Personalized Fund Recommendation with Dynamic Utility Learning

Jia Liu

Xi'an Jiaotong University

E-mail: jialiu@xjtu.edu.cn

Joint work with Jiaxin Wei (Xi'an Jiaotong University)

Outline

- Customer-interaction interface
- Preset structure of customer's utility function
- Utility learning based on customer's click sequence
- Fund recommendation with ϵ-greedy algorithm
- Numerical test
- Summary and future developments

Personalized fund recommendation

- provide personalized investment recommendation to investors with various investment goals, risk preferences
- continuous interactions between system and customer improve the customer's viscosity

* Applications: financial investment (Broker, Third party payment); E-commerce (Commercial companies, Business negotiation)

Personalized recommendation system

- **Model-free approach**
  - matching the characteristics of users or investment items
  - collaborative filtering: calculate the similarity (item-based, Sarwar, B. et al., 2001; user-based Zhao, Z.D. and Shang, M.S., 2010)
  - Reinforcement learning improves the fitness of the recommendations (Bourdache, N. et al., 2019; Alsabah, H. et al., 2021; Dong, Zhu, Xu 2022; (Cui, Li, et al., 2022)

- **Model-based approach**
  1) set a modern financial optimization model
  2) learn user's risk preferences in the optimization model through some elicitation methods
  3) provide the most suitable portfolio to the user based on the model with elicited preferences

Personalized recommendation system

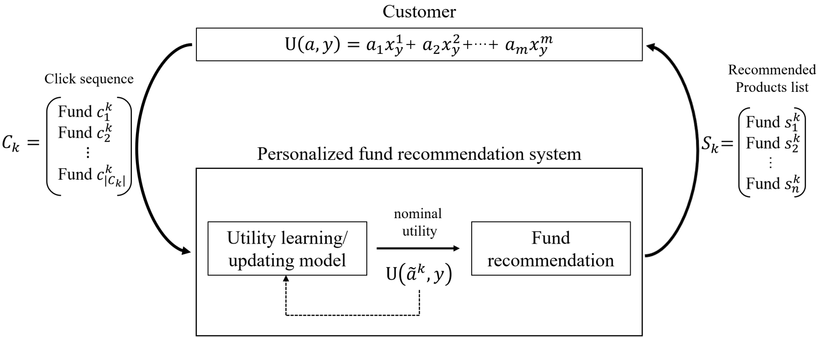


Figure: Interaction mechanism of the personalized fund recommendation system

Preference elicitation

- Mean-Risk model:
  - mean-risk utility models, reinforcement learning risk-aversion parameters (Alsabah et al., 2021)
  - mean-variance model, closed-form updating risk-aversion parameters (Dong, Zhu, Xu 2022)
  - novel dynamic asset allocation framework based on a family of mean-variance-induced utility functions (Cui, Li, et al., 2022)
- Conjoint analysis: apply a multidimensional linear utility function based on some attributes of products
- Random linear utility: assume a random term of certain distribution

Preference elicitation approach

- pairwise comparison (Armbruster and Delage, 2015)
- user's ratings on products (Liu, J. et al., 2021)
- user's historical decisions (Yu, S. et al., 2023)

Why reinforcement learning and incremental learning

- avoid infeasible issue when customer's choice conflict to historical choices
- keep updating: explore new options and preserve previous preference
- take the cost of interaction into consideration

$\Rightarrow$ utility elicitation + recommendation

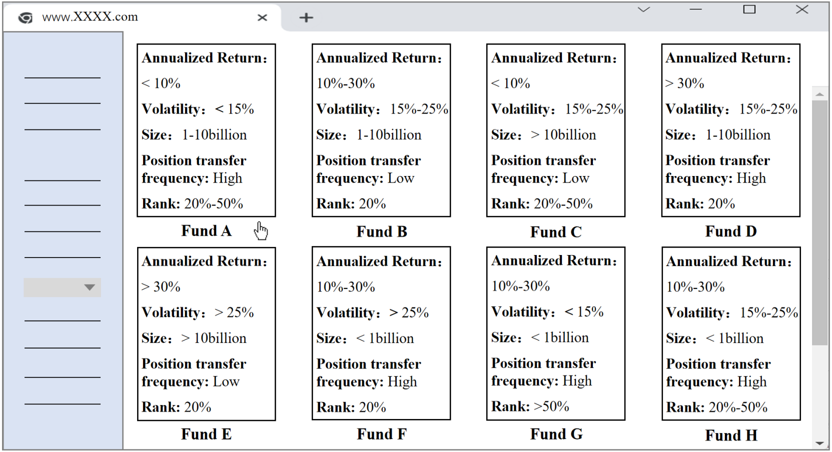
$\Rightarrow$ exploration vs exploitation

$\Rightarrow$ ϵ-greedy + incremental learning

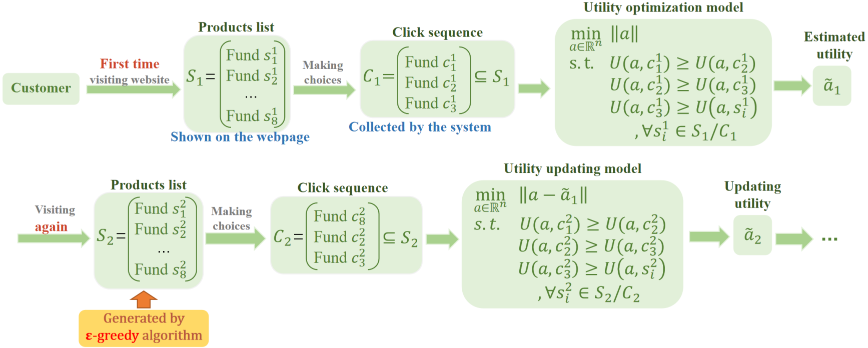
Highlighted contributions:

- Novel click-based interaction mechanism
- Consider the cost of interaction, which brings an exploitation - exploration balance problem.
- A combination of epsilon-greedy algorithm and optimization-based preference updating.

Customer-interaction interface



Customer-interaction interface



Preset structure of customer's utility function

The customer holds a utility function $U(a, y) : \mathbb{R}^m \times Y \to \mathbb{R}$ for each fund y.

We assume that the utility function is linear to the properties x_y of fund y, which is defined as

$$U(a, y) = a_1 x_y^1 + a_2 x_y^2 + \ldots + a_m x_y^m, \tag{1}$$

where $x_y = (x_y^1, \ldots, x_y^m)$ is the relative properties of the fund $y \in Y$, $a = [a_1, \ldots, a_m]$ is the loading factor corresponding to a customer.

Preset structure of customer's utility function

We can define a fund product mainly based on the following five properties.

Table: Five properties of funds

	Properties	Properties Description
x^1	Annualized return	[(Return/Principal)/Invested days] $\times 365 \times 100\%$
x^2	Volatility	The degree of volatility of the underlying price
x^3	Size	Total assets under management of the fund
x^4	Position transfer frequency	The frequency of changes in a fund's portfolio
x^5	Ranking	Fund ranking / Number of funds in the same category $\times$ 100%

Utility learning based on customer's click sequence

For the k-th visit, We denote the products list provided to the customer as $S_k = \left[s_1^k, s_2^k, ..., s_n^k\right]$.
The customer may choose $|C_k|$ funds by clicking them and the system observes a click sequence, denoted by $C_k = \left[c_1^k, c_2^k, \ldots, c_{|C_k|}^k\right]$.
When the system collects the click sequence from the customer for the first time($k = L$),

$$\begin{aligned}
\min_{a} \quad & \|a\|_2^2 && \text{(2a)}\\
\text{s.t.} \quad & U\left(a, c_i^1\right) \geq U\left(a, c_{i+1}^1\right), \quad i = 1, \cdots, |C_1| - 1, && \text{(2b)}\\
& U\left(a, c_{|C_1|}^1\right) \geq U\left(a, s_i^1\right), \quad \forall s_i^1 \in S_1 \backslash C_1, && \text{(2c)}\\
& \sum_{i=1}^{m} a_i = M, && \text{(2d)}\\
& a \in \mathbb{R}^m. && \text{(2e)}
\end{aligned}$$

(2) can be reformulated as a quadratic programming shown in (3), which is denoted as (QP_1).

(QP_1)

$$\min_{a} \quad \|a\|_2^2 \tag{3a}$$

$$\text{s.t.} \quad a_1 x^1_{c^1_i} + a_2 x^2_{c^1_i} + \ldots + a_m x^m_{c^1_i} \geqslant a_1 x^1_{c^1_{i+1}} + a_2 x^2_{c^1_{i+1}} + \ldots + a_m x^m_{c^1_{i+1}}, \quad i = 1, \cdots, |C_1| - 1, \tag{3b}$$

$$a_1 x^1_{c^1_{|C_1|}} + a_2 x^2_{c^1_{|C_1|}} + \ldots + a_m x^m_{c^1_{|C_1|}} \geqslant a_1 x^1_{s^1_i} + a_2 x^2_{s^1_i} + \ldots + a_m x^m_{s^1_i}, \quad \forall s^1_i \in S_1 \backslash C_1, \tag{3c}$$

$$\sum_{i=1}^{m} a_i = M, \tag{3d}$$

$$a \in \mathbb{R}^m. \tag{3e}$$

Utility updating with historical reference utility and new arrive clicks

When $k \geqslant L + 1$,

$$
\begin{aligned}
\min_{a} \quad & \left\| a - \tilde{a}^{k-1} \right\|_2^2 && \text{(4a)} \\
\text{s.t.} \quad & U\left(a, c_i^k\right) \geq U\left(a, c_{i+1}^k\right), \quad i = 1, \cdots, |C_k| - 1, && \text{(4b)} \\
& U\left(a, c_{|C_k|}^k\right) \geq U\left(a, s_i^k\right), \quad \forall s_i^k \in S_k \backslash C_k, && \text{(4c)} \\
& \sum_{i=1}^{m} a_i = M, && \text{(4d)} \\
& a \in \mathbb{R}^m. && \text{(4e)}
\end{aligned}
$$

Utility updating with historical reference utility and new arrive clicks

(4) can be reformulated as a quadratic programming shown in (5), which is denoted as (QP_2).

(QP_2)

$$\min_{a} \quad \left\| a - \tilde{a}^{k-1} \right\|_2^2 \tag{5a}$$

$$\text{s.t.} \quad a_1 x^1_{c^k_i} + a_2 x^2_{c^k_i} + \ldots + a_m x^m_{c^k_i} \geqslant a_1 x^1_{c^k_{i+1}} + a_2 x^2_{c^k_{i+1}} + \ldots + a_m x^m_{c^k_{i+1}}, \quad i = 1, \cdots, |C_k| - 1, \tag{5b}$$

$$a_1 x^1_{c^k_{|C_k|}} + a_2 x^2_{c^k_{|C_k|}} + \ldots + a_m x^m_{c^k_{|C_k|}} \geqslant a_1 x^1_{s^k_i} + a_2 x^2_{s^k_i} + \ldots + a_m x^m_{s^k_i}, \quad \forall s^k_i \in S_k \backslash C_k, \tag{5c}$$

$$\sum_{i=1}^{m} a_i = M, \tag{5d}$$

$$a \in \mathbb{R}^m. \tag{5e}$$

Feasibility assessment for inconsistent user click sequence

Iterative Constraint Elimination Algorithm

- **Initialization:**

$n = 1$, $\Omega = \{\text{all constraints of}\,(\mathrm{QP}_2)\}$

- **Ensure:**

while $\Omega \backslash \cup_{j=1}^{n} \Omega_j \neq \varnothing$:

solve: (Q_n)

$$\min_a \quad \left\| a - \hat{a}^{n-1} \right\|_2$$

$$\text{s.t.} \quad \Omega \backslash \cup_{j=1}^{n} \Omega_j$$

$$a \in \mathbb{R}^m$$

while (Q_n) is infeasible:

remove the last constraint from $\Omega \backslash \cup_{j=1}^{n} \Omega_j$

end while

$\hat{a}^n = \arg(\mathrm{Q}_n)$, $\Omega_n =$ feasible set of (Q_n) , $n = n + 1$

end while

- **Output:** $\hat{a}^{n-1}$

Fund recommendation with ϵ-greedy algorithm

Trade-off:

- exploitation: recommend based on the current utility estimation$\rightarrow$ short-term revenue
- exploration: provide new options randomly$\rightarrow$ long-term revenue

Generating products list

ϵ-greedy algorithm for generating products list

- **Before the first time getting click sequence** $(k \leq L)$,

Randomly generating product list $[s_1^k, ..., s_8^k]$

Reward: $R \leftarrow$ click sequence C_L

Estimated utility: $U(\tilde{a}^L, x_y) \leftarrow$ solving (QP_1)

- **After yielding estimated utility function** $(k \geq L+1)$,

ϵ-greedy algorithm:

$$i \text{ from 1 to 8 : } \quad s_i^k \leftarrow \begin{cases} \operatorname{argmax}_{y \in Y \setminus \{s_1^k, \ldots, s_{i-1}^k\}} U(\tilde{a}^{k-1}, x_y) & \text{with probability } 1-\varepsilon \\ \text{a random fund in } Y \setminus \{s_1^k, \ldots, s_{i-1}^k\} & \text{with probability } \varepsilon \end{cases}$$

Reward: $R \leftarrow$ click sequence C_k

Updated utility: $U(\tilde{a}^k, x_y) \leftarrow$ solving (QP_2)

Fund pool

We select 113 funds in Chinese market and collect their data of five properties on August 26th, 2022 from `https://fund.eastmoney.com/`.

Table: Data of 10 typical selected products in fund pool

Fund code	Annualized return	Volatility	Size(billion)	Position transfer frequency	Ranking
011236	-20.75%	27.89%	2.146	117.79%	85.20%
160418	-8.39%	17.00%	0.233	76.34%	47.80%
010989	-1.28%	28.59%	0.180	6.11%	17.03%
000729	40.37%	28.39%	2.932	386.58%	0.30%
000756	44.26%	27.90%	1.643	479.08%	0.15%
008280	42.25%	40.48%	1.916	217.58%	0.53%
007288	-1.66%	20.84%	0.211	55.62%	22.12%
001938	-16.04%	24.18%	14.969	35.97%	70.70%
001790	-3.40%	36.17%	7.486	126.61%	29.15%
000008	-4.44%	19.61%	2.014	35.21 %	27.42%

Fund pool

Table: Normalized data of 10 products in fund pool

Fund code	Annualized return	Volatility	Size	Position transfer frequency	Ranking
011236	-0.214	0.379	0.031	0.077	0.852
160418	-0.086	0.231	0.003	0.050	0.478
010989	-0.013	0.389	0.003	0.004	0.170
000729	0.416	0.386	0.043	0.252	0.003
000756	0.456	0.379	0.024	0.312	0.001
008280	0.435	0.550	0.028	0.142	0.005
007288	-0.017	0.283	0.003	0.036	0.221
001938	-0.165	0.329	0.218	0.023	0.707
001790	-0.035	0.492	0.109	0.082	0.291
000008	-0.046	0.267	0.029	0.023	0.274

True Utility function of the virtual customer

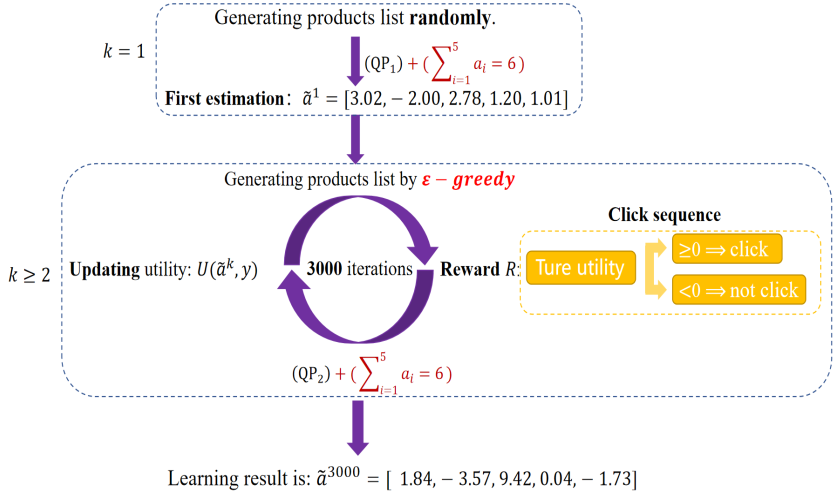
We preset the real utility function as:

$$U(a^*, x_y) = 2x_y^1 - 4x_y^2 + 10x_y^3 + 0x_y^4 - 2x_y^5,$$

i.e. the true value of the loading factor is $a^* = [2, -4, 10, 0, -2]^\top$.

Interaction process simulation

$\epsilon = 0.8$:



Performance of utility estimation

Table: Estimated loading factors after 3000 iterations

Coefficients	$\epsilon = 0.1$	$\epsilon = 0.3$	$\epsilon = 0.5$	$\epsilon = 0.8$
a_1	0.330	0.979	1.757	1.842
a_2	-3.254	-3.385	-3.506	-3.573
a_3	9.984	9.427	9.423	9.420
a_4	0.690	0.672	0.085	0.041
a_5	-1.750	-1.693	-1.758	-1.730

Recommended funds

Table 6: Recommended products list after different interactions by ϵ-greedy algorithm with ϵ=0.3

number of interactions		recommended products list							
10	fund code	161725	100058	162719	001678	002492	550019	008070	000024
	nominal utility	2.319	1.750	1.750	1.461	1.262	1.228	1.225	1.189
	true utility	8.321	0.557	0.114	-1.784	-2.176	-1.427	-1.936	-0.915
500	fund code	161725	161005	006551	009822	100058	001710	011174	519726
	nominal utility	8.18	3.07	1.60	0.64	0.64	0.20	0.12	0.12
	true utility	8.32	2.71	1.67	0.62	0.56	0.23	0.12	0.09
1500	fund code	161725	161005	006551	009822	100058	001710	**000105**	**000027**
	nominal utility	8.181	3.069	1.600	0.641	0.641	0.204	-0.953	-2.244
	true utility	8.321	2.707	1.668	0.615	0.557	0.230	-1.083	-3.026
3000	fund code	161725	161005	006551	009822	100058	001710	011174	519726
	nominal utility	8.181	3.069	1.600	0.641	0.641	0.204	0.117	0.117
	true utility	8.321	2.707	1.668	0.615	0.557	0.230	0.164	0.088

Recommended funds

Table 7: Recommended products list with ϵ=0.8

number of interactions	recommended products list								
10	fund code	161725	162719	519212	004475	**160638**	**007164**	**009379**	**007464**
	nominal utility	3.113	1.905	1.241	1.044	-0.628	-0.265	-0.419	-1.068
	true utility	8.321	0.114	-0.453	-0.082	-1.278	-2.890	-1.140	-3.074
500	fund code	161725	161005	006551	009822	**007288**	**519198**	**000024**	**009637**
	nominal utility	7.21	2.46	1.53	0.57	-1.17	-0.92	-0.70	-0.41
	true utility	8.32	2.71	1.67	0.62	-1.58	-1.42	-0.92	-0.53
1500	fund code	161725	**006061**	**011048**	**011174**	**501048**	**100058**	**012725**	**007077**
	nominal utility	7.722	-0.893	-0.181	0.207	-2.650	0.599	-0.770	-3.116
	true utility	8.321	-1.083	-0.226	0.117	-3.163	0.557	-0.954	-3.694
3000	fund code	161725	161005	006551	**002670**	**008948**	**008070**	**012365**	**005505**
	nominal utility	7.930	2.678	1.613	-1.080	-2.345	-1.681	-1.555	-0.898
	true utility	8.321	2.707	1.668	-1.227	-2.687	-1.936	-1.758	-1.024

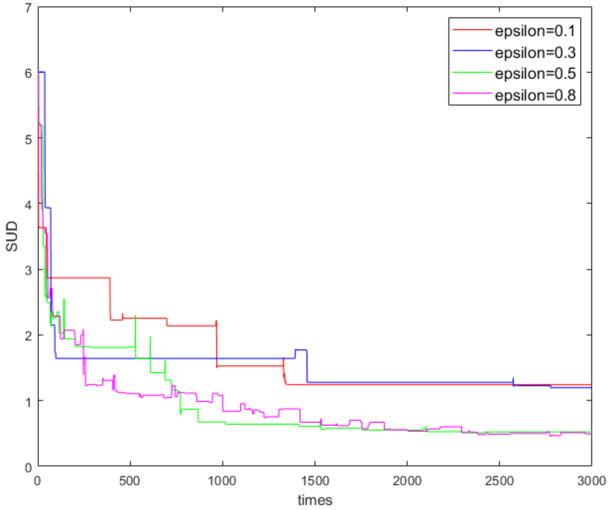
Performance of utility estimation

Definition 1

(Single Utility Difference Upper Bound(SUD)):

$$\mathrm{SUD}(\tilde{a}) = \sup_{y \in Y} \left[U(\tilde{a}, y) - U(a^*, y) \right],$$

Frame Title


epsilon=0.1
epsilon=0.3
epsilon=0.5
epsilon=0.8
SUD
times
0
1
2
3
4
5
6
7
0
500
1000
1500
2000
2500
3000

Cumulative revenue

Definition 2

Recommendation Average Utility Gap Index(RAUGI) :

$$\text{RAUGI}(k) = \frac{1}{8}\sum_{i=1}^{8}\left(1 - \frac{|U(\tilde{a}, s_i^k) - U(a^*, r_i)|}{\max_{y\in Y} U(a^*, y) - \min_{y\in Y} U(a^*, y)}\right) \tag{6}$$

Assuming that the cost of collecting customer preferences is 0.5 per interaction, we consider RAUGI as the system's revenue, the cumulative revenue of the system after the k-th interaction is given by

Definition 3

Cumulative Revenue(CR):

$$\text{CR}(k) = \sum_{n=1}^{k} \text{RAUGI}(n) - 0.5k \tag{7}$$

Cumulative revenue

Figure/compare.png

Cumulative revenue

Figure/compareDetail.png

Average revenue

Average Revenue(AR): $AR(K) = CR(K)/K$.

Table 8: Average revenue

number of interactions	$\epsilon = 0$	$\epsilon = 0.1$	$\epsilon = 0.3$	$\epsilon = 0.5$	$\epsilon = 0.8$	$\epsilon = 1$
10	0.412	0.386	0.388	0.404	0.384	0.363
100	0.415	0.430	0.443	0.435	0.389	0.325
1000	0.415	0.468	0.479	0.441	0.384	0.303
1500	0.415	0.469	0.480	0.439	0.383	0.296
2000	0.415	0.469	0.481	0.439	0.382	0.293
2500	0.415	0.470	0.481	0.438	0.382	0.291
3000	0.415	0.470	0.481	0.438	0.381	0.289

More Virtual customer

Virtual customer $U_1(x_y) = 2x_y^1 - 4x_y^2 + 10x_y^3 + 1x_y^4 - 2x_y^5$.

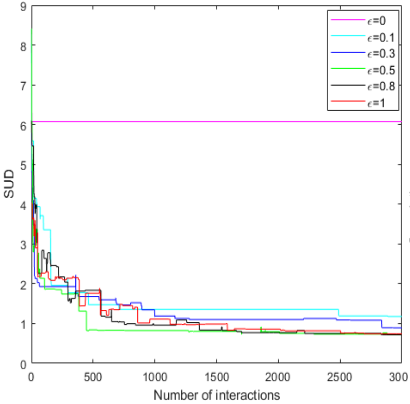


Figure: SUD of the virtual customer 1

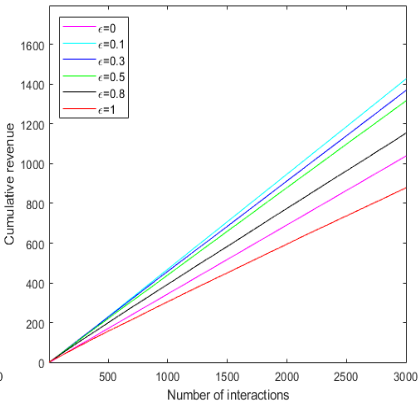


Figure: CR of the virtual customer 1

More Virtual customer

Virtual customer $U_2(x_y) = 2x_y^1 + 4x_y^2 + 10x_y^3 + 1x_y^4 - 2x_y^5$.

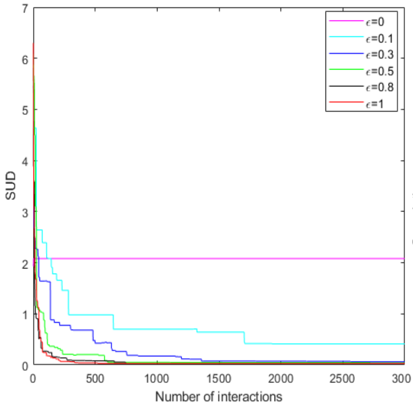


Figure: SUD of the virtual customer 2

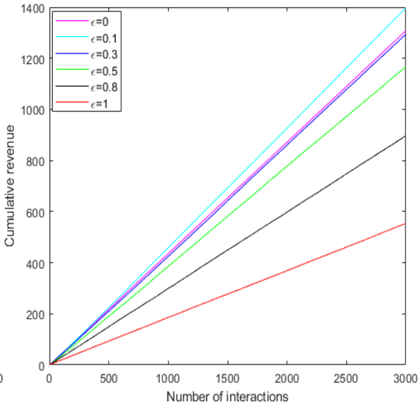


Figure: CR of the virtual customer 2

More Virtual customer

Virtual customer $U_2(x_y) = 2x_y^1 - 4x_y^2 + 10x_y^3 + 1x_y^4 + 2x_y^5$.

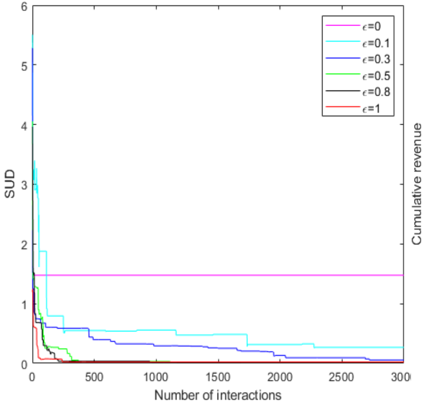


Figure: SUD of the virtual customer 3

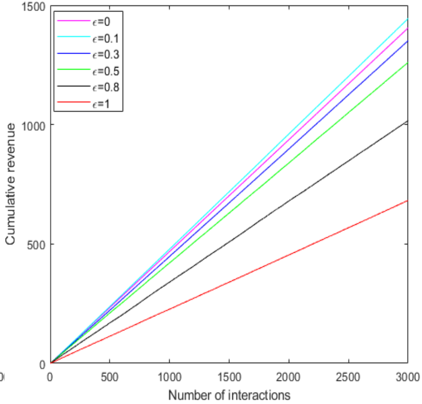


Figure: CR of the virtual customer 3

Conclusions

Summary:

- Design a fund recommendation system based on reinforcement learning framework
- Personalized recommendation by a new utility elicitation approach based on incremental learning
- Take the cost of interaction into account

Limitation:

- fixed ϵ
- linear utility function

Ongoing:

- self-adaptive ϵ selection system
- more refined fund pool to eliminate the insensitivity in utility estimation

Thank you!

Contract: jialiu@xjtu.edu.cn