Федеральное государственное автономное образовательное
учреждение высшего образования
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ «Программной инженерии и компьютерной техники (ФПИ и КТ)»

# ОТЧЕТ

по лабораторной работе №2

по курсу «Хранение и алгоритмы сжатия данных»

на тему: «Сравнение форматов хранения данных Parquet и ORC»

Студент P4135 (Группа) _____ (Подпись, дата) Постнов С. А. (Фамилия И. О.)

Преподаватель _____ (Подпись, дата) Бабаянц А. А. (Фамилия И. О.)

*2025 г.*

## СОДЕРЖАНИЕ

# 1 Теоретическая часть

## 1.1 Apache Parquet

`Apache Parquet` — колоночный формат хранения данных, оптимизированный для аналитических запросов. Основные особенности:

— Колоночное хранение данных для эффективного сжатия и быстрого чтения.

— Встроенная схема данных с типами.

— Поддержка вложенных структур данных.

— Оптимизация для больших данных и аналитических запросов.

— Использование алгоритмов сжатия Snappy, Gzip, LZ4.

## 1.2 Apache ORC

`Apache ORC` (*Optimized Row Columnar*) — высокопроизводительный колоночный формат хранения данных. Основные особенности:

— Гибридное хранение (строки и колонки).

— Встроенные индексы для быстрого поиска.

— Агрегированная статистика на уровне полос (stripes).

— Эффективное сжатие с использованием Zlib, Snappy, LZ4.

— Оптимизация для чтения больших объемов данных.

## 2 Практическая часть

В лабораторной работе использовались следующие датасеты для сравнения форматов:

1) `trade_data.csv` — данные торгов на бирже (∼266 МБ);

2) `market_orders.csv` — данные заказов на маркетплейсе (∼684 МБ);

3) `tweets.csv` — данные постов пользователей в социальной сети (∼3.9 ГБ).

Листинг 2.1 – Пример данных из файла `trade_data.csv`

```
timestamp,Open,High,Low,Close,volume
01.01.2015 00:00:00.000
   GMT+0530,119.819,119.829,119.815,119.817,24.39
```

Листинг 2.2 – Пример данных из файла `market_orders.csv`

```
ORDERID,BRANCH_ID,DATE_,USERID,NAMESURNAME,TOTALBASKET
7905270,320-DE1,2022-08-22 00:00:00,72946,Ali
   lhan,"2637,5499999999997"
```

Листинг 2.3 – Пример данных из файла `tweets.csv`

```
id;user;fullname;url;timestamp;replies;likes;retweets;text
1132977055300300800;KamdemAbdiel;Abdiel kamdem;;2019-05-27
   11:49:14+00;0;0;0; appena uscito un nuovo video! LES
   CRYPTOMONNAIES QUI PULVRISENT BITCOIN EN 2019
   https://t.co/yCsQMvRnyS
```

### 2.1 Выбранное преобразование данных

Для сравнения форматов были выполнены следующие операции:

1) Чтение исходных csv файлов.

2) Применение преобразований данных (добавление колонки с суммой длин всех полей).

3) Сохранение данных в форматах `Parquet` и `ORC`.

4) Измерение времени чтения сжатых файлов.

5) Вычисление коэффициентов сжатия.

Все измерения проводились с очисткой кэша `Spark` для обеспечения точности результатов.

## 2.2 Результаты сравнения

В таблице 2.1 представлены результаты сравнения форматов хранения данных.

Таблица 2.1 – Результаты сравнения форматов хранения данных

| Датасет | Размер csv, МБ | Размер `Parquet`, МБ | Размер `ORC`, МБ | Время чтения `Parquet` , с | Время чтения `ORC`, с |
|---|---|---|---|---|---|
| `trade_data` | 265.53 | 56.92 | 45.12 | 2.23 | 0.11 |
| `market_orders` | 684.14 | 200.70 | 182.68 | 1.88 | 0.05 |
| `tweets` | 3997.58 | 2182.62 | 1439.69 | 4.05 | 0.09 |

Коэффициенты сжатия для различных форматов представлены в таблице 2.2.

Таблица 2.2 – Коэффициенты сжатия

| Датасет | Сжатие `Parquet` | Сжатие `ORC` |
|---|---|---|
| `trade_data` | 4.7x | 5.9x |
| `market_orders` | 3.4x | 3.7x |
| `tweets` | 1.8x | 2.8x |

На рисунке 2.1 представлено сравнение размеров файлов, коэффициентов сжатия и скорости чтения для всех исследуемых датасетов.

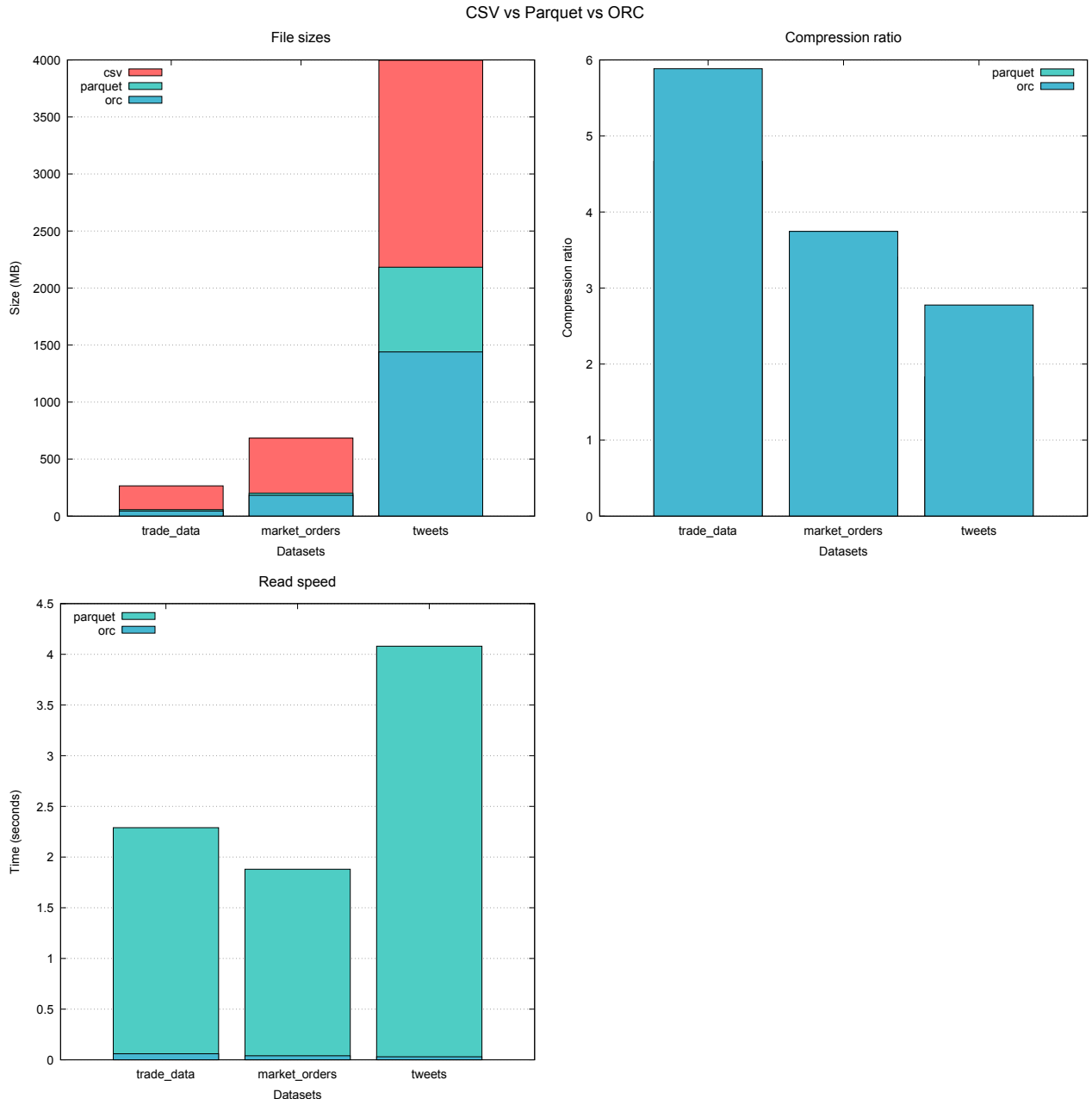


Рисунок 2.1 – Сравнение форматов хранения данных

# 3 Вывод

По результатам проведенного сравнения можно сделать следующий вывод:

1) `ORC` превосходит `Parquet` по сжатию — коэффициент сжатия ORC на $\sim$20 – 30% выше для большинства датасетов;

2) `ORC` быстрее читается — время чтения `ORC` файлов в $\sim$2 – 4 раза меньше по сравнению с `Parquet` для большинства случаев;

3) оба формата значительно превосходят csv — размер файлов уменьшается в $\sim$1.8 – 5.9 раз, а скорость чтения увеличивается в $\sim$5 – 50 раз.