 **Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

**Факультет «Информатика и системы управления»**

**Кафедра ИУ5 «Системы обработки информации и управления»**

Курс «Методы машинного обучения»

# Отчет по лабораторной работе 1

Выполнил:

студент группы ИУ5-22М Зоров В..В

20.05.2023

Москва, 2023 г.

**Лабораторная работа 1. Выполнил Зоров Владислав Витальевич ИУ5-22м**

Задание: Выбрать набор данных (датасет). Для лабораторных работ не рекомендуется выбирать датасеты очень большого размера.

Создать "историю о данных" в виде юпитер-ноутбука, с учетом следующих требований:

История должна содержать не менее 5 шагов (где 5 - рекомендуемое количество шагов). Каждый шаг содержит график и его текстовую интерпретацию. На каждом шаге наряду с удачным итоговым графиком рекомендуется в юпитер-ноутбуке оставлять результаты предварительных "неудачных" графиков. Не рекомендуется повторять виды графиков, желательно создать 5 графиков различных видов. Выбор графиков должен быть обоснован использованием методологии data-to-viz. Рекомендуется учитывать типичные ошибки построения выбранного вида графика по методологии data-to-viz. Если методология Вами отвергается, то просьба обосновать Ваше решение по выбору графика. История должна содержать итоговые выводы. В реальных "историях о данных" именно эти выводы представляют собой основную ценность для предприятия. Сформировать отчет и разместить его в своем репозитории на github.

Этот датасет представляет собой CSV-файл, в котором каждая строка соответствует твиту. Различные столбцы описаны ниже. Каждый диалог включает по крайней мере один запрос от потребителя и по крайней мере один ответ от компании. Идентификаторы пользователей, которые представляют компании, можно определить по полю "inbound".

tweet\_id Уникальный, анонимизированный ID для твита. Используется в столбцах "response\_tweet\_id" и "in\_response\_to\_tweet\_id".

author\_id Уникальный, анонимизированный ID пользователя. Упоминания "@" в датасете были заменены на соответствующие анонимизированные идентификаторы пользователей.

inbound Определяет, является ли твит "входящим" (inbound) для компании, которая занимается обслуживанием клиентов в Twitter. Эта функция полезна при переорганизации данных для обучения моделей общения.

created\_at Дата и время отправки твита.

text Содержание твита. Чувствительная информация, такая как номера телефонов и адреса электронной почты, заменена на значения-заменители типа email.

response\_tweet\_id ID твитов, которые являются ответами на этот твит, перечислены через запятую.

in\_response\_to\_tweet\_id ID твита, на который был дан ответ в этом твите, если такой ответ был. нужно в юпитер ноутбук проанализировать данный датасет и построить разные графики в нем.

In [1]:

**import** pandas **as** pd

**import** matplotlib.pyplot **as** plt

**import** numpy **as** np

**import** seaborn **as** sns

**%matplotlib** inline

In [2]:

*# Загрузка данных из csv-файла*

dataset **=** pd**.**read\_csv(r'C:\Users\User\Desktop\маг 2сем\гапан\1лаба\twitter.csv')

dataset**.**head()

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Out[2]: | **tweet\_id** | **author\_id** | **inbound** | **created\_at** | **text** | **response\_tweet\_id** | **in\_response\_to\_tweet\_id** |
|  | **0** 1 | sprintcare | False | Tue Oct 31  22:10:47 +0000 | @115712 I  understand. I would | 2 | 3.0 |
|  |  |  |  | 2017 | like to assist y... |  |  |
|  | **1** 2 | 115712 | True | Tue Oct 31  22:11:45 +0000 | @sprintcare and how do you propose we | NaN | 1.0 |
|  |  |  |  | 2017 | do that |  |  |
|  | **2** 3 | 115712 | True | Tue Oct 31  22:08:27 +0000 | @sprintcare I have sent several private | 1 | 4.0 |
|  |  |  |  | 2017 | messag... |  |  |
|  | **3** 4 | sprintcare | False | Tue Oct 31  21:54:49 +0000 | @115712 Please send us a Private Message | 3 | 5.0 |
|  |  |  |  | 2017 | so th... |  |  |
|  | **4** 5 | 115712 | True | Tue Oct 31  21:49:35 +0000 | @sprintcare I did. | 4 | 6.0 |
|  |  |  |  | 2017 |  |  |  |

In [3]:

Out[3]:

In [4]:

*# выборка запросов*

reqs **=** dataset**.**loc[dataset["inbound"] **== True**]**.**sample(n**=**5000) reqs**.**shape

(2811774, 7)

dataset**.**shape

Out[4]:

In [5]:

*# выборка ответов на запросы*

resps **=** dataset**.**loc[dataset["in\_response\_to\_tweet\_id"]**.**isin(reqs["tweet\_id"])] resps**.**shape

(5000, 7)

Out[5]:

In [6]:

(4632, 7)

Out[6]:

**tweet\_id author\_id inbound created\_at text response\_tweet\_id in\_response\_to\_tweet\_id**

resps**.**head()

@115887 Hmm.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **540** | 848 | SpotifyCares | False | Tue Oct 31  22:28:16 |
|  |  |  |  | +0000 2017 |
| **658** | 1198 | VirginTrains | False | Tue Oct 31  22:15:38 |
|  |  |  |  | +0000 2017 |
| **1268** | 1864 | Uber\_Support | False | Fri Dec 01  06:47:07 |
|  |  |  |  |  | +0000 2017 |

Can you try restarting your

devic...

@115923 We

don't have that information

here I'...

@116127 Here to help! Please DM (direct messag...

849 850.0

NaN 1199.0

1865 1866.0

In [7]:

*# отсеем строки без ответов*

reqs **=** reqs[reqs["tweet\_id"]**.**isin(resps["in\_response\_to\_tweet\_id"])] reqs**.**shape

**tweet\_id author\_id inbound created\_at text response\_tweet\_id in\_response\_to\_tweet\_id**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1379** | 2041 | VirginTrains | False | Tue Oct 31 @116159  22:29:57 Happy the team NaN 2042.0 |
|  |  |  |  | +0000 2017 could help 😊  ^CB |
| **1412** | 2101 | sprintcare | False | Tue Oct 31 @116169 Yes  22:38:41 dm please. NaN 2102.0 |
|  |  |  |  | +0000 2017 |

Out[7]:

In [8]:

(4192, 7)

Out[8]:

**tweet\_id author\_id inbound created\_at text response\_tweet\_id in\_response\_to\_tweet\_id**

reqs**.**head()

@GloCare

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **387863** | 442631 | 219927 | True | Tue Oct 31  18:32:27 |
|  |  |  |  | +0000 2017 |
| **2474656** | 2644336 | 746110 | True | Fri Nov 17  21:09:21 |
|  |  |  |  | +0000 2017 |
| **1819606** | 1975291 | 584959 | True | Wed Nov 01  14:45:24 |
|  |  |  |  | +0000 2017 |
| **2000589** | 2157791 | 633845 | True | Thu Nov 09  03:06:24 |
|  |  |  |  | +0000 2017 |
| **1868904** | 2025121 | 598894 | True | Mon Nov 06  13:47:54 |
|  |  |  |  |  | +0000 2017 |

Please DO me, I

need your assistance.

Dude at @AlaskaAir just offered anyone

in a #3...

@116062

@116062 put it in my cart and when I h...

@Delta Help! Really need to change an award ti...

@115858

&amp; a piece of shit software

that yo...

442629 NaN

2644335 NaN

1975289 1975292.0

2157789 NaN

2025120 2025122.0

In [16]:

*# диаграмма рассеяния*

merged\_df1 **=** pd**.**merge(reqs, resps, left\_on**=**'tweet\_id', right\_on**=**'in\_response\_to\_tweet\_id'

*# fig, ax = plt.subplots(figsize=(10,10))*

*# sns.scatterplot(ax=ax, x='created\_at\_req', y='created\_at\_resp', data=merged\_df1)*

merged\_df1['created\_at\_req'] **=** pd**.**to\_datetime(merged\_df1['created\_at\_req']) merged\_df1['created\_at\_resp'] **=** pd**.**to\_datetime(merged\_df1['created\_at\_resp'])

fig, ax **=** plt**.**subplots(figsize**=**(10, 10))

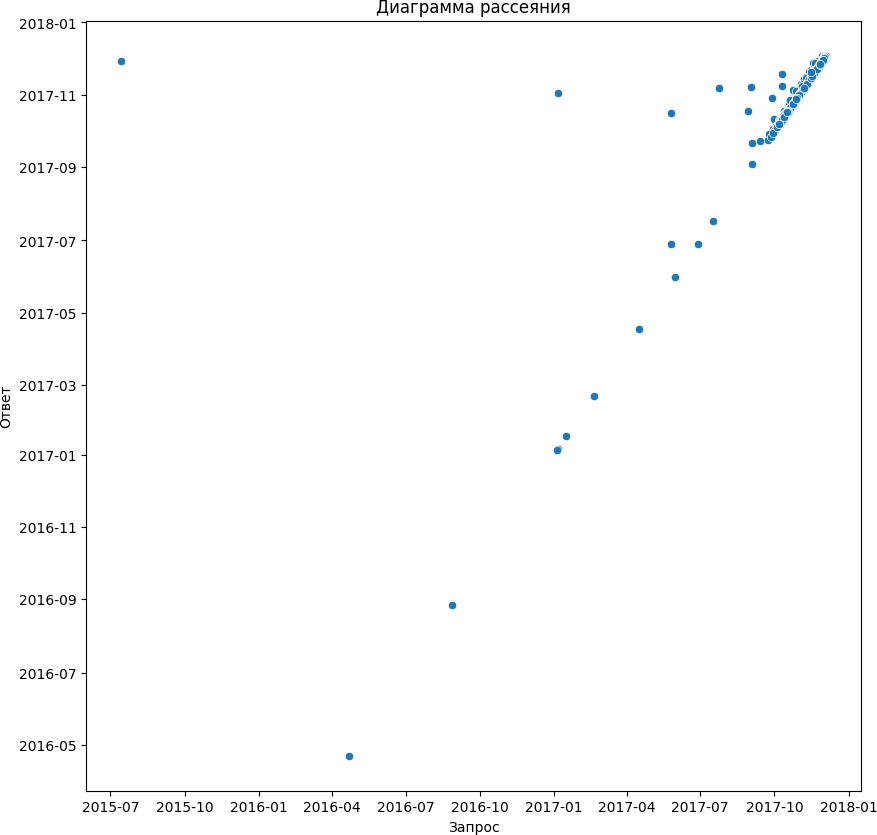
sns**.**scatterplot(ax**=**ax, x**=**'created\_at\_req', y**=**'created\_at\_resp', data**=**merged\_df1)

ax**.**set\_title('Диаграмма рассеяния') ax**.**set\_xlabel('Запрос') ax**.**set\_ylabel('Ответ')

Out[16]:

In [22]:

Text(0, 0.5, 'Ответ')



Как множно заметить большая часть данных приходится на промежуток времени в конце 2017 года. Ограничим выборку.

reqs **=** reqs**.**loc[(pd**.**to\_datetime(reqs["created\_at"]) **>=** '2017-10-01')]

resps **=** resps**.**loc[(resps["in\_response\_to\_tweet\_id"]**.**isin(reqs["tweet\_id"])) **&** (pd**.**to\_date reqs **=** reqs[reqs["tweet\_id"]**.**isin(resps["in\_response\_to\_tweet\_id"])]

reqs**.**shape, resps**.**shape

*# можно видеть, что большая часть данных сохранилась даже при таком ограничении выборки*

Out[22]:

In [24]:

((4153, 7), (4592, 7))

merged\_df1 **=** pd**.**merge(reqs, resps, left\_on**=**'tweet\_id', right\_on**=**'in\_response\_to\_tweet\_id' merged\_df1['created\_at\_req'] **=** pd**.**to\_datetime(merged\_df1['created\_at\_req']) merged\_df1['created\_at\_resp'] **=** pd**.**to\_datetime(merged\_df1['created\_at\_resp'])

fig, ax **=** plt**.**subplots(figsize**=**(10, 10))

sns**.**scatterplot(ax**=**ax, x**=**'created\_at\_req', y**=**'created\_at\_resp', data**=**merged\_df1)

ax**.**set\_title('Диаграмма рассеяния') ax**.**set\_xlabel('Запрос') ax**.**set\_ylabel('Ответ')

ax**.**set\_xticklabels(ax**.**get\_xticklabels(), rotation**=**45)

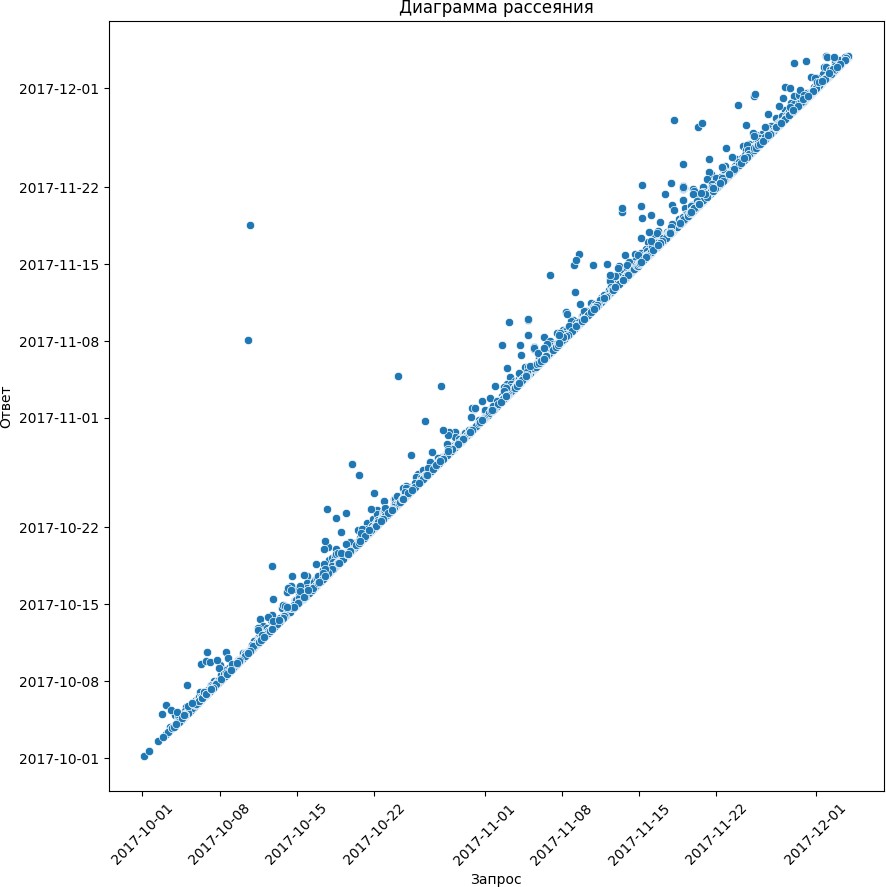
Out[24]:

In [28]:

C:\Users\User\AppData\Local\Temp\ipykernel\_1732\2100163312.py:11: UserWarning: FixedFormat ter should only be used together with FixedLocator

ax.set\_xticklabels(ax.get\_xticklabels(), rotation=45)

|  |  |  |
| --- | --- | --- |
| [Text(17440.0, | 0, | '2017-10-01'), |
| Text(17447.0, | 0, | '2017-10-08'), |
| Text(17454.0, | 0, | '2017-10-15'), |
| Text(17461.0, | 0, | '2017-10-22'), |
| Text(17471.0, | 0, | '2017-11-01'), |
| Text(17478.0, | 0, | '2017-11-08'), |
| Text(17485.0, | 0, | '2017-11-15'), |
| Text(17492.0, | 0, | '2017-11-22'), |
| Text(17501.0, | 0, | '2017-12-01')] |



merged\_df1['response\_time'] **=** (merged\_df1['created\_at\_resp'] **-** merged\_df1['created\_at\_req

avg\_response\_time **=** merged\_df1['response\_time']**.**mean()**//**60**//**60 print(f"Совокупное среднее время ответа: {avg\_response\_time} часа(-ов)")

Совокупное среднее время ответа: 4.0 часа(-ов)

На данной диаграмме рассеяния видно, что большая часть времени ответов тяготеет к отношению X=Y к времени запросов. Чем ближе точка к такому соотношению, тем быстрее пришел ответ на запрос. При этом точки не могут располагаться ниже x=y, т.к. ответ не мог прийти раншье запроса, что и видно по графику. Среднее совокупное время ответов равно примерно 4 часам.

In [122…

*# merged\_df1['created\_at\_req'] = pd.to\_datetime(merged\_df1['created\_at\_req']).astype(int) # merged\_df1['created\_at\_resp'] = pd.to\_datetime(merged\_df1['created\_at\_resp']).astype(in* g **=** sns**.**jointplot(x**=**"created\_at\_req", y**=**"created\_at\_resp", data**=**merged\_df1, kind**=**"hex", g

g**.**ax\_marg\_x**.**clear()

g**.**ax\_marg\_x**.**hist(merged\_df1["created\_at\_req"], alpha**=**0.8, edgecolor**=**"white", bins**=**50)

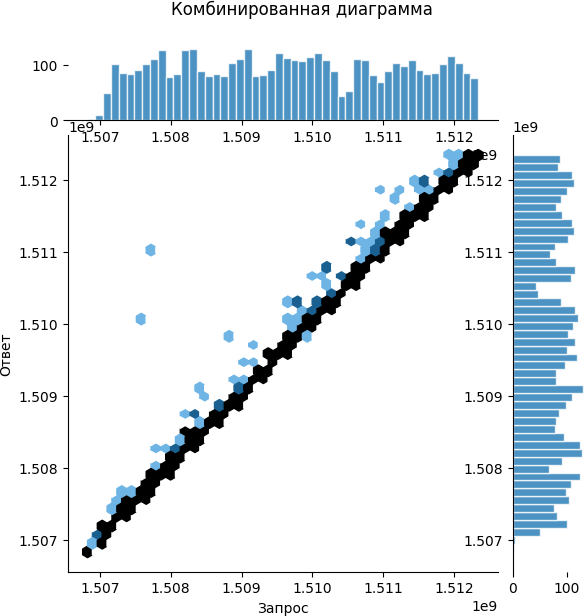
g**.**ax\_marg\_y**.**clear()

g**.**ax\_marg\_y**.**hist(merged\_df1["created\_at\_resp"], alpha**=**0.8, edgecolor**=**"white", orientation**=**"horizontal", bins**=**50)

g**.**set\_axis\_labels("Запрос", "Ответ") g**.**fig**.**suptitle("Комбинированная диаграмма", y**=**1.05)

Out[122…

Text(0.5, 1.05, 'Комбинированная диаграмма')



По гистограммам на краях можно проследить волнообразный, возможно циклический характер распределения количества запросов и ответов во времени. Россмотрим это более подробно.

In [150…

sns**.**violinplot(x**=**reqs['created\_at']**.**dt**.**week)

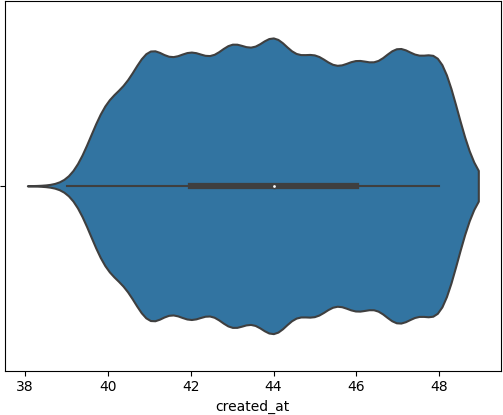
*# запросы - рапределение по неделям*

Out[150…

C:\Users\User\AppData\Local\Temp\ipykernel\_1732\2152118459.py:1: FutureWarning: Series.dt. weekofyear and Series.dt.week have been deprecated. Please use Series.dt.isocalendar().wee k instead.

sns.violinplot(x=reqs['created\_at'].dt.week)

<Axes: xlabel='created\_at'>



In [143…

sns**.**violinplot(x**=**resps['created\_at']**.**dt**.**week, color **=** "orange")

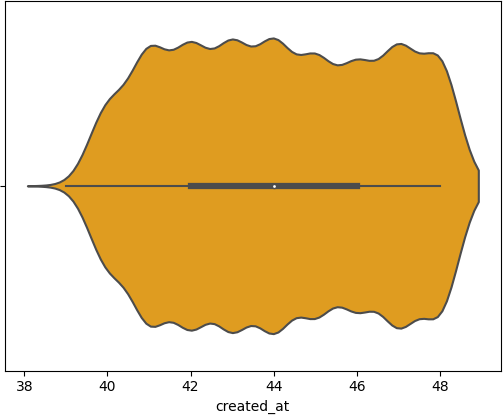
*# ответы - рапределение по неделям*

Out[143…

C:\Users\User\AppData\Local\Temp\ipykernel\_1732\3242814907.py:1: FutureWarning: Series.dt. weekofyear and Series.dt.week have been deprecated. Please use Series.dt.isocalendar().wee k instead.

sns.violinplot(x=resps['created\_at'].dt.week, color = "orange")

<Axes: xlabel='created\_at'>



In [138…

sns**.**violinplot(x**=**reqs['created\_at']**.**dt**.**hour)

*# запросы по дням*

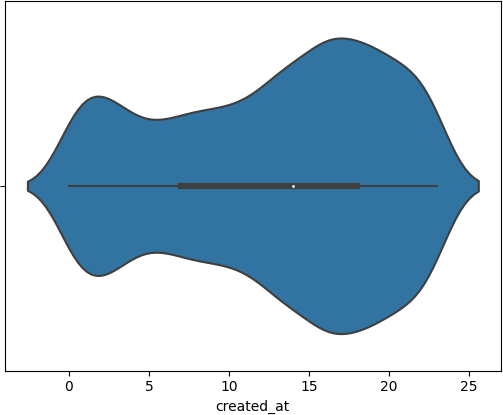
Out[138…

In [149…

sns**.**violinplot(x**=**resps['created\_at']**.**dt**.**hour, color**=**"orange")

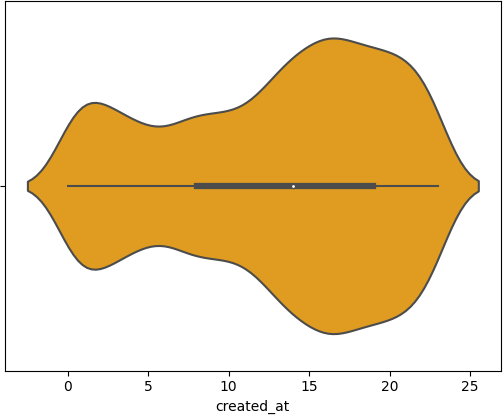
*# ответы по дням*

<Axes: xlabel='created\_at'>



Out[149…

<Axes: xlabel='created\_at'>



In [162…

fig, ax **=** plt**.**subplots(figsize**=**(10,10)) sns**.**distplot(reqs['created\_at']**.**dt**.**hour, bins**=**24) plt**.**title("Распределение запросов в сутки") plt**.**xlabel("часов в сутки")

plt**.**ylabel("Число запросов")

C:\Users\User\AppData\Local\Temp\ipykernel\_1732\1844844031.py:2: UserWarning:

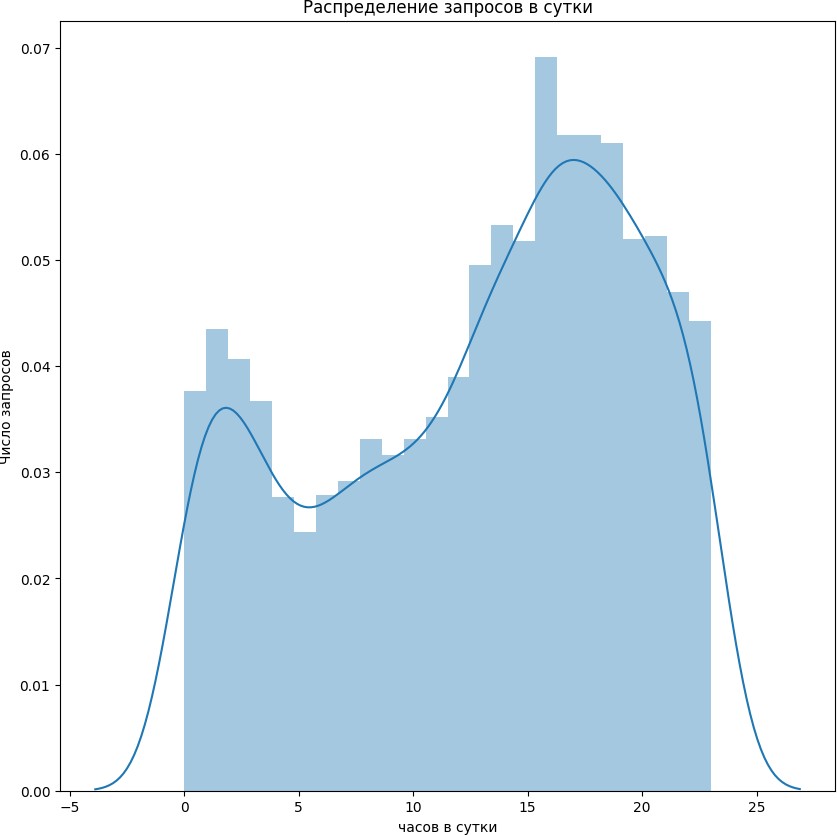
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

Out[162…

sns.distplot(reqs['created\_at'].dt.hour, bins=24) Text(0, 0.5, 'Число запросов')



In [164…

fig, ax **=** plt**.**subplots(figsize**=**(10,10)) sns**.**distplot(resps['created\_at']**.**dt**.**hour, bins**=**24,color **=** "orange") plt**.**title("Распределение ответов в сутки")

plt**.**xlabel("часов в сутки") plt**.**ylabel("Число запросов")

C:\Users\User\AppData\Local\Temp\ipykernel\_1732\1415242359.py:2: UserWarning:

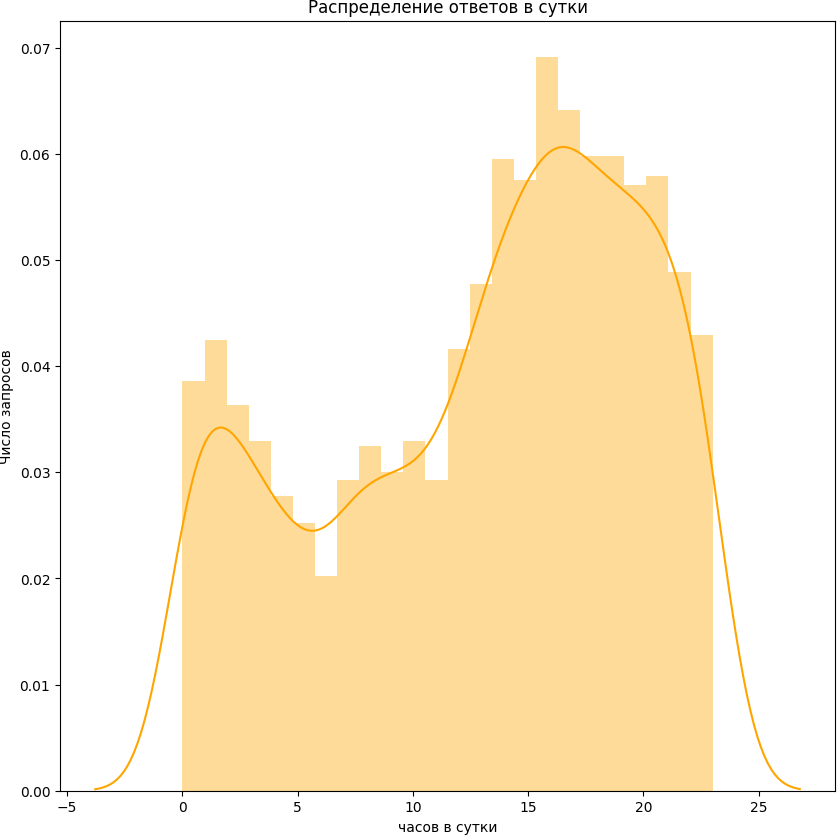
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

Out[164…

sns.distplot(resps['created\_at'].dt.hour, bins=24,color = "orange") Text(0, 0.5, 'Число запросов')



По данным гистограммам можно проанализировать распределение совокупного количества заявок или ответов по часам в течение дня.

In [183…

num\_reqs **=** len(reqs) num\_resps **=** len(resps)

values **=** [num\_reqs, num\_resps] labels **=** ['Запросы', 'Ответы'] fig, ax **=** plt**.**subplots()

ax**.**pie(values, labels**=**labels, autopct**=**'%1.1f%%', wedgeprops**=**{'alpha': 0.5})

ax**.**set\_title('Запросы и Ответы')

plt**.**show()

3a1npOCbl **l4** OTB,eTlbl

3anp0Cbl



**47.5%**



52.5%

On:ieT!bll