la optimización y la asignación de precios [43].

A modo de ejemplo, basándose en lo observado sobre el comportamiento de las reservas, se hicieron predicciones sobre el nivel de sensibilidad a los precios de los diferentes segmentos del mercado y sus respectivos plazos de entrega. Estos resultados sirvieron de base para nuevas técnicas de segmentación del mercado, que permitieron a los hoteles ofrecer productos que podían satisfacer mejor las necesidades de aquellos segmentos más restringidos, al tiempo que ayudaban a maximizar el rendimiento de cada uno de ellos.

Por otra parte, en épocas de auge o exceso de demanda, el objetivo principal de la gestión de la capacidad era lograr el negocio óptimo para cumplir con el potencial de ingresos del hotel y reducir la probabilidad de tener habitaciones vacías. Esto se logró mediante la introducción de restricciones de reserva, como el requisito de pernoctación en sábado, la reserva anticipada y una duración mínima de la estancia que ayudaron a controlar el inventario y evitar la “canibalización” de las tarifas [11]. Además, los hoteles utilizaban el exceso de reservas como una herramienta para compensar los ingresos perdidos por causa de cancelaciones tardías, ausencias o salidas anticipadas [73].

Gradualmente, el enfoque tradicional basado en los costes para establecer las tarifas de las habitaciones se fue abandonado en favor de tácticas que se basaban en el principio de que, si los ingresos superaban los costes variables de la venta de una habitación, ese excedente se utilizaría para cubrir los costes fijos más altos [39]. Al ofrecer a los hoteles una mayor flexibilidad y una gama de precios más amplia para cobrar diferentes tarifas por el mismo producto a diferentes clientes, la práctica de fijar los precios de las habitaciones se hizo más competitiva y dinámica, y se dio mayor importancia a las peculiaridades de cada segmento de clientes.

No obstante, es evidente que ha sido Internet lo que ha cambiado dramáticamente la forma en que los hoteles comunican sus precios y la disponibilidad de sus

productos. Nuevos actores han entrado en el mercado y proporcionan a los clientes una mayor transparencia en la comparación de productos [80]. Internet permitió a los hoteles actualizar la disponibilidad de productos y precios en tiempo real a través de sistemas de distribución global y agencias de viajes online. Sin embargo, estos avances tecnológicos y tipos alternativos de canales de reserva han significado más costes para los hoteles y una mayor complejidad en el proceso de optimización.

2.3. Técnicas de *revenue management* aplicadas al sector hotelero

El modelo más difundido en la literatura para tipificar el comportamiento de los clientes es la función de demanda. Con este modelo, es posible estimar la voluntad de pago de los consumidores y la elasticidad de precios de la demanda [68].

Los principales modelos de demanda utilizados en la literatura son: (1) funciones de demanda lineal [68, 40]; (2) funciones de demanda no lineal, como la función de demanda Cobb-Douglas [40]; y (3) modelos de demanda de elección del cliente, principalmente *multinomial logit* (MNL) [3].

En [71], los autores clasifican los dos primeros tipos de modelos como un enfoque determinista que es adecuado cuando la sensibilidad del cliente varía a lo largo del tiempo de reserva, y el tercero como un enfoque estocástico, que funciona mejor con las fluctuaciones aleatorias de la demanda y la opción de mantener un cierto número de habitaciones sin vender para el futuro. En [82], los autores comparan el rendimiento de las formulaciones determinísticas y estocásticas de programación dinámica (PD) cuando la demanda y las proporciones de capacidad varían, llegando a conclusiones similares.

Por otro lado, en [26] se señala que la variación del precio de las habitaciones de hotel en el horizonte de estudio tiende a ser no determinista. Sin embargo, en términos prácticos, como los hoteles no suelen variar los precios tan a menudo, la suposición de la tendencia estocástica de PD no es efectiva.

En [70], los autores sitúan el comportamiento de la demanda en el sector de las aerolíneas como no lineal y estocástico. A su vez, en [40] y [41], se señala que la estimación de la función de la demanda es una herramienta poderosa para maximizar los ingresos, ya que permite medir cómo cambia la demanda bajo diferentes condiciones de mercado y estructuras de precios.

Cuando se toman en consideración los diferentes segmentos de mercado, cada segmento está compuesto por individuos con patrones de demanda homogéneos [50]. Una vez que el vendedor es capaz de diferenciar los segmentos, se puede aplicar la discriminación de precios para fijar precios diferentes para cada uno de estos.

2.3.1. Modelos de programación dinámica

El modelo de PD para optimización se enfrenta a situaciones en las que los clientes están dispuestos a pagar un mayor precio por el mismo servicio a medida

que se acerca la fecha de entrada, por lo que el problema es establecer una política de precios capaz de maximizar los ingresos hasta dicha fecha de entrada [8].

Un modelo de PD podría maximizar los ingresos ofreciendo un precio que refleje los niveles actuales de demanda y ocupación [34], por lo que las curvas y detalles de las reservas pasadas son variables útiles para los modelos de PD [32]. El hotel, por tanto, tiene que ocuparse de las fechas de reserva, la información de la demanda y de su disponibilidad de servicio.

En [71], se destacan algunas características de los modelos dinámicos: (1) pueden ser considerados como procesos independientes de la demanda; (2) son viables en mercados competitivos en los que el proveedor es el que pone los precios y el cliente elige en función de su disposición a pagar; (3) buscan la estabilidad en la relación entre precio y demanda; y (4) buscan la simplificación de los procedimientos de optimización de los precios.

En términos prácticos, se determina que la gran mayoría de los hoteles en las capitales europeas utilizan algún tipo de RM basado en PD [2]. El segmento de clientes, la calificación de estrellas del hotel y el número de clientes influyen en la política de precios.

En [26], los autores modelan la variación del precio de las habitaciones de hotel a lo largo del horizonte de reserva para el segmento de negocios con el fin de predecir la mejor estrategia de reserva. De esta manera, descubren que los hoteles en destinos de negocios introducen mayores descuentos a lo largo del horizonte de estudio, especialmente durante eventos especiales o cuando identifican otros factores determinantes como la estacionalidad, la ubicación de los negocios, la clasificación por estrellas, las instalaciones hoteleras y los servicios adicionales.

Un ejemplo de prototipo fue el desarrollado en [7]. En este modelo de optimización de precios los precios se fijan de forma dinámica. Mientras tanto, se utiliza una simulación para estimar las llegadas y las diferentes elasticidades de la demanda. Esto permite fijar un precio óptimo para cada tiempo de reserva.

2.3.2. Modelos *multinomial logit* basados en las decisiones del cliente

En algunos artículos, que se centran principalmente en el sector de las aerolíneas, se describe el tráfico aéreo en su conjunto y en él, cada ruta puede ser operada por diferentes compañías con diferentes horarios de vuelo. El tráfico aéreo está segmentado en diferentes clases, y aunque los cambios de precios influyen directamente en una clase, esto representa sólo una parte de la suma de todos los segmentos [35]. Así cada segmento tiene su propia elasticidad, lo que significa que un cambio de precio en un segmento específico tendrá un efecto directo sobre la demanda en ese segmento, pero a su vez, afectará también al resto (aunque en menor medida). En consecuencia, en el proceso de optimización, a los segmentos restantes se les asignará una cantidad diferente de asientos disponibles, un fenómeno conocido como *efecto de elasticidad cruzada*. Por lo tanto, los modelos de *multinomial logit* MNL consideran la interdependencia de los diferentes segmentos y también son capaces de modelar

la incertidumbre y una amplia gama de comportamientos de los clientes.

Los modelos MNL se utilizan principalmente para gestionar las probabilidades de elección del cliente y medir la utilidad de dichas elecciones basándose en el uso de datos históricos [84]. Los modelos MNL consideran que las demandas individuales se ven afectadas por diferentes atributos conocidos, por lo que sus efectos influyen en las decisiones de los consumidores.

2.4. Proceso de *revenue management* del hotel

En [74] se identifican 8 pasos en el proceso de RM: conocimiento del cliente, segmentación y selección del mercado, evaluación interna, análisis competitivo, previsión de la demanda, análisis y selección de canales, precios dinámicos y gestión de canales e inventario. Las fases del proceso identificadas por los autores se derivan de la práctica general de la gestión de marketing, lo cual es comprensible teniendo en cuenta el hecho de que RM se convirtió en el ámbito de la gestión de marketing.

Por otro lado, en [20] se propuso un esquema con 5 etapas más sencillo y comprensible. La principal ventaja propuesta en [20] es la inclusión de la evaluación cualitativa y la constante monitorización de la estrategia de RM. En este proyecto se ha adoptado una arquitectura de procesos de RM similar a la propuesta en [33] que se estructura como en la Figura 2.1.

Tal y como se explicará más adelante en el Capítulo 4, la estructura del sistema pretende cubrir todas las partes del proceso reflejadas en dicha figura.

2.5. Métricas para la evaluación de los beneficios

En esta sección se van a describir algunos de las métricas clave a tener en cuenta para medir el beneficio obtenido por un hotel y los factores que influyen en los precios. Asimismo, los resultados obtenidos de esta sección ayudarán más adelante en la segmentación de los clientes y la diferenciación entre las tablas.

En [48] se hace referencia a un dicho sobre hoteles: “Cuando se mide el rendimiento, éste mejora. Cuando el rendimiento es medido y reportado, la tasa de mejora se acelera”.

En varias investigaciones, las medidas de gestión de ingresos más comúnmente utilizadas son ADR, RevPar, tasa de ocupación, etc. [29]. La calidad del servicio hotelero y la satisfacción del cliente están influenciadas por ADR [29]. La tasa de ocupación y el ADR son determinantes del desempeño de los hoteles, pero RevPar es una medida superior ya que considera tanto la tasa de ocupación como el ADR [17].

Los ingresos por habitación disponible han surgido como una medida importante, y se aplican como variable dependiente [4] que da la medida del rendimiento de los hoteles. También se considera una medida eficaz que refleja el equilibrio entre la oferta y la demanda de habitaciones de hotel, y se aplica como variable dependiente [79].

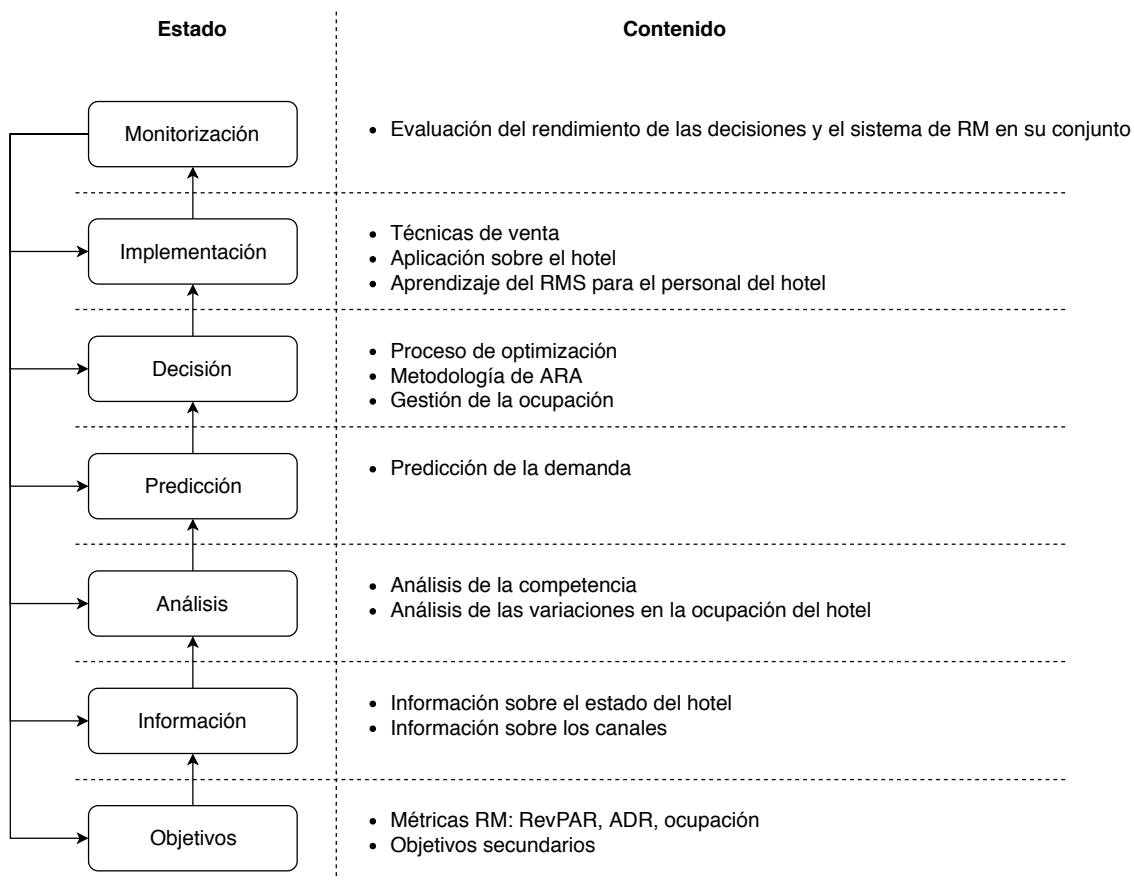


Figura 2.1: Proceso de *revenue management* del hotel.

Las relaciones entre el ADR, la demanda en términos de tasa de ocupación y RevPar son importantes no sólo para determinar la productividad de un hotel, sino también por su uso como puntos de referencia para compararse con los competidores en el mercado hotelero [44]. El crecimiento es un indicador clave del éxito en la industria hotelera. Las métricas clásicas para medir el crecimiento de una firma en el sector hotelera son el tamaño, la edad [53], la ubicación [9], etc.

El precio es uno de los medidores para el crecimiento de una empresa [21]. También se menciona que los precios altos por unidad tienen una influencia positiva en el crecimiento de la empresa. En términos generales, los investigadores sugieren que los precios altos indican una mayor calidad de servicio y, específicamente en la industria hotelera, los precios más altos indican que los clientes están dispuestos a pagar por hoteles de calidad superior [16]. En [21], los autores informaron de que la relación precio-crecimiento en los hoteles suecos no es lineal.

A continuación, se van a enunciar algunos de los factores clave resumidos en [75], que influyen sobre los precios y beneficios de los hoteles:

- Las relaciones entre los factores determinantes de la demanda y la oferta influyen en los precios de los hoteles [47].

- Las tasas de ocupación hotelera influyen positivamente en el precio de las habitaciones [60].
- Los precios difieren ampliamente dentro de la misma ubicación geográfica [72], y los atributos específicos del alojamiento, como el tamaño, la ubicación, la estacionalidad, etc., también influyen en los precios de los hoteles [10].
- Se observa que los precios de los hoteles están influenciados por la competencia local [21].
- Los precios de los hoteles aumentarán en presencia de menos competidores con habitaciones disponibles [1].
- El mayor número de competidores cercanos influye negativamente en los precios de los hoteles [25].
- Los ingresos por habitación disponible son mayores en las localidades con mayor número de hoteles de lujo [15].

Durante el análisis y modelado de la relación entre la oferta (disponibilidad de habitaciones), la demanda (la tasa de ocupación) y el precio por habitación, uno de los principales desafíos que se plantea es que las relaciones entre estas variables probablemente sean no lineales o complejas, lo que causa dificultades en la previsión de ingresos. Por este motivo, y dados los condicionantes arriba mencionados sobre los precios del hotel, en la metodología que se va a proponer se debe resolver el problema de optimización mediante metaheurísticas.

Capítulo 3

Metodología

3.1. Modelo de *Análisis de riesgo adversario*

Como base para el análisis de la calidad de los precios de un hotel, la gestión de la competencia y el manejo del riesgo se ha decidido aplicar una técnica que ha tenido un gran reconocimiento en la literatura en los últimos años, el *análisis de riesgo adversario* [63].

La *teoría de juegos* ha sido considerada durante mucho tiempo poco práctica para la toma de decisiones en la gestión de riesgos [12]. Este punto de vista se ha vuelto menos dogmático recientemente porque:

“Los sectores empresariales clave (especialmente las finanzas, el comercio electrónico y el software) se han vuelto mucho más sofisticados matemáticamente, y ahora están utilizando esta experiencia para dar forma a la estrategia corporativa para la licitación de subastas, el calendario de lanzamiento de productos, el precio de venta de sus servicios y otras decisiones” [45, 65].

“La legislación reguladora debe equilibrar los intereses de las competencias (para el crecimiento, el impacto medioambiental y la seguridad) de manera creíble y transparente” [28].

Estos desafíos están relacionados con muchos campos (Estadística, Economía, Investigación Operativa, Ingeniería, etc.) y se caracterizan por el hecho de que hay dos o más oponentes inteligentes que toman decisiones cuyo resultado es incierto. Colectivamente, a este área se le llama *análisis de riesgo adversario* (en adelante, ARA).

El análisis estadístico tradicional de riesgos creció en el contexto de la seguridad de los reactores nucleares, los seguros y otras aplicaciones en las que las pérdidas económicas se regían por el azar (a veces llamado naturaleza, y descrito como un oponente neutral) más que por las acciones maliciosas (o de interés propio) de los actores inteligentes. Pero en ARA hay que tener un modelo para la toma de decisiones de todos los participantes. Este modelo puede ser el clásico basado en teoría

de juegos, con un equilibrio (no cooperativos) de Nash como concepto central [49] o puede ser más psicológico, reflejando ya sea una formulación de Kadane-Larkey [37] o estudios empíricos de la conducta de juego [14]. En [27] se aportó información adicional sobre la combinación del análisis de riesgos y la teoría de juegos.

Gran parte de la nueva literatura de ARA tiene que ver con la lucha antiterrorista. En [62] se analizaron las estrategias para un ataque de viruela modelando el problema como un juego de suma cero con beneficios aleatorios y resolviendo el juego a través de los enfoques minimax y Bayesiano.

En [83], los autores calcularon las mejores respuestas y el equilibrio de Nash como base para asignar recursos contra el terrorismo cuando el defensor y el atacante tienen diferentes funciones de utilidad, tanto en situaciones de juego simultáneo como secuencial. Por otro lado, en [13] se presentaron modelos de optimización de dos (max-min, min-max) y tres niveles (min-max-min) para problemas de defensor-atacante, atacante-defensor y defensor-atacante-defensor que pueden ser enmarcados como juegos de Stackelberg.

En [38] abogaron por el uso de juegos estocásticos robustos para hacer frente al antiterrorismo. En el caso de [77], entre otros, decidieron adoptar un punto de vista basado en el análisis de decisiones, utilizando árboles de decisión para evaluar las contramedidas de los sistemas portátiles de defensa antiaérea; [57], que aplicó diagramas de influencia para evaluar el despliegue de varias contramedidas a corto plazo; y [54], que utilizó árboles de decisión genéricos multiobjetivo y diagramas de influencia para evaluar las amenazas bioterroristas.

También existe una amplia bibliografía en ciencias políticas sobre la teoría de juegos y el terrorismo, aunque se hace poco hincapié en los aspectos del análisis de riesgos (por ejemplo, [69, 6, 58]).

En el caso de nuestro proyecto se ha utilizado ARA para gestionar la competencia, la toma de decisiones entre los hoteles y el análisis y la gestión del riesgo, dado que a menudo, en la industria hotelera, es importante arriesgar en cierta medida para aumentar el beneficio total.

3.1.1. Teoría de juegos y teoría del equilibrio de Nash

Dado que la metodología de ARA se basa en gran medida en la teoría de juegos (del inglés, *game theory*) y en el equilibrio de Nash, se va a hacer una breve explicación teórica de ambos conceptos.

La teoría de juegos es un área de la matemática aplicada que utiliza modelos para estudiar interacciones en estructuras formalizadas de incentivos (los llamados «juegos»). La teoría de juegos se ha convertido en una herramienta sumamente importante para cuestiones económicas y ha contribuido a comprender más adecuadamente la conducta humana frente a la toma de decisiones. Sus investigadores estudian las estrategias óptimas, así como el comportamiento previsto y observado de individuos en juegos.

El equilibrio de Nash es, en la teoría de juegos, un “concepto de solución” para situaciones con dos o más participantes, el cual asume que:

- Cada jugador conoce y ha adoptado su mejor estrategia, y
- Todos conocen las estrategias de los otros.

En consecuencia, cada jugador individual no gana nada modificando su estrategia mientras los otros mantengan las suyas. Así, cada jugador está ejecutando el mejor “movimiento” posible teniendo en cuenta los movimientos de los demás jugadores.

Teorías aplicadas al dilema del prisionero

Para comprender mejor estas dos teorías se va a explicar cómo se trabajaría con ellas para resolver el dilema del prisionero, una situación muy común cuando se habla de este tipo de problemas con más de un “jugador”. Se trata de un problema de suma no nula.

El enunciado clásico del dilema es el siguiente:

La policía arresta a dos sospechosos de forma conjunta (A y B). No hay pruebas suficientes para condenarlos y, tras haberlos separado, los visita a cada uno y les ofrece el mismo trato. Si A confiesa la culpabilidad y B lo niega, entonces B será condenado a la pena total, diez años, y A será liberado. Si A decide callar y B confiesa, el primero recibirá esa pena y será B quien salga libre. Si ambos (A y B) confiesan, ambos serán condenados a seis años. Si ambos (A y B) lo niegan, todo lo que podrán hacer será encerrarlos durante un año por un cargo menor.

lo que puede resumirse como se muestra en la Tabla 3.1.

Tabla 3.1: Dilema de prisionero.

A / B	A confiesa	A lo niega
B confiesa	6 / 6	10 / 0
B lo niega	0 / 10	1 / 1

Con esta situación, la teoría de juegos analizaría los resultados y mediante simulaciones, por ejemplo, establecería el comportamiento de dos prisioneros hasta llegar a una estabilidad.

Esta estabilidad en el dilema del prisionero es el equilibrio de Nash en estrategias puras: se produce cuando ambos jugadores confiesan. A pesar de ello, “ambos confiesan” es peor que “ambos lo niegan”, en el sentido de que el tiempo total de cárcel que deben cumplir es mayor. Sin embargo, la estrategia “ambos lo niegan” es inestable, ya que un jugador puede mejorar su resultado desertando si su oponente mantiene la estrategia de negar lo ocurrido. Así, “ambos lo niegan” no es un equilibrio de Nash pero sí un óptimo de Pareto [22]. Una manera de llegar a ese resultado es logrando un pacto y mediante la promesa de cada jugador de penalizar al otro si rompe el acuerdo. También podría llegarse a una solución fuera del equilibrio de Nash si el juego se repitiese infinitas veces, cuando se utiliza la estrategia “ojo por ojo”.

3.1.2. Formulación del *Análisis de riesgo adversario*

El marco de trabajo de ARA utiliza diagramas de influencia para estructurar los problemas que son populares en las comunidades de análisis de decisiones e inteligencia artificial y que abarcan juegos simultáneos y secuenciales (en los que también se utilizan árboles de decisión). Por simplicidad, se asume que las pérdidas pueden traducirse como costes. Se supone que todos los agentes participantes buscan maximizar su utilidad esperada [23].

La Figura 3.1 representa un diagrama de influencia [55] que muestra la versión más simple de un problema de gestión del riesgo no adversario. Un diagrama de influencia es un diagrama acíclico dirigido con tres tipos de nodos: nodos de decisión, representados como rectángulos; nodos de incertidumbre, representados como óvalos; y nodos de valor, que se muestran como hexágonos. Las flechas apuntando a un nodo de valor o incertidumbre indican dependencia funcional y probabilística, respectivamente. Por lo tanto, la función de utilidad en el nodo de valor depende de sus nodos inmediatamente anteriores, y las probabilidades en un nodo de incertidumbre están condicionados a los valores de sus predecesores directos. Las flechas en un nodo de decisión indican que cuando la decisión se ha tomado, se conocen los valores de los nodos que le preceden.

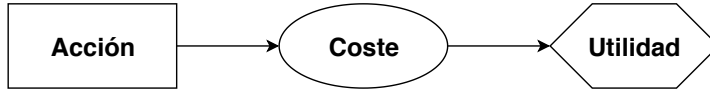


Figura 3.1: Diagrama de influencia básico.

Supongamos que el rectángulo de la Figura 3.1 representa el conjunto A de posibles decisiones o acciones, el óvalo representa los costes aleatorios asociados a las decisiones, y el hexágono representa las consecuencias netas, o valores, en términos de la utilidad de la persona que toma las decisiones. Esta situación se corresponde a un problema en el cual una organización tiene que tomar una decisión a partir de un conjunto de opciones A .

El coste c que resulta de cada decisión es incierto y se modela a través de la función de densidad $\pi(c|a)$; este coste puede reflejar el hecho de que el resultado de una decisión en particular es incierto, o que el coste asociado con un resultado en particular es incierto, o ambos. La utilidad $u(c)$ del coste es una función decreciente y es típicamente no lineal, es decir, a mayor coste, peor utilidad. Se busca la decisión que maximice la utilidad esperada:

$$\psi = \max_{a \in A} \left[\psi(a) = \int u(c) \pi(c|a) dc \right]. \quad (3.1)$$

En la práctica, determinar el coste de una acción en particular es complejo y depende del resultado; a menudo incluyen sumandos fijos y aleatorios. Cuando una organización se enfrente a esta situación, normalmente realizará una evaluación de riesgos para:

- Identificar los eventos de alto impacto E_1, E_2, \dots, E_k (se puede suponer que son mutuamente excluyentes);
- Se evalúa la probabilidad de que ocurra cada uno de ellos, $P(E_i|a) = q_i(a)$;
- Finalmente, se evalúa el coste c_i condicionado a E_i cuando se toma la decisión a (estos costes son típicamente aleatorios y la evaluación puede ser una distribución).

En la Figura 3.2 se muestra el diagrama de influencia que amplía la formulación anterior para incluir una evaluación de riesgos que tenga en cuenta los peligros de alto impacto y los costes aleatorios adicionales asociados a estos.

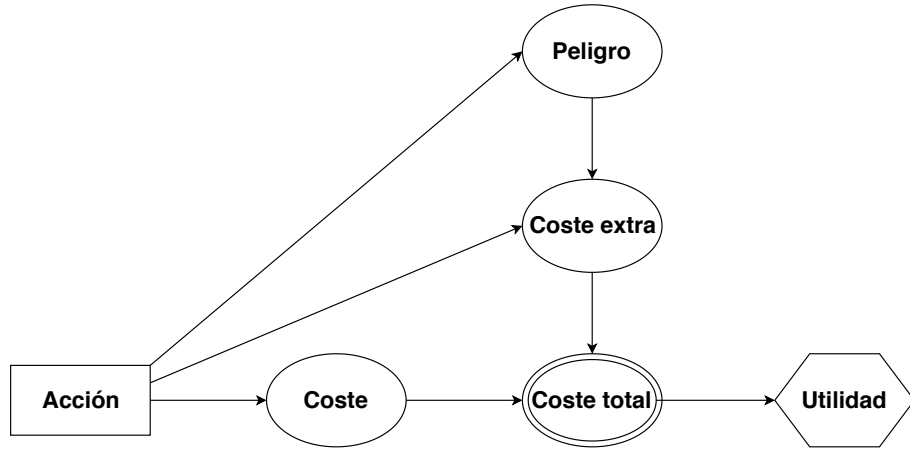


Figura 3.2: Diagrama de influencia con evaluación de riesgos.

Sea $q(a)$ el vector de probabilidades correspondiente a la decisión a y $\pi_i(c|a)$ la densidad de coste condicionado a la decisión a si ocurre el evento E_i . Por tanto, la densidad de coste asociado a la decisión a será $\sum_{i=0}^k q_i(a)\pi_i(c|a)$. Una vez realizada la evaluación de riesgos, se quiere obtener la decisión que maximiza la utilidad esperada, lo cual se consigue resolviendo:

$$\psi_r = \max_a \left[\psi_r(a) = \sum_{i=0}^k q_i(a) \int u(c) \pi_i(c|a) dc \right]. \quad (3.2)$$

Una vez obtenida esta expresión, los autores continúan desarrollándola y explicando su evolución para obtener las ecuaciones finales de ARA. Sin embargo, dado que este proceso resulta complejo de comprender, los propios autores proponen una casuística con la cual explican el funcionamiento de *framework* de ARA para un problema sencillo.

Se trata de un pequeño problema en el cual Apollo (atacante) pretende atacar a Daphne (defensora). Daphne tiene que tomar una decisión dentro de un conjunto finito de posibles decisiones $D = d_1, \dots, d_m$ y simultáneamente, Apollo debe elegir

su decisión $A = a_1, \dots, a_n$. Daphne y Apollo reciben como resultado Y_D y Y_A , respectivamente, que dependen de ambas acciones (d, a) y (en general) de un resultado aleatorio ω .

La utilidad esperada por Daphne, asociada a una pareja de acciones $(d, a) \in D \times A$ es:

$$\psi_D(d, a) = \int u_D(d, a, \omega) p_D(\omega|d, a) d\omega, \quad (3.3)$$

donde $u_D(d, a, \omega)$ representa la utilidad de Daphne con un beneficio $Y_D(d, a, \omega)$, y $p_D(\omega|d, a)$ representa la probabilidad de obtener ω como salida del sistema cuando se toman las acciones (d, a) . En caso de querer representar la utilidad esperada de Apollo, su expresión sería análoga a la de Daphne.

Siguiendo con este ejemplo [64], los autores evolucionan la expresión y la amplían y generalizan para diferentes tipos de oponentes y casuísticas. Por ejemplo, el caso de oponentes que no sigan ninguna estrategia, oponentes que busquen el equilibrio de Nash o incluso oponentes que utilicen el mismo método de ARA.

Finalmente, para obtener la defensa que maximiza la utilidad esperada, se llega a la siguiente expresión:

$$d^* = \arg \max_d \sum_a \left[\int u_D(d, a, \omega) p_D(\omega|d, a) d\omega \right] p_D(a), \quad (3.4)$$

en la cual, $p_D(a)$ representa la probabilidad que tiene Apollo de tomar la decisión a , según Daphne. Esta probabilidad, dada la incertidumbre, se suele representar como una distribución.

3.1.3. Adaptación del *Análisis de riesgo adversario* a la industria hotelera

Una vez obtenida la Expresión 3.4, se puede transformar la fórmula para que represente el problema de la competencia hotelera en el cual nuestro hotel se enfrenta a otros para conseguir los clientes. El sistema es extensible a cualquier número de hoteles competencia, sin embargo, en la representación del problema sobre un diagrama de influencia, los hoteles competencia aparecerán como una única entidad (para mayor sencillez en la explicación).

Por tanto, el nuevo diagrama de influencia que representa el problema hotelero tiene el diseño mostrado en la Figura 3.3.

En la figura, H y H_0 representan nuestro hotel y la competencia, respectivamente, u_H y u_{H_0} representan la función de utilidad de nuestro hotel y la de la competencia, también respectivamente, y por último, c representa la salida del sistema.

Como se puede observar, se trata de una versión simplificada del modelo en la cual no se han incluido los riesgos que aparecen en la Figura 3.2. Sin embargo, estos están implícitos y se tienen en cuenta en la formulación de la expresión para la utilidad esperada.

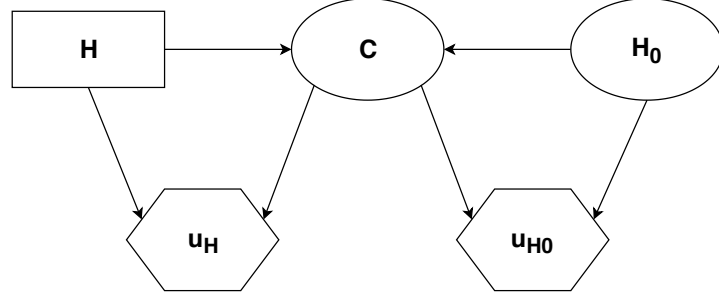


Figura 3.3: Diagrama de influencia para el sector hotelero.

La Expresión 3.4 ha sido adaptada al problema de gestión de precios dado que no todos sus componentes son iguales. Por un lado, dado que se conocen las dos únicas salidas posibles del sistema (que el cliente elija nuestro hotel o que elija a la competencia), no es necesario integrar y se puede resolver como un sumatorio. Por otro, dado que el sistema contará con más información que la que se menciona en la literatura sobre ARA, no es necesario estimar $p_D(a)$, sino que se puede representar como una distribución de probabilidad.

La nueva expresión queda de la siguiente forma:

$$p^* = \arg \max_{p \in H} \sum_{\vec{p}_0 \in H_0} \left[\sum_{c \in \{0,1\}} u_H(p, c) p_H(C = c | p, \vec{p}_0) \right] \pi_H(H_0 = \vec{p}_0), \quad (3.5)$$

donde $u_H(p, c)$ representa la utilidad de nuestro hotel de que un cliente reserve o no a un precio p (se asume que esta utilidad es 0 si el cliente no reserva con nosotros); $p_H(C = c | p, \vec{p}_0)$ representa la probabilidad de que un cliente elija nuestro hotel dado el precio que hemos puesto nosotros y los precios de la competencia; y $\pi_H(H_0 = \vec{p}_0)$ representa la probabilidad de que la competencia ponga los precios \vec{p}_0 . \vec{p}_0 se representa como vector ya que representa el conjunto de precios de la competencia.

Con esta nueva expresión, se evalúan, para cada uno de nuestros posibles precios, todas las combinaciones de precios de la competencia y todas las posibles salidas (que son únicamente dos). Dado que el trabajo se centra en maximizar el beneficio de nuestro hotel (la utilidad esperada), el diagrama de influencia se puede resumir como se muestra en la Figura 3.4.

En ella únicamente se muestran los componentes que influyen sobre nuestra utilidad esperada.

3.1.4. Modelo de datos para el *Análisis de riesgo adversario*

Este modelo de ARA sirve a nuestro *framework* como indicador de calidad para un determinado precio y nos permite seleccionar la mejor elección en cada situación (teniendo en cuenta a la competencia).

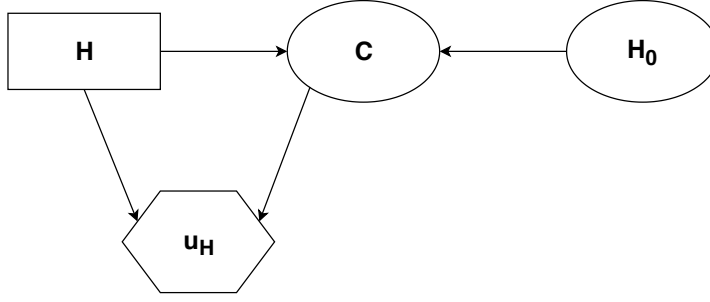


Figura 3.4: Diagrama de influencia para el sector hotelero (simplificado).

Sin embargo, para que el sistema funcione es necesario contar con una fuente de información y un modelo de datos concreto. Esto se explicará más adelante en el Capítulo 4, pero para entender bien la metodología se va a comentar someramente la información necesaria para el funcionamiento del sistema.

En primer lugar, es necesario segmentar las ventas entre diferentes características para distinguir los diferentes tipos de situaciones en las cuales se puede encontrar el modelo. Para ello se pueden distinguir, por ejemplo, el tipo de habitación, la temporada, el día de la semana, etc. Estas divisiones se almacenarán por separado en tablas.

En segundo lugar, es necesario contar con un histórico que recoja las diferentes configuraciones de precios que han aparecido para cada tabla y las probabilidades asociadas de que un cliente elija nuestro hotel frente a la competencia (se obtendrá dividiendo los clientes que eligen en un día nuestro hotel entre la demanda esperada para ese día). Esto también genera la necesidad de un estimador de demanda. Con estos datos se podrá calcular la probabilidad $p_H(C = c | p, \vec{p}_0)$.

En tercer lugar, es necesario saber el número de días (dentro de cada tabla) que los hoteles competencia han puesto cada uno de los precios. De esta manera, se podrá estimar la probabilidad $\pi_H(H_0 = \vec{p}_0)$.

Toda esta información se almacenará en las tablas anteriormente mencionadas, las cuales irán creciendo a medida que avancen los días y que el sistema obtenga más información. Dado que estos datos sirven como base directa para el modelo de ARA, cuanto más información haya, mejor será el cálculo de las probabilidades de la Expresión 3.4, y por tanto, mejor será la optimización del precio.

Con estos datos, el sistema es capaz de ejecutar diversas funcionalidades y optimizar los precios que debe poner el hotel en el horizonte de estudio. Dichas funcionalidades se explicarán con más detalle en el Capítulo 4.

3.2. Modelo estimador de demanda

El modelo estimador de demanda es una de las piezas clave para el correcto funcionamiento de la metodología. Sin embargo, dada la magnitud de este componente, no se ha desarrollado para el trabajo actual y podría ser, por si solo, un proyecto para el futuro.

Su función principal es, dado un día de estudio en el futuro I , generar una estimación de la demanda para cada tipo concreto de habitación en los i días que comprenden el periodo entre el día actual e I . Es decir, este modelo tiene que ser capaz de distribuir la demanda que tendrá un día concreto I en el futuro, entre los días que hay hasta ese momento.

De esta manera, la demanda total que tendrá un día concreto I será:

$$D_I = \sum_{i=0}^I d_{I,i}, \quad (3.6)$$

donde $d_{I,i}$ representa la demanda o número de clientes que reservarán habitaciones de un tipo concreto el día i para alojarse el día I .

Dado que en el proyecto planteado se pretende abarcar un horizonte de estudio M , el módulo estimador de demanda deberá generar los M conjuntos de valores D_I para cada tipo de habitación.

Para medir esta demanda es necesario contar con ciertas fuentes de información, tal y como se explicará en la Sección 3.2.1, y con otros módulos auxiliares que ayuden a medir la variabilidad frente a diferentes sucesos. Para ello, se pueden utilizar aproximaciones similares a la adoptada por el Trabajo de Fin de Máster que se hizo en el curso 2017 / 2018. Utilizando series temporales, o métodos de analítica sobre los datos del histórico, se puede medir la variabilidad de los eventos y factores que influyen en la demanda y así determinar en el futuro cómo afectarán las nuevas circunstancias.

Por último, como se explicará en la Sección 3.2.2, el módulo estimador de demanda debe ser capaz de distribuir la demanda $d_{I,i}$ (al final del día) entre las franjas horarias en las que ha habido cambios en los precios. De esta manera, se puede obtener una estimación de la distribución de la demanda a lo largo del día.

3.2.1. Fuentes de información

Dado de que este módulo solo se va a utilizar de manera teórica y no se va a desarrollar, no se ha profundizado en los requisitos del mismo. Sin embargo, se ha hecho un estudio (apoyado por reuniones con expertos del sector hotelero) para determinar algunos de los factores clave que intervienen en la demanda.

A continuación, se enumeran algunos de los datos que tienen mayor relevancia:

- Eventos concretos que sucedan en la ciudad o en los alrededores del hotel estudiado. Por ejemplo, un concierto o un partido podrían generar un aumento/pico en la demanda de habitaciones de hotel. Además, en función de la tipología del evento, se puede discriminar fácilmente qué tipo de habitación tendrá mayor aumento de demanda.
- La climatología también afecta en gran medida a la demanda de un día concreto. Si el sistema se conectara a fuentes de información meteorológica, se podría obtener otro buen indicador.

- El número de reservas en los aviones (autobuses, trenes, etc.) es otro indicador directo de la demanda. Tal y como hemos podido comprobar, existen empresas que agregan esta información y la comercializan para acceder mediante API. Con estos datos, se puede tener una correlación directa con la demanda de un día concreto.
- La época del año que se está evaluando, el día de la semana o las fiestas regionales o nacionales pueden influir también positiva o negativamente en la demanda.

Estos son algunos de los factores que se deberían tomar en cuenta a la hora de implementar este modelo para poder generar, con la mayor precisión posible, una estimación de la demanda. Además, tal y como se ha explicado anteriormente, el uso del histórico para medir la variación de la demanda frente a este tipo de eventos y condicionantes y los márgenes de error.

3.2.2. Segmentación de la demanda a lo largo del día

Dado que los precios de las habitaciones pueden cambiar a lo largo del día (ya sea en nuestro hotel o en la competencia), se necesita saber cómo se distribuyen las ventas de las habitaciones en estos periodos. Por este motivo, es necesario que el módulo estimador de demanda sea capaz de segmentar la demanda $d_{I,i}$ entre los periodos temporales.

Dado que la demanda no se vende de manera uniforme, es necesario estimar cómo se han distribuido las ventas (en nuestro hotel y en los hoteles competencia) a lo largo del día. Para ello, el modelo se deberá basar en el histórico y en el análisis de las tendencias en ventas para poder terminar dicha distribución.

El objetivo principal es poder discernir qué demanda ha obtenido cada configuración de precios (combinación del precio de nuestro hotel con los precios de la competencia) que ha aparecido a lo largo del día para poder actualizar correctamente las tablas.

Un ejemplo para entender mejor su funcionamiento podría ser la siguiente situación. Supongamos un hotel que ha recibido los siguientes clientes (representados con la variable $t_{clients}$) con las diferentes configuraciones de precios que se indican en la Tabla 3.2, y que la demanda $d_{I,i}$ de clientes que van a comprar habitación hoy para el día I es de 30 personas (reservas).

Tabla 3.2: Ejemplo de clientes por configuración de precios en un día.

p	p_{01}	p_{02}	$t_{clients}$
100	110	90	8
100	110	120	3
120	130	130	1

Cuando termine el día y se disponga de los datos recogidos en la Tabla 3.2 se deberá calcular el número de reservas global que se ha hecho con cada configuración de precios, es decir, la demanda que ha tenido cada configuración.

El motivo principal de este cálculo es el siguiente: para obtener la probabilidad de que un cliente elija nuestro hotel dada una configuración concreta de precios, tenemos que dividir el número de clientes que han reservado habitación en nuestro hotel entre la demanda. Sin embargo, en el caso de ejemplo, si dividimos cualquiera de los números, por ejemplo el 3, entre la demanda (30) obtendremos una probabilidad $Outcome_{prob}$ de $\frac{3}{30} = 0.1$, lo cual no es representativo de la realidad.

Por este motivo, es necesario almacenar el momento en que se han hecho los cambios entre las configuraciones de precios. Por ejemplo, el precio (100, 110, 90) se puso a las 00:00 y ha durado hasta las 11:00 de la mañana; el precio (100, 110, 120) se puso a las 11:00 y duró hasta las 18:00; y el precio (120, 130, 130) se puso a las 16:00 y se mantuvo hasta el final del día (las 24:00). Con esta información sobre los horarios, el módulo estimador de demanda debe ser capaz de dividir las 30 reservas entre los 3 periodos, en base al histórico, obteniendo por ejemplo una distribución de: (100, 110, 90) \rightarrow 20, (100, 110, 120) \rightarrow 7 y (120, 130, 130) \rightarrow 3.

Con estos nuevos datos se puede estimar de manera más acertada el valor de $Outcome_{prob}$. Por ejemplo, en el caso de (100, 110, 120) con 3 clientes, y la nueva estimación de demanda, 7, se puede estimar la probabilidad de que los clientes elijan nuestro hotel: $\frac{3}{7} = 0.42$. Este resultado es coherente y denota que al tener menor precio que la competencia, obtenemos mayor número de clientes; esto es, un 42 %.

3.3. Modelo de regresión

El modelo de regresión es uno de los principales en la metodología. Su función es estimar las probabilidades (de que un cliente elija nuestro hotel frente a la competencia) de aquellas configuraciones de precios que no aparezcan en la tabla sobre la que se está trabajando.

Tal y como se verá en la Sección 4.3, la metodología propuesta tiene un modo de ejecución proactivo, el cual reacciona ante un cambio en los precios de la competencia y genera un nuevo precio óptimo para cada día del horizonte de análisis M . Dentro de esta funcionalidad, al ejecutar el modelo de ARA previamente explicado en este capítulo, se deben evaluar todas las posibles configuraciones de precios. Sin embargo, dado que en este caso se conocen los precios de la competencia, se debe optimizar el precio que debe poner nuestro hotel para cada día.

Por este motivo, es necesario aplicar el modelo de ARA sobre todas las configuraciones de precios que contengan todos nuestros posibles valores y el precio actual de la competencia. Sin embargo, es muy probable que no todos estas configuraciones (y por tanto, sus correspondientes probabilidades de que un cliente elija nuestro hotel frente a la competencia) hayan aparecido anteriormente (en el histórico). Por este motivo, para aquellas configuraciones que no se tengan registradas, será necesario estimar las probabilidades.

Para estimar estas probabilidades se ha decidido utilizar un modelo de regresión que, utilizando datos históricos, estime los datos restantes. Se ha decidido utilizar un modelo lineal dado que al estar las reservas segmentadas en muchas tablas, el factor que más influye en la variabilidad de la probabilidad de que un cliente elija nuestro hotel es el precio, y por tanto, suele seguir una tendencia lineal. Aún así, una posible mejora a futuro podría ser el estudio de otros modelos de regresión y su calidad.

3.3.1. Modelo de regresión lineal

Dado que en la metodología propuesta se quiere evaluar cualquier número de hoteles competencia N , se tienen que tener en cuenta $N + 1$ precios (hay que contar con el precio de nuestro hotel). Por tanto, se ha utilizado una regresión lineal en la cual la variable dependiente Y es la probabilidad de que un cliente elija nuestro hotel frente a la competencia y las variables independientes o explicativas \vec{X}_t son cada una de las diferentes configuraciones de precios. El término aleatorio ϵ se estima en el entrenamiento. La función de regresión utilizada es:

$$Y_t = \beta_0 + \beta_1 \times X_{t1} + \beta_2 \times X_{t2} \dots \beta_N + 1 \times X_{tN+1} + \epsilon. \quad (3.7)$$

Por lo tanto, a la hora de estimar los parámetros $\beta_1, \beta_2, \dots, \beta_{N+1}$, estos se calcularán a partir del conjunto de variables independientes de \vec{X}_t que representan las diferentes configuraciones de precios. Por tanto, el conjunto variables \vec{X}_t en el entrenamiento se sustituirá por las diferentes configuraciones de precios que se estén evaluando.

3.3.2. Métrica de calidad para el modelo

Para medir la calidad del modelo de regresión es necesario el uso de una métrica que evalúe la precisión del modelo. Por esto motivo se ha hecho un búsqueda bibliográfica sobre algunas de las métricas más utilizadas.

Finalmente, la métrica de error utilizada para evaluar la bondad del modelo ha sido el *coeficiente de determinación* R^2 , la cual toma valores en el intervalo $(-\infty, 1]$, donde 1 representaría una predicción perfecta sobre los datos. Aunque resulte extraño que el coeficiente R^2 pueda tomar valores negativos, esto se debe a

que la fórmula para calcularlo contiene una resta. Su expresión es:

$$\bar{y} = \frac{1}{n} \sum_{t=1}^n y_t, \quad (3.8a)$$

$$SS_{tot} = \sum_t (y_t - \bar{y})^2, \quad (3.8b)$$

$$SS_{res} = \sum_t (y_t - f_t)^2, \quad (3.8c)$$

$$R^2 \equiv 1 - \frac{SS_{res}}{SS_{tot}}. \quad (3.8d)$$

donde \bar{y} representa la media de los valores y (probabilidades en este caso) observados, y_t el valor real de la probabilidad para el caso t y f_t el valor de la función de regresión en el caso t del conjunto de datos, es decir, el valor predicho por el modelo de regresión.

3.4. Modelo basado en la ocupación

Para la correcta gestión de los precios de los diferentes tipos de habitaciones de un hotel es necesario estudiar la ocupación del mismo y su evolución a lo largo del tiempo. Por este motivo, se ha decidido evolucionar la metodología inicial de manera que se incorpore este análisis.

Para ello, se ha hecho una búsqueda bibliográfica y se ha encontrado un estudio [7] en el cual se propone un análisis de la ocupación en un cierto horizonte. La idea propuesta en el artículo parte de una estimación del precio de las habitaciones y de las reservas asociadas a dicho precio que se esperan para cada día hasta el día que se está estudiando. En base a la variación de la ocupación en relación al precio, los autores proponen el estudio progresivo de estos datos para la optimización del precio de la habitación.

Tal y como se expone en [7], cuando el número de habitaciones decrece (hay nuevas reservas), el precio de la habitación debe crecer. Por tanto, el estudio pretende calcular cómo se deben distribuir las reservas a lo largo de los días y el precio que se debe asignar cada día para conseguir maximizar el beneficio, todo esto intentando que las últimas habitaciones se vendan el último día. El método de optimización tiene como restricción que el sumatorio de las reservas de cada día hasta el día que se está analizando no puede superar la capacidad restante de habitaciones de las que se dispone para dicho día.

Para calcular la ocupación esperada del hotel, se propone el uso de la siguiente expresión:

$$o = o_{nominal} \left(\frac{p}{p_{nominal}} \right)^e, \quad (3.9)$$

donde $p_{nominal}$ representa el valor medio o típico del precio de la habitación del hotel estudiado y $o_{nominal}$ representa la ocupación (número de reservas) esperada si las habitaciones se venden al precio $p_{nominal}$. La variable e , también denominada *elasticidad*, representa la relación entre el precio y el número de clientes. Un valor típico propuesto por los autores es $e = -2$, de manera que a medida que aumente el precio, disminuyen las reservas de manera exponencial. Por último, p representa el precio concreto que estamos evaluando.

Sin embargo, hay que tener en cuenta que en el modelo propuesto no se parte del precio nominal, sino que se quiere utilizar tanto el precio de nuestro hotel como los de la competencia. Además, la estimación de la ocupación se ha realizado dividiendo la demanda de forma idéntica entre todos los hoteles, dado que la diferencia en el número final de clientes vendrá indicado por la Expresión 3.9. Por estos motivos se ha adaptado la fórmula de manera que represente el escenario deseado y calcule el número de clientes que elegirán nuestro hotel frente a la competencia.

Esta relación viene definida en la fórmula en base al valor de e y a la diferencia entre los precios de los hoteles. Para este nuevo escenario, la fórmula se adapta de la siguiente manera:

$$o = \frac{D}{N+1} \left(\frac{p}{\frac{p + \bar{p}_0 * N}{N+1}} \right)^e. \quad (3.10)$$

donde D representa la demanda estimada para el día bajo estudio, N el número de hoteles que conforman la competencia y p el precio de nuestro hotel.

Por otro lado, $\frac{D}{N+1}$ representa la ocupación esperada de nuestro hotel, \bar{p}_0 el precio medio de la competencia, y $(p + \bar{p}_0 * N)/(N+1)$ el precio medio de todos los precios, es decir, el valor que se correspondería con $p_{nominal}$. Por este motivo, en adelante:

$$\overline{p_{nom}} = \frac{p + \bar{p}_0 * N}{N+1}. \quad (3.11)$$

En la Expresión 3.10, al igual que en la propuesta en el artículo, e representa lo “bueno” que es nuestro hotel frente a los hoteles de la competencia, dado que cuanto más pronunciada sea la curva (mayor sea el valor de e), peor será nuestro hotel.

Este parámetro, como ya se ha explicado, se establece por defecto en $e = -2$ pero, dado que se va a contar con una etapa de entrenamiento para el correcto funcionamiento del modelo, e es un parámetro que se puede ajustar en función de las características de nuestro hotel frente a la competencia. Para ello, obteniendo muestras de la demanda, el precio y la ocupación, se puede resolver la ecuación con e como única incógnita. De esta manera, calculando el valor de e durante varios días se podría estimar un nuevo valor (mediante la media o mediana o métodos similares) y ajustar mejor el modelo.

3.4.1. Propuesta y adaptación

Dado que el modelo que se está desarrollando pretende tener en cuenta la competencia, sus precios y las variaciones de los mismos, se ha combinado la metodología propuesta en [7] junto con la metodología de *Análisis de Riesgo Adversario*. Con ello, se pretende llegar a un método de predicción de precios a M días vista donde se optimice el precio para cada día por separado.

El objetivo del proyecto, como ya se ha comentado, es optimizar los precios que se deben poner el día de estudio (el día actual) para cada uno de los siguientes M días (siendo M el horizonte de estudio). Para ello, la metodología propuesta optimizará cada uno de los días individualmente y evaluará, para cada día, los precios que se deben poner en los días que comprenden el periodo entre el día actual y el día evaluado. Por tanto, para cada día evaluado I se obtendrá un conjunto de precios de tamaño i (equivalente al número de días entre el día actual y el día evaluado, $0 \leq I \leq M$), pero solo se utilizará el precio que se debe poner el día actual (en el que se ejecuta el modelo), ya que el resto de valores obtenidos corresponden a los precios que se debe poner en los siguientes días.

El beneficio obtenido por un hotel en un día concreto dependerá de lo ganado con cada reserva y el número de reservas. Por tanto, la función para medir el beneficio total obtenido en un día i debería ser de la forma:

$$B_i(p) = U_i(p) \times O_i(p). \quad (3.12)$$

En esta función, el beneficio óptimo B_i para un día i se obtiene multiplicando la utilidad (o calidad) de la reserva $U_i(p)$, que es relativa al precio, por la ocupación que obtiene el hotel $O_i(p)$ cuando pone las habitaciones a un precio p .

Sin embargo, dado que queremos optimizar el beneficio del día actual y de los días restantes hasta llegar al día evaluado, se debe utilizar el conjunto de precios $\vec{p}_I = (p_{I,0}, p_{I,1}, \dots, p_{I,i}, \dots, p_{I,I})$ que representa los precios que se deberían poner cada día hasta el día que se está evaluando. Es decir, el precio $p_{I,i}$ representa el precio que pondríamos el día i para reservar una habitación para el día I , lo que significa que se estaría calculando el precio de $I - i$ días de antelación.

La fórmula adaptada quedaría de la siguiente manera:

$$\begin{aligned} & \arg \max_{\vec{p}_I} && U(\vec{p}_I) \times O(\vec{p}_I) \\ & \text{s.a} && \sum_{i=0}^I O(p_{I,i}) \leq C_I \end{aligned} \quad (3.13)$$

De esta manera, el conjunto \vec{p}_I debe optimizarse para obtener el mayor beneficio tomando únicamente precios que estén dentro del rango de precios H del hotel y sin superar la capacidad C_I de habitaciones que quedan para el día evaluado.

Dado que \vec{p}_I representa al conjunto de precios de los días restantes hasta el día I , el problema se debe ajustar a la secuencia. Por este motivo el problema se puede ver como el sumatorio de lo sucedido cada día. La Expresión 3.13 reformulada como

sumatorio de los beneficios de cada día i hasta el día de estudio I quedaría de la siguiente forma:

$$\begin{aligned} \arg \max_{\vec{p}_I} \quad & \sum_{i=0}^I U(p_{I,i}) \times O(p_{I,i}) \\ \text{s.a} \quad & p_{I,i} \in H \quad \forall p_{I,i} \in \vec{p}_I \\ & \sum_{i=0}^I O(p_{I,i}) \leq C_I \end{aligned} \quad (3.14)$$

Por lo tanto, el problema final a optimizar requiere buscar el conjunto de valores $\vec{p}_I = (p_{I,0}, p_{I,1}, \dots, p_{I,i}, \dots, p_{I,I})$ de tal manera que se consiga la mayor utilidad (beneficio) y que la ocupación no exceda la capacidad del hotel para el día de estudio C_I . La fórmula final, desarrollada uniendo la Expresión 3.14 con las dos fórmulas propuestas para ARA y la ocupación queda de la siguiente forma:

$$\begin{aligned} \arg \max_{\vec{p}_I} \quad & \sum_{i=0}^I \left(\sum_{\vec{p}_0 \in H_0} \left(\sum_{c \in \{0,1\}} u_H(p_{I,i}, c) p_H(C = c | p_{I,i}, \vec{p}_0) \right) \times \right. \\ & \left. \times \pi_H(H_0 = \vec{p}_0) \right) \frac{D_{I,i}}{N+1} \left(\frac{p_{I,i}}{\overline{p_{nom}}} \right)^e \\ \text{s.a} \quad & p_{I,i} \in H \quad \forall p_{I,i} \in \vec{p}_I \\ & \sum_{i=0}^I O(p_{I,i}) \leq C_I \end{aligned} \quad (3.15)$$

donde \vec{p}_0 es un vector dado que representa al conjunto de precios de la competencia. Por tanto, su tamaño será igual al número de hoteles competencia, es decir, N . $D_{I,i}$ representa la número de clientes que se espera que reserven el día i habitaciones para el día I , es decir, con $I - i$ días de antelación.

Esta fórmula final, nos dará el conjunto de precios \vec{p}_I . Como ya se ha comentado, únicamente utilizaremos el primero de sus valores, es decir, el correspondiente al precio que se debe poner el día de estudio (el día actual). Para entender mejor su funcionamiento, se va a enunciar un ejemplo de uso.

3.4.2. Ejemplo de ejecución

Supóngase un hotel que dispone de todos los datos necesarios y quiere establecer en el día actual el precio para los próximos 30 días, es decir, a cuánto va a vender una habitación para el día actual, para el día siguiente, para dos días después y así sucesivamente hasta el trigésimo día. Dado que dispone de un sistema que implementa esta metodología, va a establecer cada uno de estos precios individualmente, es decir, va a ejecutar el sistema una vez por cada uno de los 30 días.

Supongamos ahora, que dentro de estos 30 días, quiere analizar qué precio poner para que una habitación se reserve hoy para dentro de 4 días.

La metodología propuesta debe optimizar el conjunto de precios $\vec{p}_4 = [p_{4,0}, p_{4,1}, p_{4,2}, p_{4,3}, p_{4,4}]$ de manera que $p_{4,0}$ representa el precio que se debería poner hoy para una habitación a ocupar dentro de 4 días, $p_{4,1}$ el precio que se debería establecer mañana para dicha habitación y fecha, y así sucesivamente. Esto significa que se ejecutará el algoritmo de optimización enunciado en la Ecuación 3.15 y se evaluará cada $p_{I,i} \in \vec{p}_I$.

En el caso, por ejemplo, del precio $p_{4,0}$ se deberá acceder a la tabla de 4 días de antelación (además del resto de divisiones que se hayan hecho de las tablas) para resolver la parte de ARA ($p_H(C = c | p, \vec{p}_0)$) y se comprobará el precio que ha puesto la competencia (\vec{p}_0) ese día para la parte relativa a la ocupación ($\frac{D_{I,0}}{N+1} \left(\frac{p_{4,0}}{p_{nom}} \right)^e$). En el caso del precio $p_{4,1}$ se accederá a la tabla de 3 días de antelación y se utilizarán los mismos precios de la competencia que antes; y así sucesivamente, hasta comprobar todos los precios y obtener el conjunto óptimo.

Tras finalizar la ejecución y obtener el conjunto óptimo \vec{p}_4 , el único valor que se utilizará es el primer elemento $p_{4,0}$, dado que es el precio que debemos poner hoy para la habitación a ocupar dentro de 4 días. El resto de valores se desecharán.

3.5. Optimización de la ecuación

Como ya se ha explicado, para resolver la Expresión 3.15 es necesario un método de optimización. Dado que la resolución de este problema de manera exacta podría suponer un elevado coste computacional, en este apartado se propone su resolución mediante el uso de metaheurísticas [18, 51, 76].

El problema tiene un único objetivo (la Expresión 3.15) que consta de dos funciones relacionadas entre sí. Por otro lado, el número de variables depende del horizonte M de cálculo que se esté valorando y su valor, dado que está discretizado, es el conjunto finito H de posibles precios que puede poner nuestro hotel. Por estos motivos, el problema resulta ideal para utilizar metaheurísticas que con una probabilidad muy alta aseguran un buen resultado.

Un posible algoritmo para la resolución del problema podría ser el *algoritmo genético* [19]. Esta metaheurística poblacional genera un grupo de individuos (*cromosomas*) que se cruzan y evolucionan hasta cumplir alguna de las condiciones de parada u obtener un resultado “bueno”.

Además, se ha decidido la utilización del algoritmo genético dado que su desarrollo es sencillo, es ampliamente personalizable dado que cuenta con gran cantidad de parámetros [66] y por último, porque se ha demostrado a lo largo de los años su versatilidad y su calidad en la resolución de problemas de optimización, por ejemplo, en el caso de problema del viajante [67].

El esquema general de este algoritmo se representa en la Figura 3.5.

En el caso del modelo, los cromosomas tendrían como genes a cada uno de los precios para los I días que se estén evaluando en el momento de la ejecución (en cada

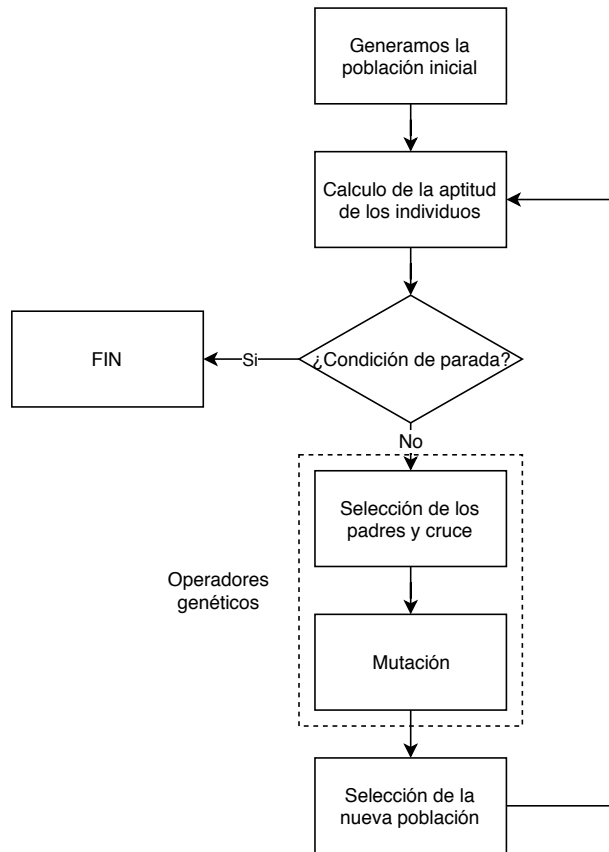


Figura 3.5: Esquema general de un algoritmo genético.

ejecución, I sería diferente y, por tanto, también lo sería el tamaño del cromosoma). Las diferentes fases se ejecutarían de la siguiente manera:

- **Generación de la población inicial:** En esta fase se genera, de manera aleatoria, la población inicial, es decir, el conjunto de individuos de la que partirá el algoritmo.
- **Calculo de aptitud:** En esta fase se evalúa la aptitud de cada uno de los individuos. Para ello, utilizando la Expresión 3.15 se obtiene un valor que sirve como medida de “calidad”.
- **Condición de parada:** Esta condición comprueba si se ha alcanzado algún individuo que cumpla con los requisitos de “calidad” o si se ha alcanzado un máximo de tiempo de ejecución o de iteraciones. Si no se cumplen las condiciones, se seguirá ejecutando el algoritmo; en caso contrario, terminará. En caso de no haber terminado, se almacena el mejor individuo encontrado hasta el momento (se compara con el mejor antiguo, y en caso de mejorarlo, se modifica el mejor individuo global encontrado). Esto se debe a que si se termina la ejecución por el tiempo o el número de iteraciones, este valor almacenado será el que devuelva el algoritmo.

- **Selección de los padres y cruce:** En esta fase se seleccionan de manera aleatoria parejas de individuos que se cruzan entre sí. En el cruce, ambos cromosomas intercambian parte de sus genes.
- **Mutación:** En esta fase, los individuos mutarán sus genes de manera aleatoria y con cierta probabilidad preestablecida. Esto significa, que algunos de los precios del conjunto \vec{p}_I se modificarán de manera aleatoria dentro del rango de precios H del hotel. Una vez terminados los operadores genéticos, se debe calcular la aptitud de los nuevos individuos de la población.
- **Selección de la nueva población:** En esta última etapa se seleccionarán los individuos que conformarán la nueva población para volver a ejecutar el proceso. Esta nueva población tiene que ser del mismo tamaño que la población inicial, es decir, cada vez que se inicie una nueva iteración del algoritmo, el número de individuos de la población tiene que ser el mismo. En esta fase se puede usar elitismo o no. Esta técnica consiste en preservar los mejores individuos de la generación anterior para la siguiente. El elitismo favorece la convergencia y hace que sea más rápida, y por otro lado reduce la exploración. Dado que en el caso de los hoteles es importante dar una respuesta rápida, se recomienda el uso de elitismo, es decir, seleccionar los mejores individuos para que formen parte de la siguiente generación.

Al finalizar el proceso, se dispone del conjunto \vec{p}_I , óptimo o cercano al óptimo, de precios para los próximos I días. Por tanto, nos quedaremos con el primer gen del cromosoma y desecharemos los demás. Este valor será el que se configure como nuevo precio ($p_{I,0}$) de la habitación estudiada.

Capítulo 4

Arquitectura y funcionamiento del sistema

El sistema está estructurado a modo de aplicación, de manera que se pueda ejecutar automáticamente o de manera manual si el usuario lo requiere. Su componente principal es el módulo de decisión ARA, el cual se encarga de recomendar el precio óptimo de los distintos tipos de habitación que tiene el hotel. Este módulo de decisión está conectado, tal y como se muestra en la Figura 4.1, a otros tres módulos auxiliares.

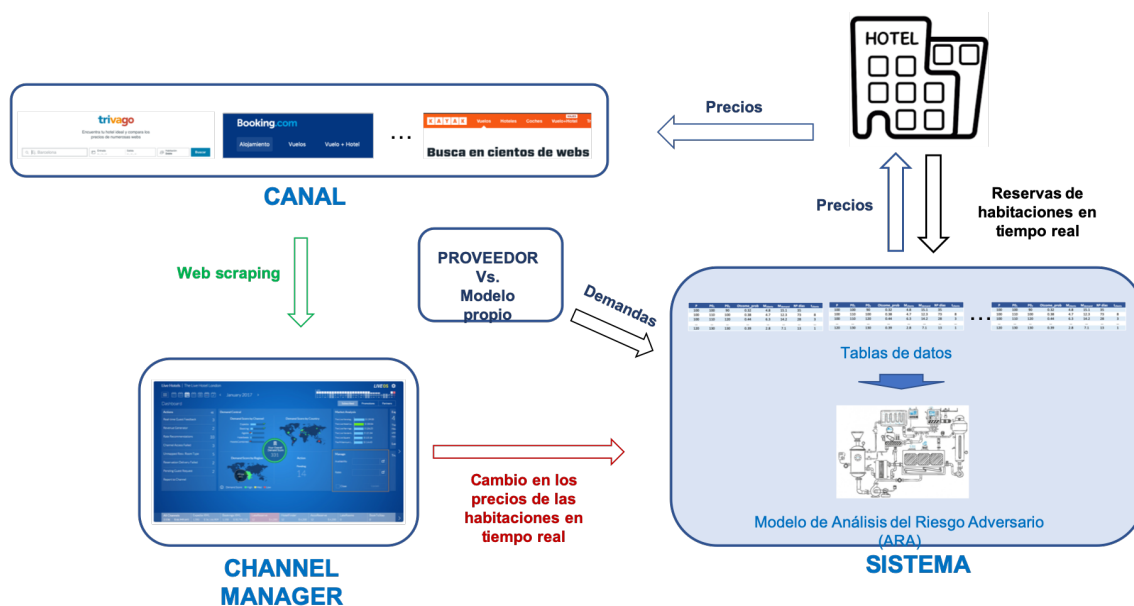


Figura 4.1: Arquitectura del sistema.

Estos módulos auxiliares son necesarios para obtener los datos que utilizaremos en el módulo de decisión. Se explican a continuación:

- **Channel Manager:** Este módulo proporciona los datos de la competencia en tiempo real. Su función principal es actualizar el sistema con los cambios

en los precios de la competencia. Así pues, activa la *versión proactiva* (véase la Sección 4.3.2) del modelo para ajustar de nuevo el precio de nuestro hotel en función de nuestros intereses y de los cambios realizados por la competencia. Este módulo necesita, además, un conector que transforme los datos a la estructura que se necesita y envíe los datos ya modelados al sistema.

- **Canal:** Hace referencia a las páginas web como *Booking*, *Trivago* y otras que, mediante *Web Scrapping* (técnica informática para el escaneo de páginas web), obtienen los precios que están poniendo los hoteles para cada habitación, tipo de alojamiento, días de antelación, etc. Por lo tanto, a través del canal, los *channel manager* obtienen los precios de los hoteles identificados como competencia del nuestro.
- **Reservas de nuestro hotel en tiempo real:** Para el correcto funcionamiento del sistema es necesario que nuestro hotel nos informe en tiempo real de las reservas que recibe. Esto es importante dado que, si se cubre un cierto cupo de reservas, se tiene que volver a actualizar el precio de la habitación, es decir, debemos ejecutar la *función proactiva* del sistema (dado que se sabe el precio actual de la competencia). Además, la información de las reservas obtenidas es necesaria para actualizar las tablas al final del día, ya que con esta información se actualiza la probabilidad de que los clientes elijan nuestro hotel. Para actualizar estas probabilidades es necesaria la estimación de la demanda por tipo de habitación proveniente del módulo estimador de demanda.
- **Estimador de demanda:** Su función principal es generar una estimación de la demanda por tipo de habitación dentro del horizonte de estudio M . Además, este módulo es capaz de dividir la demanda total entre ciertas franjas horarias dadas (aquellas en las que ha habido cambio de precios), de manera que se pueda distribuir la demanda a lo largo del día. Por último, con la estimación de la demanda y las reservas obtenidas por el módulo anteriormente comentado, se actualizan las tablas.

Tal y como se puede ver, todo ello constituye un sistema que, de manera autónoma, se actualiza y va generando conocimiento para mejorar las predicciones del sistema de decisión.

4.1. Limitaciones del sistema

El sector hotelero presenta una gran cantidad de variables en lo que respecta a las reservas: número de noches de la reserva, promociones del hotel, campañas publicitarias que puedan aumentar las ventas, reservas para familias, ofertas de grupo, etc. Además, el precio de una habitación de un hotel se debería estudiar como una variable continua dado que debería poder tomar cualquier valor.

Por todos estos motivos, ha sido necesario tomar una serie de limitaciones en la metodología desarrollada:

- Las reservas que se contemplan en la metodología son de una sola noche, y aquellas que sean más largas se dividirán en los días individuales y se evaluarán como tal.
- Las reservas serán individuales para cada tipo de habitación, de decir, solo se tendrá en cuenta el tipo de habitación y no el número de personas que la reservan.
- No se contemplarán reservas en grupo que podría aplicar reducciones de precio.
- No se contemplarán reservas con promociones u ofertas.
- Se contemplarán las ventas realizadas por canales de terceros y sus diferentes precios, pero no se tendrán en cuenta sus promociones, únicamente el precio estándar de la habitación.
- El sistema ha sido diseñado con un conjunto discreto de posibles precios para las habitaciones, por tanto, el precio de las habitaciones no será interpretado como un conjunto continuo sino discreto. Esto es necesario para poder llevar a cabo la optimización, pero la granularidad y amplitud del conjunto discreto es variable.
- Se asume que no existe variación en los precios entre los diferentes canales de venta (Booking.com, Trivago, etc).

Varias de estas limitaciones podrían resolverse en futuras versiones del sistema, pero por motivos de alcance del proyecto no se han tenido en cuenta.

4.2. Histórico del sistema: Tablas

En esta sección se va a explicar la estructura final de los datos y las tablas que se utilizan a modo de histórico para alimentar el sistema. El dato de la demanda no se almacena en las tablas sino en variables del sistema que se actualizan con los datos del módulo estimador de demanda. Por ello, no se hablará de la demanda en esta sección.

4.2.1. Configuración de las tablas

Tal y como se ha explicado en la metodología, el sistema de decisión basado en ARA requiere de cierta información acerca de los hoteles y sus reservas. Algunos de los datos más importantes son: las combinaciones de precios de nuestro hotel y la competencia y la probabilidad, asociada a cada configuración, de que los clientes elijan nuestro hotel frente a los de la competencia.

Dado que un hotel puede tener varios tipos de habitación, diferentes tipos de pensiones, vender las habitaciones con diferentes días de antelación, etc; es necesario modelar unas probabilidades diferentes para cada tipo. Además, también hay

que tener en cuenta si las reservas se hacen entre semana o en fin de semana, en temporada alta o baja, y otras posibles valoraciones.

Por este motivo, es importante distinguir qué factores y tipos se van a tener en cuenta y así generar todas las posibles combinaciones que pasarán a constituir las tablas. De esta manera, por ejemplo, en caso de haber utilizado 30 días de antelación, distinguir entre semana y fin de semana, temporada alta y baja, y 3 tipos de habitación diferente, se tendrían $30 * 2 * 2 * 3 = 360$ tablas diferentes.

4.2.2. Estructura de las tablas

Con el modelo de datos final proveniente del *Web Scrapping* y las reservas en tiempo real del hotel, la estructura de las tablas es similar a la del ejemplo adjunto (que se explicará en el Capítulo 5) puesto que las columnas que se van a utilizar para la ejecución del modelo de decisión son las mismas (el precio de nuestro hotel, el precio de la competencia y la probabilidad de que los clientes elijan nuestro hotel). Sin embargo, el modelo final que se muestra en la Tabla 4.1, cuenta con otras columnas que permitirán actualizar las tablas al finalizar el día, de tal manera que el sistema pueda aprender de lo que va sucediendo.

Tabla 4.1: Ejemplo de tabla: histórico general

p	p_{01}	p_{02}	$Outcome_{prob}$	$M_{clients}$	M_{demand}	Nº days	$t_{clients}$
100	100	90	0.32	4.8	15.1	35	
100	110	100	0.38	4.7	12.3	73	8
100	110	120	0.44	6.3	14.2	28	3
...
120	130	130	0.39	2.8	7.1	13	1

La información esencial que requiere el modelo esta contenido en las siguientes columnas:

- p . El precio de nuestro hotel, el cual recibiremos cuando tengamos una reserva.
- p_{01}, p_{02} . El precio de la competencia que habremos obtenido mediante el *channel manager*.
- $Outcome_{prob}$. La probabilidad de que los clientes elijan nuestro hotel, que habremos obtenido como resultado de dividir el número medio de clientes por día que vienen con nosotros entre la demanda media de clientes por día (calculada por el módulo estimador de demanda y dividida entre las franjas horarias donde ha habido cambios de precio). Estos precios medios están calculados para cada entrada de las tablas.

El resto de columnas son auxiliares y permiten al sistema actualizar correctamente las probabilidades:

- $M_{clients}$. El número medio de clientes por día que han elegido nuestro hotel.
- M_{demand} . La demanda media por día, dado que al dividir el número medio de clientes entre la demanda obtendremos la probabilidad de que los clientes elijan nuestro hotel.
- $N^o\ days$. El número de días que ha aparecido cada configuración de precios.
- $t_{clients}$. Una columna contador que almacena el número de reservas recibidas en el día actual, con cada configuración de precios.

Además, para calcular la probabilidad que tiene la competencia de poner cada uno de sus precios, es necesario saber cuántas veces ha puesto dichos precios. Es por esto que son necesarias unas tablas auxiliares, como la Tabla 4.2, en las cuales se recoja, para cada precio de la competencia, el número de días que se ha puesto. Por este motivo, para cada tabla de las que tenemos identificadas y por cada hotel competencia, se generará una tabla auxiliar.

Tabla 4.2: Ejemplo de tabla auxiliar

p_{0_i}	Nº days	$Today_{used}$
100	84	False
110	28	True
130	13	False
...
90	19	False

La columna $Today_{used}$ se utiliza para señalar aquellos precios que han aparecido en el día de estudio. Una vez termina el día y se actualizan las tablas principales, también se actualiza el número de días que ha aparecido cada precio.

4.3. Modos de ejecución del sistema

El sistema propuesto puede reaccionar ante diferentes eventos y se ejecutará de manera diferente en cada uno de ellos. Por este motivo, el sistema tiene dos modelos de optimización de precios: el primero en el cual no se conocen los precios de la competencia y solo queremos utilizar el historial (*versión preventiva del sistema*); y el segundo en el que se conocen los precios de la competencia y se ajustan los precios de nuestro hotel para no quedarse por detrás (*versión pro-activa del sistema*).

4.3.1. Versión preventiva del sistema

La *función preventiva* del sistema consiste en calcular cuál es el precio óptimo que debemos poner sin saber los nuevos precios que pondrá la competencia para el nuevo día. Para ello, utilizaremos únicamente el histórico de lo que hemos ido aprendiendo.

Esta función se ejecutaría, por ejemplo, todos los días a las 24:00, de tal manera que se pueda empezar el día ajustando el precio óptimo de cada habitación utilizando únicamente los datos aprendidos. Aún así, una vez se reciba la primera actualización del *channel manager* del día, se ejecutará la *versión proactiva*.

Con el objetivo de optimizar los precios se ejecutará el modelo una vez para cada tipo de habitación utilizando únicamente los datos disponibles en las tablas. Por este motivo, cuanto más tiempo haya pasado, las tablas tendrán una mayor cantidad de datos y la recomendación de precios será mas fiable.

Dado que en el planteamiento de la metodología se necesitan los precios de la competencia y este modo de ejecución omite esa información del *channel manager*, para ajustar las variables de la Expresión 3.15 se van a utilizar los siguientes datos:

- En el caso de la parte relacionada con $U(\vec{p}_I)$ que es en esencia la fórmula de ARA, se utilizarán todas las entradas recogidas en las tablas correspondientes (en caso de estar analizando el día I , habrá que acceder a todas las tablas que abarcan desde el día actual hasta el día I) que contengan cada valor p que se esté evaluando. Es decir, se utilizarán todas las configuraciones de precios de p que han aparecido en el histórico de cada tabla.
- En el caso de la parte de la ocupación $O(\vec{p}_I)$, dado que el software se ejecuta omitiendo los datos en tiempo real de la competencia, se debe ajustar el valor de $\overline{P_{nom}}$. En este caso, dentro de la Expresión 3.11 en vez de utilizar como precio de la competencia \vec{p}_0 el valor actual, se utilizará, para cada hotel competencia, el precio que tenga más probabilidad de ser usado (obteniendo esta información de las tablas auxiliares que contienen el número de días que han puesto cada hotel sus precios).

Con la fórmula reajustada se puede obtener el conjunto de precios \vec{p}_I y ajustar los M precios que se deben poner.

4.3.2. Versión proactiva del sistema

Esta función del sistema se ejecuta cuando recibimos una actualización en los precios de la competencia para cada uno de los M días que se están evaluando. Una vez obtenida esta información se debe ejecutar de nuevo el modelo para cada uno de los M días. Para cada día I evaluado hay que seguir el siguiente proceso:

- Para cada uno de los i días hasta el día I que se está evaluando se debe acceder a la correspondiente tabla.
- Una vez disponible la información de la tabla correspondiente, se debe comprobar si aparecen todas las configuraciones de precios de nuestro hotel junto a los precios que está poniendo la competencia en ese momento, es decir, comprobar que cada precio de nuestro rango H aparece junto a los precios exactos que tiene la competencia. Aquellas configuraciones que aparezcan en la tabla se almacenarán y el resto se generarán con el regresor. Esto generará una nueva tabla auxiliar similar a la del Anexo A.1.

- Esta tabla auxiliar es la que se utiliza en la primera parte de la Expresión 3.15 para resolver la calidad de un precio.
- En el caso de la parte relacionada con la ocupación, se utiliza el precio evaluado y cada iteración del algoritmo. Como \bar{p}_0 se utiliza la media de los precios de la competencia que se han recibido mediante el *channel manager*.

Con estos nuevos datos ya tenemos todo lo necesario para ejecutar el modelo. En este caso, la probabilidad de que la competencia ponga los precios que aparecen en la tabla temporal anteriormente generada es igual a uno (dado que sabemos que son sus precios reales), por lo tanto, la fórmula de ARA queda reducida únicamente a la utilidad multiplicada por la probabilidad de que los clientes se vengán con nosotros dado cada conjunto de precios, tal y como se muestra a continuación:

$$U(p) = \sum_{\vec{p}_0 \in H_0} \sum_{c \in \{0,1\}} u_H(p, c) p_H(C = c | p, \vec{p}_0), \quad (4.1)$$

dado que en este caso \vec{p}_0 es una constante y la función $U(p)$ únicamente depende de p .

Como se explicará más adelante, este modo de ejecución del sistema también cambia el resultado que genera en función de las habitaciones libres del hotel, por tanto, cuando se generen reservas, el *modelo proactivo* volverá a ser ejecutado.

4.4. Eventos del sistema y actualización de tablas

A lo largo del día, el sistema reacciona frente a ciertos eventos y actualiza las tablas de manera que no se pierda información. El sistema es capaz de reaccionar automáticamente ante estos eventos pero también puede ser ejecutado de manera manual en caso de que fuera necesario.

Estos posibles eventos se comentan a continuación:

4.4.1. Establecimiento de precios al finalizar el día y actualización de tablas

Una vez finaliza el día (a las 00:00), lo primero que se debe hacer es actualizar las tablas. En este evento participan los módulos que aparecen en la Figura 4.2. Para ello, se revisan todas las tablas buscando aquellas filas en las cuales el número de clientes temporales sea diferente a cero. En dichas filas, utilizando los datos provenientes del módulo de demanda y el número de clientes del día, se actualizarán los clientes medios y la demanda media.

El nuevo valor del número medio de clientes por día se calcula multiplicando el valor medio que aparecen en la tabla por el número de días, se suma el nuevo número de clientes y se divide todo ello entre el número de días más uno.

Para calcular el nuevo valor medio de la demanda, se comprueban los momentos del día en los cuales ha habido cambios en los precios. Estos datos son enviados al

módulo estimador de demanda para calcular como se ha dividido la demanda total entre las franjas horarias (este proceso está explicado en detalle en la Sección 3.2). Una vez obtenemos estos valores, con ellos ya se puede actualizar la columna M_{demand} de igual manera que se ha hecho anteriormente con el número medio de clientes por día.

Una vez calculados los nuevos valores, se procederá a actualizar la probabilidad de que los clientes elijan nuestro hotel. Por último, la variable contador $t_{clients}$ de todas las filas se reinicia a 0. Además, como se ya se ha mencionado, también se comprueban las tablas auxiliares para actualizar el número de días que ha aparecido cada precio.

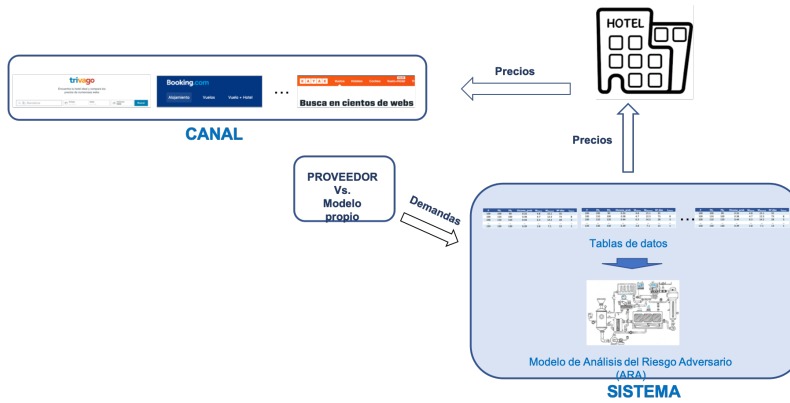


Figura 4.2: Establecimiento de precios al finalizar el día.

En segundo lugar, se ejecutará la *función preventiva* del sistema para generar el precio óptimo de partida para el día siguiente. Como ya se ha explicado, en este caso, aunque se tenga un precio de la competencia, este se omitirá para ejecutar la *versión preventiva* del sistema. El motivo principal es “tomar la iniciativa” y establecer los precios del día sin tener en cuenta lo que está haciendo en ese momento la competencia.

4.4.2. Modificación de los precios de la competencia

Este evento se activa cuando llega, a través del *channel manager*, una actualización de los precios de la competencia. Tal y como se ve en la Figura 4.3, en este proceso intervienen dichos precios de la competencia y la actualización del precio de nuestro hotel.

El cambio en los precios de la competencia significa que nuestro hotel debería ajustar el precio también. Es en esta situación cuando se ejecuta el *modo proactivo* del sistema y se estiman con el regresor las probabilidades correspondientes que no aparezcan en la tabla. Al igual que en cualquier otra situación que se ejecute el *modo proactivo* del sistema, se reestimarán los precios para los M días de estudio, es decir, el modelo se volverá a ejecutar para recalcular todos los precios.

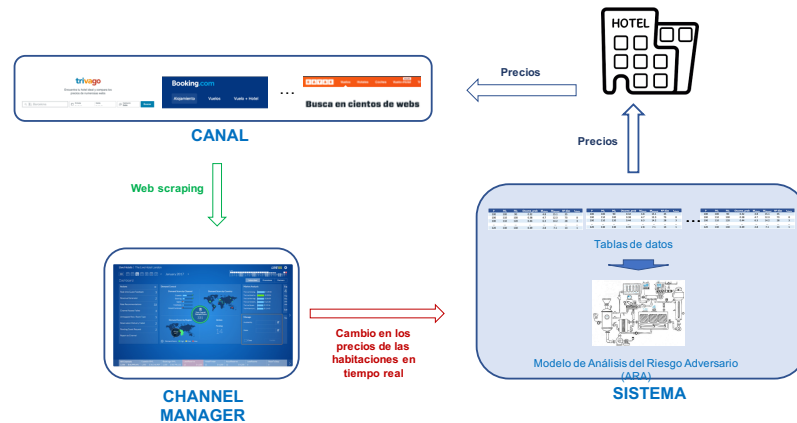


Figura 4.3: Modificación de los precios de la competencia.

4.4.3. Nueva reserva del hotel

Cada vez que nuestro hotel recibe una nueva reserva, se toman los precios de la competencia (que se tienen almacenados por la información del *channel manager*) y se comprueba si esa configuración de precios había aparecido alguna vez en el histórico. En caso afirmativo, se añade uno al contador temporal de clientes. En caso negativo, se crea la entrada en la tabla correspondiente con los precios seleccionados y se pone a uno la variable temporal de clientes, poniendo a cero el resto variables de la fila.

Tal y como se ve en la Figura 4.4, los principales componentes que intervienen en este evento son: nuestro hotel, que envía las reservas en tiempo real, y el sistema que actualiza las tablas.

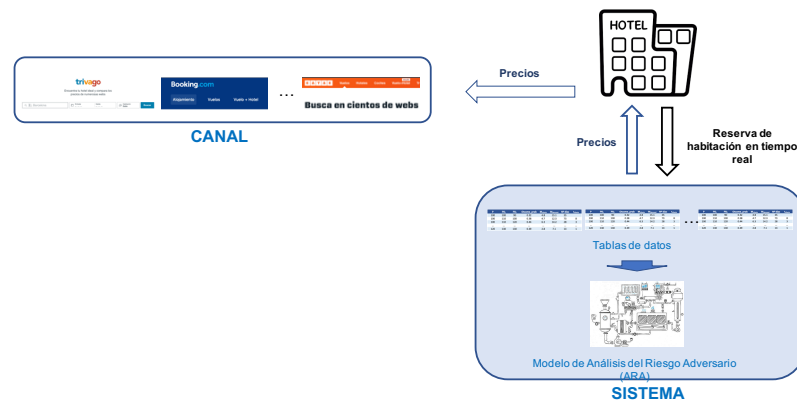


Figura 4.4: Nueva reserva del hotel.

Además, dado que el modelo cambia de resultados en función del número de habitaciones libres, se puede volver a ejecutar el modelo de decisión en su *función proactiva* (dado que se usarán los precios de la competencia) para reajustar el precio de la habitación. Esto se debe a que, una vez vendido un cierto número de habitaciones, el precio debería subir. En este sentido, también se propone la configuración

de una variable global que determine el número de habitaciones vendidas necesarias para volver a ejecutar el modelo. De esta manera, se evita ejecutar el modelo tan repetidamente (ya que los cambios entre una ejecución y otra deberían ser pequeños si se ejecuta con cada reserva) y se configura para que se ejecute cuando se alcance un cierto “*cupo*”.

4.5. Configuración del sistema

Dado que se trata de un sistema ampliamente personalizable, cualquier posible desarrollo como el que se explicará en el caso de uso, debe contar con una buena gestión de variables globales que permitan configurar el sistema correctamente. Algunas de las variables principales que se deben tener en cuenta a la hora de definir en el entorno y que toman gran relevancia en la fórmula de ARA son:

- **Rango de precios:** hace referencia al rango de precios que puede poner nuestro hotel, es decir, el precio mínimo y máximo. En un caso real no debería haber un precio máximo, pero dado que se está planteando un prototipo, como se ha explicado en la metodología, sí que se establece también un precio máximo.
- **Granularidad de los precios:** este valor servirá para definir las unidades en las que se dividirá el *rango de precios*. Por lo tanto, en caso de ser una granularidad 5, el rango de precios se dividirá en intervalos de 5 euros.
- **Tipos de habitación:** es necesario definir, mediante variables globales, cuáles serán los tipos de habitación que se evaluarán tanto de nuestro hotel como de la competencia. Para poder gestionar correctamente los precios de los hoteles, estos tipos de habitación deben ser equivalentes para todos los hoteles que se evalúen (ya sea el nuestro o los de la competencia).
- **Cupos de habitaciones:** en caso de no querer ejecutar el modelo cada vez que se reciben reservas, se puede configurar una variable global a modo de “*cupo*” que ajuste el número de reservas necesarias para volver a ejecutar el modelo.

Con estas variables definidas y el ajuste de otras que resulten auxiliares para el desarrollo, el sistema queda correctamente configurado.

Capítulo 5

Caso de uso y desarrollo del prototipo

Para probar la metodología y la arquitectura diseñadas y comentadas en los capítulos anteriores se ha desarrollado un código *Python* que implemente dichas funcionalidades. El objetivo de este proceso es validar que la metodología seleccionada es aplicable al campo hotelero y que se obtienen resultados coherentes que más adelante puedan ser comparados con los datos que dan los expertos de los hoteles.

Es importante reseñar que para el desarrollo del caso de uso no se ha podido contar con los datos del *channel manager*, el estimador de demanda y las reservas de los hoteles en tiempo real, por lo que se ha tenido que utilizar una fuente de datos antigua procedente de un proyecto anterior. Se hablará más adelante sobre el modelo de datos, pero es importante saber que es diferente al comentado en el Capítulo 4.

Además, dado que no se ha contado con la información del *channel manager* y del hotel, la integración con los mismos se ha omitido. Por tanto, cuando el proyecto avance y se obtenga esta información por parte de un cliente final, se implementarán los conectores necesarios para ambas fuentes de información. Por este mismo motivo, el sistema de actualización de tablas se ha ajustado a los datos con los que se contaba y por tanto, está implementado pero se utilizará de manera aislada con unos datos artificiales de prueba que validen su funcionamiento.

Por último, dado que tampoco se tiene un módulo estimador de demanda ni se han podido comprar estos datos, cualquier información relacionada con la ocupación y la demanda ha sido suprimida de la metodología y de la arquitectura en general.

Por estos motivos, en este capítulo se va a describir el módulo de actualización de las tablas y la implementación de ARA en una versión sencilla de la predicción de precios.

5.1. Entorno de desarrollo

Como ya se ha mencionado, se ha decidido utilizar *Python* en su versión 3.6 para el desarrollo del código de los prototipos. El entorno de desarrollo seleccionado ha

sido *jupyter-notebook*, dada su versatilidad y las facilidades que aporta al trabajo con bases de datos. Esto se debe a que permite ejecutar el código en tiempo real, lo cual ayuda a la visualización y al trabajo con *dataset*.

Para la gestión de paquetes se ha escogido *pip*, y los más utilizados han sido *pandas*, debido a que es la librería por excelencia para la gestión de *datasets* y trae implementadas muchas funcionalidades que facilitan su gestión; y *sklearn*, dado que implementa la mayoría de métodos estadísticos y de *machine learning* necesarios para el desarrollo del prototipo. Además de estas librerías, se han utilizado otras auxiliares como *datetime* y *dateutil* para la gestión de las fechas; *os*, *json* y *shutil* para la gestión de los archivos del sistema y su lectura y escritura; *numpy* para funciones matemáticas, etc.

Por último, dado que el prototipo no está conectado al *channel manager* y al hotel, se ha decidido no utilizar la arquitectura de servidor y por tanto no contratar ningún servicio *cloud*. Por tanto, todo el desarrollo se ha hecho en local a modo de código ejecutable.

5.2. Los datos

Dado que no se ha podido contar con los datos necesarios para la implementación de la metodología y la arquitectura, el proyecto utiliza los datos del anterior trabajo realizado con el *Centro Tecnológico Mixto AI.nnovation Space*. Este *dataset* está dividido entre varios hoteles de Madrid y cada uno consta de las reservas que recibió en un periodo determinado de tiempo.

5.2.1. El dataset original

Se trata de un dataset en el cual todos los hoteles comparten estructura de clases pero hay ciertas diferencias entre ellos. Según el hotel, los datos que se tienen son de meses diferentes y de mayor o menor tamaño según el tiempo de muestreo. Esto hace que de algunos hoteles se tengan muchas reservas (varios meses de información) y de otros tan solo se haya contado con dos o tres meses. El dataset consta de los atributos que se describen en la Tabla 5.1, aunque algunos han tenido que ser retocados.

Dado que no todos los atributos se han utilizado (por falta de relevancia o por estar fuera del alcance del proyecto), en la Tabla 5.1 solo se han representado aquellos atributos que son útiles y que han sido tenidos en cuenta para el desarrollo del prototipo.

Otro aspecto importante del *dataset* es que todas las reservas son de una sola noche, es decir, aunque haya habido reservas más largas, éstas se han dividido en el número de noches (también se ha prorrateado el precio) y cada noche tiene una entrada distinta en la tabla de reservas correspondiente.

Tabla 5.1: Clases del dataset original: reservas de hoteles.

Clase	Descripción	Tipo
timestamp	Fecha en la que se realizó la reserva	<i>datetime string</i>
hotel_name	Nombre del hotel	<i>string</i>
date_from	Fecha en la que empieza la reserva	<i>datetime string</i>
date_to	Fecha en la que termina la reserva	<i>datetime string</i>
days_in_advance	Número de días de antelación de las reserva	<i>integer</i>
room_normalized	Tipo de habitación normalizada	<i>string</i>
board	Tipo de pensión (desayuno o no) normalizado	<i>string</i>
price	Precio que se ha pagado por la reserva	<i>integer</i>

5.2.2. Limpieza y procesamiento de los datos

Dado que los datos de las reservas tenían información que no se iba a utilizar, formatos de atributos incorrectos y tipos no comunes entre diferentes hoteles, ha sido necesario llevar a cabo un proceso de limpieza y procesamiento de los datos para obtener un nuevo *dataset* aplicable en el prototipo ARA.

Para la mejor comprensión del proceso de limpieza de los datos, se han representado en las figuras al final de cada parte del proceso la evolución de los datos descrita en dicha parte. Por tanto, la imagen sirve como resumen y ejemplo para cada apartado de la limpieza.

El proceso que se va a describir a continuación se realiza con cada uno de los hoteles (el nuestro y los que se hayan identificado como competencia) de manera independiente:

- En primer lugar, todas las fechas que había en el set de datos tenían formato de cadena de caracteres y han sido transformados al formato *datetime* de *Python* para poder gestionarse y obtener algunas estadísticas sobre los datos. Indicar también que los campos en formato *string* que se van a utilizar se han puesto en minúsculas para facilitar las comparaciones entre cadenas de caracteres.
- En segundo lugar, dado que solo se han utilizado unas habitaciones concretas y con 30 días de antelación (se explicará detalladamente en la Sección 5.3.3), los datos, ahora bien formateados, se filtran con este criterio para eliminar aquellas reservas que no cumplen con los requisitos.
- En tercer lugar, se analizan las reservas filtradas y se clasifican según la tabla a la que pertenecen. En cada tabla se genera un diccionario de *clave-valor* donde las claves representan los precios y el valor contiene: el número de reservas que ha habido a ese precio y el número de días diferentes que se ha puesto este precio. Este último dato es importante dado que no tenemos información de aquellos días en los cuales no hay reservas y es necesario un valor medio de clientes por día. Por este motivo, teniendo el número de días que se ha puesto cada precio, se distribuye el número total de días de observación (valor

calculado al restar la fecha de la última reserva menos la fecha de la primera) de manera proporcional entre los precios, y así se obtiene una estimación del número de días que se ha puesto cada precio (independientemente de que se hayan hecho reservas o no).

- A continuación, teniendo el número de días que se ha puesto cada precio, se puede estimar el número medio de clientes por día para cada precio dividiendo los clientes de cada precio entre el nuevo valor de días obtenido en el punto anterior.
- Finalmente, se almacenan las tablas temporales con los precios y el número medio de clientes por día que servirá para el siguiente paso de la estructuración de los datos y se almacenan también las tablas del número de días que se ha puesto cada precio. Estas últimas permitirán estimar la probabilidad de que la competencia ponga cada precio según se explicará mas adelante.

Este proceso, representado en la Figura 5.1, termina con unas tablas temporales que contienen el número medio de clientes por día por cada precio. Cada hotel dispondrá de estas mismas tablas.

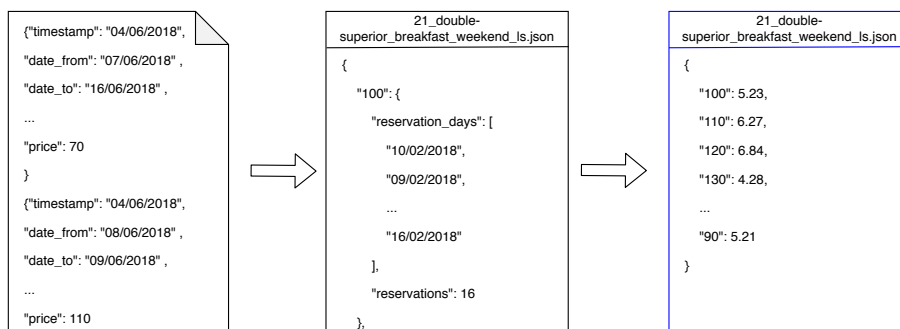


Figura 5.1: Transformación de las reservas en tabla temporal: número de clientes por día.

Una vez se dispone de estas tablas temporales, dado que se necesita la relación entre el precio de nuestro hotel y los precios que ha puesto la competencia, y dado que esta información no se puede sacar del *dataset* original por falta de información, se ha decidido cruzar los datos de los distintos hoteles.

El proceso que se explica a continuación se ejecuta de igual manera para todas las tablas pero únicamente en las correspondientes a nuestro hotel. Además, cuando se hable de los precios de la competencia, se hace referencia a los precios que aparecen en la tabla simétrica a la nuestra y con el mismo nombre pero correspondiente a cada uno de los hoteles. El proceso consiste en:

- Para cada precio de nuestra tabla seleccionamos un precio inferior, igual y superior del resto de los hoteles. En caso de que alguno de estos precios (inferior, igual y superior) no existan en la competencia, no se tomarán.

- A continuación, para cada precio, se generan todas las posibles combinaciones de los precios de la competencia. De esta manera, cada precio de nuestra tabla aparecerá repetido (en las tablas finales) junto a estas combinaciones de precios de la competencia.
- Por último, dado que cada precio de nuestra tabla tiene un número medio de clientes por día asociado, dividimos proporcionalmente este número entre las combinaciones de precios de la competencia. Esto se hace así para que la suma de todos los clientes por día de un mismo precio (que aparecerá en la tabla repetido tantas veces como diferentes configuraciones de este precios haya), sea igual al valor original de la media de clientes del precio. Para hacer la división proporcional, se utilizan los precios de la competencia (a mayor precio de la competencia, mayor número de clientes se vienen a nuestro hotel).

Una vez terminado este proceso (representado en la Figura 5.2), para cada tabla de nuestro hotel se tiene una configuración de precios que asocia nuestros precios a diferentes combinaciones de precios de la competencia. Además, cada configuración de precios es asociada a su vez con un valor medio de clientes por día.

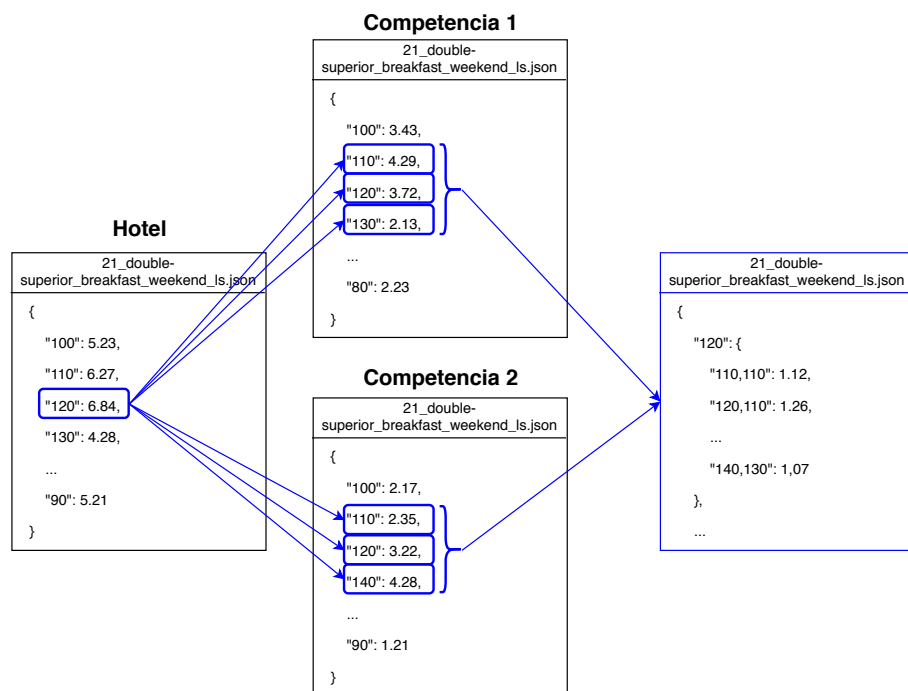


Figura 5.2: Adaptación de las tablas temporales a la competencia.

Una vez obtenidas las tablas de nuestro hotel, se calculan las de la competencia y, finalmente, la demanda media total por conjunto de precios. Este proceso, que se va a describir a continuación, se realiza únicamente para la competencia y se ejecuta igual para todas las tablas. Por lo tanto, se va a describir el proceso sobre una concreta.

- En primer lugar se accede a la tabla con el mismo nombre que la que se está calculando pero de nuestro hotel. Esto se debe a que para construir la tabla se va a utilizar exactamente las mismas configuraciones de precios que se crearon cuando se hizo la tabla correspondiente de nuestro hotel. Para ello, se toman las configuraciones de precios de nuestra tabla pero ahora el precio que representa al hotel será el precio del hotel competencia que se está analizando y nuestro precio pasará a ser el de *“la competencia”* (dado que ahora se está trabajando desde esa perspectiva).
- Una vez se tiene las configuraciones de precios con el orden cambiado se procede a hacer el reparto proporcional del número de clientes medio por precio de la competencia entre las combinaciones de precios (de igual manera que se hacía con nuestro hotel).

Una vez terminado este proceso se tienen, por cada hotel que se está evaluando, todas las tablas junto a las configuraciones de precios de cada una de ellas, y dichas configuraciones asociadas a un valor medio de clientes por día, tal y como se muestra en la Figura 5.3.

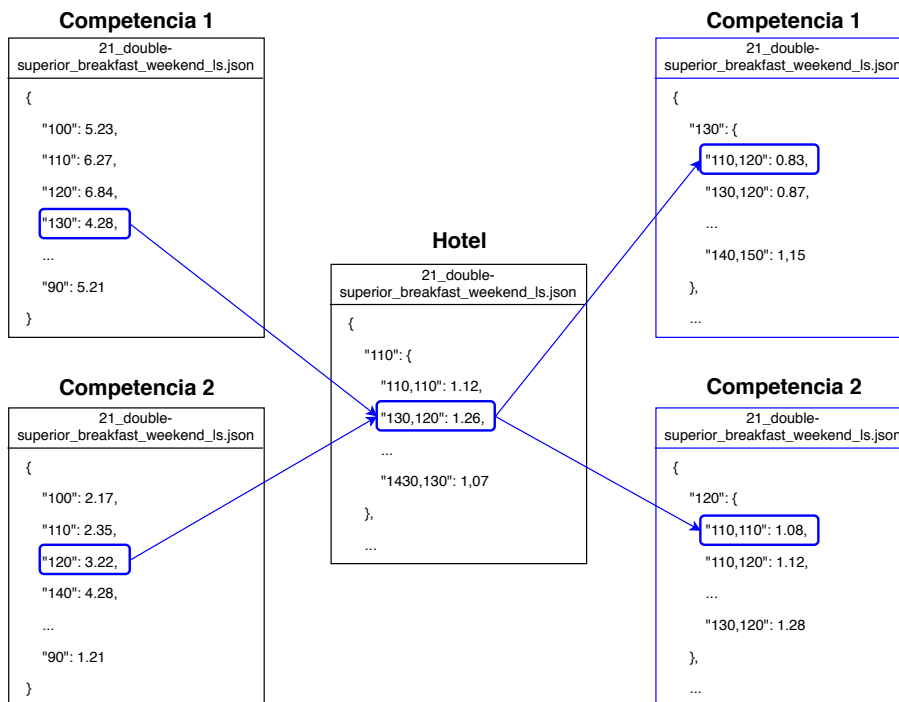


Figura 5.3: Adaptación de la competencia a las configuraciones de precios.

El último paso es calcular la demanda media por cada tabla y configuración de precios y, con ello, la probabilidad de que los clientes elijan nuestro hotel o la competencia. Este proceso se calcula de la siguiente manera:

- Para cada tabla de nuestro hotel, se buscan las tablas equivalentes en los hoteles de la competencia. Las configuraciones de precios que aparecen en

estas tablas tienen que ser iguales pero con el orden de precios cambiado en función del hotel del que sea cada tabla.

- Para cada conjunto de tablas (nuestra tabla y las de la competencia con el mismo nombre), se toma cada configuración de precios y se suman los valores medios de clientes por día, tal y como se muestra en la Figura 5.4. Con ello se calcula la demanda media de clientes por día por cada configuración de precios.
- Finalmente, para cada tabla y cada configuración de precios, dado que ya tenemos el número medio de clientes por día y la demanda media, únicamente dividimos el primer valor entre el segundo y así obtenemos la probabilidad de que los clientes elijan nuestro hotel.

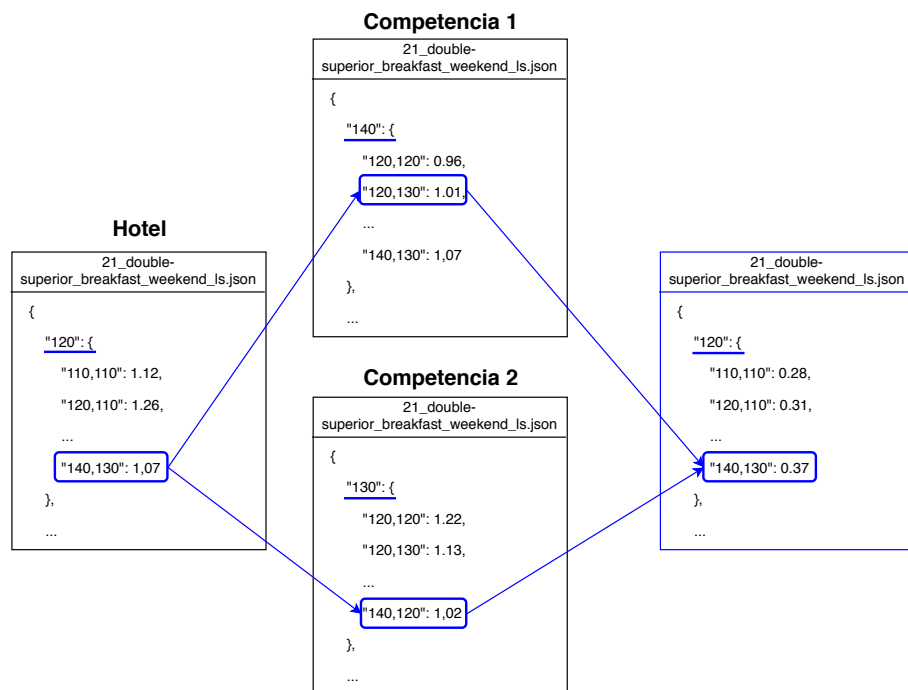


Figura 5.4: Unión de todas las tablas para calcula la probabilidad de que los clientes elijan cada hotel.

Al acabar este proceso, se dispone de las tablas finales que se utilizarán para ejecutar ARA. Como se puede comprobar, durante el proceso se han unido los datos de los precios de manera aleatoria, lo cual genera ciertas incoherencias.

A pesar de este problema, se ha seguido adelante con el prototipo dado que se está gestionando conseguir los datos reales y que el objetivo del prototipo no es otro que hacer funcionar la metodología y comprobar que se obtienen resultados coherentes.

5.2.3. Estructura final de los datos

Tras todo el proceso de limpieza descrito en la sección anterior, se obtienen las tablas que se utilizarán para el modelo ARA. Por tanto, se cuenta con los datos más importantes mencionados en capítulos anteriores: el precio de nuestro hotel, los precios de la competencia y la probabilidad de que los clientes elijan nuestro hotel.

Como ya se ha comentado, cada hotel tendría todas sus tablas aunque en el modelo solo se utilicen las de nuestro hotel. En la Tabla 5.2 se muestra un ejemplo de las tablas finales del prototipo con dos hoteles competencia.

Tabla 5.2: Ejemplo de tabla final del prototipo.

P	P_{01}	P_{02}	$Outcome_{prob}$	$M_{clients}$	M_{demand}
100	100	90	0.32	4.8	15.1
100	110	100	0.38	4.7	12.3
100	110	120	0.44	6.3	14.2
...
120	130	130	0.39	2.8	7.1

Como se puede observar, hay otras columnas auxiliares comentadas en el proceso de limpieza. Por ejemplo, en la primera fila se representa que, cuando nuestro hotel pone el precio 100, la *competencia_1* pone precio 100 y la *competencia_2* pone el precio 90, el número medio de clientes por día que eligen nuestro hotel es 4.8 personas (es un valor decimal dado que son valores medios), la demanda media por día de este tipo de habitaciones (esta tabla concreta) es 15.1 personas, y por tanto, la probabilidad de que un cliente elija nuestro hotel es del 32 %.

5.3. Prototipo ARA

El objetivo del prototipo de *Análisis de Riesgo Adversario* es comprobar que se obtienen resultado coherentes y lógicos, pero dado que no se ha podido contar con toda la información necesaria, el prototipo únicamente calcula el precio óptimo para un día concreto con el modelo ARA. Por tanto, el prototipo desarrollado es diferente al modelo presentado en el Capítulo 3 y no cuenta con un sistema de optimización global para optimizar los precios en base a la ocupación.

Esto significa que el modelo implementado optimiza el precio de cada día de manera independiente a los demás (al contrario de lo que haría el modelo teórico planteado en los capítulos anteriores). La fórmula de ARA adaptada para dos hoteles de competencia queda de la siguiente forma:

$$\begin{aligned}
 p^* = & \arg \max_{p \in H} \sum_{p_{01} \in H_{01}} \sum_{p_{02} \in H_{02}} \left(\sum_{c \in \{0,1\}} u_H(p, c) p_H(C = c | p, p_{01}, p_{02}) \right) \times \\
 & \times \pi_H(H_{01} = p_{01}, H_{02} = p_{02}).
 \end{aligned} \tag{5.1}$$

En este caso, la expresión queda particularizada a dos hoteles competencia ya que es el caso de estudio que se está realizando. Los conjuntos H_{01} y H_{02} representan los conjuntos de posibles precios de ambas competencias, p_{01} y p_{02} representan los precios de la competencia que se están evaluando, y por último, $\pi_H(H_{01} = p_{01}, H_{02} = p_{02})$ representa la probabilidad de que aparezca la combinación de precios de la competencia p_{01}, p_{02} .

Tal y como se puede observar, el código recorre todas las posibles combinaciones de precios de la competencia para cada uno de nuestros precios calculando la utilidad, utilizando las probabilidades de las tablas, y calculando la probabilidad de que aparezca esa combinación de precios. Con todo ello elegirá el argumento máximo.

5.3.1. Modelo de regresión implementado

Como ya se explicó en el Capítulo 3, el objetivo del modelo de regresión es estimar las probabilidades de que un cliente elija nuestro hotel frente a la competencia para aquellas configuraciones de precios que no aparezcan en la tabla que se está consultando. Es decir, generar dicha probabilidad para aquellas configuraciones de precios (combinaciones de nuestros precios junto a los precios actuales de la competencia). Para el prototipo se ha decidido utilizar un modelo de regresión lineal.

Para el desarrollo de este modelo se ha utilizado el módulo de regresión lineal implementado en la librería *SciPy*. Se trata de una regresión lineal ya que ésta es la función que mejor se ajusta, en general, a la variabilidad de $Outcome_{prob}$ (probabilidad de que un cliente elija nuestro hotel) en base a los precios.

Dado que en el prototipo estamos usando dos hoteles de competencia, tenemos que tener en cuenta tres precios (hay que contar con el precio de nuestro hotel). Por tanto, se ha utilizado una regresión lineal en la cual la variable dependiente Y es la probabilidad de que un cliente elija nuestro hotel frente a la competencia y las variables independientes o explicativas \vec{X}_t son cada una de las diferentes configuraciones de precios. El término aleatorio ϵ se estima en el entrenamiento. La función de regresión utilizada es:

$$Y_t = \beta_0 + \beta_1 * X_{t1} + \beta_2 * X_{t2} + \beta_3 * X_{t3} + \epsilon. \quad (5.2)$$

Por lo tanto, a la hora de estimar los parámetros β_1, β_2 y β_3 , estos se calcularán a partir del conjunto de variables independientes \vec{X}_t que representa las diferentes configuraciones de precios. Por tanto, las variables \vec{X}_t en el entrenamiento se sustituirán por las configuraciones de precios que aparezcan en la tabla que se está evaluando.

Para el entrenamiento del modelo de regresión se han utilizado todas las entradas de la correspondiente tabla, es decir, todas las configuraciones de precios que aparezcan en la tabla junto con sus correspondientes probabilidades. Por tanto, cuantas más filas tenga la tabla (más tiempo lleve aprendiendo el sistema y más configuraciones diferentes hayan aparecido), mejor será el rendimiento del modelo de regresión.

5.3.2. Funciones implementadas

Como ya se ha comentado, para el desarrollo del prototipo no se ha contado con los datos necesarios del modelo. Por lo tanto, el caso de uso se ha limitado al cálculo de la calidad de un precio $U(p)$ omitiendo la parte relacionada con la ocupación $O(p)$. Por este motivo, el análisis individual de cada día I perteneciente al rango M ya no se calculará como el sumatorio de los i días que comprenden el periodo hasta I , sino que se evaluará únicamente el día concreto. Advertir por tanto que los resultados que se expondrán más adelante podrán diferir de los resultados del modelo real.

Función preventiva

En el caso de la *función preventiva* del sistema, cuando no se tiene la información del precio de la competencia, se mantiene el método de evaluar todos los posibles precios en el día deseado teniendo en cuenta únicamente el historial.

Para evaluar todos los posibles precios H de nuestro hotel se ha utilizado la Expresión 5.1, que mide la calidad del precio aunque no tiene en cuenta la repercusión de este precio sobre el resto de días. Por este motivo, dado que se van a evaluar todos los precios de nuestro hotel y todas las combinaciones con las que aparezca en el historial, la tabla correspondiente se utiliza entera, es decir, se accede a todas las entradas de la tabla.

Función proactiva

En el caso de la *función proactiva* del sistema se tienen los datos de la competencia. Por este motivo y para poder probar la metodología, se han generado un par de precios para la competencia que son coherentes con sus precios habituales y se ha ejecutado la versión proactiva del modelo.

En este caso, la fórmula utilizada ha sido la Expresión 4.1 ya que, al contar con el precio de la competencia, la probabilidad de dicho precio es 1.

De la tabla únicamente se extraerán aquellas probabilidades correspondientes a configuraciones de precios en las que aparece el precio actual de la competencia p_0 . El resto de las probabilidades se generan con el modelo de regresión tal y como se ha explicado en la Sección 4.3.2.

5.3.3. Las tablas

Dado que la estructura final de las tablas ya se ha explicado en la Sección 5.2.3, en este apartado se explicará cómo se han configurado las variables de entorno para dividir las tablas y sus precios. Además, el código desarrollado se ha hecho lo más genérico posible de manera que todas las variables pueden ser modificadas en el archivo de variables globales y el prototipo se puede adaptar a diferentes casuísticas.

En primer lugar, tras tener una reunión con un experto en el campo hotelero (reunión facilitada por *Accenture*) se decidió dividir los precios en intervalos de 10

euros, es decir, redondear todos los precios a la decena. De esta manera, se discretizan los precios y el modelo resulta más sencillo. Relacionado con este aspecto y en base a los precios que aparecían en las reservas, se ha decidido establecer un precio mínimo de habitación de *70 euros* y un precio máximo de *300 euros*. Con este rango se abarcan todos los precios que aparecen en la base de datos y además se permite un cierto margen para que el sistema pueda proponer otros precios.

En segundo lugar, se ha decidido utilizar como hoteles “competencia” a dos hoteles que están geográficamente cerca de nuestro hotel. Se ha determinado que sean únicamente dos hoteles para simplificar el escenario pero, tanto el código como la metodología, están preparados para adaptarse a cualquier número de hoteles. Como se ha comentado, se ha tenido en cuenta únicamente la cercanía geográfica y no si son o no realmente competencia, puesto que no se dispone de las herramientas para hacer este análisis. Por tanto, el hotel seleccionado y sus hoteles “competencia” puede que realmente no compitan por el mismo sector del mercado.

En tercer lugar, se ha realizado la división de las tablas. Para ello, tal y como se ha mencionado anteriormente, se han distinguido:

- **Días de antelación:** Se han considerado 30 días de antelación, es decir, cuando se hace una estimación de los precios que se deben poner en el hotel, se estimarán los precios de todas las noches hasta pasados 30 días desde la fecha del cálculo.
- **Temporada:** Se ha distinguido entre temporada alta y temporada baja. Esto se debe a que en temporada alta los precios pueden sufrir grandes oscilaciones. Por tanto, se ha configurado qué días del año son de temporada alta y cuáles de temporada baja. En este caso concreto, como es una aproximación sencilla, se ha establecido como temporada alta únicamente el verano, Navidad y la Semana Santa.
- **Día de la semana:** Dado que no se ha contado con suficientes datos ni con la calidad deseada, no se han podido tener en cuenta cada día de la semana como día independiente para la generación de las tablas. Por ello, se ha establecido dos grupos para las tablas: los días de entrada al hotel de domingo a jueves como días de diario y los de viernes a sábado como días de fin de semana.
- **Tipo de habitación:** En base a los datos de nuestro hotel se han seleccionado 3 tipos de habitación: *doble*, *doble superior* y *triple*. Dentro de cada tipo se han incluido varios tipos de habitación que aparecen en la base de datos pero que son similares y por tanto se han agrupado. Por ejemplo, dentro del tipo *doble superior* se han recogido las habitaciones “*doble executive*” y “*doble superior con vistas a la piscina*”.
- **Tipo de alojamiento (pensión):** Dado que la base de datos original distingue las reservas clasificadas entre: “habitación con desayuno” o solo “habitación”, se ha decidido mantener la misma clasificación para las tablas.

- **Canal:** En el prototipo desarrollado se ha considerado un único canal para la obtención de los datos, pero en caso de querer utilizar más, el modelo sería fácilmente extensible.

Toda esta estructura se representa en la Figura 5.5, en la cual se puede observar el esquema de tablas y el número final de éstas.

Por tanto, nuestro sistema contará como máximo con 720 tablas correspondientes a 720 escenarios distintos posibles. 720 será el *máximo* de tablas posible dado que, cuando el prototipo empiece a funcionar, y a media que este vaya “aprendiendo”, irá aumentando el número de tablas ya que habrá recibido nuevas reservas correspondientes a diferentes segmentos (tablas). Pero esto solo sucederá cuando el programa se haya ejecutado el suficiente tiempo y se hayan obtenido las suficientes observaciones. Por ejemplo, en el prototipo desarrollado utilizando los datos de las reservas de nuestro hotel, se ha obtenido únicamente 330 tablas.

Además, se mantienen las tablas planteadas en la arquitectura que contienen el número de días que la competencia ha puesto cada precio para una tabla completa. Con estas tablas auxiliares podremos calcular la probabilidad de que la competencia ponga cada combinación de precios.

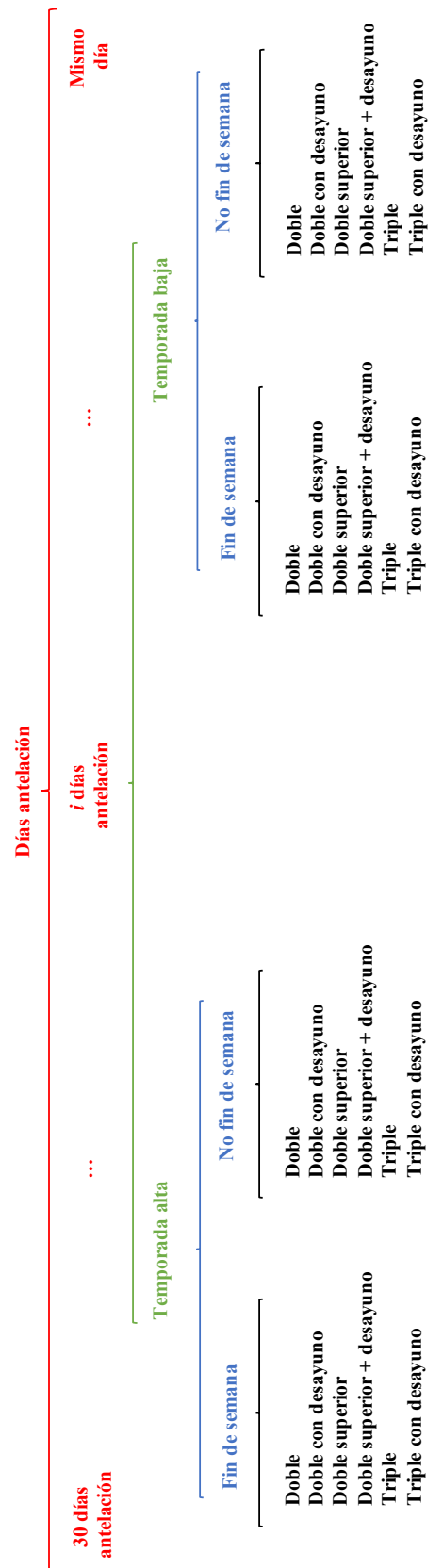
5.4. Prototipo para la carga de datos

Aunque no se ha contado con la información del *channel manager* y del hotel, se ha decidido desarrollar los módulos que se encargarán de actualizar las tablas cuando llegue una nueva reserva o cuando se reciba una nueva actualización de los precios de la competencia. Por este motivo, se ha desarrollado el código que implementa los tres eventos posibles del sistema.

En el desarrollo de este prototipo se ha configurado un formato estándar de entrada de datos para la información relativa a las actualizaciones de los precios de la competencia por parte del *channel manager* y la información en tiempo real de las reservas de nuestro hotel. Por este motivo, para conectar este *script* con las fuentes de información, será necesario un módulo que actúe de conector entre ambos y que transforme los datos de la fuente al formato deseado y gestione las llamadas a las funciones de actualización.

5.5. Resultados

El código desarrollado para el prototipo no pretende generar una propuesta definitiva ni confirmar si los resultados son correctos, dado que no podemos validarlos. El objetivo es validar que se obtienen resultados razonables con la metodología utilizada y coherentes con lo que debería hacer el hotel en la realidad. Además, solo se van a mostrar los resultados del prototipo de decisión dado que el prototipo de carga de datos se ha comprobado que funciona correctamente y no requiere mayor explicación.



$$30 \times 2 \times 2 \times 6 = 720 \text{ tablas}$$

Figura 5.5: División de las tablas para el prototipo.

Es importante mencionar, en lo relativo a las probabilidades que aparecen en la tabla, que dichas probabilidades son bastante altas en general porque los hoteles seleccionados como competencia no lo son en el tipo de habitación estudiado. Esto significa que nuestro hotel vende muchas habitaciones de tipo *doblesuperior* mientras que los hoteles de la competencia no. Por tanto, la probabilidad, en general, de que los clientes elijan nuestro hotel es alta.

Por último, también se van a comentar los resultados obtenidos por el modelo de regresión, dado que es un componente esencial para el funcionamiento del sistema. Hay que tener en cuenta que el modelo de regresión se entrena de manera individual para cada tabla. Por lo tanto, los resultados obtenidos pueden sufrir grandes cambios según la tabla concreta que se está evaluando y según el tipo de habitación (por el problema antes mencionado de las habitaciones *doblesuperior*, por ejemplo) que se considera.

5.5.1. Resultados del *modo preventivo*

En primer lugar se han evaluado los resultados obtenidos del *modo preventivo* del sistema, es decir, del modelo que no utiliza los datos de la competencia. Para probar este modo, se ha utilizado la Tabla A.2 del anexo, la expresión básica de ARA y todas las filas de la misma (dado que se está trabajando con el histórico completo). Se han obtenido los siguientes resultados:

Tabla 5.3: Valores de ARA para caso del *modo preventivo*.

p	ARA_{valor}
90	8.40340112221981
100	36.85177191785423
110	53.609589662852386
120	30.27261441450489
130	5.9391335807629435

Estos valores ARA_{valor} representan la calidad de cada uno de los precios, y el precio recomendado por el sistema sería aquel de mayor valor. En este caso concreto, el precio recomendado sería 110 euros. Se trata de un resultado coherente ya que es un precio habitual del hotel y las probabilidades de que los clientes elijan nuestro hotel son bastante altas. Además, como se puede apreciar, solo aparece el ARA_{valor} de aquellos precios que han aparecido en el histórico y no de todo el rango de precios H del hotel.

5.5.2. Resultados del *modo proactivo*

En segundo lugar, se ha probado el *modo proactivo* del sistema partiendo de una tupla de precios de la competencia inventada pero coherente. En este caso se han utilizado los precios (110, 100), es decir, $p_{01} = 110$ y $p_{02} = 100$.

Cuando se ejecuta este modo, es necesario inferir las probabilidades que no estén en la correspondiente tabla (hay otras configuraciones de precios que sí están y por tanto, se toman directamente de la tabla). Para ello se ha creado un modelo de regresión a partir de la tabla sobre la que se está trabajando y con dicho modelo de regresión se han estimado las probabilidades de las configuraciones de precios que faltan. El resultado se muestra en la Tabla A.1 del anexo. Tal y como se puede observar, la regresión infiere algunas probabilidades iguales a 0 o inferiores, por tanto, todas las probabilidades han sido acotadas en el rango $[0, 1]$.

Con la información de esta nueva tabla ya se puede ejecutar el código para optimizar el precio utilizando la Expresión 4.1. De esta manera se obtiene el conjunto de valores ARA_{valor} que se muestra en la Tabla 5.4.

Tabla 5.4: Valores de ARA para el caso del *modo proactivo*.

p	ARA_{valor}
70	70.10063431634583
80	73.53110320959927
90	75.77665868145496
100	75.29877561847354
110	70.94947401320344
120	75.40384456863566
130	72.90974635490043
140	69.23073471976748
150	64.36680966323678
160	58.317971185308366
170	51.08421928598222
180	42.665553965258304
190	33.0619752231367
200	22.27348305961736
210	10.30007747470024
220	0.0
...	...
290	0.0

Tal y como se puede observar, al igual que en la tabla de probabilidades A.1, a partir del precio 220 el ARA_{valor} se hace 0. Esto se debe a que la probabilidad de que los clientes elijan nuestro hotel es 0, lo cual es lógico dada la fórmula que se está utilizando.

Analizando los resultados se pueden observar datos curiosos. Por ejemplo, en este caso y a diferencia del *modo preventivo*, hay muchos precios con un ARA_{valor} muy similar, lo cual significa que, aunque el ARA_{valor} fuese el más alto (90), se podría recomendar el conjunto de precios con mayor ARA_{valor} . De esta manera, se podría dar una mayor información a modo de recomendación.

Esta propuesta tiene un gran sentido en el caso de uso utilizado y con los datos mostrados, ya que todos los precios cuyo $ARA_{valor} \geq 75$ son razonables para el

hotel y hay tanta igualdad porque no se está teniendo en cuenta la ocupación. Esto significa, por ejemplo, que tanto el precio 90 como el 120 son valores con los que nuestro hotel va a obtener un gran beneficio. Sin embargo, resulta razonable pensar que si la competencia pone los precios (110, 100) y el nuestro es 90, estaremos malvendiendo las habitaciones y nos llenaremos antes del momento adecuado.

Este tipo de contradicciones o “errores” aparentes, en realidad no lo son, dado que a nivel matemático los resultados tienen sentido y si el hotel contase con infinitas habitaciones, también podría ser una buena opción poner el precio 90. Por otro lado, el precio 120 por ejemplo, resulta razonable y lógico dado que si nuestro hotel domina las ventas de las habitaciones de tipo *dobles superior*, aunque ponga el precio más alto, seguirá obteniendo muchos clientes y obteniendo un buen beneficio total.

Por último, se puede observar a partir de los resultados obtenidos, tanto en el caso *proactivo* como en el *preventivo*, que si se continuase la curva por la izquierda en el caso *proactivo* y por el lado izquierdo y derecho en el caso *preventivo*, se obtendría una campana. En la Figura 5.6 se muestra el caso del *modo proactivo*.

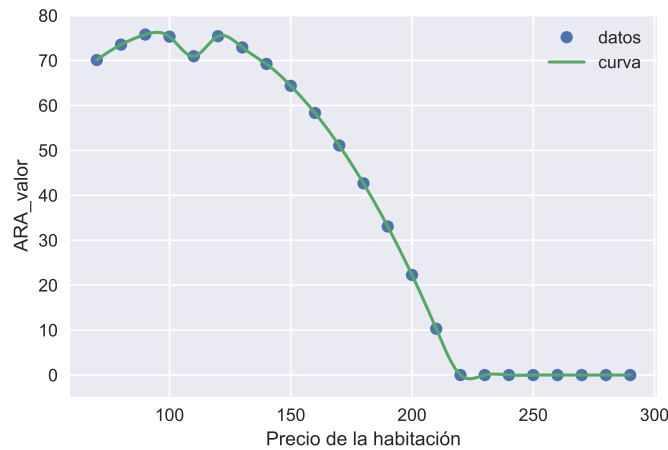


Figura 5.6: Variación del ARA_{valor} en función al precio de la habitación.

Esta forma de la curva es razonable ya que a medida que aumenta el precio de la habitación (con respecto al precio de la competencia) aumenta el beneficio pero disminuye la probabilidad de que los clientes elijan nuestro hotel.

Por último, en la Figura 5.6 se puede observar que en la parte más alta de la campana hay un pequeño “valle” que se corresponde con el precio 110, el cual no tiene aparentemente sentido. Esto se debe al ruido y el error introducido por los datos que generan un gran salto en las probabilidades entre los precios 100 y 110 (tal y como se puede observar en la Tabla A.1).

5.5.3. Resultados del modelo de regresión

Para evaluar los resultados obtenidos con el modelo de regresión se han dividido las entradas de la tabla entre *train* y *test*. Con el grupo de *train* se ha entrenado el modelo y se ha probado con el conjunto de *test* para analizar los resultados

obtenidos. La métrica de error utilizada para evaluar la bondad del modelo ha sido el coeficiente de determinación R^2 explicado en la metodología.

Dado que el dataset no es muy amplio y que los datos tienen ciertas incoherencias, los resultados de R^2 obtenidos dependen en gran medida del conjunto de entrenamiento que se elija. Por este motivo, en función del conjunto seleccionado, los resultados pueden cambiar drásticamente desde resultados negativos hasta resultados positivos relativamente buenos.

Para la selección del conjunto de *train* y *test* se ha utilizado una función de la librería *sklearn*, que de manera aleatoria separa ambos conjuntos en la proporción indicada en el parámetro *test_size*. Para permitir la *reproducibilidad* se ha fijado la semilla *random_state* = 317.

El mejor resultado ha sido $R^2 = 0.56$ para el caso de la tabla que se ha utilizado en el prototipo, y el valor de los parámetros β_1, β_2 y β_3 ha sido (-0.00649589, -0.00111435, 0.0031938). El bajo resultado de R^2 se debe, como ya se ha explicado, a los errores introducidos en el dataset por la falta de información.

Capítulo 6

Conclusiones y líneas futuras de trabajo

El trabajo realizado consiste con el planteamiento de una metodología completa de *revenue management* capaz de optimizar los precios que debe establecer un hotel dentro de un horizonte de análisis para maximizar el beneficio obtenido.

Respecto a la industria hotelera y las técnicas de los *revenue management systems* (RMS), se puede concluir que son dos sectores que han convergido rápidamente en los últimos años debido a la clara relación que hay entre ellos. Además, han generado una gran expectativa e investigación por parte de la comunidad científica, que se ha volcado en nuevas propuestas para RMS tanto en la industria hotelera como en otros tipos de sectores.

Este modelo planteado es innovador tanto por el uso del análisis de riesgo adversario en la industria hotelera, como por tener en cuenta la competencia y utilizar un sistema de optimización para obtener el conjunto de precios idóneo. Algunas de estas técnicas ya habían sido utilizadas por separado dentro o fuera de este sector, pero nunca se habían combinado. Además, la adaptación de los modelos seleccionados, y su unión como *framework* para RM, supone una aportación a la comunidad científica que podrá ser revisado, probado, mejorado y ampliado en el futuro.

A pesar de las limitaciones que se han tenido que adoptar a la hora de diseñar el sistema, la metodología propuesta incorpora la mayoría de los factores claves que se reconocen en la literatura para obtener un buen RMS. Entre ellos, el uso del análisis de riesgo adversario (ARA), que ha dado buenos resultados en los últimos estudios en los que se ha aplicado, supone una garantía y ofrece un nuevo punto de vista para la gestión del riesgo y medida de calidad a la hora de determinar los precios del hotel.

Por otro lado, aunque no se haya podido probar la metodología al completo, los prototipos diseñados para comprobar si los resultados obtenidos eran coherentes han sido satisfactorios y han verificado que el enfoque adoptado resulta prometedor.

Además, se han aplicado conocimientos adquiridos a lo largo del máster, lo que ha servido para afianzar conceptos y poner en práctica (en un caso real) estas técnicas. Algunos de estos conocimientos utilizados han sido: metaheurísticas poblacionales

para optimización de funciones, modelos de regresión, análisis de datos, etc. Gracias a estos conocimientos, el proyecto ha tenido una mayor variedad de técnicas que hacen de la metodología un modelo multidisciplinar y con gran variedad de aportes científicos punteros.

De esta manera, se aporta un nuevo *framework* para el RM extensible a otras industrias y respaldado por metodologías ya validadas. Además, dado que se ha diseñado de la manera más genérica posible, esta metodología es aplicable a gran cantidad de situaciones y casuísticas.

6.1. Trabajo futuro

Dado que el trabajo final ha sido principalmente teórico, quedan una gran cantidad de vías de estudio para trabajo futuro que puedan complementar, corregir y mejorar la metodología planteada.

En primer lugar y más importante, se deberían obtener los datos necesarios para probar la metodología, conectar el sistema con un *channel manager*, el hotel a estudiar y el módulo estimador de demanda. Una vez obtenidos estos datos, se debería dejar entrenar al modelo durante unos meses y empezar a validar los resultados obtenidos con un experto gestor de precios de hoteles.

En segundo lugar, y dada la importancia del componente para la estimación de la demanda, se debería generar un modelo aislado, teniendo en cuenta todas las características mencionadas en la Sección 3.2. Dada la envergadura de este modelo, podría constituir por si solo un Trabajo Fin de Máster. Además, resulta de gran importancia, dado que un prototipo preciso de este modelo podría suponer una gran mejora para una metodología como la propuesta.

Otra posible línea de trabajo futuro podría ser ampliar la segmentación de los datos, generar más tablas, ampliar el horizonte de estudio, etc. a modo de test de “estrés” del sistema. El objetivo de esta prueba sería comprobar el alcance de la metodología y qué volumen podría soportar.

En [50] se demostró que las reservas cancelables con reembolso podían ser una estrategia óptima para subir los precios y con ello aumentar el beneficio. La idea es vender habitaciones contando con las cancelaciones que se presupone que habrá y posteriormente aumentar el precio de estas habitaciones canceladas en los últimos días de venta.

Otra posible idea a futuro es gestionar los porcentajes de cancelaciones vendiendo por encima de la capacidad del hotel y así ganando clientes a la competencia. De esta manera, se puede reducir un poco el precio y ganar más clientes, lo cual aumenta el beneficio, jugando con el riesgo de no recibir el número de cancelaciones esperado. Esta opción tendría mucho sentido dado que ARA es una metodología preparada para casos de este tipo.

Otras de las adaptaciones que se podrían tener en cuenta en versiones futuras de la metodología, es ampliar algunas de las limitaciones que se han tenido que tomar para el desarrollo del proyecto actual. Por ejemplo, gestionar reservas en

grupo, promociones de descuento para las habitaciones, variación de precios entre los diferentes canales, etc.

Por otro lado, se podría adaptar la fórmula para aumentar los rangos de precios de nuestro hotel y de la competencia o incluso gestionarlos como un conjunto continuo e integrar sobre la función para optimizar el precio. Este requeriría de muchos cambios en el modelo dado que el sistema de optimización propuesto requiere de un conjunto discreto de precios, pero podría estudiarse como evolución futura.

Otra línea de trabajo futuro que podría mejorar el rendimiento del sistema es el estudio de los distintos modelos de regresión, y un análisis para seleccionar la función que mejor se adapte a la variabilidad de las probabilidades en base a los precios. Esto podría suponer una mejora en los resultados, dado que el modelo de regresión es un componente clave para la ejecución del *modo proactivo* del sistema.

Además, se podrían utilizar otro tipo de algoritmos de optimización y metaheurísticas para calcular el conjunto de precios a proponer, ya que puede que en ciertos escenarios sea necesario una mayor velocidad de respuesta (mayor que la que puede ofrecer un algoritmo genético) o en otros casos, que sea necesario una mayor precisión de los precios (pero disponiendo de mayor tiempo de cómputo). Por lo tanto, se podría investigar la adaptación de otras metaheurísticas para el problema.

En definitiva, todas estas adaptaciones ayudarán a mejorar y ampliar el modelo que podrá evolucionar con el paso del tiempo.

Apéndice A

Tablas de ejemplo

Tabla A.1: Ejemplo de tabla temporal del prototipo con valores generados por el regresor

p	p0_1	p0_2	outcome_prob
70	110	100	0.8762579289543229
80	110	100	0.8170122578844362
90	110	100	0.7577665868145496
100	110	100	0.6845343238043049
110	110	100	0.5912456167766954
120	110	100	0.5800295736048897
130	110	100	0.5207839025350031
140	110	100	0.4615382314651165
150	110	100	0.40229256039522987
160	110	100	0.34304688932534333
170	110	100	0.2838012182554568
180	110	100	0.22455554718557003
190	110	100	0.16530987611568349
200	110	100	0.10606420504579694
210	110	100	0.04681853397591018
220	110	100	0.0
230	110	100	0.0
240	110	100	0.0
250	110	100	0.0
260	110	100	0.0
270	110	100	0.0
280	110	100	0.0
290	110	100	0.0

Tabla A.2: Ejemplo de tabla final del prototipo

p	p0_1	p0_2	clients	demand	outcome_prob
100.0	100.0	100.0	0.18583042973286873	0.2773767203658331	0.669956835194303
100.0	100.0	110.0	0.19512195121951217	0.270564091421566	0.7211672110453585
100.0	100.0	90.0	0.17653890824622528	0.31298221953616234	0.5640541130670452
100.0	110.0	100.0	0.19512195121951217	0.28504334177886126	0.6845343238043049
100.0	110.0	110.0	0.20441347270615562	0.27711539458278545	0.7376474808045683
100.0	110.0	90.0	0.18583042973286873	0.3232038027123023	0.5749636241077412
110.0	110.0	110.0	0.1428571428571429	0.21902106101361224	0.6522529942828844
110.0	110.0	120.0	0.14935064935064937	0.22326985584561343	0.6689243775654261
110.0	110.0	100.0	0.1363636363636364	0.23063788127014379	0.5912456167766954
110.0	120.0	110.0	0.14935064935064937	0.2255489798669501	0.6621650403329262
110.0	120.0	120.0	0.15584415584415587	0.22953994308160977	0.6789413369713506
110.0	120.0	100.0	0.1428571428571429	0.23814479606302918	0.5998751399099784
110.0	100.0	110.0	0.1363636363636364	0.2154922330300197	0.632800702123867
110.0	100.0	120.0	0.1428571428571429	0.22013518179162353	0.6489518926255469
110.0	100.0	100.0	0.12987012987012989	0.22599373503474246	0.574662522614464
120.0	120.0	120.0	0.15873015873015878	0.2356301106301107	0.6736412350089309
120.0	120.0	130.0	0.16534391534391538	0.2502243666827001	0.6607826309480949
120.0	120.0	110.0	0.15211640211640215	0.2316885910403136	0.6565554282728321
120.0	130.0	120.0	0.16534391534391538	0.2551130351130352	0.6481202156944077
120.0	130.0	130.0	0.171957671957672	0.2704425204425205	0.6358381502890175
120.0	130.0	110.0	0.15873015873015878	0.25088015026562516	0.6326931746576984
120.0	110.0	120.0	0.15211640211640215	0.229327911146093	0.6633139479454667
120.0	110.0	130.0	0.15873015873015878	0.24373613479674092	0.6512376954796988
120.0	110.0	110.0	0.14550264550264555	0.22512855993895442	0.6463091379525542
130.0	130.0	130.0	0.14285714285714285	0.24528138528138527	0.5824214613483939
130.0	130.0	140.0	0.14835164835164838	0.30250749250749254	0.4904065255440706
130.0	130.0	120.0	0.13736263736263737	0.2307969807969808	0.5951665263917278
130.0	140.0	130.0	0.14835164835164838	0.29752913752913757	0.49861216815149756
130.0	140.0	140.0	0.15384615384615385	0.35839160839160844	0.4292682926829268
130.0	140.0	120.0	0.14285714285714285	0.281052281052281	0.5082938388625593
130.0	120.0	130.0	0.13736263736263737	0.22572142363809033	0.608549401951661
130.0	120.0	140.0	0.14285714285714285	0.2805683899433899	0.5091704838380654
130.0	120.0	120.0	0.1318681318681319	0.2119722484305818	0.622100925212939
90.0	100.0	90.0	0.6263736263736264	0.7559091215532934	0.8286361528307944
90.0	100.0	100.0	0.6593406593406593	0.7463096354419756	0.8834679709718447

Bibliografía

- [1] Abrate, G., Capriello, A., & Fraquelli, G. (2011). *When quality signals talk: Evidence from the Turin hotel industry*. *Tourism Management*, 32(4), 912–921.
- [2] Abrate, G., Fraquelli, G. & Viglia, G. (2012). *Dynamic pricing strategies: evidence from European hotels*. *International Journal of Hospitality Management*, 31(1), 160–168.
- [3] Akçay, Y., Natarajan, H.P. & Xu, S.H. (2010). *Joint dynamic pricing of multiple perishable products under consumer choice*. *Management Science*, 56(8), 1345–1361.
- [4] Altin, M., Schwartz, Z., & Uysal, M. (2017). “Where you do it” matters: The impact of hotels’ revenue management implementation strategies on performance. *International Journal of Hospitality Management*, 67, 46–52.
- [5] Anderson, C. K., & Kimes, S. (2011). *Revenue management for enhanced profitability an introduction for hotel owners and asset managers*. New York: John Wiley & Sons, Inc.
- [6] Arce, D., & Sandler, T. (2007). *Terrorist signalling and the value of intelligence*. *British Journal of Political Science*, 37, 573–586.
- [7] Abdel, H., & Saleh, M. (2011). *Dynamic room pricing model for hotel revenue management systems*. *Egyptian Informatics Journal*, 12(3), 177–183.
- [8] Bandinelli, R.D. (2000). *An optimal, dynamic policy for hotel yield management*. *European Journal of Operational Research*, 121(3), 476–503.
- [9] Barros, C. P. (2005). *Measuring efficiency in the hotel sector*. *Annals of Tourism Research*, 32(2), 456–477.
- [10] Becerra, M., Santaló, J., & Silva, R. (2013). *Being better vs. being different: Differentiation, competition, and pricing strategies in the Spanish hotel industry*. *Tourism Management*, 34, 71–79.
- [11] Berman, B. (2005) *Applying yield management pricing to your service business*. *Business Horizons*, 48(2), 169–179.

- [12] Bier, V., and Cox, L. A., Jr. (2007), *Probabilistic risk analysis for engineered systems*. New York: Cambridge University Press, pp. 279–301.
- [13] Brown, G., Carlyle, M., Salmeron, J., and Wood, K. (2006). *Defending Critical Infrastructure*. Interfaces, 36, 530–544.
- [14] Camerer, C. (2003). *Behavioral Game Theory*. Princeton: Princeton University Press.
- [15] Canina, L., Enz, C., & Harrison, J. (2005). *Agglomeration effects and strategic orientations: Evidence from the U.S. Lodging Industry*. The Academy of Management Journal, 48(4), 565–581.
- [16] Chiu, H.H., & Chen, C.M. (2014). *Advertising, price and hotel service quality: A signalling perspective*. Tourism Economics, 20(5), 1013–1025.
- [17] Cross, R. G., Higbie, J. A., & Cross, D. Q. (Dax). (2009). *Revenue Management's Renaissance: A Rebirth of the Art and Science of Profitable Revenue Generation*. Cornell Hospitality Quarterly, 50(1), 56–81.
- [18] Cuevas, E., Fausto, F., & González, A. (2020). *An introduction to nature-inspired metaheuristics and swarm methods*. In Intelligent Systems Reference Library, 160, 1–41.
- [19] Davis, L., (ed.) (1991). *Handbook of genetic algorithms*, New York: Van Nostrand Reinhold.
- [20] Emeksiz, M., Gursoy, D. & Icoz, O. (2006). *A yield management model for five-star hotels: Computerized and non-computerized implementation*. International Journal of Hospitality Management, 25(4), 536–551.
- [21] Falk, M., & Hagsten, E. (2015). *Modelling growth and revenue for Swedish hotel establishments*. International Journal of Hospitality Management, 45, 59–68.
- [22] Ferreira, A., & de Haan, L. (2014). *The generalized Pareto process; with a view towards application and simulation*. Bernoulli, 20(4), 1717–1737.
- [23] French, S., & Rios Insua, D. (2000). *Statistical decision theory*. London: Arnold.
- [24] Gallego, G., & van Ryzin, G. (1994). *Optimal dynamic pricing of inventories with stochastic demand over finite horizons*. Management Science, 40(8), 999–1020.
- [25] Gan, L., & Hernandez, M. A. (2012). *Making friends with your neighbors? Agglomeration and tacit collusion in the lodging industry*. The Review of Economics and Statistics, 95(3), 1002–1017.
- [26] Guizzardi, A., Emanuele Pons, F.M. & Ranieri, E. (2017). *Advance booking and hotel price variability online: any opportunity for business customers?* International Journal of Hospitality Management, 64, 85–93.

- [27] Hausken, K. (2002). *Probabilistic Risk Analysis and Game Theory*. Risk Analysis, 22, 17–27.
- [28] Heyes, A. (2000). *Implementing Environmental Regulation: Enforcement and Compliance*. Journal of Regulatory Economics, 17, 107–129.
- [29] Hung, W.T., Shang, J.K., & Wang, F.C. (2010). *Pricing determinants in the hotel industry: Quantile regression analysis*. International Journal of Hospitality Management, 29(3), 378–384.
- [30] *IDeaS*, <https://ideas.com/es/>
- [31] Insua, I. R., Rios, J., & Banks, D. (2009). *Adversarial risk analysis*. Journal of the American Statistical Association, 104(486), 841–854.
- [32] Ivanov, S. (2014). *Hotel revenue management from theory to practice*. Varna: Zangador Ltd.
- [33] Ivanov, S. & Zhechev, V. (2011). *Hotel marketing (in Bulgarian)*. Varna: Zangador Ltd.
- [34] Ivanov, S. & Zhechev, V. (2012). *Hotel revenue management: a critical literature review*. Tourism: An International Interdisciplinary Journal, 60(2), 175–197.
- [35] Jacobs, T.L., Ratliff, R. & Smith, B.C. (2010). *Understanding the relationship between price, revenue management controls and scheduled capacity: a price balance statistic for optimizing pricing strategies*. Journal of Revenue and Pricing Management, 9(4), 356–373.
- [36] *JDA*, <https://jda.com/>
- [37] Kadane, J. B., and Larkey, P. D. (1982). *Subjective Probability and the Theory of Games*. Management Science, 28, 113–120.
- [38] Kardes, E. & R. Hall (2005). *Survey of Literature on Strategic Decision Making in Presence of Adversaries*. CREATE report.
- [39] Kimes, S.E. & Wirtz, J. (2015) *Revenue management: Advanced strategies and tools to enhance firm profitability*. Foundations and Trends (R) in Marketing, 8(1), 1–68.
- [40] Lee, S. (2011). *Study of demand models and price optimization performance*. Doctoral Thesis, Georgia Tech Theses and Dissertations, School of Industrial and Systems Engineering.
- [41] Lee, S.K. & Jang, S.C. (2011). *Room rates of US airport hotels: examining the dual effects of proximities*. Journal of Travel Research, 50(2), 186–197.
- [42] Lieberman, W.H. (1993) *Debunking the myths of yield management*. The Cornell Hotel and Restaurant Administration Quarterly, 34(1): 34–41.

- [43] Lieberman, W.H. (2003). *Getting the most from revenue management*. Journal of Revenue and Pricing Management, 2(2), 103–115.
- [44] Mauri, A.G., (2013). *Hotel revenue management: Principles and practices*. Londres: Pearson Italia Spa.
- [45] McAfee, R. P., and McMillan, J. (1996). *Competition and game theory*. Journal of Marketing Research, 33, 263–267.
- [46] McAfee, P., & te Veld, V. (2008). *Dynamic pricing with constant demand elasticity*. Production and Operations Management, 17(4), 432–438.
- [47] Min, H., Min, H., & Chung, K., (2002). *Dynamic benchmarking of hotel service quality*. Journal of Services Marketing, 16 (4), 302–321.
- [48] Monson, T. S. (1985). *Favorite quotes from the collection of Thomas S. Monson*. Salt Lake City: Deseret Books.
- [49] Myerson, R. (1991). *Game Theory: Analysis of conflict*. Cambridge: Harvard University Press.
- [50] Ng, I.C.L. (2009). *A demand-based model for the advance and spot pricing of services*. Journal of Product & Brand Management, 18(7), 517–528.
- [51] Ohira, R., Islam, M. S., Jo, J., & Stantic, B. (2020). *AMGA: An adaptive and modular genetic algorithm for the Traveling Salesman problem*. In Advances in Intelligent Systems and Computing (Vol. 941, pp. 1096–1109).
- [52] Ortega, B. (2016). *Revenue management systems and hotel performance in the economic downturn*. International Journal of Contemporary Hospitality Management, 28(4), 680–680.
- [53] Park, K., & Jang, S.C., (2010). *Firm growth patterns: examining the associations with firm size and internationalization (Shawn)*. International Journal of Hospitality Management, 29 (3), 368–377
- [54] Parnell, G. (2007). *Multi-objective decision analysis*. in Handbook of Science and Technology for Homeland Security, ed. Voeller, New York: Wiley.
- [55] Pearl, J. (2005). *Influence diagrams — Historical and personal perspectives*. Decision Analysis, 2(4), 232–234.
- [56] Phillips, R. L. (2005). *Pricing and revenue optimization*. Stanford: Stanford University Press.
- [57] Pinker, E. J. (2007). *An analysis of short-term responses to threats of terrorism*. Management Science, 53, 865–880.
- [58] Powell, R. (2007). *Allocating defensive resources with private information about vulnerability*. The American Political Science Review, 101, 799–809.

- [59] *PROS*, <https://pros.com/solutions/price-optimization-management/>
- [60] Qu, H., Xu, P., & Tan, A. (2002). *A simultaneous equations model of the hotel room supply and demand in Hong Kong*. International Journal of Hospitality Management, 21(4), 455–462.
- [61] *Revenue Analytics*, <https://www.revenueanalytics.com/>
- [62] Rios, J., & Insua, D. R. (2012). *Adversarial risk analysis for counterterrorism modeling*. Risk Analysis, 32(5), 894–915.
- [63] Rios Insua, D., Rios, J., & Banks, D. (2009). *Adversarial risk analysis*. Journal of the American Statistical Association, 104(486), 841–854.
- [64] Rios Insua, D., Banks, D., & Rios, J. (2016). *Modeling Opponents in Adversarial Risk Analysis*. Risk Analysis, 36(4), 742–755.
- [65] Rothkopf, M. (2007). *Decision analysis: the right tool for auctions*. Decision Analysis, 4, 167–172.
- [66] Schaffer, J.D., R.A. Caruna, L.J. Eshelman, R. Das (1989). *A study of control parameters affecting online performance of genetic algorithms for function optimization*. J.D. Schaffer (ed.), Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmann, 51-60.
- [67] Seniwi, D. (1991). *A genetic algorithm for the traveling salesman problem*, M.Sc. Thesis, University of North Carolina.
- [68] Shy, O. (2008). *How to Price: A Guide to Pricing Techniques and Yield Management*. Cambridge: Cambridge University Press.
- [69] Siqueira, K., and Sandler, T. (2006). *Terrorists vs the government: Strategic intervention, support and sponsorship*. The Journal of Conflict Resolution, 50, 878–898.
- [70] Smith, B., Leimkuhler, J. & Darrow, R. (1992) *Yield management at American airlines*. Interfaces, 22(1), 8–31.
- [71] Talluri, K.T., & Van Ryzin, G.J. (2005). *The theory and practice of revenue management*. New York: Springer Science & Business Media.
- [72] Thrane, C. (2007). *Examining the determinants of room rates for hotels in capital cities: The Oslo experience*. Journal of Revenue and Pricing Management, 5(4), 315–323.
- [73] Toh, R.S. & Dekay, F. (2002) *Hotel room-inventory management*. Cornell Hotel and Restaurant Administration Quarterly, 43(4), 79–90.
- [74] Tranter, K.A., Stuart-Hill, T. & Parker, J. (2008). *Introduction to revenue management for the hospitality industry*. Harlow: Pearson Prentice Hall.

- [75] Vives, A., Jacob, M., & Payeras, M. (2018). *Revenue management and price optimization techniques in the hotel sector: A critical literature review*. *Tourism Economics*, 24(6), 720–752.
- [76] Wang, X., & Xia, Q. (2020). *Optimization of logistics warehouse location based on genetic algorithm*. In *Advances in Intelligent Systems and Computing*, 928, 745–752.
- [77] von Winterfeldt, D., & O’Sullivan, T. M. (2006). *Should we protect commercial airplanes against surface-to-air missile attacks by terrorists*. *Decision Analysis*, 3, 63–75.
- [78] Williams, K. R. (2018). *Dynamic airline pricing and seat availability*. Cowles Foundation discussion paper 3003U.
- [79] Yang, Y., Tang, J., Luo, H., & Law, R. (2015). *Hotel location evaluation: A combination of machine learning tools and web GIS*. *International Journal of Hospitality Management*, 47, 14–24.
- [80] Yeoman, I. (2011) *Ten years of revenue management*. *Journal of Revenue and Pricing Management*, 10(1), 1–7.
- [81] Zhang, D., & Lu, Z. (2013). *Assessing the value of dynamic pricing in network revenue management*. *INFORMS Journal on Computing*, 25(1), 102–115.
- [82] Zhang, D., & Weatherford, L. (2016). *Dynamic pricing for network revenue management: A new approach and application in the hotel industry*. *INFORMS Journal on Computing*, 29(1), 18–35.
- [83] Zhuang, J., & Bier, V. (2007). *Balancing terrorism and natural disasters. Defensive strategy with endogenous attack effort*. *Operations Research*, 55, 976–991.
- [84] Zhuang, W., Chen, J. & Fu, X. (2017). *Joint dynamic pricing and capacity control for hotels and rentals with advanced demand information*. *Operations Research Letters*, 45, 397–402.