Black-Litterman End-To-End

Jetlir Duraj, Chenyu Yu

1. Idea

We combine with ML generated strategic views then optimize the portfolio allocation. The two parts of the model are jointly optimized based on mean-variance criterion. ## 2. Model Throughout we assume that the return follows normal distributions.

```
r \sim \mathcal N(\mu, \Sigma)
```

To start with we compute a benchmark mean and covariance estimation '(μ_b, Σ_b)'. In our case, this is estimated on a K-factor rolling PCA model. That is we use '$\mathcal{N}(\mu_b, \Sigma_b)$' as our prior. Our BLEnd2End model has three distinct parts.

1. Strategic Views

We generate portfolio weights based on observed signals P. Then using FFN or RNN (to be selected in the validation phase) we forecast the mean return of the strategic views

```
\hat{q} = \texttt{ViewNet}(x)
```

The confidence 'Ω' of view modeling is the view portfolio covariance scaled by 'τ_v'.

2. BL-Update

- **BLEnd2End-Bayesian**, directly update use closed form solution

    ```
    \mu_{BL-Bayes} = (P' \Omega^{-1} P' + C^{-1})^{-1}( P'\Omega^{-1}\hat q + \mu_b),\q
    \Sigma_{BL-Bayes} = ( P'\Omega^{-1}P+C^{-1})^{-1}.
    ```

- **BLEnd2End-Solver**, projecting onto view space by minimizing probability distribution distance '$D(\cdot \mid \cdot)$':

    ```
    (\mu_{BL}, \Sigma_{BL}) = \underset{P \mu = q,\, P'\Sigma P = }{\arg\min}\,D((\mu, \Sig
    ```

    ```
    (\mu_{BL}, \Sigma_{BL}) = \arg\min_{(\mu,B,d)}D\left((\mu,B\cdot B' + Diag(d\circ d))\m
    ```

3. Portfolio Allocation

Portfolio allocation is modeled by FFN. The training criterion is either batch-data-loss, which is directly the mean-variance criterion:

```
\ell_b (w ; \hat\mu_{t}, \hat\Sigma_{t}) = \sum_{t \in \mathcal B} w_t^T \hat\mu_{t} - \frac
```

or FOC-loss, which reduces the input dimension by using the first-order condition (details see main paper) of the mean-variance criterion

$\ell_{foc}(\Delta; \hat\mu_t, \hat\Sigma_t) = \sum_{t\in \mathcal{B}}\lVert (\gamma^{ra}\hat$

Following is an illustration of BLEnd2End-Bayesian implementation:

implementation

Following is an illustration of BLEnd2End-Solver implementation:BLEnd2End-disjoint

3. Data

Our asset universe contains 14 ETFs: BIL, IEF, SHY, TLT, SPY, XLB, XLE, XLF, XLI, XLK, XLP, XLU, XLV, XLY.

downloaded from WRDS TAQ database. Our input feature also contains macroeconomic timeseries from FRED.

The time horizon of this study is from 2008-02 to 2021-01-01 and it uses monthly data, where test phase the last 30% of dates and training-validation phase is the first 70% of dates. And validation takes up 30% of dates in the training-validation phase.

4. Hyperparameter Search and Model Selection

- **Number of PCA factors for the benchmark** are selected out of 3, 5.

- **Batch sizes** are selected out of 28, 56, 112. The batch sizes have to be multiples of 14 so that each batch contains all the assets within a given date.

- **Training criterion** is selected from batch-data-loss or FOC-loss

- *ViewNet*

  is selected from either of the following two neural networks:

  - FFN with hidden units of [64,32,16] or [64,64,32,32,16,16], using SiLU (sigmoid linear unit) as activation function. The dropout rate is 0.05;
  - RNN with GRUs (Gated Recurrent Units) with 4 or 8 hidden states. The sequence length of features is 6 months ad GRU has one layer. The dropout rate is 0.05.

- *BL Updater*

  is selected from one of the following:

  - **BLEnd2End-Bayesian** with view confidence scaler '$\tau_v = 1/7$'.
  - **BLEnd2End-Solver** We use Adam Optimizer with 40 iterations and learning rate 0.0001. The probability distribution distance measure is chosen out of WSD2, JSD and KL Divergence. We also use covariance factorization with 10 factors.

- *WeightSolver*

  chooses a model out of the following

  - **WeightsNet**: an FFN with hidden unit [128,64,64,32,32] that takes a flattened vector of '$(\hat{\mu}_{BL}, \hat{\Sigma}_{BL}, w_{-1})$' with length '$batchsize \times (n+2)$' ('$n = 14$' in this case) where w_{-1} is the current portfolio weights. The activation function is SiLU and the dropout rate is 0.05;
  - **WeightsDiffNet**: an FFN with hidden unit [128,64,64,32,32] that takes a flattened vector of '$(\hat{\Sigma}_{BL}, \hat{\mu}_{BL} - \hat{\Sigma}_t\, w_{-1})$' of length '$batchsize \times (n+1)$'. The activation function is SiLU and the dropout rate is 0.05.
  - **WeightSolver** uses Adam as an optimizer with a learning rate of 0.0001 and 40 iterations to minimize the train criterion directly.

  The output weights from any of the above models are normalized with bound 0.5 on ℓ_2-norm.

For each of the models proposed by taking a combination of the above architecture, we select a hyperparameter from the following:

- **Learning Rate of Global Optimization**: 0.001 or 0.0001;
- 'ℓ_1'-**regularization of global training criteria** is selected from 0, 0.01 or 0.1;
- 'ℓ_2'-**regularization for global training criteria** is also selected from 0, 0.01, or 0.1.

5. Results

Train, Validation and Test Portfolio Performance We evaluate portfolios based on following metrics: Sharpe, Sortino, Calmar ratios, maximum drawdown (MDD)

BLEnd2End portfolio performance through train, validation and test phase.

	monthly mean return (%)	monthly vol (%)	sharpe	sortino	calmar	mdd
train	2.378	3.116	0.769	8909.853	4.930	0.005
val	0.458	1.057	0.441	53.306	0.145	0.032
test	0.256	0.971	0.266	38.685	0.109	0.023

Benchmark portfolio performance through train, validation and test phase:

	monthly mean return (%)	monthly vol (%)	sharpe	sortino	calmar	mdd
train	0.399	4.800	0.084	0.709	0.013	0.314
val	-0.296	2.022	-0.149	-4.393	-0.022	0.135

	monthly mean return (%)	monthly vol (%)	sharpe	sortino	calmar	mdd
test	0.321	2.619	0.124	2.147	0.023	0.137

Test Phase BLEnd2End and Benchmark portfolio comparison
BLEnd2End achieves much higher risk-adjusted return than the no-BL-update benchmark. | | sharpe | sortino | calmar | mdd | mean leverage | |—————|—-|————|———-|——-|—————| | BLEnd2End | 0.266 | 38.685 | 0.109 | 0.023 | 0.653 | | 5-Factor Benchmark | 0.124 | 2.147 | 0.023 | 0.137 | 1.383 |

Test Phase BLEnd2End and other ETF Strategies Comparison The sector-only BLEnd2End portfolio performs better than other common ETF strategies with smaller MDD.

	sharpe	sortino	calmar	mdd
BLEnd2End	0.266	38.685	0.109	0.023
Risk Parity	-0.028	-3.620	-0.003	0.035
60-40	0.087	2.005	0.011	0.155
Equal Weights	0.076	1.687	0.010	0.147

Test Phase Sector-only BLEnd2End and other ETF Strategies Comparison The sector only BLEnd2End portfolio performs better than Sector momentum rotation and reversal with smaller MDD and higher Sharpe, Sortino and Calmar ratio.

	sharpe	sortino	calmar	mdd
BLEnd2End Sectors	0.248	39.345	0.078	0.028
Sector Momentum Rotation	0.063	1.034	0.009	0.207
Sector Momentum Reversal	0.106	1.611	0.018	0.171