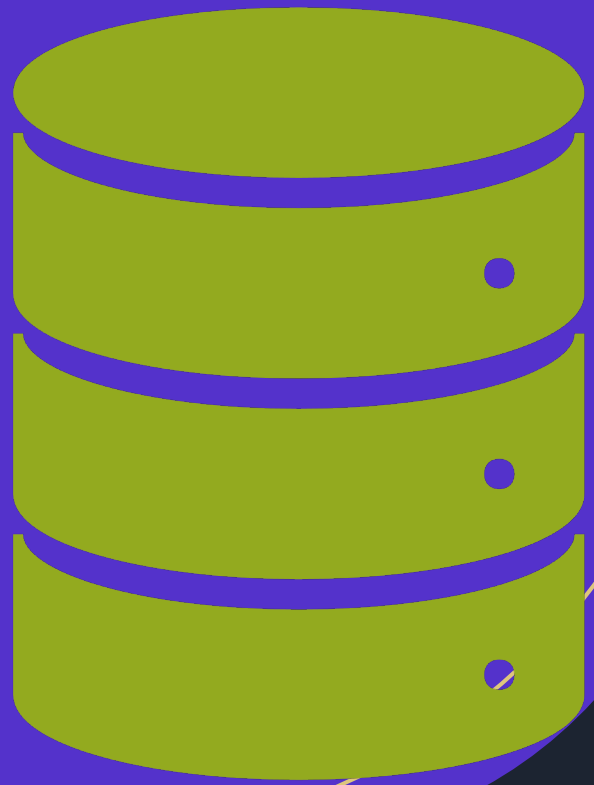
Data Management in Machine Learning

Proceedings of the 2017 ACM International Conference on Management of Data

Karolina Naranjo-Velasco

September 14, 2022



Background

- **Why do we use data management systems for ML?**

1

Integrate ML systems and languages with relational database management systems (RDBMSs)

2

Techniques: query optimization, portioning, and compression

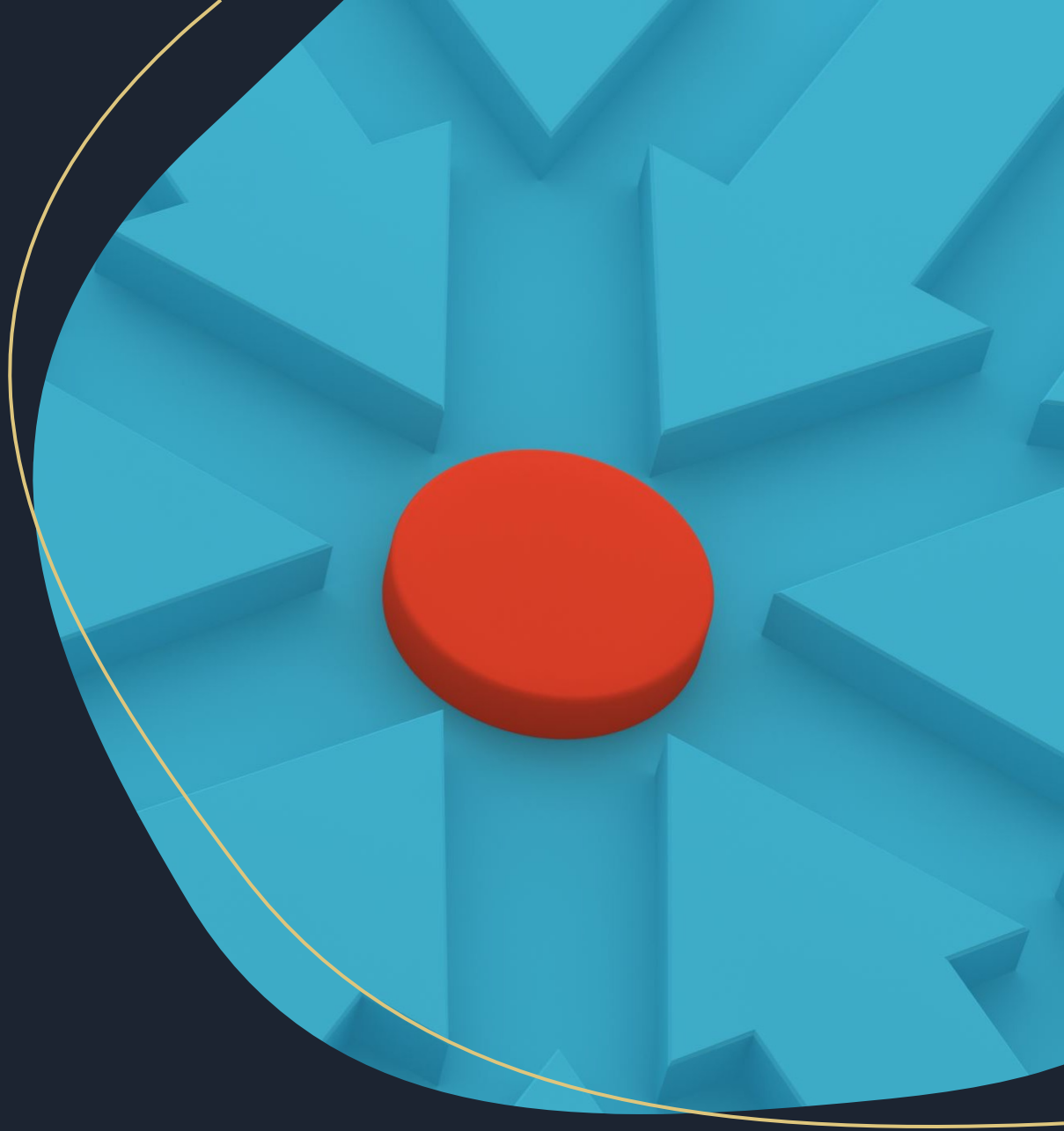
3

Combine DM and ML: ML lifecycle-related tasks

Problem Solved (1/2)

DATA-MANAGEMENT ORIENTED CHALLENGES IN ML

REVIEW: SYSTEMS AND TECHNIQUES IN THE CONTEXT OF ML WORKLOADS



Problem Solved(2/2)

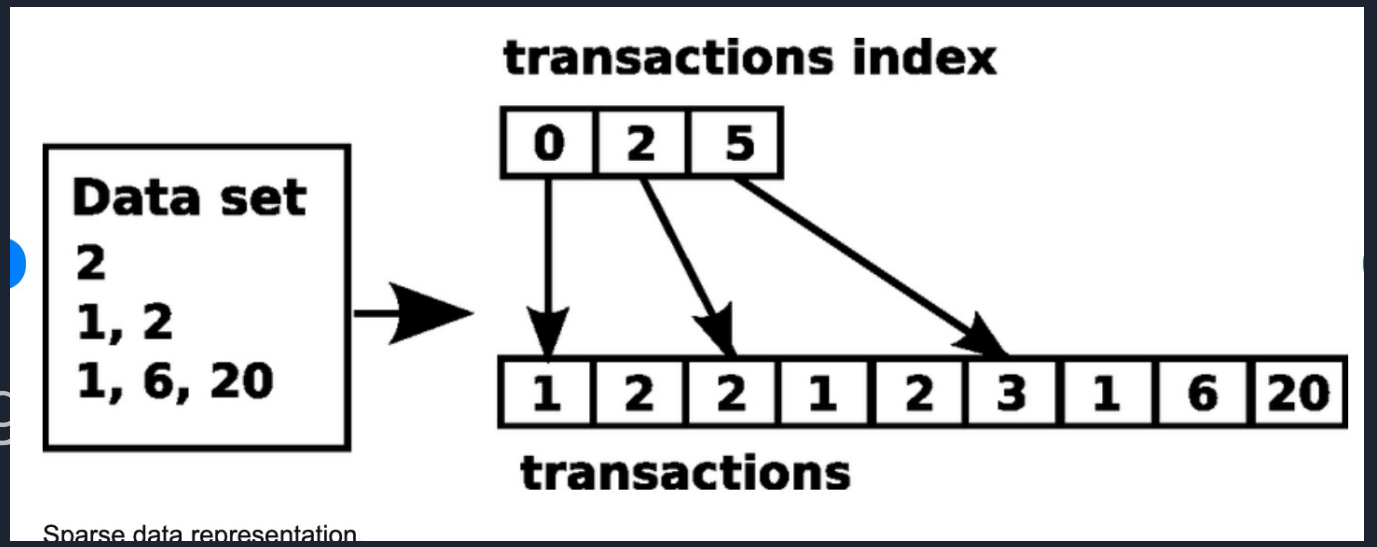
- Technical content:
  - Workload characterization
  - Data systems
  - DB-inspired techniques
  - ML lifecycle tasks
  - Open problems

Solution(1/2)

- Integrate ML algorithms with RDBMS

1) User define function (UDF) and user-defined aggregate (UDA)
   - ML + regular SQL for data processing
2) Joins queries :
   - Factorized learning: linear models/ joins
3) Statistical Relational Learning (SRL) :
   - DeepDive: join processing + large datasets + RDBMs
4) Simplify: Query generators
   - DF, matrices → queries
5) RDBMS Integrations
   - Declarative forecasting queries: creation , maintenance, and usage

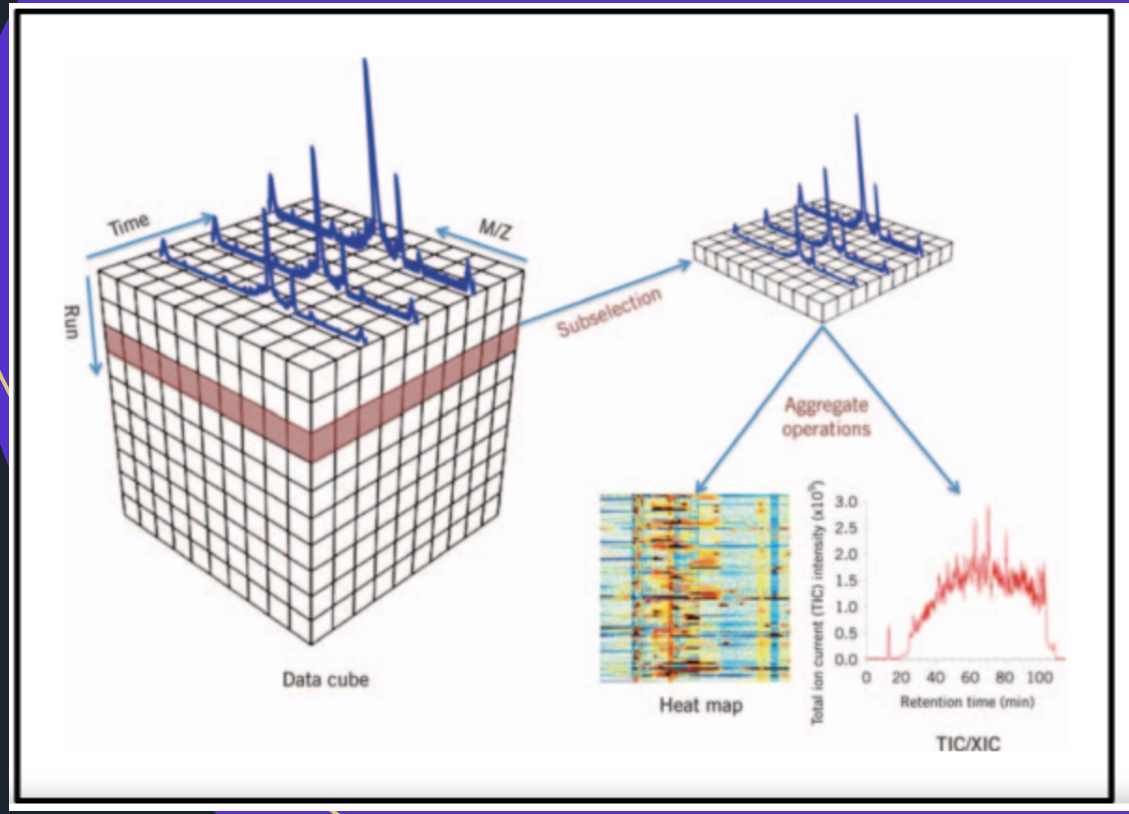
Solution (2/2)

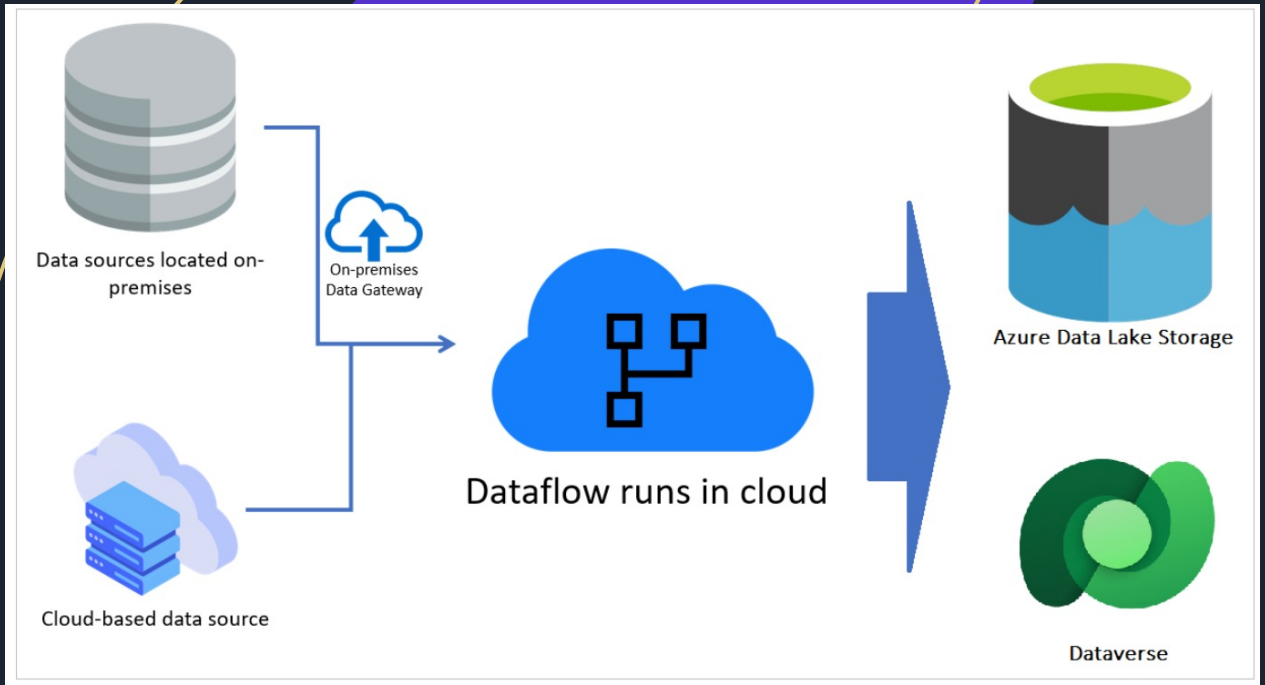


- Techniques: databases + prog
  performance computing
  - Focus on linear algebra, sparse and dense data representations
- Rewrite and Operation Selection:
  - Matrix multiplication chain
- Incremental maintenance:
  - Update LA programs → reduce iterations of fixpoints
- Operator fusion and code generation:
  - Reduce → intermediates and input scans

Solution (2/2)

- Asynchronous execution:
  - Avoid global barriers
- Compression and partitioning
  - Decompresses arrays block-wise
- Cloud ML resource elasticity
  - Cost-effective resource allocation





Key Insights

- ML lifecycle systems:
  - **Feature engineering**
    - Feature extraction → dataflow-oriented solutions
  - **Model selection and management**
    - Cloud services → construction, scaling, management end-to-end workflows

Enabling Ideas

- Size and sparsity estimation: complex function call patterns, data-dependent operations
- Convergence-based termination conditions: runtime, resource allocation, costly data re-organizations
- Value handlings: NaN
- New architectures for feature engineering and models selection : include meta-algorithms

Questions?

Thank you for listening to me!