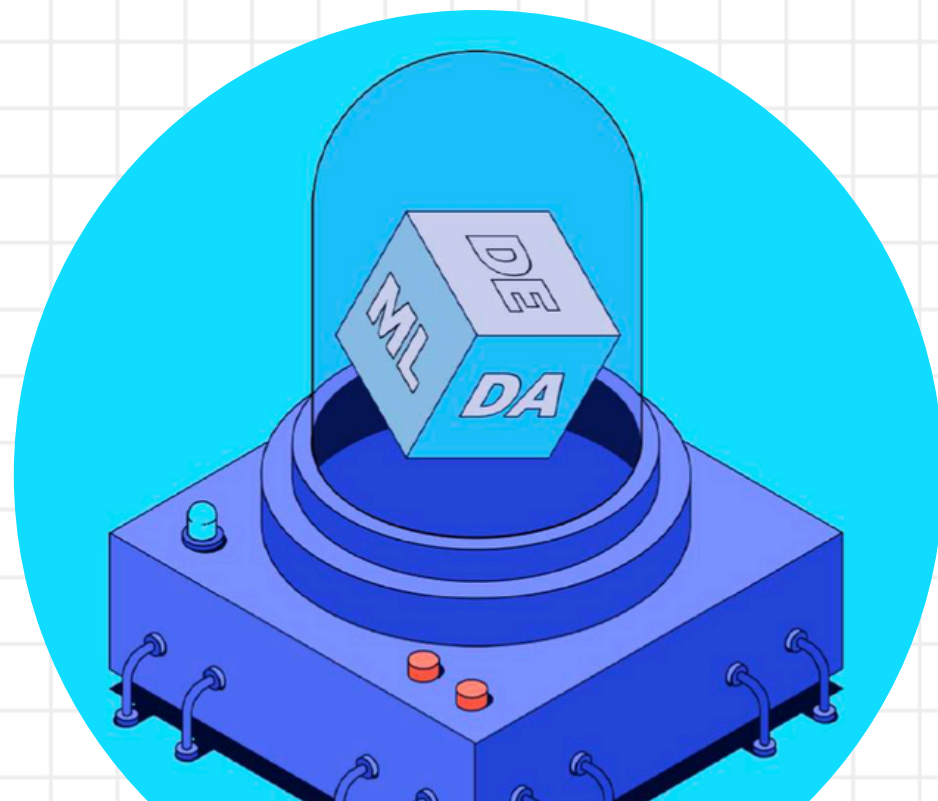


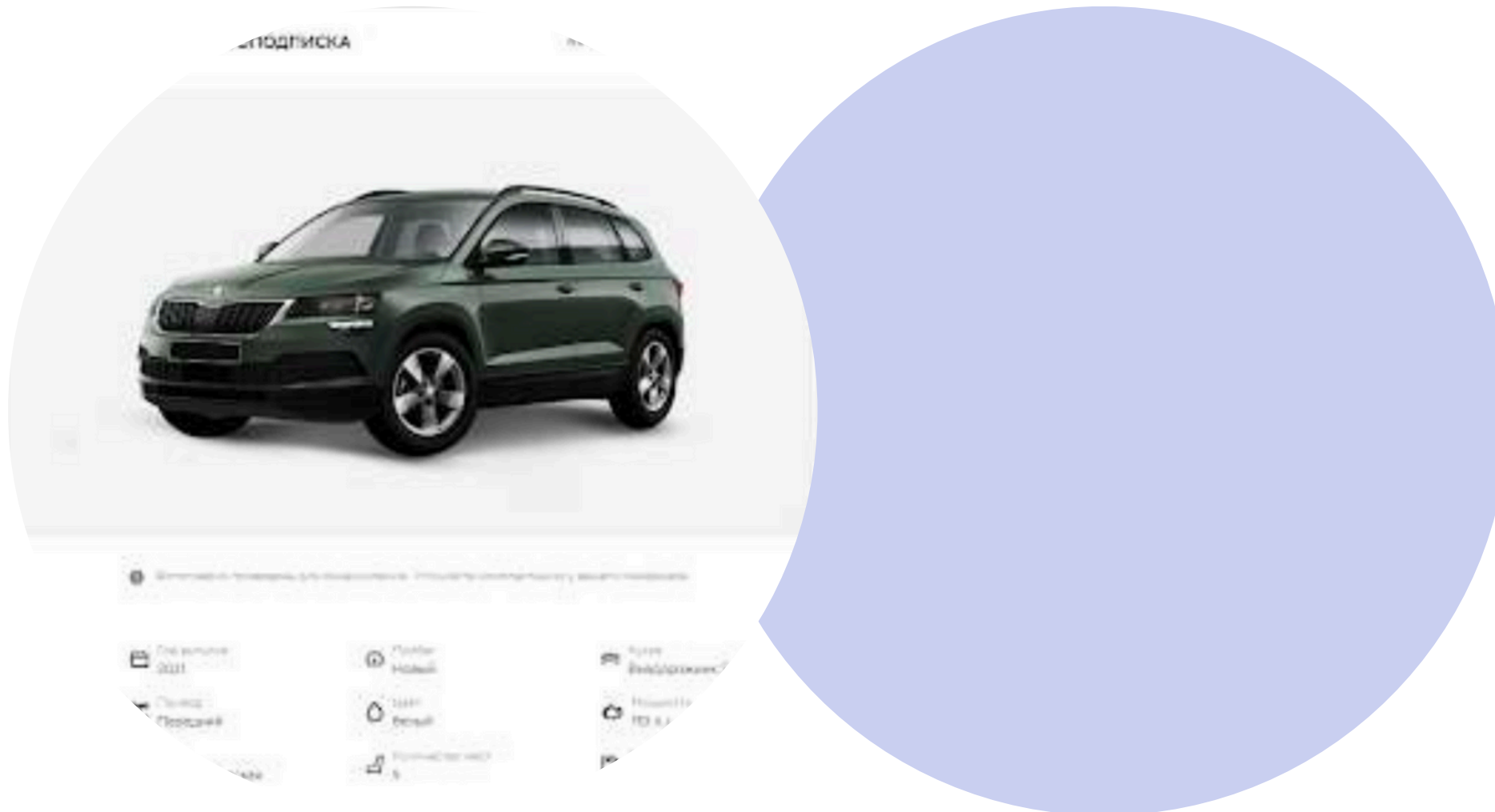
ML-ИНЖЕНЕР

ОБУЧЕНИЕ МОДЕЛИ НА ДАННЫХ С САЙТА «СБЕРАВТОПОДПИСКА»



Итоговая работа Жураева Абду Саида
Курс “Введение в Data Science”

Задача: Обучить ML модель прогнозировать Conversion Rate



Разведочный анализ и обработка данных

Анализ предоставленных данных из Google Analytics по сайту «СберАвтоподписка» в виде 2 таблиц GA Sessions и GA Hits.

Обучение ML модель

На основе обработанных данных подобрать и обучить модель для предсказания совершения целевого действия пользователем сайта. Метрика для оценки модели ROC-AUC

Развернуть сервис в виде API

Упаковать получившуюся модель в сервис, который будет брать на вход все атрибуты, типа `utm_*`, `device_*`, `geo_*`, и отдавать на выход 0/1.

Ключевые тезисы и действия с данными в `ga_hits`

!

15 726 470 записей
11 колонок

!

`ga_hits` содержит информацию о всех событиях связанных с сессиями по `session_id`

!

На основе данных из `ga_hits` можно определить совершено ли целевое действие

!

Необходимо создать новую колонку `Target Action` со значение `1` если данное действие целевое и `0` если нет

!

Имея колонку `Target Action` в `ga_hits` мы создадим целевую переменную `Conversion Rate` в Датафрейме `ga_sessions`

GA_Hits Dataframe

Ключевые тезисы разведочного анализа данных `ga_session`

!

**1 860 042 записей
18 колонок**

!

**DataFrame содержит
информацию о всех
уникальных сессиях**

!

**Содержит важные
для задачи признаки
`utm_*`, `device_*`,
`geo_*`**

!

**Не содержит данных
о целевом действии**

!

**На основе `ga_session`
будет создан
датафрей с данными
из технического
задания**

GA_Session
Dataframe

Необходимые преобразования данных в `ga_session`

I

Отбор только нужных нам колонок согласно поставленной задачи в ТЗ (utm*, device*, geo*)

GA_Session
Dataframe

II

Стандартизация пустых значений, приравниваем '(not set)' и '' к np.nan

III

Для балансировки данных применим даунсэмплинг. Сократим количество записей с cr=0

IV

На основе данных из device_*, заполнение пустот в колонках device_brand и device_os


V

Заполнить оставшиеся пустоты и стандартизировать все значения (OnehotEncode, StandartScaler)

Данные которые будут использованы

Признаки которые будет
обработать наш сервис.
Все признаки являются
категориальными

Целевая переменная,
которую мы будем
предсказывать



- utm_source
- utm_medium
- utm_campaign
- utm_adcontent
- utm_keyword
- device_category
- device_os
- device_brand
- device_model
- device_screen_resolution
- device_browser
- geo_country
- geo_city
- conversion_rate

Feature Engineering

Первоначальный список новых фитч
предположительно положительно влияющих на
метрику ROC-AUC

При проверке корреляции никакие из
созданных признаков не показали достойных
показателей. Было решено протестировать
влияние фитч после выбора модели обучения

- is_organic_visit
- device_screen_width и device_screen_height
- is_socialmedia_advert
- device_display_megapixel
- device_orientation_vertical
- from_russia
- from_moscow
- 'Population', 'Timezone', 'km_to_moscow' по
имени города из внешних источников

Основные моменты при выборе модели

Необходимо предсказывать принадлежность классу $CR=1$ или 0 .

Задача предполагает выявление фич влияющих на $CR=1$ и $CR=0$ по отдельности.

Метрика оценки: ROC-AUC

Время предсказания по API: 3 секунды

Ресурсы: MacBook Air M1 16GB DDR

Вывод: лучше всего подойдет классификатор на линейной функции



Выбор ML модели

Показатели метрики тестируемых классификаторов

**ROC-AUC:
0.6104**

Случайный лес

RandomForestClassifier

Модель работает очень медленно, на обучение затрачивается 10-15 минут.

**ROC-AUC:
0.6733**

Многослойный пресептрон

MLPClassifier

Поддерживает частичное обучение, что позволяет ускорить процесс обучения. Потенциально можно улучшить показатель метрики.

**ROC-AUC:
0.6864**

Стохастический градиентный спуск с log_loss

SGDClassifier(loss='log_loss')

Поддерживает частичное обучение, показала лучшие результаты по скорости. Данный алгоритм позволит вычленить фитчи влияющие на CR=1

Выбранный покемон:

Стохастический градиентный спуск с
логистической регрессией

SGDClassifier

- Алгоритм обучается значительно **быстрее и не нагружает железо**
- **Метрика удовлетворяет** задачу
- Есть возможность **замерить влияние фитч на классы** целевой переменной по отдельности
- Скорость предсказаний **быстрее других классификаторов.**

Тюнинг гипер-параметров



**Был использован
RandomizedSearchCV**

Этот способ больше подходит при
маленьких вычислительных мощностях
и большом количестве параметров

Изменения Feature Engineering

После развертывания сервиса в пайплайнах,
была проведена оптимизация и замеры влияния
сгенерированных фитч.



**ROC-AUC:
0.6864**

- is_organic_visit
- device_screen_width и device_screen_height
- is_socialmedia_advert
- ~~device_display_megapixel~~
- ~~device_orientation_vertical~~
- from_russia (и удаление geo_country)
- from_moscow (удалив geo_city)
- 'Population', 'Timezone', 'km_to_moscow' по имени города из внешних источников
- Так же был **удален** пайп с изменением редких значений

Доступ к модели по REST API

@GET
(/get_test_json)

**Получает: ?cr=0/1
(по умолчанию 1)
Возвращает:
случайную сессию с
учетом CR в виде
JSON**

JSON достается из файла
“data_to_test_api.pkl”
который создается при
обучение модели.

@POST
(/predict)

**Получает: json с
данными о сессии
Возвращает:
прогноз Conversion
Rate**

Пример отправляемого
JSON можно получить GET
методом get_test_json

@GET
(/all_feature_name)

**Возвращает:
Список всех
признаков которые
получает модель**

Список входных фич
может быть полезен чтобы
использовать точное
название фичи при
использование GET
метода /get_feature_imp

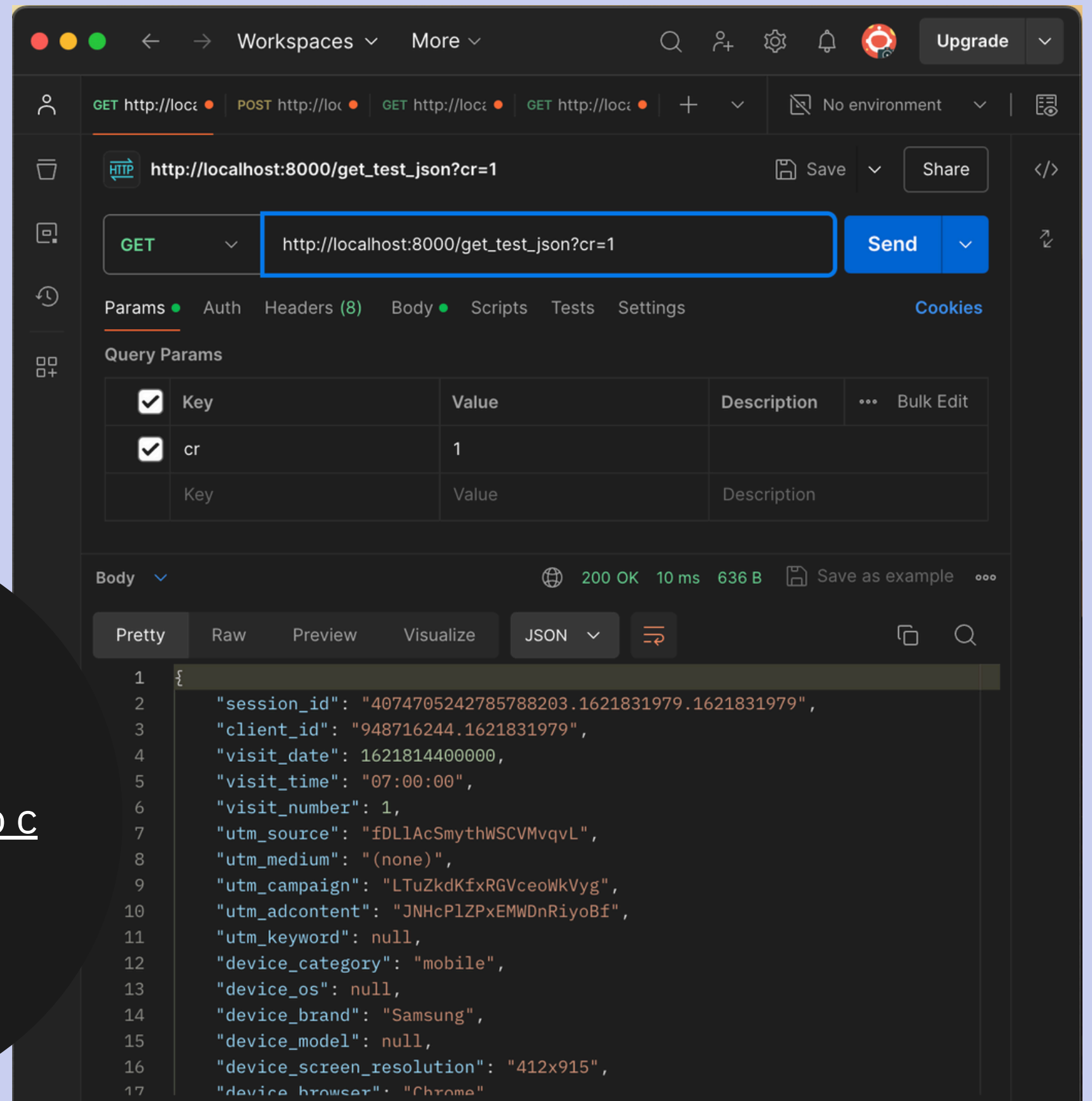
@GET
(/get_feature_imp)

**Возвращает список
топ20 фич с
коэффициентами
влияния на
положительный
класс CR**

Метод может принимать
наименование фич в виде
строки

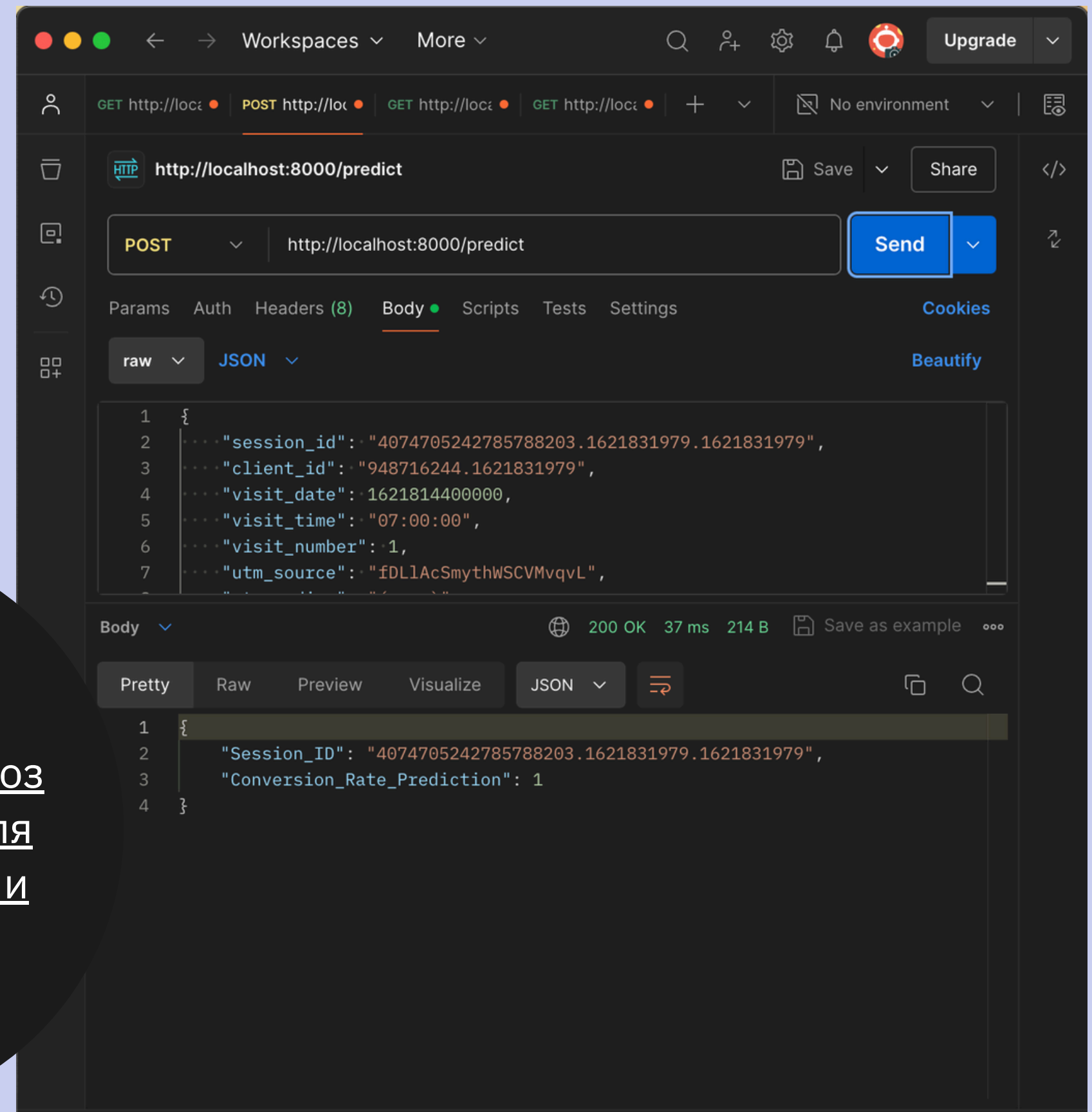
@GET (/get_test_json)

Возвращает
случайную сессию с
CR=1



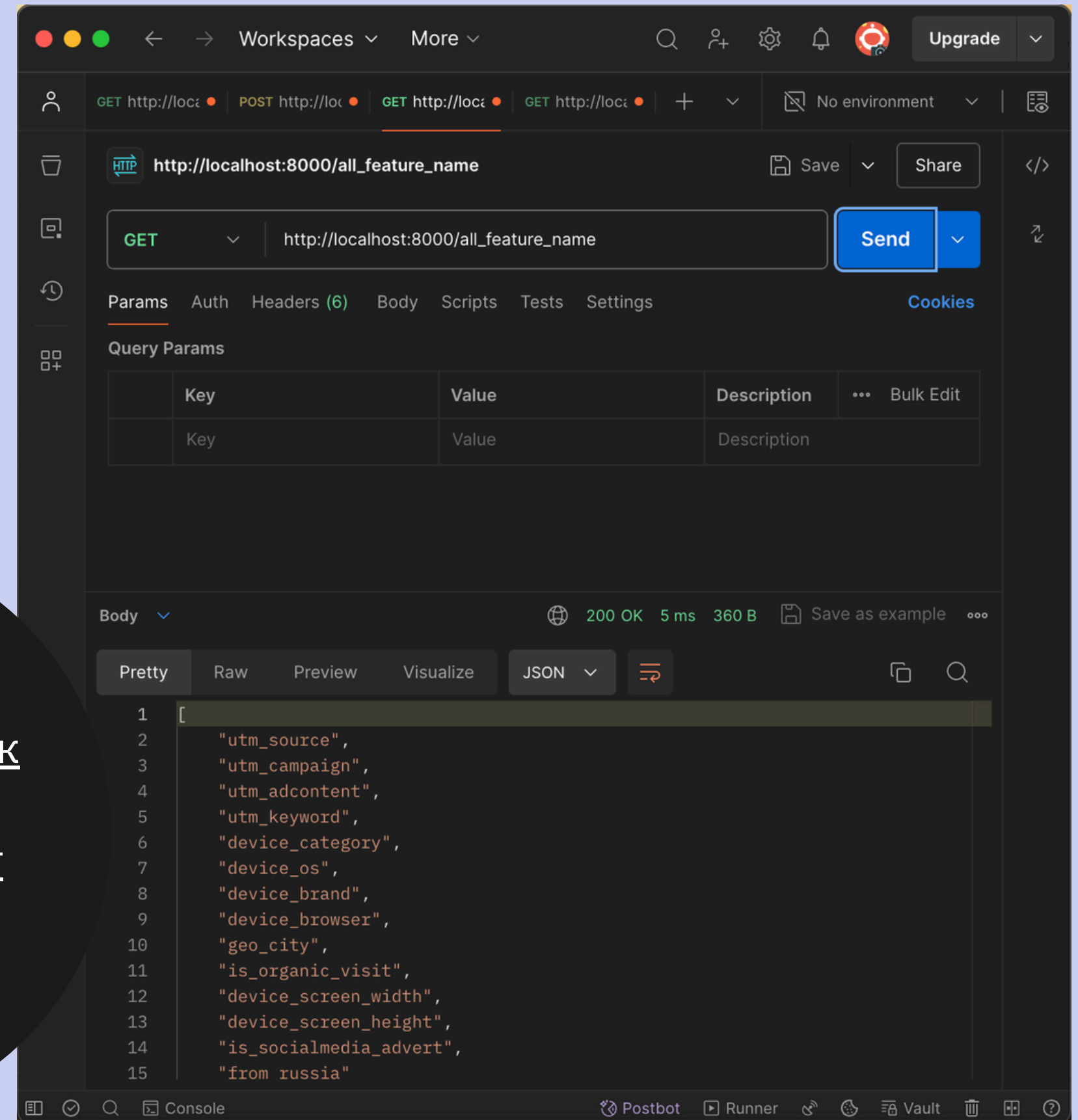
@POST (/predict)

Возвращает прогноз
Conversion Rate для
полученной сессии



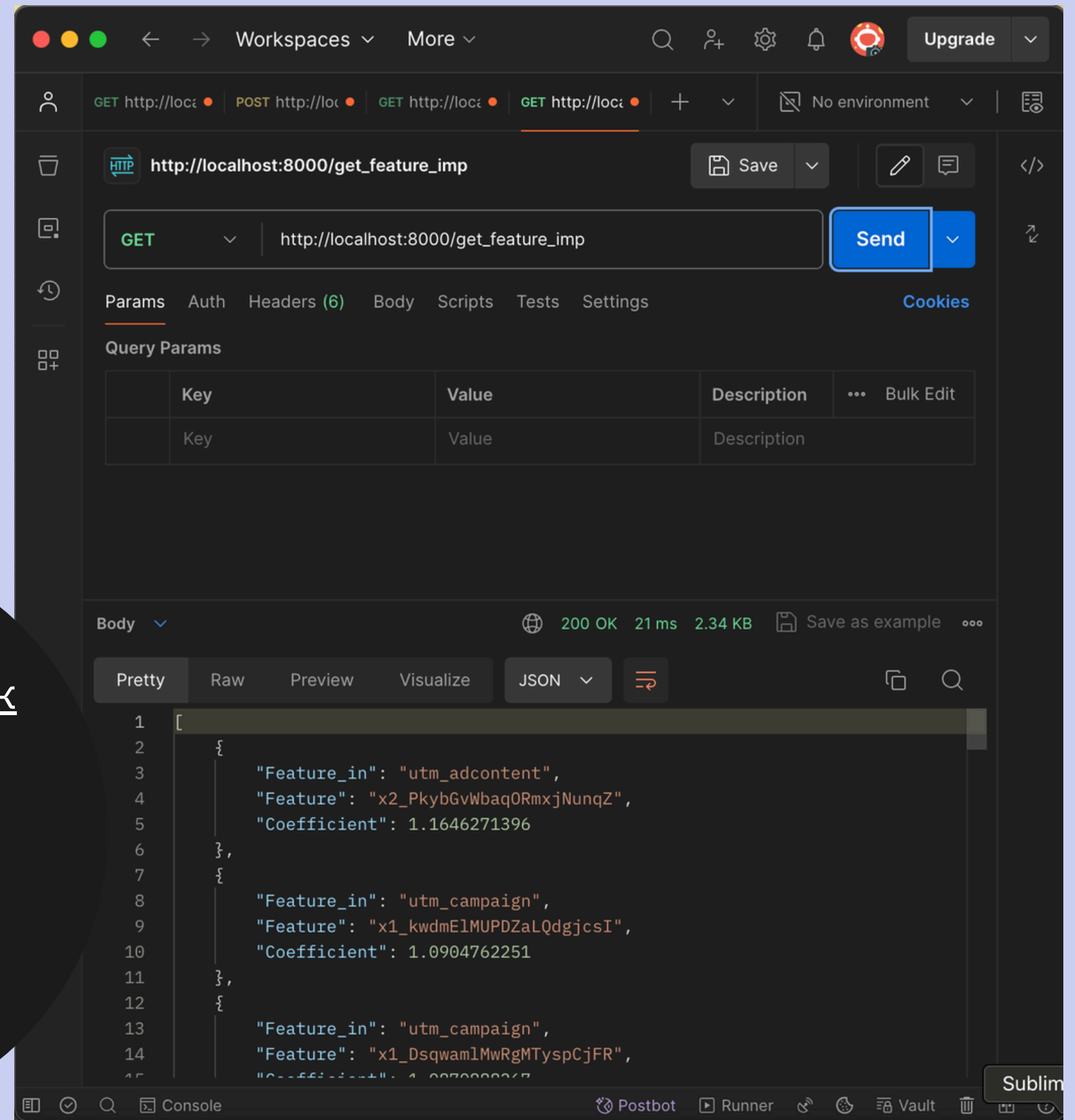
@GET (/all_feature_name)

Возвращает список
всех признаков
которые получает
модель

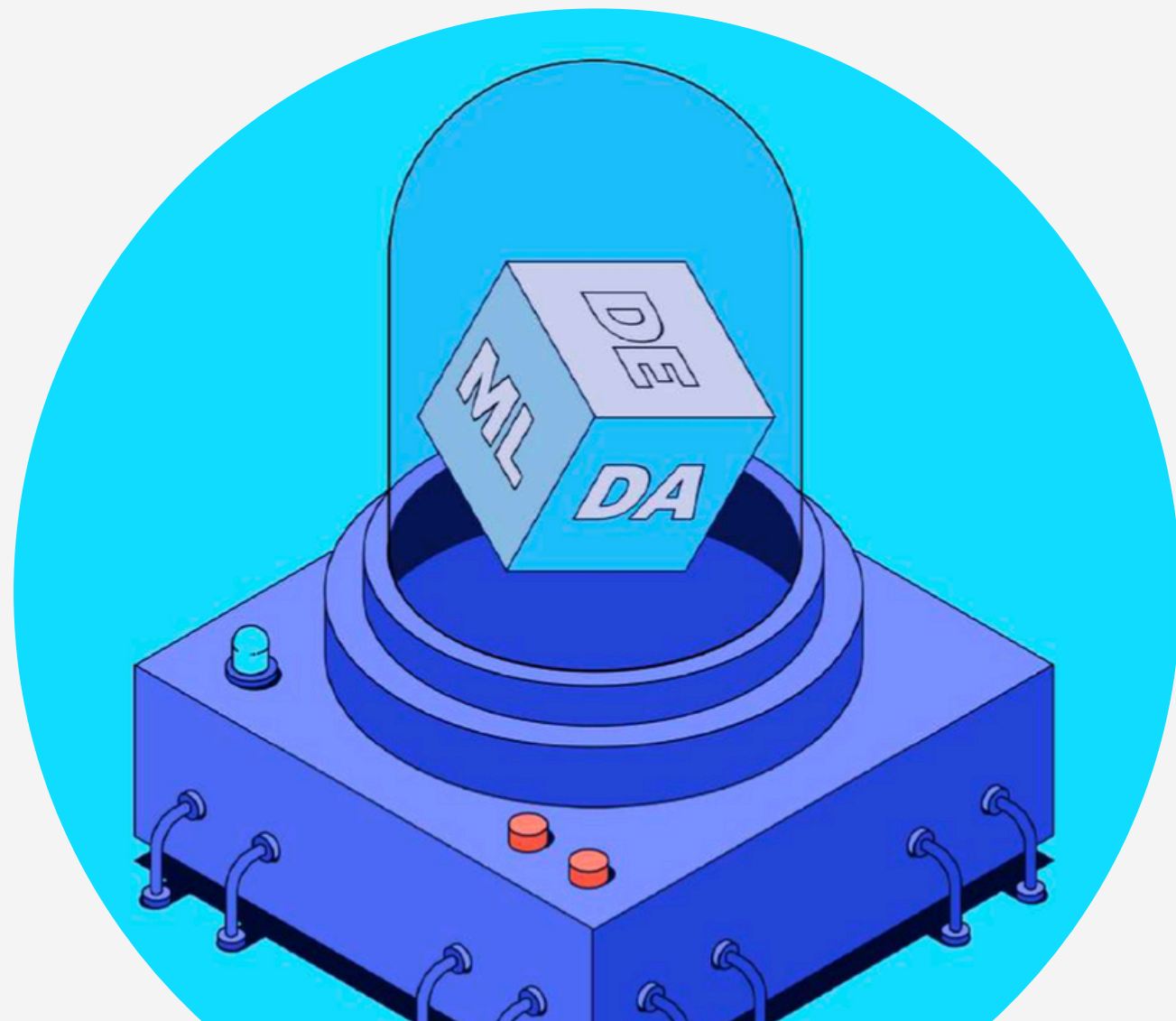


@GET (/get_feature_imp)

Возвращает список
топ20 фич с
коэффициентами
влияния на
положительный
класс CR



Спасибо!



Telegram accaunt

[@SaidPlatonov](#)

Электронный адрес

saidplatonov@gmail.com

Github

[https://github.com/saidplatonov/sber
_avto_sklrn_ML](https://github.com/saidplatonov/sber_avto_sklrn_ML)