Recall Ingredients for Portfolio Optimization

**Raw Data:**

\[ R_j(t) = \text{return on asset } j \text{ in time period } t \]

**Derived Data:**

\[ r_j = \frac{1}{T} \sum_{t=1}^{T} R_j(t) \]

\[ D_{tj} = R_j(t) - r_j. \]

**Decision Variables:**

\[ x_j = \text{fraction of portfolio to invest in asset } j \]

**Decision Criteria:**

\[ \text{reward}(x) = \sum_j r_j x_j \]

\[ \text{risk}(x) = \frac{1}{T} \sum_{t=1}^{T} \left| \sum_j D_{tj} x_j \right| \]
Set a value for *risk affinity* parameter $\mu$ (risk affinity is the reciprocal of risk aversion) and maximize a combination of *reward minus risk*:

\[
\text{maximize} \quad \mu \sum_j r_j x_j - \frac{1}{T} \sum_{t=1}^T \left| \sum_j D_{tj} x_j \right|
\]

subject to \( \sum_j x_j = 1 \)

\( x_j \geq 0 \quad \text{for all investments } j \)

Because of absolute values not a linear programming problem.

As usual, easy to convert...
A Linear Programming Formulation

maximize \[ \mu \sum_j r_j x_j - \frac{1}{T} \sum_{t=1}^{T} y_t \]

subject to \[ -y_t \leq \sum_j D_{tj} x_j \leq y_t \quad \text{for all times } t \]

\[ \sum_j x_j = 1 \]

\[ x_j \geq 0 \quad \text{for all investments } j \]

\[ y_t \geq 0 \quad \text{for all times } t \]

Note: The \( y_t \)'s are the absolute values of the deviations from the average reward. To be clear: they are not the dual variables.
Adding Slack Variables $w_t^+$ and $w_t^-$

maximize $\mu \sum_j r_j x_j - \frac{1}{T} \sum_{t=1}^{T} y_t$

subject to $-y_t - \sum_j D_{tj} x_j + w_t^- = 0$ for all times $t$

$-y_t + \sum_j D_{tj} x_j + w_t^+ = 0$ for all times $t$

$\sum_j x_j = 1$

$x_j \geq 0$ for all investments $j$

$y_t, w_t^-, w_t^+ \geq 0$ for all times $t$

A dictionary will have $2T + 1$ equations and $3T + n$ variables, $2T + 1$ of which will be basic and rest will be nonbasic (here, $n$ denotes the number of assets).
The Solution for Large $\mu$

Varying the risk bound $0 \leq \mu < \infty$ produces the efficient frontier.

Large values of $\mu$ favor reward maximization whereas small values favor minimizing risk.

Beyond some finite (but perhaps large) value for $\mu$, the optimal solution will be a portfolio consisting of just one asset—the asset $j^*$ with the largest average return:

$$r_{j^*} \geq r_j \quad \text{for all } j.$$  

For this case, it’s easy to identify basic (nonzero) vs. nonbasic (i.e. zero) variables:

- Variable $x_{j^*}$ is basic whereas the remaining $x_j$’s are nonbasic.
- All of the $y_t$’s are basic.
- If $D_{tj^*} > 0$, then $w_t^-$ is basic and $w_t^+$ is nonbasic. Otherwise, it is switched.

The algebra is tedious, but we can now write down a starting dictionary...
The Optimal Dictionary for Large $\mu$

Let

$$T^+ = \{ t : D_{t,j^*} > 0 \}, \quad T^- = \{ t : D_{t,j^*} < 0 \}, \quad \text{and} \quad \epsilon_t = \begin{cases} 1, & \text{for } t \in T^+ \\ -1, & \text{for } t \in T^- \end{cases}$$

Here's the optimal dictionary (for $\mu$ large):

$$\zeta = \frac{1}{T} \sum_{t=1}^{T} \epsilon_t D_{t,j^*} - \frac{1}{T} \sum_{j \neq j^*} \sum_{t=1}^{T} \epsilon_t (D_{t,j} - D_{t,j^*}) x_j - \frac{1}{T} \sum_{t \in T^-} w^- - \frac{1}{T} \sum_{t \in T^+} w^+ + \mu r_{j^*} + \mu \sum_{j \neq j^*} (r_j - r_{j^*}) x_j$$

$$y_t = -D_{t,j^*} - \sum_{j \neq j^*} (D_{t,j} - D_{t,j^*}) x_j + w^- \quad \text{for } t \in T^-$$

$$w_t^- = 2D_{t,j^*} + 2 \sum_{j \neq j^*} (D_{t,j} - D_{t,j^*}) x_j + w^+ \quad \text{for } t \in T^+$$

$$y_t = D_{t,j^*} + \sum_{j \neq j^*} (D_{t,j} - D_{t,j^*}) x_j + w^+ \quad \text{for } t \in T^+$$

$$w_t^+ = -2D_{t,j^*} - 2 \sum_{j \neq j^*} (D_{t,j} - D_{t,j^*}) x_j + w^- \quad \text{for } t \in T^-$$

$$x_{j^*} = 1 - \sum_{j \neq j^*} x_j$$
Collected data for 719 stocks (and bonds, etc.) from January 1, 1990, to March 18, 2002.

Hence,

\[ n = 719 \]

and

\[ T = 3080. \]
Click here for an expanded browser view.
Computing the Efficient Frontier

Using a reasonably efficient code for the parametric self-dual simplex method (simpo), it took 22,000 pivots and 1.5 hours to solve for one point on the efficient frontier.

Customizing this same code to solve parametrically for every point on the efficient frontier, it took 20,500 pivots and 57 minutes to compute every point on the frontier.

The efficient frontier consists of 1308 distinct portfolios. Click here for a complete list (warning: the file is 2.5 MBytes).
Lasso Regression

The problem is to solve a sparsity-encouraging “regularized” regression problem:

$$\text{minimize} \quad \|Ax - b\|_2^2 + \lambda \|x\|_1$$

My reaction:

Why not replace least squares (LS) with least absolute deviations (LAD)?

LAD is to LS as median is to mean. Median is a more robust statistic (i.e., insensitive to outliers).

The LAD version can be recast as a linear programming (LP) problem.

If the solution is expected to be sparse, then the simplex method can be expected to solve the problem very quickly.

No one knows the “correct” value of the parameter $\lambda$. The parametric simplex method can solve the problem for all values of $\lambda$ from $\lambda = \infty$ to a small value of $\lambda$ in the same (fast) time it takes the standard simplex method to solve the problem for one choice of $\lambda$. 