REPORT GROUP 3

Intensive Programme 2024 – UBS-Case Study.

Group members:
Mateusz Knapik, Eva Rakipaj, Simone Andreolli, Rufaro P Nziramasanga
Saulescu Silviu Bogdan

March 2023

Katowice, Poland

1. Introduction.

The task is to calculate Risk-Weighted Assets using the Internal Model approach for three counterparties (Salzburg Bank, Bank of Cluj, and Bank of Mazowsze). The calculation of the Risk-Weighted Assets for the Bank requires the calculation of two important variables first: the Probability of Default (PD) for each of the counterparties, and the calculation of the Exposure At Default (EAD).

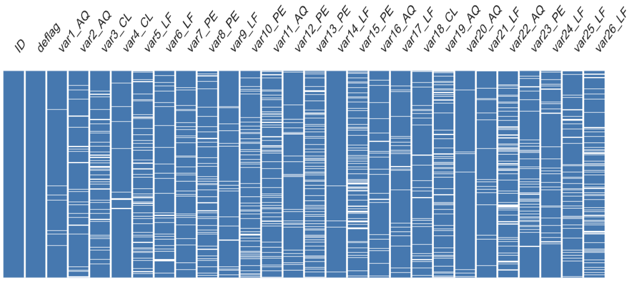
To proceed with the analysis of the exercise we used the programming language Python, with the application of many various libraries, and MS Excel.

2. Probability of Default (PD)

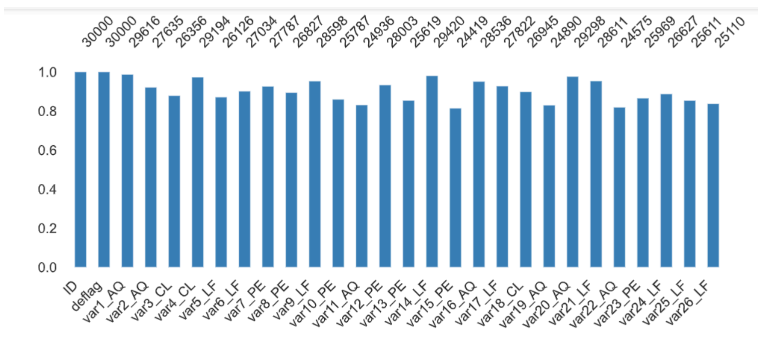
2.1 Initial dataset exploration and description

The data set we need to analyze is about 30 000 banks and for each of them are given 27 variables. Timespan is said not to be relevant by any means. The first Variable is the default flag. It is not possible to assess whether it has missing values as this variable is binary. The other 26 variables are financial ratios describing the performance of the banks. Particularly, we have 4 kinds of variables describing 4 areas of performance of the bank: 7 variables describe the *Asset Quality*, 3 variables describe the *Capital and Leverage*, 7 variables describe the *Profitability and Earnings* and the last 9 variables describe the *Liquidity and Funding.* For further data exploration, we used interactive ProfileReport from the library ydata_profiling. The major and general insights are: 78649 missing values which are 9.4% of total data, var_15PE has the most missing values (18.6%), every variable apart from 1st and 3rd are highly skewed, variables 3 and 18 are already at this stage highly correlated. In the case of the deflag variable, there are 524 values of 1 which means that default occurred (0.01746% of the total bank).

However, the data set has plenty of missing data and outliers. So, in the next section, we explain how we dealt with them. In the graphs below, it can be seen that the spread of missing values seems to be rather without a specific pattern, with the exception of a few variables that have the least missing values. Those are for example var1_AQ with 1.3% missing and



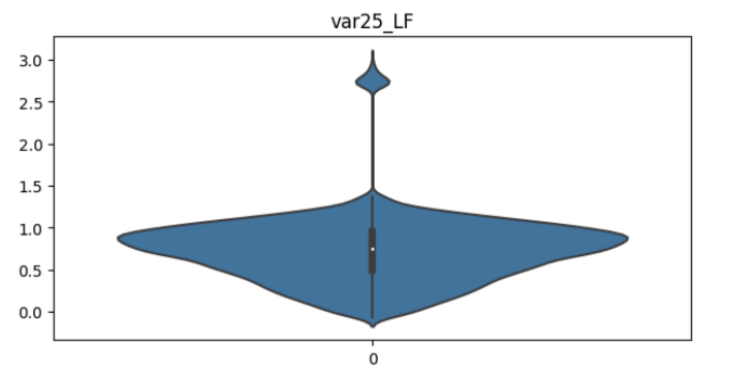
Nullity matrix is a data-dense display which lets you quickly visually pick out patterns in data completion.



var14_LF with 1.9% missing.

To better understand the issue of outliers in our dataset, we decided to first, visually assess the variables by plotting the boxplots of those variables, and also the violin plots to see the distribution too. For example, see the violin plot for the 25th variable, the upper side of distribution, detached from the main distribution, represents the outliers.

Also, we wanted to better understand the descriptive statistics and see how many outliers there are; so the table below is the sample table with the results of our



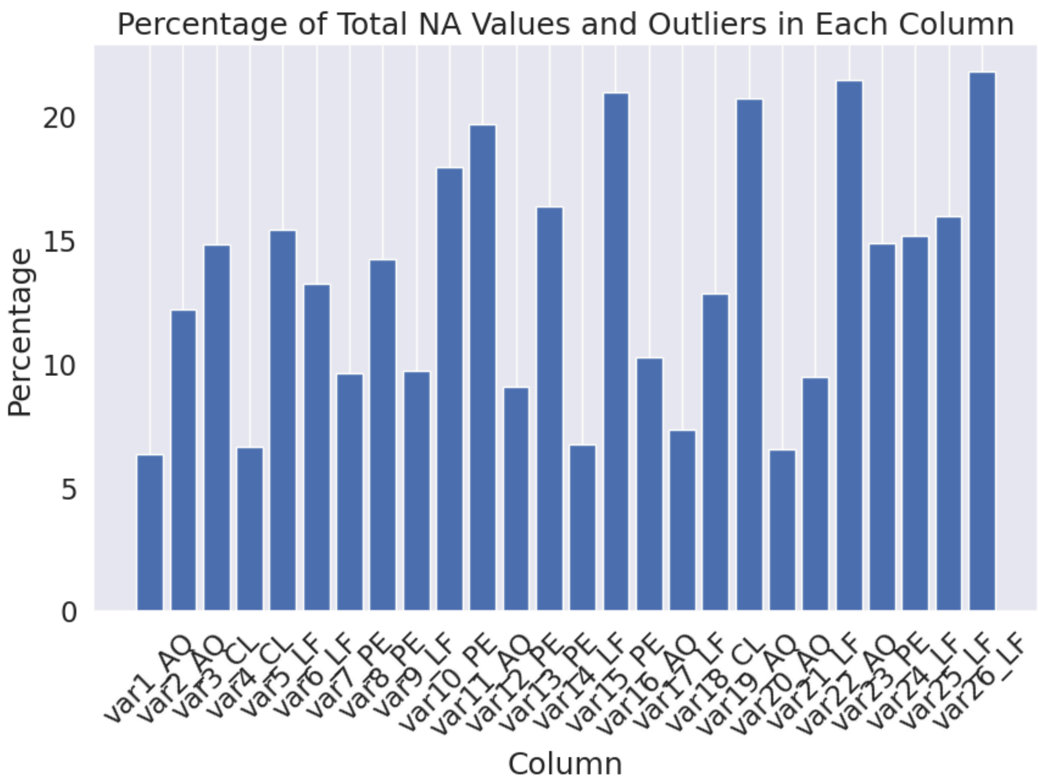
calculations. The variable with the lowest number of outliers is variable 17, and the one with the highest numbers is variable 26.

	Ilość	Średnia	Odchylenie	Minimum	Maksimum	I kwartyl	Mediana	III kwartyl	Outliers rozstęp	Outliers Z-score	% outliersów	% outliersów Z-score
var1_AQ	29616	5.501546	17.018673	-1.519354	99.097655	0.509675	1.656260	3.173067	1521	1347	5.135737	4.548217
var2_AQ	27635	0.321956	1.727811	-0.210979	99.000000	0.065042	0.211095	0.404787	1289	8	4.664375	0.028949
var3_CL	26356	21.644979	36.839109	0.574025	331.054821	10.794106	15.720088	21.609048	799	566	3.031568	2.147519
var4_CL	29194	0.111201	0.603695	-0.007078	99.000000	0.046135	0.073988	0.108512	1188	1	4.069329	0.003425
var5_LF	26126	1.140406	2.406852	0.200855	99.000000	0.630163	0.833686	1.071925	755	628	2.889842	2.403736
var6_LF	27034	0.320408	2.021522	-0.115131	99.000000	0.092208	0.211677	0.364590	1000	11	3.699046	0.040690
var7_PE	27787	1.028982	0.833596	0.966805	99.000000	0.999822	1.013135	1.026942	676	2	2.432792	0.007198

2.2 Missing Values and Outliers.

In our analysis, we identified the need to accurately identify and manage outliers to understand their impact on our dataset. We chose to use the winsorizing method, which limits extreme values using the Interquartile Range Rule (IQR). For outlier identification, we relied on the IQR, setting our bounds as follows:

```
lower_fence = q1 - 1.5 * iqr
upper_fence = q3 + 1.5 * iqr
```

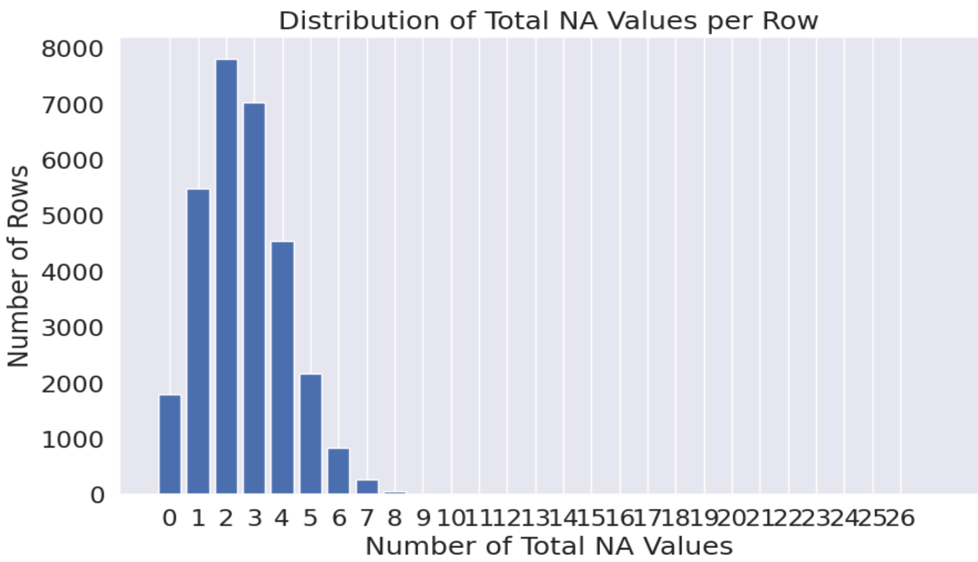


where:

- The first quartile (Q1) represents the threshold below which 25% of the values fall.
- The third quartile (Q3) indicates the threshold below which 75% of the values are found.
- The Interquartile Range (IQR) is calculated as Q3 minus Q1.

Employing this strategy reveals a significant presence of outliers in our dataset, as illustrated in the graph provided above.

Furthermore, it became imperative to assess the extent of anomalous data present for each bank, encompassing both missing (NA) values and outliers. The accompanying graph delineates the distribution of missing values per row. In our data cleansing process, we opted to omit rows exhibiting more than 5 missing values, specifically targeting row 6, 7, and 8. This criterion led to the removal of 1167 rows, corresponding to individual banks. During this filtration, we paid crucial attention to not excluding rows (banks) flagged as default as the proportion of default flags in the whole dataset is quite small, and also to not exclude our banks.



At this point, we need to decide how to replace missing values and outliers. In the beginning, we decided to adopt a more statistical approach: we replaced all the NA values with the median (we chose the median because our data contains many outliers, and the median is less influenced by them compared to other statistics, such as the mean) and all the outliers with the *upper fence*. However, this method did not yield satisfactory results. Consequently, we explored various strategies for handling these anomalies, conducting numerous tests to find a more effective approach. The outcomes of these investigations are detailed in the later parts of the report.

To address the challenge of missing data with greater precision, we delved further into the analysis of variables. Our strategy for managing missing values is as follows: when a bank exhibits a missing (NA) value for a specific variable, we first assess the presence of other missing values within the same category (AQ, CL, LF, PE). The discovery of additional NAs prompts us to infer a possible intent to conceal information pertaining to that category. Consequently, we impute the missing value based on a "worst-case" scenario—substituting lower_fence values for variables that are optimally high, and upper_fence values for those that should ideally be low. Conversely, when the remaining variables in the category are accounted for, indicating that the missing data may be incidental, we opt to replace the missing value with the median of that category. This approach ensures a nuanced handling of missing data, reflective of the underlying patterns and potential implications within the dataset.

For the outliers, we decide to select a subset of variables that we consider more important from an economic point of view:

`['var1_AQ', 'var2_AQ', 'var3_CL', 'var6_LF', 'var9_LF', 'var11_AQ', 'var18_CL', 'var23_PE']`

For these variables we proceed in the following way: if the outlier is above the *upper fence* we replace it with the *upper fence* itself, if the outlier is under the *lower fence* we replace it with the *lower fence.* For variables not included in this select group, we choose to impute outliers with the median of their respective distributions. This decision stems from the recognition that the origins of certain extreme outliers (e.g., a value of 99 in a dataset where the mean is 0.32) are unclear and may result from data entry errors or other inaccuracies. By substituting such outliers with the median, we aim to mitigate the potential impact of these anomalies on our analysis. This approach allows us to strike a balance between not unduly penalizing the data points that might be erroneous, yet still applying a cautious correction for variables of critical importance, where the risk of obscuring significant deviations due to potential human error must be carefully managed.

To assess the effectiveness and impact of our data cleaning procedures, we deemed it essential to perform a comparative analysis of the main descriptive statistics for each variable, both before and after implementing our outlier management and missing data imputation strategies. In the table below, the results of the descriptive statistics after the data manipulation are shown. From the maximum values we can observe that outliers are no longer in the dataset. Also, the mean values differ now from the initial dataset. Some variables are more influenced by our changes. For example the mean value of variable 1 got changed more (2.04 compared to 5.50 before cleaning) than mean values of variable 4 (0.07 compared to 0.11).

	count	mean	std	min	25%	50%	75%	max
ID	28831.0	15011.898547	8663.024008	3.000000	7508.500000	15028.000000	22514.500000	30000.000000
deflag	28831.0	0.017446	0.130930	0.000000	0.000000	0.000000	0.000000	1.000000
var1_AQ	28831.0	2.043058	2.120390	-3.481949	0.515257	1.654489	3.145873	7.160527
var2_AQ	28831.0	0.265329	0.269198	-0.210979	0.077395	0.210587	0.396341	0.912818
var3_CL	28831.0	16.444096	8.112389	-5.457401	11.215625	15.722687	20.667720	37.876487
var4_CL	28831.0	0.075837	0.041658	-0.047550	0.046588	0.073996	0.100496	0.202118
var5_LF	28831.0	0.861654	0.306579	0.200855	0.662107	0.834149	1.021056	1.732626
var6_LF	28831.0	0.234349	0.204873	-0.316013	0.098006	0.211695	0.344071	0.772331

2.4 PD Model

For the PD estimation, we initially wanted to see how different models and different approaches to dealing with data will influence the model's performance and outcomes. All in all, it was a very time-consuming process and very often we were receiving completely odd results. Embracing the principle that simplicity often yields the most dependable and effective solutions, we ultimately opted for logistic regression as our model of choice employing the specific strategies for managing missing values and outliers as previously detailed.

In our exploration of various models to estimate the Probability of Default (PD), we conducted tests across multiple predictive models: Logistic Regression, Random Forest, XGBoost, Bagging (Naive Bayesian), and Bagging with Decision Tree. Our initial data preprocessing strategy for these tests involved median imputation for missing values, trimming outliers to set lower and upper bounds, and data standardization using RobustScaler, applied across all variables.

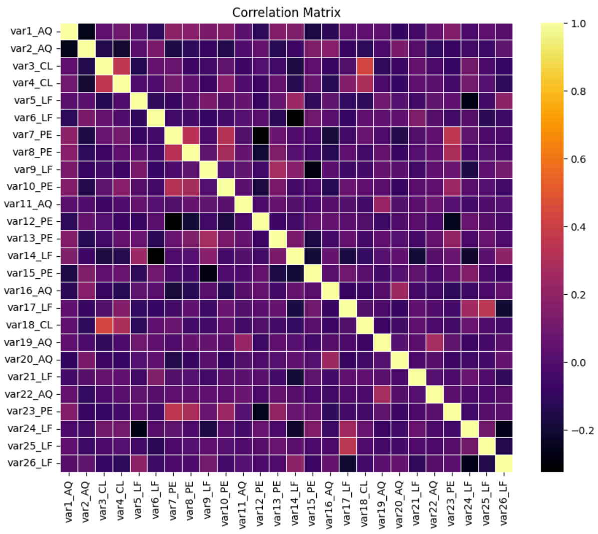
	Acc_train	Acc_test	AUC_train	AUC_test	F1_train	F1_test	false_positive	false_negative	true_positive	recall	precision
Regresja logistyczna	0.982333	0.983333	0.873961	0.873590	0.045045	0.056604	1	99	3	0.029412	0.750000
Lasy losowe	1.000000	0.983167	1.000000	0.868912	1.000000	0.019417	0	101	1	0.009804	1.000000
XGBoost	1.000000	0.983333	1.000000	0.911823	1.000000	0.122807	5	95	7	0.068627	0.583333
Bagging (naiwny bayes)	0.956292	0.956500	0.882042	0.875957	0.255500	0.243478	201	60	42	0.411765	0.172840
Bagging (drzewo decyzyjne)	0.996500	0.983000	0.999846	0.744169	0.889764	0.037736	2	100	2	0.019608	0.500000

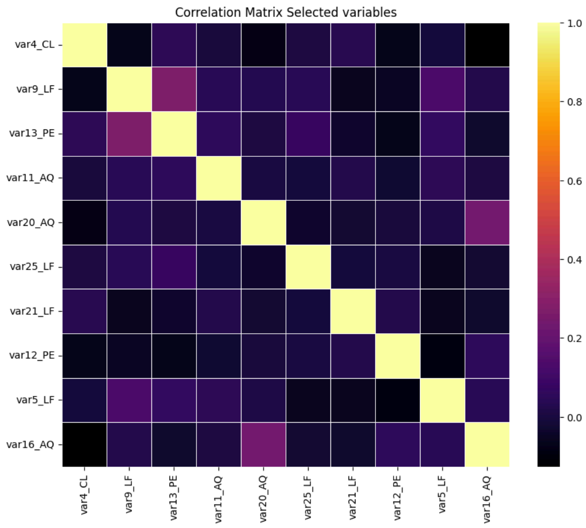
After extensive testing and comparison, we were not satisfied with the results, no matter the approach and the model. That's the reason why we decided to change the model, approach, and manipulation of data. This choice highlights the critical role of careful data preparation in enhancing the performance and reliability of predictive models. We decided to stick with the Logistic Regression model in an adjusted form for our needs. This is due to the fact that Logistic Regression is a widely known and used model for many purposes.

Furthermore, logistic regression is ideally suited for binary classification tasks, such as distinguishing between Default and Non-Default outcomes, and calculating the probability of default in our analysis. This model transforms a linear combination of predictor variables into a probability, ensuring outputs fall within the 0 to 1 range. To effectively deploy logistic regression, selecting appropriate predictors is crucial. Our strategy focuses on choosing 10 variables from a pool of 26, prioritizing those with minimal correlation to maintain their independence.

2.4.1 Choice of Variables

We start analyzing the correlation matrix between all the variables.





In analyzing the correlation matrix among all variables, we observed generally low correlations, guiding our variable selection for logistic regression. We excluded variables 15, 19, 22, and 26 due to their high count of anomalous values, and variables 13, 14, and 24 for their relative lack of importance. Then we select 3 variables: number 9 because we considered it really important, and variables 4 and 13 because they are both important and have a low number of anomalous values. From the remaining variables, we chose the 7 with the lowest correlation with the already selected and with the lowest inter-correlations to ensure independence. We obtain:

```
['var4_CL', 'var9_LF', 'var13_PE', 'var11_AQ', 'var20_AQ', 'var25_LF',
'var21_LF', 'var12_PE', 'var5_LF', 'var16_AQ']
```

2.4.2 Splitting the data

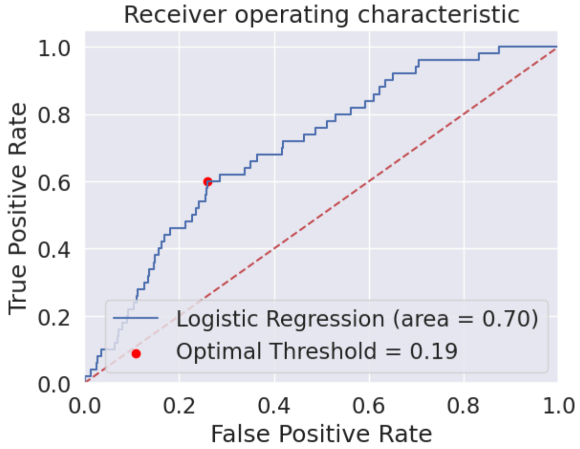
To effectively develop and evaluate our logistic regression model for predicting default, we divided the dataset into two distinct parts: 90% of the data was allocated for model training, while the remaining 10% was reserved for testing the model. This division was executed randomly to ensure a representative mix of data in both the training and test sets. Importantly, we took special care to distribute banks flagged for default (flag set to 1) uniformly between the training and testing subsets. This approach ensures that both phases of model development and evaluation have consistent exposure to default cases, facilitating a more balanced and accurate assessment of the model's predictive performance.

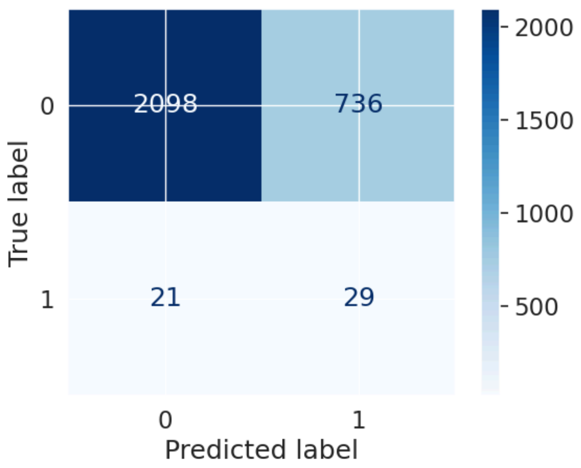
2.4.3 Predictions

In the beginning, we decided to use the logistic regression without any kind of weights on the data. However, given the huge disproportion between the 0 and 1 *default flag* indicator, the model was not reliable even if the results were very good (most of the banks had a probability of default under 0.05). Thus, we decide to balance the data to maximize the f1 score for the default indicator.

2.4.4 ROC curve analysis

We plot the true positive rate (sensitivity) against the false positive rate (1-specificity) for all threshold values: each point on the ROC curve represents a different threshold, and the curve illustrates how the model's performance varies across different threshold choices. Analyzing the shape of the ROC curve and considering the trade-offs between sensitivity and specificity at different threshold values, we choose the optimal threshold. In the end, we compute the confusion matrix for the optimal threshold:





2.4.5 PD for the three banks

Our model generates the following default probability predictions, we have also decided on the Credit Rating scales and assigned the proper rating to each Probability of Default. Please notice that the scale is designed by us and it is subjective.

Probability of Default (PD) and Credit Ratings for Selected Banks

Bank ID	PD (Decimal)	PD (%)	Credit Rating
Bank 484	0.049261867	4.93%	AA
Bank 47	0.265954696	26.60%	CCC
Bank 2741	0.121063634	12.11%	BBB

Table 9: Credit Ratings and Corresponding Probability of Default (PD) Thresholds

Rating	AAA	AA	A	BBB	BB	B	CCC	CC	C
PD	$< 1\%$	$< 5\%$	$< 10\%$	$< 15\%$	$< 20\%$	$< 25\%$	$< 40\%$	$< 60\%$	$> 60\%$

It's important to acknowledge that the predicted probabilities of default appear to be relatively elevated. A more in-depth understanding of economic principles and practical experience with the variables could enhance the model's implementation through improved data cleaning methods and a more strategic selection of variables.

In conclusion, while our model aims to be reliable, it's essential to remember that the probabilities listed are merely statistical estimations and do not definitively predict a counterparty's default. External factors such as market volatility, interest rate fluctuations, the regulatory landscape, and industry-specific dynamics can significantly influence the accuracy and applicability of our results.

3. Exposure at Default.

3.1 Implied Volatility

Our initial aim was to find the implied volatilities for a set of options using the Black-Scholes model for option pricing. The Black-Scholes formula was utilized to calculate theoretical call option prices based on inputs such as the stock price, strike price, time to maturity, risk-free interest rate, and volatility. We then compared these theoretical prices against actual market prices to determine the implied volatilities that best match the observed market data. To achieve this, we first defined a function to calculate the root mean square error (RMSE) between the market prices of options and their corresponding theoretical prices calculated using various volatilities. A lower RMSE indicates a better fit.

Next, we initiated an optimization process, starting with an initial sigma guess. We applied the (BFGS) optimization method, a popular algorithm for solving nonlinear optimization problems, to minimize the squared difference between the market and theoretical prices for each option, thereby estimating the implied volatilities. We obtained a matrix of 16 volatilities, one for each time to maturity and strike. Following, we sought to simplify our model by finding a single, optimal volatility value that best fits the entire set of options. We conducted a thorough search across a range of sigma values, from the minimum to the maximum implied volatility obtained from the BFGS optimization. For each sigma value in this range, we computed the RMSE against the market prices, using a matrix filled with that constant sigma value for all options. The sigma that resulted in the lowest RMSE was deemed the optimal volatility for the entire dataset, representing a generalized estimate that balances accuracy across all options. This approach provides a streamlined, aggregate measure of market volatility, simplifying complex datasets into a single, interpretable metric.

In summary, through iterative optimization and strategic simplification, we successfully identified an optimal sigma that minimizes discrepancies between theoretical and market prices.

Portfolio Type	Lowest Precision	Optimal Sigma
Vanilla Option	0.7964314501723353	0.24256837954109003
Asian Option	4.033286280855074	0.28302180918580316
American and Asian Options	43.95906356050317	0.33498128928364995

Table 1: Implied Volatility and Lowest Precision Results

3.2 Monte Carlo Simulation

The Monte Carlo Simulation (MCS) represents a sophisticated computational technique employed in the valuation of options, leveraging the stochastic modeling of the underlying asset's price trajectory across temporal intervals. By incorporating principles of probability and statistical sampling, the MCS method facilitates the estimation of future price fluctuations of the asset, taking into account market volatility and the time until expiration. In our study, we elected to utilize this approach to address the valuation challenges presented by portfolios of three banks, with each portfolio made by one call and one put option. This decision was motivated by the objective to extend the applicability of the MCS methodology across diverse markets, encompassing European, Asian, and American-type options, thereby demonstrating its adaptability and effectiveness in a range of financial contexts.

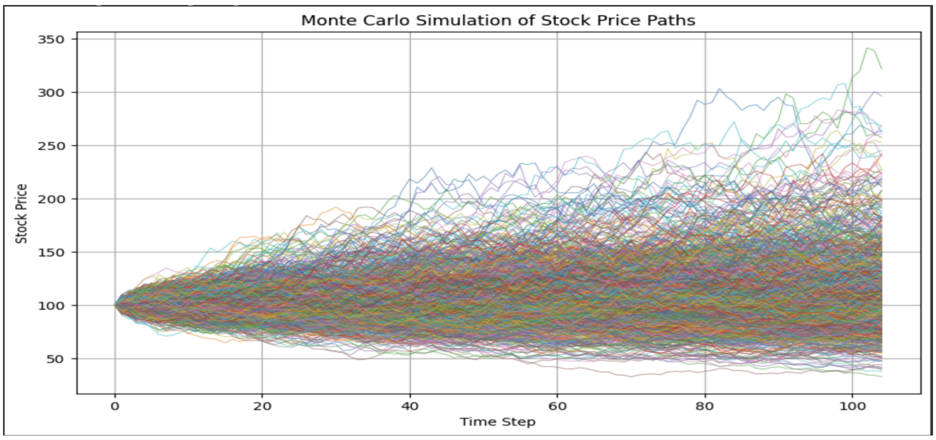
3.2.1 Monte Carlo simulation of the underlying, Salzburg Bank

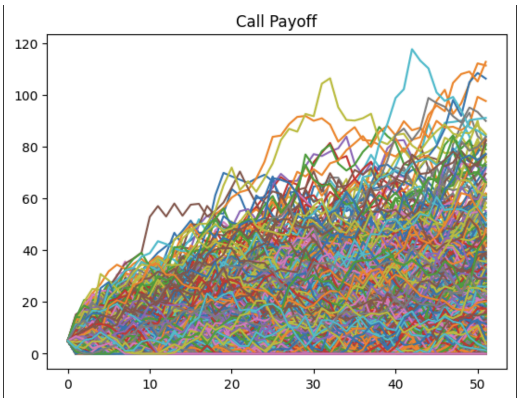
The task at hand reveals that Salzburg Bank's portfolio includes two European Options. Specifically, it holds a Call Option that matures in 1 year, with a strike price set at 95 USD, and is based on the SPDRM underlying asset. Additionally, the portfolio contains a Put Option with a 2-year maturity and a strike price of 115 USD, also tied to the SPDRM underlying.

In this comprehensive analysis, we deployed Monte Carlo simulation techniques alongside the Black-Scholes model to evaluate the pricing dynamics of European options within Salzburg Bank's portfolio. The Monte Carlo simulation method was utilized to project the stochastic paths of the underlying asset's price over a specified period, employing all the necessary parameters This approach facilitated the generation of numerous simulated asset price trajectories, each comprising a sequence of prices reflecting potential future market conditions. To estimate the price of plain vanilla put options, we first calculated the option payoffs at expiration based on these simulated paths. The method involves determining the maximum value between the strike price (K) minus the simulated asset price at maturity, and zero for put options. Similarly, for call options, the same principle applies, with payoffs calculated as the maximum value between the simulated asset price at maturity minus the strike price, and zero. The final option price was obtained by averaging these payoffs and discounting back to the present value, employing the risk-free interest rate over the option's life span.

Concurrently, the Black-Scholes model provided a theoretical benchmark for option pricing. This renowned formula calculates the option's price based on the underlying asset's current price, strike price, time to maturity, risk-free interest rate, and volatility, offering a deterministic perspective on option valuation.

The outcomes of this dual-faceted approach—incorporating both Monte Carlo simulations and the Black-Scholes model—were then meticulously compared. The comparison elucidated the practical and theoretical valuations of options, highlighting the simulated price versus the Black Scholes price. This juxtaposition not only reinforces the reliability of the Monte Carlo simulation in mirroring complex market dynamics but also validates the enduring relevance of the Black-Scholes model as a foundational tool in option pricing theory.
Please for the numerical result consult the graphs and table we provided below:





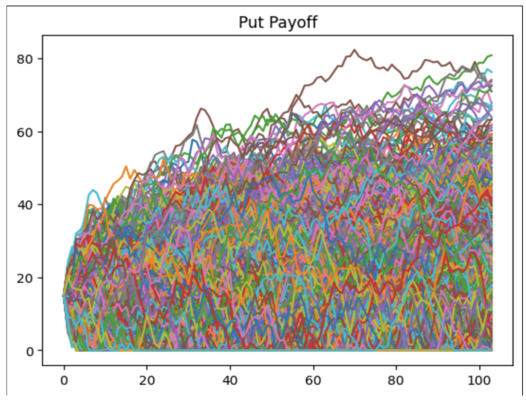


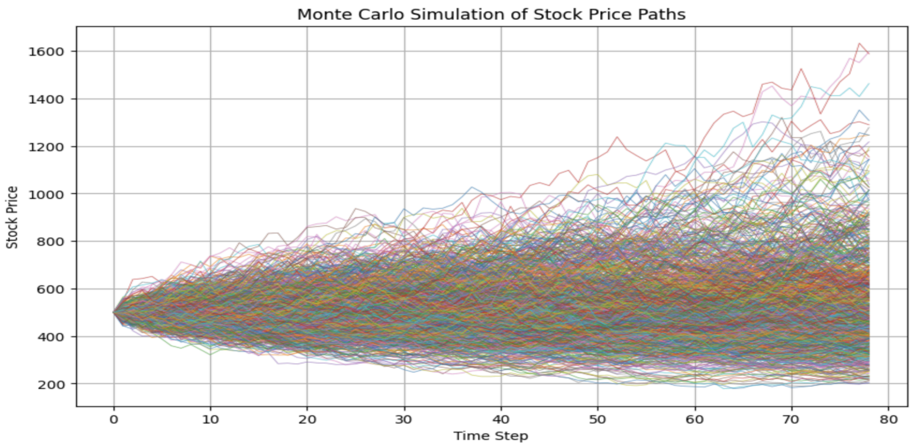
Table 2: Black Scholes vs. Simulated Prices for Options

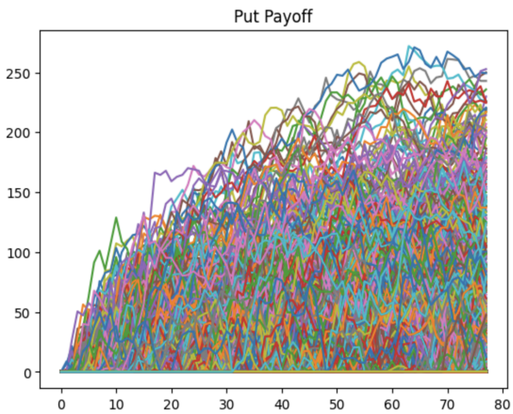
Option Type	Black Scholes Price	Simulated Price
Put Option	16.015313246959607	15.923156239685387
Call Option	14.790031380861798	14.77110482171624

3.2.2 Monte Carlo simulation of the underlying, Bank of Cluj

The second portfolio we had to analyze is a Call Option that matures in 6 months, with a strike price set at 570 USD, and is based on the SPDRM underlying asset. Additionally, the portfolio contains a Put Option with a 1.5-year maturity and a strike price of 450 USD, also tied to the SPDRM underlying.

The initial phase of our methodology involves specifying the foundational parameters of the underlying asset and the option contract. Upon establishing these parameters, the simulation process commences with the generation of random paths for the asset's price evolution. Each path is constructed iteratively, incorporating random shocks derived from a standard normal distribution to model the asset's return at each time step. This stochastic modeling approach is underpinned by the geometric Brownian motion framework, adjusted for the expected return and volatility of the asset.
The distinctive feature of an Asian option necessitates the computation of the mean price across all simulated paths. Depending on the nature of the option (call or put), the payoff is calculated as either the excess of the average price over the strike price (for a call option) or the shortfall of the strike price below the average price (for a put option), with the provision that the payoff cannot be negative. The culmination of this process is the determination of the option's price, achieved by averaging the calculated payoffs across all simulations and discounting the result to present value, taking into account the risk-free rate. This computation yields the estimated market price of the Asian option under the given parameters. For results consult Table 3 and the graphs below:





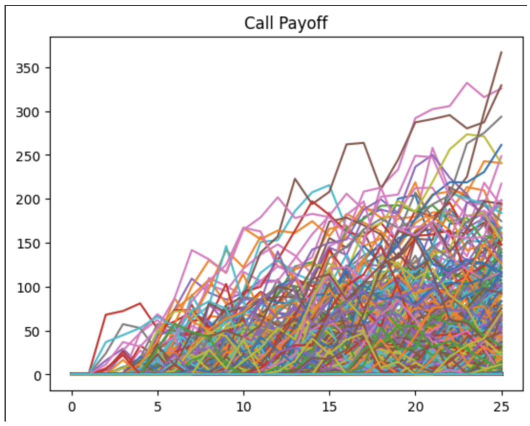


Table 3: Prices of Asian Put and Call Options

Option Type	Price
Asian Put Option	12.646805433037112
Asian Call Option	5.4569477157299175

3.2.3 Monte Carlo simulation of the underlying, Bank of Mazowsze

Instead, our third portfolio holds an American Call Option that matures in 1.5 years, with a strike price set at 4100 USD. Additionally, the portfolio contains an Asian Put Option with a 9-month maturity and a strike price of 4100 USD. For the Asian Option, we followed the same tactic as in our second portfolio, whereas on the American Option the following:

We delve into the pricing of American options through the utilization of the Longstaff-Schwartz Monte Carlo simulation method. We encapsulated the necessary components to price American options through simulation. The class is initialized with the set of needed parameters defining the option and market conditions. Then we continue generating a matrix of simulated prices for the underlying asset. Utilizing a random seed for reproducibility, the method simulates price paths employing the geometric Brownian motion model, reflecting the asset's expected return and stochastic volatility. Furthermore, we calculate the intrinsic value of the option across all simulated paths and time steps, distinguishing between call and put options. Subsequently, we leverage a polynomial regression to determine the option's early exercise boundary. The option's price is ultimately computed as the discounted average of the payoffs, adhering to the risk-neutral valuation principle through the calculation of Greeks.

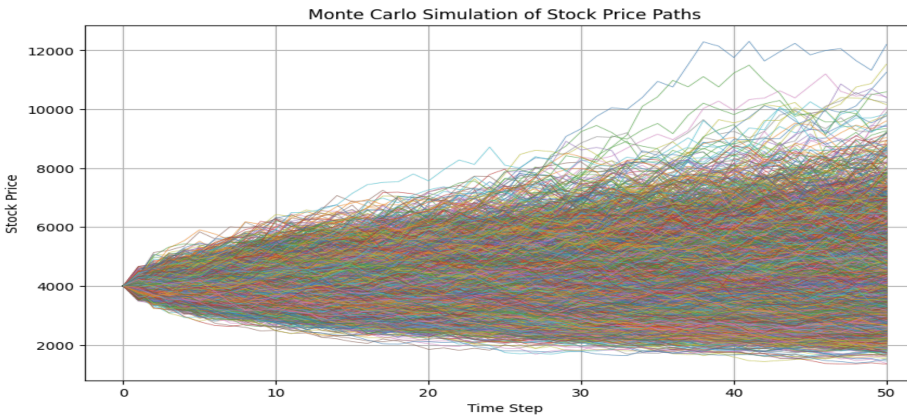


Table 4: Prices of the American and Asian Option

Option Type	Price (USD)
American Call Option	704.5496298551366
Asiatic Put Option (9 months maturity)	281.469

3.3 Methodology of the Exposure Metrics

In this chapter, we deal with the Exposure at Default (EAD). In order to evaluate it we need to find all the Exposure metrics. We start evaluating the MtM matrix by taking into consideration the option at each time point and for each simulated path of the asset. Then we calculated the Expected Exposure (EE) for each time point:(which is determined by isolating the positive MtM values and then computing, for each time point, the expectation through all paths.) We obtained an array and went on computing the cumulative maximum average of the array to obtain the

Effective Expected Exposure (EEE). Finally, we computed the Effective Expected Positive Exposure (EEPE) as the mean of the Effective Expected Exposure (EEE) across all time steps, providing a single metric that encapsulates the option's average positive exposure. After obtaining the (EEPE), we can calculate the Exposure at Default (EAD) multiplying the EEPE with the constant value $\alpha = 1.4$ (provided by financial regulators).
We applied this methodology to find the RWAs for all the options for all three portfolios. To calculate the Risk-Weighted Assets (RWAs) for the three portfolios, we differentiated between netted and non-netted portfolio scenarios.
For the netted portfolios associated with Salzburg and Mazowsze banks, we calculated the mark-to-market (Mtm) matrices for both options, ensuring to appropriately handle the matrices' different dimensions during summation. In contrast, for the non-netted portfolio of Bank of Cluj, we computed the Expected Positive Exposure (EPE) arrays for each option, summing them with careful consideration of their dimensions.Following the implementation of all relevant formulas, we obtained the results as follows:

Table 5: Salzburg Bank

Measure	Value
EAD Call	15.1547
EAD Put	22.9404
EAD Portfolio	40.5665
RWA PUT	53.9009
RWA CALL	32.5219
RWA PORTFOLIO	87.0555

Table 6: Bank of Cluj

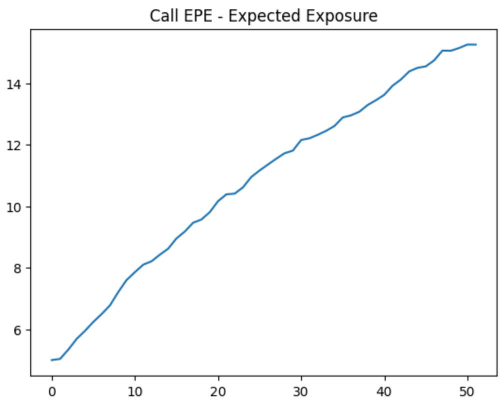
Measure	Value
EAD Call	11.9255
EAD Put	27.2640
EAD Portfolio	40.1920
RWA PUT	91.5148
RWA CALL	37.9142
RWA PORTFOLIO	127.7807

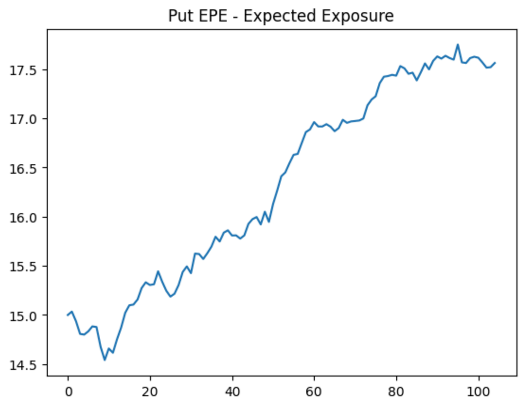
Table 7: Bank of Mazowsze

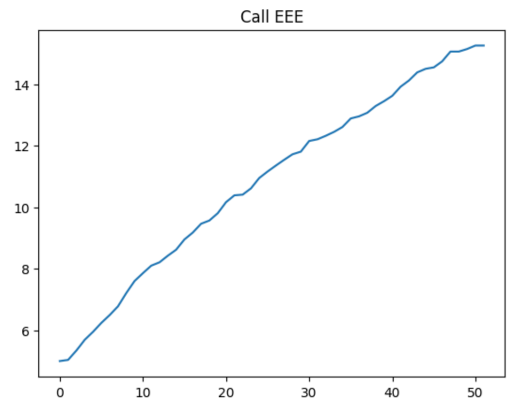
Measure	Value
American EAD	667.5845
Asian EAD	454.8494
Portfolio EAD	1094.4719
RWA American	2439.0931
RWA Asian	1592.2728
RWA Portfolio	3917.9329

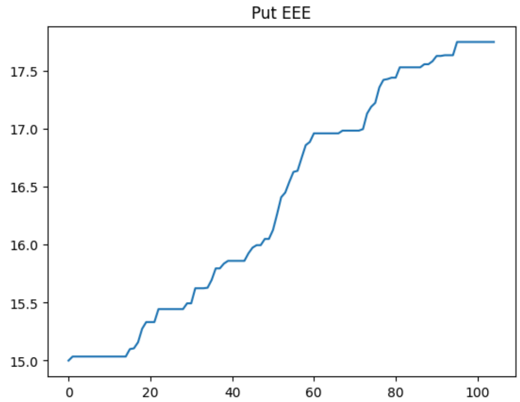
3.3.1 Illustrative Example: The Salzburg Bank Case

In this section we provide an illustrative example of the method used. Below you find the visual representation of the Exposure metrics for the options in the portfolio of Salzburg Bank. The first graphs show the EPEs: they represent how the exposure behaves through time for the option. The EEEs highlight when the exposure is increasing with respect to the past: it shows how the exposure is increasing throughout time.

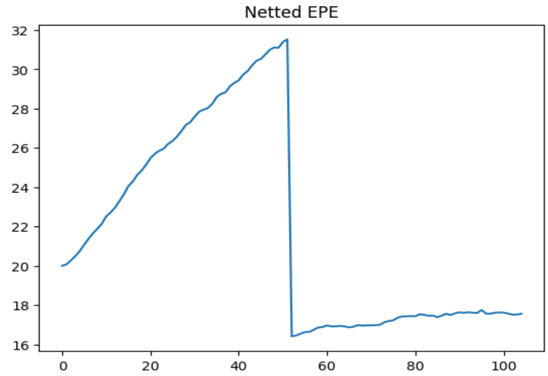


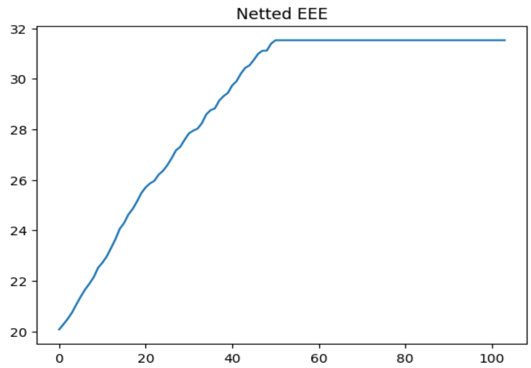






For the perspective of the whole portfolio of the bank, below we present the netted Expected Positive Exposure and the Effective Expected Exposure for the portfolio of Salzburg Bank.

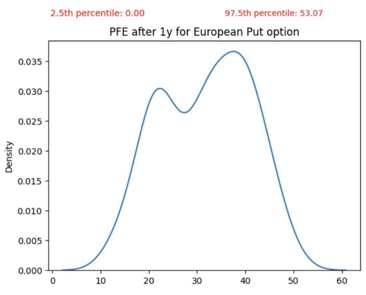


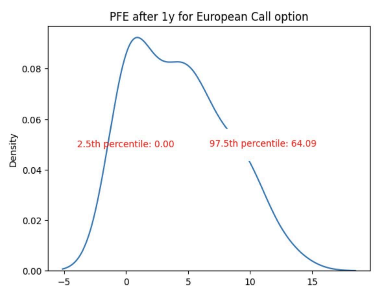


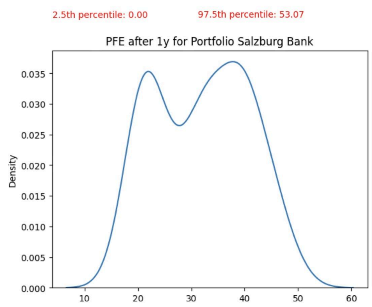
In terms of the portfolio level, we can notice a drop in the middle of the EPE graph: it represents the expiration of the first option, which is why the exposure decreases.

3.3.2 Potential Future Exposure

In this section, we deal with Potential Future Exposure. It gives the distribution of the exposure at a specific time to maturity. It is important to study the 2.5th percentile and the 97.5th percentile: the first represents the minimal amount of money we are going to lose if the counterparty default at that time, and the second one represents the biggest amount of money we are going to lose if the counterparty default at that time.
As an illustrative example, below we plot the PFE for the options of Salzburg Bank to have a visual representation.

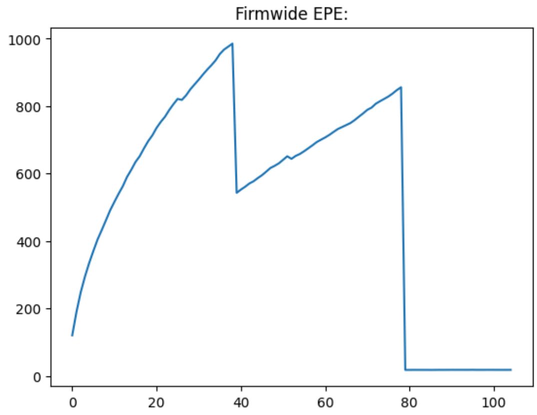


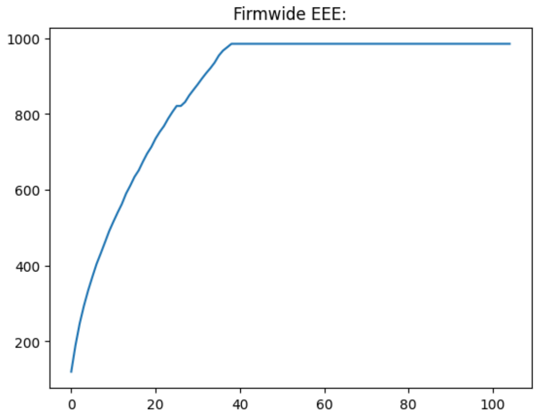




3.3.3 Exposure from a Firm level

In this section, we compute the total exposure given by the sum of three portfolios. We summed them with a non-netting approach, because we are dealing with three different counterparties so we can not apply netting between each other.
We took the three arrays of EPEs of the three portfolios, and we summed them obtaining the firmwide EPE . Then we also computed the EEE. Below we plot them in order to get a visual representation:





As before, we can see that in the EPE graph there are many pitfalls. They represents the moments when options expire. The first pitfall happens at time 26 representing the moment when the Asian Call Option from the Bank of Cluj expires. We can notice that it declines slightly because the value its smaller than the total amount.
The second pitfall happens at time 42, showing the moment when the Put Asian Option from Bank of Mazowsze expires. In this case given the higher exposure of that option we obtain a bigger pitfall.
The third pitfall , at time 52 which is the moment in which the Call Option of Salzburg expires. Notice as above the pitfall is quite small.
The last pitfall , at time 78, noticeably shows the moment where both the put asian option of the bank of Cluj and the call American Option expire, as seen from the measure of the pitfall.