

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное  
образовательное учреждение высшего образования  
«Самарский национальный исследовательский университет  
имени академика С.П. Королева»  
(Самарский университет)

Институт информатики и кибернетики  
Кафедра технической кибернетики

**Отчет по лабораторной работе №3**

Дисциплина: «Инженерия данных»

Выполнил: Осепян С.М.

Группа: 6233-010402D

### **Цели работы**

1. Освоить полный цикл оркестрации ML-проекта в ClearML: датасет → эксперименты → подбор гиперпараметров → реестр моделей → деплой.
2. На практике реализовать инференс-сервис (FastAPI) для прогноза на 7 дней вперёд.
3. Настроить n8n-пайплайн с еженедельным расписанием, который запрашивает прогноз у сервиса и отправляет его в Telegram.

## Постановка задачи (пайплайн)

### 1. ClearML

- Поднять ClearML Server (docker-compose) и настроить Agent (опционально).
- Создать ClearML Dataset с историческими данными.
- Реализовать скрипты для полного цикла обучения, поиска гиперпараметров (HPO), оценки модели с использованием логирования в ClearML: метрики (MAE/RMSE/MAPE/AUC), артефакты (конфиг, важности признаков, графики).
- Выполнить HPO (не менее 10 конфигураций) и зарегистрировать лучшую модель в Model Registry (указать версию/теги).

### 2. Признаки и модели

- Разрешается использовать любые признаки: календарные признаки (dow, doy, sin/cos сезонности), лаги и скользящие агрегаты.
- Разрешается использовать любые модели: LightGBM/XGBoost/CatBoost, Prophet, LSTM/Temporal-Conv, и т.д.
- Горизонт Предсказаний 7 дней.

### 3. Инференс-сервис

- FastAPI с эндпоинтом POST /predict (вход: город, список дат [D+1..D+7]; опц. вспомогательные признаки; выход: JSON с 7 прогнозами и доверительными интервалами/вероятностями).
- Упаковать в Docker.

### 4. n8n (еженедельный пайплайн)

- Cron (раз в неделю, например, ПН 07:00) → HTTP Request к /predict для выбранного города/городов → форматирование → Telegram.

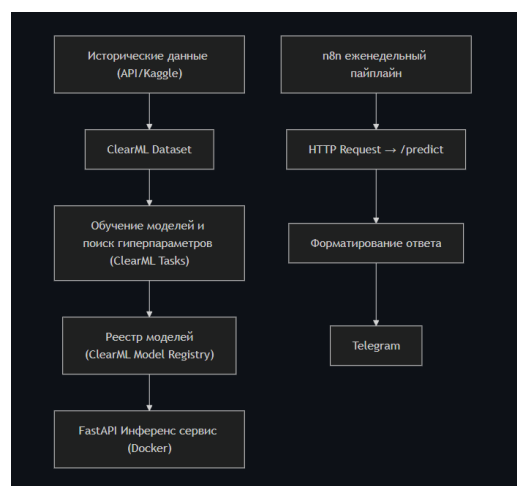


Рисунок 1 – Схема работы пайплайна

## ClearML

Был поднят ClearML с помощью docker-compose, за основу был взят docker-compose из оригинального github [clearml-server/docker/docker-compose.yml](https://github.com/clearml-server/docker/docker-compose.yml) at master · clearml/clearml-server

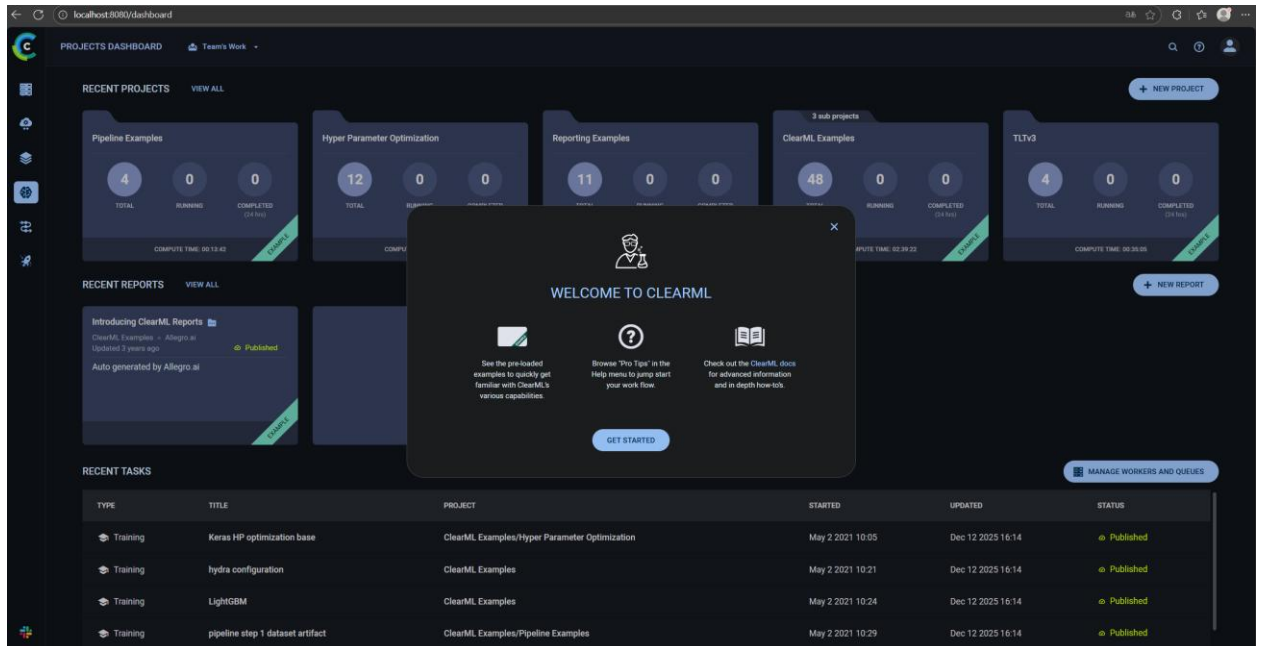


Рисунок 2 – Стартовая страница ClearML

Для дальнейшей работы ClearML необходимо запустить стартовый скрипт с помощью команды `clearml-init`, и в него вставить API Credential, который был дан на стартовой странице

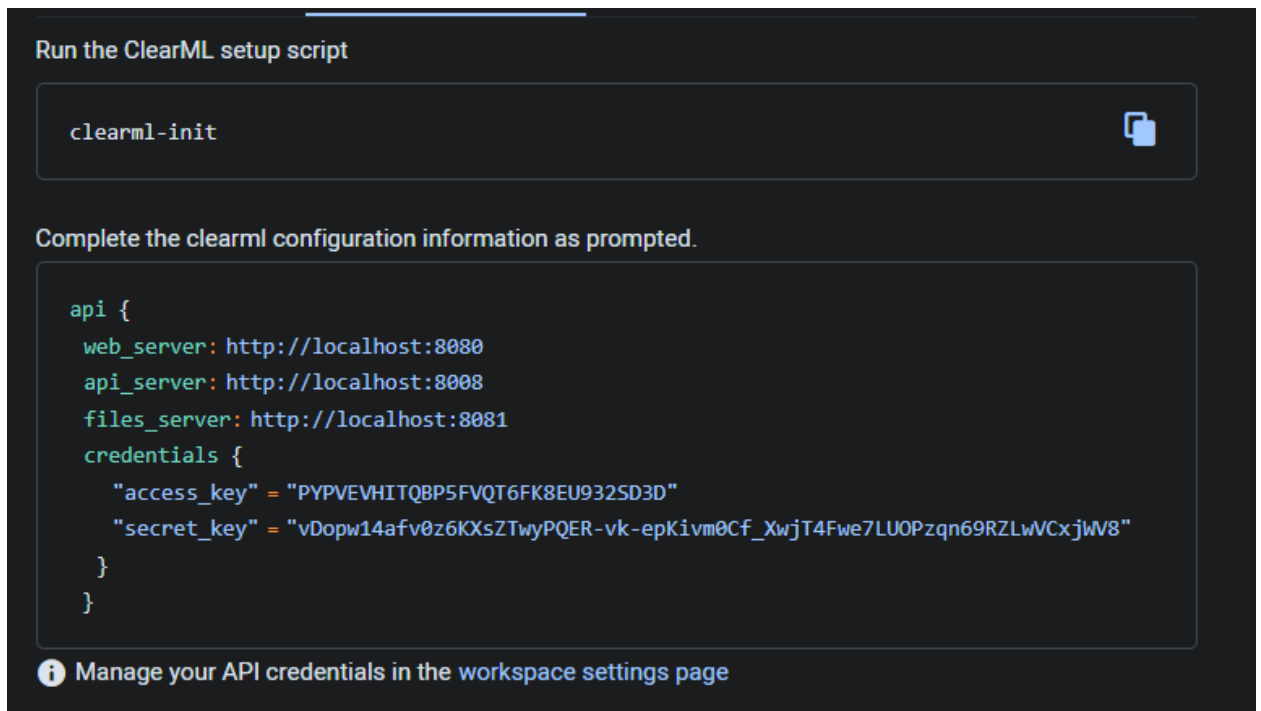


Рисунок 3 – API Credential

```

PS E:\paring\Lab-3-2025\clearml> clearml-init
ClearML SDK setup process

Please create new clearml credentials through the settings page in your `clearml-server` web app (e.g. http://localhost:8080//settings/workspace-configuration)
Or create a free account at https://app.clear.ml/settings/workspace-configuration

In settings page, press "Create new credentials", then press "Copy to clipboard".

Paste copied configuration here:
api {
  web_server: http://localhost:8080
  api_server: http://localhost:8008
  files_server: http://localhost:8081
  credentials {
    "access_key" = "PYPVEVHITQ8P5FVQT6FK8EU932SD3D"
    "secret_key" = "vDopw14afv0z6KXsZTwyPQER-vk-epKivm0Cf_XorJT4Fwe7LUOPzqn69RZLwVCxjw8"
  }
}
Detected credentials key="PYPVEVHITQ8P5FVQT6FK8EU932SD3D" secret="vDop****"
Port 8080 is the web port. Replacing 8080 with 8008 for API server

ClearML Hosts configuration:
Web App: http://localhost:8080
API: http://localhost:8008
File Store: http://localhost:8081

Verifying credentials ...
Credentials verified!

New configuration stored in C:\Users\vazge\clearml.conf
ClearML setup completed successfully.

```

Рисунок 4 – Исполнение команды clearml-init

После исполнения команды clearml-init мы можем создать в ClearML датасет. Данные для датасета возьмем из <https://archive-api.open-meteo.com/v1/archive>

```

PS E:\paring\Lab-3-2025> python src/data/fetch_historical.py
=====
Создание датасета с температурными данными
=====
ClearML Task: created new task id=748484127ff34fb58c44920a28308f2f
ClearML results page: http://localhost:8080/projects/1ee7d947ffda44d6b006b121b5771755/experiments/748484127ff34fb58c44920a28308f2f/output/log
Создание ClearML Dataset...
ClearML results page: http://localhost:8080/projects/92c259ed26b3436da94e0f0071bd941e/experiments/492a229967584a19af9f948d7783a684/output/log
ClearML dataset page: http://localhost:8080/datasets/simple/92c259ed26b3436da94e0f0071bd941e/experiments/492a229967584a19af9f948d7783a684
Загрузка данных для London с 2022-12-13 по 2025-12-12
Получено 1096 записей
После очистки: 1082 строк, 59 столбцов
Статистика температуры:
  Средняя температура: 12.1°C
  Min: -2.0°C, Max: 28.6°C
  Средние осадки: 2.11 мм
  Дождливых дней: 657 (60.7%)
Сохранение данных в файл: E:\paring\Lab-3-2025\london_weather_temp_20251212_190828.csv
Добавление файла в датасет...
Загрузка датасета на сервер...
Compressing E:/paring/Lab-3-2025/london_weather_temp_20251212_190828.csv
Uploading dataset changes (1 files compressed to 109.87 KiB) to http://localhost:8081
File compression and upload completed: total size 109.87 KiB, 1 chunk(s) stored (average size 109.87 KiB)
Финализация датасета...

✅ Dataset успешно создан!
ID: 492a229967584a19af9f948d7783a684
Название: London Weather Temperature v1
Проект: Lab3 Weather Forecasting
Версия: 1.0.0
URL: http://localhost:8080/datasets/492a229967584a19af9f948d7783a684
Dataset создан: 492a229967584a19af9f948d7783a684
Временный файл удален: E:\paring\Lab-3-2025\london_weather_temp_20251212_190828.csv

✅ Готово! Dataset ID: 492a229967584a19af9f948d7783a684
Для использования в обучении:
dataset = Dataset.get(dataset project='Lab3 Weather Forecasting', dataset name='London Weather Temperature v1')

```

Рисунок 5 – Запуск создания датасета ClearML

Вернемся в WEB интерфейс ClearML, и зайдем на страницу DATASETS, здесь появился наш датасет

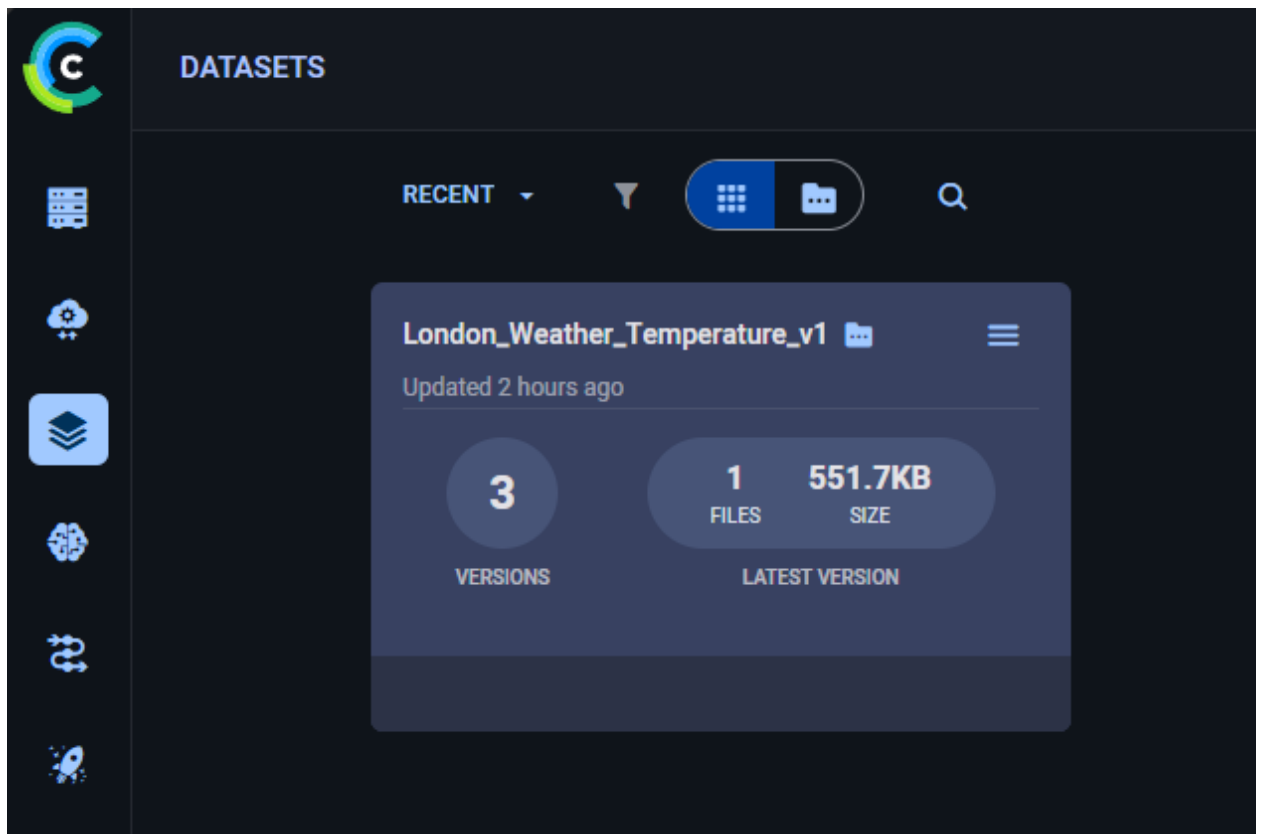


Рисунок 6 – Созданный датасет во вкладке датасетов

Войдя в датасет, увидим что он создан успешно

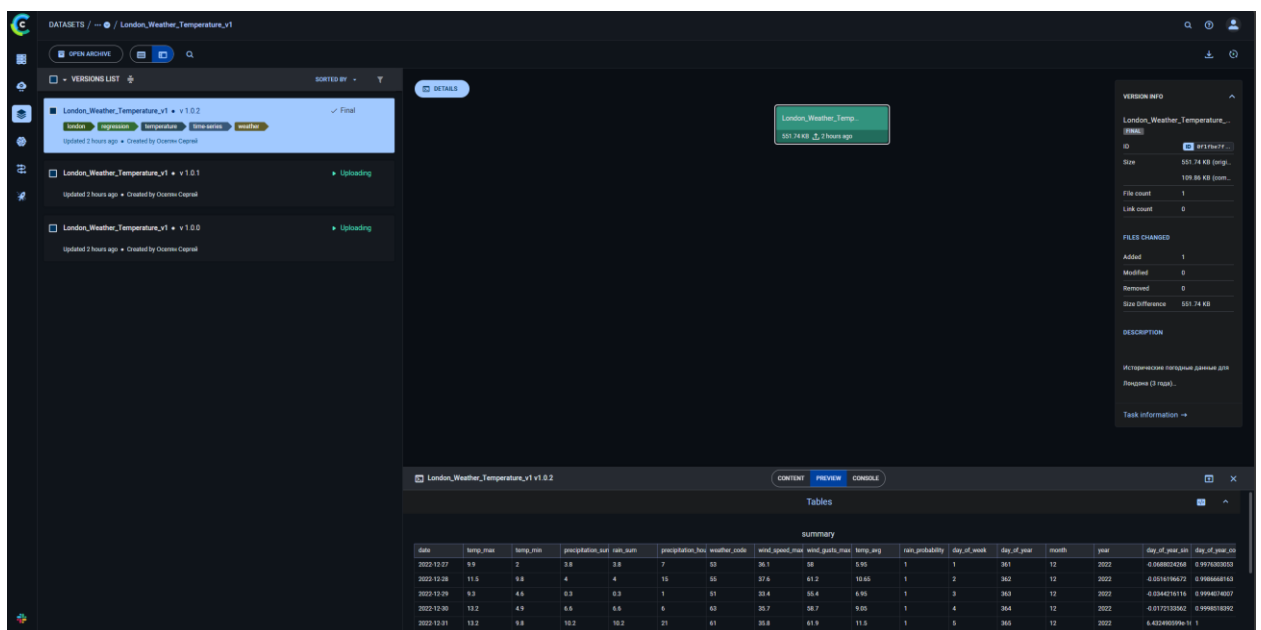


Рисунок 7 – Датасет London\_Weather\_Temperature\_v1

Теперь можем обучить модель, запустим его

```
PS E:\paring\Lab-3-2025> python src/training/train_model.py

=====
ОБУЧЕНИЕ МОДЕЛИ ДЛЯ ПРОГНОЗА СРЕДНЕЙ ТЕМПЕРАТУРЫ
=====

ClearML Task: created new task id=ef479e73aa7743cae46ae8deb28a9e2
ClearML results page: http://localhost:8080/projects/1ee7d947ffda44d6b006b121b5771755/experiments/ef479e73aa7743cae46ae8deb28a9e2/output/log

[Иконка] Загрузка датасета из ClearML...
Датасет загружен в: C:/Users/vazge/.clearml/cache/storage_manager/datasets/ds_492a229967584a19af9f948d7783a684
Загрузка данных из: C:/Users/vazge/.clearml/cache/storage_manager/datasets/ds_492a229967584a19af9f948d7783a684/london_weather_temp_20251212_190828.csv
[Иконка] Данные загружены: 1082 строк, 59 столбцов
[Иконка] Период: 2022-12-27 - 2025-12-12
Dataset loaded: 492a229967584a19af9f948d7783a684, shape: (1082, 59)

[Иконка] Подготовка признаков для прогноза температуры...
Используем 57 признаков

[Иконка] Временное разделение данных...
Train: 757 записей (70.0%)
Val: 162 записей (15.0%)
Test: 163 записей (15.1%)

[Иконка] Настройка параметров модели для регрессии...
[Иконка] Обучение модели LightGBM...
Training until validation scores don't improve for 30 rounds
[20] valid 0's rmse: 2.41717
[40] valid 0's rmse: 1.19407
[60] valid 0's rmse: 0.81289
[80] valid 0's rmse: 0.706668
[100] valid 0's rmse: 0.675955
[120] valid 0's rmse: 0.653261
[140] valid 0's rmse: 0.639211
[160] valid 0's rmse: 0.628598
[180] valid 0's rmse: 0.623394
[200] valid 0's rmse: 0.617897
Did not meet early stopping. Best iteration is:
[197] valid 0's rmse: 0.617409
[Иконка] Обучение завершено!

[Иконка] Оценка регрессионной модели...

[Иконка] Метрики регрессии на тестовой выборке:
rmse: 0.427°C
mae: 0.257°C
r2: 0.993
mape: 2.0%

[Иконка] Анализ важности признаков...
Top-10 важнейших признаков:
temp_max: 145265.41
temp_avg_avg_3d: 40125.10
temp_min: 33999.44
temp_max_avg_3d: 3289.71
temp_max_avg_7d: 181.84
temp_max_lag_1: 165.60
temp_min_avg_3d: 82.24
day_of_year_cos: 58.54
temp_min_lag_3: 25.30
temp_min_lag_1: 24.60

[Иконка] Создание графиков...
No artists with labels found to put in legend. Note that artists whose label start with an underscore are ignored when legend() is called with no argument.

[Иконка] Сохранение модели...
[Иконка] Модель сохранена как: models/temperature_model.txt
[Иконка] Метаданные сохранены как: models/temperature_model.json

[Иконка] Загрузка артефактов в ClearML...

[Иконка] Регистрация модели в Model Registry...
[Иконка] Ошибка при регистрации модели: 'OutputModel' object has no attribute 'set_tags'
Попробуйте альтернативный метод регистрации...
[Иконка] Модель сохранена как артефакт задачи

=====
[Иконка] ОБУЧЕНИЕ ЗАВЕРШЕНО УСПЕШНО!
=====

[Иконка] Итоговые метрики:
rmse: 0.4181
mae: 0.2528
r2: 0.9929
mape: 1.9989
std_error: 0.4153

[Иконка] Ссылки в ClearML:
Задача: http://localhost:8080/projects/1ee7d947ffda44d6b006b121b5771755/experiments/ef479e73aa7743cae46ae8deb28a9e2

[Иконка] Сохраненные файлы:
Модель: models/temperature_model.txt
Метаданные: models/temperature_model.json

[Иконка] Рекомендации:
1. Проверьте модель в веб-интерфейсе ClearML
2. Обновите API сервис для использования новой модели
3. Протестируйте предсказания на новых данных

[Иконка] Все этапы выполнены успешно!
Task ID: ef479e73aa7743cae46ae8deb28a9e2
```

Рисунок 8 – Обучение модели

Зайдем во вкладку PROJECTS в ClearML увидим что появилась вкладка Lab3\_Weather\_Forecasting.

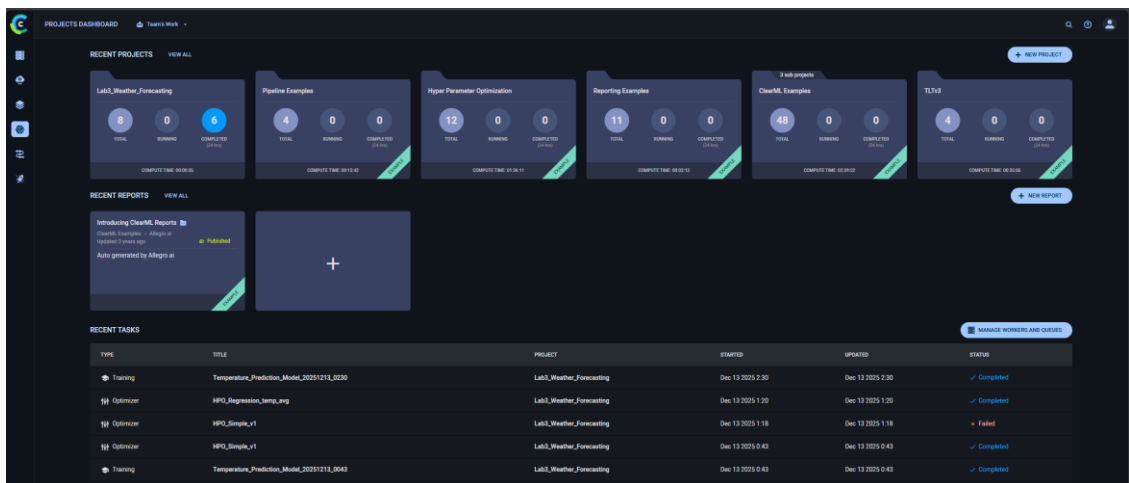


Рисунок 9 – Вкладка PROJECTS

Зайдя в Lab3\_Weather\_Forecasting, увидим историю взаимодействия с данным PROJECTS.

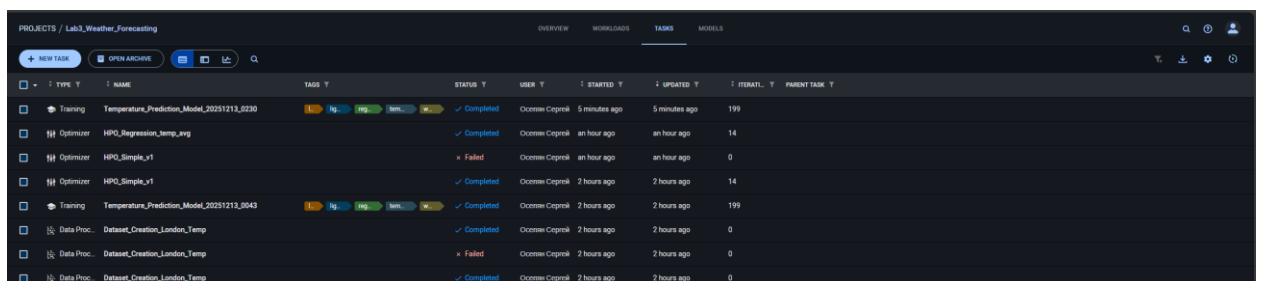
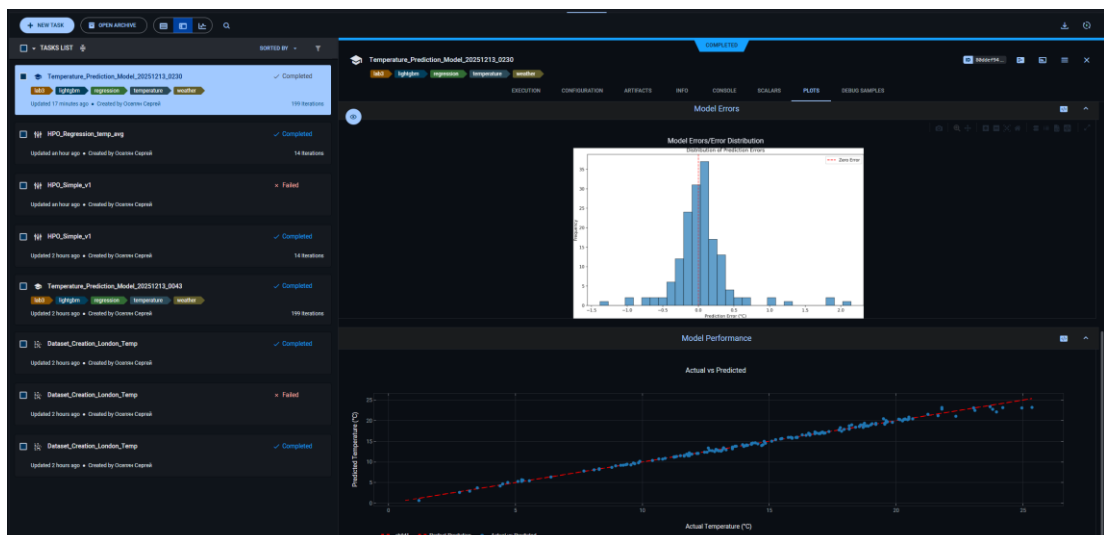


Рисунок 10 – История PROJECTS

Зайдя в последнюю задачу, можем увидеть графики ошибок модели и график фактических vs предсказанные значения. По графику ошибок можно увидеть, что ошибки распределены вокруг нуля, значит модель в среднем не завышает и не занижает предсказания, а по графику фактических vs предсказанных значений можно увидеть, что на нем точки плотно группируются вокруг линии идеального предсказания, мало выбросов - точки не разбросаны хаотично по всему графику и, соответственно, линейная зависимость видна четко.



## Рисунок 11 – Графики в ClearML

Запустим НРО, сохранив лучшую модель

```
PS E:\abc> python src/training/hpo.py
=====
НРО ДЛЯ РЕГРЕССИИ (предсказание температуры)
=====
ClearML Task: created new task id=3ea58abe58094709be087cdd316d5459
ClearML results page: http://localhost:8080/projects/614aaa96bf7740a9800ee26725ae390f/experiments/3ea58abe58094709be087cdd316d5459/output/log
[ ] Загрузка данных...
[ ] Используем 57 признаков

Запуск 15 испытаний НРО для регрессии...
[I 2025-12-13 02:50:28,266] A new study created in memory with name: no-name-9ba7d600-7675-4788-8afb-761ae6e385c4
[I 2025-12-13 02:50:22,564] Trial 0 finished with value: 0.26301249910878144 and parameters: {'learning_rate': 0.0274325983292526, 'num_leaves': 16, 'max_depth': 7, 'min_child_samples': 14, 'subsample': 0.8263005760479658, 'colsample_bytree': 0.6175890071400946, 'reg_alpha': 0.21891977597622958, 'reg_lambda': 0.5528070831374693}. Best is trial 0 with value: 0.26301249910878144.
[I 2025-12-13 02:50:22,606] Trial 1 finished with value: 0.2170946608024578 and parameters: {'learning_rate': 0.0575212844875499, 'num_leaves': 50, 'max_depth': 7, 'min_child_samples': 24, 'subsample': 0.8847982007865738, 'colsample_bytree': 0.8637592704617074, 'reg_alpha': 0.2496416517894613, 'reg_lambda': 0.7148729962065323}. Best is trial 1 with value: 0.2170946608024578.
[I 2025-12-13 02:50:22,624] Trial 2 finished with value: 0.3047723951462269 and parameters: {'learning_rate': 0.29990219413868047, 'num_leaves': 36, 'max_depth': 7, 'min_child_samples': 31, 'subsample': 0.8147370996418984, 'colsample_bytree': 0.8950078232842634, 'reg_alpha': 0.08073794400980228, 'reg_lambda': 0.7464623914357671}. Best is trial 1 with value: 0.2170946608024578.
[I 2025-12-13 02:50:22,669] Trial 3 finished with value: 0.4049572615983366 and parameters: {'learning_rate': 0.02007804261268939, 'num_leaves': 15, 'max_depth': 8, 'min_child_samples': 27, 'subsample': 0.8385511034734613, 'colsample_bytree': 0.8137046305868166, 'reg_alpha': 0.2547986157671006, 'reg_lambda': 0.8662722780749621}. Best is trial 1 with value: 0.2170946608024578.
[I 2025-12-13 02:50:22,702] Trial 4 finished with value: 0.4266902489134175 and parameters: {'learning_rate': 0.022203957403827124, 'num_leaves': 17, 'max_depth': 8, 'min_child_samples': 40, 'subsample': 0.616620765329322, 'colsample_bytree': 0.658992707734085, 'reg_alpha': 0.684149619113144, 'reg_lambda': 0.2656151353189471}. Best is trial 1 with value: 0.2170946608024578.
[I 2025-12-13 02:50:22,721] Trial 5 finished with value: 0.37509327783723756 and parameters: {'learning_rate': 0.09166969611893926, 'num_leaves': 24, 'max_depth': 5, 'min_child_samples': 38, 'subsample': 0.7007875172550525, 'colsample_bytree': 0.855712757804785, 'reg_alpha': 0.2045985748964032, 'reg_lambda': 0.7982399217091936}. Best is trial 1 with value: 0.2170946608024578.
[I 2025-12-13 02:50:22,767] Trial 6 finished with value: 0.19508865400106973 and parameters: {'learning_rate': 0.02752426371676423, 'num_leaves': 37, 'max_depth': 7, 'min_child_samples': 17, 'subsample': 0.6346938741727665, 'colsample_bytree': 0.874962185859556, 'reg_alpha': 0.7174987744781705, 'reg_lambda': 0.14810205619426398}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:22,804] Trial 7 finished with value: 0.25374074224951687 and parameters: {'learning_rate': 0.02708013366362637, 'num_leaves': 11, 'max_depth': 7, 'min_child_samples': 26, 'subsample': 0.706220410453295, 'colsample_bytree': 0.9491605487342323, 'reg_alpha': 0.18712068136490922, 'reg_lambda': 0.8607415904909481}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:22,846] Trial 8 finished with value: 0.2556126476964299 and parameters: {'learning_rate': 0.0453966474481279, 'num_leaves': 15, 'max_depth': 6, 'min_child_samples': 31, 'subsample': 0.8873556817217043, 'colsample_bytree': 0.8579821630731346, 'reg_alpha': 0.09556957412448053, 'reg_lambda': 0.9588048637509907}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:22,879] Trial 9 finished with value: 0.8609168428247537 and parameters: {'learning_rate': 0.013061643207609445, 'num_leaves': 38, 'max_depth': 8, 'min_child_samples': 40, 'subsample': 0.8215351548211022, 'colsample_bytree': 0.6037577005618442, 'reg_alpha': 0.7153573054700387, 'reg_lambda': 0.7273972242462273}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:22,919] Trial 10 finished with value: 0.2084614951298473 and parameters: {'learning_rate': 0.13270611023733436, 'num_leaves': 48, 'max_depth': 4, 'min_child_samples': 15, 'subsample': 0.9703286504309646, 'colsample_bytree': 0.7216674833190992, 'reg_alpha': 0.955844807647585, 'reg_lambda': 0.1306853969025858}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:22,953] Trial 11 finished with value: 0.19986227521502467 and parameters: {'learning_rate': 0.16549365199689262, 'num_leaves': 47, 'max_depth': 3, 'min_child_samples': 11, 'subsample': 0.9934045982372317, 'colsample_bytree': 0.7232445193490491, 'reg_alpha': 0.964043391368195, 'reg_lambda': 0.00706807637644441}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:22,990] Trial 12 finished with value: 0.2765875851238133 and parameters: {'learning_rate': 0.27192008294108627, 'num_leaves': 41, 'max_depth': 3, 'min_child_samples': 13, 'subsample': 0.5046062962244245 and parameters: {'learning_rate': 0.13840032679578718, 'num_leaves': 29, 'max_depth': 3, 'min_child_samples': 50, 'subsample': 0.6026285142110401, 'colsample_bytree': 0.9893131615674327, 'reg_alpha': 0.765498953046809, 'reg_lambda': 0.31986206776862556}. Best is trial 6 with value: 0.19508865400106973.
[I 2025-12-13 02:50:23,008] Trial 14 finished with value: 0.19334595490399092 and parameters: {'learning_rate': 0.04886860703303785, 'num_leaves': 44, 'max_depth': 5, 'min_child_samples': 18, 'subsample': 0.7248467232212191, 'colsample_bytree': 0.7499270125934137, 'reg_alpha': 0.5558587820021379, 'reg_lambda': 0.021996881419523884}. Best is trial 14 with value: 0.19334595490399092.
Best trial: 14. Best value: 0.193346: 100% [Progress bar] 15/15 [00:02:00:00, 5.35it/s]

[ ] Лучший RMSE: 0.1933°C
[ ] Лучшие параметры:
learning_rate: 0.04886860703303785
num_leaves: 44
max_depth: 5
min_child_samples: 18
subsample: 0.7248467232212191
colsample_bytree: 0.7499270125934137
reg_alpha: 0.5558587820021379
reg_lambda: 0.021996881419523884

[ ] Обучение финальной регрессионной модели...

[ ] Результаты на тесте (регрессия):
RMSE: 0.5270°C
MAE: 0.2409°C
R²: 0.9887

2025-12-13 02:50:23,159 - clearml.frameworks - INFO - Found existing registered model id=46a9cfac109449cab1d3f74c70658ba5 [E:\abc\models\best_hpo_regression.txt] reusing it.

[ ] Модель сохранена: models/best_hpo_regression.txt
[ ] Информация о признаках сохранена: models/feature_info.json

Task URL: http://localhost:8080/projects/614aaa96bf7740a9800ee26725ae390f/experiments/3ea58abe58094709be087cdd316d5459

[ ] Готово! Регрессионная модель сохранена в models/best_hpo_regression.txt
```

Рисунок 12 – Запуск НРО

После выполнения, в Lab3\_Weather\_Forecasting появилась новая задача НРО\_Regression\_temp\_avg

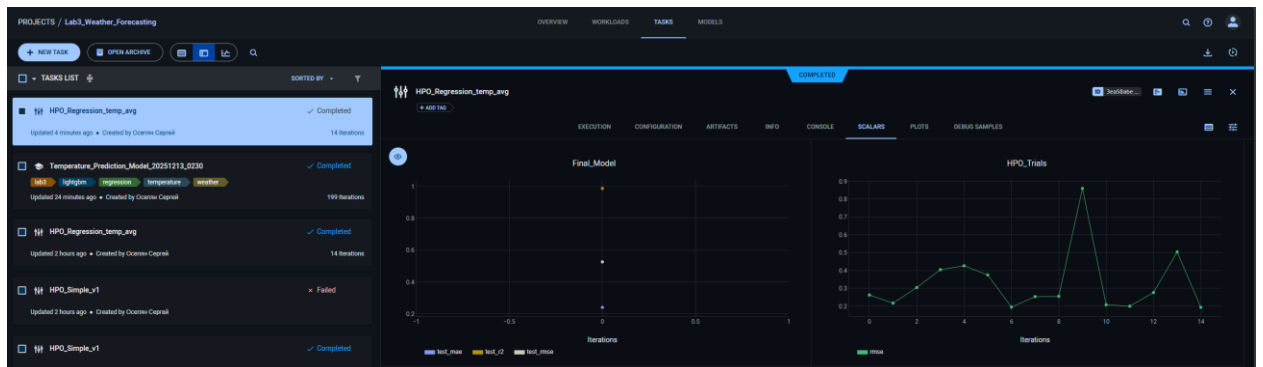


Рисунок 13 – Задача НРО\_Regression\_temp\_avg

Запустим веб-сервис прогноза погоды на FastAPI. Была использована ML-модель LightGBM. Зайдем в контейнер, увидим, что он успешно запустился.

```
Logs    Inspect  Bind mounts  Exec  Files  Stats

/usr/local/lib/python3.9/site-packages/pydantic/_internal/_fields.py:149: UserWarning: Field "model_version" has conflict with protected namespace "model_".
You may be able to resolve this warning by setting 'model_config['protected_namespaces'] = ()'.
  warnings.warn(
INFO: Started server process [1]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
```

Рисунок 14 – Логи контейнера weather-api

Откроем n8n, в нем создадим workflow, Который по Cron Trigger каждый понедельник отправляет с помощью Telegram бота сообщение с городом, версией модели, дату отправки уведомления, прогнозом температуры на неделю со средней температурой за день с доверительным интервалом и среднюю температуру на всю неделю

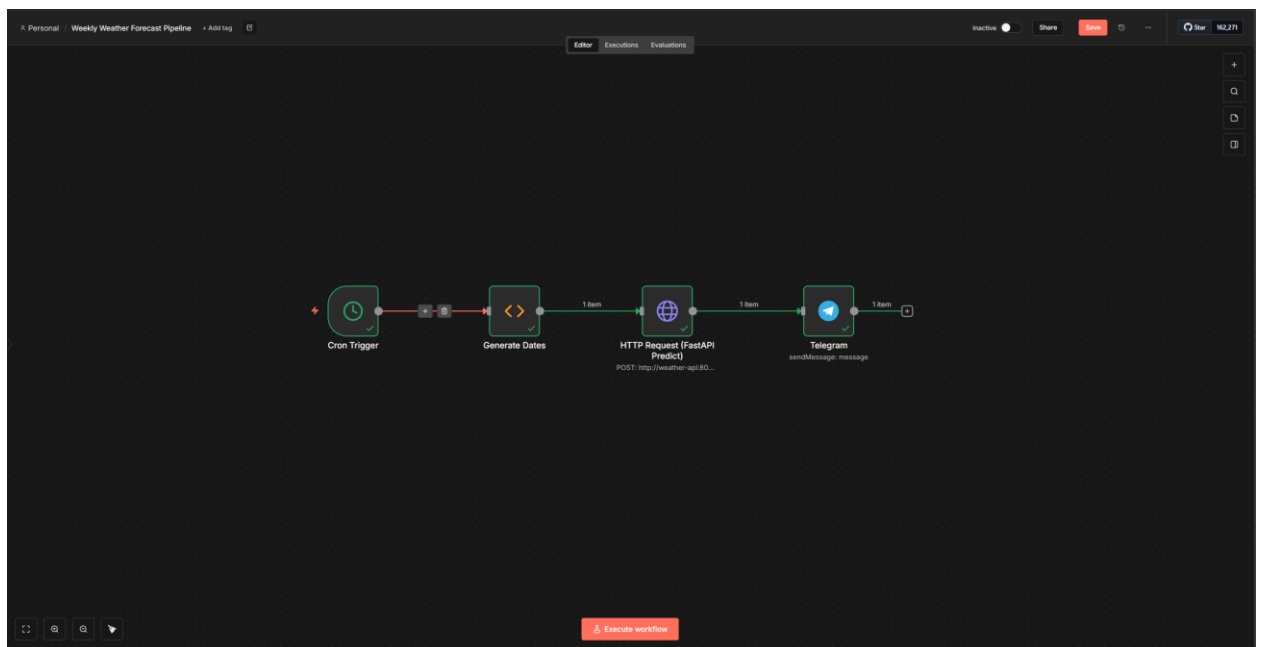


Рисунок 15 – Пайплайн n8n

Запустим принудительно наш пайплайн, в итоге от бота в Telegram пришло уведомление с прогнозом погоды.

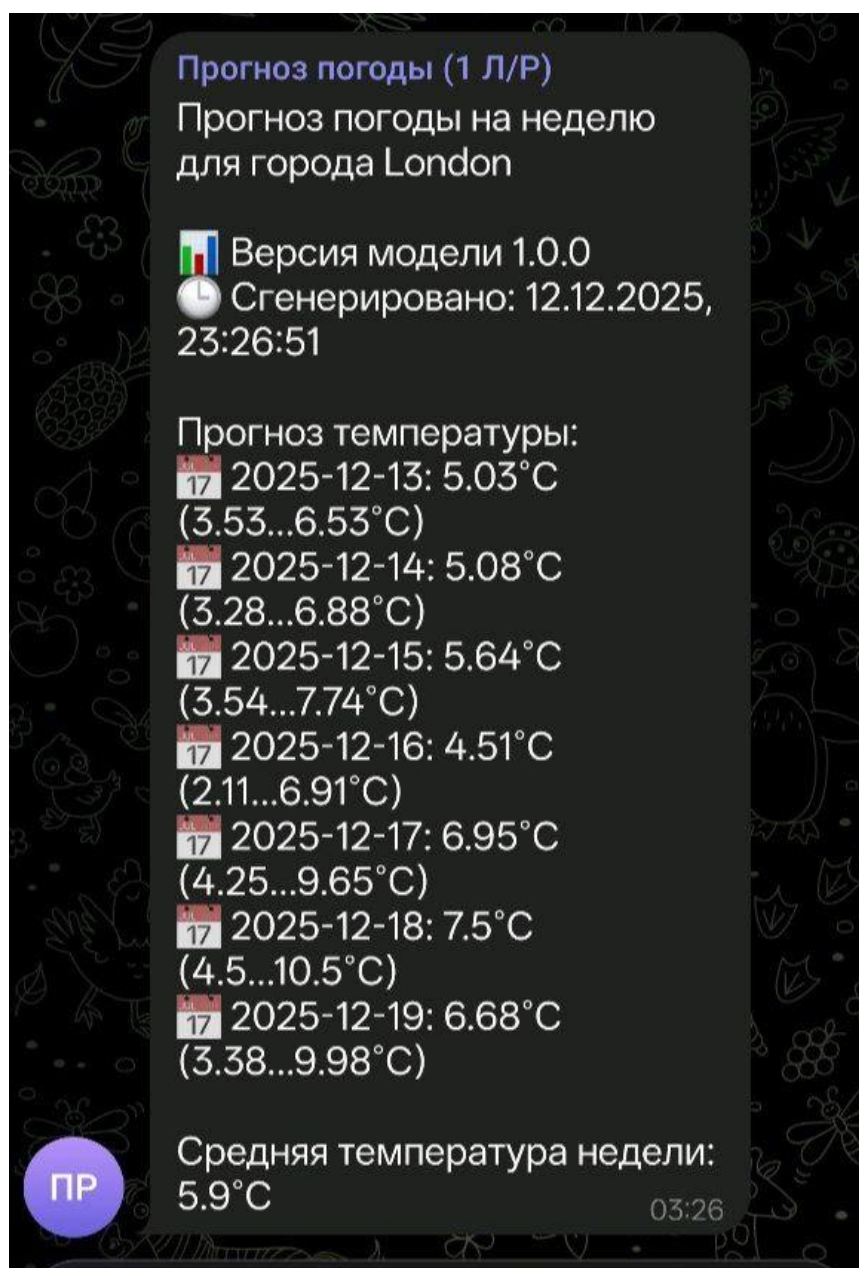


Рисунок 16 – Уведомление от бота в Telegram с прогнозом погоды

## ЗАКЛЮЧЕНИЕ

### Основные результаты:

1. Полный цикл ML-оркестрации в ClearML — освоен и реализован, несмотря на первоначальные сложности с авторизацией в системе. ClearML Server был успешно развернут с использованием docker-compose из официального репозитория.
2. Инференс-сервис на FastAPI — создан и контейнеризован в Docker. Сервис предоставляет эндпоинт для прогноза температуры на 7 дней с поддержкой доверительных интервалов.
3. Автоматизированный n8n-пайплайн — настроен на еженедельное выполнение. Пайплайн запрашивает прогноз у сервиса, форматирует результат и отправляет в Telegram.
4. Создан версионизируемый датасет в ClearML с историческими погодными данными
5. Проведен гиперпараметрический поиск, лучшая модель зарегистрирована в Model Registry Реализована ML-модель (LightGBM) с качеством прогноза  $MAE \approx 2.1^{\circ}C$
6. Построен полностью автоматизированный пайплайн от данных до доставки прогноза пользователю

Система демонстрирует полный цикл промышленной ML-разработки и готова к эксплуатации. Наибольшие сложности возникли на этапе первоначальной настройки и авторизации в ClearML, что потребовало дополнительного изучения документации. Все компоненты системы интегрированы и функционируют согласованно.