Loan Portfolio Risk and Optimization

Justin A. Sirignano

Imperial College London

September 27, 2015

Joint work with Kay Giesecke (Stanford) and Gerry Tsoukalas (UPenn)

Funded by the National Science Foundation
Methodology, Measurement, and Statistics Grant No. 1325031

Financial Crisis

- Unprecedented numbers of mortgage defaults led to the 2008 financial crisis.
- In the aftermath of the financial crisis, there is a pressing need for new models and computational methods for risk analysis of mortgages and other loans.
- Challenge: **pools of loans are very large**

How Large Can Pools be in Practice?

- Mortgage-backed securities (MBS) typically have thousands to hundreds of thousands of mortgages.
- Fannie Mae and Freddie Mac have credit exposure to 25 million mortgages.
- Major banks can have credit exposure to 10 million mortgages.
- Banks need to price thousands of mortgage-backed securities and hundreds of collateralized mortgage obligations (CMOs) on a daily basis.
- A credit card asset-backed security (ABS) can have tens of millions of credit cards.

Goals

1. Efficient computation of distribution of default and prepayment rates for pools of loans

2. Large-scale loan portfolio optimization

Pre-crisis Approaches to Risk Analysis

- Loan-by-loan modeling of such large pools is very computationally expensive!
  - **hours, days, or even weeks**
- Instead, rating agencies, banks, and investors often used simplistic approaches relying only upon average features of pools (e.g., average credit score).
- Pool-level characteristics can lead to inaccurate results due to ignoring the full loan-level distribution!

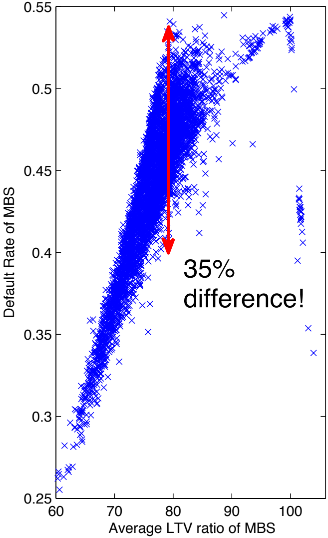
Why is Loan-level Analysis Needed?

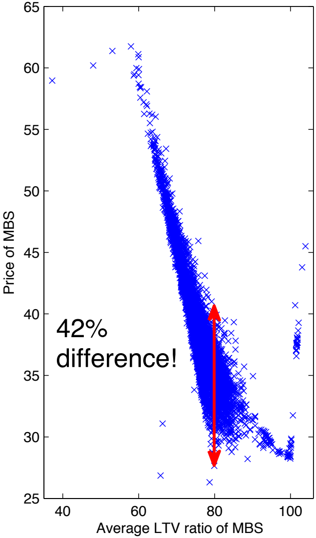
- Loan-to-value (LTV) ratio $= \frac{\text{size of loan}}{\text{value of house}} \times 100\ \%$.

LTV ratio	Default Rate
10 %	3.1 %
50 %	3.3 %
90 %	17.9 %

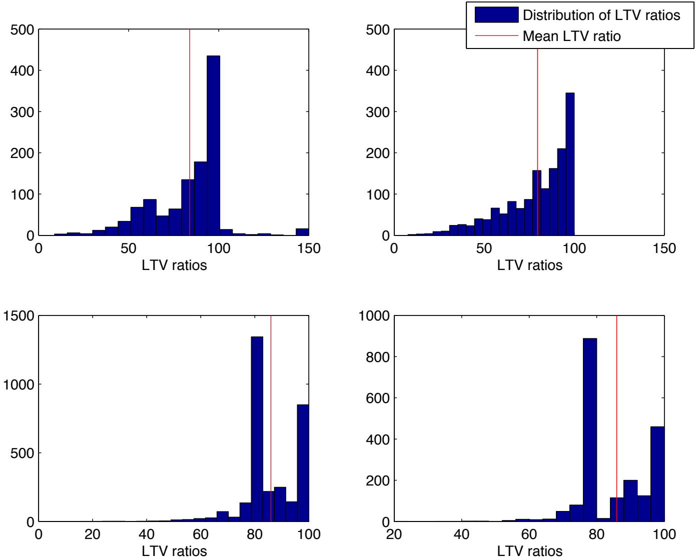
- Pool A only has mortgages with LTV ratio 50 %.
- Pool B has half its mortgages with LTV ratio 10 % and half with LTV ratio 90 %.

	Average LTV ratio	Default Rate
Pool A	50 %	3.3 %
Pool B	50 %	10.5 %


0.55
0.5
0.45
0.4
0.35
0.3
0.25
Default Rate of MBS
60
70
80
90
100
Average LTV ratio of MBS
35%
difference!


65
60
55
50
45
40
35
30
25
Price of MBS
40
60
80
100
Average LTV ratio of MBS
42%
difference!

Example MBS pools



This Talk

1. Prove weak convergence results for a broad class of models:
   - **Efficient Monte Carlo approximation** for the distribution of default and prepayment in loan pools
   - **Asymptotically optimal portfolio** (AOP) for large-scale optimization of loan portfolios
2. Numerical tests with actual mortgage data: Monte Carlo approximation is typically several orders of magnitude faster than brute-force simulation (at a similar level of accuracy).
3. Similar computational advantages using AOP.

Model Framework

The probability that the n-th loan transitions from its state U^n_{t-1} at time $t-1$ to state u at time t:

$$\mathbb{P}_\theta[U^n_t = u|\mathcal{F}_{t-1}] = h_\theta(u, U^n_{t-1}, Y^n, V_{t'<t}, H^N_{t'<t})$$

- $U^n_t \in \mathcal{U}$ is the state of the n-th mortgage at time t (e.g., default, prepaid, or outstanding).
- $Y^n \in \mathcal{Y}$ are loan-level features of the n-th mortgage (FICO score, LTV ratio, geographic location, etc.).
- V_t is a vector of common factors (such as national mortgage rate and unemployment rate).
- "Mean-field" process $H^N_t = \frac{1}{N}\sum_{n=1}^{N} f(U^n_t, Y^n)$ (e.g., contagion)

Computational Challenges

- Goal: once a model has been fitted, analyze risk for a pool of N loans
- The distribution of default and prepayment rates in the pool can be found via **brute-force simulation** of loans $1, \ldots, N$.
- For large pools of loans, brute-force simulation is very computationally expensive!
  - **hours, days, or even weeks**

Some Previous Literature

- Parallel computing, e.g., Stein et al. (2007)
- Top-down models, e.g., Fermanian (2008)
- Top-down models with pool-level characteristics, e.g., Roll (1989)
- Limiting laws for default timing models
  - Bush, Hambly, Haworth, Jin, and Reisinger (2011)
  - Cvitanic, Ma, and Zhang (2012)
  - Giesecke, Spiliopoulos, Sowers, and Sirignano (2012)
  - Spiliopoulos, Sirignano, and Giesecke (2014)

Law of Large Numbers

Define the empirical measure:

$$\mu_t^N = \frac{1}{N}\sum_{n=1}^{N}\delta_{(U_t^n, Y^n)}.$$

Theorem

The empirical measure $\mu^N \xrightarrow{d} \bar{\mu}$ *as* $N \longrightarrow \infty$, *where* $\bar{\mu}$ *satisfies:*

$$\bar{\mu}_t(u, dy) = \sum_{u' \in \mathcal{U}} h_\theta(u, u', y, V_{t'<t}, \bar{H}_{t'<t})\bar{\mu}_{t-1}(u', dy),$$

and $\bar{H}_t = \sum_{u \in \mathcal{U}} \int_{\mathbb{R}^{d_Y}} f(u, y)\bar{\mu}_t(u, dy).$

Central Limit Theorem

Define the empirical fluctuation measure:

$$\Xi_t^N = \sqrt{N}(\mu_t^N - \bar{\mu}_t)$$

Theorem

$\Xi^N \xrightarrow{d} \bar{\Xi}$ as $N \longrightarrow \infty$, where $\bar{\Xi}$ satisfies:

$$\begin{aligned}
\bar{\Xi}_t(u, dy) &= \sum_{u' \in \mathcal{U}} h_\theta(u, u', y, V_{t'<t}, \bar{H}_{t'<t}) \bar{\Xi}_{t-1}(u', dy) \\
&+ \sum_{u' \in \mathcal{U}} \Big(\frac{\partial h_\theta}{\partial H}(u, u', y, V_{t'<t}, \bar{H}_{t'<t}) \cdot \bar{E}_{t'<t}\Big) \bar{\mu}_{t-1}(u', dy) \\
&+ \bar{\mathcal{M}}_t(u, dy),
\end{aligned}$$

where $\bar{E}_t = \sum_{u \in \mathcal{U}} \int_{\mathbb{R}^{d_Y}} f(u, y) \bar{\Xi}_t(u, dy)$.

Large Pool Approximation

- The law of large numbers and central limit theorem can be combined to form an approximation for a finite pool of N mortgages:

$$\mu^N(u, dy) \approx \bar{\mu}(u, dy) + \frac{1}{\sqrt{N}} \bar{\Xi}(u, dy).$$

- The approximation is conditionally Gaussian
  - easy to simulate
- Problem: curse of dimensionality when $\mathcal{Y}$ is high-dimensional.
  - sparse grids
  - low-dimensional transformation

Loan-level Data

1. Freddie Mac data set
   - 16 million prime mortgages
   - loan-level data
2. RMBS data set
   - 10 million subprime mortgages backing over 6000 MBSs
   - loan-level data
   - unique identifier for each MBS

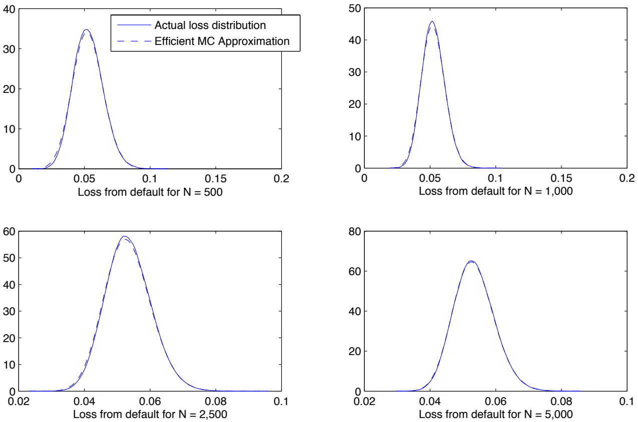
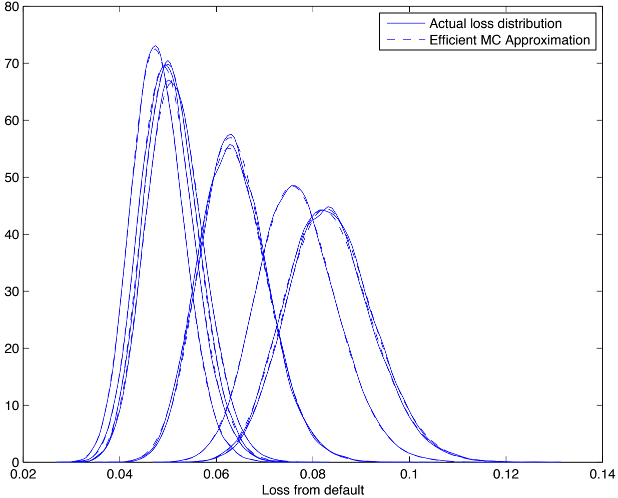


Figure : Comparison of actual distribution with approximate distribution (using both LLN and CLT). Loss reported as fraction of pool which defaulted. The horizon is 12 months.

Ten Actual MBS pools



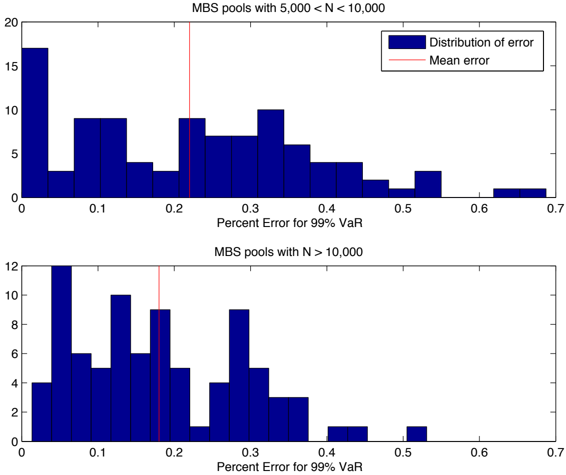


Figure : Distribution of error for 99% VaR from the efficient Monte Carlo approximation across 185 actual MBS pools. The time horizon is 12 months.

One-year time horizon

N	Time for Brute-force Simulation	Time for Approximation
1,000	44.3 seconds	2.7 seconds
5,000	2.6 minutes	2.7 seconds
10,000	4.6 minutes	2.7 seconds
25,000	10.1 minutes	2.7 seconds
100,000	47.5 minutes	2.7 seconds
1,000,000	7.9 hours	2.7 seconds
10,000,000	79.3 hours	2.7 seconds

Table : Comparison of computational times (seconds) for efficient Monte Carlo approximation and brute-force Monte Carlo simulation of the pool. 1-year time horizon.

Exploit weak convergence results for optimization

- How to optimally select N loans for a portfolio?
- Optimal selection of a loan portfolio is a **high-dimensional nonlinear integer program**.
  - High-dimensional: N can be large
  - Can only choose 0 or 1 of a loan $\rightarrow$ **integer program**
  - Objective and constraint functions are nonlinear (possibly nonconvex).
  - Objective and constraint functions can be computationally expensive to evaluate!

- Empirical measure of the loans: $\mu^N = (\mu^N_t)_{t=1,\dots,T}$
- "Performance measure": $R^N_P = f(\mu^N, V)$
  - Example: R^N_P is the return of the portfolio P of N loans.
- Optimization problem:

$$\begin{aligned} P^{N,*} \quad &= \quad \arg\min_{P^N} \mathbb{E}[g(R^N_{P^N})] \\ \text{s.t.} \quad & \quad \mathbb{E}[\phi(R^N_{P^N})] \geq c, \\ & \quad q(P^N) \leq d. \end{aligned}$$

where $P^N = (y^1, \dots, y^N) \in \mathcal{Y}$.

1. Approximate empirical measure of the loans:

$$\mu^N = (\mu_t^N)_{t=1,\ldots,T} \overset{d}{\approx} \bar{\mu}^N = (\bar{\mu}_t^N)_{t=1,\ldots,T}$$

2. Approximate "performance measure":

$$R_P^N = f(\mu^N, V) \overset{d}{\approx} \bar{R}_P^N = f(\bar{\mu}^N, V)$$

3. The portfolio choice can be equivalently be written as:

$$P^N = \frac{1}{N}\sum_{n=1}^{N} \delta_{y^n}$$

1. "True" optimal portfolio:

$$\begin{aligned} P^{N,*} &= \arg\min_{P^N} \mathbb{E}[g(R^N_{P^N})] \\ \text{s.t.} &\quad \mathbb{E}[\phi(R^N_{P^N})] \geq c, \\ &\quad q(P^N) \leq d. \end{aligned}$$

2. **Asymptotically optimal portfolio (AOP)**:

$$\begin{aligned} \bar{P}^{N,*} &= \arg\min_{P \in \mathcal{M}(\mathcal{Y})} \mathbb{E}[g(\bar{R}^N_P)], \\ \text{s.t.} &\quad \mathbb{E}[\phi(\bar{R}^N_P)] \geq c, \\ &\quad q(P) \leq d. \end{aligned}$$

Theorem

The asymptotically optimal portfolio $\bar{P}^{N,*}$ converges to the true optimal portfolio $P^{N,*}$ as $N \to \infty$ in $\big(\mathcal{M}(\mathcal{Y}), \pi\big)$, where π is the Prokhorov metric.

$$\pi(P^{N,*}, \bar{P}^{N,*}) \underset{N\to\infty}{\longrightarrow} 0.$$

Computational Performance of AOP

- Instead of solving a very computationally challenging nonlinear integer program, solve for the asymptotically optimal portfolio (AOP)!
- Compare computational performance of AOP with integer program solvers
  - MBS equity tranche
  - Mean-variance portfolio
  - Log-optimal portfolio

Select $N = 250$ loans out of a pool of $N_p = 1000$ loans for portfolio which maximizes expected return of MBS equity tranche:

Solver	Time	Exitflag	True objective	Agreement with AOP
Integer Program	35 min	Maxtime	.05949	99.4 %
AOP	1 s	Min stepsize	.05952	100 %

Table : Performance comparison between integer program solvers and AOP. One-period model.

Select $N = 2,500$ loans out of a pool of $N_p = 10,000$ loans for portfolio which maximizes expected return of MBS equity tranche:

Solver	Time	Exitflag	True objective	Agreement with AOP
Integer Program	5.4 hours	Maxtime	.05856	99.4 %
AOP	1 s	Min stepsize	.05857	100 %

Table : Performance comparison between integer program solvers and AOP.

Select $N = 250$ out of a pool of $N_p = 1,000$ loans for mean variance portfolio:

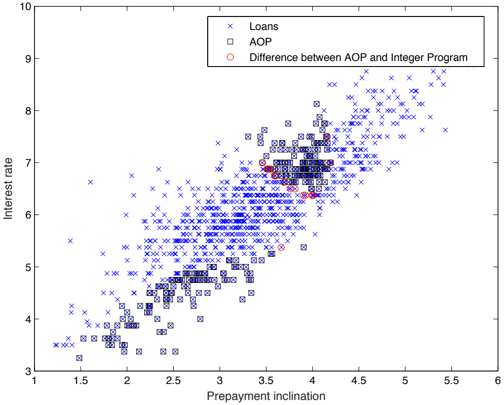


Figure : AOP computational time: 29 seconds. Integer program computational time: 1 hr 28 min. Solutions differ on 18/1000 loans.

Select 250 loans from 1,000 subprime mortgages for log-optimal portfolio:

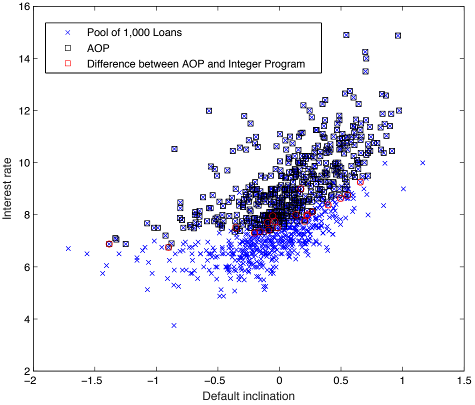


Figure : AOP computational time: 34 seconds. Integer program computational time: 39 min. Solutions differ on 20/1000 loans.

Summary

1. Prove weak convergence results for a broad class of models:
   - **Efficient Monte Carlo approximation** for the distribution of default and prepayment in loan pools
   - **Asymptotically optimal portfolio** (AOP) for large-scale optimization of loan portfolios
2. Numerical tests with actual mortgage data: Monte Carlo approximation is typically several orders of magnitude faster than brute-force simulation (at a similar level of accuracy).
3. Similar computational advantages using AOP.