个性化排序策略研究

Meggyzhang
微信广告部/引擎策略研发中心/效果引擎开发组

问题简述

➢ 排序公式现状

- 打分项较多，未分析重要性
- 打分权重根据经验设置，不能和优化目标结合
- 不能根据用户特征，个性化打分权重

$$\text{score} = \text{ecpm} + \text{user\_engagement} + \text{user\_social\_engagement}$$

$$\text{social\_engagement} + \text{circle\_engagement} - \text{nfb}$$

$$\begin{cases} -\ \text{market\_quality} - \text{bandwidth\_cost} & \text{for bidding ads} \\ +\ \text{operational\_delta} & \text{for contact ads} \end{cases}$$

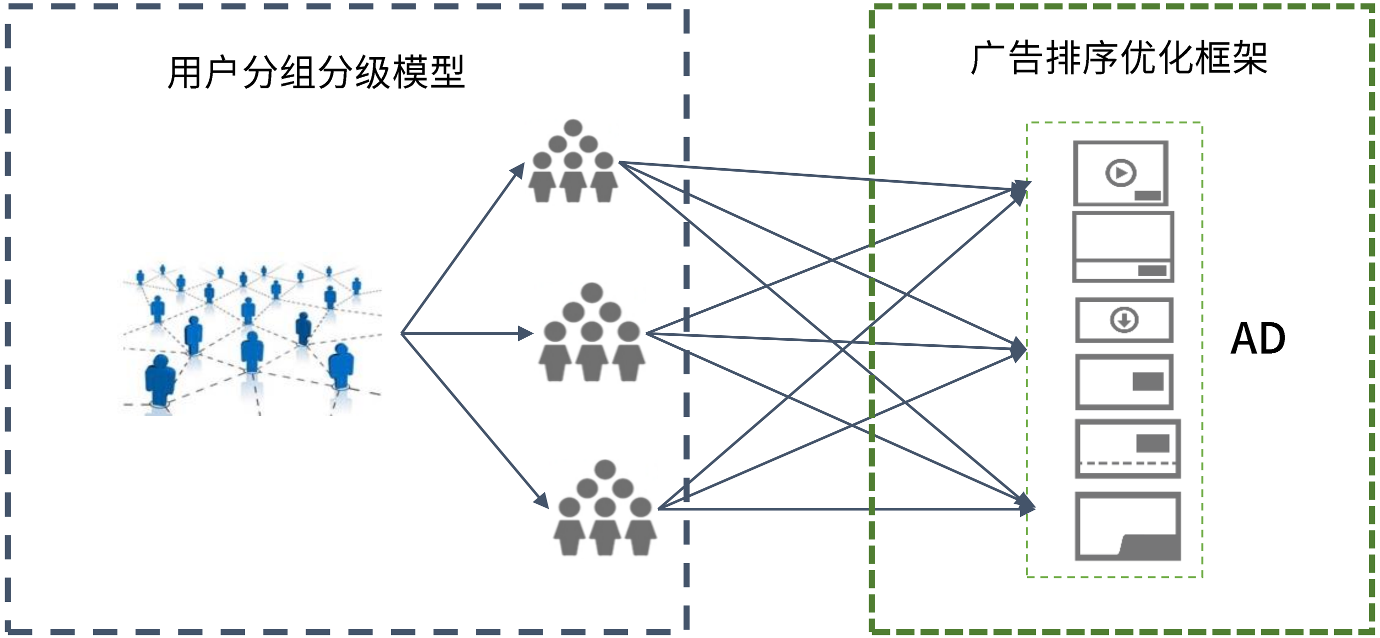
$$\text{ranking\_score} = \text{score} * \text{pacing\_alpha}$$

➢ 优化方向

- 个性化排序优化
- 达到不同用户分组中平台收入和用户体验的多目标优化
- 使排序公式和流量择优结合

问题简述

- 整体目标：通过个性化排序优化，达到不同用户分组中平台收入和用户体验的多目标优化
  - 用户分组分级模型：用户活跃度分组，用户体验敏感度分级
  - 广告排序优化框架：根据分组分级信息，输入预期收入目标，效果目标，播放目标等，输出个性化动态排序公式



问题简述

- 效果目标:
  - 前期只考虑用户体验核心项，后期需要用quality统一考虑更多细节。
  - 合约广告quality需要考虑社交因素，圈子因素，额外附加项，以及ecpm效果提升项
  - 竞价广告quality需要考虑market quality，视频广告带宽因素，用户广告相关性等

- 播放目标:
  - 合约，竞价占比
  - 不同类目占比
  - 竞价广告播放速度等

- 通过优化问题达到全局的控制。并可结合个性化排序的打分，来做流量择优。

问题建模

- 假设单位时间内有K个队列，队列k中广告i打分胜出

$$i_k = argmax_j \{S_{k,j}\}$$

- 多目标优化问题：
  - 目标函数(Objective)

$$\underset{\{S_{k,j}\}}{\text{maximize}} \quad \frac{1}{K}\sum_k ecpm_{i_k}$$

  - 约束条件(Constraints)

用户体验约束

$$\frac{1}{K}\sum_k pctr_{i_k} \geq pctr_0$$

$$\frac{1}{K}\sum_k engage_{i_k} \geq engage_0$$

$$\frac{1}{K}\sum_k nfbr_{i_i} \leq nfbr_0$$

类目占比约束

$$\frac{1}{K}\sum_k \mathbf{1}\{i_k \in \mathbb{C}_{contract}\} \geq r_l$$

$$\frac{1}{K}\sum_k \mathbf{1}\{i_k \in \mathbb{C}_{contract}\} \leq r_u$$

$$\vdots$$

问题建模

- 由于indicator的存在，该问题为**组合优化**或mixed integer programming问题
- 维度为$K \times J$，一般是指数复杂度， NP hard

$$\underset{\{S_{k,j}\}}{\text{maximize}} \quad \frac{1}{K}\sum_k ecpm_{i_k} \quad \Rightarrow \quad \frac{1}{K}\sum_k ecpm_{i_k} = \frac{1}{K}\sum_k ecpm_{k,j}\mathbf{1}\{j = argmax_j(S_{k,j})\}$$

$$\text{subject to} \quad \frac{1}{K}\sum_k pctr_{i_k} \geq pctr_0 \quad \Rightarrow \quad \frac{1}{K}\sum_k pctr_{i_k} = \frac{1}{K}\sum_k pctr_{k,j}\mathbf{1}\{j = argmax_j(S_{k,j})\}$$

$$\frac{1}{K}\sum_k engage_{i_k} \geq engage_0$$

$$\frac{1}{K}\sum_k nfbr_{i_i} \leq nfbr_0$$

$$\frac{1}{K}\sum_k \mathbf{1}\{i_k \in \mathbb{C}_{contract}\} \geq r_l$$

$$\frac{1}{K}\sum_k \mathbf{1}\{i_k \in \mathbb{C}_{contract}\} \leq r_u$$

$$\vdots$$

- Nonconvex
- Non-differentiable

Literature

➢ 非凸优化算法

- 模拟退火

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, 1983.

- 遗传算法

[2] Mitchell and Melanie, *An introduction to genetic algorithms.* MIT press, 1998.

- 凸优化松弛

[3] Daniel P. Palomar and Yonina C. Eldar, Eds., *Convex Optimization in Signal Processing and Communications*, Cambridge University Press, 2009.

[4] Daniel P. Palomar and Mung Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, 2006.

[5] Mengyi Zhang, Francisco Rubio, and Daniel P. Palomar, "Improved calibration of high-dimensional precision matrices," *IEEE Transactions on Signal Processing*,vol. 61, no. 6, 2012.

[6] Yang Yang, Marius Pesavento, Mengyi Zhang, and Daniel P. Palomar, "An online parallel algorithm for recursive estimation of sparse signals," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 3, 2016.

Literature

➢ 非凸优化算法

- 强化学习相关

[7] L. Zhang, T. Hu, Y. Min, G. Wu, J. Zhang, P. Feng, P. Gong, and J. Ye, "A taxi order dispatch model based on combinatorial optimization," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.

[8] Z. Xu, Z. Li, Q. Guan, D. Zhang, Q. Li, J. Nan, C. Liu, W. Bian, and J. Ye, "Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

➢ Ranking 相关

[9] H. Zhu, J. Jin, C. Tan, F. Pan, Y. Zeng, H. Li, and K. Gai, "Optimized cost per click in taobao display advertising," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.

[10] Y. Hu, Q. Da, A. Zeng, Y. Yu, and Y. Xu, "Reinforcement learning to rank in e-commerce search engine: Formalization, analysis, and application," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

[11] D. Agarwal, B.-C. Chen, P. Elango, and X. Wang, " Click shaping to optimize multiple objectives," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM*, 2011.

[12] D. Agarwal, K. Basu, S. Ghosh, Y. Xuan, Y. Yang, and L. Zhang, "Online parameter selection for web-based ranking problems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

一期算法 – 局部最优解

- 原始问题为组合优化或mixed integer programming
- 第一期算法，将目标函数和约束条件relax成differentiable non-convex problem，通过梯度法(Gradient method)求解
- 在tensorflow框架下实现

- Gradient Method [13]
  - 对Convex problem，收敛到全局最优解
  - 对non-convex problem, 收敛到局部最优解
  - 广泛应用，如深度神经网络中back propagation
  - 由于高维特性，DNN中常收敛到saddle point

[13] S. Boyd and L. Vandenberghe, Convex optimization. Cambridge university, press, 2004.

一期算法 – Gradient Method

- 将constraint放到objective,将整个问题转化为unconstrained problem

$$f(S_{k,j}) = \frac{1}{K}\sum_k ecpm_{i_k} - Relu(\frac{1}{K}\sum_k pctr_{i_k} - pctr_0) - Relu(\frac{1}{K}\sum_k engage_{i_k} - engage_0)$$
$$- Relu(nfbr_0 - \frac{1}{K}\sum_k nfbr_{i_i}) - Relu(\frac{1}{K}\sum_k \mathbf{1}\{i_k \in \mathbb{C}\} - r_l)$$

- 将indicator function用softmax relax

$$\frac{1}{K}\sum_k ecpm_{i_k} = \frac{1}{K}\sum_k ecpm_{k,j}\mathbf{1}\{j = argmax_j(S_{k,j})\}$$

$s_{k,j}$：打分项，可包括ecpm,pctr,engage,nfbr,类目等

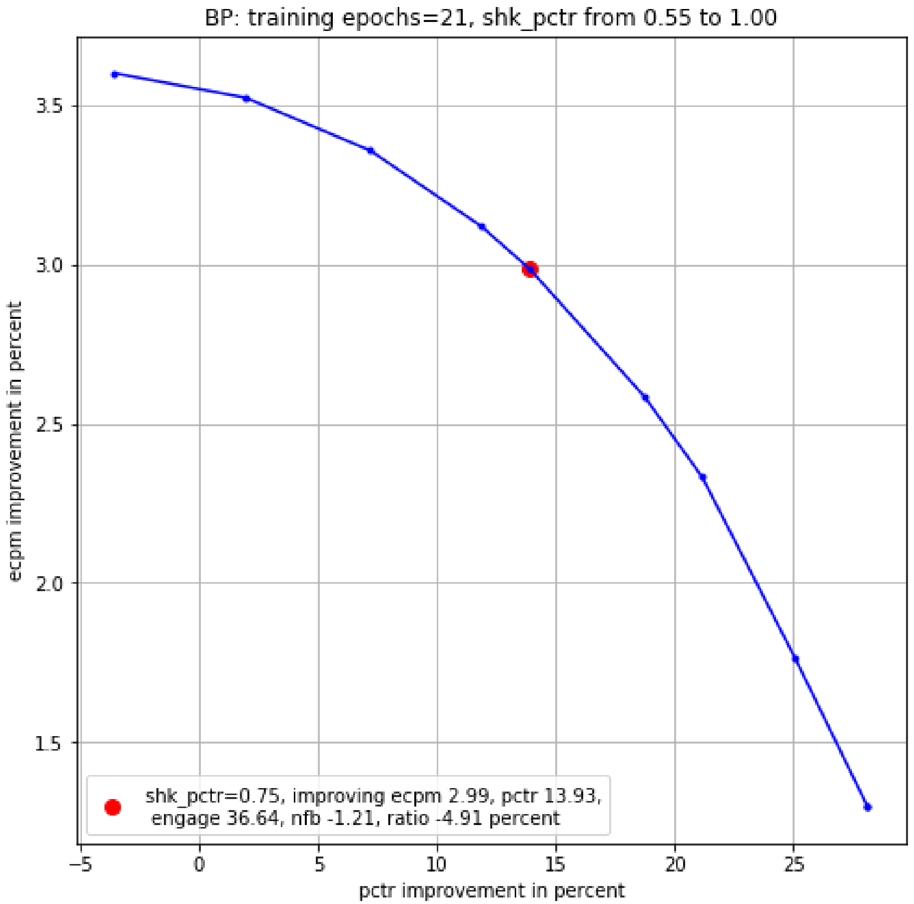
$$S_{k,j} = \alpha_{k,j}\sum w^{(l)} s_{k,j}^{(l)}$$

$$\frac{1}{K}\sum_k ecpm_{k,j} softmax(S_{k,j})$$

一期算法 – Gradient Method

- $loss_{pctr} = Relu(f_{pctr} * \frac{1}{K}\sum_k pctr_{k,0} - \frac{1}{K}\sum_{k,j} pctr_{k,j} * softmax(\alpha_{k,j} * s_{k,j}))$

- 特点
  - 基于tensorflow梯度求解方便,
  - 收敛快
  - Relaxation有时不精确
  - Shrinking factors不易调整
- 超参数学习
  - 寻找ecpm提升和用户体验提升的tradeoff



一期算法 – Simulated Annealing

- 模拟退火算法是解非凸优化问题的一种经典算法[1]，所得解依概率收敛(converge in probability)到全局最优解
- 最初的iteration，以一定概率跳出局部最优值，概率随着iteration逐渐趋近0

Algorithm 1 Simulated Annealing

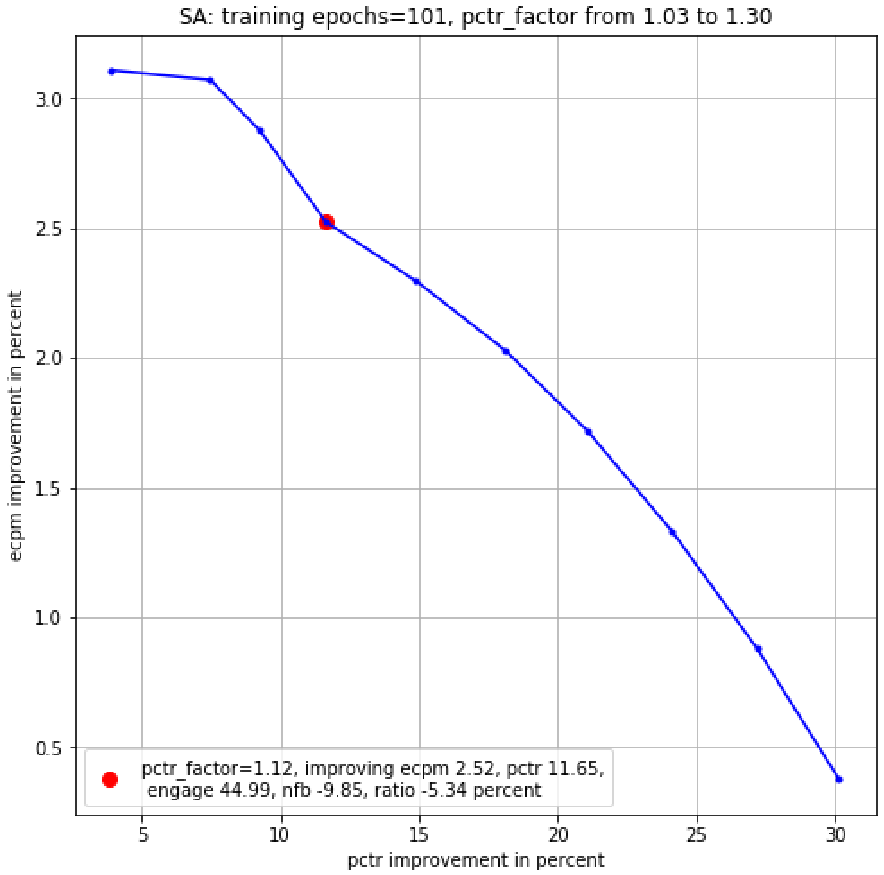
1. Initialize temperature T and ratio T_{ratio}
2. For $t = 0, 1, \ldots, N_{epochs} - 1$:
   (a) Compute objective and constraints
   (b) If constraints are not satisfied, randomly update w until satisfied
   (c) Compare f_{new} and f_{old}, update with probability $p = e^{(f_{new} - f_{old})/T}$
3. $T = T * T_{ratio}$

[1] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated annealing,” *Science*, vol. 220, no. 4598, 1983.

一期算法 – Simulated Annealing

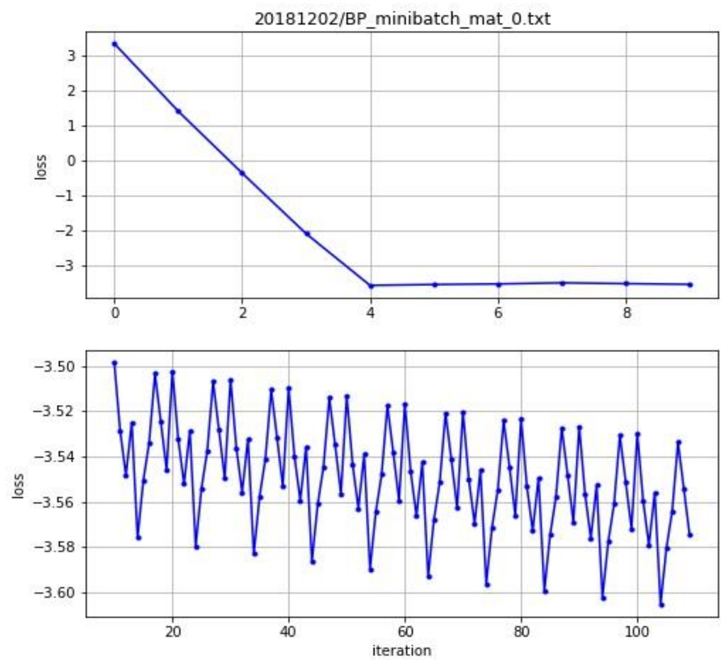
- 收敛相对慢
- 变量更新时基本为random，当constraint不满足时，可能会来回振荡

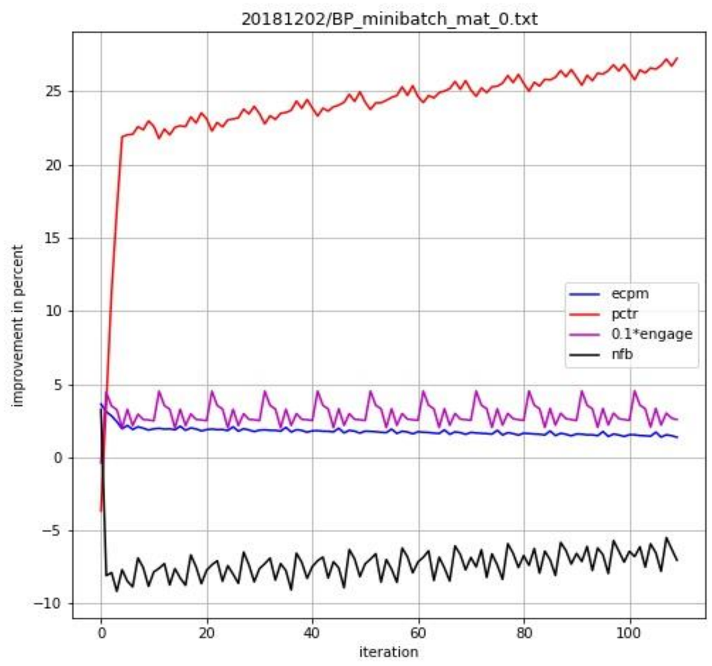
- Objective和contraints不存在relaxation,精确解
- Shrinking factors直接根据业务目标设置



一期算法 – 离线结果

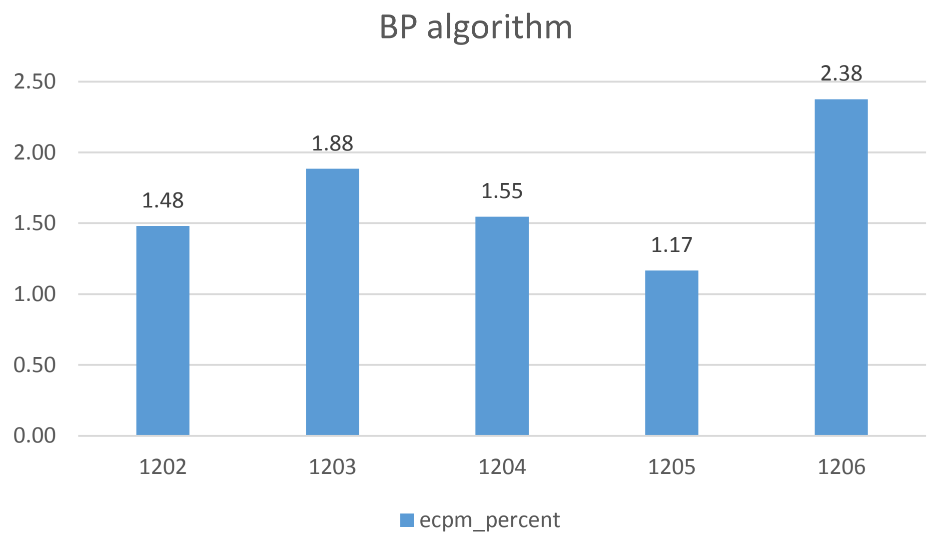
- Gradient Method
  - 训练集
  - 对于一天数据，采用mini-batch方式。对小时数据抽样组合，保证每个batch分布和一天分布基本相同。
  - 目标函数(loss) 振荡下降

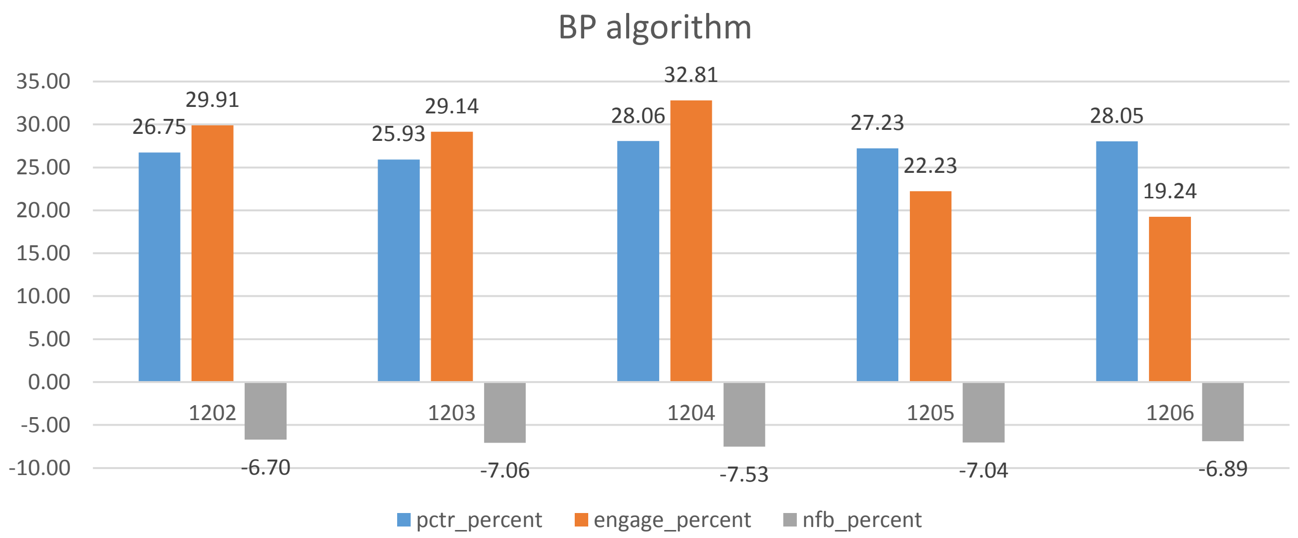




一期算法 – 离线结果

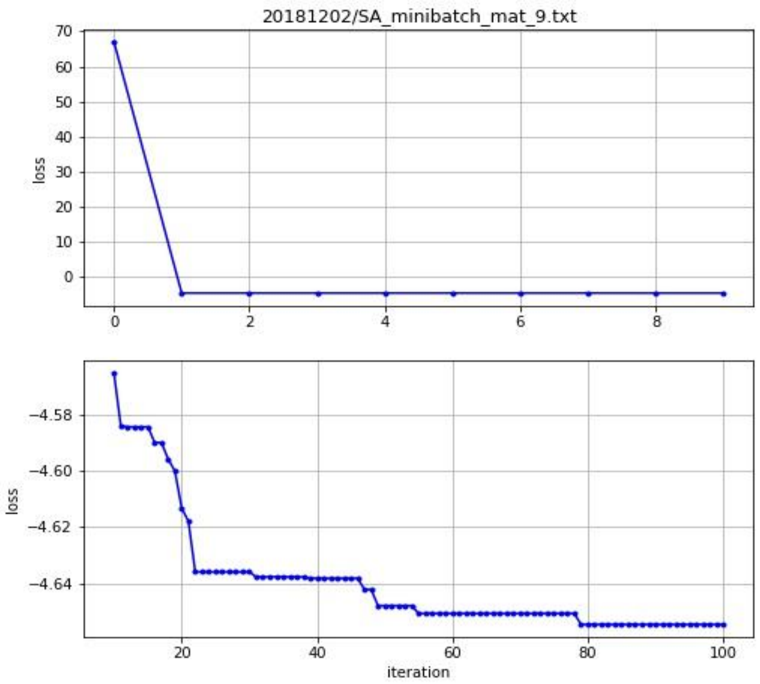
- Gradient Method
  - 测试集
  - 泛化较好

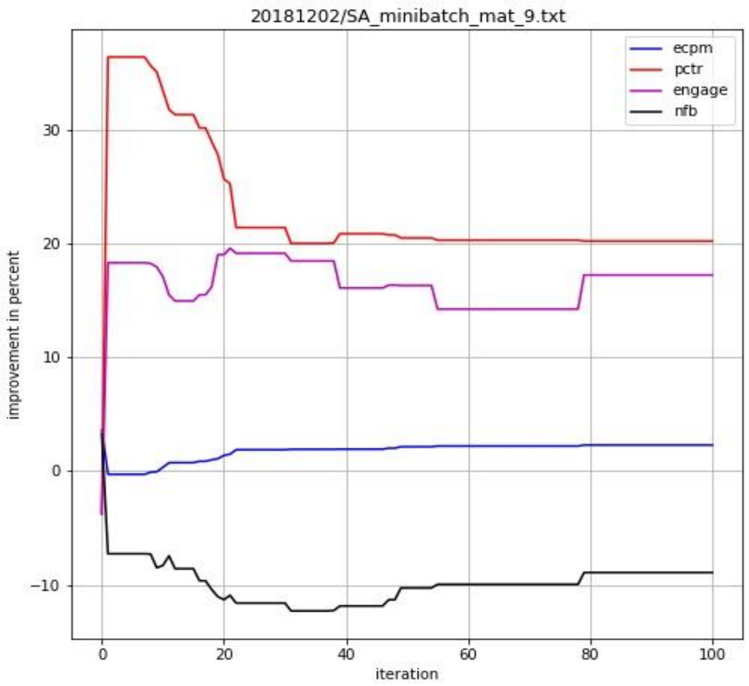




一期算法 – 离线结果

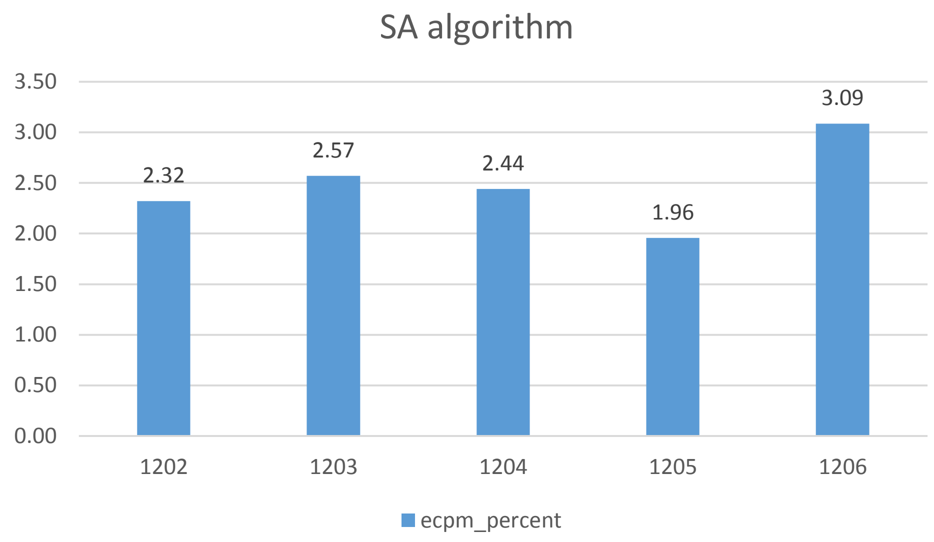
- Simulated Annealing
  - 训练集
  - 不适合梯度法采用的mini-batch方式，当constraint不满足时，可能会来回振荡
  - 对于不同的batch，可以串行优化

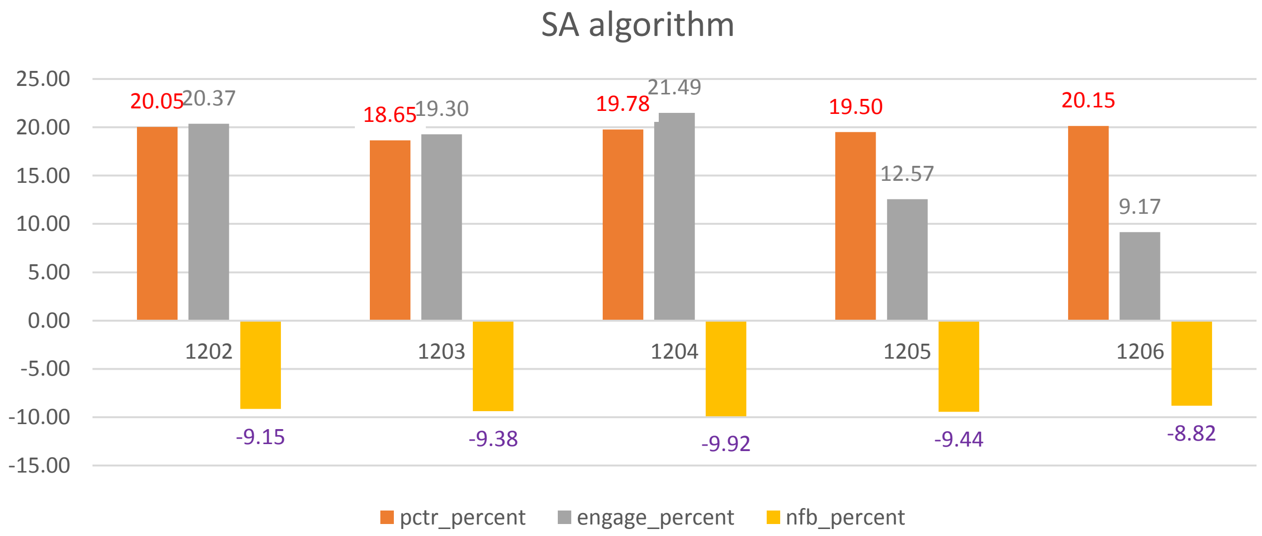




一期算法 – 离线结果

- Simulated Annealing
  - 测试集
  - 比起Gradient Method,
  - ecpm提升较大，pctr, engage rate 提升较小，负反馈率下降较大





腾讯社交广告 Tencent Social Ads 微信广告

一期算法 – 不足

- Gradient method:
  - 通过relaxation来拟合真实objective和constraint
  - relax后的obj和constraints和真实目标存在误差，需要通过校正系数反复调整
  - 原问题relax成非凸优化问题，得到的只是局部最优解

- Simulated Annealing
  - 采用随机搜索的方式
  - 理论上可以找到全局最优，但效率上不及基于凸优化的算法
  - 随着constraint的增多，搜索难度增大

二期算法 – 全局最优解

➢ Question：最理想的情况

- 如何兼顾Gradient Method和Simulated Annealing的优点
- 既可以复用convex problem的求解方式
- 又可以得到全局最优解

➢ Answer: 提出针对该优化问题的凸优化对偶方法

- 转化成constrained linear problem，可以用凸优化方式求解
- dual optimal即为optimal score weight， 全局最优解
- 证明了两个问题等价，首创(original contribution)

[4] Daniel P. Palomar and Mung Chiang, “A tutorial on decomposition methods for network utility maximization," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, 2006.

二期算法 – 全局最优解

$$
\begin{aligned}
\underset{\{p_{k,j}\}}{\text{maximize}} \quad & \frac{1}{K}\sum_k \sum_j ecpm_{k,j} \boxed{p_{k,j}} \\
\text{subject to} \quad & \frac{1}{K}\sum_k \sum_j pctr_{k,j} \boxed{p_{k,j}} \geq pctr_0 \\
& \frac{1}{K}\sum_k \sum_j engage_{k,j} p_{k,j} \geq engage_0 \\
& \frac{1}{K}\sum_k \sum_j nfbr_{k,j} p_{k,j} \leq nfbr_0 \qquad (1) \\
& \frac{1}{K}\sum_k \sum_j \mathbf{1}\{i_k \in \mathbb{C}\} p_{k,j} \geq r_l \\
& \frac{1}{K}\sum_k \sum_j \mathbf{1}\{i_k \in \mathbb{C}\} p_{k,j} \leq r_u \\
& \boxed{\begin{aligned} & p_{k,j} \geq 0, \quad \forall k, j \\ & \sum_j p_{k,j} = 1, \quad \forall k \end{aligned}}
\end{aligned}
$$

P_k,j: 用probability代替indicator

It seems we are optimizing the probabilities
But we are actually optimizing the score weights

Constraints for probabilities

二期算法 – 全局最优解

- 如何将概率和打分联系起来?
- 由对偶问题(Dual problem)证明了以下定理:

Theorem

The optimal solution to Problem (1) has the following structure:

$$p_{u,j} = \begin{cases} 1 & \text{if } j = \arg\max_j S_{k,j} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

with

$$\begin{aligned} S_{k,j} = {} & ecpm_{k,j} + \lambda_1^* pctr_{k,j} + \lambda_2^* engage_{k,j} - \lambda_3^* nfbr_{k,j} \\ & + \lambda_4^* \mathbf{1}\{i_k \in \mathbb{C}\} - \lambda_5^* \mathbf{1}\{i_k \in \mathbb{C}\} \end{aligned} \tag{3}$$

where $\{\lambda_i^*\}_{i=1}^5$ *are dual optimal of Problem (1).*

二期算法 – 全局最优解

- Optimal score 理解

$$S_{k,j} = ecpm_{k,j} + \lambda_1^* pctr_{k,j} + \lambda_2^* engage_{k,j} - \lambda_3^* nfbr_{k,j} + \lambda_4^* \mathbf{1}\{i_k \in \mathbb{C}\} - \lambda_5^* \mathbf{1}\{i_k \in \mathbb{C}\}$$

  - 线性打分，和当前线上的打分结构类似
  - 打分项为ecpm，用户体验项，类目项等
  - 打分权重由原问题(1)中的约束条件决定

- Theorem理解
  - 在原问题(1)中，我们并没有假设线性score，只输入需要达成的目标
  - 证明了最优score结构为线性score

二期算法 – Implementation

- 优化Solver选择: ECOS
- 数据量很大情况下，采用mini-batch方式, 直接求解dual problem

	LP	QP	SOCP	SDP	EXP	MIP
CBC	X					X
GLPK	X					
GLPK_MI	X					X
OSQP	X	X				
CPLEX	X	X	X			X
Elemental	X	X	X			
ECOS	X	X	X		X	
ECOS_BB	X	X	X		X	X
GUROBI	X	X	X			X
MOSEK	X	X	X	X		
CVXOPT	X	X	X	X		
SCS	X	X	X	X	X	

二期算法 – 离线结果

Gradient Method

```
==================================================================
step 20:
weights:  0.593, 1.671, -0.486
total loss: -5.1377
loss of ecpm -5.1377, pctr 0.0000, engage 0.0000, nfb 0.0000, 23ratio 0.0000
ecpm improving: 4.47 percent
pctr, engage, nbf, 23 ratio improving:  12.32, 37.05, -6.03, 2.45  percent
baseline cri: [5.951544, 0.35101882, 0.13010332, 0.29751053, 0.25949445]
current cri: [6.2174144, 0.39426893, 0.17830034, 0.27956596, 0.26584485]
```

Dual Method

```
201  +6.294e+00  +6.294e+00  +6e-04  4e-11  3e-12  3e-14  7e-10  0.9890  2e-01   1  0  0 |  0  0
202  +6.294e+00  +6.294e+00  +2e-05  4e-11  1e-13  1e-15  2e-11  0.9890  2e-02   1  0  0 |  0  0
203  +6.294e+00  +6.294e+00  +3e-07  4e-11  1e-15  2e-17  3e-13  0.9890  8e-04   1  0  0 |  0  0
204  +6.294e+00  +6.294e+00  +2e-07  4e-11  1e-15  1e-17  3e-13  0.8809  9e-01   1  0  0 |  0  0
205  +6.294e+00  +6.294e+00  +3e-09  4e-11  9e-17  2e-19  3e-15  0.9890  6e-04   1  0  0 |  0  0

OPTIMAL (within feastol=4.1e-11, reltol=4.4e-10, abstol=2.8e-09).
Runtime: 95.476416 seconds.

Solver: ECOS, solve time: 94.749 seconds, 205 iterations
Lambda: 0.440,0.024,0.203,0.000
--------------Check Dual----------------------------
Primal objective by dual: 6.293
Primal constraints by dual: -5.311e-06,2.464e-03,2.335e-03,7.100e-02

In [6]: ecpm0
Out[6]: 5.951544

In [7]: linear_prob.primal_obj_by_dual
Out[7]: 6.29330703177402

In [8]: (linear_prob.primal_obj_by_dual-ecpm0)/ecpm0
Out[8]: 0.05742429776492291
```

目标函数：

- 最大化平均ecpm

约束条件：

- pctr，互动率提升10%
- 负反馈率下降10%
- 关注类流量波动不超过30%

规划排期