Linear Algebra and Portfolio Constructions

Goals:

- Review some linear algebra and matrix decomposition
- Use the results for portfolio construction

Relevant literature:

- Numerical Recipes: The Art of Scientific Computing; Brian P. Flannery, Saul Teukolsky, William H. Press, and William T. Vetterling
- Ledoit & Wolf: Nonlinear Shrinkage Estimation of Large-Dimensional Covariance Matrices. www.ledoit.net/AOS989.pdf

Diagonalization

Recall that square $k \times k$ matrix M is diagonalizable if there is a matrix U and a diagonal matrix D so that

$$M = UDU^{-1} \tag{1}$$

The elements of D are called eigenvalues of M and the columns of U are the corresponding eigenvectors.

Formula (1) allows us to compute high powers of M very cheaply, since

$$M^2 = \left(UDU^{-1}\right)\left(UDU^{-1}\right) = UDU^{-1}UDU^{-1} = UD^2U^{-1} \tag{2}$$

and, similarly,

$$M^n = UD^nU^{-1} \tag{3}$$

Diagonalization: some uses

Using (3), we can even define and compute any functions of M that has a Taylor series, which is a pretty large collection. For example, recall that

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} \tag{4}$$

and the series converges absolutely on the whole real line. (The whole complex plane, actually.) Now consider the sum

$$\sum_{n=0}^{\infty} \frac{M^n}{n!} = \sum_{n=0}^{\infty} \frac{UD^nU^{-1}}{n!} = U\left(\sum_{n=0}^{\infty} \frac{D^n}{n!}\right)U^{-1} = Ue^DU^{-1} \tag{5}$$

where in the last equality we defined e^D in the natural way:

$$e^D = Diag\left(e^{d_1}, \ldots, e^{d_k}\right) \tag{6}$$

where k is the dimension of M.

Diagonalization: some uses

It is reasonable, therefore, to take (5) as definition of e^M.

Be careful, though,

$$e^{M_1+M_2} = e^{M_1}e^{M_2}$$

only if matrices M_1 and M_2 commute, i.e., $M_1M_2 = M_2M_1$, which is more or less equivalent to the matrices having the same sets of eigenvectors.

Singular Value Decomposition

- Unfortunately, M is not always diagonalizable and even when it is, in general, the diagonalization algorithm is not always stable. This makes it less than ideal method to use in practice, when you need software to "just run", without complaints or exceptions.
- For example, when you design automatic systems, you would then need to write code to deal with each possible complaint and exception, and that increases the software burden significantly.
- Also, we do not always need high powers of M: one power that we often have a very particular interest in is -1, i.e., the inverse, that appears in portfolio optimization as we will see a few slides later.

Singular Value Decomposition: definition

- Singular Value Decomposition of an m by n matrix M is given by an orthogonal m by n matrix U, a diagonal n by n matrix D, and orthogonal n by n matrix V such that

$$M = UDV^T \tag{7}$$

  The entries of D, which are non-negative, are called singular values and play a role similar to eigenvalues. This decomposition is very stable: it works in pretty much all practical situations and gives us a lot of information about M.

- For details and excellent discussion, see Ch.2 of Numerical Recipes.

Singular Value Decomposition

- If M is symmetric, it has an orthonormal eigenbasis. Since both SVD and diagonalization are unique (up to ordering), they must coincide whenever M is positive definite. But this allows us to diagonalize M using SVD:

$$M = UDV^T = UDV^{-1} = UDU^{-1}. \tag{8}$$

- Similarly, the inverse of M is $UD^{-1}V^T$ when M is indeed invertible, but, with svd, we can go a step further and compute its *pseudo-inverse* by leaving the zero elements of D at zero and only take reciprocals of the non-zero elements of D.

- We can also choose to set the small d_is to zero. This, by the way, is Principal Component Analysis (PCA).

Optimal Portfolio: no constraints

- Suppose we have a history of N securities for the last I days. Let H be the $I \times N$ matrix of their histories.
- Then their covariance matrix

$$C = H^T H \tag{9}$$

(We assumed for simplicity of notation and without loss of generality that these histories have zero mean.)

- Let R be the vector of the securities' expected returns.

It is natural to want to find the portfolio weights w so that

$$\langle R, w \rangle - a \langle w, Cw \rangle \tag{10}$$

is maximized. $\langle R, w \rangle$ is the expected return of the portfolio and $\langle w, Cw \rangle$ is its variance, a measure of risk. The constant a indicates the amount of risk aversion.

Optimal Portfolio: no constraints

- If the maximum is attained in the interior, the gradient has to be zero at the maximum. If we take the gradient of (10) with respect to w, we must have

$$R - 2aCw = 0. \tag{11}$$

In optimization theory (11) is called the first order condition. Solving for w, we get:

$$w = \frac{1}{2a} C^{-1} R \tag{12}$$

if C is invertible.

- Well, is it? Let's explore what happens in practice.

Covariance matrix: individual histories

- Suppose we have a large number of securities to model, i.e., the number of columns N of the history matrix H in (9) is large, say, 1000. Does it matter what I is?
- To find out, first recall from linear algebra that rank of H (and of H^T) is the smaller of I and N. It follows that the rank of $C = H^T H$ also cannot be greater than $\min(I, N)$.
- Therefore, if we want C to be invertible we must have $I \geq N$. In other words, we must have at least 1000 observations for *each* instrument if we are to have any hope of finding an optimal portfolio of 1000 instruments: normally, history should be many times longer than the number of instruments.
- If you use any method of maximizing (10), e.g., optimization, you are still trying to arrive at (12) and this non-invertibility will still be an issue, although it might manifest itself in some other way.

Covariance matrix: factors

- Some people try to avoid the problem on the previous slide by using factors. Let's look at this approach more closely. First we need some notation: let F be the matrix of histories of the chosen factors. The matrix F has the same length as H, but is narrower. Let L be the matrix of factor loadings. Every column of L holds the regression coefficients of the corresponding column of H on the factors. The matrix L, therefore, is as wide as H and its height equals the width of F.

- Regression of H onto F will not be exact, of course: it will have residuals. We gather them in a matrix Z, which will have the same shape as H.

- In summary,

$$H = FL + Z \tag{13}$$

Covariance matrix: factors

- Note that on the previous slide we did not specify what the factors are or how many of them we are looking at: for now, the analysis is completely generic.
- We have

$$\begin{aligned} C &= H^T H = (FL+Z)^T(FL+Z) \\ &= (Z^T + L^T F^T)(FL+Z) \\ &= Z^T FL + Z^T Z + L^T F^T FL + L^T F^T Z \\ &= Z^T Z + L^T F^T FL \end{aligned} \tag{14}$$

To see why the last equality holds, recall that regression of a column of H on the factors is a projection of that column onto the space spanned by the columns of F. That means that the residual must be perpendicular to the span of the columns of F. But that's just another way of saying that $Z^T F = 0$.

Covariance matrix: factors

- Let's look at (14) again:

$$
\begin{aligned}
C &= Z^T Z + L^T F^T F L \\
&= Z^T Z + (FL)^T F L
\end{aligned}
\tag{15}
$$

- Different choices of factors will lead to different decompositions. For example, if we choose no factors at all we will end up in the situation on slide 11 with $Z = H$.
- We can use just one factor, e.g., the market factor. That will give us CAPM.
- People often use Fama-French factors: market, size, book.
- Another common choice is to use sector indices as factors.
- Principal Component Analysis (PCA).

Covariance matrix: factors and rank

- One might think that the need for long history that we saw on slide 10 is gone when we use just a few factors.
- Let us look again at (15):
  1. The rank of F cannot exceed the number of factors. Therefore, the same must be true of FL.
  2. The rank of F is smaller than the rank of H, since it makes no sense to choose more factors than individual instruments.
  3. Recall that $Z = H - FL$ and that the range of Z is orthogonal to the range of FL.
  4. This means for C in (15) to be invertible, *both* $Z^T Z$ *and* $(FL)^T FL$ have to be invertible and, therefore, *both* Z *and* FL have to have more rows than columns, i.e., we must have more history than the number of instruments, *regardless of how we pick the factors.*

Covariance matrix: caution!

- What could possibly make C invertible?
- Suppose that we have the confidence (something the trading community does not lack) that we picked our factors so beautifully that the residuals are pure noise. Since they are noise, they must all be mutually orthogonal. That would make $Z^T Z$ in (15) diagonal. And, since each diagonal element is a variance of something, it must be positive.
- That would make C invertible, because it would be a sum of a strictly positive diagonal matrix and of a low-rank one.
- But the diagonal matrix, the one that makes the inversion possible, and dominates the portfolio, is made out of pure noise. By construction!
- Are we sure we want to use this method for our portfolio weights?

Covariance matrix: caution!

- There has to be a better way...
- Indeed, there are several.
- Anything is better than inverting noise.
- One thing you can do that's crude but effective is to replace $Z^T Z$ with a diagonal matrix of variances of the individual instruments. This works in the sense of giving a reasonable and reasonably stable portfolio, but has no logical justification, since it completely disregards the fact that Z is a matrix of residuals and simply plops down another matrix in its place.
- Another, more sophisticated, and also effective method is to follows Ledoit, *etal*, who blend C with a specially constructed diagonal matrix and do so with justification from statistics.

Covariance matrix: PCA

- In the methods mentioned above, the factors were picked "by hand", i.e., based on economic intuition, but with the side effect of running into mathematical difficulties.
- We could ignore economics and instead focus on math. Then we can pick the factors based on the the fraction of total variance that they hold.
- We do it by diagonalizing C so that its eigenvalues are in decreasing order. In other words, $C = UDU^{-1} = UDU^T$ for some diagonal D and orthonormal U.

Covariance matrix: PCA

- Now consider the matrix HU. We can think of columns of U as loadings on the instruments which are columns of H. In other words, columns of HU are some sort of factors.

- Let's compute the covariance matrix of these factors:

$$(HU)^T HU = U^T H^T HU = U^T CU = D. \tag{16}$$

  In other words, the factors that are the columns of HU are orthogonal.

- This method is called Principal Component Analysis. The factor that corresponds to the top eigenvalue is called the first principal component and it has the highest variance of all possible factors. The second principal component has the second highest variance, and so on.

Covariance matrix: a legitimate inverse

- How does this help us with C^{-1}?
- We use the magic of SVD. First we decide how many of the factors we trust. Normally, this is done by picking enough top eigenvalues to account for, say, 90% of the variance and setting the rest to zero.
- Then the pseudo-inverse is computed by replacing the remaining non-zero eigenvalues with their reciprocals.
- In effect, this method takes the inverse on the subspace spanned by the top few principal components.

Covariance matrix: stability of the portfolio

- The top eigenvalues of C are fairly stable under small changes to history, but the bottom ones are essentially noise.
- Therefore, the smallest eigenvalues can change dramatically when we recalculate C the next day. But it is these smallest eigenvalues that control the portfolio w. Therefore, the portfolio will jerk around day to day, which is not a desirable property at all.
- The SVD pseudo-inverse described on the previous slide helps with stability as well.
- So, finally, we have a means of putting together a portfolio that will be reasonably well-behaved.
- But all this was under the unrealistic assumption that there were no portfolio constraints.

Optimal Portfolio: with constraints

- Now let's consider (10), but with constraints

$$\max_w \{\langle R, w\rangle - a\langle w, Cw\rangle \mid \langle \vec{g}_k, w\rangle = c_k, \quad k = 1, \ldots, K\} \quad (17)$$

  where K is the number of constraints. It is usually smaller than the number of securities N.

- The corresponding Lagrangian is

$$L(w, \lambda) = \langle R, w\rangle - a\langle w, Cw\rangle - \sum_{k=1}^{K} \lambda_k \left(\langle \vec{g}_k, w\rangle - c_k\right) \qquad (18)$$

- Notice that maximizing $L(w, \lambda)$ is an *unconstrained* optimization problem (albeit with more variables) and will have the same solution as (17).

- There are many optimizers that handle such problems: I usually use IpOpt.

Optimal Portfolio: with constraints

It is actually possible to solve (18) in closed form. My preference would be to use an optimizer anyway, but I will show you the solution here to give an example of use of linear algebra.

First, it would be convenient to rewrite (18) in matrix form. To do that we define a matrix G whose rows are the vectors $\vec{g}_k$.

We also define $\lambda = (\lambda_1, \ldots, \lambda_K)$ to be the vector of all Lagrange multipliers and $c = (c_1, \ldots, c_K)$ to be the vector of all constraint values.

The matrix G is K by N and (18) becomes

$$\begin{aligned} L(w,\lambda) &= \langle R, w\rangle - a\,\langle w, Cw\rangle - \langle \lambda, (Gw - c)\rangle &(19)\\ &= \langle R, w\rangle - a\,\langle w, Cw\rangle - \langle \lambda, Gw\rangle + \langle \lambda, c\rangle &(20)\\ &= \langle R, w\rangle - a\,\langle w, Cw\rangle - \left\langle G^T\lambda, w\right\rangle + \langle \lambda, c\rangle &(21)\end{aligned}$$

Optimal Portfolio: with constraints

At the maximum of L the entire gradient of L is zero. Therefore,

$$\begin{aligned} 0 &= \nabla_w L = R - 2aCw - G^T\lambda && (22) \\ 0 &= \nabla_\lambda L = c - Gw && (23) \end{aligned}$$

It's time to stop and stare at (22) and (23). First, we can assume that the rows of G are of maximal rank. In particular, $\text{rank}(G) = \min(K, N)$. If they weren't, we could simply drop the redundant constraints. Similarly, we can assume that there are no redundant instruments, since we can remove those and reduce N.

Optimal Portfolio: with constraints

With that in mind we have three cases:

$K = N$**:** Same number of constraints and variables.

$K > N$**:** More constraints than variables.

$K < N$**:** Fewer constraints than variables.

In the first case, $K = N$, we have the same number of restrictions as instruments: a very unusual situation, but easily handled: since G is invertible in this case (why?), we can solve for w directly:

$$w = G^{-1}c. \tag{24}$$

Note that λ is not involved in the solution, which makes sense, since there is nothing to optimize.

Optimal Portfolio: with constraints

In the second case, $K > N$, we have more restrictions than instruments: an over-determined problem. Again, a strange situation in the context of portfolio construction. This occurs more naturally in the context of regression.

In this case the matrix $G^T G$ is $N \times N$ and of maximal rank, and, therefore, invertible. Multiplying (23) on the left by G^T, we get:

$$G^T G w = G^T c \tag{25}$$

and

$$w = (G^T G)^{-1} G^T c, \tag{26}$$

which is linear regression in matrix form.

Again, λ does not appear since there was, again, no optimization.

Optimal Portfolio: with constraints

Finally, the main case: $K < N$. Let's rewrite (22) and (23) here so we can stare at them again:

$$
\begin{aligned}
0 &= R - 2aCw - G^T\lambda && (27)\\
0 &= c - Gw && (28)
\end{aligned}
$$

We are after w, but we can no longer use the regression trick: the matrix G^TG is no longer invertible. (why?) But we had never used the equation (27). It connects w and λ and the rest are known quantities. We can try to solve for w in (27) and plug the result into (28). In order to do that, we will need to be able to invert C. But we had learned to do that earlier in the lecture: if necessary, we can make it a pseudo-inverse.

Optimal Portfolio: with constraints

Thus equipped, from (27), we get

$$w = \frac{1}{2a} C^{-1} \left(R - G^T \lambda \right) \tag{29}$$

where C^{-1} might be a pseudo-inverse.

Putting this into (28), we get

$$\begin{aligned} 2ac &= GC^{-1} \left(R - G^T \lambda \right) \\ &= GC^{-1} R - GC^{-1} G^T \lambda \end{aligned} \tag{30}$$

which can be solved for λ if we can find an inverse of $GC^{-1}G^T$. But, remember, in this case, G, being $K \times N$, is rank K. Therefore, multiplying C^{-1} by G and G^T will not reduce its rank. Moreover, it is a symmetric matrix and, therefore, we can find its pseudo-inverse in the same manner as we did for C.

Thus, we get λ from (30), and then w from (29) and (17) is solved!