RED: Recommendations Encouraging Diversity

Project check-in #2

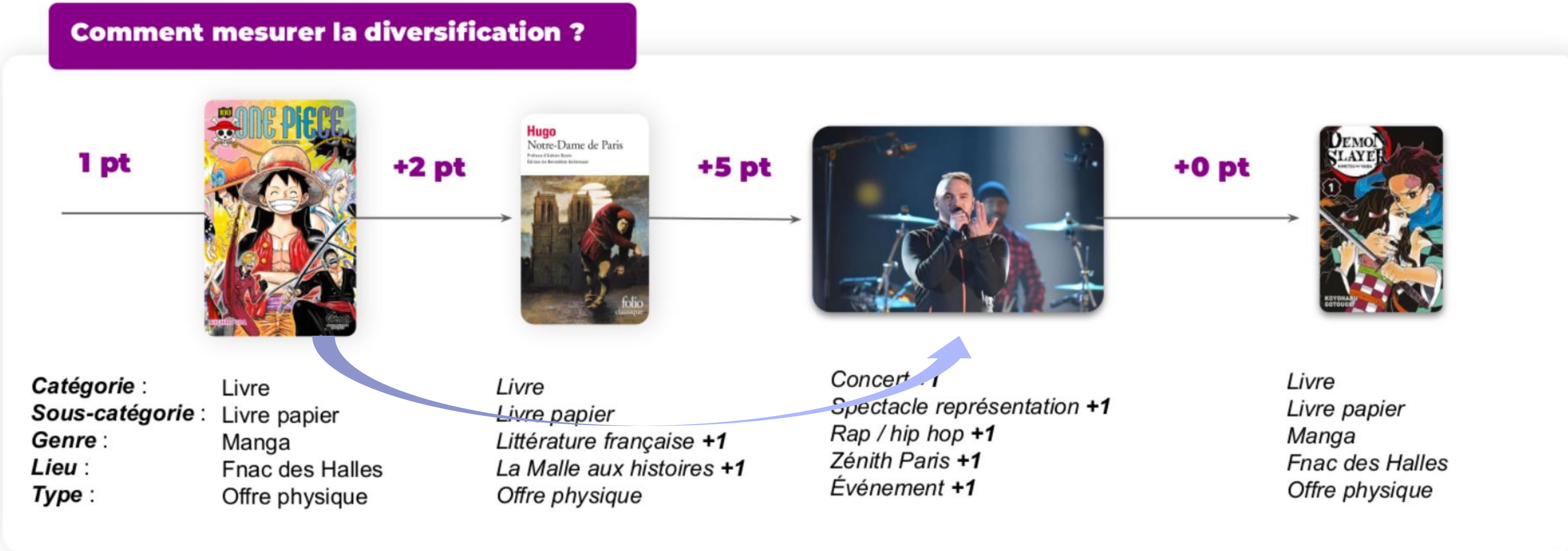
10.06.2025

Clemence Reda

Background

Pass Culture a phone app for French teens (<20yr) to browse and book cultural goods nearby with credits.

Diversification points obtained for each new category / subcategory / genre / location / type (a bit like set cover; achievement score); those are not visible to the user

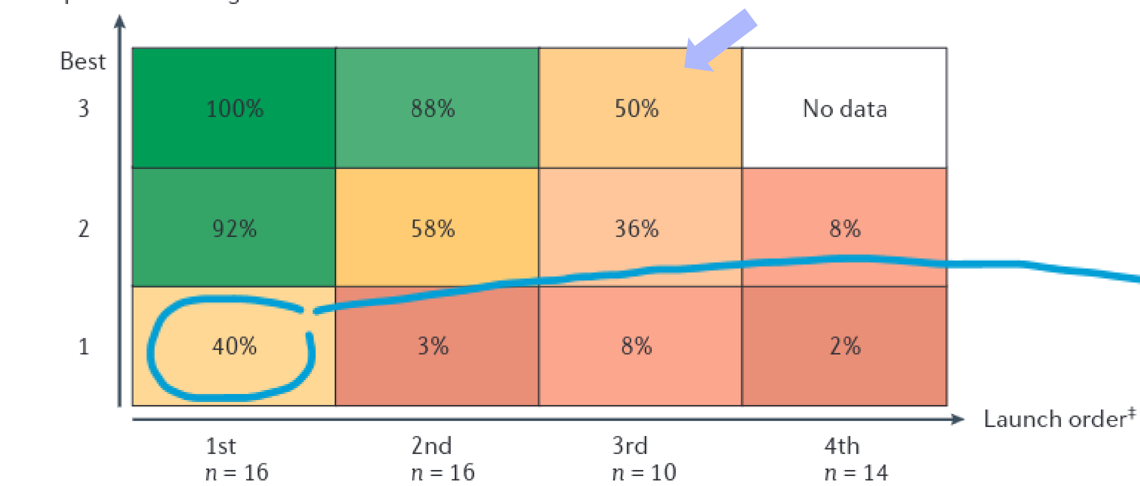


Courtesy of Jill-Jenn Vie (Inria SODA).

Drug repurposing First-in-class versus Best-in-class

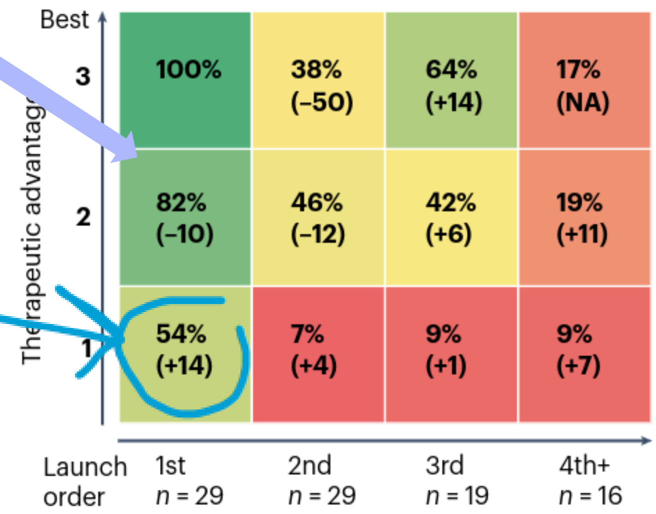
$\underbrace{\text{First-in-class}}_{\text{First launched in that mechanistic class}}$ versus $\underbrace{\text{Best-in-class}}_{\text{Highest therapeutic advantage [...]}}$

In 2013 **[1]**



Trend in market value (aka sales)

In 2023 **[2]**



"The data indicated that it is slightly better to be first than to be best"

"[...] products that are first-to-launch increasingly tend to perform better [...]"

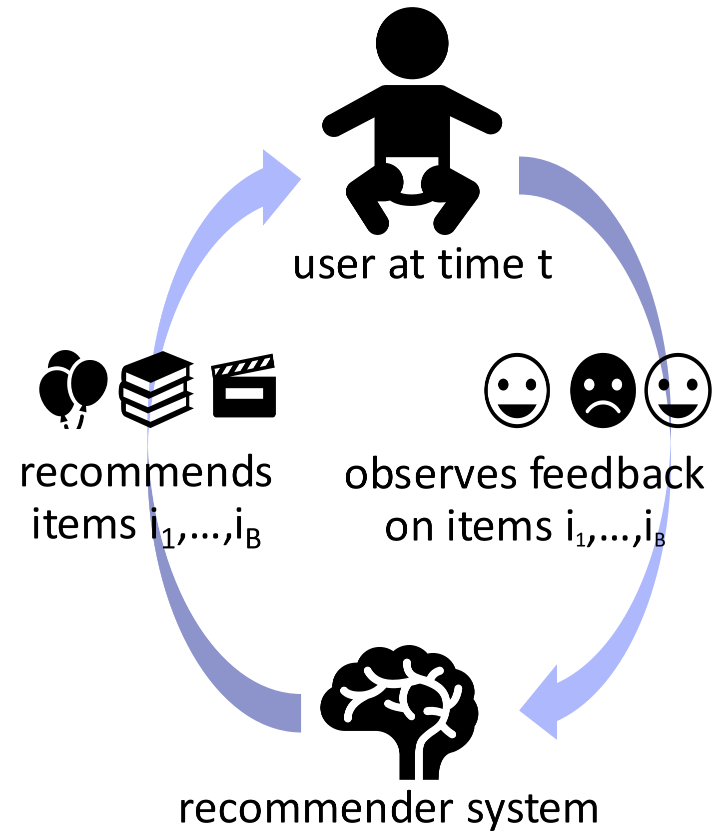
[1] Schulze, U., & Ringel, M. (2013). What matters most in commercial success: first-in-class or best-in-class?. *Nature Reviews Drug Discovery*, *12*(6), 419-420.
[2] Spring, L., Demuren, K., Ringel, M., & Wu, J. (2023). First-in-class versus best-in-class: an update for new market dynamics. *Nat Rev Drug Discov*, *22*(7), 531-532.

Objective of RED to design recommender systems for *personalized good* and *diverse* items.

taking into account the **user's interests/item history**

suggesting items with a **high probability of positive feedback** from the user

somewhat "**controllable**" recommendations: *e.g.*, out of the user's comfort zone



Scientific challenges

- Incorporate a tradeoff between quality and diversity
- Versatile enough to be applicable in all use cases
- Not too computationally expensive (millions of items)

Data

Φ_i : embedding for item i with $|\Phi_i|=1$

H_u : item history for user u (at all times)

q : *(un)known* feedback model with positive values

B : the size of the batch of items to recommend at each time

Data sets I don't have access to Pass Culture data

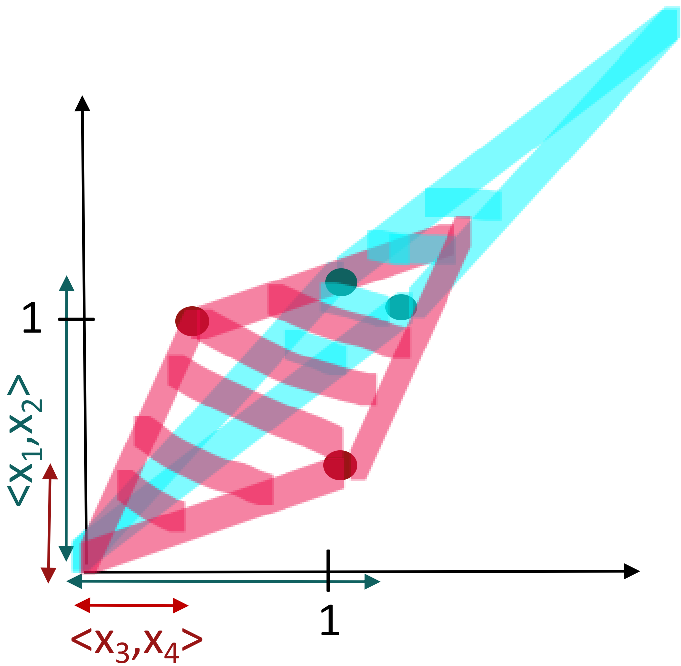
Synthetic : Gaussian Generate item and user embeddings Φ_i and c_u at random, $q(i,u) \propto \Phi_i^T c_u$

Pseudo-real : MovieLens **[1]** for movie recommendation : 0 (not seen), 1, 2, ..., 5 stars
Apply `universal-sentence-encoder` to movie title & keywords
q(i,u) := SVD(rating matrix)[i,u] # fill out 0's with Singular Value Decomposition
RMSE: `0.96` (on held-out data)

Pseudo-real : PREDICT **[2]** for drug repurposing : 0 (not tested), 1 (success), -1 (failure)
Select 10 first PCs from drug and disease feature vectors to get Φ_i and c_u
% of cumulative explained variance: 85% (drugs), 68% (diseases)
$q(i,u) := RF([\Phi_i, c_u])$ # classification with Random Forest
Average `accuracy`: `0.97` (on held-out data)

[1] https://movielens.org/
[2] Réda, C. (2023). PREDICT drug repurposing dataset (2.0.1) [Data set]. Zenodo. doi:10.5281/zenodo.7983090

Diversity metrics introduction to similarity kernels



$$K_{\{xi,xj\},\{xi,xj\}} = \begin{pmatrix} 1 & \langle x_i,x_j\rangle \\ \langle x_i,x_j\rangle & 1 \end{pmatrix}$$

$$= 1 - \langle x_i,x_j\rangle^2 \geq 0$$

$|K_{\{x1,x2\},\{x1,x2\}}| \leq |K_{\{x3,x4\},\{x3,x4\}}|$

where $\langle x_3,x_4\rangle \leq \langle x_1,x_2\rangle$ (cosine similarity)

(Item embedding) kernel K $K_{\{x\},\{y\}} = \langle \Phi(x), \Phi(y)\rangle$
e.g., linear kernel: Φ = Id, ~ similarity b/w items with K positive definite

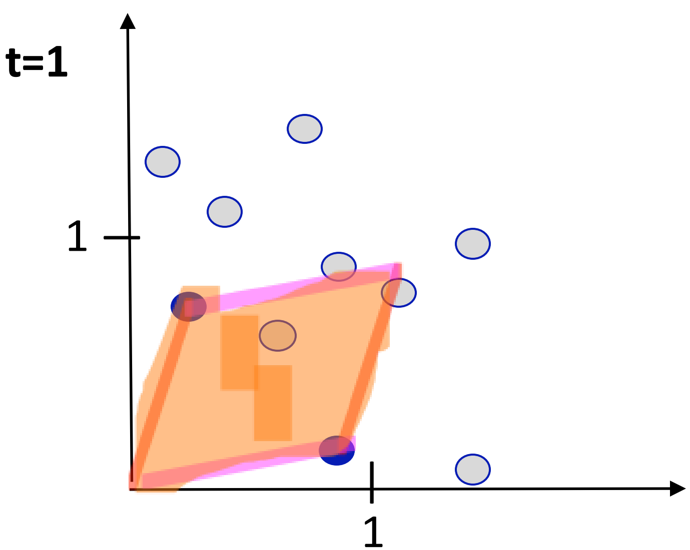
(Log)-determinant of (a subset of) the kernel Volume in space occupied by a set S of items:

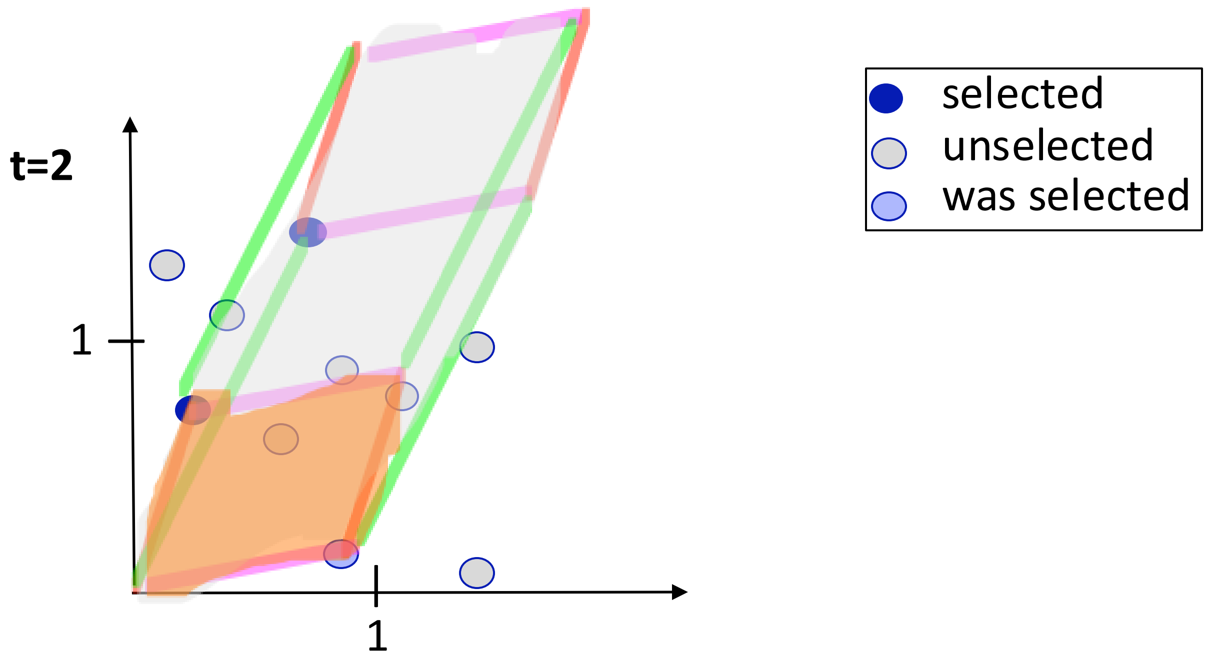
$$\mathrm{vol}(S)=|\det(K_{S,S})|^{1/2}$$

Drawn for all i $|x_i|^2=1$, K linear kernel in 2D
… but works for any # of dimension and # of points (and any symmetric kernel)

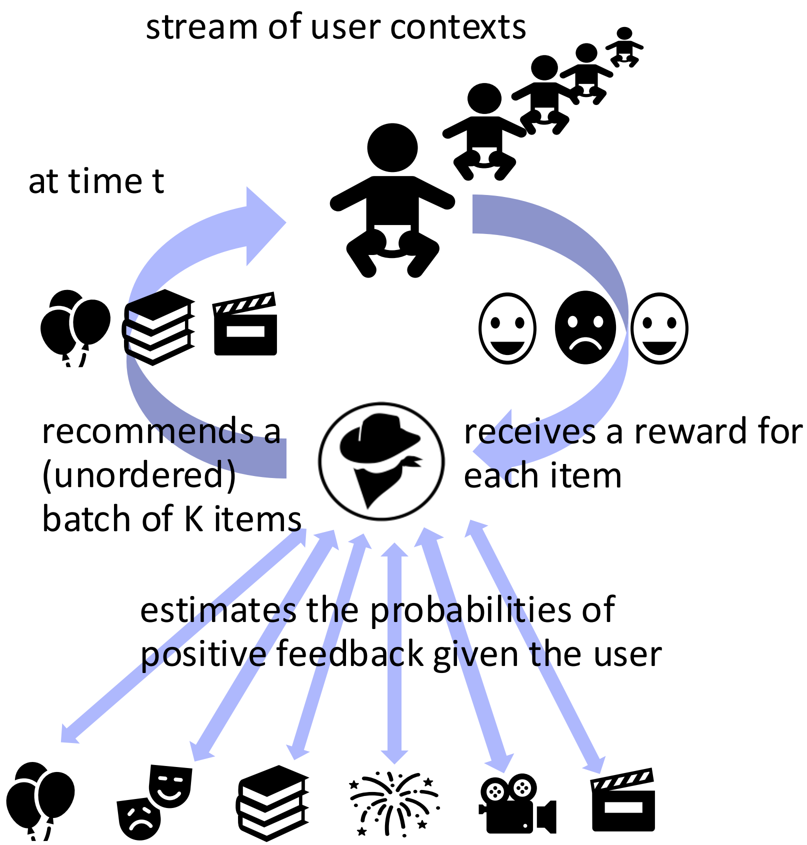
(Pointwise) relevance and diversity metrics with kernels

Relevance: "Click-through-rate" $R(S=\{i_1,\ldots,i_B\}, H_t) = \sum_{i<B} q(i,u_t) / B$
Intrabatch diversity: "inside a batch" $AD(S) = vol(K_{S,S})$
Interbatch diversity: "across history" $ED(S, H_t) = vol(K_{S \cup Ht, S \cup Ht}) - vol(K_{H,H})$
Avg # unique recommendations in history Complementary to the interbatch metric





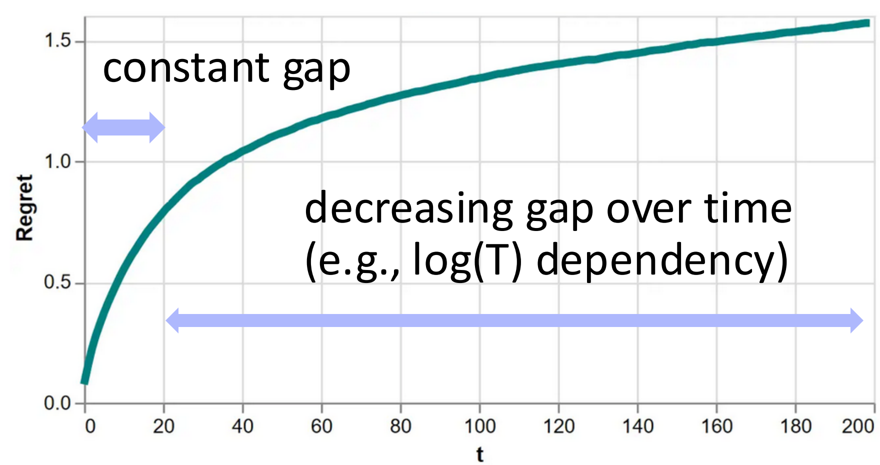
(Stochastic, contextual) multi-armed bandits or MABs
online reward maximization when q is *unknown*



Definition A MAB is defined by its sampling rule: arm with highest probability of positive feedback?

Reward maximization Performance is compared to a deterministic oracle with access to the true q

$$\text{Regret}(T) = \Sigma_t \text{ Perf}(q, t) - \text{Perf}(q_t, t)$$



(Finite) determinantal Point Processes or DPPs sampling diverse set of points in a data-driven fashion

Point Process A distribution over finite subsets of a (finite) set $X=\{x_1,x_2,\ldots,x_N\}$
Definition* The probability of a subset is correlated to its determinant

$$\mathrm{Prob}(S \subseteq X) \propto \det(K_{S,S}) \text{ where } K_{S,S} \text{ is the kernel built on the subset } S \subseteq X$$

- Sampling Algorithms in $O(N^3)$ **[1]** or O(N.poly(k)) **[2]** to sample k points out of N from a DPP

- Conditioning (SAMPLE) [3] $\mathrm{Prob}(S \subseteq X \mid H \subseteq X) \propto \det(K_{S,S} - K_{S,H}K_{H,H}^{-1}(K_{S,H})^T)$

- MAP inference (MAX) Finding a subset S of size k maximizing $\mathrm{Prob}(S \subseteq X)$ is NP-hard

[1] Theorem 7 and Algorithm 18 in Hough, J. B., Krishnapur, M., Peres, Y., & Virág, B. (2006). Determinantal processes and independence.
[2] Calandriello, D., Derezinski, M., & Valko, M. (2020). Sampling from a k-DPP without looking at all items. *Advances in Neural Information Processing Systems* 33, 6889-6899.
[3] Borodin, A., & Rains, E. M. (2005). Eynard–Mehta theorem, Schur process, and their Pfaffian analogs. *Journal of statistical physics*, *121*, 291-317.

(Finite) determinantal Point Processes or DPPs sampling diverse set of points in a data-driven fashion

Point Process A distribution over finite subsets of a (finite) set $X=\{x_1,x_2,\ldots,x_N\}$

Definition* The probability of a subset is correlated to its determinant

$$\mathrm{Prob}(S \subseteq X) \propto \det(K_{S,S}) \text{ where } K_{S,S} \text{ is the kernel built on the subset } S \subseteq X$$

- Sampling Algorithms in $O(N^3)$ **[1]** or $O(N.\mathrm{poly}(k))$ **[2]** to sample k points out of N from a DPP ✗
- Conditioning (SAMPLE) **[3]** $\mathrm{Prob}(S \subseteq X \mid H \subseteq X) \propto \det(K_{S,S} - K_{S,H}K_{H,H}^{-1}(K_{S,H})^T)$
  in $O(k^3+|H|^3+k^2|H|^2+(N-|H|)k^2)$ **[4]**

  includes history
- MAP inference (MAX) Finding a subset S of size k maximizing $\mathrm{Prob}(S \subseteq X)$ is NP-hard

  greedy approx. **[5]** in $O(k^2N)$

[1] Theorem 7 and Algorithm 18 in Hough, J. B., Krishnapur, M., Peres, Y., & Virág, B. (2006). Determinantal processes and independence.

[2] Calandriello, D., Derezinski, M., & Valko, M. (2020). Sampling from a k-DPP without looking at all items. *Advances in Neural Information Processing Systems* 33, 6889-6899.

[3] Borodin, A., & Rains, E. M. (2005). Eynard–Mehta theorem, Schur process, and their Pfaffian analogs. *Journal of statistical physics*, *121*, 291-317.

[4] Mariet, Z., Gartrell, M., & Sra, S. (2019, April). Learning determinantal point processes by corrective negative sampling. In *The 22nd International Conference on Artificial Intelligence and Statistics* (pp. 2251-2260). PMLR.

[5] Chen, L., Zhang, G., & Zhou, E. (2018). Fast greedy map inference for determinantal point process to improve recommendation diversity. *Advances in Neural Information Processing Systems*, *31*.

(Finite) determinantal Point Processes or DPPs leveraging the quality-diversity decomposition

Quality-diversity decomposition (QD) **[1]** $K = Q\Phi^T\Phi Q$

where Φ is the item embedding matrix $|\Phi_i|=1$ and $Q = \mathrm{diag}(q_1, q_2, \dots, q_N)$ where $q_i \geq 0$

$$\mathrm{Prob}(S \subseteq X) \propto \det(K_{S,S}) = \Pi_{i\in S}\, q_i^2 \det(\Phi_{S,:}{}^T\Phi_{S,:})$$

$$f_{QD}(S, u) = 2 \log \mathrm{vol}(Q_{S,u}{}^{2\lambda}\, \Phi_{S,:}{}^T\Phi_{S,:}{}^{2(1-\lambda)}\, Q_{S,u}{}^{2\lambda})$$

Scientific challenges (*bis*)

- Incorporate a tradeoff between quality and diversity
  → use (a version of) QD decomposition as objective function
- Versatile enough to be applicable in all use cases
  → for any kernel and any feedback model, including *unknown* ones
- Not computationally too expensive (millions of items)
  → a time complexity with a dependency larger than O(N) will be too expensive

[1] Kulesza, A., & Taskar, B. (2010). Structured determinantal point processes. *Advances in neural information processing systems*, *23*.

My progress on the project since last time

Generalized quality-diversity objective function

Data

Φ_i : embedding for item i with $|\Phi_i|=1$

H_u : item history for user u

q : feedback model with positive values

B : the size of the batch of items to recommend at each time

K : kernel similarity

Hyperparameters

λ : weight of the relevance task

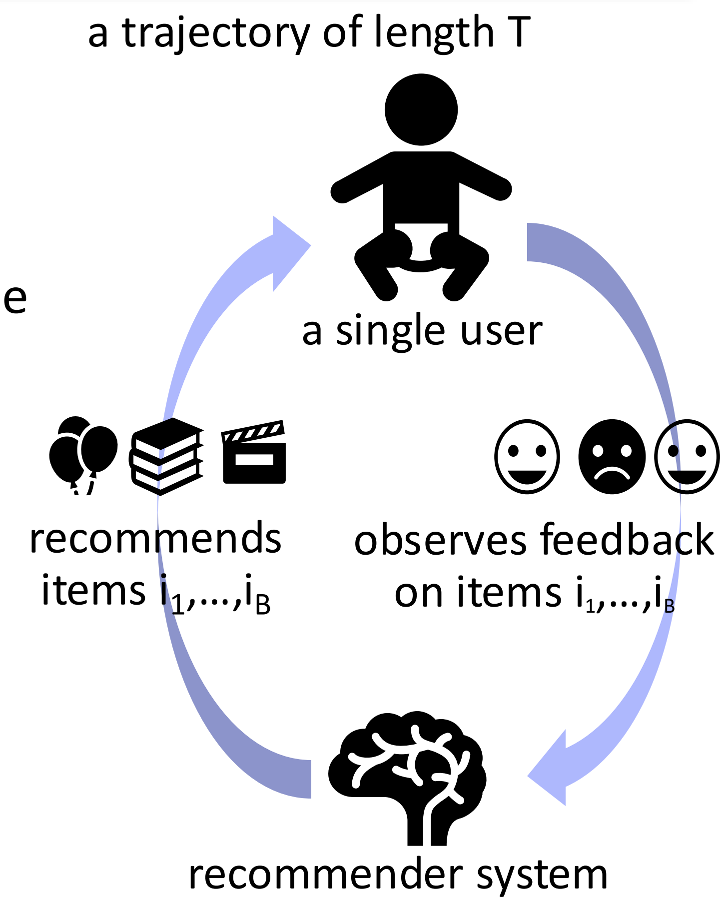
η : regularization factor

Objective function f for user u

For all S, $f_{obj}(S, u) = 2 \log \mathrm{vol}(Q_{S,u}^{2\lambda} M_{S|Hu,S|Hu}^{2(1-\lambda)} Q_{S,u}^{2\lambda})$

- $Q_{S,u} = \mathrm{diag}(\{\ q(i, u) \text{ for } i \text{ in } S\ \})$
- $M_{S|Hu,S|Hu} = K_{S,S} - K_{S,Hu}(K_{Hu,Hu}+\eta \mathrm{Id})^{-1}(K_{S,Hu})^T$

Conditioning:
dependency in $\Omega(|H|^2\log|H|)$



Generalized quality-diversity objective function with a fuzzy denuding/masking approach

Practical function f for user u

$f_{Prac}(S, u) = 2\log \mathrm{vol}(Q_{S,u}{}^{2\lambda} M_{S\cap Hu,S\cap Hu}{}^{2(1-\lambda)} Q_{S,u}{}^{2\lambda})$ where $M_{S\cap Hu,S\cap Hu} = K_{S,S} - K'_{S,S}$

and for all x, y in S, $K'_{\{x\},\{y\}} = \langle \Phi'(x), \Phi'(y) \rangle$ and $\Phi'(x) = 0$ if $\min_{\Psi \in Hu} |\Psi - x| > \alpha$, else $\Phi(x)$

dependency in $\Omega(k \log|H|)$
using a k-d tree for all of S
can be faster using GPUs

Synthetic Gaussian data, B=5, T=10, α=0, η=0.01, K linear, average across 10 iters (MAX)

	QD λ=0.5	Obj λ=0.5	Prac λ=0.5	QD λ=0.8	Obj λ=0.8	Prac λ=0.8	QD λ=0.1	Obj λ=0.1	Prac λ=0.1
Relevance	**0.586**	0.516	<u>0.559</u>	**0.571**	0.524	<u>0.558</u>	**0.586**	0.496	<u>0.560</u>
Intrabatch diversity	0.956	<u>0.971</u>	**0.976**	**0.990**	0.965	<u>0.971</u>	0.956	**0.961**	<u>0.960</u>
Interbatch diversity	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
%unique recomm	<u>0.100</u>	**1.000**	**1.000**	<u>0.100</u>	**1.000**	**1.000**	<u>0.100</u>	**1.000**	**1.000**

rewards in [0,1]

Volume collapses

Case where the reward model is known

Synthetic Gaussian data: reward $q(i,u) \propto \Phi_i^T c_u$

B=5, T=10, α=0, λ=0.5, η=0.01, K linear, average across 10 iters (MAX)

	Ɛ-greedy (Ɛ=λ)	**QD**	**Markov DPP [1]**	**Obj**	**Prac**
Relevance	0.536	0.592	0.524	<u>0.526</u>	**0.572**
Intrabatch diversity	0.582	0.956	<u>0.953</u>	**0.972**	**0.972**
Interbatch diversity	0.000	0.000	0.000	0.000	0.000
%unique recomm.	0.200	0.100	<u>0.600</u>	**1.000**	**1.000**

Baseline for diversity

Baseline for relevance

Baseline for integrating user history

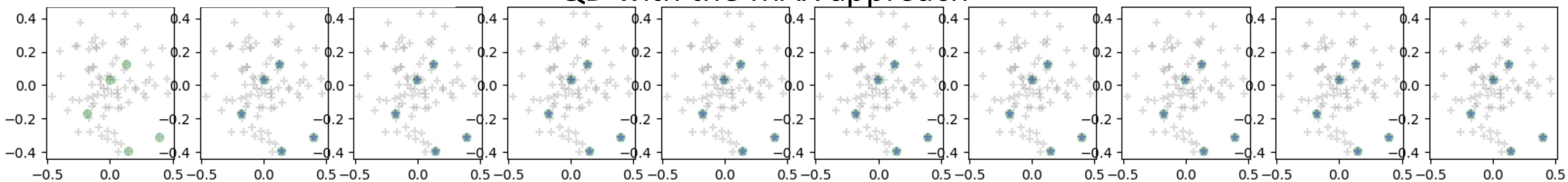
[1] Affandi, R. H., Kulesza, A., & Fox, E. B. (2012). Markov determinantal point processes. *arXiv preprint arXiv:1210.4850.*

Case where the reward model is known (SAMPLE)

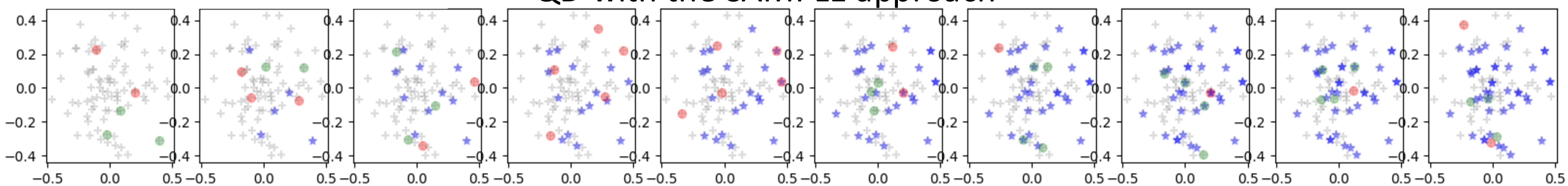
Synthetic Gaussian data: reward $q(i,u) \propto \Phi_i^T c_u$

2D PCA plot of selected points (positive ●, negative ●) at each point of the trajectory of length T=10
Previously selected ●

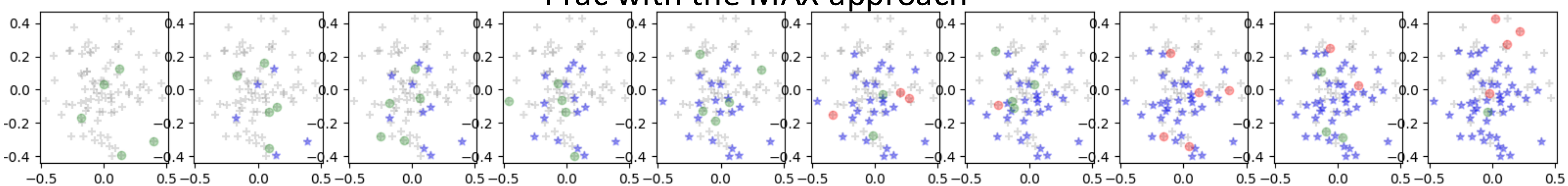
QD with the MAX approach



QD with the SAMPLE approach



Prac with the MAX approach



Case where the reward model is known

MovieLens data: reward q(i,u) := SVD(rating matrix)[i,u]

B=5, T=10, α=0, λ=0.5, η=0.01, K linear, average across 10 iters (MAX)

	Ɛ-greedy (Ɛ=λ)	**QD**	**Markov DPP [1]**	**Obj**	**Prac**
Relevance	4.581	4.577	<u>4.430</u>	4.421	**4.481**
Intrabatch diversity	0.261	0.761	0.543	<u>0.692</u>	**0.711**
Interbatch diversity	0.000	0.000	0.000	0.000	0.000
%unique recomm.	0.200	0.100	0.860	**1.000**	**1.000**
Runtime (sec.)	353 +- 35	510 +- 6	1,328 +- 12	**508 +- 3**	<u>515 +- 4</u>

rewards in [1,5]

Baseline for diversity

Baseline for relevance

Baseline for integrating user history

[1] Affandi, R. H., Kulesza, A., & Fox, E. B. (2012). Markov determinantal point processes. *arXiv preprint arXiv:1210.4850*.

Case where the reward model is known

PREDICT data: reward $q(i,u) := \mathrm{RandomForest}([\Phi_i, c_u])$

B=5, T=10, α=0, λ=0.5, η=0.01, K linear, average across 10 iters (MAX)

	ε-greedy (ε=λ)	QD	Markov DPP [1]	Obj	Prac
Relevance	1.000	1.000	1.000	1.000	1.000
Intrabatch diversity	0.637	1.000	0.881	**1.000**	<u>0.961</u>
Interbatch diversity	0.000	0.000	0.000	0.000	0.000
%unique recomm.	0.100	0.100	<u>0.967</u>	**1.000**	0.933
Runtime (sec.)	51 +- 1	52 +- 1	56 +- 1	**51 +- 0**	<u>52 +- 0</u>

rewards in {0,1} (overfitting)

Baseline for diversity

Baseline for relevance

Baseline for integrating user history

[1] Affandi, R. H., Kulesza, A., & Fox, E. B. (2012). Markov determinantal point processes. *arXiv preprint arXiv:1210.4850*.

Case where the reward model is unknown

Assumption: linear reward $q^*(i,u) := ([\Phi_i, c_u])^T\theta^*$

Cumulative regret at time T $\mathrm{Regret}(T) = \Sigma_{t<T} f_{obj}(S_t, u_t; q^*) - f_{obj}(S_t, u_t; q_t)$

At each time, we update the model $q_t(i,u) = [\Phi_i, H_u]^T\theta_t$ based on the received feedback and then use MAX on the Upper Confidence Bound of the empirical reward model q_t

(δ-uniform) Upper Confidence Bound

There is β such that for all t, i, u,

$$q^*(i,u) \leq q_t(i,u) + \beta(t,i)$$

with probability 1-δ

The baselines (almost) never learn because we forced the MAX approach

What happens if we allow SAMPLE?

Synthetic data : B=5, T=16, α=0, λ=0.5, η=0.01, K linear (MAX)

	LinUCB bandit [1]	LinOASM bandit [2]	Prac+UCB
Relevance	0.94	<u>0.95</u>	**0.99**
Intrabatch diversity	0.94	**1.00**	**1.00**
%unique recomm.	0.10	<u>0.10</u>	**1.00**
Regret(T)	16.15	<u>14.84</u>	**5.28**

Baseline for learning   Baseline for learning × diversity

[1] Li et al. "A contextual-bandit approach to personalized news article recommendation." *Proceedings of the 19th international conference on World wide web*. 2010.

[2] Gabillon et al. "Large-scale optimistic adaptive submodularity." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 28. No. 1. 2014.

Case where the reward model is unknown

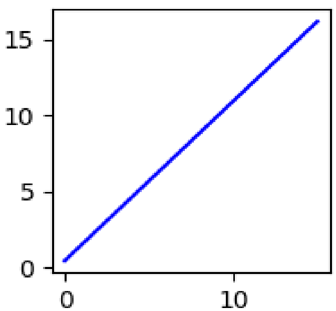
Assumption: linear reward $q^*(i,u) := ([\Phi_i, c_u])^T\theta^*$

Cumulative regret at time T $\mathrm{Regret}(T) = \Sigma_{t<T} f_{obj}(S_t, u_t; q^*) - f_{obj}(S_t, u_t; q_t)$

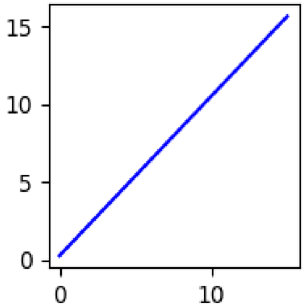
Comparing the regret curves for each bandit algorithm at T=16

LinUCB

MAX: cumulative regret 16.15

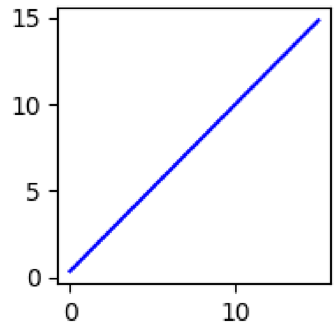


SAMPLE: cumulative regret 15.62

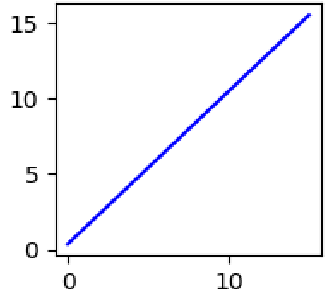


LinOASM

MAX: cumulative regret 14.84

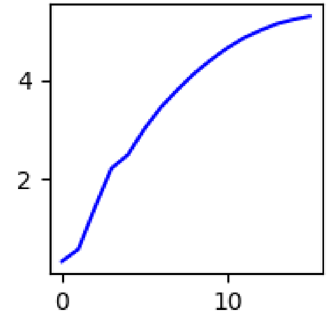


SAMPLE: cumulative regret 15.52



Prac+UCB

MAX: cumulative regret 5.28



The regret may decrease when we allow SAMPLE

But the baselines are still not learning (linear curve)...

... contrary to Prac+UCB (using MAX)

Adaptive quality-diversity tradeoff automatically tune λ in [0,1] to the user feedback

Goal: Finetune on the fly λ to a value which maximizes $\Sigma_{t<T} f(S_t, u_t ; \lambda)$

We can use any "good" learner (e.g., AdaHedge, EXP3, etc.) to learn the value of λ

Pseudo-code (for any bandit)

- Start with a value λ_0
- Initialize the online learner with the starting value
- For each time t up to T
  - Apply the bandit strategy for sampling S_t
  - Update the learner with the "gain" obtained with λ_t on S_t : $-\nabla_\lambda f(S_t, u_t ; \lambda_t)$
  - Obtain the new value λ_{t+1}
- Return λ_T and the best a posteriori $\lambda^* = \text{argmax}_\lambda \Sigma_{t<T} f(S_t, u_t ; \lambda)$

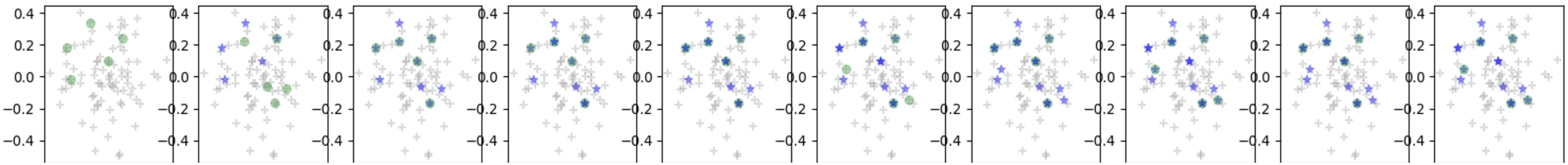
Adaptive quality-diversity tradeoff

Synthetic data: reward $q(i,u) \propto \Phi_i^T c_u$

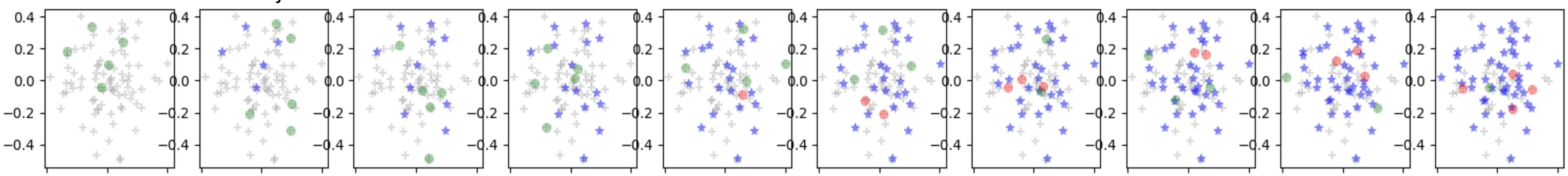
2D PCA plot of selected points (positive ●, negative ●) at each point of the trajectory of length T=10
Previously selected ●

Starting from λ_0 =0.3

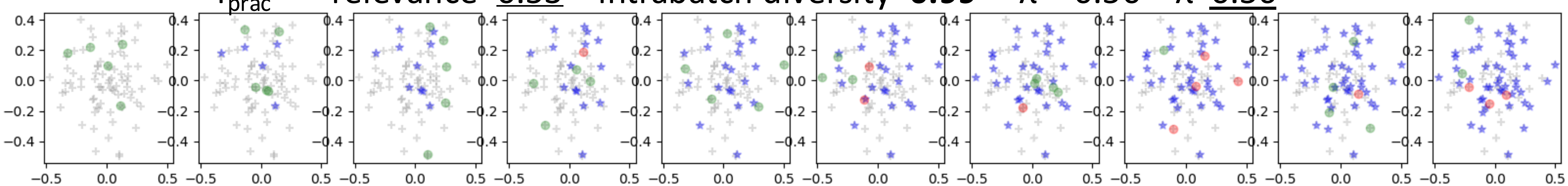
f_{QD} relevance=**0.57** intrabatch diversity=**0.99** λ*=0.49 λ=0.59



f_{obj} relevance=0.52 intrabatch diversity=<u>0.98</u> λ*=0.50 λ=**0.51**



f_{prac} relevance=<u>0.53</u> intrabatch diversity=**0.99** λ*=0.50 λ=<u>0.56</u>



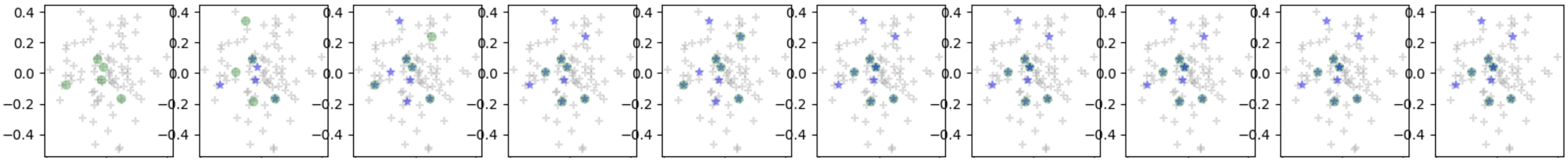
Adaptive quality-diversity tradeoff

Synthetic data: reward $q(i,u) \propto \Phi_i^T c_u$

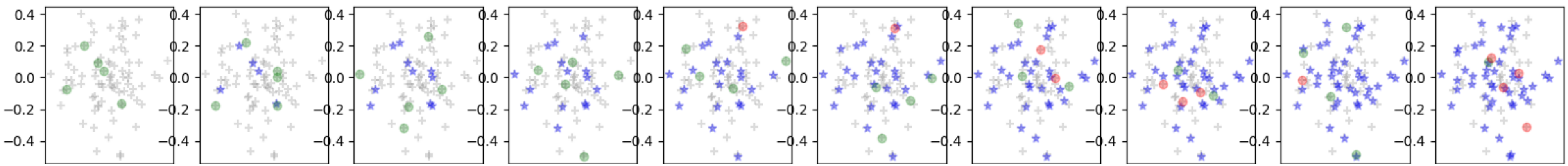
2D PCA plot of selected points (positive ●, negative ●) at each point of the trajectory of length T=10
Previously selected ●

Starting from λ_0 =0.1

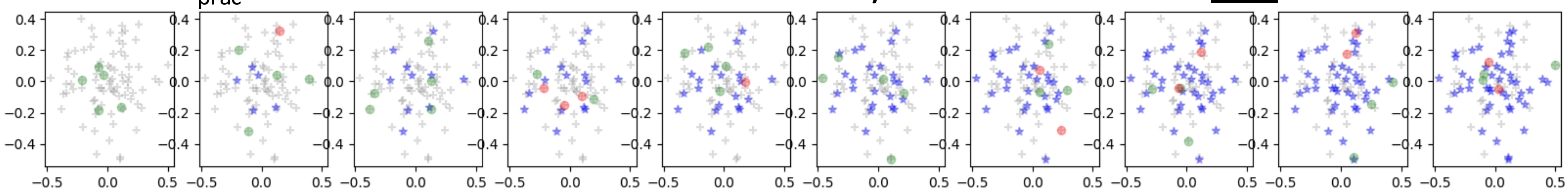
f_{QD} relevance=**0.58** intrabatch diversity=**0.99** λ^*=0.47 λ=0.61



f_{obj} relevance=<u>0.54</u> intrabatch diversity=<u>0.98</u> λ^*=0.46 λ=**0.51**



f_{prac} relevance=0.53 intrabatch diversity=**0.99** λ^*=0.46 λ=<u>0.57</u>



What happens next

1. Find a better measure of interbatch diversity
2. Solve the overfitting problem in the PREDICT data set
3. Optimize the implementation of Prac(+UCB) for millions of items
4. Test in "real conditions": alternating users
5. Derive theoretical guarantees on relevance/diversity