Two Thoughts About Cointegration and the SVD

Bryan Lewis

blewis@paradigm4.com

http://SciDB.org
http://paradigm4.com

R/Finance 2012

Cointegration

Occurs when a linear combination of I(1) variables is I(0), and also in other more general cases.

Cointegration

Occurs when a linear combination of I(1) variables is I(0), and also in other more general cases.

- A drunk walking his dog.

Cointegration

Occurs when a linear combination of I(1) variables is I(0), and also in other more general cases.

- A drunk walking his dog.
- Any group hanging out with Thomas on a given night.


Use R!
Bernhard Pfaff
Analysis of Integrated and Cointegrated Time Series with R
Second Edition
Springer

Computing and Testing Cointegration

VECM methods boil down to modeling variables x over time $t = 1, 2, \ldots, T$ like

$$\Delta x_t = \Pi x_{t-1} + \epsilon_t,$$

with the cointegration hypothesis that $\text{rank}(\Pi) \leq r$.

Computing and Testing Cointegration

Writing $\Pi = \alpha\beta^T$, $\alpha, \beta \in \mathbf{R}^{n\times r}$, the equations are often arranged in matrix form as a linear system:

$$X_0 = X_1\beta\alpha^T + E,$$

where X_j arise from vectors x_t.

The Johansen method estimates

$$\min_{\beta\alpha^T} \|X_0 - X_1\beta\alpha^T\|$$

by maximum likelihood.

The Singular Value Decomposition (SVD)

Let $A \in \mathbf{R}^{m \times n}, m \geq n$. The SVD of A is:

$$\begin{aligned} AV &= U\Sigma, \\ V^T V &= I = U^T U, \\ \Sigma &= \text{diag}(\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n \geq 0), \end{aligned}$$

where $U \in \mathbf{R}^{m \times n}, \Sigma \in \mathbf{R}^{n \times n}, V \in \mathbf{R}^{n \times n}$ *(thin version)*.

The Singular Value Decomposition

Note that the SVD:

- Always exists.
- Is (almost always) the most numerically-stable way to compute many things with matrices.
- Is a model-free way to consider data.

But, it's somewhat computationally expensive to compute.

SVD and Cointegration (Doornik, O'Brien[1])

Solve $\min_{\beta\alpha^T} \|X_0 - X_1\beta\alpha^T\|$ by SVD(s) with:

1. Compute $X_j V_j = U_j \Sigma_j$,
2. Let $Z := U_1^T U_0$,
3. Compute $Z V_z = U_z \Sigma_z$.

They show that $\beta = T^{1/2} V_1 \Sigma^{\dagger} U_z$, and $\alpha = T^{1/2} V_0 \Sigma_0 Z^T U_z$.

[1] Doornik, J.A. and O'Brien, R.J. (2002). Numerically Stable Cointegration Analysis, Computational Statistics and Data Analysis, 41, 185-193.

What about Large Data?

Doornik's approach can reliably detect numerical rank of large matrices in the presence of ill-conditioned data. But,

1. We're generally not interested in trading huge sets.
2. Statistical interpretation difficult.

SVD Again

One approach builds a candidate set for cointegration by iterating:

1. Use SVD subset selection[2] to pick a few variables.
2. Project remaining variables into subspace defined by the selected set and choose among those with smallest norm.

That is if K is the set of selected columns in a matrix A, compute $A_K V_K = U_K \Sigma_K$ and choose additional columns j where $\|U_K^T a_j\|$ is small.

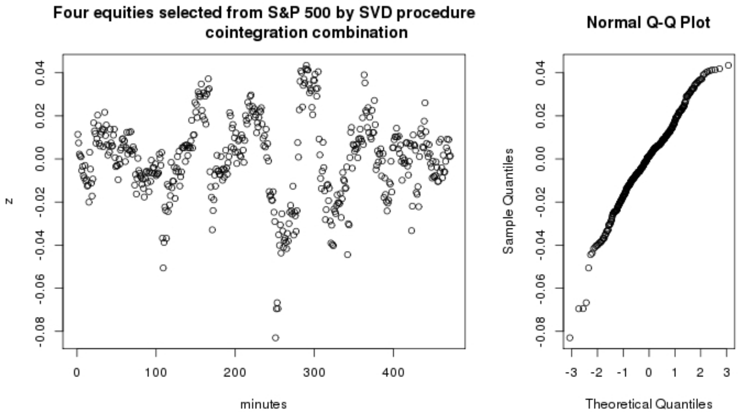
[2] Gene Golub and Charles Van Loan, Matrix Computations, Johns Hopkins University Press, 1996.

SVD Subset Selection, a Real Gem

Listing 1: SVD Subset Selection.

```
# Input matrix A
# Number of singular values n
# Number of output columns k<=n
# Returns an index subset of columns of A that *estimate*
    the k most
# linearly independent columns.
svdsubsel <- function(A,n,k=n)
{
 S <- svd(A, n)
 Q <- qr( t(S$v[,1:n]) ,LAPACK=TRUE)
 Q$pivot[1:k]
}
```

Example SVD Subset Selection



Performance

Wait, isn't it expensive to compute all these SVDs?

No! Note that many discussed computations require only partial SVDs.

Use the IRLB algorithm[3].

- IRLB is perhaps the most efficient method to compute a few singular vectors of matrix.
- IRLB can find vectors associated with largest *or* smallest singular vectors.

[3] http://cran.r-project.org/web/packages/irlba/index.html

Summary

I advocate following Doornik's advice and using the SVD to solve the cointegration problem in a numerically stable way.

Use the SVD to mine data for likely cointegrated sets.