LAB University of Applied Sciences

Consideration of Simpson's Paradox as a relevant concept for travel platforms

Dmitriy Debyolyy
LAB University of Applied Sciences
Bachelor of Business Administration
2022

Abstract

Author(s)	Publication type	Published
Debyolyy, Dmitriy	Thesis, UAS	2022
	Number of pages	
	39	

Title of Publication

Consideration of Simpson’s Paradox as a relevant concept for travel platforms

Degree and field of study

Bachelor of Business Administration, International Business

Name, title and organisation of the client (if the thesis work is commissioned by another party)

No commission.

Abstract

The main purpose of the thesis work was to examine the usefulness of considering Simpson's paradox in data analysis in the field of travel platforms and what benefits can be derived from it.

In the course of the study, there were reviewed the theoretical aspects concerning the Simpson paradox and its nature, presented some examples of the application of the concept in different areas, and carried out a research study dedicated to the analysis of data from Booking.com.

The research part involved the analysis of secondary data in order to detect Simpson's paradox. The research included examining five hotels and calculating their mean values for three variables that could potentially serve as the underlying cause (confounding variable) of the paradox. All calculations were performed using the Python programming language, and bootstrapping method was used to add statistical reliability to the study.

The conclusion states that there is an absence of Simpson's paradox in the examined data, however, the study revealed a series of patterns associated with the studied variables and the reviewer's rating. Such patterns provide a basis for the improvement and optimization of recommender systems of Booking.com. Thereby will be useful for both travel platforms and users of its services: hotels and travelers.

Keywords

Simpson’s paradox, travel platforms, optimisation of recommendation systems

Contents

Appendices

Appendix 1. Code for cleaning data and calculation of mean values using bootstrapping method

1 Introduction

In the early 1970s, it was clear that the University of California, Berkley would be sued for explicit gender discrimination related to the admission of graduate students. The basis for this was the statistical data on the results of student admissions. At the beginning of the academic year in 1973, around 44% of all male applicants were accepted while only 33% of all female applicants were able to receive admission to the same university. Fearing long trials and possible reputational embarrassment, the school instructed their statistician Peter Bickel to look at the data and investigate whether or not it was true. The findings were surprising and indicated evidence of a statistically significant gender bias in favor of women in four of the six departments, while in the remaining two departments the statistically significant bias was insignificant or absent at all. The point is that it was discovered that women tended to apply to departments that admitted a lower percentage of applicants overall. Considering this fact, the so-called "hidden variable", departments of the university influenced the marginal values of the percentage of accepted applicants in a way that reversed the trend that existed in the data originally. In this way, this pre-trial investigation went down in history as the most famous case of appearance of the Simpson paradox and marked the relevance of the question of the search for the right view through the data and its correct interpretation. (Bickel & Hammel, 1975.)

In today's business world, people live in a data-driven environment. The role of data is becoming increasingly significant as decision-making is based on facts, statistical values, and the trends that shape them. However, there are certain phenomena where data can be misleading and direct to inaccurate conclusions. One of these phenomena is Simpson’s Paradox.

The first definition of it was formulated by Edward Hugh Simpson, a statistician and former cryptanalyst at Bletchley Park, who described the statistical phenomenon in 1951. Edward explained the paradox in terms of a scenario where groups of data demonstrate a certain trend, but that trend can be reversed or disappeared when groups are combined as one group. For this reason, understanding and identifying this paradox plays a crucial role in interpreting the data correctly. (Koswara et al, 2022.)

The nature of Simpson’s paradox explains why it could play a significant role in the data interpretation process and, as a consequence, in the decision-making process and strategic management in general. Based on this, it is fair to conclude that Simpson’s Paradox could emerge in completely different fields starting with medicine and ending with the social sciences. The business area is not an exception to this statement. That is why consideration of Simpson’s paradox in different aspects of business life considers such a relevant topic.

The research part of this thesis is aimed at conducting research where Booking.com, the online agency for booking accommodation and other travel products, was taken as a prominent representative of travel platform services. In specific terms, the research part consists of analyzing reviews of hotels in Helsinki on the presence of "hidden" variables that may affect the overall hotel rating system based on certain criteria such as, for example, length of stay, season of the year, or type of reviewer. The importance and worthiness of the study is expressed in the potentially possible optimization of the Booking.com recommendation system which will lead to an overall improvement in service as a travel platform.

2 Research description

2.1 Theme relevance

The core value of this study lies in evaluating how important and worthwhile considering Simpson's paradox is for Booking.com as a representative of travel platforms. There are four main reasons identified that support the relevance of this study, aimed at assessing the potential benefits of considering a paradox for travel platforms such as Booking.com.

The first and highly essential reason for consideration of the paradox as a relevant concept for the business is the potential improvement of the service. In today's highly competitive world, it is necessary to pursue any action aimed at gaining a competitive advantage through the continuous improvement of a product or service. In our example, considering paradox could be the basis for improving Booking.com's recommendation systems.

The second and equally meaningful reason is a potential source of insights. Important to have in mind that Simpson's paradox usually leads to the discovery of these kinds of "hidden" variables, which can affect the overall perception of data interpretation and, as a consequence, become a new source of insights. Because of the nature of Simpson’s paradox, the data shown are not all data that exist. Therefore, revealing paradoxes allows travel platforms in the research case to explore new insights into the data. (Grigg, 2018.)

The possibility of cross-industry use becomes a sufficient advantage of this paper, answering the relevant question of Simpson's paradox concept usage in the business world in general. Booking.com as a travel platform is an example of how each specific area of business is able to implement such practices. Thus, this kind of training practice can easily be transferred to other areas of business, e.g. finance, sales, or even purchasing.

Last but not least, the outcome of this study could lower the chance of data misinterpretation. Satisfaction with raw numbers or figures can lead to misinterpretation of data results and, as a result, sufficient losses of money and time. In the process of this research, critical issues related to the process of generating data, the causal model, have been addressed. In this way, key factors influencing the possible misinterpretation of data results are identified. Conducting proper data analysis enhances the connection with a proactive business strategy that seeks to control the situation in advance, rather than waiting for any response to act. (Henderson, 2020.)

All of the listed above factors indicate that the topic related to the consideration of Simpson's Paradox as a relevant concept for travel platforms is currently and professionally interesting for such research study.

2.2 Research question

The main purpose of this paper is to investigate whether Simpson's paradox is a useful and feasible phenomenon for owners of services in the travel industry. How could Simpson's paradox as a concept be used in the field of travel platforms and is it even worth paying attention to it? These are the main questions this study aims to answer. In this study, it was decided to focus exclusively on the field of travel platforms. Other areas of potential application are not taken into account but are considered in detail as examples in order to ascertain how the concept has already been applied in other areas. Assessment of this concept, as a potential way for improving services, explains both the academic and practical relevance of the research question. In order to answer the research question, it is necessary to assess how important this paradox is in other fields. Thus, in addition to explaining what Simpson's paradox is about, the theoretical framework will actively include various studies of the application of the paradox in other industries such as health care, marketing, and social sciences.

It was identified four main stages needed to be taken to complete the stated objective: founding the theoretical knowledge base, conducting the data collection process, statistical data analysis, and outcomes interpretation. Each of them answers the research question posed in the paragraph above in a gradual way.

2.3 Research design

The research is aimed at examining Booking.com hotel reviews for the presence of the Simpson’s paradox to determine the significance of this phenomenon for this service in general and how it could be applied to improve Booking.com's recommendation systems as an example.

In connection with the peculiarities of the case study, the following attributes are inherent. The research is to be conducted using quantitative research methods and a deductive approach, aimed at testing the existing theory. A causal or explanatory type of research is performed to understand the effect of the presence of a particular paradox on specific changes or improvements which could be achieved with it.

Case study research is used to describe the characteristics of a specific object - Booking.com reviews. And the research is not aimed at collecting large amounts of data to reveal patterns over time or place, but at collecting detailed data of a narrowly defined object, such as the presence of Simpson's paradox.

Relevance of concept is defined as the statistical significance of Simpson's paradox existing in a sample of the research by quite an accurate estimation of confidence intervals with bootstrapping method.

The validity of this research is based on the degree of accuracy of statistical measurement of the presence of Simpson's paradox and its relevance to travel platforms such as Booking.com.

2.4 Research methods

At the stage of specifying the research question, four main phases were determined: founding the theoretical knowledge base, conducting the data collection process, statistical data analysis, and outcomes interpretation. Three of them are directly related to the research process by conducting data analyses. Statistical data analysis could be presented in a form of three sub-steps aimed at detecting and verification of the true presence of Simpson's paradox. Each stage adheres to the research method describing the way of its implementation.

The first step is to carry out a data collection process. In order to make effective data analysis, it is necessary to have clean and well-prepared data for conducting reliable data analysis. At this stage, the main operation is to implement parsing data from Booking.com. Desired retrieved data are presented in the form of comma-separated values format. The process of parsing data from the Booking.com website is implemented with Python programming language. The data should consist of information about a certain amount of hotels, their score reviews, and three potential variables that could affect the overall perception of reviewing process. Variables are predefined as "type of reviewer", "length of stay", and "season time". This choice is explained by parsing the possibilities of the case company - Booking.com.

There is an issue between the first and second steps of the research that needs to be discussed beforehand. The problem involves different scales for measuring the data collected and the requirements that need to be met for proper statistical analysis. Booking.com suggests the Likert scale, which refers to the ordinal scale of measurement. At least an interval measurement scale or ratio scale is required for statistical analysis. The incompatibility of the scales comes from the fact that it is impossible for a reviewer evaluating a hotel to calculate the distance between such rating options as "good"' and "excellent". But despite everything mentioned above, the decision was made to adopt this grading scale. Admittedly, this method of application has some weaknesses and the findings need to be considered in

light of this fact. Likert scale of measurement would be considered as a ratio scale to conduct statistical analysis.

The next step is the data analysis, which consists of three sub-steps. Statistical analysis was chosen as the research method for the necessary manipulation of the data to reveal the presence of Simpson's paradox. In particular, all three sub-steps carry out using Python programming language as well.

The first sub-step is devoted to calculating the means for both overall or total and categorical groups of every variable. The mean was chosen as the basic statistical metric for the statistical analysis. It is calculated as the sum of the values divided by the number of these values. This step is needed for further comparisons between unique combinations of hotels in order to detect paradoxes. Calculation of mean using Python programming language is the main method at this stage of the research. Table … visualizes the implementation way of this step with an example of the “type of reviewer” variable. There are 5 categories and one overall group that is specific to this variable: families, couples, solo tourists, groups of friends, and business travelers (Table 1).

Families			
Hotel	Mean	№ of reviews	%
A	…	…	…
B	…	…	…
C	…	…	…
D	…	…	…
E	…	…	…

Couples			
Hotel	Mean	№ of reviews	%
A	…	…	…
B	…	…	…
C	…	…	…
D	…	…	…
E	…	…	…

Group of friends			
Hotel	Mean	№ of reviews	%
A	…	…	…
B	…	…	…
C	…	…	…
D	…	…	…
E	…	…	…

Business tourist			
Hotel	Mean	№ of reviews	%
A	…	…	…
B	…	…	…
C	…	…	…
D	…	…	…
E	…	…	…

Solo tourist			
Hotel	Mean	№ of reviews	%
A	…	…	…
B	…	…	…
C	…	…	…
D	…	…	…
E	…	…	…

Overall			
Hotel	Mean	№ of reviews	%
A	…	…	…
B	…	…	…
C	…	…	…
D	…	…	…
E	…	…	…

Table 1. Visualization of the first sub-step findings

The second sub-step is devoted to the detection of potential Simpson’s paradox based on calculated 95% limits for the means values and pure mean value of data using the

bootstrapping method. This method would be detailed explained in the next chapters. But briefly, the essence of it lies in estimating the confidence intervals of the mean values of groups with potentially detected Simpson's paradox by using its mean values calculated by random resampling with the replacement of sample items. The aim is to assess if it is a statistically reliable statement. Running the bootstrapping 9,999 times creates a foundation for confirmation presence of Simpson's paradox in a case if the key figures are stable and a maximum of five times are out of equation F1 < F2. In this case, it is possible to conclude about statistical reliability of the equation that F1 < F2. The application of confidence intervals with bootstrapping method allows concluding in a scientifically solid way that the difference between the calculated statistics within different groups is significant.

The point is to show how each hotel should be rated as a whole, based on criteria of variables, such as the type of reviewer from the last example. The comparison process refers to how the overall mean of Hotel A, for example, relates to the means of the other categories of a particular variable inherent to the same hotel A. It appears the question of reasonableness to integrate such a mechanism of recommendation system into service. Might be it explores the true focus of some hotels regarding its target customers based on such criteria ranking. During this stage listed issues are considered in more detail and scientific way.

The result of this sub-step is a list of cases containing hotels with Simpson's paradox or a result indicating the absence of the paradox in the data under study. It is an object for the next sub-step of conducting statistical analysis.

The third step is dedicated to drawing conclusions. After rigorous data collection and implementation of statistical data analysis, the stage of conclusions states the level of significance of considering Simpson's paradox for travel platforms. Here, the study gives a clear and exact answer to the research question and explains in detail the possibility of using this concept in the industry of travel platforms.

In addition to the main study, it is presented a set of recommendations aimed at complementing the findings with concrete solutions on how this information can be used in the real world.

2.5 Time frame

To get a complete picture of the research process with a defined time frame, below there is a table 2 schematically describing the step-by-step process of conducting it.

№	Research phases	Time
1	Data collection	1 week
2	Statistical data analysis:	3 weeks
2.1	• Calculation of means	1 week
2.2	• Detection of Simpson's paradox:	1 week
2.2.1	- between two hotels	2-3 days
2.2.2	- inside each hotel	2-3 days
2.3	• Verification of paradox: bootstrapping	1 week
3	Conclusions:	1 week
3.1	• Interpretation of received data	2-3 days
3.2	• Recommendations	2-3 days
	Total	5 weeks

Table 2. Time frame of research process

The table 2 describes time frames for each step of the research process. The time needed to implement the research is estimated to be about five weeks. A buffer margin of 30-40% of the objective time needed to complete the tasks has been inserted in the time frame of each phase to take into account the appearance of unforeseen difficulties.

3 Theoretical knowledge base

3.1 Simpson’s Paradox

3.1.1 Nature of Simpson’s Paradox

In the sciences of probability and statistics, there is the phenomenon where some groups of data demonstrate a certain trend, but the trend may be reversed or disappear altogether in a case where these groups are merged into one group. This description was given to Simpson’s Paradox, which was first formulated in 1951 by the statistician and cryptanalyst Edward Hugh Simpson. In his fundamental study, published in the same year, E. H. Simpson called attention to simple facts about fractions, which have a wide range of surprising applications. These are explained by the close relationship between proportions, percentages, probabilities, and their representations in the form of fractions. (Wagner, 1982.)

Speaking in application terms, Simpson's Paradox occurs when there are so-called “hidden” variables that break up the data into several separate distributions. Such a latent variable is aptly called a lurking variable, and it is often difficult to identify. That is why the paradox has received high relevance in detection and consideration in social-science and medical-science statistics. (Holt, 2016.)

The practical role of this phenomenon arises from occasions where frequency data are unreasonably interpreted in a causal way, and as a result, leads to misunderstandings about the true meaning of these data and incorrect conclusions. (Pearl, 2000.)

Thus, Simpson's paradox has established its place in history and science as a phenomenon that shows the importance of skepticism in interpreting data in relation to the real world, and the danger of oversimplifying and overlooking key aspects of data analysis through a single point of view. (Grigg T, 2018.)

3.1.2 Confounding factors

The existence of Simpson’s paradox attributes to the presence of a so-called confounding factor explaining the paradoxical feature of this phenomenon. In the broad sense of the term, it is such a factor that distorts the direct relationship between the groups of data examined and the outcome variable. This factor is sometimes called a background factor or confounder. It takes two main conditions to meet a background factor: the groups differ on the background factor, and the background factor influences the outcome variable. (Norton & Divine, 2015.)

There is an interesting example that vividly describes the presence of these two conditions. The essence of the example was data from court cases that resulted in death sentences. The groups are criminals and the background factor is the race of the victim. For African American offenders 85.2% (2151/2526) of victims are African American while for white offenders 4.2% (100/2372) of victims are African American. For the murder of an African American victim, 0.5% (12/2251) of offenders receive the death sentence, while in the case of white victims, 2.5% (65/2647) of offenders receive the death sentence. Thus, the two main conditions aimed at confounding are met. (Norton & Divine, 2015.)

The issue of identifying confounding factors is relevant, as this process aims to rationalize the findings of the data analysis and clarify its paradoxical features of it. Thus, in a case of conducting detailed marketing research H. James Norton and George Divine propose a step-by-step plan of action aimed at preventing exposure of Simpson's paradox and the resulting inaccurate conclusions:

1. Having statistics will add credibility and facilitate the process of design, data collection, and analysis before the study starts;

2. A critical approach to evaluating data is always the basis of a robust study, especially with data from retrospective or observational studies;

3. Preliminary identification of potential confounding factors that may affect the interpretation of the data results;

4. In a case where variables are in the causal relationship, Neither of these variables is a confounding factor and no adjustment should be made to it;

5. Conduct statistical analysis to check for confounding variables and make a reasonable interpretation of the obtained data. (Norton & Divine, 2015.)

Verifying the presence of confounding variables is an extremely important process, as missing corrections for such variables can lead to incorrect conclusions. And the consequences of such inferences can be detrimental in many very important areas such as medicine, marketing, business, and so on. (Norton & Divine, 2015.)

3.1.3 The most famous example

The first case related to UC Berkeley gender bias was briefly covered in the introduction part of this paper. The point of the case was that admissions data from the fall of 1973 showed that male applicants were more likely to be accepted than female applicants. The

difference was so large that it was unlikely to be due to chance and was the reason for a judicial inquiry into gender discrimination. (Bickel et al. 1975.)

Total	All		Men		Women	
	Aplicants	Admitted	Aplicants	Admitted	Aplicants	Admitted
	12,763	41%	8,442	44%	4,321	35%

Table 3. Admission result of UC Berkeley in the 1973 year (Bickel et al. 1975.)

A detailed investigation reveals that there is a definite trend in the consideration of faculties in this case. This shows that females were more likely to apply to faculties with a high rejection rate, while males, in contrast, applied to less competitive faculties with a higher admission rate. The combined and adjusted data showed "a small but statistically significant bias in favor of women". The data from the six largest departments are listed in the second table. (Bickel et al, 1975.)

Department	All		Men		Women	
	Aplicants	Admitted	Aplicants	Admitted	Aplicants	Admitted
A	933	64%	825	62%	108	82%
B	585	63%	560	63%	25	68%
C	918	35%	325	37%	593	34%
D	792	34%	417	33%	375	35%
E	584	25%	191	28%	393	24%
F	714	6%	373	6%	341	7%
Total	4526	39%	2691	45%	1835	30%

- greater percentage of successful applicants than the other gender
- greater number of aplicants than the other gender

Table 4. Admission results of UC Berkeley based on faculties in 1973 year (Bickel et al. 1975.)

In the case of detailed data consideration, the issue of gender discrimination may be impaired by the observed findings of bias in favor of minorities. Based on table 2, the tendency becomes more notable. The green color is a sign of gender minorities representing a higher percentage of successful applicants. Received results completely change the way the data are interpreted. This example clearly explains the relevance and importance of taking into account Simpson’s paradox as a potential phenomenon that misled the true meaning of data. (Bickel et al. 1975.)

3.2 Application of the concept in the healthcare industry

3.2.1 Treatments for kidney stones case

Most probably, the second most famous example came from the healthcare industry in the 1986 year. The medical study was aimed at comparing the success rates of two treatments for kidney stones: open surgery and percutaneous nephrolithotomy. However, the study took into account the size of the stones being treated and table 3 shows the success rates of those treatments for two types of kidney stones: small stones (less than 2 cm) and large stones (more than 2 cm). One thing that has to be noted is that the treatment methods for different types of stones differ in their degree of medication effect. (Charig et al. 1986.)

Treatment / Stone size	Treatment A	Treatment B
Small stones	Group 1 **93%** (81/87)	Group 2 87% (234/270)
Large stones	Group 3 **73%** (192/263)	Group 4 69% (55/80)
Both	78% (273/350)	**83%** (289/350)

Table 5. Success rates of treatments (Charig et al. 1986.)

Based on the table, it is clear to conclude that there are two main points of view in this study: analyzing exposure of treatments separately for types of stones, and assessing the overall success rate of treatments. Both points of view lead to different outcomes and, as a result, to paradoxical conclusions. In the first case, treatment A is more effective for any kind of stones, while considering the total values of treatment B seems more effective compared to treatment A. And after a detailed examination of this case, two main peculiarities were defined. The first is about ignoring the size of the group, which is very different, in the research. Physicians have a tendency to prescribe better treatment A for patients with large stones and less effective treatment B for patients with small stones. Therefore, groups 3 and 2 predominate in the final data, rather than the two much smaller groups 1 and 4. And the second is that a hidden variable, stone size, has a big influence on the ratios. The success rate is more strongly influenced by case complexity than by treatment choice. Therefore a group of patients with large stones using treatment A performs worse than a group with small stones, even if the latter have used less effective treatment B. (Charig et al. 1986.)

An important factor in reversing the success rate is that the likelihood of open surgery or percutaneous nephrolithotomy varied according to the size of the stones. In an observational research study comparing treatments, patient characteristics such as age or severity

of the condition are likely to have influenced the initial choice of treatment. Therefore, any difference in the treatment methods can be explained by these underlying factors.(Charig et al. 1986.)

3.2.2 Epidemiological hospital patients case

The example of kidney stone treatments was the first highly publicized case of Simpson's Paradox detection. However, there are a large number of studies are ongoing in order to examine the impact of this concept in the field of health care. In this way, Reintjes et al. performed a prospective, multicenter study in eight hospitals in the Netherlands to assess the possibility of standardized surveillance for nosocomial infection. (Reintjes et al. 2020.)

The aim of this nonexperimental research was to examine the influence of possible risk factors on the development of urinary tract infections (UTI) in gynecologic patients in those hospitals. It is known that the occurrence of UTI is associated with a number of risk factors, ranging from the individual characteristics of the patient's body and his age, for example. Because all of these factors are independent, it adds a confounding effect to the study Application of the concept in the marketing industry. Antibiotic prophylaxis, an effective treatment based on randomized clinical trials, is used to prevent UTIs. The relationship between UTIs and antibiotic prophylaxis was identified by univariate and stratified analyses. When multivariate analyses were conducted with conditional logistic regression. Scientists came up with the following data presented in the table below. (Reintjes et al. 2020.)

Patients from all eight hospitals					
AB Prophylaxis	UTI	no-UTI	Total	RR	95% CI
Yes	42 (29%)	1237 (37%)	1279	0.7	0.5-1.0
No	104 (71%)	2136 (63%)	2240		
Total	146	3373	3519		

AB Prophylaxis = Antibiotic Prophulaxis
N = 3,519 (percentages)

Table 6. Overall Data on Urinary Tract Infections (UTI) and Antibiotic Prophylaxis, from eight Hospitals in The Netherlands, 1992–93 (Reintjes et al. 2020.)

Based on the data obtained during the study and presented in Tables 6, and 7, it is possible to conclude that the univariate analysis regarding the effect of antibiotic prophylaxis showed a relative risk (RR) of 0.7%, when after stratification for hospitals with a low (<2.5%) versus a high percentage (>2.5%) of UTI, the relative risks were about 2.6% and 2.0 % respectively. This statement says about the presence of Simpson's paradox and potential incorrect data interpretation. (Reintjes et al. 2020.)

Patients from four hospitals with Low Incidence of UTI (<2.5%)					
AB Prophylaxis	UTI	no-UTI	Total	RR	95% CI
Yes	20 (80%)	1093 (60%)	1113	2,6	1.0-6.9
No	5 (20%)	715 (40%)	720		
Total	25	1808	1833		
Patients from four hospitals with High Incidence of UTI (>2.5%)					
AB Prophylaxis	UTI	no-UTI	Total	RR	95% CI
Yes	22 (18%)	144 (9%)	166	2.0	1.3-3.1
No	99 (82%)	1421 (91%)	1520		
Total	121	1565	1686		
AB Prophylaxis = Antibiotic Prophulaxis					
N = 3,519 (percentages)					

Table 7. Data on Urinary Tract Infections (UTI) and Antibiotic Prophylaxis (AB-prophylaxis) Stratified by Incidence of UTI per Hospital in Two Strata of four Hospitals in The Netherlands, 1992–93 (Reintjes et al. 2020.)

Stratified analysis of the research data shows that the association between antibiotic prophylaxis and UTI has a relative risk greater than 1 in all strata, which is the expected result of a nonexperimental study. It is related to the fact that in clinical practice the decision to take antibiotics prophylactically is often made based on the risk of a patient developing a UTI. On the contrary, a single-factor analysis of the overall data showed that the association between antibiotic prophylaxis and UTI has a relative risk value of less than 1, which is consistent with the experience of clinical trials. Therefore, data for specific strata show an effect opposite to that seen in the full, unstratified data set. This phenomenon is the Simpson paradox. (Reintjes et al. 2020.)

3.3 Application of the concept in the marketing industry

3.3.1 Lipovetsky and Conklin study case

Despite the fact that the nature of Simpson's paradox is well known and researched, real-work practices show that this paradox is pretty hard to be identified, explained, and solved. The peculiarity of this is that the paradox can be detected by analyzing information in completely different areas. And the field of marketing is no exception. (Lipovetsky & Conklin, 2006.)

In the paper of Lipovetsky and Conklin, the authors consider and illustrate some of Simpson's paradox cases in marketing research fields. In table 8 the data is represented by five age groups of consumers of a particular brand among all buyers of the product during the

first and second quarters of the year. The key issue of these data lies in the fact that the brand's share change values of each age group increase when the total value of the brand's share change decreases. On closer inspection, it is possible to see that for the first age of 18-24 years old market share was increased by almost 29 points compared to the second quarter. This tendency spreads to the rest age groups, but in spite of it, total values drop by 98.56–100.00, or by 144 points. (Lipovetsky & Conklin, 2006.)

Category	Measure and notation	Grouping 1					Total
		age 18-24	age 25-34	age 35-44	age 45-59	age 60+	
Quarter 1	Prefer the brand, c	25	20	30	20	28	113
	Buy the product, q	195	450	495	650	650	2440
	Brand's share, p	12,821	4,444	6,061	3,077	2,769	4,631
Quarter 1	Prefer the brand, c	27	21	36	25	22	131
	Buy the product, q	210	470	590	810	790	2870
	Brand's share, p	12,857	4,468	6,102	3,086	2,785	4,564
Q2 / Q1 %	Brand's share change	100,282	100,542	100,671	100,306	100,571	98,563

Table 8. Sample A: Distribution by ages of the consumers of a specific brand among all consumers of the product (Lipovetsky & Conklin, 2006.)

This is the explicit representation of Simpson's paradox. The particular case could be explained by various hidden confounding variables. In some cases, it takes intuition to identify these factors affecting the whole picture of data interpretation. In this example given data have been aggregated by age categories, which introduces ambiguity into the data analysis. Might be it worth considering simplifying the data by some other potential confounding variables like the sex of consumers or rather than age groups, there is a characteristic of the financial market over time. Based on it, it is fair to conclude that each situation is absolutely unique and different. It takes creativity and, in some cases, the intuition of marketers working with a particular case of existence of Simpson's paradox. (Lipovetsky & Conklin, 2006.)

3.3.2 Mobile advertisement case

In 2018 year, Kristen Rivers wrote an excellent short blog post about addressing the Simpson paradox in mobile advertising, describing why digging a little deeper can avoid misinterpreting data. Kristen is currently the CEO of AdInMo, which is in charge of Dynamic In-Game Advertising for mobile games. During his career, Kristen has had time to work with such major companies as "Paramount Pictures" and "Apple" in the field of advertising. In her article, Kristen gives an example of how the "Simpson paradox" delusion can cause marketers to incorrectly interpret the results of their campaigns. The price of error is expensive in today's reality. In the simplified model of mobile advertising example below, the data

set is taken from an install campaign measuring install rate performance, with a breakdown of the data across the two main mobile platforms. (Rivers, 2018.)

Platform	Impressions (000's)	Installs (000's)	Install Rate
Android	267	135	50,6%
IOS	143	77	53,8%

Table 9. Install rate performance by platforms (Rivers, 2018.)

At first glance, it is obvious that iOS has a higher Install Rate than Android in this campaign, and therefore the advertising campaign should be focused more on the ISO platform in the form of allocating a larger budget. However, not everything is as it seems at first sight. With a deeper dive into the data broken down by sub-platform in the form of tablets and phones, the data can turn to the opposite. And the overall picture of marketing analysis, as well as the follow-up, will look very different as presented in table 10. (Rivers, 2018.)

Phone	Impressions (000's)	Installs (000's)	Install Rate
Android	245	115	46,9%
IOS	88	35	39,8%

Tablet	Impressions (000's)	Installs (000's)	Install Rate
Android	22	20	90,9%
IOS	55	42	76,4%

Table 10. Install rate performance by sub-platforms (devices) (Rivers, 2018.)

Based on a more detailed analysis of the data with a subdivision by device, it is clear that the number of Android installations surpassed iOS on both sub-platforms. This is where Simpson's paradox occurs, presented as the inverse results of the aggregated data and divided by the so-called confounding variable, presented as a subdivision into phone and tablet devices. To fully understand Simpson's paradox, the potential presence of a confounding or hidden variable concept must be taken into account. In this case, the sub-platform introduces a bias: 38% of iOS impressions went to tablets, while only 8% of Android impressions went to tablets. This example shows that marketers should consider whether the distribution of displays from tablets and phones accurately reflects the target audience. (Rivers, 2018.)

Of course, this is an oversimplified example, but it's based on real-world experience. In this case, a completely different variable, such as the gender or age of the target audience, could have taken on the role of a hidden variable. Sometimes determining which variable has such an impact on reverse results can be quite difficult. Intuition gained by experience assists in this confusing process. (Rivers, 2018.)

3.4 Application of the concept to the social science

In 2018 year, a group of researchers published the article “Can you Trust the Trend? Discovering Simpson’s Paradoxes in Social Data”, which explores how Simpson's paradox affects the analysis of trends in social data. Researchers presented a statistical method by which Simpson's paradox can be automatically detected in data by comparing statistical trends in aggregate data with trends in disaggregated subgroups. The study applied this approach to the popular question-answering platform “Stack Exchange” in order to analyze the factors that influence the answerer's performance. This particular metric in this case is determined by the probability that the answer written by the user will be accepted by the asker as the best answer to his question. During the analysis, Simpson's paradox was identified among the potential variables several times. The obtained results allowed to look at the social behavior of Stack Exchange users in a new way, as well as to evaluate the effectiveness of the automatic detection of Simpson's paradox. (Alipourfard et al. 2018.)

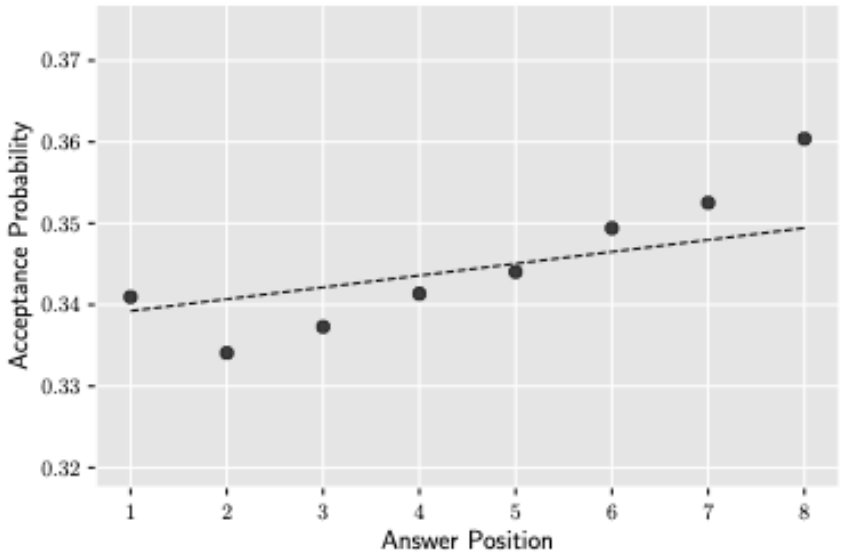
Thus, the study describes a method for identifying Simpson's paradoxes by analyzing trends in social data. The essence of this statistical method is to find a pair of variables, or so-called Simpson pairs, such that the trend in some results observed in the aggregate data disappears or reverses when the data are disaggregated into distinct subgroups by a second explanatory variable. The article provides a detailed description of the mathematical analysis that reveals two necessary conditions for the paradox to occur. The first is the presence of a correlation between the independent variable and the conditioned variable. The second is that the value of the outcome variable differs in the conditioned subgroups. (Alipourfard et al. 2018.)

Stack Exchange is a platform operating since 2008 as a forum where people can ask and get answers to both technical and non-technical questions. The principle of Stack Exchange is simple: any user can ask a question that can be answered by others. Users can also vote for answers they find useful, and the questioner can accept one of the answers as the best answer to the question. In this way, the Stack Exchange community collectively accumulates knowledge. (Alipourfard et al. 2018.)

A total of about 9.6 million questions were accumulated over the lifetime, of which about half had an accepted answer and satisfied the condition that only those questions with two or more answers would be included. To fully understand the various variables and factors affecting the correctness of Stack Exchange user answers, the relationship between user attributes and the probability that a user's answer was accepted by the questioner as the best answer to his question was examined. Therefore, for each answer, a list of variables describing the answering user's actions and attributes was created. (Alipourfard et al. 2018.)

There are nine main variables that influence the success of the response to becoming accepted as the best Simpson's Paradox in the recommendation system:

1. Reputation (overall user contribution);
2. A number of answers;
3. Tenure;
4. Percentile (User's percentile rank based on tenure);
5. Time since the previous answer;
6. Session length;
7. Answer position;
8. Words;
9. Lines of codes. (Alipourfard et al. 2018.)

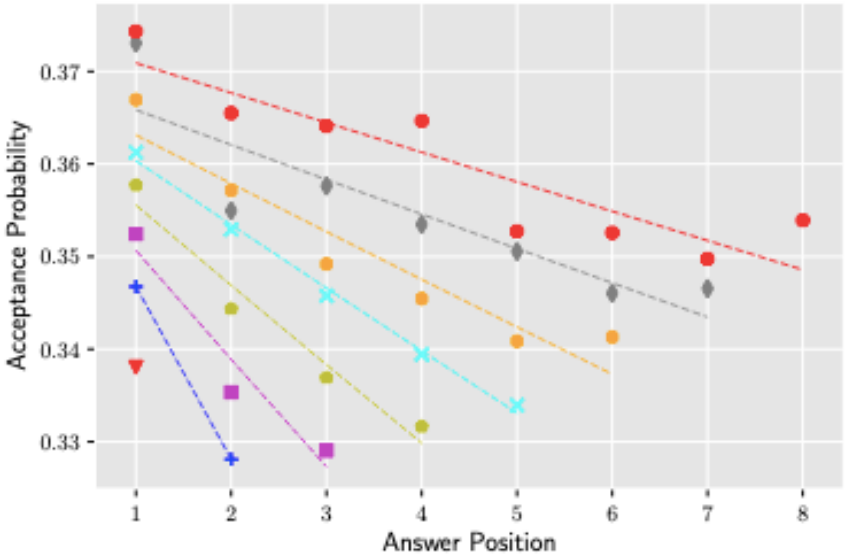


(a) Aggregated Data

Figure 1. The probability of an answer is accepted as the best answer to a question as a function of its position within the user's activity session with aggregated data (Alipourfard et al. 2018.)

Session Length	1	2	3	4	5	6	7	8
Data points	7.2M	2.6M	1.3M	0.7M	0.4M	0.3M	0.2M	0.1M

Table 11. The number of data points in each group. (Alipourfard et al. 2018.)



(b) Disaggregated Data

Figure 2. The probability of an answer is accepted as the best answer to a question as a function of its position within the user's activity session with disaggregated data. (Alipourfard et al. 2018.)

Simpson's paradox in the Stack Exchange data indicates in Figure 1 that in the aggregate condition the response acceptance probability calculated tends to increase, indicating that responses written later in the session and as a consequence are more likely to be accepted as better responses. However, when looking at the disaggregated state by session duration in Figure 2, the trend is reversed. This means that within each time group colored by different colors, individuals who take less time to respond tend to have higher acceptance rates. (Alipourfard et al. 2018.)

3.5 Summary

The above studies on the application of Simpson's paradox in various fields have shown the high applicability of this phenomenon to data analysis, causal relationships, and decision-making. Despite the different experimental environments, all of the above examples indicate a high level of relevance and applicability of the concept in order to obtain exhaustively and, most importantly, truthful data results.

In the process of analyzing and comparing the studies presented in the theoretical framework, the conclusion can be made that most of them were not conducted in a laboratory environment. In other words, in most of the cases related to healthcare, for example, the main aim of the research was not to assess the impact of Simpson's paradox, but rather to

investigate a certain medical effect, such as which medicine treats the kidneys better or to assess the effect of possible risk factors on the development of urinary tract infections. Simpson's paradox acted as a phenomenon that significantly influenced the results of this study, and was a kind of side effect of the conducted research. In other words, in the original design of those studies, this concept did not figure as a core element to be investigated.

While research in the marketing field analogously shows consideration of Simpson's paradox as a side effect discovered during certain marketing research. These cases aim to inform the marketing community how this concept can affect the perception of data analysis. Even fairly basic examples like mobile advertising campaigns demonstrate that the paradox can be revealed, both in fairly simple to grasp cases and in more complex ones, such as the development of urinary tract infection teachings.

However, with a proper and purposeful study of Simpson's paradox, patterns of its occurrence and detection become visible. Thus, a group of researchers presented a statistical method in social science by which Simpson's paradox can be automatically detected in data by comparing statistical trends. This study leads to the idea that the given concept is not something hard to catch or unpredictable. Moreover, in this way, it opens up the possibility to consider this paradox as a completely controllable phenomenon that can be analyzed and monitored. However, there are only a few studies of this kind that focus specifically on paradox detection, which is a significant shortcoming in the field of data analysis. The main conclusion is that with enough research, the paradox is practicable and tangible to detect. Hence, there is an increased demand for this kind of research, not only in the social sciences, as presented in the theoretical framework, but also in other fields.

Therefore, the level of relevance and importance of conducting research on the Simpson paradox in the field of travel platforms is greatly increased. Booking.com, as a clear representative of this sphere, is perfect for this type of research. The following chapter details the step-by-step steps for analyzing hotel reviews for Simpson's paradox.

4 Research process

4.1 Data collection

The first step was to perform a data collection process. To conduct effective data analysis, it is necessary to have clean and well-prepared data in order to conduct reliable data analysis. Initially, during the planning process of this study, it was planned to collect the required data manually by performing the HTML data parsing procedure directly from Booking.com. The desired extracted data was to be presented in a comma-separated value format while the process of parsing data from Booking.com was implemented using the Python programming language. The data should consist of information about a certain number of hotels, their rated reviews, and three potential variables that could affect the overall perception of the review process. The variables are predefined as 'reviewer type', 'length of stay, and ‘time of year’. This choice is explained by an analysis of the capabilities of the company working with Booking.com as an example.

However, in order to give a more substantial level of accessibility to the test for the reviewers, it was decided to use a secondary source of data. Based on it, the decision was admitted to use secondary data in a form of an already existing dataset. Due to the specifics of the required data, a dataset consisting of half a million hotel reviews was found ready to use. A more detailed look at the contents of this dataset is presented in chapter 5. The data was also, as originally intended, collected on the Booking.com platform. The dataset was uploaded to the public domain in comma-separated value format by Jonathan Oheix, Jiashen Liu, and Ahmed Shahriar Sakib on the Kaggle platform. The dataset consists of seventeen columns describing detailed information about both the hotels where reviews were left and the visitors by whom those reviews were left. (Oheix et al. 2022.)

There is an issue involves different scales for measuring the secondary data and the requirements that need to be met for proper statistical analysis. Booking.com suggests the Likert scale, which refers to the ordinal scale of measurement. At least an interval measurement scale or ratio scale is required for statistical analysis. The incompatibility of the scales comes from the fact that it is impossible for a reviewer evaluating a hotel to calculate the distance between such rating options as “good”' and “excellent” or from 1.0 to 10.0. But despite everything mentioned above, the decision was made to adopt this grading scale. Admittedly, this method of application has some weaknesses and the findings need to be considered in light of this fact. Likert scale of measurement would be considered as a ratio scale to conduct statistical analysis.

4.2 Data processing

The second phase of the study is to convert the imported dataset into the format necessary for further manipulation of the data. Thus, with the help of the Python programming language, a number of commands were written to convert the data in order to calculate the mean values and perform bootstrapping procedures. The code of these transformations can be found in detail in Appendix 2. Based on the data we decided to consider five hotels with the highest number of reviews: Britannia International Hotel Canary Wharf, Copthorne Tara Hotel London Kensington, DoubleTree by Hilton Hotel London Tower of London, Park Plaza Westminster Bridge London, Strand Palace Hotel.

A detailed review of the code reveals several basic commands aimed at preparing the data for the calculation of the mean and bootstrapping mean values. The first task, presented in appendix 2, was to remove all unnecessary columns from the dataset that do not relate to the examined variables in any way. The following step involves parsing the Tags column. The reason for it is that originally the column contained all possible attributes of the visitor and it was necessary to split them for analysis between the examined variables. Thus, cell four of appendix 2 presented the commands that perform this task. This produced four variables: traveler type, room type, number of nights, and season based on the review date. These variables were used to calculate the mean and bootstrapping means of hotel reviews, which are presented in cells 11-14 of appendix 1. The obtained data are presented in a more detailed way in Chapter 6.

Calculation of the mean values was performed in order to familiarize with the preliminary review of the given data. However, due to the fact that the obtained data are not statistically reliable, the decision was reached to carry out a bootstrapping procedure to ensure the academic reliability and statistical significance of the research.

The function that produces the bootstrapping procedure was scripted in cell eleven of appendix 2. Bootstrapping could be described as a statistical procedure that performs random sampling with replacement in a form of a test or any kind of metrics. This method allows for the calculation of a variety of statistical measures such as standard errors, estimation of confidence intervals, and so on. The main feature of this technics is an alternative approach to traditional hypothesis testing. The core meaning of bootstrapping is based on the resampling of data over and over again to create multiple simulated samples. Resampling of data occurs from the main sample with a random representation of that sample. Each simulated sample has its own sampling distribution and statistical properties such as mean or standard deviation. Based on a predefined certain statistical metric, a new given simulation sample constructs a new distribution of this metric. This procedure uses these sampling

distributions as the foundation for the estimation of confidence intervals and hypothesis testing. (Frost, 2022.)

The process of bootstrapping requires compliance with some attributes. Thus, it takes to keep an equal probability of randomly drawing each original data point for integration into the new samples of data. In addition, the procedure can select a data point more than once for a new data sample. For this reason, the process is described as resampling with replacement. The newly obtained data samples should have the same size as the original one. (Frost, 2022.)

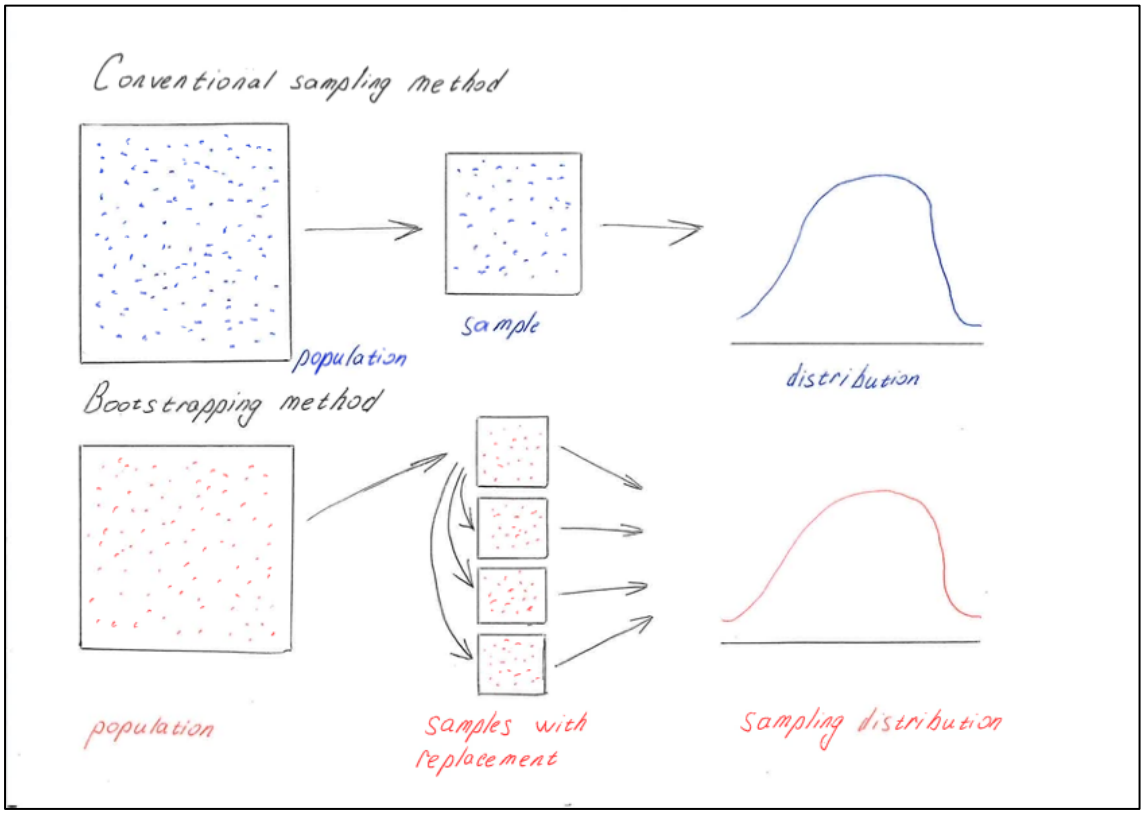


Figure 1. The main difference between conventional and bootstrapping sampling methods

The figure depicted the main difference between the conventional statistical analysis colored by blue items and bootstrapping method with yellow color in a way of resampling method. The core difference between these approaches is that bootstrapping method has an extra step, which ensures a high level of statistical reliability. This step is represented by the action of creating a series of samples based on an initial sample taken from the population.

In statistics and machine learning particularly, there are two main types of bootstrapping methods: parametric and non-parametric bootstrap. The first one uses a predefined distribution parament. This means that an assumption about what distribution the sample has

must be drawn in advance. While another method, non-parametric does not require the parameter of distribution to be known beforehand. For this reason, this type of bootstrap method operates without making assumptions about the nature of the sample distribution. (Rawat, 2021.)

However, the bootstrapping method has both advantages and disadvantages in its use. The main reason for using this method is its functionally simpler way of estimating the value of statistics that are otherwise too difficult to calculate using traditional methods. This method allows the accuracy of the model to be checked without much hassle, and simpler steps can be taken. It is also one of the best-known sampling methods, requiring no up-front assumptions for its concept to work. Unlike traditional methods that rely on a theoretical concept to produce results, the bootstrapping method simply observes the results and works through them to produce accurate results. The method does not fail even when the theory is not supported by practical observation and is thus very advantageous in this aspect. (Rawat, 2021.)

4.3 Data analysis

Part of the data analysis is based on the formation of tables with represented mean values through a bootstrapping procedure. The essence of it is to present the five hotels selected for analysis with their mean values in the form of tables for analysis of the three variables under examination: type of reviewer, the season of the year, and length of stay of the reviewer in the hotel. Thus, the expected result will be three tables, each of which will examine the data for the presence of Simpson's paradox.

By definition, Simpson's paradox is an effect in statistical analysis when two groups of data, each of which exhibits the same directional relationship, are combined and the direction of the relationship reverses when the groups are combined. Based on this, Simpson paradox detection technique is based on the condition that if the 95% range of the total mean does not overlap with the majority of subgroups with the 95% range of the group mean, then it indicates the existence of the Simpson paradox. In other words, in the case of analysing the type of reviewer variable, there should be three of the five 95% ranges of subgroups that will not overlap with the total 95% range, then it can be stated that Simpson's paradox exists in the data under study.

5 Data description

As mentioned earlier, the decision to use secondary data in this study was made to give reviewers more opportunity to conduct similar studies and to see firsthand the authenticity of the data and the findings. The point is that the data have absolutely the same reliability as it has an identical source of origin. In this case, both data collection paths are based on the fact that the data were collected from Booking.com. Meanwhile, the secondary data was already provided on the Kaggle platform, which is open for public use. Therefore, due to current circumstances, it is more rational to use pre-collected data.

This dataset consists of 17 columns detailing information about both the hotels where reviews were left and the visitors by whom those reviews were left:

1. Hotel_Address
2. Additional_Number_of_Scoring
3. Review_Date
4. Average_Score
5. Hotel_Name
6. Reviewer_Nationality
7. Negative_Review
8. Review_Total_Negative_Word_Counts
9. Total_Number_of_Reviews
10. Positive_Review
11. Review_Total_Positive_Word_Counts
12. Total_Number_of_Reviews_Reviewer_Has_Given
13. Reviewer_Score
14. Tags
15. days_since_review
16. lat
17. lng

	Hotel_Address	Additional_Number_of_Scoring	Review_Date	Average_Score	Hotel_Name	Reviewer_Nationality	Negative_Review
0	s Gravesandestraat 55 Oost 1092 AA Amsterdam ...	194	8/3/2017	7.7	Hotel Arena	Russia	I am so angry that i made this post available...
1	s Gravesandestraat 55 Oost 1092 AA Amsterdam ...	194	8/3/2017	7.7	Hotel Arena	Ireland	No Negative
2	s Gravesandestraat 55 Oost 1092 AA Amsterdam ...	194	7/31/2017	7.7	Hotel Arena	Australia	Rooms are nice but for elderly a bit difficul...
3	s Gravesandestraat 55 Oost 1092 AA Amsterdam ...	194	7/31/2017	7.7	Hotel Arena	United Kingdom	My room was dirty and I was afraid to walk ba...
4	s Gravesandestraat 55 Oost 1092 AA Amsterdam ...	194	7/24/2017	7.7	Hotel Arena	New Zealand	You When I booked with your company on line y...
...	...	...	...	...	...	...	...

Table 12. “515K Hotel Reviews Data in Europe” (Oheix et al. 2022.)

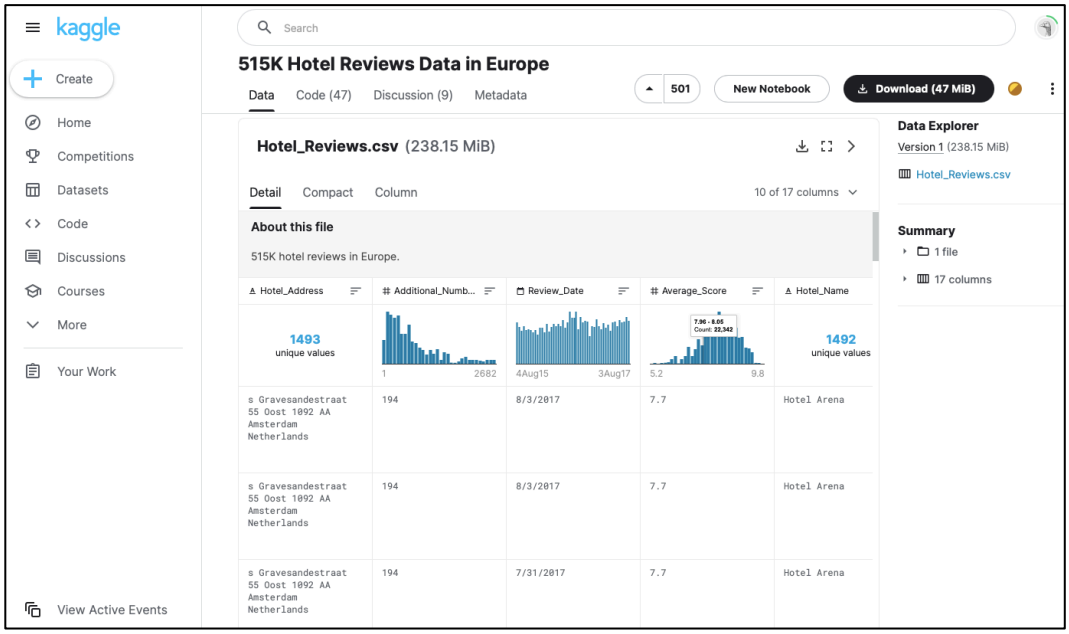


Figure 3. Page of “515K Hotel Reviews Data in Europe” dataset on the Kaggle platform(Oheix et al. 2022.)

The figure shows a screenshot of the Kaggle platform web page, where the data has been provided public access. This dataset is presented quite broadly and contains, in addition to the data required for the study, a lot of non-essential data, which can be disregarded. Furthermore, there is also data that needs to be transformed. The obvious example is the tags column, which will need to be split up into separate columns.

6 Data processing phases

This chapter presents the findings obtained during the research. Data processing involves some main steps. In the beginning, the required metrics, which are necessary to conduct a rigorous research study for the identification of Simpson's paradox were calculated. These metrics include the mean values of the reviews for three variables: type of reviewer, season, and length of stay. It was also decided to use bootstrapping procedure for calculating the 2.5% and 97.5% limits showing the range for 95% of the means. In this case, it will be a more reliable metric for calculating the presence of this paradox. It was also produced in order to give statistical reliability to the study.

The procedure for finding the bootstrap confidence interval for the mean value is the following:

1. Make N samples from the original sample with replacement. In this case, the library Scipy.stats is used to import the Bootstrap function. And the default number of samples with replacement of this function is 9,999.

2. Next, for each sample, find the sample mean and these sample means are arranged in ascending order. By doing this it is possible to get a pure mean value by bootstrapping procedure, which is presented in the column boot_mean.

3. To get a 95% confidence interval, it needs to find the middle 95% of the sample means. To do this, find the averages at the 2.5% and 97.5% percentiles. Using the scipy.stats library described above, these steps are performed automatically and presented in the boot_low, and boot_high columns, respectively. The values of these items are the lower and upper limits of the 95% bootstrap interval for the true mean. It should be noted that each time we go through this procedure, it is possible to get a slightly different bootstrap interval.

These limits must then be applied to determine whether Simpson's paradox s exist in the data under research or not in the next chapter. The reviews column indicates the sample size and is different in each case. So for example the Britannia International Hotel Canary Wharf with the type of reviewers as a group of friends has 460 reviews, while the sample size of the Copthorne Tara Hotel London Kensington has 549 reviews. The tables also include a "%" column indicating what percentage of each hotel belongs to the subgroup under research. For example, Table 13 demonstrates that at Britannia International Hotel Canary Wharf, the group of friends staying at the hotel out of the total number of visitors is only 10%. This metric was added solely for the purpose of curiosity to possibly discover interesting patterns in the research.

Group of friends	total: 2340	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.90630	6.90214	6.72019	7.08409	460	10%
2_Copthorne Tara Hotel London Kensington		8.20801	8.20546	8.07547	8.33544	549	16%
3_DoubleTree by Hilton Hotel London Tower of London		8.77216	8.76366	8.62654	8.90077	388	12%
4_Park Plaza Westminster Bridge London		8.78790	8.78055	8.65887	8.90222	496	12%
5_Strand Palace Hotel		8.31767	8.31097	8.17272	8.44922	447	11%

Solo traveler	total: 4352	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.47841	6.47943	6.38282	6.57605	1649	37%
2_Copthorne Tara Hotel London Kensington		7.74371	7.74214	7.61698	7.86730	636	18%
3_DoubleTree by Hilton Hotel London Tower of London		8.46356	8.45752	8.30496	8.61007	343	11%
4_Park Plaza Westminster Bridge London		8.44101	8.43586	8.29963	8.57210	534	13%
5_Strand Palace Hotel		7.86630	7.86302	7.77193	7.9541	1190	29%

Family with young children	total: 2212	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.82500	6.82429	6.56445	7.08413	252	6%
2_Copthorne Tara Hotel London Kensington		8.10242	8.09775	7.92284	8.27266	289	8%
3_DoubleTree by Hilton Hotel London Tower of London		8.57331	8.56319	8.37336	8.75302	251	8%
4_Park Plaza Westminster Bridge London		8.68714	8.68320	8.60052	8.76588	1151	29%
5_Strand Palace Hotel		8.22268	8.21793	8.03829	8.39757	269	6%

Couple	total: 9571	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		7.04226	7.04125	6.95597	7.12653	2068	46%
2_Copthorne Tara Hotel London Kensington		8.18804	8.18658	8.11789	8.25526	1822	53%
3_DoubleTree by Hilton Hotel London Tower of London		8.65792	8.65763	8.59585	8.71942	2072	66%
4_Park Plaza Westminster Bridge London		8.65193	8.65075	8.57324	8.72826	1525	38%
5_Strand Palace Hotel		8.19232	8.19214	8.12596	8.25832	2084	50%

Family with older children	total: 842	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.92805	6.91220	6.50854	7.31585	82	2%
2_Copthorne Tara Hotel London Kensington		8.03529	8.02327	7.78007	8.26647	170	5%
3_DoubleTree by Hilton Hotel London Tower of London		9.01413	8.99185	8.77283	9.21087	92	3%
4_Park Plaza Westminster Bridge London		8.72229	8.71863	8.57643	8.86083	314	8%
5_Strand Palace Hotel		8.54076	8.53298	8.34422	8.72174	184	4%

TOTAL	total: 19317	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.80807	6.80831	6.74980	6.86681	4511	100%
2_Copthorne Tara Hotel London Kensington		8.09504	8.09470	8.04253	8.14687	3466	100%
3_DoubleTree by Hilton Hotel London Tower of London		8.65448	8.65312	8.60322	8.70303	3146	100%
4_Park Plaza Westminster Bridge London		8.65627	8.65407	8.60671	8.70142	4020	100%
5_Strand Palace Hotel		8.13011	8.13053	8.08350	8.17755	4174	100%

Table 13. Mean, bootstrapped mean, low, and high values based on the types of reviewers (1)

Spring	total: 5129	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.83729	6.83465	6.71864	6.95067	1212	27%
2_Copthorne Tara Hotel London Kensington		8.10568	8.10312	7.99848	8.20776	863	25%
3_DoubleTree by Hilton Hotel London Tower of London		8.74175	8.73756	8.64486	8.83025	824	26%
4_Park Plaza Westminster Bridge London		8.70558	8.70126	8.61059	8.79192	1076	27%
5_Strand Palace Hotel		8.12695	8.12682	8.04082	8.21282	1154	28%

Summer	total: 4797	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.89210	6.89144	6.77712	7.00577	1127	25%
2_Copthorne Tara Hotel London Kensington		7.99136	7.98978	7.88269	8.09687	799	23%
3_DoubleTree by Hilton Hotel London Tower of London		8.66574	8.66154	8.56593	8.75714	861	27%
4_Park Plaza Westminster Bridge London		8.57630	8.57394	8.47845	8.66943	1021	25%
5_Strand Palace Hotel		8.01496	8.01340	7.91395	8.11284	989	24%

Autumn	total: 3025	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.53498	6.53490	6.39048	6.67933	729	16%
2_Copthorne Tara Hotel London Kensington		8.04690	8.04403	7.91933	8.16874	646	19%
3_DoubleTree by Hilton Hotel London Tower of London		8.58283	8.57561	8.44281	8.70840	466	15%
4_Park Plaza Westminster Bridge London		8.63039	8.62972	8.51726	8.74217	645	16%
5_Strand Palace Hotel		7.99907	7.99425	7.85584	8.13265	539	13%

Winter	total: 6366	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.85586	6.85365	6.74969	6.95761	1443	32%
2_Copthorne Tara Hotel London Kensington		8.18549	8.18442	8.10081	8.26803	1158	33%
3_DoubleTree by Hilton Hotel London Tower of London		8.60603	8.60313	8.50935	8.69691	995	32%
4_Park Plaza Westminster Bridge London		8.69171	8.68892	8.60982	8.76801	1278	32%
5_Strand Palace Hotel		8.25623	8.25339	8.17669	8.33009	1492	36%

TOTAL	total: 19317	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.80807	6.80831	6.74980	6.86681	4511	100%
2_Copthorne Tara Hotel London Kensington		8.09504	8.09470	8.04253	8.14687	3466	100%
3_DoubleTree by Hilton Hotel London Tower of London		8.65448	8.65312	8.60322	8.70303	3146	100%
4_Park Plaza Westminster Bridge London		8.65627	8.65407	8.60671	8.70142	4020	100%
5_Strand Palace Hotel		8.13011	8.13053	8.08350	8.17755	4174	100%

Table 14. Mean, bootstrapped mean, low, and high values based on the season

1 Night	total: 9982	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.86834	6.86790	6.79284	6.94296	2874	64%
2_Copthorne Tara Hotel London Kensington		8.15488	8.15465	8.07816	8.23115	1609	46%
3_DoubleTree by Hilton Hotel London Tower of London		8.62447	8.62486	8.55420	8.69553	1655	53%
4_Park Plaza Westminster Bridge London		8.79152	8.78899	8.72313	8.85485	1781	44%
5_Strand Palace Hotel		8.19549	8.19502	8.12636	8.26369	2063	49%

2-3 Nights	total: 7483	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.78134	6.78176	6.68177	6.88174	1383	31%
2_Copthorne Tara Hotel London Kensington		8.06298	8.06325	7.98578	8.14071	1456	42%
3_DoubleTree by Hilton Hotel London Tower of London		8.71257	8.70972	8.63716	8.78228	1360	43%
4_Park Plaza Westminster Bridge London		8.63236	8.62998	8.55768	8.70227	1607	40%
5_Strand Palace Hotel		8.08199	8.08197	8.00836	8.15559	1677	40%

Table 15. Mean, bootstrapped mean, low, and high values based on the length of stay

>3 Nigths	total: 1852	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.27165	6.27213	6.02559	6.51866	254	6%
2_Copthorne Tara Hotel London Kensington		7.97132	7.97045	7.82319	8.11771	401	12%
3_DoubleTree by Hilton Hotel London Tower of London		8.43053	8.40687	8.14198	8.67176	131	4%
4_Park Plaza Westminster Bridge London		8.33592	8.33009	8.20354	8.45665	632	16%
5_Strand Palace Hotel		8.0053	8.00232	7.85533	8.14931	434	10%

TOTAL	total: 19317	Reviewer Score					
Hotel_names:		mean	boot_mean	boot_low	boot_high	reviews	%
1_Britannia International Hotel Canary Wharf		6.80807	6.80831	6.74980	6.86681	4511	100%
2_Copthorne Tara Hotel London Kensington		8.09504	8.09470	8.04253	8.14687	3466	100%
3_DoubleTree by Hilton Hotel London Tower of London		8.65448	8.65312	8.60322	8.70303	3146	100%
4_Park Plaza Westminster Bridge London		8.65627	8.65407	8.60671	8.70142	4020	100%
5_Strand Palace Hotel		8.13011	8.13053	8.08350	8.17755	4174	100%

Table 15. Mean, bootstrapped mean, low, and high values based on the length of stay

Appendix 1 is a detailed code by which the data were manipulated to present them in the form shown in the tables above. The processing of the data consisted in calculating the mean value for each hotel and subsequently calculating the mean value based on the bootstrapping procedure. The purpose of this manipulation was to give the data statistical reliability.

7 Data analysis

Based on the data obtained during the research and presented in Chapter 6, it is possible to make an analysis relying on the bootstrapped calculation of the 95% limits of the mean value. The Simpson paradox detection technique is based on the condition that if the 95% range of the total mean does not overlap with the majority of subgroups with the 95% range of the group mean, then it indicates the existence of the Simpson paradox. In other words, in the case of analysing the type of reviewer variable, there should be three of the five 95% ranges of subgroups that will not overlap with the total 95% range, then it can be stated that Simpson's paradox exists in the data under study.

The following tables are developed for each of the variables monitored, which consist of an analysis of each hotel's sub-variables relative to their overall values. The green color indicates data ranges that overlap with the total value of 95% range of the mean of the variables under study, while red indicates ranges that do not overlap. If there are more red values than green values in any hotel, it will be treated as revealing Simpson's paradox.

1_Britannia International Hotel Canary Wharf	boot_low	boot_high
Total	6.74980	6.86681
Group of friends	6.72019	7.08409
Solo traveler	6.38282	6.57605
Family with young children	6.56445	7.08413
Couple	6.95597	7.12653
Family with older children	6.50854	7.31585

2_Copthorne Tara Hotel London Kensington	boot_low	boot_high
Total	8.04253	8.14687
Group of friends	8.07547	8.33544
Solo traveler	7.61698	7.86730
Family with young children	7.92284	8.27266
Couple	8.11789	8.25526
Family with older children	7.78007	8.26647

3_DoubleTree by Hilton Hotel London Tower of London	boot_low	boot_high
Total	8.60322	8.70303
Group of friends	8.62654	8.90077
Solo traveler	8.30496	8.61007
Family with young children	8.37336	8.75302
Couple	8.59585	8.71942
Family with older children	8.77283	9.21087

4_Park Plaza Westminster Bridge London	boot_low	boot_high
Total	8.60671	8.70142
Group of friends	8.65887	8.90222
Solo traveler	8.29963	8.57210
Family with young children	8.60052	8.76588
Couple	8.57324	8.72826
Family with older children	8.57643	8.86083

5_Strand Palace Hotel	boot_low	boot_high
Total	8.08350	8.17755
Group of friends	8.17272	8.44922
Solo traveler	7.77193	7.9541
Family with young children	8.03829	8.39757
Couple	8.12596	8.25832
Family with older children	8.34422	8.72174

Table 16. Simpson’s paradox detection based on the type of reviewer variable

A detailed examination of the data presented in Table 16 leads to the conclusion that the data appear to be naturally distributed in relation to the variables and that there is no appearance of the paradox. However, based on this data it is fair to conclude that certain types of travelers influence the hotel's perception of these travelers and, as a consequence, the final rating of the hotel. Therefore, it is noticeable in Table 16 that all five examined hotels showed a tendency that people traveling with a group of friends to rate the hotels higher than their average rating. And as well as people traveling solo rate it below the average rating. Most probably the social nature of human nature plays a role here. It means that the same person in the same place in a group of friends will be more pleasant and convenient than alone. Solo tourists rate hotels lower, and groups of tourists rate hotels higher. It is necessary to pay more attention to solo tourists.

1_Britannia International Hotel Canary Wharf	boot_low	boot_high
Total	6.74980	6.86681
Spring	6.71864	6.95067
Summer	6.77712	7.00577
Autumn	6.39048	6.67933
Winter	6.74969	6.95761
2_Copthorne Tara Hotel London Kensington	boot_low	boot_high
Total	8.04253	8.14687
Spring	7.99848	8.20776
Summer	7.88269	8.09687
Autumn	7.91933	8.16874
Winter	8.10081	8.26803
3_DoubleTree by Hilton Hotel London Tower of London	boot_low	boot_high
Total	8.60322	8.70303
Spring	8.64486	8.83025
Summer	8.56593	8.75714
Autumn	8.44281	8.70840
Winter	8.50935	8.69691
4_Park Plaza Westminster Bridge London	boot_low	boot_high
Total	8.60671	8.70142
Spring	8.61059	8.79192
Summer	8.47845	8.66943
Autumn	8.51726	8.74217
Winter	8.60982	8.76801
5_Strand Palace Hotel	boot_low	boot_high
Total	8.08350	8.17755
Spring	8.04082	8.21282
Summer	7.91395	8.11284
Autumn	7.85584	8.13265
Winter	8.17669	8.33009

Table 17. Simpson’s paradox detection based on the season

A detailed examination of the data presented in Table 17 leads to the conclusion that the data appear to be naturally distributed in relation to the variables and that there is no appearance of the paradox. However, it is worth noting the prominent features of the findings. For example, four hotels showed a tendency that people staying in the winter time of the year tend to rate hotels significantly higher than the annual and other season ratings. This is probably due to the existence of the Christmas holidays in the winter time period. The presence of such a strong external mood stimulant may influence people's perception of hotels in general. At the same time, the fall season has the lowest hotel ratings among the

other seasons. This is most likely due to the depressing time of year and the associated with it colder weather, apathy, less daylight, and other external factors that manifest themselves in the autumn season.

1_Britannia International Hotel Canary Wharf	boot_low	boot_high
Total	6.74980	6.86681
1 night	6.79284	6.94296
2 nights	6.68177	6.88174
3+ nights	6.02559	6.51866
2_Copthorne Tara Hotel London Kensington	boot_low	boot_high
Total	8.04253	8.14687
1 night	8.07816	8.23115
2 nights	7.98578	8.14071
3+ nights	7.82319	8.11771
3_DoubleTree by Hilton Hotel London Tower of London	boot_low	boot_high
Total	8.60322	8.70303
1 night	8.55420	8.69553
2 nights	8.63716	8.78228
3+ nights	8.14198	8.67176
4_Park Plaza Westminster Bridge London	boot_low	boot_high
Total	8.60671	8.70142
1 night	8.72313	8.85485
2 nights	8.55768	8.70227
3+ nights	8.20354	8.45665
5_Strand Palace Hotel	boot_low	boot_high
Total	8.08350	8.17755
1 night	8.12636	8.26369
2 nights	8.00836	8.15559
3+ nights	7.85533	8.14931

Table 18. Simpson’s paradox detection based on a length of stay

A detailed examination of the data presented in Table 18 allows concluding that Simpson's paradox was found among the data for the hotel N4 - Park Plaza Westminster Bridge London. The paradox involves the fact that when considering the data on the variable length of stay, two of the three subgroups of the 95% range of mean values do not overlap with the total range of the variables. Thus, it makes sense to consider this phenomenon in more detail and, perhaps, to think about compiling the recommendation systems of travel platforms taking into account these findings. However, it is worth mentioning the distinctive features of the given data. For example, there is an obvious pattern associated with the length of stay in the hotel and its evaluation. Based on the data, it can be assumed that there is an inversely proportional relationship between the number of days of stay in the hotel and its rating. In other words, the longer the guest stays in the hotel, the worse the perception of the hotel becomes, and as a consequence, it negatively affects the rating of the hotel. Table 18 shows that the ratings of tourists who stayed only one night are naturally higher than those who stayed three+ nights. This is due to the fact that a large amount of time allows for recognizing a greater number of disadvantages and defects of a particular hotel.

8 Conclusion

After the above-described research, it was possible to obtain comprehensive results in relation to the research question posed at the beginning of the work. The essence of the work was to investigate what effect Simpson's paradox can have and whether it is worth considering as a relevant concept in the work of travel platforms such as Booking.com, Trip Advisor, etc. In the process of research, there was revealed at least one case, from which it is possible to conclude that the presence of Simpson's paradox in the Booking.com data sample was detected and possible in general. Consequently, the issue of expediency in taking this phenomenon into account is quite relevant. It should be noted that there is a need for more detailed studies aimed primarily at determining the potential effects when taking into account Simpson's paradox in the recommendation systems of such tourist platforms as Booking.com, TripAdvisor, etc. That is, how much it could be improved recommendation systems and how these changes can be measured. And the study also revealed a number of interesting patterns that may be useful for both travel platforms, their end users, and for hotel owners in the case of Booking.com service.

The first findings of the research were trends related to the type of travelers who stay in hotels. This pattern can be summarized as follows: "People who traveled alone rated hotels lower than average when a group of friends staying in a hotel generally rated it higher than average. This is most likely directly related to the social characteristics of the people. The number of people staying in a hotel in this case is an external factor that affects the perception and evaluation of the hotel by the guests. This finding is useful to consider for the optimization of the recommendation systems that take into account the type of traveler, for many participants in the interaction with travel platforms. For example, for hotels, this can mean that it is probably worth thinking about the service features of the guests who are staying alone. Perhaps a more personalized service, more attention, and more social areas such as lounges and leisure rooms for relaxing and networking. From the point of view of the Booking.com platform itself, in terms of improving the recommendation system, there is a question about which group of people are the most objective. Maybe people who travel alone are more thorough in their evaluation of hotels and their reviews are more objective because there are no external factors that influence the evaluation, such as small children, friends, partners etc.. There is definitely a need for further more detailed research related to this topic.

A detailed examination and analysis of the second potential variable related to the potential presence of the Simpson paradox lead to the conclusion that the paradox was not detected. However, a certain trend was found that people rate hotels in the fall time of year much

lower than the yearly mark, when in the winter and spring time of year the ratings of most hotels show much higher results. This is most likely due to the depressing time of year and the associated with it colder weather, apathy, less daylight, and other external factors that manifest themselves in the autumn season. It is worth noting that November is the peak month with the lowest ratings. Based on this, it is worthwhile for hoteliers to start looking at the situation from a mental health standpoint and improve their own services from the same standpoint. Possible solutions could be the creation of the previously described lounges to relax and meet new people, improving the diet with the addition of more fruits and vitamins.

The third finding of the study was a pattern associated with the length of stay of guests who stay in hotels. This finding was assumed at the beginning of the study and was confirmed in the process. It can be summarized as follows: "The same people staying in the same hotel for one night and for three nights will evaluate it differently". And this hypothesis was confirmed as follows: people who stay one night, evaluate the hotel, as a rule, higher than people who stay in the same hotel for three nights. It is trivially related to the fact that after three nights a person gets to know the hotel in more detail and probably discovers some shortcomings unnoticed by people who stay only one night. A possible solution in this situation would be to offer certain privileges to guests who stay for two or three nights or more. For example, for people staying for more than one night to provide a discount at the hotel restaurant bar. The essence is to level out the nuance associated with the natural tendency of people over time to pay attention to detail, a possible economic benefit expressed in the provision of discounts for additional services.

However, it should be noted that at this stage all the above-mentioned suggestions are only assumptions expressed in the form of recommendations based on the results of the study. Definitely more detailed and in-depth research on each of these variables and obtaining a more complete and objective picture of what is happening is needed.

9 Summary

In conclusion, it can be summarized that this paper was dedicated to examining the stated thesis topic in order to discover the answer to the research question posed at the beginning: should travel platforms analyze their data for the Simpson paradox, and if so, how can it be used? After the above research, the results were obtained in a very ambiguous way.

The chronology of the research on this question was based on the initial definition and exploration of all theoretical aspects related to this topic. The second chapter provided clear definitions of the "Simpson's Paradox" phenomenon and its most famous examples in various fields such as health care, marketing, and the social sciences. The theoretical knowledge base detailed specific cases of this paradox and explained the nature of its occurrence. It was necessary to establish a theoretical framework for conducting research related to examining already existing real-world data on the example of the Booking.com platform for the presence of the paradox.

The following chapter was about conducting the quantitative analysis that used secondary data from Booking.com, as a prominent representative of this field. Using the Python programming language, it was calculated the average of reviews of guests staying in hotels based on three variables: type of reviewer, season, and length of stay. Thus, 5 major hotels with the biggest number of reviews were considered in the data analysis. To give statistical validity, a bootstrapping procedure was performed, based on creating new samples with replacement objects and calculating their mean values. Approaching this culminating point, the absence of the Simpson paradox in the studied data was revealed. However, the researchers detected a number of interesting trends related to the influence of various factors on the overall hotel rating. Such factors were the type of reviewer, the season of the year, and the length of stay of the tourists. All of these factors had certain patterns, and it seems necessary to conduct a more detailed study aimed at their further study.

Thus, it may seem quite difficult to characterize the specific conclusions of this thesis study due to some ambiguous outcomes. Because, on the one hand, it was found that there is Sipson's paradox in the studied data. And therefore it is appropriate to consider this phenomenon. On the other hand, there were found very interesting data trends associated with the pattern of reviewer feedback. For example, it was found that people staying for 1 night in a hotel usually rate it higher than if they stayed for 3 nights. Such insights are described in detail in the previous part of the study.

But given the fact that the consideration of Simpson's paradox was intended to investigate interesting data patterns, which can later be used to optimize the performance of recommendation systems, it can be stated that the main purpose of writing this work has been achieved. The results of the work are very useful, despite the fact that they were obtained in a different way than planned in the description of the study. Due to the fact that in carrying out this study have been complied with all the necessary requirements and obtained interesting results can be characterized this work as successfully executed.

The disadvantages or parts that could be improved include the unrealized attempt of parsing data directly from Booking.com, as well as taking into account the analysis of a larger number of hotels under study. Also, a significant addition to this work could be to write a program that automatically analyzes and detects Simpson's paradox in the data. This algorithm already exists and is freely available.

Suggestions for future research might include examining these variables described earlier in the study in a more detailed way. In order to optimize the recommendation systems, these variables are hotspots that can be investigated to produce more detailed and clear patterns with regard to hotel ratings. Further research in these areas can significantly improve the rating systems of travel platforms and, as a consequence, people's experience with them.

References

Printed sources

Pearl, J. 2000. Causality: Models, Reasoning, and Inference. New York: Cambridge University Press.

Electronic sources

Alipourfard, N., Fennell, P.G., & Lerman, K. 2018. Can you Trust the Trend? Discovering Simpson's Paradoxes in Social Data. ACM Digital Library. Retrieved on 14 April 2022. Available at: https://doi.org/10.1145/3159652.3159684

Bickel, P.J., Hammel, E.A., & O'Connell, J.W. 1975. Sex Bias in Graduate Admissions: Data from Berkeley. Science. Retrieved on 14 April 2022. Available at: https://doi.org/10.1126/SCIENCE.187.4175.398

Charig, C.R., Webb D.R., Payne S.R., & Wickham J.E. 1986. Comparison of treatment of renal calculi by open surgery, percutaneous nephrolithotomy, and extracorporeal shock-wave lithotripsy. The BMJ. Retrieved on 14 April 2022. Available at: https://doi.org/10.1136/BMJ.292.6524.879

Frost, J. 2022. Introduction to Bootstrapping in Statistics. Statistics by Jim. Retrieved on 14 April 2022. Available at: https://statisticsbyjim.com/hypothesis-testing/bootstrapping/

Grigg, T. 2018. Simpson's Paradox and Interpreting Data. Towards Data Science. Retrieved on 14 April 2022. Available at: https://towardsdatascience.com/simpsons-paradox-and-interpreting-data-6a0443516765

Henderson, C. 2020. Proactive and Reactive Business Strategies - Which is Better?. Any Connector. Retrieved on 14 April 2022. Available at: https://anyconnector.com/en/what-is-business-intelligence/proactive-and-reactive.html

Holt, G.B. 2016. Potential Simpson's paradox in multicenter study of intraperitoneal chemotherapy for ovarian cancer. Journal of Clinical Oncology. Retrieved on 14 April 2022. Available at: https://doi.org/10.1200/JCO.2015.64.4542

Oheix, J., Liu, J., & Sakib, A.S. 2022. 515K Hotel Reviews Data in Europe. Kaggle. Retrieved on 14 April 2022. Available at: https://www.kaggle.com/datasets/jiashenliu/515k-hotel-reviews-data-in-europe

Koswara, I., Aaniya, A. & Han, P.G. 2022. Simpson's Paradox. Brilliant Math & Science Wiki, Brilliant. Retrieved on 14 April 2022. Available at: https://brilliant.org/wiki/simpsons-paradox/

Lipovetsky, S., & Conklin, W.M. 2006. Data aggregation and Simpson's paradox gauged by index numbers. European Journal of Operational Research. Retrieved on 14 April 2022. Available at: https://doi.org/10.1016/J.EJOR.2004.10.005

Norton, H.J., & Divine, G. 2015. Simpson's paradox … and how to avoid it. Significance. Retrieved on 14 April 2022. Available at: https://doi.org/10.1111/J.1740-9713.2015.00844.X

Reintjes, R., De Boer, A., Van Pelt, W., & Mintjes-de Groot, J. 2020. Simpson's Paradox: An Example from Hospital Epidemiology. Epidemiology. Retrieved on 14 April 2022. Available at: https://journals.lww.com/epidem/fulltext/2000/01000/simpson_s_paradox__an_example_from_hospital.17.aspx

Rivers, K. 2018. Considering Simpson's Paradox in Mobile Advertising. Medium. Retrieved on 14 April 2022. Available at: https://medium.com/adinmo/considering-simpsons-paradox-in-mobile-advertising-2cbc552d39f2

Wagner, C.H. 1982. Simpson's paradox in real life. American Statistician. Retrieved on 14 April 2022. Available at: https://doi.org/10.1080/00031305.1982.10482778

Appendix 1. Code for cleaning data and calculation of mean, low, and high limits values using bootstrapping method

In [1]:
```
import pandas as pd
import numpy as np
import json
from datetime import datetime
from scipy.stats import bootstrap
from tabulate import tabulate
```

In [2]:
```
raw_df = pd.read_csv('/Users/dmitriy_debeliy/Desktop/Hotel_Reviews.csv')
```

In [3]:
```
prepared_df = raw_df.drop(columns=['Negative_Review', 'Review_Total_Negative_Word_Counts', 'Total_Numbe
```

In [6]:
```
def get_trip_type(row):
    row = json.loads(row.Tags.replace('\'', '"'))
    if len(row) > 0:
        if row[0].find('trip') != -1:
            return row[0].strip()
        else:
            pd.NA

def get_traveller_type(row):
    types = ['Group', 'Solo traveler', 'Family with young children', 'Couple', 'Family with older child
    row = json.loads(row.Tags.replace('\'', '"'))
    if len(row) > 0 and row[0].strip() in types:
        return row[0].strip()
    elif len(row) > 1 and row[1].strip() in types:
        return row[1].strip()
    else: return pd.NA

def get_room_type(row):
    types = ['room', 'apartment', 'house', 'studio', 'suite']
    row = json.loads(row.Tags.replace('\'', '"'))
    for i in row:
        if any(x in i.lower() for x in types):
            return i
    return pd.NA

def get_nights(row):
    row = json.loads(row.Tags.replace('\'', '"'))
    for i in row:
        if i.find('night') != -1 and len(i.split()) == 3:
            return int(i.strip().split(' ')[1])
    return pd.NA

def get_seasone(row):
    month = datetime.strptime(row.Review_Date, '%m/%d/%Y').month
    if month > 2 and month <6:
        seasone = 'spring'
    elif month > 5 and month < 9:
        seasone = 'summer'
    elif month > 8 and month < 11:
        seasone = 'autumn'
    else: seasone = 'winter'
    return seasone
```

In [7]:
```
row = json.loads(prepared_df.values[0][7].replace('\'', '"'))
for i in row:
    if i.find('nights') != -1 and len(i.split()) == 3:
        print(i)
```

```
 Stayed 6 nights
```

In [8]:
```
prepared_df['Trip_type'] = prepared_df.apply(get_trip_type , axis=1)
prepared_df['Traveller_type'] = prepared_df.apply(get_traveller_type , axis=1)
prepared_df['Room_type'] = prepared_df.apply(get_room_type , axis=1)
prepared_df['Nights'] = prepared_df.apply(get_nights , axis=1)
prepared_df['Seasone'] = prepared_df.apply(get_seasone, axis=1)
```

In [9]:
```
prepared_df.groupby('Hotel_Name').count().sort_values('Average_Score',ascending=False).head(5)
```

Out[9]:

Hotel_Name	Hotel_Address	Additional_Number_of_Scoring	Review_Date	Average_Score	Reviewer_Nationality	Reviewer_Sc
Britannia International Hotel Canary Wharf	4789	4789	4789	4789	4789	47
Strand Palace Hotel	4256	4256	4256	4256	4256	42
Park Plaza Westminster Bridge London	4169	4169	4169	4169	4169	41
Copthorne Tara Hotel London Kensington	3578	3578	3578	3578	3578	35
DoubleTree by Hilton Hotel London Tower of London	3212	3212	3212	3212	3212	32

In [10]:
```
top_5_hotels = ['Britannia International Hotel Canary Wharf', 'Strand Palace Hotel',
                'Park Plaza Westminster Bridge London', 'Copthorne Tara Hotel London Kensington',
                'DoubleTree by Hilton Hotel London Tower of London']
final_df = prepared_df[prepared_df['Hotel_Name'].isin(top_5_hotels)]
final_df
```

Out[10]:

	Hotel_Address	Additional_Number_of_Scoring	Review_Date	Average_Score	Hotel_Name	Reviewer_Nationality	Revi
63942	163 Marsh Wall Docklands Tower Hamlets London ...	2682	8/3/2017	7.1	Britannia International Hotel Canary Wharf	United Kingdom	
63943	163 Marsh Wall Docklands Tower Hamlets London ...	2682	8/3/2017	7.1	Britannia International Hotel Canary Wharf	United Kingdom	
63944	163 Marsh Wall Docklands Tower Hamlets London ...	2682	8/2/2017	7.1	Britannia International Hotel Canary Wharf	United Kingdom	
63945	163 Marsh Wall Docklands Tower Hamlets London ...	2682	8/2/2017	7.1	Britannia International Hotel Canary Wharf	United Kingdom	
63946	163 Marsh Wall Docklands Tower Hamlets London ...	2682	8/2/2017	7.1	Britannia International Hotel Canary Wharf	United Kingdom	
...	...	...	...	...	...	...	...
508191	Westminster Bridge Road Lambeth London SE1 7UT...	2623	8/6/2015	8.7	Park Plaza Westminster Bridge London	United States of America	

In [11]:
```
types = ['Group', 'Solo traveler', 'Family with young children', 'Couple', 'Family with older children'
```

In [12]:
```
final_df = final_df.dropna()
```

GROUP BY TRAVELLER TYPE

In [36]:
```
# Bootstrap function
def boot_mean(data):
    res = bootstrap((data,), np.mean)
    low, high = res.confidence_interval

    return np.mean([low, high]), low, high
```

In [40]:
```
view = final_df[final_df['Traveller_type'] == type] \
                        .groupby('Hotel_Name') \
                        .agg({'Reviewer_Score': ['mean', boot_mean, 'count']})

view.columns = [tup[1] if tup[1] else tup[0] for tup in view.columns]

view['boot_low'] = view['boot_mean'].apply(lambda x: x[1])
view['boot_high'] = view['boot_mean'].apply(lambda x: x[2])
view['boot_mean'] = view['boot_mean'].apply(lambda x: x[0])

view = view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]
view
```

Out[40]:

Hotel_Name	mean	boot_mean	boot_low	boot_high	count
Britannia International Hotel Canary Wharf	6.928049	6.917683	6.515854	7.319512	82
Copthorne Tara Hotel London Kensington	8.035294	8.021525	7.775294	8.267755	170
DoubleTree by Hilton Hotel London Tower of London	9.014130	8.993031	8.774105	9.211957	92
Park Plaza Westminster Bridge London	8.722293	8.714803	8.571008	8.858599	314
Strand Palace Hotel	8.540761	8.529348	8.344022	8.714674	184

In [43]:
```
for type in types:
    view = final_df[final_df['Traveller_type'] == type] \
                        .groupby('Hotel_Name') \
                        .agg({'Reviewer_Score': ['mean', boot_mean, 'count']})

    view.columns = [tup[1] if tup[1] else tup[0] for tup in view.columns]

    view['boot_low'] = view['boot_mean'].apply(lambda x: x[1])
    view['boot_high'] = view['boot_mean'].apply(lambda x: x[2])
    view['boot_mean'] = view['boot_mean'].apply(lambda x: x[0])

    view = view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]

    print(type)
    print('total: ', len(final_df[final_df['Traveller_type'] == type]))
    print(tabulate(view, headers=['Hotel_Name', 'mean', 'boot_mean', 'boot_low', 'boot_high', 'count'],
    print()


all_view = final_df.groupby('Hotel_Name').agg({'Reviewer_Score': ['mean', boot_mean, 'count']})
all_view.columns = [tup[1] if tup[1] else tup[0] for tup in all_view.columns]

all_view['boot_low'] = all_view['boot_mean'].apply(lambda x: x[1])
all_view['boot_high'] = all_view['boot_mean'].apply(lambda x: x[2])
all_view['boot_mean'] = all_view['boot_mean'].apply(lambda x: x[0])

all_view = all_view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]

print('total: ', len(final_df))
print(tabulate(all_view, headers=['Hotel_Name', 'mean', 'boot_mean', 'boot_low', 'boot_high', 'count'],
```

```
Group
total:  2340
+-----------------------------------------------+---------+------------+------------+-------------
+---------+
| Hotel_Name                                    |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-----------------------------------------------+---------+------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf    | 6.9063  |     6.90214 |    6.72019 |     7.08409
|     460 |
| Copthorne Tara Hotel London Kensington        | 8.20801 |     8.20546 |    8.07547 |     8.33544
|     549 |
| DoubleTree by Hilton Hotel London Tower of London | 8.77216 |     8.76366 |    8.62654 |     8.90077
|     388 |
| Park Plaza Westminster Bridge London          | 8.7879  |     8.78055 |    8.65887 |     8.90222
|     496 |
| Strand Palace Hotel                           | 8.31767 |     8.31097 |    8.17272 |     8.44922
|     447 |
+-----------------------------------------------+---------+------------+------------+-------------
```

Solo traveler
total: 4352

Hotel_Name	mean	boot_mean	boot_low	boot_high	count
Britannia International Hotel Canary Wharf	6.47841	6.47943	6.38282	6.57605	1649
Copthorne Tara Hotel London Kensington	7.74371	7.74214	7.61698	7.8673	636
DoubleTree by Hilton Hotel London Tower of London	8.46356	8.45752	8.30496	8.61007	343
Park Plaza Westminster Bridge London	8.44101	8.43586	8.29963	8.5721	534
Strand Palace Hotel	7.8663	7.86302	7.77193	7.9541	1190

Family with young children
total: 2212

Hotel_Name	mean	boot_mean	boot_low	boot_high	count
Britannia International Hotel Canary Wharf	6.825	6.82429	6.56445	7.08413	252
Copthorne Tara Hotel London Kensington	8.10242	8.09775	7.92284	8.27266	289
DoubleTree by Hilton Hotel London Tower of London	8.57331	8.56319	8.37336	8.75302	251
Park Plaza Westminster Bridge London	8.68714	8.6832	8.60052	8.76588	1151
Strand Palace Hotel	8.22268	8.21793	8.03829	8.39757	269

Couple
total: 9571

Hotel_Name	mean	boot_mean	boot_low	boot_high	count
Britannia International Hotel Canary Wharf	7.04226	7.04125	6.95597	7.12653	2068
Copthorne Tara Hotel London Kensington	8.18804	8.18658	8.11789	8.25526	1822
DoubleTree by Hilton Hotel London Tower of London	8.65792	8.65763	8.59585	8.71942	2072
Park Plaza Westminster Bridge London	8.65193	8.65075	8.57324	8.72826	1525
Strand Palace Hotel	8.19232	8.19214	8.12596	8.25832	2084

Family with older children
total: 842

Hotel_Name	mean	boot_mean	boot_low	boot_high	count
Britannia International Hotel Canary Wharf	6.92805	6.9122	6.50854	7.31585	82
Copthorne Tara Hotel London Kensington	8.03529	8.02327	7.78007	8.26647	170
DoubleTree by Hilton Hotel London Tower of London	9.01413	8.99185	8.77283	9.21087	92
Park Plaza Westminster Bridge London	8.72229	8.71863	8.57643	8.86083	314
Strand Palace Hotel	8.54076	8.53298	8.34422	8.72174	184

```
total:  19317
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.80807 |     6.80831 |    6.7498  |     6.86681
|    4511 |
| Copthorne Tara Hotel London Kensington          | 8.09504 |     8.0947  |    8.04253 |     8.14687
|    3466 |
| DoubleTree by Hilton Hotel London Tower of London | 8.65448 |     8.65312 |    8.60322 |     8.70303
|    3146 |
| Park Plaza Westminster Bridge London            | 8.65627 |     8.65407 |    8.60671 |     8.70142
|    4020 |
| Strand Palace Hotel                             | 8.13011 |     8.13053 |    8.0835  |     8.17755
|    4174 |
+-------------------------------------------------+---------+-------------+------------+-------------
```

GROUP BY SEASONES

In [45]:
```python
seasons = ['spring', 'summer', 'autumn', 'winter']

for season in seasons:
    view = final_df[final_df['Seasone'] == season] \
                        .groupby('Hotel_Name') \
                        .agg({'Reviewer_Score': ['mean', boot_mean, 'count']})

    view.columns = [tup[1] if tup[1] else tup[0] for tup in view.columns]

    view['boot_low'] = view['boot_mean'].apply(lambda x: x[1])
    view['boot_high'] = view['boot_mean'].apply(lambda x: x[2])
    view['boot_mean'] = view['boot_mean'].apply(lambda x: x[0])

    view = view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]

    print(season)
    print('total: ', len(final_df[final_df['Seasone'] == season]))
    print(tabulate(view, headers=['Hotel_Name', 'mean', 'boot_mean', 'boot_low', 'boot_high', 'count'],
    print()
```

```
spring
total:  5129
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.83729 |     6.83465 |    6.71864 |     6.95067
|    1212 |
| Copthorne Tara Hotel London Kensington          | 8.10568 |     8.10312 |    7.99848 |     8.20776
|     863 |
| DoubleTree by Hilton Hotel London Tower of London | 8.74175 |     8.73756 |    8.64486 |     8.83025
|     824 |
| Park Plaza Westminster Bridge London            | 8.70558 |     8.70126 |    8.61059 |     8.79192
|    1076 |
| Strand Palace Hotel                             | 8.12695 |     8.12682 |    8.04082 |     8.21282
|    1154 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
```

```
summer
total:  4797
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.8921  |     6.89144 |    6.77712 |     7.00577
|    1127 |
| Copthorne Tara Hotel London Kensington          | 7.99136 |     7.98978 |    7.88269 |     8.09687
|     799 |
| DoubleTree by Hilton Hotel London Tower of London | 8.66574 |     8.66154 |    8.56593 |     8.75714
|     861 |
| Park Plaza Westminster Bridge London            | 8.5763  |     8.57394 |    8.47845 |     8.66943
|    1021 |
| Strand Palace Hotel                             | 8.01496 |     8.0134  |    7.91395 |     8.11284
|     989 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
```

```
autumn
total:  3025
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.53498 |      6.5349 |    6.39048 |     6.67933
|     729 |
| Copthorne Tara Hotel London Kensington          | 8.0469  |      8.04403 |    7.91933 |     8.16874
|     646 |
| DoubleTree by Hilton Hotel London Tower of London | 8.58283 |      8.57561 |    8.44281 |     8.7084
|     466 |
| Park Plaza Westminster Bridge London            | 8.63039 |      8.62972 |    8.51726 |     8.74217
|     645 |
| Strand Palace Hotel                             | 7.99907 |      7.99425 |    7.85584 |     8.13265
|     539 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+

winter
total:  6366
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.85586 |      6.85365 |    6.74969 |     6.95761
|    1443 |
| Copthorne Tara Hotel London Kensington          | 8.18549 |      8.18442 |    8.10081 |     8.26803
|    1158 |
| DoubleTree by Hilton Hotel London Tower of London | 8.60603 |      8.60313 |    8.50935 |     8.69691
|     995 |
| Park Plaza Westminster Bridge London            | 8.69171 |      8.68892 |    8.60982 |     8.76801
|    1278 |
| Strand Palace Hotel                             | 8.25623 |      8.25339 |    8.17669 |     8.33009
|    1492 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
```

LENGHT OF STAY

In [47]:

```
view = final_df[final_df['Nights'].isin([1])] \
                    .groupby('Hotel_Name') \
                    .agg({'Reviewer_Score': ['mean', boot_mean, 'count']})

view.columns = [tup[1] if tup[1] else tup[0] for tup in view.columns]

view['boot_low'] = view['boot_mean'].apply(lambda x: x[1])
view['boot_high'] = view['boot_mean'].apply(lambda x: x[2])
view['boot_mean'] = view['boot_mean'].apply(lambda x: x[0])

view = view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]

print('Nights = 1')
print(tabulate(view, headers=['Hotel_Name', 'mean', 'boot_mean', 'boot_low', 'boot_high', 'count'], tab
print()


view = final_df[final_df['Nights'].isin([2,3])] \
                    .groupby('Hotel_Name') \
                    .agg({'Reviewer_Score': ['mean', boot_mean, 'count']})

view.columns = [tup[1] if tup[1] else tup[0] for tup in view.columns]

view['boot_low'] = view['boot_mean'].apply(lambda x: x[1])
view['boot_high'] = view['boot_mean'].apply(lambda x: x[2])
view['boot_mean'] = view['boot_mean'].apply(lambda x: x[0])

view = view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]

print('Nights = 2 or 3')
print(tabulate(view, headers=['Hotel_Name', 'mean', 'boot_mean', 'boot_low', 'boot_high', 'count'], tab
print()
```

```
view = final_df[final_df['Nights'] > 3] \
                    .groupby('Hotel_Name') \
                    .agg({'Reviewer_Score': ['mean', boot_mean, 'count']})

view.columns = [tup[1] if tup[1] else tup[0] for tup in view.columns]

view['boot_low'] = view['boot_mean'].apply(lambda x: x[1])
view['boot_high'] = view['boot_mean'].apply(lambda x: x[2])
view['boot_mean'] = view['boot_mean'].apply(lambda x: x[0])

view = view[['mean', 'boot_mean', 'boot_low', 'boot_high', 'count']]

print('Nights > 3')
print(tabulate(view, headers=['Hotel_Name', 'mean', 'boot_mean', 'boot_low', 'boot_high', 'count'], tab
print()
```

```
Nights = 1
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.86834 |      6.8679 |    6.79284 |     6.94296
|    2874 |
| Copthorne Tara Hotel London Kensington          | 8.15488 |     8.15465 |    8.07816 |     8.23115
|    1609 |
| DoubleTree by Hilton Hotel London Tower of London | 8.62447 |     8.62486 |    8.5542  |     8.69553
|    1655 |
| Park Plaza Westminster Bridge London            | 8.79152 |     8.78899 |    8.72313 |     8.85485
|    1781 |
| Strand Palace Hotel                             | 8.19549 |     8.19502 |    8.12636 |     8.26369
|    2063 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
```

```
Nights = 2 or 3
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.78134 |     6.78176 |    6.68177 |     6.88174
|    1383 |
| Copthorne Tara Hotel London Kensington          | 8.06298 |     8.06325 |    7.98578 |     8.14071
|    1456 |
| DoubleTree by Hilton Hotel London Tower of London | 8.71257 |     8.70972 |    8.63716 |     8.78228
|    1360 |
| Park Plaza Westminster Bridge London            | 8.63236 |     8.62998 |    8.55768 |     8.70227
|    1607 |
| Strand Palace Hotel                             | 8.08199 |     8.08197 |    8.00836 |     8.15559
|    1677 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
```

```
Nights > 3
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
| Hotel_Name                                      |    mean |   boot_mean |   boot_low |   boot_high
|   count |
|-------------------------------------------------+---------+-------------+------------+-------------
+---------|
| Britannia International Hotel Canary Wharf      | 6.27165 |     6.27213 |    6.02559 |     6.51866
|     254 |
| Copthorne Tara Hotel London Kensington          | 7.97132 |     7.97045 |    7.82319 |     8.11771
|     401 |
| DoubleTree by Hilton Hotel London Tower of London | 8.43053 |     8.40687 |    8.14198 |     8.67176
|     131 |
| Park Plaza Westminster Bridge London            | 8.33592 |     8.33009 |    8.20354 |     8.45665
|     632 |
| Strand Palace Hotel                             | 8.0053  |     8.00232 |    7.85533 |     8.14931
|     434 |
+-------------------------------------------------+---------+-------------+------------+-------------
+---------+
```