

DAN 

Задачный тур третьего (заключительного)
этапа 2022/23 уч.года.

Задача 4.

Все баллы умножаем на 4

4. Трясем Землю (40 баллов)

Вам дан [файл](#), в котором собраны все землетрясения за некий период.

Описание колонок:

date	Дата, в которую произошло землетрясение
time	Время, в которое произошло землетрясение
latitude	Широта
longitude	Долгота
depth	Глубина, метров
mag	Магнитуда землетрясения (~сила)
magtype	Метод, использованный для подсчета магнитуды
nst	Количество сейсмических станций, использованных для определения места землетрясения
net	Название системы улавливания землетрясений, которая зафиксировала событие
place	Текстовое описание места землетрясения
type	Тип события

(i). Заполните таблицу.

Дата, в которую было максимальное количество землетрясений	2011-03-11 <code>df.date.value_counts().head(1)</code>
Средняя глубина землетрясения. Округлите до 3 знаков после запятой	76.856 <code>df.depth.mean().round(3)</code>
Средняя глубина землетрясения после 01.01.1917. Округлите до 3 знаков после запятой	77.647 <code>df[df.date >= '1917-01-01'].depth.mean().round(3)</code>

25-й перцентиль магнитуды землетрясения	6.2 df.mag.quantile(0.25)
42-й перцентиль магнитуды землетрясения	6.4 df.mag.quantile(0.42)
Система улавливания землетрясений, у которой был наибольший абсолютный прирост зарегистрированных землетрясений за период с 2000 года по сравнению с периодом до 2000 года ¹ . Напишите название этой системы как в датасете, в скобочках укажите этот прирост (в землетрясениях)	pde (855) old_eq = df[df.date < '2000-01-01'].groupby('net').net.count() new_eq = df[df.date >= '2000-01-01'].groupby('net').net.count() (new_eq - old_eq).dropna().sort_values(ascending=False)
Система улавливания землетрясений, у которой был наибольший относительный прирост зарегистрированных землетрясений за период с 2000 года по сравнению с периодом до 2000 года ¹ . Напишите название этой системы как в датасете, в скобочках укажите этот прирост (в процентах)	pde (846.535%) old_eq = df[df.date < '2000-01-01'].groupby('net').net.count() new_eq = df[df.date >= '2000-01-01'].groupby('net').net.count() ((new_eq - old_eq) * 100 / old_eq).dropna().sort_values(ascending=False)

Критерии:

Мах – **28 баллов**

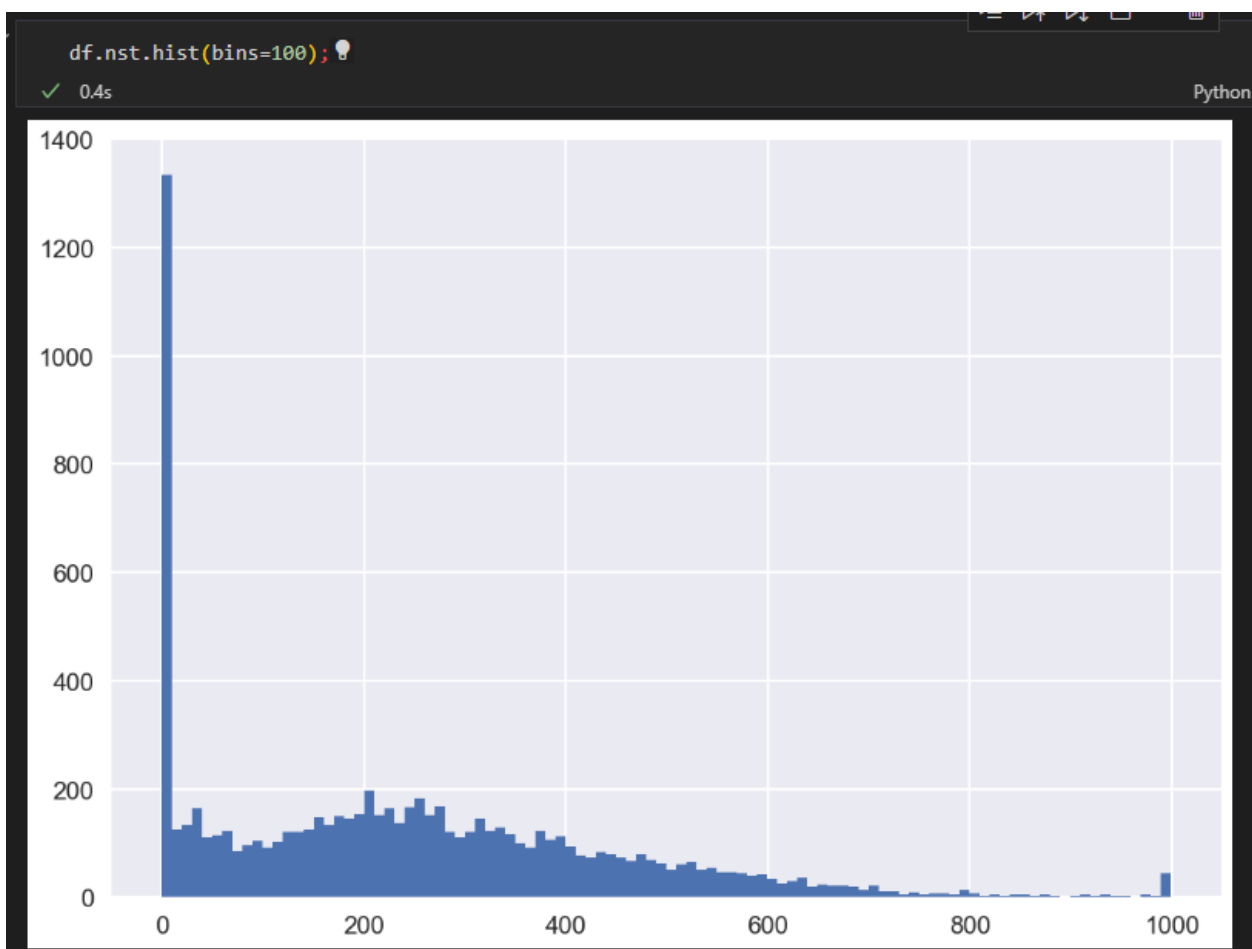
По **4 балла** за каждую из 7 ячеек

- **1 балл**, если ошибка округления в пункте 2 и 3

(ii). Посмотрите внимательнее на переменную nst. Есть ли там подозрительные значения, которые могли бы быть ошибкой в сборе или обработке данных? Прокомментируйте их и причины их появления. Если нужно, проведите дополнительный анализ, не используя данные вне предоставленного файла.

Решение:

Посмотрим на распределение nst:



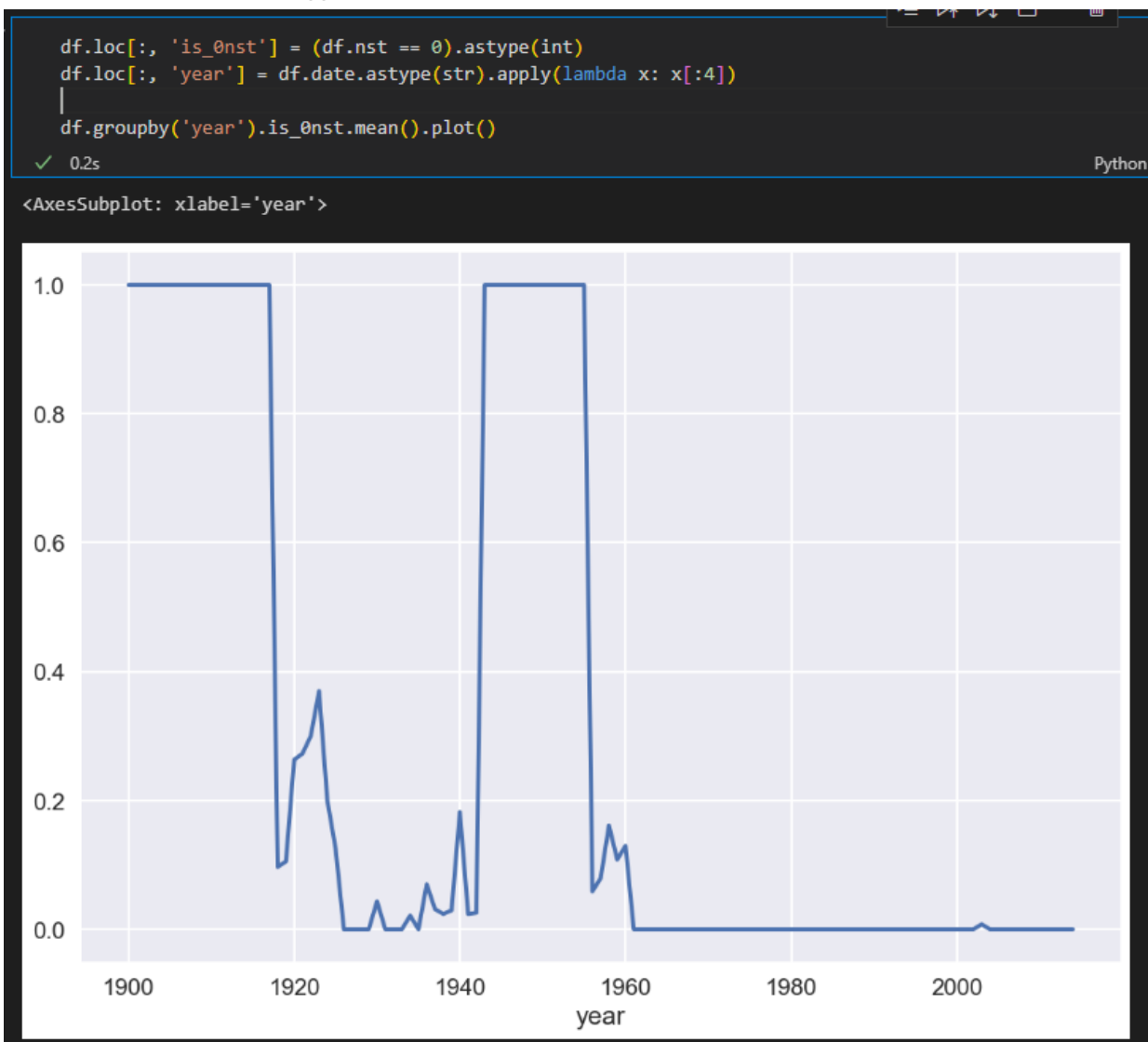
Видны подозрительные пики в 0 и 1000. Посмотрим на них внимательнее:

```
df.nst.value_counts().head(20)
✓ 0.1s
```

0.0	1283
999.0	43
203.0	27
206.0	26
208.0	26
256.0	26
245.0	24
317.0	23
229.0	23
196.0	22
214.0	22
252.0	22
276.0	21
34.0	21
155.0	21
213.0	21
241.0	21
271.0	21
223.0	21
253.0	20

Слишком много наблюдений с 0 станций и с 999 станциями. Если наблюдения с 0 станций действительно правдивые, то их доля должна падать со временем, так как станций

становится больше. Это действительно так:



Значит, скорее всего, значение 0 в nst отображает действительность.

Наблюдения с 999 станциями распределены в 2002 -2005 году и сделаны одной станцией. Результатом таких наблюдений может быть как предварительная обработка, так и ограничения фиксирующей землетрясения системы

Критерии:

Мах – **24 балла**

Нахождение 0 и 999 – по **4 балла**

Аргументация 0 – **12 баллов**

Аргументация 999 – **4 балла**

(iii). Найдите самое популярное значение в колонке place. Значит ли это, что в данном регионе было больше всего землетрясений в указанный период? *Подсказка: посмотрите на похожие названия регионов*

Решение:

```
df.place.value_counts().head(5)
✓ 0.3s
```

Vanuatu	349
Kuril Islands	224
New Britain region, Papua New Guinea	195
Fiji region	190
Solomon Islands	183

Name: place, dtype: int64

Самое популярное значение – Vanuatu.

Давайте посмотрим на похожие названия:

```
df[(df.place.str.lower().str.contains('vanuatu')) & ~ df.place.isna()].place
✓ 0.1s
```

8	262km ESE of Sola, Vanuatu
15	34km E of Port-Olry, Vanuatu
24	32km W of Sola, Vanuatu
85	58km ENE of Luganville, Vanuatu
128	53km NNE of Isangel, Vanuatu
	...
8043	Vanuatu
8054	Vanuatu
8098	Vanuatu
8222	Vanuatu region
8225	Vanuatu

Name: place, Length: 381, dtype: object

Видно, что названия не нормализованы – то есть, землетрясение может произойти в Vanuatu, но это не значит, что в place будет записано именно Vanuatu. Это повлияет на результат нашего анализа. Так, самый популярным регионом может быть Kuril Islands, которые в половине случаев будут записаны по-другому.

Критерии:

Мах – **16 баллов**

Самое популярное значение – **4 балла**

Найдено, что названия не нормализованы (есть разные написания одного и того же региона) - **8 баллов**

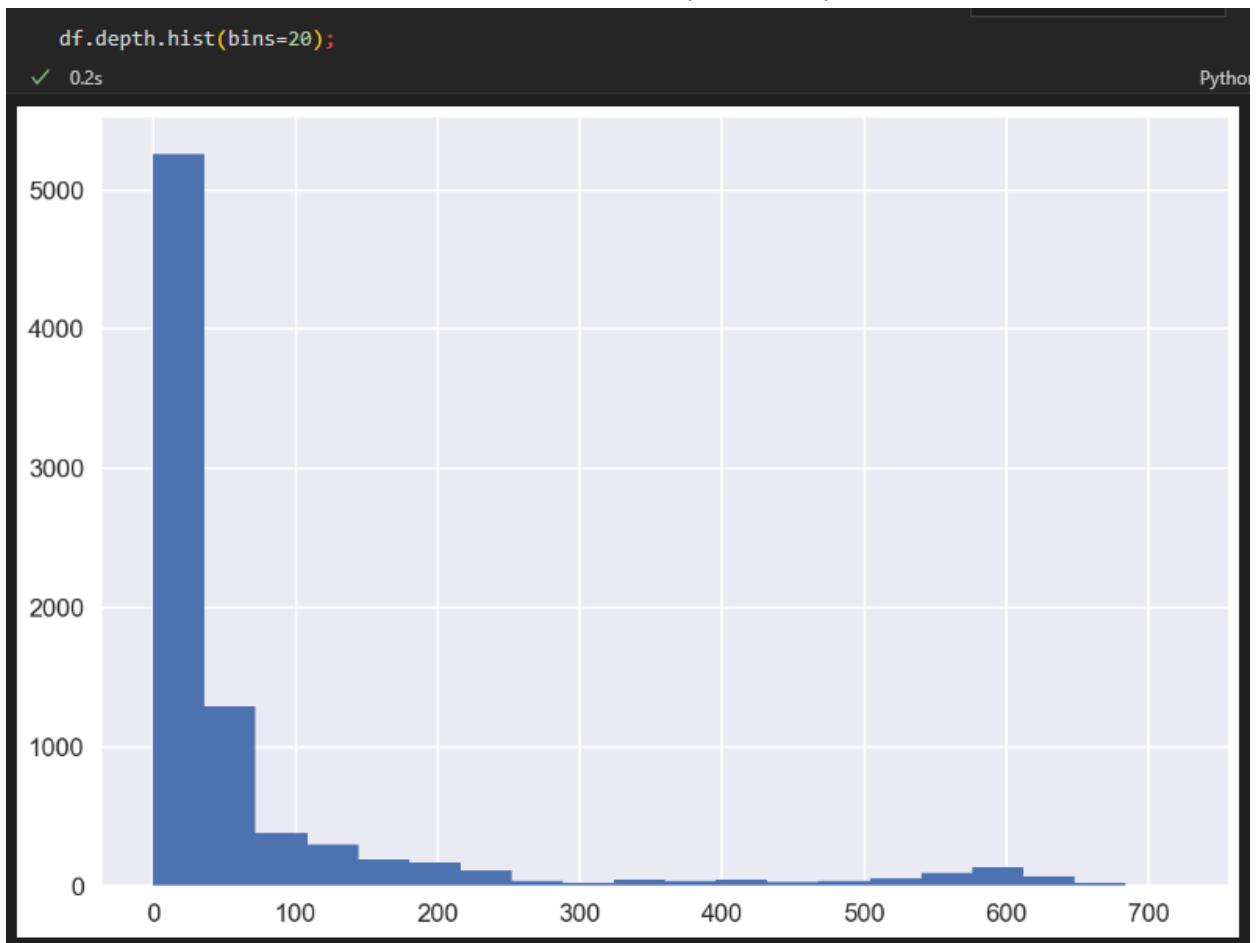
Аргументация, что Vanuatu может быть не самым популярным регионом – 4 балла

(iv). Постройте графики, отвечающие на следующие вопросы:

1. На каких глубинах случаются землетрясения?
2. Сколько землетрясений случилось в разные годы?
3. Как магнитуда землетрясения зависит от глубины?
4. В какое время суток случаются самые сильные землетрясения?

Возможные решения:

1. При первом приближении можно заметить большое количество неглубоких землетрясений, из-за которых сложно понять общую картину

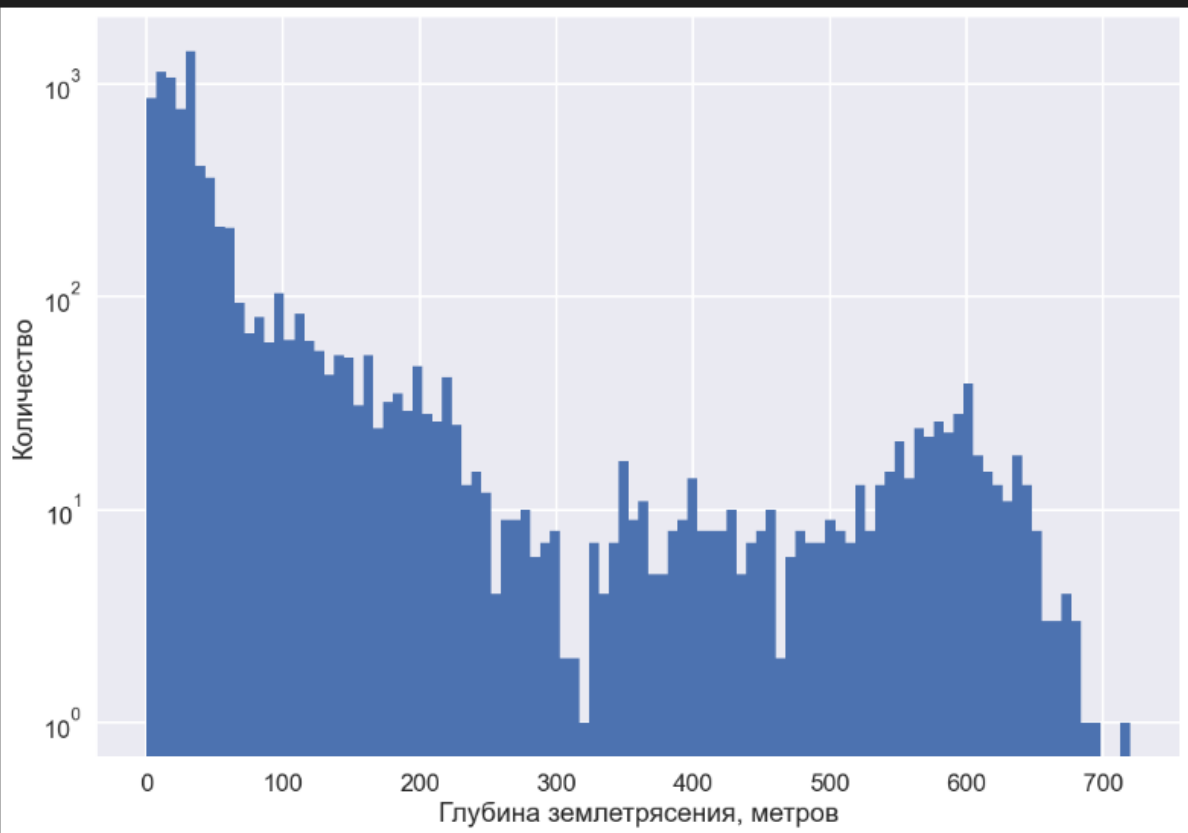


Поэтому можно либо построить два отдельных графика для больших и малых глубин, либо преобразовать ось Y в логарифмическую шкалу. Рассмотрим второй вариант.


```
df.depth.hist(bins=100)
plt.yscale('log')
plt.xlabel('Глубина землетрясения, метров')
plt.ylabel('Количество');
```

✓ 0.4s

Python

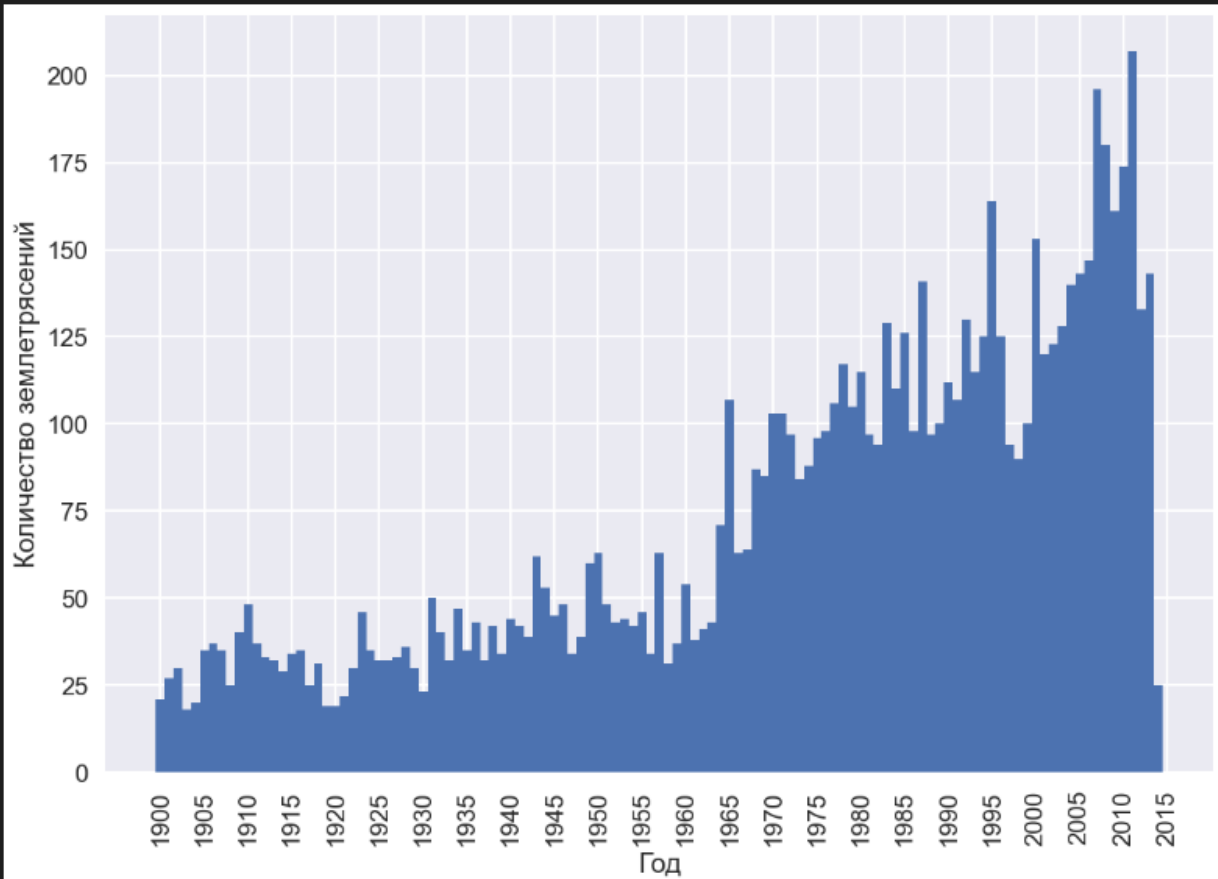


2.

```
df.loc[:, 'year'] = df.date.astype(str).apply(lambda x: x[:4])
years_cnt = df.groupby('year').date.count()
plt.bar(years_cnt.index, years_cnt, width=1)
plt.xticks([str(1900 + i*5) for i in range(24)], rotation=90)
plt.ylabel('Количество землетрясений')
plt.xlabel('Год');
```

✓ 0.4s

Python

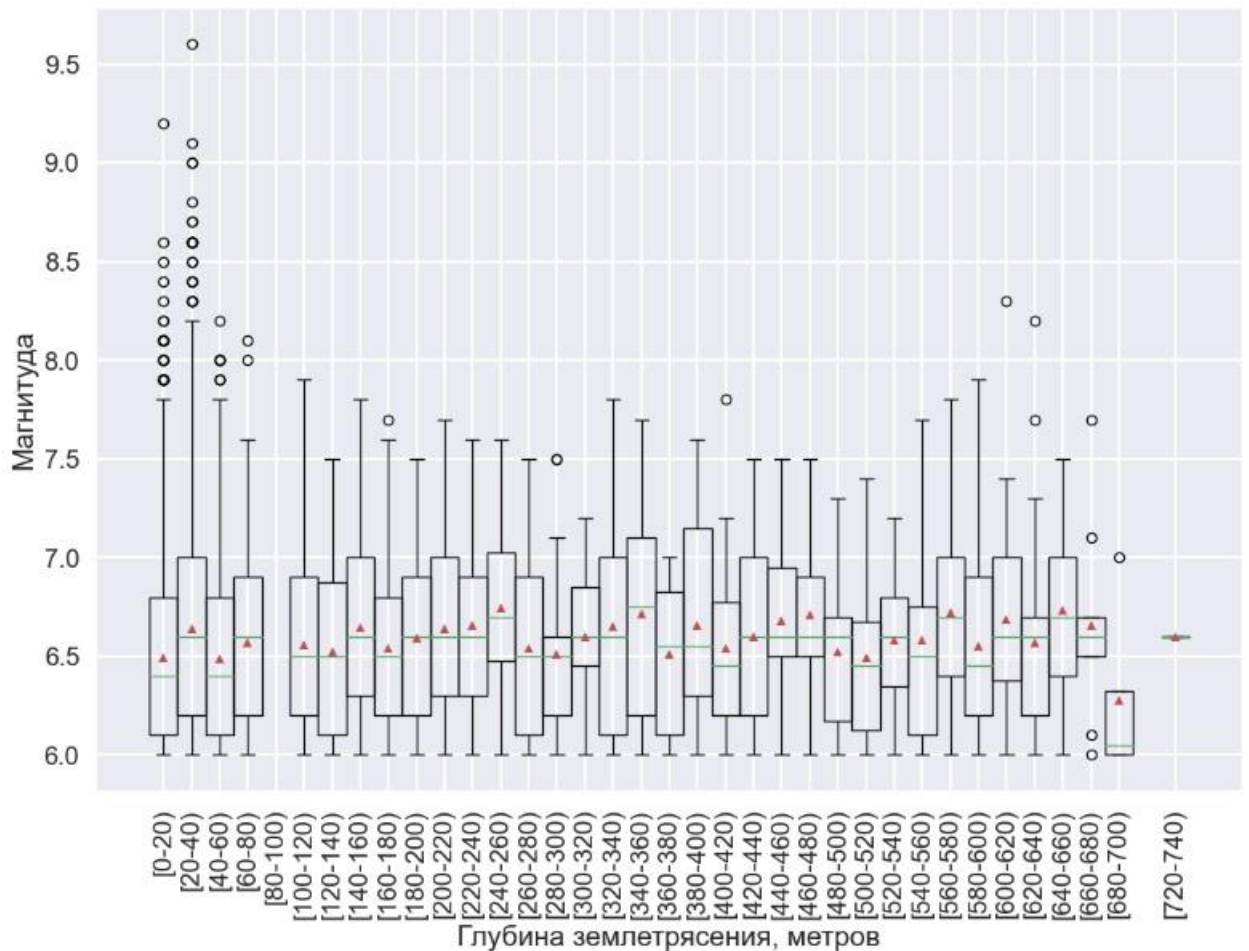


3.

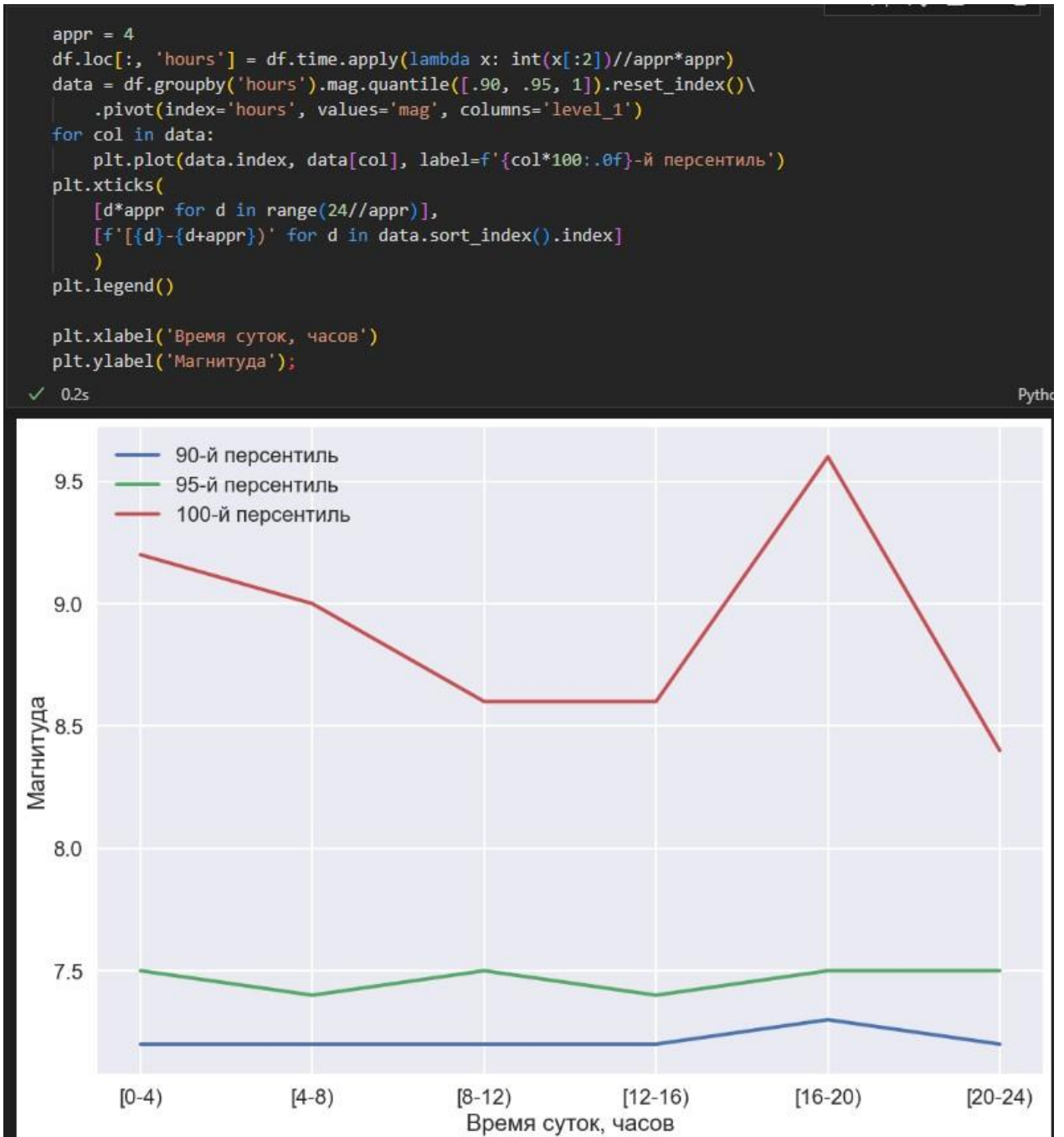
```
appr = 20
df.loc[:, 'approximate_depth'] = df.depth // appr * appr
for d in df.approximate_depth.unique():
    df_depth = df[df.approximate_depth == d]
    plt.boxplot(
        df_depth.mag,
        positions=[d],
        widths=[appr],
        showmeans=True,
        labels=[f'{d:.0f}-{d+appr:.0f}'])
)
plt.xticks(rotation=90)
df_depth = df.groupby('approximate_depth').mag.mean()

plt.xlabel('Глубина землетрясения, метров')
plt.ylabel('Магнитуда');
```

✓ 0.6s Pyth



4.



Критерии:

Мах – **92 балла**

Снимаем **4 балла** за каждый график, не имеющий подходящих подписей, если из-за этого затруднено его прочтение

Ставим **0 баллов** за график, не имеющий подходящих подписей, если из-за этого невозможно его прочтение

Снимаем еще **4 балла** за перегруженный график, если это затрудняет составление выводов.

1. Мах – 20 баллов

Полный балл ставится, если отображен вид распределения и особенности в районе 40 и 600 метров. Если есть только что-то одно – **12 баллов**

2. Мах – 20 баллов

Отображена зависимость

3. Мах – 28 баллов

Полный балл ставится, если по графику можно сделать следующие выводы:

- а. В среднем нет явной зависимости между силой землетрясения и глубиной
- б. Экстремально сильные землетрясения происходят обычно на небольшой глубине

За только один вывод – **12 баллов**

4. Мах – 24 балла

Полный балл ставится, если рассмотрена либо зависимость высоких перцентилей от времени суток, либо распределение сильных землетрясений по времени суток. При этом вывод - нет зависимости.

Если нет определения того, что такое время суток, то за задание можно получить максимум **16 баллов.**

¹ Игнорируйте системы, которые до 2000 года не регистрировали землетрясения