SESSION 2

Exercise Hints & Worked Solutions

Nowcasting with High-Frequency Data

Contents of this companion

This companion is meant to be used in stages. Try the exercise first. Open the hint if you need a direction, then compare your work with the detailed solution. The solutions emphasize the reasoning behind each calculation, not only the final number.

On the website, the hints and solutions are collapsed until selected. In the PDF, every hint and solution is printed in full so the document can be read offline or used as a conventional solutions manual.

Exercises and Solutions

Exercise 1: The covariance a factor model implies

Problem. [core, pencil] A one-factor model has three standardized series with loadings $\lambda = (0.9,\ 0.6,\ 0.3)'$, factor variance 1, and mutually uncorrelated idiosyncratic components. Write the full 3×3 covariance matrix of the observed series. Compute each series' common share and each pairwise correlation. Which series is the most valuable witness to the factor, and what, if anything, does series 3 contribute?

Hint

For standardized series, the diagonal entries are 1 by construction. Every off-diagonal entry must be carried by the factor: compute $\mathrm{Cov}(x_i, x_j)$ from $x_i = \lambda_i f + e_i$ using the orthogonality of factor and idiosyncratic components.

Detailed solution

Each series is $x_i = \lambda_i f + e_i$ with $\mathrm{Var}(f) = 1$, $\mathrm{Cov}(f, e_i) = 0$, and $\mathrm{Cov}(e_i, e_j) = 0$ for $i \neq j$. Two facts follow directly.

For the diagonal, standardization requires $1 = \lambda_i^2 + \sigma_{e,i}^2$, so the idiosyncratic variances are

$$\sigma_{e,1}^2 = 1 - 0.81 = 0.19, \qquad \sigma_{e,2}^2 = 1 - 0.36 = 0.64, \qquad \sigma_{e,3}^2 = 1 - 0.09 = 0.91.$$

For the off-diagonals, expanding the covariance and dropping every term that orthogonality kills leaves only the factor term:

$$\mathrm{Cov}(x_i, x_j) = \mathrm{Cov}(\lambda_i f + e_i,\ \lambda_j f + e_j) = \lambda_i \lambda_j.$$

The covariance matrix, which for standardized series is also the correlation matrix, is therefore

$$\Sigma = \begin{pmatrix} 1 & 0.54 & 0.27 \\ 0.54 & 1 & 0.18 \\ 0.27 & 0.18 & 1 \end{pmatrix}.$$

The common shares are the squared loadings:

Series	Loading λ_i	Common share λ_i^2	Idiosyncratic share
1	0.9	0.81	0.19
2	0.6	0.36	0.64
3	0.3	0.09	0.91

Series 1 is the most valuable witness: 81 percent of its variance is factor. Series 3 is not worthless (its covariances with the other series are nonzero, and in a precision-weighted extraction it would receive a small positive weight), but 91 percent of its movement is its own story, so a large swing in series 3 alone should barely move a factor estimate.

Two checks are worth making. First, every off-diagonal has the product form $\lambda_i \lambda_j$; a one-factor model cannot generate any other pattern, which is what makes the model testable. Second, a common mistake is to conflate the loading with the common share: series 2's loading of 0.6 may sound informative, but its common share is only 0.36. The loading is a correlation with the factor (for standardized data with unit factor variance), while the common share is that correlation squared.

Exercise 2: Two witnesses

Problem. [core, pencil] One factor with prior $f \sim \mathcal{N}(0, 1)$ is measured by two standardized series, each with loading $\lambda = 0.8$ and idiosyncratic variance $\sigma_e^2 = 0.36$, idiosyncratic components independent. The observed values are $x_1 = -1.5$ and $x_2 = -0.5$. Compute the posterior mean and variance of f, twice: once by Gaussian conditioning on the joint distribution and once by precision weighting. Compare with the posterior after observing only x_1, and explain in words why the second witness pulled the estimate in the direction it did.

Hint

Build the 3×3 joint covariance of (f, x_1, x_2) from $\mathrm{Cov}(f, x_i) = \lambda$ and $\mathrm{Cov}(x_1, x_2) = \lambda^2$, then condition. For the second route, posterior precision equals prior precision plus one term λ^2/σ_e^2 per

witness.

Detailed solution

Route 1: joint conditioning. The model implies

$$\begin{pmatrix} f \\ x_1 \\ x_2 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0.8 & 0.8 \\ 0.8 & 1 & 0.64 \\ 0.8 & 0.64 & 1 \end{pmatrix} \right),$$

using $\mathrm{Var}(x_i) = \lambda^2 + \sigma_e^2 = 1$ and $\mathrm{Cov}(x_1, x_2) = \lambda^2 = 0.64$. The conditioning weights are $w' = \Sigma_{fx}\Sigma_{xx}^{-1}$. Inverting the symmetric 2×2 block,

$$\Sigma_{xx}^{-1} = \frac{1}{1 - 0.64^2} \begin{pmatrix} 1 & -0.64 \\ -0.64 & 1 \end{pmatrix} = \frac{1}{0.5904} \begin{pmatrix} 1 & -0.64 \\ -0.64 & 1 \end{pmatrix},$$

so each weight is

$$w_1 = w_2 = \frac{0.8(1 - 0.64)}{0.5904} = \frac{0.288}{0.5904} = \frac{20}{41} \approx 0.488.$$

The posterior mean and variance follow:

$$\mathbb{E}[f \mid x_1, x_2] = \tfrac{20}{41}(-1.5) + \tfrac{20}{41}(-0.5) = -\tfrac{40}{41} \approx -0.976,$$

$$\mathrm{Var}(f \mid x_1, x_2) = 1 - 2 \cdot \tfrac{20}{41} \cdot 0.8 = 1 - \tfrac{32}{41} = \tfrac{9}{41} \approx 0.220.$$

Route 2: precision weighting. With independent idiosyncratic noise, precisions add:

$$\mathrm{Var}(f \mid x_1, x_2)^{-1} = 1 + \frac{0.64}{0.36} + \frac{0.64}{0.36} = \frac{41}{9},$$

reproducing $\mathrm{Var} = 9/41$ exactly, and

$$\mathbb{E}[f \mid x_1, x_2] = \frac{9}{41} \cdot \frac{0.8}{0.36}\,(x_1 + x_2) = \frac{9}{41} \cdot \frac{20}{9}\,(-2.0) = -\frac{40}{41}.$$

The two routes agree because they are the same projection computed in different coordinates; the precision form is the one that scales to many series.

One witness. Observing only $x_1 = -1.5$: the gain is $\lambda/(\lambda^2 + \sigma_e^2) = 0.8$, so $\mathbb{E}[f \mid x_1] = -1.2$ and $\mathrm{Var}(f \mid x_1) = 1 - 0.8 \times 0.8 = 0.36$.

The second witness moved the estimate *up*, from -1.2 to -0.976, while cutting the variance from 0.36 to 0.220. Both effects are informative. Series 2's milder reading is evidence that part of series 1's plunge was idiosyncratic rather than common; with only one witness, the model had no way to make that separation. And regardless of the values observed, a second independent witness adds precision.

A common mistake is to average the two readings to -1.0 and apply the one-series gain of 0.8, giving -0.8. That treats the average of two witnesses as if it were a single witness with the original noise level; in fact the average has idiosyncratic variance $\sigma_e^2/2$, which is why the correct weights differ.

Exercise 3: The witness floor

Problem. [core, pencil] N standardized series each measure a factor $f \sim \mathcal{N}(0, 1)$ with loading λ and idiosyncratic variance σ_e^2. Derive the posterior variance of f when the idiosyncratic components are (a) independent and (b) equicorrelated with correlation ρ_e. Show that in case (b) the posterior variance has a strictly positive limit as $N \to \infty$, and prove the statement: an unlimited supply of equicorrelated witnesses is worth exactly $1/\rho_e$ independent ones. Evaluate everything for $\lambda = 0.8$, $\sigma_e^2 = 0.36$, $\rho_e = 0.25$.

Hint

By symmetry, the optimal combination of identical witnesses is their average $\bar{x}$. Work out $\mathrm{Var}(\bar{e})$ under each correlation structure, then treat $\bar{x} = \lambda f + \bar{e}$ as a single witness.

Detailed solution

Because all witnesses share the same loading and noise variance, no linear combination can beat the equally weighted average $\bar{x} = \lambda f + \bar{e}$, where $\bar{e} = N^{-1} \sum_i e_i$. Conditioning on $\bar{x}$ is therefore equivalent to conditioning on the full vector, and a single-witness update with noise variance $\mathrm{Var}(\bar{e})$ gives the answer.

(a) Independent noise. $\mathrm{Var}(\bar{e}) = \sigma_e^2/N$, so the posterior precision is

$$1 + \frac{\lambda^2}{\sigma_e^2/N} = 1 + \frac{N\lambda^2}{\sigma_e^2}, \qquad \mathrm{Var}(f \mid x_1, \ldots, x_N) = \frac{1}{1 + N\lambda^2/\sigma_e^2} \;\longrightarrow\; 0.$$

(b) Equicorrelated noise. The average of N equicorrelated variables has variance

$$\mathrm{Var}(\bar{e}) = \frac{\sigma_e^2}{N^2}\left[N + N(N-1)\rho_e\right] = \sigma_e^2\left[\frac{1-\rho_e}{N} + \rho_e\right],$$

which falls with N but only to the limit $\sigma_e^2\rho_e$. The posterior variance is

$$\mathrm{Var}(f \mid x_1, \ldots, x_N) = \frac{1}{1 + \dfrac{N\lambda^2}{\sigma_e^2\left[1 + (N-1)\rho_e\right]}} \;\longrightarrow\; \frac{1}{1 + \lambda^2/(\sigma_e^2\rho_e)} > 0.$$

The equivalence. In the limit, the information contributed by the panel is $\lambda^2/(\sigma_e^2\rho_e) = (1/\rho_e)\cdot\lambda^2/\sigma_e^2$: exactly $1/\rho_e$ times the information of one independent witness. An infinite equicorrelated panel therefore matches a finite independent panel of $1/\rho_e$ series, no more.

Numbers. With $\lambda = 0.8$, $\sigma_e^2 = 0.36$, one witness contributes information $0.64/0.36 = 16/9$. Independent witnesses give posterior variances

$$N = 1:\ 0.360, \qquad N = 2:\ \tfrac{9}{41} \approx 0.220, \qquad N = 4:\ \tfrac{9}{73} \approx 0.123, \qquad N = 40:\ 0.014.$$

With $\rho_e = 0.25$, the floor is

$$\frac{1}{1 + 0.64/(0.36 \times 0.25)} = \frac{1}{1 + 64/9} = \frac{9}{73} \approx 0.123$$

which is identical to four independent witnesses, as the equivalence with $1/\rho_e = 4$ requires. As a finite-N check, the formula in (b) gives 0.260 at $N = 2$ and 0.155 at $N = 10$: the tenth correlated witness has already stopped paying.

The practical reading: a panel's effective size is governed by its sectoral breadth, not its column count. Ten series from three sectors are closer to three witnesses than to ten, which is why adding a fifth series from an already-crowded sector is nearly free of information, the point documented empirically by Boivin and Ng (2006).

Exercise 4: Rotation, concretely

Problem. [core, pencil] (a) For a one-factor model, exhibit two loading–factor pairs besides (Λ, f_τ) that generate identical data, and identify what each violates in the normalization $\mathrm{Var}(f_\tau) = I$, $\Lambda'\Lambda$ diagonal, plus a sign convention. (b) For $k = 2$, show that any pair $(\Lambda M^{-1}, Mf_\tau)$ with M invertible fits identically, and determine which matrices M survive the full normalization.

Hint

For (b), impose the restrictions one at a time: which M preserve $\mathrm{Var}(Mf_\tau) = I$? Of those, which also keep $(\Lambda M^{-1})'(\Lambda M^{-1})$ diagonal with distinct diagonal entries?

Detailed solution

(a) One factor. Two constructions:

- *Rescaling:* $(2\Lambda,\ f_\tau/2)$. Every product $\lambda_i f_\tau$ is unchanged, so every fitted value and every implied covariance is unchanged. It violates the variance restriction: $\mathrm{Var}(f_\tau/2) = 1/4 \neq 1$.
- *Sign flip:* $(-\Lambda,\ -f_\tau)$. Again all products are unchanged. It satisfies the variance restriction, since $\mathrm{Var}(-f_\tau) = 1$, and is excluded only by the sign convention, for example "the factor correlates positively with activity." This is why the sign convention is a necessary part of the normalization, not a nicety.

(b) Two factors. For invertible M,

$$(\Lambda M^{-1})(Mf_\tau) = \Lambda(M^{-1}M)f_\tau = \Lambda f_\tau,$$

so the common component of every series is unchanged, and with it the entire distribution of the observed panel. Now impose the normalization on the transformed pair.

Variance restriction. $\mathrm{Var}(Mf_\tau) = MM'$ must equal I, so M must be orthogonal: a rotation or reflection. This kills rescalings but leaves a continuum of rotations.

Diagonality restriction. Writing $\tilde{\Lambda} = \Lambda M'$ (using $M^{-1} = M'$ for orthogonal M), we need $\tilde{\Lambda}'\tilde{\Lambda} = M\Lambda'\Lambda M'$ diagonal. If $\Lambda'\Lambda$ is diagonal with *distinct* diagonal entries, the orthogonal matrices that keep it diagonal are exactly the signed permutations: reorderings of the factors combined with sign flips.

Ordering and sign conventions. Ordering the factors by explained variance removes the permutations; one sign rule per factor removes the flips. What remains is $M = I$: the normalization has done its job.

The edge case is instructive. If two diagonal entries of $\Lambda'\Lambda$ are *equal*, so that two factors explain exactly equal variance, rotations within that two-dimensional subspace survive, and those two factors are genuinely not separately identified. In practice near-equal eigenvalues produce the near-failure: factor estimates that swap roles across samples. A common mistake in applied work is to interpret factor 2 and factor 3 of a panel separately when their eigenvalues are nearly tied; the rotation

ambiguity makes the pair, not each member, the identified object.

Exercise 5: Replicate the P1 extraction

Problem. [core, computational, data] Reproduce the Practicum 1 factor extraction from scratch: pull the four weekly FRED series, transform each to a 52-week log difference with the claims series sign-flipped, standardize on the pre-2020 calibration window, and compute the first principal component of the calibration panel. Verify your weights against the reference build. Then compute each series' common share with respect to your estimated factor and rank the four series by usefulness. Does the ranking match the weights? Should it?

Hint

The entire extraction is one singular value decomposition of the standardized calibration panel; the weights are the first right singular vector. For the common shares, correlate each calibration-window series with the estimated factor and square.

Detailed solution

The complete computation:

```
import io, urllib.request
import numpy as np
import pandas as pd

def fred(series_id: str) -> pd.Series:
    url = f"https://fred.stlouisfed.org/graph/fredgraph.csv?id={series_id}"
    with urllib.request.urlopen(url) as r:
        df = pd.read_csv(io.BytesIO(r.read()), na_values=".",
                         parse_dates=["observation_date"])
    return df.set_index("observation_date")[series_id].astype(float)

INPUTS = {
    "ICSA":           ("initial claims", -1),
    "CCSA":           ("continued claims", -1),
    "TOTBKCR":        ("bank credit", +1),
    "CCLACBW027SBOG": ("consumer loans", +1),
}
raw = {sid: fred(sid) for sid in INPUTS}

def yoy(s):
    w = s.resample("W-SAT").last()
    return 100 * np.log(w / w.shift(52))

panel = pd.DataFrame({name: sign * yoy(raw[sid])
                      for sid, (name, sign) in INPUTS.items()}).dropna()

CAL = panel.loc[:"2019-12-31"]          # calibration window
z   = (panel - CAL.mean()) / CAL.std()  # normal-times yardstick
zc  = z.loc[:"2019-12-31"]

U, S, Vt = np.linalg.svd(zc.values, full_matrices=False)
w = Vt[0]
factor = pd.Series(z.values @ w, index=z.index)
if factor.corr(z["initial claims"]) < 0:   # sign convention
    factor, w = -factor, -w
```

At the retrieval date used for these notes, the computation reproduces the reference build exactly: weights 0.69, 0.69, −0.08, 0.20 for initial claims, continued claims, bank credit, and consumer loans, with the first component explaining 48 percent of calibration-panel variance. The assertions are deliberately tolerant: FRED revises claims for several weeks after each release, so a later retrieval produces slightly different numbers. That is not a bug in the exercise; it is the vintage problem of Session 1, and noting your retrieval date is part of a complete answer.

The common shares, computed on the calibration window:

Series	Weight	Common share
Initial claims	0.69	0.91
Continued claims	0.69	0.91
Consumer loans	0.20	0.08
Bank credit	−0.08	0.01

The ranking by common share matches the ranking by absolute weight, and with one factor it must, approximately: both quantities are monotone in how strongly the series comoves with the factor the panel defines. (With standardized data and a single factor, the correlation between a series and the extracted factor is close to proportional to its weight; the common share is its square.) The two rankings can diverge in a multi-factor model, where a series may earn a large weight on factor 2 while having a modest overall common share.

The economics deserves the last word. The panel is effectively two witnesses, the claims pair, plus two bystanders. Exercise 3 says what that implies: the tracker's precision is that of a small panel however many credit series are appended, and improving it requires a new *sector*, not a new column.

Exercise 6: State-space fluency

Problem. [core, pencil] Write the full system matrices (T, R, Q, Z, H) for a dynamic factor model with $k = 2$ factors following a VAR(2), AR(1) idiosyncratic components, and $N = 5$ monthly series. Give every matrix's dimensions and the state dimension. Then, for the one-factor version of the model, explain where the five-weight quarterly aggregation row for GDP enters and exactly which additional states it requires.

Hint

Stack $(f_\tau', f_{\tau-1}')'$ in companion form for the factor VAR, then append the five idiosyncratic states. For the GDP part, write monthly latent GDP growth as loading times factor plus its own idiosyncratic term, and ask which lags the tent weights touch.

Detailed solution

The $k = 2$, VAR(2), $N = 5$ system. The state stacks the current and lagged factor vectors and the five idiosyncratic components:

$$\alpha_\tau = \begin{pmatrix} f_\tau \\ f_{\tau-1} \\ e_{1\tau} \\ \vdots \\ e_{5\tau} \end{pmatrix} \in \mathbb{R}^9,$$

with dimension $kp + N = 2 \cdot 2 + 5 = 9$. The transition matrix is block diagonal: a companion block for the factor VAR and a diagonal block for the idiosyncratic autoregressions,

$$T = \begin{pmatrix} A_1 & A_2 & 0 \\ I_2 & 0 & 0 \\ 0 & 0 & \operatorname{diag}(\rho_1, \dots, \rho_5) \end{pmatrix},$$

where A_1, A_2 are 2×2. Shocks enter the current factor block and each idiosyncratic state, but not the lagged-factor block:

$$R = \begin{pmatrix} I_2 & 0 \\ 0 & 0 \\ 0 & I_5 \end{pmatrix} \ (9 \times 7), \qquad Q = \begin{pmatrix} \Sigma_u & 0 \\ 0 & \operatorname{diag}(\sigma_1^2, \dots, \sigma_5^2) \end{pmatrix} \ (7 \times 7).$$

The measurement equation reads each series off the current factors and its own idiosyncratic state:

$$Z = \begin{pmatrix} \Lambda & 0_{5\times 2} & I_5 \end{pmatrix} \ (5 \times 9), \qquad H = 0_{5\times 5},$$

with Λ the 5×2 loading matrix. The dimensions:

Object	Dimension
α_τ	9×1
T	9×9
R	9×7
Q	7×7
Z	5×9
H	5×5 (zero)

The zero block in Z is the answer to a frequent confusion: the lagged factors sit in the state to *drive the VAR*, not to be measured; only the current factors carry loadings.

The quarterly GDP row (one-factor version). Model monthly latent GDP growth as $g_\tau = \gamma f_\tau + e_{g,\tau}$, with $e_{g,\tau}$ GDP's own idiosyncratic component. The Session 1 aggregation result maps monthly growth into quarter-on-quarter growth with the tent weights, so the GDP measurement row is

$$y^Q_\tau = \tfrac{1}{3}g_\tau + \tfrac{2}{3}g_{\tau-1} + g_{\tau-2} + \tfrac{2}{3}g_{\tau-3} + \tfrac{1}{3}g_{\tau-4},$$

observed only in quarter-ending months and, within a vintage, only after the GDP release date. Substituting $g_{\tau-j} = \gamma f_{\tau-j} + e_{g,\tau-j}$, the row needs the current factor and its **four lags**, and GDP's idiosyncratic component and its **four lags**: ten states for the GDP block in the one-factor case (or, equivalently, five lags of the composite g_τ if monthly GDP growth itself is carried as a state). The factor's companion block must therefore be extended to five lags even if the factor VAR needs fewer, exactly as the AR example in Session 1 kept lags for the measurement equation's benefit rather than the dynamics'.

The check worth performing on any such construction: multiply out $Z\alpha_\tau$ symbolically and confirm it reproduces the intended measurement equation for every row, then confirm that no state needed by any measurement row is dropped by the transition. Missing lag states fail silently (the filter runs, and the nowcast is simply wrong), which is why this fluency exercise is tagged core.

Exercise 7: The two-step upgrade

Problem. [core, computational] Implement the two-step estimator on a simulated ragged panel: generate $n = 200$ periods of a one-factor panel with $N = 4$ series, loadings $(0.9, 0.8, 0.6, 0.3)$, factor AR(1) coefficient

0.9, and a ragged edge (the last three observations of series 1 and the last observation of series 2 missing). Step 1: principal components and regressions on the balanced rows. Step 2: Kalman filter and smoother over the full panel with parameters fixed. Add assertions. Where does the smoothed factor differ most from the PCA factor, and which estimate tracks the true factor better?

Hint

Treat the idiosyncratic components as measurement noise, so the state is the scalar factor: $T = \hat{a}$, $Z = \hat{\Lambda}$, $H = \text{diag}(\hat{\sigma}_i^2)$, $Q = 1 - \hat{a}^2$. At each period, use only the rows observed that period, exactly as in Session 1's missing-data algorithm.

Detailed solution

The simplest honest two-step keeps only the factor in the state and treats idiosyncratic components as (white) measurement noise, which is adequate here because the simulated idiosyncratics are white. Step 1 estimates $\hat{\Lambda}$ from PCA on the balanced rows, $\hat{\sigma}_i^2 = 1 - \hat{\lambda}_i^2$, and $\hat{a}$ from the factor's first autocorrelation. Step 2 is Session 1's filter with those parameters fixed.

```
import numpy as np
rng = np.random.default_rng(41)

# simulate a one-factor panel with a ragged edge
n, N = 200, 4
a_true = 0.9
lam_true = np.array([0.9, 0.8, 0.6, 0.3])
sig_e = np.sqrt(1 - lam_true**2)

f = np.zeros(n)
f[0] = rng.normal()
for t in range(1, n):
    f[t] = a_true * f[t-1] + rng.normal(scale=np.sqrt(1 - a_true**2))
x = lam_true * f[:, None] + rng.normal(scale=sig_e, size=(n, N))
x[-3:, 0] = np.nan          # series 1 reports with a three-week lag
x[-1:, 1] = np.nan          # series 2 with a one-week lag

# ---- step 1: PCA and regressions on the balanced rows
balanced = ~np.isnan(x).any(axis=1)
Xb = x[balanced]
Xb_std = (Xb - Xb.mean(0)) / Xb.std(0)
U, S, Vt = np.linalg.svd(Xb_std, full_matrices=False)
w = Vt[0]
f_pca = Xb_std @ w
if np.corrcoef(f_pca, Xb_std[:, 0])[0, 1] < 0:
    f_pca, w = -f_pca, -w
f_pca /= f_pca.std()
lam_hat  = np.array([np.cov(Xb_std[:, i], f_pca)[0, 1] for i in range(N)])
sig2_hat = np.clip(1 - lam_hat**2, 0.02, None)
a_hat    = np.corrcoef(f_pca[1:], f_pca[:-1])[0, 1]
q_hat    = 1 - a_hat**2

# ---- step 2: Kalman filter + RTS smoother over the full ragged panel
xs = (x - Xb.mean(0)) / Xb.std(0)      # calibration-window standardization
a_pred, P_pred = 0.0, 1.0
filt, pred = np.empty((n, 2)), np.empty((n, 2))
```

With the seed shown, the run produces: estimated loadings $(0.89, 0.87, 0.76, 0.28)$ against the true $(0.9, 0.8, 0.6, 0.3)$; $\hat{a} = 0.73$ against the true 0.9 (a downward bias worth noticing: the PCA factor path contains extraction noise, which attenuates its measured autocorrelation); and three results that carry the lesson:

1. **On balanced rows the two factors nearly coincide**, at correlation 0.98. Where data are complete, the filter has little to add to the cross-sectional average.
2. **The differences concentrate exactly where data are thin.** The filtered variance is 0.099 mid-sample with all four series reporting, 0.16 when series 1 drops out, and 0.31 in the final week when series 2 is also missing. PCA produces nothing at all for those weeks; the filter produces an estimate *and a widening band*.
3. **Against the true factor, the smoothed estimate wins**: correlation 0.94 versus 0.91 for PCA. The margin is the value of the dynamics: the smoother borrows strength from adjacent periods, which the static estimator cannot.

A common implementation mistake is to advance the state between two releases that belong to the same reference period, or (the two-step-specific version) to re-standardize the panel using full-sample moments in step 2 after calibrating in step 1. Both quietly change the model between the steps. The parameters, the standardization, and the reference-period bookkeeping must all be frozen when the filter takes over.

Exercise 8: Who gets credit for the news?

Problem. [core, pencil] A one-factor nowcast has prior $f \sim \mathcal{N}(0,\, 0.5)$ at the start of a week. Two releases arrive: claims, with loading 0.8 and idiosyncratic variance 0.36, surprises at $v_1 = -1.0$; credit, with loading 0.3 and idiosyncratic variance 0.91, surprises at $v_2 = +0.5$ (both measured against the prior). Compute the factor revision and its release-by-release attribution three ways: jointly, sequentially claims-first, and sequentially credit-first. Verify all three give the same final estimate, and explain why the middle numbers differ.

Hint

For the joint route, the weight vector is $w' = P_0\Lambda'(P_0\Lambda\Lambda' + \Psi)^{-1}$ with $P_0 = 0.5$. For the sequential routes, each scalar update is a Session 1 release update; the posterior from the first becomes the prior for the second, and the second innovation must be recomputed against the updated prior.

Detailed solution

Displayed values are rounded to four decimals; every calculation uses full precision.

Joint attribution. With $\Lambda = (0.8,\, 0.3)'$, $\Psi = \mathrm{diag}(0.36,\, 0.91)$, and $P_0 = 0.5$, the observation covariance is

$$\mathrm{Var}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = P_0 \Lambda\Lambda' + \Psi = \begin{pmatrix} 0.68 & 0.12 \\ 0.12 & 0.955 \end{pmatrix},$$

and the weight vector is

$$w' = P_0 \Lambda' \left(P_0 \Lambda\Lambda' + \Psi\right)^{-1} = (0.5732,\ 0.0850).$$

The attribution is

Release	Surprise v_i	Weight w_i	Contribution $w_i v_i$
Claims	-1.0	0.5732	-0.5732
Credit	$+0.5$	0.0850	$+0.0425$
Total revision			**-0.5307**

Sequential, claims first. The claims update is a scalar Session 1 update: gain $P_0\lambda_1/(P_0\lambda_1^2 + \sigma_1^2) = 0.4/0.68 = 0.5882$, revision $0.5882 \times (-1.0) = -0.5882$, posterior variance $(1 - 0.5882 \times 0.8) \times 0.5 = 0.2647$. Credit's innovation must now be recomputed against the updated prior: $v_2' = 0.5 - 0.3 \times (-0.5882) = 0.6765$. Its gain is $0.2647 \times 0.3/(0.09 \times 0.2647 + 0.91) = 0.0850$, contributing $+0.0575$. Final estimate: $-0.5882 + 0.0575 = -0.5307$.

Sequential, credit first. Credit's gain against the original prior is $0.15/0.955 = 0.1571$, contributing $+0.0785$. Claims' recomputed innovation is $-1.0 - 0.8 \times 0.0785 = -1.0628$; its update contributes -0.6092. Final estimate: $0.0785 - 0.6092 = -0.5307$.

Reconciliation. All three routes end at -0.5307: in a linear Gaussian model with fixed parameters, the posterior cannot depend on the order in which the same information is processed. But credit's *credited contribution* is $+0.0425$ jointly, $+0.0575$ when processed second, and $+0.0785$ when processed first. The reason is shared information: the two releases both carry factor news, and whichever is processed first temporarily receives credit for the overlap. The joint attribution defines one common baseline, the pre-release information set, and splits the revision simultaneously, which is why a production news table should be computed jointly for releases arriving together rather than in an arbitrary processing order.

The magnitudes are also worth a sentence. Claims earns nearly seven times credit's weight, for exactly the reasons the chapter's news section names: its loading is larger (more factor content per unit of surprise) and its idiosyncratic variance smaller (a claims surprise is unlikely to be private noise). A positive credit surprise of respectable size moves this nowcast by four basis points of a standard deviation; the model has, correctly, almost no use for it.

Exercise 9: Choosing the number of factors

Problem. [data] Implement the Bai and Ng (2002) criterion IC_{p2} and validate it on a simulated panel with a known number of factors before trusting it on real data. Then download the FRED-MD monthly panel (McCracken and Ng, 2016), apply the published transformation codes, standardize, and report the

chosen number of factors, the variance shares of the leading components, and an interpretation of factor 2's largest loadings. [extra] Repeat the criterion on the pre-2020 subsample and compare.

Hint

For a standardized $n \times N$ panel, the mean squared residual after k factors is computable from the eigenvalues alone: $V(k) = \sum_{j>k} \hat{\mu}_j / N$. The criterion is $\ln V(k) + k \frac{N+n}{Nn} \ln \min(N, n)$. Validate on simulated data with $k = 3$ true factors first; if your implementation cannot recover a known answer, it has nothing to say about an unknown one.

Detailed solution

The criterion. For a standardized panel Z $(n \times N)$ with sample covariance eigenvalues $\hat{\mu}_1 \geq \hat{\mu}_2 \geq \cdots$, the sum of squared residuals after removing k principal components is the sum of the discarded eigenvalues, so

$$IC_{p2}(k) = \ln\left(\frac{1}{N}\sum_{j>k}\hat{\mu}_j\right) + k\,\frac{N+n}{Nn}\,\ln(\min(N,n)),$$

and $\hat{k}$ minimizes it over $k = 1, \ldots, k_{\max}$. The penalty term is the point of the formula: it shrinks at exactly the rate at which spurious fit accumulates in the $N, n \to \infty$ double limit, which fixed-penalty criteria like AIC and BIC do not match.

Validation. The implementation, checked on a panel where the answer is known:

```python
import numpy as np


def ic_p2(Z, kmax=15):
    n, N = Z.shape
    mu = np.linalg.svd(Z, compute_uv=False)**2 / n
    ics = [np.log(mu[k:].sum() / N)
           + k * (N + n) / (N * n) * np.log(min(N, n))
           for k in range(1, kmax + 1)]
    return int(np.argmin(ics)) + 1, np.array(ics)


rng = np.random.default_rng(7)
n, N, k_true = 480, 120, 3
F   = rng.normal(size=(n, k_true))
Lam = rng.normal(size=(N, k_true))
X   = F @ Lam.T + rng.normal(size=(n, N))
Z   = (X - X.mean(0)) / X.std(0)

k_hat, ics = ic_p2(Z)
assert k_hat == k_true
```

With the seed shown, the criterion selects $\hat{k} = 3$ exactly, and the IC sequence falls steeply through $k = 3$ and then rises, the signature of a well-separated factor structure.

FRED-MD. Download the current vintage of `current.csv` from the FRED-MD page, apply the transformation code in each column's first row (log differences for most real series, second differences of logs for most prices), drop series with long missing stretches and rows with any remaining missingness, standardize, and run the same function. Report four things: the chosen $\hat{k}$; the variance share of each leading component; the date of the vintage you used (FRED-MD is revised monthly, and $\hat{k}$ genuinely moves across vintages and sample choices; published applications of these criteria to FRED-MD typically select on the order of seven or eight factors); and the largest loadings of factor 2.

The interpretation step is where the exercise earns its place. In most vintages, factor 1 loads broadly on real activity (production, employment, income) while factor 2's largest loadings concentrate in a recognizable block, commonly interest rates and spreads or the price series. Write the one-sentence

interpretation factor 2's loadings support, and then note the qualification Exercise 4 taught: if $\hat{\mu}_2$ and $\hat{\mu}_3$ are close, the individual identities of factors 2 and 3 are fragile even when the pair is well identified.

[extra] The pre-2020 comparison. Rerunning the criterion on data through 2019 typically changes the count. The pandemic months are so extreme that they can dominate several eigenvalues on the full sample: the standardization trap in eigenvalue form. Whichever count you obtain, the deliverable is the comparison and its explanation, not a single "right" number: the exercise's point is that the number of factors is an estimate with a sample attached, not a constant of nature.

Exercise 10: WEI archaeology

Problem. [data] The published Weekly Economic Index confronted its own version of the standardization trap in 2020. Using Lewis et al. (2022), Lewis et al. (2021), and the Dallas Fed's WEI documentation, identify what the pandemic did to the index's estimated relationships and what its maintainers changed in response. Relate each change to the calibration-window logic of the P1 build. Then propose, in one page, the v2 design for the four-series P1 tracker that best imports the lessons.

Hint

Follow the two estimated objects separately: the factor weights and the GDP-scaling regression. For each, ask what happens mechanically when observations fifty standard deviations from normal enter its estimation sample, and what "freezing" that object would have meant.

Detailed solution

What the record shows. Three findings, each traceable to the cited sources:

1. *The estimated objects.* The WEI is a first principal component of ten transformed weekly series, scaled by a regression onto four-quarter GDP growth (Lewis et al., 2022). Both the weights and the scaling are estimated on historical samples: precisely the two objects the P1 build learned to calibrate deliberately.
2. *What 2020 did.* The pandemic's movements were tens of standard deviations on the pre-2020 yardstick. Updating the scaling regression with such observations changes the fitted slope materially; the documented discrepancies in the index's depth at the April 2020 trough across versions trace to exactly this parameter updating (Lewis et al., 2021). This is the standardization trap operating on a published index: an extreme episode entering an estimation

sample redefines the mapping applied to every other episode.

3. *What the maintainers did.* Alongside the baseline index, the authors published an alternative indicator measuring changes relative to a fixed February 2020 baseline, combined using the *baseline* WEI weights: they froze the calibration rather than letting the pandemic re-estimate it. The production documentation also separates methodology revisions from data revisions so that users can tell which kind of change moved the index. Consult the Dallas Fed's current WEI documentation for the present-day procedure, and record the date you accessed it: the documentation itself is a vintage.

The mapping to P1. The build's choices are the same responses in miniature: standardization moments, factor weights, and the GDP scaling all estimated on a pre-2020 calibration window and then held fixed, so that 2020 is *measured* on a normal-times scale rather than allowed to compress it. What P1 lacks is what the WEI's episode exposes: a stated policy for when the calibration window itself should ever move.

A defensible v2 design. The one-page proposal should commit to, at minimum:

- *A frozen calibration window with a written amendment rule.* For example: parameters are re-estimated only on a fixed schedule, never mid-episode; extreme observations (beyond a stated threshold) are excluded from any future calibration sample or handled with the outlier machinery Session 1 footnoted; every parameter change is published as a methodology revision distinct from data revisions.
- *Breadth before depth.* Exercise 3 quantified why the claims-dominated panel has a hard precision floor; v2 should add sectors (fuel volumes, electricity, rail) rather than more labor or credit series, accepting the data-engineering cost recorded in the P1 journal.
- *Vintage honesty.* Claims are revised for weeks; a v2 that claims real-time validity must reconstruct from archived vintages, as the P1 brief's stretch goal specifies.

A complete answer also names the residual risk that no calibration policy removes: a frozen yardstick measures new episodes on old units, and if the economy's volatility regime genuinely changes, the frozen scaling will be honestly wrong rather than silently wrong. Choosing *which* error to accept is the design decision; Session 6's treatment of regime dependence picks up exactly here.

References

bai, J., & ng, S. (2002). Determining the number of factors in approximate factor models. *Econometrica*, *70*(1), 191–221.

boivin, J., & ng, S. (2006). Are more data always better for factor analysis? *Journal of Econometrics*, *132*(1), 169–194.

lewis, D. J., mertens, K., stock, J. H., & trivedi, M. (2021). High-frequency data and a weekly economic index during the pandemic. *AEA Papers and Proceedings*, *111*, 326–330.

lewis, D. J., mertens, K., stock, J. H., & trivedi, M. (2022). Measuring real activity using a weekly economic index. *Journal of Applied Econometrics*, *37*(4), 667–687.

mccracken, M. W., & ng, S. (2016). FRED-MD: A monthly database for macroeconomic research. *Journal of Business & Economic Statistics*, *34*(4), 574–589.