Контекстуальные бандиты

Ткачев Илья

Описание датасета и задачи

- dualingo сервис

- история показов пуш уведомлений пользователям и соответствующей конверсии в сессию в течение следующих 2 часов

- описание датасета:
    * `*eligible_templates : List[string]*` - список допустимых шаблонов для конкретного уведомления. Каждый шаблон закодирован срезом букв A-L
    * `*ui_language : string*` - ISO 639-1 код языка уведомления, который получил пользователь
    * `*selected_template : string*` - выбранный шаблон
    * `*session_end_completed : boolean*` - была ли конверсия
    * `*datetime : float*` - время отправки уведомления
- выбранный семпл датасета 500 000

Описание датасета и задачи

	datetime	ui_language	eligible_templates	selected_template	session_end_completed
0	0.0	de	[G, E, B, A, K, H, J, L, F, D]	B	True
1	0.0	es	[G, E, B, K, H, J, L, F, D]	F	True
2	0.0	en	[G, E, B, K, H, J, L, F, D]	B	False
3	0.0	zs	[G, E, B, K, H, J, L, F, D]	G	False
4	0.0	vi	[G, E, B, K, H, J, L, F, D]	D	False

Входные данные и обозначения

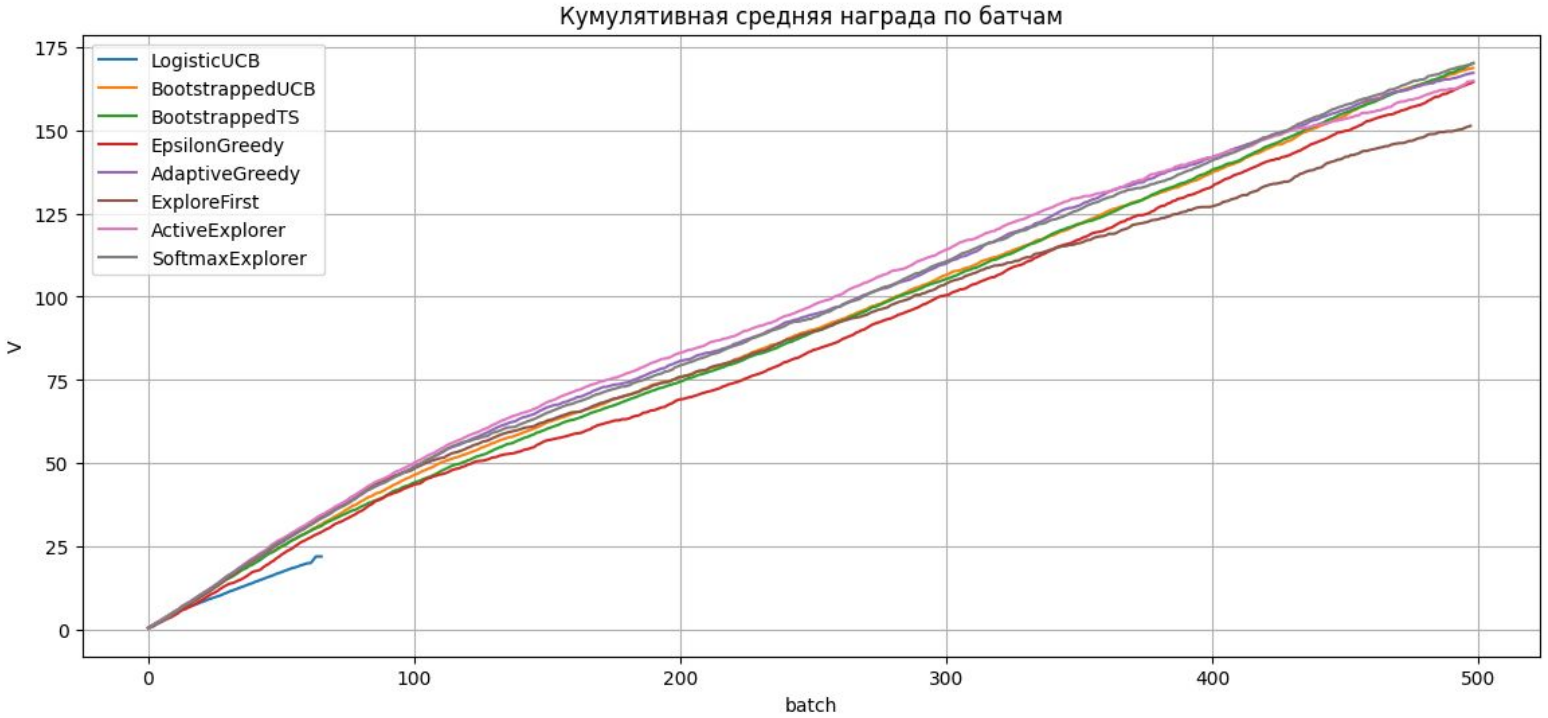
- context - набор ручных признаков, one-hot кодирование eligible_templates, ui_language
- награда - была ли конверсия (1 или 0)
- ручки бандита - 10 видов шаблонов для пуша
- базовый алгоритм для предсказания действия* - LogisticRegression
- проверяемые алгоритмы:
  - BootstrappedUCB
  - BootstrappedTS
  - LogisticUCB *базовый алгоритм не использовался
  - SeparateClassifiers
  - EpsilonGreedy
  - AdaptiveGreedy
  - ExploreFirst,
  - ActiveExplorer
  - SoftmaxExplorer

Алгоритм обучения

Для каждого батча:

- предсказываем лучшее действие на батче
- смотрим в батче, где предсказания совпали с предсказанием агента
- считаем награду по таким семплам
- обучаем алгоритм по таким семплам

Размер батча 1000

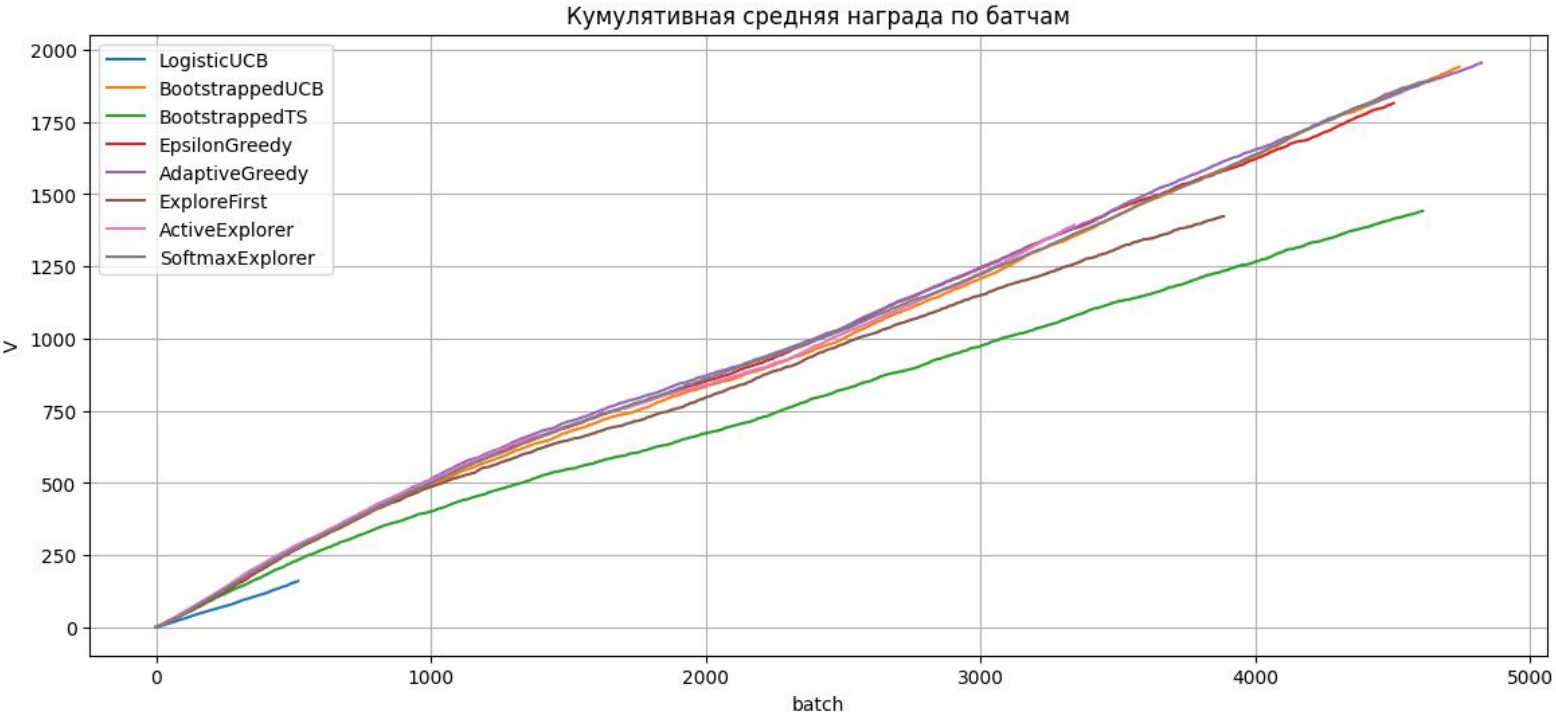


Лучшие:

BootstrappedUCB

SoftmaxExplorer

Размер батча 100

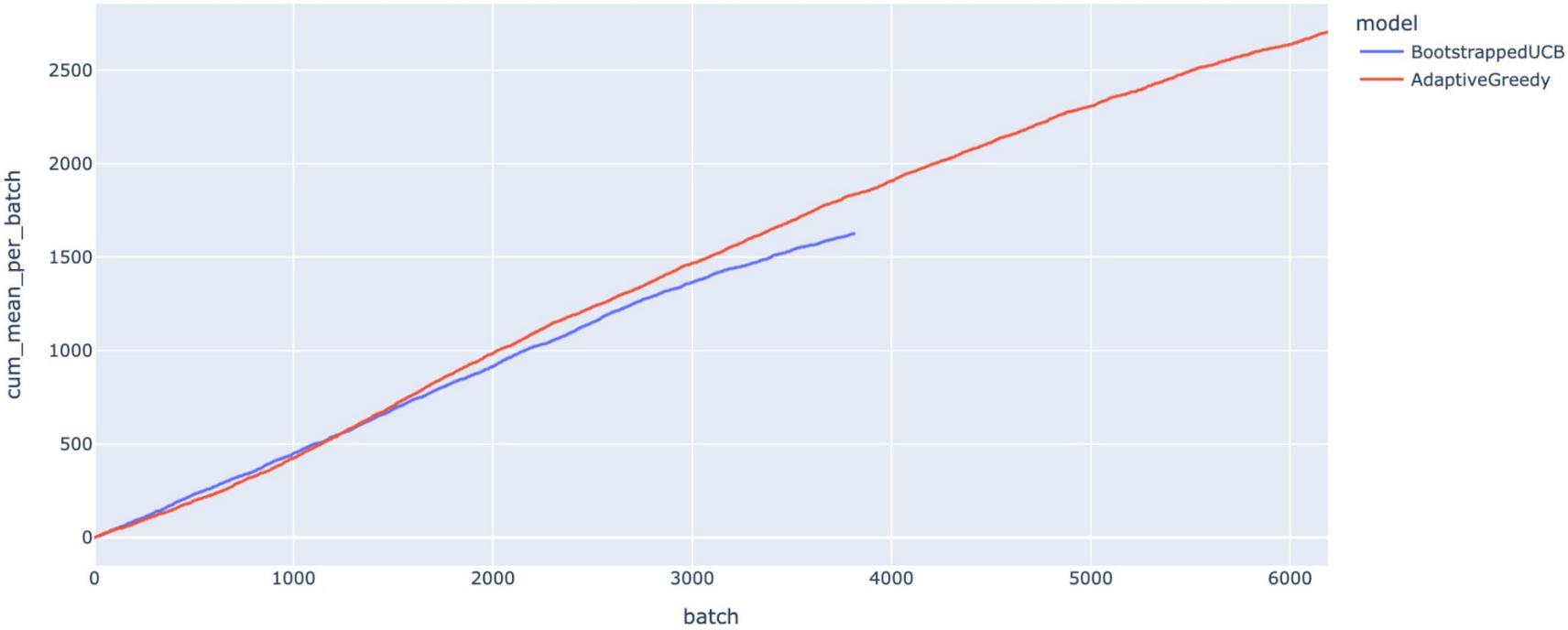


Лучшие:

BootstrappedUCB

AdaptiveGreedy

Размер батча 10



Оптимизация гиперпараметров лучшего алгоритма

Algorithm 12 ActiveAdaptiveGreedy

Inputs window size m, initial threshold z_0, decay rate $d \in (0, 1]$, oracles $\hat{f}_{1:k}$, gradient functions for oracles $g_{1:k}(\mathbf{x}, r)$

for each succesive round t with context $\mathbf{x}^t$ **do**
  if $t = 1$ **then**
    Set $z = z_0$
  Set $\hat{r}_t = \max_k \hat{f}_k(\mathbf{x}^t)$
  if $\hat{r}_t > z$ **then**
    Select action $a = \text{argmax}_k \hat{f}_k(\mathbf{x}^t)$
  else
    for arm q in 1 to k **do**
      Set $z_q = (1 - \hat{f}_q(\mathbf{x}^t))\|g_q(\mathbf{x}^t, 0)\| + \hat{f}_q(\mathbf{x}^t)\|g_q(\mathbf{x}^t, 1)\|$
    Select action $a = \text{argmax}_k z_k$
  if $t \geq m$ **then**
    Update $z := \text{Percentile}_p\{\hat{r}_t, \hat{r}_{t-1}, ..., \hat{r}_{t-m+1}\}$
  Obtain reward r_a^t, Add observation $\{\mathbf{x}^t, r_a^t\}$ to the history for arm a
  Update oracle $\hat{f}_a$ with its new history, along with $\hat{g}_a$

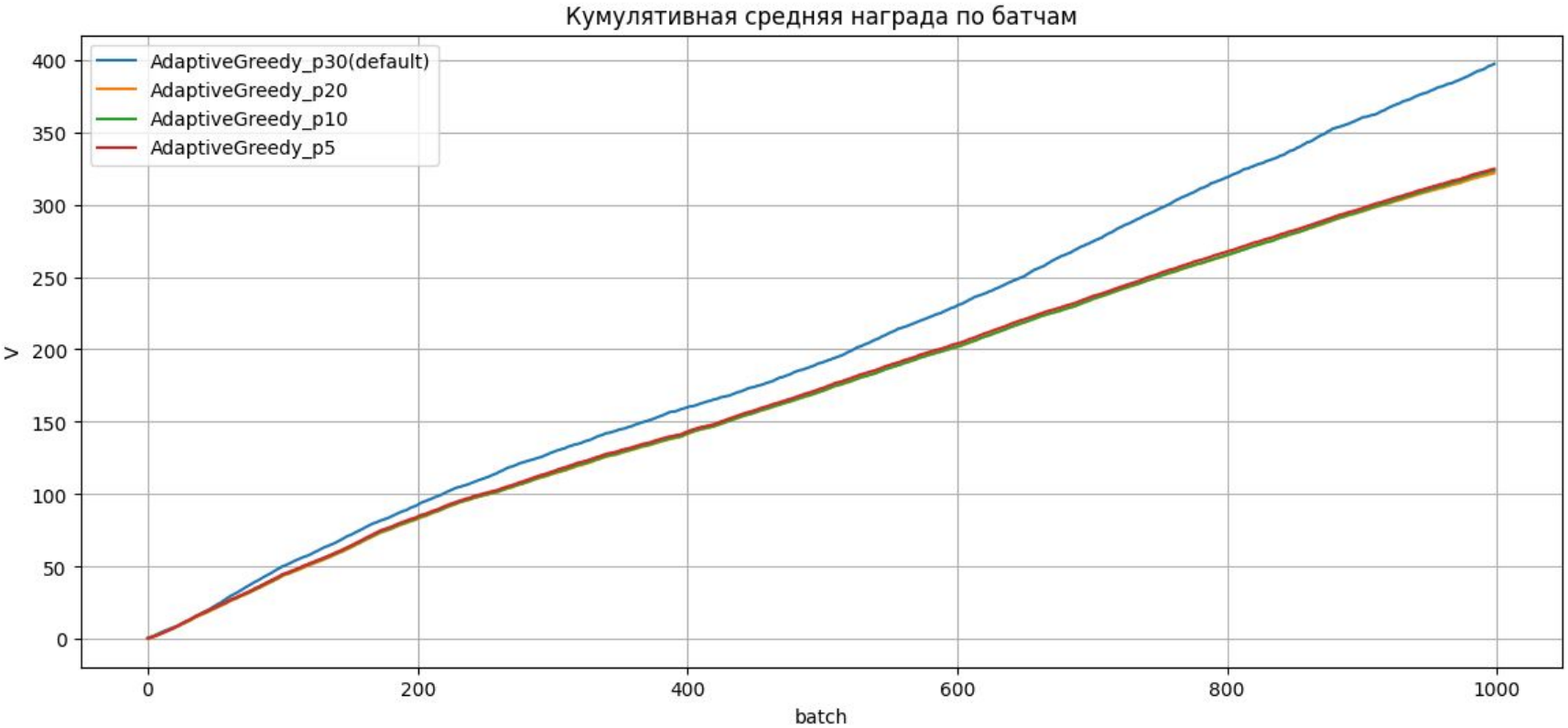
Оптимизация гиперпараметров лучшего алгоритма

Algorithm 12 ActiveAdaptiveGreedy

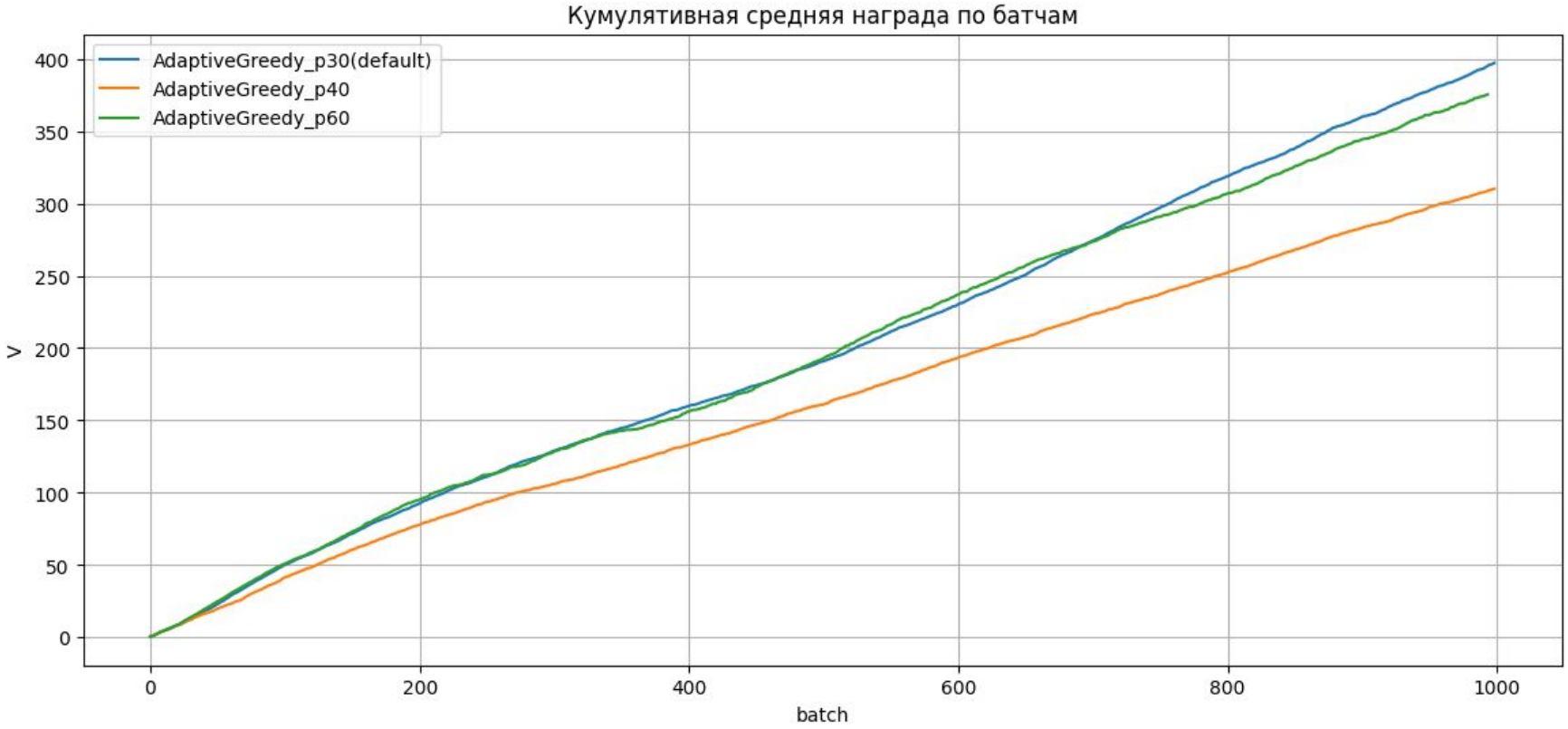
Inputs window size m, initial threshold z_0, decay rate $d \in (0, 1]$, oracles $\hat{f}_{1:k}$, gradient functions for oracles $g_{1:k}(\mathbf{x}, r)$

for each succesive round t with context $\mathbf{x}^t$ **do**
  if $t = 1$ **then**
    Set $z = z_0$
  Set $\hat{r}_t = \max_k \hat{f}_k(\mathbf{x}^t)$
  if $\hat{r}_t > z$ **then**
    Select action $a = \text{argmax}_k \hat{f}_k(\mathbf{x}^t)$
  else
    for arm q in 1 to k **do**
      Set $z_q = (1 - \hat{f}_q(\mathbf{x}^t))\|g_q(\mathbf{x}^t, 0)\| + \hat{f}_q(\mathbf{x}^t)\|g_q(\mathbf{x}^t, 1)\|$
    Select action $a = \text{argmax}_k z_k$
  if $t \geq m$ **then**
    Update $z := \text{Percentile}_p\{\hat{r}_t, \hat{r}_{t-1}, ..., \hat{r}_{t-m+1}\}$
  Obtain reward r_a^t, Add observation $\{\mathbf{x}^t, r_a^t\}$ to the history for arm a
  Update oracle $\hat{f}_a$ with its new history, along with $\hat{g}_a$

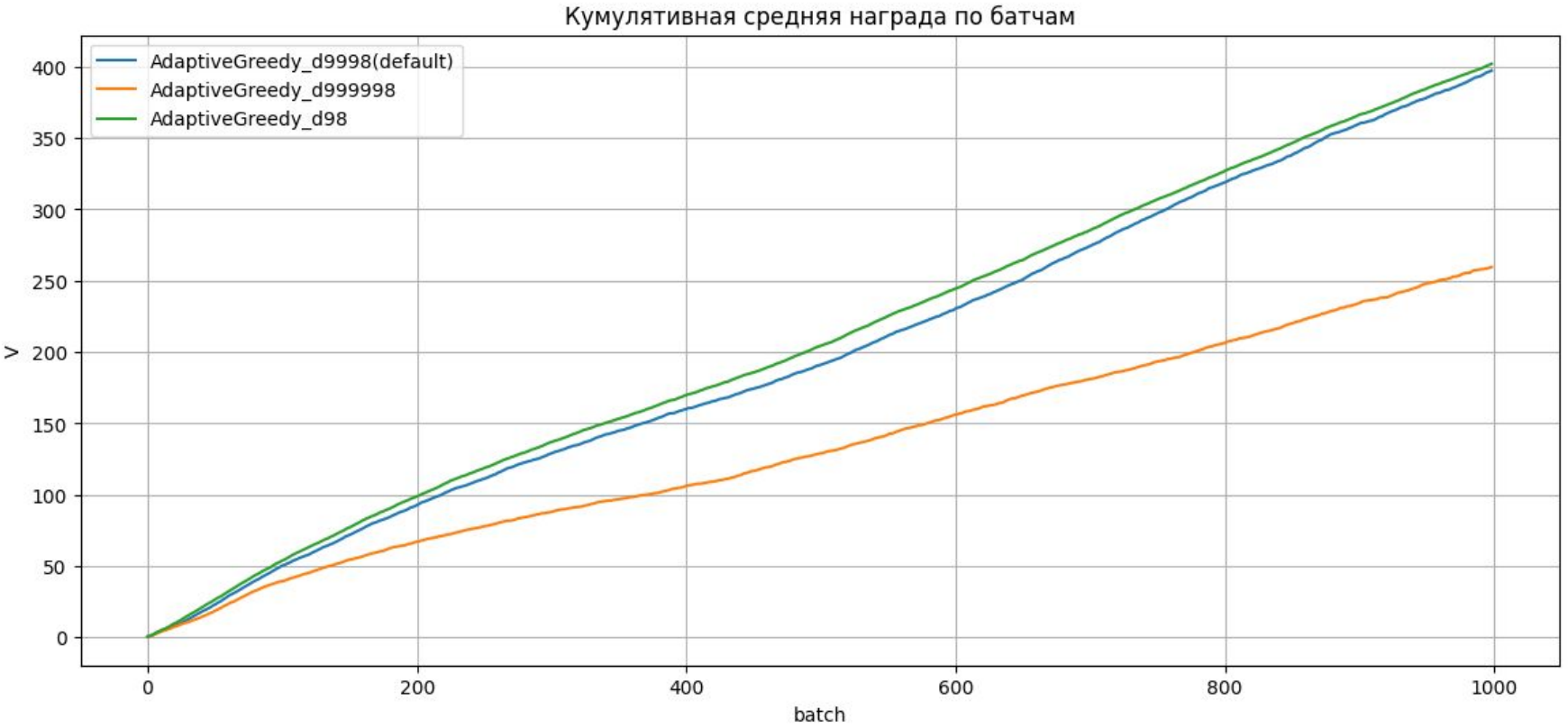
Оптимизация percentile



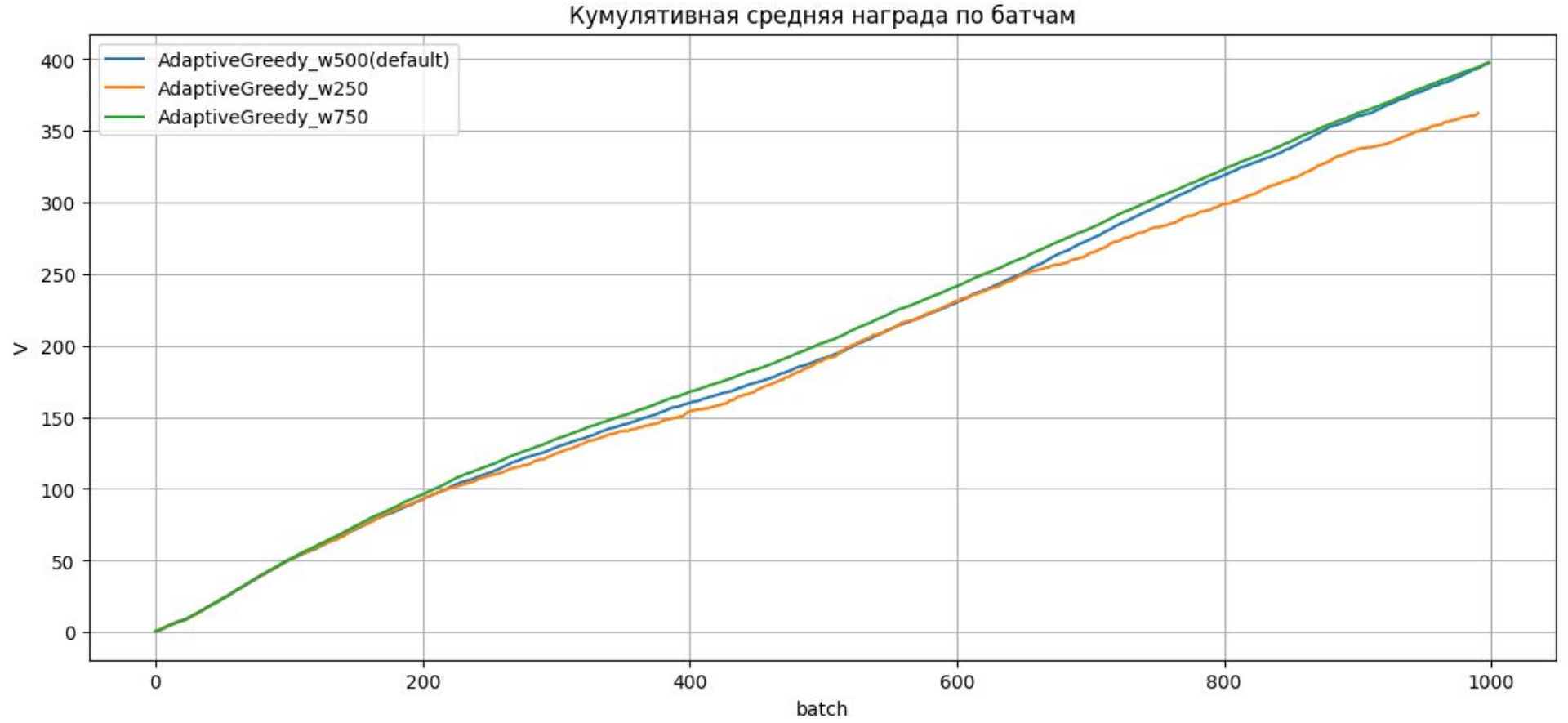
Оптимизация percentile



Оптимизация decay



Оптимизация window_size



Выводы

- Алгоритмы AdaptiveGreedy и BootstrappedUCB является самым эффективным “из коробки”, показывая наибольшую эффективность

- Размер батча имеет критически важный момент для максимизации награды, особенно это видно на примере AdaptiveGreedy, но маленький батч заметно сказывается на скорости обучения

- AdaptiveGreedy обладает очень высокой степенью адаптации, поэтому очень эффективен на маленьких итерациях обучения