FINM 32500 Computing for Finance in Python

Project Part 3

Feature Descriptions and Value to Quant Researchers Report

Name
Andy Andikko
Harrison Zhang
Matheus Raka Pradnyatama

Our key features address quant researchers' needs, from secure data handling to efficient preparation and seamless integration into research workflows, all while enhancing usability and reliability. All our codes are tested rigorously through unittests.

1. Detailed Metadata Tagging and Management

- **Feature**: Metadata collection tailored to data structure:
  - **Structured Data (Pandas DataFrame)**: Includes column names, row count, and index names.
  - **Semi-Structured Data (dict, list)**: Tags data as "JSON-like" and records item count (keys for dict, elements for list).
  - **Unstructured Data (str, bytes)**: Distinguishes textual and binary formats.
- **Value**: Provides quant researchers with a clear, descriptive summary of datasets, simplifying data handling and programmatic analysis.

2. Data Security and Role-Based Access in the Data Lake

- **Feature**:
  - **Access Key System**: Assigns unique access keys (e.g., secured_access = {134: "Andy", 245: "Matt", 367: "Harry"}) to control edits.
  - **Validation Mechanism**: Denies access for unregistered or invalid keys with clear error messaging.
- **Value**: Ensures data integrity by limiting access and protecting against accidental or intentional modifications that could disrupt workflows.

3. Data Transformation Capabilities in the Data Workbench

- **Feature**:
  - **Filtering**: Enables column-based conditional filtering for targeted analysis.
  - **Aggregation**: Facilitates summary statistics and grouped data preparation using .agg().
- **Value**: Automates repetitive tasks, allowing quant researchers to focus on generating insights from clean, analysis-ready datasets. Supports **chaining of transformations**.

4. Advanced Search in the Data Catalog

- **Feature**: Search across datasets using keywords with the search_datasets method, supported by robust categorization.
- **Value**: Saves researcher's time by streamlining the discovery of relevant data for specific analyses or strategies.

5. Integration with Python APIs for Seamless Data Access

- **Feature**: load_from_yfinance provides direct access to financial market data using the yfinance API, with flexible parameters (e.g., ticker, interval, period) and built-in error handling.
- **Value**: Simplifies data acquisition for quant researchers, enabling efficient retrieval of tailored datasets for intraday trading or long-term analysis.