MQLV: Optimal Policy of Money Management in Retail Banking with Q-Learning

Jérémy Charlier

Last Year PhD Student at University of Luxembourg
Visiting PhD Student at Columbia University

Outline

Context

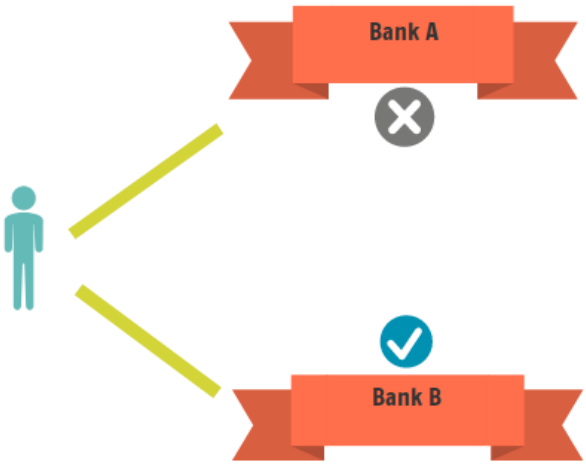

Figure 1: Loan applications represent one of the highest risks of customer churn.

Competitive retail banking market

Debt applications, such as loans or credit cards, highly sensitive

Requirement for custom decisions adapted to each client

Objective of minimizing the customer churn

Research Question

How can the client debt applications in retail banking
be tailored using their individual policy
and the optimal policy of money management?

Solution and Contributions

- A financial RL framework based on aggregated transactions
- Definition of a digital function to model default events
- First application of Q-learning for optimal policy of money management

Definitions

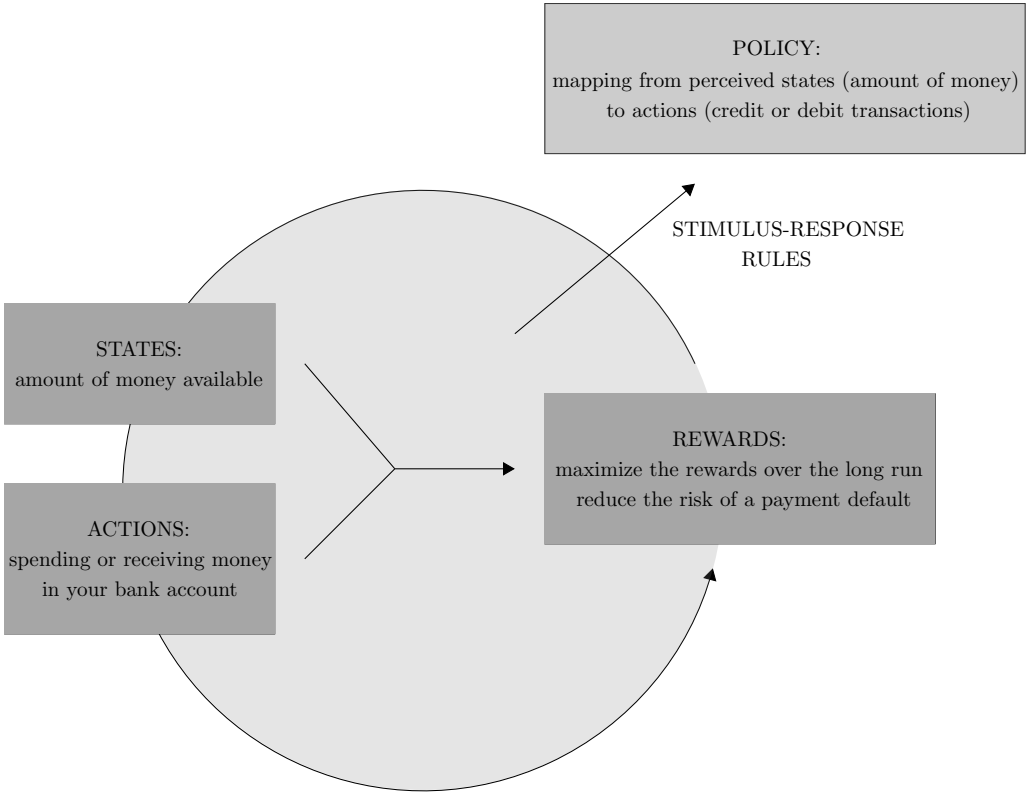

Figure 2: In reinforcement learning, the optimal policy is learned by maximizing the rewards based on state-action sequences.

MQLV: Optimal Policy of Money Management

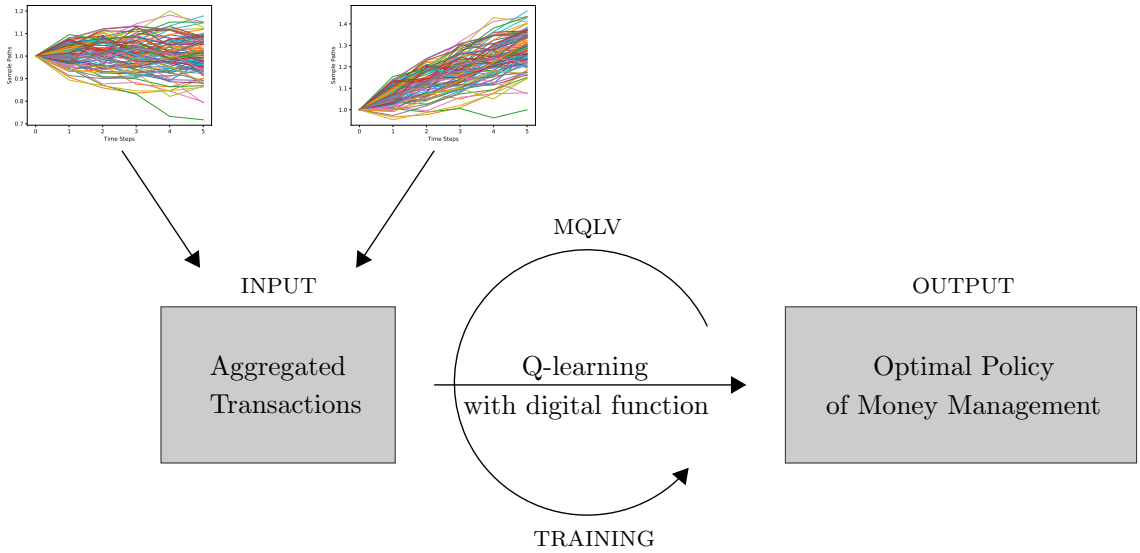

Figure 3: MQLV takes as input the aggregated financial transactions. The training is performed using the Bellman equation updated with the digital function. At convergence, the optimal policy of money management is obtained.

Experiments

Three different experiments to validate MQLV

- Validation of generated Vasicek transactions
- Validation of MQLV with the BSM option pricing formula
- MQLV and the choice of parameters

First experiment: validation of generated transactions

Comparison between generated Vasicek transactions and public anonymous transactions

- Impossible to release an anonymized transactions data set

Use of the Santander public product recommendation data set

- Released in 2016
- 16 months of customers financial behavior
- 22 transaction labels
- `https://www.kaggle.com/c/santander-product-recommendation`

First experiment: validation of generated transactions

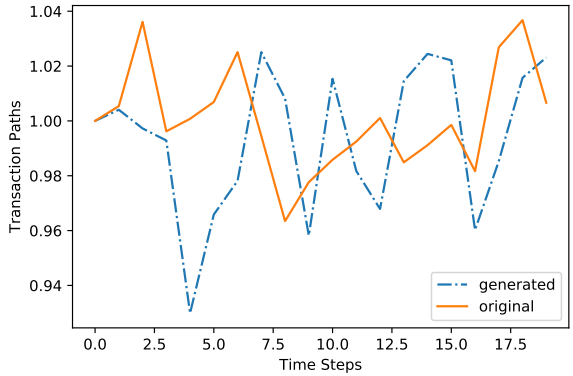

Figure 4: Samples of original and Vasicek generated transactions for one client.

Description	Value
RMSE	0.0335
Vasicek speed reversion a	0.5444
Vasicek long term mean b	0.9001
Vasicek volatility σ	0.2185

Figure 5: RMSE and calibrated parameters of the Vasicek transactions.

Main Results

- Strong similarities between the dynamic of the Santander anonymized transactions and the Vasicek generated transactions
- Supports the hypothesis that the Vasicek model could be used to generate synthetic transactions

Second experiment: comparison of MQLV with BSM

Aim at learning the optimal policy of money management

- Use of the BSM's closed formula for vanilla option pricing
- In our configuration, target is 50% at a strike of 1

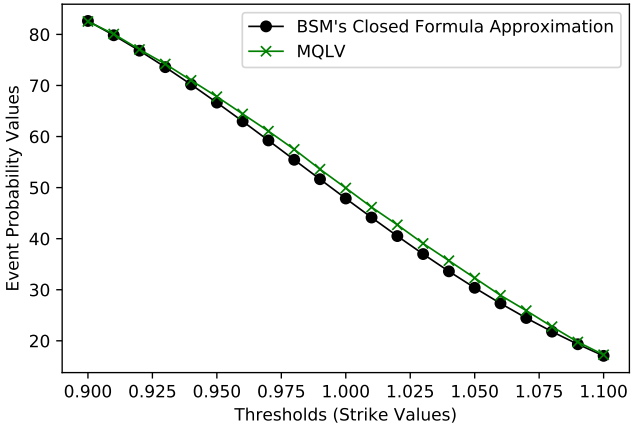

Figure 6: Event probability values calculated by MQLV and BSM.

Data Set	Number of Paths	Strike Values	BSM's Approx. Values (%)	MQLV Values (%)	Absolute Difference
1	20,000	0.92	76.810	**77.098**	0.288
1	20,000	0.98	55.447	**57.920**	2.473
1	20,000	1.00	47.867	**50.235**	2.368
1	20,000	1.02	40.509	**42.865**	2.356
2	30,000	0.92	76.810	**76.953**	0.143
2	30,000	0.98	55.447	**57.760**	2.313
2	30,000	1.00	47.867	**50.043**	2.176
2	30,000	1.02	40.509	**42.744**	2.235
3	40,000	0.92	76.810	**77.047**	0.237
3	40,000	0.98	55.447	**57.491**	2.044
3	40,000	1.00	47.867	**49.924**	2.057
3	40,000	1.02	40.509	**42.713**	2.204

Figure 7: Valuation differences of the digital values for event probabilities.

Second experiment: comparison of MQLV with BSM

Main Results

- Configuration free of any time-dependency
- Both the MQLV and the BSM's approaches are similar
  - with a RMSE of 1.5016
- Quantitative results show
  - MQLV tends to the theoretical limit of 50% at a strike of 1
  - Surprisingly, MQLV outperforms the digital approximation of BSM
- MQLV scores highlight its capability to learn the optimal policy

Third experiment: MQLV and the choice of parameters

Parameters $a; b; \sigma$	Number of Paths	Strike Values	BSM's App. Values (%)	MQLV Values (%)	Absolute Difference
0.01; 1; 0.10	50,000	0.98	59.856	**61.223**	1.366
0.01; 1; 0.10	50,000	1.00	48.562	**50.001**	1.439
0.01; 1; 0.10	50,000	1.02	37.596	**39.044**	1.447
0.01; 1; 0.30	50,000	0.98	49.558	**53.647**	4.089
0.01; 1; 0.30	50,000	1.00	45.767	**49.997**	4.230
0.01; 1; 0.30	50,000	1.02	42.088	**46.194**	4.106
0.10; 1; 0.15	50,000	0.98	55.447	**57.540**	2.093
0.10; 1; 0.15	50,000	1.00	47.867	**50.015**	2.148
0.10; 1; 0.15	50,000	1.02	40.509	**42.638**	2.129
0.30; 1; 0.15	50,000	0.98	55.447	**57.586**	2.139
0.30; 1; 0.15	50,000	1.00	47.867	**50.022**	2.155
0.30; 1; 0.15	50,000	1.02	40.509	**42.542**	2.033

Figure 8: Event probabilities for data sets generated with different Vasicek parameters.

Main Results

- MQLV tends to the theoretical limit of 50% at a strike of 1
- MQLV able to learn the optimal policy independently of the data sets considered and of the Vasicek parameters

Conclusion

Summary

- MQLV: a model-free and off-policy reinforcement learning approach
- Based on Q-learning and a digital function to simulate the risk of payment default
- Based on the aggregated transactions of the clients

In our experiments,

- MQLV values tend to the theoretical limit of 50% at a strike of 1
- Capable of evaluating an optimal policy of money management

$\longrightarrow$ MQLV allows more customization and more transparency related to loan or credit card applications

Conclusion

Future work

- Create of a fully anonymized data set
- Evaluate the MQLV's performance for data sets that violate the Vasicek assumptions
- Integrate of a deep learning framework for basis function approximator

Questions

Thank you for your attention

Jérémy Charlier

jeremy.charlier@uni.lu

Appendix

Bellman optimality equation

$$Q_t^*(x,a) = \mathbb{E}_t \left[R_t(X_t, a_t, X_{t+1}) + \gamma \max_{a_{t+1} \in \mathcal{A}} Q_{t+1}^*(X_{t+1}, a_{t+1}) | X_t = x, a_t = a \right] \quad (1)$$

Mean reverting Vasicek diffusion process

$$dS_t = \kappa(b - S_t)dt + \sigma dB_t \quad (2)$$

System of equations

$$\begin{cases} S_t = X_t + S_0 e^{-\kappa t} + b(1 - e^{-\kappa t}) \\ \text{with } X_t = \sigma e^{-\kappa t} \int_0^t e^{\kappa s} dB_s - \left[S_0 e^{-\kappa t} + b(1 - e^{-\kappa t}) \right] \end{cases} \quad (3)$$

Appendix

Terminal condition for the backward loop

$$Q_T^*(X_T, a_T = 0) = -\Pi_T - \lambda Var\left[\Pi_T(X_T)\right] \tag{4}$$

Digital function

$$\Pi_T = 1_{S_T \geq K} = \begin{cases} 1 \text{ if } S_T \geq K \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

One-step time dependent random reward

$$\begin{aligned} R_t(X_t, a_t, X_{t+1}) &= \gamma a_t \Delta S_t(X_t, X_{t+1}) - \lambda Var\left[\Pi_t | \mathcal{F}_t\right] \\ \text{with } Var\left[\Pi_t | \mathcal{F}_t\right] &= \gamma^2 \mathbb{E}_t\left[\hat{\Pi}_{t+1}^2 - 2a_t \Delta \hat{S}_t \hat{\Pi}_{t+1} + a_t^2 \Delta \hat{S}_t^2\right] \end{aligned} \tag{6}$$

Appendix

Q-learning update, Q^*, and the optimal action, a^*, to be solved within the backward loop $\forall t = T-1, ..., 0$.

$$\begin{aligned} Q_t^*(X_t, a_t) &= \gamma \mathbb{E}_t \left[Q_{t+1}^*(X_{t+1}, a_{t+1}^*) + a_t \Delta S_t \right] - \lambda Var \left[\Pi_t | \mathcal{F}_t \right] \\ a_t^*(X_t) &= \mathbb{E}_t \left[\Delta \hat{S}_t \hat{\Pi}_{t+1} + \frac{1}{2\lambda\gamma} \Delta S_t \right] \left[\mathbb{E}_t \left[\left(\Delta \hat{S}_t \right)^2 \right] \right]^{-1} \end{aligned} \tag{7}$$

Final set of linear equations

$$\begin{cases} M_n^{(t)} = \displaystyle\sum_{k=1}^{N} \Psi_n(X_t^k, a_t^k) \left[\eta \left(R_t(X_t, a_t, X_{t+1}) + \gamma \max_{a_{t+1} \in \mathcal{A}} Q_{t+1}^*(X_{t+1}, a_{t+1}) \right) \right] \\ \text{with } \eta \sim B(N, p) \end{cases} \tag{8}$$