

PGConf.CПб 2023



ML in SQL

Александр Календа рёв

Что будет:

Основы ML

Обзор ML в БД (SQL)

Как это все связано

ML in Postgres – это возможно

Чего не будет:

Нудной теории

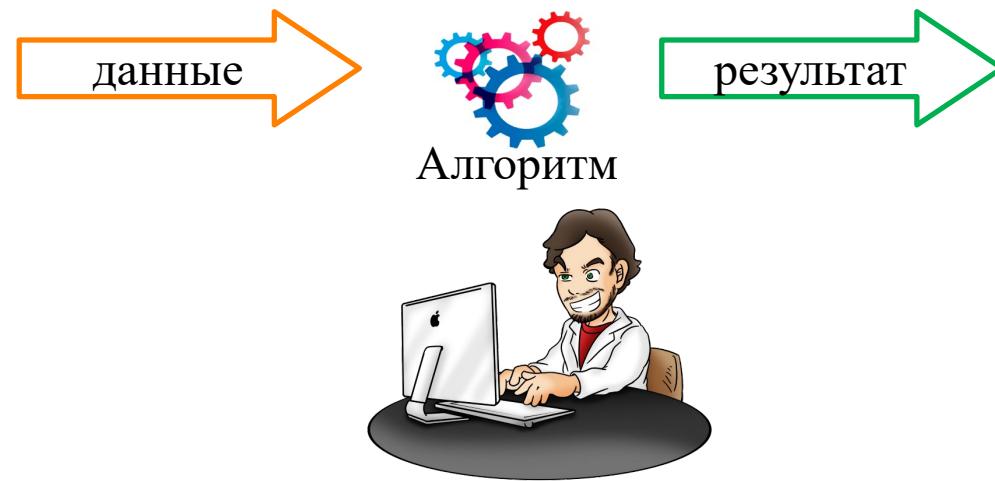
Демонстрации с другими БД

Поговорим о ML

Традиционное программирование



Традиционное программирование



Традиционное программирование



Машинное обучение

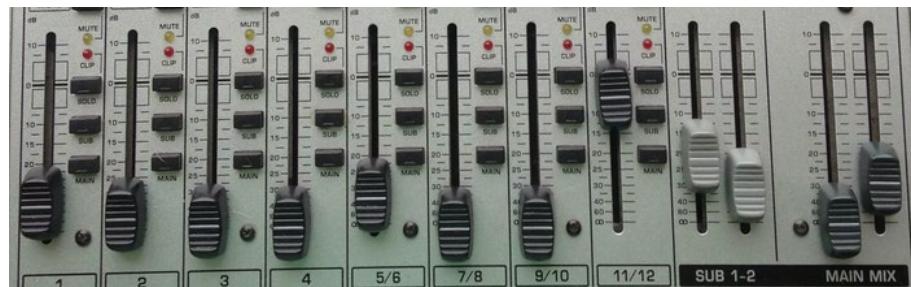


Этап 1 – тренировка/обучение модели

Традиционное программирование



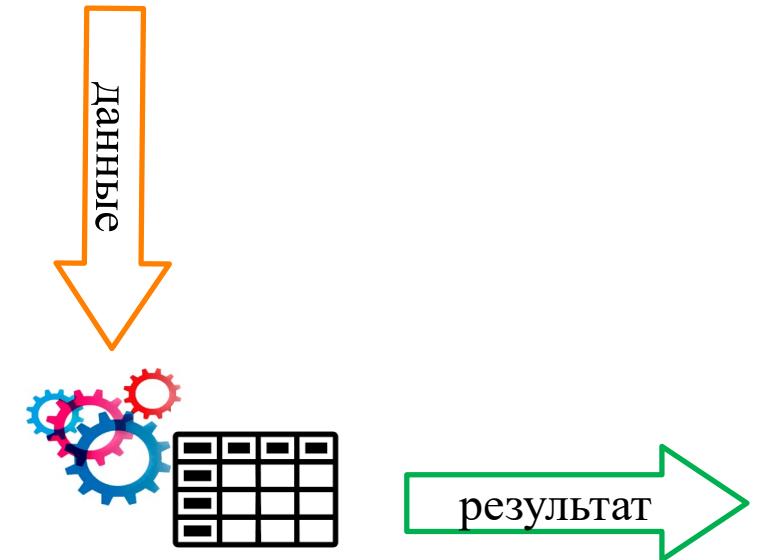
Машинное обучение



Традиционное программирование

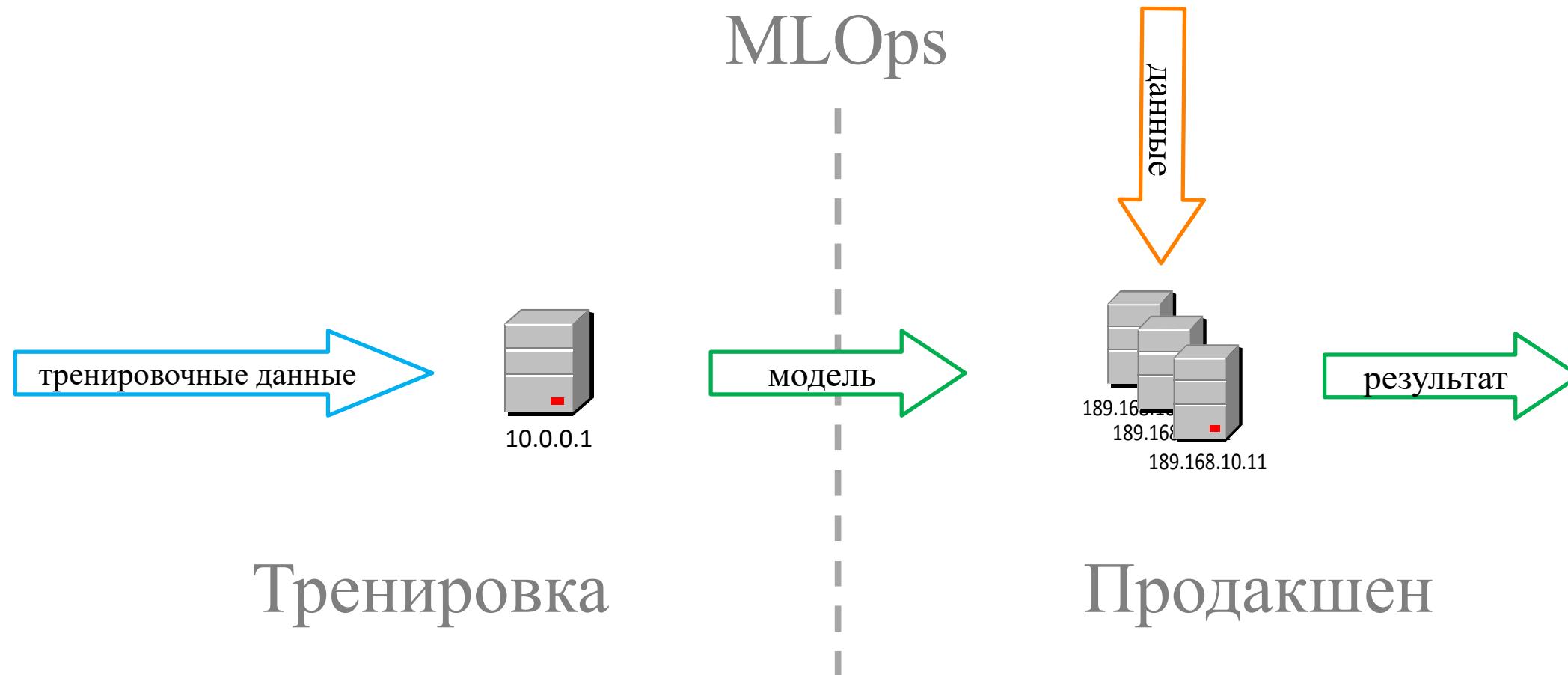


Машинное обучение



Этап 2 – эксплуатация модели

Перенос моделей требует формат сохранения



Фреймворки ML



TensorFlow (Google) : R, C#, C++, Haskell, Java, Go и Swift.



CatBoost (Yandex) C/C++ (только анализ) Python, R, cmd

XGBoost C++, Java, Python, R, Julia, Perl и Scala.



MxNet: C++, Python, Java, Julia, MATLAB, JavaScript, Go, R, Scala, Perl, и Wolfram



Torch (Facebook) Lua, PyTorch - Python



LightGBM (Microsoft) Python, R, C (не полный АПИ)



Microsoft Cognitive Toolkit: Python, C# или C++ .NET

Универсальный формат моделей ONNX

- Тренируем на одном фреймворке
- В продакшен на другой сервер
- И можем использовать другой фреймворк

Какие задачи решают с помощью ML

- Бинарная классификация
- Классификация (несколько классов)
- Регрессия
- Кластеризация
- Обнаружение аномалий
- Ранжирование
- Рекомендации
- Прогнозирование
- Анализ временных рядов

ML in DB

Что под капотом



Использует хранимые процедуры Python & R

```
EXECUTE sp_execute_external_script
@language = N'Python',
@script = N'
import pkg_resources
import pandas
OutputDataSet = pandas.DataFrame(sorted([(i.key, i.version) for i in pk
WITH result sets((Package NVARCHAR(128), Version NVARCHAR(128)));
```

Импортирует пакет microsoftml

Получаем dataset



Использование обученной модели

Считаем ошибку на тестовой выборке по метрике Logloss

```
SELECT -avg((Target * log(prob)) +
           ((1. - Target) * log(1. - prob))) AS logloss
FROM
(
  SELECT
    modelEvaluate('purchase_model', *) AS pred,
    1. / (1. + exp(-pred)) AS prob,
    Target
  FROM catBoostPool('test.cd', 'test.csv')
)
```

logloss
0.03719106223177089

Есть функции:

- линейной регрессии
- стохастического градиентного спуска

Работает с обученными моделями (libcatboostmodel)

Модели встраиваем через конфиг

Получаем dataset через пул



Amazon Redshift

```
CREATE MODEL demo_ml.customer_churn_model
FROM (SELECT state,
             area_code,
             total_charge/account_length AS average_daily_spend,
             cust_serv_calls/account_length AS average_daily_cases,
             churn
      FROM demo_ml.customer_activity
     WHERE record_date < '2020-01-01'
    )
TARGET churn
FUNCTION predict_customer_churn
IAM_ROLE 'arn:aws:iam::<accountID>:role/RedshiftML'
SETTINGS (
    S3_BUCKET 'redshiftml-<your-account-id>'
)
```

XGBoost

- Полноценная модель ML
- Может обучать и предсказывать
- Импортировать существующие модели
- Результат dataset



Amazon Redshift

XGBoost

Использование модели

```
SELECT area_code || phone accountid,  
       demo_ml.predict_customer_churn(  
           state,  
           area_code,  
           total_charge/account_length ,  
           cust_serv_calls/account_length )  
      AS "predictedActive"  
  FROM demo_ml.customer_activity  
 WHERE area_code='408' and record_date > '2020-01-01';
```

```
WITH infer_data AS (  
    SELECT area_code || phone accountid, churn,  
          demo_ml.predict_customer_churn(  
              state,  
              area_code,  
              total_charge/account_length ,  
              cust_serv_calls/account_length ) AS predicted  
     FROM demo_ml.customer_activity  
    WHERE record_date < '2020-01-01'  
)  
SELECT *  FROM infer_data where churn!=predicted;
```



```
{CREATE MODEL | CREATE MODEL IF NOT EXISTS | CREATE OR REPLACE MODEL}
model_name
[OPTIONS(MODEL_TYPE = { 'KMEANS' },
    NUM_CLUSTERS = int64_value,
    KMEANS_INIT_METHOD = { 'RANDOM' | 'KMEANS++' | 'CUSTOM' },
    KMEANS_INIT_COL = string_value,
    DISTANCE_TYPE = { 'EUCLIDEAN' | 'COSINE' },
    STANDARDIZE_FEATURES = { TRUE | FALSE },
    MAX_ITERATIONS = int64_value,
    EARLY_STOP = { TRUE | FALSE },
    MIN_REL_PROGRESS = float64_value,
    WARM_START = { TRUE | FALSE }
)];
```

- Полноценная модель ML
- Может обучать и предсказывать
- Импортировать существующие модели
- Результат dataset



Google
BigQuery

TensorFlow

AutoML
XGBoost



PostgresPro

```
CREATE MODEL `project_id.mydataset.mymodel`
OPTIONS(MODEL_TYPE='MATRIX_FACTORIZATION') AS
SELECT
    user,
    item,
    rating
FROM `mydataset.mytable`
```



```
SELECT *  
FROM  
ML.PREDICT(  
    MODEL `mydataset.mymodel`,  
    ( SELECT  
        user,  
        item,  
        rating  
    FROM `mydataset.mytable`  
    )  
)
```



O'REILLY®

Google BigQuery

Всё о хранилищах данных, аналитике и машинном обучении



Валиаппа Лакшманан
Джордан Тайджани

Google BigQuery. Всё о хранилищах данных, аналитике и машинном обучении [2021]
Валиаппа Лакшманан, Джордан Тайджани

Вас пугает необходимость обрабатывать петабайтные наборы данных? Познакомьтесь с Google BigQuery, – системой хранения информации, которая может консолидировать данные по всему предприятию, облегчает интерактивный анализ и позволяет реализовать задачи машинного обучения. Теперь вы можете эффективно хранить, запрашивать, получать и изучать данные в одной удобной среде.

Валиаппа Лакшманан и Джордан Тайджани научат вас работать в современном хранилище данных, используя все возможности масштабируемого, бессерверного публичного облака.

А ЧТО В PostgreSQL ?



<https://github.com/apache/madlib>

madlib/dpage@PostgreSQL 12 ~

Query Editor Query History

```

1 -- Drop tables from previous runs
2 DROP TABLE IF EXISTS housing_linregr, housing_linregr_summary;
3
4 -- Train the model
5 SELECT madlib.linregr_train( 'housing',
6                             'housing_linregr',
7                             'medv',
8                             'ARRAY[crim, zn, indus, chas, nox, rm, age, dis, rad, tax, ptratio, b, lstat]'
9                           );
10 -- Predict the house prices, and compare to the actual data
11 SELECT housing.*,
12        predict,
13        medv - predict AS residual,
14        sqrt(avg(power(abs(medv - predict), 2)) OVER ()) AS rmse
15 FROM housing,
16      housing_linregr,
17      madlib.linregr_predict(coef,
18                             ARRAY[crim, zn, indus, chas, nox, rm, age, dis, rad, tax, ptratio, b, lstat]
19                             ) predict;

```

Data Output Explain Notifications Messages

tax double precision	ptratio double precision	b double precision	lstat double precision	medv double precision	predict double precision	residual double precision	rmse double precision
296	15.3	396.9	4.98	24	29.098263530097405	-5.098263530097405	4.915902697381886
242	17.8	396.9	9.14	21.6	24.502275482309766	-2.902275482309765	4.915902697381886
242	17.8	392.83	4.03	34.7	31.227426410729453	3.4725735892705494	4.915902697381886
222	18.7	394.63	2.94	33.4	29.70710350127486	3.692896498725137	4.915902697381886
222	18.7	396.9	5.33	36.2	29.56479571582512	6.635204284174883	4.915902697381886
222	18.7	394.12	5.21	28.7	25.29376223617479	3.40623776382521	4.915902697381886
311	15.2	395.6	12.43	22.9	21.530411609932766	1.369588390067232	4.915902697381886
311	15.2	396.9	19.15	27.1	19.10333425509494	7.996665744905062	4.915902697381886



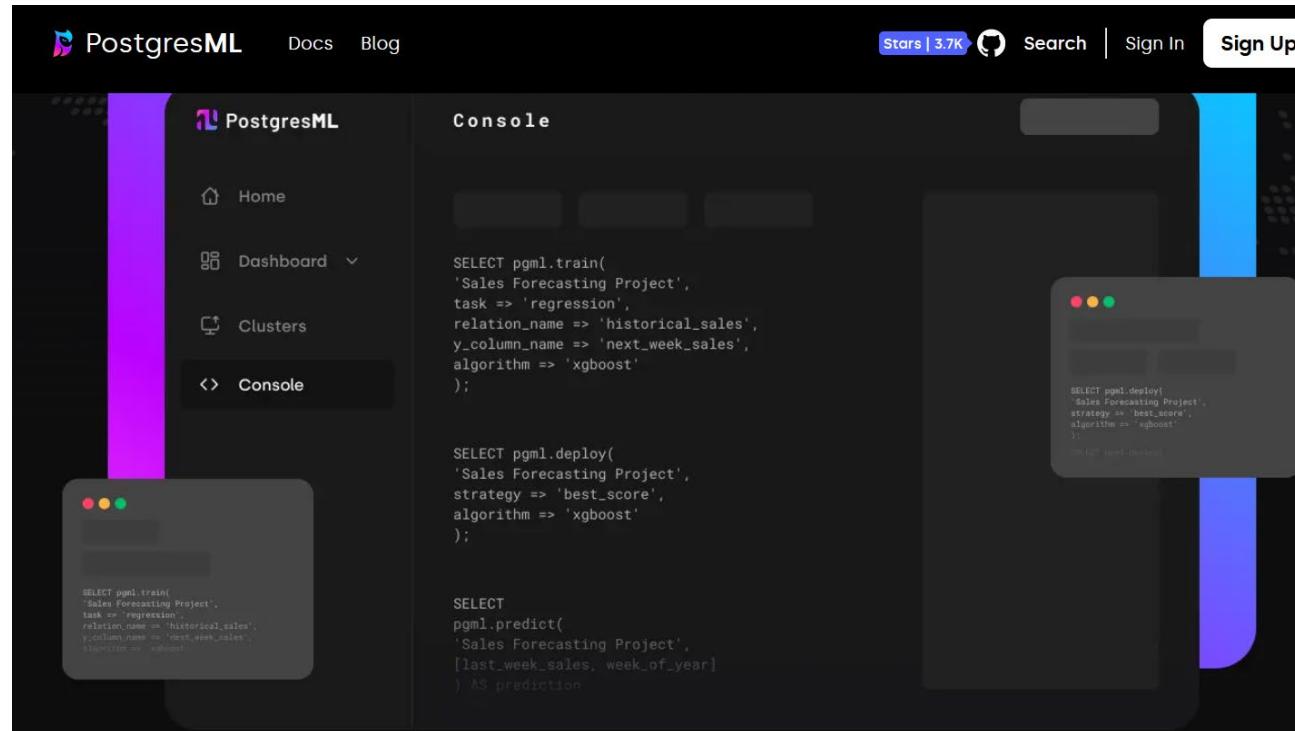
ПОД КАПОТОМ
PL/Python C/C++

25 алгоритмов

коду 9-10 лет
В коммитов в 2023 г – около 10

Postgres ML

 <https://github.com/postgresml/postgresml>



The screenshot shows the GitHub repository for PostgresML. The repository has 3.7K stars. The main page features a sidebar with links for Home, Dashboard (which is currently selected), Clusters, and Console. The main content area displays three SQL code snippets:

```

SELECT pgml.train(
    'Sales Forecasting Project',
    task => 'regression',
    relation_name => 'historical_sales',
    y_column_name => 'next_week_sales',
    algorithm => 'xgboost'
);

SELECT pgml.deploy(
    'Sales Forecasting Project',
    strategy => 'best_score',
    algorithm => 'xgboost'
);

SELECT
pgml.predict(
    'Sales Forecasting Project',
    [last_week_sales, week_of_year]
) AS prediction
  
```

XGBoost

 Yandex
CatBoost



 scikit-learn
machine learning in Python

PL/Rust под капотом

Порт в BERT GPT (NLP)

31 алгоритм

Проект с 2022г - живой



Работает только с обученными моделями
(libcatboostmodel)
Модели динамически загружаются
Получаем таблицу с которой можно
работать

```
ml=# select ml_predict('/usr/local/pgsql/example/titanic.cbm', 'titanic2');
WARNING: ALTER TABLE IF EXISTS titanic2_predict SET SCHEMA public; res=4
NOTICE: processed=418
      ml_predict
```

```
-----
public.titanic2_predict
(1 row)
```

```
ml=# select * from titanic2_predict limit 5;
```

row	passenger_id	pclass	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked	predict
1	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	-999	Q	0.142616
2	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47	1	0	363272	7	-999	S	0.310916
3	894	2	Myles, Mr. Thomas Francis	male	62	0	0	240276	9.6875	-999	Q	0.083718
4	895	3	Wirz, Mr. Albert	male	27	0	0	315154	8.6625	-999	S	0.125321
5	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22	1	1	3101298	12.2875	-999	S	0.39712

Типы входных данных:

Категориальные (текст)
categorical feature

Числовые
feature

Временные ряды
time-series feature

Типы входных данных

```
postgres@notebook-sasha: /usr/local/pgsql
adult=# \d titanic
           Table "public.titanic"
  Column   |      Type       | Collation | Nullable | Default
-----+-----+-----+-----+-----+
id      | integer        |           |          |
passenger_id | integer        |           |          |
pclass    | integer        |           |          |
name     | text            |           |          |
sex      | text            |           |          |
age      | double precision |           |          |
sibsp    | integer        |           |          |
parch    | integer        |           |          |
ticket   | text            |           |          |
fare     | double precision |           |          |
cabin    | text            |           |          |
embarked | character(1)  |           |          |
res      | boolean         |           |          |
adult=# select * from titanic limit 10;
```

Особенность libcatboostmodel:

```
:ml=# SELECT ml_info('/usr/local/pgsql/example/titanic.cbm');
               ml_info
-----
dimension:1 numeric features:2 cagorial features:9
(1 row)
```

Типы входных данных



Особенность libcatboostmodel:
Integer, bool – категориальные
feature

```
:ml=# SELECT ml_info('/usr/local/pgsql/example/titanic.cbm');
          ml_info
-----
 dimension:1 numeric features:2 cagorial features:9
(1 row)
```

Ограничения

Работает только с обученными моделями (libcatboostmodel)

Поля должны быть в том же порядке, как и в исходном датасете

catboost-for-titanic-top-7.ipynb

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка Последнее изменение: 7 сентября

+ Код + Текст

[]

res

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Survived
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	-999	Q	
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	-999	S	
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	-999	Q	
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	-999	S	
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	-999	S	
...	
413	1305	3	Spector, Mr. Woolf	male	-999.0	0	0	A.5. 3236	8.0500	-999	S	
414	1306	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108.9000	C105	C	
415	1307	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7.2500	-999	S	
416	1308	3	Ware, Mr. Frederick	male	-999.0	0	0	359309	8.0500	-999	S	
417	1309	3	Peter, Master. Michael J	male	-999.0	1	1	2668	22.3583	-999	C	

418 rows × 12 columns

```
adult=# \d titanic
Table "public.titanic"
 Column | Type | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 id | integer | | | |
 passenger_id | integer | | | |
 pclass | integer | | | |
 name | text | | | |
 sex | text | | | |
 age | double precision | | | |
 sibsp | integer | | | |
 parch | integer | | | |
 ticket | text | | | |
 fare | double precision | | | |
 cabin | text | | | |
 embarked | character(1) | | | |
 res | boolean | | | |
```

```
adult=# select * from titanic limit 10;
```



Планы

Сделать полноценный SQL интерфейс

Оттестировать все виды моделей

Добавить параметры настройки, функции оценки

Заменить на **XGBoost**, возможность обучать модель

**XGBoost**

Как это должно работать на самом деле?

CREATE MODEL <name> <options> <QUERY>

- Установить обработчик на QueryParser (Optimizer)

Синтаксический анализ запроса:

Есть лексема MODEL?

- Нет: стандартная обработка
- Да:
 - Создание объекта модели
 - Сохранение метаданных (tbl pg_ml_model)
 - Выполнение запроса
 - Тренировка и сохранение модели Model.Fit(Dataset) в bin формате



Как это должно работать на самом деле?

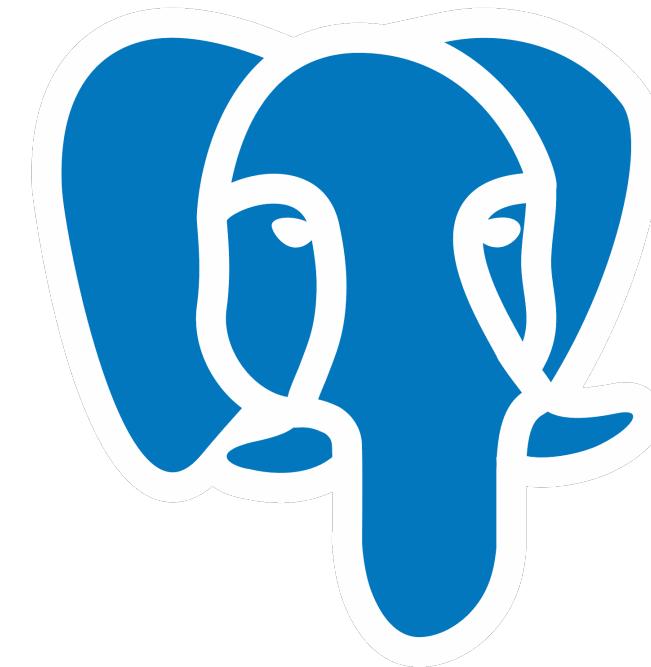
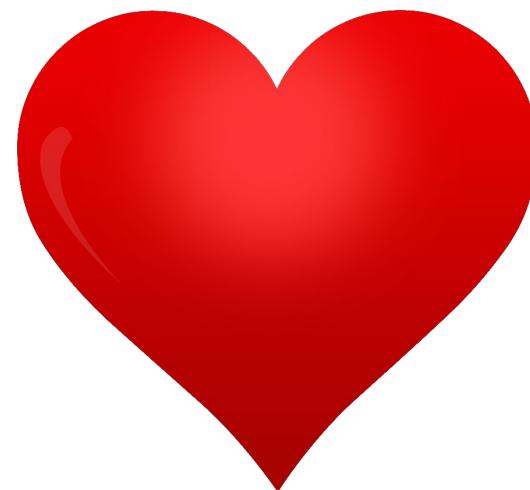
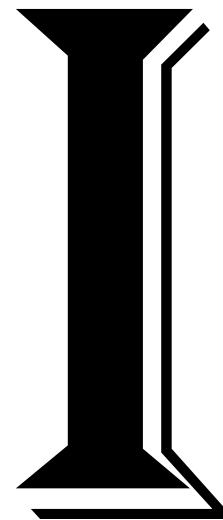
PREDICT <name> <QUERY>

- Установить обработчик на QueryParser (Optimizer)
Проверить есть лексема PREDICT?
 - Нет: стандартная обработка
 - Да:
 - Поиск модели в метаданных (tbl pg_ml_model)
 - Загрузка объекта модели
 - Выполнение запроса
 - Применение Model.Predict(Row) к каждой строке
 - Формирование выходного набора данных

Приглашаются желающие
к сотрудничеству



Давайте делать PostgreSQL лучше



PGConf.СПб 2023



Благодарности

За основную идею

Московский Финансовый Университет

Сахнюк Павел Анатольевич

За консультации по CatBoost

Stanislav Kirillov @kizillMikhail

Osipov @toomish

Q&A

Спасибо!



@akalend



akalend@mail.ru