Week 2 Assignment: Data Processing

Financial Data Science

September 20, 2025

1 Assignment Overview

This assignment focuses on data preprocessing for cryptocurrency datasets, missing data handling, and basic financial analysis.
Due Date: 2025/09/25
Total Points: 100 points

2 Problem 1: Add Weekend Data (15 points)

Task: Add synthetic (imputed) weekend prices to your crypto + stock dataset, regarding weekend prices for stocks in order to align them with cryptos.
What to do:

- Take your weekday-only stock data
- Fill in weekend prices using an imputation technique
- Compare before/after statistics (mean, std)

Deliverable: Code + short explanation (200 words)

3 Problem 2: Rolling Window for Missing Data (15 points)

Task: Find the best rolling window size for mean/median imputation.
What to do:

- Test window sizes: 3, 7, 14, 20, 30, 60 days
- Calculate MAE and RMSE for each
- Choose the best window size

Math: Rolling mean $= \frac{1}{w}\sum_{i=t-w+1}^{t} x_i$ where w is the window size.
Deliverable: Plot showing performance vs window size + recommendation

4 Problem 3: Compare Imputation Metrics (15 points)

Task: Compare different ways to measure imputation quality.
Metrics to use:

- MAE: $\frac{1}{n}\sum |actual - predicted|$
- RMSE: $\sqrt{\frac{1}{n}\sum (actual - predicted)^2}$

- MAPE: $\frac{1}{n}\sum \frac{|actual-predicted|}{|actual|} \times 100\%$

What to do:

- Test 2-3 imputation methods on your data
- Calculate all three metrics
- Explain which metric is most useful and why

Deliverable: Results table + discussion (300 words)

5 Problem 4: Try Different Interpolation Methods (20 points)

Task: Implement and test custom interpolation methods.
Methods to implement:

- Linear interpolation: $y = y_1 + \frac{x-x_1}{x_2-x_1}(y_2 - y_1)$
- Polynomial interpolation (degree 2 or 3)
- Simple exponential smoothing: $\hat{y}_t = \alpha y_t + (1-\alpha)\hat{y}_{t-1} \quad ; \alpha \in (0,1)$

What to do:

- Code each method
- Test on crypto price gaps
- Compare performance using MAE/RMSE
- Briefly explain how each method works

Deliverable: Working code + performance comparison + method explanations

6 Problem 5: Handle Weird Results (10 points)

Task: Fix common calculation problems.
Issues to address:

- Infinite MAPE (when actual values = 0)
- NaN values from calculations
- Very large correlation values

What to do:

- Find these problems in your analysis
- Explain why they happen
- Fix them (e.g., use symmetric MAPE instead of MAPE)

Deliverable: List of problems found + solutions implemented

7 Problem 6: Advanced Imputation Methods (10 points)

Task: Try KNN or MICE imputation.
What to do:

- Use scikit-learn's KNNImputer or similar
- Compare against simple mean/median imputation
- Test different parameter values (e.g., number of neighbors)

KNN formula: $\hat{x}_i = \frac{1}{k} \sum_{j \in neighbors} x_j$
Deliverable: Implementation + performance comparison

8 Problem 7: Utility Functions for Investors (10 points)

Task: Discuss if utility functions make sense for crypto investing.
Simple utility functions:

- Risk-averse: $U(W) = \sqrt{W}$
- Risk-neutral: $U(W) = W$
- Risk-seeking: $U(W) = W^2$

What to discuss:

- How might different investor types (young vs old, rich vs poor) have different utility functions?
- Would this affect how we handle missing data?
- Give one concrete example

Deliverable: Discussion (400 words) with examples

9 Problem 8: Build Final Dataset (5 points)

Task: Combine crypto with traditional assets.
Include:

- cryptocurrencies (BTC, ETH, etc.)
- stocks (Apple, Google, etc.)

Requirements:

- Daily data for 5+ years
- Same date range for all assets
- Handle missing values using your best method from above

Deliverable: Clean dataset + documentation

10 Problem 9: Convert to Returns (5 points)

Task: Turn prices into returns.

Formulas:

- Simple returns: $R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$
- Log returns: $r_t = \ln(P_t) - \ln(P_{t-1})$

What to do:

- Calculate both types of returns
- Compare their distributions (mean, std, skewness)
- Create correlation matrix between assets

Deliverable: Returns dataset + basic statistics summary

11 Submission Requirements

What to submit:

- Jupyter notebook with all code
- Final datasets (CSV files)
- Short report (2-3 pages) summarizing key findings
- Presentation

Code requirements:

- Use Python with pandas, numpy, matplotlib
- Add comments explaining what each section does
- Make sure code runs without errors

12 Grading

Component	**Points**
Problems 1-9	90
Report/Presentation clarity	10
Total	**100**

Tips:

- Keep explanations simple and practical