MSc Program

Economic Data Mining and Analysis of MakerDAO DeFi Project

Student: Sudarut Kasemsuk
Research Advisor: Yury Yanovich

Skoltech

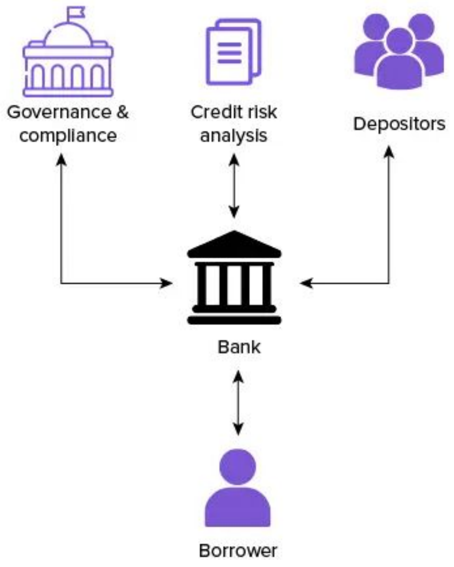
25 October, 2023

Table of contents

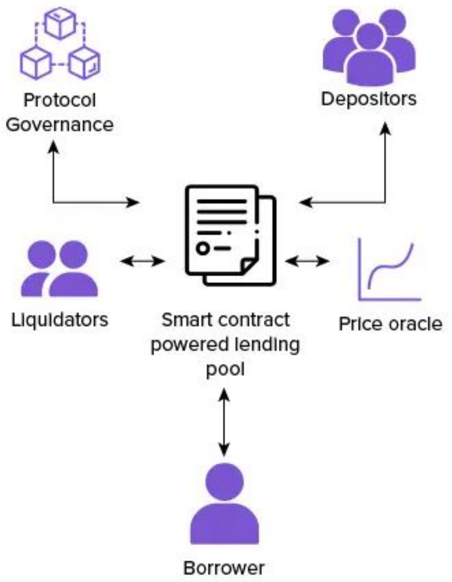
Introduction
&
Problem Statement

Project Background

Traditional Lending



DeFi Lending Platform



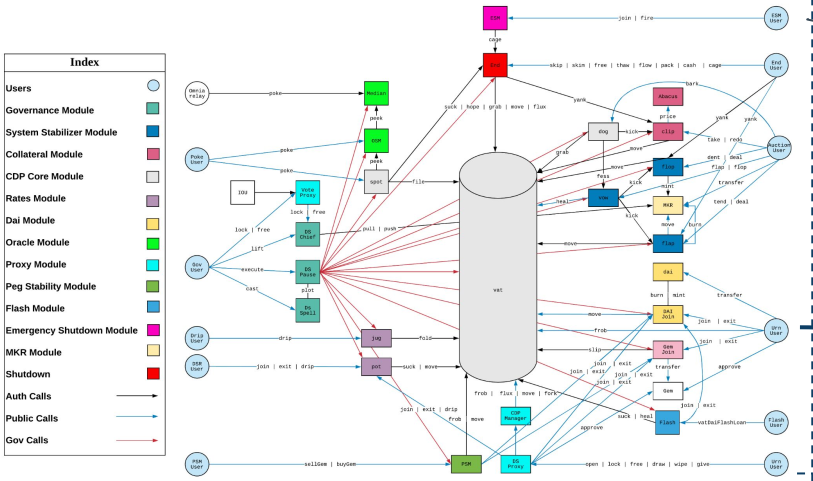
- Permissionless
- Transparency
- Lending Analysis

Image credit: https://appinventiv.com/blog/how-defi-lending-works/

Project Motivation

Skoltech

MakerDAO structure of data



Etherscan

From: 0xBfaeD4F3bc89c22e6937dF00D004B343F5E5f1F5

Interacted With (To): 0xd07e86f68C7B9f9B215A3ca3E79E74Bf94D6A847 (DAO Maker: Staking)

ERC-20 Tokens Transferred: From DAO Maker: Staking To 0xBfaeD4...F5E5f1F5 For 3,394.506999550147602754 ($2,500.41)

```
Function: stakeEnd(uint256 stakeIndex, uint40 stakeIdParam)

MethodID: 0x343009a2
[0]:  0000000000000000000000000000000000000000000000000000000000000000
[1]:  00000000000000000000000000000000000000000000000000000000000037ac
```

View Input As | Decode Input Data | Advanced Filter

#	Name	Type
0	stakeIndex	uint256
1	stakeIdParam	uint40

Input UserID / Transaction hash

Output

Enquiry loan details

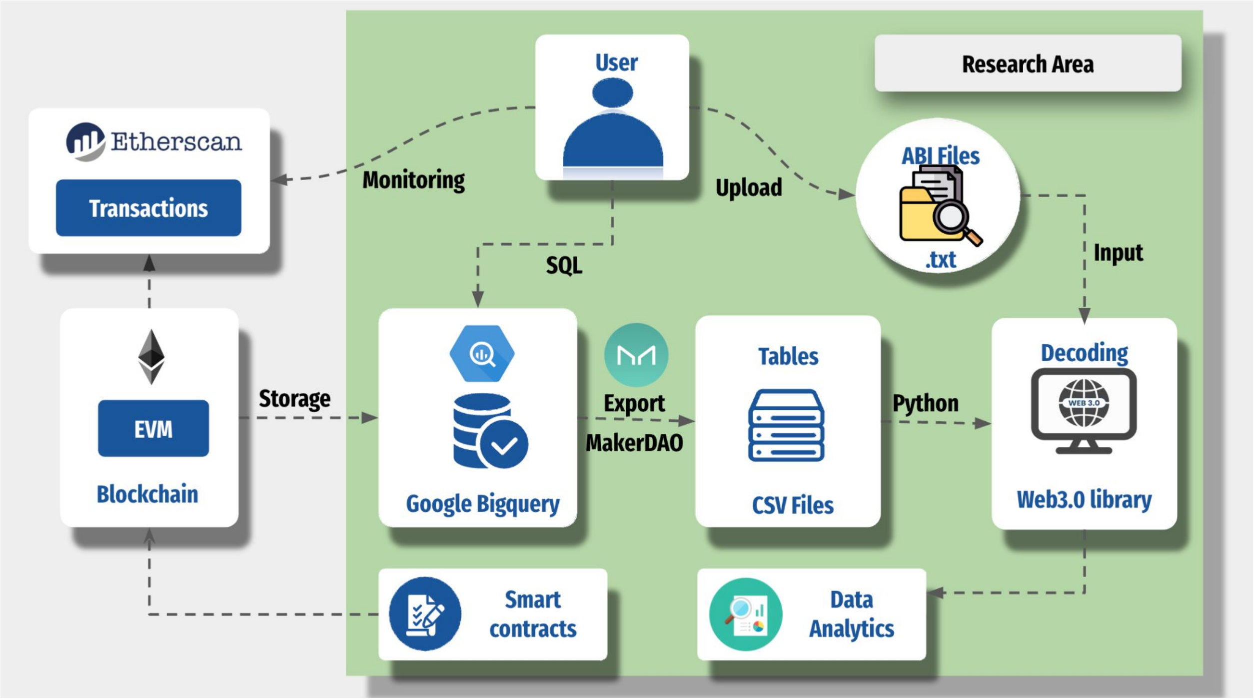
General details:
- loan status
- debt amount
- collateral ratio
- liquidation ratio

Historical details:
- No of loans
- All actions in a user

Statistical term:
- probability default (PD)
- loss given default (LGD)
- Log Equivalent Rate (LER)

Aim

Analyze the lending aspect of Maker from a traditional finance perspective



Overview Research Area

Objectives

1. Create a dataset with loan portfolios from MakerDAO
2. Equip it with financial characteristics related to borrowing
3. Develop a specialized mathematical model for the probability of default

Challenges

- Complicated data structure
- Make unstructured to structured data
- Decoding data from smart contracts
- Project-specific slang is far from the traditional terminology

Methodology

Mathematics

- Probability of default (PD) → Passage times of levels by geometric Brownian motion [1], [2]
- Probability of default (PD) → Classic finance Poisson model [3]
- Interest rate → Optimization problem by scipy(fsolve) [4]

Programming

- SQL command
- Smart contact protocol
- Python(Pandaparallel, Web3.0, Scipy etc.)

[1] A. N. Shiryaev. Essentials of stochastic finance. World Scientific, 1999.
[2] Hull, J.C. A model of the behavior of stock prices. Pearson, 2017
[3] Wasserman, L. All of Statistic. Springer Texts in Statistic. Springer, New York, NY, 2004
[4] Dennis, J. E.; Schnabel, Robert B. (1996). Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Society for Industrial and Applied Mathematics. pp. 228, 233, 265. ISBN 9781611971200.

MakerDAO

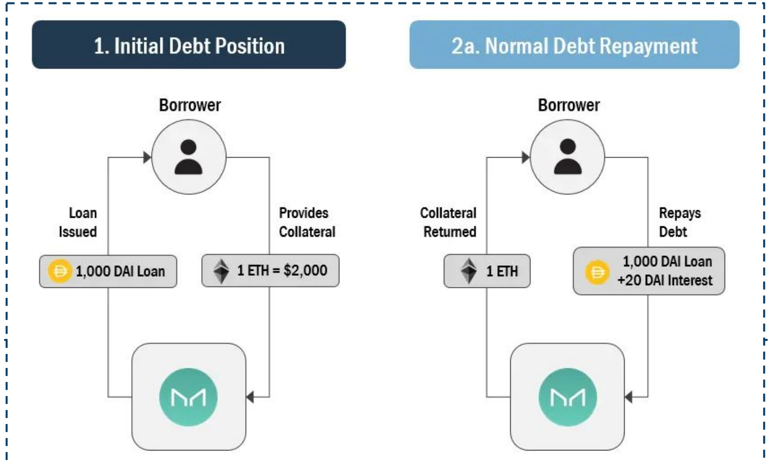
MakerDAO is an Ethereum-based **lending and borrowing platform** that give a stable **DAI** coin to borrowers. To get coins, user need to give asset to the platform as a **collateral** such as ETH, WBTC or ETC.

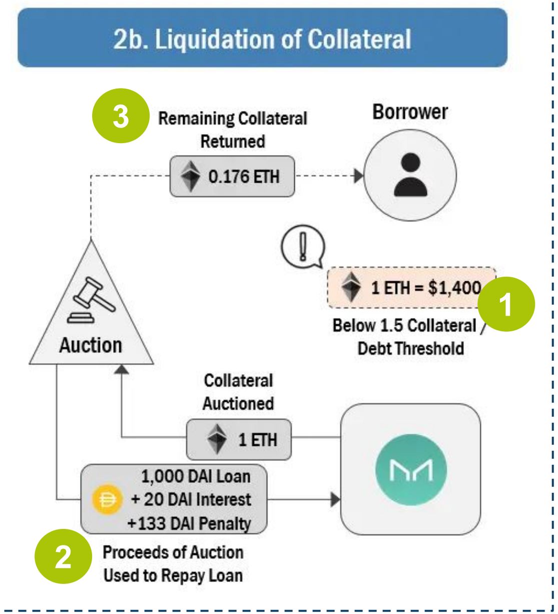
How does it works?

- Borrow DAI by **locking up assets** as collateral
- **Repay DAI + fee** to retrieve collateral back
- Liquidate if **collateral ratio < liquidation ratio**

Important parameters

stability fee: a **fee that manages the risk** associated with the coin's issuance.
collateral ratio: fraction between **collateral and Dai debt**
liquidation ratio: ratio for liquidation process is **by MakerDAO**





Source: MakerDAO. Vat detailed document, 2023
Image credit: https://messari.io/report/makerdao-valuation

Debt

Let $d(t)$ be the debt at time t, the interest is charged with the logarithm of the interest over time by $f(t)$. If no actions are taken on the debt during an interval $(t_1, t_2]$, then the debt at time t_2 can be calculate as:

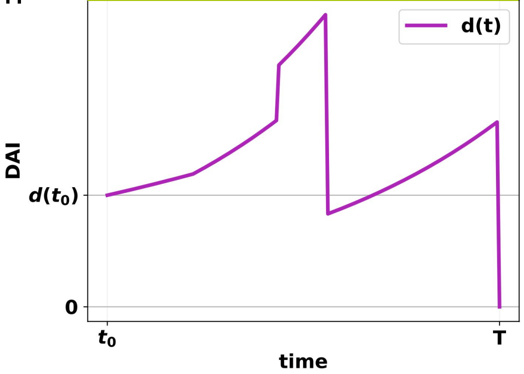
$$d(t_2) = d(t_1) \cdot \exp\left(\int_{t_1}^{t_2} f(t)dt\right).$$

The current collateral ratio $r(t)$ for $d(t) > 0$ equals to the following value:

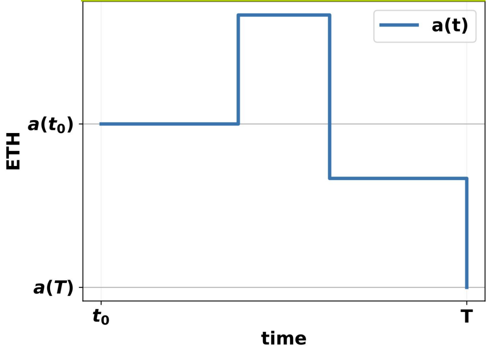
$$r(t) = \frac{e(t) \cdot a(t)}{d(t)}.$$

If $d(t) = 0$, we set $r(t) = +\infty$.
If $r(t)$ dr($r_{\min}(t)$ elow
liquidation will happen.

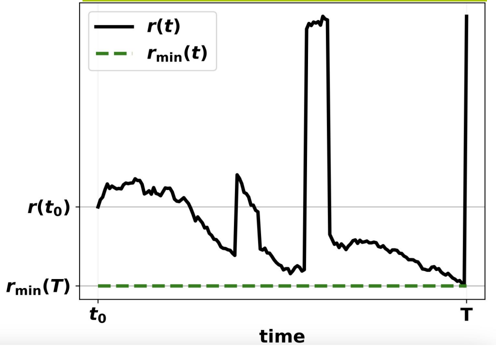
The debt d(t) as a function of time



The amount of collateral a(t)



The amount of collateral a(t)



Chaleenutthawut, Y., & Davydov, V. & Evdokimov, M. & Kasemsuk, S. & Kruglik, S. & Melnikov, G. & Yanovich, Y. (2023). "Loan Portfolio Dataset from MakerDAO Blockchain Project." Manuscript submitted for publication in the IEEE Journals.

Log Equivalent Rate (LER)

The log-interest rate is a constant log-interest rate that results in the same final debt, including liquidation losses. To LER , we use the cumul T ve $\mathrm{LER} = x,$ ime with denote by $h(x)$, which is calculated as

$$h(x) = \sum_{n=1}^{N} \Delta d_n \cdot \exp\left(x(T - t_n)\right).$$

The LER is then determined by solving the following equation for x :

$$\boxed{h(x) = d(T)} \xrightarrow[\text{happens}]{\text{if liquidation}} \boxed{h(x) = d(T) + \underbrace{(a(T) - a(T-)) \cdot e(T)}_{\text{we consider this term as the loss of collateral value during liquidation}},}$$

Where $a(T)$: collateral asset at time T

$e(T)$: price of asset at time T

$d(T)$: debt at time T

Chaleenutthawut, Y., & Davydov, V. & Evdokimov, M. & Kasemsuk, S. & Kruglik, S. & Melnikov, G. & Yanovich, Y. (2023). "Loan Portfolio Dataset from MakerDAO Blockchain Project." Manuscript submitted for publication in the IEEE Journals.

Skoltech

Log Equivalent Rate (LER)

To determine the average of D users at time t, we use a weighted average:

$$\overline{\text{LER}} = \frac{\sum_{i=1}^{D} d_i(t) \cdot \text{LER}_i}{\sum_{i=1}^{D} d_i(t)}.$$

Balance

We represent a user's balance at time *t* as:

$$\texttt{Bal}(t) = a(t) \cdot e(t) - d(t),$$

Chaleenutthawut, Y., & Davydov, V. & Evdokimov, M. & Kasemsuk, S. & Kruglik, S. & Melnikov, G. & Yanovich, Y. (2023). "Loan Portfolio Dataset from MakerDAO Blockchain Project." Manuscript submitted for publication in the IEEE Journals.

Loss Given Default (LGD)

To calculate $\mathtt{LGD}$ for a user's collateral liquidation at time *t*, we use the following formula

$$\mathtt{LGD}(t) = \frac{\mathtt{Bal}(t-) - \mathtt{Bal}(t)}{d(t-)}.$$

Where $a(t)$: collateral asset at time *t*

$e(t)$: price of asset at time *t*

$d(t)$: debt at time *t*

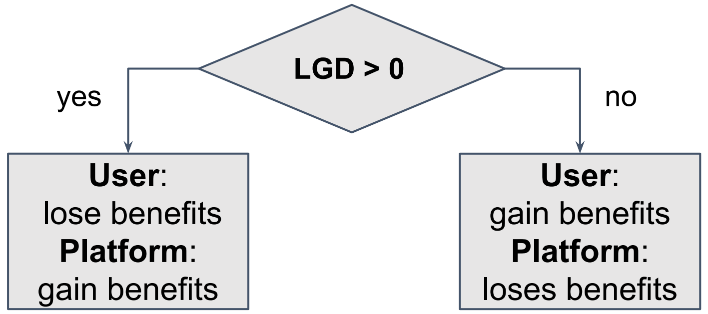
$t-$: *t* right before the liquidation

To calculate $\overline{\mathtt{LGD}}$ as a weighted average:

$$\overline{\mathtt{LGD}} = \frac{\sum_{d=1}^{D} d(t_d-) \cdot \mathtt{LGD}(t_d)}{\sum_{d=1}^{D} d(t_d-)}$$

$$= \frac{\sum_{d=1}^{D} \mathtt{Bal}(t_d-) - \mathtt{Bal}(t_d)}{\sum_{d=1}^{D} d(t_d-)}.$$

LDG Values



Chaleenutthawut, Y., & Davydov, V. & Evdokimov, M. & Kasemsuk, S. & Kruglik, S. & Melnikov, G. & Yanovich, Y. (2023). "Loan Portfolio Dataset from MakerDAO Blockchain Project." Manuscript submitted for publication in the IEEE Journals.

Poisson Model (PD)

Poisson model assumes that all debts are independent and have an exponential distribution with $\lambda > 0$ for time until default [5]. This simplification allows for the estimation of λ [1], [6]. The maximum likelihood estimater(MLE) $\hat{\lambda}$ of the parameter λ is

$$\hat{\lambda} = \frac{N+M}{\sum_{n=1}^{N} x_n + \sum_{m=N+1}^{N+M} y_m}.$$

Probability of default (PD) for a single debt during interval time T, where X is an exponential random variable with parameter λ can be written as:

$$PD(T) = P(X < T) = 1 - \exp(-\lambda T).$$

As the likelihood is functional equivariant [6], the MLE for PD is

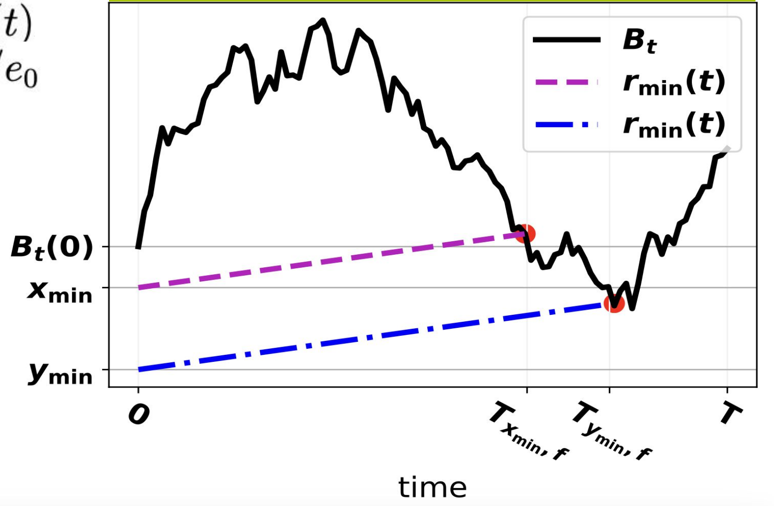
$$\hat{\mathrm{PD}}(T) = 1 - \exp(-\hat{\lambda} T)$$

[1] A. N. Shiryaev. Essentials of stochastic finance. World Scientific, 1999.
[5] A. N. Shiryayev, Probaility, ser. Graduate Texts in Mathematics 95. New York: Springer-Verlag, 1984.
[6] L. Wasserman, All of Statistics, ser. Springer Texts in Statistics. New York, NY: Springer, 2004.

Brownian Motion Model (PD)

The minimal allowed collateralization ratio $r_{\min}(t)$ in comparison to the actual user's collateralization $r(t)$ (from slide 11). We assume that the logarithm of $e(t)/e_0$ follows a Brownian Motion with zero mean and standard deviation $\sigma > 0$. Therefore, $\frac{1}{\sigma}(\ln \frac{e(t)}{e_0})$ is a Brownian motion B_t with zero mean and unit variance

The default can de describe by BM model



[1] *A. N. Shiryaev. Essentials of stochastic finance. World Scientific, 1999.*

Brownian Motion Model (PD)

Theorem 1. If

1. Normalized exchange rate $\frac{1}{\sigma}(\ln\frac{e(t)}{e_0})$ for a given constant $\sigma > 0$ is a Brownian motion B_t with zero mean and unit variance
2. Borrower has a debt d_0 and collateral a_0 at time $t = 0$
3. Borrower has no actions with debt and collateral during $t \in (0, T]$
4. Platfrom's interest rate $f \geq 0$ and the minimum collateralization ratio $r_{\min} > 0$ are constant

then the probability of the borrower's default during the time interval $(0, T]$ and its variance are given by

$$PD = \psi(x_{\min})$$

$$= \int_0^T \frac{|x_{\min} + fs|}{\sqrt{2\pi s^3}} e^{-\frac{(x_{\min}+fs)^2}{2s}} ds.$$

where

$$x_{\min} = \frac{1}{\sigma} \ln\left(\frac{d_0 \cdot r_{\min}}{a_0 \cdot e_0}\right).$$

[1] *A. N. Shiryaev. Essentials of stochastic finance. World Scientific, 1999.*

Brownian Motion Model (PD)

Theorem 2. If, in addition to the assumptions 1 - 4 of Theorem 1 slide 17,

5. the second borrower has a debt $\tilde{d}_0$ a collateral $\tilde{a}_0$ at time $t = 0$,

then the covariance of two borrowers' defaults during time interval $(0, T]$ equals

$$\mathtt{cov} = \psi(\min\{x_{\min}, y_{\min}\}) \cdot (1 - \psi(\max\{x_{\min}, y_{\min}\}))$$

where

$$y_{\min} = \frac{1}{\sigma} \ln \left(\frac{\tilde{d}_0 \cdot r_{\min}}{\tilde{a}_0 \cdot e_0} \right).$$

The maximum likelihood estimator is

$$\hat{\sigma} = \sqrt{\frac{1}{N} \sum_{n=1}^{N} \left(\ln \left(\frac{e(t_n)}{e(t_{n-1})} \right) \right)^2 / (t_n - t_{n-1})}.$$

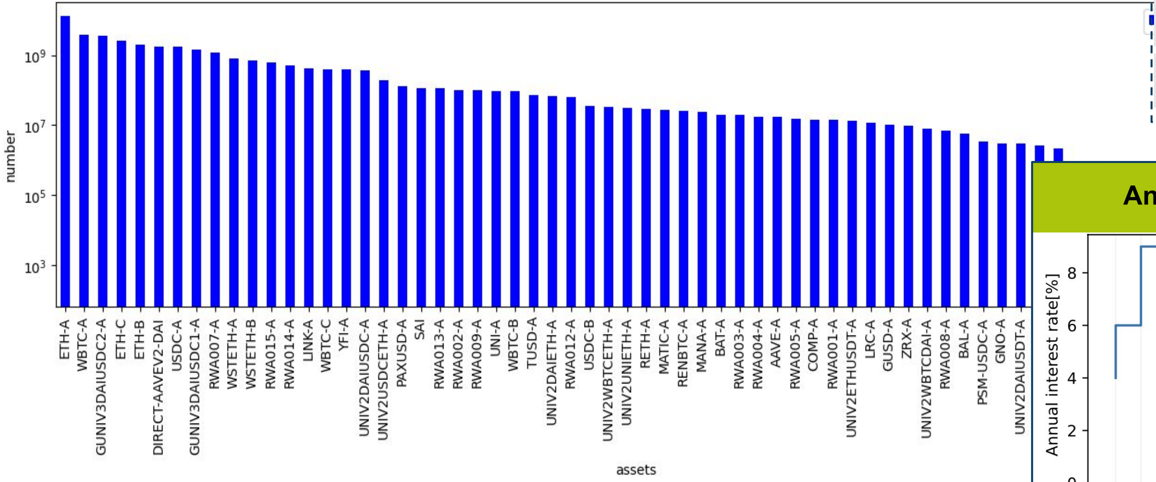
[1] A. N. Shiryaev. Essentials of stochastic finance. World Scientific, 1999.

Results

Dataset

MakerDAO datasets since platform start on **11 Nov 2019 - 31 Jul 2023**

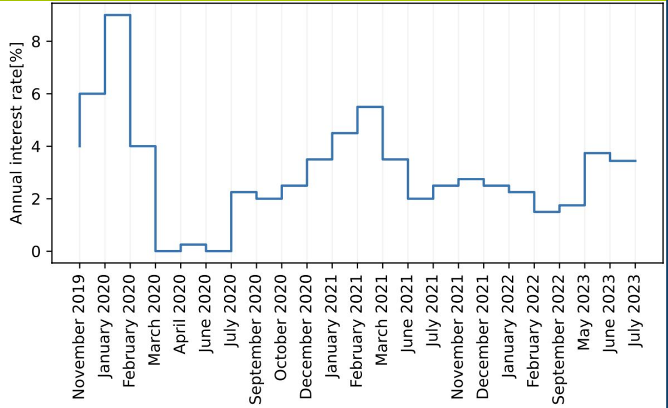
Total amount DAI debt of 46 assets



MakerDAO Dataset

	ETH-A	total
Number of debts	137,441	259,048
Volumn in DAI	13.4 B	36.9 B

Annual Maker's interest rate for ETH-A

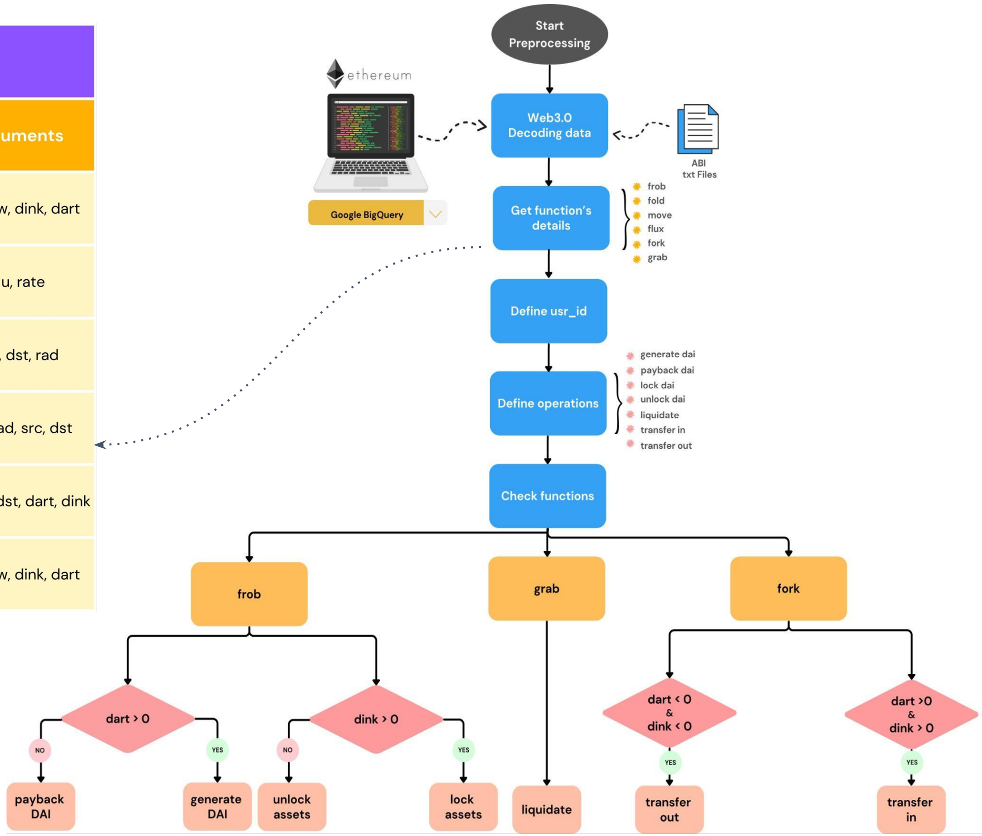


Data Extraction and Processing

Function details

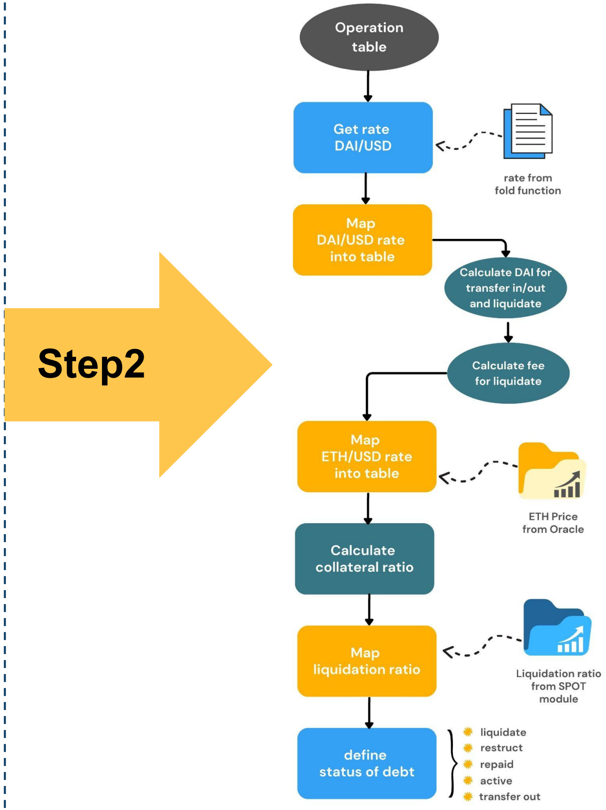
Functions	Descriptions	Arguments
frob	generate/return DAI, lock/unlock assets	i, u, v, w, dink, dart
fold	DAI/USD rate	i, u, rate
move	transfer stable coin between users	src, dst, rad
flux	transfer collateral between users	ilk, wad, src, dst
fork	splitting vault	ilk, src, dst, dart, dink
grab	liquidate a vault	i, u, v, w, dink, dart

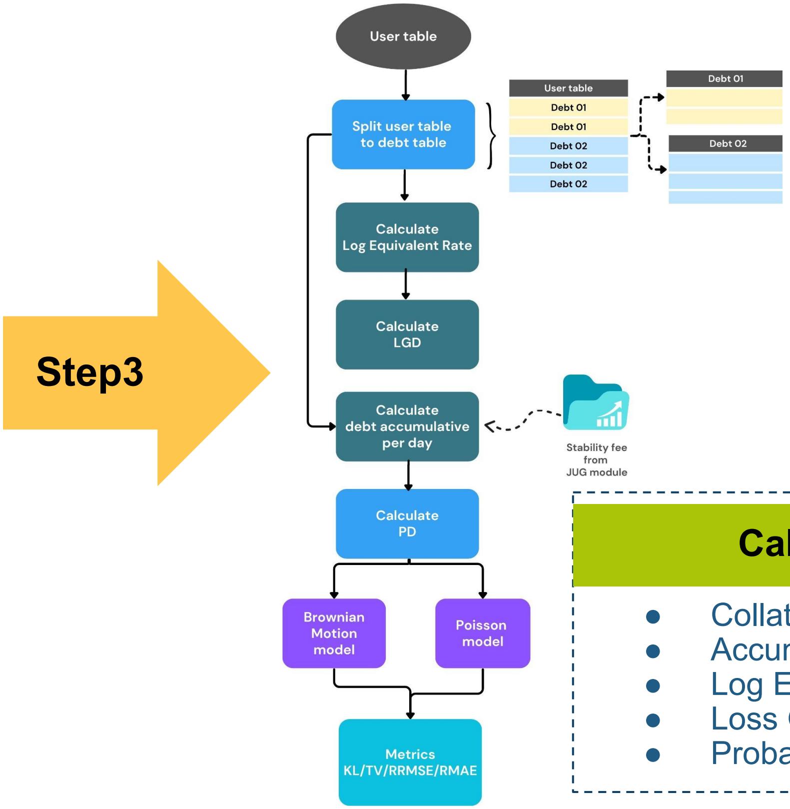
Step1



Github:
https://github.com/Sudarut-kas/Data-Mining-for-MakerDAO/tree/main

Data Processing





Calculations

- Collateral Rate
- Accumulative Debt
- Log Equivalent Rate
- Loss Given Default
- Probability Default

Structured Dataset

Table1: Loan actions of user 0x931dBd7001D14112D17304B78d305c4FE317E571

Timestamp [yy-mm-dd hh-mm-ss]	Action	Collateral [ETH]	Debt [DAI]	Fee [DAI]	Debt [USD]	Rate [DAI/USD]	Rate [ETH/USD]	Collateral Rate	Liquidation Rate
20-02-15 01:42:39	lock asset	0.3000	0.0000	0.0000	0.0000	1.013186	286.474769	0.0000	1.5
20-02-15 01:42:39	generate dai	0.0000	34.3770	0.0000	33.9296	1.013186	286.474769	2.5000	1.5
20-02-15 13:10:55	unlock asset	-0.1000	0.0000	0.0000	0.0000	1.013288	284.858357	1.6573	1.5
20-02-16 18:00:01	liquidate	-0.2000	-34.3770	-0.0122	-33.9296	1.013545	249.382446	1.4509	1.5
20-03-13 20:28:37	lock asset	0.0404	0.0000	0.0000	0.0000	1.019139	121.323152	0.0000	1.5
20-03-13 20:59:40	lock asset	2.0000	0.0000	0.0000	0.0000	1.019144	121.323152	0.0000	1.5
20-03-13 21:03:20	generate dai	0.0000	165.0297	0.0000	161.9297	1.019144	121.323152	1.5000	1.5
20-03-13 21:03:20	lock asset	3.1989	0.0000	0.0000	0.0000	1.019144	121.323152	3.8517	1.5
20-03-13 21:03:20	generate dai	0.0000	245.1908	0.0000	240.5850	1.019144	121.323152	1.5495	1.5
20-03-13 21:50:37	generate dai	0.0000	46.7635	0.0000	45.8847	1.019151	130.835000	1.5000	1.5
20-03-13 21:50:37	lock asset	1.0593	0.0000	0.0000	0.0000	1.019151	130.835000	1.8033	1.5
20-03-13 21:50:37	generate dai	0.0000	91.3185	0.0000	89.6025	1.019151	130.835000	1.5030	1.5
20-03-13 22:10:47	liquidate	-6.2986	-548.3024	-0.0045	-538.0020	1.019154	129.463665	1.4872	1.5
20-03-14 00:54:08	lock asset	0.2500	0.0000	0.0000	0.0000	1.019178	136.220527	0.0000	1.5
20-03-14 03:01:40	generate dai	0.0000	22.0042	0.0000	21.5897	1.019197	132.025000	1.5000	1.5
20-03-14 03:01:40	lock asset	0.4161	0.0000	0.0000	0.0000	1.019197	132.025000	3.9966	1.5
20-03-14 03:01:40	generate dai	0.0000	33.0255	0.0000	32.4035	1.019197	132.025000	1.5981	1.5
20-03-14 03:05:13	lock asset	0.1000	0.0000	0.0000	0.0000	1.019198	132.025000	1.8380	1.5
20-03-14 03:06:46	generate dai	0.0000	12.3999	0.0000	12.1663	1.019198	132.025000	1.5000	1.5
20-03-14 03:06:46	lock asset	0.2265	0.0000	0.0000	0.0000	1.019198	132.025000	1.9435	1.5
20-03-14 03:06:46	generate dai	0.0000	17.6159	0.0000	17.2841	1.019198	132.025000	1.5409	1.5
20-03-14 03:21:27	lock asset	1.0000	0.0000	0.0000	0.0000	1.019201	133.573992	3.1296	1.5
20-03-14 03:38:30	generate dai	0.0000	92.3912	0.0000	90.6504	1.019203	133.573992	1.5000	1.5
20-03-14 03:38:30	lock asset	1.5000	0.0000	0.0000	0.0000	1.019203	133.573992	2.6292	1.5
20-03-14 03:38:30	generate dai	0.0000	107.7055	0.0000	105.6762	1.019203	133.573992	1.6361	1.5
20-03-14 03:48:09	unlock asset	-0.2760	0.0000	0.0000	0.0000	1.019204	133.573992	1.5068	1.5
20-03-14 03:48:09	payback dai	0.0000	-31.2139	0.0000	-30.6258	1.019204	133.573992	1.6920	1.5
20-03-14 03:51:13	unlock asset	-0.3179	0.0000	0.0000	0.0000	1.019204	133.573992	1.5248	1.5
20-03-14 03:51:13	payback dai	0.0000	-36.6915	0.0000	-36.0002	1.019204	133.573992	1.7823	1.5
20-03-14 05:27:38	unlock asset	-0.4331	0.0000	0.0000	0.0000	1.019220	132.160000	1.5000	1.5
20-03-14 05:27:38	payback dai	0.0000	-129.5872	0.0000	-127.1437	1.019220	132.160000	3.7177	1.5
20-03-14 05:27:38	unlock asset	-0.5674	0.0000	0.0000	0.0000	1.019220	132.160000	2.8621	1.5
20-03-14 05:30:09	generate dai	0.0000	13.1757	0.0000	12.9272	1.019220	132.160000	2.4881	1.5
20-03-14 05:30:09	lock asset	0.0854	0.0000	0.0000	0.0000	1.019220	132.160000	2.6001	1.5
20-03-14 05:55:06	generate dai	0.0000	73.0000	0.0000	71.6232	1.019223	132.160000	1.5081	1.5
20-03-14 06:46:34	lock asset	0.1500	0.0000	0.0000	0.0000	1.019231	133.595000	1.6398	1.5
20-03-15 01:10:55	liquidate	-2.1336	-173.8252	-0.0203	-170.5509	1.019317	121.240000	1.4881	1.5
20-03-15 20:55:55	lock asset	0.4748	0.0000	0.0000	0.0000	1.019407	123.468601	0.0000	1.5
20-03-15 23:01:50	unlock asset	-0.4748	0.0000	0.0000	0.0000	1.019417	122.476715	0.0000	1.5

Actions

- Lock/ Unlock assets
- Generate/ Payback DAI
- Liquidate
- Transfer in/out

Structured Dataset

Table 2: Status all debts of user 0x931dBd7001D14112D17304B78d305c4FE317E571

debt_number	start_date	end_date	status
01	2020-02-15 01:42:39	2020-02-16 18:00:01	liquidated
02	2020-03-13 20:59:40	2020-03-13 22:10:47	restructured
03	2020-03-14 03:01:40	2020-03-15 01:10:55	restructured

Status

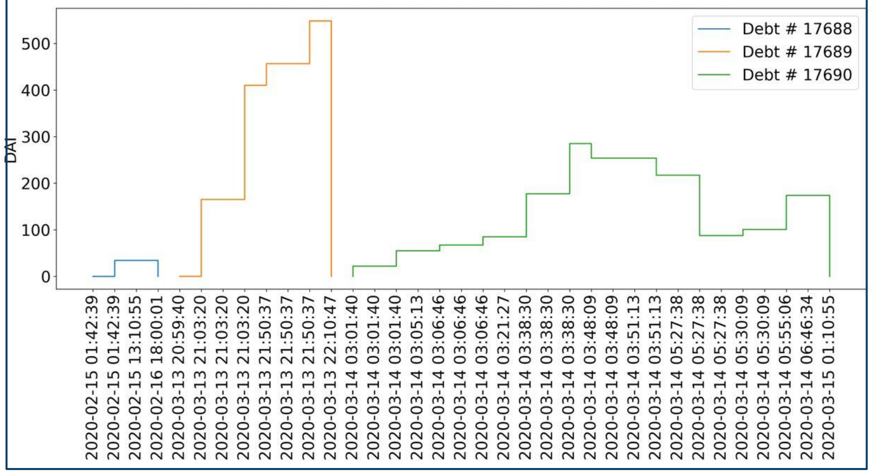
Repaid: the debt is repaid by full amount

Liquidated: the debt is fully liquidated

Restructured: the debt is partially liquidated (new debt start immediatly)

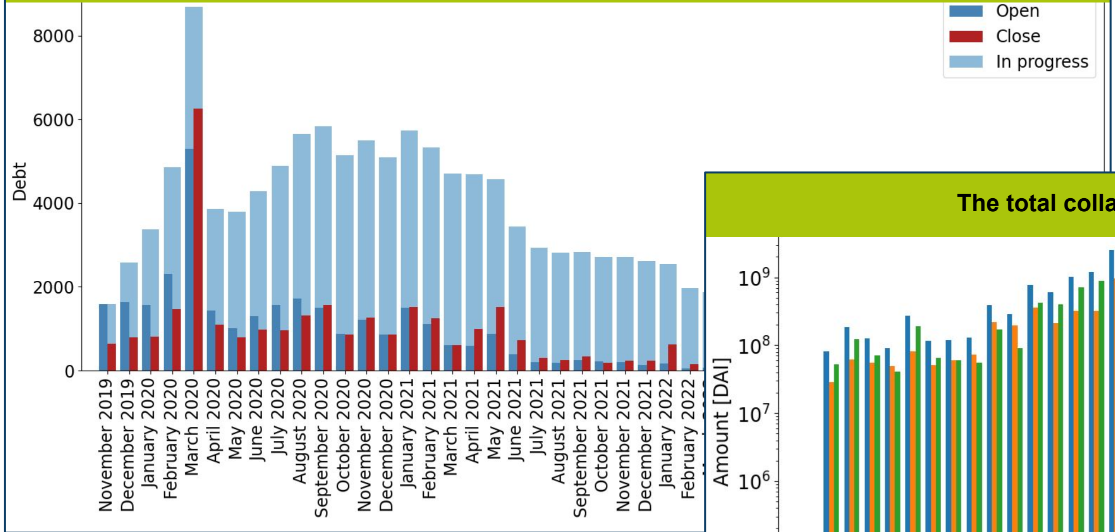
Active: the loan is still active

Debt over the tme of user 0x931dBd7001D14112D17304B78d305c4FE317E571

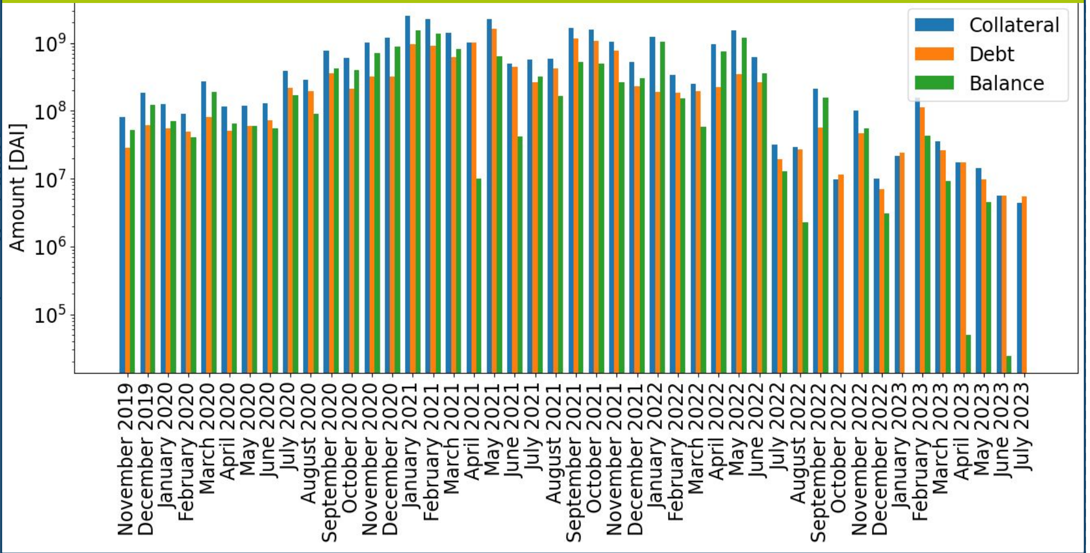


General Statistics

The number of ETH debts

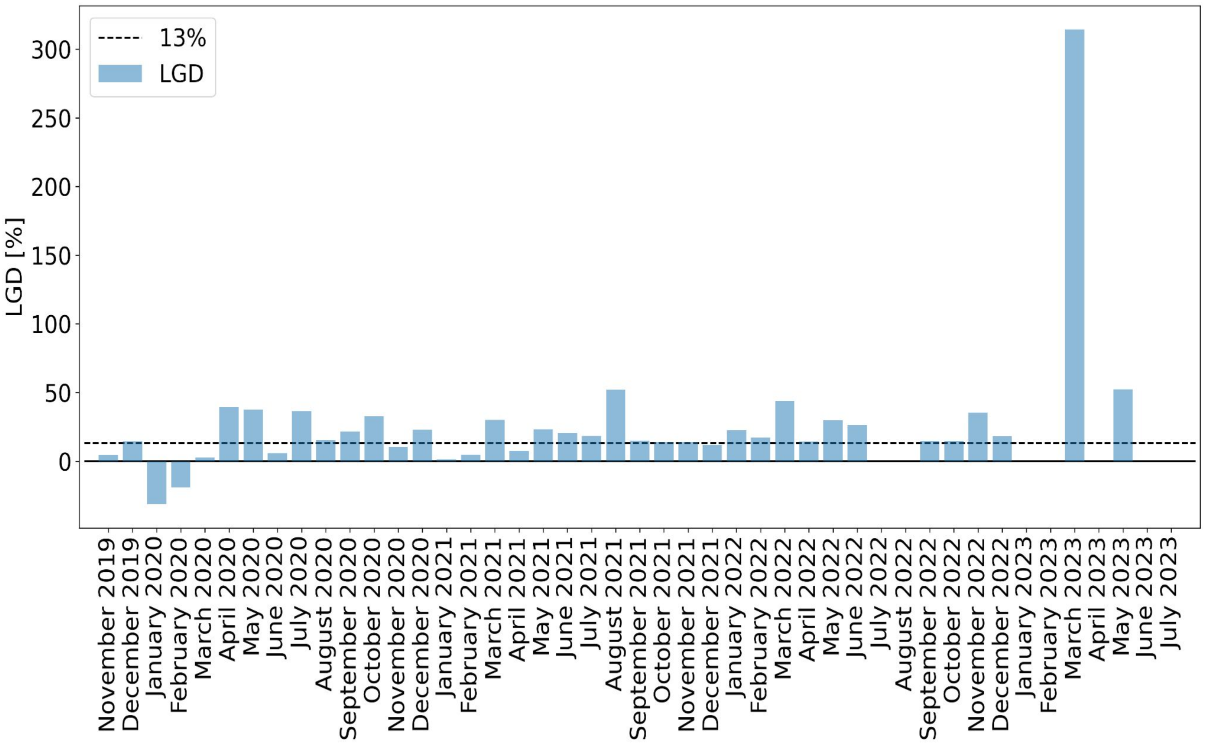


The total collateral amount and borrowers’ balance



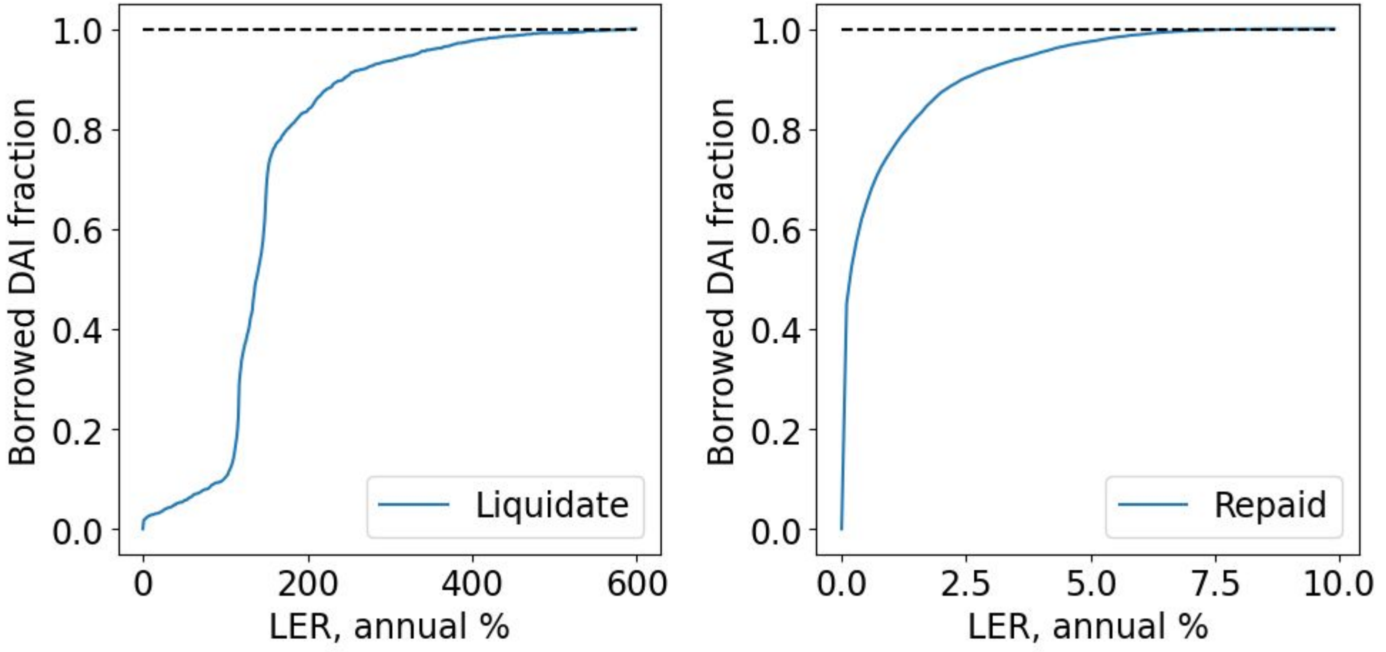
Economic Financial Results

Monthly average LGD



Economic Financial Results

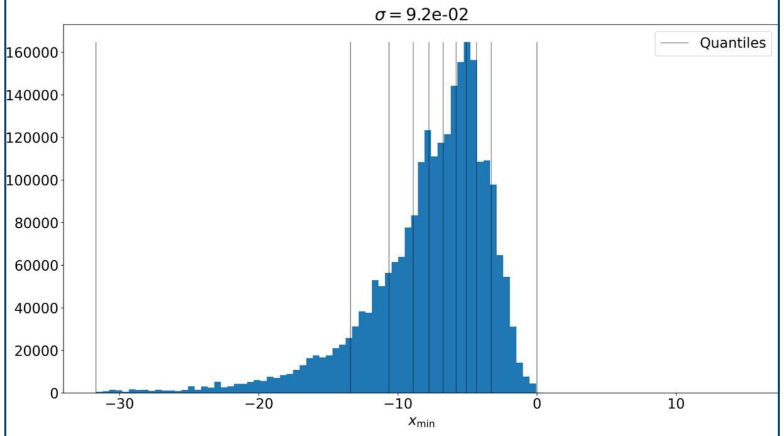
Monthly average annual equivalent rate(AER) for liquidated (left) and repaid (right) debts



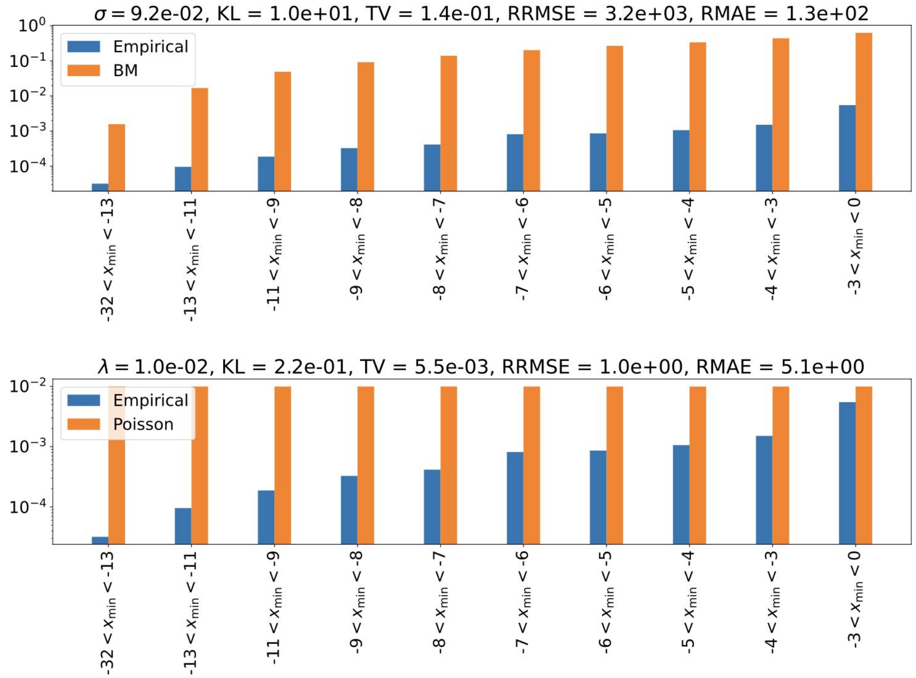
Numerical Experiments

Skoltech

The distribution of x_min 10% quantile

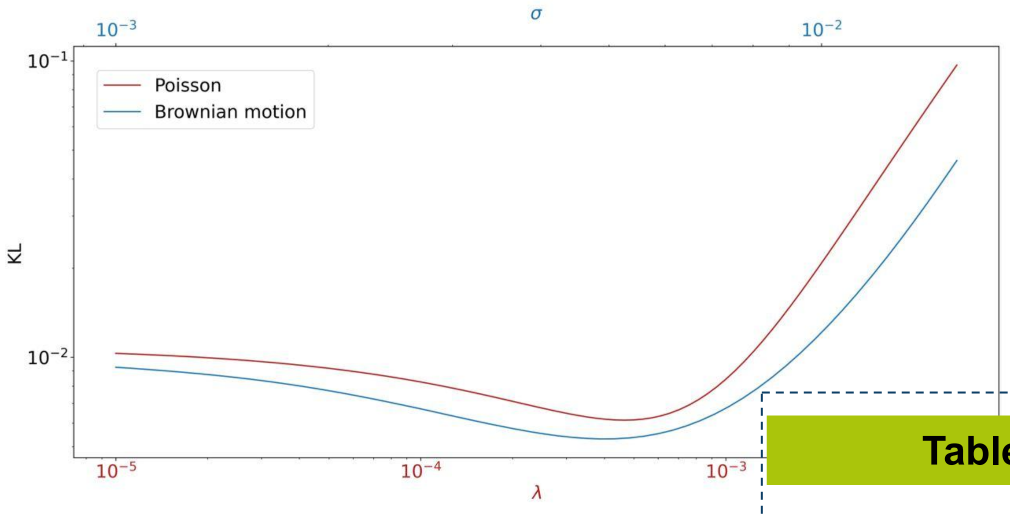


Mean empirical PD and mean theoretical PD estimated by BM(top) and PS(bottom) for each 10% quantile by x_min



Numerical Experiments

KL's PS metric value vs lambda hat value and KL's BM metric value vs sigma hat value



Optimal values vs the values predicted by MLE:

$$\hat{\lambda}_{\text{opt}} = 4.6 \cdot 10^{-4}, \quad \hat{\lambda}_{\text{MLE}} = 1.0 \cdot 10^{-2}$$
$$\hat{\sigma}_{\text{opt}} = 4.8 \cdot 10^{-3}, \quad \hat{\sigma}_{\text{MLE}} = 9.2 \cdot 10^{-2}$$

Table: comparison of PD models

	KL	TV	RRMSE	RMAE
Poisson model	0.0061	0.00077	1.00	0.72
Brownian Motion Model	0.0053	0.00078	1.08	0.72

Conclusions

Main contributions

- Created a dataset with loan portfolios from MakerDAO, which is publicly available on Github
- Formulated, implemented and computed borrowing financial characteristics for the dataset
- Formulated and proved Theorems 1 and 2 to provide project-specific model for the probability of the default.

Github:
https://github.com/Sudarut-kas/Data-Mining-for-MakerDAO/tree/main

Conclusions

- The huge unstructured data set is currently in excellent condition. It serves as the foundation for future research.
- Brownian motion can better estimate the probability of default (the profit of the platform) than other financial models.

Publications

Chaleenutthawut, Y., & Davydov, V. & Evdokimov, M. & Kasemsuk, S. & Kruglik, S. & Melnikov, G. & Yanovich, Y. (2023). ”Loan Portfolio Dataset from MakerDAO Blockchain Project.” Manuscript submitted for publication in the IEEE Journals.

Scientific novelty

- ❖ In this research, we implemented functions for transform data from smart contract into structured dataset
- ❖ Introducing dataset from MakerDAO loanfolio platfrom
- ❖ Introducing optimal interest rate for borrower
- ❖ Formulation the mathematical economic formula for probability of default

Innovation

This study focuses on data mining processing on blockchain smart contracts, which were previously unstructured on MakerDAO and have now been systematically structured.

Futurework

Extent to multiple collaterals and to other protocols

Skoltech

Skoltech

Acknowledgements

Yury Yanovich, Research advisor

thx.

Skoltech