Report on Trade Ranking Methodologies and Results

Author: Krit Prasad

---

1. Introduction

The purpose of this study is to develop a **robust trade ranking system** that evaluates trader performance based on multiple financial metrics. Three different ranking methodologies were used:

1. **Weighted Score Method** – Assigning predefined weights to key metrics.
2. **Z-Score Normalization** – Standardizing metrics for fair comparison.
3. **Machine Learning (Random Forest Regressor)** – Predicting trade performance based on historical data.

Additionally, **Buy and Sell positions** were extracted from the trade data, and **Net Positions** were computed to factor into rankings.

---

2. Data Processing & Preprocessing

2.1 Data Loading & Initial Inspection

- The dataset contained key trading metrics, such as **PnL (Profit** and **loss), ROI, Sharpe Ratio, Win Rate, Drawdown, Total Positions, and Net Positions**.
- Some columns included **timestamp data** (e.g., First_Trade, Last_Trade), which were converted into numeric values (e.g., **Trading Duration**).
- **Missing values were identified** in certain features and handled appropriately.

2.2 Trade History Parsing

- The Trade_History column was converted from a nested JSON-like structure into a structured **tabular format**.
- Key extracted features included **trade timestamps, trade sides (BUY/SELL), price, quantity, and realized profit**.
- **Buy and Sell trades were counted separately** per Port_ID, allowing us to calculate **Total_Buy_Positions** and **Total_Sell_Positions**.
- **Net Positions** were computed as:

  NetPositions=TotalBuyPositions−TotalSellPositionsNet_Positions = Total_Buy_Positions - Total_Sell_Positions

- Each trade was linked to a unique **Port_ID** (portfolio ID) for aggregation.

---

3. Outlier Detection & Handling

3.1 Identifying Outliers

Outliers were detected in key financial metrics, particularly in:

- **ROI (Return on Investment)**: Extreme values (inf and NaN) due to division by zero in **PnL** calculations.
- **Price & Quantity**: Large variance in trading behaviour.
- **Realised Profit & Sharpe Ratio**: Large deviations impacting ranking stability.
- **Net Positions**: Extreme values caused by one-sided trading activity.

3.2 Outlier Handling Approaches

1. **Winsorization (1st & 99th percentile capping)**
   - Prevents extreme values from distorting rankings.
2. **Rolling Z-Score Filtering**
   - Dynamically adjusts extreme values without removing them.
3. **Replacing inf values in ROI**
   - Used **small epsilon values** (1e-6) to avoid division by zero errors.

4. **Capping extreme Net Positions**
    - Applied a percentile-based threshold to limit highly unbalanced portfolios.

---

4. Ranking Methodologies

4.1 Weighted Score Method (WS)

A manually defined **weighted scoring system** based on key metrics:

- **PnL (40%)** – Measures total profitability.
- **ROI (25%)** – Measures efficiency of returns.
- **Sharpe Ratio (15%)** – Measures risk-adjusted performance.
- **Win Rate Bonus (10%)** – Rewards consistent profitability.
- **Net Positions Impact (10%)** – Penalizes highly unbalanced portfolios.

Formula:

Trade_Score=(0.4×PnL_Norm)+(0.25×ROI_Norm)+(0.15×Sharpe_Norm)+(0.1×WinRate_Norm)+(0.1×NetPositions_Norm)Trade_Score = (0.4 \times PnL_Norm) + (0.25 \times ROI_Norm) + (0.15 \times Sharpe_Norm) + (0.1 \times WinRate_Norm) + (0.1 \times NetPositions_Norm)

Results:

- **Traders with strong profitability, risk management, and balanced positions ranked higher.**
- **Highly unbalanced portfolios (extreme Net Positions) were penalised.**

4.2 Z-Score Normalization (ZS)

Z-score transformation standardises metrics to ensure fair comparison:

$Z=(X-\mu)/\sigma$

Where:

- **X** = Original value
- **μ** = Mean of the feature

- **σ** = Standard deviation

Results:

- **Z-Score captured traders with high deviations from the mean.**
- Helped **normalise ROI**, reducing the impact of outliers.
- **Highly unbalanced traders were adjusted based on the Net Positions Z-score.**

4.3 Machine Learning (ML) - Random Forest Regressor

A supervised learning model was trained using a **Random Forest Regressor** to predict trade rankings.

Features Used (X):

- **Financial Metrics**: PnL, ROI, Sharpe_Ratio, Win Rate (%), Drawdown, Net Positions, etc.
- **Normalized Values**: PnL_Norm, ROI_Norm, Sharpe_Norm, etc.
- **Z-Score Transformed Values**: PnL_Z, ROI_Z, Sharpe_Z, etc.

Target Variable (y):

- Trade_Weight_Score (Weighted Score from Section 4.1)

Results:

- **ML closely matched Weighted Score rankings**, proving it learned similar patterns.
- **Hidden relationships** between Net Positions and profitability were detected.

---

5. Results Comparison

Rank	S_Ranking (Weighted Sc	S_Ranking (Z-Scor	_Ranking (RandomFor
	5814617275053313	)204877254599680	)204877254599680
	)204877254599680	)382575336130560	5814617275053313
	9240873283311617	5087012661391104	9240873283311617

---

6. Conclusion & Recommendations

Best Approach:

- **Weighted Score is best for manual ranking with domain knowledge.**
- **ML is best for automated ranking systems** (e.g., AI-driven trade evaluation).
- **A hybrid approach (ML + Weighted Score) optimises results further.**
