Задание для лабораторной работы по курсу CLD

[**Данные:**](#_eazzwpl5o788) **1**

[**Задание**](#_mohvaxrzp0qs) **2**

[**Общие рекомендации**](#_fbjd9iva2yws) **2**

## Постановка:

Банк N изучает возможности оптимизации своей кредитной линии - доставьте необходимые данные в том разрезе, в котором аналитики различного профиля смогут наиболее полно их анализировать с использованием методов традиционной и “продвинутой” аналитики, а бизнес получать отчетность различного характера.

## Данные:

Используем публичный датасет BigQuery bigquery-public-data.ml\_datasets.credit\_card\_default - информация о просроченных платежах по кредитам

(\*) Бинарная классификация

- это задача классификации множества объектов в две группы, т.е. предсказание, к какой из двух групп принадлежит объект.

При этом элемент относится с некоторой вероятностью (например, 0.6) к первой группе, а с какой-то во второй (например, 0.4), причем это полная группа событий.

Порог определяет величину вероятности, “разделяющую” группы. Если группы со значениями 0 и 1 разделены порогом 0.5, то при вероятности < 0.5 объект относится к группе со значением 0, а при >= 0.5 - к группе со значением 1.

В изучаемом датасете поля default\_payment\_next\_month - признак просроченного платежа в действительности, а predicted\_default\_payment\_next\_month - признак, предсказанный при помощи некоторой модели бинарной классификации (обратите внимание, что это RECORD REPEATED с вероятностями для 0 и 1, образующими полную группу событий - суммарная вероятность = 1)

## Задание

1. Изучите структуру данных в датасете
2. Скопируйте датасет в свой проект, чтобы иметь возможность свободно им оперировать
3. Проанализируйте датасет - определите, сколько записей при пороге 0.5 алгоритм корректно классифицировал в ту или иную группу, чтобы понять, насколько эффективной была модель
4. Используя batch-шаблон Dataflow, выгрузите датасет в GCS в формате Parquet, чтобы предоставить коллегам аналитикам возможности для дальнейшего углубленного анализа
5. Разверните кластер Dataproc, откройте Jupyter
6. Интегрируйте выгруженные ранее данные в Jupyter посредством Spark, чтобы сделать их доступными для анализа
7. Выведите схему полученного DataFrame
8. Вычислите медиану кредитного лимита (поле limit\_balance) в зависимости от возраста (age)

Бонусное задание

1. В Jupyter конвертируйте данные в Pandas и визуализируйте зависимость между полями

На выходе будет пример pipeline поставки данных для использования в различных типах аналитики различными категориями пользователей (аналитики, data scientist’ы и т.д.) с примерами использования инструментария облака.

## Общие рекомендации

1. Не забудьте использовать сервисный аккаунт с широкими правами (в части Dataflow и BigQuery) при запуске выгрузки
2. Учитывайте локацию датасета при копировании, а само копирование можно выполнить посредством Create-Table-As-Select операции
3. Используйте примеры запуска из домашнего задания для корректной настройки параметров Dataflow
4. Не забудьте активировать Component Gateway и добавить Jupyter Notebook при развертывании Dataproc
5. Используйте коннектор gs:// при чтении данных из GCS посредством Spark ( spark.read.format("parquet").load("gs://bucket/path") )
6. Для вычисления медианы можно использовать функцию персентили percentile\_approx с параметром accuracy = 0.5 (<https://spark.apache.org/docs/latest/sql-ref-functions-builtin.html>), может быть удобно использовать механизм registerTempTable для использования стандартного SQL
7. В случае ошибок при запуске задачи Dataflow исследуйте логи облака через ссылку на Cloud Logging на предмет ошибки