STATS 243P PROJECT
MODELING DEFAULT PROBABILITY IN SME

YIFAN WANG - SUID#06001408 WANGYF@STANFORD.EDU

1. Introduction

The small and medium sized enterprises (SMEs) in the economy of many countries and the considerable attention placed on SMEs in the new Basel Capital Accord. The financial data variables of SMEs datasets can be coverage to financial ratios, which have been used to analyze and predict firm bankruptcies. Altman (2001) developed a Z-score that is useful in predicting firm bankruptcies (a low score indicates high probability of failure). The predictive model was based on a firm's working capital to assets, retained earnings to assets, EBIT to assets, market to book value of a share of stock, and revenues to assets.

In this project, I choice alternative approach. The goal is to find related variables and predict the failure risk of each individual in the testing set based on the various models, such as logistic regression and dynamic EB via GLMM discussed in Lai, Su and Sun (2014). To develops the specific model to estimate one-year SME probability of default, and illustrate the steps of my analysis and compare the results obtained using different statistical instruments.

2. Dataset

The SME datasets for $n = 46595$ record data (omitted the Na observations) included 1226 companies from 1994/01/31 to 2014/11/30 quarterly data, which is 20 years, 84-period observations per company if the company are still active.

In this case, set the training set $t = 80$, the last 4 periods (1 year) data set to be the testing set. Because, the interest will be adjusted per year generally. In addition, for forecasting more accurately in the k-step ahead predictions, k should be limited.

3. Features and Preprocessing

The variables in SME dataset are in the Table 1, included 19 financial data. Summary the SME data and check the quantile, for getting the first intuitive understanding. For further process the SME datasets to be longitudinal and cross-sectional, converge the date to period $t = 1, 2, ..., 84$ for specific company.

The Table 2. quantile of SME variables shows that the financial status of 1226 companies in the 20 years are complex.

TABLE 1. Variables in SME dataset

	Variable	Label
1	tic	Ticker Symbol
2	datadate	Date to start record
3	fqtr	Fiscal Quarter
4	staltq	Status Alert
5	bkrpt	1 = bankerupt, 0 = not bankrup
6	bkdate	Date to record bankruptcy
7	atq	Assets - Total
8	teqq	Stockholders Equity - Total
9	req	Retained Earnings
10	oibdpq	Operating Income Before Depreciation - Quarterly
11	ibq	Income Before Extraordinary Items
12	niq	Net Income (Loss)
13	piq	Pretax Income
14	dlttq	Long-Term Debt - Total
15	intaccq	Interest Accrued
16	ivtq	Inventories - Total
17	lctq	Current Liabilities - Total
18	ltq	Liabilities - Total
19	dlcq	Debt in Current Liabilities
20	cheq	Cash and Short-Term Investments
21	rectq	Receivables - Total
22	apq	Account Payable / Creditors - Trade
23	saleq	Sales / Turnover (Net)
24	COSTAT	Active / Inactive Status Market
25	INTANQ	Intangible Assets - Total

4. **Logistic Regression Model**

Based on the assumption that there are two populations $(\mathbf{x}, 0)$ and $(\mathbf{x}, 1)$ corresponding to $I = 0, 1$ so that

$$Pr(I = 1|\mathbf{x} = \mathbf{x}) = \frac{\pi \exp\{-d_1(\mathbf{x})/2\}}{\pi \exp\{-d_1(\mathbf{x})/2\} + (1-\pi)\exp\{-d_0(\mathbf{x})/2\}} = \frac{1}{1+e^{-s(\mathbf{x})}}$$

is the posterior probability that $I = 1$ when $\mathbf{x}$ is observed, logistic regression treats $P(I = 1|\mathbf{x}) = E(I|\mathbf{x})$ as a regression function. The transformation logit function $g(p) = \text{logit}(p)$ for defined by

$$g(p) = \log[p/(1-p)], \quad 0 < p < 1$$

Extension of the OLS estimate β in linear regression model from the normal distribution to the more general exponential family of distributions to *generalized linear models*, following the assumptions : (A1) y_t has density function

$$f(y; \theta_t, \phi) = \exp\{[y\theta_t - b(\theta_t)]/\phi + c(y, \phi)\}$$

TABLE 2. Quantile of SME variables

Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
atq	0.001	5.914	17.671	93.640	63.786	1505.92
teqq	-236.813	2.271	9.704	23.048	28.238	1232.96
req	-919.522	-36.55	-8.945	-28.559	1.860	691.960
oibdpq	-153.532	-0.745	-0.0060	-0.1641	0.8520	50.2770
ibq	-246.029	-1.073	-0.1120	-0.8269	0.3505	124.119
niq	-246.029	-1.082	-0.111	-0.829	0.357	156.765
piq	-246.029	-1.111	-0.1230	-0.7694	0.4560	167.143
dlttq	-3.833	0.000	0.370	10.736	5.053	604.534
intaccq	0.0000	0.0000	0.0270	0.2558	0.1650	17.3240
invtq	0.000	0.000	0.220	1.695	1.762	180.548
lctq	-0.012	1.261	3.378	30.442	8.882	1339.50
ltq	-0.030	2.126	6.316	70.559	24.579	1386.28
dlcq	-0.002	0.000	0.124	3.759	1.585	458.781
cheq	-32.757	0.503	2.755	10.175	10.887	575.218
rectq	0.0000	0.3155	1.4510	45.836	5.4575	1207.48
apq	0.000	0.367	1.062	52.007	3.448	1272.92
saleq	0.000	0.605	2.421	3.457	5.469	16.226
INTANQ	0.000	0.000	0.000	2.555	0.866	460.809

(A2) $g(\theta_t) = \beta^T \mathbf{x}$ for some given smooth increasing function g and unknown parameter β.
(A3) $(\mathbf{x_t}, y_t), 1 < t < n$, are independent.

The logistic regression model

$$P\{y_t = 1|\mathbf{x}_t\} = \frac{\exp(\beta^T \mathbf{x}_t)}{1 + \exp(\beta^T \mathbf{x}_t)}, \quad P\{y_t = 0|\mathbf{x}_t\} = \frac{1}{1 + \exp(\beta^T \mathbf{x}_t)}$$

Model Selection

Considering the Bayesian approach involving posterior distributions as an alternative to likelihood theory for parametric models, use an approximation to the posterior probability of a regression with p input variables, Schwarz(1978) derived the Bayesian Information Criterion

$$\text{BIC}(p) = \log(\widehat{\sigma}_p^2) + (p \log n)/n$$

Stepwise regression (backward elimination step or forward step) proceed until the information criterion does not improve; each step removes or adds the basis function in the model with the smallest value of the Wald statistic.

In this case, the fitted models via forward and backward.

TABLE 3. Stepwise forward model selection via BIC

	Estimate	Std.Error	z-value	Pr(> |z|)
(Intercept)	-4.2812394	0.0690582	-61.99	< 2e-16 ***
teqq	-0.0139858	0.0021547	-6.491	8.53e-11 ***
intaccq	1.6741462	0.2167270	7.725	1.12e-14 ***
apq	-0.0070990	0.0044660	-1.590	0.111935
INTANQ	-0.1664306	0.0312018	-5.334	9.61e-08 ***
req	-0.0032684	0.0005605	-5.832	5.49e-09 ***
cheq	-0.0359908	0.0074788	-4.812	1.49e-06 ***
saleq	0.0854831	0.0156083	5.477	4.33e-08 ***
dlttq	-0.0299953	0.0056202	-5.337	9.45e-08 ***
rectq	-0.0086147	0.0023073	-3.734	0.000189 ***
lctq	-0.0135991	0.0038227	-3.557	0.000374 ***
piq	-0.0219544	0.0053642	-4.093	4.26e-05 ***

(Dispersion parameter for binomial family taken to be 1)
Null deviance: 5445.1 on 43236 degrees of freedom
Residual deviance: 4833.8 on 43225 degrees of freedom
AIC: 4857.8
Number of Fisher Scoring iterations: 12

TABLE 4. Stepwise backward model selection via BIC

	Estimate	Std.Error	z-value	Pr(> |z|)
(Intercept)	-4.2897133	0.0691541	-62.03	< 2e-16 ***
atq	-0.0138971	0.0021704	-6.403	1.52e-10 ***
req	-0.0033208	0.0005569	-5.963	2.48e-09 ***
piq	-0.0218964	0.0053034	-4.129	3.65e-05 ***
dlttq	-0.0325289	0.0052379	-6.210	5.29e-10 ***
intaccq	1.7353652	0.1936191	8.963	< 2e-16 ***
lctq	-0.0161813	0.0034420	-4.701	2.59e-06 ***
ltq	0.0149847	0.0027413	5.466	4.59e-08 ***
cheq	-0.0365349	0.0073418	-4.976	6.48e-07 ***
rectq	-0.0107304	0.0024292	-4.417	1.00e-05 ***
saleq	0.0849443	0.0156824	5.417	6.08e-08 ***
INTANQ	-0.1639615	0.0306687	-5.346	8.98e-08 ***

(Dispersion parameter for binomial family taken to be 1)
Null deviance: 5445.1 on 43236 degrees of freedom
Residual deviance: 4836.7 on 43225 degrees of freedom
AIC: 4860.7
Number of Fisher Scoring iterations: 10

Diagnostics and Applications to Probability Forecasts

For logistic regression, the Pearson residuals are

$$e^P(\mathbf{x}_j) = \frac{y_j - m_j\widehat{p}_j}{\sqrt{m_j\widehat{p}_j}(1-\widehat{p}_j)}$$

The sum of squares of the Pearson residuals is the usual chi-square statistic

$$\chi^2 = \Sigma_j \frac{(y_j - m_j\widehat{p}_j)^2}{m_j\widehat{p}_j(1-\widehat{p}_j)}$$

Use the χ^2-test to examine the final model by BIC, both step-forward and step-backward are not significant.

In this case, use the samples $t = 80, 81, 82, 83$ to predict the one-step ahead predictions, and to calculate the Brier's Score.

$$B_t = \Sigma_{\text{i:bank i is eligible at t}}(Y_{i,t} - \widehat{Y}_{i,t})^2$$

The BIC forward final model Brier's Score is 5.7558 $\times 10^4$, and BIC backward final model Brier's Score 5.6745 $\times 10^4$.

5. **Dynamic Empirical Bayes Models**

The empirical Bayes methodology, introduced by Robbins(1956) and Stein(1956), considers n independent and structurally similar problems of statistical inference on unknown parameters θ_i from observed data $Y_i (i = 1, ..., n)$, where Y_i has probability density $f(y|\theta_i)$. The θ_i are assumed to have a common prior distribution G that has unspecified hyperharameter. Letting $d_G(y)$ be the Bayes decision rule when $Y_i = y$ is observed, the basic principle underlying empirical Bayes is that d_G can often be consistently estimated from $Y_1, ..., Y_n$ leading to the empirical Bayes rule $d_{\widehat{G}}$.

The case $Y_i \sim N(\theta_i, \sigma^2)$ with known σ, yields the following Bayes estimate for the prior distributuion $G \sim N(\mu, \nu)$ of the θ_i

$$d_{\mu,\nu}(y) = \mu + \{\nu/(\nu + \sigma^2)\}(y - \mu)$$

Since $\mu = E(E(Y_i|\theta_i))$ can be consistently estimated by $\bar{Y} = n^{-1}\sum_{i=1}^n Y_i$ and $Var(Y_i) = \nu + \sigma^2$ can be consistently estimated by $s^2 = \sum_{i=1}^n (Y_i - \bar{Y})^2/(n-1)$, replacing μ and $\nu + \sigma^2$ by these consistent estimates yields an EB estimate form

$$d_{\bar{Y},s^2} = \bar{Y} - (1 - \sigma^2/s^2) + (y - \bar{Y})$$

Longitudinal Data and Cross-sectional Means

The longitudinal data $Y_{i,t}$ is the generalized linear model that assumes $Y_{i,t}$ to have a density function of the form

$$f(y; \theta_{i,t}, \phi) = \exp\{[y\theta_{i,t} - g(\theta_{i,t})]/\phi + c(y, \phi)\}$$

in which h is a smooth increasing function and $\mathbf{x}_{i,t}$ is a d-dimensional vector of covariates such that

$$h(\mu_{i,t}) = \beta^T \mathbf{x}_{i,t}, \quad \mu_{i,t} = \frac{d_g}{d_\theta}(\theta_{i,t})$$

Then, use the EB modeling to provide a more flexible linkage of the n time series.

Buhlmann and Gisler (2005) generalized the linear EB approach to credibility theory by developing *evolutionary credibility* that assumes a first-order autoregressive model for the prior means μ_t of $\theta_{i,t}$, with $E(Y_{i,t}|\theta_{i,t}) = \theta_{i,t} = \mu_t + b_i$ and $\mu_t = \rho\mu_{t-1} + (1-\rho)\mu + \eta_t$ in which η_t are i.i.d. unobservable errors with mean 0 and variance V. They use Kalman filtering to estimate μ_t. The method of moments proceeds yields for large n a consistent estimate $\bar{Y}_{t-1}$ of μ_{t-1}, replacing μ_{t-1} by $\bar{Y}_{t-1}$ in yields

$$\mu_t = \rho\bar{Y}_{t-1} + (1-\rho)\mu + \eta_t$$

The model obtained is a linear mixed model (LMM)

$$Y_{i,t} = \rho\bar{Y}_{t-1} + (1-\rho)\mu + b_i + \epsilon_{i,t}$$

the b_i are i.i.d. random effects with $E(b_i) = 0$.

$$Y_{i,t} = \Sigma_{j=1}^{p}\theta_j\bar{Y}_{t-j} + a_i + \beta^T\mathbf{x}_{i,t} + \mathbf{b}^T\mathbf{z}_{i,t} + \epsilon_{i,t}$$

where a_i and $\mathbf{b}_i$ are subject-specific random effects, $\mathbf{x}_{i,t}$ represents a vector of subject-specific covariates that are available prior to time t, and $\mathbf{z}_{i,t}$ denotes a vector of additional covariates that are associated with the random effects $\mathbf{b}_i$.

Generalized Linear Mixed Model (GLMM)

The *generalized linear mixed models* (GLMM) are an extension of generalized linear models by adding random effects to the linear predictor to accommodate for clustered or overdispersed data. As in the linear case, we can increase the predictive power of the model by including fixed and random effects and other time-varying covariates of each subject i, thereby removing the dependence of $h(\mu_{i,t}) - h(\mu_t)$ on t in the GLMM

$$h(\mu_{i,t}) = \Sigma_{j=1}^{p}\theta_j h(\bar{Y}_{t-j}) + a_i + \beta^T\mathbf{x}_{i,t} + \mathbf{b}^T\mathbf{z}_{i,t}$$

For notational simplicity, augment $\mathbf{b}_i$ to include a_i so that equation above can be written as $h(\mu_{i,t}) = \Sigma_{j=1}^{p}\theta_j h(\bar{Y}_{t-j}) + \mathbf{x}_{i,t}^T\beta + (1, \mathbf{z}_{i,t})\mathbf{b}_i$.

Diagnostics and Applications to Probability Forecasts

In this case, we simply considering the random effects companies `(1|tic)` in `glmer` bernoulli GLMM fitted function in R, and note that the setting `(1|tic)` don't considering the random effects with covariates of other variables. The performances of $t = 80, 81, 82, 83$ in the Table.5-8 below.

The dynamic EB via GLMM (by BIC forward variables) Brier's Score is 0.0102023, and the one (by BIC backward variables) Brier's Score 0.01174199.

The Brier's Scores of dynamic EB via GLMM are higher than the ones of logistic regression. Intuitively, the random effect `tic` (specific company) catch up the residuals of the logistic regression model. The some variables, which not significant in logistic regression, are significant in the dynamic EB via GLMM model, such as `apq, INTANQ, cheq`.

TABLE 5. GLMM model by time up to $t = 80$

	Estimate	Std.Error	z-value	Pr(> |z|)
(Intercept)	-2.975e+01	4.470e+00	-6.656	2.82e-11 ***
teqq	-3.013e-02	8.518e-03	-3.538	0.000404 ***
intaccq	-9.955e-01	6.095e-01	-1.633	0.102434
apq	-1.629e-02	1.757e-02	-0.927	0.354059
INTANQ	6.525e-04	4.971e-02	0.013	0.989527
req	-7.981e-02	1.294e-02	-6.166	7.00e-10 ***
cheq	-2.432e-02	1.507e-02	-1.614	0.106488
saleq	-7.019e-02	4.669e-02	-1.503	0.132712
dlttq	3.239e-02	1.566e-02	2.068	0.038686 *
rectq	2.340e-02	8.366e-03	2.797	0.005157 **
lctq	2.454e-02	1.502e-02	1.634	0.102278
piq	4.189e-02	2.052e-02	2.042	0.041187 *

TABLE 6. GLMM model by time up to $t = 81$

	Estimate	Std.Error	z-value	Pr(> |z|)
(Intercept)	-30.548277	2.359477	-12.947	< 2e-16 ***
teqq	-0.021750	0.008534	-2.549	0.01081 *
intaccq	-1.217248	0.605078	-2.012	0.04425 *
apq	-0.020204	0.043996	-0.459	0.64607
INTANQ	-0.031638	0.072419	-0.437	0.66220
req	-0.082681	0.006637	-12.458	< 2e-16 ***
cheq	-0.022673	0.014194	-1.597	0.11017
saleq	-0.105783	0.047165	-2.243	0.02491 *
dlttq	0.037587	0.015556	2.416	0.01568 *
rectq	0.025989	0.009615	2.703	0.00687 **
lctq	0.026860	0.015153	1.773	0.07630 .
piq	0.036769	0.019965	1.842	0.06553 .

6. **Discussion**

The logistic regression modeling is easy to implement and the performance is well. The dynamic EB via GLMM approach consider the cross-sectional information and over individual time series. The dynamic EB via GLMM approach will take more time to fit, if the sample is large.

Generally, the dynamic EB via GLMM catch more information of the residuals in logistic regression model, and improve the logistic regression model performance. But in this case, the SME observations dataset included 1226 companies by 20 variables in 20 years. The high time-fluent variables of companies distributed in various industry, such as the intangible assets and the cash and short-term investments, in time-varying covariates makes the dynamic model unstable.

TABLE 7. GLMM model by time up to $t = 82$

	Estimate	Std.Error	z-value	Pr(> |z|)
(Intercept)	-30.912505	4.629288	-6.678	2.43e-11 ***
teqq	-0.027184	0.010878	-2.499	0.01245 *
intaccq	-1.207422	0.611051	-1.976	0.04816 *
apq	-0.019841	0.017050	-1.164	0.24457
INTANQ	-0.016589	0.056093	-0.296	0.76743
req	-0.081243	0.014287	-5.686	1.30e-08 ***
cheq	-0.014953	0.012574	-1.189	0.23437
saleq	-0.079851	0.046492	-1.718	0.08588 .
dlttq	0.037133	0.015319	2.424	0.01535 *
rectq	0.023722	0.008524	2.783	0.00539 **
lctq	0.027654	0.014943	1.851	0.06423 .
piq	0.045377	0.020799	2.182	0.02913 *

TABLE 8. GLMM model by time up to $t = 83$

	Estimate	Std.Error	z-value	Pr(> |z|)
(Intercept)	-30.912505	4.629288	-6.678	2.43e-11 ***
teqq	-0.027184	0.010878	-2.499	0.01245 *
intaccq	-1.207422	0.611051	-1.976	0.04816 *
apq	-0.019841	0.017050	-1.164	0.24457
INTANQ	-0.016589	0.056093	-0.296	0.76743
req	-0.081243	0.014287	-5.686	1.30e-08 ***
cheq	-0.014953	0.012574	-1.189	0.23437
saleq	-0.079851	0.046492	-1.718	0.08588 .
dlttq	0.037133	0.015319	2.424	0.01535 *
rectq	0.023722	0.008524	2.783	0.00539 **
lctq	0.027654	0.014943	1.851	0.06423 .
piq	0.045377	0.020799	2.182	0.02913 *

TABLE 9. Scaled residuals of GLMM model by $t = 80, .., 83$

	Min	1Q	Median	3Q	Max
$t = 80$	-4.82	0.00	0.00	0.00	705.07
$t = 81$	-5.04	0.00	0.00	0.00	484.39
$t = 82$	-5.12	0.00	0.00	0.00	580.93
$t = 83$	-5.12	0.00	0.00	0.00	646.17

7. **R Code**

```
# convert to period
mydata = read.csv("sme_homework.csv", head = T)
head(mydata); summary(mydata); attach(mydata)
```

```
library(lubridate)
dataDate = as.Date(datadate, format = "%m/%d/%y")

dataQrt = quarter(dataDate, with_year = T)
qrtOrder = unique(dataQrt)[order(unique(dataQrt))]
num = cbind(rep(1:84), qrtOrder)

period = rep(0, dim(mydata)[1])
SME = cbind(period, dataQrt, mydata)

for (i in 1:dim(SME)[1]) for (j in 1:dim(num)[1]) {
    if (SME$dataQrt[i] == num[j,2]) SME$period[i] = num[j,1]
}
period = SME$period
SME = na.omit(SME)
sme.n = length(unique(SME$tic))  # 1226
attach(SME)


# logistic regression
t = 80
train = SME[which(SME$period <= t), ]
# fullmod and nullmod
nullmod = glm(bkrpt ~ 1, family = binomial(link = logit),
                         data = na.omit(train))
fullmod = glm(bkrpt ~ atq + teqq + req + oibdpq + ibq + niq + piq
                    + dlttq + intaccq + invtq + lctq + ltq + dlcq
                    + cheq + rectq + apq + saleq + INTANQ,
              family = binomial(link = logit), data = na.omit(train))

# forward BIC
n = nrow(train)
fit.bic = step(nullmod, k = log(n),
               scope = list(lower = formula(nullmod),
                            upper = formula(fullmod)),
               direction = "forward")

fit.back = step(fullmod, k = log(n), direction = "backward")

# diagnostics
1 - pchisq(fit.bic$deviance, fit.bic$df.residual)
1 - pchisq(fit.back$deviance, fit.back$df.residual)

# predicted function - logistic
glm.l = function(t, fit.bic) {
    # the variables of final BIC forward model
    X.bic = coef(fit.bic)
    variable.list = names(fit.bic$coef)[-1]
```

```
    # prediction
    data.test = SME[which(SME$period == t + 1), ]
    j = nrow(data.test)
    train = cbind(rep(1,j), data.test[, variable.list])
    pre.fit = as.matrix(train) %*% as.numeric(X.bic)
    p.fit = exp(pre.fit) / (1 + exp(pre.fit))

    # the real data
    real.bkrpt = data.test$bkrpt
    Lyp2 = (real.bkrpt - p.fit)^2
    cbind(p.fit, real.bkrpt, Lyp2)
}

# forward BIC - fit.bic
mat80 = glm.l(80, fit.bic)
colnames(mat80) = c("p.hat", "real.bkrpt", "L.yp2")
mat81 = glm.l(81, fit.bic)
colnames(mat81) = c("p.hat", "real.bkrpt", "L.yp2")
mat82 = glm.l(82, fit.bic)
colnames(mat82) = c("p.hat", "real.bkrpt", "L.yp2")
mat83 = glm.l(83, fit.bic)
colnames(mat83) = c("p.hat", "real.bkrpt", "L.yp2")

# backward BIC - fit.back
mat80b = glm.l(80, fit.back)
colnames(mat80b) = c("p.hat", "real.bkrpt", "L.yp2")
mat81b = glm.l(81, fit.back)
colnames(mat81b) = c("p.hat", "real.bkrpt", "L.yp2")
mat82b = glm.l(82, fit.back)
colnames(mat82b) = c("p.hat", "real.bkrpt", "L.yp2")
mat83b = glm.l(83, fit.back)
colnames(mat83b) = c("p.hat", "real.bkrpt", "L.yp2")


# Brier's Score - forward BIC model
(glm.score = sum(mat80[,3], mat81[,3], mat82[,3], mat83[,3])
             / ((84-80)*sme.n))

# [1] 0.0005755812

# Brier's Score - backward BIC model
(glm.score.b = sum(mat80b[,3], mat81b[,3], mat82b[,3], mat83b[,3])
               / ((84-80)*sme.n))

# [1] 0.000567446
```

```
# Dynamic EB via GLMM
library(lme4)

glmm.function = function(t, fit.bic) {
    data.train = SME[which(SME$period <= t), ]
    # variable of fic.bic
    variable.mod = formula(fit.bic)[-2]
    model = paste("bkrpt ~ 1 + (1 | tic) + ", variable.mod, sep = "+")[2]
    # fit GLMM
    glmm.fit = glmer(model, data = data.train,
                     family = binomial(link = logit))
    glmm.fit
}

# prediction function
glmm.l = function(t, fit.bic, glmm.fit) {
    varialbe.list = names(fit.bic$coef)[-1]
    # parameters of fit
    fixed.eff = fixef(glmm.fit)
    rand.eff = as.matrix(ranef(glmm.fit)$tic,
                         dim(ranef(glmm.fit)$tic)[1], 1)
    # predict
    data.test = SME[which(SME$period == t + 1),]
    j = nrow(data.test)
    train = cbind(rep(1, sum(SME$period == t + 1)),
                  data.test[, varialbe.list])
    company = data.test$tic
    # predict
    logit.fitted = as.matrix(train) %*% as.numeric(fixed.eff)
    + rand.eff[match(company, rownames(ranef(glmm.fit)$tic))]
    p.hat = exp(logit.fitted) / (1 + exp(logit.fitted))
    real.bkrpt = data.test$bkrpt
    Lyp2 = (real.bkrpt - p.hat)^2
    cbind(p.hat, real.bkrpt, Lyp2)
}

# forward - modeling and predictions
glmm80 = glmm.function(80, fit.bic)
g.mat80 = glmm.l(80, fit.bic, glmm80)
glmm81 = glmm.function(81, fit.bic)
g.mat81 = glmm.l(81, fit.bic, glmm81)
glmm82 = glmm.function(82, fit.bic)
g.mat82 = glmm.l(82, fit.bic, glmm82)
glmm83 = glmm.function(83, fit.bic)
g.mat83 = glmm.l(83, fit.bic, glmm83)

# backward - modeling and predictions
glmm80b = glmm.function(80, fit.back)
```

```
g.mat80b = glmm.l(80, fit.back, glmm80b)
glmm81b = glmm.function(81, fit.back)
g.mat81b = glmm.l(81, fit.back, glmm81b)
glmm82b = glmm.function(82, fit.back)
g.mat82b = glmm.l(82, fit.back, glmm82b)
glmm83b = glmm.function(83, fit.back)
g.mat83b = glmm.l(83, fit.back, glmm83b)

# Brier's Score - forward - dynamic EB via GLMM
(glmm.score = sum(g.mat80[,3], g.mat81[,3], g.mat82[,3], g.mat83[,3])
              / ((84-80)*sme.n))
# [1] 0.0102023

# Brier's Score - backward - dynamic EB via GLMM
(glmm.score.b = sum(g.mat80b[,3], g.mat81b[,3], g.mat82b[,3], g.mat83b[,3])
                / ((84-80)*sme.n))
# [1] 0.01174199
```

8. **Reference**

1. Tze Leung Lai, Haipeng Xing (2008) *Statistical Models and Methods for Financial Markets*

2. Tze Leung Lai, Haipeng Xing *Active Risk Management: Financial Models and Statistical Methods*

3. Tze Leung Lai, Yong Su and Kevin Haoyu Sun (2014) *Dynamic Empirical Bayes Models and Their Applications to Longitudinal Data Analysis and Prediction*

4. Tze Leung Lai, Yong Su and Zhiyu Wang *Evaluation of Econometric Forecasts with Applications to Default Prediction in Small and Medium Sized Enterprises*

5. Edward I. Altman and Gabriele Sabato. *Modeling Credit Risk for SMEs: Evidence from the US Market*

5. Jurg Schelldorfer, Lukas Meier and Peter Buhlmann *GLMMLasso: An Algorithm for High-Dimensional Generalized Linear Mixed Models Using l_1-Penalization*