Building an Open-Source Barra-Like Specific Return Model

Abstract

This working paper documents the open-source **hornli-hsr** project, which builds a free alternative to MSCI Barra's proprietary specific-return model. The code ingests daily market data and quarterly fundamentals, constructs standardized descriptors (e.g., size, **beta**, value, momentum, leverage, volatility) and aggregates them into high-level style factors. Each trading day, a cross-sectional weighted-least-squares regression relates stock returns to their industry, country and style exposures; factor returns are the fitted coefficients and specific returns are the residuals. A covariance estimator based on exponential-weighted moving averages (EWMA) of factor and specific returns produces daily factor covariance and specific variance matrices. To avoid look-ahead bias, fundamentals and prices are shifted so only information available at each date is used. The project publishes descriptor-level exposures, factor returns and specific returns while respecting vendor licensing. HSR is an open research tool and is neither affiliated with nor endorsed by MSCI/Barra.

Keywords

Barra risk model; factor models; cross-sectional regression; style factors; open-source risk model; look-ahead bias; EWMA covariance; financial risk management.

Introduction

Factor-based risk models, such as MSCI's Barra US Equity models, decompose an asset's return into common factor exposures multiplied by factor returns plus a specific (idiosyncratic) return. Factors cover industries, countries/regions and style factors (size, **market beta**, value, momentum, etc.). Factor returns are estimated via cross-sectional regression of asset returns on their exposures; the resulting time series feed the factor covariance matrix. Because exposures are neutral in aggregate, the cap-weighted sum of industry returns approximates the market return. The **hornli-hsr** project follows this methodology but with open data and code. It releases descriptor-level exposures and the resulting factor and specific return panels while preserving proprietary raw inputs.

Data and Universe

Universe and dates

The model targets U.S. equities in the Russell 3000 universe. Daily market data include open, high, low, close and adjusted prices, and a daily S&P 500 index series is used to compute market beta. Fundamental data include quarterly

income-statement and balance-sheet items. The default backtest begins on **1 January 2015** and runs to the present.

Non-public data

Raw market and fundamental data (under `DEFAULT_PATH/input`) cannot be released. Instead, the project computes descriptors from these inputs and stores them under `DEFAULT_PATH/intermediate/descriptor`; these descriptor files and the processing code are released publicly. Final factor and specific return panels are written to `DEFAULT_PATH/output`.

Avoiding look-ahead bias

Look-ahead bias arises when a study uses information not available at the decision time. The code shifts both fundamentals and prices to ensure that descriptors and exposures reflect only past information. Quarterly fundamentals are aligned by their quarter-end dates, shifted by one reporting period and forward-filled, and daily market data are shifted by one day. This shift ensures that descriptors use only data that would have been available to investors at each date. When computing returns, the next day's price change is regressed on exposures constructed from yesterday's data.

Descriptor Construction

HSR's style factors are inspired by MSCI/Barra. Each factor is built from one or more descriptors computed from price, fundamentals or index data. Key descriptors include:

- **Market capitalisation & Size** – Market cap is price times diluted shares. The *size* descriptor is the natural logarithm of market cap, and a *non-linear size* descriptor uses the cube of this log.
- **Beta (market exposure)** – Daily simple returns for each stock and the S&P 500 index are computed, then a rolling exponential-weighted covariance between stock and market returns is divided by the rolling variance of market returns. With a half-life of about 63 days, this produces a time-varying market beta series per stock.
- **Value** – Book-to-price (total common equity divided by price), sales-to-price and cash-flow-to-price descriptors.
- **Earnings yield** – Trailing 12-month net income per share divided by price.
- **Growth** – Measures payout ratio, asset growth rate, earnings growth and recent earnings change.
- **Leverage** – Combines market leverage (market value of debt divided by market capitalisation) and book leverage (book debt divided by total assets).
- **Momentum** – Twelve-month momentum excluding the most recent month, computed as the mean of daily returns over a 252-day window shifted by

21 days.

- **Volatility** – Standard deviation of daily returns over 60 days and log high-low price range over 21 days.
- **Trading activity** – Rolling sums of dollar volume divided by average shares outstanding over annual, quarterly and five-year windows.
- **Earnings variability** – Rolling standard deviation divided by mean of earnings, sales and cash flows over five years, and accruals computed as (net income – cash flow) divided by total assets.
- **Dividend yield** – Trailing four quarters of dividends per share divided by price.
- **Management quality** – Growth rates of assets, shares outstanding and capital expenditures, and the CAPEX-to-assets ratio.

Daily alignment maps quarterly values to daily series by shifting each quarter's value to the first available trading day and forward-filling across days. After computing each descriptor, the series are winsorised at extreme quantiles (e.g., 0.1 % and 99.9 %) and then **standardised cross-sectionally using market-capitalisation weights**. The `winsorize_and_standardize_descriptor` function constructs cap weights per date, computes cap-weighted means and standard deviations, clips values at a multiple of the cap-weighted sigma and then re-standardises to have cap-weighted mean 0 and cap-weighted unit variance. Rows with too few observations are set to NaN to avoid unstable statistics. This cap-weighted normalisation differs from the earlier equal-weighted standardisation and reflects that large-cap stocks dominate market risk.

Aggregating Descriptors into Style Exposures

Descriptors are grouped into high-level style factors. After standardising individual descriptors, the code averages the z-scores across descriptors within each style (e.g., the value factor averages standardised book-to-price, sales-to-price and cash-flow-to-price). A *beta* factor is added to capture market sensitivity, and a *nonlinear size* factor complements the linear size descriptor. Industry one-hot dummies are built from GICS industry names, and because the universe is the U.S., country exposure is a single column of ones. Exposures are stored in `intermediate/loadings.parquet`, and daily market capitalisation panels are saved separately to support regression weighting.

Cross-Sectional Regression and Factor Returns

On each trading day, the model regresses the vector of stock returns on the design matrix of exposures. The model is:

$$r_{t,i} = \sum_{k=1}^{K} x_{t,i,k} f_{t,k} + u_{t,i},$$

where **x** contains style, country and industry exposures, **f** are factor returns and **u** are specific returns. Several innovations ensure robust estimates:

1. **Winsorisation of returns** – Daily simple returns are clipped at the 1 % and 99 % quantiles to mitigate extreme outliers.
2. **Regression weights** – Base weights are proportional to the inverse of prior specific variance estimates; they are then multiplied by the square-root of market capitalisation (or another exponent) and re-scaled so that the average weight is one. These weights down-weight small or volatile stocks.
3. **Cap-weighted constraints** – Sum-to-zero constraints for country and industry factor returns are computed using cap weights. The code normalises market-cap weights over valid assets and constructs constraint rows so that the cap-weighted sums of country and industry factor returns equal zero.
4. **Block-matrix solution** – The regression is solved via a block matrix inversion that enforces the constraints. Factor returns, specific returns and diagnostic metrics (variance explained and R^2) are returned for each day.

The resulting panel of factor returns (dates × factors) is stored in `output/factor_return.parquet`, and specific returns (dates × assets) are stored in `output/specific_return.parquet`.

Risk Model and Covariance Estimation

The factor covariance matrix is estimated using an EWMA of factor returns. Separate half-life parameters for volatility (42 days) and correlation (200 days) are used, and a regime proxy series scales volatility for changing market conditions. The factor covariance on day (t) is

$$\Sigma_t = D_t \, \rho_t \, D_t + \lambda \, I,$$

where (D_t) is the diagonal matrix of factor volatilities scaled by the regime multiplier, $(\rho\_t) is the factor correlation matrix and (\lambda) is a ridge parameter for numerical stability. Specific variances o$

Risk Attribution

Given factor exposures and the risk model, the proportion of variance explained by factors for an individual asset is computed as

$$R^2_{i,t} = \frac{\beta^T_{i,t} \Sigma_f(t) \, \beta_{i,t}}{\beta^T_{i,t} \Sigma_f(t) \, \beta_{i,t} + \sigma^2_{i,t}},$$

where $(\beta_{i,t})$ is the vector of factor exposures for asset i, $(\Sigma_f(t))$ is the factor covariance matrix at time t and $(\sigma^2_{i,t})$ is the specific variance. Portfolio-level risk attribution uses the factor-exposure vector $(\beta\_p) obtained by weighting asset exposures by portfolio weights. The ratio$

$$R^2_{\text{port},t} = \frac{\beta_p^T \Sigma_f(t)\, \beta_p}{\beta_p^T \Sigma_f(t)\, \beta_p + w^T \text{diag}(\sigma_t^2)\, w},$$

captures the fraction of portfolio variance explained by factors. The updated `analysis` module provides functions `factor_variance_explained_per_asset` and `factor_variance_explained_portfolio` to compute these statistics, as well as a `calc_realized_vol` function for realised volatility over a trailing window.

Robustness Considerations

Outliers and heteroskedasticity can distort factor estimates. HSR controls for these effects by:

- **Winsorising descriptors and returns** – Both descriptors and returns are clipped at extreme quantiles before standardisation or regression.
- **Cap-weighted standardisation** – Descriptors are standardised using market-cap weights so that exposures have cap-weighted mean zero and unit variance.
- **Weighted regression** – Regression weights incorporate inverse specific variance and market-cap scaling; cap-weighted sum-to-zero constraints ensure that industry and country factor returns measure relative performance.
- **Shifting fundamentals and prices** – Both quarterly fundamentals and daily market data are shifted backward by one period to avoid look-ahead bias.
- **Half-life parameters** – Separate half-life parameters for volatility, correlation and regime proxies (42, 200 and 21 days) follow common risk-model practice.

Limitations

HSR currently models only U.S. equities in the Russell 3000. Global exposures and emerging-market dynamics are not included. Descriptor sets and factor definitions are simplified relative to MSCI's commercial models; for example, the beta factor uses a simple market model with a 63-day half-life. Factor returns may differ from MSCI's due to proprietary weighting schemes and data differences; replicators must obtain equivalent raw data to recompute descriptors. Thus, HSR is a research-oriented approximation rather than a professional risk-model replacement.

Data Release and Usage

Deliverables include descriptor files, loadings, factor and specific returns, factor covariance and specific variance panels, and R^2 panels per asset and per portfolio. Researchers should cite the methodology and the commit hash when using the

data. Raw inputs (price and fundamental data) remain proprietary and are not distributed.

Versioning and Reproducibility

Each release corresponds to a specific code commit. This paper documents the version **after the `stats-for-validation` merge** (commit `25ce3c2eac78535c7d8a69a67bf60b41866df865`, merged **21 Oct 2025**). Relative to the initial commit (`f512a6ee...` on **14 Oct 2025**), this version introduces a **beta** factor, uses **cap-weighted winsorisation and standardisation**, and adds risk-attribution utilities. Researchers can reproduce results by checking out this commit and running the processing scripts. When citing this work, reference the commit hash and retrieval date; for example: "[Author(s)]. *Building an Open-Source Barra-Like Specific Return Model*. Working paper, 2025. Version 25ce3c2."

Conclusion

The **hornli-hsr** project provides an open-source implementation of a Barra-like specific return model. Updates since the initial release add a market beta factor and replace equal-weighted standardisation with cap-weighted winsorisation and standardisation. The workflow constructs descriptors from price and fundamentals, aggregates them into style factors, estimates factor returns via constrained weighted regression and builds a factor covariance matrix using EWMA techniques. Important safeguards include clipping extremes, weighting by market capitalisation and inverse specific variance, imposing cap-weighted sum-to-zero constraints and shifting data to avoid look-ahead bias. By releasing descriptor-level data and code, HSR offers a transparent platform for exploring factor-based risk models while respecting data licensing restrictions.