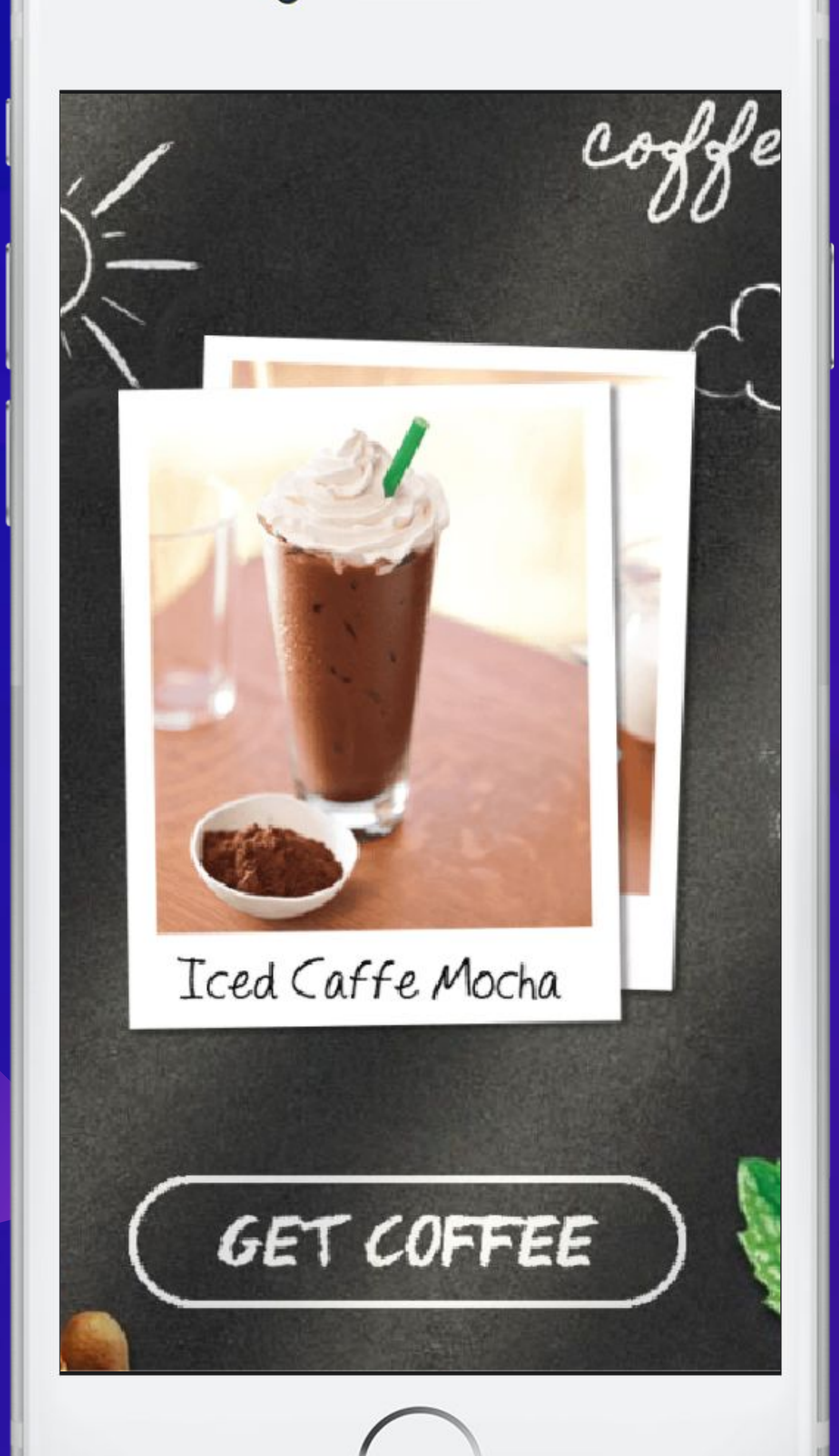
Real-time Optimization of Advertising Content

Tom Vodopivec

IADS Analytics, Data Science & Decision Making Summer School 2022
2022-08-01


coffe
Cinnamon Dolce Latte
GET COFFEE


coffe
Iced Caffe Mocha
GET COFFEE

Choosing the best variant

Difficult for marketers to predict which variant will perform better

Depends on specific environment

Might change with time

Choosing the best variant

Difficult for marketers to predict which variant will perform better

Depends on specific environment

Might change with time

Possible solution: automated real-time optimization

Business goal

Develop methodology to achieve as **high performance** as possible and **measure it with confidence** to show it to clients.

Project goals

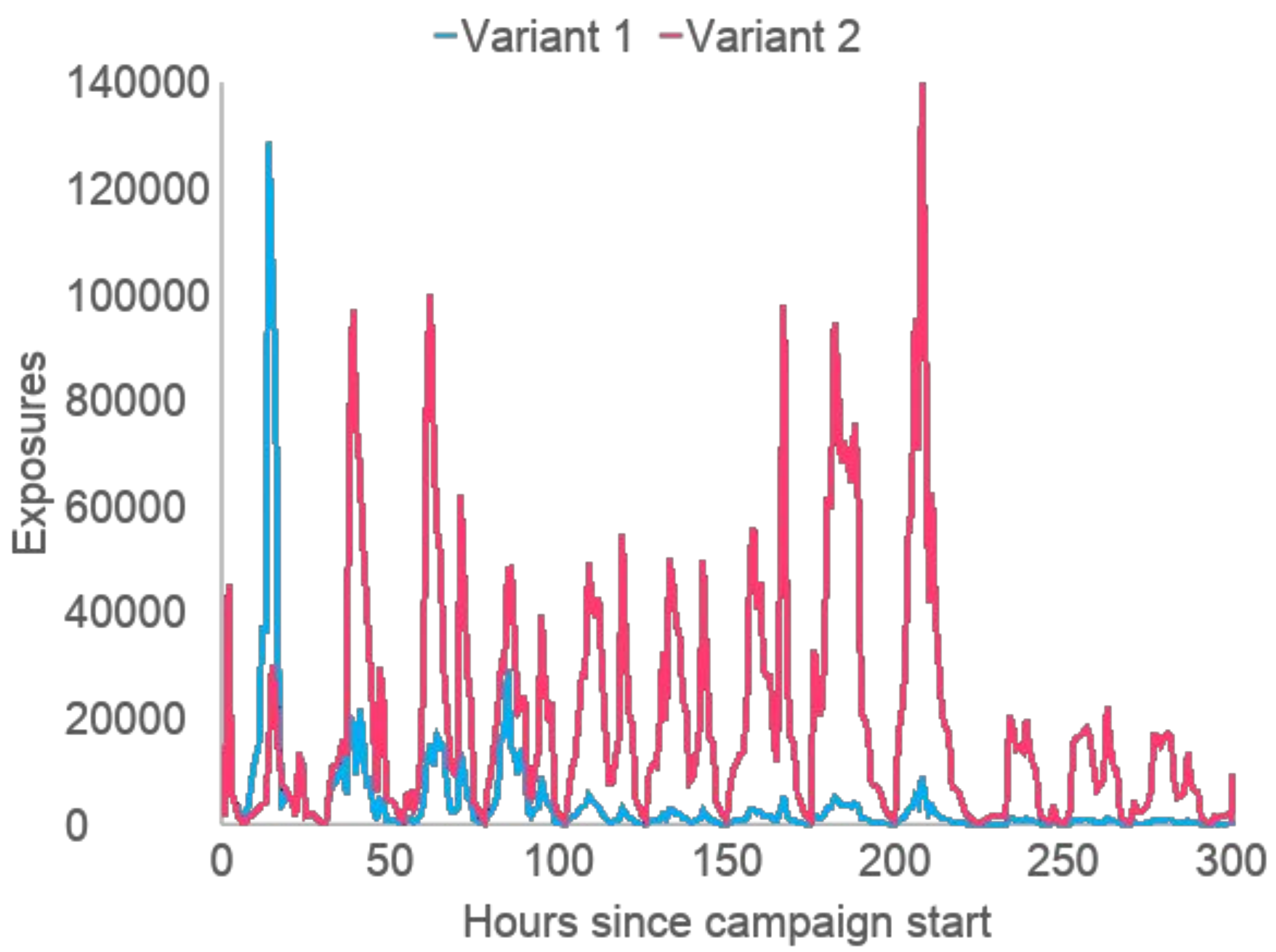
Measure the performance of the DCO system

Estimate theoretical potential of DCO optimization in Celtra campaigns

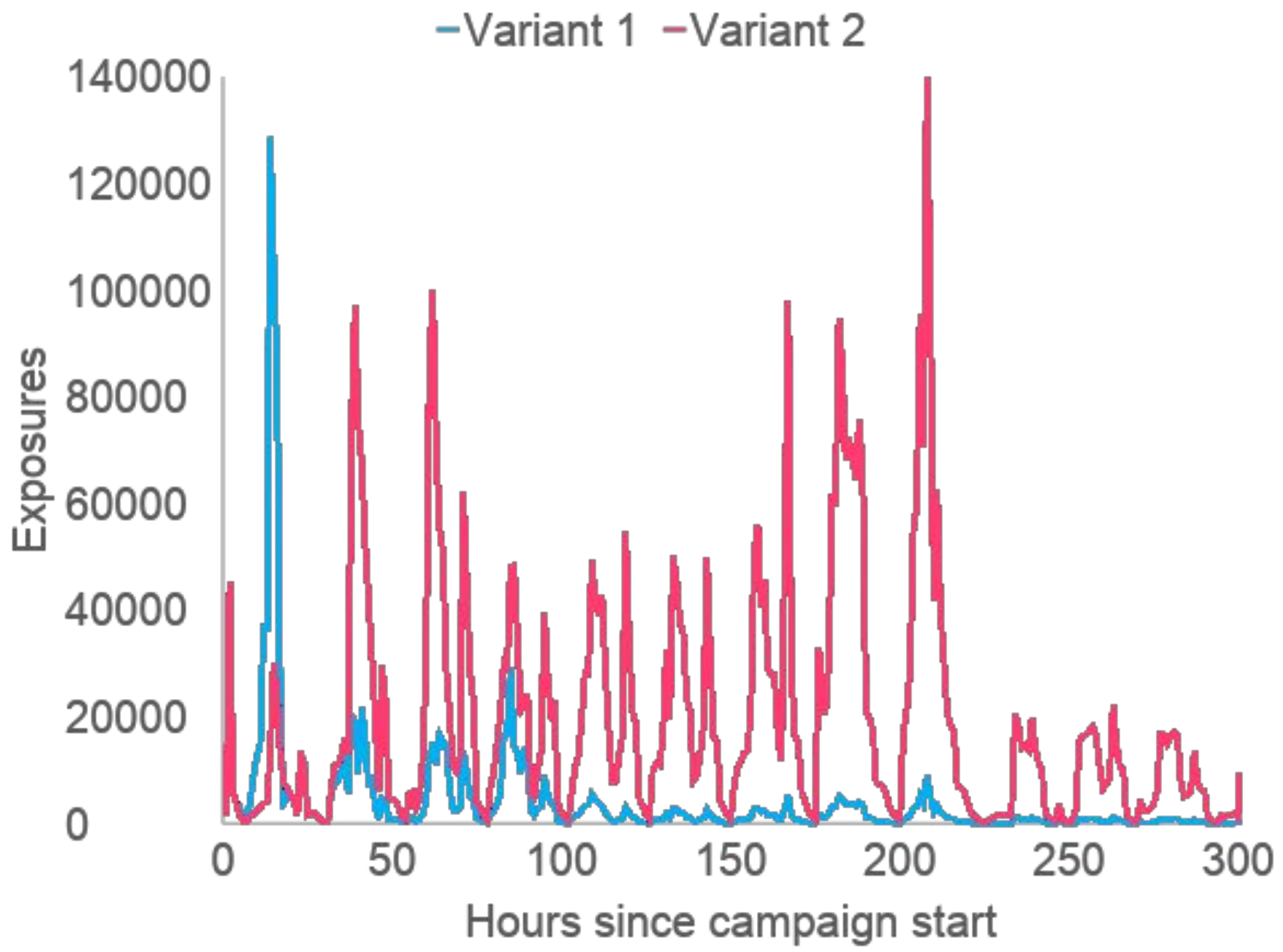
Estimate how much of the theoretical potential can algorithms achieve

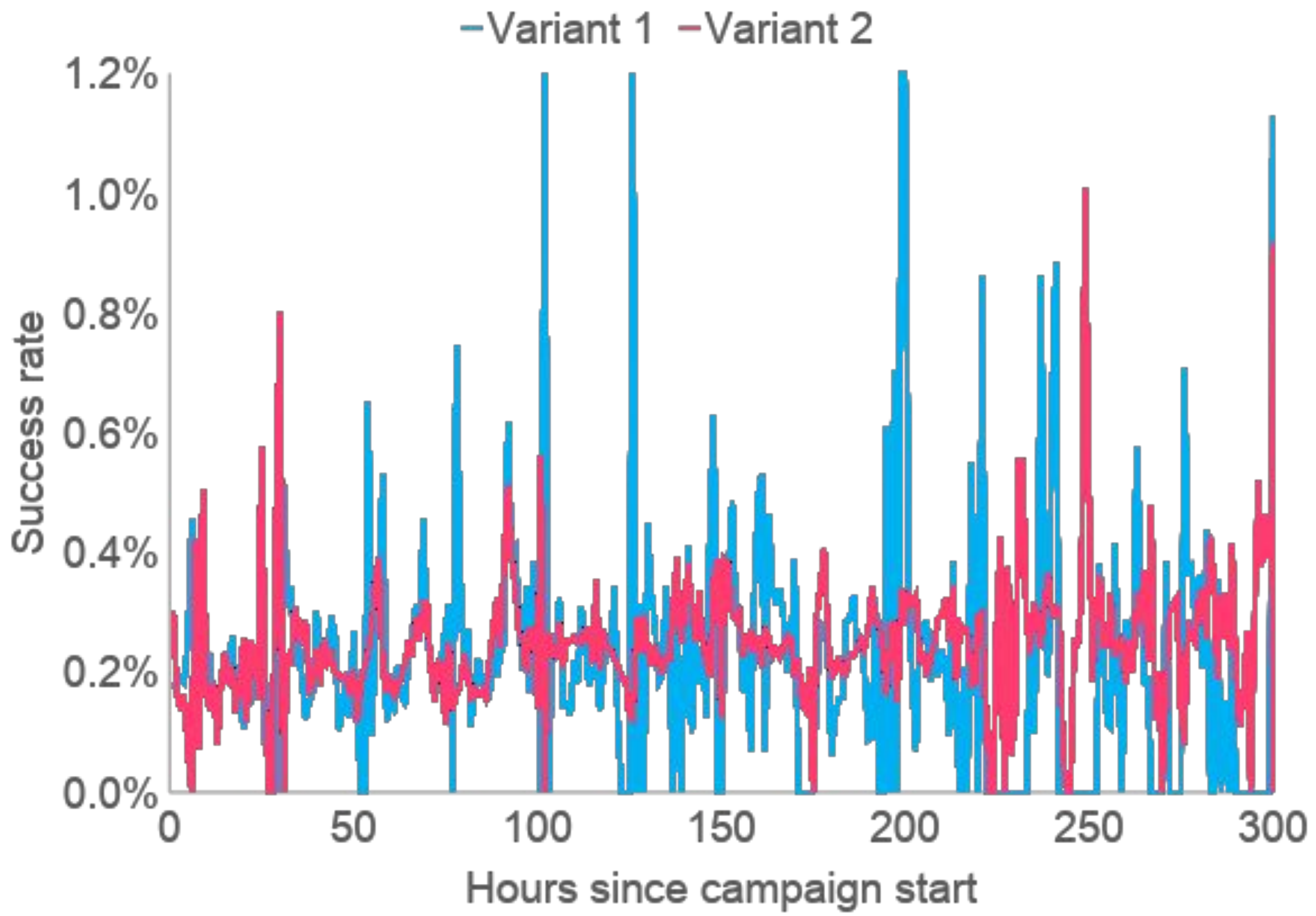
Improve performance of the algorithms = research and develop new ones

An advertising campaign



An advertising campaign





Evaluating the performance

Measure of performance – *lift*

Successes of optimization algorithm / successes of random algorithm

Measure of potential – lift of an *oracle* optimization algorithm

Challenges and solutions

Address the ***exploration-exploitation dilemma*** > bandit algorithms

Solution: ε-greedy, Thompson sampling, UCB algorithms

Real-time optimization causes ***selection bias*** > compute only from unbiased data

Need a control set for unbiased analysis

Solution: Three-set sampling

Challenges and solutions

Technical limitations: feedback **delayed** and in **batches**

Solutions: prediction + simulated feedback

Non-stationarity

Solution: Detection of abrupt changes in trends

Solution: Forgetting

Solution: Two-memory structure

Challenges and solutions

<u>Obtaining samples is expensive</u>

Low success rates - **noisy data**, few samples

Solution: Estimation of confidence bounds (Fieller‘s theorem)

How to **compare different** algorithms??

Solution: Creation of artificial campaigns as an approximation of real campaigns

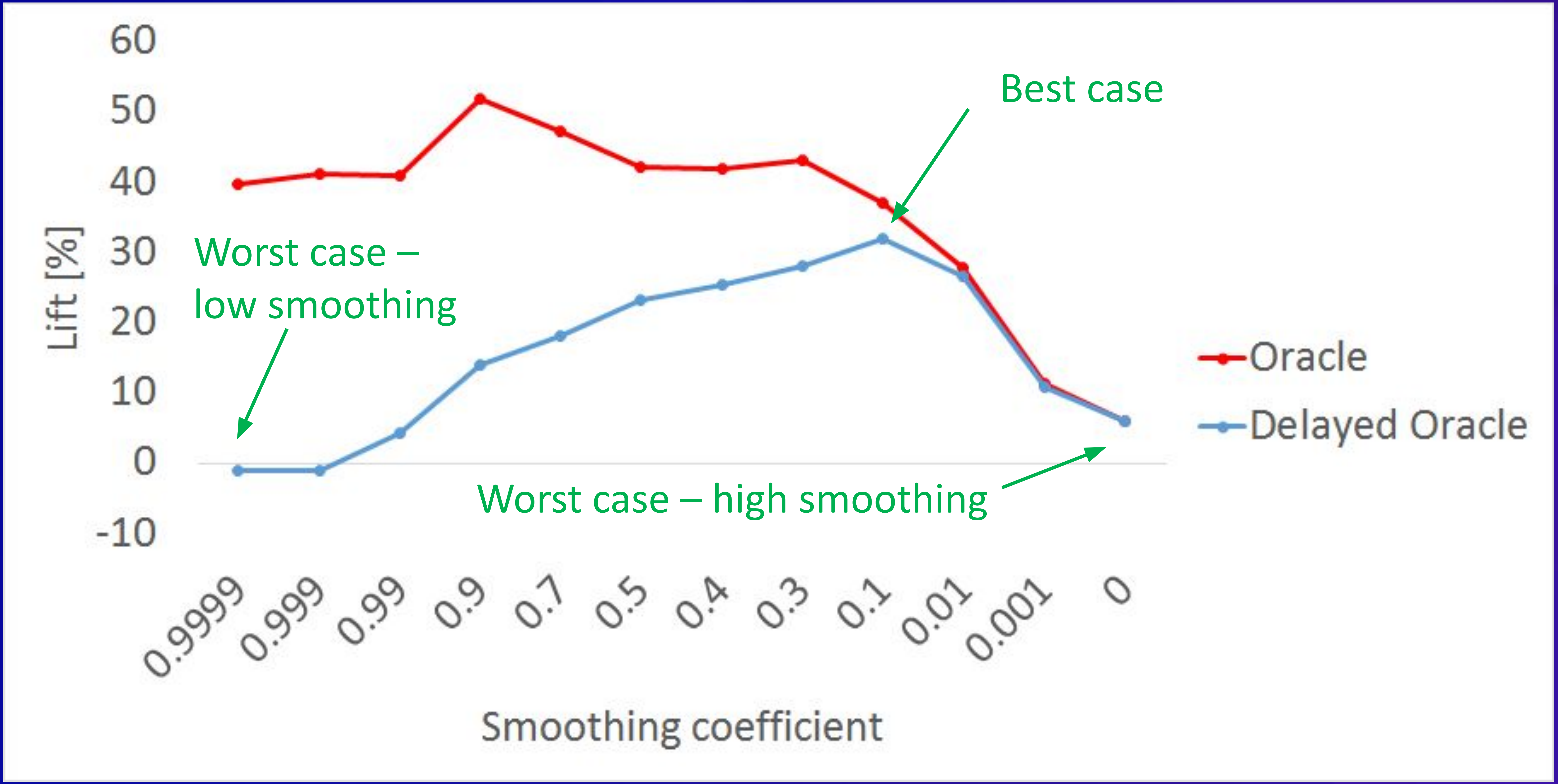
Artificial campaigns

Correction coefficient

Smoothing coefficient

Best-case scenario

Two worst-case scenarios


60
50
40
30
20
10
0
-10
Lift [%]
Worst case –
low smoothing
Best case
Worst case – high smoothing
Oracle
Delayed Oracle
0.9999
0.999
0.99
0.9
0.7
0.5
0.4
0.3
0.1
0.01
0.001
0
Smoothing coefficient

Data and experiments

Hundreds of real campaigns

- Up to 10 variants
- Up to 10 million exposures
- Up to 6 months of duration

Best-case and worst-case sets of artificial campaigns

Multiple repeats on each campaign for each algorithm

Results: potential in campaigns

How much potential is there?

- 26% – 55% of campaigns have at least 5% potential lift
- 18% – 53% of campaigns have at least 10% potential lift
- 18% – 34% of campaigns have at least 20% potential lift

Results: potential in campaigns

How much potential is there?

- 26% – 55% of campaigns have at least 5% potential lift
- 18% – 53% of campaigns have at least 10% potential lift
- 18% – 34% of campaigns have at least 20% potential lift

Roughly one in four campaigns has meaningful potential

Results: efficiency of optimization

In how many campaigns with significant potential can the algorithms achieve at least 5% lift?

Thompson sampling	in 25% – 39% of campaigns
UCB	in 30% – 44% of campaigns

Results: efficiency of optimization

In how many campaigns with significant potential can the algorithms achieve at least 5% lift?

Thompson sampling	in 25% – 39% of campaigns
UCB	in 30% – 44% of campaigns
Augmented Thompson sampling	in 36% – 55% of campaigns
Augmented UCB	in 34% – 59% of campaigns

Results: efficiency of optimization

In how many campaigns with significant potential can the algorithms achieve at least 5% lift?

Thompson sampling	in 25% – 39% of campaigns
UCB	in 30% – 44% of campaigns
Augmented Thompson sampling	in 36% – 55% of campaigns
Augmented UCB	in 34% – 59% of campaigns

The algorithms can exploit roughly half of the potential

Algorithm	Ratio [%] of artificial campaigns with lift above		
	5 %	10 %	20 %
Original TS	39	27	16
+ periodic forgetting	48	32	23
+ two-memory	51	37	29
+ prediction	55	36	30
+ change-point detection	55	36	30

Summary

Metric for performance of content-optimization algorithms

Metric for potential in advertising campaigns

Methodology for measuring algorithm performance

Evaluation of algorithms

Improvements for algorithms

Future and Conclusion

Potential improvement to algorithms' performance

- Dual-layer UCB
- Budget-limited bandits
- Optimal ratio of three sampling sets
- Signal processing and time-series prediction

Can't optimise if there is nothing to optimize

- Correlate campaign features with campaign potential (learn what to do to have exploitable potential)
- Inform and educate designers who create advertising campaigns

Impact on economy could be massive

References

Our conference paper: Vodopivec et al., Real-time Content Optimization in Digital Advertisement, 2017, pages 97-100

Fieller's theorem: Fieller, Some Problems in Interval Estimation, 1954

Variance and covariance estimation: Moineddin et al., On the Location Quotient Confidence Interval, 2003

Two-set sampling (un-biasing the data): Xu et al., Estimation Bias in Multi-Armed Bandit Algorithms for Search Advertising, 2013

Bandit algorithms: Kuleshov and Precup, Algorithms for multi-armed bandit problems, 2014

Change detection methods: Sebastiao and Gama, A study on change detection methods, 2009

Budget-limited multi-armed bandit algorithms: Tran-Thanh et al., Knapsack based optimal policies for budget-limited multi-armed bandits, 2012

4. Improve performance

List algorithms we tested

Near-zero success rate (can’t do much) > tell client that campaigns need to have enough samples (we can estimate how much)

Batch updates, algorithm less optimal due to fixed action-selection distribution for whole hour (can mitigate)

- simulate reward after each selection (or use dedicated bandit algorithms, however issue is also that “budget” for the hour is unknown)

Near-zero success rate (can’t do much) > tell client that campaigns need to have enough samples (we can estimate how much) -> need 10k to 10M impressions (depends on success rate and vairant) for oracle lift lower bound above 0%

Batch updates, algorithm less optimal due to fixed action-selection distribution for whole hour (can mitigate)

- simulate reward after each selection (or use dedicated bandit algorithms, however issue is also that “budget” for the hour is unknown) -> a set of “simualted” samples that gets replaced by “real” samples at each hour

Delayed updates (can mitigate) > time-series prediction (for generating the simulated samples between each batch update)

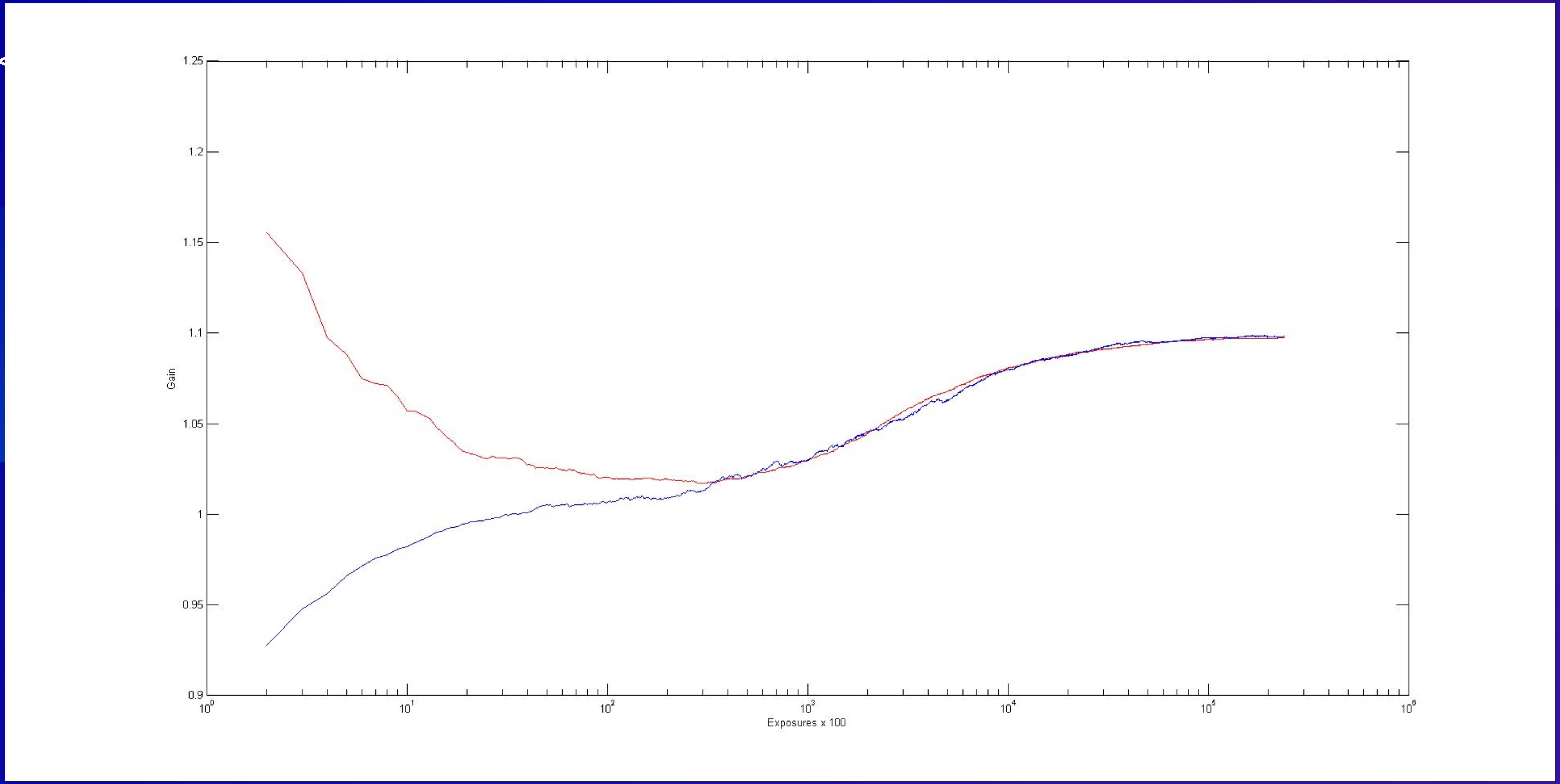
Non-stationarity - overfitting to same scenario (e-greedy very poor here) (can mitigate)

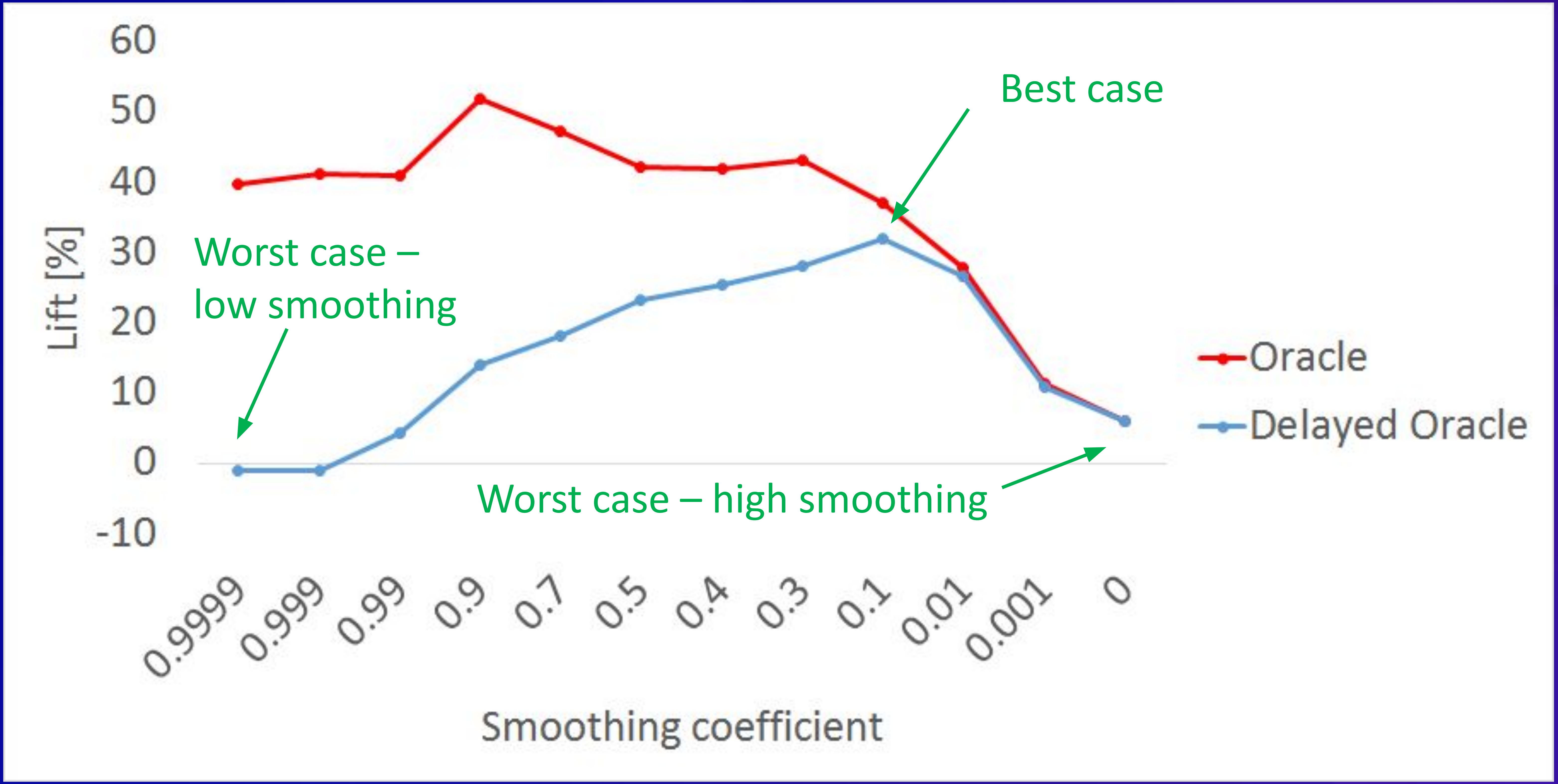
- forgetting (params: strength and frequency)
  + periodic (on time, exposures, or successes)
  + statistical change-point detection (Page-Hinkley test)
- two-memory structure (long-term and short-term / permanent and transient), forgetting for each + weight of each

Need for a control set (trade-off between “achieved” lift and “measurable” lift) (can mitigate)

- three-set sampling (evaluation, learning, random)

Generating artificial data




60
50
40
30
20
10
0
-10
Lift [%]
Best case
Worst case –
low smoothing
Worst case – high smoothing
Oracle
Delayed Oracle
0.9999
0.999
0.99
0.9
0.7
0.5
0.4
0.3
0.1
0.01
0.001
0
Smoothing coefficient

Artificial datasets

Dataset	Number of campaigns with oracle lift above 10 %
Best case	20/38 (53 %)
Low smoothing	12/38 (32 %)
High smoothing	7/38 (18 %)

Results: TS vs. UCB on best-case dataset

Algorithm	Ratio [%] of artificial campaigns with lift above		
	5 %	10 %	20 %
Original TS	39	27	16
Original UCB1-Tuned	44	35	19
Improved TS	55	36	30
Improved UCB1-Tuned	59	34	30

Results: Performance on all datasets

Dataset	Ratio [%] of artificial campaigns with lift above		
	5 %	10 %	20 %
Best case	59	36	30
Low smoothing	44	30	12
High smoothing	36	27	16

(Results given for best configurations of improvements)

Results: Exploited potential

Dataset	Algorithms‘ lift relative to delayed oracle [%]	
	Average	Median
Best case	25.3	20.0
Low smoothing	52.3	16.0
High smoothing	34.3	37.0

(Results given for best configurations of improvements)

Summary

Selection method	Ratio [%] of valid real campaigns with lift above		
	5 %	10 %	20 %
Oracle	26–55	18–53	8–34
Celtra various algorithms	7–12	6–9	3–8
Basic TS	13–20	9–14	6–8
Improved TS/UCB	19–31	14–18	8–16

Best algorithms exploit 25–50 % of potential