**Описание полей таблицы и их типов.**

Все данные находятся в одной плоской таблице, поэтому далее привожу описание полей этой таблицы и их типов.

Таблица Входных данных.

|  |  |  |
| --- | --- | --- |
| **Поле** | **Описание** | **Тип Данных** |
| VendorId | Идентификатор компании (1, 2) | SMALLINT |
| tpep\_pickup\_datetime | Время и дата, когда пассажир сел в такси | TIMESTAMP WITHOUT TIME ZONE |
| tpep\_dropoff\_datetime | Время и дата, когда пассажир вышел из такси | TIMESTAMP WITHOUT TIME ZONE |
| passanger\_count | Количество пассажиров | SMALLINT |
| trip\_distance | Пройденное расстояние | FLOAT(24) |
| RatecodeID | Код скорости | SMALLINT |
| store\_and\_fwd\_flag | Флаг, отвечающий за сохранение записи поездки перед ее отправкой поставщику | CHAR(1) |
| PUlocationID | Широта, где была начата поездка | SMALLINT |
| DOlocationID | Долгота, где была начата поездка | SMALLINT |
| payment\_type | Тип оплаты | SMALLINT |
| fare\_amount | Стоимость поездки | FLOAT(24) |
| extra | Дополнительные платежи за ночь, пробки и т.д. Мах $7 | FLOAT(24) |
| mta\_tax | Комиссия автопарка | FLOAT(24) |
| tip\_amount | Чаевые | FLOAT(24) |
| tolls\_amount | Оплата за платные дороги | FLOAT(24) |
| improvement\_surchange | Доплата за страховку | FLOAT(24) |
| total\_amount | Полная стоимость поездки | FLOAT(24) |
| congestion\_surchange | Дополнительный сбор | FLOAT(24) |
| trip\_day | День поездки | DATE |

Таблица результатов анализа.

|  |  |  |
| --- | --- | --- |
| **Поле** | **Описание** | **Тип Данных** |
| trip\_date | День поездки | DATE |
| percentage\_zero | Процент поездок без пассажиров | FLOAT(24) |
| percentage\_1p | Процент поездок с одним пассажиром | FLOAT(24) |
| percentage\_2p | Процент поездок с двумя пассажирами | FLOAT(24) |
| percentage\_3p | Процент поездок с тремя пассажирами | FLOAT(24) |
| percentage\_4p\_plus | Процент поездок с 4 и более пассажирами | FLOAT(24) |
| percentage\_zero\_min | Минимальная цена поездки без пассажира | FLOAT(24) |
| percentage\_zero\_max | Максимальная цена поездки без пассажира | FLOAT(24) |
| percentage\_1p\_min | Минимальная цена поездки с одним пассажиром | FLOAT(24) |
| percentage\_1p\_max | Максимальная цена поездки с одним пассажиром | FLOAT(24) |
| percentage\_2p\_min | Минимальная цена поездки с двумя пассажирами | FLOAT(24) |
| percentage\_2p\_max | Максимальная цена поездки с двумя пассажирами | FLOAT(24) |
| percentage\_3p\_min | Минимальная цена поездки с тремя пассажирами | FLOAT(24) |
| percentage\_3p\_max | Максимальная цена поездки с тремя пассажирами | FLOAT(24) |
| percentage\_4p\_plus\_min | Минимальная цена поездки с 4 и более пассажирами | FLOAT(24) |
| percentage\_4p\_plus\_max | Максимальная цена поездки с 4 и более пассажирами | FLOAT(24) |

**Мини-отчет по качеству входных данных и описание процесса data quality.**

Сначала импортируем все целочисленные данные как BIGINT, потом проверяем, что значения не превышают 32767 с помощью запроса “SELECT "VendorID", passenger\_count, "RatecodeID", "PULocationID", "DOLocationID", payment\_type FROM yellow\_taxi\_data WHERE "VendorID" > 32767 OR passenger\_count > 32767 OR "RatecodeID" > 32767 OR "PULocationID" > 32767 OR "DOLocationID" > 32767 OR payment\_type > 32767”, после чего можно поменять тип всех полей на SMALLINT и импортировать таблицу заново. В результате размер таблицы в памяти уменьшился с 860 Мб до 666 Мб.

Изначально в таблице 6405008 записей.

Производим очистку данных:

DELETE FROM yellow\_taxi\_data WHERE NOT (yellow\_taxi\_data IS NOT null);

Удалена 65441 запись.

DELETE FROM yellow\_taxi\_data WHERE EXTRACT(YEAR FROM (tpep\_dropoff\_datetime)) != 2020; -- Delete the records which ended before or after 2020

Удалена 101 запись.

DELETE FROM yellow\_taxi\_data WHERE EXTRACT(MONTH FROM (tpep\_pickup\_datetime)) > 1; -- Delete the records which started after Jan 2020 and in Dec 2019

Удалено 111 записей.

DELETE FROM yellow\_taxi\_data WHERE EXTRACT(EPOCH from AGE(tpep\_dropoff\_datetime, tpep\_pickup\_datetime)) > 3600\*6; -- Length > 6h

Удалено 13352 записи.

DELETE FROM yellow\_taxi\_data WHERE EXTRACT(EPOCH from AGE(tpep\_dropoff\_datetime, tpep\_pickup\_datetime)) > 3600 and tpep\_dropoff\_datetime :: TIME = '00:00:00'; -- Wrong dropoff time and > 1h

Удалено 324 записи.

DELETE FROM yellow\_taxi\_data WHERE EXTRACT(EPOCH from AGE(tpep\_dropoff\_datetime, tpep\_pickup\_datetime)) < 10; -- Length < 10s

Удалено 28843 записи.

DELETE FROM yellow\_taxi\_data WHERE passenger\_count < 0;

Удалено 0 записей.

DELETE FROM yellow\_taxi\_data WHERE trip\_distance <= 0 OR fare\_amount < 2.5 OR extra > 7;

Удалено 63329 записей.

После этого можно поискать дубликаты с помощью запроса:

SELECT (yellow\_taxi\_data.\*)::text, count(\*) FROM yellow\_taxi\_data GROUP BY yellow\_taxi\_data.\* HAVING count(\*) > 1

Дубликатов не найдено.

Итого удалено 171501 неконсистентных записей, что составляет 2.68% от исходных данных.