R Tools for Understanding Credit Risk Modelling

QRM: Concepts, Techniques & Tools

Alexander J. McNeil
Heriot-Watt University, Edinburgh

Chicago, 30 May 2015
Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
User of the S language since circa 1990.

**EVIS (1997).** Extreme Values in S-Plus. A library that was later integrated into S+FinMetrics.


**QRM.** R package maintained by Bernhard Pfaff based on QRMlib. Additional listed authors: Marius Hofert, AJM and Scott Ulmann.

Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
Scores, Ratings & Measures Inferred from Prices

Broadly speaking, there are two philosophies for quantifying the credit quality or default probability of an obligor.

1. Credit quality can be described by a credit rating or score that is based on empirical data describing the borrowing and repayment history and other characteristics of the obligor, or similar obligors.

2. For obligors whose equity is traded on financial markets, prices can be used to infer the market’s view of the credit quality of the obligor.
Rating Migration as a Markov Chain

- Let \((R_t)\) denote a continuous-time stochastic process taking values in the set \(S = \{D, C, B3, B2, B1, A3, A2, A1\}\).
- We assume that \((R_t)\) is a Markov chain so that migration probabilities depend only on the current rating.
- However, there is evidence that empirical rating histories show momentum and stickiness (Lando and Skodeberg, 2002).
- Transition probabilities are summarized by a generator matrix \(\Lambda = (\lambda_{jk})\). Over any small time step of duration \(\delta t\) the probability of a transition from rating \(j\) to \(k\) is given approximately by \(\lambda_{jk}\delta t\). The probability of staying at rating \(j\) is given by \(1 - \sum_{k \neq j} \lambda_{jk}\delta t\).
- Let \(P(t)\) be the matrix of transition probabilities for the period \([0, t]\).
- Using the so-called matrix exponential we have

\[P(t) = \exp(\Lambda t)\]
A Markov chain with generator $\Lambda$ can be constructed in the following way. An obligor remains in rating state $j$ for an exponentially distributed amount of time with parameter $\lambda = \sum_{k \neq j} \lambda_{jk}$. When a transition takes place the probability that it is from $j$ to state $k$ is given by $\lambda_{jk}/\lambda$.

This construction leads to natural estimators for the matrix $\Lambda$.

Since $\lambda_{jk}$ is the instantaneous rate of migrating from $j$ to $k$ we can estimate it by

$$\hat{\lambda}_{jk} = \frac{N_{jk}(T)}{\int_0^T Y_j(t)dt},$$

where $N_{jk}(T)$ is the total number of observed transitions from $j$ to $k$ over the time period $[0, T]$ and $Y_j(t)$ is the number of obligors with rating $j$ at time $t$.

The denominator represents the total time spent in state $j$ by all the companies in the dataset.

This is in fact the maximum likelihood estimator.
Illustration of Data

We start with 20 years of rating migration data:

```r
head(RatingEvents, n=6)
```

```
## id starttime endtime startrating endrating time
## 1 1 0.00000 6.16813 B2 B3 6.1681297
## 2 1 6.16813 16.32281 B3 C 10.1546802
## 3 1 16.32281 17.28772 C D 0.9649059
## 4 2 0.00000 10.10462 A1 A2 10.1046193
## 5 2 10.10462 12.67101 A2 B2 2.5663897
## 6 2 12.67101 15.49927 B2 B3 2.8282614
```

```r
(Njktable = table(RatingEvents$startrating, RatingEvents$endrating))
```

```
## A1 A2 A3 B1 B2 B3 C D
## A1 141 405 30 12 2 0 0 0
## A2  86 504 996 41 5 4 0 0
## A3  17 402 954 1167 49 22 0 0
## B1  3 41 781 843 1198 137 13 19
## B2  2 10 48 744 475 1411 96 39
## B3  0 9 25 30 555 376 1055 305
## C  2 0 12 6 32 268 68 1276
```
Implementation of Estimator

# Load library for matrix exponential
require(Matrix)

# compute total time spent in each rating state
RiskSet <- by(RatingEvents$time,RatingEvents$startrating,sum)
Jlevels <- levels(RatingEvents$startrating)
Klevels <- levels(RatingEvents$endrating)
Njmatrix <- matrix(nrow=length(Jlevels),ncol=length(Klevels),
                   as.numeric(RiskSet),byrow=FALSE)

# basic form of estimator
Lambda.hat <- Njktable/Njmatrix

# complete matrix by adding default row and fixing diagonal
D <- rep(0,dim(Lambda.hat)[2])
Lambda.hat <- rbind(Lambda.hat,D)
diag(Lambda.hat) <- D
rowsums <- apply(Lambda.hat,1,sum)
diag(Lambda.hat) <- -rowsums

# compute estimated transition probabilities
P.hat <- expm(Lambda.hat)
Estimated Annual Transition Probabilities

Results:

\[
P\hat{\text{.}}
\]

## 8 x 8 Matrix of class "dgeMatrix"

<table>
<thead>
<tr>
<th></th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>B1</th>
<th>B2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.922826</td>
<td>0.066737</td>
<td>0.007698</td>
<td>0.002284</td>
<td>0.000406</td>
</tr>
<tr>
<td>A2</td>
<td>0.064773</td>
<td>0.912801</td>
<td>0.074355</td>
<td>0.005382</td>
<td>0.000609</td>
</tr>
<tr>
<td>A3</td>
<td>0.009292</td>
<td>0.020157</td>
<td>0.915501</td>
<td>0.057602</td>
<td>0.004201</td>
</tr>
<tr>
<td>B1</td>
<td>0.001786</td>
<td>0.002460</td>
<td>0.038446</td>
<td>0.890662</td>
<td>0.056303</td>
</tr>
<tr>
<td>B2</td>
<td>0.001496</td>
<td>0.000828</td>
<td>0.004514</td>
<td>0.050088</td>
<td>0.837308</td>
</tr>
<tr>
<td>B3</td>
<td>0.000643</td>
<td>0.000826</td>
<td>0.002585</td>
<td>0.004107</td>
<td>0.046398</td>
</tr>
<tr>
<td>C</td>
<td>0.000006</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>D</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>1.000000</td>
</tr>
</tbody>
</table>
Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
Assumptions in Merton’s Model (1974)

- Consider firm with stochastic asset-value \( V_t \), financing itself by equity (i.e. by issuing shares) and debt.
- Assume that debt consists of single zero coupon bond with face or nominal value \( B \) and maturity \( T \).
- Denote by \( S_t \) and \( B_t \) the value at time \( t \leq T \) of equity and debt so that

\[
V_t = S_t + B_t, \quad 0 \leq t \leq T.
\]

- Assume that default occurs if the firm misses a payment to its debt holders and hence only at \( T \).
- At \( T \) we have two possible cases:
  1. \( V_T > B \). In that case the debtholders receive \( B \); shareholders receive residual value \( S_T = V_T - B \), and there is no default.
  2. \( V_T \leq B \). In that case the firm cannot meet its financial obligations, and shareholders hand over control to the bondholders, who liquidate the firm; hence we have \( B_T = V_T, S_T = 0 \).
In summary we obtain

\[ S_T = (V_T - B)^+ \]
\[ B_T = \min(V_T, B) = B - (B - V_T)^+ . \]

- The value of equity at \( T \) equals the pay-off of a European call option on \( V_T \) with exercise price equal to \( B \).
- The value of the debt at \( T \) equals the nominal value of debt minus the pay-off of a European put option on \( V_T \).
Generating Merton Model

It is assumed that asset value \((V_t)\) follows a diffusion of the form

\[
dV_t = \mu_V V_t dt + \sigma_V V_t dW_t
\]

for constants \(\mu_V \in \mathbb{R}, \sigma_V > 0\), and a Brownian motion \((W_t)_{t \geq 0}\), so that

\[
V_T = V_0 \exp\left((\mu_V - \frac{1}{2} \sigma_V^2)T + \sigma_V W_T\right).
\]

```r
require(sde)
## Parameters for Merton model
V0 <- 1; muV <- 0.03; sigmaV <- 0.25
r <- 0.02; B <- 0.85; T <- 1
N <- 364

## Simulated asset value trajectories for Merton model
npaths <- 50
paths <- matrix(NA, nrow=N+1, ncol=npaths)
for (i in 1:npaths)
{paths[,i] <- GBM(x=V0, r=muV, sigma=sigmaV, T=T, N=N)}
```
Generating Merton Model (cont.)
Pricing of Equity and Debt

- Since equity is a call option on the asset value \((V_t)\) the Black–Scholes formula yields

\[
S_t = C^{BS}(t, V_t; \sigma_V, r, T, B) := V_t \Phi(d_{t,1}) - Be^{-r(T-t)}\Phi(d_{t,2}),
\]

where the arguments are given by

\[
d_{t,1} = \frac{\ln \frac{V_t}{B} + (r + \frac{1}{2}\sigma_V^2)(T - t)}{\sigma_V \sqrt{T - t}}, \quad d_{t,2} = d_{t,1} - \sigma_V \sqrt{T - t}.
\]

- The price at \(t \leq T\) of a default-free zero-coupon bond with maturity \(T\) and a face value of one equals

\[
p_0(t, T) = \exp(-r(T - t)).
\]

- The value of the firm’s debt equals the difference between the value of default-free debt and a put option on \((V_t)\) with strike \(B\), i.e.

\[
B_t = Bp_0(t, T) - P^{BS}(t, V_t; r, \sigma_V, B, T) = p_0(t, T)B\Phi(d_{t,2}) + V_t\Phi(-d_{t,1})
\]
Draw a path of Merton GBM leading to default and superimpose value of default-free and defaultable debt.

```r
source("Black_Scholes.R")
set.seed(63)
Vt <- GBM(x=V0, r=muV, sigma=sigmaV, T=T, N=N)

times <- seq(from=0, to=1, length=N+1)
par(mar=c(3,3,2,1), mgp=c(2,1,0))
plot(times, Vt, type="l", ylim=range(0.6, max(Vt)),
     xlab="t", ylab=expression(V[t]))
abline(v=1)

# Add default free debt value as green line
lines(times, B*exp(-r*(T-times)), col=3)

# Add defaultable debt as red line
Bt <- B*exp(-r*(T-times)) - BlackScholes(times, Vt, r, sigmaV, B, T, type="put")
lines(times, Bt, col=2)
```
Default Path (cont.)
Merton's Model

Non-Default Path

©2015 (QRM Tutorial) Alexander J. McNeil R/Finance Chicago
Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
There are a number of industry models that descend from the Merton model.

An important example is the so-called public-firm EDF model that is maintained by Moody’s Analytics.

The methodology builds on earlier work by KMV (a private company named after its founders Kealhofer, McQuown and Vasicek) in the 1990s, and is also known as the KMV approach.


**Expected Default Frequency.** The EDF is an estimate of the default probability of a given firm over a one-year horizon.
Moody’s Public-Firm EDF Model

- Suppose we use Merton’s model for a company issuing debt with face value $B$ maturing at time $T = 1$. The EDF would be

$$\text{EDF}_{Merton} = 1 - \Phi \left( \frac{\ln V_0 - \ln B + (\mu V - \frac{1}{2} \sigma_V^2)}{\sigma_V} \right).$$

- The public-firm EDF model uses a formula of the form

$$\text{EDF} = g(\text{DD}) = g \left( \frac{\ln V_0 - \ln \tilde{B}}{\sigma_V} \right)$$

where

- $g$ is an empirically estimated function;
- the default point $\tilde{B}$ represents the structure of the firm’s liabilities more closely;
- the current asset value $V_0$ and the asset volatility $\sigma_V$ are inferred (or ‘backed out’) from information about the firm’s equity value.
Illustration of Public-Firm EDF Approach

<table>
<thead>
<tr>
<th>Variable</th>
<th>J&amp;J</th>
<th>RadioShack</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Market value of assets</td>
<td>V₀</td>
<td>V₁</td>
<td>Option pricing approach.</td>
</tr>
<tr>
<td>Asset volatility σ₉</td>
<td>11%</td>
<td>24%</td>
<td>Option pricing approach.</td>
</tr>
<tr>
<td>Default threshold Ê</td>
<td>39 bn</td>
<td>1042 m</td>
<td>Short-term liabilities and half of long-term liabilities.</td>
</tr>
<tr>
<td>DD</td>
<td>16.4</td>
<td>2.3</td>
<td>Given by (log V₀ − log Ê)/σ₉.</td>
</tr>
<tr>
<td>EDF (one year)</td>
<td>0.01%</td>
<td>3.58%</td>
<td>Empirical mapping g.</td>
</tr>
</tbody>
</table>

This example is taken from Sun, Munves, and Hamilton, 2012 and concerns the situations of Johnson and Johnson (J&J) and RadioShack as of April 2012. All quantities are in USD.

\[ S_t = C^{BS}(t, V_t; \sigma_V, r, T, \tilde{B}) \]
RadioShack Example

```r
require(timeSeries)
load("RadioShack.RData")
source("Black_Scholes.R")

# Use 2010 to March 2012 (as in Moody's example)
RadioShack = window(RadioShack, start="2010-01-01", end="2012-03-31")

# Number of shares in millions (approximately)
N.shares = 100

# Value of equity in millions of dollars (approx)
Svalues = RadioShack*N.shares

# Equity vol
sigmaS <- as.numeric(sd returns(Svalues))*sqrt(250)

# Value of one-year debt in millions approximately
B = 1042

Vvalues = timeSeries(rep(NA, length(Svalues)), time(Svalues))

par(mar=c(3,3,2,1), mgp=c(2,1,0))
plot(Svalues, ylab="Equity")
```
Distance-to-Default Calculations

Iterative Solution for Asset Process

```r
rooteqn <- function(V,S,t,r,sigmaV,B,T)
{
  S - BlackScholes(t,V,r,sigmaV,B,T,"call")
}

# Initial estimate of volatility
sigmaV <- sigmaS
sigmaV.old <- 0
it = 1

# iterative solution for asset values
while (abs(sigmaV-sigmaV.old)/sigmaV.old > 0.000001)
{
  it = it + 1
  for (i in 1:length(Svalues)) {
    tmp = uniroot(rooteqn, interval =c(Svalues[i],10*Svalues[i]),
                  S=Svalues[i],t=0,r=0.03,sigmaV=sigmaV,B=B,T=1)
    Vvalues[i] = tmp$root
  }
  sigmaV.old = sigmaV
  sigmaV = as.numeric(sd RETURNS(Vvalues))*sqrt(250)
}
```

©2015 (QRM Tutorial)  Alexander J. McNeil  R/Finance Chicago  28 / 42
Distance-to-Default Calculations

Results

\[
EDF = g(DD) = g \left( \frac{\ln V_0 - \ln \tilde{B}}{\sigma_V} \right)
\]

```r
# results
DD <- (log(Vvalues[length(Vvalues)]) - log(B))/sigmaV
results<-c(it=it, sigmaS=sigmaS, V0=Vvalues[length(Vvalues)], sigmaV=sigmaV, DD=DD)
results
```

```
<table>
<thead>
<tr>
<th>#</th>
<th>it</th>
<th>sigmaS</th>
<th>V0</th>
<th>sigmaV</th>
<th>DD</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>6.0000000</td>
<td>0.4764729</td>
<td>1612.2850854</td>
<td>0.2613945</td>
<td>1.6699299</td>
</tr>
</tbody>
</table>
```

```r
# graphs
par(mar=c(3,3,2,1), mgp=c(2,1,0))
plot(Vvalues, ylim=range(Vvalues,Svalues), ylab="Values")
lines(Svalues, col=2)
legend(x=1265200000, y=1100, legend=c("Asset value","Equity value"),
      lty=c(1,1), col=c(1,2))
```
Results (cont.)

<table>
<thead>
<tr>
<th>Time</th>
<th>Asset value</th>
<th>Equity value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010-07-01</td>
<td>3000</td>
<td>3000</td>
</tr>
<tr>
<td>2011-07-01</td>
<td>1500</td>
<td>1500</td>
</tr>
<tr>
<td>2012-01-01</td>
<td>1000</td>
<td>1000</td>
</tr>
</tbody>
</table>
Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
Gaussian Firm-Value Models for Portfolio

Most industry portfolio models assume default occurs if a standard normal critical variable $X_i$ (with an asset value interpretation) is less than a critical threshold $d_i$. The dependence between defaults is modelled with a factor model:

$$X_i = \sqrt{\beta_i} \tilde{F}_i + \sqrt{1 - \beta_i} \varepsilon_i$$

$$= \sqrt{\beta_i} a'_i F + \sqrt{1 - \beta_i} \varepsilon_i$$

- $F \sim N_p(0, \Omega)$ is a random vector of normally distributed common economic factors with country-industry interpretations;
- $\varepsilon_i$ is a standard normally distributed error, which is independent of $F$ and of $\varepsilon_j$ for $j \neq i$;
- $0 < \beta_i < 1$; $\text{var}(\tilde{F}_i) = 1$;
- $a_i$ are factor weights.
In this model defaults are conditionally independent Bernoulli events given $F$ (or equivalently $\Psi = -F$) the systematic factors.

The conditional independence of defaults given $F$ follows immediately from the independence of the idiosyncratic terms $\varepsilon_1, \ldots, \varepsilon_m$. The conditional default probabilities are given by

$$p_i(\psi) = \Phi \left( \frac{\Phi^{-1}(p_i) + \sqrt{\beta_i} a_i' \psi}{\sqrt{1 - \beta_i}} \right),$$

where $p_i$ is the default probability, $a_i$ are the factor weights and $\beta_i$ is the systematic risk component for obligor $i$. 

©2015 (QRM Tutorial) Alexander J. McNeil R/Finance Chicago
Computational Advantages

- Let \( e_i \) denote the exposure, \( \delta_i \) the loss given default and \( Y_i \) the default indicator of obligor \( i \).
- It is difficult to compute the df \( F_L \) of the portfolio loss \( L = \sum_{i=1}^{m} e_i \delta_i Y_i \).
- However, it is easy to use the conditional independence of the defaults to show that the Laplace–Stieltjes transform of \( F_L \) is given by

\[
\hat{F}_L(t) = E(e^{-tL}) = E \left( E(e^{-t \sum_{i=1}^{m} e_i \delta_i Y_i} \mid \Psi) \right) \\
= E \left( \prod_{i=1}^{m} (p_i(\Psi) e^{-te_i \delta_i} + 1 - p_i(\Psi)) \right)
\]

which can be obtained by integrating over distribution of factors \( \Psi \).

- The Laplace–Stieltjes transform is useful for: sampling losses from model with importance sampling; approximating probability mass function using Fourier transform.
Loss Distribution

```r
laplace.transform <- function(t, pd, exposure, lgd = rep(1, length(exposure))) {
  output <- rep(NA, length(t))
  for (i in 1:length(t))
    output[i] <- exp(sum(log(1-pd*(1 - exp(-exposure*lgd*t[i])))))
  output
}
# no common factor for simplicity
m <- 20
exposure <- c(5,5,5,5,10,10,10,10,20,20,20,20,30,30,30,30,40,40,40,40)
pd <- c(rep(0.1,10), rep(0.05,10))
N <- sum(exposure)+1
t <- 2*pi*(0:(N-1))/N
cfunction <- laplace.transform(-t*(1i),pd,exposure)
par(mar=c(3,3,2,1),mgp=c(2,1,0))
plot(cfunction)

fft.out <- round(Re(fft(cfunction)),digits=20)
probs <- fft.out/N
barplot(probs[(0:20)*5+1],names.arg=paste((0:20)*5))
```
Loss Distribution (cont.)

©2015 (QRM Tutorial) Alexander J. McNeil R/Finance Chicago
Overview

1. R Credentials
2. Markov Chains for Rating Migrations
3. Merton’s Model
4. Distance-to-Default Calculations
5. Portfolio Loss Distributions with FFT
6. Estimation of Credit Risk Models from Default Data
Since defaults are sparse, industry models generally calibrate the factor model to equity return data (or asset return data).

Where actual historical default data are available these can also be used.

Recall that conditional default probabilities are given by

$$p_i(\psi) = \Phi \left( \frac{\Phi^{-1}(p_i) + \sqrt{\beta_i} a'_i \psi}{\sqrt{1 - \beta_i}} \right).$$

In a one-factor version of this model we have a model of the form:

$$p_i(\psi) = \Phi (\mu_i + \sigma_i \psi).$$

We often assume that the $\mu_i$ and $\sigma_i$ are identical for all obligors in a homogeneous group (for example a rating group).

This kind of model can be fitted to historical default data as a so-called GLMM (generalized linear mixed model).
Default Data (from S&P)

<table>
<thead>
<tr>
<th>Year</th>
<th>CCC</th>
<th>B</th>
<th>BB</th>
<th>BBB</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>1985</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1990</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1995</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

©2015 (QRM Tutorial) Alexander J. McNeil R/Finance Chicago
Example

tail(data.frame(defaults, firms, year, rating), n=10)

<table>
<thead>
<tr>
<th>#</th>
<th>defaults</th>
<th>firms</th>
<th>year</th>
<th>rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>91</td>
<td>1</td>
<td>1208</td>
<td>1999</td>
<td>A</td>
</tr>
<tr>
<td>92</td>
<td>2</td>
<td>1085</td>
<td>1999</td>
<td>BBB</td>
</tr>
<tr>
<td>93</td>
<td>8</td>
<td>793</td>
<td>1999</td>
<td>BB</td>
</tr>
<tr>
<td>94</td>
<td>63</td>
<td>899</td>
<td>1999</td>
<td>B</td>
</tr>
<tr>
<td>95</td>
<td>22</td>
<td>73</td>
<td>1999</td>
<td>CCC</td>
</tr>
<tr>
<td>96</td>
<td>1</td>
<td>1215</td>
<td>2000</td>
<td>A</td>
</tr>
<tr>
<td>97</td>
<td>4</td>
<td>1157</td>
<td>2000</td>
<td>BBB</td>
</tr>
<tr>
<td>98</td>
<td>10</td>
<td>887</td>
<td>2000</td>
<td>BB</td>
</tr>
<tr>
<td>99</td>
<td>69</td>
<td>961</td>
<td>2000</td>
<td>B</td>
</tr>
<tr>
<td>100</td>
<td>25</td>
<td>86</td>
<td>2000</td>
<td>CCC</td>
</tr>
</tbody>
</table>

# Fit glmm
mod <- glmer(cbind(defaults, firms-defaults) ~ -1 + rating + (1|year), family=binomial(probit))
## Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]
## Family: binomial  ( probit )
## Formula: cbind(defaults, firms - defaults) ~ -1 + rating + (1 | year)
## AIC  BIC  logLik  deviance  df.resid
## 404.338 419.969 -196.169 392.338 94
## Random effects:
## Groups   Name        Std.Dev.
## year      (Intercept) 0.2415
## Number of obs: 100, groups: year, 20
## Fixed Effects:
## ratingA  ratingBBB  ratingBB  ratingB  ratingCCC
## -3.4318  -2.9185  -2.4039  -1.6895  -0.8378

sigma <- mod@theta; mu <- mod@beta
(beta <- sigma^2/(1+sigma^2)); (PD <- pnorm(mu * sqrt(1-beta)))

## [1] 0.05510481
## [1] 0.0004251567 0.0022776810 0.0097268556 0.0502693782 0.2077200911


