

# PROJECT REPORT

## Indian Market Pairs Trading – Statistical Arbitrage Framework

---

### 1. Project Title

**Market-Neutral Pairs Trading Strategy Using Cointegration and Z-Score Analysis**

---

### 2. Introduction

Pairs Trading is a **statistical arbitrage strategy** that exploits temporary mispricing between two historically related financial instruments.

Instead of predicting market direction, the strategy focuses on **mean reversion** of price relationships, making it largely **market-neutral**.

This project implements a **complete end-to-end Pairs Trading framework** for Indian equities, combining:

- Statistical theory (cointegration)
- Quantitative signal generation (spread & Z-score)
- Backtesting and performance analysis
- Interactive visualization using Streamlit

The system is designed for **data scientists, quantitative analysts, and students** interested in algorithmic trading and applied statistics.

---

### 3. Objectives

- Identify statistically valid stock pairs using cointegration
  - Construct a hedge-adjusted spread between asset pairs
  - Normalize the spread using Z-score
  - Generate long/short trading signals based on mean reversion
  - Backtest the strategy and evaluate performance
  - Build an interactive Streamlit dashboard for visualization
- 

### 4. Technology Stack

Category	Tools
Programming Language	Python
Data Analysis	Pandas, NumPy
Statistics	Statsmodels
Visualization	Matplotlib, Seaborn
Machine Learning	Scikit-learn
Web Framework	Streamlit
Development Environment	Jupyter Notebook
Deployment Ready	Yes (Streamlit)

---

## 5. System Architecture

**Data → Statistical Testing → Spread Construction → Z-Score → Signals → Backtesting → Visualization**

1. Historical price data ingestion
  2. Correlation and cointegration testing
  3. Hedge ratio estimation using linear regression
  4. Spread calculation
  5. Z-score normalization
  6. Signal generation (entry/exit)
  7. Performance evaluation and visualization
- 

## 6. Core Mathematical Formulation

### 6.1 Spread Construction

The spread represents the relative mispricing between two assets:

$$\text{Spread} = \text{Price}_A - (\beta \times \text{Price}_B)$$

Where:

- $\text{Price}_A, \text{Price}_B$  → Asset prices
  - $\beta$  (beta) → Hedge ratio from regression
- 

### 6.2 Z-Score of Spread

To standardize deviations and detect extremes:

$$Z_t = \frac{\text{Spread}_t - \text{Mean}(\text{Spread})}{\text{StdDev}(\text{Spread})}$$


---

## 7. Trading Strategy Logic

Condition	Action
Z-score > +Threshold	Short the spread
Z-score < -Threshold	Long the spread
Z-score ≈ 0	Exit position

This ensures **market neutrality** by simultaneously taking long and short positions.

---

## 8. Key Features

- Cointegration-based pair selection
  - Hedge ratio estimation
  - Mean-reversion trading logic
  - Z-score based entry and exit signals
  - Backtesting with trade-level PnL
  - Equity curve and performance visualization
  - Interactive Streamlit dashboard
- 

## 9. Implementation Highlights

- Cointegration testing using Engle-Granger method
  - Regression-based dynamic hedge ratio
  - Robust spread stationarity validation
  - Rolling statistics for Z-score calculation
  - Modular and reusable Python code
  - Streamlit app for interactive analysis
- 

## 10. Sample Use Cases

- Quantitative trading research
  - Statistical arbitrage strategy development
  - Learning cointegration and mean reversion
  - Portfolio-neutral trading experiments
  - Data science + finance project portfolio
- 

## 11. Future Enhancements

- Dynamic hedge ratio using Kalman Filter
  - Transaction cost and slippage modeling
  - Multi-pair portfolio optimization
  - Live market data integration
  - Automated trade execution
  - Risk-adjusted metrics (Sharpe, Sortino, Calmar)
- 

## 12. Conclusion

This project demonstrates how **statistical concepts can be transformed into a production-ready quantitative trading system.**

By combining **theory, data science workflows, and interactive applications**, it bridges the gap between academic finance and real-world trading systems.

---

## 13. Author

### Ranadip Manna

Data Scientist | Quantitative Finance & Machine Learning

✉️ LinkedIn: <https://www.linkedin.com/in/ranadip-manna-35534a1ba/>

### Yuvraj Sharma

Data Scientist | Quantitative Finance & Machine Learning

✉️ LinkedIn: <https://www.linkedin.com/in/yuvraj-sharma-87690311a>