When Bayes-Stein Meets Machine Learning: A Generalized Approach for Portfolio Optimization

Dimitrios Gounopoulos[1] Emmanouil Platanakis[1] Gerry Tsoukalas[2] Haoran Wu[1]

[1]University of Bath - School of Management

[2]University of Pennsylvania - The Wharton School & Boston University & Luohan Academy

What do we want to convey in this paper (ML for asset management)?

Key points:

- **FinTech is not FunTech**.
- **Interpretability**: How does ML work in the specific context?
- **Transparency**: Where is the outperformance of ML from?

Background - Markowitz mean-variance framework

The seminal Markowitz mean-variance portfolio optimization framework achieves an optimal risk-return trade-off when the input parameters (**mean returns** and **covariance matrix** of returns) are known with certainty.

However, in practice, the vanilla mean-variance framework suffers from:

- **parameter uncertainty**
- **model's extreme sensitivity**

Background - Bayes-Stein model

Many alternatives have been proposed in the literature to address this issue. Among them, strategies producing shrinkage estimators stand out because of their theoretical and empirical appeal in mitigating parameter uncertainty and offering gains for portfolio optimization.

The Bayes-Stein model of Jorion (JFQA, 1986) represents one of the first such attempts. It combines James-Stein theory and the empirical Bayes framework for portfolio optimization, and enjoys popularity **in the literature** as:

- **a go-to model for parameter estimation**
- **or a benchmark for alternative estimation methods**

Background - Bayes-Stein model

$$\hat{\boldsymbol{\mu}}_{BS} = (1 - g_{BS})\,\hat{\boldsymbol{\mu}}_S + g_{BS}\mu_G \mathbb{1}, \tag{1}$$

where $\hat{\boldsymbol{\mu}}_S$ represents the vector of sample mean returns, μ_G is the target estimator that refers to the mean return of the global minimum variance portfolio (GMV), and $\mathbb{1}$ denotes an $N \times 1$ vector of ones. The term $g_{BS} \in [0, 1]$, is the shrinkage factor, and is given by:

$$g_{BS} = \frac{N+2}{(N+2) + T\left(\hat{\boldsymbol{\mu}}_S - \mu_G \mathbb{1}\right)' \boldsymbol{\Sigma}^{-1} \left(\hat{\boldsymbol{\mu}}_S - \mu_G \mathbb{1}\right)}, \tag{2}$$

where $N > 2$ is the number of assets. As the covariance matrix $\boldsymbol{\Sigma}$ of asset returns is unknown in practice, it is replaced with $\hat{\boldsymbol{\Sigma}}_S = \frac{T-1}{T-N-2}\boldsymbol{S}$, where $\boldsymbol{S}$ is the sample covariance matrix.

Some findings about Bayes-Stein

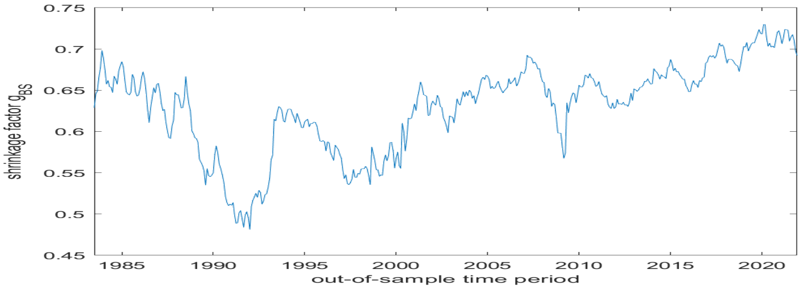
Table 1: Mean Square Forecast Error of the Mean Return Estimators

Asset	Sampe Mean ($\times 10^4$)	Bayes-Stein Shrinkage Mean($\times 10^4$)
Consumer Nondurables	16.86	16.91
Consumer Durables	55.92	55.86
Manufacturing	24.50	24.49
Energy	37.65	37.66
HiTec	44.75	44.72
Telcom	25.10	25.07
Shops	24.03	24.04
Health	21.19	21.22
Utilities	15.52	15.47
Other	26.94	26.93
Sum	292.46	292.37

Key points: Overall VS. individual components.

Some findings about Bayes-Stein

Figure 1: Shrinkage Factor for 10 Industry Portfolios



Key points: without considering the difference between individual assets.

Some findings about Bayes-Stein

Table 2: Condition Numbers for Inverse Covariance Matrix Estimates

Dataset	$\boldsymbol{S}^{-1}$		$\hat{\Sigma}_{BS}^{-1}$	
	Mean	SD	Mean	SD
Ind10	114.90	21.20	115.00	21.20
FF21	891.44	95.95	891.53	95.98
FF23	3442.20	761.34	3441.30	760.81
FF24	3469.70	766.16	3469.10	765.81
FF25	969.47	110.37	969.78	110.52
FF100BM	3018.5	840.70	3019.00	840.90
FF100OP	2815.10	1040.40	2815.80	1040.80
FF100INV	2985.00	940.08	2985.60	940.38

Key points: without considering the inverse covariance matrix.

Our motivation stems from:

- poor performance of Bayes-Stein: it can not defeat the the equally-weighted portfolio allocation rule (**1/N**) (DeMiguel, Garlappi and Uppal, 2009).
- several drawbacks shown above.
- no studies dive into it.
- some attempts have been made to improve the Black-Litterman model and move beyond the original version (Chen and Lim, 2020).

Objective

Address the **shortcomings** of the traditional Bayes-Stein model and develop a **generalized** counterpart that possesses desirable properties and performs well out of sample via the novel implementation of **well-tailored machine learning** techniques (e.g., supervised and unsupervised learning).

Overview

Table 3: Comparison of the Classical and Generalized Bayes-Stein Framework

Model Traits	Bayes-Stein Mdoel	Generalized Bayes-Stein Framework
Sample Mean	$\hat{\mu}_S$, simple average of historical asset returns	$\hat{\mu}_F$, expected returns predicted by various time-series predictors via the TW-ENet approach
Grand Mean	μ_G, mean return of GMV, depends on $\hat{\mu}_S$	μ_{GF}, mean return of GMV depends on $\hat{\mu}_F$
Shrinkage Factor	same across all assets	same across assets of the same subgroup generated by clustering ensemble
Mean Estimator	$\hat{\mu}_{BS}$, low accuracy	$\hat{\mu}_{GBS}$, relatively high accuracy
Inverse Covariance Matrix	$\hat{\Sigma}_{BS}^{-1}$, simple-based	$\hat{\Sigma}_{GBS}^{-1}$, a sparse shrinkage estimator produced by the GA-ENet

We contribute:

- We are the first to deconstruct the Bayes-Stein portfolio-optimization framework and offer ways to move beyond the original version.
- We upgrade the sample mean-based model components by exploiting the **predictability of asset returns**, contributing to the literature on asset return forecasting.
- We are the first to involve **clustering ensemble** in portfolio optimization for capturing feature **differences** between assets.
- We provide a novel shrinkage estimator (**sparse**) for the **inverse covariance matrix**, contributing to the literature on the estimation of the covariance matrix of asset returns.

Enhanced Time-series Return Forecasting via TW-ENet

We modify the conventional Elastic Net framework to allow for time-dependent weights—an approach we term time-dependent weighted Elastic Net (TW-ENet). Formally, the TW-ENEt estimation problem can be represented as follows:

$$\underset{\beta_0,\ldots,\beta_D\in\mathbb{R}}{\arg\min}\left[\frac{1}{2T}\sum_{t=1}^{T} w_t\left(r_t-\beta_0-\sum_{d=1}^{D}\beta_d x_{d,t-1}\right)^2+\tau\left(\rho\sum_{d=1}^{D}|\beta_d|+\frac{1}{2}(1-\rho)\sum_{d=1}^{D}\beta_d^2\right)\right], \tag{3}$$

where ρ represents a compromise between ridge ($\rho=0$) and LASSO ($\rho=1$), τ controls the overall penalty strength, and $w_t=t^{\delta}$ represents the time-dependent exponential weight controlled by a positive hyperparameter δ. Setting δ to zero recovers a conventional Elastic Net.

Enhanced Time-series Return Forecasting via TW-ENet

Step I - estimation in the training sample: For L weight parameter candidates, denoted by $\delta_1, \ldots, \delta_L$, we apply TW-ENet to the training sample. Further, for each candidate, we set $\rho = 0.5$ and use the corrected Akaike information criterion (AICc) to choose the optimal L1 penalizing parameter τ_*:

$$\tau_* = \underset{\tau_1,\ldots,\tau_M}{\arg\min} \left[N_S * \log(RSS / N_S) + 2 * df * N_S / (N_S - df) \right], \tag{4}$$

where RSS and df represent the residual sum of squares and degrees of freedom, respectively. $\tau_1, \ldots, \tau_M$ denotes the set of τ.

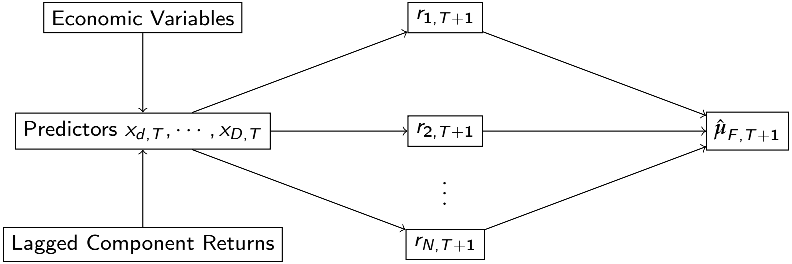
Step II - weight parameter tuning in the validation sample: We pick out the optimal weight parameter δ_* by minimizing the weighted square prediction error:

$$\delta_* = \underset{\delta_1,\ldots,\delta_L}{\arg\min} \left[\frac{\sum_{t=T/2+1}^{T} w_{t-T/2}^{T/2} \left(r_t - \beta_0 - \sum_{d=1}^{D} \beta_d x_{d,t-1} \right)^2}{\sum_{t=T/2+1}^{T} w_{t-T/2}^{T/2}} \right]. \tag{5}$$

Step III - estimation in the total sample and forecasting: We apply δ_* and TW-ENet on the total sample. As a result, we can forecast the expected asset return at the time $T+1$.

Enhanced Time-series Return Forecasting via TW-ENet

Figure 2: Graphical Representation of the Time-series Return Forecasting



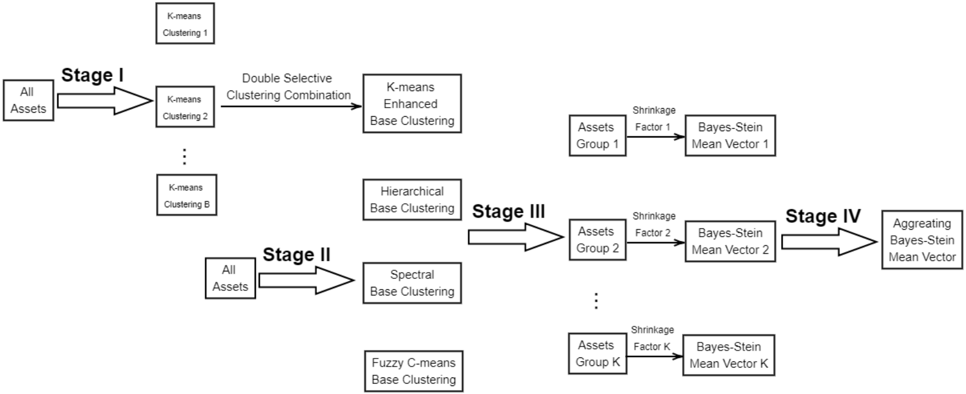
Improved Calibration of the Shrinkage Factor g_{BS}

The classical Bayes-Stein assigns the **same** shrinkage factor to all assets in the portfolio regardless of the differences of features between assets.

Inspiration:
Mynbayeva, Lamb and Zhao (EJOR, 2022): when applying shrinkage estimators to homogeneous asset subsets with indistinguishable mean returns or variance, we can create more robust portfolios than the vanilla Markowitz optimization and outperform the 1/N rule.

Improved calibration of the shrinkage factor g_{BS}

Figure 3: Grouped Bayes-Stein Shrinkage Approach



Improved Calibration of the Shrinkage Factor g_{BS}

Algorithm: double selective clustering combination algorithm

Input: B base clusterings of N assets, quality score threshold s_{thr}
Output: final clustering λ^*

for $b = 1$ *to* B **do**
  calculate the quality score CH_b
"first filtering": build the quality set $\mathcal{S}_{quality} = \{b \mid CH_b \geq s_{thr}, 1 \leq b \leq B\}$, with a base clusterings
for $q \in \mathcal{S}_{quality}$ **do**
  calculate the average normalized mutual information $m^{(q)}$ for the base clustering $\boldsymbol{\lambda}^{(q)}$
for $q \in \mathcal{S}_{quality}$ **do**
  compute the weight $w^{(q)}$ of the base clustering $\boldsymbol{\lambda}^{(q)}$
"second filtering": build the combination set $\mathcal{S}_{combination} = \left\{q \mid w^{(q)} \geq \frac{1}{a}, 1 \leq q \leq a\right\}$
for $i = 1$ *to* N **do**
  obtain the final label λ_i of asset i by weighted voting with the combination set $\mathcal{S}_{combination}$
return final clustering λ^*

Improved Calibration of the Shrinkage Factor g_{BS}

The Calinski-Harabasz score, denoted by CH, is given as follows:

$$CH = \frac{\sum_{k=1}^{K} n_k \left\| c_k - c \right\|^2}{\sum_{k=1}^{K} \sum_{i=1}^{n_k} \left\| \boldsymbol{x}_{i,k} - c_k \right\|^2} \cdot \frac{N-K}{K-1}, \tag{6}$$

where K is the number of clusters of a base clustering, N is the total number of assets, c is the global centroid of all assets, n_k and c_k are the number of assets and centroid in the kth cluster, respectively, and $\boldsymbol{x}_{i,k}$ represents the ith asset in the kth cluster.

Improved Calibration of the Shrinkage Factor g_{BS}

Denoting two label factor vectors by $\lambda^{(p)}$ and $\lambda^{(q)}$, their normalized mutual information $\Phi^{(\mathrm{NMI})}\left(\lambda^{(p)}, \lambda^{(q)}\right)$ is determined by

$$\Phi^{(\mathrm{NMI})}\left(\lambda^{(p)}, \lambda^{(q)}\right) = \frac{\sum_{h=1}^{k^{(p)}} \sum_{f=1}^{k^{(q)}} n_{h,f} \log\left(\frac{N \cdot n_{h,f}}{n_h^{(p)} n_f^{(q)}}\right)}{\sqrt{\left(\sum_{h=1}^{k^{(p)}} n_h^{(p)} \log \frac{n_h^{(p)}}{N}\right)\left(\sum_{f=1}^{k^{(q)}} n_f^{(q)} \log \frac{n_f^{(q)}}{N}\right)}}, \tag{7}$$

where $n_h^{(p)}$ and $n_f^{(q)}$ represent the number of assets in the cluster C_h of $\lambda^{(p)}$ and cluster C_f of $\lambda^{(q)}$, respectively. $n_{h,f}$ denotes the number of assets appearing in the cluster C_h and C_f simultaneously. N is the number of all assets.
Thus, the average normalized mutual information $m^{(q)}$ of the base clustering $\lambda^{(q)}$ can be computed by

$$m^{(q)} = \frac{1}{a-1} \sum_{p=1, p\neq q}^{a} \Phi^{\mathrm{NMI}}\left(\lambda^{(p)}, \lambda^{(q)}\right), \quad q = 1, \ldots, a. \tag{8}$$

Improved Estimation of the Inverse Covariance Matrix $\hat{\Sigma}_{BS}^{-1}$

In the light of Stevens(JF, 1998), the first row of the inverse covariance matrix Σ^{-1} is closely linked to a multivariate regression function that regresses the return of the first asset on those of all other assets:

$$R_{1,t} = a_1 + \sum_{n=2}^{N} \beta_{1n} R_{n,t} + \epsilon_{1,t}, \quad t = 1, \ldots, T. \tag{9}$$

The inverse covariance matrix Σ^{-1} is determined by:

$$\begin{bmatrix} \frac{1}{\sigma^2_{\epsilon_1}} & -\frac{\beta_{12}}{\sigma^2_{\epsilon_1}} & \cdots & -\frac{\beta_{1N}}{\sigma^2_{\epsilon_1}} \\ -\frac{\beta_{21}}{\sigma^2_{\epsilon_2}} & \frac{1}{\sigma^2_{\epsilon_2}} & \cdots & -\frac{\beta_{2N}}{\sigma^2_{\epsilon_2}} \\ \vdots & \vdots & \vdots & \vdots \\ -\frac{\beta_{N1}}{\sigma^2_{\epsilon_N}} & -\frac{\beta_{N2}}{\sigma^2_{\epsilon_N}} & \cdots & \frac{1}{\sigma^2_{\epsilon_N}} \end{bmatrix}. \tag{10}$$

Improved Estimation of the Inverse Covariance Matrix $\hat{\Sigma}_{BS}^{-1}$

Hedging:
The optimal weights of a vanilla mean-variance portfolio are calculated by $\frac{1}{\lambda}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$, so the holdings of the first asset hinge on $\frac{1}{\lambda\sigma_{\epsilon_1}^2}\left(\mu_1 - \sum_{n=2}^{N}\beta_{1n}\mu_n\right)$. Hence, the ith row of the inverse covariance matrix implies the hedging relationship of the ith asset with the remaining assets.

Sparse hedging:
Goto and Xu (JFQA, 2015) demonstrate that constraining the number of assets for hedging can make it more reliable and stable.

Improved Estimation of the Inverse Covariance Matrix $\hat{\Sigma}_{BS}^{-1}$

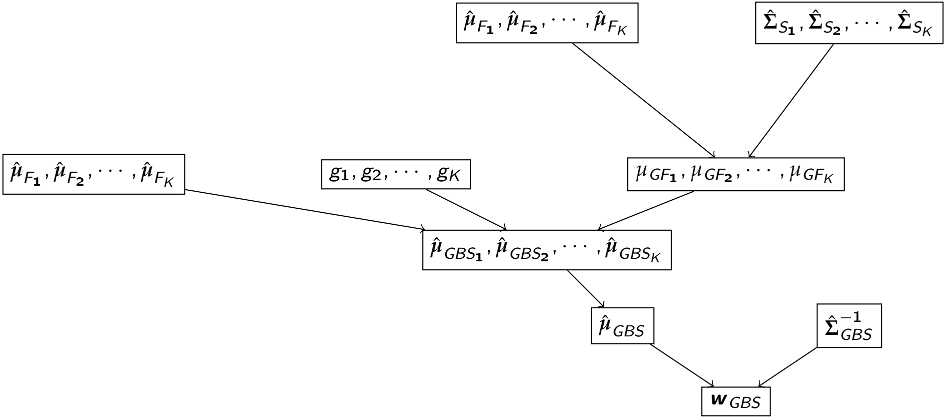
Herein we propose the graphical adaptive Elastic Net algorithm for the sparse inverse covariance matrix estimation.

$$\hat{\boldsymbol{\Theta}}_G = \text{argmax}_{\boldsymbol{\Theta}} \left(\log |\boldsymbol{\Theta}| - \text{trace}(\boldsymbol{S\Theta}) - \phi_1 \sum_{i=1}^{N} \sum_{j=1;j\neq i}^{N} \omega_{ij} |\theta_{ij}| - \phi_2 \sum_{i=1}^{N} \sum_{j=1;j\neq i}^{N} \theta_{ij}^2 \right), \quad (11)$$

where ϕ_1 and ϕ_2 denotes the non-negative L1 penalizing parameter (LASSO) and L2 penalizing parameter (ridge), respectively. ω_{ij} represents the non-negative weighting factor (adaptive lasso) for θ_{ij}. ω_{ij} is determined by $\left|-\beta_{i,j}/\sigma_{\epsilon_i}^2\right|^{-\psi}$ with $\psi > 0$, where $-\beta_{i,j}/\sigma_{\epsilon_i}^2$ is obtained by the corresponding ordinary least squares (OLS) hedging regression.

Generalized Bayes-Stein Framework

Figure 4: Graphical Representation of the Generalized Bayes-Stein Framework



We consider a dozen different datasets of monthly value-weighted stock returns over 1-month T-bills. These datasets are comparable to those utilized in previous studies such as DeMiguel, Garlappi and Uppal (2009, RFS).

Table 4: Data Description

Data	N	Time	Abbreviation
10 industry portfolios	10	1963.06-2021.12	Ind10
20 size and book-to-market portfolios and the US equity MKT	21	1963.06-2021.12	FF21
20 size and book-to-market portfolios and the MKT, SMB, and HML portfolios	23	1963.06-2021.12	FF23
20 size and book-to-market portfolios and the MKT, SMB, HML, and UMD portfolios	24	1963.06-2021.12	FF24
25 size and book-to-market portfolios	25	1963.06-2021.12	FF25
100 size and book-to-market portfolios	100	1963.06-2021.12	FF100BM
100 size and operating profitability portfolios	100	1963.07-2021.12	FF100OP
100 size and investment portfolios	100	1963.07-2021.12	FF100INV

Panel A: SRs and CERs without TCs for a 20-year expanding estimation window												
	SRs						CERs					
Dataset	$\lambda = 1$		$\lambda = 3$		$\lambda = 5$		$\lambda = 1$		$\lambda = 3$		$\lambda = 5$	
	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N
Ind10	0.6719	0.6356	0.6789	0.6356	0.6753	0.6356	0.0883 (*)	0.0756	0.0667	0.0580	0.0455	0.0403
FF21	0.6205 (*)	0.5226	0.6224 (*)	0.5226	0.6270 (*)	0.5226	0.0902 (*)	0.0760	0.0620 (*)	0.0455	0.0354 (**)	0.0150
FF23	0.6467 (**)	0.5050	0.6448 (**)	0.5050	0.6392 (**)	0.5050	0.0904 (**)	0.0688	0.0645 (**)	0.0425	0.0387 (**)	0.0161
FF24	0.6510 (**)	0.5155	0.6603 (**)	0.5155	0.6727 (**)	0.5155	0.0886 (**)	0.0676	0.0658 (**)	0.0438	0.0446 (**)	0.0200
FF25	0.7122 (***)	0.5497	0.7112 (***)	0.5497	0.7121 (***)	0.5497	0.1066 (***)	0.0783	0.0773 (***)	0.0500	0.0492 (***)	0.0217
FF100BM	0.7317 (***)	0.3816	0.7229 (***)	0.3816	0.7178 (***)	0.3816	0.1185 (***)	0.0518	0.0824 (***)	0.0206	0.0473 (***)	-0.0105
FF100OP	0.6797	0.5635	0.6772 (***)	0.5635	0.6722 (***)	0.5635	0.1104 (***)	0.0829	0.0743 (***)	0.0526	0.0383 (**)	0.0223
FF100INV	0.7401 (***)	0.5470	0.7378 (***)	0.5470	0.7359 (***)	0.5470	0.1172 (***)	0.0796	0.0842 (***)	0.0497	0.0515 (***)	0.0198

Panel B: SRs and CERs without TCs for a 40-year expanding estimation window												
	SRs						CERs					
Dataset	$\lambda = 1$		$\lambda = 3$		$\lambda = 5$		$\lambda = 1$		$\lambda = 3$		$\lambda = 5$	
	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N
Ind10	0.7866	0.7518	0.7892	0.7518	0.7911	0.7518	0.1104 (**)	0.0915	0.0863	0.0736	0.0626	0.0558
FF21	0.6882	0.6238	0.6985	0.6238	0.7167	0.6238	0.1025	0.0965	0.0750	0.0638	0.0501	0.0311
FF23	0.7300	0.6049	0.7417	0.6049	0.7441	0.6049	0.1084	0.0882	0.0818	0.0594	0.0547	0.0307
FF24	0.6862	0.6124	0.6848	0.6124	0.7006	0.6124	0.0994	0.0854	0.0717	0.0597	0.0480	0.0339
FF25	0.8089 (***)	0.6322	0.8176 (***)	0.6322	0.8274 (***)	0.6322	0.127 (***)	0.0955	0.097 (***)	0.0647	0.0683 (***)	0.0340
FF100BM	0.8779 (***)	0.5169	0.8593 (***)	0.5169	0.8575 (***)	0.5169	0.1529 (***)	0.0783	0.1108 (***)	0.0443	0.0721 (***)	0.0104
FF100OP	0.7550 (**)	0.6552	0.7562 (**)	0.6552	0.7539 (**)	0.6552	0.1226 (**)	0.1016	0.0885 (**)	0.0693	0.0540 (**)	0.0370
FF100INV	0.8022 (**)	0.6508	0.7977 (**)	0.6508	0.7949 (**)	0.6508	0.1265 (**)	0.1001	0.0942 (**)	0.0683	0.0624 (**)	0.0366

Panel A: SRs and CERs with TCs for a 20-year expanding estimation window												
	SRs						CERs					
Dataset	$\lambda = 1$		$\lambda = 3$		$\lambda = 5$		$\lambda = 1$		$\lambda = 3$		$\lambda = 5$	
	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N
Ind10	0.6664	0.6356	0.6604	0.6356	0.6548	0.6356	0.0892 (*)	0.0756	0.0648	0.058	0.0424	0.0403
FF21	0.6453 (**)	0.5226	0.6544 (**)	0.5226	0.6525 (**)	0.5226	0.0956 (**)	0.076	0.0677 (**)	0.0455	0.0392 (**)	0.015
FF23	0.6529 (**)	0.5050	0.6490 (**)	0.5050	0.6421 (**)	0.5050	0.0928 (**)	0.0688	0.0656 (**)	0.0425	0.0388 (**)	0.0161
FF24	0.6375 (*)	0.5155	0.6504 (**)	0.5155	0.6626 (**)	0.5155	0.0875 (**)	0.0676	0.0646 (**)	0.0438	0.0430 (**)	0.0200
FF25	0.7067 (***)	0.5497	0.7027 (***)	0.5497	0.7049 (***)	0.5497	0.1078 (***)	0.0783	0.0765 (***)	0.0500	0.0475 (***)	0.0217
FF100BM	0.7096 (***)	0.3816	0.7057 (***)	0.3816	0.7065 (***)	0.3816	0.1152 (***)	0.0518	0.0793 (***)	0.0206	0.0453 (***)	-0.0105
FF100OP	0.6614 (**)	0.5635	0.6644 (**)	0.5635	0.6612 (**)	0.5635	0.1075 (***)	0.0829	0.0719 (**)	0.0526	0.0362 (**)	0.0224
FF100INV	0.7242 (***)	0.5470	0.7240 (***)	0.5470	0.7259 (***)	0.5470	0.1138 (***)	0.0796	0.0816 (***)	0.0497	0.0498 (***)	0.0198

Panel B: SRs and CERs with TCs for a 40-year expanding estimation window												
	SRs						CERs					
Dataset	$\lambda = 1$		$\lambda = 3$		$\lambda = 5$		$\lambda = 1$		$\lambda = 3$		$\lambda = 5$	
	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N	GBS	1/N
Ind10	0.7613	0.7518	0.7531	0.7518	0.7524	0.7518	0.1091 (*)	0.0915	0.0819	0.0736	0.0565	0.0558
FF21	0.7102	0.6238	0.7274	0.6238	0.7413	0.6238	0.1088	0.0965	0.0808	0.0638	0.0537	0.0311
FF23	0.7476 (*)	0.6049	0.7580 (*)	0.6049	0.7612 (*)	0.6049	0.1138 (*)	0.0882	0.0856 (*)	0.0594	0.0573 (*)	0.0307
FF24	0.6834	0.6124	0.6855	0.6124	0.7025	0.6124	0.1007	0.0854	0.0724	0.0597	0.0481	0.0339
FF25	0.7802 (***)	0.6322	0.7871 (***)	0.6322	0.8004 (***)	0.6322	0.1253 (***)	0.0955	0.0930 (***)	0.0647	0.0633 (***)	0.0340
FF100BM	0.8529 (***)	0.5169	0.8567 (***)	0.5169	0.8569 (***)	0.5169	0.1493 (***)	0.0783	0.1105 (***)	0.0443	0.0720 (***)	0.0104
FF100OP	0.7387 (*)	0.6552	0.7435 (*)	0.6552	0.7470 (*)	0.6552	0.1206 (**)	0.1016	0.0864 (*)	0.0693	0.0526 (*)	0.0370
FF100INV	0.8028 (**)	0.6508	0.8025 (**)	0.6508	0.809 (***)	0.6508	0.1252 (**)	0.1001	0.0945 (**)	0.0683	0.0650 (***)	0.0366

To better understand how our enhancements are generating these gains, we **decompose our model into two sub-models**, each with some features purposefully disabled, so that we can assess their individual effectiveness.

In particular, **model 1** includes the TW-ENet approach whereas **model 2** does not. This implies that in model 1, we can only modify the sample mean component ($\hat{\mu}_S$) of the classical Bayes-Stein model from a return forecasting perspective, whereas in model 2, we can only utilize our methodologies for the shrinkage factor calibration and the inverse covariance matrix estimation.

Panel A: SRs and CERs of GBS-model 1												
	SRs						CERs					
Dataset	$\lambda = 1$		$\lambda = 3$		$\lambda = 5$		$\lambda = 1$		$\lambda = 3$		$\lambda = 5$	
	GBS-model 1	1/N	GBS-model 1	1/N	GBS-model 1	1/N	GBS-model 1	1/N	GBS-model 1	1/N	GBS-model 1	1/N
Ind10	0.6693	0.6356	0.6456	0.6356	0.6295	0.6356	0.0890 (*)	0.0756	0.0608	0.0580	0.0396	0.0403
FF21	0.6288 (*)	0.5226	0.6196	0.5226	0.6170	0.5226	0.0899	0.0760	0.0600	0.0455	0.0377 (*)	0.0150
FF23	0.5870	0.5050	0.5642	0.5050	0.5493	0.5050	0.0794	0.0688	0.0501	0.0425	0.0293	0.0161
FF24	0.5997	0.5155	0.6578 (*)	0.5155	0.6346	0.5155	0.0784	0.0676	0.0607	0.0438	0.0402 (*)	0.0200
FF25	0.6789 (***)	0.5497	0.693 (***)	0.5497	0.6952 (**)	0.5497	0.1005 (***)	0.0783	0.0723 (**)	0.0500	0.0483 (**)	0.0217
FF100BM	0.7183 (***)	0.3816	0.7311 (***)	0.3816	0.7299 (***)	0.3816	0.1161 (***)	0.0518	0.0827 (***)	0.0206	0.0526 (***)	-0.0105
FF100OP	0.6593 (**)	0.5635	0.6952 (***)	0.5635	0.6699 (**)	0.5635	0.1061 (***)	0.0829	0.0754 (***)	0.0526	0.0439 (**)	0.0224
FF100INV	0.7418 (***)	0.5470	0.7388 (***)	0.5470	0.7414 (***)	0.5470	0.1156 (***)	0.0796	0.0825 (***)	0.0497	0.0549 (***)	0.0198

Panel B: SRs and CERs of GBS-model 2												
	SRs						CERs					
Dataset	$\lambda = 1$		$\lambda = 3$		$\lambda = 5$		$\lambda = 1$		$\lambda = 3$		$\lambda = 5$	
	GBS-model 2	1/N	GBS-model 2	1/N	GBS-model 2	1/N	GBS-model 2	1/N	GBS-model 2	1/N	GBS-model 2	1/N
Ind10	0.6362	0.6356	0.6385	0.6356	0.6474	0.6356	0.0790	0.0756	0.0598	0.058	0.0417	0.0403
FF21	0.5823 (*)	0.5226	0.5867 (**)	0.5226	0.5883 (***)	0.5226	0.0917 (**)	0.0760	0.0571 (**)	0.0455	0.0244 (**)	0.015
FF23	0.5721 (**)	0.5050	0.5804 (***)	0.5050	0.5825 (***)	0.5050	0.0900 (***)	0.0688	0.0560 (**)	0.0425	0.0232	0.0161
FF24	0.5942 (**)	0.5155	0.5937 (***)	0.5155	0.5928 (***)	0.5155	0.0943 (***)	0.0676	0.0585 (**)	0.0438	0.0254	0.0200
FF25	0.5900	0.5497	0.5904	0.5497	0.5903 (*)	0.5497	0.0932 (**)	0.0783	0.0578	0.0500	0.0252	0.0217
FF100BM	0.5768 (***)	0.3816	0.5766 (***)	0.3816	0.5748 (***)	0.3816	0.0882 (***)	0.0518	0.0552 (***)	0.0206	0.0225 (***)	-0.0105
FF100OP	0.5039	0.5635	0.5288	0.5635	0.5401	0.5635	0.0752	0.0829	0.0466	0.0526	0.0167	0.0224
FF100INV	0.5744	0.5470	0.5875 (**)	0.5470	0.5925 (***)	0.5470	0.0867 (**)	0.0796	0.0570 (**)	0.0497	0.0266 (**)	0.0198

This paper sets out to improve the classical Bayes-Stein portfolio optimization model by leveraging machine learning. We identify limitations of the original model and develop specialized machine learning methods to address them. Our comprehensive **generalized Bayes-Stein framework** integrates these **innovations** and offers several **advantages**:

- Some of the core model components of the original Bayes-Stein model (the sample means vector and the grand mean) benefit from **better expected asset return estimations**, obtained through the use of our the time-dependent weighted Elastic Net (TW-ENet) approach.
- Our framework enhances the shrinkage factor measurement by employing **a four-stage Bayes-Stein shrinkage method that utilizes a clustering ensemble approach**.
- Our framework extends the traditional Bayes-Stein model by incorporating **a shrinkage estimator of the inverse covariance matrix** obtained through a graphical adaptive Elastic Net (GA-ENet) approach.

These enhancements translate into **significant out-of-sample portfolio gains over the 1/N portfolio**. But more broadly, they offer a new angle for portfolio optimization. Rather than directly implementing off-the-shelf machine learning models, we meticulously adjust them to align with the nuances of the generalized Bayes-Stein model.

This approach may offer valuable insights for both scholars and practitioners who seek to **integrate machine learning into investment decision-making processes**.

Thanks!

Feel free to ask any questions.