

Dynamic Factor Model について

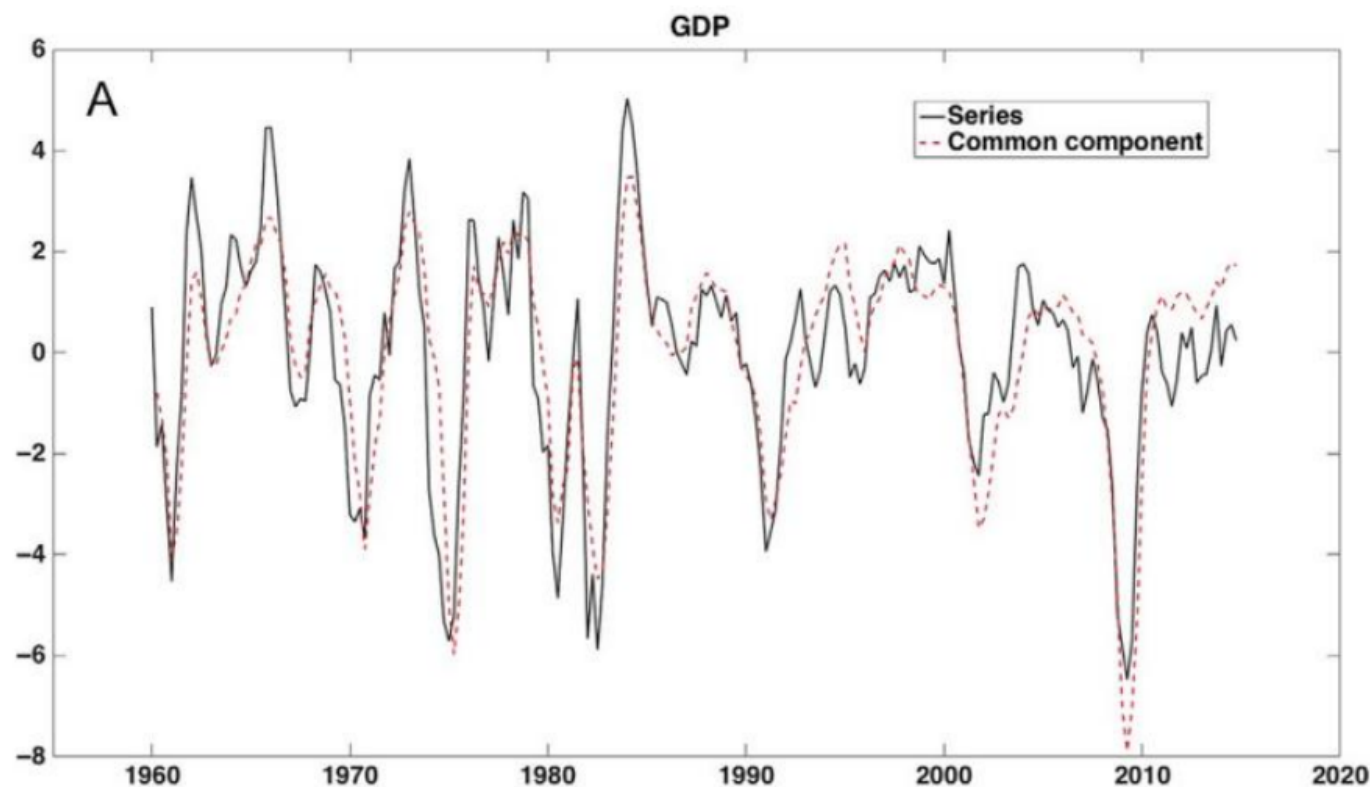
2023/7/12

仲山

ダイナミックファクターモデル(DFM)とは?

- DFMは、多数の時系列データの共通の動きを、少数の観測できない潜在的なファクターから説明しようとするモデル
- 多数のマクロ経済変数からGDPを推計するナウキャストリングなどに用いられている

- 例 1 : アメリカの58の四半期ごとの経済指標に対してシングルファクターのDFMを適用した結果とGDP



[引用]http://www.princeton.edu/~mwatson/papers/Stock_Watson_HOM_Vol2

- 推定にGDPは使用していないが、GDPとの決定係数は0.73と高くなっている

ダイナミックファクターモデルの基本的な形

$$y_t = \Lambda f_t + u_t$$

$$f_t = A_1 f_{t-1} + \cdots + A_p f_{t-p} + \eta_t \quad \eta_t \sim N(0, I)$$

$$u_t = C_1 u_{t-1} + \cdots + C_p u_{t-p} + \epsilon_t \quad \epsilon_t \sim N(0, \Sigma_\epsilon)$$

y_t : 時点 t での観測データ（例えば、経済指標）

f_t : 時点 t での共通ファクター

Λ : 共通ファクターの影響度合いを示すパラメータ行列（ファクターローディング・因子負荷行列）

u_t : 時点 t での観測データの誤差項

A : 共通ファクターの自己相関を示すパラメータ、 η_t は時点 t での共通要素の誤差項

C : 誤差項の自己相関を示すパラメータ、 ϵ_t は時点 t での誤差項の誤差項

※共通ファクターの誤差項の分散は単位行列

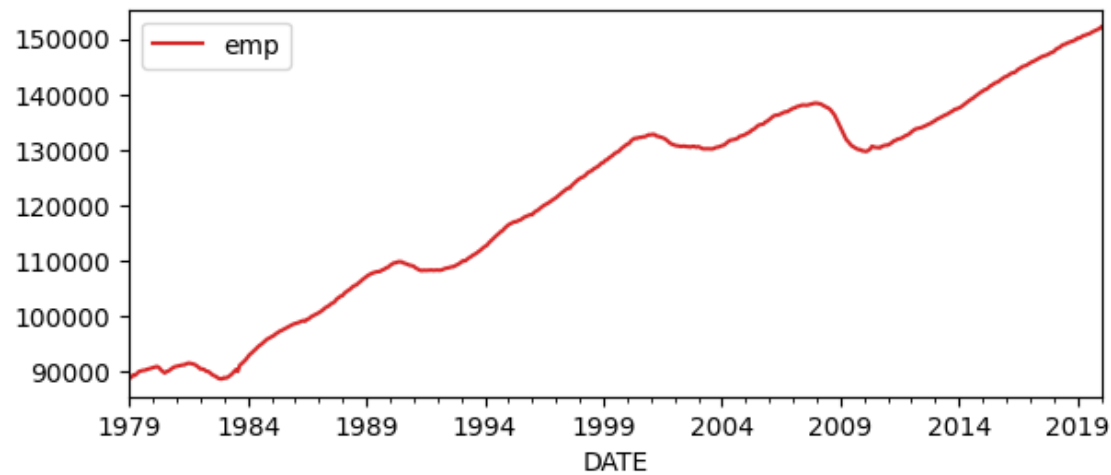
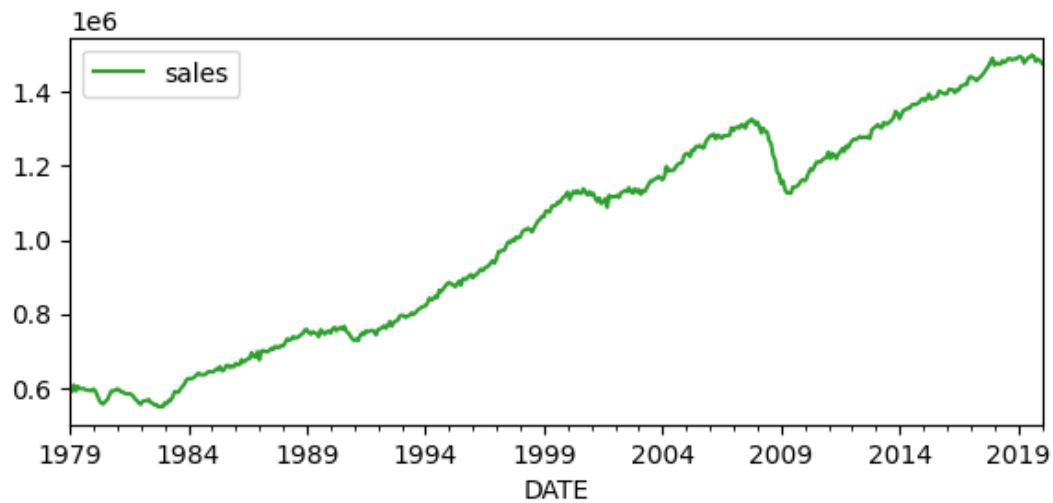
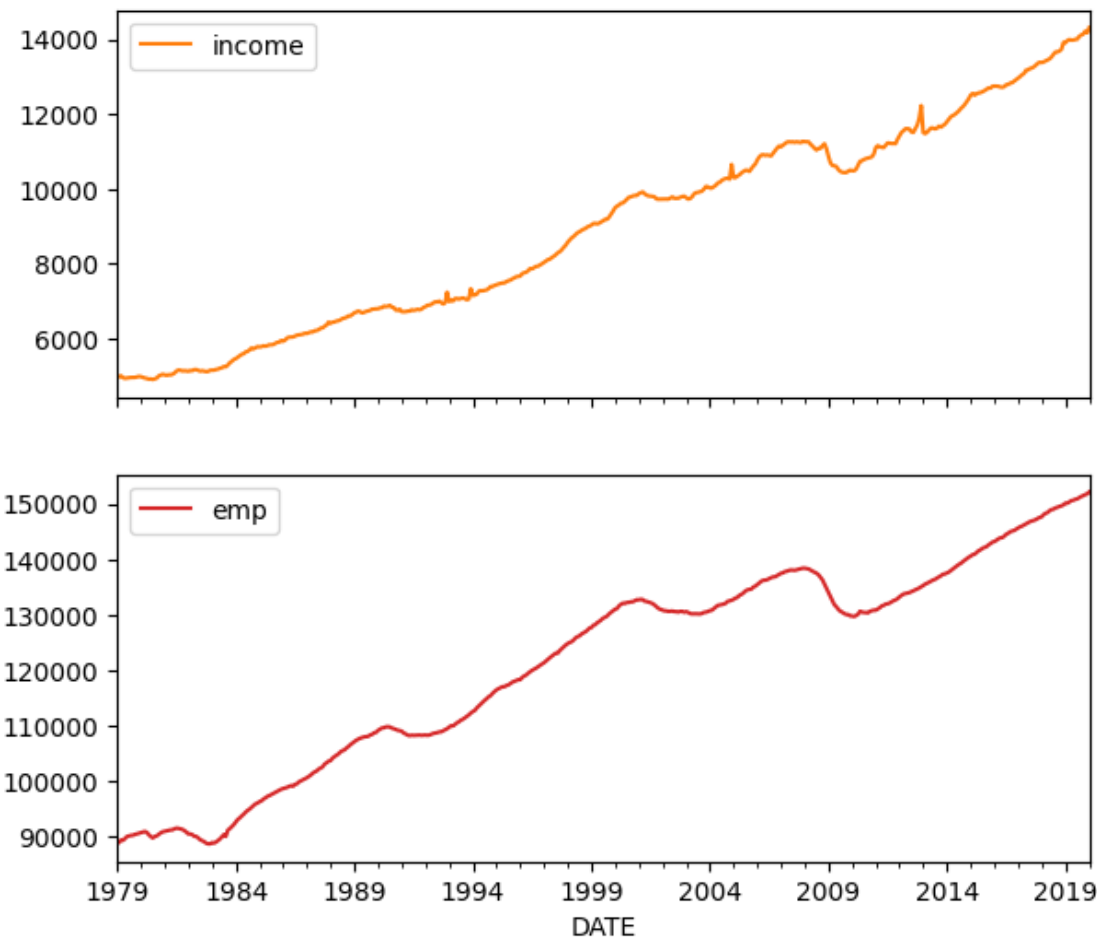
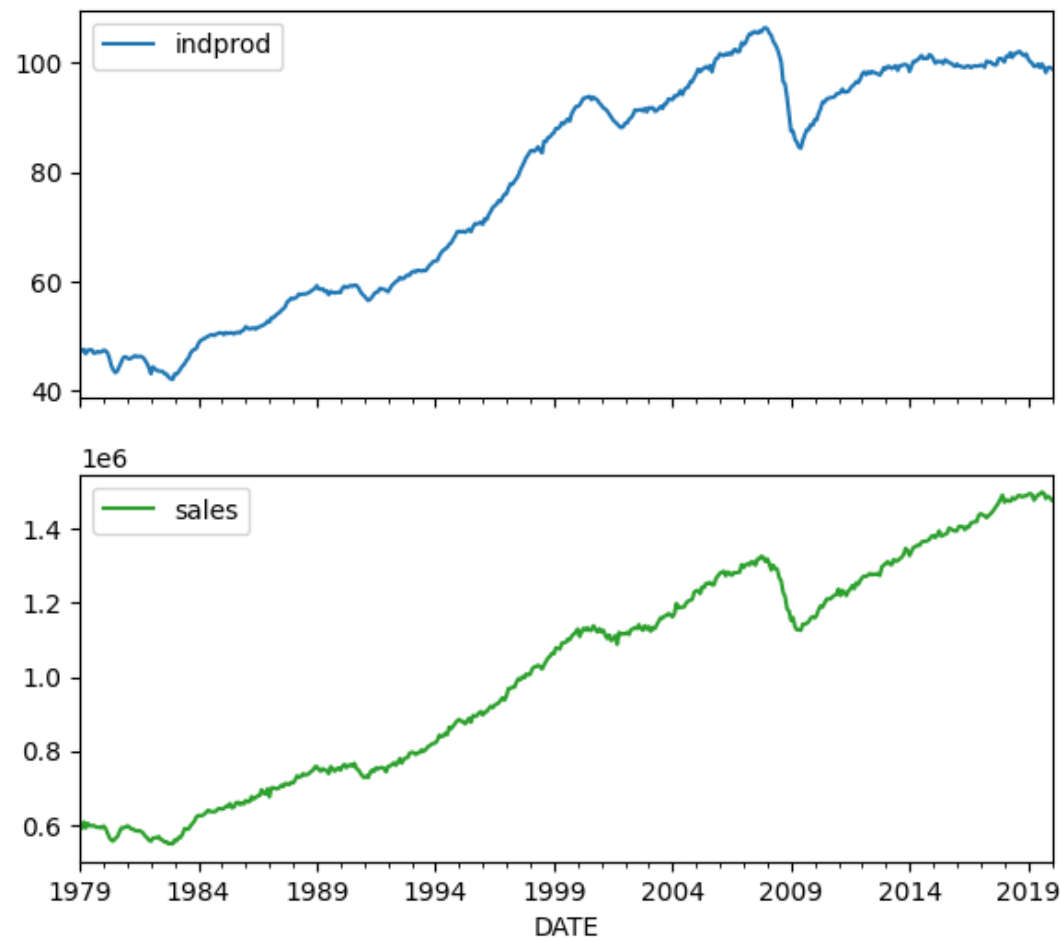
例：以下の米国の4つのマクロ経済の系列から1つの共通ファクターを作成する

- 鉱工業生産（製造業）[indprod]: Industrial Production: Manufacturing (NAICS)
- 実質個人所得 [income]: Real personal income excluding current transfer receipts
- 実質製造業および貿易産業売上高 [sales] : Real Manufacturing and Trade Industries Sales
- 非農業部門従業員数[emp] : All Employees, Total Nonfarm

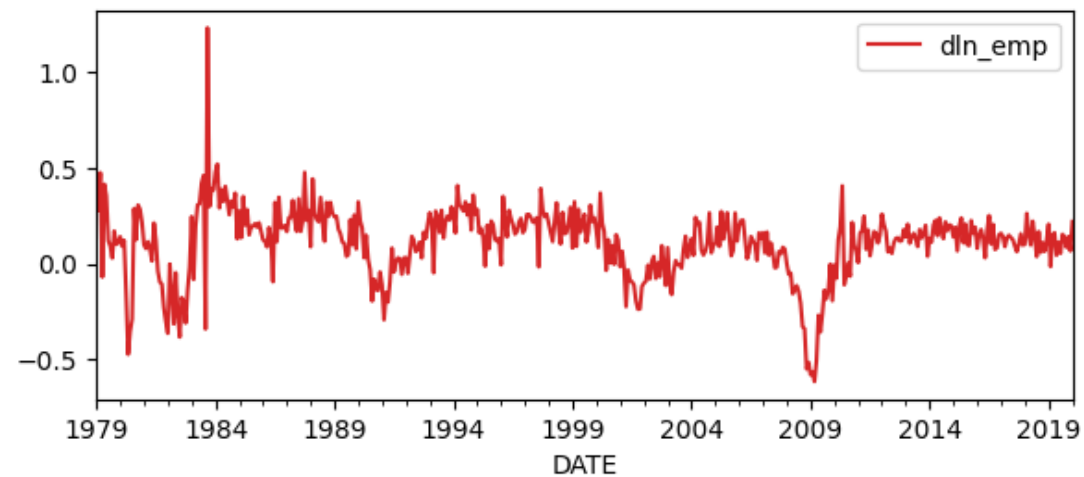
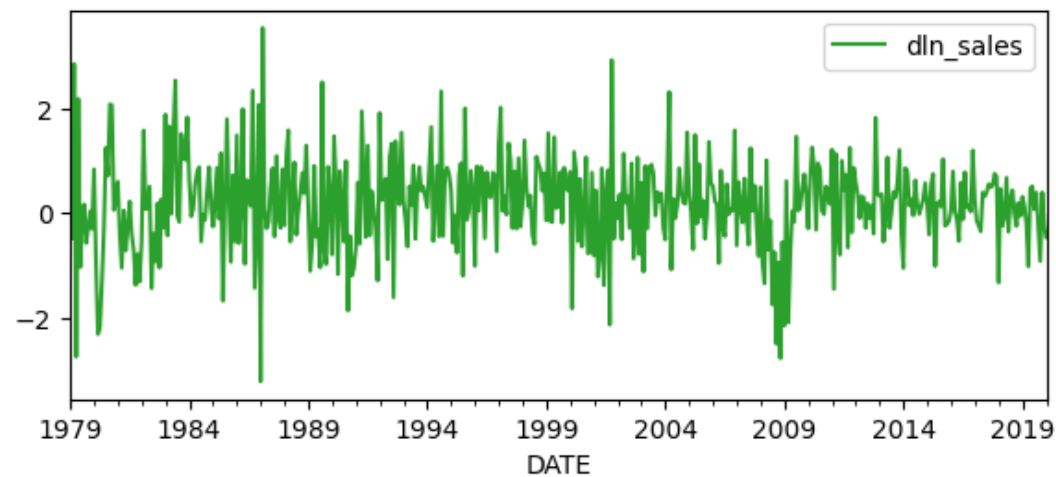
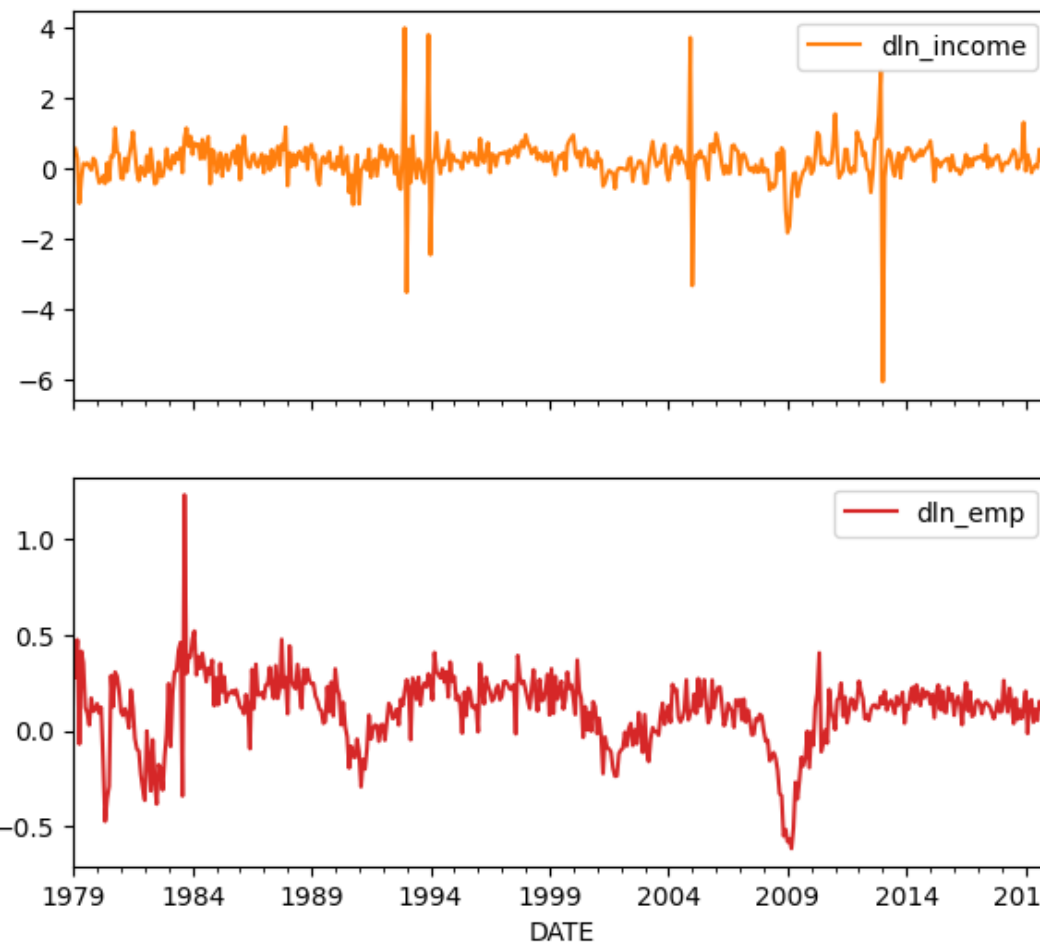
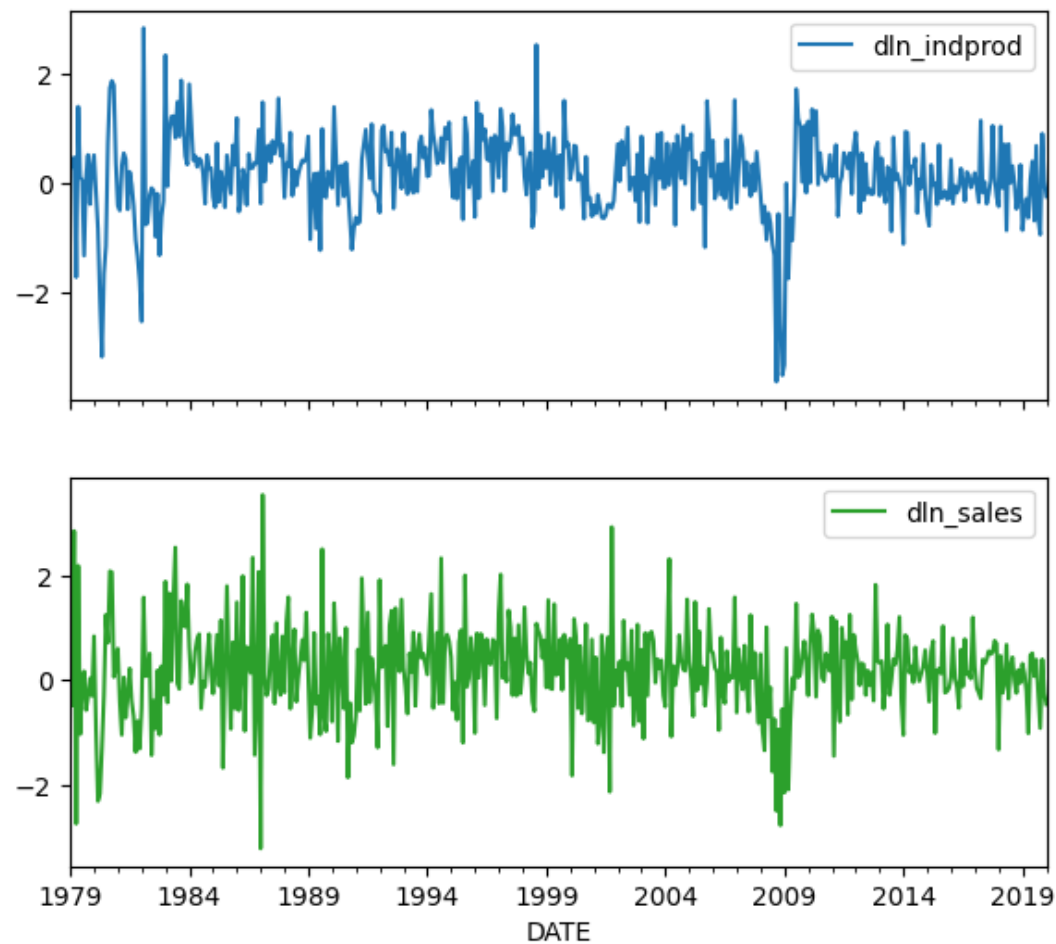
※データはすべて月次で季節調整済み

※データはFred (<https://fred.stlouisfed.org>) より取得

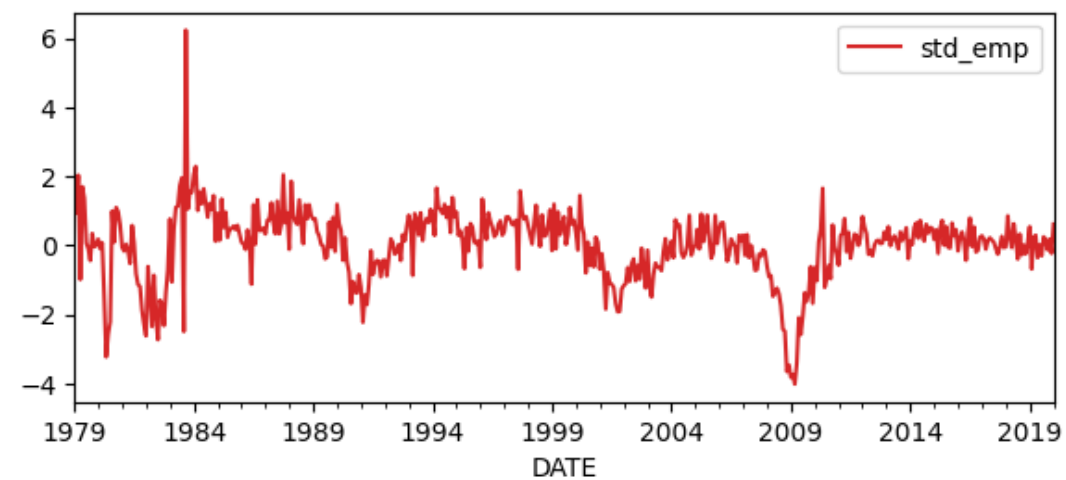
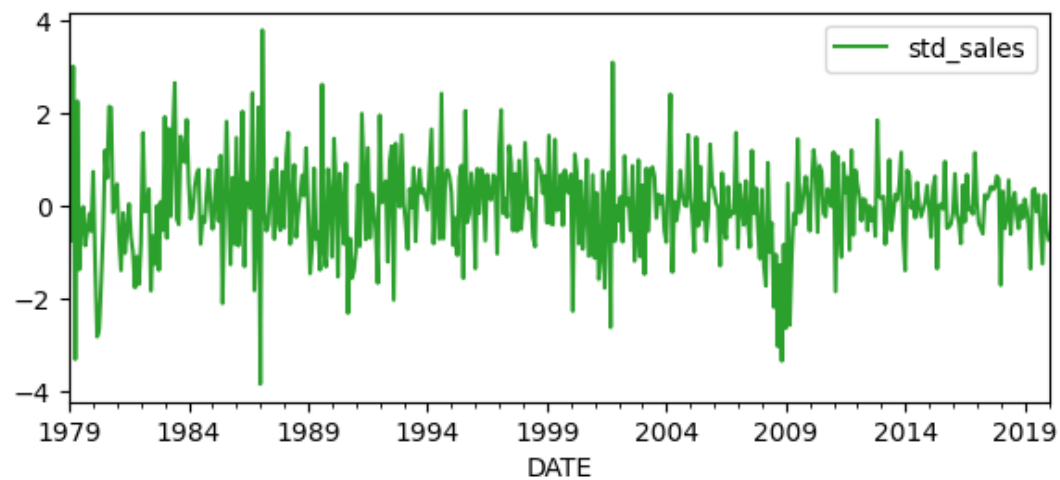
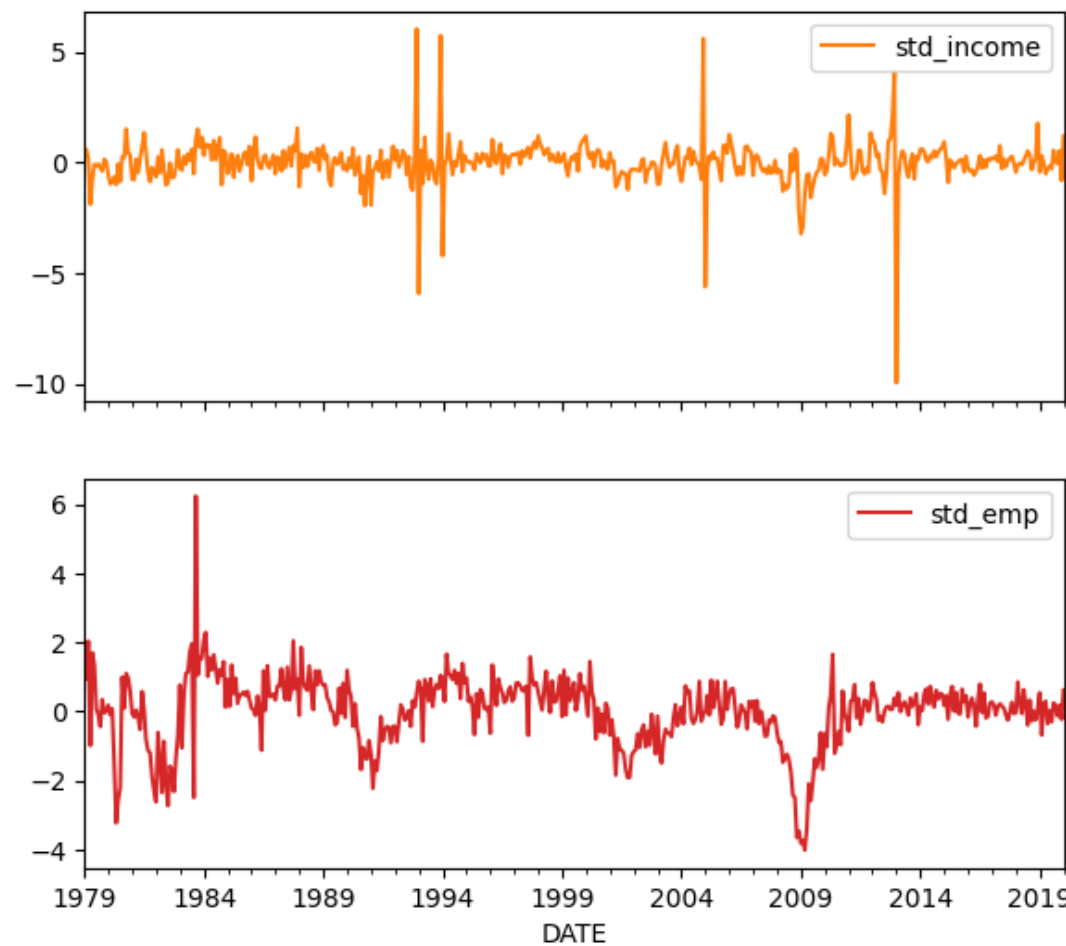
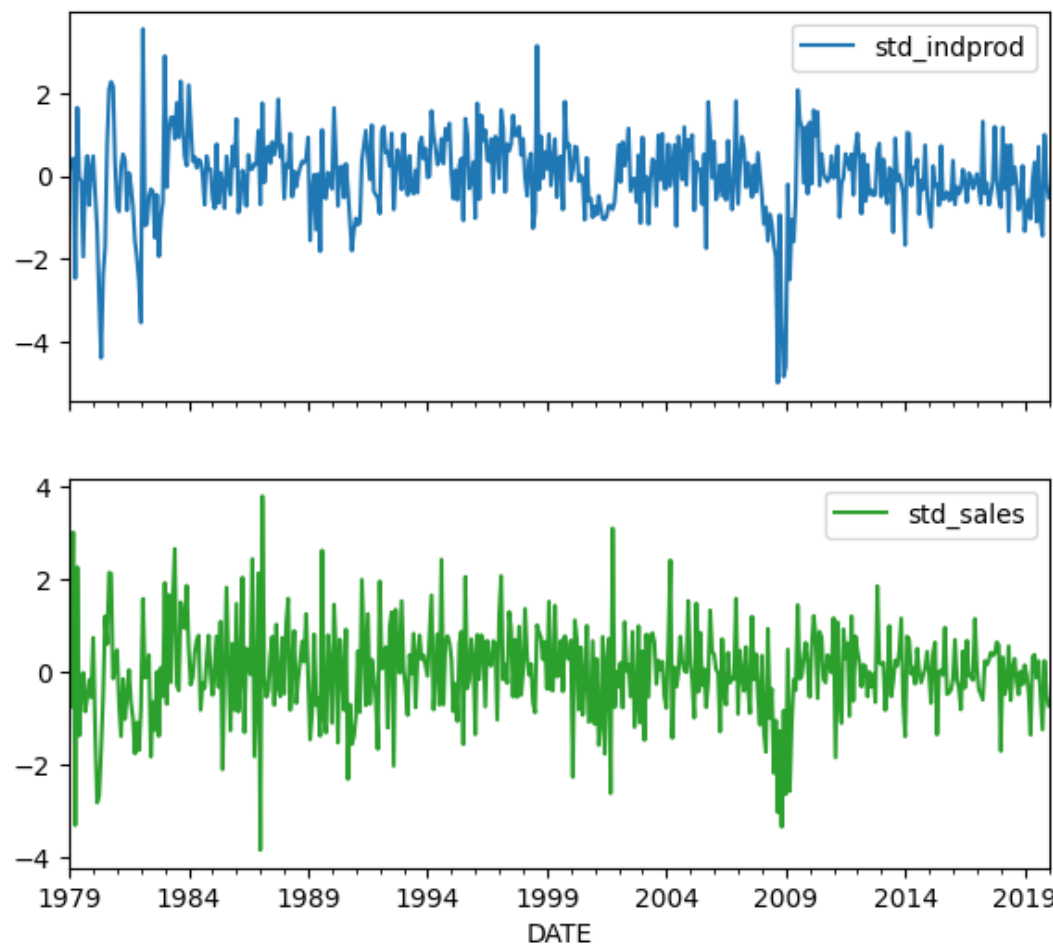
取得した4つの系列(1979年～2019年)



単位根過程を棄却できないため、対数差分系列に変換する



さらに標準化した系列を分析対象とする



モデルの設定

今回は以下のような設定で分析を行った

$$\begin{aligned}y_{i,t} &= \lambda_i f_t + u_{i,t} \\ f_t &= a_1 f_{t-1} + \eta_t \quad \eta_t \sim N(0, I) \\ u_{i,t} &= c_{i,1} u_{i,t-1} + \epsilon_{i,t} \quad \epsilon_t \sim N(0, \Sigma_\epsilon)\end{aligned}$$

- 共通ファクターは1つとする
- 共通ファクター及び誤差項はAR(1)に従うとする
- 誤差項はVARではなく、それぞれ独立したAR過程に従うものとする

パラメータの推定

- pythonのライブラリstatsmodelsを使用してパラメータを推定する
- Scipyで利用可能なパウエル法による実行結果を初期値として、最尤法でパラメータを推定

```
import statsmodels.api as sm

model = sm.tsa.DynamicFactor(endog, k_factors=1, factor_order=1, error_order=1)
initial_res = model.fit(method='powell', disp=False)
res = model.fit(initial_res.params, disp=False)
```

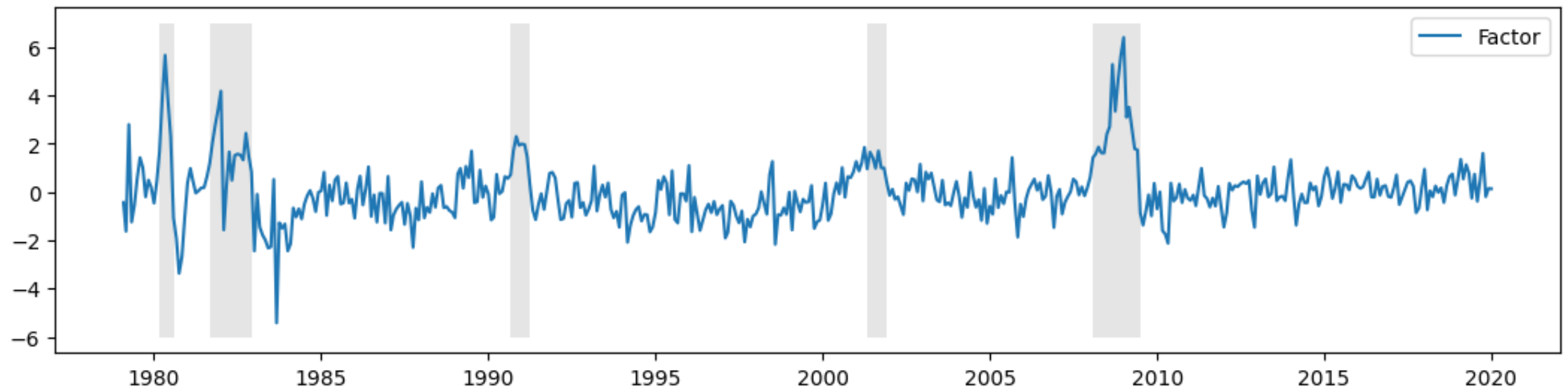
結果

```
Statespace Model Results
=====
Dep. Variable:      ['std_indprod', 'std_income', 'std_sales', 'std_emp']  No. Observations:      492
Model:              DynamicFactor(factors=1, order=1)                    Log Likelihood          -2423.402
                   + AR(1) errors                                         AIC                     4872.803
Date:               Tue, 11 Jul 2023                                       BIC                     4927.383
Time:               01:36:26                                           HQIC                    4894.235
Sample:             02-01-1979
                   - 01-01-2020
Covariance Type:    opg
=====
              coef      std err      z      P>|z|      [0.025      0.975]
-----
loading.f1.std_indprod      -0.6234      0.032     -19.476      0.000     -0.686     -0.561
loading.f1.std_income      -0.2379      0.034      -7.077      0.000     -0.304     -0.172
loading.f1.std_sales      -0.3918      0.024     -16.376      0.000     -0.439     -0.345
loading.f1.std_emp      -0.5283      0.044     -12.085      0.000     -0.614     -0.443
sigma2.std_indprod       0.3150      0.038       8.267      0.000       0.240       0.390
sigma2.std_income       0.8777      0.020     43.448      0.000       0.838       0.917
sigma2.std_sales       0.6018      0.038     16.036      0.000       0.528       0.675
sigma2.std_emp       0.3156      0.033       9.539      0.000       0.251       0.380
L1.f1.f1       0.6313      0.031     20.461      0.000       0.571       0.692
L1.e(std_indprod).e(std_indprod)     -0.3274      0.077      -4.227      0.000     -0.479     -0.176
L1.e(std_income).e(std_income)     -0.1725      0.020      -8.836      0.000     -0.211     -0.134
L1.e(std_sales).e(std_sales)     -0.4087      0.038     -10.832      0.000     -0.483     -0.335
L1.e(std_emp).e(std_emp)       0.6085      0.041     14.975      0.000       0.529       0.688
=====
Ljung-Box (L1) (Q):      3.45, 0.05, 0.81, 33.01  Jarque-Bera (JB):      263.14, 16490.94, 14.79, 7441.79
Prob(Q):      0.06, 0.82, 0.37, 0.00  Prob(JB):      0.00, 0.00, 0.00, 0.00
Heteroskedasticity (H):  0.75, 3.43, 0.37, 0.39  Skew:      0.27, -1.18, 0.15, 1.34
Prob(H) (two-sided):      0.07, 0.00, 0.00, 0.00  Kurtosis:      6.54, 31.26, 3.79, 21.86
=====
```

Warnings:

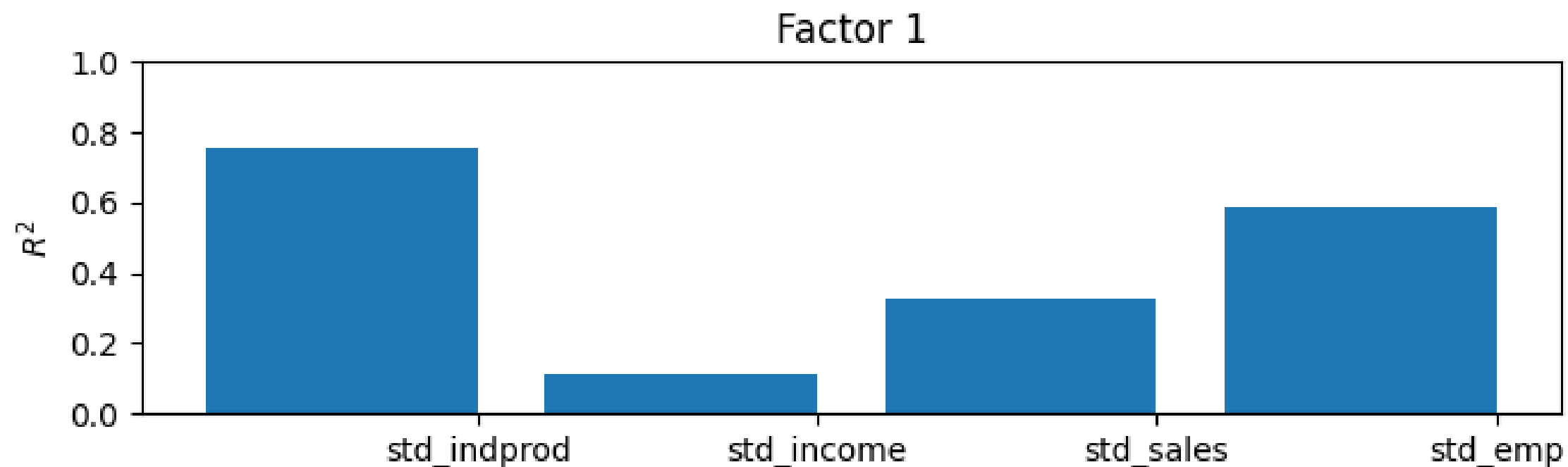
[1] Covariance matrix calculated using the outer product of gradients (complex-step).

共通ファクター



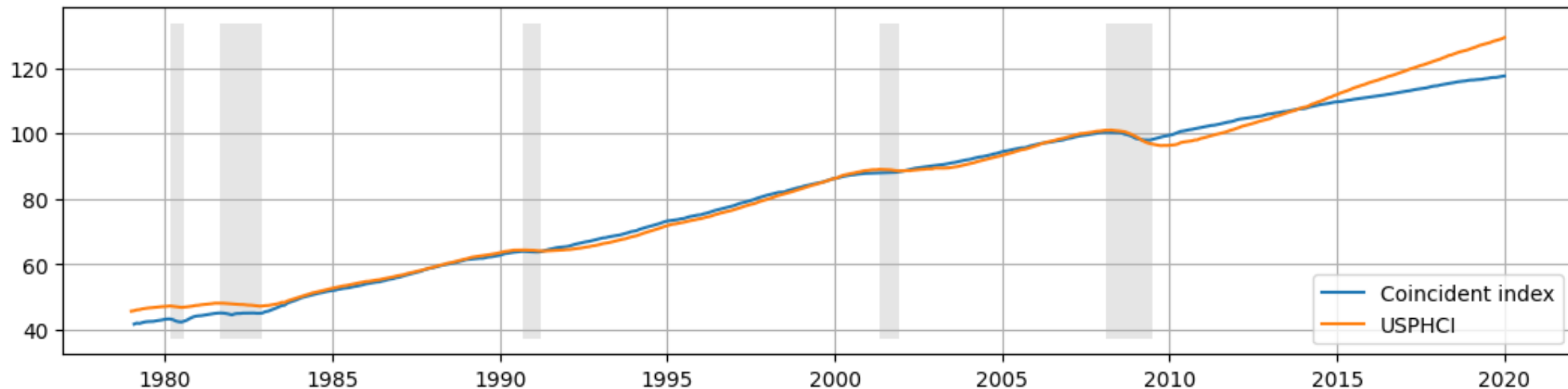
※背景が灰色の部分はNBER(全米経済研究所)が認定した景気後退局面
⇒ 景気後退局面でスコアが上昇している

共通ファクターによる決定係数



他系列との比較

フィラデルフィア連銀の米国景気一致指数（USPHCI）との比較

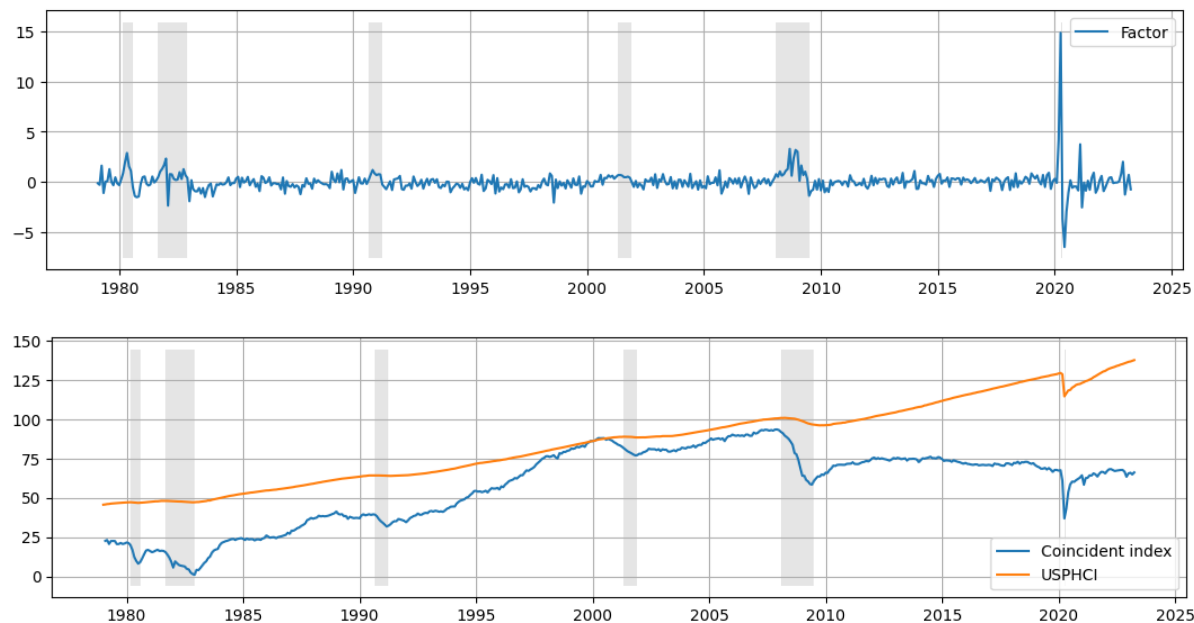


※ フィラデルフィア連銀の米国景気一致指数は、州ごとの非農業部門雇用者数、製造業の平均労働時間、失業率、賃金・給与（物価調整後）の4つの系列を使用してダイナミックファクターモデルで算出している

※ 2000年1月を基準として2系列の水準を合わせている

課題

- コロナ期を含めると全く違う結果となってしまった



- ファクターの数を変えると結果が大きく変わってしまうこともあったため、DFMのパラメータ推定方法など中身の理解をする

参考

- https://www.dir.co.jp/report/research/economics/outlook/20210226_022119.pdf
- https://www.statsmodels.org/dev/examples/notebooks/generated/statespace_dfm_coincident.html
- http://www.princeton.edu/~mwatson/papers/Stock_Watson_HOM_Vol2
- https://www.apir.or.jp/wp/wp-content/uploads/20200911_S4-3_MrKajita.pdf
- <https://www.atlantafed.org/-/media/documents/research/publications/wp/2014/wp1407.pdf>
- https://www.chadfulton.com/topics/statespace_large_dynamic_factor_models.html

番外編

- 今回はMarpというライブラリを用いてスライドを作成した
- MarpはMarkdownからプレゼンスライドを生成してくれるツール
- ChatGPTに「Marp for VS Codeを使用したMarkdown形式のコードで出力してください」等の指示をして、出力されたものをVS Codeに貼り付ければスライドが完成する（かもしれない）

- VS Codeでプレビューでき、PDFエクスポートもできる

