Asset Pricing with Nonlinear Principal Components

UdeM

January, 2021
Motivation

- Look for a parsimonious stochastic discount factor (SDF);
- Increasing number of factors explaining the cross-section (CS) (*Factor zoo.*);
- Kozak et al. (2020) show the importance of rotating the SDF into a transformed space.
- Prior literature: Rotate the SDF into the space of linear principal components (PCs);
- This paper: Rotate the SDF into the space of nonlinear principal components;
This paper

- How effective truly independent nonlinear factors are in pricing assets?
Contribution

- First paper to empirically test the effectiveness truly independent nonlinear factors
  - In an asset pricing involving the identification of an SDF that prices the CS of stocks.
Findings

- For different fixed cross-sections of returns, the nonlinear SDF consistently outperforms the linear specification;
  - For the FF25P: 65% versus 49%
  - For the 50 anomalies: 55% versus 22%
- Nonlinear SDF requires less factors.
  - For the 50 anomalies: 5 factors versus 15-20 factors
Related literature

▶ **Nonlinear factors**: Chen et al. (2009), Lawrence (2012), Gunsilius and Schennach (2019), Damianou et al. (2021)

▶ **Machine learning asset pricing models**: Feng et al. (2018), Nakagawa et al. (2019), Chen et al. (2020), and Fang and Taylor (2021).

Data

- Anomalies considered: 50 anomaly characteristics (same as Kozak et al. (2020));
- Daily returns data from November 1973 to December 2019 (2017 for Kozak et al. (2020));
- Follow the same anomalies definition as Kozak et al. (2020) to construct the anomalies.
Empirical methodology

- Let $r_t = (r_{1,t}, \ldots, r_{N,t})'$ be the vector of excess returns of $N$ portfolios, $t=1,\ldots,T$
- Let $Z_t$ be a $N$-by-$k$ matrix of asset anomaly characteristics;
- Let $F_t = Z_t' r_t$ be a $k$-by-$1$ vector of factors (raw characteristic returns or linear PCs or nonlinear PCs);
- Let $\Sigma = \text{Cov}(F)$ be a $k$-by-$k$ variance-covariance matrix of the factors;
- Let $\mu = \mathbb{E}(F)$ be a $k$-by-$1$ vector of expected factor returns;
- $SDF_t = 1 - \chi'(F_t - \mathbb{E}F_t)$
Empirical methodology

We impose two kind of penalties to estimate the SDF coefficients:

\[ L2pen : \hat{\lambda} = \arg \min_{\lambda} (\mu - \Sigma \lambda)' \Sigma^{-1} (\mu - \Sigma \lambda) + \gamma \lambda' \lambda \]  \hspace{1cm} (1)

\[ L1L2pen : \hat{\lambda} = \arg \min_{\lambda} (\mu - \Sigma \lambda)' \Sigma^{-1} (\mu - \Sigma \lambda) + \gamma_1 \sum_{i=1}^{k} |\lambda_i| + \gamma_2 \lambda' \lambda \]  \hspace{1cm} (2)

- Estimate the parameter \( \hat{\lambda} \) via Ridge or Elastic net using LAR-EN;
- choose optimally the tuning parameters \( \gamma \) or (\( \gamma_1 \) and \( \gamma_2 \)). \( \Sigma \) is a k-by-k matrix, \( \mu \) is a k-by-1 vector and \( \lambda \) is a k-by-1 vector.
LARS-EN(1/2)

- For each $\gamma_2$, the problem (2) is equivalent to a lasso problem (3);
- So, for each $\gamma_2$ we use the modified LARS algorithm to solve the problem (3) equivalently the problem (2).

$$\hat{\lambda} = \arg \min_{\lambda} \left( \mu^* - \Sigma^* \lambda \right)' \left( \mu^* - \Sigma^* \lambda \right) + \gamma_1 \sum_{i=1}^{k} |\lambda_i|$$  

(3)

where $\mu^* = (\Sigma^{-\frac{1}{2}} \mu, 0)'$ and $\Sigma^* = (\Sigma^{\frac{1}{2}}, \sqrt{\gamma_2} I)'$

- For each $\gamma_2$, we execute the algorithm described in the next slide to estimate $\hat{\lambda}$. 
LARS-EN(2/2)

1. Initialize $\hat{\lambda}^{(0)} = 0$, $A = \text{argmax}_j |\Sigma'_j \mu|$, 
   \[ \nabla \hat{\lambda}^{(0)}_A = -\text{sign}(\Sigma'_A \mu), \nabla \hat{\lambda}^{(0)}_I = 0, \quad n = 0. \]
2. While $I \neq \emptyset$ do;
3. \[ \delta_j = \min_{j \in A} - \frac{\hat{\lambda}^{(n)}_j}{\nabla \hat{\lambda}^{(n)}_j} \]
4. \[ \delta_i = \min_{i \in I} \left\{ \frac{(\Sigma_i + \Sigma_j)'(\mu - X\hat{\lambda}^{(n)})}{(\Sigma_i + \Sigma_j)'(\Sigma \nabla \hat{\lambda}^{(n)})}, \frac{(\Sigma_i - \Sigma_j)'(\mu - \Sigma \hat{\lambda}^{(n)})}{(\Sigma_i - \Sigma_j)'(\Sigma \nabla \hat{\lambda}^{(n)})} \right\} \]
   where $j$ is any index in $A$.
5. $\delta = \min(\delta_j, \delta_i)$
6. if $\delta = \delta_j$ then move $j$ from $A$ to $I$ else move $i$ from $I$ to $A$.
7. $\hat{\lambda}^{(n+1)} = \hat{\lambda}^{(n)} + \delta \nabla \hat{\lambda}^{(n)}$
8. $\nabla \hat{\lambda}^{(n+1)}_A = -\frac{1}{2} (\Sigma_A + \gamma_2 I)^{-1} \cdot \text{sign}(\hat{\lambda}^{(n+1)}_A)$
9. Update the value of $n = n + 1$
10. end while
11. Output the series of coefficients $\Lambda = (\hat{\lambda}^{(0)}, \hat{\lambda}^{(1)}, ..., \hat{\lambda}^{(k)})$
L2pen : Raw characteristics and linear PCs

**Figure** – 50 raw characteristics

**Figure** – 50 linear PCs
Sparsity
L1L2pen: Raw characteristics and linear PCs

**Figure** – 50 raw characteristics

**Figure** – 50 linear PCs
L2pen : With interaction terms

**Figure** — 2600 raw char.

**Figure** — 2600 linear PCs
L1L2pen : With interaction terms

**Figure** – 2600 raw char.

**Figure** – 2600 linear PCs
Takeaway 1

- From the previous slides, the results are quite similar to the one of Kozak et al. (2020) (replication);
- Let us turn to the second part of our analysis, which consist of integrating nonlinear factors.
Let $r_t = (r_{1,t}, ..., r_{N,t})$ be the vector of excess returns of $N$ portfolios, $t=1,...,T$

$\triangleright$ $r_t$ is orthogonalized with respect to the market and rescaled to have the same standard deviations as the market;

$\triangleright$ Nonlinear PCs construction: Follows Gunsilius and Schennach (2019)
- Extract \( N \) linear PCs from \( r \) denoted by \( f_t = (f_t^1, \ldots, f_t^N) \);
- Extract the nonlinear PCs from the first \( k \) linear PCs:
  \[ y_t = (f_t^1, \ldots, f_t^k) ; \]
- \( y \) has a density function \( g \).
- Find a map \( T \) transforming \( g(y) \) into a target density \( \Phi(x) \)
  where \( x = T(y) \)
- Change of variable formula gives:
  \[ g(y) = \Phi(T(y)) \det \left( \frac{\partial T(y)}{\partial y'} \right) \quad (4) \]
- \( T \) minimizes \( \int ||T(y) - y||^2 g(y) dy \)
- \( T(y) = \frac{\partial C(y)}{\partial y} \), where \( C \) is a convex function.
- \( C \) is determined by Gradient descent using equation (4)
Compute

\[ \tilde{J} = - \int g(y) \ln \frac{\partial T(y)}{\partial y'} dy \]  

(5)

- Extract k eigenvectors \( e = (e_1, e_2, \ldots, e_k) \) corresponding to the k largest eigenvalues of \( \tilde{J} \)

- Therefore, the \( i^{th} \) nonlinear principal component is defined by:
  \[ \tilde{f}_i = T(y)e_i, \ i = 1, 2, \ldots, k. \]
50 anomaly characteristics

- Let $r_t = (r_{1,t}, \ldots, r_{50,t})$ be the raw characteristic excess returns;
- Let $y$ be the first $k$ linear principal components;
- Set a squared grid $y$ with a size $M \times M \times \ldots \times M$ from -4 to 4 each variable;
- Estimate the Brenier map $T$ for the grid $T(y)$;
- Calculate $\tilde{J}$ over the grid points, then the eigenvectors $e = (e_1, e_2, \ldots, e_k)$;
- Interpolate the Brenier map to have the full nonlinear transformation of the data: $T(f_1, f_2, \ldots, f_k)$;
- Let $\tilde{f}_t = (\tilde{f}_{1,t}, \ldots, \tilde{f}_{k,t})$ be the time series of the $k$ nonlinear PCs;
- Since the nonlinear factors are not tradable, we construct the corresponding mimicking portfolios.
Approximation of the NLPCs using a piecewise linear function:

\[ \tilde{f}_{j,t} = \beta_{0,j} + \beta_{1,j} r_{mkt,t} + \beta_{c,j} r_t + \delta_j \max(r_{mkt,t} - k_j, 0) + \epsilon_{j,t} \quad t = 1, \ldots, T \]  

(6)

The nonlinear mimicking portfolio is:

\[ MP_{j,t}^2 = \hat{\beta}_{0,j} + \hat{\beta}_{1,j} r_{mkt,t} + \hat{\beta}_{c,j} r_t + \hat{\delta}_j \max(r_{mkt,t} - k_j, 0) \quad t = 1, \ldots, T \]  

(7)
Terminologies

Let $f_{-k}$ be a set of 50-k linear PCs, excluding the first k linear PCs and $NMP_k$ / $NPC_k$ be a set of k nonlinear MPs/PCs. $k = 2, 3, ..., 6.$

- **Base case**: Price $[f_{-k}, NMP_k]$ using risk factors derived from $[f_{-k}, NMP_k]$. In formula:
  \[ \mu = \mathbb{E}([f_{-k}, NMP_k]), \Sigma = \text{Cov}([f_{-k}, NMP_k]) \]

- **Robustness check**: Price $[f_{-k}, NMP_k]$ using risk factors derived from $[f_{-k}, NPC_k]$. In formula:
  \[ \mu = \mathbb{E}([f_{-k}, NMP_k]), \Sigma = \text{Cov}([f_{-k}, NPC_k]) \]
Terminologies

- **Base case**: Let the LARS-EN algorithm adds the factors starting by the model with all the mimicking portfolios of the NLPCs.

- **Robustness check**: Let the LARS-EN algorithm adds the factors starting by the model with no risk factors;

  Results do not depend on the mimicking portfolios (MPs), so we present the figures only for $MP^2$
Base case: 48 PCs + 2NMPs

**Figure – L1L2pen**

**Figure – L2pen**
Robustness check: 48 PCs + 2NPCs

**Figure – L1L2pen**

**Figure – L2pen**
Nonlinear principal component

Application of Kozak et al. methodology to the NLPCs

Base case: 47 PCs + 3NMPs

**Figure – L1L2pen**

**Figure – L2pen**
Robustness check: 47 PCs + 3NPCs

Figure – L1L2pen

Figure – L2pen
Takeaway 2

▶ Results do not depend on whether one use the NLPCs or the NMPs;
  ▶ There is a difference but it is not that much as one can see from previous slides;
  ▶ One explanation is the quality of the NMPs which perfectly mimic the NLPCs;
▶ Our results suggest that one should do supervised Elastic net instead of doing unsupervised Elastic net:
  ▶ Benchmark analysis is much better than no benchmark analysis.
LF vs NLF

**Figure** – LF1 versus MP1

**Figure** – LF2 versus MP2
Conclusion

- The hybrid model requires less risk factors to achieve the highest out-of-sample performance.
- Weight shifting on some anomalies. The mimicking portfolios (MPs) and the linear factors disagree on the anomalies that are marginal in terms of weights.
- We believe that the nonlinear principal components have good prediction power.
- Thus, they should be taken into account for the development of future factor model.