User Models and Interactive IR

ESSIR 2022

Romain Deffayet[1,2] and **Maarten de Rijke**[2]
[1]Naver Labs Europe, [2]University of Amsterdam

July 19, 2022, 09.00–10.30

r.e.deffayet@uva.nl, m.derijke@uva.nl

Romain Deffayet
PhD student at
Naver Labs Europe and the
University of Amsterdam

Maarten de Rijke
University professor at the
University of Amsterdam

Based on joint work and conversations with Ali Vardasbi, Jean-Michel Renders, Maria Heuss, Harrie Oosterhuis, Shashank Gupta

Materials based in part on (Chuklin et al., 2015; Oosterhuis and de Rijke, 2018; Oosterhuis et al., 2020; Saito and Joachims, 2021)

Learning objectives and outcomes

Objectives

- We will cover basic concepts and fundamental methods of learning from interactions for search and recommendation

Outcomes

- As a result of participating in this tutorial, students will be able to implement and work with basic (counterfactual) learning to rank, bandits and reinforcement learning for search and recommendation

Resources

Go to https://irlab.science.uva.nl/2022/07/17/essir-2022-tutorial-on-user-models-and-interactive-ir/

- Slides (PDF)
- Bibliography (BIB)

Agenda

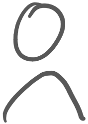
09.00	Start
09.00–09.05	Domestic matters – Maarten and Romain
09.05–09.20	Setting the scene – Maarten
09.20–09.45	Counterfactual learning-to-rank – Maarten
09.45–10.15	Bandits & Reinforcement learning in IR – Romain
10.15–10.25	Conclusion – Maarten and Romain
10.25–10.30	Final Q&A
10.30	End

Setting the scene

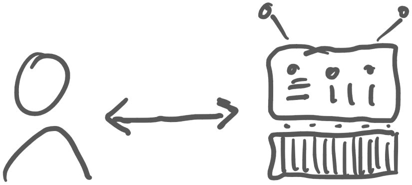
Plan for this part

- Interactions with users – our perspective on information retrieval: technology to connect people to information
- Core concepts and examples
- Core problems: learning and evaluating
- Core distinctions: on-policy vs off-policy

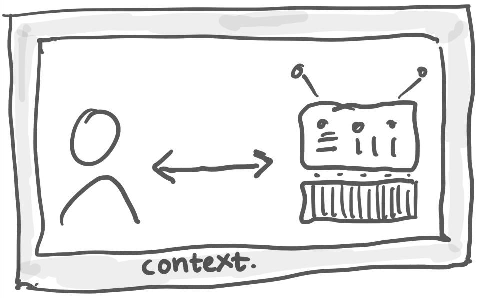
Interactions with users



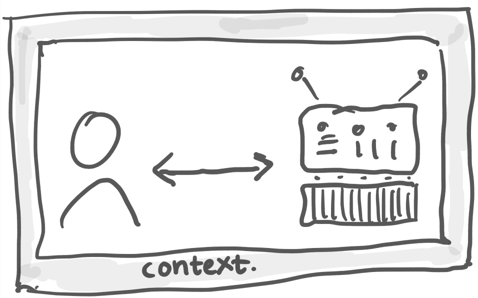
Interactions with users



Interactions with users

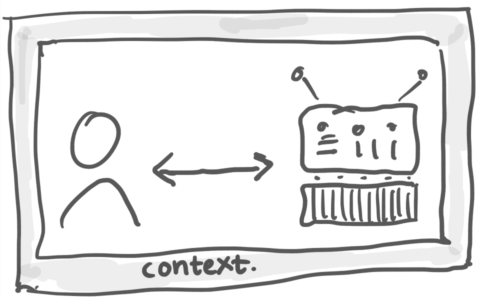


Interactions with users



How can a search engine or recommender system or conversational assistant get better by interacting with its users?

Interactions with users



How can a search engine or recommender system or conversational assistant get better by interacting with its users?

- Context x – user history, user profile, query, time of day, . . .
- Policy π that selects action a – answer, item, result list, . . .
- Rewards r that are returned – clicks, downloads, purchases, . . .

Interaction process

Policy can be . . .

- deterministic – function from contexts to actions: $\pi(x) = a$
- stochastic – conditional probability of action given context: $\pi(a \mid x)$

Policy interacts with environment and produces **log data** $\{(x_i, a_i, r_i)\}_{i=1}^{n}$

- Observe context x – $x \sim P(x)$
- Select action a – $a \sim \pi(a \mid x)$
- Observe reward r – $p(r \mid x, a)$

Examples

- Ad hoc search (not personalized): context – query; action – ranked list of documents; reward – clicks, dwell time
- Product search (personalized): context – query, user profile, past interactions; action – grid of items; reward – clicks, conversion
- Ad placement (personalized): context – user profile; action – a slate of ads; reward – clicks, conversion
- Conversational recommendation: context – user history, conversation history; action – item; reward – task completion, conversion

Two core problems

Policy learning

- Find a new policy that improves upon the current policy

Policy evaluation

- Determine the quality – often expressed as the (online) performance – of a given policy

Two key distinctions (1)

In CS, algorithms that receive input sequentially operate in online modality

- Typically includes tasks that involve sequences of decisions, like when you choose how to serve incoming queries in a stream

Two key distinctions (1)

In CS, algorithms that receive input sequentially operate in online modality

- Typically includes tasks that involve sequences of decisions, like when you choose how to serve incoming queries in a stream

Batch or **offline** processing does not need human interaction

- E.g., batch learning proceeds as follows:
  - Initialize the weights
  - Repeat the following steps: (Process all the training data; Update the weights)

Two key distinctions (1)

In CS, algorithms that receive input sequentially operate in online modality

- Typically includes tasks that involve sequences of decisions, like when you choose how to serve incoming queries in a stream

Batch or **offline** processing does not need human interaction

- E.g., batch learning proceeds as follows:
  - Initialize the weights
  - Repeat the following steps: (Process all the training data; Update the weights)

Typical offline computations in information retrieval:

- Any processing that is not query dependent (crawling, indexing, . . .)

Two key distinctions (1)

In CS, algorithms that receive input sequentially operate in online modality

- Typically includes tasks that involve sequences of decisions, like when you choose how to serve incoming queries in a stream

Batch or **offline** processing does not need human interaction

- E.g., batch learning proceeds as follows:
  - Initialize the weights
  - Repeat the following steps: (Process all the training data; Update the weights)

Typical offline computations in information retrieval:

- Any processing that is not query dependent (crawling, indexing, ...)

Typical online computations in information retrieval:

- Any processing that depends on users and their input

Two key distinctions (2)

Evaluation and learning can be **on-policy** or **off-policy**

Two key distinctions (2)

Evaluation and learning can be **on-policy** or **off-policy**

On-policy learning algorithms evaluate and improve the same policy that is being used to select actions

Two key distinctions (2)

Evaluation and learning can be **on-policy** or **off-policy**

On-policy learning algorithms evaluate and improve the same policy that is being used to select actions

Off-policy learning algorithms evaluate and improve a policy that is different from the policy that is used for action selection

- Behavior or logging policy: policy that tells the agent what action to take; used to collect actions taken and outcomes; not the target policy in off-policy learning

Online policy evaluation: A/B testing

Deploy two policies π_A and π_B to get an online estimate of performance

- Collect log data $\mathcal{D}_A = \{(x_i, a_i, r_i)\}_{i=1}^{n}$ and $\mathcal{D}_B = \{(x_j, a_j, r_j)\}_{j=1}^{m}$
- Compute quality as average reward: $\frac{1}{n}\sum_{i=1}^{n}\{r_i : (x_i, a_i, r_i) \in \mathcal{D}_A\}$ for π_A and $\frac{1}{m}\sum_{j=1}^{m}\{r_j : (x_j, a_j, r_j) \in \mathcal{D}_B\}$ for π_B
- Compare the two average rewards

Online policy evaluation: Interleaving

Again, take two policies π_A and π_B but now

- Given context x, determine most probably actions a_A and a_B
- Combine actions a_A and a_B into action $a_{A \oplus B}$ and determine credit assignment
- Execute combined action $a_{A \oplus B}$, observe reward, following credit assignment function assign credit to π_A or π_B (or both or neither)
- Repeat, take average, and compare

Online policy evaluation: Interleaving

Again, take two policies π_A and π_B but now

- Given context x, determine most probably actions a_A and a_B
- Combine actions a_A and a_B into action $a_{A \oplus B}$ and determine credit assignment
- Execute combined action $a_{A \oplus B}$, observe reward, following credit assignment function assign credit to π_A or π_B (or both or neither)
- Repeat, take average, and compare

Several design choices

- How to combine, how to satisfy constraints on actions, how to assign credit, ...

From online evaluation to off-policy evaluation

Why evaluate online?

- User behavior is indicative of their preferences

From online evaluation to off-policy evaluation

Why evaluate online?

- User behavior is indicative of their preferences

Why not to evaluate online?

- Online agents take risks to gain knowledge quickly
- Online evaluations are complex

From online evaluation to off-policy evaluation

Why evaluate online?

- User behavior is indicative of their preferences

Why not to evaluate online?

- Online agents take risks to gain knowledge quickly
- Online evaluations are complex

What if we evaluate off-policy?

- Estimate performance of a policy using only log data collected by a behavior policy
- Compare performance of candidate policies safely and helps us decide which policy should be deployed

Words, words, words

- Counterfactual evaluation = offline A/B testing = off-policy evaluation
- Counterfactual learning = unbiased learning to rank = off-policy learning

Q&A

Questions, comments, . . .

Agenda

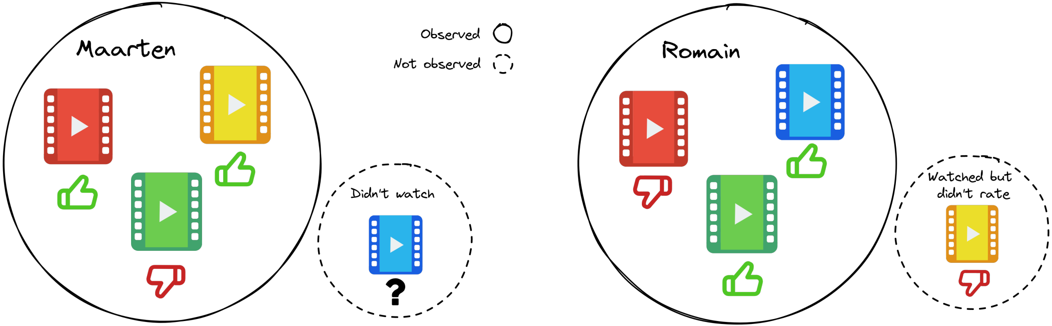
09.00	Start
09.00–09.05	Domestic matters – Maarten and Romain
09.05–09.20	Setting the scene – Maarten
09.20–09.45	Counterfactual learning-to-rank – Maarten
09.45–10.15	Bandits & Reinforcement learning in IR – Romain
10.15–10.25	Conclusion – Maarten and Romain
10.25–10.30	Final Q&A
10.30	End

Counterfactual learning-to-rank

Plan for this part

- Asking counterfactual questions: "what would have been . . ."
- Importance sampling
- Learning from logs
- Bias, and correcting for bias

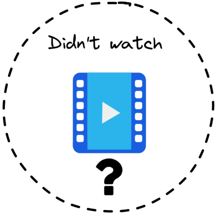
Movie recommendation from ratings



What to recommend next?

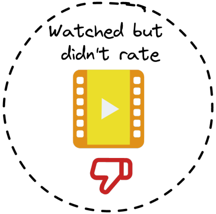
Extrinsic biases in Missing-Not-At-Random (MNAR) feedback

Popularity bias



More popular items are rated more often.

Positivity bias



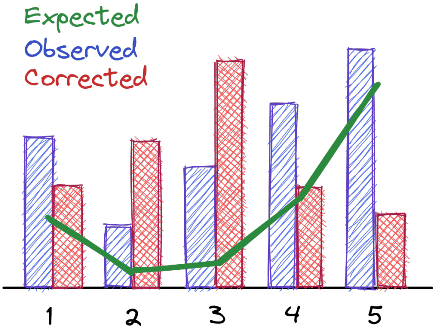
Users are more likely to rate movies they have liked.

Importance sampling

- Estimate a certain quantity θ (e.g., mean, variance) of one distribution by sampling from another.
- Here we want $\mathbb{E}_{(u,m)\sim\mathcal{U}}[r(u,m)]$ but we only observe biased instances $(u^d, m^d) \sim \mathcal{D} \ldots$
- **Counterfactual question**: what would have been the average ratings under a uniform distribution?

Importance sampling

IS correction: In the data, what was the probability of observing (u^d, m^d)? (Wasserman, 2004)



$$\mathbb{E}_{(u^d,m^d)\sim\mathcal{D}}\left[\frac{r(u^d,m^d)}{p^d(u^d,m^d)}\right] = \sum_{(u,m)} \frac{p^d(u,m)}{p^d(u,m)} r(u,m) = \mathbb{E}_{(u,m)\sim\mathcal{U}}\left[r(u,m)\right]$$

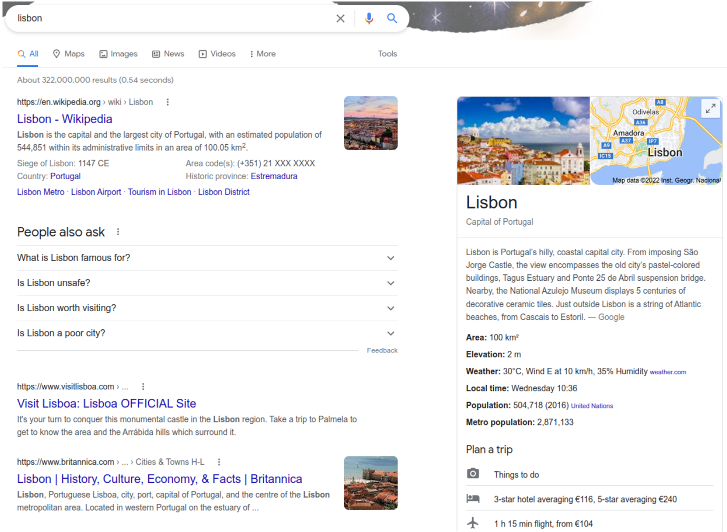
Variance and extensions

Formal definition of the counterfactual estimator:

$$\tilde{r} = \frac{1}{|\mathcal{D}|} \sum_{(u^d, m^d) \in \mathcal{D}} \frac{r(u^d, m^d)}{p^d(u^d, m^d)}$$

- $\mathbb{E}\left[\tilde{r}\right] = \mathbb{E}_{(u,m)\sim\mathcal{U}}\left[r(u, m)\right]$: $\tilde{r}$ is unbiased
- **$\tilde{\mathbf{r}}$ can have high variance if $\mathbf{p^d}$ are small**
- Tricks to decrease the variance: clipping/normalization (Saito and Joachims, 2021)

Learning from search logs: a screenshot



Learning from search logs: a snapshot

AOL-user-ct-collection — less — 116×53

```
710766  wwwpeoplesearch.comwww.reviewplace.seardh        2006-05-30 22:10:13
710766  wwwpeoplesearch.comwww.reviewplace.seardh        2006-05-30 22:10:33
711391  can not sleep with snoring husband       2006-03-01 01:24:00
711391  cannot sleep with snoring husband        2006-03-01 01:24:07     9       http://www.wjla.com
711391  cannot sleep with snoring husband        2006-03-01 01:24:07     9       http://www.wjla.com
711391  cannot sleep with snoring husband        2006-03-01 01:33:06     1       http://www.epinions.com
711391  jackie zeaman nude       2006-03-01 15:26:27
711391  jackie zeman nude        2006-03-01 15:26:38
711391  strange cosmos  2006-03-01 16:07:15      1       http://www.strangecosmos.com
711391  mansfield first assembly        2006-03-01 16:09:20     1       http://www.mansfieldfirstassembly.org
711391  mansfield first assembly        2006-03-01 16:09:20     3       http://netministries.org
711391  reverend harry myers     2006-03-01 16:10:07
711391  reverend harry myers     2006-03-01 16:10:30
711391  national enquirer        2006-03-01 17:13:14     1       http://www.nationalenquirer.com
711391  how to kill mockingbirds        2006-03-01 17:18:11
711391  how to kill mockingbirds        2006-03-01 17:18:33
711391  how to kill annoying birds in your yards        2006-03-01 17:18:58
711391  how to kill annoying birds in your yards        2006-03-01 17:19:53     2       http://www.sortprice.com
711391  how to rid your yard of noisy annoying birds    2006-03-01 17:23:08     3       http://shopping.msn.com
711391  how to rid your yard of noisy annoying birds    2006-03-01 17:23:08     10      http://www.bergen.org
711391  how to rid your yard of noisy annoying birds    2006-03-01 17:24:35     15      http://www.saferbrand.com
711391  how do i get mocking birds out of my yard       2006-03-01 17:27:17
711391  how do i get mockingbirds out of my yard        2006-03-01 17:27:36     9       http://www.asri.org
711391  how do i get mockingbirds out of my yard        2006-03-01 17:30:14
711391  how to get rid of noisy loud birds      2006-03-01 17:30:52     3       http://www.bird-x.com
711391  how to get rid of noisy loud birds      2006-03-01 17:30:52     1       http://forums2.gardenweb.com
711391  how to get rid of noisy loud birds      2006-03-01 17:30:52     10      http://www.birding.com
711391  mansfield first assembly        2006-03-01 18:31:36     3       http://netministries.org
711391  beth moore      2006-03-01 19:42:41     1       http://www.lproof.org
711391  judy baker ministries    2006-03-01 19:49:03     2       http://www.embracinggrace.com
711391  god will fulfill your hearts desires    2006-03-01 19:59:06     10      http://www.pureintimacy.org
711391  online friendships can be very special  2006-03-01 23:09:37
711391  online friendships can be very special  2006-03-01 23:09:57
711391  online friendships       2006-03-01 23:10:24
711391  cypress fairbanks isd    2006-03-02 07:56:53     1       http://www.cfisd.net
711391  people are not always how they seem over the internet   2006-03-02 08:31:51
711391  friends online can be different in person       2006-03-02 08:32:42
711391  friends online can be different in person       2006-03-02 08:33:04     13      http://www.salon.com
711391  boston butts    2006-03-02 09:47:36
711391  community christian church houston tx   2006-03-02 16:07:53
711391  gay churches in houston tx      2006-03-02 16:08:23
711391  community gospel church in houston tx   2006-03-02 16:08:45     2       http://www.communitygospel.org
711391  houston tx is one hot place     2006-03-02 18:04:44
711391  houston tx is one hot place to live     2006-03-02 18:04:55     9       http://travel.yahoo.com
711391  houston tx is one hot place to live     2006-03-02 18:16:05     1       http://www.houston-texas-online.com
711391  texas hill country and sights around san antonio tx     2006-03-02 18:19:00     5       http://www.answers.c
om
711391  can liver problems cause you to loose your hair 2006-03-02 18:27:04
711391  can liver problems cause you to loose your hair 2006-03-02 18:27:30     1       http://www.askdoctrish.com
711391  strange cosmos  2006-03-02 19:29:31      1       http://www.strangecosmos.com
711391  white hard dry skin on face     2006-03-02 20:31:29
711391  white hard dry skin on face     2006-03-02 20:32:24
:
```

Intrinsic biases in click logs

- **Position bias**: users are more likely to observe item on top of the page
- **Item-selection bias**: users cannot observe items which are not returned by the earch engine
- **Trust bias**: users may trust the search engine to return relevant results and are therefore more likely to click on top documents
- ...

Importance sampling on slates

Logging policy π_L : $\pi_L(\mathbf{s} \mid u)$ probability that the system chooses slate s for user u.

Under the logging policy, what was the probability of . . .

. . . that slate being returned? (Precup et al., 2000)

$$R^{\mathrm{IS}} = \sum_{(u,\mathbf{s},\mathbf{r})\in\mathcal{D}} \frac{1}{\underbrace{\pi_L(\mathbf{s} \mid u)}_{\text{too small !}}} \cdot \sum_{j=1}^{k} r^j$$

Importance sampling on slates

Logging policy π_L : $\pi_L(\mathbf{s} \mid u)$ probability that the system chooses slate s for user u.

Under the logging policy, what was the probability of . . .

. . . that document being placed at that position in the slate? (McInerney et al., 2020)

$$R^{\text{pos-IS}} = \sum_{(u,\mathbf{s},\mathbf{r})\in\mathcal{D}} \sum_{j=1}^{k} \frac{1}{\underbrace{\pi_L^j(s_j \mid u)}_{\text{better but still small}}} \cdot r^j$$

Leveraging user models with inverse propensity scoring (IPS)

Examination hypothesis: A clicked document is both **examined** and **relevant** .

Under the logging policy, what was the probability of ...

... the user examining that document? (Joachims et al., 2017)

$$R^{\mathrm{IPS}} = \sum_{(u,\mathbf{s},\mathbf{r})\in\mathcal{D}} \sum_{d\in\mathbf{s}} \frac{1}{\underbrace{P(E_d = 1 \mid u)}_{\text{examination prob.}}} \cdot r^d$$

For example, position-based model: $P(E_d = 1 \mid u, \mathrm{rank}(d) = j) = \gamma_j$.

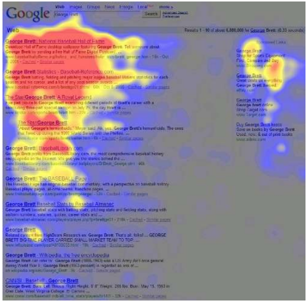
Improvements to handle item-selection and trust bias : policy-aware and affine estimators (Oosterhuis and de Rijke, 2021).

Doubly robust estimator (Saito and Joachims, 2021)

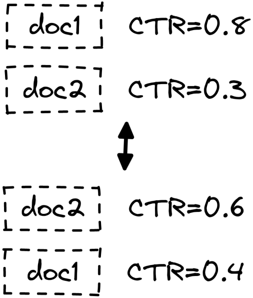
- Direct (biased) estimate of the user response: $\tilde{r}(d|u)$
- Simple idea: get a rough, biased estimate of user response and apply the correction only on the difference with this estimate (less variance-related risk)

$$R^{\mathrm{DRE}} = \sum_{(u,\mathbf{s},\mathbf{r})\in\mathcal{D}} \sum_{d\in\mathbf{s}} \underbrace{\left[\underbrace{\tilde{r}(d|u)}_{\text{high-bias, low-variance}} + \underbrace{\frac{1}{P(E_d = 1|u)} \cdot (r^d}_{\text{unbiased, high-variance}} - \tilde{r}(d|u)) \right]}_{\text{unbiased, low-variance}}$$

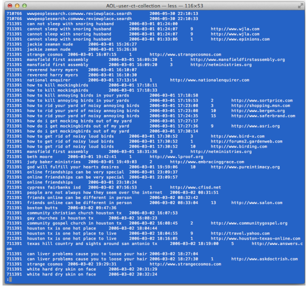
Fitting importance weights: 3 solutions



Eye tracking experiments
(Joachims et al., 2007)



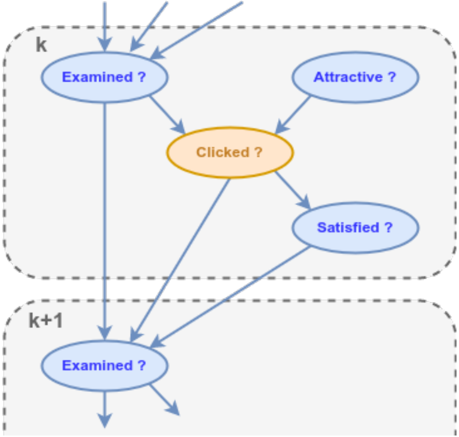
Swap interventions
(Joachims et al., 2017)



Learning from logs:
Click models

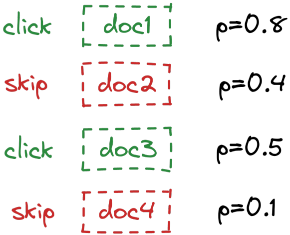
Click models (Chuklin et al., 2015)

Example: Dynamic Bayesian Networks



- Interpretable structure with latent variables and parameterized causal relations
- Learn parameters by Expectation-Maximization or Gradient Descent

Evaluation of click models



Perplexity:

$$\mathrm{PPL@}j = 2^{-\sum_{i=1}^{n} c_j^i \log_2(\tilde{p}_j^i) + (1-c_j^i)\log_2(1-\tilde{p}_j^i)}$$

$$\mathrm{PPL} = \frac{1}{k}\sum_{j=1}^{k} \mathrm{PPL@}j \approx 1.51$$

Click prediction

$$\mathrm{DCG} = \sum_{j=1}^{4} \frac{2^{\mathrm{rel(d_j)}}-1}{\log_2(j+1)}$$

$$\mathrm{nDCG} = \frac{\mathrm{DCG(CM)}}{\mathrm{iDCG}} \approx 0.62$$

Relevance estimation

Summary

- We want to learn from user feedback but **data contains biases**.
- Importance Sampling applies a **correction to observed feedback** to recover unbiased estimates.
- IS is unbiased but suffers from **high variance: we need to leverage user models** (inverse propensity scoring, doubly robust estimator)
- Propensity weights must be **adequately computed and evaluated** (eye tracking, swap interventions, click models)

Q&A

Questions, comments, . . .

Agenda

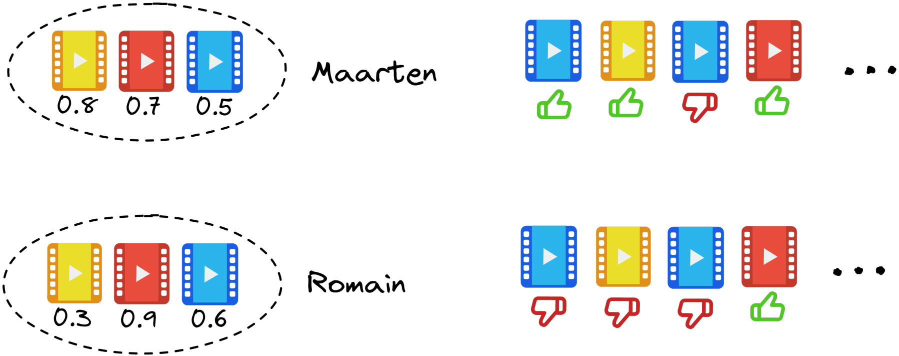
09.00	Start
09.00–09.05	Domestic matters – Maarten and Romain
09.05–09.20	Setting the scene – Maarten
09.20–09.45	Counterfactual learning-to-rank – Maarten
09.45–10.15	Bandits & Reinforcement learning in IR – Romain
10.15–10.25	Conclusion – Maarten and Romain
10.25–10.30	Final Q&A
10.30	End

Bandits & Reinforcement learning in IR

Plan for this part

- Learning to interact with new users
- More complex feedback loops
- Long-term satisfaction
- Learning from logs

Cold start



How to quickly find Maarten's and Romain's preferences ?

Reward and regret

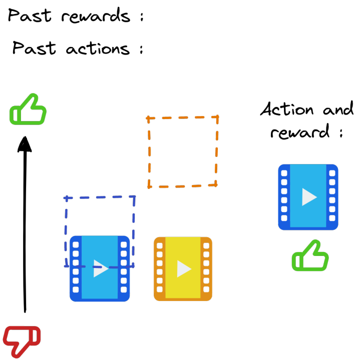
For every action a in the set $\mathcal{A}$, we define:

- the **reward** $r(a) \in \{0, 1\}$: like or dislike,
- the **regret** $\overline{r}(a) = r(a^*) - r(a)$ with $a^* = \arg\max_{a \in \mathcal{A}} \mathbb{E}\left[r(a)\right]$.

The goal can now be formulated as finding a strategy minimizing the **expected cumulative regret** $\overline{R}$:

$$\overline{R}(T) = \mathbb{E}\left[\sum_{t=1}^{T} \overline{r}(a_t)\right]$$

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



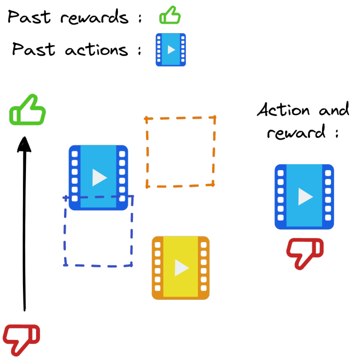
Greedy

Exploration-Exploitation dilemma:

exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



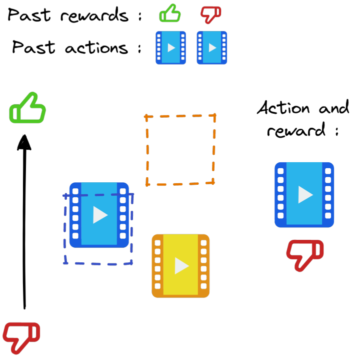
Greedy

Exploration-Exploitation dilemma:

exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



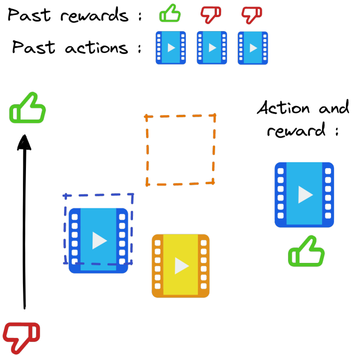
Greedy

Exploration-Exploitation dilemma:

exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



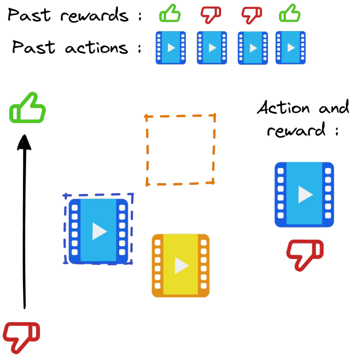
Greedy

Exploration-Exploitation dilemma:

exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



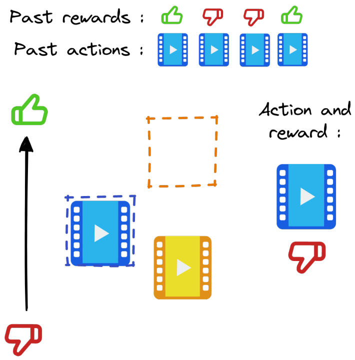
Greedy

Exploration-Exploitation dilemma:

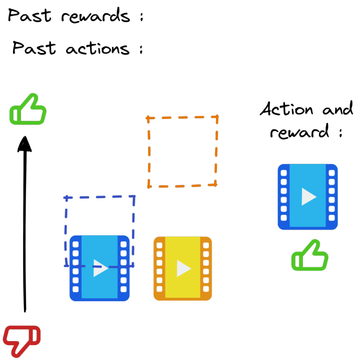
exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



Greedy



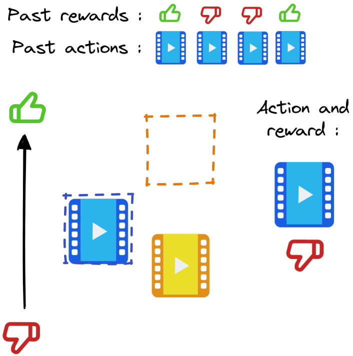
ϵ-Greedy

Exploration-Exploitation dilemma:

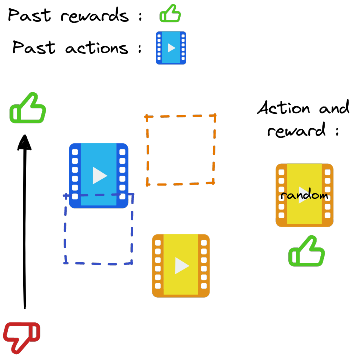
exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



Greedy



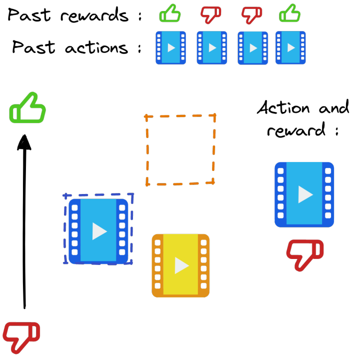
ϵ-Greedy

Exploration-Exploitation dilemma:

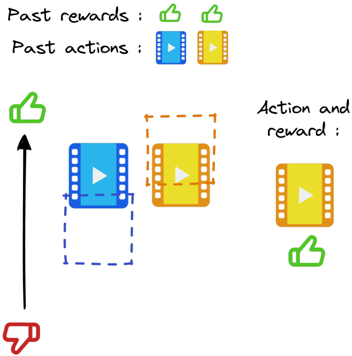
exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



Greedy



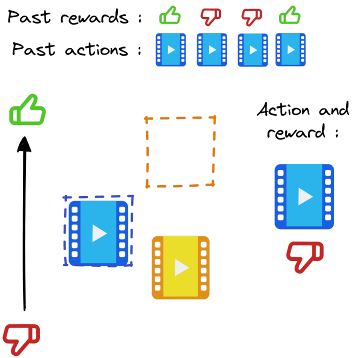
ϵ-Greedy

Exploration-Exploitation dilemma:

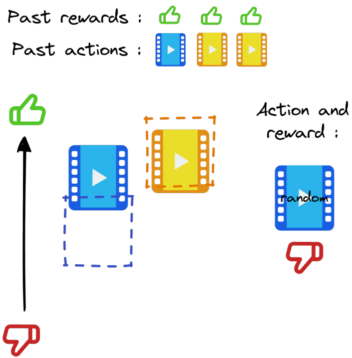
exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



Greedy



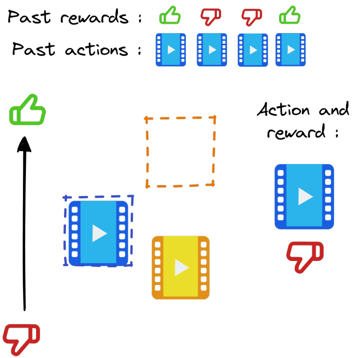
ϵ-Greedy

Exploration-Exploitation dilemma:

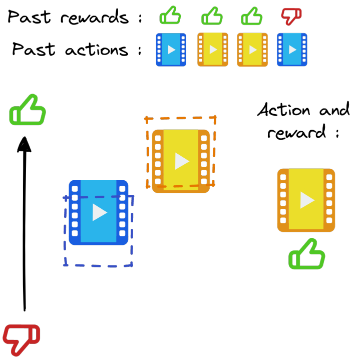
exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

Multi-Armed Bandits (MAB): greedy and ϵ-greedy



Greedy



ϵ-Greedy

Exploration-Exploitation dilemma:

exploiting the knowledge acquired from interactions lowers the regret ...

... but *exploring* different actions multiple times is required to find the optimal action.

MAB: lower bounds of regret (Silver, 2022)

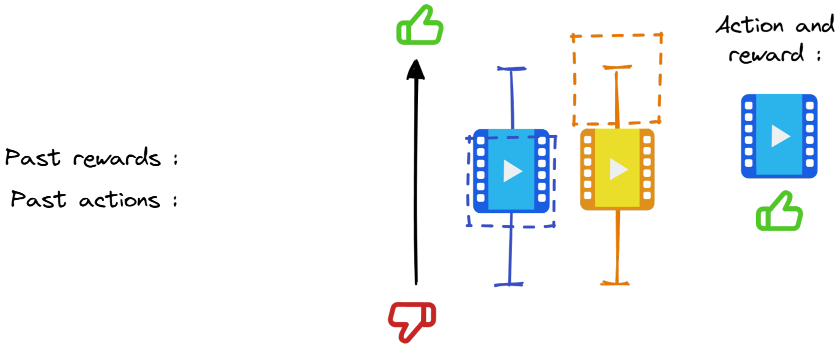
The expected cumulative regret admits a **logarithmic lower bound**:

$$\overline{R}(T) \geqslant \log(T) \left[\underbrace{\sum_{a \in \mathcal{A} \setminus a^*}}_{\text{harder with more actions}} \frac{\overbrace{\mathbb{E}\left[r(a^*)\right] - \mathbb{E}\left[r(a)\right]}^{\text{more regret when strongly suboptimal}}}{\underbrace{D_{KL}(r(a) \| r(a^*))}_{\text{harder when actions look similar}}} + o(1) \right]$$

It is impossible to find a MAB algorithm with bounded expected cumulative regret!

MAB: Upper Confidence Bound (UCB) (Silver, 2022)

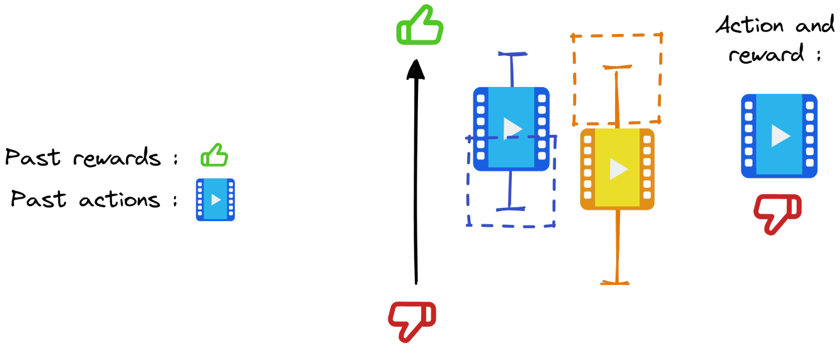
Optimism in the face of uncertainty



UCB achieves logarithmic asymptotic cumulative regret !

MAB: Upper Confidence Bound (UCB) (Silver, 2022)

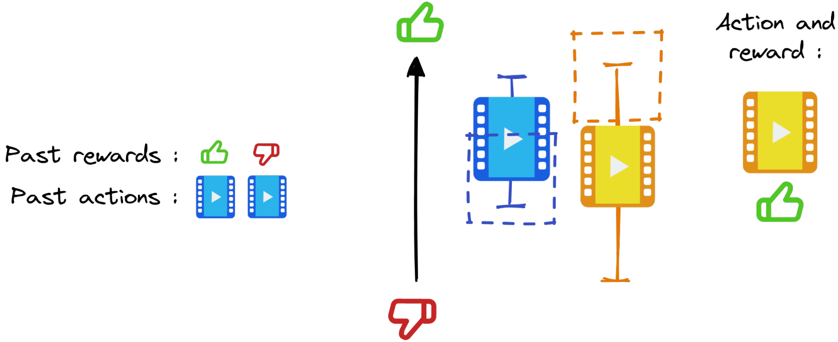
Optimism in the face of uncertainty



UCB achieves logarithmic asymptotic cumulative regret !

MAB: Upper Confidence Bound (UCB) (Silver, 2022)

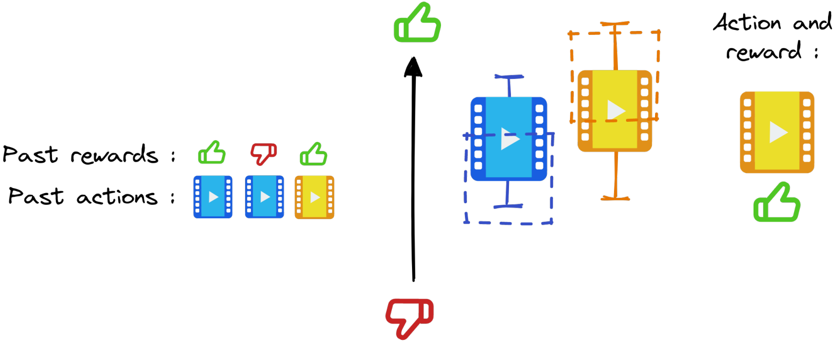
Optimism in the face of uncertainty



UCB achieves logarithmic asymptotic cumulative regret !

MAB: Upper Confidence Bound (UCB) (Silver, 2022)

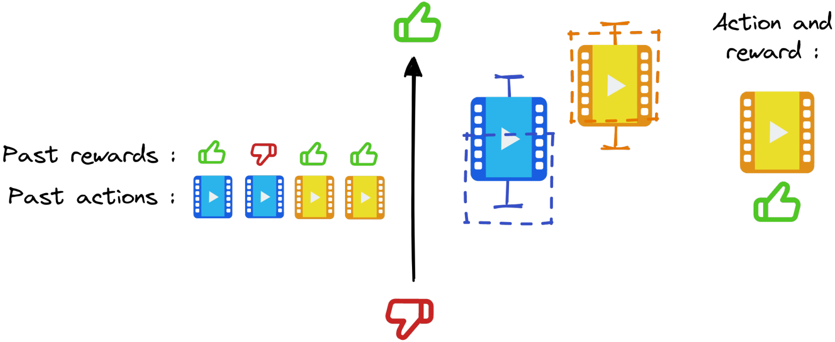
Optimism in the face of uncertainty



UCB achieves logarithmic asymptotic cumulative regret !

MAB: Upper Confidence Bound (UCB) (Silver, 2022)

Optimism in the face of uncertainty



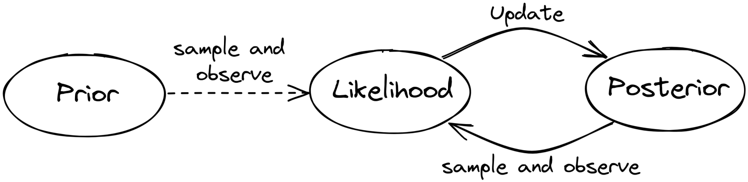
UCB achieves logarithmic asymptotic cumulative regret !

Short digression on Bayesian inference

How to maintain an estimate of the full distribution of rewards, and update it after having interacted ?

Use parameterized families of distributions for the rewards, like $\mathcal{N}(\mu, \sigma)$ or $\mathcal{B}(\mu)$.

Likelihood	**Prior**	**Posterior**
Distribution of rewards given a parameter.	Best guess of parameter before interactions.	Best guess of parameter after interactions.
$P(r \mid \theta)$	$P(\theta)$	$P(\theta \mid r)$



Bayes law

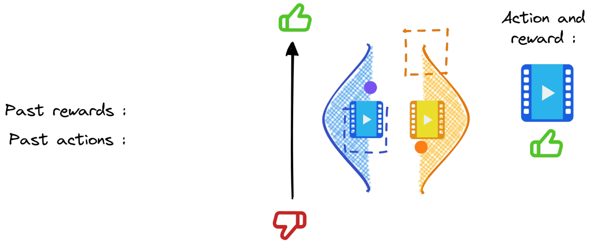
$$\underbrace{P(\theta \mid r)}_{\text{Posterior}} \propto \overbrace{P(r \mid \theta)}^{\text{Likelihood}} \underbrace{P(\theta)}_{\text{Prior}}$$

MAB: Thompson sampling

- **Probability matching**: we want to select an action according to its probability of being optimal.
- Thompson Sampling does this by
  1. sampling a parameter value for each action
  2. selecting the best action under the chosen parameters
  3. observing the reward and updating the corresponding action

MAB: Thompson sampling (Chapelle and Li, 2011)

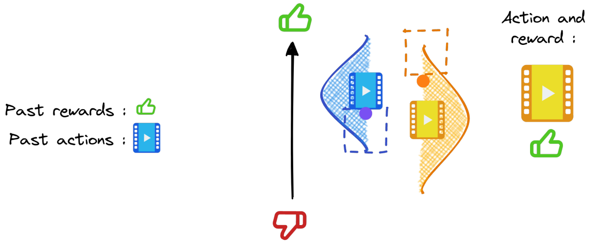
- Rewards are clicks/skips: $r(a) \sim \mathcal{B}(\mu_a)$
- Prior and posterior distributions on μ_a ? $\rightarrow$ **Conjugate prior** is $\mu_a \sim \mathrm{B}(\alpha_a, \beta_a)$
- Simple update: $\alpha_a^{t+1} = \alpha_a^t + r$ and $\beta_a^{t+1} = \beta_a^t + (1 - r)$ $(\alpha_a^0 = \beta_a^0 = 0)$



Beta-bernoulli Thompson Sampling achieves the lower bound !

MAB: Thompson sampling (Chapelle and Li, 2011)

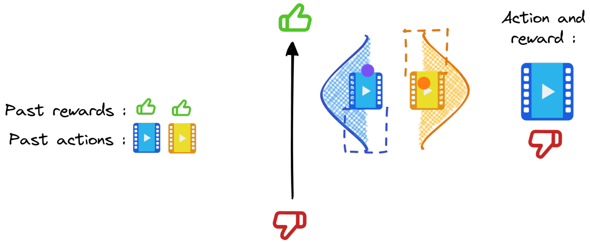
- Rewards are clicks/skips: $r(a) \sim \mathcal{B}(\mu_a)$
- Prior and posterior distributions on μ_a ? $\rightarrow$ **Conjugate prior** is $\mu_a \sim \mathrm{B}(\alpha_a, \beta_a)$
- Simple update: $\alpha_a^{t+1} = \alpha_a^t + r$ and $\beta_a^{t+1} = \beta_a^t + (1 - r)$ $(\alpha_a^0 = \beta_a^0 = 0)$



Beta-bernoulli Thompson Sampling achieves the lower bound !

MAB: Thompson sampling (Chapelle and Li, 2011)

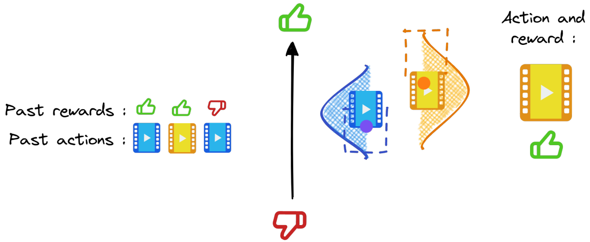
- Rewards are clicks/skips: $r(a) \sim \mathcal{B}(\mu_a)$
- Prior and posterior distributions on μ_a ? $\rightarrow$ **Conjugate prior** is $\mu_a \sim \mathrm{B}(\alpha_a, \beta_a)$
- Simple update: $\alpha_a^{t+1} = \alpha_a^t + r$ and $\beta_a^{t+1} = \beta_a^t + (1 - r)$ $(\alpha_a^0 = \beta_a^0 = 0)$



Beta-bernoulli Thompson Sampling achieves the lower bound !

MAB: Thompson sampling (Chapelle and Li, 2011)

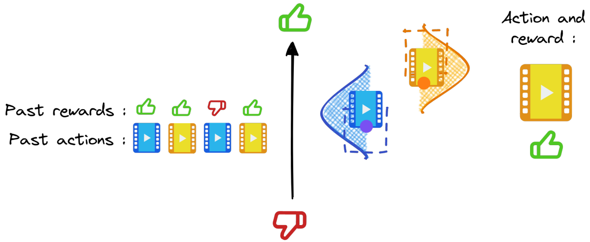
- Rewards are clicks/skips: $r(a) \sim \mathcal{B}(\mu_a)$
- Prior and posterior distributions on μ_a ? $\rightarrow$ **Conjugate prior** is $\mu_a \sim \mathrm{B}(\alpha_a, \beta_a)$
- Simple update: $\alpha_a^{t+1} = \alpha_a^t + r$ and $\beta_a^{t+1} = \beta_a^t + (1 - r)$ $(\alpha_a^0 = \beta_a^0 = 0)$



Beta-bernoulli Thompson Sampling achieves the lower bound !

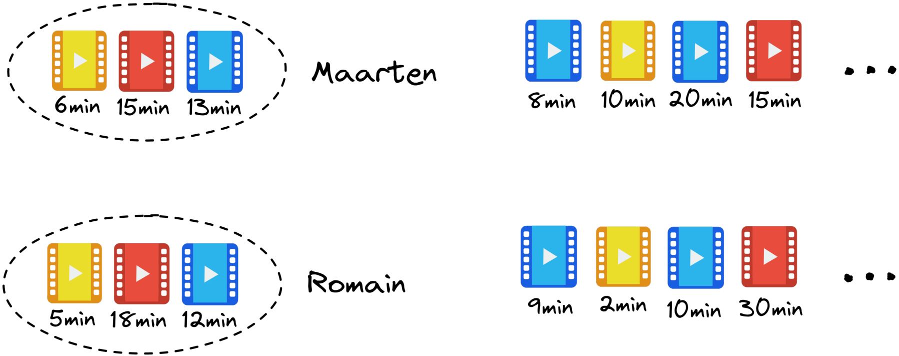
MAB: Thompson sampling (Chapelle and Li, 2011)

- Rewards are clicks/skips: $r(a) \sim \mathcal{B}(\mu_a)$
- Prior and posterior distributions on μ_a ? $\rightarrow$ **Conjugate prior** is $\mu_a \sim \mathrm{B}(\alpha_a, \beta_a)$
- Simple update: $\alpha_a^{t+1} = \alpha_a^t + r$ and $\beta_a^{t+1} = \beta_a^t + (1 - r)$ $(\alpha_a^0 = \beta_a^0 = 0)$



Beta-bernoulli Thompson Sampling achieves the lower bound !

What about dwell-time ?



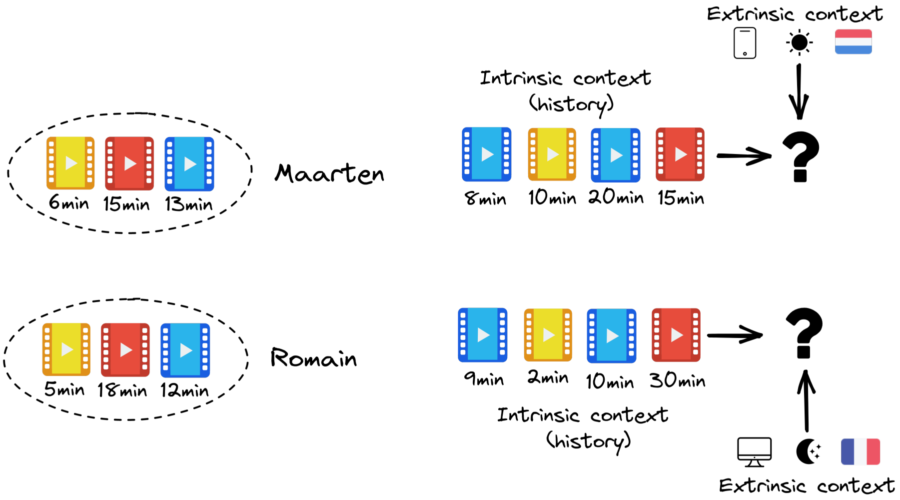
How to quickly find Maarten's and Romain's preferences?

MAB: Thompson sampling for dwell-time

- Rewards are positive real numbers sampled from a normal distribution with fixed variance: $r(a) \sim \mathcal{N}(\mu_a, \sigma^2)$
  $\rightarrow \sigma$ can be interpreted as an exploration parameter.
- Prior and posterior distributions on μ_a ? $\rightarrow$ **Conjugate prior** is $\mu_a \sim \mathcal{N}(\nu_a, \frac{\sigma^2}{\lambda_a^2})$
- Update: $(\lambda_a^{t+1})^2 = (\lambda_a^t)^2 + \frac{1}{\sigma^2}$ and $\mu_a^{t+1}\left[1 + \frac{1}{\sigma^2(\lambda_a^t)^2}\right] = \mu_a^t + \frac{r}{\sigma^2(\lambda_a^t)^2}$

Normal-Normal Thompson Sampling achieves a logarithmic lower bound! (Agrawal and Goyal, 2017)

Contextual click/dwell-time maximization



What to recommend next to Maarten and Romain ?

Contextual click/dwell-time maximization

Same … but different !

We still have no prior knowledge about the current user …

… but we use knowledge from previous users !

Contextual Bandits: LinTS (Agrawal and Goyal, 2013)

- Rewards are sampled from a normal distribution with fixed variance and where the mean is a linear combination of context features: $r^t(a) \sim \mathcal{N}(X_t^T \mu_a, \sigma)$.
- Prior and posterior distributions on μ_a ?
  $\rightarrow$ **Conjugate prior** is a multivariate normal distribution $\mu_a \sim \mathcal{N}(\nu_a, \sigma \cdot \Lambda_a^{-1})$.
- Update: $\Lambda_a^{t+1} = \Lambda_a^t + X_t^T X_t$ and $\Lambda_a^{t+1}\nu_a^{t+1} = \Lambda_a^t \nu_a^t + r_t \cdot X_t$.

Contextual Bandits: LinTS extensions (Riquelme et al., 2018)

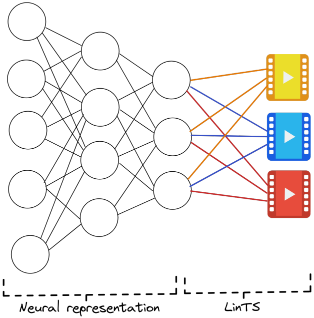
$$\sigma_a \sim \Gamma^{-1}(\alpha_a, \beta_a)$$

$$\mu_a \mid \sigma_a \sim \mathcal{N}(\nu_a, \sigma_a \cdot \Lambda_a^{-1})$$

$$\alpha_a^{t+1} = \alpha_a^t + 1/2$$

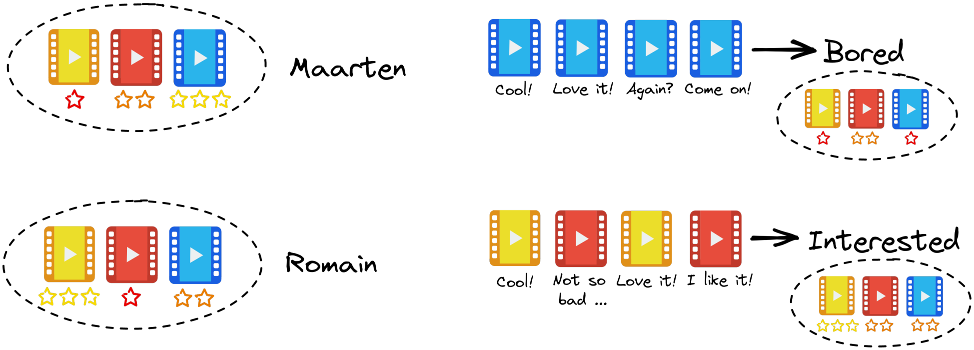
$$\beta_a^{t+1} = \beta_a^t + 1/2 \left(r_t^2 - {\mu_a^t}^T \Lambda_a^t \mu_a^t \right)$$

Unknown variance



Non-linear rewards

Long-term user engagement



How to satisfy Maarten and Romain on the long run?

Can bandits solve long-term user engagement?

The objectives match (maximize sum of rewards / minimize cumulative regret) ...

... but the methods we used assume that actions are independent of future rewards conditioned on the current context, i.e., **the user is static**.

$\rightarrow$ need explicitly capturing causal effect of recommendations on future user states: **Reinforcement Learning**

MDPs and POMDPs

Partially-observable Markov Decision Process:

- States $s \in \mathcal{S}$: user's mind.
- Observations $o \in \mathcal{O}$: history, extrinsic context, ...
- Actions $a \in \mathcal{A}$: recommendations.
- Reward function $r : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$: click, dwell-time, etc.
- Transition probabilities $T(s'|s, a)$: how recommendations influence the user.
- Initial state distribution $S(s_1)$: user state when they arrive on the platform.
- Observation probabilities $\Omega(o|s, a)$: how the user's mind is revealed.

Goal: Maximize the expected cumulative rewards

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=1}^{T} r(s_t, a_t) \right] \quad , \tau = (s_1, a_1, \ldots, s_T, a_T)$$

RL101: Policy gradients (Sutton and Barto, 2018)

- Objective function $J(\pi_\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=1}^{T} r(s_t, a_t)\right]$
- Policy gradient theorem:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{t\sim\pi_\theta}\left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \sum_{t'=t}^{T} r(s_{t'}, a_{t'})\right]$$

- **REINFORCE** algorithm alternate between two steps:

Collect a trajectory τ^k from π_{θ^k} $\leftrightarrow$ improve the policy $\theta^{k+1} = \theta^k + \alpha\nabla_\theta J(\pi_\theta)\mid_{\theta=\theta^k}$

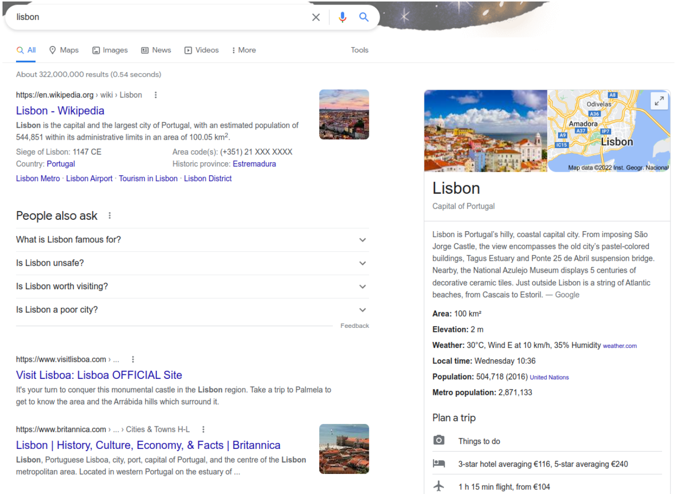
RL101: Dynamic programming (Sutton and Barto, 2018)

- Q-function: $Q^\pi(a|s) = \mathbb{E}_{\tau\sim\pi, s_1=s, a_1=a}\left[\sum_{t=1}^{T} r(s_t, a_t)\right]$
  $\rightarrow$ how good action a is in state s.
- Bellman Equation: $Q^\pi(a|s) = r(s,a) + \mathbb{E}_{a'\sim\pi(\cdot|s')}\left[Q(a'|s')\right]$
- **Q-Learning** alternates between two steps:

$$\text{Collect experience } (s,a,r,s') \text{ from an } \epsilon\text{-greedy policy w.r.t } Q \leftrightarrow$$
$$\text{improve the } Q\text{-function } Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\left[r + \max_{a'\in\mathcal{A}} Q(s',a')\right]$$

- Actor Critic combines policy gradient and dynamic programming (actually slightly more complex than that ...).

Learning from logs & session optimization



We need to cover many aspects of the query and anticipate for future needs.

Optimizer's curse and deadly triad

Learning from logged interactions, without interventions, is hard:

- **Optimizer's curse:** A maximization process is likely to select an overestimated solution. With n items of expected rewards $r_1, \dots, r_n$, we can have $\mathbb{E}[\hat{r}_k] = r_k$ and yet $\mathbb{E}[\hat{r}_{k^*} - r_{k^*}] > 0$ with $k^* = \arg\max_k \hat{r}_k$
  $\rightarrow$ We will be disappointed. (Jeunen and Goethals, 2021)
- **Deadly Triad**: Optimizer's curse is much worse when 3 conditions are satisfied: (van Hasselt et al., 2018)
  - Off-Policy training
  - Dynamic programming
  - Q-function approximation

Back to importance sampling

Under the logging policy...

... what would have been the probability of observing that sequence of rankings ?

$$R^{IS} = \sum_{\tau \in \mathcal{D}} \sum_{t=1}^{T} \underbrace{\prod_{i=1}^{t} \frac{\pi(a_i|s_i)}{\pi_L(a_i|s_i)}}_{\text{product of past IS weights}} r_t$$

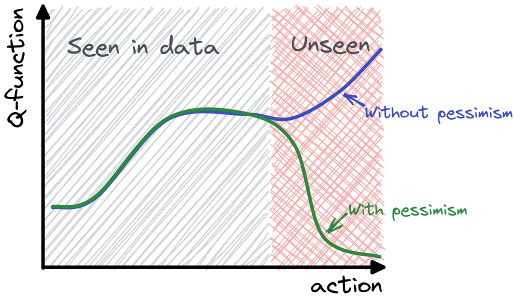
What about the variance ?

Even worse than in traditional CLTR !

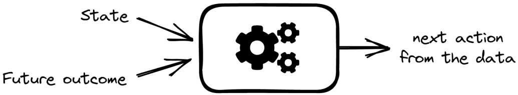
$\rightarrow$ variance grows exponentially with horizon length.

Offline RL: Find a way to recover policies staying within the support of the logging policy (Levine, 2021)

Offline RL



Pessimistic Q-Functions
(Kumar et al., 2020)



Filtered Behavior cloning
(Chen et al., 2021)

Specific challenges in IR

- The environment is highly stochastic: two users with same history may react differently to a recommendation.
- There are a lot of actions: from thousands to billions and above.
- How to integrate user models ?

Summary

- We must **balance exploration and exploitation** to find user preferences quickly and reliably (Bandits)
- We can augment bandits algorithms with **context features to leverage knowledge from other users** (Contextual Bandits)
- We must capture the **effect of recommendations on future user behavior** to enable long-term satisfaction (Reinforcement Learning)

Q&A

Questions, comments, . . .

Agenda

09.00	Start
09.00–09.05	Domestic matters – Maarten and Romain
09.05–09.20	Setting the scene – Maarten
09.20–09.45	Counterfactual learning-to-rank – Maarten
09.45–10.15	Bandits & Reinforcement learning in IR – Romain
10.15–10.25	Conclusion – Maarten and Romain
10.25–10.30	Final Q&A
10.30	End

Conclusion

Plan for this part

- Taking stock
- Directions not covered
- Challenges

Taking stock

- Using user interactions to evaluate or optimize interactive systems
  - Online vs off-policy
  - Counterfactual evaluation / learning
- Counterfactual learning to rank
- Bandits and reinforcement learning

What we have not covered

- Recent advances in bias-variance trade-offs
- Complex (very large) action spaces
- Tuning hyperparameters
- Working with multiple logging policies
- Dealing with distributional shifts
- Combinations of online & offline, with occasional online exploration to collect new data (Oosterhuis and de Rijke, 2021)
- Limitations of de-biasing in counterfactual learning to rank (Oosterhuis, 2022)
- Simulation environments
- Libraries and packages

Challenges

Guarantees on . . .

- Accuracy, also for rare phenomena
- Efficiency during both training and inference
- Reliability when assumptions begin to fail (e.g., on user behavior)
- Reproducibility of experimental results
- Resilience – against distributional shifts and adversarial attacks
- Safety of user data and proprietary data

Agenda

09.00	Start
09.00–09.05	Domestic matters – Maarten and Romain
09.05–09.20	Setting the scene – Maarten
09.20–09.45	Counterfactual learning-to-rank – Maarten
09.45–10.15	Bandits & Reinforcement learning in IR – Romain
10.15–10.25	Conclusion – Maarten and Romain
10.25–10.30	Final Q&A
10.30	End

Final Q&A

Q&A

Questions, comments, . . .

References i

S. Agrawal and N. Goyal. Thompson sampling for contextual bandits with linear payoffs. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pages III–1220–III–1228. JMLR.org, 2013.

S. Agrawal and N. Goyal. Near-optimal regret bounds for thompson sampling. *J. ACM*, 64(5), 2017.

O. Chapelle and L. Li. An empirical evaluation of thompson sampling. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, pages 2249–2257. Curran Associates Inc., 2011.

L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *NeurIPS 2021: Conference on Neural Information Processing Systems*, 2021.

A. Chuklin, I. Markov, and M. de Rijke. *Click Models for Web Search*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, August 2015.

O. Jeunen and B. Goethals. Pessimistic reward models for off-policy learning in recommendation. In *Fifteenth ACM Conference on Recommender Systems*, pages 63–74. ACM, 2021.

References ii

T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. Inf. Syst.*, 25(2): 7–es, 2007.

T. Joachims, A. Swaminathan, and T. Schnabel. Unbiased learning-to-rank with biased user feedback. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 781–789. ACM, 2017.

A. Kumar, A. Zhou, G. Tucker, and S. Levine. Conservative q-learning for offline reinforcement learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*. Curran Associates Inc., 2020.

S. Levine. A gentle introduction to offline reinforcement learning. https://www.youtube.com/watch?v=tW-BNW1ApN8, 2021.

J. McInerney, B. Brost, P. Chandar, R. Mehrotra, and B. A. Carterette. Counterfactual evaluation of slate recommendations with sequential reward interactions. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1779–1788. ACM, 2020.

H. Oosterhuis. Reaching the end of unbiasedness: Uncovering implicit limitations of click-based learning to rank. *arXiv preprint arXiv:2206.12204*, 2022.

H. Oosterhuis and M. de Rijke. Differentiable unbiased online learning to rank. In *CIKM 2018: International Conference on Information and Knowledge Management*, pages 1293–1302. ACM, October 2018.

H. Oosterhuis and M. de Rijke. Unifying online and counterfactual learning to rank. In *WSDM 2021: 14th International Conference on Web Search and Data Mining*. ACM, March 2021.

H. Oosterhuis, R. Jagerman, and M. de Rijke. Unbiased learning to rank: Counterfactual and online approaches. In *Companion Proceedings of the Web Conference 2020*, pages 299–300. ACM, April 2020.

D. Precup, R. S. Sutton, and S. P. Singh. Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 759–766, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

C. Riquelme, G. Tucker, and J. Snoek. Deep bayesian bandits showdown: An empirical comparison of bayesian deep networks for thompson sampling. In *ICLR 2018*, 2018.

References iv

Y. Saito and T. Joachims. Counterfactual learning and evaluation for recommender systems: Foundations, implementations, and recent advances. In *Fifteenth ACM Conference on Recommender Systems*, pages 828–830. ACM, 2021.

F. Sarvi, M. Heuss, M. Aliannejadi, S. Schelter, and M. de Rijke. Understanding and mitigating the effect of outliers in fair ranking. In *WSDM 2022: The Fifteenth International Conference on Web Search and Data Mining*. ACM, February 2022.

D. Silver. RL Course – Lecture 9: Exploitation and exploration. http://youtube.com, 2022.

R. Sutton and A. Barto. *Reinforcement Learning : An Introduction*. MIT Press, 2018.

H. van Hasselt, Y. Doron, F. Strub, M. Hessel, N. Sonnerat, and J. Modayil. Deep reinforcement learning and the deadly triad. *arXiv preprint arXiv:1812.02648*, 2018.

L. A. Wasserman. *All of Statistics*. Springer, 2004.

User Models and Interactive IR

ESSIR 2022

Romain Deffayet[1,2] and **Maarten de Rijke**[2]
[1]Naver Labs Europe, [2]University of Amsterdam

July 19, 2022, 09.00–10.30

r.e.deffayet@uva.nl, m.derijke@uva.nl