MDSAA

Master Degree Program in

Data Science and Advanced Analytics

Business Cases with Data Science

Case 4: Business Process Conclusion Prediction

Ana Teresa Maia, 20202562
Henrique Falcão, 20201519
Maria Benedita Elias, 20230491
Maria Leonor de Gusmão, 20230488
Pedro Carvalho, 20230487

Group H

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa
May, 2024

INDEX

1. EXECUTIVE SUMMARY

This project, aimed at enhancing Millenium Bank's operational efficiency through predictive analytics, denotes our commitment to modernizing banking processes and improving customer satisfaction. By implementing a predictive model for business process conclusions, we set out to streamline workflows, optimize resource allocation, and elevate strategic decision-making.

The initiative employed the CRISP-DM methodology, guiding us from initial business understanding to final deployment. The primary goal was to develop a model capable of accurately predicting the outcomes of various banking processes. This foresight allows for proactive management, leading to more efficient operations and better customer service. Our model showed promising performance relative to the literature found in our research, with f1-scores from 0.64 to 0.76, achieving a strong balance between precision and recall, which underscores its reliability in making accurate predictions.

Throughout the project, we analysed a comprehensive dataset, which included task execution details, user information, request specifics, and rejections. This rich dataset provided critical insights into operational inefficiencies and customer service bottlenecks. For instance, frequent task rejections and prolonged execution times were identified as key areas for improvement. Addressing these issues will not only improve workflow efficiency but also enhance client satisfaction and retention.

Looking ahead, continuous monitoring and regular updates to the model will be essential. Future improvements should focus on overcoming data incompleteness and computational limitations encountered during this project. By refining the model and integrating it seamlessly into the bank's existing systems, Millenium Bank can look forward to a more agile and responsive operational framework, ultimately fostering a more competitive and customer-centric business environment.

2. BUSINESS NEEDS AND REQUIRED OUTCOME

2.1. INDUSTRY OVERVIEW

In today's dynamic business environment, organizations are relentlessly pursuing methods to enhance operational efficiency, fuel digital transformation, and sustain competitiveness. Business Process Management (BPM) emerges as a pivotal strategy, serving as a guiding principle for streamlining processes, optimizing workflows, and leveraging digital technologies to attain heightened levels of efficiency and effectiveness. BPM encompasses a comprehensive array of methodologies, tools, and techniques tailored to refine organizational processes, foster seamless operations, and enhance performance. Acting as an incentive for digital transformation, BPM collaborates seamlessly with technologies like Intelligent Automation to propel innovation and efficiency across the enterprise.

The adoption of BPM brings forth several advantages crossing different dimensions of organizational operations. From enhancing productivity by eliminating bottlenecks and optimizing processes to driving cost reductions through the identification and elimination of inefficiencies, BPM fosters operational excellence and financial optimization. Moreover, BPM offers enhanced visibility and control over processes, empowering organizations to track items through workflows and streamline operations. By driving digital transformation initiatives with Intelligent Automation, BPM enables organizations to navigate evolving market dynamics and technological advancements, positioning them for sustained growth and competitiveness in the ever-evolving business landscape.

Traditional core banking systems were designed in an era when introducing new products was a lengthy process that often disrupted operations. However, over the last decade, banking operations have become more sophisticated both in front and back offices.

The influence of millennial culture has led to a surge in omnichannel banking. Nowadays, banking services and customer information must be accessible through a variety of channels, including ATMs, online banking, mobile banking, and IVRs (Interactive Voice Response). Delivering a seamless customer experience across these platforms is essential for banks to stand out in the market.

A recent study by Fortune Business Insights forecasts that the Core Banking Software Market will reach USD 28.83 billion by 2027. This growth is largely driven by the increased adoption of SaaS and cloud-based banking software solutions. By implementing automation processes, banks can better align their operations with strategic business goals. BPM software is crucial for automating key banking workflows such as account opening, KYC verification, loan processing, and compliance management. Leading banks and financial institutions turning to BPM solutions to minimize errors in workflows and accelerate processing times.

According to IDC Research there is a 20% to 30% revenue loss on inefficient processes. In addition to that, 60% of work time lost is on lack of coordination in business processes and 4h38m per week are spent on tasks that already have been completed.

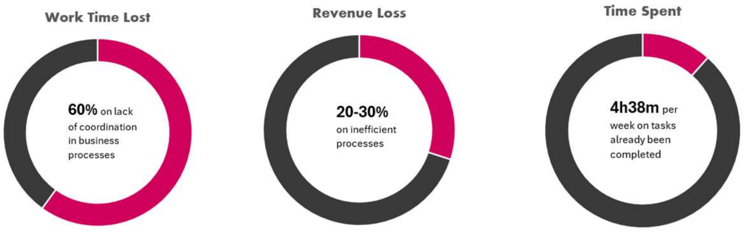


Figure 1 - Impact of Inefficient Processes on Revenue and Work Time

This is where BPM primary advantages of automation in the banking and finance sector can aid, since these include: decreased employee workload, cost and time efficiency, enhanced operational agility and flexibility, improved customer service levels, increased customer trust and loyalty and enhanced regulatory compliance.

Bill Gates famously said, *"Banking is necessary, but banks are not.".* While banking remains an essential part of daily life, digital disruption has significantly changed how people interact with banks. Tech-savvy millennials now expect a seamless experience across various digital channels. Automation in banking has transformed the industry's business processes, prompting banks to rethink and redefine their customer interactions. Digital banking focuses on providing exceptional customer experiences across multiple digital platforms.

The growing demand for digital banking services has led to the emergence of new technologies that are reshaping the banking industry landscape.

2.2. BUSINESS OBJECTIVES

When tailoring the strategy adopted to come up with business objectives, the approach passed by you branched out from overall objectives that impact the business and financial performance of the bank, to directed objectives related to the process under analysis. As so, our team made a priority understanding them, which later translates in the methodology we adopted to develop our solutions.

On a broader level business objectives encompass:

- **Understanding client needs accurately:** Client´s requests can be ambiguous due to the subjectivity of each individual request, however by training specific models that focus on these types of challenges helps enhancing client understanding, which, in the future, will lead to a higher client satisfaction rate and a better performance from robots in the department, in terms of pattern recognition.
- **Reduce financial expenditure on miscommunications:** Another objective from your side is to tackle down the financial burden associated with the failure to predict early on the outcome of the requests which, when prepared for, allows a lower financial expenditure on issues of the kind, improving the bank financial performance, attending stakeholders needs. According to IDC Research around 20-30% of revenue loss is allocated for these expenditures.
- **Reduce internal systems complexity:** Currently, internal systems are not homogeneous, which makes it difficult to have a quick request response. After a trained outcome prediction, the desire is to standardize the method used to make all systems concise, enhancing efficiency.

- **Sustain rapid creation of workflows**: With an automated outcome prediction workflows will not only be rapidly created but also planned ahead, which translates in better resource allocation and department management strategies.

Moving to the process itself, the defined objectives go through the following topics:

- **Process outcome prediction in 2 separate stages:** The most valuable take from this project is the enablement to predict the outcome of a specific process inside the bank workflow. These predictions should be designed to allow understanding of the outcome at two distinct stages- when the request is initiated and when it is being executed. This gives more flexibility and adaptability to act accordingly in different levels of the process flow. There are 4 possible outcomes, which will be described in section 3.2.
- **Understanding of relevance of given features:** How to control and be aware of any red flags in daily operations goes through having a strong knowledge of what variables/factors impact each outcome on a deeper level.

After the meeting with your representatives, we were able to understand the relevance of these objectives and design strategies to answer each one of them.

2.3. BUSINESS SUCCESS CRITERIA

To translate the eventual success of the implementation of this project our team came up with three different success criteria in the form of KPIs to measure the improvement brought by our work.

- **Client Retention Rate Increase:** With a detailed forecast of the conclusion of each request it is possible to better advise each client and provide a more tailored experience when guiding them through the process. This is of high interest of banks to achieve since it promotes client retention, lifetime value and in the end translates into competitive advantage. Banks that accurately predict processes' outcomes will be assigned to a better reputation, attracting more clients and increasing their intangible assets. Having satisfied customers will improve the overall business flow of the bank being the reason we identified it as an important KPI. *(Reference 4)*
- **Profit and Loss account recovery:** The level of financial expenditure associated with the outcome of the processes is of high impact on the bank's revenue loss, as seen before. This financial statement is highly relevant on the industry Millenium operates in. It not only provides a sense of financial health of the bank but also is a factor to attract investors and clients that provide funds to the bank, its value is directly connected to the success of the implementation.
- **Automation Rate Increase:** Since one of the main goals is to reduce internal complexity and a steadier workflow, measuring automation rate enables you to assess how efficient the solution proposed is. It concerns resource allocation plans and other efficiency-relevant factors.

With these relevant KPIs it's possible to have a relevant proxy for how good the implementation is on your internal systems.

2.4. Situation assessment

The foundation of our project lies in a comprehensive dataset containing four distinct sheets of information, each providing insights crucial to our exploration and subsequent project development:

- *Q1 - Task execution data*: This sheet offers detailed metrics on every task within Millenium's operational framework. From identifying the task requester to delineating its lifecycle from initiation to completion. Key attributes include start and end times, predicted completion estimates, task paths delineating the sequence of activities and actions, the executing department, task type, and a group of final action outcomes.
- *Q2 - User information*: Complementing the task execution data, this sheet gives insights into the profiles of task executors. From demographic details like gender and birth year to professional attributes such as role and management status, this dataset paints a comprehensive picture. Furthermore, it provides information into the tenure of employees within their respective departments and highlights whether they are internal staff or outsourced personnel.
- *Q3 - Specific request data*: This sheet offers granular insights into specific requests within your operations. With 11 distinct ID fields and their corresponding values, this dataset allows for meticulous tracking and analysis of individual requests, enabling a deeper understanding of operational patterns and trends.
- *Q4 – Rejections*: This sheet outlines the tasks that faced rejection, accompanied by distinct codes representing the various reasons for such outcomes. By dissecting these rejection codes, our team aims to identify underlying issues and implement targeted interventions to mitigate future rejections.

Besides having these datasets our team gathered some information about your bank's main objectives that were reflected on, in the previous sections. At first, we explored the approaches needed for this type of challenge individually, and then, combined our insights to start with the project development with a balanced task delegation with constant feedback. To support our work, we made use of Python programming language in collaboration with Jupyter Notebook.

2.5. Determine Data Mining goals

For the development of this project, our team followed the **CRISP-DM** methodology. It focuses on tailoring projects by iterating through its steps: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation, and Deployment Strategies. On top of that, after careful research, our data mining goals for this specific use case rely on the evaluation of different types of metrics: F1, Precision, and Recall.

- **F1 Score**: The F1 score is the harmonic mean of precision and recall. It provides a single metric that balances both false positives and false negatives, making it particularly useful when the cost of false positives and false negatives are similar. For multiclass classification, a macro-averaged F1 score is often used, which computes the F1 score independently for each class and then takes the average. For balanced classes it is simpler to achieve an F1 score closer to 1, unfortunately, this is not our case. Values around **0.7** are considered good in practice, especially in complex, real-world datasets.

- **Precision**: Precision measures the proportion of true positive predictions out of all positive predictions. It is a critical metric when the cost of false positives is high. In the context of multiclass classification, precision is calculated for each class and then averaged. In the context of banking, false positives do not have such high costs, therefore a precision score around **0.6** is generally considered good.
- **Recall**: Recall measures the proportion of true positive predictions out of all actual positives. It is particularly important when the cost of false negatives is high, such as in health care system. Like precision, recall is also calculated for each class and averaged. A high recall means that most of the actual positives are identified by the model, and a recall score around **0.6** is often obtained in similar projects.

In a multiclass classification scenario like our Business Process Conclusion Prediction, achieving high precision and recall across all classes can be challenging due to the varying distribution of classes and the complexity of distinguishing between them.

In conclusion, using the CRISP-DM methodology has allowed our team to iteratively refine our approach to this multiclass classification problem. By focusing on F1, precision, and recall metrics, we aim that our model is balanced and effective in making accurate predictions across different business process outcomes. High scores in these metrics can indicate a robust model capable of providing valuable insights and predictions for business process conclusions, ultimately aiding in better decision-making and process optimization.

3. METHODOLOGY

3.1. DATA UNDERSTANDING

In this project, we utilized a comprehensive dataset provided by you, organized into four key segments: task execution, user information, request specifics, and rejections. Initially, each section was imported separately into distinct datasets for a detailed examination and subsequently merged into a final dataset for holistic analysis. Some key insights that were identified were the following:

- Task execution data is composed of 45,772 unique requests encompassing 209,017 tasks. There's a need to convert time-related data to 'datetime' format and address missing values in critical fields such as "Task Department" and "Task executer". Some processes also appeared to be incomplete since there was no final activity registered.
- Regarding user information, there are some anomalies in the "Birth Year" column such as years listed as 1901 and 2050. In the "Sex" column some entries are marked as "U" or blank, which was later clarified that it represented automated systems. Predominantly managers and outsourced personnel, influencing the modeling of user interactions.
- Request-specific data is linked to survey data associated with the requests. Although such specifics remain confidential due to privacy constraints, the data will be rearranged to provide contextual relevance to each request, aiding in predictive accuracy.
- For the rejections' dataset, Task IDs were often repeated due to having multiple rejection triggers.

Moreover, as illustrated in the figure below, most of the processes are consisted of less than 10 steps. However, there are a few processes that contain significantly more steps, these are outliers within the dataset. Since these outliers are rare, their unique characteristics could disproportionately influence our model's predictions, potentially leading to less accurate outcomes for the typical cases.

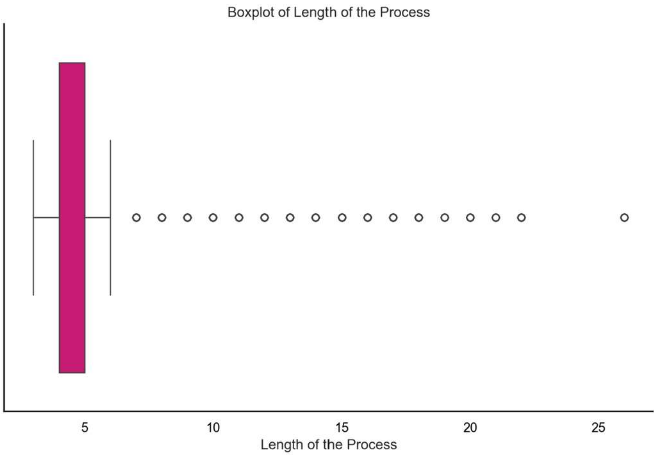


Figure 2 - *Boxplot of Length of the Process*

Lastly, as depicted in Figure 3, approximately a quarter of our dataset consists of requests with looped activities, where certain activities recur within the same request. This observation warrants further analysis to determine if these looped processes exhibit any significant relationships with our target outcomes.

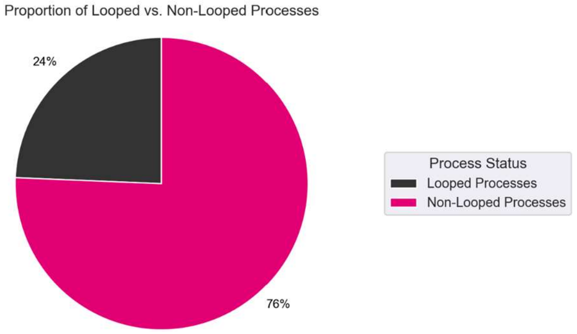


Figure 3 - *Proportion of Looped and Non-Looped Processes*

3.2. Data Preparation

Dealing with Inconsistencies and Interpretability Issues

We addressed the inconsistencies noted in the previous chapter to ensure a solid foundation for our analysis.

Missing data

When managing missing data within our dataset, to better represent the nature of our data and preserve as much information as possible, the missing values in the 'Task executer department', and 'Action' columns were filled with the label 'Non Identifiable'.

Otherwise, the ‘Task executer’ column deserved a different treatment lately. This column is the bridge to merge the Q1(Task execution data) and Q2(User information) data sets. We assumed these types of users as ‘Robots’ for consistency purposes, and since this information was provided by you, and so we filled its values with this label. To complete the user information of these ‘Robots’, we evaluate which would be the best values for each of the features from user information. For example, we defined the ‘Sex’ variable as ‘No Sex’ for these cases.

For the rows related to activity 100, the missing values in the 'Task predicted end date' were not anomalies but rather a result of the step being user-controlled rather than business-controlled.

Other treated cases

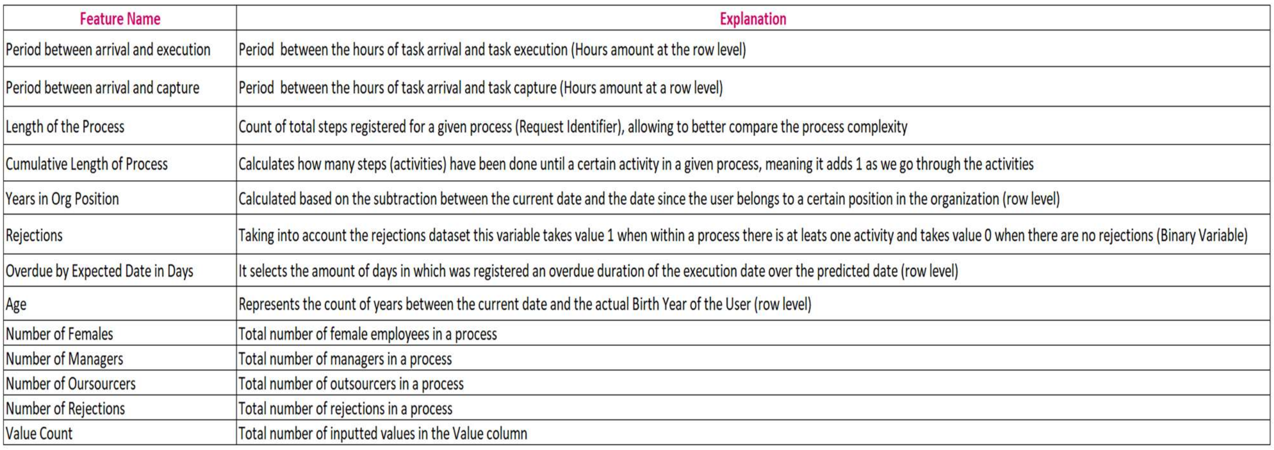
Requests lacking an outcome were excluded from our dataset. These records represent incomplete processes, which are challenging to predict reliably and could potentially skew the model’s performance. With this removal the detected anomalies in the 'Sex' column and unrealistic birth years were primarily found within these incomplete records and disappeared upon their exclusion.

As previously mentioned, analysing rejection data revealed duplications when merging the Q1(Task execution data) and Q3 (rejections data set) due to tasks being flagged for multiple rejection reasons. We streamlined this by introducing a unified rejection code “101” for all tasks with multiple rejection triggers, simplifying analyses and enhancing data integrity, facilitating the merging purposes.

Also inconsistent dates were verified, we find out 3 lines with inconsistencies regarding dates periods, which do not match. As these cases have a minimal impact in the data, corresponding to a very low percentage of instances (0.01% of rows), we decided to drop the processes related to it.

New Features Derivation

Afterward, we proceeded with the derivation of new features, which enhanced our model's ability to uncover more predictive insights. These variables help capture essential patterns and relationships, thereby improving prediction accuracy. The following table summarizes the new variables created:



Feature Name	Explanation
Period between arrival and execution	Period between the hours of task arrival and task execution (Hours amount at the row level)
Period between arrival and capture	Period between the hours of task arrival and task capture (Hours amount at a row level)
Length of the Process	Count of total steps registered for a given process (Request Identifier), allowing to better compare the process complexity
Cumulative Length of Process	Calculates how many steps (activities) have been done until a certain activity in a given process, meaning it adds 1 as we go through the activities
Years in Org Position	Calculated based on the subtraction between the current date and the date since the user belongs to a certain position in the organization (row level)
Rejections	Taking into account the rejections dataset this variable takes value 1 when within a process there is at leats one activity and takes value 0 when there are no rejections (Binary Variable)
Overdue by Expected Date in Days	It selects the amount of days in which was registered an overdue duration of the execution date over the predicted date (row level)
Age	Represents the count of years between the current date and the actual Birth Year of the User (row level)
Number of Females	Total number of female employees in a process
Number of Managers	Total number of managers in a process
Number of Oursourcers	Total number of outsourcers in a process
Number of Rejections	Total number of rejections in a process
Value Count	Total number of inputted values in the Value column

Table 1 - *New Features Explanation*

Additionally, as part of our feature engineering efforts, other new features were derived through process mining encoding techniques. Specifically, we transformed features with numerous categories, such as **'Value'** and **'idField'**, into a numerical format. Our method was to count the number of times a specific category happened in a process. This approach significantly streamlined our data, making it more amenable to analysis and model training.

As a result, some **features** such as, “Task Capture Date”, “Task predicted end date”, among others were **dropped** since they were used in the creation of other variables and were redundant and needed to be excluded from our dataset. Later on, the “Length of the Process” feature was also dropped because it would give us information about the future in the early stages of the process which implies data leakage. Nonetheless, it was helpful during our exploratory data analysis.

Target Definition and Insights

The target variables were defined in alignment with the business process model you provided. Specifically, they are based on strategic combinations of the “Activity ID” and “Action” fields within the dataset. This way we ensure that the targets for the predictive models account for both the nature of the activity and how it was executed or terminated. We concentrated exclusively on activities "101", "107", "104", "106", "108", and "103" as these were the activities recorded at the final stages of the processes. Our main combinations were the following:

- Activity ID “101”: Actions such as “Task automatically terminated – SLA time reached” or “Unknown” are associated with the "Request Cancelled" outcome, which is designated as target 0.
- Activity ID “107”: When the action is “Request accepted by requester,” the target corresponds to the "Request Finished" outcome, assigned to target 1. If the action is “Task terminated - administrative closure” or “Task automatically terminated – SLA time reached,” the outcome is "Request Closed Administratively," corresponding to target 3.
- The action “Task terminated - administrative closure” represents the outcome “Request Closed Administratively since the requester rejected the accounting impact”, which is assigned to target 2.
- Activities “106”, “108”, and “103”: All these are mapped to target 3, indicating outcomes where requests are closed administratively.

Focusing on the analysis of the target distribution, from the figure below it is possible to see that the dataset shows a class imbalance in target categories, which may lead to biased predictions favouring more common outcomes and affecting model scores. Class 2 and 3 are the most present in the data while there are very few requests with the target 0 outcome.

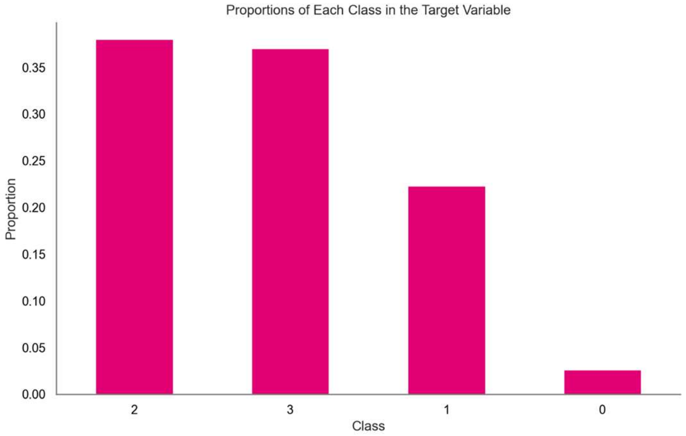


Figure 4 - *Proportions of Each Class in the Target Variable*

Long Processes

Furthermore, we conducted a thorough separate analysis to determine whether **processes with more than 10 steps** exhibited behaviors different from those with fewer steps. Although a slight shift in target class imbalances was observed, these processes generally showed patterns and relationships with our features similar to the less complex processes. Consequently, for computational efficiency, we decided to **exclude these longer processes from our analysis**, as they represented outliers in our dataset. This decision may also reflect the limited data available on these extended processes.

Final investigations

Additionally, our **investigations** did not uncover any significant distinctions between looped and non-looped processes. For more detailed insights into these findings, refer to our Jupyter Notebook on exploratory data analysis (EDA).

Lastly, a **final descriptive analysis** of our dataset can be found in the same notebook.

Prefix Extraction and Last Activity Pooling

To refine predictive modeling techniques for these processes, we have implemented a robust method of prefix extraction. This technique segments complete process flows into smaller, manageable parts called prefixes, each representing a sequential stage in our dataset. Since we are handling processes up to 10 steps, we will identify 10 prefixes, and build 10 dataframes. We systematically compiled process steps into their respective subsets based on prefix length. This accumulation was carefully managed to ensure that only relevant data up to the current prefix was included, allowing us to analyze the impact of initial process steps on subsequent outcomes. Then, we filtered out the process sequences that did not end with designated critical activities (Activity ID 102, 105 and 100) which ensures focus on processes relevant to critical decision points.

Afterward, each process sequence was transformed into a wide format. This involved separating the dataset into steps and reformatting it such that each process step became a distinct feature, suitable for machine learning models. The final step of each process was separately managed to preserve the integrity of the outcome data.

Lastly, after processing and transforming the data, we compiled all processed subsets (excluding the last subset) into a list which was then exported to CSV files for loading them in a modelling environment and develop our predictions. Since our target row (the last step) was separated from the overall dataset we ended up with 9 data frames and notebooks, respectively.

Train/Validation Split

The Split step was made in particular **for each subset**. At this stage we evaluate which was the proper percentage to attribute to the validation data set, considering the number of instances that are present in each of the subsets. We may highlight the **'stratify parameter'** definition, used to ensure that the target classes maintain the same distribution for the train and validation dataset.

Encoding

For **categorical features** we applied an encoding technique, the **target encoding**, since most models only accept numeric format features. Unlike one-hot encoding this technique does not lead to high dimensional sparse matrices (with hundreds of columns), retaining only information regarding the relationship between categorical data and the target variable. Thus, we considered this technique to be more adequate.

Normalization

Then, to normalize all features **MinMaxScaler** was used, that limits the variables values in [0,1] interval, thus the calculations of distance points for modelling techniques will not be influenced by the different scales in our data.

Resampling - SMOTETomek

Since we have an imbalanced multi-class problem, meaning that the target variable classes have an unbalanced distribution (class 0 has approximately 3.9% of representation, class 1 - 29%, class 2 – 21.4% and class 3 – 45.7%), we needed to address this so the model could have good results for each of the classes. For this reason, we tried **SMOTETomek** that oversamples the minority class using SMOTE and then undersamples both the minority and majority classes using Tomek links.

Approach for Data Leakage

To prevent any influence of data leakage (overly optimistic and biased performance regarding training and validation phases) in our analysis, we make sure for all of our subset notebooks that no information regarding the future will contribute to our predictions. For example, we exclude information regarding the period of the processes, and the number of total steps that it takes. For the same purpose, we also ensure that no target information would be available in each of the dataset's lines, meaning that all lines that identify the targets were excluded, as they will be the future outcome of the processes, not in a line but in a column instead.

Feature Selection

The feature selection phase was **specific to each of the subsets tested**, a deep analysis of each of the subsets was made in order to understand which methods thresholds should be selected in order to maintain a proper number of features, that don't lead to overoptimistic predictions of the process's outcomes.

Firstly, we check the **variances** of each feature, as features with no variability ('Constant features') would not be relevant for our predictions and need to be dropped. The **correlation matrix** was also taken into consideration in order to understand which of the features are highly correlated between them (redundant features) and which of them are low correlated with the target variable, therefore would not have a considerable impact in the predictions. Other methods were also considered. **RFE** (Recursive feature elimination method, check *Reference 1* for a detailed description of the approach) was implemented by checking a graph that represents the relationship between number of features and the Train/Validation F1 scores, we wanted to keep the lowest number of variables with the highest F1-scores and lowest score differences, ensuring a good performance of the model in new data. **Lasso method** based on coefficient importance evaluation was also implemented (Check *Reference 2*) and the last method implemented was a **Decision tree classifier** (*Reference 3*). For the **final feature selection** we developed a **Voting System**, where variables that were considered not important by 2 or more methods were removed. To have a better understanding of the techniques used you can also check our notebooks with detailed explanations for each of them. The table below represents the final results for the features retained in all the notebooks processed.

Notebook	Nr. of Features Retained	Features' Retained
1	14	['Value Count', 'idField_3391', 'idField_3491', 'idField_3935', 'idField_47', 'Value_50', 'Value_20', 'Value_100', 'Value_1', 'Value_80', 'Value_90', 'Month_act1', 'Hour_act1', 'Day_act1']
2	18	['Task Executer_act1', 'Task executer department_act1', 'Role ID_act1', 'Years in Org Position_act1', 'Age_act1', 'Value Count', 'idField_3391', 'idField_3491', 'idField_3935', 'idField_47', 'Value_10', 'Value_60', 'Value_2', 'Value_80', 'Actvity ID_last_act2', 'Month_act1', 'Hour_act1', 'Day_act1']
3	6	['Period between arrival and execution_act2', 'Value Count', 'idField_3491', 'idField_3420', 'idField_3935', 'idField_47']
4	5	['NumberOfFemales', 'idField_3491', 'idField_3935', 'idField_47', 'Value_50']
5	5	['idBPMApplicationAction_act4', 'idField_3491', 'idField_47', 'Value_50', 'Value_1']
6	6	['Actvity ID_act5', 'idBPMApplicationAction_act3', 'Value Count', 'idField_3491', 'idField_47', 'Value_50']
7	5	['idBPMRequirement_act5', 'Value Count', 'idField_3491', 'idField_47', 'Value_60']
8	5	['idBPMApplicationAction_act5', 'idBPMRequirement_act4', 'idField_47', 'Value_70', 'Value_60']

Table 2 - Features Retained among all notebooks

3.3. Modeling

For the modeling stage, we developed one model for each prefix extraction (process length). For each model, we tried two techniques to overcome **class imbalance**: SmoteTomek and using the class_weight='balanced' parameter in some models. This parameter gives more weight to the minority class (with fewer examples) and less weight to the majority class, aiming to balance the impact of each class on the model's learning process.

We applied a similar logical approach to each of these models.

First, we developed a **Baseline Model Approach**, where we tried two simple and untuned models (Logistic Regression and Gaussian Naive Bayes) to get initial results. This helped us determine if we needed to revisit the data preparation phase.

Then, we moved to the **Model Tuning Phase**, where we aimed to optimize our model's performance by finding the best combination of parameters. For this purpose, we used Random Search. This technique randomly selects configurations to try out and evaluates the model's performance with those parameters, using the F1-score as our chosen metric. Random Search is more efficient and less computationally expensive than exhaustive techniques like GridSearch, which evaluate every possible combination of parameters.

The models chosen were a **diverse set of models** for their complementary strengths: Logistic Regression for simplicity and interpretability, SVC for handling high-dimensional data, Random Forest for accuracy and feature importance insights, Gradient Boosting for high predictive power, LightGBM for speed and efficiency, Ensemble Model for combining strengths of multiple models, and XGBoost for superior performance and flexibility. A more detailed explanation of these models is written in the notebooks provided.

After the models' implementation we were able to look at their performance both on the training and on the validation set and finally, we chose our **final model** based on two criteria, described in the section below.

Lastly, for **model interpretation**, we used specific techniques for each final model chosen. You can read more about this topic in point 3.5.

3.4. Evaluation

The best model for each subset was selected based on the weighted F1 Score. First, we focused on minimizing overfitting, choosing the model with the most similar F1 scores between the training and validation sets to ensure good generalization on unseen data. Second, we prioritized the model with the highest F1 score on the validation set.

We chose this metric because it balances precision and recall, assessing the model's ability to capture positive cases (recall) and accurately identify them (precision). Its sensitivity to data distribution makes it ideal for evaluating performance in scenarios with uneven target class representation, like ours. It is also important to notice that this metric has values between 0 and 1, being 1 the best value possible, meaning that all predictions are correct, and 0 the worst possible.

Based on our analysis of the results in the table, we can draw several important conclusions. Most of our models exhibit very low levels of overfitting, indicating that they are likely to perform well on new data, maintaining their predictive power and accuracy in future applications. For each subset of data, the selected models and retained features differed, suggesting that the critical factors influencing outcomes vary across process steps. This variability suggests that using diverse, tailored models for each specific process step was effective and appropriate.

Notebook	Best Model	Train F1 Score	Validation F1 Score	F1 Score Difference	Used SMOTE
1	Gradient Boost.	0.692	0.675	0.017	No
2	Logistic Reg.	0.748	0.728	0.020	No
3	SVC	0.788	0.760	0.028	Yes
4	Logistic Reg.	0.654	0.647	0.007	No
5	Gradient Boost.	0.685	0.656	0.030	No
6	LGBM	0.745	0.744	0.002	No
7	XGBOOST	0.789	0.720	0.069	Yes
8	XGBOOST	0.712	0.701	0.011	No

Table 3 - F1 Scores Results among all notebooks

3.5. Model Interpretation

Lastly, it was time to understand our models' predictions, so for each one of the final models that has been chosen, we interpreted it. We used specific techniques for each final model chosen. For Logistic Regression, we examined its coefficients, while for models like XGBoost and LGBM, we analyzed feature importance to understand which features influenced the predictions the most. Interpreting the models will facilitate you in decision making.

We will present, as an example, the Model 4 Interpretation, a Logistic Regression one. As you can see in the graphs bellow:

- For **"Request Cancelled"** outcome (Class 0), the variables idField_3491 and idField_3935 are strongly negative so they will decrease the likelihood of the outcome being this class. In contrary, the variable idField_47 increases the likelihood of Class 0, though with smaller coefficients compared to the negative ones.

- For **“Request Finished”** outcome (Class 1), the variable id_Field_47 is the one that decreases the most the likelihood of this class and Value_50 increases its likelihood, however with a much smaller coefficient.
- Then, for **“Request Closed Administratively since the Requester Rejected Accounting Impact”** (Class 2), idField_47 and idField_3419 are the ones that increases the most its likelihood and Value_50 is the variable that decreases the most.
- Finally, for **“Request Closed Administratively”** (Class 3), idField_47 and Value_50 have the highest coefficients, increasing the likelihood of this outcome.

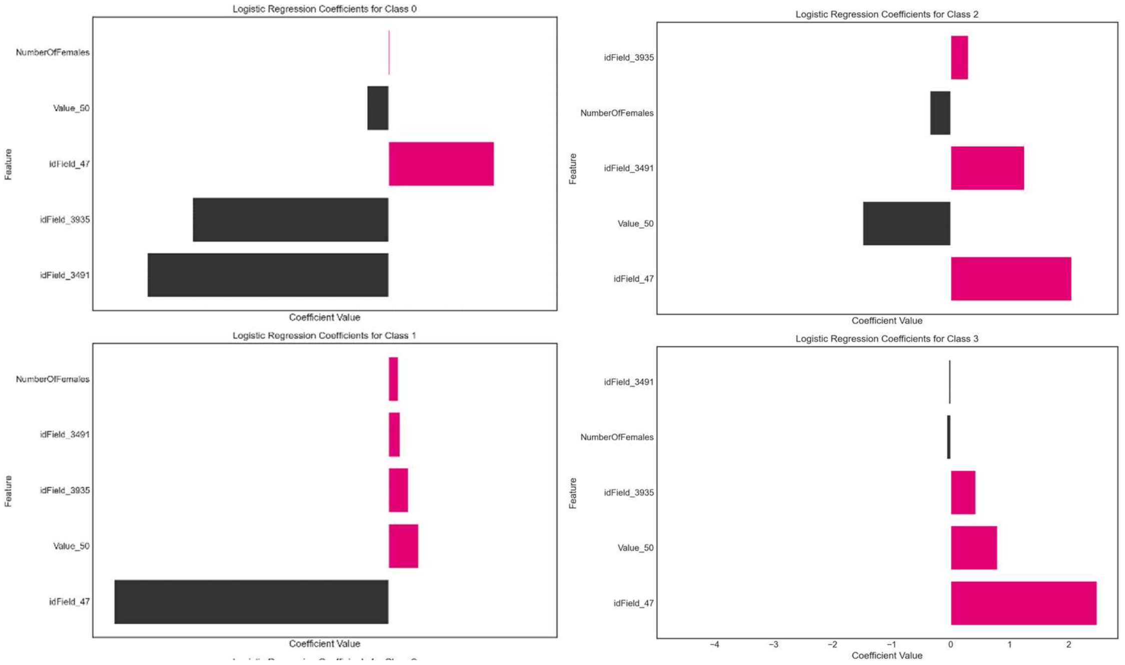


Figure 5 - Notebook 4 Model Interpretation

You can check each Model Interpretation in the respective notebooks provided, for a more detailed explanation.

4. RESULTS EVALUATION

The development of a project of such type is expected to trigger the early defined success criteria. For each one of them we can translate how a good performance enhances them:

- **Client Retention Rate Increase:** With strong initial feedback at the moments 100, 102 and 105 clients will feel more secure and therefore be more satisfied with Millenium Bank services, continuing their engagement with your services. This translates in a higher retention rate.
- **Profit and Loss Account Recovery:** A good predictive model’s implementation will improve the health of a companies’ financial performance, decreasing the bundle of expense assigned to these events. This improves flexibility and promotes a larger budget for investments.
- **Automation Rate Increase**: With proper predictions, workflows can be planned with the right timing and resources allocated in an efficient way, which overall will be reflected in the automation rate achieved.

The extent to which these metrics are reaching the desired goal and level will need to be closely discussed with you, as we believe after the implementation of the steps in the following sections, it will be easier to come up with realistic thresholds. So, we are more than willing to hear from your side and continue with the improvement of the project to reflect the threshold values if not yet met.

5. DEPLOYMENT AND MAINTENANCE PLANS

For the **deployment** of this project, the following is proposed:

Next Steps: Further model development and validation to comply with banking regulatory standards which could include stress testing, risk management compliance, and adherence to policies set by banking authorities. This involves rigorous processes to prove that the model behaves as expected across a range of scenarios, particularly those that might cause financial risk. This also highlights the need to gather more data about more lengthy processes.

Automated Deployment: Use automated deployment processes where possible by setting up continuous integration and continuous delivery (CI/CD) pipelines, which allow for seamless transitions from development to production environment.

Integration with Existing Systems: Integrate the model with existing banking IT infrastructure. This will require adapting the model to work within the bank's transaction processing systems, risk management frameworks, or customer relationship management (CRM) systems.

For the **maintenance** of our model, the following steps are below:

Continuous Monitoring: Constantly evaluate the model's performance and accuracy by tracking feature drift, data drift, and overall model health to ensure predictions remain reliable over time.

Regular Updates and Retraining: Set up processes for periodic retraining of the model to adapt to new data patterns and emerging trends. This retraining should be triggered by significant events or detected anomalies that could impact model performance. This process should be as automized as possible to enhance efficiency and responsiveness.

Feedback Loop: by gathering iterative feedback we can gather insights from the model's performance in the operational environment and can help fine-tune and optimize continuously the model. These mechanisms are comprised for instance in collecting input directly from users, automated anomaly detection systems that flag discrepancies in predictions, or through regular audits comparing model predictions with actual outcomes.

6. CONCLUSIONS

The project developed by our team on predicting business process conclusions for you, Millenium, has reached promising results, demonstrating the potential for significant improvements in operational efficiency and customer satisfaction. Through the application of the CRISP-DM methodology, our team successfully developed a predictive model that balances precision and recall, making accurate predictions across various business process outcomes.

The chosen models demonstrated a good performance in both training and validation phases, indicating small overfitting and robust predictive capabilities. By predicting process outcomes, the model enables better resource allocation, streamlined workflows, and improved customer experiences.

It is essential to maintain and improve the model, future efforts should focus on integrating the model with existing banking systems, automating deployment processes, and adhering to regulatory standards.

In conclusion, this project has laid a solid foundation for leveraging predictive analytics in Millenium Bank's business processes. By addressing the identified challenges and continuously refining the model, the bank can achieve sustained growth, enhanced operational efficiency, and superior customer satisfaction.

6.1. CONSIDERATIONS FOR MODEL IMPROVEMENT

The following topics were noticed during the development of the project. We believe that their consideration for future approaches would be a good asset to increase the model’s performance.

The **missingness of data explanations** can be considered as a key point. We believe that it would be easier to identify some patterns in the data if we had a background history. These patterns could have been used to create other relevant variables for modeling phases, which can lead to better predictive results. 'idBPMRequirement' is a variable that corresponds to some sort of rejection motive. However, it is only associated with a code, we do not have any sort of information regarding its description. In another case, with this information available it would be possible, for example, to join some codes as they could be related in some way, making it easier to identify patterns.

The **lack of computational power** was a considerable obstacle when implementing some algorithms. The amount of data leads to higher times of processing in the modeling phase. With more computational power, we believe that we would have time to check other model approaches or even do more experiments regarding other types of resampling methods that could lead to better results.

A **future model refinement** is also a key point**,** as after interpreting our chosen models, we now know which variables contribute the most to accurately predicting the outcome, and some less critical than can potentially be removed.

Lastly, **extra data** should also be considered for **long processes** (with more than 10 task ids) as it would be helpful when designing a more robust modeling phase for these cases in specific.

7. REFERENCES

1.sklearn.feature_selection.RFE — scikit-learn 0.23.1 documentation. (n.d.). Scikit-Learn.org. https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html

2.sklearn.linear_model.Lasso. (n.d.). Scikit-Learn. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html

3.Feature selection using Decision Tree. (2024, March 11). GeeksforGeeks. https://www.geeksforgeeks.org/feature-selection-using-decision-tree/

4.4Service: Mystery shopping provider, Market research agency Description. (n.d.). 4service.company. https://4service.company/en/blog/the-critical-role-of-customer-satisfaction-in-the-banking-sector

Business Process Management in Banking and Financial Industry (2024, April 9) https://www.cflowapps.com/business-process-management-banking-financial-industry/

Bank of England. (2022, October 11). *Machine learning in UK financial services*. Www.bankofengland.co.uk. https://www.bankofengland.co.uk/report/2022/machine-learning-in-uk-financial-services

Machine Learning Operations (MLOps) in Banking. (n.d.). Deloitte United States. https://www2.deloitte.com/us/en/blog/deloitte-on-cloud-blog/2023/machine-learning-operations-in-banking.html

Lee, J. (2024, February 20). *Improving Your Credit Risk Machine Learning Model Deployment*. Experian Insights. https://www.experian.com/blogs/insights/machine-learning-model-deployment/

CRISP-DM methodology overview: Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). "CRISP-DM 1.0: Step-by-step data mining guide."

Evaluation metrics in machine learning: "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron.

Multiclass classification performance metrics: "Pattern Recognition and Machine Learning" by Christopher Bishop.

Real-world data considerations: "Real-World Machine Learning" by Henrik Brink, Joseph Richards, and Mark Fetherolf.