



On estimating covariances between many assets with histories of highly varying length

Robert B. Gramacy

Statistical Laboratory

University of Cambridge

bobby@statslab.cam.ac.uk

Joo Hee Lee

Fidelity Investments

London

joohee.lee@uk.fid-intl.com

INQUIRE Europe/UK Spring Seminar

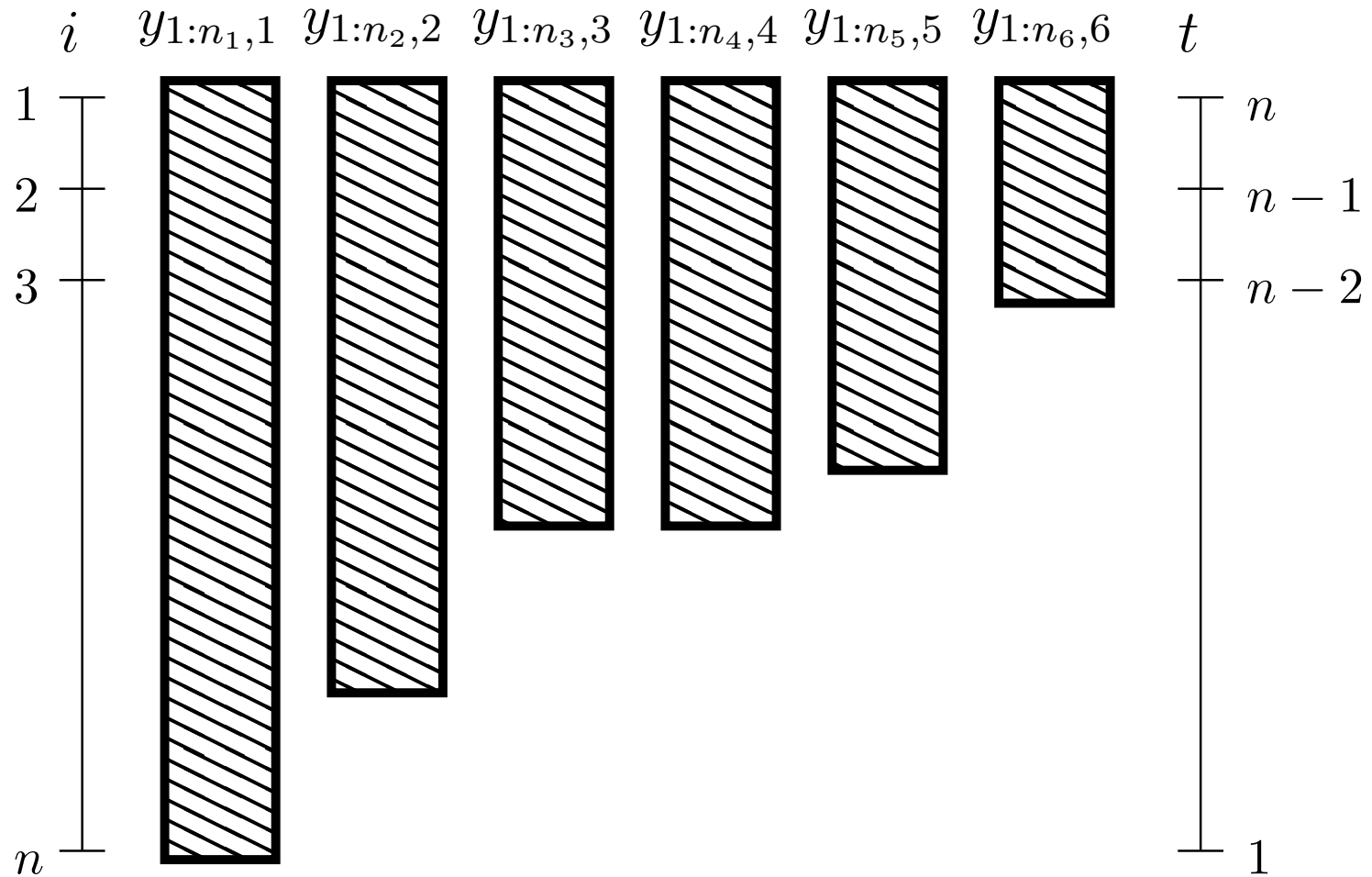
1 April 2008, Zürich Switzerland

Asset return histories vary in length

- ❑ they begin being publicly traded at different times
- ❑ they can close for various reasons
 - M&A, bankruptcy, etc.
- ❑ this can make estimating covariances challenging
- ❑ but we can treat this as a *missing data* problem
 - focus on historical missingness



Missingness pattern is *monotone*



Y: $y_{:,1}, \dots, y_{:,m}$ **and let** $y_j \equiv y_{1:n_j,j}$



Missingness pattern is *monotone*

- $y_{i,j} = \text{NA}$ if the i^{th} return of the j^{th} asset is missing
- a missingness pattern is *monotone* (Little & Rubin, 2002) if $y_{i,j} \neq \text{NA}$ whenever $y_{i,j+1} \neq \text{NA}$
- assume *missing completely at random* (MCAR)
 - missingness pattern neither depends on observed nor unobserved returns
- then the likelihood has convenient factorisation:

$$f(\mathbf{Y}|\boldsymbol{\theta}) = f(\mathbf{y}_1|\boldsymbol{\theta}_1)f(\mathbf{y}_2|\mathbf{y}_1, \boldsymbol{\theta}_2) \cdots f(\mathbf{y}_m|\mathbf{y}_1, \dots, \mathbf{y}_{m-1}, \boldsymbol{\theta}_m)$$



Easy to get MLE under MVN assumption

□ maximum likelihood estimators (MLE) of

$$\theta_j = (\mu_j, \Sigma_{1:j,j}) \quad \text{for} \quad j = 2, \dots, m$$

obtained by ordinary least squares (OLS) regression

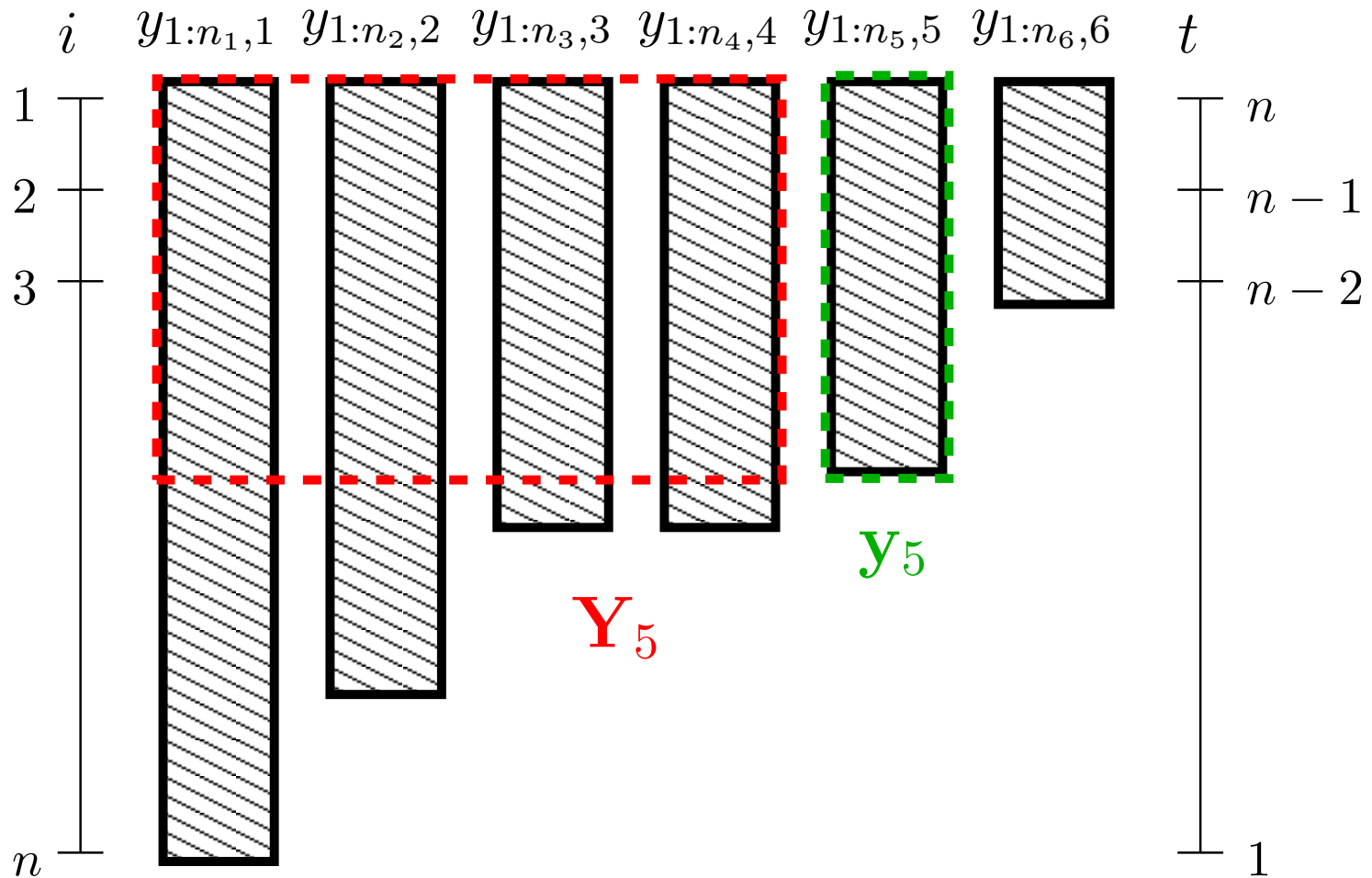
$$y_j = \mathbf{Y}_j \beta_j + \epsilon_j, \quad \{\epsilon_{i,j}\}_{i=1}^{n_j} \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma_j^2)$$

where $y_j \equiv y_{1:n_j,j}$ and

$$\mathbf{Y}_j \equiv \mathbf{Y}_{0:(j-1)}^{(n_j)} = \begin{pmatrix} 1 & y_{1,1} & \cdots & y_{1,(j-1)} \\ 1 & y_{2,1} & \cdots & y_{2,(j-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & y_{n_j,1} & \cdots & y_{n_j,(j-1)} \end{pmatrix}$$



Repeated OLS regressions



$$y_j = Y_j \beta_j + \epsilon_j$$



MLE for OLS obtained in the usual way

- When $\text{rank}(\mathbf{Y}_j) = j < n_j$, OLS gives the MLE:

$$\hat{\boldsymbol{\beta}}_j = (\mathbf{Y}_j^\top \mathbf{Y}_j)^{-1} \mathbf{Y}_j^\top \mathbf{y}_j \quad \text{and} \quad \hat{\sigma}_j^2 = \frac{1}{n_j} \sum_{i=1}^{n_j} (y_{i,j} - \hat{\boldsymbol{\beta}}_j^\top \mathbf{Y}_{j,i})^2$$

- $\hat{\boldsymbol{\theta}}_1 : \hat{\mu}_1 = \sum_{i=1}^{n_1} y_{i,1} / n_1$ and $\hat{\Sigma}_{1,1} = \sum_{i=1}^{n_1} (y_{i,1} - \hat{\mu}_1)^2 / n_1$

- Obtain $\hat{\boldsymbol{\theta}}_j$ from $\hat{\boldsymbol{\theta}}_{1:(j-1)}$ and $\hat{\boldsymbol{\beta}}_j$ and $\hat{\sigma}_j^2$ as

$$\hat{\mu}_j = \hat{\beta}_{0,j} + \hat{\boldsymbol{\beta}}_{1:(j-1),j}^\top \hat{\boldsymbol{\mu}}_{1:(j-1)}$$

$$\hat{\Sigma}_{1:j,j} = \begin{pmatrix} \hat{\boldsymbol{\beta}}_{1:(j-1),j}^\top \hat{\Sigma}_{1:(j-1),1:(j-1)} \\ \hat{\sigma}_j^2 + \hat{\boldsymbol{\beta}}_{1:(j-1),j}^\top \hat{\Sigma}_{1:(j-1),1:(j-1)} \hat{\boldsymbol{\beta}}_{1:(j-1),j} \end{pmatrix}$$



Example on cement data

Heat (y) evolved in setting of cement, as a function of its chemical composition ($x_{1:4}$) (Little & Rubin, 2002)

original ordering						monotone ordering					
n	x_1	x_2	x_3	x_4	y	n	x_3	y	x_1	x_2	x_4
1	7	26	6	60	78.50	1	6	78.50	7	26	60
2	1	29	15	52	74.30	2	15	74.30	1	29	52
3	11	56	8	20	104.30	3	8	104.30	11	56	20
4	11	31	8	47	87.60	4	8	87.60	11	31	47
5	7	52	6	33	95.90	5	6	95.90	7	52	33
6	11	55	9	22	109.20	6	9	109.20	11	55	22
7	3	71	17		102.70	7	17	102.70	3	71	
8	1	31	22		72.50	8	22	72.50	1	31	
9	2	54	18		93.10	9	18	93.10	2	54	
10			4		115.90	10	4	115.90			
11			23		83.80	11	23	83.80			
12			9		113.30	12	9	113.30			
13			8		109.40	13	8	109.40			



When the method fails

- **When** $\text{rank}(\mathbf{Y}_j) = j \geq n_j$
 - **rank deficient design matrix** \mathbf{Y}
 - **cannot invert** $\mathbf{Y}_j^\top \mathbf{Y}_j$
- **called a “big p , small n ” problem**
 - **more parameters/predictors (p) : cols of $\mathbf{Y}_j = j$**
 - **than observations (n) : rows of $\mathbf{Y}_j = n_j$**
- **Therefore to find the MLE, we cannot have:**
 - **an asset with fewer returns (n_j) than the number of assets ($j - 1$)**
 - **more assets than returns**



Solution: parsimonious regression

Instead of OLS for regressions

$$y = \mathbf{X}\beta + \epsilon, \quad \{\epsilon_i\}_{i=1}^n \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2).$$

where $y \equiv y_j$, $\mathbf{X} \equiv \mathbf{Y}_j$, and $\sigma^2 \equiv \sigma_j^2$, we consider using

□ Shrinkage

■ ridge regression & the lasso

□ Change-of-basis

■ principal components (PCR) & partial least squares regression (PLSR)

to obtain $\hat{\beta}$ and $\hat{\sigma}^2$ without having to invert $\mathbf{X}^\top \mathbf{X}$



Shrinkage: ridge regression & the lasso

Shrink coeffs of OLS by penalising their size:

$$\hat{\beta}^{(q)} = \operatorname{argmin}_{\beta} \left\{ \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j \right)^2 + \lambda \sum_{j=1}^p |\beta_j|^q \right\}$$

□ $q = 2$ for ridge regression

$$\hat{\beta}^{(2)} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$$

□ $q = 1$ for lasso

■ requires quadratic programming to get $\hat{\beta}^{(1)}$

■ $\hat{\beta}^{(1)}$ may have many components shrunk to zero



Change-of-basis: PCR & PLSR

PCR:

□ SVD on \mathbf{X} , i.e., $\mathbf{X} = (\mathbf{U}\mathbf{D})\mathbf{V}^\top = \mathbf{T}\mathbf{P}^\top$

□ regressing \mathbf{y} on the first k PCs, gives:

$$\text{(scores and loadings)} \quad \hat{\boldsymbol{\beta}}(k) = \mathbf{P}_{(k)} (\mathbf{T}_{(k)}^\top \mathbf{T}_{(k)})^{-1} \mathbf{T}_{(k)}^\top \mathbf{y} \quad (1)$$

$$\text{(from SVD on } \mathbf{X}) \quad \hat{\boldsymbol{\beta}}^{\text{pcr}}(k) = \mathbf{V}_{(k)} \mathbf{D}_{(k)}^{-1} \mathbf{U}_{(k)}^\top \mathbf{y},$$

PLSR:

□ SVD on $\mathbf{X}^\top \mathbf{y}$, including info on the correlation between, and the variance within, \mathbf{X} and \mathbf{y}

■ follow (1) to get $\hat{\boldsymbol{\beta}}^{\text{plsr}}$



Choose shrinkage λ , or # of components k

... by minimising **Cross Validation (CV) estimates of predictive error:**

- partition $\{\mathbf{X}, \mathbf{y}\}$ into 10 blocks of (roughly) equal size

$$\{\mathbf{X}, \mathbf{y}\}_1, \dots, \{\mathbf{X}, \mathbf{y}\}_{10}$$

- use $\{\mathbf{X}, \mathbf{y}\}_{(-i)}$ to predict $\{\mathbf{X}, \mathbf{y}\}_i$ for $i = 1, \dots, 10$
 - calculate the predictive error for each λ or k
- pick the best λ or k



Comparing shrinkage and change-of-basis regressions:

- ❑ ridge regression shrinks the coeffs of PCs by a factor of $d_j^2 / (d_j^2 + \lambda)$, whereas PCR truncates them at k
- ❑ if X and y are highly correlated, then PLSR may have an advantage over PCR
- ❑ lasso is part of a larger framework called *least angle regression* (LAR), including other methods like
 - stepwise and forward stagewise
 - all possible λ can be considered in time proportional to one full OLS regression



Returning to the `monomvn` algorithm

- Initialise μ_1 and Σ_{11} to the sample mean and variance of the first column y_1 of Y
- repeat for $j = 2, \dots, m$:
 - Find the MLEs of β_j and σ_j^2 in a regression of y_j onto the first n_j rows $j - 1$ columns of Y
 - if ever $n_j \leq j$ use a parsimonious regression, otherwise use OLS
 - Obtain the MLEs of μ_j and $\Sigma_{(1:j),j}$
 - from $\hat{\mu}_{1:(j-1)}$, $\hat{\Sigma}_{1:(j-1),1:(j-1)}$, $\hat{\beta}_j$ and $\hat{\sigma}_j^2$



Unnecessary parsimonious regressions

Parsimonious regressions can

- improve accuracy & yield lower variance estimators
 - so-called bias/variance tradeoff
- aid in interpretation
 - lasso et al. enable the detection of zeros in Σ

Determine a threshold p :

- force a parsimonious regression when $n_j \leq pj$
 - $p = 0$: always use a parsimonious method
 - $p = 1$: only when necessary



Implementation

`monomvn` is made freely available as an R package

`www.r-project.org`

and depends on

- `lars` package (Hastie & Efron, 2007)
- `pls` package (Mevik & Wehrens, 2007)
- `lm.ridge` from the built-in MASS library

Within R do:

```
R> install.packages(c("monomvn", "lars", "pls")) # (once)
R> library(monomvn)
```



Comparisons and empirical results

□ monomvn validated on real and synthetic data with

■ Kullback–Leibler (KL) divergence

$$D_{\text{KL}}(MVN(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}) \parallel MVN(\boldsymbol{\mu}, \boldsymbol{\Sigma}))$$

$$= \frac{1}{2} \left(\log \frac{|\boldsymbol{\Sigma}|}{|\hat{\boldsymbol{\Sigma}}|} + \text{tr}(\boldsymbol{\Sigma}^{-1} \hat{\boldsymbol{\Sigma}}) + (\hat{\boldsymbol{\mu}} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\hat{\boldsymbol{\mu}} - \boldsymbol{\mu}) \right)$$

■ Root mean squared error (RMSE)

$$\text{RMSE}(\hat{\boldsymbol{\mu}}, \boldsymbol{\mu})$$

$$= \sqrt{\frac{1}{m} \sum_{j=1}^m (\hat{\mu}_j - \mu_j)^2}$$

$$\text{RMSE}(\hat{\boldsymbol{\Sigma}}, \boldsymbol{\Sigma})$$

$$= \sqrt{\frac{1}{m^2} \sum_{i,j=1}^m (\hat{\Sigma}_{i,j} - \Sigma_{i,j})^2}$$



Comparators

Two simple estimators of μ and Σ :

- “complete”: use only the completely observed cases
 - i.e., the first n_m rows of Y
- “observed”: use all data in a naïve way

$$\hat{\mu}_j = \frac{1}{n_j} \sum_{k=1}^{n_j} y_{k,j} \quad \text{and} \quad \hat{\Sigma}_{i,j} = \frac{1}{n_j} \sum_{k=1}^{n_j} (y_{k,j} - \hat{\mu}_j)(y_{k,i} - \hat{\mu}_i)$$

One state of the art:

- `norm` in R and `ecmmle` for Matlab
 - use Expectation Conditional Maximisation (ECM) for arbitrary missingness patterns



Synthetic Data

random MVN data with uniform monotone missingness

method	KL div		RMSE μ		RMSE Σ	
	mean	var	mean	var	mean	var
plsr	53.0	2125	0.037	0.00050	0.052	0.0043
pcr	69.0	3249	0.038	0.00054	0.055	0.0049
ridge	45.4	837	0.035	0.00035	0.049	0.0038
lasso	101.9	65966	0.039	0.00043	0.066	0.0075
lar	134.2	125789	0.040	0.00048	0.079	0.0130
fwdstag	104.9	84585	0.039	0.00043	0.066	0.0081
step	258.9	625306	0.041	0.00044	0.096	0.0298
observed			0.067	0.00121	0.099	0.0081
complete			0.289	0.03751	0.302	0.0799

□ $p = 1$: parsimonious regression only when necessary



Determining the parsimonious proportion p

□ mean and 90% interval

method	optimal p			improv	
	5%	mean	95%	$p = 0$	$p = 0.5$
plsr	0.13	0.38	0.62	0.90	0.94
pcr	0.15	0.53	0.84	0.52	0.75
ridge	0.02	0.22	0.62	0.90	0.96
lasso	0.03	0.38	0.76	0.71	0.81
lar	0.06	0.44	0.78	0.63	0.65
stepwise	0.36	0.64	0.91	0.25	0.50

□ $p = 0.5$ is a good rule of thumb

□ larger p may be preferred for speed reasons



Comparing to ECM

- iterates until convergence, stops at *local* max
 - can fail to converge/numerical singularities
- $m = 100$; 100 **repeated trials**; uniform monotone

KL-div	monomvn			norm (ECM)		
	mean	var	95%	mean	var	95%
$n = 100$	3.9	27	12.78	2.4×10^{21}	4.1×10^{45}	32.8
$n = 1000$	0.5	1.8		0.62	2.6	

- **for monomvn with ridge regression and $p = 0.5$**



Real Data

From Thomson Financial's Datastream

- ❑ total returns data of each stock in the Russell 3000[®]
 - 1792 weekly returns 1973 – 2007 for 2461 assets
 - 558 have the longest history of 1792
 - shortest history has only 76 returns
 - 47% missing returns overall
- ❑ First experiment involves a complete subset
 - 635 most recent returns for the 617 most observed assets
 - ❑ gives complete–data estimator μ and Σ



Real Data: complete subset

Then uniformly interject monotone missingness

method	KL div		RMSE μ		RMSE Σ	
	mean	var	mean	var	mean	var
plsr	1.29e+04	7.04e+06	3.42e-03	4.36e-07	5.65e-04	2.59e-07
pcr	1.37e+04	8.22e+06	3.41e-03	3.11e-07	5.52e-04	2.68e-07
ridge	9.97e+03	4.75e+06	3.41e-03	6.02e-07	5.55e-04	2.51e-07
lasso	1.41e+04	1.29e+07	3.48e-03	1.74e-07	9.43e-04	1.24e-06
lar	1.43e+04	1.31e+07	3.50e-03	1.57e-07	9.45e-04	1.24e-06
step	1.54e+04	1.00e+07	3.50e-03	1.33e-07	7.15e-04	2.17e-07
obs			3.46e-03	1.40e-07	4.22e-04	8.30e-09
complete			1.64e-02	3.12e-06	8.35e-04	2.65e-08

$p = 1$: parsimonious regression only when necessary

ECM fails to converge on this data



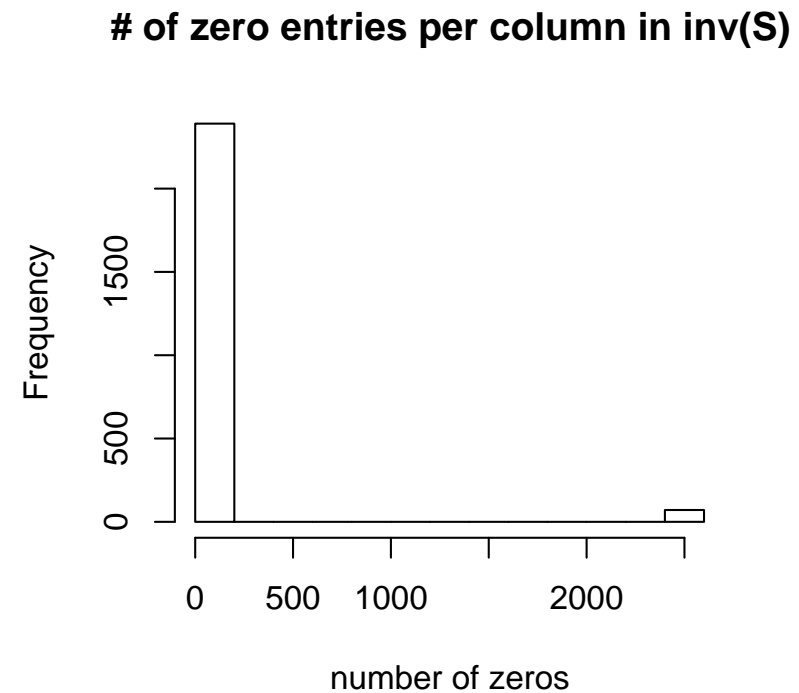
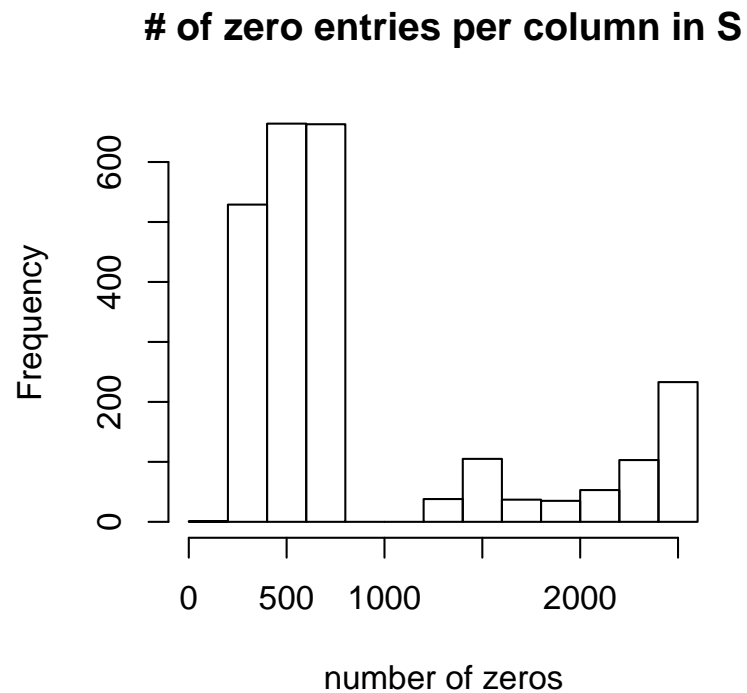
Real data: returning to the full set

- ❑ “complete” estimator uses only 3% of the data
 - shortest history has only 76 returns
- ❑ apply the lasso version of `monomvn` with $p = 0$
 - enables the detection of zeros in Σ
 - ❑ indicating marginally uncorrelated assets
 - ❑ and conditionally uncorrelated assets in Σ^{-1}
 - similarly with the other LARS methods
- ❑ ridge regression, PCR, PLSR do not offer this feature



Finding independent assets

monomvn with the lasso finds



□ 36% of $\hat{\Sigma}$ is zero; 6% of $\hat{\Sigma}^{-1}$ is zero

■ 2% of $\hat{\Sigma}$ are everywhere zero except on diagonal



Removing the market

We downloaded market returns (Russel 3000) for 1479 (of 1792) contiguous weeks ending 11/5/2007

- created a residual return series for all 2461 assets**
 - and re-ran the lasso experiment to discover that**
 - 58% of asset pairings are marginally uncorrelated**
 - 14% are conditionally independent**
- when the market is taken into account**

(with similar looking histograms)



Discussion and future work

- extended (Stambaugh, 1996) to many assets
 - from 22 indices to 2461 equities
- even when OLS suffices, the parsimonious approach has merits
- bootstrap can be used to determine the stability of $\hat{\Sigma}$
 - alternative Bayesian approach is in the works
- relaxing MVN assumption is also in the works
 - i.e., using *copulas*

