

Предиктивная аналитика

- Описание прогнозной модели
- Типы прогнозов
- Логика прогнозов

Описание прогнозной модели

Прогнозные модели и скрипты размещены на сервере `izb-ally-nodered02` (IP: 10.1.241.244).

Скрипты написаны на **Python** с использованием библиотеки **ENTA**

```
python3.10 main.py --help
Usage: main.py [OPTIONS] COMMAND [ARGS]...

Options:
  --help Show this message and exit.

Commands:
  copy    copy time_series from one DB to another
  fit     Fits model with collected lines from DB
  forecast Makes forecast for the next 7 days and saves it to DB
  test    Evaluates forecast metrics
  view    Prints one segment used for forecast
```

Наилучшие результаты при прогнозировании показала модель **CatBoost**.

Перед использованием модель необходимо обучить. После обучения она может применяться для построения прогноза.

Данные для обучения поступают из различных источников, а результаты прогнозов сохраняются в гипертаблицу `time_series`.

Для обучения используются погодные данные, которые ежедневно загружаются с сайта rp5.ru в таблицы `weather` и `weather_stations`. Загрузка осуществляется с помощью скрипта `rp5_weather`, который настраивается через файл `static/cities.txt`. Этот файл содержит список метеостанций по всей стране.

В таблице `weather` также хранятся исторические данные о погоде, начиная с 2005 года.

Кроме исторических данных, для прогнозирования требуется актуальный прогноз погоды. Он ежедневно загружается с сайта api.met.no через **Node-RED** в таблицу `weather`.

Также в прогнозе используется данные табель календаря из таблицы calendar и календарь Православных Христианских праздников.

Также при построении прогноза учитываются данные производственного календаря из таблицы calendar и календаря православных праздников.

Попытка включить мусульманские праздники в расчёт прогноза не улучшила точность.

Прогноз может строиться на заданное количество дней вперёд и с различной степенью дискретизации.

Типы прогнозов

Данные для прогнозов, сами прогнозы и результаты их анализа хранятся в таблице

`time_series_types` :

ID	Описание	Подразделение	Тип	Источник
1	Данные по доставке	Последняя миля	Исходные данные	Репликатор
2	Данные по сборке	Последняя миля	Исходные данные	Репликатор
4	Прогноз доставка	Последняя миля	Прогноз	Python
5	Прогноз сборка	Последняя миля	Прогноз	Python
6	Прогноз по геозонам доставка	Последняя миля	Прогноз	Python
7	Прогноз по геозонам сборка	Последняя миля	Прогноз	Python
51	Оценка трудоемкости розница	Розница	Исходные данные	Node-RED
52	Прогноз суммарной трудоемкости Розница	Розница	Прогноз	Python
53	Прогноз трудоемкости кассиров Розница	Розница	Прогноз	Python
55	Факт суммарной выработки Розница	Розница	Аналитика	pgAgent
56	Факт выработки кассиров Розница	Розница	Аналитика	pgAgent
154	Курьеры факт	Последняя миля	Аналитика	Node-RED
155	Курьеры план	Последняя миля	Аналитика	Node-RED
156	Курьеры прогноз	Последняя миля	Аналитика	Node-RED
157	Заказы план	Последняя миля	Аналитика	Node-RED

crontab -l

```
# m h dom mon dow  command
```

```
30 3 * * * cd /home/dtimakin-ally/rp5_weather && python3.9 main.py
```

```
0 4 * * * cd /home/dtimakin-ally/forecast && python3.9 main.py forecast --model model.zip --type vv --ts_type 4  
--days 14
```

```
15 4 * * * cd /home/dtimakin-ally/forecast && python3.9 main.py forecast --model lines.zip --type lines --ts_type  
5 --days 14
```

```
30 4 * * * cd /home/dtimakin-ally/forecast && python3.9 main.py fit --model model_new_2.zip --type vv_new
```

```
30 5 * * * cd /home/dtimakin-ally/forecast && python3.9 main.py fit --model model_lines_new.zip --type  
vv_lines_new
```

```
30 6 * * * cd /home/dtimakin-ally/forecast && python3.9 main.py forecast --model model_new_2.zip --type  
vv_new --ts_type 6 --days 14
```

```
45 6 * * * cd /home/dtimakin-ally/forecast && python3.9 main.py forecast --model model_lines_new.zip --type  
vv_lines_new --ts_type 7 --days 14
```

```
10 9 * * * cd /home/dtimakin-ally/forecast_day && python3.9 main.py forecast --model day_hours.zip --type  
day_hours --ts_type 52 --days 21 --freq D
```

```
40 9 * * * cd /home/dtimakin-ally/forecast_day && python3.9 main.py forecast --model day_cash_hours.zip --type  
day_cash_hours --ts_type 53 --days 21 --freq D
```

Логика прогнозов

Прогноз для последней мили:

Прогнозы по сборке и доставке строятся с шагом 1 час и охватывают период в 14 дней вперёд.

Обычный прогноз.

Основан на данных о сборке и доставке, где каждый заказ привязан к ТТ.

Данные загружаются в БД Ally из БД MS SQL через **Node-RED**.

Зональный прогноз.

Основан на данных о сборке и доставке, где каждый заказ привязан к ТТ, и на статистике распределения адресов по зонам той или иной ТТ.

На основе геозон ТТ формируются сводные буферы, по которым рассчитываются вероятности выполнения заказов конкретной ТТ, исходя из статистики за последние 7 дней.

Вся история заказов автоматически сопоставляется с текущими геозонами, что позволяет обучать модель так, как будто эти зоны всегда были такими. Это также даёт возможность строить прогнозы для новых ТТ с недостаточной историей данных.

Прогноз для розницы

Прогноз по трудоёмкости строится с шагом 1 день и охватывает 21 день вперёд.

Основан на оценке трудоёмкости, которая загружается во временной ряд 51 через **Node-RED**.

Расчёт требуемого количества сборщиков и курьеров на основе прогноза

Прогноз трудоёмкости сборщиков автоматически пересчитывается в требуемое количество сборщиков. Расчёт выполняется по следующему алгоритму:

1. Оценка скорости сборки сотрудников

Для каждого сотрудника рассчитывается средняя скорость сборки на основе фактических данных за последний месяц. Скорость выражается в количестве строк, которые сотрудник собирает в час.

2. Расчёт плановой производительности

Для каждого часа прогнозируемого периода рассчитывается ожидаемое количество собранных строк на основе списка запланированных сборщиков и их индивидуальной скорости работы.

3. Сравнение с прогнозом

Вычисляется разность между прогнозируемым количеством строк на сборку и плановой производительностью сборщиков.

4. Корректировка численности

Разность преобразуется в дополнительное или избыточное количество сотрудников. Для этого используется средняя скорость сборки по всем сборщикам на данной торговой точке.

5. Вывод итогового количества

Полученное число используется для формирования рекомендаций: сколько сотрудников необходимо добавить или убрать из смены в конкретный час.

Для подсчета скорости сборки заказов используется подход средней скорости сборки которая рассчитывается как количество собранных строк в течение одного часа (строки / час).

Расчёт необходимого количества курьеров выполняется по той же логике, что и для сборщиков.