finCleanPy: Module Documentation

core.py

Handles data profiling. It generates summary statistics, missing value reports, and infers data types. This is often the first step in the pipeline.

validators.py

Validates input and output DataFrames using schema definitions via the 'pandera' library. Ensures column names and types match expectations.

imputers.py

Fills missing values using simple and AI-based methods. XGBoost is used for contextual imputation, and KNN is available for more general numeric datasets.

anomaly.py

Detects and removes anomalies using Isolation Forest and AutoEncoder-based approaches. It's integrated after imputation for data integrity.

feature_engineering.py

Optionally adds derived features like rolling means or financial indicators. It is applied after outlier removal and normalization.

utils.py

Provides helper functions such as type detection to support other modules.

pipeline.py

Central manager to run custom pipelines. It allows sequential execution of profiling, imputing, anomaly removal, normalization, and feature engineering.

Module Interactions

1. User initializes CleanPipeline.
2. 'profile' step calls core.profile_data.
3. 'smart_impute' invokes imputers.smart_impute to fill missing values.
4. 'remove_outliers' cleans data using anomaly.remove_outliers.
5. 'normalize' standardizes the values.
6. Output is a clean DataFrame ready for analysis or export.

Sample Usage

```
from fincleanpy import CleanPipeline
```

```
pipeline = CleanPipeline()
pipeline.add_step('profile')
pipeline.add_step('smart_impute', method='knn')
pipeline.add_step('remove_outliers', method='autoencoder')
pipeline.add_step('normalize')
clean_df = pipeline.run(raw_df)
```