WHITE PAPER

AI-Driven Commercial Loan Pricing

July 2023

By Stiene Riemer, Cesar Torres, Martin van den Heuvel, Christian Wagner, Markus Wiemann, and Christof Naumzik

Setting the right prices for loans is one of the central challenges banks face when seeking to optimally deploy limited capital. This is particularly important in the current environment of shrinking deposits and increasing cost of funds.

Commercial loans are negotiated by relationship managers (RMs) who deal simultaneously with many different clients in highly complex environments. As such, they require reliable, accurate, and trustworthy pricing models to serve as the cornerstones of their negotiations. Data analyzed by our SmartBanking AI[1] team has revealed an opportunity for banks to better support RMs during negotiations through the application of Generative AI[2]. In fact, banks can see a significant increase in margins with this approach.

In this article, we describe how we leveraged AI and Generative AI to determine the optimal price setting and provide the best support to RMs. We began by letting two neural networks, one representing a virtual client and the other a virtual RM, compete against each other in a Generative Adversarial Network (GAN) setting. Presented with a specific loan profile, the virtual client generated acceptance probabilities for a range of suggested prices, letting RMs choose prices that optimize the risk-adjusted return rate.

We validated this approach on a loan dataset that had the rare feature of containing both accepted and rejected loans. This allowed us to compare the realized revenues to the expected revenues of our strategy, under the assumption that loans were accepted or rejected due only to price—an outcome we considered likely given that the data stems from an online platform that states prices for like-to-like loans. This analysis revealed a significant increase in the expected average risk-adjusted rate of return.

Commercial Loan Pricing Strategies Fall Short

Currently, most bank RMs set prices by leveraging pricing grids that typically differentiate price by type of loan (e.g., revolver, term), maturity, and the underlying credit-risk profile (ideally including risk-based profitability measures such as the Risk-Adjusted Return on Capital). The problem is that these pricing grids rely on a cost-plus approach to pricing (i.e., a minimum return), and the price ranges are set within rigid guardrails and not updated often enough to reflect current market conditions. RMs, tasked with negotiating with clients, therefore rely on intuition and experience to gauge important factors such as customers' willingness to pay, stickiness, and price elasticity.

Loan-pricing grids function by segmenting portfolios into buckets, usually based on a only a few dimensions such as client's risk and rate index. Most banks add minimum thresholds to ensure that risk and operating costs are covered and to achieve a desired return on capital, such as by considering RAROC. While simple to operate, these grid-based pricing strategies have several shortcomings:

- **They do not take enough relevant information into account:** To keep the pricing grids transparent and simple, banks use only a limited number of dimensions during grid development. Therefore, these pricing grids do not sufficiently account for the full client relationship.

- **They do not take market conditions into account:** Pricing grids offer only a suggested price range that, as noted, may not reflect changing market conditions, but rather the internal economics of the bank. This makes it difficult for RMs to weigh their options in the trade-off between higher prices and lower acceptance rates.

1. https://www.bcg.com/industries/financial-institutions/ai-in-financial-services
2. Although most people know Generative AI for its potential to generate text and images due to famous examples like ChatGPT and Midjourney, Generative Adversarial Networks (GAN) also represent one form of Generative AI that learns to generate new data based on patterns found in the training data.

- **They do not focus on performance:** Since pricing grids give only rough guardrails for the price negotiation process, RMs are left with a high degree of latitude. This results in the realized prices becoming overly reliant on each individual RM's experience and incentives.

To address these pricing-grid shortcomings, we developed a more sophisticated model that predicts loan-acceptance probabilities. This model can support RMs during price negotiations, enabling them to strike the optimal balance between loan prices, acceptance probabilities, and the resulting risk-adjusted returns.

Finding Selective Price Adjustments with Loan-Specific Elasticity Curves

For an approach that predicts loan-acceptance probabilities, it is crucial to have not only loans that clients historically accepted, but also loans they declined. Most banks, however, do not have detailed records of the negotiations leading up to their accepted lending contracts, or even of loan offers that clients declined due to pricing.

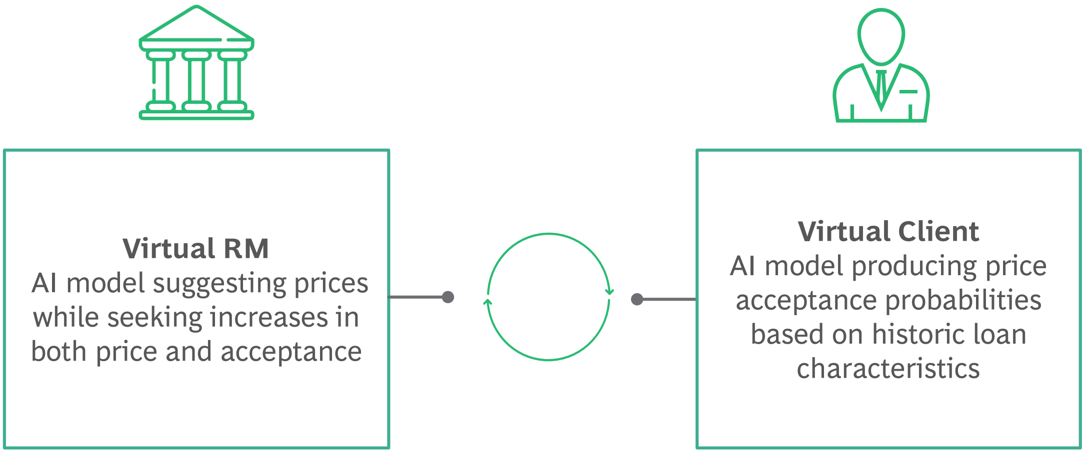
To enable banks to follow our approach and build models that predict loan-acceptance probabilities even when they lack data on rejected loans, we developed an innovative approach that constructs artificially generated rejected-loan prices using historically observed accepted loans only. In this approach, we created synthetic rejected loans by combining features from existing accepted loans with synthetic higher prices. These prices were sampled based on price-acceptance curves we constructed by averaging modelled parametric curves of loans within local clusters (as identified by a nearest-neighbor model that put greater emphasis on features with higher predicting power when predicting loan prices[3]). To validate that our approach delivers sufficiently useful results, we employed data from a platform offering loans that, along with suggested prices and client/product profiles, includes both accepted and rejected loans. This data allowed us to test our approach by training a model on real accepted loans and our synthetically generated rejected loan counterparts, and then demonstrate that the model performed well on a holdout dataset that includes both real accepted and real rejected loans.

With this enhanced dataset in hand, we leveraged Generative AI and developed two neural networks. The first neural network represented a virtual client that was trained to classify loans as either accepted or rejected. We used it to produce price-acceptance probabilities for a given client/loan profile at varying prices. After training the initial virtual client, we complemented this model with a second neural network, representing a virtual RM trained to generate prices that would maximize earnings based on given client and loan characteristics. We then simulated thousands of negotiations between the virtual client and virtual RM models (see Exhibit 1). The result was a refined virtual client to which we subjected numerous attempts to raise its price-acceptance threshold on loans specifically targeted by the virtual RM.

3. Predictive power of the features was derived from SHAP values taken from a boosted tree model trained to predict loan prices.

Exhibit 1: Simulated negotiations between RM and Virtual Client models

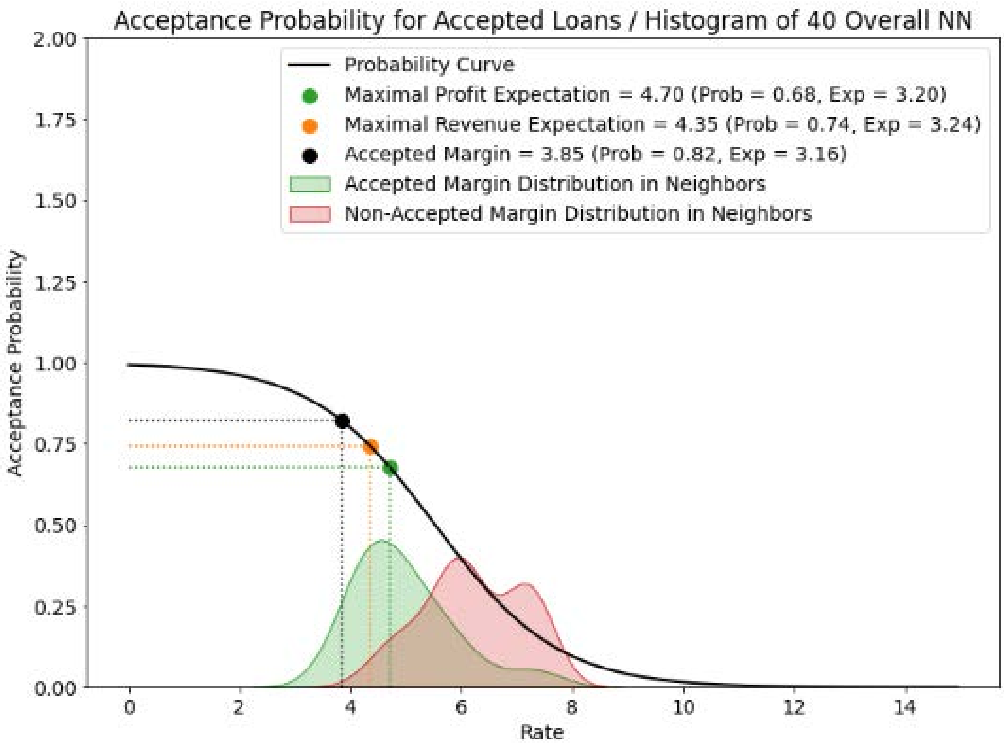
Using neural networks the modeling approach trains a Virtual RM and a Virtual Client by simulating thousands of loan negotiations



Once the virtual client is sufficiently trained, it can be of significant help to RMs since, when confronted with several prices for the same client/loan characteristics, it is able to draw an acceptance probability curve for given client and loan characteristics (see black curve in Exhibit 2 below). The RM (or, in most cases, the pricing tool supporting the RM) can then use this curve to optimize price with respect to specific strategic objectives. For example, RMs can find the price that maximizes the expected revenue calculated by multiplying price and probability (see orange point in Exhibit 2). Alternatively, by accounting for costs occurred due to defaults on loans, RMs or, precisely, the underlying pricing tool, can choose the price that maximizes the revenue-after-risk cost (corresponding to the green point in Exhibit 2). (Note that the black point in Exhibit 2 indicates the actual price negotiated by the RM for the loan. This point can, however, be displayed only in our development setup—not when predicting acceptance probabilities for new client/loan combinations as it requires underlying historical observations for the same client/loan combination.)

We complemented the derived and optimal curves by adding the green density curve, which indicates the distribution of accepted prices among similar real loans, and the red density curve, which indicates the distribution of rejected prices among similar real loans. Similar loans were found using a weighted-nearest-neighbors (KNN) approach that used higher weights for those features that showed greater predictive power in a separately fitted linear model predicting loan prices.

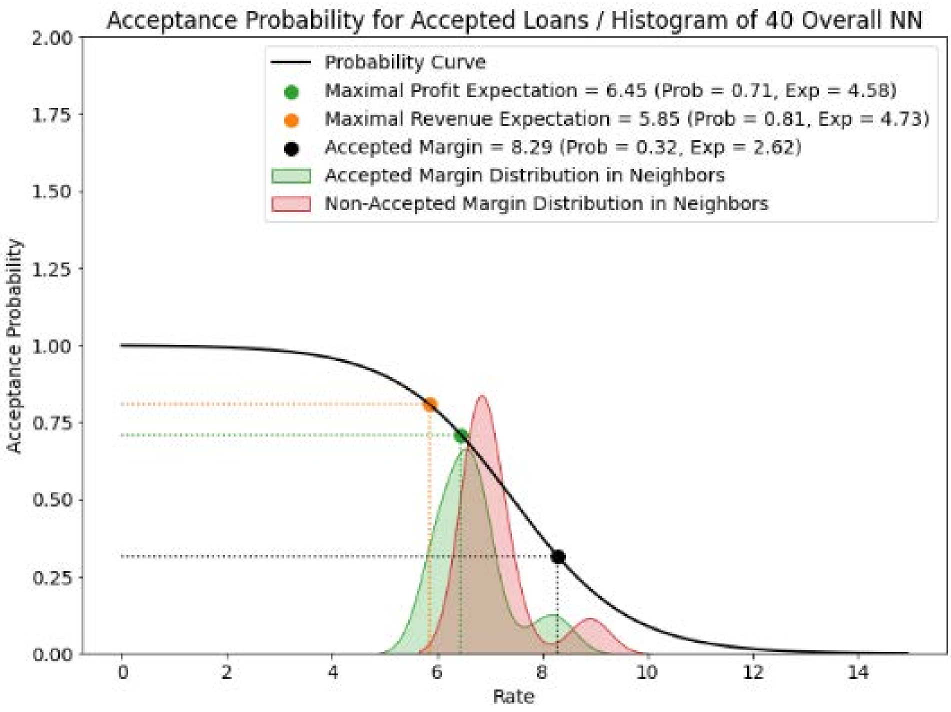
Exhibit 2: Loan-acceptance profitability curve



The model's suggested price, which leads to a maximal expected revenue-after-risk costs (green dot), is in the higher region of the historically accepted prices (green area) and retains sufficient distance from the rejected prices (red area) to keep the acceptance-probability high. In this case, the optimal price is higher than the actual price (black dot) negotiated by the RM.

A unique feature of our approach is that it can identify historical prices that were set too high and, thus, sacrificed too much in terms of potential sales volume and revenues after risk costs. In Exhibit 3 below, the RM negotiated a very high price that, in the past, would have led to rejection for most similar loans. Averaged across the entire bank portfolio, such a price would be suboptimal because it precludes too much potential revenue. Our model recognizes this inefficiency and suggests a lower loan price—one that is much closer to the intersection of the historically accepted and rejected prices.

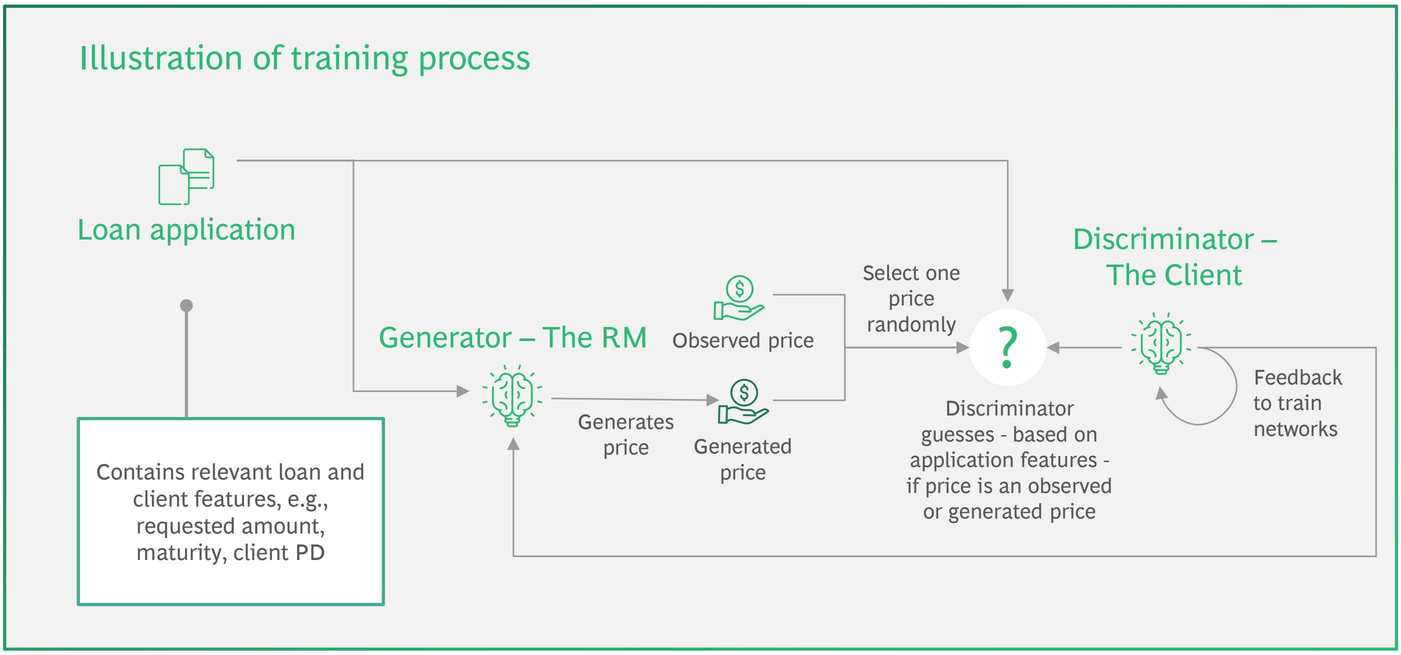
Exhibit 3: Loan-acceptance probability curve for inefficiently high price



Deep Dive: Refining the Virtual Client through Simulated Negotiations with a Virtual RM

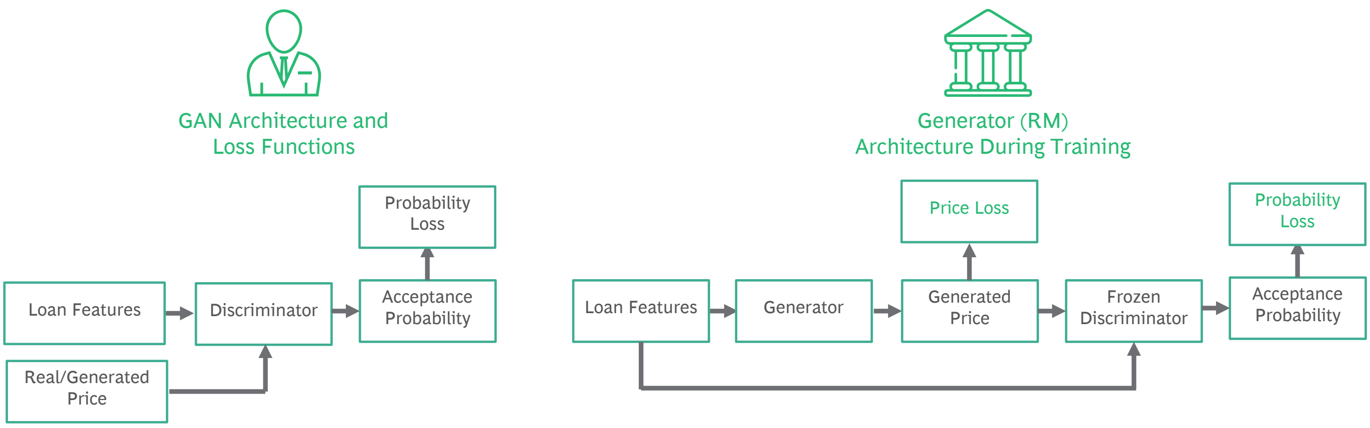
From a technical perspective, our virtual client and RM models are implemented as neural networks that can be combined to form the discriminator and generator of a generative adversarial network (GAN). During training, the objective of the virtual client (discriminator) is to distinguish real historical prices from artificially increased prices provided by the virtual RM (generator). The virtual RM is given the competing objectives of increasing the price of each loan while also increasing acceptance probability. The actual training process then consists of a continual feedback loop between these two models. During each training iteration the two models are alternately trained while staying in constant competition with each other.

Exhibit 4: The virtual client and virtual RM model training process



This process simulates real-world negotiations because the virtual RM consistently tries to implement pricing increases while maintaining the virtual client's acceptance probability for the artificially increased price. During the first half of each training iteration, the virtual RM is essentially forced to search for client/loan profiles in which willingness-to-pay has not yet been fully realized. During the second half, the virtual client is given a chance to change its own internal pricing criteria to adjust for the virtual RM's new target-price increases.

Exhibit 5: Generative Adversarial Network Training



During training the discriminator optimizes the standard classification loss (binary cross-entropy) while trying to distinguish between real and generated prices. The generator tries to optimize a sum of two losses $L_{prob} + L_{price}$ that represent its competing objectives.

$$L_{prob} = \max\left(0, -\log\frac{\text{discriminator probability for generated price}}{\text{discriminator probability for real price}}\right)$$

$$L_{price} = \alpha \max\left(0, -\log\frac{\text{generated price}}{\beta \times \text{real price}}\right) + \gamma \max\left(0, -\log\frac{\text{generated price}}{\text{real price}}\right)$$

In addition to mimicking a real-world situation, our adversarial training method leads to a quantifiable improvement in standard machine learning metrics. Adversarial training improves both (i) the likelihood, which measures the extent to which the acceptance probabilities given by the virtual client can be interpreted as probabilities in terms of the data distribution[4], and (ii) the area under the ROC curve, which measures the statistical power of the virtual client as a classifier.

Virtual Client Models Lead to Maximized Revenues after Risk Costs

To test the economic viability of our model, we compared lenders' actually realized average risk-adjusted rate of return to the expected rate of return given by our model (see Exhibit 6). We observed that optimizing the prices resulted in a 26% increase in the risk-adjusted rate of return. In contrast, the difference between the actual rate of return and the model's expected rate of return using the actual prices represents a fluctuation of only 7%. This indicates that the increase observed for optimized prices represents an improvement beyond statistical variation.

Exhibit 6: Comparison of RAROC for different pricing strategies

Average Risk-Adjusted Rate of Return per Negotiation		
Actual	Expected	Optimized
1.92%	2.06%	2.42%

Conclusion

By leveraging Generative AI, our team was able to develop an alternative to rigid pricing grids—an approach that enables commercial bankers to address the strategic optimization of prices by deriving client/loan-specific price acceptance probability curves. Rather than suggesting constant or rule-of-thumb pricing increases, the model lets bankers selectively target only those loan applications where prices should be adjusted. When used in loan simulations, the neural network model enables bankers to set targeted prices that can maximize the average revenue-after-risk costs across the entire portfolio of commercial loans. Within the used data set, this resulted in an impressive 26% increase in returns.

4. For a classifier F, optimizing the log-loss is equivalent to optimizing the likelihood of obtaining the data points (X, Y) by sampling labels Y = 1 with probability F(X) and Y = 0 with probability 1 – F(X).

About BCG

Boston Consulting Group partners with leaders in business and society to tackle their most important challenges and capture their greatest opportunities. BCG was the pioneer in business strategy when it was founded in 1963. Today, we work closely with clients to embrace a transformational approach aimed at benefiting all stakeholders—empowering organizations to grow, build sustainable competitive advantage, and drive positive societal impact.

Our diverse, global teams bring deep industry and functional expertise and a range of perspectives that question the status quo and spark change. BCG delivers solutions through leading-edge management consulting, technology and design, and corporate and digital ventures. We work in a uniquely collaborative model across the firm and throughout all levels of the client organization, fueled by the goal of helping our clients thrive and enabling them to make the world a better place.


bcg.com