

Notes on empirical methods

Statistics of time series and cross sectional regressions

1. Time Series Regression (Fama-French).

(a) *Method*: Run and interpret

$$R_t^{ei} = \alpha_i + \beta_i' f_t + \varepsilon_t^i \quad t = 1, 2, \dots, T. \text{ for each } i$$

(b) *Estimates*:

1. $\hat{\alpha}, \hat{\beta}$: OLS TS regression.

$$R_t^{ei} = \alpha_i + \beta_i' f_t + \varepsilon_t^i \quad t = 1, 2, \dots, T \text{ for each } i.$$

2. $\hat{\lambda}$: Mean of the factor,

$$\hat{\lambda} = \frac{1}{T} \sum_{t=1}^T f_t = \bar{f}.$$

(c) *Standard errors*: If ε_t^i are independent over time.

1. OLS standard errors $\hat{\alpha}_i, \hat{\beta}_i$.

2. $\hat{\lambda}$:

$$\sigma(\hat{\lambda}) = \frac{\sigma(f_t)}{\sqrt{T}}$$

(d) *Test* α are jointly zero?

1. Answer: look at

$$\hat{\alpha}' \text{cov}(\hat{\alpha}, \hat{\alpha}')^{-1} \hat{\alpha}.$$

Precise forms,

$$\hat{\alpha}' \text{cov}(\hat{\alpha})^{-1} \hat{\alpha} = T [1 + \bar{f}' \Sigma_f^{-1} \bar{f}]^{-1} \hat{\alpha}' \Sigma^{-1} \hat{\alpha} \sim \chi_N^2$$

$$\frac{T - N - K}{N} [1 + \bar{f}' \Sigma_f^{-1} \bar{f}]^{-1} \hat{\alpha}' \hat{\Sigma}^{-1} \hat{\alpha} \sim F_{N, T-N-K}$$

Intuition. $R_t^{ei} = \alpha_i + \beta_i' f_t + \varepsilon_t^i$ means that $\hat{\alpha}_i \approx \alpha + \frac{1}{T} \sum_{t=1}^T \varepsilon_t^i$ (except for beta fitting). Thus $\text{cov}(\hat{\alpha}) \approx \frac{1}{T} \Sigma$. The other terms correct for beta fitting. As usual χ^2 is asymptotic for any iid distribution, F is finite-sample for normal ε .

2. Cross-sectional regression, two steps

(a) *Procedure*

$$E(R^{ei}) = (\gamma) + \beta_i' \lambda + \alpha_i \quad i = 1, 2, \dots, N$$

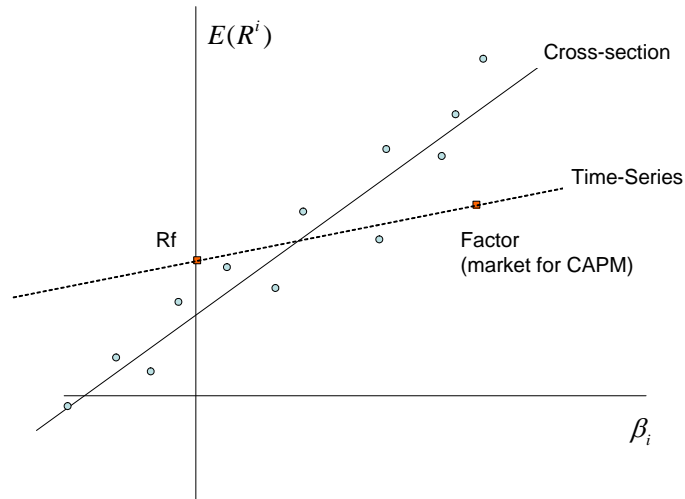
1. TS (over time for each asset) to get β_i ,

$$R_t^{ei} = a_i + \beta_i f_t + \varepsilon_t^i \quad t = 1, 2, \dots, T \text{ for each } i.$$

2. Run XS (across assets) to get λ .

$$E(R^{ei}) = (\gamma) + \beta_i \lambda + \alpha_i \quad i = 1, 2, \dots, N$$

3. TS vs. OLS CS.



(b) *Estimates:*

1. $\hat{\beta}$ from TS.
2. $\hat{\lambda}$ slope coefficient in CS.
3. $\hat{\alpha}$ from error in CS: $\hat{\alpha} = \frac{1}{T} \left(\sum_{t=1}^T R_t^e \right) - \hat{\lambda} \hat{\beta}$. $\hat{\alpha} \neq a$ is not the intercept from the time series regression any more.

(c) *Standard errors.*

1. $\sigma(\hat{\beta})$ from TS, OLS formulas.
2. $\sigma(\hat{\lambda})$. You can't use OLS formulas. Errors $\hat{\alpha}$ are correlated, β are estimated.
Answer: With no intercept in XS,

$$\sigma^2(\hat{\lambda}) = \frac{1}{T} \left[(\beta' \beta)^{-1} \beta' \Sigma \beta (\beta' \beta)^{-1} (1 + \lambda' \Sigma_f^{-1} \lambda) + \Sigma_f \right]$$

3. $cov(\hat{\alpha})$

$$cov(\hat{\alpha}) = \frac{1}{T} (I - \beta(\beta' \beta)^{-1} \beta') \Sigma (I - \beta(\beta' \beta)^{-1} \beta') (1 + \lambda' \Sigma_f^{-1} \lambda)$$

(d) *Test*

$$\hat{\alpha}' cov(\hat{\alpha}, \hat{\alpha}')^{-1} \hat{\alpha} \sim \chi_{N-K-1}^2$$

Warning $cov(\hat{\alpha})$ is singular; use pinv or eigenvalue decompose and only invert the nonzero eigenvalues.

(e) *Formulas with a free intercept*

$$E(R^{ei}) = \gamma + \beta_i \lambda + \alpha_i \quad i = 1, 2, \dots, N$$

$$X = \begin{bmatrix} 1 & | \\ 1 & \beta \\ 1 & | \end{bmatrix}$$

$$\sigma^2 \left(\begin{bmatrix} \hat{\gamma} \\ \hat{\lambda} \end{bmatrix} \right) = \frac{1}{T} \left[(X'X)^{-1} X' \Sigma X (X'X)^{-1} (1 + \lambda' \Sigma_f^{-1} \lambda) + \begin{bmatrix} 0 & 0 \\ 0 & \Sigma_f \end{bmatrix} \right]$$

$$\text{cov}(\hat{\alpha}) = \frac{1}{T} (I - X(X'X)^{-1}X') \Sigma (I - X(X'X)^{-1}X') (1 + \lambda' \Sigma_f^{-1} \lambda)$$

3. GLS cross sectional regression.

(a) Formulas

$$\hat{\lambda} = [\beta' \text{cov}(\alpha)^{-1} \beta]^{-1} \beta' \text{cov}(\alpha)^{-1} E(R^e)$$

$$\hat{\lambda} = [\beta' \Sigma^{-1} \beta]^{-1} \beta' \Sigma^{-1} E(R^e)$$

see *Asset pricing* for $\sigma(\hat{\lambda}), \text{cov}(\hat{\alpha})$

(b) Theorem. If you include f_t and R^f (0) as test assets, then Σ is singular in just the right places and GLS CS = Time series

4. Fama-MacBeth Procedure

(a) Run TS to get betas.

$$R_t^{ei} = a_i + \beta'_i f_t + \varepsilon_t^i \quad t = 1, 2, \dots, T \text{ for each } i.$$

(b) Run a cross sectional regression *at each time period*,

$$R_t^{ei} = (\gamma_t) + \beta'_i \lambda_t + \alpha_{it} \quad i = 1, 2, \dots, N \text{ for each } t.$$

(c) *Estimates* of λ, α are the *averages across time*

$$\hat{\lambda} = \frac{1}{T} \sum_{t=1}^T \hat{\lambda}_t; \quad \hat{\alpha}_i = \frac{1}{T} \sum_{t=1}^T \hat{\alpha}_{it}$$

(d) *Standard errors* use our friend $\sigma^2(\bar{x}) = \sigma^2(x)/T$

$$\sigma^2(\hat{\lambda}) = \frac{1}{T} \text{var}(\hat{\lambda}_t) = \frac{1}{T^2} \sum_{t=1}^T (\hat{\lambda}_t - \hat{\lambda})^2$$

$$\text{cov}(\hat{\alpha}) = \frac{1}{T} \text{cov}(\hat{\alpha}_t) = \frac{1}{T^2} \sum_{t=1}^T (\hat{\alpha}_{it} - \hat{\alpha}_i) (\hat{\alpha}_{jt} - \hat{\alpha}_j)$$

This is one main point. These standard errors are easy to calculate.

(e) *Test*

$$\hat{\alpha}' \text{cov}(\hat{\alpha}, \hat{\alpha}')^{-1} \hat{\alpha} \sim \chi_{N-1}^2.$$

Why don't people do this?

5. GMM. Of *course* you should be doing GMM and not assuming ε_t are iid over time, right?
6. Testing one model vs. another (see longer description below)

(a) Example. FF3F.

$$E(R^{ei}) = \alpha_i + b_i \lambda_{rmrf} + h_i \lambda_{hml} + s_i \lambda_{smb}$$

Drop size?

$$E(R^{ei}) = \alpha_i + b_i \lambda_{rmrf} + h_i \lambda_{hml}?$$

(b) Solution: “Orthogonalized factor”,

$$smb_t = \alpha_{smb} + b_s rmr f_t + h_s hml_t + \varepsilon_t$$

We can drop smb from the three factor model if and only α_{smb} is zero. .

(c) Equivalently, we are forming an “orthogonalized factor”

$$smb_t^* = \alpha_{smb} + \varepsilon_t = smb_t - b_s rmr f_t - h_s hml_t$$

and drop if $E(smb^*) = 0$

(d) This must be equivalent to a proper test whether $\alpha' \alpha$ declines, (*not* comparing GRS statistics!)

Testing whether a factor is redundant.

Smb looks pretty marginal. Can we get by in our explanation of *mean* returns (not necessarily in our understanding of the *return covariance matrix*) with a simpler model that only uses hml and rmrf? (Fama French might be perfectly right that smb is an important factor in covariances, as industry factors are, but exposure to smb might not be important to explain *mean* returns)

Lots of people think that we test for the importance of smb by looking at s_i t statistics, mistaking the time-series regression that measures decomposition of *variance* for the implied cross-sectional relation which measures the extent to which the factor model explains *means*. Lots more people mistakenly think that we test for dropping smb by testing whether $E(smb) = \lambda_{smb} = 0$, “is the factor priced.”

This is wrong however. Why? Suppose $smb_t = \frac{1}{2} rmr f_t + \frac{1}{2} hml_t$. Then, obviously, $E(smb) > 0$, but just as obviously, we can drop *smb* from the right hand side with no harm at all. We need to test whether smb is useful to *price other things*, not whether it *is priced*. Equivalently, when you drop *smb* from the regression the other coefficients change, and they may change just enough to still explain expected returns, even if $E(smb) > 0$.

We can do the right test very simply by running a regression of smb_t on $rmr f_t$ and hml_t .

$$smb_t = \alpha_s + b_s rmr f_t + h_s hml_t + \varepsilon_t$$

If $\alpha_s = 0$, then *smb* is priced by other two factors, and this *is* the test whether we can drop *smb* from the model.

Why? Think about defining an orthogonalized factor $smb_t^* = \alpha_s + \varepsilon_t = smb_t - b_s rmr f_t - h_s hml_t$. Rewriting the original model in terms of smb^* ,

$$\begin{aligned} R_t^{ei} &= \alpha_i + b_i rmr f_t + h_i hml_t + s_i smb_t + \varepsilon_t^i \\ &= \alpha_i + (b_i + s_i b_s) rmr f_t + (h_i + s_i h_s) hml_t + s_i (smb_t - b_s rmr f_t - h_s hml_t) + \varepsilon_t^i \\ &= \alpha_i + (b_i + s_i b_s) rmr f_t + (h_i + s_i h_s) hml_t + s_i smb_t^* + \varepsilon_t^i \end{aligned}$$

and hence

$$E(R^{ei}) = \alpha_i + (b_i + s_i b_s) E(rmr f_t) + (h_i + s_i h_s) E(hml_t) + s_i E(smb_t^*)$$

The other factors would now get the betas that were assigned to *smb* merely because *smb* was correlated with the other factors. This part of the *smb* premium can be captured by the other factors, we don't need *smb* to do it. The only part that we need *smb* for is the last part. And $E(smb^*) = 0$ is exactly the same as $\alpha_s = 0$. (This is equivalent for a test $b_s = 0$ in $m = a - b_m rmr f - b_h hml - b_s smb$, $0 = E(mR^e)$ as advocated in the λ vs. b Chapter 13.4 of *Asset pricing*. This must be equivalent to a test whether $\alpha' \Omega^{-1} \alpha$ has risen for some well defined common Ω , but neither I nor anyone else has worked that out.)

Eigenvalue factor decompositions

Summary of the procedure

Given an $N \times 1$ vector of random variables y , with $cov(y, y') = \Sigma$, we form $Q\Lambda Q' = cov(y)$ by the eigenvalue decomposition. If Y is a $T \times N$ matrix of data on y , then $[Q, L] = eig(cov(y))$ in matlab. Λ is diagonal and Q is orthonormal, $QQ' = Q'Q = I$.

We form "factors" by $x = Q'y$. The columns of Q thus express how to construct factors x from the data on y .

We can then write $y = Qx$, i.e. $y_t = q_1 x_t^{(1)} + q_2 x_t^{(2)} + \dots$ where $q_1 = Q(:, 1)$ denotes the first column of Q . The columns of y thus also give "loadings" that describe how each y moves if one of the factors x moves.

We have $cov(x, x') = Q'Q\Lambda Q'Q = \Lambda$, i.e. the x are uncorrelated with each other.

If some of the diagonals Λ are zero, then we express all movements in y by reference to only a few underlying factors. For example, if only the first Λ is nonzero, then we can express $x = q_1 z_1$. In practice, we often find that many of the diagonals of Λ are very small, so setting them to zero and fitting y with only a few factors leads to an excellent approximation.

Since the factors are uncorrelated, if we ignore some factors, the loadings on the remaining ones are the same as if we ran regressions,

$$\begin{aligned} y_t &= q_1 x_t^{(1)} + q_2 x_t^{(2)} + [q_3 x_t^{(3)}] \\ y_t &= q_1 x_t^{(1)} + q_2 x_t^{(2)} + \varepsilon_t \end{aligned}$$

Derivation

Factors constructed in this way solve in turn the question “what linear combinations of y have maximum variance, subject to the constraint that the sum of squared weights is one and each linear combination is orthogonal to the previous ones?” In equations, each column q_i of Q satisfies

$$\max [var(q'_i y)] \text{ s.t. } q'_i q_i = 1, q'_i q_j = 0, j < i$$

Why is this an interesting question? “What linear combination $q' y_t$ of the y_t has the highest variance?” would be too easy a question. Just make q big. The constraint $q' q = 1$ means the sum of squared q must be equal to one, so you can't boost variance by making q big. You have to find the right *pattern* of q across the elements of y .

Why is this the answer? Let's look at the first maximization – what linear combination of the y has the largest variance, if you constrain the sum of squared weights to be less than one?

$$\max_{\{q\}} var(q' y_t) \text{ s.t. } q' q = 1$$

Forming a Lagrangian,

$$L = q' \Sigma q - \lambda q' q$$

The first order condition ($\partial/\partial q$) is

$$\Sigma q = \lambda q$$

This is an eigenvalue problem! The answer to this question is then, *choose q as an eigenvector of the matrix Σ* . Now, which one? Let's see what variance of x we get out of all this

$$var(x_t) = var(q' y) = q' \Sigma q = q' \lambda q = \lambda q' q \lambda$$

Aha! *The eigenvalue gives us the the variance of x* . So, the answer to our maximization is, *choose the eigenvector q corresponding to the largest eigenvalue λ* .

In sum, to find the linear combination of y with largest variance, and sum of squared weights equal one, we choose as weights q the eigenvector corresponding to the largest eigenvalue of the covariance matrix of y . Q is a matrix of eigenvectors, so you're done. Eigenvectors are orthogonal $q'_i q_j = 0$ so the eigenvectors corresponding to successively smaller eigenvalues answer the question for the remaining factors.

Now, what does this have to do with R^2 ? Suppose we leave out some factors

$$y_t = q_1 x_t^{(1)} + q_2 x_t^{(2)} + [q_3 x_t^{(3)}]$$

Since $x^{(1)}$ and $x^{(2)}$ have *maximum* variance, this means $x^{(3)}$ and beyond have *minimum* variance. In short *we have found a factor model that for each choice of how many factors to use maximizes the R^2 in these regressions*.

Rotation

There are lots of equivalent ways to write any factor model. If we have

$$y_t = Q x_t$$

then if R is any orthogonal (rotation) matrix, i.e. any matrix with $RR' = R'R = I$, we can define new factors $z_t = R'x_t$. Then $x_t = Rz_t$

$$y_t = Q(Rz_t) = (QR)z_t$$

Now the columns of QR give us new loadings on the new factors, which are constructed by $z_t = R'x_t = R'Q'y_t = (QR)'\ y_t$

This works even better if we use unit variance shocks.

$$\begin{aligned} y_t &= \left(Q\Lambda^{\frac{1}{2}}\right) \left(\Lambda^{-\frac{1}{2}}x_t\right) = \left(Q\Lambda^{\frac{1}{2}}\right) z_t \\ \text{cov}(z_t, z_t') &= \Lambda^{-\frac{1}{2}}\Lambda\Lambda^{-\frac{1}{2}} = I \end{aligned}$$

Now if we rotate,

$$w_t = Rz_t, z_t = R'w_t$$

we have

$$\begin{aligned} y_t &= \left(Q\Lambda^{\frac{1}{2}}\right) z_t = \left(Q\Lambda^{\frac{1}{2}}\right) R'w_t \\ \text{cov}(w_t, w_t') &= \text{cov}(Rz_t, z_t'R') = RR' = I \end{aligned}$$

In sum, *if we rotate unit-variance factors, they are still uncorrelated with each other and still have unit variance.* You're free to recombine factors in any way you want to make them look pretty.