Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет Электроники и вычислительной техники   
Направление (специальность) Информатика и вычислительная техника . .  
Кафедра Электронно-вычислительные машины и системы"

Дисциплина Системы обработки больших данных

|  |  |
| --- | --- |
|  | Утверждаю  Зав. кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | «\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20 \_\_\_ г. |

ЗАДАНИЕ

на курсовую работу (проект)

Студент Перов Даниил Андреевич .

(фамилия, имя, отчество)

Группа САПР-1.4

1. Тема: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Утверждена приказом от «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г. № \_\_\_\_\_\_\_\_\_

2. Срок представления работы (проекта) к защите «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_ г.

3. Содержание расчетно-пояснительной записки: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Перечень графического материала: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Дата выдачи задания «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20 \_\_\_ г.

Руководитель работы (проекта)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата инициалы и фамилия

Задание принял к исполнению\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_ .

подпись, дата инициалы и фамилия

Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет \_Электроники и вычислительной техники\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра Системы автоматизированного проектирования и поискового конструирования

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе (проекту)**

по дисциплине Системы обработки больших данных

на тему .

Студент Перов Даниил Андреевич .

(фамилия, имя, отчество)

Группа САПР-1.4

Руководитель работы (проекта) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

Члены комиссии:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись и дата подписания) (инициалы и фамилия)

Нормоконтролер \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, дата подписания) (инициалы и фамилия)

Волгоград 2023 г.

Содержание

[Введение 4](#_Toc152286698)

[1. Разведочный анализ данных с помощью PySpark 5](#_Toc152286699)

[1.1 Постановка задачи 5](#_Toc152286700)

[1.2 Определение типов признаков в датасете 5](#_Toc152286701)

[1.3 Определение пропущенных значений и их устранение 9](#_Toc152286702)

[1.4 Определение и удаление выбросов 11](#_Toc152286703)

[1.5 Расчет статистических показателей признаков 12](#_Toc152286704)

[1.6 Визуализацией распределения наиболее важных признаков 15](#_Toc152286705)

[1.7 Корреляций между признаками 16](#_Toc152286706)

# Введение

В наше время, когда объемы информации растут с неимоверной быстротой, профессионалы в разных сферах сталкиваются с задачей анализа и обработки этих данных. Эти данные, которые называют "большими данными" (Big Data), отличаются своим размером, быстротой поступления и многообразием. Это предъявляет новые вызовы для исследователей и инженеров и требует специфического подхода к их обработке.

Выбранным основным инструментом стал PySpark, представляющий собой интерфейс для языка программирования Python в Apache Spark. Благодаря использованию PySpark, были получены практические навыки работы с RDD (Resilient Distributed Datasets) и DataFrame API - ключевыми абстракциями Apache Spark, которые обеспечивают эффективную и понятную обработку данных.

# 1. Разведочный анализ данных с помощью PySpark

## 1.1 Постановка задачи

Выполнить разведочный анализ датасета авиабилетов из Expedia согласно варианту с определением:

* типов признаков в датасете;
* пропущенных значений и их устранением;
* выбросов и их устранением;
* расчетом статистических показателей признаков (средних, квартилей и т.д.);
* визуализацией распределения наиболее важных признаков;
* корреляций между признаками.

Источник: <https://www.kaggle.com/datasets/dilwong/flightprices>

## 1.2 Определение типов признаков в датасете

Датасет содержит информацию о ценах на полеты в одну сторону самолетами согласно Expedia на период с 16.04.2022 до 05.10.2022.

Датасет представляет собой CSV-файл, где каждая строка - купленный билет в/из следующих аэропортов: ATL, DFW, DEN, ORD, LAX, CLT, MIA, JFK, EWR, SFO, DTW, BOS, PHL, LGA, IAD, OAK.

Определения столбцов:

1. legId: Идентификатор рейса.
2. searchDate: Дата (ГГГГ-ММ-ДД), когда эта запись была взята из Expedia.
3. flightDate: Дата (ГГГГ-ММ-ДД) полета.
4. startingAirport: Трехсимвольный код аэропорта IATA для начального местоположения.
5. destinationAirport: Трехсимвольный код аэропорта IATA для места прибытия.
6. fareBasisCode: Код основного тарифа.
7. travelDuration: Продолжительность поездки в часах и минутах.
8. elapsedDays: Количество прошедших дней (обычно 0).
9. isBasicEconomy: Булево значение, указывающее, предназначен ли билет для базовой экономики.
10. isRefundable: Булево значение, указывающее, можно ли вернуть деньги за билет.
11. isNonStop: Булево значение, указывающее, является ли рейс беспосадочным.
12. baseFare: Цена билета (в USD).
13. totalFare: Цена билета (в USD), включая налоги и другие сборы.
14. seatsRemaining: Целое число для количества оставшихся мест.
15. totalTravelDistance: Общее расстояние поездки в милях. Эти данные иногда отсутствуют.
16. segmentsDepartureTimeEpochSeconds: Строка, содержащая время отправления (время Unix) для каждого этапа поездки. Записи для каждого этапа разделяются символом '||'.
17. segmentsDepartureTimeRaw: Строка, содержащая время отправления (формат ISO 8601: YYYY-MM-DDThh:mm:ss.000±[hh]:00) для каждого этапа поездки. Записи для каждого из этапов разделяются символом '||'.
18. segmentsArrivalTimeEpochSeconds: Строка, содержащая время прибытия (время Unix) для каждого этапа поездки. Записи для каждого этапа разделяются символом '||'.
19. segmentsArrivalTimeRaw: Строка, содержащая время прибытия (формат ISO 8601: YYYY-MM-DDThh:mm:ss.000±[hh]:00) для каждого этапа поездки. Записи для каждого из этапов разделяются символом '||'.
20. segmentsArrivalAirportCode: Строка, содержащая код аэропорта IATA для места прибытия для каждого этапа поездки. Записи для каждого этапа разделяются символом '||'.
21. segmentsDepartureAirportCode: Строка, содержащая код аэропорта IATA для места отправления на каждом этапе путешествия. Записи для каждого этапа разделяются символом '||'.
22. segmentsAirlineName: Строка, содержащая название авиакомпании, обслуживающей каждый этап путешествия. Записи для каждого этапа разделяются символом '||'.
23. segmentsAirlineCode: Строка, содержащая двухбуквенный код авиакомпании, обслуживающей каждый этап путешествия. Записи для каждого этапа разделяются символом '||'.
24. segmentsEquipmentDescription: Строка, содержащая тип самолета, используемого на каждом этапе путешествия (например, "Airbus A321" или "Boeing 737-800"). Записи для каждого из этапов разделяются символом '||'.
25. segmentsDurationInSeconds: Строка, содержащая продолжительность полета (в секундах) для каждого этапа путешествия. Записи для каждого этапа разделяются символом '||'.
26. segmentsDistance: Строка, содержащая пройденное расстояние (в милях) для каждого этапа поездки. Записи для каждого этапа разделяются символом '||'.
27. segmentsCabinCode: Строка, содержащая код салона для каждого этапа поездки (например, "coach"). Записи для каждого этапа разделяются символом '||'.

Данные представляют собой значения следующих типов данных: integer, double, string, date, boolean. Типы колонок представлены в таблице 1.

| № | Название колонки | Тип данных | Тип признака |
| --- | --- | --- | --- |
| 0 | legId | string | Категориальный |
| 1 | searchDate | date | Порядковый |
| 2 | flightDate | date | Порядковый |
| 3 | startingAirport | string | Категориальный |
| 4 | destinationAirport | string | Категориальный |
| 5 | fareBasisCode | string | Категориальный |
| 6 | travelDuration | string | Категориальный |
| 7 | elapsedDays | int | Бинарный |
| 8 | isBasicEconomy | boolean | Бинарный |
| 9 | isRefundable | boolean | Бинарный |
| 10 | isNonStop | boolean | Бинарный |
| 11 | baseFare | double | Количественный |
| 12 | totalFare | double | Количественный |
| 13 | seatsRemaining | int | Количественный |
| 14 | totalTravelDistance | int | Количественный |
| 15 | segmentsDepartureTimeEpochSeconds | string | Категориальный |
| 16 | segmentsDepartureTimeRaw | string | Категориальный |
| 17 | segmentsArrivalTimeEpochSeconds | string | Категориальный |
| 18 | segmentsArrivalTimeRaw | string | Категориальный |
| 19 | segmentsArrivalAirportCode | string | Категориальный |
| 20 | segmentsDepartureAirportCode | string | Категориальный |
| 21 | segmentsAirlineName | string | Категориальный |
| 22 | segmentsAirlineCode | string | Категориальный |
| 23 | segmentsEquipmentDescription | string | Категориальный |
| 24 | segmentsDurationInSeconds | string | Категориальный |
| 25 | segmentsDistance | string | Категориальный |
| 26 | segmentsCabinCode | string | Категориальный |

Всего в датасете 808661 строк.

## 1.3 Определение пропущенных значений и их устранение

Для корректного анализа данных необходимо убедиться, что датасет не имеет какие-либо пропущенные и аномальные нулевые значения.

* Для строковых колонок проверим на None и null;
* Для числовых колонок, допускающих значения ноль, проверим на None и NaN;
* Для числовых колонок, недопускающих значения ноль, проверим на нули, None и NaN;
* Для булевых колонок проверим на None и null;
* Для колонок с датами проверим на None и null.

С помощью кода, изображённого на рисунке 1, было найдено количество пустых и нулевых значений, а также в каких колонках были найдены эти значения.



Рисунок 1 – Поиск нулевых значений

Количество удаленных записей и название колонок, содержащих пустые значения, представлен в таблице 2.

Таблица 2 – Удаленное количество строк за каждую колонку

|  |  |  |
| --- | --- | --- |
| № | totalTravelDistance | segmentsEquipmentDescription |
| 0 | 48700 | 15063 |

Количество строк после удаления пропущенных и аномальных нулевых значений – 746902.

## 1.4 Определение и удаление выбросов

Для удаления выбросов их необходимо определить. Воспользуемся Метод Z-оценки

Метод Z-оценки

Z-оценка, или “стандартная оценка”, — это статистическая мера, которая показывает, на сколько стандартных отклонений наблюдаемая точка удалена от среднего значения. При использовании этого метода необходимо определить порог: если точка данных имеет значение, превышающее пороговое, то она является выбросом.

Вычисляем Z следующим образом: z = (X - μ) / σ В этой формуле:

X — значение точки данных;

μ - среднее значения;

σ - стандартное отклонение

Метод Z-оценки имеет несколько недостатков:

* Его можно использовать только с одномерными данными (один столбец датафреймов, массивов, списков и т. д.);
* Он должен использоваться только с нормально распределенными данными;
* Исследователю придется определить порог, зависящий данных.

Для поиска выбросов примем ,что наблюдение считается как выброс, если его z-оценка меньше -3 или больше 3.

Выбросы = наблюдения с z-показателями > 3 или < -3.

Используя код, который изображён на рисунке 2, вычислим количество выбросов и удалим их.



Рисунок 2 – Метод Z-оценки

В результате использования метода Z-оценки были найдены выбросы. Количество оставшихся после удаления данных составило 905835 строк

## 1.5 Расчет статистических показателей признаков

Рассчитаем такие показатели, как:

* + Минимальное, среднее и максимальное значения;
  + Среднеквадратичное отклонение;
  + Квартили;
  + Коэффициент асимметрии;

Кратко рассмотрим каждый из этих статистических понятий:

1. Минимальное, среднее и максимальное значения:
   1. Минимальное значение — это наименьшее число в наборе данных.
   2. Среднее значение (или среднее арифметическое) — это сумма всех чисел в наборе, деленная на их количество.
   3. Максимальное значение — это наибольшее число в наборе данных.
2. Среднеквадратичное отклонение (или стандартное отклонение). Стандартное отклонение показывает, насколько в среднем значения в наборе отличаются от среднего значения.
3. Квартили — это значения, которые делят упорядоченный набор данных на три равные части. Они показывают распределение данных и включают в себя:

* Первый квартиль (Q1), который отделяет первые 25% данных;
* Второй квартиль (Q2), который также известен как медиана, отделяет первые 50% данных;
* Третий квартиль (Q3), который отделяет первые 75% данных;

1. Коэффициент асимметрии — это статистический показатель, который описывает степень асимметрии распределения данных относительно его среднего значения. Нормальное распределение имеет коэффициент асимметрии, равный 0, что указывает на симметричность данных. Положительное значение означает правостороннюю асимметрию, а отрицательное значение указывает на левостороннюю асимметрию.

Код реализации вычислений изображён на рисунке 3. Благодаря результатам вычислений, пример которых показан на рисунке 4, можно сделать следующие выводы:

1. Данные в колонке elapsedDays варьируется от 0 до 1, что предполагает, что данные охватывают как минимум один день.
2. Результаты для колонки seatsRemaining показывают, что количество оставшихся мест колеблется от 1 до 10. Стандартное отклонение равно 2.654, что указывает на умеренное разнообразие в количестве оставшихся мест между различными случаями. Отрицательное значение асимметрии говорит о том, что распределение немного смещено влево, с большим количеством случаев, когда оставшихся мест меньше среднего.
3. Результаты для колонок baseFare и totalFare показывают, что стоимость проезда варьируется на широкий диапазон цен. Положительное значение асимметрии указывает на то, что распределение смещено вправо.
4. Результаты для колонки totalTravelDistance показывают, что общее расстояние путешествия колеблется от 97 до 3958, что указывает на значительное разнообразие в расстояниях между различными маршрутами. Значение асимметрии близко к нулю, но положительно, что указывает на небольшое правостороннее смещение в распределении общего расстояния путешествия. Это может говорить о том, что более длинные поездки встречаются не так часто, но имеют место в данных.

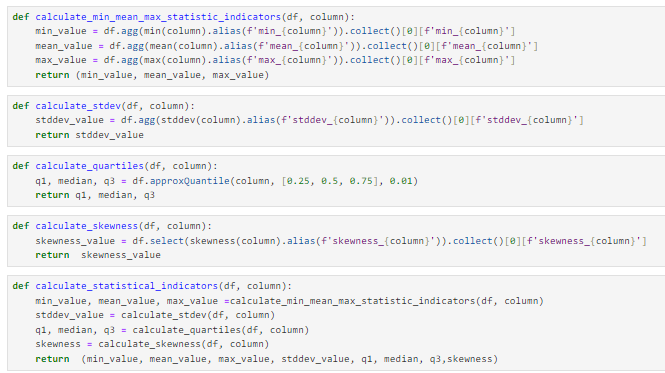


Рисунок 3 – Вычисление статистических параметров

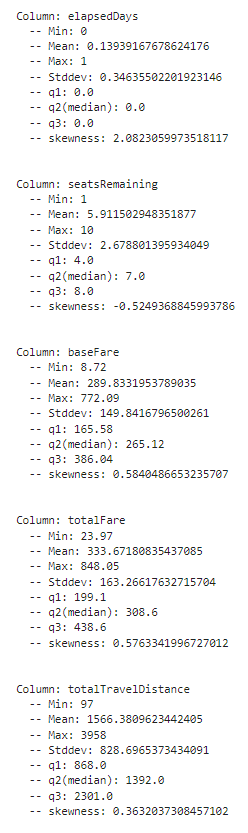


Рисунок 4 – Результат вычисления статистических параметров

## 1.6 Визуализацией распределения наиболее важных признаков

Для визуализации распределения наиболее важных признаков были использованы следующие графики:

* Гистограммы, пример которых иллюстрирует рисунок 5;
* Box plot, пример изображён на рисунке 6;
* Круговая диаграмма пример изображён на рисунке 7.

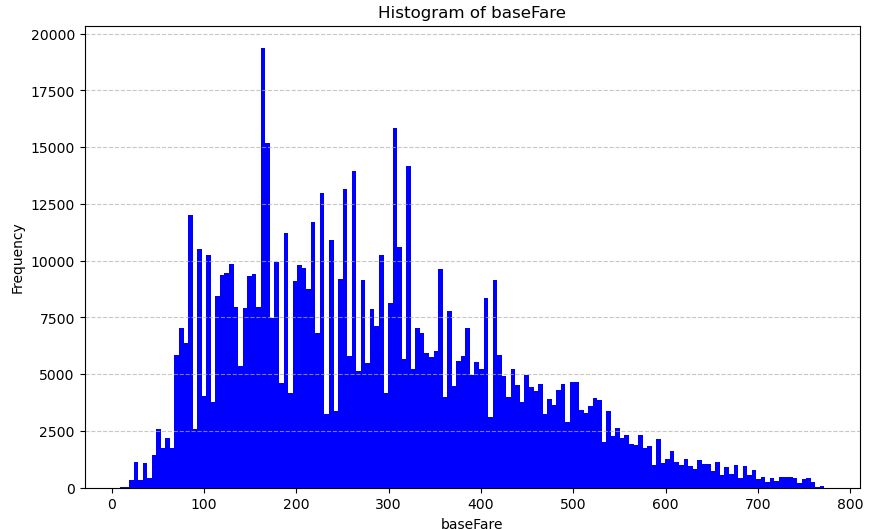


Рисунок 5 – Визуализация гистограммы распределения базовых цен

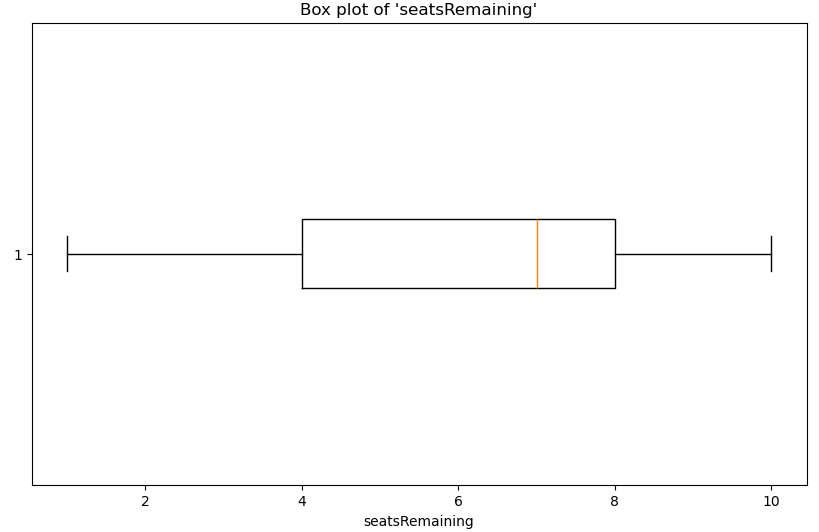


Рисунок 6 – Визуализация Box plot распределения количества оставшихся мест.

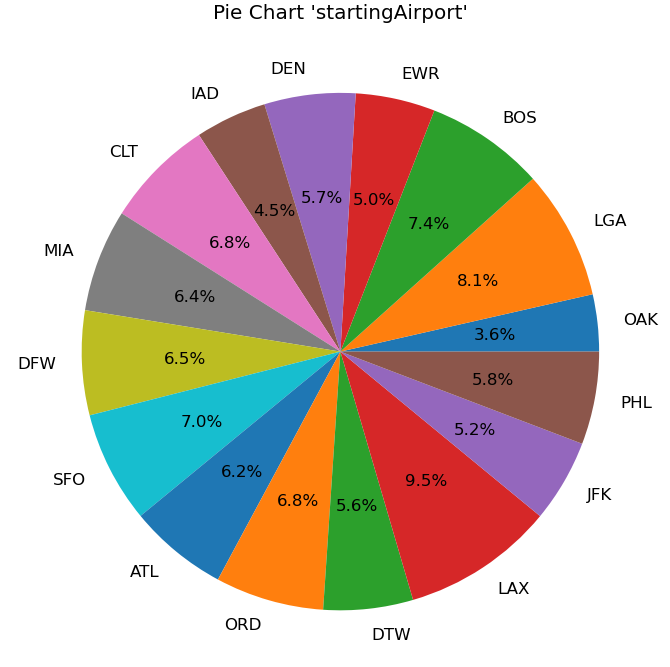


Рисунок 7 – Визуализация круговой диаграммы распределения индексов начальных аэропортов.

## 1.7 Корреляций между признаками

Чтобы выявить связи между признаками, можно построить матрицу корреляций. Матрица корреляций показана на рисунке 8.

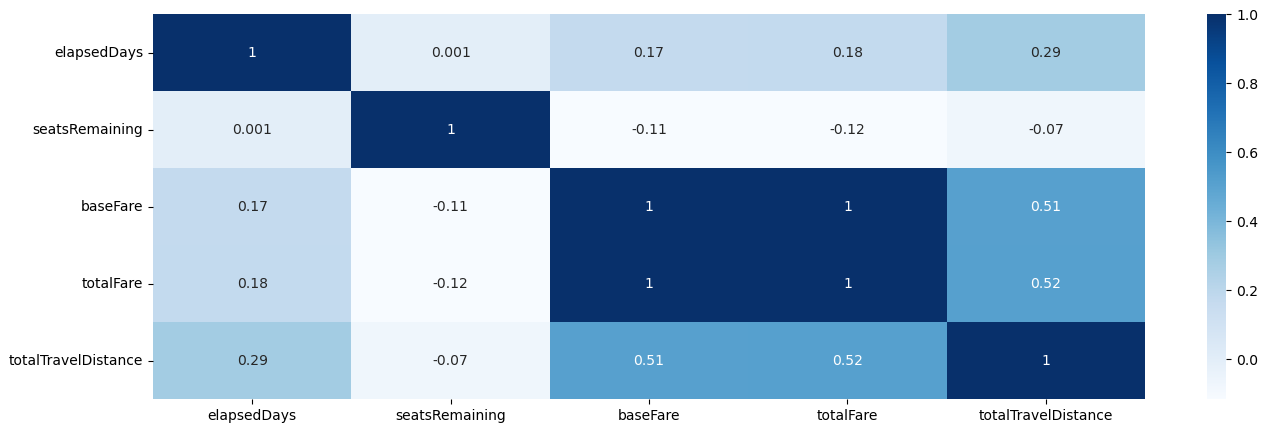


Рисунок 8 – Матрица корреляций

Из корреляционной матрицы видно, что baseFare и totalFare демонстрируют тесную положительную взаимосвязь, поскольку цена билета всегда включает одну и ту же сумму налогов.