Author: Jatin Dhurve

⭕ Beta Calculation for Each Factor

For each asset i and each factor j:

 # Market Beta
beta_market[i] = Cov(Return_asset[i], Mkt-RF) / Var(Mkt-RF)

 # Size Beta
beta_SMB[i] = Cov(Return_asset[i], SMB) / Var(SMB)

 # Value Beta
beta_HML[i] = Cov(Return_asset[i], HML) / Var(HML)

 # Momentum Beta

beta_Mom[i] = Cov(Return_asset[i], Mom) / Var(Mom)

Factor-Based Constraints/Bounds

Beta Range Methodology

Overview

This methodology calculates optimal beta coefficient ranges for factor-based portfolio construction using historical hedge fund performance data.

Data Requirements

- **Returns Data**: Monthly hedge fund returns (4 indices)
- **Beta Data**: Quarterly factor loadings (β_Mkt-RF, β_SMB, β_HML, β_Mom)

Mathematical Formulation

Step 1: Beta Range Calculation

For each factor β, calculate four risk tolerance ranges:

Conservative Range (Interquartile)

Range = $[Q_1, Q_3] = [P_{25}, P_{75}]$

Moderate Range

Range = [P_{10}, P_{90}]

Aggressive Range

- Range = [μ - 1.5σ, μ + 1.5σ]

where μ = mean(β), σ = std(β)

Historical Range

Range = [min(β) - 0.05, max(β) + 0.05]

Step 2: Optimization Criterion

For each range, calculate Sharpe ratio during periods when β falls within range:

Sharpe = (μ_r - μf) / σ_r

where:

- μ_r = mean return during qualifying periods
- μf = risk-free rate
- σ_r = return volatility during qualifying periods

Step 3: Optimal Range Selection

Select range with highest Sharpe ratio for each factor:

Optimal Range = argmax(Sharpe(Range_type))

Results

The methodology identifies conservative ranges as optimal across all factors, providing:

- Market Beta: [0.173, 0.401]
- Size Beta: [-0.232, 0.447]
- Value Beta: [-0.153, 0.374]
- Momentum Beta: [-0.272, 0.156]

Rolling Implementation

- Repeat for quarter using 7-year windows
- Q1 2012: Use 2005-2011 data
- Q2 2012: Use 2005-Mar 2012 data
- Continue for 46 quarters

⭕ Filtered Covariance

Step 1: Factor Model Estimation

$R(i,t) = \alpha(i) + \beta(i,1)\times MktRF(t) + \beta(i,2)\times SMB(t) + \beta(i,3)\times HML(t) + \beta(i,4)\times Mom(t) + \varepsilon(i,t)$

- Extract factor betas using regression
- Calculate residuals: $\varepsilon(i,t)$ = Actual - Predicted returns

Step 2: Robust Covariance (Tyler's M-Estimator)

$S(k+1) = (N/T) \times \Sigma[x(t) \times x(t)' / (x(t)' \times S(k)^{-1} \times x(t))]$

- Reduces impact of outliers on covariance matrix

Step 3: RMT Noise Filtering

$\lambda(max) = (1 + \sqrt{(T/N)})^2$

- If eigenvalue > $\lambda(max)$ → Signal (keep)
- If eigenvalue ≤ $\lambda(max)$ → Noise (filter)

Step 4: Matrix Reconstruction

Filtered Matrix = Q × New_Eigenvalues × Q'

- Keep signal eigenvalues, average out noise eigenvalues

Step 5: Rolling Implementation

- Repeat Steps 1-4 every quarter using 7-year windows
- Q1 2012: Use 2005-2011 data
- Q2 2012: Use 2005-Mar 2012 data
- Continue for 46 quarters

Result: Clean, noise-filtered covariance matrices for portfolio optimization with reduced estimation error and better out-of-sample performance.

⭕ Expected Returns Using Machine Learning

Step 1:

Data Setup Enhanced Feature Set

X = [VIX, Yield_Curve, Credit_Spreads, PMI, Market_Factor, Size_Factor]
Y = [4 hedge fund strategies: HFRI4FWC, HFRI4ELS, HFRI4ED, HFRI4EHV]

Step 2:

Rolling Window Training For Quarter Q(t):

Training Period = [t-7 years, t-1 day] Prediction Period = Quarter Q(t)

Step 3: Ensemble Model Expected Return = Huber Regression (40%) + AdaBoost (40%) + ElasticNet (20%)

- Huber: Handles market outliers and crashes robustly
- AdaBoost: Adaptive learning focusing on difficult prediction periods
- ElasticNet: Automatic feature selection from macro indicators

Step 4: Mathematical Relationship

HF_Return(i,t) = f(VIX(t), YieldCurve(t), CreditSpread(t), PMI(t), MktRF(t), RF(t)) + ε(t)

Step 5: Process Flow

1. Train ensemble model on 7-year historical data
2. Input current quarter's 6 macro indicators
3. Get weighted ensemble predictions
4. Generate quarterly expected returns

Step 6:

Model Performance Validation Results (R^2 scores):

HF3_HFRI4EHV (68.7%),
HF2_HFRI4ELS (62.9%),
HF1_HFRI4FWC (49.4%),
HF4_HFRI4EMN (37.4%)
Hit Rates: 77-85% directional accuracy across strategies

Step 7: Output Expected_Returns_Matrix = [46 quarters × 4 hedge fund strategies] Innovation: ML-predicted returns adapt to current economic conditions vs. static historical averages

---

⭕ Machine Learning Model Validation Process

Step 1: Rolling Window Setup

- 46 overlapping 7-year periods, moving quarterly
- 80% training, 20% testing for each period

Step 2: Enhanced Feature Set

X = [VIX, Yield_Curve, Credit_Spreads, PMI, Market_Factor, Size_Factor]
Y = [4 hedge fund strategies: HFRI4FWC, HFRI4ELS, HFRI4EMN, HFRI4EHV]

Step 3:

Ensemble Model Final_Prediction = Huber Regression (40%) + AdaBoost (40%) + ElasticNet (20%)

- Huber: Outlier-robust linear model for market crashes
- AdaBoost: Adaptive boosting focusing on prediction errors
- ElasticNet: Feature selection through L1/L2 regularization

Step 4:

Performance Metrics R^2 Score, RMSE, MAE, Hit Rate, Information Coefficient

Step 5:

Composite Scoring Score = $0.4\times R^2\_avg + 0.3\times Hit\_Rate\_avg + 0.2\times IC\_avg - 0.1\times Breakdown\_Rate$

Step 6:

Validation Results Best Ensemble Performance:

HF3_HFRI4EHV (68.7% R^2),
HF2_HFRI4ELS (62.9% R^2),
HF1_HFRI4FWC (49.4% R^2),
HF4_HFRI4EMN (37.4% R^2)
Hit Rates: 77-85% directional accuracy, Low breakdown rates: 0-4.3%

Step 7:

Decision Framework If $R^2 < -0.1$ OR Breakdown_Rate > 40%:
REJECT If $R^2 > 0.4$ AND Breakdown_Rate < 10%:
CORE HOLDING If $R^2 > 0.3$ AND Breakdown_Rate < 20%:
STRONG BUY If $R^2 > 0.2$ AND Breakdown_Rate < 20%: MODERATE BUY
Else: CAUTION/AVOID

Final Recommendation: All 4 strategies qualified for portfolio optimization with strong predictive performance.

Key Innovation: Validated ensemble approach achieving 37-69% R^2 scores across hedge fund strategies with robust temporal stability.

⭕ Portfolio Optimization Formula for Rolling 7-Year Periods

Objective Function (Maximize):

$$Utility = w'\mu - (\lambda/2) \times w'\Sigma w$$

Where:

- $w = [w_1, w_2, ..., w_7]$ = portfolio weights vector
- μ = ML-predicted expected returns vector (from XGBoost + LightGBM ensemble)
- Σ = RMT-filtered covariance matrix (from Tyler's M-estimator + noise filtering)
- λ = risk aversion parameter

Two Risk Aversion Scenarios:

Scenario 1: $\lambda = 0.05$ (Low Risk Aversion - Aggressive)

max $w'\mu - (0.05/2) \times w'\Sigma w$

max $w'\mu - (0.025) \times w'\Sigma w$

✓ Subject to constraints 1-6 below

Scenario 2: λ = 10 (High Risk Aversion - Conservative)

max $w'\mu - (10/2) \times w'\Sigma w$

max $w'\mu - (5.0) \times w'\Sigma w$

✓ Subject to constraints 1-6 below

Complete Optimization Problem:

Subject to:
1. $\Sigma w_i = 1$ (weights sum to 1)
2. $w_i \geq 0$ (long-only)
3. $0.6 \leq \Sigma w_i \beta_i^{mkt} \leq 1.1$ (market beta bounds)
4. $-0.2 \leq \Sigma w_i \beta_i^{smb} \leq 0.3$ (size beta bounds)
5. $-0.4 \leq \Sigma w_i \beta_i^{hml} \leq 0.2$ (value beta bounds)
6. $-0.1 \leq \Sigma w_i \beta_i^{mom} \leq 0.2$ (momentum beta bounds)

Example: Q1 2012 Optimization Using Jan 2005 - Dec 2011 data:

For λ = 0.05 (Aggressive):

- Higher weight on expected returns μ
- Lower penalty for risk $w'\Sigma w$
- Results in more concentrated, higher-risk portfolios

For λ = 10 (Conservative):

- Lower weight on expected returns μ
- Higher penalty for risk $w'\Sigma w$
- Results in more diversified, lower-risk portfolios

Rolling Process:

- Q1 2012: Use 2005-2011 → Get $w_1(\lambda=0.05)$ *and* $w_1(\lambda=10)$
- Q2 2012: Use 2005-Mar2012 → Get $w_2(\lambda=0.05)$ *and* $w_2(\lambda=10)$
- Continue for 46 quarters

Output: Two sets of optimal weights for each quarter - one aggressive portfolio and one conservative portfolio, both adapting to changing market conditions.