Component-Based IRB LGD Models

Evaluating Calibration of Probability of Cure Models

Andrija Djurovic

www.linkedin.com/in/andrija-djurovic

IRB LGD Modeling in Practice

- Depending on the characteristics of the modeling portfolio, practitioners choose different model designs for the development of IRB Loss Given Default (LGD) models.
- One common design is the so-called component-based approach, which involves developing multiple components and integrating them into a final LGD model.
- Typical components of this design include the Probability of Cure (PoC), Loss Given Cure (LGC), and Loss Given Loss (LGL).
- Among these components, the PoC model usually involves the modeling of a binary target variable, while the other components typically use continuous target variables.
- When validating a component-based LGD model, practitioners usually validate all individual components as well as the performance of the integrated LGD model.
- However, validation of the PoC model is often less rigorous compared to the Probability of Default (PD) model, even though both use the same type of target variable. In particular, one critical aspect that tends to receive less attention than in PD models is the evaluation of PoC model calibration.
- Recognizing that the PoC model is often built to produce continuous outputs (either based on continuous risk factors or a high number of possible unique outcomes), practitioners too frequently neglect specific aspects of calibration evaluation.
- The following slides provide an overview of different statistical tests and metrics that can be used to evaluate the calibration of PoC models. Since the presented methods are not accompanied by a complete discussion of their advantages and disadvantages, practitioners are strongly encouraged to engage in critical thinking when applying and interpreting the results of each method.

Spiegelhalter Calibration Test

Let $y_i \in 0, 1$ be the observed binary outcome for instance i, and let $p_i \in (0, 1)$ be the predicted probability of the event (e.g., cure or default) for that same instance. Given these inputs, the Spiegelhalter test evaluates the global calibration of probabilistic predictions.
The following expression gives the test statistic:

$$Z = \frac{\sum_{i=1}^{n}(y_i - p_i)(1 - 2p_i)}{\sqrt{\sum_{i=1}^{n}(1 - 2p_i)^2 p_i(1 - p_i)}}$$

where:

- y_i is the observed binary response (1 if event occurred, 0 otherwise);
- p_i is the predicted probability of event from the model;
- n is the number of predicted instances.

The null hypothesis of the Spiegelhalter test is that the predicted probabilities are well-calibrated, i.e., they match the observed event frequencies.
Under the null hypothesis, the test statistic Z is asymptotically standard normal. Consequently, the `p-value` is calculated as $2 \cdot \Phi(-|Z|)$, where $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal distribution.
By comparing the calculated `p-value` with the chosen significance level, practitioners can determine the test outcome.

Hosmer-Lemeshow Test

The Hosmer–Lemeshow (HL) test is used to assess the goodness-of-fit of a probabilistic model for binary outcomes by comparing observed and expected event counts across grouped predictions.
In the context of Probability of Cure (PoC) models, if the model produces output that is considered continuous, the predicted probabilities must first be discretized. This is typically done using deciles, although other approaches may also be employed. If the model already produces a discretized output, the existing score bands are used for running the test.
The Hosmer-Lemeshow test statistic is calculated using the formula:

$$HL = \sum_{g=1}^{G} \frac{(N_g PoC_g - c_g)^2}{N_g PoC_g (1 - PoC_g)}$$

where:

- G is the number of cure score bins;
- N_g is the number of facilities in the bin g;
- PoC_g is the predicted probability of cure for the bin g;
- c_g is the number of observed cures in the bin g.

The null hypothesis of the Hosmer–Lemeshow test states that the predicted probabilities of cure equal the observed cure rates.
Assuming independence, the test statistic follows an asymptotic chi-square distribution with G degrees of freedom, where G is the number of cure score bands.

Expected Calibration Error

The Expected Calibration Error (ECE) is a summary metric used to quantify how closely predicted probabilities match observed frequencies in classification models. It is commonly applied to evaluate probabilistic predictions, such as those from Probability of Default (PD) or cure models.
Similar to the Hosmer–Lemeshow test, the ECE metric requires the model output to be discretized. After discretization, ECE measures the average absolute difference between the observed and predicted event rates across these bins.
The ECE is defined as:

$$ECE = \sum_{g=1}^{G} P(i)|O_i - E_i|$$

where:

- G is the total number of bins (groups);
- $P(i)$ is the fraction of total observations falling into bin i;
- O_i is the observed fraction of positive (cure) outcomes in bin i;
- E_i is the average predicted probability in bin i.

As shown in the formula, the ECE assigns greater weight to bins with more samples.
A lower ECE indicates better calibration.

Reliability Diagram

A reliability diagram is a visual tool used to assess the calibration of probabilistic predictions for binary outcomes. It compares predicted probabilities to observed event frequencies by grouping predictions into bins, similar to those used in the Hosmer–Lemeshow test or the Expected Calibration Error metric.
For each bin:

- The average predicted probability is computed and placed on the x-axis.
- The observed event rate (i.e., the proportion of cured cases) is plotted on the y-axis.

Each bin corresponds to one point on the diagram, and these points are typically connected with lines for clearer interpretation.
A perfectly calibrated model would produce points lying on the 45-degree diagonal, indicating that predicted probabilities match observed frequencies across all bins.

Deviations from the diagonal suggest miscalibration:

- Points below the diagonal indicate overestimation (predicted probabilities are too high).
- Points above the diagonal indicate underestimation (predicted probabilities are too low).

Cox Intercept and Slope

The Cox intercept and slope method assesses calibration without requiring binning. The method fits a logistic regression model to the observed binary outcomes using the log-odds of the predicted probabilities.
The calibration test is defined within the framework of the logistic regression model:

$$y = a + b \cdot \text{logit}(p)$$

where:

- y is the binary indicator;
- $\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$ is the log-odds of the model's predicted probabilities;
- a is the intercept, capturing overall miscalibration (e.g., systematic overestimation or underestimation);
- b is the slope, indicating whether the model overestimates or underestimates the modeled rate.

A model is considered well-calibrated if $a = 0$ and $b = 1$. To jointly test whether the intercept and slope equal their expected values ($a = 0$, $b = 1$), a likelihood ratio test (LRT) can be used. In simple terms, this test compares the fully determined (restricted) model (with $a = 0$ and $b = 1$) to the unrestricted logistic regression model with freely estimated parameters.
The LRT statistic is given by:

$$-2 \cdot (\log L_0 - \log L_1)$$

where $\log L_0$ and $\log L_1$ are the log-likelihoods of the restricted and unrestricted models, respectively.

The null hypothesis of the test is that the model is well-calibrated.
Under H_0, the test statistic follows a chi-squared distribution with 1 degree of freedom, and the p-value is calculated accordingly.
By comparing the calculated p-value with the chosen significance level, practitioners can determine the test outcome.

Simulation Study

This simulation aims to present the results of the statistical tests and metrics described in the previous slides, using the dataset available at the following link.
To evaluate the calibration of the simulated Probability of Cure model, only two variables are required:

1. The observed event indicator. In this case, the variable `y` represents a binary indicator for facilities that have been cured.
2. Model predictions. In this case, the variable `p` represents the model's predicted probability of cure for each facility.

As some statistical tests and metrics require the data to be grouped, the following table provides an overview of the necessary inputs for calculating these tests and metrics.

```
##  bin  no ne      y      p
##    1 120  2 0.0167 0.0580
##    2 123 21 0.1707 0.1165
##    3 129 27 0.2093 0.1771
##    4 108 25 0.2315 0.2541
##    5 120 44 0.3667 0.3662
##    6 121 55 0.4545 0.4679
##    7 119 65 0.5462 0.5450
##    8 125 78 0.6240 0.6353
##    9 115 87 0.7565 0.7142
##   10 120 96 0.8000 0.8336
```

The following slide visualizes and presents the results of each statistical test and metric.

Simulation Results