Towards a Predictable Profitability

IFRS9 credit risk projection framework for Paygo companies

Authors : David HUGUES, Toky Axel ANDRIAMIZAKASON
Contributors : Alison BOESS, Alexandre GALICIER, Oliver REYNOLDS, Lucia SPAGGIARI, Guillaume CRUYT, Olivia JOHNSON

December 2024

Table of Contents

Purpose & Goals

This document outlines Baobab+'s ongoing efforts to build a transparent and reliable credit risk model that aligns with the specific needs of the PAYGO industry, ensuring the company has a clear path to profitability while expanding energy access.

Building clarity and trust in the path to profitability

The pay-as-you-go industry (PAYGO) is at a critical juncture, facing challenges on the path to profitability. As the industry gains in maturity, it is crucial to gain confidence and trust on the profitability model to continue attracting investments and delivering energy access for underserved populations. As of today, many stakeholders still lack clarity in future collection streams necessary to offset the high costs of last-mile distribution.

Unlike a typical retail business, PAYGO companies are distributing a financial product alongside their physical product. This financial product enables customers with limited cash flow to repay over time, increasing affordability and accessibility for millions of people. At the same time, PAYGO companies bear the costs of financing and servicing these financial products alongside the cost of credit losses, with both market conditions and operational performance informing the cost of this business model. PAYGO companies, in their path to profitability, need to consider both the quantity of sales and the quality of their customer selection and servicing to secure timely collections.

Many players in the industry (including Baobab+) have struggled to estimate accurately the quality of their sales and their servicing, and have overestimated future collections. This distorted the margins expected by the business, led to inaccurate financial projections, and failure to reach profitability through scaling up. At Baobab+, this lack of clarity cost us several strategic decisions we have since regretted : we firmly believed our way of operating was more profitable than it actually was.

Today, we think it is time the industry joins forces to build a greater clarity and trust into the PAYGO profitability model. Even if future collections are never fully guaranteed, they can be sufficiently predictable to make sound decisions and safely attract investment.

Indeed, historical collections or credit performance can be measured and translated into forward looking estimates.

While credit performance is probably the most challenging aspect of the business to evaluate and to manage, it is also one that can be spectacularly improved and can be a formidable lever to profitability.

In the last year, Baobab+ made significant efforts to improve its credit management, and reached levels that can be considered best in class in the industry, with final repayment rates for SHS & TVs products reaching 95%+ in Nigeria, 90%+ in Madagascar, 85%+ in Senegal (excluding recoveries from repossessions).

It was vital for Baobab+ to quantify precisely the improvements at every step of the journey, and build trust in these estimates, which are crucial to our profitability.

Steering Credit Performance

Credit performance is suffering from a lack of predictability, and is thus challenging to track and to steer effectively. This lack of certainty on performance leads to a legitimate lack of trust in the projected performance of the business.

But let us imagine for a moment we had a crystal ball that allowed us to see, without any doubt, the future collections of any customer.

With this crystal ball, we could :

- Understand the lifetime value, and associated profitability of each customer.
- Measure, quantify and anticipate how internal or external factors are affecting the collections on our portfolio of customers.
- Take operational decisions on which customers to serve to combine both ambitions for scaling up access to energy and profitability.
- Take operational decisions on how to react to issues customers are facing in repayment.

This would give us a great advantage in understanding the business health and potential, and effectively managing it.

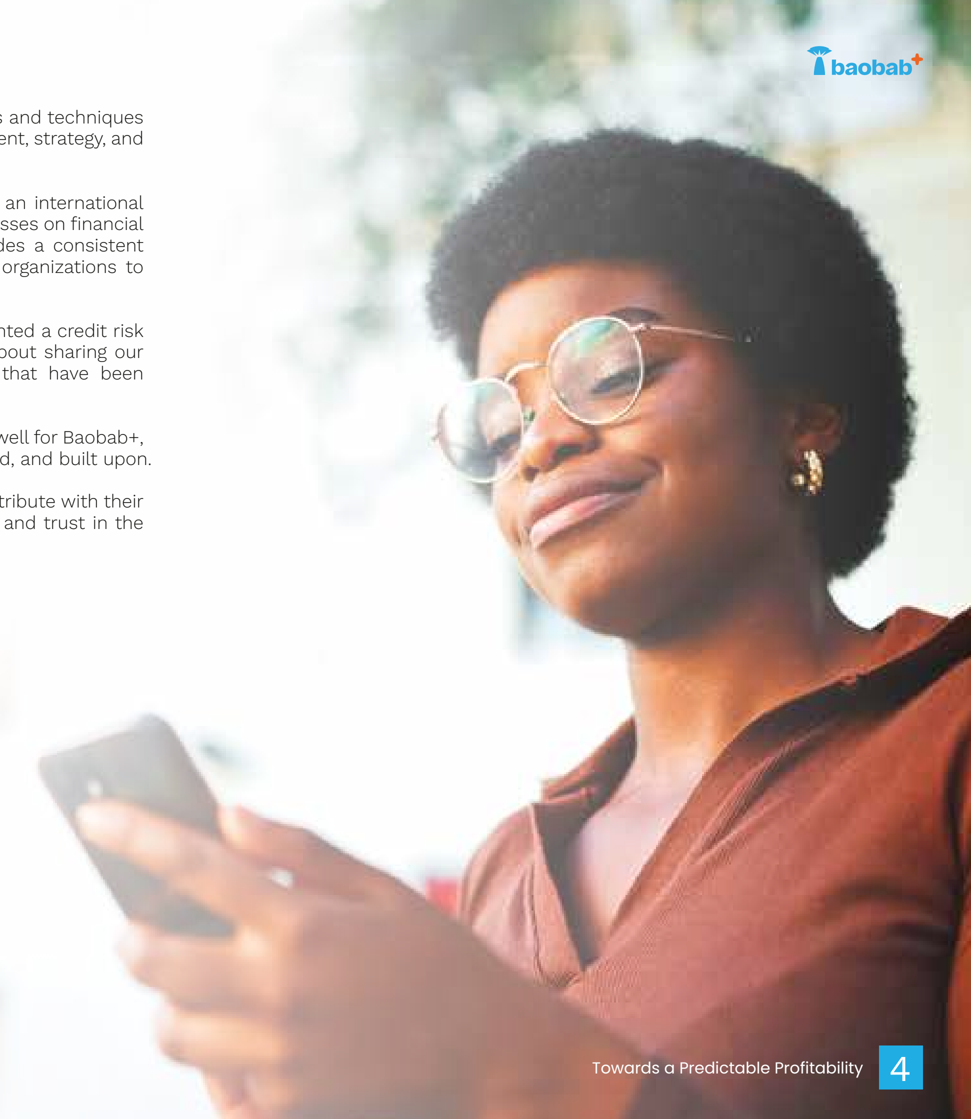
While we can't build a crystal ball, we can deploy a set of models and techniques that are accurate enough to make informed decisions for investment, strategy, and operations.

The IFRS 9 accounting norm is aimed at exactly that. IFRS 9 is an international standard developed to evaluate and recognize Expected Credit Losses on financial instruments. As a globally recognized framework, IFRS 9 provides a consistent approach for evaluating and accounting credit risk. It enables organizations to apply a forward-looking model to credit loss provisioning.

Leveraging historical data, Baobab+ has developed and implemented a credit risk projection model that follows IFRS 9 standards. This paper is about sharing our experience, the pitfalls we encountered and some solutions that have been working for us.

Although this solution is currently serving its purpose reasonably well for Baobab+, it must be considered as an input meant to be challenged, refined, and built upon.

We are therefore encouraging other players in the industry to contribute with their expertise and experience to that greater goal of reaching clarity and trust in the industry.



The challenge of predicting credit outcomes

Complexities of the PAYGO model

There are a few structural difficulties in measuring PAYGO credit risk that make it more challenging than traditional banking or microfinance. Let's explore a few of them :

High risk & informal environments

PAYGO last mile distribution is commonly targeting customers having limited or no access to electricity and formal banking services. This results in having a customer base that is generally more challenging to serve and less predictable than typical microfinance customers.

Unlike Microfinance lending, PAYGO customers are characterized by :

- A lack of formal data to prove their ability to pay.
- Mostly first-time customers with no prior credit history.
- Less stable and less predictable revenue streams or expenditures.

Unlike Microfinance lending, PAYGO loans are characterized by :

- Longer time horizons : typically 1 to 3 years.
- More flexible repayment terms and less penalties for late payments.
- Less strict definitions of customer default and write off policies.
- Mostly non-productive assets destined for household use.
- Fairly limited levers to address a defaulting customer except repossessing the device.

This leads to a structurally higher expected and more volatile credit risk in the PAYGO industry.

Long time-to-insight

Assessing credit risk for a young paygo company is like trying to assess the life expectancy of a young population. If the population is roughly 60 years old - you will get insights on mortality rates from 0 to 60 years. Of course, assuming the same mortality rates from 60 to 120 years will give you nonsensical projections. There's no other rigorous way than to wait for your population to grow older to measure mortality rates and thus life expectancy.

PAYGO offers less strict repayment terms than traditional banking, with the flexibility to have temporary interruptions in payment without consequences. This generally translates into longer repayment durations than those outlined in the contract terms.

Although the vast majority of customers repay within an acceptable timeframe, some customers can take an exceptionally long time to repay in full. This can be due to unresolved product issues or lack of rigor in following the repayment schedule.

In one of Baobab+'s countries, we observed the following trends for a product sold over 2 years : At contractual term completion (2y), 80% of customers are still in repayment. At 1.5x (3y), 30% are remaining and at 2x (4y) we still have roughly 17% that have neither defaulted nor paid in full their product (see Appendix 1).

Let's assume we want to assess our credit performance at 2x contractual term, 4 years after loan originations. For the 17% of customers that are still paying, we are blind and cannot know for sure what the end of their repayment is going to be like.

Although we can form an opinion on their level of risk, the only way to objectively assess their risk is to wait until they either default or complete their repayment.

Generalizing credit performance you observe on the other customers is very risky, as slower payers tend to show higher credit risk, but this is a common mistake (that we did make). Given the long tail on repayments, it may take 5+ years to realize the full picture of collections and default outcomes, depending on loan durations and companies tolerances to loan extensions.

Metrics becoming inefficient at the portfolio-level

As of today, most companies rely on portfolio health indicators, such as:

- Consecutive days disabled : PAR 30, PAR 90, etc...
- Cumulative days of disablement : Usage Rate, Collection Rate

These metrics have the merit of being very tangible, easily measurable and valid indicators to understand an individual loan or a group of loan's situations.

However, these metrics are most often aggregated on the portfolio level to report on a company's credit performance. The interpretation of this aggregation can become quite misleading.

Younger cohorts will naturally exhibit better metrics than older ones, introducing a bias on the average overall picture, especially in high growth scenarios.

This is well explained and demonstrated by Guillaume Cruyt, Investment Officer at EDFI MC ElectriFI. In this paper : Simple and Efficient Metrics for Solar Pay-As-You-Go Companies, Guillaume distinguishes efficient (based on cohort repayment) from inefficient metrics (portfolio aggregated metrics), and makes the case that currently adopted inefficient metrics lead to wrong interpretation of company performance.

Another danger is that portfolio-aggregated metrics can take a long time to materialize. Let's assume for a moment a new pricing you introduce strongly alters your credit performance on every new sale. It will take a certain time to materialize in the portfolio-level metrics. Portfolio-level metrics can be used to describe the portfolio health, but are not the best tool to measure credit performance.

The North Star target metrics

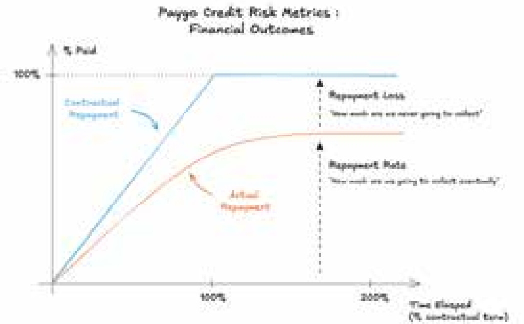
The question we are trying to answer here is : *"how much* are we going to collect?"

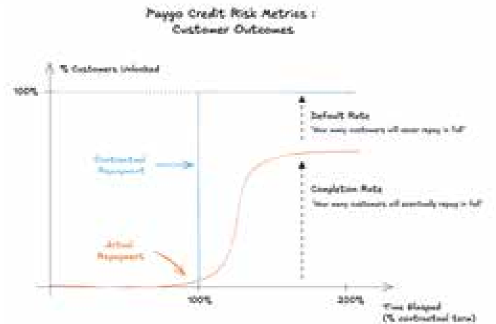
Please note that there's another important linked question "*how fast* are we going to collect this amount ?".

This question will be left outside of the scope of this document, although we recognize it is also essential. At Baobab+, the two topics are modeled independently, and associated costs are considered separately.

Measuring credit risk performance is about measuring the following set of metrics:

- Financial outcomes : proportion repaid of the total amount due, that we will call the **Repayment Rate** (opposite metric: **Repayment Loss**).
- Customer outcomes : proportion of customers paying their loan in full, that we will call the **Completion Rate** (opposite metric : **Default Rate**).





While these metrics do not exactly have the same definition, they are all pointing in the same direction, and are several facets of the same concept of customers *stopping their repayment*, or customers *defaulting.*

Provisioning Use-case & requirements

- *How much are we going to collect ?* is the key question that should be answered for every new customer, but also continuously on our portfolio of customers.

- **Use case #1 : Provisioning for future losses.** When applied to our portfolio of customers, this question solves our need for “provision for future losses”. Proper provisioning ensures that a company reflects an appropriately risk-adjusted value of the accounts receivable on the balance sheet, and ensures that profit margins on the P&L are realistically recognizing expected credit loss expenses against the up front revenue.

- **Use case #2 : Serve as a key input for our unit economics :** To build an accurate picture of the profitability model, and inform the pricing strategy aimed at profitability.

- **Use case #3 : Track credit performance improvements :** When applied at the cohort level, we can understand how our credit performance is improving from one month to another, and effectively manage our efforts on customer selection and onboarding.

- **Use case #4 : Track portfolio performance evolution :** When applied to a given portfolio at two points in time, this can give us an idea of the fair value of the portfolio, and its evolution. This will be particularly useful to assess the impact of external or internal changes

To deliver on these use cases, we need to satisfy the following requirements :

- **Requirement #1 : Provide an estimate of future losses on a group of customers.** It is clear that we will most certainly not be able to build a crystal ball that foresees the future collections of every customer - but we can aim at building a solid estimation on groups of accounts. We are going to use the output first at company level, for potentially thousands of customers in our portfolio. Then at monthly cohort level for hundreds of customers to understand the evolution in performance. Then maybe zoom in at a more granular level (e.g. per product/offer type - a must-have, per geographical area, per sales agent or per customer segment).

- **Requirement #2 : provide an estimate that will adjust based on repayment evolution.** *How much are we going to collect ?* is not a question that should be answered only once at the moment of the loan origination, but one that should be reevaluated continuously on every customer based on everything we know. We want our projection to consistently evolve and translate what we are witnessing into future collections.

- **Requirement #3 : produce interpretable outputs, where projections can be explained using simple, understandable metrics.** We do not want our model to be perceived as a black box. The model’s output should be decomposed into tangible building blocks understandable by management. This is critical to 1) building confidence and trust in the model and 2) establishing the link between operational and financial performance.

- **Requirement #4 : store projections and simulate backwards looking situations.** This will be crucial to test your model’s performance and build trust around it.

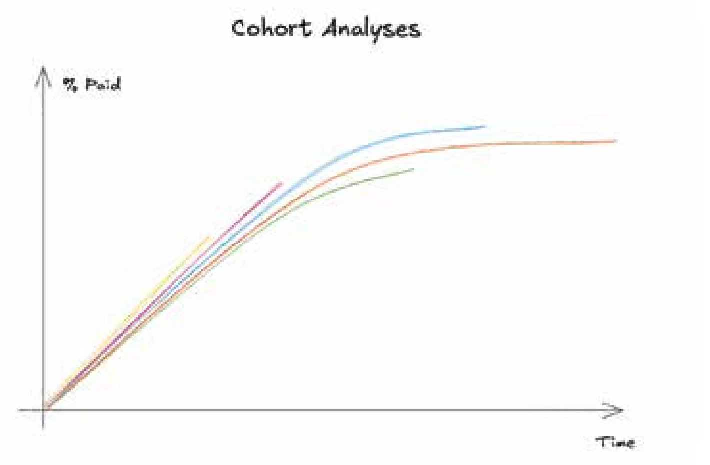
Baseline Methodology : Overview of current practices

Intro to cohort analyses

Before attempting to build any model - every company should already have performed this analysis on historical data. This simple and common framework is the first foundation to building an understanding of credit performance, and companies can extend the analysis to build a simple model to predict performance of recent cohorts.

Cohort analysis consists of displaying historical cohorts repayment (or any other KPI) since account origination. Metrics that can be qualified as inefficient on portfolio-level aggregations can become very insightful when calculated on cohorts. It is the case for example with usage rates, portfolio at risk, etc...

Appendix 2 is providing additional elements on methodology to build the following cohort repayment chart using very accessible data: 1) a dataset of loans and 2) a dataset of payments.



When plotting cohort repayment, we observe a convergence towards a plateau that is reached somewhere between 150 or 200% of the initial contractual term. This plateau is exactly the repayment rate mentioned in the previous section.

If your company has sufficient data, this simple chart can give you instantly an estimate of your historical performance.

Most importantly - we will see that the early trajectory of a cohort is a good predictor of final performance.

Based on this principle, we can build a first repayment rate projection. This link between early performance and final outcomes is essential and comforting on our ability to project credit risk. We're not taking shots in the dark : collections are quite predictable and not purely a product of unpredictable events.

Current standard : Cohort Repayment Projections

To our knowledge, this has become the standard way of evaluating collection performance and projecting credit risk in the industry. We highly recommend implementing this methodology, or a variant of it, before considering more complex models.

The paper we mentioned before, "Simple and Efficient Metrics for Solar Pay-As-You-Go (PAYGo) Companies" by Guillaume Cruyt, is making the case for a methodology that uses historical cohort projections to project future repayment rates.

The methodology consists of using the last 6 available data points on cohort collection to derive the next unknown data point. This is essentially projecting the collection performance of the last 6 cohorts on the next, and it provides very satisfactory results.

Additionally, the paper suggests adjusting the projecting with a scale factor, comparing the weighted performance of these 6 cohorts to the one we try to project. Let's assume our cohort we've collected 10% more on this cohort than the previous ones, we can scale our projections by the same factor (110%).

This technique does a great job at picking up individual cohort performance. Without it, the methodology would consist of smoothing performance over the last 6 cohorts. Using this scale factor, we have a predictive estimate that adjusts to varying cohort performance.

There are several other ways we could improve incrementally this methodology, for example :

- Linearizing payments for better interpretation of cohort relative performance
- Removing the deposit and initial free period - to avoid potential biases due to pricing changes
- Adapting this methodology on a relative time scale (% of contractual term instead of number of months), to adapt to various contractual maturities.
- Improving the scale factor adjustment - for a more accurate prediction.

All of which could potentially lead to more accurate performance but also complexification of the model.

Limitations of this approach

Limitation #1 : Focus on future repayment, but no insight on the probability of default

This model provides an estimate of future repayment rate, but not the default rate (number of customers defaulting). This is limiting in several ways :

- The default rate is an important indicator of company success and overall impact. It tells us how many customers end up having a positive experience and end up actually benefiting from the service provided by the company.
- The default rate is also crucial to model recoveries, as defaulted accounts will be the one prone to a repossession. If a company needs to estimate its recoveries thus the default rate, another model or piece of logic would have to be used to transform the insight “xx% not collected = yy% customers defaulting”.
- It doesn’t provide the information ‘when’ the default is likely occurring. We believe this is valuable information to inform operational decision making.

Limitation #2 : Difficulty to link performance to specific metrics and characteristics of non performing loans.

This methodology is effective in identifying trends and assessing the performance of groups of loans, providing reliable results and capturing cohort-level insights. However, it does not directly allow us to determine the specific drivers behind the performance within a group of accounts.

For example, if we want to estimate the risk of a loan going unpaid for 90 consecutive days to guide operational decisions, this methodology lacks the ability to apply a precise, individualized estimate for that loan.

Depending on the use case, these limitations might be acceptable. However, for Baobab+, our need to comply with IFRS9 made this approach unsuitable.

We determined that we could remain within the standard, and develop a more robust and accurate methodology enabling a larger spectrum of use-cases.

Measuring and Benchmarking Performance

In our view, this is the first question we should ask ourselves before attempting to build any model. Thinking about how to test any projection will ensure that :

- We are building a model that is actually going to solve what it is intended for.
- We are evaluating the added value of changes and improvements of methodology vs the added complexity and cost of implementation.
- We are actually building something we can trust, that delivers accurate-enough outputs.

Testing the ability to provision for future losses

Let's assume we apply our model to provision for future losses on a group of customers we just sold to. The output would be an expected loss for that group of loans.

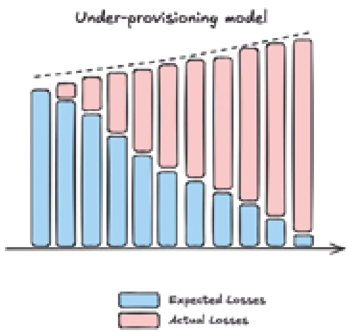
Let's also assume we stopped sales completely - and follow the future evolution of this portfolio of loans, measuring future expected and actual observed losses on that portfolio. We would expect a perfect model to behave like this :

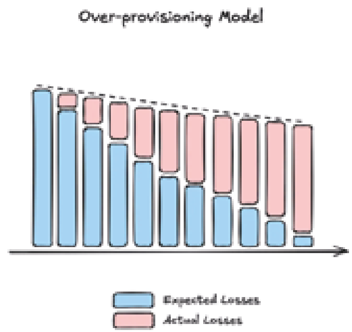
At the moment the portfolio is first provisioned, we estimate a certain amount of future losses. We expect these expected losses to gradually translate into actual observed losses. At the portfolio level, the sum of expected and actual losses should therefore remain constant in time.

There are two ways to prepare this kind of chart :

- Use historical provisioning levels and observed losses (preferred but the data is not available when you first develop your model).
- Apply your model on historical snapshots of your portfolio data (this requires having historical snapshots of your portfolio data stored).

This type of test can give you confidence your model is not either over-provisioning or under-provisioning, as illustrated in the charts below :

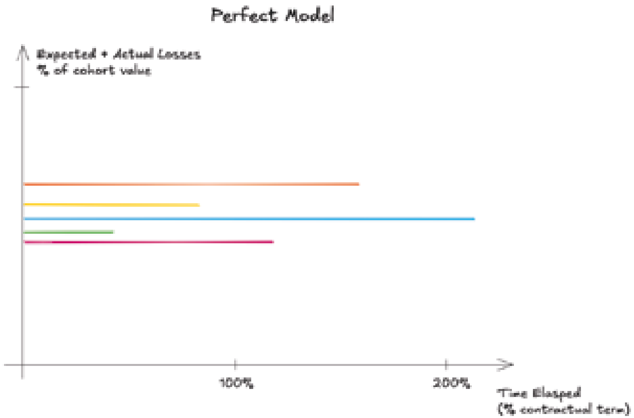




Also - you can apply this test on the level you require : first at the portfolio level, but also at product level, cohort level, or area level.

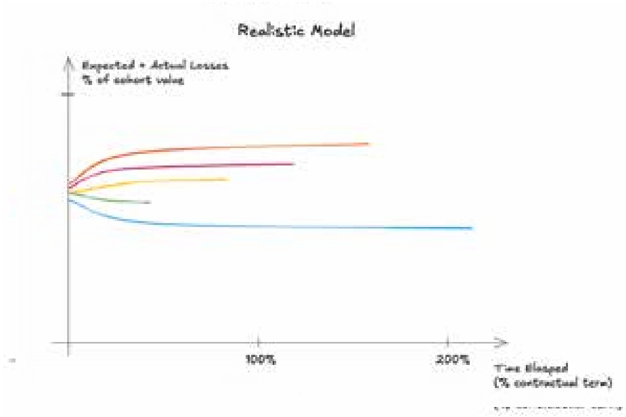
Testing cohort-level projections

Using the same principle, you can calculate for every cohort, the expected + actual losses, and see their evolution. A perfect model would guess immediately the future losses. And thus the sum of actual + expected losses would remain constant.

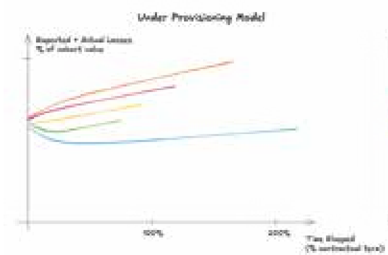


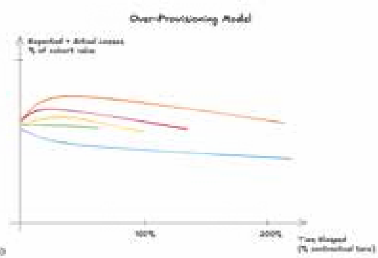
However, not much is known about the cohort quality at loan origination. We can't expect our model to guess immediately the quality of the originated cohort. However, we can expect our model to converge as fast as possible to a stable estimate of the cohort final value, depending on the cohort's repayment behavior.

A more realistic target behavior would therefore be looking like this :



This kind of visualization can help you also detect biases in your model.



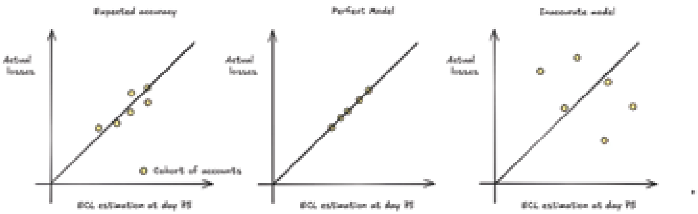


At Baobab+, we regularly assess the model behaviour this way. We recently spotted a situation where the last 6 months of all cohorts consistently increased, this informed us of a country-wide deteriorating trend of our ability to service the portfolio and collect.

Testing predictive power & biaises

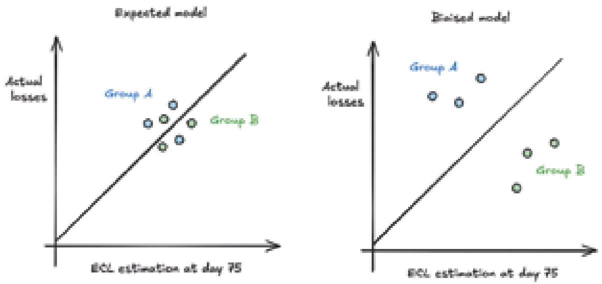
Another way to evaluate your model performance is to evaluate how much your model output at a certain point in time (for example we test at 75 days since loan origination), and plotting the projected vs actual value of a group of accounts (most commonly : a cohort).

This gives us an overview of the model's ability to predict final values at an early stage.



All previous methods can be used to detect biases, but this one is particularly convenient : You can form groups of accounts sharing a characteristic. It can be a particular behavior you want to test, or a segment of your customers. Based on those groups and simulations, you can plot their expected vs actual performance.

If you see some group of accounts sharing a characteristic systematically under or over evaluated, that would indicate that your model is potentially missing an important correlation.



The case for measuring loan-level performance

All of those methods are estimating performance for a group of loans (because projection on groups of loans is our primary requirement). It's important to note that the level of precision will strongly depend on the size of the group for which we're testing.

Indeed, a larger group of loans will tend to balance out errors made at the loan level. This is a direct consequence of the law of large numbers : The variance of an average of n identical experiments is going to be the variance of an experiment divided by n. That's why even a pretty rough model at the portfolio level can perform relatively well at the portfolio level, if the portfolio is sufficiently large.

When iterating on a model, and evaluating accuracy gains, we therefore encourage to use a loan-level indicator, such as the Mean Average Error for each loan.

Please note using a loan-level indicator does not necessarily mean we intend to use the projections at loan level. The better the accuracy at loan-level, the smaller groups of loans we can predict with a given level of accuracy. Unlocking smaller, more granular and more operational use-cases.

Let's assume we have a methodology giving satisfying results on a large portfolio of 10 000's of loans. That same methodology, if applied to a much smaller group of loans (e.g. agent portfolio - level, in the 10's of loans) might not provide trustworthy projections. A more advanced and accurate methodology will not make a significant difference on the portfolio-level use-case ; but can unlock the small-scale use-case.

Generating Artificial Data & Simulating Scenarios

Testing various models always rely on a comparison between projected and actual figures. Having sufficient historical data to have final actual losses for a significant volume of loans can be quite challenging.

Even with our scale at Baobab+, and 8 years of existence, availability of historical data has been a limiting factor to test robustly our models.

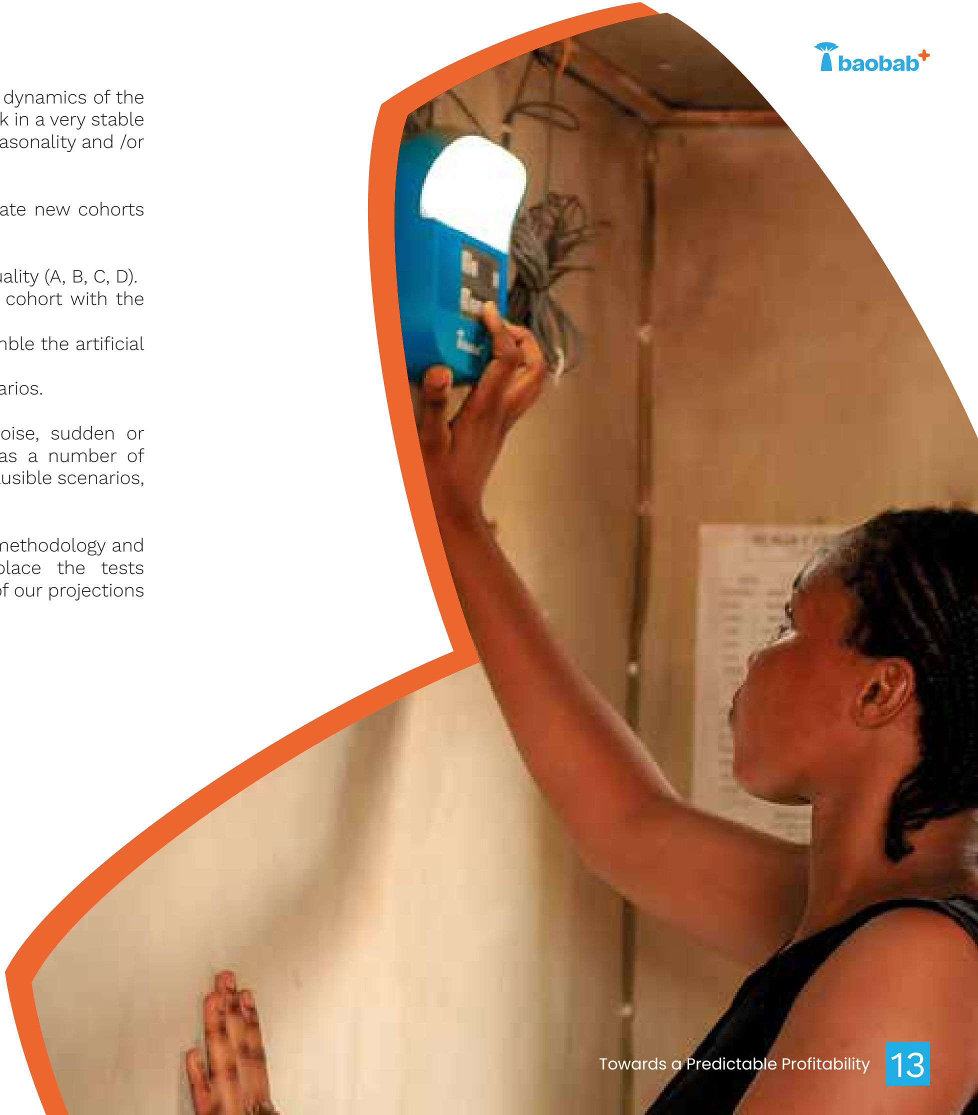
Also - models tend to behave very differently depending on the dynamics of the situations in which we operate. For example, projecting credit risk in a very stable situation is much easier than in a very unstable situation with seasonality and /or a sudden drop in performance.

To answer these issues, we have built a methodology to generate new cohorts based on historical loans. In brief it consists of :

1. Selecting a pool of historical loans with various levels of quality (A, B, C, D).
2. Defining scenarios, where we control the quality of each cohort with the proportions of class A, B, C and D.
3. Randomly selecting in the pool of historical loans to assemble the artificial cohorts, into a training dataset and a testing dataset.
4. Analyze the response of different models to different scenarios.

We can thus define stable scenarios, or simulate growth, noise, sudden or progressive variations in performance. While this approach has a number of limitations, it has the advantage of enabling us to test various plausible scenarios, and measure robustly how different models would react.

Testing on artificial data is very convenient to iterate on model methodology and benchmark and understand performance, but cannot replace the tests mentioned above in real-life situations ; testing the adequation of our projections to the strategic or operational decisions they are driving.



Introducing the Baobab+ IFRS9 Methodology

The following section outlines the methodology adopted by Baobab+ to project credit risk. In summary, this methodology assigns an expected credit loss per loan based on current outstanding per loan, discounted by the :

1. Probability the loan will default at some point based on the current % of their loan paid & a customer segmentation based on repayment pattern.
2. If the loan is currently late , add an additional probability that the loan will not repay ever again, based on their current days late & how far they are into their loan.
3. Subtract any potential value from repossession from outstanding (based on the probability of repossession and the expected value from repossessions)

It is important to note that this paper is not intended to be used as a reference for the IFRS9 norm, but to put forward a methodology aligned with IFRS9 principles that can provide accurate results for Paygo companies.

The following methodology is not strictly the one implemented at Baobab+. Several IFRS9 normative requirements applicable to Baobab+ some slight modifications from this core methodological approach. For simplification purposes, these normative requirements are not shared here.

Key concepts of IFRS9

ECL Modelling : PD, EAD and LGD

IFRS 9, an international financial reporting standard, aims to improve the accuracy of credit risk measurement by requiring financial institutions to estimate and account for expected credit losses (ECL), calculated as :

$$ECL = PD * LGD * EAD$$

- **Probability of Default (PD)** represents the likelihood that a customer will stop repayment during a specific period.
- **Exposure at Default (EAD)** refers to the total amount that is expected to be outstanding if and when a borrower defaults.
- **Loss Given Default (LGD)** part of the exposure that is not expected to be recovered after default.

By integrating PD, EAD, and LGD, IFRS 9 enables financial institutions to create a more forward-looking and accurate estimation of credit losses, contributing to better risk management and financial stability.

Stages of credit impairment

IFRS 9 also introduces a three-stage model for credit impairment to ensure that expected credit losses are recognized at an early stage:

- Stage 1 : Assets without a significant increase in risk since origination.
- Stage 2 and stage 3 : Assets with a significant increase in risk (S2) or that are already considered in default (S3).

The main goal of these stages is to distinguish between performing and non-performing assets, and adapt the methodology accordingly to accurately reflect the risk exposure.

Mapping IFRS9 concepts to Paygo

Here is how we apply these concepts to PAYGo credit risk :

$$ECL = (1)\ Amount\ not\ collected\ -\ (2)\ Amount\ Recovered$$

1. Amount Not Collected will be based on PD and EAD, representing how much might never be repaid.

2. Amount Recovered will include repossessed devices resale or reuse value.

(1) Amount not collected : PD & EAD

In this approach, default is recognized when an account is either formally written off, repossessed or reached 180 consecutive days of disablement.

PD & EAD combined represent the repayment loss, the amount that is never going to be collected, and are therefore equal to 100% - repayment rate.

$$Amount\ not\ collected = Contract\ value * (100\% - repayment\ rate)$$

$$Amount\ not\ collected = (\int_{current\ state}^{completion} PD * EAD)$$

(2) Amount Recovered : LGD

LGD represents the part that is lost in case of default. This amount is equal to the repayment loss from which we can deduct the amount recovered from repossessions and their revaluation.

There are two components involved in evaluating how much can be recovered from a default customer :

- Probability of Repossession : Estimating how many defaulting customers we'll be able to repossess.
- Net present value of a Repossession : Estimating how much value we can get from a repossessed device.

Therefore :

$$Amount\ recovered = \%\ default\ rate * P_{repossession} * NPV_{repossession}$$

$$Amount\ recovered = (\int_{current\ state}^{completion} PD) * P_{repossession} * NPV_{repossession}$$

Adapted ECL formula

The full formula becomes :

$$ECL = (\int_{current\ state}^{completion} PD * EAD) - (\int_{current\ state}^{completion} PD) * P_{repossession} * NPV_{repossession}$$

1 IFRS9 does not impose a specific threshold for considering a loan has defaulted. However financial institutions generally consider 90 days as a tipping point, and Baobab+ had to follow that rule as part of a microfinance group. This does not significantly change the methodology or its results, but introduces additional complexity in modelling the LGD.

Modular Model architecture

We will be estimating these components separately by various models.

- **Model #1 (PD & EAD) Lifetime Expected Loss** or "will an on-time customer default over the rest of its lifetime and what amount is at risk ?"
- **Model #2 (PD for S2/3) Late Loans Expected Loss** or "will a late customer ever repay again ?"
- **Model #3 (LGD) Expected Recoveries** or "will we be able to repossess and reevaluate the devices of defaulting customers ?"

Having a modular approach to this has the following advantages :

- **Advantage #1 : Explainability.** Modeling the problem this way, we are able to break the final projected value down in various probabilities reflecting operational performance (of churning immediately, of churning at a later stage, of repossessing the device, etc...). It is much easier to explain and justify a projected figure when we have this level of detail - building confidence and trust on the model.

- **Advantage #2 : Extensibility.** We can work on improving each model independently from the others. That enables us to start with very simple estimates, test the models independently, and refine the models that seem the most critical to the final projection.

- **Advantage #3 : Specialization.** Each model can be tailored for the specific problem it is trying to solve. For example, a model evaluating our ability to repossess a device won't be so dependent on customer behavior data, but more on internal operations and device location. Narrowing down the scope and expectations enables us to have simpler and more precise models in place.

Model #1 (PD & EAD) Lifetime Expected Loss

Our approach : a repayment-based model

Time-based models, estimating a PD distribution across time are used by many financial institutions and do work well for predictable EADs. But for Paygo, we do think the approach is flawed and suffers from many limitations described in Appendix 4.

We assessed that using the time since loan origination is not an effective progress indicator. Instead, we switched to using repayment progress as it provides better accuracy. This facilitates the challenges above.

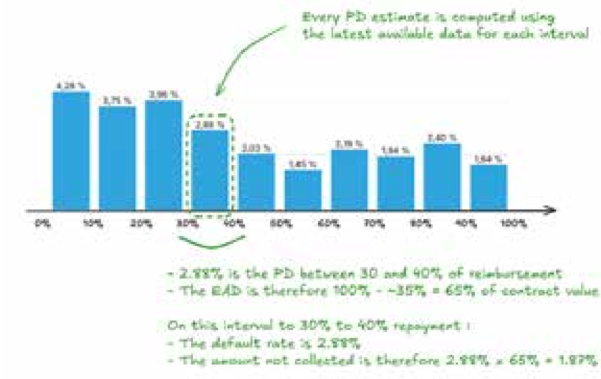
If x is the repayment progress (varying from 0 to 100%), then the remaining receivable at risk or EAD is directly (1-x).

$$\text{Amount not collected} = \int_{x=0\%}^{100\%} PD(x) * EAD(x)\,dx = \int_{x=0\%}^{100\%} PD(x) * (1 - x)\,dx$$

$$\text{Default Rate} = \int_{x=0\%}^{100\%} PD(x)\,dx$$

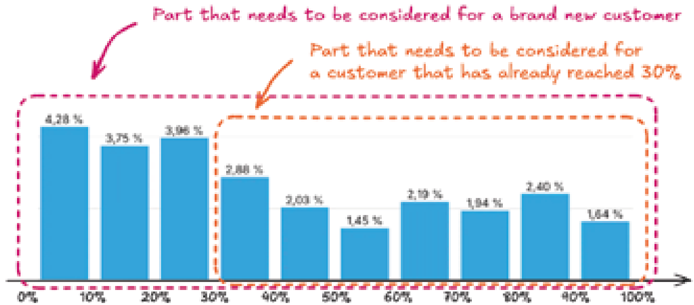
Expected Outputs

Here is an example of output produced by the model. It gives a probability of default on each 10% interval of repayment between 0 and 100%.



We can perform this calculation on each interval [0,10%], [10,20%], ... and sum back the results, forming the PD and PD * EAD on the interval [0, 100%].

This output is also quite handy to determine the PD associated with a part of the reimbursement. And thus, update our estimates based on repayment progress.



Sampling and Selection bias

However, this approach poses some new challenges about sampling data and is quite challenging to estimate with rigor.

If we try to use historical data to plot the chart above with relatively recent data, we'll immediately face the following issue about default recognition : Customers already in default or having paid in full do present clear outcomes, and can be used to compute a probability.

But what about the others ? How should we treat an account that is still paying but hasn't finished yet ? Actually, we have no way to know at this stage what the outcome will be. Our first reflex was to remove them from the sample, and calculate the defaults rates only on the ones that have either completed or churned for certain. This is a mistake and leads to a "selection bias" in the estimate. More details on this can be found in Appendix 5.

Our conclusion is that there is no real way around this except waiting for the unfinished accounts to become a minority we are comfortable with. We are indeed required to wait a considerable amount of time to have a sample of accounts that we can calculate these churn densities on.

This is a serious limitation, as it would force us to use data dating from several years to train our model. We had to develop a technique for the model to "learn faster".

Learning faster : splitting the repayment in intervals

It would not be acceptable to use data dating 4-5+ years back in time, in particular in a dynamic industry like ours.

What if we wanted to analyze the churn only on the 0-10% interval ? It would be much faster to have cohorts of accounts that reach 10% or churn. If the model was solely focused on this 0-10% interval, it would much more rapidly adapt to recent data.

In other words, to determine the PD on the 0-10% interval, you don't need all your loans to have reached 100% ; they just need to have all reached 10%. The same can be said for the 10-20% interval and so on.

Building on this principle, we decided to build several independent estimates, each one estimating the PD on these intervals : 0-10%, 10-20%, 20-30%, etc... and for each interval, select the most recent data available while not introducing any selection biais in the sample.

That way, our final estimate is built with the most recent data available for each section of the repayment, and can learn from more recent data.

How to produce a more differentiated estimation ? Accounting for payment regularity.

Initially our model behaved this way described above. However this is essentially averaging loan performance, and not picking up any other dimension except repayment progress. Provisions made under this model started with an "average value", and did take a long time to adapt to the true level of risk.

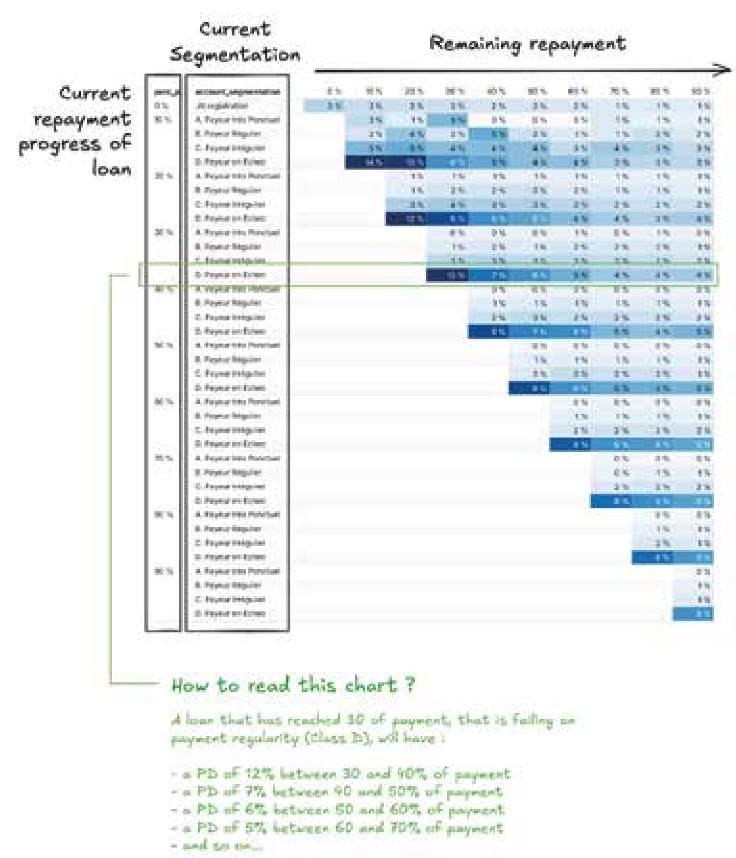
In 2024, we noticed this harmed significantly the performance of our predictions, and decided to take into account payment regularity as a main driver of future performance (a strong correlation we've witnessed).

We bucketed our loans into 4 segments : "A", "B", "C", and "D". The precise definition of those segments does not really matter. Any other segmentation you find relevant to classify various levels of performance (not limited to payment regularity) in your portfolio could be relevant

In the future, we might enhance these segmentations to take into account other factors, such as customer profile, satisfaction, geography, etc...

Before calculation - we form groups based on these segments, and do a completely independent calculation of the PDs using the previous method.

Here is an example of final output we get combining the two previous extensions :



This kind of matrix is interesting on its own to analyze, to understand how the risk is distributed, depending on payment behavior, and to inform appropriate decision making

Please note that we are using here a very basic way to factor in additional dimensions : forming groups, and averaging each group independently. This is easy to understand, troubleshoot, and works well when you have a large amount of data, and a limited set of segments to group on. If we were to factor in many additional dimensions, we would have to switch to a multi-dimensional regressive technique.

This is not much more complicated - as it would only consist of changing the way of computing the PDs based; the rest of the methodology and the data would remain the same.

Model #2 (PD for S2/3) Late Loans Expected Loss

Estimating the probability of never paying again

For the moment, in all we've described, we are missing the risk associated with accounts that are late on payment, that are disabled. The methodology described above does evaluate the risk associated with future repayment, but does not account for the risk associated with a situation without any further payment. In other words, the above methodology assumes if there is 0 repayment occuring, there is 0 risk.

Accounts that have been disabled for some time (60 days for example), have a particularly high chance to end up never paying again. To account for this - we need to add another model specialized in determining the chances a disabled account will never pay again. This also makes our provisioning much more reactive to evolutions in portfolio performance.

Fortunately - this is a much simpler model to build. And it doesn't come with so much complexity or challenges.

We are going to use survival analyses, to determine the probability of "never repaying again". As historical data, we are looking at historic repayment periods.

Although this dataset is easy to understand, it can be quite tricky to build. It will have the following form :

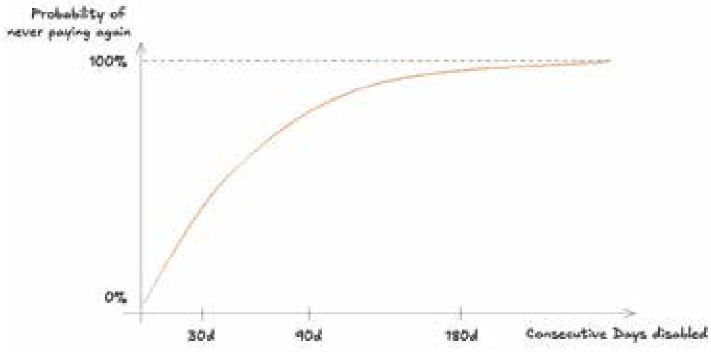
Account	Duration (days)	Outcome = has the customer paid
A	3	Yes
A	43	Yes
A	32	No
B	92	No

If customer A has spent 32 days disabled, and has not paid yet, that doesn't mean we're certain he won't pay in the future. However, the fact that the customer hasn't paid in 32 days still gives us some information.

Survival analysis excels at solving this kind of problem (handling data where the outcome is not certain yet). Conveniently, we'll be able to use 'fresh data' - without having to 'wait' as it was the case for the previous model.

Expected outputs

The typical output is a curve that will look like this :



For a given number of days of disablement, we thus have the short term probability of never paying again. This directly gives us the short term credit loss associated with the account being late.

Similarly to before, having found that these curves tend to have very different profiles depending on the repayment progress (if the account is x days disabled at the beginning of the repayment, it's much more risky than in late stages of the reimbursement), we also grouped the data into several buckets depending on the progress of payment to calculate a set of this kind of curves. We could replicate this approach on any other meaningful driver.

Model #3 (LGD) Expected Recoveries

Several companies, particularly the ones present in structurally high churn markets, rely on an asset recovery process : repossessions. Recovering unused assets and repurposing them not only makes economic sense but also helps reduce waste.

For Baobab+ this must be part of the profitability equation. However, in order to project any economic advantage out of it, we need to carefully evaluate :

- Our ability to repossess a device
- Our ability to get value from a repossessed device

We believe these questions can more easily be answered by companies using historical data (assuming the data is traced and available), and will thus not detail them so much.

Here are brief indications on how this is done at Baobab+.

- **Probability of repossession :** we calculate for every contract a probability of repossession. This is based on repossession rates for defaulting accounts, and factors a discount based on time to repossession. We use survival analyses to compute this estimation.

- **Repossession Value :** For every repossessed item, we track the lifecycle of the device, and are able to determine which portion was sold again refurbished, used as a replacement, or did end up scrapped. This analysis helps us determine an appropriate calculation of an expected value we can get from any repossessed item.

Putting the pieces back together

Recombining outputs

Let's take a look at our initial formula :

$$ECL = \left(\int_{current\ state}^{completion} PD * EAD\right) - \left(\int_{current\ state}^{completion} PD\right) * P_{repossession} * NPV_{repossession}$$

Model 3 is covering the probability of repossessions and their NPV. However, model 1 and 2 need to be layered and combined.

To illustrate this, let's consider a late loan that reached 60% of repayment (40% are still remaining to pay) where :

- Model 2 estimates a 20% probability to never repay again
- Model 1 estimates a 10% to default in the rest of the repayment

We will consider :
- This loan has a 20% chance of defaulting "right now".
- This loan has a 80% chance of not defaulting "right now". But, in this case, it has a 10% chance to default during the remainder of the repayment period. Thus we'll consider 80% * 10% = 8% chance to default later in the repayment.

Total probability of default is therefore : 20% + 8% = 28%.

Similarly, for estimating the expected loss PD * EAD :
- This loan has a 20% chance of defaulting "right now". In this case 40% of initial loan value would be lost.
- This loan has a 80% chance of not defaulting "right now", and in this case the expected loss would be spread as follows :
  - 4% * 35% = 1.4% of initial loan value between 60 and 70%
  - 3% * 25% = 0.75% of initial loan value between 70 and 80%
  - 2% * 15% = 0.3% of initial loan value between 80 and 90%
  - 1% * 5% = 0.05% of initial loan value between 90 and 100%

> The total expected loss would be (20% * 40%) + 80% * (1.4% + 0.75% + 0.3% + 0.05%) = 8% + 2% = 10% of initial loan value.

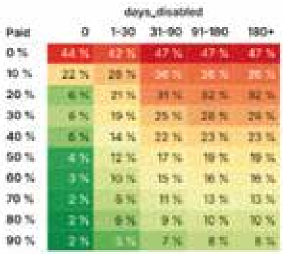
Appendix 6 is providing the source code for the implementation of the full methodology, combining results of the individual models into an ECL.

Provision Matrices

Although the projections are computed and delivered on the loan-level, for them to be better understood, we usually re-aggregate them in the following Matrices before sharing them to the organization.

The following matrices are illustrating the combination of the various model outputs applied on our current portfolio. Each matrix displays the output in % of remaining receivables, but could also display monetary values. They can be quite handy to explain the individual outputs of the model, and how they apply to different sections of our portfolio.

Model #1
Lifetime Expected Loss



Paid	0	1-30	31-90	91-180	180+
		days_disabled			
0 %	44 %	42 %	47 %	47 %	47 %
10 %	22 %	28 %	36 %	36 %	36 %
20 %	6 %	21 %	31 %	32 %	32 %
30 %	6 %	19 %	25 %	28 %	29 %
40 %	6 %	14 %	22 %	23 %	23 %
50 %	4 %	12 %	17 %	19 %	19 %
60 %	3 %	10 %	15 %	16 %	16 %
70 %	2 %	5 %	11 %	13 %	13 %
80 %	2 %	6 %	9 %	10 %	10 %
90 %	2 %	3 %	7 %	8 %	8 %

+

Model #2
Late Loans Expected Loss

Paid	0	1-30	31-90	91-180	180+
		days_disabled			
0 %	0 %	12 %	49 %	94 %	97 %
10 %	0 %	7 %	40 %	79 %	94 %
20 %	0 %	6 %	34 %	68 %	89 %
30 %	0 %	6 %	32 %	68 %	89 %
40 %	0 %	6 %	32 %	66 %	89 %
50 %	0 %	5 %	24 %	50 %	78 %
60 %	0 %	5 %	24 %	52 %	77 %
70 %	0 %	5 %	24 %	52 %	78 %
80 %	0 %	4 %	19 %	41 %	73 %
90 %	0 %	2 %	11 %	26 %	48 %

+

Model #3
Expected Recoveries

Paid	0	1-30	31-90	91-180	180+
		days_disabled			
0 %	9 %	9 %	9 %	7 %	2 %
10 %	6 %	7 %	9 %	7 %	3 %
20 %	2 %	7 %	10 %	8 %	3 %
30 %	3 %	7 %	10 %	9 %	4 %
40 %	3 %	6 %	11 %	10 %	4 %
50 %	2 %	7 %	11 %	12 %	5 %
60 %	2 %	7 %	13 %	14 %	7 %
70 %	2 %	8 %	15 %	17 %	8 %
80 %	2 %	8 %	18 %	23 %	11 %
90 %	2 %	7 %	17 %	27 %	20 %

=

Result
Expected Credit Losses

Paid	0	1-30	31-90	91-180	180+
		days_disabled			
0 %	35 %	43 %	66 %	85 %	96 %
10 %	16 %	25 %	52 %	77 %	94 %
20 %	4 %	10 %	45 %	68 %	89 %
30 %	5 %	17 %	39 %	66 %	88 %
40 %	3 %	13 %	35 %	64 %	88 %
50 %	2 %	9 %	26 %	48 %	77 %
60 %	1 %	7 %	22 %	46 %	74 %
70 %	0 %	4 %	17 %	41 %	72 %
80 %	0 %	2 %	8 %	21 %	65 %
90 %	0 %	0 %	0 %	5 %	33 %

Call to Action : Future work

This section outlines the opportunities we see for expanding and improving upon our current framework, as well as inviting contributions and insights from our industry peers.

By focusing on both methodological enhancements and collaborative initiatives, we aim to strengthen our models and build greater trust across the industry, ultimately supporting sustainable growth and financial inclusion for underserved communities.

How to estimate credit risk for a new company or product ?

As we've seen, due to the long time to insight and potential biases, this methodology requires having several years of data to be able to run these models trained on your own data. In most cases, you can't afford to wait several years to form an opinion on future credit risk.

Default approach : by analogy / benchmark

A first approach is to use another company's model or benchmark to estimate your future losses. At Baobab+ this is how we estimate the risk of new product lines, by analogy with existing ones.

In our experience, this approach is better than nothing, and is necessary for provisioning - but we've seen estimates are quite often failing backtesting. We recommend our operations taking these estimates with a grain of salt, and to not rely too heavily on them.

Potential other approach : parametric risk profile

Another idea would be to find a "common ground" in the behaviour of multiple companies and products, and be able to use early default data to project later default data.

When looking at our various markets and products, and accounting for loan quality, we observe quite similar trends, for example :

- Loans that exhibit a good repayment at the beginning (= high quality sales) tend to have a uniform risk profile across the repayment.
- Loans that exhibit a bad repayment at the beginning (= bad quality sales) tend to have a risk profile skewed towards the beginning of the repayment.

This matches intuition, as "good payers" will still be vulnerable to external stress (life events) affecting their ability to pay, like the death or sickness of a relative. These events are not likely to be predictable or avoidable.

On the other hand, for "bad payers", the risk is more short term : either the customer will adhere to the rules and come back to a righteous path, or will default early.

Although this topic deserves more research to lead to actionable outcomes, this is an encouraging hint that there might be a kind of 'parametric' curve that could fit different contexts.

Building on this assessment, we could define a "parametric typical default rate distribution". This parametric risk profile could then be projected based on the early churn rates we are observing.

How to monitor & manage our credit risk performance ?

Once we have all these models in place, it is still a challenge to clearly communicate to your stakeholders your performance. Here is how today we are presenting credit risk performance at Baobab+.

We usually split performance measurement on new sales (Year to date) vs historic.

New Sales Performance

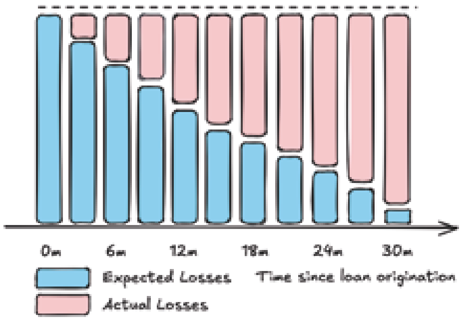
Your performance on new sales is going to be the forward looking measurement for your profitability model. To show performance and improvements, we can compute the credit risk (we can decide to include or not the effect of repossessions) on the cohort value, summing the actual and expected loss.

As you can see, our last 3 cohorts will be too young to form an estimate that is really measuring anything meaningful. This is because our model will have a first reliable estimate once the customers have demonstrated their payment behavior.

This kind of chart helps to form an opinion on the current level of performance, and its evolution.

Historical portfolio performance

In theory, if your historical portfolio is accurately provisioned, you should observe this kind of behavior in your actual and expected losses.



However, it is rarely the case in practice. We regularly observe variations coming from :

- **A change of model / judgment :** our model will be recomputed dynamically (every month), and the same situation could be evaluated differently by our models at two different points in time. Also - we can make significant methodology adjustments that create gaps in what we had provisioned in the past.

- **A change of situation :** the previous chart assumes the portfolio evolves as expected historically. If our customer care efforts decline, we can assume our portfolio quality will degrade (more than expected). In these cases, our credit risk will increase.

Let's assume we want to explain why the credit risk on the historical portfolio of the 31/12 of last year did increase.
To split the variation in these 2 sources, which do have very different interpretations, we proceed in two steps :

- **Estimating the "Model Gap"** : by applying the latest version of the model to the portfolio situation on the 31/12 of last year. The same situation will be 'judged' differently by two versions of the model. If we see an increase in the expected losses, that means our model has become more pessimistic about the same identical situation.

- **Estimating the "Portfolio Evolution Gap"** : by applying the latest model to both the situation on the 31/12 of last year and today. Calling this gap the "portfolio evolution" gap assumes the model we're applying is "right". In reality this gap could also indicate a lack of precision of the current model - but it would be quite challenging to go further and to quantify what's real degradation and what is lack of precision.

Building regression models

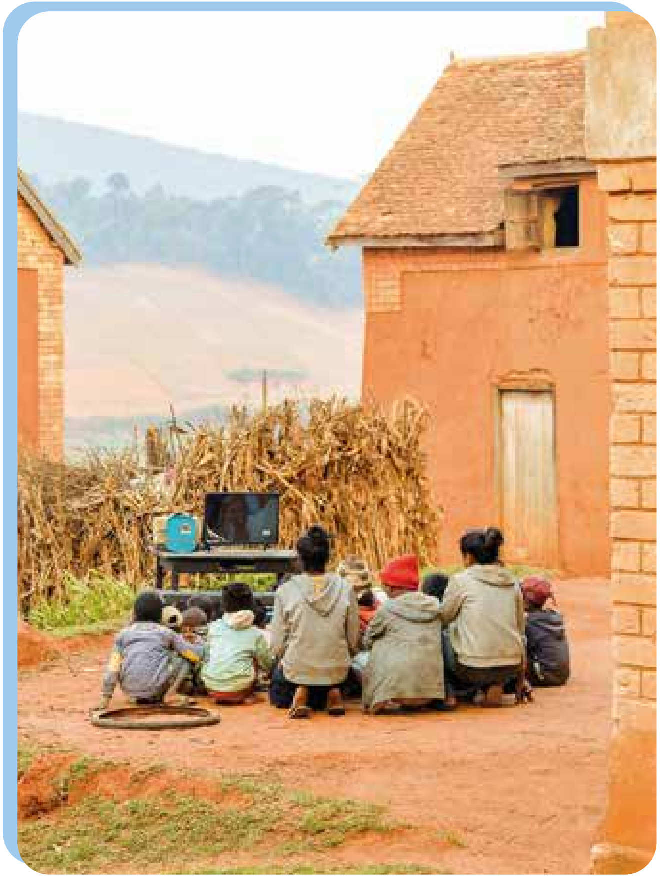
Right now, our models are blind to most of the data our company is collecting. The only data required to produce our estimates is transactional data. However, we capture plenty of data at each customer interaction: on average, we'll have dozens of touchpoints with every customer during their lifetime.

This data can tell us a lot about the risk of a customer - if the customer is constantly breaking his promises to pay ; if the customer is difficult to reach ; not collaborative.

Also, internal data such as the sales agent general performance or the proximity with our sales network, could be leveraged.

Currently our models do not take this data into account, but the foundation is there to use much more sophisticated models forming an opinion using every piece of data that could carry some signal.

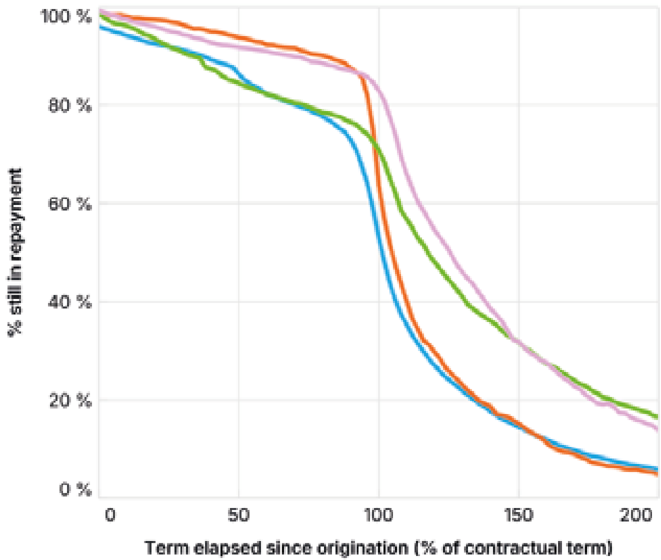
This is a research area that could lead to valuable operational results, and would be quite valuable to delve into.



Appendix 1 : Extended time to completion

The following graph shows the portion of customers still "in repayment" depending on the term elapsed for 4 of Baobab+'s countries / products.

These accounts have neither unlocked their product, nor been sufficiently late to be considered in default (180 days late) or repossessed.



Here are some takeaways :

Measuring completion at 1x contractual term seems to be a very volatile and misleading measurement, that we would recommend not to track, even though it might sometimes be asked by external parties.

Even at 2x contractual term, we have a (not negligible) fraction of our customers that are still repaying

The shape of these curves is quite representative of the credit culture of the concerned countries.

Caveat : These curves were formed on very historical data at Baobab+ (before 2020), since then, credit management has changed drastically. The curves linked to today's operations could be very different.

Appendix 2 : Practical guide to cohort analyses

Open sourced code to implement this methodology can be found in appendix 6.

Input datasets

We are going to assume that companies have at their disposal two key datasets :

- Dataset 1 : “loans”, containing the loan characteristics such as time of origination, payment plan, etc...
- Dataset 2 : “payments”, containing all individual transactions on the loan, with their amount and time

Simplifications

For simplicity, this calculation assumes that :

- Loan characteristics are going to be static.
- Token generated without a payment will be ignored.
- Every payment will generate a number of days based on the payment plan without potential arrears.

In practice - your company might not fit 100% these assumptions, in which case you should assess if they need to be factored in the methodology or not.

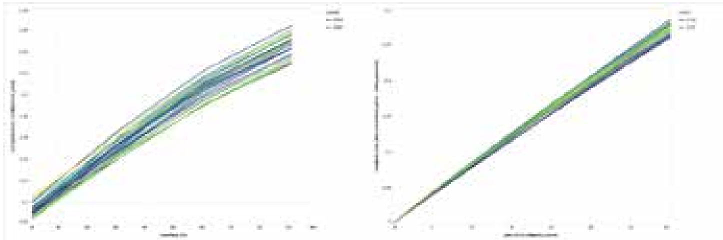
Metrics

First, we encourage companies to analyze cohorts separately for each important country - product (in some cases even offers) combination.

The first metric to display is most probably the raw repayment of each cohort. This is labelled as “beginner version” in the code sample.

Beginner vs Advanced Approaches

The code repository outlines 2 different approaches, the beginner one will focus on displaying the raw payments across time, providing a good first level of insight. The advanced one is implementing the following improvements, but results in a more complex implementation.



This chart is showing the early repayment of the beginner vs advanced approach. With the advanced one, the cohort early evolution is much easier to interpret, and more predictive of future outcomes.

Improvement #1 : Removing free initial period

PAYGo loans typically have a first initial period paid by a deposit. This period doesn't tell you much about the repayment of a customer, as they’re not expected to pay. Various pricings may have different amounts or durations for that initial period.

In order not to bias cohort early evolutions due to this initial period, and enhance comparability, we can remove this initial period and start displaying data at the first day of repayment.

Improvement #2 : Linearizing payments

Even though you can use raw payments to gain insight quickly, we strongly recommend switching to linearized payments if you have the data processing capacity.

Linearized payments or cumulative usage consists in smoothing payments over the period they are granting access to the devices. This ensures the % paid will always be lower than the % term elapsed, and will avoid having a few large payments completely distorting the picture and hiding many underperforming elements.

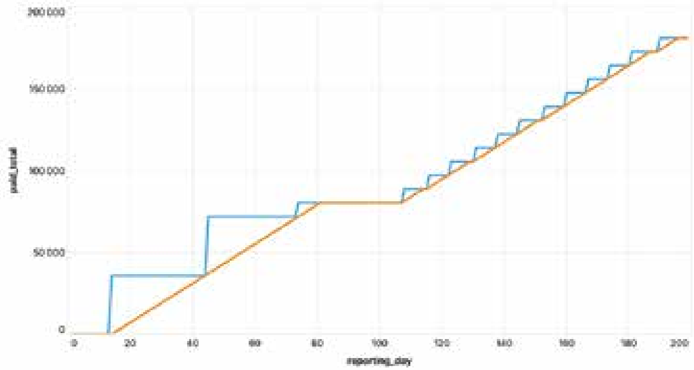


Chart : illustration of payment linearization on a single loan's payment profile.

Improvement #3 : Duration in % of contractual term

When tracking performance of multiple offers within one product group, you won't be able to compare for example the repayment of 12m offers vs 24m offers at 6m. In these cases you should switch to tracking metrics vs % contractual term; to have a comparable metric across various repayment durations.

This brings an extra complexity - you should always make sure your cohort has a static number of accounts. If you have a mix of 12m and 24m offers; the 12m will reach 50% of the contractual term in 6m ; whereas the 24m loans will only be at 25%. This could introduce significant biases in your analyses, and you should measure this cohort performance until 25%.

For this - we apply a condition in our logic to stop displaying the cohort evolution if the number of loans becomes lower than the total number of loans in the cohort.

Tip : Always include unlocked or defaulted loans in metrics

To improve interpretation, we usually make the choice to only display metrics that apply to all loans in the cohort. For example, if we display PAR 30, we will extend the definition of the metrics to apply also to unlock loans (considering them enabled) and defaulted loans (considering them disabled). This gives us more interpretable information.

Appendix 3 : Benchmarking the cohort projection methodology

Open sourced code to implement this methodology can be found in appendix 6.

Approach

To benchmark the performance of the cohort methodology against our IFRS9 methodology, we used the approach described in the document to generate artificial data and form various scenarios. It consists of :

- Selecting a pool of historical loans with various levels of quality (A, B, C and D).
- Defining scenarios, where we control the quality of each cohort with the proportions of class A, B, C and D.
- Randomly select in the pool of historical loans to assemble the artificial cohorts, separated into cohorts to learn from and cohorts to test on.

For the cohort methodology, we couldn't easily separate the data used for training and the data used for testing. But we don't think this is putting at risk the conclusions, as the methodology itself is rather coarse, and it is not at risk of overfitting.

Two alternative versions of the cohort projection methodologies were tested :

- A projection based on the 6-month average collections of month n, n+1, n+2...
- A projection based on the 6-month average collections of month n, n+1, n+2... corrected by a scale factor determined on month n-1.

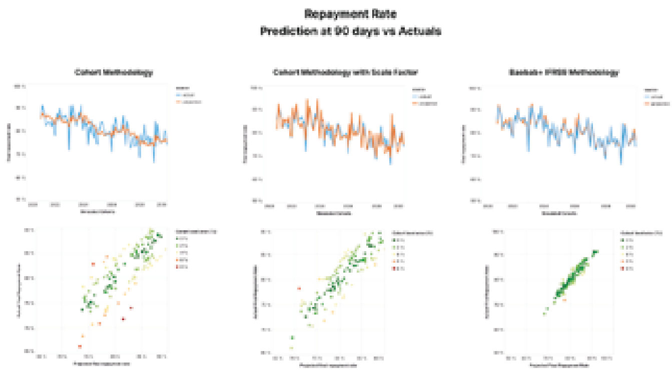
For more details on these methodologies, we would refer you to Guillaume Cruyt's paper (for data professionals : you can look at the code sample).

For the Baobab+ IFRS9 methodology, we simulated a certain number of cohorts for the model to learn from (training set); and then a different set of cohorts to test on (test set). We made sure no loan used in the training set was used in the test set.

We did test several scenarios; but the one we would like to share here is inspired by historical Baobab+ variations and trends.

Results

- The top chart shows the estimated repayment rate (orange) vs the actual one (blue).
- The bottom chart represents the projected repayment rate at 90d vs the actual one. Each dot representing a cohort, a perfect model would have all cohort fitting a straight line.



- The cohort methodology in its basic form already provides quite satisfying results, picking up trends in performance, but fails to adapt to individual cohort performance.
- The added value of the scale factor appears clear here, bringing much better cohort level accuracy to the projections.
- The IFRS9 methodology was the most accurate one, estimating well individual cohort performance.

While these results look very encouraging, we need to face the fact that this way of testing our model is not representative of reality.

Any science experiment will be first tested in a laboratory setting : that's what we are doing here, to confirm beliefs and behaviour. But a laboratory setting is not the same as real conditions.

The method used to generate artificial cohorts did only take a pool of historical loans that have originated somewhere between 2018-2020, and used this historical data to artificially generate new cohorts.

In practice, our credit management approach has changed, and it is likely that the payment behavior of a loan originated in 2018 will not be the same that one originated in 2024, even with identical metrics.

The real complexity of this topic is that any model will do nothing else but to use past situations to predict the future. If the relationship between a situation and outcomes changes with time, even a perfect model won't be able to accurately predict future outcomes.

In other words, there is some sort of glass ceiling to the accuracy we can expect from our predictions. Even a very powerful model will be challenged by underlying changes in behavior it is trying to predict.

This is the reason why, according to us, management teams should never fully trust their models, and should feel accountable and empowered to use their judgment to correct the estimates for planning their activity.

Appendix 4 : Limitations of PD time based Models

Many financial institutions are adopting a time-based approach, estimating :

$$Amount\ not\ collected = \int_{t=0}^{completion} PD\,(t) * EAD(t)\,dt$$

$$Default\ Rate = \int_{t=0}^{completion} PD\,(t)\,dt$$

This approach is difficult to apply for PAYGO loans for the following reasons :

Linking PD & EAD : To effectively link churn rates and credit losses with this approach, you need to pair a PD(t) model with another model estimating repayment progress across time, EAD(t). Indeed - assuming we have a perfect default rate prediction across time, we would still need to know the EAD. A very powerful model might be spoiled by a very poor way of modeling future EAD.

This was the case for us at Baobab+ in our first iterations of the model, where we paired this output with a very rough model to estimate repayment speed : essentially just multiplying the contractual payment plan by a factor (e.g. +30%).

Time horizon : Given the long tail on repayment times it can be quite tricky to define an appropriate time horizon to consider “lifetime”.

Lagging default recognition : Our default definition leaves room for delaying default up to 6 months depending on company policy. This would blur the estimation of PD, and affect model performance. Also, in case of company operational policy change (e.g. decision to repossess earlier than before), we might interpret temporarily the change of policy as a much higher risk.

Appendix 5 : Illustration of selection bias for PD estimation

To explain this, let's assume we are looking at a cohort of 100 accounts, and are looking at their status @150% of theoretical term :

- 50% have paid off already
- 20% have already churned
- 30% are still paying

If we assume the 30% will eventually pay in full - we'll assume our total churn rate to be around 20%. If we assume they will all churn, the churn rate will be 50%. The gap and risk of error can be significant, even when we have a minority of accounts that are still paying.

We also can't pretend these accounts do not exist, and only calculate our churn rates based on the ones that have paid and have churned. As the risk of these 30% remaining is probably higher than the average account. If we do so, we might ignore part of the sample where there is a significantly higher risk concentration. In practice, slow payers are indeed particularly risky customers.

We ran into this issue trying to build the model. The outputs we got were completely off track, and failed every test. The outputs of the model are very sensitive to the assumptions we're going to take here.

Our conclusion is that there is no real way around this except waiting for these 30% to become a minority we are comfortable with. At Baobab+ we use a sample of accounts in our training data when less than 2% of accounts are still paying. On these 2% of accounts where the outcome is uncertain, we assume a future probability of churn - higher than the average (typically 40%).

As you can see from this example, even at 150% of repayment, it is still challenging to have an accurate picture of this target output. We are indeed required to wait a considerable amount of time to have a sample of accounts that we can calculate these churn densities on.

Appendix 6 : Open Source Code Repository

Code repository

The code is hosted on a github public repository and can be found here.

Input datasets & Artificial Data

We are going to assume that companies have at their disposal two key datasets :

- Dataset 1 : "accounts", containing the loan / account characteristics such as time of origination, payment plan, etc...
- Dataset 2 : "payments", containing all individual transactions on the loan, with their amount and time

The datasets present in this open source repository are anonymized loan & payment data replicated to artificially augment the volume.

Simplifications

For simplicity, the calculations will assume that :

- Loan characteristics are going to be static.
- Token generated without a payment will be ignored.
- Every payment will generate a number of days based on the payment plan without potential arrears.

In practice - your company might not fit 100% these assumptions, in which case you should assess if they need to be factored in the methodology or not.

Core Dataset : Building historical snapshots of loan & their repayments

The core dataset that will be necessary for a variety of analytics use-cases is a dataset describing each loan's historical situation. It contains a snapshot of each day since loan origination.
If a loan originated 365 years ago, this dataset will contain 365 lines for this loan, with each line representing its metrics from day 1 to 365.

- The field called account_id will capture the loan's unique identifier.
- The field called reporting_day will capture the day since loan origination.
- The field called reporting_date will capture the date of that snapshot.

This dataset serves as raw material for most of our credit risk analyses and KPIs, including cohort analyses visualizations and credit risk projections.

The basic version contains only the total paid amount at a specific point in time, but does not include any data about how many days are remaining on balance according to the contractual terms.

The advanced version of the calculation includes a day-by-day calculation of the remaining balance, and enables us to recompute the historical balance. This balance enables us to know if the customer was enabled or disabled.

When available, historical data about remaining balance, either provided by the source systems or recorded by the company using it can complete the picture and avoid this relatively complex calculation.

Cohort visualizations

Once the core dataset is built, displaying the cohort analyses is only a matter of aggregating and filtering the data.

Though, there is some complexity in :

- Calculating estimates based on % of loan term instead of days
- Filtering out the end of the cohort, where estimates are based on not the full cohort but a reduced number of loans. This produces irrelevant estimates, and gives an inconsistent shape cohort if not cleaned out.

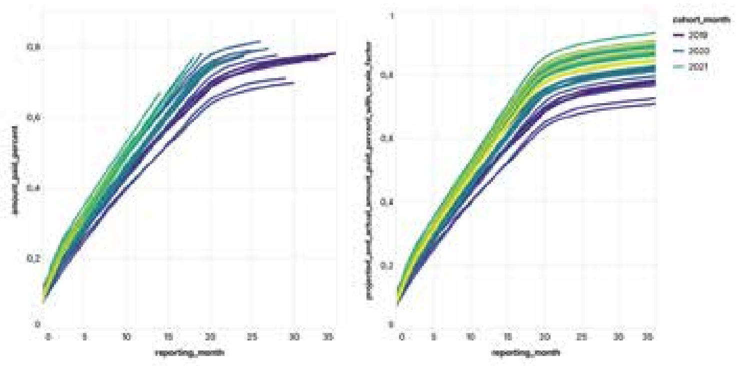
Cohort Projection Methodology

Two versions of the cohort projection methodology are provided in the code, computed in the same table as separate fields.

- The first one built on a simple average over the last 6 available cohorts
- The second one using the scale factor

Both are calculated on the beginner version of the dataset. This methodology could potentially yield better results if computed on the advanced version of the core dataset.

Chart : Visualization of the cohort projection methodology, from partial historical cohorts to projections.



Full implementation of Baobab+ IFRS9 Methodology

This implementation is split into three parts, one for each model described in this paper. Eventually, the model outputs are then recombined into a single ECL estimate for each historical situation of the portfolio. We do not only predict future ECLs for the current portfolio state, but also do a prediction for each historical daily snapshot of our portfolio. This is crucial to be able to backtest the model.

baobab+

energy, digital & more