Lecture 3

Finance Project

Somesh Jha

Maximum Likelihood estimation (MLE)

- Assume that we have N_t mortgages at time t in the mortgage pool.

- The number of pre-payments at time (denoted by c_t) follows a Poisson distribution.

MLE (Contd)

- Formally, probability $P(c_t = k)$ that c_t is equal to k is:

$$\frac{e^{-\lambda(X_t,\beta)N_t}(\lambda(X_t,\beta)N_t)^k}{k}$$

- Check that the following is true:

$$\sum_{k=0}^{\infty} kP(c_t = k) \;=\; \lambda(X_t,\beta)N_t$$

Form of $\lambda(X_t, \beta)$

- Recall that $\lambda(X_t, \beta)$ has the following form:

$$\pi_0(t) \exp(\beta_1 \cdot rf7 + \beta_2 \cdot \ln(burnout) + \beta_3 \cdot season)$$

- Meaning of each covariate is given below:
  $rf7$ (Refinancing opportunities)
  $\pi_0(t)$ (Age)
  $season$ (seasonal)
  $burnout$ (burnout)

Problem

- Suppose we a have pool of mortgages that is *similar* to the pool underlying the MBS we are trying to price.

- We have historical data about this mortgage.

Problem (Contd)

- Want to find parameters $\beta_1, \beta_2, \beta_3$ that *best fit* this historical data.

- We will use a technique called Maximum Likelihood Estimation (or MLE) for this purpose.

Basic idea of MLE

- Assume a distribution (we assume Poisson distribution for the number of prepayments).

- Estimate the probability $f(\beta)$ of observing the historical data.

Basic idea of MLE (Contd)

- The *log likelihood function* (denoted by $\mathcal{L}(\theta)$) is $\log f(\beta)$.

- The parameters β are given by the solution to the following global-optimization problem:

$$\max_{\beta} \mathcal{L}(\beta)$$

MLE (Contd)

- Assume that we have historic pre-payment data for a pool of mortgages.

- Also assume that the number of prepayments at time t only depends on the number of mortgages in the pool at time t and is independent of the history.

MLE (Contd)

- History is given for times $1, 2, \cdots, T$, and the following things are given:
  - c_t (Number of pre-payments at time t).
  - N_t (number of mortgages remaining in the pool).
- T is the lifetime of the mortgage pool under consideration.

MLE (Contd)

- Probability $P(c_t)$ that the number of prepayments is c_t at time t is:

$$\frac{e^{-\lambda(X_t,\beta)N_t}(\lambda(X_t,\beta)N_t)^{c_t}}{c_t!}$$

- The probability of observing the entire history (using independence here) is:

$$f(\beta) \ = \ \prod_{t=1}^{T} P(c_t)$$

- Log likelihood function $\mathcal{L}(\beta)$ is:

$$\sum_{t=1}^{T}(c_t \ln(\lambda N_t) - \lambda N_t - \ln(c_t!))$$

- For notational convenience, In the expression for $\mathcal{L}(\beta)$ I have suppressed X_t and β.

MLE (contd)

- The factor $\ln(c_t!)$ is a constant so we ignore it in the maximization problem.

- We have to maximize the following function with respect to β (I have suppressed the X_t and β factors for notational convenience):

$$\sum_{t=1}^{T}(c_t \ln(\lambda N_t) - \lambda N_t)$$

- Next we discuss a method for maximization.

Steepest Ascent

- Let $\mathcal{L}(\beta)$ be the log likelihood function.

- Recall that β is vector of three parameters $(\beta_1, \beta_2, \beta_3)$.

- The *gradient vector* of the log likelihood function is (denoted by $\frac{\partial \mathcal{L}(\beta)}{\partial \beta}$) is:

$$\begin{bmatrix} \frac{\partial \mathcal{L}}{\beta_1} \\ \frac{\partial \mathcal{L}}{\beta_2} \\ \frac{\partial \mathcal{L}}{\beta_3} \end{bmatrix}$$

- Intuitively, the gradient vector at β (denoted by $g(\beta)$) is the direction in which the log likelihood function increases most steeply (at the point β).

Steepest Ascent (Contd)

- Choose an initial vector β_0.
- Let β_{i-1} be the old estimate. The new estimate β_i is given by the following equations:

$$\begin{aligned} \beta_i - \beta_{i-1} &= cg(\beta_{i-1}) \\ \|\beta_i - \beta_{i-1}\| &= k \end{aligned}$$

- k is the step size. Notice that c is determined by the equations given above. Norm of the vector $\beta_i - \beta_{i-1}$ is denoted by $\|\beta_i - \beta_{i-1}\|$.

Problems with Steepest Ascent

- Convergence very slow near a local maximum.
- Variety of methods for numerical optimization.
- Judge, George G., William E. Griffiths, R. Carter Hill, and Tsoung-Chao Lee. 1980. *The Theory and Practice of Econometrics*. New York: Wiley.
- Quandt Richard E. 1983. "Computational Problems and Methods," in Zvi Griliches and Michael D. Intriligator editors., *Handbook of Econometrics*, Vol 1. Amsterdam: North-Holland.

Interesting exercise

- Assume that stock prices follow the *geometric brownian motion*.

- Using MLE estimate the drift and volatility of the stock.

- Historical prices for many stocks are available on numerous web-sites.

Interesting exercise (Contd)

- Using prices of various options on the stock find the *implied volatility curve.*

- How far is the *implied volatility curve* away from the MLE estimate?

- Let me know if you try this exercise.

Summary of MBS cash-flows

- MP_t (mortgage payment at time t)
- I_t (interest payment at time t)
- P_t (principal payment at time t)
- PP_t (prepayment at time t)
- S_t (service charge at time t)
- NI_t (net interest rate at time t)
- MB_t (mortgage balance at time t)
- CF_t (cash-flow at time t)
- SMM_t (Single Monthly Mortality Rate at time t)

Summary (Contd)

- MP_t is equal to

$$MB_{t-1}\frac{c(1+c)^{n-t+1}}{(1+c)^{n-t+1}-1}$$

- I_t, S_t, P_t, and NI_t follow the equations given below:

$$\begin{aligned} I_t &= cMB_{t-1} \\ S_t &= sMB_{t-1} \\ P_t &= MP_t - I_t \\ NI_t &= I_t - S_t \end{aligned}$$

- MB_t is given by the following expression:

$$MB_{t-1} - P_t - PP_t$$

Summary (Contd)

- CF_t is given by the following formula:

$$NI_t + P_t + PP_t$$

- SMM_t is obtained from the pre-payment model.

- Prepayment PP_t at time t is given by the following equation:

$$SMM_t(MB_{t-1} - S_t)$$

Pass-throughs

- Suppose a pass-through owns x percent of the mortgage pool.

- Cash flow of the pass-through at time t is given by the following equation:

$$\frac{CF_t x}{100}$$

CMOs

- Suppose there are m tranches $T_1, \cdots, T_m$ with par-values $P_1, \cdots, P_m$.
- At time t let the remaining par-value of tranch T_i be P_i^t.
- Let j be the least number such that T_j is not retired.
- The cash-flow of that tranch is:

$$\frac{P_j^{t-1}}{MB_{t-1}} I_t + P_t + PP_t$$

CMOs(Contd)

- The new par-value P_j^t of tranch T_j is:
$$P_j^{t-1} - P_t - PP_t$$
- If P_j^{t-1} is equal to zero, *retire* the tranch T_j.

- For all tranches T_i such that $i > j$ the cash-flow is
$$\frac{P_i^{t-1}}{MB_{t-1}} I_t$$
- P_i^t is equal to P_i^{t-1} (Why?)

Stripped MBSs

- The PO class gets $P_t + PP_t$ minus the servicing fee.

- The IO class gets I_t minus the servicing fee.

High-Level Design Document

- *Query Phase*
  Describes the steps in which the user interacts with the system. User chooses what instrument he/she wants to price and the various parameters.

- *Computation Phase*
  High-level procedure to price these instruments. Provide a description of the general technique you are using (induction on lattices, simulation, finite-difference schemes).

- *Presentation Phase*
  What is the result presented to the user.

How is the result presented to the user.

Query Phase

- Ask the user what kind of MBS they need to price.
- Pass-throughs, CMOs, or Stripped MBSs.
- Ask the parameters of the mortgage pool associated with the MBS (for description of parameters please see Lecture 1).
- In case of CMOs ask the following questions:
  - Number of tranches.
  - Par-value of each tranch.

Query Phase (Contd)

- Ask user about prepayment models.
  Support two kind of prepayment models.

- **Prepayment Option A**
  Vector of PSA speeds (see page 41 Lecture 2).

- **Prepayment Option B**
  Poisson process based model (see page 46 Lecture 2).
  Assumption: Assume that the model has been calibrated.

Points to notice

- Notice that I haven't mentioned many details (like how the interface will look to the user).

- Details belong in the low-level design document.

- Low-level design document will *refine* each step in the high-level design document.

Points to notice (Contd)

- Haven't committed to technology or methodology.
- I haven't said whether we are going to use `JAVA`, `C++`.
- Technology choice made after the high-level design document.
- Haven't even said whether we are going to use *object-oriented*, *imperative*, or *functional* programming.
- These decisions will be made after the high-level design document.

Check for completeness

- Check that all the parameters you need to price the instruments are there.

- Nothing should be missing.

Computation phase

- We will split this phase into two phases.
- Determine which prepayment option the user has given.
- Depending on the prepayment option the algorithm is very different.

Why the splitting?

- There is a much more efficient algorithm to price MBSs in case of prepayment option A.

- For example, you would not use Hull-White method (paper 1) to price a lookback option.

- The lattice for pricing a lookback option is *small* (refer back to data-structures notes).

Pricing pass-throughs (Option A)

- Notice that the cash-flow of the pass-through in this case is deterministic.

- Let CF_t be the cash-flow of the pass-through at time t.

- No randomness in CF_t.

Pass-throughs (Option A)

- The price of the pass-through at time t is given by the following equations:

$$\Sigma_{t=1}^{T} E[CF_t \Pi_{j=0}^{t-1} \frac{1}{1+r_j}]$$
$$\Sigma_{t=1}^{T} CF_t E[\Pi_{j=0}^{t-1} \frac{1}{1+r_j}]$$

- Expectation taken with respect to the risk-neutral or martingale measure. T is the lifetime of the mortgage pool underlying the pass-through.
- Do you recognize the following quantity?

$$E[\prod_{j=0}^{t-1} \frac{1}{1+r_j}]$$

Pass-through (Option A)

- The mystery expression is the price at time 0 of a zero-coupon bond paying one dollar at time t.

- So we have the following formula for valuing the pass-through security in case of option A:

$$\sum_{t=1}^{T} CF_t P(0,t)$$

- $P(0,t)$ is the price of a zero-coupon bond (at time 0) paying one dollar at time t.

Pass-through (Option A)

- Assuming prices $P(0,1), P(0,2), \cdots, P(0,T)$ are observable from the market **`we are done`**.

- Suppose the prices are only known for some times $t_1, t_2, \cdots, t_k$.

- Find the missing prices using interpolation.

- Notice how fast the algorithm is. No simulation required.

CMOs (option A)

- Price each tranch separately.

- Find out the lifetime τ_i of each tranch T_i (when it retires).

- Use the formula given below for tranch T_i

$$\sum_{t=1}^{\tau_i} CF_{t,i} P(0,t)$$

- $CF_{t,i}$ is the cash-flow of tranch T_i at time t.

Stripped MBS (option A)

- Price PO and IO classes separately.

- Use the equation given before.

Pass-throughs (option B)

- Use Monte-Carlo simulation to price the MBSs.
- Assume that we have a procedure called **`nextPath()`** which generates a random path.
- Notice that nothing is said about the specific interest-rate model. That belongs in the low-level design document.
- High-level design document only describes high-level algorithms and techniques. Very little detail about the actual

implementation.

Pass-throughs (option B)

- Determine how many paths to generate (say N).

- Let π_i be the i-th path and $r_{t,i}$ be the short-rate at time t on path i.

- Let $CF_{t,i}$ be the cash-flow on path π_i at time t.

- Let V_i be the value of this cash-flow at time 0 (given by the following equation)

$$\sum_{t=1}^{T} CF_{t,i} \prod_{j=0}^{t-1} \frac{1}{1 + r_{t,i}}$$

Pass-throughs (option B)

- Recall that the i-th path is π_i.

- Value of the pass-through at time 0 (denoted it by V_{PT}) is given by the following equation (averaging the values):

$$\frac{1}{N} \sum_{i=1}^{N} V_i$$

CMOs and Stripped MBSs

- Only the expression for cash-flows change.

- Everything remains the same.

Monte-Carlo (Contd)

- Suppose we generate N paths in the Monte-Carlo simulation.

- Let ω be the standard deviation of the value of the financial instrument calculated from the simulation runs.

- The error of the estimate calculated from the simulation runs is approximately $\frac{\omega}{\sqrt{N}}$.

Variance reduction

- There are techniques to *speed up* the convergence of the Monte-carlo simulations.

- One of such class of techniques is called *variance reduction*.

- We will consider a special case of variance reduction called *control variate technique*.

Control variate technique

- Security A is the security to be priced.

- Consider a similar security B, which you can price by other means (say lattice based or analytical techniques).

- In the simulation runs estimate the quantity $V(A) - V(B)$.

- $V(A)$ and $V(B)$ are the values of securities A and B respectively.

Control-variate technique (Contd)

- Let $V^\star$ be the estimate of $V(A) - V(B)$ calculated from the simulation.

- Let $V_{true}(B)$ be the value of security B calculated using other means (lattice based or analytical).

- The estimate for value of security A is $V^\star + V_{true}(B)$.

Interesting exercise

- This is an interesting exercise (especially for students doing Paper 1).
- Suppose we are interested in pricing an *european asian option* with maturity T and strike price K.
- European asian option is the primary security A in this case.
- Take your secondary security B as the european geometric asian option with exactly the same parameters.
- Recall that geometric asian option depends upon the geometric average of the stock price.

Interesting Exercise (Contd)

- Price the european geometric option using lattice based techniques. Call this price $V(G)_{true}$.
- Recall that the lattice for pricing european geometric option was cubic in the number of periods.
- Estimate the difference of the asian option and the geometric option. Call this estimate $V^\star$.
- The value of the asian option is $V^\star + V(G)_{true}$.

MBS and variance reduction

- What is a security similar to an MBS?
  `Another MBS`.

- Let us say we want to price an MBS A.

- We will pick a similar MBS B.

- MBS B will have *deterministic cash-flows* and hence can be priced using the closed-form formula given before. No simulation required.

- Next we describe how to pick MBS B.

In search of MBS B

- Suppose the mortgages in the mortgage pool are for T months.
- Pick r times $t_0 = 1 \leq t_1 < t_2 < \cdots < t_r = T$.
- Let SMM_i (for $0 \leq i < r$) be the SMM for the period $[t_i, t_{i+1})$.
- *Goal:* To pick $SMM_0, \cdots, SMM_{r-1}$ so that the prepayment structure of MBS B is *close* to the prepayment structure of MBS A (our original MBS).

Search continues

- Generate M random paths.

- For path i let $SMM_{1,i}, SMM_{2,i}, \cdots, SMM_{T,i}$ be the sequence of SMMs for the original MBS (security A).

- Calculate the distance between the sequence of SMMs and the SMMs of the security B.

Search continues

- The distance is given by the following equation:

$$d_i = \sum_{t=1}^{T} |SMM_{t,i} - SMM_{t,B}|$$

- $SMM_{t,B}$ is the SMM for the security B we are trying to construct.

- d_i is a function of the variables $SMM_0, \cdots, SMM_{r-1}$.

Search ends

- Add the distances over all the M paths

$$D = \sum_{i=1}^{M} d_i$$

- Find variables $SMM_0, \cdots, SMM_{r-1}$ by solving the following global optimization problem:

$$\max_{SMM_0, \cdots, SMM_{r-1}} D(SMM_0, \cdots, SMM_{r-1})$$

Presentation Phase

- Present the price of the pass-through to the user.
- In case of CMOs present the price of each tranch.
- In case of stripped MBS present the price of PO and IO classes.
- Report any convergence problems, i.e., Monte-carlo simulation didn't converge in the required number of steps.

Schedule

- High-level document due date: Feb 3, 1999, Wednesday.

- No need to divide into sub-teams for this document.

- Pay close attention to the points suggested.