

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій

Рівень вищої освіти – перший (бакалаврський)
Галузь знань – 12 Інформаційні технології
Спеціальність 124 – «Системний аналіз»
Освітньо-професійна програма – Системний аналіз

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ

 д.т.н., проф. Віталій МОКІН

Бакалаврська дипломна робота на тему:

**«СИСТЕМНИЙ АНАЛІЗ ПОВЕДІНКИ КЛІЄНТІВ ПРИ ОНЛАЙН
БРОНЮВАННІ ГОТЕЛІВ»**

НА БАКАЛАВРСЬКІЙ ДИПЛОМНІЙ РОБОТІ
Григорчук Марини Василівни

1. Тема роботи: «Системний аналіз поведінки клієнтів при онлайн бронюванні готелів»

керівник роботи: Козачко О.М., к.т.н., доц. каф. САІТ, доц. спеціальності 124 - «Системний аналіз»

2. Термін виконання студентом роботи

3. Вихідні дані до роботи:

1) Набір даних бронювання готелів.

4. Зміст текстової частини:

1) Загальна характеристика об'єкту дослідження.

2) Вибір оптимального рішення та наведення аргументації.

задачі.

3) Результати передбачення бронювання клієнтами.

5. Перелік ілюстративного матеріалу:

1) Таблиця із даними з наданої бази даних.

2) Кореляційна матриця даних;

3) Графік важливості ознак;

4) Графік важливості ознак за допомогою методу машинного навчання для

класифікації клієнтів при онлайн бронюванні готелів;

5) Графік важливості ознак за допомогою методу машинного навчання для

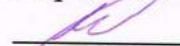
класифікації клієнтів при онлайн бронюванні готелів.

«04» 06 2024 р.

Виконала: студентка 4 курсу, групи СА-206
спеціальності 124 - «Системний аналіз»

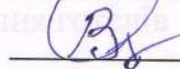
 Марина ГРИГОРЧУК

Керівник: к.т.н., доц. каф. САІТ, доц.

 Олексій КОЗАЧКО

«03» 06 2024 р.

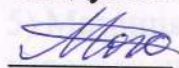
Рецензент: к.т.н., доц. каф. КН

 В. Озерний

«11» 06 2024 р.

Допущено до захисту

Завідувач кафедри САІТ

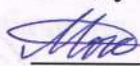
 д.т.н., проф. Віталій МОКІН

«04» 06 2024 р.

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій
Рівень вищої освіти – перший (бакалаврський)
Галузь знань – 12 Інформаційні технології
Спеціальність 124 – «Системний аналіз»
Освітньо-професійна програма – Системний аналіз

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ

 д.т.н., проф. Віталій МОКІН

“03” 03 2024 року

ЗАВДАННЯ
НА БАКАЛАВРСЬКУ ДИПЛОМНУ РОБОТУ СТУДЕНТЦІ
Григорчук Марині Василівні

1. Тема роботи: «Системний аналіз поведінки клієнтів при онлайн бронюванні готелів»

керівник роботи: Козачко О.М., к.т.н., доц. каф. САІТ

затверджені наказом ВНТУ від «11» 03 2024 року № 80

2. Термін подання студентом роботи 31.05.

3. Вихідні дані до роботи:

1) Набір даних бронювання готелів.

4. Зміст текстової частини:

1) Загальна характеристика об'єкту досліджень.

2) Вибір оптимального рішення та налаштувань для розв'язання поставленої задачі.

3) Результати передбачення бронювання готелів клієнтами.

5. Перелік ілюстративного матеріалу:

1) Таблиця із даними з наданої бази даних;

2) Кореляційна матриця даних;

3) Графік важливості ознак;

4) Візуалізація роботи найкращого методу машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів;

5) Матриця плутанини найкращої моделі машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання 11.03

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Аналіз предметної області	11.03	31.03	вс
2	Порівняльний аналіз проблематики	01.04	15.04	вс
3	Вибір оптимальних інформаційних технологій	16.04	30.04	вс
4	Передбачення поведінки клієнтів при онлайн бронюванні готелів	01.05	15.05	вс
5	Оформлення матеріалів до захисту БДР	16.05	31.05	вс

Студент



Марина ГРИГОРЧУК

Керівник роботи



Олексій КОЗАЧКО

АНОТАЦІЯ

Бакалаврська дипломна робота складається з 69 сторінок формату А4, на яких є 5 рисунка, список використаних джерел містить 20 найменувань.

Метою роботи є підвищення точності передбачення поведінки клієнтів при онлайн бронюванні готелів за допомогою системного аналізу.

В роботі проведено розвідувальний аналіз даних поведінки клієнтів при онлайн бронюванні готелів. Для передбачення поведінки клієнтів при онлайн бронюванні готелів використано моделі машинного навчання, а саме: логістична регресія, випадковий ліс та градієнтний бустинг. Для реалізації роботи використано мову Python.

Ключові слова: системний аналіз, передбачення даних, машинне навчання, Python.

ABSTRACT

The bachelor thesis consists of 69 pages of A4 format, on which there are 53 figures, the list of used sources contains 20 titles.

The purpose of the work is a systematic analysis and prediction of customer behavior when booking hotels online.

In the paper, an intelligence analysis of customer behavior data during online hotel booking is carried out. To predict customer behavior when booking hotels online, machine learning models were used, namely: logistic regression, random forest and gradient boosting. The Python language was used to implement the work.

Keywords: system analysis, data prediction, machine learning, Python.

ЗМІСТ

ВСТУП.....	3
1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ОБ'ЄКТУ ДОСЛІДЖЕНЬ.....	5
1.1 Опис систем бронювання готелів	5
1.2 Аналіз методів передбачення поведінки клієнтів при онлайн бронюванні готелів	22
1.3 Висновки.....	24
2. ВИБІР ОПТИМАЛЬНОГО РІШЕННЯ ТА НАЛАШТУВАНЬ ДЛЯ РОЗВ'ЯЗАННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ	25
2.1 Розвідувальний аналіз даних для задачі передбачення онлайн бронювання готелів	25
2.2 Відбір ознак для передбачення даних	35
2.3 Тренування моделей передбачення поведінки клієнтів при бронюванні готелів	37
2.4 Висновки.....	43
3. РЕЗУЛЬТАТИ ПЕРЕДБАЧЕННЯ БРОНЮВАННЯ ГОТЕЛІВ КЛІЄНТАМИ	44
3.1 Передбачення поведінки клієнтів при онлайн бронюванні готелів	44
3.2 Висновки.....	50
ВИСНОВКИ	51
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	51
Додаток А (обов'язковий). Технічне завдання	55
Додаток Б (обов'язковий). Протокол перевірки БДР на наявність текстових запозичень	57
Додаток В (довідниковий). Фрагмент лістингу програми.....	58
Додаток Г (обов'язковий). Ілюстративна частина.....	65

ВСТУП

Актуальність теми. З постійним ростом глобального туризму та збільшенням кількості людей, які користуються онлайн-послугами для планування подорожей, розуміння поведінки клієнтів стає ключовим для успіху в готельному бізнесі. Це також стосується не тільки готелів, а й сервісів, які пропонують проживання, таких як Airbnb.

Аналіз поведінки користувачів дозволяє готелям пропонувати більш персоналізовані пропозиції, що можуть значно підвищити рівень задоволеності клієнтів та їх лояльність. Врахування індивідуальних переваг клієнтів може допомогти в розробці більш ефективних маркетингових стратегій та підвищенні продажів.

Системний аналіз дозволяє готелям краще розуміти фактори, які впливають на рішення про бронювання, та налаштовувати ціни відповідно до змінюваної поведінки споживачів та ринкових умов. Це включає використання динамічного ціноутворення для максимізації доходів.

Знання про те, що клієнти вважають за краще та як вони реагують на різні аспекти сервісу, допомагає готелям покращувати обслуговування, швидко реагувати на відгуки клієнтів і вирішувати проблеми.

В умовах жорсткої конкуренції на ринку гостинності, готелі мають шукати способи виділитись. Використання даних для системного аналізу дає можливість готелям пропонувати унікальні послуги, адаптовані під потреби своїх клієнтів.

Застосування передових технологій для аналізу великих обсягів даних дозволяє виявляти тенденції та шаблони, які не завжди очевидні при традиційних методах аналізу.

Значимість та актуальність системного аналізу поведінки клієнтів при онлайн бронюванні готелів очевидна, і вона продовжуватиме рости з розвитком цифрових технологій та зміною споживацьких звичок.

Мета і задачі дослідження. Метою дослідження є підвищення точності передбачення поведінки клієнтів при онлайн бронюванні готелів за допомогою системного аналізу.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- провести аналіз поведінки клієнтів при онлайн бронюванні готелів;
- здійснити вибір оптимальних технологій машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів;
- застосувати обрану модель машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів.

Об'єктом дослідження є процес передбачення поведінки клієнтів при онлайн бронюванні готелів.

Предметом дослідження є моделі машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів.

1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ОБ'ЄКТУ ДОСЛІДЖЕНЬ

1.1 Опис систем бронювання готелів

Резерв номерів у готелі — це процес, під час якого гість бронює певний номер на визначений період часу перед своїм приїздом. Це дозволяє гарантувати наявність житла при приїзді, а готелю — планувати свої запаси і зайнятість. Резервація може включати вибір типу кімнати, необхідні додаткові послуги, а також може потребувати передоплати або гарантії бронювання з кредитної картки.

Об'єктом дослідження бронювання готелів є процеси та системи, які дозволяють гостям готелів здійснювати резервацію номерів. В рамках даної бакалаврської дипломної роботи розглядаються такі аспекти:

1. Вивчення різних систем бронювання, їх ефективності та інтеграції з іншими сервісами.
2. Аналіз того, як різні фактори (ціна, розташування, відгуки) впливають на вибір готелю.
3. Дослідження стратегій ціноутворення та зайнятості, спрямованих на максимізацію прибутку.

Таке дослідження може включати збір та аналіз даних, моделювання споживчих тенденцій та розробку стратегій для підвищення ефективності бронювання. [1]

Розглянемо декілька видів систем бронювання кімнат в готелі.

1. Глобальні дистрибуційні системи (GDS) — ці системи дозволяють туристичним агенціям і компаніям по всьому світу бронювати готельні номери, а також інші послуги, такі як авіаквитки та автомобілі на прокат.

2. Централізовані системи бронювання (CRS) — це внутрішні системи готелів, які об'єднують усі бронювання з різних джерел в одному місці, забезпечуючи кращий контроль над доступністю кімнат.

3. Менеджери бронювань онлайн (Booking Engine) – WEB-базовані рішення, які інтегруються з вебсайтами готелів, дозволяючи гостям бронювати номери безпосередньо через інтернет.

4. Системи управління відносинами з клієнтами (CRM) – інтегровані системи, що використовують інформацію про клієнтів для персоналізації обслуговування і збільшення задоволення гостей.

Глобальні дистрибуційні системи (GDS) — це комп'ютеризовані мережі, які забезпечують транзакції між туристичними компаніями та постачальниками послуг, такими як готелі, авіакомпанії та прокатні служби. Вони дозволяють турагенціям і корпоративним агентам ефективно бронювати і керувати подорожами, забезпечуючи доступ до широкої бази даних про наявність, ціни і умови.

До найвідоміших глобальних дистрибуційних систем належать:

- Amadeus є однією з провідних глобальних дистрибуційних систем, яка використовується в туристичній індустрії для бронювання авіаквитків, готельних номерів, автомобілів на прокат та інших подорожніх послуг. Система забезпечує зв'язок між постачальниками туристичних послуг та туристичними агентствами через високотехнологічну платформу, що дозволяє їм ефективно керувати подорожами і бронюваннями. Amadeus важлива для автоматизації процесів у туристичному бізнесі, пропонуючи широкий спектр інструментів для оптимізації та управління подорожніми послугами (рис. 1.1) [1];

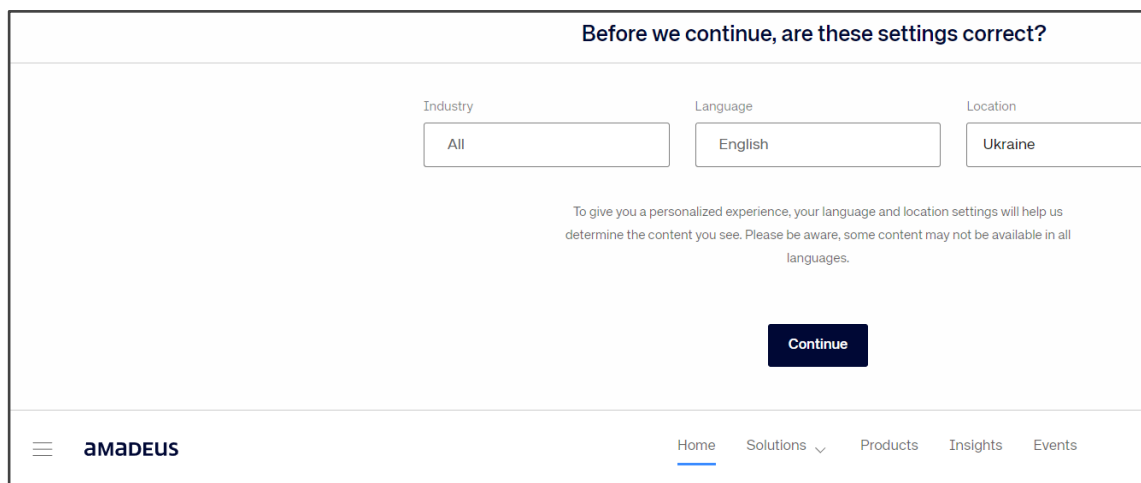


Рисунок 1.1 – Головна сторінка Amadeus

- Sabre є однією з провідних глобальних дистрибуційних систем (GDS), яка широко використовується в туристичній індустрії. Система Sabre допомагає туристичним агентствам, авіакомпаніям, готелям та іншим постачальникам послуг управляти бронюванням, ціноутворенням та розподілом туристичних продуктів. Вона надає інструменти для планування подорожей, автоматизації продажів і підвищення ефективності бізнес-операцій. Sabre має значний вплив на розвиток туристичного ринку, надаючи новітні технологічні рішення для оптимізації подорожей та туризму (рис. 1.2) [2] ;

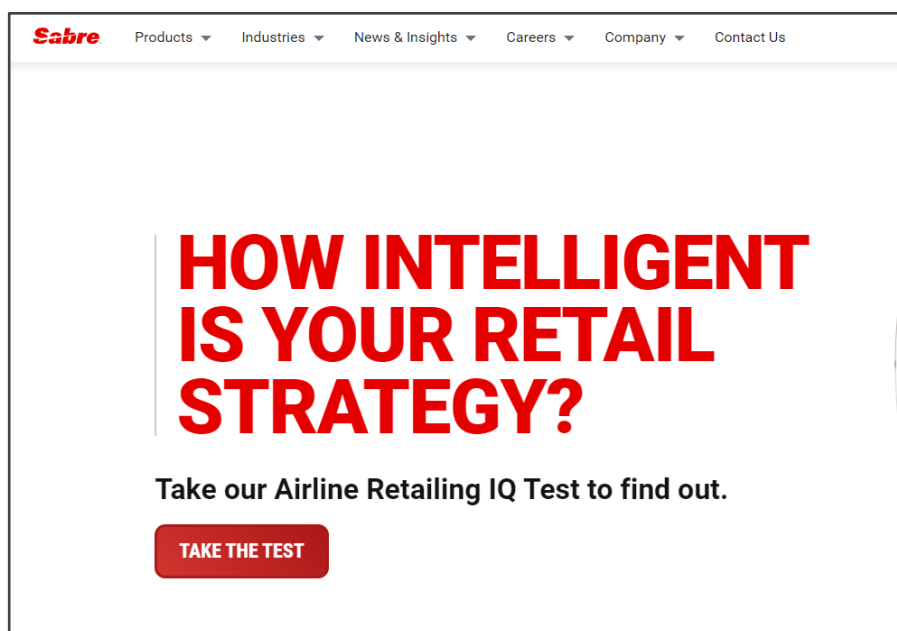


Рисунок 1.2 – Головна сторінка Sabre

- Galileo є однією з провідних глобальних дистрибуційних систем, використовуваною у туристичній індустрії для бронювання авіаквитків, готельних номерів, транспорту та інших подорожніх послуг. Ця система дозволяє туристичним операторам та агентствам швидко знаходити та резервувати потрібні ресурси, оптимізувати маршрути подорожей та управляти клієнтською інформацією. Galileo відома своєю здатністю інтегрувати різні туристичні послуги в єдину платформу, що сприяє ефективності та зручності обслуговування (рис. 1.3) [2];

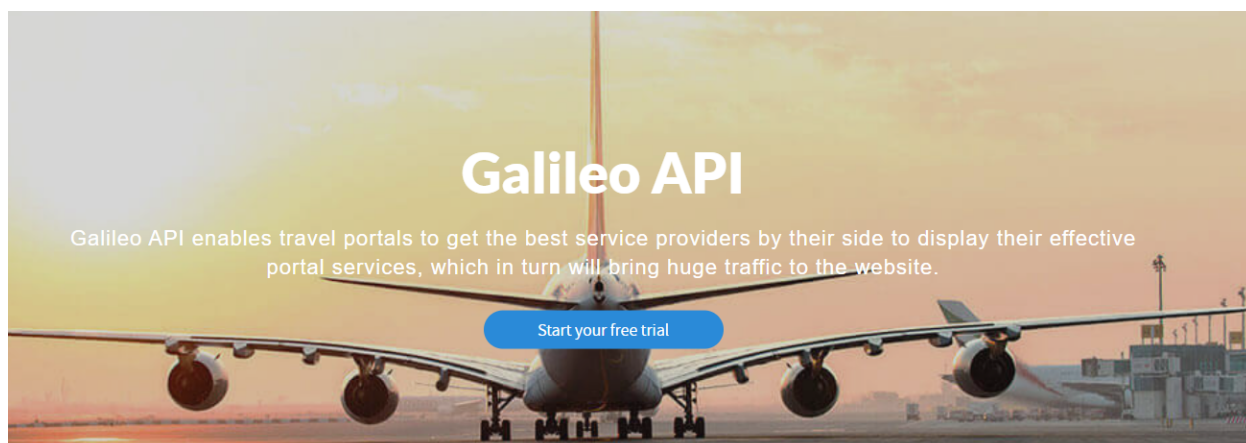


Рисунок 1.3 – Система Galileo

- Worldspan є ще однією з глобальних дистрибуційних систем, яка забезпечує технологічні послуги та рішення для туристичної індустрії. Ця система дозволяє туристичним агентствам та корпоративним клієнтам доступ до різноманітних туристичних продуктів, включаючи авіаквитки, готельні номери, прокат автомобілів та інші подорожні послуги. Worldspan використовується для бронювання, управління подорожами та оптимізації взаємодії між усіма учасниками туристичного ринку (рис. 1.4)



Рисунок 1.4 – Система Worldspan

Централізовані системи бронювання (CRS) використовуються готелями для управління всіма аспектами бронювань. CRS забезпечують інтеграцію даних про доступність номерів, ціни, та інформацію про гостей в одну систему, що полегшує доступ та обробку інформації про бронювання. Вони дозволяють готелям ефективно управляти заброньованими номерами та максимізувати доходи.

Приклади CRS:

- Opera Property Management System (PMS) є однією з найбільш розповсюджених централізованих систем бронювання, використовуваною в готельній індустрії. Ця система допомагає готелям керувати різними аспектами операцій, включаючи бронювання номерів, управління гостьовими обліковими записами, івент-менеджмент, а також функції звітності та аналізу. Opera PMS інтегрує всі готельні послуги в єдину систему, що забезпечує ефективне обслуговування гостей та оптимізацію роботи готелю (рис. 1.5)

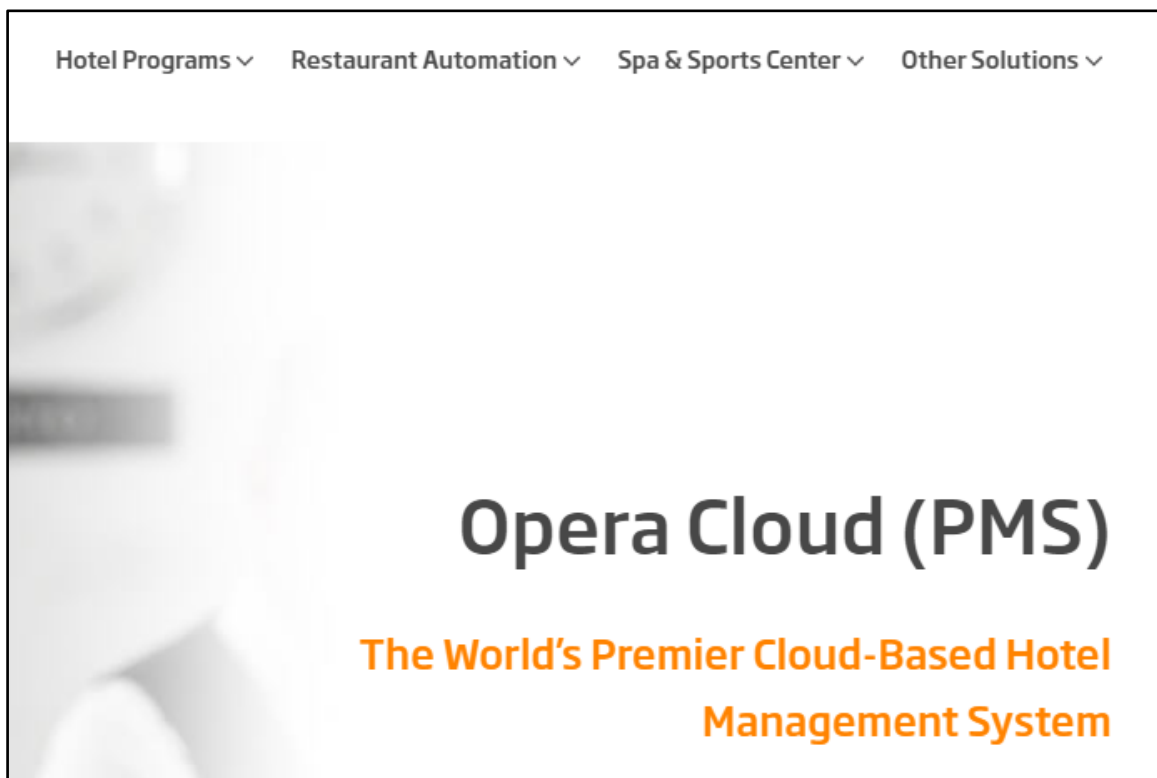


Рисунок 1.5 – Opera Property Management System

- Amadeus Hospitality є частиною ширшої групи послуг, які надає Amadeus IT Group, і включає в себе різні рішення для управління готельним бізнесом. Ця платформа допомагає готелям оптимізувати їхні процеси, від бронювань до гостьового обслуговування та управління виручкою. [3]

Amadeus Hospitality пропонує наступні функціональні можливості:

1. Central Reservations Systems (CRS) – централізована система бронювань, яка дозволяє готелям управляти бронюваннями через різні канали, забезпечуючи консолідацію інформації та поліпшене управління засобами.

2. Property Management Systems (PMS) – система управління власностями, що інтегрує всі оперативні аспекти готельної діяльності, включаючи фронт- та бек-офіс операції.

3. Service Optimization Solutions – рішення для оптимізації сервісу, що допомагає готелям управляти та координувати задачі обслуговуючого персоналу для покращення якості обслуговування.

Ця система забезпечує високий рівень інтеграції між різними службами готелю, що дозволяє більш ефективно використовувати ресурси та забезпечувати високий рівень задоволення гостей (рис. 1.6) [3].

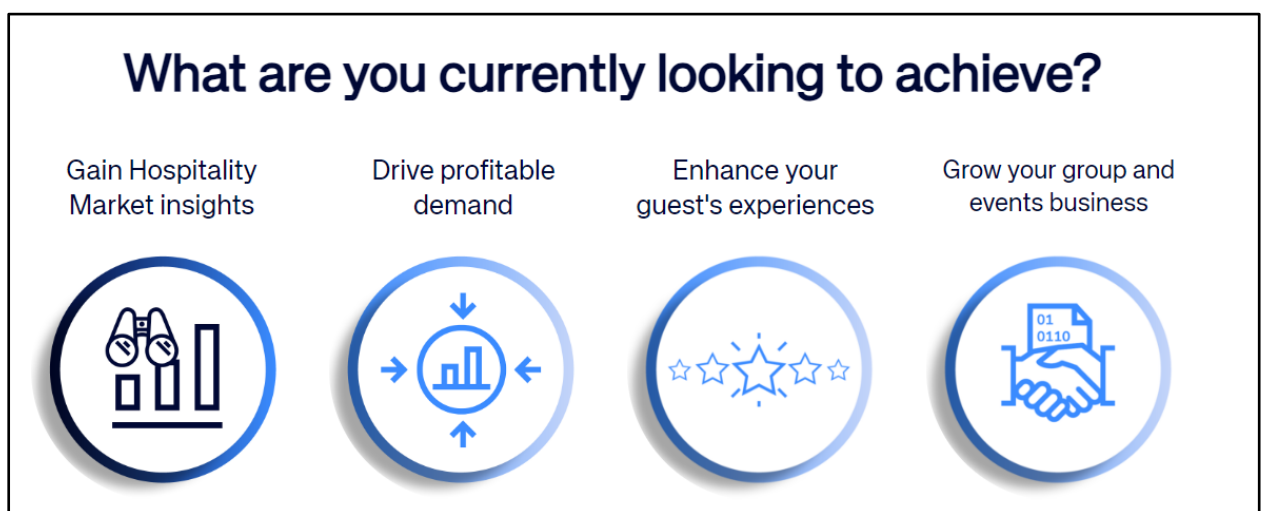


Рисунок 1.6 – Amadeus Hospitality

Менеджери бронювань онлайн, або Booking Engine, це спеціалізовані веб-базовані платформи, які дозволяють готелям приймати і обробляти резервації напряду зі своїх вебсайтів. Ці системи інтегровані з CMS готелю (система управління контентом) і забезпечують автоматизацію процесу бронювання, включаючи вибір номерів, перевірку наявності, обробку платежів та управління клієнтською інформацією. [3]

Основні переваги менеджерів бронювань онлайн:

1. Забезпечення прямого каналу продажів, який уникає комісій стороннім агентствам.
2. Готелі мають повний доступ до інформації про бронювання, що дозволяє краще планувати і управляти наявністю номерів.
3. Можливість пропонувати персоналізовані пропозиції та знижки для повторних відвідувачів або спеціальні пакети.

Приклади популярних менеджерів бронювань онлайн:

- Booking.com Engine є одним з найпопулярніших інструментів для онлайн бронювання, який дозволяє готелям інтегрувати функціонал бронювання прямо на свої вебсайти. Це рішення дозволяє гостям здійснювати резервації безпосередньо, минутих сторонніх агентів, що може сприяти збільшенню прямих продажів і зменшенню залежності від посередників. Booking.com відомий своїм широким охопленням і сильною брендовою присутністю, що робить його привабливим вибором для багатьох готелів (рис 1.7) [4].

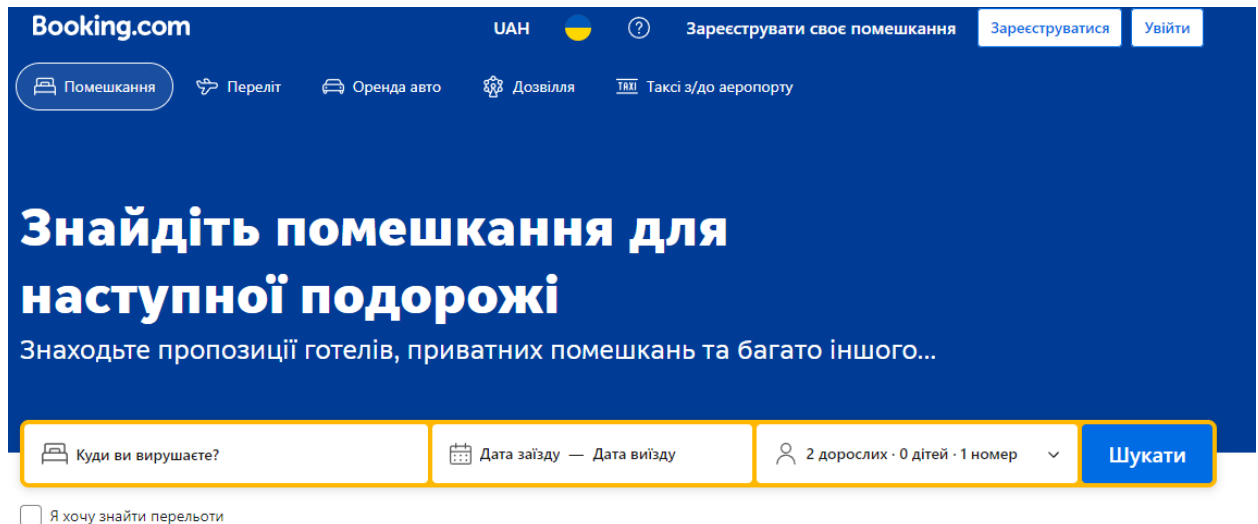


Рисунок 1.7 – Головна сторінка Booking.com

- Expedia Partner Central (EPC) є платформою для партнерів, яку надає Expedia Group, щоб допомогти готелям та іншим постачальникам туристичних послуг управляти своїми бронюваннями, тарифами, наявністю номерів, та маркетингом через один централізований інтерфейс. EPC дозволяє готелям забезпечувати оновлену інформацію про номери, встановлювати ціни та отримувати аналітику щодо ефективності їхніх пропозицій.

Основні можливості Expedia Partner Central:

1. Готелі можуть легко оновлювати наявність номерів і ціни в реальному часі, що дозволяє оптимізувати зайнятість та доходи.
2. EPC надає інструменти для створення та керування спеціальними пропозиціями та знижками, спрямованими на залучення більшої кількості бронювань.
3. Платформа включає функції для підвищення видимості на Expedia Group сайтах, що допомагає партнерам збільшити свою конкурентоспроможність.
4. Готелі мають доступ до детальної аналітики про свої продажі, відгуки клієнтів, ринкові тенденції та інші ключові показники, що допомагають приймати обґрунтовані рішення.

Expedia Partner Central забезпечує готелям потужний набір інструментів для управління своїм онлайн-присутністю і оптимізації процесу бронювання, забезпечуючи гнучкість та контроль над розподілом своїх ресурсів (рис. 1.8) [4].

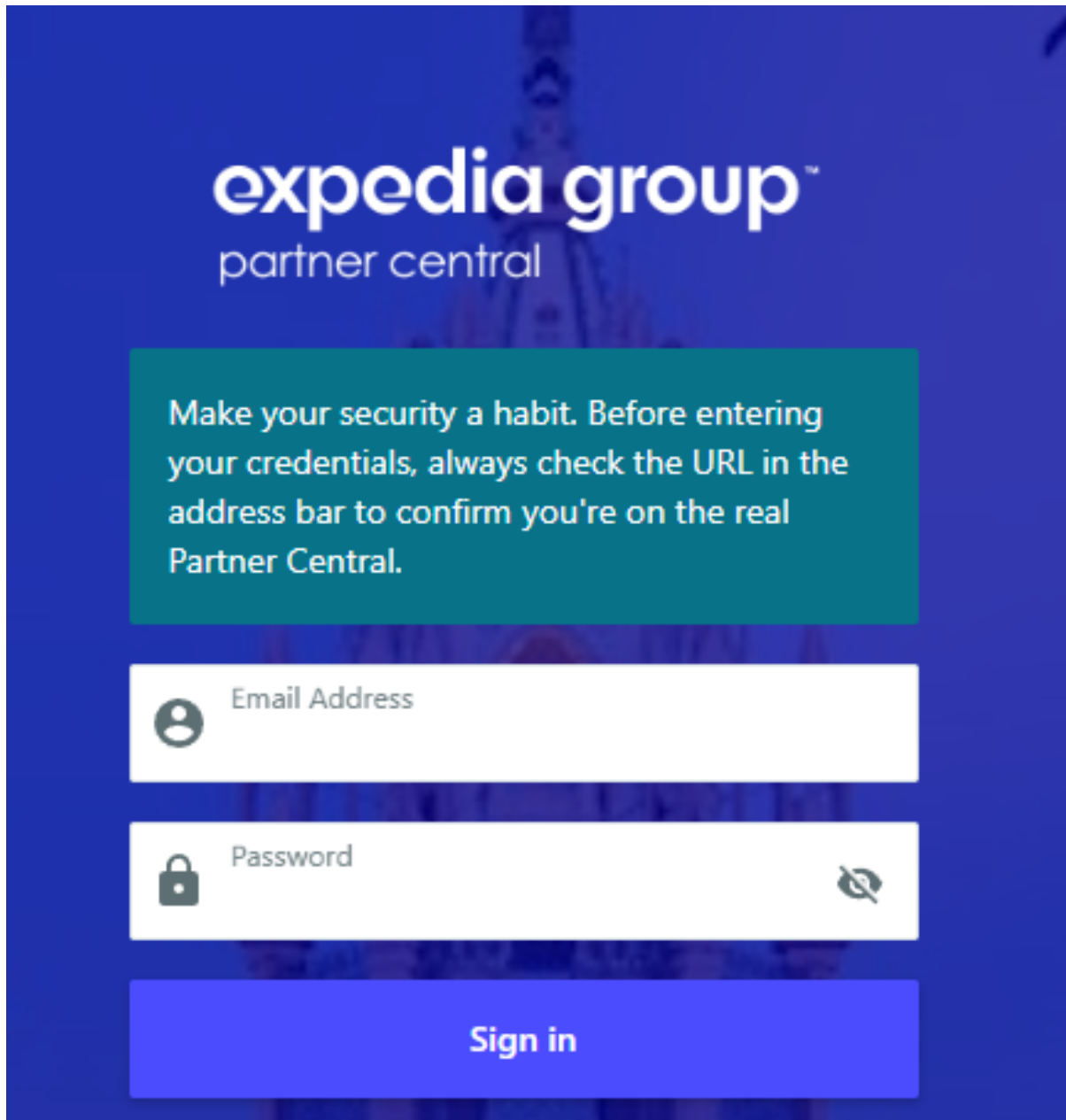


Рисунок 1.8 – Expedia Partner Central

- SynXis Booking Engine є частиною Sabre Hospitality Solutions і представляє собою передову онлайн-систему бронювання, яка дозволяє готелям оптимізувати досвід прямих бронювань через свої вебсайти. Ця платформа надає

готелям можливість керувати наявністю, цінами та бронюваннями в реальному часі, забезпечуючи безперебійний досвід бронювання для гостей (рис. 1.9).

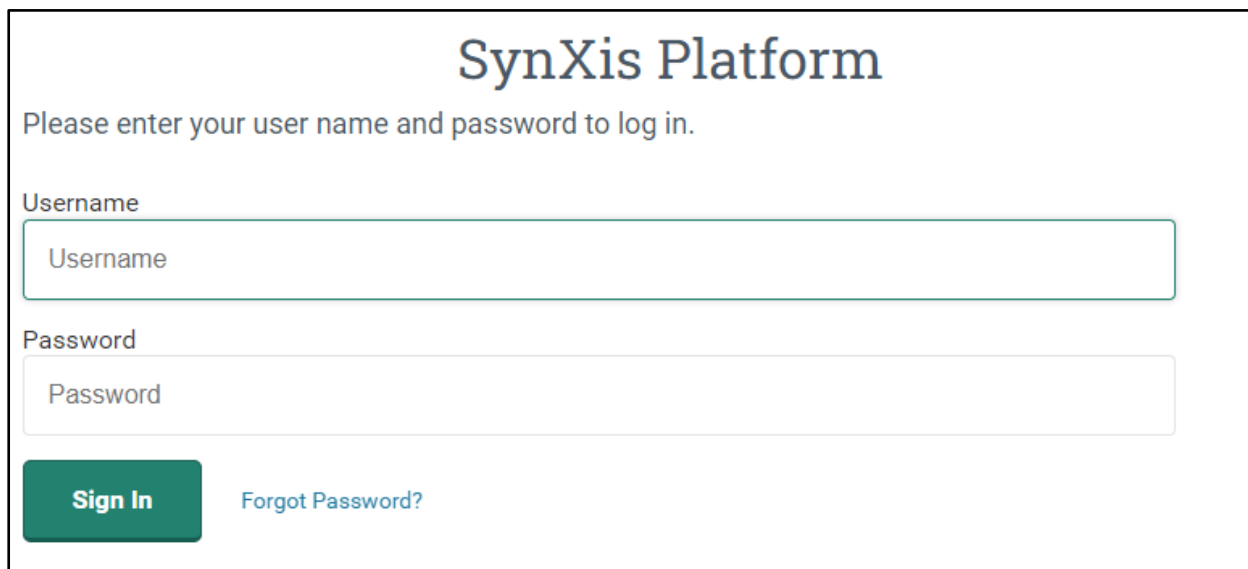
The image shows a login form for the SynXis Platform. At the top, the title "SynXis Platform" is displayed in a large, blue, serif font. Below the title, a message in a smaller, blue, sans-serif font reads "Please enter your user name and password to log in." The form contains two input fields: the first is labeled "Username" in a small, grey, sans-serif font, and the second is labeled "Password" in the same font. Both fields have a light grey border and a placeholder text matching their respective labels. Below the password field, there is a green rectangular button with the text "Sign In" in white, sans-serif font. To the right of the button, there is a link labeled "Forgot Password?" in a blue, sans-serif font.

Рисунок 1.9 – SynXis Booking Engine

Системи управління відносинами з клієнтами (CRM) — це технологічні рішення, що допомагають компаніям керувати і аналізувати взаємодії з клієнтами протягом усього їхнього життєвого циклу з метою покращення комерційних відносин, збільшення лояльності та зростання продажів. Ці системи дозволяють збирати, аналізувати та використовувати інформацію про клієнтів для вдосконалення сервісу і маркетингових стратегій.

Основні можливості CRM систем включають:

1. CRM системи зберігають детальну інформацію про клієнтів, включаючи історію покупок, переписку, переваги та будь-яку взаємодію.
2. Ці системи дозволяють компаніям кластеризувати клієнтів за різними критеріями, щоб краще зрозуміти їхні потреби та розробити цілеспрямовані маркетингові кампанії.
3. CRM допомагає автоматизувати відправлення електронних листів, соціальних медіа повідомлень та інших форм комунікації.
4. Системи CRM пропонують інструменти для аналізу ефективності бізнес-процесів, оцінки поведінки клієнтів та моніторингу продажів.

5. Більшість CRM систем можуть інтегруватися з іншими бізнес-інструментами, такими як ERP системи, платформи електронної комерції та бухгалтерські програми.

Опис популярних CRM систем.

Salesforce CRM є однією з провідних платформ управління відносинами з клієнтами, яка використовується компаніями по всьому світу для автоматизації продажів, маркетингу, сервісу та багатьох інших функцій. Salesforce відомий своєю масштабованістю, гнучкістю та глибокою інтеграцією з різноманітними бізнес-процесами. Система пропонує комплексні рішення для різних секторів та розмірів бізнесу, забезпечуючи інструменти для залучення клієнтів, збереження та розвитку відносин з ними. [5]

Основні можливості Salesforce CRM включають:

1. Інструменти для керування продажами дозволяють трекінг лідів, оптимізацію воронки продажів, автоматизацію процесів замовлення і закриття угод.

2. Інтеграція з Salesforce Marketing Cloud надає розширені можливості для кампаній, ведення електронного маркетингу та персоналізації взаємодії з клієнтами.

3. Сервісні інструменти допомагають управлінню запитами клієнтів, покращенню підтримки та швидкому вирішенні проблем.

4. Використання Salesforce Einstein AI дозволяє аналізувати великі обсяги даних, прогнозувати тенденції та приймати рішення на основі даних.

5. Salesforce дозволяє інтегрувати сторонні додатки, системи ERP, соціальні медіа та інші інструменти через свою відкриту платформу Salesforce AppExchange.

Впровадження Salesforce CRM дозволяє компаніям:

- Збільшити ефективність продажів і маркетингу;
- Покращити задоволеність та лояльність клієнтів;

- Оптимізувати внутрішні процеси та зменшити витрати;
- Приймати кращі рішення на основі аналітичних даних.

Salesforce є однією з найпотужніших CRM систем на ринку і є вибором багатьох великих та малих компаній за її гнучкість та велику екосистему інтеграційних можливостей (рис. 1.10).

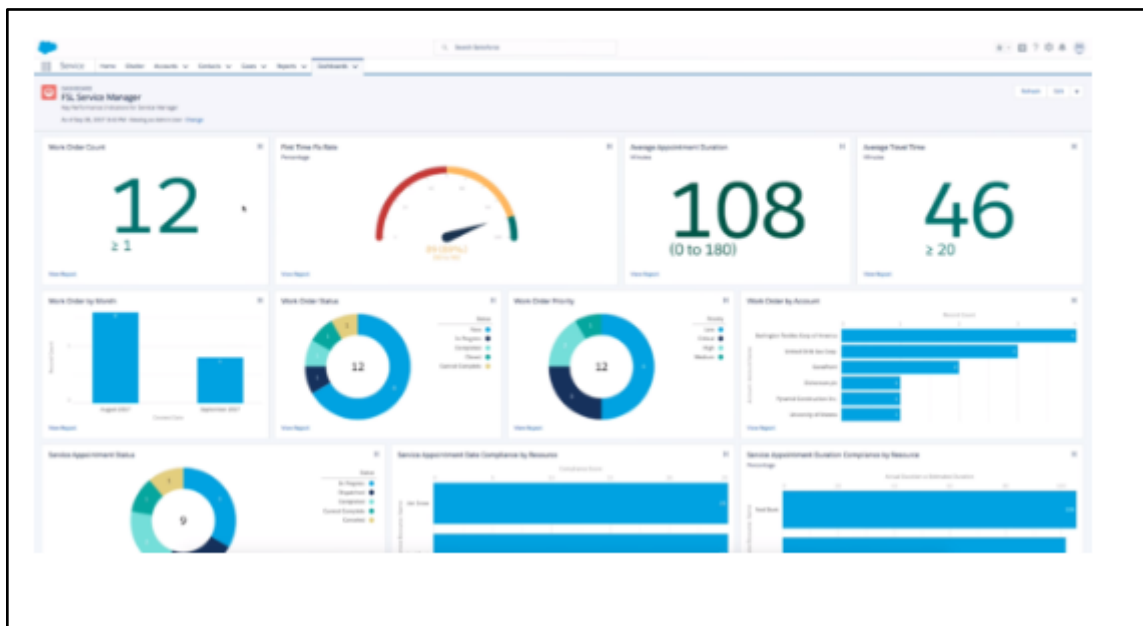


Рисунок 1.10 – Salesforce CRM

HubSpot CRM є однією з найпопулярніших платформ управління відносинами з клієнтами, особливо серед малих та середніх підприємств. Вона вирізняється тим, що пропонує безкоштовний план з основними CRM-функціями, що робить її доступною для стартапів та малих компаній, які тільки починають використовувати CRM-системи. HubSpot CRM охоплює усі основні аспекти ведення клієнтської бази та автоматизації маркетингу, продажів та обслуговування.

Основні можливості HubSpot CRM включають:

1. Збір та зберігання інформації про клієнтів, автоматичне відстеження взаємодій із контактами, включаючи електронні листи, дзвінки, та зустрічі;

2. Інтеграція з інструментами для автоматизації маркетингу дозволяє керувати електронними кампаніями, створювати лендінг-сторінки та вести блоги безпосередньо з CRM;

3. Інструменти для організації продажів, такі як ведення воронки продажів, створення звітів та керування продуктивністю продажів;

4. Платформа містить інструменти для обслуговування клієнтів, як-от тікет-система для управління запитами клієнтів і база знань для самостійного обслуговування;

5. Розширена аналітика для моніторингу KPI, вимірювання продуктивності команди, і виявлення трендів продажів та маркетингу.

6. Можливість інтегрувати з багатьма сторонніми додатками та сервісами через HubSpot App Marketplace, що дозволяє підвищити функціональність системи. [5]

Переваги HubSpot CRM:

- Легкість використання: Інтерфейс користувача інтуїтивно зрозумілий, що спрощує навчання і адаптацію співробітників;
- Безкоштовний вступний план: Доступ до базових CRM-функцій без витрат, що є ідеальним для малих бізнесів або стартапів;
- Легко масштабується з ростом бізнесу, з опціями для додавання додаткових функцій і модулів.

HubSpot CRM є відмінним вибором для тих, хто шукає комплексне рішення для керування відносинами з клієнтами, яке легко інтегрувати і адаптувати до потреб свого бізнесу (рис. 1.11).

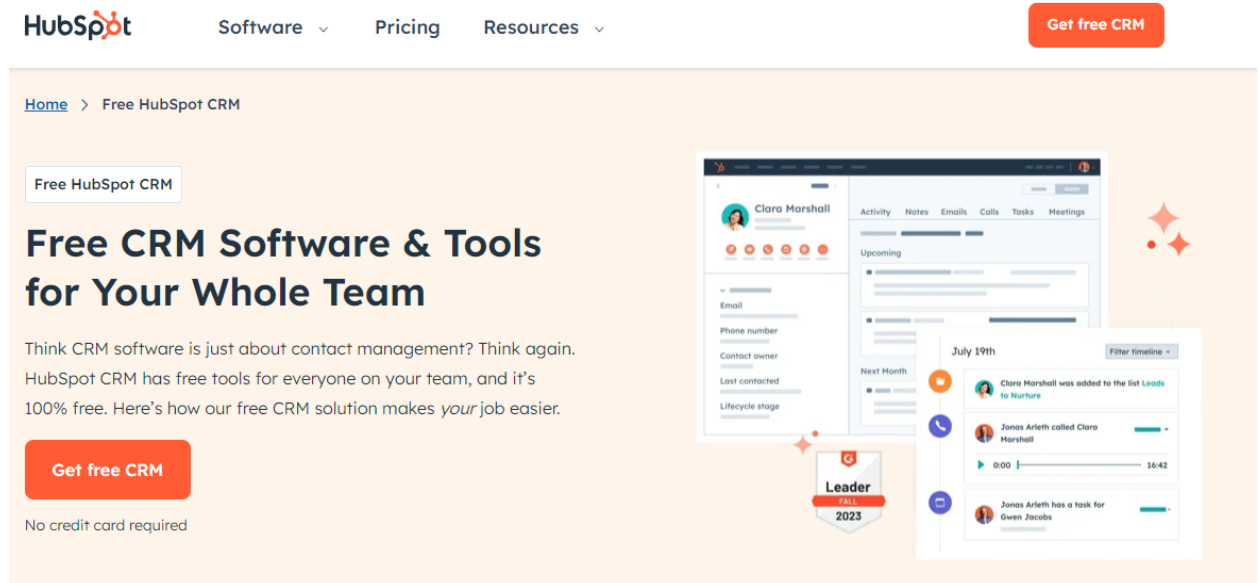


Рисунок 1.11 – HubSpot CRM

Microsoft Dynamics 365 є комплексною бізнес-платформою, яка об'єднує CRM (управління відносинами з клієнтами) та ERP (планування ресурсів підприємства) функціональності в єдину об'єднану хмарну службу. Ця платформа допомагає організаціям оптимізувати свої операції, підвищити продуктивність продажів, поліпшити обслуговування клієнтів і забезпечити більшу гнучкість у веденні бізнесу. Microsoft Dynamics 365 є потужним рішенням для компаній, які шукають єдину платформу для керування всіма аспектами своєї діяльності, від продажів і клієнтського сервісу до маркетингу та оперативного управління [5].

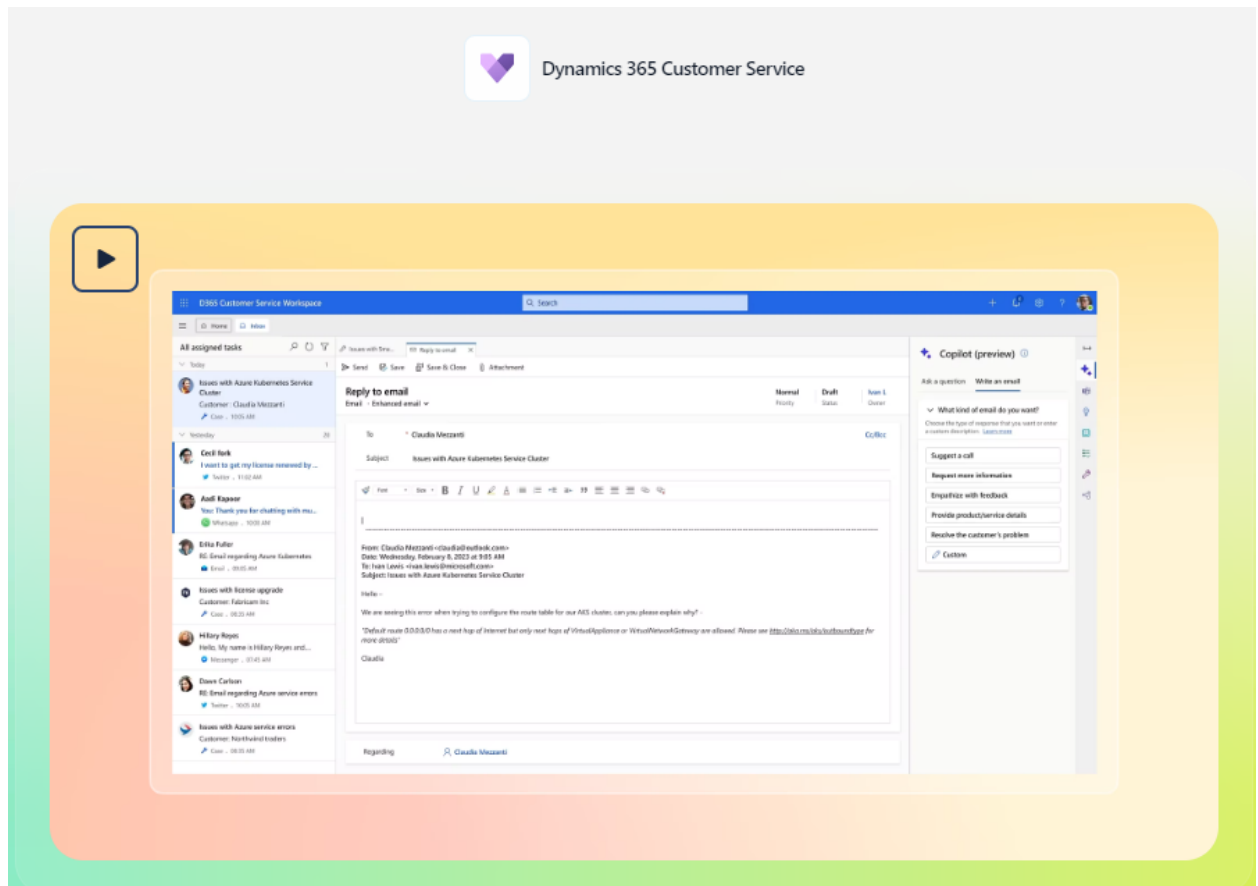


Рисунок 1.12 – Microsoft Dynamics 365

Oracle CRM є частиною Oracle Customer Experience Cloud (Oracle CX), яка надає комплексний набір рішень для управління відносинами з клієнтами. Ця платформа призначена для того, щоб допомогти організаціям зростати через збільшення взаємодії з клієнтами, підвищення задоволеності та лояльності клієнтів, а також оптимізації маркетингових і продажових процесів.

Основні компоненти Oracle CRM включають:

1. Oracle Sales Cloud – забезпечує управління продажами, продуктивність та планування. Включає автоматизацію продажів, управління контактами та воронкою продажів, аналітику та звітність;
2. Oracle Service Cloud – включає інструменти для центру обслуговування клієнтів, веб-служби для самообслуговування, управління базою знань, та автоматизацію обслуговування;

3. Oracle Marketing Cloud – пропонує рішення для автоматизації маркетингу, управління кампаніями, цільової персоналізації та залучення клієнтів через різні канали;

4. Oracle Commerce Cloud – забезпечує платформу для електронної комерції, яка дозволяє компаніям створювати та керувати динамічними, багатофункціональними онлайн-магазинами;

5. Oracle Social Cloud – інтегрує соціальні медіа з іншими каналами обслуговування, забезпечуючи аналіз та взаємодію з клієнтами через соціальні мережі.

Переваги Oracle CRM:

- Oracle CRM може масштабуватися для задоволення потреб як невеликих, так і великих підприємств;
- Забезпечує високий рівень інтеграції з іншими корпоративними рішеннями Oracle, такими як Oracle ERP, Oracle Supply Chain Management (SCM), та іншими;
- Використання передових аналітичних інструментів для виявлення трендів, прогнозування поведінки клієнтів та оптимізації стратегій взаємодії.

Oracle CRM є потужним рішенням для комплексного управління відносинами з клієнтами, яке допомагає покращити взаємодію з клієнтами, збільшити доходи та підвищити ефективність маркетингових і продажних ініціатив (рис 1.13).

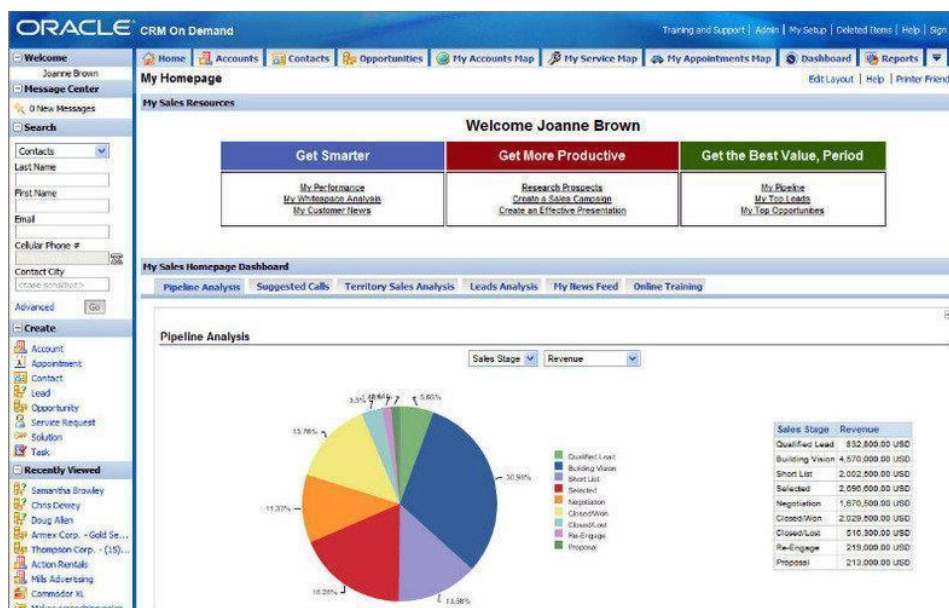


Рисунок 1.13 – Oracle CRM

Ці системи є життєво необхідними для сучасних компаній, оскільки вони допомагають підвищити задоволеність клієнтів, підтримувати з ними довгострокові відносини та збільшувати продажі.

Ці системи стають все більш популярними в готельній індустрії, оскільки вони забезпечують ефективне та зручне управління резерваціями.

Отже, аналіз та передбачення поведінки клієнтів при онлайн бронюванні готелів є популярною задачею для розвитку готельного бізнесу.

1.2 Аналіз методів передбачення поведінки клієнтів при онлайн бронюванні готелів

Аналіз поведінки клієнтів при онлайн бронюванні готелів використовує різні методи і технології для збору, обробки та інтерпретації даних про користувачів. Ці дані допомагають готелям розуміти, які фактори впливають на рішення клієнтів бронювати певний готель, і пропонувати персоналізовані послуги та пропозиції. Ось деякі з ключових методів: [6]

1. Часові ряди. Цей метод аналізує дані, зібрані протягом часу, для виявлення сезонних трендів, циклічних змін та інших тимчасових патернів. Він

дозволяє передбачати попит на певні періоди та адаптувати маркетингові та цінові стратегії.

2. Класифікація. Методи класифікації, такі як логістична регресія, випадковий ліс і градієнтний бустинг, використовуються для визначення, які фактори найбільше впливають на рішення клієнтів скасувати бронювання або обрати конкретний тип номеру.

3. Кластеризація. Методи кластеризації, як K-means або ієрархічна кластеризація, дозволяють ідентифікувати групи користувачів зі схожими поведінковими характеристиками. Це допомагає у створенні персоналізованих маркетингових кампаній.

4. Машинне навчання. Алгоритми машинного навчання, такі як нейронні мережі, можуть аналізувати великі набори даних для виявлення складних патернів та взаємозв'язків у поведінці клієнтів. Ці системи можуть прогнозувати ймовірність скасування бронювань та впливати на стратегію управління доходами.

5. Поведінковий аналіз. Аналіз веб-поведінки (через веб-аналітику) допомагає зрозуміти, як користувачі взаємодіють з онлайн-бронювальними платформами. Це включає вивчення шляхів навігації, часу на сторінках, і конверсійних шляхів.

6. Сентимент-аналіз. Аналізує відгуки та коментарі клієнтів для виявлення загальних емоційних тенденцій. Це може допомогти визначити сильні та слабкі сторони послуг готелю та реагувати на клієнтські відгуки ефективніше.

Кожен з цих методів вносить свій вклад у створення комплексного розуміння поведінки клієнтів, що дозволяє готелям не тільки оптимізувати свої процеси та пропозиції, але й покращити загальне задоволення та лояльність клієнтів.

Спираючись на дані в Kaggle Dataset «Hotel Reservations Dataset», можна обрати методи класифікації для передбачення поведінки клієнтів при онлайн бронюванні готелів.[6]

1.3 Висновки

В першому розділі проаналізовано сервіси бронювання готелів та їх специфікацію. Визначено основні інформаційні технології, які використовують готелі та сервіси резервування готельних номерів, найпоширеніші з них Amadeus, Booking.com, Sabre та інші.

Проведений огляд методів машинного навчання для передбачення поведінки клієнтів при бронюванні готелів. Обрано методи класифікації для передбачення поведінки клієнтів при онлайн бронюванні готелів, а саме: логістична регресія, випадковий ліс та градієнтний бустинг.

2. ВИБІР ОПТИМАЛЬНОГО РІШЕННЯ ТА НАЛАШТУВАНЬ ДЛЯ РОЗВ'ЯЗАННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

2.1 Розвідувальний аналіз даних для задачі передбачення онлайн бронювання готелів

Для виконання поставленої задачі, передбачення онлайн бронювання готелів, використаємо дані з датасету "Hotel Reservations" на Kaggle. Даний датасет містить інформацію моніторингу CRM системи готелю та має відповідні ознаки [6]:

1. Booking_ID: Унікальний ідентифікатор кожного бронювання;
2. no_of_adults: Кількість дорослих у бронюванні;
3. no_of_children: Кількість дітей у бронюванні;
4. no_of_weekend_nights: Кількість ночей вихідних (субота або неділя), які гість провів або планував провести в готелі;
5. no_of_week_nights: Кількість будніх ночей (понеділок по п'ятницю), які гість провів або планував провести в готелі;
6. type_of_meal_plan: Тип харчування, заброньований клієнтом;
7. required_car_parking_space: Чи потрібне клієнту місце для паркування автомобіля? (0 - Ні, 1 - Так);
8. room_type_reserved: Тип заброньованого номера, значення шифруються готелем INN;
9. lead_time: Кількість днів між датою бронювання та датою прибуття;
10. arrival_year: Рік прибуття;
11. arrival_month: Місяць прибуття;
12. arrival_date: Дата прибуття в місяці;
13. market_segment_type: Тип ринкового сегменту;
14. repeated_guest: Чи є клієнт постійним гостем? (0 - Ні, 1 - Так);
15. no_of_previous_cancellations: Кількість попередніх бронювань; скасованих клієнтом до цього бронювання;

16. `no_of_previous_bookings_not_canceled`: Кількість попередніх бронювань, не скасованих клієнтом до цього бронювання.

17. `avg_price_per_room`: Середня ціна за номер за день бронювання; ціни на номери є динамічними (в євро).

18. `no_of_special_requests`: Загальна кількість спеціальних запитів, зроблених клієнтом (наприклад, номер на високому поверсі, вид з вікна тощо).

19. `booking_status`: Показник, який вказує, чи було скасовано бронювання. Зчитуємо дані та відкрисмо їх (рис 2.1 – 2.3).

	Booking_ID	no_of_adults	no_of_children	no_of_weekend_nights	no_of_week_nights	type_of_meal_plan	required_car_parking_space	room_t
0	INN00001	2	0	1	2	Meal Plan 1	0	
1	INN00002	2	0	2	3	Not Selected	0	
2	INN00003	1	0	2	1	Meal Plan 1	0	
3	INN00004	2	0	0	2	Meal Plan 1	0	
4	INN00005	2	0	1	1	Not Selected	0	
...	...	...	...	...	...	...	...	...
36270	INN36271	3	0	2	6	Meal Plan 1	0	
36271	INN36272	2	0	1	3	Meal Plan 1	0	
36272	INN36273	2	0	2	6	Meal Plan 1	0	
36273	INN36274	2	0	0	3	Not Selected	0	
36274	INN36275	2	0	1	2	Meal Plan 1	0	

36275 rows × 19 columns

Рисунок 2.1 – Дані "Hotel Reservations"

room_type_reserved	lead_time	arrival_year	arrival_month	arrival_date	market_segment_type	repeated_guest	no_of_previous_cancellations
Room_Type 1	224	2017	10	2	Offline	0	0
Room_Type 1	5	2018	11	6	Online	0	0
Room_Type 1	1	2018	2	28	Online	0	0
Room_Type 1	211	2018	5	20	Online	0	0
Room_Type 1	48	2018	4	11	Online	0	0
...	...	...	...	...	...	...	...
Room_Type 4	85	2018	8	3	Online	0	0
Room_Type 1	228	2018	10	17	Online	0	0
Room_Type 1	148	2018	7	1	Online	0	0
Room_Type 1	63	2018	4	21	Online	0	0
Room_Type 1	207	2018	12	30	Offline	0	0

Рисунок 2.2 – Дані "Hotel Reservations"

avg_price_per_room	no_of_special_requests	booking_status
65.00	0	Not_Canceled
106.68	1	Not_Canceled
60.00	0	Canceled
100.00	0	Canceled
94.50	0	Canceled
...	...	...
167.80	1	Not_Canceled
90.95	2	Canceled
98.39	2	Not_Canceled
94.50	0	Canceled
161.67	0	Not_Canceled

Рисунок 2.3 – Дані "Hotel Reservations"

Переглянемо інформацію про типи даних та наявність відсутніх даних (рис.2.4).

#	Column	Non-Null Count	Dtype
0	Booking_ID	36275 non-null	object
1	no_of_adults	36275 non-null	int64
2	no_of_children	36275 non-null	int64
3	no_of_weekend_nights	36275 non-null	int64
4	no_of_week_nights	36275 non-null	int64
5	type_of_meal_plan	36275 non-null	object
6	required_car_parking_space	36275 non-null	int64
7	room_type_reserved	36275 non-null	object
8	lead_time	36275 non-null	int64
9	arrival_year	36275 non-null	int64
10	arrival_month	36275 non-null	int64
11	arrival_date	36275 non-null	int64
12	market_segment_type	36275 non-null	object
13	repeated_guest	36275 non-null	int64
14	no_of_previous_cancellations	36275 non-null	int64
15	no_of_previous_bookings_not_canceled	36275 non-null	int64
16	avg_price_per_room	36275 non-null	float64
17	no_of_special_requests	36275 non-null	int64
18	booking_status	36275 non-null	object

dtypes: float64(1), int64(13), object(5)

Рисунок 2.4 – Інформація про дані

Відповідно до інформації про дані, датасет не має відсутніх даних, має 14 числових ознак та 5 категоріальних.

Першим пунктом дослідження проведемо статистичний аналіз даних за допомогою describe (рис. 2.5) [7].

	no_of_adults	no_of_children	no_of_weekend_nights	no_of_week_nights	required_car_parking_space	lead_time	arrival_year	arrival_month	arrival_date	repeated_guest
count	36275.000000	36275.000000	36275.000000	36275.000000	36275.000000	36275.000000	36275.000000	36275.000000	36275.000000	36275.000000
mean	1.844962	0.105279	0.810724	2.204300	0.030986	85.232557	2017.820427	7.423653	15.596995	0.025637
std	0.518715	0.402648	0.870644	1.410905	0.173281	85.930817	0.383836	3.069894	8.740447	0.158053
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	2017.000000	1.000000	1.000000	0.000000
25%	2.000000	0.000000	0.000000	1.000000	0.000000	17.000000	2018.000000	5.000000	8.000000	0.000000
50%	2.000000	0.000000	1.000000	2.000000	0.000000	57.000000	2018.000000	8.000000	16.000000	0.000000
75%	2.000000	0.000000	2.000000	3.000000	0.000000	126.000000	2018.000000	10.000000	23.000000	0.000000
max	4.000000	10.000000	7.000000	17.000000	1.000000	443.000000	2018.000000	12.000000	31.000000	1.000000

no_of_previous_cancellations	no_of_previous_bookings_not_canceled	avg_price_per_room	no_of_special_requests
36275.000000	36275.000000	36275.000000	36275.000000
0.023349	0.153411	103.423539	0.619655
0.368331	1.754171	35.089424	0.786236
0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	80.300000	0.000000
0.000000	0.000000	99.450000	0.000000
0.000000	0.000000	120.000000	1.000000
13.000000	58.000000	540.000000	5.000000

Рисунок 2.5 – Описова статистика даних

Для аналізу даних з датасету бронювання готелів важливо правильно класифікувати ознаки на числові та категоріальні [8].

Числові ознаки представляють кількісні дані, які можуть бути оброблені математично. У цьому датасеті числовими ознаками є:

- no_of_adults: Кількість дорослих;
- no_of_children: Кількість дітей;
- no_of_weekend_nights: Кількість ночей у вихідні;
- no_of_week_nights: Кількість будніх ночей;
- lead_time: Час ведення в днях до прибуття;

- no_of_previous_cancellations: Кількість попередньо скасованих бронювань;
- no_of_previous_bookings_not_canceled: Кількість попередніх бронювань, що не були скасовані;
- avg_price_per_room: Середня ціна за номер;
- no_of_special_requests: Кількість спеціальних запитів;

Категоріальні ознаками у даному датасеті є:

- type_of_meal_plan: Тип харчування;
- required_car_parking_space: Потреба в паркувальному місці (так/ні);
- room_type_reserved: Тип заброньованого номера;
- arrival_year: Рік прибуття;
- arrival_month: Місяць прибуття;
- arrival_date: Дата прибуття;
- market_segment_type: Тип ринкового сегменту;
- repeated_guest: Чи є клієнт постійним гостем (так/ні);
- booking_status: Статус бронювання (скасовано/не скасовано);

Автоматизуємо розподіл даних та створимо окремі таблиці для кожного типу , код представлений на рисунку 2.6 [9].

```
# Виділення числових ознак
numeric_columns = df.select_dtypes(include=['int64', 'float64']).columns

# Виділення категоріальних ознак
categorical_columns = df.select_dtypes(include=['object', 'category']).columns

# Створення окремих DataFrame для кожного типу
df_numeric = df[numeric_columns]
df_categorical = df[categorical_columns]
```

Рисунок 2.6 – Розподіл ознак за типом даних.

Побудуємо кореляційну матрицю для числових даних (рис. 2.7).

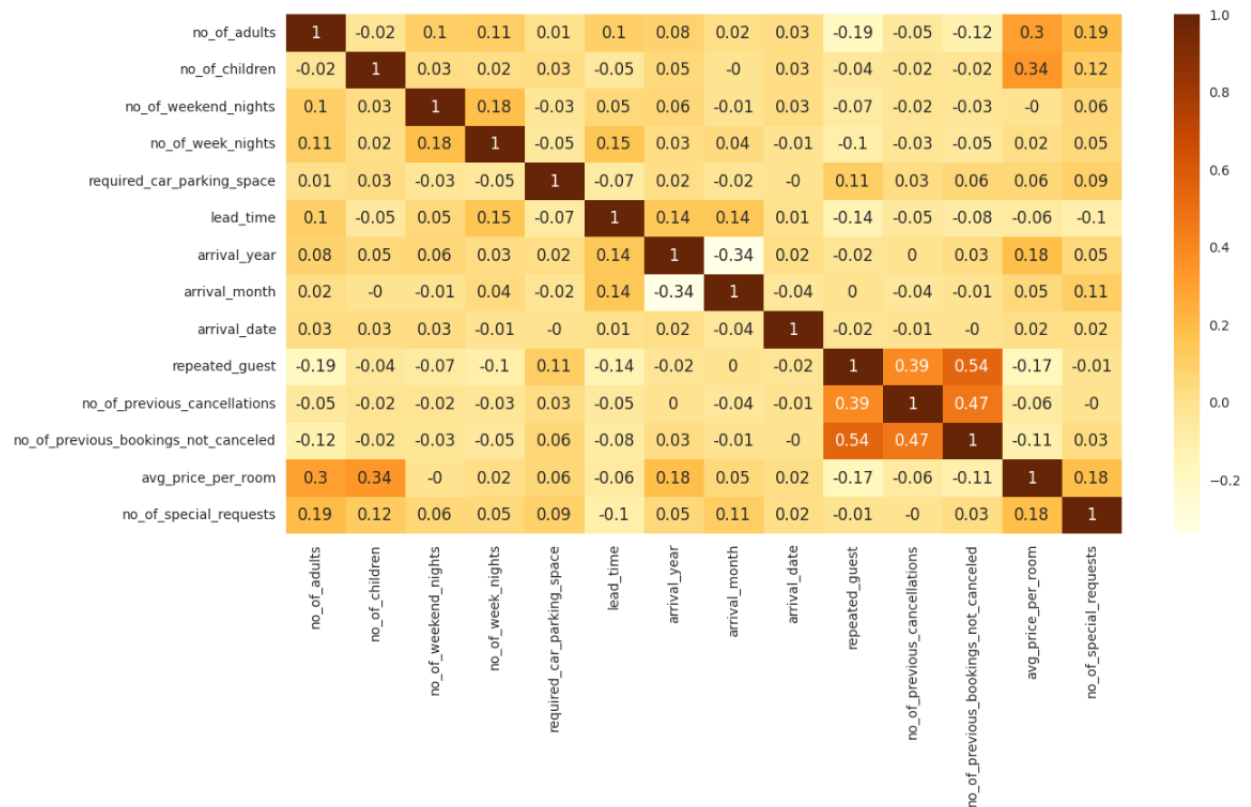


Рисунок 2.7 – Кореляційна матриця числових даних.

Отже, провівши кореляційний аналіз числових даних можна зробити висновок, що стійких впливів та зв'язків між цими даними немає. Оскільки коефіцієнт кореляції не перевищує поріг в 0,6.

Побудуємо гістограми деяких ознак та зробимо їх опис [9].

На рисунку 2.8 зображений графік розподілу ознаки no_of_adults. Основна кількість клієнтів готелю це 2 дорослих, 1 та 3 дорослих бронюють кімнати рідше.

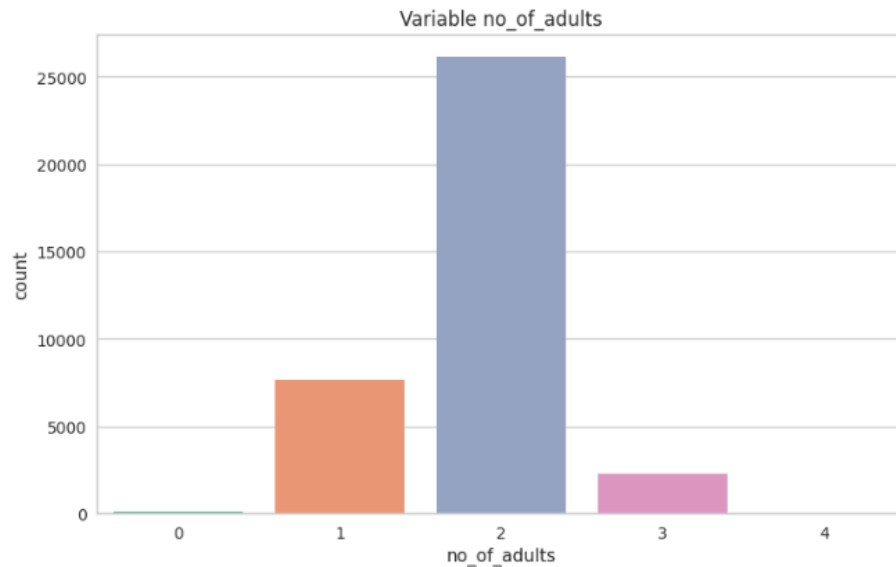


Рисунок 2.8 – Графік розподілу ознаки no_of_adults.

На рисунку 2.9 зображений графік розподілу ознаки no_of_children. Основна кількість орендаторів не мають дітей. Не велика кількість клієнтів мають 1 або 2 дитини.

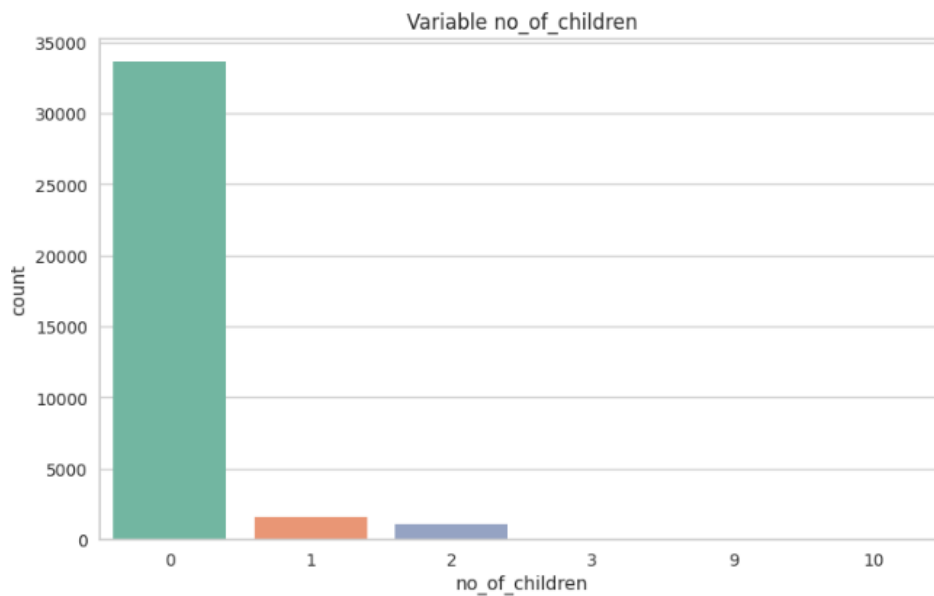


Рисунок 2.9 – Графік розподілу ознаки no_of_children.

На рисунку 2.10 зображений графік розподілу ознаки no_of_weekend_nights. На графіку зображено, що більшість клієнтів не

бронюють кімнати на вихідні дні. Вдвічі менша кількість мають у бронюванні 1 або 2 вихідні дні.

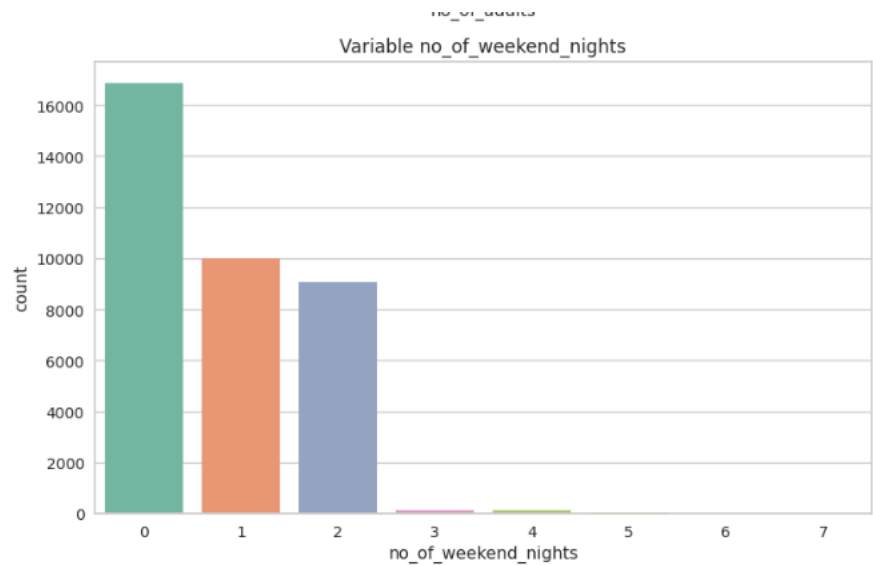


Рисунок 2.10 – Графік розподілу ознаки no_of_weekend_nights.

На рисунку 2.11 зображений розподіл бронювань в будні дні. На графіку зображено, що більшість клієнтів бронюють кімнату на 2 будніх дні.

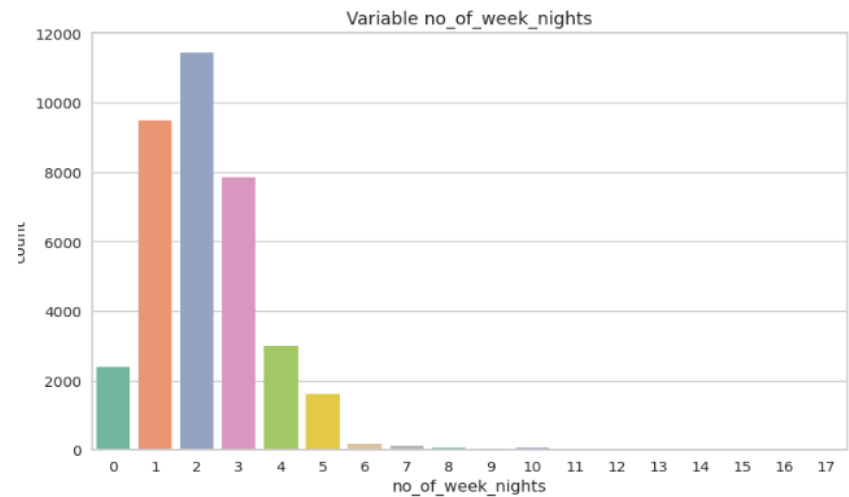


Рисунок 2.11 – Графік розподілу ознаки no_of_week_nights.

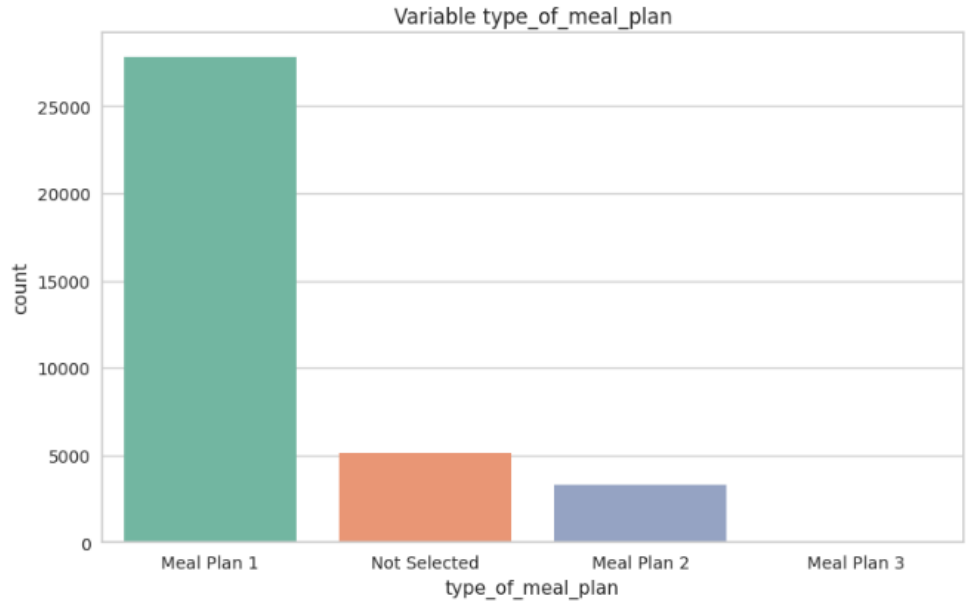


Рисунок 2.12 графік розподілу частоти бронювання за типом харчування.

Найчастіше бронюють з перший тип харчування, рідше без харчування або 2 набір меню.

На рисунку 2.13 зображено поділ бронювань за потребою парковки. Проаналізувавши графік, можна зробити висновок, що більшість клієнтів бронюють номери без парковки.

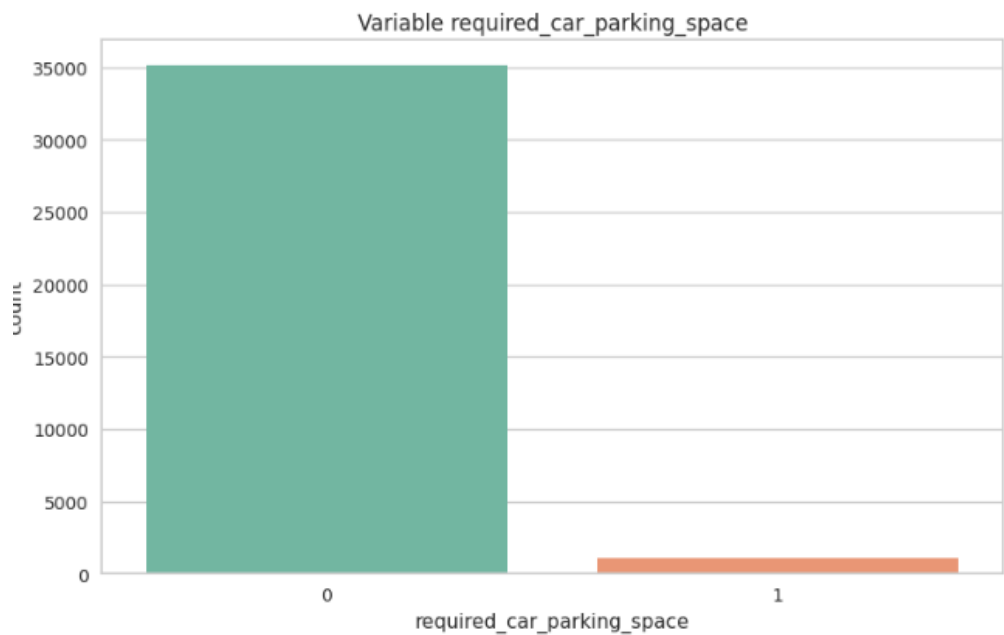


Рисунок 2.13 – Графік поділу за потребою в парковці

Проаналізувавши графік, можна зробити висновок, що більшість клієнтів бронюють номери без парковки.

На рисунку 2.14 зображено розподіл замовлень за типом номеру. На графі основним типом номера бронювання є перший та четвертий типи номерів, інші типи номерів не популярні.

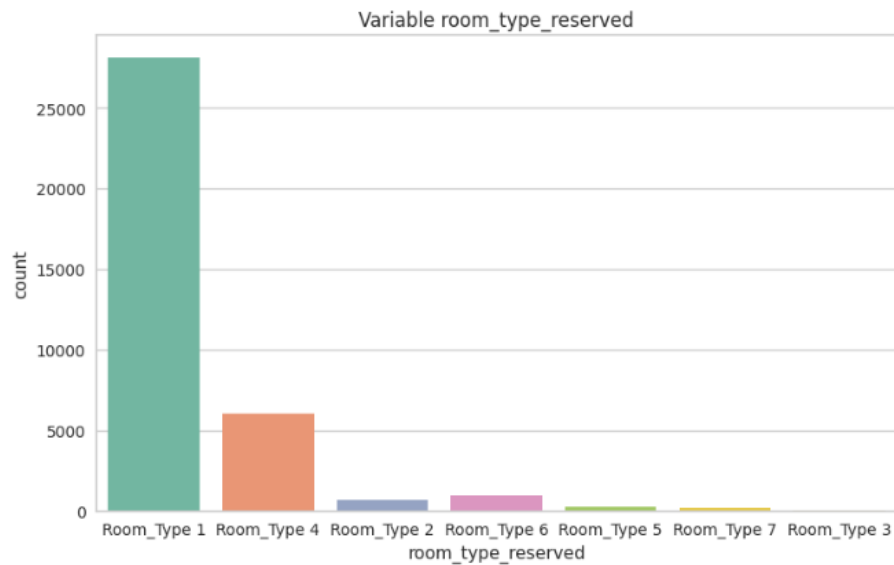


Рисунок 2.14 – Розподіл замовлень за типом номеру.

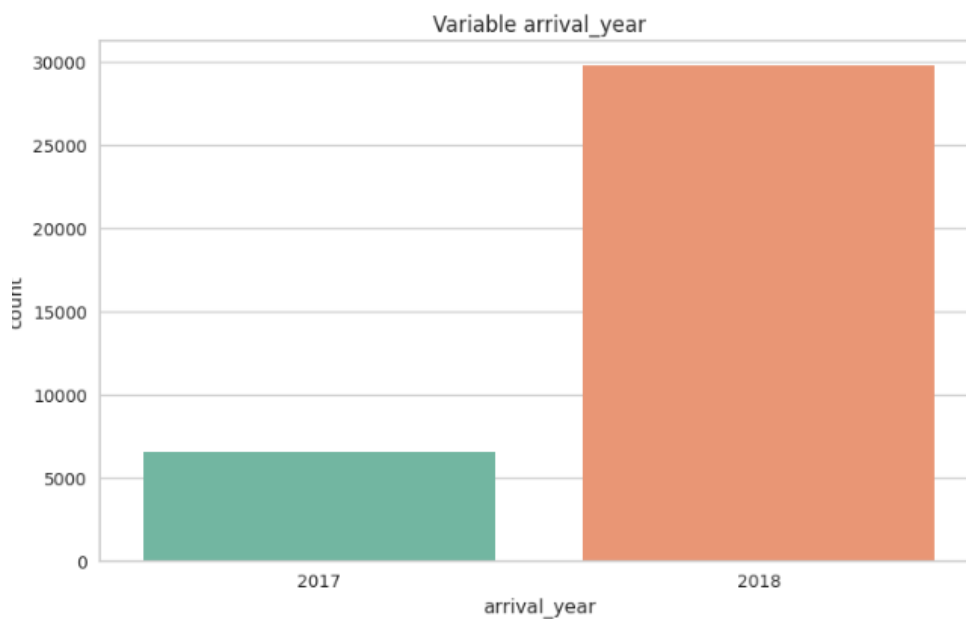


Рисунок 2.15 – Розподіл даних за роком бронювання.

2.2 Відбір ознак для передбачення даних

Перетворимо категоріальні дані в числові. Використаємо бібліотеку sklearn, а саме метод LabelEncoder.

```
from sklearn.preprocessing import LabelEncoder

label_encoder_type_of_meal_plan = LabelEncoder()
label_encoder_room_type_reserved = LabelEncoder()
label_encoder_market_segment_type = LabelEncoder()
label_encoder_booking_status = LabelEncoder()

df['type_of_meal_plan'] = label_encoder_type_of_meal_pl
df['room_type_reserved'] = label_encoder_room_type_rese
df['market_segment_type'] = label_encoder_market_segmen
df['booking_status'] = label_encoder_booking_status.fit
le_name_mapping = dict(zip(label_encoder_booking_status
print(le_name_mapping)
```

Рисунок 2.16 – Використання LabelEncoder

Оскільки дані знаходяться в різних діапазонах проведемо масштабування даних методом StandardScaler (рис. 2.17) [12].

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_standard = scaler.fit_transform(X)
```

Рисунок 2.17 – Масштабування даних методом StandardScaler

Відокремимо цільову ознаку від основного датасету (рис. 2.18).

Розділимо дані на тренувальні та тестові набори за допомогою функції train_test_split (рис. 2.19) [13].

```
X = df.drop('booking_status', axis = 1)
X = X.values
y = df['booking_status']
```

Рисунок 2.18 – Відділення цільової ознаки від основного дасету

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0)
```

Рисунок 2.19 – Розподіл на тренувальна та тестові дані

Побудуємо графік важливості ознак (рис. 2.20).

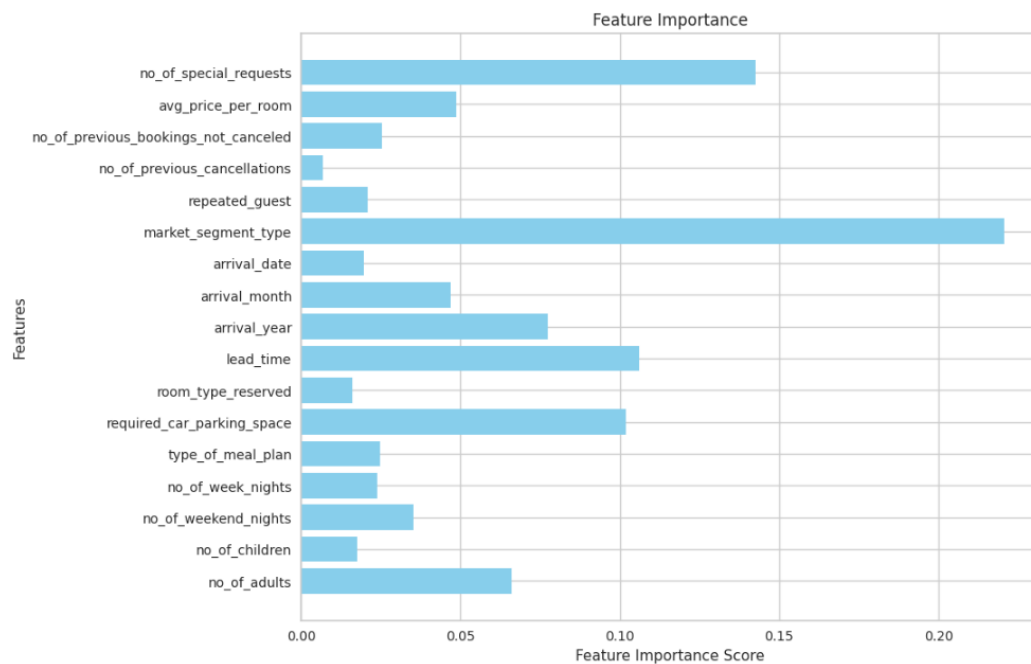


Рисунок 2.20 – Графік важливості ознак

Враховуючи проведені розвідувальний аналіз та інженерію ознак для тренування моделей машинного навчання для передбачення бронювання номерів в готелі використаємо всі наявні ознаки.

2.3 Тренування моделей передбачення поведінки клієнтів при бронюванні готелів

Оскільки основна задача – це передбачення, то для виконання її потрібно використовувати моделі бінарної класифікації. Проаналізуємо відомі моделі машинного навчання для класифікації даних [14].

Моделі бінарної класифікації в машинному навчанні використовуються для передбачення двох класів чи категорій. Ці моделі дуже поширені у багатьох сферах, включаючи медицину, фінанси, маркетинг та багато інших. Ось деякі з найпопулярніших і ефективних моделей бінарної класифікації [15]:

1. Логістична регресія – це статистичний метод, який використовується для моделювання ймовірності бінарної відповіді залежно від однієї або декількох незалежних змінних. Придатний для випадків, коли відповідь має дихотомічний характер (так/ні, успіх/невдача) [16].

2. Дерева рішень – надзвичайно інтуїтивні та легкі для інтерпретації, дерева рішень ділять простір функцій на регіони, використовуючи "якщо-тоді" правила рішень. Часто використовуються як базові класифікатори для більш складних методів, таких як випадковий ліс [17].

3. Випадковий ліс – ансамблевий метод, що базується на злитті кількох дерев рішень для покращення стійкості та точності прогнозування. Ефективний у випадках великої кількості функцій і здатний добре обробляти перепідгонку (overfitting) [18].

4. Метод опорних векторів (SVM) – ефективний класифікаційний алгоритм, який знаходить гіперплощину в багатовимірному просторі, що найкраще відділяє два класи. Часто використовується в складних класифікаційних задачах з високою роздільною здатністю [19].

5. Нейронні мережі – потужні моделі, які можуть застосовуватись для бінарної класифікації через їх здатність учення глибоких нелінійних структур. Ефективні в великих та складних наборах даних, особливо з зображеннями, звуком та текстом.

6. Градієнтний бустинг – ще один ансамблевий метод, який послідовно покращує слабкі класифікатори, зазвичай використовуючи короткі дерева рішень. Чудово підходить для різних типів даних і часто показує високу точність у змаганнях з аналізу даних [20].

Кожен із цих методів має свої переваги та недоліки, і вибір певного методу часто залежить від конкретної задачі, доступності даних, потреби в інтерпретації результатів та доступних обчислювальних ресурсів.

Для подальшої роботи виберемо моделі логістична регресія, випадковий ліс та градієнтний бустинг.

Імпортуємо відповідні бібліотеки в нашу програму (рис. 2.21).

```
from sklearn.linear_model import LogisticRegression
import xgboost as xgb
from sklearn.ensemble import RandomForestClassifier
```

Рисунок 2.21 – Імпорт бібліотек машинного навчання

Метрики для оцінки моделей класифікаторів:

- accuracy_score – обчислює точність класифікації.
- precision_score – обчислює точність прогнозу позитивних класів.
- recall_score – обчислює чутливість або кількість правильно класифікованих позитивних класів відносно всіх позитивних класів у даних.
- f1_score – обчислює середню гармоніку між точністю та чутливістю.
- roc_auc_score – обчислює площу під кривою ROC (Receiver Operating Characteristic).

Обрані 4 метрики оцінювання моделей класифікаторів імпортуємо в програму (рис. 2.22).

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
from sklearn.metrics import roc_auc_score
```

Рисунок 2.22 – Метрики оцінки моделей класифікаторів

Перед тренуванням моделей потрібно провести налаштування гіперпараметрів моделей навчання – це допоможе оптимізувати продуктивність моделі. Для автоматизації налаштування гіперпараметрів моделі використаємо функцію `GridSearchCV` з бібліотеки `sklearn.model_selection`. Для кожної моделі використовують свої параметри.

Налаштуємо параметри `XGBoost` (рис. 2.23). Основні гіперпараметри для тюнінгу моделі `XGBoost`:

- `max_depth` – максимальна глибина кожного дерева. занадто велике значення може призвести до перенавчання.
- `min_child_weight` – мінімальна сума ваги, необхідна в дочірньому вузлі. використовується для контролю перенавчання.
- `Gamma` – мінімальне зниження втрат, необхідне для роботи додаткового розбиття на дереві.
- `Subsample` – відсоток вибірки, використовуваної для кожного дерева. Низьке значення може призвести до недонавчання.
- `colsample_bytree` – відсоток функцій, використовуваних для кожного дерева. високий рівень може призвести до перенавчання.
- `learning_rate` – крок навчання, використовуваний в процесі оновлення для запобігання перенавчанню. зазвичай між 0.01 та 0.2.
- `n_estimators` – кількість дерев у ансамблі. треба балансувати з `learning_rate`.

```

import xgboost as xgb
from sklearn.metrics import accuracy_score

params = {
    'max_depth': [3, 6, 9],
    'min_child_weight': [1, 5, 10],
    'gamma': [0.5, 1, 1.5],
    'subsample': [0.6, 0.8, 1.0],
    'colsample_bytree': [0.6, 0.8, 1.0],
    'learning_rate': [0.01, 0.1, 0.2],
    'n_estimators': [100, 200, 300]
}

# Створення екземпляру класифікатора
model = xgb.XGBClassifier(use_label_encoder=False, eval_metric='logloss')
model_xgb = GridSearchCV(model, param_grid=params, scoring='accuracy', n_jobs=-1, cv=3, verbose=3)
model_xgb.fit(X_train, y_train)

```

Рисунок 2.23 – Модель XGBoost

Fitting 3 folds for each of 48 candidates, totalling 144 fits

```

GridSearchCV
GridSearchCV(cv=3,
              estimator=XGBClassifier(base_score=None, booster=None,
                                     callbacks=None, colsample_bylevel=None,
                                     colsample_bynode=None,
                                     colsample_bytree=None, device=None,
                                     early_stopping_rounds=None,
                                     enable_categorical=False,
                                     eval_metric='logloss', feature_types=None,
                                     gamma=None, grow_policy=None,
                                     importance_type=None, interaction_constraints=None,
                                     learning_rate=None, max_bin=None, max_cat_threshold=None,
                                     max_cat_to_onehot=None, max_delta_step=None, max_depth=None,
                                     max_leaves=None, min_child_weight=None, missing=nan,
                                     n_estimators=None, num_parallel_tree=None,
                                     random_state=None, silent=False, subsample=None,
                                     tree_method=None, validate_parameters=None,
                                     verbosity=None,
                                     **kwargs),
              param_grid={'max_depth': [3, 6, 9], 'min_child_weight': [1, 5, 10], 'gamma': [0.5, 1, 1.5],
                          'subsample': [0.6, 0.8, 1.0], 'colsample_bytree': [0.6, 0.8, 1.0],
                          'learning_rate': [0.01, 0.1, 0.2], 'n_estimators': [100, 200, 300]},
              scoring='accuracy',
              verbose=3)

estimator: XGBClassifier
XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None, early_stopping_rounds=None,
              enable_categorical=False, eval_metric='logloss',
              feature_types=None, gamma=None, grow_policy=None,
              importance_type=None, interaction_constraints=None,
              learning_rate=None, max_bin=None, max_cat_threshold=None,
              max_cat_to_onehot=None, max_delta_step=None, max_depth=None,
              max_leaves=None, min_child_weight=None, missing=nan,
              n_estimators=None, num_parallel_tree=None,
              random_state=None, silent=False, subsample=None,
              tree_method=None, validate_parameters=None,
              verbosity=None,
              **kwargs)

XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None, early_stopping_rounds=None,
              enable_categorical=False, eval_metric='logloss',
              feature_types=None, gamma=None, grow_policy=None,
              importance_type=None, interaction_constraints=None,
              learning_rate=None, max_bin=None, max_cat_threshold=None,
              max_cat_to_onehot=None, max_delta_step=None, max_depth=None,
              max_leaves=None, min_child_weight=None, missing=nan,
              n_estimators=None, num_parallel_tree=None,
              random_state=None, silent=False, subsample=None,
              tree_method=None, validate_parameters=None,
              verbosity=None,
              **kwargs)

```

Рисунок 2.24 – Найкращі параметри моделі XGBoost

```
y_pred_train = model_xgb.predict(X_train1)
accuracy_train = accuracy_score(y_pred_train, y_train1)
print("Точність xgboost на тренувальних даних : ", (accuracy_train ))
```

```
Точність xgboost на тренувальних даних : 0.7994823900078767
```

Рисунок 2.25 – Точність моделі на тренувальних даних.

Тюнінг моделі логістичної регресії зазвичай полягає у виборі оптимальних значень для декількох ключових гіперпараметрів. Для LogisticRegression у бібліотеці scikit-learn найважливішими параметрами для тюнінгу є:

- C – інверсна сила регуляризації. Менші значення вказують на сильнішу регуляризцію.
- penalty – використаний тип регуляризації (наприклад, 'l1', 'l2', 'elasticnet').
- solver – алгоритм, який використовується для оптимізації. деякі алгоритми підтримують лише певні типи штрафів.
- max_iter – максимальна кількість ітерацій, яка використовується для збігу до оптимального рішення (рис. 2.26).

```
# налаштування параметрів
param_grid = {
    'C': [0.01, 0.1, 1, 10, 100],
    'penalty': ['l2'], # 'l1', 'elasticnet' можливі, але потребують специфічних солверів
    'solver': ['lbfgs', 'newton-cg', 'liblinear'], # 'saga' підходить якщо потрібен 'l1' або 'elasticnet'
    'max_iter': [100, 200, 300]
}
logistic_model = LogisticRegression()
g_logistic_model = GridSearchCV(model, param_grid=param_grid, scoring='accuracy', cv=3, verbose=1)
g_logistic_model.fit(X_train, y_train)
```

Рисунок 2.26 – Модель логістична регресія

На рисунку 2.27 показана точність моделі на тренувальних даних.

```

y_pred_logistic_t = best_model.predict(X_train1)
print("Точність логістичної регресії тренування:", accuracy_score(y_train1, y_pred_logistic_t))

```

Точність логістичної регресії тренування: 0.8071340159783954

Рисунок 2.27 – Точність на тренувальних даних.

Основні гіперпараметри, які зазвичай налаштовуються для RandomForestClassifier:

- `n_estimators` – кількість дерев у лісі. зазвичай, чим більше дерев, тим краще модель може вчитися, але це також збільшує обчислювальні витрати.
- `max_features` – кількість ознак, які потрібно враховувати при пошуку найкращого поділу; зазвичай 'auto', 'sqrt', 'log2'.
- `max_depth` – максимальна глибина кожного дерева. Більша глибина може збільшити ризик перенавчання.
- `min_samples_split` – мінімальна кількість вибірок, необхідних для розгалуження внутрішнього вузла.
- `min_samples_leaf` – мінімальна кількість вибірок, які мають бути в листовому вузлі. Встановлення цього параметра допомагає уникнути перенавчання.
- `Bootstrap` – вказує, чи використовувати бутстрап-вибірки для побудови дерев. Значення за замовчуванням — True, що дозволяє збільшити різноманітність дерев.

На рисунках 2.28 та 2.29 показано налаштовану модель та точність її роботи.

```

forest_model = RandomForestClassifier()
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_features': [1.0, 'sqrt', 'log2'],
    'max_depth': [None, 10],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2],
    'bootstrap': [True, False]
}

g_forest_model = GridSearchCV(estimator=forest_model, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2, scoring='accuracy')

# Пошук найкращих гіперпараметрів
g_forest_model.fit(X_train, y_train)

```

Рисунок 2.28 – Модель випадковий ліс

```

best_rf = g_forest_model.best_estimator_
y_pred_rf_t = best_rf.predict(X_train1)
print("Точність випадковий ліс тренування:", accuracy_score(y_train1, y_pred_rf_t))

```

```

Fitting 3 folds for each of 4 candidates, totalling 12 fits
Точність випадковий ліс тренування: 0.8832001800382582

```

Рисунок 2.29 – Точність моделі випадковий ліс на тренувальних даних.

2.4 Висновки

У другому розділі проведений розвідувальний аналіз даних для передбачення поведінки клієнтів під час онлайн бронювання номерів у готелі. Визначено основну ознаку, яку потрібно передбачати, а саме `booking_status` - ознака, яка показує відміняли бронь чи ні. Також визначені ознаки, що характеризують її. Проведено налаштування основних параметрів моделей-класифікаторів.

3. РЕЗУЛЬТАТИ ПЕРЕДБАЧЕННЯ БРОНЮВАННЯ ГОТЕЛІВ КЛІЄНТАМИ

3.1 Передбачення поведінки клієнтів при онлайн бронюванні готелів

За допомогою натренованих моделей проведемо передбачення поведінки клієнтів при онлайн бронюванні готелів.

Проведемо передбачення натестових даних моделлю XGBoost, що зображено на рисунку 3.1.

```
|:
# Зробіть прогнози
y_pred_test = model_xgb.predict(X_test)
# Оцінка точності моделі
accuracy = accuracy_score(y_test, y_pred_test)
print("Точність xgboost TEST : ", (accuracy ))
```

Точність xgboost TEST : 0.7922447854451897

Рисунок 3.1 – Передбачення моделлю XGBoost на тестових даних.

Для візуалізації результатів передбачення використаємо матрицю плутанини. На рисунку 3.2 зображений код матрицю плутанини для моделі XGBoost на тестових даних.

```
# Обчислення матриці плутанини
cm = confusion_matrix(y_test, y_pred_test)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt="d", cmap='Blues')
plt.title('Матриця плутанини')
plt.xlabel('Передбачені дані')
plt.ylabel('Тестові дані')
plt.show()
```

Рисунок 3.2 – Код матриці плутанини

На рисунку 3.3 зображено результат роботи коду з рисунку 3.2.

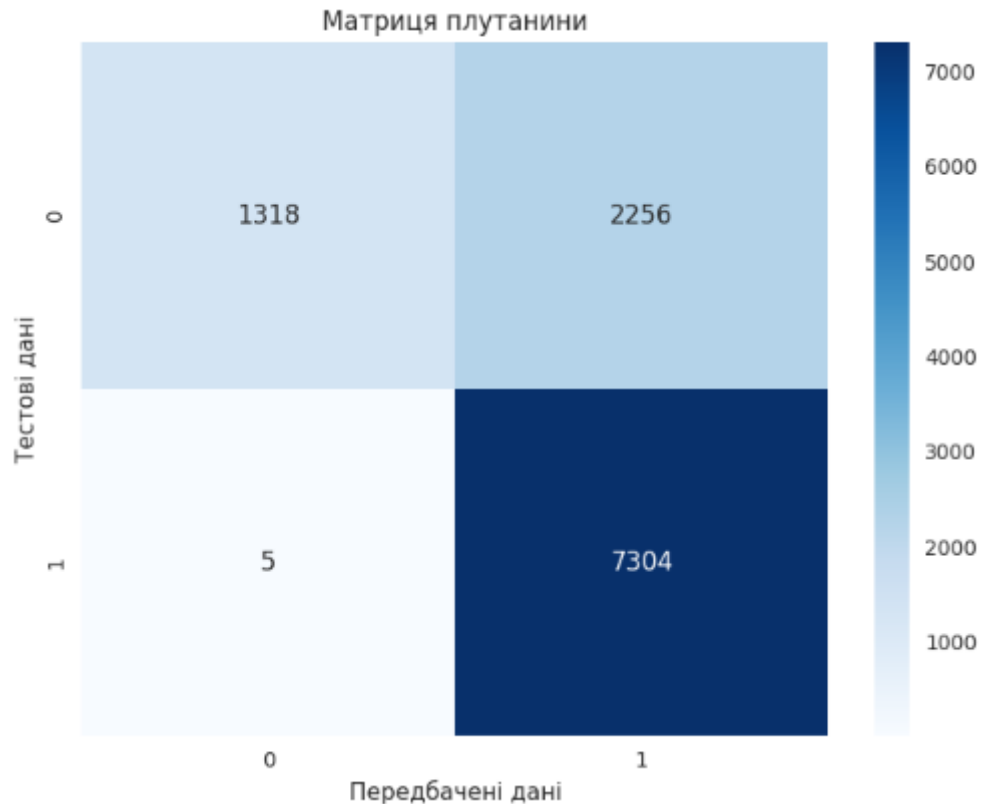


Рисунок 3.3 – Матриця плутанини результатів моделі XGBoost

З цієї матриці можна зробити кілька висновків про продуктивність моделі:

- True Negative (TN) – 1318, це кількість випадків, коли модель правильно передбачила негативний клас.
- False Positive (FP) – 2256, це ситуації, коли модель невірно класифікувала негативні випадки як позитивні. Висока кількість таких помилок може свідчити про недостатню специфічність моделі.
- False Negative (FN) – 5, це кількість випадків, коли модель невірно класифікувала позитивні випадки як негативні. Мінімальна кількість таких помилок вказує на високу чутливість моделі.
- True Positive (TP) – 7304, це кількість випадків, коли модель правильно ідентифікувала позитивний клас.

Проведемо передбачення на тестових даних логістичною регресією, що зображено на рисунку 3.4.

```
y_pred_logistic = best_model.predict(X_test)
print("Точність логістичної регресії тест:", accuracy_score(y_test, y_pred_logistic))
```

Точність логістичної регресії тест: 0.8045575668473767

Рисунок 3.4 – Передбачення моделлю логістична регресія на тестових даних.

На рисунку 3.5 зображений код матриці плутанини для моделі логістичної регресії.

```
cm1 = confusion_matrix(y_test, y_pred_logistic)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm1, annot=True, fmt="d", cmap='Blues')
plt.title('Матриця плутанини')
plt.xlabel('Передбачені дані')
plt.ylabel('Тестові дані')
plt.show()
```

Рисунок 3.5 – Код матриці плутанини для моделі логістичної регресії

На рисунку 3.6 зображено матрицю плутанини для моделі логістичної регресії для передбачення поведінки клієнтів під час онлайн бронювання номерів готелів.

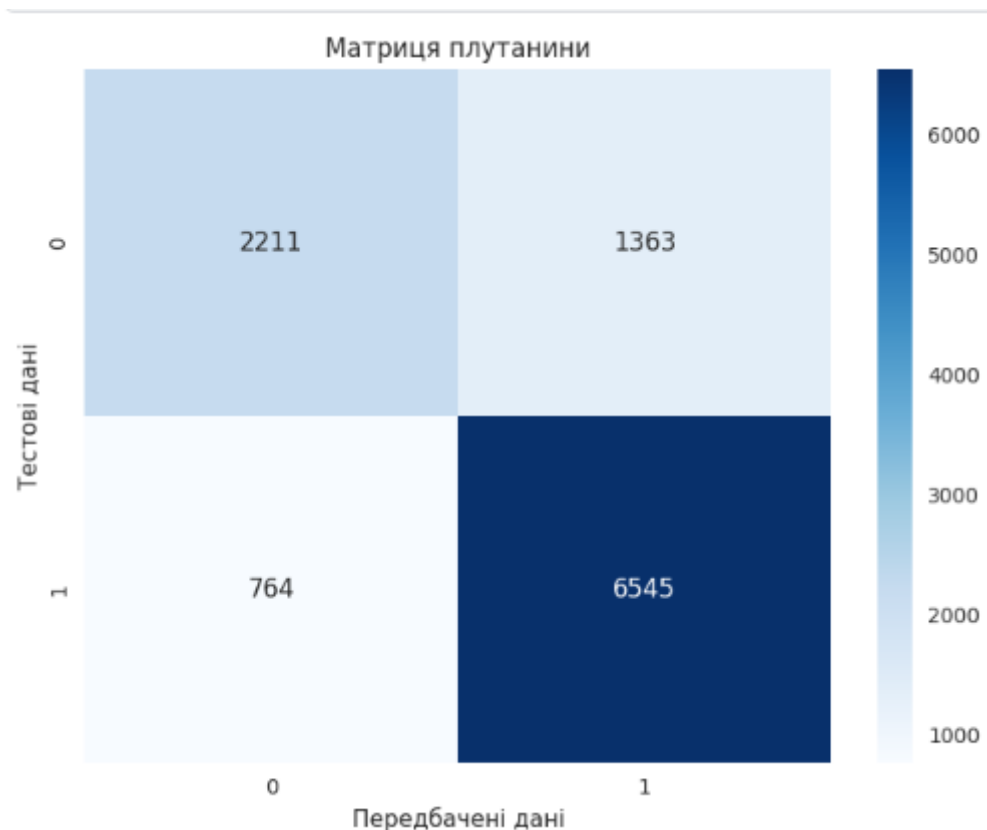


Рисунок 3.6 – Передбачення моделлю логістичної регресії.

З цієї матриці можна зробити кілька висновків про продуктивність моделі:

- True Negatives (TN) – 2211. Кількість випадків, коли модель правильно передбачила негативний клас.
- False Positives (FP) – 1363. Це ситуації, коли модель невірно класифікувала негативні випадки як позитивні.
- False Negatives (FN) – 764. Кількість випадків, коли модель невірно класифікувала позитивні випадки як негативні.
- True Positives (TP) – 6545. Кількість випадків, коли модель правильно ідентифікувала позитивний клас.

Матриця плутанини демонструє хорошу здатність моделі ідентифікувати позитивні випадки (високий TP), але також показує, що модель має певні проблеми з визначенням негативних класів, з досить високим числом FP. Залежно від контексту застосування моделі, може бути необхідно зменшити

кількість помилково позитивних результатів для підвищення загальної продуктивності моделі.

Проведемо передбачення на тестових даних випадковим лісом, що зображено на рисунку 3.7.

```
|: y_pred_forest = best_rf.predict(X_test)
   print("Точність випадкового лісу тест:", round(accuracy_score(y_test, y_pred_forest),2))
```

Точність випадкового лісу тест: 0.88

Рисунок 3.7 – Передбачення на тестових даних випадковим лісом.

На рисунку 3.8 зображено код матриці плутанини для моделі випадковий ліс.

```
cm2 = confusion_matrix(y_test, y_pred_forest)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm2, annot=True, fmt="d", cmap='Blues')
plt.title('Матриця плутанини')
plt.xlabel('Передбачені дані')
plt.ylabel('Тестові дані')
plt.show()
```

Рисунок 3.8 – Код матриці плутанини для моделі випадковий ліс

На рисунку 3.9 відображено результат роботи коду для побудови матриці плутанини результатів передбачення моделі випадкового лісу поведінки клієнтів під час онлайн бронювання готелів.

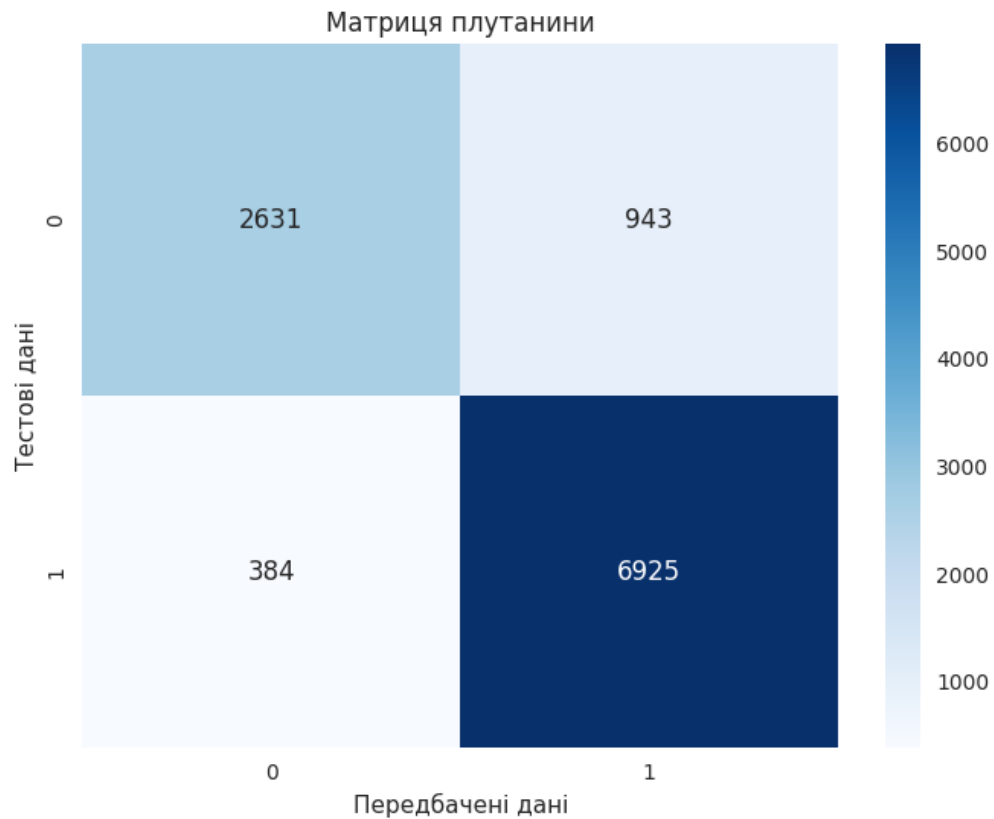


Рисунок 3.9 – Результат роботи коду для побудови матриці плутанини результатів передбачення моделі випадкового лісу

З цієї матриці можна зробити кілька висновків про продуктивність моделі:

- True Negatives (TN) – 2643. Кількість випадків, коли модель правильно ідентифікувала негативні випадки.
- False Positives (FP) – 931. Кількість випадків, коли негативні випадки були неправильно класифіковані як позитивні.
- False Negatives (FN) – 403. Кількість випадків, коли позитивні випадки були неправильно класифіковані як негативні.
- True Positives (TP) – 6906. Кількість випадків, коли модель правильно ідентифікувала позитивні випадки.

На основі цих показників можна зробити висновки про ефективність моделі. Високий рівень TP та порівняно низький FN вказують на гарну

чутливість моделі. Проте, відносно велика кількість FP може бути проблемою, особливо якщо важливо мінімізувати помилково позитивні результати.

Застосувавши метрику оцінки `accuracy_score` отримано такі дані:

	Model	Точність на тренувальних даних	Точність на тестових даних
0	LogisticRegression	0.81	0.80
1	XGBClassifier	0.80	0.79
2	RandomForestClassifier	0.88	0.88

Рисунок 3.10 - Результати передбачення усіма моделями.

3.2 Висновки

Для вибору оптимальної моделі машинного навчання для передбачення поведінки клієнтів під час онлайн бронювання готелю використаємо стандартну метрику оцінки моделей-класифікаторів, а саме `accuracy_score`.

На основі проведеного тестування моделей машинного навчання обрано оптимальну модель, а саме випадковий ліс, з точністю на тестових даних у розмірі 0,88.

ВИСНОВКИ

В першому розділі проаналізовано сервіси бронювання готелів та їх специфікацію. Визначені основні інформаційні технології, які використовують готелі та сервіси резервування готельних номерів.

Проведений аналіз методів передбачення поведінки клієнтів при онлайн бронюванні готелів. Обрано клас методів машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів.

У другому розділі проведено розвідувальний аналіз даних для задачі передбачення онлайн бронювання готелів. Здійснено вибір оптимальних технологій машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів. Відібрано основну ознаку, яку потрібно передбачати, а саме `booking_status`. Також визначені ознаки, що характеризують її. Проведено налаштування основних параметрів моделей-класифікаторів та тренування моделей.

Для вибору оптимальної моделі машинного навчання для передбачення поведінки клієнтів під час онлайн бронювання готелю використано стандартну метрику оцінки моделей-класифікаторів, а саме `accuracy_score`.

У третьому розділі застосовано обрані моделі машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів та передбачено поведінку клієнтів при онлайн бронюванні готелів. На основі проведеного тестування моделей машинного навчання обрано оптимальну модель, а саме випадковий ліс, з точністю на тестових даних у розмірі 0,88.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Amadeus, "Amadeus IT Group SA," Accessed: May 22, 2024. URL: <https://amadeus.com/en>
2. Sabre, "Sabre – An innovative technology company," Accessed: May 22, 2024. URL: <https://www.sabre.com/>
3. Amadeus Hospitality, "Amadeus Hospitality," Accessed: May 22, 2024. URL: <https://www.amadeus-hospitality.com/>
4. Booking.com, "Booking.com: The largest selection of hotels, homes, and vacation rentals," Accessed: May 22, 2024. URL: <https://www.booking.com>
5. Microsoft, "Dynamics 365—Intelligent Business Applications," Accessed: May 22, 2024. URL: <https://www.microsoft.com/en/dynamics-365> Kimes, S. E., & Singh, S. Predictive analytics in hotel revenue management: An empirical study of adoption in practice. *Cornell Hospitality Quarterly*, 62(1), 89-101, 2021
6. Kaggle Dataset „Hotel Reservations Dataset” [Електронний ресурс]. – Режим доступу: <https://www.kaggle.com/datasets/ahsan81/hotel-reservations-classification-dataset>.
7. М. В. . Дратований, О. М. Козачко, О. Л. . Мельник, і І. В. . Варчук, «Інформаційна технологія оптимізації параметрів ансамблю моделей штучного інтелекту для прогнозування наявності опадів за даними метеомоніторингу», Вісник ВПІ, вип. 6, с. 76–83, Груд. 2020.
8. В. Б. Мокін, А. В. Лосенко, і М. В. Дратований, «Інтелектуальна технологія аналізу та передбачення цін на вживані автомобілі», Вісник ВПІ, вип. 6, с. 62–72, Груд. 2019.
9. Nguyen, N., Nguyen, T. H., Nguyen, Y. N., Doan, D., Nguyen, M., & Nguyen, V. H. Machine learning-based model for customer emotion detection in hotel booking services. *Journal of Hospitality and Tourism Insights*. 2023. URL :<https://doi.org/10.1108/JHTI-03-2023-0166>
10. Chen, S., Ngai, E. W., Ku, Y., Xu, Z., Gou, X., & Zhang, C. Prediction of hotel booking cancellations: Integration of machine learning and probability model

based on interpretable feature interaction. *Decision Support Systems*, 170, 113959, 2023. URL : <https://doi.org/10.1016/j.dss.2023.113959>

11. Mendieta-Aragón, A., & Garín-Muñoz, T. Consumer behaviour in e-Tourism: Exploring new applications of machine learning in tourism studies. *Investigaciones Turísticas*, (26), 350-374, 2023. URL : <https://doi.org/10.14198/INTURI.24629>

12. Hamdan, I. Z. P., Othman, M., Hassim, Y. M. M., Marjudi, S., & Yusof, M. M. Customer Loyalty Prediction for Hotel Industry Using Machine Learning Approach. *JOIV: International Journal on Informatics Visualization*, 7(3), 695-703, 2023. URL : <http://dx.doi.org/10.30630/joiv.7.3.1335>

13. Qureshi, S., & Menezes, J. Prediction of hotel booking cancellation using machine learning algorithms. In *7th IET Smart Cities Symposium (SCS 2023)* (Vol. 2023, pp. 140-145), 2023. IET. URL : <https://doi.org/10.1049/icp.2024.0914>

14. Z. Zhao, "An exploratory analysis on users' contributions in federated learning," «2022 IEEE International Conference on Big Data (Big Data)», pp. 5855-5864, 2022. URL : <https://arxiv.org/pdf/2011.06830>

15. Nie, T. Machine Learning for Forecasting Future Reservations' Ratings: Radisson Blu Seaside in Helsinki. 2023. URL: <https://urn.fi/URN:NBN:fi-fe2023050340807>

16. Vijayarangan, N., Rashmi, K., & Sakthi Priya, M. Innovative Design Platform on Hotel Industry 4.0 Using AI/ML. In *International Conference on Research into Design* (pp. 1167-1181). Singapore: Springer Nature Singapore, 2023. URL : https://doi.org/10.1007/978-981-99-0293-4_94

17. Y. Wang, "Towards high-performance exploratory data analysis (EDA) via stable equilibrium point," «International Conference on Neural Information Processing», pp. 483-492, 2023. URL : <https://arxiv.org/pdf/2306.04425>

18. Kalnaovakul, K., & Promsivapallop, P. Hotel service quality dimensions and attributes: An analysis of online hotel customer reviews. *Tourism and Hospitality Research*, 23(3), 420-440, 2023. URL : <https://doi.org/10.1177/14673584221145819>

19. Мокін, В. Б., Дратований, М. В. Наука про дані: машинне навчання та інтелектуальний аналіз даних: електронний навчальний посібник комбінованого (локального та мережевого) використання. Вінниця: ВНТУ. 258 с. ,2024. URL: <https://docs.vntu.edu.ua/card.php?id=8163>

20. Singh, H. P., Alshallaqi, M., & Altamimi, M. Predicting Critical Factors Impacting Hotel Online Ratings: A Comparison of Religious and Commercial Destinations in Saudi Arabia. *Sustainability*, 2023. 15(15), 11998. URL: <https://doi.org/10.3390/su151511998>

Додаток А
(обов'язковий)

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ

Завідувач кафедри САІТ

_____ д.т.н., проф. Віталій МОКІН

« ____ » _____ 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ
на бакалаврську дипломну роботу
СИСТЕМНИЙ АНАЛІЗ ПОВЕДІНКИ КЛІЄНТІВ ПРИ ОНЛАЙН
БРОНЮВАННІ ГОТЕЛІВ
08-34.БДР.003.00.000 ТЗ

Керівник: к.т.н., доц.

_____ Олексій КОЗАЧКО

« ____ » _____ 2024 р.

Розробила студентка гр. СА-206

_____ Марина ГРИГОРЧУК

« ____ » _____ 2024 р.

Вінниця 2024

1. Підстава для проведення робіт.

Підставою для виконання роботи є наказ №__ по ВНТУ від «__» _____2024р., та індивідуальне завдання на БДР, затверджене протоколом №__ засідання кафедри САІТ від «__» _____ 2024р.

2. Джерела розробки:

1) Мокін, В. Б., Дратованій, М. В. (2024). Наука про дані: машинне навчання та інтелектуальний аналіз даних: електронний навчальний посібник комбінованого (локального та мережевого) використання. Вінниця: ВНТУ. 258 с. URL: <https://docs.vntu.edu.ua/card.php?id=8163>

2) Microsoft, "Dynamics 365—Intelligent Business Applications," Accessed: May 22, 2024. URL: <https://www.microsoft.com/en/dynamics-365> Kimes, S. E., & Singh, S. (2021). Predictive analytics in hotel revenue management: An empirical study of adoption in practice. Cornell Hospitality Quarterly, 62(1), 89-101.

2. Мета і призначення роботи.

Метою дослідження є підвищення точності передбачення поведінки клієнтів при онлайн бронюванні готелів за допомогою системного аналізу.

3. Вихідні дані для проведення робіт:

Kaggle Dataset «Hotel Reservations Dataset»
<https://www.kaggle.com/datasets/ahsan81/hotel-reservations-classification-dataset>.

4. Методи дослідження:

Моделі машинного навчання, мова програмування Python.

5. Етапи роботи і терміни їх виконання:

- | | | | |
|--|-------|---|-------|
| a) Аналіз предметної області | _____ | — | _____ |
| b) Порівняльний аналіз проблематики | _____ | — | _____ |
| c) Вибір оптимальних інформаційних технологій | _____ | — | _____ |
| d) Передбачення поведінки клієнтів при онлайн бронюванні готелів | _____ | — | _____ |
| e) Оформлення матеріалів до захисту БДР | _____ | — | _____ |

7. Очікувані результати та порядок реалізації

Отримання оптимальної моделі машинного навчання для передбачення поведінки клієнтів при онлайн бронюванні готелів.

8. Вимоги до розробленої документації

Текстова та ілюстративна частини роботи оформлені у відповідності до вимог «Методичних вказівок до виконання бакалаврських кваліфікаційних робіт для студентів спеціальностей: 124 «Системний аналіз», 126 «Інформаційні системи та технології» (освітня програма «Прикладні інформаційні технології»).

9. Порядок приймання роботи

Публічний захист «__» _____ 2024 р.

Початок розробки «__» _____ 2024 р.

Граничні терміни виконання БДР «__» _____ 2024 р.

Розробила студентка групи СА-206 _____ Марина ГРИГОРЧУК

Додаток Б
(обов'язковий)

Протокол перевірки бакалаврської дипломної роботи на наявність текстових
запозичень

Назва роботи: «Системний аналіз поведінки клієнтів при онлайн бронюванні готелів»

Тип роботи: бакалаврська дипломна робота

Підрозділ: кафедра САІТ

Показники звіту подібності Unicheck

Оригінальність 97%

Схожість 3%

Аналіз звіту подібності (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку


(підпис)

Сергій ЖУКОВ

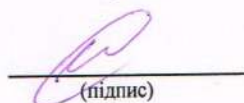
Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи


(підпис)

Марина ГРИГОРЧУК

Керівник роботи


(підпис)

Олексій КОЗАЧКО

Додаток В

(ДОВІДНИКОВИЙ)

Фрагмент лістингу програми

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from yellowbrick.classifier import ConfusionMatrix
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import r2_score
from sklearn.metrics import mean_absolute_error, mean_squared_error
from sklearn.model_selection import GridSearchCV

import warnings
warnings.filterwarnings('ignore')

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
df = pd.read_csv('/kaggle/input/hotel-reservations-classification-dataset/Hotel Reservations.csv')
df
df.info()
df.describe()
df = df.drop(['Booking_ID'], axis=1)
# Виділення числових ознак
numeric_columns = df.select_dtypes(include=['int64', 'float64']).columns

# Виділення категоріальних ознак
categorical_columns = df.select_dtypes(include=['object', 'category']).columns

# Створення окремих DataFrame для кожного типу
```

```
df_numeric = df[numeric_columns]
df_categorical = df[categorical_columns]

correlation = df_numeric.corr().round(2)
plt.figure(figsize = (14,7))
sns.heatmap(correlation, annot = True, cmap = 'YlOrBr')
plt.figure(figsize = (20,25))

plt.subplot(4,2,1)
plt.gca().set_title('Variable no_of_adults')
sns.countplot(x = 'no_of_adults', palette = 'Set2', data = df)

plt.subplot(4,2,2)
plt.gca().set_title('Variable no_of_children')
sns.countplot(x = 'no_of_children', palette = 'Set2', data = df)

plt.subplot(4,2,3)
plt.gca().set_title('Variable no_of_weekend_nights')
sns.countplot(x = 'no_of_weekend_nights', palette = 'Set2', data = df)

plt.subplot(4,2,4)
plt.gca().set_title('Variable no_of_week_nights')
sns.countplot(x = 'no_of_week_nights', palette = 'Set2', data = df)

plt.subplot(4,2,5)
plt.gca().set_title('Variable type_of_meal_plan')
sns.countplot(x = 'type_of_meal_plan', palette = 'Set2', data = df)

plt.subplot(4,2,6)
plt.gca().set_title('Variable required_car_parking_space')
sns.countplot(x = 'required_car_parking_space', palette = 'Set2', data = df)

plt.subplot(4,2,7)
plt.gca().set_title('Variable room_type_reserved')
sns.countplot(x = 'room_type_reserved', palette = 'Set2', data = df)
```

```

plt.subplot(4,2,8)
plt.gca().set_title('Variable arrival_year')
sns.countplot(x = 'arrival_year', palette = 'Set2', data = df)

from sklearn.preprocessing import LabelEncoder

label_encoder_type_of_meal_plan = LabelEncoder()
label_encoder_room_type_reserved = LabelEncoder()
label_encoder_market_segment_type = LabelEncoder()
label_encoder_booking_status = LabelEncoder()

df['type_of_meal_plan'] = label_encoder_type_of_meal_plan.fit_transform(df['type_of_meal_plan'])
df['room_type_reserved'] = label_encoder_room_type_reserved.fit_transform(df['room_type_reserved'])
df['market_segment_type'] = label_encoder_market_segment_type.fit_transform(df['market_segment_type'])
df['booking_status'] = label_encoder_booking_status.fit_transform(df['booking_status'])

le_name_mapping = dict(zip(label_encoder_booking_status.classes_,
label_encoder_booking_status.transform(label_encoder_booking_status.classes_)))

print(le_name_mapping)

X = df.drop('booking_status', axis = 1)

#X = X.values

y = df['booking_status']

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

X_standard = scaler.fit_transform(X)

from sklearn.model_selection import train_test_split

X_train,X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0)

X_train1,X_var, y_train1, y_var = train_test_split(X_train, y_train, test_size = 0.3, random_state = 0)

from sklearn.linear_model import LogisticRegression

import xgboost as xgb

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy_score

from sklearn.model_selection import GridSearchCV

import xgboost as xgb

from sklearn.metrics import accuracy_score

```

```

params = {
    'max_depth': [3, 6, 9],
    'min_child_weight': [1, 5],
    'gamma': [0.5, 1],
    'subsample': [0.6, 0.8],
    'colsample_bytree': [0.6, 0.8],
    'learning_rate': [0.01],
    'n_estimators': [50]
}

# Створення екземпляру класифікатора
model_x = xgb.XGBClassifier(use_label_encoder=False, eval_metric='logloss')
model_xgb = GridSearchCV(model_x, param_grid=params, scoring='accuracy', n_jobs=-1, cv=3,
verbose=3)
model_x.fit(X_train1, y_train1)
model_xgb.fit(X_train1, y_train1)
y_pred_train = model_xgb.predict(X_train1)
accuracy_train = accuracy_score(y_pred_train, y_train1)
print("Точність xgboost на тренувальних даних : ", (accuracy_train ))

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score, log_loss

# Обчислення метрик
accuracy = accuracy_score(y_pred_train, y_train1)
precision = precision_score(y_pred_train, y_train1)
recall = recall_score(y_pred_train, y_train1)
f1 = f1_score(y_pred_train, y_train1)

# Вивід результатів
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)

# Зробіть прогнози
y_pred_test = model_xgb.predict(X_test)

```

```

# Оцінка точності моделі
accuracy = accuracy_score(y_test, y_pred_test)
print("Точність xgboost TEST : ", (accuracy ))
accuracies = [accuracy_var, accuracy]
labels = ['Train Accuracy', 'Test Accuracy']
from sklearn.metrics import confusion_matrix
import seaborn as sns

# Обчислення матриці плутанини
cm = confusion_matrix(y_test, y_pred_test)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt="d", cmap='Blues')
plt.title('Матриця плутанини')
plt.xlabel('Передбачені дані')
plt.ylabel('Тестові дані')
plt.show()

# налаштування параметрів
param_grid = {
    'C': [0.01, 0.1, 1],
    'penalty': ['l2'], # 'l1', 'elasticnet' можливі, але потребують специфічних солверів
    'solver': ['lbfgs', 'newton-cg', 'liblinear'], # 'saga' підходить якщо потрібен 'l1' або 'elasticnet'
    'max_iter': [100]
}

logistic_model = LogisticRegression()
g_logistic_model = GridSearchCV(logistic_model, param_grid=param_grid, scoring='accuracy', cv=3,
verbose=1)
g_logistic_model.fit(X_train1, y_train1)

best_model = g_logistic_model.best_estimator_

y_pred_logistic_t = best_model.predict(X_train1)
print("Точність логістичної регресії тренування:", accuracy_score(y_train1, y_pred_logistic_t))

```

```

y_pred_logistic = best_model.predict(X_test)
print("Точність логістичної регресії:", accuracy_score(y_test, y_pred_logistic))
cm1 = confusion_matrix(y_test, y_pred_logistic)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm1, annot=True, fmt="d", cmap='Blues')
plt.title('Матриця плутанини')
plt.xlabel('Передбачені дані')
plt.ylabel('Тестові дані')
plt.show()

# Ініціалізація моделі випадкового лісу
forest_model = RandomForestClassifier()
param_grid = {
    'n_estimators': [100],
    'max_features': ['sqrt'],
    'max_depth': [10],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2],
    'bootstrap': [True]
}

g_forest_model = GridSearchCV(estimator=forest_model, param_grid=param_grid, cv=3, n_jobs=-1,
verbose=2, scoring='accuracy')

# Пошук найкращих гіперпараметрів
g_forest_model.fit(X_train, y_train)

best_rf = g_forest_model.best_estimator_
y_pred_rf_t = best_rf.predict(X_train1)
print("Точність логістичної регресії тренування:", accuracy_score(y_train1, y_pred_rf_t))
y_pred_forest = best_rf.predict(X_test)
print("Точність випадкового лісу:", round(accuracy_score(y_test, y_pred_forest),2))
cm2 = confusion_matrix(y_test, y_pred_forest)

```

```
# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm1, annot=True, fmt="d", cmap='Blues')
plt.title('Матриця плутанини')
plt.xlabel('Передбачені дані')
plt.ylabel('Тестові дані')
plt.show()
```

Додаток Г
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

**СИСТЕМНИЙ АНАЛІЗ ПОВЕДІНКИ КЛІЄНТІВ ПРИ ОНЛАЙН
БРОНЮВАННІ ГОТЕЛІВ**

Нормоконтроль: к.т.н., доцент

_____ Сергій ЖУКОВ

«___» _____ 2024 р.

```

#      Column                                     Non-Null Count  Dtype
---  -
0      Booking_ID                               36275 non-null  object
1      no_of_adults                             36275 non-null  int64
2      no_of_children                           36275 non-null  int64
3      no_of_weekend_nights                     36275 non-null  int64
4      no_of_week_nights                       36275 non-null  int64
5      type_of_meal_plan                        36275 non-null  object
6      required_car_parking_space               36275 non-null  int64
7      room_type_reserved                       36275 non-null  object
8      lead_time                                36275 non-null  int64
9      arrival_year                             36275 non-null  int64
10     arrival_month                           36275 non-null  int64
11     arrival_date                             36275 non-null  int64
12     market_segment_type                     36275 non-null  object
13     repeated_guest                           36275 non-null  int64
14     no_of_previous_cancellations             36275 non-null  int64
15     no_of_previous_bookings_not_canceled     36275 non-null  int64
16     avg_price_per_room                       36275 non-null  float64
17     no_of_special_requests                   36275 non-null  int64
18     booking_status                           36275 non-null  object
dtypes: float64(1), int64(13), object(5)

```

Рисунок Г.1 – Інформація про дані

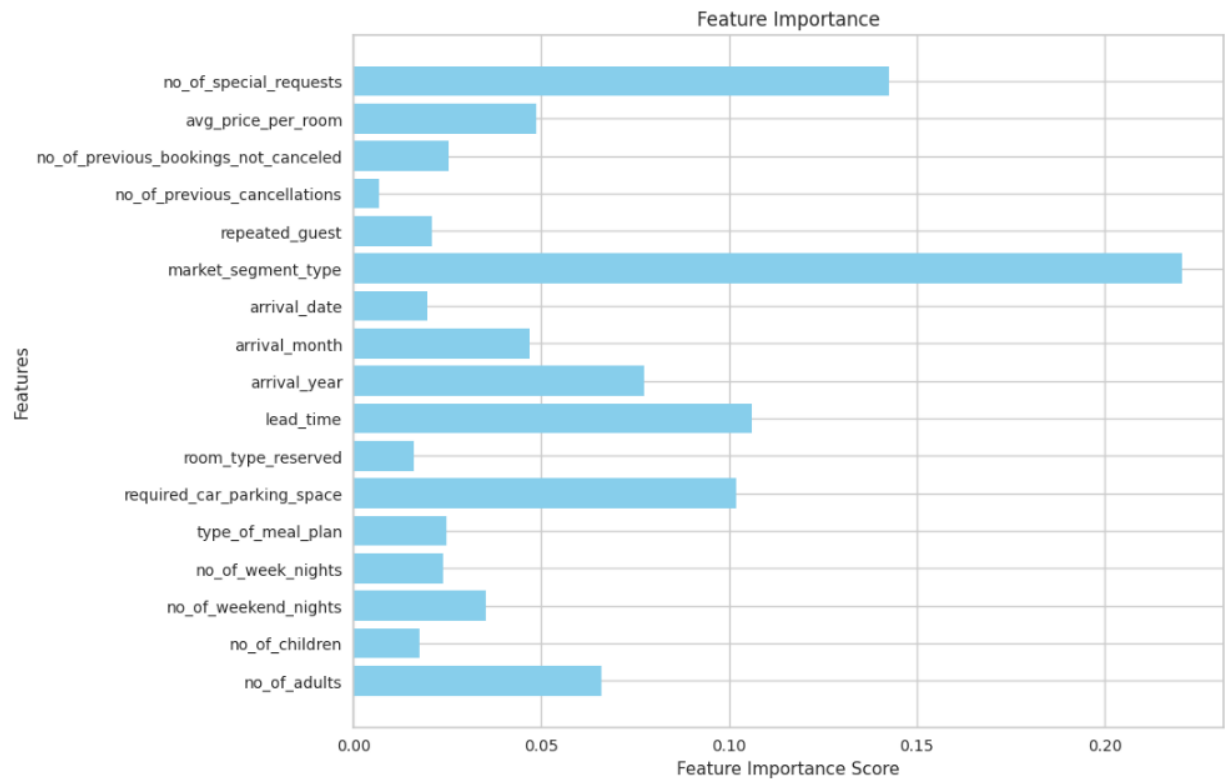


Рисунок Г.2 – Графік важливості ознак даних.

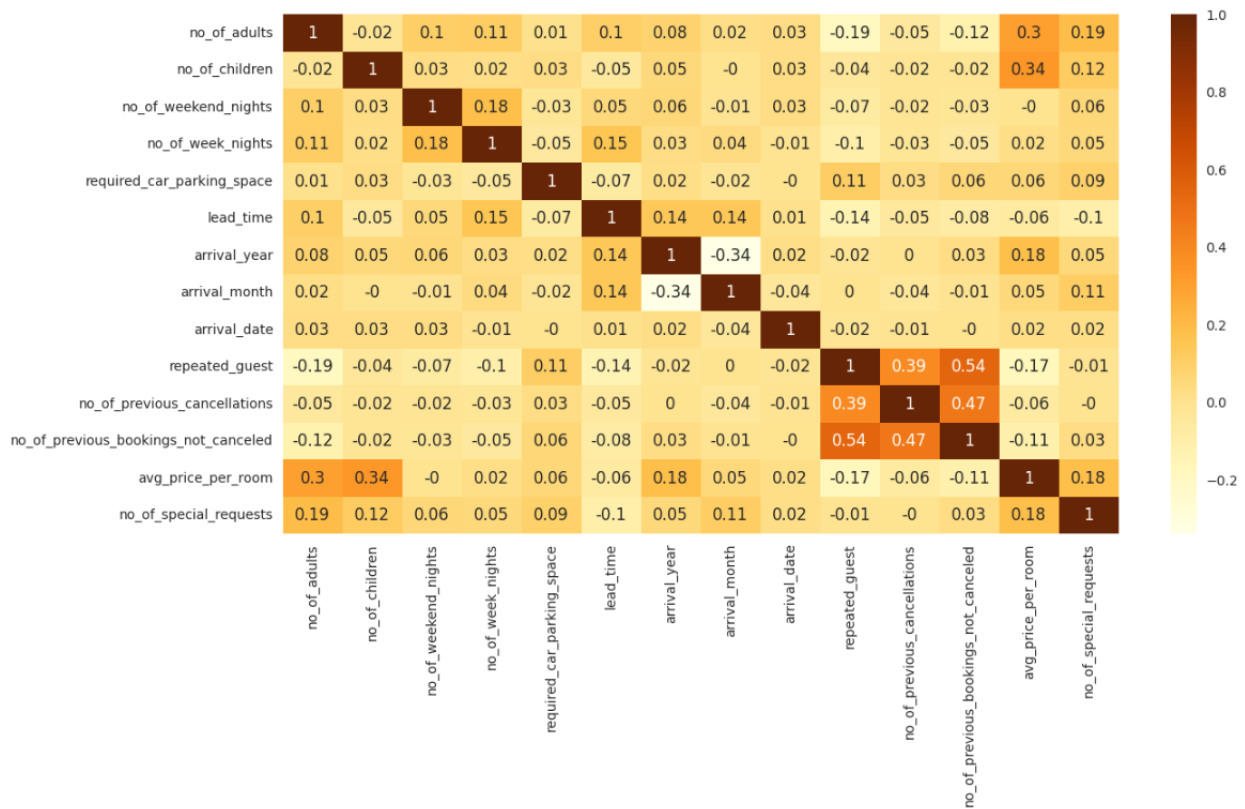


Рисунок Г.3 – Кореляційна матриця даних

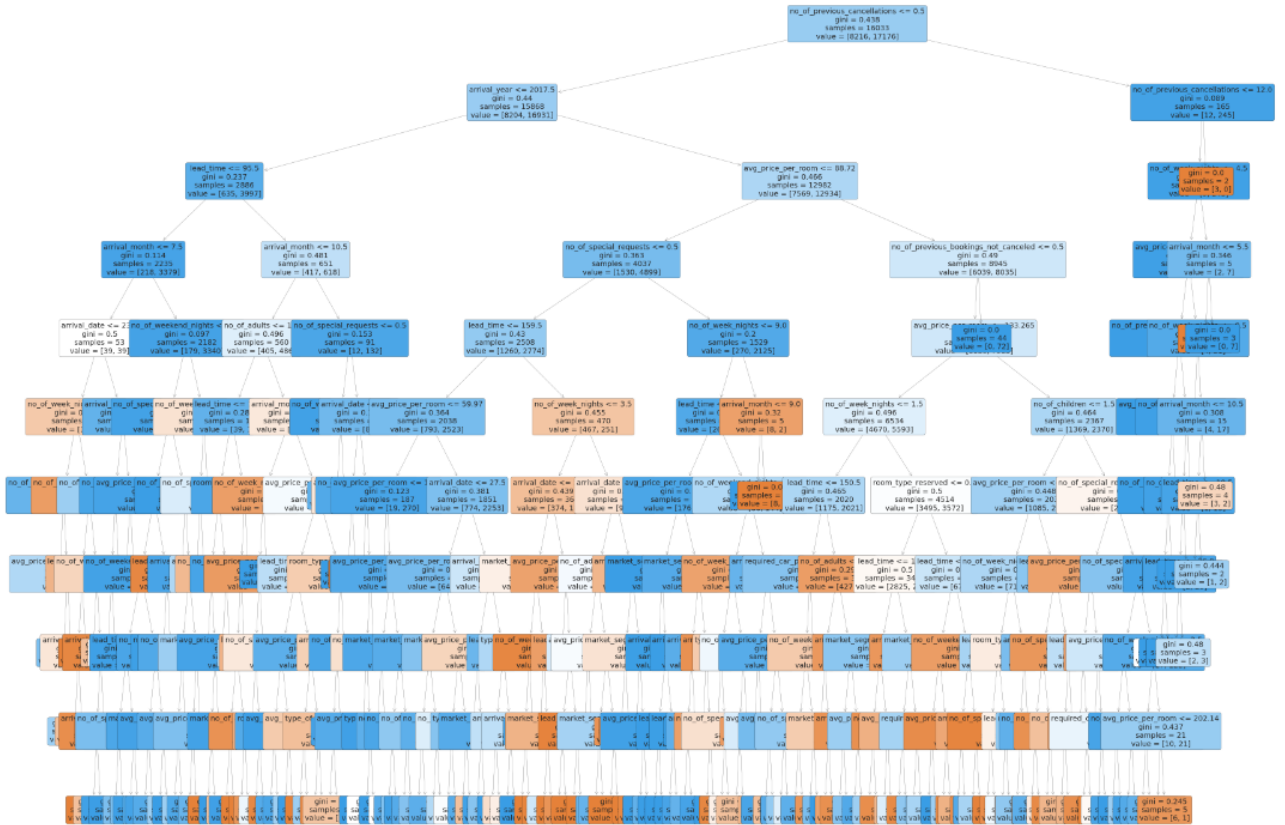


Рисунок Г.4 – Візуалізація роботи випадкового лісу

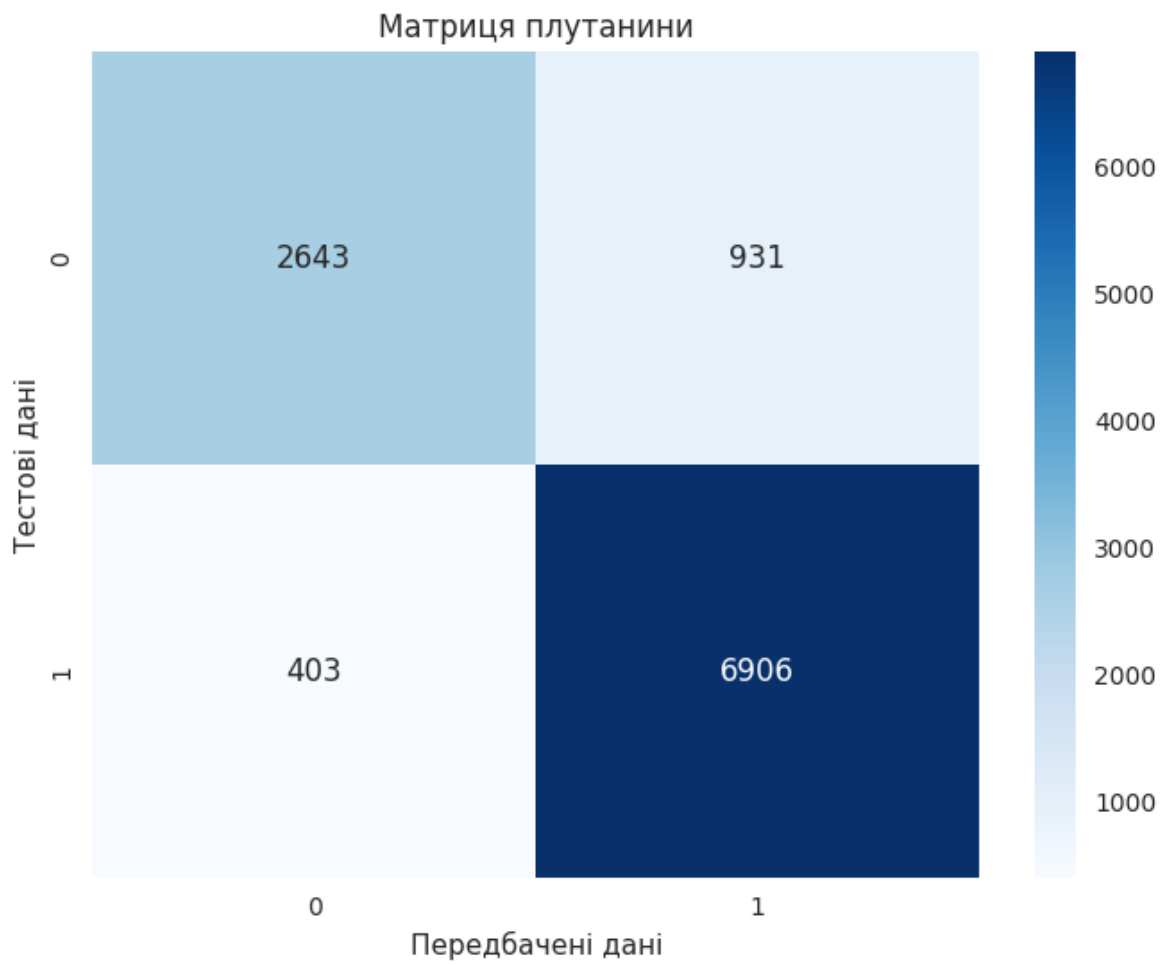


Рисунок Г.5 – Матриця плутанини результатів передбачення моделі випадкового лісу.