Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



# Отчет

**Лабораторная работа № 1**

**По курсу «Технологии машинного обучения»**

[**«Разведочный анализ данных.**](https://en.wikipedia.org/wiki/Exploratory_data_analysis) **Исследование и визуализация данных»**

**ИСПОЛНИТЕЛЬ:**

Мехмандаров Мурад

Группа ИУ5-63

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"\_\_"\_\_\_\_\_\_\_\_\_\_\_2020 г.

**ПРЕПОДАВАТЕЛЬ:**

Гапанюк Ю.Е.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ "\_\_"\_\_\_\_\_\_\_\_\_\_\_2020 г.

Москва 2020

**Цель работы**

Изучить различные методы визуализации данных

## Задание

Выбрать набор данных

Создать ноутбук, который содержить следующие разделы:

Текстовое описание выбранного наборы данных

Основные характеристики датасета

Визуальное исследование датасета

Информация о корелляции признаков

Сформировать отчет и разместить его на своем репозитории GitHub

## Ход выполнения лабораторной работы

### 1. Набор данных

Этот набор данных содержит информацию о бронировании для городской гостиницы и курортного отеля и включает в себя такую информацию, как, например, время бронирования, продолжительность пребывания, количество взрослых, детей и / или детей и количество доступных парковочных мест.

hotel - Отель (H1 = Курортный отель или H2 = Городской отель) is\_canceled

lead\_time - время выполнения заказа

arrival\_date\_year arrival\_date\_month arrival\_date\_week\_number arrival\_date\_day\_of\_month stays\_in\_weekend\_nights stays\_in\_week\_nights

adults children

babies

meal - Тип еды забронирован. Категории представлены в стандартных пакетах питания для гостей: Undefined / SC - без питания; BB -

кровать и завтрак; HB - полупансион (завтрак и еще один прием пищи - обычно ужин); FB - полный пансион (завтрак, обед и ужин)

country

market\_segment - Обозначение сегмента рынка. В категориях термин «ТА» означает «Туристические агенты», а «TO» означает

«Туроператоры».

distribution\_channel - Канал распределения бронирования. Термин «TA» означает «Туристические агенты», а «TO» означает

«Туроператоры».

is\_repeated\_guest

previous\_cancellations - Количество предыдущих заказов, которые были отменены клиентом до текущего бронирования

previous\_bookings\_not\_canceled

reserved\_room\_type

assigned\_room\_type

booking\_changes

deposit\_type

agent

company

days\_in\_waiting\_list

customer\_type

adr Средняя дневная ставка, определенная путем деления суммы всех транзакций на проживание на общее количество ночей

проживания.

required\_car\_parking\_spaces

total\_of\_special\_requests

reservation\_status

reservation\_status\_date

In

[10]:

**import**

numpy

**as**

np

**import**

seaborn

**as**

sns

**import**

matplotlib

.

pyplot

**as**

plt

**import**

pandas

**%**

matplotlib

inline

sns

.

set

(

style

**=**

"ticks"

)

data

**=**

pandas

.

read\_csv

(

'../data/hotel\_bookings.csv'

)

### 2. Основные харектеристики датасета

In

[12]:

data

.

head

(

10

)

Out[12]:

**hotel is\_canceled lead\_time arrival\_date\_year arrival\_date\_month arrival\_date\_week\_number arrival\_date\_day\_of\_month stays\_in\_weekend\_nights**

1. ResortHotel 0 342 2015 July 27 1 0
2. ResortHotel 0 737 2015 July 27 1 0
3. ResortHotel 0 7 2015 July 27 1 0
4. ResortHotel 0 13 2015 July 27 1 0
5. ResortHotel 0 14 2015 July 27 1 0
6. ResortHotel 0 14 2015 July 27 1 0
7. ResortHotel 0 0 2015 July 27 1 0
8. ResortHotel 0 9 2015 July 27 1 0
9. ResortHotel 1 85 2015 July 27 1 0
10. Resort 1 75 2015 July 27 1 0

Hotel

1. rows × 32 columns

In

[13]:

data

.

shape

Out[13]: (119390, 32)

data.columns

Out[14]: Index(['hotel', 'is\_canceled', 'lead\_time', 'arrival\_date\_year',

'arrival\_date\_month', 'arrival\_date\_week\_number',

'arrival\_date\_day\_of\_month', 'stays\_in\_weekend\_nights',

'stays\_in\_week\_nights', 'adults', 'children', 'babies', 'meal',

'country', 'market\_segment', 'distribution\_channel',

'is\_repeated\_guest', 'previous\_cancellations',

'previous\_bookings\_not\_canceled', 'reserved\_room\_type',

'assigned\_room\_type', 'booking\_changes', 'deposit\_type', 'agent',

'company', 'days\_in\_waiting\_list', 'customer\_type', 'adr',

'required\_car\_parking\_spaces', 'total\_of\_special\_requests', 'reservation\_status', 'reservation\_status\_date'], dtype='object')

data.dtypes

Out[15]: hotel object is\_canceled int64 lead\_time int64 arrival\_date\_year int64 arrival\_date\_month object arrival\_date\_week\_number int64 arrival\_date\_day\_of\_month int64 stays\_in\_weekend\_nights int64 stays\_in\_week\_nights int64 adults int64 children float64 babies int64 meal object country object market\_segment object distribution\_channel object is\_repeated\_guest int64 previous\_cancellations int64 previous\_bookings\_not\_canceled int64 reserved\_room\_type object assigned\_room\_type object booking\_changes int64 deposit\_type object agent float64 company float64 days\_in\_waiting\_list int64 customer\_type object adr float64 required\_car\_parking\_spaces int64 total\_of\_special\_requests int64 reservation\_status object reservation\_status\_date object dtype: object

data.describe()

Out[16]: **is\_canceled lead\_time arrival\_date\_year arrival\_date\_week\_number arrival\_date\_day\_of\_month stays\_in\_weekend\_nights stays\_in\_week\_**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 119390.000000 | 119390.000000 | 119390.000000 | 119390.000000 | 119390.000000 | 119390.000000 | 119390.0 |
| **mean** | 0.370416 | 104.011416 | 2016.156554 | 27.165173 | 15.798241 | 0.927599 | 2.5 |
| **std** | 0.482918 | 106.863097 | 0.707476 | 13.605138 | 8.780829 | 0.998613 | 1.9 |
| **min** | 0.000000 | 0.000000 | 2015.000000 | 1.000000 | 1.000000 | 0.000000 | 0.0 |
| **25%** | 0.000000 | 18.000000 | 2016.000000 | 16.000000 | 8.000000 | 0.000000 | 1.0 |
| **50%** | 0.000000 | 69.000000 | 2016.000000 | 28.000000 | 16.000000 | 1.000000 | 2.0 |
| **75%** | 1.000000 | 160.000000 | 2017.000000 | 38.000000 | 23.000000 | 2.000000 | 3.0 |
| **max** | 1.000000 | 737.000000 | 2017.000000 | 53.000000 | 31.000000 | 19.000000 | 50.0 |

In

[17]:

data

[

'is\_repeated\_guest'

].

unique

()

Out[17]: array([0, 1])

[18]:

hotel - 0

is\_canceled - 0

*#*

*Проверим*

*наличие*

*пустых*

*значений*

*#*

*Цикл*

*по*

*колонкам*

*датасета*

**for**

col

**in**

data

.

columns

:

*#*

*Количество*

*пустых*

*значений*

*-*

*все*

*значения*

*заполнены*

temp\_null\_count

**=**

data

[

data

[

col

].

isnull

()].

shape

[

0

]

print

(

'{} - {}'

.

format

(

col

,

temp\_null\_count

))

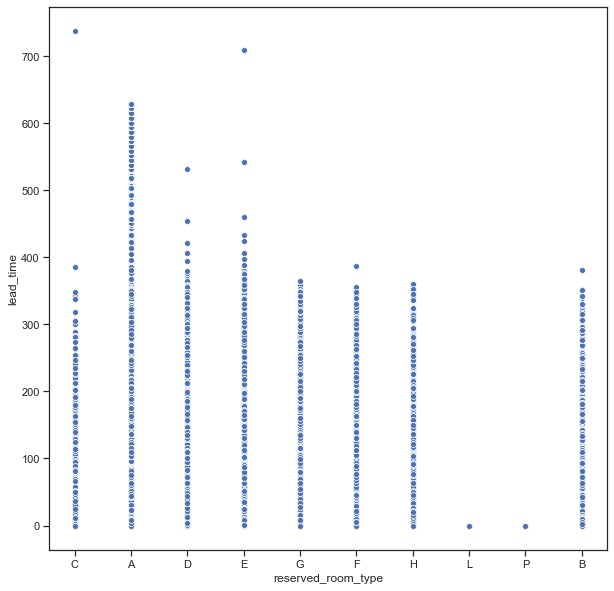
lead\_time - 0 arrival\_date\_year - 0 arrival\_date\_month - 0 arrival\_date\_week\_number - 0 arrival\_date\_day\_of\_month - 0 stays\_in\_weekend\_nights - 0 stays\_in\_week\_nights - 0 adults - 0 children - 4 babies - 0 meal - 0 country - 488 market\_segment - 0 distribution\_channel - 0 is\_repeated\_guest - 0 previous\_cancellations - 0 previous\_bookings\_not\_canceled - 0 reserved\_room\_type - 0 assigned\_room\_type - 0 booking\_changes - 0 deposit\_type - 0 agent - 16340 company - 112593 days\_in\_waiting\_list - 0 customer\_type - 0 adr - 0 required\_car\_parking\_spaces - 0 total\_of\_special\_requests - 0 reservation\_status - 0 reservation\_status\_date - 0

### 3. Визуальное исследования датасета

#### Диаграмма расеяния

[30]: fig, ax **=** plt.subplots(figsize**=**(10,10)) sns.scatterplot(ax**=**ax, x**=**'reserved\_room\_type', y**=**'lead\_time', data**=**data)

Out[30]: <matplotlib.axes.\_subplots.AxesSubplot at 0x127dcba10>



По диаграмме расеяния можно понять, что в среденем люди бронируют раньше тип А. Но если исследовать более глубоко, то в комнату B

По диаграмме расеяния можно понять, что в среденем люди бронируют раньше тип А. Но если исследовать более глубоко, то в комнату B в среднем бранируют за более ранний срок.

df[df['reserved\_room\_type’] == ‘A’][‘lead\_time’].mean() === 110

df[df[‘reserved\_room\_type’] == ‘b’][‘lead\_time’].mean() === 113

#### Гистограмма

[41]:

fig

,

ax

**=**

plt

.

subplots

(

figsize

**=**

(

10

,

10

))

sns

.

distplot

(

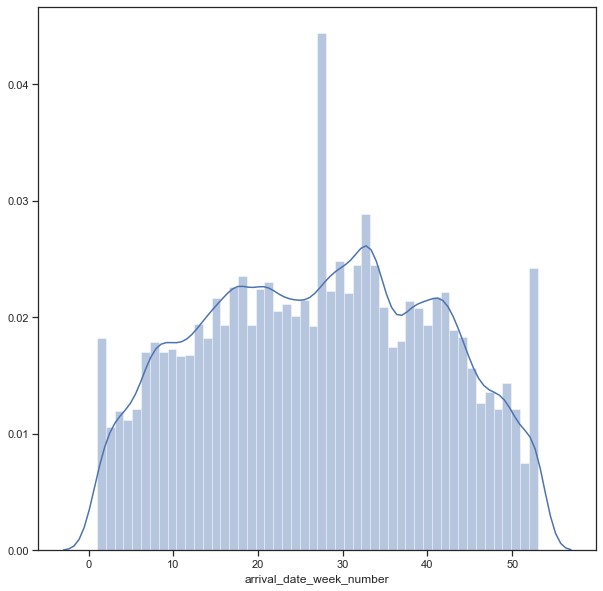
data

[

'arrival\_date\_week\_number'

])

Out[41]: <matplotlib.axes.\_subplots.AxesSubplot at 0x128c7cf10>

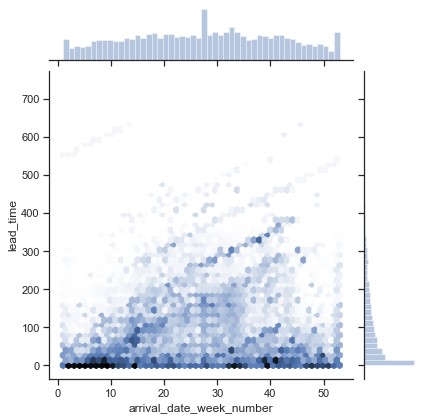


По гистограмме видно, что в 26 неделю года больше всего людей бронируют отели.

#### Jointplot

[44]: sns.jointplot(x**=**'arrival\_date\_week\_number', y**=**'lead\_time', data**=**data, kind**=**"hex")

Out[44]: <seaborn.axisgrid.JointGrid at 0x12e1373d0>



Можно сделать вывод что чем позже в году прибытие, тем дольше время прибывания.

#### Парная диаграмма

In

[48]:

sns

.

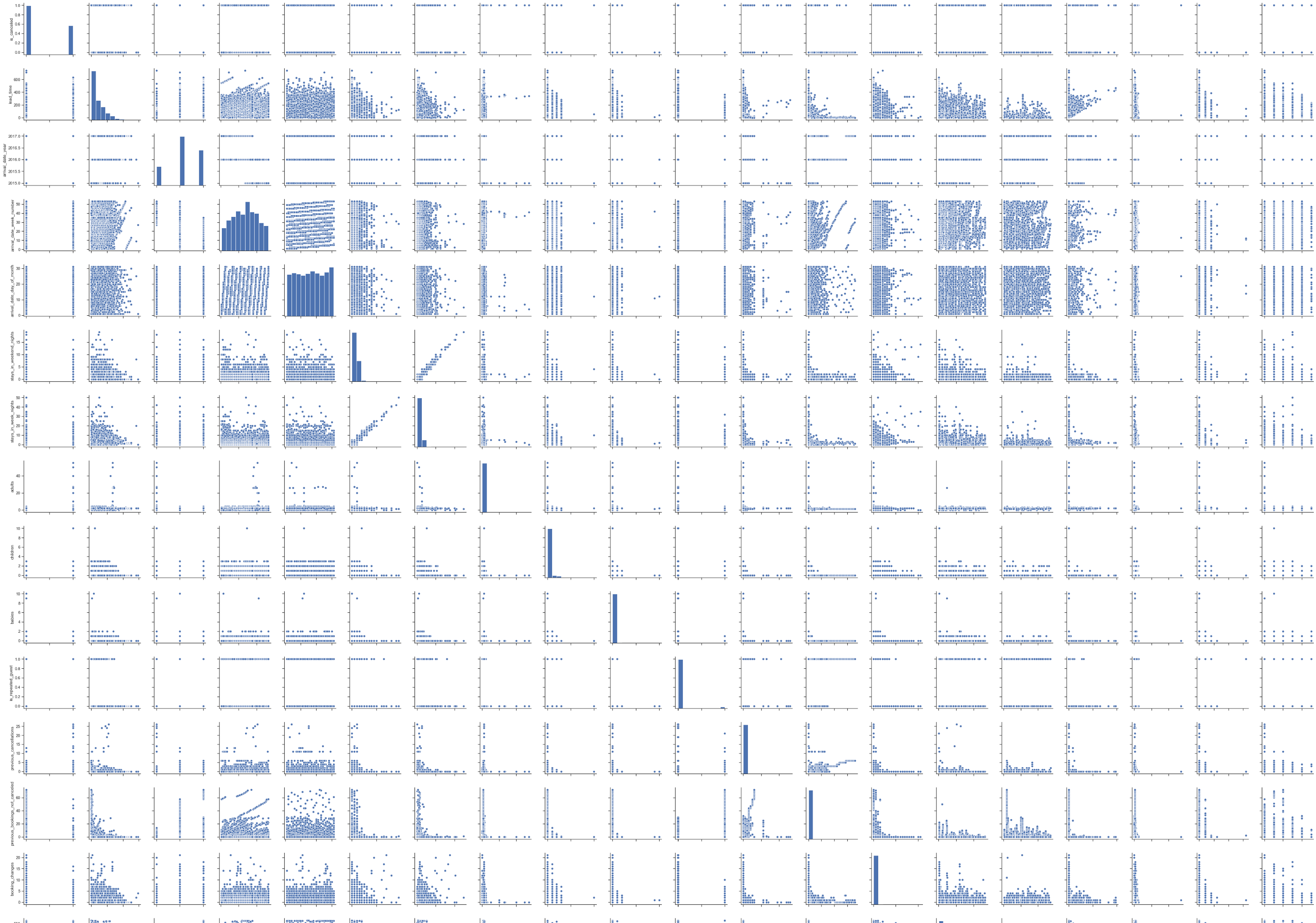
pairplot

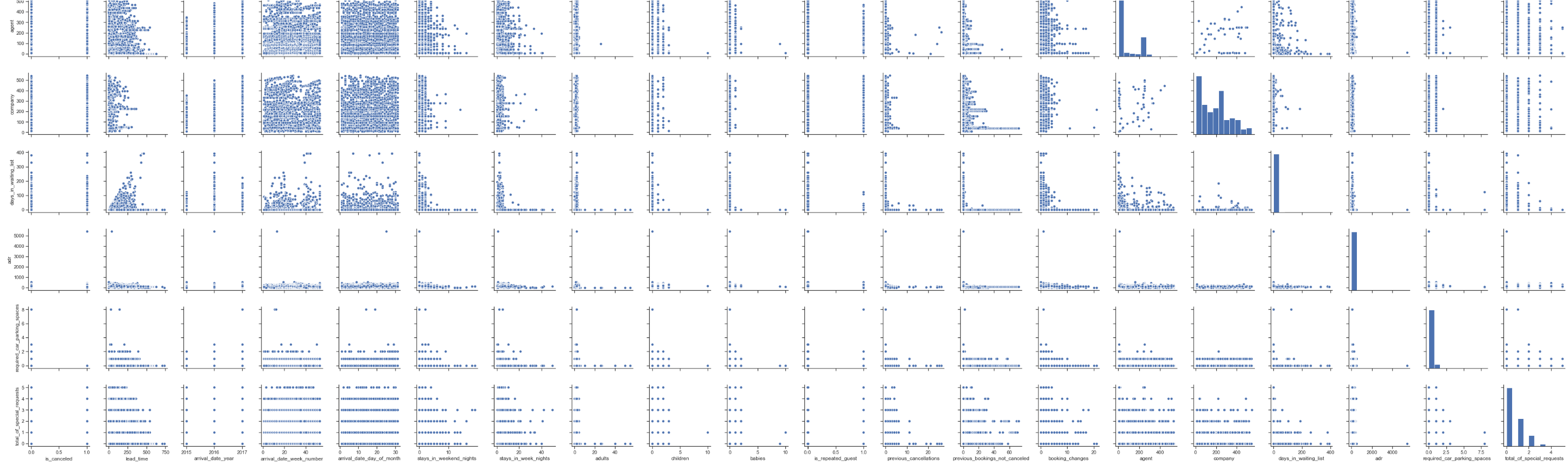
(

data

)

Out[48]: <seaborn.axisgrid.PairGrid at 0x1eb58b210>





#### Ящик с усами

In

[50]:

sns

.

boxplot

(

x

**=**

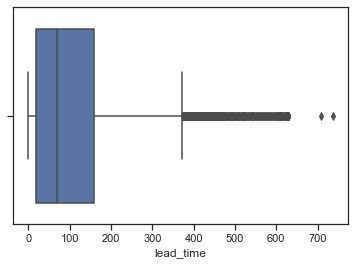
data

[

'lead\_time'

])

Out[50]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1eca81f10>



#### Информация о корреляции признаков

Построим корреляционную матрицу по всему набору данных Проверка корреляции признаков позволяет решить две задачи:

Понять какие признаки (колонки датасета) наиболее сильно коррелируют с целевым признаком (в нашем примере это колонка "lead\_team").Именно эти признаки будут наиболее информативными для моделей машинного обучения. Признаки, которые слабо коррелируют с целевым признаком, можно попробовать исключить из построения модели, иногда это повышает качество модели. Нужно отметить, что некоторые алгоритмы машинного обучения автоматически определяют ценность того или иного признака для построения модели.

Понять какие нецелевые признаки линейно зависимы между собой. Линейно зависимые признаки, как правило, очень плохо влияют на качество моделей. Поэтому если несколько признаков линейно зависимы, то для построения модели из них выбирают какой-то один признак.

In

[51]:

data

.

corr

()

Out[51]: **is\_canceled lead\_time arrival\_date\_year arrival\_date\_week\_number arrival\_date\_day\_of\_month stays\_in\_weekend\_night**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **is\_canceled** | 1.000000 | 0.293123 | 0.016660 | 0.008148 | -0.006130 | -0.00179 |
| **lead\_time** | 0.293123 | 1.000000 | 0.040142 | 0.126871 | 0.002268 | 0.08567 |
| **arrival\_date\_year** | 0.016660 | 0.040142 | 1.000000 | -0.540561 | -0.000221 | 0.02149 |
| **arrival\_date\_week\_number** | 0.008148 | 0.126871 | -0.540561 | 1.000000 | 0.066809 | 0.01820 |
| **arrival\_date\_day\_of\_month** | -0.006130 | 0.002268 | -0.000221 | 0.066809 | 1.000000 | -0.01635 |
| **stays\_in\_weekend\_nights** | -0.001791 | 0.085671 | 0.021497 | 0.018208 | -0.016354 | 1.00000 |
| **stays\_in\_week\_nights** | 0.024765 | 0.165799 | 0.030883 | 0.015558 | -0.028174 | 0.49896 |
| **adults** | 0.060017 | 0.119519 | 0.029635 | 0.025909 | -0.001566 | 0.09187 |
| **children** | 0.005048 | -0.037622 | 0.054624 | 0.005518 | 0.014544 | 0.04579 |
| **babies** | -0.032491 | -0.020915 | -0.013192 | 0.010395 | -0.000230 | 0.01848 |
| **is\_repeated\_guest** | -0.084793 | -0.124410 | 0.010341 | -0.030131 | -0.006145 | -0.08723 |

Видим, что: lead\_time кореллирует с is\_canceled (0.3). От того что отменена бронь или нет, сильно зависит время с покупки до вьезда

In

[54]:

sns

.

heatmap

(

data

.

corr

())

Out[54]: <matplotlib.axes.\_subplots.AxesSubplot at 0x229bf04d0>

