CSCI 1951-G – Optimization Methods in Finance
Part 00:
Course Logistics
Introduction to Finance Optimization Problems

January 29, 2016
Basic information

All information is available in the syllabus

Instructors: Matteo Riondato and Eli Upfal
TAs: Samuel Ainsworth (HTA) and Ragna Eide (UTA)
Time and Place: Fri, 2–4.20pm (with 15 mins break), CIT 316
Office Hours: Tue and Thu, 7–9pm, CIT 219
Website: https://cs.brown.edu/courses/cs1951g/
    Syllabus, Diary, Slides, Assignments, Solutions, ...
Mailing list: cs1951g.2015-16.s@lists.cs.brown.edu
    You will be subscribed to the list if you are enrolled into banner.
Write to the staff: cs1951gtas@cs.brown.edu
Help us help you!

Active participation creates an inclusive and productive environment for teaching and learning.

**What can we do**

**Ask lots of questions:** we all will learn more  
**Ask to explain again:** we all will learn more  

This is the **first time** the course is offered.

**What can we do**

**Help polishing the materials:** future students will learn more  
**Be patient:** we are all doing our best
Course goals

Direct goals

• Learn about different classes of optimization problems, and the theory and algorithms for solving them efficiently.
• Become familiar with some financial instruments and problems from computational finance, and understand how to solve them using optimization.
• Learn how to use popular optimization software packages and modeling languages.

At a higher level, develop

• algorithmic intuition
• mathematical and scientific writing skills
• theory-to-practice transfer skills.
Preliminary list of topics and schedule

**Linear Programming:** duality, geometry of optimal solution, the simplex and the dual simplex algorithm, short-term financing and asset pricing. *Weeks 1 to 4*

**Integer Programming:** mixed integer linear programming, branch and bound, cutting planes, constructing an index fund. *Week 4 to 6*

**Non-Linear Programming:** Newton’s method, steepest descent, generalized reduced gradient, volatility estimation. *Week 6 to 8*

**Quadratic Programming:** Interior point methods, portfolio optimization: mean-variance optimization and maximizing the Sharpe ratio. *Week 8 to 10*

**Stochastic Programming:** Two stage problems, risk measures, asset/liability management. *Week 10 to 12*

**Robust Optimization** Uncertainty sets, robust portfolio selection. *Week 12 to 13*
Textbook and other materials


**Additional readings:** a small number of additional notes will be posted on the course website.

Assessments and grading

Weekly Homework Assignments

• On website after class, due the following Friday, before class.
• “Theoretical” and programming exercises.
• Collaboration and late assignment policies in syllabus.
• Midterm is non-collaborative homework assignment.
• Final is take-home, non-collaborative.

Grading

• We care about details: more detail in your answer is better than less, but too much is bad.
• Correctness of code is more important than performances, but code that is too slow is bad.
• Final course grade is weighted:
  Homework assignments (including midterm): 60%
  Final: 40%
Why “in finance”?

Modern finance is a mathematical and computational discipline.

Goal

Given my current wealth, **maximize** my future wealth.

Formalizing the goal requires a mathematical definition of market models, financial instruments, constraints, risk measures, . . .

**Computational challenges due to:**

- the scale of modern investment strategies; and
- the speed of the markets; and
- the inherent complexity of some formalizations of the goal.
Financial workflow

A possible workflow for performing financial investments is the following:

1. Begin with a parametric mathematical model of the prices of instruments and how they relate to each other, as function of the parameters;
2. Use historical data (+ statistics / machine learning) to estimate the parameters of the model;
3. Optimize the model given your goals/constraints;
4. Analyze the optimal solution and, if it is cost-effective, perform the investments.
Some terminology

**Asset:** anything, tangible or intangible, with positive economic value. E.g., your car (if not too old).

**Liabilities:** the amounts you own to others. E.g., the balance on your credit card.

**Equity:** The difference between your assets and your liabilities.

**Security:** a tradeable financial asset.

**Stock/share:** a fraction of the equity of an entity.

**Bonds:** a debt security. The issuer of the bond owes money to the holder of the bond, and must pay them interest at a specific date.

**Return:** the profit of an investment in securities.

**Portfolio:** a collection of investments you own.

**Cash flow:** when you receive a payment or pay something, there is a positive or negative cash flow in your wallet.
Example 1: Portfolio optimization

The theory of portfolio selection was developed by Harry Markowitz in the 50s. He won the Nobel prize in Economics in 1990.

**Settings**

- We have a certain amount of money to be invested in a number of different securities (stock, bonds, ...) with random returns.
- For each security, we know the expectation and variance of its return.
- For each pair of securities, we know their correlation coefficient.

**Task (Mean-variance optimization)**

Use the available money to create a portfolio of securities such that
- the expected return of the portfolio exceed some minimal value;
- the variance of the return of the portfolio is minimized.

Can be extended to include regulations, investment preferences, short sales, etc.
Optimization problems

Optimization deals with maximizing a function of multiple variables under a set of constraints.

**Optimization problem**

Given $f : \mathbb{R}^n \to \mathbb{R}$ and $S \subseteq \mathbb{R}^n$, find $x^* \in \mathbb{R}^n$ that solves

$$\min_{x \in S} f(x)$$

$f$ is the objective function, $S$ the feasible region, and the components of $x$ are the decision variables.

Solving the optimization problem means finding a feasible assignment of the decision variables that minimizes the objective function.
Constraints

\( S \) is usually specified using functional *constraints* on the variables:
\[ S = \{ x : g(x) = 0, g \in E, \text{and } h(x) \geq 0, h \in I \} \]

Domains

Each decision variable \( x_i \) may be further restricted to a domain \( D_i \) (e.g., \( D_i = \mathbb{Z} \) or \( D_i = \{0, 1\} \)).

Hence, we write a generic optimization problem as:

**Optimization problem – general form**

\[
\begin{align*}
\min & \quad f(x_1, \ldots, x_n) \\
\text{subject to} & \quad g(x_1, \ldots, x_n) = 0, g \in E \\
& \quad h(x_1, \ldots, x_n) \geq 0, h \in I \\
& \quad x_i \in D_i, 1 \leq i \leq n
\end{align*}
\]
Example 2: asset/liability cash-flow matching

Corporations must be able to **finance their short-term cash commitments** (e.g., payrolls, taxes, ...) or face bankruptcy.

<table>
<thead>
<tr>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Given</strong></td>
</tr>
<tr>
<td>• some cash flow requirements over a time period; and</td>
</tr>
<tr>
<td>• some available sources of funds,</td>
</tr>
<tr>
<td>determine how to use the sources over the time period so that</td>
</tr>
<tr>
<td>• the cash flow requirements are always satisfied; and</td>
</tr>
<tr>
<td>• the wealth at the end of the time period is maximized.</td>
</tr>
</tbody>
</table>
asset/liability cash-flow matching (cont.)

Cash flow scenario (amounts in thousands of USD)

<table>
<thead>
<tr>
<th>Month</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
</tr>
</thead>
<tbody>
<tr>
<td>Net Cash Flow</td>
<td>-150</td>
<td>-100</td>
<td>200</td>
<td>-200</td>
<td>50</td>
<td>300</td>
</tr>
</tbody>
</table>

Sources of funds

- a line of credit up to $100k, at an interest rate of 1% per month;
- in any of the first three months, issue 90-day commercial paper bearing a total interest of 2% for the three-month period;
- invest excess funds at an interest rate of 0.3% per month.

Task

For each month (Jan to June), decide how to use each source of funds to satisfy the cash flow requirements (including any additional liability due to the use of the sources), and maximize the wealth at the end of June.
asset/liability cash-flow matching (cont.)

<table>
<thead>
<tr>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>For each month (Jan to June), decide how to use each source of funds to satisfy the cash flow requirements (including any additional liability due to the use of the sources), and maximize the wealth at the end of June.</td>
</tr>
</tbody>
</table>

How to *model* this task as an optimization problem?

<table>
<thead>
<tr>
<th>We need:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• decision variables and their domains;</td>
</tr>
<tr>
<td>• objective function;</td>
</tr>
<tr>
<td>• equality and inequality constraints.</td>
</tr>
</tbody>
</table>
asset/liability cash-flow matching (cont.)

Decision variables

- \( c_i \in [0, 100] \): amount from credit line in month \( i \), \( 1 \leq i \leq 5 \)
  balance on the credit line, not incremental borrowing;
- \( p_i \geq 0 \): amount of commercial paper issued in month \( i \), \( 1 \leq i \leq 3 \);
- \( e_i \geq 0 \): excess funds in month \( i \), \( 1 \leq i \leq 5 \)
- \( w \in \mathbb{R} \): wealth at the end of June

Objective

Maximize \( w \).
Constraints

**January:** We can borrow $c_1$ from the line of credit and issue $p_1$ commercial paper. After satisfying the cash requirement, we may have $e_1$ excess funds. It must hold

\[ c_1 + p_1 - 150 = e_1, \text{ i.e. } c_1 + p_1 - e_1 = 150. \]

**February:** In addition to what we could do in January,
- we must pay the interest of 1% on the amount $c_1$ we borrowed in January from the line of credit;
- we receive the 0.03% on the January excess funds $e_1$;

It must hold $c_2 + p_2 + 1.003e_1 - 100 - 1.01c_1 = e_2$, i.e.,

\[ c_2 + p_2 + 1.003e_1 - 1.01c_1 - e_2 = 100. \]

**March:** Similarly as in February, it must hold:

\[ c_3 + p_3 + 1.003e_2 - 1.01c_2 - e_3 = -200. \]
Constraints (cont.)

April: We can no longer issue commercial paper, rather we must pay the 2% interest on the amount \( p_1 \) of commercial paper issued in January. It must hold:
\[
c_4 - 1.02p_1 - 1.01c_3 + 1.003e_3 - e_4 = 200 .
\]

May: Similarly to April, we must have
\[
c_5 - 1.02p_2 - 1.01c_4 + 1.003e_4 - e_5 = -50 .
\]

June: We can no longer request credit, and the excess funds at the end of June are the wealth \( w \). It must hold
\[
-1.02p_3 - 1.01c_5 + 1.003e_5 - w = -300 .
\]
The complete formulation of our optimization problem is:

\[
\begin{align*}
\text{max } & \quad w \\
\text{s.t. } & \quad c_1 + p_1 - e_1 = 150 \\
& \quad c_2 + p_2 + 1.003e_1 - 1.01c_1 - e_2 = 100 \\
& \quad c_3 + p_3 + 1.003e_2 - 1.01c_2 - e_3 = -200 \\
& \quad c_4 - 1.02p_1 - 1.01c_3 + 1.003e_3 - e_4 = 200 \\
& \quad c_5 - 1.02p_2 - 1.01c_4 + 1.003e_4 - e_5 = -50 \\
& \quad -1.02p_3 - 1.01c_5 + 1.003e_5 - w = -300 \\
& \quad 0 \leq c_i \leq 100, \quad 1 \leq i \leq 5 \\
& \quad p_i > 0, \quad 1 \leq i \leq 3 \\
& \quad e_i \geq 0, \quad 1 \leq i \leq 5
\end{align*}
\]
Modeling the portfolio optimization problem

Settings:
• For each security \( i, 1 \leq i \leq n \) we know the expectation \( \mu_i \) and the variance \( \sigma_i^2 \) of its return.
• For each pair \((i, j)\) of securities, we know the correlation coefficient \( \rho_{ij} \) of their returns \( (\rho_{ii} = 1) \).
• Let \( \mu \) be the vector of the \( \mu_i \), and let \( Q \) be the \( n \times n \) matrix with entries \( Q_{ij} = \rho_{ij} \sigma_i \sigma_j \).

Decision variables: \( x_i \geq 0, 1 \leq i \leq n \), denoting the fraction of the portfolio allocated to security \( i \).

Constraints:
• We want to allocate all the portfolio: \( e^T x = 1 \) where \( e \) is the \( n \)-dimensional vector of all \( 1 \).
• The expected return of the portfolio must be at least \( R \). Using linearity of expectation, we want \( \mu^T x \geq R \).

Objective: minimize the variance of the portfolio, \( x^T Q x \).
Modeling the portfolio optimization problem (cont.)

Formulation

\[ \min x^T Q x \]

\[ e^T x = 1 \]

\[ \mu^T x \geq R \]

\[ x \geq 0 \]
Classes of optimization problems

Optimization problems are classified depending on:
• domains of the variables (e.g., continuous vs. discrete)
• types of constraints (e.g., linear, non-linear)
• the type of objective function (e.g., linear, non-linear, quadratic, convex, ...)

Example
The Cash Flow Matching problem has real variables, linear constraints, and a linear objective function: it is a Linear Programming (LP) problem.

Example
The Portfolio Optimization problem has real variables, linear constraints, and a quadratic objective function: it is a Quadratic Programming (QP) problem.

Different classes of optimization problems have different computational complexity (LP and QP are polynomial).