Learning a Factor Model via Regularized PCA

Aviad Sela Lazar & Dor Cohen

Factor Model

$$R_i = \alpha_i + \sum_{\{j=1\}}^{\{K\}} \beta_{ij} I_j + e_i$$

- The return of a stock can be explained by some factors.
- We usually assume that the number of factors is small compared to the dimensions of the problem.

Factor Model

$$R_i = \alpha_i + \sum_{\{j=1\}}^{\{K\}} \beta_{ij} I_j + e_i$$

- Clearly, We need to estimate the factors from known history of the stocks.

Factor Model

$$R_i = \alpha_i + \sum_{\{j=1\}}^{\{K\}} \beta_{ij} I_j + e_i$$

- Moreover, we need to estimate the residual risk for each stock.

Factor Model

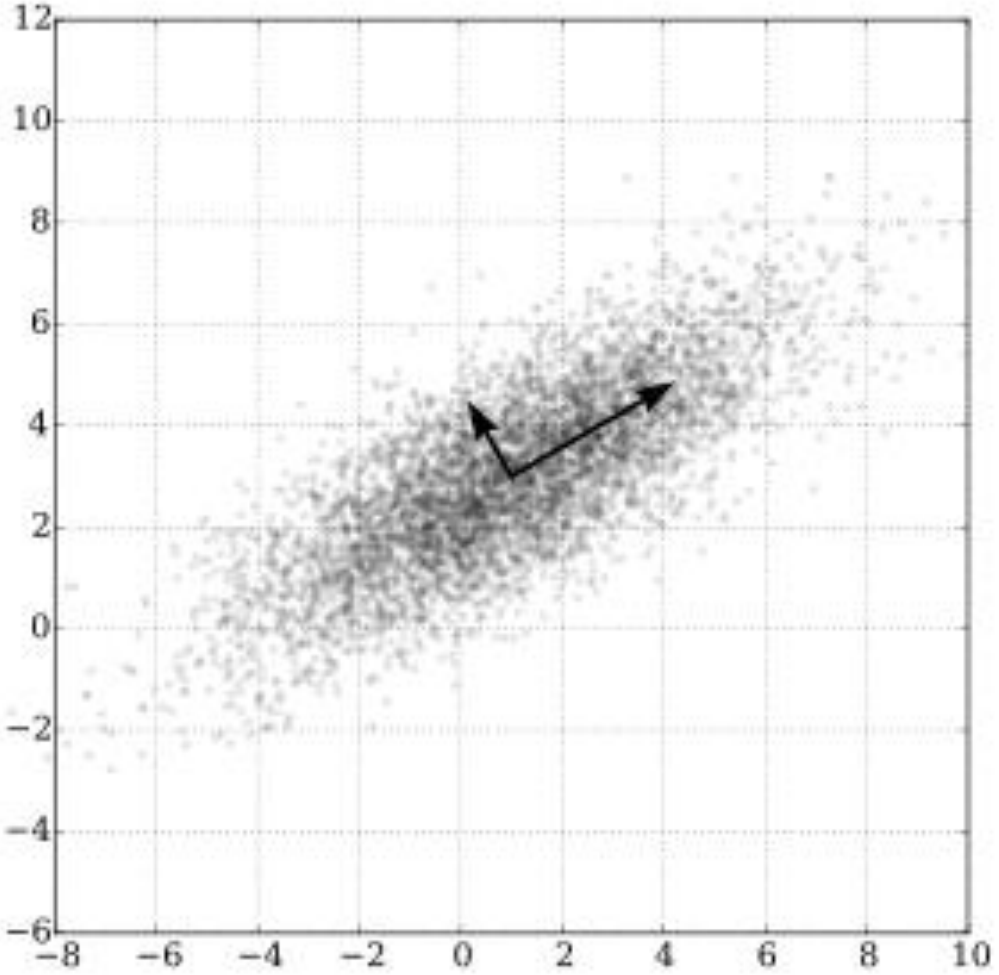
$$R_i = \alpha_i + \sum_{\{j=1\}}^{\{K\}} \beta_{ij} I_j + e_i$$

- In the article we are going to present, this component of the model and its estimation is not discussed.

PCA

Principal Component Analysis (PCA) is a procedure that produces an explanation to the data at lower dimensions.

The explanation is a set of orthogonal vectors that act as axes which explain the data variance the most.



PCA

- Generally, if the data points are of dimension M, we can select K < M principal components, and project the data to lower dimension, summarizing the data.

PCA & Factor Model

Assume we have a sample $X_1, X_2, \ldots, X_N \sim N(\mu\, \Sigma_*)$, where X_i represents the daily returns of M stocks on day i.

$$\Sigma_* = F_* + R_*\,, \qquad (F_* \in S_+^M, R \in D_+^M)$$

$$X_i = F_*^{\frac{1}{2}} Z_i + W_i\,, \qquad (Z_i \sim N(\mu, I), W_i \sim N(0, R_*))$$

Using PCA on the covariance matrix of the data, Σ_{SAM}, will allow us to estimate the factors and the residual risk.

Problem Formulation

We want to find a matrix $\hat{\Sigma}$ that will maximize the log-likelihood. Why not use the MLE (Σ_{SAM} in our case)?

It is not very good at predicting out of sample data, unless the number of samples far exceeds the dimension of the data

We will use a different method.

Problem Formulation URM

Note: for now, we will discuss a simpler case where the variances of the stocks are identical ($R^* = \sigma^2 I$).

We often believe that the data is generated by a factor model with few factors, so we will maximize the log-likelihood but constrain the rank of F.

$$\begin{aligned} \max_{F_+^M, \sigma^2 \in R_+} \quad & log p(x|\Sigma) \\ \text{s.t.} \quad & \Sigma = F + \sigma^2 I \\ & rank(F) \leq K \end{aligned}$$

Solution URM

The problem has closed form solution:

First, we compute an eigen-decomposition of the symmetrical matrix $\Sigma_{SAM} = \mathrm{BSB}^{\mathrm{T}}$, where $B = [b_1, \ldots, b_M]$ and $S = diag(\mathrm{s}_1, \ldots, \mathrm{s}_\mathrm{M})$.

$$\widehat{\sigma^2} = \frac{1}{\mathrm{M} - \mathrm{K}} \sum_{i=K+1}^{M} s_i$$

$$\hat{F} = \sum_{k=1}^{K} \left(s_k - \widehat{\sigma^2}\right) b_k b_k^T$$

The solution can be efficiently computed with PCA!

Penalizing the trace

In the previous slides, the number of factors was explicitly constrained.

However, It may be beneficial to use softer regularization that will encourage a small number of factors but allow the model to learn how many are needed.

In this article, trace penalization is proposed. **Intuition**: the trace of a matrix is also equal to the sum of its eigenvalues.

Problem Formulation UTM

$$\max_{F_+^M, \sigma^2 \in R_+} \quad log p(x|\Sigma)$$
$$\text{s.t.} \quad \Sigma = F + \sigma^2 I$$
$$tr(F) \leq t$$

Problem: not convex.

Solution:

$$define: G = \sigma^{-2} I - \Sigma^{-1}, G \in S_+^M, rank(G) = rank(F).$$

$$\max_{G_+^M, \sigma^2 \in R_+} \quad log p(x|\Sigma)$$
$$\text{s.t.} \quad \Sigma^{-1} = \sigma^{-2} I - G$$
$$tr(G) \leq t$$

Problem Formulation UTM

The final step is to change the trace constraint into regularization:

$$\max_{G_+^M, \sigma^2 \in R_+} \quad log p(x|\Sigma) - \lambda tr(G)$$
$$\text{s.t.} \quad \Sigma^{-1} = \sigma^{-2} I - G$$

The covariance matrix that is produced from this method is denoted: Σ_{UTM}^{λ}

Solution

Theorem:

Σ_{SAM} *and* Σ_{UTM}^{λ} *share the same trace and eigenvectors. Letting the eigenvalues of the two matrices, sorted in decreasing order, be denoted by* $s_1, \dots, s_m$ *and* $h_1, \dots, h_m$ *respectively, we have*

$$h_m = \max\{s_m - \frac{2\lambda}{N}, \widehat{v^{-1}}\}$$

$$(v = \sigma^{-2})$$

Algorithm 1 Procedure for computing $\Sigma^{\lambda}_{\mathrm{UTM}}$

Input: $\mathcal{X}, \lambda$

Output: $\boldsymbol{\Sigma}^{\lambda}_{\mathrm{UTM}}$

Compute eigendecomposition $\boldsymbol{\Sigma}_{\mathrm{SAM}} = \mathbf{BSB}^{\mathrm{T}}$

$v_k^{-1} \leftarrow \frac{1}{M-k}\left(k \cdot \frac{2\lambda}{N} + \sum_{m=k+1}^{M} s_m\right), \quad \forall k = 0, 1, \ldots, M-1$

$K \leftarrow \max\left\{k : s_k - \frac{2\lambda}{N} > v_k^{-1}\right\}$ // define $s_0 = \infty$

$\hat{v} \leftarrow v_K$

$h_m \leftarrow \begin{cases} s_m - \frac{2\lambda}{N} & \text{if } m \leq K \\ \hat{v}^{-1} & \text{otherwise} \end{cases}, \quad \forall m = 1, \ldots, M$

$\boldsymbol{\Sigma}^{\lambda}_{\mathrm{UTM}} \leftarrow \sum_{k=1}^{K}(h_k - \hat{v}^{-1})\mathbf{b}_k\mathbf{b}_k^{\mathrm{T}} + \hat{v}^{-1}\mathbf{I}$

UTM: Algorithm

Nonuniform Residual Variance

At the beginning we simplified the problem by assuming that the residual variance is identical for all the stocks, now we remove this assumption.

In other words, previously $\Sigma = F + \sigma^2 I$, but now $\Sigma = F + R$ where $R \in D_+^M$.

Problem Formulation STM

We could try to solve this case the same way, replacing $\sigma^{-2}I$ with a diagonal matrix V.

In practice, this approach doesn't work.

$$\max_{G_+^M, V \in D_+^M} \quad logp(x|\Sigma) - \lambda tr(G)$$
$$\text{s.t.} \quad \Sigma^{-1} = V - G$$

Problem Formulation STM

The loss on out of sample data:

$$L(\Sigma, \Sigma^*) = E[\log P(x|\Sigma)] = -\frac{1}{2}\left(Mlog(2\pi) + \log(\det(\Sigma)) + tr(\Sigma^{-1}\Sigma_*)\right)$$

Given some $T \in D_+^M$ we have that:

$$L(T\Sigma T^T, T\Sigma_* T^T) = L(\Sigma, \Sigma_*) - \log(\det(T))$$

$$T\Sigma T^T = TFT^T + TRT^T$$

If $\forall i, T_{ii} = \alpha \frac{1}{\sqrt{R_{ii}}}$ then $TRT^T = \alpha I$.

We can form a covariance matrix where the residual variance is identical.

Problem Formulation STM

Finally, we get this optimization problem:

$$\max_{G_+^M, v \in R_+, T \in D_+^M} \quad logp(Tx|\Sigma) - \lambda tr(G)$$
$$\text{s.t.} \quad \Sigma^{-1} = vI - G$$
$$log(detT) \geq 0$$

The covariance matrix that is produced from this method is denoted: Σ_{STM}^{λ}

Algorithm 3 Procedure for solving STM

Input: $\mathcal{X}, \lambda$

Output: $\boldsymbol{\Sigma}_{\mathrm{STM}}^{\lambda}$

$\mathbf{T} \leftarrow \mathbf{I}$.

repeat

$\boldsymbol{\Sigma} \leftarrow \mathrm{UTM}(\mathbf{T}\mathcal{X}, \lambda)$

$\mathbf{T} \leftarrow \underset{\bar{\mathbf{T}} \in \mathbb{D}_{+}^{M}}{\operatorname{argmax}} \log p(\bar{\mathbf{T}}\mathcal{X}|\boldsymbol{\Sigma}),\ \text{s.t.}\ \log\det\bar{\mathbf{T}} \geq 0$

until converge

$\boldsymbol{\Sigma}_{\mathrm{STM}}^{\lambda} \leftarrow \mathbf{T}^{-1}\boldsymbol{\Sigma}\mathbf{T}^{-\mathrm{T}}$

STM: Algorithm