

Запросы для составления прогнозов

Данные о доставке и сборке используются в качестве основного прогнозируемого ряда, в то время как остальные (погода, праздники) в качестве регрессионных данных.

Зональный прогноз доставки

Для зонального прогноза доставки извлекаются и агрегируются данные рассчитываемые как сумма количества заказов (cnt), умноженного на коэффициент из геоточки (coeff), сгруппированных по идентификатору торговой точки (tt_id) и часовому интервалу (bucket).

Ниже представлен запрос который делает все это и объединяет данные о заказах с географическими точками для анализа по сегментам (торговым точкам или зонам) в заданном диапазоне дат:

```
SELECT
  h.bucket AS timestamp,
  vp.tt_id AS segment,
  SUM(cnt * coeff) AS target
FROM vv_orders_ts_hash_hourly h
JOIN test_vv_points vp ON vp.geohash = h.geohash
WHERE bucket BETWEEN '{from_date}' AND '{to_date}'
GROUP BY vp.tt_id, h.bucket;
```

Запрос берет значения из:

- vv_orders_ts_hash_hourly — представление которое содержит агрегированные данные о количестве заказов по геохешу и часам.
- test_vv_points — таблица, содержащая географические зоны, связанные с торговыми точками.

Запрос формирует следующие поля:

Поле в SELECT	Исходное поле	Описание
timestamp	h.bucket	Временной интервал, начало часа для агрегации данных о заказах.
segment	vp.tt_id	Идентификатор торговой точки, к которой привязан geohash.
target	(Вычисляемое)	Сумма (cnt * coeff), где cnt — количество заказов, coeff — коэффициент из геоточки.

Зональный прогноз сборки

Для зонального прогноза сборки извлекаются и агрегируются данные рассчитываемые как сумма количества строк в заказах (cnt), умноженного на коэффициент (coeff), сгруппированных по идентификатору торговой точки (tt_id) и часовому интервалу (bucket).

Ниже представлен запрос который делает все это и объединяет данные о строках заказов с географическими точками для зонального анализа в заданном диапазоне дат.

```
SELECT
  h.bucket AS timestamp,
  vp.tt_id AS segment,
  SUM(cnt * coeff) AS target
FROM vv_lines_ts_hash_hourly h
JOIN test_vv_points vp ON vp.geohash = h.geohash
WHERE bucket BETWEEN '{from_date}' AND '{to_date}'
GROUP BY vp.tt_id, h.bucket;
```

Запрос берет значения из:

- vv_lines_ts_hash_hourly — представление которое содержит агрегированные данные о количестве заказов по геохешу и часам.
- test_vv_points — таблица, содержащая географические зоны, связанные с торговыми точками.

Запрос формирует следующие поля:

Поле в SELECT	Исходное поле	Описание
---------------	---------------	----------

timestamp	h.bucket	Временной интервал, начало часа для агрегации данных о заказах.
segment	vp.tt_id	Идентификатор торговой точки, к которой привязан geohash.
target	(Вычисляемое)	Сумма (cnt * coeff), где cnt — количество строк в заказах, coeff — коэффициент из геоточки.

Данные о погоде

В прогнозах используются данные о погоде, такие как температура, влажность и скорость ветра из активных погодных станций в заданном диапазоне дат.

Ниже представлен запрос данных о погоде с дополнением последними значениями на конечную дату.

```
WITH t AS (  
  SELECT  
    w.weather_station_id AS station_id,  
    w.date,  
    w.temperature,  
    w.humidity,  
    w.wind_speed,  
    ROW_NUMBER() OVER (PARTITION BY w.weather_station_id ORDER BY w.date DESC) AS rn  
  FROM weather w  
  JOIN weather_stations st ON st.id = w.weather_station_id  
  WHERE st.active AND date BETWEEN '{from_date}' AND '{to_date}'  
)  
SELECT  
  station_id,  
  date,  
  temperature,  
  humidity,  
  wind_speed  
FROM t  
UNION  
SELECT  
  station_id,
```

```
{to_date}',
temperature,
humidity,
wind_speed
FROM t
WHERE t.rn = 1
ORDER BY station_id, date;
```

Запрос берет значения из:

- weather — содержит исторические данные о погоде по станциям и датам.
- weather_stations — содержит информацию о погодных станциях.

Запрос формирует следующие поля:

Поле в SELECT	Исходное поле	Описание
station_id	w.weather_station_id	Идентификатор погодной станции.
date	w.date	Дата измерения погоды.
temperature	w.temperature	Температура
humidity	w.humidity	Влажность воздуха
wind_speed	w.wind_speed	Скорость ветра

На основе метеорологических данных рассчитывается эквивалентная температура, которая используется в качестве входных данных для регрессионного анализа.

$$37 - (37 - \text{temperature}) / (0.68 - 0.0014 * \text{humidity}) + 1 / (1.76 + 1.4 * \text{pow}(\text{wind_speed}, 0.75))) - 0.29 * \text{temperature} * (1 - \text{humidity} / 100)$$

В этой формуле:

- Вычисляется разница между 37°C и фактической температурой: (37 - temperature). Это базовый "дефицит тепла".
- Вычисляется фактор сопротивления: (0.68 - 0.0014 * humidity + 1 / (1.76 + 1.4 * pow(wind_speed, 0.75))). Он увеличивается при высокой влажности (меньше охлаждения) и уменьшается при сильном ветре (больше охлаждения).
- Делится разница на фактор сопротивления и вычитается из 37: это даёт основную ощущаемую температуру с учетом конвекции.

- Вычитается корректировка на испарение: $0.29 * \text{temperature} * (1 - \text{humidity} / 100)$, которая дополнительно охлаждает в сухих условиях.

Календарные данные

В прогнозах используются данные производственного календаря из таблицы и календарь Православных Христианских праздников.

Ниже представлен запрос календарных данных:

```
SELECT
  date,
  type AS holiday
FROM calendar
WHERE date BETWEEN '{from_date}' AND '{to_date}'
ORDER BY date;
```

Запрос берет значения из:

- calendar — табель-календарь, заполняемый через систему репликации

Запрос формирует следующие поля:

Поле в SELECT	Исходное поле	Описание
date	date	Дата календарного события
holiday	type	Тип события

Данные о православных праздниках хранятся в файле: [calendar.csv](#)

Revision #2

Created 19 August 2025 07:38:51 by Семен Долгов

Updated 19 August 2025 08:08:29 by Семен Долгов