BUSI 722

Session 3: Combining Signals & Backtesting

Kerry Back

Combining Multiple Signals

Three Methods

We have multiple characteristics (momentum, bm, gpa, roe, ...). How do we combine them to form portfolios?

1. **Intersecting sorts:** sort stocks into groups on each characteristic simultaneously and compare group returns.
2. **Composite ranks:** rank each stock 0 to 1 on each characteristic (1 = best), then average across characteristics. Sort on the composite.
3. **Fama-MacBeth prediction:** compute FM regression coefficients over a training period, average them, and apply the average coefficients to current characteristics to get a predicted value. Sort on the prediction.

Exercise: Intersecting Sorts

- Use a simple train-test split: data through 2019 for training, 2020 onward for testing.
- Tell Claude to sort test-period stocks into quintiles on momentum and quintiles on book-to-market each month (25 groups). Compute the average return of each group each month.
- Which corner of the 5×5 table does best? Which does worst? Limitation: with more than 2–3 characteristics, groups become too sparse for reliable results.

Exercise: Composite Ranks

- Using the same train-test split, tell Claude to rank each stock from 0 to 1 each month on momentum, bm, gpa, and roe (ascending). Average the four ranks to get a composite score.
- Sort test-period stocks into deciles on the composite each month and compute average decile returns.
- Compare the top-bottom spread to what you got from intersecting sorts and single-characteristic sorts.

Exercise: Fama-MacBeth Prediction

- Using the same train-test split, tell Claude to run cross-sectional regressions of returns on momentum, bm, gpa, and roe each training month and average the coefficients.
- Apply the average coefficients to test-period characteristics to get a predicted value for each stock each month.
- Sort test-period stocks into deciles on the predicted value and compute average decile returns. Compare to composite ranks and intersecting sorts.

Discussion

- Which method produced the best spread? The best Sharpe ratio? Composite ranks and FM prediction both scale easily to many characteristics; intersecting sorts do not.
- FM prediction is a **linear** function of characteristics. It accounts for correlations among characteristics but cannot capture nonlinear interactions.
- Can we do better with a nonlinear model? This motivates **machine learning**.

The Backtesting Process

Why Not R^2?

- The conventional R^2 measures how well we predict the **level** of returns, but to make money we only need to know which stocks will do **relatively** well and which will do poorly.
- Forecasting returns is extremely hard—monthly out-of-sample R^2 values of 0.5–1% are considered excellent—yet even very low R^2 values can generate substantial portfolio profits.
- The right evaluation metric is **portfolio performance**, not R^2.

Choosing the Target Variable

What should we predict?

- **Raw returns:** simplest, but dominated by a common market component that is very hard to forecast.
- **Returns minus the market return:** removes the hard-to-predict common risk and focuses the model on cross-sectional differences.
- **Ranks or quantiles of returns:** we only need to get the ordering right. Rank and z-score each month, or classify into quintiles/deciles to turn the problem into classification.

In each case, we don't need to forecast returns precisely — we just need to **identify winners and losers**.

Why Not Cross-Validation?

- With cross-sectional data, we randomly split into train and test sets, or use k-fold cross-validation.
- With time series, **random splits use future data to predict the past**. Stock returns exhibit regime changes, trending volatility, and evolving factor premia, so a model trained on 2020 data and tested on 2015 data would have an unfair advantage.
- We must **always train on the past and test on the future**.

Simple Train-Test Split

- Train on data through date T, test on data after T. Example: train through 2019, test 2020–2025.
- All test-period predictions are genuinely **out of sample**.
- This is the simplest valid backtesting setup — good for experimenting with targets and portfolio formation before adding complexity.

Tell Claude to:

- Read the data and split into train (through 2019) and test (2020 onward).
- Fit a model on the training data to predict next month's return from the features.
- Use the fitted model to predict in the test period.

Exercise: Comparing Targets

Using the same features and simple train-test split, tell Claude to fit separate models predicting each of the following targets:

1. Raw returns
2. Returns minus the market return
3. Ranks of returns (z-scored each month) or quantile classification (e.g., top/bottom quintile)

For each, sort test-period stocks into deciles based on predictions and compute average decile returns each month. Which target produces the best **spread** between the top and bottom deciles?

Exercise: Comparing Portfolio Formation

Using the best target from the previous exercise, compare different ways of forming portfolios from predictions:

1. Sort into deciles, go **long top decile** (long only).
2. Sort into deciles, go **long top, short bottom** (long-short).
3. Use predictions directly as **portfolio weights** (long-short, rescaled).

For each, compute the mean monthly return, standard deviation, and Sharpe ratio over the test period. Which approach performs best?

From Static to Dynamic: Walk-Forward Validation

Problem with a simple split:

- The model is static — trained once on data through T and never updated.
- Markets change. A model trained through 2019 may not work well in 2024.

Walk-forward (rolling window) validation:

1. Train on months 1 through T. Predict month $T + 1$.
2. Train on months 1 through $T + 1$. Predict month $T + 2$.
3. Continue, always training on all available past data.

Each prediction is genuinely **out of sample**, and the model is **continuously updated** with new information.

The Backtesting Loop

Each period, repeat the following steps:

1. **Train:** choose model and hyperparameters using past data only, then fit on the training window.
2. **Predict:** generate predictions for next period's returns.
3. **Form portfolio:** sort or weight stocks based on predictions.

Then advance one period:

- Calculate the portfolio return over the period.
- Add the new data to the training set.
- Return to step 1 and repeat.

Evaluating the Backtest

- Sort stocks into deciles each month based on predicted values and compute the average return of each decile each month.
- Evaluate the 10 out-of-sample portfolio return series: mean return, standard deviation, Sharpe ratio, cumulative performance.
- The spread between the top and bottom deciles measures the **predictive power** of the model.

Training Pipeline

Three Main Steps:

1. Create percentile-ranked features and train LightGBM with 12-month rolling windows
2. Form decile portfolios and analyze
3. Predict current month and save model

Key Parameters:

- `TRAINING_WINDOW` = 12 months
- `N_PORTFOLIOS` = 10 deciles
- LightGBM with 100 trees, learning rate 0.05, max depth 6

Portfolio Results

- Average monthly spread (D10 − D1): 2.50%
- Decile portfolio returns from sorting on LightGBM predictions

Current Month Predictions:

- Trains on last 12 complete months
- Outputs predictions sorted highest to lowest
- Includes all features for current month analysis

Portfolio Analysis Notebook

1. Mean returns and Sharpe ratios bar charts by decile
2. Cumulative returns (linear and log scale) and summary statistics (mean, volatility, Sharpe, min, max)
3. Long-short portfolio – D10 − D1 spread performance

Model Feature Analysis

1. Feature importances pie chart – from LightGBM split gain
2. Linear regression of predictions on percentile-ranked features with coefficient bar chart
3. Comparison table – feature importance ranks vs. coefficient ranks (large differences reveal non-linear effects)

Exercise: Run and Analyze a Backtest

1. Tell Claude to run the LightGBM rolling-window pipeline on the dataset.
2. Create a Jupyter notebook that plots mean returns by decile, cumulative returns for top and bottom deciles, and the long-short spread.
3. Which deciles perform best and worst? Is the spread monotonic? Examine feature importances for surprises.