Тестовое задание для Яндекс.Практикум

Описание задачи

Перед вами примеры лог.файлов (3 шт.) от внутреннего сервиса прогнозируюшего временные ряды. 1 день, 1 файл. Сервис используется в eLama ежедневно, обеспечивает порядка 700 прогнозов различных метрик в сутки. События записываются в базу по мере того, как они случаются.

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/ee5d5bfc-797c-45eb-a278-bd76378c77ae/logs_2022-09-14.db

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/0e78fcb3-9f31-441a-9d23-4e987ba7d919/logs_2022-09-13.db

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/49b872cf-8942-49d1-a066-18741adaabf9/logs_2022-09-12.db

Каждый из файлов представляет собой sqlite базу данных со следующими таблицами:

1. **event_name** - таблица с именами событий, которые случаются в сервисе
    a. *id* - уникальный идентификатор

b. *event_name* - название события в рамках сервиса прогнозирования

	id	event_name
1	1	Инициализируем свойства класса
2	2	Переход в функцию getParametrsAndRegressors
3	3	Выход из функции getParametrsAndRegressors с параметрами
4	4	Выход из функции getParametrsAndRegressors БЕЗ параметров
5	5	Записали в свойства класса гиперпараметры и регрессоры

2. **events** - сам лог событий.
   a. *id* - идентификатор конкретного факта наступления события
   b. *forecastMarker* - идентификатор задачи прогнозирования (так мы называем общую задачу прогноза конкретного бизнес-показателя. Есть отдельная таблица, где хранятся свойства задач прогнозирования (источники данных, гиперпараметры, регрессоры и т.д.), (необязательное поле)
   c. *request_time* - время когда зафиксировано событие
   d. *event_id* - идентификатор события
   e. *sender* - Имя системы поставившей задачу прогнозирования (необязательное поле)
   f. *context* - значения переменных или каких-либо параметров, связанных с событием (необязательно поле)
   g. *session_id* - идентификатор исполнения конкретной задачи прогнозирования (используется если задача пришла без forecastMarker и sender)

	id	forecastMarker	request_time		sender	cc	session_id
109	109	SaaS/RUS/activation	2022-09-09 04:44:40.348920	18	internal_call fc	None	9495ab6f04c3a99fff8772735c4e
110	110	SaaS/RUS/activation	2022-09-09 04:44:40.400790	19	internal_call fc	None	9495ab6f04c3a99fff8772735c4e
111	111	SaaS/RUS/activation	2022-09-09 04:44:41.396641	20	internal_call fc	None	9495ab6f04c3a99fff8772735c4e
112	112	SelfService/RUS/reg	2022-09-09 04:44:43.186943	9	forecast_for_P	None	4da5fda6387bdd1061ab5ec4a0
113	113	SelfService/RUS/reg	2022-09-09 04:44:43.260936	10	forecast_for_P	None	4da5fda6387bdd1061ab5ec4a0

Важно отметить:

1. Первое событие для каждой сессии имеет идентификатор 1. Идентификаторы последующих событий не соответствуют последовательности наступления.
2. Шаг событий - это интервал времени между событиями наступающими друг за другом.
3. Задача прогнозирования считается завершенной успешно, если наступило событие с идентификатором 31.
4. Некоторые задачи прогнозирования могут использовать прогнозы по типу рекурсии. Т.е. при построении прогноза запрашивать прогноз другого параметра, как регрессора. Например при построении прогноза активаций пользователей запрашивается подпрогноз с количеством регистраций. Т.к. регистрации напрямую влияют на количество активаций.

Вопросы

Имея эти данные ответьте на вопросы (данные из баз надо объединить):

1. Между какими событиями наибольший шаг? Укажите пару идентификаторов событий с наибольшим шагом, относительно всей базы (всех файлов).
2. Интервал времени до наступления какого события показывает наибольший разброс. Укажите пару идентификатор и sender с наибольшим разбросом. Предложите свои идей, с чем это может быть связано.
3. Какая задача прогнозирования выполняется дольше всего (название задачи)?
4. Какое количество задач прогнозирования могут выполняться одновременно? Укажите максимальное число параллельных задач.
5. Перечислите *forecastMarker* которые не завершились успешно.