



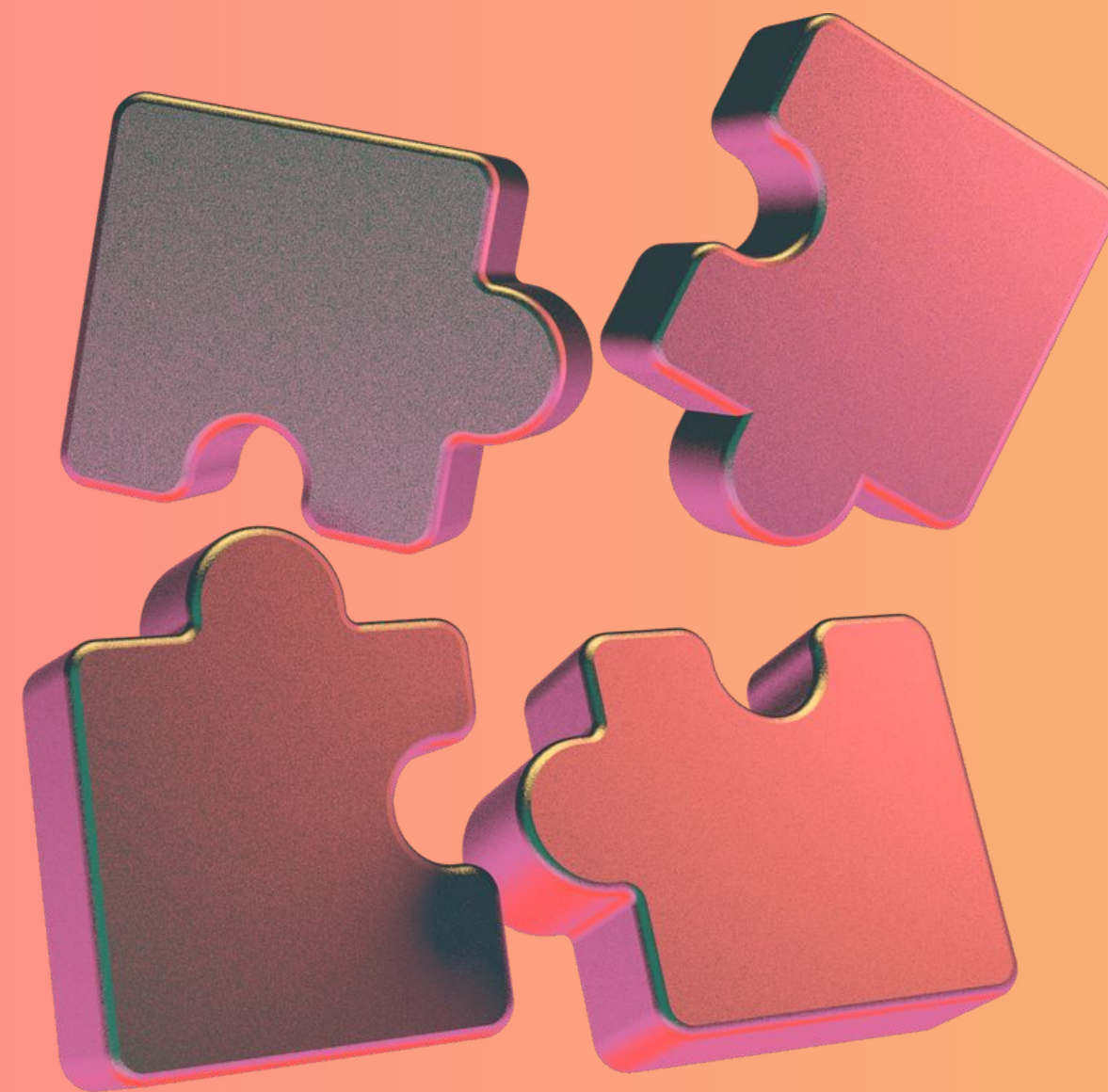
ВЫШКА
ОНЛАЙН

ПСБ
ЦИФРОВАЯ
ЛАБОРАТОРИЯ

ВШЭ ПСБ.Хак

Хакатон по искусственному
интеллекту для студентов

20-22 сентября 2024





Афонасьев
Александр



Гринченко
Евгений



Апаркин
Матвей



Таран Лев



Колотушкин
Никита

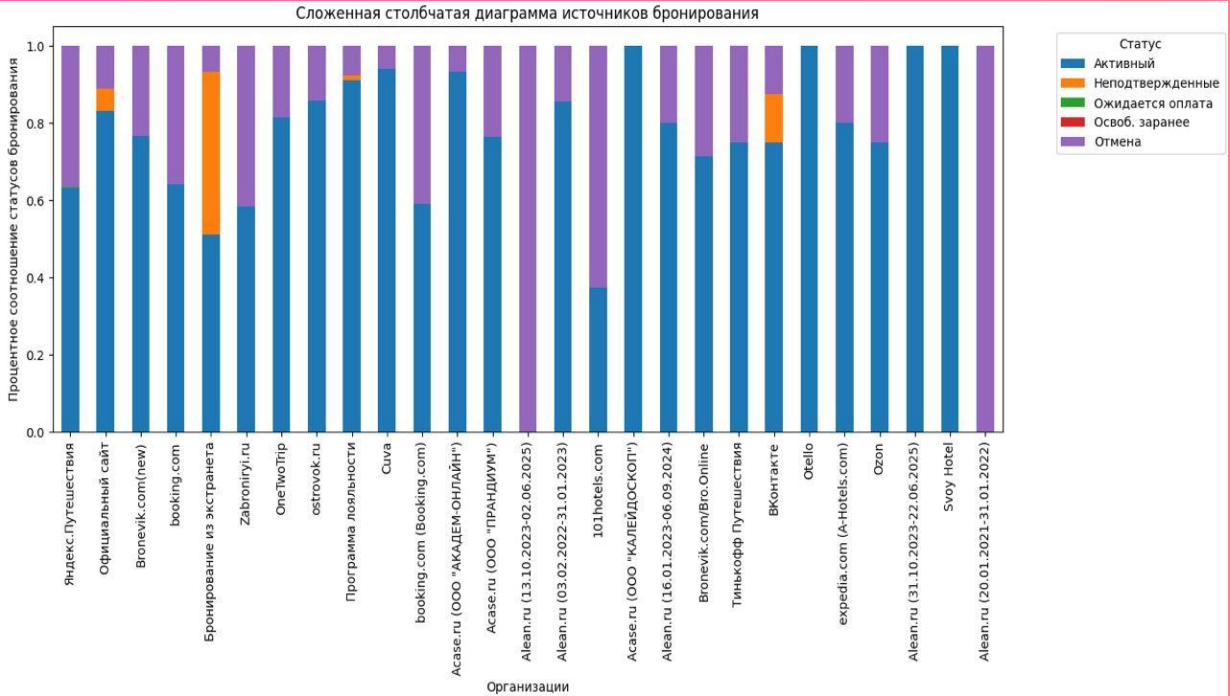
Fine tuning

«In God we trust, all others must bring data.»
W. Edwards Deming

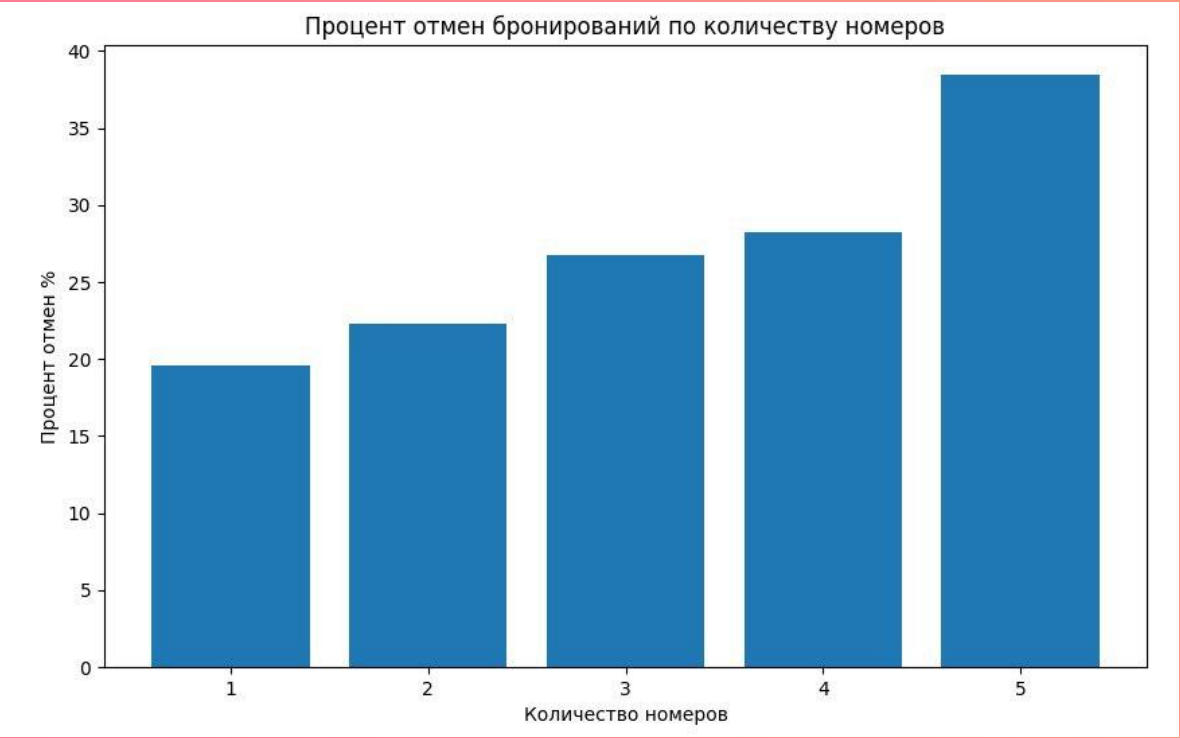


Проблема





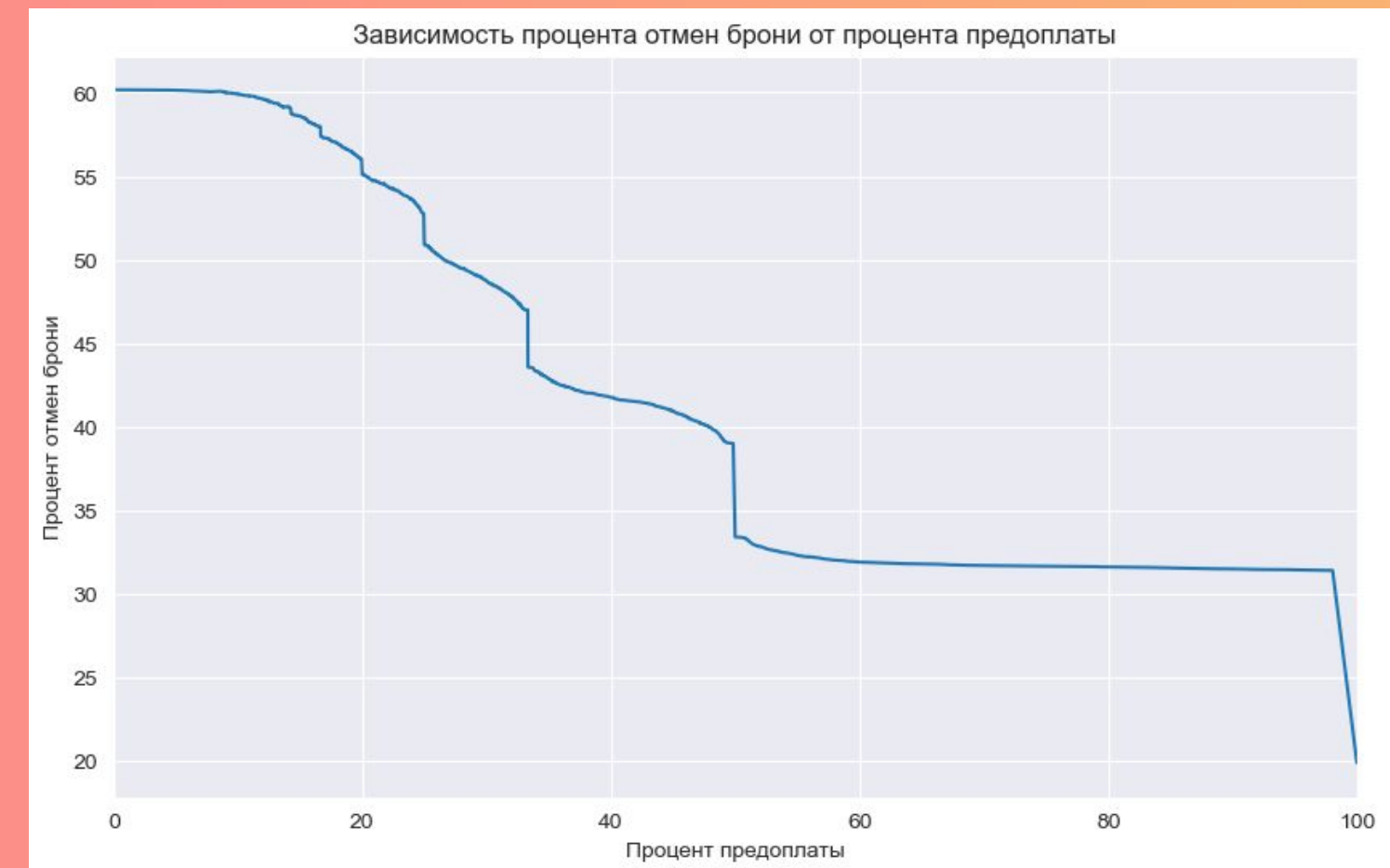
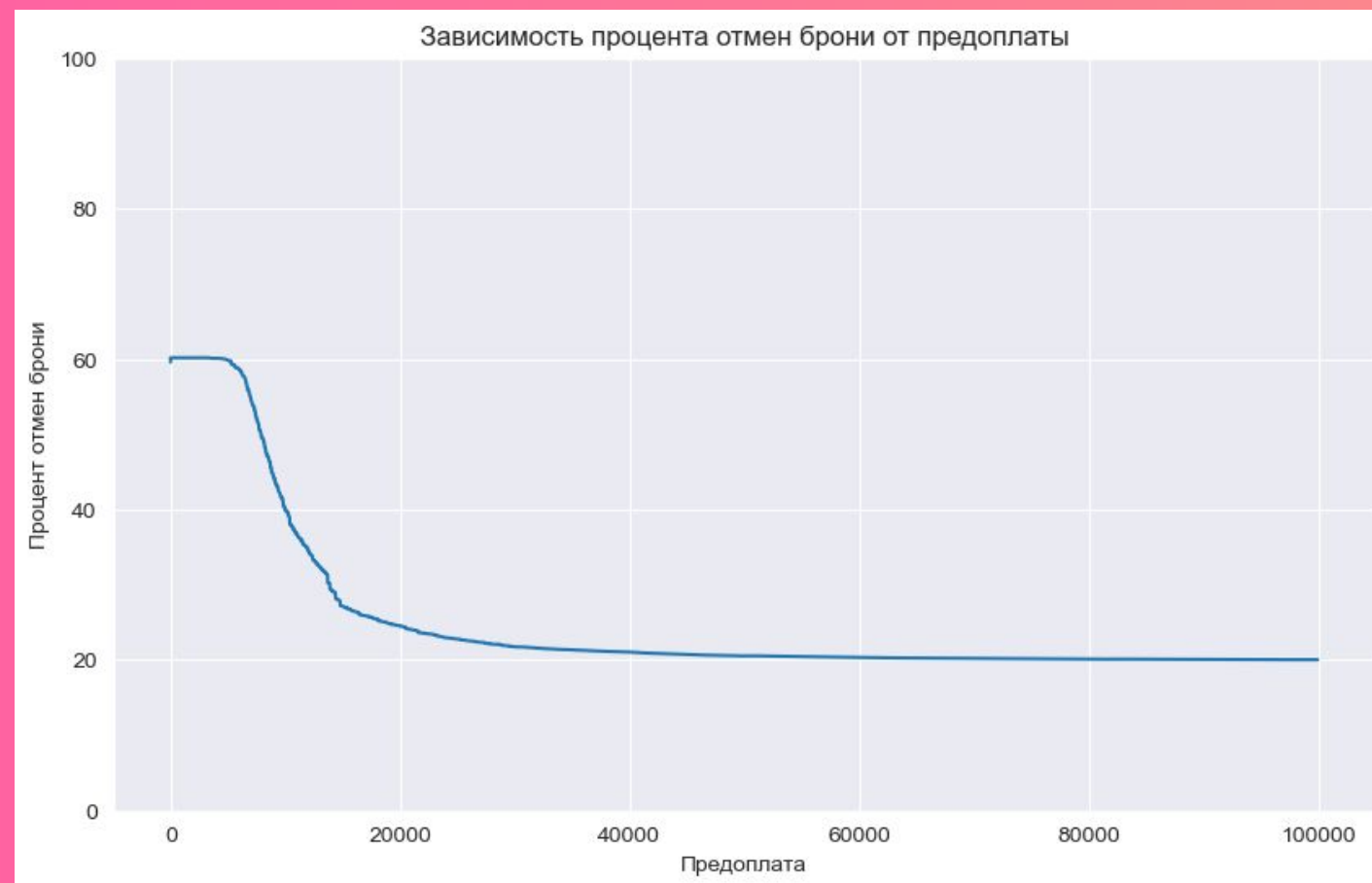
Гипотеза 1: Способ оплаты значительно влияет на вероятность отмены бронирования.



Гипотеза 2: Чем дольше планируемый срок проживания, тем ниже вероятность отмены бронирования.

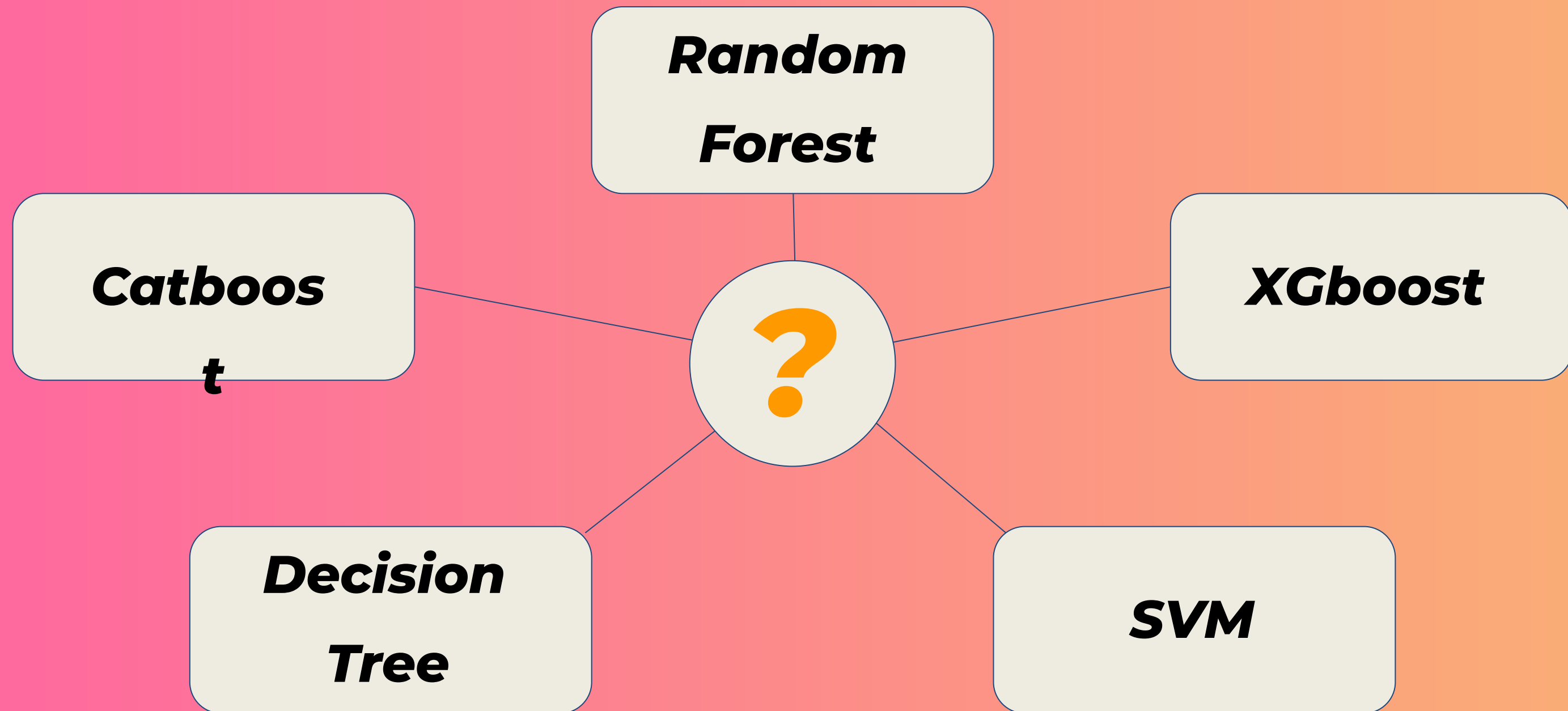
Гипотеза 3: Чем больше человек бронирует номеров, тем выше становится процент отмены

Гипотезы



Гипотеза 4: Наличие предоплаты положительно влияет на снижение числа отмен и повышение уровня подтверждений бронирований.

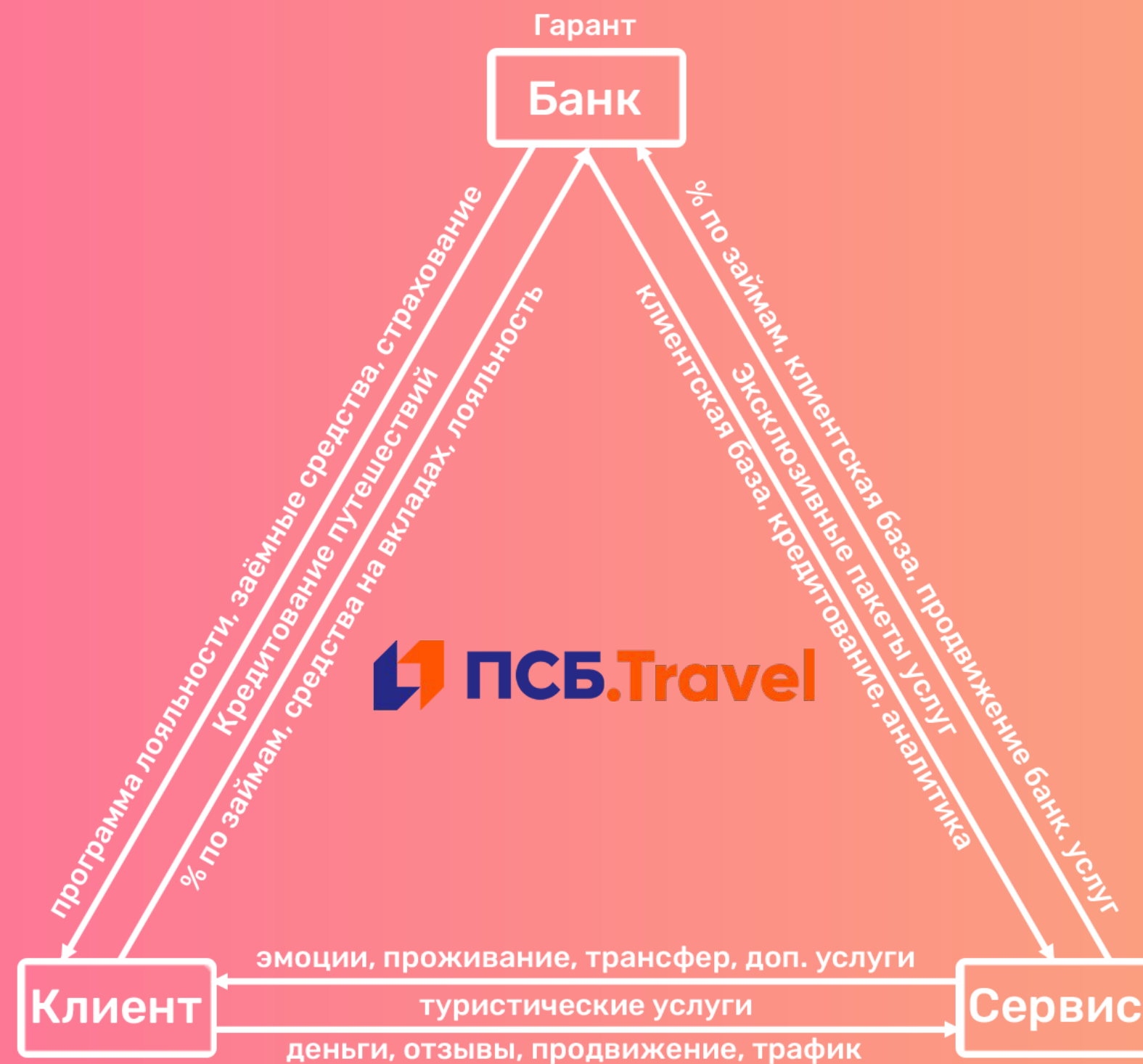
Модель



Польза



Продуктовая концепция



Количественная аналитика

Активная пользовательская аудитория ПСБ – **6,1 млн. чел.**

Трудоустроенное население в РФ – **74 млн. чел.**
Активно берущее в кредит население – **10% (порядка 7,4 млн. чел)**. Из них **12%** готовы взять кредит на путешествие, размером **не более 100 тыс. руб.** на человека. По нашим первичным оценкам, ПСБ может занять долю рынка около 30%. Итого, получаем 250тыс. потенциальных клиентов

Средний размер потребительского кредита в РФ: **177 тыс. руб.**

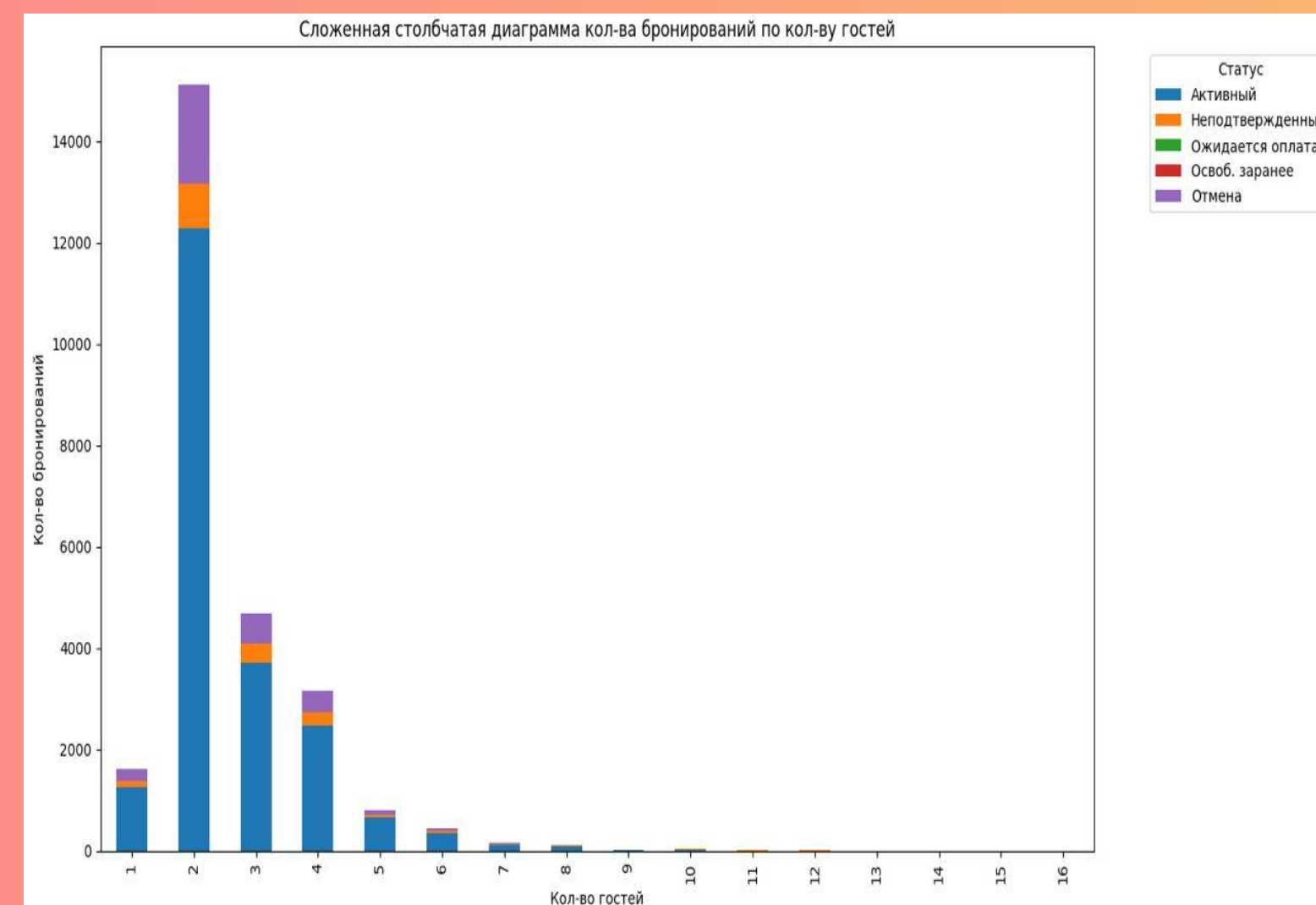
В Мск: **312 тыс. руб.**

В МО: **230 тыс. руб.**

В СПб: **253 тыс. руб.**

Большинство заемщиков - **25-30 лет**

Большинство путешественников - **25-30 лет**



Целевая аудитория

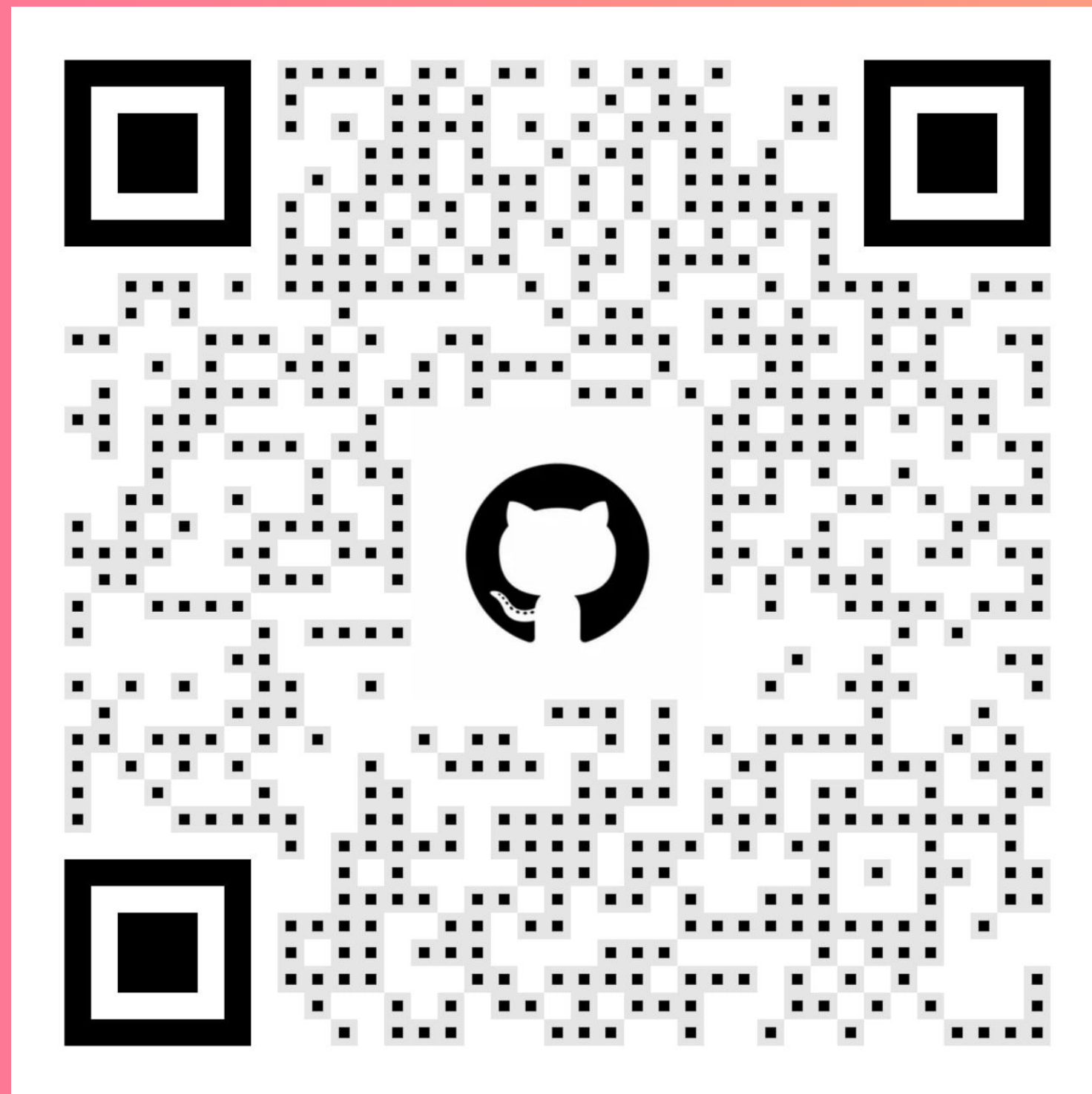


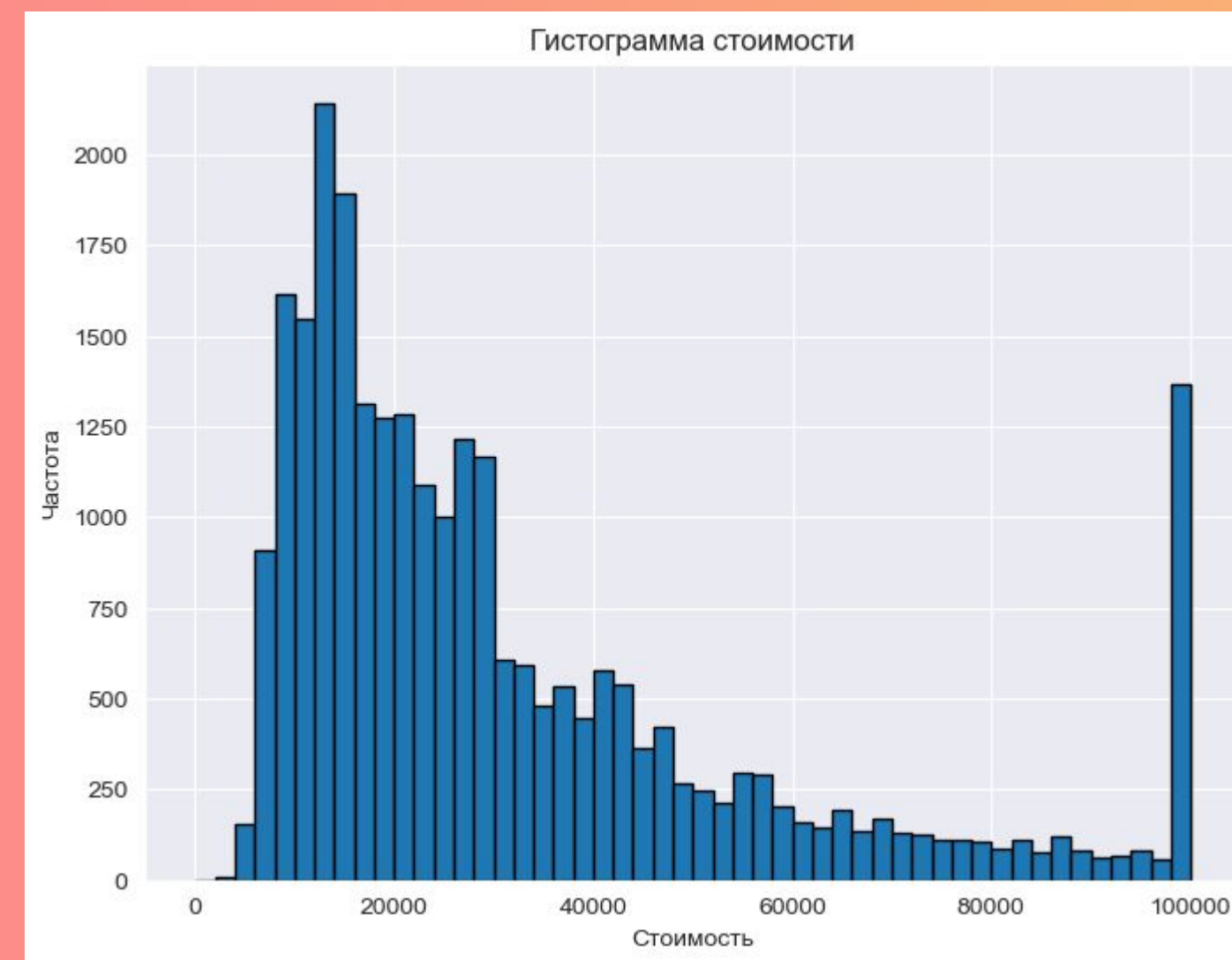
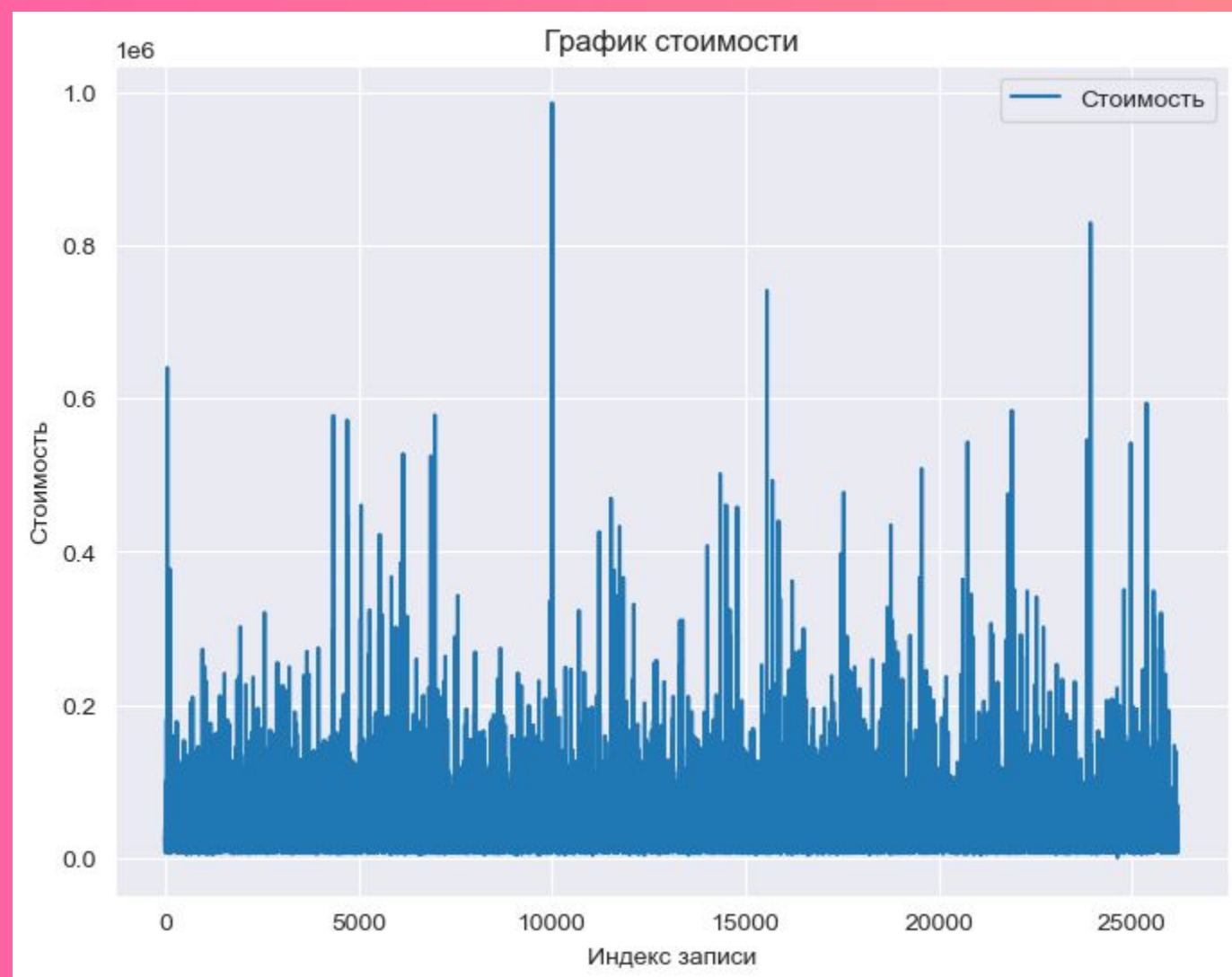
ИСТОЧНИКИ:

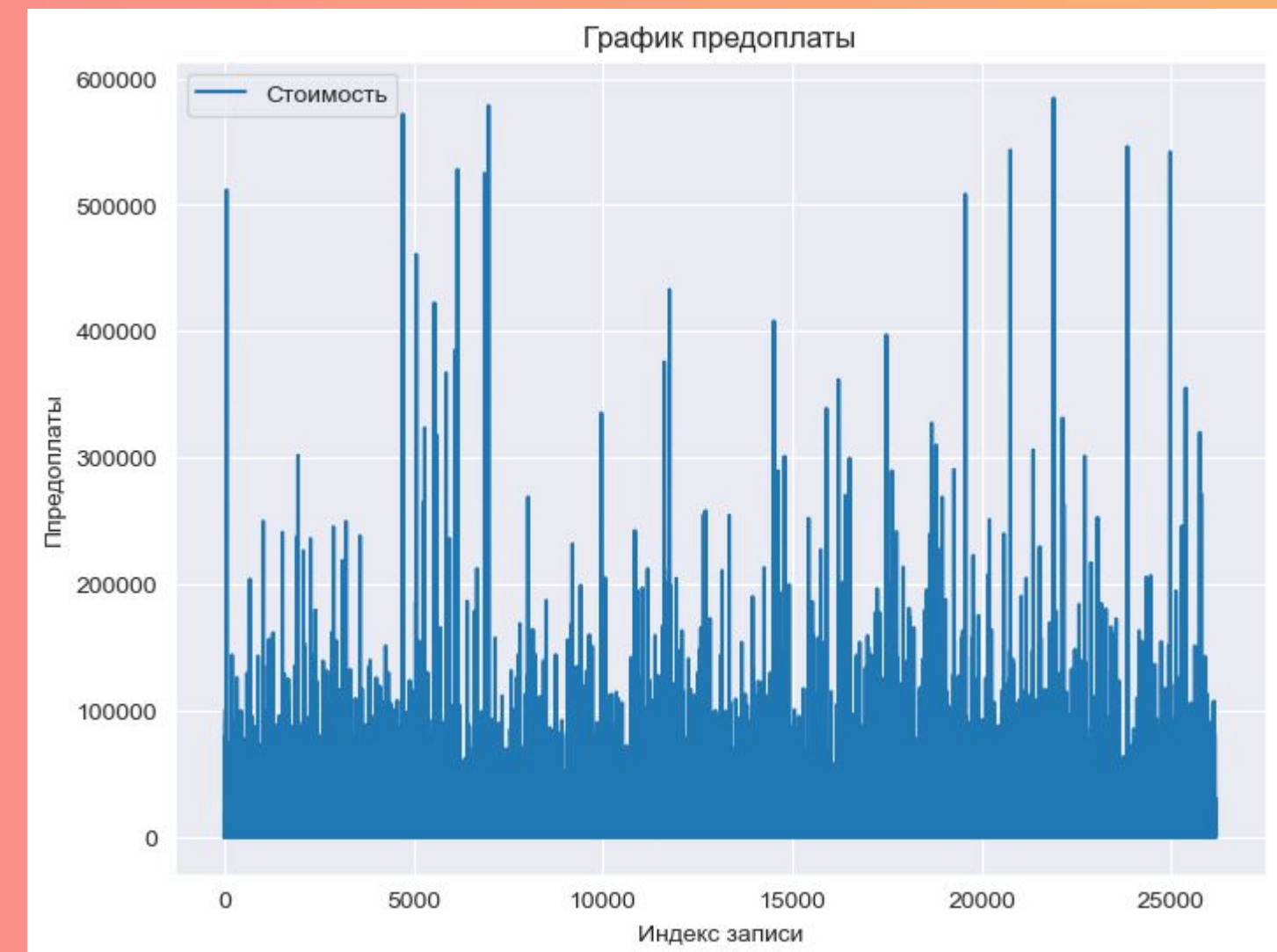
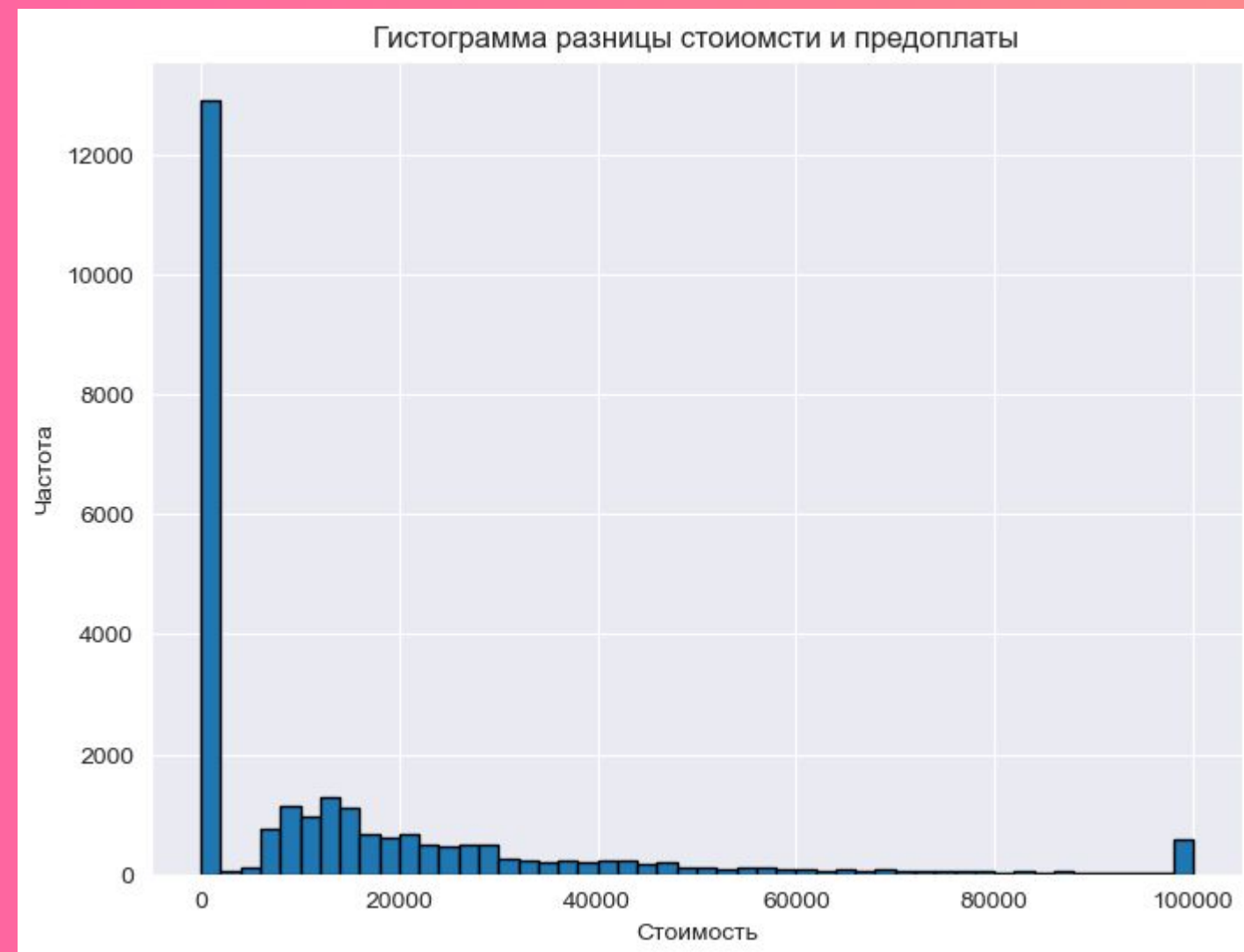
- <https://journal.sovcombank.ru/news/boleee-80-rossiyan-ne-gotovi-brat-kredit-na-otpusk>
- https://rosstat.gov.ru/labour_force
- <https://clck.ru/3DSNfk>
- <https://press.rabota.ru/rossiyane-rasskazali-skolko-gotovy-potratit-na-otpusk-v-sentyabre>
- https://cbr.ru/Collection/Collection/File/46543/inf-material_bki_2023fh.pdf
- <https://clck.ru/3DSNjs>
- <https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand/data>
- <https://clck.ru/3DSRFN>
- <https://chatgptchatapp.com/>
-

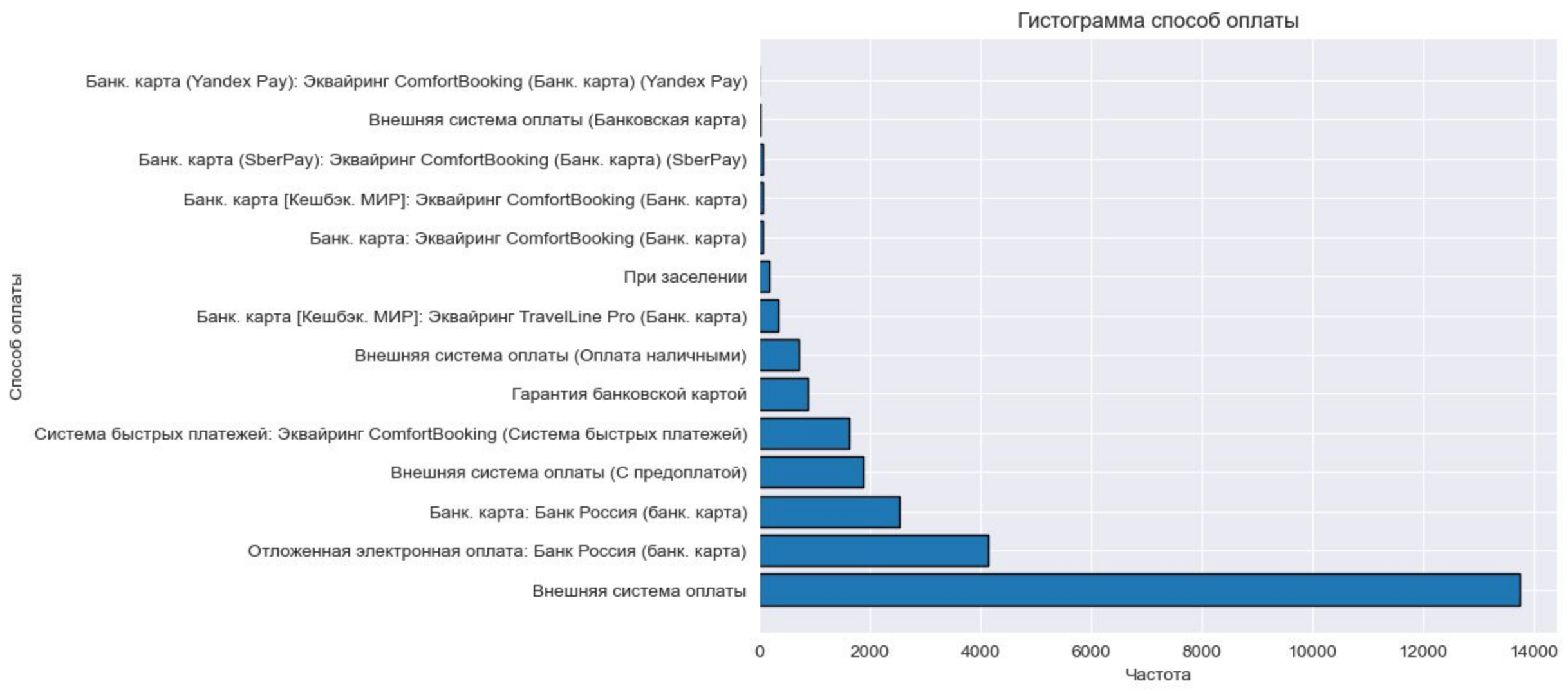


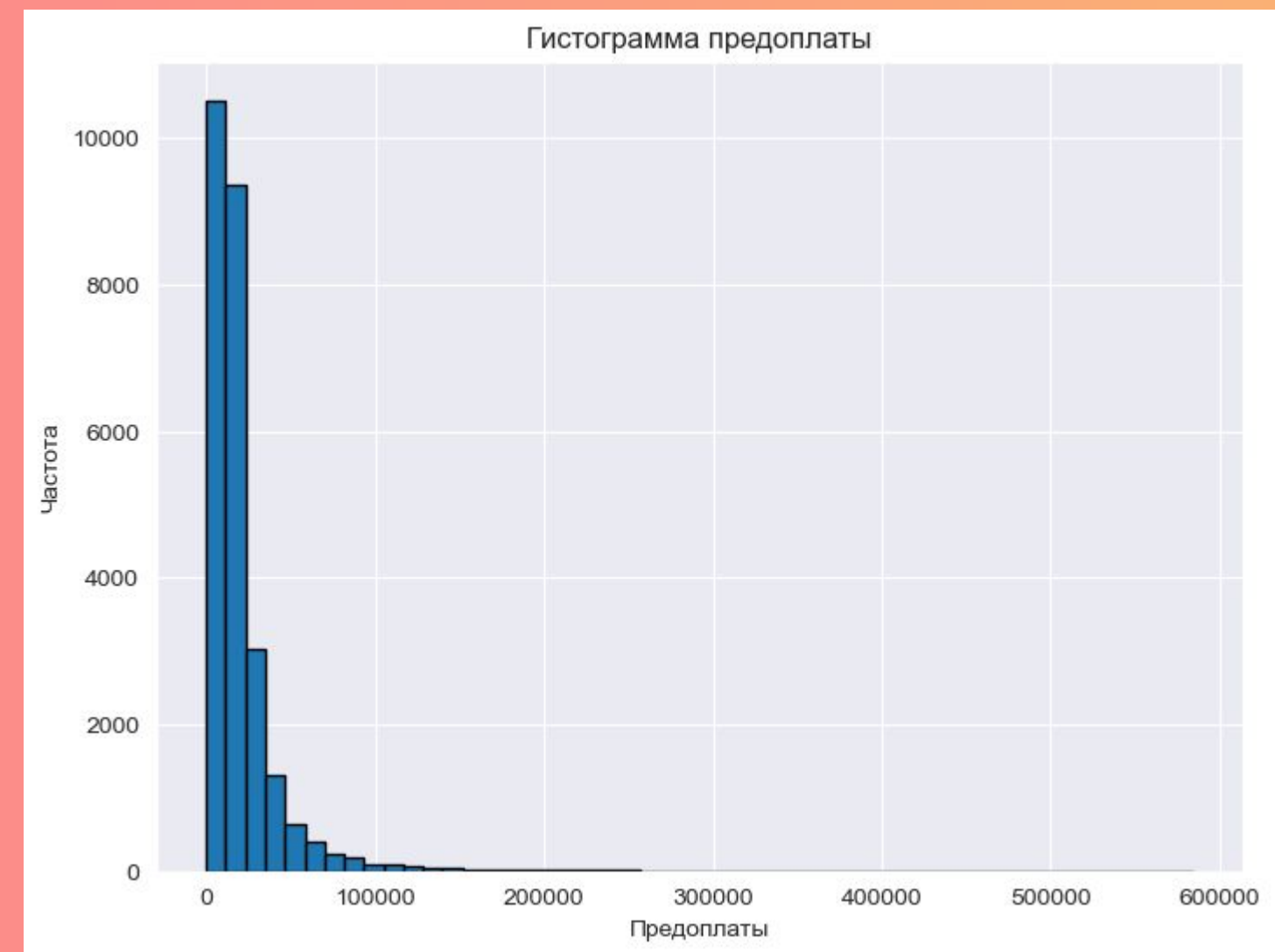
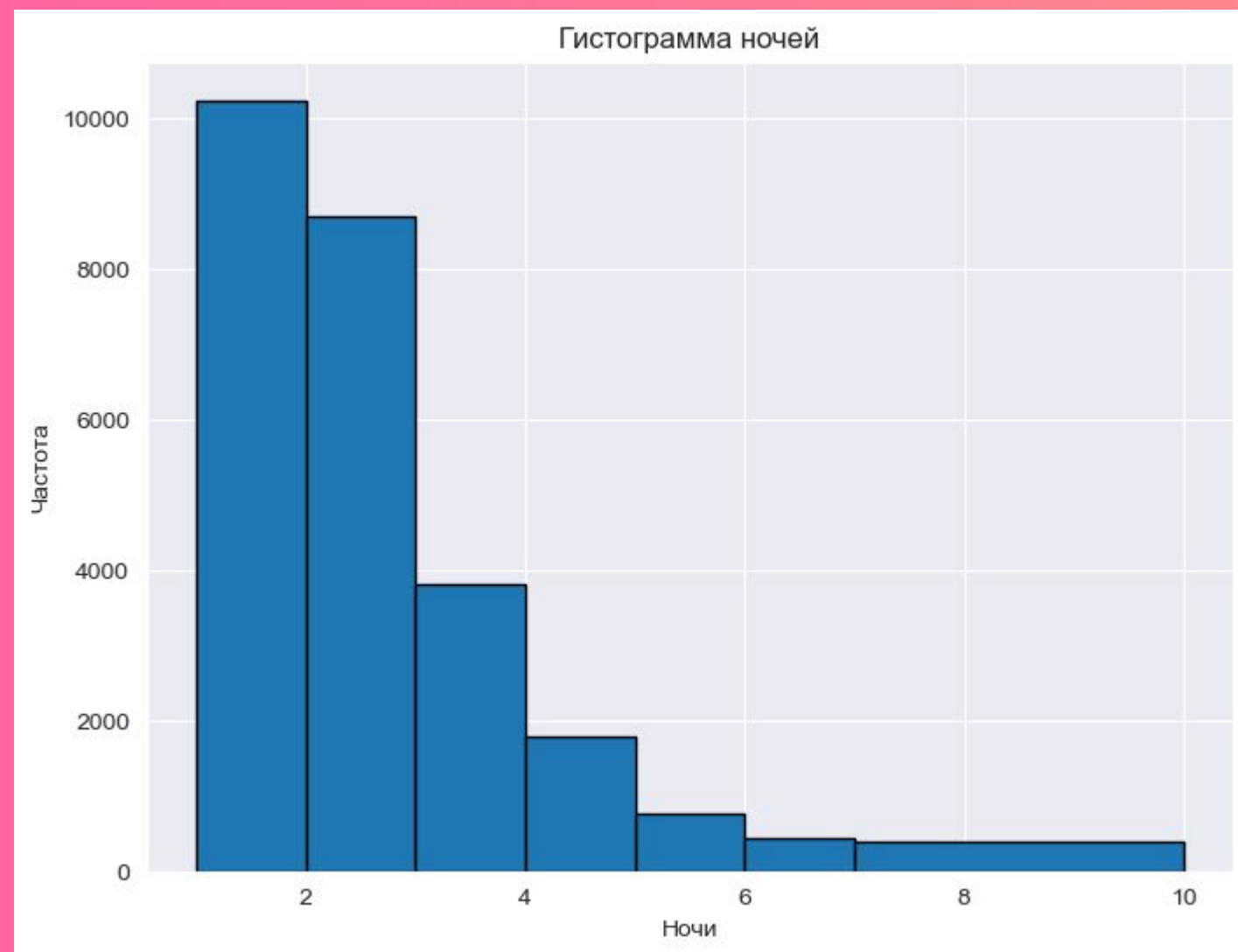
Спасибо за внимание !

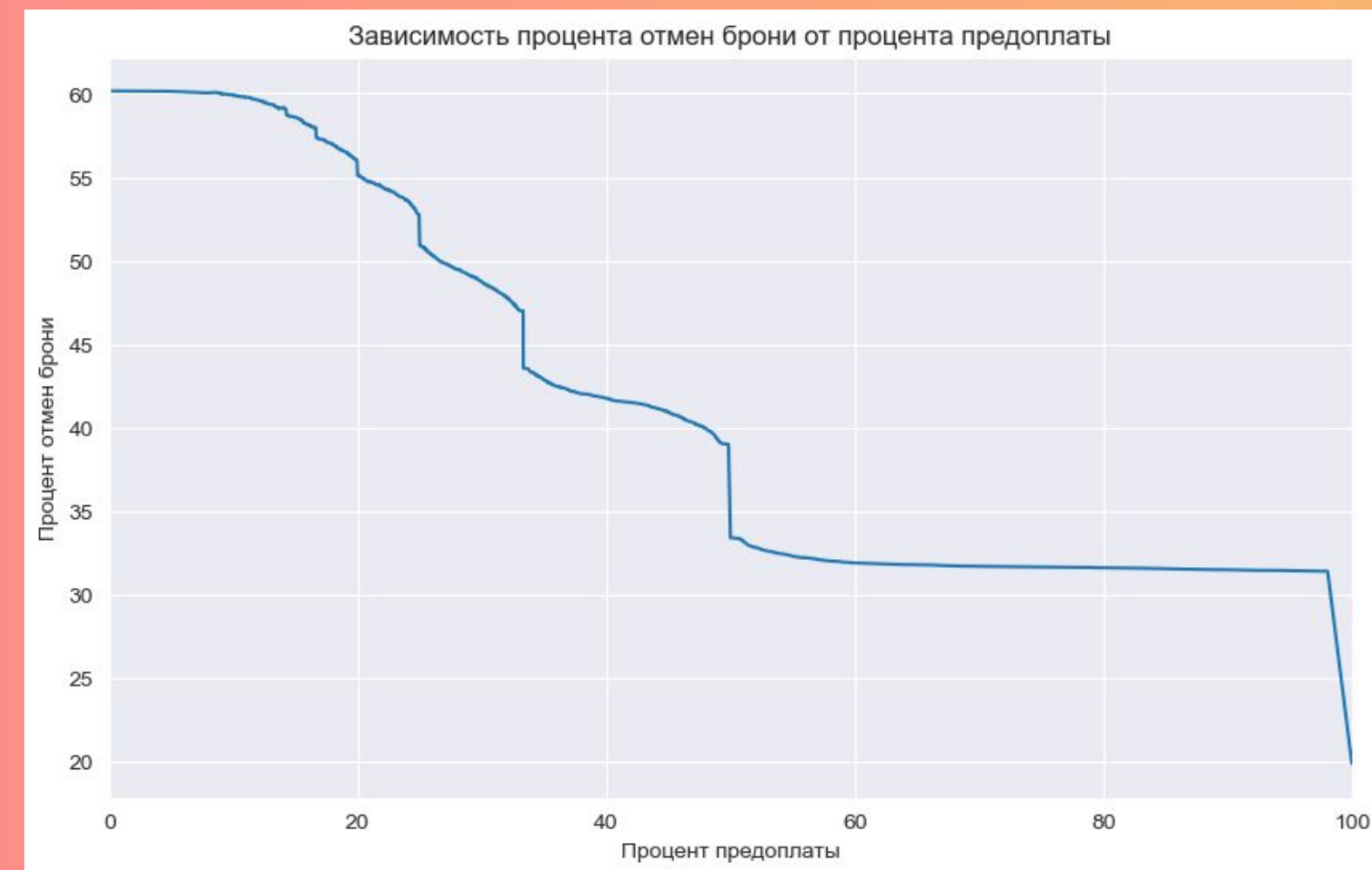
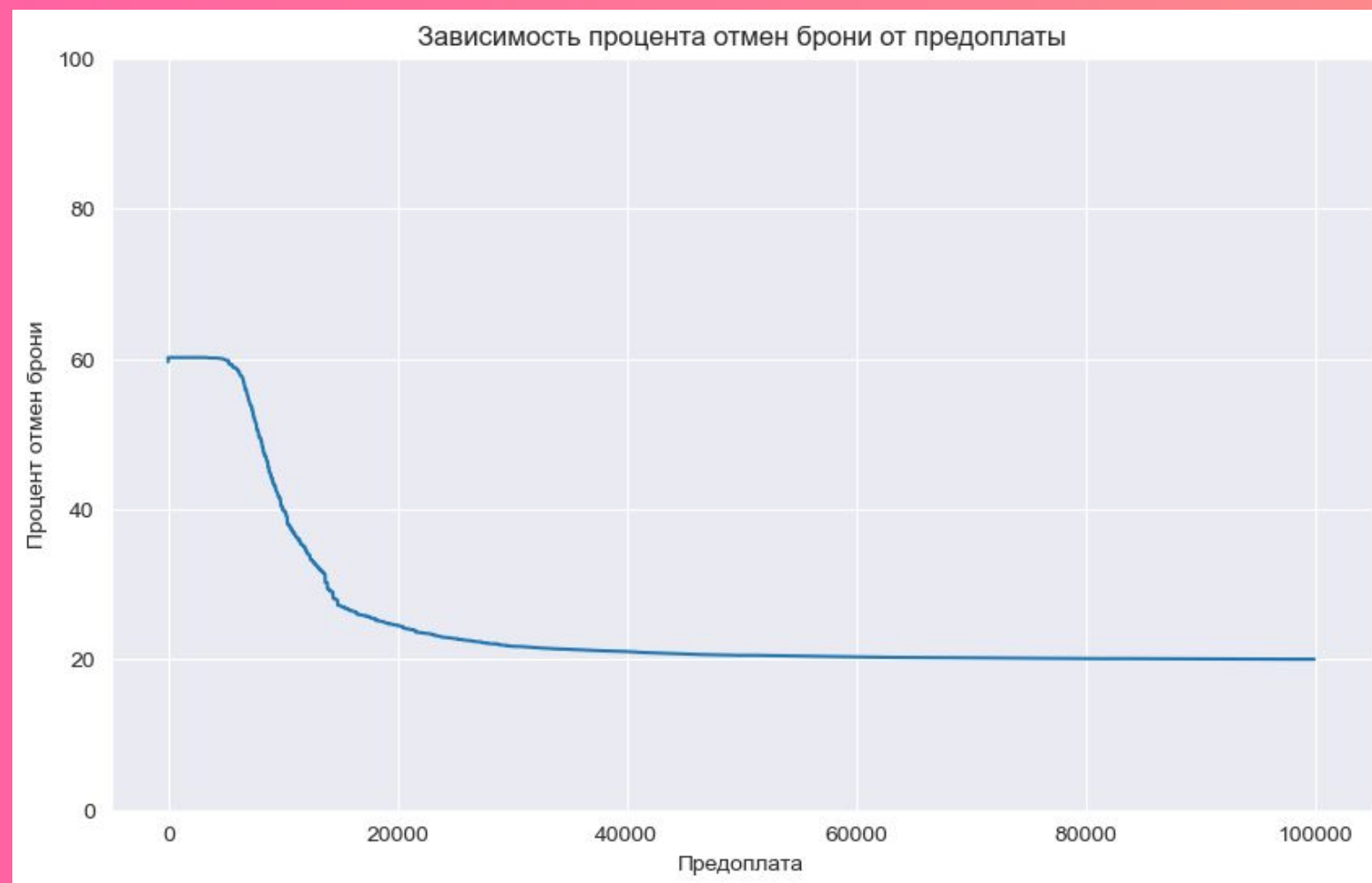


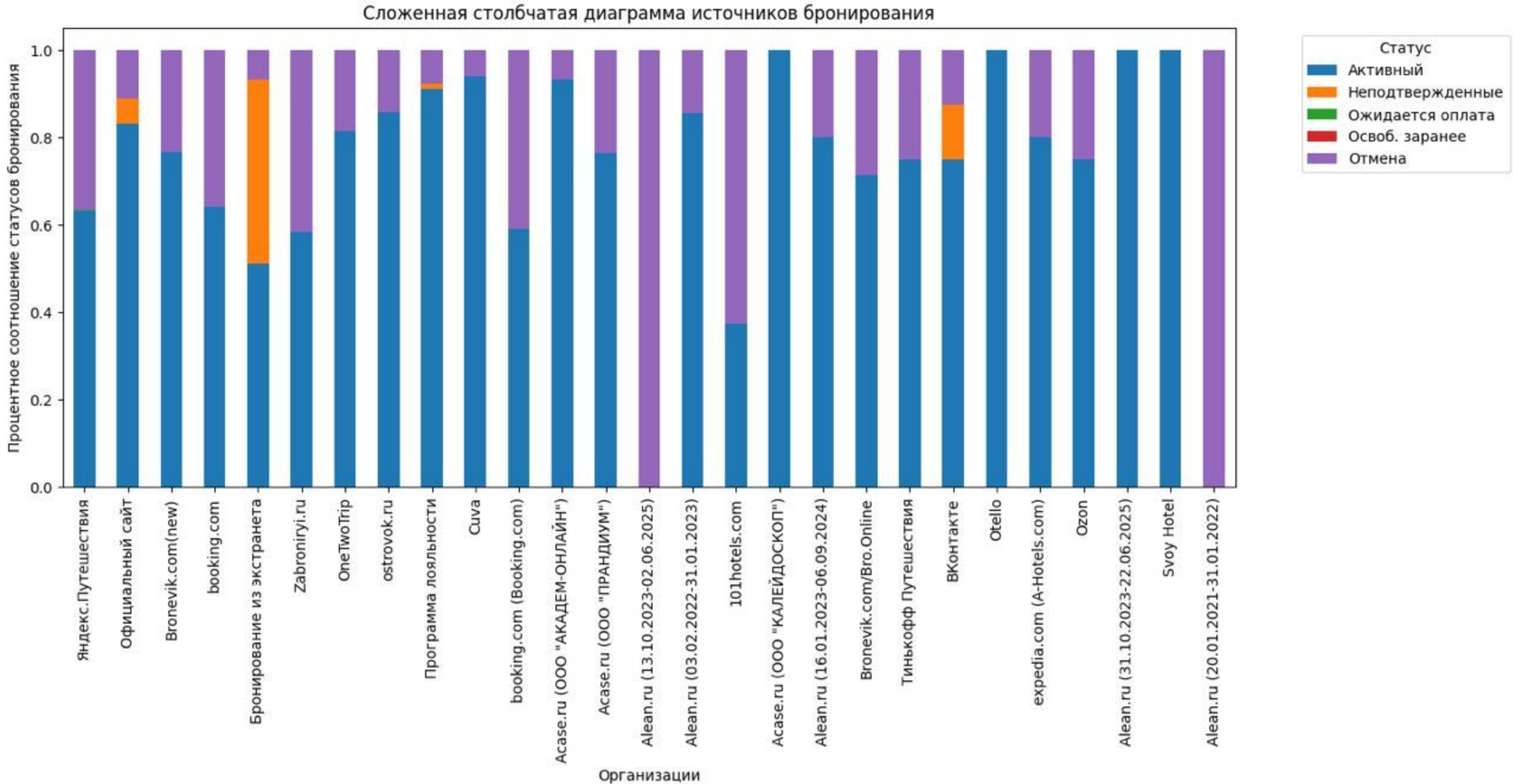


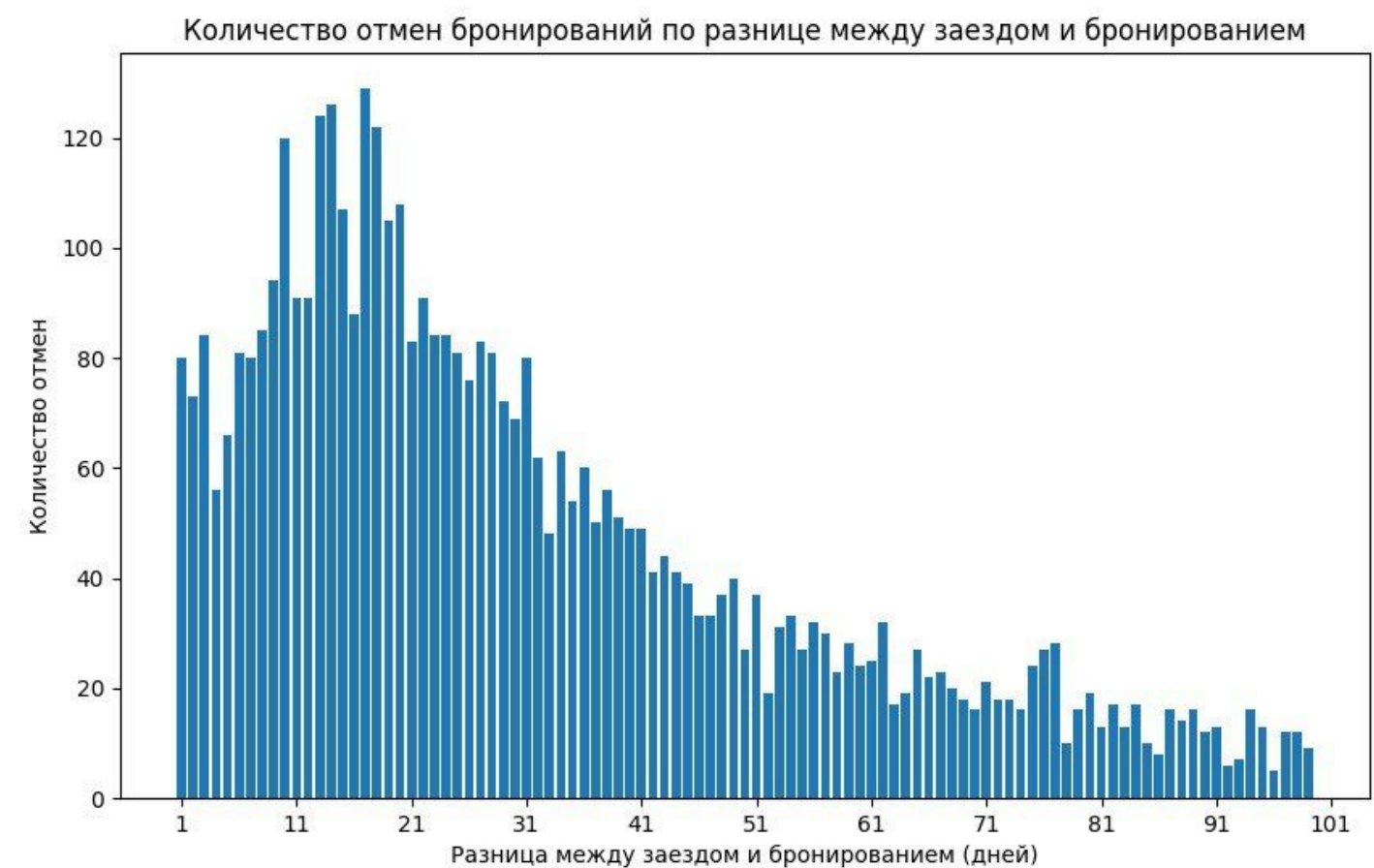
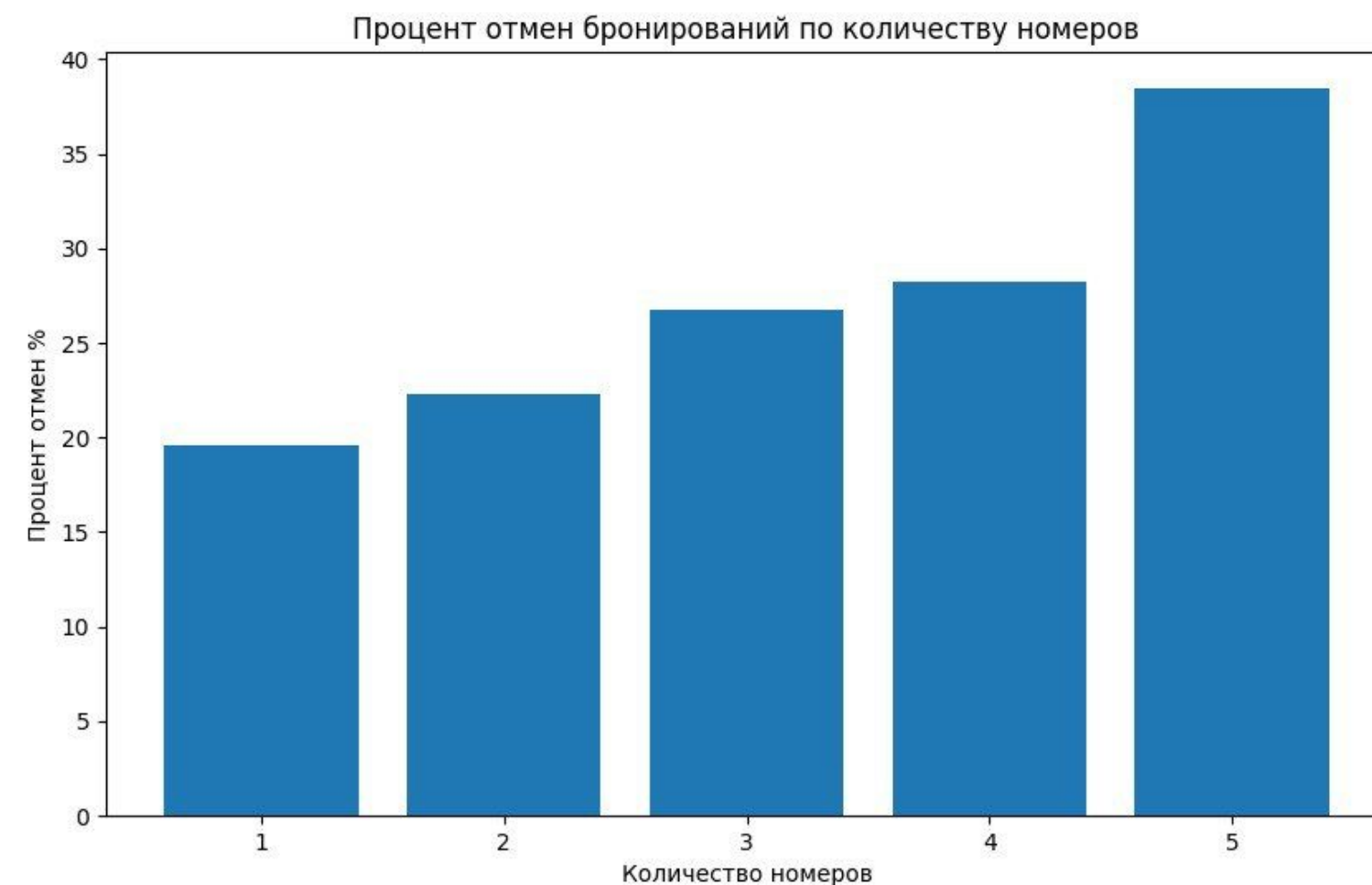













```
payment = {'Внешняя система оплаты': [83, 0, 1, 0, 248],
           'Отложенная электронная оплата: Банк Россия (банк. карта)': [1779, 1787, 3, 0, 559],
           'Банк. карта: Банк Россия (банк. карта)': [12373, 0, 0, 0, 1374],
           'Внешняя система оплаты (С предоплатой)': [1549, 0, 0, 0, 326],
           'Система быстрых платежей: Эквайринг ComfortBooking (Система быстрых платежей)': [2322, 0, 0, 0, 217],
           'Гарантия банковской картой': [545, 0, 0, 0, 331],
           'Внешняя система оплаты (Оплата наличными)': [44, 0, 0, 0, 25],
           'Банк. карта [Кешбэк. МИР]: Эквайринг TravelLine Pro (Банк. карта)': [1493, 0, 0, 2, 119],
           'При заселении': [558, 0, 0, 0, 146],
           'Банк. карта: Эквайринг ComfortBooking (Банк. карта)': [53, 0, 0, 0, 5],
           'Банк. карта [Кешбэк. МИР]: Эквайринг ComfortBooking (Банк. карта)': [141, 0, 0, 0, 29],
           'Банк. карта (SberPay): Эквайринг ComfortBooking (Банк. карта) (SberPay)': [6, 0, 0, 0, 1],
           'Внешняя система оплаты (Банковская карта)': [31, 0, 0, 0, 23],
           'Банк. карта (Yandex Pay): Эквайринг ComfortBooking (Банк. карта) (Yandex Pay)': [1, 0, 0, 0, 0]}

payment_percent = {'Внешняя система оплаты': [0.25, 0.0, 0.0030120481927710845, 0.0, 0.7469879518072289],
                  'Отложенная электронная оплата: Банк Россия (банк. карта)': [0.4309593023255814, 0.4328972868217054, 0.0007267441860465116, 0.0, 0.13541666666666666],
                  'Банк. карта: Банк Россия (банк. карта)': [0.900050920200771, 0.0, 0.0, 0.0, 0.09994907979922893],
                  'Внешняя система оплаты (С предоплатой)': [0.8261333333333334, 0.0, 0.0, 0.0, 0.17386666666666667],
                  'Система быстрых платежей: Эквайринг ComfortBooking (Система быстрых платежей)': [0.9145332808192201, 0.0, 0.0, 0.0, 0.08546671918077983],
                  'Гарантия банковской картой': [0.6221461187214612, 0.0, 0.0, 0.0, 0.3778538812785388],
                  'Внешняя система оплаты (Оплата наличными)': [0.6376811594202898, 0.0, 0.0, 0.0, 0.36231884057971014],
                  'Банк. карта [Кешбэк. МИР]: Эквайринг TravelLine Pro (Банк. карта)': [0.9250309789343246, 0.0, 0.0, 0.0012391573729863693, 0.07372986369268897],
                  'При заселении': [0.7926136363636364, 0.0, 0.0, 0.0, 0.20738636363636365],
                  'Банк. карта: Эквайринг ComfortBooking (Банк. карта)': [0.9137931034482759, 0.0, 0.0, 0.0, 0.08620689655172414],
                  'Банк. карта [Кешбэк. МИР]: Эквайринг ComfortBooking (Банк. карта)': [0.8294117647058824, 0.0, 0.0, 0.0, 0.17058823529411765],
                  'Банк. карта (SberPay): Эквайринг ComfortBooking (Банк. карта) (SberPay)': [0.8571428571428571, 0.0, 0.0, 0.0, 0.14285714285714285],
                  'Внешняя система оплаты (Банковская карта)': [0.5740740740740741, 0.0, 0.0, 0.0, 0.42592592592592593],
                  'Банк. карта (Yandex Pay): Эквайринг ComfortBooking (Банк. карта) (Yandex Pay)': [1.0, 0.0, 0.0, 0.0, 0.0]}
```

```
plt.figure(figsize=(8, 6)) # Устанавливаем размер графика
plt.plot(train_df['Стоимость'], label='Стоимость')
plt.xlabel('Индекс записи')
plt.ylabel('Стоимость')
plt.title('График стоимости')
plt.legend()
plt.show()
# Строим гистограмму с интервалами
plt.figure(figsize=(8, 6))
plt.hist(train_df['Стоимость'].clip(upper=100000), bins=50, edgecolor='black') # bins=10 - количество интервалов
plt.xlabel('Стоимость')
plt.ylabel('Частота')
plt.title('Гистограмма стоимости')
plt.show()
```

```
plt.figure(figsize=(8, 6)) # Устанавливаем размер графика
plt.plot(train_df['Внесена предоплата'], label='Стоимость')
plt.xlabel('Индекс записи')
plt.ylabel('Предоплаты')
plt.title('График предоплаты')
plt.legend()
plt.show()
# Строим гистограмму с интервалами
plt.figure(figsize=(8, 6))
plt.hist(train_df['Внесена предоплата'], bins=50, edgecolor='black') # bins=10 - количество интервалов
plt.xlabel('Предоплаты')
plt.ylabel('Частота')
plt.title('Гистограмма предоплаты')
plt.show()
```

```
plt.figure(figsize=(8, 6))
plt.hist(train_df['Внесена предоплата'].clip(upper=200000), bins=50, edgecolor='black') # bins=10 - количество интервалов
plt.xlabel('Предоплаты')
plt.ylabel('Частота')
plt.title('Гистограмма предоплаты')
plt.show()
plt.figure(figsize=(8, 6))
plt.hist(train_df['Внесена предоплата'].clip(upper=100000), bins=50, edgecolor='black') # bins=10 - количество интервалов
plt.xlabel('Предоплаты')
plt.ylabel('Частота')
plt.title('Гистограмма предоплаты')
plt.show()
```



```
train_df['Дата отмены'] = train_df['Дата отмены'].notna().astype(int)

restrictioned_data = train_df[train_df['Внесена предоплата'] < 100000]['Внесена предоплата']
prepayment_values = sorted(restrictioned_data.unique())

# Создаем список для хранения процентных значений отмен
cancellation_rates = []

# Цикл по значениям предоплаты
for value in prepayment_values:
    # Фильтруем DataFrame по предоплате не более чем value
    filtered_df = train_df[train_df['Внесена предоплата'] <= value]

    # Вычисляем процент отмен
    cancellation_rate = filtered_df['Дата отмены'].mean() * 100
    cancellation_rates.append(cancellation_rate)

# Строим график
plt.figure(figsize=(10, 6))
plt.plot(prepayment_values, cancellation_rates)

# Настройка осей и заголовка
plt.xlabel('Предоплата')
plt.ylabel('Процент отмен брони')
plt.ylim(0, 100)
plt.title('Зависимость процента отмен брони от предоплаты')

# Отображаем график
plt.show()
```



```
train_df['Процент предоплаты'] = (train_df['Внесена предоплата'] / train_df['Стоимость']) * 100

# Создаем список для хранения значений процента предоплаты
prepayment_percentages = sorted(train_df['Процент предоплаты'].unique())

# Создаем список для хранения процентных значений отмен
cancellation_rates = []

# Цикл по значениям процента предоплаты
for percentage in prepayment_percentages:
    # Фильтруем DataFrame по проценту предоплаты не более чем percentage
    filtered_df = train_df[train_df['Процент предоплаты'] <= percentage]

    # Вычисляем процент отмен
    cancellation_rate = filtered_df['Дата отмены'].mean() * 100
    cancellation_rates.append(cancellation_rate)

# Строим график
plt.figure(figsize=(10, 6))
plt.plot(prepayment_percentages, cancellation_rates)

# Настройка осей и заголовка
plt.xlabel('Процент предоплаты')
plt.ylabel('Процент отмен брони')
plt.xlim(0, 100)
plt.title('Зависимость процента отмен брони от процента предоплаты')

# Отображаем график
plt.show()
```