CSCI 1951-G – Optimization Methods in Finance
Part 04: Building Index Funds with Integer Programming

February 26, 2016
Stock market index

Definition (Stock market index)

A measure of the *composite value of a set of stocks*
E.g.: S&P500, Dow Jones Industrial Avg, NASDAQ composite

- used to quickly evaluate the value of a segment of the market, and its behavior over time
- A statistic, not an asset (you cannot invest in it)
- The composite value $V$ is a *weighted sum* of the stock prices:

$$V = \sum_{\text{stock } i} w_i p_i$$

**price-weighted:** $w_i = 1$, for each stock $i$

**market-share:** $n_i$: # of outstanding shares for stock $i$,

$$w_i = n_i / \sum_{\text{stock } i} n_i$$

**market-capitalization:** $v_i = n_i p_i$: total value of outstanding shares for stock $i$,

$$w_i = v_i / \sum_{\text{stock } i} v_i$$
Index fund

Definition (Index fund)

An investment strategy whose goal is to provide returns similar to that of an index

I.e., a portfolio of stocks whose behavior, collectively, replicates (tracks) that of the index

• the stocks are picked at the beginning
• each stock makes up a portion of the portfolio
• neither the stocks nor their portions change over time
• Exception: periodic rebalancing (adjustment) of index

Easy way to build an index fund

Buy each stock in the index proportionally to its weight

This is bad because we end up with few shares of many stocks, which is inefficient in a finance sense and has a cost in practice
An investment strategy whose goal is to provide returns similar to that of an index

I.e., a portfolio of stocks whose behavior, collectively, replicates (tracks) that of the index

- the stocks are picked at the beginning
- each stock makes up a portion of the portfolio
- neither the stocks nor their portions change over time
- Exception: periodic rebalancing (adjustment) of index

Easy way to build an index fund

Buy each stock in the index proportionally to its weight

This is bad because we end up with few shares of many stocks, which is inefficient in a finance sense and has a cost in practice
**Motivation**

- Getting above average returns consistently is difficult
- No one seems to be doing it consistently
- Why not just trying to *always hit the average*?
Motivation

- Getting above average returns consistently is difficult
- No one seems to be doing it consistently
- Why not just trying to *always hit the average*?

- Index funds are a relatively recent finance “innovation”
- Ridiculed at the beginning, now they are very successful:
  “[I couldn’t] believe that the great mass of investors are going to be satisfied with receiving just average returns”
  Fidelity Investments Chairman Edward Johnson

- Most investors are average people
  (no, not you, you’re a special flower)
- They should actually be fine with average returns!
Managed vs unmanaged investment strategies

- Index funds are *unmanaged* investment strategies: we buy stocks at the beginning and we hold them, no matter what.
- *Managed* strategies involve continuously looking for overvalued stocks to sell and undervalued stocks to buy: they look for stocks that will *outperform the markets*.

**Question**

*Which one is better?*

**Answer (It’s not clear!)**

- *Theoretical ground and empirical evidence suggest that unmanaged strategies are better*.
- *There is also evidence to the opposite.*
Why may index funds be better?

Efficient Market Hypothesis (E. Fama)

“[Stock] prices fully reflect all available information”

More precisely: the prices reflect information to the point that marginal returns that could be obtained by exploiting additional information do not exceed marginal costs
Why may index funds be better?

Efficient Market Hypothesis (E. Fama)

“[Stock] prices fully reflect all available information”

More precisely: the prices reflect information to the point that marginal returns that could be obtained by exploiting additional information do not exceed marginal costs

- Fund managers are continuously looking outperforming stocks
- The field is very competitive
- Any information about the real value of a company will be almost immediately reflected in the stock price
- This happens so fast that it is impossible to detect potentially market-outperforming stocks, not to mention act on them
- So why even try to find them?
Empirical evidence

- Historically, actively managed strategies have often underperformed with respect to many indices.
- “Average returns” would have been...better than average!
- An index fund mirrors the whole market, so it avoids the inefficiencies of having to continuously pick stocks.
- An index fund has lower management costs (you do not have to pay someone a lot for picking stocks well).

“Put 10% ... in short-term government bonds and 90% in a very low-cost S&P 500 index fund. (I suggest Vanguard’s.) I believe the trust’s long-term results from this policy will be superior to those attained by most investors – whether pension funds, institutions or individuals – who employ high-fee managers.” Warren Buffett, 2014
Why may index funds not be better?

• There are many theoretical critiques to the Efficient Market Hypothesis, including the fact that it assumes rational investors. If interested, look for *behavioral finance*

• There is also empirical evidence:
  • Indices announce in advance when they are going to rebalance and how
  • This creates changes in prices that actively managed fund managers exploit to extract value while index funds can’t use this information
How to build an index fund

Easy way to build an index fund
Buy each stock in the index proportionally to its weight

This is bad because we end up with few shares of many stocks, which has a cost in practice

Our Goal
We want to pick a few stocks that closely replicates the whole index

From a finance point of view, this won’t be the best set of stocks we could pick, but it is fine for our purposes
Data and settings

- The index contains $n$ stocks
- We want to pick $q$ stocks for our index fund, with $q \ll n$
- For each pair $(i, j)$ of stocks, let $\rho_{ij}$ be their similarity. $\rho_{ii} = 1$, $\rho_{ij} \leq 1$. E.g., the correlation of their returns, but other similarity measures are possible
- In practice, $\rho_{ij}$ is estimated from historical data: it may include the sizes of the companies, their sector, the historical behavior of their stock prices, ..
An Integer Programming formulation

- We take a clustering approach: each stock $i$ will be represented by one (and only one) stock $j$ in the index
- We want to pick the $q$ representatives that maximize the similarity between the $n$ stocks and the representatives
An Integer Programming formulation

- We take a *clustering approach*: each stock $i$ will be represented by one (and only one) stock $j$ in the index
- We want to pick the $q$ representatives that maximize the similarity between the $n$ stocks and the representatives
- Let $y_j$ be 1 if stock $j$ is chosen as representative, 0 otherwise
- Let $x_{ij} = 1$ if stock $j$ is the representative for stock $i$, 0 otherwise
An Integer Programming formulation

- We take a *clustering approach*: each stock $i$ will be represented by one (and only one) stock $j$ in the index.
- We want to pick the $q$ representatives that maximize the similarity between the $n$ stocks and the representatives.
- Let $y_j$ be 1 if stock $j$ is chosen as representative, 0 otherwise.
- Let $x_{ij} = 1$ if stock $j$ is the representative for stock $i$, 0 otherwise.

\[
\begin{align*}
\text{max} & \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} \\
\text{s.t.} & \quad \sum_{j=1}^{n} y_j = q \quad \\
& \quad \sum_{j=1}^{n} x_{ij} = 1 \text{ for } i = 1, \ldots, n \quad \\
& \quad x_{ij} \leq y_j \text{ for } i = 1, \ldots, n, j = 1, \ldots, n \quad \\
& \quad x_{ij}, y_j \in \{0, 1\} \text{ for } i = 1, \ldots, n, j = 1, \ldots, n
\end{align*}
\]
Proportions

Question

How many shares of each representative stock should we buy?

Answer

A number proportional to the value of the stocks it represents

After having selected the $q$ stocks, we compute, for each selected stock $j$, the weight $w_j = \sum_{i=1}^{n} v_i x_{ij}$, where $v_i$ is the total market value of the stocks represented by $j$ in the index fund.

The fraction of the index fund to invest into stock $j$ is proportional to $w_j$:

$p_j = \frac{w_j}{\sum_{i=1}^{n} w_i}$
Proportions

Question

How many shares of each representative stock should we buy?

Answer

A number proportional to the value of the stocks it represents

- After having selected the $q$ stocks, we compute, for each selected stock $j$, the weight

$$w_j = \sum_{i=1}^{n} v_i x_{ij},$$

where $v_i$ is the total market value of the stocks represented by $j$ in the index fund

- The fraction of the index fund to invest into stock $j$ is proportional to $w_j$:

$$p_j = \frac{w_j}{\sum_{i=1}^{n} w_i}$$
Issue with this formulation

\[
\begin{align*}
\text{max} & \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} \\
\text{s.t.} & \quad \sum_{j=1}^{n} y_j = q \\
& \quad \sum_{j=1}^{n} x_{ij} = 1 \text{ for } i = 1, \ldots, n \\
& \quad x_{ij} \leq y_j \text{ for } i = 1, \ldots, n, j = 1, \ldots, n \\
& \quad x_{ij}, y_j \in \{0, 1\} \text{ for } i = 1, \ldots, n, j = 1, \ldots, n
\end{align*}
\]

Question

Let \( n = 500 \) and \( q = 20 \)

How many variables has this formulation?

How many constraints?
Issue with this formulation

\[
\text{max } \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} \\
\text{s.t. } \sum_{j=1}^{n} y_j = q \\
\sum_{j=1}^{n} x_{ij} = 1 \text{ for } i = 1, \ldots, n \\
x_{ij} \leq y_j \text{ for } i = 1, \ldots, n, j = 1, \ldots, n \\
x_{ij}, y_j \in \{0, 1\} \text{ for } i = 1, \ldots, n, j = 1, \ldots, n
\]

Question

Let \( n = 500 \) and \( q = 20 \)

\textit{How many variables has this formulation?}

\textit{How many constraints?}

Despite the efficiency of LP algorithms, solving many LP-relaxations of this problem is not a viable option.
Speed up

Question

What is the goal of solving the LP-relaxation during the branch-and-bound algorithm?
Speed up

Question

What is the goal of solving the LP-relaxation during the branch-and-bound algorithm?

Answer

Obtaining upper and lower bounds to the optimal integral solution

- We want to obtain the bounds fast
- We obtained the LP-relaxation by relaxing the integrality constraint
- What about relaxing also the other constraints?
Lagrangian relaxation

Idea to speed up the computation of bounds

- Remove some of the constraints that specify the feasible region
- Add those constraints into the objective function, penalizing their violation

Consider the ILP:  
\[
\begin{align*}
\text{max} & \quad c^T x \\
A_1 x & \leq b_1 \\
A_2 x & \leq b_2
\end{align*}
\]  (1)

We move (2) into the objective function, penalizing its violation:

\[
L(\lambda) = \max c^T x + \lambda^T (b_2 - A_2 x)
\]

where \( \lambda \) is a vector of non-negative weights

The objective value is penalized when \( A_2 x \not\leq b_2 \), and “rewarded” otherwise
Lagrangian relaxation (cont.)

\[ L(\lambda) = \max c^T x + \lambda^T (b_2 - A_2 x) \]

\[ A_1 x \leq b_1 \]

**Definition**

The function \( L(\lambda) \) is known as the *Lagrangian function* the modified optimization problem as *Lagrangian relaxation*, and the weights \( \lambda \) as *Lagrangian multipliers*

**Question**

*What can we say about the optimal solution of the Lagrangian relaxation vs. the optimal solution of the original ILP?*
Lagrangian relaxation (cont.)

\[ L(\lambda) = \max c^T x + \lambda^T (b_2 - A_2 x) \]
\[ A_1 x \leq b_1 \]

<table>
<thead>
<tr>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>The function ( L(\lambda) ) is known as the Lagrangian function the modified optimization problem as Lagrangian relaxation, and the weights ( \lambda ) as Lagrangian multipliers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>What can we say about the optimal solution of the Lagrangian relaxation vs. the optimal solution of the original ILP?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Theorem</th>
</tr>
</thead>
<tbody>
<tr>
<td>The optimal solution of the Lagrangian relaxation is an upper bound to the optimal solution of the original ILP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Proof.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework!</td>
</tr>
</tbody>
</table>
The Lagrangian relaxation of the index fund ILP problem is:

\[
L(\lambda) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} + \sum_{i=1}^{n} \lambda_i \left(1 - \sum_{j=1}^{n} x_{ij}\right)
\]

s.t. \[\sum_{j=1}^{n} \ y_j = q\]

\[x_{ij} \leq y_j \text{ for } i = 1, \ldots, n, j = 1, \ldots, n\]

\[x_{ij}, y_j \in \{0, 1\} \text{ for } i = 1, \ldots, n, j = 1, \ldots, n\]

Question

How many constraints has this formulation?
Back to index funds

The Lagrangian relaxation of the index fund ILP problem is:

\[
L(\lambda) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} + \sum_{i=1}^{n} \lambda_i \left(1 - \sum_{j=1}^{n} x_{ij}\right)
\]

\[
s.t. \sum_{j=1}^{n} y_j = q
\]

\[
 x_{ij} \leq y_j \text{ for } i = 1, \ldots, n, j = 1, \ldots, n
\]

\[
x_{ij}, y_j \in \{0, 1\} \text{ for } i = 1, \ldots, n, j = 1, \ldots, n
\]

Question

How many constraints has this formulation?

Answer

We cut the number of constraints in half: solving the LP-relaxations will be much faster
Lagrangian relaxation for index funds

\[ L(\lambda) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} \rho_{ij} x_{ij} + \sum_{i=1}^{n} \lambda_i \left( 1 - \sum_{j=1}^{n} x_{ij} \right) \]

Property 1

Let \( Z \) be the optimal solution of the original ILP. Then \( L(\lambda) \geq Z \), for any \( \lambda \geq 0 \).
Lagrangian relaxation for index funds (cont.)

We can rewrite $L(\lambda)$ as:

$$L(\lambda) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} (\rho_{ij} - \lambda_i)x_{ij} + \sum_{i=1}^{n} \lambda_i$$

Let

$$(\rho_{ij} - \lambda_i)^+ = \begin{cases} 
\rho_{ij} & \text{if } \rho_{ij} - \lambda_i > 0 \\
0 & \text{otherwise}
\end{cases}$$

and $C_j = \sum_{i=1}^{n} (\rho_{ij} - \lambda_i)^+$. (these values are known constants)
Lagrangian relaxation for index funds (cont.)

We can rewrite $L(\lambda)$ as:

$$L(\lambda) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} (\rho_{ij} - \lambda_i)x_{ij} + \sum_{i=1}^{n} \lambda_i$$

Let

$$(\rho_{ij} - \lambda_i)^+ = \begin{cases} 
\rho_{ij} & \text{if } \rho_{ij} - \lambda_i > 0 \\
0 & \text{otherwise}
\end{cases}$$

and $C_j = \sum_{i=1}^{n} (\rho_{ij} - \lambda_i)^+$. (these values are known constants)

Property 2

We have:

$$L(\lambda) = \max \sum_{j=1}^{n} C_j y_j + \sum_{i=1}^{n} \lambda_i$$

s.t.

$$\sum_{j=1}^{n} y_j = q$$

$$y_j \in \{0, 1\} \text{ for } j = 1, \ldots, n$$

This formulation has $n$ variables and 1 constraint!
Lagrangian relaxation for index funds (cont.)

\[ L(\lambda) = \max \sum_{j=1}^{n} C_j y_j + \sum_{i=1}^{n} \lambda_i \]

s.t. \( \sum_{j=1}^{n} y_j = q \)

\( y_j \in \{0, 1\} \) for \( j = 1, \ldots, n \)

Property 3

For any optimal solution to the Lagrangian relaxation:

- \( y_i = 1 \) for the \( q \) largest value of \( C_i \), otherwise \( y_i = 0 \).
- if \( \rho_{ij} - \lambda_i > 0 \), then \( x_{ij} = y_j \), otherwise \( x_{ij} = 0 \)
Lagrangian relaxation for index funds (cont.)

\[ L(\lambda) = \max \sum_{j=1}^{n} C_j y_j + \sum_{i=1}^{n} \lambda_i \]

\[ \text{s.t. } \sum_{j=1}^{n} y_j = q \]

\[ y_j \in \{0, 1\} \text{ for } j = 1, \ldots, n \]

**Property 3**

For any optimal solution to the Lagrangian relaxation:
- \( y_i = 1 \) for the \( q \) largest value of \( C_i \), otherwise \( y_i = 0 \).
- if \( \rho_{ij} - \lambda_i > 0 \), then \( x_{ij} = y_j \), otherwise \( x_{ij} = 0 \)

**Property 4**

Consider the solution that:
- Select the \( q \) stocks with the largest \( C_j \) values as representative
- Assign each other stock to the most similar representative
This solution is feasible for the original ILP, so its objective value is a lower bound to the optimal!
Lagrangian relaxation for index funds (cont.)

- We can obtain upper and lower bounds to the optimal ILP objective value by computing $L(\lambda)$, where we fixed $\lambda \geq 0$
- To obtain a good upper bound we can compute

$$\min_{\lambda \geq 0} L(\lambda)$$

Question

Do we know how to solve this problem?

It is not a linear problem!

In the next part we will see how to solve non-linear problems.